

## .REM

## IDENTIFICATION

PRODUCT CODE: AC 10958-MC  
PRODUCT TITLE: CVTSBBO TSV05 CONTROLLER LOGIC TEST 2  
AUTHOR: DICK MITCHELL  
DEPARTMENT: COMPUTER SPECIAL SYSTEMS/PPG  
DATE: APRIL 27, 1983

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

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

COPYRIGHT (C) 1982,1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC

POP  
DECUS

UNIBUS  
DELTAPE

MASSBUS

## .REM

## IDENTIFICATION

PRODUCT CODE: AC T0958-MC  
PRODUCT TITLE: CVTSBBO TSV05 CONTROLLER LOGIC TEST 2  
AUTHOR: DICK MITCHELL  
DEPARTMENT: COMPUTER SPECIAL SYSTEMS/PPG  
DATE: APRIL 27, 1983

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

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

COPYRIGHT (C) 1982,1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC

PDP  
DECUS

UNIBUS  
QELTAP

MASSBUS

## TABLE OF CONTENTS

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

## 1.0 GENERAL INFORMATION

### 1.1 PROGRAM ABSTRACT

THIS IS A PDP-11/23 RESIDENT DIAGNOSTIC WHICH CHECKS THE FUNCTIONALITY OF A TSV05 MAGTAPE SUBSYSTEM WHILE CONNECTED TO A PDP-11/23 SYSTEM (Q-BUS). THE PROGRAM PROVIDES ERROR MESSAGES WHICH IDENTIFY FAILING FUNCTIONS THAT AID IN THE REPAIR OF THE DEVICE. THIS DIAGNOSTIC CONSIST OF TWELVE TEST. TEST 1-9 ARE EXECUTED IN SEQUENCE. TEST 10-12 ARE STAND ALONE TEST WHICH ALLOW THE OPERATOR TO PERFORM SPECIFIC FUNCTIONAL TEST ON SCOPE LOOPS ON CERTAIN FUNCTIONS.

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

PDP-11/23 PROCESSOR AND MEMORY  
CAUTION: DIAGNOSTIC REQUIRES 32K WORDS OF MEMORY  
(28K USEABLE AND 4K RESERVED FOR I/O PAGE)  
TSV05 MAGTAPE SUBSYSTEM (DRIVE AND CONTROLLER)  
CONSOLE TERMINAL  
PDP-11 DIAGNOSTIC SUPERVISOR (MSAAA.SYS VERSION 34 OR LATER)  
PDP-11 DIAGNOSTIC LOADER/MONITOR (XXDP+)

### 1.3 RELATED DOCUMENTS AND STANDARDS

DIGITAL EQUIPMENT CORPORATION DOCUMENTS:

1. CHQUS XXDP+ USERS GUIDE; DOCUMENT NUMBER AC-F348E-MC  
DATE: 14 JULY 1980.
2. TSV05 TRANSPORT SUBSYSTEM USER'S GUIDE; DOCUMENT NUMBER EK-TSV05-UG-001  
DATE: AUGUST 1982
3. TSV05 TRANSPORT SUBSYSTEM TECHNICAL MANUAL; DOCUMENT NUMBER EK-TSV05-TM-001  
DATE: AUGUST 1982
4. TSV05 TRANSPORT SUBSYSTEM INSTALLATION MANUAL; DOCUMENT NUMBER EK-TSV05-IN-001  
DATE: AUGUST 1982

### 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

FUNCTIONAL PDP 11/23 CENTRAL PROCESSOR AND MEMORY  
FUNCTIONAL CONSOLE TERMINAL  
FUNCTIONAL STANDALONE DIAGNOSTIC SUPERVISOR

## FUNCTIONAL DIAGNOSTIC LOADER/MONITOR (XXDP+)

## 1.5 ASSUMPTIONS

ALL HARDWARE EXCEPT THE HARDWARE UNDER TEST IS ASSUMED TO WORK PROPERLY OR FALSE ERRORS CAN BE REPORTED.  
THE TAPE BEING USED ON THE TS05 TRANSPORT IS A KNOWN GOOD REEL OF TAPE.  
CVTSAA HAS RUN SUCCESSFULLY.

## 2.0 OPERATING INSTRUCTIONS

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

## 2.1 COMMANDS

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

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

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

## 2.1.1 OPERATOR COMMANDS

THE TS05 DIAGNOSTIC IS A PDP 11/23 DIAGNOSTIC SUPERVISOR COMPATIBLE PROGRAM. ALL LOADING AND RUNTIME INSTRUCTIONS CAN BE REFERENCED IN THE CHQUS XXDP+ USERS GUIDE, DOCUMENT NUMBER AQ-F348E-MC. THE USER ENTRY IS IN QUOTES.

## BOOT THE DIAGNOSTIC MEDIA

OR VTSB??  
DIAG. RUN-TIME SERVICES REV D. APR 79  
CVTSB-A-0

\*\*\*\*\*TSV05 LOGIC DIAGNOSTIC\*\*\*\*\*  
 UNIT IS TSV05  
 >DR

## 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:DDDDU  | 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 "/YES:1-5" INSTEAD OF "/TESTS:1-5".

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

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

## 2.3 FLAGS

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

| FLAG | EFFECT                                                                                                    |
|------|-----------------------------------------------------------------------------------------------------------|
| HOE  | HAUT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE                                      |
| LOE  | LOOP ON ERROR                                                                                             |
| IER* | INHIBIT ALL ERROR REPORTS                                                                                 |
| IBR* | 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 MACROS)                                            |
| PRI  | DIRECT MESSAGES TO LINE PRINTER                                                                           |
| PNT  | PRINT TEST NUMBER AS TEST EXECUTES                                                                        |
| BOE  | "BELL" ON ERROR                                                                                           |
| UAM  | UNATTENDED MODE (NO MANUAL INTERVENTION)                                                                  |
| ISR  | INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)    |
| IDR  | INHIBIT PROGRAM DROPPING OF UNITS                                                                         |
| ADR  | EXECUTE AUTODROP CODE                                                                                     |
| LOT  | LOOP ON TEST                                                                                              |

\*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

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

```
/FLAGS:LOE:IER:BOE
```

## 2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 14 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).

AFTER INITIAL STARTING OF THE PROGRAM (START COMMAND TO THE DIAGNOSTIC SUPERVISOR), THE PROGRAM WILL ISSUE THE "CHANGE HW?" QUESTION TO ASK IF THE HARDWARE PARAMETERS ARE TO BE CHANGED (BY THE OPERATOR).

ON A "N" (NO) RESPONSE TO THE "CHANGE HW?" QUESTION, THE DIAGNOSTIC WILL RUN USING THE DEFAULT VALUES FOR ALL QUESTIONS. THE DEFAULT ADDRESS AND VECTOR ARE:

TSBA/TSDB = 172520, VECTOR = 224

ON A "Y" (YES) RESPONSE TO THE QUESTION, THE FOLLOWING QUESTIONS WILL THEN BE ASKED TO ALLOW THE OPERATOR TO SELECT THE UNITS TO BE TESTED. A VALUE, IF PRESENT, LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN IF ONLY A CARRIAGE RETURN IS TYPED AS A RESPONSE. A "(D)" IN A QUESTION INDICATES THAT A DECIMAL NUMBER IS REQUIRED AS A RESPONSE. AN "(O)" INDICATES AN OCTAL NUMBER IS BEING SOLICITED. AN "(L)" INDICATES THAT A LOGICAL RESPONSE IS TO BE MADE: "Y" FOR YES, "N" FOR NO.

\* UNITS (D) ? <ENTER THE NUMBER OF M7196 CONTROLLERS  
PRESENT TO BE TESTED>

UNIT 0

DEVICE ADDRESS (O) 172520 ? <ENTER THE ADDRESS OF THE  
TSBA/TSDB REGISTER>

VECTOR (O) 224 ? <ENTER ADDRESS OF INTERRUPT  
VECTOR>

THE ADDRESS AND VECTOR QUESTIONS WILL BE ASKED FOR EACH OF THE NUMBER OF UNITS (CONTROLLERS) SPECIFIED IN THE "\* UNITS?" QUESTION. LOGICAL UNIT NUMBERS ARE ASSIGNED IN ORDER, BEGINNING AT 0. UP TO FOUR UNITS CAN BE SELECTED FOR TESTING AS FOLLOWS:  
UP TO 4 TSV05 CONTROLLERS PER 11/23 AND UP TO 2 DRIVES PER CONTROLLER

## 2.5 SOFTWARE QUESTIONS

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

THE FOLLOWING QUESTIONS ARE ASKED ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXIBILITY IN THE WAY THE PROGRAM BEHAVES.

CHANGE SW (L) ? <TYPE Y TO CAUSE THE FOLLOWING  
QUESTIONS TO BE ASKED>

INHIBIT ITERATIONS (L) N ? <TYPE "Y" TO PREVENT MULTIPLE  
ITERATIONS OF CERTAIN TESTS.  
THIS CAUSES EACH TEST PASS TO  
RUN AS QUICKLY AS POSSIBLE.  
ONLY QUICK RUNNING LOGIC  
TESTS USE MULTIPLE  
ITERATIONS.>

## 2.6 EXTENDED P-TABLE DIALOGUE

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

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

\* UNITS (0) ? 8<CR>

UNIT 1

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 0<CR>

Q-FACTOR (0) 0 ? 1<CR>

UNIT 2

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 1<CR>

Q-FACTOR (0) 1 ? 0<CR>

UNIT 3

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 2<CR>

Q-FACTOR (0) 0 ? <CR>

UNIT 4

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 3<CR>

Q-FACTOR (0) 0 ? <CR>

UNIT 5

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 4<CR>

Q-FACTOR (0) 0 ? <CR>

UNIT 6

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 5<CR>

Q-FACTOR (0) 0 ? <CR>

UNIT 7

CSR ADDRESS (0) ? 160000<CR>

SUB-DEVICE # (0) ? 6<CR>

Q-FACTOR (0) 0 ? 1<CR>

UNIT 8

CSR ADDRESS (0) 160000<CR>

```
SUB-DEVICE # (0) ? 7<CR>
Q-FACTOR (0) 1 ? <CR>
```

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

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

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

```
# UNITS (0) ? 8<CR>

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

UNIT 3
CSR ADDRESS (0) ? 160000<CR>
SUB-DEVICE # (0) ? 2-5<CR>
Q-FACTOR (0) 0 ? 0<CR>

UNIT 7
CSR ADDRESS (0) ? 160000<CR>
SUB-DEVICE # (0) ? 6,7<CR>
Q-FACTOR (0) 0 ? 1<CR>
```

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

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

```
# UNITS (0) ? 8<CR>

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

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

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

TO START-UP THIS PROGRAM:

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

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

## 3.0 ERROR INFORMATION

### 3.1 TYPES OF ERROR MESSAGES

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

NAME TYPE NUMBER ON UNIT NUMBER TST NUMBER PC:XXXXXX  
ERROR MESSAGE

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

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

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

## 3.2 SPECIFIC ERROR MESSAGES

BELOW ARE SAMPLE ERROR MESSAGES. EACH ERROR MESSAGE REPRESENTS DIFFERENT TYPES OF ERRORS DETECTED BY THIS DIAGNOSTIC.

## ERROR MESSAGE EXAMPLE 1

THIS ERROR IS INDICATIVE OF AN INCORRECT REGISTER OR STATUS WORD RETURNED TO THE DIAGNOSTIC. THE FIRST PART DEFINES THE TEST FUNCTION AND UNIT THAT FAILED. THE SECOND PART PROVIDES THE REGISTER BITS AND THEIR MNEMONICS FOR THE INCORRECT REGISTER OR STATUS WORDS. THE THIRD PART IS THE EXPECTED AND RECEIVED DATA.

TST: 016 FIFO EXERCISER TEST  
CVTSB HRD ERR 01610 ON UNIT 00 TST 016 SUB 002 PC: 040624  
FIFO STATUS (IN WORD 9) INCORRECT AFTER WRITE FIFO

TAPE BUS SIGNALS IN WORD #8: - DESIGNATOR <BIT #>  
PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>  
IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>  
IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>  
TAPE BUS SIGNALS IN WORD #9:  
DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>  
MESSAGE BUFFER ADDRESS = 047352  
MESSAGE BUFFER CONTENTS:

|         |              |              |             |
|---------|--------------|--------------|-------------|
| WORD #0 | EXPD: 100020 | RECV: 100020 | XOR: 000000 |
| WORD #1 | EXPD: 000012 | RECV: 000012 | XOR: 000000 |
| WORD #2 | EXPD: 000000 | RECV: 000000 | XOR: 000000 |
| WORD #3 | EXPD: 000010 | RECV: 000010 | XOR: 000000 |
| WORD #4 | EXPD: 000000 | RECV: 000000 | XOR: 000000 |
| WORD #5 | EXPD: 000000 | RECV: 000000 | XOR: 000000 |
| WORD #6 | EXPD: 000000 | RECV: 000000 | XOR: 000000 |
| WORD #7 | EXPD: 000000 | RECV: 000000 | XOR: 000000 |
| WORD #8 | EXPD: 070217 | RECV: 070217 | XOR: 000000 |
| WORD #9 | EXPD: 000074 | RECV: 000034 | XOR: 000040 |

## ERROR MESSAGE EXAMPLE 2

THIS ERROR SHOWS A FATAL FUNCTION ERROR FROM THE TAPE DRIVE, IN THIS INSTANCE A UNRECOVERABLE ERROR OCCURED WHICH INDICATES THAT THE CONTROLLER MAY BE DEFECTIVE.

CVTSB HRD ERR 00159 ON UNIT 00 TST 001 SUB 005 PC: 026202  
TSSR NOT CORRECT AFTER SPACE RECORDS COMMAND  
TSSR = 100214  
TSSR BITS SET: SC, SSR  
TERMINATION CLASS CODE = UNRECOVERABLE ERROR  
PACKET ADDRESS = 026420  
PACKET WORD # = 140010  
PACKET WORD # = 000010  
PACKET WORD # = 000000  
PACKET WORD # = 000024

## ERROR MESSAGE EXAMPLE 3

THIS ERROR SHOWS THAT THE MOTION BIT DID NOT GET SET WHILE DOING A REWIND WITH EXTENDED FEATURES MODE ENABLED.

CVTSB HRD ERR 00121 ON UNIT 00 TST 001 SUB 002 PC: 023306  
MOT BIT (XST0) NOT SET DURING REWIND (EXTENDED FEATURES MODE)  
EXPD: 000312 RECV: 000112 XOR: 000200

## 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.

## SUCCESSFUL RUN EXAMPLE (PDP-11/23)

DR>STA/FLA:PNT:HOE:UAM

UNITS (D) ? 1

UNIT 0

DEVICE ADDRESS (0) 172520 ? <CR>

VECTOR (0) 224 ? <CR>

CHANGE SW (L) ? N<CR>

THE ABOVE COMMAND WILL START THE DIAGNOSTIC. THE COMMAND HAS THREE SWITCHES ON WHICH ARE "PRINT EACH TEST NBR AS EXECUTED", "HALT ON ERROR" AND "RUN IN UNATTENDED MODE".

NOTE: THE UAM FLAG SHOULD BE USED TO PREVENT TEST 10 12 FROM BEING EXECUTED UNLESS THE OPERATOR WANTS THESE SPECIFIC TEST.

TST: 001 INITIALIZE #3 TEST  
TST: 002 BASIC WRITE SUBSYSTEM MEMORY TEST  
TST: 003 DMA MEMORY ADDRESSING TEST  
TST: 004 RAM EXERCISER TEST  
TST: 005 FIFO EXERCISER TEST  
TST: 006 STATIC TRANSPORT BUS CHECK  
TST: 007 TRANSPORT BUS INTERFACE CHECK VIA LOOPBACK TEST  
TST: 008 READ/WRITE DATA PARITY CHECK TEST  
TST: 009 MISCELLANEOUS LOGIC CHECKS TEST  
TST: 010 STAND-ALONE MANUAL INTERVENTION NOT EXECUTED TEST  
TST: 011 STAND-ALONE CONFIGURATION TIMEOUT NOT EXECUTED TEST  
TST: 012 STAND-ALONE SCOPE LOOPS NOT EXECUTED TEST

0 ERRORS

NOTE: THE DIAGNOSTIC WILL RUN CONTINUOUSLY UNLESS A PASS LIMIT HAS BEEN SPECIFIED WITH THE "/PASS:" SWITCH.

#### PROGRAM RUN TIMES

THE AVERAGE RUN TIMES OF THE PROGRAM ARE LISTED BELOW. THESE FIGURES ARE TO BE USED AS A GUIDE. THE TIMING WAS DONE ON A PDP-11/23 PROCESSOR WITH A LA-34 CONSOLE.

THE PROGRAM RUNS IN TWO MODES, NO ITERATIONS AND DEFAULT MODE. IN THE NO ITERATIONS MODE, EACH TEST IS RUN ONCE, WITH NO ITERATIONS. IN THE DEFAULT MODE EACH TEST IS REPEATED BY THE NUMBER OF TIMES INDICATED BY THE ITERATION COUNT. NO ITERATIONS MODE IS SELECTED BY ANSWERING THE INHIBIT ITERATIONS QUESTION WITH A "Y" (YES).

| TEST<br>NUMBER | N/I<br>SECS. | ITER<br>SECS | DEF<br>SECS. |
|----------------|--------------|--------------|--------------|
| 1              | 15           | 50           | 35           |
| 2              | 1            | 6            | 5            |
| 3              | 1            | 1            | 0            |
| 4              | 110          | 540          | 430          |
| 5              | 1            | 10           | 9            |
| 6              | 10           | 120          | 110          |
| 7              | 1            | 3            | 2            |
| 8              | 15           | 15           | 12           |
| 9              | 17           | 17           | 13           |

THE TIMES REQUIRED TO RUN TESTS 1 THROUGH 9 IN ONE COMMAND:

|         |                    |
|---------|--------------------|
| G.V.    | 2 MINS 19 SECONDS  |
| DEFAULT | 11 MINS 35 SECONDS |

#### 5.0 DEVICE INFORMATION TABLES

WHENEVER THE PROGRAM IS STARTED, VIA THE STA(PT) COMMAND, THE SUPERVISOR REQUESTS THE FOLLOWING P-TABLES PARAMETER CHANGES:

CHANGE HW (L) ?

# UNITS (D) ? <ENTER THE NUMBER OF M7196 CONTROLLERS  
PRESENT TO BE TESTED>

UNIT 0

DEVICE ADDRESS (O) 172520 ? <ENTER THE ADDRESS OF THE  
TSBA/TSDB REGISTER>

VECTOR (O) 224 ? <ENTER ADDRESS OF INTERRUPT  
VECTOR>

THE ADDRESS AND VECTOR QUESTIONS WILL BE ASKED FOR EACH OF THE NUMBER OF

UNITS (CONTROLLERS) SPECIFIED IN THE "A UNITS?" QUESTION. LOGICAL UNIT NUMBERS ARE ASSIGNED IN ORDER, BEGINNING AT 0. UP TO FOUR UNITS CAN BE SELECTED FOR TESTING.

IN ADDITION, ON A START, RESTART OR CONTINUE THE SUPERVISOR REQUESTS CHANGES TO THE SOFTWARE OPERATING PARAMETERS, AS FOLLOWS:

CHANGE SW (L) ?

#### 6.0 TEST SUMMARIES

##### TEST 1: INITIALIZE AFTER WRITE CHARACTERISTICS

###### TEST DESCRIPTION:

THIS TEST VERIFIES THAT A HARDWARE INITIALIZE COMMAND INVOKED AFTER A WRITE CHARACTERISTICS COMMAND SETS UP THE COMMAND, MESSAGE AND CHARACTERISTIC IMAGE BLOCKS IN THE CONTROLLER RAM CORRECTLY.

##### TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND

THIS TEST VERIFIES THAT THE WRITE SUBSYSTEM MEMORY COMMAND WITH A BSEL0 SELECT CODE OF 0 (NO-OP) EXECUTES CORRECTLY. IT ALSO VERIFIES THAT A WRITE SUBSYSTEM MEMORY COMMAND WITH A NON-ZERO MODE FIELD IS REJECTED. THE TEST FURTHER VERIFIES MICROPROGRAM COMMAND DECODING AND HANDLING SEQUENCES.

##### TEST 3: DMA MEMORY ADDRESSING

THIS TEST VERIFIES THAT THE CONTROLLER CAN PROPERLY ADDRESS AND ACCESS ALL AVAILABLE CPU MEMORY (OTHER THAN THAT OCCUPIED BY THE DIAGNOSTIC AND DIAGNOSTIC SUPERVISOR CODE) FOR BOTH READING (DATI) AND WRITING (DATO). VERIFIED ARE THE LSI-11 BUS DRIVERS FOR ALL AVAILABLE ADDRESS LINES. UP TO THIS POINT ONLY 16 BITS HAVE BEEN USED FOR DMA TRANSFERS.

\*\*\*\*\*  
CAUTION

THE LSI BUS DRIVERS FOR ALL AVAILABLE ADDRESS LINES ARE ONLY CHECKED WHEN RUNNING ON A 11/23B SYSTEM WITH MORE THAN 128K WORDS OF MEMORY!

\*\*\*\*\*

##### TEST 4: RAM EXERCISER TEST

THIS TEST USES THE READ AND WRITE RAM (BOTH SINGLE AND 256 LOCATIONS) SELECT CODES OF THE WRITE SUBSYSTEM MEMORY COMMAND TO EXERCISE THE CONTROLLER'S RAM MEMORY AND DMA LOGIC

TEST 5: EXTENDED FEATURES SWITCH AND TIMERS A,B

TEST DESCRIPTION:

THIS TEST VERIFIES THE INVERT EXTENDED FEATURES FUNCTION CAN LOGICALLY INVERT THE EXTENDED FEATURES SWITCH AND THAT THE INTERNAL TIMERS A AND B OPERATE CORRECTLY.

TEST 6: FIFO EXERCISER

TEST DESCRIPTION:

THIS TEST USES THE WRITE SUBSYSTEM MEMORY COMMAND TO VERIFY THE CONTROLLER'S FIFO AND ASSOCIATED STATUS AND CONTROL LOGIC.

TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

TEST DESCRIPTION:

WRITE TO ISSR REGISTER TO SOFT INITIALIZE THE CONTROLLER  
DO WRITE CHARACTERISTICS TO CHECK FOR EXTENDED FEATURES SWITCH  
IF EXTENDED FEATURES HARDWARE SWITCH CLEAR THEN:  
DO WRITE SUBSYSTEM WRITE MISCELLANEOUS TO SET EXTENDED FEATURES.  
DO WRITE CHARACTERISTICS TO SELECT RESERVED UNIT 7  
DO A WRITE SUBSYSTEM READ STATUS  
IF ANY TRANSPORT INTERFACE SIGNALS ARE ASSERTED THEN PRINT ERROR

TEST 8: TRANSPORT BUS INTERFACE LOOPBACK TEST

TEST DESCRIPTION:

THIS TEST VERIFIES THE CONTROLLER'S TRANSPORT BUS DRIVERS, RECEIVERS, AND SIGNAL LOOPBACK LOGIC. NOTE THAT THE STATIC TRANSPORT BUS TEST MUST HAVE RUN CORRECTLY FOR THIS TEST TO PROVIDE MEANINGFUL RESULTS.

TEST 9: READ/WRITE DATA PARITY TEST

TEST DESCRIPTION:

THIS TEST VERIFIES THAT THE WRITE DATA PARITY GENERATOR AND THE READ DATA PARITY CHECKER OPERATE PROPERLY. THE TRANSPORT BUS SIGNAL LOOPBACK MODE IS ENABLED AND A SET WRONG PARITY FUNCTION IS EXECUTED. THEN VARIOUS WRITE SUBSYSTEM MEMORY FUNCTIONS ARE PERFORMED TO WRITE DATA TO AND FROM THE FIFO IN LOOPBACK MODE. THE PROGRAM THEN CHECKS TO INSURE A READ DATA PARITY ERROR OCCURRED.  
A RESET FIFO IS DONE AND THE READ DATA PARITY

ERROR BIT IS AGAIN TESTED TO INSURE IT CLEARED.  
FINALLY A CLEAR WRONG PARITY FUNCTION IS DONE  
AND IT IS VERIFIED THE DATA WORD CAN PASS IN LOOPBACK  
MODE WITHOUT SETTING READ DATA PARITY ERROR.

#### TEST 10: MANUAL INTERVENTION

THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A "TEST") THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF THE SUBSYSTEM THAT CANNOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE. THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR ARE BY TYPING <CTRL-C> OR SELECTING CODE 6. SELECTION CODES AND SUBROUTINES ARE:

| CODE | ROUTINE                           |
|------|-----------------------------------|
| 0    | HELP. PRINTS THIS MENU.           |
| 1    | TURN ON ALL M7196 LED INDICATORS  |
| 2    | TURN OFF ALL M7196 LED INDICATORS |
| 3    | OFFLINE/ONLINE ATTENTION TEST     |
| 4    | WRITE-PROTECT TEST                |
| 5    | PRINT EXTENDED TRANSPORT STATUS   |
| 6    | EXIT (RETURN TO SUPERVISOR)       |

#### TEST 11: CONFIGURATION TYPEOUT

THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL THE CONFIGURATION OF THE M7196 MODULE AND TSV05 SUBSYSTEM. SPECIFICALLY, THE FOLLOWING INFORMATION IS PRESENTED:

- 1.0 STATE OF THE EXTENDED FEATURES SWITCH ON THE M7196: ON (EXTENDED FEATURES ENABLED) OR OFF (EXTENDED FEATURES DISABLED),
- 2.0 STATE OF THE BUFFERING ENABLE SWITCH ON THE M7196: ON (BUFFERING ENABLED) OR OFF (BUFFERING DISABLED),
- 3.0 MICROCODE REVISION LEVEL OF THE M7196,
- 4.0 NUMBER OF TAPE TRANSPORTS CONNECTED TO THE CONTROLLER,
- 5.0 UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED) OF EACH CONNECTED TRANSPORT. IN ADDITION, THE PROGRAM WILL INDICATE, FOR EACH ON-LINE TRANSPORT, WHETHER OR NOT IT IS EQUIPPED WITH THE EXTENDED TAPE STATUS READOUT FEATURE.

#### TEST 12: SCOPE LOOPS

THIS IS A STANDALONE ROUTINE PROVIDING A NUMBER OF TIGHT "SCOPE LOOPS" USEFUL FOR DEBUGGING BASIC REGISTER ACCESS PROBLEMS WITH THE M7196 MODULE. THESE SCOPE LOOPS CAN BE USED WHEN THE NORMAL "LOOP ON ERROR" OR "LOOP ON TEST (SUBTEST)" FACILITIES DON'T

SEEM TO ALLOW THE OPERATOR TO ZERO IN A PROBLEM IN THE EARLY TESTS (I.E. THE HARDWARE MAY NOT BE RESPONDING TO A REGISTER ACCESS, CAUSING A BUS ERROR TRAP, EVEN THOUGH THE DEVICE ADDRESS SELECTED BY THE PROGRAM MATCHES THE CONFIGURATION SET UP IN THE HARDWARE DIP SWITCHES). THE FOLLOWING MENU OF SCOPE LOOPS ARE AVAILABLE:

| CODE | SCOPE LOOP                                                                                              |
|------|---------------------------------------------------------------------------------------------------------|
| 0    | HELP. PRINT THIS MENU.                                                                                  |
| 1    | TSBA READ ACCESS                                                                                        |
| 2    | TSSR READ ACCESS                                                                                        |
| 3    | INITIALIZE (TSSR WRITE ACCESS)                                                                          |
| 4    | TSDB HIGH BYTE WRITE ACCESS                                                                             |
| 5    | TSDB LOW BYTE WRITE ACCESS                                                                              |
| 6    | TSDB MAINTENANCE-MODE WORD WRITE ACCESS                                                                 |
| 7    | TSDBX (TSSR HIGH BYTE) WRITE ACCESS<br>(EXTENDED FEATURES SWITCH MUST BE ON<br>TO USE SELECTION CODE 7) |
| 8    | EXIT (RETURN TO SUPERVISOR)                                                                             |

FOR SCOPE LOOPS THAT WRITE INTO REGISTERS, THE PROGRAM PROMPTS THE OPERATOR FOR THE DATA TO BE WRITTEN, LIMITS ON THE DATA PATTERNS ARE 0-377, TYPING <RETURN> CAUSES AN EXIT FROM THE SCOPE LOOP BACK TO MENU LEVEL.

#### 7.0 MAINTENANCE HISTORY

REVISION A - MARCH 1982

REVISION B - APRIL 1983

MODIFIED THE DIAGNOSTIC TO HANDLE 11/23A'S  
WITH MORE THAN 256KB OF MEMORY. CHANGED  
TEST 3 SUBTEST 3 SO IT WON'T TRY TO CREATE  
NON-EXISTANT MEMORY ADDRESS (NXM).

F2

TSV2 PROGRAM HEADER MACRO M1113 06-FEB-84 17:14

SEQ 018

```

2          .TITLE  TSV2 - PROGRAM HEADER
3          .SBTTL  PROGRAM HEADER
4
10         .MCALL  SVC
11 000000   SVC                      ; INITIALIZE SUPERVISOR MACROS
12         .ENABLE LC
13         .INLIST BEX,CND
19 000000   .FNABL  ABS,AMA
20         .-2000
21 002000   BGNMOD  TSV2
22         TSV2::
23
24         ;++
25         ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
26         ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
27         ;--
28 002000   POINTER BGNJW,BGNSFT,BGNAU,BGNDU,BGNRPT
29 002000   HEADER  CVTSB,B,0,655,,0
          L$NAME:: ;DIAGNOSTIC NAME
          .ASCII  /C/
          .ASCII  /V/
          .ASCII  /T/
          .ASCII  /S/
          .ASCII  /B/
          .BYTE   0
          .BYTE   1
          .BYTE   0
          L$REV:: ;REVISION LEVEL
          .ASCII  /B/
          L$DEPO:: ;0
          .ASCII  /O/
          L$UNIT:: ;NUMBER OF UNITS
          .WORD   0
          L$TIML:: ;LONGEST TEST TIME
          .WORD   655.
          L$HPCP:: ;POINTER TO H.W. QUES.
          .WORD   L$HARD
          L$SPCP:: ;POINTER TO S.W. QUES.
          .WORD   L$SOFT
          L$HPTP:: ;PTR. TO DEF. H.W. PTABLE
          .WORD   L$HW
          L$SPTP:: ;PTR. TO S.W. PTABLE
          .WORD   L$SW
          L$LADP:: ;DIAG. END ADDRESS
          .WORD   L$LAST
          L$STA:: ;RESERVED FOR APT STATS
          .WORD   0
          L$CO:: ;
          .WORD   0
          L$DTYP:: ;DIAGNOSTIC TYPE
          .WORD   0
          L$AF:: ;APT EXPANSION
          .WORD   0
          L$DTP:: ;PTR. TO DISPATCH TABLE
          .WORD   L$DISPATCH
          L$PRIO:: ;DIAGNOSTIC RUN PRIORITY

```

G2

TSV2 - PROGRAM HEADER MACRO M1113 06-FEB-84 17:14  
PROGRAM HEADER

SEQ 019

|        |        |           |       |             |                                  |
|--------|--------|-----------|-------|-------------|----------------------------------|
| 002042 | 000000 |           | .WORD | 0           |                                  |
| 002044 |        | L\$ENVI:: |       |             | ;FLAGS DESCRIBE HOW IT WAS SETUP |
| 002044 | 000000 |           | .WORD | 0           |                                  |
| 002046 |        | L\$EXP1:: |       |             | ;EXPANSION WORD                  |
| 002046 | 000000 |           | .WORD | 0           |                                  |
| 002050 |        | L\$MREV:: |       |             | ;SVC REV AND EDIT #              |
| 002050 | 003    |           | .BYTE | C\$REVISION |                                  |
| 002051 | 003    |           | .BYTE | C\$EDIT     |                                  |
| 002052 |        | L\$EF::   |       |             | ;DIAG. EVENT FLAGS               |
| 002052 | 000000 |           | .WORD | 0           |                                  |
| 002054 | 000000 |           | .WORD | 0           |                                  |
| 002056 |        | L\$SPC::  |       |             |                                  |
| 002056 | 000000 |           | .WORD | 0           |                                  |
| 002060 |        | L\$DEVP:: |       |             | ; POINTER TO DEVICE TYPE LIST    |
| 002060 | 003402 |           | .WORD | L\$DVTYP    |                                  |
| 002062 |        | L\$REPP:: |       |             | ;PTR. TO REPORT CODE             |
| 002062 | 022664 |           | .WORD | L\$RPT      |                                  |
| 002064 |        | L\$EXP4:: |       |             |                                  |
| 002064 | 000000 |           | .WORD | 0           |                                  |
| 002066 |        | L\$EXP5:: |       |             |                                  |
| 002066 | 000000 |           | .WORD | 0           |                                  |
| 002070 |        | L\$AUT::  |       |             | ;PTR. TO ADD UNIT CODE           |
| 002070 | 022352 |           | .WORD | L\$AU       |                                  |
| 002072 |        | L\$DUT::  |       |             | ;PTR. TO DROP UNIT CODE          |
| 002072 | 022450 |           | .WORD | L\$DU       |                                  |
| 002074 |        | L\$LUN::  |       |             | ;LUN FOR EXERCISERS TO FILL      |
| 002074 | 000000 |           | .WORD | 0           |                                  |
| 002076 |        | L\$DESP:: |       |             | ;POINTER TO DIAG. DESCRIPTION    |
| 002076 | 003410 |           | .WORD | L\$DESC     |                                  |
| 002100 |        | L\$LOAD:: |       |             | ;GENERATE SPECIAL AUTOLOAD EMT   |
| 002100 | 104035 |           | EMT   | E\$LOAD     |                                  |
| 002102 |        | L\$ETP::  |       |             | ;POINTER TO ERRIBL               |
| 002102 | 000000 |           | .WORD | 0           |                                  |
| 002104 |        | L\$ICP::  |       |             | ;PTR. TO INIT CODE               |
| 002104 | 021556 |           | .WORD | L\$INIT     |                                  |
| 002106 |        | L\$CCP::  |       |             | ;PTR. TO CLEAN-UP CODE           |
| 002106 | 022636 |           | .WORD | L\$CLEAN    |                                  |
| 002110 |        | L\$ACP::  |       |             | ;PTR. TO AUTO CODE               |
| 002110 | 022556 |           | .WORD | L\$AUTO     |                                  |
| 002112 |        | L\$PRT::  |       |             | ;PTR. TO PROTECT TABLE           |
| 002112 | 021546 |           | .WORD | L\$PROT     |                                  |
| 002114 |        | L\$TEST:: |       |             | ;TEST NUMBER                     |
| 002114 | 000000 |           | .WORD | 0           |                                  |
| 002116 |        | L\$DLY::  |       |             | ;DELAY COUNT                     |
| 002116 | 000000 |           | .WORD | 0           |                                  |
| 002120 |        | L\$HIME:: |       |             | ;PTR. TO HIGH MEM                |
| 002120 | 000000 |           | .WORD | 0           |                                  |

H2

TSV2 - PROGRAM HEADER    MACRO M1113    06-FEB-84 17:14  
DISPATCH TABLE

SEQ 020

```
31                                     .SBTTL  DISPATCH TABLE
32
33                                     ;++
34                                     ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
35                                     ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
36                                     ;--
37
38 002122                                     DISPATCH 1
    002122 000014                                     .WORD 1
    002124                                     L$DISPATCH:
    002124 023446                                     .WORD T1
    002126 024426                                     .WORD T2
    002130 026420                                     .WORD T3
    002132 032014                                     .WORD T4
    002134 034604                                     .WORD T5
    002136 040376                                     .WORD T6
    002140 050510                                     .WORD T7
    002142 051770                                     .WORD T8
    002144 062616                                     .WORD T9
    002146 066666                                     .WORD T10
    002150 074530                                     .WORD T11
    002152 077702                                     .WORD T12
```

TSV2 - PROGRAM HEADER MACRO M1113 06-FEB-84 17:14  
 DEFAULT HARDWARE P-TABLE

SEQ 021

```

40          .SBTTL  DEFAULT HARDWARE P-TABLE
41
42          ;**
43          ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
44          ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
45          ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
46          ;--
47 002154      BGNHW    DFPTBL      ;DEFAULT HARD-P-TABLE
          002154      .WORD    L10000-L.$HW/2
          002156
          002156      L.$HW::
          DFPTBL::
48
49 002156      172520      .WORD    172520      ; 1ST (OF 2) REGISTERS.
50 002160      000224      .WORD    224        ; INTERRUPT VECTOR
51 002162      000200      .WORD    PRI04      ; INTERRUPT PRIORITY.
52 002164      ENDHW
          002164      L10000:

```

J2

TSV2 - PROGRAM HEADER    MACRO M1113    06-FEB-84 17:14  
SOFTWARE P-TABLE

SEQ 022

```

54          .SBTTL  SOFTWARE P-TABLE
55
56          ;**
57          ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
58          ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
59          ;--
60 002164          BGNSW   SFPTBL
      002164      000004      .WORD   L10001-L$SW/2
      002166
      002166      L$SW::
      002166      SFPTBL::
61
62 002166      000000      TRANSTST::      .WORD   0          ; ENABLE TEST OF TRANSPORT(S) IF =1
63 002170      000000      NOITS::          .WORD   0          ; INHIBIT ITERATION OPTION.
64                                     ; ... 0 = ITERATE.
65                                     ; ...NZ = INHIBIT ITERATE.
66 002172      000017      LERRMAX::          .WORD   15.       ; LOCAL (PER TEST) ERROR LIMIT
67 002174      000310      GERRMAX::          .WORD   200.      ; GLOBAL (PER UNIT) ERROR LIMIT
68 002176          ENDGW
      002176      L10001:
69
70 002176          ENDMOD

```

TSV3 - GLOBAL AREAS  
SOFTWARE P-TABLE

MACRO M1113 06-FEB-84 17:14

SEQ 023

7  
8  
13  
19  
20 002176  
002176  
21  
22  
23  
24  
25  
26  
27  
28  
32 002176

```

.TITLE TSV3 - GLOBAL AREAS
.SBTTL GLOBAL EQUATES SECTION

BGNMOD TSV3
TSV3::

.SBTTL GLOBAL EQUATES SECTION

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

EQUALS          ; GET STANDARD EQUATES.

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

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

;
; EVENT FLAG DEFINITIONS
; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
;
000040          EF.START== 32.          ; START COMMAND WAS ISSUED
000037          EF.RESTART== 31.        ; RESTART COMMAND WAS ISSUED
000036          EF.CONTINUE== 30.       ; CONTINUE COMMAND WAS ISSUED
000035          EF.NEW== 29.            ; A NEW PASS HAS BEEN STARTED
000034          EF.PWR== 28.            ; A POWER-FAIL/POWER-UP OCCURRED

;
; PRIORITY LEVEL DEFINITIONS

```

L2

TSV3 - GLOBAL AREAS  
GLOBAL EQUATES SECTION

MACRO M1113 06-FEB-84 17:14

SEQ 024

```

000340      ;
000300      PRI07== 340
000240      PRI06== 300
000200      PRI05== 240
000140      PRI04== 200
000100      PRI03== 140
000040      PRI02== 100
000000      PRI01== 40
            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      ISE==  10000
020000      IER==  20000
040000      LOE==  40000
100000      HOE== 100000

```

```

33
34 002176

```

```

            KT11
.SBTTL      MEMORY MANAGEMENT DEFINITIONS
; *KT11 VECTOR ADDRESS
000250      MMVEC= 250
; *KT11 STATUS REGISTER ADDRESSES
177572      SR0=   177572
177574      SR1=   177574
177576      SR2=   177576
172516      SR3=   172516
; IF NB
; *USER "I" PAGE DESCRIPTOR REGISTERS
UIPDR0= 177600
UIPDR1= 177602
UIPDR2= 177604
UIPDR3= 177606
UIPDR4= 177610
UIPDR5= 177612
UIPDR6= 177614
UIPDR7= 177616
; IF NB
; *USER "D" PAGE DESCRIPTOR REGISTERS
UDPDR0= 177620
UDPDR1= 177622
UDPDR2= 177624
UDPDR3= 177626
UDPDR4= 177630
UDPDR5= 177632
UDPDR6= 177634
UDPDR7= 177636
.ENDC

```

;DEFINE MEMORY MANAGEMENT REGISTERS

M2

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
MEMORY MANAGEMENT DEFINITIONS

SEQ 025

```
; *USER "I" PAGE ADDRESS REGISTERS
UIPAR0= 177640
UIPAR1= 177642
UIPAR2= 177644
UIPAR3= 177646
UIPAR4= 177650
UIPAR5= 177652
UIPAR6= 177654
UIPAR7= 177656
. IF NB
; *USER "D" PAGE ADDRESS REGISTERS
UDPAR0= 177660
UDPAR1= 177662
UDPAR2= 177664
UDPAR3= 177666
UDPAR4= 177670
UDPAR5= 177672
UDPAR6= 177674
UDPAR7= 177676
. ENDC
. IF NB
; *SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
SIPDR0= 172200
SIPDR1= 172202
SIPDR2= 172204
SIPDR3= 172206
SIPDR4= 172210
SIPDR5= 172212
SIPDR6= 172214
SIPDR7= 172216
. IF NB
; *SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
SDPDR0= 172220
SDPDR1= 172222
SDPDR2= 172224
SDPDR3= 172226
SDPDR4= 172230
SDPDR5= 172232
SDPDR6= 172234
SDPDR7= 172236
. ENDC
; *SUPERVISOR "I" PAGE ADDRESS REGISTERS
SIPAR0= 172240
SIPAR1= 172242
SIPAR2= 172244
SIPAR3= 172246
SIPAR4= 172250
SIPAR5= 172252
SIPAR6= 172254
SIPAR7= 172256
. IF NB
; *SUPERVISOR "D" PAGE ADDRESS REGISTERS
SDPAR0= 172260
SDPAR1= 172262
SDPAR2= 172264
SDPAR3= 172266
```

N2

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
MEMORY MANAGEMENT DEFINITIONS

SEQ 026

```

SDPAR4 = 172270
SDPAR5 = 172272
SDPAR6 = 172274
SDPAR7 = 172276
.ENDC
.ENDC
; *KERNEL "I" PAGE DESCRIPTOR REGISTERS
172300 KIPDR0 = 172300
172302 KIPDR1 = 172302
172304 KIPDR2 = 172304
172306 KIPDR3 = 172306
172310 KIPDR4 = 172310
172312 KIPDR5 = 172312
172314 KIPDR6 = 172314
172316 KIPDR7 = 172316
; IF NB
; *KERNEL "D" PAGE
DESCRIPTOR REGISTERS
KOPDR0 = 172320
KOPDR1 = 172322
KOPDR2 = 172324
KOPDR3 = 172326
KOPDR4 = 172330
KOPDR5 = 172332
KOPDR6 = 172334
KOPDR7 = 172336
.ENDC
; *KERNEL "I" PAGE ADDRESS REGISTERS
172340 KIPAR0 = 172340
172342 KIPAR1 = 172342
172344 KIPAR2 = 172344
172346 KIPAR3 = 172346
172350 KIPAR4 = 172350
172352 KIPAR5 = 172352
172354 KIPAR6 = 172354
172356 KIPAR7 = 172356
; IF NB
; *KERNEL "D" PAGE ADDRESS REGISTERS
KOPAR0 = 172360
KOPAR1 = 172362
KOPAR2 = 172364
KOPAR3 = 172366
KOPAR4 = 172370
KOPAR5 = 172372
KOPAR6 = 172374
KOPAR7 = 172376
.ENDC

```

```

39          .SBTTL   TSV05 REGISTER AND PACKET DEFINITIONS
40
41          |
42          | SOME GENERAL EQUATES.
43          |
44
45          000004      ERRVEC==          4          | POINTER TO ERROR VECTOR FOR BUS TIME OUT.
46          000060      TTIVEC==         60          | INTERRUPT VECTOR FOR CONSOLE INPUT
47          177560      TTICSR==        177560        | BUS ADDRESS OF CONSOLE INPUT
48          177562      TTIBFR==        177562        | CONSOLE INPUT DATA BUFFER
49          177520      BDVPCR==        177520        | BDV11 PAGE CONTROL REGISTER
50
51          |*
52          |BIT DEFINITIONS FOR TSSR REGISTER
53          |*
54
55          100000      SC=      BIT15          |SPECIAL CONDITION
56          040000      BIE=      BIT14          |BUS INTERFACE ERROR
57          020000      SCE=      BIT13          |SANITY CHECK ERROR
58          010000      RMR=      BIT12          |MODIFICATION REFUSED
59          004000      NXM=      BIT11          |NONEXISTANT MEMORY ERROR
60          002000      NBA=      BIT10          |NEED BUFFER ADDRESS
61          001000      HIADDR=   BIT9:BIT8      |EXTENDED ADDRESS BITS
62          000200      SSR=      BIT7          |SUB SYSTEM READY
63          000100      OFL=      BIT6          |OFF LINE BIT
64          000060      FATERR=   BIT4:BIT5      |FATAL TERMINATION ERROR CODES
65          000016      TERCLS=   BIT3:BIT2:BIT1 |TERMINATION CODES
66
67          |*
68          |
69          |BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0
70          |(XST0)
71          |
72          |*
73
74          100000      XSOTMK=   BIT15          |TAPE MARK DETECTED
75          040000      XSORLS=   BIT14          |RECORD LENGTH SHORT
76          020000      XSOLET=   BIT13          |LOGICAL END OF TAPE
77          010000      XSORLL=   BIT12          |RECORD LENGTH LONG
78          004000      XSOWLE=   BIT11          |WRITE LOCK ERROR
79          002000      XSONEF=   BIT10          |NON EXECUTABLE FUNCTION
80          001000      XS0ILC=   BIT9          |ILLEGAL COMMAND
81          000400      XS0ILA=   BIT8          |ILLEGAL ADDRESS
82          000200      XSOMOT=   BIT7          |TAPE IN MOTION
83          000100      XS0ONL=   BIT6          |TRANSPORT ON LINE
84          000040      XS0IE=    BIT5          |INTERRUPT ENABLE
85          000020      XS0VCK=   BIT4          |VOLUME CHECK BIT
86          000010      XS0PED=   BIT3          |PHASE ENCODED DRIVE
87          000004      XS0WLK=   BIT2          |WRITE LOCKED
88          000002      XS0BOT=   BIT1          |BEGINNING OF TAPE
89          000001      XS0EOT=   BIT0          |END OF TAPE
90
91          |*
92          |BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
93          |(XST1)
94          |
95          100000      X1.DLT = BIT15          |DATA LATE

```

C3

TSV3 GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
TSV05 REGISTER AND PACKET DEFINITIONS

SEQ 028

```

96      040000      X1.SPARE = BIT14      ;NOT USED
97      020000      X1.COR  = BIT13      ;CORRECTABLE DATA ERROR
98      017375      X1.MBZ  = BIT12+BIT11+BIT10+BIT9+BIT7+BIT6+BIT5+BIT4+BIT3+BIT2+BIT0 ;ALWAYS 0
99      000400      X1.RBP  = BIT8       ;READ BUS PARITY ERROR
100     000002      X1.UNC  = BIT1       ;UNCORRECTABLE DATA OR HARD ERROR
101
102     ;
103     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2
104     ;(XST2)
105     ;
106     100000      X2.OPM  = BIT15      ;OPERATION IN PROGRESS (TAPE MOVING)
107     040000      X2.RCE  = BIT14      ;RAM CHECKSUM ERROR
108     035400      X2.SPARE = BIT13+BIT12+BIT11+BIT9+BIT8 ;NOT USED BY TSV05 (ALWAYS=0)
109     002000      X2.WCF  = BIT10      ;WRITE CLOCK FAILURE (FIFO NOT EMPTIED BY TRANSPORT)
110     000200      X2.EXTF = BIT7       ;IF WRITE CHAR CMD THEN = EXTENDED FEATURES ENABLED
111     000100      X2.BUFE = BIT6       ;IF WRITE CHAR CMD THEN = BUFFERING ENABLED
112     000077      X2.REV  = 000077    ;IF WRITE CHAR CMD THEN = MICROCODE REVISION LEVEL
113     000007      X2.UNIT = BIT2+BIT1+BIT0 ;IF GET STATUS THEN = CURRENTLY SELECTED UNIT NO.
114
115     ;
116     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3
117     ;(XST3)
118     ;
119     177400      X3.MDE  = 177400    ;MICRO-DIAGNOSTIC ERROR CODE
120     000200      X3.SPARE = BIT7      ;NOT USED BY TSV05
121     000100      X3.OPI  = BIT6      ;OPERATION INCOMPLETE
122     000040      X3.REV  = BIT5      ;REVERSE
123     000020      X3.TRF  = BIT4      ;TRANSPORT RESPONSE FAILURE
124     000010      X3.DCK  = BIT3      ;DENSITY CHECK
125     000006      X3.MBZ  = BIT2+BIT1 ;NOT USED ALWAYS 0
126     000001      X3.RIB  = BIT0      ;REVERSE INTO BOT
127
128     ;
129     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4
130     ;(XST4)
131     ;
132     100000      X4.HSP  = BIT15      ;HIGH SPEED
133     040000      X4.RCE  = BIT14      ;RETRY COUNT EXCEEDED
134     020000      X4.TSM  = BIT13      ;TRANSPORT SPECIAL MODE
135     017400      X4.MBZ  = BIT12+BIT11+BIT10+BIT9+BIT8 ;NOT USED ALWAYS 0
136     000377      X4.WRC  = 000377    ;WRITE RETRY COUNT FIELD
137
138     ;
139     ;
140     ;TSSR TERMINATION CODES (BIT 0-2)
141     ;
142     ;
143     ;
144     000006      TSREJ = 3+2          ;COMMAND REJECTED
145     000006      UNREC = 6           ;UNRECOVERABLE ERROR
146
147     ;
148     ;
149     ;DEVICE REGISTER OFFSETS
150     ;
151     ;
152

```

D3

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
TSV05 REGISTER AND PACKET DEFINITIONS

SEQ 029

```

153      000000      TSBA== 0
154      000000      TSDR== 0      ;TSDB/TSBA REGISTER
155      000001      TSBAH== 1
156      000001      TSDBH== 1      ;TSDB/TSBA REGISTER HIGH BYTE
157      000002      TSSR== 2      ;TSSR REGISTER
158      000003      TSSRH== 3      ;TSSR REGISTER HIGH BYTE
159
160      ;
161      ; TSDR ADDRESS BIT DEFINITIONS
162      ;
163      000003      A1716 = BIT1:BIT0      ;ADDRESS BITS 17:16 ARE IN 1:0
164
165      ;
166      ; COMMAND DEFINITIONS
167      ;
168      000017      P.GETSTAT = 17      ;GET STATUS
169      000013      P.INIT = 13      ;INITIALIZE
170      000012      P.CONTROL = 12      ;CONTROL COMMANDS
171      000011      P.FORMAT = 11      ;FORMAT
172      000010      P.POSITION = 10      ;POSITION
173      000006      P.WRTSUB = 6      ;SUBSYSTEM WRITE
174      000005      P.WRITE = 5      ;WRITE
175      000004      P.WRTCHAR = 4      ;WRITE CHARACTERISTICS
176      000001      P.READ = 1      ;READ
177
178      ;
179      ; COMMAND PACKET HEADER WORD BIT DEFINITIONS
180      ;
181      100000      P.ACK = BIT15      ;BUFFER AVAIL FOR CONTROLLER
182      040000      P.CVC = BIT14      ;CLEAR VOLUME CHECK
183      020000      P.OPP = BIT13      ;REVERSE SEQUENCE OF DATA BITS
184      010000      P.SWB = BIT12      ;SWAP BYTES IN MEMORY
185      007400      P.MODE = BIT11:BIT10:BIT9:BIT8 ;EXTENDED COMMAND MODE FIELD
186      000200      P.IE = BIT7      ;INTERRUPT ENABLE
187      000140      P.FMT = BIT6:BIT5      ;PACKET HEADER TYPE (ALWAYS=0)
188      000037      P.CMD = 37      ;MAJOR COMMAND FIELD
189
190      ;
191      ; CONTROL COMMAND MODE CODES
192      ;
193      000000      PC.RELEASE = 0*256.      ;RELEASE BUFFER
194      000400      PC.REWIND = 1*256.      ;REWIND
195      001000      PC.NOOP = 2*256.      ;NO-OP
196      002000      PC.IEREW = 4*256.      ;REWIND IMMEDIATE INTERRUPT
197      002400      PC.ERASE = 5*256.      ;SECURITY ERASE
198
199      ;
200      ; CONTROLLER RAM DEFINITIONS
201      ;
202      000167      RMCHBEG = 167      ;CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
203      000200      RMCHEND = 200      ;CHARACTERISTICS IO DATA END RAM ADDRESS
204      000201      RMPKTBEG = 201      ;COMMAND PACKET BEGIN RAM ADDRESS
205      000210      RMPKTEND = 210      ;COMMAND PACKET END RAM ADDRESS
206      000215      RMMSGBEG = 215      ;MESSAGE BUFFER BEGIN RAM ADDRESS
207      000234      RMMSGEND = 234      ;MESSAGE BUFFER END RAM ADDRESS
208
209      ;
210      ; REGISTER DEFINITIONS IN THE MESSAGE BUFFER

```

E3

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
TSV05 REGISTER AND PACKET DEFINITIONS

SEQ 030

```

210      ;
211      ;
212      ;
213      000006      XST0== 6      ;EXTENDED STATUS REGISTER 0 (WORD 4)
214      000010      XST1== 8      ;EXTENDED STATUS REGISTER 1 (WORD 5)
215      000012      XST2== 10     ;EXTENDED STATUS REGISTER 2 (WORD 6)
216      000014      XST3== 12     ;EXTENDED STATUS REGISTER 3 (WORD 7)
217      000016      XST4== 14     ;EXTENDED STATUS REGISTER 4 (WORD 8)
218
219      ;
220      ;
221      ;OFFSETS TO WORD LOCATIONS IN PACKET DEFINITIONS
222      ;
223      ;
224
225      000002      PKLOW  = 2      ;LOW ORDER CHARACTERISTIC DATA POINTER
226      000004      PKPI   = 4      ;HIGH ORDER CHARACTERISTIC DATA POINTER
227      000006      PKBCNT = 6      ;NUMBER OF BYTES IN DATA PACKET
228
229      000010      EXBCNT=10      ;NUMBER OF BYTES IN EXTENDED DATA PACKET
230
231      ;
232      ;DATA PACKET OFFSETS FOR WRITE SUBSYSTEM COMMAND
233      ;
234      000000      BSEL0  = 0      ;BYTE 0
235      000001      BSEL1  = 1      ;BYTE 1
236      000002      SEL2   = 2      ;WORD 2
237      000004      SELDATA = 4      ;WORD 3
238
239      ;
240      ;BSEL0 SELECT CODES FOR WRITE SUBSYSTEM COMMAND
241      ;
242      000000      PW.NOP      = 0      ;NO-OP
243      000001      PW.RDRAM    = 1      ;READ RAM
244      000002      PW.WTRAM    = 2      ;WRITE RAM
245      000003      PW.RFIFO    = 3      ;READ FIFO
246      000004      PW.WFIFO    = 4      ;WRITE FIFO
247      000005      PW.RDSTAT   = 5      ;READ STATUS
248      000006      PW.WCTL     = 6      ;WRITE TAPE CONTROL
249      000007      PW.WFMT     = 7      ;WRITE TAPE FORMAT
250      000010      PW.WMISC    = 10     ;WRITE MISCELLANEOUS
251      000011      PW.WNPR     = 11     ;WRITE NPR CONTROL
252      000020      PW.D22     = 20     ;DO MICROTEST 22
253      000021      PW.D11     = 21     ;DO MICROTEST 11
254      000022      PW.D13     = 22     ;DO MICROTEST 13
255      000023      PW.NO1311   = 23     ;DISABLE MICROTEST 11 AND 13
256      000024      PW.RDXT     = 24     ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSPORTS)
257
258      ;
259      ;BSEL1 CODES FOR WRITE TAPE CONTROL
260      ;
261      000200      WC.IFAD      = BIT7    ;IFAD - FORMATTER ADDRESS
262      000100      WC.IOTAD     = BIT6    ;IOTAD - TRANSPORT ADDRESS BIT 0
263      000040      WC.I1TAD     = BIT5    ;I1TAD - TRANSPORT ADDRESS BIT 1
264      000020      WC.ISRESV    = BIT4    ;IRESV5 - RESERVED #5
265      000010      WC.IREW      = BIT3    ;IREW - REWIND
266      000004      WC.IRWU      = BIT2    ;IRWU - REWIND AND UNLOAD

```

F3

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
TSV05 REGISTER AND PACKET DEFINITIONS

SEQ 031

```

267      000002      WC.IFEN      = BIT1      ;IFEN - FORMATTER ENABLE
268      000001      WC.IGO       = BIT0      ;GO
269
270      ;*
271      ;BSEL1 CODES FOR WRITE FORMAT
272      ;-
273      000200      WF.IHISP      = BIT7      ;IHISP - HIGH SPEED
274      000100      WF.IWRT      = BIT6      ;IWRT - WRITE
275      000040      WF.IREV      = BIT5      ;IREV - REVERSE
276      000020      WF.IWFM      = BIT4      ;IWFM - WRITE FILE MARK
277      000010      WF.IEDIT     = BIT3      ;IEDIT - EDIT
278      000004      WF.IERASE    = BIT2      ;IERASE - ERASE
279      000002      WF.I3RESV    = BIT1      ;IRESV3 - RESERVED #3
280      000001      WF.I4RESV    = BIT0      ;IRESV4 - RESERVED #4
281
282      ;*
283      ;BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND
284      ;-
285      000200      MS.EXT        = BIT7      ;INVERT SENSE OF EXTENDED FEATURES SWITCH
286      000020      MS.RSFIFO     = BIT4      ;RESET FIFO AND INPUT PARITY ERRORR
287      000010      MS.RSTAPE     = BIT3      ;RESET TAPE STATUS IN 2 FLIP-FLOPS
288      000006      MS.ATTN       = BIT2:BIT1 ;ATTENTION TRIGGER FIELD
289      000001      MS.RSD        = BIT0      ;RESET TIMER A,B THEN DELAY TIMES IN SEL2
290
291      ;*
292      ; MS.ATTN SUBCODES
293      ;-
294      000000      MSA.NOP       = 0*2      ;NO-OP (NOTHING TRIGGERED)
295      000002      MSA.VOL       = 1*2      ;SIMULATE ON-LINE/OFF-LINE TRANSITION
296      000004      MSA.NRAM      = 2*2      ;FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)
297      000006      MSA.FRAME     = 3*2      ;FORCE FATAL RAM ERROR (CAUSES SCE TO SET)
298
299      ;*
300      ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
301      ;-
302      000200      NP.IR         = BIT7      ;INTERRUPT REQUEST (0-1 TRANSITION)
303      000100      NP.OUT        = BIT6      ;TAPE DATA DIRECTION OUT (0= IN)
304      000040      NP.LOOP       = BIT5      ;ENABLE TRANSPORT LOOPBACK
305      000020      NP.WRP        = BIT4      ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
306
307      ;*
308      ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
309      ;-
310      000200      S2.DIM         = BIT7      ;WORD #9 BYTE 2 DATA IN MISS
311      000100      S2.ILW        = BIT6      ;ILW H
312      000040      S2.OUTRDY     = BIT5      ;OUT RDY H
313      000020      S2.INRDY      = BIT4      ;IN RDY H
314      000010      S2.ATIMR      = BIT3      ;TIMER A FLAG H
315      000004      S2.BTIMR      = BIT2      ;TIMER B FLAG H
316      000003      S2.UNDEF      = BIT1:BIT0 ; (UNDEFINED)
317      1C0000      S1.PARIN      = BIT15     ;WORD #8 BYTE 1 PARIN H
318      040000      S1.I2RESV     = BIT14     ;IRESV2
319      020000      S1.I1RESV     = BIT13     ;IRESV1
320      010000      S1.IEOT       = BIT12     ;IEOT L
321      004000      S1.IIDENT     = BIT11     ;IIDENT H
322      002000      S1.ICER       = BIT10     ;ICER H
323      001000      S1.IFMK       = BIT9      ;IFMK H
324      000400      S1.IHER       = BIT8      ;IHER H
325      000200      S0.ISPEED     = BIT7      ;WORD #8 BYTE 0 ISPEED H

```

G3

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
TSV05 REGISTER AND PACKET DEFINITIONS

SEQ 032

```

324      000100      SO,IRDY      = BIT6      ;      IRDY L
325      000040      SO,IONL      = BIT5      ;      IONL L
326      000020      SO,ILDP      = BIT4      ;      ILDP L
327      000010      SO,IDBY      = BIT3      ;      IDBY L
328      000004      SO,IRWD      = BIT2      ;      IRWD L
329      000002      SO,IFBY      = BIT1      ;      IFBY L
330      000001      SO,IFPT      = BIT0      ;      IFPT L
331                                     .SBTTL   SPECIAL MACROS AND OPDEFS.
332
333      ;+
334      ;SAVE GENERAL REGS 1 TO 5
335      ;-
336
337      .MACRO   SAVREG
338      JSR      R5,REGSAV
339      .ENDM
340
341      ;+
342      ; MACRO TO FORCE AN ERROR
343      ;-
344      .MACRO   FORCERROR      TAG,NOTSSR
345      .NLIST
346      .IIF NDF LISTALL, .NLIST
347      .LIST
348      .IF B NOTSSR
349      MOV      TSSR(R5),R1      ;READ TSSR
350      .ENDC
351      MOV      FORCER,FORCER    ;IS FORCER SET? (LEAVE C BIT ALONE)
352      BNE      TAG              ;BR IF YES
353      .NLIST
354      .IIF NDF LISTALL, .LIST
355      .LIST
356      .ENDM
357
358      ;+
359      ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
360      ; WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
361      ; SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
362      ; FORCER TO 17777
363      ; TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
364      ;-
365      .MACRO   FORCEEXIT      TAG
366      .NLIST
367      .IIF NDF LISTALL, .NLIST
368      .LIST
369      MOV      FORCER,FORCER    ;IS FORCER NEGATIVE?
370      BMI      TAG              ;BR IF YES
371      .NLIST
372      .IIF NDF LISTALL, .LIST
373      .LIST
374      .ENDM
375
376      ;+
377      ; MACRO TO INCREMENT ERROR COUNTS
378      ;-
379      .MACRO   NEXT.ERRNO
380      .NLIST
381      .IIF NDF LISTALL, .NLIST

```

H3

TSV3 GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
SPECIAL MACROS AND OPDEFS.

SEQ 033

```

381          ERRNO=ERRNO+1
382      ;;;.IIF NDF LISTALL, .LIST
383      .LIST
384      .ENDM
385
386      ;+
387      ;MACRO TO PERFORM XOR
388      ;-
389
390          .MACRO XOR      A,B
391          MOV      A,-(SP)
392          BIC      B,(SP)
393          BIC      A,B
394          BIS      (SP)+,B
395          .ENDM
396
397          000000          EN=0          ; INITIALIZE ERROR NUMBER
398          .SBTTL  FORCER  - FORCE ERROR FLAG
399
400      ;
401      ; THE FOLLOWING LOCATIONS MAY BE PATCHED BY THE USER
402      ; TO OBTAIN THE RESULTS DESCRIBED FOR EACH.
403      ;
404
405      002176  000000  FORCER::      0          ; FORCE TYPE ALL HARD ERRORS (THE ONES CALLED -
406          ; - BY THE MACRO "IFERROR"). AN ERROR NEED NOT -
407          ; - EXIST, JUST ASSUME AND TYPE THE MESSAGE.
408          .SBTTL  GLOBAL DATA SECTION
409
410      ;++
411      ;THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
412      ;IN MORE THAN ONE TEST.
413      ;--
414
415      ;
416      ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
417      ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
418      ;
419      002200  000000  EPRTSW::      .WORD  0          ;PRINT SWITCH
420      002202  000000  UNITN::      .WORD  0          ;UNIT # UNDER TEST.
421      002204  000000  QVP::      .WORD  0          ;QUICK VERIFY FLAG.
422      002206  000000  CSRADDR::    .WORD  0          ;ADDRESS OF CSR FOR CURRENT DEVICE
423      002210  000224  IVEC::      .WORD  224        ;INTERRUPT VECTOR
424      002212  000200  IPRI::      .WORD  PRI04      ;INTERRUPT PRIORITY.
425      002214  000000  TSTCNT::    .WORD  0          ;NUMBER OF TESTS RUN IN THIS PASS
426      002216  000000  LOOPCNT::   .WORD  0          ;REMAINING ITERATION COUNT FOR TEST
427      002220  000000  DEVCNT::    .WORD  0          ;NUMBER OF DEVICE UNDER TEST
428      002222  000000  FATFLG::    .WORD  0          ;SET IF FATAL ERROR IS DETECTED IN TEST
429      002224  000000  INTRECV::   .WORD  0          ;SET IF TAPE INTERRUPT WAS RECEIVED
430      002226  000000  EXTFEA::    .WORD  0          ;EXTENDED FEATURES SOFTWARE SW 0=OFF;1=ON
431      002230  000000  BENBSW::    .WORD  0          ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
432      002232  000000  EXPD::      .WORD  0          ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
433      002234  000000  RECV::      .WORD  0          ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
434      002236  000000  ERRHI::     .WORD  0          ;HIGH ADDRESS MEMORY ERROR
435      002240  000000  ERRLO::     .WORD  0          ;LOW ADDRESS MEMORY ERROR
436      002242          RAMDATA::    .BLKW  16.        ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
437      002302  000000  RAMSIZ::    .WORD  0          ;RAM DATA SIZE FOR PRAMPKT ROUTINE

```

TSV3 - GLOBAL AREAS  
GLOBAL DATA SECTION

MACRO M1113 06-FEB-84 17:14

SEQ 034

```

438 002304 000000      RCVHIADD::      .WORD  0      ;RECEIVED BUFFER HIGH ADDRESS
439 002306 000000      RCVLOADD::      .WORD  0      ;RECEIVED BUFFER LOW ADDRESS
440 002310 000000      COUNT::          .WORD  0      ;TEST COUNT PATTERN
441 002312 000000      DATA::          .WORD  0      ;TEST DATA
442 002314 000000      TSTFLAG::        .WORD  0      ;TEST FLAG WORD
443 002316 000000      TSTPTR::         .WORD  0      ;TSTBLK POINTER
444 002320 000000      PRMNO::          .WORD  0      ;PRINT ROUTINE TEMP
445 002322      EXPMSG::      .BLKB  100.      ;EXPECTED MESSAGE BUFFER DATA
446 002466      RECMSG::      .BLKB  100.      ;RECEIVED MESSAGE BUFFER DATA
447 002632      TMPBFR::      .BLKB  80.      ;TEMPORARY STORAGE FOR PRINT
448                      .SBTTL  TSTBLK  - TEST DATA TABLE
449
450      ;*
451      ;
452      ;THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
453      ;
454      ;IN SEQUENCE THE DATA IS:
455      ;
456      ;      ALL ZEROS
457      ;      ALL ONES
458      ;      WALKING ONES
459      ;      WALKING ZEROS
460      ;      ALTERNATING ONES AND ZEROS
461      ;
462      ;-
463
464 002752      TSTBLK::
465 002752 000000      .WORD  0      ;ALL ZEROS
466 002754 177777      .WORD  177777      ;ALL ONES
467 002756 000001      .WORD  BIT0      ;DATA FOR WALKING ONES
468 002760 000002      .WORD  BIT1
469 002762 000004      .WORD  BIT2
470 002764 000010      .WORD  BIT3
471 002766 000020      .WORD  BIT4
472 002770 000040      .WORD  BIT5
473 002772 000100      .WORD  BIT6
474 002774 000200      .WORD  BIT7
475 002776 000400      .WORD  BIT8
476 003000 001000      .WORD  BIT9
477 003002 002000      .WORD  BIT10
478 003004 004000      .WORD  BIT11
479 003006 010000      .WORD  BIT12
480 003010 020000      .WORD  BIT13
481 003012 040000      .WORD  BIT14
482 003014 100000      .WORD  BIT15
483 003016 177776      .WORD  *CBIT0      ;DATA FOR WALKING ZEROS
484 003020 177775      .WORD  *CBIT1
485 003022 177773      .WORD  *CBIT2
486 003024 177767      .WORD  *CBIT3
487 003026 177757      .WORD  *CBIT4
488 003030 177737      .WORD  *CBIT5
489 003032 177677      .WORD  *CBIT6
490 003034 177577      .WORD  *CBIT7
491 003036 177377      .WORD  *CBIT8
492 003040 176777      .WORD  *CBIT9
493 003042 175777      .WORD  *CBIT10
494 003044 173777      .WORD  *CBIT11

```

J3

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
TSTBLK - TEST DATA TABLE

SEQ 035

```

495 003046 167777      .WORD      †CBIT12
496 003050 157777      .WORD      †CBIT13
497 003052 137777      .WORD      †CBIT14
498 003054 077777      .WORD      †CBIT15
499 003056 125252      .WORD      125252      ;ALTERNATING ONES, ZEROS
500 003060 052525      .WORD      052525      ;ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE
501          003062
502
503
504
505
506 003062 000000 100000 000000  DUMMY: 0,100000,0,0      ;DUMMY DEVICE REGISTERS...
507 003072 000000 000000 000000      0,0,0,0,0,0,0,0,0
508
509
510 003112 000000      DUFLG::      .WORD      0      ;"DROPPED UNIT" FLAG.
511
512 003114 000000      NODEV::      .WORD      0      ;INHIBITS CODE IN "CLEAN-UP".
513
514 003116 000000      TEMP1::      .WORD      0      ;SOME TEMP LOCATIONS.
515 003120 000000      TEMP2::      .WORD      0
516 003122 000000      XXCOMM::      .WORD      0      ;XXDP, COMM BLOCK POINTER.
517 003124 000000      FREE::      .WORD      0      ;1ST FREE MEMORY ADDRESS...
518 003126 000000      FRESIZ::      .WORD      0      ;...AND SIZE (IN WORDS).
519 003130 000000      FREEHI: .WORD      0      ;LAST WORD IN FREE SPACE
520 003132 000000      KTFLG::      .WORD      0      ;KT11, MEM AVAIL FLAG -
521
522
523 003134 000000      KTENABLE::      .WORD      0      ;- .WORD      0 * <24K OR NO KT -
524 003136 000000      NXMFLG::      .WORD      0      ;- NZ = >24K AND KT.
525 003140 000000      NXMLO::      .WORD      0      ;SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
526 003142 000000      NXMHI::      .WORD      0      ;SET IF WE CAN TEST CLEARED OTHERWISE.
527 003144 000000      T23A::      .WORD      0      ;NXM LO ADDRESS BITS
528 003146 000000      T23B::      .WORD      0      ;NXM HI ADDRESS BITS FOR DAL'S 16-21
529 003150 000000      T3BFLG::      .WORD      0      ;11/23A FLAG
530 003152 002000      PST32W::      .WORD      2000      ;11/23B FLAG
531 003154 000000      SIFLAG::      .WORD      0      ;TEST 3B FLAG †0
532 003156 000000      BADDAT::      .WORD      0      ;32W BLOCK ADDRESS FOR 32K START
533 003160 000000      GODAT::      .WORD      0      ;
534 003162 000000      LOOPFL::      .WORD      0      ;ACTUAL DATA
535 003164
536 003164 000000      CTAB::      .WORD      0      ;CONFIGURATION TABLES.
537 003166 000000      CTABM::      .WORD      0      ;CONFIG WORK.
538 003170 000000
539 003172 000000
540 003174 177777
541 003176
542
543
544
545
546
547
548
549
550
551 003176      ERTABL:      .BLKW      64,

CTABE::
;ERROR STATISTICS TABLE (1 WORD PER UNIT), 64 UNITS MAX:
;
;      0      =      UNIT NOT TESTED
;      100000 =      UNIT ONLINE, NO ERRORS
;      10XXXX =      UNIT ONLINE, ENCOUNTERED XXXX ERRORS
;      160000 =      UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
;      160001 =      UNIT DROPPED, NOT IDLE AT START
;      14XXXX =      UNIT DROPPED, ENCOUNTERED XXXX ERRORS
;

```

K3

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
GLOBAL ENVIRONMENT STORAGE

SEQ 036

552 003376    000000

ERTABE:      .WORD    0

553

554 003400    000000

SKIPT:    .WORD    0

;1\*SKIP SUBTEST 0-NO SKIP OF SUBTEST

L3

TSV3 - GLOBAL AREAS  
GLOBAL TEXT MESSAGES

MACRO M1113 06-FEB-84 17:14

SEQ 037

```

556          ,SBTTL  GLOBAL TEXT MESSAGES
557          ;**
558          ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
559          ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
560          ; MORE THAN ONE TEST.
561          ;**
562
563          ;*
564          ; NAMES OF DEVICES SUPPORTED
565          ;*
566
567          003402          DEVTYP  <TSV05>
          003402          L$DVTYP:;
          003402          .ASCIZ  /TSV05/
          124          123          126          .EVEN

568
576          ;*
577          ; TEST DESCRIPTION
578          ;*
579          003410          DESCRIPT <**** TSV05 LOGIC DIAGNOSTIC - REPLACE M7196 IF ERROR ****>
          003410          L$DESC:;
          003410          .ASCIZ  /**** TSV05 LOGIC DIAGNOSTIC - REPLACE M7196 IF ERROR ****/
          052          052          052          .EVEN

580
594
595          ;*
596          ; BIT TO ASCII CONVERSION FOR TSSR REGISTER
597          ;*
598
599          003502          003542          003545          003551          TSSRBIT:;          .WORD  1$,2$,3$,4$,5$,6$,7$,8$
600          003522          003603          003607          003613          .WORD  9$,10$,11$,12$,13$,14$,15$,16$
601          003542          123          103          000          1$:          .ASCIZ  'SC'
602          003545          102          111          105          2$:          .ASCIZ  'BIE'
603          003551          123          103          105          3$:          .ASCIZ  'SCE'
604          003555          122          115          122          4$:          .ASCIZ  'RMR'
605          003561          116          130          115          5$:          .ASCIZ  'NXM'
606          003565          116          102          101          6$:          .ASCIZ  'NBA'
607          003571          102          111          124          7$:          .ASCIZ  'BIT9'
608          003576          102          111          124          8$:          .ASCIZ  'BIT8'
609          003603          123          123          122          9$:          .ASCIZ  'SSR'
610          003607          117          106          114          10$:          .ASCIZ  'OFL'
611          003613          102          111          124          11$:          .ASCIZ  'BIT5'
612          003620          102          111          124          12$:          .ASCIZ  'BIT4'
613          003625          102          111          124          13$:          .ASCIZ  'BIT3'
614          003632          102          111          124          14$:          .ASCIZ  'BIT2'
615          003637          102          111          124          15$:          .ASCIZ  'BIT1'
616          003644          102          111          124          16$:          .ASCIZ  'BIT0'
617          .EVEN
618          003652          124          123          123          SF1ERR: .ASCIZ  'TSSR ERROR AFTER SOFT INIT'
619          003705          124          123          123          SFHLRR: .ASCIZ  'TSSR ERROR AFTER BUS RESET'
620          003740          040          040          116          NXR:      .ASCIZ  / NON-EXISTANT DEVICE REGISTER/
621          003777          045          101          040          NXR$:      .ASCIZ  /#A ADDRESS: #06/
622          004020          045          101          040          TSSX:      .ASCIZ  /#A TSBA,TSSR EXP'D: #06#A,#06#N/
623          004060          045          101          040          .ASCIZ  /#A TSBA,TSSR REC'D: #06#A,#06/
624          004117          045          116          045          FUSI:      .ASCIZ  /#N#A/
625          004123          040          040          125          USI:      .ASCIZ  / UNEXPECTED INTERRUPT/
626          004152          040          040          111          NSI:      .ASCIZ  / INTERRUPT EXPECTED, NOT RECEIVED/

```

M3

TSV3 - GLOBAL AREAS  
GLOBAL TEXT MESSAGES

MACRO M1113 06-FEB-84 17:14

SEQ 038

```

627 004215      045      116      045 FNOINTR: .ASCII /%N%A/
628 004221      040      040      116 NOINTR: .ASCIIZ / NO INTERRUPT WAS GENERATED/
629 004256      040      040      111 IFAULT: .ASCIIZ / INTERRUPT FAULT/
630 004300      045      101      040 INTX. .ASCIIZ /%A CPU PC: %06%A TSBA: %06/
631 004335      040      040      042 NOINIT: .ASCIIZ / "BUS-INIT" DIDN'T INITIALIZE CONTROLLER/
632 004407      040      040      042 NSINIT: .ASCIIZ / "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
633 004457      040      040      042 BRINIT: .ASCIIZ / "BUS-RESET" DIDN'T INITIALIZE THE DPU/
634
635 004527      000
636 004530      045      116      000 NUL: .ASCIIZ //
637 004533      045      101      040 NULCR: .ASCIIZ /%N/
638 004567      045      116      045 EXPGT2: .ASCIIZ /%N%A EXP'D: %06%A, REC'D: %06/
639 004643      045      101      040 DUAD12: .ASCIIZ /%A REG(W) WRITTEN TO: %06%A REG(R) READ: EXP'D: %06%A, REC'D: %06/
640 004745      122      101      115 PKTRAM: .ASCIIZ 'RAM Contents Do Not Match Packet Sent'
641 005013      040      040      103 SCME: .ASCIIZ / CONFIG DOESN'T MATCH MFG. MASTER/
642 005056      127      122      111 WRTMSG: .ASCIIZ 'WRITE CHARACTERISTICS Failed'
643 005113      124      123      123 WRTERR: .ASCIIZ 'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
644 005206      124      123      123 RDERR: .ASCIIZ 'TSSR Incorrect After READ Command, More Bits Set Than SSR'
645 005300      106      101      124 SCHERR: .ASCIIZ 'FATAL ERROR IN SUBTEST - CHECK TAPE, CABLES, TRANSPORT etc.'
646 005372      105      122      122 RETERR: .ASCIIZ 'ERROR IN SUBTEST - WRITE DATA RETRY FIVE TIMES FAILED'
647 005460      045      116      045 NOMEM: .ASCIIZ '%N%A ***** NO NXM ADDRESS--CANNOT TEST NXM TIMEOUT. *****N'
648 005554      045      116      045 M8186: .ASCIIZ '%N%A ***** 11/23A SYSTEM *****N'
649 005645      045      116      045 M8189: .ASCIIZ '%N%A ***** 11/23B SYSTEM *****N'
650 .EVEN
651 .SBTTL GLOBAL ERROR REPORT SECTION
652
653
654 ;++
655 ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX
656 ; CALLS THAT ARE USED IN MORE THAN ONE TEST.
657 ; ASCII TEXT STRINGS ARE FOUND IN THE GLOBAL TEXT SECTION.
658 ;--
659 005736      BGNMSG      NXRRERR      ;NON-EXISTANT DEVICE REGISTER.
660 005736      NXRRERR:
005736      PRINTX      %NXRX,NODEV      ;NODEV = NEXM ADDRESS.
005742      MOV      NODEV,-(SP)
005746      MOV      %NXRX,-(SP)
005752      MOV      %2,-(SP)
005754      MOV      SP,R0
005756      TRAP      C$PNTX
005762      ADD      %6,SP
005766      JSR      PC,EXTEND      ; PRINT EXTENSION IF REQUIRED.
005766      ENDMMSG
L10002:      TRAP      C$MSG
663
664
665 ;
666 ; THIS ROUTINE APPENDS A UNIQUE EXTENSION (IF REQUIRED)
667 ; TO ANY OF THE ABOVE ERROR SIGNATURES.
668
668 005770      005727
669 005772      000000
670 005774      001402
671 005776      004777      177770
672 006002
006002      012746      004530
006006      012746      000001

EXTEND: TST      (PC)+
EXTA:      0
BEQ      1$
JSR      PC,%EXTA      ; APPEND EXTENSION TEXT.
1$:      PRINTX      %NULCR      ; PRINT A BLANK LINE
MOV      %NULCR,-(SP)
MOV      %1,-(SP)

```

N3

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
GLOBAL ERROR REPORT SECTION

SEQ 039

|            |        |        |      |         |
|------------|--------|--------|------|---------|
| 006012     | 010600 |        | MOV  | SP,RO   |
| 006014     | 104415 |        | TRAP | C\$PNTX |
| 006016     | 062706 | 000004 | ADD  | #4,SP   |
| 673 006022 | 000207 |        | RTS  | PC      |

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 PRITSSR - PRINT TSSR CONTENTS

SEQ 040

```

675 .SBTTL PRITSSR - PRINT TSSR CONTENTS
676
677
678
679 ;ROUTINE TO DISPLAY THE CONTENTS, AND BIT DEFINITIONS, OF
680 ;THE TSSR REGISTER. THIS ROUTINE IS NORMALLY CALLED ONLY
681 ;BY A MESSAGE PRINTING ROUTINE
682
683 ;INPUTS:
684 ;
685 ; R1 CONTENTS OF TSSR
686
687 ;SUBORDINATE ROUTINES:
688 ;
689 ; CHKAMB CHECK FOR AMBIGUOUS CONTENTS
690 ;
691 ;
692
693 PRITSSR:
694 SAVREG ;SAVE GENERAL REGISTERS
695 MOV R1,R4 ;SAVE THE TSSR CONTENTS
696 PRINTB @TSSRFOR,R4 ;PRINT THE CONTENTS OF TSSR
        MOV R4,-(SP)
        MOV @TSSRFOR,-(SP)
        MOV @2,-(SP)
        MOV SP,R0
        TRAP C:PNTB
        ADD @6,SP
        MOV R4,R0 ;GET TSSR BACK FOR CHKAMB
        JSR PC,CHKAMB ;ARE CONTENTS AMBIGUOUS ?
        BCS 5$ ;BRANCH IF NOT
        PRINTX @AMBTSSR ;SHOW CONTENTS ARE AMBIGUOUS
        MOV @AMBTSSR,-(SP)
        MOV @1,-(SP)
        MOV SP,R0
        TRAP C:PNTX
        ADD @4,SP
5$: MOV R4,R3 ;CONTENTS OF TSSR
        BIC @MIADDR!FATERR!TERCLS,R3 ;CLEAR ALL MULTIPLE BIT FIELDS
        BEQ 20$ ;NO BITS ARE SET
        MOV @TMPBFR,R2 ;TEMPORARY ASCII BUFFER
        MOV @TSSRBIT,R1 ;ASCII EQUIVALENT OF BITS
10$: TST R3 ;REMAINING BITS TO CONVERT
        BEQ 15$ ;BRANCH WHEN ALL ARE DONE
        CLC ;CLEAR CARRY FOR SHIFT
        ROL R3 ;SHIFT NEXT BIT TO CARRY
        BCC 13$ ;BRANCH IF BIT NOT SET
        MOV (R1),R0 ;POINTER TO BIT DEFINITION
11$: MOVB (R0),-(R2) ;MOVE ASCII TO BUFFER
        BNE 11$ ;MOVE ALL BITS
        MOVB @'-1(R2) ;INSERT A COMMA TO TERMINATE
13$: TST (R1) ;POINT TO NEXT DESCRIPTION
        BR 10$ ;GET THE REMAINING BITS
15$: CLRB -(R2) ;TERMINATE THE LINE
        PRINTX @TSSDEF,@TMPBFR ;PRINT THE BIT DEFINITIONS
        MOV @TMPBFR,-(SP)
        MOV @TSSDEF,-(SP)

```

C4

TSV3 GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
PRITSSR PRINT TSSR CONTENTS

SEQ 041

|        |        |        |        |        |               |                                                 |
|--------|--------|--------|--------|--------|---------------|-------------------------------------------------|
| 006170 | 012746 | 000002 |        | MOV    | #2,-(SP)      |                                                 |
| 006174 | 010600 |        |        | MOV    | SP,R0         |                                                 |
| 006176 | 104415 |        |        | TRAP   | C1PNTX        |                                                 |
| 006200 | 062706 | 000006 |        | ADD    | #6,SP         |                                                 |
| 719    |        |        |        |        |               |                                                 |
| 720    | 006204 | 010403 | 20:    | MOV    | R4,R3         | ;GET THE TSSR CONTENTS                          |
| 721    | 006206 | 042703 |        | BIC    | #1CFERCLS,R3  | ;CLEAR ALL BUT TERMINATION                      |
| 722    | 006212 | 016303 | 177761 | MOV    | TCOCOD(R3),R3 | ;GET THE TERMINATION CODE MEANING               |
| 723    | 006216 |        | 006676 | PRINTX | #TCOASC,R3    | ;PRINT THE TERMINATION CODE                     |
|        | 006216 | 010346 |        | MOV    | R3,-(SP)      |                                                 |
|        | 006220 | 012746 | 006476 | MOV    | #1COASC,-(SP) |                                                 |
|        | 006224 | 012746 | 000002 | MOV    | #2,-(SP)      |                                                 |
|        | 006230 | 010600 |        | MOV    | SP,R0         |                                                 |
|        | 006232 | 104415 |        | TRAP   | C1PNTX        |                                                 |
|        | 006234 | 062706 | 000006 | ADD    | #6,SP         |                                                 |
| 724    | 006240 | 010403 |        | MOV    | R4,R3         | ;TSSR CONTENTS AGAIN                            |
| 725    | 006242 | 042703 | 177717 | BIC    | #1CFATERR,R3  | ;CLEAR ALL BUT FATAL TERMINATION                |
| 726    | 006246 | 001416 |        | BEQ    | 25:           | ;DON'T PRINT IF ZERO                            |
| 727    | 006250 | 006203 |        | ASR    | R3            |                                                 |
| 728    | 006252 | 006203 |        | ASR    | R3            |                                                 |
| 729    | 006254 | 006203 |        | ASR    | R3            | ;ALINE TERMINATION CODE FOR INDEX               |
| 730    | 006256 | 016303 | 007236 | MOV    | TSFCOD(R3),R3 | ;GET THE FATAL TERMINATION CODE                 |
| 731    | 006262 |        |        | PRINTX | #TFCASC,R3    | ;PRINT THE FATAL TERMINATION CODE               |
|        | 006262 | 010346 |        | MOV    | R3,-(SP)      |                                                 |
|        | 006264 | 012746 | 006537 | MOV    | #TFCASC,-(SP) |                                                 |
|        | 006270 | 012746 | 000002 | MOV    | #2,-(SP)      |                                                 |
|        | 006274 | 010600 |        | MOV    | SP,R0         |                                                 |
|        | 006276 | 104415 |        | TRAP   | C1PNTX        |                                                 |
|        | 006300 | 062706 | 000006 | ADD    | #6,SP         |                                                 |
| 732    | 006304 | 042704 | 176377 | 25:    | BIC           | #1CHIADDR,R4                                    |
| 733    | 006310 | 001411 |        | BEQ    | 30:           | ;DON'T PRINT IF ZERO                            |
| 734    | 006312 |        |        | PRINTX | #TEXASC,R4    | ;PRINT THE EXTENDED ADDRESS BITS                |
|        | 006312 | 010446 |        | MOV    | R4,-(SP)      |                                                 |
|        | 006314 | 012746 | 006435 | MOV    | #TEXASC,-(SP) |                                                 |
|        | 006320 | 012746 | 000002 | MOV    | #2,-(SP)      |                                                 |
|        | 006324 | 010600 |        | MOV    | SP,R0         |                                                 |
|        | 006326 | 104415 |        | TRAP   | C1PNTX        |                                                 |
|        | 006330 | 062706 | 000006 | ADD    | #6,SP         |                                                 |
| 735    | 006334 | 013703 | 002200 | 30:    | MOV           | EPRTSW,R3                                       |
| 736    | 006340 |        |        | PRINTX | R3            | ;PRINT MESSAGE BUFFER ADDRESS                   |
|        | 006340 | 010346 |        | MOV    | R3,-(SP)      | ;PRINT PROPER MESSAGE                           |
|        | 006342 | 012746 | 000001 | MOV    | #1,-(SP)      |                                                 |
|        | 006346 | 010600 |        | MOV    | SP,R0         |                                                 |
|        | 006350 | 104415 |        | TRAP   | C1PNTX        |                                                 |
|        | 006352 | 062706 | 000004 | ADD    | #4,SP         |                                                 |
| 737    | 006356 | 000207 |        | RTS    | PC            | ;RETURN TO CALLER                               |
| 738    |        |        |        |        |               |                                                 |
| 744    | 006360 |        |        | EPRT2: |               |                                                 |
| 745    | 006360 | 045    | 116    | 045    | EPRT1: .ASCIZ | '###A *****REPLACE M7196*****'                  |
| 746    |        |        |        |        |               |                                                 |
| 756    | 006415 | 045    | 116    | 045    | TSSRFOR:      | .ASCIZ '###A TSSR = #06'                        |
| 757    | 006435 | 045    | 116    | 045    | TEXASC:       | .ASCIZ '###A Extended Address Bits = #06'       |
| 758    | 006476 | 045    | 116    | 045    | TCOASC:       | .ASCIZ '###A Termination Class Code = #1'       |
| 759    | 006537 | 045    | 116    | 045    | TFCASC:       | .ASCIZ '###A Fatal Termination Class Code = #1' |
| 760    | 006606 | 045    | 116    | 045    | TSSDEF:       | .ASCIZ '###A TSSR Bits Set: #1'                 |
| 761    | 006635 | 045    | 116    | 045    | AMBTSSR:      | .ASCIZ '###A TSSR Contents Are Ambiguous'       |
| 762    |        |        |        |        |               |                                                 |

.EVEN

D4

ISV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14

SEQ 042

PRITSSR - PRINT TSSR CONTENTS

```

763 006676 006716 006741 006767 TCOCOD: .WORD 11,21,31,41,51,61,71,81
764 006716 116 157 162 11: .ASCIZ 'Normal Termination'
765 006741 124 145 162 21: .ASCIZ 'Termination Condition'
766 006767 124 141 160 31: .ASCIZ 'Tape Status Alert'
767 007011 106 165 156 41: .ASCIZ 'Function Reject'
768 007031 122 145 143 51: .ASCIZ 'Recoverable Error - Tape Position One Record Down'
769 007113 122 145 143 61: .ASCIZ 'Recoverable Error - Tape Was Not Moved'
770 007162 125 156 162 71: .ASCIZ 'Unrecoverable Error'
771 007206 106 141 164 81: .ASCIZ 'Fatal Controller Error'
772 .EVEN
773
774 007236 007246 007302 007313 TSFCOD: .WORD 11,21,31,41
775 007246 111 156 164 11: .ASCIZ 'Internal Diagnostic Failure'
776 007302 122 145 163 21: .ASCIZ 'Reserved'
777 007313 102 165 163 31: .ASCIZ 'Bus Interface or Sanity Check Error'
778 007357 122 145 163 41: .ASCIZ 'Reserved'
779 .EVEN
780 .SBTTL PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET
781
782
783 ; THIS ROUTINE PRINTS THE ADDRESS AND CONTENTS OF A COMMAND PACKET.
784 ; THIS ROUTINE IS NORMALLY ONLY CALLED FROM A PRINT ROUTINE.
785
786 ; INPUT:
787
788 ; R0 NUMBER OF WORDS IN PACKET
789 ; R3 HIGH ORDER COMMAND PACKET ADDRESS
790 ; R4 ADDRESS OF COMMAND PACKET
791
792 ; NOTE: R3 IS IGNORED IF THE KTENABLE FLAG IS CLEAR.
793
794
795 PRIPKT:
796
797 SAVREG
798 MOV R0,R5 ;SAVE THE REGISTERS
799 TST KTENABLE ;SAVE NO. OF WORDS IN PACKET
800 BNE 101 ;ABOVE 28K UNDER TEST?
801 CLR R3 ;BR IF YES
802 MOV R3,R1 ;SET HIGH ORDER ADDRESS TO 0
803 MOV R4,R0 ;COPY HIGH ORDER ADDRESS
804 ROL R0 ;GET LOWER ADDRESS
805 ROL R1 ;SHIFT BIT 15 INTO C BIT
806 PRINTB 0PKTADD,R1,R4 ;AND INTO HIGH ORDER.
807 MOV R4,-(SP) ;PRINT PACKET ADDRESS
808 MOV R1,-(SP)
809 MOV 0PKTADD,-(SP)
810 MOV 03,-(SP)
811 MOV SP,R0
812 TRAP C1PNTB
813 ADD 010,SP
814
815 151: MOV R3,R0 ;GET HIGH ORDER ADDRESS
816 BEQ 201 ;BR IF NOT ABOVE 28K.
817 MOV R4,R1 ;GET LOW ORDER ADDRESS
818 JSR PC,SETMAP ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
819 MOV R0,R4 ;GET RETURNED PAR6 ADDRESS BIAS
820 CLR R1 ;SAVE WORD NUMBER
821 MOV (R4)+,R2 ;GET PACKET CONTENTS

```

E4

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET

SEQ 043

```

813 007462          PRINTB  PKTFRM,R1,R2  ;PRINT THE DATA
      007462 010246      MOV      R2,-(SP)
      007464 010146      MOV      R1,-(SP)
      007466 012746 007516  MOV      PKTFRM,-(SP)
      007472 012746 000003  MOV      R3,-(SP)
      007476 010600      MOV      SP,R0
      007500 104414      TRAP      C$PNTB
      007502 062706 000010  ADD      R10,SP
814 007506 005201      INC      R1          ;NEXT WORD NUMBER
815 007510 020105      CMP      R1,R5      ;DONE ALL PACKET WORDS?
816 007512 002762      BLT      25'        ;LOOP TILL ALL DONE
817 007514 000207      RTS              ;RETURN
818
819 007516          045      116      045  PKTFRM: .ASCIZ  'NNA Packet Word #01NA = #06'
820 007554          045      116      045  PKTADD: .ASCIZ  'NNA Packet Address = #01#05'
821                                     .EVEN
822                                     .SBTTL  PRIBOXOR - PRINT EXPD, RECV AND XOR BYTE
823
824                                     ;*
825                                     ;
826                                     ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE DATA BYTE
827                                     ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
828                                     ;
829                                     ;INPUTS:
830                                     ;
831                                     ;      R1      RECEIVED DATA
832                                     ;      R2      EXPECTED DATA
833                                     ;
834                                     ;OUTPUT:
835                                     ;
836                                     ;      R0      XOR OF EXPECTED/RECEIVED DATA
837                                     ;
838                                     ;-
839
840                                     PRIBOXOR::
841                                     SAVREG          ;SAVE THE REGISTERS
842                                     MOV      R2,R3      ;EXPECTED DATA
843                                     XOR      R1,R3      ;FORM THE EXCLUSIVE OR
844                                     MOV      R10,<377>,R0 ;BYTE MASK
845                                     BIC      R0,R1      ;SAVE LOW BYTE RECV
846                                     BIC      R0,R2      ;SAVE LOW BYTE EXPD
847                                     BIC      R0,R3      ;SAVE LOW BYTE XOR
848                                     PRINTB  XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
      007642 010346      MOV      R3,-(SP)
      007644 010146      MOV      R1,-(SP)
      007646 010246      MOV      R2,-(SP)
      007650 012746 007674  MOV      XORBFOR,-(SP)
      007654 012746 000004  MOV      R4,-(SP)
      007660 010600      MOV      SP,R0
      007662 104414      TRAP      C$PNTB
      007664 062706 000012  ADD      R12,SP
849 007670 010300      MOV      R3,R0          ;R0 HAS XOR ON RETURN
850 007672 000207      RTS              ;RETURN TO CALLER
851
852 007674          045      116      045  XORBFOR: .ASCIZ  'NNA EXPD: #03NA RECV: #03NA XOR: #03'
853                                     .EVEN
854                                     .SBTTL  PRIBOXOR - PRINT EXPD, RECV AND XOR

```

F4

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
PRI XOR    PRINT EXPD, RECV AND XOR

SEQ 044

```

855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872 007742
873 007742
874 007746 010203
875 007750
876 007760
    007760 010346
    007762 010146
    007764 010246
    007766 012736 010012
    007772 012746 000004
    007776 010600
    010000 104414
    010002 062706 000012
877 010008 010300
878 010010 000207
879
880 010012 045 116 045 XORFOR: .ASCIZ 'NNNA EXPD: 06NA RECV: 06NA XOR: 06'
881 .EVEN
882 .SBTTL PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT
883
884
885
886
887
888
889
890
891
892
893
894
895
896 010060
897 010060
898 010064 000207
899
900
901
902
903

```

```

; *
; PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
; THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
; INPUTS:
; R1 RECEIVED DATA
; R2 EXPECTED DATA
; OUTPUT:
; R0 XOR OF EXPECTED/RECEIVED DATA
; -
PRI XOR:
    SAVREG                ;SAVE THE REGISTERS
    MOV R2,R3              ;EXPECTED DATA
    XOR R1,R3              ;FORM THE EXCLUSIVE OR
    PRINTB XORFOR,R2,R1,R3 ;PRINT THE MESSAGE
    MOV R3,-(SP)
    MOV R1,-(SP)
    MOV R2,-(SP)
    MOV XORFOR,-(SP)
    MOV #4,-(SP)
    MOV SP,R0
    TRAP C$PRINTB
    ADD #12,SP
    MOV R3,R0              ;R0 HAS XOR ON RETURN
    RTS PC                 ;RETURN TO CALLER

045 XORFOR: .ASCIZ 'NNNA EXPD: 06NA RECV: 06NA XOR: 06'
        .EVEN
        .SBTTL PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT

; *
; ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
; THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
; INPUTS:
; R0 OCTAL VALUE TO CONVERT
; R1 TABLE OF POINTERS TO ASCII EQUIVALENT
; -
PRIEQU:
    SAVREG                ;SAVE THE REGISTERS
    RTS PC                 ;RETURN TO CALLER

        .SBTTL PRI RAM - PRINT RAM ADDRESS
; *
; PRINT CONTROLLER RAM ADDRESS.

```

G4

TSV3 - GLOBAL AREA MACRO M1113 06-FEB-84 17:14  
PRIRAM PRINT RAM ADDRESS

SEQ 045

```

904                                     ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
905                                     ;
906                                     ;INPUTS:
907                                     ;
908                                     ;      R4      RAM ADDRESS
909                                     ;
910                                     ;-
911 010056 PRIRAM: SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
912 010064 PRINTB  RAMFOR,R4                            ;PRINT RAM ADDRESS IN ERROR
913 010072          MOV  R4,-(SP)
          010074 010445 010116 MOV  RAMFOR,-(SP)
          010100 010745 000002 MOV  #2,-(SP)
          010104 010600 MOV  SP,R0
          010106 100014 TRAP  C$PNTB
          010110 060706 000006 ADD  #6,SP
914 010114 000207 RTS  PC                                ;RETURN
915
916 010116      045      116      045 RAMFOR: .ASCIZ 'NWA CONTROLLER RAM ADDRESS = #06'
917                                     .EVEN
918                                     .SBTTL PRIADD - PRINT MEMORY ERROR ADDRESS
919
920                                     ;+
921                                     ;PRINT MEMORY ADDRESS
922                                     ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
923                                     ;
924                                     ;IMPLICIT INPUTS
925                                     ;
926                                     ;      ERRHI   - HIGH ORDER ADDRESS
927                                     ;      ERRLO   - LOW ORDER ADDRESS
928                                     ;
929                                     ;-
930                                     ;PRIADD:
931 010160 SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
932 010160 MOV  ERRHI,R0                            ;GET HIGH ADDRESS
933 010164 010700 002236 MOV  ERRLO,R1                ;GET LOW ADDRESS
934 010170 010701 002240 MOV  R1,R2                ;COPY LOW ADDRESS
935 010174 010102 ROL  R1                            ;SHIFT BIT 15 TO C BIT
936 010176 000101 ROL  R0                            ;SHIFT INTO HIGH ORDER
937 010200 000100 PRINTB  #PRIA0,R0,R2                ;PRINT MEMORY ADDRESS IN ERROR
938 010202          MOV  R2,-(SP)
          010202 010206          MOV  R0,-(SP)
          010204 010015          MOV  #PRIA0,-(SP)
          010206 010745 010230          MOV  #3,-(SP)
          010212 010745 000003          MOV  SP,R0
          010216 010000 TRAP  C$PNTB
          010220 100014          ADD  #10,SP
          010222 060706 000010          RTS  PC                                ;RETURN
939 010226 000207
940
941 010230      045      116      045 PRIA0: .ASCIZ 'NWA MEMORY ERROR ADDRESS = #01#05'
942                                     .EVEN
943                                     .SBTTL PRITADD - PRINT MEMORY TEST ADDRESS
944
945                                     ;+
946                                     ;PRINT MEMORY ADDRESS
947                                     ;

```

TSV3 - GLOBAL AREA MACRO M1113 06-FEB-84 17:14  
 PRITADD - PRINT MEMORY TEST ADDRESS

SEQ 046

```

948 ; THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
949 ;
950 ; IMPLICIT INPUTS
951 ;
952 ; ERRHI - HIGH ORDER ADDRESS
953 ; ERRLO - LOW ORDER ADDRESS
954 ;
955 ;
956 PRITADD: SAVREG ; SAVE R1-R5 UNTIL NEXT RETURN
957 MOV ERRHI,R2 ; GET HIGH ADDRESS
958 MOV ERRLO,R1 ; GET LOW ADDRESS
959 ; MOV R1,R2 ; COPY LOW ADDRESS
960 ; ROL R1 ; SHIFT BIT 15 TO C BIT
961 ; ROL R0 ; SHIFT INTO HIGH ORDER
962 ; PRINTB @PRIT0,R1 ; PRINT MEMORY ADDRESS LOW IN ERROR
963 ; MOV R1,-(SP)
964 ; MOV @PRIT0,-(SP)
965 ; MOV @2,-(SP)
966 ; MOV SP,R0
967 ; TRAP C:PNTB
968 ; ADD @6,SP
969 ; PRINTB @PRIT1,R2 ; PRINT MEMORY ADDRESS HIGH IN ERROR
970 ; MOV R2,-(SP)
971 ; MOV @PRIT1,-(SP)
972 ; MOV @2,-(SP)
973 ; MOV SP,R0
974 ; TRAP C:PNTB
975 ; ADD @6,SP
976 ; RTS PC ; RETURN
977
978 045 PRIT0: .ASCIZ 'UNNA MEMORY TEST ADDRESS LOW = '06'
979 045 PRIT1: .ASCIZ 'UNNA MEMORY TEST ADDRESS HIGH = '06'
980 ; EVEN
981 ; SBTTL SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND
982
983 ;
984 ; ROUTINE TO ISSUE A SPACE RECORDS
985 ; COMMAND (FORWARD OR REVERSE)
986 ;
987 ; INPUT:
988 ;
989 ; R3 NUMBER OF RECORDS TO BE SPACED OVER
990 ; BIT15 CONTROLS DIRECTION
991 ; BIT15 = 0 IS FORWARD
992 ; BIT15 = 1 IS REVERSE
993 ; R5 FIRST DEVICE UNIBUS ADDRESS
994 ;
995 ; REQUIRES A WRITE CHARACTERISTICS DONE PREVIOUSLY
996 ;
997 ; OUTPUT:
998 ;
999 ; CARRY SET - SPACE RECORDS COMMAND OK
1000 ; CLR - SPACE RECORDS FAILED
1001 ;
1002 ;

```

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 SPACE SPACE RECORDS (FORWARD AND REVERSE) COMMAND

SEQ 047

```

993      ;      RO      THE CONTENTS OF R4 IS MOVED TO RO
994      ;
995      ;
996      ;IMPLICIT OUTPUT:
997      ;
998      ;      TAPE HAS BEEN MOVED
999      ;
1000     ;SIDE EFFECTS:
1001     ;
1002     ;
1003     ;
1004     ;
1005     SPACE::
1006     SAVREG
1007     MOV     $500.,SDELAY
1008     MOV     $140010,80$
1009     TST     R3
1010     BMI     5$
1011     MOV     R3,90$
1012     BR      10$
1013     BIC     $BIT15,R3
1014     MOV     R3,90$
1015     BIS     $BIT8,80$
1016     MOV     $80$,R4
1017     MOV     R4,TSSDB(R5)
1018     JSR     PC,WAITF
1019     BCS     20$
1020     DELAY    250
1021     MOV     $250,(PC)
1022     ;
1023     ;$DLY,(PC)
1024     ;
1025     ;-6(PC)
1026     BNE     -4
1027     DEC     -22(PC)
1028     BNE     -20
1029     DEC     SDELAY
1030     BNE     15$
1031     BR      60$
1032     MOV     TSSR(R5),R1
1033     MOV     $SSR,R2
1034     CMP     R2,R1
1035     BEQ     40$
1036     BR      60$
1037     SEC
1038     BR      70$
1039     CLC
1040     MOV     R4,RO
1041     RTS     PC
1042
;SAVE THE GENERAL REGISTERS
;SET UP DELAY
;SET UP COMMAND, SPACE FORWARD
;CHECK FOR DIRECTION
;BR, IF REVERSE INDICATED
;LOAD UP NUMBER OF RECORDS TO SPACE
;GO DO COMMAND
;CLEAR DIRECTION BIT
;LOAD UP NUMBER OF RECORDS TO SPACE
;SET REVERSE BIT IN COMMAND PACKET
;SET UP R4 WITH PACKET ADDRESS
;SEND OUT COMMAND
;WAIT FOR SSR
;BR, IF SSR IS SET AND OK
;DELAY ABOUT .25 SECONDS
;BUMP DELAY COUNTER DOWN
;BR, IF MORE DELAY
;BR IF TROUBLE CARRY = CLEAR
;READ TSSR
;SET UP EXPECTED
;ARE THEY OK
;BR, IF EQUAL = OK
;TROUBLE EXIT
;SET CARRY NO TROUBLE
;EXIT
;CARRY CLEAR = ERROR
;PASS PACKET ADDRESS
;RETURN

```

J4

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
 SPACE      SPACE RECORDS (FORWARD AND REVERSE) COMMAND

SEQ 048

```

1036      ;
1037      ;
1038      ;
1039      ; PACKET FOR SPACE COMMAND
1040      ;
1041      ;
1042      ;      .=<.10>8177770
1043      ;
1044      ; COMMAND WORD
1045      010650 000000 80$: .WORD
1046      ; NUMBER OF RECORDS TO BE SPACED OVER WORD
1047      ;
1048      010652 000000 90$: .WORD
1049      010654 000000 .WORD
1050      010656 000000 .WORD
1051      010660 000000 SDELAY: .WORD 0 ; DELAY COUNTER
1052      ; EVEN
1053      ; SBTTL WRCHR - WRITE CHARACTERISTICS COMMAND
1054      ;
1055      ;
1056      ;
1057      ; ROUTINE TO ISSUE A WRITE CHARACTERISTICS
1058      ; COMMAND SO THAT OTHER COMMANDS WILL BE ACCEPTED
1059      ;
1060      ; INPUT:
1061      ;
1062      ;      R4      ADDRESS OF PACKET FROM TEST
1063      ;      R5      FIRST DEVICE UNIBUS ADDRESS
1064      ;      REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
1065      ;
1066      ; OUTPUT:
1067      ;
1068      ;      R0      TSSR CONTENTS
1069      ;      CARRY   SET - WRITE CHARACTERISTICS COMMAND OK
1070      ;           CLR - WRITE CHARACTERISTICS FAILED
1071      ;
1072      ; IMPLICIT OUTPUT:
1073      ;
1074      ;      MESSAGE BUFFER AND OTHER BUFFERS ALL SET UP
1075      ;      SOFTWARE SWITCHES SET AS FOLLOWS:
1076      ;           EXTFEA = EXTENDED FEATURES PRESENT
1077      ;           BENBSW = BUFFER ENABLE SWITCH ON OR OFF
1078      ;
1079      ;
1080      ; SIDE EFFECTS:
1081      ;
1082      ;
1083      ;
1084      ;
1085      010662 WRCHR:
1086      010662      SAVREG      ; SAVE THE GENERAL REGISTERS
1087      010666 005037 002230      CLR      BENBSW      ; CLEAR BUFFER ENABLE SWITCH
1088      010672 005037 002226      CLR      EXTFEA      ; CLEAR EXTENDED FEATURES SW SWITCH
1089      010676 010465 000000      10$: MOV      R4,TSDB(R5)      ; SEND OUT COMMAND
1090      010702 004737 016336      JSR      PC,CHKTSSR      ; WAIT FOR SSR
1091      010706 103401      BCS      20$      ; BR, IF SSR IS SET AND OK
1092      010710 000435      BR      60$      ; BR IF TROUBLE CARRY = CLEAR
1093      010712 016501 000002      20$: MOV      TSSR(R5),R1      ; READ TSSR
1094      010716 012702 000200      MOV      #SSR,R2      ; SET UP EXPECTED

```

TSV3 - GLOBAL AREAS MACRO 11113 06-FEB-84 17:14  
WRTCHR - WRITE CHARACTERISTICS COMMAND

SEQ 049

```

1095 010722 032701 000100      BIT      40FL,R1      ;WAS OFF LINE SET IN TSSR
1096 010726 001402              BEQ      25$      ;BR, IF NO OFL SET
1097 010730 052702 000100      BIS      40FL,R2      ;MAKE THEM LOOK ALIKE
1098 010734 020201              CMPL    R2,R1      ;ARE THEY OK
1099 010736 001401              BEQ      40$      ;BR, IF EQUAL = OK
1100 010740 000421              BR       60$      ;TROUBLE EXIT
1101 010742 062704 000010      40$:    ADD      4B.,R4      ;POINT TO WRT CHARA DATA PACKET
1102 010746 011403              MOV      (R4),R3      ;GET ADDRESS OF MESSAGE BUFFER
1103 010750 032763 000200 000012  BIT      4X2.EXTF,XST2(R3) ;EXTENDED FEATURES BIT SET?
1104 010756 001402              BEQ      45$      ;BR IF NO
1105 010760 005237 002226      INC      EXTFEA      ;SET EXTENDED FEA 'OFF' SW SWITCH
1106 010764              45$:    BIT      4X2.BUFE,XST2(R3) ;BUFFER ENABLE SWITCH SET
1107 010764 032763 000100 000012  BEQ      50$      ;BR, IF SWITCH NOT SET
1108 010772 001402              INC      BENBSW      ;SET SOFTWARE SWITCH FOR ENABLED
1109 010774 005237 002230      50$:    SEC              ;SET CARRY NO TROUBLE
1110 011000              BR       70$      ;EXIT
1111 011000 000261              CLC              ;CARRY CLEAR = ERROR
1112 011002 000401              60$:    CLC              ;CARRY CLEAR = ERROR
1113 011004 000241              70$:    MOV      TSSR(R5),R0 ;RETURN TSSR CONTENTS
1114 011006 016500 000002      RTS      PC          ;RETURN
1115 011012 000207              .SBTTL  REWIND - POSITION TAPE (REWIND) COMMAND
1116
1117
1118
1119
1120      ; THIS ROUTINE WILL REWIND THE SELECTED TAPE.
1121
1122      ; CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
1123      ; TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
1124      ; SSR TO SET IN THE TSSR
1125
1126
1127      ; CALLING SEQUENCE:
1128
1129      ; DO A SOFT INIT
1130      ; DO A WRITE CHARACTERISTICS
1131      ; JSR PC,REWIND
1132
1133      ; INPUT:
1134
1135      ; R5 FIRST DEVICE UNIBUS ADDRESS
1136
1137
1138      ; OUTPUT
1139
1140      ; R0 THE CONTENTS OF R4 IS PASSED TO R0
1141
1142
1143
1144      REWIND::
1145      SAVREG      ;SAVE R1-R5 UNTIL NEXT RETURN
1146      MOV      4RWPack,R4 ;GET PACKET ADDRESS
1147      MOV      R4,TSD8(R5) ;SEND PACKET ADDRESS TO EXECUTE
1148      MOV      4360.,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND
1149      JSR      PC,WAITF ;WAIT FOR SSR TO SET
1150      BCS      20$ ;LEAVE WHEN SSR IS SET
1151      DELAY     250. ;WAIT FOR .25 SECONDS

```

L4

TSV3 - GLOBAL AREAS  
REWIND - POSITION TAPE (REWIND) COMMAND

SEQ 050

|        |        |        |           |                 |                                 |
|--------|--------|--------|-----------|-----------------|---------------------------------|
| 011042 | 012727 | 000372 | MCV       | #250.,(PC)+     |                                 |
| 011046 | 000000 |        | .WORD     | 0               |                                 |
| 011050 | 013727 | 002116 | MOV       | L\$DLY,(PC)+    |                                 |
| 011054 | 000000 |        | .WORD     | 0               |                                 |
| 011056 | 005367 | 177772 | DEC       | -6(PC)          |                                 |
| 011062 | 001375 |        | BNE       | .-4             |                                 |
| 011064 | 005367 | 177756 | DEC       | -22(PC)         |                                 |
| 011070 | 001367 |        | BNE       | .-20            |                                 |
| 1152   | 011072 | 005303 | DEC       | R3              | ;BUMP COUNTER DOWN              |
| 1153   | 011074 | 001357 | BNE       | 10\$            | ;KEEP GOING                     |
| 1154   | 011076 | 000241 | CLC       |                 | ;CLEAR CARRY TO SET ERROR       |
| 1155   | 011100 | 010400 | 20\$: MOV | R4,R0           | ;PASS THE PACKET ADDRESS        |
| 1156   | 011102 | 000207 | RTS       | PC              | ;RETURN                         |
| 1157   |        |        |           |                 |                                 |
| 1159   |        | 011110 |           | .=<.+10>E177770 |                                 |
| 1161   | 011110 |        | RWPACK:   |                 |                                 |
| 1162   | 011110 | 102010 | .WORD     | 102010          | ;POSTION COMMAND (REWIND)       |
| 1163   | 011112 | 000000 | .WORD     | 0               | ;NOT USED                       |
| 1164   |        |        | .SBTTL    | CKRAM           | - COMPARE RAM TO I/O PACKET     |
| 1165   |        |        |           |                 |                                 |
| 1166   |        |        |           |                 |                                 |
| 1167   |        |        |           |                 |                                 |
| 1168   |        |        |           |                 |                                 |
| 1169   |        |        |           |                 |                                 |
| 1170   |        |        |           |                 |                                 |
| 1171   |        |        |           |                 |                                 |
| 1172   |        |        |           |                 |                                 |
| 1173   |        |        |           |                 |                                 |
| 1174   |        |        |           |                 |                                 |
| 1175   |        |        |           |                 |                                 |
| 1176   |        |        |           |                 |                                 |
| 1177   |        |        |           |                 |                                 |
| 1178   |        |        |           |                 |                                 |
| 1179   |        |        |           |                 |                                 |
| 1180   |        |        |           |                 |                                 |
| 1181   |        |        |           |                 |                                 |
| 1182   |        |        |           |                 |                                 |
| 1183   |        |        |           |                 |                                 |
| 1184   |        |        |           |                 |                                 |
| 1185   |        |        |           |                 |                                 |
| 1186   |        |        |           |                 |                                 |
| 1187   |        |        |           |                 |                                 |
| 1188   |        |        |           |                 |                                 |
| 1189   |        |        |           |                 |                                 |
| 1190   |        |        |           |                 |                                 |
| 1191   |        |        |           |                 |                                 |
| 1192   |        |        |           |                 |                                 |
| 1193   | 011114 |        | CKRAM::   |                 |                                 |
| 1194   | 011114 |        | SAVREG    |                 | ;SAVE THE GENERAL REGISTERS     |
| 1195   | 011120 | 012701 | MOV       | #RAMDATA,R1     | ;ADDRESS TO SAVE THE RAM DATA   |
| 1196   | 011124 | 012702 | MOV       | #RMPKTBEG,R2    | ;BYTE ADDRESS OF FIRST RAM DATA |
| 1197   | 011130 | 005003 | CLR       | R3              | ;CLEAR THE ERROR FLAG           |
| 1198   | 011132 | 004737 | JSR       | PC,CHKTSSR      | ;WAIT FOR SSR                   |
| 1199   | 011136 | 112765 | MOV       | #0,TSDB(R5)     | ;SET MAINTENANCE MODE           |
| 1200   | 011144 | 004737 | 10\$: JSR | PC,CHKTSSR      | ;WAIT FOR SSR TO SET            |
| 1201   | 011150 | 010265 | MOV       | R2,TSDB(R5)     | ;SELECT NEXT RAM ADDRESS        |
| 1202   | 011154 | 004737 | JSR       | PC,CHKTSSR      | ;WAIT FOR SSR TO SET            |

M4

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
CKRAM - COMPARE RAM TO I/O PACKET

SEQ 051

```

1203 011160 116511 000000      MOVB    TSBA(R5),(R1)      ; READ THE RAM DATA
1204 011164 122124      CMPB    (R1)+,(R4)+      ; COMPARE TO EXPECTED
1205 011166 001401      BEQ     20$              ; BRANCH IF OK
1206 011170 005203      INC     R3                ; SET ERROR FLAG
1207 011172 005202      20$: INC     R2            ; ADDRESS OF NEXT RAM LOCATION
1208 011174 020227 000210      CMP     R2,#RMPKTEND ; REACHED END YET ?
1209 011200 003761      BLE     10$              ; BRANCH TILL ALL READ
1210 011202 005703      TST     R3                ; WAS AN ERROR FOUND ?
1211 011204 001402      BEQ     30$              ; BRANCH IF NOT
1212 011206 000241      CLC                     ; CLEAR CARRY TO SHOW ERROR
1213 011210 000401      BR      50$              ; AND EXIT
1214 011212 000261      30$: SEC                     ; SHOW GOOD COMPARE
1215 011214 012737 000010 002302 50$: MOV     #8.,RAMSIZ ; SETUP RAMSIZ FOR PRAMPKT ROUTINE
1216 011222 000207      RTS      PC              ; RETURN
1217                                     .SBTTL  CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
1218                                     ;+
1219                                     ;
1220                                     ; ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
1221                                     ; MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
1222                                     ;
1223                                     ; INPUT:
1224                                     ;
1225                                     ; R4      ADDRESS OF THE CHARACTERISTICS DATA
1226                                     ; R5      FIRST DEVICE UNIBUS ADDRESS
1227                                     ;
1228                                     ; OUTPUT:
1229                                     ;
1230                                     ; CARRY   SET - RAM MATCHES PACKET
1231                                     ;          CLR - RAM DOES NOT MATCH PACKET
1232                                     ;
1233                                     ; IMPLICIT OUTPUT:
1234                                     ;
1235                                     ; THE TABLE RAMDATA IS FILLED WITH THE
1236                                     ; DATA HELD IN RAM.
1237                                     ; RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
1238                                     ;
1239                                     ; SIDE EFFECTS:
1240                                     ;
1241                                     ; THE SUBSYSTEM IS LEFT IN MAINTENANCE MODE
1242                                     ;
1243                                     ; -
1244
1245 011224      CKRAM2: SAVREG
1246 011224      MOV     #RAMDATA,R1              ; SAVE THE GENERAL REGISTERS
1247 011230 012701 002242      MOV     #RMCHBEG,R2 ; ADDRESS TO SAVE THE RAM DATA
1248 011234 012702 000167      CLR     R3          ; BYTE ADDRESS OF FIRST RAM DATA
1249 011240 005003      JSR     PC,CHKTSSR        ; CLEAR THE ERROR FLAG
1250 011242 004737 016336      JSR     PC,CHKTSSR ; WAIT FOR SSR
1251 011246 112765 000000 000000      MOVB    #0,TSDB(R5) ; SET MAINTENANCE MODE
1252 011254 004737 016336      10$: JSR     PC,CHKTSSR ; WAIT FOR SSR TO SET
1253 011260 010205 000000      MOV     R2,TSDB(R5) ; SELECT NEXT RAM ADDRESS
1254 011264 004737 016336      JSR     PC,CHKTSSR ; WAIT FOR SSR TO SET
1255 011270 116511 000000      MOVB    TSBA(R5),(R1) ; READ THE RAM DATA
1256 011274 122124      CMPB    (R1)+,(R4)+      ; COMPARE TO EXPECTED
1257 011276 001401      BEQ     20$              ; BRANCH IF OK
1258 011300 005203      INC     R3                ; SET ERROR FLAG
1259 011302 005202      20$: INC     R2            ; ADDRESS OF NEXT RAM LOCATION

```

