

TSV05

TSV05 CONTROL PART 3  
CVTSCC0

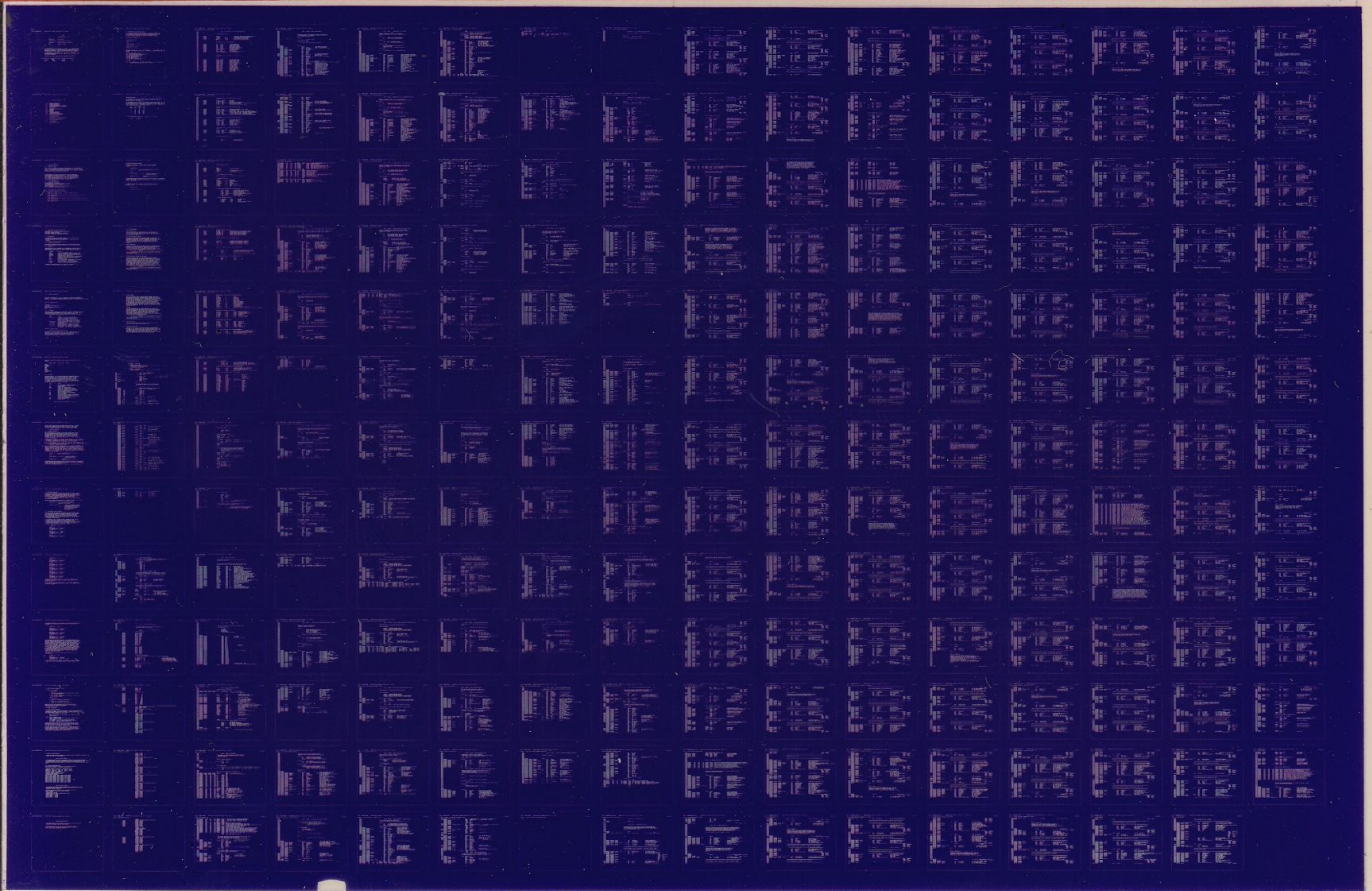
AH-T098C-MC

1 OF 2 JAN 1986

COPYRIGHT © 1982-85

**digital**

MADE IN USA





TSV05

TSV05 CONTROL PART 3  
CVTSCC0

AH-T098C-MC

2 OF 2 JAN 1986

COPYRIGHT © 1982-85

**digital**

MADE IN USA

The image displays a grid of 100 small diagrams or tables, arranged in 10 rows and 10 columns. Each cell contains a small schematic or data table, likely related to the control part of the TSV05 system. The diagrams are densely packed and appear to be technical drawings or data tables. The overall layout is a structured grid of these small components.



.REM\_  
IDENTIFICATION

PRODUCT ID: AC-T097C-MC  
PRODUCT TITLE: CVTSCCO TSV05 CTRL PART 3  
DEPARTMENT: COMPUTER SPECIAL SYSTEMS/PGG  
DATE: AUGUST 23, 1985

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) 1983, 1985 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 (HSAAS.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 HIERARCY 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 SUCESSFULLY 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 "DDDD".

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

EXAMPLE OF SWITCH USAGE:

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

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



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

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
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 (0) ? 8<CR>

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

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

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

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

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

# UNITS (0) ? 8<CR>

UNIT 1  
CSR ADDRESS (0) ? 160000<CR>  
SUB-DEVICE # (0) ? 0-7<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 SOHZ (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 (D) ? 1
UNIT 0
DEVICE ADDRESS (0) 172520 ? <CR>
VECTOR (0) 224 ? <CR>
CHANGE SW (L) ? N<CR>
```

THE ABOVE COMMAND WILL START THE DIAGNOSTIC. THE COMMAND HAS 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 REPO 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) CONRTOL 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.

REVISION C - JUNE 1985

UPDATED TO PERFORM CORRECTLY WITH XXDP V2.1 EXTENDED  
 MONITOR (DRSXM).

```

840          .TITLE   TSV2 - PROGRAM HEADER
841          .SBTTL   PROGRAM HEADER
842 000000    .PSECT   ABS
843
849          .MCALL   SVC
850 000000    SVC           ; INITIALIZE SUPERVISOR MACROS
851          .ENABLE  LC
852          .NLIST   BEX,CND
858 000000    .ENABL   ABS,AMA
859          .=2000
860 002000    BGNMOD   TSV2
      002000
861          TSV2::
862          ;**
863          ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
864          ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
865          ;--
866
867
868          POINTER BGNSW,BGNSFT,BGNAU,BGNDU,BGNRPT
869 002000    HEADER   CVTSC,C,0,655,.0
      002000    L$NAME:: ;DIAGNOSTIC NAME
      002000          .ASCII /C/
      002001          .ASCII /V/
      002002          .ASCII /I/
      002003          .ASCII /S/
      002004          .ASCII /C/
      002005          .BYTE  0
      002006          .BYTE  0
      002007          .BYTE  0
      002010          L$REV:: ;REVISION LEVEL
      002010          .ASCII /C/
      002011          L$DEPO:: ;0
      002011          .ASCII /0/
      002012          L$UNIT:: ;NUMBER OF UNITS
      002012 000000    .WORD  0
      002014          L$TIML:: ;LONGEST TEST TIME
      002014 001217    .WORD  655.
      002016          L$HPCP:: ;POINTER TO H.W. QUES.
      002016 113610    .WORD  L$HARD
      002020          L$SPCP:: ;POINTER TO S.W. QUES.
    
```

PROGRAM HEADER

002020	113742				
002022		L#HPTP::	.WORD	L#SOFT	
002022	002146				;PTR. TO DEF. H.W. PTABLE
002024		L#SPTP::	.WORD	L#HW	
002024	002156				;PTR. TO S.W. PTABLE
002026		L#LADP::	.WORD	L#SW	
002026	114404				;DIAG. END ADDRESS
002030		L#STA::	.WORD	L#LAST	
002030	000000				;RESERVED FOR APT STATS
002032		L#CO::	.WORD	0	
002032	000000				
002034		L#DTYP::	.WORD	0	;DIAGNOSTIC TYPE
002034	000000				
002036		L#APT::	.WORD	0	;APT EXPANSION
002036	000000				
002040		L#DTP::	.WORD	0	;PTR. TO DISPATCH TABLE
002040	002124			L#DISPATCH	
002042		L#PRIO::	.WORD	0	;DIAGNOSTIC RUN PRIORITY
002042	C00000				
002044		L#ENVI::	.WORD	0	;FLAGS DESCRIBE HOW IT WAS SETUP
002044	000000				
002046		L#EXP1::	.WORD	0	;EXPANSION WORD
002046	000000				
002050		L#MREV::	.WORD	0	;SVC REV AND EDIT #
002050	003		.BYTE	C#REVISION	
002051	003		.BYTE	C#EDIT	
002052		L#EF::	.WORD	0	;DIAG. EVENT FLAGS
002052	000000				
002054	000000		.WORD	0	
002056		L#SPC::	.WORD	0	
002056	000000				
002060		L#DEVP::	.WORD	0	; POINTER TO DEVICE TYPE LIST
002060	003372			L#DVTYP	
002062		L#REPP::	.WORD	0	;PTR. TO REPORT CODE
002062	023574			L#RPT	
002064		L#EXP4::	.WORD	0	
002064	000000				
002066		L#EXP5::	.WORD	0	
002066	000000				
002070		L#AUT::	.WORD	0	;PTR. TO ADD UNIT CODE
002070	023262			L#AU	
002072		L#DUT::	.WORD	0	;PTR. TO DROP UNIT CODE
002072	023360			L#DU	
002074		L#LUN::	.WORD	0	;LUN FOR EXERCISERS TO FILL
002074	000000				
002076		L#DESP::	.WORD	0	;POINTER TO DIAG. DESCRIPTION
002076	003400			L#DESC	
002100		L#LOAD::	EMT	E#LOAD	;GENERATE SPECIAL AUTOLOAD EMT
002100	104035				
002102		L#ETP::	.WORD	0	;POINTER TO ERR TBL
002102	000000				
002104		L#ICP::	.WORD	0	;PTR. TO INIT CODE
002104	022466			L#INIT	
002106		L#CCP::	.WORD	0	;PTR. TO CLEAN-UP CODE
002106	023546			L#CLEAN	
002110		L#ACP::	.WORD	0	;PTR. TO AUTO CODE
002110	023466			L#AUTO	



PROGRAM HEADER

002112		L\$PRT::		;PTR. TO PROTECT TABLE
002112	022456	.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	

870

DISPATCH TABLE

```

872
873
874
875
876
877 002122
      002122 000010
      002124
      002124 024356
      002126 025472
      002130 030152
      002132 035332
      002134 047576
      002136 056524
      002140 076106
      002142 106144

878
879
880
881
882
883
884
885 002144
      002144 000003
      002146
      002146
886 002146 172520
887 002150 000224
888 002152 000200
889 002154
      002154

890
891
892
893
894
895
896 002154
      002154 000004
      002156
      002156
897 002156 000000
898 002160 000000
899
900
901 002162 000017
902 002164 000310
903 002166
      002166
904 002166
905
914
915
920
926
927 002166

      .SBTTL DISPATCH TABLE
      ;**
      ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
      ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
      ;--
      DISPATCH 8
      .WORD 8
L$DISPATCH::
      .WORD T1
      .WORD T2
      .WORD T3
      .WORD T4
      .WORD T5
      .WORD T6
      .WORD T7
      .WORD T8

      .SBTTL DEFAULT HARDWARE P-TABLE
      ;**
      ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
      ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
      ; IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
      ;--
      BGNHW DFPTBL ;DEFAULT HARD-P-TABLE
      .WORD L10000-L$HW/2
L$HW::
DFPTBL::
      .WORD 172520 ; 1ST (OF 2) REGISTERS.
      .WORD 224 ; INTERRUPT VECTOR
      .WORD PRI04 ; INTERRUPT PRIORITY.
      ENDSW
L10000:

      .SBTTL SOFTWARE P-TABLE
      ;**
      ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
      ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
      ;--
      BGNSW SFPTBL
      .WORD L10001-L$SW/2
L$SW::
SFPTBL::
TRANSTST:: .WORD 0 ; ENABLE TEST OF TRANSPORT(S) IF =1
NOITS:: .WORD 0 ; INHIBIT ITERATION OPTION.
; ... 0 = ITERATE.
; ...NZ = INHIBIT ITERATE.
LERRMAX:: .WORD 15. ; LOCAL (PER TEST) ERROR LIMIT
GERRMAX:: .WORD 200. ; GLOBAL (PER UNIT) ERROR LIMIT
      ENDSW
L10001:
      ENDMOD

      .TITLE TSV3 - GLOBAL AREAS
      .SBTTL GLOBAL EQUATES SECTION

      BGNMOD TSV3

```

## GLOBAL EQUATES SECTION

002166  
 928  
 929  
 930  
 931  
 932  
 933  
 934  
 935  
 939 002166

TSV3::

.SBTTL GLOBAL EQUATES SECTION

```

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

```

EQUALS ; GET STANDARD EQUATES.

; BIT DIFINITIONS

```

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

```

```

;
BIT9== BIT09
BIT8== BIT08
BIT7== BIT07
BIT6== BIT06
BIT5== BIT05
BIT4== BIT04
BIT3== BIT03
BIT2== BIT02
BIT1== BIT01
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

```

## GLOBAL EQUATES SECTION

```

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

```

```

940
941 002166

```

```

000250

```

```

177572
177574
177576
172516

```

```

          KT11
.SBTL. MEMORY MANAGEMENT DEFINITIONS
;*KT11 VECTOR ADDRESS
MMVEC= 250
;*KT11 STATUS REGISTER ADDRESSES
SR0= 177572
SR1= 177574
SR2= 177576
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

```

```

;DEFINE MEMORY MANAGEMENT REGISTERS

```

## MEMORY MANAGEMENT DEFINITIONS

```
UIPAR4= 177650
UIPAR5= 177652
UIPAR6= 177654
UIPAR7= 177656
  .IF NB
  ;*USER "D" PAGE ADDRESS REGISTERS
  UDPAR0= 177660
  UDPAR1= 177662
  UDPAR2= 177664
  UDPAR3= 177666
  UDPAR4= 177670
  UDPAR5= 177672
  UDPAR6= 177674
  UDPAR7= 177676
  .ENDC
  .IF NB
  ;*SUPERVISOR "I" PAGE DESCRIPTOR REGISTERS
  SIPDR0= 172200
  SIPDR1= 172202
  SIPDR2= 172204
  SIPDR3= 172206
  SIPDR4= 172210
  SIPDR5= 172212
  SIPDR6= 172214
  SIPDR7= 172216
  .IF NB
  ;*SUPERVISOR "D" PAGE DESCRIPTOR REGISTERS
  SDPDR0= 172220
  SDPDR1= 172222
  SDPDR2= 172224
  SDPDR3= 172226
  SDPDR4= 172230
  SDPDR5= 172232
  SDPDR6= 172234
  SDPDR7= 172236
  .ENDC
  ;*SUPERVISOR "I" PAGE ADDRESS REGISTERS
  SIPAR0= 172240
  SIPAR1= 172242
  SIPAR2= 172244
  SIPAR3= 172246
  SIPAR4= 172250
  SIPAR5= 172252
  SIPAR6= 172254
  SIPAR7= 172256
  .IF NB
  ;*SUPERVISOR "D" PAGE ADDRESS REGISTERS
  SDPAR0= 172260
  SDPAR1= 172262
  SDPAR2= 172264
  SDPAR3= 172266
  SDPAR4= 172270
  SDPAR5= 172272
  SDPAR6= 172274
  SDPAR7= 172276
  .ENDC
```



MEMORY MANAGEMENT DEFINITIONS

```
.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
```

TSV05 REGISTER AND PACKET DEFINITIONS

```

946                                     .SBTTL  TSV05 REGISTER AND PACKET DEFINITIONS
947
948                                     ;
949                                     ; SOME GENERAL EQUATES.
950                                     ;
951
952      000004      ERRVEC==      4      ; POINTER TO ERROR VECTOR FOR BUS TIME OUT.
953      000060      TTIVEC==     60      ; INTERRUPT VECTOR FOR CONSOLE INPUT
954      177560      TTICSR==    177560   ; BUS ADDRESS OF CONSOLE INPUT
955      177562      TTIBFR==    177562   ; CONSOLE INPUT DATA BUFFER
956      177520      BDVPCR==    177520   ; BDV11 PAGE CONTROL REGISTER
957
958                                     ;+
959                                     ;BIT DEFINITIONS FOR TSSR REGISTER
960                                     ;-
961
962      100000      SC=      BIT15      ;SPECIAL CONDITION
963      040000      BIE=     BIT14      ;BUS INTERFACE ERROR
964      020000      SCE=     BIT13      ;SANITY CHECK ERROR
965      010000      RMR=     BIT12      ;MODIFICATION REFUSED
966      004000      NXM=     BIT11      ;NONEXISTANT MEMORY ERROR
967      002000      NBA=     BIT10      ;NEED BUFFER ADDRESS
968      001400      HIADDR= BIT9:BIT8   ;EXTENDED ADDRESS BITS
969      000200      SSR=     BIT7       ;SUB SYSTEM READY
970      000100      OFL=     BIT6       ;OFF LINE BIT
971      000060      FATERR= BIT4:BITS   ;FATAL TERMINATION ERROR CODES
972      000016      TERCLS= BIT3:BIT2:BIT1 ;TERMINATION CODES
973
974                                     ;+
975                                     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0
976                                     ;(XST0)
977                                     ;
978                                     ;-
979
980
981      100000      XSOTMK=  BIT15      ;TAPE MARK DETECTED
982      040000      XSORLS=  BIT14      ;RECORD LENGTH SHORT
983      020000      XSOLET=  BIT13      ;LOGICAL END OF TAPE
984      010000      XSORLL=  BIT12      ;RECORD LENGTH LONG
985      004000      XSOWLE=  BIT11      ;WRITE LOCK ERROR
986      002000      XSONEF=  BIT10      ;NON EXECUTABLE FUNCTION
987      001000      XSOILC=  BIT9       ;ILLEGAL COMMAND
988      000400      XSOILA=  BIT8       ;ILLEGAL ADDRESS
989      000200      XSOMOT=  BIT7       ;TAPE IN MOTION
990      000100      XSOONL=  BIT6       ;TRANSPORT ON LINE
991      000040      XSOIE=   BIT5       ;INTERRUPT ENABLE
992      000020      XSOVCK=  BIT4       ;VOLUME CHECK BIT
993      000010      XSOPED=  BIT3       ;PHASE ENCODED DRIVE
994      000004      XSOWLK=  BIT2       ;WRITE LOCKED
995      000002      XSOTOT=  BIT1       ;BEGINNING OF TAPE
996      000001      XSOTOT=  BIT0       ;END OF TAPE

```

TSV05 REGISTER AND PACKET DEFINITIONS

```

998
999
1000
1001
1002      100000
1003      040000
1004      020000
1005      017375
1006      000400
1007      000002
1008
1009
1010
1011
1012
1013      100000
1014      040000
1015      035400
1016      C02000
1017      000200
1018      000100
1019      000077
1020      000007
1021
1022
1023
1024
1025
1026      177400
1027      000200
1028      000100
1029      000040
1030      000020
1031      000010
1032      000006
1033      000001
1034
1035
1036
1037
1038
1039      100000
1040      040000
1041      020000
1042      017400
1043      000377
1044
1045
1046
1047
1048
1049
1050
1051      000006
1052      000006

;+
;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
;(XST1)
;-
X1.DLT = BIT15      ;DATA LATE
X1.SPARE= BIT14     ;NOT USED
X1.COR = BIT13     ;CORRECTABLE DATA ERROR
X1.MBZ = BIT12+BIT11+BIT10+BIT9+BIT7+BIT6+BIT5+BIT4+BIT3+BIT2+BIT0 ;ALWAYS 0
X1.RBP = BIT8      ;READ BUS PARITY ERROR
X1.UNC = BIT1      ;UNCORRECTABLE DATA OR HARD ERROR

;+
;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2
;(XST2)
;-
X2.OPM = BIT15     ;OPERATION IN PROGRESS (TAPE MOVING)
X2.RCE = BIT14     ;RAM CHECKSUM ERROR
X2.SPARE= BIT13+BIT12+BIT11+BIT9+BIT8 ;NOT USED BY TSV05 (ALWAYS=0)
X2.WCF = BIT10     ;WRITE CLOCK FAILURE (FIFO NOT EMPTIED BY TRANSPORT)
X2.EXTF = BIT7     ;IF WRITE CHAR CMD THEN = EXTENDED FEATURES ENABLED
X2.BUFE = BIT6     ;IF WRITE CHAR CMD THEN = BUFFERING ENABLED
X2.REV = 000077   ;IF WRITE CHAR CMD THEN = MICROCODE REVISION LEVEL
X2.UNIT = BIT2+BIT1+BIT0 ;IF GET STATUS THEN = CURRENTLY SELECTED UNIT NO.

;+
;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3
;(XST3)
;-
X3.MDE = 177400   ;MICRO-DIAGNOSTIC ERROR CODE
X3.SPARE= BIT7     ;NOT USED BY TSV05
X3.OPI = BIT6     ;OPERATION INCOMPLETE
X3.REV = BIT5     ;REVERSE
X3.TRF = BIT4     ;TRANSPORT RESPONSE FAILURE
X3.DCK = BIT3     ;DENSITY CHECK
X3.MBZ =BIT2+BIT1 ;NOT USED ALWAYS 0
X3.RIB = BIT0     ;REVERSE INTO BOT

;+
;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4
;(XST4)
;-
X4.HSP = BIT15     ;HIGH SPEED
X4.RCE = BIT14     ;RETRY COUNT EXCEEDED
X4.TSM = BIT13     ;TRANSPORT SPECIAL MODE
X4.MBZ = BIT12+BIT11+BIT10+BIT9+BIT8 ;NOT USED ALWAYS 0
X4.WRC = 000377   ;WRITE RETRY COUNT FIELD

;+
;TSSR TERMINATION CODES (BIT 0-2)
;-
TSREJ= 3+2      ;COMMAND REJECTED
UNREC= 6        ;UNRECOVERABLE ERROR

```

TSV05 REGISTER AND PACKET DEFINITIONS

```

1054      ;*
1055      ;
1056      ;DEVICE REGISTER OFFSETS
1057      ;
1058      ;-
1059
1060      000000      TSBA== 0
1061      000000      TSDB== 0      ;TSDB/TSBA REGISTER
1062      000001      TSBAH== 1
1063      000001      TSDBH== 1      ;TSDB/TSBA REGISTER HIGH BYTE
1064      000002      TSSR== 2      ;TSSR REGISTER
1065      000003      TSSRH== 3      ;TSSR REGISTER HIGH BYTE
1066
1067      ;*
1068      ; TSDB ADDRESS BIT DEFINITIONS
1069      ;-
1070      000003      A1716 = BIT1+BIT0      ;ADDRESS BITS 17:16 ARE IN 1:0
1071
1072      ;*
1073      ; COMMAND DEFINITIONS
1074      ;-
1075      000017      P.GETSTAT      = 17      ;GET STATUS
1076      000013      P.INIT        = 13      ;INITIALIZE
1077      000012      P.CONTROL     = 12      ;CONTROL COMMANDS
1078      000011      P.FORMAT      = 11      ;FORMAT
1079      000010      P.POSITION    = 10      ;POSITION
1080      000006      P.WRTSUB      = 6       ;SUBSYSTEM WRITE
1081      000005      P.WRITE       = 5       ;WRITE
1082      000004      P.WRTCHAR    = 4       ;WRITE CHARACTERISTICS
1083      000001      P.READ       = 1       ;READ
1084
1085      ;*
1086      ; COMMAND PACKET HEADER WORD BIT DEFINITIONS
1087      ;-
1088      100000      P.ACK        = BIT15      ;BUFFER AVAIL FOR CONTROLLER
1089      040000      P.CVC        = BIT14      ;CLEAR VOLUME CHECK
1090      020000      P.OPP        = BIT13      ;REVERSE SEQUENCE OF DATA BITS
1091      010000      P.SWB        = BIT12      ;SWAP BYTES IN MEMORY
1092      007400      P.MODE      = BIT11:BIT10:BIT9:BIT8 ;EXTENDED COMMAND MODE FIELD
1093      000200      P.IE        = BIT7       ;INTERRUPT ENABLE
1094      000140      P.FMT= BIT6:BIT5      ;PACKET HEADER TYPE (ALWAYS=0)
1095      000037      P.CMD       = 37        ;MAJOR COMMAND FIELD
1096
1097      ;*
1098      ; CONTROL COMMAND MODE CODES
1099      ;-
1099      000000      PC.RELEASE   = 0*256.    ;RELEASE BUFFER
1100      000400      PC.REWIND   = 1*256.    ;REWIND
1101      001000      PC.NOOP     = 2*256.    ;NO-OP
1102      002000      PC.IEREW    = 4*256.    ;REWIND IMMEDIATE INTERRUPT
1103      002400      PC.ERASE     = 5*256.    ;SECURITY ERASE

```

TSV05 REGISTER AND PACKET DEFINITIONS

```

1105      ;*
1106      ; CONTROLLER RAM DEFINITIONS
1107      ;-
1108      000167      RMCHBEG = 167      ; CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
1109      000200      RMCHEND = 200      ; CHARACTERISTICS IO DATA END RAM ADDRESS
1110      000201      RMPKTBEG = 201      ; COMMAND PACKET BEGIN RAM ADDRESS
1111      000210      RMPKTEND = 210      ; COMMAND PACKET END RAM ADDRESS
1112      000215      RMMSGBEG = 215      ; MESSAGE BUFFER BEGIN RAM ADDRESS
1113      000234      RMMSGEND = 234      ; MESSAGE BUFFER END RAM ADDRESS
1114      ;*
1115      ;
1116      ; REGISTER DEFINITIONS IN THE MESSAGE BUFFER
1117      ;
1118      ;-
1119
1120      000006      XST0 = 6      ; EXTENDED STATUS REGISTER 0 (WORD 4)
1121      000010      XST1 = 8      ; EXTENDED STATUS REGISTER 1 (WORD 5)
1122      000012      XST2 = 10      ; EXTENDED STATUS REGISTER 2 (WORD 6)
1123      C00014      XST3 = 12      ; EXTENDED STATUS REGISTER 3 (WORD 7)
1124      000016      XST4 = 14      ; EXTENDED STATUS REGISTER 4 (WORD 8)
1125
1126      ;*
1127      ;
1128      ; OFFSETS TO WORD LOCATIONS IN PACKET DEFINITIONS
1129      ;
1130      ;-
1131
1132      000002      PKLOW = 2      ; LOW ORDER CHARACTERISTIC DATA POINTER
1133      000004      PKHI = 4      ; HIGH ORDER CHARACTERISTIC DATA POINTER
1134      000006      PKBCNT = 6      ; NUMBER OF BYTES IN DATA PACKET
1135
1136      000010      EXBCNT = 10      ; NUMBER OF BYTES IN EXTENDED DATA PACKET
1137
1138      ;*
1139      ; DATA PACKET OFFSETS FOR WRITE SUBSYSTEM COMMAND
1140      ;-
1141      000000      BSEL0 = 0      ; BYTE 0
1142      000001      BSEL1 = 1      ; BYTE 1
1143      000002      SEL2 = 2      ; WORD 2
1144      000004      SELDATA = 4      ; WORD 3

```



TSV05 REGISTER AND PACKET DEFINITIONS

```

1146
1147
1148
1149      000000
1150      000001
1151      000002
1152      000003
1153      000004
1154      000005
1155      000006
1156      000007
1157      000010
1158      000011
1159      000020
1160      000021
1161      000022
1162      000023
1163      000024
1164
1165
1166
1167
1168      000200
1169      000100
1170      000040
1171      000020
1172      000010
1173      000004
1174      000002
1175      000001
1176
1177
1178
1179
1180      000200
1181      000100
1182      000040
1183      000020
1184      000010
1185      000004
1186      000002
1187      000001
1188
1189
1190
1191
1192      000200
1193      000020
1194      000010
1195      000006
1196      000001

```

```

;+
;BSELO SELECT CODES FOR WRITE SUBSYSTEM COMMAND
;-
PW.NOP      = 0      ;NO-OP
PW.RDRAM    = 1      ;READ RAM
PW.WTRAM    = 2      ;WRITE RAM
PW.RFIFO    = 3      ;READ FIFO
PW.WFIFO    = 4      ;WRITE FIFO
PW.RDSTAT   = 5      ;READ STATUS
PW.WCTL     = 6      ;WRITE TAPE CONTROL
PW.WFMT     = 7      ;WRITE TAPE FORMAT
PW.WMISC    = 10     ;WRITE MISCELLANEOUS
PW.WNPR     = 11     ;WRITE NPR CONTROL
PW.D22      = 20     ;DO MICROTEST 22
PW.D11      = 21     ;DO MICROTEST 11
PW.D13      = 22     ;DO MICROTEST 13
PW.NO1311   = 23     ;DISABLE MICROTEST 11 AND 13
PW.RDXT     = 24     ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSPORTS)

;+
;BSEL1 CODES FOR WRITE TAPE CONTROL
;-
WC.IFAD     = BIT7   ;IFAD - FORMATTER ADDRESS
WC.IOTAD    = BIT6   ;ITADO - TRANSPORT ADDRESS BIT 0
WC.I1TAD    = BIT5   ;ITAD1 - TRANSPORT ADDRESS BIT 1
WC.I5RESV   = BIT4   ;IRESV5 - RESERVED #5
WC.IREW     = BIT3   ;IREW - REWIND
WC.IRWU     = BIT2   ;IRWU - REWIND AND UNLOAD
WC.IFEN     = BIT1   ;IFEN - FORMATTER ENABLE
WC.IGO      = BIT0   ;GO

;+
;BSEL1 CODES FOR WRITE FORMAT
;-
WF.IHISP    = BIT7   ;IHISP - HIGH SPEED
WF.IWRT     = BIT6   ;IWRT - WRITE
WF.IREV     = BIT5   ;IREV - REVERSE
WF.IWFM     = BIT4   ;IWFM - WRITE FILE MARK
WF.IEDIT    = BIT3   ;IEDIT - EDIT
WF.IERASE   = BIT2   ;IERASE - ERASE
WF.I3RESV   = BIT1   ;IRESV3 - RESERVED #3
WF.I4RESV   = BIT0   ;IRESV4 - RESERVED #4

;+
;BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND
;-
MS.EXT      = BIT7   ;INVERT SENSE OF EXTENDED FEATURES SWITCH
MS.RSFIFO   = BIT4   ;RESET FIFO AND INPUT PARITY ERRORR
MS.RSTAPE   = BIT3   ;RESET TAPE STATUS IN 2 FLIP-FLOPS
MS.ATTN     = BIT2:BIT1 ;ATTENTION TRIGGER FIELD
MS.RSD      = BIT0   ;RESET TIMER A,B THEN DELAY TIMES IN SEL2

```

TSV05 REGISTER AND PACKET DEFINITIONS

```

1198
1199      ; MS.ATTN SUBCODES
1200      ;-
1201      000000      MSA.NOP = 0*2      ;NO-OP (NOTHING TRIGGERED)
1202      000002      MSA.VOL = 1*2      ;SIMULATE ON-LINE/OFF-LINE TRANSITION
1203      000004      MSA.NRAM= 2*2      ;FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)
1204      000006      MSA.FRAM= 3*2      ;FORCE FATAL RAM ERROR (CAUSES SCE TO SET)
1205
1206      ; WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS
1207      ;-
1208      000200      NP.IR      = BIT7      ;INTERRUPT REQUEST (0-1 TRANSITION)
1209      000100      NP.OUT     = BIT6      ;TAPE DATA DIRECTION OUT (0= IN)
1210      000040      NP.LOOP   = BIT5      ;ENABLE TRANSPORT LOOPBACK
1211      000020      NP.WRP    = BIT4      ;WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)
1212
1213      ; READ STATUS MESSAGE BUFFER BIT DEFINITIONS
1214      ;-
1215
1216      C00200      S2.DIM      = BIT7      ;WORD #9 BYTE 2 DATA IN MISS
1217      000100      S2.ILW     = BIT6      ; ILW H
1218      000040      S2.OUTRDY  = BIT5      ; OUT RDY H
1219      000020      S2.INRDY   = BIT4      ; IN RDY H
1220      000010      S2.ATIMR   = BIT3      ; TIMER A FLAG H
1221      000004      S2.BTIMR   = BIT2      ; TIMER B FLAG H
1222      000003      S2.UNDEF   = BIT1.BITO ;(UNDEFINED)
1223      100000      S1.PARIN   = BIT15     ;WORD #8 BYTE 1 PARIN H
1224      040000      S1.I2RESV  = BIT14     ; IRESV2
1225      020000      S1.I1RESV  = BIT13     ; IRESV1
1226      010000      S1.IEOT    = BIT12     ; IEOT L
1227      004000      S1.IIDENT  = BIT11     ; IIDENT H
1228      002000      S1.ICER    = BIT10     ; ICER H
1229      001000      S1.IFMK    = BIT9      ; IFMK H
1230      000400      S1.IHER    = BIT8      ; IHER H
1231      000200      S0.ISPEED  = BIT7      ;WORD #8 BYTE 0 ISPEED H
1232      000100      S0.IRDY    = BIT6      ; IRDY L
1233      000040      S0.IONL    = BIT5      ; IONL L
1234      000020      S0.ILDP    = BIT4      ; ILDP L
1235      000010      S0.IDBY    = BIT3      ; IDBY L
1236      000004      S0.IRWD    = BIT2      ; IRWD L
1237      000002      S0.IFBY    = BIT1      ; IFBY L
1238      000001      S0.IFPT    = BIT0      ; IFPT L

```

## SPECIAL MACROS AND OPDEFS.

```

1240             .SBTTL SPECIAL MACROS AND OPDEFS.
1241
1242             ;+
1243             ;SAVE GENERAL REGS 1 TO 5
1244             ;-
1245
1246             .MACRO SAVREG
1247             JSR     R5,REGSAV
1248             .ENDM
1249
1250             ;+
1251             ; MACRO TO FORCE AN ERROR
1252             ;-
1253             .MACRO FORCERROR TAG,NOTSSR
1254             .NLIST
1255             .IIF NDF LISTALL, .NLIST
1256             .LIST
1257             .IF B NOTSSR
1258                 MOV     TSSR(R5),R1      ;READ TSSR
1259             .ENDC
1260                 MOV     FORCER,FORCER    ;IS FORCER SET? (LEAVE C BIT ALONE)
1261                 BNE     TAG              ;BR IF YES
1262             .NLIST
1263             .IIF NDF LISTALL, .LIST
1264             .LIST
1265             .ENDM
1266
1267             ;+
1268             ; MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
1269             ; WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
1270             ; SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
1271             ; FORCER TO 17777
1272             ; TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
1273             ;-
1274             .MACRO FORCEEXIT TAG
1275             .NLIST
1276             .IIF NDF LISTALL, .NLIST
1277             .LIST
1278                 MOV     FORCER,FORCER    ;IS FORCER NEGATIVE?
1279                 BMI     TAG              ;BR IF YES
1280             .NLIST
1281             .IIF NDF LISTALL, .LIST
1282             .LIST
1283             .ENDM
1284             ;+
1285             ; MACRO TO INCREMENT ERROR COUNTS
1286             ;-
1287             .MACRO NEXT.ERRNO
1288             .NLIST
1289             ;;;.IIF NDF LISTALL, .NLIST
1290                 ERRNO=ERRNO+1
1291             ;;;.IIF NDF LISTALL, .LIST
1292             .LIST
1293             .ENDM

```

SPECIAL MACROS AND OPDEFS.

