



TSV05

TSV05 CTRL LT3  
CVTSCB0

COPYRIGHT (c) 1982-84  
AH-T098B-MC  
FICHE 02 OF 02

APR 1985

digital

Made In USA

LE 00	LE 01	LE 02	LE 03	LE 04	LE 05	LE 06	LE 07	LE 08	LE 09	LE 10	LE 11	LE 12	LE 13	LE 14	LE 15	LE 16	LE 17	LE 18	LE 19	LE 20	LE 21	LE 22	LE 23	LE 24	LE 25	LE 26	LE 27	LE 28	LE 29	LE 30	LE 31	LE 32	LE 33	LE 34	LE 35	LE 36	LE 37	LE 38	LE 39	LE 40	LE 41	LE 42	LE 43	LE 44	LE 45	LE 46	LE 47	LE 48	LE 49	LE 50	LE 51	LE 52	LE 53	LE 54	LE 55	LE 56	LE 57	LE 58	LE 59	LE 60	LE 61	LE 62	LE 63	LE 64	LE 65	LE 66	LE 67	LE 68	LE 69	LE 70	LE 71	LE 72	LE 73	LE 74	LE 75	LE 76	LE 77	LE 78	LE 79	LE 80	LE 81	LE 82	LE 83	LE 84	LE 85	LE 86	LE 87	LE 88	LE 89	LE 90	LE 91	LE 92	LE 93	LE 94	LE 95	LE 96	LE 97	LE 98	LE 99
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

LE 00  
LE 01  
LE 02  
LE 03  
LE 04  
LE 05  
LE 06  
LE 07  
LE 08  
LE 09  
LE 10  
LE 11  
LE 12  
LE 13  
LE 14  
LE 15  
LE 16  
LE 17  
LE 18  
LE 19  
LE 20  
LE 21  
LE 22  
LE 23  
LE 24  
LE 25  
LE 26  
LE 27  
LE 28  
LE 29  
LE 30  
LE 31  
LE 32  
LE 33  
LE 34  
LE 35  
LE 36  
LE 37  
LE 38  
LE 39  
LE 40  
LE 41  
LE 42  
LE 43  
LE 44  
LE 45  
LE 46  
LE 47  
LE 48  
LE 49  
LE 50  
LE 51  
LE 52  
LE 53  
LE 54  
LE 55  
LE 56  
LE 57  
LE 58  
LE 59  
LE 60  
LE 61  
LE 62  
LE 63  
LE 64  
LE 65  
LE 66  
LE 67  
LE 68  
LE 69  
LE 70  
LE 71  
LE 72  
LE 73  
LE 74  
LE 75  
LE 76  
LE 77  
LE 78  
LE 79  
LE 80  
LE 81  
LE 82  
LE 83  
LE 84  
LE 85  
LE 86  
LE 87  
LE 88  
LE 89  
LE 90  
LE 91  
LE 92  
LE 93  
LE 94  
LE 95  
LE 96  
LE 97  
LE 98  
LE 99

.REM\_  
IDENTIFICATION

PRODUCT ID: AC-T097B-MC  
PRODUCT TITLE: CVTSCBO TSV05 CTRL LT3  
DECO/DEPO: 1.0  
DEPARTMENT: COMPUTER SPECIAL SYSTEMS/PPG  
DATE: JUNE 4, 1984

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, 1984 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL  
DEC

PDP  
DECUS

UNIBUS  
DECTAPE

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 LSI-11 RESIDENT DIAGNOSTIC WHICH CHECKS THE FUNCTIONALITY OF A TSV05 MAGTAPE SUBSYSTEM WHILE CONNECTED TO A LSI-11/23 SYSTEM (QBUS). THE PROGRAM PROVIDES ERROR MESSAGES WHICH IDENTIFY FAILING FUNCTIONS THAT AID IN THE REPAIR OF THE DEVICE. THIS DIAGNOSTIC CONSIST OF EIGHT TEST WHICH ARE EXECUTED IN SEQUENCE.

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

LSI-11 PROCESSOR AND MEMORY  
CAUTION:DIAGNOSTIC REQUIRES 32K WORDS OF MEMORY  
(28K USEABLE I.E. 4K FOR I/O PAGE)  
TSV05 MAGTAPE SUBSYSTEM (DRIVE AND CONTROLLER)  
CONSOLE TERMINAL  
PDP-11 DIAGNOSTIC SUPERVISOR (HSAAA.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 MANUAL; 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 LSI-11 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 AND CVTSBA HAVE SUCCESSFULLY RUN.

#### 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 TSV05 DIAGNOSTIC IS A LSI-11 DIAGNOSTIC SUPERVISOR COMPATIBLE PROGRAM. ALL LOADING AND RUNTIME INSTRUCTIONS CAN BE REFERENCED IN THE CHQUS XXDP+ USERS MANUAL, DOCUMENT NUMBER AC-F348E-MC. THE USER ENTRY IS IN QUOTES.

BOOT THE DIAGNOSTIC MEDIA

```
.R VTSC??
DIAG. RUN-TIME SERVICES REV D. APR 79
CVTSC-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:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

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

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

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

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

### 2.3 FLAGS

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

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

\*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1



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

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

#### 2.4 HARDWARE QUESTIONS

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

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 M7455 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 LSI-11 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 (D) ? 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 (D) ? 8<CR>

UNIT 1
CSR ADDRESS (0) ? 160000<CR>
SUB-DEVICE # (0) ? 0-7<CR>
Q-FACTOR (0) 0 ? 0,1,0,....,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 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
6. ANSWER ALL THE HARDWARE QUESTIONS
7. ANSWER THE "CHANGE SW" QUESTION WITH "N"

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

## 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  
CVTSC 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 AN UNRECOVERABLE ERROR OCCURED WHICH INDICATES THAT THE CONTROLLER MAY BE DEFECTIVE.

CVTSC 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.

CVTSC 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 (LSI-11)

```
DR>STA/FLA:PNT:HOE
```

```
UNITS (0) ? 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 TWO SWITCHES ON WHICH ARE "PRINT EACH TEST NBR AS EXECUTED" AND "HALT ON ERROR".

```
TST: 001 INITIALIZE #4 TEST  
TST: 002 OFF-LINE REJECT AND REWIND TEST  
TST: 003 BASIC WRITE DATA TEST  
TST: 004 BASIC READ DATA TEST  
TST: 005 SPACE RECORDS TEST  
TST: 006 REREADS TEST  
TST: 007 WRITE DATA RETRY TEST  
TST: 008 WRITE TAPE MARK TEST
```

```
0 ERRORS
```

NOTE: THE DIAGNOSTIC WILL RUN CONTINUOUSLY UNLESS A PASS NUMBER 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 LSI-11 PROCESSOR WITH A LA34 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 NUMBER	N/I SECS.	NUMBER ITER	DEF SECS.
1	3	10	7
2	3	8	5
3	38	250	212
4	60	300	240
5	60	300	240
6	120	360	240
7	120	600	480
8	22	90	68

THE TIMES REQUIRED TO RUN TESTS 1 THROUGH 8 IN ONE COMMAND:

Q.V. 7 MINUTES  
DEFAULT 31 MINUTES

## 5.0 DEVICE INFORMATION TABLES

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

CHANGE HW (L) ?

# UNITS (D) ? <ENTER THE NUMBER OF M7455 CONTROLLERS  
PRESENT TO BE TESTED>

UNIT 0

DEVICE ADDRESS (0) 172520 ? <ENTER THE ADDRESS OF THE  
TSBA/TSDB REGISTER>

VECTOR (0) 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.

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

CHANGE SW (L) ?

INHIBIT ITERATIONS (L) N ?

6.0 TEST SUMMARIES

TEST 1: INITIALIZE #4 TEST

THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITIONS (I.E. LOOPBACK ENABLED, FORCING WRONG PARITY, INVERTING SENSE OF EXTENDED FEATURES SWITCH, ETC.)

TEST 2: OFF-LINE AND REJECT REWIND

THIS TEST VERIFIES BASIC TAPE-MOTION COMMAND DECODING AND BASIC OPERATION OF THE REWIND POSITIONING COMMAND. IT DOES NOT NECESSARILY DEMONSTRATE THAT THE TRANSPORT CAN BE REWOUND FROM AN ARBITRARY POSITION ON THE TAPE. SUBSEQUENT TESTS IMPLICITLY CHECK THE OPERATION OF THE REWIND COMMAND SINCE THEY MUST TYPICALLY REWIND THE TAPE IN THE NORMAL COURSE OF THEIR TEST SEQUENCES.

TEST 3: BASIC WRITE DATA

THIS TEST VERIFIES THAT THE WRITE DATA (NEXT) COMMAND OPERATES PROPERLY, UP TO THE POINT OF CHECKING THAT THE DATA WAS ACTUALLY WRITTEN ONTO THE TAPE CORRECTLY. CHECKING IN THIS TEST IS LIMITED TO VERIFYING THAT THE COMMAND TERMINATED CORRECTLY WITH THE CORRECT REGISTER, MESSAGE BUFFER AND RAM CONTENTS.

\*\*\*\*\*  
CAUTION  
THE LSI BUS DRIVERS FOR ALL AVAILABLE ADDRESS LINES(16-21)  
ARE ONLY CHECKED WHEN RUNNING ON A LSI-11 SYSTEM WITH MORE THAN  
128K WORDS OF MEMORY!  
\*\*\*\*\*

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

THIS TEST VERIFIES THAT THE READ FORWARD AND READ REVERSE COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN DATA BUFFER BOUNDARIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY SPACE IS AVAILIABLE), AND BYTE-SWAP CONTROL ARE USED. THIS TEST OF COURSE, FURTHER VERIFIES THE WRITE DATA COMMAND BY ACTUALLY READING AND VERIFYING WRITTEN DATA. ALSO TESTED ARE PROPER TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES, ILLEGAL CODES IN THE MODE FIELD OF THE BASIC READ COMMAND, AND DATA BUFFERS IN NON-EXISTANT MEMORY.

\*\*\*\*\*  
CAUTION  
THE LSI BUS DRIVERS FOR ALL AVAILABLE ADDRESS LINES(16-21)  
ARE ONLY CHECKED WHEN RUNNING ON A LSI-11 SYSTEM WITH MORE THAN  
128K WORDS OF MEMORY!  
\*\*\*\*\*

## TEST 5: SPACE RECORDS

THIS TEST VERIFIES THAT THE SPACE RECORDS FORWARD AND SPACE RECORDS REVERSE POSITION COMMANDS OPERATE PROPERLY WHEN SPACING OVER NORMAL DATA RECORDS. OPERATION WHEN SPACING OVER TAPE MARKS IS VERIFIED IN A SUBSEQUENT TEST. THE BASIC WRITE DATA TEST SHOULD HAVE BEEN RUN SUCCESSFULLY FOR THIS TEST TO PRODUCE MEANINGFUL RESULTS. THIS TEST CONSISTS OF A SERIES OF SUBTESTS. IN EACH OF THE SUBTESTS, THE TAPE IS ENTIRELY WRITTEN WITH RECORDS OF VARYING SIZES AND DATA CONTENT; THE FIRST 4 BYTES OF EACH RECORD INDICATE THAT RECORD'S RELATIVE POSITION ON TAPE. AFTER EACH SPACING OPERATION, THE TAPE POSITION IS VERIFIED BY READING THE NEXT OR PREVIOUS RECORD AND COMPARING THE POSITION DATA WITH THE EXPECTED RESULT.

## TEST 6: REREADS

THIS TEST VERIFIES THAT THE REREAD PREVIOUS AND REREAD NEXT COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN DATA BUFFER BOUNDRIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY SPACE IS AVAILIABLE), AND BYTE-SWAP (SWP) AND OPPOSITE (OPP) CONTROL ARE USED. ALSO TESTED ARE PROPER TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES, AND DATA BUFFERS IN NONEXISTENT MEMORY.

\*\*\*\*\*

## CAUTION

THE LSI BUS DRIVERS FOR ALL AVAILABLE ADDRESS LINES(16-21) ARE ONLY CHECKED WHEN RUNNING ON A LSI-11 SYSTEM WITH MORE THAN 128K WORDS OF MEMORY!

\*\*\*\*\*

## TEST 7: WRITE DATA RETRY

THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY COMMAND (SPACE REVERSE, ERASE, WRITE DATA)

## TEST 8: WRITE/READ TAPE MARK

THIS TEST VERIFIES THAT THE WRITE TAPE MARK COMMAND OPERATES PROPERLY. IT IS VERIFIED THAT THE TAPE MARK IS WRITTEN ONTO TAPE BY CHECKING THAT THE READ AND SPACE RECORDS COMMANDS DETECT THE TAPE MARK. IN ADDITION, SINCE WRITE TAPE MARK IS THE FIRST SUBCOMMAND UNDER THE FORMAT COMMAND BEING TESTED, IT IS VERIFIED THAT THE CLEAR VOLUME CHECK (CVC) BIT OPERATES PROPERLY AND THAT FORMAT COMMANDS WITH ILLEGAL MODE CODES ARE REJECTED.

7.0 MAINTENANCE HISTORY

REVISION A - JUNE 1983

REVISION B - JUNE 1984

UPDATED FOR NEW ORION CPU PROBLEMS, RELATING TO TIMEOUT  
ERRORS (#311 & #320) ON REWINDS.  
ELIMINATED CPU ID MESSAGE.

```

1          .TITLE  TSV2 - PROGRAM HEADER
2          .SBTTL  PROGRAM HEADER
3 000000   .PSECT  ABS
4
10         .MCALL  SVC
11 000000   SVC          ; INITIALIZE SUPERVISOR MACROS
12         .ENABLE LC
13         .NLIST  BEX,CND
19 000000   .ENABL  ABS,AMA
20         .=2000
21 002000   BGNMOD  TSV2
22         002000

```

TSV2::

```

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

```

```

29 002000   POINTER BGNSW,BGNSFT,BGNAU,BGNDU,BGNRPT
30 002000   HEADER  CVTSC,B,0,655.,0
          L$NAME:: ;DIAGNOSTIC NAME
          .ASCII /C/
          .ASCII /V/
          .ASCII /T/
          .ASCII /S/
          .ASCII /C/
          .BYTE  0
          .BYTE  0
          .BYTE  0
L$REV:: ;REVISION LEVEL
          .ASCII /B/
L$DEPO:: ;0
          .ASCII /0/
L$UNIT:: ;NUMBER OF UNITS
          .WORD  0
L$TIML:: ;LONGEST TEST TIME
          .WORD  655.
L$HPCP:: ;PTR. TO H.W. QUES.
          .WORD  L$HARD
L$SPCP:: ;PTR. 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$APT:: ;APT EXPANSION
          .WORD  0
L$DTP:: ;PTR. TO DISPATCH TABLE

```

```

1
2
3 000000
4
10
11 000000
12
13
19 000000
20         002000
21 002000
22         002000
23
24
25
26
27
28
29 002000
30 002000
   002000
   002000   103
   002001   126
   002002   124
   002003   123
   002004   103
   002005   000
   002006   000
   002007   000
   002010
   002010   102
   002011
   002011   060
   002012
   002012   000000
   002014
   002014   001217
   002016
   002016   112730
   002020
   002020   113062
   002022
   002022   002146
   002024
   002024   002156
   002026
   002026   113404
   002030
   002030   000000
   002032
   002032   000000
   002034
   002034   000000
   002036
   002036   000000
   002040

```

002040	002124		.WORD	L\$DISPATCH	
002042		L\$PRIO::			;DIAGNOSTIC RUN PRIORITY
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	003372		.WORD	L\$DVTYP	
002062		L\$REPP::			;PTR. TO REPORT CODE
002062	022710		.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	022376		.WORD	L\$AU	
002072		L\$DUT::			;PTR. TO DROP UNIT CODE
002072	022474		.WORD	L\$DU	
002074		L\$LUN::			;LUN FOR EXERCISERS TO FILL
002074	000000		.WORD	0	
002076		L\$DESP::			;POINTER TO DIAG. DESCRIPTION
002076	003400		.WORD	L\$DESC	
002100		L\$LOAD::			;GENERATE SPECIAL AUTOLOAD EMT
002100	104035		EMT	E\$LOAD	
002102		L\$ETP::			;POINTER TO ERR_TBL
002102	000000		.WORD	0	
002104		L\$ICP::			;PTR. TO INIT CODE
002104	021602		.WORD	L\$INIT	
002106		L\$CCP::			;PTR. TO CLEAN-UP CODE
002106	022662		.WORD	L\$CLEAN	
002110		L\$ACP::			;PTR. TO AUTO CODE
002110	022602		.WORD	L\$AUTO	
002112		L\$PRT::			;PTR. TO PROTECT TABLE
002112	021572		.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	

```

33          .SBTTL DISPATCH TABLE
34          ;**
35          ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
36          ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
37          ;--
38 002122          DISPATCH 8
002122 000010          .WORD 8
002124          L$DISPATCH::
002124 023472          .WORD T1
002126 024612          .WORD T2
002130 027272          .WORD T3
002132 034452          .WORD T4
002134 046716          .WORD T5
002136 055644          .WORD T6
002140 075226          .WORD T7
002142 105264          .WORD T8

39          .SBTTL DEFAULT HARDWARE P-TABLE
40          ;**
41          ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
42          ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
43          ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
44          ;--
46 002144          BGNHW DFPTBL ;DEFAULT HARD-P-TABLE
002144 000003          .WORD L10000-L$HW/2
002146          L$HW::
002146          DFPTBL::
47 002146 172520          .WORD 172520 ; 1ST (OF 2) REGISTERS.
48 002150 000224          .WORD 224 ; INTERRUPT VECTOR
49 002152 000200          .WORD PRI04 ; INTERRUPT PRIORITY.
50 002154          ENDHW
002154          L10000:

51          .SBTTL SOFTWARE P-TABLE
52          ;**
53          ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
54          ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
55          ;--
57 002154          BGNLW SFPTBL
002154 000004          .WORD L10001-L$SW/2
002156          L$SW::
002156          SFPTBL::
58 002156 000000          TRANSTST:: .WORD 0 ; ENABLE TEST OF TRANSPORT(S) IF =1
59 002160 000000          NOITS:: .WORD 0 ; INHIBIT ITERATION OPTION.
60          ; ... 0 = ITERATE.
61          ; ...NZ = INHIBIT ITERATE.
62 002162 000017          LERRMAX:: .WORD 15. ; LOCAL (PER TEST) ERROR LIMIT
63 002164 000310          GERRMAX:: .WORD 200. ; GLOBAL (PER UNIT) ERROR LIMIT
64 002166          ENDSW
002166          L10001:
65 002166          ENDMOD
66

```



7  
8  
13  
19  
20 002166  
002166  
21  
22  
23  
24  
25  
26  
27  
28  
32 002166

.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 DEFINITIONS

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

```
000340      PRI07== 340
000300      PRI06== 300
000240      PRI05== 240
000200      PRI04== 200
000140      PRI03== 140
000100      PRI02== 100
000040      PRI01== 40
000000      PRI00== 0
```

```
      ;
      ;OPERATOR FLAG BITS
```

```
000004      EVL==      4
000010      LOT==     10
000020      ADR==     20
000040      IDU==     40
000100      ISR==    100
000200      UAM==    200
000400      BOE==    400
001000      PNT==   1000
002000      PRI==   2000
004000      IXE==   4000
010000      IBE==  10000
020000      IER==  20000
040000      LOE==  40000
100000      HOE== 100000
```

33  
34 002166

```
      ;DEFINE MEMORY MANAGEMENT REGISTERS
      .SBTTL  KT11      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
```

```
;*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
. 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
```

```

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
KDPDR0= 172320
KDPDR1= 172322
KDPDR2= 172324
KDPDR3= 172326
KDPDR4= 172330
KDPDR5= 172332
KDPDR6= 172334
KDPDR7= 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
KDPAR0= 172360
KDPAR1= 172362
KDPAR2= 172364
KDPAR3= 172366
KDPAR4= 172370
KDPAR5= 172372
KDPAR6= 172374
KDPAR7= 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=        BIT:3       ;SANITY CHECK ERROR
58          010000  RMR=        BIT12       ;MODIFICATION REFUSED
59          004000  NXM=        BIT11       ;NONEXISTANT MEMORY ERROR
60          002000  NBA=        BIT10       ;NEED BUFFER ADDRESS
61          001400  HIADDR=    BIT9!BIT8    ;EXTENDED ADDRESS BITS
62          000200  SSR=        BIT7        ;SUB SYSTEM READY
63          000100  OFL=        BIT6        ;OFF LINE BIT
64          000060  FATERR=    BIT4!BITS    ;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  XSCMLE=    BIT11       ;WRITE LOCK ERROR
79          002000  XSONEF=    BIT10       ;NON EXECUTABLE FUNCTION
80          001000  XSOILC=    BIT9        ;ILLEGAL COMMAND
81          000400  XSOILA=    BIT8        ;ILLEGAL ADDRESS
82          000200  XSOMOT=    BIT7        ;TAPE IN MOTION
83          000100  XSOONL=    BIT6        ;TRANSPORT ON LINE
84          000040  XSOIE=     BITS        ;INTERRUPT ENABLE
85          000020  XSOVCK=    BIT4        ;VOLUME CHECK BIT
86          000010  XSOPED=    BIT3        ;PHASE ENCODED DRIVE
87          000004  XSOMLK=    BIT2        ;WRITE LOCKED
88          000002  XS0BOT=    BIT1        ;BEGINNING OF TAPE
89          000001  XS0EOT=    BIT0        ;END OF TAPE

```

```

91      ;*
92      ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
93      ;(XST1)
94      ;-
95      100000 X1.DLT = BIT15      ;DATA LATE
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

```

```

147      ;*
148      ;
149      ;DEVICE REGISTER OFFSETS
150      ;
151      ;-
152
153      000000      TSBA== 0
154      000000      TSDB== 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      ; TSDB 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:BITS      ;PACKET HEADER TYPE (ALWAYS=0)
188      000037      P.CMD          = 37       ;MAJOR COMMAND FIELD
189
190      ;*
191      ; CONTROL COMMAND MODE CODES
192      ;-
192      000000      PC.RELEASE    = 0*256.    ;RELEASE BUFFER
193      000400      PC.REWIND     = 1*256.    ;REWIND
194      001000      PC.NOOP       = 2*256.    ;NO-OP
195      002000      PC.IEREW      = 4*256.    ;REWIND IMMEDIATE INTERRUPT
196      002400      PC.ERASE      = 5*256.    ;SECURITY ERASE

```

```

198      ;*
199      ; CONTROLLER RAM DEFINITIONS
200      ;-
201      000167      RMCHBEG = 167      ;CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
202      000200      RMCHEND = 200      ;CHARACTERISTICS IO DATA END RAM ADDRESS
203      000201      RMPKTBEG= 201      ;COMMAND PACKET BEGIN RAM ADDRESS
204      000210      RMPKTEND= 210      ;COMMAND PACKET END RAM ADDRESS
205      000215      RMMSGBEG= 215      ;MESSAGE BUFFER BEGIN RAM ADDRESS
206      000234      RMMSGEND= 234      ;MESSAGE BUFFER END RAM ADDRESS
207      ;*
208      ;
209      ;REGISTER DEFINITIONS IN THE MESSAGE BUFFER
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      PKHI   = 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      BSELO  = 0      ;BYTE 0
235      000001      BSEL1  = 1      ;BYTE 1
236      000002      SEL2   = 2      ;WORD 2
237      000004      SELDATA = 4      ;WORD 3

```



```

239      ;*
240      ;BSELO 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.RDEXT       = 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       ;ITADO - TRANSPORT ADDRESS BIT 0
263      000040      WC.I1TAD        = BIT5       ;ITAD1 - 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
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

```

```

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      ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
300      ;-
301      000200      NP.IR      = BIT7      ;INTERRUPT REQUEST (0-1 TRANSITION)
302      000100      NP.OUT     = BIT6      ;TAPE DATA DIRECTION OUT (0= IN)
303      000040      NP.LOOP    = BIT5      ;ENABLE TRANSPORT LOOPBACK
304      000020      NP.WRP     = BIT4      ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
305      ;+
306      ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
307      ;-
308
309      000200      S2.DIM      = BIT7      ;WORD #9 BYTE 2 DATA IN MISS
310      000100      S2.ILW     = BIT6      ;           ILW H
311      000040      S2.OUTRDY   = BIT5      ;           OUT RDY H
312      000020      S2.INRDY   = BIT4      ;           IN RDY H
313      000010      S2.ATIMR   = BIT3      ;           TIMER A FLAG H
314      000004      S2.BTIMR   = BIT2      ;           TIMER B FLAG H
315      000003      S2.UNDEF    = BIT1+BIT0 ;(UNDEFINED)
316      100000      S1.PARIN    = BIT15     ;WORD #8 BYTE 1 PARIN H
317      040000      S1.I2RESV   = BIT14     ;           IRESV2
318      020000      S1.I1RESV   = BIT13     ;           IRESV1
319      010000      S1.IEOT     = BIT12     ;           IEOT L
320      004000      S1.IIDENT   = BIT11     ;           IIDENT H
321      002000      S1.ICER     = BIT10     ;           ICER H
322      001000      S1.IFMK     = BIT9      ;           IFMK H
323      000400      S1.IHER     = BIT8      ;           IHER H
324      000200      S0.ISPEED   = BIT7      ;WORD #8 BYTE 0 ISPEED H
325      000100      S0.IRDY     = BIT6      ;           IRDY L
326      000040      S0.IONL     = BIT5      ;           IONL L
327      000020      S0.ILDPL    = BIT4      ;           ILDP L
328      000010      S0.IDBY     = BIT3      ;           IDBY L
329      000004      S0.IRWD     = BIT2      ;           IRWD L
330      000002      S0.IFBY     = BIT1      ;           IFBY L
331      000001      S0.IFPT     = BIT0      ;           IFPT L

```

```

333             .SBTTL SPECIAL MACROS AND OPDEFS.
334
335             ;+
336             ;SAVE GENERAL REGS 1 TO 5
337             ;-
338
339             .MACRO SAVREG
340             JSR   R5,REGSAV
341             .ENDM
342
343             ;+
344             ; MACRO TO FORCE AN ERROR
345             ;-
346             .MACRO FORCERROR TAG,NOTSSR
347             .NLIST
348             .IIF NDF LISTALL, .NLIST
349             .LIST
350             .IF B NOTSSR
351                 MOV   TSSR(R5),R1      ;READ TSSR
352             .ENDC
353                 MOV   FORCER,FORCER    ;IS FORCER SET? (LEAVE C BIT ALONE)
354                 BNE   TAG              ;BR IF YES
355             .NLIST
356             .IIF NDF LISTALL, .LIST
357             .LIST
358             .ENDM
359
360             ;+
361             ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
362             ; WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
363             ; SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
364             ; FORCER TO 177777
365             ; TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
366             ;-
367             .MACRO FORCEEXIT TAG
368             .NLIST
369             .IIF NDF LISTALL, .NLIST
370             .LIST
371                 MOV   FORCER,FORCER    ;IS FORCER NEGATIVE?
372                 BMI   TAG              ;BR IF YES
373             .NLIST
374             .IIF NDF LISTALL, .LIST
375             .LIST
376             .ENDM
377             ;+
378             ; MACRO TO INCREMENT ERROR COUNTS
379             ;-
380             .MACRO NEXT.ERRNO
381             .NLIST
382             ;;;.IIF NDF LISTALL, .NLIST
383                 ERRNO=ERRNO+1
384             ;;;.IIF NDF LISTALL, .LIST
385             .LIST
386             .ENDM

```

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

SEQ 0034

```

388      ;*
389      ;MACRO TO PERFORM XOR
390      ;-
391
392      .MACRO XOR      A,B
393      MOV      A,-(SP)
394      BIC      B,(SP)
395      BIC      A,B
396      BIS      (SP),B
397      .ENDM
398
399      000000      EN=0                    ; INITIALIZE ERROR NUMBER
400                    .SBTTL FORCER - FORCE ERROR FLAG
401
402      ;
403      ; THE FOLLOWING LOCATIONS MAY BE PATCHED BY THE USER
404      ; TO OBTAIN THE RESULTS DESCRIBED FOR EACH.
405      ;
406
407 002166 000000    FORCER::            0            ; FORCE TYPE ALL HARD ERRORS (THE ONES CALLED -
408                    ; - BY THE MACRO "IFERROR"). AN ERROR NEED NOT -
409                    ; - EXIST, JUST ASSUME AND TYPE THE MESSAGE.

```

.SBTTL GLOBAL DATA SECTION

```

411
412
413      ;**
414      ;THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
415      ;IN MORE THAN ONE TEST.
416      ;--
417
418      ;
419      ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
420      ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
421      ;
422 002170 000000 EPRTSW::      .WORD 0      ;PRINT SWITCH
423 002172 000000 UNITN::      .WORD 0      ;UNIT # UNDER TEST.
424 002174 000000 QVP::        .WORD 0      ;QUICK VERIFY FLAG.
425 002176 000000 CSRADDR::    .WORD 0      ;ADDRESS OF CSR FOR CURRENT DEVICE
426 002200 000224 IVEC::        .WORD 224    ;INTERRUPT VECTOR
427 002202 000200 IPRI::        .WORD PRI04  ;INTERRUPT PRIORITY.
428 002204 000000 TSTCNT::    .WORD 0      ;NUMBER OF TESTS RUN IN THIS PASS
429 002206 000000 LOOPCNT::   .WORD 0      ;REMAINING ITERATION COUNT FOR TEST
430 002210 000000 DEVCNT::   .WORD 0      ;NUMBER OF DEVICE UNDER TEST
431 002212 000000 FATFLG::   .WORD 0      ;SET IF FATAL ERROR IS DETECTED IN TEST
432 002214 000000 INTRECV::   .WORD 0      ;SET IF TAPE INTERRUPT WAS RECEIVED
433 002216 000000 EXTFEA::   .WORD 0      ;EXTENDED FEATURES SOFTWARE SW 0=OFF;1=ON
434 002220 000000 BENBSW::   .WORD 0      ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
435 002222 000000 EXPD::     .WORD 0      ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
436 002224 000000 RECV::     .WORD 0      ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
437 002226 000000 ERRHI::    .WORD 0      ;HIGH ADDRESS MEMORY ERROR
438 002230 000000 ERRLO::    .WORD 0      ;LOW ADDRESS MEMORY ERROR
439 002232 000000 RAMDATA::  .BLKW 16.   ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
440 002272 000000 RAMSIZ::  .WORD 0      ;RAM DATA SIZE FOR PRAMPKT ROUTINE
441 002274 000000 RCVHIADD::.WORD 0      ;RECEIVED BUFFER HIGH ADDRESS
442 002276 000000 RCVLOADD::.WORD 0      ;RECEIVED BUFFER LOW ADDRESS
443 002300 000000 COUNT::   .WORD 0      ;TEST COUNT PATTERN
444 002302 000000 DATA::   .WORD 0      ;TEST DATA
445 002304 000000 TSTFLAG:: .WORD 0      ;TEST FLAG WORD
446 002306 000000 TSTPTR::  .WORD 0      ;TSTBLK POINTER
447 002310 000000 PRMNO::   .WORD 0      ;PRINT ROUTINE TEMP
448 002312 000000 EXPMSG::  .BLKB 100.  ;EXPECTED MESSAGE BUFFER DATA
449 002456 000000 RECMMSG:: .BLKB 100.  ;RECEIVED MESSAGE BUFFER DATA
450 002622 000000 TMPBFR::  .BLKB 80.   ;TEMPORARY STORAGE FOR PRINT

```

```

452                            .SBTTL TSTBLK - TEST DATA TABLE
453
454                            ;*
455                            ; THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
456                            ;
457                            ; IN SEQUENCE THE DATA IS:
458                            ;
459                            ;
460                            ;        ALL ZEROS
461                            ;        ALL ONES
462                            ;        WALKING ONES
463                            ;        WALKING ZEROS
464                            ;        ALTERNATING ONES AND ZEROS
465                            ;
466                            ; -
467
468                            TSTBLK::
469                            .WORD    0                            ; ALL ZEROS
470                            .WORD    177777                        ; ALL ONES
471                            .WORD    BIT0                        ; DATA FOR WALKING ONES
472                            .WORD    BIT1
473                            .WORD    BIT2
474                            .WORD    BIT3
475                            .WORD    BIT4
476                            .WORD    BIT5
477                            .WORD    BIT6
478                            .WORD    BIT7
479                            .WORD    BIT8
480                            .WORD    BIT9
481                            .WORD    BIT10
482                            .WORD    BIT11
483                            .WORD    BIT12
484                            .WORD    BIT13
485                            .WORD    BIT14
486                            .WORD    BIT15
487                            .WORD    †CBIT0                        ; DATA FOR WALKING ZEROS
488                            .WORD    †CBIT1
489                            .WORD    †CBIT2
490                            .WORD    †CBIT3
491                            .WORD    †CBIT4
492                            .WORD    †CBIT5
493                            .WORD    †CBIT6
494                            .WORD    †CBIT7
495                            .WORD    †CBIT8
496                            .WORD    †CBIT9
497                            .WORD    †CBIT10
498                            .WORD    †CBIT11
499                            .WORD    †CBIT12
500                            .WORD    †CBIT13
501                            .WORD    †CBIT14
502                            .WORD    †CBIT15
503                            .WORD    125252                        ; ALTERNATING ONES, ZEROS
504                            .WORD    052525                        ; ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE
505                            .WORD    003052
                              TBLEND==.

```

```

507          .SBTTL GLOBAL ENVIRONMENT STORAGE
508          ;
509          ; STORAGE FOR DEVICE REGISTERS
510          ;
511 003052 000000 100000 000000 DUMMY: 0,100000,0,0 ; DUMMY DEVICE REGISTERS...
512 003062 000000 000000 000000      0,0,0,0,0,0,0,0,0 ; ...FOR MULTI-UNIT CHECKOUT.
513
514
515 003102 000000 DUFLG::          .WORD 0 ; "DROPPED UNIT" FLAG.
516                                     ; INHIBITS CODE IN "CLEAN-UP".
517 003104 000000 NODEV::          .WORD 0 ; FLAG TO SAY NO DEVICE.
518
519 003106 000000 TEMP1::          .WORD 0 ; SOME TEMP LOCATIONS.
520 003110 000000 TEMP2::          .WORD 0
521 003112 000000 XXCOMM::         .WORD 0 ; XXDP, COMM BLOCK POINTER.
522 003114 000000 FREE::          .WORD 0 ; 1ST FREE MEMORY ADDRESS...
523 003116 000000 FRESIZ::         .WORD 0 ; ...AND SIZE (IN WORDS).
524 003120 000000 FREEHI: .WORD 0 ; LAST WORD IN FREE SPACE
525 003122 000000 KTFLG::          .WORD 0 ; KT11, MEM AVAIL FLAG -
526                                     ; - .WORD 0 = <24K OR NO KT -
527                                     ; - NZ = >24K AND KT.
528 003124 000000 KTENABLE::        .WORD 0 ; SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
529 003126 000000 NXMFLG::         .WORD 0 ; SET IF WE CAN TEST CLEARED OTHERWISE
530 003130 000000 NXMLO::          .WORD 0 ; NXM LO ADDRESS BITS
531 003132 000000 NXMHI::          .WORD 0 ; NXM HI ADDRESS BITS FOR DAL'S 16-21
532 003134 000000 T23A::          .WORD 0 ; 11/23A FLAG
533 003136 000000 T23B::          .WORD 0 ; 11/23B FLAG
534 003140 000000 T3BFLG::         .WORD 0 ; TEST 3B FLAG +0
535 003142 002000 PST32W::         .WORD 2000 ; 32W BLOCK ADDRESS FOR 32K START
536 003144 000000 SIFLAG::         .WORD 0 ;
537 003146 000000 BADDAT::         .WORD 0 ; ACTUAL DATA
538 003150 000000 GDDAT::         .WORD 0 ; EXPECTED DATA
539 003152 000000 LOOPFL::        .WORD 0 ;
540 003154          CTAB::          ; CONFIGURATION TABLES.
541 003154 000000 CTABM::          .WORD 0 ; CONFIG WORK.
542 003156 000000          .WORD 0
543 003160 000000          .WORD 0
544 003162 000000          .WORD 0
545 003164 177777          .WORD -1 ; END OF MEM TABLE.
546 003166          CTABE::
547          ; ERROR STATISTICS TABLE (1 WORD PER UNIT), 64 UNITS MAX:
548          ;
549          ; 0 = UNIT NOT TESTED
550          ; 100000 = UNIT ONLINE, NO ERRORS
551          ; 10XXXX = UNIT ONLINE, ENCOUNTERED XXXX ERRORS
552          ; 160000 = UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
553          ; 160001 = UNIT DROPPED, NOT IDLE AT START
554          ; 14XXXX = UNIT DROPPED, ENCOUNTERED XXXX ERRORS
555          ;
556 003166          ERTABL:          .BLKW 64.
557 003366 000000          ERTABE:          .WORD 0
558
559 003370 000000          SKIPT: .WORD 0 ; 1=SKIP SUBTEST 0=NO SKIP OF SUBTEST

```

```

561          .SBTTL GLOBAL TEXT MESSAGES
562          ;**
563          ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
564          ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
565          ; MORE THAN ONE TEST.
566          ;--
567          ;*
568          ;NAMES OF DEVICES SUPPORTED
569          ;
570 003372          DEVTYP <TSV05>
571          L$DVTYP::
572          003372          .ASCIZ /TSV05/
573          003372          .EVEN
574
575          ;*
576          ;TEST DESCRIPTION
577          ;-
578          ;
579 003400          DESCRIPT <**** TSV05 LOGIC DIAGNOSTIC - CHK CABLES-TRANSPORT IF ERR ****>
580          003400          L$DESC::
581          003400          .ASCIZ /**** TSV05 LOGIC DIAGNOSTIC - CHK CABLES-TRANSPORT IF ERR ****/
582          .EVEN
583
584          ;*
585          ;BIT TO ASCII CONVERSION FOR TSSR REGISTER
586          ;-
587          ;
588 601 003500 003540 003543 003547 TSSRBIT::          .WORD 1$,2$,3$,4$,5$,6$,7$,8$
589 602 003520 003601 003605 003611          .WORD 9$,10$,11$,12$,13$,14$,15$,16$
590 603 003540          123          103          000 1$:          .ASCIZ 'SC'
591 604 003543          102          111          105 2$:          .ASCIZ 'BIE'
592 605 003547          123          103          105 3$:          .ASCIZ 'SCE'
593 606 003553          122          115          122 4$:          .ASCIZ 'RMR'
594 607 003557          116          130          115 5$:          .ASCIZ 'NXM'
595 608 003563          116          102          101 6$:          .ASCIZ 'NBA'
596 609 003567          102          111          124 7$:          .ASCIZ 'BIT9'
597 610 003574          102          111          124 8$:          .ASCIZ 'BIT8'
598 611 003601          123          123          122 9$:          .ASCIZ 'SSR'
599 612 003605          117          106          114 10$:          .ASCIZ 'OFL'
600 613 003611          102          111          124 11$:          .ASCIZ 'BIT5'
601 614 003616          102          111          124 12$:          .ASCIZ 'BIT4'
602 615 003623          102          111          124 13$:          .ASCIZ 'BIT3'
603 616 003630          102          111          124 14$:          .ASCIZ 'BIT2'
604 617 003635          102          111          124 15$:          .ASCIZ 'BIT1'
605 618 003642          102          111          124 16$:          .ASCIZ 'BIT0'
606 619          .EVEN
607 620 003650          124          123          123 SFIERR: .ASCIZ 'TSSR ERROR AFTER SOFT INIT'
608 621 003703          124          123          123 SFHERR: .ASCIZ 'TSSR ERROR AFTER BUS RESET'
609 622 003736          040          040          116 NXR: .ASCIZ / NON-EXISTANT DEVICE REGISTER/
610 623 003775          045          101          040 NXR: .ASCIZ /#A ADDRESS: #06/
611 624 004016          045          101          040 TSSX: .ASCII /#A TSBA,TSSR EXP'D: #06#A,#06#N/
612 625 004056          045          101          040 TSSX: .ASCII /#A TSBA,TSSR REC'D: #06#A,#06#N/
613 626 004115          045          116          045 FUSI: .ASCII /#N#A/
614 627 004121          040          040          125 USI: .ASCIZ / UNEXPECTED INTERRUPT/
615 628 004150          040          040          111 NSI: .ASCIZ / INTERRUPT EXPECTED, NOT RECEIVED/
616 629 004213          045          116          045 FNOINTR: .ASCII /#N#A/
617 630 004217          040          040          116 NOINTR: .ASCIZ / NO INTERRUPT WAS GENERATED/
618 631 004254          040          040          111 IFAULT: .ASCIZ / INTERRUPT FAULT/
619 632 004276          045          101          040 INTX: .ASCIZ /#A CPU PC: #06#A TSBA: #06/

```



```

633 004333      040      040      042 NOINIT: .ASCIZ / "BUS-INIT" DIDN'T INITIALIZE CONTROLLER/
634 004405      040      040      042 NSINIT: .ASCIZ / "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
635 004455      040      040      042 BRINIT: .ASCIZ / "BUS-RESET" DIDN'T INITIALIZE THE DPU/
636 004525      000      000      NUL: .ASCIZ //
637 004525      045      116      000 NULCR: .ASCIZ /#N/
638 004531      045      101      040 EXPGOT: .ASCIZ /#A EXP'D: #06#A, REC'D: #06/
639 004565      045      116      045 EXPGT2: .ASCIZ /#N#A EXP'D: #06#A, #06#N#A REC'D: #0#A, #06/
640 004641      045      101      040 DUAD12: .ASCIZ /#A REG(W) WRITTEN TO: #06#A REG(R) READ; EXP'D: #06#A, REC'D: #06/
641 004743      122      101      115 PKTRAM: .ASCIZ 'RAM Contents Do Not Match Packet Sent'
642 005011      040      040      103 SCME: .ASCIZ / CONFIG DOESN'T MATCH MFG. MASTER/
643 005054      127      122      111 WRTMSG: .ASCIZ 'WRITE CHARACTERISTICS Failed'
644 005111      124      123      123 WRTERR: .ASCIZ 'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
645 005204      124      123      123 RDERR: .ASCIZ 'TSSR Incorrect After READ Command, More Bits Set Than SSR'
646 005276      106      101      124 SCHERR: .ASCIZ 'FATAL ERROR IN SUBTEST - CHECK TAPE,CABLES,TRANSPORT etc.'
647 005370      105      122      122 RETERR: .ASCIZ 'ERROR IN SUBTEST - WRITE DATA RETRY FIVE TIMES FAILED'
648 005456      045      116      045 NOMEM: .ASCIZ '#N#A ***** NO NXM ADDRESS--CANNOT TEST NXM TIMEOUT. *****N'
649 005552      045      116      045 M8186: .ASCIZ '#N#A ***** 11/23A SYSTEM *****N'
650 005643      045      116      045 M8189: .ASCIZ '#N#A ***** 11/23B SYSTEM *****N'

```

```

651 .EVEN
652 .SBTTL GLOBAL ERROR REPORT SECTION
653
654

```

```

655 ;**
656 ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX
657 ; CALLS THAT ARE USED IN MORE THAN ONE TEST.
658 ; ASCII TEXT STRINGS ARE FOUND IN THE GLOBAL TEXT SECTION.
659 ;--

```

```

659 005734      BGNMSG  NXRERR          ;NON-EXISTANT DEVICE REGISTER.
005734
660 005734      NXRERR: :
005734      PRINTX  #NXRX,NODEV ;NODEV = NEXM ADDRESS.
013746 003104      MOV      NODEV,-(SP)
005740 012746 003775      MOV      #NXRX,-(SP)
005744 012746 000002      MOV      #2,-(SP)
005750 010600      MOV      SP,R0
005752 104415      TRAP    C#PNTX
005754 062706 000006      ADD      #6,SP
661 005760 004737 005766      JSR     PC,EXTEND ; PRINT EXTENSION IF REQUIRED.
662 005764      ENDMSG
005764

```

```

662 005764      L10002: TRAP    C#MSG
005764 104423
663

```

```

664 ; THIS ROUTINE APPENDS A UNIQUE EXTENSION (IF REQUIRED)
665 ; TO ANY OF THE ABOVE ERROR SIGNATURES.
666 ;

```

```

667 005766 005727      EXTEND: TST    (PC)+
668 005770 000000      EXTA:  0          ; 0 = NO EXTENSION.
669 005772 001402      BEQ     1$
670 005774 004777 177770      JSR     PC,@EXTA ; APPEND EXTENSION TEXT.
671 006000      1$: PRINTX  #NULCR ; PRINT A BLANK LINE
006000 012746 004526      MOV      #NULCR,-(SP)
006004 012746 000001      MOV      #1,-(SP)
006010 010600      MOV      SP,R0
006012 104415      TRAP    C#PNTX
006014 062706 000004      ADD      #4,SP
672 006020 000207      RTS     PC

```



```

006166 012746 000002      MOV      #2,-(SP)
006172 010600      MOV      SP,R0
006174 104415      TRAP    C:PNTX
006176 062706 000006      ADD      #6,SP

718
719 006202 010403      20$:   MOV      R4,R3          ;GET THE TSSR CONTENTS
720 006204 042703 177761      BIC      #+CTERCLS,R3   ;CLEAR ALL BUT TERMINATION
721 006210 016303 006766      MOV      TCOCOD(R3),R3  ;GET THE TERMINATION CODE MEANING
722 006214      PRINTX #TCOASC,R3      ;PRINT THE TERMINATION CODE
      006214 010346      MOV      R3,-(SP)
      006216 012746 006566      MOV      #TCOASC,-(SP)
      006222 012746 000002      MOV      #2,-(SP)
      006226 010600      MOV      SP,R0
      006230 104415      TRAP    C:PNTX
      006232 062706 000006      ADD      #6,SP

723 006236 010403      MOV      R4,R3          ;TSSR CONTENTS AGAIN
724 006240 042703 177717      BIC      #+CFATERR,R3  ;CLEAR ALL BUT FATAL TERMINATION
725 006244 001416      BEQ     25$            ;DON'T PRINT IF ZERO
726 006246 006203      ASR     R3
727 006250 006203      ASR     R3
728 006252 006203      ASR     R3          ;ALINE TERMINATION CODE FOR INDEX
729 006254 016303 007326      MOV      TSFCOD(R3),R3 ;GET THE FATAL TERMINATION CODE
730 006260      PRINTX #TFCASC,R3      ;PRINT THE FATAL TERMINATION CODE
      006260 010346      MOV      R3,-(SP)
      006262 012746 006627      MOV      #TFCASC,-(SP)
      006266 012746 000002      MOV      #2,-(SP)
      006272 010600      MOV      SP,R0
      006274 104415      TRAP    C:PNTX
      006276 062706 000006      ADD      #6,SP

731 006302 042704 176377      25$:   BIC      #+CHIADDR,R4  ;CLEAR ALL BUT EXTENDED ADDRESS
732 006306 001411      BEQ     30$            ;DON'T PRINT IF ZERO
733 006310      PRINTX #TEXASC,R4      ;PRINT THE EXTENDED ADDRESS BITS
      006310 010446      MOV      R4,-(SP)
      006312 012746 006525      MOV      #TEXASC,-(SP)
      006316 012746 000002      MOV      #2,-(SP)
      006322 010600      MOV      SP,R0
      006324 104415      TRAP    C:PNTX
      006326 062706 000006      ADD      #6,SP

734 006332 013703 002170      30$:   MOV      EPRTSW,R3      ;PRINT MEASGE BUFFER ADDRESS
735 006336      PRINTX R3            ;PRINT PROPER MESSAGE
      006336 010346      MOV      R3,-(SP)
      006340 012746 000001      MOV      #1,-(SP)
      006344 010600      MOV      SP,R0
      006346 104415      TRAP    C:PNTX
      006350 062706 000004      ADD      #4,SP
736 006354 000207      RTS     PC            ;RETURN TO CALLER

```

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

SEQ 0042

748	006356	045	116	045	EPRT1:	.ASCIZ	'#NSA *****CHECK CABLES BETWEEN M7196 AND TRANSPORT*****'
749	006446	045	116	045	EPRT2:	.ASCIZ	'#NSA *****CHECK TRANSPORT*****'
755	006505	045	116	045	TSSRFOR:	.ASCIZ	'#NSA TSSR = #06'
756	006525	045	116	045	TEXASC:	.ASCIZ	'#NSA Extended Address Bits = #06'
757	006566	045	116	045	TCOASC:	.ASCIZ	'#NSA Termination Class Code = #T'
758	006627	045	116	045	TFCASC:	.ASCIZ	'#NSA Fatal Termination Class Code = #T'
759	006676	045	116	045	TSSDEF:	.ASCIZ	'#NSA TSSR Bits Set: #T'
760	006725	045	116	045	AMBTSSR:	.ASCIZ	'#NSA TSSR Contents Are Ambiguous'
761						.EVEN	
762	006766	007006	007031	007057	TCOCOD:	.WORD	1#,2#,3#,4#,5#,6#,7#,8#
763	007006	116	157	162	1#:	.ASCIZ	'Normal Termination'
764	007031	124	145	162	2#:	.ASCIZ	'Termination Condition'
765	007057	124	141	160	3#:	.ASCIZ	'Tape Status Alert'
766	007101	106	165	156	4#:	.ASCIZ	'Function Reject'
767	007121	122	145	143	5#:	.ASCIZ	'Recoverable Error - Tape Position One Record Down'
768	007203	122	145	143	6#:	.ASCIZ	'Recoverable Error - Tape Was Not Moved'
769	007252	125	156	162	7#:	.ASCIZ	'Unrecoverable Error'
770	007276	106	141	164	8#:	.ASCIZ	'Fatal Controller Error'
771						.EVEN	
772							
773	007326	007336	007372	007403	TSFCOD:	.WORD	1#,2#,3#,4#
774	007336	111	156	164	1#:	.ASCIZ	'Internal Diagnostic Failure'
775	007372	122	145	163	2#:	.ASCIZ	'Reserved'
776	007403	102	165	163	3#:	.ASCIZ	'Bus Interface or Sanity Check Error'
777	007447	122	145	163	4#:	.ASCIZ	'Reserved'
778						.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 007460 PRIPKT::  
796 007460 SAVREG ;SAVE THE REGISTERS  
797 007464 010005 MOV R0,R5 ;SAVE NO. OF WORDS IN PACKET  
798 007466 005737 003124 TST KTENABLE ;ABOVE 28K UNDER TEST?  
799 007472 001001 BNE 10\$ ;BR IF YES  
800 007474 005003 CLR R3 ;SET HIGH ORDER ADDRESS TO 0  
801 007476 010301 10\$: MOV R3,R1 ;COPY HIGH ORDER ADDRESS  
802 007500 010400 MOV R4,R0 ;GET LOWER ADDRESS  
803 007502 006100 ROL R0 ;SHIFT BIT 15 INTO C BIT  
804 007504 006101 ROL R1 ;AND INTO HIGH ORDER.  
805 007506 PRINTB #PKTADD,R1,R4 ;PRINT PACKET ADDRESS  
007506 010446 MOV R4,-(SP)  
007510 010146 MOV R1,-(SP)  
007512 012746 007644 MOV #PKTADD,-(SP)  
007516 012746 000003 MOV #3,-(SP)  
007522 010600 MOV SP,R0  
007524 104414 TRAP C#PNTB  
007526 062706 000010 ADD #10,SP  
806 007532 010300 15\$: MOV R3,R0 ;GET HIGH ORDER ADDRESS  
807 007534 001404 BEQ 20\$ ;BR IF NOT ABOVE 28K.  
808 007536 010401 MOV R4,R1 ;GET LOW ORDER ADDRESS  
809 007540 004737 017406 JSR PC,SETMAP ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS  
810 007544 010004 MOV R0,R4 ;GET RETURNED PAR6 ADDRESS BIAS  
811 007546 005001 20\$: CLR R1 ;SAVE WORD NUMBER  
812 007550 012402 25\$: MOV (R4)+,R2 ;GET PACKET CONTENTS  
813 007552 PRINTB #PKTFRM,R1,R2 ;PRINT THE DATA  
007552 010246 MOV R2,-(SP)  
007554 010146 MOV R1,-(SP)  
007556 012746 007606 MOV #PKTFRM,-(SP)  
007562 012746 000003 MOV #3,-(SP)  
007566 010600 MOV SP,R0  
007570 104414 TRAP C#PNTB  
007572 062706 000010 ADD #10,SP  
814 007576 005201 INC R1 ;NEXT WORD NUMBER  
815 007600 020105 CMP R1,R5 ;DONE ALL PACKET WORDS?  
816 007602 002762 BLT 25\$ ;LOOP TILL ALL DONE  
817 007604 000207 RTS PC ;RETURN

818  
819 007606 045 116 045 PKTFRM: .ASCIZ '##A Packet Word #D1##A = #06'  
820 007644 045 116 045 PKTADD: .ASCIZ '##A Packet Address = #01#05'  
821 .EVEN

```

823                               .SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR BYTE
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 007702                       PRIBXOR::
839 007702                        SAVREG                       ;SAVE THE REGISTERS
840 007706   010203               MOV       R2,R3               ;EXPECTED DATA
841 007710                       XOR       R1,R3               ;FORM THE EXCLUSIVE OR
842 007720   012700   177400       MOV       #C<377>,R0       ;BYTE MASK
843 007724   040001               BIC       R0,R1               ;SAVE LOW BYTE RECV
844 007726   040002               BIC       R0,R2               ;SAVE LOW BYTE EXPD
845 007730   040003               BIC       R0,R3               ;SAVE LOW BYTE XOR
846 007732                       PRINTB   #XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
                              MOV       R3,-(SP)
                              MOV       R1,-(SP)
                              MOV       R2,-(SP)
                              MOV       #XORBFOR,-(SP)
                              MOV       #4,-(SP)
                              MOV       SP,R0
                              TRAP      C#PNTB
                              ADD       #12,SP
847 007760   010300               MOV       R3,R0               ;R0 HAS XOR ON RETURN
848 007762   000207               RTS       PC                ;RETURN TO CALLER
849
850 007764       045       116       045 XORBFOR:       .ASCIZ '##N##A EXPD: ##03##A RECV: ##03##A XOR: ##03'
851                               .EVEN
852                               .SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR
853                               ;*
854                               ;
855                               ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
856                               ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
857                               ;
858                               ;INPUTS:
859                               ;
860                               ;       R1       RECEIVED DATA
861                               ;       R2       EXPECTED DATA
862                               ;
863                               ;OUTPUT:
864                               ;
865                               ;       R0       XOR OF EXPECTED/RECEIVED DATA
866                               ;-
867 010032                       PRIBXOR::
868 010032                        SAVREG                       ;SAVE THE REGISTERS
869 010036   010203               MOV       R2,R3               ;EXPECTED DATA
870 010040                       XOR       R1,R3               ;FORM THE EXCLUSIVE OR
871 010050                       PRINTB   #XORFOR,R2,R1,R3 ;PRINT THE MESSAGE

```

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

SEQ 0045

	010050	010346				MOV	R3, -(SP)	
	010052	010146				MOV	R1, -(SP)	
	010054	010246				MOV	R2, -(SP)	
	010056	012746	010102			MOV	@XORFOR, -(SP)	
	010062	012746	000004			MOV	@4, -(SP)	
	010066	010600				MOV	SP, R0	
	010070	104414				TRAP	C:PNTB	
	010072	062706	000012			ADD	@12, SP	
872	010076	010300				MOV	R3, R0	;R0 HAS XOR ON RETURN
873	010100	000207				RTS	PC	;RETURN TO CALLER
874								
875	010102	045	116	045	XORFOR:	.ASCIZ	'#N#A EXPD: #06#A RECV: #06#A XOR: #06'	
876						.EVEN		

```

878                           .SBTTL PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT
879
880                           ;*
881                           ;
882                           ;ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
883                           ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
884                           ;
885                           ;INPUTS:
886                           ;
887                           ;       R0       OCTAL VALUE TO CONVERT
888                           ;       R1       TABLE OF POINTERS TO ASCII EQUIVALENT
889                           ;
890                           ;-
891
892 010150                   PRIEQU:
893 010150                    SAVREG                               ;SAVE THE REGISTERS
894 010154   000207         RTS       PC                           ;RETURN TO CALLER
895
896                           .SBTTL PRIRAM - PRINT RAM ADDRESS
897                           ;*
898                           ;
899                           ;PRINT CONTROLLER RAM ADDRESS.
900                           ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
901                           ;
902                           ;INPUTS:
903                           ;
904                           ;       R4       RAM ADDRESS
905                           ;
906                           ;-
907 010156                   PRIRAM:
908 010156                    SAVREG                               ;SAVE R1-R5 UNTIL NEXT RETURN
909 010162                    PRINTB   #RAMFOR,R4                 ;PRINT RAM ADDRESS IN ERROR
                          MOV       R4,-(SP)
                          MOV       #RAMFOR,-(SP)
                          MOV       #2,-(SP)
                          MOV       SP,R0
                          TRAP     C#PNTB
                          ADD       #6,SP
910 010204   000207         RTS       PC                           ;RETURN
911
912 010206       045       116       045 RAMFOR: .ASCIZ 'N/A CONTROLLER RAM ADDRESS = #06'
913                            .EVEN

```



```

915 .SBTTL PRIADD - PRINT MEMORY ERROR ADDRESS
916 ;*
917 ;
918 ;PRINT MEMORY ADDRESS
919 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
920 ;
921 ; IMPLICIT INPUTS
922 ;
923 ; ERRHI - HIGH ORDER ADDRESS
924 ; ERRLO - LOW ORDER ADDRESS
925 ;
926 ;-
927 PRIADD: SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
928 010250 MOV ERRHI,R0 ;GET HIGH ADDRESS
929 010254 013700 002226 MOV ERRLO,R1 ;GET LOW ADDRESS
930 010260 013701 002230 MOV R1,R2 ;COPY LOW ADDRESS
931 010264 010102 ROL R1 ;SHIFT BIT 15 TO C BIT
932 010266 006101 ROL R0 ;SHIFT INTO HIGH ORDER
933 010270 006100 PRINTB #PRIA0,R0,R2 ;PRINT MEMORY ADDRESS IN ERROR
934 010272 MOV R2,-(SP)
010272 010246 MOV R0,-(SP)
010274 010046 MOV #PRIA0,-(SP)
010276 012746 010320 MOV #3,-(SP)
010302 012746 000003 MOV SP,R0
010306 010600 TRAP C#PNTB
010310 104414 ADD #10,SP
010312 062706 000010 RTS PC ;RETURN
935 010316 000207
936
937 010320 045 116 045 PRIA0: .ASCIZ '#N#A MEMORY ERROR ADDRESS = #01#05'
938 .EVEN
939
940 .SBTTL PRITADD - PRINT MEMORY TEST ADDRESS
941 ;*
942 ;
943 ;PRINT MEMORY ADDRESS
944 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
945 ;
946 ; IMPLICIT INPUTS
947 ;
948 ; ERRHI - HIGH ORDER ADDRESS
949 ; ERRLO - LOW ORDER ADDRESS
950 ;
951 ;-
952 PRITADD: SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
953 010364 MOV ERRHI,R2 ;GET HIGH ADDRESS
954 010370 013702 002226 MOV ERRLO,R1 ;GET LOW ADDRESS
955 010374 013701 002230 ;MOV R1,R2 ;COPY LOW ADDRESS
956 ;ROL R1 ;SHIFT BIT 15 TO C BIT
957 ;ROL R0 ;SHIFT INTO HIGH ORDER
958 PRINTB #PRITO,R1 ;PRINT MEMORY ADDRESS LOW IN ERROR
959 010400 MOV R1,-(SP)
010400 010146 MOV #PRITO,-(SP)
010402 012746 010446 MOV #2,-(SP)
010406 012746 000002 MOV SP,R0
010412 010600 TRAP C#PNTB
010414 104414

```

TSV3 - GLOBAL AREAS      MACRO M1113    14-JUN-84 14:17  
 PRITADD - PRINT MEMORY TEST ADDRESS

SEQ 0048

```

960 010416 062706 000006      ADD     #6,SP
    010422                PRINTB  #PRIT1,R2      ;PRINT MEMORY ADDRESS HIGH IN ERROR
    010422 010246          MOV     R2,-(SP)
    010424 012746 010511     MOV     #PRIT1,-(SP)
    010430 012746 000002     MOV     #2,-(SP)
    010434 010600          MOV     SP,R0
    010436 104414          TRAP   C#PNTB
    010440 062706 000006      ADD     #6,SP
961 010444 000207          RTS     PC              ;RETURN
962
963 010446      045      116      045 PRIT0:  .ASCIZ  'N/A MEMORY TEST ADDRESS LOW = #06'
964 010511      045      116      045 PRIT1:  .ASCIZ  'N/A MEMORY TEST ADDRESS HIGH = #06'
965                                     .EVEN

```

```

967 .SBTTL SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND
968
969 ;+
970 ;
971 ;ROUTINE TO ISSUE A SPACE RECORDS
972 ;COMMAND (FORWARD OR REVERSE)
973 ;
974 ;INPUT:
975 ;
976 ; R3 NUMBER OF RECORDS TO BE SPACED OVER
977 ; BIT15 CONTROLS DIRECTION
978 ; BIT15 = 0 IS FORWARD
979 ; BIT15 = 1 IS REVERSE
980 ;
981 ; R5 FIRST DEVICE UNIBUS ADDRESS
982 ;
983 ; REQUIRES A WRITE CHARACTERISTICS DONE PREVIOUSLY
984 ;
985 ;OUTPUT:
986 ;
987 ; CARRY SET - SPACE RECORDS COMMAND OK
988 ; CLR - SPACE RECORDS FAILED
989 ;
990 ; R0 THE CONTENTS OF R4 IS MOVED TO R0
991 ;
992 ;
993 ;IMPLICIT OUTPUT:
994 ;
995 ; TAPE HAS BEEN MOVED
996 ;
997 ;SIDE EFFECTS:
998 ;
999 ;
1000 ;-
1001
1002 010556 SPACE:: SAVREG ;SAVE THE GENERAL REGISTERS
1003 010556 MOV #500,,SDELAY ;SET UP DELAY
1004 010562 012737 000764 010750 MOV #140010,80$ ;SET UP COMMAND, SPACE FORWARD
1005 010570 012737 140010 010740 TST R3 ;CHECK FOR DIRECTION
1006 010576 005703 BMI 5$ ;BR, IF REVERSE INDICATED
1007 010600 100403 MOV R3,90$ ;LOAD UP NUMBER OF RECORDS TO SPACE
1008 010602 010337 010742 BR 10$ ;GO DO COMMAND
1009 010606 000407 5$: BIC #BIT15,R3 ;CLEAR DIRECTION BIT
1010 010610 042703 100000 MOV R3,90$ ;LOAD UP NUMBER OF RECORDS TO SPACE
1011 010614 010337 010742 BIS #BIT8,80$ ;SET REVERSE BIT IN COMMAND PACKET
1012 010620 052737 000400 010740 10$: MOV #80$,R4 ;SET UP R4 WITH PACKET ADDRESS
1013 010626 012704 010740 MOV R4,TSDB(R5) ;SEND OUT COMMAND
1014 010632 010465 000000 15$: JSR PC,WAITF ;WAIT FOR SSR
1015 010636 004737 016340 BCS 20$ ;BR, IF SSR IS SET AND OK
1016 010642 103420 DELAY 250 ;DELAY ABOUT .25 SECONDS
1017 010644 MOV #250,(PC)+
010644 012727 000250 .WORD 0
010650 000000 MOV L#DLY,(PC)+
010652 013727 002116 .WORD 0
010656 000000 DEC -6(PC)
010660 005367 177772 BNE .-4
010664 001375

```

```

010666 005367 177756            DEC    -22(PC)
010672 001367            BNE    .-20
1018 010674 005337 010750        DEC    SDELAY            ;BUMP DELAY COUNTER DOWN
1019 010700 001356            BNE    15$            ;BR, IF MORE DELAY
1020 010702 000411            BR     60$            ;BR IF TROUBLE CARRY = CLEAR
1021 010704 016501 000002        20$: MOV    TSSR(R5),R1        ;READ TSSR
1022 010710 012702 000200        MOV    @SSR,R2        ;SET UP EXPECTED
1023 010714 020201        25$: CMP    R2,R1        ;ARE THEY OK
1024 010716 001401            BEQ    40$            ;BR, IF EQUAL = OK
1025 010720 000402            BR     60$            ;TROUBLE EXIT
1026 010722 000261        40$: SEC            ;SET CARRY NO TROUBLE
1027 010724 000401            BR     70$            ;EXIT
1028 010726 000241        60$: CLC            ;CARRY CLEAR = ERROR
1029 010730        70$:            ;
1030 010730 010400            MOV    R4,R0            ;PASS PACKET ADDRESS
1031 010732 000207            RTS    PC            ;RETURN
1032                        ;
1033                        ;
1034                        ;
1035                        ;PACKET FOR SPACE COMMAND
1036                        ;
1038                        010740            . * < . + 10 > & 177770
1040                        ;
1041                        ;COMMAND WORD
1042 010740 000000        80$: .WORD            ;NUMBER OF RECORDS TO BE SPACED OVER WORD
1043                        ;
1044 010742 000000        90$: .WORD            ;
1045 010744 000000            .WORD            ;
1046 010746 000000            .WORD            ;
1047 010750 000000        SDELAY: .WORD    0            ;DELAY COUNTER
1048                        .EVEN            ;
1049                        .SBTTL    WRTCHR    - WRITE CHARACTERISTICS COMMAND

```

```

1051      ;*
1052      ;ROUTINE TO ISSUE A WRITE CHARACTERISTICS
1053      ;COMMAND SO THAT OTHER COMMANDS WILL BE ACCEPTED
1054      ;
1055      ;INPUT:
1056      ;      R4      ADDRESS OF PACKET FROM TEST
1057      ;      R5      FIRST DEVICE UNIBUS ADDRESS
1058      ;      REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
1059      ;
1060      ;OUTPUT:
1061      ;      R0      TSSR CONTENTS
1062      ;      CARRY   SET - WRITE CHARACTERISTICS COMMAND OK
1063      ;             CLR - WRITE CHARACTERISTICS FAILED
1064      ;
1065      ;IMPLICIT OUTPUT:
1066      ;
1067      ;      MESSAGE BUFFER AND OTHER BUFFERS ALL SET UP
1068      ;      SOFTWARE SWITCHES SET AS FOLLOWS:
1069      ;             EXTFEA = EXTENDED FEATURES PRESENT
1070      ;             BENBSW = BUFFER ENABLE SWITCH ON OR OFF
1071      ;
1072      ;SIDE EFFECTS:
1073      ;-
1074      WRTCHR::
1075      SAVREG      ;SAVE THE GENERAL REGISTERS
1076      CLR        BENBSW      ;CLEAR BUFFER ENABLE SWITCH
1077      CLR        EXTFEA     ;CLEAR EXTENDED FEATURES SW SWITCH
1078      MOV        R4,TSDB(R5) ;SEND OUT COMMAND
1079      JSR        PC,CHKTSSR ;WAIT FOR SSR
1080      BCS        20$        ;BR, IF SSR IS SET AND OK
1081      BR         60$        ;BR IF TROUBLE CARRY = CLEAR
1082      MOV        TSSR(R5),R1 ;READ TSSR
1083      MOV        @SSR,R2    ;SET UP EXPECTED
1084      BIT        @OFL,R1    ;WAS OFF LINE SET IN TSSR
1085      BEQ        25$        ;BR, IF NO OFL SET
1086      BIS        @OFL,R2    ;MAKE THEM LOOK ALIKE
1087      CMP        R2,R1     ;ARE THEY OK
1088      BEQ        40$        ;BR, IF EQUAL = OK
1089      BR         60$        ;TROUBLE EXIT
1090      ADD        @B.,R4     ;POINT TO WRT CHARA DATA PACKET
1091      MOV        (R4),R3    ;GET ADDRESS OF MESSAGE BUFFER
1092      BIT        @X2.EXTF,XST2(R3) ;EXTENDED FEATURES BIT SET?
1093      BEQ        45$        ;BR IF NO
1094      INC        EXTFEA     ;SET EXTENDED FEATURES SW SWITCH
1095      BR         45$
1096      BIT        @X2.BUFE,XST2(R3) ;BUFFER ENABLE SWITCH SET
1097      BEQ        50$        ;BR, IF SWITCH NOT SET
1098      INC        BENBSW    ;SET SOFTWARE SWITCH FOR ENABLED
1099      BR         50$
1100      SEC        ;SET CARRY NO TROUBLE
1101      BR         70$        ;EXIT
1102      CLC        ;CARRY CLEAR = ERROR
1103      MOV        TSSR(R5),R0 ;RETURN TSSR CONTENTS
1104      RTS        PC        ;RETURN

```

```

1106          .SBTTL  REWIND - POSITION TAPE (REWIND) COMMAND
1107          ;*
1108          ;
1109          ; THIS ROUTINE WILL REWIND THE SELECTED TAPE.
1110          ;
1111          ; CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
1112          ; TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
1113          ; SSR TO SET IN THE TSSR
1114          ;
1115          ;
1116          ; CALLING SEQUENCE:
1117          ;
1118          ; DO A SOFT INIT
1119          ; DO A WRITE CHARACTERISTICS
1120          ; JSR      PC,REWIND
1121          ;
1122          ; INPUT:
1123          ;
1124          ; R5      FIRST DEVICE UNIBUS ADDRESS
1125          ;
1126          ;
1127          ; OUTPUT
1128          ;
1129          ; R0      THE CONTENTS OF R4 IS PASSED TO R0
1130          ;
1131          ;
1132          ; -
1133          REWIND::
1134          SAVREG          ;SAVE R1-R5 UNTIL NEXT RETURN
1135          MOV             #RWPACK,R4          ;GET PACKET ADDRESS
1136          MOV             R4,TSDB(R5)        ;SEND PACKET ADDRESS TO EXECUTE
1137          MOV             #360.,R3           ;ENOUGH TIME FOR 2400' REEL TO REWIND
1138          JSR             PC,WAITF           ;WAIT FOR SSR TO SET
1139          BCS             20$                ;LEAVE WHEN SSR IS SET
1140          DELAY           250.              ;WAIT FOR .25 SECONDS
1141          MOV             #250.,(PC)+        ;
1142          .WORD            0
1143          MOV             L#DLY,(PC)+
1144          .WORD            0
1145          DEC             -6(PC)
1146          BNE             .-4
1147          DEC             -22(PC)
1148          BNE             .-20
1149          DEC             R3
1150          BNE             10$                ;BUMP COUNTER DOWN
1151          CLC
1152          ;KEEP GOING
1153          ;CLEAR CARRY TO SET ERROR
1154          MOV             R4,R0              ;PASS THE PACKET ADDRESS
1155          RTS             PC                 ;RETURN
1156          ;
1157          ;
1158          RWPACK:         .=<.+10>E177770
1159          .WORD            102010           ;POSTION COMMAND (REWIND)
1160          .WORD            0                ;NOT USED

```

1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
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  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205

011204  
011204  
011210 012701 002232  
011214 012702 000201  
011220 005003  
011222 004737 016426  
011226 112765 000000 000000  
011234 004737 016426  
011240 010265 000000  
011244 004737 016426  
011250 116511 000000  
011254 122124  
011256 001401  
011260 005203  
011262 005202  
011264 020227 000210  
011270 003761  
011272 005703  
011274 001402  
011276 000241  
011300 000401  
011302 000261  
011304 012737 000010 002272  
011312 000207

```
.SBTTL CKRAM - COMPARE RAM TO I/O PACKET
;*
;
;ROUTINE TO READ THE FIRST 8 BYTES FROM RAM
;MEMORY AND COMPARE THIS DATA TO A COMMAND PACKET.
;
;INPUT:
;
;      R4      ADDRESS OF THE COMMAND PACKET
;      R5      FIRST DEVICE UNIBUS ADDRESS
;
;OUTPUT:
;
;      CARRY   SET - RAM MATCHES PACKET
;             CLR - RAM DOES NOT MATCH PACKET
;
;IMPLICIT OUTPUT:
;
;      THE TABLE RAMDATA IS FILLED WITH THE
;      DATA HELD IN RAM.
;      RAMSIZ IS SET TO 8. FOR PRAMPKT ROUTINE
;
;SIDE EFFECTS:
;
;      THE SUBSYSTEM IS LEFT IN MAINTENANCE MODE
;
;-
```

```
CKRAM::
  SAVREG
  MOV    @RAMDATA,R1      ;SAVE THE GENERAL REGISTERS
  MOV    @RMPKTBEGR,R2   ;ADDRESS TO SAVE THE RAM DATA
  CLR    R3               ;BYTE ADDRESS OF FIRST RAM DATA
  JSR    PC,CHKTSSR      ;CLEAR THE ERROR FLAG
  MOVB   #0,TSDB(R5)     ;WAIT FOR SSR
  JSR    PC,CHKTSSR      ;SET MAINTENANCE MODE
  MOV    R2,TSDB(R5)     ;WAIT FOR SSR TO SET
  JSR    PC,CHKTSSR      ;SELECT NEXT RAM ADDRESS
  MOVB   TSBA(R5),(R1)   ;WAIT FOR SSR TO SET
  CMPB   (R1)+,(R4)+    ;READ THE RAM DATA
  BEQ    20$             ;COMPARE TO EXPECTED
  INC    R3               ;BRANCH IF OK
  INC    R2               ;SET ERROR FLAG
  CMP    R2,@RMPKTEND    ;ADDRESS OF NEXT RAM LOCATION
  BLE    10$             ;REACHED END YET ?
  TST    R3               ;BRANCH TILL ALL READ
  BEQ    30$             ;WAS AN ERROR FOUND ?
  CLC
  BR     50$            ;BRANCH IF NOT
  SEC
  MOV    @8.,RAMSIZ     ;CLEAR CARRY TO SHOW ERROR
  RTS    PC              ;AND EXIT
  ;SHOW GOOD COMPARE
  ;SETUP RAMSIZ FOR PRAMPKT ROUTINE
  ;RETURN
```

```

1207 .SBTTL CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
1208 ;*
1209 ;
1210 ;ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
1211 ;MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
1212 ;
1213 ;INPUT:
1214 ;
1215 ; R4 ADDRESS OF THE CHARACTERISTICS DATA
1216 ; R5 FIRST DEVICE UNIBUS ADDRESS
1217 ;
1218 ;OUTPUT:
1219 ;
1220 ; CARRY SET - RAM MATCHES PACKET
1221 ; CLR - RAM DOES NOT MATCH PACKET
1222 ;
1223 ;IMPLICIT OUTPUT:
1224 ;
1225 ; THE TABLE RAMDATA IS FILLED WITH THE
1226 ; DATA HELD IN RAM.
1227 ; RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
1228 ;
1229 ;SIDE EFFECTS:
1230 ;
1231 ; THE SUBSYSTEM IS LEFT IN MAINTENANCE MODE
1232 ;
1233 CKRAM2::
1234 SAVREG ;SAVE THE GENERAL REGISTERS
1235 MOV @RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
1236 MOV @RMCHBEG,R2 ;BYTE ADDRESS OF FIRST RAM DATA
1237 CLR R3 ;CLEAR THE ERROR FLAG
1238 JSR PC,CHKTSSR ;WAIT FOR SSR
1239 MOVB @C,TSDB(R5) ;SET MAINTENANCE MODE
1240 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
1241 MOV R2,TSDB(R5) ;SELECT NEXT RAM ADDRESS
1242 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
1243 MOVB TSBA(R5),(R1) ;READ THE RAM DATA
1244 CMPB (R1)+,(R4)+ ;COMPARE TO EXPECTED
1245 BEQ 20$ ;BRANCH IF OK
1246 INC R3 ;SET ERROR FLAG
1247 INC R2 ;ADDRESS OF NEXT RAM LOCATION
1248 MOV @8.,RAMSIZ ;ASSUME EXTFEA NOT SET
1249 TST EXTFEA ;IS THE SOFTWARE EXTENDED FEATURES SET
1250 BEQ 25$ ;BR, IF NOT SET
1251 MOV @10.,RAMSIZ ;SET RAMSIZ FOR EXTEND FEATURES
1252 CMP R2,@RMCHEND ;AT END OF EXTENDED BUFFER
1253 BLE 10$ ;BR, IF NOT AT END YET
1254 BR 27$ ;AT END BRANCH
1255 CMP R2,@RMCHEND-2 ;REACHED END YET ?
1256 BLE 10$ ;BRANCH TILL ALL READ
1257 TST R3 ;WAS AN ERROR FOUND ?
1258 BEQ 30$ ;BRANCH IF NOT
1259 CLC ;CLEAR CARRY TO SHOW ERROR
1260 BR 50$ ;AND EXIT
1261 SEC ;SHOW GOOD COMPARE
1262 RTS PC ;RETURN

```



```

1264 .SBTTL CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS
1265 ;*
1266 ;
1267 ;ROUTINE TO COMPARE A WRITE CHARACTERISTICS EXPD AND RECV
1268 ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
1269 ;ERROR PRINT ROUTINES.
1270 ;
1271 ;INPUT:
1272 ;
1273 ; R0 RECV MESSAGE BUFFER HIGH ORDER ADDRESS
1274 ; R1 RECV MESSAGE BUFFER LOW ORDER ADDRESS
1275 ; R2 EXPD MESSAGE BUFFER ADDRESS
1276 ;OUTPUT:
1277 ;
1278 ; CARRY SET - MESSAGE BUFFERS MATCH
1279 ; CLR -MESSAGE BUFFERS DON'T MATCH
1280 ;
1281 ;IMPLICIT OUTPUT:
1282 ;
1283 ; EXPMSG BUFFER IS SET TO EXPD DATA
1284 ; RECVMSG BUFFER IS SET TO RECV DATA
1285 ; RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
1286 ; RCVLOADD SET TO LOW ORDER ADDRESS OF RECV
1287 ;
1288 ;-
1289 CKMSG::
1290 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
1291 MOV R0,RCVHIADD ;SAVE RECV HIGH ADDRESS
1292 MOV R1,RCVLOAD ;SAVE RECV LOW ADDRESS
1293 TST KTENABLE ;TESTING ABOVE 28K?
1294 BEQ 10$ ;BR IF NO
1295 JSR PC,SETMAP ;RETURN ADDRESS BIASED TO PAR6 IN R0
1296 MOV R0,R1 ;GET RETURNED ADDRESS BIASED TO PAR6
1297 10$: CLR R4 ;WORD IN BUFFER
1298 CLR R3 ;CLEAR ERROR SEEN FLAG
1299 MOV R2,R5 ;GET EXPD BUFFER ADDRESS
1300 15$: MOV (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
1301 MOV (R1),RECVMSG(R4) ;SAVE RECV FOR ERROR REPORT
1302 CMP (R2)*,(R1)* ;EXPD EQUAL RECV?
1303 BEQ 25$ ;BR IF YES
1304 INC R3 ;SET ERROR SEEN FLAG
1305 25$: ADD #2,R4 ;POINT TO NEXT WORD ADDRESS
1306 CMP R4,#14 ;DONE FIRST 7 WORDS?
1307 BLE 15$ ;BR IF NO
1308 30$: BIT #X2.EXTF,XST2(R5) ;IS EXTENDED FEATURES SET IN EXPD?
1309 BEQ 50$ ;BR IF NO
1310 CMP R4,#16 ;DONE EXTENDED FEATURES WORD?
1311 BLE 15$ ;BR IF NO
1312 50$: TST R3 ;ANY ERRORS SEEN?
1313 BEQ 55$ ;BR IF NO
1314 CLC ;SET FAILURE
1315 BR 60$ ;
1316 55$: SEC ;SET SUCCESS
1317 60$: RTS PC ;RETURN
    
```

```

1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341 011570
1342 011570
1343 011574 020327 000144
1344 011600 003412
1345 011602 012703 000144
1346 011606
      011606 012746 011722
      011612 012746 000001
      011616 010600
      011620 104417
      011622 062706 000004
1347 011626 010037 002274
1348 011632 010137 002276
1349 011636 005737 003124
1350 011642 001403
1351 011644 004737 017406
1352 011650 010001
1353 011652 005004
1354 011654 005005
1355 011656 111264 002312
1356 011662 111164 002456
1357 011666 122221
1358 011670 001401
1359 011672 005205
1360 011674 062704 000001
1361 011700 020403
1362 011702 002001
1363 011704 000764
1364 011706 005705
1365 011710 001402
1366 011712 000241
1367 011714 000401
1368 011716 000261
1369 011720 000207

```

```

.SBTTL CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS
;*
;ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
;ERROR PRINT ROUTINES.
;
;INPUT:
;      R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
;      R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
;      R2      EXPD MESSAGE BUFFER ADDRESS
;      R3      NUMBER OF BYTES TO COMPARE
;
;OUTPUT:
;      CARRY   SET - MESSAGE BUFFERS MATCH
;             CLR - MESSAGE BUFFERS DON'T MATCH
;
;IMPLICIT OUTPUT:
;      EXPMSG      BUFFER IS SET TO EXPD DATA
;      RECVMSG     BUFFER IS SET TO RECV DATA
;      RCVHIADD    SET TO HIGH ORDER ADDRESS OF RECV
;      RCVLOADD    SET TO LOW ORDER ADDRESS OF RECV
;-
CKMSG2::
      SAVREG                      ;SAVE R1-R5 UNTIL NEXT RETURN
      CMP      R3,#RECVMSG-EXPMSG;800 IS COUNT ABOVE MAX ALLOWED?
      BLE     5$,
      MOV     #RECVMSG-EXPMSG,R3;800
      PRINTF  #DEBUGMSG          ;800
      MOV     #DEBUGMSG,-(SP)
      MOV     #1,-(SP)
      MOV     SP,R0
      TRAP   C:PNTF
      ADD     #4,SP
5$:   MOV     R0,RCVHIADD          ;SAVE RECV HIGH ADDRESS
      MOV     R1,RCVLOAD          ;SAVE RECV LOW ADDRESS
      TST    KTENABLE            ;TESTING ABOVE 28K?
      BEQ    10$,
      JSR   PC,SETHAP           ;RETURN ADDRESS BIASED TO PAR6 IN R0
      MOV     R0,R1             ;GET RETURNED ADDRESS BIASED TO PAR6
10$:  CLR     R4
      CLR     R5
      MOVB   (R2),EXPMSG(R4)    ;SAVE EXPD FOR ERROR REPORT
      MOVB   (R1),RECVMSG(R4)  ;SAVE RECV FOR ERROR REPORT
      CMPB   (R2)+,(R1)+       ;EXPD EQUAL RECV?
      BEQ    25$,
      INC    R5                 ;SET ERROR SEEN FLAG
25$:  ADD     #1,R4              ;POINT TO NEXT BYTE
      CMP    R4,R3              ;DONE ALL BYTES?
      BGE    50$,
      BR     15$,
50$:  TST    R5                 ;ANY ERRORS SEEN?
      BEQ    55$,
      CLC
      BR     60$,
55$:  SEC
60$:  RTS      PC              ;SET SUCCESS
      ;RETURN

```

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

SEQ 0057

```

1371 011722      120      122      117  DEBUGMSG: .ASCIZ 'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-';@@D
1372 012012      045      116      045  FERCM:  .ASCII /N#A ***/
1373 012023      040      040      124  ERCM:   .ASCIZ / TSSR ERROR CODE REC'D = /
1374 012056      056      056      056  SIMSG:  .ASCIZ /... AFTER DOING SOFT INIT/
1375 012111      124      105      123  TINERR: .ASCIZ /TEST: .../
1376                                     .EVEN
1377                                     ;*
1378                                     ;
1379                                     ;PRINT ROUTINE TO FATAL SOFT INIT ERRORS
1380                                     ;
1381                                     ;INPUT:
1382                                     ;
1383                                     ;      R1      CONTENTS OF TSSR AT ERROR
1384                                     ;
1385                                     ;SIDE EFFECTS:
1386                                     ;
1387                                     ;      EXECUTES DROP UNIT TO CEASE TESTING
1388                                     ;
1389                                     ;-
1390
1391 012124          BGNMSG  SFIMSG
1392 012124          SFIMSG:: JSR    PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
1393 012130          004737 017272 JSR    PC,CKDROP      ;DROP UNIT, IF ALLOWED
1394 012134          ENDMSG
1395 012134          L10003: TRAP  C#MSG
1396                                     ;
1397                                     ;*
1398                                     ;PRINT ROUTINE TO PRINT THE CONTENTS OF
1399                                     ;TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
1400                                     ;
1401                                     ;INPUTS:
1402                                     ;
1403                                     ;      R1      TSSR CONTENTS
1404                                     ;      R4      ADDRESS OF COMMAND PACKET
1405                                     ;
1406                                     ;-
1407 012136          BGNMSG  PKTSSR
1408 012136          PKTSSR:: JSR    PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
1409 012142          004737 000004 MOV    #4,R0          ;NO. OF WORDS IN PACKET
1410 012146          004737 007460 JSR    PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
1411 012152          ENDMSG
1412 012152          L10004: TRAP  C#MSG
1413 012152          104423

```

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

SEQ 0058

```

1413
1414
1415
1416
1417
1418
1419
1420
1421
1422 012154
      012154
1423 012154 004737 006022
1424 012160 012700 000002
1425 012164 004737 007460
1426 012170
      012170
      012170 104423
1427
1428
1429
1430
1431
1432
1433
1434 012172
      012172
1435 012172 004737 006022
1436 012176
      012176
      012176 104423
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449 012200
      012200
1450 012200 004737 006022
1451 012204 010200
1452 012206 010301
1453 012210 004737 014332
1454 012214
      012214
      012214 104423

;
;PRINT ROUTINE TO PRINT THE CONTENTS OF
;TSSR AND A GET STATUS COMMAND PACKET.
;
;INPUTS:
;
; R1 TSSR CONTENTS
; R4 ADDRESS OF COMMAND PACKET
;-
BGNMSG PKTGETS
PKTGETS::
JSR PC,PRITSSR ;PRINT THE CONTENTS OF TSSR REGISTER
MOV #2,R0 ;NO. OF WORDS IN GET STATUS PACKET
JSR PC,PRIPKT ;PRINT THE CONTENTS OF COMMAND PACKET
ENDMSG
L10005:
TRAP C#MSG

;
;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
;
;INPUTS:
;
; R1 TSSR CONTENTS
; R4 ADDRESS OF COMMAND PACKET
;-
BGNMSG SFFMSG
SFFMSG::
JSR PC,PRITSSR ;PRINT CONTENTS OF TSSR REGISTER
ENDMSG
L10006:
TRAP C#MSG
.SBTTL PKTMES - PRINT TSSR AND MESSAGE BUFFER

;
;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
;BUFFER FOR ERROR REPORTS
;
;INPUTS:
;
; R1 CONTENTS OF TSSR
; R2 LOW ORDER MESSAGE BUFFER
; R3 HIGH ORDER MESSAGE BUFFER ADDRESS
; NOTE: R3 IS IGNORED IF KENABLE FLAG IS CLEAR
;-
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