```
TSV3 - GLOBAL AREAS      MACRO M1113  06-FEB-84 17:14
CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
```

SEQ 052

| PC | Op | OpC | OpD | OpE | OpF | OpG | OpH | OpI | OpJ | OpK | OpL | OpM | OpN | OpO | OpP | OpQ | OpR | OpS | OpT | OpU | OpV | OpW | OpX | OpY | OpZ | OpAA | OpAB | OpAC | OpAD | OpAE | OpAF | OpAG | OpAH | OpAI | OpAJ | OpAK | OpAL | OpAM | OpAN | OpAO | OpAP | OpAQ | OpAR | OpAS | OpAT | OpAU | OpAV | OpAW | OpAX | OpAY | OpAZ | OpBA | OpBB | OpBC | OpBD | OpBE | OpBF | OpBG | OpBH | OpBI | OpBJ | OpBK | OpBL | OpBM | OpBN | OpBO | OpBP | OpBQ | OpBR | OpBS | OpBT | OpBU | OpBV | OpBW | OpBX | OpBY | OpBZ | OpCA | OpCB | OpCC | OpCD | OpCE | OpCF | OpCG | OpCH | OpCI | OpCJ | OpCK | OpCL | OpCM | OpCN | OpCO | OpCP | OpCQ | OpCR | OpCS | OpCT | OpCU | OpCV | OpCW | OpCX | OpCY | OpCZ | OpDA | OpDB | OpDC | OpDD | OpDE | OpDF | OpDG | OpDH | OpDI | OpDJ | OpDK | OpDL | OpDM | OpDN | OpDO | OpDP | OpDQ | OpDR | OpDS | OpDT | OpDU | OpDV | OpDW | OpDX | OpDY | OpDZ | OpEA | OpEB | OpEC | OpED | OpEE | OpEF | OpEG | OpEH | OpEI | OpEJ | OpEK | OpEL | OpEM | OpEN | OpEO | OpEP | OpEQ | OpER | OpES | OpET | OpEU | OpEV | OpEW | OpEX | OpEY | OpEZ | OpFA | OpFB | OpFC | OpFD | OpFE | OpFF | OpFG | OpFH | OpFI | OpFJ | OpFK | OpFL | OpFM | OpFN | OpFO | OpFP | OpFQ | OpFR | OpFS | OpFT | OpFU | OpFV | OpFW | OpFX | OpFY | OpFZ | OpGA | OpGB | OpGC | OpGD | OpGE | OpGF | OpGG | OpGH | OpGI | OpGJ | OpGK | OpGL | OpGM | OpGN | OpGO | OpGP | OpGQ | OpGR | OpGS | OpGT | OpGU | OpGV | OpGW | OpGX | OpGY | OpGZ | OpHA | OpHB | OpHC | OpHD | OpHE | OpHF | OpHG | OpHH | OpHI | OpHJ | OpHK | OpHL | OpHM | OpHN | OpHO | OpHP | OpHQ | OpHR | OpHS | OpHT | OpHU | OpHV | OpHW | OpHX | OpHY | OpHZ | OpIA | OpIB | OpIC | OpID | OpIE | OpIF | OpIG | OpIH | OpII | OpIJ | OpIK | OpIL | OpIM | OpIN | OpIO | OpIP | OpIQ | OpIR | OpIS | OpIT | OpIU | OpIV | OpIW | OpIX | OpIY | OpIZ | OpJA | OpJB | OpJC | OpJD | OpJE | OpJF | OpJG | OpJH | OpJI | OpJJ | OpJK | OpJL | OpJM | OpJN | OpJO | OpJP | OpJQ | OpJR | OpJS | OpJT | OpJU | OpJV | OpJW | OpJX | OpJY | OpJZ | OpKA | OpKB | OpKC | OpKD | OpKE | OpKF | OpKG | OpKH | OpKI | OpKJ | OpKK | OpKL | OpKM | OpKN | OpKO | OpKP | OpKQ | OpKR | OpKS | OpKT | OpKU | OpKV | OpKW | OpKX | OpKY | OpKZ | OpLA | OpLB | OpLC | OpLD | OpLE | OpLF | OpLG | OpLH | OpLI | OpLJ | OpLK | OpLL | OpLM | OpLN | OpLO | OpLP | OpLQ | OpLR | OpLS | OpLT | OpLU | OpLV | OpLW | OpLX | OpLY | OpLZ | OpMA | OpMB | OpMC | OpMD | OpME | OpMF | OpMG | OpMH | OpMI | OpMJ | OpMK | OpML | OpMM | OpMN | OpMO | OpMP | OpMQ | OpMR | OpMS | OpMT | OpMU | OpMV | OpMW | OpMX | OpMY | OpMZ | OpNA | OpNB | OpNC | OpND | OpNE | OpNF | OpNG | OpNH | OpNI | OpNJ | OpNK | OpNL | OpNM | OpNN | OpNO | OpNP | OpNQ | OpNR | OpNS | OpNT | OpNU | OpNV | OpNW | OpNX | OpNY | OpNZ | OpOA | OpOB | OpOC | OpOD | OpOE | OpOF | OpOG | OpOH | OpOI | OpOJ | OpOK | OpOL | OpOM | OpON | OpOO | OpOP | OpOQ | OpOR | OpOS | OpOT | OpOU | OpOV | OpOW | OpOX | OpOY | OpOZ | OpPA | OpPB | OpPC | OpPD | OpPE | OpPF | OpPG | OpPH | OpPI | OpPJ | OpPK | OpPL | OpPM | OpPN | OpPO | OpPP | OpPQ | OpPR | OpPS | OpPT | OpPU | OpPV | OpPW | OpPX | OpPY | OpPZ | OpQA | OpQB | OpQC | OpQD | OpQE | OpQF | OpQG | OpQH | OpQI | OpQJ | OpQK | OpQL | OpQM | OpQN | OpQO | OpQP | OpQQ | OpQR | OpQS | OpQT | OpQU | OpQV | OpQW | OpQX | OpQY | OpQZ | OpRA | OpRB | OpRC | OpRD | OpRE | OpRF | OpRG | OpRH | OpRI | OpRJ | OpRK | OpRL | OpRM | OpRN | OpRO | OpRP | OpRQ | OpRR | OpRS | OpRT | OpRU | OpRV | OpRW | OpRX | OpRY | OpRZ | OpSA | OpSB | OpSC | OpSD | OpSE | OpSF | OpSG | OpSH | OpSI | OpSJ |
|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|

B5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS

SEQ 053

```

1317 0114 0 020427 000014      CMP      R4,014      ;DONE FIRST 7 WORDS?
1318 0114 44 003764      BLE      15;      ;BR IF NO
1319 0114 46 032765 000200 000012      BIT      0X2,EXTF,XST2(R5);IS EXTENDED FEATURES SET IN EXPD?
1320 0114 54 001403      BEQ      50;      ;BR IF NO
1321 011456 020427 000016      CMP      R4,016      ;DONE EXTENDED FEATURES WORD?
1322 011462 003755      BLE      15;      ;BR IF NO
1323 011464 005703      50;:      TST      R3      ;ANY ERRORS SEEN?
1324 011466 001402      BEQ      55;      ;BR IF NO
1325 011470 000241      CLC      ;SET FAILURE
1326 011472 000401      BR      60;      ;
1327 011474 000261      55;:      SEC      ;SET SUCCESS
1328 011476 000207      60;:      RTS      PC      ;RETURN
1329      ;SBTTL CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS
1330      ;
1331      ;
1332      ;ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
1333      ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
1334      ;ERROR PRINT ROUTINES.
1335      ;
1336      ;INPUT:
1337      ;
1338      ;      R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
1339      ;      R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
1340      ;      R2      EXPD MESSAGE BUFFER ADDRESS
1341      ;      R3      NUMBER OF BYTES TO COMPARE
1342      ;
1343      ;OUTPUT:
1344      ;
1345      ;      CARRY    SET - MESSAGE BUFFERS MATCH
1346      ;              CLR - MESSAGE BUFFERS DON'T MATCH
1347      ;
1348      ;IMPLICIT OUTPUT:
1349      ;
1350      ;      EXPMSG    BUFFER IS SET TO EXPD DATA
1351      ;      RECVMSG    BUFFER IS SET TO RECV DATA
1352      ;      RCVHIADD    SET TO HIGH ORDER ADDRESS OF RECV
1353      ;      RCVLOADD    SET TO LOW ORDER ADDRESS OF RECV
1354      ;
1355      ;
1356 011500      CKMSG2::
1357 011500      SAVREG      ;SAVE R1-R5 UNTIL NEXT RETURN
1358 011504 020327 000144      CMP      R3,0RECVMSG-EXPMSG;000 IS COUNT ABOVE MAX ALLOWED?
1359 011510 003412      BLE      5;      ;000 BR IF NO
1360 011512 012703 000144      MOV      0RECVMSG-EXPMSG,R3;000
1361 011516      PRINTF      0DEBUGMSG      ;000
1362      011516 012746 011632      MOV      0DEBUGMSG,-(SP)
1363      011522 012746 000001      MOV      01,(SP)
1364      011526 010600      MOV      SP,R0
1365      011530 104417      TRAP      C0PNTF
1366      011532 062706 000004      ADD      04,SP
1367 011536 010037 002304      5;:      MOV      R0,RCVHIADD      ;SAVE RECV HIGH ADDRESS
1368 011542 010137 002306      MOV      R1,RCVLOAD      ;SAVE RECV LOW ADDRESS
1369 011546 005737 003134      TST      KTNABLE      ;TESTING ABOVE 20K?
1370 011552 001403      BEQ      10;      ;BR IF NO
1371 011554 004737 017316      JSR      PC,SETMAP      ;RETURN ADDRESS BIASED TO PAR6 IN R0
1372 011560 010001      MOV      R0,R1      ;GET RETURNED ADDRESS BIASED TO PAR6
1373 011562 005004      10;:      CLR      R4      ;WORD IN BUFFER

```

C5

 ISV3 GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

SEQ 054

```

1369 011564 005005          CLR      R5          ;CLEAR ERROR SEEN FLAG
1370 011566 111264 002322 15$:  MOVB     (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
1371 011572 111164 002466      MOVB     (R1),RECVMSG(R4) ;SAVE RECV FOR ERROR REPORT
1372 011576 122221          CMPB     (R2)+,(R1)+      ;EXPD EQUAL RECV?
1373 011600 001401          BEQ      25$          ;BR IF YES
1374 011602 005205          INC      R5          ;SET ERROR SEEN FLAG
1375 011604 062704 000001 25$:  ADD      #1,R4      ;POINT TO NEXT BYTE
1376 011610 020403          CMP      R4,R3      ;DONE ALL BYTES?
1377 011612 002001          BGE      50$          ;BR IF YES
1378 011614 000764          BR       15$          ;DO NEXT BYTE
1379 011616 005705          50$:  TST      R5          ;ANY ERRORS SEEN?
1380 011620 001402          BEQ      55$          ;BR IF NO
1381 011622 000241          CLC          ;SET FAILURE
1382 011624 000401          BR       60$          ;
1383 011626 000261          55$:  SEC          ;SET SUCCESS
1384 011630 000207          60$:  RTS      PC      ;RETURN
1385
1386 011632      120      122      117  DEBUGMSG:      .ASCIZ  'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-' ;000
1387 011722      045      116      045  FERCM:      .ASCII  /NMA ***/
1388 011733      040      040      124  ERCM:      .ASCIZ  / TSSR ERROR CODE REC'D * /
1389 011766      056      056      056  SIMSG:      .ASCIZ  /.... AFTER DOING SOFT INIT/
1390 012021      124      105      123  TINERR:      .ASCIZ  /TEST: .../
1391
1392
1393
1394
1395      ;
1396      ;PRINT ROUTINE TO FATAL SOFT INIT ERRORS
1397      ;
1398      ;INPUT:
1399      ;
1400      ;      R1      CONTENTS OF TSSR AT ERROR
1401      ;
1402      ;SIDE EFFECTS:
1403      ;
1404      ;      EXECUTES DROP UNIT TO CEASE TESTING
1405      ;
1406      ;
1407 012034      BGNMSG  SFIMSG
1408 012034      SFIMSG: ;
1409 012040      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
1410 012044      JSR      PC,CKDROP      ;DROP UNIT, IF ALLOWED
1411 012044      ENDMMSG
1412
1413 L10003:      TRAP      CMSG
1414
1415      ;
1416      ;PRINT ROUTINE TO PRINT THE CONTENTS OF
1417      ;TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
1418      ;
1419      ;INPUTS:
1420      ;
1421      ;      R1      TSSR CONTENTS
1422      ;      R4      ADDRESS OF COMMAND PACKET

```

D5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

SEQ 055

```

1423 012046          BGNMSG  PKTSSR
      012046          PKTSSR:
1424 012046 004737 006024      JSR      PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
1425 012052 012700 000004      MOV      #4,R0          ;NO. OF WORDS IN PACKET
1426 012056 004737 007370      JSR      PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
1427 012062          ENDMSG
      012062          L10004:
      012062 104423          TRAP      C#MSG

1428
1429
1430      ;
1431      ;PRINT ROUTINE TO PRINT THE CONTENTS OF
1432      ;TSSR AND A GET STATUS COMMAND PACKET.
1433      ;
1434      ;INPUTS:
1435      ;
1436      ;      R1      TSSR CONTENTS
1437      ;      R4      ADDRESS OF COMMAND PACKET
1438      ;
1439      ;-

1440 012064          BGNMSG  PKTGETS
      012064          PKTGETS:
1441 012064 004737 006024      JSR      PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
1442 012070 012700 000002      MOV      #2,R0          ;NO. OF WORDS IN GET STATUS PACKET
1443 012074 004737 007370      JSR      PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
1444 012100          ENDMSG
      012100          L10005:
      012100 104423          TRAP      C#MSG

1445
1446
1447      ;
1448      ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
1449      ;
1450      ;INPUTS:
1451      ;
1452      ;      R1      TSSR CONTENTS
1453      ;      R4      ADDRESS OF COMMAND PACKET
1454      ;
1455      ;-

1456 012102          BGNMSG  SFFMSG
      012102          SFFMSG:
1457 012102 004737 006024      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
      012106          ENDMSG
      012106          L10006:
      012106 104423          TRAP      C#MSG

1458
1459      .SBTTL  PKTIMES - PRINT TSSR AND MESSAGE BUFFER
1460
1461      ;
1462      ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
1463      ;BUFFER FOR ERROR REPORTS
1464      ;
1465      ;INPUTS:
1466      ;
1467      ;      R1      CONTENTS OF TSSR
1468      ;      R2      LOW ORDER MESSAGE BUFFER
1469      ;      R3      HIGH ORDER MESSAGE BUFFER ADDRESS
1470      ;      NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR

```

ES

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
PKTMES - PRINT TSSR AND MESSAGE BUFFER

SEQ 056

1471  
1472 012110  
012110  
1473 012110 004737 006024  
1474 012114 010200  
1475 012116 010301  
1476 012120 004737 014242  
1477 012124  
012124  
012124 104423  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490 012126  
012126  
1491 012126 004737 010274  
1492 012132 016501 000002  
1493 012136 004737 006024  
1494 012142  
012142  
012142 104423  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508 012144  
012144  
1509 012144 012700 000007  
1510 012150 005737 002226  
1511 012154 001402  
1512 012156 012700 000010  
1513 012162 004737 014552  
1514 012166  
012166  
012166 104423  
1515  
1516  
1517  
1518

```

; -
BGNMSG PKTMES
PKTMES:
    JSR    PC,PRITSSR    ;PRINT CONTENTS OF TSSR
    MOV    R2,R0         ;LOW ORDER ADDRESS
    MOV    R3,R1         ;HIGH ORDER ADDRESS
    JSR    PC,PRMESS     ;PRINT THE MESSAGE BUFFER
    ENDMSG
L10007:
    TRAP    C#MSG
    .SBTTL  ADDSSR - PRINT TEST ADDRESS AND TSSR
;+
;PRINT ROUTINE TO PRINT THE CONTENTS OF
;TSSR AND A MEMORY TEST ADDRESS
;
;INPUTS:
;
;    R5      FIRST DEVICE UNIBUS ADDRESS
;    ERRHI   HIGH ORDER MEMORY TEST ADDRESS
;    ERRLO   LOW ORDER MEMORY TEST ADDRESS
; -
BGNMSG ADDSSR
ADDSSR:
    JSR    PC,PRITADD    ;PRINT MEMORY TEST ADDRESS
    MOV    TSSR(R5),R1   ;GET CURRENT TSSR
    JSR    PC,PRITSSR    ;PRINT THE CONTENTS OF TSSR REGISTER
    ENDMSG
L10010:
    TRAP    C#MSG
    .SBTTL  MSGEXP - PRINT WRITE CHAR. EXPD-RCV MESSAGE BUFFERS
;+
;PRINT ROUTINE TO PRINT WRITE CHARACTERISTIC MESSAGE BUFFER
;
;IMPLICIT INPUTS:
;
;    EXPMSG  - EXPECTED MESSAGE BUFFER
;    RECMMSG - RECEIVED MESSAGE BUFFER
;    RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
;    RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
; -
BGNMSG MSGEXP
MSGEXP:
    MOV    #7,R0         ;ASSUME NO EXT FEATURES
    TST    EXTFEA        ;EXT FEATURES SET?
    BEQ    5$            ;BR IF NO
    MOV    #8,,R0        ;EXT FEATURE BUFFER IS 8 WORDS
    JSR    PC,PRMSGEXP   ;PRINT EXPD/RCV MESSAGE BUFFERS
    ENDMSG
5$:
L10011:
    TRAP    C#MSG
    .SBTTL  FIFEXP - PRINT FIFO EXP/RCV DATA
;+
;PRINT ROUTINE TO PRINT FIFO EXP/RCV DATA

```

F5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 FIFEXP - PRINT FIFO EXP/RECV DATA

SEQ 057

```

1519
1520
1521
1522
1523
1524
1525
1526
1527 012170
      012170
1528 012170
      012170 010146
      012172 012746 012242
      012176 012746 000002
      012202 010600
      012204 104415
      012206 062706 000006
1529 012212
      012212 012746 012311
      012216 012746 000001
      012222 010600
      012224 104415
      012226 062706 000004
1530 012232 010100
1531 012234 004737 015122
1532 012240
      012240
      012240 104423
1533 012242 045 116 045 FIF1MSG:
1534 012311 045 116 045 FIF2MSG:
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549 012350
      012350
1550 012350 012701 012412
1551 012354 012100
1552 012356 001410
1553 012360
      012360 010046
      012362 012746 000001
      012366 010600
      012370 104415
      012372 062706 000004
1554 012376 000766
1555 012400 012700 000012

;
; R1 - BYTE COUNT
;
; IMPLICIT INPUTS:
;
; EXPMSG - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
; RECMSG - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
;
; BGNMSG FIFEXP
FIFEXP:
  PRINTX 0FIF1MSG,R1 ;PRINT BYTES TRANSFERRED
  MOV R1,-(SP)
  MOV 0FIF1MSG,-(SP)
  MOV 02,-(SP)
  MOV SP,R0
  TRAP C:PNTX
  ADD 06,SP
  PRINTX 0FIF2MSG ;PRINT HEADER MSG
  MOV 0FIF2MSG,-(SP)
  MOV 01,-(SP)
  MOV SP,R0
  TRAP C:PNTX
  ADD 04,SP
  MOV R1,R0 ;GET BYTE COUNT
  JSR PC,PRBYTEXP ;PRINT FIFO BYTES IN ERROR
  ENDMSG
L10012:
  TRAP C:MSG
  .ASCIZ '***A NUMBER OF BYTES TRANSFERRED = #D2'
  .ASCIZ '***A FIFO DATA BYTES IN ERROR:'
  .EVEN
  .SBTTL MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
;
; PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
;
; IMPLICIT INPUTS:
;
; EXPMSG - EXPECTED MESSAGE BUFFER
; RECMSG - RECEIVED MESSAGE BUFFER
; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
;
; BGNMSG MSGSTAT
MSGSTAT:
  MOV 0STATCOD,R1 ;ASCII ADDRESS TABLE
  MOV (R1)+,R0 ;DONE ALL MSG LINES?
  BEQ 20$ ;BR IF YES
  PRINTX R0 ;PRINT STATUS BIT NAMES
  MOV R0,-(SP)
  MOV 01,-(SP)
  MOV SP,R0
  TRAP C:PNTX
  ADD 04,SP
  BR 10$ ;DO ANOTHER MSG LINE
  20$: MOV 010,,R0 ;NUMBER OF WORDS IN A READ STATUS BUFFER

```

G5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS

SEQ 058

```

1556 012404 004737 014552      JSR      PC,PRMSGEXP      ;PRINT EXPD/RECV MESSAGE BUFFERS
1557 012410      ENDMSG
      012410      L10013:
      012410 104423      TRAP      C$MSG
1558
1559 012412 012430 012472 012563 STATCOD:      .WORD      1$,2$,3$,4$,5$,6$,0
1560 012430      045      116      045 1$:ASCIZ 'N$A Tape Bus Signals in Word #8:'
1561 012472      045      116      045 2$:ASCIZ 'N$A      PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>'
1562 012563      045      116      045 3$:ASCIZ 'N$A      IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>'
1563 012654      045      116      045 4$:ASCIZ 'N$A      IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>'
1564 012745      045      116      045 5$:ASCIZ 'N$A Tape Bus Signals in Word #9:'
1565 013007      045      116      045 6$:ASCIZ 'N$A      DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>'
1566      .EVEN
1567
1568      .SBTTL MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
1569
1570
1571      ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
1572
1573      ;IMPLICIT INPUTS:
1574
1575      ;
1576      ; EXPMSG - EXPECTED MESSAGE BUFFER
1577      ; RECMSG - RECEIVED MESSAGE BUFFER
1578      ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1579      ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1580
1580 013064      BGNMSG MSGLOOP
      013064
1581 013064 012701 013126      MSGLOOP:
1582 013070 012100      MOV      @LOOPCOD,R1      ;ASCII ADDRESS TABLE
1583 013072 001410      10$:      MOV      (R1)+,R0      ;DONE ALL MSG LINES?
1584 013074      BEQ      20$      ;BR IF YES
      013074 010046      PRINTX   R0      ;PRINT STATUS BIT NAMES
      013076 012746 000001      MOV      R0,-(SP)
      013102 010600      MOV      @1,-(SP)
      013104 104415      MOV      SP,R0
      013106 062706 000004      TRAP      C$PNTX
1585 013112 000766      ADD      @4,SP
1586 013114 012700 00001?      BR      10$      ;DO ANOTHER MSG LINE
1587 013120 004737 014552      20$:      MOV      @10,R0      ;NUMBER OF WORDS IN A READ STATUS BUFFER
1588 013124      JSR      PC,PRMSGEXP      ;PRINT EXPD/RECV MESSAGE BUFFERS
      013124      ENDMSG
      013124 104423      L10014:
      013124      TRAP      C$MSG
1589
1590 013126 013146 013221 013320 LOOPCOD:      .WORD      1$,2$,3$,4$,5$,6$,7$,0
1591 013146      045      116      045 1$:ASCIZ 'N$A Tape Bus Loopback Signals in Word #8:'
1592 013221      045      116      045 2$:ASCIZ 'N$A      PARERR<15> IRESV2<14> IRESV1<13>'
1593 013320      045      116      045 3$:ASCIZ 'N$A IHISP->IEOT<12> IHRT->IIDENT<11> IREV ->ICER <10>'
1594 013417      045      116      045 4$:ASCIZ 'N$A IWMF ->IFMK<9> IICIT->IHER <8> IFAD ->ISPEED<7>'
1595 013516      045      116      045 5$:ASCIZ 'N$A ITADO->IRDY<6> ITAD1->IONL <5> IERASE->ILDP <4>'
1596 013615      045      116      045 6$:ASCIZ 'N$A IREW ->IDBY<3> IRWU ->IRWD <2> IFEN ->IFBY <0>'
1597 013714      045      116      045 7$:ASCIZ 'N$A IGO ->IFPT<0>'
1598      .EVEN
1599      .SBTTL MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
1600
1601
1602      ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV

```

H5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 MSGSUB PRINT WRITE SUBSYSTEM MESSAGE BUFFER

SEQ 059

```

1603
1604
1605
1606
1607
1608
1609
1610
1611
1612 013742
      013742
1613 013742 012700 000012
1614 013746 004737 014552
1615 013752
      013752
      013752 104423
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629 013754
      013754
1630 013754 004737 010160
1631 013760 013701 002232
1632 013764 013702 002234
1633 013770 004737 007742
1634 013774
      013774
      013774 104423
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653

;
;
;IMPLICIT INPUTS:
;
;   EXPMSG - EXPECTED MESSAGE BUFFER
;   RECMSG - RECEIVED MESSAGE BUFFER
;   RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
;   RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
;
;   BGNMSG MSGSUB
MSGSUB:
  MOV     #10,,R0      ;SIZE OF WRITE SUBSYSTEM BUFFER
  JSR     PC,PRMSGEXP   ;PRINT EXPD/RCV MESSAGE BUFFERS
  ENDMSG
L10015:
  TRAP    C:MSG
      .SBTTL MEMADD - PRINT MEMORY ADDRESS DATA ERROR
;
;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
;
;IMPLICIT INPUTS:
;
;   ERRHI - MEMORY ERROR HIGH ORDER ADDRESS
;   ERRLO - MEMORY ERROR LOW ORDER ADDRESS
;   EXP - EXPECTED DATA
;   RECV - RECEIVED DATA
;
;   BGNMSG MEMADD
MEMADD:
  JSR     PC,PRIADD     ;PRINT MEMORY ADDRESS IN ERROR
  MOV     EXPD,R1       ;GET EXPD DATA
  MOV     RECV,R2       ;GET RECEIVED DATA
  JSR     PC,PRIXOR     ;PRINT EXPD/RCV
  ENDMSG
L10016:
  TRAP    C:MSG
      .SBTTL PRAMPKT - PRINT RAM AND PACKET DATA
;
;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
;WHEN THE RAM DATA DOES NOT MATCH.
;
;INPUTS:
;
;   R4 - POINTER TO COMMAND PACKET
;
;IMPLICIT INPUTS:
;
;   RAMDATA - DATA AS READ FROM THE RAM
;   RAMSIZ - NUMBER OF BYTES IN PACKET
;           IF RAMSIZ=0 THEN DEFAULT TO 8.
;
;IMPLICIT OUTPUTS:
;
;   RAMSIZ SET TO 0

```

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
PRAMPKT - PRINT RAM AND PACKET DATA

SEQ 060

```

1654
1655
1656 013776
1657 013776
1658 014002 012701 002242
1659 014006 005002
1660 014010 122124
1661 014012 001005
1662 014014
1663 014024 000436
1664 014026 116105 177777
1665 014032 116403 177777
1666 014036
1667 014046 042703 177400
1668 014052 116137 177777 002234
1669 014060 116437 177777 002232
1670 014066
    014066 010346
    014070 013746 002232
    014074 013746 002234
    014100 010246
    014102 012746 014156
    014106 012746 000005
    014112 010600
    014114 104414
    014116 062706 000014
1671 014122 005202
1672 014124 005737 002302
1673 014130 001404
1674 014132 020237 002302
1675 014136 003724
1676 014140 000403
1677 014142 020227 000010
1678 014146 002720
1679 014150 005037 002302
1680 014154 000207
1681
1682 014156 045 116 045 RAMASC: .ASCIZ 'N-A BYTE: D2A RAM: 03A Packet: 03A XOR: 03'
1683 .EVEN
1684 .SBTTL PRMESS - PRINT CONTENTS OF MESSAGE BUFFER
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701 014242

```

```

PRAMPKT:
    SAVREG
    MOV     #RAMDATA,R1
    CLR     R2
5$:    CMPB  (R1)+,(R4)+
    BNE     7$
    FORCERROR 7$,NOTSSR
    BR      10$
7$:    MOVB  -1(R1),R5
    MOVB  -1(R4),R3
    XOR    R5,R3
    BIC    #177400,R3
    MOVB  -1(R1),RECV
    MOVB  -1(R4),EXPD
    PRINTB #RAMASC,R2,RECV,EXPD,R3
    MOV    R3,-(SP)
    MOV    EXPD,-(SP)
    MOV    RECV,-(SP)
    MOV    R2,-(SP)
    MOV    #RAMASC,-(SP)
    MOV    #5,-(SP)
    MOV    SP,R0
    TRAP   C$PNTB
    ADD    #14,SP
10$:   INC    R2
    IST    RAMSIZ
    BEQ    15$
    CMP    R2,RAMSIZ
    RLE    5$
    BR     25$
15$:   CMP    R2,#8
20$:   BLT    5$
25$:   CLR    RAMSIZ
    RTS     PC

```

```

;SAVE R1-R5 UNTIL NEXT RETURN
;DATA FROM THE RAM
;INIT BYTE NUMBER
;COMPARE EXPECTED, RECEIVED
;BR IF NO MATCH
;000
;GET RECV RAM DATA
;GET EXPD PACKET DATA
;XOR EXPD/RECV
;LOW BYTE ONLY
;GET RECEIVED RAM DATA
;GET EXPECTED RAM DATA
;UPDATE BYTE COUNT
;DEFAULT TO 8.
;BR IF YES
;DONE ALL BYTES?
;BR IF NO
;
;DONE DEFAULT NUMBER OF BYTES?
;BR IF NO
;SET DEFAULT RAMSIZ
;RETURN

```

```

;
; THIS ROUTINE PRINTS THE CONTENTS OF
; THE 7 OR 8 WORD MESSAGE BUFFER RETURNED BY THE
; TSV-05.
;
; INPUT:
;
; R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
; R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
; NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
;
; THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
;
;
;
PRMESS:

```

J5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

SEQ 061

```

1702 014242 SAVREG ;SAVE THE REGISTERS
1703 014246 010005 MOV R0,R5 ;SAVE LOW ORDER ADDRESS
1704 014250 005737 003134 TST KTENABLE ;ADDRESS ABOVE 28K?
1705 014254 001001 BNE 10$ ;BR IF YES
1706 014256 005001 CLR R1 ;SET HIGH ORDER ADDRESS TO 0
1707 014260 010103 10$: MOV R1,R3 ;SAVE HIGH ORDER ADDRESS
1708 014262 006100 ROL R0 ;SHIFT BIT15 TO C BIT
1709 014264 006101 ROL R1 ;SHIFT TO HIGH ORDER FOR PRINTOUT
1710 014266 PRINTX #PROASC,R1,R5 ;PRINT MESSAGE BUFFER ADDRESS
      014266 010546 MOV R5,-(SP)
      014270 010146 MOV R1,-(SP)
      014272 012746 014420 MOV #PROASC,-(SP)
      014276 012746 000003 MOV #3,-(SP)
      014302 010600 MOV SP,R0
      014304 104415 TRAP C$PNTX
      014306 062706 000010 ADD #10,SP
1711 014312 PRINTX #PRIASC ;PRINT HEADER FOR CONTENTS
      014312 012746 014465 MOV #PRIASC,-(SP)
      014316 012746 000001 MOV #1,-(SP)
      014322 010600 MOV SP,R0
      014324 104415 TRAP C$PNTX
      014326 062706 000004 ADD #4,SP
1712 014332 005004 CLR R4 ;NUMBER OF THE NEXT WORD
1713 014334 010501 MOV R5,R1 ;COPY LOW ORDER ADDRESS
1714 014336 010300 MOV R3,R0 ;COPY HIGH ORDER ADDRESS
1715 014340 001403 BEQ 20$ ;BR IF NOT ABOVE 28K
1716 014342 004737 017316 JSR PC,SETMAP ;SETUP PAR ADDRESS IN R0
1717 014346 010005 MOV R0,R5 ;GET PAR FORMAT ADDRESS ABOVE 28K
1718 014350 20$: PRINTX #PRASC,R4,(R5)+ ;PRINT THE CONTENTS OF MEMORY BUFFER
      014350 012546 MOV (R5)+,-(SP)
      014352 010446 MOV R4,-(SP)
      014354 012746 014523 MOV #PRASC,-(SP)
      014360 012746 000003 MOV #3,-(SP)
      014364 010600 MOV SP,R0
      014366 104415 TRAP C$PNTX
      014370 062706 000010 ADD #10,SP
1719 014374 005204 INC R4 ;NUMBER OF THE NEXT
1720 014376 020427 000007 CMP R4,#7 ;DONE ALL YET ?
1721 014402 003005 BGT 50$ ;BRANCH IF ALL DONE
1722 014404 002761 BLT 20$ ;PRINT FIRST 7 WORDS
1723 014406 032763 000200 000012 BIT #X2.EXTF,XST2(R3);EXTENDED FEATUTES ON ?
1724 014414 001355 BNE 20$ ;PRINT EXTENDED STATUS WORD
1725 014416 000207 50$: RTS PC ;RETURN
1726
1727 014420 045 116 045 PROASC: .ASCII2 '###A Message Buffer Address - #01#05'
1728 014465 045 116 045 PRIASC: .ASCII2 '###A Message Buffer Contents:'
1729 014523 045 116 045 PRASC: .ASCII2 '### Word#01#A: #0'
1730 .EVEN
1731 .SBTTL PRMSGEXP - PRINT EXPD/RECV MESSAGE BUFFERS
1732
1733 ;
1734 ;ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
1735 ;
1736 ; R0 - NUMBER OF WORDS IN BUFFER
1737 ;
1738 ;IMPLICIT INPUTS:
1739 ;

```

K5

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS

SEQ 062

```

1740      ; EXPMSG - EXPECTED MESSAGE BUFFER
1741      ; RECMG - RECEIVED MESSAGE BUFFER
1742      ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1743      ; RCVLOAD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1744      ;
1745      PRMSGEXP::
1746      SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
1747      MOV R0,R5                            ;SAVE NUMBER OF WORDS
1748      MOV RCVLOAD,R0                      ;GET RECV LOW ADDRESS
1749      MOV R0,R4                            ;COPY LOW ADDRESS
1750      MOV RCVHIADD,R1                     ;GET RECV HIGH ADDRESS
1751      ROL R0                               ;SHIFT BIT15 TO C BIT
1752      ROL R1                               ;SHIFT TO HIGH ORDER FOR PRINTOUT
1753      PRINTX @PRMSG0,R1,R4                ;PRINT MESSAGE BUFFER ADDRESS
1754      MOV R4,-(SP)
1755      MOV R1,-(SP)
1756      MOV @PRMSG0,-(SP)
1757      MOV @3,-(SP)
1758      MOV SP,R0
1759      TRAP C$PNTX
1760      ADD @10,SP
1761      PRINTX @PRMSG1                      ;PRINT HEADER FOR CONTENTS
1762      MOV @PRMSG1,-(SP)
1763      MOV @1,-(SP)
1764      MOV SP,R0
1765      TRAP C$PNTX
1766      ADD @4,SP
1767      CLR R4                                ;NUMBER OF THE CURRENT WORD
1768      MOV @EXPMSG,R1                       ;GET EXPD BUFFER ADDRESS
1769      MOV @RCMSG,R2                       ;GET RECV BUFFER ADDRESS
1770      MOV (R1),R0                         ;GET EXPD
1771      MOV (R2),R3                         ;GET RECV
1772      XOR R0,R3                           ;XOR EXPD/RCV
1773      PRINTX @PRMSG2,R4,(R1)+,(R2)+,R3
1774      MOV R3,-(SP)
1775      MOV (R2)+,-(SP)
1776      MOV (R1)+,-(SP)
1777      MOV R4,-(SP)
1778      MOV @PRMSG2,-(SP)
1779      MOV @5,-(SP)
1780      MOV SP,R0
1781      TRAP C$PNTX
1782      ADD @14,SP
1783      INC R4                                ;NUMBER OF THE NEXT
1784      CMP R4,R5                            ;DONE ALL YET?
1785      BGE 50$                             ;BR IF YES
1786      BR 20$                             ;DO ANOTHER
1787      50$: RTS PC                         ;RETURN
1788      20$:
1789      045 PRMSG0: .ASCII 'N#A Message Buffer Address - #01#05'
1790      045 PRMSG1: .ASCII 'N#A Message Buffer Contents:'
1791      045 PRMSG2: .ASCII 'N#A WORD #D2#A EXPD: #06#A RECV: #06#A XOR: #06#A'
1792      .EVEN
1793      .SBTTL PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
1794      ;
1795      ;
1796      ;ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS

```

CH SET

L5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER

SEQ 063

```

1776      ; ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
1777      ;
1778      ; RO      - NUMBER OF BYTES IN BUFFER
1779      ;
1780      ;IMPLICIT INPUTS:
1781      ;
1782      ; EXPMSG  - EXPECTED MESSAGE BUFFER
1783      ; RECMSG  - RECEIVED MESSAGE BUFFER
1784      ;
1785 015122 PRBYTEXP:
1786 015122 SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
1787 015126 010005 MOV      R0,R5                ;SAVE NUMBER OF BYTES
1788 015130 005037 002320 CLR      PRMNO          ;INIT ERROR COUNT
1789 015134 005004 CLR      R4                    ;NUMBER OF THE CURRENT BYTE
1790 015136 012701 002322 MOV      @EXPMSG,R1      ;GET EXPD BUFFER ADDRESS
1791 015142 012702 002466 MOV      @RECMSG,R2      ;GET RECV BUFFER ADDRESS
1792 015146 111100 20$: MOVB      (R1),R0          ;GET EXPD BYTE
1793 015150 042700 177400 BIC      @C<377>,R0      ;CLEAR UPPER BYTE
1794 015154 110037 015470 MOVB     R0,PRBEXP       ;SAVE FOR ERROR REPORT
1795 015160 111203 MOVB      (R2),R3              ;GET RECV BYTE
1796 015162 042703 177400 BIC      @C<377>,R3      ;CLEAR UPPER BYTE
1797 015166 110337 015472 MOVB     R3,PRBREC       ;FOR ERROR REPORT
1798 015172 XOR      R0,R3                        ;XOR EXPD/RECV
1799 015202 122122 CMPB      (R1)+,(R2)+          ;EXPD = RECV?
1800 015204 001431 BEQ      30$                    ;BR IF YES
1801 015206 005237 002320 INC      PRMNO          ;UPDATE ERROR COUNT
1802 015212 023727 002320 000010 CMP      PRMNO,#8.  ;PRINTED 8?
1803 015220 101023 BHI      30$                    ;BR IF YES
1804 015222 27$: PRINTX     @PRBMSG,R4,PRBEXP,PRBREC,R3
1805 015222 010346 MOV      R3,-(SP)
1806 015224 013746 015472 MOV      PRBREC,-(SP)
1807 015230 013746 015470 MOV      PRBEXP,-(SP)
1808 015234 010446 MOV      R4,-(SP)
1809 015236 012746 015336 MOV      @PRBMSG,-(SP)
1810 015242 012746 000005 MOV      #5,-(SP)
1811 015246 010600 MOV      SP,R0
1812 015250 104415 TRAP     C$PNTX
1813 015252 062706 000014 ADD      #14,SP
1814 015256 FORCEEXIT 50$ 50$ ;000
1815 015266 000404 BR      35$ 35$ ;000
1816 015270 30$: FORCEERROR 27$,NOTSSR ;000
1817 015270 35$: ;000
1818 015300 INC      R4                            ;NUMBER OF THE NEXT
1819 015300 005204 CMP      R4,R5                    ;DONE ALL YET?
1820 015302 020405 BGE      50$                    ;BR IF YES
1821 015304 002001 BR      20$                    ;DO ANOTHER
1822 015306 000717 50$: PRINTX     @PRBTOT,PRMNO    ;PRINT TOTAL ERROR COUNT
1823 015310 013746 002320 MOV      PRMNO,-(SP)
1824 015314 012746 015423 MOV      @PRBTOT,-(SP)
1825 015320 012746 000002 MOV      #2,-(SP)
1826 015324 010600 MOV      SP,R0
1827 015326 104415 TRAP     C$PNTX
1828 015330 062706 000006 ADD      #6,SP
1829 015334 000207 RTS      PC                      ;RETURN
1830 015336 045 116 045 PRBMSG: ,ASCIIZ 'NNA BYTE C$D2NA EXPD: #03NA RECV: #03NA XOR: #03'

```

M5

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER

SEQ 064

```

1818 015423      045      116      045 PRBTOT: .ASCIZ  'N#A NUMBER OF BYTES IN ERROR = #D2'
1819                                     .EVEN
1820 015470      000000      PRBEXP: .WORD  0          ;EXPD
1821 015472      000000      PRBREC: .WORD  0          ;RECV
1822                                     .SBTTL  EXPREC - PRINT EXPD/RECV WORD DATA
1823                                     ;+
1824                                     ;
1825                                     ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1826                                     ;
1827                                     ;INPUTS:
1828                                     ;
1829                                     ;      R1      RECEIVED DATA
1830                                     ;      R2      EXPECTED DATA
1831                                     ;
1832                                     ;-
1833
1834 015474                                     BGNMSG  EXPREC
1835 015474                                     EXPREC: ;
1836 015474      004737      007742      JSR      PC,PRIXOR          ;PRINT THE DATA
1837 015500      015500      L10017:      ENDMMSG
1838 015500      104423      TRAP      C$MSG
1839                                     .SBTTL  EXPBREC - PRINT EXPD/RECV BYTE DATA
1840                                     ;+
1841                                     ;
1842                                     ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
1843                                     ;
1844                                     ;INPUTS:
1845                                     ;
1846                                     ;      R1      RECEIVED DATA BYTE
1847                                     ;      R2      EXPECTED DATA BYTE
1848                                     ;
1849                                     ;-
1850
1851 015502                                     BGNMSG  EXPBREC
1852 015502      004737      007612      EXPBREC: ;
1853 015506      015506      L10020:      JSR      PC,PRIBXOR          ;PRINT THE DATA
1854 015506      104423      ENDMMSG
1855                                     TRAP      C$MSG
1856                                     .SBTTL  RAMERR - PRINT RAM AND PACKET DATA
1857                                     ;+
1858                                     ;
1859                                     ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
1860                                     ;
1861                                     ;INPUTS:
1862                                     ;
1863                                     ;      R4      POINTER TO COMMAND PACKET
1864                                     ;
1865                                     ;IMPLICIT INPUTS:
1866                                     ;
1867                                     ;      RAMDATA      DATA AS READ FROM THE RAM
1868                                     ;      RAMSIZ      NUMBER OF BYTES IN PACKET
1869                                     ;                  IF RAMSIZ=0 THEN DEFAULT TO 8.

```

N5

TSVS - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
 RAMERR - PRINT RAM AND PACKET DATA

SEQ 065

```

1869
1870      ;IMPLICIT OUTPUTS:
1871      ;
1872      ;      RAMSIZ  SET TO 0
1873      ;
1874      ;
1875      015510      BGNMSG  RAMERR
1876      015510      RAMERR:: JSR      PC,PRAMPKT      ;PRINT RAM/PACKET DATA
1877      015510      004737  013776      ENDMMSG
1878      015514      L10021: TRAP      C$MSG
1879      015514      104423      .SBTTL  RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
1880      ;
1881      ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
1882      ;
1883      ;INPUTS:
1884      ;
1885      ;      R4      POINTER TO COMMAND PACKET
1886      ;
1887      ;IMPLICIT INPUTS:
1888      ;
1889      ;      RAMDATA      DATA AS READ FROM THE RAM
1890      ;      RAMSIZ      NUMBER OF BYTES IN PACKET
1891      ;                  IF RAMSIZ=0 THEN DEFAULT TO 8.
1892      ;      ERRHI      HIGH ORDER TEST ADDRESS
1893      ;      ERRLO      LOW ORDER TEST ADDRESS
1894      ;
1895      ;IMPLICIT OUTPUTS:
1896      ;
1897      ;      RAMSIZ  SET TO 0
1898      ;
1899      ;
1900      ;
1901      015516      BGNMSG  RAMTADD
1902      015516      RAMTADD:: JSR      PC,PRITADD      ;PRINT TEST ADDRESS
1903      015516      004737  010274      JSR      PC,PRAMPKT      ;PRINT RAM/PACKET DATA
1904      015522      004737  013776      ENDMMSG
1905      015526      L10022: TRAP      C$MSG
1906      015526      104423      .SBTTL  RAMEXP - PRINT RAM EXPD/RECV DATA
1907      ;
1908      ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1909      ;
1910      ;INPUTS:
1911      ;
1912      ;      R1      RECEIVED DATA
1913      ;      R2      EXPECTED DATA
1914      ;      R4      CONTROLLER RAM ADDRESS
1915      ;
1916      ;
1917      ;
1918      015530      BGNMSG  RAMEXP
1919      015530      RAMEXP::

```

B6

TSV3 - GLOBAL AREAS MACRO M1115 06-FEB-84 17:14  
RAMEXP - PRINT RAM EXPD/RECV DATA

SEQ 066

```

1919 015530 042701 177400      BIC    #C<377>,R1      ;SAVE EXPD RAM DATA BYTE
1920 015534 042702 177400      BIC    #C<377>,R2      ;SAVE EXPD RAM DATA BYTE
1921 015540 004737 010066      JSR    PC,PRIRAM    ;PRINT THE RAM ADDRESS
1922 015544 004737 007742      JSR    PC,PRIXOR    ;PRINT THE DATA
1923 015550      ENDMMSG
      015550
      015550 104423      L10023: TRAP    CMSG
1924
1925      .SBTTL  TIMEXP - PRINT TIMER A,B AND EXP/REC
1926
1927      ;
1928      ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1929      ;AND TIMER A,B HEADER MESSAGE
1930      ;
1931      ;INPUTS:
1932      ;
1933      ;      R1      RECEIVED DATA
1934      ;      R2      EXPECTED DATA
1935      ;
1936
1937 015552      BGNMSG  TIMEXP
      015552
1938 015552      TIMEXP: PRINTX  #TIMSGO      ;PRINT HEADER
      015552 012746 015600      MOV    #TIMSGO,-(SP)
      015556 012746 000001      MOV    #1,-(SP)
      015562 010600      MOV    SP,R0
      015564 104415      TRAP    CIPNTX
      015566 062706 000004      ADD    #4,SP
1939 015572 004737 007742      JSR    PC,PRIXOR    ;PRINT THE DATA
1940 015576      ENDMMSG
      015576
      015576 104423      L10024: TRAP    CMSG
1941
1942 015600      045      116      045 TIMSGO: .ASCIZ  'TIMER A STATUS IS IN BIT 3'
1943      .EVEN      ;TIMER B STATUS IS IN BIT 2'
1944      .SBTTL  BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS
1945
1946      ;
1947      ;PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
1948      ;
1949      ;INPUTS:
1950      ;
1951      ;      R1      CONTENTS OF TSSR
1952      ;      R2      DATA WRITTEN (8 BITS)
1953      ;
1954      ;
1955
1956
1957 015700      BGNMSG  BADSSR
      015700
1958 015700 010246      BADSSR: MOV    R2,-(SP)      ;SAVE DATA TRANSFERRED
1959 015702 042702 177400      BIC    #177400,R2      ;GET JUST ONE BYTE
1960 015706      PRINTB  #XFERASC,R2
      015706 010246      MOV    R2,-(SP)
      015710 012746 015740      MOV    #XFERASC,-(SP)
      015714 012746 000002      MOV    #2,-(SP)
      015720 010600      MOV    SP,R0

```

C6

TSV3 GLOBAL AREAS MACRO M1113 06 FEB 84 17:14  
BADSSR PRINT TSSR ERRORS ON DATA TRANSFERS

SEQ 067

|      |        |        |        |        |          |        |                               |                            |
|------|--------|--------|--------|--------|----------|--------|-------------------------------|----------------------------|
| 1961 | 015722 | 104414 |        |        |          | TRAP   | C#PNTB                        |                            |
| 1962 | 015724 | 062706 | 000006 |        |          | ADD    | #6,SP                         |                            |
| 1963 | 015730 | 012602 |        |        |          | MOV    | (SP),R2                       | ;RESTORE R2                |
| 1964 | 015732 | 004737 | 006024 |        |          | JSR    | PC,PRITSSR                    | ;DECODE TSSR CONTENTS      |
| 1965 | 015736 |        |        |        |          | ENDMSG |                               |                            |
| 1966 | 015736 |        |        |        | L10025:  |        |                               |                            |
| 1967 | 015736 | 104423 |        |        |          | TRAP   | C#MSG                         |                            |
| 1968 | 015740 | 045    | 116    | 045    | XFERASC: | .ASCIZ | '#NMA Data Transferred = #03' |                            |
| 1969 |        |        |        |        |          | .SBTTL | GLOBAL SUBROUTINES SECTION    |                            |
| 1970 |        |        |        |        |          |        |                               |                            |
| 1971 |        |        |        |        |          |        |                               |                            |
| 1972 |        |        |        |        |          |        |                               |                            |
| 1973 |        |        |        |        |          |        |                               |                            |
| 1974 |        |        |        |        |          |        |                               |                            |
| 1975 |        |        |        |        |          |        |                               |                            |
| 1976 |        |        |        |        |          |        |                               |                            |
| 1977 |        |        |        |        |          |        |                               |                            |
| 1978 |        |        |        |        |          |        |                               |                            |
| 1979 |        |        |        |        |          |        |                               |                            |
| 1980 |        |        |        |        |          |        |                               |                            |
| 1981 |        |        |        |        |          |        |                               |                            |
| 1982 |        |        |        |        |          |        |                               |                            |
| 1983 |        |        |        |        |          |        |                               |                            |
| 1984 |        |        |        |        |          |        |                               |                            |
| 1985 |        |        |        |        |          |        |                               |                            |
| 1986 |        |        |        |        |          |        |                               |                            |
| 1987 |        |        |        |        |          |        |                               |                            |
| 1988 |        |        |        |        |          |        |                               |                            |
| 1989 |        |        |        |        |          |        |                               |                            |
| 1990 |        |        |        |        |          |        |                               |                            |
| 1991 |        |        |        |        |          |        |                               |                            |
| 1992 |        |        |        |        |          |        |                               |                            |
| 1993 |        |        |        |        |          |        |                               |                            |
| 1994 |        |        |        |        |          |        |                               |                            |
| 1995 |        |        |        |        |          |        |                               |                            |
| 1996 |        |        |        |        |          |        |                               |                            |
| 1997 |        |        |        |        |          |        |                               |                            |
| 1998 |        |        |        |        |          |        |                               |                            |
| 1999 | 015774 |        |        |        |          |        |                               |                            |
| 2000 | 015774 |        |        |        |          |        |                               |                            |
| 2001 | 016000 | 012765 | 000000 | 000002 |          | SAVREG |                               | ;SAVE THE REGISTERS        |
| 2002 | 016006 | 004737 | 016250 |        |          | MOV    | #0,TSSR(R5)                   | ;DO THE INIT.              |
| 2003 | 016012 | 016500 | 000002 |        |          | JSR    | PC,WAITF                      | ;WAIT FOR SSR              |
| 2004 | 016016 | 010004 |        |        |          | MOV    | TSSR(R5),R0                   | ;GET THE TSSR REGISTER     |
| 2005 | 016020 | 042704 | 176277 |        |          | MOV    | R0,R4                         | ;TSSR CONTENTS             |
| 2006 | 016024 | 052704 | 002200 |        |          | BIC    | #1C<HIADDR!OFL>,R4            |                            |
| 2007 | 016030 | 020400 |        |        |          | BIS    | #SSR!NBA,R4                   | ;R4 HAS EXPECTED CONTENTS  |
| 2008 | 016032 | 001402 |        |        |          | CMP    | R4,R0                         | ;ONLY EXPECTED BITS SET ?  |
| 2009 | 016034 | 000241 |        |        |          | BEQ    | 51                            | ;BRANCH IF OKAY            |
| 2010 | 016036 | 000401 |        |        |          | CLC    |                               | ;CLEAR THE CARRY FOR ERROR |
| 2011 | 016040 | 000261 |        |        |          | BR     | 101                           | ;GO TO EXIT                |
| 2012 | 016042 | 000207 |        |        |          | SEC    |                               | ;SET THE CARRY BIT         |
| 2013 |        |        |        |        |          | RTS    | PC                            | ;RETURN TO CALLER          |
|      |        |        |        |        |          | .SBTTL | CHKAMB                        | - CHECK TSSR FOR AMBIGUITY |

D6

 ISV3 GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 CHKAMB - CHECK TSSR FOR AMBIGUITY

SEQ 068

```

2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033 016044
2034 016044
2035 016050 010004
2036 016052 032700 100000
2037 016056 001004
2038 016060 032700 174077
2039 016064 001023
2040 016066 000424
2041 016070 032700 000200
2042 016074 001011
2043 016076 032700 000040
2044 016102 001414
2045 016104 042704 177761
2046 016110 020427 000016
2047 016114 001007
2048 016116 000410
2049 016120 032700 000040
2050 016124 001405
2051 016126 032700 000006
2052 016132 001002
2053 016134 000241
2054 016136 000401
2055 016140 000261
2056 016142 000207
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066 000200
2067 000001
2068
2069
2070 016144 000

;
; THIS ROUTINE TESTS THE CONTENTS OF THE TSSR REGISTER
; FOR AMBIGUITY
;
; INPUT:
;
;      RO      CONTENTS OF TSSR
;
; OUTPUT:
;
;      RO      CONTENTS OF TSSR
;
;      CARRY   SET - NO AMBIGUITY
;              CLR - AMBIGUOUS CONTENTS
;
;
CHKAMB:
    SAVREG                ;SAVE THE GENERAL REGISTERS
    MOV      RO,R4        ;CONTENTS OF TSSR
    BIT      #SC,R0       ;IS BIT 15 SET ?
    BNE      5$           ;BRANCH IF YES
    BIT      #C<NBA!OFL!SSR!HIADDR>,R0 ;ANY OTHER BITS SET ?
    BNE      40$          ;MUST BE AN ERROR
    BR       45$          ;RETURN WITH SUCCESS
5$:   BIT      #SSR,R0    ;IS READY BIT SET ?
    BNE      10$         ;BRANCH IF READY BIT IS SET.
    BIT      #BIT5,R0    ;IS FATAL ERROR BIT SET ?
    BEQ      40$         ;ERROR IF NOT
    BIC      #CTERCLS,R4 ;CLEAR ALL BUT TERMINATION CODE
    CMP      R4,#16      ;ALL THREE BITS MUST BE SET
    BNE      40$         ;ERROR IF NOT SET
    BR       45$         ;OK IF ALL ARE SET
10$:  BIT      #BIT5,R0    ;IS FATAL ERROR BIT SET ?
    BEQ      45$         ;ERROR IF BIT IS SET WITH SSR
    BIT      #BIT2!BIT1,R0 ;IS THIS A FUNCTION REJECT
    BNE      45$         ;BR, IF TSSR IS OK
40$:  CLC                ;AMBIGUOUS CONTENTS
    BR       50$
45$:  SEC                ;SHOW SUCCESS - NO AMBIGUITY
50$:  RTS      PC         ;RETURN TO CALLER
    .SBITL ENAINT,OSBINT - ENABLE/DISABLE INTERRUPTS

;
; DEFAULT DISPLAY INTERRUPT HANDLERS.
; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
; OTHERWISE, SAVE DPU REGISTERS AND DISMISS.
;
;
; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
;
;      IOKCKIN=BIT7      ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
;      IOKSTP=BIT0       ; EXPECT "STOP" INTERRUPT.
;
; INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
INTMASK: .BYTE 0

```

E6

TSV3 - GLOBAL AREAS      MACRO M1113 06-FEB-84 17:14  
ENAINIT,DSBINT - ENABLE/DISABLE INTERRUPTS

SEQ 069

```

2071                                ; INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
2072 016145      000                INTFLAG: .BYTE 0
2073
2074                                ; SAVED INTERRUPT VECTOR:
2075 016146      000000             INTVEC: .WORD 0
2076                                ; SAVE CPU PC
2077 016150      000000             INTCPC: .WORD 0
2078
2079                                ; SUBROUTINE TO ENABLE INTERRUPTS:
2080 016152      010046             ENAINIT: MOV R0, -(SP) ; SAVE R0
2081 016154      013700 002210      MOV IVEC, R0 ; GET POINTER TO VECTORS
2082 016160      012720 016216      MOV @INTR, (R0), ; SET IP INTERRUPT VECTOR
2083 016164      012720 000340      MOV @PRI07, (R0),
2084 016170      012600              MOV (SP)+, R0 ; RESTORE R0
2085 016172      011646              MOV (SP), -(SP)
2086 016174      012766 000000 000002 MOV @0,2(SP) ; SET CPU TO LEVEL 0
2087 016202      000002              RTI
2088
2089                                ; SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
2090 016204      011646             DSBINT: MOV (SP), -(SP)
2091 016206      012766 000340 000002 MOV @PRI07,2(SP)
2092 016214      000002              RTI
2093                                .SBTTL INTR - INTERRUPT HANDLERS
2094
2095 016216              BGNISRV INTR ; DEFINE INTERRUPT ENTRY
2096 016216              INTR::
2097 016216      012737 000001 002224 MOV @1,INTIECV ; SET FLAG TO SHOW INTERRUPT RECEIVED
2098 016224      105037 016145      CLRB INTFLAG ; CLEAR FLAG TO SAY WE GOT INTERRUPT
2099 016230      132737 000001 016144 BITB @IOKSTP,INTMASK ; EXPECTING STOP INTERRUPT?
2100 016236      001003              BNE 1$ ; BR IF YES
2101 016240      152737 000001 016145 BISB @IOKSTP,INTFLAG ; NO, SET THE ERROR FLAG.
2102
2103                                ; SAVE REGISTERS, MSG BUFFER, ETC.
2104 016246              1$:
2105 016246              ENDSRV
2106 016246              L10026:
2107 016246      000002              RTI
2108                                .SBTTL WAITF - WAIT FOR SUBSYSTEM READY
2109
2110                                ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
2111                                ;
2112                                ; INPUTS:
2113                                ;
2114                                ; R5 ADDRESS OF FIRST DEVICE REGISTER
2115                                ;
2116                                ; OUTPUTS:
2117                                ;
2118                                ; R0 CONTENTS OF LAST TSSR READ
2119                                ; CARRY SET - READY BIT SET
2120                                ; CLR - TIMEOUT WAITING FOR READY
2121 016250      000401             WAITF:: BR 1$ ; NOP WHEN SUPER FIXED
2122 016252      104422             BREAK ; DO A SUPVSR BREAK FIRST.
2123 016252      012746 011000      TRAP C$BRK
2124 016254      016500 000002      MOV @11000, -(SP) ; 25-APRIL-83 REV B - 1100 MSEC TIMER
2125 016260      105700              MOV TSSR(R5), R0 ; READ THE TSSR REGISTER
2126 016264              TSTB R0 ; TEST FOR READY BIT SET

```

F6

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
WAITF    - WAIT FOR SUBSYSTEM READY

SEQ 070

```

2124
2125 016266 100420          BMI      3$          ; EXIT ON STOP FLAG.
2126 016270          DELAY     1          ; WAIT 100 USEC
      016270 012727 000001    MOV      01,(PC)+
      016274 000000          .WORD     0
      016276 013727 002116    MOV      L$DLY,(PC)+
      016302 000000          .WORD     0
      016304 005367 177772    DEC       -6(PC)
      016310 001375          BNE      .-4
      016312 005367 177756    DEC      -22(PC)
      016316 001367          BNE      .-20
2127 016320 005316          DEC      (SP)          ; REDUCE DELAY COUNT
2128 016322 001356          BNE      2$          ; RETRY UNTIL TIMER EXPIRES
2129 016324 000241          CLC              ; C = 0, CONTROLLER STILL RUNNING...
2130 016326 000401          BR       4$          ; ...OR HUNG-UP AFTER 300 MSEC.
2131 016330 000261          3$: SEC              ; C = 1, CONTROLLER IS STOPPED.
2132 016332 005326          4$: DEC      (SP)+      ; RESTORE STACK WITHOUT CHANGING CARRY BIT
2133 016334 000207          RTS       PC
      .SBTTL  CHKTSSR - CHECK TSSR FOR READY
2134
2135
2136
2137
2138 ; THIS ROUTINE WAITS FOR READY IN THE TSSR
2139 ; AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
2140
2141 ; INPUT:
2142
2143 ; R5      ADDRESS OF CSR REGISTERS
2144
2145 ; OUTPUT:
2146
2147 ; R0      CONTENTS OF TSSR
2148 ; CARRY   SET - OKAY
2149 ;         CLR - NOT READY AMBIGUOUS, OR SC SET
2150
2151
2152
2153 CHKTSSR:
2154 016336 004737 016250    JSR      PC,WAITF      ; WAIT FOR READY
2155 016342 103014          BCC      20$          ; BRANCH IF TIME OUT
2156 016344 004737 016044    JSR      PC,CHKAMB    ; TSSR AMBIGUOUS?
2157 016350 103006          BCC      10$          ; BR IF YES
2158 016352 032700 100000    BIT      0$C,R0      ; SPECIAL CONDITION SET?
2159 016356 001405          BEQ      15$          ; BR IF NO
2160 016360 032700 074000    BIT      0$<SCE!BIE!RMR!NXM>,R0 ; ANY ERROR BITS SET?
2161 016364 001402          BEQ      15$          ; BR IF NO
2162 016366 000241          10$: CLC              ; SET FAILURE
2163 016370 000401          BR       20$          ;
2164 016372 000261          15$: SEC              ; SET SUCCESS
2165 016374 000207          20$: RTS       PC      ; RETURN TO CALLER
2166          .SBTTL  NXNM      - CHECK FOR NONEXISTENT MEMORY
2167
2168 ; *
2169 ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
2170 ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
2171 ; "C" = 0, ALL ADDRESSES OK.
2172 ;
;CALL:  MOV  ADR1,R1

```

G6

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
XNXM - CHECK FOR NONEXISTENT MEMORY

SEQ 071

```

2173      ;      MOV ADR2,R2
2174      ;      JSR PC,NXM
2175      ;      RETURN      ;TEST "C" AND PROCEED.
2176      ;
2177 016376 012737 016430 000004 XNXM: MOV    #2#,R04      ; SET BUSERR VECTOR.
2178 016404 012737 000200 000006      MOV    #PRI04,R06
2179 016412 005003      CLR    R3      ;FLAG.
2180 016414 005711 1$: TST    (R1)      ;TEST THE ADDRESS(ES).
2181      ;IF ANY TRAP, CONTINUE AT 2$.
2182 016416 020102      CMP    R1,R2      ;OTHERWISE, CONTINUE HERE.
2183 016420 001407      BEQ    3$      ;BR IF FINISHED (NO NEXM'S).
2184 016422 062701 000002      ADD    #2,R1      ;SET NEXT ADDRESS...
2185 016426 000772      BR     1$      ;...AND CONTINUE.
2186      ;
2187 016430 005103 2$: COM    R3      ;GOT ONE, SET FLAG...
2188 016432 012716 016440      MOV    #3#,(SP)
2189 016436 000002      RTI     ;...AND DISMISS INTERRUPT...
2190 016440 012700 000004 3$: CLRVEC #4      ;...AND GIVE BACK THE VECTOR.
2191 016444 104436      MOV    #4,R0
2192 016446 005703      TRAP   C#CVEC
2193 016450 001401      TST    R3      ;DID WE CATCH ONE ??
2194 016452 000261      BEQ    .+4      ;NO, "C" = 0, SKIP NEXT.
2195 016454 000207      SEC     ;YES, "C" = 1, (R1) = NEXM ADDR.
2196      RTS    PC
2197
2198      .SBTTL TSTLOOP - CHECK ITERATION COUNT
2199      ;
2200      ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
2201      ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
2202      ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
2203      ;
2204      ; CALL: LOOPTO ARG
2205      ;
2206 016456 005737 002170 TSTLOOP: TST    NOITS      ; ITERATIONS INHIBITED?
2207 016462 001006      BNE    1$      ; YES.
2208 016464 005737 002204      TST    QVP      ; NO.
2209 016470 100403      BMI    1$      ;LOOPS DISALLOWED IN QUICK PASS.
2210 016472 005337 002216      DEC    LOOPCNT      ; BUMP LOOP COUNTER.
2211 016476 001002      BNE    2$
2212 016500 000241 1$: CLC     ;LOOP DISALLOWED, OR DONE.
2213 016502 000401      BR     3$
2214 016504 000261 2$: SEC     ;LOOP ENABLED.
2215 016506 000207 3$: RTS    PC
2216
2217      .SBTTL TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS
2218      ;
2219      ; PRINT THE NUMBER AND NAME OF EACH TEST AS WE GO ALONG.
2220      ; INCREMENT "TESTK" TO INDICATE THE NUMBER OF TESTS
2221      ; IN THE CURRENT RUN SEQUENCE.
2222      ; CLEAR THE ERROR COUNTER AND SIGNATURE EXTENSION FLAGS.
2223      ;
2224      ; INPUT:
2225      ;
2226      ; R0      POINTER TO TEST ID ASCIZ STRING
2227      ;

```

H6

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS

SEQ 072

```

2228      ;OUTPUT;
2229      ;
2230      ;      R5      ADDRESS OF FIRST DEVICE REGISTER
2231      ;
2232      ;IMPLICIT OUTPUTS:
2233      ;
2234      ;      TSTCNT  UPDATED TO COUNT TESTS PERFORMED SINCE START OR RESTART
2235      ;
2236      ;SIDE EFFECTS:
2237      ;
2238      ;      INTERRUPT LEVEL IS RASIED TO LEVEL OF
2239      ;      THE DEVICE UNDER TEST
2240      ;
2241      ;-
2242
2243      TSTSETUP::
2244      016510      010046      MOV      RO,-(SP)      ;SAVE THE TEST ID MESSAGE
2245      016512      005037      003154      CLR      SIFLAG      ; CLEAR "SOFT INIT" FLAG
2246      016516      005037      016756      CLR      ERRK      ; CLEAR LOCAL ERROR COUNTER.
2247      016522      005037      005772      CLR      EXTA      ; CLEAR ERROR EXTENSION FLAG.
2248      016526      105037      016144      CLRB     INTMASK     ; CLEAR INTERRUPT MASK (CHECK ERROR)
2249      016532      013700      002202      MOV      UNITN,RO     ; GET THE UNIT NUMBER,
2250      016536      006300      ASL      RO      ; ... AND MAKE IT A WORD OFFSET.
2251      016540      005737      003114      TST      NODEV      ; DID STARTUP FIND THE DEVICE?
2252      016544      001430      BEQ      4$      ; BR IF YES
2253      016546      100010      BPL      3$      ; BR IF NOT IDLE
2254      016550      052760      160000      003176      BIS      0160000,ERTABL(RO) ; FLAG ERROR IN THE ERROR TABLE
2255      016556      104455      ERDF      1,NXR,NXRERR      ; NO DEVICE HERE -- PRINT IT
2256      016566      000407      TRAP      C$ERDF
2257      016570      052760      160001      003176      3$:    .WORD      1
2258      016576      104455      .WORD      NXR
2259      016606      012737      177777      003112      2$:    .WORD      NXRERR
2260      016614      013700      002202      BR      2$
2261      016622      104444      BIS      0160001,ERTABL(RO) ; FLAG ERROR IN THE ERROR TABLE
2262      016624      000423      ERDF      2,NOINIT      ; DEVICE NOT IDLE
2263      016626      104421      TRAP      C$ERDF
2264      016630      032700      001000      .WORD      2
2265      016634      001412      .WORD      NOINIT
2266      016636      011600      .WORD      0
2267      016640      010046      MOV      0-1,DUFLG      ; DROP THE UNIT
2268      016642      012746      016704      DODU      UNITN
2269      016646      012746      000002      MOV      UNITN,RO
2270      016652      010600      TRAP      C$DODU
2271      016654      010600      DOCLN      ; ABORT THE PASS
2272      016656      010600      TRAP      C$DOCLN
2273      016658      010600      BR      5$
2274      016660      010600      4$:    RFLAGS      RO      ; GET THE OPERATOR FLAGS.
2275      016662      010600      TRAP      C$RFLA
2276      016664      010600      BIT      0PNT,RO      ; PRINT THE TEST NUMBERS?
2277      016666      010600      BEQ      1$      ; BR IF NO
2278      016668      010600      MOV      (SP),RO      ;GET THE ID MESSAGE
2279      016670      010600      PRINTF      0TNAM,RO      ;DISPLAY THE TEST ID
2280      016672      010600      MOV      RO,-(SP)
2281      016674      010600      MOV      0TNAM,-(SP)
2282      016676      010600      MOV      02,-(SP)
2283      016678      010600      MOV      SP,RO

```

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS

SEQ 073

```

016654 104417
016656 062706 000006
2269 016662 005237 002214
2270 016666 013700 002212
016672 104441
2271 016674 005726
2272 016676 013705 002206
2273 016702 000207
2274 016704 045 123 045 TNAM:
2275
2276
2277
2278
2279
2280
2281 016720
016720 104421
2282 016722 030027 020000
2283 016726 001412
2284 016730
016730 013746 016756
016734 012746 016760
016740 012746 000002
016744 010600
016746 104417
016750 062706 000006
2285 016754 000207
2286
2287 016756 000000
2288 016760 045 101 040
2289 016777 105 122 122
2290
2291
2292
2293
2294
2295
2296 017044 005237 016756
2297 017050 010046
2298 017052 013700 002202
2299 017056 006300
2300 017060 062700 003176
2301 017064 005210
2302 017066 032710 007777
2303 017072 001001
2304 017074 005310
2305 017076 012600
2306 017100 000207
2307
2308 017102 010046
2309 017104 013700 002202
2310 017110 006300
2311 017112 016000 003176
2312 017116 042700 170000
2313 017122 020037 002174
2314 017126 103004

TRAP C$PNTF
ADD #6,SP
1$: INC TSTCNT ; BUMP TEST COUNTER.
SETPRI IPRI ; PRIORITY THAT OF DEVICE
MOV IPRI,RO
TRAP C$SPRI
5$: TST (SP); ; FIX UP THE STACK
MOV CSRADDR,R5 ; ADDRESS OF TSV REGISTERS ON UNIBUS
RTS PC
045 TNAM: .ASCIZ 'TSV3 Test'
.EVEN
.SBTTL TSTEND - PRINT ERRORS RECEIVED
;
; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
;
TSTEND: RFLAGS RO
TRAP C$RFLA
BIT RO,#IER
1$ BEQ 1$ ; BR IF "IER" NOT SET.
PRINTF #ESUM,ERRK ; PRINT ERROR COUNT.
MOV ERRK,-(SP)
MOV #ESUM,-(SP)
MOV #2,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #6,SP
1$: RTS PC

ERRK: 0 ; LOCAL ERROR COUNT.
040 ESUM: .ASCIZ /#A #DMA ERRORS/
122 EMAXDU: .ASCIZ /ERROR LIMIT REACHED -- DROPPING UNIT/
.EVEN
.SBTTL INCERK - INCREMENT LOCAL ERROR COUNT
;
; ROUTINES TO INCREMENT LOCAL ERROR COUNT AND CHECK FOR LIMIT:
;
INCERK: INC ERRK ; INCREMENT LOCAL ERROR COUNT
MOV RO,-(SP) ; SAVE RO
MOV UNITN,RO ; GET UNIT NUMBER
ASL RO ; ... AND MAKE IT A WORD OFFSET.
ADD #ERTABL,RO ; RO GETS ADDRESS OF ERROR TABLE ENTRY.
INC (RO) ; INCREMENT THE DEVICE ERROR COUNT
BIT #7777,(RO) ; DID WE OVERFLOW THE FIELD?
BNE 1$ ; BR IF NO.
DEC (RO) ; YES -- BACK IT UP TO 7777.
1$: MOV (SP)+,RO ; RESTORE RO
RTS PC ; RETURN TO CALLER.

CKEMAX: MOV RO,-(SP) ; SAVE RO
MOV UNITN,RO ; GET UNIT NUMBER
ASL RO ; ... AND MAKE IT A WORD OFFSET
MOV ERTABL(RO),RO ; GET ERROR TABLE ENTRY
BIC #170000,RO ; EXTRACT ERROR COUNT FIELD
CMP RO,GERRMAX ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
1$ BHS 1$ ; BR IF YES

```

J6

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
 INCERK - INCREMENT LOCAL ERROR COUNT

SEQ 074

```

2315 017130 023737 016756 002172      CMP      ERRK,LERRMAX      ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
2316 017136 103417                      BLO      2$              ; BR IF NO
2317 017140                      1$:    RFLAGS  RO              ; GET OPERATOR FLAGS
      017140 104421                      TRAP     C$RFLA
2318 017142 032700 000040              BIT      $IDU,RO          ; IS DROPPING INHIBITED?
2319 017146 001013                      BNE      2$              ; BR IF YES.
2320 017150 012737 177777 003112      MOV      #-1,DUFLG      ; NO -- DROP THE UNIT
2321 017156                      ERRDF    4,EMAXDU
      017156 104455                      TRAP     C$ERDF
      017160 000004                      .WORD    4
      017162 016777                      .WORD    EMAXDU
      017164 000000                      .WORD    0
2322 017166                      DODU     UNITN
      017166 013700 002202              MOV      UNITN,RO
      017172 104451                      TRAP     C$DODU
2323 017174                      DOCLN
      017174 104444                      TRAP     C$DCLN
2324 017176 012600                      2$:    MOV      (SP)+,RO      ; RESTORE RO
2325 017200 000207                      RTS      PC              ; RETURN TO CALLER
2326                      .SBTTL  CKDROP - CHECK IF UNIT SHOULD BE DROPPED
2327                      ;
2328                      ; CHECK IF UNIT SHOULD BE DROPPED
2329                      ;
2330 017202 010046                      CKDROP: MOV      RO,-(SP)
2331 017204                      FORCERROR 1$,NOTSSR
2332 017214                      RFLAGS  RO
      017214 104421                      TRAP     C$RFLA
2333 017216 032700 000040              BIT      $IDU,RO
2334 017222 001010                      BNE      1$
2335 017224 011600                      MOV      (SP),RO
2336 017226 012737 177777 003112      MOV      #-1,DUFLG
2337 017234                      DODU     UNITN
      017234 013700 002202              MOV      UNITN,RO
      017240 104451                      TRAP     C$DODU
2338 017242                      DOCLN
      017242 104444                      TRAP     C$DCLN
      ;ABORT THE PASS
2339 017244 012600                      1$:    MOV      (SP)+,RO
2340 017246 000207                      RTS      PC
2341
2342
2343                      .SBTTL  CONFIG - DETERMINE CONFIGURATION OF SYSTEM
2344                      ;
2345                      ; SUBROUTINE - DETERMINE CONFIGURATION OF TSV05 SYSTEM.
2346                      ;
2347 017250                      CONFIG:
2348 017250 004737 015774              JSR      PC,SOFINIT
2349 017254 000207                      RTS      PC
2350                      .SBTTL  KTON,KTOFF - ENABLE/DISABLE MEMORY MANAGEMENT
2351                      ;
2352                      ; SUBROUTINE - ENABLE MEM MGT.
2353                      ;
2354 017256 005737 003132              KTON:  TST      KYFLG      ; GOT KT?
2355 017262 001403                      BEQ      1$              ; NO.
2356 017264 012737 000001 177572      MOV      $1,SRO      ; YES. ENABLE KT11.
2357 017272 000207                      1$:    RTS      PC
2358
2359                      ;

```

K6

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 KTON,KTOFF - ENABLE/DISABLE MEMORY MANAGEMENT

SEQ 075

```

2360 ; SUBROUTINE - DISABLE MEM MGT.
2361 ;
2362 017274 005737 003132 KTOFF; TST KTFLG ; GO1 KT11?
2363 017300 001405 BEQ 1$ ; NO.
2364 017302 000240 NOP
2365 017304 000240 NOP
2366 017306 012737 000000 177572 MOV #0,SRO ; DISABLE KT.
2367 017314 000207 1$; RTS PC
2368 ;SBTTL SETMAP - SETUP PAR6 MAPPING
2369
2370 ;*
2371 ;
2372 ; THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
2373 ; AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
2374 ; IS RETURNED BIASED TO PAR6.
2375 ;
2376 ; INPUTS:
2377 ;
2378 ; R0 HIGH ORDER ADDRESS BITS
2379 ; R1 LOW ORDER ADDRESS BITS
2380 ;
2381 ; OUTPUTS:
2382 ;
2383 ; R0 OFFSET INTO BLOCK WITH PAR6 BIAS (I.E. THE ADDRESS)
2384 ; CARRY SET IF SUCCESS
2385 ; CLR IF ERROR
2386 ;
2387 017316 SETMAP;
2388 017316 SAVREG
2389 017322 005737 003132 TST KTFLG ;SAVE R1-R4 UNTIL NEXT RETURN
2390 017326 001433 BEQ 10$ ;SYSTEM HAVE ABOVE 28K?
2391 017330 010102 MOV R1,R2 ;BR IF NO
2392 000006 .REPT 6 ;SAVE LOW ORDER BITS
2393 ASR R0 ;CONVERT WORD ADDRESS TO 32W BLOCKS
2394 ROR R1 ;MAKE IT DOUBLE PRECISION
2395 .ENDR
2396 017362 042701 000177 BIC #177,R1 ;ALINE FOR LOWER 4K BOUNDARY
2397 017366 020137 003132 CMP R1,KTFLG ;HIGHER THAN EXISTING MEMORY?
2398 017372 103011 BHIS 10$ ;BR IF YES
2399 017374 010137 172354 MOV R1,#KIPAR6 ;SETUP MAPPING REGISTER PAR6
2400 017400 042702 160000 BIC #160000,R2 ;SETUP DISPLACEMENT IN PAGE
2401 017404 062702 140000 ADD #140000,R2 ;ADD IN PAR6 BIAS
2402 017410 010200 MOV R2,R0 ;RETURN IN R0
2403 017412 000261 SEC ;SET SUCCESS
2404 017414 000401 BR 15$ ;
2405 017416 000241 10$; CLC ;SET FAILURE
2406 017420 000207 15$; RTS PC ;RETURN
2407 ;SBTTL FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
2408 ;*
2409 ; FILL MEMORY WITH A BACKGROUND PATTERN
2410 ;
2411 ; INPUTS:
2412 ;
2413 ; R0 - BACKGROUND PATTERN
2414 ; FREE - FIRST LOCATION AVAILABLE TO DIAGNOSTIC
2415 ; KTFLG - SET TO HIGHEST MEMORY LOCATION IF > 28K.
2416 ;

```

L6

TSV3 - GLOBAL AREAS      MACRO M1113 06-FEB-84 17:14  
FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN

SEQ 076

```

2417      ; OUTPUTS:
2418      ;
2419      ;      NONE
2420      ;
2421      ;
2422      FILLMEM:
2423      SAVREG
2424      JSR    PC,KTOFF      ;SAVE R1-R5 UNTIL NEXT RETURN
2425      MOV    RO,R3        ;DISABLE KT.
2426      MOV    FREE,R1      ;COPY TEST PATTERN
2427      MOV    FRESIZ,R2    ;GET FIRST FREE LOCATION
2428      10$: MOV    R3,(R1)+ ;SIZE OF FREE SPACE BELOW 28K.
2429      DEC    R2           ;STORE A BACKGROUND WORD
2430      BGT    10$         ;DONE ALL MEMORY IN FREE SPACE?
2431      TST    KTFLG        ;BR IF NO
2432      BEQ    55$         ; GOT KT?
2433      JSR    PC,KTON      ; NO. GET OUT.
2434      CLR    RO          ; YES. ENABLE KT.
2435      MOV    PST32W,R1    ;HIGH ORDER ADDRESS START
2436      .REPT 6             ;GET >28K START ADDRESS (IN 32W BLOCKS)
2437      CLC                ;CLEAR C BIT
2438      ROL    R1           ;CONVERT BLOCKS TO WORDS
2439      ROL    RO          ;MAKE IT DOUBLE PRECISION
2440      .ENDR
2441      JSR    PC,SETMAP    ;SETUP PAR6 MAPPING REGISTER
2442      30$: MOV    R3,(R0)+ ;STORE TEST PATTERN IN >28K ADDRESS
2443      CMP    RO,#160000  ;END OF PAR6 MAPPING AREA?
2444      BLO    30$         ;BR IF NO
2445      SUB    #20000,RO    ;BACKUP INTO PAR6 MAPPING BEGIN
2446      ADD    #200,#KIPAR6 ;POINT TO NEXT 4K BLOCK >28K.
2447      CMP    #KIPAR6,KTFLG ;END OF MEMORY?
2448      BEQ    50$         ;BR IF YES
2449      TST    T23A        ;11/23A?
2450      BEQ    35$         ;NO KEEP GOING
2451      MOV    SRO,R4       ;GET SRO CONTENTS
2452      BIC    #177761,R4  ;CLEAR ALL BUT PAGE NUMBER
2453      CMP    #16,R4      ;SEE IF PAGE 7
2454      BEQ    50$         ;EXIT IF THERE
2455      35$: TST    T23B    ;11/23B?
2456      BEQ    45$         ;NO KEEP GOING
2457      CMP    #KIPAR6,#7600 ;REACHED 18 BITS?
2458      BHS    40$         ;YES
2459      BR     45$         ;NO KEEP GOING
2460      MOV    #20,SRS      ;SET 22 BIT RFLOCATION
2461      40$: JMP    30$     ;KEEP GOING ON ETC.
2462      50$: JSR    PC,KTOFF ;DISABLE KT.
2463      55$: RTS    PC
2464      .SBTTL CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN
2465      ;
2466      ; COMPARE MEMORY WITH A BACKGROUND PATTERN
2467      ;
2468      ; INPUTS:
2469      ;
2470      ;      RO - BACKGROUND PATTERN
2471      ;      FREE - FIRST LOCATION AVAILABLE TO DIAGNOSTIC
2472      ;      KTFLG - SET TO HIGHEST MEMORY LOCATION IF > 28K.
2473      ;

```

M6

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN

SEQ 077

```

2474      ; OUTPUTS:
2475      ;
2476      ; CARRY - SET IF NO ERROR
2477      ; CARRY - CLR IF ERROR
2478      ;
2479      ; IMPLICIT OUTPUTS:
2480      ;
2481      ; ERRHI - ERROR HIGH ADDRESS
2482      ; ERRLO - ERROR LOW ADDRESS
2483      ; EXPD - EXPECTED DATA
2484      ; RECV - RECEIVED DATA
2485      ;
2486 017660 CMPMEM:
2487 017660 SAVREG
2488 017664 010003 MOV R0,R3 ;SAVE R1-R5 UNTIL NEXT RETURN
2489 017666 004737 017274 JSR PC,KTOFF ;COPY TEST PATTERN
2490 017672 013701 C03124 MOV FREE,R1 ;DISABLE KT.
2491 017676 013702 003126 MOV FRESIZ,R2 ;GET FIRST FREE LOCATION
2492 017702 020311 10$: CMP R3,(R1) ;SIZE OF FREE SPACE BELOW 28K.
2493 017704 001411 BEQ 15$ ;FREE SPACE LOCATION EQUAL TO EXPD?
2494 017706 010137 002240 MOV R1,ERRLO ;BR IF YES
2495 017712 005037 002236 CLR ERRHI ;SAVE ADDRESS IN ERROR
2496 017716 010337 002232 MOV R3,EXPD ;NO HIGH ADDRESS
2497 017722 011137 002234 MOV (R1),RECV ;SAVE EXPD FOR ERROR REPORT
2498 017726 000474 BR 50$ ;SAVE RECV FOR ERROR REPORT
2499 017730 005721 15$: TST (R1)+ ;
2500 017732 005302 DEC R2 ;POINT TO NEXT ADDRESS
2501 017734 003362 BGT 10$ ;DONE ALL MEMORY IN FREE SPACE?
2502 017736 005737 003132 TST KTFLG ;BR IF NO
2503 017742 001472 BEQ 55$ ;GOT KT?
2504 017744 004737 017256 JSR PC,KTON ;NO. GET OUT.
2505 017750 005000 CLR R0 ;YES. ENABLE KT.
2506 017752 013701 003152 MOV PST32W,R1 ;HIGH ORDER ADDRESS START
2507 000006 .REPT 6 ;GET >28K START ADDRESS (IN 32W BLOCKS)
2508 ROL R1 ;
2509 ROL R0 ;CONVERT BLOCKS TO WORDS
2510 .ENDR ;MAKE IT DOUBLE PRECISION
2511 020006 042701 000177 BIC #177,R1 ;ALINE 4K BOUNDARY
2512 020012 010046 MOV R0,-(SP) ;SAVE HIGH ORDER
2513 020014 010146 MOV R1,-(SP) ;SAVE LOW ORDER
2514 020016 004737 017316 JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
2515 020022 010004 MOV R0,R4 ;COPY ADDRESS BIASED TO PAR6
2516 020024 012601 MOV (SP)+,R1 ;RESTORE LOW ORDER IN NON PAR6 FORMAT
2517 020026 012600 MOV (SP)+,R0 ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
2518 020030 020314 30$: CMP R3,(R4) ;ABOVE 28K LOCATION EQUAL EXPD?
2519 020032 001411 BEQ 32$ ;BR IF YES
2520 020034 010037 002236 MOV R0,ERRHI ;SAVE HIGH ORDER IN ERROR
2521 020040 010137 002240 MOV R1,ERRLO ;SAVE LOW ORDER IN ERROR
2522 020044 010337 002232 MOV R3,EXPD ;SAVE EXPD FOR ERROR REPORT
2523 020050 011437 002234 MOV (R4),RECV ;SAVE RECV FOR ERROR REPORT
2524 020054 000421 BR 50$ ;
2525 020056 062701 000002 32$: ADD #2,R1 ;UPDATE NON PAR6 ADDRESS
2526 020062 005500 ADC R0 ;MAKE IT DOUBLE PRECISION ADD
2527 020064 062704 000002 ADD #2,R4 ;UPDATE PAR FORMAT ADDRESS
2528 020070 020427 160000 CMP R4,#160000 ;END OF PAR6 MAPPING AREA?
2529 020074 103755 BLO 30$ ;BR IF NO
2530 020076 162704 020000 SUB #20000,R4 ;BACKUP INTO PAR6 MAPPING BEGIN

```

N6

TSV3 - GLOBAL AREAS      MACRO M1113 06-FEB-84 17:14  
CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN

SEQ 078

```

2531 020102 062737 000200 172354      ADD    #200,#KIPAR6 ;POINT TO NEXT 4K BLOCK >28K.
2532 020110 023737 172354 003132      CMP    #KIPAR6,KTFLG ;END OF MEMORY?
2533 020116 101744                      BLOS   30$ ;BR IF NO
2534 020120 004737 017274      50$:    JSR    PC,KTOFF ;TURN OFF MEMORY MAPPING
2535 020124 000241                      CLC    ;SET FAILURE
2536 020126 000403                      BR     60$ ;
2537 020130 004737 017274      55$:    JSR    PC,KTOFF ;TURN OFF MEMORY MAPPING
2538 020134 000261                      SEC    ;SET SUCCESS
2539 020136 000207      60$:    RTS     PC
2540                      .SBTTL  REGSAV - SAVE R1-R5 ON STACK
2541                      ;+
2542                      ;
2543                      ;ROUTINE TO
2544                      ;SAVE R1 THROUGH R5 ON THE STACK
2545                      ;
2546                      ;CALLING SEQUENCE:
2547                      ;
2548                      ;      JSR    R5,REGSAV
2549                      ;
2550                      ;THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
2551                      ;THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
2552                      ;THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTORE
2553                      ;REGISTERS.
2554                      ;
2555                      ;THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
2556                      ;CALLED VIA A JSR PC INSTRUCTION
2557                      ;
2558                      ;-
2559
2560 020140                      REGSAV:
2561 020140 010446                      MOV    R4,-(SP)
2562 020142 010346                      MOV    R3,-(SP)
2563 020144 010246                      MOV    R2,-(SP)
2564 020146 010146                      MOV    R1,-(SP)
2565 020150 010546                      MOV    R5,-(SP)
2566 020152 016605 000012          MOV    10.(SP),R5
2567 020156 004736                      JSR    PC,#(SP)+
2568 020160 012601                      MOV    (SP)+,R1
2569 020162 012602                      MOV    (SP)+,R2
2570 020164 012603                      MOV    (SP)+,R3
2571 020166 012604                      MOV    (SP)+,R4
2572 020170 012605                      MOV    (SP)+,R5
2573 020172 000207                      RTS     PC
2574                      .SBTTL  GETPAT - GET 8 BIT PATTERN FROM OPERATOR
2575                      ;+
2576                      ;
2577                      ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
2578                      ;
2579                      ;INPUTS:
2580                      ;
2581                      ;      NONE.
2582                      ;
2583                      ;OUTPUTS:
2584                      ;
2585                      ;      R0      OCTAL NUMBER FROM THE OPERATOR
2586                      ;
2587                      ;CALLING SEQUENCE:

```

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
GETPAT - GET A BIT PATTERN FROM OPERATOR

SEQ 079

```

2588      ;
2589      ;      JSR      PC,GETPAT
2590      ;
2591      ;
2592      ;
2593 020174      GETPAT::
2594 020174      ;      SAVREG      ;SAVE THE GENERAL REGISTERS
2595 020200      1$:      GMANID    DATASC,PATDAT,0,377,0,377,NO
                TRAP      C$GMAN
                BR        10000$
                .WORD     PATDAT
                .WORD     T$CODE
                .WORD     DATASC
                .WORD     377
                .WORD     T$LOLIM
                .WORD     T$HILIM
                10000$:
2596 020220      BNCOMPLETE      1$      ;RETRY IF ERROR
                BCC      1$
2597 020222      013700      020230      MOV      PATDAT,R0      ;DATA PATTERN FROM OPERATOR
2598 020226      000207      RTS      PC      ;RETURN TO CALLER
2599
2600      ;
2601      ;LOCAL DATA AREA
2602      ;
2603
2604 020230      000C00      116      124      PATDAT: .WORD     0      ;TEMPORARY STORAGE FOR DATA
2605 020232      105      DATASC: .ASCIZ  'ENTER DATA PATTERN'
2606      .EVEN
2607      .SBTTL  GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE
2608      ;
2609      ;
2610      ;ROUTINE TO ISSUE A MENU AND GET
2611      ;THE OPERATOR'S RESPONSE.
2612      ;
2613      ;INPUTS:
2614      ;
2615      ;      R0      ADDRESS OF ASCIZ STRING OF MENU
2616      ;      R1      MAXIMUM ALLOWABLE OPERATOR RESPONSE
2617      ;
2618      ;OUTPUTS:
2619      ;
2620      ;      R0      NUMBER OF THE OPERATOR'S SELECTION
2621      ;
2622      ;
2623      ;
2624 020256      GETSEL::
2625 020256      ;      SAVREG      ;SAVE GENERAL REGISTERS
2626 020262      010002      MOV      R0,R2      ;SAVE THE MENU ADDRESS
2627 020264      010203      1$:      MOV      R2,R3      ;START OF MENU STRING
2628 020266      005713      2$:      TST      (R3)      ;END OF ASCII ?
2629 020270      001412      BEQ      3$      ;BRANCH IF ALL LINES DISPLAYED
2630 020272      PRINTF      0SELASC,(R3),      ;DISPLAY THE MENU
                MOV      (R3)+,(SP)
                MOV      0SELASC,-(SP)
                MOV      02,(SP)
                MOV      SP,R0

```

TSV3 GLOBAL AREAS MACHO M1113 06 FEB 84 17:14  
GETSEL ISSUE MENU AND GET OPERATOR RESPONSE

TSV3 GLOBAL AREAS MACHO M1113 06 FEB 84  
GETSEL ISSUE MENU AND GET OPERATOR RESPONSE

```

TRAP      C#PNTF
ADD       #6,SP
BR        2#
3#:       GMANID  MENASC,MENRES,D,-1,0,-1,NO
TRAP      C#GMAN
BR        10001#
        .WORD   MENRES
        .WORD   T#CODE
        .WORD   MENASC
        .WORD   -1
        .WORD   T#LOLIM
        .WORD   T#HILIM
10001#:   BNCOMPLETE      1#           ;RETRY IF ERROR
BCC       1#
MOV       MENRES,RO        ;GET THE OPERATOR'S REPLY
CMP       RO,R1            ;COMPARE TO MAXIMUM ALLOWED
BLOS     5#               ;BRANCH IF OK
PRINTF    #MENERR          ;DISPLAY ERROR MESSAGE
MOV       #MENERR,-(SP)
MOV       #1,-(SP)
MOV       SP,RO
TRAP      C#PNTF
ADD       #4,SP
BR        1#               ;RETRY
5#:       RTS            PC          ;RETURN TO CALLER
MENERR:   .ASCIZ  '###A *** Menu Selection Too Large ***'
SELASC:   .ASCIZ  '###T'
MENASC:   .ASCIZ  'Enter Menu Selection: '
        .EVEN
MENRES:   .WORD    0
        .SBTTL  CHKMAN - CHECK MANUAL INTERVENTION LEGALITY
;
;ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
;
;INPUT:
;
;      NONE.
;
;OUTPUT:
;
;      CARRY    0      MANUAL INTERVENTION NOT ALLOWED
;               1      MANUAL INTERVENTION IS OK
;
;SIDE EFFECTS:
;
;      A MESSAGE IS DISPLAYED WARNING THAT TEST IS
;      NOT EXECUTED IF MANUAL INTERVENTION IS NOT
;      ALLOWED.
;
;
CHKMAN:   SAVREG          ;SAVE THE REGISTERS
MANUAL    ;SEE IF MANUAL INTERVENTION
TRAP      C#MANI

```

```
CHKMAN: : SAVREG          I SAVE THE REGISTERS
          MANUAL          I SEE IF MANUAL INTERVENTION OK
          TRAP           C$MANI
```

ISV3 GLOBAL AREAS MACRO M1113 06 FEB 84 17:14  
 CHKMAN - CHECK MANUAL INTERVENTION LEGALITY

SEQ 081

```

2670 020506                                BCOMPLETE 1$          ;BRANCH IF ALLOWED
      020506 103411                        BCS 1$
2671 020510                                PRINTF #NOMAN          ;PRINT THE WARNING MESSAGE
      020510 012746 020534                MOV #NOMAN, -(SP)
      020514 012746 000001                MOV #1, -(SP)
      020520 010600                        MOV SP, R0
      020522 104417                        TRAP C$PNTF
      020524 062706 000004                ADD #4, SP
2672 020530 000241                        CLC                      ;CLEAR CARRY FOR ERROR
2673 020532 000207                        1$: RTS PC              ;RETURN
2674
2675 020534 045 116 045 NOMAN: .ASCIZ 'NMA *** Manual Intervention not Allowed - Test Aborted ***'
2676 .even
2677 .SBTTL ENVIRN - SETUP FREE DIAGNOSTIC SPACE
2678
2679 ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
2680
2681 020630                                ENVIRN: MEMORY R0
      020630 104431                        TRAP C$MEM
2682 020632 010037 003124                MOV R0, FREE          ; GET 1ST FREE ADDRESS...
2683 020636 062737 000002 003124        ADD #2, FREE
2684 020644 011037 003126                MOV (R0), FRESIZ      ;...AND WORD COUNT.
2685 020650 162737 000004 003126        SUB #4, FRESIZ
2686 020656 013702 002012                MOV L$UNIT, R2        ; GET NUMBER OF UNITS
2687 020662 162737 000007 003126 10$: SUB #7, FRESIZ          ; TAKE AWAY 7 WORDS PER UNIT
2688 020670 005302                        DEC R2
2689 020672 001373                        BNE 10$
2690 020674 013700 003124                MOV FREE, R0          ;GET FIRST FREE ADDRESS
2691 020700 063700 003126                ADD FRESIZ, R0         ;POINT TO LAST FREE ADDRESS
2692 020704 162700 000002                SUB #2, R0            ;BACKUP 1 WORD
2693 020710 010037 003130                MOV R0, FREEH1        ;STORE LAST FREE ADDRESS
2694 020714 000240                        NOP
2695 020716 012701 177520                MOV #BDVPCR, R1      ;GET BDV11 PCR ADDRESS
2696 020722 010102                        MOV R1, R2            ;COPY TO R2
2697 020724 062702 000002                ADD #2, R2            ;SET THE RANGE
2698 020730 004737 016376                JSR PC, XNXM          ;SEE IF WE HAVE ONE
2699 020734 103001                        BCC 15$              ;OK TO SET FLAGS
2700 020736 000445                        BR 40$                ;RETURN WITH FLAGS CLEAR
2701 020740 013701 177520                15$: MOV BDVPCR, R1    ;SAVE PCR CONTENTS
2702 020744 062701 000001                ADD #1, R1            ;ADD ONE TO IT
2703 020750 012702 177520                MOV #BDVPCR, R2      ;GET BDV11 PCR ADDRESS
2704 020754 005212                        INC (R2)              ;TRY TO WRITE TO IT
2705 020756 013703 177520                MOV BDVPCR, R3        ;GET RESULTS
2706 020762 020103                        CMP R1, R3            ;DID IT CHANGE?
2707 020764 001017                        BNE 20$              ;NO, MUST BE 11/23B
2708 020766 005237 003144                INC T23A              ;SET THE FLAG
2709 020772 042737 170000 002120        BIC #170000, L$HIME    ;SUPERVISOR COULD BE WRONG
2710 021000 000240                        NOP
2711 021002                                PRINTF #M8186
      021002 012746 005554                MOV #M8186, -(SP)
      021006 012746 000001                MOV #1, -(SP)
      021012 010600                        MOV SP, R0
      021014 104417                        TRAP C$PNTF
      021016 062706 000004                ADD #4, SP
2712 021022 000413                        BR 40$                ;RETURN
2713 021024 005237 003146                20$: INC T23B          ;SET THE FLAG
2714 021030 000240                        NOP                    ;BR 40$ FOR RELEASE

```

E7

TSV3 - GLOBAL AREAS      MACRO M1113    06-FEB-84 17:14  
 ENVIRN - SETUP FREE DIAGNOSTIC SPACE

SEQ 082

```

2715 021032          PRINTF 0M8189      ;TELL THE SYSTEM TYPE
      021032 012746 005645      MOV 0M8189,-(SP)
      021036 012746 000001      MOV 01,-(SP)
      021042 010600          MOV SP,R0
      021044 104417          TRAP C#PNTF
      021046 062706 000004      ADD 04,SP
2716 021052 000207      40$: RTS PC      ;RETURN
2717          .SBTTL KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS
2718          ;
2719          ;
2720          ;ROUTINE TO INIT KT-11
2721          ;
2722          ;
2723          ;
2724 021054          KTINIT:
2725 021054 005037 003132      CLR KTFLG      ; INIT >28K MEMORY FLAG
2726 021060 005037 003134      CLR KTENABLE   ; INIT TEST >28K FLAG
2727 021064 023727 002120 001577  CMP L#HIME,01577 ; GOT ENOUGH MEMORY (>28K)?
2728 021072 101444          BLOS 9$          ; NO.
2729 021074 013700 000004      MOV 00ERRVEC,R0 ; SAVE OLD ERR VEC PTR.
2730 021100 012737 021172 000004  MOV 02$,00ERRVEC ; SET ERR VEC PTR.
2731 021106 005737 177572      TST 00$R0      ; GOT KT11?
2732 021112 000240          NOP              ; (TRAP IF NO).
2733 021114 013737 002120 003132  MOV L#HIME,KTFLG ; YES. SET KT FLAG.
2734 021122 042737 000177 003132  BIC 0177,KTFLG ;
2735 021130 010037 000004      MOV R0,00ERRVEC ; RESTORE OLD ERR VEC PTR.
2736 021134 005000          CLR R0          ; R0 = AR DATA.
2737 021136 012701 172340      MOV 0KIPAR0,R1 ; R1 = KI REGS PTR.
2738 021142 012761 077406 177740 1$: MOV 077406,-40(R1) ; SET DESCRIPTOR REG.
2739 021150 010021          MOV R0,(R1)+    ; SET KIPAR REG.
2740 021152 062700 000200      ADD 0200,R0    ; BUMP AR DATA BY "4K".
2741 021156 020027 002000      CMP R0,02000  ; AT "I/O"?
2742 021162 001367          BNE 1$          ; NO.
2743 021164 012741 177600      MOV 0177600,-(R1) ; YES. SET KTPAR7 FOR I/O.
2744 021170 000405          BR 9$
2745          ;
2746 021172 012716 021200      2$: MOV 06$,(SP) ; SET UP RETURN
2747 021176 000002          RTI              ; RTI TO NEXT LOCATION
2748          ;
2749 021200 010037 000004      6$: MOV R0,00ERRVEC ; RESTORE OLD ERR VEC PTR.
2750          ;
2751 021204 000207      9$: RTS PC
2752          ;
2753          ;
2754          ; SUBROUTINE TO SET EXTENDED FEATURES SWITCH
2755          ;
2756          ; Requires that SOFINIT and WRTCHR have been done previous to call.
2757          ;
2758          ;
2759          ; INPUTS:
2760          ; R5      CURRENT UNIT NUMBER
2761          ;
2762          ; OUTPUTS:
2763          ; The Extended Features Switch is set.
2764          ;
2765          ;
2766          ;
2765 021206          INVERT::

```

F7

TSV3 - GLOBAL AREAS      MACRO M1113 06-FEB-84 17:14  
 KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

SEQ 083

```

2767 021206 005737 002226          TST      EXTFEA          ; IS SWITCH SET?
2768 021212 001020          BNE      1$          ; YES,EXIT STAGE RIGHT!(on the next one outa town!)
2769 021214 012737 100206 021260    MOV      $100206,CMDPKT    ; WRT SUB-SYS MEM CMD
2770 021222 012737 021270 021262    MOV      $WSMBK,CMDPKT+2  ; MSG BUF ADDR
2771 021230 012737 000006 021266    MOV      $6,CMDPKT+6      ; BYTE COUNT
2772 021236 012737 100010 021270    MOV      $100010,WSMBK    ; INVERT THE SWITCH
2773 021244 012704 021260          MOV      $CMDPKT,R4        ; SET CMDPKT INTO R4
2774 021250 004737 010662          JSR      PC,WRTCHR        ; DO IT
2775 021254 000207          1$:      RTS      PC              ; RETURN
2776
2777          ;          COMMAND PACKET.
2778
2779          021260          .          *          <.,+3>&177774    ;MUST BE ON MOD 4 BOUNDRY.
2780
2781 021260 000000          CMDPKT:: 0          ;1ST WORD IS TS05 COMMAND.
2782 021262 000000          0          ;2ND WORD IS THE BUFFER LOW ADDRESS.
2783 021264 000000          0          ;3RD WORD IS THE BUFFER HIGH ADDRESS.
2784 021266 000000          0          ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.
2785
2786          ;          WRITE SUB-SYSTEM MEMORY CHARACTERISTIC BLOCK.
2787
2788 021270 000000          WSMBK:: 0          ;1ST WORD:: SEL 0
2789 021272 000000          0          ;2ND WORD:: SEL 2
2790 021274 000000          0          ;3RD WORD:: SEL 4
2791          .EVEN
2792
2793          ;          SUBROUTINE TO CHECK WETHER OR NOT WE'LL TEST NXM
2794          ;
2795          ;
2796          ;INPUTS:
2797          ;OUTPUTS:
2798          ;          The NXMFLG is set if we can test.
2799          ;          The NXML0 and NXMHI addresses are setup.
2800          ;
2801          ;
2802 021276          MEMCK::
2803
2804 021276          SAVREG          ;SAVE THE REGISTERS
2805 021302 005037 003136          CLR      NXMFLG          ;CLEAR THE FLAG
2806 021306 005037 003140          CLR      NXML0          ;CLEAR THE TEST ADDRESS LO
2807 021312 005037 003142          CLR      NXMHI          ;CLEAR THE TEST ADDRESS HI
2808 021316 005737 003146          TST      T23B          ;IS IT A 11/23B?
2809 021322 001407          BEQ      1$          ;NO
2810 021324 023727 002120 007777    CMP      L$HIME,$7777    ; GREATER THAN 128K
2811 021332 103406          BLO      2$          ; NO
2812 021334 004737 021452          JSR      PC,NXMTST        ;SETUP THE ADDRESS
2813 021340 000427          BR      13$          ;SET THE FLAG AND EXIT
2814 021342 005737 003144          1$:      TST      T23A          ;IS IT A 11/23A?
2815 021346 001413          BEQ      4$          ;NO
2816 021350 023727 002120 005777    CMP      L$HIME,$5777    ;GREATER THAN 96K
2817 021356 101023          BHI      14$          ;YES,23A/23B WITH 128K MEMORY
2818 021360 023727 002120 003777    CMP      L$HIME,$3777    ;GREATER THAN 64K BUT LESS THAN 92K?
2819 021366 103403          BLO      4$          ;NO, CHECK 24K
2820 021370 004737 021452          JSR      PC,NXMTST        ;SETUP THE ADDRESS
2821 021374 000411          BR      13$          ;SET THE FLAG AND EXIT
2822 021376 023727 002120 001577    CMP      L$HIME,$1577    ;GREATER THAN 24K BUT LESS THAN 64K?
2823 021404 103410          BLO      14$          ;NO, TELL THEM AND EXIT WITH FLAG CLEAR

```

G7

TSV3 - GLOBAL AREAS MACRO M1113 06-FEB-84 17:14  
 KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

SEQ 084

```

2824 021406 004737 021452      JSR    PC,NXMTST      ;SETUP THE ADDRESS
2825 021412 062737 000077 003142  ADD    #77,NXMHI    ;FOOL THE 11/02 & 11/03
2826 021420 005237 003136      INC    NXMFLG      ;SET THE FLAG
2827 021424 000411              BR     15$        ;EXIT
2828 021426 000410              BR     15$        ;NOP FOR PRINTOUT
2829 021430              PRINTF  #NOMEM          ;TELL THEM & EXIT ***NO PRINT*****
      021430 012746 005460      MOV    #NOMEM,-(SP)
      021434 012746 000001      MOV    #1,-(SP)
      021440 010600              MOV    SP,R0
      021442 104417              TRAP   C#PNTF
      021444 062706 000004      ADD    #4,SP
2830 021450 000207              15$: RTS    PC      ;RETURN
2831
2832
2833      ;*
2834      ; SUBROUTINE TO SETUP THE NXM ADDRESS FOR TESTING
2835      ;
2836      ; OUTPUTS: NXMLO,NXMHI      ;SETUP WITH NXM ADDRESS
2837      ;
2838      ; -
2839 021452 013701 002120      NXMTST: MOV    L#HIME,R1      ;GET TOP OF MEMORY
2840 021456 062701 000200      ADD    #200,R1      ;MAKE IT I/O BLOCK OR OTHER NXM
2841 021462 042701 000177      BIC    #177,R1
2842 021466 010102              MOV    R1,R2      ;RESAVE RESULTS
2843              000006      .REPT    6
2844              ASL    R1      ;PUT IN PLACE FOR XFER
2845              .ENDR
2846 021504 010137 003140      MOV    R1,NXMLO      ;SAVE TEST ADDRESS LOW
2847              000012      .REPT    10
2848              ASR    R2      ;PUT IN PLACE FOR XFER
2849              .ENDR
2850 021534 042702 177700      BIC    #177700,R2      ;DON'T WANT ILA!
2851 021540 010237 003142      MOV    R2,NXMHI      ;SAVE TEST ADDRESS HIGH
2852 021544 000207              RTS    PC      ;RETURN
2853
2854
2855
2856 021546              ENDMOD

```

H7

TSV4 - MISCELLANEOUS SECTIONS MACRO M1113 06-FEB-84 17:14  
KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

SEQ 085

7  
8 .TITLE TSV4 - MISCELLANEOUS SECTIONS  
9 021546 BGNMOD TSV4  
10 021546 TSV4::

I7

TSV4 - MISCELLANEOUS SECTIONS    MACRO M1113    06-FEB-84 17:14  
PROTECTION TABLE

SEQ 086

```
17                                     .SBTTL  PROTECTION TABLE
18 021546                             BGNPROT
   021546
19 021546 177777 177777 177777      L$PROT:: .WORD  -1, -1, -1, -1
20 021556                             ENDPROT
```

;NO DEVICE PROTECTION REQUIRED.

TSV4 - MISCELLANEOUS SECTIONS  
INITIALIZE SECTION

MACRO M1113 06-FEB-84 17:14

SEQ 087

```

22                                     .SBTTL  INITIALIZE SECTION
23
24                                     ;**
25                                     ;THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
26                                     ;AT THE BEGINNING OF EACH PASS.
27
28                                     ;IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
29                                     ;IF "CONTINUE", NOTHING IS REQUIRED.
30
31                                     ;
32                                     ;--
33                                     ;*
34                                     ;INSERT TEMPORARY JUMP TO ODT
35                                     ;-
36 021556                                BGNINIT
37 021556                                L$INIT::
38 021556 005037 002226                   40$: CLR      EXTFEA
39 021562 005037 003136                   CLR      NXMFLG
40 021566 012737 006360 002200           MOV      @EPT1,EPTSW
41 021574 005037 003154                   CLR      SIFLAG
42 021600 005037 003134                   CLR      KTENABLE
43 021604 005037 002302                   CLR      RAMSIZ
44 021610                                READEF   @EF,CONTINUE
45 021610 012700 000036                   MOV      @EF,CONTINUE,RO
46 021614 104447                         TRAP     C$REFG
47 021616                                BNCOMPLETE 1$
48 021616 103023                         BCC      1$
49 021620 023737 002202 002012           CMP      UNITN,L$UNIT
50 021626 103070                         BHIS     4$
51 021630 005737 003112                   TST      DUFLG
52 021634 100472                         BMI      NXTU
53 021636 013701 002202                   MOV      UNITN,R1
54 021642 006301                         ASL      R1
55 021644 005761 003176                   TST      ERTABL(R1)
56 021650 001516                         BEQ      SETU
57 021652 032761 040000 003176           BIT      @BIT14,ERTABL(R1)
58 021660 001060                         SNE      NXTU
59 021662                                EXIT      INIT
60 021662 104432                         TRAP     C$EXIT
61 021664 000416                         .WORD    L10030-
62 021666                                READEF   @EF,NEW
63 021666 012700 000035                   MOV      @EF,NEW,RO
64 021672 104447                         TRAP     C$REFG
65 021674                                BNCOMPLETE NXTU
66 021674 103052                         BCC      NXTU
67 021676                                READEF   @EF,START
68 021676 012700 000040                   MOV      @EF,START,RO
69 021702 104447                         TRAP     C$REFG
70 021704                                BCOMPLETE 2$
71 021704 103404                         BCS      2$
72 021706                                READEF   @EF,RESTART
73 021706 012700 000037                   MOV      @EF,RESTART,RO
74 021712 104447                         TRAP     C$REFG
75 021714                                BNCOMPLETE 31$
76 021714 103031                         BCC      31$
77 021716                                BRESET
78 021716 104433                         TRAP     C$RESET

```

```

;SET UP PRIMARY MESSAGE FOR REPLACEMENT
;CLEAR "SOFT INIT" FLAG
;CLEAR TEST ABOVE 28K FLAG
;CLEAR RAM SIZE FOR RAMERR ROUTINE

```

```

;UNIT IN RANGE?
;BR IF NO.
;DROPPED UNIT?
;BR IF YES

```

```

;DROPPED?
;DO NOTHING IF "CONTINUE".

```

```

;TAKE NEXT UNIT IF NOT NEW PASS.

```

```

;1ST PASS, BUS-INIT...
;BUS RESET.

```

K7

TSV4 - MISCELLANEOUS SECTIONS    MACRO M1113    06-FEB-84 17:14  
INITIALIZE SECTION

SEQ 088

|     |        |        |        |         |           |               |           |                                          |
|-----|--------|--------|--------|---------|-----------|---------------|-----------|------------------------------------------|
| 63  | 021720 | 005037 | 002214 |         | CLR       | TSTCNT        |           | ;NUMBER OF TESTS RUN IN PASS             |
| 64  | 021724 | 005037 | 002222 |         | CLR       | FATFLG        |           | ;CLEAR FATAL ERROR COUNT                 |
| 65  | 021730 | 005037 | 003144 |         | CLR       | T23A          |           | ;CLEAR 11/23A FLAG                       |
| 66  | 021734 | 005037 | 003146 |         | CLR       | T23B          |           | ;CLEAR 11/23B FLAG                       |
| 67  |        |        |        |         | MOV       | #340,-(SP)    |           |                                          |
| 68  |        |        |        |         | MOV       | #20,-(SP)     |           | ;RETURN TO DEBUGGER                      |
| 69  |        |        |        |         | JMP       | 0.ODT         |           | ;ENTER THE DEBUGGER                      |
| 70  | 021740 | 005037 | 003400 |         | CLR       | SKIPT         |           | ;CLEAR THE SUBTEST "SKIPPER"             |
| 71  | 021744 |        |        | 20\$:   |           |               |           |                                          |
| 72  | 021744 | 012737 | 177777 | 002204  | MOV       | #-1,QVP       |           | ;...QUICK VERIFY...                      |
| 73  | 021752 | 004737 | 020630 |         | JSR       | PC,ENVIRN     |           | ;SET ENVIRONMENT.                        |
| 74  | 021756 | 004737 | 021054 |         | JSR       | PC,KTINIT     |           | ;INITIALIZE KT MEMORY MANAGEMENT         |
| 75  | 021762 | 012700 | 003176 |         | MOV       | #ERTABL,RO    |           |                                          |
| 76  | 021766 | 005020 |        | 30\$:   | CLR       | (RO)+         |           | ;CLEAR THE ERROR TABLE                   |
| 77  | 021770 | 020027 | 003376 |         | CMP       | RO,#ERTABE    |           |                                          |
| 78  | 021774 | 103774 |        |         | BLO       | 30\$          |           |                                          |
| 79  | 021776 | 000404 |        |         | BR        | 4\$           |           |                                          |
| 80  | 022000 | 005037 | 002204 | 31\$:   | CLR       | QVP           |           |                                          |
| 81  | 022004 | 000137 | 022054 |         | JMP       | PASRPT        |           | ;GO REPORT THE STATUS                    |
| 82  |        |        |        |         |           |               |           |                                          |
| 83  | 022010 |        |        | 4\$:    |           |               |           |                                          |
| 84  | 022010 | 012737 | 177777 | 002202  | NEWPAS:   | MOV           | #-1,UNITN | ;INIT UNIT NUMBER...                     |
| 85  | 022016 | 005037 | 002220 |         | CLR       | DEVcnt        |           | ;CLEAR COUNT OF DEVICES RUNNING          |
| 86  | 022022 |        |        | NXTU:   | BREAK     |               |           |                                          |
| 87  | 022024 | 005237 | 002202 |         | TRAP      | C\$BRK        |           |                                          |
| 88  | 022030 | 023737 | 002202 | 002012  | INC       | UNITN         |           | ;...AND SET NEXT UNIT NUMBER.            |
| 89  | 022036 | 103423 |        |         | CMP       | UNITN,L\$UNIT |           |                                          |
| 90  | 022040 | 012737 | 177777 | 003112  | BLO       | SETU          |           |                                          |
| 91  | 022046 | 000401 |        |         | MOV       | #-1,DUFLG     |           |                                          |
| 92  | 022050 |        |        |         | BR        | 11\$          |           |                                          |
| 93  | 022052 | 000240 |        | 11\$:   | DOCLN     |               |           | ;ABORT, NO MORE UNITS.                   |
| 94  | 022054 |        |        | PASRPT: | TRAP      | C\$DCLN       |           |                                          |
| 95  | 022054 | 023727 | 002012 | 000001  | NOB       |               |           |                                          |
| 96  | 022062 | 101752 |        |         | CMP       | L\$UNIT,#1    |           | ;HOW MANY UNITS SELECTED?                |
| 97  | 022064 | 005737 | 002220 |         | BLOS      | NEWPAS        |           | ;BR IF ONLY 1                            |
| 98  | 022070 | 001747 |        |         | TST       | DEVcnt        |           | ;ARE ANY STILL RUNNING?                  |
| 99  | 022072 |        |        |         | BEQ       | NEWPAS        |           | ;BR IF NO                                |
| 100 | 022074 | 032700 | 000100 |         | RFLAGS    | RO            |           |                                          |
| 101 | 022100 | 001343 |        |         | TRAP      | C\$RFLA       |           |                                          |
| 102 |        |        |        |         | BIT       | #ISR,RO       |           | ;SHOULD WE PRINT STATISTICS              |
| 103 | 022102 |        |        |         | BNE       | NEWPAS        |           | ;BR IF NO                                |
| 104 | 022104 | 000741 |        |         | DORPT     |               |           |                                          |
| 105 | 022106 |        |        | 10\$:   | TRAP      | C\$DRPT       |           |                                          |
| 106 |        |        |        |         | BR        | NEWPAS        |           |                                          |
| 107 | 022106 |        |        | SETU:   | GPHARD    | UNITN,RO      |           | ;GET UNIT N P-TABLE POINTER.             |
| 108 | 022114 | 013700 | 002202 |         | MOV       | UNITN,RO      |           |                                          |
| 109 | 022116 | 005037 | 003112 |         | TRAP      | C\$GPHRD      |           |                                          |
| 110 | 022122 | 005237 | 002220 |         | BNCMPLETE | NXTU          |           | ;BR IF UNIT NOT AVAILABLE.               |
| 111 | 022126 | 012001 |        |         | BCC       | NXTU          |           |                                          |
| 112 | 022130 | 010137 | 002206 |         | CLR       | DUFLG         |           | ;CLEAR "DROPPED" FLAG.                   |
|     |        |        |        |         | INC       | DEVcnt        |           |                                          |
|     |        |        |        |         | MOV       | (RO)+,R1      |           | ;GET 1ST REGISTER ADDRESS.               |
|     |        |        |        |         | MOV       | R1,CSRADDR    |           | ;ADDRESS OF REGISTERS OF UNIT UNDER TEST |

TSV4 - MISCELLANEOUS SECTIONS MACRO M1113 06-FEB-84 17:14  
INITIALIZE SECTION

SEQ 089

```

113
114 022134 012001      MOV      (R0)+,R1      ;GET VECTOR ADDRESS.
115                  ;MOV      (R0),R2      ;GET INTERRUPT PRIORITY
116                  ;MOV      R2,IPRI      ;SET INTERRUPT PRIORITY.
117 022136 010137 002210  MOV      R1,IVEC      ;SET INTERRUPT VECTOR POINTER...
118 022142 012721 016216  MOV      @INTR,(R1)+    ;...VECTOR...
119 022146 013721 002212  MOV      IPRI,(R1)+    ;...AND PRIORITY.
120
121 022152              1$:
122                  ;      TST      QVP      ;1ST PASS ??
123                  ;      BEQ      5$      ;NO, SKIP THE PASS 1 STUFF.
124
125                  ;
126                  ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
127                  ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
128                  ;
129 022152 013701 002202      MOV      UNITN,R1
130 022156 006301      ASL      R1
131 022160 052761 100000 003176  BIS      @BIT15,ERTABL(R1)      ;SAY DEVICE RUNNING
132 022166 005037 005772      CLR      EXTA      ;CLEAR ERROR EXTENSION FLAG.
133 022172 023727 002012 000001  CMP      L$UNIT,@1      ;ARE WE TESTING MULTIPLE UNITS?
134 022200 101416      BLOS      10$      ;BR IF NO.
135 022202      RFLAGS      RO      ;YES -- GET OPERATOR FLAGS.
136 022202 104421      TRAP      C$RFLA
137 022204 032700 001000      BIT      @PNT,RO      ;SHOULD WE PRINT UNIT @?
138 022210 001412      BEQ      10$      ;BR IF NOT.
139 022212      PRINTF      @PUNIT,UNITN      ;PRINT THE UNIT @
140 022212 013746 002202      MOV      UNITN,-(SP)
141 022216 012746 022304      MOV      @PUNIT,-(SP)
142 022222 012746 000002      MOV      @2,-(SP)
143 022226 010600      MOV      SP,RO
144 022230 104417      TRAP      C$PNTF
145 022232 062706 000006      ADD      @6,SP
146 022236      10$:
147 022236 005037 003114      CLR      NODEV
148 022242 013701 002206      MOV      CSRADDR,R1      ;ADDRESS OF FIRST REGISTER
149 022246 010102      MOV      R1,R2      ;START OF REGISTERS
150 022250 062702 000002      ADD      @TSSR,R2      ;ADDRESS OF TSSR REGISTER
151 022254 004737 016376      JSR      PC,XNXM      ;TEST BOTH CONTROLLER REGISTERS...
152 022260 103005      BCC      2$      ;...AND BR IF ALL OK.
153 022262 010137 003114      MOV      R1,NODEV      ;FLAG DEVICE AS NON-EXISTENT
154 022266 012737 177777 003112  MOV      @-1,DUFLG      ;DROP THIS UNIT.
155 022274      2$:
156                  ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
157                  ;
158 022274      5$:      SETPRI      @PRI00      ;ENABLE INTERRUPTS.
159 022274 012700 000000      MOV      @PRI00,RO
160 022300 104441      TRAP      C$SPRI
161 022302      ENDINIT
162 022302      L10030:
163 022302 104411      TRAP      C$INIT
164
165 022304      045      116      045  PUNIT: .ASCIZ  /%NON%A***** TESTING UNIT %D2%A *****/
166                  .EVEN

```

TSV4 - MISCELLANEOUS SECTIONS  
ADD AND DROP UNITS SECTIONS

MACRO M1113 06-FEB-84 17:14

SEQ 090