```

1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306      000000
1307
1308
1309
1310
1311
1312
1313
1314 002166 000000
1315
1316

```

```

;+
;MACRO TO PERFORM XOR
;-
      .MACRO XOR      A,B
MOV    A,-(SP)
BIC    B,(SP)
BIC    A,B
BIS    (SP)+.B
      .ENDM

EN=0      ; INITIALIZE ERROR NUMBER
.SBTTL   FORCER - FORCE ERROR FLAG

;
; THE FOLLOWING LOCATIONS MAY BE PATCHED BY THE USER
; TO OBTAIN THE RESULTS DESCRIBED FOR EACH.
;
FORCER::      0      ; FORCE TYPE ALL HARD ERRORS (THE ONES CALLED -
; - BY THE MACRO "IFERROR"). AN ERROR NEED NOT -
; - EXIST, JUST ASSUME AND TYPE THE MESSAGE.

```

## GLOBAL DATA SECTION

```

1318             .SBTTL  GLOBAL DATA SECTION
1319
1320             ;**
1321             ;THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1322             ;IN MORE THAN ONE TEST.
1323             ;--
1324
1325             ;
1326             ;THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
1327             ;SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
1328             ;
1329 002170 000000 EPRTSW::      .WORD  0      ;PRINT SWITCH
1330 002172 000000 UNITN::      .WORD  0      ;UNIT # UNDER TEST.
1331 002174 000000 QVP::        .WORD  0      ;QUICK VERIFY FLAG.
1332 002176 000000 CSRADDR::   .WORD  0      ;ADDRESS OF CSR FOR CURRENT DEVICE
1333 002200 000224 IVEC::        .WORD  224     ;INTERRUPT VECTOR
1334 002202 000200 IPRI::        .WORD  PRI04   ;INTERRUPT PRIORITY.
1335 002204 000000 TSTCNT::    .WORD  0      ;NUMBER OF TESTS RUN IN THIS PASS
1336 002206 000000 LOOPCNT::   .WORD  0      ;REMAINING ITERATION COUNT FOR TEST
1337 002210 000000 DEVCNT::    .WORD  0      ;NUMBER OF DEVICE UNDER TEST
1338 002212 000000 FATFLG::    .WORD  0      ;SET IF FATAL ERROR IS DETECTED IN TEST
1339 002214 000000 INTRECV::   .WORD  0      ;SET IF TAPE INTERRUPT WAS RECEIVED
1340 002216 000000 EXTFEA::    .WORD  0      ;EXTENDED FEATURES SOFTWARE SW 0-OFF;1-ON
1341 002220 000000 BENBSW::    .WORD  0      ;BUFFER ENABLE SWITCH SW 0-OFF;1-ON
1342 002222 000000 EXPD::      .WORD  0      ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
1343 002224 000000 RECV::      .WORD  0      ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
1344 002226 000000 ERRHI::     .WORD  0      ;HIGH ADDRESS MEMORY ERROR
1345 002230 000000 ERRLO::     .WORD  0      ;LOW ADDRESS MEMORY ERROR
1346 002232 000000 RAMDATA::   .BLKW  16.    ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
1347 002272 000000 RAMSIZ::    .WORD  0      ;RAM DATA SIZE FOR PRAMPKT ROUTINE
1348 002274 000000 RCVHIADD::  .WORD  0      ;RECEIVED BUFFER HIGH ADDRESS
1349 002276 000000 RCVLOADD::  .WORD  0      ;RECEIVED BUFFER LOW ADDRESS
1350 002300 000000 COUNT::     .WORD  0      ;TEST COUNT PATTERN
1351 002302 000000 DATA::     .WORD  0      ;TEST DATA
1352 002304 000000 TSTFLAG::   .WORD  0      ;TEST FLAG WORD
1353 002306 000000 TSTPTR::    .WORD  0      ;TSTBLK POINTER
1354 002310 000000 PRMNO::     .WORD  0      ;PRINT ROUTINE TEMP
1355 002312 000000 EXPMSG::    .BLKB  100.   ;EXPECTED MESSAGE BUFFER DATA
1356 002456 000000 RECMMSG::   .BLKB  100.   ;RECEIVED MESSAGE BUFFER DATA
1357 002622 000000 TMPBFR::    .BLKB  80.    ;TEMPORARY STORAGE FOR PRINT

```



TSTBLK - TEST DATA TABLE

.SBTTL TSTBLK - TEST DATA TABLE

1359  
 1360  
 1361  
 1362  
 1363  
 1364  
 1365  
 1366  
 1367  
 1368  
 1369  
 1370  
 1371  
 1372  
 1373  
 1374  
 1375 002742  
 1376 002742 000000  
 1377 002744 177777  
 1378 002746 000001  
 1379 002750 000002  
 1380 002752 000004  
 1381 002754 000010  
 1382 002756 000020  
 1383 002760 000040  
 1384 002762 000100  
 1385 002764 000200  
 1386 002766 000400  
 1387 002770 001000  
 1388 002772 002000  
 1389 002774 004000  
 1390 002776 010000  
 1391 003000 020000  
 1392 003002 040000  
 1393 003004 100000  
 1394 003006 177776  
 1395 003010 177775  
 1396 003012 177773  
 1397 003014 177767  
 1398 003016 177757  
 1399 003020 177737  
 1400 003022 177677  
 1401 003024 177577  
 1402 003026 177377  
 1403 003030 176777  
 1404 003032 175777  
 1405 003034 173777  
 1406 003036 167777  
 1407 003040 157777  
 1408 003042 137777  
 1409 003044 077777  
 1410 003046 125252  
 1411 003050 052525  
 1412            003052

```

; *
; THIS TABLE CONTAINS TEST DATA USED IN SEVERAL TESTS
; IN SEQUENCE THE DATA IS:
;
;     ALL ZEROS
;     ALL ONES
;     WALKING ONES
;     WALKING ZEROS
;     ALTERNATING ONES AND ZEROS
; -
    
```

```

TSTBLK::
    .WORD 0 ;ALL ZEROS
    .WORD 177777 ;ALL ONES
    .WORD BIT0 ;DATA FOR WALKING ONES
    .WORD BIT1
    .WORD BIT2
    .WORD BIT3
    .WORD BIT4
    .WORD BIT5
    .WORD BIT6
    .WORD BIT7
    .WORD BIT8
    .WORD BIT9
    .WORD BIT10
    .WORD BIT11
    .WORD BIT12
    .WORD BIT13
    .WORD BIT14
    .WORD BIT15 ;DATA FOR WALKING ZEROS
    .WORD †CBIT0
    .WORD †CBIT1
    .WORD †CBIT2
    .WORD †CBIT3
    .WORD †CBIT4
    .WORD †CBIT5
    .WORD †CBIT6
    .WORD †CBIT7
    .WORD †CBIT8
    .WORD †CBIT9
    .WORD †CBIT10
    .WORD †CBIT11
    .WORD †CBIT12
    .WORD †CBIT13
    .WORD †CBIT14
    .WORD †CBIT15
    .WORD 125252 ;ALTERNATING ONES, ZEROS
    .WORD 052525 ;ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE

TBLEND= .
    
```

GLOBAL ENVIRONMENT STORAGE

```

1414          .SBTTL GLOBAL ENVIRONMENT STORAGE
1415          ;
1416          ;STORAGE FOR DEVICE REGISTERS
1417          ;
1418 003052 000000 100000 000000 DUMMY: 0,100000,0,0 ;DUMMY DEVICE REGISTERS...
1419 003062 000000 000000 000000      0,0,0,0,0,0,0,0 ;...FOR MULTI-UNIT CHECKOUT.
1420          ;
1421          ;
1422 003102 000000 DUFLG::          .WORD 0          ;"DROPPED UNIT" FLAG.
1423          ;INHIBITS CODE IN "CLEAN-UP".
1424 003104 000000 NODEV::          .WORD 0          ;FLAG TO SAY NO DEVICE.
1425          ;
1426 003106 000000 TEMP1::          .WORD 0          ;SOME TEMP LOCATIONS.
1427 003110 000000 TEMP2::          .WORD 0
1428 003112 000000 XXCOMM::         .WORD 0          ;XXDP* COMM BLOCK POINTER.
1429 003114 000000 FREE::          .WORD 0          ;1ST FREE MEMORY ADDRESS...
1430 003116 000000 FRESIZ::         .WORD 0          ;...AND SIZE (IN WORDS).
1431 003120 000000 FREEHI: .WORD 0          ;LAST WORD IN FREE SPACE
1432 003122 C00000 KTFLG::          .WORD 0          ;KT11, MEM AVAIL FLAG -
1433          ;- .WORD 0 = <24K OR NO KT -
1434          ;- NZ = >24K AND KT.
1435 003124 000000 KTENABLE::       .WORD 0          ;SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
1436 003126 000000 NXMFLG::         .WORD 0          ;SET IF WE CAN TEST CLEARED OTHERWISE
1437 003130 000000 NXMLO::          .WORD 0          ;NXM LO ADDRESS BITS
1438 003132 000000 NXMHI::          .WORD 0          ;NXM HI ADDRESS BITS FOR DAL'S 16-21
1439 003134 000000 T23A::          .WORD 0          ;11/23A FLAG
1440 003136 000000 T23B::          .WORD 0          ;11/23B FLAG
1441 003140 000000 T3BFLG::         .WORD 0          ;TEST 3B FLAG +0
1442 003142 002000 PST32W::         .WORD 2000        ;32W BLOCK ADDRESS FOR 32K START
1443 003144 000000 SIFLAG::         .WORD 0
1444 003146 000000 BADDAT::         .WORD 0          ;ACTUAL DATA
1445 003150 000000 GDDAT::          .WORD 0          ;EXPECTED DATA
1446 003152 000000 LOOPFL::         .WORD 0
1447 003154          CTAB::          ;CONFIGURATION TABLES.
1448 003154 000000 CTABM::          .WORD 0          ;CONFIG WORK.
1449          .WORD 0
1450          .WORD 0
1451          .WORD 0
1452 003164 177777          .WORD -1          ;END OF MEM TABLE.
1453 003166          CTABE::
1454          ;ERROR STATISTICS TABLE (1 WORD PER UNIT), 64 UNITS MAX:
1455          ;
1456          ;          0          =          UNIT NOT TESTED
1457          ;          100000      =          UNIT ONLINE, NO ERRORS
1458          ;          10XXXX      =          UNIT ONLINE, ENCOUNTERED XXXX ERRORS
1459          ;          160000      =          UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
1460          ;          160001      =          UNIT DROPPED, NOT IDLE AT START
1461          ;          14XXXX      =          UNIT DROPPED, ENCOUNTERED XXXX ERRORS
1462          ;
1463 003166          ERTABL:          .BLKW 64.
1464 003366 000000          ERTABE:          .WORD 0
1465          ;
1466 003370 000000          SKIPT: .WORD 0          ;1-SKIP SUBTEST 0=NO SKIP OF SUBTEST

```

GLOBAL TEXT MESSAGES

```

1468                                     .SBTTL GLOBAL TEXT MESSAGES
1469                                     ;**
1470                                     ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
1471                                     ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
1472                                     ; MORE THAN ONE TEST.
1473                                     ;--
1474                                     ;*
1475                                     ; NAMES OF DEVICES SUPPORTED
1476                                     ;-
1477 003372                                DEVTYP <TSV05>
003372                                L#DVTYP::
003372                                .ASCIZ /TSV05/
124                                .EVEN
123
126

1478                                     ;*
1493                                     ; TEST DESCRIPTION
1494                                     ;-
1495                                     ;
1496 003400                                DESCRIPT <**** TSV05 LOGIC DIAGNOSTIC - CHK CABLES-TRANSPORT IF ERR ****>
003400                                L#DESC::
003400                                .ASCIZ /**** TSV05 LOGIC DIAGNOSTIC - CHK CABLES-TRANSPORT IF ERR ****/
052                                .EVEN
052
052

1504                                     ;*
1505                                     ; BIT TO ASCII CONVERSION FOR TSSR REGISTER
1506                                     ;-
1507                                     ;
1508 003500                                003540 003543 003547 TSSRBIT:: .WORD 1#,2#,3#,4#,5#,6#,7#,8#
1509 003520                                003601 003605 003611 .WORD 9#,10#,11#,12#,13#,14#,15#,16#
1510 003540                                123 103 000 1#: .ASCIZ 'SC'
1511 003543                                102 111 105 2#: .ASCIZ 'BIE'
1512 003547                                123 103 105 3#: .ASCIZ 'SCE'
1513 003553                                122 115 122 4#: .ASCIZ 'RMR'
1514 003557                                116 130 115 5#: .ASCIZ 'NXM'
1515 003563                                116 102 101 6#: .ASCIZ 'NBA'
1516 003567                                102 111 124 7#: .ASCIZ 'BIT9'
1517 003574                                102 111 124 8#: .ASCIZ 'BIT8'
1518 003601                                123 123 122 9#: .ASCIZ 'SSR'
1519 003605                                117 106 114 10#: .ASCIZ 'OFL'
1520 003611                                102 111 124 11#: .ASCIZ 'BIT5'
1521 003616                                102 111 124 12#: .ASCIZ 'BIT4'
1522 003623                                102 111 124 13#: .ASCIZ 'BIT3'
1523 003630                                102 111 124 14#: .ASCIZ 'BIT2'
1524 003635                                102 111 124 15#: .ASCIZ 'BIT1'
1525 003642                                102 111 124 16#: .ASCIZ 'BIT0'
1526                                     .EVEN
1527 003650                                124 123 123 SFIERR: .ASCIZ 'TSSR ERROR AFTER SOFT INIT'
1528 003703                                124 123 123 SFHERR: .ASCIZ 'TSSR ERROR AFTER BUS RESET'
1529 003736                                040 040 116 NXR: .ASCIZ / NON-EXISTANT DEVICE REGISTER/
1530 003775                                045 101 040 NXR: .ASCIZ /#A ADDRESS: #06/
1531 004016                                045 101 040 TSSX: .ASCII /#A TSBA,TSSR EXP'D: #06#A,#0#N/
1532 004056                                045 101 040 TSSX: .ASCIZ /#A TSBA,TSSR REC'D: #06#A,#06/
1533 004115                                045 116 045 FUSI: .ASCII /#N#A/
1534 004121                                040 040 125 USI: .ASCIZ / UNEXPECTED INTERRUPT/
1535 004150                                040 040 111 NSI: .ASCIZ / INTERRUPT EXPECTED, NOT RECEIVED/
1536 004213                                045 116 045 FNOINTR: .ASCII /#N#A/
1537 004217                                040 040 116 NOINTR: .ASCIZ / NO INTERRUPT WAS GENERATED/
1538 004254                                040 040 111 IFAULT: .ASCIZ / INTERRUPT FAULT/
1539 004276                                045 101 040 INTX: .ASCIZ /#A CPU PC: #06#A TSBA: #06/

```

GLOBAL TEXT MESSAGES

```

1540 004333      040      040      042 NOINIT: .ASCIZ / "BUS-INIT" DIDN'T INITIALIZE CONTROLLER/
1541 004405      040      040      042 NSINIT: .ASCIZ / "SOFT-INIT" DIDN'T INITIALIZE THE DPU/
1542 004455      040      040      042 BRINIT: .ASCIZ / "BUS-RESET" DIDN'T INITIALIZE THE DPU/
1543 004525      000
1544 004526      045      116      000 NULCR: .ASCIZ /#N/
1545 004531      045      101      040 EXPGOT: .ASCIZ /#A EXP'D: #06#A, REC'D: #06/
1546 004565      045      116      045 EXPGT2: .ASCIZ /#NA EXP'D: #06#A, #06#NA REC'D: #0#A, #06/
1547 004641      045      101      040 DUAD12: .ASCIZ /#A REG(W) WRITTEN TO: #06#A REG(R) READ; EXP'D: #06#A, REC'D: #06/
1548 004743      122      101      115 PKTRAM: .ASCIZ 'RAM Contents Do Not Match Packet Sent'
1549 005011      040      040      103 SCME: .ASCIZ / CONFIG DOESN'T MATCH MFG. MASTER/
1550 005054      127      122      111 WRTMSG: .ASCIZ 'WRITE CHARACTERISTICS Failed'
1551 005111      124      123      123 WRTERR: .ASCIZ 'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
1552 005204      124      123      123 RDERR: .ASCIZ 'TSSR Incorrect After READ Command, More Bits Set Than SSR'
1553 005276      106      101      124 SCHERR: .ASCIZ 'FATAL ERROR IN SUBTEST - CHECK TAPE, CABLES, TRANSPORT etc.'
1554 005370      105      122      122 RETERR: .ASCIZ 'ERROR IN SUBTEST - WRITE DATA RETRY FIVE TIMES FAILED'
1555 005456      045      116      045 NOMEM: .ASCIZ '#NA ***** NO NXM ADDRESS--CANNOT TEST NXM TIMEOUT. *****N'
1556 005552      045      116      045 M8186: .ASCIZ '#NA ***** 11/23A SYSTEM *****N'
1557 005643      045      116      045 M8189: .ASCIZ '#NA ***** 11/23B SYSTEM *****N'
1558
1559
1560
1561
1562
1563
1564
1565
1566 005734
1567 005734
1568 005760
1569 005764
1570
1571
1572
1573
1574 005766
1575 005770
1576 005772
1577 005774
1578 006000
1579 006020
013746 003104
012746 003775
012746 000002
010600
104415
062706 000006
004737 005766
104423
005727
000000
001402
004777 177770
012746 004526
012746 000001
010600
104415
062706 000004
000207

```

.SBTTL GLOBAL ERROR REPORT SECTION

```

;
; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX
; CALLS THAT ARE USED IN MORE THAN ONE TEST.
; ASCII TEXT STRINGS ARE FOUND IN THE GLOBAL TEXT SECTION.
;

```

```

;--
BGNMSG NXRERR ;NON-EXISTANT DEVICE REGISTER.
NXRERR:
PRINTX @NXRX,NODEV ;NODEV = NEXM ADDRESS.
MOV NODEV,-(SP)
MOV @NXRX,-(SP)
MOV @2,-(SP)
MOV SP,RO
TRAP C@PNTX
ADD @6,SP
JSR PC,EXTEND ; PRINT EXTENSION IF REQUIRED.
ENDMSG
;10002:
TRAP C@MSG

```

```

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

```

```

EXTEND: TST (PC).
EXTA: 0 ; 0 = NO EXTENSION.
BEQ 1@
JSR PC,@EXTA ; APPEND EXTENSION TEXT.
1@: PRINTX @NULCR ; PRINT A BLANK LINE
MOV @NULCR,-(SP)
MOV @1,-(SP)
MOV SP,RO
TRAP C@PNTX
ADD @4,SP
RTS PC

```

PRITSSR - PRINT TSSR CONTENTS

```

1581 .SBTTL PRITSSR - PRINT TSSR CONTENTS
1582
1583 ;*
1584 ;
1585 ;ROUTINE TO DISPLAY THE CONTENTS, AND BIT DEFINITIONS, OF
1586 ;THE TSSR REGISTER. THIS ROUTINE IS NORMALLY CALLED ONLY
1587 ;BY A MESSAGE PRINTING ROUTINE
1588 ;
1589 ;INPUTS:
1590 ;
1591 ; R1 CONTENTS OF TSSR
1592 ;
1593 ;SUBORDINATE ROUTINES:
1594 ;
1595 ; CHKAMB CHECK FOR AMBIGUOUS CONTENTS
1596 ;
1597 ;-
1598
1599 PRITSSR:
1600 006022 SAVREG ;SAVE GENERAL REGISTERS
1601 006022 MOV R1,R4 ;SAVE THE TSSR CONTENTS
1602 006026 010104 PRINTB #TSSRFOR,R4 ;PRINT THE CONTENTS OF TSSR
1603 006030 MOV R4,-(SP)
1604 006032 010446 006505 MOV #TSSRFOR,-(SP)
1605 006036 012746 000002 MOV #2,-(SP)
1606 006042 010600 MOV SP,R0
1607 006044 104414 TRAP C#PNTB
1608 006046 062706 000006 ADD #6,SP
1609 006052 010400 MOV R4,R0 ;GET TSSR BACK FOR CHKAMB
1610 006054 004737 016134 JSR PC,CHKAMB ;ARE CONTENTS AMBIGUOUS ?
1611 006060 103410 BCS 5# ;BRANCH IF NOT
1612 006062 PRINTX #AMBTSSR ;SHOW CONTENTS ARE AMBIGUOUS
1613 006062 012746 006725 MOV #AMBTSSR,-(SP)
1614 006066 012746 000001 MOV #1,-(SP)
1615 006072 010600 MOV SP,R0
1616 006074 104415 TRAP C#PNTX
1617 006076 062706 000004 ADD #4,SP
1618 006102 010403 5# MOV R4,R3 ;CONTENTS OF TSSR
1619 006104 042703 001476 BIC #HIADDR!FATERR!TERCLS,R3 ;CLEAR ALL MULTIPLE BIT FIELDS
1620 006110 001434 BEQ 20# ;NO BITS ARE SET
1621 006112 012702 002622 MOV #TMPBFR,R2 ;TEMPORARY ASCII BUFFER
1622 006116 012701 003500 MOV #TSSRBIT,R1 ;ASCII EQUIVALENT OF BITS
1623 006122 005703 10# TST R3 ;REMAINING BITS TO CONVERT
1624 006124 001413 BEQ 15# ;BRANCH WHEN ALL ARE DONE
1625 006126 000241 CLC ;CLEAR CARRY FOR SHIFT
1626 006130 006103 ROL R3 ;SHIFT NEXT BIT TO CARRY
1627 006132 103006 BCC 13# ;BRANCH IF BIT NOT SET
1628 006134 011100 MOV (R1),R0 ;POINTER TO BIT DEFINITION
1629 006136 112022 11# MOVB (R0)+,(R2)+ ;MOVE ASCII TO BUFFER
1630 006140 001376 BNE 11# ;MOVE ALL BITS
1631 006142 112762 000054 177777 MOVB #' , -1(R2) ;INSERT A COMMA TO TERMINATE
1632 006150 005721 13# TST (R1)+ ;POINT TO NEXT DESCRIPTION
1633 006152 000763 BR 10# ;GET THE REMAINING BITS
1634 006154 105042 15# CLRB -(R2) ;TERMINATE THE LINE
1635 006156 PRINTX #TSSDEF,#TMPBFR ;PRINT THE BIT DEFINITIONS
1636 006156 012746 002622 MOV #TMPBFR,-(SP)
1637 006162 012746 006676 MOV #TSSDEF,-(SP)

```

PRITSSR - PRINT TSSR CONTENTS

006166	012746	000002		MOV	#2,-(SP)	
006172	010600			MOV	SP,R0	
006174	104415			TRAP	C#PNTX	
006176	062706	000006		ADD	#6,SP	
1625						
1626	006202	010403	20#:	MOV	R4,R3	;GET THE TSSR CONTENTS
1627	006204	042703		BIC	#+CTERCLS,R3	;CLEAR ALL BUT TERMINATION
1628	006210	016303	177761	MOV	TCOCOD(R3),R3	;GET THE TERMINATION CODE MEANING
1629	006214		006766	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	
1630	006236	010403		MOV	R4,R3	;TSSR CONTENTS AGAIN
1631	006240	042703	177717	BIC	#+CFATERR,R3	;CLEAR ALL BUT FATAL TERMINATION
1632	006244	001416		BEQ	25#	;DON'T PRINT IF ZERO
1633	006246	006203		ASR	R3	
1634	006250	006203		ASR	R3	
1635	006252	006203		ASR	R3	;ALINE TERMINATION CODE FOR INDEX
1636	006254	016303	007326	MOV	TSFCOD(R3),R3	;GET THE FATAL TERMINATION CODE
1637	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	
1638	006302	042704	176377	25#:	BIC	#+CHIADDR,R4
1639	006306	001411		BEQ	30#	;CLEAR ALL BUT EXTENDED ADDRESS
1640	006310			PRINTX	#TEXASC,R4	;DON'T PRINT IF ZERO
	006310	010446		MOV	R4,-(SP)	;PRINT THE EXTENDED ADDRESS BITS
	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	
1641	006332	013703	002170	30#:	MOV	EPRTSW,R3
1642	006336			PRINTX	R3	;PRINT MESSAGE BUFFER ADDRESS
	006336	010346		MOV	R3,-(SP)	;PRINT PROPER MESSAGE
	006340	012746	000001	MOV	#1,-(SP)	
	006344	010600		MOV	SP,R0	
	006346	104415		TRAP	C#PNTX	
	006350	062706	000004	ADD	#4,SP	
1643	006354	000207		RTS	PC	;RETURN TO CALLER

PRITSSR - PRINT TSSR CONTENTS

1655	006356	045	116	045	EPRT1:	.ASCIZ	'#N#A *****CHECK CABLES BETWEEN M7196 AND TRANSPORT*****'
1656	006446	045	116	045	EPRT2:	.ASCIZ	'#N#A *****CHECK TRANSPORT*****'
1662	006505	045	116	045	TSSRFOR:	.ASCIZ	'#N#A TSSR = #06'
1663	006525	045	116	045	TEXASC:	.ASCIZ	'#N#A Extended Address Bits = #06'
1664	006566	045	116	045	TCOASC:	.ASCIZ	'#N#A Termination Class Code = #T'
1665	006627	045	116	045	TFCASC:	.ASCIZ	'#N#A Fatal Termination Class Code = #T'
1666	006676	045	116	045	TSSDEF:	.ASCIZ	'#N#A TSSR Bits Set: #T'
1667	006725	045	116	045	AMBTSSR:	.ASCIZ	'#N#A TSSR Contents Are Ambiguous'
1668						.EVEN	
1669	006766	007006	007031	007057	TCOCOD:	.WORD	1#,2#,3#,4#,5#,6#,7#,8#
1670	007006	116	157	162	1#:	.ASCIZ	'Normal Termination'
1671	007031	124	145	162	2#:	.ASCIZ	'Termination Condition'
1672	007057	124	141	160	3#:	.ASCIZ	'Tape Status Alert'
1673	007101	106	165	156	4#:	.ASCIZ	'Function Reject'
1674	007121	122	145	143	5#:	.ASCIZ	'Recoverable Error - Tape Position One Record Down'
1675	007203	122	145	143	6#:	.ASCIZ	'Recoverable Error - Tape Was Not Moved'
1676	007252	125	156	162	7#:	.ASCIZ	'Unrecoverable Error'
1677	007276	106	141	164	8#:	.ASCIZ	'Fatal Controller Error'
1678						.EVEN	
1679							
1680	007326	007336	007372	007403	TSFCOD:	.WORD	1#,2#,3#,4#
1681	007336	111	156	164	1#:	.ASCIZ	'Internal Diagnostic Failure'
1682	007372	122	145	163	2#:	.ASCIZ	'Reserved'
1683	007403	102	165	163	3#:	.ASCIZ	'Bus Interface or Sanity Check Error'
1684	007447	122	145	163	4#:	.ASCIZ	'Reserved'
1685						.EVEN	



PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET

```

1687 .SBTTL PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET
1688
1689 ;*
1690 ;THIS ROUTINE PRINTS THE ADDRESS AND CONTENTS OF A COMMAND PACKET.
1691 ;THIS ROUTINE IS NORMALLY ONLY CALLED FROM A PRINT ROUTINE.
1692
1693 ;INPUT:
1694
1695 ; R0 NUMBER OF WORDS IN PACKET
1696 ; R3 HIGH ORDER COMMAND PACKET ADDRESS
1697 ; R4 ADDRESS OF COMMAND PACKET
1698
1699 ; NOTE: R3 IS IGNORED IF THE KTENABLE FLAG IS CLEAR.
1700 ; -
1701
1702 007460 PRIPKT:: SAVREG ;SAVE THE REGISTERS
1703 007460 MOV R0,R5 ;SAVE NO. OF WORDS IN PACKET
1704 007464 010005 TST KTENABLE ;ABOVE 28K UNDER TEST?
1705 007466 C05737 003124 BNE 10$ ;BR IF YES
1706 007472 001001 CLR R3 ;SET HIGH ORDER ADDRESS TO 0
1707 007474 005003 10$: MOV R3,R1 ;COPY HIGH ORDER ADDRESS
1708 007476 010301 MOV R4,R0 ;GET LOWER ADDRESS
1709 007500 010400 ROL R0 ;SHIFT BIT 15 INTO C BIT
1710 007502 006100 ROL R1 ;AND INTO HIGH ORDER.
1711 007504 006101 PRINTB #PKTADD,R1,R4 ;PRINT PACKET ADDRESS
1712 007506 010446 MOV R4,-(SP)
1713 007510 010146 MOV R1,-(SP)
1714 007512 012746 007644 MOV #PKTADU,-(SP)
1715 007516 012746 000003 MOV #3,-(SP)
1716 007522 010600 MOV SP,R0
1717 007524 104414 TRAP C#PNTB
1718 007526 062706 000010 ADD #10,SP
1719 007532 010300 15$: MOV R3,R0 ;GET HIGH ORDER ADDRESS
1720 007534 001404 BEQ 20$ ;BR IF NOT ABOVE 28K.
1721 007536 010401 MOV R4,R1 ;GET LOW ORDER ADDRESS
1722 007540 004737 017406 JSR PC,SETMAP ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
1723 007544 010004 MOV R0,R4 ;GET RETURNED PAR6 ADDRESS BIAS
1724 007546 005001 20$: CLR R1 ;SAVE WORD NUMBER
1725 007550 012402 25$: MOV (R4)+,R2 ;GET PACKET CONTENTS
1726 007552 PRINTB #PKTFRM,R1,R2 ;PRINT THE DATA
1727 007552 010246 MOV R2,-(SP)
1728 007554 010146 MOV R1,-(SP)
1729 007556 012746 007606 MOV #PKTFRM,-(SP)
1730 007562 012746 000003 MOV #3,-(SP)
1731 007566 010600 MOV SP,R0
1732 007570 104414 TRAP C#PNTB
1733 007572 062706 000010 ADD #10,SP
1734 007576 005201 INC R1 ;NEXT WORD NUMBER
1735 007600 020105 CMP R1,R5 ;DONE ALL PACKET WORDS?
1736 007602 002762 BLT 25$ ;LOOP TILL ALL DONE
1737 007604 000207 RTS ;RETURN
1738
1739 007606 045 116 045 PKTFRM: .ASCIZ '#N#A Packet Word #D1#A = #06#'
1740 007644 045 116 045 PKTADD: .ASCIZ '#N#A Packet Address = #01#05#'
1741 .EVEN

```

PRIBXOR - PRINT EXPD, RECV AND XOR BYTE

```

1730 .SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR BYTE
1731
1732 ;*
1733 ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE DATA BYTE
1734 ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
1735 ;
1736 ;INPUTS:
1737 ;
1738 ; R1 RECEIVED DATA
1739 ; R2 EXPECTED DATA
1740 ;
1741 ;OUTPUT:
1742 ;
1743 ; R0 XOR OF EXPECTED/RECEIVED DATA
1744 ;-
1745 PRIBXOR::
1746 SAVREG ;SAVE THE REGISTERS
1747 MOV R2,R3 ;EXPECTED DATA
1748 XOR R1,R3 ;FORM THE EXCLUSIVE OR
1749 MOV #C<377>,R0 ;BYTE MASK
1750 BIC R0,R1 ;SAVE LOW BYTE RECV
1751 BIC R0,R2 ;SAVE LOW BYTE EXPD
1752 BIC R0,R3 ;SAVE LOW BYTE XOR
1753 PRINTB @XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
1754 MOV R3,-(SP)
1755 MOV R1,-(SP)
1756 MOV R2,-(SP)
1757 MOV @XORBFOR,-(SP)
1758 MOV #4,-(SP)
1759 MOV SP,R0
1760 TRAP C#PNTB
1761 ADD #12,SP
1762 MOV R3,R0 ;R0 HAS XOR ON RETURN
1763 RTS PC ;RETURN TO CALLER
1764
1765 .ASCIZ '#N#A EXPD: #03#A RECV: #03#A XOR: #03'
1766 .EVEN
1767 .SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR
1768 ;*
1769 ;PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
1770 ;THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
1771 ;
1772 ;INPUTS:
1773 ;
1774 ; R1 RECEIVED DATA
1775 ; R2 EXPECTED DATA
1776 ;
1777 ;OUTPUT:
1778 ;
1779 ; R0 XOR OF EXPECTED/RECEIVED DATA
1780 ;-
1781 PRIBXOR::
1782 SAVREG ;SAVE THE REGISTERS
1783 MOV R2,R3 ;EXPECTED DATA
1784 XOR R1,R3 ;FORM THE EXCLUSIVE OR
1785 PRINTB @XORFOR,R2,R1,R3 ;PRINT THE MESSAGE

```

```

1745 007702
1746 007702
1747 007706 010203
1748 007710
1749 007720 012700 177400
1750 007724 040001
1751 007726 040002
1752 007730 040003
1753 007732
1754 007732 010346
1755 007734 010146
1756 007736 010246
1757 007740 012746 007764
1758 007744 012746 000004
1759 007750 010600
1760 007752 104414
1761 007754 062706 000012
1762 007760 010300
1763 007762 000207
1764
1765 007764 045 116 045
1766
1767
1768
1769
1770
1771
1772
1773
1774 010032
1775 010032
1776 010036 010203
1777 010040
1778 010050

```

PRIXOR - PRINT EXPD, RECV AND XOR

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			
1779	010076	010300			MOV	R3,R0			;R0 HAS XOR ON RETURN
1780	010100	000207			RTS	PC			;RETURN TO CALLER
1781									
1782	010102	045	116	045	XORFOR:	.ASCIZ	'#N#A EXPD: #06#A RECV: #06#A XOR: #06'		
1783						.EVEN			

PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT

```

1785 .SBTTL PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT
1786
1787 ;*
1788 ;
1789 ;ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
1790 ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
1791 ;
1792 ;INPUTS:
1793 ;
1794 ; R0 OCTAL VALUE TO CONVERT
1795 ; R1 TABLE OF POINTERS TO ASCII EQUIVALENT
1796 ;
1797 ;-
1798
1799 PRIEQU: SAVREG ;SAVE THE REGISTERS
1800 010150 RTS PC ;RETURN TO CALLER
1801 010154 000207
1802
1803 .SBTTL PRIRAM - PRINT RAM ADDRESS
1804
1805 ;*
1806 ;
1807 ;PRINT CONTROLLER RAM ADDRESS.
1808 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
1809 ;
1810 ;INPUTS:
1811 ;
1812 ; R4 RAM ADDRESS
1813 ;-
1814 PRIRAM: SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
1815 010156 PRINTB #RAMFOR,R4 ;PRINT RAM ADDRESS IN ERROR
1816 010162 MOV R4,-(SP)
1817 010164 010446 MOV #RAMFOR,-(SP)
1818 010170 012746 010206 MOV #2,-(SP)
1819 010174 012746 000002 MOV SP,R0
1820 010176 010600 TRAP C#PNTB
010200 062706 000006 ADD #6,SP
010204 000207 RTS PC ;RETURN
1819 010206 045 116 045 RAMFOR: .ASCIZ '#N#A CONTROLLER RAM ADDRESS = #06'
1820 .EVEN

```

PRIADD - PRINT MEMORY ERROR ADDRESS

```

1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834 010250
1835 010250
1836 010254 013700 002226
1837 010260 013701 002230
1838 010264 010102
1839 010266 006101
1840 010270 C06100
1841 010272
      010272 010246
      010274 010046
      010276 012746 010320
      010302 012746 000003
      010306 010600
      010310 104414
      010312 062706 000010
1842 010316 000207
1843
1844 010320 045 116 045
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859 010364
1860 010364
1861 010370 013702 002226
1862 010374 013701 002230
1863
1864
1865
1866 010400
      010400 010146
      010402 012746 010446
      010406 012746 000002
      010412 010600
      010414 104414
    
```

```

.SBTTL PRIADD - PRINT MEMORY ERROR ADDRESS
;+
;PRINT MEMORY ADDRESS
;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
;
; IMPLICIT INPUTS
;
; ERRHI - HIGH ORDER ADDRESS
; ERRLO - LOW ORDER ADDRESS
;
;-
PRIADD:
  SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
  MOV ERRHI,R0                          ;GET HIGH ADDRESS
  MOV ERRLO,R1                          ;GET LOW ADDRESS
  MOV R1,R2                             ;COPY LOW ADDRESS
  ROL R1                                 ;SHIFT BIT 15 TO C BIT
  ROL R0                                 ;SHIFT INTO HIGH ORDER
  PRINTB #PRIA0,R0,R2                   ;PRINT MEMORY ADDRESS IN ERROR
  MOV R2,-(SP)
  MOV R0,-(SP)
  MOV #PRIA0,-(SP)
  MOV #3,-(SP)
  MOV SP,R0
  TRAP C#PNTB
  ADD #10,SP
  RTS PC                                ;RETURN

045 PRIA0: .ASCIZ 'NONA MEMORY ERROR ADDRESS = #01#05'
          .EVEN

.SBTTL PRITADD - PRINT MEMORY TEST ADDRESS
;+
;PRINT MEMORY ADDRESS
;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
;
; IMPLICIT INPUTS
;
; ERRHI - HIGH ORDER ADDRESS
; ERRLO - LOW ORDER ADDRESS
;
;-
PRITADD:
  SAVREG                                ;SAVE R1-R5 UNTIL NEXT RETURN
  MOV ERRHI,R2                          ;GET HIGH ADDRESS
  MOV ERRLO,R1                          ;GET LOW ADDRESS
  ;MOV R1,R2                             ;COPY LOW ADDRESS
  ;ROL R1                                 ;SHIFT BIT 15 TO C BIT
  ;ROL R0                                 ;SHIFT INTO HIGH ORDER
  PRINTB #PRITO,P1                      ;PRINT MEMORY ADDRESS LOW IN ERROR
  MOV R1,-(SP)
  MOV #PRITO,-(SP)
  MOV #2,-(SP)
  MOV SP,R0
  TRAP C#PNTB
    
```

PRITADD - PRINT MEMORY TEST ADDRESS

1867	010416	062706	000006							
	010422			ADD	#6,SP					
	010422	010246		PRINTB	#PRIT1,R2					;PRINT MEMORY ADDRESS HIGH IN ERROR
	010424	012746	010511	MOV	R2,-(SP)					
	010430	012746	000002	MOV	#PRIT1,-(SP)					
	010434	010600		MOV	#2,-(SP)					
	010436	104414		MOV	SP,R0					
	010440	062706	000006	TRAP	C#PNTB					
1868	010444	000207		ADD	#6,SP					
1869				RTS	PC					;RETURN
1870	010446	045	116	045	PRIT0:	.ASCIZ	'#N#A MEMORY TEST ADDRESS LOW = #06'			
1871	010511	045	116	045	PRIT1:	.ASCIZ	'#N#A MEMORY TEST ADDRESS HIGH = #06'			
1872						.EVEN				

SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND

```

1874 .SBTTL SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND
1875
1876 ;*
1877 ;
1878 ;ROUTINE TO ISSUE A SPACE RECORDS
1879 ;COMMAND (FORWARD OR REVERSE)
1880 ;
1881 ;INPUT:
1882 ;
1883 ; R3 NUMBER OF RECORDS TO BE SPACED OVER
1884 ; BIT15 CONTROLS DIRECTION
1885 ; BIT15 = 0 IS FORWARD
1886 ; BIT15 = 1 IS REVERSE
1887 ; R5 FIRST DEVICE UNIBUS ADDRESS
1888 ;
1889 ; REQUIRES A WRITE CHARACTERISTICS DONE PREVIOUSLY
1890 ;
1891 ;OUTPUT:
1892 ;
1893 ; CARRY SET - SPACE RECORDS COMMAND OK
1894 ; CLR - SPACE RECORDS FAILED
1895 ;
1896 ;
1897 ; R0 THE CONTENTS OF R4 IS MOVED TO R0
1898 ;
1899 ;
1900 ;IMPLICIT OUTPUT:
1901 ;
1902 ; TAPE HAS BEEN MOVED
1903 ;
1904 ;SIDE EFFECTS:
1905 ;
1906 ;
1907 ;-
1908
1909 SPACE::
1910 010556 SAVREG ;SAVE THE GENERAL REGISTERS
1911 010556 MOV #500.,SDELAY ;SET UP DELAY
1912 010562 012737 000764 010750 MOV #140010.80# ;SET UP COMMAND, SPACE FORWARD
1913 010570 012737 140010 010740 TST R3 ;CHECK FOR DIRECTION
1914 010576 005703 BMI 5# ;BR, IF REVERSE INDICATED
1915 010602 010337 010742 MOV R3,90# ;LOAD UP NUMBER OF RECORDS TO SPACE
1916 010606 000407 BR 10# ;GO DO COMMAND
1917 010610 042703 100000 5#: BIC #BIT15,R3 ;CLEAR DIRECTION BIT
1918 010614 010337 010742 MOV R3,90# ;LOAD UP NUMBER OF RECORDS TO SPACE
1919 010620 052737 000400 010740 BIS #BIT8,80# ;SET REVERSE BIT IN COMMAND PACKET
1920 010626 012704 010740 10#: MOV #80#,R4 ;SET UP R4 WITH PACKET ADDRESS
1921 010632 010465 000000 MOV R4,TSDB(R5) ;SEND OUT COMMAND
1922 010636 004737 016340 15#: JSR PC,WAITF ;WAIT FOR SSR
1923 010642 103420 BCS 20# ;BR, IF SSR IS SET AND OK
1924 010644 DELAY 250 ;DELAY ABOUT .25 SECONDS
010644 012727 000250 MOV #250,(PC)+
010650 000000 .WORD 0
010652 013727 002116 MOV L#DLY,(PC)+
010656 000000 .WORD 0
010660 005367 177772 DEC -6(PC)
010664 001375 BNE .-4

```

SPACE    -    SPACE RECORDS (FORWARD AND REVERSE) COMMAND