```

```

1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468 012216
      012216
1469 012216 004737 010364
1470 012222 016501 000002
1471 012226 004737 006022
1472 012232
      012232
      012232 104423
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486 012234
      012234
1487 012234 012700 000007
1488 012240 005737 002216
1489 012244 001402
1490 012246 012700 000010
1491 012252 004737 014642
1492 012256
      012256
      012256 104423
  
```

```

      .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
      ;      RECMSG - 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
  
```

```

1494 .SBTTL FIFEXP - PRINT FIFO EXP/RECV DATA
1495 ;*
1496 ;
1497 ;PRINT ROUTINE TO PRINT FIFO EXP/RECV DATA
1498 ;
1499 ; R1 - BYTE COUNT
1500 ;
1501 ;IMPLICIT INPUTS:
1502 ;
1503 ; EXPMSG - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
1504 ; RECMMSG - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
1505 ;
1506 012260 BGNMSG FIFEXP
012260 FIFEXP::
1507 012260 PRINTX #FIF1MSG,R1 ;PRINT BYTES TRANSFERRED
012260 MOV R1,-(SP)
012262 010146 MOV #FIF1MSG,-(SP)
012266 012746 012332 MOV #2,-(SP)
012272 010600 MOV SP,R0
012274 104415 TRAP C#PNTX
012276 062706 000006 ADD #6,SP
1508 012302 PRINTX #FIF2MSG ;PRINT HEADER MSG
012302 012746 012401 MOV #FIF2MSG,-(SP)
012306 012746 000001 MOV #1,-(SP)
012312 010600 MOV SP,R0
012314 104415 TRAP C#PNTX
012316 062706 000004 ADD #4,SP
1509 012322 010100 MOV R1,R0 ;GET BYTE COUNT
1510 012324 004737 015212 JSR PC,PRBYTEXP ;PRINT FIFO BYTES IN ERROR
1511 012330 ENDMSG
012330 L10012:
012330 104423 TRAP C#MSG
1512 012332 045 116 045 FIF1MSG: .ASCIZ '#N#A NUMBER OF BYTES TRANSFERRED = #D2'
1513 012401 045 116 045 FIF2MSG: .ASCIZ '#N#A FIFO DATA BYTES IN ERROR:'
1514 .EVEN
    
```

```

1516 .SBTTL MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
1517 ;*
1518 ;
1519 ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
1520 ;
1521 ;
1522 ;IMPLICIT INPUTS:
1523 ;
1524 ; EXPMSG - EXPECTED MESSAGE BUFFER
1525 ; RECMSG - RECEIVED MESSAGE BUFFER
1526 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1527 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1528 ;
1529 012440 BGNMSG MSGSTAT
012440 MSGSTAT:
1530 012440 012701 012502 MOV #STATCOD,R1 ;ASCII ADDRESS TABLE
1531 012444 012100 10$: MOV (R1)+,R0 ;DONE ALL MSG LINES?
1532 012446 001410 BEQ 20$ ;BR IF YES
1533 012450 PRINTX R0 ;PRINT STATUS BIT NAMES
012450 010046 MOV R0,-(SP)
012452 012746 000001 MOV #1,-(SP)
012456 010600 MOV SP,R0
012460 104415 TRAP C#PNTX
012462 062706 000004 ADD #4,SP
1534 012466 000766 BR 10$ ;DO ANOTHER MSG LINE
1535 012470 012700 000012 20$: MOV #10,R0 ;NUMBER OF WORDS IN A READ STATUS BUFFER
1536 012474 004737 014642 JSR PC,PRMSGEXP ;PRINT EXPD/RECV MESSAGE BUFFERS
1537 012500 ENDMSG
012500 L10013:
012500 104423 TRAP C#MSG
1538
1539 012502 012520 012562 012653 STATCOD: .WORD 1$,2$,3$,4$,5$,6$,0
1540 012520 045 116 045 1$: .ASCIZ 'N/A Tape Bus Signals in Word #8:'
1541 012562 045 116 045 2$: .ASCIZ 'N/A PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>'
1542 012653 045 116 045 3$: .ASCIZ 'N/A IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>'
1543 012744 045 116 045 4$: .ASCIZ 'N/A IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>'
1544 013035 045 116 045 5$: .ASCIZ 'N/A Tape Bus Signals in Word #9:'
1545 013077 045 116 045 6$: .ASCIZ 'N/A DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>'
1546 .EVEN
1547

```

```

1549                                .SBTTL MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
1550                                ;*
1551                                ;
1552                                ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
1553                                ;
1554                                ;IMPLICIT INPUTS:
1555                                ;
1556                                ;       EXPMSG - EXPECTED MESSAGE BUFFER
1557                                ;       RCVMSG - RECEIVED MESSAGE BUFFER
1558                                ;       RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1559                                ;       RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1560                                ;-
1561                                BGNMSG MSGLOOP
1562                                MSGLOOP:
1563                                10$: MOV     #LOOPCOD,R1        ;ASCII ADDRESS TABLE
1564                                BEQ     20$                ;DONE ALL MSG LINES?
1565                                PRINTX R0                ;BR IF YES
1566                                MOV     R0,-(SP)         ;PRINT STATUS BIT NAMES
1567                                MOV     #1,-(SP)
1568                                MOV     SP,R0
1569                                TRAP    C#PNTX
1570                                ADD     #4,SP
1571                                BR     10$                ;DO ANOTHER MSG LINE
1572                                20$: MOV     #10,R0        ;NUMBER OF WORDS IN A READ STATUS BUFFER
1573                                JSR     PC,PRMSGEXP       ;PRINT EXPD/RCV MESSAGE BUFFERS
1574                                ENDMSG
1575                                L10014:
1576                                TRAP    C#MSG
1577                                LOOPCOD:        .WORD   1$,2$,3$,4$,5$,6$,7$,0
1578                                1$: .ASCIZ 'NSA Tape Bus Loopback Signals in Word #8:'
1579                                2$: .ASCIZ 'NSA        PARERR<15>                IRESV2<14>        IRESV1<13>'
1580                                3$: .ASCIZ 'NSA        IHISP=>IEOT<12>        IWRT=>IIDENT<11>        IREV  =>ICER  <10>'
1581                                4$: .ASCIZ 'NSA        IWMF =>IFMK<09>        IEDIT=>IHER  <08>        IFAD  =>ISPEED<07>'
1582                                5$: .ASCIZ 'NSA        ITADO=>IRDY<06>        ITAD1=>IONL  <05>        IERASE=>ILDP  <04>'
1583                                6$: .ASCIZ 'NSA        IREW =>IDBY<03>        IRWU =>IRWD  <02>        IFEN  =>IFBY  <01>'
1584                                7$: .ASCIZ 'NSA        IGO  =>IFPT<00>'
1585                                .EVEN

```



```

1581 .SBTTL MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
1582 ;*
1583 ;
1584 ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
1585 ;
1586 ;
1587 ;IMPLICIT INPUTS:
1588 ;
1589 ; EXPMSG - EXPECTED MESSAGE BUFFER
1590 ; RECMG - RECEIVED MESSAGE BUFFER
1591 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1592 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1593 ;
1594 014032 BGNMSG MSGSUB
014032 MSGSUB::
1595 014032 012700 000012 MOV #10.,R0 ;SIZE OF WRITE SUBSYSTEM BUFFER
1596 014036 004737 014642 JSR PC,PRMSGEXP ;PRINT EXPD/RECV MESSAGE BUFFERS
1597 014042 ENDMG
014042 L10015:
014042 104423 TRAP C#MSG

1598 .SBTTL MEMADD - PRINT MEMORY ADDRESS DATA ERROR
1599 ;*
1600 ;
1601 ;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
1602 ;
1603 ;IMPLICIT INPUTS:
1604 ;
1605 ; ERRHI - MEMORY ERROR HIGH ORDER ADDRESS
1606 ; ERRLO - MEMORY ERROR LOW ORDER ADDRESS
1607 ; EXP - EXPECTED DATA
1608 ; RECV - RECEIVED DATA
1609 ;
1610 ;
1611 014044 BGNMSG MEMADD
014044 MEMADD::
1612 014044 004737 010250 JSR PC,PRIADD ;PRINT MEMORY ADDRESS IN ERROR
1613 014050 013701 002222 MOV EXPD,R1 ;GET EXPD DATA
1614 014054 013702 002224 MOV RECV,R2 ;GET RECEIVED DATA
1615 014060 004737 010032 JSR PC,PRIXOR ;PRINT EXPD/RECV
1616 014064 ENDMG
014064 L10016:
014064 104423 TRAP C#MSG

```

```

1618 .SBTTL PRAMPKT - PRINT RAM AND PACKET DATA
1619 ;*
1620 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
1621 ;WHEN THE RAM DATA DOES NOT MATCH.
1622 ;
1623 ;INPUTS:
1624 ;
1625 ; R4 POINTER TO COMMAND PACKET
1626 ;IMPLICIT INPUTS:
1627 ; RAMDATA DATA AS READ FROM THE RAM
1628 ; RAMSIZ NUMBER OF BYTES IN PACKET
1629 ; IF RAMSIZ=0 THEN DEFAULT TO 8.
1630 ;
1631 ;IMPLICIT OUTPUTS:
1632 ; RAMSIZ SET TO 0
1633 ;
1634 PRAMPKT:
1635 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
1636 MOV #RAMDATA,R1 ;DATA FROM THE RAM
1637 CLR R2 ;INIT BYTE NUMBER
1638 5$: CMPB (R1)+,(R4)+ ;COMPARE EXPECTED, RECEIVED
1639 BNE 7$ ;BR IF NO MATCH
1640 FORCERROR 7$,NOTSSR
1641 BR 10$ ;BBD
1642 7$: MOVB -1(R1),R5 ;GET RECV RAM DATA
1643 MOVB -1(R4),R3 ;GET EXPD PACKET DATA
1644 XOR R5,R3 ;XOR EXPD/RECV
1645 BIC #177400,R3 ;LOW BYTE ONLY
1646 MOVB -1(R1),RECV ;GET RECEIVED RAM DATA
1647 MOVB -1(R4),EXPD ;GET EXPECTED RAM DATA
1648 PRINTB #RAMASC,R2,RECV,EXPD,R3
1649 MOV P3,-(SP)
1650 MOV EXPD,-(SP)
1651 MOV RECV,-(SP)
1652 MOV R2,-(SP)
1653 MOV #RAMASC,-(SP)
1654 MOV #5,-(SP)
1655 MOV SP,R0
1656 TRAP C#PNTB
1657 ADD #14,SP
1658 10$: INC R2 ;UPDATE BYTE COUNT
1659 TST RAMSIZ ;DEFAULT TO 8.?
1660 BEQ 15$ ;BR IF YES
1661 CMP R2,RAMSIZ ;DONE ALL BYTES?
1662 BLE 5$ ;BR IF NO
1663 BR 25$
1664 15$: CMP R2,#8. ;DONE DEFAULT NUMBER OF BYTES?
1665 BLT 5$ ;BR IF NO
1666 20$: CLR RAMSIZ ;SET DEFAULT RAMSIZ
1667 25$: RTS PC ;RETURN
1668
1669 045 116 045 RAMASC: .ASCIZ '##A BYTE: #D2#A RAM: #03#A Packet: #03#A XOR:#03'
1670 .EVEN
1671

```

```

1663          .SBTTL PRMESS - PRINT CONTENTS OF MESSAGE BUFFER
1664          ;;
1665          ; THIS ROUTINE PRINTS THE CONTENTS OF
1666          ; THE 7 OR 8 WORD MESSAGE BUFFER RETURNED BY THE TSV-05.
1667          ;
1668          ; INPUT:
1669          ;     R0     LOW ORDER ADDRESS OF MESSAGE BUFFER
1670          ;     R1     HIGH ORDER ADDRESS OF MESSAGE BUFFER
1671          ;     NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
1672          ; THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
1673          ;
1674 014332 PRMESS: SAVREG          ;SAVE THE REGISTERS
1675 014336 010005      MOV      R0,R5          ;SAVE LOW ORDER ADDRESS
1676 014340 005737 003124  TST      KTENABLE      ;ADDRESS ABOVE 28K?
1677 014344 001001      BNE      10$          ;BR IF YES
1678 014346 005001      CLR      R1          ;SET HIGH ORDER ADDRESS TO 0
1679 014350 010103 10$:  MOV      R1,R3          ;SAVE HIGH ORDER ADDRESS
1680 014352 006100      ROL      R0          ;SHIFT BIT15 TO C BIT
1681 014354 006101      ROL      R1          ;SHIFT TO HIGH ORDER FOR PRINTOUT
1682 014356      PRINTX @PROASC,R1,R5 ;PRINT MESSAGE BUFFER ADDRESS
1683 014356 010546      MOV      R5,-(SP)
1684 014360 010146      MOV      R1,-(SP)
1685 014362 012746 014510  MOV      @PROASC,-(SP)
1686 014366 012746 000003  MOV      @3,-(SP)
1687 014372 010600      MOV      SP,R0
1688 014374 104415      TRAP    C$PNTX
1689 014376 062706 000010  ADD      @10,SP
1690 014402      PRINTX @PRIASC          ;PRINT HEADER FOR CONTENTS
1691 014402 012746 014555  MOV      @PRIASC,-(SP)
1692 014406 012746 000001  MOV      @1,-(SP)
1693 014412 010600      MOV      SP,R0
1694 014414 104415      TRAP    C$PNTX
1695 014416 062706 000004  ADD      @4,SP
1696 014422 005004      CLR      R4          ;NUMBER OF THE NEXT WORD
1697 014424 010501      MOV      R5,R1          ;COPY LOW ORDER ADDRESS
1698 014426 010300      MOV      R3,R0          ;COPY HIGH ORDER ADDRESS
1699 014430 001403      BEQ     20$          ;BR IF NOT ABOVE 28K
1700 014432 004737 017406  JSR     PC,SETMAP      ;SETUP PAR ADDRESS IN R0
1701 014436 010005      MOV      R0,R5          ;GET PAR FORMAT ADDRESS ABOVE 28K
1702 014440      PRINTX @PRASC,R4,(R5) ;PRINT THE CONTENTS OF MEMORY BUFFER
1703 014440 012546      MOV      (R5)+,-(SP)
1704 014442 010446      MOV      R4,-(SP)
1705 014444 012746 014613  MOV      @PRASC,-(SP)
1706 014450 012746 000003  MOV      @3,-(SP)
1707 014454 010600      MOV      SP,R0
1708 014456 104415      TRAP    C$PNTX
1709 014460 062706 000010  ADD      @10,SP
1710 014464 005204      INC      R4          ;NUMBER OF THE NEXT
1711 014466 020427 000007  CMP     R4,@7          ;DONE ALL YET ?
1712 014472 003005      BGT     50$          ;BRANCH IF ALL DONE
1713 014474 002761      BLT     20$          ;PRINT FIRST 7 WORDS
1714 014476 032763 000200 000012  BIT     @X2.EXTF,XST2(R3);EXTENDED FEATUTES ON ?
1715 014504 001355      BNE     20$          ;PRINT EXTENDED STATUS WORD
1716 014506 000207      RTS     PC          ;RETURN
1717 014510 045 116 045 PROASC: .ASCIZ  '%N%A Message Buffer Address = %01%05'
1718 014555 045 116 045 PRIASC: .ASCIZ  '%N%A Message Buffer Contents:'
1719 014613 045 116 045 PRASC: .ASCIZ  '%N%A Word%D1%A: %0'

```

```

1702          .EVEN
1703          .SBTTL PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS
1704          ;*
1705          ;ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
1706          ; RO - NUMBER OF WORDS IN BUFFER
1707          ;IMPLICIT INPUTS:
1708          ; EXPMSG - EXPECTED MESSAGE BUFFER
1709          ; RECMMSG - RECEIVED MESSAGE BUFFER
1710          ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
1711          ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
1712          ;-
1713 014642 PRMSGEXP::
1714 014642 SAVREG          ;SAVE R1-R5 UNTIL NEXT RETURN
1715 014646 010005 MOV R0,R5          ;SAVE NUMBER OF WORDS
1716 014650 013700 002276 MOV RCVLOADD,R0      ;GET RECV LOW ADDRESS
1717 014654 010004 MOV R0,R4          ;COPY LOW ADDRESS
1718 014656 013701 002274 MOV RCVHIADD,R1      ;GET RECV HIGH ADDRESS
1719 014662 006100 ROL R0          ;SHIFT BIT15 TO C BIT
1720 014664 006101 ROL R1          ;SHIFT TO HIGH ORDER FOR PRINTOUT
1721 014666 PRINTX #PRMSG0,R1,R4 ;PRINT MESSAGE BUFFER ADDRESS
      014666 010446 MOV R4,-(SP)
      014670 010146 MOV R1,-(SP)
      014672 012746 015022 MOV #PRMSG0,-(SP)
      014676 012746 000003 MOV #3,-(SP)
      014702 010600 MOV SP,R0
      014704 104415 TRAP C:PNTX
      014706 062706 000010 ADD #10,SP
1722 014712 PRINTX #PRMSG1          ;PRINT HEADER FOR CONTENTS
      014712 012746 015067 MOV #PRMSG1,-(SP)
      014716 012746 000001 MOV #1,-(SP)
      014722 010600 MOV SP,R0
      014724 104415 TRAP C:PNTX
      014726 062706 000004 ADD #4,SP
1723 014732 005004 CLR R4          ;NUMBER OF THE CURRENT WORD
1724 014734 012701 002312 MOV #EXPMSG,R1      ;GET EXPD BUFFER ADDRESS
1725 014740 012702 002456 MOV #RECMMSG,R2     ;GET RECV BUFFER ADDRESS
1726 014744 011100 201: MOV (R1),R0      ;GET EXPD
1727 014746 011203 MOV (R2),R3      ;GET RECV
1728 014750 XOR R0,R3          ;XOR EXPD/RCV
1729 014760 PRINTX #PRMSG2,R4,(R1),R3
      014760 010346 MOV R3,-(SP)
      014762 012246 MOV (R2),-(SP)
      014764 012146 MOV (R1),-(SP)
      014766 010446 MOV R4,-(SP)
      014770 012746 015125 MOV #PRMSG2,-(SP)
      014774 012746 000005 MOV #5,-(SP)
      015000 010600 MOV SP,R0
      015002 104415 TRAP C:PNTX
      015004 062706 000014 ADD #14,SP
1730 015010 INC R4          ;NUMBER OF THE NEXT
1731 015012 020405 CMP R4,R5          ;DONE ALL YET?
1732 015014 002001 BGE 501          ;BR IF YES
1733 015016 000752 BR 201          ;DG ANOTHER
1734 015020 000207 501: RTS PC          ;RETURN
1735 015022 045 116 045 PRMSG0: .ASCIZ '#N#A Message Buffer Address = #01#05'
1736 015067 045 116 045 PRMSG1: .ASCIZ '#N#A Message Buffer Contents:'
1737 015125 045 116 045 PRMSG2: .ASCIZ '#N#A WORD #D2#A EXPD: #06#A RECV: #06#A XOR: #06#'
    
```

```

1739          .EVEN
1740          .SBTTL PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
1741          ;;
1742          ;
1743          ;ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS
1744          ; ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
1745          ;
1746          ; RO - NUMBER OF BYTES IN BUFFER
1747          ;
1748          ;IMPLICIT INPUTS:
1749          ;
1750          ; EXPMSG - EXPECTED MESSAGE BUFFER
1751          ; RECMSG - RECEIVED MESSAGE BUFFER
1752          ;-
1753          PRBYTEXP::
1754          SAVREG          ;SAVE R1-R5 UNTIL NEXT RETURN
1755          MOV             RO,R5          ;SAVE NUMBER OF BYTES
1756          CLR             PRMNO         ;INIT ERROR COUNT
1757          CLR             R4           ;NUMBER OF THE CURRENT BYTE
1758          MOV             #EXPMSG,R1    ;GET EXPD BUFFER ADDRESS
1759          MOV             #RECMSG,R2    ;GET RCV BUFFER ADDRESS
1760          MOV             (R1),R0      ;GET EXPD BYTE
1761          BIC             #C<377>,R0   ;CLEAR UPPER BYTE
1762          MOV             R0,PRBEXP    ;SAVE FOR ERROR REPORT
1763          MOV             (R2),R3     ;GET RCV BYTE
1764          BIC             #C<377>,R3   ;CLEAR UPPER BYTE
1765          MOV             R3,PRBREC    ;FOR ERROR REPORT
1766          XOR             R0,R3       ;XOR EXPD/RCV
1767          CMPB           (R1)*,(R2)*  ;EXPD = RCV?
1768          BEQ             30$         ;BR IF YES
1769          INC             PRMNO        ;UPDATE ERROR COUNT
1770          CMP             PRMNO,#8.    ;PRINTED 8?
1771          BHI             30$         ;BR IF YES
1772          PRINTX         #PRBMSG,R4,PRBEXP,PRBREC,R3
1773          MOV             R3,-(SP)
1774          MOV             PRBREC,-(SP)
1775          MOV             PRBEXP,-(SP)
1776          MOV             R4,-(SP)
1777          MOV             #PRBMSG,-(SP)
1778          MOV             #5,-(SP)
1779          MOV             SP,R0
1780          TRAP           C#PNTX
1781          ADD             #14,SP
1782          FORCEEXIT      50$          ;@@D
1783          BR              35$        ;@@D
1784          FORCERROR     27$,NOTSSR  ;@@D
1785          INC             R4          ;NUMBER OF THE NEXT
1786          CMP             R4,R5       ;DONE ALL YET?
1787          BGE             50$        ;BR IF YES
1788          BR              20$        ;DO ANOTHER
1789          PRINTX         #PRBTOT,PRMNO ;PRINT TOTAL ERROR COUNT
1790          MOV             PRMNO,-(SP)
1791          MOV             #PRBTOT,-(SP)
1792          MOV             #2,-(SP)
1793          MOV             SP,R0
  
```

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

SEQ 0068

```

015416 104415 TRAP C#PNTX
015420 062706 000006 ADD #6,SP
1783 015424 000207 RTS PC ;RETURN
1784
1785 015426 045 116 045 PRBMSG: .ASCIZ 'N#A BYTE #D2#A EXPD: #03#A RECV: #03#A XOR: #03'
1786 015513 045 116 045 PRBTOT: .ASCIZ 'N#A NUMBER OF BYTES IN ERROR = #D2'
1787 .EVEN
1788 015560 000000 PRBEXP: .WORD 0 ;EXPD
1789 015562 000000 PRBREC: .WORD 0 ;RECV
1790 .SBTTL EXPREC - PRINT EXPD/RECV WORD DATA
1791 ;*
1792 ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1793 ;
1794 ;INPUTS:
1795 ;
1796 ; R1 RECEIVED DATA
1797 ; R2 EXPECTED DATA
1798 ;
1799 ;-
1800
1801
1802 015564 BGNMSG EXPREC
015564 EXPREC::
1803 015564 004737 010032 JSR PC,PRIXOR ;PRINT THE DATA
1804 015570 ENDMMSG
015570 L10017:
015570 104423 TRAP C#MSG
.SBTTL EXPBREC - PRINT EXPD/RECV BYTE DATA
1805 ;*
1806 ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
1807 ;
1808 ;INPUTS:
1809 ;
1810 ; R1 RECEIVED DATA BYTE
1811 ; R2 EXPECTED DATA BYTE
1812 ;
1813 ;-
1814
1815
1816
1817
1818 015572 BGNMSG EXPBREC
015572 EXPBREC::
1819 015572 004737 007702 JSR PC,PRIBXOR ;PRINT THE DATA
1820 015576 ENDMMSG
015576 L10020:
015576 104423 TRAP C#MSG
1821 .SBTTL RAMERR - PRINT RAM AND PACKET DATA
1822 ;*
1823 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
1824 ;
1825 ;INPUTS:
1826 ;
1827 ; R4 POINTER TO COMMAND PACKET
1828 ;
1829 ;
1830 ;
1831 ;

```

TSV3 - GLOBAL AREAS MACRO M1113 14-JUN-84 14:17  
 RAMERR - PRINT RAM AND PACKET DATA

SEQ 0069

```

1832 ;IMPLICIT INPUTS:
1833 ;
1834 ;     RAMDATA     DATA AS READ FROM THE RAM
1835 ;     RAMSIZ     NUMBER OF BYTES IN PACKET
1836 ;                IF RAMSIZ=0 THEN DEFAULT TO 8.
1837 ;
1838 ;IMPLICIT OUTPUTS:
1839 ;
1840 ;     RAMSIZ SET TO 0
1841 ;
1842 ;
1843 015600 BGNMSG RAMERR
1844 015600 RAMERR:: JSR PC,PRAMPKT ;PRINT RAM/PACKET DATA
1845 015600 004737 014066 ENDMSG
1846 015604 L10021: TRAP C#MSG
1847 015604 104423
1848 ;
1849 ;     .SBTTL RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
1850 ;
1851 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
1852 ;
1853 ;INPUTS:
1854 ;
1855 ;     R4     POINTER TO COMMAND PACKET
1856 ;
1857 ;IMPLICIT INPUTS:
1858 ;
1859 ;     RAMDATA     DATA AS READ FROM THE RAM
1860 ;     RAMSIZ     NUMBER OF BYTES IN PACKET
1861 ;                IF RAMSIZ=0 THEN DEFAULT TO 8.
1862 ;     ERRHI     HIGH ORDER TEST ADDRESS
1863 ;     ERRLO     LOW ORDER TEST ADDRESS
1864 ;
1865 ;IMPLICIT OUTPUTS:
1866 ;
1867 ;     RAMSIZ SET TO 0
1868 ;
1869 015606 BGNMSG RAMTADD
1870 015606 RAMTADD:: JSR PC,PRITADD ;PRINT TEST ADDRESS
1871 015612 004737 010364 JSR PC,PRAMPKT ;PRINT RAM/PACKET DATA
1872 015616 ENDMSG
1873 015616 L10022: TRAP C#MSG
1874 015616 104423
1875 ;
1876 ;     .SBTTL RAMEXP - PRINT RAM EXPD/RECV DATA
1877 ;
1878 ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1879 ;
1880 ;INPUTS:
1881 ;
1882 ;     R1     RECEIVED DATA
1883 ;     R2     EXPECTED DATA

```

```

1883      :      R4      CONTROLLER RAM ADDRESS
1884      :-
1885
1886 015620      BGNMSG  RAMEXP
      015620      RAMEXP::
1887 015620 042701 177400      BIC      @+C<377>,R1      ;SAVE EXPD RAM DATA BYTE
1888 015624 042702 177400      BIC      @+C<377>,R2      ;SAVE EXPD RAM DATA BYTE
1889 015630 004737 010156      JSR      PC,PRIRAM      ;PRINT THE RAM ADDRESS
1890 015634 004737 010032      JSR      PC,PRIXOR      ;PRINT THE DATA
1891 015640      ENDMSG
      015640
      015640 104423
1892
1893      .SBTTL  TIMEXP - PRINT TIMER A,B AND EXP/REC
1894      ;*
1895      ;
1896      ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
1897      ;AND TIMER A,B HEADER MESSAGE
1898      ;
1899      ;INPUTS:
1900      ;
1901      ;      R1      RECEIVED DATA
1902      ;      R2      EXPECTED DATA
1903      ;
1904      ;
1905 015642      BGNMSG  TIMEXP
      015642      TIMEXP::
1906 015642      PRINTX  @TIMSGO      ;PRINT HEADER
      015642 012746 015670      MOV      @TIMSGO,-(SP)
      015646 012746 000001      MOV      @1,-(SP)
      015652 010600      MOV      SP,R0
      015654 104415      TRAP   C#PNTX
      015656 062706 000004      ADD      @4,SP
1907 015662 004737 010032      JSR      PC,PRIXOR      ;PRINT THE DATA
1908 015666      ENDMSG
      015666
      015666 104423
1909
1910 015670      045      116      045  TIMSGO: .ASCIZ  'N/A TIMER A STATUS IS IN BIT 3N/A TIMER B STATUS IS IN BIT 2'
1911      .EVEN
1912      .SBTTL  BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS
1913      ;*
1914      ;
1915      ;PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
1916      ;
1917      ;INPUTS:
1918      ;
1919      ;      R1      CONTENTS OF TSSR
1920      ;      R2      DATA WRITTEN (8 BITS)
1921      ;
1922      ;
1923      ;
1924      ;
1925 015770      BGNMSG  BADSSR
      015770      BADSSR::
1926 015770 010246      MOV      R2,-(SP)      ;SAVE DATA TRANSFERRED
1927 015772 042702 177400      BIC      @177400,R2      ;GET JUST ONE BYTE
  
```



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

SEQ 0071

1928	015776				PRINTB	@XFERASC,R2		
	015776	010246			MOV	R2,-(SP)		
	016000	012746	016030		MOV	@XFERASC,-(SP)		
	016004	012746	000002		MOV	@2,-(SP)		
	016010	010600			MOV	SP,R0		
	016012	104414			TRAP	C#PNTB		
	016014	062706	000006		ADD	@6,SP		
1929	016020	012602			MOV	(SP),R2	;RESTORE R2	
1930	016022	004737	006022		JSR	PC,PRITSSR	;DECODE TSSR CONTENTS	
1931	016026				ENDMSG			
	016026			L10025:				
	016026	104423			TRAP	C#MSG		
1932	016030	045	116	045	XFERASC:	.ASCIZ	'#N#A Data Transferred = #03'	

1934  
 1935  
 1936  
 1937  
 1938  
 1939  
 1940  
 1941  
 1942  
 1943  
 1944  
 1945  
 1946  
 1947  
 1948  
 1949  
 1950  
 1951  
 1952  
 1953  
 1954  
 1955  
 1956  
 1957  
 1958  
 1959  
 1960  
 1961  
 1962  
 1963  
 1964  
 1965  
 1966  
 1967

1968 016064  
 1969 016064  
 1970 016070 012765 000000 000002  
 1971 016076 004737 016340  
 1972 016102 016500 000002  
 1973 016106 010004  
 1974 016110 042704 176277  
 1975 016114 052704 002200  
 1976 016120 020400  
 1977 016122 001402  
 1978 016124 000241  
 1979 016126 000401  
 1980 016130 000261  
 1981 016132 000207

```

.SBTTL GLOBAL SUBROUTINES SECTION
; **
; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
; THAT ARE USED IN MORE THAN ONE TEST.
; --
.SBTTL SOFINIT - SOFT INITIALIZE OF CONTROLLER
; *
; ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
; BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
; THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
; DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
;
; INPUTS:
;
; R5 ADDRESS OF FIRST REGISTER
;
; OUTPUTS:
;
; R0 CONTENTS OF TSSR, IF ERROR
; CARRY SET IF INIT WAS OKAY
; CLEAR IF FATAL ERROR
;
; CALLING SEQUENCE:
;
; MOV #ADDRESS,R5
; JSR PC,SOFINIT
; BCS CONTINUE
; ERDF ;REPORT FATAL ERROR
; -
SOFINIT:
; SAVREG ; SAVE THE REGISTERS
MOV #0,TSSR(R5) ; DO THE INIT.
JSR PC,WAITF ; WAIT FOR SSR
MOV TSSR(R5),R0 ; GET THE TSSR REGISTER
MOV R0,R4 ; TSSR CONTENTS
BIC #C<HIADDR!OFL>,R4
BIS #SSR!NBA,R4 ; R4 HAS EXPECTED CONTENTS
CMP R4,R0 ; ONLY EXPECTED BITS SET ?
BEQ 5$ ; BRANCH IF OKAY
CLC ; CLEAR THE CARRY FOR ERROR
BR 10$ ; GO TO EXIT
5$: SEC ; SET THE CARRY BIT
10$: RTS PC ; RETURN TO CALLER
    
```

1983  
 1984  
 1985  
 1986  
 1987  
 1988  
 1989  
 1990  
 1991  
 1992  
 1993  
 1994  
 1995  
 1996  
 1997  
 1998  
 1999  
 2000  
 2001  
 2002  
 2003 016134  
 2004 016134  
 2005 016140 010004  
 2006 016142 032700 100000  
 2007 016146 001004  
 2008 016150 032700 174077  
 2009 016154 001023  
 2010 016156 000424  
 2011 016160 032700 000200  
 2012 016164 001011  
 2013 016166 032700 000040  
 2014 016172 001414  
 2015 016174 042704 177761  
 2016 016200 020427 000016  
 2017 016204 001007  
 2018 016206 000410  
 2019 016210 032700 000040  
 2020 016214 001405  
 2021 016216 032700 000006  
 2022 016222 001002  
 2023 016224 000241  
 2024 016226 000401  
 2025 016230 000261  
 2026 016232 000207

```

.SBTTL  CHKAMB - CHECK TSSR FOR AMBIGUITY
;*
;
;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,RO        ;IS BIT 15 SET ?
    BNE     5$            ;BRANCH IF YES
    BIT     @+C<NBA!OFL!SSR!HIADDR>,RO ;ANY OTHER BITS SET ?
    BNE     40$           ;MUST BE AN ERROR
    BR     45$           ;RETURN WITH SUCCESS
    5$:    BIT     @SSR,RO ;IS READY BIT SET ?
           BNE     10$   ;BRANCH IF READY BIT IS SET.
           BIT     @BITS,RO ;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$:  BIT     @BITS,RO ;IS FATAL ERROR BIT SET ?
           BEQ     45$   ;ERROR IF BIT IS SET WITH SSR
           BIT     @BIT2!BIT1,RO ;IS THIS A FUNCTION REJECT
           BNE     45$   ;BR, IF TSSR IS OK
           CLC                ;AMBIGUOUS CONTENTS
           BR     50$
           SEC                ;SHOW SUCCESS - NO AMBIGUITY
    45$:  SEC
           RTS      PC        ;RETURN TO CALLER
    50$:  RTS
    
```

```

2028 .SBTTL ENAIN,DSBINT - ENABLE/DISABLE INTERRUPTS
2029 ;
2030 ; DEFAULT DISPLAY INTERRUPT HANDLERS.
2031 ; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
2032 ; OTHERWISE, SAVE DPU REGISTERS AND DISMISS.
2033 ;
2034 ;
2035 ; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
2036 ;
2037 ; IOKCKIN=BIT7 ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
2038 ; IOKSTP=BIT0 ; EXPECT "STOP" INTERRUPT.
2039 ;
2040 ; INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
2041 INTMASK: .BYTE 0
2042 ; INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
2043 INTFLAG: .BYTE 0
2044 ;
2045 ; SAVED INTERRUPT VECTOR:
2046 INTVEC: .WORD 0
2047 ; SAVE CPU PC
2048 INTCPC: .WORD 0
2049 ;
2050 ; SUBROUTINE TO ENABLE INTERRUPTS:
2051 ENAIN: MOV RO,-(SP) ;SAVE RO
2052 MOV IVEC,RO ;GET POINTER TO VECTORS
2053 MOV @INTR,(RO)+ ;SET UP INTERRUPT VECTOR
2054 MOV @PRI07,(RO)+
2055 MOV (SP)+,RO ;RESTORE RO
2056 MOV (SP),-(SP)
2057 MOV @0,2(SP) ;SET CPU TO LEVEL 0
2058 RTI
2059 ;
2060 ; SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
2061 DSBINT: MOV (SP),-(SP)
2062 MOV @PRI07,2(SP)
2063 RTI
2064 .SBTTL INTR - INTERRUPT HANDLERS
2065 ;
2066 BGNSRV INTR ;DEFINE INTERRUPT ENTRY
2067 INTR:: MOV #1,INTRECV ;SET FLAG TO SHOW INTERRUPT RECEIVED
2068 CLRB INTFLAG ;CLEAR FLAG TO SAY WE GOT INTERRUPT
2069 BITB @IOKSTP,INTMASK ;EXPECTING STOP INTERRUPT?
2070 BNE 1$ ;BR IF YES
2071 BISB @IOKSTP,INTFLAG ;NO. SET THE ERROR FLAG.
2072 ;
2073 ;SAVE REGISTERS, MSG BUFFER, ETC.
2074 1$:
2075 ENDSRV
L1002E: RTI
016336 000002

```

```

2077                    .SBTTL WAITF - WAIT FOR SUBSYSTEM READY
2078                    ;
2079                    ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
2080                    ;
2081                    ; INPUTS:
2082                    ;
2083                    ;        R5        ADDRESS OF FIRST DEVICE REGISTER
2084                    ;
2085                    ; OUTPUTS:
2086                    ;
2087                    ;        R0        CONTENTS OF LAST TSSR READ
2088                    ;        CARRY    SET - READY BIT SET
2089                    ;                    CLR - TIMEOUT WAITING FOR READY
2090                    ;
2091 016340 000401        WAITF:: BR        1$                    ;NOP WHEN SUPER FIXED
2092 016342                    BREAK        ; DO A SUPVSR BREAK FIRST.
                             TRAP        C$BRK
2093 016344 012746 011000    1$:        MOV        #11000,-(SP)        ;25-APRIL-83 REV B - 1100 MSEC TIMER
2094 016350 016500 000002    2$:        MOV        TSSR(R5),R0        ;READ THE TSSR REGISTER
2095 016354 105700                    TSTB        R0                    ;TEST FOR READY BIT SET
2096
2097 016356 100420                    BMI        3$                    ; EXIT ON STOP FLAG.
2098 016360                    DELAY        1                    ; WAIT 100 USEC
                             MOV        #1,(PC)+
                             .WORD        0
                             MOV        L$DLY,(PC)+
                             .WORD        0
                             DEC        -6(PC)
                             BNE        .-4
                             DEC        -22(PC)
                             BNE        .-20
2099 016410 005316                    DEC        (SP)                    ;REDUCE DELAY COUNT
2100 016412 001356                    BNE        2$                    ;RETRY UNTIL TIMER EXPIRES
2101 016414 000241                    CLC                    ; C = 0, CONTROLLER STILL RUNNING...
2102 016416 000401                    BR        4$                    ;...OR HUNG-UP AFTER 300 MSEC.
2103 016420 000261                    3$:        SEC                    ; C = 1, CONTROLLER IS STOPPED.
2104 016422 005326                    4$:        DEC        (SP)+                    ;RESTORE STACK WITHOUT CHANGING CARRY BIT
2105 016424 000207                    RTS        PC

```

```

2107      .SBTTL  CHKTSSR - CHECK TSSR FOR READY
2108      ;*
2109      ;THIS ROUTINE WAITS FOR READY IN THE TSSR
2110      ;AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
2111      ;
2112      ;INPUT:
2113      ;      R5      ADDRESS OF CSR REGISTERS
2114      ;
2115      ;OUTPUT:
2116      ;      R0      CONTENTS OF TSSR
2117      ;      CARRY   SET - OKAY
2118      ;              CLR - NOT READY AMBIGUOUS, OR SC SET
2119      ;
2120      CHKTSSR:
2121      JSR      PC, WAITF      ;WAIT FOR READY
2122      BCC      20$           ;BRANCH IF TIME OUT
2123      JSR      PC, CHKAMB    ;TSSR AMBIGUOUS?
2124      BCC      10$           ;BR IF YES
2125      BIT      #SC, R0      ;SPECIAL CONDITION SET?
2126      BEQ      15$           ;BR IF NO
2127      BIT      #<SCE!BIE!RMR!NXM>, R0 ;ANY ERROR BITS SET?
2128      BEQ      15$           ;BR IF NO
2129      10$:     CLC              ;SET FAILURE
2130      BR      20$           ;
2131      15$:     SEC              ;SET SUCCESS
2132      20$:     RTS      PC      ;RETURN TO CALLER
2133      .SBTTL  XNXM      - CHECK FOR NONEXISTENT MEMORY
2134      ;*
2135      ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
2136      ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
2137      ;              "C" = 0, ALL ADDRESSES OK.
2138      ;
2139      ;CALL:  MOV ADR1,R1
2140      ;      MOV ADR2,R2
2141      ;      JSR PC,NXM
2142      ;      RETURN      ;TEST "C" AND PROCEED.
2143      XNXM:  MOV      #2$,R0    ; SET BUSERR VECTOR.
2144      MOV      #PRI04,R06
2145      CLR      R3              ;FLAG.
2146      1$:    TST      (R1)     ;TEST THE ADDRESS(ES).
2147      ;      ;IF ANY TRAP, CONTINUE AT 2$.
2148      ;      ;OTHERWISE, CONTINUE HERE.
2149      CMP      R1,R2
2150      BEQ      3$             ;BR IF FINISHED (NO NEXM'S).
2151      ADD      #2,R1          ;SET NEXT ADDRESS...
2152      BR      1$             ;...AND CONTINUE.
2153      2$:    COM      R3
2154      MOV      #3$, (SP)     ;GOT ONE, SET FLAG...
2155      3$:    CLRVEC  #4
2156      MOV      #4,R0
2157      TRAP    C#CVEC
2158      TST      R3              ;...AND DISMISS INTERRUPT...
2159      BEQ      .+4           ;...AND GIVE BACK THE VECTOR.
2156      TST      R3              ;DID WE CATCH ONE ??
2157      BEQ      .+4           ;NO, "C" = 0, SKIP NEXT.
2158      SEC
2159      RTS      PC              ;YES, "C" = 1, (R1) = NEXM ADDR.

```

TSV3 - GLOBAL AREAS MACRO M1113 14-JUN-84 14:17  
 TSTLOOP - CHECK ITERATION COUNT

SEQ 0077

```

2161          .SBTTL  TSTLOOP - CHECK ITERATION COUNT
2162          ;*
2163          ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
2164          ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
2165          ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
2166          ;
2167          ; CALL: LOOPTO  ARG
2168          ;
2169          TSTLOOP::
2170          TST      NOITS          ; ITERATIONS INHIBITED?
2171          BNE      1$             ; YES.
2172          TST      QVP            ; NO.
2173          BMI      1$             ;LOOPS DISALLOWED IN QUICK PASS.
2174          DEC      LOOPCNT        ; BUMP LOOP COUNTER.
2175          BNE      2$
2176          1$:      CLC             ;LOOP DISALLOWED, OR DONE.
2177          BR       3$
2178          2$:      SEC             ;LOOP ENABLED.
2179          3$:      RTS      PC
2180
2181          .SBTTL  TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS
2182          ;*
2183          ; PRINT THE NUMBER AND NAME OF EACH TEST AS WE GO ALONG.
2184          ; INCREMENT "TESTK" TO INDICATE THE NUMBER OF TESTS
2185          ; IN THE CURRENT RUN SEQUENCE.
2186          ; CLEAR THE ERROR COUNTER AND SIGNATURE EXTENSION FLAGS.
2187          ;
2188          ;INPUT:
2189          ;
2190          ;      R0      POINTER TO TEST ID ASCIZ STRING
2191          ;
2192          ;OUTPUT:
2193          ;
2194          ;      R5      ADDRESS OF FIRST DEVICE REGISTER
2195          ;
2196          ;IMPLICIT OUTPUTS:
2197          ;
2198          ;      TSTCNT  UPDATED TO COUNT TESTS PERFORMED SINCE START OR RESTART
2199          ;
2200          ;SIDE EFFECTS:
2201          ;
2202          ;      INTERRUPT LEVEL IS RASIED TO LEVEL OF
2203          ;      THE DEVICE UNDER TEST
2204          ;
2205          ;-
2206
2207          TSTSETUP::
2208          MOV      R0, -(SP)      ;SAVE THE TEST ID MESSAGE
2209          CLR      SIFLAG        ; CLEAR "SOFT INIT" FLAG
2210          CLR      ERRK          ; CLEAR LOCAL ERROR COUNTER.
2211          CLR      EXTA         ; CLEAR ERROR EXTENSION FLAG.
2212          CLR      INTMASK      ; CLEAR INTERRUPT MASK (CHECK ERROR)
2213          MOV      UNITN, R0     ; GET THE UNIT NUMBER.
2214          ASL      R0           ; ... AND MAKE IT A WORD OFFSET.
2215          TST      NODEV        ; DID STARTUP FIND THE DEVICE?
2216          BEQ      4$           ; BR IF YES
2217          BPL      3$           ; BR IF NOT IDLE

```

```

2218 016640 052760 160000 003166      BIS      @160000,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
2219 016646      ERRDF    1,NXR,NXRERR ; NO DEVICE HERE -- PRINT IT
      016646 104455      TRAP    C#ERDF
      016650 000001      .WORD   1
      016652 003736      .WORD   NXR
      016654 005734      .WORD   NXRERR
2220 016656 000407      BR      2$
2221 016660 052760 160001 003166 3$:  BIS      @160001,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
2222 016666      ERRDF    2,NOINIT ; DEVICE NOT IDLE
      016666 104455      TRAP    C#ERDF
      016670 000002      .WORD   2
      016672 004333      .WORD   NOINIT
      016674 000000      .WORD   0
2223 016676 012737 177777 003102 2$:  MOV      @-1,DUFLG ; DROP THE UNIT
2224 016704      DODU     UNITN
      016704 013700 002172      MOV     UNITN,R0
      016710 104451      TRAP    C#DODU
2225 016712      DOCLN   ; ABORT THE PASS
      016712 104444      TRAP    C#DCLN
2226 016714 000423      BR      5$
2227
2228 016716      4$:  RFLAGS   R0 ; GET THE OPERATOR FLAGS.
      016716 104421      TRAP    C#RFLA
2229 016720 032700 001000      BIT     @PNT,R0 ; PRINT THE TEST NUMBERS?
2230 016724 001412      BEQ     1$ ; BR IF NO
2231 016726 011600      MOV     (SP),R0 ;GET THE ID MESSAGE
2232 016730      PRINTF  @TNAM,R0 ;DISPLAY THE TEST ID
      016730 010046      MOV     R0,-(SP)
      016732 012746 016774      MOV     @TNAM,-(SP)
      016736 012746 000002      MOV     @2,-(SP)
      016742 010600      MOV     SP,R0
      016744 104417      TRAP    C#PNTF
      016746 062706 000006      ADD     @6,SP
2233 016752 005237 002204      1$:  INC     TSTCNT ; BUMP TEST COUNTER.
2234 016756      SETPRI  IPRI ;PRIORITY THAT OF DEVICE
      016756 013700 002202      MOV     IPRI,R0
      016762 104441      TRAP    C#SPRI
2235 016764 005726      5$:  TST     (SP)+ ;FIX UP THE STACK
2236 016766 013705 002176      MOV     CSRADDR,R5 ; ADDRESS OF TSV REGISTERS ON UNIBUS
2237 016772 000207      RTS     PC
2238 016774 045 123 045 TNAM: .ASCIZ 'S#T#A Test'
2239      .EVEN
2240      .SBTTL TSTEND - PRINT ERRORS RECEIVED
2241
2242 ; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
2243 ; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
2244
2245 TSTEND: RFLAGS   R0
      017010 104421      TRAP    C#RFLA
2246 017012 030027 020000      BIT     R0,#IER
2247 017016 001412      BEQ     1$ ; BR IF "IER" NOT SET.
2248 017020      PRINTF  @ESUM,ERRK ; PRINT ERROR COUNT.
      017020 013746 017046      MOV     ERRK,-(SP)
      017024 012746 017050      MOV     @ESUM,-(SP)
      017030 012746 000002      MOV     @2,-(SP)
      017034 010600      MOV     SP,R0
      017036 104417      TRAP    C#PNTF
    
```



TSV3 - GLOBAL AREAS      MACRO M1113 14-JUN-84 14:17  
TSTEND - PRINT ERRORS RECEIVED

SEQ 0079

2249	017040	062706	000006			ADD	06.SP	
2250	017044	000207			18:	RTS	PC	
2251	017046	000000			ERRK:	0		; LOCAL ERROR COUNT.
2252	017050	045	101	040	ESUM:	.ASCIZ	/#A #D#A ERRORS/	
2253	017067	105	122	122	EMAXDU:	.ASCIZ	/ERROR LIMIT REACHED -- DROPPING UNIT/	
2254						.EVEN		

```

2256                                     .SBTTL  INCERK  - INCREMENT LOCAL ERROR COUNT
2257                                     ;*
2258                                     ; ROUTINES TO INCREMENT LOCAL ERROR COUNT AND CHECK FOR LIMIT:
2259                                     ;
2260 017134 005237 017046 INCERK: INC  ERRK          ; INCREMENT LOCAL ERROR COUNT
2261 017140 010046      MOV  RO,-(SP)      ; SAVE RO
2262 017142 013700      MOV  UNITN,RO      ; GET UNIT NUMBER,
2263 017146 006300      ASL  RO              ; ... AND MAKE IT A WORD OFFSET,
2264 017150 062700      ADD  @ERTABL,RO    ; RO GETS ADDRESS OF ERROR TABLE ENTRY.
2265 017154 005210      INC  (RO)          ; INCREMENT THE DEVICE ERROR COUNT
2266 017156 032710      BIT  @7777,(RO)    ; DID WE OVERFLOW THE FIELD?
2267 017162 001001      BNE  1$          ; BR IF NO.
2268 017164 005310      DEC  (RO)          ; YES -- BACK IT UP TO 7777.
2269 017166 012600      1$: MOV  (SP)+,RO    ; RESTORE RO
2270 017170 000207      RTS  PC            ; RETURN TO CALLER.
2271
2272 017172 010046      CKEMAX: MOV  RO,-(SP)  ; SAVE RO
2273 017174 013700      MOV  UNITN,RO      ; GET UNIT NUMBER
2274 017200 006300      ASL  RO              ; ... AND MAKE IT A WORD OFFSET
2275 017202 016000      MOV  ERTABL(RO),RO  ; GET ERROR TABLE ENTRY
2276 017206 042700      BIC  @170000,RO    ; EXTRACT ERROR COUNT FIELD
2277 017212 020037      CMP  RO,GERRMAX   ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
2278 017216 103004      BHIS 1$          ; BR IF YES
2279 017220 023737      CMP  ERRK,LERRMAX  ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
2280 017226 103417      BLO  2$          ; BR IF NO
2281 017230      1$: RFLAGS RO          ; GET OPERATOR FLAGS
2282 017230 104421      TRAP C#RFLA
2283 017232 032700      BIT  @IDU,RO      ; IS DROPPING INHIBITED?
2284 017236 001013      BNE  2$          ; BR IF YES.
2285 017240 012737      MOV  @-1,DUFLG   ; NO -- DROP THE UNIT
2286 017246      ERRDF 4,EMAXDU
2287 017246 104455      TRAP C#ERDF
2288 017250 000004      .WORD 4
2289 017252 017067      .WORD EMAXDU
2290 017254 000000      .WORD 0
2291 017256      DODU  UNITN
2292 017256 013700      MOV  UNITN,RO
2293 017262 104451      TRAP C#DODU
2294 017264      DOCLN
2295 017264 104444      TRAP C#DCLN
2296 017266 012600      2$: MOV  (SP)+,RO    ; RESTORE RO
2297 017270 000207      RTS  PC            ; RETURN TO CALLER

```

```

2291          .SBTTL CKDROP - CHECK IF UNIT SHOULD BE DROPPED
2292          ;*
2293          ; CHECK IF UNIT SHOULD BE DROPPED
2294          ;
2295 017272 010046 CKDROP: MOV     RO, -(SP)
2296 017274          FORCERROR      1$,NOTSSR
2297 017304          RFLAGS     RO
2298 017306 104421   TRAP     C#RFLA
2299 017312 032700 000040 BIT     #IDU,RO
2300 017314 001010   BNE     1$
2301 017316 011600   MOV     (SP),RO
2302 017324 012737 177777 003102 MOV     #-1,DUFLG
2303 017324 013700 002172   DODU    UNITN
2304 017332 104451   MOV     UNITN,RO
2305 017334 012600   TRAP     C#DODU
2306 017336 000207   DOCLN          ;ABORT THE PASS
2307          TRAP     C#DCLN
2308          ;
2309          .SBTTL CONFIG - DETERMINE CONFIGURATION OF SYSTEM
2310          ;
2311          ; SUBROUTINE - DETERMINE CONFIGURATION OF TSV05 SYSTEM.
2312          ;
2313          ; CONFIG:
2314          JSR     PC,SOFINIT
2315          RTS     PC
2316          .SBTTL KTON,KTOFF - ENABLE/DISABLE MEMORY MANAGEMENT
2317          ;
2318          ; SUBROUTINE - ENABLE MEM MGT.
2319          ;
2320          KTON:  TST     KTF LG          ; GOT KT?
2321          BEQ     1$                  ; NO.
2322          MOV     #1,SRO              ; YES. ENABLE KT11.
2323          1$:   RTS     PC
2324          ;
2325          ; SUBROUTINE - DISABLE MEM MGT.
2326          ;
2327          KTOFF: TST     KTF LG          ; GOT KT11?
2328          BEQ     1$                  ; NO.
2329          NOP
2330          NOP
2331          MOV     #0,SRO              ; DISABLE KT.
2332          1$:   RTS     PC

```

```

2334                               .SBTTL SETMAP - SETUP PAR6 MAPPING
2335
2336                               ;*
2337                               ;
2338                               ; THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
2339                               ; AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
2340                               ; IS RETURNED BIASED TO PAR6.
2341                               ;
2342                               ; INPUTS:
2343                               ;
2344                               ;       R0       HIGH ORDER ADDRESS BITS
2345                               ;       R1       LOW ORDER ADDRESS BITS
2346                               ;
2347                               ; OUTPUTS:
2348                               ;
2349                               ;       R0       OFFSET INTO BLOCK WITH PAR6 BIAS (I.E. THE ADDRESS)
2350                               ;       CARRY    SET IF SUCCESS
2351                               ;                CLR IF ERROR
2352                               ;
2353                               ;--
2353 017406                       SETMAP:       SAVREG                               ;SAVE R1-R4 UNTIL NEXT RETURN
2354 017406                       TST       KTFLG                               ;SYSTEM HAVE ABOVE 28K?
2355 017412   005737   003122       BEQ       10$                               ;BR IF NO
2356 017416   001433                       MOV       R1,R2                       ;SAVE LOW ORDER BITS
2357 017420   C10102                        .REPT       6
2358                               ASR       R0                               ;CONVERT WORD ADDRESS TO 32W BLOCKS
2359                               ROR       R1                               ;MAKE IT DOUBLE PRECISION
2360                                .ENDR
2361                               BIC       @177,R1                            ;ALINE FOR LOWER 4K BOUNDARY
2362 017452   042701   000177       CMP       R1,KTFLG                       ;HIGHER THAN EXISTING MEMORY?
2363 017456   020137   003122       BHIS     10$                               ;BR IF YES
2364 017462   103011                       MOV       R1,@KIPAR6                    ;SETUP MAPPING REGISTER PAR6
2365 017464   010137   172354       BIC       @160000,R2                       ;SETUP DISPLACEMENT IN PAGE
2366 017470   042702   160000       ADD       @140000,R2                       ;ADD IN PAR6 BIAS
2367 017474   062702   140000       MOV       R2,R0                            ;RETURN IN R0
2368 017500   010200                       SEC                               ;SET SUCCESS
2369 017502   000261                       BR       15$                            ;
2370 017504   000401                       10$: CLC                               ;SET FAILURE
2371 017506   000241                       15$: RTS       PC                               ;RETURN
2372 017510   000207                       .SBTTL FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
2373
2374                               ;*
2375                               ; FILL MEMORY WITH A BACKGROUND PATTERN
2376                               ;
2377                               ; INPUTS:
2378                               ;
2379                               ;       R0 = BACKGROUND PATTERN
2380                               ;       FREE    = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
2381                               ;       KTFLG   = SET TO HIGHEST MEMORY LOCATION IF > 28K.
2382                               ;
2383                               ; OUTPUTS:
2384                               ;
2385                               ;       NONE
2386                               ;
2387                               ;--
2388 017512                       FILLMEM:     SAVREG                            ;SAVE R1-R5 UNTIL NEXT RETURN
2389 017512                       JSR       PC,KTOFF                           ;DISABLE KT.
2390 017516   004737   017364

```

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

SEQ 0083

```

2391 017522 010003          MOV      R0,R3          ;COPY TEST PATTERN
2392 017524 013701 003114  MOV      FREE,R1       ;GET FIRST FREE LOCATION
2393 017530 013702 003116  MOV      FRESIZ,R2     ;SIZE OF FREE SPACE BELOW 28K.
2394 017534 010321          10$:    MOV      R3,(R1)+      ;STORE A BACKGROUND WORD
2395 017536 005302          DEC      R2           ;DONE ALL MEMORY IN FREE SPACE?
2396 017540 003375          BGT     10$          ;BR IF NO
2397 017542 005737 003122  TST     KTFLG         ; GOT KT?
2398 017546 001477          BEQ     55$          ; NO. GET OUT.
2399 017550 004737 017346  JSR     PC,KTON       ; YES. ENABLE KT.
2400 017554 005000          CLR     R0           ;HIGH ORDER ADDRESS START
2401 017556 013701 003142  MOV     PST32W,R1     ;GET >28K START ADDRESS (IN 32W BLOCKS)
2402          000006          .REPT   6
2403          CLC           ;CLEAR C BIT
2404          ROL     R1     ;CONVERT BLOCKS TO WORDS
2405          ROL     R0     ;MAKE IT DOUBLE PRECISION
2406          .ENDR
2407 017626 004737 017406  JSR     PC,SETMAP     ;SETUP PAR6 MAPPING REGISTER
2408 017632 010320          30$:    MOV     R3,(R0)+      ;STORE TEST PATTERN IN >28K ADDRESS
2409 017634 020027 160000  CMP     R0,#160000    ;END OF PAR6 MAPPING AREA?
2410 017640 103774          BLO     30$          ;BR IF NO
2411 017642 162700 020000  SUB     #20000,R0     ;BACKUP INTO PAR6 MAPPING BEGIN
2412 017646 062737 000200 172354  ADD     #200,#KIPAR6  ;POINT TO NEXT 4K BLOCK >28K.
2413 017654 023737 172354 003122  CMP     #KIPAR6,KTFLG ;END OF MEMORY?
2414 017662 001427          BEQ     50$          ;BR IF YES
2415 017664 005737 003134  TST     T23A         ;11/23A?
2416 017670 001407          BEQ     35$          ;NO KEEP GOING
2417 017672 013704 177572  MOV     SR0,R4        ;GET SR0 CONTENTS
2418 017676 042704 177761  BIC     #177761,R4    ;CLEAR ALL BUT PAGE NUMBER
2419 017702 022704 000016  CMP     #16,R4       ;SEE IF PAGE 7
2420 017706 001415          BEQ     50$          ;EXIT IF THERE
2421 017710 005737 003136          35$:    TST     T23B         ;11/23B?
2422 017714 001410          BEQ     45$          ;NO KEEP GOING
2423 017716 023727 172354 007600  CMP     #KIPAR6,#7600 ;REACHED 18 BITS?
2424 017724 103001          BHIS   40$          ;YES
2425 017726 000403          BR     45$          ;NO KEEP GOING
2426 017730 012737 000020 172516 40$:    MOV     #20,SR3      ;SET 22 BIT RELOCATION
2427 017736 000137 017632          45$:    JMP     30$          ;KEEP GOING ON ETC.
2428 017742 004737 017364          50$:    JSR     PC,KTOFF     ; DISABLE KT.
2429 017746 000207          55$:    RTS     PC

```

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

SEQ 0084

```

2431 .SBTTL CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN
2432 ;*
2433 ; COMPARE MEMORY WITH A BACKGROUND PATTERN
2434 ;
2435 ; INPUTS:
2436 ;
2437 ; RO = BACKGROUND PATTERN
2438 ; FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
2439 ; KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
2440 ;
2441 ; OUTPUTS:
2442 ;
2443 ; CARRY - SET IF NO ERROR
2444 ; CARRY - CLR IF ERROR
2445 ;
2446 ; IMPLICIT OUTPUTS:
2447 ;
2448 ; ERRHI - ERROR HIGH ADDRESS
2449 ; ERRLO - ERROR LOW ADDRESS
2450 ; EXPD - EXPECTED DATA
2451 ; RECV - RECEIVED DATA
2452 ;-
2453 CMPMEM:
2454 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2455 MOV RO,R3 ;COPY TEST PATTERN
2456 JSR PC,KTOFF ;DISABLE KT.
2457 MOV FREE,R1 ;GET FIRST FREE LOCATION
2458 MOV FRESIZ,R2 ;SIZE OF FREE SPACE BELOW 28K.
2459 10$: CMP R3,(R1) ;FREE SPACE LOCATION EQUAL TO EXPD?
2460 BEQ 15$ ;BR IF YES
2461 MOV R1,ERRLO ;SAVE ADDRESS IN ERROR
2462 CLR ERRHI ;NO HIGH ADDRESS
2463 MOV R3,EXPD ;SAVE EXPD FOR ERROR REPORT
2464 MOV (R1),RECV ;SAVE RECV FOR ERROR REPORT
2465 BR 50$ ;
2466 15$: TST (R1)+ ;POINT TO NEXT ADDRESS
2467 DEC R2 ;DONE ALL MEMORY IN FREE SPACE?
2468 BGT 10$ ;BR IF NO
2469 TST KTFLG ; GOT KT?
2470 BEQ 55$ ; NO. GET OUT.
2471 JSR PC,KTON ; YES. ENABLE KT.
2472 CLR RO ;HIGH ORDER ADDRESS START
2473 MOV PST32W,R1 ;GET >28K START ADDRESS (IN 32W BLOCKS)
2474 .REPT 6
2475 ROL R1 ;CONVERT BLOCKS TO WORDS
2476 ROL RO ;MAKE IT DOUBLE PRECISION
2477 .ENDR
2478 BIC #177,R1 ;ALINE 4K BOUNDARY
2479 MOV RO,-(SP) ;SAVE HIGH ORDER
2480 MOV R1,-(SP) ;SAVE LOW ORDER
2481 JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
2482 MOV RO,R4 ;COPY ADDRESS BIASED TO PAR6
2483 MOV (SP)+,R1 ;RESTORE LOW ORDER IN NON PAR6 FORMAT
2484 MOV (SP)+,RO ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
2485 30$: CMP R3,(R4) ;ABOVE 28K LOCATION EQUAL EXPD?
2486 BEQ 32$ ;BR IF YES
2487 MOV RO,ERRHI ;SAVE HIGH ORDER IN ERROR

```

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

SEQ 0085

```

2488 020130 010137 002230      MOV      R1,ERRLO      ;SAVE LOW ORDER IN ERROR
2489 020134 010337 002222      MOV      R3,EXPD      ;SAVE EXPD FOR ERROR REPORT
2490 020140 011437 002224      MOV      (R4),RECV    ;SAVE RECV FOR ERROR REPORT
2491 020144 000421              BR       50$          ;
2492 020146 062701 000002      32$:    ADD      @2,R1    ;UPDATE NON PAR6 ADDRESS
2493 020152 005500              ADC      R0          ;MAKE IT DOUBLE PRECISION ADD
2494 020154 062704 000002      ADD      @2,R4        ;UPDATE PAR FORMAT ADDRESS
2495 020160 020427 160000      CMP      R4,@160000  ;END OF PAR6 MAPPING AREA?
2496 020164 103755              BLO     30$          ;BR IF NO
2497 020166 162704 020000      SUB      @20000,R4    ;BACKUP INTO PAR6 MAPPING BEGIN
2498 020172 062737 000200      ADD      @200,@#KIPAR6 ;POINT TO NEXT 4K BLOCK >28K.
2499 020200 023737 172354      CMP      @#KIPAR6,KTFLG ;END OF MEMORY?
2500 020206 101744              BLOS   30$          ;BR IF NO
2501 020210 004737 017364      50$:    JSR      PC,KTOFF   ;TURN OFF MEMORY MAPPING
2502 020214 000241              CLC                    ;SET FAILURE
2503 020216 000403              BR       60$          ;
2504 020220 004737 017364      55$:    JSR      PC,KTOFF   ;TURN OFF MEMORY MAPPING
2505 020224 000261              SEC                    ;SET SUCCESS
2506 020226 000207      60$:    RTS       PC
2507              .SBTTL  REGSAV - SAVE R1-R5 ON STACK
2508              ;*
2509              ;
2510              ;ROUTINE TO
2511              ;SAVE R1 THROUGH R5 ON THE STACK
2512              ;
2513              ;CALLING SEQUENCE:
2514              ;
2515              ;      JSR      R5,REGSAV
2516              ;
2517              ;THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
2518              ;THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
2519              ;THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTORE
2520              ;REGISTERS.
2521              ;
2522              ;THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
2523              ;CALLED VIA A JSR PC INSTRUCTION
2524              ;
2525              ;-
2526
2527 020230      REGSAV:
2528 020230 010446      MOV      R4,-(SP)
2529 020232 010346      MOV      R3,-(SP)
2530 020234 010246      MOV      R2,-(SP)
2531 020236 010146      MOV      R1,-(SP)
2532 020240 010546      MOV      R5,-(SP)
2533 020242 016605 000012      MOV      10.(SP),R5
2534 020246 004736      JSR      PC,@(SP)+
2535 020250 012601      MOV      (SP)+,R1
2536 020252 012602      MOV      (SP)+,R2
2537 020254 012603      MOV      (SP)+,R3
2538 020256 012604      MOV      (SP)+,R4
2539 020260 012605      MOV      (SP)+,R5
2540 020262 000207      RTS       PC

```

TSV3 - GLOBAL AREAS      MACRO M1113    14-JUN-84 14:17  
 GETPAT - GET 8 BIT PATTERN FROM OPERATOR

SEQ 0086

```

2542                                .SBTTL GETPAT - GET 8 BIT PATTERN FROM OPERATOR
2543                                ;*
2544                                ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
2545                                ;
2546                                ;INPUTS:                NONE.
2547                                ;
2548                                ;OUTPUTS:
2549                                ;        RO                OCTAL NUMBER FROM THE OPERATOR
2550                                ;
2551                                ;CALLING SEQUENCE:
2552                                ;        JSR        PC,GETPAT
2553                                ;-
2554 020264                        GETPAT::
2555 020264                               SAVREG                        ;SAVE THE GENERAL REGISTERS
2556 020270                        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$HILIM
2557 020310                        10000$:        BNCOMPLETE        1$        ;RETRY IF ERROR
                                      BCC        1$
2558 020312                               MOV        PATDAT,RO        ;DATA PATTERN FROM OPERATOR
2559 020316                               RTS        PC                ;RETURN TO CALLER
2560
2561                                ;*
2562                                ;LOCAL DATA AREA
2563                                ;-
2564
2565 020320                               PATDAT: .WORD    0                ;TEMPORARY STORAGE FOR DATA
2566 020322                               DATASC: .ASCIZ  'ENTER DATA PATTERN'
2567                                       .EVEN

```



```

2569      .SBTTL  GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE
2570      ;*
2571      ;ROUTINE TO ISSUE A MENU AND GET THE OPERATOR'S RESPONSE.
2572      ;
2573      ;INPUTS:
2574      ;      RO      ADDRESS OF ASCIZ STRING OF MENU
2575      ;      R1      MAXIMUM ALLOWABLE OPERATOR RESPONSE
2576      ;
2577      ;OUTPUTS:
2578      ;      RO      NUMBER OF THE OPERATOR'S SELECTION
2579      ;-
2580      GETSEL::
2581      SAVREG                      ;SAVE GENERAL REGISTERS
2582      MOV      R0,R2              ;SAVE THE MENU ADDRESS
2583      1$:  MOV      R2,R3              ;START OF MENU STRING
2584      2$:  TST      (R3)              ;END OF ASCII ?
2585      BEQ      3$                  ;BRANCH IF ALL LINES DISPLAYED
2586      PRINTF  @SELASC,(R3)+        ;DISPLAY THE MENU
2587      MOV      (R3)+,-(SP)
2588      MOV      @SELASC,-(SP)
2589      MOV      @2,-(SP)
2590      MOV      SP,R0
2591      TRAP    C#PNTF
2592      ADD     @6,SP
2593      BR      2$
2594      3$:  GMANID  MENASC,MENRES,D,-1,0,-1,NO
2595      TRAP    C#GMAN
2596      BR      10001$
2597      .WORD  MENRES
2598      .WORD  T#CODE
2599      .WORD  MENASC
2600      .WORD  -1
2601      .WORD  T#LOLIM
2602      .WORD  T#HILIM
2603      10001$: BNCOMPLETE  1$          ;RETRY IF ERROR
2604      BCC     1$
2605      MOV     MENRES,R0              ;GET THE OPERATOR'S REPLY
2606      CMP     R0,R1                  ;COMPARE TO MAXIMUM ALLOWED
2607      BLOS   5$                      ;BRANCH IF OK
2608      PRINTF  @MENERR                ;DISPLAY ERROR MESSAGE
2609      MOV     @MENERR,-(SP)
2610      MOV     @1,-(SP)
2611      MOV     SP,R0
2612      TRAP    C#PNTF
2613      ADD     @4,SP
2614      BR      1$                      ;RETRY
2615      5$:  RTS     PC                  ;RETURN TO CALLER
2616      045  MENERR: .ASCIZ  '%N%#A *** Menu Selection Too Large ***'
2617      045  SELASC: .ASCIZ  '%N#T'
2618      164  MENASC: .ASCIZ  'Enter Menu Selection: '
2619      .EVEN
2620      MENRES: .WORD  0

```

```

2602                    .SBTTL  CHKMAN  - CHECK MANUAL INTERVENTION LEGALITY
2603                    ;*
2604                    ;
2605                    ;ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
2606                    ;
2607                    ;INPUT:
2608                    ;
2609                    ;      NONE.
2610                    ;
2611                    ;OUTPUT:
2612                    ;
2613                    ;      CARRY   0      MANUAL INTERVENTION NOT ALLOWED
2614                    ;             1      MANUAL INTERVENTION IS OK
2615                    ;
2616                    ;SIDE EFFECTS:
2617                    ;
2618                    ;      A MESSAGE IS DISPLAYED WARNING THAT TEST IS
2619                    ;      NOT EXECUTED IF MANUAL INTERVENTION IS NOT
2620                    ;      ALLOWED.
2621                    ;
2622                    ;-
2623
2624 020570               CHKMAN::
2625 020570                SAVREG                   ;SAVE THE REGISTERS
2626 020574                MANUAL                   ;SEE IF MANUAL INTERVENTION OK
2627 020574   104450        TRAP    C$MANI
2628 020576                BCOMplete 1$           ;BRANCH IF ALLOWED
2629 020576   103411        BCS       1$
2630 020600                PRINTF #NOMAN           ;PRINT THE WARNING MESSAGE
2631 020600   012746   020624   MOV       #NOMAN, -(SP)
2632 020604   012746   000001   MOV       #1, -(SP)
2633 020610   010600           MOV       SP, R0
2634 020612   104417           TRAP    C$PRINTF
2635 020614   062706   000004   ADD       #4, SP
2636 020620   000241           CLC                   ;CLEAR CARRY FOR ERROR
2637 020622   000207           1$:   RTS       PC                   ;RETURN
2638 020624   045       116    045  NOMAN:   .ASCIZ  'N/A *** Manual Intervention not Allowed - Test Aborted ***'
2639                    .even

```

```

2635          .SBTTL  ENVIRN  - SETUP FREE DIAGNOSTIC SPACE
2636          ;
2637          ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
2638          ;
2639 020720     ENVIRN: MEMORY  R0
                020720     104431     TRAP      C$MEM
2640 020722     010037     003114     MOV       RO,FREE      ; GET 1ST FREE ADDRESS...
2641 020726     062737     000002     003114     ADD       #2,FREE
2642 020734     011037     003116     MOV       (RO),FRESIZ ; ...AND WORD COUNT.
2643 020740     162737     000004     003116     SUB       #4,FRESIZ
2644 020746     013702     002012     MOV       L$UNIT,R2  ; GET NUMBER OF UNITS
2645 020752     162737     000007     003116     10$:     SUB       #7,FRESIZ ; TAKE AWAY 7 WORDS PER UNIT
2646 020760     005302     DEC       R2
2647 020762     001373     BNE      10$
2648 020764     013700     003114     MOV       FREE,R0    ;GET FIRST FREE ADDRESS
2649 020770     063700     003116     ADD       FRESIZ,R0  ;POINT TO LAST FREE ADDRESS
2650 020774     162700     000002     SUB       #2,R0      ;BACKUP 1 WORD
2651 021000     010037     003120     MOV       RO,FREEHI  ;STORF LAST FREE ADDRESS
2652 021004     000240     NOP
2653 021006     012701     177520     MOV       #BDVPCR,R1 ;GET BDV11 PCR ADDRESS
2654 021012     010102     MOV       R1,R2      ;COPY TO R2
2655 021014     062702     000002     ADD       #2,R2      ;SET THE RANGE
2656 021020     004737     016466     JSR      PC, XNXM    ;SEE IF WE HAVE ONE
2657 021024     103001     BCC      15$         ;OK TO SET FLAGS
2658 021026     000423     BR       40$         ;RETURN WITH FLAGS CLEAR
2659 021030     013701     177520     15$:     MOV       BDVPCR,R1  ;SAVE PCR CONTENTS
2660 021034     062701     000001     ADD       #1,R1      ;ADD ONE TO IT
2661 021040     012702     177520     MOV       #BDVPCR,R2 ;GET BDV11 PCR ADDRESS
2662 021044     005212     INC      (R2)        ;TRY TO WRITE TO IT
2663 021046     013703     177520     MOV       BDVPCR,R3  ;GET RESULTS
2664 021052     020103     CMP      R1,R3       ;DID IT CHANGE?
2665 021054     001006     BNE      20$         ;NO, MUST BE 11/23B
2666 021056     005237     003134     INC      T23A       ;SET THE FLAG
2667 021062     042737     170000     002120   BIC      #170000,L$HIME ;SUPERVISOR COULD BE WRONG
2668          ;
2669          ;
2670 021070     000402     BR       40$         ;BR 40$ FOR RELEASE
2671 021072     005237     003136     20$:     PRINTF   #M8106     ;TELL THE SYSTEM TYPE
                NOP
2672          ;
2673          ;
2674 021076     000207     40$:     PRINTF   #M8189     ;TELL THE SYSTEM TYPE
                ;
2675 021076     000207     RTS      PC          ;RETURN

```

```

2677                               .SBTTL  KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS
2678                               ;*
2679                               ;
2680                               ;ROUTINE TO INIT KT-11
2681                               ;
2682                               ;-
2683
2684 021100          KTINIT:
2685 021100 005037 003122          CLR      KTFLG          ; INIT >28K MEMORY FLAG
2686 021104 005037 003124          CLR      KTENABLE       ; INIT TEST >28K FLAG
2687 021110 023727 002120 001577  CMP      L$HIME,#1577   ; GOT ENOUGH MEMORY (>28K)?
2688 021116 101444          BLOS     9$                ; NO.
2689 021120 013700 000004          MOV      @#ERRVEC,R0    ; SAVE OLD ERR VEC PTR.
2690 021124 012737 021216 000004  MOV      @2$,@#ERRVEC  ; SET ERR VEC PTR.
2691 021132 005737 177572          TST      @#SRO         ; GOT KT11?
2692 021136 000240          NOP                       ; (TRAP IF NO).
2693 021140 013737 002120 003122  MOV      L$HIME,KTFLG ; YES. SET KT FLAG.
2694 021146 042737 000177 003122  BIC      @177,KTFLG   ;
2695 021154 010037 000004          MOV      R0,@#ERRVEC ; RESTORE OLD ERR VEC PTR.
2696 021160 005000          CLR      R0          ; R0 = AR DATA.
2697 021162 012701 172340          MOV      @KIPAR,R1   ; R1 = KI REGS PTR.
2698 021166 012761 077406 177740 1$: MOV      @77406,-40(R1) ; SET DESCRIPTOR REG.
2699 021174 010021          MOV      R0,(R1)+   ; SET KIPAR REG.
2700 021176 062700 000200          ADD      @200,R0    ; BUMP AR DATA BY "4K".
2701 021202 020027 002000          CMP      R0,#2000   ; AT "I/O"?
2702 021206 001367          BNE     1$          ; NO.
2703 021210 012741 177600          MOV      @177600,-(R1) ; YES. SET KTPAR7 FOR I/O.
2704 021214 000405          BR      9$
2705
2706 021216 012716 021224          2$: MOV      @6$, (SP)  ; SET UP RETURN
2707 021222 000002          RTI                       ; RTI TO NEXT LOCATION
2708
2709 021224 010037 000004          6$: MOV      R0,@#ERRVEC ; RESTORE OLD ERR VEC PTR.
2710
2711 021230 000207          9$: RTS      PC

```

```

2713      ;*
2714      ;       SUBROUTINE TO SET EXTENDED FEATURES SWITCH
2715      ;
2716      ;       Requires that SOFINIT and WRTCHR have been done previous to call.
2717      ;
2718      ;
2719      ;INPUTS:
2720      ;       R5       CURRENT UNIT NUMBER
2721      ;OUTPUTS:
2722      ;       The Extended Features Switch is set.
2723      ;
2724      ;-
2725
2726 021232 INVERT::
2727
2728 021232 005737 002216          TST     EXTFEA          ; IS SWITCH SET?
2729 021236 001020                BNE     1$              ; YES,EXIT STAGE RIGHT!(or the next one outa town!)
2730 021240 012737 100206 021304  MOV     @100206,CMDPKT  ; WRT SUB-SYS MEM CMD
2731 021246 012737 021314 021306  MOV     @WSMBK,CMDPKT+2 ; MSG BUF ADDR
2732 021254 012737 000006 021312  MOV     @6,CMDPKT+6    ; BYTE COUNT
2733 021262 012737 100010 021314  MOV     @100010,WSMBK  ; INVERT THE SWITCH
2734 021270 012704 021304          MOV     @CMDPKT,R4     ; SET CMDPKT INTO R4
2735 021274 004737 010752          JSR     PC,WRTCHR     ; DO IT
2736 021300 000207          1$: RTS     PC              ; RETURN
2737
2738      ;       COMMAND PACKET.
2739
2740          021304          =       <..3>E177774    ;MUST BE ON MOD 4 BOUNDRY.
2741
2742 021304 000000          CMDPKT:: 0          ;1ST WORD IS TS05 COMMAND.
2743 021306 000000          0          ;2ND WORD IS THE BUFFER LOW ADDRESS.
2744 021310 000000          0          ;3RD WORD IS THE BUFFER HIGH ADDRESS.
2745 021312 000000          0          ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.
2746
2747      ;       WRITE SUB-SYSTEM MEMORY CHARACTERISTIC BLOCK.
2748
2749 021314 000000          WSMBK:: 0          ;1ST WORD:: SEL 0
2750 021316 000000          0          ;2ND WORD:: SEL 2
2751 021320 000000          0          ;3RD WORD:: SEL 4
2752          .EVEN
2753
2754      ;*
2755      ;       SUBROUTINE TO CHECK WETHER OR NOT WE'LL TEST NXM
2756      ;
2757      ;INPUTS:
2758      ;OUTPUTS:
2759      ;       The NXMFLG is set if we can test.
2760      ;       The NXML0 and NXMHI addresses are setup.
2761      ;-
2762
2763 021322 MEMCK::
2764
2765 021322          SAVREG          ;SAVE THE REGISTERS
2766 021326 005037 003126          CLR     NXMFLG        ;CLEAR THE FLAG
2767 021332 005037 003130          CLR     NXML0        ;CLEAR THE TEST ADDRESS LO
2768 021336 005037 003132          CLR     NXMHI        ;CLEAR THE TEST ADDRESS HI
2769 021342 005737 003136          TST     T23B         ;IS IT A 11/238?

```

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

SEQ 0092

```

2770 021346 001407          BEQ      1#          ;NO
2771 021350 023727 002120 007777  CMP      L#HIME,#07777 ; GREATER THAN 128K
2772 021356 103406          BLO      2#          ; NO
2773 021360 004737 021476          JSR      PC,NXMTST   ;SETUP THE ADDRESS
2774 021364 000427          BR       13#         ;SET THE FLAG AND EXIT
2775 021366 005737 003134          TST      T23A       ;IS IT A 11/23A?
2776 021372 001413          BEQ      4#          ;NO
2777 021374 023727 002120 005777 2#:    CMP      L#HIME,#05777 ;GREATER THAN 96K
2778 021402 101023          BHI      14#         ;YES,23A/23B WITH 128K MEMORY
2779 021404 023727 002120 003777          CMP      L#HIME,#03777 ;GREATER THAN 64K BUT LESS THAN 92K?
2780 021412 103403          BLO      4#          ;NO, CHECK 24K
2781 021414 004737 021476          JSR      PC,NXMTST   ;SETUP THE ADDRESS
2782 021420 000411          BR       13#         ;SET THE FLAG AND EXIT
2783 021422 023727 002120 001577 4#:    CMP      L#HIME,#01577 ;GREATER THAN 24K BUT LESS THAN 64K?
2784 021430 103410          BLO      14#         ;NO, TELL THEM AND EXIT WITH FLAG CLEAR
2785 021432 004737 021476          JSR      PC,NXMTST   ;SETUP THE ADDRESS
2786 021436 062737 000077 003132          ADD      #77,NXMHI   ;FOOL THE 11/02 & 11/03
2787 021444 005237 003126          INC      NXMFLG      ;SET THE FLAG
2788 021450 000411          BR       15#         ;EXIT
2789 021452 000410          BR       14#         ;NOP FOR PRINTOUT
2790 021454          PRINTF   @NXMEM     ;TELL THEM & EXIT ***NO PRINT*****
      021454 012746 005456          MOV      @NXMEM,-(SP)
      021460 012746 000001          MOV      #1,-(SP)
      021464 010600          MOV      SP,R0
      021466 104417          TRAP    C#PNTF
      021470 062706 000004          ADD      #4,SP
2791 021474 000207          15#:    RTS      PC          ;RETURN
2792
2793
2794          ;*
2795          ; SUBROUTINE TO SETUP THE NXM ADDRESS FOR TESTING
2796          ;
2797          ; OUTPUTS: NXMLO,NXMHI          ; SETUP WITH NXM ADDRESS
2798          ;
2799          ;-
2800 021476 013701 002120          NXMTST: MOV      L#HIME,R1          ;GET TOP OF MEMORY
2801 021502 062701 000200          ADD      #200,R1          ;MAKE IT I/O BLOCK OR OTHER NXM
2802 021506 042701 000177          BIC      #177,R1
2803 021512 010102          MOV      R1,R2          ;RESAVE RESULTS
2804          000006          .REPT   6
2805          .ASL   R1          ;PUT IN PLACE FOR XFER
2806          .ENDR
2807 021530 010137 003130          MOV      R1,NXMLO        ;SAVE TEST ADDRESS LOW
2808          000012          .REPT   10.
2809          .ASR   R2          ;PUT IN PLACE FOR XFER
2810          .ENDR
2811 021560 042702 177700          BIC      #177700,R2      ;DON'T WANT ILA!
2812 021564 010237 003132          MOV      R2,NXMHI        ;SAVE TEST ADDRESS HIGH
2813 021570 000207          RTS      PC          ;RETURN
2814
2815 021572          ENDMOD

```

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

SEQ 0093

```
7          .TITLE  TSV4 - MISCELLANEOUS SECTIONS
8
9 021572   BGNMOD  TSV4
021572   TSV4::
10
16
17
18
19          .SBTTL  PROTECTION TABLE
20 021572   BGNPROT
021572   L$PROT::
21 021572   177777 177777 177777   .WORD  -1. -1. -1. -1
22 021602   ENDPROT                ;NO DEVICE PROTECTION REQUIRED.
```

```

24                                     .SBTTL INITIALIZE SECTION
25
26                                     ;**
27                                     ;THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
28                                     ;AT THE BEGINNING OF EACH PASS.
29
30                                     ;IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
31                                     ;IF "CONTINUE", NOTHING IS REQUIRED.
32
33                                     ;--
34                                     ;*
35                                     ;INSERT TEMPORARY JUMP TO ODT
36                                     ;-
37 021602                                BGNINIT
    021602                                L$INIT::
38 021602 005037 002216                    40$: CLR      EXTFEA
39 021606 005037 003126                    CLR      NXMFLG
40 021612 012737 006356 002170            MOV      @EPRT1,EPRTSW           ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
41 021620 005037 003144                    CLR      SIFLAG                ;CLEAR "SOFT INIT" FLAG
42 021624 005037 003124                    CLR      KTENABLE              ;CLEAR TEST ABOVE 28K FLAG
43 021630 005037 002272                    CLR      RAMSIZ                ;CLEAR RAM SIZE FOR RAMERR ROUTINE
44 021634                                READEF   @EF.CONTINUE
    021634 012700 000036                    MOV      @EF.CONTINUE,RO
    021640 104447                                TRAP     C$REFG
45 021642                                BNCOMPLETE 1$
    021642 103023                                BCC     1$
46 021644 023737 002172 002012            CMP      UNITN,L$UNIT           ;UNIT IN RANGE?
47 021652 103070                                BHIS    4$                      ;BR IF NO.
48 021654 005737 003102                    TST     DUFLAG                 ;DROPPED UNIT?
49 021660 100472                                BMI     NXTU                    ;BR IF YES
50 021662 013701 002172                    MOV     UNITN,R1
51 021666 006301                                ASL     R1
52 021670 005761 003166                    TST     ERTABL(R1)
53 021674 001516                                BEQ     SETU
54 021676 032761 040000 003166            BIT     @BIT14,ERTABL(R1)       ;DROPPED?
55 021704 001060                                BNE     NXTU
56 021706                                EXIT     INIT                    ;DO NOTHING IF "CONTINUE".
    021706 104432                                TRAP     C$EXIT
    021710 000416                                .WORD   L10030-.
57 021712                                1$: READEF   @EF.NEW
    021712 012700 000035                    MOV     @EF.NEW,RO
    021716 104447                                TRAP     C$REFG
58 021720                                BNCOMPLETE NXTU                 ;TAKE NEXT UNIT IF NOT NEW PASS.
    021720 103052                                BCC     NXTU
59 021722                                READEF   @EF.START
    021722 012700 000040                    MOV     @EF.START,RO
    021726 104447                                TRAP     C$REFG
60 021730                                BCOMPLETE 2$
    021730 103404                                BCS     2$
61 021732                                READEF   @EF.RESTART
    021732 012700 000037                    MOV     @EF.RESTART,RO
    021736 104447                                TRAP     C$REFG
62 021740                                BNCOMPLETE 31$
    021740 103031                                BCC     31$
63 021742                                2$: BRESET
64 021742                                TRAP     C$RESET                ;1ST PASS, BUS-INIT...
    021742 104433                                ;BUS RESET.