```

158                                     .SEITL  ADD AND DROP UNITS SECTIONS
159
160
161                                     ;**
162                                     ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
163                                     ; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
164                                     ; OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
165                                     ;--
166 022352                                BGNAU
167 022352                                L$AU::
168 022352 010001                        MOV      R0,R1                ; GET UNIT TO BE ADDED (R0)
169 022354 006301                        ASL      R1                  ; MAKE IT A WORD INDEX
170 022356 052761 100000 003176          BIS      #100000,ERTABL(R1)    ; SET THE "ACTIVE" BIT
171 022364 042761 040000 003176          BIC      #40000,ERTABL(R1)    ; CLEAR THE "DROPPED" BIT
172 022372                                PRINTF  #1$,R0
173 022372 010046                        MOV      R0,-(SP)
174 022374 012746 022420                MOV      #1$,-(SP)
175 022400 012746 000002                MOV      #2,-(SP)
176 022404 010600                        MOV      SP,R0
177 022406 104417                        TRAP     C$PNTF
178 022410 062706 000006                ADD      #6,SP
179 022414                                EXIT     AU
180 022414 000167                        .WORD    J$JMP
181 022416 000020                        .WORD    L10031-2-.
182 022420 045 116 045 1$:              .ASCIZ  /NNA UNIT DDA ADDED/
183                                     .EVEN
184
185                                     ENDAU                                ; UNUSED.
186
187 022446                                L10031:
188 022446 104452                        TRAP     C$AU
189
190                                     ;**
191                                     ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
192                                     ; TO BE REMOVED FROM THE TEST LIST.
193
194                                     ;
195                                     ; SUPVSR DOES THE "DROPPING". THIS IS JUST TO TELL THE MAN,
196                                     ; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
197                                     ; COMMAND, OTHERWISE REMAIN INACTIVE. THE "DISPLAY" COMMAND
198                                     ; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
199                                     ; WHICH ARE STILL ACTIVE.
200                                     ; UPON ENTRY, R0 CONTAINS THE UNIT TO BE DROPPED.
201
202                                     BGNDU
203 022450                                L$DU::
204 022450 012737 177777 003112          MOV      # -1,DUFLG
205 022456 010001                        MOV      R0,R1
206 022460 006301                        ASL      R1
207 022462 052761 140000 003176          BIS      #140000,ERTABL(R1)    ; SAY DROPPED
208 022470 000240 000240 000240          240,240,240                ; ??????????
209 022476                                PRINTF  #1$,R0
210 022476 010046                        MOV      R0,-(SP)
211 022500 012746 022524                MOV      #1$,-(SP)
212 022504 012746 000002                MOV      #2,-(SP)
213 022510 010600                        MOV      SP,R0
214 022512 104417                        TRAP     C$PNTF
215 022514 062706 000006                ADD      #6,SP
216 022520                                EXIT     DU
217 022520 000167                        .WORD    J$JMP
218 022522 000030                        .WORD    L10032-2-.

```

N7

TSV4 . MISCELLANEOUS SECTIONS  
ADD AND DROP UNITS SECTIONS

MACRO M1113 06-FEB-84 17:14

SEQ 091

```

195 022524      045      116      045 1$: .ASCIZ /NNA UNIT DNA DROPPED/
196                                     .EVEN
197 022554                                     ENDDU
    022554                                     L10032:
    022554 104453                                     TRAP      C$DU
198                                     ;**
199                                     ; AUTO-DROP CODE SECTION.
200                                     ;--
201 022556                                     BGNAUTO
    022556                                     L$AUTO::
202 022556 013705 002206                                     MOV      CSRADDR,R5                                     ;POINT TO DEVICE REGISTER
203 022562 012703 000550                                     MOV      #360,R3                                     ;ENOUGH TIME FOR 2400' REEL TO REWIND
204 022566 004737 016250 10$: JSR      PC,WAITF                                     ;WAIT FOR SSR TO SET
205 022572 103420                                     BCS      20$                                     ;LEAVE WHEN SSR IS SET
206 022574                                     DELAY    250.                                     ;WAIT FOR .25 SECONDS
    022574 012727 000372                                     MOV      #250.,(PC)+
    022600 000000                                     .WORD    0
    022602 013727 002116                                     MOV      L$DLY,(PC)+
    022606 000000                                     .WORD    0
    022610 005367 177772                                     DEC      -6(PC)
    022614 001375                                     BNE      .-4
    022616 005367 177756                                     DEC      -22(PC)
    022622 001367                                     BNE      .-20
207 022624 005303                                     DEC      R3                                     ;BUMP COUNTER DOWN
208 022626 001357                                     BNE      10$                                     ;KEEP GOING
209 022630 004737 017202                                     JSR      PC,CKDROP                                     ;TRY AND DROP UNIT
210 022634                                     20$:
211 022634                                     ENDAUTO
    022634                                     L10033:
    022634 104461                                     TRAP      C$AUTO

```

; UNUSED.

TSV4 - MISCELLANEOUS SECTIONS MACRO M1113 06-FEB-84 17:14  
 CLEAN-UP AND REPORT CODING SECTIONS

SEQ 092

```

213                                     ,SBTTL  CLEAN-UP AND REPORT CODING SECTIONS
214
215
216                                     ***
217                                     ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
218                                     ; EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
219                                     ; USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
220                                     ***
220 022636                                BGNCLN
221 022636                                L$CLEAN:
222 022636 013705 002206                MOV     CSRADDR,R5          ;POINT TO DEVICE REGISTER
223 022642 005737 003112                TST     DUFLG              ;"DROPPED" FLAG IS SET ON...
224                                     BMI     1$                  ;...AND GROSS CONTROLLER FAULT...
225                                     ;...DON'T TRY TO XCT CLEANUP CODE.
226 022650 012765 000000 000002        MOV     #0,TSSR(R5)        ;DO SOFT INIT
227 022656 004737 016250                JSR     PC,WAITF
228 022662
229 022662                                1$:
230 022662                                2$:  ENDCLN
231                                     L10034:
232                                     TRAP     C$CLEAN
233                                     ***
234                                     ; THE REPORT CODING SECTION CONTAINS THE
235                                     ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
236                                     ***
236 022664                                BGNRPT
237 022664                                L$RPT:
238 022664                                PRINTS  #DEVSUM
239 022664 012746 023126                MOV     #DEVSUM, -(SP)
240 022670 012746 000001                MOV     #1, -(SP)
241 022674 010600                MOV     SP,R0
242 022676 104416                TRAP     C$PNTS
243 022700 062706 000004                ADD     #4,SP
244 022704 010246                MOV     R2, -(SP)
245 022706 010346                MOV     R3, -(SP)
246 022710 010446                MOV     R4, -(SP)
247 022712 012704 003176                MOV     #ERTABL,R4          ; GET START OF ERROR TABLE.
248 022716 005003                CLR     R5                      ; CLEAR UNIT NUMBER
249 022720 011402                1$:  MOV     (R4),R2          ; GET ERROR TABLE ENTRY & TEST IT.
250 022722 001467                BEQ     4$                      ; ZERO IF UNIT NOT RUN
251 022724 100066                BPL     4$
252 022726 032702 040000                DIT     #BIT14,R2          ; WAS UNIT DROPPED?
253 022732 001015                BNE     2$                      ; BR IF YES
254 022734 042702 170000                BIC     #C7777,R2          ; GET ERROR COUNT FIELD
255 022740                                PRINTS  #DEVONL,R3,R2    ; PRINT
256 022740 010246                MOV     R2, (SP)
257 022742 010346                MOV     R3, (SP)
258 022744 012746 023163                MOV     #DEVONL, -(SP)
259 022750 012746 000003                MOV     #7, -(SP)
260 022754 010600                MOV     SP,R0
261 022756 104416                TRAP     C$PNTS
262 022760 062706 000010                ADD     #4,SP
263 022764 000446                BR      1$
264 022766 020227 160000                2$:  CMP     R2,#160000        ; WAS UNIT NON-EXISTENT?
265 022772 001012                BNE     3$                      ; BR IF NO
266 022774                                PRINTS  #DEVNXR,R3
267 022776 010346                MOV     R3, (SP)
268 022776 012746 023233                MOV     #DEVNXR, -(SP)

```

C8

TSV4 MISCELLANEOUS SECTIONS MACRO M1113 06-FEB 84 17:14  
CLEAN-UP AND REPORT CODING SECTIONS

SEQ 093

```

023002 012746 000002      MOV    #2,-(SP)
023006 010600      MOV    SP,R0
023010 104416      TRAP   C#PNTS
023012 062706 000006      ADD    #6,SP
252 023016 000431      BR     4#
253 023020 020227 160001      3#:  CMP    R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
254 023024 001012      BNE    30#      ; BR IF NO.
255 023026      PRINTS  #DEVNRD,R3
      023026 010346      MOV    R3,-(SP)
      023030 012746 023315      MOV    #DEVNRD,-(SP)
      023034 012746 000002      MOV    #2,-(SP)
      023040 010600      MOV    SP,R0
      023042 104416      TRAP   C#PNTS
      023044 062706 000006      ADD    #6,SP
256 023050 000414      BR     4#
257 023052 042702 170000      30#: BIC    #C7777,R2
258 023056      PRINTS  #DEVDR0,R3,R2
      023056 010246      MOV    R2,-(SP)
      023060 010346      MOV    R3,-(SP)
      023062 012746 023376      MOV    #DEVDR0,-(SP)
      023066 012746 000003      MOV    #3,-(SP)
      023072 010600      MOV    SP,R0
      023074 104416      TRAP   C#PNTS
      023076 062706 000010      ADD    #10,SP
259 023102 062704 000002      4#:  ADD    #2,R4
260 023106 005203      INC    R3
261 023110 020427 003376      CMP    R4,#ERTABE
262 023114 103701      BLO    1#
263 023116 012604      MOV    (SP),R4
264 023120 012603      MOV    (SP),R3
265 023122 012602      MOV    (SP),R2
266 023124      ENDRPT      ; UNUSED.
      023124      L10035:
      023124 104425      TRAP   C#RPT
267
268 023126      045      116      045  DEVSUM: .ASCIZ  /UN#ADEVICE STATUS SUMMARY: #N/
269 023163      045      101      040  DEVONL: .ASCIZ  /#A UNIT #D3#A  ONLINE,  ERRORS = #D#N/
270 023233      045      101      040  DEVNXR: .ASCIZ  /#A UNIT #D3#A  DROPPED, NON-EXISTENT REGISTER#N/
271 023315      045      101      040  DEVNRD: .ASCIZ  /#A UNIT #D3#A  DROPPED, NOT READY AT STARTUP#N/
272 023376      045      101      040  DEVDR0: .ASCIZ  /#A UNIT #D3#A  DROPPED, ERRORS = #D#N/
273      .EVEN
274
275 023446      ENDMOD
276

```

D8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
CLEAN UP AND REPORT CODING SECTIONS

SEQ 094

1  
2  
9  
10 023446  
023446  
16  
24

.TITLE TSV5 - HARDWARE TESTS  
BGNMOD TSV5  
TSV5::

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 1: INITIALIZE AFTER WRITE CHARACTERISTICS

SEQ 095

```

26          .SBTTL TEST 1: INITIALIZE AFTER WRITE CHARACTERISTICS
27
28          ;* TEST DESCRIPTION:
29
30          ; This test verifies that a Hardware Initialize command
31          ; invoked after a Write Characteristics command sets up
32          ; the Command, Message and Characteristic image blocks
33          ; in the controller ram correctly.
34
35          ; TEST STEPS:
36
37          ; REPEAT FOR LOOPCNT
38          ; BEGIN
39          ; Do WRITE CHARACTERISTICS command.
40          ; If the NBA bit in the TSSR register is NOT=0 then Print Error.
41          ; Write to TSSR register to soft initialize the controller
42          ; If controller RAM 310-377 NOT=0 then Print Error
43          ; END
44          ;--
45
46          BGNTST
47 023446          T1::
023446          ;ASCII MESSAGE TO IDENTIFY TEST
53 023446 012700 024102 MOV  #TST13ID,R0 ;DO INITIAL TEST SETUP
54 023452 004737 016510 JSR  PC,TSTSETUP ;PERFORM 10 ITERATIONS
55 023456 012737 000012 002216 MOV  #10.,LOOPCNT
56 023464          T13LOOP:
57 023464 004737 024356 JSR  PC,T13REST ;SET PACKET TO START-UP VALUES
58
59 023470 012703 002764 MOV  #TSTBLK+10.,R3 ;START OF TEST DATA
60 023474 012704 024040 MOV  #T13PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
61 023500 012764 000010 000006 MOV  #8.,PKBCNT(R4) ;START WITH MINIMUM ALLOWABLE VALUE
62 023506          5$:
63 023506 004737 015774 JSR  PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
64 023512 103405 BCS  10$ ;BR IF SOFT INIT OKAY
65 023514 010001 MOV  R0,R1 ;SAVE CONTENTS OF TSSR
66 023516 ERRDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
    023516 104455 TRAP  C$ERRDF
    023520 000144 .WORD 100
    023522 003652 .WORD SFIERR
    023524 012034 .WORD SFIMSG
67
68          ;Do WRITE CHARACTERISTICS command.
69 023526 005037 002222 10$: CLR  FATFLG ;CLEAR FATAL ERROR FLAG
70 023532 010465 000000 MOV  R4,TSD8(R5) ;SET THE PACKET ADDRESS TO EXECUTE
71 023536 004737 016336 JSR  PC,CHKTSSR ;WAIT FOR SSR TO SET
72 023542          FORCERROR 12$ ;DO FORCE ERROR IF FORCER=1
73 023556 103407 BCS  15$ ;BR IF CARRY SET (GOOD RETURN)
74 023560 010001 MOV  R0,R1 ;SAVE CONTENTS OF TSSR
75 023562          NEXT.ERRNO
76 023562 12$: ERRDF ERRNO,T13SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
    023562 104455 TRAP  C$ERRDF
    023564 000145 .WORD 101
    023566 024267 .WORD T13SSR
    023570 012046 .WORD PKTSSR
77 023572 005237 002222 15$: INC  FATFLG ;SET FATAL ERROR FLAG
78 023576          CKLOOP ;LOOP ON ERROR, IF FLAG SET

```

F8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 1: INITIALIZE AFTER WRITE CHARACTERISTICS

SEQ 096

```

023576 104406
79 023600 016501 000002      MOV    TSSR(R5),R1      ;GET THE CONTENTS OF TSSR
80 023604 012702 000200      MOV    #SSR,R2        ;EXPECTED CONTENTS OF TSSR
81 023610 032701 000100      BIT    #OFL,R1        ;IS OFF-LINE BIT SET ?
82 023614 001402              BEQ     25$      ;BRANCH IF NOT OFF-LINE
83 023616 052702 000100      BIS     #OFL,R2        ;SET OFF-LINE IN EXPECTED DATA
84
85      ;If the NBA bit in the TSSR register is NOT=0 then Print Error.
25$:      FORCERROR      27$      ;000
86 023622              CMP     R2,R1        ;DOES EXPECTED MATCH RECEIVED ?
87 023622              BEQ     30$      ;OKAY IF MATCH
88 023636 020201              NEXT.ERRNO
89 023640 001404      27$:      ERRHRD  ERRNO,T13NBA,PKTSSR      ;NBA NOT ZERO
90 023642              TRAP     C$ERHRD
91 023642              .WORD   102
92 023642 104456              .WORD   T13NBA
93 023644 000146              .WORD   PKTSSR
94 023646 024214
95 023650 012046
96 023652              30$:      CKLOOP      ;LOOP ON ERROR ?
97 023652 104406              TRAP     C$CLP1
98
99      ;Write to TSSR register to soft initialize the controller
40$:      JSR     PC,SOFINIT      ;WRITE TO TSSR TO SOFT INITIALIZE
100 023654 004737 015774      FORCERROR      42$      ;000
101 023660              BCS     50$      ;BR IF SOFT INIT OKAY
102 023674 103405              MOV     R0,R1      ;SAVE CONTENTS OF TSSR
103 023676 010001              NEXT.ERRNO
104 023700              42$:      ERRDF  ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL DURING INIT
105 023700 104455              TRAP     C$ERDF
106 023702 000147              .WORD   103
107 023704 003652              .WORD   SFIERR
108 023706 012034              .WORD   SFIMSG
109
110      ;If controller RAM 310-377 NOT=0 then Print Error
50$:      MOV     #310,R4      ;START WITH LOC 310
111 023710 012704 000310      CLR     R2        ;MEMORY EXPECTED SHOULD BE 000000
112 023714 005002              CLRB    TSOB(R5)      ;SET MAINTENANCE MODE
113 023716 105065 000000      JSR     PC,CHK.SSR      ;WAIT FOR SSR READY
114 023722 004737 016336      60$:      MOV     R4,TSOB(R5)      ;SELECT RAM ADDRESS
115 023726 010465 000000      JSR     PC,CHKTSSR      ;WAIT FOR SSR READY
116 023732 004737 016336      MOVB    TSBA(R5),R1      ;READ LOC CONTENTS
117 023736 116501 000000      FORCERROR      62$,NOTSSR      ;000
118 023742              CMPB    R1,R2        ;CHECK MEMORY FOR 000000
119 023752 120102              BEQ     70$      ;BRANCH IF DATA OKAY
120 023754 001406              NEXT.ERRNO
121 023756              62$:      ERRDF  ERRNO,T13MEM,RAMEXP      ;MEMORY NOT ZERO AFTER INIT.
122 023756 104455              TRAP     C$ERDF
123 023760 000150              .WORD   104
124 023762 024155              .WORD   T13MEM
125 023764 015530              .WORD   RAMEXP
126 023766 005237 002222      70$:      INC     FATFLG      ;SET THE FATAL ERROR FLAG
127 023772              CKLOOP
128 023772 104406              TRAP     C$CLP1
129 023774              ESCAPE  TST      ;EXIT ON FATAL ERROR
130 023774 104410              TRAP     C$ESCAPE
131 023776 000426              .WORD   L10036-

```

G8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 1: INITIALIZE AFTER WRITE CHARACTERISTICS

SEQ 097

```

119
120 024000 005204      82$: INC R4      ;LOOK AT NEXT RAM LOC.
121 024002 020427 000400  CMP R4,#400 ;AT TOP OF RAM ADDRESS SPACE
122 024006 001347      BNE 60$      ;BRANCH TILL ALL MEMORY TESTED
123
124
125 024010 005737 002222      TST FATFLG ;ANY FATAL ERRORS ?
126 024014 001402      BEQ 160$      ;BRANCH IF NOT
127 024016 004737 017202      JSR PC,CKDROP ;TRY TO DROP THE UNIT
128 024022 004737 016456 160$: JSR PC,TSTLOOP ;DONE ALL ITERATIONS?
129 024026 103002      BCC 165$      ;BR IF YES
130 024030 000137 023464      JMP T13LOOP ;LOOP UNTIL ITERATION COUNT DONE
131 024034
132 024034      EXIT TST
    024034 104432
    024036 000366
                                     TRAP C$EXIT
                                     .WORD L10036-.

133
134
135
136      ;*
137      ;LOCAL STORAGE FOR THIS TEST
138      ;-
142 024040      T13PACKET:
143 024040 100004      .WORD 100004      ;COMMAND PACKET FOR TEST
144 024042 024050      .WORD T13DATA      ;WRITE CHARACTERISTICS COMMAND, WITH ACK
145 024044 000000      .WORD 0          ;ADDRESS OF CHARACTERISTICS BLOCK
146 024046 000010      .WORD 8.          ;STARTING VALUE OF BLOCK SIZE
147
148 024050      T13DATA:
149 024050 024062      .WORD T13BFR      ;CHARACTERISTICS DATA BLOCK
150 024052 000000      .WORD 0          ;ADDRESS OF MESSAGE BUFFER
151 024054 000016      .WORD 14.         ;LENGTH OF MESSAGE BUFFER
152 024056 000000 000000      .WORD 0,0
153
154 024062      T13BFR: .BLKW 8.          ;MESSAGE BUFFER
155
156      ;LOCAL TEXT MESSAGES FOR TEST
157      ;-
158 024102      111      156      151 TST13ID: .ASCIIZ 'Initialization After WRITE CHARACTERISTICS'
159 024155      111      156      143 T13MEM: .ASCIIZ 'Incorrect RAM Data After Init'
160
161 024214      127      122      111 T13NBA: .ASCIIZ 'WRITE CHARACTERISTICS Command Not Accepted'
162 024267      103      157      156 T13SSR: .ASCIIZ 'Contents of TSSR Incorrect After WRITE CHARACTERISTICS'
163
164
165
166
167      ;*
168      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
169      ;-
170
171      .EVEN
172
173 024356      T13REST:
174 024356      SAVREG      ;SAVE THE REGISTERS
175 024362 012701 024040      MOV #T13PACKET,R1 ;START OF THE PACKET
176 024366 012721 100004      MOV #100004,(R1)+ ;WRITE CHARACTERISTICS WITH ACK

```

|     |        |        |        |        |                                                                     |                                     |         |           |
|-----|--------|--------|--------|--------|---------------------------------------------------------------------|-------------------------------------|---------|-----------|
|     |        |        |        | MOV    | #T13DATA,(R1)+                                                      | ;ADDRESS OF CHAR DATA BLOCK         |         |           |
| 178 | 024376 | 005021 | 024050 | CLR    | (R1)+                                                               | ;EXTENDED ADDRESS                   |         |           |
| 179 | 024400 | 012721 | 000010 | MOV    | #8.,(R1)+                                                           | ;SIZE OF DATA BLOCK IN BYTES        |         |           |
| 180 | 024404 | 012721 | 024062 | MOV    | #T13BFR,(R1)+                                                       | ;ADDRESS OF MESSAGE BUFFER          |         |           |
| 181 | 024410 | 005021 |        | CLR    | (R1)+                                                               |                                     |         |           |
| 182 | 024412 | 012721 | 000016 | MOV    | #14.,(R1)+                                                          | ;LENGTH OF MESSAGE BUFFER           |         |           |
| 183 | 024416 | 005021 |        | CLR    | (R1)+                                                               |                                     |         |           |
| 184 | 024420 | 005011 |        | CLR    | (R1)                                                                |                                     |         |           |
| 185 | 024422 | 000207 |        | RTS    | PC                                                                  | ;RETURN                             |         |           |
| 186 | 024424 |        |        | ENDTST |                                                                     |                                     |         |           |
|     | 024424 |        |        |        |                                                                     |                                     | L10036: |           |
|     | 024424 | 104401 |        |        |                                                                     |                                     | TRAP    | C\$ETST   |
| 187 |        |        |        |        | .SBTTL TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND                 |                                     |         |           |
| 188 |        |        |        |        | :                                                                   |                                     |         |           |
| 189 |        |        |        |        | : THIS TEST VERIFIES THAT THE WRITE SUBSYSTEM MEMORY COMMAND WITH A |                                     |         |           |
| 190 |        |        |        |        | : BSELO SELECT CODE OF 0 (NO-OP) EXECUTES CORRECTLY. IT ALSO        |                                     |         |           |
| 191 |        |        |        |        | : VERIFIES THAT A WRITE SUBSYSTEM MEMORY COMMAND WITH A NON-ZERO    |                                     |         |           |
| 192 |        |        |        |        | : MODE FIELD IS REJECTED. THE TEST FURTHER VERIFIES MICROPROGRAM    |                                     |         |           |
| 193 |        |        |        |        | : COMMAND DECODING AND HANDLING SEQUENCES.                          |                                     |         |           |
| 194 |        |        |        |        | :                                                                   |                                     |         |           |
| 195 |        |        |        |        | :-                                                                  |                                     |         |           |
| 196 |        |        |        |        | BGNTST                                                              |                                     |         |           |
| 197 |        |        |        |        |                                                                     | T2::                                |         |           |
| 198 | 024426 |        |        |        |                                                                     | ;ASCII MESSAGE TO IDENTIFY TEST     |         |           |
|     | 024426 |        |        |        |                                                                     | ;DO INITIAL TEST SETUP              |         |           |
| 203 | 024426 | 012700 | 026241 | MOV    | #T14ID,R0                                                           | ;PERFORM 20 ITERATIONS              |         |           |
| 204 | 024432 | 004737 | 016510 | JSR    | PC,T14SETUP                                                         |                                     |         |           |
| 205 | 024436 | 012737 | 000024 | MOV    | #20.,LOOPCNT                                                        |                                     |         |           |
| 206 | 024444 |        |        |        |                                                                     | /***** BEGIN SUBTEST *****/         |         |           |
| 207 | 024444 |        |        |        |                                                                     | T2.1:                               |         |           |
|     | 024444 | 104402 |        |        |                                                                     | TRAP                                |         | C\$BSUB   |
| 208 | 024446 | 004737 | 026306 | JSR    | PC,T14REST                                                          | ;SET PACKET TO INITIAL VALUES       |         |           |
| 209 | 024452 | 004737 | 026344 | JSR    | PC,T14RST                                                           | ;SET PACKET TO INITIAL VALUES       |         |           |
| 210 | 024456 |        |        | SETPRI | #PRI00                                                              | ;LOWER PRIORITY TO ALLOW INTERRUPTS |         |           |
|     | 024456 | 012700 | 000000 |        |                                                                     | MOV                                 |         | #PRI00,R0 |
|     | 024462 | 104441 |        |        |                                                                     | TRAP                                |         | C\$SPRI   |
| 211 | 024464 |        |        | 5\$:   |                                                                     |                                     |         |           |
| 212 | 024464 |        |        |        | BGNSEG                                                              | >>>>>>>> BEGIN SEGMENT >>>>>>>>     |         |           |
|     | 024464 | 104404 |        |        |                                                                     | TRAP                                |         | C\$BSEG   |
| 213 | 024466 | 004737 | 015774 | JSR    | PC,SOFINIT                                                          | ;DO SOFT INIT OF CONTROLLER         |         |           |
| 214 | 024472 | 103405 |        | BCS    | 10\$                                                                | ;BR IF SOFT INIT = OK               |         |           |
| 218 | 024474 | 010001 |        | MOV    | R0,R1                                                               | ;SAVE CONTENTS OF TSSR              |         |           |
| 219 | 024476 |        |        | ERRDF  | ERRNO,SFIERR,SFIMSG                                                 | DEVICE FATAL ERROR DURING INIT      |         |           |
|     | 024476 | 104455 |        |        |                                                                     | TRAP                                |         | C\$ERDF   |
|     | 024500 | 000311 |        |        |                                                                     | .WORD                               |         | 201       |
|     | 024502 | 003652 |        |        |                                                                     | .WORD                               |         | SFIERR    |
|     | 024504 | 012034 |        |        |                                                                     | .WORD                               |         | SFIMSG    |
| 220 | 024506 |        |        | 10\$:  |                                                                     |                                     |         |           |
| 221 | 024506 | 012704 | 025600 | MOV    | #T14PK2,R4                                                          | ;SUBROUTINE NEEDS PACKET ADDRESS    |         |           |
| 222 | 024512 | 004737 | 010662 | JSR    | PC,WRTCHR                                                           | ;ISSUE WRITE CHARACTERISTICS        |         |           |
| 223 | 024516 | 103405 |        | BCS    | 11\$                                                                | ;BR, IF COMMAND ISSUED OK           |         |           |
| 227 | 024520 | 010001 |        | MOV    | R0,R1                                                               | ;SAVE CONTENTS OF TSSR              |         |           |
| 228 | 024522 |        |        | ERRHRD | ERRNO,WRTMSG,SFIMSG                                                 | ;WRITE CHARACTERISTICSC FAILED      |         |           |
|     | 024522 | 104456 |        |        |                                                                     | TRAP                                |         | C\$ERRHD  |
|     | 024524 | 000312 |        |        |                                                                     | .WORD                               |         | POL       |
|     | 024526 | 005056 |        |        |                                                                     | .WORD                               |         | WRTMSG    |

| Address | Hex    | Hex    | Hex    | Label  | Assembly             | Comments                             | Trap  | Word      |
|---------|--------|--------|--------|--------|----------------------|--------------------------------------|-------|-----------|
| 024530  | 012034 |        |        |        |                      |                                      |       | SFIMSG    |
| 229     | 024532 |        | 11\$:  |        |                      |                                      |       |           |
| 230     | 024532 | 005037 | 002222 | CLR    | FATFLG               | ;CLEAR FATAL ERROR FLAG              |       |           |
| 231     | 024536 | 005037 | 002224 | CLR    | INTRECV              | ;CLEAR INTERRUPT RECEIVED FLAG       |       |           |
| 232     | 024542 | 012704 | 025160 | MOV    | #T14PACKET,R4        | ;SET UP NEW WRT. SUBSYS MEM. COMMAND |       |           |
| 233     | 024546 | 010465 | 000000 | MOV    | R4,TSD8(R5)          | ;SET THE PACKET ADDRESS              |       |           |
| 234     | 024552 | 004737 | 016336 | JSR    | PC,CHKTSSR           | ;WAIT FOR SSR TO SET                 |       |           |
| 235     | 024556 | 103407 |        | BCS    | 15\$                 | ;BR IF CARRY SET (GOOD RETURN)       |       |           |
| 236     | 024560 | 010001 |        | MOV    | R0,R1                | ;SAVE CONTENTS OF TSSR               |       |           |
| 240     | 024562 |        |        | ERRDF  | ERRNO,T14SSR,PKTSSR  | ;DEVICE FATAL SSR FAILED TO SET      |       |           |
|         | 024562 | 104455 |        |        |                      |                                      | TRAP  | C\$ERDF   |
|         | 024564 | 000313 |        |        |                      |                                      | .WORD | 203       |
|         | 024566 | 025773 |        |        |                      |                                      | .WORD | T14SSR    |
|         | 024570 | 012046 |        |        |                      |                                      | .WORD | PKTSSR    |
| 241     | 024572 | 005237 | 002222 | INC    | FATFLG               | ;SET FATAL ERROR FLAG                |       |           |
| 242     | 024576 |        | 15\$:  | CKLOOP |                      | ;LOOP ON ERROR, IF FLAG SET          |       |           |
|         | 024576 | 104406 |        |        |                      |                                      | TRAP  | C\$CLP1   |
| 243     | 024600 |        |        | ESCAPE | SEG                  | ;BY-PASS SUBTEST IF FATAL ERROR      |       |           |
|         | 024600 | 104410 |        |        |                      |                                      | TRAP  | C\$ESCAPE |
|         | 024602 | 000074 |        |        |                      |                                      | .WORD | 10000\$-. |
| 244     | 024604 | 005737 | 002224 | TST    | INTRECV              | ;DID AN INTERRUPT OCCUR ?            |       |           |
| 245     | 024610 | 001004 |        | BNE    | 22\$                 | ;BRANCH IF YES                       |       |           |
| 249     | 024612 |        |        | ERRHRD | ERRNO,T14NINT,PKTSSR |                                      |       |           |
|         | 024612 | 104456 |        |        |                      |                                      | TRAP  | C\$ERHRD  |
|         | 024614 | 000314 |        |        |                      |                                      | .WORD | 204       |
|         | 024616 | 026063 |        |        |                      |                                      | .WORD | T14NINT   |
|         | 024620 | 012046 |        |        |                      |                                      | .WORD | PKTSSR    |
| 250     | 024622 | 016501 | 000002 | MOV    | TSSR(R5),R1          | ;GET THE CONTENTS OF TSSR            |       |           |
| 251     | 024626 | 012702 | 000200 | MOV    | #SSR,R2              | ;EXPECTED CONTENTS OF TSSR           |       |           |
| 252     | 024632 | 032701 | 000100 | BIT    | #OFL,R1              | ;IS OFF-LINE BIT SET ?               |       |           |
| 253     | 024636 | 001402 |        | BEQ    | 25\$                 | ;BRANCH IF NOT OFF-LINE              |       |           |
| 254     | 024640 | 052702 | 000100 | BIS    | #OFL,R2              | ;SET OFF-LINE IN EXPECTED DATA       |       |           |
| 255     | 024644 | 020201 |        | CMP    | R2,R1                | ;DOES EXPECTED MATCH RECEIVED ?      |       |           |
| 256     | 024646 | 001404 |        | BEQ    | 30\$                 | ;OKAY IF MATCH                       |       |           |
| 260     | 024650 |        |        | ERRHRD | ERRNO,T14NBA,PKTSSR  | ;NBA NOT ZERO                        |       |           |
|         | 024650 | 104456 |        |        |                      |                                      | TRAP  | C\$ERHRD  |
|         | 024652 | 000315 |        |        |                      |                                      | .WORD | 205       |
|         | 024654 | 025622 |        |        |                      |                                      | .WORD | T14NBA    |
|         | 024656 | 012046 |        |        |                      |                                      | .WORD | PKTSSR    |
| 261     | 024660 |        | 30\$:  |        |                      |                                      |       |           |
| 262     | 024660 | 004737 | 011114 | 35\$:  | JSR                  | PC,CKRAM                             |       |           |
| 263     | 024664 | 103405 |        | BCS    | 59\$                 | ;CHECK RAM TO MEMORY                 |       |           |
| 267     | 024666 |        |        | ERRHRD | ERRNO,PKTRAM,RAMERR  | ;RAM OK GO ON                        |       |           |
|         | 024666 | 104456 |        |        |                      | ;THEY DON'T MATCH                    |       |           |
|         | 024670 | 000316 |        |        |                      |                                      | TRAP  | C\$ERHRD  |
|         | 024672 | 004745 |        |        |                      |                                      | .WORD | 206       |
|         | 024674 | 015510 |        |        |                      |                                      | .WORD | PKTRAM    |
|         |        |        |        |        |                      |                                      | .WORD | RAMERR    |
| 268     | 024676 |        |        | ENDSEG |                      | ;***** END SEGMENT *****             |       |           |
|         | 024676 |        |        |        |                      | 10000\$:                             |       |           |
|         | 024676 | 104405 |        |        |                      |                                      |       |           |

| Address | Hex | Label | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416 | Op417 |  |
|---------|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|
|---------|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|

K8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND

SEQ 101

|     |        |        |        |            |                      |                                |       |           |
|-----|--------|--------|--------|------------|----------------------|--------------------------------|-------|-----------|
| 322 | 025044 |        | ESCAPE | SEG        |                      | ;BY-PASS CHECKS IF FATAL ERROR |       |           |
|     | 025044 | 104410 |        |            |                      |                                | TRAP  | C\$ESCAPE |
|     | 025046 | 000074 |        |            |                      |                                | .WORD | 10000\$-. |
| 323 | 025050 | 005737 | 002224 | TST        | INTRECV              | ;DID AN INTERRUPT OCCUR ?      |       |           |
| 324 | 025054 | 001004 |        | BNE        | 22\$                 | ;BRANCH IF YES                 |       |           |
| 328 | 025056 |        |        | ERRHRD     | ERRNO,T14NINT,PKTSSR |                                |       |           |
|     | 025056 | 104456 |        |            |                      |                                | TRAP  | C\$ERHRD  |
|     | 025060 | 000322 |        |            |                      |                                | .WORD | 210       |
|     | 025062 | 026063 |        |            |                      |                                | .WORD | T14NINT   |
|     | 025064 | 012046 |        |            |                      |                                | .WORD | PKTSSR    |
| 329 | 025066 | 016501 | 000002 | 22\$:      | MOV                  | TSSR(R5),R1                    |       |           |
| 330 | 025072 | 012702 | 100206 |            | MOV                  | #SC!SSR!TSREJ,R2               |       |           |
| 331 | 025076 | 032701 | 000100 |            | BIT                  | #OFL,R1                        |       |           |
| 332 | 025102 | 001402 |        |            | BEQ                  | 25\$                           |       |           |
| 333 | 025104 | 052702 | 000100 |            | BIS                  | #OFL,R2                        |       |           |
| 334 | 025110 | 020201 |        | 25\$:      | CMP                  | R2,R1                          |       |           |
| 335 | 025112 | 001404 |        |            | BEQ                  | 30\$                           |       |           |
| 339 | 025114 |        |        | ERRHRD     | ERRNO,T142REJ,PKTSSR | ;COMMAND NOT REJECTED          |       |           |
|     | 025114 | 104456 |        |            |                      |                                | TRAP  | C\$ERHRD  |
|     | 025116 | 000323 |        |            |                      |                                | .WORD | 211       |
|     | 025120 | 025676 |        |            |                      |                                | .WORD | T142REJ   |
|     | 025122 | 012046 |        |            |                      |                                | .WORD | PKTSSR    |
| 340 | 025124 |        |        | 30\$:      |                      |                                |       |           |
| 341 | 025124 | 004737 | 011114 | 35\$:      | JSR                  | PC,CKRAM                       |       |           |
| 342 | 025130 | 103405 |        |            | BCS                  | 59\$                           |       |           |
| 346 | 025132 |        |        | ERRHRD     | ERRNO,PKTRAM,RAMERR  | ;THEY DON'T MATCH              |       |           |
|     | 025132 | 104456 |        |            |                      |                                | TRAP  | C\$ERHRD  |
|     | 025134 | 000324 |        |            |                      |                                | .WORD | 212       |
|     | 025136 | 004745 |        |            |                      |                                | .WORD | PKTRAM    |
|     | 025140 | 015510 |        |            |                      |                                | .WORD | RAMERR    |
| 347 | 025142 |        |        | ENDSEG     |                      | ;***** END SEGMENT *****       |       |           |
|     | 025142 |        |        |            |                      | 10000\$:                       |       |           |
|     | 025142 | 104405 |        |            |                      |                                | TRAP  | C\$ESEG   |
| 348 |        |        |        |            |                      |                                |       |           |
| 349 | 025144 |        |        | 59\$:      | ENDSUB               | ;***** END SUBTEST *****       |       |           |
|     | 025144 |        |        |            |                      | L10041:                        |       |           |
|     | 025144 | 104403 |        |            |                      |                                | TRAP  | C\$ESUB   |
| 350 | 025146 |        |        | EXIT       | TST                  | ;ALL DONE THIS TEST            |       |           |
|     | 025146 | 104432 |        |            |                      |                                | TRAP  | C\$EXIT   |
|     | 025150 | 001246 |        |            |                      |                                | .WORD | L10037-.  |
| 351 |        |        |        |            |                      |                                |       |           |
| 352 |        |        |        |            |                      |                                |       |           |
| 353 |        |        |        |            |                      |                                |       |           |
| 354 |        |        |        |            |                      |                                |       |           |
| 355 |        |        |        |            |                      |                                |       |           |
| 357 |        | 025160 |        |            |                      |                                |       |           |
| 359 | 025160 |        |        | T14PACKET: | *-<.+10>E177770      | ;COMMAND PACKET FOR TEST       |       |           |
| 360 | 025160 | 100206 |        |            | .WORD                | 100206                         |       |           |
| 361 | 025162 | 025170 |        |            | .WORD                | T14DATA                        |       |           |
| 362 | 025164 | 000000 |        |            | .WORD                | 0                              |       |           |
| 363 | 025166 | 000006 |        |            | .WORD                | 6.                             |       |           |
| 364 | 025170 |        |        | T14DATA:   |                      | ;STARTING VALUE OF BLOCK SIZE  |       |           |
| 365 | 025170 | 000    |        | T14BS0:    | .BYTE                | 0                              |       |           |
| 366 | 025171 | 000    |        | T14BS1:    | .BYTE                | 0                              |       |           |
| 367 | 025172 | 000000 |        | T14BS2:    | .WORD                | 0                              |       |           |
| 368 | 025174 | 000000 |        |            | .WORD                | 0                              |       |           |
| 369 | 025176 |        |        | T14BFR:    | .BLKW                | 128.                           |       |           |
|     |        |        |        |            |                      | ;MESSAGE BUFFER                |       |           |

L8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND

SEQ 102

```

370
371
373      025600
375 025600      025600
376 025600      100204
377 025602      025610
378 025604      000000
379 025606      000010
380
381
382 025610
383 025610      025176
384 025612      000000
385 025614      000400
386 025616      000000 000000
387
388
389
390
391
392
393 025622      127      122      111 T14NBA: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted'
394 025676      127      122      111 T142REJ: .ASCIZ 'WRITE SUBSYSTEM MEMORY Not Rejected With Non-Zero Mode Field'
395 025773      103      157      156 T14SSR: .ASCIZ 'Contents of TSSR Incorrect After WRITE SUBSYSTEM MEMORY'
396 026063      105      170      160 T14NINT: .ASCIZ 'Expected Interrupt Not Received On WRITE SUBSYSTEM MEMORY'
397 026155      111      156      143 T14TSBA: .ASCIZ 'Incorrect TSBA Address After WRITE SUBSYSTEM MEMORY'
398 026241      102      141      163 TST14ID: .ASCIZ 'Basic WRITE SUBSYSTEM MEMORY Command'
399
400
401
402
403
404
405
406
407
408
409 026306
410 026306
411 026312      012701      025160
412 026316      012721      100206
413 026322      012721      025170
414 026326      005021
415 026330      012721      000006
416 026334      005021
417 026336      005021
418 026340      005011
419 026342      000207
420
421
422 026344
423 026344
424 026350      012701      025600
425 026354      012721      100204
426 026360      012721      025610
427 026364      005021
428 026366      012721      000010

      .=<.,+10>E177770
T14PK2: .WORD 100204
        .WORD T14DTA
        .WORD 0
        .WORD 8.
;COMMAND PACKET FOR TEST
;WRITE CHARA. MEM. CMND., WITH IE, ACK
;ADDRESS OF SELECT DATA BLOCK
;STARTING VALUE OF BLOCK SIZE

T14DTA: .WORD T14BFR
        .WORD 0
        .WORD 256.
        .WORD 0.0
;SELECT DATA BLOCK
;ADDRESS OF MESSAGE BUFFER
;LENGTH OF MESSAGE BUFFER

;+
;LOCAL TEXT MESSAGES FOR TEST
;-

111 T14NBA: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted'
111 T142REJ: .ASCIZ 'WRITE SUBSYSTEM MEMORY Not Rejected With Non-Zero Mode Field'
156 T14SSR: .ASCIZ 'Contents of TSSR Incorrect After WRITE SUBSYSTEM MEMORY'
160 T14NINT: .ASCIZ 'Expected Interrupt Not Received On WRITE SUBSYSTEM MEMORY'
143 T14TSBA: .ASCIZ 'Incorrect TSBA Address After WRITE SUBSYSTEM MEMORY'
163 TST14ID: .ASCIZ 'Basic WRITE SUBSYSTEM MEMORY Command'
      .EVEN

;+
;
;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
;WRITE SUBSYSTEM MEMORY COMMAND
;
;-

T14REST:
      SAVREG
      MOV     @T14PACKET,R1
      MOV     @100206,(R1)+
      MOV     @T14DATA,(R1)+
      CLR     (R1)+
      MOV     @6.,(R1)+
      CLR     (R1)+
      CLR     (R1)+
      CLR     (R1)
      RTS     PC
;SAVE THE REGISTERS
;START OF THE PACKET
;WRITE SUBSYSTEM MEM. WITH ACK, IE
;ADDRESS OF DATA BLOCK
;EXTENDED ADDRESS
;SIZE OF DATA BLOCK IN BYTES
;CLEAR BSEL0 AND BSEL1
;CLEAR SEL2
;CLEAR DATA AREA
;RETURN

T14RST:
      SAVREG
      MOV     @T14PK2,R1
      MOV     @100204,(R1)+
      MOV     @T14DTA,(R1)+
      CLR     (R1)+
      MOV     @8.,(R1)+
;SAVE THE REGISTERS
;START OF THE PACKET
;WRITE CHARA. WITH ACK, IE
;ADDRESS OF CHARACTERISTICS DATA BLOCK
;EXTENDED ADDRESS
;SIZE OF DATA BLOCK IN BYTES

```

M8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND

SEQ 103

429 026372 012721 025176  
430 026376 005021  
431 026400 012721 000400  
432 026404 005021  
433 026406 005011  
434 026410 005037 025176  
435 026414 000207  
436 026416  
026416  
026416 104401

MOV T14BFR,(R1)+  
CLR (R1)+  
MOV #256.,(R1)+  
CLR (R1)+  
CLR (R1)  
CLR T14BFR  
RTS PC  
ENDTST

;MESSAGE BUFFER ADDRESS  
;LENGTH OF MESSAGE BUFFER  
;CLEAR 1ST LOC IN MESSAGE BUFFER  
;RETURN

L10037: TRAP C\$ETST

N8

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 2: BASIC WRITE SUBSYSTEM MEMORY COMMAND

SEQ 104

```

438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465 026420
      026420
470 026420 012700 030470
471 026424 004737 016510
472 026430 012737 000012
473 026436 005237 003150
474 026442 004737 021276
475
476 026446
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497

      .SBTTL TEST 3: DMA MEMORY ADDRESSING

      **
      : TEST 3
      :
      : TEST DESCRIPTION
      :
      : This test verifies that the controller can properly address and
      : access all available CPU memory (other than that occupied by the
      : diagnostic and diagnostic supervisor code) for both reading (DATI)
      : and writing (DATO). Verified are the LSI-11 Bus drivers for all
      : available address lines. Up to this point only 16 bits have been
      : used for DMA transfers.
      :
      : TEST STEPS
      :
      : REPEAT FROM 1 TO LOOPCNT
      : BEGIN
      :   Do Subtest 1 - Verify GET STATUS selected locations
      :   Do Subtest 2 - Verify message packets selected locations
      :   Do Subtest 3 - Verify Characteristic data selected locations
      :   Do Subtest 4 - Verify NXM to selected invalid addresses
      : END
      :
      : --

      BGNTST
      T3::
      MOV    #TST12ID,R0      ;ASCII MESSAGE TO IDENTIFY TEST
      JSR    PC,TSTSETUP      ;DO INITIAL TEST SETUP
      MOV    #10,,LOOPCNT     ;PERFORM 10 ITERATIONS
      INC    T3BFLG           ;SET TEST FLAG
      JSR    PC,MEMCHK        ;CHECK MEMORY

      T12LOOP:                ;LOOP ON TEST LABEL.

      .SBTTL TEST 3: SUBTEST 1: GET STATUS SELECTED LOCATIONS

      **
      : TEST 3: SUBTEST 1:
      :
      : SUBTEST DESCRIPTION:
      :
      : This subtest verifies the controller can fetch a get status
      : command from all available memory locations.
      : Two word blocks are tested one at a time by first setting
      : all available memory to a background pattern of 125252.
      : A Get Status command is then executed to various addresses in
      : each available memory 4k word block. The various addresses
      : are determined by floating a 1 then a 0 through the address bits.
      :
      : TEST STEPS:
      :
      : BEGIN
      :   Write to TSSR to soft initialize
      :   Do a WRITE CHARACTERISTICS to setup a message buffer
      :

```

TSV5 HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 3: SUBTEST 1: GET STATUS SELECTED LOCATIONS

SEQ 105

```

498      REPEAT FOR SELECTED VALID ADDRESSES IN DIAGNOSTIC FREE SPACE AND ABOVE 32K
499      BEGIN
500      Get a valid modulo-4 test address
501      Do a GET STATUS command from the test address
502      END
503      END
504
505
506      BGNSUB                                // BEGIN SUBTEST //
                                           T3.1:
                                           TRAP    C1BSUB
507
508
509      ;Write to TSSR to soft initialize
510      JSR    PC,SOFINIT                    ;DO SOFT INIT OF CONTROLLER
511      BCS    15:                          ;BR IF SOFT INIT = OK
512      NEXT,ERRNO
513      MOV     R0,R1
514      ERDF    ERRNO,SFIERR,SFIMSG          ;SAVE CONTENTS OF TSSR
                                           ;DEVICE FATAL ERROR DURING INIT
                                           TRAP    C1ERDF
                                           .WORD    301
                                           .WORD    SFIERR
                                           .WORD    SFIMSG
515
516      ;Do a WRITE CHARACTERISTICS to setup a message buffer
517      15:
518      MOV     @T12PACKET,R4                ;GET THE ADDRESS OF COMMAND PACKET
519      JSR     PC,T12SWRT                  ;RESTORE PACKET TO STARTING VALUES
520      CLR     KTENABLE                    ;TURN OFF KT-11
521      MOV     R4,TSD8(R5)                 ;SET THE PACKET ADDRESS
522      JSR     PC,CHKTSSR                  ;WAIT FOR SSR TO SET
523      FORCERROR 17:
524      BCS     20:                          ;BR IF SSR SET IN CHKTSSR
525      MOV     R0,R1
526      NEXT,ERRNO
527      17:
528      ERDF    ERRNO,T12WRTSSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                                           TRAP    C1ERDF
                                           .WORD    302
                                           .WORD    T12WRTSSR
                                           .WORD    PKTSSR
529
530      ;Verify a Get Status can be fetched from each address
531      ;Get a valid modulo-4 test address
532      ;Do a GET STATUS command from the test address
533      20:
534      CLR     FATELG                      ;CLEAR FATAL ERROR FLAG
535      CLR     T12KT                       ;TEST ABOVE 28K SWITCH
536      MOV     @T12BLK,R2                  ;POINT TO TEST PATTERN TABLE
537      T121LOOP:
538      CLR     KTENABLE                    ;TURN OFF ABOVE 28K TEST FLAG
539      MOV     (R2),R1                     ;GET TEST PATTERN ADDRESS
540      CLR     R0                          ;ASSUME NO TEST ABOVE 28K
541      TST     T12KT                       ;TEST ABOVE 28K THIS TIME?
542      BEQ     25:                          ;BR IF NO
543      MOV     -2(R2),R0                    ;GET TEST PATTERN AGAIN
544      BIC     @C<A1716>,R0                ;SAVE 18 BIT ADDRESS ON 1
545      MOV     @1,KTENABLE                 ;TURN ON ABOVE 28K TEST FLAG
546      JSR     PC,T12CONVERT                ;CONVERT TEST PATTERN TO TEST ADDRESS
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

TSV5 : HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
 TEST 3: SUBTEST 1: GET STATUS SELECTED LOCATIONS

SEQ 106

|     |        |        |               |             |                           |  |                                     |
|-----|--------|--------|---------------|-------------|---------------------------|--|-------------------------------------|
| 545 | 026620 | 103034 |               | BCC         | 65:                       |  | ;BR IF INVALID PACKET ADDRESS       |
| 546 | 026622 | 013704 | 030324        | MOV         | T12LOAD, R4               |  | ;COPY CURRENT PACKET LOW ADDRESS    |
| 547 | 026626 | 013703 | 030322        | MOV         | T12HIADD, R3              |  | ;COPY CURRENT PACKET HIGH ADDRESS   |
| 548 | 026632 | 004737 | 031706        | JSR         | PC, T12SETGET             |  | ;SETUP CURRENT PACKET TO GET STATUS |
| 549 | 026636 | 042703 | 177774        | BIC         | #C<A1716>, R3             |  | ;SAVE ADDRESS BITS 17-16            |
| 550 | 026642 | 050304 |               | BIS         | R3, R4                    |  | ;SETUP 18 BIT PACKET ADDRESS        |
| 551 | 026644 | 004737 | 017274        | JSR         | PC, KTOFF                 |  | ;TURN OFF KT-11                     |
| 552 | 026650 | 010465 | 000000        | MOV         | R4, TSDR(R5)              |  | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 553 | 026654 | 004737 | 016336        | JSR         | PC, CHKTSR                |  | ;WAIT FOR SSR TO SET                |
| 554 | 026660 |        |               | FORCERROR   | 32:                       |  |                                     |
| 555 | 026674 | 103405 |               | BCS         | 40:                       |  | ;BR IF SSR SET IN CHKTSR            |
| 556 | 026676 | 010001 |               | MOV         | R0, R1                    |  | ;SAVE CONTENTS OF TSSR              |
| 557 | 026700 |        |               | NEXT, ERRNO |                           |  |                                     |
| 558 | 026700 |        |               | ERRDF       | ERRNO, T12GETSSR, PKTGETS |  | ;DEVICE FATAL JSR FAILED TO SET     |
|     | 026700 | 104455 |               |             |                           |  | TRAP C#ERDF                         |
|     | 026702 | 000457 |               |             |                           |  | .WORD 303                           |
|     | 026704 | 030516 |               |             |                           |  | .WORD T12GETSSR                     |
|     | 026706 | 012064 |               |             |                           |  | .WORD PKTGETS                       |
| 559 | 026710 |        |               | 40:         | CKLOOP                    |  | ;LOOP ON ERROR, IF FLAG SET         |
|     | 026710 | 104406 |               |             |                           |  | TRAP C#CLP1                         |
| 560 | 026712 |        |               | 65:         |                           |  |                                     |
| 561 | 026712 |        |               | FORCEEXIT   | 80:                       |  |                                     |
| 562 | 026722 | 020227 | 030466        | CMP         | R2, #T12TBE               |  | ;DONE ALL TSTBLK TEST PATTERNS?     |
| 563 | 026726 | 103002 |               | BHIS        | 70:                       |  | ;BR IF YES                          |
| 564 | 026730 | 000157 | 026560        | JMP         | T121LOOP                  |  | ;DO ANOTHER MODULO-4 ADDRESS        |
| 565 | 026734 | 005737 | 030330        | 70:         | TST T12KT                 |  | ;DONE ABOVE 28K TESTING TOO?        |
| 566 | 026740 | 003012 |               | BGT         | 80:                       |  | ;BR IF YES                          |
| 567 | 026742 | 005737 | 003132        | TST         | KTFLG                     |  | ;ANY MEMORY ABOVE 28K ON SYSTEM?    |
| 568 | 026746 | 001407 |               | BEQ         | 80:                       |  | ;BR IF NO                           |
| 569 | 026750 | 012737 | 000001 030330 | MOV         | #1, T12KT                 |  | ;SET SWITCH                         |
| 570 | 026756 | 012702 | 030334        | MOV         | #T12BLK, R2               |  | ;RESET TEST PATTERN TABLE           |
| 571 | 026762 | 000137 | 026560        | JMP         | T121LOOP                  |  | ;DO ABOVE 28K TESTING               |
| 572 | 026766 | 004737 | 017274        | 80:         | JSR PC, KTOFF             |  | ;TURN OFF KT11                      |
| 573 | 026772 |        |               | ENDSUB      |                           |  | ////////// END SUBTEST //////////   |
|     | 026772 | 104403 |               |             |                           |  | L10043:                             |
| 574 | 026774 | 005737 | 002222        | TST         | FATFLG                    |  | ;ANY FATAL ERRORS ?                 |
| 575 | 027000 | 001402 |               | BEQ         | 100:                      |  | ;BRANCH IF NOT                      |
| 576 | 027002 | 004737 | 017202        | JSR         | PC, CKDROP                |  | ;TRY TO DROP THE UNIT               |
| 577 | 027006 |        |               | 100:        |                           |  |                                     |

.SBTTL TEST 3: SUBTEST 2: MESSAGE PACKETS TO SELECTED LOCATIONS

\*\*\*  
 TEST 3: SUBTEST 2:

SUBTEST DESCRIPTION:

This subtest verifies the controller can deposit message packets to all available memory locations. Write Characteristics commands are executed with message buffer addresses set to various addresses in each available memory location. The various addresses are determined by floating a 1 then a 0 through the address bits.

TEST STEPS:

594

TSV5 HARDWARE TESTS MACRO M1113 06 FEB-84 17:14  
TEST 3: SUBTEST 2: MESSAGE PACKETS TO SELECTED LOCATIONS

SEQ 107

```

595      BEGIN
596      ; Write to TSSR to soft initialize
597      ; Do a WRITE CHARACTERISTICS to setup a message buffer to compare
598      ;
599      ; REPEAT FOR SELECTED ADDRESSES IN DIAGNOSTIC FREE SPACE AND ABOVE 32K
600      ; BEGIN
601      ; Get a valid modulo-4 test address
602      ; Set the packet message buffer to the TEST ADDRESS
603      ; Do a WRITE CHARACTERISTICS
604      ; Restore the test message buffer to background pattern
605      ; END
606      ; END
607      ; --
608
609      BGNSUB                                ;////////// BEGIN SUBTEST ////////////
                                           ;T3.2:
                                           ;TRAP      C$BSUB
610
611
612      ;Write to TSSR to soft initialize
613      027010 004737 015774      JSR      PC,SOFINIT      ;DO SOFT INIT OF CONTROLLER
614      027014 103405      BCS      15$      ;BR IF SOFT INIT = OK
615      027016      NEXT,ERRNO
616      027016 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
617      027020      ERRDF      ERRNO,SFIERR,SFIMSG ;DEVICE FATAL ERROR DURING INIT
                                           ;TRAP      C$ERDF
                                           ;.WORD      304
                                           ;.WORD      SFIERR
                                           ;.WORD      SFIMSG
        027020 104455
        027022 000460
        027024 003652
        027026 012034
618
619      ;Do a WRITE CHARACTERISTICS to setup a message buffer to compare
620      027030 15$:
621      027030 012704 030260      MOV      @T12PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
622      027034 004737 031640      JSR      PC,T12SWRT      ;SET PACKET TO WRITE CHARACTERISTICS
623      027040 004737 017274      JSR      PC,KTOFF      ;TURN OFF KT-11
624      027044 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS
625      027050 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
626      027054      FORCERROR      17$
627      027070 103405      BCS      20$      ;BR IF SSR SET IN CHKTSSR
628      027072 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
629      027074      NEXT,ERRNO
630      027074 17$:
        027074 104455
        027076 000461
        027100 030572
        027102 012046
                                           ;DEVICE FATAL SSR FAILED TO SET
                                           ;TRAP      C$ERDF
                                           ;.WORD      305
                                           ;.WORD      T12WRTSSR
                                           ;.WORD      PKTSSR
631
632
633      ;Get a valid modulo-4 test address
634      ;Set the packet message buffer to the test address
635      027104 005037 002222      ;Do a WRITE CHARACTERISTICS
636      027110 012703 030334      20$:
        027114      CLR      FATFLG      ;CLEAR FATAL ERROR FLAG
        027114      MOV      @T12BLK,R3      ;POINT TO TEST PATTERN TABLE
637      027114      T122LOOP:
        027114 012301      MOV      (R3)+,R1      ;GET TEST PATTERN ADDRESS
        027116 010100      MOV      R1,R0      ;GET ADDRESS ALL "18 BITS"
        027120 042700 177774      BIC      @177774,R0      ;LEAVE ONLY A17 AND A16
        027124 042701 000003      BIC      @3,R1      ;GET RID OF A17 AND A16

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 2: MESSAGE PACKETS TO SELECTED LOCATIONS

SEQ 108

```

642 027130 004737 031336      JSR      PC,T12CONVERT      ;CONVERT TEST PATTERN TO TEST ADDRESS
643 027134 103402              BCS      25$      ;BR IF VALID MESSAGE BUFFER ADDRESS
644 027136 000137 027234      JMP      150$      ;GET ANOTHER TEST PATTERN TO TRY
645 027142 012704 030260      25$:  MOV     #T12PACKET,R4      ;SET THE COMMAND PACKET ADDRESS
646 027146 004737 031640      JSR      PC,T12SWRT      ;SETUP T12PACKET TO WRITE CHAR.
647 027152 013737 030324 030270  MOV     T12LOADD,T12DATA      ;SETUP LOW ORDER MESSAGE BUFFER ADD.
648 027160 013737 030322 030272  MOV     T12HIADD,T12DATA+2      ;SETUP HIGH ORDER MESSAGE BUFFER ADD.
649 027166 004737 017274      JSR      PC,KTOFF      ;TURN OFF KT-11
650 027172 010465 000000      MOV     R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
651 027176 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
652 027202              FORCERROR      32$
653 027216 103405              BCS      50$      ;BR IF SSR SET IN CHKTSSR
654 027220 010001              MOV     R0,R1      ;SAVE CONTENTS OF TSSR
655 027222              NEXT.ERRNO
656 027222              32$:  ERRDF      ERRNO,T12WRTSSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERDF
                                .WORD      306
                                .WORD      T12WRTSSR
                                .WORD      PKTSSR
657 027232              50$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
658 027234              150$:
659 027234              FORCEXIT      160$
660 027244 020327 030466      CMP      R3,#T12TBE      ;DONE ALL TST12BLK TEST PATTERNS?
661 027250 103002              BHS      160$      ;BR IF YES
662 027252 000137 027114      JMP      T122LOOP      ;DO ANOTHER MODULO- 4 ADDRESS
663 027256 004737 017274      160$:  JSR      PC,KTOFF      ;TURN OFF KT11
664 027262              ENDSUB      ;////////// END SUBTEST ////////////
                                L10044:
                                TRAP      C$ESUB
665 027264 005737 002222      TST      FATFLG      ;ANY FATAL ERRORS ?
666 027270 001402              BEQ      180$      ;BRANCH IF NOT
667 027272 004737 017202      JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
668 027276              180$:
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691

```

.SBTTL TEST 3: SUBTEST 3: CHARACTERISTIC DATA SELECTED LOCATIONS

\*\*\*  
 ; TEST 3: SUBTEST 3:

; SUBTEST DESCRIPTION:

; This subtest verifies the controller can fetch a  
 ; Write Characteristic data block from all available  
 ; memory locations.  
 ; Write Characteristics commands are executed with  
 ; characteristic data blocks at various memory addresses.  
 ; The various memory addresses are determined by floating  
 ; a 1 then a 0 through the address bits.

; TEST STEPS:

; BEGIN

; Write to TSSR to soft initialize

; REPEAT FOR SELECTED VALID ADDRESSES IN DIAGNOSTIC FREE SPACE AND ABOVE 32K

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 3: CHARACTERISTIC DATA SELECTED LOCATIONS

SEQ 109

```

692      ; BEGIN
693      ; Get a valid test address
694      ; Set the test packet characteristics data pointer to the
695      ; test address.
696      ; Store expected characteristic data in test address block
697      ; Do a WRITE CHARACTERISTIC command
698      ; END
699      ; END
700      ; --
701
702      BGNSUB                      ;////////// BEGIN SUBTEST //////////
703      027276                      T3.3:
704      027276 104402              TRAP C$BSUB
705
706      ;Write to TSSR to soft initialize
707      JSR PC,SOFINIT             ;DO SOFT INIT OF CONTROLLER
708      BCS 20$                   ;BR IF SOFT INIT = OK
709      NEXT.ERRNO
710      MOV RO,R1                 ;SAVE CONTENTS OF TSSR
711      ERDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL ERROR DURING INIT
712
713      027310 104455              TRAP C$ERDF
714      027312 000463              .WORD 307
715      027314 003652              .WORD SFIERR
716      027316 012034              .WORD SFIMSG
717
718      ;Get a valid test address
719      20$: CLR FATFLG             ;CLEAR FATAL ERROR FLAG
720      CLR T12KT                 ;TEST ABOVE 28K SWITCH
721      MOV #T12BLK,R3           ;POINT TO TEST PATTERN TABLE
722
723      T123LOOP:
724      CLR KTENABLE             ;TURN OFF ABOVE 28K TEST FLAG
725      MOV (R3)+,R1             ;GET TEST PATTERN ADDRESS
726      MOV R1,R0                ;GET ADDRESS ALI. "18 BITS"
727      BIC #177774,R0           ;LEAVE ONLY A17 AND A16
728      BIC #3,R1                ;GET RID OF A17 AND A16
729      TST T12KT                ;TEST ABOVE 28K THIS TIME?
730      BEQ 25$                  ;BR IF N
731      MOV -2(R3),R0             ;GET TEST PATTERN AGAIN
732      BIC #C<A1716>,R0         ;SAVE 18 BIT ADDRESS ONLY
733      MOV #1,KTENABLE          ;TURN ON ABOVE 28K TEST FLAG
734      25$: JSR PC,T12CONVERT    ;CONVERT TEST PATTERN TO TEST ADDRESS
735      BCS 30$                  ;BR IF VALID TEST ADDRESS
736      JMP 60$                  ;GET NEXT TEST PATTERN
737
738      ;Set the test packet characteristics data pointer to the test address
739      30$: MOV #T12PACKET,R4    ;GET THE ADDRESS OF COMMAND PACKET
740      JSR PC,T12SWRT            ;RESTORE PACKET TO STARTING VALUES
741      MOV T12LOAD,PKLOW(R4)     ;STORE CHAR. DATA PTR LOW ADDRESS
742      MOV T12HIADD,PKHI(R4)    ;STORE CHAR. DATA PTR HIGH ADDRESS
743      JSR PC,T12CHAR            ;STORE EXPECTED DATA IN DATA BLOCK
744
745      ;Do a WRITE CHARACTERISTIC command
746      JSR PC,KTOFF              ;TURN OFF KT-11
747      MOV R4,TSD8(R5)           ;SET THE PACKET ADDRESS TO EXECUTE
748      JSR PC,CHKTSSR            ;WAIT FOR SSR TO SET
749      FORCERROR 32$
750      BCS 40$                   ;BR IF SSR SET IN CHKTSSR
751      MOV RO,R1                 ;SAVE CONTENTS OF TSSR

```

TSV5 : HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 3: CHARACTERISTIC DATA SELECTED LOCATIONS

SEQ 110

```

733 027476          NEXT.ERRNO
744 027476          32$: ERRDF  ERRNO,T12WRTSSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      027476 104455                                     TRAP  C$ERDF
      027500 000464                                     .WORD 308
      027502 030572                                     .WORD T12WRTSSR
      027504 012046                                     .WORD PKTSSR
745 027506          40$: CKLOOP                          ;LOOP ON ERROR, IF FLAG SET
      027506 104406                                     TRAP  C$CLP1
746 027510          60$:
747 027510 020327 030466      CMP  R3,#T12TBE          ;DONE ALL TSTBLK TEST PATTERNS?
748 027514 103002          BHIS 65$                   ;BR IF YES
749 027516 000137 027334      JMP  T123LOOP           ;DO ANOTHER MODULO- 4 ADDRESS
750 027522 005737 030330      65$: TST  T12KT          ;DONE ABOVE 28K TESTING TOO?
751 027526 003012          BGT  70$                   ;BR IF YES
752 027530 005737 003132      TST  KTFLG              ;ANY MEMORY ABOVE 28K ON SYSTEM?
753 027534 001407          BEQ  70$                   ;BR IF NO
754 027536 012737 000001 030330  MOV  #1,T12KT        ;SET SWITCH
755 027544 012703 030334      MOV  #T12BLK,R3         ;RESET TEST PATTERN TABLE
756 027550 000137 027334      JMP  T123LOOP           ;DO ABOVE 28K TESTING
757 027554 004737 017274      70$: JSR  PC,KTOFF       ;TURN OFF KT11
758 027560          ENDSUB                             ;///////////////// END SUBTEST ///////////////////
      027560          L10045:                          L10045:
      027560 104403          TRAP  C$ESUB
759 027562 005737 002222      TST  FATFLG             ;ANY FATAL ERRORS ?
760 027566 001402          BEQ  75$                   ;BRANCH IF NOT
761 027570 004737 017202      JSR  PC,CKDROP           ;TRY TO DROP THE UNIT
762 027574          75$:

```

.SBTTL TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

\*\*\*  
 ; TEST 3: SUBTEST 4:

; SUBTEST DESCRIPTION:

; This subtest verifies the NXM error bit in the TSSR  
 ; register is set when attempting to fetch data (a characteristic  
 ; data block) from selected nonexistent locations.  
 ; If NXM fails to set it is likely that an LSI-11 Bus driver is  
 ; failing to assert an address line.  
 ; Addresses tested include all combinations of high-order address  
 ; bits (i.e bits 16-21).

\*\*\*\*\*  
 CAUTION

; The LSI BUS drivers for all available address lines(16-21)  
 ; are only checked when running on a 11/23B system with more than  
 ; 128K words of memory!

\*\*\*\*\*

; TEST STEPS:

; BEGIN

; Write to TSSR to soft initialize  
 ; Do a write characteristic command  
 ; Invert the extended features switch

; REPEAT FOR SELECTED NON-EXISTENT MEMORY ADDRESSES

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 111

```

793      ;          BEGIN
794      ;          Get an invalid test address
795      ;          Set the test packet characteristics data pointer to the
796      ;          test address.
797      ;          Do a WRITE CHARACTERISTIC command
798      ;          If TSSR register NXM bit not set then print error message
799      ;          END
800      ;          END
801      ;          ---
802
803      027574      BGNSUB          ;//////////////// BEGIN SUBTEST //////////////////
      027574      T3.4:          TRAP      C$BSUB
      027574      104402
804
805
806      027576      005737      003144      TST      T23A          ;26-APR-83 REV B - CHK FOR 23A CPU
807      027602      001406      BEQ      5$          ;26-APR-83 REV B - BR, IF NOT 23A
808      027604      023727      002120      007777      CMP      L$HIME,07777      ;26-APR-83 REV B - CHK FOR > 256KB
809      027612      103402      BLO      5$          ;26-APR-83 REV B - BR, IF < 256KB
810      027614      000137      030206      JMP      NOEXTF      ;26-APR-83 REV B - JMP OVER 256KB
811      027620      5$:
812      027620      005737      003136      TST      NXMFLG      ;GOT ENOUGH MEMORY?
813      027624      001002      BNE      10$          ;IF SET STAY
814      027626      000137      030206      JMP      NOEXTF      ;LEAVE IF NOT SET
815
816      ;Write to TSSR to soft initialize
817
818      027632      004737      015774      10$:      JSR      PC,SOFINIT      ;DO SOFT INIT OF CONTROLLER
819      027636      103405      BCS      11$          ;BR IF SOFT INIT = OK
820      027640      NEXT.ERRNO
821      027640      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
822      027642      ERRDF      ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL ERROR DURING INIT
      027642      104455      TRAP      C$ERDF
      027644      000465      .WORD      309
      027646      003652      .WORD      SFIERR
      027650      012034      .WORD      SFIMSG
823
824
825      ;Do a WRITE CHARACTERISTIC command so to invert switch
826      027652      104406      11$:      CKLOOP          ;LOOP IF SELECTED
      027652      012704      030260      MOV      @T12PACKET,R4          TRAP      C$CLP1
827      027660      004737      031640      JSR      PC,T12SWRT      ;GET THE ADDRESS OF COMMAND PACKET
828      027664      005037      003134      CLR      KTENABLE      ;RESTORE PACKET TO STARTING VALUES
829      027670      010465      000000      MOV      R4,TSD8(R5)      ;TURN OFF KT-11
830      027674      004737      016336      JSR      PC,CHKTSSR      ;SET THE PACKET ADDRESS
831      027700      FORCERROR      15$      ;WAIT FOR SSR TO SET
832      027714      103405      BCS      17$          ;BR IF SSR SET IN CHKTSSR
833      027716      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
834      027720      NEXT.ERRNO
835      027720      104455      15$:      ERRDF      ERRNO,T12WRTSSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      027722      000466      TRAP      C$ERDF
      027724      030572      .WORD      310
      027726      012046      .WORD      T12WRTSSR
836      027730      104406      17$:      CKLOOP          ;LOOP IF SELECTED
      027730      .WORD      PKTSSR
      TRAP      C$CLP1

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 112

```

838 027732 004737 021206          JSR      PC,INVERT          ;INVERT THE SWITCH
839
840          ;Get an invalid test address
841
842 027736 005037 002222          20$:   CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
843 027742          25$:
844 027742 013737 003142 030322          MOV      NXMHI,T12HIADD      ;SAVE TEST ADDRESS HIGH
845 027750 013737 003140 030324          MOV      NXMLO,T12LOADD      ;SAVE TEST ADDRESS LOW
846 027756          T124LOOP:
847
848          ;Set the test packet characteristics data pointer to the
849          ; test address.
850
851 027756 012704 030260          30$:   MOV      @T12PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
852 027762 004737 031640          JSR      PC,T12SWRT      ;RESTORE PACKET TO STARTING VALUES
853 027766 013764 030324 000002          MOV      T12LOADC,PKLOW(R4)      ;STORE CHAR. DATA PTR LOW ADDRESS
854 027774 013764 030322 000004          MOV      T12HIADD,PKHI(R4)      ;STORE CHAR. DATA PTR HIGH ADDRESS
855
856          ;Do a WRITE CHARACTERISTIC command
857 030002 004737 017274          JSR      PC,KTOFF          ;TURN OFF KT-11
858 030006 010465 000000          MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
859 030012 004737 016250          JSR      PC,WAITF          ;WAIT FOR SSR TO SET
860 030016          FORCERROR      32$
861 030032 103407          BCS      40$          ;BR IF SSR SET IN CHKTSSR
862 030034 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
863 030036          NEXT,ERRNO
864 030036          32$:   ERDRF      ERRNO,T12WRTSSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
865          030036 104455          TRAP      C$ERDF
866          030040 000467          .WORD      311
867          030042 030572          .WORD      T12WRTSSR
868          030044 012046          .WORD      PKTSSR
869
870 030046 005237 002222          40$:   INC      FATFLG          ;SET FATAL ERROR FLAG
871 030052          CKLOOP          ;LOOP ON ERROR, IF FLAG SET
872 030054 104406          TRAP      C$CLP1
873 030054          FORCERROR      45$,NOTSSR
874 030064          ESCAPE      SUB          ;BY-PASS SUBTEST IF FATAL ERROR
875          030064 104410          TRAP      C$ESCAPE
876          030066 000124          .WORD      L10046-
877
878          ;If TSSR register NXM bit not set then print error message
879          45$:
880 030070 016501 000002          MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
881 030074          FORCERROR      52$
882 030110 032701 004000          BIT      @NXM,R1          ;NXM SET?
883 030114 001012          BNE      60$          ;BR IF YES
884 030116          NEXT,ERRNO
885 030116 013737 030324 002240          52$:   MOV      T12LOADD,ERRLO      ;MEMORY TEST ADDRESS LOW
886 030124 013737 030322 002236          MOV      T12HIADD,ERRHI      ;MEMORY TEST ADDRESS HIGH
887 030132          ERRHRD      ERRNO,T12NXM,ADDSSR      ;REPORT ADDRESS AND TSSR ERROR
888          030132 104456          TRAP      C$ERHRD
889          030134 000470          .WORD      312
890          030136 031227          .WORD      T12NXM
891          030140 012126          .WORD      ADDSSR
892
893 030142          60$:   CKLOOP          ;LOOP ON ERROR, IF FLAG SET
894 030142 104406          TRAP      C$CLP1
895 030144          FORCEEXIT      90$
896 030154 005737 003144          TST      T23A          ;IS IT A 11/23A?

```

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 113

|     |        |        |        |
|-----|--------|--------|--------|
| 883 | 030160 | 001012 |        |
| 884 | 030162 | 013700 | 030322 |
| 885 | 030166 | 005200 |        |
| 886 | 030170 | 020027 | 000077 |
| 887 | 030174 | 101004 |        |
| 888 | 030176 | 010037 | 030322 |
| 889 | 030202 | 000137 | 027756 |
| 890 | 030206 |        |        |
| 891 | 030206 |        |        |
| 892 | 030206 | 004737 | 017274 |
| 893 | 030212 |        |        |
|     | 030212 |        |        |
|     | 030212 | 104403 |        |
| 894 | 030214 | 005737 | 002222 |
| 895 | 030220 | 001402 |        |
| 896 | 030222 | 004737 | 017202 |
| 897 | 030226 | 004737 | 016456 |
| 898 | 030232 | 103002 |        |
| 899 | 030234 | 000137 | 026446 |
| 900 | 030240 |        |        |
| 901 | 030240 | 004737 | 017274 |
| 902 | 030244 | 005037 | 003150 |
| 903 | 030250 |        |        |
|     | 030250 | 104432 |        |
|     | 030252 | 001540 |        |
| 904 |        |        |        |
| 905 |        |        |        |
| 906 |        |        |        |
| 907 |        |        |        |
| 908 |        |        |        |
| 909 |        |        |        |
| 911 |        | 030260 |        |
| 913 | 030260 |        |        |
| 914 | 030260 | 100004 |        |
| 915 | 030262 | 030270 |        |
| 916 | 030264 | 000000 |        |
| 917 | 030266 | 000010 |        |
| 918 |        |        |        |
| 919 | 030270 |        |        |
| 920 | 030270 | 030302 |        |
| 921 | 030272 | 000000 |        |
| 922 | 030274 | 000016 |        |
| 923 | 030276 | 000000 | 000000 |
| 924 |        |        |        |
| 925 | 030302 |        |        |
| 926 |        |        |        |
| 927 | 030322 | 000000 |        |
| 928 | 030324 | 000000 |        |
| 929 | 030326 | 000000 |        |
| 930 | 030330 | 000000 |        |
| 931 | 030332 | 000000 |        |
| 932 |        |        |        |
| 933 |        |        |        |
| 934 |        |        |        |
| 935 |        |        |        |
| 936 |        |        |        |
| 937 | 030334 | 000001 |        |

```

BNE      90$
MOV      T12HIADD,RO
65$:     INC      RO
        CMP      RO,#77
        BHI      90$
75$:     MOV      RO,T12HIADD
        JMP      T124LOOP
90$:
NOEXTF:  JSR      PC,KTOFF
        ENDSUB

        TST      FATFLG
        BEQ      100$
        JSR      PC,CKDROF
100$:    JSR      PC,TSTLOOP
        BCC      105$
        JMP      T12LOOP
105$:    JSR      PC,KTOFF
        CLR      T3BFLG
        EXIT     TST

;+
;LOCAL STORAGE FOR THIS TEST
;-

        .-<..+10>E177770
T12PACKET:
        .WORD    100004
        .WORD    T12DATA
        .WORD    0
        .WORD    8.

T12DATA:
        .WORD    T12BFR
        .WORD    0
        .WORD    14.
        .WORD    0,0

T12BFR:  .BLKW   8.

T12HIADD:      .WORD    0
T12LOADD:      .WORD    0
T12PAR6:       .WORD    0
T12KT:         .WORD    0
T124TST:       .WORD    0
;+
;
;TABLE OF ADDRESSES
;
;
T12BLK: .WORD    000001

```

```

;YES WERE DONE
;GET CURRENT HIGH ADDRESS
;GET NEXT ADDRESS
;DONE A21-A16?
;BR IF YES
;SETUP NEW HIGH ORDER ADDRESS
;DO ANOTHER NON-EXISTENT ADDRESS

;TURN OFF KT11
;////////////////// END SUBTEST ////////////////////
                        L10046:
                                TRAP      C$ESUB

;ANY FATAL ERRORS ?
;BRANCH IF NOT
;TRY TO DROP THE UNIT
;SHOULD WE DO ITERATIONS?
;BR IF NO
;LOOP UNTIL ITERATION COUNT DONE

;TURN OFF MEMORY MANAGEMENT
;CLEAR TEST FLAG
;ALL DONE THIS TEST
                                TRAP      C$EXIT
                                .WORD     L10042-.

;COMMAND PACKET FOR TEST
;WRITE CHARACTERISTICS COMMAND, WITH ACK
;ADDRESS OF CHARACTERISTICS BLOCK

;STARTING VALUE OF BLOCK SIZE

;CHARACTERISTICS DATA BLOCK
;LOW ADDRESS OF MESSAGE BUFFER
;HIGH ORDER OF MESSAGE BUFFER
;LENGTH OF MESSAGE BUFFER

;MESSAGE BUFFER

;HIGH ADDRESS
;LOW ADDRESS
;ADDRESS IN PAR FORMAT
;TEST ABOVE 28K SWITCH
;ADDRESS TEST BIT

```

K9

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 114

|     |        |        |       |        |
|-----|--------|--------|-------|--------|
| 938 | 030336 | 000002 | .WORD | 000002 |
| 939 | 030340 | 000003 | .WORD | 000003 |
| 940 | 030342 | 000005 | .WORD | 000005 |
| 941 | 030344 | 000006 | .WORD | 000006 |
| 942 | 030346 | 000007 | .WORD | 000007 |
| 943 | 030350 | 000011 | .WORD | 000011 |
| 944 | 030352 | 000012 | .WORD | 000012 |
| 945 | 030354 | 000013 | .WORD | 000013 |
| 946 | 030356 | 000021 | .WORD | 000021 |
| 947 | 030360 | 000022 | .WORD | 000022 |
| 948 | 030362 | 000023 | .WORD | 000023 |
| 949 | 030364 | 000041 | .WORD | 000041 |
| 950 | 030366 | 000042 | .WORD | 000042 |
| 951 | 030370 | 000043 | .WORD | 000043 |
| 952 | 030372 | 000101 | .WORD | 000101 |
| 953 | 030374 | 000102 | .WORD | 000102 |
| 954 | 030376 | 000103 | .WORD | 000103 |
| 955 | 030400 | 000201 | .WORD | 000201 |
| 956 | 030402 | 000202 | .WORD | 000202 |
| 957 | 030404 | 000203 | .WORD | 000203 |
| 958 | 030406 | 000401 | .WORD | 000401 |
| 959 | 030410 | 000402 | .WORD | 000402 |
| 960 | 030412 | 000403 | .WORD | 000403 |
| 961 | 030414 | 001001 | .WORD | 001001 |
| 962 | 030416 | 001002 | .WORD | 001002 |
| 963 | 030420 | 001003 | .WORD | 001003 |
| 964 | 030422 | 002001 | .WORD | 002001 |
| 965 | 030424 | 002002 | .WORD | 002002 |
| 966 | 030426 | 002003 | .WORD | 002003 |
| 967 | 030430 | 004001 | .WORD | 004001 |
| 968 | 030432 | 004002 | .WORD | 004002 |
| 969 | 030434 | 004003 | .WORD | 004003 |
| 970 | 030436 | 010001 | .WORD | 010001 |
| 971 | 030440 | 010002 | .WORD | 010002 |
| 972 | 030442 | 010003 | .WORD | 010003 |
| 973 | 030444 | 020001 | .WORD | 020001 |
| 974 | 030446 | 020002 | .WORD | 020002 |
| 975 | 030450 | 020003 | .WORD | 020003 |
| 976 | 030452 | 040001 | .WORD | 040001 |
| 977 | 030454 | 040002 | .WORD | 040002 |
| 978 | 030456 | 040003 | .WORD | 040003 |
| 979 | 030460 | 100001 | .WORD | 100001 |
| 980 | 030462 | 100002 | .WORD | 100002 |
| 981 | 030464 | 100003 | .WORD | 100003 |
| 982 | 030466 | 177777 | .WORD | 177777 |

T12TBE: .WORD 177777

```

;
;LOCAL TEXT MESSAGES FOR TEST
;

```

|     |        |     |     |     |            |        |                                                                    |
|-----|--------|-----|-----|-----|------------|--------|--------------------------------------------------------------------|
| 987 | 030470 | 104 | 115 | 101 | TST12ID:   | .ASCIZ | 'DMA Memory Addressing'                                            |
| 988 | 030516 | 103 | 157 | 156 | T12GETSSR: | .ASCIZ | 'Contents of TSSR Incorrect After GET STATUS'                      |
| 989 | 030572 | 103 | 157 | 156 | T12VRTSSR: | .ASCIZ | 'Contents of TSSR Incorrect After WRITE CHARACTERISTICS'           |
| 990 | 030661 | 115 | 145 | 163 | T12MSGBUF: | .ASCIZ | 'Message Buffer Contents Incorrect After WRITE CHARACTERISTICS'    |
| 991 | 030757 | 102 | 141 | 143 | T12BKGNB:  | .ASCIZ | 'Background Pattern Disturbed By WRITE CHARACTERISTICS'            |
| 992 | 031045 | 105 | 170 | 160 | T12NINT:   | .ASCIZ | 'Expected Interrupt Not Received On WRITE CHARACTERISTICS'         |
| 993 | 031136 | 127 | 162 | 151 | T12DPR:    | .ASCIZ | 'Write Characteristic data in ram does not match expected'         |
| 994 | 031227 | 124 | 123 | 123 | T12NXM:    | .ASCIZ | 'TSSR NXM bit failed to set when non-existent memory address speci |

ed'

```

995                                     .EVEN
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017 031336
1018 031336
1019 031342 005037 030324
1020 031346 005037 030322
1021 031352 005037 030326
1022 031356 042701 170000
1023 031362 010005
1024 031364 004737 017274
1025 031370 013702 003124
1026 031374 062702 000620
1027 031400 060102
1028 031402 042702 000003
1029 031406 013703 003130
1030 031412 162703 000020
1031 031416 010237 030324
1032 031422 010237 030326
1033 031426 020203
1034 031430 191007
1035 031432 020237 003124
1036 031436 103007
1037 031440 005737 003134
1038 031444 001004
1039 031446 000424
1040 031450 162702 000020
1041 031454 000754
1042 031456
1043 031456 005737 003134
1044 031462 001420
1045 031464 005737 003132
1046 031470 001413
1047 031472 004737 017256
1048 031476 010500
1049 031500 010037 030322
1050 031504 010201
1051 031506 004737 017316

```

| Address | Hex | Dec | Label | Op | Op1 | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416</ |
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 117

```

1109 031636 000207          RTS      PC          ;RETURN
1110
1111
1112
1113
1114
1115
1116 031640
1117 031640
1118 031644 012701 030260
1119 031650 012721 100004
1120 031654 012721 030270
1121 031660 005021
1122 031662 012721 000010
1123 031666 012721 030302
1124 031672 005021
1125 031674 012721 000016
1126 031700 005021
1127 031702 005011
1128 031704 000207
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139 031706
1140 031706
1141 031712 010401
1142 031714 005737 003134
1143 031720 001404
1144 031722 010300
1145 031724 004737 017316
1146 031730 010001
1147 031732 012700 000017
1148 031736 052700 100000
1149 031742 010021
1150 031744 005021
1151 031746 000207
1152
1153
1154
1155
1156
1157
1158
1159 031750
1160 031750
1161 031754 012700 030270
1162 031760 013701 030324
1163 031764 005737 003134
1164 031770 001402
1165 031772 013701 030326

          ;+
          ;
          ;ROUTINE TO SETUP PACKET TO WRITE CHARACTERISTICS
          ;
          ;-
T12SWRT:
          SAVREG          ;SAVE THE REGISTERS
          MOV      #T12PACKET,R1      ;START OF THE PACKET
          MOV      #100004,(R1)+      ;WRITE CHARACTERISTICS WITH ACK
          MOV      #T12DATA,(R1)+    ;ADDRESS OF CHAR DATA BLOCK
          CLR      (R1)+              ;EXTENDED ADDRESS
          MOV      #8,(R1)+          ;SIZE OF DATA BLOCK IN BYTES
          MOV      #T12BFR,(R1)+     ;ADDRESS OF MESSAGE BUFFER
          CLR      (R1)+
          MOV      #14,(R1)+         ;LENGTH OF MESSAGE BUFFER
          CLR      (R1)+
          CLR      (R1)
          RTS      PC              ;RETURN

          ;+
          ;
          ;ROUTINE TO SETUP A GET STATUS COMMAND PACKET AT CURRENT PACKET ADDRESS
          ;
          ;      R3      HIGH ORDER PACKET ADDRESS
          ;      R4      LOW ORDER PACKET ADDRESS
          ;      NOTE: R3 IS IGNORED IF KTENABLE FLAG CLEAR
          ;
          ;-
T12SETGET:
          SAVREG          ;SAVE THE REGISTERS
          MOV      R4,R1      ;GET LOW ORDER ADDRESS
          TST      KTENABLE    ;TESTING ABOVE 28K?
          BEQ      10$        ;BR IF NO
          MOV      R3,R0      ;GET HIGH ORDER ADDRESS
          JSR      PC,SETMAP   ;RETURN ADDRESS BIASED TO PAR6 IN R0
          MOV      R0,R1      ;GET ADDRESS
10$:      MOV      #P.GETSTATUS,R0 ;GET STATUS COMMAND CODE NO IE
          B1S      #P.ACK,R0   ;SET ACK
          MOV      R0,(R1)+    ;STORE GET STATUS IN PACKET
          CLR      (R1)+       ;CLEAR UNUSED WORD
          RTS      PC          ;RETURN

          ;+
          ;
          ;ROUTINE TO SETUP A CHARACTERISTIC DATA BLOCK AT A TEST ADDRESS
          ;
          ;-
T12CHAR:
          SAVREG          ;SAVE R1-R5 UNTIL NEXT RETURN
          MOV      #T12DATA,R0    ;GET T12PACKET DATA POINTER
          MOV      T12LOAD,R1     ;ASSUME NOT ABOVE 28K
          TST      KTENABLE      ;TESTING ABOVE 28K?
          BEQ      10$          ;BR IF NO
          MOV      T12PAR6,R1     ;SET TEST ADDRESS ABOVE 28K

```

B10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 3; SUBTEST 4; NXM TO SELECTED INVALID ADDRESSES

SEQ 118

```
1166 031776 012021      101:  MOV      (R0)+,(R1)+      ;STORE DATA WORD 1
1167 032000 012021      MOV      (R0)+,(R1)+      ;STORE DATA WORD 2
1168 032002 012021      MOV      (R0)+,(R1)+      ;STORE DATA WORD 3
1169 032004 012021      MOV      (R0)+,(R1)+      ;STORE DATA WORD 4
1170 032006 012021      MOV      (R0)+,(R1)+      ;STORE DATA WORD 5
1171 032010 000207      RTS      PC                ;RETURN
1172
1173 032012                      ENDTST
      032012
      032012 104401
```

L10042: TRAP C\$ETST

C10

TSVS HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 3: SUBTEST 4: NXM TO SELECTED INVALID ADDRESSES

SEQ 119

```

1175
1176
1177
1178
1179
1180
1181
1182
1183
1184 032014
      032014
1185
1186
1191 032014 005737 002214
1192 032020 001402
1193 032022 005237 003400
1194 032026 012700 034453
1195 032032 004737 016510
1196 032036 012737 000005 002216
1197 032044
1198
1199
1200
1201
1202
1203
1204
1205
1206 032044
      032044
      032044 104402
1207 032046
      032046 012700 000000
      032052 104441
1208 032054 005737 003400
1209 032060 001402
1210 032062 000137 032344
1211 032066 004737 034472
1212 032072 004737 034544
1213 032076 004737 015774
1214 032102 103405
1218 032104 010001
1219 032106
      032106 104455
      032110 000621
      032112 003652
      032114 012034
1220 032116
1221 032116 012704 033370
1222 032122 004737 010662
1223 032126 103405
1227 032130 010001
1228 032132
      032132 104456
      032134 000622
      032136 005056
      032140 012034

      .SBTTL TEST 4: RAM EXERCISER TEST
      ;
      ; THIS TEST USES THE READ AND WRITE RAM (BOTH SINGLE AND 256
      ; LOCATIONS) SELECT CODES OF THE WRITE SUBSYSTEM MEMORY COMMAND
      ; TO EXERCISE THE CONTROLLER'S RAM MEMORY AND DMA LOGIC
      ;
      ;
      BGNTEST
      T4::
      TST TSTCNT ;CHECK FOR RUN MODE
      BEQ 101 ;BR, IF NOT ONLY PROGRAM RUN
      INC SKIP ;SET SKIP SW
      101: MOV #T15ID,R0 ;ASCII MESSAGE TO IDENTIFY TEST
      JSR PC,T15SETUP ;DO INITIAL TEST SETUP
      MOV #5,LOOPCNT ;PERFORM 5 ITERATIONS
T15LOOP:
      ;
      ; TEST 4, SUBTEST 1
      ;
      ; THIS SUBTEST WRITES THE ADDRESS (8 BITS) INTO THE
      ; RAM MEMORY SINGLE WORD (8 BITS) MODE
      ;
      ;
      BGNSUB ;////////// BEGIN SUBTEST //////////
      T4.1:
      SETPRI #PRI00 ;LOWER PRIORITY TO ALLOW INTERRUPTS
      TRAP C#BSUB
      MOV #PRI00,R0
      TRAP C#SPRI
      TST SKIP ;SHOULD WE SKIP THIS SUBTEST
      BEQ 101 ;BR, IF NOW SKIP REQUIRED
      JMP 501 ;SKIP SUBTEST
      101: JSR PC,T15REST ;SET COMMAND PACKET
      JSR PC,T15R12 ;SET UP OTHER COMMAND PACKET
      JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
      BCS 201 ;BR IF INIT WAS OK
      MOV R0,R1 ;CONTENTS OF TSSR REGISTER
      ERROF ERNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      TRAP C#ERDF
      .WORD 401
      .WORD SFIERR
      .WORD SFIMSG
      201: MOV #T15PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
      JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
      BCS 231 ;BR, IF COMMAND ISSUED OK
      MOV R0,R1 ;SAVE CONTENTS OF TSSR
      ERHRD ERNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
      TRAP C#ERHRD
      .WORD 402
      .WORD WRTMSG
      .WORD SFIMSG

```

D10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 120

|      |        |        |        |        |      |        |                      |  |                                       |
|------|--------|--------|--------|--------|------|--------|----------------------|--|---------------------------------------|
| 1229 | 032142 | 012703 | 000400 |        | 23%: | MOV    | #256,R3              |  | ;STARTING ADDRESS FOR RAM WRITE       |
| 1230 | 032146 | 112737 | 000001 | 034101 |      | MOVB   | #1,T15BS1            |  | ;SIZE OF TRANSFER                     |
| 1231 | 032154 | 112737 | 000002 | 034100 |      | MOVB   | #2,T15BS0            |  | ;WRITE RAM "COMMAND"                  |
| 1232 | 032162 |        |        |        | 25%: |        |                      |  |                                       |
| 1233 | 032162 | 010337 | 034102 |        |      | MOV    | R3,T15S2             |  | ;ADDRESS FOR RAM                      |
| 1234 | 032166 | 012704 | 034070 |        |      | MOV    | #T15PK2,R4           |  | ;WRITE SUBSYS MEM PACKET              |
| 1235 | 032172 | 110337 | 034104 |        |      | MOVB   | R3,T15S3             |  | ;DATA FOR WRITE (ADDRESS)             |
| 1236 | 032176 | 010465 | 000000 |        |      | MOV    | R4,TSD8(R5)          |  | ;ISSUE COMMAND                        |
| 1237 | 032202 | 004737 | 016336 |        |      | JSR    | PC,CHKTSSR           |  | ;WAIT FOR SSR                         |
| 1238 | 032206 | 103407 |        |        |      | BCS    | 30%                  |  | ;BR, IF NO ERROR                      |
| 1239 | 032210 | 010001 |        |        |      | MOV    | R0,R1                |  | ;ERROR, SAVE TSSR                     |
| 1243 | 032212 |        |        |        |      | ERRHRD | ERRNO,T15SSR,PKTSSR  |  | ;TSSR NOT CORRECT AFTER WRITE SUB MEM |
|      | 032212 | 104456 |        |        |      |        |                      |  | TRAP C\$ERHRD                         |
|      | 032214 | 000623 |        |        |      |        |                      |  | .WORD 403                             |
|      | 032216 | 034106 |        |        |      |        |                      |  | .WORD T15SSR                          |
|      | 032220 | 012046 |        |        |      |        |                      |  | .WORD PKTSSR                          |
| 1244 | 032222 |        |        |        |      | ESCAPE | SUB                  |  | ;DON'T CONTINUE IF ERROR ON WRITE     |
|      | 032222 | 104410 |        |        |      |        |                      |  | TRAP C\$ESCAPE                        |
|      | 032224 | 000122 |        |        |      |        |                      |  | .WORD L10050-                         |
| 1245 | 032226 |        |        |        | 30%: | CKLOOP |                      |  | ;SCOPE LOOP                           |
|      | 032226 | 104406 |        |        |      |        |                      |  | TRAP C\$CLP1                          |
| 1246 |        |        |        |        |      |        |                      |  |                                       |
| 1247 |        |        |        |        |      |        |                      |  |                                       |
| 1248 | 032230 | 005203 |        |        |      | INC    | R3                   |  | ;NEXT ADDRESS                         |
| 1249 | 032232 | 020327 | 010000 |        |      | CMP    | R3,#10000            |  | ;END OF RAM MEMORY CHECK              |
| 1250 | 032236 | 001351 |        |        |      | BNE    | 25%                  |  | ;LOOP TILL ALL RAM WRITTEN            |
| 1251 | 032240 | 005002 |        |        |      | CLR    | R2                   |  | ;CLEAR OUT R2 HIGH BITS               |
| 1252 | 032242 | 005303 |        |        |      | DEC    | R3                   |  | ;SET BACK TO 7777                     |
| 1253 | 032244 | 110337 | 034104 |        | 40%: | MOVB   | R3,T15S3             |  | ;GET DATA PATTERN BACK IN SHAPE       |
| 1254 | 032250 | 010337 | 034102 |        |      | MOV    | R3,T15S2             |  | ;ADDRESS FOR RAM READ                 |
| 1255 | 032254 | 112737 | 000001 | 034100 |      | MOVB   | #1,T15BS0            |  | ;READ RAM COMMAND                     |
| 1256 | 032262 | 010465 | 000000 |        |      | MOV    | R4,TSD8(R5)          |  | ;SEND OUT PACKET ADDRESS TO CONTR.    |
| 1257 | 032266 | 004737 | 016336 |        |      | JSR    | PC,CHKTSSR           |  | ;WAIT FOR READY, NON-AMBIGUOUS        |
| 1258 | 032272 | 103405 |        |        |      | BCS    | 43%                  |  | ;BR, IF NO PROBLEM                    |
| 1259 | 032274 | 010001 |        |        |      | MOV    | R0,R1                |  | ;SAVE TSSR                            |
| 1263 | 032276 |        |        |        |      | ERRDF  | ERRNO,T15SSR,PKTSSR  |  | ;TSSR NOT CORRECT                     |
|      | 032276 | 104455 |        |        |      |        |                      |  | TRAP C\$ERDF                          |
|      | 032300 | 000624 |        |        |      |        |                      |  | .WORD 404                             |
|      | 032302 | 034106 |        |        |      |        |                      |  | .WORD T15SSR                          |
|      | 032304 | 012046 |        |        |      |        |                      |  | .WORD PKTSSR                          |
| 1264 | 032306 |        |        |        | 43%: | CKLOOP |                      |  | ;SCOPE LOOP                           |
|      | 032306 | 104406 |        |        |      |        |                      |  | TRAP C\$CLP1                          |
| 1265 | 032310 | 013701 | 033432 |        |      | MOV    | T15BFR+20,R1         |  | ;GET RAM READ DATA                    |
| 1266 | 032314 | 010302 |        |        |      | MOV    | R3,R2                |  | ;SET UP FOR COMPARE                   |
| 1267 | 032316 | 120102 |        |        |      | CMPB   | R1,R2                |  | ;CHECK WITH DATA WRITTEN              |
| 1268 | 032320 | 001404 |        |        |      | BEG    | 45%                  |  | ;BR IF OK, DATA IN * DATA OUT         |
| 1272 | 032322 |        |        |        |      | ERRHRD | ERRNO,T15AM4,EXPBREC |  | ;WRITTEN DATA NOT * TO READ           |
|      | 032322 | 104456 |        |        |      |        |                      |  | TRAP C\$ERHRD                         |
|      | 032324 | 000625 |        |        |      |        |                      |  | .WORD 405                             |
|      | 032326 | 034365 |        |        |      |        |                      |  | .WORD T15AM4                          |
|      | 032330 | 015502 |        |        |      |        |                      |  | .WORD EXPBREC                         |
| 1273 | 032332 |        |        |        | 45%: | CKLOOP |                      |  | ;SCOPE LOOP                           |
|      | 032332 | 104406 |        |        |      |        |                      |  | TRAP C\$CLP1                          |
| 1274 | 032334 | 005303 |        |        |      | DEC    | R3                   |  | ;DROP DATA COUNTER (PATTERN)          |
| 1275 | 032336 | 020327 | 000377 |        |      | CMP    | R3,#255              |  | ;AT BOTTOM YET                        |
| 1276 | 032342 | 001340 |        |        |      | BNE    | 40%                  |  | ;BR, IF MORE TO CHECK                 |
| 1277 | 032344 |        |        |        | 50%: | CKLOOP |                      |  | ;SCOPE LOOP                           |

E10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 121

```

1278 032344 104406          ENDSUB
      032346
      032346 104403
1279
1280
1281 032350          BGNSUB
      032350
      032350 104402
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291 032352 004737 034472
1292 032356 004737 034544
1293 032362 004737 015774
1294 032366 103405
1298 032370 010001
1299 032372
      032372 104455
      032374 000626
      032376 003652
      032400 012034
1300 032402
1301 032402 012704 033370
1302 032406 004737 010662
1303 032412 103405
1307 032414 010001
1308 032416
      032416 104456
      032420 000627
      032422 005056
      032424 012034
1309 032426
1310 032426 112737 000001 034101
1311 032434 012704 034070
1312 032440 012703 000400
1313 032444 112737 000002 034100
1314 032452 105037 034104
1315 032456 010337 034102
1316 032462 010465 000000
1317 032466 004737 016336
1318 032472 103405
1319 032474 010001
1323 032476
      032476 104456
      032500 000630
      032502 034106
      032504 012046
1324 032506
      032506 104406
1325

; *
;
; TEST 4, SUBTEST 2
;
;
; THIS SUBTEST WRITES RAM WITH ALL ZEROS
; THEN WALKS AN ALL ONES WORD DOWN THROUGH MEMORY
;
;
; JSR      PC,T15REST          ;RESTORE PACKET FOR WRITE CHARA
; JSR      PC,T15RT2           ;RESTORE PACKET FOR WRT SUB SYS MEM
; JSR      PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
; BCS      20$                 ;BR IF INIT WAS OK
; MOV      R0,R1               ;CONTENTS OF TSSR REGISTER
; ERDF     ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
;                                TRAP      C$ERDF
;                                .WORD     406
;                                .WORD     SFIERR
;                                .WORD     SFIMSG
;
; 20$:
; MOV      @T15PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
; JSR      PC,WRTCHR           ;ISSUE WRITE CHARACTERISTICS
; BCS      25$                 ;BR, IF COMMAND ISSUED OK
; MOV      R0,R1               ;SAVE CONTENTS OF TSSR
; ERHRD    ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
;                                TRAP      C$ERHRD
;                                .WORD     407
;                                .WORD     WRTMSG
;                                .WORD     SFIMSG
;
; 25$:
; MOV      @1,T15BS1           ;SET SIZE OF TRANSFER 1 BYTE
; MOV      @T15PK2,R4          ;SET NEW PACKET ADDRESS
; MOV      @256.,R3            ;STARTING ADDRESS IN RAM
; MOV      @2,T15BS0           ;WRITE RAM COMMAND
; CLRB     T15S3               ;SET DATA TO 000
; 30$:
; MOV      R3,T15S2            ;ADDRESS TO PACKET DATA AREA
; MOV      R4,T5DB(R5)         ;SEND OUT PACKET ADDRESS
; JSR      PC,CHKTSSR          ;WAIT FOR SSR
; BCS      33$                 ;BR, IF NO PROBLEM
; MOV      R0,R1               ;SAVE TSSR
; ERHRD    ERRNO,T15SSR,PKTSSR ;TSSR NOT CORRECT
;                                TRAP      C$ERHRD
;                                .WORD     408
;                                .WORD     T15SSR
;                                .WORD     PKTSSR
;
; 33$: CKLOOP                  ;SCOPE LOOP
;                                TRAP      C$CLP1

```

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 122

| Line | Address | Offset | Label | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416 | Op417 | Op418 | Op419 | Op420 | Op421 | Op422 | Op423 | Op424 | Op425 | Op426 | Op427 | Op428 | Op429 | Op430 | Op431 | Op432 | Op433 | Op434 | Op435 | Op436 | Op437 | Op438 | Op439 | Op440 | Op441 | Op442 | Op443 | Op444 | Op445 | Op446 | Op447 | Op448 | Op449 | Op450 | Op451 | Op452 | Op453 | Op454 | Op455 | Op456 | Op457 | Op458 | Op459 | Op460 | Op461 | Op462 | Op463 |  |
|------|---------|--------|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|
|------|---------|--------|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|

G10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 123

|        |        |        |        |        |                            |       |                                     |
|--------|--------|--------|--------|--------|----------------------------|-------|-------------------------------------|
| 032716 | 104456 |        |        |        |                            | TRAP  | C\$ERHRD                            |
| 032720 | 000635 |        |        |        |                            | .WORD | 413                                 |
| 032722 | 034162 |        |        |        |                            | .WORD | T15AM2                              |
| 032724 | 015502 |        |        |        |                            | .WORD | EXPBREC                             |
| 1379   | 032726 |        |        | 45\$:  | CKLOOP                     |       | ;SCOPE LOOP                         |
|        | 032726 | 104406 |        |        |                            | TRAP  | C\$CLP1                             |
| 1380   | 032730 | 005303 |        |        | DEC R3                     |       | ;DROP RAM ADDRESS POINTER           |
| 1381   | 032732 | 020327 | 000377 |        | CMP R3,#255.               |       | ;AT START YET                       |
| 1382   | 032736 | 001271 |        |        | BNE 40\$                   |       | ;BR, IF MORE RAM TO CHECK           |
| 1383   |        |        |        |        |                            |       |                                     |
| 1384   | 032740 |        |        |        | ENDSUB                     |       | ////////// END SUBTEST //////////   |
|        | 032740 |        |        |        |                            |       | L10051:                             |
|        | 032740 | 104403 |        |        |                            | TRAP  | C\$ESUB                             |
| 1385   |        |        |        |        |                            |       |                                     |
| 1386   | 032742 |        |        |        | BGNSUB                     |       | ////////// BEGIN SUBTEST ////////// |
|        | 032742 |        |        |        |                            |       | T4.3:                               |
|        | 032742 | 104402 |        |        |                            | TRAP  | C\$BSUB                             |
| 1387   |        |        |        |        |                            |       |                                     |
| 1388   |        |        |        |        |                            |       |                                     |
| 1389   |        |        |        |        |                            |       |                                     |
| 1390   |        |        |        |        |                            |       |                                     |
| 1391   |        |        |        |        |                            |       |                                     |
| 1392   |        |        |        |        |                            |       |                                     |
| 1393   |        |        |        |        |                            |       |                                     |
| 1394   |        |        |        |        |                            |       |                                     |
| 1395   | 032744 | 005737 | 003400 |        | TST SKIPT                  |       | ;CHECK RUN MODE                     |
| 1396   | 032750 | 001402 |        |        | BEG 10\$                   |       | ;BR, IF NO SKIP                     |
| 1397   | 032752 | 000137 | 033300 |        | JMP 50\$                   |       | ;SKIP SUBTEST                       |
| 1398   | 032756 | 004737 | 034472 | 10\$:  | JSR PC,T15REST             |       | ;RESTORE PACKET FOR WRITE CHARA     |
| 1399   | 032762 | 004737 | 034544 |        | JSR PC,T15RT2              |       | ;RESTORE PACKET FOR WRT SUB SYS MEM |
| 1400   | 032766 | 004737 | 015774 |        | JSR PC,SOFINIT             |       | ;DO INITIALIZE ON CONTROLLER        |
| 1401   | 032772 | 105405 |        |        | BCS 20\$                   |       | ;BR IF INIT WAS OK                  |
| 1405   | 032774 | 010001 |        |        | MOV R0,R1                  |       | ;CONTENTS OF TSSR REGISTER          |
| 1406   | 032776 |        |        |        | ERRDF ERRNO,SFIERR,SFIMSG  |       | ;FATAL ERROR TSSR WAS NOT OK        |
|        | 032776 | 104455 |        |        |                            | TRAP  | C\$ERDF                             |
|        | 033000 | 000636 |        |        |                            | .WORD | 414                                 |
|        | 033002 | 003652 |        |        |                            | .WORD | SFIERR                              |
|        | 033004 | 012034 |        |        |                            | .WORD | SFIMSG                              |
| 1407   | 033006 |        |        | 20\$:  |                            |       |                                     |
| 1408   | 033006 | 01.704 | 033370 |        | MOV #T15PACKET,R4          |       | ;SUBROUTINE NEEDS PACKET ADDRESS    |
| 1409   | 033012 | 004737 | 010662 |        | JSR PC,WRTCHR              |       | ;ISSUE WRITE CHARACTERISTICS        |
| 1410   | 033016 | 103405 |        |        | BCS 25\$                   |       | ;BR, IF COMMAND ISSUED OK           |
| 1414   | 033020 | 010001 |        |        | MOV R0,R1                  |       | ;SAVE CONTENTS OF TSSR              |
| 1415   | 033022 |        |        |        | ERRHRD ERRNO,WRTMSG,SFIMSG |       | ;WRITE CHARACTERISTICS FAILED       |
|        | 033022 | 104456 |        |        |                            | TRAP  | C\$ERHRD                            |
|        | 033024 | 000637 |        |        |                            | .WORD | 415                                 |
|        | 033026 | 005056 |        |        |                            | .WORD | WRTMSG                              |
|        | 033030 | 012034 |        |        |                            | .WORD | SFIMSG                              |
| 1416   | 033032 |        |        | 25\$:  |                            |       |                                     |
| 1417   | 033032 | 112737 | 000001 | 034101 | MOVB #1,T15BS1             |       | ;SET SIZE TO 1 BYTE                 |
| 1418   | 033040 | 012704 | 034070 |        | MOV #T15PK2,R4             |       | ;SET NEW PACKET ADDRESS             |
| 1419   | 033044 | 012703 | 000400 |        | MOV #256.,R3               |       | ;STARTING ADDRESS IN RAM            |
| 1420   | 033050 | 112737 | 000002 | 034100 | MOVB #2,T15BS0             |       | ;WRITE RAM COMMAND                  |
| 1421   | 033056 | 112737 | 000377 | 034104 | MOVB #377,T15S3            |       | ;SET DATA TO 377                    |
| 1422   | 033064 | 010337 | 034102 |        | MOV R3,T15S2               |       | ;ADDRESS TO PACKET DATA AREA        |
| 1423   | 033070 | 010465 | 000000 |        | MOV R4,T5DB(R5)            |       | ;SEND OUT PACKET ADDRESS            |
| 1424   | 033074 | 004737 | 016336 |        | JSR PC,CHKTSSR             |       | ;WAIT FOR SSR                       |

## H10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 124

|      |        |        |        |        |        |                      |            |                                    |       |          |
|------|--------|--------|--------|--------|--------|----------------------|------------|------------------------------------|-------|----------|
| 1425 | 033100 | 103405 |        |        | BCS    | 33\$                 |            | ;BR, IF NO PROBLEM                 |       |          |
| 1426 | 033102 | 010001 |        |        | MOV    | R0,R1                |            | ;SAVE TSSR                         |       |          |
| 1430 | 033104 |        |        |        | ERRHRD | ERRNO,T15SSR,PKTSSR  |            | ;TSSR NOT CORRECT                  |       |          |
|      | 033104 | 104456 |        |        |        |                      |            |                                    | TRAP  | C\$ERHRD |
|      | 033106 | 000640 |        |        |        |                      |            |                                    | .WORD | 416      |
|      | 033110 | 034106 |        |        |        |                      |            |                                    | .WORD | T15SSR   |
|      | 033112 | 012046 |        |        |        |                      |            |                                    | .WORD | PKTSSR   |
| 1431 | 033114 |        |        |        | 33\$:  | CKLOOP               |            | ;SCOPE LOOP                        |       |          |
|      | 033114 | 104406 |        |        |        |                      |            |                                    | TRAP  | C\$CLP1  |
| 1432 |        |        |        |        |        |                      |            |                                    |       |          |
| 1433 |        |        |        |        |        |                      |            |                                    |       |          |
| 1434 | 033116 | 005203 |        |        | INC    | R3                   |            | ;NEXT ADDRESS                      |       |          |
| 1435 | 033120 | 020327 | 010000 |        | CMP    | R3,#10000            |            | ;END OF RAM MEMORY CHECK           |       |          |
| 1436 | 033124 | 001357 |        |        | BNE    | 30\$                 |            | ;BR, MORE RAM TO GO                |       |          |
| 1437 | 033126 | 005303 |        |        | DEC    | R3                   |            | ;SET BACK TO 7777                  |       |          |
| 1438 | 033130 | 112702 | 000377 |        | 40\$:  | MOVB                 | #377,R2    | ;SET TO ALL ONES                   |       |          |
| 1439 | 033134 | 112737 | 000001 | 034100 | MOVB   | #1,T15BS0            |            | ;READ RAM COMMAND                  |       |          |
| 1440 | 033142 | 010337 | 034102 |        | MOV    | R3,T15S2             |            | ;ADDRESS TO BE READ TO PACKET DATA |       |          |
| 1441 | 033146 | 010465 | 000000 |        | MOV    | R4,TSDB(R5)          |            | ;SEND OUT PACKET ADDRESS           |       |          |
| 1442 | 033152 | 004737 | 016336 |        | JSR    | PC,CHKTSSR           |            | ;WAIT FOR SSR TO SET               |       |          |
| 1443 | 033156 | 103405 |        |        | BCS    | 41\$                 |            | ;BR, IF ALL IS WELL                |       |          |
| 1444 | 033160 | 010001 |        |        | MOV    | R0,R1                |            | ;SAVE TSSR                         |       |          |
| 1448 | 033162 |        |        |        | ERRHRD | ERRNO,T15SSR,PKTSSR  |            | ;TSSR NOT CORRECT                  |       |          |
|      | 033162 | 104456 |        |        |        |                      |            |                                    | TRAP  | C\$ERHRD |
|      | 033164 | 000641 |        |        |        |                      |            |                                    | .WORD | 417      |
|      | 033166 | 034106 |        |        |        |                      |            |                                    | .WORD | T15SSR   |
|      | 033170 | 012046 |        |        |        |                      |            |                                    | .WORD | PKTSSR   |
| 1449 | 033172 |        |        |        | 41\$:  | CKLOOP               |            | ;SCOPE LOOP                        |       |          |
|      | 033172 | 104406 |        |        |        |                      |            |                                    | TRAP  | C\$CLP1  |
| 1450 | 033174 | 013701 | 033432 |        | MOV    | T15BFR+20,R1         |            | ;PICK UP READ DATA                 |       |          |
| 1451 | 033200 | 120102 |        |        | CMPB   | R1,R2                |            | ;BOTH SHOULD BE 11111111 BINARY    |       |          |
| 1452 | 033202 | 001404 |        |        | BEQ    | 42\$                 |            | ;BR, IF DATA IS GOOD               |       |          |
| 1456 | 033204 |        |        |        | ERRHRD | ERRNO,T15AM3,EXPBREC |            | ;CHARACTERISTICS DATA NOT CORRECT  |       |          |
|      | 033204 | 104456 |        |        |        |                      |            |                                    | TRAP  | C\$ERHRD |
|      | 033206 | 000642 |        |        |        |                      |            |                                    | .WORD | 418      |
|      | 033210 | 034263 |        |        |        |                      |            |                                    | .WORD | T15AM3   |
|      | 033212 | 015502 |        |        |        |                      |            |                                    | .WORD | EXPBREC  |
| 1457 | 033214 | 012702 | 000377 |        | 42\$:  | MOV                  | #000377,R2 | ;SET ALL ONES WORD                 |       |          |
| 1458 | 033220 | 012737 | 000002 | 034100 | MOV    | #2,T15BS0            |            | ;WRITE RAM COMMAND                 |       |          |
| 1459 | 033226 | 112737 | 000377 | 034104 | MOVB   | #000377,T15S3        |            | ;ALL ONES PATTERN                  |       |          |
| 1460 | 033234 | 010465 | 000000 |        | MOV    | R4,TSDB(R5)          |            | ;PASS PACKET ADDRESS TO CONTR.     |       |          |
| 1461 | 033240 | 004737 | 016336 |        | JSR    | PC,CHKTSSR           |            | ;WAIT FOR SSR                      |       |          |
| 1462 | 033244 | 103405 |        |        | BCS    | 43\$                 |            | ;BR, IF OK (NO ERROR)              |       |          |
| 1463 | 033246 | 010001 |        |        | MOV    | R0,R1                |            | ;SAVE TSSR                         |       |          |
| 1467 | 033250 |        |        |        | ERRHRD | ERRNO,T15SSR,PKTSSR  |            | ;TSSR NOT CORRECT                  |       |          |
|      | 033250 | 104456 |        |        |        |                      |            |                                    | TRAP  | C\$ERHRD |
|      | 033252 | 000643 |        |        |        |                      |            |                                    | .WORD | 419      |
|      | 033254 | 034106 |        |        |        |                      |            |                                    | .WORD | T15SSR   |
|      | 033256 | 012046 |        |        |        |                      |            |                                    | .WORD | PKTSSR   |
| 1468 | 033260 |        |        |        | 43\$:  | CKLOOP               |            | ;SCOPE LOOP                        |       |          |
|      | 033260 | 104406 |        |        |        |                      |            |                                    | TRAP  | C\$CLP1  |
| 1469 | 033262 | 112737 | 000001 | 034100 | MOVB   | #1,T15BS0            |            | ;SET UP FOR RAM READ               |       |          |
| 1470 | 033270 | 010465 | 000000 |        | MOV    | R4,TSDB(R5)          |            | ;ISSUE RAM READ                    |       |          |
| 1471 | 033274 | 004737 | 016336 |        | JSR    | PC,CHKTSSR           |            | ;WAIT FOR SSR TO SET               |       |          |
| 1472 | 033300 | 103405 |        |        | BCS    | 44\$                 |            | ;BR, IF OK (NO ERROR)              |       |          |
| 1473 | 033302 | 010001 |        |        | MOV    | R0,R1                |            | ;SAVE TSSR                         |       |          |
| 1477 | 033304 |        |        |        | ERRHRD | ERRNO,T15SSR,PKTSSR  |            | ;TSSR NOT CORRECT                  |       |          |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 125

|        |        |        |        |               |                      |                                              |
|--------|--------|--------|--------|---------------|----------------------|----------------------------------------------|
| 033304 | 104456 |        |        |               | TRAP                 | C\$ERHRD                                     |
| 033306 | 000644 |        |        |               | .WORD                | 420                                          |
| 033310 | 034106 |        |        |               | .WORD                | T15SSR                                       |
| 033312 | 012046 |        |        |               | .WORD                | PKTSSR                                       |
| 1478   | 033314 | 013701 | 033432 | 44\$: MOV     | T15BFR+20,R1         | ;PICK UP REC'D DATA                          |
| 1479   | 033320 | 120102 |        | CMPB          | R1,R2                | ;CHECK WITH DATA WRITTEN                     |
| 1480   | 033322 | 001404 |        | BEQ           | 45\$                 | ;BR IF OK, DATA IN = DATA OUT                |
| 1484   | 033324 |        |        | ERRHRD        | ERRNU,T15AM2,EXPBREC | ;WRITTEN DATA NOT = TO READ                  |
| 033324 | 104456 |        |        |               |                      | TRAP                                         |
| 033326 | 000645 |        |        |               |                      | .WORD                                        |
| 033330 | 034162 |        |        |               |                      | .WORD                                        |
| 033332 | 015502 |        |        |               |                      | .WORD                                        |
| 1485   | 033334 |        |        | 45\$: CKLOOP  |                      | ;SCOPE LOOP                                  |
| 033334 | 104406 |        |        |               |                      | TRAP                                         |
| 1486   | 033336 | 005303 |        | DEC           | R3                   | ;DROP RAM ADDRESS POINTER                    |
| 1487   | 033340 | 020327 | 000377 | CMP           | R3,4255.             | ;AT START YET                                |
| 1488   | 033344 | 001271 |        | BNE           | 40\$                 | ;BR, IF MORE RAM TO CHECK                    |
| 1489   |        |        |        |               |                      |                                              |
| 1490   | 033346 |        |        | 50\$: ENDSUB  |                      | ;////////// END SUBTEST //////////           |
| 1491   | 033346 |        |        |               |                      | L10052:                                      |
| 033346 | 104403 |        |        |               |                      | TRAP                                         |
| 1492   |        |        |        |               |                      | C\$ESUB                                      |
| 1493   | 033350 | 004737 | 016456 | JSR           | PC,TSTLOOP           | ;DO WE NEED TO ITERATE TEST ?                |
| 1494   | 033354 | 103002 |        | BCC           | 63\$                 | ;BRANCH IF NOT                               |
| 1495   | 033356 | 000137 | 032044 | JMP           | T15LOOP              | ;EXECUTE AGAIN                               |
| 1496   | 033362 |        |        | 63\$: EXIT    | TST                  | ;ALL DONE THIS TEST                          |
| 033362 | 104432 |        |        |               |                      | TRAP                                         |
| 033364 | 001216 |        |        |               |                      | .WORD                                        |
| 1497   |        |        |        |               |                      | C\$EXIT                                      |
| 1498   |        |        |        |               |                      | L10047-                                      |
| 1499   |        |        |        |               |                      |                                              |
| 1500   |        |        |        |               |                      |                                              |
| 1502   |        | 033370 |        |               |                      |                                              |
| 1504   | 033370 |        |        | T15PACKET:    |                      | ;COMMAND PACKET FOR TEST                     |
| 1505   | 033370 | 100204 |        | .WORD         | 100204               | ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK |
| 1506   | 033372 | 033400 |        | .WORD         | T15DATA              | ;ADDRESS OF CHARACTERISTICS BLOCK            |
| 1507   | 033374 | 000000 |        | .WORD         | 0                    |                                              |
| 1508   | 033376 | 000010 |        | .WORD         | 8.                   | ;STARTING VALUE OF BLOCK SIZE                |
| 1509   | 033400 |        |        | T15DATA:      |                      | ;CHARACTERISTICS DATA BLOCK                  |
| 1510   | 033400 | 033412 |        | .WORD         | T15BFR               | ;ADDRESS OF MESSAGE BUFFER                   |
| 1511   | 033402 | 000000 |        | .WORD         | 0                    |                                              |
| 1512   | 033404 | 000400 |        | .WORD         | 256.                 | ;LENGTH OF MESSAGE BUFFER                    |
| 1513   | 033406 | 000000 | 000000 | .WORD         | 0.0                  |                                              |
| 1514   | 033412 |        |        | T15BFR: .BLKW | 150.                 | ;MESSAGE BUFFER                              |
| 1515   |        |        |        |               |                      |                                              |
| 1516   |        |        |        |               |                      |                                              |
| 1517   |        |        |        |               |                      |                                              |
| 1519   |        | 034070 |        |               |                      |                                              |
| 1521   | 034070 |        |        | T15PK2:       |                      |                                              |
| 1522   | 034070 | 100206 |        | .WORD         | 100206               | ;WRITE SUB SYS MEM COMMAND, IE AND ACK       |
| 1523   | 034072 | 034100 |        | .WORD         | T15BF2               | ;ADDRESS OF SELECT BLOCK DATA                |
| 1524   | 034074 | 000000 |        | .WORD         | 0                    |                                              |
| 1525   | 034076 | 000006 |        | .WORD         | 6.                   | ;SIZE OF DATA PACKET                         |
| 1526   |        |        |        |               |                      |                                              |
| 1527   |        |        |        |               |                      |                                              |
| 1528   | 034100 |        |        | T15BF2:       |                      |                                              |