```

010666 005367 177756            DEC    -22(PC)
010672 001367            BNE    .-20
1925 010674 005337 010750        DEC    SDELAY            ;BUMP DELAY COUNTER DOWN
1926 010700 001356            BNE    15#            ;BR, IF MORE DELAY
1927 010702 000411            BR     60#            ;BR IF TROUBLE CARRY = CLEAR
1928 010704 016501 000002        20#:    MOV    TSSR(R5),R1        ;READ TSSR
1929 010710 012702 000200        MOV    #SSR,R2        ;SET UP EXPECTED
1930 010714 020201        25#:    CMP    R2,R1            ;ARE THEY OK
1931 010716 001401            BEQ    40#            ;BR, IF EQUAL = OK
1932 010720 000402            BR     60#            ;TROUBLE EXIT
1933 010722 000261        40#:    SEC                    ;SET CARRY NO TROUBLE
1934 010724 000401            BR     70#            ;EXIT
1935 010726 000241        60#:    CLC                    ;CARRY CLEAR = ERROR
1936 010730            70#:                   
1937 010730 010400            MOV    R4,R0            ;PASS PACKET ADDRESS
1938 010732 000207            RTS    PC              ;RETURN
1939                    ;
1940                    ;
1941                    ;
1942                    ;PACKET FOR SPACE COMMAND
1943                    ;
1945                    010740            .=<. +10>&177770
1947                    ;
1948                    ;COMMAND WORD
1949 010740 000000        80#:    .WORD                ;NUMBER OF RECORDS TO BE SPACED OVER WORD
1950                    ;
1951 010742 000000        90#:    .WORD                ;
1952 010744 000000            .WORD                ;
1953 010746 000000            .WORD                ;
1954 010750 000000        SDELAY: .WORD    0                    ;DELAY COUNTER
1955                    .EVEN                ;
1956                    .SBTTL    WRTCHR    - WRITE CHARACTERISTICS COMMAND

```



WRTCHR - WRITE CHARACTERISTICS COMMAND

```

1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981 010752
1982 010752
1983 010756 005037 002220
1984 010762 005037 002216
1985 010766 010465 000000
1986 010772 004737 016426
1987 010776 103401
1988 011000 000435
1989 011002 016501 000002
1990 011006 012702 000200
1991 011012 032701 000100
1992 011016 001402
1993 011020 052702 000100
1994 011024 020201
1995 011026 001401
1996 011030 000421
1997 011032 062704 000010
1998 011036 011403
1999 011040 032763 000200 000012
2000 011046 001402
2001 011050 005237 002216
2002 011054
2003 011054 032763 000100 000012
2004 011062 001402
2005 011064 005237 002220
2006 011070
2007 011070 000261
2008 011072 000401
2009 011074 000241
2010 011076 016500 000002
2011 011102 000207

```

```

;
;ROUTINE TO ISSUE A WRITE CHARACTERISTICS
;COMMAND SO THAT OTHER COMMANDS WILL BE ACCEPTED
;
;INPUT:
;   R4      ADDRESS OF PACKET FROM TEST
;   R5      FIRST DEVICE UNIBUS ADDRESS
;           REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
;
;OUTPUT:
;   R0      TSSR CONTENTS
;   CARRY   SET - WRITE CHARACTERISTICS COMMAND OK
;           CLR - WRITE CHARACTERISTICS FAILED
;
;IMPLICIT OUTPUT:
;
;   MESSAGE BUFFER AND OTHER BUFFERS ALL SET UP
;   SOFTWARE SWITCHES SET AS FOLLOWS:
;       EXTFEA = EXTENDED FEATURES PRESENT
;       BENBSW = BUFFER ENABLE SWITCH ON OR OFF
;
;SIDE EFFECTS:
;-
WRTCHR::
    SAVREG                ;SAVE THE GENERAL REGISTERS
    CLR BENBSW            ;CLEAR BUFFER ENABLE SWITCH
    CLR EXTFEA           ;CLEAR EXTENDED FEATURES SW SWITCH
10:  MOV R4,TSDB(R5)      ;SEND OUT COMMAND
    JSR PC,CHKTSSR       ;WAIT FOR SSR
    BCS 20:              ;BR, IF SSR IS SET AND OK
    BR 60:               ;BR IF TROUBLE CARRY = CLEAR
20:  MOV TSSR(R5),R1     ;READ TSSR
    MOV #SSR,R2          ;SET UP EXPECTED
    BIT #OFL,R1          ;WAS OFF LINE SET IN TSSR
    BEQ 25:              ;BR, IF NO OFL SET
    BIS #OFL,R2          ;MAKE THEM LOOK ALIKE
25:  CMP R2,R1           ;ARE THEY OK
    BEQ 40:              ;BR, IF EQUAL = OK
    BR 60:               ;TROUBLE EXIT
40:  ADD #8,R4           ;POINT TO WRT CHARA DATA PACKET
    MOV (R4),R3          ;GET ADDRESS OF MESSAGE BUFFER
    BIT #X2.EXTF,XST2(R3);EXTENDED FEATURES BIT SET?
    BEQ 45:              ;BR IF NO
    INC EXTFEA           ;SET EXTENDED FEATURES SW SWITCH
45:  BIT #X2.BUFE,XST2(R3);BUFFER ENABLE SWITCH SET
    BEQ 50:              ;BR, IF SWITCH NOT SET
    INC BENBSW          ;SET SOFTWARE SWITCH FOR ENABLED
50:  SEC                ;SET CARRY NO TROUBLE
    BR 70:              ;EXIT
60:  CLC                ;CARRY CLEAR = ERROR
70:  MOV TSSR(R5),R0    ;RETURN TSSR CONTENTS
    RTS PC              ;RETURN

```

REWIND - POSITION TAPE (REWIND) COMMAND

```

2013 .SBTTL REWIND - POSITION TAPE (REWIND) COMMAND
2014
2015 ;*
2016 ; THIS ROUTINE WILL REWIND THE SELECTED TAPE.
2017
2018 ; CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
2019 ; TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
2020 ; SSR TO SET IN THE TSSR
2021
2022 ;
2023 ; CALLING SEQUENCE:
2024
2025 ; DO A SOFT INIT
2026 ; DO A WRITE CHARACTERISTICS
2027 ; JSR PC,REWIND
2028
2029 ; INPUT:
2030
2031 ; R5 FIRST DEVICE UNIBUS ADDRESS
2032
2033 ;
2034 ; OUTPUT
2035
2036 ; R0 THE CONTENTS OF R4 IS PASSED TO R0
2037
2038 ;
2039 ;-
2040 REWIND::
2041 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2042 MOV #RMPACK,R4 ;GET PACKET ADDRESS
2043 MOV R4,TSDB(R5) ;SEND PACKET ADDRESS TO EXECUTE
2044 MOV #360,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND
2045 104: JSR PC,WAITF ;WAIT FOR SSR TO SET
2046 BCS 204 ;LEAVE WHEN SSR IS SET
2047 DELAY 250 ;WAIT FOR .25 SECONDS
2048 MOV #250,.(PC)+
2049 .WORD 0
2050 MOV L#DLY,.(PC)+
2051 .WORD 0
2052 DEC -6(PC)
2053 BNE -4
2054 DEC -22(PC)
2055 BNE -20
2056 DEC R3 ;BUMP COUNTER DOWN
2057 BNE 104 ;KEEP GOING
2058 CLC ;CLEAR CARRY TO SET ERROR
2059 204: MOV R4,R0 ;PASS THE PACKET ADDRESS
RTS PC ;RETURN
RMPACK: .=<.10>&177770
;POSTION COMMAND (REWIND)
;NOT USED

```

CKRAM - COMPARE RAM TO I/O PACKET

2061 .SBTTL CKRAM - COMPARE RAM TO I/O PACKET

2062 ;  
 2063 ;  
 2064 ;ROUTINE TO READ THE FIRST 8 BYTES FROM RAM  
 2065 ;MEMORY AND COMPARE THIS DATA TO A COMMAND PACKET.  
 2066 ;

2067 ;INPUT:  
 2068 ;

2069 ; R4 ADDRESS OF THE COMMAND PACKET  
 2070 ; R5 FIRST DEVICE UNIBUS ADDRESS  
 2071 ;

2072 ;OUTPUT:  
 2073 ;

2074 ; CARRY SET - RAM MATCHES PACKET  
 2075 ; CLR - RAM DOES NOT MATCH PACKET  
 2076 ;

2077 ;IMPLICIT OUTPUT:  
 2078 ;

2079 ; THE TABLE RAMDATA IS FILLED WITH THE  
 2080 ; DATA HELD IN RAM.  
 2081 ; RAMSIZ IS SET TO 8. FOR PRAMPKT ROUTINE  
 2082 ;

2083 ;SIDE EFFECTS:  
 2084 ;

2085 ; THE SUBSYSTEM IS LEFT IN MAINTENANCE MODE  
 2086 ;  
 2087 ;-  
 2088 ;

2089	011204				CKRAM::	SAVREG		;SAVE THE GENERAL REGISTERS
2090	011204					MOV	#RAMDATA,R1	;ADDRESS TO SAVE THE RAM DATA
2091	011210	012701	002232			MOV	#RMPKTBEG,R2	;BYTE ADDRESS OF FIRST RAM DATA
2092	011214	012702	000201			CLR	R3	;CLEAR THE ERROR FLAG
2093	011220	005003				JSR	PC,CHKTSSR	;WAIT FOR SSR
2094	011222	004737	016426			MOVB	#0,TSDB(R5)	;SET MAINTENANCE MODE
2095	011226	112765	000000	000000	10#:	JSR	PC,CHKTSSR	;WAIT FOR SSR TO SET
2096	011234	004737	016426			MOV	R2,TSDB(R5)	;SELECT NEXT RAM ADDRESS
2097	011240	010265	000000			JSR	PC,CHKTSSR	;WAIT FOR SSR TO SET
2098	011244	004737	016426			MOVB	TSBA(R5),(R1)	;READ THE RAM DATA
2099	011250	116511	000000			CMPB	(R1)+,(R4)+	;COMPARE TO EXPECTED
2100	011254	122124				BEQ	20#	;BRANCH IF OK
2101	011256	001401				INC	R3	;SET ERROR FLAG
2102	011260	005203			20#:	INC	R2	;ADDRESS OF NEXT RAM LOCATION
2103	011262	005202				CMP	R2,#RMPKTEND	;REACHED END YET ?
2104	011264	020227	000210			BLE	10#	;BRANCH TILL ALL READ
2105	011270	003761				TST	R3	;WAS AN ERROR FOUND ?
2106	011272	005703				BEQ	30#	;BRANCH IF NOT
2107	011274	001402				CLC		;CLEAR CARRY TO SHOW ERROR
2108	011276	000241				BR	50#	;AND EXIT
2109	011300	000401			30#:	SEC		;SHOW GOOD COMPARE
2110	011302	000261			50#:	MOV	#8,.RAMSIZ	;SETUP RAMSIZ FOR PRAMPKT ROUTINE
2111	011304	012737	000010	002272		RTS	PC	;RETURN
2112	011312	000207						

CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA

```

2114 .SBTTL CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
2115 ;*
2116 ;ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
2117 ;MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
2118 ;
2119 ;INPUT:
2120 ;
2121 ; R4 ADDRESS OF THE CHARACTERISTICS DATA
2122 ; R5 FIRST DEVICE UNIBUS ADDRESS
2123 ;
2124 ;OUTPUT:
2125 ;
2126 ; CARRY SET - RAM MATCHES PACKET
2127 ; CLR - RAM DOES NOT MATCH PACKET
2128 ;
2129 ;IMPLICIT OUTPUT:
2130 ;
2131 ; THE TABLE RAMDATA IS FILLED WITH THE
2132 ; DATA HELD IN RAM.
2133 ; RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
2134 ;
2135 ;SIDE EFFECTS:
2136 ;
2137 ; THE SUBSYSTEM IS LEFT IN MAINTENANCE MODE
2138 ;
2139 ;
2140 CKRAM2::
2141 SAVREG ;SAVE THE GENERAL REGISTERS
2142 MOV #RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
2143 MOV #RMCHBEG,R2 ;BYTE ADDRESS OF FIRST RAM DATA
2144 CLR R3 ;CLEAR THE ERROR FLAG
2145 JSR PC,CHKTSSR ;WAIT FOR SSR
2146 MOVB #0,TSDB(R5) ;SET MAINTENANCE MODE
2147 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2148 MOV R2,TSDB(R5) ;SELECT NEXT RAM ADDRESS
2149 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
2150 MOVB TSBA(R5),(R1) ;READ THE RAM DATA
2151 CMPB (R1),.(R4) ;COMPARE TO EXPECTED
2152 BEQ 20$ ;BRANCH IF OK
2153 INC R3 ;SET ERROR FLAG
2154 INC R2 ;ADDRESS OF NEXT RAM LOCATION
2155 MOV #8.,RAMSIZ ;ASSUME EXTFEA NOT SET
2156 TST EXTFEA ;IS THE SOFTWARE EXTENDED FEATURES SET
2157 BEQ 25$ ;BR, IF NOT SET
2158 MOV #10.,RAMSIZ ;SET RAMSIZ FOR EXTEND FEATURES
2159 CMP R2,#RMCHEND ;AT END OF EXTENDED BUFFER
2160 BLE 10$ ;BR, IF NOT AT END YET
2161 BR 27$ ;AT END BRANCH
2162 CMP R2,#RMCHEND-2 ;REACHED END YET ?
2163 BLE 10$ ;BRANCH TILL ALL READ
2164 TST R3 ;WAS AN ERROR FOUND ?
2165 BEQ 30$ ;BRANCH IF NOT
2166 CLC ;CLEAR CARRY TO SHOW ERROR
2167 BR 50$ ;AND EXIT
2168 SEC ;SHOW GOOD COMPARE
2169 BR 50$ ;RETURN

```

## CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS

```

2171          .SBTTL CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS
2172          ;*
2173          ;
2174          ;ROUTINE TO COMPARE A WRITE CHARACTERISTICS EXPD AND RECV
2175          ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
2176          ;ERROR PRINT ROUTINES.
2177          ;
2178          ;INPUT:
2179          ;
2180          ;      R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
2181          ;      R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
2182          ;      R2      EXPD MESSAGE BUFFER ADDRESS
2183          ;
2184          ;OUTPUT:
2185          ;
2186          ;      CARRY   SET - MESSAGE BUFFERS MATCH
2187          ;              CLR MESSAGE BUFFERS DON'T MATCH
2188          ;
2189          ;IMPLICIT OUTPUT:
2190          ;
2191          ;      EXPMSG   BUFFER IS SET TO EXPD DATA
2192          ;      RECVMSG  BUFFER IS SET TO RECV DATA
2193          ;      RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
2194          ;      RCVLOADD SET TO LOW ORDER ADDRESS OF RECV
2195          ;
2196          ;-
2197          CKMSG:  SAVREG          ;SAVE R1-R5 UNTIL NEXT RETURN
2198          ;
2199          ;      MOV      R0,RCVHIADD ;SAVE RECV HIGH ADDRESS
2200          ;      MOV      R1,RCVLOAD ;SAVE RECV LOW ADDRESS
2201          ;      TST     KTENABLE    ;TESTING ABOVE 28K?
2202          ;      BEQ     10$         ;BR IF NO
2203          ;      JSR     PC,SETMAP   ;RETURN ADDRESS BIASED TO PAR6 IN R0
2204          ;      MOV     R0,R1      ;GET RETURNED ADDRESS BIASED TO PAR6
2205          ;      CLR     R4         ;WORD IN BUFFER
2206          ;      CLR     R3         ;CLEAR ERROR SEEN FLAG
2207          ;      MOV     R2,R5     ;GET EXPD BUFFER ADDRESS
2208          ;      MOV     (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
2209          ;      MOV     (R1),RECVMSG(R4) ;SAVE RECV FOR ERROR REPORT
2210          ;      CMP     (R2)*,(R1)* ;EXPD EQUAL RECV?
2211          ;      BEQ     25$         ;BR IF YES
2212          ;      INC     R3         ;SET ERROR SEEN FLAG
2213          ;      ADD     @2,R4      ;POINT TO NEXT WORD ADDRESS
2214          ;      CMP     R4,@14     ;DONE FIRST 7 WORDS?
2215          ;      BLE     15$         ;BR IF NO
2216          ;      BIT     @X2.EXTF,XST2(R5) ;IS EXTENDED FEATURES SET IN EXPD?
2217          ;      BFO     50$         ;BR IF NO
2218          ;      CMF     R4,@16     ;DONE EXTENDED FEATURES WORD?
2219          ;      BLE     15$         ;BR IF NO
2220          ;      TST     R3         ;ANY ERRORS SEEN?
2221          ;      BEQ     55$         ;BR IF NO
2222          ;      CLC     CLC        ;SET FAILURE
2223          ;      BR     60$         ;
2224          ;      SEC     SEC        ;SET SUCCESS
2225          ;      RTS     PC         ;RETURN

```

CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2226 .SBTTL CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS
2227 ;*
2228 ;ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
2229 ;BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
2230 ;ERROR PRINT ROUTINES.
2231 ;
2232 ;INPUT:
2233 ; R0 RECV MESSAGE BUFFER HIGH ORDER ADDRESS
2234 ; R1 RECV MESSAGE BUFFER LOW ORDER ADDRESS
2235 ; R2 EXPD MESSAGE BUFFER ADDRESS
2236 ; R3 NUMBER OF BYTES TO COMPARE
2237 ;
2238 ;OUTPUT:
2239 ; CARRY SET - MESSAGE BUFFERS MATCH
2240 ; CLR - MESSAGE BUFFERS DON'T MATCH
2241 ;
2242 ;IMPLICIT OUTPUT:
2243 ; EXPMSG BUFFER IS SET TO EXPD DATA
2244 ; RECVMSG BUFFER IS SET TO RECV DATA
2245 ; RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
2246 ; RCVLOAD SET TO LOW ORDER ADDRESS OF RECV
2247 ;
2248 CKMSG2::
2249 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2250 CMP R3,#RECVMSG-EXPMSG;#00 IS COUNT ABOVE MAX ALLOWED?
2251 BLE 5# ;#00 BR IF NO
2252 MOV #RECVMSG-EXPMSG,R3;#00
2253 PRINTF #DEBUGMSG ;#00
2254 MOV #DEBUGMSG,-(SP)
2255 MOV #1,-(SP)
2256 MOV SP,R0
2257 TRAP C#PNTF
2258 ADD #4,SP
2259 5# : MOV R0,RCVHIADD ;SAVE RECV HIGH ADDRESS
2260 MOV R1,RCVLOAD ;SAVE RECV LOW ADDRESS
2261 TST KTENABLE ;TESTING ABOVE 28K?
2262 BEQ 10# ;BR IF NO
2263 JSR PC,SETMAP ;RETURN ADDRESS BIASED TO PAR6 IN R0
2264 MOV R0,R1 ;GET RETURNED ADDRESS BIASED TO PAR6
2265 10# : CLR R4 ;WORD IN BUFFER
2266 CLR R5 ;CLEAR ERROR SEEN FLAG
2267 15# : MOVB (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
2268 MOVB (R1),RECVMSG(R4) ;SAVE RECV FOR ERROR REPORT
2269 CMPB (R2)+,(R1)+ ;EXPD EQUAL RECV?
2270 BEQ 25# ;BR IF YES
2271 INC R5 ;SET ERROR SEEN FLAG
2272 25# : ADD #1,R4 ;POINT TO NEXT BYTE
2273 CMP R4,R3 ;DONE ALL BYTES?
2274 BGE 50# ;BR IF YES
2275 BR 15# ;DO NEXT BYTE
2276 50# : TST R5 ;ANY ERRORS SEEN?
2277 BEQ 55# ;BR IF NO
2278 CLC ;SET FAILURE
2279 BR 60# ;
2280 55# : SEC ;SET SUCCESS
2281 60# : RTS PC ;RETURN

```

CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2278 011722      120      122      117 DEBUGMSG: .ASCIZ 'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-';000
2279 012012      045      116      045 FERCM:  .ASCII /NMA ***/
2280 012023      040      040      124 ERCH:  .ASCIZ / TSSR ERROR CODE REC'D = /
2281 012056      056      056      056 SIMSG:  .ASCIZ /... AFTER DOING SOFT INIT/
2282 012111      124      105      123 TINERR: .ASCIZ /TEST: .../
2283                                     .EVEN
2284                                     ;*
2285                                     ;
2286                                     ;PRINT ROUTINE TO FATAL SOFT INIT ERRORS
2287                                     ;
2288                                     ;INPUT:
2289                                     ;
2290                                     ;      R1      CONTENTS OF TSSR AT ERROR
2291                                     ;
2292                                     ;SIDE EFFECTS:
2293                                     ;
2294                                     ;      EXECUTES DROP UNIT TO CEASE TESTING
2295                                     ;
2296                                     ;-
2297
2298 012124      BGNMSG  SFMSG
2299 012124      004737  006022  SFMSG:: JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2300 012130      004737  017272      JSR      PC,CKDROP        ;DROP UNIT, IF ALLOWED
2301 012134      ENDMMSG
2302 012134      104423  L10003: TRAP      C$MSG
2303
2304                                     ;*
2305                                     ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2306                                     ;TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
2307                                     ;
2308                                     ;INPUTS:
2309                                     ;
2310                                     ;      R1      TSSR CONTENTS
2311                                     ;      R4      ADDRESS OF COMMAND PACKET
2312                                     ;
2313                                     ;-
2314 012136      BGNMSG  PKTSSR
2315 012136      004737  006022  PKTSSR:: JSR      PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2316 012142      012700  000004      MOV      #4,R0            ;NO. OF WORDS IN PACKET
2317 012146      004737  007460      JSR      PC,PRIPKT        ;PRINT THE CONTENTS OF COMMAND PACKET
2318 012152      ENDMMSG
2319 012152      104423  L10004: TRAP      C$MSG

```

CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2320
2321 ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2322 ;TSSR AND A GET STATUS COMMAND PACKET.
2323
2324 ;INPUTS:
2325
2326 ; R1 TSSR CONTENTS
2327 ; R4 ADDRESS OF COMMAND PACKET
2328
2329 012154 BGNMSG PKTGETS
012154 PKTGETS:
2330 012154 004737 006022 JSR PC,PRITSSR ;PRINT THE CONTENTS OF TSSR REGISTER
2331 012160 012700 000002 MOV #2,R0 ;NO. OF WORDS IN GET STATUS PACKET
2332 012164 004737 007460 JSR PC,PRIPKT ;PRINT THE CONTENTS OF COMMAND PACKET
2333 012170 ENDMSG
012170
012170 104423
L10005: TRAP C#MSG

2334
2335 ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
2336
2337 ;INPUTS:
2338 ; R1 TSSR CONTENTS
2339 ; R4 ADDRESS OF COMMAND PACKET
2340
2341 012172 BGNMSG SFFMSG
012172 SFFMSG:
2342 012172 004737 006022 JSR PC,PRITSSR ;PRINT CONTENTS OF TSSR REGISTER
2343 012176 ENDMSG
012176
012176 104423
L10006: TRAP C#MSG
.SBTTL PKTMES - PRINT TSSR AND MESSAGE BUFFER

2344
2345 ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
2346 ;BUFFER FOR ERROR REPORTS
2347
2348 ;INPUTS:
2349
2350 ; R1 CONTENTS OF TSSR
2351 ; R2 LOW ORDER MESSAGE BUFFER
2352 ; R3 HIGH ORDER MESSAGE BUFFER ADDRESS
2353 ; NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR
2354
2355 ;-
2356 012200 BGNMSG PKTMES
012200 PKTMES:
2357 012200 004737 006022 JSR PC,PRITSSR ;PRINT CONTENTS OF TSSR
2358 012204 010200 MOV R2,R0 ;LOW ORDER ADDRESS
2359 012206 010301 MOV R3,R1 ;HIGH ORDER ADDRESS
2360 012210 004737 014332 JSR PC,PRMESS ;PRINT THE MESSAGE BUFFER
2361 012214 ENDMSG
012214
012214 104423
L10007: TRAP C#MSG
    
```



ADDSSR - PRINT TEST ADDRESS AND TSSR

```

2363          .SBTTL  ADDSSR  - PRINT TEST ADDRESS AND TSSR
2364          ;*
2365          ;PRINT ROUTINE TO PRINT THE CONTENTS OF
2366          ;TSSR AND A MEMORY TEST ADDRESS
2367          ;
2368          ;INPUTS:
2369          ;
2370          ;      R5      FIRST DEVICE UNIBUS ADDRESS
2371          ;      ERRHI   HIGH ORDER MEMORY TEST ADDRESS
2372          ;      ERRLO   LOW ORDER MEMORY TEST ADDRESS
2373          ;
2374          ;
2375          BGNMSG  ADDSSR
012216 ADDSSR:
2376          JSR    PC,PRITADD      ;PRINT MEMORY TEST ADDRESS
012216          MOV    TSSR(R5),R1   ;GET CURRENT TSSR
2377          JSR    PC,PRITSSR     ;PRINT THE CONTENTS OF TSSR REGISTER
012222          ENDMSG
2378          JSR    PC,PRITSSR     ;PRINT THE CONTENTS OF TSSR REGISTER
012226          ENDMSG
2379          L10010:
012232          TRAP   C#MSG
012232          104423

2380          .SBTTL  MSGEXP  - PRINT WRITE CHAR. EXPD-RCV MESSAGE BUFFERS
2381          ;*
2382          ;PRINT ROUTINE TO PRINT WRITE CHARACTERISTIC MESSAGE BUFFER
2383          ;
2384          ;IMPLICIT INPUTS:
2385          ;
2386          ;      EXPMSG  - EXPECTED MESSAGE BUFFER
2387          ;      RECMSG  - RECEIVED MESSAGE BUFFER
2388          ;      RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2389          ;      RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2390          ;
2391          ;
2392          ;
2393          BGNMSG  MSGEXP
012234 MSGEXP:
2394          MOV    #7,R0           ;ASSUME NO EXT FEATURES
012234          TST   EXTFEA        ;EXT FEATURES SET?
2395          BEQ   S#              ;BR IF NO
012240          MOV    #8.,R0      ;EXT FEATURE BUFFER IS 8 WORDS
2396          JSR    PC,PRMSGEXP    ;PRINT EXPD/RCV MESSAGE BUFFERS
012244          ENDMSG
2397          L10011:
2398          TRAP   C#MSG
012252          104423
2399          012256
012256          104423

```

FIFEXP - PRINT FIFO EXP/RECV DATA

```

2401          .SBTTL  FIFEXP  - PRINT FIFO EXP/RECV DATA
2402          ;
2403          ;PRINT ROUTINE TO PRINT FIFO EXP/RECV DATA
2404          ;
2405          ;      R1      - BYTE COUNT
2406          ;
2407          ;IMPLICIT INPUTS:
2408          ;
2409          ;      EXPMSG  - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY
2410          ;      RECVMSG - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
2411          ;
2412          ;-
2413          BGNMSG  FIFEXP
2414          FIFEXP:
2415          PRINTX  #FIF1MSG,R1      ;PRINT BYTES TRANSFERRED
2416          MOV     R1,-(SP)
2417          MOV     #FIF1MSG,-(SP)
2418          MOV     #2,-(SP)
2419          MOV     SP,R0
2420          TRAP   C#PNTX
2421          ADD     #6,SP
2422          PRINTX  #FIF2MSG      ;PRINT HEADER MSG
2423          MOV     #FIF2MSG,-(SP)
2424          MOV     #1,-(SP)
2425          MOV     SP,R0
2426          TRAP   C#PNTX
2427          ADD     #4,SP
2428          MOV     R1,R0      ;GET BYTE COUNT
2429          JSR    PC,PRBYTEXP  ;PRINT FIFO BYTES IN ERROR
2430          ENDMSG
2431          L10012:
2432          TRAP   C#MSG
2433          .ASCIZ  '###A NUMBER OF BYTES TRANSFERRED = #D2'
2434          .ASCIZ  '###A FIFO DATA BYTES IN ERROR:'
2435          .EVEN

```

```

2413 012260
2414 012260 010146 012332
2415 012262 012746 000002
2416 012266 012746 000002
2417 012272 C10600
2418 012274 104415
2419 012276 062706 000006
2420 012302
2421 012302 012746 012401
2422 012306 012746 000001
2423 012312 C10600
2424 012314 104415
2425 012316 062706 000004
2426 012322 010100
2427 012324 004737 015212
2428 012330
2429 012330 104423
2430 012332 045 116
2431 012401 045 116
2432

```

MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS

```

2423          .SBTTL MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
2424          ;*
2425          ;
2426          ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
2427          ;
2428          ;
2429          ;IMPLICIT INPUTS:
2430          ;
2431          ;   EXPMSG - EXPECTED MESSAGE BUFFER
2432          ;   RECMMSG - RECEIVED MESSAGE BUFFER
2433          ;   RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2434          ;   RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2435          ;-
2436 012440      BGNMSG MSGSTAT
          MSGSTAT:
2437 012440      012701 012502
          MOV     #STATCOD,R1      ;ASCII ADDRESS TABLE
2438 012444      012100
          MOV     (R1)+,R0         ;DONE ALL MSG LINES?
2439 012446      001410
          BEQ     20$              ;BR IF YES
2440 012450      PRINTX R0         ;PRINT STATUS BIT NAMES
          MOV     R0,-(SP)
          MOV     #1,-(SP)
          MOV     SP,R0
          TRAP   C#PNTX
          ADD     #4,SP
2441 012466      000766
          BR     10$              ;DO ANOTHER MSG LINE
2442 012470      012700 000012
          MOV     #10,R0          ;NUMBER OF WORDS IN A READ STATUS BUFFER
2443 012474      004737 014642
          JSR     PC,PRMSGEXP     ;PRINT EXPD/RCV MESSAGE BUFFERS
2444 012500      ENDMSG
          L10013:
          TRAP   C#MSG
2445
2446 012502      012520 012562 012653 STATCOD: .WORD 1$,2$,3$,4$,5$,6$,0
2447 012520      045 116 045 1$: .ASCIZ '#N$A Tape Bus Signals in Word #8:'
2448 012562      045 116 045 2$: .ASCIZ '#N$A PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>'
2449 012653      045 116 045 3$: .ASCIZ '#N$A IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>'
2450 012744      045 116 045 4$: .ASCIZ '#N$A IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>'
2451 013035      045 116 045 5$: .ASCIZ '#N$A Tape Bus Signals in Word #9:'
2452 013077      045 116 045 6$: .ASCIZ '#N$A DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>'
2453          .EVEN
2454

```

MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS

```

2456 .SBTTL MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
2457 ;*
2458 ;
2459 ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RECV
2460 ;
2461 ;IMPLICIT INPUTS:
2462 ;
2463 ; EXPMSG - EXPECTED MESSAGE BUFFER
2464 ; RECMMSG - RECEIVED MESSAGE BUFFER
2465 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2466 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2467 ;-
2468 013154 BGNMSG MSGLOOP
013154 MSGLOOP::
2469 013154 012701 013216 MOV #LOOPCOD,R1 ;ASCII ADDRESS TABLE
2470 013160 012100 10#: MOV (R1)+,RO ;DONE ALL MSG LINES?
2471 013162 001410 BEQ 20# ;BR IF YES
2472 013164 PRINTX RO ;PRINT STATUS BIT NAMES
013164 C10046 MOV RO,-(SP)
013166 012746 000001 MOV #1,-(SP)
013172 010600 MOV SP,RO
013174 104415 TRAP C#PNTX
013176 062706 000004 ADD #4,SP
2473 013202 000766 BR 10# ;DO ANOTHER MSG LINE
2474 013204 012700 000012 20#: MOV #10,RO ;NUMBER OF WORDS IN A READ STATUS BUFFER
2475 013210 004737 014642 JSR PC,PRMSGEXP ;PRINT EXPD/RECV MESSAGE BUFFERS
2476 013214 ENDMMSG
013214 L10014: TRAP C#MSG
013214 104423

2477 LOOPCOD: .WORD 1#,2#,3#,4#,5#,6#,7#,0
2478 013216 013236 013311 013410 1#: .ASCIZ '#N#A Tape Bus Loopback Signals in Word #8:'
2479 013236 045 116 045 2#: .ASCIZ '#N#A PAREPR<15> IRESV2<14> IRESV1<13>'
2480 013311 045 116 045 3#: .ASCIZ '#N#A IHISP=>IEOT<12> IWRT=>IIDENT<11> IREV =>ICER <10>'
2481 013410 045 116 045 4#: .ASCIZ '#N#A IWFM =>IFMK<09> IEDIT=>IHER <08> IFAD =>ISPEED<07>'
2482 013507 045 116 045 5#: .ASCIZ '#N#A ITADO=>IRDY<06> ITAD1=>IOML <05> IERASE=>ILDPA <04>'
2483 013606 045 116 045 6#: .ASCIZ '#N#A IREW =>IDBY<03> IRWU =>IRWD <02> IFEN =>IFBY <01>'
2484 013705 045 116 045 7#: .ASCIZ '#N#A IGO =>IFPT<00>'
2485 014004 045 116 045 .EVEN
2486

```

MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER

```

2488          .SBTTL MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
2489          ;*
2490          ;PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
2491          ;
2492          ;
2493          ;IMPLICIT INPUTS:
2494          ;
2495          ;     EXPMSG - EXPECTED MESSAGE BUFFER
2496          ;     RECMMSG - RECEIVED MESSAGE BUFFER
2497          ;     RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2498          ;     RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2499          ;
2500          ;-
2501          BGNMSG MSGSUB
MSGSUB:      MOV     #10,R0          ;SIZE OF WRITE SUBSYSTEM BUFFER
2502          JSR     PC,PRMSGEXP   ;PRINT EXPD/RCV MESSAGE BUFFERS
2503          ENDMMSG
2504          L10015: TRAP    C#MSG
2505          .SBTTL MEMADD - PRINT MEMORY ADDRESS DATA ERROR
2506          ;*
2507          ;PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
2508          ;
2509          ;IMPLICIT INPUTS:
2510          ;
2511          ;     ERRHI  - MEMORY ERROR HIGH ORDER ADDRESS
2512          ;     ERRLO  - MEMORY ERROR LOW ORDER ADDRESS
2513          ;     EXP    - EXPECTED DATA
2514          ;     RECV   - RECEIVED DATA
2515          ;
2516          ;-
2517          BGNMSG MEMADD
MEMADD:     JSR     PC,PRIADD      ;PRINT MEMORY ADDRESS IN ERROR
2518          MOV     EXPD,R1        ;GET EXPD DATA
2519          MOV     RECV,R2        ;GET RECEIVED DATA
2520          JSR     PC,PRIXOR      ;PRINT EXPD/RCV
2521          ENDMMSG
2522          L10016: TRAP    C#MSG
2523          ;

```

```

014032
014032
2502 014032 012700 000012
2503 014036 004737 014642
2504 014042
014042 104423
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518 014044
014044
2519 014044 004737 010250
2520 014050 013701 002222
2521 014054 013702 002224
2522 014060 004737 010032
2523 014064
014064
014064 104423

```

PRAMPKT - PRINT RAM AND PACKET DATA

```

2525 .SBTTL PRAMPKT - PRINT RAM AND PACKET DATA
2526 ;*
2527 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2528 ;WHEN THE RAM DATA DOES NOT MATCH.
2529 ;
2530 ;INPUTS:
2531 ;
2532 ; R4 POINTER TO COMMAND PACKET
2533 ;IMPLICIT INPUTS:
2534 ; RAMDATA DATA AS READ FROM THE RAM
2535 ; RAMSIZ NUMBER OF BYTES IN PACKET
2536 ; IF RAMSIZ=0 THEN DEFAULT TO 8.
2537 ;
2538 ;IMPLICIT OUTPUTS:
2539 ; RAMSIZ SET TO 0
2540 ;-
2541 PRAMPKT:
2542 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2543 MOV #RAMDATA,R1 ;DATA FROM THE RAM
2544 CLR R2 ;INIT BYTE NUMBER
2545 5#: CMPB (R1),.(R4) ;COMPARE EXPECTED, RECEIVED
2546 BNE 7# ;BR IF NO MATCH
2547 FORCERROR 7#,NOTSSR
2548 BR 10# ;BND
2549 7#: MOVB -1(R1),R5 ;GET RECV RAM DATA
2550 MOVB -1(R4),R3 ;GET EXPD PACKET DATA
2551 XOR R5,R3 ;XOR EXPD/RECV
2552 BIC #177400,R3 ;LOW BYTE ONLY
2553 MOVB -1(R1),RECV ;GET RECEIVED RAM DATA
2554 MOVB -1(R4),EXPD ;GET EXPECTED RAM DATA
2555 PRINTB #RAMASC,R2,RECV,EXPD,R3
2556 MOV R3,-(SP)
2557 MOV EXPD,-(SP)
2558 MOV RECV,-(SP)
2559 MOV R2,-(SP)
2560 MOV #RAMASC,-(SP)
2561 MOV #5,-(SP)
2562 MOV SP,R0
2563 TRAP C#PNTB
2564 ADD #14,SP
2565 10#: INC R2 ;UPDATE BYTE COUNT
2566 TST RAMSIZ ;DEFAULT TO 8.?
2567 BEQ 15# ;BR IF YES
2568 CMP R2,RAMSIZ ;DONE ALL BYTES?
2569 BLE 5# ;BR IF NO
2570 BR 25# ;
2571 15#: CMP R2,#8. ;DONE DEFAULT NUMBER OF BYTES?
2572 BLT 5# ;BR IF NO
2573 20#: CLR RAMSIZ ;SET DEFAULT RAMSIZ
2574 25#: RTS PC ;RETURN
2575
2576 2567 014246 045 116 045 RAMASC: .ASCIZ '##A BYTE: #D2#A RAM: #03#A Packet: #03#A XOR:#03'
2577 2568 .EVEN