```



TSV4 - MISCELLANEOUS SECTIONS  
INITIALIZE SECTION

MACRO M1113 14-JUN-84 14:17

SEQ 0095

```

65 021744 005037 002204          CLR      TSTCNT          ;NUMBER OF TESTS RUN IN PASS
66 021750 005037 002212          CLR      FATFLG         ;CLEAR FATAL ERROR COUNT
67 021754 005037 003134          CLR      T23A          ;CLEAR 11/23A FLAG
68 021760 005037 003136          CLR      T23B          ;CLEAR 11/23B FLAG
69                               ;      MOV      #340,-(SP)
70                               ;      MOV      #20,-(SP)          ;RETURN TO DEBUGGER
71                               ;      JMP      0.ODT          ;;ENTER THE DEBUGGER
72 021764 005037 003370          CLR      SKIPT          ;CLEAR THE SUBTEST "SKIPPER"
73 021770                               20$:
74 021770 012737 177777 002174    MOV      #-1,QVP        ;...QUICK VERIFY...
75 021776 004737 020720          JSR      PC,ENVIRN      ;SET ENVIRONMENT.
76 022002 004737 021100          JSR      PC,KTINIT     ;INITIALIZE KT MEMORY MANAGEMENT
77 022006 012700 003166          MOV      #ERTABL,RO
78 022012 005020 30$:          CLR      (RO)          ;CLEAR THE ERROR TABLE
79 022014 020027 003366          CMP      RO,#ERTABE
80 022020 103774          BLO     30$
81 022022 000404          BR      4$
82 022024 005037 002174 31$:    CLR      QVP
83 022030 000137 022100          JMP      PASRPT        ;GO REPORT THE STATUS
84
85 022034                               4$:
86 022034 012737 177777 002172    NEWPAS: MOV      #-1,UNITN ;INIT UNIT NUMBER...
87 022042 005037 062210          CLR      DEVCNT        ;CLEAR COUNT OF DEVICES RUNNING
88 022046                               NXTU:
89 022050 005237 002172          BREAK   C#BRK
90 022054 023737 002172 002012    TRAP   UNITN
91 022062 103423          INC    UNITN,L$UNIT   ;...AND SET NEXT UNIT NUMBER.
92 022064 012737 177777 003102    CMP    UNITN,L$UNIT
93 022072 000401          BLO   SETU
94 022074                               MOV    #-1,DUFLG
95 022076 000240          BR    11$
96 022100                               DOCLN  C#DCLN
97 022100 023727 002012 000001    TRAP   ;ABORT, NO MORE UNITS.
98 022106 101752          NOP
99 022110 005737 002210 11$:    PASRPT:
100 022114 001747          CMP    L$UNIT,#1      ;HOW MANY UNITS SELECTED?
101 022116                               BLOS   NEWPAS         ;BR IF ONLY 1
102 022120 032700 000100          TST   DEVCNT         ;ARE ANY STILL RUNNING?
103 022124 001343          BEQ   NEWPAS         ;BR IF NO
104                               RFLAGS RO
105 022126                               TRAP  C#RFLA
106 022130 000741          BIT   #ISR,RO        ;SHOULD WE PRINT STATISTICS
107 022132                               BNE   NEWPAS         ;BR IF NO
108                               DORPT  C#DRPT
109 022132                               TRAP  NEWPAS
110 022132 013700 002172          BR    10$
111 022136 104442          SETU:  GPHARD UNITN,RO ;GET UNIT N P-TABLE POINTER.
112 022140                               MOV   UNITN,RO
113 022142 005037 003102          TRAP  C#GPHRD
114 022146 005237 002210          BNCOMPLETE NXTU      ;BR IF UNIT NOT AVAILABLE.
115 022152 012001          BCC   NXTU
116 022154 010137 002176          CLR   DUFLG          ;CLEAR "DROPPED" FLAG.
117                               INC   DEVCNT
118                               MOV   (RO)+,R1        ;GET 1ST REGISTER ADDRESS.
119                               MOV   R1,CSRADDR     ;ADDRESS OF REGISTERS OF UNIT UNDER TEST

```

```

115
116 022160 012001      MOV      (R0)+,R1      ;GET VECTOR ADDRESS.
117                   ;MOV      (R0),R2      ;GET INTERRUPT PRIORITY
118                   ;MOV      R2,IPRI     ;SET INTERRUPT PRIORITY.
119 022162 010137 002200 MOV      R1,IVEC      ;SET INTERRUPT VECTOR POINTER...
120 022166 012721 016306 MOV      #INTR,(R1)+  ;...VECTOR...
121 022172 013721 002202 MOV      IPRI,(R1)+  ;...AND PRIORITY.
122
123 022176              1$:
124                   ;      TST      QVP          ;1ST PASS ??
125                   ;      BEQ      5$          ;NO, SKIP THE PASS 1 STUFF.
126
127                   ;
128                   ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
129                   ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
130                   ;
131 022176 013701 002172      MOV      UNITN,R1
132 022202 006301            ASL      R1
133 022204 052761 100000 003166 BIS      #BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
134 022212 005037 005770      CLR      EXTA          ;CLEAR ERROR EXTENSION FLAG.
135 022216 023727 002012 000001 CMP      L#UNIT,#1      ;ARE WE TESTING MULTIPLE UNITS?
136 022224 101416            BLOS    10$          ;BR IF NO.
137 022226            RFLAGS  RO          ;YES -- GET OPERATOR FLAGS.
138 022226 104421            TRAP   C#RFLA
139 022230 032700 001000      BIT      #PNT,RO        ;SHOULD WE PRINT UNIT #?
140 022234 001412            BEQ      10$          ;BR IF NOT.
141 022236            PRINTF  #PUNIT,UNITN ;PRINT THE UNIT #
142 022236 013746 002172      MOV      UNITN,-(SP)
143 022242 012746 022330      MOV      #PUNIT,-(SP)
144 022246 012746 000002      MOV      #2,-(SP)
145 022252 010600            MOV      SP,RO
146 022254 104417            TRAP   C#PNTF
147 022256 062706 000006      ADD      #6,SP
148 022262            10$:
149 022262 005037 003104      CLR      NODEV
150 022266 013701 002176      MOV      CSRADDR,R1   ;ADDRESS OF FIRST REGISTER
151 022272 010102            MOV      R1,R2        ;START OF REGISTERS
152 022274 062702 000002      ADD      #TSSR,R2    ;ADDRESS OF TSSR REGISTER
153 022300 004737 016466      JSR      PC,XNXM     ;TEST BOTH CONTROLLER REGISTERS...
154 022304 103005            BCC     2$          ;...AND BR IF ALL OK.
155 022306 010137 003104      MOV      R1,NODEV    ;FLAG DEVICE AS NON-EXISTENT
156 022312 012737 177777 003102 MOV      #-1,DUFLG   ;DROP THIS UNIT.
157 022320            2$:
158                   ;
159                   ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
160                   ;
161 022320            5$:      SETPRI  #PRI00      ;ENABLE INTERRUPTS.
162 022320 012700 000000      MOV      #PRI00,RO
163 022324 104441            TRAP   C#SPRI
164 022326            ENDINIT
165 022326 104411            L10030: TRAP   C#INIT
166 022330 045 116 045 PUNIT: .ASCIZ  /%N%N%A***** TESTING UNIT #D2%A *****/
167                   .EVEN

```

```

160                                     .SBTTL  ADD AND DROP UNITS SECTIONS
161
162                                     ;**
163                                     ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
164                                     ; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
165                                     ; OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
166                                     ;--
167 022376                                BGNAU
168 022376                                L$AU::
169 022400 010001                          MOV     R0,R1                ; GET UNIT TO BE ADDED (R0)
170 022402 052761 100000 003166          ASL     R1                  ; MAKE IT A WORD INDEX
171 022410 042761 040000 003166          BIS     #100000,ERTABL(R1) ; SET THE "ACTIVE" BIT
172 022416                                BIC     #40000,ERTABL(R1)  ; CLEAR THE "DROPPED" BIT
173 022416 010046                          PRINTF  #1$,R0
174 022420 012746 022444                  MOV     RO,-(SP)
175 022424 012746 000002                  MOV     #1$,-(SP)
176 022430 010600                          MOV     #2,-(SP)
177 022432 104417                          MOV     SP,R0
178 022434 062706 000006                  TRAP   C$PNTF
179 022440                                ADD     #6,SP
180 022440 000167                          EXIT    AU
181 022442 000026                          .WORD  J$JMP
182 022444 045 116 045 1$:                .WORD  L10031-2-.
183                                     .ASCIZ  /#N$A UNIT #D$A ADDED/
184                                     .EVEN
185
186                                     ENDAU                                ; UNUSED.
187
188 L10031:                                TRAP   C$AU
189
190                                     ;**
191                                     ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
192                                     ; TO BE REMOVED FROM THE TEST LIST.
193                                     ;
194                                     ; SUPVSR DOES THE "DROPPING". THIS IS JUST TO TELL THE MAN.
195                                     ; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
196                                     ; COMMAND, OTHERWISE REMAIN INACTIVE. THE "DISPLAY" COMMAND
197                                     ; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
198                                     ; WHICH ARE STILL ACTIVE.
199                                     ; UPON ENTRY, R0 CONTAINS THE UNIT TO BE DROPPED.
200
201                                     BGNDU
202 L$DU::
203 022474 012737 177777 003102          MOV     #-1,DUFLG
204 022502 010001                          MOV     R0,R1
205 022504 006301                          ASL     R1
206 022506 052761 140000 003166          BIS     #140000,ERTABL(R1) ; SAY DROPPED
207 022514 000240 000240 000240          240,240,240                ; ??????????
208 022522                                PRINTF  #1$,R0
209 022524 010046                          MOV     RO,-(SP)
210 022524 012746 022550                  MOV     #1$,-(SP)
211 022530 012746 000002                  MOV     #2,-(SP)
212 022534 010600                          MOV     SP,R0
213 022536 104417                          TRAP   C$PNTF
214 022540 062706 000006                  ADD     #6,SP
215 022544                                EXIT    DU
216 022544 000167                          .WORD  J$JMP
217 022546 000030                          .WORD  L10032-2-.

```





```

023026 012746 000002      MOV      #2,-(SP)
023032 010600      MOV      SP,R0
023034 104416      TRAP     C#PNTS
023036 062706 000006      ADD      #6,SP
254 023042 000431      BR       4$
255 023044 020227 160001      3$:     CMP      R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
256 023050 001012      BNE     30$      ; BR IF NO.
257 023052      PRINTS  #DEVNRD,R3
023052 010346      MOV      R3,-(SP)
023054 012746 023341      MOV      #DEVNRD,-(SP)
023060 012746 000002      MOV      #2,-(SP)
023064 010600      MOV      SP,R0
023066 104416      TRAP     C#PNTS
023070 062706 000006      ADD      #6,SP
258 023074 000414      BR       4$
259 023076 042702 170000      30$:    BIC      #C7777,R2
260 023102      PRINTS  #DEVDR0,R3,R2
023102 010246      MOV      R2,-(SP)
023104 010346      MOV      R3,-(SP)
023106 012746 023422      MOV      #DEVDR0,-(SP)
023112 012746 000003      MOV      #3,-(SP)
023116 010600      MOV      SP,R0
023120 104416      TRAP     C#PNTS
023122 062706 000010      ADD      #10,SP
261 023126 062704 000002      4$:     ADD      #2,R4
262 023132 005203      INC      R3
263 023134 020427 003366      CMP      R4,#ERTABE
264 023140 103701      BLO     1$
265 023142 012604      MOV      (SP)+,R4
266 023144 012603      MOV      (SP)+,R3
267 023146 012602      MOV      (SP)+,R2
268 023150      ENDRPT      ; UNUSED.
023150      L10035:
023150 104425      TRAP     C#RPT
269
270 023152      045      116      045  DEVSUM: .ASCIZ /#N#ADEVICE STATUS SUMMARY:#N/
271 023207      045      101      040  DEVONL: .ASCIZ /#A UNIT #D3#A ONLINE, ERRORS = #D#N/
272 023257      045      101      040  DEVNXR: .ASCIZ /#A UNIT #D3#A DROPPED, NON-EXISTENT REGISTER#N/
273 023341      045      101      040  DEVNRD: .ASCIZ /#A UNIT #D3#A DROPPED, NOT READY AT STARTUP#N/
274 023422      045      101      040  DEVDR0: .ASCIZ /#A UNIT #D3#A DROPPED, ERRORS = #D#N/
275      .EVEN
276
277 023472      ENDMOD
278

```

```

1          .TITLE  TSV7 - HARDWARE TESTS 1-8
2
9
10 023472  BGNMOD  TSV7
    023472  TSV7::

16
24          .SBTTL  TEST  1: INITIALIZE #4 TEST
25
26          ;*
27          ;
28          ;THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE
29          ;CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITIONS
30          ;(I.E. LOOPBACK ENABLED, FORCING WRONG PARITY, INVERTING SENSE OF
31          ;EXTENDED FEATURES SWITCH, ETC.)
32          ;
33          ;-
34 023472  BGNTST
    023472
35 023472  012737  006356  002170          MOV      #EPRT1,EPRTSW          ;SET UP PRIMARY ERROR MESSAGE
36
37          ;
38          ;
39          ;TEST 1
40          ;
41          ;
42          ;-
43
48 023500  004737  016274          JSR      PC,DSBINT          ;DISABLE INTERRUPTS
49 023504  012700  024434          MOV      #TST21ID,R0      ;ASCII MESSAGE TO IDENTIFY TEST
50 023510  004737  016600          JSR      PC,TSTSETUP      ;DO INITIAL TEST SETUP
51 023514  012737  000005  002206      MOV      #5,LOOPCNT       ;PERFORM 5 ITERATIONS
52 023522  T21LOOP:
53 023522  004737  024456          JSR      PC,T21REST       ;SET COMMAND PACKET
54 023526  004737  024546          JSR      PC,T21RT2        ;SET UP OTHER COMMAND PACKET
55
56          ;*****
57          ;
58          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
59          ;
60          ;*****
61
62 023532  012737  176750  024112      MOV      #65000.,T21DLY   ;SET DELAY ROUTINE
63 023540  004737  016064          JSR      PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
64 023544  103426          BCS     20$              ;BR IF INIT WAS OK
65 023546          DELAY  250            ;DELAY FOR A REWIND TO FINISH
    023546  012727  000250          MOV      #250,(PC)+
    023552  000000          .WORD  0
    023554  013727  002116          MOV      L$DLY,(PC)+
    023560  000000          .WORD  0
    023562  005367  177772          DEC     -6(PC)
    023566  001375          BNE     -.4
    023570  005367  177756          DEC     -22(PC)
    023574  001367          BNE     .-20
66 023576  005337  024112          DEC     T21DLY           ;BUMP COUNTER DOWN
67 023602  001356          BNE     11$             ;BR, IF MORE TIME TO GO
68 023604  005237  002212          INC     FATFLG          ;BUMP COUNT
72 023610  010001          MOV     R0,R1           ;CONTENTS OF TSSR REGISTER

```

```

73 023612          ERRDF  ERRNO,SFIERR,SFIMSG      ;FATAL ERROR TSSR WAS NOT OK
      023612 104455          TRAP  C$ERDF
      023614 000145          .WORD 101
      023616 003650          .WORD SFIERR
      023620 012124          .WORD SFIMSG
74 023622          20$:
75 023622 012704 024070      MOV    #T21PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
76
77      ;*****
78      ;
79      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
80      ;
81      ;*****
82
83 023626 013737 002172 024110      MOV    UNITN,T21DSW      ;SET UP DRIVE NUMBER
84 023634 004737 010752          JSR    PC,WRTPHR        ;ISSUE WRITE CHARACTERISTICS
85 023640 103407          BCS   23$              ;BR, IF COMMAND ISSUED OK
86 023642 005237 002212          INC   FATFLG           ;BUMP COUNT
90 023646 010001          MOV    R0,R1           ;SAVE CONTENTS OF TSSR
91 023650          ERRHRD  ERRNO,WRTPHR,SFIMSG      ;WRITE CHARACTERISTICS FAILED
      023650 104456          TRAP  C$ERHRD
      023652 000146          .WORD 102
      023654 005054          .WORD WRTPHR
      023656 012124          .WORD SFIMSG
92 023660          23$:  CKLOOP
      023660 104406          TRAP  C$CLP1
93 023662 112737 000200 024210      MOVB   #200,T21BS0      ;WRITE MISCELLANEOUS CONT/READ STATUS
94 023670 112737 000010 024211      MOVB   #10,T21BS1      ;FUNCTION SELECTION BIT
95 023676          25$:
96 023676 012704 024200          MOV    #T21PK2,R4      ;WRITE SUBSYS MEM PACKET
97 023702 010465 000000          MOV    R4,TSDB(R5)    ;ISSUE COMMAND
98 023706 004737 016426          JSR    PC,CHKTSSR     ;WAIT FOR SSR
99 023712 103407          BCS   30$              ;BR, IF NO ERROR
100 023714 010001          MOV    R0,R1           ;ERROR, SAVE TSSR
101 023716 005237 002212          INC   FATFLG           ;BUMP COUNT
105 023722          ERRHRD  ERRNO,T21SSR,PKTSSR      ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      023722 104456          TRAP  C$ERHRD
      023724 000147          .WORD 103
      023726 024216          .WORD T21SSR
      023730 012136          .WORD PKTSSR
106 023732          30$:  CKLOOP
      023732 104406          TRAP  C$CLP1
107 023734 012765 000000 000002      MOV    #0,TSSR(R5)    ;ISSUE A SOFT INITIALIZE
108 023742 004737 016340          JSR    PC,WAITF        ;WAIT FOR JUST THE SSR BIT TO SET
109 023746 016501 000002          MOV    TSSR(R5),R1    ;READ THE TSSR BACK
110 023752 010102          MOV    R1,R2           ;WORK REGISTER
111 023754 042702 176377          BIC   #C<HIADDR>,R2   ;CLEAR OUT OTHER BITS
112 023760 052702 002200          BIS   #SSR!NBA,R2    ;SOME OF THE BITS THAT SHOULD BE SET
113 023764 032701 000100          BIT   #OFL,R1        ;IS OFF LINE BIT SET
114 023770 001012          BNE   38$              ;BR, IF DRIVE IS OFF LINE
115 023772 020102          35$:  CMP    R1,R2        ;EXPECTED (R2) = RECEIVED (R1)
116 023774 001406          BEQ   37$              ;BR, IF THEY ARE EQUAL (OK)
117 023776 005237 002212          INC   FATFLG           ;BUMP COUNT
121 024002          ERRHRD  ERRNO,T21AM3,EXPREC      ;"ERROR TRYING TO INIT AFTER WRITE MISC.
      024002 104456          TRAP  C$ERHRD
      024004 000150          .WORD 104
      024006 024313          .WORD T21AM3

```



TSV7 - HARDWARE TESTS 1-8  
TEST 1: INITIALIZE #4 TEST

MACRO M1113 14-JUN-84 14:17

SEQ 0103

```

    024010 015564
122 024012          37$: CKLOOP          ;LOOP IF SELECTED          .WORD  EXPREC
    024012 104406
123 024014 000406          BR          40$          ;SKIP OVER OFF-LINE STUFF      TRAP   C$CLP1
124 024016          38$: ERRDF  ERRNO,T21OFL,EXPREC  ;DRIVE IS OFF LINE
128 024016          024016 104455          TRAP   C$ERDF
    024020 000151          .WORD  105
    024022 024413          .WORD  T21OFL
    024024 015564          .WORD  EXPREC
129 024026 004737 017272          JSR    PC,CKDROP          ;TRY AND DROP UNIT
130 024032 000241          40$:  CLC          ;DON'T LET CARRY SNEAK IN
131 024034 106037 024211          RORB   T21BS1          ;TRY NEXT "LOWEST" BIT POSITION
132 024040 001316          BNE    25$          ;LOOP UNTIL ALL EIGHT BITS TESTED
133 024042          50$:  CKLOOP          ;SCOPE LOOP
    024042 104406          JSR    PC,TSTLOOP          ;DO WE NEED TO ITERATE TEST      TRAP   C$CLP1
134 024044 004737 016546          BCC    63$          ;BR, IF NO LOOP REQUIRED
135 024050 103002          JMP    T21LOOP          ;EXECUTE AGAIN
136 024052 000137 023522          63$:  EXIT  TST          ;ALL DONE THIS TEST
137 024056          024056 104432          TRAP   C$EXIT
    024060 000530          .WORD  L10036-.

138
139
140          ;*
141          ;LOCAL STORAGE FOR THIS TEST
142          ;-
143 024062          .BLKB  10-<.-TSV2E7>
145 024070          T21PACKET:          ;COMMAND PACKET FOR TEST
146 024070 100004          .WORD  100004          ;WRITE CHARACTERISTICS COMMAND, WITH, ACK
147 024072 024100          .WORD  T21DATA          ;ADDRESS OF CHARACTERISTICS BLOCK
148 024074 000000          .WORD  0
149 024076 000012          .WORD  10.          ;STARTING VALUE OF BLOCK SIZE
150 024100          T21DATA:          ;CHARACTERISTICS DATA BLOCK
151 024100 024114          .WORD  T21BFR          ;ADDRESS OF MESSAGE BUFFER
152 024102 000000          .WORD  0
153 024104 000024          .WORD  20.          ;LENGTH OF MESSAGE BUFFER
154 024106 000000          .WORD  0
155 024110 000000          T21DSW: .WORD  0          ;DRIVE SELECT WORD
156 024112 000000          T21DLY: .WORD  0          ;DELAY COUNTER
157 024114          T21BFR: .BLKW  25.          ;MESSAGE BUFFER
158
159          ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
160          ;
162 024176          .BLKB  10-<.-TSV2E7>
164 024200          T21PK2:
165 024200 100206          .WORD  100206          ;WRITE SUB SYS MEM COMMAND, IE AND ACK
166 024202 024210          .WORD  T21BF2          ;ADDRESS OF SELECT BLOCK DATA
167 024204 000000          .WORD  0
168 024206 000006          .WORD  6.          ;SIZE OF DATA PACKET
169
170          .EVEN
171 024210          T21BF2:
172 024210          T21BS0: .BYTE  0          ;BSELO AREA --- "COMMAND" BYTE
173 024211          T21BS1: .BYTE  0          ;BSEL1 AREA
174 024212 000000          T21S2: .WORD  0          ;SEL 2 AREA
175 024214 000000          T21S3: .WORD  0          ;DATA AREA
176

```

TSV7 - HARDWARE TESTS 1-8  
TEST 1: INITIALIZE #4 TEST

MACRO M1113 14-JUN-84 14:17

SEQ 0104

```

177
178
179      ;*
180      ;LOCAL TEXT MESSAGES FOR TEST
181      ;-
182 024216      127      122      111  T21SSR: .ASCIZ 'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
183 024313      124      123      123  T21AM3: .ASCIZ 'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
184 024413      104      162      151  T21OFL: .ASCIZ 'Drive is OFFLINE'
185 024434      111      156      151  T21ID:  .ASCIZ 'Initialization #4'
186
187
188      ;*
189      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
190      ;WRITE SUBSYSTEM MEMORY COMMAND
191      ;
192      ;-
193
194 024456      T21REST:
195 024456      SAVREG
196 024462      012701  024070      MOV      #T21PACKET,R1      ;SAVE THE REGISTERS
197 024466      012721  100004      MOV      #100004,(R1)+      ;START OF THE PACKET
198 024472      012721  024100      MOV      #T21DATA,(R1)+    ;WRITE SUBSYSTEM MEM. WITH ACK,
199 024476      005021      CLR      (R1)+              ;ADDRESS OF CHARAISTICS DATA BLOCK
200 024500      012721  000010      MOV      #8.,(R1)+         ;EXTENDED ADDRESS
201 024504      012721  024114      MOV      #T21BFR,(R1)+     ;SIZE OF DATA BLOCK IN BYTES
202 024510      005021      CLR      (R1)+              ;ADDRESS OF MESSAGE BUFFER
203 024512      012721  000024      MOV      #20.,(R1)+        ;LENGTH OF MESSAGE BUFFER
204 024516      005021      CLR      (R1)+
205 024520      005011      CLR      (R1)
206 024522      012702  000020      MOV      #20,R2            ;NUMBER OF LOCATIONS TO BE CLEARED
207 024526      012762  177777  024114  64$:  MOV      #177777,T21BFR(R2) ;ALL ONES TO MESSAGE BIJFFER
208 024534      005742      TST      -(R2)              ;NEXT LOCATION
209 024536      020227  000000      CMP      R2,#0             ;CHECK R2 FOR ZERO
210 024542      001371      BNE      64$                ;BR, IF NOT AT ZERO YET
211 024544      000207      RTS      PC                 ;RETURN
212
213
214 024546      T21RT2:
215 024546      SAVREG
216 024552      012701  024200      MOV      #T21PK2,R1        ;SAVE THE REGISTERS
217 024556      012721  100206      MOV      #100206,(R1)+     ;START OF THE PACKET
218 024562      012721  024210      MOV      #T21BF2,(R1)+    ;WRITE SUBSYSTEM MEM. WITH ACK, IE
219 024566      005021      CLR      (R1)+              ;ADDRESS OF DATA BLOCK
220 024570      012721  000006      MOV      #6.,(R1)+         ;EXTENDED ADDRESS
221 024574      005021      CLR      (R1)+              ;SIZE OF DATA BLOCK IN BYTES
222 024576      012701  024210      MOV      #T21BF2,R1        ;ADDRESS OF DATA FOR WRT SUB SYS MEM
223 024602      005021      CLR      (R1)+
224 024604      005011      CLR      (R1)
225 024606      000207      RTS      PC                 ;RETURN
226 024610      ENDTST
227      024610      L10036:
228      024610      TRAP      C$ETST
229
230      ;*
231      ;.SBTTL TEST 2: OFF-LINE AND REJECT REWIND
232
233      ;THIS TEST VERIFIES BASIC TAPE-MOTION COMMAND DECODING AND BASIC

```



```

289
290 ;
291 ;*****
292 024704 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
293 024710 103407 BCS 23; ;BR, IF COMMAND ISSUED OK
294 024712 005237 002212 INC FATFLG ;BUMP COUNT
298 024716 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
299 024720 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
    024720 104456 TRAP C$ERHRD
    024722 000312 .WORD 202
    024724 005054 .WORD WRTMSG
    024726 012124 .WORD SFIMSG
300 024730 23: CKLOOP TRAP C$CLP1
    024730 104406
301 024732 013701 026210 MOV T22FR*6,R1 ;PICK UP XT50
302 024736 032701 000004 BIT #4,R1 ;IS UNIT WRITE-LOCKED?
303 024742 001407 BEQ 24; ;NO,PROCEED WITH TESTING
304 024744 005237 002212 INC FATFLG ;BUMP COUNT
308 024750 ERRDF ERRNO,T22WLK,SFIMSG ;TAPE IS WRITE LOCKED
    024750 104455 TRAP C$ERDF
    024752 000313 .WORD 203
    024754 027012 .WORD T22WLK
    024756 012124 .WORD SFIMSG
309 024760 DOCLN TRAP C$DOCLN
    024760 104444
310 024762 24: CKLOOP TRAP C$CLP1
    024762 104406
311 024764 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
312 024770 001041 JNE 50; ;BR IF SWITCH IS ON
313 024772 112737 000200 026301 MOVB #200,T22BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
314 025000 112737 000010 026300 MOVB #10,T22BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
315 025006 012704 026270 MOV #T22PK2,R4 ;WRITE SUBSYS MEM PACKET
316 025012 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
317 025016 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
318 025022 103407 BCS 30; ;BR, IF NO ERROR
319 025024 010001 MOV RO,R1 ;ERROR, SAVE TSSR
320 025026 005237 002212 INC FATFLG ;BUMP COUNT
324 025032 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
    025032 104456 TRAP C$ERHRD
    025034 000314 .WORD 204
    025036 026320 .WORD T22SSR
    025040 012136 .WORD PKTSSR
325 025042 30: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
    025042 104406
326 025044 012704 026160 MOV #T22PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
327
328 ;*****
329 ;
330 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
331 ;
332 ;*****
333
334 025050 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
335 025054 103407 BCS 50; ;BR, IF COMMAND ISSUED OK
336 025056 005237 002212 INC FATFLG ;BUMP COUNT
340 025062 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
341 025064 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED

```











```

025712 104456                                TRAP    C$ERHRD
025714 000330                                .WORD  216
025716 026320                                .WORD  T22SSR
025720 012136                                .WORD  PKTSSR
544 025722 30$: CKLOOP                        ;LOOP IF SELECTED
025722 104406                                TRAP    C$CLP1
545 025724 012704 026160                    MOV     #T22PACKET,R4                ;SUBROUTINE NEEDS PACKET ADDRESS
546
547 ;*****
548 ;
549 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
550 ;
551 ;*****
552
553 025730 004737 010752                    JSR     PC,WRTCHR                    ;ISSUE WRITE CHARACTERISTICS
554 025734 103407                            BCS    50$                          ;BR, IF COMMAND ISSUED OK
555 025736 005237 002212                    INC     FATFLG                       ;BUMP COUNT
559 025742 010001                            MOV     R0,R1                        ;SAVE CONTENTS OF TSSR
560 025744                                ERRHRD  ERRNO,WRTMSG,SFIMSG         ;WRITE CHARACTERISTIC FAILED
025744 104456                                TRAP    C$ERHRD
025746 000331                                .WORD  217
025750 005054                                .WORD  WRTMSG
025752 012124                                .WORD  SFIMSG
561 025754 50$: CKLOOP                        ;SCOPE LOOP
025754 104406                                TRAP    C$CLP1
562 025756 016501 000002                    MOV     TSSR(R5),R1                 ;GET TSSR CONTENTS
563 025762 032701 000100                    BIT     #OFL,R1                     ;CHECK FOR THE OFFLINE BIT SET
564 025766 001006                            BNE    60$                          ;BR, IF OFFLINE (GOOD)
565 025770 005237 002212                    INC     FATFLG                       ;BUMP COUNT
569 025774                                ERRDF  ERRNO,T22OFL,SFIMSG         ;OFF LINE SHOULD HAVE BEEN SET (BAD)
025774 104455                                TRAP    C$ERDF
025776 000332                                .WORD  218
026000 026515                                .WORD  T22OFL
026002 012124                                .WORD  SFIMSG
570 026004 60$: CKLOOP                        ;LOOP IF SELECTED
026004 104406                                TRAP    C$CLP1
571 026006 012737 142010 026270 65$: MOV     #142010,T22PK2                ;POSITION COMMAND (REWIND MODE) CVC-1
572 026014 012704 026270                    MOV     #T22PK2,R4                  ;R4 = POINTER TO PACKET
573 026020 010465 000000                    MOV     R4,TSDB(R5)                 ;ISSUE COMMAND
574 026024 004737 016340                    JSR     PC,WAITF                     ;WAIT FOR SSR TO SET
575 026030 016501 000002                    MOV     TSSR(R5),R1                 ;GET TSSR CONTENTS
576 026034 012702 100306                    MOV     #SSR!OFL!SC!BIT1!BIT2,R2   ;SET UP EXPECTED
577 026040 020102                            CMP     R1,R2                       ;ARE THEY EQUAL
578 026042 001406                            BEQ    80$                          ;BR, IF OK ESP. FUNCTION REJECT
579 026044 005237 002212                    INC     FATFLG                       ;BUMP COUNT
583 026050                                ERRHRD  ERRNO,T22RWJ,EXPREC         ;TSSR INCORRECT AFTER TAPE MOTION CMD
026050 104456                                TRAP    C$ERHRD
026052 000333                                .WORD  219
026054 026664                                .WORD  T22RWJ
026056 015564                                .WORD  EXPREC
584 026060 80$: CKLOOP                        ;LOOP IF SELECTED
026060 104406                                TRAP    C$CLP1
585 026062 012703 026202                    MOV     #T22BFR,R3                  ;POINTER TO MESSAGE BUFFER
586 026066 016301 000006                    MOV     XSTO(R3),R1                 ;PICK UP XSTO FROM MESSAGE BUFFER
587 026072 010102                            MOV     R1,R2                       ;SET UP EXPECTED
588 026074 042702 000020                    BIC     #BIT4,R2                    ;VCK SHOULD BE CLEAR
589 026100 020102                            CMP     R1,R2                       ;ARE THEY EQUAL
    
```

TSV7 - HARDWARE TESTS 1-8      MACRO M1113 14-JUN-84 14:17  
TEST 2: OFF-LINE AND REJECT REWIND

SEQ 0112

```

590 026102 001406           BEQ      90$           ;BR, IF OK (GOOD)
591 026104 005237 002212   INC      FATFLG      ;BUMP COUNT
595 026110           ERRHRD  ERRNO,T22VCK,EXPREC ;VCK WASN'T CLEAR (BAD)
      026110 104456           TRAP    C$ERHRD
      026112 000334           .WORD  220
      026114 026737           .WORD  T22VCK
      026116 015564           .WORD  EXPREC
596 026120           90$:
597 026120           ENDSUB
      026120           ;>>>>>>>>>> END SUBTEST >>>>>>>>>
      026120           L10042:
      026122 104403           TRAP    C$ESUB
598 026122 023727 002212 000017   CMP      FATFLG,#15. ;IS ERROR COUNT AT 25
599 026130 103402           BLO     999$         ;BR, IF LESS THAN 25
600 026132 004737 017272   JSR     PC,CKDROP   ;TRY TO DROP THE UNIT
601 026136           999$:
602 026136 004737 016546   JSR     PC,TSTLOOP ;DO WE NEED TO ITERATE TEST
603 026142 103002           BCC     163$         ;BR, IF NO LOOP REQUIRED
604 026144 000137 024642   JMP     T22LOOP    ;EXECUTE AGAIN
605 026150           163$:
      026150 104432           EXIT    TST         ;ALL DONE THIS TEST
      026152 001116           TRAP    C$EXIT
                        .WORD  L10037-..

606
607
608      ;*
609      ;LOCAL STORAGE FOR THIS TEST
610      ;-
611 026154           .BLKB  10-<.-TSV2&7>
612 026160           T22PACKET:
      026160 100204           .WORD  100204       ;COMMAND PACKET FOR TEST
      026162 026170           .WORD  T22DATA      ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
      026164 000000           .WORD  0            ;ADDRESS OF CHARACTERISTICS BLOCK
      026166 000012           .WORD  10.         ;STARTING VALUE OF BLOCK SIZE
618 026170           T22DATA:
      026170 026202           .WORD  T22BFR       ;CHARACTERISTICS DATA BLOCK
      026172 000000           .WORD  0            ;ADDRESS OF MESSAGE BUFFER
621 026174 000024           .WORD  20.         ;LENGTH OF MESSAGE BUFFER
622 026176 000000           .WORD  0
623 026200 000007           .WORD  7
624 026202           T22BFR: .BLKW  25.   ;SELECT DRIVE 7
                        ;MESSAGE BUFFER
625
626      ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
627      ;
629 026264           .BLKB  10-<.-TSV2&7>
631 026270           T22PK2:
      026270 100206           .WORD  100206       ;WRITE SUB SYS MEM COMMAND, IE AND ACK
      026272 026300           .WORD  T22BF2       ;ADDRESS OF SELECT BLOCK DATA
634 026274 000000           .WORD  0
635 026276 000006           .WORD  6.         ;SIZE OF DATA PACKET
636
637      .EVEN
638 026300           T22BF2:
639 026300 000           T22BS0: .BYTE  0     ;BSEL0 AREA
640 026301 000           T22BS1: .BYTE  0     ;BSEL1 AREA
641 026302 000000       T22S2:  .WORD  0     ;SEL 2 AREA
642 026304 000000       T22S3:  .WORD  0     ;DATA AREA
643
644      ;
645      .EVEN

```

```

646                   ;TAPE MOTION PACKET COMMAND VALUES
647 026306   100201   T22RD:  .WORD   100201               ;READ TAPE FORWARD
648 026310   100205   T22WRT: .WORD   100205               ;WRITE TAPE FORWARD
649 026312   100210   T22POS: .WORD   100210               ;POSITION TAPE
650 026314   100211   T22FOR: .WORD   100211               ;FORMAT TAPE
651 026316   177777                .WORD   177777               ;END OF DATA
652
653
654                   ;+
655               ;LOCAL TEXT MESSAGES FOR TEST
656               ;-
657
658 026320       127       122       111   T22SSR: .ASCIZ  'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
659 026415       124       123       123   T22AM3: .ASCIZ  'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
660 026515       104       162       151   T22OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
661 026570       124       123       123   T22TM:  .ASCIZ  'TSSR Incorrect After Tape Motion Command To Off-Line Device'
662 026664       124       123       123   T22RWJ: .ASCIZ  'TSSR Not Correct After REWIND With VCK Set'
663 026737       103       126       103   T22VCK: .ASCIZ  'CVC Set, Didn't Reset VCK In Message Buffer'
664 027012       052       052       052   T22WLK: .ASCIZ  '*****TAPE IS WRITE-LOCKED AND WILL CAUSE ERRORS*****'
665 027077       117       146       146   TST22ID: .ASCIZ  'Off-Line And Reject Rewind'
666                                .EVEN
667                   ;+
668               ;
669               ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
670               ;WRITE SUBSYSTEM MEMORY COMMAND
671               ;
672               ;-
673
674 027132           T22REST:
675 027132                SAVREG                   ;SAVE THE REGISTERS
676 027136   012701   026160           MOV       #T22PACKET,R1           ;START OF THE PACKET
677 027142   012721   100204           MOV       #100204,(R1)+       ;WRITE SUBSYSTEM MEM. WITH ACK, IE
678 027146   012721   026170           MOV       #T22DATA,(R1)+     ;ADDRESS OF CHARAISTICS DATA BLOCK
679 027152   005021                    CLR       (R1)+            ;EXTENDED ADDRESS
680 027154   012721   000012           MOV       #10.,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
681 027160   012721   026202           MOV       #T22BFR,(R1)+     ;ADDRESS OF MESSAGE BUFFER
682 027164   005021                    CLR       (R1)+
683 027166   012721   000024           MOV       #20.,(R1)+       ;LENGTH OF MESSAGE BUFFER
684 027172   005021                    CLR       (R1)+
685 027174   012711   000007           MOV       #7,(R1)           ;SELECT DRIVE SEVEN
686 027200   012702   000020           MOV       #20,R2           ;NUMBER OF LOCATIONS TO BE CLEARED
687 027204   012762   177777   026202   64$:  MOV       #177777,T22BFR(R2)   ;ALL ONES TO MESSAGE BUFFER
688 027212   005742                    TST       -(R2)           ;BUMP R2 DOWN
689 027214   020227   000000           CMP       R2,#0           ;IS R2 AT ZERO YET
690 027220   001371                    BNE       64$            ;KEEP GOING UNTIL DONE
691 027222   000207                    RTS       PC             ;RETURN
692
693
694 027224           T22RT2:
695 027224                SAVREG                   ;SAVE THE REGISTERS
696 027230   012701   026270           MOV       #T22PK2,R1       ;START OF THE PACKET
697 027234   012721   100206           MOV       #100206,(R1)+     ;WRITE SUBSYSTEM MEM. WITH ACK, IE
698 027240   012721   026300           MOV       #T22BF2,(R1)+    ;ADDRESS OF DATA BLOCK
699 027244   005021                    CLR       (R1)+           ;EXTENDED ADDRESS
700 027246   012721   000006           MOV       #6.,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
701 027252   005021                    CLR       (R1)+
702 027254   012701   026300           MOV       #T22BF2,R1       ;POINT TO DATA SEL AREA

```



```

759 027354 004737 016064      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
760 027360 103407              BCS    20$             ;BR IF INIT WAS OK
761 027362 005237 002212      INC    FATFLG          ;BUMP COUNT
765 027366 010001              MOV    R0,R1           ;CONTENTS OF TSSR REGISTER
766 027370              ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   301
                                .WORD   SFIERR
                                .WORD   SFIMSG
767 027400              20$:
768 027400 012737 000007 032600  MOV    #7,T23DSW      ;SET DRIVE NUMBER IN PACKET
769 027406 012704 032560      MOV    #T23PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
770
771 ;*****
772 ;
773 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
774 ;
775 ;*****
776
777 027412 004737 010752      JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
778 027416 103407              BCS    23$             ;BR, IF COMMAND ISSUED OK
779 027420 005237 002212      INC    FATFLG          ;BUMP COUNT
783 027424 010001              MOV    R0,R1           ;SAVE CONTENTS OF TSSR
784 027426              ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP    C$ERHRD
                                .WORD   302
                                .WORD   WRTMSG
                                .WORD   SFIMSG
785 027436 005737 002216      23$:  TST    EXTFEA        ;CHECK FOR EXTENDED FEATURES SW SWITCH
786 027442 001044              BNE    50$             ;BR IF SWITCH IS ON
787
788 027444 112737 000200 032723  MOVB   #200,T23BS1    ;WRITE MISCELLANEOUS CONT/READ STATUS
789 027452 112737 000010 032722  MOVB   #10,T23BS0     ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
790 027460 012704 032670      MOV    #T23PK2,R4     ;WRITE SUBSYS MEM PACKET
791 027464 010465 000000      MOV    R4,TSDB(R5)    ;ISSUE COMMAND
792 027470 004737 016426      JSR    PC,CHKTSSR     ;WAIT FOR SSR
793 027474 103407              BCS    30$             ;BR, IF NO ERROR
794 027476 010001              MOV    R0,R1           ;ERROR, SAVE TSSR
795 027500 005237 002212      INC    FATFLG          ;BUMP COUNT
799 027504              ERRHRD ERRNO,T23SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
                                TRAP    C$ERHRD
                                .WORD   303
                                .WORD   T23SSR
                                .WORD   PKTSSR
800 027514              30$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C$CLP1
801 027516 012737 000007 032600  MOV    #7,T23DSW      ;SET DRIVE NUMBER IN PACKET
802 027524 012704 032560      MOV    #T23PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
803
804 ;*****
805 ;
806 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
807 ;
808 ;*****
809
810 027530 004737 010752      JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
811 027534 103407              BCS    50$             ;BR, IF COMMAND ISSUED OK

```

812	027536	005237	002212	INC	FATFLG	;	BUMP COUNT		
816	027542	010001		MOV	RO,R1	;	SAVE CONTENTS OF TSSR		
817	027544			ERRHRD	ERRNO,WRTMSG,SFIMSG	;	WRITE CHARACTERISTIC FAILED		
	027544	104456					TRAP	C#ERHRD	
	027546	000460					.WORD	304	
	027550	005054					.WORD	WRTMSG	
	027552	012124					.WORD	SFIMSG	
818	027554			50\$:	CKLOOP	;	SCOPE LOOP		
	027554	104406					TRAP	C#CLP1	
819	027556	016501	000002	MOV	TSSR(R5),R1	;	GET TSSR CONTENTS		
820	027562	032701	000100	BIT	#OFL,R1	;	CHECK FOR THE OFFLINE BIT SET		
821	027566	001006		BNE	60\$	;	BR, IF OFFLINE (GOOD)		
822	027570	005237	002212	INC	FATFLG	;	BUMP COUNT		
826	027574			ERRDF	ERRNO,T23OFL,SFIMSG	;	OFF LINE SHOULD HAVE BEEN SET (BAD)		
	027574	104455					TRAP	C#ERDF	
	027576	000461					.WORD	305	
	027600	033406					.WORD	T23OFL	
	027602	012124					.WORD	SFIMSG	
827	027604			60\$:	CKLOOP	;	LOOP IF SELECTED		
	027604	104406					TRAP	C#CLP1	
828	027606	012703	032734	MOV	#T23WD,R3	;	POINTER FOR COMMANDS		
829	027612	011337	032670	65\$:	MOV (R3),T23PK2	;	TAPE MOTION COMMAND IN PLACE		
830	027616	012704	032670	MOV	#T23PK2,R4	;	R4 = POINTER TO PACKET		
831	027622	010465	000000	MOV	R4,TSDB(R5)	;	ISSUE COMMAND		
832	027626	004737	016340	JSR	PC,WAITF	;	WAIT FOR SSR TO SET		
833	027632	016501	000002	MOV	TSSR(R5),R1	;	GET TSSR CONTENTS		
834	027636	012702	100306	MOV	#SSR!SC!OFL!BIT1!BIT2,R2	;	SET UP EXPECTED		
835	027642	020102		CMP	R1,R2	;	ARE THEY EQUAL		
836	027644	001406		BEQ	80\$	;	BR, IF OK ESP. FUNCTION REJECT		
837	027646	005237	002212	INC	FATFLG	;	BUMP COUNT		
841	027652			ERRHRD	ERRNO,T23TM,EXPREC	;	TSSR INCORRECT AFTER TAPE MOTION CMD		
	027652	104456					TRAP	C#ERHRD	
	027654	000462					.WORD	306	
	027656	033142					.WORD	T23TM	
	027660	015564					.WORD	EXPREC	
842	027662			80\$:	CKLOOP	;	LOOP IF SELECTED		
	027662	104406					TRAP	C#CLP1	
843	027664	005723		TST	(R3).	;	POINT TO NEXT COMMAND		
844	027666	022713	177777	CMP	#177777,(R3)	;	END OF THE COMMANDS YET		
845	027672	001401		BEQ	90\$	;	BR, IF DONE		
846	027674	000746		BR	65\$	;	MORE COMMAND(S) TO GO		
847	027676			90\$:	ENDSUB	;	;>>>>>>>>> END SUBTEST >>>>>>>>>		
848	027676						L10044:		
	027676	104403					TRAP	C#ESUB	
849	027700	023727	002212 000017	CMP	FATFLG,#15.	;	IS ERROR COUNT AT 25		
850	027706	103402		BLO	999\$	;	BR, IF LESS THAN 25		
851	027710	004737	017272	JSR	PC,CKDROP	;	TRY TO DROP THE UNIT		
852	027714			999\$:					
853									
854				::					
855				:					
856				;	TEST 3, SUBTEST 2				
857				:					
858				;	VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE				
859				;	SWAP BYTES (SWB) BIT CLEAR OPERATES PROPERLY. THE				
860				;	BYTE COUNT (RECORD SIZE) VARIES FROM 20 THROUGH 64K				

```
861 ;IN VARYING INCREMENTS (DEPENDING UPON WHETHER OR NOT
862 ;THE DIAGNOSTIC IS RUNNING IN THE LONG VERIFICATION
863 ;MODE). THE TAPE IS NOT REWOUND BETWEEN SUCCESSIVE
864 ;RECORDS BUT IS REWOUND AFTER THE FINAL RECORD IS
865 ;WRITTEN. AN INCREMENTING COUNT PATTERN IS SUPPLIED
866 ;IN THE DATA BUFFER. AFTER EACH BLOCK IS WRITTEN, THE
867 ;TSSR AND TSBA REGISTERS, THE MESSAGE BUFFER, AND THE
868 ;RAM CONTENTS ARE CHECKED. (THE RAM CONTENTS ARE CHECKED
869 ;USING THE WRITE SUBSYSTEM MEMORY COMMAND.
870 ;
871 ;
872 ;
873 ;
874 027714 BGNSUB ;>>>>>>>>>>>> BEGIN SUBTEST :>>>>>>>>>>>>
      027714 T3.2:
      027714 104402 TRAP C#BSUB
875 027716 013701 003116 MOV FRESIZ,R1 ;GET SIZE OF AVAILIABLE MEMORY
876 027722 000241 CLC ;CLEAR THE CARRY BIT
877 027724 006101 ROL R1 ;MAKE INTO BYTES
878 027726 010137 032720 MOV R1,T23RSZ ;STORE IN TEST FOR USE LATER
879 027732 004737 034102 JSR PC,T23REST ;SET COMMAND PACKET
880 027736 004737 034236 JSR PC,T23RT3 ;RESTORE PACKET
881 027742 004737 034174 JSR PC,T23RT2 ;SET UP OTHER COMMAND PACKET
882
883 ;*****
884 ;
885 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
886 ;
887 ;*****
888
889 027746 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
890 027752 103407 BCS 20$ ;BR IF INIT WAS OK
891 027754 005237 002212 INC FATFLG ;BUMP COUNT
895 027760 010001 MOV RO,R1 ;CONTENTS OF TSSR REGISTER
896 027762 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      027762 104455 TRAP C$ERDF
      027764 000463 .WORD 307
      027766 003650 .WORD SFIERR
      027770 012124 .WORD SFIMSG
897 027772 20$:
898 027772 013737 002172 032600 MOV UNITN,T23DSW ;LOAD UP SELECTED UNIT NUMBER
899 030000 012704 032560 MOV #T23PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
900
901 ;*****
902 ;
903 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
904 ;
905 ;*****
906
907 030004 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
908 030010 103407 BCS 23$ ;BR, IF COMMAND ISSUED OK
909 030012 005237 002212 INC FATFLG ;BUMP COUNT
913 030016 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
914 030020 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      030020 104456 TRAP C$ERHRD
      030022 000464 .WORD 308
      030024 005054 .WORD WRTMSG
```

```

915 030026 012124
030030 104406
916
917
918
919
920
921
922
923 030032 004737 011104 JSR PC,REWIND ;CALL THE TAPE REWIND
924 030036 012703 000024 MOV #20.,R3 ;STARTING RECORD SIZE
925 030042 013737 003114 032712 65: MOV FREE,T23WB ;STARTING WRITE BUFFER ADDRESS
926
927
928
929 ;WRITE DATA,CVC=1,ACK COMMAND
930
931
932
933 030050 012737 140005 032710 MOV #140005,T23PK3 ;WRITE DATA,CVC=1,ACK COMMAND
934 030056 012737 140005 032732 MOV #140005,T23WRT ;SETUP FOR RETRY COMMAND
935 030064 052737 004000 032732 BIS #4000,T23WRT ;MAKE IT A RETRY
936 030072 012704 032710 MOV #T23PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
937 030076 010300 MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
938 030100 004737 017512 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
939 030104 010337 032716 MOV R3,T23SZ ;SET UP RECORD SIZE IN PACKET
940 030110 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
941 030114 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
942 030120 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
943 030124 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
944 030130 020102 CMP R1,R2 ;ARE THEY EQUAL
945 030132 001402 BEQ 80: ;BR, IF OK
946 030134 004737 034262 JSR PC,T23CHK ;CHECK SPECIAL CONDITION
947 030140 80: CKLOOP ;LOOP IF SELECTED
030140 104406 TRAP C:CLP1
948 030142 016501 000000 MOV TSBA(R5),R1 ;GET TSBA CONTENTS
949 030146 012702 032602 MOV #T23BFR,R2 ;SET UP EXPECTED
950 030152 062702 000016 ADD #16,R2 ;SET TO END OF MESSAGE BUFFER
951 030156 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SET
952 030162 001402 BEQ 85: ;BR, IF IT NOT SET
953 030164 062702 000002 ADD #2,R2 ;BUMP R2 FOR EXTRA DATA
954 030170 020102 85: CMP R1,R2 ;ARE THEY EQUAL
955 030172 001406 BEQ 90: ;BR, IF TSBA IS CORRECT
956 030174 005237 002212 INC FATFLG ;BUMP COUNT
960 030200 ERRHRD ERRNO,T23BA,EXPREC ;TSBA WAS NOT CORRECT AFTER WRITE DATA
030200 104456 TRAP C:ERHRD
030202 000465 .WORD 309
030204 033725 .WORD T23BA
030206 015564 .WORD EXPREC
961 030210 90: CKLOOP ;LCOP IF SELECTED
030210 104406 TRAP C:CLP1
962 030212 020327 007376 CMP R3,#7376 ;ONLY CHECK RAM UNTIL ITS FULL
963 030216 002114 BGE 115: ;IT WRAPS AROUND ETC.
964 030220 004737 034174 JSR PC,T23RT2 ;MAKE SURE PACKET AND DATA ARE CLEAN
965 030224 012737 000400 032724 MOV #256.,T23S2 ;STARTING RAM ADDRESS
966 030232 112737 000000 032722 MOVB #0,T23BS0 ;STOP INTERNAL TSV05 DIAGNOSTICS

```



```

967 030240 112737 000000 032723      MOV      #0,T23B51      ;SIZE OF RAM READ
968 030246 012704 032670      MOV      #T23PK2,R4    ;SET R4 WITH PACKET ADDRESS
969 030252 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE WRITE SUB SYS MEM COMMAND
970 030256 004737 016426      JSR      PC,CHKTSSR    ;CHECK TSSR AND WAIT FOR SSR TO SET
971 030262 103407                BCS      92#           ;BR, IF NO ERRORS IN TSSR
972 030264 010001                MOV      R0,R1         ;SAVE TSSR
973 030266 005237 002212      INC      FATFLG        ;BUMP COUNT
977 030272                ERRHRD   ERRNO,T23WSS,PKTSSR ;TSSR BAD AFTER WRITE SUB SYS MEM
                                TRAP      C#ERHRD
                                .WORD    310
                                .WORD    T23WSS
                                .WORD    PKTSSR
                                030272 104456
                                030274 000466
                                030276 033777
                                030300 012136
978 030302                92# :   CKLOOP        ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                030302 104406
979 030304 004737 034174      JSR      PC,T23RT2     ;MAKE SURE PACKET AND DATA ARE CLEAN
980 030310 012737 000400 032724      MOV      #256.,T23S2  ;STARTING RAM ADDRESS
981 030316 112737 000001 032722      MOV      #1,T23B50    ;READ RAM COMMAND FOR WRITE SUB SYS M.
982 030324 112737 000002 032723      MOV      #2,T23B51    ;SIZE OF RAM READ
983 030332 012704 032670      MOV      #T23PK2,R4   ;SET R4 WITH PACKET ADDRESS
984 030336 010465 000000      95# :   MOV      R4,TSDB(R5) ;ISSUE WRITE SUB SYS MEM COMMAND
985 030342 004737 016426      JSR      PC,CHKTSSR    ;CHECK TSSR AND WAIT FOR SSR TO SET
986 030346 103407                BCS      100#         ;BR, IF NO ERRORS IN TSSR
987 030350 010001                MOV      R0,R1         ;SAVE TSSR
988 030352 005237 002212      INC      FATFLG        ;BUMP COUNT
992 030356                ERRHRD   ERRNO,T23WSS,PKTSSR ;TSSR BAD AFTER WRITE SUB SYS MEM
                                TRAP      C#ERHRD
                                .WORD    311
                                .WORD    T23WSS
                                .WORD    PKTSSR
                                030356 104456
                                030360 000467
                                030362 033777
                                030364 012136
993 030366                100# : CKLOOP        ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                030366 104406
994 030370 005001                CLR      R1            ;CLEAR REGISTER
995 030372 005002                CLR      R2            ;CLEAR REGISTER
996 030374 013701 032622      MOV      T23BFR+20,R1 ;PICK UP BYTE READ FROM RAM
997 030400 010302                MOV      R3,R2        ;SET UP EXPECTED
998 030402 020102                CMP      R1,R2        ;IS RAM DATA CORRECT
999 030404 001406                BEQ      110#         ;BR, IF OK (EQUAL)
1000 030406 005237 002212      INC      FATFLG        ;BUMP COUNT
1004 030412                ERRHRD   ERRNO,T23RNC,EXPREC ;RNC=RAM NOT CORRECT
                                TRAP      C#ERHRD
                                .WORD    312
                                .WORD    T23RNC
                                .WORD    EXPREC
                                030412 104456
                                030414 000470
                                030416 033265
                                030420 015564
1005 030422                110# : CKLOOP        ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                030422 104406
1006 030424 005237 032724      INC      T23S2         ;BUMP RAM ADDRESS TO BE CHECKED
1007 030430 005237 032724      INC      T23S2         ;BUMP RAM ADDRESS TO BE CHECKED
1008 030434 010301                MOV      R3,R1        ;GET SIZE OF RECORD
1009 030436 062701 000400      ADD      #256.,R1     ;FIGURE OUT END RECORD ADDRESS
1010 030442 023701 032724      CMP      T23S2,R1     ;AT END OF RAM CHECK YET
1011 030446 001333                BNE      95#          ;BR, IF MORE TO CHECK
1012 030450 062703 001750      115# : ADD      #1000.,R3 ;NEXT RECORD SIZE/DATA PATTERN
1013 030454 020337 032720      CMP      R3,T23RSZ    ;IS R3 OVER MAX RECORD SIZE
1014 030460 002005                BGE      120#         ;IF RECORD SIZE IS TOO BIG QUIT
1015 030462 020327 177776      CMP      R3,#65534.   ;END OF SUBTEST MAX RECORD SIZE
1016 030466 001402                BEQ      120#         ;BR, IF COMPLETED
1017 030470 000137 030042      JMP      65#          ;DO MORE RECORDS

```



```

1069 030622          20$:
1070 030622 013737 002172 032600      MOV     UNITN,T23DSW      ;SET UP UNIT NUMBER
1071 030630 012704 032560          MOV     @T23PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
1072
1073          ;*****
1074          ;
1075          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHK)
1076          ;
1077          ;*****
1078
1079 030634 004737 010752          JSR     PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
1080 030640 103407          BCS     23$             ;BR, IF COMMAND ISSUED OK
1081 030642 005237 002212          INC     FATFLG          ;BUMP COUNT
1085 030646 010001          MOV     R0,R1           ;SAVE CONTENTS OF TSSR
1086 030650          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP     C$ERHRD
                                .WORD    315
                                .WORD    WRTMSG
                                .WORD    SFIMSG
1087 030660
1088 030660 012703 000024          23$:  MOV     @20.,R3        ;STARTING RECORD SIZE
1089 030664 013737 003114 032712 65$:  MOV     FREE,T23WB      ;STARTING WRITE BUFFER ADDRESS
1090
1091          ;*****
1092          ;
1093          ;WRITE DATA,CVC=1,ACK,SWB COMMAND
1094          ;
1095          ;*****
1096
1097 030672 012737 150005 032710          MOV     @150005,T23PK3  ;WRITE DATA,CVC=1,ACK,SWB COMMAND
1098 030700 012737 150005 032732          MOV     @150005,T23WRT  ;SETUP FOR RETRY COMMAND
1099 030706 052737 004000 032732          BIS     @4000,T23WRT    ;MAKE IT A RETRY
1100 030714 012704 032710          MOV     @T23PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
1101 030720 010300          MOV     R3,R0          ;SET PATTERN IN CORRECT REGISTER
1102 030722 004737 017512          JSR     PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
1103 030726 010337 032716          MOV     R3,T23SZ       ;SET UP RECORD SIZE IN PACKET
1104 030732 010465 000000          MOV     R4,TSDB(R5)    ;ISSUE COMMAND
1105 030736 004737 016340          JSR     PC,WAITF        ;WAIT FOR SSR TO SET
1106 030742 016501 000002          MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
1107 030746 012702 000200          MOV     @SSR,R2        ;SET UP EXPECTED
1108 030752 020102          CMP     R1,R2          ;ARE THEY EQUAL
1109 030754 001402          BEQ     80$            ;BR, IF OK
1110 030756 004737 034262          JSR     PC,T23CHK      ;CHECK SPECIAL CONDITION
1111 030762          CKLOOP              ;LOOP IF SELECTED
                                TRAP     C$CLP1
1112 030764 016501 000000          MOV     TSBA(R5),R1    ;GET TSBA CONTENTS
1113 030770 012702 032602          MOV     @T23BFR,R2    ;SET UP EXPECTED
1114 030774 062702 000016          ADD     @16,R2         ;SET TO END OF MESSAGE BUFFER
1115 031000 005737 002216          TST     EXTFEA         ;CHECK FOR EXTENDED FEATURES SW SET
1116 031004 001402          BEQ     85$            ;BR, IF IT NOT SET
1117 031006 062702 000002          ADD     @2,R2          ;BUMP R2 FOR EXTRA DATA
1118 031012 020102          85$:  CMP     R1,R2          ;ARE THEY EQUAL
1119 031014 001406          BEQ     90$            ;BR, IF TSBA IS CORRECT
1120 031016 005237 002212          INC     FATFLG         ;BUMP COUNT
1124 031022          ERRHRD  ERRNO,T23BA,EXPREC ;TSBA WAS NOT CORRECT AFTER WRITE DATA
                                TRAP     C$ERHRD
                                .WORD    316
                                031022 104456
                                031024 000474

```

```

031026 033725 .WORD T238A
031030 015564 .WORD EXPREC
1125 031032 904: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
031032 104406 ;ONLY CHECK RAM UNTIL ITS FULL
1126 031034 020327 007376 CMP R3,#7376 ;IT WRAPS AROUND ETC.
1127 031040 002115 BGE 1154 ;MAKE SURE PACKET AND DATA ARE CLEAN
1128 031042 004737 034174 JSR PC,T23RT2 ;STARTING RAM ADDRESS
1129 031046 012737 000400 032724 MOV #256.,T23S2 ;STOP INTERNAL TSV05 DIAGNOSTICS
1130 031054 112737 000000 032722 MOVB #0,T238S0 ;SIZE OF RAM READ
1131 031062 112737 000000 032723 MOVB #0,T238S1 ;SET R4 WITH PACKET ADDRESS
1132 031070 012704 032670 MOV #T23PK2,R4 ;ISSUE WRITE SUB SYS MEM COMMAND
1133 031074 010465 000000 MOV R4,TSDB(R5) ;CHECK TSSR AND WAIT FOR SSR TO SET
1134 031100 004737 016426 JSR PC,CHKTSSR ;BR, IF NO ERRORS IN TSSR
1135 031104 103407 BCS 924 ;SAVE TSSR
1136 031106 010001 MOV R0,R1 ;BUMP COUNT
1137 031110 005237 002212 INC FATFLG ;TSSR BAD AFTER WRITE SUB SYS MEM
1141 031114 ERRHRD ERRNO,T23WSS,PKTSSR TRAP C$ERHRD
031114 104456 .WORD 317
031116 000475 .WORD T23WSS
031120 033777 .WORD PKTSSR
031122 012136
1142 031124 924: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
031124 104406 ;MAKE SURE PACKET AND DATA ARE CLEAN
1143 031126 004737 034174 JSR PC,T23RT2 ;STARTING RAM ADDRESS
1144 031132 012737 000400 032724 MOV #256.,T23S2 ;READ RAM COMMAND FOR WRITE SUB SYS M.
1145 031140 112737 000001 032722 MOVB #1,T238S0 ;SIZE OF RAM READ
1146 031146 112737 000002 032723 MOVB #2,T238S1 ;SET R4 WITH PACKET ADDRESS
1147 031154 012704 032670 MOV #T23PK2,R4 ;ISSUE WRITE SUB SYS MEM CMD (READ RAM)
1148 031160 010465 000000 954: MOV R4,TSDB(R5) ;CHECK TSSR AND WAIT FOR SSR TO SET
1149 031164 004737 016426 JSR PC,CHKTSSR ;BR, IF NO ERRORS IN TSSR
1150 031170 103407 BCS 1004 ;SAVE TSSR
1151 031172 010001 MOV R0,R1 ;BUMP COUNT
1152 031174 005237 002212 INC FATFLG ;TSSR BAD AFTER WRITE SUB SYS MEM
1156 031200 ERRHRD ERRNO,T23WSS,PKTSSR TRAP C$ERHRD
031200 104456 .WORD 318
031202 000476 .WORD T23WSS
031204 033777 .WORD PKTSSR
031206 012136
1157 031210 1004: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
031210 104406 ;CLEAR REGISTERS
1158 031212 005001 CLR R1 ;CLEAR REGISTERS
1159 031214 005002 CLR R2 ;PICK UP BYTE READ FROM RAM
1160 031216 013701 032622 MOV T23BFR+20,R1 ;SET UP EXPECTED
1161 031222 010302 MOV R3,R2 ;SWAP BYTES
1162 031224 000302 SWAB R2 ;IS RAM DATA CORRECT
1163 031226 020102 CMP R1,R2 ;BR, IF OK (EQUAL)
1164 031230 001406 BEQ 1104 ;BUMP COUNT
1165 031232 005237 002212 INC FATFLG ;RNC=RAM NOT CORRECT
1169 031236 ERRHRD ERRNO,T23RNC,EXPREC TRAP C$ERHRD
031236 104456 .WORD 319
031240 000477 .WORD T23RNC
031242 033265 .WORD EXPREC
031244 015564
1170 031246 1104: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
031246 104406 ;BUMP RAM ADDRESS TO BE CHECKED
1171 031250 005237 032724 INC T23S2 ;BUMP RAM ADDRESS TO BE CHECKED
1172 031254 005237 032724 INC T23S2

```

TSV7 - HARDWARE TESTS 1-8  
TEST 3: BASIC WRITE DATA

MACRO M1113 14-JUN-84 14:17

SEQ 0123

```

1173 031260 010301      MOV     R3,R1           ;GET SIZE OF RECORD
1174 031262 062701 000400  ADD     #256.,R1       ;FIGURE OUT END RECORD ADDRESS
1175 031266 023701 032724  CMP     T23S2,R1      ;AT END OF RAM CHECK YET
1176 031272 001332      BNE     95#           ;BR, IF MORE TO CHECK
1177 031274 062703 001750 115#:  ADD     #1000.,R3     ;NEXT RECORD SIZE/DATA PATTERN
1178 031300 020337 032720  CMP     R3,T23RSZ     ;IS R3 OVER MAX RECORD SIZE
1179 031304 002005      BGE     120#         ;IF RECORD SIZE IS TOO BIG QUIT
1180 031306 020327 177776  CMP     R3,#65534.    ;END OF SUBTEST MAX RECORD SIZE
1181 031312 001402      BEQ     120#         ;BR, IF COMPLETED
1182 031314 000137 030664  JMP     65#           ;DO MORE RECORDS
1183 031320          120#:
1184 031320 004737 034174      JSR     PC,T23RT2     ;CLEAN UP PACKET
1185 031324 012737 102010 032670  MOV     #102010,T23PK2 ;REWIND (POSITION) COMMAND
1186 031332 012704 032670  MOV     #T23PK2,R4    ;LOAD R4 WITH PACKET ADDRESS
1187 031336 010465 000000  MOV     R4,TSD8(R5)   ;ISSUE REWIND COMMAND
1188 031342 004737 016426  JSR     PC,CHKTSSR    ;WAIT FOR SSR TO SET
1189 031346 103407      BCS     130#         ;BR, IF TSSR IS OK (GOOD)
1190 031350 010001      MOV     R0,R1         ;SAVE TSSR CONTENTS
1191 031352 005237 002212  INC     FATFLG        ;BUMP COUNT
1195 031356          ERRHRD  ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND
           031356 104456          TRAP     C$ERHRD
           031360 000500          .WORD   320
           031362 033216          .WORD   T23RWN
           031364 012136          .WORD   PKTSSR
1196 031366          130#:
1197 031366          ENDSUB                ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
           031366          L10046:
           031366 104403          TRAP     C$ESUB
1198 031370 023727 002212 000017  CMP     FATFLG,#15.   ;IS ERROR COUNT AT 25
1199 031376 103402      BLO     999#         ;BR, IF LESS THAN 25
1200 031400 004737 017272  JSR     PC,CKDROP    ;TRY TO DROP THE UNIT
1201 031404          999#:
1202          ;*
1203          ;
1204          ;TEST 3, SUBTEST 4
1205          ;
1206          ;VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL MODE
1207          ;FIELD OR AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH
1208          ;THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE
1209          ;
1210          ;
1211          ;
1212          ;-
1213          ;
1214 031404          BGNSUB                ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
           031404          T3.4:
           031404 104402          TRAP     C$BSUB
1215 031406 004737 034102  JSR     PC,T23REST    ;SET COMMAND PACKET
1216 031412 004737 034236  JSR     PC,T23RT3     ;RESTORE PACKET
1217 031416 004737 034174  JSR     PC,T23RT2     ;SET UP OTHER COMMAND PACKET
1218          ;
1219          ;*****
1220          ;
1221          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
1222          ;
1223          ;*****
1224          ;

```



```

1276 031604 023727 002212 000017      CMP      FATFLG,#15.      ;IS ERROR COUNT AT 25
1277 031612 103402                  BLO      999$           ;BR, IF LESS THAN 25
1278 031614 004737 017272            JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
1279 031620                 999$;
1280
1281                      ;*
1282                      ;
1283                      ;TEST 3, SUBTEST 5
1284                      ;
1285                      ;VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA
1286                      ;BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH
1287                      ;THE PROPER ERROR STATUS WITHOUT MOVING TAPE
1288                      ;
1289                      ;
1290                      ;-
1291                      BGNSUB           ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
                                           T3.5:
                                           TRAP      C$BSUB
1292 031622 104402 005737 003126      TST      NXMFLG          ;DO WE HAVE IT?
1293 031626 001002                  BNE      10$             ;BR, IF ENOUGH
1294 031630 000137 032150            JMP      90$             ;SKIP THIS TEST IF NOT
1295 031634 004737 034102          10$:   JSR      PC,T23REST     ;SET COMMAND PACKET
1296 031640 004737 034236            JSR      PC,T23RT3      ;RESTORE PACKET
1297 031644 004737 034174            JSR      PC,T23RT2      ;SET UP OTHER COMMAND PACKET
1298
1299                      ;*****
1300                      ;
1301                      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
1302                      ;
1303                      ;*****
1304
1305 031650 004737 016064            JSR      PC,SOFINIT     ;DO INITIALIZE ON CONTROLLER
1306 031654 103407                  BCS      20$             ;BR IF INIT WAS OK
1307 031656 005237 002212            INC      FATFLG          ;BUMP COUNT
1311 031662 010001                  MOV      R0,R1           ;CONTENTS OF TSSR REGISTER
1312 031664            ERRDF      ERRNO,SFIERR,SFIMSG    ;FATAL ERROR TSSR WAS NOT OK
                                           TRAP      C$ERDF
                                           .WORD    324
                                           .WORD    SFIERR
                                           .WORD    SFIMSG
1313 031674            20$:   MOV      UNITN,T23DSW      ;SET DRIVE NUMBER UP
1314 031674 013737 002172 032600    MOV      @T23PACKET,R4  ;SUBROUTINE NEEDS PACKET ADDRESS
1315 031702 012704 032560
1316
1317                      ;*****
1318                      ;
1319                      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
1320                      ;
1321                      ;*****
1322
1323 031706 004737 010752            JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
1324 031712 103407                  BCS      123$           ;BR, IF COMMAND ISSUED OK
1325 031714 005237 002212            INC      FATFLG          ;BUMP COUNT
1329 031720 010001                  MOV      R0,R1           ;SAVE CONTENTS OF TSSR
1330 031722            ERRHRD      ERRNO,WRTMSG,SFIMSG    ;WRITE CHARACTERISTICS FAILED
                                           TRAP      C$ERHRD
                                           .WORD    325
1330 031722 104456
1330 031724 000505

```

```

031726 005054 .WORD WRTMSG
031730 012124 .WORD SFIMSG

1331
1332 ;*****
1333 ;
1334 ;WRITE DATA, ACK, CVC=1
1335 ;
1336 ;*****
1337
1338 031732 123$:
1339 031732 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
1340 031736 001026 BNE 130$ ;BR IF SWITCH IS ON
1341 031740 005237 002216 INC EXTFEA ;ONLY ONE TIME
1342 031744 112737 000200 032723 MOVB #200,T23B51 ;WRITE MISCELLANEOUS CONT/READ STATUS
1343 031752 112737 000010 032722 MOVB #10,T23B50 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
1344 031760 012704 032670 MOV #T23PK2,R4 ;WRITE SUBSYS MEM PACKET
1345 031764 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
1346 031770 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
1347 031774 103407 BCS 130$ ;BR, IF NO ERROR
1348 031776 010001 MOV R0,R1 ;ERROR, SAVE TSSR
1349 032000 005237 002212 INC FATFLG ;BUMP COUNT
1353 032004 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
                                TRAP C$ERHRD
                                .WORD 326
                                .WORD T22SSR
                                .WORD PKTSSR
032004 104456
032006 000506
032010 026320
032012 012136
1354 032014 130$: CKLOOP ;LOOP IF SELECTED
032014 104406 TRAP C$CLP1
1355
1356 032016 012701 160000 MOV #160000,R1 ;NXM LOW ADDRESS START
1357 032022 012702 177776 MOV #177776,R2 ;LIMIT CHECK FOR NXM (HIGHEST)
1358 032026 004737 016466 JSR PC, NXM ;LOOK FOR NXM ADDRESS
1359 032032 103045 BCC 80$ ;BR, IF NON FOUND
1360 032034 010137 003130 MOV R1,NXMLO ;SET ADDRESS UP FOR TEST
1361
1362
1363 032040 005037 032714 CLR T23WB+2 ;CLEAR OUT THE HIGH BITS AREA
1364 032044 24$:
1365 032044 012737 140005 032710 MOV #140005,T23PK3 ;WRITE DATA, ACK, CVC=1
1366 032052 013737 003130 032712 MOV NXMLO,T23WB ;SET UP WRITE BUFFER ADDRESS
1367 032060 012737 000100 032716 MOV #64.,T23SZ ;SET UP BUFFER SIZE
1368 032066 012704 032710 MOV #T23PK3,R4 ;R4 = POINTER TO PACKET
1369 032072 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
1370 032076 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
1371 032102 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
1372 032106 012702 104210 MOV #SC!NXM!SSR!BIT3,R2 ;SET UP EXPECTED
1373 032112 020102 CMP R1,R2 ;ARE THEY EQUAL
1374 032114 001414 BEQ 80$ ;BR, IF OK ESP. FUNCTION REJECT
1375 032116 005237 032714 INC T23WB+2 ;BUMP TO NEXT ADDRESS BIT
1376 032122 023727 032714 000004 CMP T23WB+2,#4 ;CHECK TO SEE IF OVERFLOW INTO 19 BIT
1377 032130 001345 BNE 24$ ;BR, IF BITS 17 AND 18
1378 032132 005237 002212 25$: INC FATFLG ;BUMP COUNT
1382 032136 ERRHRD ERRNO,T23TM,PKTSSR ;TSSR INCORRECT AFTER WRITE COMMAND
                                TRAP C$ERHRD
                                .WORD 327
                                .WORD T23TM
                                .WORD PKTSSR
032136 104456
032140 000507
032142 033142
032144 012136

```





```

1434 032254 004737 010752      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
1435 032260 103407              BCS      23$           ;BR, IF COMMAND ISSUED OK
1436 032262 005237 002212      INC      FATFLG        ;BUMP COUNT
1440 032266 010001              MOV      R0,R1         ;SAVE CONTENTS OF TSSR
1441 032270              ERRHRD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                                TRAP      C$ERHRD
                                .WORD     329
                                .WORD     WRTMSG
                                .WORD     SFIMSG
1442
1443      ;*****
1444      ;
1445      ;WRITE DATA, ACK,CVC=1
1446      ;
1447      ;*****
1448
1449 032300      23$:      MOV      #160000,R1     ;NXM LOW ADDRESS START
1450 032300 012701 160000      MOV      #177776,R2     ;LIMIT CHECK FOR NXM (HIGHEST)
1451 032304 012702 177776      JSR      PC, XNXM       ;LOOK FOR NXM ADDRESS
1452 032310 004737 016466      BCC      80$           ;BR, IF NON FOUND
1453 032314 103051              MOV      R1,NXML0      ;SET ADDRESS UP FOR TEST
1454 032316 010137 003130      MOV      #0,T23WB+2    ;SET TO 16 BIT ADDRESS
1455 032322 012737 000000 032714
1456 032330      24$:      MOV      #140005,T23PK3  ;WRITE DATA, ACK,CVC=1
1457 032330 012737 140005 032710      MOV      NXML0,R1      ;HIGHEST MEMORY ADDRESS LOW BITS
1458 032336 013701 003130      SUB      #500,R1       ;SET ADDRESS A LITTLE LOWER
1459 032342 162701 000500      MOV      R1,T23WB      ;LOAD INTO THE PACKET
1460 032346 010137 032712      MOV      #0.,T23SZ     ;SET UP BUFFER SIZE (64K BYTES)
1461 032352 012737 000000 032716      MOV      #T23PK3,R4    ;R4 = POINTER TO PACKET
1462 032360 012704 032710      MOV      R4,TSDB(R5)   ;ISSUE COMMAND
1463 032364 010465 000000      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
1464 032370 004737 016340      MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
1465 032374 016501 000002      MOV      #SC!NXM!SSR!BIT3,R2 ;SET UP EXPECTED
1466 032400 012702 104210      CMP      R1,R2        ;ARE THEY EQUAL
1467 032404 020102              BEQ      80$           ;BR, IF OK ESP. FUNCTION REJECT
1468 032406 001414              INC      T23WB+2      ;BUMP TO NEXT ADDRESS RANGE
1469 032410 005237 032714      CMP      T23WB+2,#4    ;CHECK TO SEE IF WE WENT TO HIGH
1470 032414 023727 032714 000004      BNE      24$          ;BR, IF NO OVER FLOW
1471 032422 001342              25$:      INC      FATFLG        ;BUMP COUNT
1472 032424 005237 002212      ERRHRD   ERRNO,T23TM,PKTSSR ;TSSR INCORRECT AFTER WRITE COMMAND
                                TRAP      C$ERHRD
                                .WORD     330
                                .WORD     T23TM
                                .WORD     PKTSSR
1476 032430      80$:      CKLOOP      ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD     104456
                                .WORD     000512
                                .WORD     033142
                                .WORD     012136
1477 032440      104406
1478 032442 004737 034174      JSR      PC,T23RT2     ;CLEAN UP PACKET
1479 032446 004737 034236      JSR      PC,T23RT3     ;RESTORE PACKET
1480 032452 012737 102010 032670      MOV      #102010,T23PK2 ;REWIND (POSITION) COMMAND
1481 032460 012704 032670      MOV      #T23PK2,R4    ;LOAD R4 WITH PACKET ADDRESS
1482 032464 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE REWIND COMMAND
1483 032470 004737 016426      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
1484 032474 103407              BCS      130$         ;BR, IF TSSR IS OK (GOOD)
1485 032476 010001              MOV      R0,R1         ;SAVE TSSR CONTENTS
1486 032500 005237 002212      INC      FATFLG        ;BUMP COUNT
1490 032504      ERRHRD   ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND

```

	032504	104456					TRAP	C\$ERHRD	
	032506	000513					.WORD	331	
	032510	033216					.WORD	T23RWN	
	032512	012136					.WORD	PKTSSR	
1491	032514			130\$:					
1492	032514			ENDSUB					
	032514								
	032514	104403							
1493	032516	023727	002212	000017	CMP	FATFLG,#15.		TRAP	C\$ESUB
1494	032524	103402			BLO	999\$		;IS ERROR COUNT AT 25	
1495	032526	004737	017272		JSR	PC,CKDROP		;BR, IF LESS THAN 25	
1496	032532				999\$:			;TRY TO DROP THE UNIT	
1497	032532	004737	016546		JSR	PC,TSTLOOP			
1498	032536	103002			BCC	163\$		;DO WE NEED TO ITERATE TEST	
1499	032540	000137	027342		JMP	T23LOOP		;BR, IF NO LOOP REQUIRED	
1500	032544				163\$:			;EXECUTE AGAIN	
1501	032544				EXIT	TST			
	032544	104432						;ALL DONE THIS TEST	
	032546	001702						TRAP	C\$EXIT
								.WORD	L10043-
1502									
1503									
1504									
1505									
1507	032550								
1509	032560								
1510	032560	100004							
1511	032562	032570							
1512	032564	000000							
1513	032566	000010							
1514	032570								
1515	032570	032602							
1516	032572	000000							
1517	032574	000012							
1518	032576	000000							
1519	032600	000000							
1520	032602								
1521									
1522									
1523									
1525	032664								
1527	032670								
1528	032670	100006							
1529	032672	032722							
1530	032674	000000							
1531	032676	000006							
1532									
1534	032700								
1536	032710								
1537	032710	100005							
1538	032712	000000							
1539	032714	000000							
1540	032716	000000							
1541									
1542									
1543	032720	000000							
1544									
1545									

```

;+
;LOCAL STORAGE FOR THIS TEST
;-
      .BLKB   10-<.-TSV2&7>
T23PACKET:
      .WORD   100004
      .WORD   T23DATA
      .WORD   0
      .WORD   8.
T23DATA:
      .WORD   T23BFR
      .WORD   0
      .WORD   10.
      .WORD   0
T23DSW: .WORD 0
T23BFR: .BLKW 25.
;
;WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
      .BLKB   10-<.-TSV2&7>
T23PK2:
      .WORD   100006
      .WORD   T23BFR2
      .WORD   0
      .WORD   6.
;
      .BLKB   10-<.-TSV2&7>
T23PK3:
      .WORD   100005
T23WB: .WORD 0
      .WORD   0
T23SZ: .WORD 0
      .EVEN
;
T23RSZ: .WORD 0
;
;

```

```

1546 032722          T23BF2:
1547 032722          T23BS0: .BYTE 10          ;BSEL0 AREA
1548 032723          T23BS1: .BYTE 200        ;BSEL1 AREA
1549 032724          T23S2:  .WORD 0          ;SEL 2 AREA
1550 032726          T23S3:  .WORD 0          ;DATA AREA
1551
1552
1553 032730          T23TMP: .WORD 0          ;TEMPORARY REGISTER
1554 032732          T23WRT: .WORD 0          ;RETRY COMMAND
1555
1556
1557
1558
1559 032734          T23WD:  .WORD 100005       ;WRITE DATA (NEXT)
1560 032736          T23WDR: .WORD 100405       ;WRITE DATA RETRY
1561 032740          T23CON: .WORD 102005       ;WRITE CONTINUOUS
1562 032742          .WORD 177777           ;END OF DATA
1563
1564
1565
1566
1567
1568 032744          127      122      111  T23SSR: .ASCIZ 'WRITE Command Not Accepted'
1569 032777          105      117      124  T23ET:  .ASCIZ 'EOT Not Found In 12000 4k Writes, (Use Shorter Tape)'
1570 033064          127      122      111  T23EOT: .ASCIZ 'WRITE DATA OVER EOT GAVE NO TAPE STATUS ALERT'
1571 033142          124      123      123  T23TM:  .ASCIZ 'TSSR Not Correct After WRITE Command Reject'
1572 033216          122      145      167  T23RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
1573 033265          122      101      115  T23RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
1574 033340          124      123      123  T23AM3: .ASCIZ 'TSSR Init. Failed After WRITE Command'
1575 033406          104      162      151  T23OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
1576 033461          124      123      123  T23WDD: .ASCIZ 'TSSR Not Correct After WRITE DATA Command, SWB Bit Set'
1577 033550          124      123      123  T23WDC: .ASCIZ 'TSSR Not Correct After WRITE DATA Command, Check For Tape Offline'
1578 033652          103      126      103  T23VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
1579 033725          124      123      102  T23BA:  .ASCIZ 'TSBA Not Correct After WRITE DATA Command'
1580 033777          127      122      111  T23WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
1581 034066          102      141      163  TST23ID: .ASCIZ 'Basic Write'
1582
1583
1584
1585
1586
1587
1588
1589
1590 034102          T23REST:
1591 034102          SAVREG
1592 034106          012701  032560          MOV      #T23PACKET,R1          ;SAVE THE REGISTERS
1593 034112          012721  100004          MOV      #100004,(R1)+         ;START OF THE PACKET
1594 034116          012721  032570          MOV      #T23DATA,(R1)+       ;WRITE SUBSYSTEM MEM. WITH ACK
1595 034122          005021                      CLR      (R1)+                 ;ADDRESS OF CHARAISTICS DATA BLOCK
1596 034124          012721  000012          MOV      #10.,(R1)+           ;EXTENDED ADDRESS
1597 034130          012721  032602          MOV      #T23BFR,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
1598 034134          005021                      CLR      (R1)+                 ;ADDRESS OF MESSAGE BUFFER
1599 034136          012721  000024          MOV      #20.,(R1)+           ;LENGTH OF MESSAGE BUFFER
1600 034142          005021                      CLR      (R1)+
1601 034144          012711  000000          MOV      #0,(R1)              ;SELECT DRIVE ZERO
1602 034150          012702  000030          MOV      #24.,R2              ;NUMBER OF LOCATIONS TO BE CLEARED
;TAPE MOTION PACKET COMMAND VALUES
;LOCAL TEXT MESSAGES FOR TEST
;+
;-
;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
;WRITE SUBSYSTEM MEMORY COMMAND
;+
;-