J10

TSVS - HARDWARE TESTS    MACRO M1113    06-FEB-84 17:14  
TEST 4: RAM EXERCISER TEST

SEQ 126

```

1529 034100      000      T15BS0: .BYTE 0      ;BSEL0 AREA
1530 034101      000      T15BS1: .BYTE 0      ;BSEL1 AREA
1531 034102 000000      T15S2:  .WORD 0      ;SEL 2 AREA
1532 034104 000000      T15S3:  .WORD 0      ;DATA AREA
1533
1534
1535
1536
1537
1538      ;*
1539      ;LOCAL TEXT MESSAGES FOR TEST
1540      ;-
1541 034106      127      122      111 T15SSR: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted'
1542 034162      127      122      111 T15AM2: .ASCIZ 'WRITE SUBSYSTEM MEMORY COMMAND Failed On All Ones Word Read Back'
1543 034263      127      122      111 T15AM3: .ASCIZ 'WRITE SUBSYSTEM MEMORY COMMAND Failed On All Zeros Word Read Back'
1544 034365      127      122      111 T15AM4: .ASCIZ 'WRITE SUBSYSTEM MEMORY COMMAND Failed On Address Test'
1545 034453      122      101      115 TST15ID: .ASCIZ 'RAM Exerciser'
1546      .EVEN
1547
1548      ;*
1549      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
1550      ;WRITE SUBSYSTEM MEMORY COMMAND
1551      ;
1552      ;-
1553
1554 034472      T15REST:
1555 034472      SAVREG      ;SAVE THE REGISTERS
1556 034476 012701 033370      MOV      @T15PACKET,R1      ;START OF THE PACKET
1557 034502 012721 100204      MOV      @100204,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK, IE
1558 034506 012721 033400      MOV      @T15DATA,(R1)+      ;ADDRESS OF CHARACTERISTIC DATA BLOCK
1559 034512 005021      CLR      (R1)+      ;EXTENDED ADDRESS
1560 034514 012721 000010      MOV      @8.,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
1561 034520 012721 033412      MOV      @T15BFR,(R1)+      ;ADDRESS OF MESSAGE BUFFER
1562 034524 005021      CLR      (R1)+
1563 034526 012721 000400      MOV      @256.,(R1)+      ;LENGTH OF MESSAGE BUFFER
1564 034532 005021      CLR      (R1)+
1565 034534 005011      CLR      (R1)
1566 034536 005037 033412      CLR      T15BFR      ;CLEAR 1ST LOC IN MESSAGE BUFFER
1567 034542 000207      RTS      PC      ;RETURN
1568
1569
1570 034544      T15RT2:
1571 034544      SAVREG      ;SAVE THE REGISTERS
1572 034550 012701 034070      MOV      @T15PK2,R1      ;START OF THE PACKET
1573 034554 012721 100206      MOV      @100206,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK, IE
1574 034560 012721 034100      MOV      @T15BF2,(R1)+      ;ADDRESS OF DATA BLOCK
1575 034564 005021      CLR      (R1)+      ;EXTENDED ADDRESS
1576 034566 012721 000006      MOV      @6.,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
1577 034572 005021      CLR      (R1)+
1578 034574 005021      CLR      (R1)+
1579 034576 005011      CLR      (R1)
1580 034600 000207      RTS      PC      ;RETURN
1581 034602      ENDTST
      034602
      034602 104401

```

L10047:    TRAP    C\$ETST

K10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB 84 17:14  
TEST 5: EXTENDED FEATURES SWITCH AND TIMERS A,B

SEQ 127

```

1583          .SBTTL TEST 5: EXTENDED FEATURES SWITCH AND TIMERS A,B
1584          ;**
1585          ; TEST DESCRIPTION:
1586          ;
1587          ; This test verifies the Invert Extended Features function
1588          ; can logically invert the Extended features switch and
1589          ; that the internal timers A and B operate correctly.
1590          ;
1591          ; TEST STEPS:
1592          ;
1593          ; REPEAT FOR LOOPCNT
1594          ; BEGIN
1595          ; Do Subtest 1 - Verify Extended Features Switch
1596          ; Do Subtest 2 - Verify Timers A,B
1597          ; END
1598          ;--
1599
1600
1601          034604          BGNTS1
1602          034604
1603          1606 034604 012700 036662          MOV      #TST16ID,RO          ;ASCII MESSAGE TO IDENTIFY TEST
1604          1607 034610 004737 016510          JSR      PC,TSTSETUP          ;DO INITIAL TEST SETUP
1605          1608 034614 012737 000012 002216          MOV      #10,,LOOPCNT          ;PERFORM 10 ITERATIONS
1606          1609 034622          T16LOOP:
1607          1610          .SBTTL TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST
1608          1611          ;**
1609          ; TEST 5: SUBTEST 1:
1610          ;
1611          ; SUBTEST DESCRIPTION:
1612          ;
1613          ; This subtest verifies that the Invert Sense of Extended features
1614          ; Switch function (Write Subsystem Memory,Write Misc command)
1615          ; operates properly.
1616          ; First the state of the Extended Features switch is read in the
1617          ; message packet supplied by the write characteristics command.
1618          ; Then, the sense of the switch is logically inverted.
1619          ; A Write characteristics command is executed and it is verified
1620          ; that the Extended status register (XST4) is returned when
1621          ; in Extended mode, and not returned if not in extended mode.
1622          ; The subtest also verifies that specifying a Message Buffer
1623          ; address with any of bits 21-19 ,set will cause the command to
1624          ; be rejected.
1625          ;
1626          ; TEST STEPS:
1627          ;
1628          ; BEGIN
1629          ; Write to TSSR register to soft initialize the controller
1630          ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
1631          ; IF Extended Features Hardware Switch CLEAR
1632          ; THEN
1633          ; (* Verify Extended Features switch can be Inverted to SET *)
1634          ; Do Write Subsystem Write Miscellaneous to SET Extended Features.
1635          ; DO a WRITE CHARACTERISTICS with an extended characteristic word
1636          ; Compare the controller ram to the extended characteristic word
1637          ;
1638          ;
1639          ;
1640          ;
1641          ;
1642          ;

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST

SEQ 128

```

1643      ;      If Data word in controller ram NOT= to word sent Then Print Error
1644      ;      If Message Buffer Data Length NOT= 12. Then Print Error
1645      ;      ELSE
1646      ;      (* Verify Extended Features switch can be Inverted to CLEAR *)
1647      ;      Do Write Subsystem Write Miscellaneous to CLEAR Extended Features.
1648      ;      Do a WRITE CHARACTERISTICS without an extended characteristic word
1649      ;      If Message Buffer Data Length NOT= 10. Then Print Error
1650      ;      END-IF
1651      ;      (* Verify Function Reject when Message Buffer 21-19 are non-zero *)
1652      ;      Write to TSSR register to soft initialize the controller
1653      ;      REPEAT FOR MESSAGE BUFFER ADDRESS bits <21:19> FROM 0 TO 7
1654      ;      DO a WRITE CHARACTERISTICS with a message address bit<21:19> non-zero
1655      ;      If TSSR termination code NOT= Function Reject Then Print Error
1656      ;      END-REPEAT
1657      ;      END
1658      ;      --
1659      034622      BGNSUB                      ;////////// BEGIN SUBTEST //////////
           034622                      T5.1:
           034622      104402                      TRAP      C$BSUB

1660
1661
1662      034624      5$:
1663      ;      Write to TSSR register to soft initialize the controller
1664      034624      004737      015774      JSR      PC,SOFINIT      ;WRITE TO TSSR TO SOFT INITIALIZE
1665      034630      103405      BCS      10$      ;BR IF SOFT INIT OKAY
1666      034632      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
1667      034634      ERRDF      ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL DURING INIT
           034634      104455                      TRAP      C$ERDF
           034636      000764                      .WORD      500
           034640      003652                      .WORD      SFIERR
           034642      012034                      .WORD      SFIMSG

1668      ;      Do WRITE CHARACTERISTICS to check for Extended Features Switch
1669      034644      004737      040030      10$:      JSR      PC,T16REST      ;RESTORE PACKET DEFAULTS
1670      034650      005037      002222      CLR      FATFLG      ;CLEAR FATAL ERROR FLAG
1671      034654      012704      040210      MOV      @T16PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
1672      034660      004737      010662      JSR      PC,WRTCHR      ;DO WRITE CHARACTERISTICS COMMAND
1673      034664      FORCERROR      12$      ;GOODFORCE ERROR IF FORCER=1
1674      034700      103407      BCS      15$      ;BR IF CARRY SET (GOOD RETURN)
1675      034702      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
1676      034704      NEXT.ERRNO
1677      034704      12$:      ERRDF      ERRNO,T16SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
           034704      104455                      TRAP      C$ERDF
           034706      000765                      .WORD      501
           034710      036732                      .WORD      T16SSR
           034712      012046                      .WORD      PKTSSR

1678      034714      005237      002222      INC      FATFLG      ;SET FATAL ERROR FLAG
1679      034720      15$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
           034720      104406                      TRAP      C$CLP1

1680
1681      ;      If Extended Features Hardware Switch Clear then:
1682      ;      (* Verify Extended Features switch can be Inverted to SET *)
1683      ;      REPEAT FOR TEST PATTERNS IN TSTBLK TABLE
1684      034722      012701      040232      MOV      @T16BFR,R1      ;MESSAGE BUFFER ADDRESS
1685      034726      032761      000200      000012      BIT      @X2.EXTF,XST2(R1)      ;EXTENDED FEATURES SWITCH CLEAR?
1686      034734      001402      BEQ      20$      ;BR IF YES
1687      034736      000137      035306      JMP      200$
1688      034742      012703      002764      20$:      MOV      @TSTBLK+10,,R3      ;START OF TEST DATA

```

## M10

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST

SEQ 129

```

1689 ; Do Write Subsystem Write Miscellaneous to SET Extended Features.
1690
1691 034746 004737 040170 JSR PC,T16SEXT ;SETUP PACKET FOR WRITE MISC INVERT
1692 034752 012704 040260 MOV #T16PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
1693 034756 010465 000000 MOV R4,TSD8(R5) ;SET THE PACKET ADDRESS TO EXECUTE
1694 034762 004737 016336 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
1695 034766 FORCERROR 32$ ;BDDFORCE ERROR IF FORCER=1
1696 035002 103407 BCS 40$ ;BR IF CARRY SET (GOOD RETURN)
1697 035004 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
1698 035006 NEXT.ERRNO
1699 035006 32$: ERRDF ERRNO,T162SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP C$ERDF
                                .WORD 502
                                .WORD T162SSR
                                .WORD PKTSSR
                                1700 035016 005237 002222 INC FATFLG ;SET FATAL ERROR FLAG
                                1701 035022 40$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
                                035022 104406 TRAP C$CLP1
1702
1703 ; DO a WRITE CHARACTERISTICS with an extended characteristic word
1704 035024 012737 125252 002312 MOV #125252,DATA ;SETUP TEST DATA FOR EXTENDED WORD
1705 035032 012704 040210 MOV #T16PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
1706 035036 012764 000020 000006 MOV #16.,PKBCNT(R4) ;STORE MESSAGE PACKET SIZE
1707 035044 013737 002312 040230 MOV DATA,T16DATA+10 ;STORE TEST DATA IN EXTENDED WORD
1708 035052 004737 010662 JSR PC,WRCHR ;DO WRITE CHARACTERISTICS COMMAND
1709 035056 FORCERROR 42$ ;BDDFORCE ERROR IF FORCER=1
1710 035072 103407 BCS 50$ ;BR IF CARRY SET (GOOD RETURN)
1711 035074 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
1712 035076 NEXT.ERRNO
1713 035076 42$: ERRDF ERRNO,T16SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP C$ERDF
                                .WORD 503
                                .WORD T16SSR
                                .WORD PKTSSR
                                1714 035106 005237 002222 INC FATFLG ;SET FATAL ERROR FLAG
                                1715 035112 50$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
                                035112 104406 TRAP C$CLP1
1716 ; If the TSBA Address Register NOT= Expected Then Print Error
1717 035114 016501 000000 MOV TSBA(R5),R1 ;GET TSBA REGISTER CONTENTS
1718 035120 012702 040232 MOV #T16BFR,R2 ;START OF THE DATA BUFFER
1719 035124 062702 000020 62$: ADD #16.,R2 ;EXPECTED CONTENTS OF TSBA
1720 035130 FORCERROR 72$,NOTSSR ;BDDFORCE ERROR IF FORCER=1
1721 035140 020102 CMP R1,R2 ;COMPARE EXPECTED TO RECEIVED
1722 035142 001404 BEQ 80$ ;ERROR IF NOT EQUAL
1723 035144 NEXT.ERRNO
1724 035144 72$: ERRHRD ERRNO,T16TSBA,EXPREC ;PRINT THE ERROR & EXPD/RECV
                                TRAP C$ERHRD
                                .WORD 504
                                .WORD T16TSBA
                                .WORD EXPREC
                                1725 035154 80$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
                                035154 104406 TRAP C$CLP1
1726 ; Compare the controller ram to the extended characteristic word
1727 ; If Data word in controller ram NOT= to word sent Then Print Error
1728 035156 012704 040220 MOV #T16DATA,R4 ;GET CHARACTERISTIC DATA ADDRESS
1729 035162 004737 011224 JSR PC,CKRAM2 ;DOES RAM DATA EQUAL DATA SENT?
1730 035166 FORCERROR 92$ ;BDDFORCE ERROR IF FORCER=1

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST

SEQ 130

| Line | Address | Offset | Label  | Instruction                                                          | Comment                             | Trap  | Code     |
|------|---------|--------|--------|----------------------------------------------------------------------|-------------------------------------|-------|----------|
| 1731 | 035202  | 103404 |        | BCS 100\$                                                            | ;BR IF YES                          |       |          |
| 1732 | 035204  |        |        | NEXT.ERRNO                                                           |                                     |       |          |
| 1733 | 035204  |        | 92\$:  | ERRHRD ERRNO,PKTRAM,RAMERR                                           | ;REPORT THE RAM ERROR(S)            |       |          |
|      | 035204  | 104456 |        |                                                                      |                                     | TRAP  | C\$ERHRD |
|      | 035206  | 000771 |        |                                                                      |                                     | .WORD | 505      |
|      | 035210  | 004745 |        |                                                                      |                                     | .WORD | PKTRAM   |
|      | 035212  | 015510 |        |                                                                      |                                     | .WORD | RAMERR   |
| 1734 | 035214  |        | 100\$: | CKLOOP                                                               | ;LOOP ON ERROR, IF FLAG SET         |       |          |
|      | 035214  | 104406 |        |                                                                      |                                     | TRAP  | C\$CLP1  |
| 1735 |         |        |        | ; If Message Buffer Data Length NOT= 12. Then Print Error            |                                     |       |          |
| 1736 | 035216  | 012702 | 040232 | MOV #T16BFR,R2                                                       | ;GET MESSAGE BUFFER ADDRESS         |       |          |
| 1737 | 035222  | 016201 | 000002 | MOV 2(R2),R1                                                         | ;GET RECV DATA FIELD LENGTH         |       |          |
| 1738 | 035226  | 012702 | 000014 | MOV #12.,R2                                                          | ;GET EXPD DATA FIELD LENGTH         |       |          |
| 1739 | 035232  |        |        | FORCERROR 112\$,NOTSSR                                               | ;GOODFORCE ERROR IF FORCER=1        |       |          |
| 1740 | 035242  | 020102 |        | CMP R1,R2                                                            | ;COMPARE EXPECTED TO RECEIVED       |       |          |
| 1741 | 035244  | 001404 |        | BEQ 120\$                                                            | ;ERROR IF NOT EQUAL                 |       |          |
| 1742 | 035246  |        |        | NEXT.ERRNO                                                           |                                     |       |          |
| 1743 | 035246  |        | 112\$: | ERRHRD ERRNO,T16LEN,EXPREC                                           | ;PRINT THE ERROR & EXPD/RECV        |       |          |
|      | 035246  | 104456 |        |                                                                      |                                     | TRAP  | C\$ERHRD |
|      | 035250  | 000772 |        |                                                                      |                                     | .WORD | 506      |
|      | 035252  | 037202 |        |                                                                      |                                     | .WORD | T16LEN   |
|      | 035254  | 015474 |        |                                                                      |                                     | .WORD | EXPREC   |
| 1744 | 035256  |        | 120\$: | CKLOOP                                                               | ;LOOP ON ERROR, IF FLAG SET         |       |          |
|      | 035256  | 104406 |        |                                                                      |                                     | TRAP  | C\$CLP1  |
| 1745 |         |        |        |                                                                      |                                     |       |          |
| 1746 | 035260  | 004737 | 015774 | JSR PC,SOFINIT                                                       | ;WRITE TO TSSR TO SOFT INITIALIZE   |       |          |
| 1747 | 035264  | 103405 |        | BCS 125\$                                                            | ;BR IF SOFT INIT OKAY               |       |          |
| 1748 | 035266  | 010001 |        | MOV R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |       |          |
| 1749 | 035270  |        |        | ERRDF ERRNO,SFIERR,SFIMSG                                            | ;DEVICE FATAL DURING INIT           |       |          |
|      | 035270  | 104455 |        |                                                                      |                                     | TRAP  | C\$ERDF  |
|      | 035272  | 000772 |        |                                                                      |                                     | .WORD | 506      |
|      | 035274  | 003652 |        |                                                                      |                                     | .WORD | SFIERR   |
|      | 035276  | 012034 |        |                                                                      |                                     | .WORD | SFIMSG   |
| 1750 | 035300  |        | 125\$: | CKLOOP                                                               | ;LOOP IF SELECTED                   |       |          |
|      | 035300  | 104406 |        |                                                                      |                                     | TRAP  | C\$CLP1  |
| 1751 | 035302  | 000137 | 035466 | JMP 300\$                                                            |                                     |       |          |
| 1752 |         |        |        |                                                                      |                                     |       |          |
| 1753 |         |        |        | ; (* Verify Extended Features switch can be Inverted to CLEAR *)     |                                     |       |          |
| 1754 | 035306  |        | 200\$: |                                                                      |                                     |       |          |
| 1755 |         |        |        | ; Do Write Subsystem Write Miscellaneous to CLEAR Extended Features. |                                     |       |          |
| 1756 | 035306  | 001737 | 040170 | JSR PC,I16SEXT                                                       | ;SETUP PACKET FOR WRITE MISC INVERT |       |          |
| 1757 | 035312  | 012704 | 040260 | MOV #T16PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |          |
| 1758 | 035316  | 010465 | 000000 | MOV R4,TSDB(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE  |       |          |
| 1759 | 035322  | 004737 | 016336 | JSR PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                |       |          |
| 1760 | 035326  |        |        | FORCERROR 232\$                                                      | ;GOODFORCE ERROR IF FORCER=1        |       |          |
| 1761 | 035342  | 103407 |        | BCS 240\$                                                            | ;BR IF CARRY SET (GOOD RETURN)      |       |          |
| 1762 | 035344  | 010001 |        | MOV R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |       |          |
| 1763 | 035346  |        |        | NEXT.ERRNO                                                           |                                     |       |          |
| 1764 | 035346  |        | 232\$: | ERRDF ERRNO,T162SSR,PKTSSR                                           | ;DEVICE FATAL SSR FAILED TO SET     |       |          |
|      | 035346  | 104455 |        |                                                                      |                                     | TRAP  | C\$ERDF  |
|      | 035350  | 000773 |        |                                                                      |                                     | .WORD | 507      |
|      | 035352  | 036767 |        |                                                                      |                                     | .WORD | T162SSR  |
|      | 035354  | 012046 |        |                                                                      |                                     | .WORD | PKTSSR   |
| 1765 | 035356  | 005237 | 002222 | INC FATFLG                                                           | ;SET FATAL ERROR FLAG               |       |          |
| 1766 | 035362  |        | 240\$: | CKLOOP                                                               | ;LOOP ON ERROR, IF FLAG SET         |       |          |
|      | 035362  | 104406 |        |                                                                      |                                     | TRAP  | C\$CLP1  |
| 1767 |         |        |        |                                                                      |                                     |       |          |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST

SEQ 131

```

1768      DO a WRITE CHARACTERISTICS without an extended characteristic word
1769 035364 012704 040210      MOV      #T16PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
1770 035370 012764 000016 000006      MOV      #14.,PKBCNT(R4)      ;STORE MESSAGE PACKET SIZE
1771 035376 004737 010662      JSR      PC,WRTCHR      ;DO WRITE CHARACTERISTICS COMMAND
1772 035402      FORCERROR      2421      ;BDDFORCE ERROR IF FORCER=1
1773 035416 103407      BCS      2501      ;BR IF CARRY SET (GOOD RETURN)
1774 035420 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
1775 035422      NEXT,ERRNO
1776 035422 2421:      ERRDF      ERRNO,T16SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C1ERDF
                                .WORD      508
                                .WORD      T16SSR
                                .WORD      PKTSSR
                                035422 104455
                                035424 000774
                                035426 036732
                                035430 012046
1777 035432 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
1778 035436 2501:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C1CLP1
                                035436 104406
1779      If Message Buffer Data Length NOT= 10. Then Print Error
1780 035440 013701 040234      MOV      T16BFR+2,R1      ;GET RECV DATA FIELD LENGTH
1781 035444 012702 000012      MOV      #10.,R2      ;GET EXPD DATA FIELD LENGTH
1782 035450 020102      CMP      R1,R2      ;COMPARE EXPECTED TO RECEIVED
1783 035452 001404      BEQ      2701      ;ERROR IF NOT EQUAL
1784 035454      NEXT,ERRNO
1785 035454 2621:      ERRHRD      ERRNO,T16LEN,EXPREC      ;PRINT THE ERROR & EXPD/RECV
                                TRAP      C1ERHRD
                                .WORD      509
                                .WORD      T16LEN
                                .WORD      EXPREC
                                035454 104456
                                035456 000775
                                035460 037202
                                035462 015474
1786 035464 2701:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C1CLP1
                                035464 104406
1787
1788
1789      (* Verify Function Reject when Message Buffer 21:19 are non-zero *)
1790      Write to TSSR register to soft initialize the controller
1791 035466 3001:
1792      REPEAT FOR MESSAGE BUFFER ADDRESS bits <21:19> FROM 0 TO 7
1793 035466 012737 000001 002312 3201:      MOV      #1,DATA      ;START AT BITS<21:19>=001
1794      DO a WRITE CHARACTERISTICS with a message address bit<21:19> non zero
1795 035474 3251:
1796 035474 012704 040210      MOV      #T16PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
1797 035500 012764 000016 000006      MOV      #14.,PKBCNT(R4)      ;STORE MESSAGE PACKET SIZE
1798 035506 013700 002312      MOV      DATA,R0      ;GET TEST DATA
1799      .REPT      3
1800      ASL      R0      ;SHIFT INTO BITS 21:19
1801      .ENDR
1802 035520 010037 040222      MOV      R0,T16DATA+2      ;STORE BUFFER ADDRESS BITS 21:19
1803 035524 010465 000000      MOV      R4,TSSR(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
1804 035530 004737 016250      JSR      PC,WAITF      ;WAIT FOR SSR
1805 035534      FORCERROR      3421      ;BDDFORCE ERROR IF FORCER=1
1806 035550 103407      BCS      3501      ;BR IF CARRY SET (GOOD RETURN)
1807 035552 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
1808 035554      NEXT,ERRNO
1809 035554 3421:      ERRDF      ERRNO,T16SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C1ERDF
                                .WORD      510
                                .WORD      T16SSR
                                .WORD      PKTSSR
                                035554 104455
                                035556 000776
                                035560 036732
                                035562 012046
1810 035564 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG

```

C11

TSVS - HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 5: SUBTEST 1: VERIFY EXTENDED FEATURES TEST

SEQ 132

```

1811 035570          350$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      035570 104406          TRAP      C$CLP1
1812
1813          ; If TSSR termination code NOT= Function Reject Then Print Error
1814 035572 016501 000002          MOV      TSSR(R5),R1          ;GET RECV TSSR
1815 035576 010102          MOV      R1,R2          ;COPY RECV TSSR
1816 035600 042702 000016          BIC      #TERCLS,R2          ;CLEAR TC<2:0> EXPD
1817 035604 052702 000006          BIS      #TSREJ,R2          ;SET EXPD TC<2:0>= FUNCTION REJECT
1818 035610          FORCEERROR 352$,NOTSSR          ;800FORCE ERROR IF FORCER=1
1819 035620 020102          CMP      R1,R2          ;EXPD EQUAL RECV?
1820 035622 001404          BEQ      360$          ;BR IF YES
1821 035624          NEXT.ERRNO
1822 035624          352$: ERRHRD  ERRNO,T16REJ,PKTSSR          ;DEVICE FATAL SSR FAILED TO SET
      035624 104456          TRAP      C$ERRHRD
      035626 000777          .WORD      511
      035630 037314          .WORD      T16REJ
      035632 012046          .WORD      PKTSSR
1823 035634          360$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      035634 104406          TRAP      C$CLP1
1824 035636          FORCEEXIT 370$
1825 035646 005237 002312          INC      DATA
1826 035652 023727 002312 000007          CMP      DATA,#7          ;GET NEXT TST PATTERN
1827 035660 101002          BHI      370$          ;DONE ALL DATA?
1828 035662 000137 035474          JMP      325$          ;BR IF YES
1829          END-REPEAT          ;DO ANOTHER TEST PATTERN
1830 035666          370$:
1831 035666          ENDSUB          ;////////// END SUBTEST //////////
      035666          L10054:
      035666 104403          TRAP      C$ESUB
1832
1833 035670 005737 002222          TST      FATELG          ;ANY FATAL ERRORS ?
1834 035674 001402          BEQ      460$          ;BRANCH IF NOT
1835 035676 004737 017202          JSR      PC,CKDROP          ;TRY TO DROP THE UNIT
1836 035702          460$:
1837
1838
1839
1840
1841
1842          .SBTTL TEST 5: SUBTEST 2: VERIFY TIMERS A,B
1843
1844          ;**
1845          ; TEST 5: SUBTEST 2:
1846          ;
1847          ; SUBTEST DESCRIPTION:
1848          ;
1849          ; This subtest verifies that timers A,B can be reset
1850          ; and that Timer A is twice the frequency of Timer B.
1851          ; Timer A has a period of 25 microseconds and Timer B
1852          ; has a period of 50 microseconds. The timers are
1853          ; checked at 1, 28, 53, and 78 microseconds.
1854          ;
1855          ; TEST STEPS:
1856          ;
1857          ;
1858          ; Write to TSSR register to soft initialize the controller
1859          ; Do WRITE CHARACTERISTICS to setup a Message Buffer

```

```

1860      (* Verify Timers A,B after RESET TIMER with 0 microsecond delay *)
1861      Do a Write Control RESET TIMER with 1 microsecond delay
1862      Do a Write Subsystem READ STATUS
1863      If Timer A NOT= 0 Then Print Error
1864      If Timer B NOT= 0 Then Print Error
1865      (* Verify Timers A,B after RESET TIMER with 28 microsecond delay *)
1866      Do a Write Control RESET TIMER with 28 microsecond delay
1867      If Timer A NOT= 1 Then Print Error
1868      If Timer B NOT= 1 Then Print Error
1869      Do a Write Control RESET TIMER with 53 microsecond delay
1870      If Timer A NOT= 0 Then Print Error
1871      If Timer B NOT= 1 Then Print Error
1872      Do a Write Control RESET TIMER with 78 microsecond delay
1873      If Timer A NOT= 1 Then Print Error
1874      If Timer B NOT= 0 Then Print Error
1875      ---
1876 035702      BGNSUB                      ;////////// BEGIN SUBTEST //////////
      035702                      TS.2:
      035702 104402                      TRAP      C#BSUB
1877      ;
1878 035704      ;
1879 035704 004737 015774      5$:      Write to TSSR register to soft initialize the controller
1880 035710 103405      JSR      PC,SOFINIT          ;WRITE TO TSSR TO SOFT INITIALIZE
1881 035712 010001      BCS      10$                ;BR IF SOFT INIT OKAY
1882 035714      MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
      035714 104455      ERRDF  ERRNO,SFIERR,SFIMSG    ;DEVICE FATAL DURING INIT
      035716 000777      TRAP      C#ERDF
      035720 003652      .WORD    511
      035722 012034      .WORD    SFIERR
      .WORD    SFIMSG
1883      ;
1884 035724 004737 040030      10$:      Do WRITE CHARACTERISTICS to setup a Message Buffer
1885 035730 005037 002222      JSR      PC,T16REST      ;RESTORE PACKET DEFAULTS
1886 035734 012704 040210      CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
1887 035740 012764 000010 000006      MOV      #T16PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
1888 035746 004737 010662      MOV      #8.,PKBCNT(R4)      ;MESSAGE PACKET SIZE NO EXTEND
1889 035752      JSR      PC,WRTCHR          ;DO WRITE CHARACTERISTICS COMMAND
1890 035766 103407      FORCERROR 12$          ;GOODFORCE ERROR IF FORCER=1
1891 035770 010001      BCS      15$                ;BR IF CARRY SET (GOOD RETURN)
1892 035772      MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
1893 035772      NEXT,ERRNO
      035772 104455      12$:      ERRDF  ERRNO,T16SSR,PKTSSR    ;DEVICE FATAL SSR FAILED TO SET
      035774 001000      TRAP      C#ERDF
      035776 036732      .WORD    512
      036000 012046      .WORD    T16SSR
      .WORD    PKTSSR
1894 036002 005237 002222      15$:      INC      FATFLG          ;SET FATAL ERROR FLAG
1895 036006      CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      036006 104406      TRAP      C#CLP1
1896      ;
1897      ;
1898      (* Verify Timers A,B after RESET TIMER with 1 microsecond delay *)
1899 036010 012700 000001      Do a Write Control RESET TIMER with 1 microsecond delay
1900 036014 013701 036652      MOV      #MS,RSD,R0      ;RESET TIMER COMMAND
1901 036020 004737 040142      MOV      T16D01,R1      ;1 MICROSECOND DELAY
1902 036024 012704 040260      JSR      PC,T16WMISC      ;SETUP T16PK2 COMMAND PACKET
1903 036030 010465 000000      MOV      #T16PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
1904 036034 004737 016336      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
1905 036040      JSR      PC,CHKTSSR          ;WAIT FOR SSR TO SET
      FORCERROR 32$          ;GOODFORCE ERROR IF FORCER=1

```

E11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 2: VERIFY TIMERS A,B

SEQ 134

|      |        |        |        |            |            |                                                          |                                     |          |  |
|------|--------|--------|--------|------------|------------|----------------------------------------------------------|-------------------------------------|----------|--|
| 1906 | 036054 | 103407 |        | BCS        | 40\$       |                                                          | ;BR IF CARRY SET (GOOD RETURN)      |          |  |
| 1907 | 036056 | 010001 |        | MOV        | R0,R1      |                                                          | ;SAVE CONTENTS OF TSSR              |          |  |
| 1908 | 036060 |        |        | NEXT,ERRNO |            |                                                          |                                     |          |  |
| 1909 | 036060 |        |        | 32\$:      | ERRDF      | ERRNO,T162SSR,PKTSSR                                     | ;DEVICE FATAL SSR FAILED TO SET     |          |  |
|      | 036060 | 104455 |        |            |            |                                                          | TRAP                                | C\$ERDF  |  |
|      | 036062 | 001001 |        |            |            |                                                          | .WORD                               | 513      |  |
|      | 036064 | 036767 |        |            |            |                                                          | .WORD                               | T162SSR  |  |
|      | 036066 | 012046 |        |            |            |                                                          | .WORD                               | PKTSSR   |  |
| 1910 | 036070 | 005237 | 002222 |            | INC        | FATFLG                                                   | ;SET FATAL ERROR FLAG               |          |  |
| 1911 | 036074 |        |        | 40\$:      | CKLOOP     |                                                          | ;LOOP ON ERROR, IF FLAG SET         |          |  |
|      | 036074 | 104406 |        |            |            |                                                          | TRAP                                | C\$CLP1  |  |
| 1912 |        |        |        | :          |            | If Timer A NOT= 0 Then Print Error                       |                                     |          |  |
| 1913 |        |        |        | :          |            | If Timer B NOT= 0 Then Print Error                       |                                     |          |  |
| 1914 | 036076 | 005002 |        |            | CLR        | R2                                                       | ;INIT EXPD                          |          |  |
| 1915 | 036100 | 042702 | 000010 |            | BIC        | #S2,ATIM,R2                                              | ;TIMER A EXPD=0                     |          |  |
| 1916 | 036104 | 042702 | 000004 |            | BIC        | #S2,BTIM,R2                                              | ;TIMER B EXPD=0                     |          |  |
| 1917 | 036110 | 012700 | 040252 |            | MOV        | #T16BFSTA,P0                                             | ;GET RECV READ STATUS               |          |  |
| 1918 | 036114 | 016001 | 000002 |            | MOV        | 2(R0),R1                                                 | ;GET RECV BYTE 2                    |          |  |
| 1919 | 036120 | 042701 | 177763 |            | BIC        | #C<S2,ATIM!S2,BTIM>,R1                                   | ;SAVE TIMER A;B RECV ONLY           |          |  |
| 1920 | 036124 |        |        |            | FORCERROR  | 72\$,NOTSSR                                              | ;000                                |          |  |
| 1921 | 036134 | 020201 |        |            | CMP        | R2,R1                                                    | ;EXPD EQUAL RECV?                   |          |  |
| 1922 | 036136 | 001404 |        |            | BEQ        | 80\$                                                     | ;BR IF YES                          |          |  |
| 1923 | 036140 |        |        |            | NEXT,ERRNO |                                                          |                                     |          |  |
| 1924 | 036140 |        |        | 72\$:      | ERRHRD     | ERRNO,T16T01,TIMEXP                                      | ;REPORT ERROR                       |          |  |
|      | 036140 | 104456 |        |            |            |                                                          | TRAP                                | C\$ERHRD |  |
|      | 036142 | 001002 |        |            |            |                                                          | .WORD                               | 514      |  |
|      | 036144 | 037431 |        |            |            |                                                          | .WORD                               | T16T01   |  |
|      | 036146 | 015552 |        |            |            |                                                          | .WORD                               | TIMEXP   |  |
| 1925 | 036150 |        |        | 80\$:      | CKLOOP     |                                                          | ;LOOP ON ERROR, IF FLAG SET         |          |  |
|      | 036150 | 104406 |        |            |            |                                                          | TRAP                                | C\$CLP1  |  |
| 1926 |        |        |        | :          |            | Do a Write Control RESET TIMER with 28 microsecond delay |                                     |          |  |
| 1927 |        |        |        |            | MOV        | #MS,RSD,R0                                               | ;RESET TIMER COMMAND                |          |  |
| 1928 | 036152 | 012700 | 000001 |            | MOV        | T16D28,R1                                                | ;28 MICROSECOND DELAY               |          |  |
| 1929 | 036156 | 013701 | 036654 |            | JSR        | PC,T16WMISC                                              | ;SETUP T16PK2 COMMAND PACKET        |          |  |
| 1930 | 036162 | 004737 | 040142 |            | MOV        | #T16PK2,R4                                               | ;GET WRITE SUBSYSTEM COMMAND PACKET |          |  |
| 1931 | 036166 | 012704 | 040260 |            | MOV        | R4,TSDB(R5)                                              | ;SET THE PACKET ADDRESS TO EXECUTE  |          |  |
| 1932 | 036172 | 010465 | 000000 |            | JSR        | PC,CHKTSSR                                               | ;WAIT FOR SSR TO SET                |          |  |
| 1933 | 036176 | 004737 | 016336 |            | FORCERROR  | 112\$                                                    | ;000FORCE ERROR IF FORCER=1         |          |  |
| 1934 | 036202 |        |        |            | BCS        | 120\$                                                    | ;BR IF CARRY SET (GOOD RETURN)      |          |  |
| 1935 | 036216 | 103407 |        |            | MOV        | R0,R1                                                    | ;SAVE CONTENTS OF TSSR              |          |  |
| 1936 | 036220 | 010001 |        |            | NEXT,ERRNO |                                                          |                                     |          |  |
| 1937 | 036222 |        |        |            | ERRDF      | ERRNO,T162SSR,PKTSSR                                     | ;DEVICE FATAL SSR FAILED TO SET     |          |  |
| 1938 | 036222 |        |        | 112\$:     |            |                                                          | TRAP                                | C\$ERDF  |  |
|      | 036224 | 001003 |        |            |            |                                                          | .WORD                               | 515      |  |
|      | 036226 | 036767 |        |            |            |                                                          | .WORD                               | T162SSR  |  |
|      | 036230 | 012046 |        |            |            |                                                          | .WORD                               | PKTSSR   |  |
| 1939 | 036232 | 005237 | 002222 |            | INC        | FATFLG                                                   | ;SET FATAL ERROR FLAG               |          |  |
| 1940 | 036236 |        |        | 120\$:     | CKLOOP     |                                                          | ;LOOP ON ERROR, IF FLAG SET         |          |  |
|      | 036236 | 104406 |        |            |            |                                                          | TRAP                                | C\$CLP1  |  |
| 1941 |        |        |        | :          |            | If Timer A NOT= 1 Then Print Error                       |                                     |          |  |
| 1942 |        |        |        | :          |            | If Timer B NOT= 1 Then Print Error                       |                                     |          |  |
| 1943 | 036240 | 005002 |        |            | CLR        | R2                                                       | ;INIT EXPD                          |          |  |
| 1944 | 036242 | 052702 | 000010 |            | BIS        | #S2,ATIM,R2                                              | ;TIMER A EXPD=1                     |          |  |
| 1945 | 036246 | 052702 | 000004 |            | BIS        | #S2,BTIM,R2                                              | ;TIMER B EXPD=1                     |          |  |
| 1946 | 036252 | 012700 | 040252 |            | MOV        | #T16BFSTA,R0                                             | ;GET RECV READ STATUS               |          |  |
| 1947 | 036256 | 016001 | 000002 |            | MOV        | 2(R0),R1                                                 | ;GET RECV BYTE 2                    |          |  |

F11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 5: SUBTEST 2: VERIFY TIMERS A,B

SEQ 135

|      |        |        |        |               |                                                          |                                     |       |          |
|------|--------|--------|--------|---------------|----------------------------------------------------------|-------------------------------------|-------|----------|
| 1948 | 036262 | 042701 | 177763 | BIC           | *C<S2.ATIM!S2.BTIM>,R1                                   | ;SAVE TIMER A:B RECV ONLY           |       |          |
| 1949 | 036266 |        |        | FORCERROR     | 172\$,NOTSSR                                             | ;000                                |       |          |
| 1950 | 036276 | 020201 |        | CMP           | R2,R1                                                    | ;EXPD EQUAL RECV?                   |       |          |
| 1951 | 036300 | 001404 |        | BEQ           | 180\$                                                    | ;BR IF YES                          |       |          |
| 1952 | 036302 |        |        | NEXT,ERRNO    |                                                          |                                     |       |          |
| 1953 | 036302 |        |        | 172\$: ERRHRD | ERRNO,T16T28,TIMEXP                                      | ;REPORT ERROR                       |       |          |
|      | 036302 | 104456 |        |               |                                                          |                                     | TRAP  | C\$ERHRD |
|      | 036304 | 001004 |        |               |                                                          |                                     | .WORD | 516      |
|      | 036306 | 037530 |        |               |                                                          |                                     | .WORD | T16T28   |
|      | 036310 | 015552 |        |               |                                                          |                                     | .WORD | TIMEXP   |
| 1954 | 036312 |        |        | 180\$: CKLOOP |                                                          | ;LOOP ON ERROR, IF FLAG SET         | SET   |          |
|      | 036312 | 104406 |        |               |                                                          |                                     | TRAP  | C\$CLP1  |
| 1955 |        |        |        |               |                                                          |                                     |       |          |
| 1956 |        |        |        | :             | Do a Write Control RESET TIMER with 53 microsecond delay |                                     |       |          |
| 1957 | 036314 | 012700 | 000001 | MOV           | *MS,RSD,R0                                               | ;RESET TIMER COMMAND                |       |          |
| 1958 | 036320 | 013701 | 036656 | MOV           | T16D53,R1                                                | ;53 MICROSECOND DELAY               |       |          |
| 1959 | 036324 | 004737 | 040142 | JSR           | PC,T16WMISC                                              | ;SETUP T16PK2 COMMAND PACKET        |       |          |
| 1960 | 036330 | 012704 | 040260 | MOV           | *T16PK2,R4                                               | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |          |
| 1961 | 036334 | 010465 | 000000 | MOV           | R4,TSDB(R5)                                              | ;SET THE PACKET ADDRESS TO EXECUTE  |       |          |
| 1962 | 036340 | 004737 | 016336 | JSR           | PC,CHKTSSR                                               | ;WAIT FOR SSR TO SET                |       |          |
| 1963 | 036344 |        |        | FORCERROR     | 212\$                                                    | ;000FORCE ERROR IF FORCER=1         |       |          |
| 1964 | 036360 | 103407 |        | BCS           | 220\$                                                    | ;BR IF CARRY SET (GOOD RETURN)      |       |          |
| 1965 | 036362 | 010001 |        | MOV           | R0,R1                                                    | ;SAVE CONTENTS OF TSSR              |       |          |
| 1966 | 036364 |        |        | NEXT,ERRNO    |                                                          |                                     |       |          |
| 1967 | 036364 |        |        | 212\$: ERROF  | ERRNO,T162SSR,PKTSSR                                     | ;DEVICE FATAL SSR FAILED TO SET     |       |          |
|      | 036364 | 104455 |        |               |                                                          |                                     | TRAP  | C\$EROF  |
|      | 036366 | 001005 |        |               |                                                          |                                     | .WORD | 517      |
|      | 036370 | 036767 |        |               |                                                          |                                     | .WORD | T162SSR  |
|      | 036372 | 012046 |        |               |                                                          |                                     | .WORD | PKTSSR   |
| 1968 | 036374 | 005237 | 002222 | INC           | FATFLG                                                   | ;SET FATAL ERROR FLAG               |       |          |
| 1969 | 036400 |        |        | 220\$: CKLOOP |                                                          | ;LOOP ON ERROR, IF FLAG SET         | SET   |          |
|      | 036400 | 104406 |        |               |                                                          |                                     | TRAP  | C\$CLP1  |
| 1970 |        |        |        | :             | If Timer A NOT= 0 Then Print Error                       |                                     |       |          |
| 1971 |        |        |        | :             | If Timer B NOT= 1 Then Print Error                       |                                     |       |          |
| 1972 | 036402 | 005002 |        | CLR           | R2                                                       | ;INIT EXPD                          |       |          |
| 1973 | 036404 | 042702 | 000010 | BIC           | *S2.ATIM,R2                                              | ;TIMER A EXPD=0                     |       |          |
| 1974 | 036410 | 052702 | 000004 | BIS           | *S2.BTIM,R2                                              | ;TIMER B EXPD=1                     |       |          |
| 1975 | 036414 | 012700 | 040152 | MOV           | *T16BFSTA,R0                                             | ;GET RECV READ STATUS               |       |          |
| 1976 | 036420 | 016001 | 000002 | MOV           | 2(R0),R1                                                 | ;GET RECV BYTE 2                    |       |          |
| 1977 | 036424 | 042701 | 177763 | BIC           | *C<S2.ATIM!S2.BTIM>,R1                                   | ;SAVE TIMER A:B RECV ONLY           |       |          |
| 1978 | 036430 |        |        | FORCERROR     | 272\$,NOTSSR                                             | ;000                                |       |          |
| 1979 | 036440 | 020201 |        | CMP           | R2,R1                                                    | ;EXPD EQUAL RECV?                   |       |          |
| 1980 | 036442 | 001404 |        | BEQ           | 280\$                                                    | ;BR IF YES                          |       |          |
| 1981 | 036444 |        |        | NEXT,ERRNO    |                                                          |                                     |       |          |
| 1982 | 036444 |        |        | 272\$: ERRHRD | ERRNO,T16T53,TIMEXP                                      | ;REPORT ERROR                       |       |          |
|      | 036444 | 104456 |        |               |                                                          |                                     | TRAP  | C\$ERHRD |
|      | 036446 | 001006 |        |               |                                                          |                                     | .WORD | 518      |
|      | 036450 | 037630 |        |               |                                                          |                                     | .WORD | T16T53   |
|      | 036452 | 015552 |        |               |                                                          |                                     | .WORD | TIMEXP   |
| 1983 | 036454 |        |        | 280\$: CKLOOP |                                                          | ;LOOP ON ERROR, IF FLAG SET         | SET   |          |
|      | 036454 | 104406 |        |               |                                                          |                                     | TRAP  | C\$CLP1  |
| 1984 |        |        |        | :             | Do a Write Control RESET TIMER with 78 microsecond delay |                                     |       |          |
| 1985 | 036456 | 012700 | 000001 | MOV           | *MS,RSD,R0                                               | ;RESET TIMER COMMAND                |       |          |
| 1986 | 036462 | 013701 | 036660 | MOV           | T16D78,R1                                                | ;78 MICROSECOND DELAY               |       |          |
| 1987 | 036466 | 004737 | 040142 | JSR           | PC,T16WMISC                                              | ;SETUP T16PK2 COMMAND PACKET        |       |          |
| 1988 | 036472 | 012704 | 040260 | MOV           | *T16PK2,R4                                               | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |          |
| 1989 | 036476 | 010465 | 000000 | MOV           | R4,TSDB(R5)                                              | ;SET THE PACKET ADDRESS TO EXECUTE  |       |          |

G11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 2: VERIFY TIMERS A,B

SEQ 136

```

1990 036502 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
1991 036506                      FORCERROR      312$      ;###FORCE ERROR IF FORCER=1
1992 036522 103407      BCS      320$      ;BR IF CARRY SET (GOOD RETURN)
1993 036524 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
1994 036526                      NEXT,ERRNO
1995 036526 312$:      ERRDF      ERRNO,T162SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                      TRAP      C$ERDF
                      .WORD      519
                      .WORD      T162SSR
                      .WORD      PKTSSR
                      036526 104455
                      036530 001007
                      036532 036767
                      036534 012046
1996 036536 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
1997 036542 320$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                      TRAP      C$CLP1
                      036542 104406
1998      ;      If Timer A NOT= 1 Then Print Error
1999      ;      If Timer B NOT= 0 Then Print Error
2000 036544 005002      CLR      R2      ;INIT EXPD
2001 036546 052702 000010      BIS      $S2,ATIM,R2      ;TIMER A EXPD=1
2002 036552 042702 000004      BIC      $S2,BTIM,R2      ;TIMER B EXPD=0
2003 036556 012700 040252      MOV      $T16BFSTA,R0      ;GET RECV READ STATUS
2004 036562 016001 000002      MOV      2(R0),R1      ;GET RECV BYTE 2
2005 036566 042701 177763      BIC      $+C<$S2,ATIM:$S2,BTIM>,R1      ;SAVE TIMER A:B RECV ONLY
2006 036572                      FORCERROR      372$,NOTSSR      ;###
2007 036602 020201      CMP      R2,R1      ;EXPD EQUAL RECV?
2008 036604 001404      BEQ      380$      ;BR IF YES
2009 036606                      NEXT,ERRNO
2010 036606 372$:      ERRHRD      ERRNO,T16T78,TIMEXP      ;REPORT ERROR
                      TRAP      C$ERHRD
                      .WORD      520
                      .WORD      T16T78
                      .WORD      TIMEXP
2011 036616 380$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
2012 036616 104406                      TRAP      C$CLP1
2013 036620                      ENDSUB
2014 036620                      ;///////////////// END SUBTEST ///////////////////
2015 036620 104403                      L10055:      TRAP      C$ESUB
2016 036622 005737 002222      TST      FATFLG      ;ANY FATAL ERRORS ?
2017 036626 001402      BEQ      460$      ;BRANCH IF NOT
2018 036630 004737 017202      JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
2019 036634 004737 016456 460$:      JSR      PC,TSTLOOP      ;SHOULD WE DO ITERATIONS?
2020 036640 103002      BCC      465$      ;BR IF NO
2021 036642 000137 034622      JMP      T16LOOP      ;LOOP UNTIL ITERATIONS DONE
2022 036646 465$:
2023
2024 036646      EXIT      TST      ;///////////////// EXIT TEST ///////////////////
2025 036646 104432                      TRAP      C$EXIT
2026 036650 001524                      .WORD      L10053-
2027
2028      ;*
2029      ;* LOCAL STORAGE FOR THIS TEST
2030 036652 000001      T16D01:      .WORD      1      ;1 MICROSECOND DELAY (ACTUALLY .8 MIC)
2031 036654 000040      T16D28:      .WORD      40      ;28 MICROSECOND DELAY (.8 MICROS PER)
2032 036656 000076      T16D53:      .WORD      76      ;53 MICROSECOND

```

H11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 2: VERIFY TIMERS A,B

SEQ 137

```

2033 036660 000142      T16D78:      .WORD 142      ;78 MICROSECOND
2034                      ;+
2035                      ;LOCAL TEXT MESSAGES FOR TEST
2036                      ;-
2037
2038 036662      105      170      164  TST16ID:      .ASCIIZ 'Extended Features Switch and Timers A,B'
2039 03732      127      122      111  T16SSR: .ASCIIZ 'WRITE CHARACTERISTICS Failed'
2040 036767      127      122      111  T162SSR: .ASCIIZ 'WRITE SUBSYSTEM (Write Misc) Failed'
2041 037033      127      122      111  T163SSR: .ASCIIZ 'WRITE SUBSYSTEM (Read Status) Failed'
2042 037100      102      165      163  T16TSBA: .ASCIIZ 'Bus Address Register (TSBA) Incorrect after Write Characteristics'
2043 037202      104      141      164  T16LEN: .ASCIIZ 'Data Field Length in Message Buffer Incorrect after Write Characteristics'
2044 037314      124      123      123  T16REJ: .ASCIIZ 'TSSR Function Reject Not Returned When Non-Existent Buffer Address Specified'
d'
2045 037431      124      151      155  T16T01: .ASCIIZ 'Timer A,B Incorrect after Reset Timer with 1 microsecond Delay'
2046 037530      124      151      155  T16T28: .ASCIIZ 'Timer A,B Incorrect after Reset Timer with 28 microsecond Delay'
2047 037630      124      151      155  T16T53: .ASCIIZ 'Timer A,B Incorrect after Reset Timer with 53 microsecond Delay'
2048 037730      124      151      155  T16T78: .ASCIIZ 'Timer A,B Incorrect after Reset Timer with 78 microsecond Delay'
2049                      .EVEN
2050
2051                      ;+
2052                      ;SET DEFAULT PACKET
2053                      ;-
2054 040030
2055 040030      012700      040210      MOV      #T16PACKET,R0      ;PACKET ADDRESS
2056 040034      012720      100004      MOV      #100004,(R0)+      ;WRITE CHARACTERISTICS WITH ACK
2057 040040      012720      040220      MOV      #T16DATA,(R0)+      ;ADDRESS OF CHAR DATA BLOCK
2058 040044      005020                      CLR      (R0)+      ;EXTENDED ADDRESS
2059 040046      012720      000012      MOV      #10,(R0)+      ;SIZE OF MESSAGE PACKET
2060 040052      012720      040232      MOV      #T16BFR,(R0)+      ;MESSAGE BUFFER ADDRESS
2061 040056      005020                      CLR      (R0)+      ;CLEAR EXTENDED BUFFER ADDRESS
2062 040060      012720      000024      MOV      #20,(R0)+      ;LENGTH OF MESSAGE BUFFER
2063 040064      005020                      CLR      (R0)+      ;CLEAR ESS,ENB,EAI,ERI
2064 040066      005010                      CLR      (R0)      ;CLEAR EXTENDED FEATURES WORD
2065 040070      005037      040232      CLR      T16BFR      ;CLEAR 1ST LOCATION IN MESSAGE BUFFER
2066 040074      000207                      RTS      PC      ;
2067
2068                      ;+
2069                      ;CLEAR MESSAGE BUFFER
2070                      ;-
2071 040076
2072 040076
2073 040102      012701      040232      SAVREG                      ;SAVE R1-R5 UNTIL NEXT RETURN
2074 040106      012702      000026      MOV      #T16BFR,R1      ;GET MESSAGE BUFFER ADDRESS
2075 040112      105021      MOV      #T16BEND-T16BFR,R2      ;SIZE OF MESSAGE BUFFER IN BYTES
2076 040114      005302      10$:      CLRB      (R1)+      ;CLEAR A BYTE
2077 040116      003375      DEC      R2      ;DONE?
2078 040120      000207      BGT      10$      ;BR IF NO
2079                      RTS      PC      ;RETURN
2080
2081                      ;+
2082                      ;SETUP T16PK2 PACKET FOR READ STATUS
2083                      ;-
2083 040122
2084 040122      004737      040076      JSR      PC,T16CLRBUF      ;CLEAR MESSAGE BUFFER
2085 040126      012700      040270      MOV      #T16DT2,R0      ;WRITE SUBSYSTEM DATA BUFFER
2086 040132      112720      000005      MOVWB      #PW,RDSTATUS,(R0)+      ;STORE READ STATUS COMMAND IN BSEL0
2087 040136      105010      CLRB      (R0)      ;CLEAR BSEL1
2088 040140      000207      RTS      PC      ;RETURN
2089

```

```

2090
2091
2092      ;+
2093      ; SETUP T16PK2 PACKET FOR WRITE MISC.
2094      ;
2095      ; INPUT:
2096      ;      R0      CONTAINS WRITE MISC FUNCTION CODE (BSEL1)
2097      ;      R1      CONTAINS DELAY (TIMES 800 NS) FOR BSEL2
2098      ;-
2098 040142 T16WMISC:
2099 040142      SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
2100 040146      JSR      PC,T16CLRBUF                ;CLEAR MESSAGE BUFFER
2101 040152      MOV      #T16DT2,R2                 ;WRITE SUBSYSTEM DATA BUFFER
2102 040156      MOV      #PW.WMISC,(R2)+             ;STORE WRITE MISCELLANEOUS IN BSEL0
2103 040162      MOV      R0,(R2)+                    ;STORE WRITE MISC CODE IN BSEL1
2104 040164      MOV      R1,(R2)+                    ;STORE DELAY (RESET TIMER) IN BSEL2
2105 040166      MOV      PC,R1                       ;RETURN
2106
2107      ;+
2108      ; SETUP T16PK2 PACKET FOR WRITE MISC. INVERT EXTENDED FEATURES SWITCH
2109      ;-
2109 040170 T16SEXT:
2110 040170      MOV      #T16DT2,R0                 ;WRITE SUBSYSTEM DATA BUFFER
2111 040174      MOV      #PW.WMISC,(R0)+             ;STORE WRITE MISCELLANEOUS IN BSEL0
2112 040200      MOV      #MS.EXT,(R0)+               ;STORE INVERT EXTENDED FEATURES IN BSEL1
2113 040204      MOV      PC,R1                       ;RETURN
2114
2115
2116
2117
2119      040210      .-<+.10>E177770
2121
2122      ;
2123      ;WRITE CHARACTERISTICS COMMAND PACKET
2124      ;
2124 040210 T16PACKET:
2125 040210      .WORD      100004                    ;COMMAND PACKET FOR TEST
2126 040212      .WORD      T16DATA                   ;WRITE CHARACTERISTICS COMMAND, WITH ACK
2127 040214      .WORD      0                          ;ADDRESS OF CHARACTERISTICS BLOCK
2128 040216      .WORD      10.                       ;MESSAGE PACKET SIZE
2129
2130 040220 T16DATA:
2131 040220      .WORD      T16BFR                     ;CHARACTERISTICS DATA BLOCK
2132 040222      .WORD      0                          ;ADDRESS OF MESSAGE BUFFER
2133 040224      .WORD      20.                       ;LENGTH OF MESSAGE BUFFER
2134 040226      .WORD      0                          ;ESS,ENB,EAI,ERI
2135 040230      .WORD      0                          ;EXTENDED FEATURES WORD
2136
2137
2138      ;MESSAGE BUFFER
2139
2140 040232 T16BFR:
2141 040232      .WORD      0                          ;BEGIN MESSAGE BUFFER
2142 040234      .WORD      0                          ;MESSAGE TYPE
2143 040236      .WORD      0                          ;DATA FIELD LENGTH
2144 040240      .WORD      0                          ;RBCR
2145 040242      .WORD      0                          ;XST0
2146 040244      .WORD      0                          ;XST1
2147 040246      .WORD      0                          ;XST2
2148 040250      .WORD      0                          ;XST3
2148 040250      .WORD      0                          ;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM

```

J11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 5: SUBTEST 2: VERIFY TIMERS A,8

SEQ 139

```

2149 040252      T16BFSTA: .BLKB 6.                ;READ STATUS AND WRITE FIFO BUFFER
2150 040260      T16BEND:                          ;END OF MESSAGE BUFFER
2151              ;
2152              ;WRITE SUBSYSTEM READ STATUS COMMAND PACKET
2153              ;
2157 040260      T16PK2:
2158 040260      100006          .WORD    P.WRTSUB!P.ACK        ;WRITE SUBSYSTEM WITH ACK
2159 040262      040270          .WORD    T16DT2               ;LOW ADDRESS OF DATA BLOCK
2160 040264      000000          .WORD    0                   ;HIGH ADDRESS OF DATA BLOCK
2161 040266      000012          .WORD    10.                 ;MINIMUM MESSAGE PACKET SIZE
2162
2163 040270      T16DT2:                          ;DATA BLOCK
2164 040270           000          .BYTE    0                   ;BSEL0
2165 040271           000          .BYTE    0                   ;BSEL1
2166 040272      000000          .WORD    0                   ;SEL2
2167 040274          .BLKB      64.                 ;WRITE FIFO DATA OUTPUT BUFFER
2168
2169
2170 040374          ENDTST1
                                L10053:
                                TRAP      C$ETST

```

K11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: FIFO EXERCISER

SEQ 140

```

2172 .SBYTL TEST 6: FIFO EXERCISER
2173 ;**
2174 ; TEST DESCRIPTION:
2175 ;
2176 ; This test uses the Write Subsystem Memory command to
2177 ; verify the controller's FIFO and associated status and
2178 ; control logic.
2179 ;
2180 ; TEST STEPS:
2181 ;
2182 ; REPEAT FOR LOOPCNT
2183 ; BEGIN
2184 ; Do Subtest 1 - FIFO Initialize status test
2185 ; Do Subtest 2 - FIFO Write Single Byte test
2186 ; Do Subtest 3 - FIFO Write Multiple Bytes test
2187 ; Do Subtest 4 - FIFO Verify ILW Status test
2188 ; Do Subtest 5 - FIFO Input Ready test
2189 ; Do Subtest 6 - FIFO Verify Reset FIFO test
2190 ; END
2191 ;--
2192
2193
2194 040376 BGNTST
2195 040376
2196 040376 012700 046026
2197 040402 004737 016510
2198 040406 012737 0000'2 002216
2199 040414 004737 017274
2200 040420 005037 003134
2201 040424
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231

```

```

MOV #TST17ID,RO ;ASCII MESSAGE TO IDENTIFY TEST
JSR PC,TSTSETUP ;DO INITIAL TEST SETUP
MOV #10,LOOPCNT ;PERFORM 10 ITERATIONS
JSR PC,KTOFF ;SHUT OFF MEMORY MANAGEMENT
CLR KTENABLE ;REALLY SHUT DOWN KT-11

```

```

T17LOOP:

.SBYTL TEST 6: SUBTEST 1: FIFO INITIALIZE STATUS TEST
;**
; TEST 6: SUBTEST 1:
;
; SUBTEST DESCRIPTION:
;
; This test verifies, by using the Read Status select code,
; that the FIFO status is in the correct initial state after
; the controller is initialized (Input Ready TRUE,
; Output Ready and Data In Miss FALSE). These status
; signals are checked by the controller's self-test
; sequence, so this subtest is actually more of a partial
; check of the Read Status function than the FIFO status.
;
; TEST STEPS:
;
; BEGIN
; Write to TSSR to soft initialize
; Do a WRITE CHARACTERISTICS to setup a message buffer
; Do a WRITE SUBSYSTEM Read Status
; If Input Ready NOT=1 Then Print Error
; If Output Ready NOT=0 Then Print Error
; If Data In Miss NOT=0 Then Print Error
; END

```

L11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 1: FIFO INITIALIZE STATUS TEST

SEQ 141

```

2232          ;--
2233 040424      BGNSUB                      ;////////// BEGIN SUBTEST ///////////
          040424      T6.1: TRAP C$BSUB
          040424      104402
2234
2235          ; Write to TSSR register to soft initialize the controller
2236 040426      5$: JSR PC,SOFINIT          ;WRITE TO TSSR TO SOFT INITIALIZE
2237 040426      004737 015774      BCS 10$      ;BR IF SOFT INIT OKAY
2238 040432      103405      MOV R0,R1          ;SAVE CONTENTS OF TSSR
2239 040434      010001      ERRDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
2240 040436      104455      TRAP C$ERDF
          040440      001130      .WORD 600
          040442      003652      .WORD SFIERR
          040444      012034      .WORD SFIMSG
2241          ; Do a WRITE CHARACTERISTICS to setup a message buffer
2242 040446      005037 002222      10$: CLR FATFLG          ;CLEAR FATAL ERROR FLAG
2243 040452      012704 050220      MOV #T17PK2,R4          ;GET THE ADDRESS OF COMMAND PACKET
2244 040456      004737 010662      JSR PC,WRTCHR          ;DO WRITE CHARACTERISTICS COMMAND
2245 040462      103407      FORCERROR 42$      ;GOODFORCE ERROR IF FORCER=1
2246 040476      010001      BCS 50$      ;BR IF CARRY SET (GOOD RETURN)
2247 040500      010001      MOV R0,R1          ;SAVE CONTENTS OF TSSR
2248 040502      42$: NEXT,ERRNO
2249 040502      104455      ERRDF ERRNO,T17SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
          040504      001131      TRAP C$ERDF
          040506      046645      .WORD 601
          040510      012046      .WORD T17SSR
          040512      005237 002222      INC FATFLG          ;SET FATAL ERROR FLAG
2250 040516      104406      50$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
          040516      104406      TRAP C$CLP1
2252
2253          ; Do a Write Subsystem READ STATUS
2254 040520      004737 050004      JSR PC,T17SRD          ;SETUP PACKET FOR READ STATUS
2255 040524      012704 050370      MOV #T17PK2,R4          ;GET WRITE SUBSYSTEM COMMAND PACKET
2256 040530      010465 000000      MOV R4,TSD8(R5)          ;SET THE PACKET ADDRESS TO EXECUTE
2257 040534      004737 016336      JSR PC,CHKTSSR          ;WAIT FOR SSR TO SET
2258 040540      103407      FORCERROR 62$      ;GOODFORCE ERROR IF FORCER=1
2259 040554      010001      BCS 70$      ;BR IF CARRY SET (GOOD RETURN)
2260 040556      010001      MOV R0,R1          ;SAVE CONTENTS OF TSSR
2261 040560      62$: NEXT,ERRNO
2262 040560      104455      ERRDF ERRNO,T173SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
          040562      001132      TRAP C$ERDF
          040564      046746      .WORD 602
          040566      012046      .WORD T173SSR
          040570      005237 002222      INC FATFLG          ;SET FATAL ERROR FLAG
2263 040574      104406      70$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
          040574      104406      TRAP C$CLP1
2265          ; Set WORDS 0-7 of expd message buffer - to recv since not testing
2266 040576      004737 050166      JSR PC,T17SETEXP          ;SET WORDS 0-7 EXPD=RECV
2267 040602      012701 046422      MOV #T17EXS1A,R1          ;GET EXPECTED READ STATUS
2268 040606      012702 050262      MOV #T17BFSTA,R2          ;GET RECV READ STATUS
2269 040612      012221      MOV (R2)+,(R1)+          ;SET EXPD WORD #8 = RECV TEMP
2270 040614      011211      MOV (R2),(R1)          ;SET EXPD WORD #9 = RECV TEMP
2271 040616      052711 000020      BIS #S2.INRDY,(R1)          ;SET EXP INPUT READY = TRUE
2272 040622      042711 000040      BIC #S2.OTRDY,(R1)          ;SET EXP OUTPUT READY = FALSE

```

M11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 1: FIFO INITIALIZE STATUS TEST

SEQ 142

```

2273 040626 042711 000200      BIC      #S2.DIM,(R1)      ;SET EXP DATA IN MISS = FALSE
2274                          ; If Input Ready NOT=1 then Print Error
2275                          ; If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error
2276 040632 005000      CLR      R0      ;HIGH RECV ADDRESS FOR CKMSG2
2277 040634 012701 050242      MOV      #T17BFR,R1      ;LOW RECV ADDRESS FOR CKMSG2
2278 040640 012702 046402      MOV      #T17EXP,R2      ;EXPD ADDRESS
2279 040644 012703 000024      MOV      #20.,R3      ;NUMBER OF BYTES TO COMPARE
2280 040650 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
2281 040654      FORCERROR      82$,NOTSSR      ;880
2282 040664 103404      BCS      90$      ;BR IF YES
2283 040666      NEXT.ERRNO
2284 040666      82$: ERRHRD ERRNO,T171CMP,MSGSTAT ;REPORT ERROR
                                TRAP      C$ERHRD
                                .WORD     603
                                .WORD     T171CMP
                                .WORD     MSGSTAT
2285 040676      90$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
2286 040676 104406
2287 040700      ENDSUB      ;////////// END SUBTEST //////////
                                L10057:
                                TRAP      C$ESUB
2288 040700 104403
2289 040702 005737 002222      TST      FATEFLG      ;ANY FATAL ERRORS ?
2290 040706 001402      BEQ      160$      ;BRANCH IF NOT
2291 040710 004737 017202      JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
2292 040714      160$:
2293
2294      .SBTTL TEST 6: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST
2295
2296      ;**
2297      ; TEST 6: SUBTEST 2:
2298      ;
2299      ; SUBTEST DESCRIPTION:
2300      ;
2301      ; This subtest verifies the ability of the FIFO to correctly
2302      ; pass a single data byte from input to output. For each
2303      ; of 256 data values (0-377 octal) the following is done:
2304      ; 1. Initial FIFO status is checked
2305      ; 2. The Write FIFO function, specifying a count of
2306      ; one byte to be written is executed.
2307      ; 3. Read Status is executed and FIFO status is checked.
2308      ; 4. Read FIFO is executed and the data and final status
2309      ; is checked.
2310      ;
2311      ; TEST STEPS:
2312      ;
2313      ; BEGIN
2314      ; Write to TSSR to soft initialize
2315      ; Do a WRITE CHARACTERISTICS to setup a message buffer
2316      ; Do a Write Subsystem READ STATUS
2317      ; If Input Ready NOT=1 Then Print Error
2318      ; If Output Ready NOT=0 Then Print Error
2319      ; If Data In Miss NOT=0 Then Print Error
2320      ;
2321      ; REPEAT FOR DATA FROM 0 TO 377 OCTAL
2322      ; BEGIN

```

N11

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

SEQ 143

```

2323      ; Do a Write Subsystem WRITE NPR to set tape direction out
2324      ; Do a Write Subsystem WRITE FIFO with byte count equal to 1
2325      ; Do a Write Subsystem READ STATUS
2326      ; If Input Ready NOT=1 Then Print Error
2327      ; If Output Ready NOT=1 Then Print Error
2328      ; If Data In Miss NOT=0 Then Print Error
2329      ; Do Write Subsystem READ FIFO with byte count equal to 1
2330      ; If Data read from FIFO NOT= to Data sent Then Print Error
2331      ; Do a Write Subsystem READ STATUS
2332      ; If Input Ready NOT=1 Then Print Error
2333      ; If Output Ready NOT=0 Then Print Error
2334      ; If Data In Miss NOT=0 Then Print Error
2335      ; END
2336      ; END
2337      ;--
2338 040714      BGNSUB                      ;////////// BEGIN SUBTEST //////////
      040714      T6.2:                      TRAP      C$BSUB
      040714 104402
2339
2340      ; Write to TSSR register to soft initialize the controller
2341 040716      5$:
2342 040716 004737 015774      JSR      PC,SOFINIT          ;WRITE TO TSSR TO SOFT INITIALIZE
2343 040722 103405      BCS      10$                      ;BR IF SOFT INIT OKAY
2344 040724 010001      MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
2345 040726      ERRDF      ERRNO,SFIERR,SFIMSG          ;DEVICE FATAL DURING INIT
      040726 104455      TRAP      C$ERDF
      040730 001133      .WORD      603
      040732 003652      .WORD      SFIERR
      040734 012034      .WORD      SFIMSG
2346
2347 040736 005037 002222      ; Do a WRITE CHARACTERISTICS to setup a message buffer
2348 040742 012704 050220      10$:      CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
2349 040746 004737 010662      MOV      @T17PACKET,R4          ;GET THE ADDRESS OF COMMAND PACKET
2350 040752      FORCERROR 42$          ;DO WRITE CHARACTERISTICS COMMAND
2351 040766 103407      BCS      50$          ;BR IF CARRY SET (GOOD RETURN)
2352 040770 010001      MOV      R0,R1          ;SAVE CONTENTS OF TSSR
2353 040772      NEXT,ERRNO
2354 040772      42$:      ERRDF      ERRNO,T17SSR,PKTSSR          ;DEVICE FATAL SSR FAILED TO SET
      040772 104455      TRAP      C$ERDF
      040774 001134      .WORD      604
      040776 046645      .WORD      T17SSR
      041000 012046      .WORD      PKTSSR
2355 041002 005237 002227      INC      FATFLG          ;SET FATAL ERROR FLAG
2356 041006      50$:      CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      041006 104406      TRAP      C$CLP1
2357
2358 041010 004737 050004      ; Do a Write Subsystem READ STATUS
2359 041014 012704 050370      JSR      PC,T17SRD          ;SETUP PACKET FOR READ STATUS
2360 041020 010465 000000      MOV      @T17PK2,R4          ;GET WRITE SUBSYSTEM COMMAND PACKET
2361 041024 004737 016336      MOV      R4,TSDB(R5)          ;SET THE PACKET ADDRESS TO EXECUTE
2362 041030      JSR      PC,CHKTSSR          ;WAIT FOR SSR TO SET
2363 041044 103407      FORCERROR 62$          ;BR IF CARRY SET (GOOD RETURN)
2364 041046 010001      BCS      70$          ;BR IF CARRY SET (GOOD RETURN)
2365 041050      MOV      R0,R1          ;SAVE CONTENTS OF TSSR
2366 041050      62$:      NEXT,ERRNO
      041050 104455      ERRDF      ERRNO,T17SSR,PKTSSR          ;DEVICE FATAL SSR FAILED TO SET
      041052 001135      TRAP      C$ERDF
      .WORD      605

```

|      |        |        |        |        |            |                                                                 |       |                                     |
|------|--------|--------|--------|--------|------------|-----------------------------------------------------------------|-------|-------------------------------------|
|      | 041054 | 046746 |        |        |            |                                                                 | .WORD | T173SSR                             |
|      | 041056 | 012046 |        |        |            |                                                                 | .WORD | PKTSSR                              |
| 2367 | 041060 | 005237 | 002222 |        | INC        | FATFLG                                                          |       | ;SET FATAL ERROR FLAG               |
| 2368 | 041064 |        |        | 70\$:  | CKLOOP     |                                                                 |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 041064 | 104406 |        |        |            |                                                                 |       | TRAP C\$CLP1                        |
| 2369 |        |        |        | :      |            | Set WORDS 0-7 of expd message buffer = to rcv since not testing |       |                                     |
| 2370 | 041066 | 004737 | 050166 |        | JSR        | PC,T17SETEXP                                                    |       | ;SET WORDS 0-7 EXPD=RCV             |
| 2371 | 041072 | 012701 | 046422 |        | MOV        | #T17EXSTA,R1                                                    |       | ;GET EXPECTED READ STATUS           |
| 2372 | 041076 | 012702 | 050262 |        | MOV        | #T17BFSTA,R2                                                    |       | ;GET RCV READ STATUS                |
| 2373 | 041102 | 012221 |        |        | MOV        | (R2), (R1)                                                      |       | ;SET EXPD WORD #8 = RCV TEMP        |
| 2374 | 041104 | 011211 |        |        | MOV        | (R2), (R1)                                                      |       | ;SET EXPD WORD #9 = RCV TEMP        |
| 2375 | 041106 | 052711 | 000020 |        | BIS        | #S2.INRDY,(R1)                                                  |       | ;SET EXP INPUT READY= TRUE          |
| 2376 | 041112 | 042711 | 000040 |        | BIC        | #S2.OUTRDY,(R1)                                                 |       | ;SET EXP OUTPUT READY= FALSE        |
| 2377 | 041116 | 042711 | 000200 |        | BIC        | #S2.DIM,(R1)                                                    |       | ;SET EXP DATA IN MISS = FALSE       |
| 2378 |        |        |        | :      |            | If Input Ready NOT=1 then Print Error                           |       |                                     |
| 2379 |        |        |        | :      |            | If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error    |       |                                     |
| 2380 | 041122 | 005000 |        |        | CLR        | R0                                                              |       | ;HIGH RCV ADDRESS FOR CKMSG2        |
| 2381 | 041124 | 012701 | 050242 |        | MOV        | #T17BFR,R1                                                      |       | ;LOW RCV ADDRESS FOR CKMSG2         |
| 2382 | 041130 | 012702 | 046402 |        | MOV        | #T17EXP,R2                                                      |       | ;EXPD ADDRESS                       |
| 2383 | 041134 | 012703 | 000024 |        | MOV        | #20.,R3                                                         |       | ;NUMBER OF BYTES TO COMPARE         |
| 2384 | 041140 | 004737 | 011500 |        | JSR        | PC,CKMSG2                                                       |       | ;EXPD EQUAL RCV?                    |
| 2385 | 041144 |        |        |        | FORCERROR  | 82\$,NOTSSR                                                     |       | ;BAD                                |
| 2386 | 041154 | 103404 |        |        | BCS        | 90\$                                                            |       | ;BR IF YES                          |
| 2387 | 041156 |        |        |        | NEXT.ERRNO |                                                                 |       |                                     |
| 2388 | 041156 |        |        | 82\$:  | ERRHRD     | ERRNO,T171CMP,MSGSTAT                                           |       | ;REPORT ERROR                       |
|      | 041156 | 104456 |        |        |            |                                                                 | TRAP  | C\$ERHRD                            |
|      | 041160 | 001136 |        |        |            |                                                                 | .WORD | 606                                 |
|      | 041162 | 047165 |        |        |            |                                                                 | .WORD | T171CMP                             |
|      | 041164 | 012350 |        |        |            |                                                                 | .WORD | MSGSTAT                             |
| 2389 | 041166 |        |        | 90\$:  | CKLOOP     |                                                                 |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 041166 | 104406 |        |        |            |                                                                 | TRAP  | C\$CLP1                             |
| 2390 |        |        |        |        |            |                                                                 |       |                                     |
| 2391 |        |        |        |        |            | Repeat for DATA from 0 to 377                                   |       |                                     |
| 2392 | 041170 | 012737 | 000000 | 002312 | MOV        | #0,DATA                                                         |       | ;GET FIRST DATA                     |
| 2393 | 041176 |        |        | 100\$: |            |                                                                 |       | ;REPEAT LABEL                       |
| 2394 |        |        |        | :      |            | Do a Write Subsystem WRITE NPR to set tape direction out        |       |                                     |
| 2395 | 041176 | 012700 | 000100 |        | MOV        | #NPR.OUT,R0                                                     |       | ;SET TAPE DIRECTION OUT             |
| 2396 | 041202 | 004737 | 050046 |        | JSR        | PC,T17SNPR                                                      |       | ;SETUP T17PK2 FOR WRITE NPR         |
| 2397 | 041206 | 012704 | 050370 |        | MOV        | #T17PK2,R4                                                      |       | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 2398 | 041212 | 010465 | 000000 |        | MOV        | R4,TSDB(R5)                                                     |       | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 2399 | 041216 | 004737 | 016336 |        | JSR        | PC,CHKTSSR                                                      |       | ;WAIT FOR SSR TO SET                |
| 2400 | 041222 |        |        |        | FORCERROR  | 102\$                                                           |       | ;BADFORCE ERROR IF FORCER=1         |
| 2401 | 041236 | 103407 |        |        | BCS        | 105\$                                                           |       | ;BR IF CARRY SET (GOOD RETURN)      |
| 2402 | 041240 | 010001 |        |        | MOV        | R0,R1                                                           |       | ;SAVE CONTENTS OF TSSR              |
| 2403 | 041242 |        |        |        | NEXT.ERRNO |                                                                 |       |                                     |
| 2404 | 041242 |        |        | 102\$: | ERRDF      | ERRNO,T174SSR,PKTSSR                                            |       | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 041242 | 104455 |        |        |            |                                                                 | TRAP  | C\$ERDF                             |
|      | 041244 | 001137 |        |        |            |                                                                 | .WORD | 607                                 |
|      | 041246 | 047013 |        |        |            |                                                                 | .WORD | T174SSR                             |
|      | 041250 | 012046 |        |        |            |                                                                 | .WORD | PKTSSR                              |
| 2405 | 041252 | 005237 | 002222 |        | INC        | FATFLG                                                          |       | ;SET FATAL ERROR FLAG               |
| 2406 | 041256 |        |        | 105\$: | CKLOOP     |                                                                 |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 041256 | 104406 |        |        |            |                                                                 | TRAP  | C\$CLP1                             |
| 2407 |        |        |        | :      |            | Do a Write Subsystem WRITE FIFO with byte count equal to 1      |       |                                     |
| 2408 | 041260 | 012700 | 000001 |        | MOV        | #1,R0                                                           |       | ;WRITE 1 BYTE                       |
| 2409 | 041264 | 012701 | 002312 |        | MOV        | #DATA,R1                                                        |       | ;FIFO WRITE DATA ADDRESS            |
| 2410 | 041270 |        |        |        |            |                                                                 |       |                                     |

C12

ISVS - HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 6: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

SEQ 145

|      |        |        |        |            |                                                                  |                                     |       |         |
|------|--------|--------|--------|------------|------------------------------------------------------------------|-------------------------------------|-------|---------|
| 2411 | 041274 | 012704 | 050370 | MOV        | #T17PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |         |
| 2412 | 041300 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE  |       |         |
| 2413 | 041304 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                |       |         |
| 2414 | 041310 |        |        | FORCERROR  | 107:                                                             | ;SSDFORCE ERROR IF FORCER=1         |       |         |
| 2415 | 041324 | 103407 |        | BCS        | 110:                                                             | ;BR IF CARRY SET (GOOD RETURN)      |       |         |
| 2416 | 041326 | 010001 |        | MOV        | R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |       |         |
| 2417 | 041330 |        |        | NEXT,ERRNO |                                                                  |                                     |       |         |
| 2418 | 041330 |        |        | 107:       | ERRDF ERRNO,T175SSR,PKTSSR                                       | ;DEVICE FATAL SSR FAILED TO SET     |       |         |
|      | 041330 | 104455 |        |            |                                                                  |                                     | TRAP  | C#ERDF  |
|      | 041332 | 001140 |        |            |                                                                  |                                     | .WORD | 608     |
|      | 041334 | 047056 |        |            |                                                                  |                                     | .WORD | T175SSR |
|      | 041336 | 012046 |        |            |                                                                  |                                     | .WORD | PKTSSR  |
| 2419 | 041340 | 005237 | 002222 | INC        | FATFLG                                                           | ;SET FATAL ERROR FLAG               |       |         |
| 2420 | 041344 |        |        | 110:       | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         |       |         |
|      | 041344 | 104406 |        |            |                                                                  |                                     | TRAP  | C#CLP1  |
| 2421 |        |        |        |            |                                                                  |                                     |       |         |
| 2422 |        |        |        |            |                                                                  |                                     |       |         |
| 2423 | 041346 | 004737 | 050004 |            | Do a Write Subsystem READ STATUS                                 |                                     |       |         |
| 2424 | 041352 | 012704 | 050370 | JSR        | PC,T17SRD                                                        | ;SETUP PACKET FOR READ STATUS       |       |         |
| 2425 | 041356 | 010465 | 000000 | MOV        | #T17PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |         |
| 2426 | 041362 | 004737 | 016336 | MOV        | R4,TSD8(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE  |       |         |
| 2427 | 041366 |        |        | JSR        | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                |       |         |
| 2428 | 041402 | 103407 |        | FORCERROR  | 112:                                                             | ;SSDFORCE ERROR IF FORCER=1         |       |         |
| 2429 | 041404 | 010001 |        | BCS        | 120:                                                             | ;BR IF CARRY SET (GOOD RETURN)      |       |         |
| 2430 | 041406 |        |        | MOV        | R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |       |         |
| 2431 | 041406 |        |        | NEXT,ERRNO |                                                                  |                                     |       |         |
|      | 041406 | 104455 |        | 112:       | ERRDF ERRNO,T173SSR,PKTSSR                                       | ;DEVICE FATAL SSR FAILED TO SET     |       |         |
|      | 041410 | 001141 |        |            |                                                                  |                                     | TRAP  | C#ERDF  |
|      | 041412 | 046746 |        |            |                                                                  |                                     | .WORD | 609     |
|      | 041414 | 012046 |        |            |                                                                  |                                     | .WORD | T173SSR |
|      | 041416 | 005237 | 002222 |            |                                                                  |                                     | .WORD | PKTSSR  |
| 2432 | 041416 | 005237 | 002222 | INC        | FATFLG                                                           | ;SET FATAL ERROR FLAG               |       |         |
| 2433 | 041422 |        |        | 120:       | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         |       |         |
|      | 041422 | 104406 |        |            |                                                                  |                                     | TRAP  | C#CLP1  |
| 2434 |        |        |        |            |                                                                  |                                     |       |         |
| 2435 | 041424 | 004737 | 050166 |            | Set WORDS 0-7 of expd message buffer - to recv since not testing |                                     |       |         |
| 2436 | 041430 | 012701 | 046422 | JSR        | PC,T17SETEXP                                                     | ;SET WORDS 0-7 EXPD=RCV             |       |         |
| 2437 | 041434 | 012702 | 050262 | MOV        | #T17EXSTA,R1                                                     | ;GET EXPECTED READ STATUS           |       |         |
| 2438 | 041440 | 012221 |        | MOV        | #T17BFSTA,R2                                                     | ;GET RCV READ STATUS                |       |         |
| 2439 | 041442 | 011211 |        | MOV        | (R2), (R1)                                                       | ;SET EXPD WORD #8 = RCV TEMP        |       |         |
| 2440 | 041444 | 052711 | 000020 | MOV        | (R2), (R1)                                                       | ;SET EXPD WORD #9 = RCV TEMP        |       |         |
| 2441 | 041450 | 052711 | 000040 | BIS        | #S2.INRDY,(R1)                                                   | ;SET EXP INPUT READY= 1             |       |         |
| 2442 | 041454 | 042711 | 000200 | BIS        | #S2.OURDY,(R1)                                                   | ;SET EXP OUTPUT READY= 1            |       |         |
| 2443 |        |        |        | BIC        | #S2.DIH,(R1)                                                     | ;SET EXP DATA IN MISS = 0           |       |         |
| 2444 |        |        |        |            | If Input Ready NOT=1 then Print Error                            |                                     |       |         |
| 2445 | 041460 | 005000 |        |            | If Output Ready NOT=1 or Data in Miss NOT=0 Then Print Error     |                                     |       |         |
| 2446 | 041462 | 012701 | 050242 | CLR        | R0                                                               | ;HIGH RCV ADDRESS FOR CKMSG2        |       |         |
| 2447 | 041466 | 012702 | 046402 | MOV        | #T17BFR,R1                                                       | ;LOW RCV ADDRESS FOR CKMSG2         |       |         |
| 2448 | 041472 | 012703 | 000024 | MOV        | #T17EXP,R2                                                       | ;EXPD ADDRESS                       |       |         |
| 2449 | 041476 | 004737 | 011500 | MOV        | #20,R3                                                           | ;NUMBER OF BYTES TO COMPARE         |       |         |
| 2450 | 041502 |        |        | JSR        | PC,CKMSG2                                                        | ;EXPD EQUAL RCV?                    |       |         |
| 2451 | 041512 | 103404 |        | FORCERROR  | 132:,NOTSSR                                                      | ;SSDFORCE ERROR IF FORCER=1         |       |         |
| 2452 | 041514 |        |        | BCS        | 140:                                                             | ;BR IF YES                          |       |         |
| 2453 | 041514 |        |        | NEXT,ERRNO |                                                                  |                                     |       |         |
|      | 041514 | 104456 |        | 132:       | ERRHRD ERRNO,T173CMP,MSGSTAT                                     | ;REPORT ERROR                       |       |         |
|      | 041516 | 001142 |        |            |                                                                  |                                     | TRAP  | C#ERHRD |
|      | 041520 | 047343 |        |            |                                                                  |                                     | .WORD | #10     |
|      | 041522 | 012350 |        |            |                                                                  |                                     | .WORD | T173CMP |
|      |        |        |        |            |                                                                  |                                     | .WORD | MSGSTAT |

ISV5 - HARDWARE TESTS, MACRO M1113, 06-FEB-84, 17:14  
 TEST 6: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

SEQ 146

```

2454 041524 140$: CKLOOP ;LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
      041524 104406
2455
2456 ; Do Write Subsystem READ FIFO with byte count equal to 1
2457 041526 012700 000001 MOV #1,R0 ;SET READ BYTE COUNT
2458 041532 004737 050126 JSR PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
2459 041536 012704 050370 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
2460 041542 010465 000000 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
2461 041546 004737 016336 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2462 041552 FORCERROR 142$ ;GOODFORCE ERROR IF FORCER=1
2463 041566 103407 BCS 150$ ;BR IF CARRY SET (GOOD RETURN)
2464 041570 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
2465 041572 NEXT,ERRNO
2466 142$: ERRDF ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      041572 104455 TRAP C$ERDF
      041574 001143 .WORD 611
      041576 047122 .WORD T176SSR
      041600 012046 .WORD PKTSSR
2467 041602 005237 002222 INC FATFLG ;SET FATAL ERROR FLAG
2468 041606 150$: CKLOOP ;LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
      041606 104406
2469 ; Set WORDS 0-7 of expd message buffer = to recv since not testing
2470 041610 004737 050166 JSR PC,T17SETEXP ;SET WORDS 0-7 EXPD=RCV
2471 041614 012701 046422 MOV #T17EXSTA,R1 ;GET EXPECTED READ STATUS
2472 041620 012702 050262 MOV #T17BFSTA,R2 ;GET RCV READ STATUS
2473 041624 013721 002312 MOV DATA,(R1) ;SET EXPD WORD #8 = COUNT DATA
2474 041630 011211 MOV (R2),(R1) ;SET EXPD WORD #9 = RCV (NOT TESTING)
2475 ; If Data read from FIFO NOT= to Data sent Then Print Error
2476 ; The data is in WORD #8 of the message buffer
2477 041632 005000 CLR R0 ;HIGH RCV ADDRESS FOR CKMSG2
2478 041634 012701 050242 MOV #T17BFR,R1 ;LOW RCV ADDRESS FOR CKMSG2
2479 041640 012702 046402 MOV #T17EXP,R2 ;EXPD ADDRESS
2480 041644 011703 000022 MOV #18,R3 ;NUMBER OF BYTES TO COMPARE
2481 041650 004737 011500 JSR PC,CKMSG2 ;EXPD EQUAL RCV?
2482 041654 FORCERROR 152$,NOTSSR ;GOOD
2483 041664 103404 BCS 160$ ;BR IF YES
2484 041666 NEXT,ERRNO
2485 152$: ERRHRD ERRNO,T172CMP,MSGSUB ;REPORT ERROR
      041666 104456 TRAP C$ERRHRD
      041670 001144 .WORD 612
      041672 047247 .WORD T172CMP
      041674 013742 .WORD MSGSUB
2486 041676 160$: CKLOOP ;LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
      041676 104406
2487
2488 ; Do a Write Subsystem READ STATUS
2489 041700 004737 050004 JSR PC,T17SRD ;SETUP PACKET FOR READ STATUS
2490 041704 012704 050370 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
2491 041710 010465 000000 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
2492 041714 004737 016336 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2493 041720 FORCERROR 162$ ;GOODFORCE ERROR IF FORCER=1
2494 041734 103407 BCS 170$ ;BR IF CARRY SET (GOOD RETURN)
2495 041736 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
2496 041740 NEXT,ERRNO
2497 162$: ERRDF ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      041740 104455 TRAP C$ERDF
      041742 001145 .WORD 613

```

|      |        |        |        |        |                      |                                                                 |  |                                              |          |         |
|------|--------|--------|--------|--------|----------------------|-----------------------------------------------------------------|--|----------------------------------------------|----------|---------|
|      | 041744 | 046746 |        |        |                      |                                                                 |  |                                              | .WORD    | T173SSR |
|      | 041746 | 012046 |        |        |                      |                                                                 |  |                                              | .WORD    | PKTSSR  |
| 2498 | 041750 | 005237 | 002222 |        | INC                  | FATFLG                                                          |  | ;SET FATAL ERROR FLAG                        |          |         |
| 2499 | 041754 |        |        | 170\$: | CKLOOP               |                                                                 |  | ;LOOP ON ERROR, IF FLAG SET                  |          |         |
|      | 041754 | 104406 |        |        |                      |                                                                 |  | TRAP                                         | C\$CLP1  |         |
| 2500 |        |        |        | :      |                      | Set WORDS 0-7 of expd message buffer = to rcv since not testing |  |                                              |          |         |
| 2501 | 041756 | 004737 | 050166 |        | JSR                  | PC,T17SETEXP                                                    |  | ;SET WORDS 0-7 EXPD=RCV                      |          |         |
| 2502 | 041762 | 012701 | 046422 |        | MOV                  | *T17EXSTA,R1                                                    |  | ;GET EXPECTED READ STATUS                    |          |         |
| 2503 | 041766 | 012702 | 050262 |        | MOV                  | *T17BFSTA,R2                                                    |  | ;GET RCV READ STATUS                         |          |         |
| 2504 | 041772 | 012221 |        |        | MOV                  | (R2)+,(R1)+                                                     |  | ;SET EXPD WORD #8 = RCV TEMP                 |          |         |
| 2505 | 041774 | 011211 |        |        | MOV                  | (R2),(R1)                                                       |  | ;SET EXPD WORD #9 = RCV TEMP                 |          |         |
| 2506 | 041776 | 052711 | 000020 |        | BIS                  | *S2.INRDY,(R1)                                                  |  | ;SET EXP INPUT READY= 1                      |          |         |
| 2507 | 042002 | 042711 | 000040 |        | BIC                  | *S2.OUTRDY,(R1)                                                 |  | ;SET EXP OUTPUT READY= 0                     |          |         |
| 2508 | 042006 | 042711 | 000200 |        | BIC                  | *S2.DIM,(R1)                                                    |  | ;SET EXP DATA IN MISS = 0                    |          |         |
| 2509 |        |        |        | :      |                      | If Input Ready NOT=1 then Print Error                           |  |                                              |          |         |
| 2510 |        |        |        | :      |                      | If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error    |  |                                              |          |         |
| 2511 | 042012 | 005000 |        |        | CLR                  | R0                                                              |  | ;HIGH RCV ADDRESS FOR CKMSG2                 |          |         |
| 2512 | 042014 | 012701 | 050242 |        | MOV                  | *T17BFR,R1                                                      |  | ;LOW RCV ADDRESS FOR CKMSG2                  |          |         |
| 2513 | 042020 | 012702 | 046402 |        | MOV                  | *T17EXP,R2                                                      |  | ;EXPD ADDRESS                                |          |         |
| 2514 | 042024 | 012703 | 000024 |        | MOV                  | *20.,R3                                                         |  | ;NUMBER OF BYTES TO COMPARE                  |          |         |
| 2515 | 042030 | 004737 | 011500 |        | JSR                  | PC,CKMSG2                                                       |  | ;EXPD EQUAL RCV?                             |          |         |
| 2516 | 042034 |        |        |        | FORCEERROR           | 172\$,NOTSSR                                                    |  | ;###                                         |          |         |
| 2517 | 042044 | 103404 |        |        | BCS                  | 180\$                                                           |  | ;BR IF YES                                   |          |         |
| 2518 | 042046 |        |        |        | NEXT,ERRNO           |                                                                 |  |                                              |          |         |
| 2519 | 042046 |        |        | 172\$: | ERRHRD               | ERRNO,T174CMP,MSGSTAT                                           |  | ;REPORT ERROR                                |          |         |
|      | 042046 | 104456 |        |        |                      |                                                                 |  | TRAP                                         | C\$ERHRD |         |
|      | 042050 | 001146 |        |        |                      |                                                                 |  | .WORD                                        | 614      |         |
|      | 042052 | 047427 |        |        |                      |                                                                 |  | .WORD                                        | T174CMP  |         |
|      | 042054 | 012350 |        |        |                      |                                                                 |  | .WORD                                        | MSGSTAT  |         |
| 2520 | 042056 |        |        | 180\$: | CKLOOP               |                                                                 |  | ;LOOP ON ERROR, IF FLAG SET                  |          |         |
|      | 042056 | 104406 |        |        |                      |                                                                 |  | TRAP                                         | C\$CLP1  |         |
| 2521 | 042060 |        |        |        | FORCEXIT             | 205\$                                                           |  | ;###                                         |          |         |
| 2522 | 042070 | 005237 | 002312 |        | INC                  | DATA                                                            |  | ;GET NEXT TEST DATA                          |          |         |
| 2523 | 042074 | 023727 | 002312 | 000377 | CMP                  | DATA,*377                                                       |  | ;DONE 0 TO 377?                              |          |         |
| 2524 | 042102 | 101002 |        |        | BHI                  | 205\$                                                           |  | ;BR IF YES                                   |          |         |
| 2525 | 042104 | 000137 | 041176 |        | JMP                  | 100\$                                                           |  | ;DO ANOTHER TEST PATTERN                     |          |         |
| 2526 | 042110 |        |        | 205\$: |                      |                                                                 |  |                                              |          |         |
| 2527 |        |        |        |        |                      |                                                                 |  |                                              |          |         |
| 2528 | 042110 |        |        |        | ENDSUB               |                                                                 |  | ;/!!!!!!!!!!!!!! END SUBTEST /!!!!!!!!!!!!!! |          |         |
|      | 042110 |        |        |        |                      |                                                                 |  | L10060:                                      |          |         |
|      | 042110 | 104403 |        |        |                      |                                                                 |  | TRAP                                         | C\$ESUB  |         |
| 2529 |        |        |        |        |                      |                                                                 |  |                                              |          |         |
| 2530 | 042112 | 005737 | 002222 |        | TST                  | FATFLG                                                          |  | ;ANY FATAL ERRORS ?                          |          |         |
| 2531 | 042116 | 001402 |        |        | BEG                  | 260\$                                                           |  | ;BRANCH IF NOT                               |          |         |
| 2532 | 042120 | 004737 | 017202 |        | JSR                  | PC,CKDROP                                                       |  | ;TRY TO DROP THE UNIT                        |          |         |
| 2533 | 042124 |        |        | 260\$: |                      |                                                                 |  |                                              |          |         |
| 2534 |        |        |        |        |                      |                                                                 |  |                                              |          |         |
| 2535 |        |        |        |        | .SBTTL               | TEST 6: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST               |  |                                              |          |         |
| 2536 |        |        |        |        |                      |                                                                 |  |                                              |          |         |
| 2537 |        |        |        |        | **                   |                                                                 |  |                                              |          |         |
| 2538 |        |        |        |        | ; TEST 6: SUBTEST 3; |                                                                 |  |                                              |          |         |
| 2539 |        |        |        |        |                      |                                                                 |  |                                              |          |         |

```

2545      ; and byte counts from 2 to 64.
2546      ; 1. Initial FIFO status is checked
2547      ; 2. The Write FIFO function.
2548      ; 3. Read Status is executed and FIFO status is checked.
2549      ; 4. Read FIFO is executed and the data and final status
2550      ; is checked.
2551      ;
2552      ; TEST STEPS:
2553      ;
2554      ; BEGIN
2555      ; Write to TSSR to soft initialize
2556      ; Do a WRITE CHARACTERISTICS to setup a message buffer
2557      ; Do a Write Subsystem READ STATUS
2558      ; If Input Ready NOT=1 Then Print Error
2559      ; If Output Ready NOT=0 Then Print Error
2560      ; If Data In Miss NOT=0 Then Print Error
2561      ; If Last Word NOT=0 Then Print Error
2562      ; REPEAT FOR DATA 0 TO 377, 377 TO 0, FLOATING 1'S,0'S AND ALL 1'S/0'S
2563      ; REPEAT FOR BYTE COUNT 2 TO 64 DECIMAL
2564      ; BEGIN
2565      ; Do a Write Subsystem WRITE NPR to set tape direction out
2566      ; Do a Write Subsystem WRITE FIFO
2567      ; Do a Write Subsystem READ STATUS
2568      ; If Input Ready NOT=1 Then Print Error
2569      ; If Output Ready NOT=1 Then Print Error
2570      ; If Data In Miss NOT=0 Then Print Error
2571      ; If Last Word NOT=0 Then Print Error
2572      ; Do Write Subsystem READ FIFO
2573      ; If Data read from FIFO NOT= to Data sent Then Print Error
2574      ; Do a Write Subsystem READ STATUS
2575      ; If Input Ready NOT=1 Then Print Error
2576      ; If Output Ready NOT=0 Then Print Error
2577      ; If Data In Miss NOT=0 Then Print Error
2578      ; If Last Word NOT=0 Then Print Error
2579      ; END
2580      ; END
2581      ;
2582      ;-- RGNSUB ;//////////////// BEGIN SUBTEST //////////////////
2583      ; T6.3: TRAP C$BSUB
2584      ;
2585      ; Write to TSSR register to soft initialize the controller
2586      5$: JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
2587      ; BCS 10$ ;BR IF SOFT INIT OKAY
2588      ; MOV R0,R1 ;SAVE CONTENTS OF TSSR
2589      ; ERDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
2590      ; TRAP C$ERDF
2591      ; .WORD 614
2592      ; .WORD SFIERR
2593      ; .WORD SFIMSG
2594      ;
2595      ; Do a WRITE CHARACTERISTICS to setup a message buffer
2596      10$: CLR FATFLG ;CLEAR FATAL ERROR FLAG
2597      ; MOV #T17PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
2598      ; JSR PC,WRTCHR ;DO WRITE CHARACTERISTICS COMMAND
2599      ; FORCERROR 42$ ;FORCE ERROR IF FORCER=1
2600      ; BCS 50$ ;BR IF CARRY SET (GOOD RETURN)

```

| Line | Address | Offset | Label  | Instruction                                                      | Comment                             | Trap  | Word      |
|------|---------|--------|--------|------------------------------------------------------------------|-------------------------------------|-------|-----------|
| 2596 | 042200  | 010001 |        | MOV RO,R1                                                        | ;SAVE CONTENTS OF TSSR              |       |           |
| 2597 | 042202  |        |        | NEXT.ERRNO                                                       |                                     |       |           |
| 2598 | 042202  |        | 42\$:  | ERRDF ERRNO,T17SSR,PKTSSR                                        | ;DEVICE FATAL SSR FAILED TO SET     |       |           |
|      | 042202  | 104455 |        |                                                                  |                                     | TRAP  | C\$ERDF   |
|      | 042204  | 001147 |        |                                                                  |                                     | .WORD | 615       |
|      | 042206  | 046645 |        |                                                                  |                                     | .WORD | T17SSR    |
|      | 042210  | 012046 |        |                                                                  |                                     | .WORD | PKTSSR    |
| 2599 | 042212  | 005237 | 002222 | INC FATFLG                                                       | ;SET FATAL ERROR FLAG               |       |           |
| 2600 | 042216  |        | 50\$:  | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         | SET   |           |
|      | 042216  | 104406 |        |                                                                  |                                     | TRAP  | C\$CLP1   |
| 2601 |         |        | :      | Do a Write Subsystem READ STATUS                                 |                                     |       |           |
| 2602 | 042220  | 004737 | 050004 | JSR PC,T17SRD                                                    | ;SETUP PACKET FOR READ STATUS       |       |           |
| 2603 | 042224  | 012704 | 050370 | MOV #T17PK2,R4                                                   | ;GET WRITE SUBSYSTEM COMMAND PACKET |       |           |
| 2604 | 042230  | 010465 | 000000 | MOV R4,TSD8(R5)                                                  | ;SET THE PACKET ADDRESS TO EXECUTE  |       |           |
| 2605 | 042234  | 004737 | 016336 | JSR PC,CHKTSSR                                                   | ;WAIT FOR SSR TO SET                |       |           |
| 2606 | 042240  |        |        | FORCERROR 62\$                                                   | ;GOODFORCE ERROR IF FORCER=1        |       |           |
| 2607 | 042254  | 103407 |        | BCS 70\$                                                         | ;BR IF CARRY SET (GOOD RETURN)      |       |           |
| 2608 | 042256  | 010001 |        | MOV RO,R1                                                        | ;SAVE CONTENTS OF TSSR              |       |           |
| 2609 | 042260  |        |        | NEXT.ERRNO                                                       |                                     |       |           |
| 2610 | 042260  |        | 62\$:  | ERRDF ERRNO,T173SSR,PKTSSR                                       | ;DEVICE FATAL SSR FAILED TO SET     |       |           |
|      | 042260  | 104455 |        |                                                                  |                                     | TRAP  | C\$ERDF   |
|      | 042262  | 001150 |        |                                                                  |                                     | .WORD | 616       |
|      | 042264  | 046746 |        |                                                                  |                                     | .WORD | T173SSR   |
|      | 042266  | 012046 |        |                                                                  |                                     | .WORD | PKTSSR    |
| 2611 | 042270  | 005237 | 002222 | INC FATFLG                                                       | ;SET FATAL ERROR FLAG               |       |           |
| 2612 | 042274  |        | 70\$:  | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         | SET   |           |
|      | 042274  | 104406 |        |                                                                  |                                     | TRAP  | C\$CLP1   |
| 2613 |         |        | :      | Set WORDS 0-7 of expd message buffer = to recv since not testing |                                     |       |           |
| 2614 | 042276  | 004737 | 050166 | JSR PC,T17SETEXP                                                 | ;SET WORDS 0-7 EXPD=RCV             |       |           |
| 2615 | 042302  | 012701 | 046422 | MOV #T17EXSTA,R1                                                 | ;GET EXPECTED READ STATUS           |       |           |
| 2616 | 042306  | 012702 | 050262 | MOV #T17BFSTA,R2                                                 | ;GET RCV READ STATUS                |       |           |
| 2617 | 042312  | 012221 |        | MOV (R2)+,(R1)+                                                  | ;SET EXPD WORD #8 = RCV TEMP        |       |           |
| 2618 | 042314  | 011211 |        | MOV (R2),(R1)                                                    | ;SET EXPD WORD #9 = RCV TEMP        |       |           |
| 2619 | 042316  | 052711 | 000020 | BIS #S2.INRDY,(R1)                                               | ;SET EXP INPUT READY= 1             |       |           |
| 2620 | 042322  | 042711 | 000040 | BIC #S2.OTRDY,(R1)                                               | ;SET EXP OUTPUT READY= 0            |       |           |
| 2621 | 042326  | 042711 | 000200 | BIC #S2.DIM,(R1)                                                 | ;SET EXP DATA IN MISS = 0           |       |           |
| 2622 | 042332  | 042711 | 000100 | BIC #S2.ILW,(R1)                                                 | ;SET EXP LAST WORD (ILW)=0          |       |           |
| 2623 |         |        | :      | If Input Ready NOT=1 then Print Error                            |                                     |       |           |
| 2624 |         |        | :      | If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error     |                                     |       |           |
| 2625 |         |        | :      | If Last Word NOT=0 Then Print Error                              |                                     |       |           |
| 2626 | 042336  | 005000 |        | CLR RO                                                           | ;HIGH RCV ADDRESS FOR CKMSG2        |       |           |
| 2627 | 042340  | 012701 | 050242 | MOV #T17BFR,R1                                                   | ;LOW RCV ADDRESS FOR CKMSG2         |       |           |
| 2628 | 042344  | 012702 | 046402 | MOV #T17EXP,R2                                                   | ;EXPD ADDRESS                       |       |           |
| 2629 | 042350  | 012703 | 000024 | MOV #20.,R3                                                      | ;NUMBER OF BYTES TO COMPARE         |       |           |
| 2630 | 042354  | 004737 | 011500 | JSR PC,CKMSG2                                                    | ;EXPD EQUAL RCV?                    |       |           |
| 2631 | 042360  |        |        | FORCERROR 82\$,NOTSSR                                            | ;GOOD                               |       |           |
| 2632 | 042370  | 103404 |        | BCS 90\$                                                         | ;BR IF YES                          |       |           |
| 2633 | 042372  |        |        | NEXT.ERRNO                                                       |                                     |       |           |
| 2634 | 042372  |        | 82\$:  | ERRHRD ERRNO,T171CMP,MSGSTAT                                     | ;REPORT ERROR                       |       |           |
|      | 042372  | 104456 |        |                                                                  |                                     | TRAP  | C\$ERRHRD |
|      | 042374  | 001151 |        |                                                                  |                                     | .WORD | 617       |
|      | 042376  | 047165 |        |                                                                  |                                     | .WORD | T171CMP   |
|      | 042400  | 012350 |        |                                                                  |                                     | .WORD | MSGSTAT   |
| 2635 | 042402  |        | 90\$:  | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         | SET   |           |
|      | 042402  | 104406 |        |                                                                  |                                     | TRAP  | C\$CLP1   |
| 2636 |         |        |        |                                                                  |                                     |       |           |
| 2637 |         |        |        |                                                                  |                                     |       |           |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

SEQ 150

```

2638
2639
2640
2641
2642
2643 042404 012737 000001 002314
2644 042412
2645 042412 012737 000002 002310
2646 042420
2647
2648 042420 012700 000100
2649 042424 004737 050046
2650 042430 012704 050370
2651 042434 010465 000000
2652 042440 004737 016336
2653 042444
2654 042460 103407
2655 042462 010001
2656 042464
2657 042464
      042464 104455
      042466 001152
      042470 047013
      042472 012046
2658 042474 005237 002222
2659 042500
      042500 104406
2660
2661 042502 004737 050146
2662 042506 012701 046524
2663 042512 013702 002310
2664 042516 022737 000001 002314
2665 042524 001005
2666 042526 005000
2667 042530 110021
2668 042532 005200
2669 042534 005302
2670 042536 003374
2671 042540 022737 000002 002314
2672 042546 001006
2673 042550 012700 000377
2674 042554 110021
2675 042556 005300
2676 042560 005302
2677 042562 003374
2678 042564 022737 000003 002314
2679 042572 001005
2680 042574 012700 002752
2681 042600 112021
2682 042602 005302
2683 042604 003375
2684 042606
2685 042606 013700 002310
2686 042612 012701 046524
2687 042616 004737 050072
2688 042622 012704 050370
2689 042626 010465 000000

; REPEAT FOR BYTE COUNT 2 TO 64 DECIMAL
; TSTFLAG =1 FOR INCREMENT TEST PATTERN
;          =2 FOR DECREMENT TEST PATTERN
;          =3 FOR TSTBLK TABLE PATTERN
MOV      #1,TSTFLAG ;TEST PATTERN FLAG
95$: MOV      #2,COUNT ;GET FIRST BYTE COUNT
100$:
; Do a Write Subsystem WRITE NPR to set tape direction out
MOV      #NP.OUT,R0 ;SET TAPE DIRECTION OUT
JSR      PC,T17SNPR ;SETUP T17PK2 FOR WRITE NPR
MOV      #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
JSR      PC,CHKTSSR ;WAIT FOR SSR TO SET
FORCERROR 102$ ;GOODFORCE ERROR IF FORCER=1
BCS      105$ ;BR IF CARRY SET (GOOD RETURN)
MOV      R0,R1 ;SAVE CONTENTS OF TSSR
NEXT.ERRNO
102$: ERRDF ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP C$ERDF
                                .WORD 618
                                .WORD T174SSR
                                .WORD PKTSSR
105$: INC      FATFLG ;SET FATAL ERROR FLAG
        CKLOOP ;LOOP ON ERROR, IF FLAG SET
                                TRAP C$CLP1
; Do a Write Subsystem WRITE FIFO
JSR      PC,T17CLEXP ;CLEAR EXPD BUFFER
MOV      #T17WFDATA,R1 ;EXPD WRITE FIFO DATA BUFFER
MOV      COUNT,R2 ;TEST PATTERN SIZE
CMP      #1,TSTFLAG ;INCREMENT PATTERN THIS TIME THRU?
BNE      115$ ;BR IF NO
CLR      R0 ;INCREMENT TEST PATTERN
110$: MOV      R0,(R1)+ ;STORE INCREMENT TEST BYTE
        INC      R0 ;SET NEXT PATTERN
        DEC      R2 ;DONE?
        BGT      110$ ;BR IF NO
        CMP      #2,TSTFLAG ;DECREMENT PATTERN THIS TIME THRU?
        BNE      125$ ;BR IF NO
        MOV      #377,R0 ;DECREMENT TEST PATTERN
        MOV      R0,(R1)+ ;STORE DECREMENT TEST BYTE
        DEC      R0 ;SET NEXT PATTERN
        DEC      R2 ;DONE?
        BGT      120$ ;BR IF NO
        CMP      #3,TSTFLAG ;TSTBLK PATTERNS THIS TIME THRU?
        BNE      135$ ;BR IF NO
        MOV      #TSTBLK,R0 ;FLOAT 1'S/0'S ETC. TEST TABLE
        MOV      (R0)+,(R1)+ ;STORE A TSTBLK BYTE
        DEC      R2 ;DONE?
        BGT      130$ ;BR IF NO
135$: MOV      COUNT,R0 ;FIFO BYTE COUNT
        MOV      #T17WFDATA,R1 ;FIFO WRITE DATA ADDRESS
        JSR      PC,T17WFIF ;SETUP T17PK2 FOR WRITE FIFO
        MOV      #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
        MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE

```

stics'  
 pecifie

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

SEQ 151

```

2690 042632 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
2691 042636                FORCERROR      142$      ;###FORCE ERROR IF FORCER=1
2692 042652 103407                BCS      150$      ;BR IF CARRY SET (GOOD RETURN)
2693 042654 010001                MOV      R0,R1      ;SAVE CONTENTS OF TSSR
2694 042656                NEXT,ERRNO
2695 042656 142$:      ERRDF      ERRNO,T175SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                        TRAP      C$ERDF
                        .WORD      619
                        .WORD      T175SSR
                        .WORD      PKTSSR
                        042656 104455
                        042660 001153
                        042662 047056
                        042664 012046
2696 042666 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
2697 042672 150$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                        TRAP      C$CLP1
                        042672 104406
2698
2699      ;      Do a Write Subsystem READ STATUS
2700 042674 004737 050004      JSR      PC,T17SRD      ;SETUP PACKET FOR READ STATUS
2701 042700 012704 050370      MOV      #T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
2702 042704 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
2703 042710 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
2704 042714                FORCERROR      157$      ;###FORCE ERROR IF FORCER=1
2705 042730 103407                BCS      160$      ;BR IF CARRY SET (GOOD RETURN)
2706 042732 010001                MOV      R0,R1      ;SAVE CONTENTS OF TSSR
2707 042734                NEXT,ERRNO
2708 042734 157$:      ERRDF      ERRNO,T173SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
                        TRAP      C$ERDF
                        .WORD      620
                        .WORD      T173SSR
                        .WORD      PKTSSR
                        042734 104455
                        042736 001154
                        042740 046746
                        042742 012046
2709 042744 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
2710 042750 160$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                        TRAP      C$CLP1
                        042750 104406
2711
2712      ;      Set WORDS 0-7 of expd message buffer = to recv since not testing
2713 042752 004737 050166      JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RCV
2714 042756 012701 046422      MOV      #T17EXSTA,R1      ;GET EXPECTED READ STATUS
2715 042762 012702 050262      MOV      #T17BFSTA,R2      ;GET RCV READ STATUS
2716 042766 012221                MOV      (R2)+,(R1)+      ;SET EXPD WORD #8 = RCV TEMP
2717 042770 011211                MOV      (R2),(R1)      ;SET EXPD WORD #9 = RCV TEMP
2718 042772 052711 000020      BIS      #S2.INRDY,(R1)      ;SET EXP INPUT READY= 1
2719 042776 052711 000040      BIS      #S2.OTRDY,(R1)      ;SET EXP OUTPUT READY= 1
2720 043002 042711 000200      BIC      #S2.DIM,(R1)      ;SET EXP DATA IN MISS = 0
2721 043006 042711 000100      BIC      #S2.ILW,(R1)      ;SET EXP LAST WORD (ILW)=0
2722
2723      ;      If Input Ready NOT=1 then Print Error
2724 043012 005000                CLR      R0      ;HIGH RCV ADDRESS FOR CKMSG2
2725 043014 012701 050242      MOV      #T17BFR,R1      ;LOW RCV ADDRESS FOR CKMSG2
2726 043020 012702 046402      MOV      #T17EXP,R2      ;EXPD ADDRESS
2727 043024 012703 000024      MOV      #20,R3      ;NUMBER OF BYTES TO COMPARE
2728 043030 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RCV?
2729 043034                FORCERROR      162$,NOTSSR      ;###
2730 043044 103404                BCS      170$      ;BR IF YES
2731 043046                NEXT,ERRNO
2732 043046 162$:      ERHRD      ERRNO,T173CMP,MSGSTAT      ;REPORT ERROR
                        TRAP      C$ERHRD
                        .WORD      621
                        .WORD      T173CMP
                        .WORD      MSGSTAT
                        043046 104456
                        043050 001155
                        043052 047343
                        043054 012350

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

SEQ 152

```

2733 043056      170$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      043056 104406      TRAP      C$CLP1
2734
2735      ;      Do Write Subsystem READ FIFO
2736 043060 013700 002310      MOV      COUNT,R0      ;SET READ BYTE COUNT
2737 043064 004737 050126      JSR      PC,T17RFIF      ;SETUP T17PK2 FOR READ FIFO
2738 043070 012704 050370      MOV      @T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
2739 043074 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
2740 043100 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
2741 043104      FORCERROR 172$      ;GOOD FORCE ERROR IF FORCER=1
2742 043120 103407      BCS      180$      ;BR IF CARRY SET (GOOD RETURN)
2743 043122 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
2744 043124      NEXT,ERRNO
2745 043124      172$: ERRDF  ERRNO,T176SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      043124 104455      TRAP      C$ERDF
      043126 001156      .WORD      622
      043130 047122      .WORD      T176SSR
      043132 012046      .WORD      PKTSSR
2746 043134 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
2747 043140      180$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      043140 104406      TRAP      C$CLP1
2748
2749      ;      If Data read from FIFO NOT= to Data sent Then Print Error
2750 043142 005000      CLR      R0      ;HIGH RECV ADDRESS FOR CKMSG2
2751 043144 012702 046524      MOV      @T17WFDATA,R2      ;GET EXPECTED ADDRESS FOR CKMSG2
2752 043150 012701 050262      MOV      @T17BFSTA,R1      ;GET RECEIVED ADDRESS FOR CKMSG2
2753 043154 013703 002310      MOV      COUNT,R3      ;NUMBER OF BYTES TO COMPARE
2754 043160 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
2755 043164      FORCERROR 192$,NOTSSR      ;GOOD
2756 043174 103406      BCS      200$      ;BR IF YES
2757 043176      NEXT,ERRNO
2758 043176 013701 002310      192$: MOV      COUNT,R1      ;GET BYTE COUNT
2759 043202      ERRHRD  ERRNO,T175CMP,FIFEXP      ;REPORT ERROR
      043202 104456      TRAP      C$ERHRD
      043204 001157      .WORD      623
      043206 047512      .WORD      T175CMP
      043210 012170      .WORD      FIFEXP
2760 043212      200$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      043212 104406      TRAP      C$CLP1
2761
2762      ;      Do a Write Subsystem READ STATUS
2763 043214 004737 050004      JSR      PC,T17SRD      ;SETUP PACKET FOR READ STATUS
2764 043220 012704 050370      MOV      @T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
2765 043224 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
2766 043230 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
2767 043234      FORCERROR 212$      ;GOOD FORCE ERROR IF FORCER=1
2768 043250 103407      BCS      220$      ;BR IF CARRY SET (GOOD RETURN)
2769 043252 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
2770 043254      NEXT,ERRNO
2771 043254      212$: ERRDF  ERRNO,T173SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      043254 104455      TRAP      C$ERDF
      043256 001160      .WORD      624
      043260 046746      .WORD      T173SSR
      043262 012046      .WORD      PKTSSR
2772 043264 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
2773 043270      220$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      043270 104406      TRAP      C$CLP1

```

K12

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

SEQ 153