```

PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

```

2570           .SBTTL PRMESS - PRINT CONTENTS OF MESSAGE BUFFER
2571           ;*
2572           ;THIS ROUTINE PRINTS THE CONTENTS OF
2573           ;THE 7 OR 8 WORD MESSAGE BUFFER RETURNED BY THE TSV-05.
2574           ;
2575           ;INPUT:
2576           ;     R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
2577           ;     R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
2578           ;     NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
2579           ;THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
2580           ;-
2581 PRMESS: SAVREG           ;SAVE THE REGISTERS
2582           MOV      R0,R5   ;SAVE LOW ORDER ADDRESS
2583           TST      KTENABLE ;ADDRESS ABOVE 28K?
2584           BNE     100      ;BR IF YES
2585           CLR      R1       ;SET HIGH ORDER ADDRESS TO 0
2586 100:        MOV      R1,R3   ;SAVE HIGH ORDER ADDRESS
2587           ROL     R0        ;SHIFT BIT15 TO C BIT
2588           ROL     R1        ;SHIFT TO HIGH ORDER FOR PRINTOUT
2589           PRINTX  #PROASC,R1,R5 ;PRINT MESSAGE BUFFER ADDRESS
2590           MOV      R5,-(SP)
2591           MOV      R1,-(SP)
2592           MOV      #PROASC,-(SP)
2593           MOV      #3,-(SP)
2594           MOV      SP,R0
2595           TRAP    C#PNTX
2596           ADD     #10,SP
2597 2590:       PRINTX  #PRIASC           ;PRINT HEADER FOR CONTENTS
2598           MOV      #PRIASC,-(SP)
2599           MOV      #1,-(SP)
2600           MOV      SP,R0
2601           TRAP    C#PNTX
2602           ADD     #4,SP
2603           CLR     R4          ;NUMBER OF THE NEXT WORD
2604           MOV     R5,R1        ;COPY LOW ORDER ADDRESS
2605           MOV     R3,R0        ;COPY HIGH ORDER ADDRESS
2606           BEQ    200          ;BR IF NOT ABOVE 28K
2607           JSR    PC,SETHAP    ;SETUP PAR ADDRESS IN R0
2608           MOV     R0,R5        ;GET PAR FORMAT ADDRESS ABOVE 28K
2609 200:        PRINTX  #PRASC,R4,(R5) ;PRINT THE CONTENTS OF MEMORY BUFFER
2610           MOV     (R5),-(SP)
2611           MOV     R4,-(SP)
2612           MOV     #PRASC,-(SP)
2613           MOV     #3,-(SP)
2614           MOV     SP,R0
2615           TRAP    C#PNTX
2616           ADD     #10,SP
2617 2598:       INC     R4          ;NUMBER OF THE NEXT
2618           CMP     R4,#7        ;DONE ALL YET?
2619           BGT     500          ;BRANCH IF ALL DONE
2620           BLT     200          ;PRINT FIRST 7 WORDS
2621           BIT     #X2.EXTF,XST2(R3) ;EXTENDED FEATUTES ON?
2622           BNE     200          ;PRINT EXTENDED STATUS WORD
2623           RTS     PC          ;RETURN
2624           ;
2625           ;
2626           ;
2627           ;
045 PROASC:  .ASCIZ '%%MA Message Buffer Address = #01#05'
045 PRIASC:  .ASCIZ '%%MA Message Buffer Contents:'
045 PRASC:   .ASCIZ '%%MA Word#D1%A: #0'

```

PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

```

2609 .EVEN
2610 .SBTTL PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS
2611 ;*
2612 ;ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
2613 ; R0 - NUMBER OF WORDS IN BUFFER
2614 ;IMPLICIT INPUTS:
2615 ; EXPMSG - EXPECTED MESSAGE BUFFER
2616 ; RECMMSG - RECEIVED MESSAGE BUFFER
2617 ; RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2618 ; RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2619 ;-
2620 PRMSGEXP::
2621 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2622 MOV RO,R5 ;SAVE NUMBER OF WORDS
2623 MOV RCVLOADD,R0 ;GET RCV LOW ADDRESS
2624 MOV RO,R4 ;COPY LOW ADDRESS
2625 MOV RCVHIADD,R1 ;GET RCV HIGH ADDRESS
2626 ROL RO ;SHIFT BIT15 TO C BIT
2627 ROL R1 ;SHIFT TO HIGH ORDER FOR PRINTOUT
2628 PRINTX @PRMSGO,R1,R4 ;PRINT MESSAGE BUFFER ADDRESS
      MOV R4,-(SP)
      MOV R1,-(SP)
      MOV @PRMSGO,-(SP)
      MOV @3,-(SP)
      MOV SP,R0
      TRAP C#PNTX
      ADD @10,SP
2629 PRINTX @PRMSG1 ;PRINT HEADER FOR CONTENTS
      MOV @PRMSG1,-(SP)
      MOV @1,-(SP)
      MOV SP,R0
      TRAP C#PNTX
      ADD @4,SP
2630 CLR R4 ;NUMBER OF THE CURRENT WORD
2631 MOV @EXPMSG,R1 ;GET EXPD BUFFER ADDRESS
2632 MOV @RECMMSG,R2 ;GET RCV BUFFER ADDRESS
2633 MOV (R1),R0 ;GET EXPD
2634 MOV (R2),R3 ;GET RCV
2635 XOR R0,R3 ;XOR EXPD/RCV
2636 PRINTX @PRMSG2,R4,(R1)+,(R2)+,R3
      MOV R3,-(SP)
      MOV (R2)+,-(SP)
      MOV (R1)+,-(SP)
      MOV R4,-(SP)
      MOV @PRMSG2,-(SP)
      MOV @5,-(SP)
      MOV SP,R0
      TRAP C#PNTX
      ADD @14,SP
2637 INC R4 ;NUMBER OF THE NEXT
2638 CMP R4,R5 ;DONE ALL YET?
2639 BGE 50# ;BR IF YES
2640 BR 20# ;DO ANOTHER
2641 RTS PC ;RETURN
2642 045 116 045 PRMSG0: .ASCIZ '##A Message Buffer Address = #01#05'
2643 045 116 045 PRMSG1: .ASCIZ '##A Message Buffer Contents:'
2644 045 116 045 PRMSG2: .ASCIZ '##A WORD #02#A EXPD: #06#A RCV: #06#A XOR: #06'

```



PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS

```

2646          .EVEN
2647          .SBTTL PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
2648
2649          ;
2650          ;ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS
2651          ; ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
2652          ;
2653          ; R0 - NUMBER OF BYTES IN BUFFER
2654          ;
2655          ;IMPLICIT INPUTS:
2656          ;
2657          ; EXPMSG - EXPECTED MESSAGE BUFFER
2658          ; RECMG  - RECEIVED MESSAGE BUFFER
2659          ;
2660          PRBYTEXP
2661          SAVREG
2662          MOV R0,R5 ;SAVE R1-R5 UNTIL NEXT RETURN
2663          CLR PRMNO ;SAVE NUMBER OF BYTES
2664          CLR R4 ;INIT ERROR COUNT
2665          MOV #EXPMSG,R1 ;NUMBER OF THE CURRENT BYTE
2666          MOV #RECMG,R2 ;GET EXPD BUFFER ADDRESS
2667          MOV (R1),R0 ;GET RECV BUFFER ADDRESS
2668          BIC #C<377>,R0 ;GET EXPD BYTE
2669          MOV R0,PRBEXP ;CLEAR UPPER BYTE
2670          MOV (R2),R3 ;SAVE FOR ERROR REPORT
2671          BIC #C<377>,R3 ;GET RECV BYTE
2672          MOV R3,PRBREC ;CLEAR UPPER BYTE
2673          XOR R0,R3 ;FOR ERROR REPORT
2674          CMPB (R1),.(R2) ;XOR EXPD/RCV
2675          BEQ 30$ ;EXPD = RECV?
2676          INC PRMNO ;BR IF YES
2677          CMP PRMNO,#8 ;UPDATE ERROR COUNT
2678          BHI 30$ ;PRINTED 8?
2679          PRINTX #PRBMSG,R4,PRBEXP,PRBREC,R3 ;BR IF YES
2680          MOV R3,-(SP)
2681          MOV PRBREC,-(SP)
2682          MOV PRBEXP,(SP)
2683          MOV R4,-(SP)
2684          MOV #PRBMSG,-(SP)
2685          MOV #5,-(SP)
2686          MOV SP,R0
2687          TRAP C:PNTX
2688          ADD #14,SP
2689          FORCEXIT 50$ ;880
2690          BR 35$ ;880
2691          30$:
2692          FORCERROR 27$,NOTSSR ;880
2693          35$:
2694          INC R4 ;NUMBER OF THE NEXT
2695          CMP R4,R5 ;DONE ALL YET?
2696          BGE 50$ ;BR IF YES
2697          BR 20$ ;DO ANOTHER
2698          50$:
2699          PRINTX #PRBTOT,PRMNO ;PRINT TOTAL ERROR COUNT
2700          MOV PRMNO,-(SP)
2701          MOV #PRBTOT,-(SP)
2702          MOV #2,-(SP)
2703          MOV SP,R0

```

PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER

```

015416 104415 TRAP C:PNTX
015420 062706 000006 ADD #6,SP
2690 015424 000207 RTS PC ;RETURN
2691
2692 015426 045 116 045 PRBMSG: .ASCIZ 'N#A BYTE #D2#A EXPD: #03#A RECV: #03#A <OR: #03#A
2693 015513 045 116 045 PRBTOT: .ASCIZ 'N#A NUMBER OF BYTES IN ERROR = #D2'
2694 .EVEN
2695 015560 000000 PRBEXP: .WORD 0 ;EXPD
2696 015562 000000 PRBREC: .WORD 0 ;RECV
2697 .SBTTL EXPREC - PRINT EXPD/RECV WORD DATA
2698 ;*
2699 ;
2700 ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2701 ;
2702 ;INPUTS
2703 ;
2704 ; R1 RECEIVED DATA
2705 ; R2 EXPECTED DATA
2706 ;
2707 ;
2708 ;
2709 015564 BGNMSG EXPREC
015564
2710 015564 004737 010032 EXPPFC: JSR PC,PRIXOR ;PRINT THE DATA
2711 015570 ENDMSG
015570
015570 104423 L10017: TRAP C:MSG
.SBTTL EXPBREC - PRINT EXPD/RECV BYTE DATA
2712 ;*
2713 ;
2714 ;PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
2715 ;
2716 ;
2717 ;INPUTS:
2718 ;
2719 ; R1 RECEIVED DATA BYTE
2720 ; R2 EXPECTED DATA BYTE
2721 ;
2722 ;
2723 ;-
2724 ;
2725 015572 BGNMSG EXPBREC
015572
2726 015572 004737 007702 EXPBREC: JSR PC,PRIBXOR ;PRINT THE DATA
2727 015576 ENDMSG
015576
015576 104423 L10020: TRAP C:MSG
.SBTTL RAMERR - PRINT RAM AND PACKET DATA
2728 ;*
2729 ;
2730 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2731 ;
2732 ;INPUTS:
2733 ;
2734 ; R4 POINTER TO COMMAND PACKET
2735 ;
2736 ;
2737 ;
2738 ;

```

RAMERR - PRINT RAM AND PACKET DATA

```

2739 ;IMPLICIT INPUTS:
2740 ;
2741 ;     RAMDATA     DATA AS READ FROM THE RAM
2742 ;     RAMSIZ     NUMBER OF BYTES IN PACKET
2743 ;                IF RAMSIZ=0 THEN DEFAULT TO 8.
2744 ;
2745 ;IMPLICIT OUTPUTS:
2746 ;
2747 ;     RAMSIZ     SET TO 0
2748 ;
2749 ;-
2750 015600 BGNMSG RAMERR
2751 015600 RAMERR:: JSR PC,PRAMPKT ;PRINT RAM/PACKET DATA
2752 015604 004737 014066 ENDMG
2753 015604 L10021: TRAP C#MSG
2754 015604 104423
2755 ;
2756 ;     .SBTTL RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
2757 ;
2758 ;*
2759 ;PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2760 ;
2761 ;INPUTS:
2762 ;
2763 ;     R4         POINTER TO COMMAND PACKET
2764 ;
2765 ;IMPLICIT INPUTS:
2766 ;
2767 ;     RAMDATA     DATA AS READ FROM THE RAM
2768 ;     RAMSIZ     NUMBER OF BYTES IN PACKET
2769 ;                IF RAMSIZ=0 THEN DEFAULT TO 8.
2770 ;     ERRHI     HIGH ORDER TEST ADDRESS
2771 ;     ERRLO     LOW ORDER TEST ADDRESS
2772 ;
2773 ;IMPLICIT OUTPUTS:
2774 ;
2775 ;     RAMSIZ     SET TO 0
2776 ;
2777 015606 BGNMSG RAMTADD
2778 015606 RAMTADD:: JSR PC,PRITADD ;PRINT TEST ADDRESS
2779 015612 004737 010364 JSR PC,PRAMPKT ;PRINT RAM/PACKET DATA
2780 015616 004737 014066 ENDMG
2781 015616 L10022: TRAP C#MSG
2782 015616 104423
2783 ;
2784 ;     .SBTTL RAMEXP - PRINT RAM EXPD/RECV DATA
2785 ;
2786 ;*
2787 ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2788 ;
2789 ;INPUTS:
2790 ;
2791 ;     R1         RECEIVED DATA
2792 ;     R2         EXPECTED DATA

```

RAMEXP - PRINT RAM EXPD/RECV DATA

```

2790          ;          R4          CONTROLLER RAM ADDRESS
2791          ;--
2792
2793 015620          BGNMSG  RAMEXP
          RAMEXP::
2794 015620 042701 177400          BIC      #C<377>,R1          ;SAVE EXPD RAM DATA BYTE
2795 015624 042702 177400          BIC      #C<377>,R2          ;SAVE EXPD RAM DATA BYTE
2796 015630 004737 010156          JSR      PC,PRIRAM          ;PRINT THE RAM ADDRESS
2797 015634 004737 010032          JSR      PC,PRIXOR          ;PRINT THE DATA
2798 015640          ENDMSG
          L10023:
          TRAP      C#MSG
          .SBTTL   TIMEXP - PRINT TIMER A,B AND EXP/REC
          ;*
          ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
          ;AND TIMER A,B HEADER MESSAGE
          ;
          ;INPUTS:
          ;
          ;          R1          RECEIVED DATA
          ;          R2          EXPECTED DATA
          ;--
2812 015642          BGNMSG  TIMEXP
          TIMEXP::
2813 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          ADR      #4,SP
2814 015662 004737 010032          JSR      PC,PRIXOR          ;PRINT THE DATA
2815 015666          ENDMSG
          L10024:
          TRAP      C#MSG
2816
2817 015670          045          116          045  TIMSGO: .ASCIZ  'TIMER A STATUS IS IN BIT 3/TIMER B STATUS IS IN BIT 2'
2818          .EVEN
2819          .SBTTL   BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS
2820
2821          ;*
2822          ;PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
2823          ;
2824          ;INPUTS:
2825          ;
2826          ;          R1          CONTENTS OF TSSR
2827          ;          R2          DATA WRITTEN (8 BITS)
2828          ;
2829          ;--
2830
2831          BGNMSG  BADSSR
          BADSSR::
2832 015770          MOV      R2,-(SP)          ;SAVE DATA TRANSFERRED
          015770          BIC      #177400,R2          ;GET JUST ONE BYTE
2833 015770 010246 177400
2834 015772 042702 177400

```

BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS

2835	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	
2836	016020	012602			MOV	(SP)+,R2	;RESTORE R2
2837	016022	004737	006022		JSR	PC.PRITSSR	;DECODE TSSR CONTENTS
2838	016026				ENDMSG		
	016026			L10025:			
	016026	104423			TRAP	C#MSG	
2839	016030	045	116	045	XFERASC:	.ASCIZ	'#N#A Data Transferred = #03'

GLOBAL SUBROUTINES SECTION

```

2841      .SBTTL  GLOBAL SUBROUTINES SECTION
2842
2843      ;**
2844      ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
2845      ; THAT ARE USED IN MORE THAN ONE TEST.
2846      ;--
2847      .SBTTL  SOFINIT - SOFT INITIALIZE OF CONTROLLER
2848
2849      ;*
2850      ; ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
2851      ; BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
2852      ; THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
2853      ; DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
2854
2855      ; INPUTS:
2856      ;
2857      ;       R5      ADDRESS OF FIRST REGISTER
2858
2859      ; OUTPUTS:
2860      ;
2861      ;       R0      CONTENTS OF TSSR, IF ERROR
2862      ;       CARRY   SET IF INIT WAS OKAY
2863      ;               CLEAR IF FATAL ERROR
2864
2865      ; CALLING SEQUENCE:
2866      ;
2867      ;       MOV     #ADDRESS,R5
2868      ;       JSR     PC,SOFINIT
2869      ;       BCS    CONTINUE
2870      ;
2871      ;       ERRDF          ;REPORT FATAL ERROR
2872
2873      ;-
2874
2875 016064      SOFINIT::
2876 016064      SAVREG          ; SAVE THE REGISTERS
2877 016070 012765 000000 000002      MOV     #0,TSSR(R5)      ; DO THE INIT.
2878 016076 004737 016340      JSR     PC,WAITF        ; WAIT FOR SSR
2879 016102 016500 000002      MOV     TSSR(R5),R0    ; GET THE TSSR REGISTER
2880 016106 010004      MOV     R0,R4          ; TSSR CONTENTS
2881 016110 042704 176277      BIC     #C<HIADDR!OFL>,R4
2882 016114 052704 002200      BIS     #SSR!NBA,R4   ; R4 HAS EXPECTED CONTENTS
2883 016120 020400      CMP     R4,R0         ; ONLY EXPECTED BITS SET ?
2884 016122 001402      BEQ     5#           ; BRANCH IF OKAY
2885 016124 000241      CLC                    ; CLEAR THE CARRY FOR ERROR
2886 016126 000401      BR     10#          ; GO TO EXIT
2887 016130 000261 5#:      SEC                    ; SET THE CARRY BIT
2888 016132 000207 10#:     RTS     PC          ; RETURN TO CALLER

```

CHKAMB - CHECK TSSR FOR AMBIGUITY

```

2890 .SBTTL CHKAMB - CHECK TSSR FOR AMBIGUITY
2891
2892 ;*
2893 ; THIS ROUTINE TESTS THE CONTENTS OF THE TSSR REGISTER
2894 ; FOR AMBIGUITY
2895
2896 ; INPUT:
2897 ;
2898 ; RO CONTENTS OF TSSR
2899 ;
2900 ; OUTPUT:
2901 ;
2902 ; RO CONTENTS OF TSSR
2903 ;
2904 ; CARRY SET - NO AMBIGUITY
2905 ; CLR - AMBIGUOUS CONTENTS
2906 ;
2907 ; -
2908 ; -
2909
2910 CHKAMB:
2911 SAVREG ;SAVE THE GENERAL REGISTERS
2912 MOV RO,R4 ;CONTENTS OF TSSR
2913 BIT #SC,R0 ;IS BIT 15 SET ?
2914 BNE 5# ;BRANCH IF YES
2915 BIT #C<NBA!OFL!SSR!HIADDR>,RO ;ANY OTHER BITS SET ?
2916 BNE 40# ;MUST BE AN ERROR
2917 BR 45# ;RETURN WITH SUCCESS
2918 5#: BIT #SSR,R0 ;IS READY BIT SET ?
2919 BNE 10# ;BRANCH IF READY BIT IS SET.
2920 BIT #BITS,R0 ;IS FATAL ERROR BIT SET ?
2921 BEQ 40# ;ERROR IF NOT
2922 BIC #CTERCLS,R4 ;CLEAR ALL BUT TERMINATION CODE
2923 CMP R4,#16 ;ALL THREE BITS MUST BE SET
2924 BNE 40# ;ERROR IF NOT SET
2925 BR 45# ;OK IF ALL ARE SET
2926 10#: BIT #BITS,R0 ;IS FATAL ERROR BIT SET ?
2927 BEQ 45# ;ERROR IF BIT IS SET WITH SSR
2928 BIT #BIT2:BIT1,R0 ;IS THIS A FUNCTION REJECT
2929 BNE 45# ;BR, IF TSSR IS OK
2930 40#: CLC ;AMBIGUOUS CONTENTS
2931 BR 50#
2932 45#: SEC ;SHOW SUCCESS - NO AMBIGUITY
2933 50#: RTS PC ;RETURN TO CALLER

```

ENAIN,DSBINT - ENABLE/DISABLE INTERRUPTS

```

2935      .SBTTL ENAIN,DSBINT - ENABLE/DISABLE INTERRUPTS
2936      ;
2937      ; DEFAULT DISPLAY INTERRUPT HANDLERS.
2938      ; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
2939      ; OTHERWISE, SAVE DPU REGISTERS AND DISMISS.
2940      ;
2941      ; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
2942      ;
2943      ;
2944      000200      IOKCKIN=BIT7      ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
2945      000001      IOKSTP=BIT0      ; EXPECT "STOP" INTERRUPT.
2946      ;
2947      ; INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
2948      016234      000      INTMASK:      .BYTE      0
2949      ; INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
2950      016235      000      INTFLAG:     .BYTE      0
2951      ; SAVED INTERRUPT VECTOR:
2952      ;
2953      016236      C00000      INTVEC:     .WORD      0
2954      ; SAVE CPU PC
2955      016240      000000      INTCPC:    .WORD      0
2956      ;
2957      ; SUBROUTINE TO ENABLE INTERRUPTS:
2958      016242      010046      ENAIN:      MOV      RO,-(SP)      ; SAVE RO
2959      016244      013700      002200      MOV      IVEC,RO      ; GET POINTER TO VECTORS
2960      016245      012720      016306      MOV      #INTR,(RO)+  ; SET UP INTERRUPT VECTOR
2961      016246      012720      000340      MOV      #PRI07,(RO)+
2962      016247      012600      MOV      (SP)+,RO      ; RESTORE RO
2963      016248      011646      MOV      (SP),-(SP)
2964      016264      012766      000000      000002      MOV      #0,2(SP)      ; SET CPU TO LEVEL 0
2965      016272      000002      RTI
2966      ;
2967      ; SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
2968      016274      011646      DSBINT:   MOV      (SP),-(SP)
2969      016276      012766      000340      000002      MOV      #PRI07,2(SP)
2970      016304      000002      RTI
2971      .SBTTL INTR - INTERRUPT HANDLERS
2972      ;
2973      016306      BGNSRV INTR      ; DEFINE INTERRUPT ENTRY
2974      016306      012737      000001      002214      INTR::   MOV      #1,INTRECV      ; SET FLAG TO SHOW INTERRUPT RECEIVED
2975      016314      105037      016235      CLR      INTFLAG      ; CLEAR FLAG TO SAY WE GOT INTERRUPT
2976      016320      132737      000001      016234      RITB     #IOKSTP,INTMASK ; EXPECTING STOP INTERRUPT?
2977      016326      001003      BNE     1$      ; BR IF YES
2978      016330      152737      000001      016235      BISB     #IOKSTP,INTFLAG ; NO. SET THE ERROR FLAG.
2979      ;
2980      ; SAVE REGISTERS, MSG BUFFER, ETC.
2981      016336      1$:
2982      016336      ENDSRV
2983      016336      L10026:
2984      016336      000002      RTI

```



WAITF - WAIT FOR SUBSYSTEM READY

```

2984 .SBTTL WAITF - WAIT FOR SUBSYSTEM READY
2985 ;
2986 ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
2987 ;
2988 ; INPUTS:
2989 ;
2990 ; R5 ADDRESS OF FIRST DEVICE REGISTER
2991 ;
2992 ; OUTPUTS:
2993 ;
2994 ; R0 CONTENTS OF LAST TSSR READ
2995 ; CARRY SET - READY BIT SET
2996 ; CLR - TIMEOUT WAITING FOR READY
2997 ;
2998 016340 000401 WAITF:: BR 1# ;NOP WHEN SUPER FIXED
2999 016342 BREAK 1# ; DO A SUPVSR BREAK FIRST.
016342 104422 TRAP C#BRK
3000 016344 012746 011000 1# : MOV #11000,-(SP) ;25-APRIL-83 REV B - 1100 MSEC TIMER
3001 016350 C16500 000002 2# : MOV TSSR(R5),R0 ;READ THE TSSR REGISTER
3002 016354 105700 TSTB R0 ;TEST FOR READY BIT SET
3003
3004 016356 100420 BMI 3# ; EXIT ON STOP FLAG.
3005 016360 DELAY 1 ; WAIT 100 USEC
016360 012727 000001 MOV #1,(PC)+
016364 000000 .WORD 0
016366 013727 002116 MOV L#DLY,(PC)+
016372 000000 .WORD 0
016374 005367 177772 DEC -6(PC)
016400 001375 BNE -4
016402 005367 177756 DEC -22(PC)
016406 001367 BNE -20
3006 016410 005316 DEC (SP) ;REDUCE DELAY COUNT
3007 016412 001356 BNE 2# ;RETRY UNTIL TIMER EXPIRES
3008 016414 000241 CLC ; C = 0, CONTROLLER STILL RUNNING...
3009 016416 000401 BR 4# ;...OR HUNG-UP AFTER 300 MSEC.
3010 016420 000261 3# : SEC ; C = 1, CONTROLLER IS STOPPED.
3011 016422 005326 4# : DEC (SP)+ ;RESTORE STACK WITHOUT CHANGING CARRY BIT
3012 016424 000207 RTS PC

```

CHKTSSR - CHECK TSSR FOR READY

```

3014 .SBTTL CHKTSSR - CHECK TSSR FOR READY
3015 ;*
3016 ; THIS ROUTINE WAITS FOR READY IN THE TSSR
3017 ; AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
3018 ;
3019 ; INPUT:
3020 ;       RS      ADDRESS OF CSR REGISTERS
3021 ;
3022 ; OUTPUT:
3023 ;       RO      CONTENTS OF TSSR
3024 ;       CARRY   SET - OKAY
3025 ;             CLR - NOT READY AMBIGUOUS, OR SC SET
3026 ;
3027 016426 CHKTSSR:
3028 016426 004737 016340 JSR PC, WAITF ; WAIT FOR READY
3029 016432 103014 BCC 20# ; BRANCH IF TIME OUT
3030 016434 004737 016134 JSR PC, CHKAMB ; TSSR AMBIGUOUS?
3031 016440 103006 BCC 10# ; BR IF YES
3032 016442 C32700 100000 BIT #SC, RO ; SPECIAL CONDITION SET?
3033 016446 001405 BEQ 15# ; BR IF NO
3034 016450 032700 074000 BIT #<SCE!BIE!RMR!NXM>, RO ; ANY ERROR BITS SET?
3035 016454 001402 BEQ 15# ; BR IF NO
3036 016456 000241 10#: CLC ; SET FAILURE
3037 016460 000401 BR 20# ;
3038 016462 000261 15#: SEC ; SET SUCCESS
3039 016464 000207 20#: RTS PC ; RETURN TO CALLER
3040 .SBTTL XNXM - CHECK FOR NONEXISTENT MEMORY
3041 ;*
3042 ; ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
3043 ; ON RETURN, IF "C" = 1, (R1) = NEXM ADDRESS.
3044 ; "C" = 0, ALL ADDRESSES OK.
3045 ;
3046 ; CALL: MOV ADR1, R1
3047 ;       MOV ADR2, R2
3048 ;       JSR PC, NXM
3049 ;       RETURN ; TEST "C" AND PROCEED.
3050 016466 012737 016520 000004 XNXM: MOV #2#, #04 ; SET BUSERR VECTOR.
3051 016474 012737 000200 000006 MOV #PRI04, #06
3052 016502 005003 CLR R3 ; FLAG.
3053 016504 005711 1#: TST (R1) ; TEST THE ADDRESS(ES).
3054 ; IF ANY TRAP, CONTINUE AT 2#.
3055 016506 020102 CMP R1, R2 ; OTHERWISE, CONTINUE HERE.
3056 016510 001407 BEQ 3# ; BR IF FINISHED (NO NEXM'S).
3057 016512 062701 000002 ADD #2, R1 ; SET NEXT ADDRESS...
3058 016516 000772 BR 1# ; ...AND CONTINUE.
3059 016520 005103 2#: COM R3 ; GOT ONE, SET FLAG...
3060 016522 012716 016530 MOV #3#, (SP)
3061 016526 000002 RTI ; ...AND DISMISS INTERRUPT...
3062 016530 3#: CLRVEC #4 ; ...AND GIVE BACK THE VECTOR.
3063 016530 012700 000004 MOV #4, RO
3064 016534 104436 TRAP C: CVEC
3065 016536 005703 TST R3 ; DID WE CATCH ONE ??
3066 016540 001401 BEQ .+4 ; NO, "C" = 0, SKIP NEXT.
3067 016542 000261 SEC ; YES, "C" = 1, (R1) = NEXM ADDR.
3068 016544 000207 RTS PC

```

TSTLOOP - CHECK ITERATION COUNT

```

3068 .SBTTL TSTLOOP - CHECK ITERATION COUNT
3069
3070 ; SUBROUTINE TO EXECUTE TEST ITERATIONS.
3071 ; EXIT WITH "C" SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
3072 ; LOOP COUNTER IS SET BY "BEGIN.TEST" MACRO.
3073
3074 ; CALL: LOOPTO ARG
3075
3076 TSTLOOP::
3077     TST     NOITS           ; ITERATIONS INHIBITED?
3078     BNE     1#             ; YES.
3079     TST     QVP            ; NO.
3080     BMI     1#             ; LOOPS DISALLOWED IN QUICK PASS.
3081     DEC     LOOPCNT        ; BUMP LOOP COUNTER.
3082     BNE     2#
3083     1#:    CLC              ; LOOP DISALLOWED, OR DONE.
3084     BR     3#
3085     2#:    SEC              ; LOOP ENABLED.
3086     3#:    RTS             PC
3087
3088 .SBTTL TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS
3089
3090 ; PRINT THE NUMBER AND NAME OF EACH TEST AS WE GO ALONG.
3091 ; INCREMENT "TESTK" TO INDICATE THE NUMBER OF TESTS
3092 ; IN THE CURRENT RUN SEQUENCE.
3093 ; CLEAR THE ERROR COUNTER AND SIGNATURE EXTENSION FLAGS.
3094
3095 ; INPUT:
3096 ;
3097 ;     R0     POINTER TO TEST ID ASCIZ STRING
3098
3099 ; OUTPUT:
3100 ;
3101 ;     R5     ADDRESS OF FIRST DEVICE REGISTER
3102
3103 ; IMPLICIT OUTPUTS:
3104 ;
3105 ;     TSTCNT UPDATED TO COUNT TESTS PERFORMED SINCE START OR RESTART
3106
3107 ; SIDE EFFECTS:
3108 ;
3109 ;     INTERRUPT LEVEL IS RASIED TO LEVEL OF
3110 ;     THE DEVICE UNDER TEST
3111
3112 ; -
3113
3114 TSTSETUP::
3115     MOV     R0, -(SP)      ; SAVE THE TEST ID MESSAGE
3116     CLR     SIFLAG        ; CLEAR "SOFT INIT" FLAG
3117     CLR     ERRK          ; CLEAR LOCAL ERROR COUNTER.
3118     CLR     EXTA          ; CLEAR ERROR EXTENSION FLAG.
3119     CLRB    INTMASK       ; CLEAR INTERRUPT MASK (CHECK ERROR)
3120     MOV     UNITN, R0     ; GET THE UNIT NUMBER.
3121     ASL     R0            ; ... AND MAKE IT A WORD OFFSET.
3122     TST     NODEV        ; DID STARTUP FIND THE DEVICE?
3123     BEQ     4#           ; BR IF YES
3124     BPL     3#           ; BR IF NOT IDLE

```

TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS

```

3125 016640 052760 160000 003166      BIS      #160000,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
3126 016646      ERRDF    1,NXR,NXRERR ; NO DEVICE HERE -- PRINT IT
      016646 104455      TRAP    C#ERRDF
      016650 000001      .WORD   1
      016652 003736      .WORD   NXR
      016654 005734      .WORD   NXRERR
3127 016656 000407      BR      2#
3128 016660 052760 160001 003166 3# :  BIS      #160001,ERTABL(R0) ; FLAG ERROR IN THE ERROR TABLE
3129 016666      ERRDF    2,NOINIT ; DEVICE NOT IDLE
      016666 104455      TRAP    C#ERRDF
      016670 000002      .WORD   2
      016672 004333      .WORD   NOINIT
      016674 000000      .WORD   0
3130 016676 012737 177777 003102 2# :  MOV      #-1,DUFLG ; DROP THE UNIT
3131 016704      DODU     UNITN
      016704 013700 002172      MOV      UNITN,R0
      016710 104451      TRAP    C#DODU
3132 016712      DOCLN   ; ABORT THE PASS
      016712 104444      TRAP    C#DOCLN
3133 016714 000423      BR      5#
3134
3135 016716      RFLAGS   R0 ; GET THE OPERATOR FLAGS.
      016716 104421      TRAP    C#RFLA
3136 016720 032700 001000      BIT      #PNT,R0 ; PRINT THE TEST NUMBERS?
3137 016724 001412      BEQ     1# ; BR IF NO
3138 016726 011600      MOV      (SP),R0 ; GET THE ID MESSAGE
3139 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
3140 016752 005237 002204      1# :  INC      TSTCNT ; BUMP TEST COUNTER.
3141 016756      SETPRI   IPRI ; PRIORITY THAT OF DEVICE
      016756 013700 002202      MOV      IPRI,R0
      016762 104441      TRAP    C#SPRI
3142 016764 005726      5# :  TST      (SP)+ ; FIX UP THE STACK
3143 016766 013705 002176      MOV      CSRADDR,R5 ; ADDRESS OF TSV REGISTERS ON UNIBUS
3144 016772 000207      RTS     PC
3145 016774      045     123     045  TNAM:  .ASCIZ  '#S#T#A Test'
3146
3147      .EVEN
3148      .SBTTL  TSTEND - PRINT ERRORS RECEIVED
3149
3150 ; AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
3151 ; IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
3152
3152 TSTEND: RFLAGS   R0
      017010      TRAP    C#RFLA
3153 017012 030027 020000      BIT      R0,#IER
3154 017016 001412      BEQ     1# ; BR IF "IER" NOT SET.
3155 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
    
```

TSTEND - PRINT ERRORS RECEIVED

3156	017040	062706	000006			ADD	#6.SP	
	017044	000207			1#:	RTS	PC	
3157								
3158	017046	000000			ERRK:	0		LOCAL ERROR COUNT.
3159	017050	045	101	040	ESUM:	.ASCIZ	/#A #D#A ERRORS/	
3160	017067	105	122	122	EMAXDU:	.ASCIZ	/ERROR LIMIT REACHED -- DROPPING UNIT/	
3161						.EVEN		



CKDROP - CHECK IF UNIT SHOULD BE DROPPED

```

3198 .SBTTL CKDROP CHECK IF UNIT SHOULD BE DROPPED
3199 ;
3200 ; CHECK IF UNIT SHOULD BE DROPPED
3201 ;
3202 017272 010046 CKDROP: MOV RO, -(SP)
3203 017274 FORCERROR 18, NOTSSR
3204 017304 RFLAGS RO
017304 104421 TRAP C#RFLA
3205 017306 032700 000040 BIT @IDU, RO
3206 017312 001010 BNE 18
3207 017314 011600 MOV (SP), RO
3208 017316 012737 177777 003102 MOV @-1, DUFLG
3209 017324 DODU UNITN
017324 013700 002172 MOV UNITN, RO
017330 104451 TRAP C#DODU
3210 017332 DOCLN ; ABORT THE PASS
017332 104444 TRAP C#DOCLN
3211 017334 012600 18: MOV (SP)+, RO
3212 017336 C00207 RTS PC
3213
3214
3215 .SBTTL CONFIG - DETERMINE CONFIGURATION OF SYSTEM
3216 ;
3217 ; SUBROUTINE - DETERMINE CONFIGURATION OF TSV05 SYSTEM.
3218 ;
3219 017340 CONFIG: JSR PC, SOFINIT
3220 017340 004737 016064 RTS PC
3221 017344 000207 .SBTTL KTON, KTOFF ENABLE/DISABLE MEMORY MANAGEMENT
3222
3223 ;
3224 ; SUBROUTINE - ENABLE MEM MGT.
3225 ;
3226 017346 005737 003122 KTON: TST KFLG ; GOT KT?
3227 017352 001403 BEQ 18 ; NO.
3228 017354 012737 000001 177572 MOV @1, SRO ; YES. ENABLE KT11.
3229 017362 000207 18: RTS PC
3230
3231 ;
3232 ; SUBROUTINE - DISABLE MEM MGT.
3233 ;
3234 017364 005737 003122 KTOFF: TST KFLG ; GOT KT11?
3235 017370 001405 BEQ 18 ; NO.
3236 017372 000240 NOP
3237 017374 000240 NOP
3238 017376 012737 000000 177572 MOV @0, SRO ; DISABLE KT.
3239 017404 000207 18: RTS PC