```

```

1603 034154 012762 177777 032602 64:  MOV    #177777,T23BFR(R2)  ;ALL ONES TO MESSAGE BUFFER
1604 034162 005742                TST    -(R2)                ;BUMP DOWN TO NEXT LOCATION
1605 034164 020227 000000                CMP    R2,#0                ;R2 AT ZERO YET
1606 034170 001371                BNE   64:                  ;KEEP GOING UNTIL DONE
1607 034172 000207                RTS    PC                    ;RETURN
1608
1609
1610 034174                T23RT2:
1611 034174                SAVREG                      ;SAVE THE REGISTERS
1612 034200 012701 032670                MOV    #T23PK2,R1          ;START OF THE PACKET
1613 034204 012721 100006                MOV    #100006,(R1).       ;WRITE SUBSYSTEM MEM. WITH ACK
1614 034210 012721 032722                MOV    #T23BF2,(R1).      ;ADDRESS OF DATA BLOCK
1615 034214 005021                CLR    (R1).               ;EXTENDED ADDRESS
1616 034216 012721 000006                MOV    #6,(R1).           ;SIZE OF DATA BLOCK IN BYTES
1617 034222 012701 032722                MOV    #T23BF2,R1         ;POINT TO DATA SEL AREA
1618 034226 005021                CLR    (R1).
1619 034230 005021                CLR    (R1).
1620 034232 005011                CLR    (R1).
1621 034234 000207                RTS    PC                    ;RETURN
1622 034236                T23RT3:
1623 034236                SAVREG                      ;SAVE THE REGISTERS
1624 034242 012701 032710                MOV    #T23PK3,R1          ;START OF THE PACKET
1625 034246 012721 100005                MOV    #100005,(R1).       ;WRITE TAPE. WITH ACK
1626 034252 005021                CLR    (R1).               ;ADDRESS OF DATA BLOCK
1627 034254 005021                CLR    (R1).               ;EXTENDED ADDRESS
1628 034256 005011                CLR    (R1).               ;SIZE OF DATA BLOCK
1629 034260 000207                RTS    PC                    ;RETURN
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639 034262                ;*
1640 034262                ;ROUTINE TO RETRY WRITE DATA IN CASE OF BAD TAPE FOR TEST
1641 034266 005037 032730                ;3.SUBTEST 2 & 3
1642 034272 032701 100000                ;
1643 034276 001452                ;INPUTS:      R1=TSSR
1644 034300 013702 032612                ;SUBROUTINE SETS UP T23WRT FOR RETRY
1645 034304 032702 000002                ;
1646 034310 001401                ;
1647 034312 000405                ;
1648 034314 032702 020000                T23CHK:
1649 034320 001002                SAVREG                      ;SAVE THE REGISTERS
1650 034322 000440                CLR    T23TMP              ;CLEAR LOCAL REGISTER
1651 034324 000207                BIT    #SC,R1              ;IS SC SET IN TSSR?
1652                BEQ    FATAL              ;NO, YOU GOT PROBLEMS!
1653 034326                MOV    T23BFR+10,R2        ;YES,GET XSTAT1
1654 034328                BIT    #X1.UNC,R2         ;IS UNC SET IN XSTAT1?
1655 034330                BEQ    1:                 ;NO, CHECK COR
1656 034332                BR    RETRY               ;YES,DO WRITE DATA RETRY
1657 034334                1:  BIT    #X1.COR,R2        ;IS COR SET IN XSTAT1 THEN?
1658 034336                BNE   RETRY               ;YES SO RETRY
1659 034338                BR    FATAL              ;NO, YOU GOT PROBLEMS
1660 034340                EXIT:  RTS    PC          ;RETURN
1661 034342                RETRY:
1662 034344                2:  MOV    #20,R3           ;STARTING RECORD SIZE
1663 034346                MOV    FREE,T23WB         ;STARTING WRITE BUFFER ADDRESS
1664 034348                MOV    #T23WRT,T23PK3    ;WRITE DATA RETRY COMMAND SETUP BY SUBROUTINE
1665 034350                MOV    #T23PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
1666 034352                MOV    R3,R0             ;SET PATTERN IN CORRECT REGISTER
1667 034354                JSR    PC,FILLMEM        ;FILL MEMORY WITH RECORD SIZE

```

```

1660 034360 010337 032716      MOV      R3,T23SZ      ;SET UP RECORD SIZE IN PACKET
1661 034364 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
1662 034370 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
1663 034374 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
1664 034400 012702 000200      MOV      @SSR,R2      ;SET UP EXPECTED
1665 034404 020102              CMP      R1,R2        ;ARE THEY EQUAL
1666 034406 001746              BEQ      EXIT         ;BR, IF OK
1667 034410 005237 032730      INC      T23TMP       ;TRY FIVE TIMES THEN EXIT
1668 034414 022737 000005 032730  CMP      @5,T23TMP    ;DONE FIVE YET?
1669 034422 001341              BNE      2#          ;NO GO AGAIN
1670 034424 005237 002212      FATAL:  INC      FATFLG ;BUMP COUNT
1674 034430 013702 032602      MOV      T23BFR,R2   ;LOW ORDER MSGBUF
1675 034434              ERRHRD  ERRNO,SCHERR,PKTMES ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERRRD
                                .WORD    332
                                .WORD    SCHERR
                                .WORD    PKTMES
1676 034444 004737 017272      JSR      PC,CKDROP    ;DROP THE UNIT
1677 034450              ENDTST
                                L10043:
                                TRAP      C#ETST
034450 104401

```

1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716

.SBTTL TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

;
; THIS TEST VERIFIES THAT THE READ FORWARD AND READ REVERSE
; COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN
; DATA BUFFER BOUNDARIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY
; SPACE IS AVAILIABLE), AND BYTE-SWAP CONTROL ARE USED. THIS TEST
; OF COURSE, FURTHER VERIFIES THE WRITE DATA COMMAND BY ACTUALLY
; READING AND VERIFYING WRITTEN DATA. ALSO TESTED ARE PROPER
; TERMINATIONS ON EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH
; LONG, RECORD LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA
; BUFFER ADDRESSES, ILLEGAL CODES IN THE MODE FIELD OF THE BASIC
; READ COMMAND, AND DATA BUFFERS IN NON-EXISTANT MEMORY. THE TEST
;
; THE TEST CONSISTS OF THE FOLLOWING 14 SUBTESTS
;
;
;
;

```

BGN7ST

T4::

```

MOV      @EPRT1,EPRTSW      ;SET UP PRIMARY ERROR MESSAGE
CLR      KTENABLE          ;TURN OFF KT11
JSR      PC,KTOFF          ;TURN KT11 OFF
MOV      @TST24ID,R0       ;ASCII MESSAGE TO IDENTIFY TEST
JSR      PC,TSTSETUP       ;DO INITIAL TEST SETUP
JSR      PC,MEMCK          ;CHECK FOR MEMORY
CLR      NXMFLG            ;SET FLAG
MOV      @5,LOOPCNT        ;PERFORM 5 ITERATIONS

```

TEST 4, SUBTEST 1







```

1813 035032 012703 044206            MOV    #T24RN,R3                            ; POINTER FOR COMMANDS
1814
1815                                    ; *****
1816                                    ;
1817                                    ; TAPE READ COMMAND IN PLACE
1818                                    ;
1819                                    ; *****
1820
1821 035036 011337 044170    65$:  MOV    (R3),T24PK3                        ; TAPE READ COMMAND IN PLACE
1822 035042 012704 044170            MOV    #T24PK3,R4                            ; R4 = POINTER TO PACKET
1823 035046 010465 000000            MOV    R4,TSDB(R5)                          ; ISSUE COMMAND
1824 035052 004737 016340            JSR    PC,WAITF                            ; WAIT FOR SSR TO SET
1825 035056 016501 000002            MOV    TSSR(R5),R1                          ; GET TSSR CONTENTS
1826 035062 012702 100306            MOV    #SSR!SC!OFL!BIT1!BIT2,R2            ; SET UP EXPECTED
1827 035066 020102            CMP    R1,R2                                ; ARE THEY EQUAL
1828 035070 001406            BEQ    80$                                  ; BR, IF OK ESP. FUNCTION REJECT
1829 035072 005237 002212            INC    FATFLG                               ; BUMP COUNT
1833 035076            ERRHRD    ERRNO,T24TM,PKTSSR               ; TSSR INCORRECT AFTER TAPE MOTION CMD
                                                                      TRAP    C$ERHRD
                                                                      .WORD    406
                                                                      .WORD    T24TM
                                                                      .WORD    PKTSSR
1834 035106            80$:  CKLOOP                                    ; LOOP IF SELECTED
                                                                      TRAP    C$CLP1
                                                                      .WORD    0
                                                                      .WORD    0
1835 035110 104406            TST    (R3)+                                ; BUMP TO NEXT COMMAND
1836 035112 005723            CMP    #177777,(R3)                        ; END OF THE COMMANDS YET
1837 035116 001401 177777            BEQ    90$                                  ; BR, IF DONE
1838 035120 000746            BR     65$                                  ; MORE COMMAND(S) TO GO
1839 035122            90$:  ENDSUB                                    ; >>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
1840 035122                                                                L10053:
                                                                      TRAP    C$ESUB
                                                                      .WORD    0
1841 035124 023727 002212 000017        CMP    FATFLG,#15.                                ; IS ERROR COUNT AT 25
1842 035132 103402            BLO    999$                                 ; BR, IF LESS THAN 25
1843 035134 004737 017272            JSR    PC,CKDROP                             ; TRY TO DROP THE UNIT
1844 035140            999$:
1845
1846                                    ; *
1847                                    ;
1848                                    ; TEST 4, SUBTEST 2
1849                                    ;
1850                                    ; VERIFIES THAT READ FORWARD COMMANDS WITH SWB=0
1851                                    ; OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN
1852                                    ; WRITTEN WITH A SERIES OF TEST RECORDS VARYING IN
1853                                    ; LENGTH AND DATA CONTENT. THE TAPE IS THEN REWOUND
1854                                    ; AGAIN AND THE RECORD READ SEQUENTIALLY AND RESULTS
1855                                    ; (STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON
1856                                    ; EACH READ FORWARD COMMAND IS SET TO THE LENGTH OF THE
1857                                    ; EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD
1858                                    ; OCCUR.
1859                                    ;
1860                                    ;
1861                                    ;
1862                                    ;
1863 035140            BGNSUB                                    ; >>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
                                                                      T4.2:
                                                                      TRAP    C$BSUB
                                                                      .WORD    0
                                                                      .WORD    0
                                                                      .WORD    0
1863 035140 104402            .WORD    104402
    
```

```

1864 035142 004737 046664          JSR    PC,T24RT3          ;SET UP OTHER COMMAND PACKET
1865 035146 004737 046530          JSR    PC,T24REST        ;SET COMMAND PACKET
1866 035152 004737 046622          JSR    PC,T24RT2        ;SET UP OTHER COMMAND PACKET
1867
1868 ;*****
1869 ;
1870 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
1871 ;
1872 ;*****
1873
1874 035156 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
1875 035162 103407                  BCS    20$               ;BR IF INIT WAS OK
1876 035164 005237 002212          INC    FATFLG            ;BUMP COUNT
1880 035170 010001                  MOV    RO,R1             ;CONTENTS OF TSSR REGISTER
1881 035172                  ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   407
                                .WORD   SFIERR
                                .WORD   SFIMSG
1882 035202                  20$:
1883 035202 013737 002172 044060    MOV    UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
1884 035210 012704 044040          MOV    #T24PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
1885
1886 ;*****
1887 ;
1888 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
1889 ;
1890 ;*****
1891
1892 035214 004737 010752          JSR    PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
1893 035220 103407                  BCS    24$               ;BR, IF COMMAND ISSUED OK
1894 035222 005237 002212          INC    FATFLG            ;BUMP COUNT
1898 035226 010001                  MOV    RO,R1             ;SAVE CONTENTS OF TSSR
1899 035230                  ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C$ERHRD
                                .WORD   408
                                .WORD   WRTMSG
                                .WORD   SFIMSG
1900 035240                  24$:  CKLOOP            ;LOOP IF SELECTED
                                TRAP    C$CLP1
1901 035240 104406
1902
1903 ;*****
1904 ;
1905 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
1906 ;
1907 ;*****
1908 035242 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
1909 035246 103407                  BCS    30$               ;BR, IF NO PROBLEM
1910 035250 010001                  MOV    RO,R1             ;SAVE TSSR
1911 035252 005237 002212          INC    FATFLG            ;BUMP COUNT
1915 035256                  ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   409
                                .WORD   T24RWN
                                .WORD   PKTSSR
1916 035266                  30$:  CKLOOP            ;LOOP IF SELECTED

```

```

035266 104406                                TRAP    C$CLP1
1917
1918 ;*****
1919 ;
1920 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
1921 ;
1922 ;*****
1923
1924 035270 013701 044070                    MOV     T24BFR+6,R1      ;PICK UP XSTO
1925 035274 010102                          MOV     R1,R2           ;SET UP EXPECTED
1926 035276 052702 000002                    BIS     #BIT1,R2       ;SET BOT BIT IN EXPECTED
1927 035302 020102                          CMP     R1,R2          ;DOES EXP = REC'D
1928 035304 001406                          BEQ     40$            ;BR, IF EQUAL (OK)
1929 035306 005237 002212                    INC     FATFLG         ;BUMP COUNT
1933 035312                                ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   410
                                .WORD   T24BOT
                                .WORD   EXPREC
1934 035322                                40$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
1935 035324 104406                          MOV     #256.,R3       ;RECORD SIZE
1936 035330 012703 000400                    MOV     FREE,T24RB     ;STARTING WRITE BUFFER ADDRESS
1937
1938 ;*****
1939 ;
1940 ;WRITE DATA,CVC=1,ACK COMMAND
1941 ;
1942 ;*****
1943
1944 035336 012737 140005 044170            MOV     #140005,T24PK3 ;WRITE DATA,CVC=1,ACK COMMAND
1945 035344 012704 044170                    MOV     #T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
1946 035350                                65$:   MOV     R3,R0       ;SET PATTERN IN CORRECT REGISTER
1947 035350 010300                          JSR     PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
1948 035352 004737 017512                    MOV     R3,T24SZ      ;SET UP RECORD SIZE IN PACKET
1949 035356 010337 044176                    MOV     R4,TSDB(R5)   ;ISSUE COMMAND
1950 035362 010465 000000                    JSR     PC,WAITF      ;WAIT FOR SSR TO SET
1951 035366 004737 016340                    MOV     TSSR(R5),R1  ;GET TSSR CONTENTS
1952 035372 016501 000002                    MOV     #SSR,R2      ;SET UP EXPECTED
1953 035376 012702 000200                    CMP     R1,R2        ;ARE THEY EQUAL
1954 035402 020102                          BEQ     75$          ;BR, IF OK
1955 035404 001406                          INC     FATFLG       ;BUMP COUNT
1956 035406 005237 002212                    ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP    C$ERHRD
                                .WORD   411
                                .WORD   WRERR
                                .WORD   PKTSSR
1961 035422                                75$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
1962 035424 104406                          TST     (R3)+         ;BUMP RECORD SIZE
1963 035426 022703 000414                    CMP     #268.,R3     ;END OF RECORD YET
1964 035432 001346                          BNE     65$          ;BR, IF MORE RECORDS TO WRITE
1965 035434                                80$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
1966 035436                                120$:
1967

```

```

1968 ;*****
1969 ;
1970 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
1971 ;
1972 ;*****
1973
1974 035436 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
1975 035442 004737 016426 JSR PC,CHKTSSR ;SEE HOW TSSR IS
1976 035446 103407 BCS 130$ ;BR, IF NO PROBLEM
1977 035450 010001 MOV R0,R1 ;SAVE TSSR
1978 035452 005237 002212 INC FATFLG ;BUMP COUNT
1982 035456 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      035456 104456 TRAP C$ERHRD
      035460 000634 .WORD 412
      035462 045326 .WORD T24RWN
      035464 012136 .WORD PKTSSR
1983 035466 130$: CKLOOP ;LOOP IF SELECTED
      035466 104406 TRAP C$CLP1
1984
1985 ;*****
1986 ;
1987 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
1988 ;
1989 ;*****
1990
1991 035470 013701 044070 MOV T24BFR+6,R1 ;PICK UP XSTO
1992 035474 010102 MOV R1,R2 ;SET UP EXPECTED
1993 035476 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
1994 035502 020102 CMP R1,R2 ;DOES EXP = REC'D
1995 035504 001406 BEQ 140$ ;BR, IF EQUAL (OK)
1996 035506 005237 002212 INC FATFLG ;BUMP COUNT
2000 035512 ERRHRD ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      035512 104456 TRAP C$ERHRD
      035514 000635 .WORD 413
      035516 045043 .WORD T24BOT
      035520 015564 .WORD EXPREC
2001 035522 140$: CKLOOP ;LOOP IF SELECTED
      035522 104406 TRAP C$CLP1
2002 035524 012703 000400 MOV #256.,R3 ;RECORD SIZE
2003 035530 013737 003114 044172 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
2004
2005 ;*****
2006 ;
2007 ;READ DATA,CVC=1,ACK COMMAND
2008 ;
2009 ;*****
2010
2011 035536 012737 140001 044170 MOV #140001,T24PK3 ;READ DATA,CVC=1,ACK COMMAND
2012 035544 012704 044170 165$: MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
2013 035550 010337 044176 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
2014 035554 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
2015 035560 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
2016 035564 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
2017 035570 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
2018 035574 020102 CMP R1,R2 ;ARE THEY EQUAL
2019 035576 001406 BEQ 170$ ;BR, IF OK
2020 035600 005237 002212 INC FATFLG ;BUMP COUNT

```

```

2024 035604              ERRHRD  ERRNO,RDERR,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
    035604 104456              TRAP            C$ERHRD
    035606 000636              .WORD         414
    035610 005204              .WORD         RDERR
    035612 012136              .WORD         PKTSSR
2025 035614              170$:  CKLOOP                    ;LOOP IF SELECTED
    035614 104406              TRAP            C$CLP1
2026 035616 013702 003114     MOV          FREE,R2          ;GET BUFFER ADDRESS
2027 035622 010304              MOV          R3,R4            ;CURRENT RECORD SIZE
2028 035624 162704 000400     SUB          #256.,R4         ;FIRST LOCATION IN BUFFER
2029 035630 060204              173$:  ADD          R2,R4      ;GET LOCATION IN BUFFER (ADDRESS)
2030 035532 021403              CMP          (R4),R3         ;CHECK DATA READ (R3=DATA ALSO)
2031 035634 001410              BEQ          180$           ;BR, IF ALL IS WELL
2032 035636 011401              MOV          (R4),R1         ;RECD DATA
2033 035640 010302              MOV          R3,R2          ;EXPECTED DATA
2034 035642 005237 002212     INC          FATFLG           ;BUMP COUNT
2038 035646              ERRHRD  ERRNO,T24DTA,EXPREC   ;DATA READ NOT = WRITTEN
    035646 104456              TRAP            C$ERHRD
    035650 000637              .WORD         415
    035652 045110              .WORD         T24DTA
    035654 015564              .WORD         EXPREC
2039 035656              180$:  CKLOOP                    ;LOOP IF SELECTED
    035656 104406              TRAP            C$CLP1
2040 035660 005724              TST          (R4).           ;BUMP TO NEXT LOCATION
2041 035662 160204              SUB          R2,R4            ;GET BACK TO CORRECT SIZE
2042 035664 020403              CMP          R4,R3            ;END OF RECORD YET
2043 035666 001360              BNE          173$           ;BR, IF NOT AT END OF RECORD
2044 035670 005723              TST          (R3).           ;BUMP RECORD SIZE
2045 035672 022703 000412     CMP          #266.,R3        ;END OF RECORD YET
2046 035676 001322              BNE          165$           ;BR, IF MORE RECORDS TO WRITE
2047 035700              190$:  CKLOOP                    ;LOOP IF SELECTED
    035700 104406              TRAP            C$CLP1
2048 035702              ENDSUB                       ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
    035702              L10054:                       TRAP            C$ESUB
    035702 104403              002212  000017              CMP          FATFLG,#15.     ;IS ERROR COUNT AT 25
2049 035704 023727 002212 000017  BLO          999$           ;BR, IF LESS THAN 25
2050 035712 103402              JSR          PC,CKDROP       ;TRY TO DROP THE UNIT
2051 035714 004737 017272          999$:
2052 035720
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066 035720              BGNSUB                       ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
    035720              T4.3:                       TRAP            C$BSUB
2067 035722 004737 046664          JSR          PC,T24RT3       ;SET UP OTHER COMMAND PACKET
2068 035726 004737 046530          JSR          PC,T24REST     ;SET COMMAND PACKET
    
```

```

2069 035732 004737 046622          JSR    PC,T24RT2          ;SET UP OTHER COMMAND PACKET
2070
2071          ;*****
2072          ;
2073          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
2074          ;
2075          ;*****
2076
2077 035736 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
2078 035742 103407          BCS    20$              ;BR IF INIT WAS OK
2079 035744 005237 002212          INC    FATFLG          ;BUMP COUNT
2083 035750 010001          MOV    R0,R1          ;CONTENTS OF TSSR REGISTER
2084 035752          ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                104455          TRAP    C$ERDF
                035754 000640          .WORD  416
                035756 003650          .WORD  SFIERR
                035760 012124          .WORD  SFIMSG
2085 035762          20$:
2086 035762 013737 002172 044060  MOV    UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
2087 035770 012704 044040          MOV    #T24PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
2088
2089          ;*****
2090          ;
2091          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2092          ;
2093          ;*****
2094
2095 035774 004737 010752          JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
2096 036000 103407          BCS    24$              ;BR, IF COMMAND ISSUED OK
2097 036002 005237 002212          INC    FATFLG          ;BUMP COUNT
2101 036006 010001          MOV    R0,R1          ;SAVE CONTENTS OF TSSR
2102 036010          ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                104456          TRAP    C$ERHRD
                036012 000641          .WORD  417
                036014 005054          .WORD  WRTMSG
                036016 012124          .WORD  SFIMSG
2103 036020          24$:  CKLOOP          ;LOOP IF SELECTED
                036020 104406          TRAP    C$CLP1
2104
2105          ;*****
2106          ;
2107          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2108          ;
2109          ;*****
2110
2111 036022 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
2112 036026 103407          BCS    30$              ;BR, IF NO PROBLEM
2113 036030 010001          MOV    R0,R1          ;SAVE TSSR
2114 036032 005237 002212          INC    FATFLG          ;BUMP COUNT
2118 036036          ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                104456          TRAP    C$ERHRD
                036040 000642          .WORD  418
                036042 045326          .WORD  T24RWN
                036044 012136          .WORD  PKTSSR
2119 036046          30$:  CKLOOP          ;LOOP IF SELECTED
                036046 104406          TRAP    C$CLP1
2120

```

```

2121 ;*****
2122 ;
2123 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
2124 ;
2125 ;*****
2126
2127 036050 013701 044070      MOV      T24BFR+6,R1      ;PICK UP XSTO
2128 036054 010102           MOV      R1,R2           ;SET UP EXPECTED
2129 036056 052702 000002     BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
2130 036062 020102           CMP      R1,R2           ;DOES EXP = REC'D
2131 036064 001406           BEQ      40$             ;BR, IF EQUAL (OK)
2132 036066 005237 002212     INC      FATFLG          ;BUMP COUNT
2136 036072           ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    419
                                .WORD    T24BOT
                                .WORD    EXPREC
2137 036102           40$:  CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
2138 036104 012703 000400     MOV      @256.,R3        ;RECORD SIZE
2139 036110 013737 003114 044172  MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
2140
2141 ;*****
2142 ;
2143 ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
2144 ;
2145 ;*****
2146
2147 036116 012737 150005 044170  MOV      @150005,T24PK3  ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
2148 036124 012704 044170     MOV      @T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
2149 036130           65$:  MOV      R3,R0          ;SET PATTERN IN CORRECT REGISTER
2150 036130 010300           JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
2151 036132 004737 017512     MOV      R3,T24SZ        ;SET UP RECORD SIZE IN PACKET
2152 036136 010337 044176     MOV      R4,TSDB(25)     ;ISSUE COMMAND
2153 036142 010465 000000     JSR      PC,WAITF        ;WAIT FOR SSR TO SET
2154 036146 004737 016340     MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
2155 036152 016501 000002     MOV      @SSR,R2        ;SET UP EXPECTED
2156 036156 012702 000200     CMP      R1,R2          ;ARE THEY EQUAL
2157 036162 020102           BEQ      75$             ;BR, IF OK
2158 036164 001406           INC      FATFLG          ;BUMP COUNT
2159 036166 005237 002212     ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
2163 036172           TRAP      C$ERHRD
                                .WORD    420
                                .WORD    WRERR
                                .WORD    PKTSSR
2164 036202           75$:  CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
2165 036204 005723           TST      (R3)+           ;BUMP RECORD SIZE
2166 036206 022703 000414     CMP      @268.,R3        ;END OF RECORD YET
2167 036212 001346           BNE      65$             ;BR, IF MORE RECORDS TO WRITE
2168 036214           80$:  CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
2169 036216           120$:
2170
2171 ;*****
2172 ;

```

```

2173      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2174      ;
2175      ;*****
2176
2177 036216 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
2178 036222 103407      BCS      130$          ;BR, IF NO PROBLEM
2179 036224 010001      MOV      R0,R1        ;SAVE TSSR
2180 036226 005237 002212      INC      FATFLG      ;BUMP COUNT
2184 036232      ERRHRD  ERRNO,T24RWN,EXPREC ;REWIND NOT ACCEPTED
          036232 104456      TRAP      C$ERHRD
          036234 000645      .WORD    421
          036236 045326      .WORD    T24RWN
          036240 015564      .WORD    EXPREC
2185 036242      130$:  CKLOOP      ;LOOP IF SELECTED      TRAP      C$CLP1
          036242 104406
2186
2187      ;*****
2188      ;
2189      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
2190      ;
2191      ;*****
2192
2193 036244 013701 044070      MOV      T24BFR+6,R1  ;PICK UP XSTO
2194 036250 010102      MOV      R1,R2        ;SET UP EXPECTED
2195 036252 052702 000002      BIS      @BIT1,R2    ;SET BOT BIT IN EXPECTED
2196 036256 020102      CMP      R1,R2        ;DOES EXP = REC'D
2197 036260 001406      BEQ      140$        ;BR, IF EQUAL (OK)
2198 036262 005237 002212      INC      FATFLG      ;BUMP COUNT
2202 036266      ERRHRD  ERRNO,T24BOT,FXPREC ;TAPE NOT AT BOT AFTER REWIND
          036266 104456      TRAP      C$ERHRD
          036270 000646      .WORD    422
          036272 045043      .WORD    T24BOT
          036274 015564      .WORD    EXPREC
2203 036276      140$:  CKLOOP      ;LOOP IF SELECTED      TRAP      C$CLP1
          036276 104406
2204 036300 012703 000400      MOV      @256.,R3    ;RECORD SIZE
2205 036304 013737 003114 044172  MUV     FREE,T24RB   ;STARTING READ BUFFER ADDRESS
2206
2207      ;*****
2208      ;
2209      ;READ DATA,IE,ACK,SWB COMMAND
2210      ;
2211      ;*****
2212
2213 036312 012737 110001 044170 165$:  MOV      @110001,T24PK3 ;READ DATA,IE,ACK,SWB COMMAND
2214 036320 012704 044170      MOV      @T24PK3,R4  ;SET UP R4 WITH PACKET ADDRESS
2215 036324 010337 044176      MOV      R3,T24SZ    ;SET UP RECORD SIZE IN PACKET
2216 036330 010465 000000      MOV      R4,TSD8(R5) ;ISSUE COMMAND
2217 036334 004737 016340      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
2218 036340 016501 000002      MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
2219 036344 012702 000200      MOV      @SSR,R2    ;SET UP EXPECTED
2220 036350 020102      CMP      R1,R2        ;ARE THEY EQUAL
2221 036352 001406      BEQ      170$        ;BR, IF OK
2222 036354 005237 002212      INC      FATFLG      ;BUMP COUNT
2226 036360      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          036360 104456      TRAP      C$ERHRD
          036362 000647      .WORD    423

```





```

2274
2275      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
2276      ;
2277      ;.....
2278
2279 036512 004737 016064      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
2280 036516 103407      BCS    20$             ;BR IF INIT WAS OK
2281 036520 005237 002212      INC    FATFLG         ;BUMP COUNT
2285 036524 010001      MOV    RO,R1          ;CONTENTS OF TSSR REGISTER
2286 036526      ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                104455      TRAP    C$ERDF
                036530 000651      .WORD  425
                036532 003650      .WORD  SFIERR
                036534 012124      .WORD  SFIMSG
2287 036536      20$:      MOV    UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
2288 036536 013737 002172 044060  MOV    @T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
2289 036544 012704 044040
2290
2291      ;.....
2292      ;
2293      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2294      ;
2295      ;.....
2296
2297 036550 004737 010752      JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
2298 036554 103407      BCS    24$             ;BR, IF COMMAND ISSUED OK
2299 036556 005237 002212      INC    FATFLG         ;BUMP COUNT
2303 036562 010001      MOV    RO,R1          ;SAVE CONTENTS OF TSSR
2304 036564      ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                104456      TRAP    C$ERHRD
                036566 000652      .WORD  426
                036570 005054      .WORD  WRTMSG
                036572 012124      .WORD  SFIMSG
2305 036574      24$:      CKLOOP      ;LOOP IF SELECTED
                036574 104406      TRAP    C$CLP1
2306
2307      ;.....
2308      ;
2309      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2310      ;
2311      ;.....
2312
2313 036576 004737 011104      JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
2314 036602 103407      BCS    30$             ;BR, IF NO PROBLEM
2315 036604 010001      MOV    RO,R1          ;SAVE TSSR
2316 036606 005237 002212      INC    FATFLG         ;BUMP COUNT
2320 036612      ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                104456      TRAP    C$ERHRD
                036614 000653      .WORD  427
                036616 045326      .WORD  T24RWN
                036620 012136      .WORD  PKTSSR
2321 036622      30$:      CKLOOP      ;LOOP IF SELECTED
                036622 104406      TRAP    C$CLP1
2322
2323      ;.....
2324      ;
2325      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)

```

```

2326 ;
2327 ;.....
2328
2329 036624 013701 044070      MOV      T24BFR+6,R1      ;PICK UP XSTO
2330 036630 010102      MOV      R1,R2           ;SET UP EXPECTED
2331 036632 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
2332 036636 020102      CMP      R1,R2           ;DOES EXP = REC'D
2333 036640 001406      BEQ      40$            ;BR, IF EQUAL (OK)
2334 036642 005237 002212      INC      FATFLG          ;BUMP COUNT
2338 036646      ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    428
                                .WORD    T24BOT
                                .WORD    EXPREC
2339 036656      40$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
2340 036660 012703 001000      MOV      @512.,R3        ;RECORD SIZE
2341 036664 013737 003114 044172      MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
2342 ;.....
2343 ;
2344 ;WRITE DATA,ACK,CVC=1 COMMAND
2345 ;
2346 ;.....
2347 ;
2348
2349 036672 012737 140005 044170      MOV      @140005,T24PK3  ;WRITE DATA,ACK,CVC=1 COMMAND
2350 036700 012704 044170      MOV      @T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
2351 036704      65$:
2352 036704 010337 044176      MOV      R3,T24SZ        ;SET UP RECORD SIZE IN PACKET
2353 036710 010465 000000      MOV      R4,TSDB(R5)     ;ISSUE COMMAND
2354 036714 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
2355 036720 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
2356 036724 012702 000200      MOV      @SSR,R2         ;SET UP EXPECTED
2357 036730 020102      CMP      R1,R2           ;ARE THEY EQUAL
2358 036732 001406      BEQ      75$            ;BR, IF OK
2359 036734 005237 002212      INC      FATFLG          ;BUMP COUNT
2363 036740      ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    429
                                .WORD    WRTErr
                                .WORD    PKTSSR
2364 036750      75$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
2365 036752      120$:
2366 ;.....
2367 ;
2368 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2369 ;
2370 ;.....
2371 ;
2372
2373 036752 004737 011104      JSR      PC,REWIND       ;CALL TAPE REWIND COMMAND
2374 036756 103407      BCS     130$            ;BR, IF NO PROBLEM
2375 036760 010001      MOV     RO,R1           ;SAVE TSSR
2376 036762 005237 002212      INC     FATFLG          ;BUMP COUNT
2380 036766      ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    429
                                .WORD    T24RWN
                                .WORD    PKTSSR
    
```

```

036770 000656 .WORD 430
036772 045326 .WORD T24RWN
036774 012136 .WORD PKTSSR
2381 036776 104406 130$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
036776 104406
2382
2383 ;*****
2384 ;
2385 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
2386 ;
2387 ;*****
2388
2389 037000 013701 044070 MOV T24BFR+6,R1 ;PICK UP XSTO
2390 037004 010102 MOV R1,R2 ;SET UP EXPECTED
2391 037006 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
2392 037012 020102 CMP R1,R2 ;DOES EXP = REC'D
2393 037014 001406 BEQ 140$ ;BR, IF EQUAL (OK)
2394 037016 005237 002212 INC FATFLG ;BUMP COUNT
2398 037022 ERRHRD ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
037022 104456 TRAP C$ERHRD
037024 000657 .WORD 431
037026 045043 .WORD T24BOT
037030 015564 .WORD EXPREC
2399 037032 140$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
037032 104406
2400 037034 012703 000400 MOV #256.,R3 ;RECORD SIZE
2401 037040 013737 003114 044172 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
2402
2403 ;*****
2404 ;
2405 ;READ DATA,ACK,CVC=1 COMMAND
2406 ;
2407 ;*****
2408
2409 037046 012737 140001 044170 MOV #140001,T24PK3 ;READ DATA,ACK,CVC=1 COMMAND
2410 037054 012704 044170 165$: MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
2411 037060 010337 044176 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
2412 037064 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
2413 037070 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
2414 037074 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
2415 037100 012702 100204 MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
2416 037104 020102 CMP R1,R2 ;ARE THEY EQUAL
2417 037106 001406 BEQ 170$ ;BR, IF OK
2418 037110 005237 002212 INC FATFLG ;BUMP COUNT
2422 037114 ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
037114 104456 TRAP C$ERHRD
037116 000660 .WORD 432
037120 046374 .WORD T24TRL
037122 012136 .WORD PKTSSR
2423 037124 170$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
037124 104406
2424
2425 ;*****
2426 ;
2427 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
2428 ;
2429 ;*****

```



```

037232 000662 .WORD 434
037234 003650 .WORD SFIERR
037236 012124 .WORD SFIMSG
2484 037240 20$: MOV UNITN,T24DSW ;SET DRIVE NUMBER IN PACKET
2485 037240 013737 002172 044060 MOV #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
2486 037246 012704 044040
2487
2488 ;*****
2489 ;
2490 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2491 ;
2492 ;*****
2493
2494 037252 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
2495 037256 103407 BCS 24$ ;BR, IF COMMAND ISSUED OK
2496 037260 005237 002212 INC FATFLG ;BUMP COUNT
2500 037264 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
2501 037266 ERRHRD ERRNJ,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
037266 104456 TRAP C$ERHRD
037270 000663 .WORD 435
037272 005054 .WORD WRTMSG
037274 012124 .WORD SFIMSG
2502 037276 24$: CKLOOP ;LOOP IF SELECTED
037276 104406 TRAP C$CLP1
2503
2504 ;*****
2505 ;
2506 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2507 ;
2508 ;*****
2509
2510 037300 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
2511 037304 103407 BCS 30$ ;BR, IF NO PROBLEM
2512 037306 010001 MOV RO,R1 ;SAVE TSSR
2513 037310 005237 002212 INC FATFLG ;BUMP COUNT
2517 037314 ERRHRD ERRNJ,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
037314 104456 TRAP C$ERHRD
037316 000664 .WORD 436
037320 045326 .WORD T24RWN
037322 012136 .WORD PKTSSR
2518 037324 30$: CKLOOP ;LOOP IF SELECTED
037324 104406 TRAP C$CLP1
2519 037326 012703 000400 MOV #256.,R3 ;RECORD SIZE
2520 037332 013737 003114 044172 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS
2521
2522 ;*****
2523 ;
2524 ;WRITE DATA,ACK,CVC=1 COMMAND
2525 ;
2526 ;*****
2527
2528 037340 012737 140005 044170 MOV #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
2529 037346 012704 044170 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
2530 037352 63$:
2531 037352 010337 044176 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
2532 037356 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
2533 037362 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
    
```

```

2534 037366 016501 000002          MOV    TSSR(R5),R1          ;GET TSSR CONTENTS
2535 037372 012702 000200          MOV    #SSR,R2            ;SET UP EXPECTED
2536 037376 020102                  CMP    R1,R2              ;ARE THEY EQUAL
2537 037400 001406                  BEQ    75$                 ;BR, IF OK
2538 037402 005237 002212          INC    FATFLG              ;BUMP COUNT
2542 037406                          ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP    C$ERHRD
                                .WORD   437
                                .WORD   WRTErr
                                .WORD   PKTSSR
                                TRAP    C$CLP1
    037406 104456
    037410 000665
    037412 005111
    037414 012136
2543 037416 75$: CKLOOP                ;LOOP IF SELECTED
    037416 104406
2544 037420                          TRAP    C$CLP1
2545
2546 ;*****
2547 ;
2548 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2549 ;
2550 ;*****
2551
2552 037420 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
2553 037424 103407                  BCS    130$               ;BR, IF NO PROBLEM
2554 037426 010001                  MOV    R0,R1              ;SAVE TSSR
2555 037430 005237 002212          INC    FATFLG              ;BUMP COUNT
2559 037434                          ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   438
                                .WORD   T24RWN
                                .WORD   PKTSSR
    037434 104456
    037436 000666
    037440 045326
    037442 012136
2560 037444 130$: CKLOOP                ;LOOP IF SELECTED
    037444 104406                          TRAP    C$CLP1
2561 037446 012703 001000          MOV    #512.,R3           ;RECORD SIZE
2562 037452 013737 003114 044172  MOV    FREE,T24RB         ;STARTING READ BUFFER ADDRESS
2563
2564 ;*****
2565 ;
2566 ;READ DATA,ACK,CVC=1 COMMAND
2567 ;
2568 ;*****
2569
2570 037460 012737 140001 044170 165$: MOV    #140001,T24PK3    ;READ DATA,ACK,CVC=1 COMMAND
2571 037466 012704 044170          MOV    #T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
2572 037472 010337 044176          MOV    R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
2573 037476 010465 000000          MOV    R4,TSDB(R5)       ;ISSUE COMMAND
2574 037502 004737 016340          JSR    PC,WAITF           ;WAIT FOR SSR TO SET
2575 037506 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
2576 037512 012702 100204          MOV    #SSR!SC!BIT2,R2   ;SET UP EXPECTED
2577 037516 020102                  CMP    R1,R2              ;ARE THEY EQUAL
2578 037520 001406                  BEQ    170$               ;BR, IF OK
2579 037522 005237 002212          INC    FATFLG              ;BUMP COUNT
2583 037526                          ERRHRD  ERRNO,T24TRL,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C$ERHRD
                                .WORD   439
                                .WORD   T24TRL
                                .WORD   EXPREC
    037526 104456
    037530 000667
    037532 046374
    037534 015564
2584 037536 170$: CKLOOP                ;LOOP IF SELECTED
    037536 104406                          TRAP    C$CLP1

```





```

                                T4.6:
037642                                TRAP      C$BSUB
037642 104402
2637 037644 004737 046664          JSR      PC,T24RT3          ;SET UP OTHER COMMAND PACKET
2638 037650 004737 046530          JSR      PC,T24REST        ;SET COMMAND PACKET
2639 037654 004737 046622          JSR      PC,T24RT2        ;SET UP OTHER COMMAND PACKET
2640
2641          ;*****
2642          ;
2643          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
2644          ;
2645          ;*****
2646
2647 037660 004737 016064          JSR      PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
2648 037664 103407                    BCS      20$                ;BR IF INIT WAS OK
2649 037666 005237 002212          INC      FATFLG            ;BUMP COUNT
2653 037672 010001                    MOV      RO,R1              ;CONTENTS OF TSSR REGISTER
2654 037674                    ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP      C$ERDF
                                .WORD    442
                                .WORD    SFIERR
                                .WORD    SFIMSG
037674 104455
037676 000672
037700 003650
037702 012124
2655 037704                    20$:  MOV      UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
2656 037704 013737 002172 044060  MOV      @T24PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
2657 037712 012704 044040
2658
2659          ;*****
2660          ;
2661          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2662          ;
2663          ;*****
2664
2665 037716 004737 010752          JSR      PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
2666 037722 103407                    BCS      24$                ;BR, IF COMMAND ISSUED OK
2667 037724 005237 002212          INC      FATFLG            ;BUMP COUNT
2671 037730 010001                    MOV      RO,R1              ;SAVE CONTENTS OF TSSR
2672 037732          ERRHRD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP      C$ERHRD
                                .WORD    443
                                .WORD    WRTMSG
                                .WORD    SFIMSG
037732 104456
037734 000673
037736 005054
037740 012124
2673 037742          24$:  CKLOOP                ;LOOP IF SELECTED
037742 104406                                TRAP      C$CLP1
2674
2675          ;*****
2676          ;
2677          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2678          ;
2679          ;*****
2680
2681 037744 004737 011104          JSR      PC,REWIND         ;CALL TAPE REWIND COMMAND
2682 037750 103407                    BCS      30$                ;BR, IF NO PROBLEM
2683 037752 010001                    MOV      RO,R1              ;SAVE TSSR
2684 037754 005237 002212          INC      FATFLG            ;BUMP COUNT
2688 037760          ERRHRD   ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    444
                                .WORD    T24RWN
037760 104456
037762 000674
037764 045326
    
```

```

2689 037766 012136          30$:  CKLOOP          ;LOOP IF SELECTED          .WORD  PKTSSR
037770          104406          ;                               TRAP    C$CLP1
2690 037772 012703 000400      MOV    #256.,R3          ;RECORD SIZE
2691 037776 013737 003114 044172  MOV    FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
2692
2693 ;*****
2694 ;
2695 ;WRITE DATA,ACK,CVC=1 COMMAND
2696 ;
2697 ;*****
2698
2699 040004 012737 140005 044170      MOV    #140005,T24PK3  ;WRITE DATA,ACK,CVC=1 COMMAND
2700 040012 012704 044170      MOV    #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
2701 040016          65$:
2702 040016 010300      MOV    R3,R0          ;SET PATTERN IN CORRECT REGISTER
2703 040020 004737 017512      JSR    PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
2704 040024 010337 044176      MOV    R3,T24SZ        ;SET UP RECORD SIZE IN PACKET
2705 040030 010465 000000      MOV    R4,TSDB(R5)     ;ISSUE COMMAND
2706 040034 004737 016340      JSR    PC,WAITF        ;WAIT FOR SSR TO SET
2707 040040 016501 000002      MOV    TSSR(R5),R1     ;GET TSSR CONTENTS
2708 040044 012702 000200      MOV    #SSR,R2        ;SET UP EXPECTED
2709 040050 020102      CMP    R1,R2          ;ARE THEY EQUAL
2710 040052 001406      BEQ    75$            ;BR, IF OK
2711 040054 005237 002212      INC    FATFLG         ;BUMP COUNT
2715 040060      ERRHRD  ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
040060 104456          TRAP    C$ERHRD
040062 000675          .WORD  445
040064 005111          .WORD  WRTERR
040066 012136          .WORD  PKTSSR
2716 040070          75$:  CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
040070 104406          ;
2717 040072 005723      TST    (R3)+          ;BUMP RECORD SIZE
2718 040074 022703 000414      CMP    #268.,R3      ;END OF RECORD YET
2719 040100 001346      BNE    65$            ;BR, IF MORE RECORDS TO WRITE
2720 040102          80$:  CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
040102 104406          ;
2721 040104 005743      TST    -(R3)         ;SET BACK TO 512.
2722 040106 013737 003114 044172  MOV    FREE,T24RB      ;STARTING READ BUFFER ADDRESS
2723
2724 ;*****
2725 ;
2726 ;READ REVERSE DATA,ACK COMMAND
2727 ;
2728 ;*****
2729
2730 040114 012737 100401 044170      MOV    #100401,T24PK3  ;READ REVERSE DATA,ACK COMMAND
2731 040122 012704 044170      MOV    #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
2732 040126 010337 044176      MOV    R3,T24SZ        ;SET UP RECORD SIZE IN PACKET
2733 040132 010465 000000      MOV    R4,TSDB(R5)     ;ISSUE COMMAND
2734 040136 004737 016340      JSR    PC,WAITF        ;WAIT FOR SSR TO SET
2735 040142 016501 000002      MOV    TSSR(R5),R1     ;GET TSSR CONTENTS
2736 040146 012702 000200      MOV    #SSR,R2        ;SET UP EXPECTED
2737 040152 020102      CMP    R1,R2          ;ARE THEY EQUAL
2738 040154 001406      BEQ    170$           ;BR, IF OK
2739 040156 005237 002212      INC    FATFLG         ;BUMP COUNT
2743 040162      ERRHRD  ERRNO,T24WDC,PKTSSR ;TSSR INCORRECT AFTER READ DATA

```

```
040162 104456 TRAP C$ERHRD
040164 000676 .WORD 446
040166 045656 .WORD T24WDC
040170 012136 .WORD PKTSSR
2744 040172 170$: CKLOOP ;LOOP IF SELECTED
040172 104406 TRAP C$CLP1
2745 040174 013702 003114 MOV FREE,R2 ;GET BUFFER ADDRESS
2746 040200 010304 MOV R3,R4 ;CURRENT RECORD SIZE
2747 040202 162704 000400 SUB #256.,R4 ;FIRST LOCATION IN BUFFER
2748 040206 060204 173$: ADD R2,R4 ;SET POINTER TO FRAME (WORD)
2749 040210 021403 CMP (R4),R3 ;CHECK DATA READ (R3=DATA ALSO)
2750 040212 001410 BEQ 180$ ;BR, IF ALL IS WELL
2751 040214 011401 MOV (R4),R1 ;RECD DATA
2752 040216 010302 MOV R3,R2 ;EXPECTED DATA
2753 040220 005237 002212 INC FATFLG ;BUMP COUNT
2757 040224 ERRHRD ERRNO,T24DTA,EXPREC ;DATA READ NOT = WRITTEN
040224 104456 TRAP C$ERHRD
040226 000677 .WORD 447
040230 045110 .WORD T24DTA
040232 015564 .WORD EXPREC
2758 040234 180$: CKLOOP ;LOOP IF SELECTED
040234 104406 TRAP C$CLP1
2759 040236 005724 TST (R4)+ ;BUMP TO NEXT LOCATION
2760 040240 160204 SUB R2,R4 ;GET RID OF BASE ADDRESS
2761 040242 020403 CMP R4,R3 ;END OF RECORD YET
2762 040244 001360 BNE 173$ ;BR, IF NOT AT END OF RECORD
2763 040246 005743 TST -(R3) ;BUMP RECORD SIZE
2764 040250 022703 000400 CMP #256.,R3 ;END OF RECORD YET
2765 040254 001322 BNE 165$ ;BR, IF MORE RECORDS TO WRITE
2766 040256 190$: CKLOOP ;LOOP IF SELECTED
040256 104406 TRAP C$CLP1
2767 040260 ENDSUB ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
040260 104403 L10060: TRAP C$CLP1
2768 040262 023727 002212 000017 CMP FATFLG,#15. ;IS ERROR COUNT AT 25
2769 040270 103402 BLO 999$ ;BR, IF LESS THAN 25
2770 040272 004737 017272 JSR PC,CKDROP ;TRY TO DROP THE UNIT
2771 040276 999$:
2772
2773 ;+
2774 ;
2775 ;TEST 4, SUBTEST 7
2776 ;
2777 ;VERIFIES THAT READ DATA COMMANDS WITH CVC=1 AND THE
2778 ;SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST
2779 ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
2780 ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
2781 ;
2782 ;
2783 ;
2784 ;-
2785 040276 BGNSUB ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
040276 T4.7: TRAP C$SUB
040276 104402 TRAP C$SUB
2786 040300 004737 046664 JSR PC,T24RT3 ;SET UP OTHER COMMAND PACKET
2787 040304 004737 046530 JSR PC,T24REST ;SET COMMAND PACKET
2788 040310 004737 046622 JSR PC,T24RT2 ;SET UP OTHER COMMAND PACKET
```

```

2789
2790 ;*****
2791 ;
2792 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
2793 ;
2794 ;*****
2795
2796 040314 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
2797 040320 103407 BCS 20$ ;BR IF INIT WAS OK
2798 040322 005237 002212 INC FATFLG ;BUMP COUNT
2802 040326 010001 MOV RO,R1 ;CONTENTS OF TSSR REGISTER
2803 040330 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      040330 104455 TRAP C$ERDF
      040332 000700 .WORD 448
      040334 003650 .WORD SFIERR
      040336 012124 .WORD SFIMSG
2804 040340
2805 040340 013737 002172 044060 20$: MOV UNITN,T24DSW ;SET DRIVE NUMBER IN PACKET
2806 040346 012704 044040 MOV #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
2807
2808 ;*****
2809 ;
2810 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2811 ;
2812 ;*****
2813
2814 040352 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
2815 040356 103407 BCS 24$ ;BR, IF COMMAND ISSUED OK
2816 040360 005237 002212 INC FATFLG ;BUMP COUNT
2820 040364 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
2821 040366 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      040366 104456 TRAP C$ERHRD
      040370 000701 .WORD 449
      040372 005054 .WORD WRTMSG
      040374 012124 .WORD SFIMSG
2822 040376 24$: CKLOOP ;LOOP IF SELECTED
      040376 104406 TRAP C$CLP1
2823
2824 ;*****
2825 ;
2826 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2827 ;
2828 ;*****
2829
2830 040400 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
2831 040404 103407 BCS 30$ ;BR, IF NO PROBLEM
2832 040406 010001 MOV RO,R1 ;SAVE TSSR
2833 040410 005237 002212 INC FATFLG ;BUMP COUNT
2837 040414 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      040414 104456 TRAP C$ERHRD
      040416 000702 .WORD 450
      040420 045326 .WORD T24RWN
      040422 012136 .WORD PKTSSR
2838 040424 30$: CKLOOP ;LOOP IF SELECTED
      040424 104406 TRAP C$CLP1
2839 040426 012703 000400 MOV #256.,R3 ;RECORD SIZE
2840 040432 013737 003114 044172 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS

```

```

2841
2842 ;*****
2843 ;
2844 ;WRITE DATA,ACK,CVC-1,SWB COMMAND
2845 ;
2846 ;*****
2847
2848 040440 012737 150005 044170      MOV      #150005,T24PK3      ;WRITE DATA,ACK,CVC-1,SWB COMMAND
2849 040446 012704 044170      MOV      #T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
2850 040452      65$:
2851 040452 010300      MOV      R3,R0             ;SET PATTERN IN CORRECT REGISTER
2852 040454 004737 017512      JSR      PC,FILLMEM        ;FILL MEMORY WITH RECORD SIZE
2853 040460 010337 044176      MOV      R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
2854 040464 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
2855 040470 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
2856 040474 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
2857 040500 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED
2858 040504 020102      CMP      R1,R2            ;ARE THEY EQUAL
2859 040506 001406      BEQ      75$              ;BR, IF OK
2860 040510 005237 002212      INC      FATFLG           ;BUMP COUNT
2864 040514      ERRHRD  ERRNO,WRterr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      040514 104456      TRAP    C$ERHRD
      040516 000703      .WORD  451
      040520 005111      .WORD  WRERR
      040522 012136      .WORD  PKTSSR
2865 040524      75$:  CKLOOP              ;LOOP IF SELECTED      TRAP    C$CLP1
      040524 104406
2866 040526 005723      TST      (R3)+            ;BUMP RECORD SIZE
2867 040530 022703 000414      CMP      #268.,R3        ;END OF RECORD YET
2868 040534 001346      BNE      65$              ;BR, IF MORE RECORDS TO WRITE
2869 040536      80$:  CKLOOP              ;LOOP IF SELECTED
      040536 104406      TRAP    C$CLP1
2870 040540 005743      TST      -(R3)            ;SET RECORD SIZE BACK TO 512.
2871 040542 013737 003114 044172      MOV      FREE,T24RB      ;STARTING READ BUFFER ADDRESS
2872
2873 ;*****
2874 ;
2875 ;READ REVERSE DATA,ACK,SWB COMMAND
2876 ;
2877 ;*****
2878
2879 040550 012737 110401 044170      MOV      #110401,T24PK3   ;READ REVERSE DATA,ACK,SWB COMMAND
2880 040556 012704 044170      165$:  MOV      #T24PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
2881 040562 010337 044176      MOV      R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
2882 040566 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
2883 040572 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
2884 040576 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
2885 040602 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED
2886 040606 020102      CMP      R1,R2            ;ARE THEY EQUAL
2887 040610 001406      BEQ      170$             ;BR, IF OK
2888 040612 005237 002212      INC      FATFLG           ;BUMP COUNT
2892 040616      ERRHRD  ERRNO,T24WDC,EXPREC ;TSSR INCORRECT AFTER READ DATA
      040616 104456      TRAP    C$ERHRD
      040620 000704      .WORD  452
      040622 045656      .WORD  T24WDC
      040624 015564      .WORD  EXPREC
2893 040626      170$:  CKLOOP              ;LOOP IF SELECTED

```



```

2943
2944 040750 004737 016061      JSR      PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
2945 040754 103407              BCS      20$             ;BR IF INIT WAS OK
2946 040756 005237 002212      INC      FATFLG          ;BUMP COUNT
2950 040762 010001              MOV      RO,R1           ;CONTENTS OF TSSR REGISTER
2951 040764              ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP      C$ERDF
                                .WORD    454
                                .WORD    SFIERR
                                .WORD    SFIMSG
2952 040774              20$:
2953 040774 013737 002172 044060      MOV      UNITN,T24DSW    ;SET DRIVE NUMBER IN PACKET
2954 041002 012704 044040      MOV      #T24PACKET,R4  ;SUBROUTINE NEEDS PACKET ADDRESS
2955
2956 ;*****
2957 ;
2958 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
2959 ;
2960 ;*****
2961
2962 041006 004737 010752      JSR      PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
2963 041012 103407              BCS      24$             ;BR, IF COMMAND ISSUED OK
2964 041014 005237 002212      INC      FATFLG          ;BUMP COUNT
2968 041020 010001              MOV      RO,R1           ;SAVE CONTENTS OF TSSR
2969 041022              ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP      C$ERHRD
                                .WORD    455
                                .WORD    WRTMSG
                                .WORD    SFIMSG
2970 041032              24$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
2971
2972 ;*****
2973 ;
2974 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
2975 ;
2976 ;*****
2977
2978 041034 004737 011104      JSR      PC,REWIND       ;CALL TAPE REWIND COMMAND
2979 041040 004737 016426      JSR      PC,CHKTSSR      ;SEE HOW TSSR IS
2980 041044 103407              BCS      30$             ;BR, IF NO PROBLEM
2981 041046 010001              MOV      RO,R1           ;SAVE TSSR
2982 041050 005237 002212      INC      FATFLG          ;BUMP COUNT
2986 041054              ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    456
                                .WORD    T24RWN
                                .WORD    PKTSSR
2987 041064              30$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
2988 041066 012703 001000      MOV      #512.,R3        ;RECORD SIZE
2989 041072 013737 003114 044172      MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
2990
2991 ;*****
2992 ;
2993 ;WRITE DATA,ACK,CVC=1 COMMAND
2994 ;
    
```

```

2995 ;*****
2996
2997 041100 012737 140005 044170      MOV      #140005,T24PK3      ;WRITE DATA,ACK,CVC-1 COMMAND
2998 041106 012704 044170              MOV      @T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
2999 041112
3000 041112 010337 044176      65$:    MOV      R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
3001 041116 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
3002 041122 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
3003 041126 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
3004 041132 012702 000200      MOV      @SSR,R2          ;SET UP EXPECTED
3005 041136 020102              CMP      R1,R2            ;ARE THEY EQUAL
3006 041140 001406              BEQ      75$              ;BR, IF OK
3007 041142 005237 002212      INC      FATFLG           ;BUMP COUNT
3011 041146              ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
          041146 104456              TRAP    C$ERHRD
          041150 000711              .WORD  457
          041152 005111              .WORD  WRERR
          041154 012136              .WORD  PKTSSR
3012 041156      75$:    CKLOOP                ;LOOP IF SELECTED
          041156 104406              TRAP    C$CLP1
3013 041160 012703 000400      MOV      #256.,R3         ;SIZE OF RECORD
3014 041164 013737 003114 044172      MOV      FREE,T24RB       ;STARTING READ BUFFER ADDRESS
3015
3016 ;*****
3017 ;
3018 ;READ DATA,ACK COMMAND
3019 ;
3020 ;*****
3021
3022 041172 012737 100401 044170      MOV      #100401,T24PK3   ;READ DATA,ACK COMMAND
3023 041200 012704 044170      165$:  MOV      @T24PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
3024 041204 010337 044176      MOV      R3,T24SZ        ;SET UP RECORD SIZE IN PACKET
3025 041210 010465 000000      MOV      R4,TSDB(R5)     ;ISSUE COMMAND
3026 041214 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
3027 041220 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
3028 041224 012702 100204      MOV      @SSR!SC!BIT2,R2 ;SET UP EXPECTED
3029 041230 020102              CMP      R1,R2          ;ARE THEY EQUAL
3030 041232 001406              BEQ      170$           ;BR, IF OK
3031 041234 005237 002212      INC      FATFLG         ;BUMP COUNT
3035 041240              ERRHRD  ERRNO,T24TRL,EXPREC ;TSSR INCORRECT AFTER READ DATA
          041240 104456              TRAP    C$ERHRD
          041242 000712              .WORD  458
          041244 046374              .WORD  T24TRL
          041246 015564              .WORD  EXPREC
3036 041250      170$:  CKLOOP                ;LOOP IF SELECTED
          041250 104406              TRAP    C$CLP1
3037
3038 ;*****
3039 ;
3040 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
3041 ;
3042 ;*****
3043
3044 041252 013701 044070      MOV      T248FR+6,R1     ;GET MESSAGE BUFFER (XSTO)
3045 041256 010102              MOV      R1,R2          ;SET UP EXPECTED
3046 041260 052702 010000      BIS      @BIT12,R2       ;SET THE RLL BIT IN EXPECTED
3047 041264 020102              CMP      R1,R2          ;ARE THEY EQUAL
    
```





```

3098 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
3099 ;
3100 ;*****
3101
3102 041412 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
3103 041416 103407 BCS 24$ ;BR, IF COMMAND ISSUED OK
3104 041420 005237 002212 INC FATFLG ;BUMP COUNT
3108 041424 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
3109 041426 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTICS FAILED
      041426 104456 TRAP C$ERHRD
      041430 000715 .WORD 461
      041432 005054 .WORD WRTMSG
      041434 012124 .WORD SFMSG
3110 041436 24$: CKLOOP ;LOOP IF SELECTED
      041436 104406 TRAP C$CLP1
3111
3112 ;*****
3113 ;
3114 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
3115 ;
3116 ;*****
3117
3118
3119 041440 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
3120 041444 001024 BNE 27$ ;BR IF SWITCH IS ON
3121 041446 112737 000200 044201 MOVB #200,T24BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
3122 041454 112737 000010 044200 MOVB #10,T24BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
3123 041462 012704 044150 MOV #T24PK2,R4 ;WRITE SUBSYS MEM PACKET
3124 041466 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
3125 041472 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
3126 041476 103407 BCS 28$ ;BR, IF NO ERROR
3127 041500 010001 MOV RO,R1 ;ERROR, SAVE TSSR
3128 041502 005237 002212 INC FATFLG ;BUMP COUNT
3132 041506 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      041506 104456 TRAP C$ERHRD
      041510 000716 .WORD 462
      041512 026320 .WORD T22SSR
      041514 012136 .WORD PKTSSR
3133 041516 27$:
3134 041516 28$: CKLOOP ;LOOP IF SELECTED
      041516 104406 TRAP C$CLP1
3135
3136
3137
3138 041520 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
3139 041524 004737 016426 JSR PC,CHKTSSR ;SEE HOW TSSR IS
3140 041530 103407 BCS 30$ ;BR, IF NO PROBLEM
3141 041532 010001 MOV RO,R1 ;SAVE TSSR
3142 041534 005237 002212 INC FATFLG ;BUMP COUNT
3146 041540 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      041540 104456 TRAP C$ERHRD
      041542 000717 .WORD 463
      041544 045326 .WORD T24RWN
      041546 012136 .WORD PKTSSR
3147 041550 30$: CKLOOP ;LOOP IF SELECTED
      041550 104406 TRAP C$CLP1
3148 041552 012703 000005 MOV #5.,R3 ;NUMBER OF RECORDS

```





```

3254
3255 042102 004737 010752          JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
3256 042106 103407                  BCS    24$                ;BR, IF COMMAND ISSUED OK
3257 042110 005237 002212          INC    FATFLG            ;BUMP COUNT
3261 042114 010001                  MOV    RO,R1             ;SAVE CONTENTS OF TSSR
3262 042116                  ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP    C$ERHRD
                                .WORD   467
                                .WORD   WRTMSG
                                .WORD   SFIMSG
                                042116 104456
                                042120 000723
                                042122 005054
                                042124 012124
3263 042126 24$: CKLOOP                ;LOOP IF SELECTED
                                TRAP    C$CLP1
3264 042130 013737 003114 044172    MOV    FREE,T24RB        ;STARTING WRITE BUFFER ADDRESS
3265
3266 ;*****
3267 ;
3268 ;ILLEGAL MODE DATA,ACK COMMAND
3269 ;
3270 ;*****
3271
3272 042136 012737 104001 044170    MOV    #104001,T24PK3    ;ILLEGAL MODE DATA,ACK COMMAND
3273 042144 012704 044170          MOV    #T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
3274 042150 012737 000400 044176    MOV    #256.,T24SZ       ;SET UP RECORD SIZE IN PACKET
3275 042156 010465 000000          MOV    R4,TSDB(R5)       ;ISSUE COMMAND
3276 042162 004737 016340          JSR    PC,WAITF          ;WAIT FOR SSR!BIT1!BIT2 TO SET
3277 042166 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
3278 042172 012702 100206          MOV    #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
3279 042176 020102                  CMP    R1,R2             ;ARE THEY EQUAL
3280 042200 001406                  BEQ    75$                ;BR, IF OK
3281 042202 005237 002212          INC    FATFLG            ;BUMP COUNT
3285 042206                  ERRHRD  ERRNO,T24WDF,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C$ERHRD
                                .WORD   468
                                .WORD   T24WDF
                                .WORD   PKTSSR
                                042206 104456
                                042210 000724
                                042212 044515
                                042214 012136
3286 042216 75$: CKLOOP                ;LOOP IF SELECTED
                                TRAP    C$CLP1
3287
3288 ;*****
3289 ;
3290 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
3291 ;
3292 ;*****
3293
3294 042220 013701 044070          MOV    T24BFR+6,R1       ;GET MESSAGE BUFFER
3295 042224 010102                  MOV    R1,R2             ;SET UP EXPECTED
3296 042226 052702 001000          BIS    #BIT9,R2         ;SET THE ILC BIT IN EXPECTED
3297 042232 020102                  CMP    R1,R2             ;ARE THEY EQUAL
3298 042234 001406                  BEQ    180$              ;BR, IF EQUAL (ALL IS WELL)
3299 042236 005237 002212          INC    FATFLG            ;BUMP COUNT
3303 042242                  ERRHRD  ERRNO,T24LOQ,EXPREC ;THE ILC BIT WAS NOT SET IN XSTO
                                TRAP    C$ERHRD
                                .WORD   469
                                .WORD   T24LOQ
                                .WORD   EXPREC
                                042242 104456
                                042244 000725
                                042246 044656
                                042250 015564
3304 042252 180$: CKLOOP                ;LOOP IF SELECTED
                                TRAP    C$CLP1
                                042252 104406
    
```

```

3305 042254                                ENDSUB                                : >>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
          042254                                L10064:
          042254 104403                                TRAP                                C#ESUB
3306 042256 023727 002212 000017          CMP        FATFLG,#15.                        ;IS ERROR COUNT AT 25
3307 042264 103402                                BLO        999$                                ;BR, IF LESS THAN 25
3308 042266 004737 017272                        JSR        PC,CKDROP                        ;TRY TO DROP THE UNIT
3309 042272                                999$:
3310
3311                                :
3312                                :
3313                                ;TEST 4, SUBTEST 11
3314                                :
3315                                ;VERIFIES THAT ILLEGAL BUFFER ADDRESSES CAUSE A
3316                                ;FUNCTION REJECT TERMINATION WITH ILLEGAL ADDRESS
3317                                ;(ILA) ERROR BIT SET.
3318                                :
3319                                :
3320                                :
3321                                :
3322                                :
3323 042272                                BGNSUB                                : >>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
          042272                                T4.11:
          042272 104402                                TRAP                                C#BSUB
3324 042274 004737 046664                        JSR        PC,T24RT3                        ;SET COMMAND PACKET UP CLEAR
3325 042300 004737 046530                        JSR        PC,T24REST                       ;SET COMMAND PACKET
3326 042304 004737 046622                        JSR        PC,T24RT2                       ;SET UP OTHER COMMAND PACKET
3327
3328                                :*****
3329                                :
3330                                ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
3331                                :
3332                                :*****
3333
3334 042310 004737 016064                        JSR        PC,SOFINIT                      ;DO INITIALIZE ON CONTROLLER
3335 042314 103407                                BCS        20$                              ;BR IF INIT WAS OK
3336 042316 005237 002212                        INC        FATFLG                          ;BUMP COUNT
3340 042322 010001                                MOV        RO,R1                          ;CONTENTS OF TSSR REGISTER
3341 042324                                ERRDF     ERRNO,SFIERR,SFIMSG            ;FATAL ERROR TSSR WAS NOT OK
          042324 104455                                                                TRAP                                C#ERDF
          042326 000726                                                                .WORD                              470
          042330 003650                                                                .WORD                              SFIERR
          042332 012124                                                                .WORD                              SFIMSG
3342 042334                                20$:
3343 042334 013737 002172 044060            MOV        UNITN,T24DSW                     ;SET UP DRIVE NUMBER
3344 042342 012704 044040                        MOV        #T24PACKET,R4                   ;SUBROUTINE NEEDS PACKET ADDRESS
3345
3346                                :*****
3347                                :
3348                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
3349                                :
3350                                :*****
3351
3352 042346 004737 010752                        JSR        PC,WRTCHR                      ;ISSUE WRITE CHARACTERISTICS
3353 042352 103407                                BCS        24$                              ;BR, IF COMMAND ISSUED OK
3354 042354 005237 002212                        INC        FATFLG                          ;BUMP COUNT
3358 042360 010001                                MOV        RO,R1                          ;SAVE CONTENTS OF TSSR
3359 042362                                ERRHRD    ERRNO,WRTMSG,SFIMSG            ;WRITE CHARACTERISTIC FAILED

```



```

3406 042540 004737 017272          JSR    PC,CKDROP          ;TRY TO DROP THE UNIT
3407 042544          999$:
3408
3409          ;;
3410          ;
3411          ;TEST 4, SUBTEST 12
3412          ;
3413          ;VERIFIES THAT A DATA BUFFER ADDRESS REFERENCING
3414          ;NONEXISTANT MEMORY CAUSES RECOVERABLE ERROR
3415          ;TERMINATION (TC=4), WITH THE NXM BIT SET IN THE TSSR, AND
3416          ;THAT THE TAPE IS ULTIMATELY POSITIONED PROPERLY.
3417          ;
3418
3419 042544          BGNSUB          ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
          042544          T4.12:
          042544 104402          TRAP    C$BSUB
3420 042546 005737 003126          TST    NXMFLG          ;DO WE HAVE IT?
3421 042552 001002          BNE    10$          ;BR, IF ENOUGH
3422 042554 000137 043012          JMP    80$          ;SKIP THIS TEST IF NOT
3423 042560 004737 046664 10$:   JSR    PC,T24RT3      ;SET COMMAND PACKET UP CLEAR
3424 042564 004737 046530          JSR    PC,T24REST     ;SET COMMAND PACKET
3425 042570 004737 046622          JSR    PC,T24RT2     ;SET UP OTHER COMMAND PACKET
3426
3427          ;*****
3428          ;
3429          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
3430          ;
3431          ;*****
3432
3433 042574 004737 016064          JSR    PC,SOFINIT     ;DO INITIALIZE ON CONTROLLER
3434 042600 103407          BCS    20$          ;BR IF INIT WAS OK
3435 042602 005237 002212          INC    FATFLG        ;BUMP COUNT
3439 042606 010001          MOV    R0,R1         ;CONTENTS OF TSSR REGISTER
3440 042610          ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
          042610 104455          TRAP    C$ERDF
          042612 000732          .WORD  474
          042614 003650          .WORD  SFIERR
          042616 012124          .WORD  SFIMSG
3441 042620
3442 042620 013737 002172 044060 20$:   MOV    UNITN,T24DSW   ;SET UP DRIVE NUMBER
3443 042626 012704 044040          MOV    #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
3444
3445          ;*****
3446          ;
3447          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
3448          ;
3449          ;*****
3450
3451 042632 004737 010752          JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
3452 042636 103407          BCS    24$          ;BR, IF COMMAND ISSUED OK
3453 042640 005237 002212          INC    FATFLG        ;BUMP COUNT
3457 042644 010001          MOV    R0,R1         ;SAVE CONTENTS OF TSSR
3458 042646          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
          042646 104456          TRAP    C$ERHRD
          042650 000733          .WORD  475
          042652 005054          .WORD  WRTMSG
          042654 012124          .WORD  SFIMSG
    
```











```

043430 104406                                     TRAP      C#CLP1
3665
3666 ;*****
3667 ;
3668 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
3669 ;
3670 ;*****
3671
3672 043432 004737 011104          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
3673 043436 004737 016426          JSR      PC,CHKTSSR        ;SEE HOW TSSR IS
3674 043442 103407                  BCS      30$               ;BR, IF NO PROBLEM
3675 043444 010001                  MOV      R0,R1            ;SAVE TSSR
3676 043446 005237 002212          INC      FATFLG           ;BUMP COUNT
3680 043452                  ERRHRD   ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    484
                                .WORD    T24RWN
                                .WORD    PKTSSR
                                TRAP      C#CLP1
043452 104456
043454 000744
043456 045326
043460 012136
3681 043462          30$:  CKLOOP                ;LOOP IF SELECTED
043462 104406                                     TRAP      C#CLP1
3682 043464 012703 000400          MOV      #256.,R3         ;RECORD SIZE
3683 043470 013737 003114 044172    MOV      FREE,T24RB       ;STARTING WRITE BUFFER ADDRESS
3684
3685 ;*****
3686 ;
3687 ;WRITE DATA,ACK,CVC=1 COMMAND
3688 ;
3689 ;*****
3690
3691 043476 012737 140005 044170    MOV      #140005,T24PK3   ;WRITE DATA,ACK,CVC=1 COMMAND
3692 043504 012704 044170          MOV      #T24PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
3693 043510          65$:
3694 043510 010337 044176          MOV      R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
3695 043514 010465 000000          MOV      R4,TSDB(R5)     ;ISSUE COMMAND
3696 043520 004737 016340          JSR      PC,WAITF         ;WAIT FOR SSR TO SET
3697 043524 016501 000002          MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
3698 043530 012702 000200          MOV      #SSR,R2         ;SET UP EXPECTED
3699 043534 020102                  CMP      R1,R2            ;ARE THEY EQUAL
3700 043536 001406                  BEQ      75$              ;BR, IF OK
3701 043540 005237 002212          INC      FATFLG           ;BUMP COUNT
3705 043544          ERRHRD   ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERHRD
                                .WORD    485
                                .WORD    WRERR
                                .WORD    PKTSSR
                                TRAP      C#CLP1
043544 104456
043546 000745
043550 005111
043552 012136
3706 043554          75$:  CKLOOP                ;LOOP IF SELECTED
043554 104406                                     TRAP      C#CLP1
3707 043556 012703 000400          MOV      #256.,R3         ;RECORD SIZE
3708 043562 013737 003114 044172    MOV      FREE,T24RB       ;STARTING READ BUFFER ADDRESS
3709
3710 ;*****
3711 ;
3712 ;READ REVERSE DATA,ACK COMMAND
3713 ;
3714 ;*****
3715
3716 043570 012737 100401 044170    MOV      #100401,T24PK3   ;READ REVERSE DATA,ACK COMMAND

```

```

3717 043576 012704 044170      165$: MOV    @T24PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
3718 043602 010337 044176      MOV    R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
3719 043606 010465 000000      MOV    R4,TSDB(R5)       ;ISSUE COMMAND
3720 043612 004737 016340      JSR    PC,WAITF          ;WAIT FOR SSR TO SET
3721 043616 016501 000002      MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
3722 043622 012702 000200      MOV    @SSR,R2          ;SET UP EXPECTED
3723 043626 020102              CMP    R1,R2             ;ARE THEY EQUAL
3724 043630 001406              BEQ    170$              ;BR, IF OK
3725 043632 005237 002212      INC    FATFLG            ;BUMP COUNT
3729 043636              ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                               TRAP  C$ERHRD
                               .WORD 486
                               .WORD T24TRL
                               .WORD PKTSSR
                               TRAP  C$CLP1
043636 104456
043640 000746
043642 046374
043644 012136
3730 043646              170$: CKLOOP          ;LOOP IF SELECTED
043646 104406              TRAP  C$CLP1
3731 043650 012703 000400      MOV    @256,R3           ;RECORD SIZE
3732 043654 013737 003114 044172  MOV    FREE,T24RB        ;STARTING READ BUFFER ADDRESS
3733
3734 ;*****
3735 ;
3736 ;READ REVERSE DATA,ACK COMMAND
3737 ;
3738 ;*****
3739
3740 043662 012737 100401 044170      195$: MOV    @100401,T24PK3      ;READ REVERSE DATA,ACK COMMAND
3741 043670 012704 044170      MOV    @T24PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
3742 043674 010337 044176      MOV    R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
3743 043700 010465 000000      MOV    R4,TSDB(R5)       ;ISSUE COMMAND
3744 043704 004737 016340      JSR    PC,WAITF          ;WAIT FOR SSR TO SET
3745 043710 016501 000002      MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
3746 043714 012702 100204      MOV    @SSR!SC!BIT2,R2   ;SET UP EXPECTED
3747 043720 020102              CMP    R1,R2             ;ARE THEY EQUAL
3748 043722 001406              BEQ    200$              ;BR, IF OK
3749 043724 005237 002212      INC    FATFLG            ;BUMP COUNT
3753 043730              ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                               TRAP  C$ERHRD
                               .WORD 487
                               .WORD T24TRL
                               .WORD PKTSSR
043730 104456
043732 000747
043734 046374
043736 012136
3754 043740              200$: CKLOOP          ;LOOP IF SELECTED
043740 104406              TRAP  C$CLP1
3755 043742 013701 044076      MOV    T24BFR+14,R1      ;GET MESSAGE BUFFER (XST3)
3756 043746 010102              MOV    R1,R2             ;SET UP EXPECTED
3757 043750 052702 000001      BIS    @BIT0,R2          ;SET THE RIB BIT IN EXPECTED
3758 043754 020102              CMP    R1,R2             ;ARE THEY EQUAL
3759 043756 001406              BEQ    210$              ;BR, IF EQUAL (ALL IS WELL)
3760 043760 005237 002212      INC    FATFLG            ;BUMP COUNT
3764 043764              ERRHRD ERRNO,T24LOR,EXPREC ;THE RIB BIT WAS NOT SET IN XSTO
                               TRAP  C$ERHRD
                               .WORD 488
                               .WORD T24LOR
                               .WORD EXPREC
043764 104456
043766 000750
043770 044272
043772 015564
3765 043774              210$: CKLOOP
043774 104406              TRAP  C$CLP1
3766 043776              ENDSUB
                               ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>>>>>>
                               L10070:

```

TSV7 - HARDWARE TESTS 1-8      MACRO M1113 14-JUN-84 14:17  
 TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

SEQ 0173

```

    043776 104403
3767 044000 023727 002212 000017      CMP   FATFLG,#15.      TRAP      C#ESUB
3768 044006 103402                      BLO   999$            ;IS ERROR COUNT AT 25
3769 044010 004737 017272                      JSR   PC,CKDROP      ;BR, IF LESS THAN 25
3770 044014                      999$:                ;TRY TO DROP THE UNIT
3771                      ;
3772                      ;
3773                      ;
3774 044014 004737 016546      JSR   PC,TSTLOOP     ;DO WE NEED TO ITERATE TEST
3775 044020 103002                      BCC   163$           ;BR, IF NO LOOP REQUIRED
3776 044022 000137 034516      JMP   T24LOOP        ;EXECUTE AGAIN
3777 044026                      163$:
3778 044026      EXIT   TST          ;ALL DONE THIS TEST
    044026 104432                      TRAP      C#EXIT
    044030 002664                      .WORD    L10052-.
3779
3780
3781      ;*
3782      ;LOCAL STORAGE FOR THIS TEST
3783      ;-
3784 044032      .BLKB  10-<.-TSV2&7>
3786 044040      T24PACKET:
3787 044040 100204      .WORD  100204      ;COMMAND PACKET FOR TEST
3788 044042 044050      .WORD  T24DATA     ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
3789 044044 000000      .WORD   0           ;ADDRESS OF CHARACTERISTICS BLOCK
3790 044046 000012      .WORD  10.         ;STARTING VALUE OF BLOCK SIZE
3791 044050      T24DATA:
3792 044050 044062      .WORD  T24BFR      ;CHARACTERISTICS DATA BLOCK
3793 044052 000000      .WORD   0           ;ADDRESS OF MESSAGE BUFFER
3794 044054 000024      .WORD  20.        ;LENGTH OF MESSAGE BUFFER
3795 044056 000000      .WORD   0           ;DRIVE SELECTION BITS 2-0
3796 044060 000000      T24DSW: .WORD  0     ;MESSAGE BUFFER
3797 044062      T24BFR: .BLKB  25.
3798
3799      ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
3800
3802 044144      .BLKB  10-<.-TSV2&7>
3804 044150      T24PK2:
3805 044150 100206      .WORD  100206     ;WRITE SUB SYS MEM COMMAND, IE AND ACK
3806 044152 044200      .WORD  T24BF2     ;ADDRESS OF SELECT BLOCK DATA
3807 044154 000000      .WORD   0
3808 044156 000006      .WORD   6.        ;SIZE OF DATA PACKET
3809
3811 044160      .BLKB  10-<.-TSV2&7>
3813 044170      T24PK3:
3814 044170 100205      .WORD  100205     ;READ COMMAND, IE AND ACK
3815 044172      T24RB:
3816 044172 003114      T24WB: .WORD  FREE  ;ADDRESS OF WRITE BUFFER
3817 044174 000000      .WORD   0
3818 044176 000000      T24SZ: .WORD  0     ;SIZE OF BUFFER (EXTENT)
3819      .EVEN
3820
3821
3822
3823 044200      T24BF2:
3824 044200      T24BS0: .BYTE  10   ;BSELO AREA
3825 044201      T24BS1: .BYTE  200  ;BSEL1 AREA
3826 044202 000000      T24S2: .WORD  0     ;SEL 2 AREA
  
```





```

3884 046550 005021          CLR      (R1).          ;EXTENDED ADDRESS
3885 046552 012721 000012  MOV      @10.,(R1).     ;SIZE OF DATA BLOCK IN BYTES
3886 046556 012721 044062  MOV      @T24BFR,(R1). ;ADDRESS OF MESSAGE BUFFER
3887 046562 005021          CLR      (R1).          ;
3888 046564 012721 000024  MOV      @20.,(R1).     ;LENGTH OF MESSAGE BUFFER
3889 046570 005021          CLR      (R1).          ;
3890 046572 012711 000000  MOV      @0,(R1)        ;SELECT DRIVE ZERO
3891 046576 012702 000030  MOV      @24.,R2        ;NUMBER OF LOCATIONS TO BE CLEARED
3892 046602 012762 177777 044062 64$: MOV      @177777,T24BFR(R2) ;ALL ONES TO MESSAGE BUFFER
3893 046610 005742          TST      -(R2)          ;NEXT LOCATION
3894 046612 022702 000000  CMP      @0,R2          ;CHECK FOR END OF LOOP
3895 046616 001371          BNE      64$           ;KEEP GOING UNTIL DONE
3896 046620 000207          RTS      PC            ;RETURN
3897
3898
3899 046622          T24RT2:
3900 046622          SAVREG          ;SAVE THE REGISTERS
3901 046626 012701 044150  MOV      @T24PK2,R1     ;START OF THE PACKET
3902 046632 012721 100206  MOV      @100206,(R1).  ;WRITE SUBSYSTEM MEM. WITH ACK, IE
3903 046636 012721 044200  MOV      @T24BF2,(R1). ;ADDRESS OF DATA BLOCK
3904 046642 005021          CLR      (R1).          ;EXTENDED ADDRESS
3905 046644 012721 000006  MOV      @6.,(R1).     ;SIZE OF DATA BLOCK IN BYTES
3906 046650 005021          CLR      (R1).          ;
3907 046652 012701 044200  MOV      @T24BF2,R1     ;POINT TO DATA SEL AREA
3908 046656 005021          CLR      (R1).          ;
3909 046660 005011          CLR      (R1)          ;
3910 046662 000207          RTS      PC            ;RETURN
3911 046664          T24RT3:
3912 046664          SAVREG          ;SAVE THE REGISTERS
3913 046670 012701 044170  MOV      @T24PK3,R1     ;START OF THE PACKET
3914 046674 012721 000000  MOV      @0,(R1).       ;CLEAR AREA OUT
3915 046700 012721 000000  MOV      @0,(R1).       ;ADDRESS OF DATA BLOCK
3916 046704 005021          CLR      (R1).          ;EXTENDED ADDRESS
3917 046706 012711 000000  MOV      @0,(R1)        ;SIZE OF DATA BLOCK IN BYTES
3918 046712 000207          RTS      PC            ;RETURN
3919 046714          ENDTST
      046714
      046714 104401          L10052: TRAP C$ETST

```

```

3920
3921          .SBTTL TEST 5: SPACE RECORDS
3922

```

```

3923 ;*
3924 ;
3925 ;THIS TEST VERIFIES THAT THE SPACE RECORDS FORWARD AND SPACE
3926 ;RECORDS REVERSE POSITION COMMANDS OPERATE PROPERLY WHEN SPACING
3927 ;OVER NORMAL DATA RECORDS. OPERATION WHEN SPACING OVER TAPE MARKS
3928 ;IS VERIFIED IN A SUBSEQUENT TEST. THE BASIC WRITE DATA TEST
3929 ;SHOULD HAVE BEEN RUN SUCCESSFULLY FOR THIS TEST TO PRODUCE MEANINGFUL
3930 ;RESULTS. THIS TEST CONSISTS OF A SERIES OF SUBTESTS. IN EACH
3931 ;OF THE SUBTESTS, THE TAPE IS ENTIRELY WRITTEN WITH RECORDS
3932 ;OF VARYING SIZES AND DATA CONTENT; THE FIRST 4 BYTES OF EACH
3933 ;RECORD INDICATE THAT RECORD'S RELATIVE POSITION ON TAPE. AFTER
3934 ;EACH SPACING OPERATION, THE TAPE POSITION IS VERIFIED BY READING
3935 ;THE NEXT OR PREVIOUS RECORD AND COMPARING THE POSITION DATA WITH
3936 ;THE EXPECTED RESULT.
3937 ;
3938 ;THE TEST CONSISTS OF THE FOLLOWING 8 SUBTESTS

```



```

047056 003650 .WORD SFIERR
047060 012124 .WORD SFIMSG
3990 047062
3991 047062 013737 002172 054130 10$: MOV UNITN,T25DSW ;SET UP DRIVE NUMBER
3992 047070 012704 054110 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
3993
3994 ;*****
3995 ;
3996 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
3997 ;
3998 ;*****
3999
4000 047074 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4001 047100 103407 BCS 15$ ;BR, IF COMMAND ISSUED OK
4002 047102 005237 002212 INC FATFLG ;BUMP COUNT
4006 047106 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4007 047110 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
047110 104456 TRAP C$ERHRD
047112 000766 .WORD 502
047114 005054 .WORD WRTMSG
047116 012124 .WORD SFIMSG
4008
4009 ;*****
4010 ;
4011 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4012 ;
4013 ;*****
4014
4015 047120 15$: CKLOOP TRAP C$CLP1
047120 104406
4016 047122 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4017 047126 103407 BCS 30$ ;BR, IF NO PROBLEM
4018 047130 010001 MOV RO,R1 ;SAVE TSSR
4019 047132 005237 002212 INC FATFLG ;BUMP COUNT
4023 047136 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
047136 104456 TRAP C$ERHRD
047140 000767 .WORD 503
047142 055245 .WORD T25RWN
047144 012136 .WORD PKTSSR
4024 047146 30$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
047146 104406
4025
4026 ;*****
4027 ;
4028 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4029 ;
4030 ;*****
4031
4032 047150 013701 054140 MOV T25BFR+6,R1 ;PICK UP XSTO
4033 047154 010102 MOV R1,R2 ;SET UP EXPECTED
4034 047156 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
4035 047162 020102 CMP R1,R2 ;DOES EXP = REC'D
4036 047164 001406 BEQ 40$ ;BR, IF EQUAL (OK)
4037 047166 005237 002212 INC FATFLG ;BUMP COUNT
4041 047172 ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
047172 104456 TRAP C$ERHRD
047174 000770 .WORD 504

```

```

047176 054435 .WORD T25BOT
047200 015564 .WORD EXPREC
4042 047202 40$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
047202 104406
4043 047204 012703 000400 MOV #256.,R3 ;RECORD SIZE
4044 047210 013737 003114 054242 MOV FREE,T25RB ;STARTING WRITE BUFFER ADDRESS
4045
4046 ;*****
4047 ;
4048 ;WRITE DATA,ACK,CVC=1 COMMAND
4049 ;
4050 ;*****
4051
4052 047216 012737 140005 054240 MOV #140005,T25PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
4053 047224 012704 054240 MOV #T25PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
4054 047230 65$:
4055 047230 010337 054246 MOV R3,T25SZ ;SET UP RECORD SIZE IN PACKET
4056 047234 013777 054270 133652 MOV T25CNT,#FREE ;LOAD UP RECORD COUNTER IN WRT BUFFER
4057 047242 062737 000001 054270 ADD #1,T25CNT ;GET READY FOR NEXT RECORD
4058 047250 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
4059 047254 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
4060 047260 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4061 047264 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
4062 047270 020102 CMP R1,R2 ;ARE THEY EQUAL
4063 047272 001411 BEQ 75$ ;BR, IF OK
4064 047274 032701 000004 BIT #BIT2,R1 ;CHECK FOR TAPE STATUS ALERT
4065 047300 001014 BNE 120$ ;BR, IF TSA IS SET (SUSPECT IS EOT)
4066 047302 005237 002212 INC FATFLG ;BUMP COUNT
4070 047306 ERRHRD ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
047306 104456 TRAP C$ERHRD
047310 000771 .WORD 505
047312 005111 .WORD WRTERR
047314 012136 .WORD PKTSSR
4071 047316 75$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
047316 104406
4072 047320 005203 INC R3 ;BUMP RECORD SIZE
4073 047322 022703 001000 CMP #512.,R3 ;END OF RECORD YET
4074 047326 001340 BNE 65$ ;BR, IF MORE RECORDS TO WRITE
4075 047330 000415 BR 125$ ;ENOUGH RECORDS
4076 047332 120$:
4077 ;*****
4078 ;
4079 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4080 ;
4081 ;
4082 ;*****
4083
4084 047332 013701 054140 MOV T25BFR+6,R1 ;QUICK CHECK FOR EOT SET
4085 047336 010102 MOV R1,R2 ;SET UP EXPECTED
4086 047340 052702 000001 BIS #BIT0,R2 ;SET THE EOT BIT XSTO
4087 047344 020102 CMP R1,R2 ;IS THE EOT BIT SET IN XSTO
4088 047346 001406 BEQ 125$ ;BR, IF SET (GOOD)
4089 047350 005237 002212 INC FATFLG ;BUMP COUNT
4093 047354 ERRDF ERRNO,T25NET,EXPREC ;DEVICE FATAL NOT EOT FOUND ETC.
047354 104455 TRAP C$ERDF
047356 000772 .WORD 506
047360 054571 .WORD T25NET

```