```

2774      ; Set WORDS 0-7 of expd message buffer = to recv since not testing
2775 043272 004737 050166      JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RECV
2776 043276 012701 046422      MOV      #T17EXSTA,R1      ;GET EXPECTED READ STATUS
2777 043302 012702 050262      MOV      #T17BFSTA,R2      ;GET RECV READ STATUS
2778 043306 012221              MOV      (R2)+,(R1)+      ;SET EXPD WORD #8 = RECV TEMP
2779 043310 011211              MOV      (R2),(R1)        ;SET EXPD WORD #9 = RECV TEMP
2780 043312 052711 000020      BIS      #S2.INRDY,(R1)      ;SET EXP INPUT READY= 1
2781 043316 042711 000040      BIC      #S2.OTRDY,(R1)      ;SET EXP OUTPUT READY= 0
2782 043322 042711 000200      BIC      #S2.DIM,(R1)      ;SET EXP DATA IN MISS = 0
2783 043326 042711 000100      BIC      #S2.ILW,(R1)      ;SET EXP LAST WORD (ILW)=0
2784      ; If Input Ready NOT=1 then Print Error
2785      ; If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error
2786 043332 005000              CLR      RO              ;HIGH RECV ADDRESS FOR CKMSG2
2787 043334 012701 050242      MOV      #T17BFR,R1      ;LOW RECV ADDRESS FOR CKMSG2
2788 043340 012702 046402      MOV      #T17EXP,R2      ;EXPD ADDRESS
2789 043344 012703 000024      MOV      #20,R3          ;NUMBER OF BYTES TO COMPARE
2790 043350 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
2791 043354              FORCEERROR      232$,NOTSSR      ;880
2792 043364 103404              BCS      240$          ;BR IF YES
2793 043366              NEXT.ERRNO
2794 043366      232$: ERRHRD ERRNO,T174CMP,MSGSTAT ;REPORT ERROR
                                TRAP      C$ERHRD
                                .WORD     625
                                .WORD     T174CMP
                                .WORD     MSGSTAT
2795 043376      240$: CKLOOP              ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
2796 043400              FORCEEXIT      250$          ;880
2797 043410 005237 002310      INC      COUNT      ;GET NEXT BYTE COUNT
2798 043414 023727 002310 000077      CMP      COUNT,#77      ;DONE 0 TO 77
2799 043422 101002              BHI      250$          ;BR IF YES
2800 043424 000137 042420      JMP      100$          ;DO ANOTHER BYTE COUNT
2801 043430 005237 002314      250$: INC      TSTFLAG      ;GET NEXT TEST PATTERN CODE
2802 043434 023727 002314 000003      CMP      TSTFLAG,#3      ;DONE INC,DEC,TSTRLK PATTERNS?
2803 043442 101002              BHI      255$          ;BR IF YES
2804 043444 000137 042412      JMP      95$           ;DO ANOTHER TEST PATTERN
2805 043450      255$:
2806 043450              ENDSUB              ;////////// END SUBTEST //////////
                                L10061:
                                TRAP      C$ESUB
2807 043452 005737 002222      TST      FATALFLG      ;ANY FATAL ERRORS ?
2808 043456 001402              BEQ      260$          ;BRANCH IF NOT
2809 043460 004737 017202      JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
2810 043464      260$:
2811
2812
2813
2814
2815      .SBTTL TEST 6: SUBTEST 4: FIFO Verify ILW Status
2816
2817      ;**
2818      ; TEST 6: SUBTEST 4:
2819      ;
2820      ; SUBTEST DESCRIPTION:
2821      ;
2822      ; This subtest verifies that reading the FIFO when it is
2823      ; empty causes the Last Word (ILW) status to assert.

```

```

2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837 043464
      043464
      043464 104402
2838
2839
2840 043466
2841 043466 004737 015774
2842 043472 103405
2843 043474 010001
2844 043476
      043476 104455
      043500 001161
      043502 003652
      043504 012034
2845
2846 043506 005037 002222
2847 043512 012704 050220
2848 043516 004737 010662
2849 043522
2850 043536 103407
2851 043540 010001
2852 043542
2853 043542
      043542 104455
      043544 001162
      043546 046645
      043550 012046
2854 043552 005237 002222
2855 043556
      043556 104406
2856
2857
2858 043560 012700 000001
2859 043564 004737 050126
2860 043570 012704 050370
2861 043574 010465 000000
2862 043600 004737 016336
2863 043604
2864 043620 103407
2865 043622 010001
2866 043624
2867 043624
      043624 104455
      043626 001163

; TEST STEPS:
; BEGIN
;   Write to TSSR to soft initialize
;   Do Write Subsystem READ FIFO with byte count equal to 1
;   Do a Write Subsystem READ STATUS
;   If Input Ready NOT=1 Then Print Error
;   If Output Ready NOT=0 Then Print Error
;   If Data In Miss NOT=0 Then Print Error
;   If Last Word (ILW) NOT=1 Then Print Error
; END
;--
      BGNSUB
;//////////////// BEGIN SUBTEST //////////////////
;T6.4:
;TRAP C$BSUB

;   Write to TSSR register to soft initialize the controller
5$:
      JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
      BCS 10$ ;BR IF SOFT INIT OKAY
      MOV R0,R1 ;SAVE CONTENTS OF TSSR
      ERDf ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
;TRAP C$ERDF
;WORD 625
;WORD SFIERR
;WORD SFIMSG

;   Do a WRITE CHARACTERISTICS to setup a message buffer
10$:
      CLR FATFLG ;CLEAR FATAL ERROR FLAG
      MOV #T17PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
      JSR PC,WRTCHR ;DO WRITE CHARACTERISTICS COMMAND
      FORCERROR 42$ ;DO FORCE ERROR IF FORCER=1
      BCS 50$ ;BR IF CARRY SET (GOOD RETURN)
      MOV R0,R1 ;SAVE CONTENTS OF TSSR
      NEXT,ERRNO
42$:
      ERDf ERRNO,T17SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
;TRAP C$ERDF
;WORD 626
;WORD T17SSR
;WORD PKTSSR

      INC FATFLG ;SET FATAL ERROR FLAG
50$:
      CKLOOP ;LOOP ON ERROR, IF FLAG SET
;TRAP C$CLP1

;   Do Write Subsystem READ FIFO with byte count equal to 1
      MOV #1,R0 ;SET READ BYTE COUNT
      JSR PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
      MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
      MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
      JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
      FORCERROR 142$ ;DO FORCE ERROR IF FORCER=1
      BCS 150$ ;BR IF CARRY SET (GOOD RETURN)
      MOV R0,R1 ;SAVE CONTENTS OF TSSR
      NEXT,ERRNO
142$:
      ERDf ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
;TRAP C$ERDF
;WORD 627

```

| Address | Offset | Label  | Operation | Comments   | Trap                       | Word    |
|---------|--------|--------|-----------|------------|----------------------------|---------|
| 2868    | 043630 | 005237 | 002222    | INC FATFLG | SET FATAL ERROR FLAG       | T176SSR |
| 2869    | 043632 | 005237 | 002222    | CKLOOP     | LOOP ON ERROR, IF FLAG SET | PKTSSR  |
| 2870    | 043634 | 005237 | 002222    |            |                            |         |
| 2871    | 043636 | 005237 | 002222    |            |                            |         |
| 2872    | 043638 | 005237 | 002222    |            |                            |         |
| 2873    | 043640 | 005237 | 002222    |            |                            |         |
| 2874    | 043642 | 005237 | 002222    |            |                            |         |
| 2875    | 043644 | 005237 | 002222    |            |                            |         |
| 2876    | 043646 | 005237 | 002222    |            |                            |         |
| 2877    | 043648 | 005237 | 002222    |            |                            |         |
| 2878    | 043650 | 005237 | 002222    |            |                            |         |
| 2879    | 043652 | 005237 | 002222    |            |                            |         |
| 2880    | 043654 | 005237 | 002222    |            |                            |         |
| 2881    | 043656 | 005237 | 002222    |            |                            |         |
| 2882    | 043658 | 005237 | 002222    |            |                            |         |
| 2883    | 043660 | 005237 | 002222    |            |                            |         |
| 2884    | 043662 | 005237 | 002222    |            |                            |         |
| 2885    | 043664 | 005237 | 002222    |            |                            |         |
| 2886    | 043666 | 005237 | 002222    |            |                            |         |
| 2887    | 043668 | 005237 | 002222    |            |                            |         |
| 2888    | 043670 | 005237 | 002222    |            |                            |         |
| 2889    | 043672 | 005237 | 002222    |            |                            |         |
| 2890    | 043674 | 005237 | 002222    |            |                            |         |
| 2891    | 043676 | 005237 | 002222    |            |                            |         |
| 2892    | 043678 | 005237 | 002222    |            |                            |         |
| 2893    | 043680 | 005237 | 002222    |            |                            |         |
| 2894    | 043682 | 005237 | 002222    |            |                            |         |
| 2895    | 043684 | 005237 | 002222    |            |                            |         |
| 2896    | 043686 | 005237 | 002222    |            |                            |         |
| 2897    | 043688 | 005237 | 002222    |            |                            |         |
| 2898    | 043690 | 005237 | 002222    |            |                            |         |
| 2899    | 043692 | 005237 | 002222    |            |                            |         |
| 2900    | 043694 | 005237 | 002222    |            |                            |         |
| 2901    | 043696 | 005237 | 002222    |            |                            |         |
| 2902    | 043698 | 005237 | 002222    |            |                            |         |
| 2903    | 043700 | 005237 | 002222    |            |                            |         |
| 2904    | 043702 | 005237 | 002222    |            |                            |         |
| 2905    | 043704 | 005237 | 002222    |            |                            |         |
| 2906    | 043706 | 005237 | 002222    |            |                            |         |
| 2907    | 043708 | 005237 | 002222    |            |                            |         |
| 2908    | 043710 | 005237 | 002222    |            |                            |         |
| 2909    | 043712 | 005237 | 002222    |            |                            |         |
| 2910    | 043714 | 005237 | 002222    |            |                            |         |
| 2911    | 043716 | 005237 | 002222    |            |                            |         |
| 2912    | 043718 | 005237 | 002222    |            |                            |         |
| 2913    | 043720 | 005237 | 002222    |            |                            |         |
| 2914    | 043722 | 005237 | 002222    |            |                            |         |
| 2915    | 043724 | 005237 | 002222    |            |                            |         |
| 2916    | 043726 | 005237 | 002222    |            |                            |         |
| 2917    | 043728 | 005237 | 002222    |            |                            |         |
| 2918    | 043730 | 005237 | 002222    |            |                            |         |
| 2919    | 043732 | 005237 | 002222    |            |                            |         |
| 2920    | 043734 | 005237 | 002222    |            |                            |         |
| 2921    | 043736 | 005237 | 002222    |            |                            |         |
| 2922    | 043738 | 005237 | 002222    |            |                            |         |
| 2923    | 043740 | 005237 | 002222    |            |                            |         |
| 2924    | 043742 | 005237 | 002222    |            |                            |         |
| 2925    | 043744 | 005237 | 002222    |            |                            |         |
| 2926    | 043746 | 005237 | 002222    |            |                            |         |
| 2927    | 043748 | 005237 | 002222    |            |                            |         |
| 2928    | 043750 | 005237 | 002222    |            |                            |         |
| 2929    | 043752 | 005237 | 002222    |            |                            |         |
| 2930    | 043754 | 005237 | 002222    |            |                            |         |
| 2931    | 043756 | 005237 | 002222    |            |                            |         |
| 2932    | 043758 | 005237 | 002222    |            |                            |         |
| 2933    | 043760 | 005237 | 002222    |            |                            |         |
| 2934    | 043762 | 005237 | 002       |            |                            |         |

N12

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 4: FIFO VERIFY ILW STATUS

SEQ 156

2910 044034 001402  
2911 044036 004737 017202  
2912 044042

260\$:

BEQ 260\$  
JSR PC,CKDROP

;BRANCH IF NOT  
;TRY TO DROP THE UNIT

2913

2914

2915

2916

2917

2918

2919

2920

2921

2922

2923

2924

2925

2926

2927

2928

2929

2930

2931

2932

2933

2934

2935

2936

2937

2938

2939

2940

2941

2942

2943

2944

2945

2946

2947

2948

2949

2950

2951

2952

2953 044042

044042

044042 104402

2954

2955

2956 044044

2957 044044 004737 015774

2958 044050 103405

2959 044052 010001

2960 044054

044054 104455

044056 001165

044060 003652

044062 012034

.SBTTL TEST 6: SUBTEST 5: FIFO Verify Input Ready

!--

; TEST 6: SUBTEST 5:

; SUBTEST DESCRIPTION:

This subtest verifies that writing 64. bytes into the FIFO without reading any out causes the Input Ready status to negate. The Subtest then verifies that writing a 65th byte into the FIFO causes the Data In Miss status to assert. Next it is verified that the original 64 bytes can be read out correctly and that the data has not been corrupted.

; TEST STEPS:

; BEGIN

Write to TSSR to soft initialize  
Do a WRITE CHARACTERISTICS to setup a message buffer  
Do a Write Subsystem WRITE NPR to set tape direction out  
Do a Write Subsystem WRITE FIFO 64. bytes incrementing pattern  
Do a Write Subsystem READ STATUS  
If Input Ready NOT=0 Then Print Error  
If Output Ready NOT=1 Then Print Error  
If Data In Miss NOT=0 Then Print Error  
Do a Write Subsystem WRITE FIFO 1 byte for a total of 65. written  
Do a Write Subsystem READ STATUS  
If Input Ready NOT=0 Then Print Error  
If Output Ready NOT=1 Then Print Error  
If Data In Miss NOT=1 Then Print Error  
Do Write Subsystem READ FIFO  
If Data read from FIFO NOT= to Data sent Then Print Error  
Do a Write Subsystem READ STATUS  
If Input Ready NOT=1 Then Print Error  
If Output Ready NOT=0 Then Print Error  
If Data In Miss NOT=1 Then Print Error

; END

!--

BGNSUB

;/;;;;;;;;; BEGIN SUBTEST /;;;;;;;;;  
T6.5: TRAP C\$BSUB

; Write to TSSR register to soft initialize the controller

5\$:

JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE  
BCS 10\$ ;BR IF SOFT INIT OKAY  
MOV R0,R1 ;SAVE CONTENTS OF TSSR  
ERRDF ERRNO,SFERR,SFMSG ;DEVICE FATAL DURING INIT

TRAP C\$ERRDF  
.WORD 629  
.WORD SFERR  
.WORD SFMSG

TSV5 - HARDWARE TESTS MACRO M1113 06 FEB-84 17:14  
TEST 6: SUBTEST 5: FIFO VERIFY INPUT READY

SEQ 157

|      |        |        |        |        |                                                                |                                     |
|------|--------|--------|--------|--------|----------------------------------------------------------------|-------------------------------------|
| 2961 |        |        |        | 1      | Do a WRITE CHARACTERISTICS to setup a message buffer           |                                     |
| 2962 | 044064 | 005037 | 002222 | 101:   | CLR FATELG                                                     | ;CLEAR FATAL ERROR FLAG             |
| 2963 | 044070 | 012704 | 050220 |        | MOV #T17PACKET,R4                                              | ;GET THE ADDRESS OF COMMAND PACKET  |
| 2964 | 044074 | 004737 | 010662 |        | JSR PC,WRCHR                                                   | ;DO WRITE CHARACTERISTICS COMMAND   |
| 2965 | 044100 |        |        |        | FORCERROR 42:                                                  | ;GOODFORCE ERROR IF FORCER=1        |
| 2966 | 044114 | 103407 |        |        | BCS 50:                                                        | ;BR IF CARRY SET (GOOD RETURN)      |
| 2967 | 044116 | 010001 |        |        | MOV R0,R1                                                      | ;SAVE CONTENTS OF TSSR              |
| 2968 | 044120 |        |        |        | NEXT,ERRNO                                                     |                                     |
| 2969 | 044120 |        |        | 42:    | ERRDF ERRNO,T17SSR,PKTSSR                                      | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 044120 | 104455 |        |        |                                                                | TRAP C1ERDF                         |
|      | 044122 | 001166 |        |        |                                                                | .WORD 630                           |
|      | 044124 | 046645 |        |        |                                                                | .WORD T17SSR                        |
|      | 044126 | 012046 |        |        |                                                                | .WORD PKTSSR                        |
| 2970 | 044130 | 005237 | 002222 |        | INC FATELG                                                     | ;SET FATAL ERROR FLAG               |
| 2971 | 044134 |        |        | 50:    | CKLOOP                                                         | ;LOOP ON ERROR, IF FLAG SET         |
|      | 044134 | 104406 |        |        |                                                                | TRAP C1CLP1                         |
| 2972 |        |        |        |        |                                                                |                                     |
| 2973 |        |        |        | 1      | Do a Write Subsystem WRITE NPR to set tape direction out       |                                     |
| 2974 | 044136 | 012700 | 000100 | 100:   | MOV #NP.OUT,R0                                                 | ;SET TAPE DIRECTION OUT             |
| 2975 | 044142 | 004737 | 050046 |        | JSR PC,T17SNPR                                                 | ;SETUP T17PK2 FOR WRITE NPR         |
| 2976 | 044146 | 012704 | 050370 |        | MOV #T17PK2,R4                                                 | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 2977 | 044152 | 010465 | 000000 |        | MOV R4,ISDB(R5)                                                | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 2978 | 044156 | 004737 | 016336 |        | JSR PC,CHKTSSR                                                 | ;WAIT FOR SSR TO SET                |
| 2979 | 044162 |        |        |        | FORCERROR 102:                                                 | ;GOODFORCE ERROR IF FORCER=1        |
| 2980 | 044176 | 103407 |        |        | BCS 105:                                                       | ;BR IF CARRY SET (GOOD RETURN)      |
| 2981 | 044200 | 010001 |        |        | MOV R0,R1                                                      | ;SAVE CONTENTS OF TSSR              |
| 2982 | 044202 |        |        |        | NEXT,ERRNO                                                     |                                     |
| 2983 | 044202 |        |        | 102:   | ERRDF ERRNO,T174SSR,PKTSSR                                     | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 044202 | 104455 |        |        |                                                                | TRAP C1ERDF                         |
|      | 044204 | 001167 |        |        |                                                                | .WORD 631                           |
|      | 044206 | 047013 |        |        |                                                                | .WORD T174SSR                       |
|      | 044210 | 012046 |        |        |                                                                | .WORD PKTSSR                        |
| 2984 | 044212 | 005237 | 002222 |        | INC FATELG                                                     | ;SET FATAL ERROR FLAG               |
| 2985 | 044216 |        |        | 105:   | CKLOOP                                                         | ;LOOP ON ERROR, IF FLAG SET         |
|      | 044216 | 104406 |        |        |                                                                | TRAP C1CLP1                         |
| 2986 |        |        |        |        |                                                                |                                     |
| 2987 |        |        |        | 1      | Do a Write Subsystem WRITE FIFO 64, bytes incrementing pattern |                                     |
| 2988 | 044220 | 012737 | 000100 | 002310 | MOV #64,COUNT                                                  | ;WRITE 64 BYTES                     |
| 2989 | 044226 | 012701 | 046524 |        | MOV #T17WFDATA,R1                                              | ;EXPD WRITE FIFO DATA BUFFER        |
| 2990 | 044232 | 012702 | 000100 |        | MOV #64,R2                                                     | ;TEST PATTERN SIZE                  |
| 2991 | 044236 | 005000 |        |        | CLR R0                                                         | ;INCREMENT TEST PATTERN             |
| 2992 | 044240 | 110021 |        | 110:   | MOVB R0,(R1).                                                  | ;STORE INCREMENT TEST BYTE          |
| 2993 | 044242 | 005200 |        |        | INC R0                                                         | ;SET NEXT PATTERN                   |
| 2994 | 044244 | 005302 |        |        | DEC R2                                                         | ;DONE?                              |
| 2995 | 044246 | 003374 |        |        | BGT 110:                                                       | ;BR IF NO                           |
| 2996 | 044250 | 013700 | 002310 |        | MOV COUNT,R0                                                   | ;FIFO BYTE COUNT                    |
| 2997 | 044254 | 012701 | 046524 |        | MOV #T17WFDATA,R1                                              | ;FIFO WRITE DATA ADDRESS            |
| 2998 | 044260 | 004737 | 050072 |        | JSR PC,T17WFI                                                  | ;SETUP T17PK2 FOR WRITE FIFO        |
| 2999 | 044264 | 012704 | 050370 |        | MOV #T17PK2,R4                                                 | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 3000 | 044270 | 010465 | 000000 |        | MOV R4,ISDB(R5)                                                | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 3001 | 044274 | 004737 | 016336 |        | JSR PC,CHKTSSR                                                 | ;WAIT FOR SSR TO SET                |
| 3002 | 044300 |        |        |        | FORCERROR 142:                                                 | ;GOODFORCE ERROR IF FORCER=1        |
| 3003 | 044314 | 103407 |        |        | BCS 150:                                                       | ;BR IF CARRY SET (GOOD RETURN)      |
| 3004 | 044316 | 010001 |        |        | MOV R0,R1                                                      | ;SAVE CONTENTS OF TSSR              |
| 3005 | 044320 |        |        |        | NEXT,ERRNO                                                     |                                     |
| 3006 | 044320 |        |        | 142:   | ERRDF ERRNO,T175SSR,PKTSSR                                     | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 044320 | 104455 |        |        |                                                                | TRAP C1ERDF                         |



| Address | Offset | Hex    | Label  | Operation  | Comment               | Trap | Word    |
|---------|--------|--------|--------|------------|-----------------------|------|---------|
| 0000    | 001170 | 044322 |        |            |                       |      | 632     |
| 0001    | 047056 | 044324 |        |            |                       |      | T175SSR |
| 0002    | 012046 | 044326 |        |            |                       |      | PKTSSR  |
| 3007    | 005237 | 044330 | 002222 | INC        | FATFLG                |      |         |
| 3008    | 104406 | 044334 | 1504:  | CKLOOP     |                       |      |         |
| 3009    |        |        |        |            |                       |      |         |
| 3010    |        |        |        |            |                       |      |         |
| 3011    |        |        |        |            |                       |      |         |
| 3012    |        |        |        |            |                       |      |         |
| 3013    |        |        |        |            |                       |      |         |
| 3014    | 004737 | 044336 | 050004 | JSR        | PC,T17SPD             |      |         |
| 3015    | 012704 | 044342 | 050370 | MOV        | #T17PK2,R4            |      |         |
| 3016    | 010465 | 044346 | 000000 | MOV        | R4,TSD8(R5)           |      |         |
| 3017    | 004737 | 044352 | 016336 | JSR        | PC,CHKTSSR            |      |         |
| 3018    |        | 044356 |        | FORCERROR  | 1574                  |      |         |
| 3019    | 103407 | 044372 |        | BCC        | 1604                  |      |         |
| 3020    | 010001 | 044374 |        | MOV        | R0,R1                 |      |         |
| 3021    |        | 044376 |        | NEXT.ERRNO |                       |      |         |
| 3022    |        | 044376 | 1574:  | ERRDF      | ERRNO,T173SSR,PKTSSR  |      |         |
|         | 104455 | 044376 |        |            |                       |      |         |
|         | 001171 | 044400 |        |            |                       |      |         |
|         | 046746 | 044402 |        |            |                       |      |         |
|         | 012046 | 044404 |        |            |                       |      |         |
| 3023    | 005237 | 044406 | 002222 | INC        | FATFLG                |      |         |
| 3024    | 104406 | 044412 | 1604:  | CKLOOP     |                       |      |         |
| 3025    |        |        |        |            |                       |      |         |
| 3026    | 004737 | 044414 | 050166 | JSR        | PC,T17SEIEXP          |      |         |
| 3027    | 012701 | 044420 | 046422 | MOV        | #T17EXSTA,R1          |      |         |
| 3028    | 012702 | 044424 | 050262 | MOV        | #T17BFSTA,R2          |      |         |
| 3029    | 012221 | 044430 |        | MOV        | (R2), (R1)            |      |         |
| 3030    | 011211 | 044432 |        | MOV        | (R2), (R1)            |      |         |
| 3031    | 042711 | 044434 | 000020 | BIC        | #S2,INRDY,(R1)        |      |         |
| 3032    | 052711 | 044440 | 000040 | BIS        | #S2,OUTRDY,(R1)       |      |         |
| 3033    | 042711 | 044444 | 000200 | BIC        | #S2,DIM,(R1)          |      |         |
| 3034    | 005000 | 044450 |        | CLR        | R0                    |      |         |
| 3035    | 012701 | 044452 | 050242 | MOV        | #T17BFR,R1            |      |         |
| 3036    | 012702 | 044456 | 046402 | MOV        | #T17EXP,R2            |      |         |
| 3037    | 012703 | 044462 | 000024 | MOV        | #20,R3                |      |         |
| 3038    | 004737 | 044466 | 011500 | JSR        | PC,CKMSG2             |      |         |
| 3039    |        | 044472 |        | FORCERROR  | 1624,NOTSSR           |      |         |
| 3040    | 103404 | 044502 |        | BCC        | 1704                  |      |         |
| 3041    |        | 044504 |        | NEXT.ERRNO |                       |      |         |
| 3042    |        | 044504 | 1624:  | ERRHRD     | ERRNO,T173CMP,MSGSTAT |      |         |
|         | 104456 | 044504 |        |            |                       |      |         |
|         | 001172 | 044506 |        |            |                       |      |         |
|         | 047343 | 044510 |        |            |                       |      |         |
|         | 012350 | 044512 |        |            |                       |      |         |
| 3043    | 104406 | 044514 | 1704:  | CKLOOP     |                       |      |         |
| 3044    |        |        |        |            |                       |      |         |
| 3045    |        |        |        |            |                       |      |         |
| 3046    |        |        |        |            |                       |      |         |
| 3047    | 012700 | 044516 | 000001 | MOV        | #1,R0                 |      |         |
| 3048    | 012701 | 044522 | 046524 | MOV        | #T17WFDATA,R1         |      |         |
| 3049    | 004737 | 044526 | 050072 | JSR        | PC,T17WFIF            |      |         |

D13

ISVS - HARDWARE TESTS MACRO M1113 06 FEB-84 17:14  
TEST 6: SUBTEST 5: FIFO VERIFY INPUT READY

SEQ 159

|      |        |        |        |            |                                                                  |                                     |          |
|------|--------|--------|--------|------------|------------------------------------------------------------------|-------------------------------------|----------|
| 3050 | 044532 | 012704 | 050370 | MOV        | #T17PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET |          |
| 3051 | 044536 | 010465 | 000000 | MOV        | R4,TSDH(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE  |          |
| 3052 | 044542 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                |          |
| 3053 | 044546 |        |        | FORCERROR  | 172#                                                             | ;###FORCE ERROR IF FORCER=1         |          |
| 3054 | 044562 | 103407 |        | BCS        | 180#                                                             | ;BR IF CARRY SET (GOOD RETURN)      |          |
| 3055 | 044564 | 010001 |        | MOV        | R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |          |
| 3056 | 044566 |        |        | NEXT,ERRNO |                                                                  |                                     |          |
| 3057 | 044566 |        |        | 172#:      | ERRDF ERRNO,T175SSR,PKTSSR                                       | ;DEVICE FATAL SSR FAILED TO SET     |          |
|      | 044566 | 104455 |        |            |                                                                  | TRAP                                | C#ERDF   |
|      | 044570 | 001173 |        |            |                                                                  | .WORD                               | 635      |
|      | 044572 | 047056 |        |            |                                                                  | .WORD                               | T175SSR  |
|      | 044574 | 012046 |        |            |                                                                  | .WORD                               | PKTSSR   |
| 3058 | 044576 | 005237 | 002222 | INC        | FATFLG                                                           | ;SET FATAL ERROR FLAG               |          |
| 3059 | 044602 |        |        | 180#:      | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         |          |
|      | 044602 | 104406 |        |            |                                                                  | TRAP                                | C#CLP1   |
| 3060 |        |        |        |            |                                                                  |                                     |          |
| 3061 |        |        |        |            | Do a Write Subsystem READ STATUS                                 |                                     |          |
| 3062 |        |        |        |            | If Input Ready NOT=0 Then Print Error                            |                                     |          |
| 3063 |        |        |        |            | If Output Ready NOT=1 Then Print Error                           |                                     |          |
| 3064 |        |        |        |            | If Data In Miss NOT=1 Then Print Error                           |                                     |          |
| 3065 | 044604 | 004737 | 050004 | JSR        | PC,T17SRD                                                        | ;SETUP PACKET FOR READ STATUS       |          |
| 3066 | 044610 | 012704 | 050370 | MOV        | #T17PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET |          |
| 3067 | 044614 | 010465 | 000000 | MOV        | R4,TSDH(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE  |          |
| 3068 | 044620 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                |          |
| 3069 | 044624 |        |        | FORCERROR  | 187#                                                             | ;###FORCE ERROR IF FORCER=1         |          |
| 3070 | 044640 | 103407 |        | BCS        | 190#                                                             | ;BR IF CARRY SET (GOOD RETURN)      |          |
| 3071 | 044642 | 010001 |        | MOV        | R0,R1                                                            | ;SAVE CONTENTS OF TSSR              |          |
| 3072 | 044644 |        |        | NEXT,ERRNO |                                                                  |                                     |          |
| 3073 | 044644 |        |        | 187#:      | ERRDF ERRNO,T173SSR,PKTSSR                                       | ;DEVICE FATAL SSR FAILED TO SET     |          |
|      | 044644 | 104455 |        |            |                                                                  | TRAP                                | C#ERDF   |
|      | 044646 | 001174 |        |            |                                                                  | .WORD                               | 636      |
|      | 044650 | 046746 |        |            |                                                                  | .WORD                               | T173SSR  |
|      | 044652 | 012046 |        |            |                                                                  | .WORD                               | PKTSSR   |
| 3074 | 044654 | 005237 | 002222 | INC        | FATFLG                                                           | ;SET FATAL ERROR FLAG               |          |
| 3075 | 044660 |        |        | 190#:      | CKLOOP                                                           | ;LOOP ON ERROR, IF FLAG SET         |          |
|      | 044660 | 104406 |        |            |                                                                  | TRAP                                | C#CLP1   |
| 3076 |        |        |        |            | Set WORDS 0-7 of expd message buffer = to recv since not testing |                                     |          |
| 3077 | 044662 | 004737 | 050166 | JSR        | PC,T17SETEXP                                                     | ;SET WORDS 0-7 EXPD=RCV             |          |
| 3078 | 044666 | 012701 | 046422 | MOV        | #T17EXSTA,R1                                                     | ;GET EXPECTED READ STATUS           |          |
| 3079 | 044672 | 012702 | 050262 | MOV        | #T17BFSTA,R2                                                     | ;GET RCV READ STATUS                |          |
| 3080 | 044676 | 012221 |        | MOV        | (R2), (R1)                                                       | ;SET EXPD WORD #8 = RCV TEMP        |          |
| 3081 | 044700 | 011211 |        | MOV        | (R2), (R1)                                                       | ;SET EXPD WORD #9 = RCV TEMP        |          |
| 3082 | 044702 | 042711 | 000020 | BIC        | #S2,INRDY,(R1)                                                   | ;SET EXP INPUT READY = 0            |          |
| 3083 | 044706 | 052711 | 000040 | BIS        | #S2,OUTRDY,(R1)                                                  | ;SET EXP OUTPUT READY = 1           |          |
| 3084 | 044712 | 052711 | 000200 | BIS        | #S2,DIM,(R1)                                                     | ;SET EXP DATA IN MISS = 1           |          |
| 3085 | 044716 | 005000 |        | CLR        | R0                                                               | ;HIGH RCV ADDRESS FOR CKMSG2        |          |
| 3086 | 044720 | 012701 | 050242 | MOV        | #T17BFR,R1                                                       | ;LOW RCV ADDRESS FOR CKMSG2         |          |
| 3087 | 044724 | 012702 | 046402 | MOV        | #T17EXP,R2                                                       | ;EXPD ADDRESS                       |          |
| 3088 | 044730 | 012703 | 000024 | MOV        | #20, R3                                                          | ;NUMBER OF BYTES TO COMPARE         |          |
| 3089 | 044734 | 004737 | 011500 | JSR        | PC,CKMSG2                                                        | ;EXPD EQUAL RCV?                    |          |
| 3090 | 044740 |        |        | FORCERROR  | 192# NOTSSR                                                      | ;###                                |          |
| 3091 | 044750 | 103404 |        | BCS        | 200#                                                             | ;BR IF YES                          |          |
| 3092 | 044752 |        |        | NEXT,ERRNO |                                                                  |                                     |          |
| 3093 | 044752 |        |        | 192#:      | ERRHRD ERRNO,T173CMP,MSGSTAT                                     | ;REPORT ERROR                       |          |
|      | 044752 | 104456 |        |            |                                                                  | TRAP                                | C#ERRHRD |
|      | 044754 | 001175 |        |            |                                                                  | .WORD                               | 637      |
|      | 044756 | 047343 |        |            |                                                                  | .WORD                               | T173CMP  |

E13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 5: FIFO VERIFY INPUT READY

SEQ 160

|      |        |        |        |        |                                                           |                                     |          |         |
|------|--------|--------|--------|--------|-----------------------------------------------------------|-------------------------------------|----------|---------|
| 3094 | 044760 | 012350 |        | 200\$: | CKLOOP                                                    | ;LOOP ON ERROR, IF FLAG SET         | .WORD    | MSGSTAT |
|      | 044762 |        |        |        |                                                           | TRAP                                | C\$CLP1  |         |
| 3095 | 044762 | 104406 |        |        | Do Write Subsystem READ FIFO                              |                                     |          |         |
| 3096 | 044764 | 013700 | 002310 |        | MOV COUNT,R0                                              | ;SET READ BYTE COUNT                |          |         |
| 3097 | 044770 | 004737 | 050126 |        | JSR PC,T17RFIF                                            | ;SETUP T17PK2 FOR READ FIFO         |          |         |
| 3098 | 044774 | 012704 | 050370 |        | MOV #T17PK2,R4                                            | ;GET WRITE SUBSYSTEM COMMAND PACKET |          |         |
| 3099 | 045000 | 010465 | 000000 |        | MOV R4,TSD8(R5)                                           | ;SET THE PACKET ADDRESS TO EXECUTE  |          |         |
| 3100 | 045004 | 004737 | 016336 |        | JSR PC,CHKTSSR                                            | ;WAIT FOR SSR TO SET                |          |         |
| 3101 | 045010 |        |        |        | FORCERROR 212\$                                           | ;BDOFORCE ERROR IF FORCER=1         |          |         |
| 3102 | 045024 | 103407 |        |        | BCS 220\$                                                 | ;BR IF CARRY SET (GOOD RETURN)      |          |         |
| 3103 | 045026 | 010001 |        |        | MOV R0,R1                                                 | ;SAVE CONTENTS OF TSSR              |          |         |
| 3104 | 045030 |        |        |        | NEXT,ERRNO                                                |                                     |          |         |
| 3105 | 045030 |        |        | 212\$: | ERRDF ERRNO,T176SSR,PKTSSR                                | ;DEVICE FATAL SSR FAILED TO SET     |          |         |
|      | 045030 | 104455 |        |        |                                                           | TRAP                                | C\$ERDF  |         |
|      | 045032 | 001176 |        |        |                                                           | .WORD                               | 638      |         |
|      | 045034 | 047122 |        |        |                                                           | .WORD                               | T176SSR  |         |
|      | 045036 | 012046 |        |        |                                                           | .WORD                               | PKTSSR   |         |
| 3106 | 045040 | 005237 | 002222 |        | INC FATFLG                                                | ;SET FATAL ERROR FLAG               |          |         |
| 3107 | 045044 |        |        | 220\$: | CKLOOP                                                    | ;LOOP ON ERROR, IF FLAG SET         |          |         |
|      | 045044 | 104406 |        |        |                                                           | TRAP                                | C\$CLP1  |         |
| 3108 |        |        |        |        | If Data read from FIFO NOT= to Data sent Then Print Error |                                     |          |         |
| 3109 |        |        |        |        | CLR R0                                                    | ;HIGH RECV ADDRESS FOR CKMSG2       |          |         |
| 3110 | 045046 | 005000 |        |        | MOV #T17WFDATA,R2                                         | ;GET EXPECTED ADDRESS FOR CKMSG2    |          |         |
| 3111 | 045050 | 012702 | 046524 |        | MOV #T17BFSTA,R1                                          | ;GET RECEIVED ADDRESS FOR CKMSG2    |          |         |
| 3112 | 045054 | 012701 | 050262 |        | MOV COUNT,R3                                              | ;NUMBER OF BYTES TO COMPARE         |          |         |
| 3113 | 045060 | 013703 | 002310 |        | JSR PC,CKMSG2                                             | ;EXPD EQUAL RECV?                   |          |         |
| 3114 | 045064 | 004737 | 011500 |        | FORCERROR 232\$,NOTSSR                                    | ;BDO                                |          |         |
| 3115 | 045070 |        |        |        | BCS 240\$                                                 | ;BR IF YES                          |          |         |
| 3116 | 045100 | 103406 |        |        | NEXT,ERRNO                                                |                                     |          |         |
| 3117 | 045102 |        |        |        | MOV COUNT,R1                                              | ;GET BYTE COUNT                     |          |         |
| 3118 | 045102 | 013701 | 002310 | 232\$: | ERRHRD ERRNO,T175CMP,FIFEXP                               | ;REPORT ERROR                       |          |         |
| 3119 | 045106 |        |        |        |                                                           | TRAP                                | C\$ERHRD |         |
|      | 045106 | 104456 |        |        |                                                           | .WORD                               | 639      |         |
|      | 045110 | 001177 |        |        |                                                           | .WORD                               | T175CMP  |         |
|      | 045112 | 047512 |        |        |                                                           | .WORD                               | FIFEXP   |         |
|      | 045114 | 012170 |        |        |                                                           |                                     |          |         |
| 3120 | 045116 |        |        | 240\$: | CKLOOP                                                    | ;LOOP ON ERROR, IF FLAG SET         |          |         |
|      | 045116 | 104406 |        |        |                                                           | TRAP                                | C\$CLP1  |         |
| 3121 |        |        |        |        | Do a Write Subsystem READ STATUS                          |                                     |          |         |
| 3122 |        |        |        |        | If Input Ready NOT=1 Then Print Error                     |                                     |          |         |
| 3123 |        |        |        |        | If Output Ready NOT=0 Then Print Error                    |                                     |          |         |
| 3124 |        |        |        |        | If Data In Miss NOT=1 Then Print Error                    |                                     |          |         |
| 3125 |        |        |        |        | JSR PC,T17SRD                                             | ;SETUP PACKET FOR READ STATUS       |          |         |
| 3126 | 045120 | 004737 | 050004 |        | MOV #T17PK2,R4                                            | ;GET WRITE SUBSYSTEM COMMAND PACKET |          |         |
| 3127 | 045124 | 012704 | 050370 |        | MOV R4,TSD8(R5)                                           | ;SET THE PACKET ADDRESS TO EXECUTE  |          |         |
| 3128 | 045130 | 010465 | 000000 |        | JSR PC,CHKTSSR                                            | ;WAIT FOR SSR TO SET                |          |         |
| 3129 | 045134 | 004737 | 016336 |        | FORCERROR 252\$                                           | ;BDOFORCE ERROR IF FORCER=1         |          |         |
| 3130 | 045140 |        |        |        | BCS 260\$                                                 | ;BR IF CARRY SET (GOOD RETURN)      |          |         |
| 3131 | 045154 | 103407 |        |        | MOV R0,R1                                                 | ;SAVE CONTENTS OF TSSR              |          |         |
| 3132 | 045156 | 010001 |        |        | NEXT,ERRNO                                                |                                     |          |         |
| 3133 | 045160 |        |        |        | ERRDF ERRNO,T173SSR,PKTSSR                                | ;DEVICE FATAL SSR FAILED TO SET     |          |         |
| 3134 | 045160 |        |        | 252\$: |                                                           | TRAP                                | C\$ERDF  |         |
|      | 045160 | 104455 |        |        |                                                           | .WORD                               | 640      |         |
|      | 045162 | 001200 |        |        |                                                           | .WORD                               | T173SSR  |         |
|      | 045164 | 046746 |        |        |                                                           | .WORD                               | PKTSSR   |         |
|      | 045166 | 012046 |        |        |                                                           |                                     |          |         |

F13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 5: FIFO VERIFY INPUT READY

SEQ 161

```

3135 045170 005237 002222          INC      FATFLG          ;SET FATAL ERROR FLAG
3136 045174          260$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
          045174 104406          TRAP      C$CLP1
3137          ;      Set WORDS 0-7 of expd message buffer = to recv since not testing
3138 045176 004737 050166          JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RECV
3139 045202 012701 046422          MOV      @T17EXSTA,R1      ;GET EXPECTED READ STATUS
3140 045206 012702 050262          MOV      @T17BFSTA,R2      ;GET RECV READ STATUS
3141 045212 012221          MOV      (R2)+,(R1)+      ;SET EXPD WORD #8 = RECV TEMP
3142 045214 011211          MOV      (R2),(R1)      ;SET EXPD WORD #9 = RECV TEMP
3143 045216 052711 000020          BIS      @S2.INRDY,(R1)      ;SET EXP INPUT READY= 1
3144 045222 042711 000040          BIC      @S2.OUTRDY,(R1)      ;SET EXP OUTPUT READY= 0
3145 045226 052711 000200          BIS      @S2.DIM,(R1)      ;SET EXP DATA IN MISS = 1
3146 045232 005000          CLR      R0      ;HIGH RECV ADDRESS FOR CKMSG2
3147 045234 012701 050242          MOV      @T17BFR,R1      ;LOW RECV ADDRESS FOR CKMSG2
3148 045240 012702 046402          MOV      @T17EXP,R2      ;EXPD ADDRESS
3149 045244 012703 000024          MOV      @20.,R3      ;NUMBER OF BYTES TO COMPARE
3150 045250 004737 011500          JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
3151 045254          FORCERROR 272$,NOTSSR      ;ERR
3152 045264 103404          BCS      280$      ;BR IF YES
3153 045266          NEXT.ERRNO
3154 045266          272$: ERRHRD ERRNO,T174CMP,MSGSTAT ;REPORT ERROR
          045266 104456          TRAP      C$ERRHRD
          045270 001201          .WORD      641
          045272 047427          .WORD      T174CMP
          045274 012350          .WORD      MSGSTAT
3155 045276          280$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
          045276 104406          TRAP      C$CLP1
3156          ;
3157 045300          ENDSUB          ;////////// END SUBTEST //////////
          045300          L10063:
          045300 104403          TRAP      C$ESUB
3158          ;
3159 045302 005737 002222          TST      FATFLG          ;ANY FATAL ERRORS ?
3160 045306 001402          BEQ      300$      ;BRANCH IF NOT
3161 045310 004737 017202          JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
3162 045314          300$:
3163          ;
3164          ;
3165          ;
3166          .SBTTL TEST 6: SUBTEST 6: FIFO Verify Reset FIFO Test
3167          ;
3168          ;**
3169          ; TEST 6: SUBTEST 6:
3170          ;
3171          ; SUBTEST DESCRIPTION:
3172          ;
3173          ; This subtest verifies that the Reset FIFO function within
3174          ; the Write Miscellaneous Control 1 function initializes
3175          ; the FIFO to correct initial status. The following steps
3176          ; are performed:
3177          ; 1. Reset an already initialized FIFO and check for
3178          ; proper status.
3179          ; 2. Write a varying number of bytes (1-65,) into the
3180          ; FIFO and verify that after each block of bytes is
3181          ; written the FIFO can be reset to it's initial
3182          ; state.
3183          ;

```

```

3184      ; TEST STEPS:
3185      ;
3186      ; BEGIN
3187      ;       Write to TSSR to soft initialize
3188      ;       Do a WRITE CHARACTERISTICS to setup a message buffer
3189      ;       Do a Write Subsystem Write Misc to Reset FIFO
3190      ;       Do a Write Subsystem READ STATUS
3191      ;       If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
3192      ;       signals NOT=0 Then Print Error
3193      ;       Do a Write Subsystem WRITE NPR to set tape direction out
3194      ;
3195      ; REPEAT FOR BYTE COUNT 1 TO 65.
3196      ; BEGIN
3197      ;       Do a Write Subsystem WRITE FIFO with the current byte count
3198      ;       Do a Write Subsystem Write Misc to Reset FIFO
3199      ;       Do a Write Subsystem READ STATUS
3200      ;       If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
3201      ;       signals NOT=0 Then Print Error
3202      ; END
3203      ; --
3204      045314      BGNSUB                                ;////////// BEGIN SUBTEST ///////////
3205      045314      T6.6:                                TRAP      C$BSUB
3206      045314      104402
3207      ;
3208      5$:      Write to TSSR register to soft initialize the controller
3209      045316      004737      015774      JSR      PC,SOFINIT      ;WRITE TO TSSR TO SOFT INITIALIZE
3210      045322      103405      BCS      10$      ;BR IF SOFT INIT OKAY
3211      045324      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3212      045326      104455      ERRDF      ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL DURING INIT
3213      045330      001201      TRAP      C$ERRDF
3214      045332      003652      .WORD      641
3215      045334      012034      .WORD      SFIERR
3216      ;
3217      10$:      Do a WRITE CHARACTERISTICS to setup a message buffer
3218      045336      005037      002222      CLR      FATFLG      ;CLEAR FATAL ERROR FLAG
3219      045342      012704      050220      MOV      @T17PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
3220      045346      004737      010662      JSR      PC,WRCHR      ;DO WRITE CHARACTERISTICS COMMAND
3221      045352      FORCERROR      42$      ;GOODFORCE ERROR IF FORCER=1
3222      045366      103407      BCS      50$      ;BR IF CARRY SET (GOOD RETURN)
3223      045370      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3224      045372      NEXT,ERRNO
3225      42$:      ERRDF      ERRNO,T17SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
3226      045372      104455      TRAP      C$ERRDF
3227      045374      001202      .WORD      642
3228      045376      046645      .WORD      T17SSR
3229      045400      012046      .WORD      PKTSSR
3230      045402      005237      002222      INC      FATFLG      ;SET FATAL ERROR FLAG
3231      045406      104406      50$:      CKLOOP      ;LOOP ON ERROR, IF FLAG SET
3232      045406      104406      TRAP      C$CLP1
3233      ;
3234      Do a Write Subsystem Write Misc to Reset FIFO
3235      045410      004737      050024      JSR      PC,T17RSFIF      ;SETUP PKT FOR WRITE MISC RESET FIFO
3236      045414      012704      050370      MOV      @T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
3237      045420      010465      000000      MOV      R4,T5DB(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
3238      045424      004737      016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
3239      045430      FORCERROR      62$      ;GOODFORCE ERROR IF FORCER=1
3240      045444      103407      BCS      70$      ;BR IF CARRY SET (GOOD RETURN)

```

H13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 163

```

3230 045446 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3231 045450      NEXT,ERRNO
3232 045450      62$:  ERRDF  ERRNO,T172SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERDF
                                .WORD    643
                                .WORD    T172SSR
                                .WORD    PKTSSR
                                045450 104455
                                045452 001203
                                045454 046702
                                045456 012046
3233 045460 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
3234 045464      70$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                045464 104406
3235
3236      ;      Do a Write Subsystem READ STATUS
3237      ;      If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
3238      ;      signals NOT=0 Then Print Error
3239 045466 004737 050004      JSR      PC,T17SRD      ;SETUP PACKET FOR READ STATUS
3240 045472 012704 050370      MOV      @T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
3241 045476 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
3242 045502 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
3243 045506      FORCERROR 77$      ;000FORCE ERROR IF FORCER=1
3244 045522 103407      BCS      80$      ;BR IF CARRY SET (GOOD RETURN)
3245 045524 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3246 045526      NEXT,ERRNO
3247 045526      77$:  ERPDF  ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERDF
                                .WORD    644
                                .WORD    T173SSR
                                .WORD    PKTSSR
                                045526 104455
                                045530 001204
                                045532 046746
                                045534 012046
3248 045536 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
3249 045542      80$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                045542 104406
3250 045544 004737 050166      JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RECV (NOT TESTING)
3251 045550 012701 046422      MOV      @T17EXSTA,R1      ;GET EXPECTED READ STATUS
3252 045554 012702 050262      MOV      @T17BFSTA,R2      ;GET RECV READ STATUS
3253 045560 011211      MOV      (R2),(R1)      ;SET EXPD WORD #8 = RECV TEMP
3254 045562 042711 002000      BIC      @S1.ICER,(R1)      ;SET EXPD ICER =0
3255 045566 042711 001000      BIC      @S1.IFMK,(R1)      ;SET EXPD IFMK =0
3256 045572 042711 000400      BIC      @S1.IHER,(R1)      ;SET EXPD IHER =0
3257 045576 016261 000002 000002      MOV      2(R2),2(R1)      ;SET EXPD WORD #9 = RECV (NOT TESTING)
3258 045604 005000      CLR      R0      ;HIGH RECV ADDRESS FOR CKMSG2
3259 045606 012701 050242      MOV      @T17BFR,R1      ;LOW RECV ADDRESS FOR CKMSG2
3260 045612 012702 046402      MOV      @T17EXP,R2      ;EXPD ADDRESS
3261 045616 012703 000024      MOV      @20.,R3      ;NUMBER OF BYTES TO COMPARE
3262 045622 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
3263 045626      FORCERROR 92$,NOTSSR      ;000
3264 045636 103404      BCS      100$      ;BR IF YES
3265 045640      NEXT,ERRNO
3266 045640      92$:  ERRHRD ERRNO,T177CMP,MSGSTAT ;REPORT ERROR
                                TRAP      C$ERHRD
                                .WORD    645
                                .WORD    T177CMP
                                .WORD    MSGSTAT
                                045640 104456
                                045642 001205
                                045644 047674
                                045646 012350
3267 045650      100$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                045650 104406
3268
3269      ;      Do a Write Subsystem WRITE NPR to set tape direction out
3270 045652 012700 000100      MOV      @NPR.OUT,R0      ;SET TAPE DIRECTION OUT
3271 045656 004737 050046      JSR      PC,T17SNPR      ;SETUP T17PK2 FOR WRITE NPR

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 164

```

3272 045662 012704 050370      MOV      #T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
3273 045666 010465 000000      MOV      R4,TSDB(R5)    ;SET THE PACKET ADDRESS TO EXECUTE
3274 045672 004737 016336      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
3275 045676      FORCERROR      112$      ;###FORCE ERROR IF FORCER=1
3276 045712 103407      BCS      120$      ;BR IF CARRY SET (GOOD RETURN)
3277 045714 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3278 045716      NEXT,ERRNO
3279 045716 112$: ERRDF      ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERDF
                                .WORD      646
                                .WORD      T174SSR
                                .WORD      PKTSSR
                                045716 104455
                                045720 001206
                                045722 047013
                                045724 012046
3280 045726 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
3281 045732 120$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                045732 104406
3282
3283      ;      Setup incrementing pattern in FIFO data buffer
3284 045734 012701 046422      MOV      #T17EXSTA,R1    ;EXPD WRITE FIFO DATA BUFFER
3285 045740 012702 000100      MOV      #64.,R2      ;TEST PATTERN SIZE
3286 045744 005000      CLR      R0      ;INCREMENT TEST PATTERN
3287 045746 110021 130$: MOVB      R0,(R1)+      ;STORE INCREMENT TEST BYTE
3288 045750 005200      INC      R0      ;SET NEXT PATTERN
3289 045752 005302      DEC      R2      ;DONE?
3290 045754 003374      BGT      130$      ;BR IF NO
3291
3292      ; REPEAT FOR BYTE COUNT 1 TO 65.
3293 045756 012737 000001 002310 MOV      #1,COUNT    ;GET FIRST BYTE COUNT
3294      ; Do a Write Subsystem WRITE FIFO with the current byte count
3295 045764 150$: MOV      COUNT,R0    ;REPEAT LOOP LABEL
3296 045764 013700 002310      MOV      #T17EXSTA,R1    ;FIFO BYTE COUNT
3297 045770 012701 046422      MOV      #T17EXSTA,R1    ;FIFO WRITE DATA ADDRESS
3298 045774 004737 050072      JSR      PC,T17WFIF    ;SETUP T17PK2 FOR WRITE FIFO
3299 046000 012704 050370      MOV      #T17PK2,R4    ;GET WRITE SUBSYSTEM COMMAND PACKET
3300 046004 010465 000000      MOV      R4,TSDB(R5)    ;SET THE PACKET ADDRESS TO EXECUTE
3301 046010 004737 016336      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
3302 046014      FORCERROR      152$      ;###FORCE ERROR IF FORCER=1
3303 046030 103407      BCS      160$      ;BR IF CARRY SET (GOOD RETURN)
3304 046032 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3305 046034      NEXT,ERRNO
3306 046034 152$: ERRDF      ERRNO,T175SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERDF
                                .WORD      647
                                .WORD      T175SSR
                                .WORD      PKTSSR
                                046034 104455
                                046036 001207
                                046040 047056
                                046042 012046
3307 046044 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
3308 046050 160$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                046050 104406
3309
3310      ; Do a Write Subsystem Write Misc to Reset FIFO
3311 046052 004737 050024      JSR      PC,T17RSFIF    ;SETUP PKT FOR WRITE MISC RESET FIFO
3312 046056 012704 050370      MOV      #T17PK2,R4    ;GET WRITE SUBSYSTEM COMMAND PACKET
3313 046062 010465 000000      MOV      R4,TSDB(R5)    ;SET THE PACKET ADDRESS TO EXECUTE
3314 046066 004737 016336      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
3315 046072      FORCERROR      162$      ;###FORCE ERROR IF FORCER=1
3316 046106 103407      BCS      170$      ;BR IF CARRY SET (GOOD RETURN)
3317 046110 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
3318 046112      NEXT,ERRNO

```

J13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 165

```

3319 046112      162$:  ERRDF  ERRNO,T172SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      046112 104455      TRAP  C$ERDF
      046114 001210      .WORD 648
      046116 046702      .WORD T172SSR
      046120 012046      .WORD PKTSSR
3320 046122 005237 002222      INC  FATFLG  ;SET FATAL ERROR FLAG
3321 046126      170$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      046126 104406      TRAP  C$CLP1
3322
3323      ; Do a Write Subsystem READ STATUS
3324      ; If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
3325      ; signals NOT=0 Then Print Error
3326 046130 004737 050004      JSR  PC,T17SRD  ;SETUP PACKET FOR READ STATUS
3327 046134 012704 050370      MOV  #T17PK2,R4  ;GET WRITE SUBSYSTEM COMMAND PACKET
3328 046140 010465 000000      MOV  R4,TSD8(R5)  ;SET THE PACKET ADDRESS TO EXECUTE
3329 046144 004737 016336      JSR  PC,CHKTSSR  ;WAIT FOR SSR TO SET
3330 046150      FORCERROR 177$  ;GOODFORCE ERROR IF FORCER=1
3331 046164 103407      BCS 180$  ;BR IF CARRY SET (GOOD RETURN)
3332 046166 010001      MOV  R0,R1  ;SAVE CONTENTS OF TSSR
3333 046170      NEXT,ERRNO
3334 046170      177$:  ERRDF  ERRNO,T173SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      046170 104455      TRAP  C$ERDF
      046172 001211      .WORD 649
      046174 046746      .WORD T173SSR
      046176 012046      .WORD PKTSSR
3335 046200 005237 002222      INC  FATFLG  ;SET FATAL ERROR FLAG
3336 046204      180$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      046204 104406      TRAP  C$CLP1
3337 046206 004737 050166      JSR  PC,T17SETEXP  ;SET WORDS 0-7 EXPD=RCV (NOT TESTING)
3338 046212 012701 046422      MOV  #T17EXSTA,R1  ;GET EXPECTED READ STATUS
3339 046216 012702 050262      MOV  #T17BFSTA,R2  ;GET RCV READ STATUS
3340 046222 011211      MOV  (R2),(R1)  ;SET EXPD WORD #8 = RCV TEMP
3341 046224 042711 002000      BIC  #S1.ICER,(R1)  ;SET EXPD ICER =0
3342 046230 042711 001000      BIC  #S1.IFMK,(R1)  ;SET EXPD IFMK =0
3343 046234 042711 000400      BIC  #S1.IHER,(R1)  ;SET EXPD IHER =0
3344 046240 016261 000002 000002      MOV  2(R2),2(R1)  ;SET EXPD WORD #9 = RCV (NOT TESTING)
3345 046246 005000      CLR  R0  ;HIGH RCV ADDRESS FOR CKMSG2
3346 046250 012701 050242      MOV  #T17BFR,R1  ;LOW RCV ADDRESS FOR CKMSG2
3347 046254 012702 046402      MOV  #T17EXP,R2  ;EXPD ADDRESS
3348 046260 012703 000024      MOV  #20,R3  ;NUMBER OF BYTES TO COMPARE
3349 046264 004737 011500      JSR  PC,CKMSG2  ;EXPD EQUAL RCV?
3350 046270      FORCERROR 192$,NOTSSR  ;GOOD
3351 046300 103404      BCS 200$  ;BR IF YES
3352 046302      NEXT,ERRNO
3353 046302      192$:  ERRHRD  ERRNO,T177CMP,MSGSTAT  ;REPORT ERROR
      046302 104456      TRAP  C$ERHRD
      046304 001212      .WORD 650
      046306 047674      .WORD T177CMP
      046310 012350      .WORD MSGSTAT
3354 046312      200$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      046312 104406      TRAP  C$CLP1
3355
3356
3357 046314      250$:
3358 046314      FORCEEXIT 260$
3359 046324 005237 002310      INC  COUNT  ;GET NEXT BYTE COUNT
3360 046330 023727 002310 000101      CMP  * COUNT,#65,  ;DONE ALL BYTES?

```

K13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 166

```

3361 046336 101002      BHI 260$
3362 046340 000137 045764 JMP 150$
3363 046344      260$:
3364
3365 046344      ENDSUB
      046344
      046344 104403
3366
3367 046346 005737 002222      TST FATFLG
3368 046352 001402      BEQ 300$
3369 046354 004737 017202      JSR PC,CKDROP
3370 046360 004737 016456      JSR PC,TSTLOOP
3371 046364 103002      BCC 305$
3372 046366 000137 040424      JMP T17LOOP
3373 046372      305$:
3374
3375 046372      EXIT TST
      046372 104432
      046374 002112
3376
3377
3378
3379      ;*
3380      ;LOCAL STORAGE FOR THIS TEST
3381      ;*
3382
3383 046376      T17MSK:
3384
3385 046376      377      .BYTE +C<000>
3386 046377      037      .BYTE +C<340>
3387 046400      360      .BYTE +C<017>
3388 046401      000      .BYTE 0
3389
3390 046402      T17EXP:
3391 046402      000000      .WORD 0
3392 046404      000000      .WORD 0
3393 046406      000000      .WORD 0
3394 046410      000000      .WORD 0
3395 046412      000000      .WORD 0
3396 046414      000000      .WORD 0
3397 046416      000000      .WORD 0
3398 046420      000000      .WORD 0
3399 046422      T17EXSTA: .BLKB 66.
3400 046524      T17EXEND:
3401
3402 046524      T17WFDATA: .BLKB 66.
3403
3404
3405      ;*
3406      ;LOCAL TEXT MESSAGES FOR TEST
3407      ;*
3408 046626      106      111      106 TST17ID:      .ASCIZ 'FIFO Exerciser'
3409 046645      127      122      111 T17SSR:      .ASCIZ 'WRITE CHARACTERISTICS Failed'
3410 046702      127      122      111 T172SSR:      .ASCIZ 'WRITE SUBSYSTEM (Write Misc) Failed'
3411 046746      127      122      111 T173SSR:      .ASCIZ 'WRITE SUBSYSTEM (Read Status) Failed'
3412 047013      127      122      111 T174SSR:      .ASCIZ 'WRITE SUBSYSTEM (Write Npr) Failed'
3413 047056      127      122      111 T175SSR:      .ASCIZ 'WRITE SUBSYSTEM (Write FIFO) Failed'

```

```

;BR IF YES
;DO ANOTHER BYTE COUNT

;//////////////// END SUBTEST //////////////////
L10064:
TRAP C$ESUB

;ANY FATAL ERRORS ?
;BRANCH IF NOT
;TRY TO DROP THE UNIT
;DO ITERATIONS?
;BR IF NO
;LOOP UNTIL ITERATIONS DONE

;//////////////// EXIT TEST //////////////////
TRAP C$EXIT
.WORD L10056-.

;MASK OF UNTESTED BITS IN READ STATUS BYTES
;UNTESTED BITS ARE SET TO 1
;BYTE 0 MASK
;BYTE 1 MASK (PARERR,IRESV2,IRESV1)
;BYTE 2 (TIMER A,TIMER B,UNDEFINED<1:0>)
;MAKE IT EVEN

;BEGIN EXPECTED DATA BUFFER
;MESSAGE TYPE
;DATA FIELD LENGTH
;RBPGR
;XST0
;XST1
;XST2
;XST3
;XST4 (ALWAYS PRESENT FOR WRITE SUB.)
;EXPECTED READ STATUS AND WRITE FIFO DATA
;END EXPECTED DATA BUFFER

;WRITE FIFO EXPECTED DATA BUFFER

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 167

|      |        |        |        |       |                                                   |                                                                         |
|------|--------|--------|--------|-------|---------------------------------------------------|-------------------------------------------------------------------------|
| 3414 | 047122 | 127    | 122    | 111   | T176SSR: .ASCIIZ                                  | 'WRITE SUBSYSTEM (Read FIFO) Failed'                                    |
| 3415 | 047165 | 106    | 111    | 106   | T171CMP: .ASCIIZ                                  | 'FIFO Status in WORD #9 Incorrect after Initialize'                     |
| 3416 | 047247 | 122    | 145    | 141   | T172CMP: .ASCIIZ                                  | 'Read FIFO Data not equal to Write FIFO , Data is in WORD #8'           |
| 3417 | 047343 | 106    | 111    | 106   | T173CMP: .ASCIIZ                                  | 'FIFO Status (In WORD #9) Incorrect after WRITE FIFO'                   |
| 3418 | 047427 | 106    | 111    | 106   | T174CMP: .ASCIIZ                                  | 'FIFO Status (In WORD #9) Incorrect after READ FIFO'                    |
| 3419 | 047512 | 122    | 145    | 141   | T175CMP: .ASCIIZ                                  | 'Read FIFO Data not equal to Write FIFO Data'                           |
| 3420 | 047566 | 106    | 111    | 106   | T176CMP: .ASCIIZ                                  | 'FIFO Status (In WORD #9) Incorrect after READ FIFO from an Empty FIFO' |
| 3421 | 047674 | 106    | 111    | 106   | T177CMP: .ASCIIZ                                  | 'FIFO Status (In WORD #9) Incorrect after RESET FIFO'                   |
| 3422 |        |        |        |       | .EVEN                                             |                                                                         |
| 3423 |        |        |        |       |                                                   |                                                                         |
| 3424 |        |        |        |       |                                                   |                                                                         |
| 3425 |        |        |        |       | ;; CLEAR MESSAGE BUFFER                           |                                                                         |
| 3426 |        |        |        |       | ;;                                                |                                                                         |
| 3427 | 047760 |        |        |       | T17CLRBUF:                                        |                                                                         |
| 3428 | 047760 |        |        |       | SAVREG                                            | ; SAVE R1-R5 UNTIL NEXT RETURN                                          |
| 3429 | 047764 | 012701 | 050242 |       | MOV #T17BFR,R1                                    | ; GET MESSAGE BUFFER ADDRESS                                            |
| 3430 | 047770 | 012702 | 000120 |       | MOV #T17BEND-T17BFR,R2                            | ; SIZE OF MESSAGE BUFFER IN BYTES                                       |
| 3431 | 047774 | 105021 |        | 10\$: | CLRB (R1)+                                        | ; CLEAR A BYTE                                                          |
| 3432 | 047776 | 005302 |        |       | DEC R2                                            | ; DONE?                                                                 |
| 3433 | 050000 | 003375 |        |       | BGT 10\$                                          | ; BR IF NO                                                              |
| 3434 | 050202 | 000207 |        |       | RTS PC                                            | ; RETURN                                                                |
| 3435 |        |        |        |       |                                                   |                                                                         |
| 3436 |        |        |        |       |                                                   |                                                                         |
| 3437 |        |        |        |       | ;; SETUP T17PK2 PACKET FOR READ STATUS            |                                                                         |
| 3438 |        |        |        |       | ;;                                                |                                                                         |
| 3439 | 050004 |        |        |       | T17SRD:                                           |                                                                         |
| 3440 | 050004 | 004737 | 047760 |       | JSR PC,T17CLRBUF                                  | ; CLEAR MESSAGE BUFFER                                                  |
| 3441 | 050010 | 012700 | 050400 |       | MOV #T17DT2,R0                                    | ; WRITE SUBSYSTEM DATA BUFFER                                           |
| 3442 | 050014 | 112720 | 000005 |       | MOVB #PW,RDSTATUS,(R0)+                           | ; STORE READ STATUS COMMAND IN BSEL0                                    |
| 3443 | 050020 | 105010 |        |       | CLRB (R0)                                         | ; CLEAR BSEL1                                                           |
| 3444 | 050022 | 000207 |        |       | RTS PC                                            | ; RETURN                                                                |
| 3445 |        |        |        |       |                                                   |                                                                         |
| 3446 |        |        |        |       |                                                   |                                                                         |
| 3447 |        |        |        |       | ;; SETUP T17PK2 PACKET FOR WRITE MISC RESET FIFO  |                                                                         |
| 3448 |        |        |        |       | ;;                                                |                                                                         |
| 3449 | 050024 |        |        |       | T17RSFIF:                                         |                                                                         |
| 3450 | 050024 | 004737 | 047760 |       | JSR PC,T17CLRBUF                                  | ; CLEAR MESSAGE BUFFER                                                  |
| 3451 | 050030 | 012700 | 050400 |       | MOV #T17DT2,R0                                    | ; WRITE SUBSYSTEM DATA BUFFER                                           |
| 3452 | 050034 | 112720 | 000010 |       | MOVB #PW,WMISC,(R0)+                              | ; STORE WRITE MISCELLANEOUS IN BSEL0                                    |
| 3453 | 050040 | 112710 | 000030 |       | MOVB #MS,RSFIF!MS,RSTAP,(R0)                      | ; STORE BSEL1 CLEAR FIFO CODES                                          |
| 3454 | 050044 | 000207 |        |       | RTS PC                                            | ; RETURN                                                                |
| 3455 |        |        |        |       |                                                   |                                                                         |
| 3456 |        |        |        |       |                                                   |                                                                         |
| 3457 |        |        |        |       | ;; SETUP T17PK2 PACKET FOR WRITE NPR              |                                                                         |
| 3458 |        |        |        |       | ;;                                                |                                                                         |
| 3459 |        |        |        |       | ;; INPUT:                                         |                                                                         |
| 3460 |        |        |        |       | ;; R0 CONTAINS BSEL1 NPR DATA                     |                                                                         |
| 3461 |        |        |        |       | ;;                                                |                                                                         |
| 3462 |        |        |        |       | ;; SETS NP,WRP SINCE IF 0 IT WRITES WRONG PARITY. |                                                                         |
| 3463 |        |        |        |       | ;;                                                |                                                                         |
| 3464 | 050046 |        |        |       | T17SNPR:                                          |                                                                         |
| 3465 | 050046 | 004737 | 047760 |       | JSR PC,T17CLRBUF                                  | ; CLEAR MESSAGE BUFFER                                                  |
| 3466 | 050052 | 012701 | 050400 |       | MOV #T17DT2,R1                                    | ; WRITE SUBSYSTEM DATA BUFFER                                           |
| 3467 | 050056 | 112721 | 000011 |       | MOVB #PW,WNPR,(R1)+                               | ; STORE WRITE NPR IN BSEL0                                              |
| 3468 | 050062 | 052700 | 000020 |       | BIS #NP,WRP,R0                                    | ; DON'T WRITE WRONG PARITY                                              |
| 3469 | 050066 | 110011 |        |       | MOVB R0,(R1)                                      | ; STORE NPR DATA IN BSEL1                                               |
| 3470 | 050070 | 000207 |        |       | RTS PC                                            | ; RETURN                                                                |

M13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 168

```

3471
3472
3473      ;+
3474      ; SETUP T17PK2 PACKET FOR WRITE FIFO
3475      ;
3476      ; INPUT:
3477      ;      R0 CONTAINS BYTE COUNT
3478      ;      R1 CONTAINS DATA PATTERN BLOCK ADDRESS
3479      ;-
3479 050072      T17WFIF:
3480 050072      SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
3481 050076      JSR      PC,T17CLRBUF                    ;CLEAR MESSAGE BUFFER
3482 050102      MOV      @T17DT2,R2                      ;WRITE SUBSYSTEM DATA BUFFER
3483 050106      MOV      @PW,WFIFO,(R2)+                 ;STORE WRITE FIFO IN BSEL0
3484 050112      MOV      R0,(R2)+                         ;STORE BYTE COUNT IN BSEL1
3485 050114      CLR      (R2)+                            ;CLEAR SEL2 (UNUSED)
3486 050116      10$:    MOV      (R1)+,(R2)+             ;STORE DATA PATTERN BYTE
3487 050120      DEC      R0                                ;DONE ALL BYTES?
3488 050122      BGT      10$                              ;BR IF NO
3489 050124      RTS      PC                              ;RETURN
3490
3491
3492      ;+
3493      ; SETUP T17PK2 PACKET FOR READ FIFO
3494      ;
3495      ; INPUT:
3496      ;      R0 CONTAINS SEL2 BYTE COUNT
3497      ;-
3497 050126      T17RFIF:
3498 050126      JSR      PC,T17CLRBUF                    ;CLEAR MESSAGE BUFFER
3499 050132      MOV      @T17DT2,R1                      ;WRITE SUBSYSTEM DATA BUFFER
3500 050136      MOV      @PW,RFIFO,(R1)+                 ;STORE READ FIFO IN BSEL0
3501 050142      MOV      R0,(R1)+                         ;STORE BYTE COUNT IN BSEL1
3502 050144      RTS      PC                              ;RETURN
3503
3504      ;+
3505      ; CLEAR EXPECTED DATA MESSAGE BUFFER
3506      ;-
3506 050146      T17CLEXP:
3507 050146      MOV      @T17EXP,R1                      ;GET EXPD ADDRESS
3508 050152      MOV      @T17EXEND-T17EXP,R0             ;GET EXPD SIZE
3509 050156      10$:    CLRB      (R1)+                   ;CLEAR A BYTE
3510 050160      DEC      R0                                ;DONE?
3511 050162      BGT      10$                              ;BR IF NO
3512 050164      RTS      PC                              ;RETURN
3513
3514
3515      ;+
3516      ; Set WORDS 0-7 of expd message buffer = to recv since not testing
3517      ;-
3517 050166      T17SETEXP:
3518 050166      MOV      @T17EXP,R2                      ;GET EXPD
3519 050172      MOV      @T17BFR,R3                      ;GET READ STATUS RECV BUFFER
3520 050176      MOV      @R0,R0                          ;SET WORDS 0-7 EXP=RECV
3521 050202      5$:     MOV      (R3)+,(R2)+             ;SET EXPD=RECV
3522 050204      DEC      R0                                ;DONE WORDS 0-7 WORDS?
3523 050206      BGT      5$                              ;BR IF NO
3524 050210      RTS      PC                              ;RETURN
3525
3527      .<<.10>6177770
3529

```

N13

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 6: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

SEQ 169

```

3530      ;WRITE CHARACTERISTICS COMMAND PACKET
3531      ;
3532 050220      T17PACKET:      ;COMMAND PACKET FOR TEST
3533 050220      .WORD 100004      ;WRITE CHARACTERISTICS COMMAND, WITH ACK
3534 050222      .WORD T17DATA      ;ADDRESS OF CHARACTERISTICS BLOCK
3535 050224      .WORD 0
3536 050226      .WORD 10.      ;MINIMUM MESSAGE PACKET SIZE
3537
3538 050230      T17DATA:      ;CHARACTERISTICS DATA BLOCK
3539 050230      .WORD T17BFR      ;ADDRESS OF MESSAGE BUFFER
3540 050232      .WORD 0
3541 050234      .WORD 20.      ;LENGTH OF MESSAGE BUFFER
3542 050236      .WORD 0      ;ESS,ENB,EAI,ERI
3543 050240      .WORD 0      ;EXTENDED FEATURES UNIT NO, ETC.
3544
3545
3546      ;MESSAGE BUFFER FOR ALL TEST 6 COMMANDS
3547
3548 050242      T17BFR:      ;BEGIN MESSAGE BUFFER
3549 050242      .WORD 0      ;MESSAGE TYPE
3550 050244      .WORD 0      ;DATA FIELD LENGTH
3551 050246      .WORD 0      ;RBPOR
3552 050250      .WORD 0      ;XST0
3553 050252      .WORD 0      ;XST1
3554 050254      .WORD 0      ;XST2
3555 050256      .WORD 0      ;XST3
3556 050260      .WORD 0      ;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM
3557 050262      T17BFSTA: .BLKB 64.      ;READ STATUS AND WRITE FIFO BUFFER
3558 050362      T17BEND:      ;END OF MESSAGE BUFFER
3559
3560      ;WRITE SUBSYSTEM READ STATUS COMMAND PACKET
3561      ;
3562      .=<..10>E177770
3563 050370      T17PK2:      ;WRITE SUBSYSTEM WITH ACK
3564 050370      .WORD P.WRISUB:P.ACK      ;LOW ADDRESS OF DATA BLOCK
3565 050372      .WORD T17DT2      ;HIGH ADDRESS OF DATA BLOCK
3566 050374      .WORD 0
3567 050376      .WORD 10.      ;MINIMUM MESSAGE PACKET SIZE
3568
3569
3570      T17DT2:      ;DATA BLOCK
3571 050400      .BYTE 0      ;BSELO
3572 050400      .BYTE 0      ;BSEL1
3573 050401      .WORD 0      ;SEL2
3574 050402      .BLKB 66.      ;WRITE FIFO DATA OUTPUT BUFFER
3575
3576
3577 050506      ENDTST
3578 050506      L10056:      TRAP      CSETST
3579 050506      104401
3580
3581      ;SBTTL TEST 7: STATIC TRANSPORT BUS INTERFACE TEST
3582
3583      ; TEST DESCRIPTION:
3584
3585      ; TEST STEPS:
3586
3587      ; REPEAT FOR LOOPCNT

```

TSV5 HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
 TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

SEQ 170

```

3587      BEGIN
3588      Write to TSSR register to soft initialize the controller
3589      Do WRITE CHARACTERISTICS to check for Extended Features Switch
3590      IF Extended Features Hardware Switch Clear then:
3591          Do Write Subsystem Write Miscellaneous to Set Extended Features.
3592      Do WRITE CHARACTERISTICS to select reserved unit 7
3593      Do Write Subsystem READ STATUS
3594      If any transport interface signals are asserted then Print Error
3595      END
3596
3597
3598
3599      BGNTEST
3600      050510
3601      050510
3602      050510 012700 051216
3603      050514 004737 016510
3604      050520 012737 000012 002216
3605      050526
3606      050526
3607      050526
3608      050526
3609      050526 004737 015774
3610      050532 103405
3611      050534 010001
3612      050536 104455
3613      050540 001274
3614      050542 003652
3615      050544 012034
3616      050546 005037 002222
3617      050552 012704 051700
3618      050556 004737 010662
3619      050562 103407
3620      050576 010001
3621      050600
3622      050602
3623      050602 104455
3624      050604 001275
3625      050606 051255
3626      050610 012046
3627      050612 005237 002222
3628      050616 104406
3629      050620 012701 051722
3630      050624 032761 000200 000012
3631      050632 001026
3632      050634 004737 051546
3633      050640 012704 051750
3634      050644 010465 000000
3635      050650 004737 016336
3636      050654 103407
3637      050670
  
```

```

      MOV     #TST18ID,R0      ;ASCII MESSAGE TO IDENTIFY TEST
      JSR     PC,TSTSETUP      ;DO INITIAL TEST SETUP
      MOV     #10,,LOOPCNT     ;PERFORM 10 ITERATIONS
      118LOOP:
      Write to TSSR register to soft initialize the controller
      51:
      JSR     PC,SOFINIT        ;WRITE TO TSSR TO SOFT INITIALIZE
      BCS     101               ;BR IF SOFT INIT OKAY
      MOV     R0,R1             ;SAVE CONTENTS OF TSSR
      ERROF   ERRNO,SF1ERR,SF1MSG ;DEVICE FATAL DURING INIT
      TRAP    C1ERR0F          ;WORD 700
      .WORD   700
      .WORD   SF1ERR
      .WORD   SF1MSG
      101:
      Do WRITE CHARACTERISTICS to check for Extended Features Switch
      CLR     FATEFLG           ;CLEAR FATAL ERROR FLAG
      MOV     #T18PK2,R4        ;GET THE ADDRESS OF COMMAND PACKET
      JSR     PC,WRITCHR        ;DO WRITE CHARACTERISTICS COMMAND
      FORCERROR 121             ;GOODFORCE ERROR IF FORCER=1
      BCS     151               ;BR IF CARRY SET (GOOD RETURN)
      MOV     R0,R1             ;SAVE CONTENTS OF TSSR
      NEXT,ERRNO
      121:
      ERROF   ERRNO,T18SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      TRAP    C1ERR0F          ;WORD 701
      .WORD   701
      .WORD   T18SSR
      .WORD   PKTSSR
      151:
      INC     FATEFLG           ;SET FATAL ERROR FLAG
      CKLOOP
      LOOP ON ERROR, IF FLAG SET
      TRAP    C1CIP1
      1:
      IF Extended Features Hardware Switch Clear then:
      Do Write Subsystem Write Miscellaneous to Set Extended Features.
      MOV     #T18BFR,R1        ;MESSAGE BUFFER ADDRESS
      BIT     #X2,EXTF,XST2(R1) ;EXTENDED FEATURES SWITCH SET?
      BNE     301               ;BR IF YES
      JSR     PC,T18SMISC        ;SETUP PACKET FOR WRITE MISCELLANEOUS
      MOV     #T18PK2,R4        ;GET WRITE SUBSYSTEM COMMAND PACKET
      MOV     R4,TSD8(R5)        ;SET THE PACKET ADDRESS TO EXECUTE
      JSR     PC,CHKTSSR        ;WAIT FOR SSR TO SET
      FORCERROR 221             ;GOODFORCE ERROR IF FORCER=1
      BCS     301               ;BR IF CARRY SET (GOOD RETURN)
  
```

C14

ISV5 - HARDWARE TESTS MACRO MILLIS 06-FEB-84 17:14  
TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

SEQ 171

|      |        |        |        |                                                                    |                                    |
|------|--------|--------|--------|--------------------------------------------------------------------|------------------------------------|
| 3638 | 050672 | 010001 |        | MOV R0,R1                                                          | SAVE CONTENTS OF TSSR              |
| 3639 | 050674 |        |        | NEXT,ERRNO                                                         |                                    |
| 3640 | 050674 |        | 22:    | ERRDF ERRNO,T182SSR,PKTSSR                                         | DEVICE FATAL SSR FAILED TO SET     |
|      | 050674 | 104455 |        |                                                                    | TRAP C1ERDF                        |
|      | 050676 | 001276 |        |                                                                    | .WORD 702                          |
|      | 050700 | 051312 |        |                                                                    | .WORD T182SSR                      |
|      | 050702 | 012046 |        |                                                                    | .WORD PKTSSR                       |
| 3641 | 050704 | 005237 | 002222 | INC FATFLG                                                         | SET FATAL ERROR FLAG               |
| 3642 | 050710 |        | 30:    | CKLOOP                                                             | LOOP ON ERROR, IF FLAG SET         |
|      | 050710 | 104406 |        |                                                                    | TRAP C1CLP1                        |
| 3643 |        |        |        |                                                                    |                                    |
| 3644 |        |        |        |                                                                    |                                    |
| 3645 |        |        |        |                                                                    |                                    |
| 3646 | 050712 | 005037 | 002222 | Do WRITE CHARACTERISTICS to select reserved unit 7                 |                                    |
| 3647 | 050716 | 012704 | 051700 | CLR FATFLG                                                         | CLEAR FATAL ERROR FLAG             |
| 3648 | 050722 | 004737 | 010662 | MOV @T18PACKET,R4                                                  | GET THE ADDRESS OF COMMAND PACKET  |
| 3649 | 050726 |        |        | JSR PC,WRCHR                                                       | DO WRITE CHARACTERISTICS COMMAND   |
| 3650 | 050742 | 103407 |        | FORCERROR 42:                                                      | DO FORCE ERROR IF FORCER=1         |
| 3651 | 050744 | 010001 |        | BCS 50:                                                            | BR IF CARRY SET (GOOD RETURN)      |
| 3652 | 050746 |        |        | MOV R0,R1                                                          | SAVE CONTENTS OF TSSR              |
| 3653 | 050746 |        | 42:    | ERRDF ERRNO,T18SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET     |
|      | 050746 | 104455 |        |                                                                    | TRAP C1ERDF                        |
|      | 050750 | 001277 |        |                                                                    | .WORD 703                          |
|      | 050752 | 051255 |        |                                                                    | .WORD T18SSR                       |
|      | 050754 | 012046 |        |                                                                    | .WORD PKTSSR                       |
| 3654 | 050756 | 005237 | 002222 | INC FATFLG                                                         | SET FATAL ERROR FLAG               |
| 3655 | 050762 |        | 50:    | CKLOOP                                                             | LOOP ON ERROR, IF FLAG SET         |
|      | 050762 | 104406 |        |                                                                    | TRAP C1CLP1                        |
| 3656 |        |        |        |                                                                    |                                    |
| 3657 |        |        |        |                                                                    |                                    |
| 3658 | 050764 | 012701 | 051722 | Clear message buffer                                               |                                    |
| 3659 | 050770 | 013700 | 051714 | MOV @T18BFR,R1                                                     | GET MESSAGE BUFFER ADDRESS         |
| 3660 | 050774 | 105021 |        | MOV T18DATA+4,R0                                                   | SIZE OF MESSAGE BUFFER IN BYTES    |
| 3661 | 050776 | 005300 | 60:    | CLRB (R1)+                                                         | CLEAR A BYTE                       |
| 3662 | 051000 | 003375 |        | DEC R0                                                             | DONE?                              |
| 3663 |        |        |        | BGT 60:                                                            | BR IF NO                           |
| 3664 | 051002 | 004737 | 051526 | Do a Write Subsystem READ STATUS                                   |                                    |
| 3665 | 051006 | 012704 | 051750 | JSR PC,T18SRD                                                      | SETUP PACKET FOR READ STATUS       |
| 3666 | 051012 | 010465 | 000000 | MOV @T18PK2,R4                                                     | GET WRITE SUBSYSTEM COMMAND PACKET |
| 3667 | 051016 | 004737 | 016336 | MOV R4,TSD8(R5)                                                    | SET THE PACKET ADDRESS TO EXECUTE  |
| 3668 | 051022 |        |        | JSR PC,CHKTSSR                                                     | WAIT FOR SSR TO SET                |
| 3669 | 051036 | 103407 |        | FORCERROR 62:                                                      | DO FORCE ERROR IF FORCER=1         |
| 3670 | 051040 | 010001 |        | BCS 70:                                                            | BR IF CARRY SET (GOOD RETURN)      |
| 3671 | 051042 |        |        | MOV R0,R1                                                          | SAVE CONTENTS OF TSSR              |
| 3672 | 051042 |        | 62:    | ERRDF ERRNO,T183SSR,PKTSSR                                         | DEVICE FATAL SSR FAILED TO SET     |
|      | 051042 | 104455 |        |                                                                    | TRAP C1ERDF                        |
|      | 051044 | 001300 |        |                                                                    | .WORD 704                          |
|      | 051046 | 051356 |        |                                                                    | .WORD T183SSR                      |
|      | 051050 | 012046 |        |                                                                    | .WORD PKTSSR                       |
| 3673 | 051052 | 005237 | 002222 | INC FATFLG                                                         | SET FATAL ERROR FLAG               |
| 3674 | 051056 |        | 70:    | CKLOOP                                                             | LOOP ON ERROR, IF FLAG SET         |
|      | 051056 | 104406 |        |                                                                    | TRAP C1CLP1                        |
| 3675 |        |        |        |                                                                    |                                    |
| 3676 |        |        |        |                                                                    |                                    |
| 3677 |        |        |        |                                                                    |                                    |
| 3678 |        |        |        |                                                                    |                                    |
| 3679 | 051060 | 004737 | 051570 | Set first 8 words of expd message buffer to recv since not testing |                                    |
|      |        |        |        | Set unused bits in Read Status expd equal recv                     |                                    |
|      |        |        |        | JSR PC,T18SETEXP                                                   | SET SOME EXPD TO RECV              |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB 84 17:14  
TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

SEQ 172

```

3680                                     | If any transport interface signals are asserted then Print Error
3681 051064 005000                       CLR      R0                      ;HIGH RECV ADDRESS FOR CKMSG2
3682 051066 012701 051722                MOV      @T18BFR,R1              ;LOW RECV ADDRESS FOR CKMSG2
3683 051072 012702 051166                MOV      @T18EXP,R2              ;EXPD ADDRESS
3684 051076 012703 000012                MOV      @10,R3                  ;NUMBER OF WORDS TO COMPARE
3685 051102 004737 011500                JSR      PC,CKMSG2                ;EXPD EQUAL RECV?
3686 051106                                FORCERROR 82$,NOTSSR          ;880
3687 051116 103404                        BCS      90$                      ;BR IF YES
3688 051120                                NEXT,ERRNO
3689 051120 82$: ERRNRD ERRNO,T18CMP,MSGSTAT ;REPORT ERROR
                                TRAP      C1ERRNRD
                                .WORD    705
                                .WORD    T18CMP
                                .WORD    MSGSTAT
                                TRAP      C1CLP1
                                .WORD    104456
                                .WORD    001301
                                .WORD    051423
                                .WORD    012350
3690 051130 90$: CKLOOP                  ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C1CLP1
                                .WORD    051130 104406
3691                                     ;ANY FATAL ERRORS ?
3692 051132 005737 002222                TST      FATELG                      ;BRANCH IF NOT
3693 051136 001402                        BEQ      160$                      ;TRY TO DROP THE UNIT
3694 051140 004737 017202                JSR      PC,CKDROP
3695 051144 004737 016456                JSR      PC,TSTLOOP
3696 051150 103002                        BCC      165$                      ;DO ITERATIONS?
3697 051152 000137 050526                JMP      T18LOOP
3698 051156                                ;BR IF NO
3699 051156                                ;LOOP UNTIL ITERATIONS DONE
                                EXIT      TST
                                TRAP      C1EXIT
                                .WORD    L10065-
                                .WORD    051156 104432
                                .WORD    051160 000606
3700                                     ;
3701                                     ;
3702                                     ;
3703                                     ;
3704                                     ;
3705                                     ;
3706 051162                                T18MSK:                          ;MASK OF UNUSED BITS IN READ STATUS BYTES
3707 051162 377                            .BYTE    @C<000>              ;BYTE 0 MASK
3708 051163 037                            .BYTE    @C<340>              ;BYTE 1
3709 051164 100                            .BYTE    @C<277>              ;BYTE 2
3710 051165 000                            .BYTE    0                      ;MAKE IT EVEN
3711                                     ;
3712 051166                                T18EXP:                          ;EXPECTED DATA BUFFER
3713 051166 000000                        .WORD    0                      ;MESSAGE TYPE
3714 051170 000000                        .WORD    0                      ;DATA FIELD LENGTH
3715 051172 000000                        .WORD    0                      ;RBPOR
3716 051174 000000                        .WORD    0                      ;XST0
3717 051176 000000                        .WORD    0                      ;XST1
3718 051200 000000                        .WORD    0                      ;XST2
3719 051202 000000                        .WORD    0                      ;XST3
3720 051204 000000                        .WORD    0                      ;XST4 (ALWAYS PRESENT FOR WRITE SUB)
3721 051206 000000                        .WORD    0                      ;READ STATUS BYTE 1/0
3722 051210 000000                        .WORD    0                      ;READ STATUS BYTE 2
3723                                     ;
3724 051212 377 020                        T18XS: .BYTE    377,020          ;READ STATUS BYTE 0/1 EXPECTED BASE
3725 051214 000000                        .WORD    0                      ;READ STATUS BYTE 2 EXPECTED BASE
3726                                     ;
3727                                     ;
3728                                     ;
3729                                     ;

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

SEQ 1/3

```

3730
3731 051216      123      164      141  TST18ID:      .ASCIZ  'Static Transport Bus Interface'
3732 051255      127      122      111  T18SSR:  .ASCIZ  'WRITE CHARACTERISTICS Failed'
3733 051312      127      122      111  T182SSR: .ASCIZ  'WRITE SUBSYSTEM (Write Misc) Failed'
3734 051356      127      122      111  T183SSR: .ASCIZ  'WRITE SUBSYSTEM (Read Status) Failed'
3735 051423      124      162      141  T18CMP:  .ASCIZ  'Transport Bus Interface Signals NOT Negated After Unit 7 Selected'
3736
3737
3738
3739
3740
3741 051526
3742 051526      012700    051760
3743 051532      112720    000005
3744 051536      105010
3745 051542      000207
3746 051544
3747
3748
3749
3750
3751 051546
3752 051546      012700    051760
3753 051552      112720    000010
3754 051556      112710    000200
3755 051562
3756 051566
3757
3758
3759
3760
3761
3762 051570
3763 051570      012702    051166
3764 051574      012703    051722
3765 051600      012700    000010
3766 051604      012322
3767 051606      005300
3768 051610      003375
3769 051612      012701    051162
3770 051616      013712    051212
3771 051622      013762    051214    000002
3772 051630      011300
3773 051632      041100
3774 051634      040012
3775 051636      050012
3776 051640      016300    000002
3777 051644      046100    000002
3778 051650      040062    000002
3779 051654      050062    000002
3780 051660      105062    000003
3781 051664      105063    000003
3782 051670      000207
3783
3784
3785      051700
3786
3787
3788

```

```

;
; SETUP T18PK2 PACKET FOR READ STATUS
;
T18SRD:
    SAVREG
    MOV     @T18DT2,R0
    MOV8    @PW,RDSTATUS,(R0)
    CLRB    (R0)
    RTS     PC
;SAVE R1-R5 UNTIL NEXT RETURN
;WRITE SUBSYSTEM DATA BUFFER
;STORE READ STATUS COMMAND IN BSEL0
;CLEAR BSEL1
;RETURN

;
; SETUP T18PK2 PACKET FOR WRITE MISC.
;
T18SMISC:
    SAVREG
    MOV     @T18DT2,R0
    MOV8    @PW,WMISC,(R0)
    MOV8    @MS,EXT,(R0)
    RTS     PC
;SAVE R1-R5 UNTIL NEXT RETURN
;WRITE SUBSYSTEM DATA BUFFER
;STORE WRITE MISCELLANEOUS IN BSEL0
;STORE INVERT EXTENDED FEATURES IN BSEL1
;RETURN

;
; Set first 8 words of expd message buffer - to recv since not testing
; Set unused bits in Read Status expd equal rcvd
;
T18SETEXP:
    MOV     @T18EXP,R2
    MOV     @T18BFR,R3
    MOV     @B,,R0
    5$:    MOV     (R3)+,(R2)
    DEC     R0
    BGT     5$
    MOV     @T18MSK,R1
    MOV     T18XS,(R2)
    MOV     T18XS+2,2(R2)
    MOV     (R3),R0
    BIC     (R1),R0
    BIC     R0,(R2)
    BIS     R0,(R2)
    MOV     2(R3),R0
    BIC     2(R1),R0
    BIC     R0,2(R2)
    BIS     R0,2(R2)
    CLRB    3(R2)
    CLRB    3(R3)
    RTS     PC
;GET EXPD
;GET READ STATUS RECV BUFFER
;SET FIRST 8 WORDS EXP=RCV
;SET EXPD=RCV
;DONE FIRST 8 WORDS?
;BR IF NO
;GET UNUSED BIT MASK
;SETUP BASE EXPECTED BYTE 1/0
;SETUP BASE EXPECTED BYTE 2
;GET RECV BYTE 1 AND BYTE 0
;CLEAR ALL BUT UNUSED
;CLEAR UNUSED IN EXP
;SET UNUSED EXPD=RCV FOR COMPARE
;GET RECV BYTE 2
;CLEAR ALL BUT UNUSED
;CLEAR UNUSED IN EXPD
;SET UNUSED EXPD=RCV FOR COMPARE
;CLEAR EXPD BYTE 3 (UNUSED)
;CLEAR RECV BYTE 3 (UNUSED)
;RETURN

;...+10+0177770
;WRITE CHARACTERISTICS COMMAND PACKET

```

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 7: STATIC TRANSPORT BUS INTERFACE TEST

SEQ 174

```

3789
3790 051700      T18PACKET:      ;COMMAND PACKET FOR TEST
3791 051700 100004      .WORD    100004      ;WRITE CHARACTERISTICS COMMAND, WITH ACK
3792 051702 051710      .WORD    T18DATA     ;ADDRESS OF CHARACTERISTICS BLOCK
3793 051704 000000      .WORD    0           ;
3794 051706 000012      .WORD    10          ;MESSAGE PACKET MINIMUM SIZE
3795
3796 051710      T18DATA:        ;CHARACTERISTICS DATA BLOCK
3797 051710 051722      .WORD    T18BFR      ;ADDRESS OF MESSAGE BUFFER
3798 051712 000000      .WORD    0           ;
3799 051714 000024      .WORD    20          ;LENGTH OF MESSAGE BUFFER
3800 051716 000000      .WORD    0           ;ESS,ENB,EAI,ERI
3801 051720 000007      .WORD    7           ;SELECT RESERVED UNIT 7
3802
3803
3804 051722      T18BFR:         ;MESSAGE BUFFER
3805 051722 000000      .WORD    0           ;MESSAGE TYPE
3806 051724 000000      .WORD    0           ;DATA FIELD LENGTH
3807 051726 000000      .WORD    0           ;RBPGR
3808 051730 000000      .WORD    0           ;XST0
3809 051732 000000      .WORD    0           ;XST1
3810 051734 000000      .WORD    0           ;XST2
3811 051736 000000      .WORD    0           ;XST3
3812 051740 000000      .WORD    0           ;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM
3813 051742 000000      .WORD    0           ;READ STATUS BYTE 1/0 RETURNED
3814 051744 000000      .WORD    0           ;READ STATUS BYTE 2
3815
3816      ;WRITE SUBSYSTEM READ STATUS COMMAND PACKET
3817
3819      051750      ;
3821 051750      T18PK2:        ;
3822 051750 100006      .WORD    P.WRTSUB!P.ACK ;WRITE SUBSYSTEM WITH ACK
3823 051752 051760      .WORD    T18DT2       ;LOW ADDRESS OF DATA BLOCK
3824 051754 000000      .WORD    0           ;HIGH ADDRESS OF DATA BLOCK
3825 051756 000010      .WORD    8           ;BUFFER EXTENT
3826
3827 051760      T18DT2:        ;DATA BLOCK
3828 051760      .BYTE    0           ;BSEL0
3829 051761      .BYTE    0           ;B EL1
3830 051762 000000      .WORD    0           ;SEL2
3831 051764 000000      .WORD    0           ;DATA
3832
3833
3834 051766      ENDTST
3835      051766      ;
3836      051766 104401      ;
3837
3838      .SBTIL TEST 8; TRANSPORT BUS INTERFACE LOOPBACK TEST
3839
3840      ;**
3841      ; TEST DESCRIPTION:
3842      ;
3843      ; This test verifies the controller's Transport Bus
3844      ; drivers, receivers, and signal loopback logic. Note
3845      ; that the Static Transport Bus test must have run
3846      ; correctly for this test to provide meaningful results.
3847
3848      ; TEST STEPS:
3849      ;

```

(10005: TRAP C3ETST

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 8: TRANSPORT BUS INTERFACE LOOPBACK TEST

SEQ 175

```

3846      ; REPEAT FOR LOOPCNT
3847      ; BEGIN
3848      ;   Do Subtest 1      - Loopback Control signals test
3849      ;   Do Subtest 2      - Loopback Read/Write signals 'st
3850      ;   Do Subtest 3      - Loopback Write Strobe test
3851      ;   Do Subtest 4      - Loopback Read Strobe test
3852      ; END
3853      ; --
3854
3855
3856      051770      BGNTST
3857      051770
3861      051770      012700      060202      MOV      @TST19ID,R0      ;ASCII MESSAGE TO IDENTIFY TEST
3862      051774      004737      016510      JSR      PC,TSTSETUP      ;DO INITIAL TEST SETUP
3863      052000      012737      000012      002216      MOV      @10,,LOOPCNT      ;PERFORM 10 ITERATIONS
3864      052006      T19LOOP;
3865
3866      .SBTTL TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST
3867
3868      ; **
3869      ; TEST 8: SUBTEST 1:
3870      ;
3871      ; SUBTEST DESCRIPTION:
3872      ;
3873      ;   This subtest verifies the Transport Control loopback
3874      ;   path can transmit and receive correctly. The
3875      ;   control signals are all loopback signals other
3876      ;   than the read/write data (IW<7:0> and IR<7:0>).
3877      ;
3878      ; TEST STEPS:
3879      ;
3880      ;   The loopback signals IFAD,ITADO,ITADI are the tape unit select
3881      ;   lines. Since reserved unit 7 must remain selected these signals
3882      ;   are always set low. This further means the signals they drive
3883      ;   (ISPEED,IRDY,IONL) are only tested in the low state.
3884      ;
3885      ; BEGIN
3886      ;   Write to TSSR register to soft initialize the controller
3887      ;   Do WRITE CHARACTERISTICS to check for Extended Features Switch
3888      ;   If Extended Features Hardware Switch Clear then:
3889      ;   Do Write Subsystem Write Miscellaneous to Set Extended Features.
3890      ;   Do WRITE CHARACTERISTICS to select reserved unit 7 and setup BUFFER
3891      ;   Do a Write Subsystem WRITE NPR to set tape direction out and Loopback
3892      ;   Do Write Subsystem Write Control to CLEAR loopback signals group 1.
3893      ;   Do Write Subsystem Write Format to CLEAR loopback signals group 2.
3894      ;   (the loopback signals have to be cleared here due to the flip-flops
3895      ;   that are set on a 1 to 0 transition (IHER,IFMK,ICER))
3896      ;   Do a Write Subsystem Write Misc to Reset Tape Status F-FLOPS
3897      ;   Do a Write Subsystem READ STATUS
3898      ;   If all Tape Status 2 (ICER,IFMK,IHER) Flip-Flop
3899      ;   signals NOT=0 Then Print Error
3900      ;
3901      ; REPEAT FOR ALL TEST PATTERNS IN 1 BLK TABLE
3902      ; BEGIN
3903      ;   Do Write Subsystem Write Control to Drive loopback signals group 1.
3904      ;   Do Write Subsystem Write Format to Drive loopback signals group 2.
3905      ;   Do a Write Subsystem READ STATUS
  
```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 176

```

3906      ; If loopback data NOT= data sent Then Print Error
3907      ; Do a Write Subsystem Write Misc to Reset Tape Status F-FLOPS
3908      ; Do a Write Subsystem READ STATUS
3909      ; If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
3910      ; signals NOT=0 Then Print Error
3911      ; END
3912      ;--
3913 052006      BGNSUB                      ;////////// BEGIN SUBTEST ////////////
      052006                      T8.1:
      052006 104402                      TRAP      C$BSUB
3914
3915      ; Write to TSSR register to soft initialize the controller
3916 052010      5$:
3917 052010 004737 015774      JSR      PC,SOFINIT          ;WRITE TO TSSR TO SOFT INITIALIZE
3918 052014 103405      BCS      10$          ;BR IF SOFT INIT OKAY
3919 052016 010001      MOV      R0,R1          ;SAVE CONTENTS OF TSSR
3920 052020      ERRDF      ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL DURING INIT
      052020 104455                      TRAP      C$ERRDF
      052022 001440                      .WORD      800
      052024 003652                      .WORD      SFIERR
      052026 012034                      .WORD      SFIMSG
3921      ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
3922 052030 005037 002222      10$:      CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
3923 052034 012704 062330      MOV      @T19PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
3924 052040 004737 010662      JSR      PC,WRCHR          ;DO WRITE CHARACTERISTICS COMMAND
3925 052044      FORCERROR      12$          ;DO FORCE ERROR IF FORCER=1
3926 052060 103407      BCS      15$          ;BR IF CARRY SET (GOOD RETURN)
3927 052062 010001      MOV      R0,R1          ;SAVE CONTENTS OF TSSR
3928 052064      NEXT,ERRNO
3929 052064      12$:      ERRDF      ERRNO,T19SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      052064 104455                      TRAP      C$ERRDF
      052066 001441                      .WORD      801
      052070 060243                      .WORD      T19SSR
      052072 012046                      .WORD      PKTSSR
3930 052074 005237 002222      INC      FATFLG          ;SET FATAL ERROR FLAG
3931 052100      15$:      CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      052100 104406                      TRAP      C$CLP1
3932      ; If Extended Features Hardware Switch Clear then:
3933      ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
3934 052102 012701 062352      MOV      @T19BFR,R1          ;MESSAGE BUFFER ADDRESS
3935 052106 032761 000200 000012      BIT      @X2.EXTF,XST2(R1)      ;EXTENDED FEATURES SWITCH SET?
3936 052114 001026      BNE      30$          ;BR IF YES
3937 052116 004737 062202      JSR      PC,T19SEXT          ;SETUP PACKET FOR WRITE MISC INVERT
3938 052122 012704 062500      MOV      @T19PK2,R4          ;GET WRITE SUBSYSTEM COMMAND PACKET
3939 052126 010465 000000      MOV      R4,TSD8(R5)          ;SET THE PACKET ADDRESS TO EXECUTE
3940 052132 004737 016336      JSR      PC,CHKTSSR          ;WAIT FOR SSR TO SET
3941 052136      FORCERROR      22$          ;DO FORCE ERROR IF FORCER=1
3942 052152 103407      BCS      30$          ;BR IF CARRY SET (GOOD RETURN)
3943 052154 010001      MOV      R0,R1          ;SAVE CONTENTS OF TSSR
3944 052156      NEXT,ERRNO
3945 052156      22$:      ERRDF      ERRNO,T192SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      052156 104455                      TRAP      C$ERRDF
      052160 001442                      .WORD      802
      052162 060300                      .WORD      T192SSR
      052164 012046                      .WORD      PKTSSR
3946 052166 005237 002222      INC      FATFLG          ;SET FATAL ERROR FLAG
3947 052172      30$:      CKLOOP          ;LOOP ON ERROR, IF FLAG SET

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 177

```

052172 104406                                TRAP      C$CLP1
3948                                     ; Do WRITE CHARACTERISTICS to select reserved unit 7
3949 052174 005037 002222                   CLR      FATFLG                      ;CLEAR FATAL ERROR FLAG
3950 052200 012704 060330                   MOV      @T19PACKET,R4              ;GET THE ADDRESS OF COMMAND PACKET
3951 052204 004737 010662                   JSR      PC,WRTCHR                  ;DO WRITE CHARACTERISTICS COMMAND
3952 052210                                     FORCERROR      42$                      ;GOODFORCE ERROR IF FORCER=1
3953 052224 103407                           BCS      50$                      ;BR IF CARRY SET (GOOD RETURN)
3954 052226 010001                           MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
3955 052230                                     NEXT,ERRNO
3956 052230 42$: ERRDF  ERRNO,T19SSR,PKTSSR    ;DEVICE FATAL SSR FAILED TO SET
                                           TRAP      C$ERDF
                                           .WORD      803
                                           .WORD      T19SSR
                                           .WORD      PKTSSR
                                           TRAP      C$CLP1
052230 104455
052232 001443
052234 060243
052236 012046
3957 052240 005237 002222                   INC      FATFLG                      ;SET FATAL ERROR FLAG
3958 052244 50$: CKLOOP                      ;LOOP ON ERROR, IF FLAG SET
                                           TRAP      C$CLP1
052244 104406
3959                                     ; Do a Write Subsystem WRITE NPR to set tape direction out and Loopback
3960 052246 012700 000100                   MOV      @NPR.OUT,R0              ;SET TAPE DIRECTION OUT
3961 052252 052700 000040                   BIS      @NPR.LOOP,R0            ;SET LOOPBACK ENABLE
3962 052256 004737 062042                   JSR      PC,T19SNPR              ;SETUP T19PK2 FOR WRITE NPR
3963 052262 012704 062500                   MOV      @T19PK2,R4              ;GET WRITE SUBSYSTEM COMMAND PACKET
3964 052266 010465 000000                   MOV      R4,TSDB(R5)            ;SET THE PACKET ADDRESS TO EXECUTE
3965 052272 004737 016336                   JSR      PC,CHKTSSR              ;WAIT FOR SSR TO SET
3966 052276                                     FORCERROR      62$                      ;GOODFORCE ERROR IF FORCER=1
3967 052312 103407                           BCS      70$                      ;BR IF CARRY SET (GOOD RETURN)
3968 052314 010001                           MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
3969 052316                                     NEXT,ERRNO
3970 052316 62$: ERRDF  ERRNO,T194SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
                                           TRAP      C$ERDF
                                           .WORD      804
                                           .WORD      T194SSR
                                           .WORD      PKTSSR
                                           TRAP      C$CLP1
052316 104455
052320 001444
052322 060411
052324 012046
3971 052326 005237 002222                   INC      FATFLG                      ;SET FATAL ERROR FLAG
3972 052332 70$: CKLOOP                      ;LOOP ON ERROR, IF FLAG SET
                                           TRAP      C$CLP1
052332 104406
3973                                     ; Do Write Subsystem Write Control to CLEAR loopback signals group 1.
3974                                     ; Do Write Subsystem Write Format to CLEAR loopback signals group 2.
3975                                     ; (the loopback signals have to be cleared here due to the flip-flops
3976                                     ; that are set on a 1 to 0 transition (IHER,IFMK,ICER))
3977 052334 005000                   CLR      R0                      ;WRITE 0'S
3978 052336 042700 000200                   BIC      @WC.IFAD,R0              ;IFAD MUST ALWAYS =0
3979 052342 042700 000100                   BIC      @WC.IOTAD,R0            ;ITADO MUST ALWAYS =0
3980 052346 042700 000040                   BIC      @WC.IITAD,R0           ;ITAD1 MUST ALWAYS =0
3981 052352 004737 062142                   JSR      PC,T19WCTL              ;SETUP PACKET FOR WRITE CONTROL
3982 052356 012704 062500                   MOV      @T19PK2,R4              ;GET WRITE SUBSYSTEM COMMAND PACKET
3983 052362 010465 000000                   MOV      R4,TSDB(R5)            ;SET THE PACKET ADDRESS TO EXECUTE
3984 052366 004737 016336                   JSR      PC,CHKTSSR              ;WAIT FOR SSR TO SET
3985 052372                                     FORCERROR      82$                      ;GOODFORCE ERROR IF FORCER=1
3986 052406 103407                           BCS      90$                      ;BR IF CARRY SET (GOOD RETURN)
3987 052410 010001                           MOV      R0,R1                      ;SAVE CONTENTS OF TSSR
3988 052412                                     NEXT,ERRNO
3989 052412 82$: ERRDF  ERRNO,T197SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
                                           TRAP      C$ERDF
                                           .WORD      805
                                           .WORD      T197SSR
                                           .WORD      PKTSSR
052412 104455
052414 001445
052416 060563
052420 012046