```

SETMAP - SETUP PAR6 MAPPING

```

3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260 017406
3261 017406
3262 017412 005737 003122
3263 017416 001433
3264 017420 010102
3265 000006
3266
3267
3268
3269 017452 042701 000177
3270 017456 020137 003122
3271 017462 103011
3272 017464 010137 172352
3273 017470 042702 160000
3274 017474 062702 120000
3275 017500 010200
3276 017502 000261
3277 017504 000401
3278 017506 000241
3279 017510 000207
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295 017512
3296 017512
3297 017516 004737 017364
    
```

```

.SBTTL SETMAP - SETUP PAR6 MAPPING
;
; THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
; AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
; IS RETURNED BIASED TO PAR6.
;
; INPUTS:
;
; R0 HIGH ORDER ADDRESS BITS
; R1 LOW ORDER ADDRESS BITS
;
; OUTPUTS:
;
; R0 OFFSET INTO BLOCK WITH PAR6 BIAS (I.E. THE ADDRESS)
; CARRY SET IF SUCCESS
; CLR IF ERROR
;
; SETMAP:
; SAVREG ;SAVE R1-R4 UNTIL NEXT RETURN
; TST KTF LG ;SYSTEM HAVE ABOVE 28K?
; BEQ 10$ ;BR IF NO
; MOV R1,R2 ;SAVE LOW ORDER BITS
; .REPT 6
; ASR R0 ;CONVERT WORD ADDRESS TO 32W BLOCKS
; ROR R1 ;MAKE IT DOUBLE PRECISION
; .ENOR
; BIC @177,R1 ;ALINE FOR LOWER 4K BOUNDARY
; CMP R1,KTF LG ;HIGHER THAN EXISTING MEMORY?
; BHIS 10$ ;BR IF YES
; MOV R1,@KIPARS ;SETUP MAPPING REGISTER PARS
; BIC @160000,R2 ;SETUP DISPLACEMENT IN PAGE
; ADD @120000,R2 ;ADD IN PARS BIAS
; MOV R2,R0 ;RETURN IN R0
; SEC ;SET SUCCESS
; BR 15$
;
; 10$: CLC ;SET FAILURE
; 15$: RTS PC ;RETURN
;
; .SBTTL FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
;
; FILL MEMORY WITH A BACKGROUND PATTERN
;
; INPUTS:
;
; R0 = BACKGROUND PATTERN
; FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
; KTF LG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
;
; OUTPUTS:
;
; NONE
;
; FILLMEM:
; SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
; JSR PC,KTOFF ;DISABLE KT.
    
```



FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN

3298	017522	010003				MOV	R0,R3	;COPY TEST PATTERN
3299	017524	013701	003114			MOV	FREE,R1	;GET FIRST FREE LOCATION
3300	017530	013702	003116			MOV	FRESIZ,R2	;SIZE OF FREE SPACE BELOW 28K.
3301	017534	010321		10#:		MOV	R3,(R1)+	;STORE A BACKGROUND WORD
3302	017536	005302				DEC	R2	;DONE ALL MEMORY IN FREE SPACE?
3303	017540	003375				BGT	10#	;BR IF NO
3304	017542	005737	003122			TST	KTFLG	; GOT KT?
3305	017546	001477				BEQ	55#	; NO. GET OUT.
3306	017550	004737	017346			JSR	PC,KTON	; YES. ENABLE KT.
3307	017554	005000				CLR	R0	;HIGH ORDER ADDRESS START
3308	017556	013701	003142			MOV	PST32W,R1	;GET >28K START ADDRESS (IN 32W BLOCKS)
3309		000006				.REPT	6	
3310						CLC		;CLEAR C BIT
3311						ROL	R1	;CONVERT BLOCKS TO WORDS
3312						ROL	R0	;MAKE IT DOUBLE PRECISION
3313						.ENDR		
3314	017626	004737	017406			JSR	PC,SETMAP	;SETUP PARS MAPPING REGISTER
3315	017632	010320		30#:		MOV	R3,(R0)+	;STORE TEST PATTERN IN >28K ADDRESS
3316	017634	C20027	140000			CMP	R0,#140000	;END OF PARS MAPPING AREA?
3317	017640	103774				BLO	30#	;BR IF NO
3318	017642	162700	020000			SUB	#20000,R0	;BACKUP INTO PARS MAPPING BEGIN
3319	017646	062737	000200	172352		ADD	#200,#KIPARS	;POINT TO NEXT 4K BLOCK >28K.
3320	017654	023727	172352	006000		CMP	#KIPARS,#6000	;END OF MEMORY BELOW XMON AREA?
3321					!!!			
3322	017662	001427				BEQ	50#	;BR IF YES
3323	017664	005737	003134			TST	T23A	;11/23A?
3324	017670	001407				BEQ	35#	;NO KEEP GOING
3325	017672	013704	177572			MOV	SRO,R4	;GET SRO CONTENTS
3326	017676	042704	177761			BIC	#177761,R4	;CLEAR ALL BUT PAGE NUMBER
3327	017702	022704	000016			CMP	#16,R4	;SEE IF PAGE 7
3328	017706	001415				BEQ	50#	;EXIT IF THERE
3329	017710	005737	003136		35#:	TST	T23B	;11/23B?
3330	017714	001410				BEQ	45#	;NO KEEP GOING
3331	017716	023727	172352	007600		CMP	#KIPARS,#7600	;REACHED 18 BITS?
3332	017724	103001				BHIS	40#	;YES
3333	017726	000403				BR	45#	;NO KEEP GOING
3334	017730	012737	000020	172516	40#:	MOV	#20,SR3	;SET 22 BIT RELOCATION
3335	017736	000137	017632		45#:	JMP	30#	;KEEP GOING ON ETC.
3336	017742	004737	017364		50#:	JSR	PC,KTOFF	; DISABLE KT.
3337	017746	000207			55#:	RTS	PC	

CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN

```

3339 .SBTTL CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN
3340
3341 ; COMPARE MEMORY WITH A BACKGROUND PATTERN
3342 ;
3343 ; INPUTS:
3344 ;
3345 ; RO = BACKGROUND PATTERN
3346 ; FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
3347 ; KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
3348 ;
3349 ; OUTPUTS:
3350 ;
3351 ; CARRY - SET IF NO ERROR
3352 ; CARRY - CLR IF ERROR
3353 ;
3354 ; IMPLICIT OUTPUTS:
3355 ;
3356 ; ERRHI - ERROR HIGH ADDRESS
3357 ; ERRLO - ERROR LOW ADDRESS
3358 ; EXPD - EXPECTED DATA
3359 ; RECV - RECEIVED DATA
3360 ;
3361 CMPMEM:
3362 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
3363 MOV RO,R3 ;COPY TEST PATTERN
3364 JSR PC,KTOFF ;DISABLE KT.
3365 MOV FREE,R1 ;GET FIRST FREE LOCATION
3366 MOV FRESIZ,R2 ;SIZE OF FREE SPACE BELOW 28K.
3367 10#: CMP R3,(R1) ;FREE SPACE LOCATION EQUAL TO EXPD?
3368 BEQ 15# ;BR IF YES
3369 MOV R1,ERRLO ;SAVE ADDRESS IN ERROR
3370 CLR ERRHI ;NO HIGH ADDRESS
3371 MOV R3,EXPD ;SAVE EXPD FOR ERROR REPORT
3372 MOV (R1),RECV ;SAVE RECV FOR ERROR REPORT
3373 BR 50#
3374 15#: TST (R1)+ ;POINT TO NEXT ADDRESS
3375 DEC R2 ;DJNE ALL MEMORY IN FREE SPACE?
3376 BGT 10# ;BR IF NO
3377 TST KTFLG ; GOT KT?
3378 BEQ 55# ; NO. GET OUT.
3379 JSR PC,KTON ; YES. ENABLE KT.
3380 CLR RO ;HIGH ORDER ADDRESS START
3381 MOV PST32W,R1 ;GET >28K START ADDRESS (IN 32W BLOCKS)
3382 .REPT 6
3383 ROL R1 ;CONVERT BLOCKS TO WORDS
3384 ROL RO ;MAKE IT DOUBLE PRECISION
3385 .ENDR
3386 BIC #177,R1 ;ALINE 4K BOUNDARY
3387 MOV RO,-(SP) ;SAVE HIGH ORDER
3388 MOV R1,-(SP) ;SAVE LOW ORDER
3389 JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
3390 MOV RO,R4 ;COPY ADDRESS BIASED TO PAR6
3391 MOV (SP)+,R1 ;RESTORE LOW ORDER IN NON PAR6 FORMAT
3392 MOV (SP)+,RO ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
3393 30#: CMP R3,(R4) ;ABOVE 28K LOCATION EQUAL EXPD?
3394 BEQ 32# ;BR IF YES
3395 MOV RO,ERRHI ;SAVE HIGH ORDER IN ERROR

```

CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN

```

3396 020130 010137 002230          MOV    R1,ERRLO      ;SAVE LOW ORDER IN ERROR
3397 020134 010337 002222          MOV    R3,EXPD      ;SAVE EXPD FOR ERROR REPORT
3398 020140 011437 002224          MOV    (R4),RECV    ;SAVE RECV FOR ERROR REPORT
3399 020144 000421                    BR     50$          ;
3400 020146 062701 000002          32$:  ADD    #2,R1      ;UPDATE NON PAR6 ADDRESS
3401 020152 005500                    ADC    R0           ;MAKE IT DOUBLE PRECISION ADD
3402 020154 062704 000002          ADD    #2,R4        ;UPDATE PAR FORMAT ADDRESS
3403 020160 020427 140000          CMP    R4,#140000   ;END OF PAR5 MAPPING AREA?
3404 020164 103755                    BLO   30$          ;BR IF NO
3405 020166 162704 020000          SUB    #20000,R4    ;BACKUP INTO PAR6 MAPPING BEGIN
3406 020172 062737 000200 172352  ADD    #200,#KIPARS ;POINT TO NEXT 4K BLOCK >28K.
3407 020200 023737 172352 003122  CMP    #KIPARS,KTFLG ;END OF MEMORY?
3408 020206 101744                    BLOS  30$          ;BR IF NO
3409 020210 004737 017364          50$:  JSR    PC,KTOFF  ;TURN OFF MEMORY MAPPING
3410 020214 000241                    CLC                    ;SET FAILURE
3411 020216 000403                    BR     60$          ;
3412 020220 004737 017364          55$:  JSR    PC,KTOFF  ;TURN OFF MEMORY MAPPING
3413 020224 000261                    SEC                    ;SET SUCCESS
3414 020226 C00207          60$:  RTS     PC
3415                    .SBTTL  REGSAV - SAVE R1-R5 ON STACK
3416                    ;*
3417                    ;
3418                    ;ROUTINE TO
3419                    ;SAVE R1 THROUGH R5 ON THE STACK
3420                    ;
3421                    ;CALLING SEQUENCE:
3422                    ;
3423                    ;      JSR    R5,REGSAV
3424                    ;
3425                    ;THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
3426                    ;THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
3427                    ;THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTORE
3428                    ;REGISTERS.
3429                    ;
3430                    ;THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
3431                    ;CALLED VIA A JSR PC INSTRUCTION
3432                    ;
3433                    ;-
3434
3435 020230          REGSAV:
3436 020230 010446          MOV    R4,-(SP)
3437 020232 010346          MOV    R3,-(SP)
3438 020234 010246          MOV    R2,-(SP)
3439 020236 010146          MOV    R1,-(SP)
3440 020240 010546          MOV    R5,-(SP)
3441 020242 016605 000012  MOV    10.(SP),R5
3442 020246 004736          JSR    PC,@(SP)+
3443 020250 012601          MOV    (SP)+,R1
3444 020252 012602          MOV    (SP)+,R2
3445 020254 012603          MOV    (SP)+,R3
3446 020256 012604          MOV    (SP)+,R4
3447 020260 012605          MOV    (SP)+,R5
3448 020262 000207          RTS     PC

```

GETPAT - GET 8 BIT PATTERN FROM OPERATOR

```

3450          .SBTTL  GETPAT - GET 8 BIT PATTERN FROM OPERATOR
3451          ;*
3452          ;ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
3453          ;
3454          ;INPUTS:          NONE.
3455          ;
3456          ;OUTPUTS:
3457          ;          RO          OCTAL NUMBER FROM THE OPERATOR
3458          ;
3459          ;CALLING SEQUENCE:
3460          ;          JSR          PC,GETPAT
3461          ;-
3462          GETPAT::
3463          1$:          SAVREG          ;SAVE THE GENERAL REGISTERS
3464          GMANID          DATASC,PATDAT,0,377,0,377,NO
3465          TRAP          C#GMAN
3466          BR          10000$
3467          .WORD          PATDAT
3468          .WORD          T#CODE
3469          .WORD          DATASC
3470          .WORD          377
3471          .WORD          T#LOLIM
3472          .WORD          T#HILIM
3473          10000$:
3474          BNCOMPLETE          1$          ;RETRY IF ERROR
3475          BCC          1$
3476          MOV          PATDAT,RO          ;DATA PATTERN FROM OPERATOR
3477          RTS          PC          ;RETURN TO CALLER
3478          ;*
3479          ;LOCAL DATA AREA
3480          ;-
3481          PATDAT: .WORD          0          ;TEMPORARY STORAGE FOR DATA
3482          DATASC: .ASCIZ          'ENTER DATA PATTERN'
3483          .EVEN

```

GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE

```

3477      .SBTTL GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE
3478      ;
3479      ;ROUTINE TO ISSUE A MENU AND GET THE OPERATOR'S RESPONSE.
3480      ;
3481      ;INPUTS:
3482      ;      R0      ADDRESS OF ASCIZ STRING OF MENU
3483      ;      R1      MAXIMUM ALLOWABLE OPERATOR RESPONSE
3484      ;
3485      ;OUTPUTS:
3486      ;      R0      NUMBER OF THE OPERATOR'S SELECTION
3487      ;
3488      GETSEL::
3489      SAVREG          ;SAVE GENERAL REGISTERS
3490      MOV R0,R2      ;SAVE THE MENU ADDRESS
3491      MOV R2,R3      ;START OF MENU STRING
3492      TST (R3)      ;END OF ASCII ?
3493      BEQ 3$        ;BRANCH IF ALL LINES DISPLAYED
3494      PRINTF #SELASC,(R3)+ ;DISPLAY THE MENU
3495      MOV #SELASC,-(SP)
3496      MOV #2,-(SP)
3497      MOV SP,R0
3498      TRAP C#PNTF
3499      ADD #6,SP
3500      BR 2$
3501      3$: GMANID MENASC,MENRES,D,-1,0,-1,NO
3502      TRAP C#GMAN
3503      BR 10001$
3504      .WORD MENRES
3505      .WORD T#CODE
3506      .WORD MENASC
3507      .WORD -1
3508      .WORD T#LOLIM
3509      .WORD T#HILIM
3510      10001$: BNCOMPLETE 1$ ;RETRY IF ERROR
3511      BCC 1$
3512      MOV MENRES,R0 ;GET THE OPERATOR'S REPLY
3513      CMP R0,R1 ;COMPARE TO MAXIMUM ALLOWED
3514      BLOS 5$ ;BRANCH IF OK
3515      PRINTF #MENERR ;DISPLAY ERROR MESSAGE
3516      MOV #MENERR,-(SP)
3517      MOV #1,-(SP)
3518      MOV SP,R0
3519      TRAP C#PNTF
3520      ADD #4,SP
3521      BR 1$ ;RETRY
3522      5$: RTS PC ;RETURN TO CALLER
3523      MENERR: .ASCIZ '#N#A *** Menu Selection Too Large ***'
3524      SELASC: .ASCIZ '#N#T'
3525      MENASC: .ASCIZ 'Enter Menu Selection: '
3526      .EVEN
3527      MENRES: .WORD 0

```

CHKMAN - CHECK MANUAL INTERVENTION LEGALITY

```

3510          .SBTTL  CHKMAN  - CHECK MANUAL INTERVENTION LEGALITY
3511          ;*
3512          ;
3513          ;ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
3514          ;
3515          ;INPUT:
3516          ;
3517          ;    NONE.
3518          ;
3519          ;OUTPUT:
3520          ;
3521          ;    CARRY  0      MANUAL INTERVENTION NOT ALLOWED
3522          ;    CARRY  1      MANUAL INTERVENTION IS OK
3523          ;
3524          ;SIDE EFFECTS:
3525          ;
3526          ;    A MESSAGE IS DISPLAYED WARNING THAT TEST IS
3527          ;    NOT EXECUTED IF MANUAL INTERVENTION IS NOT
3528          ;    ALLOWED.
3529          ;
3530          ;-
3531          ;
3532          CHKMAN::
3533          SAVREG          ;SAVE THE REGISTERS
3534          MANUAL        ;SEE IF MANUAL INTERVENTION OK
3535          TRAP  C#MANI
3536          BCOMPLETE 1#   ;BRANCH IF ALLOWED
3537          BCS  1#
3538          PRINTF #NOMAN  ;PRINT THE WARNING MESSAGE
3539          MOV  #NOMAN,-(SP)
3540          MOV  #1,-(SP)
3541          MOV  SP,RO
3542          TRAP C#PNTF
3543          ADD  #4,SP
3544          CLC          ;CLEAR CARRY FOR ERROR
3545          1#: RTS  PC   ;RETURN
3546          NOMAN: .ASCIZ 'N#A *** Manual Intervention not Allowed - Test Aborted ***'.
3547          .even

```

ENVIRN - SETUP FREE DIAGNOSTIC SPACE

```

3543          .SBTTL  ENVIRN  - SETUP FREE DIAGNOSTIC SPACE
3544          ;
3545          ; SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
3546          ;
3547          ENVIRN: MEMORY  R0
020720          TRAP      C#MEM
3548 020722 010037 003114      MOV      R0,FREE      ; GET 1ST FREE ADDRESS...
3549 020726 062737 000002 003114  ADD      #2,FREE
3550 020734 011037 003116      MOV      (R0),FRESIZ ; ...AND WORD COUNT.
3551 020740 162737 000004 003116  SUB      #4,FRESIZ
3552 020746 013702 002012      MOV      L#UNIT,R2   ; GET NUMBER OF UNITS
3553 020752 162737 000007 003116 10#: SUB      #7,FRESIZ   ; TAKE AWAY 7 WORDS PER UNIT
3554 020760 005302          DEC      R2
3555 020762 001373          BNE     10#
3556 020764 013700 003114      MOV      FREE,R0     ;GET FIRST FREE ADDRESS
3557 020770 063700 003116      ADD      FRESIZ,R0   ;POINT TO LAST FREE ADDRESS
3558 020774 162700 000002      SUB      #2,R0       ;BACKUP 1 WORD
3559 021000 010037 003120      MOV      R0,FREEHI  ;STORE LAST FREE ADDRESS
3560 021004 C00240          NOP
3561 021006 012701 177520      MOV      #BDVPCR,R1 ;GET BDV11 PCR ADDRESS
3562 021012 010102          MOV      R1,R2       ;COPY TO R2
3563 021014 062702 000002      ADD      #2,R2       ;SET THE RANGE
3564 021020 004737 016466      JSR     PC,XNXM     ;SEE IF WE HAVE ONE
3565 021024 103001          BCC     15#         ;OK TO SET FLAGS
3566 021026 000423          BR      40#         ;RETURN WITH FLAGS CLEAR
3567 021030 013701 177520      15#: MOV      BDVPCR,R1  ;SAVE PCR CONTENTS
3568 021034 062701 000001      ADD      #1,R1       ;ADD ONE TO IT
3569 021040 012702 177520      MOV      #BDVPCR,R2 ;GET BDV11 PCR ADDRESS
3570 021044 005212          INC      (R2)        ;TRY TO WRITE TO IT
3571 021046 013703 177520      MOV      BDVPCR,R3  ;GET RESULTS
3572 021052 020103          CMP      R1,R3       ;DID IT CHANGE?
3573 021054 001006          BNE     20#         ;NO, MUST BE 11/23B
3574 021056 005237 003134      INC      T23A       ;SET THE FLAG
3575 021062 042737 170000 002120  BIC     #170000,L#HIME ;SUPERVISOR COULD BE WRONG
3576          NOP
3577          ;
3578          ; PRINTF #M8186 ;TELL THE SYSTEM TYPE
3579 021072 005237 003136      20#: BR      40#
3580          ;
3581          ; PRINTF #M8189 ;TELL THE SYSTEM TYPE
3582 021076          40#: INC      T23B
3583 021076 000207          RTS     PC         ;RETURN

```

KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```

3585 .SBTTL KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS
3586
3587 ;
3588 ;ROUTINE TO INIT KT-11
3589 ;
3590 ;-
3591
3592 021100          KTINIT:
3593 021100 005037 003122 CLR      KTFLG      ; INIT >28K MEMORY FLAG
3594 021104 005037 003124 CLR      KTENABLE   ; INIT TEST >28K FLAG
3595 021110 023727 002120 001577 CMP      L#HIME,#1577 ; GOT ENOUGH MEMORY (>28K)?
3596 021116 101454          BLOS     9#           ; NO.
3597 021120 013700 000004 MOV      @ERRVEC,R0  ; SAVE OLD ERR VEC PTR.
3598 021124 012737 021236 000004 MOV      #2#,@ERRVEC ; SET ERR VEC PTR.
3599 021132 005737 177572 TST      @SRO        ; GOT KT11?
3600 021136 000240          MOF      #           ; (TRAP IF NO).
3601 021140 013737 002120 003122 MOV      L#HIME,KTFLG ; YES. SET KT FLAG.
3602 021146 022737 007777 003122 CMP      #7777,KTFLG ; >256K ?
3603 021154 100404          BMI      4#           ; NO
3604 021156 042737 003777 003122 BIC      #3777,KTFLG ; ALIGN ON BOUNDARY
3605 021164 000403          BR       5#
3606 021166 042737 000177 003122 4#: BIC      #177,KTFLG  ;
3607 021174 010037 000004 5#: MOV      R0,@ERRVEC  ; RESTORE OLD ERR VEC PTR.
3608 021200 005000          CLR      R0          ; R0 = AR DATA.
3609 021202 012701 172340 MOV      @KIPAR0,R1  ; F.1 = KI REGS PTR.
3610 021206 012761 077406 177740 1#: MOV      #77406,-40(R1) ; SET DESCRIPTOR REG.
3611 021214 010021          MOV      R0,(R1)+    ; SET KIPAR REG.
3612 021216 062700 000200 ADD      #200,R0     ; BUMP AR DATA BY "4K".
3613 021222 020027 002000 CMP      R0,#2000    ; AT "I/O"?
3614 021226 001367          BNE     1#           ; NO.
3615 021230 012741 177600 MOV      #177600,-(R1) ; YES. SET KTPAR7 FOR I/O.
3616 021234 000405          BR       9#
3617
3618 021236 012716 021244 2#: MOV      #6#,(SP)    ; SET UP RETURN
3619 021242 000002          RTI                    ; RTI TO NEXT LOCATION
3620
3621 021244 010037 000004 6#: MOV      R0,@ERRVEC  ; RESTORE OLD ERR VEC PTR.
3622
3623 021250 000207          RTS      PC

```



KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

3625

KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```
3627      ;*
3628      ;      SUBROUTINE TO SET EXTENDED FEATURES SWITCH
3629      ;
3630      ;      Requires that SOFINIT and WRTCHR have been done previous to call.
3631      ;
3632      ;
3633      ;INPUTS:
3634      ;      R5      CURRENT UNIT NUMBER
3635      ;OUTPUTS:
3636      ;      The Extended Features Switch is set.
3637      ;
3638      ;-
3639
3640
```

## KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```

3642          ;*
3643          ;   SUBROUTINE TO SET EXTENDED FEATURES SWITCH
3644          ;
3645          ;   Requires that SOFINIT and WRTCHR have been done previous to call.
3646          ;
3647          ;
3648          ; INPUTS:
3649          ; R5      CURRENT UNIT NUMBER
3650          ; OUTPUTS:
3651          ; The Extended features Switch is set.
3652          ;
3653          ;-
3654
3655          INVERT::
3656          021252      013705      002176      mov      csraddr,r5
3657          021256      004737      016064      jsr      pc,sof nit
3658          021262      103406      bcs      25f
3659          021264      010001      mov      r0,r1
3660          021266      104455      errrdf   errno,ofierr,sfimg
3661          021270      001513      TRAP    C$ERDF
3662          021272      003650      .WORD   843
3663          021274      012124      .WORD   sfimg
3664          021276      104455      trap cterdf
3665          021300      104406      25f:   ckloop
3666          021302      013737      002172      022140      TRAP    C$CLP1
3667          021310      012704      022120      mov      unitn,t39dsw
3668          021314      004737      010752      mov      @t39pk2,r4
3669          021320      103406      jsr      pc,wrtchr
3670          021322      010001      bcs      50f
3671          021324      104456      mov      r0,r1
3672          021326      001513      errhrd   errno,wrtmsg,sf msg
3673          021330      005054      TRAP    C$ERHRD
3674          021332      012124      .WORD   843
3675          021334      104456      .WORD   wrtmsg
3676          021336      104406      .WORD   sfimg
3677          021340      013701      021450      trap cterhrd
3678          021344      032701      000200      50f:   ckloop
3679          021350      001020      TRAP    C$CLP1
3680          021352      012737      100206      022170      mov      t39bfr+12,r1
3681          021356      012737      022200      022172      bit      @bit7,r1
3682          021360      012737      000006      022176      bne      1f
3683          021364      012737      100010      022200      10f:   MOV     @100206,CMDPKT
3684          021368      004737      010752      MOV     @WSMBK,CMDPKT.2
3685          021372      000207      MOV     @6,CMDPKT.6
3686          021376      012737      100010      MOV     @100010,WSMBK
3687          021380      004737      010752      MOV     @CMDPKT,R4
3688          021384      000207      JSR     PC,WRTCHR
3689          021388      000207      RTS     PC
3690          021392      000000      ; WRT SUB-SYS MEM CMD
3691          021396      000000      ; MSG BUF ADDR
3692          021400      000000      ; BYTE COUNT
3693          021404      000000      ; INVERT THE SWITCH
3694          021408      000000      ; SET CMDPKT INTO R4
3695          021412      000000      ; DO IT
3696          021416      000000      ; RETURN
3697          021420      000000      T39DLY: .WORD 0
3698          021424      000000      ; DELAY COUNTER FOR TEST
3699          021428      000000      T39PACKET:
3700          021432      000000      .WORD 1400J6
3701          021436      000000      ; COMMAND PACKET FOR TEST
3702          021440      000000      .WORD T39TAD
3703          021444      000000      ; WRITE SUBSYSTEM MEM. CMD, ACK,CVC=1
3704          021448      000000      .WORD 0
3705          021452      000000      ; ADDRESS OF CHARACTERISTICS BLOCK
3706          021456      000012      .WORD 10.
3707          021460      000012      ; STARTING VALUE OF BLOCK SIZE

```

KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```

3691 021430          T39TAD:          ;CHARACTERISTICS DATA BLOCK
3692 021430          T398S0: .BYTE 0          ;BSEL0 BYTE
3693 021431          T398S1: .BYTE 0          ;BSEL1 BYTE
3694 021432          T398S2: .WORD 0          ;BSEL1 WORD
3695 021434          .WORD 0          ;DATA
3696 021436          T398FR: .BLKW 150.       ;MESSAGE BUFFER
3697
3698
3700                .=<..+10>&177770
3702 022120          T39PK2:          ;COMMAND PACKET FOR TEST
3703 022120          .WORD 140004          ;WRITE CHARA. MEM. CMND., ACK.CVC=1
3704 022122          .WORD T39DTA          ;ADDRESS OF SELECT DATA BLOCK
3705 022124          .WORD 0
3706 022126          .WORD 10.           ;STARTING VALUE OF BLOCK SIZE
3707
3708
3709 022130          T39DTA:          ;SELECT DATA BLOCK
3710 022130          .WORD T398FR          ;ADDRESS OF MESSAGE BUFFER
3711 022132          .WORD 0
3712 022134          .WORD 256.          ;LENGTH OF MESSAGE BUFFER
3713 022136          T39EAI: .WORD 0          ;EAI BIT WORD
3714 022140          T39DSW: .WORD 0          ;DRIVE SELECT WORD ETC
3716                .=<..+10>&177770
3718 022150          T39PK3: .WORD 140012          ;MESSAGE BUFFER RELEASE COMMAND
3719 022152          .WORD 0          ;NOT USED
3720
3721                ;WRITE TAPE PACKET
3722                ;
3724                .=<..+10>&177770
3726 022160          T39PK4: .WORD 140005          ;WRITE, ACK, CVC=1 COMMAND
3727 022162          T39WR: .WORD 0          ;ADDRESS OF WRITE BUFFER
3728 022164          .WORD 0          ;MORE ADDRESS OF WRITE BUFFER
3729 022166          T39SIZ: .WORD 256.       ;SIZE OF RECORD
3730
3731                ; COMMAND PACKET.
3732
3733                . = <..+3>&177774          ;MUST BE ON MOD 4 BOUNDARY.
3734
3735 022170          CMDPKT.: 0          ;1ST WORD IS TS05 COMMAND.
3736 022172          0          ;2ND WORD IS THE BUFFER LOW ADDRESS.
3737 022174          0          ;3RD WORD IS THE BUFFER HIGH ADDRESS.
3738 022176          0          ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.
3739
3740                ; WRITE SUB-SYSTEM MEMORY CHARACTERISTIC BLOCK.
3741
3742 022200          WSMBK.: 0          ;1ST WORD: SEL 0
3743 022202          0          ;2ND WORD: SEL 2
3744 022204          0          ;3RD WORD: SEL 4
3745                .EVEN
3746
3747                ;*
3748                ; SUBROUTINE TO CHECK WETHER OR NOT WE'LL TEST NXM
3749                ;
3750                ; INPUTS:
3751                ; OUTPUTS:
3752                ; The NXMFLG is set if we can test.
3753                ; The NXML0 and NXMHI addresses are setup.

```

KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```

3754
3755
3756 022206
3757
3758 022206 SAVREG ;SAVE THE REGISTERS
3759 022212 005037 003126 CLR NXMFLG ;CLEAR THE FLAG
3760 022216 005037 003130 CLR NXMLO ;CLEAR THE TEST ADDRESS LO
3761 022222 005037 003132 CLR NXMHI ;CLEAR THE TEST ADDRESS HI
3762 022226 005737 003136 TST T23B ;IS IT A 11/23B?
3763 022232 001407 BEQ 1# ;NO
3764 022234 023727 002120 007777 CMP L#HIME,#7777 ; GREATER THAN 128K
3765 022242 103406 BLO 2# ; NO
3766 022244 004737 022362 JSR PC,NXMTST ;SETUP THE ADDRESS
3767 022250 000427 BR 13# ;SET THE FLAG AND EXIT
3768 022252 005737 003134 1# : TST T23A ;IS IT A 11/23A?
3769 022256 001413 BEQ 4# ;NO
3770 022260 023727 002120 005777 2# : CMP L#HIME,#5777 ;GREATER THAN 96K
3771 022266 101023 BHI 14# ;YES,23A/23B WITH 128K MEMORY
3772 022270 023727 002120 003777 CMP L#HIME,#3777 ;GREATER THAN 64K BUT LESS THAN 92K?
3773 022276 103403 BLO 4# ;NO, CHECK 24K
3774 022300 004737 022362 JSR PC,NXMTST ;SETUP THE ADDRESS
3775 022304 000411 BR 13# ;SET THE FLAG AND EXIT
3776 022306 023727 002120 001577 4# : CMP L#HIME,#1577 ;GREATER THAN 24K BUT LESS THAN 64K?
3777 022314 103410 BLO 14# ;NO, TELL THEM AND EXIT WITH FLAG CLEAR
3778 022316 004737 022362 JSR PC,NXMTST ;SETUP THE ADDRESS
3779 022322 062737 000077 003132 ADD #77,NXMHI ;FOOL THE 11/02 & 11/03
3780 022330 005237 003126 13# : INC NXMFLG ;SET THE FLAG
3781 022334 000411 BR 15# ;EXIT
3782 022336 000410 14# : BR 15# ;NOP FOR PRINTOUT
3783 022340 PRINTF #NOMEM ;TELL THEM & EXIT ***NO PRINT*****
      022340 012746 005456 MOV #NOMEM,-(SP)
      022344 012746 000001 MOV #1,-(SP)
      022350 010600 MOV SP,R0
      022352 104417 TRAP C#PNTF
      022354 062706 000004 ADD #4,SP
3784 022360 000207 15# : RTS PC ;RETURN
3785
3786
3787
3788
3789
3790
3791
3792
3793 022362 013701 002120 NXMTST: MOV L#HIME,R1 ;GET TOP OF MEMORY
3794 022366 062701 000200 ADD #200,R1 ;MAKE IT I/O BLOCK OR OTHER NXM
3795 022372 042701 000177 BIC #177,R1
3796 022376 010102 MOV R1,R2 ;RESAVE RESULTS
3797 000006 .REPT 6
3798 ASL R1 ;PUT IN PLACE FOR XFER
3799 .ENDR
3800 022414 010137 003130 MOV R1,NXMLO ;SAVE TEST ADDRESS LOW
3801 000012 .REPT 10
3802 ASR R2 ;PUT IN PLACE FOR XFER
3803 .ENDR
3804 022444 042702 177700 BIC #177700,R2 ;DON'T WANT ILA!
3805 022450 010237 003132 MOV R2,NXMHI ;SAVE TEST ADDRESS HIGH

```

KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```
3806 022454 000207                    RTS    PC                    ;RETURN
3807
3808 022456                    ENDMOD
3817                    .TITLE    TSV4    MISCELLANEOUS SECTIONS
3818
3819 022456                    BGNMOD    TSV4
      022456                    TSV4::
3820
3826
3827
3828
3829                    .SBTTL    PROTECTION TABLE
3830 022456                    BGNPROT
      022456                    L#PROT::
3831 022456                    .WORD    -1, -1, 1, -1                    ;NO DEVICE PROTECTION REQUIRED.
3832 022466                    ENDPROT
      177777    177777    177777
```

INITIALIZE SECTION

```

3834 .SBTTL INITIALIZE SECTION
3835
3836 ;**
3837 ;THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
3838 ;AT THE BEGINNING OF EACH PASS.
3839 ;
3840 ;IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
3841 ;IF "CONTINUE", NOTHING IS REQUIRED.
3842 ;
3843 ;--
3844 ;*
3845 ;INSERT TEMPORARY JUMP TO ODT
3846 ;-
3847 022466 .BGNINIT
022466 L$INIT::
3848 022466 005037 002216 40$: CLR EXTFEA
3849 022472 005037 003126 CLR NXMFLG
3850 022476 012737 006356 002170 MOV #EPRT1,EPRTSW ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
3851 022504 005037 003144 CLR SIFLAG ;CLEAR "SOFT INIT" FLAG
3852 022510 005037 003124 CLR KTENABLE ;CLEAR TEST ABOVE 28K FLAG
3853 022514 005037 002272 CLR RAMSIZ ;CLEAR RAM SIZE FOR RAMERR ROUTINE
3854 022520 READEF #EF.CONTINUE
022520 012700 000036 MOV #EF.CONTINUE,R0
022524 104447 TRAP C$REFG
3855 022526 BNCOMplete 1$
022526 103023 BCC 1$
3856 022530 023737 002172 002012 CMP UNITN,L$UNIT ;UNIT IN RANGE?
3857 022536 103070 BHIS 4$ ;BR IF NO.
3858 022540 005737 003102 TST DUFLG ;DROPPED UNIT?
3859 022544 100472 BMI NXTU ;BR IF YES
3860 022546 013701 002172 MOV UNITN,R1
3861 022552 006301 ASL R1
3862 022554 005761 003166 TST ERTABL(R1)
3863 022560 001516 BEQ SETU
3864 022562 032761 040000 003166 BIT #BIT14,ERTABL(R1) ;DROPPED?
3865 022570 001060 BNE NXTU
3866 022572 EXIT INIT ;DO NOTHING IF "CONTINUE".
022572 104432 TRAP C$EXIT
022574 000416 .WORD L10030-.
3867 022576 1$: READEF #EF.NEW
022576 012700 000035 MOV #EF.NEW,R0
022602 104447 TRAP C$REFG
3868 022604 BNCOMplete NXTU ;TAKE NEXT UNIT IF NOT NEW PASS.
022604 103052 BCC NXTU
3869 022606 READEF #EF.START
022606 012700 000040 MOV #EF.START,R0
022612 104447 TRAP C$REFG
3870 022614 BCOMplete 2$
022614 103404 BCS 2$
3871 022616 READEF #EF.RESTART
022616 012700 000037 MOV #EF.RESTART,R0
022622 104447 TRAP C$REFG
3872 022624 BNCOMplete 31$
022624 103031 BCC 31$
3873 022626 2$: BRESET ;1ST PASS, BUS-INIT...
3874 022626 104433 TRAP C$RESET ;BUS RESET.