```

047362 015564
4094 047364 125$:
4095
4096 ;*****
4097 ;
4098 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4099 ;
4100 ;*****
4101
4102 047364 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4103 047370 103407 BCS 130$ ;BR, IF NO PROBLEM
4104 047372 010001 MOV R0,R1 ;SAVE TSSR
4105 047374 005237 002212 INC FATFLG ;BUMP COUNT
4109 047400 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
047400 104456 TRAP C$ERHRD
047402 000773 .WORD 507
047404 055245 .WORD T25RWN
047406 012136 .WORD PKTSSR
4110 047410 130$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
047410 104406
4111 047412 012737 000007 054130 MOV #7,T25DSW ;SET UP DRIVE NUMBER
4112 047420 012704 054110 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4113
4114 ;*****
4115 ;
4116 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4117 ;
4118 ;*****
4119
4120 047424 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4121 047430 103407 BCS 140$ ;BR, IF COMMAND ISSUED OK
4122 047432 005237 002212 INC FATFLG ;BUMP COUNT
4126 047436 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4127 047440 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
047440 104456 TRAP C$ERHRD
047442 000774 .WORD 508
047444 005054 .WORD WRTMSG
047446 012124 .WORD SFIMSG
4128 047450 140$: CKLOOP ;SCOPE LOOP TRAP C$CLP1
047450 104406
4129 047452 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES
4130 047456 001044 BNE 160$ ;BR IF SWITCH IS ON
4131
4132 047460 112737 000200 054251 MOVB #200,T25BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
4133 047466 112737 000010 054250 MOVB #10,T25BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
4134 047474 012704 054220 MOV #T25PK2,R4 ;WRITE SUBSYS MEM PACKET
4135 047500 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
4136 047504 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
4137 047510 103407 BCS 150$ ;BR, IF NO ERROR
4138 047512 010001 MOV R0,R1 ;ERROR, SAVE TSSR
4139 047514 005237 002212 INC FATFLG ;BUMP COUNT
4143 047520 ERRHRD ERRNO,T25SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
047520 104456 TRAP C$ERHRD
047522 000775 .WORD 509
047524 054274 .WORD T25SSR
047526 012136 .WORD PKTSSR
4144 047530 150$: CKLOOP ;LOOP IF SELECTED

```





```

050016 005054 .WORD WRTMSG
050020 012124 .WORD SFIMSG
4248
4249
4250 ;*****
4251 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4252 ;
4253 ;*****
4254
4255 050022 15$: CKLOOP
050022 104406 TRAP C$CLP1
4256 050024 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4257 050030 103407 BCS 30$ ;BR, IF NO PROBLEM
4258 050032 010001 MOV R0,R1 ;SAVE TSSR
4259 050034 005237 002212 INC FATFLG ;BUMP COUNT
4263 050040 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
050040 104456 TRAP C$ERHRD
050042 001003 .WORD 515
050044 055245 .WORD T25RWN
050046 012136 .WORD PKTSSR
4264 050050 30$: CKLOOP ;LOOP IF SELECTED
050050 104406 TRAP C$CLP1
4265 050052 005737 002216 140$: TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
4266 050056 001044 BNE 160$ ;BR IF SWITCH IS ON
4267
4268 050060 112737 000200 054251 MOVB #200,T25BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
4269 050066 112737 000010 054250 MOVB #10,T25BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
4270 050074 012704 054220 MOV #T25PK2,R4 ;WRITE SUBSYS MEM PACKET
4271 050100 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
4272 050104 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
4273 050110 103407 BCS 150$ ;BR, IF NO ERROR
4274 050112 010001 MOV R0,R1 ;ERROR, SAVE TSSR
4275 050114 005237 002212 INC FATFLG ;BUMP COUNT
4279 050120 ERRHRD ERRNO,T25SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
050120 104456 TRAP C$ERHRD
050122 001004 .WORD 516
050124 054274 .WORD T25SSR
050126 012136 .WORD PKTSSR
4280 050130 150$: CKLOOP ;LOOP IF SELECTED
050130 104406 TRAP C$CLP1
4281 050132 012737 000007 054130 MOV #7,T25DSW ;SET UP DRIVE NUMBER
4282 050140 012704 054110 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4283
4284 ;*****
4285 ;
4286 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4287 ;
4288 ;*****
4289
4290 050144 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4291 050150 103407 BCS 160$ ;BR, IF COMMAND ISSUED OK
4292 050152 005237 002212 INC FATFLG ;BUMP COUNT
4296 050156 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4297 050160 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
050160 104456 TRAP C$ERHRD
050162 001005 .WORD 517
050164 005054 .WORD WRTMSG

```



```

050166 012124
4298 050170 104406 160: CKLOOP ;SCOPE LOOP .WORD SFIMSG
050170 104406 TRAP C#CLP1
4299 050172 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4300 050176 032701 000100 BIT #OFL,R1 ;CHECK FOR THE OFFLINE BIT SET
4301 050202 001006 BNE 170: ;BR, IF OFFLINE (GOOD)
4302 050204 005237 002212 INC FATFLG ;BUMP COUNT
4306 050210 ERRDF ERRNO,T25OFL,SFIMSG ;OFF LINE SHOULD HAVE BEEN SET (BAD)
050210 104455 TRAP C#ERDF
050212 001006 .WORD 518
050214 055314 .WORD T25OFL
050216 012124 .WORD SFIMSG
4307 050220 170: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
050220 104406
4308
4309 ;*****
4310 ;
4311 ;SPACE REVERSE COMMAND IN PLACE
4312 ;
4313 ;*****
4314
4315 050222 012737 100410 054240 180: MOV #100410,T25PK3 ;SPACE REVERSE COMMAND IN PLACE
4316 050230 012737 000001 054242 MOV #1,T25R8 ;NUMBER OF RECORDS TO SPACE
4317 050236 012704 054240 MOV #T25PK3,R4 ;R4 = POINTER TO PACKET
4318 050242 010465 000000 MOV R4,TSD8(R5) ;ISSUE COMMAND
4319 050246 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
4320 050252 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4321 050256 012702 100306 MOV #SSR!SC!OFL!BIT1!BIT2,R2 ;SET UP EXPECTED
4322 050262 020102 CMP R1,R2 ;ARE THEY EQUAL
4323 050264 001406 BEQ 190: ;BR, IF OK ESP. FUNCTION REJECT
4324 050266 005237 002212 INC FATFLG ;BUMP COUNT
4328 050272 ERRHRD ERRNO,T25TM,PKTSSR ;TSSR INCORRECT AFTER TAPE MOTION CMD
050272 104456 TRAP C#ERHRD
050274 001007 .WORD 519
050276 054502 .WORD T25TM
050300 012136 .WORD PKTSSR
4329 050302 190: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
050302 104406 ;***** END SUBTEST *****
4330 050304 ENDSUB L10073: TRAP C#ESUB
050304 104403
4331 050306 023727 002212 000017 CMP FATFLG,#15. ;IS ERROR COUNT AT 25
4332 050314 103402 BLO 999: ;BR, IF LESS THAN 25
4333 050316 004737 017272 JSR PC,CKDROP ;TRY TO DROP THE UNIT
4334 050322 999:
4335 ;
4336 ;
4337 ;
4338 ;TEST 5, SUBTEST 3
4339 ;
4340 ;VERIFIES THAT SPACE RECORDS FORWARD CAN SPACE ONE
4341 ;RECORD OFF BOT AND CAUSES BOT STATUS TO BE CLEARED.
4342 ;
4343 ;
4344 ;
4345 ;
4346 050322 BGNSUB ;***** BEGIN SUBTEST *****

```

```

                                T5.3:
                                TRAP   C#BSUB
4347 050322 104402                JSR    PC,T25REST      ;SET COMMAND PACKET
4348 050324 004737 055456        JSR    PC,T25RT2      ;SET UP OTHER COMMAND PACKET
4349 050330 004737 055550        JSR    PC,T25RT3      ;SET UP OTHER COMMAND PACKET
4350
4351 ;*****
4352 ;
4353 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4354 ;
4355 ;*****
4356
4357 050340 004737 016064        JSR    PC,SOFINIT     ;DO INITIALIZE ON CONTROLLER
4358 050344 103407                BCS    10$            ;BR IF INIT WAS OK
4359 050346 005237 002212        INC    FATFLG         ;BUMP COUNT
4363 050352 010001                MOV    R0,R1          ;CONTENTS OF TSSR REGISTER
4364 050354                ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP   C#ERDF
                                .WORD  520
                                .WORD  SFIERR
                                .WORD  SFIMSG
4365 050364 013737 002172 054130 10$:  MOV    UNITN,T25DSW    ;SET UP DRIVE NUMBER
4366
4367 050372 012704 054110        MOV    @T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4368
4369 ;*****
4370 ;
4371 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4372 ;
4373 ;*****
4374
4375 050376 004737 010752        JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
4376 050402 103407                BCS    15$            ;BR, IF COMMAND ISSUED OK
4377 050404 005237 002212        INC    FATFLG         ;BUMP COUNT
4381 050410 010001                MOV    R0,R1          ;SAVE CONTENTS OF TSSR
4382 050412                ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP   C#ERHRD
                                .WORD  521
                                .WORD  WRTMSG
                                .WORD  SFIMSG
4383
4384 ;*****
4385 ;
4386 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4387 ;
4388 ;*****
4389
4390 050422                15$:  CKLOOP
                                TRAP   C#CLP1
4391 050422 104406                JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
4392 050424 004737 011104        BCS    30$            ;BR, IF NO PROBLEM
4393 050430 103407                MOV    R0,R1          ;SAVE TSSR
4394 050432 010001                INC    FATFLG         ;BUMP COUNT
4398 050434 005237 002212        ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP   C#ERHRD
                                .WORD  522
                                .WORD  T25RWN
4398 050440 104456
4399 050442 001012
4400 050444 055245

```

```

4399 050446 012136          30$:  CKLOOP          ;LOOP IF SELECTED          .WORD  PKTSSR
      050450          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
      050450 104406          ;*****
4400          ;
4401          ;
4402          ;
4403          ;
4404          ;
4405          ;*****
4406          ;
4407 050452 013701 054140      MOV     T25BFR+6,R1      ;PICK UP XSTO
4408 050456 010102          MOV     R1,R2           ;SET UP EXPECTED
4409 050460 052702 000002      BIS     @BIT1,R2        ;SET BOT BIT IN EXPECTED
4410 050464 020102          CMP     P1,R2           ;DOES EXP = REC'D
4411 050466 001406          BEQ     40$            ;BR, IF EQUAL (OK)
4412 050470 005237 002212      INC     FATFLG          ;BUMP COUNT
4416 050474          ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      050474 104456          TRAP   C$ERHRD
      050476 001013          .WORD  523
      050500 054435          .WORD  T25BOT
      050502 015564          .WORD  EXPREC
4417 050504          40$:  CKLOOP          ;LOOP IF SELECTED          TRAP   C$CLP1
      050504 104406          ;NUMBER OF RECORDS TO SPACE OVER
4418 050506 012737 000001 054242  MOV     @000001,T25RB
4419          ;*****
4420          ;
4421          ;
4422          ;SPACE FORWARD,ACK,CVC=1 COMMAND
4423          ;
4424          ;*****
4425          ;
4426 050514 012737 140010 054240  MOV     @140010,T25PK3 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4427 050522 012704 054240      MOV     @T25PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
4428 050526          65$:
4429 050526 010465 000000      MOV     R4,T5DB(R5)    ;ISSUE COMMAND
4430 050532 004737 016340      JSR     PC,WAITF       ;WAIT FOR SSR TO SET
4431 050536 016501 000002      MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
4432 050542 012702 000200      MOV     @SSR,R2       ;SET UP EXPECTED
4433 050546 020102          CMP     R1,R2         ;ARE THEY EQUAL
4434 050550 001411          BEQ     75$           ;BR, IF OK
4435 050552 032701 000004      BIT     @BIT2,R1      ;CHECK FOR TAPE STATUS ALERT
4436 050556 001006          BNE     75$           ;BR, IF TSA IS SET (SUSPECT IS EOT)
4437 050560 005237 002212      INC     FATFLG        ;BUMP COUNT
4441 050564          ERRHRD  ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
      050564 104456          TRAP   C$ERHRD
      050566 001014          .WORD  524
      050570 054355          .WORD  T25WDE
      050572 015564          .WORD  EXPREC
4442 050574          75$:  CKLOOP          ;LOOP IF SELECTED          TRAP   C$CLP1
      050574 104406          ;
4443 050576          120$:
4444          ;*****
4445          ;
4446          ;
4447          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4448          ;
4449          ;*****

```





```

4551 ;*****
4552 ;
4553 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4554 ;
4555 ;*****
4556
4557 051100 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
4558 051104 103407                   BCS    30$                ;BR, IF NO PROBLEM
4559 051106 010001                   MOV    R0,R1              ;SAVE TSSR
4560 051110 005237 002212          INC    FATFLG             ;BUMP COUNT
4564 051114                   ERRHRD  ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                051114 104456                   TRAP   C$ERHRD
                                051116 001022                   .WORD 530
                                051120 055245                   .WORD T25RWN
                                051122 012136                   .WORD PKTSSR
4565 051124 30$: CKLOOP                ;LOOP IF SELECTED
                                051124 104406                   TRAP   C$CLP1
4566
4567 ;*****
4568 ;
4569 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4570 ;
4571 ;*****
4572
4573 051126 013701 054140          MOV    T25BFR+6,R1       ;PICK UP XSTO
4574 051132 010102                   MOV    R1,R2              ;SET UP EXPECTED
4575 051134 052702 000002          BIS    #BIT1,R2          ;SET BOT BIT IN EXPECTED
4576 051140 020102                   CMP    R1,R2              ;DOES EXP = REC'D
4577 051142 001406                   BEQ    40$                ;BR, IF EQUAL (OK)
4578 051144 005237 002212          INC    FATFLG             ;BUMP COUNT
4582 051150                   ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                051150 104456                   TRAP   C$ERHRD
                                051152 001023                   .WORD 531
                                051154 054435                   .WORD T25BOT
                                051156 015564                   .WORD EXPREC
4583 051160 40$: CKLOOP                ;LOOP IF SELECTED
                                051160 104406                   TRAP   C$CLP1
4584
4585 ;*****
4586 ;
4587 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
4588 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
4589 ;
4590 ;*****
4591
4592 051162 012703 000001          MOV    #000001,R3        ;NUMBER OF RECORDS TO SPACE FORWARD
4593 051166 004737 010556          JSR    PC,SPACE          ;CALL SPACE COMMAND
4594 051172 103410                   BCS    50$                ;CHECK FOR ERROR
4595 051174 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
4596 051200 005237 002212          INC    FATFLG             ;BUMP COUNT
4600 051204                   ERRHRD  ERRNO,T25WDE,SFFMSG ;SPACE FORWARD FAILED
                                051204 104456                   TRAP   C$ERHRD
                                051206 001024                   .WORD 532
                                051210 054355                   .WORD T25WDE
                                051212 012172                   .WORD SFFMSG
4601 051214 50$: CKLOOP                ;LOOP IF SELECTED
                                051214 104406                   TRAP   C$CLP1

```

```

4602 051216 012737 000001 054242      MOV      #1,T25RB          ;NUMBER OF RECORDS TO SPACE OVER
4603
4604      ;*****
4605      ;
4606      ;SPACE REVERSE,ACK,CVC=1 COMMAND
4607      ;
4608      ;*****
4609
4610 051224 012737 140410 054240      MOV      #140410,T25PK3    ;SPACE REVERSE,ACK,CVC=1 COMMAND
4611 051232 012704 054240      MOV      #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
4612 051236      65$:
4613 051236 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
4614 051242 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
4615 051246 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
4616 051252 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
4617 051256 020102      CMP      R1,R2             ;ARE THEY EQUAL
4618 051260 001406      BEQ      75$               ;BR, IF OK
4619 051262 005237 002212      INC      FATFLG            ;BUMP COUNT
4623 051266      ERRHRD  ERRNO,T25WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C$ERHRD
                                .WORD    533
                                .WORD    T25WDE
                                .WORD    PKTSSR
4624 051276      75$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
4625 051300      120$:
4626 051300 012703 000400      MOV      #256.,R3          ;RECORD SIZE
4627 051304 013737 003114 054242      MOV      FREE,T25RB        ;STARTING READ BUFFER ADDRESS
4628
4629      ;*****
4630      ;
4631      ;READ DATA,ACK,CVC=1 COMMAND
4632      ;
4633      ;*****
4634
4635 051312 012737 140001 054240      MOV      #140001,T25PK3    ;READ DATA,ACK,CVC=1 COMMAND
4636 051320 012704 054240      MOV      #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
4637 051324 010337 054246      MOV      R3,T25SZ          ;SET UP RECORD SIZE IN PACKET
4638 051330 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
4639 051334 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
4640 051340 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
4641 051344 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
4642 051350 020102      CMP      R1,R2             ;ARE THEY EQUAL
4643 051352 001406      BEQ      170$              ;BR, IF OK
4644 051354 005237 002212      INC      FATFLG            ;BUMP COUNT
4648 051360      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C$ERHRD
                                .WORD    534
                                .WORD    RDERR
                                .WORD    PKTSSR
4649 051370      170$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
4650 051372 017701 131516      MOV      @FREE,R1          ;GET FIRST WORD FROM BUFFER
4651 051376 012702 000000      MOV      #0,R2             ;SET UP EXPECTED
4652 051402 020102      CMP      R1,R2             ;WAS RECORD NUMBERED 1
4653 051404 001406      BEQ      200$              ;BR, IF CORRECT RECORD
4654 051406 005237 002212      INC      FATFLG            ;BUMP COUNT

```

```
4658 051412          ERRHRD  ERRNO,T25WNG,EXPREC      ; SHOULD HAVE BEEN RECORD NUMBER 1
      051412  104456
      051414  001027
      051416  054645
      051420  015564
4659 051422          200$:  CKLOOP
      051422  104406
4660 051424          ENDSUB                      ; >>>>>>>>>>>>>>>>>>>>>>>>>>>>
      051424
      051424  104403
4661 051426  023727  002212  000017          CMP      FATFLG,#15.          ; IS ERROR COUNT AT 25
4662 051434  103402                                     BLJ      999$                ; BR, IF LESS THAN 25
4663 051436  004737  017272          JSR      PC,CKDROP          ; TRY TO DROP THE UNIT
4664 051442
4665
4666
4667
4668
4669
4670
4671
4672
4673
4674
4675
4676
4677
4678
4679 051442          BGNSUB                      ; >>>>>>>>>>>>>>>>>>>>>>>>>>>>
      051442
      051442  104402
4680 051444  004737  055456          JSR      PC,T25REST        ; SET COMMAND PACKET
4681 051450  004737  055550          JSR      PC,T25RT2        ; SET UP OTHER COMMAND PACKET
4682 051454  004737  055612          JSR      PC,T25RT3        ; SET UP OTHER COMMAND PACKET
4683 051460  013737  054270  054266          MOV      T25CNT,T25CN2    ; HOLD NUMBER OF RECORDS
4684 051466  162737  000002  054266          SUB      #2,T25CN2        ; ACTUAL NUMBER OF RECORDS ON TAPE
4685 051474  013737  054270  054272          MOV      T25CNT,T25DLY    ; SET UP REWIND DELAY COUNTER
4686
4687
4688
4689
4690
4691
4692
4693 051502  004737  016064          10$:  JSR      PC,SOFINIT    ; DO INITIALIZE ON CONTROLLER
4694 051506  103427                                     BCS     20$                ; BR IF INIT WAS OK
4695 051510                                     DELAY   250                ; WAIT ABOUT .25 SECONDS
      051510  012727  000250
      051514  000000
      051516  013727  002116
      051522  000000
      051524  005367  177772
      051530  001375
      051532  005367  177756
      051536  001367
4696 051540  005337  054272          DEC      T25DLY            ; DEC COUNTER
4697 051544  001356          BNE     10$                ; BR, IF MORE LOOPS REQUIRED
                                     TRAP  C$ERHRD
                                     .WORD 535
                                     .WORD T25WNG
                                     .WORD EXPREC
                                     TRAP  C$CLP1
                                     ; >>>>>>>>>>>>>>>>>>>>>>>>>>>>
                                     L10075:
                                     TRAP  C$ESUB
                                     ; *****
                                     ; ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
                                     ; *****
                                     MOV      #250,(PC)+
                                     .WORD   0
                                     MOV      L$DLY,(PC)+
                                     .WORD   0
                                     DEC      -6(PC)
                                     BNE     -.4
                                     DEC      -22(PC)
                                     BNE     -.20
```



```

4698 051546 016501 000002      MOV    TSSR(R5),R1          ;CONTENTS OF TSSR REGISTER
4699 051552 005237 002212      INC    FATFLG              ;BUMP COUNT
4703 051556                      ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   536
                                .WORD   SFIERR
                                .WORD   SFIMSG
4704 051566                      20$:
4705 051566 013737 002172 054130  MOV    UNITN,T25DSW        ;SET UP UNIT NUMBER
4706 051574 012704 054110      MOV    @T25PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
4707
4708      ;*****
4709      ;
4710      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4711      ;
4712      ;*****
4713
4714 051600 004737 010752      JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
4715 051604 103407                      BCS    25$                ;BR, IF COMMAND ISSUED OK
4716 051606 005237 002212      INC    FATFLG            ;BUMP COUNT
4720 051612 010001                      MOV    R0,R1              ;SAVE CONTENTS OF TSSR
4721 051614                      ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                                TRAP    C$ERHRD
                                .WORD   537
                                .WORD   WRTMSG
                                .WORD   SFIMSG
4722 051624                      25$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C$CLP1
4723 051624 104406
4724      ;*****
4725      ;
4726      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4727      ;
4728      ;*****
4729
4730 051626 004737 011104      JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
4731 051632 103407                      BCS    30$                ;BR, IF NO PROBLEM
4732 051634 010001                      MOV    R0,R1              ;SAVE TSSR
4733 051636 005237 002212      INC    FATFLG            ;BUMP COUNT
4737 051642                      ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   538
                                .WORD   T25RWN
                                .WORD   PKTSSR
4738 051652                      30$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C$CLP1
4739 051654 013701 054266      MOV    T25CN2,R1          ;NUMBER OF RECORDS ON TAPE
4740 051660 012702 177776      MOV    @65534.,R2        ;MAX IT CAN SPACE OVER
4741 051664 020201                      CMP    R2,R1              ;WHICH VALUE CAN WE USE
4742 051666 003002                      BGT    46$                ;BR, IF # WRITTEN > 64K
4743 051670 010103                      MOV    R1,R3              ;# WRITTEN CAN BE USED
4744 051672 000401                      BR     47$                ;MOVE ON
4745 051674 010203                      46$:  MOV    R2,R3          ;USE MAX NUMBER
4746 051676 162703 000001      47$:  SJB    @1,R3          ;DON'T GO ALL THE WAY YET
4747 051702 010337 054242      MOV    R3,T25RB          ;NUMBER OF RECORDS TO SPACE OVER
4748
4749      ;*****

```

```

4750 ;
4751 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4752 ;
4753 ;*****
4754
4755 051706 012737 140010 054240      MOV      #140010,T25PK3      ;SPACE FORWARD,ACK,CVC=1 COMMAND
4756 051714 012704 054240      MOV      #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
4757 051720
4758 051720 013737 054266 054272 65$:  MOV      T25CN2,T25DLY      ;NUMBER OF RECORDS USED AS DELAY COUNTER
4759 051726 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
4760 051732 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
4761 051736 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
4762 051742 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
4763 051746 020102      CMP      R1,R2            ;ARE THEY EQUAL
4764 051750 001425      BEQ      75$              ;BR, IF OK
4765 051752      DELAY   250              ;DELAY .25 SECONDS
      MOV      #250,(PC)+
      .WORD   0
      MOV      L$DLY,(PC)+
      .WORD   0
      DEC     -6(PC)
      BNE     -.4
      DEC     -22(PC)
      BNE     .-20
4766 052002 005337 054272      DEC     T25DLY            ;BUMP DOWN COUNTER
4767 052006 001351      BNE     67$              ;BR, IF NOT AT END OF DELAY
4768 052010 005237 002212      INC     FATFLG           ;BUMP COUNT
4772 052014      ERRHRD ERRNO,T25WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      TRAP   C$ERRHRD
      .WORD  539
      .WORD  T25WDE
      .WORD  PKTSSR
      052014 104456
      052016 001033
      052020 054355
      052022 012136
4773 052024      75$:  CKLOOP                ;LOOP IF SELECTED
      TRAP   C$CLP1
4774 052026 012703 010000      MOV     #4096.,R3        ;RECORD SIZE
4775 052032 013737 003114 054242      MOV     FREE,T25RB       ;STARTING READ BUFFER ADDRESS
4776
4777 ;*****
4778 ;
4779 ;READ DATA,ACK COMMAND
4780 ;
4781 ;*****
4782
4783 052040 012737 100001 054240 165$: MOV     #100001,T25PK3     ;READ DATA,ACK COMMAND
4784 052046 012704 054240      MOV     #T25PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
4785 052052 010337 054246      MOV     R3,T25SZ         ;SET UP RECORD SIZE IN PACKET
4786 052056 010465 000000      MOV     R4,TSDB(R5)      ;ISSUE COMMAND
4787 052062 004737 016340      JSR     PC,WAITF         ;WAIT FOR SSR TO SET
4788 052066 016501 000002      MOV     TSSR(R5),R1      ;GET TSSR CONTENTS
4789 052072 012702 000200      MOV     #SSR,R2         ;SET UP EXPECTED
4790 052076 020102      CMP     R1,R2           ;ARE THEY EQUAL
4791 052100 001411      BEQ     170$            ;BR, IF OK
4792 052102 032701 000004      BIT     #BIT2,R1        ;CHECK FOR TAPE STATUS ALERT
4793 052106 001006      BNE     170$            ;IF SET ALL IS WELL
4794 052110 005237 002212      INC     FATFLG           ;BUMP COUNT
4798 052114      ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      TRAP   C$ERRHRD
      052114 104456

```



```

052246 000000 .WORD 0
052250 013727 002116 MOV L$DLY,(PC)
052254 000000 .WORD 0
052256 005367 177772 DEC -6(PC)
052262 001375 BNE -.4
052264 005367 177756 DEC -22(PC)
052270 001367 BNE .-20
4845 052272 005337 054272 DEC T25DLY ;DEC COUNTER
4846 052276 001356 BNE 10$ ;BR, IF MORE LOOPS REQUIRED
4847 052300 016501 000002 MOV TSSR(R5),R1 ;CONTENTS OF TSSR REGISTER
4848 052304 005237 002212 INC FATFLG ;BUMP COUNT
4852 052310 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
052310 104455 TRAP C$ERDF
052312 001036 .WORD 542
052314 003650 .WORD SFIERR
052316 012124 .WORD SFIMSG
4853 052320 013737 002172 054130 20$: MOV UNITN,T25DSW ;SET UP UNIT NUMBER
4854
4855 052326 012704 054110 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4856
4857 ;*****
4858 ;
4859 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4860 ;
4861 ;*****
4862
4863 052332 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4864 052336 103407 BCS 25$ ;BR, IF COMMAND ISSUED OK
4865 052340 005237 002212 INC FATFLG ;BUMP COUNT
4869 052344 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4870 052346 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
052346 104456 TRAP C$ERHRD
052350 001037 .WORD 543
052352 005054 .WORD WRTMSG
052354 012124 .WORD SFIMSG
4871 052356 25$: CKLOOP ;LOOP IF SELECTED
052356 104406 TRAP C$CLP1
4872
4873 ;*****
4874 ;
4875 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
4876 ;
4877 ;*****
4878
4879 052360 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4880 052364 103407 BCS 30$ ;BR, IF NO PROBLEM
4881 052366 010001 MOV R0,R1 ;SAVE TSSR
4882 052370 005237 002212 INC FATFLG ;BUMP COUNT
4886 052374 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
052374 104456 TRAP C$ERHRD
052376 001040 .WORD 544
052400 055245 .WORD T25RWN
052402 012136 .WORD PKTSSR
4887 052404 30$: CKLOOP ;LOOP IF SELECTED
052404 104406 TRAP C$CLP1
4888
4889 ;*****

```

```

4890
4891 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
4892 ;
4893 ;*****
4894
4895 052406 013701 054140      MOV      T25BFR+6,R1      ;PICK UP XSTO
4896 052412 010102           MOV      R1,R2          ;SET UP EXPECTED
4897 052414 052702 000002     BIS      @BIT1,R2       ;SET BOT BIT IN EXPECTED
4898 052420 020102           CMP      R1,R2          ;DOES EXP = REC'D
4899 052422 001406           BEQ      40$           ;BR, IF EQUAL (OK)
4900 052424 005237 002212     INC      FATFLG         ;BUMP COUNT
4904 052430           ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    545
                                .WORD    T25BOT
                                .WORD    EXPREC
                                TRAP      C$CLP1
                                .WORD    545
                                .WORD    T25BOT
                                .WORD    EXPREC
4905 052440           40$:   CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    545
                                .WORD    T25BOT
                                .WORD    EXPREC
4906 052442 013701 054266     MOV      T25CN2,R1      ;NUMBER OF RECORDS ON TAPE
4907 052446 012702 177776     MOV      @65534.,R2     ;MAX IT CA: SPACE OVER
4908 052452 020201           CMP      R2,R1          ;WHICH VALUE CAN WE USE
4909 052454 003002           BGT      46$           ;BR, IF # WRITTEN > 64K
4910 052456 010103           MOV      R1,R3          ;# WRITTEN CAN BE USED
4911 052460 000401           BR       47$           ;MOVE ON
4912 052462 010203           46$:   MOV      R2,R3          ;USE MAX NUMBER
4913 052464           47$:
4914 052464 010337 054242     MOV      R3,T25RB       ;NUMBER OF RECORDS TO SPACE OVER
4915
4916 ;*****
4917 ;
4918 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4919 ;
4920 ;*****
4921
4922 052470 012737 140010 054240  MOV      @140010,T25PK3 ;SPACE FORWARD,ACK,CVC=1 COMMAND
4923 052476 012704 054240      MOV      @T25PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
4924 052502 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
4925 052506 013737 054266 054272  MOV      T25CN2,T25DLY ;SET UP DELAY COUNTER
4926 052514 004737 016340 48$:   JSR      PC,WAITF       ;WAIT FOR SSR TO SET
4927 052520 016501 000002     MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
4928 052524 012702 000200     MOV      @SSR,R2        ;SET UP EXPECTED
4929 052530 020102           CMP      R1,R2          ;ARE THEY EQUAL
4930 052532 001425           BEQ      50$           ;BR, IF OK
4931 052534           DELAY   250           ;WAIT .25 SECONDS
                                MOV      #250,(PC)+
                                .WORD    0
                                MOV      L$DLY,(PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE     .-4
                                DEC      -22(PC)
                                BNE     .-20
4932 052564 005337 054272     DEC      T25DLY         ;DEC THE DELAY COUNTER
4933 052570 001351           BNE     48$           ;BR, IF COUNTER HASN'T EXPIRED
4934 052572 005237 002212     INC      FATFLG         ;BUMP COUNT
4938 052576           ERRHRD  ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C$ERHRD
                                .WORD    545
                                .WORD    T25BOT
                                .WORD    EXPREC
                                TRAP      C$ERHRD
                                .WORD    545
                                .WORD    T25BOT
                                .WORD    EXPREC

```

```

052600 001042 .WORD 546
052602 054355 .WORD T25WDE
052604 015564 .WORD EXPREC
4939 052606 504: CKLOOP TRAP C$CLP1
052606 104406
4940 052610 013701 054266 MOV T25CN2,R1 ;NUMBER OF RECORDS ON TAPE
4941 052614 012702 177776 MOV #65534.,R2 ;MAX IT CAN SPACE OVER
4942 052620 020201 CMP R2,R1 ;WHICH VALUE CAN WE USE
4943 052622 003002 BGT 554 ;BR, IF # WRITTEN > 64K
4944 052624 010103 MOV R1,R3 ;# WRITTEN CAN BE USED
4945 052626 000401 BR 604 ;MOVE ON
4946 052630 010203 554: MOV R2,R3 ;USE MAX NUMBER
4947 052632 162703 000001 604: SUB #1,R3 ;DON'T GO ALL THE WAY YET
4948 052636 010337 054242 MOV R3,T25RB ;NUMBER OF RECORDS TO SPACE OVER
4949
4950 ;*****
4951 ;
4952 ;SPACE REVERSE,ACK,CVC-1 COMMAND
4953 ;
4954 ;*****
4955
4956 052642 012737 140410 054240 MOV #140410,T25PK3 ;SPACE REVERSE,ACK,CVC-1 COMMAND
4957 052650 012704 054240 MOV #T25PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
4958 052654 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
4959 052660 013737 054266 054272 MOV T25CN2,T25CLY ;SET UP COUNTER
4960 052666 004737 016340 704: JSR PC,WAITF ;WAIT FOR SSR TO SET
4961 052672 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4962 052676 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
4963 052702 020102 CMP R1,R2 ;ARE THEY EQUAL
4964 052704 001425 BEQ 754 ;BR, IF OK
4965 052706 DELAY 250 ;WAIT ABOUT .25 SECONDS
052706 012727 000250 MOV #250,(PC)-
052712 000000 .WORD 0
052714 013727 002116 MOV L$DLY,(PC)-
052720 000000 .WORD 0
052722 005367 177772 DEC -6(PC)
052726 001375 BNE .-4
052730 005367 177756 DEC -22(PC)
052734 001367 BNE .-20
4966 052736 005337 054272 DEC T25DLY ;BUMP COUNTER DOWN
4967 052742 001351 BNE 704 ;BR, IF COUNTER HASN'T EXPIRED
4968 052744 005237 002212 INC FATFLG ;BUMP COUNT
4972 052750 ERRHRD ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
052750 104456 TRAP C$ERRRD
052752 001043 .WORD 547
052754 054355 .WORD T25WDE
052756 015564 .WORD EXPREC
4973 052760 754: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
052760 104406
4974 052762 012703 010000 MOV #4096.,R3 ;RECORD SIZE
4975 052766 013737 003114 054242 MOV FREE,T25RB ;STARTING READ BUFFER ADDRESS
4976
4977 ;*****
4978 ;
4979 ;READ DATA,ACK COMMAND
4980 ;
4981 ;*****

```



```

5031 053150 004737 055550          JSR    PC,T25RT2          ;SET UP OTHER COMMAND PACKET
5032 053154 004737 055612          JSR    PC,T25RT3          ;SET UP OTHER COMMAND PACKET
5033
5034          ;*****
5035          ;
5036          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5037          ;
5038          ;*****
5039
5040 053160 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
5041 053164 103407                  BCS    20$              ;BR IF INIT WAS OK
5042 053166 005237 002212          INC    FATFLG           ;BUMP COUNT
5046 053172 010001                  MOV    R0,R1            ;CONTENTS OF TSSR REGISTER
5047 053174                  ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                    104455                  TRAP   C$ERDF
                    053176 001046                  .WORD 550
                    053200 003650                  .WORD SFIERR
                    053202 012124                  .WORD SFIMSG
5048 053204 013737 002172 054130 20$: MOV    UNITN,T25DSW        ;SET UP UNIT NUMBER
5049
5050 053212 012704 054110          MOV    @T25PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
5051
5052          ;*****
5053          ;
5054          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5055          ;
5056          ;*****
5057
5058 053216 004737 010752          JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
5059 053222 103407                  BCS    25$              ;BR, IF COMMAND ISSUED OK
5060 053224 005237 002212          INC    FATFLG           ;BUMP COUNT
5064 053230 010001                  MOV    R0,R1            ;SAVE CONTENTS OF TSSR
5065 053232                  ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
                    104456                  TRAP   C$ERHRD
                    053234 001047                  .WORD 551
                    053236 005054                  .WORD WRTMSG
                    053240 012124                  .WORD SFIMSG
5066 053242                  25$:  CKLOOP                ;LOOP IF SELECTED
                    104406                  TRAP   C$CLP1
5067
5068          ;*****
5069          ;
5070          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5071          ;
5072          ;*****
5073
5074 053244 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
5075 053250 103407                  BCS    30$              ;BR, IF NO PROBLEM
5076 053252 010001                  MOV    R0,R1            ;SAVE TSSR
5077 053254 005237 002212          INC    FATFLG           ;BUMP COUNT
5081 053260                  ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                    104456                  TRAP   C$ERHRD
                    053262 001050                  .WORD 552
                    053264 055245                  .WORD T25RWN
                    053266 012136                  .WORD PKTSSR
5082 053270                  30$:  CKLOOP                ;LOOP IF SELECTED
                    104406                  TRAP   C$CLP1

```



```

5083
5084
5085
5086
5087
5088
5089
5090 053272 013701 054140      MOV      T25BFR+6,R1      ;PICK UP XSTO
5091 053276 010102              MOV      R1,R2           ;SET UP EXPECTED
5092 053300 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
5093 053304 020102              CMP      R1,R2           ;DOES EXP = REC'D
5094 053306 001406              BEQ      40$             ;BR, IF EQUAL (OK)
5095 053310 005237 002212      INC      FATFLG          ;BUMP COUNT
5099 053314              ERRHRD   ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    553
                                .WORD    T25BOT
                                .WORD    EXPREC
                                TRAP      C$CLP1
5099 053314 104456
5099 053316 001051
5099 053320 054435
5099 053322 015564
5100 053324              40$:   CKLOOP           ;LOOP IF SELECTED
5100 053324 104406
5101 053326 012737 000001 054242      MOV      @1,T25RB        ;NUMBER OF RECORDS TO SPACE OVER
5102
5103
5104
5105
5106
5107
5108
5109 053334 012737 100410 054240      MOV      @100410,T25PK3  ;SPACE REVERSE,ACK COMMAND
5110 053342 012704 054240              MOV      @T25PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
5111 053346              65$:   MOV      R4,T5DB(R5)    ;ISSUE COMMAND
5112 053346 010465 000000      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
5113 053352 004737 016340      MOV      T5SR(R5),R1    ;GET T5SR CONTENTS
5114 053356 016501 000002      MOV      @5SR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
5115 053362 012702 100206      CMP      R1,R2          ;ARE THEY EQUAL
5116 053366 020102              BEQ      75$            ;BR, IF OK
5117 053370 001406              INC      FATFLG          ;BUMP COUNT
5118 053372 005237 002212      ERRHRD   ERRNO,T25WDE,PKTSSR ;T5SR INCORRECT AFTER READ DATA
5122 053376              TRAP      C$ERHRD
                                .WORD    554
                                .WORD    T25WDE
                                .WORD    PKT5SR
5122 053376 104456
5122 053400 001052
5122 053402 054355
5122 053404 012136
5123 053406              75$:   CKLOOP           ;LOOP IF SELECTED
5123 053406 104406              TRAP      C$CLP1
5124
5125
5126
5127
5128
5129
5130
5131 053410 013701 054140      MOV      T25BFR+6,R1    ;GET XSTO STATUS WORD
5132 053414 010102              MOV      R1,R2           ;SET UP EXPECTED
5133 053416 052702 002000      BIS      @BIT10,R2      ;SET THE NEF BIT
5134 053422 020102              CMP      R1,R2          ;ARE THEY EQUAL
5135 053424 001406              BEQ      170$           ;BR, IF EQUAL (GOOD)

```



```

5186 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
5187 ;
5188 ;*****
5189
5190 053536 004737 010752 JSR PC,WRTPHR ;ISSUE WRITE CHARACTERISTICS
5191 053542 103407 BCS 25$ ;BR, IF COMMAND ISSUED OK
5192 053544 005237 002212 INC FATFLG ;BUMP COUNT
5196 053550 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
5197 053552 ERRHRD ERRNO,WRTPHR,SFIMSG ;WRITE CHARACTERISTICS FAILED
053552 104456 TRAP C$ERRHRD
053554 001055 .WORD 557
053556 005054 .WORD WRTPHR
053560 012124 .WORD SFIMSG
5198 053562 25$: CKLOOP ;LOOP IF SELECTED
053562 104406 TRAP C$CLP1
5199
5200 ;*****
5201 ;
5202 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5203 ;
5204 ;*****
5205
5206 053564 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
5207 053570 103407 BCS 30$ ;BR, IF NO PROBLEM
5208 053572 010001 MOV R0,R1 ;SAVE TSSR
5209 053574 005237 002212 INC FATFLG ;BUMP COUNT
5213 053600 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
053600 104456 TRAP C$ERRHRD
053602 001056 .WORD 558
053604 055245 .WORD T25RWN
053606 012136 .WORD PKTSSR
5214 053610 30$: CKLOOP ;LOOP IF SELECTED
053610 104406 TRAP C$CLP1
5215
5216 ;*****
5217 ;
5218 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
5219 ;
5220 ;*****
5221
5222 053612 013701 054140 MOV T25BFR+6,R1 ;PICK UP XST0
5223 053616 010102 MOV R1,R2 ;SET UP EXPECTED
5224 053620 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
5225 053624 020102 CMP R1,R2 ;DOES EXP = REC'D
5226 053626 001406 BEQ 40$ ;BR, IF EQUAL (OK)
5227 053630 005237 002212 INC FATFLG ;BUMP COUNT
5231 053634 ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
053634 104456 TRAP C$ERRHRD
053636 001057 .WORD 559
053640 054435 .WORD T25BOT
053642 015564 .WORD EXPREC
5232 053644 40$: CKLOOP
053644 104406 TRAP C$CLP1
5233 053646 012737 000001 054242 MOV #1,T25RB ;NUMBER OF RECORDS TO SPACE OVER
5234
5235 ;*****
5236 ;

```





```

5347 054251      200          T25BS1: .BYTE 200          ;BSEL1 AREA
5348 054252 000000          T25S2: .WORD 0           ;SEL 2 AREA
5349 054254 000000          T25S3: .WORD 0           ;DATA AREA
5350
5351
5352
5353
5354
5355 054256 100005          T25RN: .WORD 100005      ;READ DATA (NEXT)
5356 054260 100405          T25WDR: .WORD 100405     ;READ DATA RETRY
5357 054262 102005          T25CON: .WORD 102005     ;WRITE CONTINUOUS
5358 054264 177777          .WORD 177777            ;END OF DATA
5359
5360 054266 000000          T25CN2: .WORD 0          ;COUNTER FOR RECORDS
5361 054270 000000          T25CNT: .WORD 0          ;COUNTER FOR RECORDS
5362 054272 000000          T25DLY: .WORD 0          ;COUNTER FOR RECORDS
5363
5364
5365
5366
5367
5368 054274      127      122      111 T25SSR: .ASCIZ 'WRITE SUBSYSTEM Miscellaneous Read Status Failed'
5369 054355      124      123      123 T25WDE: .ASCIZ 'TSSR Not Correct After POSITION (SPACE) Command'
5370 054435      124      141      160 T25BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
5371 054502      124      123      123 T25TM: .ASCIZ 'TSSR Not Correct After POSITION (Space Command) Reject'
5372 054571      127      162      151 T25NET: .ASCIZ 'Write Tape, Status Alert, But No EOT Sensed'
5373 054645      123      160      141 T25WNG: .ASCIZ 'Space Forward Failed To Position On Correct Record'
5374 054730      123      160      141 T25BNC: .ASCIZ 'Space Forward, From BOT, Failed To Clear BOT Indication'
5375 055020      123      160      141 T25WNH: .ASCIZ 'Space Reverse Failed To Position On Correct Record'
5376 055103      123      160      141 T25NEF: .ASCIZ 'Space Reverse, At BOT, Failed To Set NEF (XST0)'
5377 055163      123      160      141 T25RIB: .ASCIZ 'Space Reverse, Into BOT, Failed To Set RIB (XST3)'
5378 055245      122      145      167 T25RWV: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
5379 055314      104      162      151 T25OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
5380 055367      124      123      123 T25WDC: .ASCIZ 'TSSR Not Correct After READ DATA Command'
5381 055440      123      160      141 TST25ID: .ASCIZ 'Space Records'
5382
5383
5384
5385
5386
5387
5388
5389
5390 055456
5391 055456
5392 055462 012701 054110          SAVREG
5393 055466 012721 100004          MOV #T25PACKET,R1      ;SAVE THE REGISTERS
5394 055472 012721 054120          MOV #100004,(R1)+     ;START OF THE PACKET
5395 055476 005021          MOV #T25DATA,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK
5396 055500 012721 000012          CLR (R1)+             ;ADDRESS OF CHARAISTICS DATA BLOCK
5397 055504 012721 054132          MOV #10.,(R1)+       ;EXTENDED ADDRESS
5398 055510 005021          MOV #T25BFR,(R1)+   ;SIZE OF DATA BLOCK IN BYTES
5399 055512 012721 000024          CLR (R1)+             ;ADDRESS OF MESSAGE BUFFER
5400 055516 005021          MOV #20.,(R1)+      ;LENGTH OF MESSAGE BUFFER
5401 055520 012711 000000          CLR (R1)+
5402 055524 012702 000030          MOV #0,(R1)          ;SELECT DRIVE ZERO
5403 055530 012762 177777 054132 64# : MOV #24.,R2            ;NUMBER OF LOCATIONS TO BE CLEARED
                                          MOV #177777,T25BFR(R2) ;ALL ONES TO MESSAGE BUFFER

```

```

5404 055536 005742          TST      -(R2)          ;NEXT LOCATION
5405 055540 022702 000000  CMP      #0,R2         ;IS R2 AT ZERO YET
5406 055544 001371          BNE      64$          ;KEEP GOING UNTIL DONE
5407 055546 000207          RTS      PC           ;RETURN
5408
5409 055550          T25RT2: SAVREG        ;SAVE THE REGISTERS
5410 055550          MOV      #T25PK2,R1   ;START OF THE PACKET
5411 055554 012701 054220  MOV      #100006,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK.
5412 055560 012721 100006  MOV      #T25BF2,(R1)+ ;ADDRESS OF DATA BLOCK
5413 055564 012721 054250  CLR      (R1)+        ;EXTENDED ADDRESS
5414 055570 005021          MOV      #6.,(R1)+   ;SIZE OF DATA BLOCK IN BYTES
5415 055572 012721 000006  CLR      (R1)+
5416 055576 005021          MOV      #T25BF2,R1  ;POINT TO DATA SEL AREA
5417 055600 012701 054250  CLR      (R1)+
5418 055604 005021          CLR      (R1)+
5419 055606 005011          RTS      PC           ;RETURN
5420 055610 000207          T25RT3: SAVREG        ;SAVE THE REGISTERS
5421 055612          MOV      #T25PK3,R1  ;START OF THE PACKET
5422 055612          MOV      #0,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK.
5423 055616 012701 054240  MOV      #0,(R1)+   ;ADDRESS OF DATA BLOCK
5424 055622 012721 000000  CLR      (R1)+        ;EXTENDED ADDRESS
5425 055626 012721 000000  MOV      #0,(R1)+   ;SIZE OF DATA BLOCK IN BYTES
5426 055632 005021          RTS      PC           ;RETURN
5427 055634 012721 000000  ENDTST
5428 055640 000207
5429 055642
5430 055642 104401          L10071: TRAP      C$ETST

```

```

5430
5431          .SBTTL TEST 6: REREADS
5432          ;+
5433          ;
5434          ;THIS TEST VERIFIES THAT THE REREAD PREVIOUS AND REREAD NEXT
5435          ;COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN
5436          ;DATA BUFFER BOUNDRIES, RECORD SIZES (UP TO 64K BYTES IF MEMORY
5437          ;SPACE IS AVAILIABLE), AND BYTE-SWAP (SWP) AND OPPOSITE (OPP)
5438          ;CONRTOL ARE USED. ALSO TESTED ARE PROPER TERMINATIONS ON
5439          ;EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD
5440          ;LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES,
5441          ;AND DATA BUFFERS IN NONEXISTENT MEMORY.
5442          ;
5443          ;
5444          ;THE TEST CONSISTS OF THE FOLLOWING 15 SUBTESTS
5445          ;
5446          ;
5447          ;
5448          ;-
5449          BGNTST
5450 055644 012737 006446 002170  MOV      #EPRT2,EPRTSW ;SECONDARY ERROR MESSAGE
5451 055652 004737 017364          JSR      PC,KTOFF     ;DON'T NEED KT11
5452 055656 012700 075027          MOV      #TST26ID,R0 ;ASCII MESSAGE TO IDENTIFY TEST
5453 055662 004737 016600          JSR      PC,TSTSETUP ;DO INITIAL TEST SETUP
5454 055666 012737 000005 002206  MOV      #5,LOOPCNT   ;PERFORM 5 ITERATIONS
5455 055674 004737 021322          JSR      PC,MEMCK    ;CHECK FOR MEMORY
5456 055700 005037 003126          CLR      NXMFLG      ;SET FLAG
5457 055704 005037 072276          CLR      T26CNT     ;CLEAR TAPE RECORD COUNTER

```





```

5508 056024 012704 072120      MOV      #T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
5509
5510      ;*****
5511      ;
5512      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5513      ;
5514      ;*****
5515
5516 056030 004737 010752      JSR      PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
5517 056034 103407      BCS     26$                ;BR, IF COMMAND ISSUED OK
5518 056036 005237 002212      INC     FATFLG             ;BUMP COUNT
5522 056042 010001      MOV     R0,R1              ;SAVE CONTENTS OF TSSR
5523 056044      ERRHRD  ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTICS FAILED
5523 056044 104456      TRAP   C$ERHRD            ;TRAP
5523 056046 001132      .WORD  602                ;.WORD
5523 056050 005054      .WORD  WRTMSG              ;.WORD
5523 056052 012124      .WORD  SFMSG              ;.WORD
5524 056054      26$:   CKLOOP              ;LOOP IF SELECTED
5524 056054 104406      TRAP   C$CLP1            ;TRAP
5525
5526      ;*****
5527      ;
5528      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5529      ;
5530      ;*****
5531
5532 056056 004737 011104      JSR     PC,REWIND          ;CALL TAPE REWIND COMMAND
5533 056062 103413      BCS     30$                ;BR, IF NO PROBLEM
5534 056064 016501 000002      MOV     TSSR(R5),R1        ;GET TSSR
5535 056070 012702 000200      MOV     #SSR,R2           ;SET UP EXPECTED TSSR
5536 056074 010004      MOV     R0,R4             ;PACKET ADDRESS SET UP
5537 056076 005237 002212      INC     FATFLG             ;BUMP COUNT
5541 056102      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
5541 056102 104456      TRAP   C$ERHRD            ;TRAP
5541 056104 001133      .WORD  603                ;.WORD
5541 056106 073604      .WORD  T26RWN             ;.WORD
5541 056110 012136      .WORD  PKTSSR             ;.WORD
5542 056112      30$:   CKLOOP              ;LOOP IF SELECTED
5542 056112 104406      TRAP   C$CLP1            ;TRAP
5543
5544      ;*****
5545      ;
5546      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5547      ;
5548      ;*****
5549
5550 056114 013701 072150      MOV     T26BFR+6,R1        ;PICK UP XSTO
5551 056120 010102      MOV     R1,R2             ;SET UP EXPECTED
5552 056122 052702 000002      BIS     #BIT1,R2          ;SET BOT BIT IN EXPECTED
5553 056126 020102      CMP     R1,R2             ;DOES EXP = REC'D
5554 056130 001406      BEQ     40$                ;BR, IF EQUAL (OK)
5555 056132 005237 002212      INC     FATFLG             ;BUMP COUNT
5559 056136      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
5559 056136 104456      TRAP   C$ERHRD            ;TRAP
5559 056140 001134      .WORD  604                ;.WORD
5559 056142 073315      .WORD  T26BOT             ;.WORD
5559 056144 015564      .WORD  EXPREC            ;.WORD

```

```

5560 056146          404:  CKLOOP                ;LOOP IF SELECTED
      056146 104406          ;                                TRAP  C$CLP1
5561 056150 012703 000400          MOV  #256.,R3          ;RECORD SIZE
5562 056154 013737 003114 072252  MOV  FREE,T26RB       ;STARTING WRITE BUFFER ADDRESS
5563
5564 ;.....
5565 ;
5566 ;WRITE DATA,ACK,CVC=1 COMMAND
5567 ;
5568 ;.....
5569
5570 056162 012737 140005 072250  MOV  #140005,T26PK3   ;WRITE DATA,ACK,CVC=1 COMMAND
5571 056170 012704 072250          MOV  #T26PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
5572 056174          654:
5573 056174 010300          MOV  R3,R0            ;SET PATTERN IN CORRECT REGISTER
5574 056176 004737 017512          JSR  PC,FILLMEM       ;FILL MEMORY WITH RECORD SIZE
5575 056202 010337 072256          MOV  R3,T26SZ         ;SET UP RECORD SIZE IN PACKET
5576 056206 010465 000000          MOV  R4,TSDB(R5)     ;ISSUE COMMAND
5577 056212 004737 016340          JSR  PC,WAITF         ;WAIT FOR SSR TO SET
5578 056216 016501 000002          MOV  TSSR(R5),R1     ;GET TSSR CONTENTS
5579 056222 012702 000200          MOV  #SSR,R2         ;SET UP EXPECTED
5580 056226 020102          CMP  R1,R2           ;ARE THEY EQUAL
5581 056230 001406          BEQ  754             ;BR, IF OK
5582 056232 005237 002212          INC  FATFLG          ;BUMP COUNT
5586 056236          ERRHRD  ERRNO,WRERR,EXPREC ;TSSR INCORRECT AFTER WRITE DATA
      056236 104456          ;                                TRAP  C$ERHRD
      056240 001135          ;                                .WORD 605
      056242 005111          ;                                .WORD WRERR
      056244 015564          ;                                .WORD EXPREC
5587 056246          754:  CKLOOP                ;LOOP IF SELECTED
      056246 104406          ;                                TRAP  C$CLP1
5588 056250          TST  (R3).           ;BUMP RECORD SIZE
5589 056252 022703 000414          CMP  #268.,R3        ;END OF RECORD YET
5590 056256 001346          BNE  654             ;BR, IF MORE RECORDS TO WRITE
5591 056260          804:  CKLOOP                ;LOOP IF SELECTED
      056260 104406          ;                                TRAP  C$CLP1
5592 056262          1204:
5593 ;.....
5594 ;
5595 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5596 ;
5597 ;.....
5598 ;
5599
5600 056262 004737 011104          JSR  PC,REWIND        ;CALL TAPE REWIND COMMAND
5601 056266 103413          BCS  1304            ;BR, IF NO PROBLEM
5602 056270 016501 000002          MOV  TSSR(R5),R1     ;GET TSSR
5603 056274 012702 000200          MOV  #SSR,R2         ;SET UP EXPECTED TSSR
5604 056300 010004          MOV  R0,R4           ;PACKET ADDRESS SET UP
5605 056302 005237 002212          INC  FATFLG          ;BUMP COUNT
5609 056306          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      056306 104456          ;                                TRAP  C$ERHRD
      056310 001136          ;                                .WORD 606
      056312 073604          ;                                .WORD T26RWN
      056314 012136          ;                                .WORD PKTSSR
5610 056316          1304:  CKLOOP                ;LOOP IF SELECTED
      056316 104406          ;                                TRAP  C$CLP1

```

```

5611
5612
5613
5614
5615
5616
5617
5618 056320 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO
5619 056324 010102      MOV      R1,R2           ;SET UP EXPECTED
5620 056326 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
5621 056332 020102      CMP      R1,R2           ;DOES EXP = REC'D
5622 056334 001406      BEQ      140$           ;BR, IF EQUAL (OK)
5623 056336 005237 002212      INC      FATFLG          ;BUMP COUNT
5627 056342      ERRHRD  ERRNO,T26BOT,PKTSSR ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    607
                                .WORD    T26BOT
                                .WORD    PKTSSR
5628 056352      140$:  CKLOOP           ;LOOP IF SELECTED      TRAP      C$CLP1
                                .WORD    104406
5629 056354 012737 000400 072302      MOV      @256.,T26RSZ    ;SET RECORD SIZE
5630
5631
5632
5633
5634
5635
5636
5637
5638 056362 012703 000001      145$:  MOV      @1,R3      ;SPACE ONE RECORD PARAMETER
5639 056366 004737 010556      JSR      PC,SPACE        ;CALL SPACE ROUTINE
5640 056372 103412      BCS      150$           ;BR, IF NO PROBLEM WITH SPACE COMMAND
5641 056374 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
5642 056400 012702 000200      MOV      @SSR,R2        ;SET UP EXPECTED TSSR
5643 056404 005237 002212      INC      FATFLG          ;BUMP COUNT
5647 056410      ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
                                TRAP      C$ERHRD
                                .WORD    608
                                .WORD    T26SC
                                .WORD    EXPREC
5648 056420      150$:  CKLOOP           ;RECORD SIZE          TRAP      C$CLP1
                                .WORD    104406
5649 056422 013703 072302      MOV      T26RSZ,R3       ;RECORD SIZE
5650 056426 013737 003114 072252      MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
5651
5652
5653
5654
5655
5656
5657
5658 056434 012737 141001 072250      165$:  MOV      @141001,T26PK3 ;REREREAD DATA,CVC=1,ACK COMMAND
5659 056442 012704 072250      MOV      @T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
5660 056446 010337 072256      MOV      R3,T26SZ        ;SET UP RECORD SIZE IN PACKET
5661 056452 010465 000000      MOV      R4,TSDB(R5)     ;ISSUE COMMAND
5662 056456 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
5663 056462 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS

```

```

5664 056466 012702 000200      MOV    #SSR,R2          ;SET UP EXPECTED
5665 056472 020102           CMP    R1,R2          ;ARE THEY EQUAL
5666 056474 001406           BEQ    1704           ;BR, IF OK
5667 056476 005237 002212      INC    FATFLG         ;BUMP COUNT
5672 056502           ERRHRD  ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
           056502 104456           TRAP   C1ERHRD
           056504 001141           .WORD  609
           056506 074140           .WORD  T26WDC
           056510 012136           .WORD  PKTSSR
5672 056512           1704:  CKLOOP         ;LOOP IF SELECTED
           056512 104406           TRAP   C1CLP1
5673 056514 013702 003114      MOV    FREE,R2        ;CURRENT BUFFER ADDRESS TO R2
5674 056520 010304           MOV    R3,R4          ;CURRENT RECORD SIZE
5675 056522 162704 000400      SUB    #256.,R4      ;FIRST LOCATION IN BUFFER
5676 056526 060204           1734:  ADD    R2,R4      ;SET UP POINTER
5677 056530 021403           CMP    (R4),R3        ;CHECK DATA READ (R3=DATA ALSO)
5678 056532 001410           BEQ    1804           ;BR, IF ALL IS WELL
5679 056534 011401           MOV    (R4),R1        ;RECD DATA
5680 056536 010302           MOV    R3,R2          ;EXPECTED DATA
5681 056540 005237 002212      INC    FATFLG         ;BUMP COUNT
5685 056544           ERRHRD  ERRNO,T26DTA,EXPREC ;DATA READ NOT - WRITTEN
           056544 104456           TRAP   C1ERHRD
           056546 001142           .WORD  610
           056550 073362           .WORD  T26DTA
           056552 015564           .WORD  EXPREC
5686 056554           1804:  CKLOOP         ;LOOP IF SELECTED
           056554 104406           TRAP   C1CLP1
5687 056556 005724           TST    (R4),          ;BUMP TO NEXT LOCATION
5688 056560 160204           SUB    R2,R4          ;CORRECT RECORDS SIZE VALUE
5689 056562 020403           CMP    R4,R3          ;END OF RECORD YET
5690 056564 001360           BNE    1734           ;BR, IF NOT AT END OF RECORD
5691 056566 005723           TST    (R3),          ;BUMP RECORD SIZE
5692 056570 010337 072302      MOV    R3,T26RSZ     ;RESET RECORD SIZE
5693 056574 022703 000412      CMP    #266.,R3      ;END OF RECORD YET
5694 056600 001270           BNE    1454           ;BR, IF MORE RECORDS TO READ
5695 056602           1904:  CKLOOP         ;LOOP IF SELECTED
           056602 104406           TRAP   C1CLP1
5696 056604           ENDSUB              ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
           056604           L10103:
           056604 104403           TRAP   C1ESUB
5697 056606 023727 002212 000017      CMP    FATFLG,#15.   ;IS ERROR COUNT AT 25
5698 056614 103402           BLO    9994           ;BR, IF LESS THAN 25
5699 056616 004737 017272      JSR    PC,CKDROP     ;TRY TO DROP THE UNIT
5700 056622           9994:
5701
5702
5703 ;*
5704 ;TEST 6, SUBTEST 2
5705 ;
5706 ;VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=0
5707 ;AND SMB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS
5708 ;THE SAME AS THAT USED IN SUBTEST 1, BUT IT IS
5709 ;VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS
5710 ;SWAPPED BYTES.
5711 ;
5712 ;
5713 ;

```



```

5769 056750          ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      056750 104456
      056752 001145
      056754 073604
      056756 012136
5770 056760          30$:  CKLOOP                      ;LOOP IF SELECTED
      056760 104406
5771
5772
5773
5774
5775
5776
5777
5778 056762 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO
5779 056766 010102            MOV      R1,R2           ;SET UP EXPECTED
5780 056770 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
5781 056774 020102            CMP      R1,R2          ;DOES EXP = REC'D
5782 056776 001406            BEQ      40$            ;BR, IF EQUAL (OK)
5783 057000 005237 002212      INC      FATFLG         ;BUMP COUNT
5787 057004          ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      057004 104456
      057006 001146
      057010 073315
      057012 015564
5788 057014          40$:  CKLOOP                      ;LOOP IF SELECTED
      057014 104406
5789 057016 012703 000400      MOV      #256.,R3       ;RECORD SIZE
5790 057022 013737 003114 072252  MOV      FREE,T26RB     ;STARTING WRITE BUFFER ADDRESS
5791
5792
5793
5794
5795
5796
5797
5798 057030 012737 110005 072250  MOV      #110005,T26PK3 ;WRITE DATA,ACK,SWB COMMAND
5799 057036 012704 072250      MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
5800 057042          65$:
5801 057042 010300            MOV      R3,R0         ;SET PATTERN IN CORRECT REGISTER
5802 057044 004737 017512      JSR      PC,FILLMEM    ;FILL MEMORY WITH RECORD SIZE
5803 057050 010337 072256      MOV      R3,T26SZ     ;SET UP RECORD SIZE IN PACKET
5804 057054 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
5805 057060 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
5806 057064 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
5807 057070 012702 000200      MOV      #SSR,R2     ;SET UP EXPECTED
5808 057074 020102            CMP      R1,R2        ;ARE THEY EQUAL
5809 057076 001406            BEQ      75$          ;BR, IF OK
5810 057100 005237 002212      INC      FATFLG         ;BUMP COUNT
5814 057104          ERRHRD  ERRNO,WRTErr,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
      057104 104456
      057106 001147
      057110 005111
      057112 012136
5815 057114          75$:  CKLOOP                      ;LOOP IF SELECTED
      057114 104406
5816 057116 005723            TST      (R3),         ;BUMP RECORD SIZE

```

```

TRAP  C#ERHRD
.WORD 613
.WORD T26RWN
.WORD PKTSSR
TRAP  C#CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;
;*****
TRAP  C#ERHRD
.WORD 614
.WORD T26BOT
.WORD EXPREC
TRAP  C#CLP1
TRAP  C#ERHRD
.WORD 615
.WORD WRTErr
.WORD PKTSSR
TRAP  C#CLP1

```

```

5817 057120 022703 000414          CMP      #268.,R3          ;END OF RECORD YET
5818 057124 001346          BNE      65$              ;BR, IF MORE RECORDS TO WRITE
5819 057126          80$:  CKLOOP          ;LOOP IF SELECTED
5819 057126 104406          TRAP      C$CLP1
5820 057130          120$:
5821
5822          ;*****
5823          ;
5824          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5825          ;
5826          ;*****
5827
5828 057130 004737 011104          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
5829 057134 103413          BCS      130$              ;BR, IF NO PROBLEM
5830 057136 016501 000002          MOV      TSSR(R5),R1       ;GET TSSR
5831 057142 012702 000200          MOV      #SSR,R2          ;SET UP EXPECTED TSSR
5832 057146 010004          MOV      R0,R4            ;PACKET ADDRESS SET UP
5833 057150 005237 002212          INC      FATFLG           ;BUMP COUNT
5837 057154          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
5837 057154 104456          TRAP      C$ERHRD
5837 057156 001150          .WORD    616
5837 057160 073604          .WORD    T26RWN
5837 057162 012136          .WORD    PKTSSR
5838 057164          130$:  CKLOOP          ;LOOP IF SELECTED
5838 057164 104406          TRAP      C$CLP1
5839
5840          ;*****
5841          ;
5842          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
5843          ;
5844          ;*****
5845
5846 057166 013701 072150          MOV      T26BFR+6,R1       ;PICK UP XSTO
5847 057172 010102          MOV      R1,R2            ;SET UP EXPECTED
5848 057174 052702 000002          BIS      #BIT1,R2         ;SET BOT BIT IN EXPECTED
5849 057200 020102          CMP      R1,R2            ;DOES EXP = REC'D
5850 057202 001406          BEQ      140$              ;BR, IF EQUAL (OK)
5851 057204 005237 002212          INC      FATFLG           ;BUMP COUNT
5855 057210          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
5855 057210 104456          TRAP      C$ERHRD
5855 057212 001151          .WORD    617
5855 057214 073315          .WORD    T26BOT
5855 057216 015564          .WORD    EXPREC
5856 057220          140$:  CKLOOP          ;LOOP IF SELECTED
5856 057220 104406          TRAP      C$CLP1
5857 057222 012737 000400 072302          MOV      #256.,T26RSZ      ;SET UP RECORD SIZE
5858
5859          ;*****
5860          ;
5861          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
5862          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
5863          ;
5864          ;*****
5865
5866 057230 012703 000001          145$:  MOV      #1,R3          ;SPACE ONE RECORD PARAMETER
5867 057234 004737 010556          JSR      PC,SPACE          ;CALL SPACE ROUTINE
5868 057240 103412          BCS      150$              ;BR, IF NO PROBLEM WITH SPACE COMMAND

```

```

5869 057242 016501 000002          MOV    TSSR(R5),R1          ;GET TSSR
5870 057246 012702 000200          MOV    #SSR,R2            ;SET UP EXPECTED TSSR
5871 057252 005237 002212          INC    FATFLG             ;BUMP COUNT
5875 057256          ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
                                TRAP    C#ERHRD
                                .WORD   618
                                .WORD   T26SC
                                .WORD   EXPREC
5876 057266          150$:  CKLOOP
                                TRAP    C#CLP1
5877 057270 013703 072302          MOV    T26RSZ,R3         ;RECORD SIZE
5878 057274 013737 003114 072252      MOV    FREE,T26RB        ;STARTING READ BUFFER ADDRESS
5879
5880          ;*****
5881          ;
5882          ;REREAD DATA,CVC=1,ACK,SWB COMMAND
5883          ;
5884          ;*****
5885
5886 057302 012737 151001 072250      MOV    #151001,T26PK3    ;REREAD DATA,CVC=1,ACK,SWB COMMAND
5887 057310 012704 072250      165$:  MOV    #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
5888 057314 010337 072256      MOV    R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
5889 057320 010465 000000      MOV    R4,TSDB(R5)       ;ISSUE COMMAND
5890 057324 004737 016340      JSR    PC,WAITF          ;WAIT FOR SSR TO SET
5891 057330 016501 000002      MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
5892 057334 012702 000200      MOV    #SSR,R2          ;SET UP EXPECTED
5893 057340 020102          CMP    R1,R2            ;ARE THEY EQUAL
5894 057342 001406          BEQ    170$             ;BR, IF OK
5895 057344 005237 002212      INC    FATFLG           ;BUMP COUNT
5899 057350          ERRHRD  ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP    C#ERHRD
                                .WORD   619
                                .WORD   T26WDC
                                .WORD   PKTSSR
5900 057360          170$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
5901 057362 013702 003114          MOV    FREE,R2          ;CURRENT BUFFER ADDRESS TO R2
5902 057366 010304          MOV    R3,R4            ;CURRENT RECORD SIZE
5903 057370 162704 000400      SUB    #256.,R4         ;FIRST LOCATION IN BUFFER
5904 057374 060204      173$:  ADD    R2,R4            ;SET UP POINTER
5905 057376 021403      CMP    (R4),R3         ;CHECK DATA READ (R3=DATA ALSO)
5906 057400 001410      BEQ    180$             ;BR, IF ALL IS WELL
5907 057402 011401      MOV    (R4),R1         ;RECD DATA
5908 057404 010302      MOV    R3,R2           ;EXPECTED DATA
5909 057406 005237 002212      INC    FATFLG           ;BUMP COUNT
5913 057412          ERRHRD  ERRNO,T26DTA,EXPREC ;DATA READ NOT = WRITTEN
                                TRAP    C#ERHRD
                                .WORD   620
                                .WORD   T26DTA
                                .WORD   EXPREC
5914 057422          180$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
5915 057424 005724          TST    (R4)+            ;BUMP TO NEXT LOCATION
5916 057426 160204          SUB    R2,R4            ;CORRECT RECORDS SIZE VALUE
5917 057430 020403          CMP    R4,R3            ;END OF RECORD YET
5918 057432 001360          BNE    173$             ;BR, IF NOT AT END OF RECORD
5919 057434 005723          TST    (R3)+            ;BUMP RECORD SIZE

```



TSV7 - HARDWARE TESTS 1-8  
TEST 6: REREADS

MACRO M1113 14-JUN-84 14:17

I 1

SEQ 0215

```
5920 057436 010337 072302      MOV    R3,T26RSZ          ;STORE RECORD SIZE
5921 057442 022703 000412      CMP    0266.,R3          ;END OF RECORD YET
5922 057446 001270              BNE    145$              ;BR, IF MORE RECORDS TO READ
5923 057450 190$: CKLOOP          ;LOOP IF SELECTED
    057450 104406              TRAP   C#CLP1
5924 057452              ENDSUB                    ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
    057452 104406              L10104:
    057452 104403              TRAP   C#ESUB
5925 057454 023727 002212 000017 CMP    FATFLG,#15.        ;IS ERROR COUNT AT 25
5926 057462 103402              BLO    999$              ;BR, IF LESS THAN 25
5927 057464 004737 017272      JSR    PC,CKDROP         ;TRY TO DROP THE UNIT
5928 057470 999$:
5929
5930 ;*
5931 ;
5932 ;TEST 6, SUBTEST 3
5933 ;
5934 ;VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=1
5935 ;(READ REVERSE, SPACE FORWARD) AND SMB=0 OPERATES
5936 ;PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN
5937 ;WITH A SERIES OF TEST RECORDS VARYING IN LENGTH AND
5938 ;DATA CONTENT; THE FIRST FOUR BYTES OF EACH RECORD
5939 ;CONTAIN ITS RECORD NUMBER (INDICATING POSITION ON
5940 ;TAPE). THE TAPE IS THEN REWOUND AGAIN. FOR EACH
5941 ;TEST RECORD, THE FOLLOWING SEQUENCE IS EXECUTED.
5942 ;
5943 ;1. THE REREAD PREVIOUS COMMAND WITH OPP=1 IS ISSUED
5944 ;AND THE RESULTS CHECKED
5945 ;
5946 ;2. A READ FORWARD COMMAND IS THEN ISSUED AND THE
5947 ;DATA IS CHECKED TO VERIFY THAT THE TAPE WAS
5948 ;POSITIONED PROPERLY AFTER THE REREAD PREVIOUS
5949 ;COMMAND (E.G. THE TAPE SHOULD HAVE BEEN LEFT
5950 ;POSITIONED AT THE START OF THE TEST RECORD). THE
5951 ;READ FORWARD COMMAND LEAVES THE TAPE POSITIONED
5952 ;PROPERLY AT THE START OF THE NEXT TEST RECORD.
5953 ;
5954 ;THE BYTE COUNT ON EACH REREAD PREVIOUS COMMAND IS SET
5955 ;TO THE LENGTH OF THE EXPECTED RECORD, SO NO
5956 ;EXCEPTIONAL CONDITIONS SHOULD OCCUR.
5957 ;
5958 ;
5959 ;
5960 ;
5961 ;-
5962 ;
5963 057470      BGNSUB                    ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
    057470      T6.3:
    057470 104402              TRAP   C#BSUB
5964 057472 004737 075040      JSR    PC,T26REST        ;SET COMMAND PACKET
5965 057476 005037 072276      CLR    T26CNT            ;CLEAR TAPE RECORD COUNTER
5966 057502 004737 075132      JSR    PC,T26RT2        ;SET UP OTHER COMMAND PACKET
5967 057506 004737 075174      JSR    PC,T26RT3        ;SET UP OTHER COMMAND PACKET
5968
5969 ;*****
5970 ;
5971 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
```

```

5972
5973
5974
5975 057512 004737 016064      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
5976 057516 103407             BCS    20$             ;BR IF INIT WAS OK
5977 057520 005237 002212      INC    FATFLG          ;BUMP COUNT
5981 057524 010001             MOV    R0,R1           ;CONTENTS OF TSSR REGISTER
5982 057526             ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                    057526 104455             TRAP   C$ERDF
                    057530 001155             .WORD 621
                    057532 003650             .WORD SFIERR
                    057534 012124             .WORD SFIMSG
5983 057536 013737 002172 072140 20$:  MOV    UNITN,T26DSW      ;SET UP UNIT NUMBER
5984
5985 057544 012704 072120             MOV    #T26PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
5986
5987
5988
5989
5990
5991
5992
5993 057550 004737 010752      JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
5994 057554 103407             BCS    26$             ;BR, IF COMMAND ISSUED OK
5995 057556 005237 002212      INC    FATFLG          ;BUMP COUNT
5999 057562 010001             MOV    R0,R1           ;SAVE CONTENTS OF TSSR
6000 057564             ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                    057564 104456             TRAP   C$ERHRD
                    057566 001156             .WORD 622
                    057570 005054             .WORD WRTMSG
                    057572 012124             .WORD SFIMSG
6001 057574             26$:  CKLOOP              ;LOOP IF SELECTED
                    057574 104406             TRAP   C$CLP1
6002
6003
6004
6005
6006
6007
6008
6009 057576 004737 011104      JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
6010 057602 103413             BCS    30$             ;BR, IF NO PROBLEM
6011 057604 016501 000002      MOV    TSSR(R5),R1     ;GET TSSR
6012 057610 012702 000200      MOV    #SSR,R2         ;SET UP EXPECTED TSSR
6013 057614 010004             MOV    R0,R4           ;PACKET ADDRESS SET UP
6014 057616 005237 002212      INC    FATFLG          ;BUMP COUNT
6018 057622             ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                    057622 104456             TRAP   C$ERHRD
                    057624 001157             .WORD 623
                    057626 073604             .WORD T26RWN
                    057630 012136             .WORD PKTSSR
6019 057632             30$:  CKLOOP              ;LOOP IF SELECTED
                    057632 104406             TRAP   C$CLP1
6020
6021
6022
6023

```

```

6024
6025
6026
6027 057634 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO
6028 057640 010102      MOV      R1,R2           ;SET UP EXPECTED
6029 057642 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
6030 057646 020102      CMP      R1,R2           ;DOES EXP = REC'D
6031 057650 001406      BEQ      40$             ;BR, IF EQUAL (OK)
6032 057652 005237 002212      INC      FATFLG          ;BUMP COUNT
6036 057656      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    624
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C$CLP1
6037 057666      40$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
6038 057670 012703 000400      MOV      @256.,R3        ;RECORD SIZE
6039 057674 013737 003114 072252      MOV      FREE,T26RB      ;STARTING WRITE BUFFER ADDRESS
6040
6041
6042
6043
6044
6045
6046
6047 057702 012737 140005 072250      MOV      @140005,T26PK3  ;WRITE DATA,CVC-1,ACK COMMAND
6048 057710 012704 072250      MOV      @T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
6049 057714      65$:
6050 057714 010300      MOV      R3,R0           ;SET PATTERN IN CORRECT REGISTER
6051 057716 004737 017512      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
6052 057722 010337 072256      MOV      R3,T26SZ        ;SET UP RECORD SIZE IN PACKET
6053 057726 013777 072276 123160      MOV      T26CNT,@FREE   ;MOVE TAPE RECORD NUMBER TO BUFFER
6054 057734 062737 000001 072276      ADD      @1,T25CNT       ;NUMBER READY FOR NEXT RECORD
6055 057742 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
6056 057746 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
6057 057752 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
6058 057756 012702 000200      MOV      @SSR,R2        ;SET UP EXPECTED
6059 057762 020102      CMP      R1,R2          ;ARE THEY EQUAL
6060 057764 001406      BEQ      75$            ;BR, IF OK
6061 057766 005237 002212      INC      FATFLG          ;BUMP COUNT
6065 057772      ERRHRD  FRRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    625
                                .WORD    WRERR
                                .WORD    PKTSSR
                                TRAP      C$CLP1
                                .WORD    104406
6066 060002      75$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    005723
6067 060004 005723      TST      (R3).          ;BUMP THE RECORD SIZE
6068 060006 022703 000414      CMP      @268.,R3       ;MAXIMUM SIZE YET
6069 060012 001401      BEQ      120$           ;BR, IF AT END OF WRITE SEQUENCE
6070 060014 000737      BR       65$            ;WRITE MORE RECORDS
6071 060016      120$:
6072 060016 005037 072276      CLR      T26CNT         ;SET RECORD COUNTER BACK TO ZERO
6073
6074
6075
6076
;*****
;
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE

```

```

6077      ;
6078      ;*****
6079      ;
6080 060022 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
6081 060026 103413              BCS      130$          ;BR, IF NO PROBLEM
6082 060030 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR
6083 060034 012702 000200      MOV      @SSR,R2      ;SET UP EXPECTED TSSR
6084 060040 010004              MOV      R0,R4         ;PACKET ADDRESS SET UP
6085 060042 005237 002212      INC      FATFLG        ;BUMP COUNT
6089 060046              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
        060046 104456              TRAP     C:ERHRD
        060050 001162              .WORD   626
        060052 073604              .WORD   T26RWN
        060054 012136              .WORD   PKTSSR
6090 060056              130$:  CKLOOP          ;LOOP IF SELECTED
        060056 104406              TRAP     C:CLP1
6091      ;
6092      ;*****
6093      ;
6094      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6095      ;
6096      ;*****
6097      ;
6098 060060 013701 072150      MOV      T26BFR+6,R1   ;PICK UP XSTO
6099 060064 010102              MOV      R1,R2         ;SET UP EXPECTED
6100 060066 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
6101 060072 020102              CMP      R1,R2         ;DOES EXP = REC'D
6102 060074 001406              BEQ      140$          ;BR, IF EQUAL (OK)
6103 060076 005237 002212      INC      FATFLG        ;BUMP COUNT
6107 060102              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        060102 104456              TRAP     C:ERHRD
        060104 001163              .WORD   627
        060106 073315              .WORD   T26BOT
        060110 015564              .WORD   EXPREC
6108 060112              140$:  CKLOOP          ;LOOP IF SELECTED
        060112 104406              TRAP     C:CLP1
6109      ;
6110      ;*****
6111      ;
6112      ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
6113      ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
6114      ;
6115      ;*****
6116      ;
6117 060114 012703 000001      MOV      @1,R3         ;SPACE 1 RECORD FORWARD
6118 060120 004737 010556      JSR      PC,SPACE      ;SPACE CALL
6119 060124 012703 000400      MOV      @256.,R3     ;RECORD SIZE
6120 060130 013737 003114 072252 150$:  MOV      FREE,T26RB    ;STARTING READ BUFFER ADDRESS
6121      ;
6122      ;*****
6123      ;
6124      ;REREAD DATA,CVC=1,ACK, OPP COMMAND
6125      ;
6126      ;*****
6127      ;
6128 060136 012737 161001 072250 165$:  MOV      @161001,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
6129 060144 012704 072250      MOV      @T26PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
  
```



```

                                L10105:
                                TRAP      C$ESUB
6181 060344 023727 002212 000017      CMP      FATFLG,#15.      ;IS ERROR COUNT AT 25
6182 060344 103402                  BLO      999$             ;BR, IF LESS THAN 25
6183 060356 004737 017272            JSR      PC,CKDROP        ;TRY TO DROP THE UNIT
6184 060362 999$:
6185
6186
6187
6188 ;TEST 6, SUBTEST 4
6189
6190 ;VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=1
6191 ;AND SWB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS
6192 ;THE SAME THAT IS USED IN SUBTEST 3, BUT IT IS
6193 ;VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS
6194 ;SWAPPED BYTES.
6195
6196
6197
6198
6199
6200
6201 060362 BGNSUB                      ;>>>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>>>
                                T6.4:
                                TRAP      C$BSUB
6202 060362 104402                  JSR      PC,T26REST      ;SET COMMAND PACKET
6203 060370 005037 072276            CLR      T26CNT          ;CLEAR TAPE RECORD COUNTER
6204 060374 004737 075132            JSR      PC,T26RT2       ;SET UP OTHER COMMAND PACKET
6205 060400 004737 075174            JSR      PC,T26RT3       ;SET UP OTHER COMMAND PACKET
6206
6207
6208
6209 ;*****
6210 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6211 ;*****
6212
6213 060404 004737 016064              JSR      PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
6214 060410 103407                  BCS      20$             ;BR IF INIT WAS OK
6215 060412 005237 002212            INC      FATFLG          ;BUMP COUNT
6219 060416 010001                  MOV      R0,R1           ;CONTENTS OF TSSR REGISTER
6220 060420 010001                  ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP      C$ERDF
                                .WORD    631
                                .WORD    SFIERR
                                .WORD    SFIMSG
6221 060430 013737 002172 072140 20$: MCV  UNITN,T26DSW      ;SET UP UNIT NUMBER
6222
6223 060436 012704 072120            MOV      #T26PACKET,R4  ;SUBROUTINE NEEDS PACKET ADDRESS
6224
6225 ;*****
6226 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6227 ;*****
6228
6229
6230
6231 060442 004737 010752              JSR      PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
6232 060446 103407                  BCS      26$             ;BR, IF COMMAND ISSUED OK

```

```

6233 060450 005237 002212      INC    FATFLG      ;BUMP COUNT
6237 060454 010001              MOV    R0,R1      ;SAVE CONTENTS OF TSSR
6238 060456              ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP    C$ERHRD
                                .WORD   632
                                .WORD   WRTMSG
                                .WORD   SFIMSG
6239 060466 104406      264:  CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
6240
6241      ;*****
6242      ;
6243      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6244      ;
6245      ;*****
6246
6247 060470 004737 011104      JSR    PC,REWIND  ;CALL TAPE REWIND COMMAND
6248 060474 103413              BCS    304        ;BR, IF NO PROBLEM
6249 060476 016501 000002      MOV    TSSR(R5),R1 ;GET TSSR
6250 060502 012702 000200      MOV    @SSR,R2    ;SET UP EXPECTED TSSR
6251 060506 010004              MOV    R0,R4      ;PACKET ADDRESS SET UP
6252 060510 005237 002212      INC    FATFLG      ;BUMP COUNT
6256 060514              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   633
                                .WORD   T26RWN
                                .WORD   PKTSSR
6257 060524 104406      304:  CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
6258
6259      ;*****
6260      ;
6261      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6262      ;
6263      ;*****
6264
6265 060526 013701 072150      MOV    T26BFR+6,R1 ;PICK UP XSTO
6266 060532 010102              MOV    R1,R2      ;SET UP EXPECTED
6267 060534 052702 000002      BIS    @BIT1,R2   ;SET BOT BIT IN EXPECTED
6268 060540 020102              CMP    R1,R2      ;DOES EXP = REC'D
6269 060542 001406              BEQ    404        ;BR, IF EQUAL (OK)
6270 060544 005237 002212      INC    FATFLG      ;BUMP COUNT
6274 060550              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   634
                                .WORD   T26BOT
                                .WORD   EXPREC
6275 060560 104406      404:  CKLOOP      ;LOOP IF SELECTED
                                TRAP    C$CLP1
6276 060562 012703 000400      MOV    @256.,R3   ;RECORD SIZE
6277 060566 013737 003114 072252  MOV    FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
6278
6279      ;*****
6280      ;
6281      ;WRITE DATA,CVC=1,ACK COMMAND
6282      ;
6283      ;*****

```

```

6284
6285 060574 012737 140005 072250      MOV      #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
6286 060602 012704 072250      MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
6287 060606                                654:
6288 060606 010300      MOV      R3,R0              ;SET PATTERN IN CORRECT REGISTER
6289 060610 004737 017512      JSR      PC,FILLMEM         ;FILL MEMORY WITH RECORD SIZE
6290 060614 010337 072256      MOV      R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
6291 060620 013777 072276 122266      MOV      T26CNT,#FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
6292 060626 062737 000001 072276      ADD      #1,T26CNT         ;NUMBER READY FOR NEXT RECORD
6293 060634 010465 000000      MOV      R4,TSD8(R5)       ;ISSUE COMMAND
6294 060640 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
6295 060644 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
6296 060650 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
6297 060654 020102      CMP      R1,R2             ;ARE THEY EQUAL
6298 060656 001406      BEQ      754               ;BR, IF OK
6299 060660 005237 002212      INC      FATFLG            ;BUMP COUNT
6303 060664                                ERRHRD  ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP  C:ERHRD
                                .WORD 635
                                .WORD WRTERR
                                .WORD  PKTSSR
6304 060674                                754:  CKLOOP                ;LOOP IF SELECTED
                                TRAP  C:CLP1
6305 060676 005723      TST      (R3).             ;BUMP THE RECORD SIZE
6306 060700 022703 000412      CMP      #266.,R3         ;MAXIMUM SIZE YET
6307 060704 001401      BEQ      1204             ;BR, IF AT END OF WRITE SEQUENCE
6308 060706 000737      BR       654              ;WRITE MORE RECORDS
6309 060710                                1204:
6310 060710 005037 072276      CLR      T26CNT           ;SET RECORD COUNTER BACK TO ZERO
6311
6312 ;*****
6313 ;
6314 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6315 ;
6316 ;*****
6317
6318 060714 004737 011104      JSR      PC,REWIND         ;CALL TAPE REWIND COMMAND
6319 060720 103413      BCS     1304              ;BR, IF NO PROBLEM
6320 060722 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR
6321 060726 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED TSSR
6322 060732 010004      MOV      R0,R4            ;PACKET ADDRESS SET UP
6323 060734 005237 002212      INC      FATFLG            ;BUMP COUNT
6327 060740                                ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP  C:ERHRD
                                .WORD 636
                                .WORD T26RWN
                                .WORD  PKTSSR
6328 060750                                1304:  CKLOOP                ;LOOP IF SELECTED
                                TRAP  C:CLP1
6329
6330 ;*****
6331 ;
6332 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6333 ;
6334 ;*****
6335
6336 060752 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO

```