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 178

|      |        |        |        |        |                                                 |                              |       |                                                     |
|------|--------|--------|--------|--------|-------------------------------------------------|------------------------------|-------|-----------------------------------------------------|
| 3990 | 052422 | 005237 | 002222 |        | INC                                             | FATFLG                       |       | ;SET FATAL ERROR FLAG                               |
| 3991 | 052426 |        |        | 90\$:  | CKLOOP                                          |                              |       | ;LOOP ON ERROR, IF FLAG SET                         |
|      | 052426 | 104406 |        |        |                                                 |                              |       | TRAP C\$CLP1                                        |
| 3992 | 052430 | 005000 |        |        | CLR                                             | R0                           |       | ;SET FORMAT DRIVE DATA=0                            |
| 3993 | 052432 | 004737 | 062162 |        | JSR                                             | PC,T19WFM                    |       | ;SETUP PACKET FOR WRITE FORMAT                      |
| 3994 | 052436 | 012704 | 062500 |        | MOV                                             | R4,T19PK2,R4                 |       | ;GET WRITE SUBSYSTEM COMMAND PACKET                 |
| 3995 | 052442 | 010465 | 000000 |        | MOV                                             | R4,TSDB(R5)                  |       | ;SET THE PACKET ADDRESS TO EXECUTE                  |
| 3996 | 052446 | 004737 | 016336 |        | JSR                                             | PC,CHKTSSR                   |       | ;WAIT FOR SSR TO SET                                |
| 3997 | 052452 |        |        |        | FORCERROR                                       | 102\$                        |       | ;B00FORCE ERROR IF FORCER=1                         |
| 3998 | 052466 | 103407 |        |        | BCS                                             | 110\$                        |       | ;BR IF CARRY SET (GOOD RETURN)                      |
| 3999 | 052470 | 010001 |        |        | MOV                                             | R0,R1                        |       | ;SAVE CONTENTS OF TSSR                              |
| 4000 | 052472 |        |        |        | NEXT,ERRNO                                      |                              |       |                                                     |
| 4001 | 052472 |        |        | 102\$: | ERRDF                                           | ERRNO,T198SSR,PKTSSR         |       | ;DEVICE FATAL SSR FAILED TO SET                     |
|      | 052472 | 104455 |        |        |                                                 |                              | TRAP  | C\$ERDF                                             |
|      | 052474 | 001446 |        |        |                                                 |                              | .WORD | 806                                                 |
|      | 052476 | 060632 |        |        |                                                 |                              | .WORD | T198SSR                                             |
|      | 052500 | 012046 |        |        |                                                 |                              | .WORD | PKTSSR                                              |
| 4002 | 052502 | 005237 | 002222 |        | INC                                             | FATFLG                       |       | ;SET FATAL ERROR FLAG                               |
| 4003 | 052506 |        |        | 110\$: | CKLOOP                                          |                              |       | ;LOOP ON ERROR, IF FLAG SET                         |
|      | 052506 | 104406 |        |        |                                                 |                              | TRAP  | C\$CLP1                                             |
| 4004 |        |        |        | :      | Do a Write Subsystem Write Misc                 | to Reset Tape Status F-FLOPS |       |                                                     |
| 4005 | 052510 | 004737 | 062020 |        | JSR                                             | PC,T19RSEIF                  |       | ;SETUP PKT FOR WRITE MISC Reset Tape Status F-FLOPS |
| 4006 | 052514 | 012704 | 062500 |        | MOV                                             | R4,T19PK2,R4                 |       | ;GET WRITE SUBSYSTEM COMMAND PACKET                 |
| 4007 | 052520 | 010465 | 000000 |        | MOV                                             | R4,TSDB(R5)                  |       | ;SET THE PACKET ADDRESS TO EXECUTE                  |
| 4008 | 052524 | 004737 | 016336 |        | JSR                                             | PC,CHKTSSR                   |       | ;WAIT FOR SSR TO SET                                |
| 4009 | 052530 |        |        |        | FORCERROR                                       | 122\$                        |       | ;B00FORCE ERROR IF FORCER=1                         |
| 4010 | 052544 | 103407 |        |        | BCS                                             | 130\$                        |       | ;BR IF CARRY SET (GOOD RETURN)                      |
| 4011 | 052546 | 010001 |        |        | MOV                                             | R0,R1                        |       | ;SAVE CONTENTS OF TSSR                              |
| 4012 | 052550 |        |        |        | NEXT,ERRNO                                      |                              |       |                                                     |
| 4013 | 052550 |        |        | 122\$: | ERRDF                                           | ERRNO,T192SSR,PKTSSR         |       | ;DEVICE FATAL SSR FAILED TO SET                     |
|      | 052550 | 104455 |        |        |                                                 |                              | TRAP  | C\$ERDF                                             |
|      | 052552 | 001447 |        |        |                                                 |                              | .WORD | 807                                                 |
|      | 052554 | 060300 |        |        |                                                 |                              | .WORD | T192SSR                                             |
|      | 052556 | 012046 |        |        |                                                 |                              | .WORD | PKTSSR                                              |
| 4014 | 052560 | 005237 | 002222 |        | INC                                             | FATFLG                       |       | ;SET FATAL ERROR FLAG                               |
| 4015 | 052564 |        |        | 130\$: | CKLOOP                                          |                              |       | ;LOOP ON ERROR, IF FLAG SET                         |
|      | 052564 | 104406 |        |        |                                                 |                              | TRAP  | C\$CLP1                                             |
| 4016 |        |        |        | :      | Do a Write Subsystem READ STATUS                |                              |       |                                                     |
| 4017 |        |        |        | :      | If all Tape Status 2 (ICER,IFMK,IHER) flip-flop |                              |       |                                                     |
| 4018 |        |        |        | :      | signals NOT=0 Then Print Error                  |                              |       |                                                     |
| 4019 | 052566 | 004737 | 062000 |        | JSR                                             | PC,T19SRD                    |       | ;SETUP PACKET FOR READ STATUS                       |
| 4020 | 052572 | 012704 | 062500 |        | MOV                                             | R4,T19PK2,R4                 |       | ;GET WRITE SUBSYSTEM COMMAND PACKET                 |
| 4021 | 052576 | 010465 | 000000 |        | MOV                                             | R4,TSDB(R5)                  |       | ;SET THE PACKET ADDRESS TO EXECUTE                  |
| 4022 | 052602 | 004737 | 016336 |        | JSR                                             | PC,CHKTSSR                   |       | ;WAIT FOR SSR TO SET                                |
| 4023 | 052606 |        |        |        | FORCERROR                                       | 132\$                        |       | ;B00FORCE ERROR IF FORCER=1                         |
| 4024 | 052622 | 103407 |        |        | BCS                                             | 140\$                        |       | ;BR IF CARRY SET (GOOD RETURN)                      |
| 4025 | 052624 | 010001 |        |        | MOV                                             | R0,R1                        |       | ;SAVE CONTENTS OF TSSR                              |
| 4026 | 052626 |        |        |        | NEXT,ERRNO                                      |                              |       |                                                     |
| 4027 | 052626 |        |        | 132\$: | ERRDF                                           | ERRNO,T193SSR,PKTSSR         |       | ;DEVICE FATAL SSR FAILED TO SET                     |
|      | 052626 | 104455 |        |        |                                                 |                              | TRAP  | C\$ERDF                                             |
|      | 052630 | 001450 |        |        |                                                 |                              | .WORD | 808                                                 |
|      | 052632 | 060344 |        |        |                                                 |                              | .WORD | T193SSR                                             |
|      | 052634 | 012046 |        |        |                                                 |                              | .WORD | PKTSSR                                              |
| 4028 | 052636 | 005237 | 002222 |        | INC                                             | FATFLG                       |       | ;SET FATAL ERROR FLAG                               |
| 4029 | 052642 |        |        | 140\$: | CKLOOP                                          |                              |       | ;LOOP ON ERROR, IF FLAG SET                         |
|      | 052642 | 104406 |        |        |                                                 |                              | TRAP  | C\$CLP1                                             |
| 4030 | 052644 | 004737 | 062240 |        | JSR                                             | PC,T19SETEXP                 |       | ;SET WORDS 0-7 EXPD=RECV (NOT TESTING)              |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 179

|      |        |        |               |            |                                                                     |                                        |       |           |
|------|--------|--------|---------------|------------|---------------------------------------------------------------------|----------------------------------------|-------|-----------|
| 4031 | 052650 | 012701 | 060102        | MOV        | 0T19EXSTA,R1                                                        | ;GET EXPECTED READ STATUS              |       |           |
| 4032 | 052654 | 012702 | 062372        | MOV        | 0T19BFSTA,R2                                                        | ;GET RECV READ STATUS                  |       |           |
| 4033 | 052660 | 011211 |               | MOV        | (R2),(R1)                                                           | ;SET EXPD WORD 08 = RECV TEMP          |       |           |
| 4034 | 052662 | 042711 | 002000        | BIC        | 0S1.ICER,(R1)                                                       | ;SET EXPD ICER =0                      |       |           |
| 4035 | 052666 | 042711 | 001000        | BIC        | 0S1.IFMK,(R1)                                                       | ;SET EXPD IFMK =0                      |       |           |
| 4036 | 052672 | 042711 | 000400        | BIC        | 0S1.IHER,(R1)                                                       | ;SET EXPD IHER =0                      |       |           |
| 4037 | 052676 | 016261 | 000002 000002 | MOV        | 2(R2),2(R1)                                                         | ;SET EXPD WORD 09 = RECV (NOT TESTING) |       |           |
| 4038 | 052704 | 005000 |               | CLR        | R0                                                                  | ;HIGH RECV ADDRESS FOR CKMSG2          |       |           |
| 4039 | 052706 | 012701 | 062352        | MOV        | 0T19BFR,R1                                                          | ;LOW RECV ADDRESS FOR CKMSG2           |       |           |
| 4040 | 052712 | 012702 | 060062        | MOV        | 0T19EXP,R2                                                          | ;EXPD ADDRESS                          |       |           |
| 4041 | 052716 | 012703 | 000024        | MOV        | 020,R3                                                              | ;NUMBER OF BYTES TO COMPARE            |       |           |
| 4042 | 052722 | 004737 | 011500        | JSR        | PC,CKMSG2                                                           | ;EXPD EQUAL RECV?                      |       |           |
| 4043 | 052726 |        |               | FORCERROR  | 152\$,NOTSSR                                                        | ;000                                   |       |           |
| 4044 | 052736 | 103404 |               | BCS        | 160\$                                                               | ;BR IF YES                             |       |           |
| 4045 | 052740 |        |               | NEXT,ERRNO |                                                                     |                                        |       |           |
| 4046 | 052740 |        |               | 152\$:     | ERRHRD ERRNO,T197CMP,MSGLOOP                                        | ;REPORT ERROR                          |       |           |
|      | 052740 | 104456 |               |            |                                                                     |                                        | TRAP  | C\$ERRHRD |
|      | 052742 | 001451 |               |            |                                                                     |                                        | .WORD | 809       |
|      | 052744 | 061303 |               |            |                                                                     |                                        | .WORD | T197CMP   |
|      | 052746 | 013064 |               |            |                                                                     |                                        | .WORD | MSGLOOP   |
| 4047 | 052750 |        |               | 160\$:     | CKLOOP                                                              | ;LOOP ON ERROR, IF FLAG SET            |       |           |
|      | 052750 | 104406 |               |            |                                                                     |                                        | TRAP  | C\$CLP1   |
| 4048 |        |        |               |            | ; REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE                      |                                        |       |           |
| 4049 | 052752 | 005037 | 060014        | CLR        | T19PREV                                                             | ;INIT 1-0 TRANSITION FLAG              |       |           |
| 4050 | 052756 | 012703 | 002752        | MOV        | 0TSTBLK,R3                                                          | ;GET FIRST PATTERN ADDRESS             |       |           |
| 4051 | 052762 | 012300 |               | 200\$:     | MOV (R3)+,R0                                                        | ;GET A TEST PATTERN                    |       |           |
| 4052 | 052764 | 010337 | 002316        | MOV        | R3,TSTPTR                                                           | ;SAVE POINTER INTO TSTBLK              |       |           |
| 4053 | 052770 | 042700 | 000200        | BIC        | 0WC.IFAD,R0                                                         | ;IFAD MUST ALWAYS =0                   |       |           |
| 4054 | 052774 | 042700 | 000100        | BIC        | 0WC.IOTAD,R0                                                        | ;ITADO MUST ALWAYS =0                  |       |           |
| 4055 | 053000 | 042700 | 000040        | BIC        | 0WC.IITAD,R0                                                        | ;ITAD1 MUST ALWAYS =0                  |       |           |
| 4056 | 053004 | 010037 | 002312        | MOV        | R0,DATA                                                             | ;SET DATA PATTERN                      |       |           |
| 4057 |        |        |               |            | Do Write Subsystem Write Control to Drive loopback signals group 1. |                                        |       |           |
| 4058 |        |        |               | ;000       | CALL T19CNVT TO SETUP WRITE CONTROL PATTERN                         |                                        |       |           |
| 4059 | 053010 | 013700 | 002312        | MOV        | DATA,R0                                                             | ;GET TEST PATTERN                      |       |           |
| 4060 | 053014 | 004737 | 062264        | JSR        | PC,T19CNVT                                                          | ;CONVERT PATTERN TO CONTROL DRIVE MASK |       |           |
| 4061 |        |        |               |            | R0 CONTAINS WRITE CONTROL DATA HERE                                 |                                        |       |           |
| 4062 | 053020 | 004737 | 062142        | JSR        | PC,T19WCTL                                                          | ;SETUP PACKET FOR WRITE CONTROL        |       |           |
| 4063 | 053024 | 012704 | 062500        | MOV        | 0T19PK2,R4                                                          | ;GET WRITE SUBSYSTEM COMMAND PACKET    |       |           |
| 4064 | 053030 | 010465 | 000000        | MOV        | R4,TSD8(R5)                                                         | ;SET THE PACKET ADDRESS TO EXECUTE     |       |           |
| 4065 | 053034 | 004737 | 016336        | JSR        | PC,CHKTSSR                                                          | ;WAIT FOR SSR TO SET                   |       |           |
| 4066 | 053040 |        |               | FORCERROR  | 212\$                                                               | ;000FORCE ERROR IF FORCER=1            |       |           |
| 4067 | 053054 | 104407 |               | BCS        | 220\$                                                               | ;BR IF CARRY SET (GOOD RETURN)         |       |           |
| 4068 | 053056 | 010001 |               | MOV        | R0,R1                                                               | ;SAVE CONTENTS OF TSSR                 |       |           |
| 4069 | 053060 |        |               | NEXT,ERRNO |                                                                     |                                        |       |           |
| 4070 | 053060 |        |               | 212\$:     | ERRDF ERRNO,T197SSR,PKTSSR                                          | ;DEVICE FATAL SSR FAILED TO SET        |       |           |
|      | 053060 | 104455 |               |            |                                                                     |                                        | TRAP  | C\$ERRDF  |
|      | 053062 | 001452 |               |            |                                                                     |                                        | .WORD | 810       |
|      | 053064 | 060563 |               |            |                                                                     |                                        | .WORD | T197SSR   |
|      | 053066 | 012046 |               |            |                                                                     |                                        | .WORD | PKTSSR    |
| 4071 | 053070 | 005237 | 002222        | INC        | FATFLG                                                              | ;SET FATAL ERROR FLAG                  |       |           |
| 4072 | 053074 |        |               | 220\$:     | CKLOOP                                                              | ;LOOP ON ERROR, IF FLAG SET            |       |           |
|      | 053074 | 104406 |               |            |                                                                     |                                        | TRAP  | C\$CLP1   |
| 4073 |        |        |               |            |                                                                     |                                        |       |           |
| 4074 |        |        |               |            | Do Write Subsystem Write Format to Drive loopback signals group 2.  |                                        |       |           |
| 4075 |        |        |               | ;000       | CALL T19CNVT TO SETUP WRITE CONTROL PATTERN                         |                                        |       |           |
| 4076 | 053076 | 013700 | 002312        | MOV        | DATA,R0                                                             | ;GET TEST PATTERN                      |       |           |
| 4077 | 053102 | 004737 | 062264        | JSR        | PC,T19CNVT                                                          | ;CONVERT PATTERN TO FORMAT DRIVE MASK  |       |           |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 180

|      |        |        |        |            |                                                  |                      |                                          |
|------|--------|--------|--------|------------|--------------------------------------------------|----------------------|------------------------------------------|
| 4078 | 053106 | 000300 |        | SWAB       | R0                                               |                      | ;WRITE FORMAT DATA RETURNED IN HIGH BYTE |
| 4079 | 053110 | 004737 | 062162 | JSR        | PC,T19WFMT                                       |                      | ;SETUP PACKET FOR WRITE FORMAT           |
| 4080 | 053114 | 012704 | 062500 | MOV        | #T19PK2,R4                                       |                      | ;GET WRITE SUBSYSTEM COMMAND PACKET      |
| 4081 | 053120 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                      |                      | ;SET THE PACKET ADDRESS TO EXECUTE       |
| 4082 | 053124 | 004737 | 016336 | JSR        | PC,CHKTSSR                                       |                      | ;WAIT FOR SSR TO SET                     |
| 4083 | 053130 |        |        | FORCERROR  | 232\$                                            |                      | ;DO FORCE ERROR IF FORCER=1              |
| 4084 | 053144 | 103407 |        | BCS        | 240\$                                            |                      | ;BR IF CARRY SET (GOOD RETURN)           |
| 4085 | 053146 | 010001 |        | MOV        | R0,R1                                            |                      | ;SAVE CONTENTS OF TSSR                   |
| 4086 | 053150 |        |        | NEXT,ERRNO |                                                  |                      |                                          |
| 4087 | 053150 |        |        | 232\$:     | ERRDF                                            | ERRNO,T198SSR,PKTSSR | ;DEVICE FATAL SSR FAILED TO SET          |
|      | 053150 | 104455 |        |            |                                                  |                      | TRAP C\$ERDF                             |
|      | 053152 | 001453 |        |            |                                                  |                      | .WORD 811                                |
|      | 053154 | 060632 |        |            |                                                  |                      | .WORD T198SSR                            |
|      | 053156 | 012046 |        |            |                                                  |                      | .WORD PKTSSR                             |
| 4088 | 053160 | 005237 | 002222 | INC        | FATFLG                                           |                      | ;SET FATAL ERROR FLAG                    |
| 4089 | 053164 |        |        | 240\$:     | CKLOOP                                           |                      | ;LOOP ON ERROR, IF FLAG SET              |
|      | 053164 | 104406 |        |            |                                                  |                      | TRAP C\$CLP1                             |
| 4090 |        |        |        | :          | Do a Write Subsystem READ STATUS                 |                      |                                          |
| 4091 | 053166 | 004737 | 062000 | JSR        | PC,T19SRD                                        |                      | ;SETUP PACKET FOR READ STATUS            |
| 4092 | 053172 | 012704 | 062500 | MOV        | #T19PK2,R4                                       |                      | ;GET WRITE SUBSYSTEM COMMAND PACKET      |
| 4093 | 053176 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                      |                      | ;SET THE PACKET ADDRESS TO EXECUTE       |
| 4094 | 053202 | 004737 | 016336 | JSR        | PC,CHKTSSR                                       |                      | ;WAIT FOR SSR TO SET                     |
| 4095 | 053206 |        |        | FORCERROR  | 252\$                                            |                      | ;DO FORCE ERROR IF FORCER=1              |
| 4096 | 053222 | 103407 |        | BCS        | 260\$                                            |                      | ;BR IF CARRY SET (GOOD RETURN)           |
| 4097 | 053224 | 010001 |        | MOV        | R0,R1                                            |                      | ;SAVE CONTENTS OF TSSR                   |
| 4098 | 053226 |        |        | NEXT,ERRNO |                                                  |                      |                                          |
| 4099 | 053226 |        |        | 252\$:     | ERRDF                                            | ERRNO,T193SSR,PKTSSR | ;DEVICE FATAL SSR FAILED TO SET          |
|      | 053226 | 104455 |        |            |                                                  |                      | TRAP C\$ERDF                             |
|      | 053230 | 001454 |        |            |                                                  |                      | .WORD 812                                |
|      | 053232 | 060344 |        |            |                                                  |                      | .WORD T193SSR                            |
|      | 053234 | 012046 |        |            |                                                  |                      | .WORD PKTSSR                             |
| 4100 | 053236 | 005237 | 002222 | INC        | FATFLG                                           |                      | ;SET FATAL ERROR FLAG                    |
| 4101 | 053242 |        |        | 260\$:     | CKLOOP                                           |                      | ;LOOP ON ERROR, IF FLAG SET              |
|      | 053242 | 104406 |        |            |                                                  |                      | TRAP C\$CLP1                             |
| 4102 |        |        |        | :          | If loopback data NOT= data sent Then Print Error |                      |                                          |
| 4103 | 053244 | 004737 | 062240 | JSR        | PC,T19SETEXP                                     |                      | ;SET WORDS 0-7 EXPD=RECV (NOT TESTING)   |
| 4104 | 053250 | 012701 | 060102 | MOV        | #T19EXSTA,R1                                     |                      | ;GET EXPECTED READ STATUS                |
| 4105 | 053254 | 012702 | 062372 | MOV        | #T19BFSTA,R2                                     |                      | ;GET RECV READ STATUS                    |
| 4106 | 053260 | 013711 | 002312 | MOV        | DATA,(R1)                                        |                      | ;SET EXPD WORD #8 TO TEST DATA FIRST     |
| 4107 | 053264 | 013700 | 060014 | MOV        | T19PREV,R0                                       |                      | ;GET PREVIOUS DATA PATTERN               |
| 4108 | 053270 | 013703 | 002312 | MOV        | DATA,R3                                          |                      | ;GET CURRENT PATTERN                     |
| 4109 | 053274 | 012704 | 000400 | MOV        | #S1.IHER,R4                                      |                      | ;SETUP IHER EXPECTED                     |
| 4110 | 053300 | 040411 |        | BIC        | R4,(R1)                                          |                      | ;SET EXPD IHER =0                        |
| 4111 | 053302 | 030400 |        | BIT        | R4,R0                                            |                      | ;PREVIOUS =1?                            |
| 4112 | 053304 | 001403 |        | BEQ        | 275\$                                            |                      | ;BR IF NO                                |
| 4113 | 053306 | 030403 |        | BIT        | R4,R5                                            |                      | ;CURRENT =0?                             |
| 4114 | 053310 | 001001 |        | BNE        | 275\$                                            |                      | ;BR IF NO                                |
| 4115 | 053312 | 050411 |        | BIS        | R4,(R1)                                          |                      | ;SET EXPD IHER =1                        |
| 4116 | 053314 | 012704 | 001000 | 275\$:     | MOV                                              | #S1.IFMK,R4          | ;SETUP IFMK EXPECTED                     |
| 4117 | 053320 | 040411 |        | BIC        | R4,(R1)                                          |                      | ;SET EXPD IFMK =0                        |
| 4118 | 053322 | 030400 |        | BIT        | R4,R0                                            |                      | ;PREVIOUS =1?                            |
| 4119 | 053324 | 001403 |        | BEQ        | 280\$                                            |                      | ;BR IF NO                                |
| 4120 | 053326 | 030403 |        | BIT        | R4,R5                                            |                      | ;CURRENT =0?                             |
| 4121 | 053330 | 001001 |        | BNE        | 280\$                                            |                      | ;BR IF NO                                |
| 4122 | 053332 | 050411 |        | BIS        | R4,(R1)                                          |                      | ;SET EXPD IFMK =1                        |
| 4123 | 053334 | 012704 | 002000 | 280\$:     | MOV                                              | #S1.ICER,R4          | ;SETUP ICER EXPECTED                     |
| 4124 | 053340 | 040411 |        | BIC        | R4,(R1)                                          |                      | ;SET EXPD ICER =0                        |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 181

```

4125 053342 030400      BIT      R4,R0      ;PREVIOUS =1?
4126 053344 001403      BEQ      285$      ;BR IF NO
4127 053346 030403      BIT      R4,R3      ;CURRENT =0?
4128 053350 001001      BNE      285$      ;BR IF NO
4129 053352 050411      BIS      R4,(R1)    ;SET EXPD ICER =1
4130 053354 011100      MOV      (R1),R0    ;GET EXPD WORD
4131          ; If previous IIDENT=1 and current is IIDENT=1 then EXPD= 0 else 1
4132 053356 012704 004000      MOV      $S1.IIDENT,R4 ;IIDENT
4133 053362 050400      BIS      R4,R0      ;ASSUME EXPD=1
4134 053364 030437 060014      BIT      R4,T19PREV ;PREVIOUS IIDENT=1?
4135 053370 001403      BEQ      288$      ;BR IF NO
4136 053372 030403      BIT      R4,R3      ;IS CURRENT IIDENT=1?
4137 053374 001401      BEQ      288$      ;BR IF NO
4138 053376 040400      BIC      R4,R0      ;SET EXPD=0
4139 053400 052700 040000      BIS      $S1.I2RES,R0 ;IRESV2 EXPD ALWAYS=1
4140 053404 052700 020000      BIS      $S1.I1RES,R0 ;IRESV1 EXPD ALWAYS=1
4141 053410 042700 100000      BIC      $S1.PARERR,R0 ;IGNORE PARERR
4142 053414 032712 100000      BIT      $S1.PARERR,(R2) ;IS PARERR SET IN RECV?
4143 053420 001402      BEQ      290$      ;BR IF NO
4144 053422 052700 100000      BIS      $S1.PARERR,R0 ;SET IN EXPD
4145 053426 010011      MOV      R0,(R1)    ;SETUP FINAL EXPD IN WORD #8
4146 053430 016261 000002 000002      MOV      2(R2),2(R1) ;SET EXPD WORD #9 = RECV (NOT TESTING)
4147 053436 005000      CLR      R0        ;HIGH RECV ADDRESS FOR CKMSG2
4148 053440 012701 062352      MOV      $T19BFR,R1 ;LOW RECV ADDRESS FOR CKMSG2
4149 053444 012702 060062      MOV      $T19EXP,R2 ;EXPD ADDRESS
4150 053450 012703 000024      MOV      $20.,R3 ;NUMBER OF BYTES TO COMPARE
4151 053454 004737 011500      JSR      PC,CKMSG2 ;EXPD EQUAL RECV?
4152 053460      FORCERROR 302$,NOTSSR ;GOOD
4153 053470 103404      BCS      310$      ;BR IF YES
4154 053472      NEXT,ERRNO
4155 053472 302$: ERRHRD  ERRNO,T198CMP,MSGLOOP ;REPORT ERROR
                                TRAP      C$ERRHRD
                                .WORD     813
                                .WORD     T198CMP
                                .WORD     MSGLOOP
4156 053502 310$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
4157          ; Do a Write Subsystem Write Misc to Reset Tape Status F-FLOPS
4158 053504 004737 062020      JSR      PC,T19RSFIF ;SETUP PKT FOR WRITE MISC Reset STATUS
4159 053510 012704 062500      MOV      $T19PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4160 053514 010465 000000      MOV      R4,TSD8(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4161 053520 004737 016336      JSR      PC,CHKTSSR ;WAIT FOR SSR TO SET
4162 053524      FORCERROR 322$ ;GOODFORCE ERROR IF FORCER=1
4163 053540 103407      BCS      330$      ;BR IF CARRY SET (GOOD RETURN)
4164 053542 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
4165 053544      NEXT,ERRNO
4166 053544 322$: ERRDF  ERRNO,T192SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP      C$ERRDF
                                .WORD     814
                                .WORD     T192SSR
                                .WORD     PKTSSR
4167 053554 005237 002222      INC      FATEFLG ;SET FATAL ERROR FLAG
4168 053560 330$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
4169          ; Do a Write Subsystem READ STATUS
4170 053562 004737 062000      JSR      PC,T19SRD ;SETUP PACKET FOR READ STATUS
4171 053566 012704 062500      MOV      $T19PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET

```

N14

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 182

|      |        |        |        |        |                              |                                       |
|------|--------|--------|--------|--------|------------------------------|---------------------------------------|
| 4172 | 053572 | 010465 | 000000 |        | MOV R4,TSD8(R5)              | ;SET THE PACKET ADDRESS TO EXECUTE    |
| 4173 | 053576 | 001737 | 016336 |        | JSR PC,CHKTSSR               | ;WAIT FOR SSR TO SET                  |
| 4174 | 053602 |        |        |        | FORCERROR 342\$              | ;GOODFORCE ERROR IF FORCER=1          |
| 4175 | 053616 | 103407 |        |        | BCS 350\$                    | ;BR IF CARRY SET (GOOD RETURN)        |
| 4176 | 053620 | 010001 |        |        | MOV R0,R1                    | ;SAVE CONTENTS OF TSSR                |
| 4177 | 053622 |        |        |        | NEXT,ERRNO                   |                                       |
| 4178 | 053622 |        |        | 342\$: | ERRDF ERRNO,T193SSR,PKTSSR   | ;DEVICE FATAL SSR FAILED TO SET       |
|      | 053622 | 104455 |        |        |                              | TRAP C\$ERDF                          |
|      | 053624 | 001457 |        |        |                              | .WORD 815                             |
|      | 053626 | 060344 |        |        |                              | .WORD T193SSR                         |
|      | 053630 | 012046 |        |        |                              | .WORD PKTSSR                          |
| 4179 | 053632 | 005237 | 002222 |        | INC FATFLG                   | ;SET FATAL ERROR FLAG                 |
| 4180 | 053636 |        |        | 350\$: | CKLOOP                       | ;LOOP ON ERROR, IF FLAG SET           |
|      | 053636 | 104406 |        |        |                              | TRAP C\$CLP1                          |
| 4181 | 053640 | 004737 | 062240 |        | JSR PC,T19SETEXP             | ;SET WORDS 0-7 EXPD=RCV (NOT TESTING) |
| 4182 | 053644 | 012701 | 060102 |        | MOV #T19EXSTA,R1             | ;GET EXPECTED READ STATUS             |
| 4183 | 053650 | 012702 | 062372 |        | MOV #T198FSTA,R2             | ;GET RCV READ STATUS                  |
| 4184 | 053654 | 011211 |        |        | MOV (R2),(R1)                | ;SET EXPD WORD #8 = RCV TEMP          |
| 4185 | 053656 | 042711 | 002000 |        | BIC #S1.ICER,(R1)            | ;SET EXPD ICER =0                     |
| 4186 | 053662 | 042711 | 001000 |        | BIC #S1.IFMK,(R1)            | ;SET EXPD IFMK =0                     |
| 4187 | 053666 | 042711 | 000400 |        | BIC #S1.IHER,(R1)            | ;SET EXPD IHER =0                     |
| 4188 | 053672 | 016261 | 000002 | 000002 | MOV 2(R2),2(R1)              | ;SET EXPD WORD #9 = RCV (NOT TESTING) |
| 4189 | 053700 | 005000 |        |        | CLR R0                       | ;HIGH RCV ADDRESS FOR CKMSG2          |
| 4190 | 053702 | 012701 | 062352 |        | MOV #T198FR,R1               | ;LOW RCV ADDRESS FOR CKMSG2           |
| 4191 | 053706 | 012702 | 060062 |        | MOV #T19EXP,R2               | ;EXPD ADDRESS                         |
| 4192 | 053712 | 012703 | 000024 |        | MOV #20,R3                   | ;NUMBER OF BYTES TO COMPARE           |
| 4193 | 053716 | 004737 | 011500 |        | JSR PC,CKMSG2                | ;EXPD EQUAL RCV?                      |
| 4194 | 053722 |        |        |        | FORCERROR 362\$,NOTSSR       | ;GOOD                                 |
| 4195 | 053732 | 103404 |        |        | BCS 370\$                    | ;BR IF YES                            |
| 4196 | 053734 |        |        |        | NEXT,ERRNO                   |                                       |
| 4197 | 053734 |        |        | 362\$: | ERRHRD ERRNO,T197CMP,MSGSTAT | ;REPORT ERROR                         |
|      | 053734 | 104456 |        |        |                              | TRAP C\$ERHRD                         |
|      | 053736 | 001460 |        |        |                              | .WORD 816                             |
|      | 053740 | 061303 |        |        |                              | .WORD T197CMP                         |
|      | 053742 | 012350 |        |        |                              | .WORD MSGSTAT                         |
| 4198 | 053744 |        |        | 370\$: | CKLOOP                       | ;LOOP ON ERROR, IF FLAG SET           |
|      | 053744 | 104406 |        |        |                              | TRAP C\$CLP1                          |
| 4199 |        |        |        |        |                              |                                       |
| 4200 | 053746 | 013737 | 002312 | 060014 | MOV DATA,T19PREV             | ;SETUP PREVIOUS DATA FOR EXPD CALC.   |
| 4201 | 053754 | 013703 | 002316 |        | MOV TSTPTR,R3                | ;RESTORE CURRENT TSTBLK POINTER       |
| 4202 | 053760 | 020327 | 003062 |        | CMP R3,#TBLEND               | ;END OF TSTBLK?                       |
| 4203 | 053764 | 103002 |        |        | BHIS 400\$                   | ;BR IF YES                            |
| 4204 | 053766 | 000137 | 052762 |        | JMP 200\$                    | ;DO NEXT TSTBLK PATTERN               |
| 4205 | 053772 |        |        | 400\$: |                              |                                       |
| 4206 |        |        |        |        |                              |                                       |
| 4207 | 053772 |        |        |        | ENDSUB                       | ;////////// END SUBTEST //////////    |
|      | 053772 |        |        |        |                              | L10067:                               |
|      | 053772 | 104403 |        |        |                              | TRAP C\$ESUB                          |
| 4208 |        |        |        |        |                              |                                       |
| 4209 | 053774 | 005737 | 002222 |        | TST FATFLG                   | ;ANY FATAL ERRORS ?                   |
| 4210 | 054000 | 001402 |        |        | BEQ 460\$                    | ;BRANCH IF NOT                        |
| 4211 | 054002 | 004737 | 017202 |        | JSR PC,CKDROP                | ;TRY TO DROP THE UNIT                 |
| 4212 | 054006 |        |        | 460\$: |                              |                                       |
| 4213 |        |        |        |        |                              |                                       |
| 4214 |        |        |        |        |                              |                                       |
| 4215 |        |        |        |        |                              |                                       |
| 4216 |        |        |        |        |                              |                                       |

B15

TSV5 - HARDWARE TESTS - MACRO M1115 06 FEB-84 17:14  
TEST 8: SUBTEST 1: LOOPBACK CONTROL SIGNAL TEST

SEQ 185

4217  
4218  
4219  
4220  
4221  
4222  
4223  
4224  
4225  
4226  
4227  
4228  
4229  
4230  
4231  
4232  
4233  
4234  
4235  
4236  
4237  
4238  
4239  
4240  
4241  
4242  
4243  
4244  
4245  
4246  
4247  
4248  
4249  
4250  
4251  
4252  
4253  
4254 054006  
054006  
054006 104402  
4255  
4256 054010  
4257 054010 004737 015774  
4258 054014 103405  
4259 054016 010001  
4260 054020  
054020 104455  
054022 001460  
054024 003652  
054026 012034  
4261  
4262 054030 005037 002222  
4263 054034 012704 062330  
4264 054040 004737 010662  
4265 054044  
4266 054060 103407  
4267 054062 010001

.SBTTL TEST 8: SUBTEST 2: LOOPBACK READ/WRITE SIGNALS TEST

```

***
TEST 8: SUBTEST 2:
SUBTEST DESCRIPTION:
    This subtest verifies the Read/Write data loopback path.
    The Read/Write data signals are IR<7:0> and IW<7:0>
    respectively.
TEST STEPS:
    REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
    BEGIN
        Write to TSSR register to soft initialize the controller
        Do WRITE CHARACTERISTICS to check for Extended Features Switch
        IF Extended Features Hardware Switch Clear then:
            Do Write Subsystem Write Miscellaneous to Set Extended Features.
        Do WRITE CHARACTERISTICS to select reserved unit 7 and setup BUFFER
        Do a Write Subsystem WRITE NPR to set tape direction out and Loopback
        Do a WRITE NPR to set loopback and tape direction OUT
        Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
        Do a READ FIFO with tape direction OUT to load tape out write latch
        Do a WRITE NPR to set loopback and tape direction IN
        Do a WRITE FIFO with byte count equal to 1 and Tape direction IN
        to strobe loopback data into FIFO.
        Do a READ FIFO with tape direction IN to read data
        IF Data read from FIFO NOT= to Data sent Then Print Error
        Do a Write Subsystem READ STATUS
        IF Input Ready NOT=1 Then Print Error
        IF Output Ready NOT=0 Then Print Error
        IF Data In Miss NOT=0 Then Print Error
    END
    BGNSUB
    /////////////////////////////////////////////////// BEGIN SUBTEST 2 ///////////////////////////////////////////////////
    T8.2:
    TRAP CTRSUB
    Write to TSSR register to soft initialize the controller
51:
    JSR PC,SOFT INIT          ;WRITE TO TSSR TO SOFT INITIALIZE
    BCS 101                   ;BR IF SOFT INIT OKAY
    MOV R0,R1                 ;SAVE CONTENTS OF TSSR
    ERROF ERRNO,SFERR,SFMSG   ;DEVICE FATAL DURING INIT
                                TRAP CTRERR
                                .WORD 816
                                .WORD SFERR
                                .WORD SFMSG
101:
    Do WRITE CHARACTERISTICS to check for Extended Features Switch
    CLR FATELG                ;CLEAR FATAL ERROR FLAG
    MOV #119PACKET,R4         ;GET THE ADDRESS OF COMMAND PACKET
    JSR PC,WRITCHR            ;DO WRITE CHARACTERISTICS COMMAND
    FORCERROR 121             ;FORCE ERROR IF FORCER=1
    BCS 151                   ;BR IF CARRY SET (GOOD RETURN)
    MOV R0,R1                 ;SAVE CONTENTS OF TSSR

```

C15

ISVS HARDWARE TESTS MACRO M1113 OF FEB 84 17:14  
TEST R: SUBTEST 2: LOOPBACK READ/WRITE SIGNALS TEST

SEQ 184

```

4268 054064
4269 054064 124: NEXT,ERRNO
      054064 104455 ERRDF ERRNO,T19SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      054066 001461 TRAP C1ERDF
      054070 060243 .WORD 817
      054072 012046 .WORD T19SSR
      054072 012046 .WORD PKTSSR
4270 054074 005237 002222 154: INC FATELG ;SET FATAL ERROR FLAG
4271 054100 CKLOOP ;LOOP ON ERROR, IF FLAG SET
      054100 104406 TRAP C1CLP1
4272 ; If Extended Features Hardware Switch Clear then:
4273 ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
4274 054102 012701 062352 MOV @T19BFR,R1 ;MESSAGE BUFFER ADDRESS
4275 054106 032761 000200 000012 BIT @X2.EXTF,XST2(R1) ;EXTENDED FEATURES SWITCH SET?
4276 054114 001026 BNE 304 ;BR IF YES
4277 054116 004737 062202 JSR PC,T19SEXT ;SETUP PACKET FOR WRITE MISC INVERT
4278 054122 012704 062500 MOV @T19PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4279 054126 010465 000000 MOV R4,TSD8(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4280 054132 004737 016336 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4281 054136 FORCERROR 224 ;GOODFORCE ERROR IF FORCER=1
4282 054152 103407 BCS 304 ;BR IF CARRY SET (GOOD RETURN)
4283 054154 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4284 054156 NEXT,ERRNO
4285 054156 224: ERRDF ERRNO,T192SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      054156 104455 TRAP C1ERDF
      054160 001462 .WORD 818
      054162 060300 .WORD T192SSR
      054164 012046 .WORD PKTSSR
4286 054166 005237 002222 304: INC FATELG ;SET FATAL ERROR FLAG
4287 054172 CKLOOP ;LOOP ON ERROR, IF FLAG SET
      054172 104406 TRAP C1CLP1
4288 ; Do WRITE CHARACTERISTICS CS to select reserved unit 7
4289 054174 012704 062330 MOV @T19PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
4290 054200 004737 010662 JSR PC,WRITCHR ;DO WRITE CHARACTERISTICS COMMAND
4291 054204 FORCERROR 424 ;GOODFORCE ERROR IF FORCER=1
4292 054220 103407 BCS 504 ;BR IF CARRY SET (GOOD RETURN)
4293 054222 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4294 054224 NEXT,ERRNO
4295 054224 424: ERRDF ERRNO,T19SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      054224 104455 TRAP C1ERDF
      054226 001463 .WORD 819
      054230 060243 .WORD T19SSR
      054232 012046 .WORD PKTSSR
4296 054234 005237 002222 504: INC FATELG ;SET FATAL ERROR FLAG
4297 054240 CKLOOP ;LOOP ON ERROR, IF FLAG SET
      054240 104406 TRAP C1CLP1
4298
4299
4300 ; REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
4301 054242 012703 002752 MOV @TSTBLK,R3 ;GET FIRST PATTERN ADDRESS
4302 054246 012337 002312 1004: MOV (R3),DATA ;GET A TEST PATTERN
4303 054252 042737 177400 002312 BIC @C<377>,DATA ;DATA IS BYTE
4304 054260 010337 002316 MOV R3,TSTPTR ;SETUP CURRENT TSTBLK POINTER
4305 ; Do a WRITE NPR to set loopback and tape direction OUT
4306 054264 012700 000100 MOV @NP.OUT,R0 ;SET TAPE DIRECTION OUT
4307 054270 052700 000040 BIS @NP.LOOP,R0 ;SET LOOPBACK
4308 054274 004737 062042 JSR PC,T19SNPR ;SETUP T19PK2 FOR WRITE NPR
4309 054300 012704 062500 MOV @T19PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET

```

TSV5 - HARDWARE TESTS MACRO M1113 OF FEB 84 12:14  
 TEST 8: SCS TEST 2: LOOPBACK READ/WRITE SIGNALS TEST

SEQ 185

|      |        |        |        |            |                                                                     |                                     |
|------|--------|--------|--------|------------|---------------------------------------------------------------------|-------------------------------------|
| 4310 | 054304 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                                         | SET THE PACKET ADDRESS TO EXECUTE   |
| 4311 | 054310 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                          | WAIT FOR SSR TO SET                 |
| 4312 | 054314 |        |        | FORCERROR  | 102                                                                 | FORCE ERROR IF FORCER=1             |
| 4313 | 054330 | 103407 |        | BCS        | 105                                                                 | BR IF CARRY SET (GOOD RETURN)       |
| 4314 | 054332 | 010001 |        | MOV        | R0,R1                                                               | SAVE CONTENTS OF TSSR               |
| 4315 | 054334 |        |        | NEXT,ERRNO |                                                                     |                                     |
| 4316 | 054334 |        |        | 102:       | ERRDF ERRNO,T194SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 054334 | 104455 |        |            |                                                                     | TRAP C1ERDF                         |
|      | 054336 | 001464 |        |            |                                                                     | .WORD 820                           |
|      | 054340 | 060411 |        |            |                                                                     | .WORD T194SSR                       |
|      | 054342 | 012046 |        |            |                                                                     | .WORD PKTSSR                        |
| 4317 | 054344 | 005237 | 002222 | INC        | FATEFLG                                                             | SET FATAL ERROR FLAG                |
| 4318 | 054350 |        |        | 105:       | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 054350 | 104406 |        |            |                                                                     | TRAP C1CLP1                         |
| 4319 |        |        |        | :          | Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT   |                                     |
| 4320 | 054352 | 012700 | 000001 | MOV        | #1,R0                                                               | WRITE 1 BYTE                        |
| 4321 | 054356 | 012701 | 002312 | MOV        | #DATA,R1                                                            | FIFO WRITE DATA ADDRESS             |
| 4322 | 054362 | 004737 | 062106 | JSR        | PC,T19WFIF                                                          | SETUP T19PK2 FOR WRITE FIFO         |
| 4323 | 054366 | 012704 | 062500 | MOV        | #T19PK2,R4                                                          | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4324 | 054372 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                                         | SET THE PACKET ADDRESS TO EXECUTE   |
| 4325 | 054376 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                          | WAIT FOR SSR TO SET                 |
| 4326 | 054402 |        |        | FORCERROR  | 107                                                                 | FORCE ERROR IF FORCER=1             |
| 4327 | 054416 | 103407 |        | BCS        | 110                                                                 | BR IF CARRY SET (GOOD RETURN)       |
| 4328 | 054420 | 010001 |        | MOV        | R0,R1                                                               | SAVE CONTENTS OF TSSR               |
| 4329 | 054422 |        |        | NEXT,ERRNO |                                                                     |                                     |
| 4330 | 054422 |        |        | 107:       | ERRDF ERRNO,T195SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 054422 | 104455 |        |            |                                                                     | TRAP C1ERDF                         |
|      | 054424 | 001465 |        |            |                                                                     | .WORD 821                           |
|      | 054426 | 060454 |        |            |                                                                     | .WORD T195SSR                       |
|      | 054430 | 012046 |        |            |                                                                     | .WORD PKTSSR                        |
| 4331 | 054432 | 005237 | 002222 | INC        | FATEFLG                                                             | SET FATAL ERROR FLAG                |
| 4332 | 054436 |        |        | 110:       | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 054436 | 104406 |        |            |                                                                     | TRAP C1CLP1                         |
| 4333 |        |        |        | :          | Do a READ FIFO with tape direction OUT to load tape out write latch |                                     |
| 4334 | 054440 | 012700 | 000001 | MOV        | #1,R0                                                               | SET READ BYTE COUNT                 |
| 4335 | 054444 | 004737 | 062066 | JSR        | PC,T19RFIF                                                          | SETUP T19PK2 FOR READ FIFO          |
| 4336 | 054450 | 012704 | 062500 | MOV        | #T19PK2,R4                                                          | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4337 | 054454 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                                         | SET THE PACKET ADDRESS TO EXECUTE   |
| 4338 | 054460 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                          | WAIT FOR SSR TO SET                 |
| 4339 | 054464 |        |        | FORCERROR  | 122                                                                 | FORCE ERROR IF FORCER=1             |
| 4340 | 054500 | 103407 |        | BCS        | 130                                                                 | BR IF CARRY SET (GOOD RETURN)       |
| 4341 | 054502 | 010001 |        | MOV        | R0,R1                                                               | SAVE CONTENTS OF TSSR               |
| 4342 | 054504 |        |        | NEXT,ERRNO |                                                                     |                                     |
| 4343 | 054504 |        |        | 122:       | ERRDF ERRNO,T196SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 054504 | 104455 |        |            |                                                                     | TRAP C1ERDF                         |
|      | 054506 | 001466 |        |            |                                                                     | .WORD 822                           |
|      | 054510 | 060520 |        |            |                                                                     | .WORD T196SSR                       |
|      | 054512 | 012046 |        |            |                                                                     | .WORD PKTSSR                        |
| 4344 | 054514 | 005237 | 002222 | INC        | FATEFLG                                                             | SET FATAL ERROR FLAG                |
| 4345 | 054520 |        |        | 130:       | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 054520 | 104406 |        |            |                                                                     | TRAP C1CLP1                         |
| 4346 |        |        |        | :          | Do a WRITE NPR to set loopback and tape direction IN                |                                     |
| 4347 | 054522 | 005000 |        | CLR        | R0                                                                  | CLR NP OUT TO SET TAPE DIRECTION IN |
| 4348 | 054524 | 052700 | 000040 | BIS        | #NP,LOOP,R0                                                         | SET LOOPBACK                        |
| 4349 | 054530 | 004737 | 062042 | JSR        | PC,T19SNPR                                                          | SETUP T19PK2 FOR WRITE NPR          |
| 4350 | 054534 | 012704 | 062500 | MOV        | #T19PK2,R4                                                          | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4351 | 054540 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                                         | SET THE PACKET ADDRESS TO EXECUTE   |

TSVS - HARDWARE TESTS MACRO M1113 06-FEB 84 17:14  
 TEST 8: SUBTEST 2: LOOPBACK READ/WRITE SIGNALS TEST

SEQ 186

|      |        |        |               |               |                                                                  |                                        |
|------|--------|--------|---------------|---------------|------------------------------------------------------------------|----------------------------------------|
| 4352 | 054544 | 004737 | 016336        | JSR           | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                   |
| 4353 | 054550 |        |               | FORCERROR     | 142\$                                                            | ;BDDFORCE ERROR IF FORCER=1            |
| 4354 | 054    | 103407 |               | BCS           | 150\$                                                            | ;BR IF CARRY SET (GOOD RETURN)         |
| 4355 | 054    | 010001 |               | MOV           | R0,R1                                                            | ;SAVE CONTENTS OF TSSR                 |
| 4356 | 054570 |        |               | NEXT,ERRNO    |                                                                  |                                        |
| 4357 | 054570 |        |               | 142\$: ERRDF  | ERRNO,T194SSR,PKTSSR                                             | ;DEVICE FATAL SSR FAILED TO SET        |
|      | 054570 | 104455 |               |               |                                                                  | TRAP CIERDF                            |
|      | 054572 | 001467 |               |               |                                                                  | .WORD 823                              |
|      | 054574 | 060411 |               |               |                                                                  | .WORD T194SSR                          |
|      | 054576 | 012046 |               |               |                                                                  | .WORD PKTSSR                           |
| 4358 | 054600 | 005237 | 002222        | INC           | FATFLG                                                           | ;SET FATAL ERROR FLAG                  |
| 4359 | 054604 |        |               | 150\$: CKLOOP |                                                                  | ;LOOP ON ERROR, IF FLAG SET            |
|      | 054604 | 104406 |               |               |                                                                  | TRAP C\$CLP1                           |
| 4360 |        |        |               | ;             | Do a WRITE FIFO with byte count equal to 1 and Tape direction IN |                                        |
| 4361 | 054606 | 012700 | 000001        | MOV           | #1,R0                                                            | ;WRITE 1 BYTE                          |
| 4362 | 054612 | 012701 | 002312        | MOV           | #DATA,R1                                                         | ;FIFO WRITE DATA ADDRESS               |
| 4363 | 054616 | 004737 | 062106        | JSR           | PC,T19WFIF                                                       | ;SETUP T19PK2 FOR WRITE FIFO           |
| 4364 | 054622 | 012704 | 062500        | MOV           | #T19PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET    |
| 4365 | 054626 | 010465 | 000000        | MOV           | R4,TSD8(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE     |
| 4366 | 054632 | 004737 | 016336        | JSR           | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                   |
| 4367 | 054636 |        |               | FORCERROR     | 162\$                                                            | ;BDDFORCE ERROR IF FORCER=1            |
| 4368 | 054652 | 103407 |               | BCS           | 170\$                                                            | ;BR IF CARRY SET (GOOD RETURN)         |
| 4369 | 054654 | 010001 |               | MOV           | R0,R1                                                            | ;SAVE CONTENTS OF TSSR                 |
| 4370 | 054656 |        |               | NEXT,ERRNO    |                                                                  |                                        |
| 4371 | 054656 |        |               | 162\$: ERRDF  | ERRNO,T195SSR,PKTSSR                                             | ;DEVICE FATAL SSR FAILED TO SET        |
|      | 054656 | 104455 |               |               |                                                                  | TRAP CIERDF                            |
|      | 054660 | 001470 |               |               |                                                                  | .WORD 824                              |
|      | 054662 | 060454 |               |               |                                                                  | .WORD T195SSR                          |
|      | 054664 | 012046 |               |               |                                                                  | .WORD PKTSSR                           |
| 4372 | 054666 | 005237 | 002222        | INC           | FATFLG                                                           | ;SET FATAL ERROR FLAG                  |
| 4373 | 054672 |        |               | 170\$: CKLOOP |                                                                  | ;LOOP ON ERROR, IF FLAG SET            |
|      | 054672 | 104406 |               |               |                                                                  | TRAP C\$CLP1                           |
| 4374 |        |        |               | ;             | Do a READ FIFO with tape direction IN to read data               |                                        |
| 4375 |        |        |               | ;             | If Data read from FIFO NOT= to Data sent Then Print Error        |                                        |
| 4376 | 054674 | 012700 | 000001        | MOV           | #1,R0                                                            | ;SET READ BYTE COUNT                   |
| 4377 | 054700 | 004737 | 062066        | JSR           | PC,T19RFIF                                                       | ;SETUP T19PK2 FOR READ FIFO            |
| 4378 | 054704 | 012704 | 062500        | MOV           | #T19PK2,R4                                                       | ;GET WRITE SUBSYSTEM COMMAND PACKET    |
| 4379 | 054710 | 010465 | 000000        | MOV           | R4,TSD8(R5)                                                      | ;SET THE PACKET ADDRESS TO EXECUTE     |
| 4380 | 054714 | 004737 | 016336        | JSR           | PC,CHKTSSR                                                       | ;WAIT FOR SSR TO SET                   |
| 4381 | 054720 |        |               | FORCERROR     | 182\$                                                            | ;BDDFORCE ERROR IF FORCER=1            |
| 4382 | 054734 | 103407 |               | BCS           | 190\$                                                            | ;BR IF CARRY SET (GOOD RETURN)         |
| 4383 | 054736 | 010001 |               | MOV           | R0,R1                                                            | ;SAVE CONTENTS OF TSSR                 |
| 4384 | 054740 |        |               | NEXT,ERRNO    |                                                                  |                                        |
| 4385 | 054740 |        |               | 182\$: ERRDF  | ERRNO,T196SSR,PKTSSR                                             | ;DEVICE FATAL SSR FAILED TO SET        |
|      | 054740 | 104455 |               |               |                                                                  | TRAP CIERDF                            |
|      | 054742 | 001471 |               |               |                                                                  | .WORD 825                              |
|      | 054744 | 060520 |               |               |                                                                  | .WORD T196SSR                          |
|      | 054746 | 012046 |               |               |                                                                  | .WORD PKTSSR                           |
| 4386 | 054750 | 005237 | 002222        | INC           | FATFLG                                                           | ;SET FATAL ERROR FLAG                  |
| 4387 | 054754 |        |               | 190\$: CKLOOP |                                                                  | ;LOOP ON ERROR, IF FLAG SET            |
|      | 054754 | 104406 |               |               |                                                                  | TRAP C\$CLP1                           |
| 4388 | 054756 | 004737 | 062240        | JSR           | PC,T19SETEXP                                                     | ;SET WORDS 0-7 EXPD-RECV (NOT TESTING) |
| 4389 | 054762 | 012701 | 060102        | MOV           | #T19EXSTA,R1                                                     | ;GET EXPECTED READ STATUS              |
| 4390 | 054766 | 012702 | 062372        | MOV           | #T19RFSTA,R1                                                     | ;GET RECV READ STATUS                  |
| 4391 | 054772 | 013711 | 002312        | MOV           | DATA,(R1)                                                        | ;SET EXPD WORD #8 = DATA               |
| 4392 | 054776 | 016261 | 000002 000002 | MOV           | #(R2),2(R1)                                                      | ;SET EXPD WORD #9 = RECV (NOT TESTING) |
| 4393 | 055004 | 005000 |               | CLR           | R0                                                               | ;HIGH RECV ADDRESS FOR CKMSG2          |

F15

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 2: LOOPBACK READ/WRITE SIGNALS TEST

SEQ 187

|      |        |        |        |            |                                        |                                       |       |           |
|------|--------|--------|--------|------------|----------------------------------------|---------------------------------------|-------|-----------|
| 4394 | 055006 | 012701 | 062352 | MOV        | 0T19BFR,R1                             | LOW RECV ADDRESS FOR CKMSG2           |       |           |
| 4395 | 055012 | 012702 | 060062 | MOV        | 0T19EXP,R2                             | EXPD ADDRESS                          |       |           |
| 4396 | 055016 | 012703 | 000022 | MOV        | 018.,R3                                | NUMBER OF BYTES TO COMPARE            |       |           |
| 4397 | 055022 | 004737 | 011500 | JSR        | PC,CKMSG2                              | EXPD EQUAL RECV?                      |       |           |
| 4398 | 055026 |        |        | FORCERROR  | 202\$,NOTSSR                           | 000                                   |       |           |
| 4399 | 055036 | 103404 |        | BCS        | 210\$                                  | BR IF YES                             |       |           |
| 4400 | 055040 |        |        | NEXT,ERRNO |                                        |                                       |       |           |
| 4401 | 055040 |        |        | 202\$:     | ERRHRD ERRNO,T199CMP,MSGSUB            | REPORT ERROR                          |       |           |
|      | 055040 | 104456 |        |            |                                        |                                       | TRAP  | C\$ERRHRD |
|      | 055042 | 001472 |        |            |                                        |                                       | .WORD | 826       |
|      | 055044 | 061460 |        |            |                                        |                                       | .WORD | T199CMP   |
|      | 055046 | 013742 |        |            |                                        |                                       | .WORD | MSGSUB    |
| 4402 | 055050 |        |        | 210\$:     | CKLOOP                                 | LOOP ON ERROR, IF FLAG SET            | SET   |           |
|      | 055050 | 104406 |        |            |                                        |                                       | TRAP  | C\$CLP1   |
| 4403 |        |        |        | :          | Do a Write Subsystem READ STATUS       |                                       |       |           |
| 4404 |        |        |        | :          | If Input Ready NOT=1 Then Print Error  |                                       |       |           |
| 4405 |        |        |        | :          | If Output Ready NOT=0 Then Print Error |                                       |       |           |
| 4406 |        |        |        | :          | If Data In Miss NOT=0 Then Print Error |                                       |       |           |
| 4407 | 055052 | 004737 | 062000 | JSR        | PC,T19SRD                              | SETUP PACKET FOR READ STATUS          |       |           |
| 4408 | 055056 | 012704 | 062500 | MOV        | 0T19PK2,R4                             | GET WRITE SUBSYSTEM COMMAND PACKET    |       |           |
| 4409 | 055062 | 010465 | 000000 | MOV        | R4,TSD8(R5)                            | SET THE PACKET ADDRESS TO EXECUTE     |       |           |
| 4410 | 055066 | 004737 | 016336 | JSR        | PC,CHKTSSR                             | WAIT FOR SSR TO SET                   |       |           |
| 4411 | 055072 |        |        | FORCERROR  | 212\$                                  | 000FORCE ERROR IF FORCED=1            |       |           |
| 4412 | 055106 | 103407 |        | BCS        | 220\$                                  | BR IF CARRY SET (GOOD RETURN)         |       |           |
| 4413 | 055110 | 010001 |        | MOV        | R0,R1                                  | SAVE CONTENTS OF TSSR                 |       |           |
| 4414 | 055112 |        |        | NEXT,ERRNO |                                        |                                       |       |           |
| 4415 | 055112 |        |        | 212\$:     | ERRDF ERRNO,T193SSR,PKTSSR             | DEVICE FATAL SSR FAILED TO SET        |       |           |
|      | 055112 | 104455 |        |            |                                        |                                       | TRAP  | C\$ERRDF  |
|      | 055114 | 001473 |        |            |                                        |                                       | .WORD | 827       |
|      | 055116 | 060344 |        |            |                                        |                                       | .WORD | T193SSR   |
|      | 055120 | 012046 |        |            |                                        |                                       | .WORD | PKTSSR    |
| 4416 | 055122 | 005237 | 002222 | INC        | FATFLG                                 | SET FATAL ERROR FLAG                  |       |           |
| 4417 | 055126 |        |        | 220\$:     | CKLOOP                                 | LOOP ON ERROR, IF FLAG SET            |       |           |
|      | 055126 | 104406 |        |            |                                        |                                       | TRAP  | C\$CLP1   |
| 4418 | 055130 | 004737 | 062240 | JSR        | PC,T19SETEXP                           | SET WORDS 0-7 EXPD=RECV (NOT TESTING) |       |           |
| 4419 | 055134 | 012701 | 060102 | MOV        | 0T19EXSTA,R1                           | GET EXPECTED READ STATUS              |       |           |
| 4420 | 055140 | 012702 | 062372 | MOV        | 0T19BFA,R2                             | GET RECV READ STATUS                  |       |           |
| 4421 | 055144 | 012221 |        | MOV        | (R2), (R1)                             | SET EXPD WORD 08 = RECV TEMP          |       |           |
| 4422 | 055146 | 011211 |        | MOV        | (R2), (R1)                             | SET EXPD WORD 09 = RECV TEMP          |       |           |
| 4423 | 055150 | 052711 | 000020 | BIS        | 0S2.INRD1,(R1)                         | SET EXP INPUT READY = 1               |       |           |
| 4424 | 055154 | 042711 | 000040 | BIC        | 0S2.OUTRD1,(R1)                        | SET EXP OUTPUT READY = 0              |       |           |
| 4425 | 055160 | 042711 | 000200 | BIC        | 0S2.DIM,(R1)                           | SET EXP DATA IN MISS = 0              |       |           |
| 4426 | 055164 | 005000 |        | CLR        | R0                                     | HIGH RECV ADDRESS FOR CKMSG2          |       |           |
| 4427 | 055166 | 012701 | 062352 | MOV        | 0T19BFR,R1                             | LOW RECV ADDRESS FOR CKMSG2           |       |           |
| 4428 | 055172 | 012702 | 060062 | MOV        | 0T19EXP,R2                             | EXPD ADDRESS                          |       |           |
| 4429 | 055176 | 012703 | 000024 | MOV        | 020.,R3                                | NUMBER OF BYTES TO COMPARE            |       |           |
| 4430 | 055202 | 004737 | 011500 | JSR        | PC,CKMSG2                              | EXPD EQUAL RECV?                      |       |           |
| 4431 | 055206 |        |        | FORCERROR  | 232\$,NOTSSR                           | 000                                   |       |           |
| 4432 | 055216 | 103404 |        | BCS        | 240\$                                  | BR IF YES                             |       |           |
| 4433 | 055220 |        |        | NEXT,ERRNO |                                        |                                       |       |           |
| 4434 | 055220 |        |        | 232\$:     | ERRHRD ERRNO,T196CMP,MSGSTAT           | REPORT ERROR                          |       |           |
|      | 055220 | 104456 |        |            |                                        |                                       | TRAP  | C\$ERRHRD |
|      | 055222 | 001474 |        |            |                                        |                                       | .WORD | 828       |
|      | 055224 | 061220 |        |            |                                        |                                       | .WORD | T196CMP   |
|      | 055226 | 012350 |        |            |                                        |                                       | .WORD | MSGSTAT   |
| 4435 | 055230 |        |        | 240\$:     | CKLOOP                                 | LOOP ON ERROR, IF FLAG SET            | SET   |           |
|      | 055230 | 104406 |        |            |                                        |                                       | TRAP  | C\$CLP1   |

G15

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB 84 17:14  
TEST 8: SUBTEST 2: LOOPBACK READ/WRITE SIGNALS TEST

SEQ 188

```

4436
4437
4438
4439 055232          FORCEEXIT          255$          ;000
4440 055242 013703 002316      MOV      TSTPTR,R3      ;RESTORE CURRENT TSTBLK POINTER
4441 055246 020327 003062      CMP      R3,0TBLEND      ;END OF TSTBLK?
4442 055252 103002          BHS      255$          ;BR IF YES
4443 055254 000137 054246      JMP      100$          ;DO ANOTHER TSTBLK PATTERN
4444 055260          255$:
4445
4446 055260          ENDSUB          ;////////// END SUBTEST //////////
4447 055260 104403          ;L10070: TRAP C$ESUB
4448 055262 005737 002222      TST      FATALFG          ;ANY FATAL ERRORS?
4449 055266 001402          BEQ      260$          ;BRANCH IF NOT
4450 055270 004737 017202      JSR      PC,CKDROP          ;TRY TO DROP THE UNIT
4451 055274          260$:
4452
4453
4454          .SBTTL TEST 8: SUBTEST 3: LOOPBACK WRITE STROBE TEST
4455
4456
4457
4458
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4470
4471
4472
4473
4474
4475
4476
4477
4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489 055274          ;**
4490 055274          ; TEST 8: SUBTEST 3:
4491
4492          ; SUBTEST DESCRIPTION:
4493
4494          ; This subtest verifies the Write Strobe loopback path
4495          ; can strobe data from the FIFO to the Data lines.
4496          ; The signal IRESV3 drives IWSTR (write strobe) to write
4497          ; data from the FIFO to the tape data out latch.
4498
4499          ; TEST STEPS:
4500
4501          ; REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
4502          ; BEGIN
4503          ; Write to TSSR register to soft initialize the controller
4504          ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
4505          ; If Extended Features Hardware Switch Clear then:
4506          ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
4507          ; Do WRITE CHARACTERISTICS to select reserved unit 7 and setup BUFFER
4508          ; Do a Write Subsystem WRITE NPR to set tape direction out and Loopback
4509          ; Do a WRITE NPR to set loopback and tape direction OUT
4510          ; Do a WRITE FORMAT to set IRESV3-->IWSTR = 1
4511          ; Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
4512          ; Do a WRITE FORMAT to set IRESV3-->IWSTR = 0 to load write data latch
4513          ; Do a WRITE FORMAT to set IRESV3-->IWSTR = 1
4514          ; Do a WRITE NPR to set loopback and tape direction IN
4515          ; Do a WRITE FIFO with byte count equal to 1 and Tape direction IN
4516          ; to strobe loopback data into FIFO.
4517          ; Do a READ FIFO with tape direction IN to read data
4518          ; If Data read from FIFO NOT= to Data sent Then Print Error
4519          ; END
4520          ;**
4521          ; BGNSUB
4522          ;////////// BEGIN SUBTEST //////////
4523          ; T8.3:

```

| Address | Offset | Hex    | Label  | Comment                                                          | Trap  | Code    |
|---------|--------|--------|--------|------------------------------------------------------------------|-------|---------|
| 4490    | 055274 | 104402 |        | Write to TSSR register to soft initialize the controller         | TRAP  | C1BSUB  |
| 4491    | 055276 |        | 54:    |                                                                  |       |         |
| 4492    | 055276 | 004737 | 015774 | JSR PC,SOFINIT                                                   |       |         |
| 4493    | 055302 | 103405 |        | BCS 101                                                          |       |         |
| 4494    | 055304 | 010001 |        | MOV R0,R1                                                        |       |         |
| 4495    | 055306 |        |        | ERRDF ERRNO,SFIERR,SFIMSG                                        |       |         |
|         | 055306 | 104455 |        |                                                                  | TRAP  | C1ERDF  |
|         | 055310 | 001474 |        |                                                                  | .WORD | 828     |
|         | 055312 | 003652 |        |                                                                  | .WORD | SFIERR  |
|         | 055314 | 012034 |        |                                                                  | .WORD | SFIMSG  |
| 4496    |        |        | :      | Do WRITE CHARACTERISTICS to check for Extended Features Switch   |       |         |
| 4497    | 055316 | 005037 | 002222 | CLR FATFLG                                                       |       |         |
| 4498    | 055322 | 012704 | 062330 | MOV #T19PACKET,R4                                                |       |         |
| 4499    | 055326 | 004737 | 010662 | JSR PC,WRTCHR                                                    |       |         |
| 4500    | 055332 |        |        | FORCERROR 121                                                    |       |         |
| 4501    | 055346 | 103407 |        | BCS 151                                                          |       |         |
| 4502    | 055350 | 010001 |        | MOV R0,R1                                                        |       |         |
| 4503    | 055352 |        |        | NEXT,ERRNO                                                       |       |         |
| 4504    | 055352 |        | 12:    | ERRDF ERRNO,T19SSR,PKTSSR                                        |       |         |
|         | 055352 | 104455 |        |                                                                  | TRAP  | C1ERDF  |
|         | 055354 | 001475 |        |                                                                  | .WORD | 829     |
|         | 055356 | 060243 |        |                                                                  | .WORD | T19SSR  |
|         | 055360 | 012046 |        |                                                                  | .WORD | PKTSSR  |
| 4505    | 055362 | 005237 | 002222 | INC FATFLG                                                       |       |         |
| 4506    | 055366 |        | 15:    | CKLOOP                                                           |       |         |
|         | 055366 | 104406 |        |                                                                  | TRAP  | C1CLP1  |
| 4507    |        |        | :      | If Extended Features Hardware Switch Clear then:                 |       |         |
| 4508    |        |        | :      | Do Write Subsystem Write Miscellaneous to Set Extended Features. |       |         |
| 4509    | 055370 | 012701 | 062352 | MOV #T19BFR,R1                                                   |       |         |
| 4510    | 055374 | 032761 | 000200 | RIT #X2,EXTF,XST2(R1)                                            |       |         |
| 4511    | 055402 | 001026 |        | BNE 301                                                          |       |         |
| 4512    | 055404 | 004737 | 062202 | JSR PC,T19SEXT                                                   |       |         |
| 4513    | 055410 | 012704 | 062500 | MOV #T19PK2,R4                                                   |       |         |
| 4514    | 055414 | 010465 | 000000 | MOV R4,T5DB(R5)                                                  |       |         |
| 4515    | 055420 | 004737 | 016336 | JSR PC,CHKTSSR                                                   |       |         |
| 4516    | 055424 |        |        | FORCERROR 221                                                    |       |         |
| 4517    | 055440 | 103407 |        | BCS 301                                                          |       |         |
| 4518    | 055442 | 010001 |        | MOV R0,R1                                                        |       |         |
| 4519    | 055444 |        |        | NEXT,ERRNO                                                       |       |         |
| 4520    | 055444 |        | 22:    | ERRDF ERRNO,T192SSR,PKTSSR                                       |       |         |
|         | 055444 | 104455 |        |                                                                  | TRAP  | C1ERDF  |
|         | 055446 | 001476 |        |                                                                  | .WORD | 830     |
|         | 055450 | 060300 |        |                                                                  | .WORD | T192SSR |
|         | 055452 | 012046 |        |                                                                  | .WORD | PKTSSR  |
| 4521    | 055454 | 005237 | 002222 | INC FATFLG                                                       |       |         |
| 4522    | 055460 |        | 30:    | CKLOOP                                                           |       |         |
|         | 055460 | 104406 |        |                                                                  | TRAP  | C1CLP1  |
| 4523    |        |        | :      | Do WRITE CHARACTERISTICS to select reserved unit 7               |       |         |
| 4524    | 055462 | 012704 | 062330 | MOV #T19PACKET,R4                                                |       |         |
| 4525    | 055466 | 004737 | 010662 | JSR PC,WRTCHR                                                    |       |         |
| 4526    | 055472 |        |        | FORCERROR 421                                                    |       |         |
| 4527    | 055506 | 103407 |        | BCS 501                                                          |       |         |
| 4528    | 055510 | 010001 |        | MOV R0,R1                                                        |       |         |
| 4529    | 055512 |        |        | NEXT,ERRNO                                                       |       |         |
| 4530    | 055512 |        | 42:    | ERRDF ERRNO,T19SSR,PKTSSR                                        |       |         |
|         | 055512 | 104455 |        |                                                                  | TRAP  | C1ERDF  |

| Address | Offset | Label  | Instruction | Comment | Register   | Value                | Register | Value |
|---------|--------|--------|-------------|---------|------------|----------------------|----------|-------|
| 055514  | 001477 |        |             |         |            |                      |          |       |
| 055516  | 060243 |        |             |         |            |                      |          |       |
| 055520  | 012046 |        |             |         |            |                      |          |       |
| 4531    | 055522 | 005237 | 002222      | INC     | FATFLG     |                      |          |       |
| 4532    | 055526 |        |             | 50\$:   | CKLOOP     |                      |          |       |
| 055526  | 104406 |        |             |         |            |                      |          |       |
| 4533    |        |        |             |         |            |                      |          |       |
| 4534    |        |        |             |         |            |                      |          |       |
| 4535    | 055530 | 012703 | 002752      |         |            |                      |          |       |
| 4536    | 055534 | 012337 | 002312      | 100\$:  | MOV        | (R3)+,DATA           |          |       |
| 4537    | 055540 | 042737 | 177400      |         | BIC        | *C<377>,DATA         |          |       |
| 4538    | 055546 | 010337 | 002316      |         | MOV        | R3,TSTPTR            |          |       |
| 4539    |        |        |             |         |            |                      |          |       |
| 4540    | 055552 | 012700 | 000100      |         | MOV        | *NP.OUT,R0           |          |       |
| 4541    | 055556 | 052700 | 000040      |         | BIS        | *NP.LOOP,R0          |          |       |
| 4542    | 055562 | 004737 | 062042      |         | JSR        | PC,T19SNPR           |          |       |
| 4543    | 055566 | 012704 | 062500      |         | MOV        | *T19PK2,R4           |          |       |
| 4544    | 055572 | 010465 | 000000      |         | MOV        | R4,TSD8(R5)          |          |       |
| 4545    | 055576 | 004737 | 016336      |         | JSR        | PC,CHKTSSR           |          |       |
| 4546    | 055602 |        |             |         | FORCERROR  | 102\$                |          |       |
| 4547    | 055616 | 103407 |             |         | BCS        | 105\$                |          |       |
| 4548    | 055620 | 010001 |             |         | MOV        | R0,R1                |          |       |
| 4549    | 055622 |        |             |         | NEXT,ERRNO |                      |          |       |
| 4550    | 055622 |        |             | 102\$:  | ERRDF      | ERRNO,T194SSR,PKTSSR |          |       |
| 055622  | 104455 |        |             |         |            |                      |          |       |
| 055624  | 001500 |        |             |         |            |                      |          |       |
| 055626  | 060411 |        |             |         |            |                      |          |       |
| 055630  | 012046 |        |             |         |            |                      |          |       |
| 4551    | 055632 | 005237 | 002222      |         | INC        | FATFLG               |          |       |
| 4552    | 055636 |        |             | 105\$:  | CKLOOP     |                      |          |       |
| 055636  | 104406 |        |             |         |            |                      |          |       |
| 4553    |        |        |             |         |            |                      |          |       |
| 4554    | 055640 | 012700 | 000002      |         | MOV        | *WF.I3RES,R0         |          |       |
| 4555    | 055644 | 004737 | 062162      |         | JSR        | PC,T19WFMT           |          |       |
| 4556    | 055650 | 012704 | 062500      |         | MOV        | *T19PK2,R4           |          |       |
| 4557    | 055654 | 010465 | 000000      |         | MOV        | R4,TSD8(R5)          |          |       |
| 4558    | 055660 | 004737 | 016336      |         | JSR        | PC,CHKTSSR           |          |       |
| 4559    | 055664 |        |             |         | FORCERROR  | 112\$                |          |       |
| 4560    | 055700 | 103407 |             |         | BCS        | 120\$                |          |       |
| 4561    | 055702 | 010001 |             |         | MOV        | R0,R1                |          |       |
| 4562    | 055704 |        |             |         | NEXT,ERRNO |                      |          |       |
| 4563    | 055704 |        |             | 112\$:  | ERRDF      | ERRNO,T198SSR,PKTSSR |          |       |
| 055704  | 104455 |        |             |         |            |                      |          |       |
| 055706  | 001501 |        |             |         |            |                      |          |       |
| 055710  | 060632 |        |             |         |            |                      |          |       |
| 055712  | 012046 |        |             |         |            |                      |          |       |
| 4564    | 055714 | 005237 | 002222      |         | INC        | FATFLG               |          |       |
| 4565    | 055720 |        |             | 120\$:  | CKLOOP     |                      |          |       |
| 055720  | 104406 |        |             |         |            |                      |          |       |
| 4566    |        |        |             |         |            |                      |          |       |
| 4567    | 055722 | 012700 | 000001      |         | MOV        | *1,R0                |          |       |
| 4568    | 055726 | 012701 | 002312      |         | MOV        | *DATA,R1             |          |       |
| 4569    | 055732 | 004737 | 062106      |         | JSR        | PC,T19WFIF           |          |       |
|         |        |        |             |         |            |                      |          |       |

J15

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 3: LOOPBACK WRITE STROBE TEST

SEQ 191

|      |        |        |        |            |                                                      |  |                                      |
|------|--------|--------|--------|------------|------------------------------------------------------|--|--------------------------------------|
| 4574 | 055766 | 103407 |        | BCS        | 140\$                                                |  | ;BR IF CARRY SET (GOOD RETURN)       |
| 4575 | 055770 | 010001 |        | MOV        | R0,R1                                                |  | ;SAVE CONTENTS OF TSSR               |
| 4576 | 055772 |        |        | NEXT,ERRNO |                                                      |  |                                      |
| 4577 | 055772 |        | 132\$: | ERRDF      | ERRNO,T195SSR,PKTSSR                                 |  | ;DEVICE FATAL SSR FAILED TO SET      |
|      | 055772 | 104455 |        |            |                                                      |  | TRAP C\$ERDF                         |
|      | 055774 | 001502 |        |            |                                                      |  | .WORD 834                            |
|      | 055776 | 060454 |        |            |                                                      |  | .WORD T195SSR                        |
|      | 056000 | 012046 |        |            |                                                      |  | .WORD PKTSSR                         |
| 4578 | 056002 | 005237 | 002222 | INC        | FATFLG                                               |  | ;SET FATAL ERROR FLAG                |
| 4579 | 056006 |        |        | CKLOOP     |                                                      |  | ;LOOP ON ERROR, IF FLAG SET          |
|      | 056006 | 104406 |        |            |                                                      |  | TRAP C\$CLP1                         |
| 4580 |        |        |        |            | Do a WRITE FORMAT to set IRESV3-->IWSTR = 0          |  |                                      |
| 4581 | 056010 | 005000 |        | CLR        | R0                                                   |  | ;SET IRESV3-->IWSTR=0                |
| 4582 | 056012 | 004737 | 062162 | JSR        | PC,T19WFMT                                           |  | ;SETUP T9PK2 FOR WRITE FORMAT        |
| 4583 | 056016 | 012704 | 062500 | MOV        | T19PK2,R4                                            |  | ;GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4584 | 056022 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                          |  | ;SET THE PACKET ADDRESS TO EXECUTE   |
| 4585 | 056026 | 004737 | 016336 | JSR        | PC,CHKTSSR                                           |  | ;WAIT FOR SSR TO SET                 |
| 4586 | 056032 |        |        | FORCERROR  | 152\$                                                |  | ;DO FORCE ERROR IF FORCER=1          |
| 4587 | 056046 | 103407 |        | BCS        | 160\$                                                |  | ;BR IF CARRY SET (GOOD RETURN)       |
| 4588 | 056050 | 010001 |        | MOV        | R0,R1                                                |  | ;SAVE CONTENTS OF TSSR               |
| 4589 | 056052 |        |        | NEXT,ERRNO |                                                      |  |                                      |
| 4590 | 056052 |        | 152\$: | ERRDF      | ERRNO,T198SSR,PKTSSR                                 |  | ;DEVICE FATAL SSR FAILED TO SET      |
|      | 056052 | 104455 |        |            |                                                      |  | TRAP C\$ERDF                         |
|      | 056054 | 001503 |        |            |                                                      |  | .WORD 835                            |
|      | 056056 | 060632 |        |            |                                                      |  | .WORD T198SSR                        |
|      | 056060 | 012046 |        |            |                                                      |  | .WORD PKTSSR                         |
| 4591 | 056062 | 005237 | 002222 | INC        | FATFLG                                               |  | ;SET FATAL ERROR FLAG                |
| 4592 | 056066 |        |        | CKLOOP     |                                                      |  | ;LOOP ON ERROR, IF FLAG SET          |
|      | 056066 | 104406 |        |            |                                                      |  | TRAP C\$CLP1                         |
| 4593 |        |        |        |            | Do a WRITE FORMAT to set IRESV3-->IWSTR = 1          |  |                                      |
| 4594 | 056070 | 012700 | 000002 | MOV        | WF.IRES,R0                                           |  | ;IRESV3-->IWSTR=1                    |
| 4595 | 056074 | 004737 | 062162 | JSR        | PC,T19WFMT                                           |  | ;SETUP T9PK2 FOR WRITE FORMAT        |
| 4596 | 056100 | 012704 | 062500 | MOV        | T19PK2,R4                                            |  | ;GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4597 | 056104 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                          |  | ;SET THE PACKET ADDRESS TO EXECUTE   |
| 4598 | 056110 | 004737 | 016336 | JSR        | PC,CHKTSSR                                           |  | ;WAIT FOR SSR TO SET                 |
| 4599 | 056114 |        |        | FORCERROR  | 172\$                                                |  | ;DO FORCE ERROR IF FORCER=1          |
| 4600 | 056130 | 103407 |        | BCS        | 180\$                                                |  | ;BR IF CARRY SET (GOOD RETURN)       |
| 4601 | 056132 | 010001 |        | MOV        | R0,R1                                                |  | ;SAVE CONTENTS OF TSSR               |
| 4602 | 056134 |        |        | NEXT,ERRNO |                                                      |  |                                      |
| 4603 | 056134 |        | 172\$: | ERRDF      | ERRNO,T198SSR,PKTSSR                                 |  | ;DEVICE FATAL SSR FAILED TO SET      |
|      | 056134 | 104455 |        |            |                                                      |  | TRAP C\$ERDF                         |
|      | 056136 | 001504 |        |            |                                                      |  | .WORD 836                            |
|      | 056140 | 060632 |        |            |                                                      |  | .WORD T198SSR                        |
|      | 056142 | 012046 |        |            |                                                      |  | .WORD PKTSSR                         |
| 4604 | 056144 | 005237 | 002222 | INC        | FATFLG                                               |  | ;SET FATAL ERROR FLAG                |
| 4605 | 056150 |        |        | CKLOOP     |                                                      |  | ;LOOP ON ERROR, IF FLAG SET          |
|      | 056150 | 104406 |        |            |                                                      |  | TRAP C\$CLP1                         |
| 4606 |        |        |        |            | Do a WRITE NPR to set loopback and tape direction IN |  |                                      |
| 4607 |        |        |        | CLR        | R0                                                   |  | ;CLR NP.OUT TO SET TAPE DIRECTION IN |
| 4608 | 056152 | 005000 |        | BIS        | NP.LOOP,R0                                           |  | ;SET LOOPBACK                        |
| 4609 | 056154 | 052700 | 000040 | JSR        | PC,T19SNPR                                           |  | ;SETUP T19PK2 FOR WRITE NPR          |
| 4610 | 056160 | 004737 | 062042 | MOV        | T19PK2,R4                                            |  | ;GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4611 | 056164 | 012704 | 062500 | MOV        | R4,TSD8(R5)                                          |  | ;SET THE PACKET ADDRESS TO EXECUTE   |
| 4612 | 056170 | 010465 | 000000 | JSR        | PC,CHKTSSR                                           |  | ;WAIT FOR SSR TO SET                 |
| 4613 | 056174 | 004737 | 016336 | FORCERROR  | 182\$                                                |  | ;DO FORCE ERROR IF FORCER=1          |
| 4614 | 056200 |        |        | BCS        | 190\$                                                |  | ;BR IF CARRY SET (GOOD RETURN)       |
| 4615 | 056214 | 103407 |        |            |                                                      |  |                                      |

FLOPS

K15

TSV5 - HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 8: SUBTEST 3: LOOPBACK WRITE STROBE TEST

SEQ 192

|      |        |        |        |                                                                  |                                      |
|------|--------|--------|--------|------------------------------------------------------------------|--------------------------------------|
| 4616 | 056216 | 010001 |        | MOV R0,R1                                                        | SAVE CONTENTS OF TSSR                |
| 4617 | 056220 |        |        | NEXT,ERRNO                                                       |                                      |
| 4618 | 056220 |        | 182\$: | ERRDF ERRNO,T194SSR,PKTSSR                                       | DEVICE FATAL SSR FAILED TO SET       |
|      | 056220 | 104455 |        |                                                                  | TRAP C\$ERDF                         |
|      | 056222 | 001505 |        |                                                                  | .WORD 837                            |
|      | 056224 | 060411 |        |                                                                  | .WORD T194SSR                        |
|      | 056226 | 012046 |        |                                                                  | .WORD PKTSSR                         |
| 4619 | 056230 | 005237 | 002222 | INC FATFLG                                                       | SET FATAL ERROR FLAG                 |
| 4620 | 056234 |        | 190\$: | CKLOOP                                                           | LOOP ON ERROR, IF FLAG SET           |
|      | 056234 | 104406 |        |                                                                  | TRAP C\$CLP1                         |
| 4621 |        |        | :      | Do a WRITE FIFO with byte count equal to 1 and Tape direction IN |                                      |
| 4622 | 056236 | 012700 | 000001 | MOV #1,R0                                                        | WRITE 1 BYTE                         |
| 4623 | 056242 | 012701 | 002312 | MOV #DATA,R1                                                     | FIFO WRITE DATA ADDRESS              |
| 4624 | 056246 | 004737 | 062106 | JSR PC,T19WFIF                                                   | SETUP T19PK2 FOR WRITE FIFO          |
| 4625 | 056252 | 012704 | 062500 | MOV #T19PK2,R4                                                   | GET WRITE SUBSYSTEM COMMAND PACKET   |
| 4626 | 056256 | 010465 | 000000 | MOV R4,TSDB(R5)                                                  | SET THE PACKET ADDRESS TO EXECUTE    |
| 4627 | 056262 | 004737 | 016336 | JSR PC,CHKTSSR                                                   | WAIT FOR SSR TO SET                  |
| 4628 | 056266 |        |        | FORCERROR 202\$                                                  | FORCE ERROR IF FORCER=1              |
| 4629 | 056302 | 103407 |        | BCS 210\$                                                        | BR IF CARRY SET (GOOD RETURN)        |
| 4630 | 056304 | 010001 |        | MOV R0,R1                                                        | SAVE CONTENTS OF TSSR                |
| 4631 | 056306 |        |        | NEXT,ERRNO                                                       |                                      |
| 4632 | 056306 |        | 202\$: | ERRDF ERRNO,T195SSR,PKTSSR                                       | DEVICE FATAL SSR FAILED TO SET       |
|      | 056306 | 104455 |        |                                                                  | TRAP C\$ERDF                         |
|      | 056310 | 001506 |        |                                                                  | .WORD 838                            |
|      | 056312 | 060454 |        |                                                                  | .WORD T195SSR                        |
|      | 056314 | 012046 |        |                                                                  | .WORD PKTSSR                         |
| 4633 | 056316 | 005237 | 002222 | INC FATFLG                                                       | SET FATAL ERROR FLAG                 |
| 4634 | 056322 |        | 210\$: | CKLOOP                                                           | LOOP ON ERROR, IF FLAG SET           |
|      | 056322 | 104406 |        |                                                                  | TRAP C\$CLP1                         |
| 4635 |        |        | :      | Do a READ FIFO with tape direction IN to read data               |                                      |
| 4636 | 056324 | 012700 | 000001 | MOV #1,R0                                                        | SET READ BYTE COUNT                  |
| 4637 | 056330 | 004737 | 062066 | JSR PC,T19RFIF                                                   | SETUP T19PK2 FOR READ FIFO           |
| 4638 | 056334 | 012704 | 062500 | MOV #T19PK2,R4                                                   | GET WRITE SUBSYSTEM COMMAND PACKET   |
| 4639 | 056340 | 010465 | 000000 | MOV R4,TSDB(R5)                                                  | SET THE PACKET ADDRESS TO EXECUTE    |
| 4640 | 056344 | 004737 | 016336 | JSR PC,CHKTSSR                                                   | WAIT FOR SSR TO SET                  |
| 4641 | 056350 |        |        | FORCERROR 222\$                                                  | FORCE ERROR IF FORCER=1              |
| 4642 | 056364 | 103407 |        | BCS 230\$                                                        | BR IF CARRY SET (GOOD RETURN)        |
| 4643 | 056366 | 010001 |        | MOV R0,R1                                                        | SAVE CONTENTS OF TSSR                |
| 4644 | 056370 |        |        | NEXT,ERRNO                                                       |                                      |
| 4645 | 056370 |        | 222\$: | ERRDF ERRNO,T196SSR,PKTSSR                                       | DEVICE FATAL SSR FAILED TO SET       |
|      | 056370 | 104455 |        |                                                                  | TRAP C\$ERDF                         |
|      | 056372 | 001507 |        |                                                                  | .WORD 839                            |
|      | 056374 | 060520 |        |                                                                  | .WORD T196SSR                        |
|      | 056376 | 012046 |        |                                                                  | .WORD PKTSSR                         |
| 4646 | 056400 | 005237 | 002222 | INC FATFLG                                                       | SET FATAL ERROR FLAG                 |
| 4647 | 056404 |        | 230\$: | CKLOOP                                                           | LOOP ON ERROR, IF FLAG SET           |
|      | 056404 | 104406 |        |                                                                  | TRAP C\$CLP1                         |
| 4648 |        |        | :      | If Data read from FIFO NOT= to Data sent Then Print Error        |                                      |
| 4649 | 056406 | 004737 | 062240 | JSR PC,T19SETEXP                                                 | SET WORDS 0-7 EXPD=RCV (NOT TESTING) |
| 4650 | 056412 | 012701 | 060102 | MOV #T19EXSTA,R1                                                 | GET EXPECTED READ STATUS             |
| 4651 | 056416 | 012702 | 062372 | MOV #T19BFSTA,R2                                                 | GET RCV READ STATUS                  |
| 4652 | 056422 | 013711 | 002312 | MOV DATA,(R1)                                                    | SET EXPD WORD #8 = DATA              |
| 4653 | 056426 | 016261 | 000002 | MOV 2(R2),2(R1)                                                  | SET EXPD WORD #9 = RCV (NOT TESTING) |
| 4654 | 056434 | 005000 |        | CLR R0                                                           | HIGH RCV ADDRESS FOR CKMSG2          |
| 4655 | 056436 | 012701 | 062352 | MOV #T19BFR,R1                                                   | LOW RCV ADDRESS FOR CKMSG2           |
| 4656 | 056442 | 012702 | 060062 | MOV #T19EXP,R2                                                   | EXPD ADDRESS                         |
| 4657 | 056446 | 012703 | 000022 | MOV #18,R3                                                       | NUMBER OF BYTES TO COMPARE           |

L15

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 3: LOOPBACK WRITE STROBE TEST

SEQ 193

```

4658 056452 004737 011500      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
4659 056456      FORCERROR 242$,NOTSSR ;BDD
4660 056466 103404      BCS      250$      ;BR IF YES
4661 056470      NEXT.ERRNO
4662 056470      242$:  ERRHRD  ERRNO,T19WSTR,MSGSUB ;REPORT ERROR
                                TRAP      C$ERHRD
                                .WORD     840
                                .WORD     T19WSTR
                                .WORD     MSGSUB
4663 056500      250$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
                                TRAP      C$CLP1
                                .WORD     840
                                .WORD     T19WSTR
                                .WORD     MSGSUB
4664 056500 104406
4665
4666 056502      FORCEEXIT 255$      ;BDD
4667 056512 013703 002316      MOV      TSTPTR,R3      ;RESTORE CURRENT TSTBLK POINTER
4668 056516 020327 003062      CMP      R3,$TBLEND    ;END OF TSTBLK?
4669 056522 103002      BHS      255$      ;BR IF YES
4670 056524 000137 055534      JMP      100$      ;DO ANOTHER TSTBLK PATTERN
4671 056530      255$:
4672
4673 056530      ENDSUB      ;////////// END SUBTEST ///////////
                                L10071:
                                TRAP      C$ESUB
056530 104403
4674
4675 056532 005737 002222      TST      FATFLG      ;ANY FATAL ERRORS ?
4676 056536 001402      BEQ      260$      ;BRANCH IF NOT
4677 056540 004737 017202      JSR      PC,CKDROP    ;TRY TO DROP THE UNIT
4678 056544      260$:
4679      .SBTTL  TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST
4680
4681
4682 ; **
4683 ; TEST 8: SUBTEST 4:
4684 ;
4685 ; SUBTEST DESCRIPTION:
4686 ;
4687 ; This subtest verifies the Read Strobe loopback path
4688 ; can strobe the data from the Data lines to the FIFO.
4689 ; The signal IRESV4 drives IRSTR (read strobe) to write
4690 ; from the data lines to the FIFO.
4691 ;
4692 ; TEST STEPS:
4693 ;
4694 ; REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
4695 ; BEGIN
4696 ; Write to ISSR register to soft initialize the controller
4697 ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
4698 ; If Extended Features Hardware Switch Clear then:
4699 ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
4700 ; Do WRITE CHARACTERISTICS to select reserved unit 7 and setup BUFFER
4701 ; Do a Write Subsystem WRITE NPR to set tape direction out and Loopback
4702 ; Do a WRITE NPR to set loopback and tape direction OUT
4703 ; Do a WRITE FORMAT to set IRESV4==>IRSTR = 1
4704 ; Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
4705 ; Do a READ FIFO with tape direction OUT to load tape out write latch
4706 ; Do a WRITE NPR to set loopback and tape direction IN
4707 ; Do a WRITE FORMAT to set IRESV4==>IRSTR = 0 to write loop data to FIFO

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 194

```

4708 ; Do a WRITE FORMAT to set IRESV4==>IRSTR = 1
4709 ; (to strobe loopback data into FIFO.)
4710 ; Do a READ FIFO with tape direction IN to read data
4711 ; If Data read from FIFO NOT= to Data sent Then Print Error
4712 ; END
4713 ;--
4714 056544 BGNSUB ;////////// BEGIN SUBTEST //////////
      056544 T8.4:
      056544 104402 TRAP C$BSUB
4715 ; Write to TSSR register to soft initialize the controller
4716 056546 5$:
4717 056546 004737 015774 JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
4718 056552 103405 BCS 10$ ;BR IF SOFT INIT OKAY
4719 056554 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4720 056556 ERRDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
      056556 104455 TRAP C$ERDF
      056560 001510 .WORD 840
      056562 003652 .WORD SFIERR
      056564 012034 .WORD SFIMSG
4721 ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
4722 056566 005037 002222 10$: CLR FATFLG ;CLEAR FATAL ERROR FLAG
4723 056572 012704 062330 MOV #T19PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
4724 056576 004737 010662 JSR PC,WRTCHR ;DO WRITE CHARACTERISTICS COMMAND
4725 056602 FORCERROR 12$ ;000FORCE ERROR IF FORCER=1
4726 056616 103407 BCS 15$ ;BR IF CARRY SET (GOOD RETURN)
4727 056620 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4728 056622 NEXT,ERRNO
4729 056622 12$: ERRDF ERRNO,T19SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      056622 104455 TRAP C$ERDF
      056624 001511 .WORD 841
      056626 060243 .WORD T19SSR
      056630 012046 .WORD PKTSSR
4730 056632 005237 002222 INC FATFLG ;SET FATAL ERROR FLAG
4731 056636 15$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      056636 104406 TRAP C$CLP1
4732 ; If Extended Features Hardware Switch Clear then:
4733 ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
4734 056640 012701 062352 MOV #T198FR,R1 ;MESSAGE BUFFER ADDRESS
4735 056644 032761 000200 000012 BIT #X2,EXTF,XST2(R1) ;EXTENDED FEATURES SWITCH SET?
4736 056652 001026 BNE 30$ ;BR IF YES
4737 056654 004737 062202 JSR PC,T19SEXT ;SETUP PACKET FOR WRITE MISC INVERT
4738 056660 012704 062500 MOV #T19PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4739 056664 010465 000000 MOV R4,TSDR(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4740 056670 004737 016336 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4741 056674 FORCERROR 22$ ;000FORCE ERROR IF FORCER=1
4742 056710 103407 BCS 30$ ;BR IF CARRY SET (GOOD RETURN)
4743 056712 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4744 056714 NEXT,ERRNO
4745 056714 22$: ERRDF ERRNO,T192SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      056714 104455 TRAP C$ERDF
      056716 001512 .WORD 842
      056720 060300 .WORD T192SSR
      056722 012046 .WORD PKTSSR
4746 056724 005237 002222 INC FATFLG ;SET FATAL ERROR FLAG
4747 056730 30$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      056730 104406 TRAP C$CLP1
4748 ; Do WRITE CHARACTERISTICS to select reserved unit 7

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 8: SUBTES 4 LOOPBACK READ STROBE TEST

SEQ 195

|      |        |        |        |        |                                                       |                      |       |                                     |
|------|--------|--------|--------|--------|-------------------------------------------------------|----------------------|-------|-------------------------------------|
| 4749 | 056732 | 012704 | 062330 |        | MOV                                                   | 0T19PACKET,R4        |       | ;GET THE ADDRESS OF COMMAND PACKET  |
| 4750 | 056736 | 004737 | 010662 |        | JSR                                                   | PC,WRTCHR            |       | ;DO WRITE CHARACTERISTICS COMMAND   |
| 4751 | 056742 |        |        |        | FORCERROR                                             | 42\$                 |       | ;000FORCE ERROR IF FORCER=1         |
| 4752 | 056756 | 103407 |        |        | BCS                                                   | 50\$                 |       | ;BR IF CARRY SET (GOOD RETURN)      |
| 4753 | 056760 | 010001 |        |        | MOV                                                   | R0,R1                |       | ;SAVE CONTENTS OF TSSR              |
| 4754 | 056762 |        |        |        | NEXT,ERRNO                                            |                      |       |                                     |
| 4755 | 056762 |        |        | 42\$:  | ERRDF                                                 | ERRNO,T19SSR,PKTSSR  |       | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 056762 | 104455 |        |        |                                                       |                      | TRAP  | C\$ERDF                             |
|      | 056764 | 001513 |        |        |                                                       |                      | .WORD | 843                                 |
|      | 056766 | 060243 |        |        |                                                       |                      | .WORD | T19SSR                              |
|      | 056770 | 012046 |        |        |                                                       |                      | .WORD | PKTSSR                              |
| 4756 | 056772 | 005237 | 002222 |        | INC                                                   | FATFLG               |       | ;SET FATAL ERROR FLAG               |
| 4757 | 056776 |        |        | 50\$:  | CKLOOP                                                |                      |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 056776 | 104406 |        |        |                                                       |                      | TRAP  | C\$CLP1                             |
| 4758 |        |        |        |        |                                                       |                      |       |                                     |
| 4759 |        |        |        |        |                                                       |                      |       |                                     |
| 4760 | 057000 | 012703 | 002752 |        |                                                       |                      |       |                                     |
| 4761 | 057004 | 012337 | 002312 |        | MOV                                                   | 0TSTBLK,R3           |       | ;GET FIRST PATTERN ADDRESS          |
| 4762 | 057010 | 042737 | 177400 | 002312 | 100\$:                                                | (R3)+,DATA           |       | ;GET A TEST PATTERN                 |
| 4763 | 057016 | 010337 | 002316 |        | BIC                                                   | 0+C<377>,DATA        |       | ;DATA IS BYTE                       |
| 4764 |        |        |        |        | MOV                                                   | R3,TSTPTR            |       | ;SETUP CURRENT TSTBLK POINTER       |
| 4765 | 057022 | 012700 | 000100 |        |                                                       |                      |       |                                     |
| 4766 | 057026 | 052700 | 000040 |        | Do a WRITE NPR to set loopback and tape direction OUT |                      |       |                                     |
| 4767 | 057032 | 004737 | 062042 |        | MOV                                                   | 0NP.OUT,R0           |       | ;SET TAPE DIRECTION OUT             |
| 4768 | 057036 | 012704 | 062500 |        | BIS                                                   | 0NP.LOOP,R0          |       | ;SET LOOPBACK                       |
| 4769 | 057042 | 010465 | 000000 |        | JSR                                                   | PC,T19SNPR           |       | ;SETUP T19PK2 FOR WRITE NPR         |
| 4770 | 057046 | 004737 | 016336 |        | MOV                                                   | 0T19PK2,R4           |       | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 4771 | 057052 |        |        |        | MOV                                                   | R4,TSD8(R5)          |       | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 4772 | 057066 | 103407 |        |        | JSR                                                   | PC,CHKTSSR           |       | ;WAIT FOR SSR TO SET                |
| 4773 | 057070 | 010001 |        |        | FORCERROR                                             | 102\$                |       | ;000FORCE ERROR IF FORCER=1         |
| 4774 | 057072 |        |        |        | BCS                                                   | 105\$                |       | ;BR IF CARRY SET (GOOD RETURN)      |
| 4775 | 057072 |        |        |        | MOV                                                   | R0,R1                |       | ;SAVE CONTENTS OF TSSR              |
|      | 057072 |        |        | 102\$: | NEXT,ERRNO                                            |                      |       |                                     |
|      | 057072 | 104455 |        |        | ERRDF                                                 | ERRNO,T194SSR,PKTSSR |       | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 057072 | 001514 |        |        |                                                       |                      | TRAP  | C\$ERDF                             |
|      | 057072 | 060411 |        |        |                                                       |                      | .WORD | 844                                 |
|      | 057100 | 012046 |        |        |                                                       |                      | .WORD | T194SSR                             |
|      | 057100 |        |        |        |                                                       |                      | .WORD | PKTSSR                              |
| 4776 | 057102 | 005237 | 002222 |        | INC                                                   | FATFLG               |       | ;SET FATAL ERROR FLAG               |
| 4777 | 057106 |        |        | 105\$: | CKLOOP                                                |                      |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 057106 | 104406 |        |        |                                                       |                      | TRAP  | C\$CLP1                             |
| 4778 |        |        |        |        |                                                       |                      |       |                                     |
| 4779 | 057110 | 012700 | 000001 |        | Do a WRITE FORMAT to set IRESV4==>IRSTR = 1           |                      |       |                                     |
| 4780 | 057114 | 004737 | 062162 |        | MOV                                                   | 0WF.I4RES,R0         |       | ;IRESV4==>IRSTR=1                   |
| 4781 | 057120 | 012704 | 062500 |        | JSR                                                   | PC,T19WFM1           |       | ;SETUP T19PK2 FOR WRITE FORMAT      |
| 4782 | 057124 | 010465 | 000000 |        | MOV                                                   | 0T19PK2,R4           |       | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 4783 | 057130 | 004737 | 016336 |        | MOV                                                   | R4,TSD8(R5)          |       | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 4784 | 057134 |        |        |        | JSR                                                   | PC,CHKTSSR           |       | ;WAIT FOR SSR TO SET                |
| 4785 | 057150 | 103407 |        |        | FORCERROR                                             | 112\$                |       | ;000FORCE ERROR IF FORCER=1         |
| 4786 | 057152 | 010001 |        |        | BCS                                                   | 120\$                |       | ;BR IF CARRY SET (GOOD RETURN)      |
| 4787 | 057154 |        |        |        | MOV                                                   | R0,R1                |       | ;SAVE CONTENTS OF TSSR              |
| 4788 | 057154 |        |        |        | NEXT,ERRNO                                            |                      |       |                                     |
|      | 057154 | 104455 |        | 112\$: | ERRDF                                                 | ERRNO,T198SSR,PKTSSR |       | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 057156 | 001515 |        |        |                                                       |                      | TRAP  | C\$ERDF                             |
|      | 057160 | 060632 |        |        |                                                       |                      | .WORD | 845                                 |
|      | 057162 | 012046 |        |        |                                                       |                      | .WORD | T198SSR                             |
|      | 057162 |        |        |        |                                                       |                      | .WORD | PKTSSR                              |
| 4789 | 057164 | 005237 | 002222 |        | INC                                                   | FATFLG               |       | ;SET FATAL ERROR FLAG               |
| 4790 | 057170 |        |        | 120\$: | CKLOOP                                                |                      |       | ;LOOP ON ERROR, IF FLAG SET         |
|      | 057170 | 104406 |        |        |                                                       |                      | TRAP  | C\$CLP1                             |

TSV: HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 196

|      |        |        |        |       |                                                                     |                                     |
|------|--------|--------|--------|-------|---------------------------------------------------------------------|-------------------------------------|
| 4791 |        |        |        | 1     | Do a WRITE FIFO with byte count equal to 1 and tape direction OUT   |                                     |
| 4792 | 057172 | 012700 | 000001 |       | MOV #1,R0                                                           | WRITE 1 BYTE                        |
| 4793 | 057176 | 012701 | 002312 |       | MOV #DATA,R1                                                        | FIFO WRITE DATA ADDRESS             |
| 4794 | 057202 | 004737 | 062106 |       | JSR PC,T19WFIF                                                      | SETUP T19PK2 FOR WRITE FIFO         |
| 4795 | 057206 | 012704 | 062500 |       | MOV #T19PK2,R4                                                      | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4796 | 057212 | 010465 | 000000 |       | MOV R4,TSD8(R5)                                                     | SET THE PACKET ADDRESS TO EXECUTE   |
| 4797 | 057216 | 004737 | 016336 |       | JSR PC,CHKTSSR                                                      | WAIT FOR SSR TO SET                 |
| 4798 | 057222 |        |        |       | FORCERROR 1321                                                      | FORCE ERROR IF FORCER=1             |
| 4799 | 057236 | 103407 |        |       | BCS 1401                                                            | BR IF CARRY SET (GOOD RETURN)       |
| 4800 | 057240 | 010001 |        |       | MOV R0,R1                                                           | SAVE CONTENTS OF TSSR               |
| 4801 | 057242 |        |        |       | NEXT,ERRNO                                                          |                                     |
| 4802 | 057242 |        |        | 1321: | ERRDF ERRNO,T195SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 057242 | 104455 |        |       |                                                                     | TRAP C1ERDF                         |
|      | 057244 | 001516 |        |       |                                                                     | .WORD 846                           |
|      | 057246 | 060454 |        |       |                                                                     | .WORD T195SSR                       |
|      | 057250 | 012046 |        |       |                                                                     | .WORD PKTSSR                        |
| 4803 | 057252 | 005237 | 002222 |       | INC FATFLG                                                          | SET FATAL ERROR FLAG                |
| 4804 | 057256 |        |        | 1401: | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 057256 | 104406 |        |       |                                                                     | TRAP C1CLP1                         |
| 4805 |        |        |        | 1     | Do a READ FIFO with tape direction OUT to load tape out write latch |                                     |
| 4806 | 057260 | 012700 | 000001 |       | MOV #1,R0                                                           | SET READ BYTE COUNT                 |
| 4807 | 057264 | 004737 | 062066 |       | JSR PC,T19RFIF                                                      | SETUP T19PK2 FOR READ FIFO          |
| 4808 | 057270 | 012704 | 062500 |       | MOV #T19PK2,R4                                                      | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4809 | 057274 | 010465 | 000000 |       | MOV R4,TSD8(R5)                                                     | SET THE PACKET ADDRESS TO EXECUTE   |
| 4810 | 057300 | 004737 | 016336 |       | JSR PC,CHKTSSR                                                      | WAIT FOR SSR TO SET                 |
| 4811 | 057304 |        |        |       | FORCERROR 1521                                                      | FORCE ERROR IF FORCER=1             |
| 4812 | 057320 | 103407 |        |       | BCS 1601                                                            | BR IF CARRY SET (GOOD RETURN)       |
| 4813 | 057322 | 010001 |        |       | MOV R0,R1                                                           | SAVE CONTENTS OF TSSR               |
| 4814 | 057324 |        |        |       | NEXT,ERRNO                                                          |                                     |
| 4815 | 057324 |        |        | 1521: | ERRDF ERRNO,T196SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 057324 | 104455 |        |       |                                                                     | TRAP C1ERDF                         |
|      | 057326 | 001517 |        |       |                                                                     | .WORD 847                           |
|      | 057330 | 060520 |        |       |                                                                     | .WORD T196SSR                       |
|      | 057332 | 012046 |        |       |                                                                     | .WORD PKTSSR                        |
| 4816 | 057334 | 005237 | 002222 |       | INC FATFLG                                                          | SET FATAL ERROR FLAG                |
| 4817 | 057340 |        |        | 1601: | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 057340 | 104406 |        |       |                                                                     | TRAP C1CLP1                         |
| 4818 |        |        |        | 1     | Do a WRITE NPR to set loopback and tape direction IN                |                                     |
| 4819 | 057342 | 005000 |        |       | CLR R0                                                              | CLR NP.OUT TO SET TAPE DIRECTION IN |
| 4820 | 057344 | 052700 | 000040 |       | BIS #NP.LOOP,R0                                                     | SET LOOPBACK                        |
| 4821 | 057350 | 004737 | 062042 |       | JSR PC,T19SNPR                                                      | SETUP T19PK2 FOR WRITE NPR          |
| 4822 | 057354 | 012704 | 062500 |       | MOV #T19PK2,R4                                                      | GET WRITE SUBSYSTEM COMMAND PACKET  |
| 4823 | 057360 | 010465 | 000000 |       | MOV R4,TSD8(R5)                                                     | SET THE PACKET ADDRESS TO EXECUTE   |
| 4824 | 057364 | 004737 | 016336 |       | JSR PC,CHKTSSR                                                      | WAIT FOR SSR TO SET                 |
| 4825 | 057370 |        |        |       | FORCERROR 1821                                                      | FORCE ERROR IF FORCER=1             |
| 4826 | 057404 | 103407 |        |       | BCS 1901                                                            | BR IF CARRY SET (GOOD RETURN)       |
| 4827 | 057406 | 010001 |        |       | MOV R0,R1                                                           | SAVE CONTENTS OF TSSR               |
| 4828 | 057410 |        |        |       | NEXT,ERRNO                                                          |                                     |
| 4829 | 057410 |        |        | 1821: | ERRDF ERRNO,T194SSR,PKTSSR                                          | DEVICE FATAL SSR FAILED TO SET      |
|      | 057410 | 104455 |        |       |                                                                     | TRAP C1ERDF                         |
|      | 057412 | 001520 |        |       |                                                                     | .WORD 848                           |
|      | 057414 | 060411 |        |       |                                                                     | .WORD T194SSR                       |
|      | 057416 | 012046 |        |       |                                                                     | .WORD PKTSSR                        |
| 4830 | 057420 | 005237 | 002222 |       | INC FATFLG                                                          | SET FATAL ERROR FLAG                |
| 4831 | 057424 |        |        | 1901: | CKLOOP                                                              | LOOP ON ERROR, IF FLAG SET          |
|      | 057424 | 104406 |        |       |                                                                     | TRAP C1CLP1                         |
| 4832 |        |        |        | 1     | Do a WRITE FORMAT to set IRESV4***IRGIR = 0                         |                                     |

C16

TSV5 HARDWARE TESTS MACRO M1113 06 FEB 84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 197

|      |        |        |        |            |                                                          |                                        |
|------|--------|--------|--------|------------|----------------------------------------------------------|----------------------------------------|
| 4833 | 057426 | 005000 |        | CLR        | R0                                                       | ISET IRESV4-->IRSTR=0                  |
| 4834 | 057430 | 004737 | 062162 | JSR        | PC,T19WFMT                                               | ISUP T9PK2 FOR WRITE FORMAT            |
| 4835 | 057434 | 012704 | 062500 | MOV        | #T19PK2,R4                                               | IGET WRITE SUBSYSTEM COMMAND PACKET    |
| 4836 | 057440 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                              | ISUP THE PACKET ADDRESS TO EXECUTE     |
| 4837 | 057444 | 004737 | 016336 | JSR        | PC,CHKTSR                                                | IWAIT FOR SSR TO SET                   |
| 4838 | 057450 |        |        | FORCERROR  | 2021                                                     | ISUPFORCE ERROR IF FORCER=1            |
| 4839 | 057464 | 103407 |        | BCS        | 2101                                                     | IBR IF CARRY SET (GOOD RETURN)         |
| 4840 | 057466 | 010001 |        | MOV        | R0,R1                                                    | ISAVE CONTENTS OF TSSR                 |
| 4841 | 057470 |        |        | NEXT,ERRNO |                                                          |                                        |
| 4842 | 057470 |        |        | ERRDF      | ERRNO,T198SSR,PKTSSR                                     | IDEVICE FATAL SSR FAILED TO SET        |
|      | 057470 | 104455 |        |            |                                                          | TRAP C1ERDF                            |
|      | 057472 | 001521 |        |            |                                                          | .WORD 849                              |
|      | 057474 | 060632 |        |            |                                                          | .WORD T198SSR                          |
|      | 057476 | 012046 |        |            |                                                          | .WORD PKTSSR                           |
| 4843 | 057500 | 005237 | 002222 | INC        | FATFLG                                                   | ISET FATAL ERROR FLAG                  |
| 4844 | 057504 |        |        | CKLOOP     |                                                          | ILOOP ON ERROR, IF FLAG SET            |
|      | 057504 | 104406 |        |            |                                                          | TRAP C1CLP1                            |
| 4845 |        |        |        |            | Do a WRITE FORMAT to set IRESV4-->IRSTR = 1              |                                        |
| 4846 | 057506 | 012700 | 000001 | MOV        | #WF,I4RES,R0                                             | IRESV4-->IRSTR=1                       |
| 4847 | 057512 | 004737 | 062162 | JSR        | PC,T19WFMT                                               | ISUP T9PK2 FOR WRITE FORMAT            |
| 4848 | 057516 | 012704 | 062500 | MOV        | #T19PK2,R4                                               | IGET WRITE SUBSYSTEM COMMAND PACKET    |
| 4849 | 057522 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                              | ISUP THE PACKET ADDRESS TO EXECUTE     |
| 4850 | 057526 | 004737 | 016336 | JSR        | PC,CHKTSR                                                | IWAIT FOR SSR TO SET                   |
| 4851 | 057532 |        |        | FORCERROR  | 2221                                                     | ISUPFORCE ERROR IF FORCER=1            |
| 4852 | 057546 | 103407 |        | BCS        | 2301                                                     | IBR IF CARRY SET (GOOD RETURN)         |
| 4853 | 057550 | 010001 |        | MOV        | R0,R1                                                    | ISAVE CONTENTS OF TSSR                 |
| 4854 | 057552 |        |        | NEXT,ERRNO |                                                          |                                        |
| 4855 | 057552 |        |        | ERRDF      | ERRNO,T198SSR,PKTSSR                                     | IDEVICE FATAL SSR FAILED TO SET        |
|      | 057552 | 104455 |        |            |                                                          | TRAP C1ERDF                            |
|      | 057554 | 001522 |        |            |                                                          | .WORD 850                              |
|      | 057556 | 060632 |        |            |                                                          | .WORD T198SSR                          |
|      | 057560 | 012046 |        |            |                                                          | .WORD PKTSSR                           |
| 4856 | 057562 | 005237 | 002222 | INC        | FATFLG                                                   | ISET FATAL ERROR FLAG                  |
| 4857 | 057566 |        |        | CKLOOP     |                                                          | ILOOP ON ERROR, IF FLAG SET            |
|      | 057566 | 104406 |        |            |                                                          | TRAP C1CLP1                            |
| 4858 |        |        |        |            | Do a READ FIFO with tape direction IN to read data       |                                        |
| 4859 | 057570 | 012700 | 000001 | MOV        | #1,R0                                                    | ISET READ BYTE COUNT                   |
| 4860 | 057574 | 004737 | 062066 | JSR        | PC,T19RFIT                                               | ISUP T19PK2 FOR READ FIFO              |
| 4861 | 057600 | 012704 | 062500 | MOV        | #T19PK2,R4                                               | IGET WRITE SUBSYSTEM COMMAND PACKET    |
| 4862 | 057604 | 010465 | 000000 | MOV        | R4,TSDB(R5)                                              | ISUP THE PACKET ADDRESS TO EXECUTE     |
| 4863 | 057610 | 004737 | 016336 | JSR        | PC,CHKTSR                                                | IWAIT FOR SSR TO SET                   |
| 4864 | 057614 |        |        | FORCERROR  | 2821                                                     | ISUPFORCE ERROR IF FORCER=1            |
| 4865 | 057630 | 103407 |        | BCS        | 2901                                                     | IBR IF CARRY SET (GOOD RETURN)         |
| 4866 | 057632 | 010001 |        | MOV        | R0,R1                                                    | ISAVE CONTENTS OF TSSR                 |
| 4867 | 057634 |        |        | NEXT,ERRNO |                                                          |                                        |
| 4868 | 057634 |        |        | ERRDF      | ERRNO,T196SSR,PKTSSR                                     | IDEVICE FATAL SSR FAILED TO SET        |
|      | 057634 | 104455 |        |            |                                                          | TRAP C1ERDF                            |
|      | 057636 | 001523 |        |            |                                                          | .WORD 851                              |
|      | 057640 | 060520 |        |            |                                                          | .WORD T196SSR                          |
|      | 057642 | 012046 |        |            |                                                          | .WORD PKTSSR                           |
| 4869 | 057644 | 005237 | 002222 | INC        | FATFLG                                                   | ISET FATAL ERROR FLAG                  |
| 4870 | 057650 |        |        | CKLOOP     |                                                          | ILOOP ON ERROR, IF FLAG SET            |
|      | 057650 | 104406 |        |            |                                                          | TRAP C1CLP1                            |
| 4871 |        |        |        |            | If Data read from FIFO NOT to Data sent Then Print Error |                                        |
| 4872 | 057652 | 004737 | 062240 | JSR        | PC,T19SETEXP                                             | ISET WORDS 0-7 EXPD-RECV (NOT TESTING) |
| 4873 | 057656 | 012701 | 060102 | MOV        | #T19EXSTA,R1                                             | IGET EXPECTED READ STATUS              |
| 4874 | 057662 | 012702 | 062372 | MOV        | #T19BFSTA,R2                                             | IGET RECV READ STATUS                  |

016

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 198

|      |        |        |               |            |                             |   |                                       |               |
|------|--------|--------|---------------|------------|-----------------------------|---|---------------------------------------|---------------|
| 4875 | 057666 | 013711 | 002312        | MOV        | DATA,(R1)                   | ; | SET EXPD WORD #8 = DATA               |               |
| 4876 | 057672 | 016261 | 000002 000002 | MOV        | 2(R2),2(R1)                 | ; | SET EXPD WORD #9 = RECV (NOT TESTING) |               |
| 4877 | 057700 | 005000 |               | CLR        | R0                          | ; | HIGH RECV ADDRESS FOR CKMSG2          |               |
| 4878 | 057702 | 012701 | 062352        | MOV        | #T19BFR,R1                  | ; | LOW RECV ADDRESS FOR CKMSG2           |               |
| 4879 | 057706 | 012702 | 060062        | MOV        | #T19EXP,R2                  | ; | EXPD ADDRESS                          |               |
| 4880 | 057712 | 012703 | 000022        | MOV        | #18.,R3                     | ; | NUMBER OF BYTES TO COMPARE            |               |
| 4881 | 057716 | 004737 | 011500        | JSR        | PC,CKMSG2                   | ; | EXPD EQUAL RECV?                      |               |
| 4882 | 057722 |        |               | FORCERROR  | 302\$,NOTSSR                | ; | 880                                   |               |
| 4883 | 057732 | 103404 |               | BCS        | 310\$                       | ; | BR IF YES                             |               |
| 4884 | 057734 |        |               | NEXT,ERRNO |                             |   |                                       |               |
| 4885 | 057734 |        |               | 302\$:     | ERRHRD ERRNO,T19RSTR,MSGSUB | ; | REPORT ERROR                          |               |
|      | 057734 | 104456 |               |            |                             |   |                                       | TRAP C\$ERHRD |
|      | 057736 | 001524 |               |            |                             |   |                                       | .WORD 852     |
|      | 057740 | 061650 |               |            |                             |   |                                       | .WORD T19RSTR |
|      | 057742 | 013742 |               |            |                             |   |                                       | .WORD MSGSUB  |
| 4886 | 057744 |        |               | 310\$:     | CKLOOP                      | ; | LOOP ON ERROR, IF FLAG SET            |               |
|      | 057744 | 104406 |               |            |                             |   |                                       | TRAP C\$CLP1  |
| 4887 |        |        |               |            |                             |   |                                       |               |
| 4888 |        |        |               |            |                             |   |                                       |               |
| 4889 | 057746 |        |               | FORCEEXIT  | 355\$                       | ; | 880                                   |               |
| 4890 | 057756 | 013703 | 002316        | MOV        | TSTPTR,R3                   | ; | RESTORE CURRENT TSTBLK POINTER        |               |
| 4891 | 057762 | 020327 | 003062        | CMP        | R3,#TBLEND                  | ; | END OF TSTBLK?                        |               |
| 4892 | 057766 | 103002 |               | BHIS       | 355\$                       | ; | BR IF YES                             |               |
| 4893 | 057770 | 000137 | 057004        | JMP        | 100\$                       | ; | DO ANOTHER TSTBLK PATTERN             |               |
| 4894 | 057774 |        |               | 355\$:     |                             |   |                                       |               |
| 4895 |        |        |               |            |                             |   |                                       |               |
| 4896 | 057774 |        |               | ENDSUB     |                             | ; | ////////// END SUBTEST //////////     |               |
|      | 057774 |        |               |            |                             |   |                                       | L10072:       |
|      | 057774 | 104403 |               |            |                             |   |                                       | TRAP C\$ESUB  |
| 4897 |        |        |               |            |                             |   |                                       |               |
| 4898 | 057776 | 005737 | 002222        | TST        | FATFLG                      | ; | ANY FATAL ERRORS?                     |               |
| 4899 | 060002 | 001402 |               | BEQ        | 360\$                       | ; | BRANCH IF NOT                         |               |
| 4900 | 060004 | 004737 | 017202        | JSR        | PC,CKDROP                   | ; | TRY TO DROP THE UNIT                  |               |
| 4901 | 060010 |        |               | 360\$:     |                             |   |                                       |               |
| 4902 |        |        |               |            |                             |   |                                       |               |
| 4903 | 060010 |        |               | EXIT       | TST                         | ; | ////////// EXIT TEST //////////       |               |
|      | 060010 | 104432 |               |            |                             |   |                                       | TRAP C\$EXIT  |
|      | 060012 | 002602 |               |            |                             |   |                                       | .WORD L10066- |
| 4904 |        |        |               |            |                             |   |                                       |               |
| 4905 |        |        |               |            |                             |   |                                       |               |
| 4906 |        |        |               |            |                             |   |                                       |               |
| 4907 |        |        |               |            |                             |   |                                       |               |
| 4908 |        |        |               |            |                             |   |                                       |               |
| 4909 |        |        |               |            |                             |   |                                       |               |
| 4910 | 060014 | 000000 |               | T19PREV:   | .WORD 0                     | ; | DRIVE SIGNAL 1-0 TRANSITION FLAG      |               |
| 4911 |        |        |               |            |                             |   |                                       |               |
| 4912 |        |        |               |            |                             |   |                                       |               |
| 4913 |        |        |               |            |                             |   |                                       |               |
| 4914 |        |        |               |            |                             |   |                                       |               |
| 4915 |        |        |               |            |                             |   |                                       |               |
| 4916 |        |        |               |            |                             |   |                                       |               |
| 4917 |        |        |               |            |                             |   |                                       |               |
| 4918 |        |        |               |            |                             |   |                                       |               |
| 4919 | 060016 |        |               | T19BCTL:   |                             | ; | WRITE CONTROL DRIVE SIGNALS           |               |
| 4920 | 060016 | 000001 |               |            | WC,IG0                      | ; | IG0...IFPT DATA<0>                    |               |
| 4921 | 060020 | 000002 |               |            | WC,IFEN                     | ; | IFEN...IFBY DATA<1>                   |               |
| 4922 | 060022 | 000004 |               |            | WC,IRWU                     | ; | IRWU...IRWD DATA<2>                   |               |

TSV5 HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 199

4923 060024 000010  
4924 060026 002000  
4925 060030 000040  
4926 060032 000100  
4927 060034 000200  
4928 060036 004000  
4929 060040 010000  
4930 060042 020000  
4931 060044 040000  
4932 060046 100000  
4933 060050 000000  
4934 060052 000000  
4935 060054 000000

WC,IREW  
WF,IERASE\*256.  
WC,IITAD  
WC,IOTAD  
WC,IFAD  
WF,IEDIT\*256.  
WF,IWFM\*256.  
WF,IREV\*256.  
WF,IWRT\*256.  
WF,IHISP\*256.  
.WORD 0  
.WORD 0  
.WORD 0

;IREW==>IDBY DATA<3>  
;IFAD==>ILDY DATA<4>  
;ITAD1==>IONL DATA<5>  
;ITAD0==>IRDY DATA<6>  
;IERASE==>ISPEED DATA<7>  
;IEDIT==>IHER DATA<8>  
;IWFM==>IFMK DATA<9>  
;IREV==>ICER DATA<10>  
;IWRT==>IIDENT DATA<11>  
;IHISP==>IEOT DATA<12>  
;IRESV2 (UNUSED)DATA<13>  
;IRESV1 (UNUSED)DATA<14>  
;PARERR (UNTESTED)DATA<15>

4936  
4937 060056  
4938  
4939 060056 377  
4940 060057 037  
4941 060060 360  
4942 060061 000  
4943

T19MSK:

.BYTE \*C<000>  
.BYTE \*C<340>  
.BYTE \*C<017>  
.BYTE 0

;MASK OF UNTESTED BITS IN READ STATUS BYTES  
;UNTESTED BITS ARE SET TO 1  
;BYTE 0 MASK  
;BYTE 1 MASK (PARERR,IRESV2,IRESV1)  
;BYTE 2 (TIMER A,TIMER B,UNDEFINED<1:0>)  
;MAKE IT EVEN

4944 060062  
4945 060062 000000  
4946 060064 000000  
4947 060066 000000  
4948 060070 000000  
4949 060072 000000  
4950 060074 000000  
4951 060076 000000  
4952 060100 000000

T19EXP:

.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0  
.WORD 0

;BEGIN EXPECTED DATA BUFFER  
;MESSAGE TYPE  
;DATA FIELD LENGTH  
;RBCR  
;XST0  
;XST1  
;XST2  
;XST3  
;XST4 (ALWAYS PRESENT FOR WRITE SUB.)  
;EXPECTED READ STATUS AND WRITE FIFO DATA  
;END EXPECTED DATA BUFFER

4953 060102  
4954 060202  
4955  
4956  
4957  
4958

T19EXSTA: .BLKB 64.  
T19EXEND:

;LOCAL TEXT MESSAGES FOR TEST  
;-

4959 060202 124  
4960 060243 127  
4961 060300 127  
4962 060344 127  
4963 060411 127  
4964 060454 127  
4965 060520 127  
4966 060563 127  
4967 060632 127  
4968 060700 106  
4969 060762 127  
4970 061056 124  
4971 061144 122  
4972 061220 106  
4973 061303 124  
4974 061371 103  
4975 061460 127  
4976 061543 114  
4977 061650 114  
4978  
4979

162  
122  
122  
122  
122  
122  
122  
122  
122  
111  
145  
141  
145  
111  
141  
157  
145  
157  
157

141  
111  
111  
111  
111  
111  
111  
111  
111  
106  
141  
160  
141  
106  
160  
156  
141  
157  
157

TST19ID: .ASCIZ 'Transport Bus Interface Loopback'  
T19SSR: .ASCIZ 'WRITE CHARACTERISTICS Failed'  
T192SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Misc) Failed'  
T193SSR: .ASCIZ 'WRITE SUBSYSTEM (Read Status) Failed'  
T194SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Npr) Failed'  
T195SSR: .ASCIZ 'WRITE SUBSYSTEM (Write FIFO) Failed'  
T196SSR: .ASCIZ 'WRITE SUBSYSTEM (Read FIFO) Failed'  
T197SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Control) Failed'  
T198SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Format) Failed'  
T191CMP: .ASCIZ 'FIFO Status in WORD 09 Incorrect after Initialize'  
T192CMP: .ASCIZ 'Read FIFO Data not equal to Write FIFO, Data is in WORD 08'  
T193CMP: .ASCIZ 'Tape Status 2 (in WORD 08) Incorrect after RESET TAPE'  
T195CMP: .ASCIZ 'Read FIFO Data not equal to Write FIFO Data'  
T196CMP: .ASCIZ 'FIFO Status (in WORD 09) Incorrect after READ FIFO'  
T197CMP: .ASCIZ 'Tape Status 2 (in WORD 08) Incorrect after RESET TAPE'  
T198CMP: .ASCIZ 'Control Signal Loopback Data Error, Data is in WORD 08'  
T199CMP: .ASCIZ 'Read/Write Loopback Data Error, Data is in WORD 08'  
T19WSTR: .ASCIZ 'Loopback Data Error when strobed by Write Strobe, Data is in WORD 08'  
T19RSTR: .ASCIZ 'Loopback Data Error when strobed by Read Strobe, Data is in WORD 08'

.EVEN

F16

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 200

```

4980
4981
4982
4983
4984 061754
4985 061754
4986 061760 012701 062352
4987 061764 012702 000120
4988 061770 105021
4989 061772 005302
4990 061774 003375
4991 061776 000207
4992
4993
4994
4995
4996 062000
4997 062000 004737 061754
4998 062004 012700 062510
4999 062010 112720 000005
5000 062014 105010
5001 062016 000207
5002
5003
5004
5005
5006 062020
5007 062020 004737 061754
5008 062024 012700 062510
5009 062030 112720 000010
5010 062034 112710 000030
5011 062040 000207
5012
5013
5014
5015
5016
5017
5018
5019
5020
5021 062042
5022 062042 004737 061754
5023 062046 012701 062510
5024 062052 112721 000011
5025 062056 052700 000020
5026 062062 110011
5027 062064 000207
5028
5029
5030
5031
5032
5033
5034
5035 062066
5036 062066 004737 061754

; CLEAR MESSAGE BUFFER
;
T19CLRBUF:
    SAVREG
    MOV     @T19BFR,R1
    MOV     @T19BEND-T19BFR,R2
10$:   CLR    (R1)+
    DEC     R2
    BGT     10$
    RTS     PC

; SAVE R1 R5 UNTIL NEXT RETURN
; GET MESSAGE BUFFER ADDRESS
; SIZE OF MESSAGE BUFFER IN BYTES
; CLEAR A BYTE
; DONE?
; BR IF NO
; RETURN

; SETUP T19PK2 PACKET FOR READ STATUS
;
T19SRD:
    JSR     PC,T19CLRBUF
    MOV     @T19DT2,R0
    MOV     @PW,RDSTATUS,(R0)+
    CLR     (R0)
    RTS     PC

; CLEAR MESSAGE BUFFER
; WRITE SUBSYSTEM DATA BUFFER
; STORE READ STATUS COMMAND IN BSEL0
; CLEAR BSEL1
; RETURN

; SETUP T19PK2 PACKET FOR WRITE MISC Reset Tape Status F-FLOPS
;
T19RSFIF:
    JSR     PC,T19CLRBUF
    MOV     @T19DT2,R0
    MOV     @PW,WMISC,(R0)+
    MOV     @MS,RSFIF!MS,RSTAP,(R0)+
    RTS     PC

; CLEAR MESSAGE BUFFER
; WRITE SUBSYSTEM DATA BUFFER
; STORE WRITE MISCELLANEOUS IN BSEL0
; STORE BSEL1 CLEAR FIFO CODES
; RETURN

; SETUP T19PK2 PACKET FOR WRITE NPR
;
; INPUT:
;     R0 CONTAINS BSEL1 NPR DATA
;
;     SELS NP,WRP SINCE IF C IT WRITES WRONG PARITY.
;
T19SNPR:
    JSR     PC,T19CLRBUF
    MOV     @T19DT2,R1
    MOV     @PW,WNPR,(R1)+
    BIS     @NP,WRP,R0
    MOV     R0,(R1)
    RTS     PC

; CLEAR MESSAGE BUFFER
; WRITE SUBSYSTEM DATA BUFFER
; STORE WRITE NPR IN BSEL0
; DON'T WRITE WRONG PARITY
; STORE NPR DATA IN BSEL1
; RETURN

; SETUP T19PK2 PACKET FOR READ FIFO
;
; INPUT:
;     R0 CONTAINS SEL2 BYTE COUNT
;
T19RFIF:
    JSR     PC,T19CLRBUF
; CLEAR MESSAGE BUFFER