```

INITIALIZE SECTION

```

3875 022630 005037 002204      CLR      TSTCNT      ;NUMBER OF TESTS RUN IN PASS
3876 022634 005037 002212      CLR      FATFLG     ;CLEAR FATAL ERROR COUNT
3877 022640 005037 003134      CLR      T23A      ;CLEAR 11/23A FLAG
3878 022644 005037 003136      CLR      T23B      ;CLEAR 11/23B FLAG
3879      :      MOV      #340,-(SP)
3880      :      MOV      #20,-(SP)      ;RETURN TO DEBUGGER
3881      :      JMP      0.ODT      ;ENTER THE DEBUGGER
3882 022650 005037 003370      CLR      SKIPT      ;CLEAR THE SUBTEST "SKIPPER"
3883 022654      :      :
3884 022654 012737 177777 002174 20$:      MOV      #-1,QVP      ;...QUICK VERIFY...
3885 022662 004737 020720      JSR      PC,ENVIRN   ;SET ENVIRONMENT.
3886 022666 004737 021100      JSR      PC,KTINIT  ;INITIALIZE KT MEMORY MANAGEMENT
3887 022672 012700 003166      MOV      #ERTABL,RO
3888 022676 005020 30$:      CLR      (RO)+      ;CLEAR THE ERROR TABLE
3889 022700 020027 003366      CMP      RO,#ERTABE
3890 022704 103774      BLO     30$
3891 022706 000404      BR      4$
3892 022710 005037 002174 31$:      CLR      QVP
3893 022714 000137 022764      JMP      PASRPT     ;GO REPORT THE STATUS
3894
3895 022720      4$:
3896 022720 012737 177777 002172 NEWPAS: MOV      #-1,UNITN   ;INIT UNIT NUMBER...
3897 022726 005037 002210      CLR      DEVCNT     ;CLEAR COUNT OF DEVICES RUNNING
3898 022732      NXTU:      BREAK
3899 022734 005237 002172      TRAP     C#BRK
3900 022740 023737 002172 002012      INC      UNITN      ;...AND SET NEXT UNIT NUMBER.
3901 022746 103423      CMP      UNITN,L#UNIT
3902 022750 012737 177777 003102      BLO     SETU
3903 022756 000401      MOV      #-1,DUFLG
3904 022760      BR      11$
3905 022762 000240      DOCLN   C#DCLN
3906 022764      TRAP     C#DCLN
3907 022764 023727 002012 000001 11$:      NOP
3908 022772 101752      PASRPT: CMP      L#UNIT,#1   ;HOW MANY UNITS SELECTED?
3909 022774 005737 002210      BLOS    NEWPAS      ;BR IF ONLY 1
3910 023000 001747      TST     DEVCNT      ;ARE ANY STILL RUNNING?
3911 023002      BEQ     NEWPAS      ;BR IF NO
3912 023002 104421      RFLAGS  RO
3913 023004 032700 000100      TRAP     C#RFLA
3914 023010 001343      BIT     #ISR,RO    ;SHOULD WE PRINT STATISTICS
3915 023012      BNE     NEWPAS      ;BR IF NO
3916 023012 104424      DORPT   C#DRPT
3917 023014 000741      TRAP     C#DRPT
3918 023016      BR      NEWPAS
3919 023016      10$:
3920 023022 104442      SETU:   GPHARD  UNITN,RO   ;GET UNIT N P-TABLE POINTER.
3921 023024 103342      MOV     UNITN,RO
3922 023026 005037 003102      TRAP     C#GPHRD
3923 023032 005237 002210      BNCOMPLETE NXTU    ;BR IF UNIT NOT AVAILABLE.
3924 023036 012001      BCC     NXTU
3925 023040 010137 002176      CLR     DUFLG      ;CLEAR "DROPPED" FLAG.
3926      INC     DEVCNT
3927      MOV     (RO)+,R1   ;GET 1ST REGISTER ADDRESS.
3928      MOV     R1,CSRADDR ;ADDRESS OF REGISTERS OF UNIT UNDER TEST

```



INITIALIZE SECTION

```

3925
3926 023044 012001      MOV      (R0)+,R1      ;GET VECTOR ADDRESS.
3927                  ;MOV      (R0),R2      ;GET INTERRUPT PRIORITY
3928                  ;MOV      R2,IPRI      ;SET INTERRUPT PRIORITY.
3929 023046 010137 002200  MOV      R1,IVEC      ;SET INTERRUPT VECTOR POINTER...
3930 023052 012721 016306  MOV      @INTR,(R1)+  ;...VECTOR...
3931 023056 013721 002202  MOV      IPRI,(R1)+  ;...AND PRIORITY.
3932
3933 023062          1$:
3934                  ;      TST      QVP          ;1ST PASS ??
3935                  ;      BEQ      5$          ;NO, SKIP THE PASS 1 STUFF.
3936
3937
3938                  ;
3939                  ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
3940                  ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
3941 023062 013701 002172      MOV      UNITN,R1
3942 023066 006301          ASL      R1
3943 023070 C52761 100000 003166  BIS      @BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
3944 023076 005037 005770      CLR      EXTA          ;CLEAR ERROR EXTENSION FLAG.
3945 023102 023727 002012 000001  CMP      L$UNIT,#1     ;ARE WE TESTING MULTIPLE UNITS?
3946 023110 101416          BLOS    10$           ;BR IF NO.
3947 023112          RFLAGS  R0          ;YES -- GET OPERATOR FLAGS.
3948 023112 104421          TRAP    C$RFLA
3949 023114 032700 001000      BIT      @PNT,R0      ;SHOULD WE PRINT UNIT #?
3950 023120 001412          BEQ      10$           ;BR IF NOT.
3951 023122          PRINTF  @PUNIT,UNITN ;PRINT THE UNIT #
3952 023122 013746 002172      MOV      UNITN,-(SP)
3953 023126 012746 023214      MOV      @PUNIT,-(SP)
3954 023132 012746 000002      MOV      #2,-(SP)
3955 023136 010600          MOV      SP,R0
3956 023140 104417          TRAP    C$PNTF
3957 023142 062706 000006      ADD      #6,SP
3958 023146          10$:
3959 023146 005037 003104      CLR      NODEV
3960 023152 013701 002176      MOV      CSRADDR,R1  ;ADDRESS OF FIRST REGISTER
3961 023156 010102          MOV      R1,R2      ;START OF REGISTERS
3962 023160 062702 000002      ADD      @TSSR,R2   ;ADDRESS OF TSSR REGISTER
3963 023164 004737 016466      JSR      PC,XXM     ;TEST BOTH CONTROLLER REGISTERS...
3964 023170 103005          BCC     2$          ;...AND BR IF ALL OK.
3965 023172 010137 003104      MOV      R1,NODEV   ;FLAG DEVICE AS NON-EXISTENT
3966 023176 012737 177777 003102  MOV      #-1,DUFLG  ;DROP THIS UNIT.
3967 023204          2$:
3968          ;
3969          ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
3970          5$:
3971 023204          SETPRI  @PRI00      ;ENABLE INTERRUPTS.
3972 023204 012700 000000      MOV      @PRI00,R0
3973 023210 104441          TRAP    C$SPRI
3974 023212          ENDINIT
3975 023212          L10030:
3976 023212 104411          TRAP    C$INIT
3977 023214          045 116 045 PUNIT: .ASCIZ /#N#N#A***** TESTING UNIT #D2#A *****/
3978          .EVEN

```

ADD AND DROP UNITS SECTIONS

.SBTTL ADD AND DROP UNITS SECTIONS

```

3970
3971
3972
3973
3974
3975
3976
3977 023262
      023262
3978 023262 010001
3979 023264 006301
3980 023266 052761 100000 003166
3981 023274 042761 040000 003166
3982 023302
      023302 010046
      023304 012746 023330
      023310 012746 000002
      023314 010600
      023316 104417
      023320 062706 000006
3983 023324
      023324 000167
      023326 000026
3984 023330 045 116 045 14:
3985
3986
3987 023356
      023356
      023356 104452
3988
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999 023360
      023360
4000 023360 012737 177777 003102
4001 023366 010001
4002 023370 006301
4003 023372 052761 140000 003166
4004 023400 000240 000240 000240
4005 023406
      023406 010046
      023410 012746 023434
      023414 012746 000002
      023420 010600
      023422 104417
      023424 062706 000006
4006 023430
      023430 000167
      023432 000030

```

```

;***
; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
; TO BE (A) ADDED TO THE TEST LIST FOR THE FIRST TIME,
; OR (B) RE-INSERTED IF IT HAD BEEN PREVIOUSLY DROPPED.
;--
      BGNAU
L#AU::
      MOV      RO,R1          ; GET UNIT TO BE ADDED (RO)
      ASL      R1            ; MAKE IT A WORD INDEX
      BIS      #100000,ERTABL(R1) ; SET THE "ACTIVE" BIT
      BIC      #40000,ERTABL(R1) ; CLEAR THE "DROPPED" BIT
      PRINTF   #1$,RO
      MOV      RO,-(SP)
      MOV      #1$,-(SP)
      MOV      #2$,-(SP)
      MOV      SP,RO
      TRAP     C#PNTF
      ADD      #6,SP
      EXIT     AU
      .WORD    J#JMP
      .WORD    L10031-2-.
      .ASCIZ   /#N#A UNIT #D#A ADDED/
      .EVEN

      ENDAU          ; UNUSED.
L10031:
      TRAP     C#AU

;***
; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
; TO BE REMOVED FROM THE TEST LIST.
;
; SUPVSR DOES THE "DROPPING". THIS IS JUST TO TELL THE MAN.
; "DROPPED" UNITS ARE RE-SELECTED ON OPERATOR "STA" OR "ADD"
; COMMAND, OTHERWISE REMAIN INACTIVE. THE "DISPLAY" COMMAND
; WILL PRINT ALL DROPPED UNITS, AND THE P-TABLES OF THOSE
; WHICH ARE STILL ACTIVE.
; UPON ENTRY, RO CONTAINS THE UNIT TO BE DROPPED.
      BGNDU
L#DU::
      MOV      #-1,DUFLG
      MOV      RO,R1
      ASL      R1
      BIS      #140000,ERTABL(R1) ; SAY DROPPED
      240,240,240 ; ??????????
      PRINTF   #1$,RO
      MOV      RO,-(SP)
      MOV      #1$,-(SP)
      MOV      #2$,-(SP)
      MOV      SP,RO
      TRAP     C#PNTF
      ADD      #6,SP
      EXIT     DU
      .WORD    J#JMP
      .WORD    L10032-2-.

```

ADD AND DROP UNITS SECTIONS

```

4007 023434      045      116      045 10: .ASCIZ /#N#A UNIT #D#A DROPPED/
4008                                     .EVEN
4009 023464      104453 L10032: ENDDU
                                     TRAP C#DU
4010                                     ;**
4011                                     ; AUTO-DROP CODE SECTION.
4012                                     ;--
4013 023466      013705 002176 L#AUTO: BNAUTO
                                     MOV CSRADDR,R5 ;POINT TO DEVICE REGISTER
4014 023466      012703 000550      MOV #360.,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND
4015 023472      004737 016340      JSR PC,WAITF ;WAIT FOR SSR TO SET
4016 023502      103420      BCS 20# ;LEAVE WHEN SSR IS SET
4017 023504      012727 000372      DELAY 250. ;WAIT FOR .25 SECONDS
                                     MOV #250.,(PC)+
                                     .WORD 0
4018 023510      013727 002116      MOV L#DLY,(PC)+
                                     .WORD 0
                                     DEC -6(PC)
                                     BNE -4
4019 023512      005367 177772      DEC -22(PC)
4020 023516      001367 ;BUMP COUNTER DOWN
                                     BNE -20 ;KEEP GOING
4021 023520      005303      DEC R3 ;TRY AND DROP UNIT
4022 023524      001357      BNE 10#
4023 023526      004737 017272      JSR PC,CKDROP
4024 023532      001357      ENDAUTO ; UNUSED.
4025 023534      001357      L10033:
4026 023540      004737 017272      TRAP C#AUTO
4027 023544      001357
4028 023544      104461

```

CLEAN-UP AND REPORT CODING SECTIONS

```

4025          .SBTTL CLEAN-UP AND REPORT CODING SECTIONS
4026
4027          ;**
4028          ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
4029          ; EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
4030          ; USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
4031          ;--
4032 023546      BGNCLN
4033 023546      L$CLEAN::
4034 023546 013705 002176      MOV      CSRADDR,R5          ;POINT TO DEVICE REGISTER
4035 023552 005737 003102      TST      DUFLG          ;"DROPPED" FLAG IS SET ON...
4036 023556 100405          BMI      1$          ;...AND GROSS CONTROLLER FAULT...
4037          ;...DON'T TRY TO XCT CLEANUP CODE.
4038 023560 012765 000000 000002      MOV      #0,TSSR(R5)      ;DO SOFT INIT
4039 023566 004737 016340          JSR      PC,WAITF
4040 023572          1$:
4041 023572          2$:      ENDCLN
4042 023572          L10034:      TRAP      C$CLEAN
4043          ;**
4044          ; THE REPORT CODING SECTION CONTAINS THE
4045          ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
4046          ;--
4047 023574      BGNRPT
4048 023574      L$RPT::
4049 023574          PRINTS  #DEVSUM
4050 023574          MOV      #DEVSUM,-(SP)
4051 023600          MOV      #1,-(SP)
4052 023604          MOV      SP,R0
4053 023606          TRAP      C$PNTS
4054 023610          ADD      #4,SP
4055 023614          MOV      R2,-(SP)
4056 023616          MOV      R3,-(SP)
4057 023620          MOV      R4,-(SP)
4058 023622          MOV      #ERTABL,R4          ; GET START OF ERROR TABLE.
4059 023626          CLR      R3          ; CLEAR UNIT NUMBER
4060 023630          1$:      MOV      (R4),R2          ; GET ERROR TABLE ENTRY & TEST IT.
4061 023632          BEQ      4$          ; ZERO IF UNIT NOT RUN
4062 023634          BPL      4$
4063 023636          BIT      #BIT14,R2          ; WAS UNIT DROPPED?
4064 023642          BNE      2$          ; BR IF YES
4065 023644          BIC      #+C7777,R2          ; GET ERROR COUNT FIELD
4066 023650          PRINTS  #DEVONL,R3,R2          ; PRINT
4067 023650          MOV      R2,-(SP)
4068 023652          MOV      R3,-(SP)
4069 023654          MOV      #DEVONL,-(SP)
4070 023660          MOV      #3,-(SP)
4071 023664          MOV      SP,R0
4072 023666          TRAP      C$PNTS
4073 023670          ADD      #10,SP
4074 023674          BR      4$
4075 023676          2$:      CMP      R2,#160000          ; WAS UNIT NON-EXISTENT?
4076 023702          BNE      3$          ; BR IF NO
4077 023704          PRINTS  #DEVNXR,R3
4078 023706          MOV      R3,-(SP)
4079          MOV      #DEVNXR,-(SP)

```

CLEAN-UP AND REPORT CODING SECTIONS

```

023712 012746 000002      MOV      #2,-(SP)
023716 010600      MOV      SP,R0
023720 104416      TRAP     C#PNTS
023722 062706 000006      ADD      #6,SP
4064 023726 000431      BR       4#
4065 023730 020227 160001      3# :    CMP      R2,#160001      ; WAS UNIT NOT READY AT STARTUP?
4066 023734 001012      BNE     30#                ; BR IF NO.
4067 023736      PRINTS  #DEVNRD,R3
023736 010346      MOV      R3,-(SP)
023740 012746 024225      MOV      #DEVNRD,-(SP)
023744 012746 000002      MOV      #2,-(SP)
023750 010600      MOV      SP,R0
023752 104416      TRAP     C#PNTS
023754 062706 000006      ADD      #6,SP
4068 023760 000414      BR       4#
4069 023762 042702 170000      30# :   BIC      #1C7777,R2
4070 023766      PRINTS  #DEVDR0,R3,R2
023766 010246      MOV      R2,-(SP)
023770 C10346      MOV      R3,-(SP)
023772 012746 024306      MOV      #DEVDR0,-(SP)
023776 012746 000003      MOV      #3,-(SP)
024002 C10600      MOV      SP,R0
024004 104416      TRAP     C#PNTS
024006 062706 000010      ADD      #10,SP
4071 024012 062704 000002      4# :    ADD      #2,R4
4072 024016 005203      INC      R3
4073 024020 020427 003366      CMP      R4,#ERTABE
4074 024024 103701      BLO     1#
4075 024026 012604      MOV      (SP),R4
4076 024030 012603      MOV      (SP),R3
4077 024032 012602      MOV      (SP),R2
4078 024034      ENDRPT      ; UNUSED.
024034      L10035:
024034 104425      TRAP     C#RPT
4079
4080 024036      045      116      045  DEVSUM: .ASCIZ /#N#ADEVICE STATUS SUMMARY:#N/
4081 024073      045      101      040  DEVONL: .ASCIZ /#A UNIT #D3#A ONLINE, ERRORS = #D#N/
4082 024143      045      101      040  DEVNXR: .ASCIZ /#A UNIT #D3#A DROPPED, NON-EXISTENT REGISTER#N/
4083 024225      045      101      040  DEVNRD: .ASCIZ /#A UNIT #D3#A DROPPED, NOT READY AT STARTUP#N/
4084 024306      045      101      040  DEVDR0: .ASCIZ /#A UNIT #D3#A DROPPED, ERRORS = #D#N/
4085      .EVEN
4086
4087 024356      ENDMOD
4088

```

CLEAN-UP AND REPORT CODING SECTIONS

```

4092          .TITLE  TSV7 - HARDWARE TESTS 1-8
4093
4100
4101 024356    BGNMOD  TSV7
4101 024356    TSV7::
4107
4115          .SBTTL  TEST 1: INITIALIZE 04 TEST
4116
4117          ;*
4118          ; THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE
4119          ; CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITIONS
4120          ; (I.E. LOOPBACK ENABLED, FORCING WRONG PARITY, INVERTING SENSE OF
4121          ; EXTENDED FEATURES SWITCH, ETC.)
4122          ;
4123          ;
4124          ;-
4125 024356    BGNTST
4125 024356
4126 024356    C12737 006356 002170    MOV      @EPR1,EPR1SW      ;SET UP PRIMARY ERROR MESSAGE
4127
4128          ;
4129          ;
4130          ; TEST 1
4131          ;
4132          ;
4133          ;-
4134
4139 024364    004737 016274          JSR      PC,DSBINT        ;DISABLE INTERRUPTS
4140 024370    012700 025314          MOV      @TST21ID,R0     ;ASCII MESSAGE TO IDENTIFY TEST
4141 024374    004737 016600          JSR      PC,TSTSETUP     ;DO INITIAL TEST SETUP
4142 024400    012737 000005 002206    MOV      @5,LOOPCNT     ;PERFORM 5 ITERATIONS
4143 024406
4144 024406    004737 025336    T21LOOP: JSR      PC,T21REST     ;SET COMMAND PACKET
4145 024412    004737 025426          JSR      PC,T21RT2      ;SET UP OTHER COMMAND PACKET
4146
4147          ;*****
4148          ;
4149          ; ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4150          ;
4151          ;*****
4152
4153 024416    012737 176750 024772    MOV      @65000.,T21DLY ;SET DELAY ROUTINE
4154 024424    004737 016064    110:   JSR      PC,SOFINIT    ;DO INITIALIZE ON CONTROLLER
4155 024430    103426          BCS     200             ;BR IF INIT WAS OK
4156 024432          DELAY    250             ;DELAY FOR A REWIND TO FINISH
4156 024432    012727 000250          MOV      @250,(PC)
4156 024436    000000          .WORD   0
4156 024440    013727 002116          MOV      L#DLY,(PC)
4156 024444    000000          .WORD   0
4156 024446    005367 177772          DEC     -6(PC)
4156 024452    001375          BNE     -4
4156 024454    005367 177756          DEC     -22(PC)
4156 024460    001367          BNE     -20
4157 024462    005337 024772          DEC     T21DLY         ;BUMP COUNTER DOWN
4158 024466    001356          BNE     110           ;BR, IF MORE TIME TO GO
4159 024470    005237 002212          INC     FATFLG        ;BUMP COUNT
4163 024474    010001          MOV     R0,R1         ;CONTENTS OF TSSR REGISTER
    
```

TEST 1: INITIALIZE #4 TEST

```

4164 024476          ERRDF  ERRNO,SFIERR,SFIMSG      ;FATAL ERROR TSSR WAS NOT OK
      024476 104455          TRAP  C#ERDF
      024500 000145          .WORD 101
      024502 003650          .WORD SFIERR
      024504 012124          .WORD SFIMSG
4165 024506          20#:
4166 024506 012704 024750      MOV  #T21PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
4167
4168      ;*****
4169      ;
4170      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4171      ;
4172      ;*****
4173
4174 024512 013737 002172 024770      MOV  UNITN,T21DSW      ;SET UP DRIVE NUMBER
4175 024520 004737 010752          JSR  PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
4176 024524 103407          BCS  23#           ;BR, IF COMMAND ISSUED OK
4177 024526 005237 002212          INC  FATFLG      ;BUMP COUNT
4181 024532 C10001          MOV  R0,R1      ;SAVE CONTENTS OF TSSR
4182 024534          ERRHRD ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTIC FAILED
      024534 104456          TRAP  C#ERHRD
      024536 000146          .WORD 102
      024540 005054          .WORD WRTMSG
      024542 012124          .WORD SFIMSG
4183 024544          23#:  CKLOOP
4184 024546 112737 000200 025070      MOVB #200,T21BS0      ;WRITE MISCELLANEOUS CONT/READ STATUS
4185 024554 112737 000010 025071      MOVB #10,T21BS1      ;FUNCTION SELECTION BIT
4186 024562          25#:
4187 024562 012704 025060          MOV  #T21PK2,R4      ;WRITE SUBSYS MEM PACKET
4188 024566 010465 000000          MOV  R4,TSDB(R5)     ;ISSUE COMMAND
4189 024572 004737 016426          JSR  PC,CHKTSSR     ;WAIT FOR SSR
4190 024576 103407          BCS  30#           ;BR, IF NO ERROR
4191 024600 010001          MOV  R0,R1      ;ERROR, SAVE TSSR
4192 024602 005237 002212          INC  FATFLG      ;BUMP COUNT
4196 024606          ERRHRD ERRNO,T21SSR,PKTSSR      ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      024606 104456          TRAP  C#ERHRD
      024610 000147          .WORD 103
      024612 025076          .WORD T21SSR
      024614 012136          .WORD PKTSSR
4197 024616          30#:  CKLOOP      ;SCOPE LOOP
      024616 104406          TRAP  C#CLP1
4198 024620 012765 000000 000002      MOV  #0,TSSR(R5)     ;ISSUE A SOFT INITIALIZE
4199 024626 004737 016340          JSR  PC,WAITF      ;WAIT FOR JUST THE SSR BIT TO SET
4200 024632 016501 000002          MOV  TSSR(R5),R1     ;READ THE TSSR BACK
4201 024636 010102          MOV  R1,R2      ;WORK REGISTER
4202 024640 042702 176377          BIC  #C<HIADDR>,R2  ;CLEAR OUT OTHER BITS
4203 024644 052702 002200          BIS  #SSR!NBA,R2   ;SOME OF THE BITS THAT SHOULD BE SET
4204 024650 032701 000100          BIT  #OFL,R1      ;IS OFF LINE BIT SET
4205 024654 001012          BNE  38#           ;BR, IF DRIVE IS OFF LINE
4206 024656 020102          35#:  CMP  R1,R2      ;EXPECTED (R2) = RECEIVED (R1)
4207 024660 001406          BEQ  37#           ;BR, IF THEY ARE EQUAL (OK)
4208 024662 005237 002212          INC  FATFLG      ;BUMP COUNT
4212 024666          ERRHRD ERRNO,T21AM3,EXPREC      ;"ERROR TRYING TO INIT AFTER WRITE MISC.
      024666 104456          TRAP  C#ERHRD
      024670 000150          .WORD 104
      024672 025173          .WORD T21AM3
    
```

TEST 1: INITIALIZE #4 TEST

4213	024674	015564							
	024676		37:	CKLOOP			; LOOP IF SELECTED	.WORD	EXPREC
4214	024700	000406		BR	40:		; SKIP OVER OFF-LINE STUFF	TRAP	C%CLP1
4215	024702		38:	ERRDF	ERRNO, T21OFL, EXPREC		; DRIVE IS OFF LINE		
4219	024702	104455						TRAP	C%ERDF
	024704	000151						.WORD	105
	024706	025273						.WORD	T21OFL
	024710	015564						.WORD	EXPREC
4220	024712	004737	017272	JSR	PC, CKDROP		; TRY AND DROP UNIT		
4221	024716	000241	40:	CLC			; DON'T LET CARRY SNEAK IN		
4222	024720	106037	025071	RORB	T21BS1		; TRY NEXT "LOWEST" BIT POSITION		
4223	024724	001316		BNE	25:		; LOOP UNTIL ALL EIGHT BITS TESTED		
4224	024726		50:	CKLOOP			; SCOPE LOOP		
4225	024730	004737	016546	JSR	PC, TSTLOOP		; DO WE NEED TO ITERATE TEST	TRAP	C%CLP1
4226	024734	103002		BCC	63:		; BR, IF NO LOOP REQUIRED		
4227	024736	C00137	024406	JMP	T21LOOP		; EXECUTE AGAIN		
4228	024742		63:	EXIT	TST		; ALL DONE THIS TEST	TRAP	C%EXIT
	024742	104432						.WORD	L10036--
	024744	000524							
4229									
4230									
4231									
4232									
4234	024746								
4236	024750								
4237	024750	100004							
4238	024752	024760							
4239	024754	000000							
4240	024756	000012							
4241	024760								
4242	024760	024774							
4243	024762	000000							
4244	024764	000024							
4245	024766	000000							
4246	024770	000000							
4247	024772	000000							
4248	024774								
4249									
4250									
4251									
4253	025056								
4255	025060								
4256	025060	100206							
4257	025062	025070							
4258	025064	000000							
4259	025066	000006							
4260									
4261									
4262	025070								
4263	025070	000							
4264	025071	000							
4265	025072	000000							
4266	025074	000000							
4267									

```

; LOCAL STORAGE FOR THIS TEST
;
;
; T21PACKET:
; .BLKB 10-<. TSV2E7>
; .WORD 100004
; .WORD T21DATA
; .WORD 0
; .WORD 10
; T21DATA:
; .WORD T21BFR
; .WORD 0
; .WORD 20.
; .WORD 0
; T21DSW: .WORD 0
; T21DLY: .WORD 0
; T21BFR: .BLKW 25.
;
; WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
; .BLKB 10-<.-TSV2E7>
; T21PK2:
; .WORD 100206
; .WORD T21BF2
; .WORD 0
; .WORD 6.
;
; .EVEN
; T21BF2:
; T21BS0: .BYTE 0
; T21BS1: .BYTE 0
; T21S2: .WORD 0
; T21S3: .WORD 0
;
; COMMAND PACKET FOR TEST
; WRITE CHARACTERISTICS COMMAND, WITH, ACK
; ADDRESS OF CHARACTERISTICS BLOCK
;
; STARTING VALUE OF BLOCK SIZE
; CHARACTERISTICS DATA BLOCK
; ADDRESS OF MESSAGE BUFFER
;
; LENGTH OF MESSAGE BUFFER
;
; DRIVE SELECT WORD
; DELAY COUNTER
; MESSAGE BUFFER
;
; WRITE SUB SYS MEM COMMAND, IE AND ACK
; ADDRESS OF SELECT BLOCK DATA
;
; SIZE OF DATA PACKET
;
; BSEL0 AREA --- "COMMAND" BYTE
; BSEL1 AREA
; SEL 2 AREA
; DATA AREA
    
```



## TEST 1: INITIALIZE #4 TEST

```

4268
4269
4270      ;*
4271      ;LOCAL TEXT MESSAGES FOR TEST
4272      ;-
4273 025076      127      122      111 T21SSR: .ASCIZ 'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
4274 025173      124      123      123 T21AM3: .ASCIZ 'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
4275 025273      104      162      151 T21OFL: .ASCIZ 'Drive is OFFLINE'
4276 025314      111      156      151 T21ID:  .ASCIZ 'Initialization #4'
4277      .EVEN
4278
4279      ;*
4280      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
4281      ;WRITE SUBSYSTEM MEMORY COMMAND
4282      ;
4283      ;-
4284
4285 025336      T21REST:
4286 025336      SAVREG      ;SAVE THE REGISTERS
4287 025342      012701      024750      MOV      #T21PACKET,R1      ;START OF THE PACKET
4288 025346      012721      100004      MOV      #100004,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK.
4289 025352      012721      024760      MOV      #T21DATA,(R1)+      ;ADDRESS OF CHARAISTICS DATA BLOCK
4290 025356      005021      CLR      (R1)+      ;EXTENDED ADDRESS
4291 025360      012721      000010      MOV      #8,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
4292 025364      012721      024774      MOV      #T21BFR,(R1)+      ;ADDRESS OF MESSAGE BUFFER
4293 025370      005021      CLR      (R1)+
4294 025372      012721      000024      MOV      #20,(R1)+      ;LENGTH OF MESSAGE BUFFER
4295 025376      005021      CLR      (R1)+
4296 025400      005011      CLR      (R1)
4297 025402      012702      000020      MOV      #20,R2      ;NUMBER OF LOCATIONS TO BE CLEARED
4298 025406      012762      177777      024774      64$: MOV      #177777,T21BFR(R2)      ;ALL ONES TO MESSAGE BUFFER
4299 025414      005742      TST      -(R2)      ;NEXT LOCATION
4300 025416      020227      000000      CMP      R2,#0      ;CHECK R2 FOR ZERO
4301 025422      001371      BNE      64$      ;BR, IF NOT AT ZERO YET
4302 025424      000207      RTS      PC      ;RETURN
4303
4304
4305 025426      T21RT2:
4306 025426      SAVREG      ;SAVE THE REGISTERS
4307 025432      012701      025060      MOV      #T21PK2,R1      ;START OF THE PACKET
4308 025436      012721      100206      MOV      #100206,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK, IE
4309 025442      012721      025070      MOV      #T21BF2,(R1)+      ;ADDRESS OF DATA BLOCK
4310 025446      005021      CLR      (R1)+      ;EXTENDED ADDRESS
4311 025450      012721      000006      MOV      #6,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
4312 025454      005021      CLR      (R1)+
4313 025456      012701      025070      MOV      #T21BF2,R1      ;ADDRESS OF DATA FOR WRT SUB SYS MEM
4314 025462      005021      CLR      (R1)+
4315 025464      005011      CLR      (R1)
4316 025466      000207      RTS      PC      ;RETURN
4317 025470      ENDTST
4318      025470      104401      L10036:      TRAP      C#ETST
4319
4320      .SBTTL TEST 2: OFF-LINE AND REJECT REWIND
4321      ;*
4322      ;THIS TEST VERIFIES BASIC TAPE-MOTION COMMAND DECODING AND BASIC

```



TEST 2: OFF-LINE AND REJECT REWIND

```

4380
4381 ;*****
4382
4383 025564 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4384 025570 103407 BCS 23$ ;BR, IF COMMAND ISSUED OK
4385 025572 005237 002212 INC FATFLG ;BUMP COUNT
4389 025576 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4390 025600 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      025600 104456 TRAP C#ERHRD
      025602 000312 .WORD 202
      025604 005054 .WORD WRTMSG
      025606 012124 .WORD SFIMSG
4391 025610 23$: CKLOOP TRAP C#CLP1
      025610 104406
4392 025612 013701 027070 MOV T22BFR+6,R1 ;PICK UP XTSO
4393 025616 032701 000004 BIT #4,R1 ;IS UNIT WRITE-LOCKED?
4394 025622 001407 BEQ 24$ ;NO,PROCEED WITH TESTING
4395 025624 005237 002212 INC FATFLG ;BUMP COUNT
4399 025630 ERRDF ERRNO,T22WLK,SFIMSG ;TAPE IS WRITE LOCKED
      025630 104455 TRAP C#ERDF
      025632 000313 .WORD 203
      025634 027672 .WORD T22WLK
      025636 012124 .WORD SFIMSG
4400 025640 DOCLN TRAP C#DOCLN
      025640 104444
4401 025642 24$: CKLOOP TRAP C#CLP1
      025642 104406
4402 025644 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
4403 025650 001041 BNE 50$ ;BR IF SWITCH IS ON
4404 025652 112737 000200 027161 MOVB #200,T22BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
4405 025660 112737 000010 027160 MOVB #10,T22BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
4406 025666 012704 027150 MOV #T22PK2,R4 ;WRITE SUBSYS MEM PACKET
4407 025672 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
4408 025676 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
4409 025702 103407 BCS 30$ ;BR, IF NO ERROR
4410 025704 010001 MOV RO,R1 ;ERROR, SAVE TSSR
4411 025706 005237 002212 INC FATFLG ;BUMP COUNT
4415 025712 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      025712 104456 TRAP C#ERHRD
      025714 000314 .WORD 204
      025716 027200 .WORD T22SSR
      025720 012136 .WORD PKTSSR
4416 025722 30$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
      025722 104406
4417 025724 012704 027040 MOV #T22PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4418
4419 ;*****
4420 ;
4421 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4422 ;
4423 ;*****
4424
4425 025730 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4426 025734 103407 BCS 50$ ;BR, IF COMMAND ISSUED OK
4427 025736 005237 002212 INC FATFLG ;BUMP COUNT
4431 025742 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4432 025744 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
    
```







TEST 2: OFF-LINE AND REJECT REWIND

```

4579
4580 ;VERIFIES THAT A REWIND COMMAND WITH CVC=1 CLEARS VCK
4581 ;AND RETURNS PROPER STATUS IN THE MESSAGE BUFFER.
4582
4583
4584
4585 026436      ;-----
         BGNSUB                               ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
         026436      T2.3:
4586 026436 104402 004737 030012      JSR    PC,T22REST      ;SET COMMAND PACKET
4587 026444 004737 030104      JSR    PC,T22RT2      ;SET UP OTHER COMMAND PACKET
4588
4589 ;*****
4590 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
4591 ;
4592 ;*****
4593
4594      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
4595 026450 004737 016064      BCS    20$      ;BR IF INIT WAS OK
4596 026454 103407      INC    FATFLG      ;BUMP COUNT
4597 026456 005237 002212      MOV    R0,R1      ;CONTENTS OF TSSR REGISTER
4601 026462 010001      ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
4602 026464      TRAP    C:ERDF
         026464 104455      .WORD  214
         026466 000326      .WORD  SFIERR
         026470 003650      .WORD  SFIMSG
         02f 72 012124
4603 026474      20$:
4604 026474 012704 027040      MOV    #T22PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
4605
4606 ;*****
4607 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4608 ;
4609 ;*****
4610
4611      JSR    PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
4612 026500 004737 010752      BCS    23$      ;BR, IF COMMAND ISSUED OK
4613 026504 103407      INC    FATFLG      ;BUMP COUNT
4614 026506 005237 002212      MOV    R0,R1      ;SAVE CONTENTS OF TSSR
4618 026512 010001      ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
4619 026514      TRAP    C:ERHRD
         026514 104456      .WORD  215
         026516 000327      .WORD  WRTMSG
         026520 005054      .WORD  SFIMSG
         026522 012124
4620 026524 005737 002216      23$: TST    EXTFEA      ;CHECK FOR EXTENDED FEATURES SW SWITCH
4621 026530 001041      BNE    50$      ;BR IF SWITCH IS ON
4622
4623 026532 112737 000200 027161      MOVB   #200,T22BS1      ;WRITE MISCELLANEOUS CONT/READ STATUS
4624 026540 112737 000010 027160      MOVB   #10,T22BS0      ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
4625 026546 012704 027150      MOV    #T22PK2,R4      ;WRITE SUBSYS MEM PACKET
4626 026552 010465 000000      MOV    R4,TSD8(R5)      ;ISSUE COMMAND
4627 026556 004737 016426      JSR    PC,CHKTSSR      ;WAIT FOR SSR
4628 026562 103407      BCS    30$      ;BR, IF NO ERROR
4629 026564 010001      MOV    R0,R1      ;ERROR, SAVE TSSR
4630 026566 005237 002212      INC    FATFLG      ;BUMP COUNT
4634 026572      ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS

```

## TEST 2: OFF-LINE AND REJECT REWIND

```

026572 104456                                TRAP C#ERHRD
026574 000330                                .WORD 216
026576 027200                                .WORD T22SSR
026600 012136                                .WORD PKTSSR
4635 026602 30#: CKLOOP                      ;LOOP IF SELECTED
026602 104406                                TRAP C#CLP1
4636 026604 012704 027040                   MOV #T22PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4637
4638 ;*****
4639 ;
4640 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4641 ;
4642 ;*****
4643
4644 026610 004737 010752                   .JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4645 026614 103407                           BCS 50# ;BR, IF COMMAND ISSUED OK
4646 026616 005237 002212                   INC FATFLG ;BUMP COUNT
4650 026622 010001                           MOV R0,R1 ;SAVE CONTENTS OF TSSR
4651 026624 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
026624 104456                                TRAP C#ERHRD
026626 000331                                .WORD 217
026630 005054                                .WORD WRTMSG
026632 012124                                .WORD SFMSG
4652 026634 50#: CKLOOP                      ;SCOPE LOOP
026634 104406                                TRAP C#CLP1
4653 026636 016501 000002                   MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4654 026642 032701 000100                   BIT #OFL,R1 ;CHECK FOR THE OFFLINE BIT SET
4655 026646 001006                           BNE 60# ;BR, IF OFFLINE (GOOD)
4656 026650 005237 002212                   INC FATFLG ;BUMP COUNT
4660 026654 ERRDF ERRNO,T22OFL,SFMSG ;OFF LINE SHOULD HAVE BEEN SET (BAD)
026654 104455                                TRAP C#ERDF
026656 000332                                .WORD 218
026660 027375                                .WORD T22OFL
026662 012124                                .WORD SFMSG
4661 026664 60#: CKLOOP                      ;LOOP IF SELECTED
026664 104406                                TRAP C#CLP1
4662 026666 012737 142010 027150 65#: MOV #142010,T22PK2 ;POSITION COMMAND (REWIND MODE) CVC=1
4663 026674 012704 027150                   MOV #T22PK2,R4 ;R4 = POINTER TO PACKET
4664 026700 010465 000000                   MOV R4,TSD(B5) ;ISSUE COMMAND
4665 026704 004737 016340                   JSR PC,WAITF ;WAIT FOR SSR TO SET
4666 026710 016501 000002                   MOV TSSR(R5),R1 ;GET TSSR CONTENTS
4667 026714 012702 100306                   MOV #SSR!OFL!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
4668 026720 020102                           CMP R1,R2 ;ARE THEY EQUAL
4669 026722 001406                           BEQ 80# ;BR, IF OK ESP. FUNCTION REJECT
4670 026724 005237 002212                   INC FATFLG ;BUMP COUNT
4674 026730 ERRHRD ERRNO,T22RWJ,EXPREC ;TSSR INCORRECT AFTER TAPE MOTION CMD
026730 104456                                TRAP C#ERHRD
026732 000333                                .WORD 219
026734 027544                                .WORD T22RWJ
026736 015564                                .WORD EXPREC
4675 026740 80#: CKLOOP                      ;LOOP IF SELECTED
026740 104406                                TRAP C#CLP1
4676 026742 012703 027062                   MOV #T22BFR,R3 ;POINTER TO MESSAGE BUFFER
4677 026746 016301 000006                   MOV XSTO(R3),R1 ;PICK UP XSTO FROM MESSAGE BUFFER
4678 026752 010102                           MOV R1,R2 ;SET UP EXPECTED
4679 026754 042702 000020                   BIC #BIT4,R2 ;VCK SHOULD BE CLEAR
4680 026760 020102                           CMP R1,R2 ;ARE THEY EQUAL