```

6337 060756 010102          MOV      R1,R2          ;SET UP EXPECTED
6338 060760 052702 000002  BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
6339 060764 020102          CMP      R1,R2          ;DOES EXP = REC'D
6340 060766 001405          BEQ     140$           ;BR, IF EQUAL (OK)
6341 060770 005237 002212  INC     FATFLG         ;BUMP COUNT
6345 060774          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP   C$ERHRD
                                .WORD  637
                                .WORD  T26BOT
                                .WORD  EXPREC
                                TRAP   C$CLP1
6346 061004          140$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP   C$CLP1
6347 061004 104406
6348
6349
6350
6351
6352
6353
6354
6355 061006 012703 000001          MOV     #1,R3          ;SET UP SPACE FORWARD 1
6356 061012 004737 010556          JSR    PC,SPACE       ;ISSUE SPACE COMMAND
6357 061016 012703 000400          MOV     @256.,R3      ;RECORD SIZE
6358 061022 013737 003114 072252 150$:  MOV     FREE,T26RB    ;STARTING READ BUFFER ADDRESS
6359
6360
6361
6362
6363
6364
6365
6366 061030 012737 171001 072250 165$:  MOV     @171001,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
6367 061036 012704 072250          MOV     @T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
6368 061042 010337 072256          MOV     R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
6369 061046 010465 000000          MOV     R4,TSD8(R5)   ;ISSUE COMMAND
6370 061052 004737 016340          JSR    PC,WAITF       ;WAIT FOR SSR TO SET
6371 061056 016501 000002          MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
6372 061062 012702 000200          MOV     @SSR,R2      ;SET UP EXPECTED
6373 061066 020102          CMP     R1,R2         ;ARE THEY EQUAL
6374 061070 001406          BEQ     170$           ;BR, IF OK
6375 061072 005237 002212  INC     FATFLG         ;BUMP COUNT
6379 061076          ERRHRD  ERRNO,T26RRF,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP   C$ERHRD
                                .WORD  638
                                .WORD  T26RRF
                                .WORD  PKTSSR
6380 061106          170$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP   C$CLP1
6381 061110 017701 122000          MOV     @FREE,R1      ;FIRST WORD FROM READ BUFFER
6382 061114 013702 072276          MOV     T26CNT,R2     ;SET UP EXPECTED
6383 061120 000302          SWAB   R2             ;SWAP BYTES IN EXPECTED
6384 061122 020102          CMP     R1,R2         ;IS TAPE POSITION CORRECT
6385 061124 001406          BEQ     190$           ;KEEP GOING POSITION OK
6386 061126 005237 002212  INC     FATFLG         ;BUMP COUNT
6390 061132          ERRHRD  ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
                                TRAP   C$ERHRD
                                .WORD  639
061132 104456
061134 001177

```

6391	061136	072306						.WORD	T26WNG
	061140	015564						.WORD	EXPREC
6392	061142	104406			190\$: CKLOOP			TRAP	C\$CLP1
6393	061144	005723			TST (R3),				
6394	061146	062737	000001	072276	ADD #1,T26CNT				
6395									
6396									
6397									
6398									
6399									
6400									
6401	061154	012737	140001	072250	MOV #140001,T26PK3				
6402	061162	010337	072256		MOV R3,T26SZ				
6403	061166	010465	000000		MOV R4,TSDB(R5)				
6404	061172	004737	016340		JSR PC,WAITF				
6405	061176	016501	000002		MOV TSSR(R5),R1				
6406	061202	012702	000200		MOV #SSR,R2				
6407	061206	020102			CMP R1,R2				
6408	061210	001406			BEQ 215\$				
6409	061212	005237	002212		INC FATFLG				
6413	061216				ERRHRD ERRNO,T26RDF,PKTSSR				
	061216	104456						TRAP	C\$ERHRD
	061220	001200						.WORD	640
	061222	072456						.WORD	T26RDF
	061224	012136						.WORD	PKTSSR
6414	061226				215\$: CKLOOP				
	061226	104406							
6415	061230	017701	121660		MOV %FREE,R1				
6416	061234	013702	072276		MOV T26CNT,R2				
6417	061240	020102			CMP R1,R2				
6418	061242	001406			BEQ 217\$				
6419	061244	005237	002212		INC FATFLG				
6423	061250				ERRHRD ERRNO,T26WNG,EXPREC				
	061250	104456						TRAP	C\$ERHRD
	061252	001201						.WORD	641
	061254	072306						.WORD	T26WNG
	061256	015564						.WORD	EXPREC
6424	061260				217\$: CKLOOP				
	061260	104406							
6425	061262	022703	000410		CMP #264.,R3				
6426	061266	001401			BEQ 220\$				
6427	061270	000654			BR 150\$				
6428	061272				220\$: ENDSUB				
6429	061272								
	061272	104403							
6430	061274	023727	002212	000017	CMP FATFLG,#15.				
6431	061302	103402			BLO 999\$				
6432	061304	004737	017272		JSR PC,CKDROP				
6433	061310				999\$:				
6434									
6435									
6436									
6437									
6438									

```

;*****
;
;READ DATA, CVC=1, ACK COMMAND
;
;*****

```

```

;READ DATA, CVC=1, ACK COMMAND
;SET SIZE INTO PACKET
;ISSUE READ DATA COMMAND
;WAIT FOR SSR
;PICK UP THE TSSR
;SET UP EXPECTED
;IS THE TSSR OK
;BR, IF TSSR OK (GOOD)
;BUMP COUNT
;READ DATA COMMAND FAILED

```

```

;LOOP IF SELECTED
;FIRST WORD FROM READ BUFFER
;SET UP EXPECTED
;IS TAPE POSITION CORRECT
;KEEP GOING POSITION OK
;BUMP COUNT
;TAPE POSITION INCORRECT

```

```

;AT MAX SIZE YET
;BR, IF AT END OF THE SUBTEST
;KEEP GOING MORE RECORDS

```

```

;***** END SUBTEST *****
L10106:
;IS ERROR COUNT AT 25
;BR, IF LESS THAN 25
;TRY TO DROP THE UNIT

```

```

;
;
;TEST 6, SUBTEST 5
;
```

```

6439          ;VERIFIES THAT A REREAD PREVIOUS COMMAND READING A
6440          ;RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES
6441          ;TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH
6442          ;LONG (RL) BIT SET. RESULTS ARE VERIFIED FOR BOTH
6443          ;STATES OF OPP (0 AND 1).
6444          ;
6445          ;
6446          ;
6447          ;
6448          ;
6449          ;
6450          061310          BGNSUB          ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
           061310          T6.5:
           061310    104402          TRAP      C$BSUB
6451    061312    004737    075040          JSR      PC,T26REST          ;SET COMMAND PACKET
6452    061316    004737    075132          JSR      PC,T26RT2         ;SET UP OTHER COMMAND PACKET
6453    061322    004737    075174          JSR      PC,T26RT3         ;SET UP OTHER COMMAND PACKET
6454
6455          ;*****
6456          ;
6457          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6458          ;
6459          ;*****
6460
6461    061326    004737    016064          JSR      PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
6462    061332    103407          BCS      20$              ;BR IF INIT WAS OK
6463    061334    005237    002212          INC      FATFLC          ;BUMP COUNT
6464    061340    010001          MOV      RO,R1           ;CONTENTS OF TSSR REGISTER
6465    061342          ERRDF      ERRNO,SFIERR,SFIMSG          ;FATAL ERROR TSSR WAS NOT OK
           061342    104455          TRAP      C$ERDF
           061344    001202          .WORD    642
           061346    003650          .WORD    SFIERR
           061350    012124          .WORD    SFIMSG
6466
6469    061352    013737    002172    072140    20$:  MOV      UNITN,T26DSW          ;SET UP UNIT NUMBER
6470
6471    061360    012704    072120          MOV      @T26PACKET,R4        ;SUBROUTINE NEEDS PACKET ADDRESS
6472
6473          ;*****
6474          ;
6475          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6476          ;
6477          ;*****
6478
6479    061364    004737    010752          JSR      PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
6480    061370    103407          BCS      26$              ;BR, IF COMMAND ISSUED OK
6481    061372    005237    002212          INC      FATFLG          ;BUMP COUNT
6482    061376    010001          MOV      RO,R1           ;SAVE CONTENTS OF TSSR
6483    061400          ERRHRU      ERRNO,WRTMSG,SFIMSG          ;WRITE CHARACTERISTIC FAILED
           061400    104456          TRAP      C$ERHRD
           061402    001203          .WORD    643
           061404    005054          .WORD    WRTMSG
           061406    012124          .WORD    SFIMSG
6484
6487    061410          26$:  CKLOOP          ;LOOP IF SELECTED
           061410    104406          TRAP      C$CLP1
6488
6489          ;*****
6490          ;

```

```

6491      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6492      ;
6493      ;*****
6494
6495 061412 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
6496 061416 103413      BCS      30$           ;BR, IF NO PROBLEM
6497 061420 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR
6498 061424 012702 000200      MOV      @SSR,R2      ;SET UP EXPECTED TSSR
6499 061430 010004      MOV      R0,R4        ;PACKET ADDRESS SET UP
6500 061432 005237 002212      INC      FATFLG       ;BUMP COUNT
6504 061436      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
        ;
        ;TRAP      C$ERHRD
        ;.WORD    644
        ;.WORD    T26RWN
        ;.WORD    PKTSSR
        30$:  CKLOOP      ;LOOP IF SELECTED
        ;TRAP      C$CLP1
6506
6507      ;*****
6508      ;
6509      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6510      ;
6511      ;*****
6512
6513 061450 013701 072150      MOV      T26BFR+6,R1  ;PICK UP XSTO
6514 061454 010102      MOV      R1,R2        ;SET UP EXPECTED
6515 061456 052702 000002      BIS      @BIT1,R2     ;SET BOT BIT IN EXPECTED
6516 061462 020102      CMP      R1,R2       ;DOES EXP = REC'D
6517 061464 001406      BEQ      40$         ;BR, IF EQUAL (OK)
6518 061466 005237 002212      INC      FATFLG       ;BUMP COUNT
6522 061472      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        ;
        ;TRAP      C$ERHRD
        ;.WORD    645
        ;.WORD    T26BOT
        ;.WORD    EXPREC
        40$:  CKLOOP      ;LOOP IF SELECTED
        ;TRAP      C$CLP1
6523 061502 061502 104406      MOV      @512.,R3     ;RECORD SIZE
6524 061504 012703 001000      MOV      FREE,T26RB  ;STARTING WRITE BUFFER ADDRESS
6525 061510 013737 003114 072252
6526
6527      ;*****
6528      ;
6529      ;WRITE DATA,CVC=1,ACK COMMAND
6530      ;
6531      ;*****
6532
6533 061516 012737 140005 072250      MOV      @140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
6534 061524 012704 072250      MOV      @T26PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
6535 061530      65$:  MOV      R3,T26SZ     ;SET UP RECORD SIZE IN PACKET
6536 061530 010337 072256      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
6537 061534 010465 000000      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
6538 061540 004737 016340      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
6539 061544 016501 000002      MOV      @SSR,R2     ;SET UP EXPECTED
6540 061550 012702 000200      CMP      R1,R2       ;ARE THEY EQUAL
6541 061554 020102      BEQ      75$         ;BR, IF OK
6542 061556 001406      INC      FATFLG       ;BUMP COUNT
6543 061560 005237 002212

```

```

6547 061564          ERRHRD  ERRNO,WRTErr,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
      061564 104456          TRAP          C$ERHRD
      061566 001206          .WORD        646
      061570 005111          .WORD        WRTErr
      061572 012136          .WORD        PKTSSR
6548 061574          75$:   CKLOOP                    ;LOOP IF SELECTED
      061574 104406          TRAP          C$CLP1
6549 061576 005303          DEC          R3            ;SET RECORD SIZE TO 511.
6550 061600 013737 003114 072252  MOV        FREE,T26RB      ;STARTING READ BUFFER ADDRESS
6551
6552 ;*****
6553 ;
6554 ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
6555 ;
6556 ;*****
6557
6558 061606 012737 161001 072250  MOV        #161001,T26PK3      ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
6559 061614 012704 072250 165$:   MOV        #T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6560 061620 010337 072256          MOV        R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
6561 061624 010465 000000          MOV        R4,TSD8(R5)      ;ISSUE COMMAND
6562 061630 004737 016340          JSR        PC,WAITF        ;WAIT FOR SSR TO SET
6563 061634 016501 000002          MOV        TSSR(R5),R1     ;GET TSSR CONTENTS
6564 061640 012702 100204          MOV        #SSR!SC!BIT2,R2 ;SET UP EXPECTED
6565 061644 020102          CMP        R1,R2          ;ARE THEY EQUAL
6566 061646 001406          BEQ        170$          ;BR, IF OK
6567 061650 005237 002212          INC        FATFLG        ;BUMP COUNT
6571 061654          ERRHRD  ERRNO,T26TRL,PKTSSR      ;TSSR INCORRECT AFTER REREAD DATA
      061654 104456          TRAP          C$ERHRD
      061656 001207          .WORD        647
      061660 074662          .WORD        T26TRL
      061662 012136          .WORD        PKTSSR
6572 061664          170$:  CKLOOP                    ;LOOP IF SELECTED
      061664 104406          TRAP          C$CLP1
6573
6574 ;*****
6575 ;
6576 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6577 ;
6578 ;*****
6579
6580 061666 013701 072150          MOV        T26BFR+6,R1     ;GET MESSAGE BUFFER
6581 061672 010102          MOV        R1,R2          ;SET UP EXPECTED
6582 061674 052702 010000          BIS        #BIT12,R2     ;SET THE RLL BIT IN EXPECTED
6583 061700 020102          CMP        R1,R2          ;ARE THEY EQUAL
6584 061702 001406          BEQ        180$          ;BR, IF EQUAL (ALL IS WELL)
6585 061704 005237 002212          INC        FATFLG        ;BUMP COUNT
6589 061710          ERRHRD  ERRNO,T26LON,EXPREC      ;THE RLL BIT WAS NOT SET IN XSTO
      061710 104456          TRAP          C$ERHRD
      061712 001210          .WORD        648
      061714 074430          .WORD        T26LON
      061716 015564          .WORD        EXPREC
6590 061720          180$:  CKLOOP                    ;LOOP IF SELECTED
      061720 104406          TRAP          C$CLP1
6591 061722 012703 000777          MOV        #511.,R3       ;SET RECORD SIZE
6592 061726 013737 003114 072252  MOV        FREE,T26RB      ;STARTING READ BUFFER ADDRESS
6593
6594 ;*****

```



```

6646 ;SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE
6647 ;RESIDUAL BYTE COUNTER (RBPCR) IN THE MESSAGE BUFFER
6648 ;CONTAINS THE PROPER NONZERO VALUE (E.G., THE
6649 ;DIFFERENCE BETWEEN THE ORIGINAL BYTE COUNT AND THE
6650 ;ACTUAL RECORD LENGTH). RESULTS ARE VERIFIED FOR BOTH
6651 ;STATES OF OPP (0 AND 1).
6652 :
6653 :
6654 :
6655 :-
6656 062066 BGNSUB ;>>>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
          062066 T6.6:
          062066 104402 TRAP C$BSUB
6657 062070 004737 075040 JSR PC,T26REST ;SET COMMAND PACKET
6658 062074 004737 075132 JSR PC,T26RT2 ;SET UP OTHER COMMAND PACKET
6659 062100 004737 075174 JSR PC,T26RT3 ;SET UP OTHER COMMAND PACKET
6660
6661 ;*****
6662 ;
6663 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6664 ;
6665 ;*****
6666
6667 062104 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
6668 062110 103407 BCS 20$ ;BR IF INIT WAS OK
6669 062112 005237 002212 INC FATFLG ;BUMP COUNT
6673 062116 010001 MOV R0,R1 ;CONTENTS OF TSSR REGISTER
6674 062120 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
          062120 104455 TRAP C$ERDF
          062122 001213 .WORD 651
          062124 003650 .WORD SFIERR
          062126 012124 .WORD SFIMSG
6675 062130 013737 002172 072140 20$: MOV UNITN,T26DSW ;SET UP UNIT NUMBER
6676
6677 062136 012704 072120 MOV #T26PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
6678
6679 ;*****
6680 ;
6681 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6682 ;
6683 ;*****
6684
6685 062142 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
6686 062146 103407 BCS 26$ ;BR, IF COMMAND ISSUED OK
6687 062150 005237 002212 INC FATFLG ;BUMP COUNT
6691 062154 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
6692 062156 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
          062156 104456 TRAP C$ERHRD
          062160 001214 .WORD 652
          062162 005054 .WORD WRTMSG
          062164 012124 .WORD SFIMSG
6693 062166 26$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
          062166 104406
6694
6695 ;*****
6696 ;
6697 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE

```

```

6698 ;
6699 ;*****
6700
6701 062170 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6702 062174 103413 BCS 30$ ;BR, IF NO PROBLEM
6703 062176 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
6704 062202 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
6705 062206 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
6706 062210 005237 002212 INC FATFLG ;BUMP COUNT
6710 062214 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
        062214 104456 TRAP C$ERHRD
        062216 001215 .WORD 653
        062220 073604 .WORD T26RWN
        062222 012136 .WORD PKTSSR
6711 062224 30$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
        062224 104406
6712 ;*****
6713 ;
6714 ;
6715 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6716 ;
6717 ;*****
6718
6719 062226 013701 072150 MOV T26BFR+6,R1 ;PICK UP XSTO
6720 062232 010102 MOV R1,R2 ;SET UP EXPECTED
6721 062234 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
6722 062240 020102 CMP R1,R2 ;DOES EXP = REC'D
6723 062242 001406 BEQ 40$ ;BR, IF EQUAL (OK)
6724 062244 005237 002212 INC FATFLG ;BUMP COUNT
6728 062250 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        062250 104456 TRAP C$ERHRD
        062252 001216 .WORD 654
        062254 073315 .WORD T26BOT
        062256 015564 .WORD EXPREC
6729 062260 40$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
        062260 104406
6730 062262 012703 000400 MOV #256.,R3 ;RECORD SIZE
6731 062266 013737 003114 072252 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
6732 ;*****
6733 ;
6734 ;
6735 ;WRITE DATA,CVC=1,ACK COMMAND
6736 ;
6737 ;*****
6738
6739 062274 012737 140005 072250 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
6740 062302 012704 072250 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6741 062306 65$:
6742 062306 010337 072256 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
6743 062312 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6744 062316 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6745 062322 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6746 062326 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6747 062332 020102 CMP R1,R2 ;ARE THEY EQUAL
6748 062334 001406 BEQ 75$ ;BR, IF OK
6749 062336 005237 002212 INC FATFLG ;BUMP COUNT
6753 062342 ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA

```





```

6805 062522          ERRHRD  ERRNO,T26PBP,EXPREC      ;RBPBR NOT CORRECT
      062522 104456          TRAP  C1ERHRD
      062524 001222          .WORD 658
      062526 074574          .WORD T26PBP
      062530 015564          .WORD EXPREC
6806 062532          1901:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C1CLP1
      062532 104406          MOV      #512.,R3          ;RECORD SIZE
6807 062534 012703 001000          MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
6808 062540 013737 003114 072252
6809
6810 ;.....
6811 ;
6812 ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
6813 ;
6814 ;.....
6815
6816 062546 012737 141001 072250          MOV      #141001,T26PK3      ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
6817 062554 012704 072250          MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
6818 062560 010337 072256          MOV      R3,T26SZ         ;SET UP RECORD SIZE IN PACKET
6819 062564 010465 000000          MOV      R4,TSDB(R5)      ;ISSUE COMMAND
6820 062570 004737 016340          JSR      PC,WAITF         ;WAIT FOR SSR TO SET
6821 062574 016501 000002          MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
6822 062600 012702 100204          MOV      #SSR!SC!BIT2,R2  ;SET UP EXPECTED
6823 062604 020102          CMP      R1,R2           ;ARE THEY EQUAL
6824 062606 001406          BEQ      2701            ;BR, IF OK
6825 062610 005237 002212          INC      FATFLG          ;BUMP COUNT
6829 062614          ERRHRD  ERRNO,T26TRL,PKTSSR      ;TSSR INCORRECT AFTER READ DATA
      062614 104456          TRAP  C1ERHRD
      062616 001223          .WORD 659
      062620 074662          .WORD T26TRL
      062622 012136          .WORD PKTSSR
6830 062624          2701:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C1CLP1
      062624 104406
6831
6832 ;.....
6833 ;
6834 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6835 ;
6836 ;.....
6837
6838 062626 013701 072150          MOV      T26BFR+6,R1      ;GET MESSAGE BUFFER
6839 062632 010102          MOV      R1,R2           ;SET UP EXPECTED
6840 062634 052702 040000          BIS      #BIT14,R2       ;SET THE RLS BIT IN EXPECTED
6841 062640 020102          CMP      R1,R2           ;ARE THEY EQUAL
6842 062642 001406          BEQ      2801            ;BR, IF EQUAL (ALL IS WELL)
6843 062644 005237 002212          INC      FATFLG          ;BUMP COUNT
6847 062650          ERRHRD  ERRNO,T26LOP,EXPREC      ;THE RLS BIT WAS NOT SET IN XSTO
      062650 104456          TRAP  C1ERHRD
      062652 001224          .WORD 660
      062654 074512          .WORD T26LOP
      062656 015564          .WORD EXPREC
6848 062660          2801:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C1CLP1
      062660 104406
6849 062662 013701 072146          MOV      T26BFR+4,R1      ;PICK UP RESIDUAL BYTE COUNTER
6850 062666 012702 000400          MOV      #256.,R2        ;THIS SHOULD BE THE DIFFERENCE
6851 062672 020102          CMP      R1,R2           ;IS THE DIFFERENCE CORRECT
6852 062674 001405          BEQ      2901            ;BR, IF CORRECT

```



```

6903 062772 013737 002172 072140 204:  MOV    UNITN,T26DSW      ;SET UP UNIT NUMBER
6904                                     ;
6905 063000 012704 072120          MOV    @T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
6906                                     ;
6907                                     ;*****
6908                                     ;
6909                                     ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
6910                                     ;
6911                                     ;*****
6912                                     ;
6913 063004 004737 010752          JSR    PC,WRTPHR          ;ISSUE WRITE CHARACTERISTICS
6914 063010 103407                BCS    264               ;BR, IF COMMAND ISSUED OK
6915 063012 005237 002212          INC    FATFLG            ;BUMP COUNT
6919 063016 010001                MOV    R0,R1             ;SAVE CONTENTS OF TSSR
6920 063020          ERRHRD  ERRNO,WRTPHR,SFIMSG ;WRITE CHARACTERISTICS FAILED
        063020 104456                TRAP                                C$ERRHRD
        063022 001226                .WORD                               662
        063024 005054                .WORD                               WRTPHR
        063026 012124                .WORD                               SFIMSG
6921 063030          264:  CKLOOP                ;LOOP IF SELECTED
        063030 104406                TRAP                                C$CLP1
6922                                     ;*****
6923                                     ;
6924                                     ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6925                                     ;
6926                                     ;*****
6927                                     ;
6928                                     ;
6929 063032 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
6930 063036 103413                BCS    304               ;BR, IF NO PROBLEM
6931 063040 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR
6932 063044 012702 000200          MOV    @SSR,R2           ;SET UP EXPECTED TSSR
6933 063050 010004                MOV    R0,R4             ;PACKET ADDRESS SET UP
6934 063052 005237 002212          INC    FATFLG            ;BUMP COUNT
6938 063056          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
        063056 104456                TRAP                                C$ERRHRD
        063060 001227                .WORD                               663
        063062 073604                .WORD                               T26RWN
        063064 012136                .WORD                               PKTSSR
6939 063066          304:  CKLOOP                ;LOOP IF SELECTED
        063066 104406                TRAP                                C$CLP1
6940                                     ;*****
6941                                     ;
6942                                     ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6943                                     ;
6944                                     ;*****
6945                                     ;
6946                                     ;
6947 063070 013701 072150          MOV    T26BFR+6,R1       ;PICK UP XSTO
6948 063074 010102                MOV    R1,R2             ;SET UP EXPECTED
6949 063076 052702 000002          BIS    @BIT1,R2          ;SET BOT BIT IN EXPECTED
6950 063102 020102                CMP    R1,R2             ;DOES EXP = REC'D
6951 063104 001406                BEQ    404               ;BR, IF EQUAL (OK)
6952 063106 005237 002212          INC    FATFLG            ;BUMP COUNT
6956 063112          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        063112 104456                TRAP                                C$ERRHRD
        063114 001230                .WORD                               664

```

```

063116 073315 .WORD T26BOT
063120 015564 .WORD EXPREC
6957 063122 40$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
063122 104406 ;RECORD SIZE
6958 063124 012703 000400 MOV #256.,R3 ;STARTING WRITE BUFFER ADDRESS
6959 063130 013737 003114 072252 MOV FREE,T26RB
6960
6961 ;*****
6962 ;
6963 ;WRITE DATA,CVC=1,ACK COMMAND
6964 ;
6965 ;*****
6966
6967 063136 012737 140005 072250 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
6968 063144 012704 072250 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6969 063150 65$:
6970 063150 010300 MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
6971 063152 004737 017512 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
6972 063156 010337 072256 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
6973 063162 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6974 063166 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6975 063172 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6976 063176 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6977 063202 020102 CMP R1,R2 ;ARE THEY EQUAL
6978 063204 001406 BEQ 75$ ;BR, IF OK
6979 063206 005237 002212 INC FATFLG ;BUMP COUNT
6983 063212 ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
063212 104456 TRAP C#ERHRD
063214 001231 .WORD 665
063216 005111 .WORD WRERR
063220 012136 .WORD PKTSSR
6984 063222 75$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
063222 104406 ;BUMP RECORD SIZE
6985 063224 005723 TST (R3). ;END OF RECORD YET
6986 063226 022703 000414 CMP #268.,R3 ;BR, IF MORE RECORDS TO WRITE
6987 063232 001346 BNE 65$ ;LOOP IF SELECTED
6988 063234 80$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
063234 104406
6989 063236 120$:
6990
6991 ;*****
6992 ;
6993 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6994 ;
6995 ;*****
6996
6997 063236 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6998 063242 103413 BCS 130$ ;BR, IF NO PROBLEM
6999 063244 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
7000 063250 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7001 063254 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7002 063256 005237 002212 INC FATFLG ;BUMP COUNT
7006 063262 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
063262 104456 TRAP C#ERHRD
063264 001232 .WORD 666
063266 073604 .WORD T26RWN
063270 012136 .WORD PKTSSR

```

```

7007 063272 1304: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
      063272 104406
7008
7009 ;*****
7010 ;
7011 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7012 ;
7013 ;*****
7014
7015 063274 013701 072150 MOV T26BFR+6,R1 ;PICK UP XST0
7016 063300 010102 MOV R1,R2 ;SET UP EXPECTED
7017 063302 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
7018 063306 020102 CMP R1,R2 ;DOES EXP = REC'D
7019 063310 001406 BEQ 1404 ;BR, IF EQUAL (OK)
7020 063312 005237 002212 INC FATFLG ;BUMP COUNT
7024 063316 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      063316 104456 TRAP C$ERHRD
      063320 001233 .WORD 667
      063322 073315 .WORD T26BOT
      063324 015564 .WORD EXPREC
7025 063326 1404: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
      063326 104406
7026 063330 012737 000400 072302 MOV #256.,T26RSZ ;STORE START RECORD SIZE
7027 063336 000420 BR 1504 ;SKIP THE SPACE THIS TIME
7028
7029 ;*****
7030 ;
7031 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
7032 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
7033 ;
7034 ;*****
7035
7036 063340 012703 000001 1454: MOV #1,R3 ;SPACE ONE RECORD PARAMETER
7037 063344 004737 010556 JSR PC,SPACE ;CALL SPACE ROUTINE
7038 063350 103413 BCS 1504 ;BR, IF NO PROBLEM WITH SPACE COMMAND
7039 063352 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
7040 063356 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7041 063362 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7042 063364 005237 002212 INC FATFLG ;BUMP COUNT
7046 063370 ERRHRD ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
      063370 104456 TRAP C$ERHRD
      063372 001234 .WORD 668
      063374 072717 .WORD T26SC
      063376 015564 .WORD EXPREC
7047 063400 1504: CKLOOP ;RECORD SIZE TRAP C$CLP1
      063400 104406
7048 063402 013703 072302 MOV T26RSZ,R3 ;STARTING READ BUFFER ADDRESS
7049 063406 013737 003114 072252 MOV FREE,T26RB
7050
7051 ;*****
7052 ;
7053 ;REREREAD DATA,CVC=1,ACK COMMAND
7054 ;
7055 ;*****
7056
7057 063414 012737 141401 072250 MOV #141401,T26PK3 ;REREREAD DATA,CVC=1,ACK COMMAND
7058 063422 012704 072250 1654: MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS

```







```

7161 063712 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR
7162 063716 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED TSSR
7163 063722 010004              MOV      R0,R4          ;PACKET ADDRESS SET UP
7164 063724 005237 002212      INC      FATFLG         ;BUMP COUNT
7168 063730              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    673
                                .WORD    T26RWN
                                .WORD    PKTSSR
                                TRAP      C#CLP1
7169 063740 30$: CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                TRAP      C#CLP1
7170
7171 ;*****
7172 ;
7173 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7174 ;
7175 ;*****
7176
7177 063742 013701 072150      MOV      T26BFR+6,R1    ;PICK UP XSTO
7178 063746 010102              MOV      R1,R2          ;SET UP EXPECTED
7179 063750 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
7180 063754 020102              CMP      R1,R2          ;DOES EXP = REC'D
7181 063756 001406              BEQ     40$             ;BR, IF EQUAL (OK)
7182 063760 005237 002212      INC      FATFLG         ;BUMP COUNT
7186 063764              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    674
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
7187 063774 40$: CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
7188 063776 012703 000400      MOV      #256.,R3       ;RECORD SIZE
7189 064002 013737 003114 072252  MOV      FREE,T26RB     ;STARTING WRITE BUFFER ADDRESS
7190
7191 ;*****
7192 ;
7193 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
7194 ;
7195 ;*****
7196
7197 064010 012737 150005 072250      MOV      #150005,T26PK3 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
7198 064016 012704 072250              MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
7199 064022 65$:
7200 064022 010300              MOV      R3,R0          ;SET PATTERN IN CORRECT REGISTER
7201 064024 004737 017512      JSR     PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
7202 064030 010337 072256      MOV      R3,T26SZ       ;SET UP RECORD SIZE IN PACKET
7203 064034 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
7204 064040 004737 016340      JSR     PC,WAITF        ;WAIT FOR SSR TO SET
7205 064044 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
7206 064050 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED
7207 064054 020102              CMP      R1,R2          ;ARE THEY EQUAL
7208 064056 001406              BEQ     75$             ;BR, IF OK
7209 064060 005237 002212      INC      FATFLG         ;BUMP COUNT
7213 064064              ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERHRD
                                .WORD    675
                                .WORD    WRTErr
064064 104456
064066 001243
064070 005111

```

```

064072 012136
7214 064074 104406      75$:  CKLOOP                ;LOOP IF SELECTED      .WORD  PKTSSR
064074 104406                TST      (R3)+          ;BUMP RECORD SIZE     TRAP   C$CLP1
7215 064076 005723                CMP      #268.,R3      ;END OF RECORD YET
7216 064100 022703 000414                BNE     65$            ;BR, IF MORE RECORDS TO WRITE
7217 064104 001346                80$:  CKLOOP                ;LOOP IF SELECTED
7218 064106 104406                120$:
064106 104406                ;*****
7219 064110                ;
7220                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7221                ;
7222                ;*****
7223                ;
7224                ;
7225                ;
7226                ;
7227 064110 004737 011104                JSR     PC,REWIND      ;CALL TAPE REWIND COMMAND
7228 064114 103413                BCS     130$          ;BR, IF NO PROBLEM
7229 064116 016501 000002                MOV     TSSR(R5),R1   ;GET TSSR
7230 064122 012702 000200                MOV     #SSR,R2       ;SET UP EXPECTED TSSR
7231 064126 010004                MOV     R0,R4         ;PACKET ADDRESS SET UP
7232 064130 005237 002212                INC     FATFLG        ;BUMP COUNT
7236 064134                ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
064134 104456                TRAP   C$ERHRD
064136 001244                .WORD  676
064140 073604                .WORD  T26RWN
064142 012136                .WORD  PKTSSR
7237 064144 104406      130$:  CKLOOP                ;LOOP IF SELECTED      .WORD  PKTSSR
064144 104406                TRAP   C$CLP1
7238                ;*****
7239                ;
7240                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7241                ;
7242                ;*****
7243                ;
7244                ;
7245 064146 013701 072150                MOV     T26BFR+6,R1   ;PICK UP XSTO
7246 064152 010102                MOV     R1,R2         ;SET UP EXPECTED
7247 064154 052702 000002                BIS     #BIT1,R2      ;SET BOT BIT IN EXPECTED
7248 064160 020102                CMP     R1,R2         ;DOES EXP = REC'D
7249 064162 001406                BEQ     140$          ;BR, IF EQUAL (OK)
7250 064164 005237 002212                INC     FATFLG        ;BUMP COUNT
7254 064170                ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
064170 104456                TRAP   C$ERHRD
064172 001245                .WORD  677
064174 073315                .WORD  T26BOT
064176 015564                .WORD  EXPREC
7255 064200 104406      140$:  CKLOOP                ;LOOP IF SELECTED      .WORD  PKTSSR
064200 104406                TRAP   C$CLP1
7256 064202 012737 000400 072302                MOV     #256.,T26RSZ ;START RECORD SIZE
7257 064210 000420                BR      150$          ;SKIP SPACE THIS TIME
7258                ;*****
7259                ;
7260                ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
7261                ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
7262                ;
7263                ;

```

```

7264 ;*****
7265
7266 064212 012703 000001 145$: MOV #1,R3 ;SPACE ONE RECORD PARAMETER
7267 064216 004737 010556 JSR PC,SPACE ;CALL SPACE ROUTINE
7268 064222 103413 BCS 150$ ;BR, IF NO PROBLEM WITH SPACE COMMAND
7269 064224 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
7270 064230 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7271 064234 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7272 064236 005237 002212 INC FATFLG ;BUMP COUNT
7276 064242 ERRHRD ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
      064242 104456 TRAP C$ERHRD
      064244 001246 .WORD 678
      064246 072717 .WORD T26SC
      064250 015564 .WORD EXPREC
7277 064252 150$: CKLOOP TRAP C$CLP1
      064252 104406
7278 064254 013703 072302 MOV T26RSZ,R3 ;RECORD SIZE
7279 064260 013737 003114 072252 MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
7280
7281 ;*****
7282 ;
7283 ;REREAD DATA,ACK,CVC=1,SWB COMMAND
7284 ;
7285 ;*****
7286
7287 064266 012737 151401 072250 165$: MOV #151401,T26PK3 ;REREAD DATA,ACK,CVC=1,SWB COMMAND
7288 064274 012704 072250 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7289 064300 010337 072256 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
7290 064304 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
7291 064310 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
7292 064314 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
7293 064320 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
7294 064324 020102 CMP R1,R2 ;ARE THEY EQUAL
7295 064326 001406 BEQ 170$ ;BR, IF OK
7296 064330 005237 002212 INC FATFLG ;BUMP COUNT
7300 064334 ERRHRD ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
      064334 104456 TRAP C$ERHRD
      064336 001247 .WORD 679
      064340 074140 .WORD T26WDC
      064342 012136 .WORD PKTSSR
7301 064344 170$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
      064344 104406
7302 064346 013702 003114 MOV FREE,R2 ;CURRENT BUFFER ADDRESS TO R2
7303 064352 010304 MOV R3,R4 ;CURRENT RECORD SIZE
7304 064354 162704 000400 SUB #256.,R4 ;FIRST LOCATION IN BUFFER
7305 064360 060204 173$: ADD R2,R4 ;SET UP POINTER
7306 064362 021403 CMP (R4),R3 ;CHECK DATA READ (R3=DATA ALSO)
7307 064364 001410 BEQ 180$ ;BR, IF ALL IS WELL
7308 064366 011401 MOV (R4),R1 ;RECD DATA
7309 064370 010302 MOV R3,R2 ;EXPECTED DATA
7310 064372 005237 002212 INC FATFLG ;BUMP COUNT
7314 064376 ERRHRD ERRNO,T26DTA,EXPREC ;DATA READ NOT = WRITTEN
      064376 104456 TRAP C$ERHRD
      064400 001250 .WORD 680
      064402 073362 .WORD T26DTA
      064404 015564 .WORD EXPREC
7315 064406 180$: CKLOOP ;LOOP IF SELECTED

```

```

064406 104406                                TRAP    C#CLP1
7316 064410 005724                          TST     (R4)+     ;BUMP TO NEXT LOCATION
7317 064412 160204                          SUB     R2,R4     ;CORRECT RECORDS SIZE VALUE
7318 064414 020403                          CMP     R4,R3     ;END OF RECORD YET
7319 064416 001360                          BNE     173$     ;BR, IF NOT AT END OF RECORD
7320 064420 005723                          TST     (R3)+     ;BUMP RECORD SIZE
7321 064422 010337 072302                  MOV     R3,T26RSZ ;STORE RECORD SIZE
7322 064426 022703 000410                  CMP     #264.,R3 ;END OF RECORD YET
7323 064432 001267                          BNE     145$     ;BR, IF MORE RECORDS TO WRITE
7324 064434 104406                          190$: CKLOOP    ;LOOP IF SELECTED
064434 104406                                TRAP    C#CLP1
7325 064436                                ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
                                L10112: TRAP    C#ESUB
064436 104403                                TRAP    C#ESUB
7326 064440 023727 002212 000017          CMP     FATFLG,#15. ;IS ERROR COUNT AT 25
7327 064446 103402                          BLO     999$     ;BR, IF LESS THAN 25
7328 064450 004737 017272                  JSR     PC,CKDROP ;TRY TO DROP THE UNIT
7329 064454 999$:
7330
7331 ;+
7332 ;
7333 ;TEST 6. SUBTEST 9
7334 ;
7335 ;VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=1
7336 ;:(READ FORWARD, SPACE REVERSE) AND SWB=0 OPERATES
7337 ;PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN
7338 ;WITH A SERIES OF TEST RECORDS VARYING IN LENGTH AND
7339 ;DATA CONTENT; THE FIRST FOUR BYTES OF EACH RECORD
7340 ;CONTAIN ITS RECORD NUMBER (INDICATING POSITION ON
7341 ;TAPE). THE TAPE IS THEN REWOUND AGAIN. FOR EACH
7342 ;TEST RECORD, THE FOLLOWING SEQUENCE IS EXECUTED.
7343 ;
7344 ;1. THE REREAD NEXT COMMAND WITH OPP=1 IS ISSUED
7345 ; AND THE RESULTS CHECKED
7346 ;
7347 ;2. A READ FORWARD COMMAND IS THEN ISSUED AND THE
7348 ; DATA IS CHECKED TO VERIFY THAT THE TAPE WAS
7349 ; POSITIONED PROPERLY AFTER THE REREAD NEXT
7350 ; COMMAND (E.G. THE TAPE SHOULD HAVE BEEN LEFT
7351 ; POSITIONED AT THE START OF THE TEST RECORD). THE
7352 ; READ FORWARD COMMAND LEAVES THE TAPE POSITIONED
7353 ; PROPERLY AT THE START OF THE NEXT TEST RECORD.
7354 ;
7355 ;THE BYTE COUNT ON EACH REREAD NEXT COMMAND IS SET
7356 ;TO THE LENGTH OF THE EXPECTED RECORD, SO NO
7357 ;EXCEPTIONAL CONDITIONS SHOULD OCCUR.
7358 ;
7359 ;
7360 ;
7361 ;
7362 ;-
7363
7364 064454                                BGNSUB    ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
                                T6.9: TRAP    C#BSUB
064454 104402                                TRAP    C#BSUB
7365 064456 004737 075040                  JSR     PC,T26REST ;SET COMMAND PACKET
7366 064462 005037 072276                  CLR     T26CNT   ;CLEAR TAPE RECORD COUNTER

```

```

7367 064466 004737 075132          JSR    PC,T26RT2          ;SET UP OTHER COMMAND PACKET
7368 064472 004737 075174          JSR    PC,T26RT3          ;SET UP OTHER COMMAND PACKET
7369
7370          ;*****
7371          ;
7372          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7373          ;
7374          ;*****
7375
7376 064476 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
7377 064502 103407                  BCS    20$               ;BR IF INIT WAS OK
7378 064504 005237 002212          INC    FATFLG            ;BUMP COUNT
7382 064510 010001                  MOV    R0,R1             ;CONTENTS OF TSSR REGISTER
7383 064512                  ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   681
                                .WORD   SFIERR
                                .WORD   SFIMSG
7384 064522 013737 002172 072140 20$: MOV    UNITN,T26DSW        ;SET UP UNIT NUMBER
7385
7386 064530 012704 072120          MOV    #T26PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
7387
7388          ;*****
7389          ;
7390          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7391          ;
7392          ;*****
7393
7394 064534 004737 010752          JSR    PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
7395 064540 103407                  BCS    26$               ;BR, IF COMMAND ISSUED OK
7396 064542 005237 002212          INC    FATFLG            ;BUMP COUNT
7400 064546 010001                  MOV    R0,R1             ;SAVE CONTENTS OF TSSR
7401 064550                  ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
                                TRAP    C$ERHRD
                                .WORD   682
                                .WORD   WRTMSG
                                .WORD   SFIMSG
7402 064560                  26$:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C$CLP1
7403
7404          ;*****
7405          ;
7406          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7407          ;
7408          ;*****
7409
7410 064562 004737 011104          JSR    PC,REWIND         ;CALL TAPE REWIND COMMAND
7411 064566 103413                  BCS    30$               ;BR, IF NO PROBLEM
7412 064570 016501 000002          MOV    TSSR(R5),R1      ;GET TSSR
7413 064574 012702 000200          MOV    #SSR,R2          ;SET UP EXPECTED TSSR
7414 064600 010004                  MOV    R0,R4             ;PACKET ADDRESS SET UP
7415 064602 005237 002212          INC    FATFLG            ;BUMP COUNT
7419 064606                  ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   683
                                .WORD   T26RWN
                                .WORD   PKTSSR
064606 104456
064610 001253
064612 073604
064614 012136

```

```

7420 064616          30$:  CKLOOP                      ;LOOP IF SELECTED
      064616 104406                                TRAP      C$CLP1
7421
7422          ;*****
7423          ;
7424          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7425          ;
7426          ;*****
7427
7428 064620 013701 072150          MOV      T26BFR+6,R1          ;PICK UP XSTO
7429 064624 010102          MOV      R1,R2              ;SET UP EXPECTED
7430 064626 052702 000002          BIS      @BIT1,R2          ;SET BOT BIT IN EXPECTED
7431 064632 020102          CMP      R1,R2              ;DOES EXP = REC'D
7432 064634 001406          BEQ     40$                ;BR, IF EQUAL (OK)
7433 064636 005237 002212          INC     FATFLG            ;BUMP COUNT
7437 064642          ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      064642 104456                                TRAP      C$ERHRD
      064644 001254                                .WORD    684
      064646 073315                                .WORD    T26BOT
      064650 015564                                .WORD    EXPREC
7438 064652          40$:  CKLOOP                      ;LOOP IF SELECTED
      064652 104406                                TRAP      C$CLP1
7439 064654 012703 000400          MOV     @256.,R3          ;RECORD SIZE
7440 064660 013737 003114 072252  MOV     FREE,T26RB        ;STARTING WRITE BUFFER ADDRESS
7441
7442          ;*****
7443          ;
7444          ;WRITE DATA,CVC=1,ACK COMMAND
7445          ;
7446          ;*****
7447
7448 064666 012737 140005 072250  MOV     @140005,T26PK3    ;WRITE DATA,CVC=1,ACK COMMAND
7449 064674 012704 072250          MOV     @T26PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
7450 064700          65$:
7451 064700 010337 072256          MOV     R3,T26S7         ;SET UP RECORD SIZE IN PACKET
7452 064704 013777 072276 116202  MOV     T26CNT,@FREE     ;MOVE TAPE RECORD NUMBER TO BUFFER
7453 064712 062737 000001 072276  ADD     @1,T26CNT        ;NUMBER READY FOR NEXT RECORD
7454 064720 010465 000000          MOV     R4,TSDB(R5)      ;ISSUE COMMAND
7455 064724 004737 016340          JSR    PC,WAITF         ;WAIT FOR SSR TO SET
7456 064730 016501 000002          MOV     TSSR(R5),R1      ;GET TSSR CONTENTS
7457 064734 012702 000200          MOV     @SSR,R2         ;SET UP EXPECTED
7458 064740 020102          CMP     R1,R2           ;ARE THEY EQUAL
7459 064742 001406          BEQ     75$             ;BR, IF OK
7460 064744 005237 002212          INC     FATFLG          ;BUMP COUNT
7464 064750          ERRHRD  ERRNO,WRERR,PKTSSR      ;TSSR INCORRECT AFTER REREAD DATA
      064750 104456                                TRAP      C$ERHRD
      064752 001255                                .WORD    685
      064754 005111                                .WORD    WRERR
      064756 012136                                .WORD    PKTSSR
7465 064760          75$:  CKLOOP                      ;LOOP IF SELECTED
      064760 104406                                TRAP      C$CLP1
7466 064762 005723          TST     (R3)+           ;BUMP THE RECORD SIZE
7467 064764 022703 000414          CMP     @268.,R3        ;MAXIMUM SIZE YET
7468 064770 001401          BEQ     120$            ;BR, IF AT END OF WRITE SEQUENCE
7469 064772 000742          BR      65$             ;WRITE MORE RECORDS
7470 064774          120$:
7471 064774 005037 072276          CLR     T26CNT          ;SET RECORD COUNTER BACK TO ZERO

```

```

7472
7473
7474
7475
7476
7477
7478
7479 065000 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7480 065004 103411              BCS      1304           ;BR, IF NO PROBLEM
7481 065006 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR
7482 065012 010004              MOV      R0,R4         ;PACKET ADDRESS SET UP
7483 065014 005237 002212      INC      FATFLG        ;BUMP COUNT
7487 065020              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C1ERHRD
                                .WORD    686
                                .WORD    T26RWN
                                .WORD    PKTSSR
7488 065030 1304:  CKLOOP              ;LOOP IF SELECTED      TRAP      C1CLP1
                                .WORD    104406
7489
7490
7491
7492
7493
7494
7495
7496 065032 013701 072150      MOV      T26BFR+6,R1   ;PICK UP XSTO
7497 065036 010102              MOV      R1,R2         ;SET UP EXPECTED
7498 065040 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
7499 065044 020102              CMP      R1,R2         ;DOES EXP = REC'D
7500 065046 001406              BEQ      1354           ;BR, IF EQUAL (OK)
7501 065050 005237 002212      INC      FATFLG        ;BUMP COUNT
7505 065054              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C1ERHRD
                                .WORD    687
                                .WORD    T26BOT
                                .WORD    EXPREC
7506 065064 1354:  CKLOOP              ;LOOP IF SELECTED      TRAP      C1CLP1
                                .WORD    104406
7507 065066 012737 000400 072302      MOV      @256.,T26RSZ  ;STARTING RECORD SIZE
7508 065074 000420              BR       1404           ;SKIP OVER THE SPACE THIS TIME
7509
7510
7511
7512
7513
7514
7515
7516
7517 065076 012703 000001      1324:  MOV      @000001,R3   ;SET UP SPACE COMMAND (1 FORWARD)
7518 065102 004737 010556      JSR      PC,SPACE      ;CALL SPACE ROUTINE
7519 065106 103413              BCS      1404           ;BR, IF NO TROUBLE
7520 065110 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR
7521 065114 012702 000200      MOV      @SSR,R2       ;SET UP EXPECTED TSSR
7522 065120 010004              MOV      R0,R4         ;PACKET ADDRESS SET UP
7523 065122 005237 002212      INC      FATFLG        ;BUMP COUNT
7527 065126              ERRHRD  ERRNO,T26SC,PKTSSR ;SPACE (FORWARD) FAILED

```





7574 065330 9996:  
7575 ;\*  
7576 ;  
7577 ;  
7578 ;TEST 6, SUBTEST 10  
7579 ;  
7580 ;VERIFIES THAT THE REREAD NEXT COMMAND WITH OPP=1  
7581 ;AND SMB=1 OPERATES PROPERLY. THE TEST SEQUENCE IS  
7582 ;THE SAME THAT IS USED IN SUBTEST 3, BUT IT IS  
7583 ;VERIFIED THAT DATA STORED BY THE COMMAND CONTAINS  
7584 ;SWAPPED BYTES.  
7585 ;  
7586 ;  
7587 ;  
7588 ;  
7589 ;-  
7590

7591 065330 BGNSUB ;>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>  
065330 104402 T6.10:  
7592 065332 004737 075040 JSR PC,T26REST ;SET COMMAND PACKET TRAP C:BSUB  
7593 065336 005037 072276 CLR 126CNT ;CLEAR TAPE RECORD COUNTER  
7594 065342 004737 075132 JSR PC,T26RT2 ;SET UP OTHER COMMAND PACKET  
7595 065346 004737 075174 JSR PC,T26RT3 ;SET UP OTHER COMMAND PACKET  
7596  
7597 ;\*\*\*\*\*  
7598 ;  
7599 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR  
7600 ;  
7601 ;\*\*\*\*\*  
7602  
7603 065352 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER  
7604 065356 103407 BCS 20: ;BR IF INIT WAS OK  
7605 065360 005237 002212 INC FATFLG ;BUMP COUNT  
7609 065364 010001 MOV RO,R1 ;CONTENTS OF TSSR REGISTER  
7610 065366 104455 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK  
065370 001263 TRAP C:ERDF  
065372 003650 .WORD 691  
065374 012124 .WORD SFIERR  
7611 065376 013737 002172 072140 20: MOV UNITN,T26DSW ;SET UP UNIT NUMBER  
7612  
7613 065404 012704 072120 MOV @T26PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS  
7614  
7615 ;\*\*\*\*\*  
7616 ;  
7617 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)  
7618 ;  
7619 ;\*\*\*\*\*  
7620  
7621 065410 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS  
7622 065414 103407 BCS 26: ;BR, IF COMMAND ISSUED OK  
7623 065416 005237 002212 INC FATFLG ;BUMP COUNT  
7627 065422 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR  
7628 065424 104456 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED  
065426 001264 TRAP C:ERHRD  
 .WORD 692

```

065430 005054 .WORD WRTMSG
065432 012124 .WORD SFIMSG
7629 065434 104406 26: CKLOOP ;LOOP IF SELECTED TRAP C:CLP1
065434 104406
7630 ;*****
7631 ;
7632 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7633 ;
7634 ;*****
7635 ;
7636 ;
7637 065436 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
7638 065442 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
7639 065446 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7640 065452 103407 BCS 30: ;BR, IF NO PROBLEM
7641 065454 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7642 065456 005237 002212 INC FATFLG ;BUMP COUNT
7646 065462 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
065462 104456 TRAP C:ERHRD
065464 001265 .WORD 693
065466 073604 .WORD T26RWN
065470 012136 .WORD PKTSSR
7647 065472 104406 30: CKLOOP ;LOOP IF SELECTED TRAP C:CLP1
065472 104406
7648 ;*****
7649 ;
7650 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7651 ;
7652 ;*****
7653 ;
7654 ;
7655 065474 013701 072150 MOV T26FR-6,R1 ;PICK UP XSTO
7656 065500 010102 MOV R1,R2 ;SET UP EXPECTED
7657 065502 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
7658 065506 020102 CMP R1,R2 ;DOES EXP = REC'D
7659 065510 001406 BEQ 40: ;BR, IF EQUAL (OK)
7660 065512 005237 002212 INC FATFLG ;BUMP COUNT
7664 065516 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT 30T AFTER REWIND
065516 104456 TRAP C:ERHRD
065520 001266 .WORD 694
065522 073315 .WORD T26BOT
065524 015564 .WORD EXPREC
7665 065526 104406 40: CKLOOP ;LOOP IF SELECTED TRAP C:CLP1
065526 104406
7666 065530 012703 000400 MOV #256.,R3 ;RECORD SIZE
7667 065534 013737 003114 072252 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
7668 ;*****
7669 ;
7670 ;WRITE DATA,CVC=1,ACK COMMAND
7671 ;
7672 ;*****
7673 ;
7674 ;
7675 065542 012737 140005 072250 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
7676 065550 012704 072250 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7677 065554
7678 065554 010337 072256 65: MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET

```

```

7679 065560 013777 072276 115326      MOV      T26CNT,0FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
7680 065566 062737 000001 072276      ADD      #1,T26CNT        ;NUMBER READY FOR NEXT RECORD
7681 065574 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
7682 065600 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
7683 065604 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
7684 065610 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED
7685 065614 020102      CMP      R1,R2           ;ARE THEY EQUAL
7686 065616 001406      BEQ      75$             ;BR, IF OK
7687 065620 005237 002212      INC      FATFLG          ;BUMP COUNT
7691 065624      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    695
                                .WORD    WRERR
                                .WORD    PKTSSR
7692 065634      75$:   CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
7693 065636 005723      TST      (R3).           ;BUMP THE RECORD SIZE
7694 065640 022703 000414      CMP      #268.,R3       ;MAXIMUM SIZE YET
7695 065644 001401      BEQ      120$           ;BR, IF AT END OF WRITE SEQUENCE
7696 065646 000742      BR       65$            ;WRITE MORE RECORDS
7697 065650      120$: CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
7698 065650 005037 072276
7699
7700      ;*****
7701      ;
7702      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7703      ;
7704      ;*****
7705
7706 065654 004737 011104      JSR      PC,REWIND       ;CALL TAPE REWIND COMMAND
7707 065660 103411      BCS      130$           ;BR, IF NO PROBLEM
7708 065662 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
7709 065666 010004      MOV      R0,R4           ;PACKET ADDRESS SET UP
7710 065670 005237 002212      INC      FATFLG          ;BUMP COUNT
7714 065674      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    696
                                .WORD    T26RWN
                                .WORD    PKTSSR
7715 065704      130$: CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
7716
7717      ;*****
7718      ;
7719      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7720      ;
7721      ;*****
7722
7723 065706 013701 072150      MOV      T268FR+6,R1     ;PICK UP XSTO
7724 065712 010102      MOV      R1,R2           ;SET UP EXPECTED
7725 065714 052702 000002      BIS      #BIT1,R2        ;SET BOT BIT IN EXPECTED
7726 065720 020102      CMP      R1,R2           ;DOES EXP = REC'D
7727 065722 001406      BEQ      135$           ;BR, IF EQUAL (OK)
7728 065724 005237 002212      INC      FATFLG          ;BUMP COUNT
7732 065730      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    697
065730 104456
065732 001271

```

```

065734 073315 .WORD T26BOT
065736 015564 .WORD EXPREC
7733 065740 135$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
065740 104406 ;START RECORD SIZE
7734 065742 012737 000400 072302 MOV #256.,T26RSZ ;SKIP OVER SPACE
7735 065750 000420 BR 140$
7736
7737 ;*****
7738 ;
7739 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
7740 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
7741 ;
7742 ;*****
7743
7744 065752 012703 000001 136$: MOV #000001,R3 ;SET UP SPACE COMMAND (1 FORWARD)
7745 065756 004737 010556 JSR PC,SPACE ;CALL SPACE ROUTINE
7746 065762 103413 BCS 140$ ;BR, IF NO TROUBLE
7747 065764 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
7748 065770 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
7749 065774 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
7750 065776 005237 002212 INC FATFLG ;BUMP COUNT
7754 066002 ERRHRD ERRNO,T26SC,PKTSSR ;SPACE (FORWARD) FAILED
066002 104456 TRAP C$ERHRD
066004 001272 .WORD 698
066006 072717 .WORD T26SC
066010 012136 .WORD PKTSSR
7755 066012 140$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
066012 104406 ;RECORD SIZE
7756 066014 013703 072302 MOV T26RSZ,R3 ;STARTING READ BUFFER ADDRESS
7757 066020 013737 003114 072252 150$: MOV FREE,T26R8
7758
7759 ;*****
7760 ;
7761 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
7762 ;
7763 ;*****
7764
7765 066026 012737 161401 072250 165$: MOV #161401,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
7766 066034 012704 072250 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7767 066040 010337 072256 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
7768 066044 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
7769 066050 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
7770 066054 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
7771 066060 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
7772 066064 020102 CMP R1,R2 ;ARE THEY EQUAL
7773 066066 001406 BEQ 170$ ;BR, IF OK
7774 066070 005237 002212 INC FATFLG ;BUMP COUNT
7778 066074 ERRHRD ERRNO,T26RRF,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
066074 104456 TRAP C$ERHRD
066076 001273 .WORD 699
066100 072525 .WORD T26RRF
066102 012136 .WORD PKTSSR
7779 066104 170$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
066104 104406 ;FIRST WORD FROM READ BUFFER
7780 066106 017701 115002 MOV #FREE,R1 ;SET UP EXPECTED
7781 066112 013702 072276 MOV T26CNT,R2 ;IS TAPE POSITION CORRECT
7782 066116 020102 CMP R1,R2

```

```
7783 066120 001406          BEQ     190$          ;KEEP GOING POSITION OK
7784 066122 005237 002212    INC     FATFLG        ;BUMP COUNT
7788 066126          ERRMRD  ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
           066126 104456                                     TRAP   C#ERHRD
           066130 001274                                     .WORD  700
           066132 072306                                     .WORD  T26WNG
           066134 015564                                     .WORD  EXPREC
7789 066136          190$:   CKLOOP                                     TRAP   C#CLP1
           066136 104406
7790 066140 062737 000001 072276  ADD     #1,T26CNT     ;BUMP TAPE RECORD COUNTER
7791 066146 005723          TST     (R3)+         ;NEXT RECORD SIZE
7792 066150 010337 072302    MOV     R3,T26RSZ    ;STORE RECORD SIZE
7793 066154 022703 000412    CMP     #266.,R3    ;AT MAX SIZE YET
7794 066160 001402          BEQ     220$          ;BR, IF AT END OF THE SUBTEST
7795 066162 000137 065752    JMP     136$         ;KEEP GOING MORE RECORDS
7796 066166          220$:   ENDSUB                                     ;>>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
7797 066166          L10114:
           066166 104403                                     TRAP   C#ESUB
7798 066170 023727 002212 000017  CMP     FATFLG,#15.  ;IS ERROR COUNT AT 25
7799 066176 103402          BLQ     999$         ;BR, IF LESS THAN 25
7800 066200 004737 017272    JSR     PC,CKDROP   ;TRY TO DROP THE UNIT
7801 066204          999$:
7802
7803          ;*
7804          ;
7805          ;TEST 6. SUBTEST 11
7806          ;
7807          ;VERIFIES THAT A REREAD NEXT COMMAND READING A
7808          ;RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES
7809          ;TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH
7810          ;LONG (RLL) BIT SET. RESULTS ARE VERIFIED FOR BOTH
7811          ;STATES OF OPP (0 AND 1).
7812          ;
7813          ;
7814          ;
7815          ;-
7816
7817
7818 066204          BGNSUB                                     ;>>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
           066204          T6.11:
           066204 104402                                     TRAP   C#BSUB
7819 066206 004737 075040    JSR     PC,T26REST   ;SET COMMAND PACKET
7820 066212 004737 075132    JSR     PC,T26RT2    ;SET UP OTHER COMMAND PACKET
7821 066216 004737 075174    JSR     PC,T26RT3    ;SET UP OTHER COMMAND PACKET
7822
7823          ;*****
7824          ;
7825          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7826          ;
7827          ;*****
7828
7829 066222 004737 016064    JSR     PC,SOFINIT   ;DO INITIALIZE ON CONTROLLER
7830 066226 103407          BCS     20$         ;BR IF INIT WAS OK
7831 066230 005237 002212    INC     FATFLG        ;BUMP COUNT
7835 066234 010001          MOV     R0,R1        ;CONTENTS OF TSSR REGISTER
7836 066236          ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
```

TSV7 - HARDWARE TESTS 1-8  
TEST 6: REREADS

MACRO M1113 14-JUN-84 14:17

SEQ 0252

```

066236 104455                                TRAP    C$ERDF
066240 001275                                .WORD  701
066242 003650                                .WORD  SFIERR
066244 012124                                .WORD  SFIMSG
7837 066246 013737 002172 072140 20$:  MOV    UNITN,T26DSW          ;SET UP UNIT NUMBER
7838                                         ;
7839 066254 012704 072120                MOV    @T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
7840                                         ;
7841                                         ;*****
7842                                         ;
7843                                         ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
7844                                         ;
7845                                         ;*****
7846                                         ;
7847 066260 004737 010752                JSR    PC,WRTPHR          ;ISSUE WRITE CHARACTERISTICS
7848 066264 103407                        BCS    26$                ;BR, IF COMMAND ISSUED OK
7849 066266 005237 002212                INC    FATFLG             ;BUMP COUNT
7853 066272 010001                        MOV    R0,R1              ;SAVE CONTENTS OF TSSR
7854 066274                                ERRHRD  ERRNO,WRTPHR,SFIMSG ;WRITE CHARACTERISTICS FAILED
066274 104456                                TRAP    C$ERRHRD
066276 001276                                .WORD  702
066300 005054                                .WORD  WRTPHR
066302 012124                                .WORD  SFIMSG
7855 066304                                26$:  CKLOOP                ;LOOP IF SELECTED
066304 104406                                TRAP    C$CLP1
7856                                         ;
7857                                         ;*****
7858                                         ;
7859                                         ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7860                                         ;
7861                                         ;*****
7862                                         ;
7863 066306 004737 011104                JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
7864 066312 016501 000002                MOV    TSSR(R5),R1        ;GET TSSR
7865 066316 012702 000200                MOV    @SSR,R2           ;SET UP EXPECTED TSSR
7866 066322 103407                        BCS    30$                ;BR, IF NO PROBLEM
7867 066324 010004                        MOV    R0,R4             ;PACKET ADDRESS SET UP
7868 066326 005237 002212                INC    FATFLG             ;BUMP COUNT
7872 066332                                ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
066332 104456                                TRAP    C$ERRHRD
066334 001277                                .WORD  703
066336 073604                                .WORD  T26RWN
066340 012136                                .WORD  PKTSSR
7873 066342                                30$:  CKLOOP                ;LOOP IF SELECTED
066342 104406                                TRAP    C$CLP1
7874                                         ;
7875                                         ;*****
7876                                         ;
7877                                         ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7878                                         ;
7879                                         ;*****
7880                                         ;
7881 066344 013701 072150                MOV    T26BFR+6,R1        ;PICK UP XSTO
7882 066350 010102                        MOV    R1,R2             ;SET UP EXPECTED
7883 066352 052702 000002                BIS    @BIT1,R2          ;SET BOT BIT IN EXPECTED
7884 066356 020102                        CMP    R1,R2             ;DOES EXP = REC'D
7885 066360 001406                        BEQ    40$                ;BR, IF EQUAL (OK)

```

```

7886 066362 005237 002212          INC    FATFLG          ;BUMP COUNT
7890 066366          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      066366 104456          TRAP    C$ERHRD
      066370 001300          .WORD  704
      066372 073315          .WORD  T26BOT
      066374 015564          .WORD  EXPREC
7891 066376          40$:   CKLOOP          ;LOOP IF SELECTED
      066376 104406          TRAP    C$CLP1
7892 066400 012703 001000          MOV    #512.,R3        ;RECORD SIZE
7893 066404 013737 003114 072252          MOV    FREE,T26RB      ;STARTING WRITE BUFFER ADDRESS
7894
7895          ;*****
7896          ;
7897          ;WRITE DATA,CVC=1,ACK COMMAND
7898          ;
7899          ;*****
7900
7901 066412 012737 140005 072250          MOV    #140005,T26PK3  ;WRITE DATA,CVC=1,ACK COMMAND
7902 066420 012704 072250          MOV    #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
7903 066424          65$:
7904 066424 010337 072256          MOV    R3,T26SZ       ;SET UP RECORD SIZE IN PACKET
7905 066430 010465 000000          MOV    R4,TSDB(R5)    ;ISSUE COMMAND
7906 066434 004737 016340          JSR    PC,WAITF       ;WAIT FOR SSR TO SET
7907 066440 016501 000002          MOV    TSSR(R5),R1    ;GET TSSR CONTENTS
7908 066444 012702 000200          MOV    #SSR,R2       ;SET UP EXPECTED
7909 066450 020102          CMP    R1,R2         ;ARE THEY EQUAL
7910 066452 001406          BEQ    75$           ;BR, IF OK
7911 066454 005237 002212          INC    FATFLG          ;BUMP COUNT
7915 066460          ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      066460 104456          TRAP    C$ERHRD
      066462 001301          .WORD  705
      066464 005111          .WORD  WRERR
      066466 012136          .WORD  PKTSSR
7916 066470          75$:   CKLOOP          ;LOOP IF SELECTED
      066470 104406          TRAP    C$CLP1
7917
7918          ;*****
7919          ;
7920          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7921          ;
7922          ;*****
7923
7924 066472 004737 011104          JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
7925 066476 016501 000002          MOV    TSSR(R5),R1    ;GET TSSR
7926 066502 012702 000200          MOV    #SSR,R2       ;SET UP EXPECTED TSSR
7927 066506 103407          BCS    130$          ;BR, IF NO PROBLEM
7928 066510 010004          MOV    R0,R4         ;PACKET ADDRESS SET UP
7929 066512 005237 002212          INC    FATFLG          ;BUMP COUNT
7933 066516          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      066516 104456          TRAP    C$ERHRD
      066520 001302          .WORD  706
      066522 073604          .WORD  T26RWN
      066524 012136          .WORD  PKTSSR
7934 066526          130$:  CKLOOP          ;LOOP IF SELECTED
      066526 104406          TRAP    C$CLP1
7935
7936          ;*****

```

```

7937 ;
7938 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7939 ;
7940 ;*****
7941 ;
7942 066530 013701 072150          MOV      T26BFR+6,R1          ;PICK UP XSTO
7943 066534 010102                MOV      R1,R2              ;SET UP EXPECTED
7944 066536 052702 000002        BIS      @BIT1,R2           ;SET BOT BIT IN EXPECTED
7945 066542 020102                CMP      R1,R2              ;DOES FYP = REC'D
7946 066544 001406                BEQ      140$               ;BR. IF EQUAL (OK)
7947 066546 005237 002212        INC      FATFLG             ;BUMP COUNT
7951 066552                ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    707
                                .WORD    T26BOT
                                .WORD    EXPREC
7952 066562                140$: CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
7953 066564 005303                DEC      R3                  ;SET RECORD SIZE TO 511.
7954 066566 013737 003114 072252  MOV      FREE,T26RB         ;STARTING READ BUFFER ADDRESS
7955 ;
7956 ;*****
7957 ;
7958 ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
7959 ;
7960 ;*****
7961 ;
7962 066574 012737 161401 072250  MOV      @161401,T26PK3     ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
7963 066602 012704 072250        165$: MOV      @T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
7964 066606 010337 072256        MOV      R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
7965 066612 010465 000000        MOV      R4,TSDB(R5)       ;ISSUE COMMAND
7966 066616 004737 016340        JSR      PC,WAITF          ;WAIT FOR SSR TO SET
7967 066622 016501 000002        MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
7968 066626 012702 100204        MOV      @SSR!SC!BIT2,R2   ;SET UP EXPECTED
7969 066632 020102                CMP      R1,R2              ;ARE THEY EQUAL
7970 066634 001406                BEQ      170$               ;BR, IF OK
7971 066636 005237 002212        INC      FATFLG             ;BUMP COUNT
7975 066642                ERRHRD  ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C$ERHRD
                                .WORD    708
                                .WORD    T26TRL
                                .WORD    PKTSSR
7976 066652                170$: CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
7977 ;
7978 ;*****
7979 ;
7980 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7981 ;
7982 ;*****
7983 ;
7984 066654 013701 072150          MOV      T26BFR+6,R1          ;GET MESSAGE BUFFER
7985 066660 010102                MOV      R1,R2              ;SET UP EXPECTED
7986 066662 052702 010000        BIS      @BIT12,R2         ;SET THE RLL BIT IN EXPECTED
7987 066666 020102                CMP      R1,R2              ;ARE THEY EQUAL
7988 066670 001406                BEQ      180$               ;BR, IF EQUAL (ALL IS WELL)
7989 066672 005237 002212        INC      FATFLG             ;BUMP COUNT

```







```

8096 067144          ERRHRD  ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTISC FAILED
      067144 104456          TRAP                   C$ERHRD
      067146 001311          .WORD                   713
      067150 005054          .WORD                   WRTMSG
      067152 012124          .WORD                   SFIMSG
8097 067154          26$:  CKLOOP                    ;LOOP IF SELECTED
      067154 104406          TRAP                   C$CLP1
8098
8099          ;*****
8100          ;
8101          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8102          ;
8103          ;*****
8104
8105 067156 004737 011104      JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
8106 067162 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR
8107 067166 012702 000200      MOV      @SSR,R2         ;SET UP EXPECTED TSSR
8108 067172 103407              BCS      30$              ;BR, IF NO PROBLEM
8109 067174 010004              MOV      R0,R4            ;PACKET ADDRESS SET UP
8110 067176 005237 002212      INC      FATFLG          ;BUMP COUNT
8114 067202          ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      067202 104456          TRAP                   C$ERHRD
      067204 001312          .WORD                   714
      067206 073604          .WORD                   T26RWN
      067210 012136          .WORD                   PKTSSR
8115 067212          30$:  CKLOOP                    ;LOOP IF SELECTED
      067212 104406          TRAP                   C$CLP1
8116
8117          ;*****
8118          ;
8119          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8120          ;
8121          ;*****
8122
8123 067214 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO
8124 067220 010102              MOV      R1,R2           ;SET UP EXPECTED
8125 067222 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
8126 067226 020102              CMP      R1,R2           ;DOES EXP = REC'D
8127 067230 001406              BEQ      40$              ;BR, IF EQUAL (OK)
8128 067232 005237 002212      INC      FATFLG          ;BUMP COUNT
8132 067236          ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      067236 104456          TRAP                   C$ERHRD
      067240 001313          .WORD                   715
      067242 073315          .WORD                   T26BOT
      067244 015564          .WORD                   EXPREC
8133 067246          40$:  CKLOOP                    ;LOOP IF SELECTED
      067246 104406          TRAP                   C$CLP1
8134 067250 012703 000400      MOV      @256.,R3        ;RECORD SIZE
8135 067254 013737 003114 072252  MOV      FREE,T26RB       ;STARTING WRITE BUFFER ADDRESS
8136
8137          ;*****
8138          ;
8139          ;WRITE DATA,CVC=1,ACK COMMAND
8140          ;
8141          ;*****
8142
8143 067262 012737 140005 072250  MOV      @140005,T26PK3   ;WRITE DATA,CVC=1,ACK COMMAND

```

```

8144 067270 012704 072250          MOV    #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
8145 067274          65$:  MOV    R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
8146 067274 010337 072256          MOV    R4,TSDB(R5)        ;ISSUE COMMAND
8147 067300 010465 000000          JSR    PC,WAITF          ;WAIT FOR SSR TO SET
8148 067304 004737 016340          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
8149 067310 016501 000002          MOV    #SSR,R2          ;SET UP EXPECTED
8150 067314 012702 000200          CMP    R1,R2            ;ARE THEY EQUAL
8151 067320 020102          BEQ    75$              ;BR, IF OK
8152 067322 001406          INC    FATFLG           ;BUMP COUNT
8153 067324 005237 002212          ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
8157 067330          TRAP  C#ERHRD
      067330 104456          .WORD 716
      067332 001314          .WORD WRERR
      067334 005111          .WORD PKTSSR
      067336 012136
8158 067340          75$:  CKLOOP              ;LOOP IF SELECTED          TRAP  C#CLP1
      067340 104406
8159 067342          120$:
8160
8161          ;*****
8162          ;
8163          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8164          ;
8165          ;*****
8166
8167 067342 004737 011104          JSR    PC,REWIND         ;CALL TAPE REWIND COMMAND
8168 067346 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR
8169 067352 012702 000200          MOV    #SSR,R2          ;SET UP EXPECTED TSSR
8170 067356 103407          BCS    130$            ;BR, IF NO PROBLEM
8171 067360 010004          MOV    R0,R4            ;PACKET ADDRESS SET UP
8172 067362 005237 002212          INC    FATFLG           ;BUMP COUNT
8176 067366          ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      067366 104456          TRAP  C#ERHRD
      067370 001315          .WORD 717
      067372 073604          .WORD T26RWN
      067374 012136          .WORD PKTSSR
8177 067376          130$:  CKLOOP              ;LOOP IF SELECTED          TRAP  C#CLP1
      067376 104406
8178
8179          ;*****
8180          ;
8181          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8182          ;
8183          ;*****
8184
8185 067400 013701 072150          MOV    T268FR+6,R1       ;PICK UP XST0
8186 067404 010102          MOV    R1,R2            ;SET UP EXPECTED
8187 067406 052702 000002          BIS    #BIT1,R2         ;SET BOT BIT IN EXPECTED
8188 067412 020102          CMP    R1,R2            ;DOES EXP = REC'D
8189 067414 001406          BEQ    135$            ;BR, IF EQUAL (OK)
8190 067416 005237 002212          INC    FATFLG           ;BUMP COUNT
8194 067422          ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      067422 104456          TRAP  C#ERHRD
      067424 001316          .WORD 718
      067426 073315          .WORD T26BOT
      067430 015564          .WORD EXPREC
8195 067432          135$:  CKLOOP              ;LOOP IF SELECTED

```

```

067432 104406
8196 067434 012703 001000          MOV    #512.,R3          ;RECORD SIZE          TRAP    C#CLP1
8197 067440 013737 003114 072252  MOV    FREE,T26RB      ;STARTING READ BUFFER ADDRESS
8198
8199
8200
8201          ;*****
8202          ;REREAD NEXT,ACK,CVC=1,OPP=1
8203          ;*****
8204
8205 067446 012737 161401 072250          MOV    #161401,T26PK3  ;REREAD NEXT,ACK,CVC=1,OPP=1
8206 067454 012704 072250          165$: MOV    #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
8207 067460 010337 072256          MOV    R3,T26SZ       ;SET UP RECORD SIZE IN PACKET
8208 067464 010465 000000          MOV    R4,TSD8(R5)    ;ISSUE COMMAND
8209 067470 004737 016340          JSR    PC,WAITF       ;WAIT FOR SSR TO SET
8210 067474 016501 000002          MOV    TSSR(R5),R1    ;GET TSSR CONTENTS
8211 067500 012702 100204          MOV    #SSR!SC!BIT2,R2 ;SET UP EXPECTED
8212 067504 020102          CMP    R1,R2         ;ARE THEY EQUAL
8213 067506 001406          BEQ    170$         ;BR, IF OK
8214 067510 005237 002212          INC    FATFLG        ;BUMP COUNT
8218 067514          ERRHRD  ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C#ERHRD
                                .WORD  719
                                .WORD  T26TRL
                                .WORD  PKTSSR
                                TRAP    C#CLP1
8219 067524          170$: CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
8220
8221          ;*****
8222          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8223          ;*****
8224
8225
8226
8227 067526 013701 072150          MOV    T26BFR+6,R1    ;GET MESSAGE BUFFER
8228 067532 010102          MOV    R1,R2         ;SET UP EXPECTED
8229 067534 052702 040000          BIS    #BIT14,R2     ;SET THE RLS BIT IN EXPECTED
8230 067540 020102          CMP    R1,R2         ;ARE THEY EQUAL
8231 067542 001406          BEQ    180$         ;BR, IF EQUAL (ALL IS WELL)
8232 067544 005237 002212          INC    FATFLG        ;BUMP COUNT
8236 067550          ERRHRD  ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
                                TRAP    C#ERHRD
                                .WORD  720
                                .WORD  T26LOP
                                .WORD  EXPREC
                                TRAP    C#CLP1
8237 067560          180$: CKLOOP
8238 067562 013701 072146          MOV    T26BFR+4,R1    ;PICK UP RESIDUAL BYTE COUNTER
8239 067566 012702 000400          MOV    #256.,R2      ;THIS SHOULD BE THE DIFFERENCE
8240 067572 020102          CMP    R1,R2         ;IS THE DIFFERENCE CORRECT
8241 067574 001405          BEQ    190$         ;BR, IF CORRECT
8245 067600          ERRHRD  ERRNO,T26PBP,EXPREC ;RBPCT NOT CORRECT
                                TRAP    C#ERHRD
                                .WORD  720
                                .WORD  T26PBP
                                .WORD  EXPREC
8246 067610          190$: CKLOOP          ;LOOP IF SELECTED

```

```

067610 104406
8247 067612 012703 001000      MOV      #512.,R3      ;RECORD SIZE
8248 067616 013737 003114 072252  MOV      FREE,T26RB   ;STARTING READ BUFFER ADDRESS
8249
8250      ;*****
8251      ;
8252      ;REREAD NEXT,ACK,CVC=1,OPP=0
8253      ;
8254      ;*****
8255
8256 067624 012737 141401 072250      MOV      #141401,T26PK3 ;REREAD NEXT,ACK,CVC=1,OPP=0
8257 067632 012704 072250      MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
8258 067636 010337 072256      MOV      R3,T26SZ     ;SET UP RECORD SIZE IN PACKET
8259 067642 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
8260 067646 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
8261 067652 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
8262 067656 012702 100204      MOV      #SSR!SC!BIT2,R2 ;SET UP EXPECTED
8263 067662 020102                CMP      R1,R2        ;ARE THEY EQUAL
8264 067664 001406                BEQ      270#         ;BR, IF OK
8265 067666 005237 002212      INC      FATFLG       ;BUMP COUNT
8269 067672                ERRHRD  ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C!ERHRD
                                .WORD    721
                                .WORD    T26TRL
                                .WORD    PKTSSR
067672 104456
067674 001321
067676 074662
067700 012136
8270 067702                270# :   CKLOOP      ;LOOP IF SELECTED
                                TRAP      C!CLP1
067702 104406
8271
8272      ;*****
8273      ;
8274      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8275      ;
8276      ;*****
8277
8278 067704 013701 072150      MOV      T26BFR+6,R1  ;GET MESSAGE BUFFER
8279 067710 010102      MOV      R1,R2        ;SET UP EXPECTED
8280 067712 052702 040000      BIS      #BIT14,R2    ;SET THE RLS BIT IN EXPECTED
8281 067716 020102      CMP      R1,R2        ;ARE THEY EQUAL
8282 067720 001406      BEQ      280#         ;BR, IF EQUAL (ALL IS WELL)
8283 067722 005237 002212      INC      FATFLG       ;BUMP COUNT
8287 067726      ERRHRD  ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
                                TRAP      C!ERHRD
                                .WORD    722
                                .WORD    T26LOP
                                .WORD    EXPREC
067726 104456
067730 001322
067732 074512
067734 015564
8288 067736                280# :   CKLOOP
                                TRAP      C!CLP1
067736 104406
8289 067740 013701 072146      MOV      T26BFR+4,R1  ;PICK UP RESIDUAL BYTE COUNTER
8290 067744 012702 000400      MOV      #256.,R2    ;THIS SHOULD BE THE DIFFERENCE
8291 067750 020102      CMP      R1,R2        ;IS THE DIFFERENCE CORRECT
8292 067752 001405      BEQ      290#         ;BR, IF CORRECT
8296 067756      ERRHRD  ERRNO,T26PBP,EXPREC ;RBPBR NOT CORRECT
                                TRAP      C!ERHRD
                                .WORD    722
                                .WORD    T26PBP
                                .WORD    EXPREC
067756 104456
067760 001322
067762 074574
067764 015564
8297 067766                290# :   CKLOOP      ;LOOP IF SELECTED

```



```

8349 070104 103407          BCS      26:          ;BR, IF COMMAND ISSUED OK
8350 070106 005237 002212  INC      FATFLG      ;BUMP COUNT
8354 070112 010001          MOV      R0,R1        ;SAVE CONTENTS OF TSSR
8355 070114          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      070114 104456          TRAP      C:ERHRD
      070116 001324          .WORD    724
      070120 005054          .WORD    WRTMSG
      070122 012124          .WORD    SFIMSG
8356 070124          26:      CKLOOP          ;LOOP IF SELECTED
      070124 104406          TRAP      C:CLP1
8357
8358 ;*****
8359 ;
8360 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8361 ;
8362 ;*****
8363
8364 070126 004737 021232  JSR      PC,INVERT    ;INVERT THE EXTENDED FEATURES SWITCH
8365 070132 004737 011104  JSR      PC,REWIND    ;CALL TAPE REWIND COMMAND
8366 070136 103411          BCS      30:          ;BR, IF NO PROBLEM
8367 070140 016501 000002  MOV      TSSR(R5),R1  ;GET TSSR
8368 070144 010004          MOV      R0,R4        ;PACKET ADDRESS SET UP
8369 070146 005237 002212  INC      FATFLG      ;BUMP COUNT
8373 070152          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      070152 104456          TRAP      C:ERHRD
      070154 001325          .WORD    725
      070156 073604          .WORD    T26RWN
      070160 012136          .WORD    PKTSSR
8374 070162          30:      CKLOOP          ;LOOP IF SELECTED
      070162 104406          TRAP      C:CLP1
8375
8376 ;*****
8377 ;
8378 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8379 ;
8380 ;*****
8381
8382 070164 013701 072150  MOV      T26BFR+6,R1  ;PICK UP XST0
8383 070170 010102          MOV      R1,R2        ;SET UP EXPECTED
8384 070172 052702 000002  BIS      @BIT1,R2     ;SET BOT BIT IN EXPECTED
8385 070176 020102          CMP      R1,R2        ;DOES EXP = REC'D
8386 070200 001406          BEQ      40:          ;BR, IF EQUAL (OK)
8387 070202 005237 002212  INC      FATFLG      ;BUMP COUNT
8391 070206          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      070206 104456          TRAP      C:ERHRD
      070210 001326          .WORD    726
      070212 073315          .WORD    T26BOT
      070214 015564          .WORD    EXPREC
8392 070216          40:      CKLOOP          ;LOOP IF SELECTED
      070216 104406          TRAP      C:CLP1
8393 070220 013737 003114 072252  MOV      FREE,T26RB   ;STARTING WRITE BUFFER ADDRESS
8394
8395 ;*****
8396 ;
8397 ;WRITE DATA,CVC=1,ACK COMMAND
8398 ;
8399 ;*****

```



```

8400
8401 070226 012737 140005 072250      MOV      #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
8402 070234 012704 072250      MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
8403 070240 012737 000400 072256 65$:  MOV      #256.,T26SZ        ;SET UP RECORD SIZE IN PACKET
8404 070246 013777 072276 112640      MOV      T26CNT,#FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
8405 070254 062737 000001 072276      ADD      #1,T26CNT         ;NUMBER READY FOR NEXT RECORD
8406 070262 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
8407 070266 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
8408 070272 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
8409 070276 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
8410 070302 020102      CMP      R1,R2             ;ARE THEY EQUAL
8411 070304 001406      BEQ      75$              ;BR, IF OK
8412 070306 005237 002212      INC      FATFLG            ;BUMP COUNT
8416 070312      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      070312 104456      TRAP    C#ERHRD
      070314 001327      .WORD  727
      070316 005111      .WORD  WRERR
      070320 012136      .WORD  PKTSSR
8417 070322      75$:  CKLOOP                ;LOOP IF SELECTED
      070322 104406      TRAP    C#CLP1
8418 070324 022737 000013 072276      CMP      #11.,T26CNT       ;CHECK NUMBER OF RECORDS WRITTEN
8419 070332 001401      BEQ      120$             ;BR, IF AT END OF WRITE SEQUENCE
8420 070334 000741      BR      65$              ;WRITE MORE RECORDS
8421 070336      120$:
8422 070336 005037 003132      CLR      NXMMI            ;SET TO 16 BIT ADDRESS
8423 070342      125$:
8424 070342 005037 072276      CLR      T26CNT           ;SET RECORD COUNTER BACK TO ZERO
8425
8426 ;*****
8427 ;
8428 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8429 ;
8430 ;*****
8431
8432 070346 004737 011104      JSR      PC,REWIND         ;CALL TAPE REWIND COMMAND
8433 070352 103411      BCS     130$              ;BR, IF NO PROBLEM
8434 070354 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR
8435 070360 010004      MOV      R0,R4            ;PACKET ADDRESS SET UP
8436 070362 005237 002212      INC      FATFLG            ;BUMP COUNT
8440 070366      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      070366 104456      TRAP    C#ERHRD
      070370 001330      .WORD  728
      070372 073604      .WORD  T26RWN
      070374 012136      .WORD  PKTSSR
8441 070376      130$:  CKLOOP                ;LOOP IF SELECTED
      070376 104406      TRAP    C#CLP1
8442
8443 ;*****
8444 ;
8445 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8446 ;
8447 ;*****
8448
8449 070400 013701 072150      MOV      T26BFR+6,R1       ;PICK UP XSTO
8450 070404 010102      MOV      R1,R2            ;SET UP EXPECTED
8451 070406 052702 000002      BIS     #BIT1,R2          ;SET BOT BIT IN EXPECTED
8452 070412 020102      CMP      R1,R2            ;DOES EXP = REC'D

```