```

ISV5 - HARDWARE TESTS MACRO M1113 06-FEB 84 17:14  
 TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 201

|      |        |        |        |          |                                                                     |                                          |
|------|--------|--------|--------|----------|---------------------------------------------------------------------|------------------------------------------|
| 5037 | 062072 | 012701 | 062510 | MOV      | 0T19DT2,R1                                                          | ;WRITE SUBSYSTEM DATA BUFFER             |
| 5038 | 062076 | 112721 | 000003 | MOV      | 0PW.RFIFO,(R1)+                                                     | ;STORE READ FIFO IN BSEL0                |
| 5039 | 062102 | 110021 |        | MOV      | RO,(R1)+                                                            | ;STORE BYTE COUNT IN BSEL1               |
| 5040 | 062104 | 000207 |        | RTS      | PC                                                                  | ;RETURN                                  |
| 5041 |        |        |        | ;        |                                                                     |                                          |
| 5042 |        |        |        | ;        | SETUP T19PK2 PACKET FOR WRITE FIFO                                  |                                          |
| 5043 |        |        |        | ;        |                                                                     |                                          |
| 5044 |        |        |        | ;        | 1 PUT:                                                              |                                          |
| 5045 |        |        |        | ;        | RO CONTAINS BYTE COUNT                                              |                                          |
| 5046 |        |        |        | ;        | R1 CONTAINS DATA PATTERN BLOCK ADDRESS                              |                                          |
| 5047 |        |        |        | ;        |                                                                     |                                          |
| 5048 | 062106 |        |        | T19WFIF: |                                                                     |                                          |
| 5049 | 062106 |        |        | SAVREG   |                                                                     | ;SAVE R1-R5 UNTIL NEXT RETURN            |
| 5050 | 062112 | 004737 | 061754 | JSR      | PC,T19CLRBUF                                                        | ;CLEAR MESSAGE BUFFER                    |
| 5051 | 062116 | 012702 | 062510 | MOV      | 0T19DT2,R2                                                          | ;WRITE SUBSYSTEM DATA BUFFER             |
| 5052 | 062122 | 112722 | 000004 | MOV      | 0PW.WFIFO,(R2)+                                                     | ;STORE WRITE FIFO IN BSEL0               |
| 5053 | 062126 | 110022 |        | MOV      | RO,(R2)+                                                            | ;STORE BYTE COUNT IN BSEL1               |
| 5054 | 062130 | 005022 |        | CLR      | (R2)+                                                               | ;CLEAR SEL2 (UNUSED)                     |
| 5055 | 062132 | 112122 |        | 10\$:    | MOV (R1)+,(R2)+                                                     | ;STORE DATA PATTERN BYTE                 |
| 5056 | 062134 | 005300 |        | DEC      | RO                                                                  | ;DONE ALL BYTES?                         |
| 5057 | 062136 | 003375 |        | BGT      | 10\$                                                                | ;BR IF NO                                |
| 5058 | 062140 | 000207 |        | RTS      | PC                                                                  | ;RETURN                                  |
| 5059 |        |        |        | ;        |                                                                     |                                          |
| 5060 |        |        |        | ;        | SETUP T19PK2 FOR WRITE CONTROL                                      |                                          |
| 5061 |        |        |        | ;        |                                                                     |                                          |
| 5062 |        |        |        | ;        | INPUT:                                                              |                                          |
| 5063 |        |        |        | ;        | RO CONTAINS DRIVING DATA PATTERN                                    |                                          |
| 5064 |        |        |        | ;        |                                                                     |                                          |
| 5065 | 062142 |        |        | T19WCTL: |                                                                     |                                          |
| 5066 | 062142 | 004737 | 061754 | JSR      | PC,T19CLRBUF                                                        | ;CLEAR MESSAGE BUFFER                    |
| 5067 | 062146 | 012701 | 062510 | MOV      | 0T19DT2,R1                                                          | ;WRITE SUBSYSTEM DATA BUFFER             |
| 5068 | 062152 | 112721 | 000006 | MOV      | 0PW.WCTL,(R1)+                                                      | ;STORE WRITE CONTROL IN BSEL0            |
| 5069 | 062156 | 110021 |        | MOV      | RO,(R1)+                                                            | ;STORE DATA WORD IN BSEL1                |
| 5070 | 062160 | 000207 |        | RTS      | PC                                                                  | ;RETURN                                  |
| 5071 |        |        |        | ;        |                                                                     |                                          |
| 5072 |        |        |        | ;        | SETUP T19PK2 FOR WRITE FORMAT TRANSPORT REGISTER                    |                                          |
| 5073 |        |        |        | ;        |                                                                     |                                          |
| 5074 |        |        |        | ;        | INPUT:                                                              |                                          |
| 5075 |        |        |        | ;        | RO CONTAINS DRIVING DATA PATTERN                                    |                                          |
| 5076 |        |        |        | ;        |                                                                     |                                          |
| 5077 | 062162 |        |        | T19WFMT: |                                                                     |                                          |
| 5078 | 062162 | 004737 | 061754 | JSR      | PC,T19CLRBUF                                                        | ;CLEAR MESSAGE BUFFER                    |
| 5079 | 062166 | 012701 | 062510 | MOV      | 0T19DT2,R1                                                          | ;WRITE SUBSYSTEM DATA BUFFER             |
| 5080 | 062172 | 112721 | 000007 | MOV      | 0PW.WFMT,(R1)+                                                      | ;STORE WRITE FORMAT IN BSEL0             |
| 5081 | 062176 | 110021 |        | MOV      | RO,(R1)+                                                            | ;STORE DATA WORD IN BSEL1                |
| 5082 | 062200 | 000207 |        | RTS      | PC                                                                  | ;RETURN                                  |
| 5083 |        |        |        | ;        |                                                                     |                                          |
| 5084 |        |        |        | ;        | SETUP T19PK2 PACKET FOR WRITE MISC. INVERT EXTENDED FEATURES SWITCH |                                          |
| 5085 |        |        |        | ;        |                                                                     |                                          |
| 5086 | 062202 |        |        | T19SEXT: |                                                                     |                                          |
| 5087 | 062202 | 012700 | 062510 | MOV      | 0T19DT2,RO                                                          | ;WRITE SUBSYSTEM DATA BUFFER             |
| 5088 | 062206 | 112720 | 000010 | MOV      | 0PW.WMISC,(RO)+                                                     | ;STORE WRITE MISCELLANEOUS IN BSEL0      |
| 5089 | 062212 | 112710 | 000200 | MOV      | 0MS.EXT,(RO)                                                        | ;STORE INVERT EXTENDED FEATURES IN BSEL1 |
| 5090 | 062216 | 000207 |        | RTS      | PC                                                                  | ;RETURN                                  |
| 5091 |        |        |        | ;        |                                                                     |                                          |
| 5092 |        |        |        | ;        | CLEAR EXPECTED DATA MESSAGE BUFFER                                  |                                          |
| 5093 |        |        |        | ;        |                                                                     |                                          |

H16

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 202

```

5094 062220
5095 062220 012701 060062
5096 062224 012700 000120
5097 062230 105021
5098 062232 005300
5099 062234 003375
5100 062236 000207
5101
5102
5103
5104
5105 062240
5106 062240 012702 060062
5107 062244 012703 062352
5108 062250 012700 000010
5109 062254 012322
5110 062256 005300
5111 062260 003375
5112 062262 000207
5113
5114
5115
5116
5117
5118
5119
5120
5121
5122
5123
5124
5125
5126
5127
5128
5129
5130 062264
5131 062264
5132 062270 012701 060016
5133 062274 005002
5134 062276 012703 000020
5135 062302 006000
5136 062304 103001
5137 062306 051102
5138 062310 005721
5139 062312 005303
5140 062314 003372
5141 062316 010200
5142 062320 000207
5143
5144
5145
5147 062330
5149
5150
5151
5152 062330

T19CLEXP:
MOV    #T19EXP,R1      ;GET EXPD ADDRESS
MOV    #T19XEND-T19EXP,R0 ;GET EXPD SIZE
10$:   CLRB    (R1)+     ;CLEAR A BYTE
        DEC    R0        ;DONE?
        BGT    10$       ;BR IF NO
        RTS    PC        ;RETURN

;
; Set WORDS 0-7 of expd message BUFFER = to recv since not testing
;
T19SETEXP:
MOV    #T19EXP,R2      ;GET EXPD
MOV    #T19RFR,R3      ;GET READ STATUS RECV BUFFER
MOV    #8,,R0          ;SET WORDS 0-7 EXP=RECV
5$:    MOV    (R3)+,(R2)+ ;SET EXPD=RECV
        DEC    R0        ;DONE WORDS 0-7 WORDS?
        BGT    5$        ;BR IF NO
        RTS    PC        ;RETURN

;
; CONVERT A TEST PATTERN DATA WORD TO LOOPBACK DRIVE SIGNALS
;
; INPUTS:
;
;     R0      TEST PATTERN
;
; IMPLICIT INPUTS:
;
;     T19BFCTL - CONTAINS WRITE CONTROL / WRITE FORMAT CONVERSION BITS
;
; OUTPUTS:
;
;     R0      - LOW BYTE CONTAINS WRITE CONTROL DATA
;              - HIGH BYTE CONTAINS WRITE FORMAT DATA
;
T19CNVT:
SAVREG
MOV    #T19BFCTL,R1    ;SAVE R1-R5 UNTIL NEXT RETURN
CLR    R2              ;CONVERSION TABLE ADDRESS
MOV    #16,,R3         ;INIT RESULT OF CONVERSION
10$:   MOV    #16,,R3   ;BIT COUNT
        ROR    R0       ;IS THIS BIT EQUAL TO 1?
        BCC    20$      ;BR IF NO
        HIS    (R1),R2   ;SET CONVERTED BIT
        BCC    20$      ;POINT TO NEXT BIT IN CONVERSION TABLE
        DEC    R3        ;DONE?
        BGT    10$       ;BR IF NO
        MOV    R2,R0     ;COPY RESULT
        RTS    PC        ;RETURN

;
; WRITE CHARACTERISTICS COMMAND PACKET
;
T19PACKET:
; COMMAND PACKET FOR TEST

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 8: SUBTEST 4 LOOPBACK READ STROBE TEST

SEQ 203

```

5153 062330 100004      .WORD 100004      ;WRITE CHARACTERISTICS COMMAND, WITH ACK
5154 062332 062340      .WORD T19DATA     ;ADDRESS OF CHARACTERISTICS BLOCK
5155 062334 000000      .WORD 0
5156 062336 000012      .WORD 10.         ;MINIMUM MESSAGE PACKET SIZE
5157
5158 062340      T19DATA:      ;CHARACTERISTICS DATA BLOCK
5159 062340 062352      .WORD T19BFR       ;ADDRESS OF MESSAGE BUFFER
5160 062342 000000      .WORD 0
5161 062344 000024      .WORD 20.         ;LENGTH OF MESSAGE BUFFER
5162 062346 000000      .WORD 0           ;ESS,ENB,EAI,ERI
5163 062350 000007      .WORD 7           ;EXTENDED FEATURES UNIT NO.
5164
5165
5166      ;MESSAGE BUFFER FOR ALL TEST 8 COMMANDS
5167
5168 062352      T19BFR:      ;BEGIN MESSAGE BUFFER
5169 062352 000000      .WORD 0           ;MESSAGE TYPE
5170 062354 000000      .WORD 0           ;DATA FIELD LENGTH
5171 062356 000000      .WORD 0           ;RBPGR
5172 062360 000000      .WORD 0           ;XST0
5173 062362 000000      .WORD 0           ;XST1
5174 062364 000000      .WORD 0           ;XST2
5175 062366 000000      .WORD 0           ;XST3
5176 062370 000000      .WORD 0           ;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM
5177 062372      T19BFSTA: .BLKB 64.      ;READ STATUS AND WRITE FIFO BUFFER
5178 062472      T19BEND:      ;END OF MESSAGE BUFFER
5179
5180      ;WRITE SUBSYSTEM READ STATUS COMMAND PACKET
5181
5182      ;
5183      .* < .10 > &177770
5184 062500      T19PK2:      ;WRITE SUBSYSTEM WITH ACK
5185 062500 100006      .WORD P.WRISUB!P.ACK ;LOW ADDRESS OF DATA BLOCK
5186 062502 062510      .WORD T19DT2       ;HIGH ADDRESS OF DATA BLOCK
5187 062504 000000      .WORD 0           ;MINIMUM MESSAGE PACKET SIZE
5188 062506 000012      .WORD 10.
5189
5190      T19DT2:      ;DATA BLOCK
5191 062510      .BYTE 0           ;BSEL0
5192 062510 000      .BYTE 0           ;BSEL1
5193 062511 000      .WORD 0           ;SEL2
5194 062512 000000      .BLKB 64.         ;WRITE FIFO DATA OUTPUT BUFFER
5195 062514
5196
5197
5198 062614      ENDTST
5199
5200      L10066:      TRAP C$ETST
5201
5202      ;SBTTL TEST 9: READ/WRITE DATA PARITY TEST
5203
5204      ;TEST DESCRIPTION:
5205
5206      ; This test verifies that the Write Data Parity generator
5207      ; and the Read Data Parity checker operate properly. The
5208      ; Transport Bus signal loopback mode is enabled and a
5209      ; Set Wrong parity function is executed. Then various
5210      ; Write Subsystem Memory functions are performed to
5211      ; write data to and from the FIFO in loopback mode.
5212      ; The program then checks to insure a Read Data parity

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 9: READ/WRITE DATA PARITY TEST

SEQ 204

```

5210      error occurred.
5211      A Reset FIFO is done and the Read Data parity
5212      error bit is again tested to insure it cleared.
5213      Finally a Clear wrong parity function is done
5214      and it is verified the data word can pass in loopback
5215      mode without setting Read Data parity error.
5216
5217      TEST STEPS:
5218
5219      REPEAT FOR LOOPCNT
5220      BEGIN
5221      Write to TSSR register to soft initialize the controller
5222      Do WRITE CHARACTERISTICS to check for Extended Features Switch
5223      If Extended Features hardware Switch Clear then:
5224      Do Write Subsystem Write Miscellaneous to Set Extended Features.
5225      Do WRITE CHARACTERISTICS to select reserved unit 7 and setup BUFFER
5226      REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
5227      BEGIN
5228      (* Verify Write Wrong Parity Sets Parity Error *)
5229      Do a WRITE NPR to set loopback and tape direction OUT
5230      and SET Write Wrong Parity.
5231      Do a WRITE FORMAT to set IRESV4-->IRSTR = 1 (sets read strobe high)
5232      Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
5233      Do a READ FIFO with tape direction OUT to load tape out write latch
5234      (this is when wrong parity (IWP) is set)
5235      Do a WRITE FORMAT to set IRESV4-->IRSTR = 0 (sets read strobe low)
5236      (Read Strobe sets PAR IN H [Parity Error])
5237      Do a WRITE FORMAT to set IRESV4-->IRSTR = 1 (sets read strobe high)
5238      Do a Write Subsystem READ STATUS
5239      If Read Data parity error NOT=1 Then Print Error
5240      Do a Write Misc to RESET FIFO
5241      Do a Write Subsystem READ STATUS
5242      If Read Data parity error NOT=0 Then Print Error
5243
5244      (* Verify Data can be transferred without a Parity Error *)
5245      Do a WRITE FORMAT to set IRESV4-->IRSTR = 1 (sets read strobe high)
5246      Do a WRITE NPR to set loopback and tape direction OUT
5247      and CLEAR Write Wrong Parity.
5248      Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
5249      Do a READ FIFO with tape direction OUT to load tape out write latch
5250      Do a WRITE FORMAT to set IRESV4-->IRSTR = 0 (sets read strobe low)
5251      (Read Strobe should NOT set PAR IN H [Parity Error] here)
5252      Do a WRITE FORMAT to set IRESV4-->IRSTR = 1 (sets read strobe high)
5253      Do a Write Subsystem READ STATUS
5254      If Read Data parity error NOT=0 Then Print Error
5255
5256      END
5257
5258
5259
5260      062616      BGNTST
5261      062616
5262      062616      012700      065202      MOV      #TST2CID,R0
5263      062622      004737      016510      JSR      PC,TSTSETUP
5264      062626      012737      000012      002216      MOV      #10.,LOOPCNT
5265      062634      T20LOOP;
5266
5267
5268
5269

```

T9:;  
 ;ASCII MESSAGE TO IDENTIFY TEST  
 ;DO INITIAL TEST SETUP  
 ;PERFORM 10 ITERATIONS

K16

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 205

```

5270 062634          BGNSUB                      ;////////// BEGIN SUBTEST ///////////
      062634          T9.1:                      TRAP      C$BSUB
      062634 104402          Write to TSSR register to soft initialize the controller
5271          ;
5272 062636          5$:
5273 062636 004737 015774      JSR      PC,SOFINIT          ;WRITE TO TSSR TO SOFT INITIALIZE
5274 062642 103405          BCS      10$                  ;BR IF SOFT INIT OKAY
5275 062644 010001          MOV      R0,R1                  ;SAVE CONTENTS OF TSSR
5276 062646          ERRDF      ERRNO,SFIERR,SFIMSG          ;DEVICE FATAL DURING INIT
      062646 104455          TRAP      C$ERDF
      062650 001604          .WORD    900
      062652 003652          .WORD    SFIERR
      062654 012034          .WORD    SFIMSG
5277          ; Do WRITE CHARACTERISTICS to check for Extended Features Switch
5278 062656 005037 002222      10$:      CLR      FATFLG          ;CLEAR FATAL ERROR FLAG
5279 062662 012704 066400      MOV      @T20PACKET,R4          ;GET THE ADDRESS OF COMMAND PACKET
5280 062666 004737 010662      JSR      PC,WRTCHR          ;DO WRITE CHARACTERISTICS COMMAND
5281 062672          FORCERROR      12$          ;GOODFORCE ERROR IF FORCER=1
5282 062706 103407          BCS      15$          ;BR IF CARRY SET (GOOD RETURN)
5283 062710 010001          MOV      R0,R1                  ;SAVE CONTENTS OF TSSR
5284 062712          NEXT,ERRNO
5285 062712          12$:      ERRDF      ERRNO,T20SSR,PKTSSR          ;DEVICE FATAL SSR FAILED TO SET
      062712 104455          TRAP      C$ERDF
      062714 001605          .WORD    901
      062716 065231          .WORD    T20SSR
      062720 012046          .WORD    PKTSSR
5286 062722 005237 002222      15$:      INC      FATFLG          ;SET FATAL ERROR FLAG
5287 062726          CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      062726 104406          TRAP      C$CLP1
5288          ; If Extended Features Hardware Switch Clear then:
5289          ; Do Write Subsystem Write Miscellaneous to Set Extended Features.
5290 062730 012701 066422      MOV      @T20BFR,R1          ;MESSAGE BUFFER ADDRESS
5291 062734 032761 000200 000012      BIT      @X2,EXTF,XST2(R1)          ;EXTENDED FEATURES SWITCH SET?
5292 062742 001026          BNE      30$          ;BR IF YES
5293 062744 004737 066316      JSR      PC,T20SEXT          ;SETUP PACKET FOR WRITE MISC INVERT
5294 062750 012704 066550      MOV      @T20PK2,R4          ;GET WRITE SUBSYSTEM COMMAND PACKET
5295 062754 010465 000000      MOV      R4,T20DB(R5)          ;SET THE PACKET ADDRESS TO EXECUTE
5296 062760 004737 016336      JSR      PC,CHKTSSR          ;WAIT FOR SSR TO SET
5297 062764          FORCERROR      22$          ;GOODFORCE ERROR IF FORCER=1
5298 063000 103407          BCS      30$          ;BR IF CARRY SET (GOOD RETURN)
5299 063002 010001          MOV      R0,R1                  ;SAVE CONTENTS OF TSSR
5300 063004          NEXT,ERRNO
5301 063004          22$:      ERRDF      ERRNO,T202SSR,PKTSSR          ;DEVICE FATAL SSR FAILED TO SET
      063004 104455          TRAP      C$ERDF
      063006 001606          .WORD    902
      063010 065266          .WORD    T202SSR
      063012 012046          .WORD    PKTSSR
5302 063014 005237 002222      30$:      INC      FATFLG          ;SET FATAL ERROR FLAG
5303 063020          CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      063020 104406          TRAP      C$CLP1
5304          ; Do WRITE CHARACTERISTICS to select reserved unit 7
5305 063022 012704 066400      MOV      @T20PACKET,R4          ;GET THE ADDRESS OF COMMAND PACKET
5306 063026 004737 010662      JSR      PC,WRTCHR          ;DO WRITE CHARACTERISTICS COMMAND
5307 063032          FORCERROR      42$          ;GOODFORCE ERROR IF FORCER=1
5308 063046 103407          BCS      50$          ;BR IF CARRY SET (GOOD RETURN)
5309 063050 010001          MOV      R0,R1                  ;SAVE CONTENTS OF TSSR
5310 063052          NEXT,ERRNO

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 9: READ/WRITE DATA PARITY TEST

SEQ 206

```

5311 063052      42$:  ERRDF  ERRNO,T20SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      063052      104455                                TRAP      C$ERDF
      063054      001607                                .WORD      903
      063056      065231                                .WORD      T20SSR
      063060      012046                                .WORD      PKTSSR
5312 063062      005237  002222      50$:  INC      FATFLG      ;SET FATAL ERROR FLAG
5313 063066      104406      50$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      063066      104406                                TRAP      C$CLP1
5314
5315
5316      ; REPEAT FOR ALL TEST PATTERNS IN TSTBLK TABLE
5317 063070      012703  002752      100$:  MOV      @TSTBLK,R3      ;GET FIRST PATTERN ADDRESS
5318 063074      012337  002312      100$:  MOV      (R3)+,DATA      ;GET A TEST PATTERN
5319 063100      042737  177400      002312  BIC      @+C<377>,DATA      ;DATA IS BYTE
5320 063106      010337  002316      100$:  MOV      R3,TSTPTR      ;SETUP CURRENT TSTBLK POINTER
5321      ; Do a WRITE NPR to set loopback and tape direction OUT and
5322      ; and SET Write Wrong Parity.
5323 063112      012700  000100      100$:  MOV      @NP.OUT,R0      ;SET TAPE DIRECTION OUT
5324 063116      052700  000040      100$:  BIS      @NP.LOOP,R0      ;SET LOOPBACK
5325 063122      042700  000020      100$:  BIC      @NP.WRP,R0      ;SET WRITE WRONG PARITY (INVERTED)
5326 063126      004737  066166      100$:  JSR      PC,T20WNPR      ;SETUP T20PK2 FOR WRITE NPR
5327 063132      012704  066550      100$:  MOV      @T20PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5328 063136      010465  000000      100$:  MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
5329 063142      004737  016336      100$:  JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5330 063146      103407      102$:  FORCERROR 102$      ;FORCE ERROR IF FORCER=1
5331 063162      010001      105$:  BCS      105$      ;BR IF CARRY SET (GOOD RETURN)
5332 063164      010001      105$:  MOV      R0,R1      ;SAVE CONTENTS OF TSSR
5333 063166      103407      102$:  NEXT.ERRNO
5334 063166      104455      102$:  ERRDF  ERRNO,T204SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      063166      104455                                TRAP      C$ERDF
      063170      001610                                .WORD      904
      063172      065377                                .WORD      T204SSR
      063174      012046                                .WORD      PKTSSR
5335 063176      005237  002222      105$:  INC      FATFLG      ;SET FATAL ERROR FLAG
5336 063202      104406      105$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      063202      104406                                TRAP      C$CLP1
5337      ; Do a WRITE FORMAT to set IRESV4=>IRSTR = 1 (sets read strobe high)
5338 063204      012700  000001      112$:  MOV      @WF.I4RES,R0      ;IRESV4=>IRSTR = 1
5339 063210      004737  066262      112$:  JSR      PC,T20WFMT      ;SETUP T20PK2 FOR WRITE FORMAT
5340 063214      012704  066550      112$:  MOV      @T20PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5341 063220      010465  000000      112$:  MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
5342 063224      004737  016336      112$:  JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5343 063230      103407      112$:  FORCERROR 112$      ;FORCE ERROR IF FORCER=1
5344 063244      010001      120$:  BCS      120$      ;BR IF CARRY SET (GOOD RETURN)
5345 063246      010001      120$:  MOV      R0,R1      ;SAVE CONTENTS OF TSSR
5346 063250      103407      112$:  NEXT.ERRNO
5347 063250      104455      112$:  ERRDF  ERRNO,T208SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      063250      104455                                TRAP      C$ERDF
      063252      001611                                .WORD      905
      063254      065551                                .WORD      T208SSR
      063256      012046                                .WORD      PKTSSR
5348 063260      005237  002222      120$:  INC      FATFLG      ;SET FATAL ERROR FLAG
5349 063264      104406      120$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      063264      104406                                TRAP      C$CLP1
5350      ; Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
5351 063266      012700  000001      120$:  MOV      @1,R0      ;WRITE 1 BYTE
5352 063272      012701  002312      120$:  MOV      @DATA,R1      ;FIFO WRITE DATA ADDRESS

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
 TEST 9: READ/WRITE DATA PARITY TEST

SEQ 207

|      |        |        |        |               |                                                                     |                                     |
|------|--------|--------|--------|---------------|---------------------------------------------------------------------|-------------------------------------|
| 5353 | 063276 | 004737 | 066226 | JSR           | PC,T20WFIF                                                          | ;SETUP T20PK2 FOR WRITE FIFO        |
| 5354 | 063302 | 012704 | 066550 | MOV           | *T20PK2,R4                                                          | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 5355 | 063306 | 010465 | 000000 | MOV           | R4,TSD8(R5)                                                         | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 5356 | 063312 | 004737 | 016336 | JSR           | PC,CHKTSSR                                                          | ;WAIT FOR SSR TO SET                |
| 5357 | 063316 |        |        | FORCERROR     | 152\$                                                               | ;###FORCE ERROR IF FORCER=1         |
| 5358 | 063332 | 103407 |        | BCS           | 160\$                                                               | ;BR IF CARRY SET (GOOD RETURN)      |
| 5359 | 063334 | 010001 |        | MOV           | R0,R1                                                               | ;SAVE CONTENTS OF TSSR              |
| 5360 | 063336 |        |        | NEXT,ERRNO    |                                                                     |                                     |
| 5361 | 063336 |        |        | 152\$: ERRDF  | ERRNO,T205SSR,PKTSSR                                                | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 063336 | 104455 |        |               |                                                                     | TRAP C\$ERDF                        |
|      | 063340 | 001612 |        |               |                                                                     | .WORD 906                           |
|      | 063342 | 065442 |        |               |                                                                     | .WORD T205SSR                       |
|      | 063344 | 012046 |        |               |                                                                     | .WORD PKTSSR                        |
| 5362 | 063346 | 005237 | 002222 | INC           | FATFLG                                                              | ;SET FATAL ERROR FLAG               |
| 5363 | 063352 |        |        | 160\$: CKLOOP |                                                                     | ;LOOP ON ERROR, IF FLAG SET         |
|      | 063352 | 104406 |        |               |                                                                     | TRAP C\$CLP1                        |
| 5364 |        |        |        | :             | Do a READ FIFO with tape direction OUT to load tape out write latch |                                     |
| 5365 |        |        |        | :             | (this is when wrong parity (IWP) is set)                            |                                     |
| 5366 | 063354 | 012700 | 000001 | MOV           | *1,R0                                                               | ;SET READ BYTE COUNT                |
| 5367 | 063360 | 004737 | 066206 | JSR           | PC,T20RFIF                                                          | ;SETUP T20PK2 FOR READ FIFO         |
| 5368 | 063364 | 012704 | 066550 | MOV           | *T20PK2,R4                                                          | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 5369 | 063370 | 010465 | 000000 | MOV           | R4,TSD8(R5)                                                         | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 5370 | 063374 | 004737 | 016336 | JSR           | PC,CHKTSSR                                                          | ;WAIT FOR SSR TO SET                |
| 5371 | 063400 |        |        | FORCERROR     | 172\$                                                               | ;###FORCE ERROR IF FORCER=1         |
| 5372 | 063414 | 103407 |        | BCS           | 180\$                                                               | ;BR IF CARRY SET (GOOD RETURN)      |
| 5373 | 063416 | 010001 |        | MOV           | R0,R1                                                               | ;SAVE CONTENTS OF TSSR              |
| 5374 | 063420 |        |        | NEXT,ERRNO    |                                                                     |                                     |
| 5375 | 063420 |        |        | 172\$: ERRDF  | ERRNO,T206SSR,PKTSSR                                                | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 063420 | 104455 |        |               |                                                                     | TRAP C\$ERDF                        |
|      | 063422 | 001613 |        |               |                                                                     | .WORD 907                           |
|      | 063424 | 065506 |        |               |                                                                     | .WORD T206SSR                       |
|      | 063426 | 012046 |        |               |                                                                     | .WORD PKTSSR                        |
| 5376 | 063430 | 005237 | 002222 | INC           | FATFLG                                                              | ;SET FATAL ERROR FLAG               |
| 5377 | 063434 |        |        | 180\$: CKLOOP |                                                                     | ;LOOP ON ERROR, IF FLAG SET         |
|      | 063434 | 104406 |        |               |                                                                     | TRAP C\$CLP1                        |
| 5378 |        |        |        | :             | Do a WRITE FORMAT to set IRESV4==>IRSTR = 0 (sets read strobe low)  |                                     |
| 5379 |        |        |        | :             | (Read Strobe sets PAR IN H [Par ty Error])                          |                                     |
| 5380 | 063436 | 005000 |        | CLR           | R0                                                                  | ;IRESV4==>IRSTR = 0                 |
| 5381 | 063440 | 004737 | 066262 | JSR           | PC,T20WFMT                                                          | ;SETUP T20PK2 FOR WRITE FORMAT      |
| 5382 | 063444 | 012704 | 066550 | MOV           | *T20PK2,R4                                                          | ;GET WRITE SUBSYSTEM COMMAND PACKET |
| 5383 | 063450 | 010465 | 000000 | MOV           | R4,TSD8(R5)                                                         | ;SET THE PACKET ADDRESS TO EXECUTE  |
| 5384 | 063454 | 004737 | 016336 | JSR           | PC,CHKTSSR                                                          | ;WAIT FOR SSR TO SET                |
| 5385 | 063460 |        |        | FORCERROR     | 192\$                                                               | ;###FORCE ERROR IF FORCER=1         |
| 5386 | 063474 | 103407 |        | BCS           | 200\$                                                               | ;BR IF CARRY SET (GOOD RETURN)      |
| 5387 | 063476 | 010001 |        | MOV           | R0,R1                                                               | ;SAVE CONTENTS OF TSSR              |
| 5388 | 063500 |        |        | NEXT,ERRNO    |                                                                     |                                     |
| 5389 | 063500 |        |        | 192\$: ERRDF  | ERRNO,T208SSR,PKTSSR                                                | ;DEVICE FATAL SSR FAILED TO SET     |
|      | 063500 | 104455 |        |               |                                                                     | TRAP C\$ERDF                        |
|      | 063502 | 001614 |        |               |                                                                     | .WORD 908                           |
|      | 063504 | 065551 |        |               |                                                                     | .WORD T208SSR                       |
|      | 063506 | 012046 |        |               |                                                                     | .WORD PKTSSR                        |
| 5390 | 063510 | 005237 | 002222 | INC           | FATFLG                                                              | ;SET FATAL ERROR FLAG               |
| 5391 | 063514 |        |        | 200\$: CKLOOP |                                                                     | ;LOOP ON ERROR, IF FLAG SET         |
|      | 063514 | 104406 |        |               |                                                                     | TRAP C\$CLP1                        |
| 5392 |        |        |        | :             | Do a WRITE FORMAT to set IRESV4==>IRSTR = 1 (sets read strobe high) |                                     |
| 5393 | 063516 | 012700 | 000001 | MOV           | *WF,I4RES,R0                                                        | ;IRESV4==>IRSTR = 1                 |
| 5394 | 063522 | 004737 | 066262 | JSR           | PC,T20WFMT                                                          | ;SETUP T20PK2 FOR WRITE FORMAT      |

|                            |                             |                             |                            |
|----------------------------|-----------------------------|-----------------------------|----------------------------|
| ....B1                     | CKMSG - COMPARE WR....B5    | TEST 3: SUBTEST 1: G....B9  | TEST 6: SUBTEST 5: ....B13 |
| ....C1                     | CKMSG2 - COMPARE EX....C5   | TEST 3: SUBTEST 1: G....C9  | TEST 6: SUBTEST 5: ....C13 |
| ....D1                     | CKMSG2 - COMPARE EX....D5   | TEST 3: SUBTEST 2: M....D9  | TEST 6: SUBTEST 5: ....D13 |
| ....E1                     | PKTMES - PRINT TSSR....E5   | TEST 3: SUBTEST 2: M....E9  | TEST 6: SUBTEST 5: ....E13 |
| ....F1                     | FIFEXP - PRINT FIFO....F5   | TEST 3: SUBTEST 3: C....F9  | TEST 6: SUBTEST 5: ....F13 |
| ....G1                     | MSGSTAT - PRINT STAT....G5  | TEST 3: SUBTEST 3: C....G9  | TEST 6: SUBTEST 6: ....G13 |
| ....H1                     | MSGSUB - PRINT WRITE....H5  | TEST 3: SUBTEST 4: N....H9  | TEST 6: SUBTEST 6: ....H13 |
| ....I1                     | PRAMPKT - PRINT RAM ....I5  | TEST 3: SUBTEST 4: N....I9  | TEST 6: SUBTEST 6: ....I13 |
| ....J1                     | PRMESS - PRINT CONT....J5   | TEST 3: SUBTEST 4: N....J9  | TEST 6: SUBTEST 6: ....J13 |
| ....K1                     | PRMSGEXP - PRINT EXP....K5  | TEST 3: SUBTEST 4: N....K9  | TEST 6: SUBTEST 6: ....K13 |
| ....L1                     | PRBYTEXP - PRINT ERR....L5  | TEST 3: SUBTEST 4: N....L9  | TEST 6: SUBTEST 6: ....L13 |
| ....M1                     | PRBYTEXP - PRINT ERR....M5  | TEST 3: SUBTEST 4: N....M9  | TEST 6: SUBTEST 6: ....M13 |
| ....N1                     | RAMERR - PRINT RAM ....N5   | TEST 3: SUBTEST 4: N....N9  | TEST 6: SUBTEST 6: ....N13 |
| ....B2                     | RAMEXP - PRINT RAM ....B6   | TEST 3: SUBTEST 4: N....B10 | TEST 7: STATIC TRAN....B14 |
| ....C2                     | BADSSR - PRINT TSSR ....C6  | TEST 3: SUBTEST 4: N....C10 | TEST 7: STATIC TRAN....C14 |
| ....D2                     | CHKAMB - CHECK TSSR....D6   | TEST 4: RAM EXERCIS....D10  | TEST 7: STATIC TRAN....D14 |
| ....E2                     | ENAINTE,DSBINT - ENAB....E6 | TEST 4: RAM EXERCIS....E10  | TEST 7: STATIC TRAN....E14 |
| ....F2                     | WAITF - WAIT FOR S....F6    | TEST 4: RAM EXERCIS....F10  | TEST 7: STATIC TRAN....F14 |
| PROGRAM HEADER             | XNXM - CHECK FOR ....G6     | TEST 4: RAM EXERCIS....G10  | TEST 8: TRANSPORT B....G14 |
| DISPATCH TABLE             | TSTSETUP - PRINT TES....H6  | TEST 4: RAM EXERCIS....H10  | TEST 8: SUBTEST 1: ....H14 |
| DEFAULT HARDWARE P-T       | TSTSETUP - PRINT TES....I6  | TEST 4: RAM EXERCIS....I10  | TEST 8: SUBTEST 1: ....I14 |
| SOFTWARE P-TABLE           | INCERK - INCREMENT ....J6   | TEST 4: RAM EXERCIS....J10  | TEST 8: SUBTEST 1: ....J14 |
| SOFTWARE P-TABLE           | KTON,KTOFF - EN....K6       | TEST 5: EXTENDED FE....K10  | TEST 8: SUBTEST 1: ....K14 |
| GLOBAL EQUATES SECTI       | FILLMEM - FILL MEMOR....L6  | TEST 5: SUBTEST 1: ....L10  | TEST 8: SUBTEST 1: ....L14 |
| MEMORY MANAGEMENT DE       | CHPMEM - COMPARE ME....M6   | TEST 5: SUBTEST 1: ....M10  | TEST 8: SUBTEST 1: ....M14 |
| MEMORY MANAGEMENT DE       | CHPMEM - COMPARE ME....N6   | TEST 5: SUBTEST 1: ....N10  | TEST 8: SUBTEST 1: ....N14 |
| TSV05 REGISTER AND P....B3 | GETPAT - GET 8 BIT ....B7   | TEST 5: SUBTEST 1: ....B11  | TEST 8: SUBTEST 1: ....B15 |
| TSV05 REGISTER AND P....C3 | GETSEL - ISSUE MENU....C7   | TEST 5: SUBTEST 1: ....C11  | TEST 8: SUBTEST 2: ....C15 |
| TSV05 REGISTER AND P....D3 | CHKMAN - CHECK MANU....D7   | TEST 5: SUBTEST 2: ....D11  | TEST 8: SUBTEST 2: ....D15 |
| TSV05 REGISTER AND P....E3 | ENVIRN - SETUP FREE....E7   | TEST 5: SUBTEST 2: ....E11  | TEST 8: SUBTEST 2: ....E15 |
| TSV05 REGISTER AND P....F3 | KTINIT - SETUP KT11....F7   | TEST 5: SUBTEST 2: ....F11  | TEST 8: SUBTEST 2: ....F15 |
| TSV05 REGISTER AND P....G3 | KTINIT - SETUP KT11....G7   | TEST 5: SUBTEST 2: ....G11  | TEST 8: SUBTEST 2: ....G15 |
| SPECIAL MACROS AND O....H3 | KTINIT - SETUP KT11....H7   | TEST 5: SUBTEST 2: ....H11  | TEST 8: SUBTEST 3: ....H15 |
| GLOBAL DATA SECTION        | PROTECTION TABLE ....I7     | TEST 5: SUBTEST 2: ....I11  | TEST 8: SUBTEST 3: ....I15 |
| TSTBLK - TEST DATA         | INITIALIZE SECTION ....J7   | TEST 5: SUBTEST 2: ....J11  | TEST 8: SUBTEST 3: ....J15 |
| GLOBAL ENVIRONMENT S....K3 | INITIALIZE SECTION ....K7   | TEST 6: FIFO EXERCIS....K11 | TEST 8: SUBTEST 3: ....K15 |
| GLOBAL TEXT MESSAGES....L3 | INITIALIZE SECTION ....L7   | TEST 6: SUBTEST 1: ....L11  | TEST 8: SUBTEST 3: ....L15 |
| GLOBAL TEXT MESSAGES....M3 | ADD AND DROP UNITS S....M7  | TEST 6: SUBTEST 1: ....M11  | TEST 8: SUBTEST 4 L....M15 |
| GLOBAL ERROR REPORT ....N3 | ADD AND DROP UNITS S....N7  | TEST 6: SUBTEST 2: ....N11  | TEST 8: SUBTEST 4 L....N15 |
| PRITSSR - PRINT TSSR....B4 | CLEAN UP AND REPORT ....B8  | TEST 6: SUBTEST 2: ....B12  | TEST 8: SUBTEST 4 L....B16 |
| PRITSSR - PRINT TSSR....C4 | CLEAN-UP AND REPORT ....C8  | TEST 6: SUBTEST 2: ....C12  | TEST 8: SUBTEST 4 L....C16 |
| PRITSSR - PRINT TSSR....D4 | CLEAN-UP AND REPORT ....D8  | TEST 6: SUBTEST 2: ....D12  | TEST 8: SUBTEST 4 L....D16 |
| PRIPKT - PRINT THE ....E4  | TEST 1: INITIALIZE ....E8   | TEST 6: SUBTEST 2: ....E12  | TEST 8: SUBTEST 4 L....E16 |
| PRIXOR - PRINT EXPD....F4  | TEST 1: INITIALIZE ....F8   | TEST 6: SUBTEST 3: ....F12  | TEST 8: SUBTEST 4 L....F16 |
| PRIRAM - PRINT RAM ....G4  | TEST 1: INITIALIZE ....G8   | TEST 6: SUBTEST 3: ....G12  | TEST 8: SUBTEST 4 L....G16 |
| PRITADD - PRINT MEMO....H4 | TEST 1: INITIALIZE ....H8   | TEST 6: SUBTEST 3: ....H12  | TEST 8: SUBTEST 4 L....H16 |
| SPACE - SPACE RECO....I4   | TEST 2: BASIC WRITE ....I8  | TEST 6: SUBTEST 3: ....I12  | TEST 8: SUBTEST 4 L....I16 |
| SPACE - SPACE RECO....J4   | TEST 2: BASIC WRITE ....J8  | TEST 6: SUBTEST 3: ....J12  | TEST 9: READ/WRITE ....J16 |
| WRITCHR - WRITE CHAR....K4 | TEST 2: BASIC WRITE ....K8  | TEST 6: SUBTEST 3: ....K12  | TEST 9: READ/WRITE ....K16 |
| REWIND - POSITION T....L4  | TEST 2: BASIC WRITE ....L8  | TEST 6: SUBTEST 4: ....L12  | TEST 9: READ/WRITE ....L16 |
| CKRAM - COMPARE RA....M4   | TEST 2: BASIC WRITE ....M8  | TEST 6: SUBTEST 4: ....M12  | TEST 9: READ/WRITE ....M16 |
| CKRAM2 - COMPARE RA....N4  | TEST 2: BASIC WRITE ....N8  | TEST 6: SUBTEST 4: ....N12  |                            |

|      |        |        |        |        |                                                  |                      |       |                                       |
|------|--------|--------|--------|--------|--------------------------------------------------|----------------------|-------|---------------------------------------|
| 5395 | 063526 | 012704 | 066550 |        | MOV                                              | #T20PK2,R4           |       | ;GET WRITE SUBSYSTEM COMMAND PACKET   |
| 5396 | 063532 | 010465 | 000000 |        | MOV                                              | R4,TSD8(R5)          |       | ;SET THE PACKET ADDRESS TO EXECUTE    |
| 5397 | 063536 | 004737 | 016336 |        | JSR                                              | PC,CHKTSR            |       | ;WAIT FOR SSR TO SET                  |
| 5398 | 063542 |        |        |        | FORCERROR                                        | 2121                 |       | ;BDDFORCE ERROR IF FORCER=1           |
| 5399 | 063556 | 103407 |        |        | BCS                                              | 2201                 |       | ;BR IF CARRY SET (GOOD RETURN)        |
| 5400 | 063560 | 010001 |        |        | MOV                                              | R0,R1                |       | ;SAVE CONTENTS OF TSSR                |
| 5401 | 063562 |        |        |        | NEXT.ERRNO                                       |                      |       |                                       |
| 5402 | 063562 |        |        | 2121:  | ERRDF                                            | ERRNO,T208SSR,PKTSSR |       | ;DEVICE FATAL SSR FAILED TO SET       |
|      | 063562 | 104455 |        |        |                                                  |                      | TRAP  | C#ERDF                                |
|      | 063564 | 001615 |        |        |                                                  |                      | .WORD | 909                                   |
|      | 063566 | 065551 |        |        |                                                  |                      | .WORD | T208SSR                               |
|      | 063570 | 012046 |        |        |                                                  |                      | .WORD | PKTSSR                                |
| 5403 | 063572 | 005237 | 002222 |        | INC                                              | FATFLG               |       | ;SET FATAL ERROR FLAG                 |
| 5404 | 063576 |        |        | 2201:  | CKLOOP                                           |                      |       | ;LOOP ON ERROR, IF FLAG SET           |
|      | 063576 | 104406 |        |        |                                                  |                      | TRAP  | C#CLP1                                |
| 5405 |        |        |        | :      | Do a Write Subsystem READ STATUS                 |                      |       |                                       |
| 5406 | 063600 | 004737 | 066146 |        | JSR                                              | PC,T20SRD            |       | ;SETUP PACKET FOR READ STATUS         |
| 5407 | 063604 | 012704 | 066550 |        | MOV                                              | #T20PK2,R4           |       | ;GET WRITE SUBSYSTEM COMMAND PACKET   |
| 5408 | 063610 | 010465 | 000000 |        | MOV                                              | R4,TSD8(R5)          |       | ;SET THE PACKET ADDRESS TO EXECUTE    |
| 5409 | 063614 | 004737 | 016336 |        | JSR                                              | PC,CHKTSR            |       | ;WAIT FOR SSR TO SET                  |
| 5410 | 063620 |        |        |        | FORCERROR                                        | 2321                 |       | ;BDDFORCE ERROR IF FORCER=1           |
| 5411 | 063634 | 103407 |        |        | BCS                                              | 2401                 |       | ;BR IF CARRY SET (GOOD RETURN)        |
| 5412 | 063636 | 010001 |        |        | MOV                                              | R0,R1                |       | ;SAVE CONTENTS OF TSSR                |
| 5413 | 063640 |        |        |        | NEXT.ERRNO                                       |                      |       |                                       |
| 5414 | 063640 |        |        | 2321:  | ERRDF                                            | ERRNO,T203SSR,PKTSSR |       | ;DEVICE FATAL SSR FAILED TO SET       |
|      | 063640 | 104455 |        |        |                                                  |                      | TRAP  | C#ERDF                                |
|      | 063642 | 001616 |        |        |                                                  |                      | .WORD | 910                                   |
|      | 063644 | 063332 |        |        |                                                  |                      | .WORD | T203SSR                               |
|      | 063646 | 012046 |        |        |                                                  |                      | .WORD | PKTSSR                                |
| 5415 | 063650 | 005237 | 002222 |        | INC                                              | FATFLG               |       | ;SET FATAL ERROR FLAG                 |
| 5416 | 063654 |        |        | 2401:  | CKLOOP                                           |                      |       | ;LOOP ON ERROR, IF FLAG SET           |
|      | 063654 | 104406 |        |        |                                                  |                      | TRAP  | C#CLP1                                |
| 5417 |        |        |        | :      | If Read Data parity error NOT=1 Then Print Error |                      |       |                                       |
| 5418 | 063656 | 004737 | 066354 |        | JSR                                              | PC,T20SETEXP         |       | ;SET WORDS 0-7 EXPD=RCV (NOT TESTING) |
| 5419 | 063662 | 012701 | 065102 |        | MOV                                              | #T20EXSTA,R1         |       | ;GET EXPECTED READ STATUS             |
| 5420 | 063666 | 012702 | 066442 |        | MOV                                              | #T20BFSTA,R2         |       | ;GET RECV READ STATUS                 |
| 5421 | 063672 | 011211 |        |        | MOV                                              | (R2),(R1)            |       | ;SET EXPD WORD #8 = RCV TEMP          |
| 5422 | 063674 | 016261 | 000002 | 000002 | MOV                                              | 2(R2),2(R1)          |       | ;SET EXPD WORD #9 = RCV (NOT TESTED)  |
| 5423 | 063702 | 052711 | 100000 |        | BIS                                              | #S1.PARERR,(R1)      |       | ;SET EXP PAR ERR =1                   |
| 5424 | 063706 | 005000 |        |        | CLR                                              | R0                   |       | ;HIGH RECV ADDRESS FOR CKMSG2         |
| 5425 | 063710 | 012701 | 066422 |        | MOV                                              | #T20BFR,R1           |       | ;LOW RECV ADDRESS FOR CKMSG2          |
| 5426 | 063714 | 012702 | 065062 |        | MOV                                              | #T20EXP,R2           |       | ;EXPD ADDRESS                         |
| 5427 | 063720 | 012703 | 000024 |        | MOV                                              | #20..R3              |       | ;NUMBER OF BYTES TO COMPARE           |
| 5428 | 063724 | 004737 | 011500 |        | JSR                                              | PC,CKMSG2            |       | ;EXPD EQUAL RECV?                     |
| 5429 | 063730 |        |        |        | FORCERROR                                        | 2521,NOTSSR          |       | ;BDD                                  |
| 5430 | 063740 | 103404 |        |        | BCS                                              | 2601                 |       | ;BR IF YES                            |
| 5431 | 063742 |        |        |        | NEXT.ERRNO                                       |                      |       |                                       |
| 5432 | 063742 |        |        | 2521:  | ERRHFD                                           | ERRNO,T20SWP,MSGSTAT |       | ;REPORT ERROR                         |
|      | 063742 | 104456 |        |        |                                                  |                      | TRAP  | C#ERRHD                               |
|      | 063744 | 001617 |        |        |                                                  |                      | .WORD | 911                                   |
|      | 063746 | 065617 |        |        |                                                  |                      | .WORD | T20SWP                                |
|      | 063750 | 012350 |        |        |                                                  |                      | .WORD | MSGSTAT                               |
| 5433 | 063752 |        |        | 2601:  | CKLOOP                                           |                      |       | ;LOOP ON ERROR, IF FLAG SET           |
|      | 063752 | 104406 |        |        |                                                  |                      | TRAP  | C#CLP1                                |
| 5434 |        |        |        | :      | Do a Write Misc to RESET FIFO                    |                      |       |                                       |
| 5435 | 063754 | 012700 | 000020 |        | MOV                                              | #MS.RSFIF,R0         |       | ;SET RESET FIFO COMMAND               |
| 5436 | 063760 | 004737 | 06     |        |                                                  |                      |       |                                       |

C1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 209

|      |        |        |        |            |                                                                     |   |                                      |         |
|------|--------|--------|--------|------------|---------------------------------------------------------------------|---|--------------------------------------|---------|
| 5437 | 063764 | 012704 | 066550 | MOV        | #T20PK2,H4                                                          | ; | GET WRITE SUBSYSTEM COMMAND PACKET   |         |
| 5438 | 063770 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                                         | ; | SET THE PACKET ADDRESS TO EXECUTE    |         |
| 5439 | 063774 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                          | ; | WAIT FOR SSR TO SET                  |         |
| 5440 | 064000 |        |        | FORCERROR  | 282#                                                                | ; | FORCE ERROR IF FORCER=1              |         |
| 5441 | 064014 | 103407 |        | BCS        | 290#                                                                | ; | BR IF CARRY SET (GOOD RETURN)        |         |
| 5442 | 064016 | 010001 |        | MOV        | RO,R1                                                               | ; | SAVE CONTENTS OF TSSR                |         |
| 5443 | 064020 |        |        | NEXT,ERRNO |                                                                     |   |                                      |         |
| 5444 | 064020 |        |        | 282#:      | ERRDF ERRNO,T202SSR,PKTSSR                                          | ; | DEVICE FATAL SSR FAILED TO SET       |         |
|      | 064020 | 104455 |        |            |                                                                     |   | TRAP                                 | C#ERDF  |
|      | 064022 | 001620 |        |            |                                                                     |   | .WORD                                | 912     |
|      | 064024 | 065266 |        |            |                                                                     |   | .WORD                                | T202SSR |
|      | 064026 | 012046 |        |            |                                                                     |   | .WORD                                | PKTSSR  |
| 5445 | 064030 | 005237 | 002222 | INC        | FATFLG                                                              | ; | SET FATAL ERROR FLAG                 |         |
| 5446 | 064034 |        |        | 290#:      | CKLOOP                                                              | ; | LOOP ON ERROR, IF FLAG SET           |         |
|      | 064034 | 104406 |        |            |                                                                     |   | TRAP                                 | C#CLP1  |
| 5447 |        |        |        | ;          | Do a Write Subsystem READ STATUS                                    |   |                                      |         |
| 5448 |        |        |        | ;          | IF Read Data parity error NOT=0 Then Print Error                    |   |                                      |         |
| 5449 | 064036 | 004737 | 066354 | JSR        | PC,T20SETEXP                                                        | ; | SET WORDS 0-7 EXPD=RCV (NOT TESTING) |         |
| 5450 | 064042 | 012701 | 065102 | MOV        | #T20EXSTA,R1                                                        | ; | GET EXPECTED READ STATUS             |         |
| 5451 | 064046 | 012702 | 066442 | MOV        | #T20FSTA,R2                                                         | ; | GET RCV READ STATUS                  |         |
| 5452 | 064052 | 011211 |        | MOV        | (R2),(R1)                                                           | ; | SET EXPD WORD #8 = RCV TEMP          |         |
| 5453 | 064054 | 016261 | 000002 | MOV        | 2(R2),2(R1)                                                         | ; | SET EXPD WORD #9 = RCV (NOT TESTED)  |         |
| 5454 | 064062 | 042711 | 100000 | BIC        | #S1.PARERR,(R1)                                                     | ; | SET EXP PAR ERR =0                   |         |
| 5455 | 064066 | 005000 |        | CLR        | RO                                                                  | ; | HIGH RCV ADDRESS FOR CKMSG2          |         |
| 5456 | 064070 | 012701 | 066422 | MOV        | #T208FR,R1                                                          | ; | LOW RCV ADDRESS FOR CKMSG2           |         |
| 5457 | 064074 | 012702 | 065062 | MOV        | #T20EXP,R2                                                          | ; | EXPD ADDRESS                         |         |
| 5458 | 064100 | 012703 | 000024 | MOV        | #20.,R3                                                             | ; | NUMBER OF BYTES TO COMPARE           |         |
| 5459 | 064104 | 004737 | 011500 | JSR        | PC,CKMSG2                                                           | ; | EXPD EQUAL RCV?                      |         |
| 5460 | 064110 |        |        | FORCERROR  | 302#,NOTSSR                                                         | ; | NO                                   |         |
| 5461 | 064120 | 103404 |        | BCS        | 320#                                                                | ; | BR IF YES                            |         |
| 5462 | 064122 |        |        | NEXT,ERRNO |                                                                     |   |                                      |         |
| 5463 | 064122 |        |        | 302#:      | ERRHRD ERRNO,T20RSF,MSGSTAT                                         | ; | REPORT ERROR                         |         |
|      | 064122 | 104456 |        |            |                                                                     |   | TRAP                                 | C#ERHRD |
|      | 064124 | 001621 |        |            |                                                                     |   | .WORD                                | 913     |
|      | 064126 | 065726 |        |            |                                                                     |   | .WORD                                | T20RSF  |
|      | 064130 | 012350 |        |            |                                                                     |   | .WORD                                | MSGSTAT |
| 5464 | 064132 |        |        | 320#:      | CKLOOP                                                              | ; | LOOP ON ERROR, IF FLAG SET           |         |
|      | 064132 | 104406 |        |            |                                                                     |   | TRAP                                 | C#CLP1  |
| 5465 |        |        |        | ;          | (* Verify Data can be transferred without a Parity Error *)         |   |                                      |         |
| 5466 |        |        |        | ;          | Do a WRITE FORMAT to set IRESV4==>IRSTR = 1 (sets read strobe high) |   |                                      |         |
| 5467 | 064134 | 012700 | 000001 | MOV        | #WF.I4RES,RO                                                        | ; | IRESV4==>IRSTR = 1                   |         |
| 5468 | 064140 | 004737 | 066262 | JSR        | PC,T20WFMT                                                          | ; | SETUP T20PK2 FOR WRITE FORMAT        |         |
| 5469 | 064144 | 012704 | 066550 | MOV        | #T20PK2,R4                                                          | ; | GET WRITE SUBSYSTEM COMMAND PACKET   |         |
| 5470 | 064150 | 010465 | 000000 | MOV        | R4,TSD8(R5)                                                         | ; | SET THE PACKET ADDRESS TO EXECUTE    |         |
| 5471 | 064154 | 004737 | 016336 | JSR        | PC,CHKTSSR                                                          | ; | WAIT FOR SSR TO SET                  |         |
| 5472 | 064160 |        |        | FORCERROR  | 332#                                                                | ; | FORCE ERROR IF FORCER=1              |         |
| 5473 | 064174 | 103407 |        | BCS        | 340#                                                                | ; | BR IF CARRY SET (GOOD RETURN)        |         |
| 5474 | 064176 | 010001 |        | MOV        | RO,R1                                                               | ; | SAVE CONTENTS OF TSSR                |         |
| 5475 | 064200 |        |        | NEXT,ERRNO |                                                                     |   |                                      |         |
| 5476 | 064200 |        |        | 332#:      | ERRDF ERRNO,T208SSR,PKTSSR                                          | ; | DEVICE FATAL SSR FAILED TO SET       |         |
|      | 064200 | 104455 |        |            |                                                                     |   | TRAP                                 | C#ERDF  |
|      | 064202 | 001622 |        |            |                                                                     |   | .WORD                                | 914     |
|      | 064204 | 065551 |        |            |                                                                     |   | .WORD                                | T208SSR |
|      | 064206 | 012046 |        |            |                                                                     |   | .WORD                                | PKTSSR  |
| 5477 | 064210 | 005237 | 002222 | INC        | FATFLG                                                              | ; | SET FATAL ERROR FLAG                 |         |
| 5478 | 064214 |        |        | 340#:      | CKLOOP                                                              | ; | LOOP ON ERROR, IF FLAG SET           |         |
|      | 064214 | 104406 |        |            |                                                                     |   | TRAP                                 | C#CLP1  |

D1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 210

```

5479      ; Do a WRITE NPR to set loopback and tape direction OUT and
5480      ; and CLEAR Write Wrong Par ty.
5481 064216 012700 000100      MOV      #NP.OUT,R0      ;SET TAPE DIRECTION OUT
5482 064222 052700 000040      BIS      #NP.LOOP,R0      ;SET LOOPBACK
5483 064226 052700 000020      BIS      #NP.WRP,R0      ;CLEAR WRITE WRONG PARITY (INVERTED)
5484 064232 004737 066166      JSR      PC,T20WNPR      ;SETUP T20PK2 FOR WRITE NPR
5485 064236 012704 066550      MOV      #T20PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5486 064242 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
5487 064246 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5488 064252      FORCERROR      352$      ;GOODFORCE ERROR IF FORCER=1
5489 064266 103407      BCS      360$      ;BR IF CARRY SET (GOOD RETURN)
5490 064270 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
5491 064272      NEXT.ERRNO
5492 064272 352$: ERRDF      ERRNO,T204SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      TRAP      C$ERDF
      .WORD      915
      .WORD      T204SSR
      .WORD      PKTSSR
5493 064302 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
5494 064306 360$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      TRAP      C$CLP1
5495      ; Do a WRITE FIFO with byte count equal to 1 and Tape direction OUT
5496 064310 012700 000001      MOV      #1,R0      ;WRITE 1 BYTE
5497 064314 012701 002312      MOVB     #DATA,R1      ;FIFO WRITE DATA ADDRESS
5498 064320 004737 066226      JSR      PC,T20WFIF      ;SETUP T20PK2 FOR WRITE FIFO
5499 064324 012704 066550      MOV      #T20PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5500 064330 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
5501 064334 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5502 064340      FORCERROR      372$      ;GOODFORCE ERROR IF FORCER=1
5503 064354 103407      BCS      380$      ;BR IF CARRY SET (GOOD RETURN)
5504 064356 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
5505 064360      NEXT.ERRNO
5506 064360 372$: ERRDF      ERRNO,T205SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      TRAP      C$ERDF
      .WORD      916
      .WORD      T205SSR
      .WORD      PKTSSR
5507 064370 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
5508 064374 380$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      TRAP      C$CLP1
5509      ; Do a READ FIFO with tape direction OUT to load tape out write latch
5510 064376 012700 000001      MOV      #1,R0      ;SET READ BYTE COUNT
5511 064402 004737 066206      JSR      PC,T20RFIF      ;SETUP T20PK2 FOR READ FIFO
5512 064406 012704 066550      MOV      #T20PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5513 064412 010465 000000      MOV      R4,TSD8(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
5514 064416 004737 016336      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5515 064422      FORCERROR      392$      ;GOODFORCE ERROR IF FORCER=1
5516 064436 103407      BCS      400$      ;BR IF CARRY SET (GOOD RETURN)
5517 064440 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
5518 064442      NEXT.ERRNO
5519 064442 392$: ERRDF      ERRNO,T206SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      TRAP      C$ERDF
      .WORD      917
      .WORD      T206SSR
      .WORD      PKTSSR
5520 064452 005237 002222      INC      FATFLG      ;SET FATAL ERROR FLAG
5521 064456 400$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET

```

E1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 211

```

5522 064456 104406 ; Do a WRITE FORMAT to set IRESV4==>IRSTR = 0 (sets read strobe low) TRAP C$CLP1
5523 ; (Read Strobe sets PAR IN H (Parity Error))
5524 064460 005000 CLR R0 ; IRESV4==>IRSTR = 0
5525 064462 004737 066262 JSR PC,T20WFM ; SETUP T20PK2 FOR WRITE FORMAT
5526 064466 012704 066550 MOV #T20PK2,R4 ; GET WRITE SUBSYSTEM COMMAND PACKET
5527 064472 010465 000000 MOV R4,TSDB(R5) ; SET THE PACKET ADDRESS TO EXECUTE
5528 064476 004737 016336 JSR PC,CHKTSSR ; WAIT FOR SSR TO SET
5529 064502 FORCERROR 412$ ; GOODFORCE ERROR IF FORCER=1
5530 064516 103407 BCS 420$ ; BR IF CARRY SET (GOOD RETURN)
5531 064520 010001 MOV R0,R1 ; SAVE CONTENTS OF TSSR
5532 064522 NEXT,ERRNO
5533 064522 412$: ERDF ERRNO,T208SSR,PKTSSR ; DEVICE FATAL SSR FAILED TO SET TRAP C$ERDF
; ; .WORD 918
; ; .WORD T208SSR
; ; .WORD PKTSSR
5534 064532 005237 002222 INC FATFLG ; SET FATAL ERROR FLAG
5535 064536 420$: CKLOOP ; LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
5536 064536 104406 ; Do a WRITE FORMAT to set IRESV4==>IRSTR = 1 (sets read strobe high)
5537 064540 012700 000001 MOV #WF,I4RES,R0 ; IRESV4==>IRSTR = 1
5538 064544 004737 066262 JSR PC,T20WFM ; SETUP T20PK2 FOR WRITE FORMAT
5539 064550 012704 066550 MOV #T20PK2,R4 ; GET WRITE SUBSYSTEM COMMAND PACKET
5540 064554 010465 000000 MOV R4,TSDB(R5) ; SET THE PACKET ADDRESS TO EXECUTE
5541 064560 004737 016336 JSR PC,CHKTSSR ; WAIT FOR SSR TO SET
5542 064564 FORCERROR 432$ ; GOODFORCE ERROR IF FORCER=1
5543 064600 103407 BCS 440$ ; BR IF CARRY SET (GOOD RETURN)
5544 064602 010001 MOV R0,R1 ; SAVE CONTENTS OF TSSR
5545 064604 NEXT,ERRNO
5546 064604 432$: ERDF ERRNO,T208SSR,PKTSSR ; DEVICE FATAL SSR FAILED TO SET TRAP C$ERDF
; ; .WORD 919
; ; .WORD T208SSR
; ; .WORD PKTSSR
5547 064614 005237 002222 INC FATFLG ; SET FATAL ERROR FLAG
5548 064620 440$: CKLOOP ; LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
5549 064620 104406 ; Do a Write Subsystem READ STATUS
5550 ; JSR PC,T20SRD ; SETUP PACKET FOR READ STATUS
5551 064622 004737 066146 MOV #T20PK2,R4 ; GET WRITE SUBSYSTEM COMMAND PACKET
5552 064626 012704 066550 MOV R4,TSDB(R5) ; SET THE PACKET ADDRESS TO EXECUTE
5553 064632 010465 000000 JSR PC,CHKTSSR ; WAIT FOR SSR TO SET
5554 064636 004737 016336 FORCERROR 452$ ; GOODFORCE ERROR IF FORCER=1
5555 064642 FORCERROR 452$ ; GOODFORCE ERROR IF FORCER=1
5556 064656 103407 BCS 460$ ; BR IF CARRY SET (GOOD RETURN)
5557 064660 010001 MOV R0,R1 ; SAVE CONTENTS OF TSSR
5558 064662 NEXT,ERRNO
5559 064662 452$: ERDF ERRNO,T203SSR,PKTSSR ; DEVICE FATAL SSR FAILED TO SET TRAP C$ERDF
; ; .WORD 920
; ; .WORD T203SSR
; ; .WORD PKTSSR
5560 064672 005237 002222 INC FATFLG ; SET FATAL ERROR FLAG
5561 064676 460$: CKLOOP ; LOOP ON ERROR, IF FLAG SET TRAP C$CLP1
5562 064676 104406 ; If Read Data parity error NOT=0 Then Print Error

```

F1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 212

|                                  |            |                      |                                         |
|----------------------------------|------------|----------------------|-----------------------------------------|
| 5563 064700 004737 066354        | JSR        | PC,T20SETEXP         | SET WORDS 0-7 EXPD=RECV (NOT TESTING)   |
| 5564 064704 012701 065102        | MOV        | #T20EXSTA,R1         | GET EXPECTED READ STATUS                |
| 5565 064710 012702 066442        | MOV        | #T20BFSTA,R2         | GET RECV READ STATUS                    |
| 5566 064714 011211               | MOV        | (R2),(R1)            | SET EXPD WORD #8 = RECV TEMP            |
| 5567 064716 016261 000002 000002 | MOV        | 2(R2),2(R1)          | SET EXPD WORD #9 = RECV (NOT TESTED)    |
| 5568 064724 042711 100000        | BIC        | #S1.PARERR,(R1)      | SET EXP PAR ERR =0                      |
| 5569 064730 005000               | CLR        | R0                   | HIGH RECV ADDRESS FOR CKMSG2            |
| 5570 064732 012701 066422        | MOV        | #T20BFR,R1           | LOW RECV ADDRESS FOR CKMSG2             |
| 5571 064736 012702 065062        | MOV        | #T20EXP,R2           | EXPD ADDRESS                            |
| 5572 064742 012703 000024        | MOV        | #20.,R3              | NUMBER OF BYTES TO COMPARE              |
| 5573 064746 004737 011500        | JSR        | PC,CKMSG2            | EXPD EQUAL RECV?                        |
| 5574 064752                      | FORCERROR  | 472\$,NOTSSR         | 000                                     |
| 5575 064762 103404               | BCS        | 480\$                | BR IF YES                               |
| 5576 064764                      | NEXT,ERRNO |                      |                                         |
| 5577 064764                      | ERRHRD     | ERRNO,T20CWP,MSGSTAT | REPORT ERROR                            |
| 064764 104456                    |            |                      | TRAP C\$ERHRD                           |
| 064766 001631                    |            |                      | .WORD 921                               |
| 064770 066027                    |            |                      | .WORD T20CWP                            |
| 064772 012350                    |            |                      | .WORD MSGSTAT                           |
| 5578 064774                      | 480\$:     | CKLOOP               | LOOP ON ERROR, IF FLAG SET              |
| 064774 104406                    |            |                      | TRAP C\$CLP1                            |
| 5579                             |            |                      |                                         |
| 5580 064776                      | FORCEEXIT  | 555\$                | 000                                     |
| 5581 065006 013703 002316        | MOV        | TSTPTR,R3            | RESTORE CURRENT TSTBLK POINTER          |
| 5582 065012 020327 003062        | CMP        | R3,#TBLEND           | END OF TSTBLK?                          |
| 5583 065016 103002               | BHIS       | 555\$                | BR IF YES                               |
| 5584 065020 000137 063074        | JMP        | 100\$                | DO ANOTHER TSTBLK PATTERN               |
| 5585 065024                      | 555\$:     |                      |                                         |
| 5586                             |            |                      |                                         |
| 5587 065024                      | ENDSUB     |                      | ////////// END SUBTEST //////////       |
| 065024 104403                    |            |                      | L10074:                                 |
| 5588                             |            |                      | TRAP C\$ESUB                            |
| 5589 065026 005737 002222        | TST        | FATFLG               | ANY FATAL ERRORS ?                      |
| 5590 065032 001402               | BEQ        | 560\$                | BRANCH IF NOT                           |
| 5591 065034 004737 017202        | JSR        | PC,CKDROP            | TRY TO DROP THE UNIT                    |
| 5592 065040                      | 560\$:     |                      |                                         |
| 5593 065040 004737 016456        | JSR        | PC,TSTLOOP           | DO ITERATIONS?                          |
| 5594 065044 103002               | BCC        | 565\$                | BR IF NO                                |
| 5595 065046 000137 050526        | JMP        | T18LOOP              | LOOP UNTIL ITERATIONS DONE              |
| 5596 065052                      | 565\$:     |                      |                                         |
| 5597 065052                      | EXIT       | TST                  | ////////// EXIT TEST //////////         |
| 065052 104432                    |            |                      | TRAP C\$EXIT                            |
| 065054 001610                    |            |                      | .WORD L10073-                           |
| 5598                             |            |                      |                                         |
| 5599                             |            |                      |                                         |
| 5600                             |            |                      |                                         |
| 5601                             |            |                      |                                         |
| 5602                             |            |                      |                                         |
| 5603                             |            |                      |                                         |
| 5604                             |            |                      |                                         |
| 5605 065056                      | T20MSK:    |                      | MASK OF UNTESTED BITS IN READ STATUS    |
| 5606                             |            |                      | UNTESTED BITS ARE SET TO 1              |
| 5607 065056 377                  | .BYTE      | #C<000>              | BYTE 0 MASK                             |
| 5608 065057 037                  | .BYTE      | #C<340>              | BYTE 1 MASK (PARERR,IRESV2,IRESV1)      |
| 5609 065060 360                  | .BYTE      | #C<017>              | BYTE 2 (TIMER A,TIMER B,UNDEFINED<1;0>) |
| 5610 065061 000                  | .BYTE      | 0                    | MAKE IT EVEN                            |

```

;
;LOCAL STORAGE FOR THIS TEST
;

```

G1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 213

```

5611
5612 065062          T20EXP:          ;BEGIN EXPECTED DATA BUFFER
5613 065062 000000          .WORD 0          ;MESSAGE TYPE
5614 065064 000000          .WORD 0          ;DATA FIELD LENGTH
5615 065066 000000          .WORD 0          ;RBPGR
5616 065070 000000          .WORD 0          ;XST0
5617 065072 000000          .WORD 0          ;XST1
5618 065074 000000          .WORD 0          ;XST2
5619 065076 000000          .WORD 0          ;XST3
5620 065100 000000          .WORD 0          ;XST4 (ALWAYS PRESENT FOR WRITE SUB.)
5621 065102          T20EXSTA: .BLKB 64.      ;EXPECTED READ STATUS AND WRITE FIFO DATA
5622 065202          T20EXEND:              ;END EXPECTED DATA BUFFER
5623
5624          ;*
5625          ;LOCAL TEXT MESSAGES FOR TEST
5626          ;*
5627 065202          122      145      141  TST20ID:          .ASCIIZ 'Read/Write Data Parity'
5628 065231          127      122      111  T20SSR: .ASCIIZ 'WRITE CHARACTERISTICS Failed'
5629 065266          127      122      111  T202SSR: .ASCIIZ 'WRITE SUBSYSTEM (Write Misc) Failed'
5630 065332          127      122      111  T203SSR: .ASCIIZ 'WRITE SUBSYSTEM (Read Status) Failed'
5631 065377          127      122      111  T204SSR: .ASCIIZ 'WRITE SUBSYSTEM (Write Npr) Failed'
5632 065442          127      122      111  T205SSR: .ASCIIZ 'WRITE SUBSYSTEM (Write FIFO) Failed'
5633 065506          127      122      111  T206SSR: .ASCIIZ 'WRITE SUBSYSTEM (Read FIFO) Failed'
5634 065551          127      122      111  T208SSR: .ASCIIZ 'WRITE SUBSYSTEM (Write Format) Failed'
5635 065617          122      145      141  T20SWP: .ASCIIZ 'Read Data Parity Error (PARERR) Failed to Set after Write Wrong Parity'
5636 065726          122      145      141  T20RSF: .ASCIIZ 'Read Data Parity Error (PARERR) Failed to Clear after RESET FIFO'
5637 066027          122      145      141  T20CWP: .ASCIIZ 'Read Data Parity Error (PARERR) occurred in Data Loopback'
5638
5639
5640
5641          ;*
5642          ; CLEAR MESSAGE BUFFER
5643          ;*
5643 066122          T20CLRBUF:
5644 066122          SAVREG
5645 066126 012701 066422          MOV      #T20BFR,R1          ;SAVE R1-R5 UNTIL NEXT RETURN
5646 066132 012702 000120          MOV      #T20BEND-T20BFR,R2      ;GET MESSAGE BUFFER ADDRESS
5647 066136 105021          10$: CLR      (R1)+          ;SIZE OF MESSAGE BUFFER IN BYTES
5648 066140 005302          DEC      R2          ;CLEAR A BYTE
5649 066142 003375          BGT      10$          ;DONE?
5650 066144 000207          RTS      PC          ;BR IF NO
5651
5652
5653          ;*
5654          ; SETUP T20PK2 PACKET FOR READ STATUS
5655          ;*
5655 066146          T20SRD:
5656 066146 004737 066122          JSR      PC,T20CLRBUF          ;CLEAR MESSAGE BUFFER
5657 066152 012700 066560          MOV      #T20DT2,R0          ;WRITE SUBSYSTEM DATA BUFFER
5658 066156 112720 000005          MOV      #PW,RDSTATUS,(R0)+      ;STORE READ STATUS COMMAND IN BSEL0
5659 066162 105010          CLR      (R0)          ;CLEAR BSEL1
5660 066164 000207          RTS      PC          ;RETURN
5661
5662
5663          ;*
5664          ; SETUP T20PK2 PACKET FOR WRITE NPR
5665          ;*
5666          ; INPUT:
5667          ; RO CONTAINS BSEL1 NPR DATA

```

```

5668
5669
5670 066166
5671 066166 004737 066122
5672 066172 012701 066560
5673 066176 112721 000011
5674 066202 110011
5675 066204 000207
5676
5677
5678
5679
5680
5681
5682
5683 066206
5684 066206 004737 066122
5685 066212 012701 066560
5686 066216 112721 000003
5687 066222 110021
5688 066224 000207
5689
5690
5691
5692
5693
5694
5695
5696 066226
5697 066226
5698 066232 004737 066122
5699 066236 012702 066560
5700 066242 112722 000004
5701 066246 110022
5702 066250 005022
5703 066252 112122
5704 066254 005300
5705 066256 003375
5706 066260 000207
5707
5708
5709
5710
5711
5712
5713
5714 066262
5715 066262 004737 066122
5716 066266 012701 066560
5717 066272 112721 000007
5718 066276 110021
5719 066300 000207
5720
5721
5722
5723
5724

;
;
; T2OWNPR:
;     JSR     PC,T2OCLRBUF      ;CLEAR MESSAGE BUFFER
;     MOV     @T2ODT2,R1        ;WRITE SUBSYSTEM DATA BUFFER
;     MOVB    @PW.WNPR,(R1)+    ;STORE WRITE NPR IN BSEL0
;     MOVB    R0,(R1)           ;STORE NPR DATA IN BSEL1
;     RTS     PC                ;RETURN

;
; *
; SETUP T20PK2 PACKET FOR READ FIFO
;
; INPUT:
;     R0 CONTAINS SEL2 BYTE COUNT
;
; T2ORFIF:
;     JSR     PC,T2OCLRBUF      ;CLEAR MESSAGE BUFFER
;     MOV     @T2ODT2,R1        ;WRITE SUBSYSTEM DATA BUFFER
;     MOVB    @PW.RFIFO,(R1)+   ;STORE READ FIFO IN BSEL0
;     MOVB    R0,(R1)+          ;STORE BYTE COUNT IN BSEL1
;     RTS     PC                ;RETURN

;
; *
; SETUP T20PK2 PACKET FOR WRITE FIFO
;
; INPUT:
;     R0 CONTAINS BYTE COUNT
;     R1 CONTAINS DATA PATTERN BLOCK ADDRESS
;
; T2OWFIF:
;     SAVREG
;     JSR     PC,T2OCLRBUF      ;SAVE R1-R5 UNTIL NEXT RETURN
;     MOV     @T2ODT2,R2        ;CLEAR MESSAGE BUFFER
;     MOVB    @PW.WFIFO,(R2)+   ;WRITE SUBSYSTEM DATA BUFFER
;     MOVB    R0,(R2)+          ;STORE WRITE FIFO IN BSEL0
;     MOVB    R0,(R2)+          ;STORE BYTE COUNT IN BSEL1
;     CLR     (R2)+             ;CLEAR SEL2 (UNUSED)
;     MOVB    (R1)+,(R2)+       ;STORE DATA PATTERN BYTE
;     DEC     R0                ;DONE ALL BYTES?
;     BGT     10$               ;BR IF NO
;     RTS     PC                ;RETURN

10$:
;
; *
; SETUP T20PK2 FOR WRITE FORMAT TRANSPORT REGISTER
;
; INPUT:
;     R0 CONTAINS DRIVING DATA PATTERN
;
; T2OWFMT:
;     JSR     PC,T2OCLRBUF      ;CLEAR MESSAGE BUFFER
;     MOV     @T2ODT2,R1        ;WRITE SUBSYSTEM DATA BUFFER
;     MOVB    @PW.WFMT,(R1)+   ;STORE WRITE FORMAT IN BSEL0
;     MOVB    R0,(R1)+          ;STORE DATA WORD IN BSEL1
;     RTS     PC                ;RETURN

;
; *
; SETUP T20PK2 PACKET FOR WRITE MISC.
;
;     R0 CONTAINS WRITE MISC DATA

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 215

```

5725 066302
5726 066302 012701 065560
5727 066306 112721 000010
5728 066312 110011 000010
5729 066314 000207
5730
5731
5732
5733 066316
5734 066316 012700 065560
5735 066322 112720 000010
5736 066326 112710 000200
5737 066332 000207
5738
5739
5740
5741 066334
5742 066334 012701 065062
5743 066340 012700 000120
5744 066344 105021
5745 066346 005300
5746 066350 003375
5747 066352 000207
5748
5749
5750
5751
5752 066354
5753 066354 012702 065062
5754 066360 012703 066422
5755 066364 012700 000010
5756 066370 012322
5757 066372 005300
5758 066374 003375
5759 066376 000207
5760
5761
5762
5766
5767
5768
5769 066400
5770 066400 100004
5771 066402 066410
5772 066404 000000
5773 066406 000012
5774
5775 066410
5776 066410 066422
5777 066412 000000
5778 066414 000024
5779 066416 000000
5780 066420 000007
5781
5782
5783
5784

T20WMISC:
MOV    #T20DT2,R1          ;WRITE SUBSYSTEM DATA BUFFER
MOVB   #PW.WMISC,(R1)+      ;STORE WRITE MISCELLANEOUS IN BSEL0
MOVB   R0,(R1)              ;STORE INVERT EXTENDED FEATURES IN BSEL1
RTS     PC                  ;RETURN

;
; SETUP T20PK2 PACKET FOR WRITE MISC. INVERT EXTENDED FEATURES SWITCH
;
T20SEXT:
MOV     #T20DT2,R0          ;WRITE SUBSYSTEM DATA BUFFER
MOVB    #PW.WMISC,(R0)+     ;STORE WRITE MISCELLANEOUS IN BSEL0
MOVB    #MS.EXT,(R0)        ;STORE INVERT EXTENDED FEATURES IN BSEL1
RTS     PC                  ;RETURN

;
; CLEAR EXPECTED DATA MESSAGE BUFFER
;
T20CLEXP:
MOV     #T20EXP,R1          ;GET EXPD ADDRESS
MOV     #T20EXEND-T20EXP,R0 ;GET EXPD SIZE
10$:   CLRB   (R1)+          ;CLEAR A BYTE
DEC     R0                  ;DONE?
BGT     10$                 ;BR IF NO
RTS     PC                  ;RETURN

;
; Set WORDS 0-7 of expd message BUFFER = to recv since not testing
;
T20SETEXP:
MOV     #T20EXP,R2          ;GET EXPD
MOV     #T20BFR,R3          ;GET READ STATUS RECV BUFFER
MOV     #8,R0               ;SET WORDS 0-7 EXP=RECV
5$:     MOV    (R3)+,(R2)+   ;SET EXPD=RECV
DEC     R0                  ;DONE WORDS 0-7 WORDS?
BGT     5$                  ;BR IF NO
RTS     PC                  ;RETURN

;
; WRITE CHARACTERISTICS COMMAND PACKET
;
T20PACKET:
;COMMAND PACKET FOR TEST
;WRITE CHARACTERISTICS COMMAND. WITH ACK
;ADDRESS OF CHARACTERISTICS BLOCK
;MINIMUM MESSAGE PACKET SIZE
WORD    100004
WORD    T20DATA
WORD    0
WORD    10.

T20DATA:
;CHARACTERISTICS DATA BLOCK
;ADDRESS OF MESSAGE BUFFER
WORD    T20BFR
WORD    0
WORD    20.
WORD    0
WORD    7
;LENGTH OF MESSAGE BUFFER
;ESS,END,EAI,ERI
;EXTENDED FEATURES UNIT NO.

;MESSAGE BUFFER FOR ALL TEST 17 COMMANDS

```

J1

TSV5 - HARDWARE TESTS MACRO M11.3 06-FEB-84 17:14  
TEST 9: READ/WRITE DATA PARITY TEST

SEQ 216

5785 066422  
5786 066422 000000  
5787 066424 000000  
5788 066426 000000  
5789 066430 000000  
5790 066432 000000  
5791 066434 000000  
5792 066436 000000  
5793 066440 000000  
5794 066442  
5795 066542  
5796  
5797  
5798  
5800 066550 066550  
5802 066550  
5803 066550 100006  
5804 066552 066560  
5805 066554 000000  
5806 066556 000012  
5807  
5808 066560  
5809 066560 000  
5810 066561 000  
5811 066562 000000  
5812 066564  
5813  
5814  
5815 066664  
066664  
066664 1C4401

T20BFR: ;BEGIN MESSAGE BUFFER  
;MESSAGE TYPE  
;DATA FIELD LENGTH  
;RBPCR  
;XST0  
;XST1  
;XST2  
;XST3  
;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM  
;READ STATUS AND WRITE FIFO BUFFER  
;END OF MESSAGE BUFFER  
T20BFSTA: .BLKB 64.  
T20BEND:  
;WRITE SUBSYSTEM READ STATUS COMMAND PACKET  
;.  
;.=<..10>E177770  
T20PK2: ;WRITE SUBSYSTEM WITH ACK  
;LOW ADDRESS OF DATA BLOCK  
;HIGH ADDRESS OF DATA BLOCK  
;MINIMUM MESSAGE PACKET SIZE  
T20DT2: ;DATA BLOCK  
;BSEL0  
;BSEL1  
;SEL2  
;WRITE FIFO DATA OUTPUT BUFFER  
ENDTST

L10073: TRAP C\$ETST

.SBTTL TEST 10: MANUAL INTERVENTION

;THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A "TEST")  
;THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF  
;THE SUBSYSTEM THAT CANNOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN  
;THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE  
;SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE.  
;THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR  
;ARE BY TYPING <CTRL-C> OR SELECTING CODE 7.  
;SELECTION CODES AND SUBROUTINES ARE:

| CODE | ROUTINE                            |
|------|------------------------------------|
| 0    | HELP. PRINTS THIS MENU.            |
| 1    | TURN ON ALL M7196 LED INDICATORS   |
| 2    | TURN OFF ALL M7196 LED INDICATORS  |
| 3    | OFFLINE/ONLINE ATTENTION TEST      |
| 4    | WRITE-PROTECT TEST                 |
| 5    | INITIATE TRANSPORT SERVO EXERCISER |
| 6    | PRINT EXTENDED TRANSPORT STATUS    |
| 7    | EXIT (RETURN TO SUPERVISOR)        |

;EACH MENU ITEM CORRESPONDS TO A SUBTEST, AS FOLLOWS:

```

5842      ;
5843      ;
5844      ;PRINTS OUT THE MENU ON THE CONSOLE TERMINAL.
5845      ;
5846      ;
5847      ;CAUSES ALL THREE LED INDICATORS ON THE M7196 MODULE
5848      ;TO BE ILLUMINATED. AFTER INITIATING THIS ROUTINE, THE OPERATOR
5849      ;SHOULD OBSERVE THE LED'S AND VERIFY THAT THEY ARE INDEED ALL LIT.
5850      ;THIS ROUTINE FIRST USES THE WRITE SUBSYSTEM MEMORY COMMAND TO
5851      ;SET THE FORCE WRONG PARITY FLIP-FLOP, WHICH SERVES TO DRIVE THE
5852      ;"PROCESSOR NOT OK" LED. THEN IT ENTERS A LOOP THAT CONTINUALLY
5853      ;WRITES THE LOW BYTE OF TSOB AND READS THE TSSR. THESE LATTER TWO
5854      ;OPERATIONS WILL CAUSE THE "NOT SSR" AND "DRIVING BUS" LED'S TO
5855      ;GLOW -- THEY ARE NOT REALLY LIT AT ALL TIMES BUT SHOULD APPEAR
5856      ;REASONABLY VISIBLE.
5857      ;
5858      ;
5859      ;INITIALIZES THE CONTROLLER TO CAUSE ALL LED'S TO
5860      ;EXTINGUISH.
5861      ;
5862      ;
5863      ;
5864      ;
5865      ;THIS ROUTINE INITIALIZES THE CONTROLLER, ISSUES A
5866      ;WRITE CHARACTERISTICS COMMAND TO ENABLE ATTENTION INTERRUPTS,
5867      ;ISSUES A MESSAGE BUFFER RELEASE COMMAND, PRINTS A MESSAGE ON THE
5868      ;CONSOLE TERMINAL INSTRUCTING THE OPERATOR TO TOGGLE THE ON-LINE
5869      ;SWITCH ON THE TRANSPORT, THEN WAITS FOR AN ATTENTION INTERRUPT.
5870      ;EACH TIME THE TRANSPORT TRANSITIONS FROM ON-LINE TO OFF-LINE OR
5871      ;VICE-VERSA, AN ATTENTION INTERRUPT SHOULD BE GENERATED. THE PROGRAM
5872      ;WILL REPORT THE INTERRUPT AND THE CURRENT STATE OF THE TRANSPORT.
5873      ;THE OPERATOR SHOULD VERIFY THAT THE REPORTED STATE MATCHES THE
5874      ;STATE INDICATED BY THE LED ON THE FRONT PANEL OF THE TRANSPORT.
5875      ;IN ADDITION, WHEN THE TRANSPORT IS PLACED OFF-LINE, THE PROGRAM
5876      ;ISSUES A SEQUENCE OF TAPE-MOTION COMMANDS (READ, WRITE, POSITION, ETC.)
5877      ;AND VERIFIES THAT, FOR EACH COMMAND, FUNCTION REJECT TERMINATION
5878      ;RESULTS, ALONG WITH THE NON-EXECUTABLE FUNCTION (NEF) ERROR BIT BEING
5879      ;SET.
5880      ;
5881      ;THIS ROUTINE INSTRUCTS THE OPERATOR TO MOUNT A SCRATCH
5882      ;TAPE REEL THAT DOES NOT HAVE A WRITE-ENABLE RING INSTALLED, THEN
5883      ;WAITS FOR THE OPERATOR TO RESPOND THAT THIS HAS BEEN ACCOMPLISHED.
5884      ;UPON THE RESPONSE, THE PROGRAM VERIFIES THAT THE TRANSPORT SHOWS
5885      ;A WRITE-PROTECTED STATUS, THEN ATTEMPTS TO WRITE DATA ON THE
5886      ;TAPE AND EXPECTS THE APPROPRIATE ERROR TERMINATION INDICATING THAT
5887      ;THE WRITE FUNCTION COULD NOT BE PERFORMED BECAUSE THE REEL IS
5888      ;WRITE-PROTECTED. IF THE APPROPRIATE TERMINATION IS NOT RECEIVED,
5889      ;AN ERROR IS REPORTED.
5890      ;
5891      ;
5892      ;
5893      ;INSTRUCTS THE OPERATOR TO PLACE THE TAPE TRANSPORT(S)
5894      ;ON-LINE (IF ANY ARE OFF-LINE) THEN ATTEMPTS TO PERFORM AN EXTENDED
5895      ;STATUS READOUT. FOR EACH TRANSPORT EQUIPPED WITH THIS FEATURE,
5896      ;THE PROGRAM FORMATS AND PRINTS OUT THE RESULTING STATUS. IF THE
5897      ;TRANSPORT IS NOT EQUIPPED WITH THIS FEATURE, A MESSAGE INDICATING
5898      ;SUCH IS ISSUED.

```

TSV5 - HARL ARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 218

| Address | Hex | Dec | Label | Op | Op1 | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416</ |
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|

| Address | Offset | PC | PC+1 | PC+2 | PC+3 | PC+4 | PC+5 | PC+6 | PC+7 | PC+8 | PC+9 | PC+10 | PC+11 | PC+12 | PC+13 | PC+14 | PC+15 | PC+16 | PC+17 | PC+18 | PC+19 | PC+20 | PC+21 | PC+22 | PC+23 | PC+24 | PC+25 | PC+26 | PC+27 | PC+28 | PC+29 | PC+30 | PC+31 | PC+32 | PC+33 | PC+34 | PC+35 | PC+36 | PC+37 | PC+38 | PC+39 | PC+40 | PC+41 | PC+42 | PC+43 | PC+44 | PC+45 | PC+46 | PC+47 | PC+48 | PC+49 | PC+50 | PC+51 | PC+52 | PC+53 | PC+54 | PC+55 | PC+56 | PC+57 | PC+58 | PC+59 | PC+60 | PC+61 | PC+62 | PC+63 | PC+64 | PC+65 | PC+66 | PC+67 | PC+68 | PC+69 | PC+70 | PC+71 | PC+72 | PC+73 | PC+74 | PC+75 | PC+76 | PC+77 | PC+78 | PC+79 | PC+80 | PC+81 | PC+82 | PC+83 | PC+84 | PC+85 | PC+86 | PC+87 | PC+88 | PC+89 | PC+90 | PC+91 | PC+92 | PC+93 | PC+94 | PC+95 | PC+96 | PC+97 | PC+98 | PC+99 | PC+100 | PC+101 | PC+102 | PC+103 | PC+104 | PC+105 | PC+106 | PC+107 | PC+108 | PC+109 | PC+110 | PC+111 | PC+112 | PC+113 | PC+114 | PC+115 | PC+116 | PC+117 | PC+118 | PC+119 | PC+120 | PC+121 | PC+122 | PC+123 | PC+124 | PC+125 | PC+126 | PC+127 | PC+128 | PC+129 | PC+130 | PC+131 | PC+132 | PC+133 | PC+134 | PC+135 | PC+136 | PC+137 | PC+138 | PC+139 | PC+140 | PC+141 | PC+142 | PC+143 | PC+144 | PC+145 | PC+146 | PC+147 | PC+148 | PC+149 | PC+150 | PC+151 | PC+152 | PC+153 | PC+154 | PC+155 | PC+156 | PC+157 | PC+158 | PC+159 | PC+160 | PC+161 | PC+162 | PC+163 | PC+164 | PC+165 | PC+166 | PC+167 | PC+168 | PC+169 | PC+170 | PC+171 | PC+172 | PC+173 | PC+174 | PC+175 | PC+176 | PC+177 | PC+178 | PC+179 | PC+180 | PC+181 | PC+182 | PC+183 | PC+184 | PC+185 | PC+186 | PC+187 | PC+188 | PC+189 | PC+190 | PC+191 | PC+192 | PC+193 | PC+194 | PC+195 | PC+196 | PC+197 | PC+198 | PC+199 | PC+200 | PC+201 | PC+202 | PC+203 | PC+204 | PC+205 | PC+206 | PC+207 | PC+208 | PC+209 | PC+210 | PC+211 | PC+212 | PC+213 | PC+214 | PC+215 | PC+216 | PC+217 | PC+218 | PC+219 | PC+220 | PC+221 | PC+222 | PC+223 | PC+224 | PC+225 | PC+226 | PC+227 | PC+228 | PC+229 | PC+230 | PC+231 | PC+232 | PC+233 | PC+234 | PC+235 | PC+236 | PC+237 | PC+238 | PC+239 | PC+240 | PC+241 | PC+242 | PC+243 | PC+244 | PC+245 | PC+246 | PC+247 | PC+248 | PC+249 | PC+250 | PC+251 | PC+252 | PC+253 | PC+254 | PC+255 | PC+256 | PC+257 | PC+258 | PC+259 | PC+260 | PC+261 | PC+262 | PC+263 | PC+264 | PC+265 | PC+266 | PC+267 | PC+268 | PC+269 | PC+270 | PC+271 | PC+272 | PC+273 | PC+274 | PC+275 | PC+276 | PC+277 | PC+278 | PC+279 | PC+280 | PC+281 | PC+282 | PC+283 | PC+284 | PC+285 | PC+286 | PC+287 | PC+288 | PC+289 | PC+290 | PC+291 | PC+292 | PC+293 | PC+294 | PC+295 | PC+296 | PC+297 | PC+298 | PC+299 | PC+300 | PC+301 | PC+302 | PC+303 | PC+304 | PC+305 | PC+306 | PC+307 | PC+308 | PC+309 | PC+310 | PC+311 | PC+312 | PC+313 | PC+314 | PC+315 | PC+316 | PC+317 | PC+318 | PC+319 | PC+320 | PC+321 | PC+322 | PC+323 | PC+324 | PC+325 | PC+326 | PC+327 | PC+328 | PC+329 | PC+330 | PC+331 | PC+332 | PC+333 | PC+334 | PC+335 | PC+336 | PC+337 | PC+338 | PC+339 | PC+340 | PC+341 | PC+342 | PC+343 | PC+344 | PC+345 | PC+346 | PC+347 | PC+348 | PC+349 | PC+350 | PC+351 | PC+352 | PC+353 | PC+354 | PC+355 | PC+356 | PC+357 | PC+358 | PC+359 | PC+360 | PC+361 | PC+362 | PC+363 | PC+364 | PC+365 | PC+366 | PC+367 | PC+368 | PC+369 | PC+370 | PC+371 | PC+372 | PC+373 | PC+374 | PC+375 | PC+376 | PC+377 | PC+378 | PC+379 |  |
|---------|--------|----|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|
|---------|--------|----|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|

N1

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 220

```

5993 067314 012701 177777      MOV      #1,R1      ;DATA TO WRITE TO TSDB
5994 067320 00C240      12$: NOP      ;ALLOW OPERATOR TO TYPE +C
5995 067322 012702 001750      MOV      #1000.,R2    ;SET-UP INNER LOOP
5996 067326 110165 000000      14$: MOVB    R1,TSDB(R5)    ;WRITE DATA TO TSDB
5997 067332 016500 000002      MOV      TSSR(R5),R0    ;READ TSSR
5998 067336 005302      DEC      R2      ;REDUCE INNER COUNT
5999 067340 001372      BNE     14$      ;LOOP TILL EXPIRES
6000 067342 000766      BR      12$      ;LOOP UNTIL HALTED
6001
6002 067344      15$: PRINTF  #T38MS2      ;TYPE CNTL C TO EXIT
      067344 012746 073261      MOV      #T38MS2,-(SP)
      057350 012746 000001      MOV      #1,-(SP)
      067354 010600      MOV      SP,R0
      067356 104417      TRAP    C$PNTF
      067360 062706 000004      ADD     #4,SP
6003 067364 004737 015774      JSR      PC,SOFINIT    ;DO SOFT INIT OF CONTROLLER
6004 057370 103405      BCS     200$      ;BR IF SOFT INIT = OK
6008 067372 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
6009 067374      ERDIF   ERRNO,SFIERR,SFIMSG ;DEVICE FATAL ERROR DURING INIT
      067374 104455      TRAP    C$ERDF
      067376 001755      .WORD   1005
      067400 003652      .WORD   SFIERR
      067402 012034      .WORD   SFIMSG
6010 067404
6011 057404 013737 002202 072210 200$: MOV      UNITN,T38DSW    ;SET UNIT NUMBER
6012 067412 012704 072170      MOV      #T38PK2,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
6013 067416 004737 010662      JSR      PC,WRTCHR    ;ISSUE WRITE CHARACTERISTICS
6014 067422 103405      BCS     210$      ;BR, IF COMMAND ISSUED OK
6018 067424 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
6019 067426      ERHRD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
      067426 104456      TRAP    C$ERHRD
      067430 001756      .WORD   1006
      067432 005056      .WORD   WRTMSG
      067434 012034      .WORD   SFIMSG
6020
6021
6022
6023
6024
6025 067436      :*****
      :
      :   THIS WRITE SUB-SYSTEM MEMORY COMMAND JUST HOLDS THE LEDS OFF
      :
      :*****
6026 067436 112737 000000 071501 210$: MOVB    #0,T38BS1    ;CLEAR BIT #4
6027 067444 112737 000025 071500      MOVB    #25,T38BS0    ;STOP DRIVE TEST 22
6028 067452 012704 071470      MOV      #T38PACKET,R4    ;SET UP NEW WRT. SUBSYS MEM. COMMAND
6029 067456 010465 000000      MOV      R4,TSDB(R5)    ;SET THE PACKET ADDRESS
6030 067462 004737 016336      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
6031 067466 103405      BCS     250$      ;BR IF CARRY SET (GOOD RETURN)
6032 067470 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
6036 067472      ERDIF   ERRNO,T38SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      067472 104455      TRAP    C$ERDF
      067474 001757      .WORD   1007
      067476 072666      .WORD   T38SSR
      067500 012046      .WORD   PKTSSR
6037 067502      250$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      067502 104406      TRAP    C$ERRI
6038 067504      SETPRI  #PR107      ;RAISE THE PRIORITY
      067504 012700 000340      MOV      #PR107,R0
      067510 104441      TRAP    C$SPRI

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 221

```

6039 067512 005037 071454      CLR      TTION2
6040 067516 032737 000100 177560  BIT      @100,@TTICSR
6041 067524 001005              BNE      710$
6042 067526 005237 071454      INC      TTION2
6043 067532 052737 000100 177560  BIS      @100,@TTICSR
6044 067540 012701 000060 710$:  MOV      @TTIVEC,R1
6045 067544 011137 071456      MOV      (R1),IVSAV2
6046 067550 012721 070750      MOV      @590$,(R1)
6047 067554 011137 071460      MOV      (R1),TPSAV2
6048 067560 012711 000340      MOV      @PRI07,(R1)
6049 067564              SETPRI  @PRI00
        067564 012700 000000
        067570 104441
6050 067572 000240      260$:  NOP
6051 067574 000776      BR      260$
6052
6053
6054              20$:  PRINTF  @T38MS2
        067576 012746 073261
        067602 012746 000001
        067606 010600
        067610 104417
        067612 062706 000004
6055 067616              SETPRI  @PRI00
        067616 012700 000000
        067622 104441
6056 067624 005037 002224      CLR      INTRECV
6057 067630 004737 015774      JSR      PC,SOFINIT
6058 067634 103405      BCS      300$
6062 067636 010001      MOV      R0,R1
6063 067640      ERRDF  ERRNO,SFIERR,SFIMSG
        067640 104455
        067642 001760
        067644 003652
        067646 012034
6064 067650      300$:  MOV      UNITN,T38DSW
6065 067650 013737 002202 072210  MOV      @BIT5,T38EA1
6066 067656 012737 000040 072206  MOV      @T38PK2,R4
6067 067664 012704 072170      MOV      PC,WRTCHR
6068 067670 004737 010662      BCS      310$
6069 067674 103405      MOV      R0,R1
6073 067676 010001      ERRHRD  ERRNO,WRTMSG,SFIMSG
6074 067700
        067700 104456
        067702 001761
        067704 005056
        067706 012034
6075 067710      310$:  MOV      @T38PK3,R4
6076 067710 012704 072220      MOV      R4,T50B(R5)
6077 067714 010455 0000C0      JSR      PC,WAITF
6078 067720 004737 016250      CLR      R2
6079 067724 005002      MOV      TSSR(R5),R1
6080 067726 016501 000002      BIT      @OFL,R1
6081 067732 032701 000100      BEQ      320$
6082 067736 001402      BIS      @OFL,R2
6083 067740 052702 000100      BIS      @SSR,R2
6084 067744 052702 000200

```

```

;ASSUME INTERRUPTS ARE ENABLED
;ARE TTI INTERRUPTS ON ?
;BRANCH IF YES
;FLAG SET IF INTERRUPTS OFF
;ENABLE INTERRUPTS
;START OF TTI VECTORS
;SAVE THE CURRENT TTI VECTOR
;SET NEW INTERRUPT ROUTINE
;SAVE THE VECTOR PRIORITY
;USE PRIORITY SEVEN
;LOWER INTERRUPT BR LEVEL
        MOV      @PRI00,R0
        TRAP
;ALLOW CNTL C
;LOOP UNTIL STOPPED
;TELL'EM WHAT TO TYPE
        MOV      @T38MS2,-(SP)
        MOV      @1,-(SP)
        MOV      SP,R0
        TRAP      C$PNTF
        ADD      @4,SP
;LOWER PRIORITY TO ALLOW INTERRUPTS
        MOV      @PRI00,R0
        TRAP      C$SPRI
;CLEAR INTERRUPT RECEIVED FLAG
;DO SOFT INIT OF CONTROLLER
;BR IF SOFT INIT = OK
;SAVE CONTENTS OF TSSR
;DEVICE FATAL ERROR DURING INIT
        TRAP      C$ERDF
        .WORD      1008
        .WORD      SFIERR
        .WORD      SFIMSG
;SET UNIT NUMBER IN PACKET
;ENABLE ATTENTION INTERRUPTS
;SUBROUTINE NEEDS PACKET ADDRESS
;ISSUE WRITE CHARACTERISTICS
;BR, IF COMMAND ISSUED OK
;SAVE CONTENTS OF TSSR
;WRITE CHARACTERISTICS FAILED
        TRAP      C$ERHRD
        .WORD      1009
        .WORD      WRTMSG
        .WORD      SFIMSG
;SET UP NEW PACKET FOR MESS BUF REL
;MESSAGE BUFFER RELEASE.ACK,CVC#1 CMD
;WAIT FOR SSR TO SET
;MAKE SURE ALL IS CLEAR
;GET TSSR STATUS
;IS OFL SET
;BR, IF OFL IS NOT SET
;SET OFL IN EXPECTED
;SET UP EXPECTED

```

C2

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 222

|      |        |        |        |        |                     |        |                                |       |               |
|------|--------|--------|--------|--------|---------------------|--------|--------------------------------|-------|---------------|
| 6085 | 067750 | 020201 |        | CMP    | R2,R1               |        | IS EVERYTHING OK               |       |               |
| 6086 | 067752 | 001404 |        | BEG    | 350\$               |        | BR, IF ALL IS WFL.             |       |               |
| 6090 | 067754 |        |        | ERRHRD | ERRNO,T38SST,PKTSSR |        | DEVICE FATAL SSR FAILED TO SET |       |               |
|      | 067754 | 104456 |        |        |                     |        |                                | TRAP  | C\$ERHRD      |
|      | 067756 | 071762 |        |        |                     |        |                                | .WORD | 1010          |
|      | 067760 | 073076 |        |        |                     |        |                                | .WORD | T38SST        |
|      | 067762 | 012046 |        |        |                     |        |                                | .WORD | PKTSSR        |
| 6091 | 067764 |        | 350\$: | CKLOOP |                     |        | LOOP ON ERROR, IF FLAG SET     |       |               |
|      | 067764 | 104406 |        |        |                     |        |                                | TRAP  | C\$CLP1       |
| 6092 | 067766 |        |        | PRINTF | @T38MS1             |        | TELL OPERATOR TO TOGGLE SWITCH |       |               |
|      | 067766 | 012746 | 073166 |        |                     |        |                                | MOV   | @T38MS1,-(SP) |
|      | 067772 | 012746 | 000001 |        |                     |        |                                | MOV   | @1,-(SP)      |
|      | 067776 | 010600 |        |        |                     |        |                                | MOV   | SP,RO         |
|      | 070000 | 104417 |        |        |                     |        |                                | TRAP  | C\$PNTF       |
|      | 070002 | 062706 | 000004 |        |                     |        |                                | ADD   | @4,SP         |
| 6093 | 070006 |        |        | PRINTF | @T38MS2             |        | TELL OPERATOR TO DO TC TO EXIT |       |               |
|      | 070006 | 012746 | 073261 |        |                     |        |                                | MOV   | @T38MS2,-(SP) |
|      | 070012 | 012746 | 000001 |        |                     |        |                                | MOV   | @1,-(SP)      |
|      | 070016 | 010600 |        |        |                     |        |                                | MOV   | SP,RO         |
|      | 070020 | 104417 |        |        |                     |        |                                | TRAP  | C\$PNTF       |
|      | 070022 | 062706 | 000004 |        |                     |        |                                | ADD   | @4,SP         |
| 6094 | 070026 |        |        | SETPRI | @PRI07              |        | RAISE THE PRIORITY             |       |               |
|      | 070026 | 012700 | 000340 |        |                     |        |                                | MOV   | @PRI07,RO     |
|      | 070032 | 104441 |        |        |                     |        |                                | TRAP  | C\$SPRI       |
| 6095 | 070034 | 005037 | 071454 | CLR    | TTION2              |        | ASSUME INTERRUPTS ARE ENABLED  |       |               |
| 6096 | 070040 | 032737 | 000100 | BIT    | @100,@TTICSR        | 177560 | ARE TTI INTERRUPTS ON ?        |       |               |
| 6097 | 070046 | 001005 |        | BNE    | 720\$               |        | BRANCH IF YES                  |       |               |
| 6098 | 070050 | 005237 | 071454 | INC    | TTION2              |        | FLAG SET IF INTERRUPTS OFF     |       |               |
| 6099 | 070054 | 052737 | 000100 | BIS    | @100,@TTICSR        | 177560 | ENABLE INTERRUPTS              |       |               |
| 6100 | 070062 | 012701 | 000060 | MOV    | @TTIVEC,R1          | 720\$: | START OF TTI VECTORS           |       |               |
| 6101 | 070066 | 011137 | 071456 | MOV    | (R1),TVSAV2         |        | SAVE THE CURRENT TTI VECTOR    |       |               |
| 6102 | 070072 | 012721 | 070750 | MOV    | @590,(R1)           |        | SET NEW INTERRUPT ROUTINE      |       |               |
| 6103 | 070076 | 011137 | 071460 | MOV    | (R1),TPSAV2         |        | SAVE THE VECTOR PRIORITY       |       |               |
| 6104 | 070102 | 012711 | 000340 | MOV    | @PRI07,(R1)         |        | USE PRIORITY SEVEN             |       |               |
| 6105 | 070106 |        |        | SETPRI | @PRI00              |        | LOWER INTERRUPT BR LEVEL       |       |               |
|      | 070106 | 012700 | 000000 |        |                     |        |                                | MOV   | @PRI00,RO     |
|      | 070112 | 104441 |        |        |                     |        |                                | TRAP  | C\$SPRI       |
| 6106 | 070114 | 000240 |        | NOP    |                     | 360\$: | ALLOW CONTROL C                |       |               |
| 6107 | 070116 | 005737 | 002224 | TST    | INTRECV             |        | DID AN INTERRUPT OCCUR ?       |       |               |
| 6108 | 070122 | 001001 |        | BNE    | 370\$               |        | BRANCH IF YES                  |       |               |
| 6109 | 070124 | 000773 |        | BR     | 360\$               |        | WAIT SOME MORE FOR INTERRUPT   |       |               |
| 6110 | 070126 |        |        | PRINTF | @T38INT             | 370\$: | "INTERRUPT RECEIVED"           |       |               |
|      | 070126 | 012746 | 072756 |        |                     |        |                                | MOV   | @T38INT,-(SP) |
|      | 070132 | 012746 | 000001 |        |                     |        |                                | MOV   | @1,-(SP)      |
|      | 070136 | 010600 |        |        |                     |        |                                | MOV   | SP,RO         |
|      | 070140 | 104417 |        |        |                     |        |                                | TRAP  | C\$PNYF       |
|      | 070142 | 062706 | 000004 |        |                     |        |                                | ADD   | @4,SP         |
| 6111 | 070146 | 016501 | 000002 | MOV    | TSSR(R5),R1         |        | READ TSSR STATUS               |       |               |
| 6112 | 070152 | 032701 | 000100 | BIT    | @OFL,R1             |        | CHECK THE OFF-LINE BIT         |       |               |
| 6113 | 070156 | 001011 |        | BNE    | 380\$               |        | BR, IF DRIVE IS OFF-LINE       |       |               |
| 6114 | 070160 |        |        | PRINTF | @T38ONL             |        | "DRIVE IS NOW ON-LINE"         |       |               |
|      | 070160 | 012746 | 073006 |        |                     |        |                                | MOV   | @T38ONL,-(SP) |
|      | 070164 | 012746 | 000001 |        |                     |        |                                | MOV   | @1,-(SP)      |
|      | 070170 | 010600 |        |        |                     |        |                                | MOV   | SP,RO         |
|      | 070172 | 104417 |        |        |                     |        |                                | TRAP  | C\$PNTF       |
|      | 070174 | 062706 | 000004 |        |                     |        |                                | ADD   | @4,SP         |
| 6115 | 070200 | 000410 |        | BR     | 390\$               |        | ALMOST DONE                    |       |               |

D2

TSV5 : HARDWARE IESIS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 223

|      |        |        |        |        |            |                        |  |                                   |          |                |
|------|--------|--------|--------|--------|------------|------------------------|--|-----------------------------------|----------|----------------|
| 6116 | 070202 |        |        | 380\$: | PRINTF     | 0T38OFL                |  | "DRIVE IS NOW OFF-LINE"           |          |                |
|      | 070202 | 012746 | 073042 |        |            |                        |  |                                   | MOV      | 0T38OFL, -(SP) |
|      | 070206 | 012746 | 000001 |        |            |                        |  |                                   | MOV      | 01, -(SP)      |
|      | 070212 | 010600 |        |        |            |                        |  |                                   | MOV      | SP, R0         |
|      | 070214 | 104417 |        |        |            |                        |  |                                   | TRAP     | C\$PNTF        |
|      | 070216 | 062706 | 000004 |        |            |                        |  |                                   | ADD      | 04, SP         |
| 6117 | 070222 | 005037 | 002224 | 390\$: | CLR        | INTRECV                |  | CLEAR INTERRUPT FLAG              |          |                |
| 6118 | 070226 | 000137 | 067650 |        | JMP        | 300\$                  |  | TRY AGAIN                         |          |                |
| 6119 | 070232 |        |        | 25\$:  | GMANIL     | T38MSG, T38DAT, -1, NO |  | WAIT FOR OPERATOR TO MOUNT TAPE   |          |                |
|      | 070232 | 104443 |        |        |            |                        |  |                                   | TRAP     | C\$GMAN        |
|      | 070234 | 000404 |        |        |            |                        |  |                                   | BR       | 10000\$        |
|      | 070236 | 074014 |        |        |            |                        |  |                                   | .WORD    | T38DAT         |
|      | 070240 | 000120 |        |        |            |                        |  |                                   | .WORD    | T\$CODE        |
|      | 070242 | 073325 |        |        |            |                        |  |                                   | .WORD    | T38MSG         |
|      | 070244 | 177777 |        |        |            |                        |  |                                   | .WORD    | -1             |
|      | 070246 |        |        |        |            |                        |  |                                   |          |                |
| 6120 | 070246 |        |        |        | BNCOMPLETE | 25\$                   |  | RETRY IF ERROR                    | 10000\$: |                |
|      | 070246 | 103371 |        |        |            |                        |  |                                   |          |                |
| 6121 | 070250 | 005737 | 074014 |        | TST        | T38DAT                 |  | DID OPERATOR SAY 'YES'            | BCC      | 25\$           |
| 6122 | 070254 | 001002 |        |        | BNE        | 27\$                   |  | BRANCH IF YES                     |          |                |
| 6123 | 070256 | 000137 | 066722 |        | JMP        | 2\$                    |  | RETURN TO MAIN MENU               |          |                |
| 6124 | 070262 |        |        | 27\$:  |            |                        |  |                                   |          |                |
| 6125 | 070262 | 004737 | 015774 |        | JSR        | PC, SOFINIT            |  | DO SOFT INIT OF CONTROLLER        |          |                |
| 6126 | 070266 | 103405 |        |        | BCS        | 400\$                  |  | BR IF SOFT INIT - OK              |          |                |
| 6130 | 070270 | 010001 |        |        | MOV        | R0, R1                 |  | SAVE CONTENTS OF TSSR             |          |                |
| 6131 | 070272 |        |        |        | ERRDF      | ERRNO, SFIERR, SFIMSG  |  | DEVICE FATAL ERROR DURING INIT    |          |                |
|      | 070272 | 104455 |        |        |            |                        |  |                                   | TRAP     | C\$ERDF        |
|      | 070274 | 001763 |        |        |            |                        |  |                                   | .WORD    | 1011           |
|      | 070276 | 003652 |        |        |            |                        |  |                                   | .WORD    | SFIERR         |
|      | 070300 | 012034 |        |        |            |                        |  |                                   | .WORD    | SFIMSG         |
| 6132 | 070302 |        |        | 400\$: | CKLOOP     |                        |  | LOOP IF SELECTED                  |          |                |
|      | 070302 | 104406 |        |        |            |                        |  |                                   | TRAP     | C\$CLP1        |
| 6133 | 070304 | 013737 | 002202 | 072210 | MOV        | UNITN, T38DSW          |  | SET UNIT NUMBER                   |          |                |
| 6134 | 070312 | 012704 | 072170 |        | MOV        | 0T38PK2, R4            |  | SUBROUTINE NEEDS PACKET ADDRESS   |          |                |
| 6135 | 070316 | 004737 | 010662 |        | JSR        | PC, WRTCHR             |  | ISSUE WRITE CHARACTERISTICS       |          |                |
| 6136 | 070322 | 103405 |        |        | BCS        | 410\$                  |  | BR, IF COMMAND ISSUED OK          |          |                |
| 6140 | 070324 | 010001 |        |        | MOV        | R0, R1                 |  | SAVE CONTENTS OF TSSR             |          |                |
| 6141 | 070326 |        |        |        | ERRHRD     | ERRNO, WRTMSG, SFIMSG  |  | WRITE CHARACTERISTICS FAILED      |          |                |
|      | 070326 | 104456 |        |        |            |                        |  |                                   | TRAP     | C\$ERHRD       |
|      | 070330 | 001764 |        |        |            |                        |  |                                   | .WORD    | 1012           |
|      | 070332 | 005056 |        |        |            |                        |  |                                   | .WORD    | WRTMSG         |
|      | 070334 | 012034 |        |        |            |                        |  |                                   | .WORD    | SFIMSG         |
| 6142 | 070336 |        |        | 410\$: | CKLOOP     |                        |  | LOOP IF SELECTED                  |          |                |
|      | 070336 | 104406 |        |        |            |                        |  |                                   | TRAP     | C\$CLP1        |
| 6143 | 070340 | 013701 | 071514 |        | MOV        | T38BFR+6, R1           |  | PICK UP XSTO CONTENTS             |          |                |
| 6144 | 070344 | 010102 |        |        | MOV        | R1, R2                 |  | SET UP EXPECTED                   |          |                |
| 6145 | 070346 | 052702 | 000004 |        | BIS        | 0BIT2, R2              |  | SET UP THE WRITE LOCKED BIT       |          |                |
| 6146 | 070352 | 020102 |        |        | CMP        | R1, R2                 |  | ARE THEY CORRECT                  |          |                |
| 6147 | 070354 | 001406 |        |        | BEQ        | 430\$                  |  | BR, IF ALL IS WELL (OK)           |          |                |
| 6151 | 070356 |        |        |        | ERRHRD     | ERRNO, T38WRL, EXPREC  |  | WRITE LOCKED BIT IS NOT SET ETC." |          |                |
|      | 070356 | 104456 |        |        |            |                        |  |                                   | TRAP     | C\$ERHRD       |
|      | 070360 | 001765 |        |        |            |                        |  |                                   | .WORD    | 1013           |
|      | 070362 | 072504 |        |        |            |                        |  |                                   | .WORD    | T38WRL         |
|      | 070364 | 015474 |        |        |            |                        |  |                                   | .WORD    | EXPREC         |
| 6152 | 070366 | 005237 | 002222 |        | INC        | FATFLG                 |  | SET FATAL FLAG                    |          |                |
| 6153 | 070372 |        |        | 430\$: | CKLOOP     |                        |  | LOOP IF SELECTED                  |          |                |
|      | 070372 | 104406 |        |        |            |                        |  |                                   | TRAP     | C\$CLP1        |

E2

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 224

|      |        |        |        |        |        |                      |              |                                         |
|------|--------|--------|--------|--------|--------|----------------------|--------------|-----------------------------------------|
| 6154 | 070374 | 005737 | 002222 |        | TST    | FATFLG               |              | ; WAS THE DRIVE NOT WRITE LOCKED        |
| 6155 | 070400 | 001402 |        |        | BEQ    | 435\$                |              | ; BR, IF FLAG NOT SET                   |
| 6156 | 070402 | 000137 | 066722 |        | JMP    | 2\$                  |              | ; RE-WRITE MENU                         |
| 6157 | 070406 | 017737 | 112512 | 072242 | MOV    | 0FREE,T38WR          | 435\$;       | ; SET UP WRITE BUFFER ADDRESS           |
| 6158 | 070414 | 012704 | 072240 |        | MOV    | 0T38PK4,R4           |              | ; GET PACKET ADDRESS                    |
| 6159 | 070420 | 010465 | 000000 |        | MOV    | R4,TSD8(R5)          |              | ; SET THE PACKET ADDRESS                |
| 6160 | 070424 | 004737 | 016250 |        | JSR    | PC,WAITF             |              | ; WAIT FOR SSR TO SET                   |
| 6161 | 070430 | 016501 | 000002 |        | MOV    | TSSR(R5),R1          |              | ; GET TSSR                              |
| 6162 | 070434 | 012702 | 100206 |        | MOV    | 0SC!SSR!BIT1!BIT2,R2 |              | ; SET UP EXPECTED                       |
| 6163 | 070440 | 020102 |        |        | CMP    | R1,R2                |              | ; ARE THEY EQUAL (CORRECT)              |
| 6164 | 070442 | 001404 |        |        | BEQ    | 440\$                |              | ; BR, IF CORRECT STATUS                 |
| 6168 | 070444 |        |        |        | ERRHRD | ERRNO,T38WRT,PKTSSR  |              | ; "TSSR INCORRECT AFTER WRITE COMMAND   |
|      | 070444 | 104456 |        |        |        |                      |              | TRAP C\$ERHRD                           |
|      | 070446 | 001766 |        |        |        |                      |              | .WORD 1014                              |
|      | 070450 | 072420 |        |        |        |                      |              | .WORD T38WRT                            |
|      | 070452 | 012046 |        |        |        |                      |              | .WORD PKTSSR                            |
| 6169 | 070454 |        |        |        | 440\$: | CKLOOP               |              | ; LOOP ON ERROR, IF FLAG SET            |
|      | 070454 | 104406 |        |        |        |                      |              | TRAP C\$CLP1                            |
| 6170 | 070456 | 013701 | 071514 |        | MOV    | T38BFR+6,R1          |              | ; READ XSTO CONTENTS                    |
| 6171 | 070462 | 010102 |        |        | MOV    | R1,R2                |              | ; SET UP EXPECTED                       |
| 6172 | 070464 | 052702 | 004000 |        | BIS    | 0BIT11,R2            |              | ; SET THE WRITE LOCK ERROR BIT (XSTO)   |
| 6173 | 070470 | 020102 |        |        | CMP    | R1,R2                |              | ; WAS THE BIT SET                       |
| 6174 | 070472 | 001404 |        |        | BEQ    | 450\$                |              | ; BR, IF IT WAS (GOOD)                  |
| 6178 | 070474 |        |        |        | ERRHRD | ERRNO,T38WLE,EXPREC  |              | ; "WRITE LOCK ERROR BIT NOT SET"        |
|      | 070474 | 104456 |        |        |        |                      |              | TRAP C\$ERHRD                           |
|      | 070476 | 001767 |        |        |        |                      |              | .WORD 1015                              |
|      | 070500 | 072545 |        |        |        |                      |              | .WORD T38WLE                            |
|      | 070502 | 015474 |        |        |        |                      |              | .WORD EXPREC                            |
| 6179 | 070504 |        |        |        | 450\$: | CKLOOP               |              | ; LOOP IF SELECTED                      |
|      | 070504 | 104406 |        |        |        |                      |              | TRAP C\$CLP1                            |
| 6180 | 070506 | 000137 | 066722 |        | JMP    | 2\$                  |              | ; GO BACK TO MENU                       |
| 6181 |        |        |        |        |        |                      |              |                                         |
| 6182 |        |        |        |        |        |                      |              |                                         |
| 6183 |        |        |        |        |        |                      |              |                                         |
| 6184 |        |        |        |        |        |                      |              |                                         |
| 6185 | 070512 |        |        |        |        |                      |              |                                         |
| 6186 | 070512 |        |        |        | 30\$:  | PRINTB               | 0T38MS3      | ; "EXE ANY OTHER MENU SELECTION TO STOP |
|      | 070512 | 012746 | 072325 |        |        |                      |              | MOV 0T38MS3,-(SP)                       |
|      | 070516 | 012746 | 000001 |        |        |                      |              | MOV 01,-(SP)                            |
|      | 070522 | 010600 |        |        |        |                      |              | MOV SP,R0                               |
|      | 070524 | 104414 |        |        |        |                      |              | TRAP C\$PNTB                            |
|      | 070526 | 062706 | 000004 |        |        |                      |              | ADD 04,SP                               |
| 6187 | 070532 | 004737 | 074016 |        | JSR    | PC,T38REST           |              | ; SET PACKET TO INITIAL VALUES          |
| 6188 | 070536 | 004737 | 015774 |        | JSR    | PC,SOFINIT           |              | ; DO SOFT INIT OF CONTROLLER            |
| 6189 | 070542 | 103405 |        |        | BCS    | 500\$                |              | ; BR IF SOFT INIT - OK                  |
| 6193 | 070544 | 010001 |        |        | MOV    | R0,R1                |              | ; SAVE CONTENTS OF TSSR                 |
| 6194 | 070546 |        |        |        | ERRDF  | ERRNO,SFIERR,SFIMSG  |              | ; DEVICE FATAL ERROR DURING INIT        |
|      | 070546 | 104455 |        |        |        |                      |              | TRAP C\$ERDF                            |
|      | 070550 | 001770 |        |        |        |                      |              | .WORD 1016                              |
|      | 070552 | 003652 |        |        |        |                      |              | .WORD SFIERR                            |
|      | 070554 | 012034 |        |        |        |                      |              | .WORD SFIMSG                            |
| 6195 | 070556 | 013737 | 002202 | 072210 | 500\$: | MOV                  | UNITN,T38DSW | ; SET UNIT NUMBER                       |
| 6196 | 070564 | 012704 | 072170 |        | MOV    | 0T38PK2,R4           |              | ; SUBROUTINE NEEDS PACKET ADDRESS       |
| 6197 | 070570 | 004737 | 010662 |        | JSR    | PC,WRTCHR            |              | ; ISSUE WRITE CHARACTERISTICS           |
| 6198 | 070574 | 103405 |        |        | BCS    | 510\$                |              | ; BR, IF COMMAND ISSUED OK              |
| 6202 | 070576 | 010001 |        |        | MOV    | R0,R1                |              | ; SAVE CONTENTS OF TSSR                 |
| 6203 | 070600 |        |        |        | ERRHRD | ERRNO,WRTMSG,SFIMSG  |              | ; WRITE CHARACTERISTICS FAILED          |