```





## TEST 2: OFF-LINE AND REJECT REWIND

```

4737 ;TAPE MOTION PACKET COMMAND VALUES
4738 027166 100201 T22RD: .WORD 100201 ;READ TAPE FORWARD
4739 027170 100205 T22WRT: .WORD 100205 ;WRITE TAPE FORWARD
4740 027172 100210 T22POS: .WORD 100210 ;POSITION TAPE
4741 027174 100211 T22FOR: .WORD 100211 ;FORMAT TAPE
4742 027176 177777 .WORD 177777 ;END OF DATA
4743
4744
4745
4746 ;*
4747 ;LOCAL TEXT MESSAGES FOR TEST
4748 ;-
4749 027200 127 122 111 T22SSR: .ASCIZ 'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
4750 027275 124 123 123 T22AM3: .ASCIZ 'TSSR Init. Failed After WRITE MISCELLANEOUS CONTROL/READ STATUS'
4751 027375 104 162 151 T22OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
4752 027450 124 123 123 T22TM: .ASCIZ 'TSSR Incorrect After Tape Motion Command To Off-Line Device'
4753 027544 124 123 123 T22RWJ: .ASCIZ 'TSSR Not Correct After REWIND With VCK Set'
4754 027617 103 126 103 T22VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
4755 027672 052 052 052 T22WLK: .ASCIZ '*****TAPE IS WRITE-LOCKED AND WILL CAUSE ERRORS*****'
4756 027757 117 146 146 T22ID: .ASCIZ 'Off-Line And Reject Rewind'
4757 .EVEN
4758 ;*
4759 ;
4760 ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
4761 ;WRITE SUBSYSTEM MEMORY COMMAND
4762 ;
4763 ;-
4764
4765 030012 T22REST:
4766 030012 SAVREG ;SAVE THE REGISTERS
4767 030016 012701 027040 MOV #T22PACKET,R1 ;START OF THE PACKET
4768 030022 012721 100204 MOV #100204,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK, IE
4769 030026 012721 027050 MOV #T22DATA,(R1)+ ;ADDRESS OF CHARAISTICS DATA BLOCK
4770 030032 005021 CLR (R1)+ ;EXTENDED ADDRESS
4771 030034 012721 000012 MOV #10,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
4772 030040 012721 027062 MOV #T22BFR,(R1)+ ;ADDRESS OF MESSAGE BUFFER
4773 030044 005021 CLR (R1)+
4774 030046 012721 000024 MOV #20,(R1)+ ;LENGTH OF MESSAGE BUFFER
4775 030052 005021 CLR (R1)+
4776 030054 012711 000007 MOV #7,(R1) ;SELECT DRIVE SEVEN
4777 030060 012702 000020 MOV #20,R2 ;NUMBER OF LOCATIONS TO BE CLEARED
4778 030064 012762 177777 027062 64: MOV #177777,T22BFR(R2) ;ALL ONES TO MESSAGE BUFFER
4779 030072 005742 TST -(R2) ;BUMP R2 DOWN
4780 030074 020227 000000 CMP R2,#0 ;IS R2 AT ZERO YET
4781 030100 001371 BNE 64: ;KEEP GOING UNTIL DONE
4782 030102 000207 RTS PC ;RETURN
4783
4784
4785 030104 T22RT2:
4786 030104 SAVREG ;SAVE THE REGISTERS
4787 030110 012701 027150 MOV #T22PK2,R1 ;START OF THE PACKET
4788 030114 012721 100206 MOV #100206,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK, IE
4789 030120 012721 027160 MOV #T22BF2,(R1)+ ;ADDRESS OF DATA BLOCK
4790 030124 005021 CLR (R1)+ ;EXTENDED ADDRESS
4791 030126 012721 000006 MOV #6,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
4792 030132 005021 CLR (R1)+
4793 030134 012701 027160 MOV #T22BF2,R1 ;POINT TO DATA SEL AREA

```



TEST 3: BASIC WRITE DATA

```

4850 030234 004737 016064          JSR    PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
4851 030240 103407                  BCS    20$                 ;BR IF INIT WAS OK
4852 030242 005237 002212          INC    FATFLG             ;BUMP COUNT
4856 030246 010001                  MOV    R0,R1              ;CONTENTS OF TSSR REGISTER
4857 030250                  ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   301
                                .WORD   SFIERR
                                .WORD   SFIMSG
4858 030260                  20$:
4859 030260 012737 000007 033460      MOV    #7,T23DSW          ;SET DRIVE NUMBER IN PACKET
4860 030266 012704 033440      MOV    #T23PACKET,R4     ;SUBROUTINE NEEDS PACKET ADDRESS
4861
4862          ;*****
4863          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4864          ;
4865          ;*****
4866
4867
4868 030272 004737 010752          JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
4869 030276 103407                  BCS    23$                 ;BR, IF COMMAND ISSUED OK
4870 030300 005237 002212          INC    FATFLG             ;BUMP COUNT
4874 030304 010001                  MOV    R0,R1              ;SAVE CONTENTS OF TSSR
4875 030306                  ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C$ERHRD
                                .WORD   302
                                .WORD   WRTMSG
                                .WORD   SFIMSG
4876 030316 005737 002216      23$:  TST    EXTFEA           ;CHECK FOR EXTENDED FEATURES SW SWITCH
4877 030322 001044                  BNE    50$                 ;BR IF SWITCH IS ON
4878
4879 030324 112737 000200 033603      MOVB   #200,T23B51        ;WRITE MISCELLANEOUS CONT/READ STATUS
4880 030332 112737 000010 033602      MOVB   #10,T23B50         ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
4881 030340 012704 033550      MOV    #T23PK2,R4         ;WRITE SUBSYS MEM PACKET
4882 030344 010465 000000      MOV    R4,TSDB(R5)        ;ISSUE COMMAND
4883 030350 004737 016426      JSR    PC,CHKTSSR         ;WAIT FOR SSR
4884 030354 103407                  BCS    30$                 ;BR, IF NO ERROR
4885 030356 010001                  MOV    R0,R1              ;ERROR, SAVE TSSR
4886 030360 005237 002212          INC    FATFLG             ;BUMP COUNT
4890 030364                  ERRHRD ERRNO,T23SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
                                TRAP    C$ERHRD
                                .WORD   303
                                .WORD   T23SSR
                                .WORD   PKTSSR
4891 030374                  30$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    L$CLP1
4892 030376 012737 000007 033460      MOV    #7,T23DSW          ;SET DRIVE NUMBER IN PACKET
4893 030404 012704 033440      MOV    #T23PACKET,R4     ;SUBROUTINE NEEDS PACKET ADDRESS
4894
4895          ;*****
4896          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4897          ;
4898          ;*****
4899
4900
4901 030410 004737 010752          JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
4902 030414 103407                  BCS    50$                 ;BR, IF COMMAND ISSUED OK

```





TEST 3: BASIC WRITE DATA

```

5006 030706 012124          23$:  CKLOOP          ;LOOP IF SELECTED          .WORD  SFIMSG
      030710 104406          TRAP  C:CLP1
5007
5008 ;*****
5009 ;
5010 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5011 ;
5012 ;*****
5013
5014 030712 004737 011104          JSR  PC,REWIND          ;CALL THE TAPE REWIND
5015 030716 012703 000024          MOV  #20,R3             ;STARTING RECORD SIZE
5016 030722 013737 003114 033572 65$:  MOV  FREE,T23WB         ;STARTING WRITE BUFFER ADDRESS
5017
5018 ;*****
5019 ;
5020 ;WRITE DATA,CVC=1,ACK COMMAND
5021 ;
5022 ;*****
5023
5024 030730 012737 140005 033570          MOV  #140005,T23PK3     ;WRITE DATA,CVC=1,ACK COMMAND
5025 030736 012737 140005 033612          MOV  #140005,T23WRT     ;SETUP FOR RETRY COMMAND
5026 030744 052737 004000 033612          BIS  #4000,T23WRT      ;MAKE IT A RETRY
5027 030752 012704 033570          MOV  #T23PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
5028 030756 010300          MOV  R3,R0             ;SET PATTERN IN CORRECT REGISTER
5029 030760 004737 017512          JSR  PC,FILLMEM         ;FILL MEMORY WITH RECORD SIZE
5030 030764 010337 033576          MOV  R3,T23SZ          ;SET UP RECORD SIZE IN PACKET
5031 030770 010465 000000          MOV  R4,TSDB(R5)       ;ISSUE COMMAND
5032 030774 004737 016340          JSR  PC,WAITF           ;WAIT FOR SSR TO SET
5033 031000 016501 000002          MOV  TSSR(R5),R1       ;GET TSSR CONTENTS
5034 031004 012702 000200          MOV  #SSR,R2           ;SET UP EXPECTED
5035 031010 020102          CMP  R1,R2             ;ARE THEY EQUAL
5036 031012 001402          BEQ  80$               ;BR, IF OK
5037 031014 004737 035142          JSR  PC,T23CHK         ;CHECK SPECIAL CONDITION
5038 031020          80$:  CKLOOP          ;LOOP IF SELECTED          TRAP  C:CLP1
      031020 104406
5039 031022 016501 000000          MOV  TSBA(R5),R1       ;GET TSBA CONTENTS
5040 031026 012702 033462          MOV  #T23BFR,R2       ;SET UP EXPECTED
5041 031032 062702 000016          ADD  #16,R2            ;SET TO END OF MESSAGE BUFFER
5042 031036 005737 002216          TST  EXTFEA            ;CHECK FOR EXTENDED FEATURES SW SET
5043 031042 001402          BEQ  85$               ;BR, IF IT NOT SET
5044 031044 062702 000002          ADD  #2,R2             ;BUMP R2 FOR EXTRA DATA
5045 031050 020102          85$:  CMP  R1,R2            ;ARE THEY EQUAL
5046 031052 001406          BEQ  90$               ;BR, IF TSBA IS CORRECT
5047 031054 005237 002212          INC  FATFLG            ;BUMP COUNT
5051 031060          ERRHRD  ERRNO,T23BA,EXPREC ;TSBA WAS NOT CORRECT AFTER WRITE DATA
      031060 104456          TRAP  C:ERHRD
      031062 000465          .WORD  309
      031064 034605          .WORD  T23BA
      031066 015564          .WORD  EXPREC
5052 031070          90$:  CKLOOP          ;LOOP IF SELECTED          TRAP  C:CLP1
      031070 104406
5053 031072 020327 007376          CMP  R3,#7376          ;ONLY CHECK RAM UNTIL ITS FULL
5054 031076 002114          BGE  115$              ;IT WRAPS AROUND ETC.
5055 031100 004737 035054          JSR  PC,T23RT2         ;MAKE SURE PACKET AND DATA ARE CLEAN
5056 031104 012737 000400 033604          MOV  #256,T23S2        ;STARTING RAM ADDRESS
5057 031112 112737 000000 033602          MOVB #0,T23BS0         ;STOP INTERNAL TSV05 DIAGNOSTICS

```

## TEST 3: BASIC WRITE DATA

5058	031120	112737	000000	033603	MOVB	#0,T23BS1	;SIZE OF RAM READ
5059	031126	012704	033550		MOV	#T23PK2,R4	;SET R4 WITH PACKET ADDRESS
5060	031132	010465	000000		MOV	R4,TSDB(R5)	;ISSUE WRITE SUB SYS MEM COMMAND
5061	031136	004737	016426		JSR	PC,CHKTSSR	;CHECK TSSR AND WAIT FOR SSR TO SET
5062	031142	103407			BCS	92#	;BR, IF NO ERRORS IN TSSR
5063	031144	010001			MOV	R0,R1	;SAVE TSSR
5064	031146	005237	002212		INC	FATFLG	;BUMP COUNT
5068	031152				ERRHRD	ERRNO,T23WSS,PKTSSR	;TSSR BAD AFTER WRITE SUB SYS MEM
	031152	104456					TRAP C#ERHRD
	031154	000466					.WORD 310
	031156	034657					.WORD T23WSS
	031160	012136					.WORD PKTSSR
5069	031162			92#:	CKLOOP		;LOOP IF SELECTED
	031162	104406					TRAP C#CLP1
5070	031164	004737	035054		JSR	PC,T23RT2	;MAKE SURE PACKET AND DATA ARE CLEAN
5071	031170	012737	000400	033604	MOV	#256.,T23S2	;STARTING RAM ADDRESS
5072	031176	112737	000001	033602	MOVB	#1,T23BS0	;READ RAM COMMAND FOR WRITE SUB SYS M.
5073	031204	112737	000002	033603	MOVB	#2,T23BS1	;SIZE OF RAM READ
5074	031212	C12704	033550		MOV	#T23PK2,R4	;SET R4 WITH PACKET ADDRESS
5075	031216	010465	000000	95#:	MOV	R4,TSDB(R5)	;ISSUE WRITE SUB SYS MEM COMMAND
5076	031222	004737	016426		JSR	PC,CHKTSSR	;CHECK TSSR AND WAIT FOR SSR TO SET
5077	031226	103407			BCS	100#	;BR, IF NO ERRORS IN TSSR
5078	031230	010001			MOV	R0,R1	;SAVE TSSR
5079	031232	005237	002212		INC	FATFLG	;BUMP COUNT
5083	031236				ERRHRD	ERRNO,T23WSS,PKTSSR	;TSSR BAD AFTER WRITE SUB SYS MEM
	031236	104456					TRAP C#ERHRD
	031240	000467					.WORD 311
	031242	034657					.WORD T23WSS
	031244	012136					.WORD PKTSSR
5084	031246			100#:	CKLOOP		;LOOP IF SELECTED
	031246	104406					TRAP C#CLP1
5085	031250	005001			CLR	R1	;CLEAR REGISTER
5086	031252	005002			CLR	R2	;CLEAR REGISTER
5087	031254	013701	033502		MOV	T23BFR+20,R1	;PICK UP BYTE READ FROM RAM
5088	031260	010302			MOV	R3,R2	;SET UP EXPECTED
5089	031262	020102			CMP	R1,R2	;IS RAM DATA CORRECT
5090	031264	001406			BEQ	110#	;BR, IF OK (EQUAL)
5091	031266	005237	002212		INC	FATFLG	;BUMP COUNT
5095	031272				ERRHRD	ERRNO,T23RNC,EXPREC	;RNC-RAM NOT CORRECT
	031272	104456					TRAP C#ERHRD
	031274	000470					.WORD 312
	031276	034145					.WORD T23RNC
	031300	015564					.WORD EXPREC
5096	031302			110#:	CKLOOP		;LOOP IF SELECTED
	031302	104406					TRAP C#CLP1
5097	031304	005237	033604		INC	T23S2	;BUMP RAM ADDRESS TO BE CHECKED
5098	031310	005237	033604		INC	T23S2	;BUMP RAM ADDRESS TO BE CHECKED
5099	031314	010301			MOV	R3,R1	;GET SIZE OF RECORD
5100	031316	062701	000400		ADD	#256.,R1	;FIGURE OUT END RECORD ADDRESS
5101	031322	023701	033604		CMP	T23S2,R1	;AT END OF RAM CHECK YET
5102	031326	001333			BNE	95#	;BR, IF MORE TO CHECK
5103	031330	062703	001750	115#:	ADD	#1000.,R3	;NEXT RECORD SIZE/DATA PATTERN
5104	031334	020337	033600		CMP	R3,T23RSZ	;IS R3 OVER MAX RECORD SIZE
5105	031340	002005			BGE	120#	;IF RECORD SIZE IS TOO BIG QUIT
5106	031342	020327	177776		CMP	R3,#65534.	;END OF SUBTEST MAX RECORD SIZE
5107	031346	001402			BEQ	120#	;BR, IF COMPLETED
5108	031350	000137	030722		JMP	65#	;DO MORE RECORDS



TEST 3: BASIC WRITE DATA

```

5109 031354          120#:
5110 031354 004737 035116      JSR     PC,T23RT3           ;RESTORE PACKET
5111 031360 004737 035054      JSR     PC,T23RT2           ;CLEAN UP PACKET
5112 031364 012737 102010 033550  MOV     #102010,T23PK2      ;REWIND (POSITION) COMMAND
5113 031372 012704 033550      MOV     #T23PK2,R4         ;LOAD R4 WITH PACKET ADDRESS
5114 031376 010465 000000      MOV     R4,TSDB(R5)        ;ISSUE REWIND COMMAND
5115 031402 004737 016426      JSR     PC,CHKTSSR         ;WAIT FOR SSR TO SET
5116 031406 103407              BCS     130#               ;BR, IF TSSR IS OK (GOOD)
5117 031410 010001              MOV     R0,R1              ;SAVE TSSR CONTENTS
5118 031412 005237 002212      INC     FATFLG              ;BUMP COUNT
5122 031416              ERRHRD  ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND
                                TRAP     C#ERHRD
                                .WORD   313
                                .WORD   T23RWN
                                .WORD   PKTSSR

5123 031426          130#:
5124 031426          ENDSUB                               ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
                                L10045:
                                TRAP     C#ESUB

5125 031430 104403 023727 002212 000017  CMP     FATFLG,#15        ;IS ERROR COUNT AT 25
5126 031436 103402              BLO     999#               ;BR, IF LESS THAN 25
5127 031440 004737 017272      JSR     PC,CKDROP           ;TRY TO DROP THE UNIT
5128 031444          999#:
5129
5130          ;+
5131          ;
5132          ;TEST 3, SUBTEST 3
5133          ;
5134          ;VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE
5135          ;SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST
5136          ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
5137          ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
5138          ;
5139          ;
5140          ;
5141          ;-
5142          BGNSUB                               ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
                                T3.3:
                                TRAP     C#BSUB

5143 031444 104402 004737 034762      JSR     PC,T23REST         ;SET COMMAND PACKET
5144 031452 004737 035054      JSR     PC,T23RT2         ;SET UP OTHER COMMAND PACKET
5145
5146          ;*****
5147          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5148          ;
5149          ;*****
5150
5151          JSR     PC,SOFINIT       ;DO INITIALIZE ON CONTROLLER
5152 031456 004737 016064      BCS     20#               ;BR IF INIT WAS OK
5153 031462 103407              INC     FATFLG             ;BUMP COUNT
5154 031464 005237 002212      MOV     R0,R1              ;CONTENTS OF TSSR REGISTER
5158 031470 010001              ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
5159 031472          ;
                                TRAP     C#ERDF
                                .WORD   314
                                .WORD   SFIERR
                                .WORD   SFIMSG
5159 031472 104455
5159 031474 000472
5159 031476 003650
5159 031500 012124
    
```

TEST 3: BASIC WRITE DATA

```

5160 031502
5161 031502 013737 002172 033460 20$: MOV UNITN,T23DSW ;SET UP UNIT NUMBER
5162 031510 012704 033440 MOV #T23PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
5163
5164 ;*****
5165 ;
5166 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5167 ;
5168 ;*****
5169
5170 031514 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
5171 031520 103407 BCS 23$ ;BR, IF COMMAND ISSUED OK
5172 031522 005237 002212 INC FATFLG ;BUMP COUNT
5176 031526 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
5177 031530 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
      031530 104456 TRAP C$ERHRD
      031532 000473 .WORD 315
      031534 005054 .WORD WRTMSG
      031536 C12124 .WORD SFIMSG
5178 031540 23$:
5179 031540 012703 000024 MOV #20.,R3 ;STARTING RECORD SIZE
5180 031544 013737 003114 033572 65$: MOV FREE,T23WB ;STARTING WRITE BUFFER ADDRESS
5181
5182 ;*****
5183 ;
5184 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
5185 ;
5186 ;*****
5187
5188 031552 012737 150005 033570 MOV #150005,T23PK3 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
5189 031560 012737 150005 033612 MOV #150005,T23WRT ;SETUP FOR RETRY COMMAND
5190 031566 052737 004000 033612 BIS #4000,T23WRT ;MAKE IT A RETRY
5191 031574 012704 033570 MOV #T23PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
5192 031600 010300 MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
5193 031602 004737 017512 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
5194 031606 010337 033576 MOV R3,T23SZ ;SET UP RECORD SIZE IN PACKET
5195 031612 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
5196 031616 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
5197 031622 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
5198 031626 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
5199 031632 020102 CMP R1,R2 ;ARE THEY EQUAL
5200 031634 001402 BEQ 80$ ;BR, IF OK
5201 031636 004737 035142 JSR PC,T23CHK ;CHECK SPECIAL CONDITION
5202 031642 80$: CKLOOP ;LOOP IF SELECTED
      031642 104406 TRAP C$CLP1
5203 031644 016501 000000 MOV TSBA(R5),R1 ;GET TSBA CONTENTS
5204 031650 012702 033462 MOV #T23BFR,R2 ;SET UP EXPECTED
5205 031654 062702 0J0016 ADD #16,R2 ;SET TO END OF MESSAGE BUFFER
5206 031660 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SET
5207 031664 001402 BEQ 85$ ;BR, IF IT NOT SET
5208 031666 062702 000002 ADD #2,R2 ;BUMP R2 FOR EXTRA DATA
5209 031672 020102 85$: CMP R1,R2 ;ARE THEY EQUAL
5210 031674 001406 BEQ 90$ ;BR, IF TSBA IS CORRECT
5211 031676 005237 002212 INC FATFLG ;BUMP COUNT
5215 031702 ERRHRD ERRNO,T23BA,EXPREC ;TSBA WAS NOT CORRECT AFTER WRITE DATA
      031702 104456 TRAP C$ERHRD
      031704 000474 .WORD 316

```

## TEST 3: BASIC WRITE DATA

```

031706 034605 .WORD T23BA
031710 015564 .WORD EXPREC
5216 031712 90#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
031712 104406 ;ONLY CHECK RAM UNTIL ITS FULL
5217 031714 020327 007376 CMP R3,#7376 ;IT WRAPS AROUND ETC.
5218 031720 002115 BGE 115# ;MAKE SURE PACKET AND DATA ARE CLEAN
5219 031722 004737 035054 JSR PC,T23RT2 ;STARTING RAM ADDRESS
5220 031726 012737 000400 033604 MOV #256.,T23S2 ;STOP INTERNAL TSV05 DIAGNOSTICS
5221 031734 112737 000000 033602 MOVB #0,T23BS0 ;SIZE OF RAM READ
5222 031742 112737 000000 033603 MOVB #0,T23BS1 ;SET R4 WITH PACKET ADDRESS
5223 031750 012704 033550 MOV #T23PK2,R4 ;ISSUE WRITE SUB SYS MEM COMMAND
5224 031754 010465 000000 MOV R4,TSDB(R5) ;CHECK TSSR AND WAIT FOR SSR TO SET
5225 031760 004737 016426 JSR PC,CHKTSSR ;BR, IF NO ERRORS IN TSSR
5226 031764 103407 BCS 92# ;SAVE TSSR
5227 031766 010001 MOV R0,R1 ;BUMP COUNT
5228 031770 005237 002212 INC FATFLG ;TSSR BAD AFTER WRITE SUB SYS MEM
5232 031774 ERRHRD ERRNO,T23WSS,PKTSSR TRAP C#ERHRD
031774 104456 .WORD 317
031776 C00475 .WORD T23WSS
032000 034657 .WORD PKTSSR
032002 012136
5233 032004 92#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
032004 104406 ;MAKE SURE PACKET AND DATA ARE CLEAN
5234 032006 004737 035054 JSR PC,T23RT2 ;STARTING RAM ADDRESS
5235 032012 012737 000400 033604 MOV #256.,T23S2 ;READ RAM COMMAND FOR WRITE SUB SYS M.
5236 032020 112737 000001 033602 MOVB #1,T23BS0 ;SIZE OF RAM READ
5237 032026 112737 000002 033603 MOVB #2,T23BS1 ;SET R4 WITH PACKET ADDRESS
5238 032034 012704 033550 MOV #T23PK2,R4 ;ISSUE WRITE SUB SYS MEM CMD (READ RAM)
5239 032040 010465 000000 MOV R4,TSDB(R5) ;CHECK TSSR AND WAIT FOR SSR TO SET
5240 032044 004737 016426 JSR PC,CHKTSSR ;BR, IF NO ERRORS IN TSSR
5241 032050 103407 BCS 100# ;SAVE TSSR
5242 032052 010001 MOV R0,R1 ;BUMP COUNT
5243 032054 005237 002212 INC FATFLG ;TSSR BAD AFTER WRITE SUB SYS MEM
5247 032060 ERRHRD ERRNO,T23WSS,PKTSSR TRAP C#ERHRD
032060 104456 .WORD 318
032062 000476 .WORD T23WSS
032064 034657 .WORD PKTSSR
032066 012136
5248 032070 100#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
032070 104406 ;CLEAR REGISTERS
5249 032072 005001 CLR R1 ;CLEAR REGISTERS
5250 032074 005002 CLR R2 ;PICK UP BYTE READ FROM RAM
5251 032076 013701 033502 MOV T23BFR+20,R1 ;SET UP EXPECTED
5252 032102 010302 MOV R3,R2 ;SWAP BYTES
5253 032104 000302 SWAB R2 ;IS RAM DATA CORRECT
5254 032106 020102 CMP R1,R2 ;BR, IF OK (EQUAL)
5255 032110 001406 BEQ 110# ;BUMP COUNT
5256 032112 005237 002212 INC FATFLG ;RNC=RAM NOT CORRECT
5260 032116 ERRHRD ERRNO,T23RNC,EXPREC TRAP C#ERHRD
032116 104456 .WORD 319
032118 000477 .WORD T23RNC
032120 034145 .WORD EXPREC
032124 015564
5261 032126 110#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
032126 104406 ;BUMP RAM ADDRESS TO BE CHECKED
5262 032130 005237 033604 INC T23S2 ;BUMP RAM ADDRESS TO BE CHECKED
5263 032134 005237 033604 INC T23S2

```

TEST 3: BASIC WRITE DATA

```

5264 032140 010301          MOV      R3,R1          ;GET SIZE OF RECORD
5265 032142 062701 000400  ADD      #256.,R1      ;FIGURE OUT END RECORD ADDRESS
5266 032146 023701 033604  CMP      T23S2,R1     ;AT END OF RAM CHECK YET
5267 032152 001332          BNE      95#          ;BR, IF MORE TO CHECK
5268 032154 062703 001750 115#:  ADD      #1000.,R3     ;NEXT RECORD SIZE/DATA PATTERN
5269 032160 020337 033600  CMP      R3,T23RSZ    ;IS R3 OVER MAX RECORD SIZE
5270 032164 002005          BGE      120#         ;IF RECORD SIZE IS TOO BIG QUIT
5271 032166 020327 177776  CMP      R3,#65534.   ;END OF SUBTEST MAX RECORD SIZE
5272 032172 001402          BEQ      120#         ;BR, IF COMPLETED
5273 032174 000137 031544  JMP      65#          ;DO MORE RECORDS
5274 032200          120#:
5275 032200 004737 035054          JSR      PC,T23RT2    ;CLEAN UP PACKET
5276 032204 012737 102010 033550  MOV      #102010,T23PK2 ;REWIND (POSITION) COMMAND
5277 032212 012704 033550  MOV      #T23PK2,R4   ;LOAD R4 WITH PACKET ADDRESS
5278 032216 010465 000000  MOV      R4,TSD8(R5)  ;ISSUE REWIND COMMAND
5279 032222 004737 016426  JSR      PC,CHKTSSR   ;WAIT FOR SSR TO SET
5280 032226 103407          BCS      130#         ;BR, IF TSSR IS OK (GOOD)
5281 032230 010001          MOV      R0,R1        ;SAVE TSSR CONTENTS
5282 032232 005237 002212  INC      FATFLG       ;BUMP COUNT
5286 032236          ERRHRD   ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND
           032236 104456          TRAP      C#ERHRD
           032240 000500          .WORD    320
           032242 034076          .WORD    T23RWN
           032244 012136          .WORD    PKTSSR
5287 032246          130#:
5288 032246          ENDSUB      ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
           032246 104403          L10046:    TRAP      C#ESUB
5289 032250 023727 002212 000017  CMP      FATFLG,#15.  ;IS ERROR COUNT AT 25
5290 032256 103402          BLO      999#         ;BR, IF LESS THAN 25
5291 032260 004737 017272  JSR      PC,CKDROP    ;TRY TO DROP THE UNIT
5292 032264          999#:
5293
5294
5295
5296 ; TEST 3, SUBTEST 4
5297
5298 ; VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL MODE
5299 ; FIELD OR AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH
5300 ; THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE
5301
5302
5303 :-
5304
5305 032264          BGNSUB      ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
           032264          T3.4:      TRAP      C#BSUB
5306 032266 004737 034762          JSR      PC,T23REST  ;SET COMMAND PACKET
5307 032272 004737 035116  JSR      PC,T23RT3   ;RESTORE PACKET
5308 032276 004737 035054  JSR      PC,T23RT2   ;SET UP OTHER COMMAND PACKET
5309
5310 ;*****
5311
5312 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5313
5314 ;*****
5315

```

TEST 3: BASIC WRITE DATA

```

5316 032302 004737 016064          JSR      PC,SOFINIT             ;DO INITIALIZE ON CONTROLLER
5317 032306 103407                   BCS      20$                   ;BR IF INIT WAS OK
5318 032310 005237 002212          INC      FATFLG               ;BUMP COUNT
5322 032314 010001                   MOV      RO,R1                ;CONTENTS OF TSSR REGISTER
5323 032316                                     ERRDF   ERRNO,SFIERR,SFIMSG   ;FATAL ERROR TSSR WAS NOT OK
                                     TRAP    C#ERDF
                                     .WORD  321
                                     .WORD  SFIERR
                                     .WORD  SFIMSG
032316 104455
032320 000501
032322 003650
032324 012124
5324 032326                                     20$:
5325 032326 013737 002172 033460     MOV      UNITN,T23DSW           ;SET DRIVE NUMBER IN PACKET
5326 032334 012704 033440     MOV      #T23PACKET,R4        ;SUBROUTINE NEEDS PACKET ADDRESS
5327
5328 ;*****
5329 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5330 ;
5331 ;*****
5332
5333
5334 032340 004737 010752          JSR      PC,WRTCHR             ;ISSUE WRITE CHARACTERISTICS
5335 032344 103407                   BCS      23$                   ;BR, IF COMMAND ISSUED OK
5336 032346 005237 002212          INC      FATFLG               ;BUMP COUNT
5340 032352 010001                   MOV      RO,R1                ;SAVE CONTENTS OF TSSR
5341 032354                                     ERRHRD  ERRNO,WRTMSG,SFIMSG   ;WRITE CHARACTERISTIC FAILED
                                     TRAP    C#ERHRD
                                     .WORD  322
                                     .WORD  WRTMSG
                                     .WORD  SFIMSG
032354 104456
032356 000502
032360 005054
032362 012124
5342
5343 ;*****
5344 ;WRITE DATA, ACK, ILLEGAL BITS
5345 ;
5346 ;*****
5347
5348
5349 032364 012737 104405 033570 23$:  MOV      #104405,T23PK3        ;WRITE DATA, ACK, ILLEGAL BITS
5350 032372 013737 003114 033572     MOV      FREE,T23WB           ;SET UP WRITE BUFFER ADDRESS
5351 032400 062737 000001 033572     ADD      #1,T23WB              ;MAKE ADDRESS ODD (ILLEGAL)
5352 032406 012737 000400 033576     MOV      #256,,T23SZ          ;SET UP BUFFER SIZE
5353 032414 012704 033570          MOV      #T23PK3,R4           ;R4 = POINTER TO PACKET
5354 032420 010465 000000          MOV      R4,TSD8(R5)          ;ISSUE COMMAND
5355 032424 004737 016340          JSR      PC,WAITF             ;WAIT FOR SSR TO SET
5356 032430 016501 000002          MOV      TSSR(R5),R1          ;GET TSSR CONTENTS
5357 032434 012702 100206          MOV      #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
5358 032440 020102                   CMP      R1,R2                ;ARE THEY EQUAL
5359 032442 001406                   BEQ      80$                   ;BR, IF OK ESP. FUNCTION REJECT
5360 032444 005237 002212          INC      FATFLG               ;BUMP COUNT
5364 032450                                     ERRHRD  ERRNO,T23TM,PKTSSR   ;TSSR INCORRECT AFTER WRITE COMMAND
                                     TRAP    C#ERHRD
                                     .WORD  323
                                     .WORD  T23TM
                                     .WORD  PKTSSR
032450 104456
032452 000503
032454 034022
032456 012136
5365 032460                                     80$:  CKLOOP                     ;LOOP IF SELECTED
032460 104406                                     TRAP    C#CLP1
5366 032462                                     ENDSUB                        ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
032462 104403                                     L10047:
                                     TRAP    C#ESUB

```



TEST 3: BASIC WRITE DATA

```

032606 005054 .WORD WRTMSG
032610 012124 .WORD SFIMSG

5422
5423 ;*****
5424 ;
5425 ;WRITE DATA, ACK, CVC=1
5426 ;
5427 ;*****
5428
5429 032612 123:
5430 032612 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
5431 032616 001026 BNE 130: ;BR IF SWITCH IS ON
5432 032620 005237 002216 INC EXTFEA ;ONLY ONE TIME
5433 032624 112737 000200 033603 MOVB #200,T23S1 ;WRITE MISCELLANEOUS CONT/READ STATUS
5434 032632 112737 000010 033602 MOVB #10,T23S0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
5435 032640 012704 033550 MOV #T23PK2,R4 ;WRITE SUBSYS MEM PACKET
5436 032644 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
5437 032650 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
5438 032654 103407 BCS 130: ;BR, IF NO ERROR
5439 032656 010001 MOV R0,R1 ;ERROR, SAVE TSSR
5440 032660 005237 002212 INC FATFLG ;BUMP COUNT
5444 032664 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
; TRAP C#ERRRD
; .WORD 326
; .WORD T22SSR
; .WORD PKTSSR
032664 104456
032666 000506
032670 027200
032672 012136
5445 032674 130: CKLOOP ;LOOP IF SELECTED
032674 104406 ; TRAP C#CLP1
5446
5447 032676 012701 160000 MOV #160000,R1 ;NXM LOW ADDRESS START
5448 032702 012702 177776 MOV #177776,R2 ;LIMIT CHECK FOR NXM (HIGHEST)
5449 032706 004737 016466 JSR PC,NXM ;LOOK FOR NXM ADDRESS
5450 032712 103045 BCC 80: ;BR, IF NON FOUND
5451 032714 010137 003130 MOV R1,NXMLO ;SET ADDRESS UP FOR TEST
5452
5453
5454 032720 005037 033574 CLR T23WB+2 ;CLEAR OUT THE HIGH BITS AREA
5455 032724 24:
5456 032724 012737 140005 033570 MOV #140005,T23PK3 ;WRITE DATA, ACK, CVC=1
5457 032732 013737 003130 033572 MOV NXMLO,T23WB ;SET UP WRITE BUFFER ADDRESS
5458 032740 012737 000100 033576 MOV #64,T23SZ ;SET UP BUFFER SIZE
5459 032746 012704 033570 MOV #T23PK3,R4 ;R4 = POINTER TO PACKET
5460 032752 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
5461 032756 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
5462 032762 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
5463 032766 012702 104210 MOV #SC!NXM!SSR!BIT3,R2 ;SET UP EXPECTED
5464 032772 020102 CMP R1,R2 ;ARE THEY EQUAL
5465 032774 001414 BEQ 80: ;BR, IF OK ESP. FUNCTION REJECT
5466 032776 005237 033574 INC T23WB+2 ;BUMP TO NEXT ADDRESS BIT
5467 033002 023727 033574 000004 CMP T23WB+2,#4 ;CHECK TO SEE IF OVERFLOW INTO 19 BIT
5468 033010 001345 BNE 24: ;BR, IF BITS 17 AND 18
5469 033012 005237 002212 25: INC FATFLG ;BUMP COUNT
5473 033016 ERRHRD ERRNO,T23TH,PKTSSR ;TSSR INCORRECT AFTER WRITE COMMAND
; TRAP C#ERRRD
; .WORD 327
; .WORD T23TH
; .WORD PKTSSR
033016 104456
033020 000507
033022 034022
033024 012136