```

8453 070414 001406          BEQ      140$          ;BR, IF EQUAL (OK)
8454 070416 005237 002212  INC      FATFLG      ;BUMP COUNT
8458 070422          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      070422 104456          TRAP      C$ERHRD
      070424 001331          .WORD   729
      070426 073315          .WORD   T26BOT
      070430 015564          .WORD   EXPREC
8459 070432          140$:  CKLOOP          ;LOOP IF SELECTED
      070432 104406          TRAP      C$CLP1
8460 070434 012703 072266  MOV      #T26RN,R3    ;COMMAND BUFFER ADDRESS
8461 070440 013737 003130 072252 150$:  MOV      NXMLO,T26RB ;STARTING READ BUFFER ADDRESS
8462 070446 013737 003132 072254  MOV      NXMHI,T26RB+2 ;SET UP HIGH ORDER ADDRESS BITS
8463
8464 ;*****
8465 ;
8466 ;REREAD DATA,IE,ACK, OPP COMMAND
8467 ;
8468 ;*****
8469
8470 070454 011337 072250  MOV      (R3),T26PK3  ;REREAD DATA,IE,ACK, OPP COMMAND
8471 070460 012704 072250 165$:  MOV      #T26PK3,R4  ;SET UP R4 WITH PACKET ADDRESS
8472 070464 012737 000400 072256  MOV      #256.,T26SZ ;SET UP RECORD SIZE IN PACKET
8473 070472 010465 000000  MOV      R4,TSDB(R5) ;ISSUE COMMAND
8474 070476 004737 016340  JSR      PC,WAITF    ;WAIT FOR SSR TO SET
8475 070502 016501 000002  MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
8476 070506 012702 104210  MOV      #SSR!NXM!SC!BIT3,R2 ;SET UP EXPECTED
8477 070512 020102  CMP      R1,R2       ;ARE THEY EQUAL
8478 070514 001422  BEQ      170$       ;BR, IF OK
8479 070516 031327 001000  BIT      (R3),#BIT9  ;CHECK FOR A READ COMMAND
8480 070522 001403  BEQ      168$       ;BR, IF IT WAS A READ COMMAND
8481 070524 030127 000002  BIT      R1,#BIT1    ;WAS BIT1 SET
8482 070530 001014  BNE      170$       ;BR, IF REREAD AND BIT1 SET
8483 070532
8484 070532 005237 003132 168$:  INC      NXMHI       ;BUMP TO NEXT ADDRESS RANGE
8485 070536 023727 003132 000004  CMP      NXMHI,#4    ;CHECK FOR OVERFLOW
8486 070544 001276  BNE      125$       ;BR, IF MORE BITS TO GO
8487 070546 005237 002212  INC      FATFLG     ;BUMP COUNT
8491 070552          ERRHRD  ERRNO,T26RRF,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
      070552 104456          TRAP      C$ERHRD
      070554 001332          .WORD   730
      070556 072525          .WORD   T26RRF
      070560 012136          .WORD   PKTSSR
8492 070562          170$:  CKLOOP          ;LOOP IF SELECTED
      070562 104406          TRAP      C$CLP1
8493
8494 ;*****
8495 ;
8496 ;READ DATA, ACK,CVC-1 COMMAND
8497 ;
8498 ;*****
8499
8500 070564 012737 140001 072250  MOV      #140001,T26PK3 ;READ DATA, ACK,CVC-1 COMMAND
8501 070572 012737 000400 072256  MOV      #256.,T26SZ  ;SET SIZE INTO PACKET
8502 070600 005037 072254  CLR      T26RB+2     ;CLEAR OUT HIGH ADDRESS BITS
8503 070604 013737 003114 072252  MOV      FREE,T26RB  ;GIVE READ A GOOD BUFFER
8504 070612 010465 000000  MOV      R4,TSDB(R5) ;ISSUE READ DATA COMMAND
8505 070616 004737 016340  JSR      PC,WAITF    ;WAIT FOR SSR

```

```

8506 070622 016501 000002      MOV      TSSR(R5),R1      ;PICK UP THE TSSR
8507 070626 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED
8508 070632 020102              CMP      R1,R2          ;IS THE TSSR OK
8509 070634 001406              BEQ      180$           ;BR, IF TSSR OK (GOOD)
8510 070636 005237 002212      INC      FATFLG         ;BUMP COUNT
8514 070642              ERRHRD  ERRNO, RDERR, PKTSSR ;READ DATA COMMAND FAILED
                                TRAP      C$ERHRD
                                .WORD     731
                                .WORD     RDERR
                                .WORD     PKTSSR
      070642 104456
      070644 001333
      070646 005204
      070650 012136
8515 070652              180$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
      070652 104406
8516 070654 017701 112234      MOV      @FREE,R1       ;FIRST WORD FROM READ BUFFER
8517 070660 012702 000001      MOV      #1,R2         ;SET UP EXPECTED
8518 070664 020102              CMP      R1,R2          ;IS TAPE POSITION CORRECT
8519 070666 001406              BEQ      190$           ;KEEP GOING POSITION OK
8520 070670 005237 002212      INC      FATFLG         ;BUMP COUNT
8524 070674              ERRHRD  ERRNO, T26WNG, EXPREC ;TAPE POSITION INCORRECT
                                TRAP      C$ERHRD
                                .WORD     732
                                .WORD     T26WNG
                                .WORD     EXPREC
      070674 104456
      070676 001334
      070700 072306
      070702 015564
8525 070704              190$:  CKLOOP
                                TRAP      C$CLP1
      070704 104406
8526
8527      ;*****
8528      ;
8529      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8530      ;
8531      ;*****
8532
8533 070706 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8534 070712 103411              BCS      194$           ;BR, IF NO PROBLEM
8535 070714 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR
8536 070720 010004              MOV      R0,R4          ;PACKET ADDRESS SET UP
8537 070722 005237 002212      INC      FATFLG         ;BUMP COUNT
8541 070726              ERRHRD  ERRNO, T26RWN, PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD     733
                                .WORD     T26RWN
                                .WORD     PKTSSR
      070726 104456
      070730 001335
      070732 073604
      070734 012136
8542 070736              194$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
      070736 104406
8543
8544      ;*****
8545      ;
8546      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8547      ;
8548      ;*****
8549
8550 070740 013701 072150      MOV      T26BFR+6,R1   ;PICK UP XSTO
8551 070744 010102              MOV      R1,R2         ;SET UP EXPECTED
8552 070746 052702 000002      BIS      @BIT1,R2      ;SET BOT BIT IN EXPECTED
8553 070752 020102              CMP      R1,R2          ;DOES EXP = REC'D
8554 070754 001406              BEQ      196$           ;BR, IF EQUAL (OK)
8555 070756 005237 002212      INC      FATFLG         ;BUMP COUNT
8559 070762              ERRHRD  ERRNO, T26BOT, EXPREC ;TAPE NOT AT BOT AFTER REWIND

```



```

8608 071056 004737 016064      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
8609 071062 103407              BCS    20$             ;BR IF INIT WAS OK
8610 071064 005237 002212      INC    FATFLG          ;BUMP COUNT
8614 071070 010001              MOV    R0,R1           ;CONTENTS OF TSSR REGISTER
8615 071072              ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      071072 104455              TRAP   C$ERDF         ;
      071074 001337              .WORD  735            ;
      071076 003650              .WORD  SFIERR         ;
      071100 012124              .WORD  SFIMSG        ;
8616 071102 013737 002172 072140 20$:  MOV    UNITN,T26DSW      ;SET UP UNIT NUMBER
8617                                ;
8618 071110 012704 072120      MOV    @T26PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
8619                                ;
8620                                ;*****
8621                                ;
8622                                ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8623                                ;
8624                                ;*****
8625                                ;
8626 071114 004737 010752      JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
8627 071120 103407              BCS    25$             ;BR, IF COMMAND ISSUED OK
8628 071122 005237 002212      INC    FATFLG          ;BUMP COUNT
8632 071126 010001              MOV    R0,R1           ;SAVE CONTENTS OF TSSR
8633 071130              ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      071130 104456              TRAP   C$ERHRD        ;
      071132 001340              .WORD  736            ;
      071134 005054              .WORD  WRTMSG         ;
      071136 012124              .WORD  SFIMSG        ;
8634 071140              25$:  CKLOOP              ;LOOP IF SELECTED
      071140 104406              TRAP   C$CLP1         ;
8635                                ;
8636                                ;*****
8637                                ;
8638                                ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8639                                ;
8640                                ;*****
8641                                ;
8642 071142 004737 011104      26$:  JSR    PC,REWIND    ;CALL TAPE REWIND COMMAND
8643 071146 016501 000002      MOV    TSSR(R5),R1      ;GET TSSR
8644 071152 012702 000200      MOV    @SSR,R2          ;SET UP EXPECTED TSSR
8645 071156 103407              BCS    30$             ;BR, IF NO PROBLEM
8646 071160 010004              MOV    R0,R4           ;PACKET ADDRESS SET UP
8647 071162 005237 002212      INC    FATFLG          ;BUMP COUNT
8651 071166              ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      071166 104456              TRAP   C$ERHRD        ;
      071170 001341              .WORD  737            ;
      071172 073604              .WORD  T26RWN         ;
      071174 012136              .WORD  PKTSSR        ;
8652 071176              30$:  CKLOOP              ;LOOP IF SELECTED
      071176 104406              TRAP   C$CLP1         ;
8653                                ;
8654                                ;*****
8655                                ;
8656                                ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8657                                ;
8658                                ;*****
8659                                ;

```

```

8660 071200 013701 072150      MOV      T26BFR+6,R1      ;PICK UP XSTO
8661 071204 010102              MOV      R1,R2           ;SET UP EXPECTED
8662 071206 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
8663 071212 020102              CMP      R1,R2           ;DOES EXP = REC'D
8664 071214 001406              BEQ      40$             ;BR, IF EQUAL (OK)
8665 071216 005237 002212      INC      FATFLG          ;BUMP COUNT
8669 071222              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      071222 104456              TRAP    C$ERHRD
      071224 001342              .WORD  738
      071226 073315              .WORD  T26BOT
      071230 015564              .WORD  EXPREC
8670 071232              40$:  CKLOOP              ;LOOP IF SELECTED
      071232 104406              TRAP    C$CLP1
8671 071234 012737 000400 072256      MOV      @256.,T26SZ     ;SET UP RECORD SIZE IN PACKET
8672 071242 013737 003114 072252      MOV      FREE,T26RB     ;ADDRESS OF READ BUFFER
8673 071250 005703              TST      R3              ;CHECK NUMBER OF TIMES THROUGH HERE
8674 071252 001404              BEQ      50$             ;BR, IF FIRST TIME THROUGH HERE
8675
8676      ;*****
8677      ;
8678      ;REREAD,CVC=1,ACK COMMAND
8679      ;
8680      ;*****
8681
8682 071254 012737 161001 072250      MOV      @161001,T26PK3 ;REREAD,CVC=1,ACK COMMAND
8683 071262 000403              BR       55$             ;SKIP NEXT COMMAND
8684
8685      ;*****
8686      ;
8687      ;REREAD,ACK COMMAND
8688      ;
8689      ;*****
8690
8691 071264 012737 141001 072250 50$:  MOV      @141001,T26PK3 ;REREAD,ACK COMMAND
8692 071272              55$:
8693 071272 012704 072250              MOV      @T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
8694 071276              65$:
8695 071276 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
8696 071302 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
8697 071306 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
8698 071312 012702 100206      MOV      @SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
8699 071316 020102              CMP      R1,R2          ;ARE THEY EQUAL
8700 071320 001406              BEQ      75$             ;BR, IF OK
8701 071322 005237 002212      INC      FATFLG          ;BUMP COUNT
8705 071326              ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      071326 104456              TRAP    C$ERHRD
      071330 001343              .WORD  739
      071332 073243              .WORD  T26WDE
      071334 012136              .WORD  PKTSSR
8706 071336              75$:  CKLOOP              ;LOOP IF SELECTED
      071336 104406              TRAP    C$CLP1
8707
8708      ;*****
8709      ;
8710      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8711      ;
8712      ;*****

```



```

      071454 104455                                TRAP    C$ERDF
      071456 001345                                .WORD  741
      071460 003650                                .WORD  SFIERR
      071462 012124                                .WORD  SFIMSG
8767 071464 013737 002172 072140 20$:  MOV    UNITN,T26DSW          ;SET UP UNIT NUMBER
8768
8769 071472 012704 072120                MOV    @T26PACKET,R4        ;SUBROUTINE NEEDS PACKET ADDRESS
8770
8771      ;*****
8772      ;
8773      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
8774      ;
8775      ;*****
8776
8777 071476 004737 010752                JSR    PC,WRTPHR           ;ISSUE WRITE CHARACTERISTICS
8778 071502 103407                        BCS    25$                ;BR, IF COMMAND ISSUED OK
8779 071504 005237 002212                INC    FATFLG             ;BUMP COUNT
8783 071510 010001                        MOV    R0,R1              ;SAVE CONTENTS OF TSSR
8784 071512                        ERRHRD  ERRNO,WRTPHR,SFIMSG ;WRITE CHARACTERISTICS FAILED
      071512 104456                                TRAP    C$ERHRD
      071514 001346                                .WORD  742
      071516 005054                                .WORD  WRTPHR
      071520 012124                                .WORD  SFIMSG
8785 071522                        25$:  CKLOOP                ;LOOP IF SELECTED
      071522 104406                                TRAP    C$CLP1
8786
8787      ;*****
8788      ;
8789      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8790      ;
8791      ;*****
8792
8793 071524 004737 011104                26$:  JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
8794 071530 016501 000002                MOV    TSSR(R5),R1        ;GET TSSR
8795 071534 012702 000200                MOV    @SSR,R2           ;SET UP EXPECTED TSSR
8796 071540 103407                        BCS    30$                ;BR, IF NO PROBLEM
8797 071542 010004                        MOV    R0,R4             ;PACKET ADDRESS SET UP
8798 071544 005237 002212                INC    FATFLG             ;BUMP COUNT
8802 071550                        ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      071550 104456                                TRAP    C$ERHRD
      071552 001347                                .WORD  743
      071554 073604                                .WORD  T26RWN
      071556 012136                                .WORD  PKTSSR
8803 071560                        30$:  CKLOOP                ;LOOP IF SELECTED
      071560 104406                                TRAP    C$CLP1
8804
8805      ;*****
8806      ;
8807      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
8808      ;
8809      ;*****
8810
8811 071562 013701 072150                MOV    T26BFR+6,R1        ;PICK UP XSTO
8812 071566 010102                        MOV    R1,R2             ;SET UP EXPECTED
8813 071570 052702 000002                BIS    @BIT1,R2          ;SET BOT BIT IN EXPECTED
8814 071574 020102                        CMP    R1,R2             ;DOES EXP = REC'D
8815 071576 001406                        BEQ    40$                ;BR, IF EQUAL (OK)

```



```

8816 071600 005237 002212          INC    FATFLG          ;BUMP COUNT
8820 071604          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      071604 104456          TRAP    C$ERHRD
      071606 001350          .WORD  744
      071610 073315          .WORD  T26BOT
      071612 015564          .WORD  EXPREC
8821 071614          40$:   CKLOOP          TRAP    C$CLP1
      071614 104406
8822
8823          ;*****
8824          ;
8825          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
8826          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
8827          ;
8828          ;*****
8829
8830 071616 012703 000001          MOV    #000001,R3          ;SET UP SPACE FORWARD 1 RECORD
8831 071622 004737 010556          JSR    PC,SPACE          ;ISSUE SPACE COMMAND
8832 071626 103411          BCS    75$              ;BR, IF OK
8833 071630 016501 000002          MOV    TSSR(R5),R1       ;GET STATUS DATA
8834 071634 010004          MOV    R0,R4            ;GET PACKET ADDRESS
8835 071636 005237 002212          INC    FATFLG          ;BUMP COUNT
8839 071642          ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER HEAD DATA
      071642 104456          TRAP    C$ERHRD
      071644 001351          .WORD  745
      071646 073243          .WORD  T26WDE
      071650 012136          .WORD  PKTSSR
8840 071652          75$:   CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
      071652 104406
8841
8842          ;*****
8843          ;
8844          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
8845          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
8846          ;
8847          ;*****
8848
8849 071654 012703 100001          MOV    #100001,R3       ;SET SPACE REVERSE 1 RECORD
8850 071660 004737 010556          JSR    PC,SPACE          ;ISSUE COMMAND
8851 071664 103411          BCS    175$             ;GO ON IF ALL IS WELL
8852 071666 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
8853 071672 010004          MOV    R0,R4            ;SET UP EXPECTED (PACKET CONTENTS)
8854 071674 005237 002212          INC    FATFLG          ;BUMP COUNT
8858 071700          ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      071700 104456          TRAP    C$ERHRD
      071702 001352          .WORD  746
      071704 073243          .WORD  T26WDE
      071706 012136          .WORD  PKTSSR
8859 071710          175$:  CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
      071710 104406
8860 071712 013737 003114 072252          MOV    FREE,T26RB       ;ADDRESS OF BUFFER
8861 071720 005737 072300          TST    T26CNU          ;CHECK FOR TIMES THROUGH HERE
8862 071724 001404          BEQ    176$             ;BR, IF FIRST TIME THROUGH
8863
8864          ;*****
8865          ;
8866          ;REREAD (PREVIOUS),IE,ACK,OPP=1 CMD.

```



TSV7 - HARDWARE TESTS 1-8  
TEST 6: REREADS

MACRO M1113 14-JUN-84 14:17

SEQ 0273

```

8918 072104
8919 072104
      072104 104432
      072106 003116
8920
8921
8922
8923
8925 072110
8927 072120
8928 072120 014004
8929 072122 072130
8930 072124 000000
8931 072126 000012
8932 072130
8933 072130 072142
8934 072132 000000
8935 072134 000024
8936 072136 000000
8937 072140 000000
8938 072142
8939
8940
8941
8943 072224
8945 072230
8946 072230 100006
8947 072232 072260
8948 072234 000000
8949 072236 000006
8950
8952 072240
8954 072250
8955 072250 140005
8956 072252
8957 072252 003114
8958 072254 000000
8959 072256 000000
8960
8961
8962
8963
8964 072260
8965 072260 010
8966 072261 200
8967 072262 000000
8968 072264 000000
8969
8970
8971
8972
8973
8974 072266 140001
8975 072270 141401
8976 072272 161401
8977 072274 177777
8978
1634: EXIT TST ;ALL DONE THIS TEST
                                TRAP C$EXIT
                                .WORD L10102-.

;*
;LOCAL STORAGE FOR THIS TEST
;
;
      .BLKB 10-<.-TSV2&7>
T26PACKET: ;COMMAND PACKET FOR TEST
      .WORD 14004 ;WRITE CHARACTERISTICS COMMAND, WITH CVC=1, ACK
      .WORD T26DATA ;ADDRESS OF CHARACTERISTICS BLOCK
      .WORD 0
      .WORD 10. ;STARTING VALUE OF BLOCK SIZE
T26DATA: ;CHARACTERISTICS DATA BLOCK
      .WORD T26BFR ;ADDRESS OF MESSAGE BUFFER
      .WORD 0
      .WORD 20. ;LENGTH OF MESSAGE BUFFER
      .WORD 0
T26DSW: .WORD 0 ;SELECT DRIVE 0
T26BFR: .BLKW 25. ;MESSAGE BUFFER

;WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
      .BLKB 10-<.-TSV2&7>
T26PK2: ;WRITE SUB SYS MEM COMMAND, AND ACK
      .WORD 100006 ;ADDRESS OF SELECT BLOCK DATA
      .WORD T26BF2
      .WORD 0
      .WORD 6. ;SIZE OF DATA PACKET

      .BLKB 10-<.-TSV2&7>
T26PK3: ;REREAD COMMAND, CVC=1 AND ACK
      .WORD 140005
T26R8: ;ADDRESS OF WRITE BUFFER
      .WORD FREE
T26WB: .WORD 0
      .WORD 0 ;SIZE OF BUFFER (EXTENT)
T26SZ: .WORD 0
      .EVEN

;
;
;
T26BF2:
T26BS0: .BYTE 10 ;BSELO AREA
T26BS1: .BYTE 200 ;BSEL1 AREA
T26S2: .WORD 0 ;SEL 2 AREA
T26S3: .WORD 0 ;DATA AREA

;
;
      .EVEN
;TAPE MOTION PACKET COMMAND VALUES
T26RN: .WORD 140001 ;READ DATA
      .WORD 141401 ;REREAD NEXT OPP=0
      .WORD 161401 ;REREAD NEXT OPP=1
      .WORD 177777 ;END OF DATA

```

```

8979
8980 072276 000000
8981 072300 000000
8982
8983 072302 000000
8984
8985 072304 000000
8986
8987
8988
8989
8990
8991
8992
8993 072306 124 141 160 T26WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
8994 072374 122 105 122 T26NEF: .ASCIZ 'REREAD PREVIOUS, At BOT, Failed To Set NEF (XST0)'
8995 072456 124 123 123 T26RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
8996 072525 122 105 122 T26RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
8997 072622 122 105 122 T26RRG: .ASCIZ 'REREAD Previous (Read Reverse, Space Forward) Command Failed'
8998 072717 120 117 123 T26SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
8999 073001 122 111 102 T26LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
9000 073051 124 123 123 T26MDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
9001 073126 111 154 154 T26LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
9002 073207 122 105 122 T26SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
9003 073243 124 123 123 T26MDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
9004 073315 124 141 160 T26BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
9005 073362 104 141 164 T26DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
9006 073450 122 105 122 T26EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
9007 073527 124 123 123 T26TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
9008 073604 122 145 167 T26RMN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
9009 073753 122 101 115 T26RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
9010 073765 124 123 123 T26AM3: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
9011 073775 104 162 151 T26OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
9012 074050 124 123 123 T26MDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SMB Bit Set'
9013 074140 124 123 123 T26MDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
9014 074213 103 126 103 T26VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
9015 074266 124 123 102 T26BA: .ASCIZ 'TSBA Not Correct After REREAD DATA Command'
9016 074341 127 122 111 T26WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
9017 074430 122 145 141 T26LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XST0'
9018 074512 122 145 141 T26LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XST0'
9019 074574 122 145 163 T26PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
9020 074662 122 145 141 T26TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
9021 074750 104 141 164 T26NEQ: .ASCIZ 'Data REREAD From Tape Not Correct, After SMB=1'
9022 075027 122 145 162 TST26ID: .ASCIZ 'Rereads'
9023
9024
9025
9026
9027
9028
9029
9030
9031 075040
9032 075040
9033 075044 012701 072120
9034 075050 012721 140004
9035 075054 012721 072130
;
T26CNT: .WORD 0 ;TAPE RECORD COUNTER STORAGE AREA
T26CNU: .WORD 0 ;TAPE RECORD COUNTER STORAGE AREA
T26RSZ: .WORD 0 ;RECORD STORAGE SIZE AREA
T26DLY: .WORD 0 ;DELAY COUNTER AREA
;*
;LOCAL TEXT MESSAGES FOR TEST
;
T26WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
T26NEF: .ASCIZ 'REREAD PREVIOUS, At BOT, Failed To Set NEF (XST0)'
T26RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
T26RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
T26RRG: .ASCIZ 'REREAD Previous (Read Reverse, Space Forward) Command Failed'
T26SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
T26LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
T26MDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
T26LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
T26SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
T26MDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
T26BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
T26DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
T26EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
T26TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
T26RMN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
T26RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
T26AM3: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
T26OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
T26MDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SMB Bit Set'
T26MDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
T26VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
T26BA: .ASCIZ 'TSBA Not Correct After REREAD DATA Command'
T26WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
T26LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XST0'
T26LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XST0'
T26PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
T26TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
T26NEQ: .ASCIZ 'Data REREAD From Tape Not Correct, After SMB=1'
TST26ID: .ASCIZ 'Rereads'
.EVEN
;*
;
;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
;WRITE SUBSYSTEM MEMORY COMMAND
;
;
T26REST:
SAVREG ;SAVE THE REGISTERS
MOV #T26PACKET,R1 ;START OF THE PACKET
MOV #140004,(R1) ;WRITE SUBSYSTEM MEM. WITH ACK, CVC=1
MOV #T26DATA,(R1) ;ADDRESS OF CHARAISTICS DATA BLOCK

```

```

9036 075060 005021          CLR      (R1).          ;EXTENDED ADDRESS
9037 075062 012721 000012  MOV      #10.,(R1).    ;SIZE OF DATA BLOCK IN BYTES
9038 075066 012721 072142  MOV      #T268FR,(R1). ;ADDRESS OF MESSAGE BUFFER
9039 075072 005021          CLR      (R1).
9040 075074 012721 000024  MOV      #20.,(R1).    ;LENGTH OF MESSAGE BUFFER
9041 075100 005021          CLR      (R1).
9042 075102 012711 000000  MOV      #0,(R1)       ;SELECT DRIVE ZERO (0)
9043 075106 012702 000030  MOV      #24.,R2       ;NUMBER OF LOCATIONS TO BE CLEARED
9044 075112 012762 177777 072142 64: MOV      #177777,T268FR(R2) ;ALL ONES TO MESSAGE BUFFER
9045 075120 005742          TST      -(R2)         ;NEXT LOCATION
9046 075122 020227 000000  CMP      R2,#0         ;CHECK FOR END OF LOOP
9047 075126 001371          BNE      64:          ;KEEP GOING UNTIL DONE
9048 075130 000207          RTS      PC           ;RETURN
9049
9050
9051 075132          T26RT2:
9052 075132          SAVREG          ;SAVE THE REGISTERS
9053 075136 012701 072230  MOV      #T26PK2,R1    ;START OF THE PACKET
9054 075142 012721 140006  MOV      #140006,(R1). ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1.
9055 075146 012721 072260  MOV      #T268F2,(R1). ;ADDRESS OF DATA BLOCK
9056 075152 005021          CLR      (R1).        ;EXTENDED ADDRESS
9057 075154 012721 000006  MOV      #6.,(R1).    ;SIZE OF DATA BLOCK IN BYTES
9058 075160 005021          CLR      (R1).
9059 075162 012701 072260  MOV      #T268F2,R1    ;POINT TO DATA SEL AREA
9060 075166 005021          CLR      (R1).
9061 075170 005011          CLR      (R1)
9062 075172 000207          RTS      PC           ;RETURN
9063 075174          T26RT3:
9064 075174          SAVREG          ;SAVE THE REGISTERS
9065 075200 012701 072250  MOV      #T26PK3,R1    ;START OF THE PACKET
9066 075204 012721 000000  MOV      #0,(R1).     ;WRITE SUBSYSTEM MEM. WITH ACK.
9067 075210 012721 000000  MOV      #0,(R1).     ;ADDRESS OF DATA BLOCK
9068 075214 005021          CLR      (R1).        ;EXTENDED ADDRESS
9069 075216 012711 000000  MOV      #0,(R1)     ;SIZE OF DATA BLOCK IN BYTES
9070 075222 000207          RTS      PC           ;RETURN
9071 075224          ENDTST
          L10102: TRAP C#ETST
          075224 104401
9072
9073          .SBTTL TEST 7: WRITE DATA RETRY
9074          ;*
9075          ;
9076          ;THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY
9077          ;COMMAND (SPACE REVERSE, ERASE, WRITE DATA)
9078          ;
9079          ;
9080          ;THE TEST CONSISTS OF THE FOLLOWING 5 SUBTESTS
9081          ;
9082          ;
9083          ;
9084          ;-
9085 075226          BGNTST
          075226
9086 075226 012737 006446 002170 MOV      #EPRT2,EPRTSW ;SECONDARY ERROR MESSAGE
9087 075234 005037 003124          CLR      KTNABLE     ;TURN OFF KT11
9088 075240 004737 017364          JSR      PC,KTOFF     ;TURN KT11 BACK OFF IF THERE
9093 075244 012700 105063          MOV      #TST27ID,RO  ;ASCII MESSAGE TO IDENTIFY TEST

```



```

9140 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
9141 ;
9142 ;*****
9143
9144 075412 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
9145 075416 103407 BCS 25$ ;BR, IF COMMAND ISSUED OK
9146 075420 005237 002212 INC FATFLG ;BUMP COUNT
9150 075424 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
9151 075426 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
          075426 104456 TRAP C$ERHRD
          075430 001276 .WORD 702
          075432 005054 .WORD WRTMSG
          075434 012124 .WORD SFIMSG
9152 075436 25$: CKLOOP ;LOOP IF SELECTED
          075436 104406 TRAP C$CLP1
9153 ;*****
9154 ;
9155 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9156 ;
9157 ;*****
9158 ;
9159
9160 075440 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9161 075444 103407 BCS 30$ ;BR, IF NO PROBLEM
9162 075446 010004 MOV RO,R4 ;SET UP REWIND PACKET ADDRESS
9163 075450 005237 002212 INC FATFLG ;BUMP COUNT
9167 075454 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
          075454 104456 TRAP C$ERHRD
          075456 001277 .WORD 703
          075460 103305 .WORD T27RWN
          075462 012136 .WORD PKTSSR
9168 075464 30$: CKLOOP ;LOOP IF SELECTED
          075464 104406 TRAP C$CLP1
9169 ;*****
9170 ;
9171 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9172 ;
9173 ;*****
9174 ;
9175
9176 075466 013701 102000 MOV T27BFR+6,R1 ;PICK UP XSTO
9177 075472 010102 MOV R1,R2 ;SET UP EXPECTED
9178 075474 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
9179 075500 020102 CMP R1,R2 ;DOES EXP = REC'D
9180 075502 001406 BEQ 40$ ;BR, IF EQUAL (OK)
9181 075504 005237 002212 INC FATFLG ;BUMP COUNT
9185 075510 ERRHRD ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          075510 104456 TRAP C$ERHRD
          075512 001300 .WORD 704
          075514 103001 .WORD T27BOT
          075516 015564 .WORD EXPREC
9186 075520 40$: CKLOOP ;LOOP IF SELECTED
          075520 104406 TRAP C$CLP1
9187 075522 012737 000400 102106 MOV #256.,T27SZ ;SET UP RECORD SIZE
9188 075530 013737 003114 102102 MOV FREE,T27WB ;ADDRESS OF WRITE BUFFER
9189 ;*****
9190 ;

```

```

9191
9192 ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
9193 ;
9194 ;*****
9195
9196 075536 012737 141005 102100      MOV      #141005,T27PK3      ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
9197 075544 012704 102100      MOV      #T27PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
9198 075550 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
9199 075554 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
9200 075560 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9201 075564 012702 100206      MOV      #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
9202 075570 020102      CMP      R1,R2            ;ARE THEY EQUAL
9203 075572 001406      BEQ      75$             ;BR, IF OK
9204 075574 005237 002212      INC      FATFLG           ;BUMP COUNT
9208 075600      ERRHRD  ERRNO,T27WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          075600 104456      TRAP    C$ERHRD
          075602 001301      .WORD  705
          075604 102712      .WORD  T27WDE
          075606 012136      .WORD  PKTSSR
9209 075610      75$:   CKLOOP           ;LOOP IF SELECTED
          075610 104406      TRAP    C$CLP1
9210
9211 ;*****
9212 ;
9213 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9214 ;
9215 ;*****
9216
9217 075612 013701 102000      MOV      T27BFR+6,R1      ;GET XSTO STATUS WORD
9218 075616 010102      MOV      R1,R2            ;SET UP EXPECTED
9219 075620 052702 002000      BIS      #BIT10,R2       ;SET THE NEF BIT
9220 075624 020102      CMP      R1,R2            ;ARE THEY EQUAL
9221 075626 001406      BEQ      170$            ;BR, IF EQUAL (GOOD)
9222 075630 005237 002212      INC      FATFLG           ;BUMP COUNT
9226 075634      ERRHRD  ERRNO,T27NEF,EXPREC ;NEF SHOULD BE SET
          075634 104456      TRAP    C$ERHRD
          075636 001302      .WORD  706
          075640 104451      .WORD  T27NEF
          075642 015564      .WORD  EXPREC
9227 075644      170$:  CKLOOP
          075644 104406      TRAP    C$CLP1
9228 075646      ENDSUB
          075646      L10123:  TRAP    C$ESUB
          075646 104403
9229 075650 023727 002212 000017      CMP      FATFLG,#15.      ;IS ERROR COUNT AT 25
9230 075656 103402      BLO     999$             ;BR, IF LESS THAN 25
9231 075660 004737 017272      JSR      PC,CKDROP        ;TRY TO DROP THE UNIT
9232 075664      999$:
9233
9234 ;+
9235 ;
9236 ;TEST 7, SUBTEST 2
9237 ;
9238 ;VERIFIES THAT WRITE DATA RETRY COMMAND ISSUED WHILE
9239 ;THE TAPE IS POSITIONED BEFORE THE FIRST RECORD ON
9240 ;TAPE (BUT NOT AT BOT) RESULTS IN TAPE STATUS ALERT
9241 ;TERMINATION, WITH THE REVERSE INTO BOT (RIB) STATUS

```





```

9294 075776 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9295 076002 005237 002212      INC      FATFLG          ;BUMP COUNT
9299 076006      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    709
                                .WORD    T27RWN
                                .WORD    PKTSSR
9300 076016      26$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
9301 076020 012703 000400      MOV      #256.,R3      ;STARTING RECORD SIZE
9302 076024 013737 003114 102102  MOV      FREE,T27WB     ;STARTING WRITE BUFFER ADDRESS
9303
9304 ;*****
9305 ;
9306 ;WRITE DATA,CVC=1,ACK COMMAND
9307 ;
9308 ;*****
9309
9310 076032 012737 140005 102100  MOV      #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
9311 076040 012704 102100      MOV      #T27PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
9312 076044 010337 102106      MOV      R3,T27SZ     ;SET UP RECORD SIZE IN PACKET
9313 076050 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
9314 076054 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
9315 076060 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
9316 076064 012702 000200      MOV      #SSR,R2     ;SET UP EXPECTED
9317 076070 020102      CMP      R1,R2       ;ARE THEY EQUAL
9318 076072 001406      BEQ      28$         ;BR, IF OK
9319 076074 005237 002212      INC      FATFLG      ;BUMP COUNT
9323 076100      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    710
                                .WORD    WRERR
                                .WORD    PKTSSR
9324 076110      28$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
9325
9326 ;*****
9327 ;
9328 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9329 ;
9330 ;*****
9331
9332 076112 004737 011104      JSR      PC,REWIND    ;CALL TAPE REWIND COMMAND
9333 076116 103411      BCS     30$         ;BR, IF NO PROBLEM
9334 076120 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
9335 076124 010004      MOV      R0,R4       ;SET UP REWIND PACKET ADDRESS
9336 076126 005237 002212      INC      FATFLG      ;BUMP COUNT
9340 076132      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    711
                                .WORD    T27RWN
                                .WORD    PKTSSR
9341 076142      30$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
9342
9343 ;*****
9344 ;

```

```

9345 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9346 ;
9347 ;*****
9348
9349 076144 013701 102000      MOV      T27BFR+6,R1      ;PICK UP XSTO
9350 076150 010102            MOV      R1,R2           ;SET UP EXPECTED
9351 076152 052702 000002     BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
9352 076156 020102            CMP      R1,R2           ;DOES EXP = REC'D
9353 076160 001406            BEQ      40$             ;BR, IF EQUAL (OK)
9354 076162 005237 002212     INC      FATFLG          ;BUMP COUNT
9358 076166            ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    712
                                .WORD    T27BOT
                                .WORD    EXPREC
                                076166 104456
                                076170 001310
                                076172 103001
                                076174 015564
9359 076176            40$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                076176 104406
9360 ;*****
9361 ;
9362 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9363 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9364 ;
9365 ;*****
9366
9367
9368 076200 012703 000001     MOV      @1,R3           ;PARAMETER SPACE FORWARD 1 RECORD
9369 076204 004737 010556     JSR      PC,SPACE        ;CALL SPACE RECORDS ROUTINE
9370 076210 103413            BCS      50$             ;BR, IF NO ERRORS
9371 076212 016501 000002     MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
9372 076216 012702 000200     MOV      @SSR,R2        ;SET UP EXPECTED
9373 076222 010004            MOV      R0,R4           ;SET UP REWIND PACKET ADDRESS
9374 076224 005237 002212     INC      FATFLG          ;BUMP COUNT
9378 076230            ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE RECORDS COMMAND FAILED
                                TRAP      C$ERHRD
                                .WORD    713
                                .WORD    T27SCF
                                .WORD    PKTSSR
                                076230 104456
                                076232 001311
                                076234 104547
                                076236 012136
9379 076240            50$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                076240 104406
9380 ;*****
9381 ;
9382 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9383 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9384 ;
9385 ;*****
9386
9387
9388 076242 012703 100001     MOV      @100001,R3      ;PARAMETER SPACE REVERSE 1 RECORD
9389 076246 004737 010556     JSR      PC,SPACE        ;CALL SPACE RECORDS ROUTINE
9390 076252 103413            BCS      60$             ;BR, IF NO ERRORS
9391 076254 016501 000002     MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
9392 076260 012702 000200     MOV      @SSR,R2        ;SET UP EXPECTED
9393 076264 010004            MOV      R0,R4           ;SET UP REWIND PACKET ADDRESS
9394 076266 005237 002212     INC      FATFLG          ;BUMP COUNT
9398 076272            ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE RECORDS COMMAND FAILED
                                TRAP      C$ERHRD
                                .WORD    714
                                076272 104456
                                076274 001312

```

```

    076276 104547                .WORD T27SCF
    076300 012136                .WORD PKTSSR
9399 076302 014406 003114 102102 60$: CKLOOP                ;LOOP IF SELECTED
    076302 104406                TRAP C$CLP1
9400 076304 013737 003114 102102      MOV     FREE,T27RB      ;ADDRESS OF BUFFER
9401
9402   ;*****
9403   ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
9404   ;
9405   ;*****
9406
9407
9408 076312 012737 141005 102100      MOV     #141005,T27PK3 ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
9409 076320 012737 000400 102106      MOV     #256.,T27SZ    ;SET UP THE SIZE OF RECORD
9410 076326 012704 102100                MOV     #T27PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
9411 076332 010465 000000                MOV     R4,TSDB(R5)   ;ISSUE COMMAND
9412 076336 004737 016340                JSR     PC,WAITF      ;WAIT FOR SSR TO SET
9413 076342 016501 000002                MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
9414 076346 012702 100204                MOV     #SSR!SC!BIT2,R2 ;SET UP EXPECTED TAPE STATUS ALERT
9415 076352 020102                CMP     R1,R2         ;ARE THEY EQUAL
9416 076354 001406                BEQ     180$         ;BR, IF OK
9417 076356 005237 002212                INC     FATFLG        ;BUMP COUNT
9421 076362                ERRHRD  ERRNO,T27TSA,PKTSSR ;TSSR INCORRECT AFTER READ DATA
    076362 104456                TRAP   C$ERHRD
    076364 001313                .WORD  715
    076366 104624                .WORD  T27TSA
    076370 012136                .WORD  PKTSSR
9422 076372 014406 003114 102102 180$: CKLOOP                ;LOOP IF SELECTED
    076372 104406                TRAP   C$CLP1
9423 076374 013701 102006      MOV     T27BFR+14,R1  ;GET XST3 STATUS WORD
9424 076400 010102                MOV     R1,R2        ;SET UP EXPECTED
9425 076402 052702 000001                BIS     #BIT0,R2     ;SET THE RIB BIT
9426 076406 020102                CMP     R1,R2        ;ARE THEY EQUAL
9427 076410 001406                BEQ     190$         ;BR, IF EQUAL (GOOD)
9428 076412 005237 002212                INC     FATFLG        ;BUMP COUNT
9432 076416                ERRHRD  ERRNO,T27NEF,EXPREC ;NEF SHOULD BE SET
    076416 104456                TRAP   C$ERHRD
    076420 001314                .WORD  716
    076422 104451                .WORD  T27NEF
    076424 015564                .WORD  EXPREC
9433 076426 014406 003114 102102 190$: CKLOOP                ;LOOP IF SELECTED
    076426 104406                TRAP   C$CLP1
9434 076430                ENDSUB                ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
    076430                L10124:                TRAP   C$ESUB
9435 076432 023727 002212 000017      CMP     FATFLG,#15.  ;IS ERROR COUNT AT 25
9436 076440 103402                BLO     999$         ;BR, IF LESS THAN 25
9437 076442 004737 017272                JSR     PC,CKDROP    ;TRY TO DROP THE UNIT
9438 076446                999$:
9439
9440
9441   ;*
9442   ;TEST 7, SUBTEST 3
9443   ;
9444   ;VERIFIES THAT A WRITE DATA RETRY COMMAND WITH SWB=0
9445   ;TERMINATES PROPERLY AND WRITES CORRECT DATA ON TAPE
9446   ;(THE WRITTEN RECORD IS READ AND CHECKED). VARIOUS

```



```

9499 076560 005237 002212          INC    FATFLG          ;BUMP COUNT
9503 076564          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      076564 104456          TRAP    C$ERHRD
      076566 001317          .WORD  719
      076570 103305          .WORD  T27RWN
      076572 012136          .WORD  PKTSSR
9504 076574          30$:   CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
      076574 104406
9505
9506          ;*****
9507          ;
9508          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9509          ;
9510          ;*****
9511
9512 076576 013701 102000          MOV    T27BFR+6,R1      ;PICK UP XSTO
9513 076602 010102          MOV    R1,R2           ;SET UP EXPECTED
9514 076604 052702 000002          BIS    @BIT1,R2        ;SET BOT BIT IN EXPECTED
9515 076610 020102          CMP    R1,R2           ;DOES EXP = REC'D
9516 076612 001406          BEQ    40$             ;BR, IF EQUAL (OK)
9517 076614 005237 002212          INC    FATFLG          ;BUMP COUNT
9521 076620          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      076620 104456          TRAP    C$ERHRD
      076622 001320          .WORD  720
      076624 103001          .WORD  T27BOT
      076626 015564          .WORD  EXPREC
9522 076630          40$:   CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
      076630 104406
9523 076632 012703 000024          MOV    @20.,R3         ;STARTING RECORD SIZE
9524 076636 013737 003114 102102      MOV    FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
9525
9526          ;*****
9527          ;
9528          ;WRITE DATA,CVC=1,ACK COMMAND
9529          ;
9530          ;*****
9531
9532 076644 012737 140005 102100 65$:  MOV    @140005,T27PK3   ;WRITE DATA,CVC=1,ACK COMMAND
9533 076652 012704 102100          MOV    @T27PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
9534 076656 010300          MOV    R3,R0          ;SET PATTERN IN CORRECT REGISTER
9535 076660 004737 017512          JSR    PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
9536 076664 010337 102106          MOV    R3,T27SZ       ;SET UP RECORD SIZE IN PACKET
9537 076670 010465 000000          MOV    R4,TSDB(R5)    ;ISSUE COMMAND
9538 076674 004737 016340          JSR    PC,WAITF       ;WAIT FOR SSR TO SET
9539 076700 016501 000002          MOV    TSSR(R5),R1    ;GET TSSR CONTENTS
9540 076704 012702 000200          MOV    @SSR,R2        ;SET UP EXPECTED
9541 076710 020102          CMP    R1,R2          ;ARE THEY EQUAL
9542 076712 001406          BEQ    80$             ;BR, IF OK
9543 076714 005237 002212          INC    FATFLG          ;BUMP COUNT
9547 076720          ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      076720 104456          TRAP    C$ERHRD
      076722 001321          .WORD  721
      076724 005111          .WORD  WRTErr
      076726 012136          .WORD  PKTSSR
9548 076730          80$:   CKLOOP          ;LOOP IF SELECTED          TRAP    C$CLP1
      076730 104406
9549

```

```

9550 ;*****
9551 ;
9552 ;WRITE DATA RETRY,CVC=1,ACK COMMAND
9553 ;
9554 ;*****
9555
9556 076732 012737 141005 102100      MOV      @141005,T27PK3      ;WRITE DATA RETRY,CVC=1,ACK COMMAND
9557 076740 010465 000000              MOV      R4,TSDB(R5)      ;ISSUE COMMAND
9558 076744 004737 016340              JSR      PC,WAITF        ;WAIT FOR SSR TO SET
9559 076750 016501 000002              MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
9560 076754 012702 000200              MOV      @SSR,R2        ;SET UP EXPECTED
9561 076760 020102                      CMP      R1,R2          ;ARE THEY EQUAL
9562 076762 001406                      BEQ      90$            ;BR, IF OK
9563 076764 005237 002212              INC      FATFLG          ;BUMP COUNT
9567 076770                      ERRHRD  ERRNO,T27WRF,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA RETRY
                                TRAP      C$ERHRD
                                .WORD    722
                                .WORD    T27WRF
                                .WORD    PKTSSR
                                9568 077000      90$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                9569 077002      TST      (R3)+          ;BUMP RECORD SIZE COUNTER
                                9570 077004      CMP      R3,@40.       ;AT 40 SIZE YET
                                9571 077010      BNE      65$            ;BR, IF MORE RECORDS TO WRITE
9572 ;*****
9573 ;
9574 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9575 ;
9576 ;*****
9577 ;
9578
9579 077012 004737 011104              JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
9580 077016 103407                      BCS      230$          ;BR, IF NO PROBLEM
9581 077020 010001                      MOV      R0,R1          ;SAVE TSSR
9582 077022 005237 002212              INC      FATFLG          ;BUMP COUNT
9586 077026                      ERRHRD  ERRNO,T27RWN,EXPREC ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    723
                                .WORD    T27RWN
                                .WORD    EXPREC
                                9587 077036      230$: CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
9588 ;*****
9589 ;
9590 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9591 ;
9592 ;*****
9593 ;
9594
9595 077040 013701 102000              MOV      T27BFR+6,R1    ;PICK UP XSTO
9596 077044 010102                      MOV      R1,R2          ;SET UP EXPECTED
9597 077046 052702 000002              BIS      @BIT1,R2       ;SET BOT BIT IN EXPECTED
9598 077052 020102                      CMP      R1,R2          ;DOES EXP = REC'D
9599 077054 001406                      BEQ      240$          ;BR, IF EQUAL (OK)
9600 077056 005237 002212              INC      FATFLG          ;BUMP COUNT
9604 077062                      ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                077062 104456

```

```

077064 001324                                .WORD 724
077066 103001                                .WORD T27BOT
077070 015564                                .WORD EXPREC
9605 077072 2401: CKLOOP                      ;LOOP IF SELECTED                                TRAP C1CLP1
077072 104406                                ;STARTING RECORD SIZE
9606 077074 012703 000024                    MOV @20.,R3                                       ;STARTING READ BUFFER ADDRESS
9607 077170 013737 003114 102102            MOV FREE,T27RB
9608
9609 ;*****
9610 ;
9611 ;READ DATA,ACK COMMAND
9612 ;
9613 ;*****
9614
9615 077106 012737 100001 102100 2651: MOV @100001,T27PK3 ;READ DATA,ACK COMMAND
9616 077114 012704 102100                    MOV @T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
9617 077120 010337 102106                    MOV R3,T27SZ ;SET UP RECORD SIZE IN PACKET
9618 077124 010465 000000                    MOV R4,TSDB(R5) ;ISSUE COMMAND
9619 077130 004737 016340                    JSR PC,WAITF ;WAIT FOR SSR TO SET
9620 077134 016501 000002                    MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9621 077140 012702 000200                    MOV @SSR,R2 ;SET UP EXPECTED
9622 077144 020102                            CMP R1,R2 ;ARE THEY EQUAL
9623 077146 001406                            BEQ 2801 ;BR, IF OK
9624 077150 005237 002212                    INC FATFLG ;BUMP COUNT
9628 077154 ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
077154 104456                                TRAP C1ERHRD
077156 001325                                .WORD 725
077160 005204                                .WORD RDERR
077162 012136                                .WORD PKTSSR
9629 077164 2801: CKLOOP                      ;LOOP IF SELECTED                                TRAP C1CLP1
077164 104406                                ;GET BUFFER ADDRESS
9630 077166 013702 003114                    MOV FREE,R2 ;GET RECORD SIZE
9631 077172 010304                            MOV R3,R4 ;POINT BACK TO 1ST RECORD
9632 077174 162704 000024                    SUB @20.,R4 ;POINT TO 1ST LOC IN BUFFER
9633 077200 060204 2851: ADD R2,R4 ;DATA WRITTEN = READ
9634 077202 021403                            CMP (R4),R3 ;BR, IF DATA OK (GOOD)
9635 077204 001410                            BEQ 2901 ;PICK UP BAD DATA
9636 077206 011401                            MOV (R4),R1 ;SET UP EXPECTED
9637 077210 010302                            MOV R3,R2 ;BUMP COUNT
9638 077212 005237 002212                    INC FATFLG ;DATA IN BUFFER NOT CORRECT
9642 077216 ERRHRD ERRNO,T27DTA,EXPREC ;DATA IN BUFFER NOT CORRECT
077216 104456                                TRAP C1ERHRD
077220 001326                                .WORD 726
077222 104766                                .WORD T27DTA
077224 015564                                .WORD EXPREC
9643 077226 2901: CKLOOP                      ;LOOP IF SELECTED                                TRAP C1CLP1
077226 104406                                ;BUMP TO NEXT ADDRESS
9644 077230 005724                            TST (R4). ;BACK TO RECORD SIZE
9645 077232 160204                            SUB R2,R4 ;AT END OF RECORD YET
9646 077234 020403                            CMP R4,R3 ;BR, IF MORE DATA TO CHECK
9647 077236 001360                            BNE 2851 ;BUMP RECORD SIZE
9648 077240 005723                            TST (R3). ;DONE YET
9649 077242 020327 000050                    CMP R3,@40. ;BR, IF NOT DONE YET (MORE READS)
9650 077246 001317                            BNE 2651 ;LOOP IF SELECTED
9651 077250 3001: CKLOOP                      ;LOOP IF SELECTED                                TRAP C1CLP1
077250 104406
9652 077252 3301:

```



```

9653 077252                ENDSUB                                ;>>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
          077252                L10125:
          077252 104403                TRAP              C#ESUB
9654 077254 023727 002212 000017      CMP      FATFLG,#15.     ;IS ERROR COUNT AT 25
9655 077262 103402                BLO      999#           ;BR, IF LESS THAN 25
9656 077264 004737 017272                JSR      PC,CKDROP     ;TRY TO DROP THE UNIT
9657 077270                999#:
```

```

9658 :
9659 :*
9660 :
9661 ;TEST 7, SUBTEST 4
9662 :
9663 ;VERIFIES THAT A WRITE DATA RETRY COMMAND WITH SWB=1
9664 ;TERMINATES PROPERLY AND WRITES CORRECT DATA ON TAPE
9665 ;(THE WRITTEN RECORD IS READ AND CHECKED). VARIOUS
9666 ;BYTE COUNTS AND DATA PATTERNS ARE USED.
9667 :
9668 :
9669 :
9670 :-
```

```

9671 077270                BGNSUB                                ;>>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
          077270                T7.4:
          077270 104402                TRAP              C#BSUB
9672 077272 004737 105104                JSR      PC,T27REST    ;SET COMMAND PACKET
9673 077276 004737 105176                JSR      PC,T27RT2     ;SET UP OTHER COMMAND PACKET
9674 077302 004737 105240                JSR      PC,T27RT3     ;SET UP OTHER COMMAND PACKET
9675 077306 012737 176750 102132        MOV      #65000.,T27DLY ;SET UP DELAY COUNTER
```

```

9676 :
9677 ;*****
9678 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
9679 :
9680 ;*****
```

```

9681 :
9682 :
9683 077314 004737 016064 10#:          JSR      PC,SOFINIT    ;DO INITIALIZE ON CONTROLLER
9684 077320 103426                BCS      20#           ;BR IF INIT WAS OK
9685 077322                DELAY      250           ;DELAY ABOUT .25 SEC
          077322 012727 000250                MOV      #250,(PC)
          077326 000000                .WORD   0
          077330 013727 002116                MOV      L#DLY,(PC)
          077334 000000                .WORD   0
          077336 005367 177772                DEC      -6(PC)
          077342 001375                BNE      -4
          077344 005367 177756                DEC      -22(PC)
          077350 001367                BNE      -20
```

```

9686 077352 005337 102132                DEC      T27DLY       ;BUMP COUNTER
9687 077356 001356                BNE      10#          ;BR, IF COUNTER NOT DONE
9688 077360 005237 002212                INC      FATFLG       ;BUMP COUNT
9692 077364 010001                MOV      R0,R1        ;CONTENTS OF TSSR REGISTER
9693 077366                ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
          077366 104455                TRAP              C#ERRDF
          077370 001327                .WORD   727
          077372 003650                .WORD   SFIERR
          077374 012124                .WORD   SFIMSG
```

```

9694 077376 013737 002172 101770 20#:    MOV      UNITN,T27DSW  ;SET UP UNIT (DRIVE) NUMBER
9695 077404 012704 101750                MOV      #T27PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
9696
```

```

9697 ;*****
9698 ;
9699 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
9700 ;
9701 ;*****
9702
9703 077410 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
9704 077414 103407 BCS 23$ ;BR, IF COMMAND ISSUED OK
9705 077416 005237 002212 INC FATFLG ;BUMP COUNT
9709 077422 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
9710 077424 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
077424 104456 TRAP C$ERHRD
077426 001330 .WORD 728
077430 005054 .WORD WRTMSG
077432 012124 .WORD SFMSG
9711 077434 23$: CKLOOP ;LOOP IF SELECTED
077434 104406 TRAP C$CLP1
9712
9713 ;*****
9714 ;
9715 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9716 ;
9717 ;*****
9718
9719 077436 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9720 077442 103411 BCS 30$ ;BR, IF NO PROBLEM
9721 077444 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9722 077450 010004 MOV R0,R4 ;GET PACKET ADDRESS
9723 077452 005237 002212 INC FATFLG ;BUMP COUNT
9727 077456 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
077456 104456 TRAP C$ERHRD
077460 001331 .WORD 729
077462 103305 .WORD T27RWN
077464 012136 .WORD PKTSSR
9728 077466 30$: CKLOOP ;LOOP IF SELECTED
077466 104406 TRAP C$CLP1
9729
9730 ;*****
9731 ;
9732 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9733 ;
9734 ;*****
9735
9736 077470 013701 102000 MOV T27BFR+6,R1 ;PICK UP XSTO
9737 077474 010102 MOV R1,R2 ;SET UP EXPECTED
9738 077476 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
9739 077502 020102 CMP R1,R2 ;DOES EXP = REC'D
9740 077504 001406 BEQ 40$ ;BR, IF EQUAL (OK)
9741 077506 005237 002212 INC FATFLG ;BUMP COUNT
9745 077512 ERRHRD ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
077512 104456 TRAP C$ERHRD
077514 001332 .WORD 730
077516 103001 .WORD T27BOT
077520 015564 .WORD EXPREC
9746 077522 40$: CKLOOP ;LOOP IF SELECTED
077522 104406 TRAP C$CLP1
9747 077524 012703 000024 MOV #20.,R3 ;STARTING RECORD SIZE

```

```

9748 077530 013737 003114 102102      MOV      FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
9749
9750      ;*****
9751      ;
9752      ;WRITE DATA,CVC=1,ACK COMMAND
9753      ;
9754      ;*****
9755
9756 077536 012737 140005 102100 65$:  MOV      #140005,T27PK3  ;WRITE DATA,CVC=1,ACK COMMAND
9757 077544 012704 102100      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
9758 077550 010300      MOV      R3,R0           ;SET PATTERN IN CORRECT REGISTER
9759 077552 004737 017512      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
9760 077556 010337 102106      MOV      R3,T27SZ       ;SET UP RECORD SIZE IN PACKET
9761 077562 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
9762 077566 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
9763 077572 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
9764 077576 012702 000200      MOV      #SSR,R2       ;SET UP EXPECTED
9765 077602 020102      CMP      R1,R2         ;ARE THEY EQUAL
9766 077604 001406      BEQ      80$           ;BR, IF OK
9767 077606 005237 002212      INC      FATFLG        ;BUMP COUNT
9771 077612      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
          077612 104456      TRAP      C$ERHRD
          077614 001333      .WORD    731
          077616 005111      .WORD    WRERR
          077620 012136      .WORD    PKTSSR
9772 077622      80$:  CKLOOP           ;LOOP IF SELECTED
          077622 104406      TRAP      C$CLP1
9773
9774      ;*****
9775      ;
9776      ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
9777      ;
9778      ;*****
9779
9780 077624 012737 111005 102100      MOV      #111005,T27PK3  ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
9781 077632 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
9782 077636 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
9783 077642 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
9784 077646 012702 000200      MOV      #SSR,R2       ;SET UP EXPECTED
9785 077652 020102      CMP      R1,R2         ;ARE THEY EQUAL
9786 077654 001406      BEQ      90$           ;BR, IF OK
9787 077656 005237 002212      INC      FATFLG        ;BUMP COUNT
9791 077662      ERRHRD  ERRNO,T27WRF,EXPREC ;TSSR INCORRECT AFTER WRITE DATA RETRY
          077662 104456      TRAP      C$ERHRD
          077664 001334      .WORD    732
          077666 104706      .WORD    T27WRF
          077670 015564      .WORD    EXPREC
9792 077672      90$:  CKLOOP           ;LOOP IF SELECTED
          077672 104406      TRAP      C$CLP1
9793 077674 005723      TST      (R3)+         ;BUMP RECORD SIZE COUNTER
9794 077676 020327 000050      CMP      R3,#40.       ;AT 40 SIZE YET
9795 077702 001315      BNE      65$           ;BR, IF MORE RECORDS TO WRITE
9796
9797      ;*****
9798      ;
9799      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9800      ;

```

```

9801 ;*****
9802
9803 077704 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9804 077710 103411 BCS 230$ ;BR, IF NO PROBLEM
9805 077712 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9806 077716 010004 MOV R0,R4 ;GET PACKET ADDRESS
9807 077720 005237 002212 INC FATFLG ;BUMP COUNT
9811 077724 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
          077724 104456 TRAP C$ERHRD
          077726 001335 .WORD 733
          077730 103305 .WORD T27RWN
          077732 012136 .WORD PKTSSR
9812 077734 230$: CKLOOP ;LOOP IF SELECTED
          077734 104406 TRAP C$CLP1
9813 ;*****
9814 ;
9815 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9816 ;
9817 ;
9818 ;*****
9819
9820 077736 013701 102000 MOV T27BFR+6,R1 ;PICK UP XSTO
9821 077742 010102 MOV R1,R2 ;SET UP EXPECTED
9822 077744 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
9823 077750 020102 CMP R1,R2 ;DOES EXP = REC'D
9824 077752 001406 BEQ 240$ ;BR, IF EQUAL (OK)
9825 077754 005237 002212 INC FATFLG ;BUMP COUNT
9829 077760 ERRHRD ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          077760 104456 TRAP C$ERHRD
          077762 001336 .WORD 734
          077764 103001 .WORD T27BOT
          077766 015564 .WORD EXPREC
9830 077770 240$: CKLOOP ;LOOP IF SELECTED
          077770 104406 TRAP C$CLP1
9831 077772 012703 000024 MOV #20.,R3 ;STARTING RECORD SIZE
9832 077776 013737 003114 102102 MOV FREE,T27RB ;STARTING READ BUFFER ADDRESS
9833 ;*****
9834 ;
9835 ;READ DATA,ACK COMMAND
9836 ;
9837 ;
9838 ;*****
9839
9840 100004 012737 100001 102100 265$: MOV #100001,T27PK3 ;READ DATA,ACK COMMAND
9841 100012 012704 102100 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
9842 100016 010337 102106 MOV R3,T27SZ ;SET UP RECORD SIZE IN PACKET
9843 100022 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
9844 100026 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
9845 100032 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9846 100036 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
9847 100042 020102 CMP R1,R2 ;ARE THEY EQUAL
9848 100044 001406 BEQ 280$ ;BR, IF OK
9849 100046 005237 002212 INC FATFLG ;BUMP COUNT
9853 100052 ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          100052 104456 TRAP C$ERHRD
          100054 001337 .WORD 735
          100056 005204 .WORD RDERR

```



9904 ;3. THE TAPE IS AGAIN REWOUND AND THE SAME SERIES OF  
9905 ; RECORDS WRITTEN AGAIN, THIS TIME USING THE WRITE  
9906 ; DATA RETRY COMMAND. THIS SHOULD RESULT IN  
9907 ; RECORDS SEPARATED BY A LONG INTERRECORD GAP.  
9908 ;  
9909 ;4. THE TAPE IS AGAIN REWOUND, THE SPACING COMMAND  
9910 ; ISSUED, AND THE NUMBER OF TIMING LOOP CYCLES  
9911 ; COUNTED TO COMPLETE THE OPERATION.  
9912 ;  
9913 ;5. THE TWO LOOP COUNTS ARE COMPARED, CHECKING TO SEE  
9914 ; THAT THEY DIFFER BY A SIGNIFICANT AMOUNT.  
9915 ;  
9916 ;  
9917 ;  
9918 ;  
9919 ;  
9920 ;-  
9921 100172 BGNSUB ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>  
100172 T7.5:  
100172 104402 TRAP C#BSUB  
9922 100174 005037 002214 CLR INTRECV ;INTERRUPT INDICATOR  
9923 100200 005037 102126 CLR T27CNT ;TIMER FOR WRITE DATA SPACING  
9924 100204 005037 102130 CLR T27CNU ;TIMER FOR WRITE DATA RETRY SPACING  
9925 100210 004737 105104 JSR PC,T27REST ;SET COMMAND PACKET  
9926 100214 004737 105176 JSR PC,T27RT2 ;SET UP OTHER COMMAND PACKET  
9927 100220 004737 105240 JSR PC,T27RT3 ;SET UP OTHER COMMAND PACKET  
9928 100224 012737 176750 102132 MOV #65000.,T27DLY ;SET UP DELAY COUNTER  
9929 ;  
9930 ;\*\*\*\*\*  
9931 ;  
9932 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR  
9933 ;  
9934 ;\*\*\*\*\*  
9935 ;  
9936 100232 004737 016064 10\$: JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER  
9937 100236 103426 BCS 20\$ ;BR IF INIT WAS OK  
9938 100240 DELAY 250 ;DELAY ABOUT .25 SEC  
100240 012727 000250 MOV #250,(PC)+  
100244 000000 .WORD 0  
100246 013727 002116 MOV L\$DLY,(PC)+  
100252 000000 .WORD 0  
100254 005367 177772 DEC -6(PC)  
100260 001375 BNE .-4  
100262 005367 177756 DEC -22(PC)  
100266 001367 BNE .-20  
9939 100270 005337 102132 DEC T27DLY ;BUMP COUNTER  
9940 100274 001356 BNE 10\$ ;BR, IF COUNTER NOT DONE  
9941 100276 005237 002212 INC FATFLG ;BUMP COUNT  
9945 100302 010001 MOV R0,R1 ;CONTENTS OF TSSR REGISTER  
9946 100304 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK  
100304 104455 TRAP C#ERDF  
100306 001341 .WORD 737  
100310 003650 .WORD SFIERR  
100312 012124 .WORD SFIMSG  
9947 100314 013737 002172 101770 20\$: MOV UNITN,T27DSW ;SET UP UNIT NUMBER  
9948  
9949 100322 012704 101750 MOV #T27PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS

```

9950
9951 ;*****
9952 ;
9953 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
9954 ;
9955 ;*****
9956
9957 100326 004737 010752          JSR    PC,WRTPHR          ;ISSUE WRITE CHARACTERISTICS
9958 100332 103407                BCS    23$                ;BR, IF COMMAND ISSUED OK
9959 100334 005237 002212          INC    FATFLG             ;BUMP COUNT
9963 100340 010001                MOV    R0,R1              ;SAVE CONTENTS OF TSSR
9964 100342                ERRHRD  ERRNO,WRTPHR,SFMSG          ;WRITE CHARACTERISTICS FAILED
                                TRAP    C$ERHRD
                                .WORD   738
                                .WORD   WRTPHR
                                .WORD   SFMSG
100342 104456
100344 001342
100346 005054
100350 012124
9965 100352                23$:   CKLOOP                ;LOOP IF SELECTED
100352 104406                TRAP    C$CLP1
9966
9967 ;*****
9968 ;
9969 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9970 ;
9971 ;*****
9972
9973 100354 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
9974 100360 103411                BCS    30$                ;BR, IF NO PROBLEM
9975 100362 016501 000002          MOV    TSSR(R5),R1        ;GET TSSR CONTENTS
9976 100366 010004                MOV    R0,R4              ;GET PACKET ADDRESS
9977 100370 005237 002212          INC    FATFLG             ;BUMP COUNT
9981 100374                ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   739
                                .WORD   T27RWN
                                .WORD   PKTSSR
100374 104456
100376 001343
100400 103305
100402 012136
9982 100404                30$:   CKLOOP                ;LOOP IF SELECTED
100404 104406                TRAP    C$CLP1
9983
9984 ;*****
9985 ;
9986 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
9987 ;
9988 ;*****
9989
9990 100406 013701 102000          MOV    T27BFR+6,R1        ;PICK UP XST0
9991 100412 010102                MOV    R1,R2              ;SET UP EXPECTED
9992 100414 052702 000002          BIS    #BIT1,R2           ;SET BOT BIT IN EXPECTED
9993 100420 020102                CMP    R1,R2              ;DOES EXP = REC'D
9994 100422 001406                BEQ    40$                ;BR, IF EQUAL (OK)
9995 100424 005237 002212          INC    FATFLG             ;BUMP COUNT
9999 100430                ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   740
                                .WORD   T27BOT
                                .WORD   EXPREC
100430 104456
100432 001344
100434 103001
100436 015564
10000 100440                40$:   CKLOOP                ;LOOP IF SELECTED
100440 104406                TRAP    C$CLP1

```

```

10001 100442 012703 000144          MOV    #100.,R3          ;NUMBER OF RECORDS TO BE WRITTEN
10002 100446 013737 003114 102102  MOV    FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
10003
10004          ;*****
10005          ;
10006          ;WRITE DATA,ACK,CVC=1 COMMAND
10007          ;
10008          ;*****
10009
10010 100454 012737 140005 102100 65$:  MOV    #140005,T27PK3   ;WRITE DATA,ACK,CVC=1 COMMAND
10011 100462 012704 102100          MOV    #T27PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
10012 100466 012737 000024 102106  MOV    #20.,T27SZ     ;SET UP RECORD SIZE IN PACKET
10013 100474 010465 000000          MOV    R4,TSDB(R5)   ;ISSUE COMMAND
10014 100500 004737 016340          JSR    PC,WAITF      ;WAIT FOR SSR TO SET
10015 100504 016501 000002          MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
10016 100510 012702 000200          MOV    #SSR,R2       ;SET UP EXPECTED
10017 100514 020102          CMP    R1,R2         ;ARE THEY EQUAL
10018 100516 001406          BEQ    70$           ;BR, IF OK
10019 100520 005237 002212          INC    FATFLG        ;BUMP COUNT
10023 100524          ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
          TRAP    C$ERHRD
          .WORD   741
          .WORD   WRTErr
          .WORD   PKTSSR
10024 100534          70$:  CKLOOP          ;LOOP IF SELECTED
          TRAP    C$CLP1
10025 100536 005303          DEC    R3            ;DEC RECORD COUNTER
10026 100540 001345          BNE    65$          ;BR, IF MORE RECORDS TO WRITE
10027
10028          ;*****
10029          ;
10030          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10031          ;
10032          ;*****
10033
10034 100542 004737 011104          JSR    PC,REWIND     ;CALL TAPE REWIND COMMAND
10035 100546 103411          BCS    130$         ;BR, IF NO PROBLEM
10036 100550 016501 000002          MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
10037 100554 010004          MOV    R0,R4         ;GET PACKET ADDRESS
10038 100556 005237 002212          INC    FATFLG        ;BUMP COUNT
10042 100562          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
          TRAP    C$ERHRD
          .WORD   742
          .WORD   T27RWN
          .WORD   PKTSSR
10043 100572          130$: CKLOOP          ;LOOP IF SELECTED
          TRAP    C$CLP1
          100572 104406
10044
10045          ;*****
10046          ;
10047          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
10048          ;
10049          ;*****
10050
10051 100574 013701 102000          MOV    T27BFR+6,R1   ;PICK UP XST0
10052 100600 010102          MOV    R1,R2         ;SET UP EXPECTED
10053 100602 052702 000002          BIS    #BIT1,R2      ;SET BOT BIT IN EXPECTED

```



```

10054 100606 020102          CMP      R1,R2          ;DOES EXP = REC'D
10055 100610 001406          BEQ      140$          ;BR, IF EQUAL (OK)
10056 100612 005237 002212  INC      FATFLG        ;BUMP COUNT
10060 100616          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          100616 104456          TRAP      C$ERHRD
          100620 001347          .WORD    743
          100622 103001          .WORD    T27BOT
          100624 015564          .WORD    EXPREC
10061 100626          140$:  CKLOOP          ;LOOP IF SELECTED
          100626 104406          TRAP      C$CLP1
10062 100630 012704 102100  MOV      #T27PK3,R4    ;SET UP PACKET ADDRESS
10063 100634 012737 000010 102102  MOV      #10,T27RB    ;SET UP RECORDS TO SPACE OVER
10064
10065 ;*****
10066 ;
10067 ;ACK,CVC=1,SPACE FORWARD COMMAND
10068 ;
10069 ;*****
10070
10071 100642 012737 140010 102100  MOV      #140010,T27PK3 ;ACK,CVC=1,SPACE FORWARD COMMAND
10072 100650 010465 000000 150$:  MOV      R4,TSDB(R5)  ;ISSUE COMMAND
10073 100654 005237 102126 152$:  INC      T27CNT      ;BUMP TIMER
10074 100660          DELAY    1            ;DELAY ABOUT 100US
          100660 012727 000001          MOV      #1,(PC)+
          100664 000000          .WORD    0
          100666 013727 002116          MOV      L$DLY,(PC)+
          100672 000000          .WORD    0
          100674 005367 177772          DEC      -6(PC)
          100700 001375          BNE     .-4
          100702 005367 177756          DEC      -22(PC)
          100706 001367          BNE     .-20
10075 100710 016501 000002  MOV      TSSR(R5),R1   ;GET TSSR
10076 100714 032701 000200  BIT      #BIT7,R1     ;CHECK FOR TSSR'S SSR SET
10077 100720 001755          BEQ     152$          ;KEEP COUNTING UNTIL SET
10078 100722 016501 000002  MOV      TSSR(R5),R1   ;GET STATUS FROM TSSR
10079 100726 012702 000200  MOV      #SSR,R2      ;SET UP EXPECTED
10080 100732 020201          CMP     R2,R1        ;WAS EVERYTHING OK
10081 100734 001406          BEQ     160$          ;BR, IF ALL IS WELL
10082 100736 005237 002212  INC      FATFLG        ;BUMP COUNT
10086 100742          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD DIDN'T WORK OUT
          100742 104456          TRAP      C$ERHRD
          100744 001350          .WORD    744
          100746 104547          .WORD    T27SCF
          100750 012136          .WORD    PKTSSR
10087 100752          160$:  CKLOOP          ;LOOP IF SELECTED
          100752 104406          TRAP      C$CLP1
10088
10089 ;*****
10090 ;
10091 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10092 ;
10093 ;*****
10094
10095 100754 004737 011104  JSR     PC,REWIND     ;CALL TAPE REWIND COMMAND
10096 100760 004737 016426  JSR     PC,CHKTSSR   ;SEE HOW TSSR IS
10097 100764 103407          BCS     170$          ;BR, IF NO PROBLEM
10098 100766 010001          MOV     R0,R1        ;SAVE TSSR

```

```

10099 100770 005237 002212      INC    FATFLG      ;BUMP COUNT
10103 100774      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      100774 104456      TRAP    C$ERHRD
      100776 001351      .WORD  745
      101000 103305      .WORD  T27RWN
      101002 012136      .WORD  PKTSSR
10104 101004      170$:  CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      101004 104406
10105
10106      ;*****
10107      ;
10108      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10109      ;
10110      ;*****
10111
10112 101006 013701 102000      MOV    T27BFR+6,R1      ;PICK UP XSTO
10113 101012 010102      MOV    R1,R2            ;SET UP EXPECTED
10114 101014 052702 000002      BIS    #BIT1,R2        ;SET BOT BIT IN EXPECTED
10115 101020 020102      CMP    R1,R2            ;DOES EXP = REC'D
10116 101022 001406      BEQ    175$            ;BR, IF EQUAL (OK)
10117 101024 005237 002212      INC    FATFLG          ;BUMP COUNT
10121 101030      ERRHRD  ERRNO.T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      101030 104456      TRAP    C$ERHRD
      101032 001352      .WORD  746
      101034 103001      .WORD  T27BOT
      101036 015564      .WORD  EXPREC
10122 101040      175$:  CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      101040 104406
10123 101042 012703 000144      MOV    #100.,R3        ;STARTING RECORD SIZE
10124 101046 013737 003114 102102 177$:  MOV    FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
10125
10126      ;*****
10127      ;
10128      ;WRITE DATA,CVC=1,ACK COMMAND
10129      ;
10130      ;*****
10131
10132 101054 012737 140005 102100      MOV    #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10133 101062 012704 102100      MOV    #T27PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
10134 101066 012737 000024 102106      MOV    #20.,T27SZ     ;SET UP RECORD SIZE IN PACKET
10135 101074 010465 000000      MOV    R4,TSDB(R5)    ;ISSUE COMMAND
10136 101100 004737 016340      JSR    PC,WAITF       ;WAIT FOR SSR TO SET
10137 101104 016501 000002      MOV    TSSR(R5),R1    ;GET TSSR CONTENTS
10138 101110 012702 000200      MOV    #SSR,R2        ;SET UP EXPECTED
10139 101114 020102      CMP    R1,R2          ;ARE THEY EQUAL
10140 101116 001406      BEQ    180$            ;BR, IF OK
10141 101120 005237 002212      INC    FATFLG          ;BUMP COUNT
10145 101124      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      101124 104456      TRAP    C$ERHRD
      101126 001353      .WORD  747
      101130 005111      .WORD  WRERR
      101132 012136      .WORD  PKTSSR
10146 101134      180$:  CKLOOP      ;LOOP IF SELECTED      TRAP    C$CLP1
      101134 104406
10147 101136 005303      DEC    R3              ;COUNT NUMBER OF RECORDS
10148 101140 001342      BNE    177$            ;BR, IF MORE RECORDS TO WRITE
10149

```

```

10150      ;*****
10151      ;
10152      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10153      ;
10154      ;*****
10155
10156 101142 004737 011104      JSR      PC,REWIND      ;ISSUE REWIND
10157 101146 103411      BCS      182$          ;BR, IF ALL IS WELL
10158 101150 010004      MOV      R0,R4         ;GET PACKET ADDRESS
10159 101152 016501 000002      MCV      TSSR(R5),R1   ;GET TSSR CONTENTS
10160 101156 005237 002212      INC      FATFLG       ;BUMP COUNT
10164 101162      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND FAILED
      101162 104456      TRAP      C$ERHRD
      101164 001354      .WORD    748
      101166 103305      .WORD    T27RWN
      101170 012136      .WORD    PKTSSR
10165 101172      182$:  CKLOOP      ;SELECT LOOP MAYBE
      101172 104406      TRAP      C$CLP1
10166
10167      ;*****
10168      ;
10169      ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
10170      ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
10171      ;
10172      ;*****
10173
10174 101174 012703 000001      MOV      #1.,R3       ;SPACE 1 RECORD FORWARD
10175 101200 004737 010556      JSR      PC,SPACE     ;ISSUE SPACE COMMAND
10176 101204 103411      BCS      185$          ;BR, IF COMMAND OK
10177 101206 010004      MOV      R0,R4         ;GET PACKET ADDRESS
10178 101210 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR STATUS
10179 101214 005237 002212      INC      FATFLG       ;BUMP COUNT
10183 101220      ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD COMMAND FAILED
      101220 104456      TRAP      C$ERHRD
      101222 001355      .WORD    749
      101224 104547      .WORD    T27SCF
      101226 012136      .WORD    PKTSSR
10184 101230      185$:  CKLOOP      ;LOOP IF SELECTED
      101230 104406      TRAP      C$CLP1
10185 101232 012703 000144      MOV      #100.,R3     ;NUMBER OF RECORDS TO BE WRITTEN
10186 101236 013737 003114 102102  MOV      FREE,T27WB    ;STARTING WRITE BUFFER ADDRESS
10187
10188      ;*****
10189      ;
10190      ;WRITE DATA RETRY,ACK COMMAND
10191      ;
10192      ;*****
10193
10194 101244 012737 101005 102100 190$:  MOV      #101005,T27PK3 ;WRITE DATA RETRY,ACK COMMAND
10195 101252 012704 102100      MOV      #T27PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
10196 101256 012737 000024 102106      MOV      #20.,T27SZ   ;SET UP RECORD SIZE IN PACKET
10197 101264 010465 000000      MOV      R4,TSD8(R5)  ;ISSUE COMMAND
10198 101270 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
10199 101274 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
10200 101300 012702 000200      MOV      #SSR,R2     ;SET UP EXPECTED
10201 101304 020102      CMP      R1,R2        ;ARE THEY EQUAL
10202 101306 001406      BEQ      200$        ;BR, IF OK

```

```

10203 101310 005237 002212          INC    FATFLG          ;BUMP COUNT
10207 101314          ERRHRD  ERRNO,T27WDC,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      101314 104456          TRAP    C$ERHRD
      101316 001356          .WORD  750
      101320 103641          .WORD  T27WDC
      101322 012136          .WORD  PKTSSR
10208 101324          200$:  CKLOOP          ;LOOP IF SELECTED
      101324 104406          TRAP    C$CLP1
10209 101326 013737 003114 102102    MOV    FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
10210
10211          ;*****
10212          ;WRITE DATA,CVC=1,ACK COMMAND
10213          ;
10214          ;*****
10215
10216
10217 101334 012737 140005 102100    MOV    @140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10218 101342 012704 102100          MOV    @T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
10219 101346 012737 000024 102106    MOV    @20.,T27SZ      ;SET UP RECORD SIZE IN PACKET
10220 101354 010465 000000          MOV    R4,TSDB(R5)     ;ISSUE COMMAND
10221 101360 004737 016340          JSR    PC,WAITF        ;WAIT FOR SSR TO SET
10222 101364 016501 000002          MOV    TSSR(R5),R1     ;GET TSSR CONTENTS
10223 101370 012702 000200          MOV    @SSR,R2        ;SET UP EXPECTED
10224 101374 020102          CMP    R1,R2          ;ARE THEY EQUAL
10225 101376 001406          BEQ    210$           ;BR, IF OK
10226 101400 005237 002212          INC    FATFLG          ;BUMP COUNT
10230 101404          ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      101404 104456          TRAP    C$ERHRD
      101406 001357          .WORD  751
      101410 005111          .WORD  WRERR
      101412 012136          .WORD  PKTSSR
10231 101414          210$:  CKLOOP          ;LOOP IF SELECTED
      101414 104406          TRAP    C$CLP1
10232 101416 005303          DEC    R3              ;BUMP DOWN RECORD COUNTER
10233 101420 001311          BNE    190$           ;BR, IF MORE RECORDS TO WRITE RETRY
10234
10235          ;*****
10236          ;
10237          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10238          ;
10239          ;*****
10240
10241 101422 004737 011104          JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
10242 101426 103411          BCS    230$           ;BR, IF NO PROBLEM
10243 101430 016501 000002          MOV    TSSR(R5),R1     ;GET TSSR CONTENTS
10244 101434 010004          MOV    R0,R4          ;GET PACKET ADDRESS
10245 101436 005237 002212          INC    FATFLG          ;BUMP COUNT
10249 101442          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      101442 104456          TRAP    C$ERHRD
      101444 001360          .WORD  752
      101446 103305          .WORD  T27RWN
      101450 012136          .WORD  PKTSSR
10250 101452          230$:  CKLOOP          ;LOOP IF SELECTED
      101452 104406          TRAP    C$CLP1
10251
10252          ;*****
10253          ;

```

```

10254 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10255 ;
10256 ;.....
10257
10258 101454 013701 102000          MOV    T27BFR+6,R1      ;PICK UP XSTO
10259 101460 010102                MOV    R1,R2           ;SET UP EXPECTED
10260 101462 052702 000002          BIS    #BIT1,R2        ;SET BOT BIT IN EXPECTED
10261 101466 020102                CMP    R1,R2           ;DOES EXP = REC'D
10262 101470 001406                BEQ    240$            ;BR, IF EQUAL (OK)
10263 101472 005237 002212          INC    FATFLG          ;BUMP COUNT
10267 101476                ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   753
                                .WORD   T27BOT
                                .WORD   EXPREC
                                TRAP    C$CLP1
10268 101506                240$: CKLOOP          ;LOOP IF SELECTED
10269 101510 012704 102100          MOV    #T27PK3,R4      ;SET UP PACKET ADDRESS
10270 101514 012737 000010 102102  MOV    #10,T27RB       ;SET UP RECORDS TO SPACE OVER
10271 ;
10272 ;.....
10273 ;
10274 ;ACK,CVC=1,SPACE FORWARD COMMAND
10275 ;
10276 ;.....
10277
10278 101522 012737 140010 102100    MOV    #140010,T27PK3  ;ACK,CVC=1,SPACE FORWARD COMMAND
10279 101530 010465 000000          250$: MOV    R4,TSDB(R5) ;ISSUE COMMAND
10280 101534 005237 102130          252$: INC    T27CNU     ;BUMP TIMER
10281 101540                DELAY  1               ;DELAY ABOUT 100US
                                MOV    #1,(PC)+
                                .WORD   0
                                MOV    L$DLY,(PC)+
                                .WORD   0
                                DEC    -6(PC)
                                BNE    -.4
                                DEC    -22(PC)
                                BNE    -.20
10282 101570 016501 000002          MOV    TSSR(R5),R1     ;GET TSSR
10283 101574 032701 000200          BIT    #BIT7,R1        ;CHECK FOR TSSR'S SSR SET
10284 101600 001755                BEQ    252$            ;KEEP COUNTING UNTIL SET
10285 101602 016501 000002          MOV    TSSR(R5),R1     ;GET STATUS FROM TSSR
10286 101606 012702 000200          MOV    #SSR,R2         ;SET UP EXPECTED
10287 101612 020201                CMP    R2,R1           ;WAS EVERYTHING OK
10288 101614 001406                BEQ    260$            ;BR, IF ALL IS WELL
10289 101616 005237 002212          INC    FATFLG          ;BUMP COUNT
10293 101622                ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD DIDN'T WORK OUT
                                TRAP    C$ERHRD
                                .WORD   754
                                .WORD   T27SCF
                                .WORD   PKTSSR
10294 101632                260$: CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
10295 101634 013701 102126          MOV    T27CNT,R1       ;TIME FOR WRITE SPACING
10296 101640 013702 102130          MOV    T27CNU,R2      ;TIME FOR WRITE RETRY SPACING
10297 101644 160102                SUB    R1,R2           ;GET'EM PRETTY CLOSE
10298 101646 160102                SUB    R1,R2           ;GET'EM PRETTY CLOSE

```



TSV7 - HARDWARE TESTS 1-8  
TEST 7: WRITE DATA RETRY

MACRO M1113 14-JUN-84 14:17

SEQ 0301

```

10355 102070
10357 102100
10358 102100 100005
10359 102102
10360 102102 003114
10361 102104 000000
10362 102106 000000
10363
10364
10365
10366
10367 102110
10368 102110 010
10369 102111 200
10370 102112 000000
10371 102114 000000
10372
10373
10374
10375
10376
10377 102116 100205
10378 102120 100605
10379 102122 102205
10380 102124 177777
10381
10382
10383 102126 000000
10384 102130 000000
10385 102132 000000
10386
10387
10388
10389
10390
10391
10392
10393 102134 124 141 160
10394 102222 124 123 123
10395 102271 122 105 122
10396 102366 120 117 123
10397 102450 122 111 102
10398 102520 124 123 123
10399 102575 111 154 154
10400 102656 122 105 122
10401 102712 124 123 123
10402 103001 124 141 160
10403 103074 127 122 111
10404 103151 122 105 122
10405 103230 124 123 123
10406 103305 122 145 167
10407 103354 122 101 115
10408 103427 124 123 123
10409 103476 104 162 151
10410 103551 124 123 123
10411 103641 124 123 123
10412 103714 103 126 103