F2

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 225

|             |        |        |        |        |        |                                |       |           |
|-------------|--------|--------|--------|--------|--------|--------------------------------|-------|-----------|
| 070600      | 104456 |        |        |        |        |                                | TRAP  | C\$ERHRD  |
| 070602      | 001771 |        |        |        |        |                                | .WORD | 1017      |
| 070604      | 005056 |        |        |        |        |                                | .WORD | WRTMSG    |
| 070606      | 012034 |        |        |        |        |                                | .WORD | SFIMSG    |
| 6204 070610 |        |        |        | 510\$: | MOVB   | #0,T38BS1                      |       |           |
| 6205 070610 | 112737 | 000000 | 071501 |        | MOVB   | #20,T38BS0                     |       |           |
| 6206 070616 | 112737 | 000020 | 071500 |        | MOV    | #T38PACKET,R4                  |       |           |
| 6207 070524 | 012704 | 071470 |        |        | MOV    | R4,TSD8(R5)                    |       |           |
| 6208 070630 | 010465 | 000000 |        |        | JSR    | PC,CHKTSSR                     |       |           |
| 6209 070634 | 004737 | 016336 |        |        | BCS    | 550\$                          |       |           |
| 6210 070640 | 103405 |        |        |        | MOV    | R0,R1                          |       |           |
| 6211 070642 | 010001 |        |        |        | ERRDF  | ERRNO,T38SSR,PKTSSR            |       |           |
| 6215 070644 |        |        |        |        |        |                                |       |           |
| 070644      | 104455 |        |        |        |        |                                | TRAP  | C\$ERDF   |
| 070646      | 001772 |        |        |        |        |                                | .WORD | 1018      |
| 070650      | 072666 |        |        |        |        |                                | .WORD | T38SSR    |
| 070652      | 012046 |        |        |        |        |                                | .WORD | PKTSSR    |
| 6216 070654 |        |        |        | 550\$: | CKLOOP |                                |       |           |
| 070654      | 104406 |        |        |        | SETPRI | #PRI07                         |       |           |
| 6217 070656 |        |        |        |        |        |                                |       |           |
| 070656      | 012700 | 000340 |        |        |        |                                | MOV   | #PRI07,R0 |
| 070662      | 104441 |        |        |        |        |                                | TRAP  | C\$SPRI   |
| 6218 070664 | 005037 | 071454 |        |        | CLR    | TTION2                         |       |           |
| 6219 070670 | 032737 | 000100 | 177560 |        | BIT    | #100,#TTICSR                   |       |           |
| 6220 070676 | 001005 |        |        |        | BNE    | 555\$                          |       |           |
| 6221 070700 | 005237 | 071454 |        |        | INC    | TTION2                         |       |           |
| 6222 070704 | 052737 | 000100 | 177560 |        | BIS    | #100,#TTICSR                   |       |           |
| 6223 070712 | 012701 | 000060 |        | 555\$: | MOV    | #TTIVEC,R1                     |       |           |
| 6224 070716 | 011137 | 071456 |        |        | MOV    | (R1),TVSAV2                    |       |           |
| 6225 070722 | 012721 | 070750 |        |        | MOV    | #590\$,(R1)                    |       |           |
| 6226 070726 | 011137 | 071460 |        |        | MOV    | (R1),TPSAV2                    |       |           |
| 6227 070732 | 012711 | 000340 |        |        | MOV    | #PRI07,(R1)                    |       |           |
| 6228 070736 |        |        |        |        | SETPRI | #PRI00                         |       |           |
| 070736      | 012700 | 000000 |        |        |        |                                | MOV   | #PRI00,R0 |
| 070742      | 104441 |        |        |        |        |                                | TRAP  | C\$SPRI   |
| 6229 070744 | 000240 |        |        | 560\$: | NOP    |                                |       |           |
| 6230 070746 | 000776 |        |        |        | BR     | 560\$                          |       |           |
| 6231        |        |        |        |        |        |                                |       |           |
| 6232        |        |        |        |        |        |                                |       |           |
| 6233        |        |        |        |        |        |                                |       |           |
| 6234        |        |        |        |        |        |                                |       |           |
| 6235 070750 | 010046 |        |        | 590\$: | MOV    | R0,-(SP)                       |       |           |
| 6236 070752 | 113700 | 177562 |        |        | MOVB   | #TTIBFR,R0                     |       |           |
| 6237 070756 | 042700 | 000200 |        |        | BIC    | #200,R0                        |       |           |
| 6238 070762 | 122700 | 000015 |        |        | CMPS   | #15,R0                         |       |           |
| 6239 070766 | 001075 |        |        |        | BNE    | 591\$                          |       |           |
| 6240 070770 | 012766 | 066722 | 000002 |        | MOV    | #2\$,(SP)                      |       |           |
| 6241 070776 | 005066 | 000004 |        |        | CLR    | 4(SP)                          |       |           |
| 6242 071002 | 013737 | 071456 | 000060 |        | MOV    | TVSAV2,#TTIVEC ;RESTORE VECTOR |       |           |
| 6243 071010 | 013737 | 071460 | 000062 |        | MOV    | TPSAV2,#TTIVEC+2               |       |           |
| 6244 071016 | 112737 | 000025 | 071500 |        | MOVB   | #25,T38BS0                     |       |           |
| 6245 071024 | 112737 | 000000 | 071501 |        | MOVB   | #0,T38BS1                      |       |           |
| 6246 071032 | 012704 | 071470 |        |        | MOV    | #T38PACKET,R4                  |       |           |
| 6247 071036 | 010465 | 000000 |        |        | MOV    | R4,TSD8(R5)                    |       |           |
| 6248 071042 | 012737 | 176750 | 071462 |        | MOV    | #65000,T38DLY                  |       |           |
| 6249 071050 | 004737 | 016250 |        | 592\$: | JSR    | PC,WAITF                       |       |           |
| 6250 071054 | 016501 | 000002 |        |        | MOV    | TSSR(R5),R1                    |       |           |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 226

| Address | Op-Code | Register | Value  | Label  | Comment | Register | Value | Label | Comment |
|---------|---------|----------|--------|--------|---------|----------|-------|-------|---------|
| 6251    | 071060  | 032701   | 000200 |        |         |          |       |       |         |
| 6252    | 071064  | 001017   |        |        |         |          |       |       |         |
| 6253    | 071066  |          |        |        |         |          |       |       |         |
|         | 071066  | 012727   | 000250 |        |         |          |       |       |         |
|         | 071072  | 000000   |        |        |         |          |       |       |         |
|         | 071074  | 013727   | 002116 |        |         |          |       |       |         |
|         | 071100  | 000000   |        |        |         |          |       |       |         |
|         | 071102  | 005367   | 177772 |        |         |          |       |       |         |
|         | 071106  | 001375   |        |        |         |          |       |       |         |
|         | 071110  | 005367   | 177756 |        |         |          |       |       |         |
|         | 071114  | 001367   |        |        |         |          |       |       |         |
| 6254    | 071116  | 005337   | 071462 |        |         |          |       |       |         |
| 6255    | 071122  | 001352   |        |        |         |          |       |       |         |
| 6256    | 071124  | 004737   | 016336 | 595\$: |         |          |       |       |         |
| 6257    | 071130  | 103405   |        |        |         |          |       |       |         |
| 6258    | 071132  | 010C01   |        |        |         |          |       |       |         |
| 6262    | 071134  |          |        |        |         |          |       |       |         |
|         | 071134  | 104455   |        |        |         |          |       |       |         |
|         | 071136  | 001773   |        |        |         |          |       |       |         |
|         | 071140  | 072666   |        |        |         |          |       |       |         |
|         | 071142  | 012046   |        |        |         |          |       |       |         |
| 6263    | 071144  |          |        | 580\$: | CKLOOP  |          |       |       |         |
|         | 071144  | 104406   |        |        |         |          |       |       |         |
| 6264    | 071146  | 005737   | 071454 |        |         |          |       |       |         |
| 6265    | 071152  | 001403   |        |        |         |          |       |       |         |
| 6266    | 071154  | 042737   | 000100 | 177560 |         |          |       |       |         |
| 6267    | 071162  | 012600   |        | 591\$: |         |          |       |       |         |
| 6268    | 071164  | 000002   |        |        |         |          |       |       |         |
| 6269    | 071166  |          |        | 35\$:  |         |          |       |       |         |
| 6270    | 071166  | 004737   | 074016 |        |         |          |       |       |         |
| 6271    | 071172  | 004737   | 015774 |        |         |          |       |       |         |
| 6272    | 071176  | 103405   |        |        |         |          |       |       |         |
| 6276    | 071200  | 010001   |        |        |         |          |       |       |         |
| 6277    | 071202  |          |        |        |         |          |       |       |         |
|         | 071202  | 104455   |        |        |         |          |       |       |         |
|         | 071204  | 001774   |        |        |         |          |       |       |         |
|         | 071206  | 003652   |        |        |         |          |       |       |         |
|         | 071210  | 012034   |        |        |         |          |       |       |         |
| 6278    | 071212  |          |        | 600\$: | CKLOOP  |          |       |       |         |
|         | 071212  | 104406   |        |        |         |          |       |       |         |
| 6279    | 071214  | 012701   | 071506 |        |         |          |       |       |         |
| 6280    | 071220  | 012702   | 125252 |        |         |          |       |       |         |
| 6281    |         |          |        |        |         |          |       |       |         |
| 6282    | 071224  | 010221   |        | 601\$: |         |          |       |       |         |
| 6283    | 071226  | 022701   | 072162 |        |         |          |       |       |         |
| 6284    | 071232  | 0014C1   |        |        |         |          |       |       |         |
| 6285    | 071234  | 000773   |        |        |         |          |       |       |         |
| 6286    | 071236  | 013737   | 002202 | 072210 | 605\$:  |          |       |       |         |
| 6287    | 071244  | 012704   | 072170 |        |         |          |       |       |         |
| 6288    | 071250  | 004737   | 010662 |        |         |          |       |       |         |
| 6289    | 071254  | 103405   |        |        |         |          |       |       |         |
| 6293    | 071256  | 010001   |        |        |         |          |       |       |         |
| 6294    | 071260  |          |        |        |         |          |       |       |         |
|         | 071260  | 104456   |        |        |         |          |       |       |         |
|         | 071262  | 001775   |        |        |         |          |       |       |         |
|         | 071264  | 005056   |        |        |         |          |       |       |         |
|         | 071266  | 012034   |        |        |         |          |       |       |         |

| Address | Offset | Hex    | Dec    | Label  | Operation | Comments |
|---------|--------|--------|--------|--------|-----------|----------|
| 6295    | 071270 | 104406 |        |        |           |          |
| 6296    | 071272 | 112737 | 000000 | 071501 | MOV       | MOV      |
| 6297    | 071300 | 112737 | 000024 | 071500 | MOV       | MOV      |
| 6298    | 071306 | 012704 | 071470 |        | MOV       | MOV      |
| 6299    | 071312 | 010465 | 000000 |        | MOV       | MOV      |
| 6300    | 071316 | 012737 | 000144 | 071462 | MOV       | MOV      |
| 6301    | 071324 | 004737 | 016250 |        | JSR       | JSR      |
| 6302    | 071330 | 016501 | 000002 |        | MOV       | MOV      |
| 6303    | 071334 | 032701 | 000200 |        | BIT       | BIT      |
| 6304    | 071340 | 001017 |        |        | BNE       | BNE      |
| 6305    | 071342 |        |        |        | DELAY     | DELAY    |
|         | 071342 | 012727 | 000250 |        |           |          |
|         | 071346 | 000000 |        |        |           |          |
|         | 071350 | 013727 | 002116 |        |           |          |
|         | 071354 | 000000 |        |        |           |          |
|         | 071356 | 005367 | 177772 |        |           |          |
|         | 071362 | 001375 |        |        |           |          |
|         | 071364 | 005367 | 177756 |        |           |          |
|         | 071370 | 001367 |        |        |           |          |
| 6306    | 071372 | 005337 | 071462 |        | DEC       | DEC      |
| 6307    | 071376 | 001352 |        |        | BNE       | BNE      |
| 6308    | 071400 | 004737 | 016336 |        | JSR       | JSR      |
| 6309    | 071404 | 103405 |        |        | BCS       | BCS      |
| 6310    | 071406 | 010001 |        |        | MOV       | MOV      |
| 6314    | 071410 |        |        |        | ERRDF     | ERRDF    |
|         | 071410 | 104455 |        |        |           |          |
|         | 071412 | 001776 |        |        |           |          |
|         | 071414 | 072666 |        |        |           |          |
|         | 071416 | 012046 |        |        |           |          |
| 6315    | 071420 |        |        |        |           |          |
|         | 071420 | 104406 |        |        |           |          |
| 6316    | 071422 | 012700 | 071526 |        | MOV       | MOV      |
| 6317    | 071426 | 005001 |        |        | CLR       | CLR      |
| 6318    | 071430 | 005037 | 003134 |        | CLR       | CLR      |
| 6319    | 071434 | 004737 | 074054 |        | JSR       | JSR      |
| 6320    | 071440 | 000137 | 066722 |        | JMP       | JMP      |
| 6321    |        |        |        |        |           |          |
| 6322    |        |        |        |        |           |          |
| 6323    | 071444 | 000137 | 000200 |        | JMP       | JMP      |
| 6324    | 071450 |        |        |        |           |          |
|         | 071450 | 104432 |        |        |           |          |
|         | 071452 | 003054 |        |        |           |          |
| 6325    |        |        |        |        |           |          |
| 6326    |        |        |        |        |           |          |
| 6327    |        |        |        |        |           |          |
| 6328    |        |        |        |        |           |          |
| 6329    |        |        |        |        |           |          |
| 6330    |        |        |        |        |           |          |
| 6331    |        |        |        |        |           |          |
| 6332    |        |        |        |        |           |          |
| 6333    |        |        |        |        |           |          |
| 6334    |        |        |        |        |           |          |
| 6335    |        |        |        |        |           |          |
| 6336    | 071454 | 000000 |        |        |           |          |
| 6337    | 071456 | 000000 |        |        |           |          |
| 6338    | 071460 | 000000 |        |        |           |          |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 228

|      |        |        |                               |                  |                                      |
|------|--------|--------|-------------------------------|------------------|--------------------------------------|
| 6339 |        |        |                               |                  |                                      |
| 6340 | 071467 | 000000 | T38DLY: .WORD                 | 0                | ;DELAY COUNTER FOR TEST              |
| 6342 |        | 071470 |                               | .*<..+10>E177770 |                                      |
| 6344 | 071470 |        | T38PACKET:                    |                  | ;COMMAND PACKET FOR TEST             |
| 6345 | 071470 | 140006 | .WORD                         | 140006           | ;WRITE SUBSYSTEM MEM. CMD. ACK,CVC=1 |
| 6346 | 071472 | 071500 | .WORD                         | T38TAD           | ;ADDRESS OF CHARACTERISTICS BLOCK    |
| 6347 | 071474 | 000000 | .WORD                         | 0                |                                      |
| 6348 | 071476 | 000012 | .WORD                         | 10.              | ;STARTING VALUE OF BLOCK SIZE        |
| 6349 | 071500 |        | T38TAD:                       |                  | ;CHARACTERISTICS DATA BLOCK          |
| 6350 | 071500 | 000    | T38BS0: .BYTE                 | 0                | ;BSEL0 BYTE                          |
| 6351 | 071501 | 000    | T38BS1: .BYTE                 | 0                | ;BSEL1 BYTE                          |
| 6352 | 071502 | 000000 | T38BS2: .WORD                 | 0                | ;BSEL1 WORD                          |
| 6353 | 071504 | 000000 | .WORD                         | 0                | ;DATA                                |
| 6354 | 071506 |        | T38BFR: .BLKW                 | 150.             | ;MESSAGE BUFFER                      |
| 6355 | 072162 | 000000 | T38EB: .WORD                  |                  | ;END OF BUFFER ADDRESS               |
| 6356 |        |        |                               |                  |                                      |
| 6357 |        |        |                               |                  |                                      |
| 6359 |        | 072170 |                               | .*<..+10>E177770 |                                      |
| 6361 | 072170 |        | T38PK2:                       |                  | ;COMMAND PACKET FOR TEST             |
| 6362 | 072170 | 140004 | .WORD                         | 140004           | ;WRITE CHARA. MEM. CMD., ACK,CVC=1   |
| 6363 | 072172 | 072200 | .WORD                         | T38DTA           | ;ADDRESS OF SELECT DATA BLOCK        |
| 6364 | 072174 | 000000 | .WORD                         | 0                |                                      |
| 6365 | 072176 | 000012 | .WORD                         | 10.              | ;STARTING VALUE OF BLOCK SIZE        |
| 6366 |        |        |                               |                  |                                      |
| 6367 |        |        |                               |                  |                                      |
| 6368 | 072200 |        | T38DTA:                       |                  | ;SELECT DATA BLOCK                   |
| 6369 | 072200 | 071506 | .WORD                         | T38BFR           | ;ADDRESS OF MESSAGE BUFFER           |
| 6370 | 072202 | 000000 | .WORD                         | 0                |                                      |
| 6371 | 072204 | 000400 | .WORD                         | 256.             | ;LENGTH OF MESSAGE BUFFER            |
| 6372 | 072206 | 000000 | T38EAI: .WORD                 | 0                | ;EAI BIT WORD                        |
| 6373 | 072210 | 000000 | T38DSW: .WORD                 | 0                | ;DRIVE SELECT WORD ETC               |
| 6375 |        | 072220 |                               | .*<..+10>E177770 |                                      |
| 6377 | 072220 | 140212 | T38PK3: .WORD                 | 140212           | ;MESSAGE BUFFER RELEASE COMMAND      |
| 6378 | 072222 | 000000 | .WORD                         | 0                | ;NOT USED                            |
| 6379 | 072224 | 000000 | .WORD                         | 0                | ;NOT USED                            |
| 6380 | 072226 | 000000 | .WORD                         | 0                | ;NOT USED                            |
| 6381 | 072230 | 000000 | .WORD                         | 0                | ;NOT USED                            |
| 6382 |        |        |                               |                  |                                      |
| 6383 |        |        | ;WRITE TAPE PACKET            |                  |                                      |
| 6384 |        |        |                               |                  |                                      |
| 6386 |        | 072240 |                               | .*<..+10>E177770 |                                      |
| 6388 | 072240 | 140005 | T38PK4: .WORD                 | 140005           | ;WRITE, ACK, CVC=1 COMMAND           |
| 6389 | 072242 | 000000 | T38WR: .WORD                  | 0                | ;ADDRESS OF WRITE BUFFER             |
| 6390 | 072244 | 000000 | .WORD                         | 0                | ;MORE ADDRESS OF WRITE BUFFER        |
| 6391 | 072246 | 000400 | T38SIZ: .WORD                 | 256.             | ;SIZE OF RECORD                      |
| 6392 |        |        |                               |                  |                                      |
| 6393 |        |        |                               |                  |                                      |
| 6394 |        |        |                               |                  |                                      |
| 6395 |        |        |                               |                  |                                      |
| 6396 |        |        |                               |                  |                                      |
| 6397 |        |        | ;;                            |                  |                                      |
| 6398 |        |        | LOCAL TEXT" MESSAGES FOR TEST |                  |                                      |
| 6399 |        |        | ;                             |                  |                                      |
| 6400 |        |        |                               |                  |                                      |
| 6401 |        |        |                               |                  |                                      |
| 6402 |        |        |                               |                  |                                      |
| 6403 |        |        |                               |                  |                                      |

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 229

|      |        |     |     |     |         |        |                                                              |
|------|--------|-----|-----|-----|---------|--------|--------------------------------------------------------------|
| 6404 | 072250 | 123 | 164 | 141 | T38NE:  | .ASCIZ | 'Stand-alone Manual Intervention Not Executed'               |
| 6405 | 072325 | 045 | 116 | 045 | T38MS3: | .ASCIZ | 'NNA Type <RETURN> To Stop Servo Exerciser, Return To Menu'  |
| 6406 | 072420 | 124 | 123 | 123 | T38WRT: | .ASCIZ | 'TSSR Not Correct After WRITE, With WRITE PROTECT On'        |
| 6407 | 072504 | 127 | 122 | 111 | T38WRL: | .ASCIZ | 'WRITE LOCKED Bit Not Set In XST0'                           |
| 6408 | 072545 | 127 | 122 | 111 | T38WLE: | .ASCIZ | 'WRITE LOCK ERROR Bit Not Set In XST0'                       |
| 6409 | 072612 | 127 | 122 | 111 | T38NBA: | .ASCIZ | 'WRITE SUBSYSTEM MEMORY Command Not Accepted'                |
| 6410 | 072666 | 103 | 157 | 156 | T38SSR: | .ASCIZ | 'Contents of TSSR Incorrect After WRITE SUBSYSTEM MEMORY'    |
| 6411 | 072756 | 045 | 116 | 045 | T38INT: | .ASCIZ | 'NNA Interrupt Received'                                     |
| 6412 | 073006 | 045 | 116 | 045 | T38ONL: | .ASCIZ | 'NNA Drive Is Now ON-LINE'                                   |
| 6413 | 073042 | 045 | 116 | 045 | T38OFL: | .ASCIZ | 'NNA Drive Is Now OFF-LINE'                                  |
| 6414 | 073076 | 103 | 157 | 156 | T38SST: | .ASCIZ | 'Contents Of TSSR Incorrect After MESSAGE BUFFER RELEASE'    |
| 6415 | 073166 | 045 | 116 | 045 | T38MS1: | .ASCIZ | 'NNA Toggle ON-LINE Switch to Generate ATTENTION Interrupts' |
| 6416 | 073261 | 045 | 116 | 045 | T38MS2: | .ASCIZ | 'NNA Type RETURN To Return To Menu'                          |
| 6417 | 073325 | 111 | 163 | 040 | T38MSG: | .ASCIZ | 'Is Write-Protected Tape Mounted'                            |
| 6418 | 073365 | 115 | 141 | 156 | T38ID:  | .ASCIZ | 'Manual Intervention'                                        |

|      |        |        |        |        |         |       |                         |
|------|--------|--------|--------|--------|---------|-------|-------------------------|
| 6419 |        |        |        |        |         | .EVEN |                         |
| 6420 | 073412 | 073436 | 073510 | 073536 | MIMENU: | .WORD | 1\$,2\$,3\$,4\$,5\$,6\$ |
| 6421 | 073426 | 073705 | 073750 | 074013 |         | .WORD | 8\$,9\$,10\$,0          |

|      |        |     |     |     |       |        |                                                |
|------|--------|-----|-----|-----|-------|--------|------------------------------------------------|
| 6422 |        |     |     |     |       |        |                                                |
| 6423 | 073436 | 012 | 123 | 105 | 1\$:  | .ASCIZ | <12>'SELECT OPERATION FROM FOLLOWING OPTIONS:' |
| 6424 | 073510 | 012 | 011 | 060 | 2\$:  | .ASCIZ | <12>' 0 Display This Menu'                     |
| 6425 | 073536 | 011 | 061 | 011 | 3\$:  | .ASCIZ | ' 1 Turn On All M7196 LED's'                   |
| 6426 | 073570 | 011 | 062 | 011 | 4\$:  | .ASCIZ | ' 2 Turn Off All M7196 LED's'                  |
| 6427 | 073623 | 011 | 063 | 011 | 5\$:  | .ASCIZ | ' 3 Offline/Online Attention'                  |
| 6428 | 073657 | 011 | 064 | 011 | 6\$:  | .ASCIZ | ' 4 Write Protect Test'                        |
| 6429 | 073705 | 011 | 065 | 011 | 8\$:  | .ASCIZ | ' 5 Print Extended Transport Status'           |
| 6430 | 073750 | 011 | 066 | 011 | 9\$:  | .ASCIZ | ' 6 Return to Diagnostic Supervisor'           |
| 6431 | 074013 | 000 |     |     | 10\$: | .ASCIZ | ' '                                            |
| 6432 |        |     |     |     |       | .EVEN  |                                                |

6433  
6434  
6435  
6436  
6437  
; LOCAL STORAGE FOR THIS TEST  
;-

|      |        |        |        |  |          |       |               |                                       |
|------|--------|--------|--------|--|----------|-------|---------------|---------------------------------------|
| 6438 | 074014 | 000000 |        |  | T38DAT:  | .WORD | 0             | ; LOGICAL RESPONSE TO QUESTION        |
| 6439 | 074016 |        |        |  | T38REST: |       |               |                                       |
| 6440 | 074016 |        |        |  | SAVREG   |       |               | ; SAVE THE REGISTERS                  |
| 6441 | 074022 | 012701 | 071470 |  | MOV      |       | @T38PACKET,R1 | ; START OF THE PACKET                 |
| 6442 | 074026 | 012721 | 140206 |  | MOV      |       | @140206,(R1)+ | ; WRITE SUBSYSTEM MEM. WITH ACK,CVC=1 |
| 6443 | 074032 | 012721 | 071500 |  | MOV      |       | @T38TAD,(R1)+ | ; ADDRESS OF DATA BLOCK               |
| 6444 | 074036 | 005021 |        |  | CLR      |       | (R1)+         | ; EXTENDED ADDRESS                    |
| 6445 | 074040 | 012721 | 000006 |  | MOV      |       | @6,(R1)+      | ; SIZE OF DATA BLOCK IN BYTES         |
| 6446 | 074044 | 005021 |        |  | CLR      |       | (R1)+         | ; CLEAR BSEL0 AND BSEL1               |
| 6447 | 074046 | 005021 |        |  | CLR      |       | (R1)+         | ; CLEAR SEL2                          |
| 6448 | 074050 | 005011 |        |  | CLR      |       | (R1)          | ; CLEAR DATA AREA                     |
| 6449 | 074052 | 000207 |        |  | RTS      |       | PC            | ; RETURN                              |

6450  
6451  
6452  
6453  
6454  
6455  
6456  
6457  
6458  
6459  
6460  
; THIS ROUTINE PRINTS THE CONTENTS OF  
; THE 256 BYTE MESSAGE BUFFER RETURNED BY THE  
; TSV-05.

; INPUT:

; R0 LOW ORDER ADDRESS OF MESSAGE BUFFER

K2

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 230

```

6461      ;      R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
6462      ;      NOTE: R1 IS IGNORED IF KENABLE FLAG IS CLEAR
6463      ;
6464      ;
6465      ;
6466      T38MBP;
6467 074054
6468 074054
6469 074060 010005
6470 074062 005737 003134
6471 074066 001001
6472 074070 005001
6473 074072 010103
6474 074074 006100
6475 074076 006101
6476 074100
      074100 010546
      074102 010146
      074104 012746 074356
      074110 012746 000003
      074114 010600
      074116 104415
      074120 062706 000010
6477 074124
      074124 012746 074423
      074130 012746 000001
      074134 010600
      074136 104415
      074140 062706 000004
6478 074144 010501
6479 074146 010300
6480 074150 001403
6481 074152 004737 017316
6482 074156 010005
6483 074160 010537 074524
6484 074164 011504
6485 074166 022704 125252
6486 074172 001417
6487 074174 010403
6488 074176 042704 170377
6489 074202 000241
6490 074204 006004
6491 074206 006004
6492 074210 006004
6493 074212 006004
6494 074214 042703 177760
6495 074220 060403
6496 074222 010325
6497 074224 020527 072162
6498 074230 001355
6499 074232 013705 074524
6500 074236 012704 000001
6501 074242
      074242 012546
      074244 010446
      074246 012746 074500
      074252 012746 000003

      SAVREG
      MOV R0,R5
      TST KENABLE
      BNE 910$
      CLR R1
      MOV R1,R3
      ROL R0
      ROL R1
      PRINTX @T38AS0,R1,R5
      ;SA : THE REGISTERS
      ;SAVE LOW ORDER ADDRESS
      ;ADDRESS ABOVE 28K?
      ;BR IF YES
      ;SET HIGH ORDER ADDRESS TO 0
      ;SAVE HIGH ORDER ADDRESS
      ;SHIFT BIT15 TO C BIT
      ;SHIFT TO HIGH ORDER FOR PRINTOUT
      ;PRINT MESSAGE BUFFER ADDRESS
      MOV R5,-(SP)
      MOV R1,-(SP)
      MOV @T38AS0,-(SP)
      MOV @3,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD @10,SP
      PRINTX @T38AS1
      ;PRINT HEADER FOR CONTENTS
      MOV @T38AS1,-(SP)
      MOV @1,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD @4,SP
      MOV R5,R1
      MOV R3,R0
      BEQ 913$
      JSR PC,SETMAP
      MOV R0,R5
      MOV R5,T38CNT
      MOV (R5),R4
      CMP @125252,R4
      BEQ 912$
      MOV R4,R3
      BIC @170377,R4
      CLC
      ROR R4
      ROR R4
      ROR R4
      ROR R4
      BIC @177760,R3
      ADD R4,R3
      MOV R3,(R5)+
      CMP R5,@T38EB
      BNE 911$
      MOV T38CNT,R5
      MOV @1,R4
      PRINTX @T38ASN,R4,(R5)+
      ;COPY LOW ORDER ADDRESS
      ;COPY HIGH ORDER ADDRESS
      ;BR IF NOT ABOVE 28K
      ;SETUP PAR ADDRESS IN R0
      ;GET PAR FORMAT ADDRESS ABOVE 28K
      ;HOLD ADDRESS
      ;GET BUFFER ENTRY
      ;CHECK FOR NO LOAD CONDITION
      ;BR, IF BUFFER WASN'T LOADED
      ;MAKE COPY
      ;ONLY BITS 11,10,9 AND 8 ARE SAVED
      ;CLEAR CARRY
      ;11 TO 10 BIT POSITION
      ;10 TO 9 BIT POSITION
      ;9 TO 8 BIT POSITION
      ;8 TO 7 BIT POSITION
      ;ONLY BITS 3,2,1 AND 0 ARE SAVED
      ;"OR'EM TOGETHER
      ;PUT BACK IN BUFFER
      ;END OF BUFFER YET
      ;BR, IF NOT AT END YET
      ;PUT ADDRESS BACK
      ;START BYTE NUMBER AT ONE
      ;PRT MEM BUFFER W/NEWLINE
      MOV (R5)+,-(SP)
      MOV R4,-(SP)
      MOV @T38ASN,-(SP)
      MOV @3,-(SP)

```

L2

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 10: MANUAL INTERVENTION

SEQ 231

```

074256 010600
074260 104415
074262 062706 000010
6502 074266 005037 074524
6503 074272 000412
6504 074274
074274 012546
074276 010446
074300 012746 074461
074304 012746 000003
074310 010600
074312 104415
074314 062706 000010
6505 074320 005237 074524
6506 074324 005204
6507 074326 020427 000200
6508 074332 003010
6509 074334 023727 074524 000004
6510 074342 001401
6511 074344 000753
6512 074346 005037 074524
6513 074352 000733
6514 074354 000207
6515
6516 074356 045 116 045 T38AS0: .ASCIZ 'NWA Message Buffer Address = 0105'
6517 074423 045 116 045 T38AS1: .ASCIZ 'NWA Message Buffer Contents:'
6518 074461 045 101 040 T38ASC: .ASCIZ 'A D4A: 03'
6519 074500 045 116 045 T38ASN: .ASCIZ 'NWA ByteD4A: 03'
6520 .EVEN
6521 074524 000000 T38CNT: .WORD
6522 074526
074526
074526 104401
6523
6524
6525
6526
6527
6528
6529
6530
6531
6532
6533
6534
6535
6536
6537
6538
6539
6540
6541
6542
6543
6544
6545
6546

          CLR      T38CNT          ;CLEAR COUNTER
          BR       921$            ;SKIP OTHER PRINT
          PRINTX   0T38ASC,R4,(R5)+ ;PRINT THE CONTENTS OF MEMORY BUFFER

920$:
          MOV      (R5)+, -(SP)
          MOV      R4, -(SP)
          MOV      0T38ASC, -(SP)
          MOV      03, -(SP)
          MOV      SP, R0
          TRAP     C$PNTX
          ADD      010, SP

921$: INC      T38CNT          ;BUMP COUNTER
      INC      R4             ;NUMBER OF THE NEXT
      CMP      R4, 0128.      ;DONE ALL YET ?
      BGT      50$            ;BRANCH IF ALL DONE
      CMP      T38CNT, 04      ;DONE FOUR YET
      BEQ      925$            ;BR, IF THREE DONE
      BR       920$            ;KEEP GOING
925$: CLR      T38CNT          ;CLEAR COUNTER
      BR       915$            ;PRINT WITH NEW LINE
50$:  RTS      PC             ;RETURN

T38AS0: .ASCIZ 'NWA Message Buffer Address = 0105'
T38AS1: .ASCIZ 'NWA Message Buffer Contents:'
T38ASC: .ASCIZ 'A D4A: 03'
T38ASN: .ASCIZ 'NWA ByteD4A: 03'
      .EVEN
T38CNT: .WORD
      ENDTST

          L10075: TRAP     C$ETST

.SBTTL TEST 11: CONFIGURATION TYPEOUT

;THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL
;THE CONFIGURATION OF THE M7196 MODULE AND TSV05 SUBSYSTEM. SPECIFICALLY,
;THE FOLLOWING INFORMATION IS PRESENTED:
;
; 1.0 STATE OF THE EXTENDED FEATURES SWITCH ON THE M7196: ON (EXTENDED
;     FEATURES ENABLED) OR OFF (EXTENDED FEATURES DISABLED),
;
; 2.0 STATE OF THE BUFFERING ENABLE SWITCH ON THE M7196: ON
;     (BUFFERING ENABLED) OR OFF (BUFFERING DISABLED),
;
; 3.0 MICROCODE REVSION LEVEL OF THE M7196,
;
; 4.0 NUMBER OF TAPE TRANSPORTS CONNECTED TO THE CONTROLLER,
;
; 5.0 UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED)
;     OF EACH CONNECTED TRANSPORT. IN ADDITION, THE PROGRAM WILL INDICATE,
;     FOR EACH ON-LINE TRANSPORT, WHETHER OR NOT IT IS EQUIPPED WITH THE
;     EXTENDED TAPE STATUS READOUT FEATURE.
;
;THE OPERATOR IS EXPECTED TO READ THE PRINTOUT AND VERIFY THAT IT MATCHES

```

```

6547 ;THE ACTUAL CONFIGURATION AT HAND. IF, FOR EXAMPLE, THE PROGRAM INDICATES
6548 ;THAT IT "SEES" TWO TRANSPORTS CONNECTED WHEN IN FACT ONLY ONE IS PRESENT,
6549 ;THE OPERATOR MUST INTERPRET THIS AS AN ERROR AND ATTEMPT TO FIND THE
6550 ;CAUSE (BAD CABLE, FAULTY UNIT-SELECT DECODING IN THE TRANSPORT, ETC.).
6551 ;[SINCE THE CONTROLLER CAN ONLY ACCESS UNIT 0 IF IT IS IN "STANDARD"
6552 ;MODE, THE PROGRAM WILL FORCE THE MODULE INTO EXTENDED MODE VIA THE
6553 ;WRITE SUBSYSTEM MEMORY COMMAND IN ORDER TO SCAN FOR CONNECTED TRANSPORTS.]
6554 ;
6555 ;
6556 ;THIS ROUTINE, WHEN ITS ACTIONS ARE COMPLETED, WILL EXIT BACK TO THE
6557 ;DIAGNOSTIC SUPERVISOR SO THAT IF ADDITIONAL UNITS (CONTROLLERS) ARE
6558 ;SELECTED (E.G., FROM THE INITIAL STARTUP DIALOG), THE ROUTINE WILL BE
6559 ;REENTERED SO THAT THEIR CONFIGURATIONS CAN BE PRINTED.
6560 ;
6561 074530      BGNTST
6562 074530      RFLAGS  R0
6563 074530      T11::
6564 074530      ;GET OPERATOR FLAGS
6565 074530      TRAP  C$RFLA
6566 074530      104421
6567 074532      001403
6568 074534      012700 076523
6569 074540      000402
6570 074542      012700 077652
6571 074546      004737 016510
6572 074552      004737 020500
6573 074556      103402
6574 074560      000137 075740
6575 074564      20$:
6576 074564      004737 015774
6577 074570      103405
6581 074572      010001
6582 074574      104455
6583 074576      002115
6584 074600      003652
6585 074602      012034
6586 074604      25$: CKLOOP
6587 074604      104406
6588 074606      013737 002202 076470
6589 074614      012704 076450
6590 074620      004737 010662
6591 074624      103405
6592 074626      010001
6593 074630      074630 104456
6594 074632      074632 002116
6595 074634      074634 005056
6596 074636      074636 012034
6597 074640      50$: CKLOOP
6598 074640      104406
6599 074642      013701 076000
6600 074646      074646 012746 077371
6601 074652      074652 012746 000001
6602 074656      074656 010600
6603 074660      074660 104415
6604 074662      074662 062706 000004
6605 074666      074666 032701 000200
6606      BIT  #BIT7,R1
6607      ;DO SOFT INIT OF CONTROLLER
6608      ;BR IF SOFT INIT = OK
6609      ;SAVE CONTENTS OF TSSR
6610      ;DEVICE FATAL ERROR DURING INIT
6611      TRAP  C$ERDF
6612      .WORD 1101
6613      .WORD SFIERR
6614      .WORD SFIMSG
6615      ;LOOP IF SELECTED
6616      TRAP  C$CLP1
6617      ;SET UNIT NUMBER
6618      ;SUBROUTINE NEEDS PACKET ADDRESS
6619      ;ISSUE WRITE CHARACTERISTICS
6620      ;BR, IF COMMAND ISSUED OK
6621      ;SAVE CONTENTS OF TSSR
6622      ;WRITE CHARACTERISTICS FAILED
6623      TRAP  C$ERHRD
6624      .WORD 1102
6625      .WORD WRTMSG
6626      .WORD SFIMSG
6627      ;LOOP IF SELECTED
6628      TRAP  C$CLP1
6629      ;GET XST2 STATUS FROM MESSAGE BUFFER
6630      ;"STATE OF EXTENDED FEATURES SW = "
6631      MOV  #T39SFS, -(SP)
6632      MOV  #1, -(SP)
6633      MOV  SP, R0
6634      TRAP C$PNTX
6635      ADD  #4, SP
6636      ;CHECK STATE OF E.F.S.

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

SEQ 233

|      |        |        |        |
|------|--------|--------|--------|
| 6597 | 074672 | 001011 |        |
| 6598 | 074674 |        |        |
|      | 074674 | 012746 | 077515 |
|      | 074700 | 012746 | 000001 |
|      | 074704 | 010600 |        |
|      | 074706 | 104415 |        |
|      | 074710 | 062706 | 000004 |
| 6599 | 074714 | 000410 |        |
| 6600 | 074716 |        |        |
|      | 074716 | 012746 | 077524 |
|      | 074722 | 012746 | 000001 |
|      | 074726 | 010600 |        |
|      | 074730 | 104415 |        |
|      | 074732 | 062706 | 000004 |
| 6601 | 074736 |        |        |
|      | 074736 | 012746 | 077443 |
|      | 074742 | 012746 | 000001 |
|      | 074746 | 010600 |        |
|      | 074750 | 104415 |        |
|      | 074752 | 062706 | 000004 |
| 6602 | 074756 | 032701 | 000100 |
| 6603 | 074762 | 001011 |        |
| 6604 | 074764 |        |        |
|      | 074764 | 012746 | 077515 |
|      | 074770 | 012746 | 000001 |
|      | 074774 | 010600 |        |
|      | 074776 | 104415 |        |
|      | 075000 | 062706 | 000004 |
| 6605 | 075004 | 000410 |        |
| 6606 | 075006 |        |        |
|      | 075006 | 012746 | 077524 |
|      | 075012 | 012746 | 000001 |
|      | 075016 | 010600 |        |
|      | 075020 | 104415 |        |
|      | 075022 | 062706 | 000004 |
| 6607 | 075026 | 042701 | 177700 |
| 6608 | 075032 | 010137 | 077610 |
| 6609 | 075036 |        |        |
|      | 075036 | 013746 | 077610 |
|      | 075042 | 012746 | 077533 |
|      | 075046 | 012746 | 000002 |
|      | 075052 | 010600 |        |
|      | 075054 | 104415 |        |
|      | 075056 | 062706 | 000006 |
| 6610 | 075062 | 004737 | 015774 |
| 6611 | 075056 | 103405 |        |
| 6615 | 075070 | 010001 |        |
| 6616 | 075072 |        |        |
|      | 075072 | 104455 |        |
|      | 075074 | 002117 |        |
|      | 075076 | 003652 |        |
|      | 075100 | 012034 |        |
| 6617 | 075102 |        |        |
|      | 075102 | 104406 |        |
| 6618 | 075104 | 013737 | 002202 |
| 6619 | 075112 | 012704 | 076450 |
| 6620 | 075116 | 004737 | 010662 |

```

BNE      100$
PRINTX   0T390FF

100$:    BR      110$
PRINTX   0T390N

110$:    PRINTX   0T39SBS

        BIT      0BIT6,R1
        BNE      120$
        PRINTX   0T390FF

120$:    BR      130$
PRINTX   0T390N

130$:    BIC      0177700,R1
        MOV      R1,T39RL
        PRINTX   0T39MCL,T39RL

        JSR      PC,SOFINIT
        BCS      140$
        MOV      R0,R1
        ERWDF    ERRNO,SFIERR,SFIMSG

140$:    CKLOOP

        MOV      UNITN,T39DSW
        MOV      0T39PK2,R4
        JSR      PC,WATCHR

```

```

;BR, IF EXT. FEA. SW. IS ON
;" OFF"
                                MOV      T39OFF, -(SP)
                                MOV      T1, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T4, SP
;SKIP OTHER PRINT STATEMENT
;" ON "
                                MOV      T39ON, -(SP)
                                MOV      T1, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T4, SP
;"STATE OF BUFFERING SWITCH ="
                                MOV      T39SBS, -(SP)
                                MOV      T1, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T4, SP
;CHECK STATE OF BUFFERING SW
;BR, IF BUFFERING IS ON
;" OFF"
                                MOV      T39OFF, -(SP)
                                MOV      T1, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T4, SP
;SKIP OTHER PRINT STATEMENT
;" ON "
                                MOV      T39ON, -(SP)
                                MOV      T1, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T4, SP
;ONLY LEAVE MICROCODE REV LEVEL
;LOAD UP REV LEVEL
;"MICROCODE REVISION LEVEL -000XXX"
                                MOV      T39RL, -(SP)
                                MOV      T39MCL, -(SP)
                                MOV      T2, -(SP)
                                MOV      SP, R0
                                TRAP     C$PNTX
                                ADD      T6, SP
;DO SOFT INIT OF CONTROLLER
;BR IF SOFT INIT = ON
;SAVE CONTENTS OF TSSR
;DEVICE FATAL ERROR DURING INIT
                                TRAP     C$EPDF
                                .WORD    1103
                                .WORD    $FIERQ
                                .WORD    $FIMSC
;LOOP IF SELECTED
                                TRAP     C$CLP1
;SET UNIT NUMBER
;SUBROUTINE NEEDS PACKET ADDRESS
;ISSUE WRITE CHARACTERISTICS

```

TSVS - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

SEQ 234

|      |        |        |        |        |       |                                                |                     |  |                                                   |         |  |
|------|--------|--------|--------|--------|-------|------------------------------------------------|---------------------|--|---------------------------------------------------|---------|--|
| 6621 | 075122 | 103405 |        |        |       | BCS                                            | 150#                |  | BR, IF COMMAND ISSUED OK                          |         |  |
| 6625 | 075124 | 010001 |        |        |       | MOV                                            | R0,R1               |  | SAVE CONTENTS OF TSSR                             |         |  |
| 6626 | 075126 |        |        |        |       | ERRHRD                                         | ERRNO,WRTMSG,SFIMSG |  | WRITE CHARACTERISTICS FAILED                      |         |  |
|      | 075126 | 104456 |        |        |       |                                                |                     |  | TRAP                                              | C%ERHND |  |
|      | 075130 | 002120 |        |        |       |                                                |                     |  | .WORD                                             | 1104    |  |
|      | 075132 | 005056 |        |        |       |                                                |                     |  | .WORD                                             | WRTMSG  |  |
|      | 075134 | 012034 |        |        |       |                                                |                     |  | .WORD                                             | SFIMSG  |  |
| 6627 | 075136 |        |        |        | 150#: | CKLOOP                                         |                     |  | LOOP IF SELECTED                                  |         |  |
|      | 075136 | 104406 |        |        |       |                                                |                     |  | TRAP                                              | C%CLP1  |  |
| 6628 | 075140 | 005737 | 002226 |        |       | TST                                            | EXTFEA              |  | CHECK FOR EXTENDED FEATURES SW SWITCH             |         |  |
| 6629 | 075144 | 001036 |        |        |       | BNE                                            | 174#                |  | BR IF SWITCH IS ON                                |         |  |
| 6630 | 075146 | 112737 | 000200 | 075761 |       | MOVB                                           | #200,T39BS1         |  | WRITE MISCELLANEOUS CONT/READ STATUS              |         |  |
| 6631 | 075154 | 112737 | 000010 | 075760 |       | MOVB                                           | #10,T39BS0          |  | FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH) |         |  |
| 6632 | 075162 | 012704 | 075750 |        |       | MOV                                            | #T39PACKET,R4       |  | WRITE SUBSYS MEM PACKET                           |         |  |
| 6633 | 075166 | 010465 | 000000 |        |       | MOV                                            | R4,TSD(R5)          |  | ISSUE COMMAND                                     |         |  |
| 6634 | 075172 | 004737 | 016336 |        |       | JSR                                            | PC,CHKTSSR          |  | WAIT FOR SSR                                      |         |  |
| 6635 | 075176 | 103405 |        |        |       | BCS                                            | 160#                |  | BR, IF NO ERROR                                   |         |  |
| 6636 | 075200 | 010001 |        |        |       | MOV                                            | R0,R1               |  | ERROR, SAVE TSSR                                  |         |  |
| 6640 | 075202 |        |        |        |       | ERRHRD                                         | ERRNO,T39NBA,PKTSSR |  | TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS         |         |  |
|      | 075202 | 104456 |        |        |       |                                                |                     |  | TRAP                                              | C%ERHND |  |
|      | 075204 | 002121 |        |        |       |                                                |                     |  | .WORD                                             | 1105    |  |
|      | 075206 | 077225 |        |        |       |                                                |                     |  | .WORD                                             | T39NBA  |  |
|      | 075210 | 012040 |        |        |       |                                                |                     |  | .WORD                                             | PKTSSR  |  |
| 6641 | 075212 |        |        |        | 160#: | CKLOOP                                         |                     |  | LOOP IF SELECTED                                  |         |  |
|      | 075212 | 104406 |        |        |       |                                                |                     |  | TRAP                                              | C%CLP1  |  |
| 6642 | 075214 | 012704 | 076450 |        |       | MOV                                            | #T39PK2,R4          |  | SUBROUTINE NEEDS PACKET ADDRESS                   |         |  |
| 6643 |        |        |        |        |       | ;*****                                         |                     |  |                                                   |         |  |
| 6644 |        |        |        |        |       | ;                                              |                     |  |                                                   |         |  |
| 6645 |        |        |        |        |       | WRITE CHARACTERISTICS COMMAND (CALL TO WRTHCR) |                     |  |                                                   |         |  |
| 6646 |        |        |        |        |       | ;                                              |                     |  |                                                   |         |  |
| 6647 |        |        |        |        |       | ;*****                                         |                     |  |                                                   |         |  |
| 6648 |        |        |        |        |       |                                                |                     |  |                                                   |         |  |
| 6649 | 075220 | 004737 | 010662 |        |       | JSR                                            | PC,WRTHCR           |  | ISSUE WRITE CHARACTERISTICS                       |         |  |
| 6650 | 075224 | 103405 |        |        |       | BCS                                            | 170#                |  | BR, IF COMMAND ISSUED OK                          |         |  |
| 6654 | 075226 | 010001 |        |        |       | MOV                                            | R0,R1               |  | SAVE CONTENTS OF TSSR                             |         |  |
| 6655 | 075230 |        |        |        |       | ERRHRD                                         | ERRNO,WRTMSG,SFIMSG |  | WRITE CHARACTERISTICS FAILED                      |         |  |
|      | 075230 | 104456 |        |        |       |                                                |                     |  | TRAP                                              | C%ERHND |  |
|      | 075232 | 002122 |        |        |       |                                                |                     |  | .WORD                                             | 1106    |  |
|      | 075234 | 005056 |        |        |       |                                                |                     |  | .WORD                                             | WRTMSG  |  |
|      | 075236 | 012034 |        |        |       |                                                |                     |  | .WORD                                             | SFIMSG  |  |
| 6656 | 075240 |        |        |        | 170#: | CKLOOP                                         |                     |  | SCOPE LOOP                                        |         |  |
|      | 075240 | 104406 |        |        |       |                                                |                     |  | TRAP                                              | C%CLP1  |  |
| 6657 | 075242 | 005037 | 002202 |        |       | CLR                                            | UNITN               |  | SET TO DRIVE 0                                    |         |  |
| 6658 | 075246 | 013737 | 002202 | 076470 | 174#: | MOV                                            | UNITN,T39DSW        |  | SET UNIT NUMBER                                   |         |  |
| 6659 | 075254 | 012704 | 076450 |        | 175#: | MOV                                            | #T39PK2,R4          |  | SUBROUTINE NEEDS PACKET ADDRESS                   |         |  |
| 6660 | 075260 | 004737 | 010662 |        |       | JSR                                            | PC,WRTHCR           |  | ISSUE WRITE CHARACTERISTICS                       |         |  |
| 6661 | 075264 | 103405 |        |        |       | BCS                                            | 180#                |  | BR, IF COMMAND ISSUED OK                          |         |  |
| 6665 | 075266 | 010001 |        |        |       | MOV                                            | R0,R1               |  | SAVE CONTENT                                      |         |  |

C3

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

```

6670 075306 032701 000100          BIT    #0FL,R1
6671 075312 001414          BEQ     200$
6672 075314          PRINTX   #T390F2,UNITN
      075314 013746 002202
      075320 012746 076764
      075324 012746 000002
      075330 010600
      075332 104415
      075334 062703 000006
6673 075340 000137 075674          JMP     250$
6674 075344          PRINTX   #T390N2,UNITN
      075344 013746 002202
      075350 012746 077030
      075354 012746 000002
      075360 010600
      075362 104415
      075364 062706 000006
6675 075370 013701 075774          MOV     T398FR+6,R1
6676 075374 032701 000004          BIT     #BIT2,R1
6677 075400 001013          BNE     210$
6678 075402          PRINTX   #T39WPN,UNITN
      075402 013746 002202
      075406 012746 077146
      075412 012746 000002
      075416 010600
      075420 104415
      075422 062706 000006
6679 075426 000412          BR      220$
6680 075430          PRINTX   #T39WRT,UNITN
      075430 013746 002202
      075434 012746 077073
      075440 012746 000002
      075444 010600
      075446 104415
      075450 062706 000006
6681 075454 012737 125252 076066 220$: MOV     #125252,T398FR+100
6682 075462 112737 000000 075761      MOV     #0,T39BS1
6683 075470 112737 000024 075760      MOV     #24,T39BS0
6684 075476 012704 075750      MOV     #T39PACKET,R4
6685 075502 010465 000000      MOV     R4,T39DB(R5)
6686 075506 012737 000144 075744      MOV     #100.,T39DLY
6687 075514 004737 016250      JSR     PC,WAITF
6688 075520 016501 000002      MOV     YSSR(R5),R1
6689 075524 032701 000200      BIT     #SSR,R1
6690 075530 001017          BNE     225$
6691 075532          DELAY     250
      075532 012727 000250
      075536 000000
      075540 013727 002116
      075544 000000
      075546 005367 177772
      075552 001375
      075554 005367 177756
      075560 001367
6692 075562 005337 075744          DEC     T39DLY
6693 075566 001352          BNE     222$
6694 075570 004737 016336          JSR     PC,CHKTSSR

```

SEQ 235

```

;CHECK FOR OFF-LINE
;BR, IF DRIVE IS ON-LINE
;"DRIVE NUMBER XX IS OFF-LINE"
      MOV     UNITN,-(SP)
      MOV     #T390F2,-(SP)
      MOV     #2,-(SP)
      MOV     SP,R0
      TRAP    C#PNTX
      ADD     #6,SP

;DO NOT TRY TO GET ANYMORE INFO.
;"DRIVE NUMBER XX IS ON-LINE"
      MOV     UNITN,-(SP)
      MOV     #T390N2,-(SP)
      MOV     #2,-(SP)
      MOV     SP,R0
      TRAP    C#PNTX
      ADD     #6,SP

;READ EXTENDED STATUS (XSTO)
;IS DRIVE WRITE PROTECTED
;BR, IF WRITE PROTECTED
;"DRIVE NUMBER IS NOT WRT PRO"
      MOV     UNITN,-(SP)
      MOV     #T39WPN,-(SP)
      MOV     #2,-(SP)
      MOV     SP,R0
      TRAP    C#PNTX
      ADD     #6,SP

;SKIP OVER
;"DRIVE NUMBER XX IS WRT PRO"
      MOV     UNITN,-(SP)
      MOV     #T39WRT,-(SP)
      MOV     #2,-(SP)
      MOV     SP,R0
      TRAP    C#PNTX
      ADD     #6,SP

;SET 1 LOC TO KNOWN VALUE
;EXTENDED TAPE STATUS
;EXTENDED TAPE STATUS
;WRITE SUBSYS MEM PACKET
;ISSUE COMMAND
;SET UP DELAY ROUTINE
;WAIT AWHILE FOR SSR TO SET
;SEE IF IT REALLY DID
;JUST CHECK THAT BIT
;BR, IF SSR IS SET
;DELAY ABOUT .25 SEC
      MOV     #250,(PC)+
      .WORD   0
      MOV     L$DLY,(PC)+
      .WORD   0
      DEC     -6(PC)
      BNE     .4
      DEC     -22(PC)
      BNE     .-20

;START DELAY COUNT DOWN
;BR, IF COUNTER IS NOT AT DONE
;WAIT FOR SSR

```

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

SEQ 236

|      |        |        |        |        |            |                     |
|------|--------|--------|--------|--------|------------|---------------------|
| 6695 | 075574 | 103405 |        |        | BCS        | 230:                |
| 6696 | 075576 | 010001 |        |        | MOV        | R0,R1               |
| 6700 | 075600 |        |        |        | ERRHRD     | ERRNO,T39NBA,PKTSSR |
|      | 075600 | 104456 |        |        |            |                     |
|      | 075602 | 002124 |        |        |            |                     |
|      | 075604 | 077225 |        |        |            |                     |
|      | 075606 | 012046 |        |        |            |                     |
| 6701 | 075610 |        |        | 230:   | CKLOOP     |                     |
|      | 075610 | 104406 |        |        |            |                     |
| 6702 | 075612 | 023727 | 076066 | 125252 | CMP        | T39BFR+100,0125252  |
| 6703 | 075620 | 001013 |        |        | BNE        | 240:                |
| 6704 | 075622 |        |        |        | PRINTX     | 0T39ETN,UNITN       |
|      | 075622 | 013746 | 002202 |        |            |                     |
|      | 075626 | 012746 | 076671 |        |            |                     |
|      | 075632 | 012746 | 000002 |        |            |                     |
|      | 075636 | 010600 |        |        |            |                     |
|      | 075640 | 104415 |        |        |            |                     |
|      | 075642 | 062706 | 000006 |        |            |                     |
| 6705 | 075646 | 000412 |        |        | BR         | 250:                |
| 6706 | 075650 |        |        | 240:   | PRINTX     | 0T39ETS,UNITN       |
|      | 075650 | 013746 | 002202 |        |            |                     |
|      | 075654 | 012746 | 076602 |        |            |                     |
|      | 075660 | 012746 | 000002 |        |            |                     |
|      | 075664 | 010600 |        |        |            |                     |
|      | 075666 | 104415 |        |        |            |                     |
|      | 075670 | 062706 | 000006 |        |            |                     |
| 6707 | 075674 | 005237 | 002202 | 250:   | INC        | UNITN               |
| 6708 | 075700 | 023727 | 002202 | 000003 | CMP        | UNITN,03            |
| 6709 | 075706 | 001402 |        |        | BEQ        | 63:                 |
| 6710 | 075710 | 000137 | 075246 |        | JMP        | 175:                |
| 6711 | 075714 |        |        | 63:    | PRINTX     | 0T39NFL             |
|      | 075714 | 012746 | 076520 |        |            |                     |
|      | 075720 | 012746 | 000001 |        |            |                     |
|      | 075724 | 010600 |        |        |            |                     |
|      | 075726 | 104415 |        |        |            |                     |
|      | 075730 | 062706 | 000004 |        |            |                     |
| 6712 | 075734 | 000137 | 000200 |        | JMP        | 200                 |
| 6713 | 075740 |        |        | 64:    | EXIT       | TST                 |
|      | 075740 | 104432 |        |        |            |                     |
|      | 075742 | 001736 |        |        |            |                     |
| 6714 |        |        |        |        |            |                     |
| 6715 |        |        |        |        |            |                     |
| 6716 |        |        |        |        |            |                     |
| 6717 |        |        |        |        |            |                     |
| 6718 |        |        |        |        |            |                     |
| 6719 |        |        |        |        |            |                     |
| 6720 |        |        |        |        |            |                     |
| 6721 | 075744 | 000000 |        |        | T39DLY:    | .WORD 0             |
| 6723 |        | 075750 |        |        |            | .<.,10>E177770      |
| 6725 | 075750 |        |        |        | T39PACKET: |                     |
| 6726 | 075750 | 140006 |        |        |            | .WORD 140006        |
| 6727 | 075752 | 075760 |        |        |            | .WORD T39TAD        |
| 6728 | 075754 | 000000 |        |        |            | .WORD 0             |
| 6729 | 075756 | 000012 |        |        |            | .WORD 10.           |
| 6730 | 075760 |        |        |        | T39TAD:    |                     |
| 6731 | 075760 | 000    |        |        | T39BS0:    | .BYTE 0             |
| 6732 | 075761 | 000    |        |        | T39BS1:    | .BYTE 0             |

```

;BR, IF NO ERROR
;ERROR, SAVE TSSR
;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
                                TRAP      C#ERHRD
                                .WORD     1108
                                .WORD     T39NBA
                                .WORD     PKTSSR
;LOOP IF SELECTED
                                TRAP      C#CLP1
;DID LOC GET OVER WRITTEN
;BR, IF IT DIDN'T GET ETC.
;"DRIVE DOESN'T HAVE EXT TAPE STATUS
                                MOV        UNITN,-(SP)
                                MOV        #T39ETN,-(SP)
                                MOV        #2,-(SP)
                                MOV        SP,R0
                                TRAP      C#PNTX
                                ADD        #6,SP
;SKIP OVER
;"DRIVE HAS EXT TAPE STATUS"
                                MOV        UNITN,-(SP)
                                MOV        #T39ETS,-(SP)
                                MOV        #2,-(SP)
                                MOV        SP,R0
                                TRAP      C#PNTX
                                ADD        #6,SP
;BUMP DRIVE NUMBER
;AT END OF DRIVES YET
;BR, IF NO MORE DRIVES
;DO NEXT DRIVE
                                ;NEW LINE
                                MOV        #T39NFL,-(SP)
                                MOV        #1,-(SP)
                                MOV        SP,R0
                                TRAP      C#PNTX
                                ADD        #4,SP
;RETURN TO SUPERVISOR
;EXIT THIS SECTION
                                TRAP      C#EXIT
                                .WORD     L10076-.

                                ;DELAY COUNTER FOR TEST

;COMMAND PACKET FOR TEST
;WRITE SUBSYSTEM MEM. CMD. ACK,CVC*1
;ADDRESS OF CHARACTERISTICS BLOCK

;STARTING VALUE OF BLOCK SIZE
;CHARACTERISTICS DATA BLOCK
;BSEL0 BYTE
;BSEL1 BYTE

```

E3

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

SEQ 237

|                         |                                                                           |                                     |
|-------------------------|---------------------------------------------------------------------------|-------------------------------------|
| 6733 075762 000000      | T39BS2: .WORD 0                                                           | ;BSEL1 WORD                         |
| 6734 075764 000000      | .WORD 0                                                                   | ;DATA                               |
| 6735 075766             | T39BFR: .BLKW 150.                                                        | ;MESSAGE BUFFER                     |
| 6736                    |                                                                           |                                     |
| 6737                    |                                                                           |                                     |
| 6739 076450             | T39PK2: .<..10>E177770                                                    | ;COMMAND PACKET FOR TEST            |
| 6741 076450             | .WORD 140004                                                              | ;WRITE CHARA. MEM. CMND., ACK,CVC=1 |
| 6742 076450 140004      | .WORD T39DTA                                                              | ;ADDRESS OF SELECT DATA BLOCK       |
| 6743 076452 076460      | .WORD 0                                                                   |                                     |
| 6744 076454 000000      | .WORD 10.                                                                 | ;STARTING VALUE OF BLOCK SIZE       |
| 6745 076456 000012      |                                                                           |                                     |
| 6746                    |                                                                           |                                     |
| 6747                    |                                                                           |                                     |
| 6748 076460             | T39DTA:                                                                   | ;SELECT DATA BLOCK                  |
| 6749 076460 075766      | .WORD T39BFR                                                              | ;ADDRESS OF MESSAGE BUFFER          |
| 6750 076462 000000      | .WORD 0                                                                   |                                     |
| 6751 076464 000400      | .WORD 256.                                                                | ;LENGTH OF MESSAGE BUFFER           |
| 6752 076466 000000      | T39EAT: .WORD 0                                                           | ;EAI BIT WORD                       |
| 6753 076470 000000      | T39DSW: .WORD 0                                                           | ;DRIVE SELECT WORD ETC              |
| 6755 076500             | .<..10>E177770                                                            |                                     |
| 6757 076500 140012      | T39PK3: .WORD 1400'2                                                      | ;MESSAGE BUFFER RELEASE COMMAND     |
| 6758 076502 000000      | .WORD 0                                                                   | ;NOT USED                           |
| 6759                    |                                                                           |                                     |
| 6760                    | ;WRITE TAPE PACKET                                                        |                                     |
| 6761                    |                                                                           |                                     |
| 6763 076510             | .<..10>E177770                                                            |                                     |
| 6765 076510 140005      | T39PK4: .WORD 140005                                                      | ;WRITE, ACK, CVC=1 COMMAND          |
| 6766 076512 000000      | T39WR: .WORD 0                                                            | ;ADDRESS OF WRITE BUFFER            |
| 6767 076514 000000      | .WORD 0                                                                   | ;MORE ADDRESS OF WRITE BUFFER       |
| 6768 076516 000400      | T39SI2: .WORD 256.                                                        | ;SIZE OF RECORD                     |
| 6769                    |                                                                           |                                     |
| 6770                    |                                                                           |                                     |
| 6771                    |                                                                           |                                     |
| 6772                    |                                                                           |                                     |
| 6773                    |                                                                           |                                     |
| 6774                    | ;LOCAL TEXT MESSAGES FOR TEST                                             |                                     |
| 6775                    |                                                                           |                                     |
| 6776                    |                                                                           |                                     |
| 6777                    |                                                                           |                                     |
| 6778                    |                                                                           |                                     |
| 6779                    |                                                                           |                                     |
| 6780 076520 045 116 000 | T39NFL: .ASCIZ 'N'                                                        |                                     |
| 6781 076523 123 164 141 | T39NE: .ASCIZ 'Stand-alone Configuration Typeout Not Executed'            |                                     |
| 6782 076602 045 116 045 | T39ETS: .ASCIZ 'NNA Extended Tape Status Available, Drive Number #D2'     |                                     |
| 6783 076671 045 116 045 | T39ETN: .ASCIZ 'NNA Extended Tape Status NOT Available, Drive Number #D2' |                                     |
| 6784 076764 045 116 045 | T39OF2: .ASCIZ 'NNA Drive Number #D2NA Is Off-Line'                       |                                     |
| 6785 077030 045 116 045 | T39ON2: .ASCIZ 'NNA Drive Number #D2NA Is On-Line'                        |                                     |
| 6786 077073 045 116 045 | T39WRT: .ASCIZ 'NNA Drive Number #D2NA Is Write Protected'                |                                     |
| 6787 077146 045 116 045 | T39WPN: .ASCIZ 'NNA Drive Number #D2NA Is NOT Write Protected'            |                                     |
| 6788 077225 127 122 111 | T39NBA: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted'              |                                     |
| 6789 077301 103 157 156 | T39SSR: .ASCIZ 'Contents of TSSR Incorrect After WRITE SUBSYSTEM MEMORY'  |                                     |
| 6790                    |                                                                           |                                     |
| 6791 077371 045 116 045 | T39SFS: .ASCIZ 'NNA State Of Extended Features Switch ='                  |                                     |
| 6792 077443 045 116 045 | T39SBS: .ASCIZ 'NNA State Of Buffering Switch ='                          |                                     |
| 6793 077515 045 101 040 | T39OFF: .ASCIZ 'NA OFF'                                                   |                                     |
| 6794 077524 045 101 040 | T39ON: .ASCIZ 'NA ON'                                                     |                                     |
| 6795 077533 045 116 045 | T39MCL: .ASCIZ 'NNA M7196 Microcode Revision Level #02'                   |                                     |

F3

TSV5 - HARDWARE TESTS MACRO M1113 06-FEB-84 17:14  
TEST 11: CONFIGURATION TYPEOUT

SEQ 238

```

6796
6797 077610 000000      T39RL: .EVEN      0
6798                    .WORD
6799                    .EVEN
6800                    .EVEN
6801
6802                    ;+
6803                    ;LOCAL STORAGE FOR THIS TEST
6804                    ;-
6805 077612 000000      T39DAT: .WORD      0      ;LOGICAL RESPONSE TO QUESTION
6806 077614            T39REST:
6807 077614            SAVREG                    ;SAVE THE REGISTERS
6808 077620 012701 075750      MOV      #T39PACKET,R1      ;START OF THE PACKET
6809 077624 012721 140006      MOV      #140006,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1
6810 077630 012721 075760      MOV      #T39TAD,(R1)+      ;ADDRESS OF DATA BLOCK
6811 077634 005021            CLR      (R1)+      ;EXTENDED ADDRESS
6812 077636 012721 000006      MOV      #6,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
6813 077642 005021            CLR      (R1)+      ;CLEAR BSEL0 AND BSEL1
6814 077644 005021            CLR      (R1)+      ;CLEAR SEL2
6815 077646 005011            CLR      (R1)      ;CLEAR DATA AREA
6816 077650 000207      RTS      PC      ;RETURN
6817
6818                    ;+
6819                    ;LOCAL TEXT MESSAGES FOR TEST
6820                    ;-
6821
6822 077652      103      157      156      TST39ID: .ASCIZ 'Configuration Typeout'
6823
6824 077700            .EVEN
6825 077700            ENDTST
6826
6827
6828
6829
6830
6831
6832
6833
6834
6835
6836
6837
6838
6839
6840
6841
6842
6843
6844
6845
6846
6847
6848
6849
6850

```

L10076: TRAP C#ETST

.SBTTL TEST 12: SCOPE LOOPS

THIS IS A STANDALONE ROUTINE PROVIDING A NUMBER OF TIGHT "SCOPE LOOPS" USEFUL FOR DEBUGGING BASIC REGISTER ACCESS PROBLEMS WITH THE M7196 MODULE. THESE SCOPE LOOPS CAN BE USED WHEN THE NORMAL "LOOP ON ERROR" OR "LOOP ON TEST (SUBTEST)" FACILITIES DON'T SEEM TO ALLOW THE OPERATOR TO ZERO IN A PROBLEM IN THE EARLY TESTS (I.E. THE HARDWARE MAY NOT BE RESPONDING TO A REGISTER ACCESS, CAUSING A BUS ERROR TRAP, EVEN THOUGH THE DEVICE ADDRESS SELECTED BY THE PROGRAM MATCHES THE CONFIGURATION SET UP IN THE HARDWARE DIP SWITCHES). THE FOLLOWING MENU OF SCOPE LOOPS ARE AVAILABLE:

| CODE | SCOPE LOOP                     |
|------|--------------------------------|
| 0    | HELP. PRINT THIS MENU.         |
| 1    | TSBA READ ACCESS               |
| 2    | TSSR READ ACCESS               |
| 3    | INITIALIZE (TSSR WRITE ACCESS) |
| 4    | TSDB HIGH BYTE WRITE ACCESS    |

```

6851      ;
6852      ;
6853      ;
6854      ;
6855      ;
6856      ;
6857      ;
6858      ;
6859      ;
6860      ;
6861      ;
6862      ;
6863      ;
6864      ;
6865      ;
6865 077702      BGNTST
6865 077702
6870 077702      RFLAGS RO      ;GET OPERATOR FLAGS      T12::
6870 077702 104421      ;BR, IF OK TO RUN      TRAP C$RFLA
6871 077704 001403      ;"TEST NOT EXECUTED"
6872 077706 012700 101275      ;JUST EXIT IF NOT
6873 077712 000402      ;TEST ID MESSAGE
6874 077714 012700 101342      1$: MOV 0TST40ID,RO
6875 077720 004737 016510      100$: JSR PC,TSTSETUP
6876 077724 004737 020500      JSR PC,CHKMAN
6877 077730 103402      BCS 2$
6878 077732 000137 100416      JMP 64$
6879 077736 004737 015774      2$: JSR PC,SOFINIT
6880 077742 103405      BCS 5$
6881 077744 010001      MOV RO,R1
6885 077746      ERRDF ERRNO,SFIERR,SFIMSG      ;CONTENTS OF TSSR REGISTER
6885 077746 104455      ;REPORT FATAL ERROR
6885 077750 002261      ;
6885 077752 003652      ;
6885 077754 012034      ;
6886 077756 012700 100434      5$: MOV 0SCMENU,RO
6887 077762 012701 000010      MOV 08.,R1
6888 077766 004737 020256      JSR PC,GETSEL
6889 077772 005700      TST RO
6890 077774 001760      BEQ 2$
6891 077776 020027 000007      CMP RO,07
6892 100002 001015      BNE 3$
6893 100004 005737 002226      TST EXTFEA
6894 100010 001012      BNE 3$
6895 100012      PRINTF 0EXFMSG
6895 100012 012746 101217      ;MENU OF SCOPE LOOP SELECTIONS
6895 100016 012746 000001      ;MAXIMUM ALLOWED SELECTION
6895 100022 010600      ;GO GET THE OPERATOR'S SELECTION
6895 100024 104417      ;WAS ZERO SPECIFIED ?
6895 100026 062706 000004      ;REPEAT MENU IF YES.
6896 100032 000137 077736      ;EXTENDED TSSR ?
6897 100036 010004      ;BRANCH IF NOT
6898 100040      ;CHECK FOR EXTENDED FEATURES SET
6898 100040 012700 000340      ;BR, IF IT IS ON
6898 100044 104441      ;WARN OPERATOR EXTENDED FEATURES CLEAR
6899 100046 005037 100426      MOV 0EXFMSG,-(SP)
6900 100052 032737 000100 177560      MOV 01,-(SP)
6901 100060 001005      MOV SP,RO
6901 100060 001005      TRAP C$PNTF
6901 100060 001005      ADD 04,SP
6901 100060 001005      ;GO BACK TO BASIC MENU
6901 100060 001005      ;SAVE THE MENU SELECTION
6901 100060 001005      ;RAISE THE PRIORITY
6901 100060 001005      MOV 0PRI07,RO
6901 100060 001005      TRAP C$SPRI
6901 100060 001005      ;ASSUME INTERRUPTS ARE ENABLED
6901 100060 001005      ;ARE TTI INTERRUPTS ON ?
6901 100060 001005      ;BRANCH IF YES

```

| Address | Hex | Dec | Label | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416 | Op417 | Op418 | Op419 | Op420 | Op421 | Op422 | Op423 | Op424 | Op425 | Op426 | Op427 | Op428 | Op429 | Op430 | Op431 | Op432 | Op433 | Op434 | Op435 | Op436 | Op437 | Op438 | Op439 | Op440 | Op441 | Op442 | Op443 | Op444 | Op445 | Op446 | Op447 | Op448 | Op449 | Op450 | Op451 | Op452 | Op453 | Op454 | Op455 | Op456 | Op457 | Op458 | Op459 | Op460 | Op461 | Op462 | Op463 |  |
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|
|---------|-----|-----|-------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--|

TSV5 - HARDWARE TESTS    MACRO M1113    06-FEB-84 17:14  
TEST 12: SCOPE LOOPS

SEQ 241

```

6957
6958 100270 004737 020174      35$: JSR    PC,GETPAT      ;READ THE DATA PATTERN
6959 100274 010001              MOV    R0,R1      ;DATA PATTERN FOR LOOP
6960 100276 012703 000000      MOV    @TSD8,R3    ;SELECT TSD8
6961 100302 060503              ADD    R5,R3      ;POINT TO TSV05'S REGISTERS
6962 100304 010113      37$: MOV    R1,(R3)      ;WRITE THE DATA PATTERN
6963
6964 100306 000776              BR     37$          ;LOOP UNTIL HALTED
6965
6966 100310 004737 020174      40$: JSR    PC,GETPAT      ;READ THE DATA PATTERN
6967 100314 010001              MOV    R0,R1      ;SAVE THE DATA PATTERN
6968 100316 012703 000003      MOV    @TSSRH,R3    ;BYTE ADDRESS OF TSSR, HIGH BYTE
6969 100322 060503              ADD    R5,R3      ;POINT TO TSV05'S REGISTERS
6970 100324 110113      42$: MOV8   R1,(R3)      ;WRITE THE DATA TO REGISTER
6971 100326 000776              BR     42$          ;LOOP UNTIL HALTED
6972
6973
6974
6975      ;*
6976      ;PROCESS CONSOLE INTERRUPTS
6977      ;-
6978 100330 010046              60$: MOV    R0,-(SP)      ;SAVE WORK REGISTER
6979 100332 113700 177562      MOV8   @TTIBFR,R0    ;GET THE OPERATOR INPUT
6980 100336 042700 000200      BIC    @200,R0    ;STRIP OFF PARITY BIT
6981 100342 122700 000015      CMPB   @15,R0      ;IS IT A CARRIAGE RETURN ?
6982 100346 001021              BNE    61$          ;JUST EXIT IF NOT
6983 100350 012766 077736 000002      MOV    @2$,2(SP)    ;RETURN TO MASTER MENU
6984 100356 005066 000004      CLR     4(SP)      ;FORCE PRIORITY ZERO
6985 100362 013737 100430 000060      MOV    TVECSAV,@TTIVEC    ;RESTORE SUPERVISOR VECTOR
6986 100370 013737 100432 000062      MOV    TPRISAV,@TTIVEC+2  ;RESTORE SUPERVISOR PRIORITY
6987 100376 005737 100426      TST     TTION      ;ARE SUPERVISOR INTERRUPTS ENABLED ?
6988 100402 001403              BEQ    61$          ;BRANCH IF YES
6989 100404 042737 000100 177560      BIC    @100,@TTICSR    ;TURN OFF TTI INTERRUPTS
6990 100412 012600      61$: MOV    (SP)+,R0      ;RESTORE REGISTER
6991 100414 000002              RTI                ;RETURN FROM INTERRUPT
6992
6993 100416
6994 100416      64$:
6995 100416 104432      63$: EXIT    TST                ;EXIT THE TEST
6996 100420 000736
6997 100422 000137 000200      65$: JMP     200                ;RETURN TO SUPERVISOR
6998
6999      ;*
7000      ;LOCAL STORAGE FOR THIS TEST
7001      ;-
7001 100426 000000      TTION:      .WORD    0      ;WORD SET IF SUPERVISOR TTI INTER OFF
7002 100430 000000      TVECSAV:   .WORD    0      ;SAVE TTI VECTOR
7003 100432 000000      TPRISAV:   .WORD    0      ;SAVE TTI PRIORITY
7004
7005
7006      ;*
7007      ;MENU FOR OPERATOR INPUT FOR SCOPE LOOPS
7008      ;-
7009
7010      .EVEN
7011 100434 100466 100541 100567 SCMENU: .WORD    1$,2$,3$,4$,5$,6$

```

J3

TSV5 - HARDWARE TESTS    MACRO M1113    06-FEB-84 17:14  
TEST 12: SCOPE LOOPS

SEQ 242

```

7012 100450 100740 100776 101044      .WORD  7$,8$,9$,10$,11$,12$,0
7013
7014
7015 100466      012      123      105  1$:  .ASCIZ  <12>'SELECT SCOPE LOOP FROM FOLLOWING OPTIONS:'
7016 100541      012      011      060  2$:  .ASCIZ  <12>'  0      Display This Menu'
7017 100567      011      061      011  3$:  .ASCIZ  '      1      TSBA Read Access'
7018 100613      011      062      011  4$:  .ASCIZ  '      2      TSSR Read Access'
7019 100637      011      063      011  5$:  .ASCIZ  '      3      Initialize (TSSR Write Access)'
7020 100701      011      064      011  6$:  .ASCIZ  '      4      TSDB High Byte Write Access'
7021 100740      011      065      011  7$:  .ASCIZ  '      5      TSDB Low Byte Write Access'
7022 100776      011      066      011  8$:  .ASCIZ  '      6      TSDR Maintenance Mode Write Access'
7023 101044      011      067      011  9$:  .ASCIZ  '      7      TSDBX (TSSR High Byte) Write Access'
7024 101113      011      070      011 10$:  .ASCIZ  '      8      Return to Diagnostic Supervisor'
7025 101156      000              11$:  .ASCIZ  ''
7026 101157      124      171      160 12$:  .ASCIZ  'Type RETURN To Stop Scope Loops'
7027 101217      045      116      045 EXFMSG: .ASCIZ  '*** Extended Features Switch Not On *** '
7028 101275      123      164      141 T4ONE: .ASCIZ  'Stand-alone Scope Loops Not Executed'
7029 101342      123      143      157 TST40ID: .ASCIZ  'Scope Loops'
7030
7031 101356      .EVEN
      101356      ENDTST
      101356      L10077: TRAP      C$ETST
7032 101360      104401      ENDMOD

```

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
 TEST 12: SCOPE LOOPS

SEQ 243

```

1          .TITLE  TSV6 - PARAMETER CODING
7
12
18
19 101360      BGNMOD  TSV6
    101360      TSV6::
20
21          .SBTTL  HARDWARE PARAMETER CODING SECTION
22
23          ;**
24          ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
25          ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
26          ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
27          ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
28          ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
29          ; WITH THE OPERATOR.
30          ;**
31 101360      BGNHRD
    101360      .WORD L10100-L$HARD/2
    101362      L$HARD::
32
33 101362      GPRMA  HPM1,0,0,160010,177776,YES      ;GET TSBA/TSDB REGISTER ADDRESS.
    101362      .WORD  T$CODE
    101364      .WORD  HPM1
    101366      .WORD  T$LOLIM
    101370      .WORD  T$HILIM
34 101372      GPRMA  HPM2,2,0,0,776,YES              ;GET VECTOR ADDRESS.
    101372      .WORD  T$CODE
    101374      .WORD  HPM2
    101376      .WORD  T$LOLIM
    101400      .WORD  T$HILIM
35          .GPRMD  HPM3,4,0,340,0,7,YES              ;GET INTERRUPT PRIORITY.
36 101402      ENDRD
    101402      .EVEN
    101402      L10100:
37 101402      104      105      126  HPM1:  .ASCIZ  'DEVICE ADDRESS (TSBA/TSDB) '
38 101436      111      116      124  HPM2:  .ASCIZ  'INTERRUPT VECTOR '
39 101462      111      116      124  HPM3:  .ASCIZ  'INTERRUPT PRIORITY '
40          .EVEN

```

```

42          .SBTTL  SOFTWARE PARAMETER CODING SECTION
43
44          ;
45          ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
46          ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
47          ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
48          ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
49          ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
50          ; WITH THE OPERATOR.
51          ;--
52 101512      BGNSFT
53      101512  000003      .WORD L10101-L$SOFT/2
54      101514
55          L$SOFT::
56          ; GPRML  SPM1,0,-1,YES      ; GET TRANSPORT TEST FLAG.
57          ; GPRML  SPM4,2,-1,YES      ; GET ITERATION CONTROL.
58      101514  001130      .WORD  T$CODE
59      101516  101552      .WORD  SPM4
60      101520  177777      .WORD  -1
61          ; GPRMD  SPM6,4,D,7777,0,7777,YES      ; GET LOCAL ERROR LIMIT
62          ; GPRMD  SPM7,6,D,7777,0,7777,YES      ; GET GLOBAL ERROR LIMIT
63          ENDSFT
64          .EVEN
65          L10101:
66 101522      SPM1:  .ASCIZ  'ENABLE TRANSPORT TESTS '
67 101552      SPM4:  .ASCIZ  'INHIBIT ITERATIONS '
68 101602      SPM6:  .ASCIZ  'PER TEST ERROR LIMIT '
69 101632      SPM7:  .ASCIZ  'PER UNIT ERROR LIMIT '
70          .SBTTL  PATCH AREA
71
72          ;
73          ; FINALLY A GENEROUS PATCH AREA.
74          ;
75          ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
76          ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
77          ;
78          PATCH::
79          .BLKW  32.
80          . = !377+1
81          LASTAD      ;SET LAST USED ADDRESS.
82          .EVEN
83          .WORD  0
84          .WORD  0
85          L$LAST::
86          ENDMOD
87          .END

```

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
SYMBOL TABLE

SEQ 245

|                 |                 |                  |                 |                 |
|-----------------|-----------------|------------------|-----------------|-----------------|
| ADDSR 012126 G  | C\$AU 000052    | DEVDR0 023376    | FREEHI 003130   | INTCPC 016150   |
| ADR 000020 G    | C\$AUTO 000061  | DEVNRD 023315    | FRESIZ 003126 G | INTFLA 016145   |
| AMBTSS 006635   | C\$BRK 000022   | DEVNXR 023233    | FUSI 004117     | INTMAS 016144   |
| ASSEMB 000010   | C\$BSEG 000004  | DEVONL 023163    | F\$AU 000015    | INTR 016216 G   |
| A1716 000003    | C\$BSUB 000002  | DEVSUM 023126    | F\$AUTO 000020  | INTREC 002224 G |
| BADDAT 003156 G | C\$CEFG 000045  | DFPTBL 002156 G  | F\$BGN 000040   | INTVEC 016146   |
| BADSSR 015700 G | C\$CLCK 000062  | DIAGMC 000000    | F\$CLEA 000007  | INTX 004300     |
| BDVPCR 177520 G | C\$CLEA 000012  | DICEB 000001     | F\$DU 000016    | INVERT 021206 G |
| BENBSW 002230 G | C\$CLOS 000035  | DSBINT 016204    | F\$END 000041   | IOKCKI 000200   |
| BIE 040000      | C\$CLP1 000006  | DUAD12 004643    | F\$HAR 000004   | IOKSTP 000001   |
| BIT0 000001 G   | C\$CVEC 000036  | DUFLG 003112 G   | F\$HW 000013    | IPRI 002212 G   |
| BIT00 000001 G  | C\$DCLN 000044  | DUMMY 003062     | F\$INIT 000006  | ISR 000100 G    |
| BIT01 000002 G  | C\$DODU 000051  | EF.CON 000036 G  | F\$JMP 000050   | IVEC 002210 G   |
| BIT02 000004 G  | C\$DRPT 000024  | EF.NEW 000035 G  | F\$MOD 000000   | IXE 004000 G    |
| BIT03 000010 G  | C\$DU 000053    | EF.PWR 000034 G  | F\$MSG 000011   | I\$AU 000041    |
| BIT04 000020 G  | C\$EDIT 000003  | EF.RES 000037 G  | F\$PROT 000021  | I\$AUTO 000041  |
| BIT05 000040 G  | C\$ERDF 000055  | EF.STA 000040 G  | F\$PWR 000017   | I\$CLN 000041   |
| BIT06 000100 G  | C\$ERHR 000056  | EMAXDU 016777    | F\$RPT 000012   | I\$DU 000041    |
| BIT07 000200 G  | C\$ERRO 000060  | EN 000000        | F\$SEG 000003   | I\$HRD 000041   |
| BIT08 000400 G  | C\$ERSF 000054  | ENAINI 016152    | F\$SOFT 000005  | I\$INIT 000041  |
| BIT09 000100 G  | C\$ERSO 000057  | ENVIRN 020630    | F\$SRV 000010   | I\$MOD 000041   |
| BIT1 000002 G   | C\$ESCA 000010  | EPRTSW 002200 G  | F\$SUB 000002   | I\$MSG 000041   |
| BIT10 002000 G  | C\$ESEG 000005  | EPRT1 006360     | F\$SW 000014    | I\$PROT 000040  |
| BIT11 004000 G  | C\$ESUB 000003  | EPRT2 006360     | F\$TEST 000001  | I\$PTAB 000041  |
| BIT12 010000 G  | C\$ETST 000001  | ERCM 011733      | GDDAT 003160 G  | I\$PWR 000041   |
| BIT13 020000 G  | C\$EXIT 000032  | ERRHI 002236 G   | GERRMA 002174 G | I\$RPT 000041   |
| BIT14 040000 G  | C\$GETB 000026  | ERRK 016756      | GETPAT 020174 G | I\$SEG 000041   |
| BIT15 100000 G  | C\$GETW 000027  | ERRI 0 002240 G  | GETSEL 020256 G | I\$SETU 000041  |
| BIT2 000004 G   | C\$GMAN 000043  | ERRNO 002261     | G\$CNT0 000200  | I\$SFT 000041   |
| BIT3 000010 G   | C\$GPHR 000042  | ERRVEC 000004 G  | G\$DELM 000372  | I\$SRV 000041   |
| BIT4 000020 G   | C\$GPLO 000030  | ERTABE 003376    | G\$DISP 000003  | I\$SUB 000041   |
| BIT5 000040 G   | C\$GPRI 000040  | ERTABL 003176    | G\$EXCP 000400  | I\$TST 000041   |
| BIT6 000100 G   | C\$INIT 000011  | ESUM 016760      | G\$HILI 000002  | J\$JMP 000167   |
| BIT7 000200 G   | C\$INLP 000020  | EVL 000004 G     | G\$LOLI 000001  | KIPAR0 172340   |
| BIT8 000400 G   | C\$MANI 000050  | EXBCNT 000010    | G\$NO 000000    | KIPAR1 172342   |
| BIT9 001000 G   | C\$MEM 000031   | EXFMSG 101217    | G\$OFFS 000400  | KIPAR2 172344   |
| BOE 000400 G    | C\$MSG 000023   | EXPBRE 015502 G  | G\$OFST 000376  | KIPAR3 172346   |
| BRINIT 004457   | C\$OPEN 000034  | EXPD 002232 G    | G\$PRMA 000001  | KIPAR4 172350   |
| BSEL0 000000    | C\$PNTB 000014  | EXPGOT 004533    | G\$PRMD 000002  | KIPAR5 172352   |
| BSEL1 000001    | C\$PNTF 000017  | EXPGT2 004567    | G\$PRML 000000  | KIPAR6 172354   |
| CHKAMB 016044   | C\$PNTS 000016  | EXPMMSG 002322 G | G\$RADA 000140  | KIPAR7 172356   |
| CHKMAN 020500 G | C\$PNTX 000015  | EXPREC 015474 G  | G\$RADB 000000  | KIPDR0 172300   |
| CHKTSS 016336   | C\$QIO 000377   | EXTA 005772      | G\$RADD 000040  | KIPDR1 172302   |
| CKDROP 017202   | C\$RBU 000007   | EXTEND 005770    | G\$RADL 000120  | KIPDR2 172304   |
| CKEMAX 017102   | C\$REFG 000047  | EXTFEA 002226 G  | G\$RADO 000020  | KIPDR3 172306   |
| CKMSG 011360 G  | C\$RESE 000033  | E\$END 002100    | G\$XFER 000004  | KIPDR4 172310   |
| CKMSG2 011500 G | C\$REVI 000003  | E\$LOAD 000035   | G\$YES 000010   | KIPDR5 172312   |
| CKRAM 011114 G  | C\$RFLA 000021  | FATERR 000060    | HIADDR 001400   | KIPDR6 172314   |
| CKRAM2 011224 G | C\$RPT 000025   | FATFLG 002222 G  | HOE 100000 G    | KIPDR7 172316   |
| CMOPKT 021260 G | C\$SEFG 000046  | FERCM 011722     | HPM1 101402     | KTENAB 003134 G |
| CMPMEM 017660   | C\$SPRI 000041  | FIFEXP 012170 G  | HPM2 101436     | KTFLG 003132 G  |
| CONFIG 017250   | C\$SVEC 000037  | FIF1MS 012242    | HPM3 101467     | KTINIT 021054   |
| COUNT 002310 G  | C\$TPRI 000013  | FIF2MS 012311    | IBE 010000 G    | KTOFF 017274    |
| CSRADD 002206 G | DATA 002312 G   | FILLME 017422    | IDU 000040 G    | KTON 017256     |
| CTAB 003164 G   | DATASC 020232   | FNOINT 004215    | IER 020000 G    | KERRMA 002172 G |
| CTABE 003176 G  | DEBUGM 011632   | FORCER 002176 G  | IFAU 004256     | KERRNO 000000   |
| CTABM 003164 G  | DEVCNT 002220 G | FREE 003124 G    | INCERK 017044   | LISTN 000000    |

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
SYMBOL TABLE

SEQ 246

|                  |                  |                 |                  |                 |
|------------------|------------------|-----------------|------------------|-----------------|
| LOE = 040000 G   | L\$UNIT 002012 G | L10071 056530   | OFL = 000100     | PRMNO 002320 G  |
| LOOPCN 002216 G  | L10000 002164    | L10072 057774   | ONEFIL = 000000  | PRMSG0 014552 G |
| LOOPCO 013126    | L10001 002176    | L10073 066664   | O\$APTS = 000000 | PRMSG1 014732   |
| LOOPFL 003162 G  | L10002 005766    | L10074 065024   | O\$AU = 000001   | PRMSG2 015035   |
| LOT = 000010 G   | L10003 012044    | L10075 074526   | O\$BGNR = 000001 | PRASC 014420    |
| L\$ACP 002110 G  | L10004 012062    | L10076 077700   | O\$BGNS = 000001 | PR1ASC 014465   |
| L\$APT 002036 G  | L10005 012100    | L10077 101356   | O\$DU = 000001   | PST32W 003152 G |
| L\$AU 022352 G   | L10006 012106    | L10100 101402   | O\$ERRT = 000000 | PUNIT 022304    |
| L\$AUT 002070 G  | L10007 012124    | L10101 101522   | O\$GNSW = 000001 | PW.D11 = 000021 |
| L\$AUTO 022556 G | L10010 012142    | MEMADD 013754 G | O\$POIN = 000001 | PW.D13 = 000022 |
| L\$CCP 002106 G  | L10011 012166    | MEMCK 021276 G  | O\$SETU = 000000 | PW.D22 = 000020 |
| L\$CLEA 022636 G | L10012 012240    | MENASC 020447   | PASRPT 022054    | PW.NOP = 000000 |
| L\$CO 002032 G   | L10013 012410    | MENERR 020374   | PATCH 101662 G   | PW.NO1 = 000023 |
| L\$DEPO 002011 G | L10014 013124    | MENRES 020476   | PATDAT 020230    | PW.RDE = 000024 |
| L\$DESC 003410 G | L10015 013752    | MIMENU 073412   | PC.ERA = 002400  | PW.RDR = 000001 |
| L\$DESP 002076 G | L10016 013774    | MMI EC = 000250 | PC.IER = 002000  | PW.RDS = 000005 |
| L\$DEVP 002060 G | L10017 015500    | MSA.FR = 000006 | PC.NOO = 001000  | PW.RFI = 000003 |
| L\$DISP 002124 G | L10020 015506    | MSA.NO = 000000 | PC.REL = 000000  | PW.WCT = 000006 |
| L\$DI.Y 002116 G | L10021 015514    | MSA.NR = 000004 | PC.REW = 000400  | PW.WFI = 000004 |
| L\$DTP 002040 G  | L10022 015526    | MSA.VO = 000002 | PKBCNT = 000006  | PW.WFM = 000007 |
| L\$DTYP 002034 G | L10023 015550    | MSGEXP 012144 G | PKHI = 000004    | PW.WMI = 000010 |
| L\$DU 022450 G   | L10024 015576    | MSGLOO 013064 G | PKLOW = 000002   | PW.WNP = 000011 |
| L\$DUT 002072 G  | L10025 015736    | MSGSTA 012350 G | PKTADD 007554    | PW.WTR = 000002 |
| L\$DVTY 003402 G | L10026 016246    | MSGSUB 013742 G | PKTFRM 007516    | P.ACK = 100000  |
| L\$EF 002052 G   | L10030 022302    | MS.ATT = 000006 | PKTGET 012064 G  | P.CMD = 000037  |
| L\$ENVI 002044 G | L10031 022446    | MS.EXT = 000200 | PKTMES 012110 G  | P.CONT = 000012 |
| L\$ETP 002102 G  | L10032 022554    | MS.RSD = 000001 | PKTRAM 004745 G  | P.CVC = 040000  |
| L\$EXP1 002046 G | L10033 022634    | MS.RSF = 000020 | PKTSSR 012046 G  | P.FMT = 000140  |
| L\$EXP4 002064 G | L10034 022662    | MS.RST = 000010 | PNT = 001000 G   | P.FORM = 000011 |
| L\$EXP5 002066 G | L10035 023124    | M8186 005554    | PRAMPK 013776    | P.GETS = 000017 |
| L\$HARD 101362 G | L10036 024424    | M8189 005645    | PRASC 014523     | P.IE = 000200   |
| L\$HIME 002120 G | L10037 026416    | NBA = 002000    | PRBEXP 015470    | P.INIT = 000013 |
| L\$HPCP 002016 G | L10040 024700    | NEWPAS 022010   | PRBM3G 015336    | P.MODE = 007400 |
| L\$HPTP 002022 G | L10041 025144    | NODEV 003114 G  | PRBREC 015472    | P.OPP = 020000  |
| L\$HW 002156 G   | L10042 032012    | NOEXTF 030206   | PRBTOT 015423    | P.POSI = 000010 |
| L\$ICP 002104 G  | L10043 026772    | NOINIT 004335   | PRBYTE 015122 G  | P.READ = 000001 |
| L\$INIT 021556 G | L10044 027262    | NOINTR 004221   | PRI = 002000 G   | P.SWB = 010000  |
| L\$LADP 002026 G | L10045 027560    | NUITS 002170 G  | PRIADD 010160    | P.WRIT = 000005 |
| L\$LAST 102004 G | L10046 030212    | NOMAN 020534    | PRIAO 010230     | P.WRTC = 000004 |
| L\$LOAD 002100 G | L10047 034602    | NOMEM 005460    | PRIBXO 007612 G  | P.WRTS = 000006 |
| L\$LUN 002074 G  | L10050 032346    | NP.IR = 000200  | PRIEQU 010060    | QVP 002204 G    |
| L\$MREV 002050 G | L10051 032740    | NP.LOO = 000040 | PRIPKT 007370 G  | RAMASC 014156   |
| L\$NAME 002000 G | L10052 033346    | NP.OUT = 000100 | PRIRAM 010066    | RAMDAT 002242 G |
| L\$PRIO 002042 G | L10053 040374    | NP.WRP = 000020 | PRITAD 010274    | RAMERR 015510 G |
| L\$PROT 021546 G | L10054 035666    | NSI 004152      | PRITSS 006024    | RAMEXP 015530 G |
| L\$PRT 002112 G  | L10055 036620    | NSINIT 004407   | PRITO 010356     | RAMFOR 010116   |
| L\$REPP 002062 G | L10056 050506    | NUL 004527      | PRIT1 010421     | RAMSIZ 002302 G |
| L\$REV 02010 G   | L10057 040700    | NULCR 004530    | PRIXOR 007742 G  | RAMTAD 015516 G |
| L\$RPT 022664 G  | L10060 042110    | NXN = 004000    | PRI00 = 000000 G | RCVHIA 002304 G |
| L\$SOFT 101514 G | L10061 043450    | NXNFIG 003136 G | PRI01 = 000040 G | RCVLOA 002306 G |
| L\$SFC 002056 G  | L10062 044026    | NXNHI 003142 G  | PRI02 = 000100 G | RDERR 005206    |
| L\$SPCP 002020 G | L10063 045300    | NXNLO 003140 G  | PRI03 = 000140 G | RECMSG 002466 G |
| L\$SPTP 002024 G | L10064 046344    | IXMTSY 021452   | PRI04 = 000200 G | RECV 002234 G   |
| L\$STA 002034 G  | L10065 051766    | NXR 003740      | PRI05 = 000240 G | REGSAV 020140   |
| L\$SW 002034 G   | L10066 062614    | NXRERR 005736 G | PRI06 = 000300 G | RETErr 005372   |
| L\$TEST 002034 G | L10067 053772    | NXRX 003777     | PRI07 = 000340 G | REWIND 011014 G |
| L\$TML 002034 G  | L10070 055260    | NXTU 022022     | PRMESS 014242    |                 |

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
SYMBOL TABLE

SEQ 247

|         |          |         |          |          |          |        |          |        |        |
|---------|----------|---------|----------|----------|----------|--------|----------|--------|--------|
| RMCHBE= | 000167   | S1.IID= | 004000   | TST40I   | 101342   | T10    | 066666 G | T158FR | 033412 |
| RMCHEN= | 000200   | S1.I1R= | 020000   | TSV2     | 002000 G | T11    | 074530 G | T158F2 | 034100 |
| RMMSG8= | 000215   | S1.I2R= | 040000   | TSV3     | 002176 G | T12    | 077702 G | T158S0 | 034100 |
| RMMSG8= | 000234   | S1.PAR= | 100000   | TSV4     | 021546 G | T128FR | 030302   | T158S1 | 034101 |
| RMPKTB= | 000201   | S2.ATI= | 000010   | TSV5     | 023446 G | T128KG | 030757   | T158AT | 033400 |
| RMPKTE= | 000210   | S2.BTI= | 000004   | TSV6     | 101360 G | T128LK | 030334   | T15L00 | 032044 |
| RMR     | 010000   | S2.DIM= | 000200   | TTI8FR=  | 177562 G | T12CHA | 031750   | T15PAC | 033370 |
| RWPACK  | 011110   | S2.ILW= | 000100   | TTICSR=  | 177560 G | T12CKR | 031530 G | T15PK2 | 034070 |
| SC      | 100000   | S2.INR= | 000020   | TTION    | 100426   | T12CON | 031336   | T15RES | 034472 |
| SCE     | 020000   | S2.OUT= | 000040   | TTION2   | 071454   | T12DAT | 030270   | T15RT2 | 034544 |
| SCHERR  | 005300   | S2.UND= | 000003   | TTIVEC=  | 000060 G | T12DPR | 031136   | T15SSR | 034106 |
| SCME    | 005013   | TBLEND= | 003062 G | TVECSA   | 100430   | T12GET | 030516   | T15S2  | 034102 |
| SCMENU  | 100434   | TCOASC  | 006476   | TVSAV2   | 071456   | T12HIA | 030322   | T15S3  | 034104 |
| SDELAY  | 010660   | TCOCOD  | 006676   | T1ARGC=  | 000001   | T12KT  | 030330   | T168EN | 040260 |
| SELASC  | 020442   | TEMP1   | 003116 G | T1CODE=  | 001130   | T12LOA | 030324   | T168FR | 040232 |
| SELDAT= | 000004   | TEMP2   | 003120 G | T1ERRN=  | 002261   | T12L00 | 026446   | T168FS | 040252 |
| SEL2    | 000002   | TERCLS= | 000016   | T1EXCP=  | 000000   | T12MSG | 030661   | T16CLR | 040076 |
| SETMAP  | 017316   | TESTNO= | 000014   | T1FLAG=  | 000040   | T12NIN | 031045   | T16DAT | 040220 |
| SETU    | 022106   | TEXASC  | 006435   | T1GMAN=  | 000000   | T12NXM | 031227   | T16DT2 | 040270 |
| SFFMSG  | 012102 G | TFCASC  | 006537   | T1HILI=  | 000776   | T12PAC | 030260   | T16D01 | 036652 |
| SFHERR  | 003705   | TIMEXP  | 015552 G | T1LAST=  | 000001   | T12PAR | 030326   | T16D28 | 036654 |
| SFIERR  | 003652   | TIMSGO  | 015600   | T1LOLI=  | 000000   | T12SET | 031706   | T16D53 | 036656 |
| SFIMSG  | 012034 G | TINERR  | 012021   | T1LSYM=  | 010000   | T12SWR | 031640   | T16D78 | 036660 |
| SFPTBL  | 002166 G | TMPBFR  | 002632 G | T1LTNO=  | 000014   | T12TBE | 030466   | T16LEN | 037202 |
| SIFLAG  | 003154 G | TNAM    | 016704   | T1NEST=  | 177777   | T12WRT | 030572   | T16L00 | 034622 |
| SIMSG   | 011766   | TPRISA  | 100432   | T1NS0 =  | 000000   | T121LO | 026560   | T16PAC | 040210 |
| SKIPT   | 003400   | TPSAV2  | 071460   | T1NS1 =  | 000005   | T122LO | 027114   | T16PK2 | 040260 |
| SOFINI  | 015774 G | TRANST  | 002166 G | T1NS2 =  | 000002   | T123LO | 027334   | T16REJ | 037314 |
| SPACE   | 010466 G | TSBA =  | 000000 G | T1NS3 =  | 000003   | T124LO | 027756   | T16RES | 040030 |
| SPM1    | 101522   | TSBAH = | 000001 G | T1PTNU=  | 000000   | T124TS | 030332   | T16SEX | 040170 |
| SPM4    | 101552   | TSDB =  | 000000 G | T1SAVL=  | 177777   | T138FR | 024062   | T16SRD | 040122 |
| SPM6    | 101602   | TSDBH = | 000001 G | T1SEGL=  | 177777   | T13DAT | 024050   | T16SSR | 036732 |
| SPM7    | 101632   | TSFCOD  | 007236   | T1SEKO=  | 010000   | T13LOO | 023464   | T16T58 | 037100 |
| SR0     | 177572   | TSREJ = | 000006   | T1SUBN=  | 000000   | T13MEM | 024155   | T16T01 | 037431 |
| SR1     | 177574   | TSSDEF  | 006606   | T1TAGL=  | 177777   | T13NBA | 024214   | T16T28 | 037530 |
| SR2     | 177576   | TSSR =  | 000002 G | T1TAGN=  | 010102   | T13PAC | 024040   | T16T53 | 037630 |
| SR3     | 172516   | TSSRBI  | 003502 G | T1TEMP=  | 000000   | T13RES | 024356   | T16T78 | 037730 |
| SSR     | 000200   | TSSRFO  | 006415   | T1TEST=  | 000014   | T13SSR | 024267   | T16MMI | 040142 |
| STATCO  | 012412   | TSSRH = | 000003 G | T1TSTM=  | 177777   | T148FR | 025176   | T162SS | 036767 |
| SVCGBL= | 000000   | TSSX    | 004020   | T1TSTS=  | 000001   | T14BS0 | 025170   | T163SS | 037033 |
| SVCINS= | 000000   | TSTBLK  | 002752 G | T1TAU =  | 010031   | T14BS1 | 025171   | T178EN | 050362 |
| SVCSUB= | 000001   | TSTCNT  | 002214 G | T1TAUT=  | 010033   | T14BS2 | 025172   | T178FR | 050242 |
| SVCTAG= | 000000   | TSTEND  | 016720   | T1TCLE=  | 010034   | T14DAT | 025170   | T178FS | 050262 |
| SVCTST= | 000001   | TSTFLA  | 002314 G | T1TDU =  | 010032   | T14DTA | 025610   | T17CLE | 050146 |
| S1LSYM= | 010000   | TSTL00  | 016456 G | T1THAR=  | 010100   | T14LOO | 024444   | T17CLR | 047760 |
| S0.IDB= | 000010   | TSTPTR  | 002316 G | T1THW =  | 010000   | T14NBA | 025622   | T17DAT | 050230 |
| S0.IFB= | 000002   | TSTSET  | 016510 G | T1THNI=  | 010030   | T14NIN | 026063   | T17DT2 | 050400 |
| S0.IFP= | 000001   | TST12I  | 030470   | T1THMSG= | 010025   | T14PAC | 025160   | T17EXE | 046524 |
| S0.ILD= | 000020   | TST13I  | 024102   | T1THPRO= | 010027   | T14PK2 | 025600   | T17EXP | 046402 |
| S0.ION= | 000040   | TST14I  | 026241   | T1THRPT= | 010035   | T14RES | 026306   | T17EXS | 046422 |
| S0.IRD= | 000100   | TST15I  | 034453   | T1HSEG=  | 010000   | T14RST | 026344   | T17LOO | 040424 |
| S0.IRW= | 000004   | TST16I  | 036662   | T1HSHF=  | 010101   | T14SSR | 025773   | T17MSK | 046376 |
| S0.ISP= | 000200   | TST17I  | 046626   | T1HSHV=  | 010026   | T14TSB | 026155   | T17PAC | 050220 |
| S1.ICE= | 002000   | TST18I  | 051216   | T1HSHB=  | 010074   | T142RE | 025676   | T17PK2 | 050370 |
| S1.IEO= | 010000   | TST19I  | 060202   | T1HSHW=  | 010001   | T15AM2 | 034162   | T17RFI | 050126 |
| S1.IFM= | 001000   | TST20I  | 065202   | T1HSHS=  | 010077   | T15AM3 | 034263   | T17RSF | 050024 |
| S1.IHE= | 000400   | TST39I  | 077652   | T1       | 023446 G | T15AMA | 034365   | T17SET | 050166 |

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
SYMBOL TABLE

SEQ 248

|        |        |        |          |        |          |        |          |        |          |
|--------|--------|--------|----------|--------|----------|--------|----------|--------|----------|
| T17SNP | 050046 | T19SSR | 060243   | T3     | 026420 G | T39NBA | 077225   | WF.IED | 000010   |
| T17SRD | 050004 | T19WCT | 062142   | T3BFLG | 003150 G | T39NE  | 076523   | WF.IER | 000004   |
| T17SSR | 046645 | T19WFI | 062106   | T3.1   | 026446   | T39NFL | 076520   | WF.IHI | 000200   |
| T17WFD | 046524 | T19WFM | 062162   | T3.2   | 027006   | T39OFF | 077515   | WF.IRE | 000040   |
| T17WFI | 050072 | T19WST | 061543   | T3.3   | 027276   | T39OF2 | 076764   | WF.IWF | 000020   |
| T171CM | 047165 | T191CM | 060700   | T3.4   | 027574   | T39ON  | 077524   | WF.IWR | 000100   |
| T172CM | 047247 | T192CM | 060762   | T3ASC  | 074461   | T39ON2 | 077030   | WF.I3R | 000002   |
| T172SS | 046702 | T192SS | 060300   | T3ASN  | 074500   | T39PAC | 075750   | WF.I4R | 000001   |
| T173CM | 047343 | T193CM | 061056   | T3ASO  | 074356   | T39PK2 | 076450   | WRTCHR | 010662 G |
| T173SS | 046746 | T193SS | 060344   | T3AS1  | 074423   | T39PK3 | 076500   | WRTERR | 005113   |
| T174CM | 047427 | T194SS | 060411   | T3BFR  | 071506   | T39PK4 | 076510   | WRTMSG | 005056   |
| T174SS | 047013 | T195CM | 061144   | T3BSSO | 071500   | T39RES | 077614   | WSMBK  | 021270 G |
| T175CM | 047512 | T195SS | 060454   | T3BSS1 | 071501   | T39RL  | 077610   | XFERAS | 015740   |
| T175SS | 047056 | T196CM | 061220   | T3BSS2 | 071502   | T39SBS | 077443   | XNXM   | 016376   |
| T176CM | 047566 | T196SS | 060520   | T3CNT  | 074524   | T39SFS | 077371   | XORBFO | 007674   |
| T176SS | 047122 | T197CM | 061303   | T3DAT  | 074014   | T39SI2 | 076516   | XORFOR | 010012   |
| T177CM | 047674 | T197SS | 060563   | T3DLY  | 071462   | T39SSR | 077301   | XST0   | 000006 G |
| T18BFR | 051722 | T198CM | 061371   | T3DSW  | 072210   | T39TAD | 075760   | XST1   | 000010 G |
| T18CMP | 051423 | T198SS | 060632   | T3DTA  | 072200   | T39WPN | 077146   | XST2   | 000012 G |
| T18DAT | 051710 | T199CM | 061460   | T3EAI  | 072206   | T39WR  | 076512   | XST3   | 000014 G |
| T18DT2 | 051760 | T2     | 024426 G | T3EB   | 072162   | T39WRT | 077073   | XST4   | 000016 G |
| T18EXP | 051166 | T2.1   | 024444   | T3ID   | 073365   | T4     | 032014 G | XSOBOT | 000002   |
| T18LOO | 050526 | T2.2   | 024714   | T3INT  | 072756   | T4.1   | 032044   | XSOEOT | 000001   |
| T18MSK | 051162 | T20BEN | 066542   | T3MBP  | 074054   | T4.2   | 032350   | XSOIE  | 000040   |
| T18PAC | 051700 | T20BFR | 066422   | T3MSG  | 073325   | T4.3   | 032742   | XSOILA | 000400   |
| T18PK2 | 051750 | T20BFS | 066442   | T3MS1  | 073166   | T4ONE  | 101275   | XSOILC | 001000   |
| T18SET | 051570 | T20CLE | 066334   | T3MS2  | 073261   | T5     | 034604 G | XSOLET | 020000   |
| T18SMI | 051546 | T20CLR | 066122   | T3MS3  | 072325   | T5.1   | 034622   | XSOMOT | 000200   |
| T18SRD | 051526 | T20CWP | 066027   | T3NBA  | 072612   | T5.2   | 035702   | XSONEF | 002000   |
| T18SSR | 051255 | T20DAT | 066410   | T3NE   | 072250   | T6     | 040376 G | XSOONL | 000100   |
| T18XS  | 051212 | T20DT2 | 066560   | T3OFFL | 073042   | T6.1   | 040424   | XSOPED | 000010   |
| T182SS | 051312 | T20EXE | 065202   | T3ONL  | 073006   | T6.2   | 040714   | XSORLL | 010000   |
| T183SS | 051356 | T20EXP | 065062   | T3PAC  | 071470   | T6.3   | 042124   | XSORLS | 040000   |
| T19BEN | 062472 | T20EXS | 065102   | T3PK2  | 072170   | T6.4   | 043464   | XSOTMK | 100000   |
| T19BFC | 060016 | T20LOO | 062634   | T3PK3  | 072220   | T6.5   | 044042   | XSOVCK | 000020   |
| T19BFR | 062352 | T20MSK | 065056   | T3PK4  | 072240   | T6.6   | 045314   | XSOWLE | 004000   |
| T19BFS | 062372 | T20PAC | 065400   | T3RES  | 074016   | T7     | 050510 G | XSOWLK | 000004   |
| T19CLE | 062220 | T20PK2 | 066550   | T3SI2  | 072246   | T8     | 051770 G | XXCOMM | 003122 G |
| T19CLR | 061754 | T20RFI | 066206   | T3SSR  | 072666   | T8.1   | 052006   | X#ALWA | 000000   |
| T19CNV | 062264 | T20RSF | 065726   | T3SST  | 073076   | T8.2   | 054006   | X#FALS | 000040   |
| T19DAT | 062340 | T20SET | 066354   | T3TAD  | 071500   | T8.3   | 055274   | X#OFFS | 000400   |
| T19DT2 | 062510 | T20SEA | 066316   | T3WLE  | 072545   | T8.4   | 056544   | X#TRUE | 000020   |
| T19EXE | 060202 | T20SFD | 066146   | T3WR   | 072242   | T9     | 062616 G | X1.COR | 020000   |
| T19EXP | 060062 | T20SSR | 065231   | T3WRL  | 072504   | T9.1   | 062634   | X1.DLT | 100000   |
| T19EXS | 060102 | T20SWP | 065617   | T3WRT  | 072420   | UAM    | 000200 G | X1.MBZ | 017375   |
| T19LOO | 052006 | T20WFI | 066226   | T3BFR  | 075766   | UNITN  | 002202 G | X1.RBP | 000400   |
| T19MSK | 060056 | T20WFM | 066262   | T3BSSO | 075760   | UNREC  | 000006   | X1.SPA | 040000   |
| T19PAC | 062330 | T20WMI | 066302   | T3BSS1 | 075761   | USI    | 004123   | X1.UNC | 000002   |
| T19PK2 | 062500 | T20WNP | 066166   | T3BSS2 | 075762   | WAITF  | 016250 G | X2.BUF | 000100   |
| T19PRE | 060014 | T202SS | 065266   | T3DAT  | 077612   | WC.IFA | 000200   | X2.EXT | 000200   |
| T19RFI | 062066 | T203SS | 065332   | T3DLY  | 075744   | WC.IFE | 000002   | X2.OPM | 100000   |
| T19RSF | 062020 | T204SS | 065377   | T3DSW  | 076470   | WC.IGD | 000001   | X2.RCE | 040000   |
| T19RST | 061650 | T205SS | 065442   | T3DTA  | 076460   | WC.IRE | 000010   | X2.REV | 000077   |
| T19SET | 062240 | T206SS | 065506   | T3EAI  | 076466   | WC.IRW | 000004   | X2.SPA | 035400   |
| T19SEX | 062202 | T20BSS | 065551   | T3ETN  | 076671   | WC.IOT | 000100   | X2.UNI | 000007   |
| T19SNP | 062042 | T23A   | 003144 G | T3ETS  | 076602   | WC.IIT | 000040   | X2.WCF | 002000   |
| T19SRD | 062000 | T23B   | 003146 G | T3MCL  | 077533   | WC.ISR | 000020   | X3.DCK | 000010   |

D4

TSV6 - PARAMETER CODING MACRO M1113 06-FEB-84 17:14  
SYMBOL TABLE

SEQ 249

|                |                |                |                |                |
|----------------|----------------|----------------|----------------|----------------|
| X3.MBZ= 000006 | X3.REV= 000040 | X3.TRF= 000020 | X4.MBZ= 017400 | X4.TSM= 020000 |
| X3.MDE= 177400 | X3.RIB= 000001 | X4.HSP= 100000 | X4.RCE= 040000 | X4.WRC= 000377 |
| X3.OPI= 000100 | X3.SPA= 000200 |                |                |                |

. ABS. 102004 000  
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 31032 WORDS ( 122 PAGES)  
DYNAMIC MEMORY: 20614 WORDS ( 79 PAGES)  
ELAPSED TIME: 00:49:10  
CVTSBB0, CVTSBB0, SEQ/-SP=SVC/ML, TSV1B, TSV22B, TSV3B, TSV4, TSV55B, TSV6

TEST 9: READ/WRITE ....B1  
TEST 9: READ/WRITE ....C1  
TEST 9: READ/WRITE ....D1  
TEST 9: READ/WRITE ....E1  
TEST 9: READ/WRITE ....F1  
TEST 9: READ/WRITE ....G1  
TEST 9: READ/WRITE ....H1  
TEST 9: READ/WRITE ....I1  
TEST 9: READ/WRITE ....J1  
TEST 10: MANUAL INTE....K1  
TEST 10: MANUAL INTE....L1  
TEST 10: MANUAL INTE....M1  
TEST 10: MANUAL INTE....N1

TEST 10: MANUAL INTE....B2  
TEST 10: MANUAL INTE....C2  
TEST 10: MANUAL INTE....D2  
TEST 10: MANUAL INTE....E2  
TEST 10: MANUAL INTE....F2  
TEST 10: MANUAL INTE....G2  
TEST 10: MANUAL INTE....H2  
TEST 10: MANUAL INTE....I2  
TEST 10: MANUAL INTE....J2  
TEST 10: MANUAL INTE....K2  
TEST 10: MANUAL INTE....L2  
TEST 11: CONFIGURATI....M2  
TEST 11: CONFIGURATI....N2

TEST 11: CONFIGURATI....B3  
TEST 11: CONFIGURATI....C3  
TEST 11: CONFIGURATI....D3  
TEST 11: CONFIGURATI....E3  
TEST 11: CONFIGURATI....F3  
TEST 12: SCOPE LOOPS....G3  
TEST 12: SCOPE LOOPS....H3  
TEST 12: SCOPE LOOPS....I3  
TEST 12: SCOPE LOOPS....J3  
TEST 12: SCOPE LOOPS....K3  
SOFTWARE PARAMETER C....L3  
SYMBOL TABLE ....M3  
SYMBOL TABLE ....N3

SYMBOL TABLE ....B4  
SYMBOL TABLE ....C4  
SYMBOL TABLE ....D4