      .BLKB 10-<.-TSV2&7>
T27PK3:
      .WORD 100005 ;REREAD COMMAND, AND ACK
T27RB:
T27WB: .WORD FREE ;ADDRESS OF WRITE BUFFER
      .WORD 0
T27SZ: .WORD 0 ;SIZE OF BUFFER (EXTENT)
      .EVEN
;
;
T27BF2:
T27BS0: .BYTE 10 ;BSELO AREA
T27BS1: .BYTE 200 ;BSEL1 AREA
T27S2: .WORD 0 ;SEL 2 AREA
T27S3: .WORD 0 ;DATA AREA
;
;
      .EVEN
;TAPE MOTION PACKET COMMAND VALUES
T27RN: .WORD 100205 ;REREAD DATA (NEXT)
T27WR: .WORD 100605 ;REREAD DATA RETRY
T27CON: .WORD 102205 ;WRITE CONTINUOUS
      .WORD 177777 ;END OF DATA
;
T27CNT: .WORD 0 ;TAPE TIMER COUNTER STORAGE AREA
T27CNU: .WORD 0 ;TAPE TIMER COUNTER STORAGE AREA
T27DLY: .WORD 0 ;DELAY COUNTER
;
;*
;LOCAL TEXT MESSAGES FOR TEST
;-
T27WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
T27RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
T27RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
T27SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
T27LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
T27WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
T27LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
T27SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
T27WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA RETRY Command, At BOT'
T27BOT: .ASCIZ 'Tape Not At BOT After REWIND Command (BOT Not Set In XSTO)'
T27TIM: .ASCIZ 'WRITE DATA RETRY'S Erase Tape Not Long Enough'
T27EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
T27TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
T27RMN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
T27RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
T27AMS: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
T27OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
T27WDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SWB Bit Set'
T27WDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
T27VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'

```

10413	103767	124	123	102	T27BA:	.ASCIZ	'TSBA Not Correct After REREAD DATA Command'
10414	104042	127	122	111	T27WSS:	.ASCIZ	'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
10415	104131	122	145	141	T27LON:	.ASCIZ	'Reading Long Record Failed To Set RLL Bit In XST0'
10416	104213	122	145	141	T27LOP:	.ASCIZ	'Reading Long Record Failed To Set RLS Bit In XST0'
10417	104275	122	145	163	T27PBP:	.ASCIZ	'Residual Byte Count Incorrect After Short Record Read'
10418	104363	122	145	141	T27TRL:	.ASCIZ	'Reading Long Record Failed To Give Tape Status Alert'
10419	104451	127	122	111	T27NEF:	.ASCIZ	'WRITE DATA RETRY, At First Record, Failed To Set RIB Bit XST3'
10420	104547	124	123	123	T27SCF:	.ASCIZ	'TSSR Not Correct After SPACE RECORDS Command'
10421	104624	124	123	123	T27TSA:	.ASCIZ	'TSSR Not Correct After WRITE DATA RETRY, Into BOT'
10422	104706	124	123	123	T27WRF:	.ASCIZ	'TSSR Not Correct After WRITE DATA RETRY Command'
10423	104766	104	141	164	T27DTA:	.ASCIZ	'Data Compare Error, Data Read From Tape Not Equal To Written'
10424	105063	127	162	151	TST27ID:	.ASCIZ	'Write Data Retry'
10425						.EVEN	
10426							
10427							
10428							
10429							
10430							
10431							
10432							
10433	105104				T27REST:		
10434	105104				SAVREG		;SAVE THE REGISTERS
10435	105110	012701	101750		MOV	#T27PACKET,R1	;START OF THE PACKET
10436	105114	012721	100004		MOV	#100004,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK,
10437	105120	012721	101760		MOV	#T27DATA,(R1)+	;ADDRESS OF CHARAISTICS DATA BLOCK
10438	105124	005021			CLR	(R1)+	;EXTENDED ADDRESS
10439	105126	012721	000012		MOV	#10.,(R1)+	;SIZE OF DATA BLOCK IN BYTES
10440	105132	012721	101772		MOV	#T27BFR,(R1)+	;ADDRESS OF MESSAGE BUFFER
10441	105136	005021			CLR	(R1)+	
10442	105140	012721	000024		MOV	#20.,(R1)+	;LENGTH OF MESSAGE BUFFER
10443	105144	005021			CLR	(R1)+	
10444	105146	012711	000000		MOV	#0,(R1)	;SELECT DRIVE ZERO
10445	105152	012702	000030		MOV	#24.,R2	;NUMBER OF LOCATIONS TO BE CLEARED
10446	105156	012762	177777	101772 64#:	MOV	#177777,T27BFR(R2)	;ALL ONES TO MESSAGE BUFFER
10447	105164	005742			TST	-(R2)	;NEXT LOCATION
10448	105166	022702	000000		CMP	#0,R2	;AT END OF LOOP YET
10449	105172	001371			BNE	64#	;KEEP GOING UNTIL DONE
10450	105174	000207			RTS	PC	;RETURN
10451							
10452							
10453	105176				T27RT2:		
10454	105176				SAVREG		;SAVE THE REGISTERS
10455	105202	012701	102060		MOV	#T27PK2,R1	;START OF THE PACKET
10456	105206	012721	100006		MOV	#100006,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK,
10457	105212	012721	102110		MOV	#T27BF2,(R1)+	;ADDRESS OF DATA BLOCK
10458	105216	005021			CLR	(R1)+	;EXTENDED ADDRESS
10459	105220	012721	000006		MOV	#6.,(R1)+	;SIZE OF DATA BLOCK IN BYTES
10460	105224	005021			CLR	(R1)+	
10461	105226	012701	102110		MOV	#T27BF2,R1	;POINT TO DATA SEL AREA
10462	105232	005021			CLR	(R1)+	
10463	105234	005011			CLR	(R1)	
10464	105236	000207			RTS	PC	;RETURN
10465	105240				T27RT3:		
10466	105240				SAVREG		;SAVE REGISTERS
10467	105244	012701	102100		MOV	#T27PK3,R1	;SET UP POINTER ADDRESS
10468	105250	005021			CLR	(R1)+	;COMMAND SPACE
10469	105252	005021			CLR	(R1)+	;ADDRESS OF DATA BLOCK





	105352	104455							TRAP	C#ERDF
	105354	001441							.WORD	801
	105356	003650							.WORD	SFIERR
	105360	012124							.WORD	SFIMSG
10529	105362	012737	000007	110500	20#:	MOV	#7,T28DSW			;SET UP DRIVE NUMBER
10530	105370	012704	110460			MOV	#T28PACKET,R4			;SUBROUTINE NEEDS PACKET ADDRESS
10531	105374	004737	010752			JSR	PC,WRTCHR			;ISSUE WRITE CHARACTERISTICS
10532	105400	103407				BCS	24#			;BR, IF COMMAND ISSUED OK
10533	105402	005237	002212			INC	FATFLG			;BUMP COUNT
10537	105406	010001				MOV	R0,R1			;SAVE CONTENTS OF TSSR
10538	105410					ERRHRD	ERRNO,WRTMSG,SFIMSG			;WRITE CHARACTERISTISC FAILED
	105410	104456							TRAP	C#ERHRD
	105412	001442							.WORD	802
	105414	005054							.WORD	WRTMSG
	105416	012124							.WORD	SFIMSG
10539	105420				24#:	CKLOOP				
	105420	104406							TRAP	C#CLP1
10540	105422	005737	002216			TST	EXTFEA			;CHECK FOR EXTENDED FEATURES SW SWITCH
10541	105426	001044				BNE	50#			;BR IF SWITCH IS ON
10542										
10543	105430	112737	000200	110621		MOVB	#200,T28BS1			;WRITE MISCELLANEOUS CONT/READ STATUS
10544	105436	112737	000010	110620		MOVB	#10,T28BS0			;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
10545	105444	012704	110570			MOV	#T28PK2,R4			;WRITE SUBSYS MEM PACKET
10546	105450	010465	000000			MOV	R4,TSD8(R5)			;ISSUE COMMAND
10547	105454	004737	016426			JSR	PC,CHKTSSR			;WAIT FOR SSR
10548	105460	103407				BCS	30#			;BR, IF NO ERROR
10549	105462	010001				MOV	R0,R1			;ERROR, SAVE TSSR
10550	105464	005237	002212			INC	FATFLG			;BUMP COUNT
10554	105470					ERRHRD	ERRNO,T28SSR,PKTSSR			;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
	105470	104456							TRAP	C#ERHRD
	105472	001443							.WORD	803
	105474	111315							.WORD	T28SSR
	105476	012136							.WORD	PKTSSR
10555	105500				30#:	CKLOOP				;LOOP IF SELECTED
	105500	104406							TRAP	C#CLP1
10556	105502	012704	110460			MOV	#T28PACKET,R4			;SUBROUTINE NEEDS PACKET ADDRESS
10557	105506	012737	000007	110500		MOV	#7,T28DSW			;SELECT DRIVE 7
10558	105514	004737	010752			JSR	PC,WRTCHR			;ISSUE WRITE CHARACTERISTICS
10559	105520	103407				BCS	50#			;BR, IF COMMAND ISSUED OK
10560	105522	005237	002212			INC	FATFLG			;BUMP COUNT
10564	105526	010001				MOV	R0,R1			;SAVE CONTENTS OF TSSR
10565	105530					ERRHRD	ERRNO,WRTMSG,SFIMSG			;WRITE CHARACTERISTISC FAILED
	105530	104456							TRAP	C#ERHRD
	105532	001444							.WORD	804
	105534	005054							.WORD	WRTMSG
	105536	012124							.WORD	SFIMSG
10566	105540				50#:	CKLOOP				;SCOPE LOOP
	105540	104406							TRAP	C#CLP1
10567	105542	016501	000002			MOV	TSSR(R5),R1			;GET TSSR CONTENTS
10568	105546	032701	000100			BIT	#OFL,R1			;CHECK FOR THE OFFLINE BIT SET
10569	105552	001006				BNE	60#			;BR, IF OFFLINE (GOOD)
10570	105554	005237	002212			INC	FATFLG			;BUMP COUNT
10574	105560					ERRDF	ERRNO,T28OFL,SFIMSG			;OFF LINE SHOULD HAVE BEEN SET (BAD)
	105560	104455							TRAP	C#ERDF
	105562	001445							.WORD	805
	105564	111650							.WORD	T28OFL
	105566	012124							.WORD	SFIMSG



```
105736 003650          .WORD      SFIERR
105740 012124          .WORD      SFIMSG
10626 105742 013737 002172 110500 20:      MOV      UNITN,T28DSW      ;SET UP DRIVE NUMBER
10627 105750 012704 110460      MOV      @T28PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
10628 105754 004737 010752      JSR      PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
10629 105760 103407          BCS      24$            ;BR, IF COMMAND ISSUED OK
10630 105762 005237 002212      INC      FATFLG         ;BUMP COUNT
10634 105766 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
10635 105770          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
105770 104456          TRAP     C#ERHRD
105772 001450          .WORD    808
105774 005054          .WORD    WRTMSG
105776 012124          .WORD    SFIMSG
10636 106000          24$:      CKLOOP      ;LOOP IF SELECTED
106000 104406          TRAP     C#CLP1
10637 106002 012703 110626      MOV      @T28IMV,R3     ;SET POINTER FOR COMMANDS
10638 106006 011337 110610      MOV      (R3),T28PK3    ;SET UP NEXT COMMAND
10639 106012 013737 003114 110612 30:      MOV      FREE,T28RB     ;STARTING WRITE BUFFER ADDRESS
10640 106020 012704 110610      MOV      @T28PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
10641 106024 012737 000400 110616      MOV      @256.,T28SZ    ;SET UP RECORD SIZE IN PACKET
10642 106032 010465 000000      MOV      R4,T28B(R5)    ;ISSUE COMMAND
10643 106036 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR!BIT1!BIT2 TO SET
10644 106042 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
10645 106046 012702 100206      MOV      @SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
10646 106052 020102          CMP      R1,R2          ;ARE THEY EQUAL
10647 106054 001406          BEQ      75$            ;BR, IF OK
10648 106056 005237 002212      INC      FATFLG         ;BUMP COUNT
10652 106062          ERRHRD  ERRNO,T28WDF,PKTSSR ;TSSR INCORRECT AFTER READ DATA
106062 104456          TRAP     C#ERHRD
106064 001451          .WORD    809
106066 111157          .WORD    T28WDF
106070 012136          .WORD    PKTSSR
10653 106072          75$:      CKLOOP      ;LOOP IF SELECTED
106072 104406          TRAP     C#CLP1
10654 106074 013701 110510      MOV      T28BFR+6,R1    ;GET MESSAGE BUFFER
10655 106100 010102          MOV      R1,R2          ;SET UP EXPECTED
10656 106102 052702 001000      BIS      @BIT9,R2       ;SET THE ILC BIT IN EXPECTED
10657 106106 020102          CMP      R1,R2          ;ARE THEY EQUAL
10658 106110 001406          BEQ      180$           ;BR, IF EQUAL (ALL IS WELL)
10659 106112 005237 002212      INC      FATFLG         ;BUMP COUNT
10663 106116          ERRHRD  ERRNO,T28LOQ,EXPREC ;THE ILC BIT WAS NOT SET IN XSTO
106116 104456          TRAP     C#ERHRD
106120 001452          .WORD    810
106122 111234          .WORD    T28LOQ
106124 015564          .WORD    EXPREC
10664 106126          180$:     CKLOOP      ;LOOP IF SELECTED
106126 104406          TRAP     C#CLP1
10665 106130 005723          TST      (R3)+          ;BUMP TO NEXT ADDRESS (COMMAND)
10666 106132 021327 177777      CMP      (R3),@177777  ;CHECK FOR END OF COMMAND BUFFER
10667 106136 001323          BNE      30$            ;BR, IF MORE COMMANDS TO TRY
10668 106140          190$:     CKLOOP      ;LOOP IF SELECTED
106140 104406          TRAP     C#CLP1
10669 106142          ENDSUB      ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
106142          L10132:
106142 104403          TRAP     C#ESUB
10670 106144 023727 002212 000017      CMP      FATFLG,#15.   ;IS ERROR COUNT AT 25
10671 106152 103402          BLO      999$           ;BR, IF LESS THAN 25
```

10672 106154 004737 017272  
 10673 106160  
 10674  
 10675  
 10676  
 10677  
 10678  
 10679  
 10680  
 10681  
 10682  
 10683  
 10684  
 10685  
 10686  
 10687  
 10688  
 10689  
 10690  
 10691  
 10692  
 10693  
 10694  
 10695  
 10696  
 10697  
 10698  
 10699  
 10700  
 10701  
 10702  
 10703  
 10704  
 10705  
 10706  
 10707  
 10708  
 10709  
 10710  
 10711  
 10712  
 10713  
 10714  
 10715  
 10716  
 10717  
 10718  
 10719  
 10720  
 10721  
 10722  
 10723  
 10724  
 10725  
 10726  
 10727  
 10728

```

          JSR      PC,CKDROP                      ;TRY TO DROP THE UNIT
999$:
;
;+
;
;TEST 8, SUBTEST 3
;
;VERIFIES THAT WRITE TAPE MARK COMMANDS OPERATE
;PROPERLY, AND THAT READ COMMANDS SUBSEQUENTLY ISSUED
;TO DETECT THE WRITTEN TAPE MARKS TERMINATE WITH TAPE
;STATUS ALERT WITH THE TAPE MARK DETECTED (TMK) STATUS
;BIT SET. THE FOLLOWING SEQUENCE IS EXECUTED.
;
;1.      THE CONTROLLER IS INITIALIZED AND TAPE REWOUND.
;        THIS SETS THE VOLUME CHECK (VCK) STATUS BIT.
;
;2.      A WRITE TAPE MARK COMMAND WITH CVC=1 IS ISSUED
;        AND PROPER TERMINATION AND STATUS IS VERIFIED
;        (I.E. VCK=0 AND TMK=1).
;
;3.      SEVERAL MORE WRITE TAPE MARK COMMANDS, THESE WITH
;        CVC=0 ARE ISSUED AND PROPER TERMINATION (NORMAL)
;        AND STATUS (TMK) VERIFIED.
;
;4.      A READ REVERSE COMMAND IS ISSUED AND PROPER
;        TERMINATION (TAPE STATUS ALERT) AND STATUS (TMK)
;        VERIFIED. IT IS ALSO VERIFIED THAT NO DATA IS
;        TRANSFERRED INTO MEMORY.
;
;5.      A SPACE RECORDS REVERSE COMMAND IS ISSUED AND
;        PROPER TERMINATION (TAPE STATUS ALERT) AND STATUS
;        (TMK) VERIFIED.
;
;6.      THE TAPE IS REWOUND AND A READ FORWARD COMMAND IS
;        ISSUED AND PROPER TERMINATION (TAPE STATUS ALERT)
;        AND STATUS (TMK) VERIFIED. IT IS ALSO VERIFIED
;        THAT NO DATA IS TRANSFERRED INTO MEMORY.
;
;7.      A SPACE RECORDS REVERSE COMMAND THAT CONTAINS A
;        RECORD COUNT GREATER THAN 1 IS ISSUED AND IT IS
;        VERIFIED THAT TAPE STATUS ALERT TERMINATION
;        OCCURED, TMK=1 AND THAT RBPGR (RESIDUAL
;        BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO
;        VALUE. THIS OPERATION VERIFIES THAT DETECTION OF
;        THE TAPE MARK CAUSES THE SPACE RECORDS OPERATION
;        TO BE PREMATURELY TERMINATED. THIS SHOULD LEAVE
;        THE POSITION JUST BEFORE THE FIRST RECORD ON
;        TAPE.
;
;8.      TAPE POSITION IS VERIFIED BY ISSUING ANOTHER
;        SPACE RECORDS REVERSE COMMAND AND VERIFYING THAT
;        TAPE STATUS ALERT TERMINATION OCCURS, WITH THE
;        REVERSE INTO BOT (RIB) STATUS ERROR BIT SET.
;
;9.      A SPACE RECORDS FORWARD COMMAND THAT CONTAINS A
;        RECORD COUNT GREATER THAN 1 IS ISSUED AND IT IS
;        VERIFIED THAT TAPE STATUS ALERT TERMINATION

```

```

10729          :   OCCURED, TMK=1, AND THAT RBPCR (RESIDUAL
10730          :   BYTE/RECORD COUNTER) CONTAINS THE PROPER NONZERO
10731          :   VALUE. THIS OPERATION VERIFIES THAT DETECTION OF
10732          :   THE TAPE MARK CAUSES THE SPACE RECORDS OPERATION
10733          :   TO BE PREMATURELY TERMINATED.
10734          :
10735          :
10736          :
10737          :
10738          :
10739          :
10740          :-
10741 106160        BGNSUB          ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
106160          T8.3:
106160 104402          TRAP          C#BSUB
10742 106162 004737 112546        JSR          PC,T28REST       ;SET COMMAND PACKET
10743 106166 004737 112640        JSR          PC,T28RT2      ;SET UP OTHER COMMAND PACKET
10744 106172 004737 112702        JSR          PC,T28RT3      ;SET UP OTHER COMMAND PACKET
10745 106176 012737 023420 110652 10$: MOV          #10000.,T28DLY ;SET UP DELAY ROUTINE
10746 106204 004737 016064        JSR          PC,SOFINIT     ;DO INITIALIZE ON CONTROLLER
10747 106210 103426          BCS          20$           ;BR IF INIT WAS OK
10748 106212          DELAY        250             ;DELAY ABOUT .25 SECONDS
106212 012727 000250          MOV          #250.(PC)+
106216 000000          .WORD        0
106220 013727 002116          MOV          L#DLY,(PC)+
106224 000000          .WORD        0
106226 005367 177772          DEC          -6(PC)
106232 001375          BNE          -.4
106234 005367 177756          DEC          -22(PC)
106240 001367          BNE          .-20
10749 106242 005337 110652        DEC          T28DLY        ;BUMP DELAY ROUTINE DOWN
10750 106246 001356          BNE          10$          ;BR, IF MORE DELAY TIME LEFT
10751 106250 005237 002212        INC          FATFLG       ;BUMP COUNT
10755 106254 010001        MOV          RO,R1        ;CONTENTS OF TSSR REGISTER
10756 106256          ERRDF      ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
106256 104455          TRAP          C#ERDF
106260 001453          .WORD        811
106262 003650          .WORD        SFIERR
106264 012124          .WORD        SFIMSG
10757 106266 013737 002172 110500 20$: MOV          UNITN,T28DSW  ;SET UP DRIVE NUMBER
10758 106274 012704 110460        MOV          #T28PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
10759 106300 004737 010752        JSR          PC,WRTCHR     ;ISSUE WRITE CHARACTERISTICS
10760 106304 103407          BCS          23$          ;BR, IF COMMAND ISSUED OK
10761 106306 005237 002212        INC          FATFLG       ;BUMP COUNT
10765 106312 010001        MOV          RO,R1        ;SAVE CONTENTS OF TSSR
10766 106314          ERRHRD    ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
106314 104456          TRAP          C#ERHRD
106316 001454          .WORD        812
106320 005054          .WORD        WRTMSG
106322 012124          .WORD        SFIMSG
10767 106324          23$:  CKLOOP          ;LOOP IF SELECTED
106324 104406          TRAP          C#CLP1
10768 106326 004737 011104        JSR          PC,REWIND     ;CALL TAPE REWIND COMMAND
10769 106332 103411          BCS          30$          ;BR, IF NO PROBLEM
10770 106334 016501 000002        MOV          TSSR(R5),R1  ;GET TSSR
10771 106340 010004        MOV          RO,R4        ;SAVE PACKET ADDRESS
10772 106342 005237 002212        INC          FATFLG       ;BUMP COUNT

```

```

10776 106346          ERRHRD  ERRNO,T28RWN,PKTSSR      ;REWIND NOT ACCEPTED
      106346 104456          TRAP          C#ERHRD
      106350 001455          .WORD          813
      106352 111601          .WORD          T28RWN
      106354 012136          .WORD          PKTSSR
10777 106356          30$:  CKLOOP          ;LOOP IF SELECTED
      106356 104406          TRAP          C#CLP1
10778 106360 013701 110510      MOV      T28BFR+6,R1      ;PICK UP XSTO
10779 106364 010102          MOV      R1,R2          ;SET UP EXPECTED
10780 106366 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
10781 106372 020102          CMP      R1,R2          ;DOES EXP = REC'D
10782 106374 001406          BEQ      40$          ;BR, IF EQUAL (OK)
10783 106376 005237 002212      INC      FATFLG        ;BUMP COUNT
10787 106402          ERRHRD  ERRNO,T28BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      106402 104456          TRAP          C#ERHRD
      106404 001456          .WORD          814
      106406 111457          .WORD          T28BOT
      106410 015564          .WORD          EXPREC
10788 106412          40$:  CKLOOP          ;LOOP IF SELECTED
      106412 104406          TRAP          C#CLP1
10789 106414 005737 002216      42$:  TST      EXTFEA      ;CHECK FOR EXTENDED FEATURES SW SWITCH
10790 106420 001024          BNE      50$          ;BR IF SWITCH IS ON
10791 106422 112737 000200 110621  MOVB     #200,T28BS1      ;WRITE MISCELLANEOUS CONT/READ STATUS
10792 106430 112737 000010 110620  MOVB     #10,T28BS0      ;FUNC. SEL. BIT (TURN ON EXTFEA SWITCH)
10793 106436 012704 110570      MOV      #T28PK2,R4      ;WRITE SUBSYS MEM PACKET
10794 106442 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
10795 106446 004737 016426      JSR      PC,CHKTSSR      ;WAIT FOR SSR
10796 106452 103407          BCS      50$          ;BR, IF NO ERROR
10797 106454 010001          MOV      R0,R1          ;ERROR, SAVE TSSR
10798 106456 005237 002212      INC      FATFLG        ;BUMP COUNT
10802 106462          ERRHRD  ERRNO,T28SSR,PKTSSR      ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      106462 104456          TRAP          C#ERHRD
      106464 001457          .WORD          815
      106466 111315          .WORD          T28SSR
      106470 012136          .WORD          PKTSSR
10803 106472          50$:  CKLOOP          ;LOOP IF SELECTED
      106472 104406          TRAP          C#CLP1
10804 106474 012737 000007 110500  MOV      #7,T28DSW      ;SET UP DRIVE NUMBER
10805 106502 012704 110460      MOV      #T28PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
10806 106506 004737 010752      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
10807 106512 103407          BCS      60$          ;BR, IF COMMAND ISSUED OK
10808 106514 005237 002212      INC      FATFLG        ;BUMP COUNT
10812 106520 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
10813 106522          ERRHRD  ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTICSC FAILED
      106522 104456          TRAP          C#ERHRD
      106524 001460          .WORD          816
      106526 005054          .WORD          WRTMSG
      106530 012124          .WORD          SFIMSG
10814 106532          60$:  CKLOOP          ;SCOPE LOOP
      106532 104406          TRAP          C#CLP1
10815 106534 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
10816 106540 032701 000100      BIT      #OFL,R1        ;CHECK FOR THE OFFLINE BIT SET
10817 106544 001006          BNE      65$          ;BR, IF OFFLINE (GOOD)
10818 106546 005237 002212      INC      FATFLG        ;BUMP COUNT
10822 106552          ERRDF  ERRNO,T28OFL,SFIMSG      ;OFF LINE SHOULD HAVE BEEN SET (BAD)
      106552 104455          TRAP          C#ERDF
      106554 001461          .WORD          817

```

TSV7 - HARDWARE TESTS 1-8  
TEST 8: WRITE/READ TAPE MARK

MACRO M1113 14-JUN 84 14:17

SEQ 0310

	106556	111650						.WORD	T28OFL
	106560	012124						.WORD	SFIMSG
10823	106562			65:	CKLOOP				;LOOP IF SELECTED
	106562	104406						TRAP	C#CLP1
10824	106564	013737	002172	110500	MOV	UNITN,T28DSW			;SET UP DRIVE NUMBER
10825	106572	012704	110460		MOV	#T28PACKET,R4			;SUBROUTINE NEEDS PACKET ADDRESS
10826	106576	004737	010752		JSR	PC,WRTCHR			;ISSUE WRITE CHARACTERISTICS
10827	106602	103407			BCS	68:			;BR, IF COMMAND ISSUED OK
10828	106604	005237	002212		INC	FATFLG			;BUMP COUNT
10832	106610	010001			MOV	R0,R1			;SAVE CONTENTS OF TSSR
10833	106612				ERRHRD	ERRNO,WRTMSG,SFIMSG			;WRITE CHARACTERISTIC FAILED
	106612	104456						TRAP	C#ERHRD
	106614	001462						.WORD	818
	106616	005054						.WORD	WRTMSG
	106620	012124						.WORD	SFIMSG
10834	106622			68:	CKLOOP				;LOOP IF SELECTED
	106622	104406						TRAP	C#CLP1
10835	106624	012737	140011	110610	MOV	#140011,T28PK3			;WRITE TAPE MARK,ACK,CVC=1 COMMAND
10836	106632	012704	110610		MOV	#T28PK3,R4			;SET UP R4 WITH PACKET ADDRESS
10837	106636	010465	000000		MOV	R4,TSDB(R5)			;ISSUE COMMAND
10838	106642	004737	016340		JSR	PC,WAITF			;WAIT FOR SSR TO SET
10839	106646	016501	000002		MOV	TSSR(R5),R1			;GET TSSR CONTENTS
10840	106652	012702	000200		MOV	#SSR,R2			;SET UP EXPECTED
10841	106656	020102			CMP	R1,R2			;ARE THEY EQUAL
10842	106660	001406			BEQ	70:			;BR, IF OK
10843	106662	005237	002212		INC	FATFLG			;BUMP COUNT
10847	106666				ERRHRD	ERRNO,T28WDC,PKTSSR			;TSSR INCORRECT AFTER WRITE TAPE MARK
	106666	104456						TRAP	C#ERHRD
	106670	001463						.WORD	819
	106672	111723						.WORD	T28WDC
	106674	012136						.WORD	PKTSSR
10848	106676			70:	CKLOOP				;LOOP IF SELECTED
	106676	104406						TRAP	C#CLP1
10849	106700	013701	110510		MOV	T28BFR+6,R1			;PICK UP XSTO (VCK CHECK)
10850	106704	010102			MOV	R1,R2			;SET UP EXPECTED
10851	106706	042702	000020		BIC	#BIT4,R2			;VCK SHOULD BE 0
10852	106712	020102			CMP	R1,R2			;IS VCK SET CORRECTLY
10853	106714	001406			BEQ	80:			;BR, IF VCK IS CLEAR
10854	106716	005237	002212		INC	FATFLG			;BUMP COUNT
10858	106722				ERRHRD	ERRNO,T28VCK,EXPREC			;VCK WAS NOT CLEAR AFTER CVC=1
	106722	104456						TRAP	C#ERHRD
	106724	001464						.WORD	820
	106726	112002						.WORD	T28VCK
	106730	015564						.WORD	EXPREC
10859	106732			80:	CKLOOP				;LOOP IF SELECTED
	106732	104406						TRAP	C#CLP1
10860	106734	013701	110510		MOV	T28BFR+6,R1			;PICK UP XSTO (CHECK TMK)
10861	106740	010102			MOV	R1,R2			;SET UP EXPECTED
10862	106742	052702	100000		BIS	#BIT15,R2			;TMK SHOULD BE SET
10863	106746	020102			CMP	R1,R2			;WAS TMK SET
10864	106750	001406			BEQ	90:			;BR, IF TMK WAS SET
10865	106752	005237	002212		INC	FATFLG			;BUMP COUNT
10869	106756				ERRHRD	ERRNO,T28TMK,EXPREC			;TMK WAS NOT SET AFTER WRT TAPE MARK
	106756	104456						TRAP	C#ERHRD
	106760	001465						.WORD	821
	106762	112055						.WORD	T28TMK
	106764	015564						.WORD	EXPREC



10870	106766			90\$:	CKLOOP		:LOOP IF SELECTED		
	106766	104406						TRAP	C\$ERHRD
10871	106770	004737	011104		JSR	PC,REWIND	:CALL TAPE REWIND COMMAND		
10872	106774	103411			BCS	130\$	:BR, IF NO PROBLEM		
10873	106776	010004			MOV	R0,R4	:SAVE PACKET ADDRESS		
10874	107000	016501	000002		MOV	TSSR(R5),R1	:GET TSSR STATUS		
10875	107004	005237	002212		INC	FATFLG	:BUMP COUNT		
10879	107010				ERRHRD	ERRNO,T28RWN,PKTSSR	:REWIND NOT ACCEPTED		
	107010	104456						TRAP	C\$ERHRD
	107012	001466						.WORD	822
	107014	111601						.WORD	T28RWN
	107016	012136						.WORD	PKTSSR
10880	107020			130\$:	CKLOOP		:LOOP IF SELECTED		
	107020	104406						TRAP	C\$CLP1
10881	107022	013701	110510		MOV	T28BFR+6,R1	:PICK UP XSTO		
10882	107026	010102			MOV	R1,R2	:SET UP EXPECTED		
10883	107030	052702	000002		BIS	#BIT1,R2	:SET BOT BIT IN EXPECTED		
10884	107034	020102			CMP	R1,R2	:DOES EXP = REC'D		
10885	107036	001406			BEQ	140\$	:BR, IF EQUAL (OK)		
10886	107040	005237	002212		INC	FATFLG	:BUMP COUNT		
10890	107044				ERRHRD	ERRNO,T28BOT,EXPREC	:TAPE NOT AT BOT AFTER REWIND		
	107044	104456						TRAP	C\$ERHRD
	107046	001467						.WORD	823
	107050	111457						.WORD	T28BOT
	107052	015564						.WORD	EXPREC
10891	107054			140\$:	CKLOOP		:LOOP IF SELECTED		
	107054	104406						TRAP	C\$CLP1
10892	107056	012703	000012		MOV	#10.,R3	:NUMBER OF RECORDS TO WRITE TM		
10893	107062	012737	140011	110610	MOV	#140011,T28PK3	:WRITE TAPE MARK,ACK,CVC=1 COMMAND		
10894	107070	012704	110610		MOV	#T28PK3,R4	:SET UP R4 WITH PACKET ADDRESS		
10895	107074	010465	000000		MOV	R4,TSD8(R5)	:ISSUE COMMAND		
10896	107100	004737	016340		JSR	PC,WAITF	:WAIT FOR SSR TO SET		
10897	107104	016501	000002		MOV	TSSR(R5),R1	:PICK UP TSSR		
10898	107110	012702	000200		MOV	#SSR,R2	:SET UP EXPECTED (SSR ONLY)		
10899	107114	020102			CMP	R1,R2	:WAS STATUS GOOD		
10900	107116	001406			BEQ	165\$	:BR, IF TERMINATION WAS GOOD		
10901	107120	005237	002212		INC	FATFLG	:BUMP COUNT		
10905	107124				ERRHRD	ERRNO,T28WDC,PKTSSR	:TSSR NOT CORRECT AFTER WRT TAPE M.		
	107124	104456						TRAP	C\$ERHRD
	107126	001470						.WORD	824
	107130	111723						.WORD	T28WDC
	107132	012136						.WORD	PKTSSR
10906	107134			165\$:	CKLOOP		:LOOP IF SELECTED		
	107134	104406						TRAP	C\$CLP1
10907	107136	013701	110510		MOV	T28BFR+6,R1	:PICK UP XSTO		
10908	107142	010102			MOV	R1,R2	:SET UP EXPECTED		
10909	107144	052702	100000		BIS	#BIT15,R2	:SET TMK BIT IN EXPECTED		
10910	107150	020102			CMP	R1,R2	:DOES EXP = REC'D		
10911	107152	001406			BEQ	180\$	:BR, IF EQUAL (OK)		
10912	107154	005237	002212		INC	FATFLG	:BUMP COUNT		
10916	107160				ERRHRD	ERRNO,T28TMK,EXPREC	:TMK NOT SET AFTER WRT TAPE MARK		
	107160	104456						TRAP	C\$ERHRD
	107162	001471						.WORD	825
	107164	112055						.WORD	T28TMK
	107166	015564						.WORD	EXPREC
10917	107170			180\$:	CKLOOP		:LOOP IF SELECTED		
	107170	104406						TRAP	C\$CLP1

10918	107172	005303			DEC	R3			;BUMP COUNTER DOWN
10919	107174	001337			BNE	1554			;BR, IF LESS THAN 10 TAPE MARKS
10920	107176	012700	177777		MOV	#177777,R0			;VALUE TO WRITTEN TO MEMORY
10921	107202	004737	017512		JSR	PC,FILLMEM			;FILL MEM WITH ALL ONES
10922	107206	013737	003114	110612	MOV	FREE,T28WB			;STARTING READ BUFFER ADDRESS
10923	107214	012737	140401	110610	MOV	#140401,T28PK3			;READ REVERSE,ACK, COMMAND
10924	107222	012704	110610		MOV	#T28PK3,R4			;SET UP R4 WITH PACKET ADDRESS
10925	107226	013737	000024	110616	MOV	20.,T28SZ			;SET UP RECORD SIZE IN PACKET
10926	107234	010465	000000		MOV	R4,TSD8(R5)			;ISSUE COMMAND
10927	107240	004737	016340		JSR	PC,WAITF			;WAIT FOR SSR TO SET
10928	107244	016501	000002		MOV	TSSR(R5),R1			;GET TSSR CONTENTS
10929	107250	012702	100204		MOV	#SSR!SC!BIT2,R2			;SET UP EXPECTED
10930	107254	020102			CMP	R1,R2			;ARE THEY EQUAL
10931	107256	001406			BEQ	2004			;BR, IF OK
10932	107260	005237	002212		INC	FATFLG			;BUMP COUNT
10936	107264				ERRHRD	ERRNO,T28RDF,PKTSSR			;TSSR INCORRECT AFTER WRITE DATA
	107264	104456						TRAP	C#ERRRD
	107266	001472						.WORD	826
	107270	111014						.WORD	T28RDF
	107272	012136						.WORD	PKTSSR
10937	107274			2004:	CKLOOP				;LOOP IF SELECTED
	107274	104406						TRAP	C#CLP1
10938	107276	013701	110510		MOV	T28FR+6,R1			;PICK UP XSTO
10939	107302	010102			MOV	R1,R2			;SET UP EXPECTED
10940	107304	052702	100300		BIS	#BIT15,R2			;TMK SHOULD BE SET
10941	107310	020102			CMP	R1,R2			;IS TMK SET
10942	107312	001406			BEQ	2104			;BR, IF TMK WAS SET (GOOD)
10943	107314	005237	002212		INC	FATFLG			;BUMP COUNT
10947	107320				ERRHRD	ERRNO,T28RRM,EXPREC			;TMK NOT SET AFTER READ REV
	107320	104456						TRAP	C#ERRRD
	107322	001473						.WORD	827
	107324	112127						.WORD	T28RRM
	107326	015564						.WORD	EXPREC
10948	107330			2104:	CKLOOP				;LOOP IF SELECTED
	107330	104406						TRAP	C#CLP1
10949	107332	017701	073556		MOV	#FREE,R1			;FIRST LOC IN READ BUFFER
10950	107336	012702	177777		MOV	#177777,R2			;EXPECTED IF NO DATA TRANS.
10951	107342	020102			CMP	R1,R2			;DID ANY DATA GET TRANSFERRED
10952	107344	001406			BEQ	2204			;BR, IF NO DATA TRANS (GOOD)
10953	107346	005237	002212		INC	FATFLG			;BUMP COUNT
10957	107352				ERRHRD	ERRNO,T28DTR,EXPREC			;DATA TRANSFERRED ON READ TAPE MARK
	107352	104456						TRAP	C#ERRRD
	107354	001474						.WORD	828
	107356	112342						.WORD	T28DTR
	107360	015564						.WORD	EXPREC
10958	107362			2204:	CKLOOP				;LOOP IF SELECTED
	107362	104406						TRAP	C#CLP1
10959	107364	012737	100410	110610	MOV	#100410,T28PK3			;SPACE REVERSE,ACK, COMMAND
10960	107372	012737	000001	110612	MOV	#1,T28RB			;NUMBER OF RECORDS TO SPACE BACK
10961	107400	012704	110610		MOV	#T28PK3,R4			;SET UP R4 WITH PACKET ADDRESS
10962	107404	010465	000000		MOV	R4,TSD8(R5)			;ISSUE COMMAND
10963	107410	004737	016340		JSR	PC,WAITF			;WAIT FOR SSR TO SET
10964	107414	016501	000002		MOV	TSSR(R5),R1			;GET TSSR CONTENTS
10965	107420	012702	100204		MOV	#SSR!SC!BIT2,R2			;SET UP EXPECTED
10966	107424	020102			CMP	R1,R2			;ARE THEY EQUAL
10967	107426	001406			BEQ	2224			;BR, IF OK
10968	107430	005237	002212		INC	FATFLG			;BUMP COUNT

10972	107434			ERRHRD	ERRNO,T28RDG,PKTSSR			;TSSR INCORRECT AFTER SPACE CMD.		
	107434	104456						TRAP	C1ERHRD	
	107436	001475						.WORD	829	
	107440	111075						.WORD	T28RDG	
	107442	012136						.WORD	PKTSSR	
10973	107444			2224:	CKLOOP			;LOOP IF SELECTED		
	107444	104406						TRAP	C1CLP1	
10974	107446	013701	110510		MOV	T28BFR+6,R1		;PICK UP XSTO		
10975	107452	010102			MOV	R1,R2		;SET UP EXPECTED		
10976	107454	052702	100000		BIS	#BIT15,R2		;TMK SHOULD BE SET		
10977	107460	020102			CMP	R1,R2		;IS TMK SET		
10978	107462	001406			BEQ	2264		;BR, IF TMK WAS SET (GOOD)		
10979	107464	005237	002212		INC	FATFLG		;BUMP COUNT		
10983	107470				ERRHRD	ERRNO,T28RRN,EXPREC		;TMK NOT SET AFTER SPACE REV		
	107470	104456						TRAP	C1ERHRD	
	107472	001476						.WORD	830	
	107474	112205						.WORD	T28RRN	
	107476	015564						.WORD	EXPREC	
10984	107500			2264:	CKLOOP			;LOOP IF SELECTED		
	107500	104406						TRAP	C1CLP1	
10985	107502	004737	011104		JSR	PC,REWIND		;CALL TAPE REWIND COMMAND		
10986	107506	103411			BCS	2304		;BR, IF NO PROBLEM		
10987	107510	010004			MOV	R0,R4		;SAVE PACKET ADDRESS		
10988	107512	016501	000002		MOV	TSSR(R5),R1		;GET TSSR		
10989	107516	005237	002212		INC	FATFLG		;BUMP COUNT		
10993	107522				ERRHRD	ERRNO,T28RWN,PKTSSR		;REWIND NOT ACCEPTED		
	107522	104456						TRAP	C1ERHRD	
	107524	001477						.WORD	831	
	107526	111601						.WORD	T28RWN	
	107530	012136						.WORD	PKTSSR	
10994	107532			2304:	CKLOOP			;LOOP IF SELECTED		
	107532	104406						TRAP	C1CLP1	
10995	107534	013701	110510		MOV	T28BFR+6,R1		;PICK UP XSTO		
10996	107540	010102			MOV	R1,R2		;SET UP EXPECTED		
10997	107542	052702	000002		BIS	#BIT1,R2		;SET BOT BIT IN EXPECTED		
10998	107546	020102			CMP	R1,R2		;DOES EXP = REC'D		
10999	107550	001406			BEQ	2404		;BR, IF EQUAL (OK)		
11000	107552	005237	002212		INC	FATFLG		;BUMP COUNT		
11004	107556				ERRHRD	ERRNO,T28BOT,EXPREC		;TAPE NOT AT BOT AFTER REWIND		
	107556	104456						TRAP	C1ERHRD	
	107560	001500						.WORD	832	
	107562	111457						.WORD	T28BOT	
	107564	015564						.WORD	EXPREC	
11005	107566			2404:	CKLOOP			;LOOP IF SELECTED		
	107566	104406						TRAP	C1CLP1	
11006	107570	012700	177777		MOV	#177777,R0		;VALUE TO WRITTEN TO MEMORY		
11007	107574	004737	017512		JSR	PC,FILLMEM		;FILL MEM WITH ALL ONES		
11008	107600	013737	003114	110612	MOV	FREE,T28RB		;STARTING READ BUFFER ADDRESS		
11009	107606	012737	100001	110610	MOV	#100001,T28PK3		;READ FORWARD,ACK, COMMAND		
11010	107614	012704	110610		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS		
11011	107620	013737	000024	110616	MOV	20.,T28SZ		;SET UP RECORD SIZE IN PACKET		
11012	107626	010465	000000		MOV	R4,TSD8(R5)		;ISSUE COMMAND		
11013	107632	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET		
11014	107636	016501	000002		MOV	TSSR(R5),R1		;GET TSSR CONTENTS		
11015	107642	012702	100204		MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED		
11016	107646	020102			CMP	R1,R2		;ARE THEY EQUAL		
11017	107650	001406			BEQ	2454		;BR, IF OK		



	110062	104456					TRAP	C#ERHRD
	110064	001505					.WORD	837
	110066	112205					.WORD	T28ARN
	110070	015564					.WORD	EXPREC
11070	110072			2704:	CKLOOP			;LOOP IF SELECTED
	110072	104406					TRAP	C#CLP1
11071	110074	013701	110506		MOV	T28BFR+4,R1		;PICK UP RESIDUAL BYTE COUNTER
11072	110100	012702	000004		MOV	#4.,R2		;SHOULD BE THE DIFFERENCE
11073	110104	020102			CMP	R1,R2		;IS COUNTER CORRECT
11074	110106	001406			BEQ	2804		;BR, IF COUNTER CORRECT
11075	110110	005237	002212		INC	FATFLG		;BUMP COUNT
11079	110114				ERRHRD	ERRNO,T28PBP,EXPREC		;RESIDUAL BYTE COUNTER NOT CORRECT
	110114	104456					TRAP	C#ERHRD
	110116	001506					.WORD	838
	110120	110731					.WORD	T28PBP
	110122	015564					.WORD	EXPREC
11080	110124			2804:	CKLOOP			;LOOP IF SELECTED
	110124	104406					TRAP	C#CLP1
11081	110126	012737	100410	110610	MOV	#100410,T28PK3		;SPACE REVERSE,ACK, COMMAND
11082	110134	012737	000001	110612	MOV	#1,T28RB		;NUMBER OF RECORDS TO SPACE BACK
11083	110142	012704	110610		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
11084	110146	010465	000000		MOV	R4,TSD8(R5)		;ISSUE COMMAND
11085	110152	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET
11086	110156	016501	000002		MOV	TSSR(R5),R1		;GET TSSR CONTENTS
11087	110162	012702	100204		MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED
11088	110166	020102			CMP	R1,R2		;ARE THEY EQUAL
11089	110170	001406			BEQ	2904		;BR, IF OK
11090	110172	005237	002212		INC	FATFLG		;BUMP COUNT
11094	110176				ERRHRD	ERRNO,T28RDG,PKTSSR		;TSSR INCORRECT AFTER SPACE CMD.
	110176	104456					TRAP	C#ERHRD
	110200	001507					.WORD	839
	110202	111075					.WORD	T28RDG
	110204	012136					.WORD	PKTSSR
11095	110206			2904:	CKLOOP			;LOOP IF SELECTED
	110206	104406					TRAP	C#CLP1
11096	110210	013701	110516		MOV	T28BFR+14,R1		;PICK UP XST3
11097	110214	010102			MOV	R1,R2		;SET UP EXPECTED
11098	110216	052702	000001		BIS	#BIT0,R2		;RIB SHOULD BE SET
11099	110222	020102			CMP	R1,R2		;IS RIB SET
11100	110224	001406			BEQ	3004		;BR, IF RIB WAS SET (GOOD)
11101	110226	005237	002212		INC	FATFLG		;BUMP COUNT
11105	110232				ERRHRD	ERRNO,T28RIB,EXPREC		;RIB NOT SET AFTER READ REV
	110232	104456					TRAP	C#ERHRD
	110234	001510					.WORD	840
	110236	110654					.WORD	T28RIB
	110240	015564					.WORD	EXPREC
11106	110242			3004:	CKLOOP			;LOOP IF SELECTED
	110242	104406					TRAP	C#CLP1
11107	110244	012737	100010	110610	MOV	#100010,T28PK3		;SPACE FORWARD,ACK, COMMAND
11108	110252	012737	000005	110612	MOV	#5,T28RB		;NUMBER OF RECORDS TO SPACE FORW.
11109	110260	012704	110610		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
11110	110264	010465	000000		MOV	R4,TSD8(R5)		;ISSUE COMMAND
11111	110270	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET
11112	110274	016501	000002		MOV	TSSR(R5),R1		;GET TSSR CONTENTS
11113	110300	012702	100204		MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED
11114	110304	020102			CMP	R1,R2		;ARE THEY EQUAL
11115	110306	001406			BEQ	3104		;BR, IF OK



TSV7 - HARDWARE TESTS 1-8  
TEST 8: WRITE/READ TAPE MARK

MACRO M1113 14-JUN-84 14:17

SEQ 0317

11165	110462	110470	.WORD	T28DATA	; ADDRESS OF CHARACTERISTICS BLOCK
11166	110464	000000	.WORD	0	
11167	110466	000012	.WORD	10.	; STARTING VALUE OF BLOCK SIZE
11168	110470		T28DATA:		; CHARACTERISTICS DATA BLOCK
11169	110470	110502	.WORD	T28BFR	; ADDRESS OF MESSAGE BUFFER
11170	110472	000000	.WORD	0	
11171	110474	000024	.WORD	20.	; LENGTH OF MESSAGE BUFFER
11172	110476	000000	.WORD	0	
11173	110500	000000	T28DSW: .WORD	0	; SELECT DRIVE 0
11174	110502		T28BFR: .BLKW	25.	; MESSAGE BUFFER
11175			:		
11176			:		
11177			:		
11179	110564		:		
11181	110570		T28PK2: .BLKB	10-<.-TSV2&7>	
11182	110570	100006	.WORD	100006	; WRITE SUB SYS MEM COMMAND, IE AND ACK
11183	110572	110620	.WORD	T28BF2	; ADDRESS OF SELECT BLOCK DATA
11184	110574	000000	.WORD	0	
11185	110576	000006	.WORD	6.	; SIZE OF DATA PACKET
11186					
11188	110600		.BLKB	10-<.-TSV2&7>	
11190	110610		T28PK3: .WORD	100005	; REREAD COMMAND, AND ACK
11191	110610	100005	.WORD	100005	
11192	110612		T28RB: .WORD	FREE	; ADDRESS OF WRITE BUFFER
11193	110612	003114	.WORD	0	
11194	110614	000000	T28SZ: .WORD	0	; SIZE OF BUFFER (EXTENT)
11195	110616	000000	.EVEN		
11196			:		
11197			:		
11198			:		
11199			:		
11200	110620		T28BF2: .BYTE	10	; BSELC AREA
11201	110620	010	T28BS0: .BYTE	200	; BSEL1 AREA
11202	110621	200	T28BS1: .WORD	0	; SEL 2 AREA
11203	110622	000000	T28S2: .WORD	0	; DATA AREA
11204	110624	000000	T28S3: .WORD	0	
11205			:		
11206			:		
11207			.EVEN		
11208			:		
11209			:		
11210	110626		T28IMV: .WORD	101411	; ILLEGAL MODE BITS TEST DATA
11211	110626	101411	.WORD	102011	
11212	110630	102011	.WORD	103411	
11213	110632	103411	.WORD	177777	
11214	110634	177777	T28RN: .WORD	100011	; WRITE TAPE MARK COMMAND
11215	110636	100011	T28WDR: .WORD	100411	; ERASE COMMAND
11216	110640	100411	T28CON: .WORD	101011	; WRITE TAPE MARK RETRY
11217	110642	101011	.WORD	177777	; END OF DATA
11218	110644	177777	:		
11219			:		
11220			:		
11221	110646	000000	T28CNT: .WORD	0	; TAPE TIMER COUNTER STORAGE AREA
11222	110650	000000	T28CNU: .WORD	0	; TAPE TIMER COUNTER STORAGE AREA
11223	110652	000000	T28DLY: .WORD	0	; DELAY COUNTER
11224			.EVEN		
11225					

```

11226
11227
11228
11229
11230
11231 110654 124 141 160 T28RIB: .ASCIZ 'Tape Position Not Correct, RIB Should Be Set'
11232 110731 122 145 163 T28PBP: .ASCIZ 'Residual Byte Counter Register (RBPCR) Not Correct'
11233 111014 124 123 123 T28RDF: .ASCIZ 'TSSR Incorrect After READ REVERSE Into TAPE MARK'
11234 111075 124 123 123 T28RDG: .ASCIZ 'TSSR Incorrect After SPACE Command Into TAPE MARK'
11235 111157 124 123 123 T28WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
11236 111234 111 154 154 T28LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
11237 111315 127 122 111 T28SSR: .ASCIZ 'WRITE MISCELLANEOUS Command Not Accepted'
11238 111366 124 123 123 T28WDE: .ASCIZ 'TSSR Not Correct After READ DATA Command, Into TAPE MARK'
11239 111457 124 141 160 T28BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
11240 111524 124 123 123 T28TM: .ASCIZ 'TSSR Not Correct After FJRMAT Command Reject'
11241 111601 122 145 167 T28RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
11242 111650 104 162 151 T28OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
11243 111723 124 123 123 T28WDC: .ASCIZ 'TSSR Not Correct After WRITE TAPE MARK Command'
11244 112002 103 126 103 T28VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
11245 112055 124 115 113 T28TMK: .ASCIZ 'TMK Not Set After WRITE TAPE MARK Command'
11246 112127 124 115 113 T28RRM: .ASCIZ 'TMK Not Set After READ REVERSE Into TAPE MARK'
11247 112205 124 115 113 T28RRN: .ASCIZ 'TMK Not Set After SPACE REVERSE Into TAPE MARK'
11248 112264 124 115 113 T28RRP: .ASCIZ 'TMK Not Set After READ FORWARD Into TAPE MARK'
11249 112342 104 141 164 T28DTR: .ASCIZ 'Data Transferred On READ REVERSE Into A TAPE MARK'
11250 112424 104 141 164 T28DTA: .ASCIZ 'Data Compare Error, Data Read From Tape Not Equal To Written'
11251 112521 127 162 151 TST28ID: .ASCIZ 'Write/Read Tape Mark'
11252
11253
11254
11255
11256
11257
11258
11259
11260 112546
11261 112546
11262 112552 012701 110460
11263 112556 012721 100004
11264 112562 012721 110470
11265 112566 005021
11266 112570 012721 000012
11267 112574 012721 110502
11268 112600 005021
11269 112602 012721 000024
11270 112606 005021
11271 112610 012711 000000
11272 112614 012702 000030
11273 112620 012762 177777 110502 64:
11274 112626 005742
11275 112630 020227 000000
11276 112634 001371
11277 112636 000207
11278
11279
11280 112640
11281 112640
11282 112644 012701 110570

;*
;LOCAL TEXT MESSAGES FOR TEST
;-

T28REST:
        SAVREG
        MOV     @T28PACKET,R1
        MOV     @100004,(R1)+
        MOV     @T28DATA,(R1)+
        CLR     (R1)+
        MOV     @10.,(R1)+
        MOV     @T28BFR,(R1)+
        CLR     (R1)+
        MOV     @20.,(R1)+
        CLR     (R1)+
        MOV     @0,(R1)
        MOV     @24.,R2
        MOV     @177777,T28BFR(R2)
        TST     -(R2)
        CMP     R2,@0
        BNE     64$
        RTS     PC

;SAVE THE REGISTERS
;START OF THE PACKET
;WRITE SUBSYSTEM MEM. WITH ACK.
;ADDRESS OF CHARAISTICS DATA BLOCK
;EXTENDED ADDRESS
;SIZE OF DATA BLOCK IN BYTES
;ADDRESS OF MESSAGE BUFFER

;LENGTH OF MESSAGE BUFFER

;SELECT DRIVE ZERO
;NUMBER OF LOCATIONS TO BE CLEARED
;ALL ONES TO MESSAGE BUFFER
;NEXT LOCATION
;CHECK FOR END
;KEEP GOING UNTIL DONE
;RETURN

T28RT2:
        SAVREG
        MOV     @T28PK2,R1

;SAVE THE REGISTERS
;START OF THE PACKET

```



TSV7 - HARDWARE TESTS 1-8  
TEST 8: WRITE/READ TAPE MARK

MACRO M1113 14-JUN-84 14:17

SEQ 0319

11283	112650	012721	100006	MOV	#100006,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK,
11284	112654	012721	110620	MOV	#T28BF2,(R1)+	;ADDRESS OF DATA BLOCK
11285	112660	005021		CLR	(R1)+	;EXTENDED ADDRESS
11286	112662	012721	000006	MOV	#6.,(R1)+	;SIZE OF DATA BLOCK IN BYTES
11287	112666	005021		CLR	(R1)+	
11288	112670	012701	110620	MOV	#T28BF2,R1	;POINT TO DATA SEL AREA
11289	112674	005021		CLR	(R1)+	
11290	112676	005011		CLR	(R1)	
11291	112700	000207		RTS	PC	;RETURN
11292	112702					
11293	112702					
11294	112706	012701	110610	SAVREG		
11295	112712	005021		MOV	#T28PK3,R1	;GET PACKET ADDRESS
11296	112714	005021		CLR	(R1)+	;CLEAR COMMAND AREA
11297	112716	005021		CLR	(R1)+	;CLEAR ADDRESS AREA
11298	112720	005011		CLR	(R1)+	;CLEAR EXTENDED ADDRESS AREA
11299	112722	000207		CLR	(R1)	;SIZE OF DATA TRANSFER
11300	112724			RTS	PC	;RETURN
	112724			ENDTST		
	112724	104401				
11301	112726			ENDMOD		

L10130: TRAP C#ETST

```

1          .TITLE  TSV6 - PARAMETER CODING
7
12
18
19 112726  BGNMOD  TSV6
    112726  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 112726  BGNHRD
    112726  .WORD  L10134-L$HARD/2
    112730  L$HARD::
32
33 112730  GPRNA  HPM1,0,0,160010,177776,YES      ;GET TSBA/TSDB REGISTER ADDRESS.
    112730  .WORD  T$CODE
    112732  .WORD  HPM1
    112734  .WORD  T$LLOLIM
    112736  .WORD  T$HILIM
34 112740  GPRNA  HPM2,2,0,0,776,YES              ;GET VECTOR ADDRESS.
    112740  .WORD  T$CODE
    112742  .WORD  HPM2
    112744  .WORD  T$LLOLIM
    112746  .WORD  T$HILIM
35          ;GPRMD  HPM3,4,0,340,0,7,YES          ;GET INTERRUPT PRIORITY.
36 112750  ENDHRD
    112750  .EVEN
37 112750  104    105    126  HPM1:  .ASCIZ  'DEVICE ADDRESS (TSBA/TSDB) '
38 113004  111    116    124  HPM2:  .ASCIZ  'INTERRUPT VECTOR '
39 113030  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 113060                                BGNSFT
53 113060 000003                          .WORD L10135-L#SOFT/2
54 113062                                L#SOFT::
55                                     ; GPRML SPM1,0,-1,YES ; GET TRANSPORT TEST FLAG.
56 113062 001130                          ; GPRML SPM4,2,-1,YES ; GET ITERATION CONTROL.
57 113064 113120                          .WORD T#CODE
58 113066 177777                          .WORD SPM4
59                                     ; GPRMD SPM6,4,D,7777,0,7777,YES ; GET LOCAL ERROR LIMIT
60                                     ; GPRMD SPM7,6,D,7777,0,7777,YES ; GET GLOBAL ERROR LIMIT
61 113070                                ENDSFT
62                                     .EVEN
63 113070                                L10135:
64 113070 105 116 101 SPM1: .ASCIZ 'ENABLE TRANSPORT TESTS '
65 113120 111 116 110 SPM4: .ASCIZ 'INHIBIT ITERATIONS '
66 113150 120 105 122 SPM6: .ASCIZ 'PER TEST ERROR LIMIT '
67 113200 120 105 122 SPM7: .ASCIZ 'PER UNIT ERROR LIMIT '
68                                     .SBTTL PATCH AREA
69
70                                     ;
71                                     ; FINALLY A GENEROUS PATCH AREA.
72                                     ;
73                                     ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
74                                     ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
75                                     ;
76                                     ;
77 113230                                PATCH::
78                                     .BLKW 32.
79 113400 113400                          .=.!377*1
80 113400 000000                          LASTAD ;SET LAST USED ADDRESS.
81 113402 000000                          .EVEN
82 113404 000000                          .WORD 0
83 113404 000000                          .WORD 0
84 113404 000001                          L#LAST::
85 113404 000001                          ENDMOD
86 113404 000001                          .END

```

ADSSR	012216	G	C#AU	=	000052	DEVDR0	023422	FREE	003114	G	INCERK	017134					
ADR	=	000020	G	C#AUTO	=	000061	DEVNRD	023341	FREEM	003120	INTCPC	016240					
AMBTSS	006725		C#BRK	=	000022	DEVNXR	023257	FRESIZ	003116	G	INTFLA	016235					
ASSEMB	=	000010	C#BSEG	=	000004	DEVONL	023207	FUSI	004115		INTMAS	016234					
A1716	=	000003	C#BSUB	=	000002	DEVSUM	023152	F#AU	=	000015	INTR	016306	G				
BADDAT	003146	G	C#CEFG	=	000045	DFPTBL	002146	G	F#AUTO	=	000020	INTREC	002214	G			
BADSSR	015770	G	C#CLCK	=	000062	DIAGMC	=	000000	F#BGN	=	000040	INTVEC	016236				
BDVPCR	=	177520	G	C#CLEA	=	000012	DICEC	=	000001	F#CLEA	=	000007	INTX	004276			
BENBSW	002220	G	C#CLOS	=	000035	DSBINT	016274		F#DU	=	000016	INVERT	021232	G			
BIE	=	040000	C#CLP1	=	000006	DUAD12	004641		F#END	=	000041	IOKCKI	=	000200			
BIT0	=	000001	G	C#CVEC	=	000036	DUFLG	003102	G	F#HARD	=	000004	IOKSTP	=	000001		
BIT00	=	000001	G	C#DCLN	=	000044	DUMMY	003052		F#HW	=	000013	IPRI	002202	G		
BIT01	=	000002	G	C#DODU	=	000051	EF.CON	=	000036	G	F#INIT	=	000006	ISR	=	000100	G
BIT02	=	000004	G	C#DRPT	=	000024	EF.NEW	=	000035	G	F#JMP	=	000050	IVEC	002200	G	
BIT03	=	000010	G	C#DU	=	000053	EF.PWR	=	000034	G	F#MOD	=	000000	IXE	=	004000	G
BIT04	=	000020	G	C#EDIT	=	000003	EF.RES	=	000037	G	F#MSG	=	000011	I#AU	=	000041	
BIT05	=	000040	G	C#ERDF	=	000055	EF.STA	=	000040	G	F#PROT	=	000021	I#AUTO	=	000041	
BIT06	=	000100	G	C#ERHR	=	000056	EMAXDU	017067		F#PWR	=	000017	I#CLN	=	000041		
BIT07	=	000200	G	C#ERRO	=	000060	EN	=	000000	F#RPT	=	000012	I#DU	=	000041		
BIT08	=	000400	G	C#ERSF	=	000054	ENAIN	016242		F#SEG	=	000003	I#HRD	=	000041		
BIT09	=	001000	G	C#ERSO	=	000057	ENVIRN	020720		F#SOFT	=	000005	I#INIT	=	000041		
BIT1	=	000002	G	C#ESCA	=	000010	EPRTSW	002170	G	F#SRV	=	000010	I#MOD	=	000041		
BIT10	=	002000	G	C#ESEG	=	000005	EPRT1	006356		F#SUB	=	000002	I#MSG	=	000041		
BIT11	=	004000	G	C#ESUB	=	000003	EPRT2	006446		F#SW	=	000014	I#PROT	=	000040		
BIT12	=	010000	G	C#ETST	=	000001	ERCM	012023		F#TEST	=	000001	I#PTAB	=	000041		
BIT13	=	020000	G	C#EXIT	=	000032	ERRHI	002226	G	GDDAT	003150	G	I#PWR	=	000041		
BIT14	=	040000	G	C#GETB	=	000026	ERRK	017046		GERRMA	002164	G	I#RPT	=	000041		
BIT15	=	100000	G	C#GETW	=	000027	ERRLO	002230	G	GETPAT	020264	G	I#SEG	=	000041		
BIT2	=	000004	G	C#GMAN	=	000043	ERRNO	=	001513	GETSEL	020346	G	I#SETU	=	000041		
BIT3	=	000010	G	C#GPHR	=	000042	ERRVEC	=	000004	G	G#CNT0	=	000200	I#SFT	=	000041	
BIT4	=	000020	G	C#GPL0	=	000030	ERTABE	003366		G#DELM	=	000372	I#SRV	=	000041		
BIT5	=	000040	G	C#GPRI	=	000040	ERTABL	003166		G#DISP	=	000003	I#SUB	=	000041		
BIT6	=	000100	G	C#INIT	=	000011	ESUM	017050		G#EXCP	=	000400	I#TST	=	000041		
BIT7	=	000200	G	C#INLP	=	000020	EVL	=	000004	G	G#HILI	=	000002	J#JMP	=	000167	
BIT8	=	000400	G	C#MANI	=	000050	EXBCNT	=	000010	G	G#LOLI	=	000001	KIPAR0	=	172340	
BIT9	=	001000	G	C#MEM	=	000031	EXIT	034324		G#NO	=	000000	KIPAR1	=	172342		
BOE	=	000400	G	C#MSG	=	000023	EXPBRE	015572	G	G#OFFS	=	000400	KIPAR2	=	172344		
BRINIT	004455		C#OPEN	=	000034	EXPD	002222	G	G#OSI	=	000376	KIPAR3	=	172346			
BSELO	=	000000	C#PNTB	=	000014	EXPGOT	004531		G#PRMA	=	000001	KIPAR4	=	172350			
BSEL1	=	000001	C#PNTF	=	000017	EXPGT2	004565		G#PRMD	=	000002	KIPAR5	=	172352			
CHKAMB	016134		C#PNTS	=	000016	EXPHSG	002312	G	G#PRML	=	000000	KIPAR6	=	172354			
CHKMAN	020570	G	C#PNTX	=	000015	EXPREC	015564	G	G#RADA	=	000140	KIPAR7	=	172356			
CHKTSS	016426		C#QIO	=	000377	EXTA	005770		G#RADB	=	000000	KIPDR0	=	172300			
CKDROP	017272		C#RDBU	=	000007	EXTEND	005766		G#RADD	=	000040	KIPDR1	=	172302			
CKEMAX	017172		C#REFG	=	000047	EXTFEA	002216	G	G#RADL	=	000120	KIPDR2	=	172304			
CKMSG	011450	G	C#RESE	=	000033	E#END	=	002100	G#RADO	=	000020	KIPDR3	=	172306			
CKMSG2	011570	G	C#REVI	=	000003	E#LOAD	=	000035	G#XFER	=	000004	KIPDR4	=	172310			
CKRAM	011204	G	C#FLA	=	000021	FATAL	034424		G#YES	=	000010	KIPDR5	=	172312			
CKRAM2	011314	G	C#RPT	=	000025	FATERR	=	000060	HIADDR	=	001400	KIPDR6	=	172314			
CHDPKT	021304	G	C#SEFG	=	000046	FATFLG	002212	G	HOE	=	100000	G	KIPDR7	=	172316		
CHPMEM	017750		C#SPRI	=	000041	FERCM	012012		HPM1	112750		KTENAB	003124	G			
CONFIG	017340		C#SVEC	=	000037	FIFEXP	012260	G	HPM2	113004		KTFLG	003122	G			
COUNT	002300	G	C#TPRI	=	000013	FIF1MS	012332		HPM3	113030		KTINIT	021100				
CSRADD	002176	G	DATA	002302	G	FIF2MS	012401		IBE	=	010000	G	KTOFF	017364			
CTAB	003154	G	DATASC	020322		FILLME	017512		IDU	=	000040	G	KTON	017346			
CTABE	003166	G	DEBUGM	011722		FNOINT	004213		IER	=	020000	G	LERRMA	002162	G		
CTABM	003154	G	DEVcnt	002210	G	FORCER	002166	G	IFALT	004254		LISTAL	=	000001			

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

SEQ 0323

L\$OE = 040000 G	L\$UNIT 002012 G	L10071 055642	M8189 005643	PRBEXP 015560
LOOPCN 002206 G	L10000 002154	L10072 047704	NBA = 002000	PRBMSG 015426
LOOPCO 013216	L10001 002166	L10073 050304	NEWPAS 022034	PRBREC 015562
LOOPFL 003152 G	L10002 005764	L10074 050760	NODEV 003104 G	PRBTOT 015513
LOT = 000010 G	L10003 012134	L10075 051424	NOINIT 004333	PRBYTE 015212 G
L\$ACP 002110 G	L10004 012152	L10076 052164	NOINTR 004217	PRI = 002000 G
L\$APT 002036 G	L10005 012170	L10077 053124	NOITS 002160 G	PRIADD 010250
L\$AU 022376 G	L10006 012176	L10100 053444	NOMAN 020624	PRIAO 010320
L\$AUT 002070 G	L10007 012214	L10101 054046	NOMEM 005456	PRIBXO 007702 G
L\$AUTO 022602 G	L10010 012232	L10102 075224	NP.IR = 000200	PRIEQU 010150
L\$CCP 002106 G	L10011 012256	L10103 056604	NP.LOO= 000040	PRIPKT 007460 G
L\$CLEA 022662 G	L10012 012330	L10104 057452	NP.OUT= 000100	PRIRAM 010156
L\$CO 002032 G	L10013 012500	L10105 060344	NP.WRP= 000020	PRITAD 010364
L\$DEPO 002011 G	L10014 013214	L10106 061272	NSI 004150	PRITSS 006022
L\$DESC 003400 G	L10015 014042	L10107 062050	NSINIT 004405	PRITO 010446
L\$DESP 002076 G	L10016 014064	L10110 062712	NUL 004525	PRIT1 010511
L\$DEVP 002060 G	L10017 015570	L10111 063564	NULCR 004526	PRIXOR 010032 G
L\$DISP 002124 G	L10020 015576	L10112 064436	NXM = 004000	PRI00 = 000000 G
L\$DLY 002116 G	L10021 015604	L10113 065312	NXMFLG 003126 G	PRI01 = 000040 G
L\$DTP 002040 G	L10022 015616	L10114 066166	NXMI 003132 G	PRI02 = 000100 G
L\$DTYP 002034 G	L10023 015640	L10115 067036	NXML0 003130 G	PRI03 = 000140 G
L\$DU 022474 G	L10024 015666	L10116 067770	NXMTST 021476	PRI04 = 000200 G
L\$DUT 002072 G	L10025 016026	L10117 071020	NXR 003736	PRI05 = 000240 G
L\$DVTY 003372 G	L10026 016336	L10120 071400	NXRERR 005734 G	PRI06 = 000300 G
L\$EF 002052 G	L10030 022326	L10121 072054	NXR 003775	PRI07 = 000340 G
L\$ENVI 002044 G	L10031 022472	L10122 105262	NXTU 022046	PRMESS 014332
L\$ETP 002102 G	L10032 022600	L10123 075646	OFL = 000100	PRMNO 002310 G
L\$EXP1 002046 G	L10033 022660	L10124 076430	ONEFIL= 000000	PRMSGE 014642 G
L\$EXP4 002064 G	L10034 022706	L10125 077252	O\$APTS= 000000	PRMSG0 015022
L\$EXP5 002066 G	L10035 023150	L10126 100154	O\$AU = 000001	PRMSG1 015067
L\$HARD 112730 G	L10036 024610	L10127 101704	O\$BGNR= 000001	PRMSG2 015125
L\$HIME 002120 G	L10037 027270	L10130 112724	O\$BGNS= 000001	PROASC 014510
L\$HPCP 002016 G	L10040 025216	L10131 105662	O\$DU = 000001	PR1ASC 014555
L\$HPTP 002022 G	L10041 025540	L10132 106142	O\$ERRT= 000000	PST32W 003142 G
L\$HM 002146 G	L10042 026120	L10133 110414	O\$GNSW= 000001	PUNIT 022330
L\$ICP 002104 G	L10043 034450	L10134 112750	O\$POIN= 000001	PW.D11= 000021
L\$INIT 021602 G	L10044 027676	L10135 113070	O\$SETU= 000000	PW.D13= 000022
L\$LADP 002026 G	L10045 030546	MEMADD 014044 G	PASRPT 022100	PW.D22= 000020
L\$LAST 113404 G	L10046 031366	MEMCK 021322 G	PATCH 113230 G	PW.NOP= 000000
L\$LOAD 002100 G	L10047 031602	MENASC 020537	PATDAT 020320	PW.NO1= 000023
L\$LUN 002074 G	L10050 032150	MENERR 020464	PC.ERA= 002400	PW.RDE= 000024
L\$MREV 002050 G	L10051 032514	MENRES 020566	PC.IER= 002000	PW.RDR= 000001
L\$NAME 002000 G	L10052 046714	MMVEC = 000250	PC.NOO= 001000	PW.RDS= 000005
L\$PRIO 002042 G	L10053 035122	MSA.FR= 000006	PC.REL= 000000	PW.RFI= 000003
L\$PROT 021572 G	L10054 035702	MSA.NO= 000000	PC.REW= 000400	PW.WCT= 000006
L\$PRT 002112 G	L10055 036456	MSA.NR= 000004	PKBCNT= 000006	PW.WFI= 000004
L\$REPP 002062 G	L10056 037160	MSA.VO= 000002	PKHI = 000004	PW.WFM= 000007
L\$REV 002010 G	L10057 037624	MSGEXP 012234 G	PKLOW = 000002	PW.WMI= 000010
L\$RPT 022710 G	L10060 040260	MSGLOO 013154 G	PKTADD 007644	PW.WNP= 000011
L\$SOFT 113062 G	L10061 040714	MSGSTA 012440 G	PKTFRM 007606	PW.WTR= 000002
L\$SPC 002056 G	L10062 041306	MSGSUB 014032 G	PKTGET 012154 G	P.ACK = 100000
L\$SPCP 002020 G	L10063 042010	MS.ATT= 000006	PKTMES 012200 G	P.CMD = 000037
L\$SPTP 002024 G	L10064 042254	MS.EXT= 000200	PKTRAM 004743 G	P.CONT= 000012
L\$STA 002030 G	L10065 042526	MS.RSD= 000001	PKTSSR 012136 G	P.CVC = 040000
L\$SW 002156 G	L10066 043012	MS.RSF= 000020	PNT = 001000 G	P.FMT = 000140
L\$TEST 002114 G	L10067 043312	MS.RST= 000010	PRAMPK 014066	P.FORM= 000011
L\$TIML 002014 G	L10070 043776	M8186 005552	PRASC 014613	P.GETS= 000017

P.IE = 000200	SPM6 113150	TSREJ = 000006	T##CLE = 010034	T22WRT 026310
P.INIT = 000013	SPM7 113200	TSSDEF 006676	T##DU = 010032	T23A 003134 G
P.MODE = 007400	SR0 = 177572	TSSR = 000002 G	T##HAR = 010134	T23AM3 033340
P.OPP = 020000	SR1 = 177574	TSSRBI 003500 G	T##HW = 010000	T23B 003136 G
P.POSI = 000010	SR2 = 177576	TSSRFO 006505	T##INI = 010030	T23BA 033725
P.READ = 000001	SR3 = 172516	TSSRH = 000003 G	T##MSG = 010025	T23BFR 032602
P.SWB = 010000	SSR = 000200	TSSX 004016	T##PRO = 010027	T23BF2 032722
P.WRIT = 000005	STATCO 012502	TSTBLK 002742 G	T##RPT = 010035	T23BS0 032722
P.WRTC = 000004	SVCGBL = 000000	TSTCNT 002204 G	T##SOF = 010135	T23BS1 032723
P.WRTS = 000006	SVCINS = 000000	TSTEND 017010	T##SRV = 010026	T23CHK 034262
QVP 002174 G	SVCSUB = 000001	TSTFLA 002304 G	T##SUB = 010133	T23CON 032740
RAMASC 014246	SVCTAG = 000000	TSTL00 016546 G	T##SW = 010001	T23DAT 032570
RAMDAT 002232 G	S'CTST = 000001	TSTPTR 002306 G	T##TES = 010130	T23DSW 032600
RAMERR 015600 G	S'LSYM = 010000	TSTSET 016600 G	T1 023472 G	T23E0T 033064
RAMEXP 015620 G	SO.IDB = 000010	TST21I 024434	T2 024612 G	T23ET 032777
RAMFOR 010206	SO.IFB = 000002	TST22I 027077	T2.1 024642	T23L00 027342
RAMSIZ 002272 G	SO.IFP = 000001	TST23I 034066	T2.2 025234	T23OFL 033406
RAMTAD 015606 G	SO.ILD = 000020	TST24I 046462	T2.3 025556	T23PAC 032560
RCVHIA 002274 G	SO.ION = 000040	TST25I 055440	T21AM3 024313	T23PK2 032670
RCVLOA 002276 G	SO.IRD = 000100	TST26I 075027	T21BFR 024114	T23PK3 032710
RDERR 005204	SO.IRW = 000004	TST27I 105063	T21BF2 024210	T23RES 034102
RECMG 002456 G	SO.ISP = 000200	TST28I 112521	T21BS0 024210	T23RNC 033265
RECV 002224 G	S1.ICE = 002000	TSV2 002000 G	T21BS1 024211	T23RSZ 032720
REGSAV 020230	S1.IEO = 010000	TSV3 002166 G	T21DAT 024100	T23RT2 034174
RETErr 005370	S1.IF.1 = 001000	TSV4 021572 G	T21DLY 024112	T23RT3 034236
RETRY 034326	S1.IHE = 000400	TSV6 112726 G	T21DSW 024110	T23RWN 033216
REWIND 011104 G	S1.IID = 004000	TSV7 023472 G	T21L00 023522	T23SSR 032744
RMCHBE = 000167	S1.IIR = 020000	TTIBFR = 177562 G	T21OFL 024413	T23SZ 032716
RMCHEN = 000200	S1.IIR = 020000	TTICSR = 177560 G	T21PAC 024070	T23S2 032724
RMMSGB = 000215	S1.I2R = 040000	TTIVEC = 000060 G	T21PK2 024200	T23S3 032726
RMMSGE = 000234	S1.PAR = 100000	T#ARGC = 000003	T21RES 024456	T23TM 033142
RMPKTB = 000201	S2.ATI = 000010	T#CODE = 001130	T21RT2 024546	T23TMP 032730
RMPKTE = 000210	S2.BTI = 000004	T#ERRN = 001513	T21SSR 024216	T23VCK 033652
RMR = 010000	S2.DIM = 000200	T#EXCP = 000000	T21S2 024212	T23WB 032712
RMPACK 011200	S2.ILW = 000100	T#FLAG = 000040	T21S3 024214	T23WD 032734
SC = 100000	S2.INR = 000020	T#GMAN = 000000	T22AM3 026415	T23WDC 033550
SCE = 020000	S2.OUT = 000040	T#HILI = 000776	T22BFR 026202	T23WDD 033461
SC'ERR 005276	S2.UND = 000003	T#LAST = 000001	T22BF2 026300	T23WDR 032736
SC'IE 005011	TBLEND = 003052 G	T#LOI.I = 000000	T22BS0 026300	T23WRT 032732
SDELAY 010750	TCOASC 006566	T#LSYM = 010000	T22BS1 026301	T23WSS 033777
SELASC 020532	TCOCOD 006766	T#LTND = 000010	T22DAT 026170	T24AM3 045450
SELDAT = 000004	TEMP1 003106 G	T#NESI = 177777	T22FOR 026314	T24BA 046002
SEL2 = 000002	TEMP2 003110 G	T#NS0 = 000000	T22L00 024642	T24BFR 044062
SETMAP 017406	TERCLS = 000016	T#NS1 = 000005	T22OFL 026515	T24BF2 044200
SETU 022132	TESTND = 000010	T#NS2 = 000002	T22PAC 026160	T24BOT 045043
SFFMSG 012172 G	TEXASC 006525	T#PTAU = 000000	T22PK2 026270	T24BS0 044200
SFHERR 003703	TFCASC 006627	T#SAVL = 177777	T22POS 026312	T24BS1 044201
SFIERR 003650	TIMEXP 015642 G	T#SEGL = 177777	T22RD 026306	T24CON 044212
SFIMSG 012124 G	TIMSGO 015670	T#SUBN = 000003	T22RES 027132	T24DAT 044050
SFPTBL 002156 G	TINERR 012111	T#TAGL = 177777	T22RT2 027224	T24DLY 044216
SIFLAG 003144 G	TMPBFR 002622 G	T#TAGN = 010136	T22RWJ 026664	T24DSW 044060
SIMSG 012056	TNAM 016774	T#TEMP = 000000	T22SSR 026320	T24DTA 045110
SKIPT 003370	TRANST 002156 G	T#TEST = 000010	T22S2 026302	T24E0T 045176
SOFINI 016064 G	TSBA = 000000 G	T#TSTM = 177777	T22S3 026304	T24ILA 044572
SPACE 010556 G	TSBAH = 000001 G	T#TSTS = 000001	T22TM 026570	T24LON 046142
SPM1 113070	TSDB = 000000 G	T#AU = 010031	T22VCK 026737	T24L00 034516
SPM4 113120	TSDBH = 000001 G	T#AUT = 010033	T22WLK 027012	T24LOP 046224
	TSFCOD 007326			

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

SEQ 0325

T24LOQ	044656	T25SSR	054274	T26WDC	074140	T27WDR	102120	T4	034452 G
T24LOR	044272	T25SZ	054246	T26WDD	074050	T27WNG	102134	T4.1	034516
T24NEF	044220	T25S2	054252	T26WDE	073243	T27WRF	104706	T4.10	042026
T24NOM	044431	T25S3	054254	T26WDF	073051	T27WSS	104042	T4.11	042272
T24OFL	045515	T25TM	054502	T26WNG	072306	T28BFR	110502	T4.12	042544
T24PAC	044040	T25WB	054242	T26WSS	074341	T28BF2	110620	T4.13	043030
T24PBP	046306	T25WDC	055367	T27AM3	103427	T28BOT	111457	T4.14	043330
T24PK2	044150	T25WDE	054355	T27BA	103767	T28BS0	110620	T4.2	035140
T24PK3	044170	T25WDR	054260	T27BFR	101772	T28BS1	110621	T4.3	035720
T24RB	044172	T25WNG	054645	T27BF2	102110	T28CNT	110646	T4.4	036474
T24RES	046530	T25WNH	055020	T27BOT	103001	T28CNU	110650	T4.5	037176
T24RN	044206	T26AM3	073726	T27BS0	102110	T28CON	110642	T4.6	037642
T24RNC	045375	T26BA	074266	T27BS1	102111	T28DAT	110470	T4.7	040276
T24RT2	046622	T26BFR	072142	T27CNT	102126	T28DLY	110652	T4.8	040732
T24RT3	046664	T26BF2	072260	T27CNU	102130	T28DSW	110500	T4.9	041324
T24RMN	045326	T26BOT	073315	T27CON	102122	T28DTA	112424	T5	046716 G
T24SSR	044737	T26BS0	072260	T27DAT	101760	T28DTR	112342	T5.1	046746
T24SZ	044176	T26BS1	072261	T27DLY	102132	T28IMV	110626	T5.2	047722
T24S2	044202	T26CNT	072276	T27DSW	101770	T28LOO	105320	T5.3	050322
T24S3	044204	T26CNU	072300	T27DTA	104766	T28LOQ	111234	T5.4	050776
T24TH	045253	T26DAT	072130	T27EOT	103151	T28OFL	111650	T5.5	051442
T24TRL	046374	T26DLY	072304	T27LON	104131	T28PAC	110460	T5.6	052202
T24VCK	045727	T26DSW	072140	T27LOO	075266	T28PBP	110731	T5.7	053142
T24WB	044172	T26DTA	073362	T27LOP	104213	T28PK2	110570	T5.8	053462
T24WDC	045656	T26EOT	073450	T27LOQ	102575	T28PK3	110610	T6	055644 G
T24WDD	045570	T26LON	074430	T27LOR	102450	T28RB	110612	T6.1	055710
T24WDE	044771	T26LOO	055710	T27NEF	104451	T28RDF	111014	T6.10	065330
T24WDF	044515	T26LOP	074512	T27OFL	103476	T28RDG	111075	T6.11	066204
T24WDG	044342	T26LOQ	073126	T27PAC	101750	T28RES	112546	T6.12	067054
T24WDR	044210	T26LOR	073001	T27PBP	104275	T28RIB	110654	T6.13	070006
T24WSS	046053	T26NEF	072374	T27PK2	102060	T28RN	110636	T6.14	071036
T25BFR	054132	T26NEQ	074750	T27PK3	102100	T28RRM	112127	T6.15	071416
T25BF2	054250	T26OFL	073775	T27RB	102102	T28RRN	112205	T6.2	056622
T25BNC	054730	T26PAC	072120	T27RDF	102222	T28RRP	112264	T6.3	057470
T25BOT	054435	T26PBP	074574	T27RES	105104	T28RT2	112640	T6.4	060362
T25BS0	054250	T26PK2	072230	T27RN	102116	T28RT3	112702	T6.5	061310
T25BS1	054251	T26PK3	072250	T27RNC	103354	T28RMN	111601	T6.6	062066
T25CNT	054270	T26RB	072252	T27RRF	102271	T28SSR	111315	T6.7	062730
T25CNU	054266	T26RDF	072456	T27RT2	105176	T28SZ	110616	T6.8	063602
T25CON	054262	T26RES	075040	T27RT3	105240	T28S2	110622	T6.9	064454
T25DAT	054120	T26RN	072266	T27RMN	103305	T28S3	110624	T7	075226 G
T25DLY	054272	T26RNC	073653	T27SC	102366	T28TH	111524	T7.1	075266
T25DSW	054130	T26RRF	072525	T27SCF	104547	T28THK	112055	T7.2	075664
T25LOO	046746	T26RRG	072622	T27SSR	102656	T28VCK	112002	T7.3	076446
T25NEF	055103	T26RSZ	072302	T27SZ	102106	T28WB	110612	T7.4	077270
T25NET	054571	T26RT2	075132	T27S2	102112	T28WDC	111723	T7.5	100172
T25OFL	055314	T26RT3	075174	T27S3	102114	T28WDE	111366	T8	105264 G
T25PAC	054110	T26RMN	073604	T27TIM	103074	T28WDF	111157	T8.1	105320
T25PK2	054220	T26SC	072717	T27TM	103230	T28WDR	110640	T8.2	105700
T25PK3	054240	T26SSR	073207	T27TRL	104363	T3	027272 G	T8.3	106160
T25RB	054242	T26SZ	072256	T27TSA	104624	T3BFLG	003140 G	UAM	= 000200 G
T25RES	055456	T26S2	072262	T27VCK	103714	T3.1	027342	UNITN	= 002172 G
T25RIB	055163	T26S3	072264	T27WB	102102	T3.2	027714	UNREC	= 000006
T25RN	054256	T26TM	073527	T27WDC	103641	T3.3	030564	USI	004121
T25RT2	055550	T26TRL	074662	T27WDD	103551	T3.4	031404	WAITF	016340 G
T25RT3	055612	T26VCK	074213	T27WDE	102712	T3.5	031620	WC. IFA	= 000200
T25RMN	055245	T26WB	072252	T27WDF	102520	T3.6	032166	WC. IFE	= 000002

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

SEQ 0326

WC.IGO= 000001	WRTERR 005111	XSOILA= 000400	X#FALS= 000040	X2.UNI= 000007
WC.IRE= 000010	WRTMSG 005054	XSOILC= 001000	X#OFFS= 000400	X2.WCF= 002000
WC.IRW= 000004	WSMBK 021314 G	XSOLET= 020000	X#TRUE= 000020	X3.DCK= 000010
WC.IOT= 000100	XFERAS 016030	XSOMOT= 000200	X1.COR= 020000	X3.MBZ= 000006
WC.IIT= 000040	XNXM 016466	XSONEF= 002000	X1.DLT= 100000	X3.MDE= 177400
WC.ISR= 000020	XORBFO 007764	XSOONL= 000100	X1.MBZ= 017375	X3.OPI= 000100
WF.IED= 000010	XORFOR 010102	XSOPED= 000010	X1.RBP= 000400	X3.REV= 000040
WF.IER= 000004	XST0 = 000006 G	XSORLL= 010000	X1.SPA= 040000	X3.RIB= 000001
WF.IHI= 000200	XST1 = 000010 G	XSORLS= 040000	X1.UNC= 000002	X3.SPA= 000200
WF.IRE= 000040	XST2 = 000012 G	XSOTMK= 100000	X2.BUF= 000100	X3.TRF= 000020
WF.IWF= 000020	XST3 = 000014 G	XSOVCK= 000020	X2.EXT= 000200	X4.HSP= 100000
WF.IWR= 000100	XST4 = 000016 G	XSOWLE= 004000	X2.OPM= 100000	X4.MBZ= 017400
WF.I3R= 000002	XSOBOT= 000002	XSOWLK= 000004	X2.RCE= 040000	X4.RCE= 040000
WF.I4R= 000001	XSOEOT= 000001	XXCOMM 003112 G	X2.REV= 000077	X4.TSM= 020000
WRTCHR 010752 G	XSOIE = 000040	X#ALWA= 000000	X2.SPA= 035400	X4.WRC= 000377

. ABS. 113404 000  
 000000 001  
 ABS 000000 002  
 ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 31592 WORDS ( 124 PAGES)  
 DYNAMIC MEMORY: 20614 WORDS ( 79 PAGES)  
 ELAPSED TIME: 00:58:44  
 CVTSCB,CVTSCB/-SP=SVC/ML,TSV1C,TSV22C,TSV3B,TSV4,TSV7A,TSV6