```





TEST 3: BASIC WRITE DATA

```

5525 033134 004737 010752      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
5526 033140 103407              BCS      23$           ;BR, IF COMMAND ISSUED OK
5527 033142 005237 002212      INC      FATFLG        ;BUMP COUNT
5531 033146 010001              MOV      R0,R1         ;SAVE CONTENTS OF TSSR
5532 033150              ERRHRD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP      C$ERHRD
                                .WORD    329
                                .WORD    WRTMSG
                                .WORD    SFIMSG
5533
5534 ;*****
5535 ;
5536 ;WRITE DATA, ACK,CVC=1
5537 ;
5538 ;*****
5539
5540 033160      23$:
5541 033160      MOV      #160000,R1    ;NXM LOW ADDRESS START
5542 033164      MOV      #177776,R2    ;LIMIT CHECK FOR NXM (HIGHEST)
5543 033170      JSR      PC, NXM      ;LOOK FOR NXM ADDRESS
5544 033174      BCC      80$           ;BR, IF NON FOUND
5545 033176      MOV      R1,NXMLO    ;SET ADDRESS UP FOR TEST
5546 033202      MOV      #0,T23WB+2    ;SET TO 16 BIT ADDRESS
5547 033210
5548 033210      MOV      #140005,T23PK3    ;WRITE DATA, ACK,CVC=1
5549 033216      MOV      NXMLO,R1    ;HIGHEST MEMORY ADDRESS LOW BITS
5550 033222      SUB      #500,R1    ;SET ADDRESS A LITTLE LOWER
5551 033226      MOV      R1,T23WB    ;LOAD INTO THE PACKET
5552 033232      MOV      #0,T23SZ    ;SET UP BUFFER SIZE (64K BYTES)
5553 033240      MOV      #T23PK3,R4    ;R4 = POINTER TO PACKET
5554 033244      MOV      R4,TSD8(R5)    ;ISSUE COMMAND
5555 033250      JSR      PC, WAITF    ;WAIT FOR SSR TO SET
5556 033254      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
5557 033260      MOV      #SC!NXM!SSR!BIT3,R2    ;SET UP EXPECTED
5558 033264      CMP      R1,R2        ;ARE THEY EQUAL
5559 033266      BEQ      80$           ;BR, IF OK ESP. FUNCTION REJECT
5560 033270      INC      T23WB+2    ;BUMP TO NEXT ADDRESS RANGE
5561 033274      CMP      T23WB+2,#4    ;CHECK TO SEE IF WE WENT TO HIGH
5562 033302      BNE      24$           ;BR, IF NO OVER FLOW
5563 033304      INC      FATFLG        ;BUMP COUNT
5567 033310      ERRHRD   ERRNO,T23TM,PKTSSR ;TSSR INCORRECT AFTER WRITE COMMAND
                                TRAP      C$ERHRD
                                .WORD    330
                                .WORD    T23TM
                                .WORD    PKTSSR
                                TRAP      C$CLP1
5568 033310      80$: CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    330
                                .WORD    T23TM
                                .WORD    PKTSSR
                                TRAP      C$CLP1
5569 033320      MOV      104406
5569 033322      JSR      PC,T23RT2    ;CLEAN UP PACKET
5570 033326      JSR      PC,T23RT3    ;RESTORE PACKET
5571 033332      MOV      #102010,T23PK2    ;REWIND (POSITION) COMMAND
5572 033340      MOV      #T23PK2,R4    ;LOAD R4 WITH PACKET ADDRESS
5573 033344      MOV      R4,TSD8(R5)    ;ISSUE REWIND COMMAND
5574 033350      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
5575 033354      BCS      130$         ;BR, IF TSSR IS OK (GOOD)
5576 033356      MOV      R0,R1         ;SAVE TSSR CONTENTS
5577 033360      INC      FATFLG        ;BUMP COUNT
5581 033364      ERRHRD   ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND

```

TEST 3: BASIC WRITE DATA

```

033364 104456
033366 000513
033370 034076
033372 012136
5582 033374
5583 033374
033374
033374 104403
5584 033376 023727 002212 000017
5585 033404 103402
5586 033406 004737 017272
5587 033412
5588 033412 004737 016546
5589 033416 103002
5590 033420 000137 030222
5591 033424
5592 033424
033424 104432
033426 C01702
5593
5594
5595
5596
5598 033430
5600 033440
5601 033440 100004
5602 033442 033450
5603 033444 000000
5604 033446 000010
5605 033450
5606 033450 033462
5607 033452 000000
5608 033454 000012
5609 033456 000000
5610 033460 000000
5611 033462
5612
5613
5614
5616 033544
5618 033550
5619 033550 100006
5620 033552 033602
5621 033554 000000
5622 033556 000006
5623
5625 033560
5627 033570
5628 033570 100005
5629 033572 000000
5630 033574 000000
5631 033576 000000
5632
5633
5634 033600 000000
5635
5636

1304:
ENDSUB

9994:
CMP FATFLG,#15.
BLO 9994
JSR PC,CKDROP

9994:
JSR PC,TSTLOOP
BCC 1634
JMP T23LTOP

1634:
EXIT TST

;***** END SUBTEST *****
L10051:
TRAP C#ERMRD
WORD 331
WORD T23RWN
WORD PKTSSR

;IS ERROR COUNT AT 25
;BR, IF LESS THAN 25
;TRY TO DROP THE UNIT

;DO WE NEED TO ITERATE TEST
;BR, IF NO LOOP REQUIRED
;EXECUTE AGAIN

;ALL DONE THIS TEST
TRAP C#EXIT
WORD L10043-

;*
;LOCAL STORAGE FOR THIS TEST
;
;
T23PACKET: .BLKB 10-<.-TSV2E7>
;COMMAND PACKET FOR TEST
;WRITE CHARACTERISTICS COMMAND, WITH ACK
;ADDRESS OF CHARACTERISTICS BLOCK
;
;
T23DATA: .WORD 100004
;STARTING VALUE OF BLOCK SIZE
;CHARACTERISTICS DATA BLOCK
;ADDRESS OF MESSAGE BUFFER
;
;
;WORD T23BFR
;LENGTH OF MESSAGE BUFFER
;
;WORD 0
;
;WORD 10.
;SELECT DRIVE 0
;MESSAGE BUFFER
;
T23DSW: .WORD 0
T23BFR: .BLKW 25.

;WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
;
T23PK2: .BLKB 10-<.-TSV2E7>
;WRITE SUB SYS MEM COMMAND, AND ACK
;ADDRESS OF SELECT BLOCK DATA
;
;
;WORD 100006
;SIZE OF DATA PACKET
;
;WORD T23BF2
;
;WORD 0
;
;WORD 6.

T23PK3: .BLKB 10-<.-TSV2E7>
;WRITE COMMAND, AND ACK
;ADDRESS OF WRITE BUFFER
;
;
T23WB: .WORD 0
;
;WORD 0
;SIZE OF BUFFER (EXTENT)
;
;WORD 0
;EVEN

;
T23RSZ: .WORD 0
;LARGEST TAPE RECORD IN BYTES
;
;

```

TEST 3: BASIC WRITE DATA

```

5637 033602
5638 033602      010
5639 033603      200
5640 033604 000000
5641 033606 000000
5642
5643
5644 033610 000000
5645 033612 000000
5646
5647
5648
5649
5650 033614 100005
5651 033616 100405
5652 033620 102005
5653 033622 177777
5654
5655
5656
5657
5658
5659 033624      127      122      111
5660 033657      105      117      124
5661 033744      127      122      111
5662 034022      124      123      123
5663 034076      122      145      167
5664 034145      122      101      115
5665 034220      124      123      123
5666 034266      104      162      151
5667 034341      124      123      123
5668 034430      124      123      123
5669 034532      103      126      103
5670 034605      124      123      102
5671 034657      127      122      111
5672 034746      102      141      163
5673
5674
5675
5676
5677
5678
5679
5680
5681 034762
5682 034762
5683 034766 012701 033440
5684 034772 012721 100004
5685 034776 012721 033450
5686 035002 005021
5687 035004 012721 000012
5688 035010 012721 033462
5689 035014 005021
5690 035016 012721 000024
5691 035022 005021
5692 035024 012711 000000
5693 035030 012702 000030
    
```

```

T23BF2:
T23BS0: .BYTE 10 ;BSEL0 AREA
T23BS1: .BYTE 200 ;BSEL1 AREA
T23S2: .WORD 0 ;SEL 2 AREA
T23S3: .WORD 0 ;DATA AREA
;
;
T23TMP: .WORD 0 ;TEMPORARY REGISTER
T23WRT: .WORD 0 ;RETRY COMMAND
;
.EVEN
;TAPE MOTION PACKET COMMAND VALUES
T23WD: .WORD 100005 ;WRITE DATA (NEXT)
T23WDR: .WORD 100405 ;WRITE DATA RETRY
T23CON: .WORD 102005 ;WRITE CONTINUOUS
        .WORD 177777 ;END OF DATA
;
;*
;LOCAL TEXT MESSAGES FOR TEST
;-
T23SSR: .ASCIZ 'WRITE Command Not Accepted'
T23ET: .ASCIZ 'EOT Not Found In 12000 4k Writes. (Use Shorter Tape)'
T23EOT: .ASCIZ 'WRITE DATA OVER EOT GAVE NO TAPE STATUS ALERT'
T23TM: .ASCIZ 'TSSR Not Correct After WRITE Command Reject'
T23RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
T23RNC: .ASCIZ 'RAM Error. Correct Data Pattern Not In Ram'
T23AM3: .ASCIZ 'TSSR Init. Failed After WRITE Command'
T23OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
T23WDD: .ASCIZ 'TSSR Not Correct After WRITE DATA Command. SWB Bit Set'
T23WDC: .ASCIZ 'TSSR Not Correct After WRITE DATA Command. Check For Tape Offline'
T23VCK: .ASCIZ 'CVC Set. Didn't Reset VCK In Message Buffer'
T23BA: .ASCIZ 'TSBA Not Correct After WRITE DATA Command'
T23WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
TST23ID: .ASCIZ 'Basic Write'
        .EVEN
;
;
;ROUTINE TO RESTORE COMMAND PACKET TO START UP (DEFAULT) VALUES
;WRITE SUBSYSTEM MEMORY COMMAND
;-
T23REST:
        SAVREG ;SAVE THE REGISTERS
        MOV #T23PACKET,R1 ;START OF THE PACKET
        MOV #100004,(R1); ;WRITE SUBSYSTEM MEM. WITH ACK
        MOV #T23DATA,(R1); ;ADDRESS OF CHARAISTICS DATA BLOCK
        CLR (R1); ;EXTENDED ADDRESS
        MOV #10,(R1); ;SIZE OF DATA BLOCK IN BYTES
        MOV #T23BFR,(R1); ;ADDRESS OF MESSAGE BUFFER
        CLR (R1);
        MOV #20,(R1); ;LENGTH OF MESSAGE BUFFER
        CLR (R1);
        MOV #0,(R1) ;SELECT DRIVE ZERO
        MOV #24,,R2 ;NUMBER OF LOCATIONS TO BE CLEARED
    
```

## TEST 3: BASIC WRITE DATA

```

5694 035034 012762 177777 033462 64:  MOV    #177777,T23BFR(R2)    ; ALL ONES TO MESSAGE BUFFER
5695 035042 005742                TST    -(R2)                ; BUMP DOWN TO NEXT LOCATION
5696 035044 020227 000000        CMP    R2,#0                ; R2 AT ZERO YET
5697 035050 001371                BNE   64:                   ; KEEP GOING UNTIL DONE
5698 035052 000207                RTS    PC                    ; RETURN
5699
5700
5701 035054                T23RT2:
5702 035054                SAVREG                       ; SAVE THE REGISTERS
5703 035060 012701 033550        MOV    #T23PK2,R1           ; START OF THE PACKET
5704 035064 012721 100006        MOV    #100006,(R1)+        ; WRITE SUBSYSTEM MEM. WITH ACK
5705 035070 012721 033602        MOV    #T23BF2,(R1)+        ; ADDRESS OF DATA BLOCK
5706 035074 005021                CLR    (R1)+                ; EXTENDED ADDRESS
5707 035076 012721 000006        MOV    #6,(R1)+            ; SIZE OF DATA BLOCK IN BYTES
5708 035102 012701 033602        MOV    #T23BF2,R1           ; POINT TO DATA SEL AREA
5709 035106 005021                CLR    (R1)+
5710 035110 005021                CLR    (R1)+
5711 035112 005011                CLR    (R1)+
5712 035114 000207                RTS    PC                    ; RETURN
5713 035116
5714 035116                T23RT3:
5715 035122 012701 033570        SAVREG                       ; SAVE THE REGISTERS
5716 035126 012721 100005        MOV    #T23PK3,R1           ; START OF THE PACKET
5717 035132 005021                MOV    #100005,(R1)+        ; WRITE TAPE. WITH ACK
5718 035134 005021                CLR    (R1)+                ; ADDRESS OF DATA BLOCK
5719 035136 005011                CLR    (R1)+                ; EXTENDED ADDRESS
5720 035140 000207                CLR    (R1)+                ; SIZE OF DATA BLOCK
5721                RTS    PC                    ; RETURN
5722
5723                ;
5724                ; ROUTINE TO RETRY WRITE DATA IN CASE OF BAD TAPE FOR TEST
5725                ; 3.SUBTEST 2 & 3
5726                ; INPUTS:          R1=TSSR
5727                ; SUBROUTINE SETS UP T23WRT FOR RETRY
5728                ;
5729                ;
5730 035142                T23CHK:
5731 035142                SAVREG                       ; SAVE THE REGISTERS
5732 035146 005037 033610        CLR    T23TMP               ; CLEAR LOCAL REGISTER
5733 035152 032701 100000        BIT    #SC,R1               ; IS SC SET IN TSSR?
5734 035156 001452                BEQ   FATAL                  ; NO, YOU GOT PROBLEMS!
5735 035160 013702 033472        MOV    T23BFR+10,R2         ; YES, GET XSTAT1
5736 035164 032702 000002        BIT    #X1.UNC,R2           ; IS UNC SET IN XSTAT1?
5737 035170 001401                BEQ   1$                     ; NO, CHECK COR
5738 035172 000405                BR    RETRY                  ; YES, DO WRITE DATA RETRY
5739 035174 032702 020000        1$: BIT    #X1.COR,R2         ; IS COR SET IN XSTAT1 THEN?
5740 035200 001002                BNE   RETRY                  ; YES SO RETRY
5741 035202 000440                BR    FATAL                  ; NO, YOU GOT PROBLEMS
5742 035204 000207                EXIT: RTS    PC              ; RETURN
5743
5744 035206                RETRY:
5745 035206 012703 000024        2$: MOV    #20,R3              ; STARTING RECORD SIZE
5746 035212 013737 003114        MOV    FREE,T23WB           ; STARTING WRITE BUFFER ADDRESS
5747 035220 012737 033612 033572  MOV    #T23WRT,T23PK3       ; WRITE DATA RETRY COMMAND SETUP BY SUBROUTINE
5748 035226 012704 033570        MOV    #T23PK3,R4           ; SET UP R4 WITH PACKET ADDRESS
5749 035232 010300                MOV    R3,R0                ; SET PATTERN IN CORRECT REGISTER
5750 035234 004737 017512        JSR   PC,FILLMEM            ; FILL MEMORY WITH RECORD SIZE

```

## TEST 3: BASIC WRITE DATA

```

5751 035240 010337 033576      MOV      R3,T23SZ      ;SET UP RECORD SIZE IN PACKET
5752 035244 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
5753 035250 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
5754 035254 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
5755 035260 012702 000200      MOV      #SSR,R2     ;SET UP EXPECTED
5756 035264 020102      CMP      R1,R2       ;ARE THEY EQUAL
5757 035266 001746      BEQ      EXIT        ;BR, IF OK
5758 035270 005237 033610      INC      T23TMP      ;TRY FIVE TIMES THEN EXIT
5759 035274 022737 000005 033610  CMP      #5,T23TMP    ;DONE FIVE YET?
5760 035302 001341      BNE      2#         ;NO GO AGAIN
5761 035304 005237 002212      FATAL: INC      FATFLG ;BUMP COUNT
5765 035310 013702 033462      MOV      T23BFR,R2   ;LOW ORDER MSGBUF
5766 035314      ERRHRD  ERRNO,SCHERR,PKTMES ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    332
                                .WORD    SCHERR
                                .WORD    PKTMES
5767 035324 004737 017272      JSR      PC,CKDROP   ;DROP THE UNIT
5768 035330      ENDTST
                                L10043:
                                TRAP      C$ETST
035330 104401

```

```

5769
5770
5771
5772
5773
5774
5775
5776
5777
5778
5779
5780
5781
5782
5783
5784
5785
5786
5787
5788
5789
5790

```

## .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
;
;
; -

```

## BGNTST

```

5791 035332 012737 006356 002170      MOV      #EPRT1,EPRTSW ;SET UP PRIMARY ERROR MESSAGE
5792 035340 005037 003124      CLR      KTENABLE    ;TURN OFF KT11
5793 035344 004737 017364      JSR      PC,KTOFF    ;TURN KT11 OFF
5798 035350 012700 047342      MOV      #TST24ID,R0 ;ASCII MESSAGE TO IDENTIFY TEST
5799 035354 004737 016600      JSR      PC,TSTSETUP ;DO INITIAL TEST SETUP
5800 035360 004737 022206      JSR      PC,MEMCK    ;CHECK FOR MEMORY
5801 035364 005037 003126      CLR      NXMFLG      ;SET FLAG
5802 035370 012737 000005 002206  MOV      #5,LOOPCNT  ;PERFORM 5 ITERATIONS

```

```

5803
5804
5805
5806
5807
;+
;
; TEST 4. SUBTEST 1
;
;

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

5854 035522 103407          BCS      24#          ;BR, IF COMMAND ISSUED OK
5855 035524 005237 00212  INC      FATFLG      ;BUMP COUNT
5859 035530 010001          MOV      R0,R1       ;SAVE CONTENTS OF TSSR
5860 035532          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTISC FAILED
          035532 104456          TRAP     C#ERHRD
          035534 000622          .WORD   402
          035536 005054          .WORD   WRTMSG
          035540 012124          .WORD   SFIMSG
5861 035542 005737 002216 24# :  TST      EXTFEA      ;CHECK FOR EXTENDED FEATURES SW SWITCH
5862 035546 001044          BNE     50#          ;BR IF SWITCH IS ON
5863
5864 035550 112737 000200 045061  MOVB    #200,T24BS1  ;WRITE MISCELLANEOUS CONT/READ STATUS
5865 035556 112737 000010 045060  MOVB    #10,T24BS0   ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
5866 035564 012704 045030  MOV     #T24PK2,R4   ;WRITE SUBSYS MEM PACKET
5867 035570 010465 000000  MOV     R4,TSDB(R5)  ;ISSUE COMMAND
5868 035574 004737 016426  JSR     PC,CHKTSSR   ;WAIT FOR SSR
5869 035600 103407          BCS     30#          ;BR, IF NO ERROR
5870 035602 010001          MOV     R0,R1       ;ERROR, SAVE TSSR
5871 035604 005237 002212  INC     FATFLG      ;BUMP COUNT
5875 035610          ERRHRD  ERRNO,T24SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
          035610 104456          TRAP     C#ERHRD
          035612 000623          .WORD   403
          035614 045617          .WORD   T24SSR
          035616 012136          .WORD   PKTSSR
5876 035620          30# :  CKLOOP          ;LOOP IF SELECTED
          035620 104406          TRAP     C#CLP1
5877 035622 012737 000007 044740  MOV     #7,T24DSW    ;SET DRIVE NUMBER IN PACKET
5878 035630 012704 044720  MOV     #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
5879
5880 ;*****
5881 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5882 ;
5883 ;*****
5884
5885
5886 035634 004737 010752          JSR     PC,WRTCHR    ;ISSUE WRITE CHARACTERISTICS
5887 035640 103407          BCS     50#          ;BR, IF COMMAND ISSUED OK
5888 035642 005237 002212  INC     FATFLG      ;BUMP COUNT
5892 035646 010001          MOV     R0,R1       ;SAVE CONTENTS OF TSSR
5893 035650          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTISC FAILED
          035650 104456          TRAP     C#ERHRD
          035652 000624          .WORD   404
          035654 005054          .WORD   WRTMSG
          035656 012124          .WORD   SFIMSG
5894 035660          50# :  CKLOOP          ;SCOPE LOOP
          035660 104406          TRAP     C#CLP1
5895 035662 016501 000002  MOV     TSSR(R5),R1  ;GET TSSR CONTENTS
5896 035666 032701 000100  BIT     #0FL,R1     ;CHECK FOR THE OFFLINE BIT SET
5897 035672 001006          BNE     60#          ;BR, IF OFFLINE (GOOD)
5898 035674 005237 002212  INC     FATFLG      ;BUMP COUNT
5902 035700          ERRDF  ERRNO,T24OFL,SFIMSG ;OFF LINE SHOULD HAVE BEEN SET (BAD)
          035700 104455          TRAP     C#ERDF
          035702 000625          .WORD   405
          035704 046375          .WORD   T24OFL
          035706 012124          .WORD   SFIMSG
5903 035710          60# :  CKLOOP          ;LOOP IF SELECTED
          035710 104406          TRAP     C#CLP1

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

5904 035712 012703 045066           MOV     #T24RN,R3           ;POINTER FOR COMMANDS
5905
5906 ;*****
5907 ;
5908 ;TAPE READ COMMAND IN PLACE
5909 ;
5910 ;*****
5911
5912 035716 011337 045050 65$:  MOV     (R3),T24PK3       ;TAPE READ COMMAND IN PLACE
5913 035722 012704 045050         MOV     #T24PK3,R4       ;R4 = POINTER TO PACKET
5914 035726 010465 000000         MOV     R4,TSDB(R5)     ;ISSUE COMMAND
5915 035732 004737 016340         SR     PC,WAITF         ;WAIT FOR SSR TO SET
5916 035736 016501 000002         MOV     TSSR(R5),R1     ;GET TSSR CONTENTS
5917 035742 012702 100306         MOV     #SSR!SC!OFL!BIT1!BIT2,R2 ;SET UP EXPECTED
5918 035746 020102                 CMP     R1,R2           ;ARE THEY EQUAL
5919 035750 001406                 BEQ     80$             ;BR, IF OK ESP. FUNCTION REJECT
5920 035752 005237 002212         INC     FATFLG         ;BUMP COUNT
5924 035756                 ERRHRD  ERRNO,T24TM,PKTSSR ;TSSR INCORRECT AFTER TAPE MOTION CMD
                    TRAP   C#ERHRD
                    .WORD  406
                    .WORD  T24TM
                    .WORD  PKTSSR
                    035756 104456
                    035760 000626
                    035762 046133
                    035764 012136
5925 035766                 80$:  CKLOOP           ;LOOP IF SELECTED
                    TRAP   C#CLP1
                    035766 104406
5926 035770 005723                 TST     (R3)+           ;BUMP TO NEXT COMMAND
5927 035772 022713 177777         CMP     #177777,(R3)   ;END OF THE COMMANDS YET
5928 035776 001401                 BEQ     90$             ;BR, IF DONE
5929 036000 000746                 BR     65$             ;MORE COMMAND(S) TO GO
5930 036002
5931 036002                 90$:  ENDSUB           ;>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
                    L10053:
                    TRAP   C#ESUB
                    036002 104403
5932 036004 023727 002212 000017        CMP     FATFLG,#15     ;IS ERROR COUNT AT 25
5933 036012 103402                 BLO     999$           ;BR, IF LESS THAN 25
5934 036014 004737 017272         JSR     PC,CKDROP      ;TRY TO DROP THE UNIT
5935 036020                 999$:
5936
5937 ;
5938 ;
5939 ;TEST 4, SUBTEST 2
5940 ;
5941 ;VERIFIES THAT READ FORWARD COMMANDS WITH SWB=0
5942 ;OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN
5943 ;WRITTEN WITH A SERIES OF TEST RECORDS VARIOUS IN
5944 ;LENGTH AND DATA CONTENT. THE TAPE IS THEN REWOUND
5945 ;AGAIN AND THE RECORDS READ SEQUENTIALLY AND RESULTS
5946 ; (STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON
5947 ; EACH READ FORWARD COMMAND IS SET TO THE LENGTH OF THE
5948 ; EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD
5949 ; OCCUR.
5950 ;
5951 ;
5952 ;
5953 ;-
5954 036020                 BGNSUB           ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
                    036020
                    036020 104402                 T4.2:
                    TRAP   C#BSUB

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

5955 036022 004737 047544          JSR    PC,T24RT3          ;SET UP OTHER COMMAND PACKET
5956 036026 004737 047410          JSR    PC,T24REST        ;SET COMMAND PACKET
5957 036032 004737 047502          JSR    PC,T24RT2        ;SET UP OTHER COMMAND PACKET
5958
5959 ;*****
5960 ;
5961 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
5962 ;
5963 ;*****
5964
5965 036036 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
5966 036042 103407                  BCS    20$              ;BR IF INIT WAS OK
5967 036044 005237 002212          INC    FATFLG           ;BUMP COUNT
5971 036050 010001                  MOV    R0,R1            ;CONTENTS OF TSSR REGISTER
5972 036052          ERRDF    ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   407
                                .WORD   SFIERR
                                .WORD   SFIMSG
5973 036062          20$:
5974 036062 013737 002172 044740      MOV    UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
5975 036070 012704 044720          MOV    #T24PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
5976
5977 ;*****
5978 ;
5979 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
5980 ;
5981 ;*****
5982
5983 036074 004737 010752          JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
5984 036100 103407                  BCS    20$              ;BR, IF COMMAND ISSUED OK
5985 036102 005237 002212          INC    FATFLG           ;BUMP COUNT
5989 036106 010001                  MOV    R0,R1            ;SAVE CONTENTS OF TSSR
5990 036110          ERRHRD   ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C$ERHRD
                                .WORD   408
                                .WORD   WRTMSG
                                .WORD   SFIMSG
5991 036120          24$:   CKLOOP          ;LOOP IF SELECTED
5991 036120 104406          TRAP    C$CLP1
5992
5993 ;*****
5994 ;
5995 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5996 ;
5997 ;*****
5998
5999 036122 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
6000 036126 103407                  BCS    30$              ;BR, IF NO PROBLEM
6001 036130 010001                  MOV    R0,R1            ;SAVE TSSR
6002 036132 005237 002212          INC    FATFLG           ;BUMP COUNT
6006 036136          ERRHRD   ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   409
                                .WORD   T24RWN
                                .WORD   PKTSSR
6007 036146          30$:   CKLOOP          ;LOOP IF SELECTED

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

036146 104406                                TRAP    C#CLP1
6008
6009 ;*****
6010 ;
6011 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6012 ;
6013 ;*****
6014
6015 036150 013701 044750                    MOV     T24BFR+6,R1      ;PICK UP XSTO
6016 036154 010102                          MOV     R1,R2           ;SET UP EXPECTED
6017 036156 052702 000002                    BIS     #BIT1,R2       ;SET BOT BIT IN EXPECTED
6018 036162 020102                          CMP     R1,R2           ;DOES EXP = REC'D
6019 036164 001406                          BEQ     40#             ;BR, IF EQUAL (OK)
6020 036166 005237 002212                    INC     FATFLG          ;BUMP COUNT
6024 036172                          ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C#ERHRD
                                .WORD  410
                                .WORD  T24BOT
                                .WORD  EXPREC
                                TRAP    C#CLP1
6025 036202 104406 000400                    40#:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
6026 036204 012703 000400                    MOV     #256.,R3       ;RECORD SIZE
6027 036210 013737 003114 045052            MOV     FREE,T24RB     ;STARTING WRITE BUFFER ADDRESS
6028
6029 ;*****
6030 ;
6031 ;WRITE DATA,CVC=1,ACK COMMAND
6032 ;
6033 ;*****
6034
6035 036216 012737 140005 045050            MOV     #140005,T24PK3 ;WRITE DATA,CVC=1,ACK COMMAND
6036 036224 012704 045050                    MOV     #T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
6037 036230 65#:
6038 036230 010300                          MOV     R3,R0          ;SET PATTERN IN CORRECT REGISTER
6039 036232 004737 017512                    JSR     PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
6040 036236 010337 045056                    MOV     R3,T24SZ       ;SET UP RECORD SIZE IN PACKET
6041 036242 010465 000000                    MOV     R4,TSDB(R5)   ;ISSUE COMMAND
6042 036246 004737 016340                    JSR     PC,WAITF       ;WAIT FOR SSR TO SET
6043 036252 016501 000002                    MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
6044 036256 012702 000200                    MOV     #SSR,R2       ;SET UP EXPECTED
6045 036262 020102                          CMP     R1,R2          ;ARE THEY EQUAL
6046 036264 001406                          BEQ     75#            ;BR, IF OK
6047 036266 005237 002212                    INC     FATFLG          ;BUMP COUNT
6051 036272                          ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP    C#ERHRD
                                .WORD  411
                                .WORD  WRERR
                                .WORD  PKTSSR
                                TRAP    C#CLP1
6052 036302 104406 000414                    75#:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
6053 036304 005723                          TST     (R3)+          ;BUMP RECORD SIZE
6054 036306 022703 000414                    CMP     #268.,R3      ;END OF RECORD YET
6055 036312 001346                          BNE     65#           ;BR, IF MORE RECORDS TO WRITE
6056 036314 104406 80#:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C#CLP1
6057 036316 120#:
6058

```

## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6059 ;*****
6060 ;
6061 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6062 ;
6063 ;*****
6064
6065 036316 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6066 036322 004737 016426 JSR PC,CHKTSSR ;SEE HOW TSSR IS
6067 036326 103407 BCS 130# ;BR, IF NO PROBLEM
6068 036330 010001 MOV R0,R1 ;SAVE TSSR
6069 036332 005237 002212 INC FATFLG ;BUMP COUNT
6073 036336 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
        036336 104456 TRAP C#ERHRD
        036340 000634 .WORD 412
        036342 046206 .WORD T24RWN
        036344 012136 .WORD PKTSSR
6074 036346 104406 130#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
        036346 104406
6075 ;*****
6076 ;
6077 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6078 ;
6079 ;*****
6080
6081
6082 036350 013701 044750 MOV T24BFR+6,R1 ;PICK UP XSTO
6083 036354 010102 MOV R1,R2 ;SET UP EXPECTED
6084 036356 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
6085 036362 020102 CMP R1,R2 ;DOES EXP = REC'D
6086 036364 001406 BEQ 140# ;BR, IF EQUAL (OK)
6087 036366 005237 002212 INC FATFLG ;BUMP COUNT
6091 036372 ERRHRD ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        036372 104456 TRAP C#ERHRD
        036374 000635 .WORD 413
        036376 045723 .WORD T24BOT
        036400 015564 .WORD EXPREC
6092 036402 104406 140#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
        036402 104406
6093 036404 012703 000400 MOV #256.,R3 ;RECORD SIZE
6094 036410 013737 003114 045052 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
6095 ;*****
6096 ;
6097 ;READ DATA,CVC=1,ACK COMMAND
6098 ;
6099 ;*****
6100
6101
6102 036416 012737 140001 045050 165#: MOV #140001,T24PK3 ;READ DATA,CVC=1,ACK COMMAND
6103 036424 012704 045050 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6104 036430 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6105 036434 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6106 036440 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6107 036444 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6108 036450 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6109 036454 020102 CMP R1,R2 ;ARE THEY EQUAL
6110 036456 001406 BEQ 170# ;BR, IF OK
6111 036460 005237 002212 INC FATFLG ;BUMP COUNT

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6160 036612 004737 047502          JSR    PC,T24RT2          ;SET UP OTHER COMMAND PACKET
6161
6162          ;*****
6163          ;
6164          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6165          ;
6166          ;*****
6167
6168 036616 004737 016064          JSR    PC,SOFINIT        ;DO INITIALIZE ON CONTROLLER
6169 036622 103407                BCS    20:              ;BR IF INIT WAS OK
6170 036624 005237 002212          INC    FATFLG           ;BUMP COUNT
6174 036630 010001                MOV    R0,R1            ;CONTENTS OF TSSR REGISTER
6175 036632                ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C:ERDF
                                .WORD   416
                                .WORD   SFIERR
                                .WORD   SFIMSG
6176 036642                104455
6177 036642 C13737 002172 044740 20:  MOV    UNITN,T24DSW      ;SET DRIVE NUMBER IN PACKET
6178 036650 012704 044720          MOV    @T24PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
6179
6180          ;*****
6181          ;
6182          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6183          ;
6184          ;*****
6185
6186 036654 004737 010752          JSR    PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
6187 036660 103407                BCS    24:              ;BR, IF COMMAND ISSUED OK
6188 036662 005237 002212          INC    FATFLG           ;BUMP COUNT
6192 036666 010001                MOV    R0,R1            ;SAVE CONTENTS OF TSSR
6193 036670                ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C:ERHRD
                                .WORD   417
                                .WORD   WRTMSG
                                .WORD   SFIMSG
6194 036700                104406
6194 036700 104406          24:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C:CLP1
6195
6196          ;*****
6197          ;
6198          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6199          ;
6200          ;*****
6201
6202 036702 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
6203 036706 103407                BCS    30:              ;BR, IF NO PROBLEM
6204 036710 010001                MOV    R0,R1            ;SAVE TSSR
6205 036712 005237 002212          INC    FATFLG           ;BUMP COUNT
6209 036716                ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C:ERHRD
                                .WORD   418
                                .WORD   T24RWN
                                .WORD   PKTSSR
6210 036726                104406
6210 036726 104406          30:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C:CLP1
6211

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6212 ;*****
6213 ;
6214 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6215 ;
6216 ;*****
6217
6218 036730 013701 044750          MOV      T24BFR+6,R1      ;PICK UP XSTO
6219 036734 010102          MOV      R1,R2          ;SET UP EXPECTED
6220 036736 052702 000002      BIS      #BIT1,R2        ;SET BOT BIT IN EXPECTED
6221 036742 020102          CMP      R1,R2          ;DOES EXP = REC'D
6222 036744 001406          BEQ     40$             ;BR, IF EQUAL (OK)
6223 036746 005237 002212      INC      FATFLG          ;BUMP COUNT
6227 036752          ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP     C#ERHRD
                                .WORD   419
                                .WORD   T24BOT
                                .WORD   EXPREC
6228 036762          40$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP     C#CLP1
6229 036764 012703 000400      MOV      #256.,R3        ;RECORD SIZE
6230 036770 013737 003114 045052  MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
6231 ;*****
6232 ;
6233 ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
6234 ;
6235 ;*****
6236
6237
6238 036776 012737 150005 045050  MOV      #150005,T24PK3 ;WRITE DATA,ACK,SWB,CVC=1 COMMAND
6239 037004 012704 045050      MOV      #T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
6240 037010          65$:  MOV      R3,R0          ;SET PATTERN IN CORRECT REGISTER
6241 037010 010300          JSR     PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
6242 037012 004737 017512      MOV      R3,T24SZ       ;SET UP RECORD SIZE IN PACKET
6243 037016 010337 045056      MOV      R4,TSD8(R5)    ;ISSUE COMMAND
6244 037022 010465 000000      JSR     PC,WAITF        ;WAIT FOR SSR TO SET
6245 037026 004737 016340      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
6246 037032 016501 000002      MOV      #SSR,R2        ;SET UP EXPECTED
6247 037036 012702 000200      CMP     R1,R2          ;ARE THEY EQUAL
6248 037042 020102          BEQ     75$             ;BR, IF OK
6249 037044 001406          INC     FATFLG          ;BUMP COUNT
6250 037046 005237 002212      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
6254 037052          TRAP     C#ERHRD
                                .WORD   420
                                .WORD   WRERR
                                .WORD   PKTSSR
6255 037062          75$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP     C#CLP1
6256 037064 104406          TST     (R3)+           ;BUMP RECORD SIZE
6257 037066 005723 000414      CMP     #268.,R3        ;END OF RECORD YET
6258 037072 001316          BNE     65$             ;BR, IF MORE RECORDS TO WRITE
6259 037074          80$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP     C#CLP1
6260 037076          120$:
6261 ;*****
6262 ;
6263 ;

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6264 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6265 ;
6266 ;*****
6267 ;
6268 037076 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6269 037102 103407 BCS 130# ;BR, IF NO PROBLEM
6270 037104 010001 MOV R0,R1 ;SAVE TSSR
6271 037106 005237 002212 INC FATFLG ;BUMP COUNT
6275 037112 ERRHRD ERRNO,T24RWN,EXPREC ;REWIND NOT ACCEPTED
        037112 104456 TRAP C#ERHRD
        037114 000645 .WORD 421
        037116 046206 .WORD T24RWN
        037120 015564 .WORD EXPREC
6276 037122 130# CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
        037122 104406
6277 ;*****
6278 ;
6279 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6280 ;
6281 ;*****
6282 ;
6283 ;
6284 037124 013701 044750 MOV T24BFR+6,R1 ;PICK UP XSTO
6285 037130 010102 MOV R1,R2 ;SET UP EXPECTED
6286 037132 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
6287 037136 020102 CMP R1,R2 ;DOES EXP = REC'D
6288 037140 001406 BEQ 140# ;BR, IF EQUAL (OK)
6289 037142 005237 002212 INC FATFLG ;BUMP COUNT
6293 037146 ERRHRD ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
        037146 104456 TRAP C#ERHRD
        037150 000646 .WORD 422
        037152 045723 .WORD T24BOT
        037154 015564 .WORD EXPREC
6294 037156 140# CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
        037156 104406
6295 037160 012703 000400 MOV #256,R3 ;RECORD SIZE
6296 037164 013737 003114 045052 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
6297 ;*****
6298 ;
6299 ;READ DATA,IE,ACK,SWB COMMAND
6300 ;
6301 ;*****
6302 ;
6303 ;
6304 037172 012737 110001 045050 MOV #110001,T24PK3 ;READ DATA,IE,ACK,SWB COMMAND
6305 037200 012704 045050 165# MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6306 037204 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6307 037210 010465 000000 MOV R4,TSD8(R5) ;ISSUE COMMAND
6308 037214 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6309 037220 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6310 037224 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6311 037230 020102 CMP R1,R2 ;ARE THEY EQUAL
6312 037232 001406 BEQ 170# ;BR, IF OK
6313 037234 005237 002212 INC FATFLG ;BUMP COUNT
6317 037240 ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
        037240 104456 TRAP C#ERHRD
        037242 000647 .WORD 423
    
```





TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6365                                     ;
6366                                     ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6367                                     ;
6368                                     ;*****
6369                                     ;
6370 037372 004737 016064                JSR    PC,SOFINIT                ;DO INITIALIZE ON CONTROLLER
6371 037376 103407                        BCS    20$                       ;BR IF INIT WAS OK
6372 037400 005237 002212                INC    FATFLG                    ;BUMP COUNT
6376 037404 010001                        MOV    R0,R1                     ;CONTENTS OF TSSR REGISTER
6377 037406                                ERRDF  ERRNO,SFIERR,SFIMSG        ;FATAL ERROR TSSR WAS NOT OK
                                           TRAP  C$ERDF
                                           .WORD 425
                                           .WORD SFIERR
                                           .WORD SFIMSG
6378 037416                                20$:
6379 037416 013737 002172 044740        MOV    UNITN,T24DSW              ;SET DRIVE NUMBER IN PACKET
6380 037424 012704 044720                MOV    @T24PACKET,R4            ;SUBROUTINE NEEDS PACKET ADDRESS
6381                                     ;
6382                                     ;*****
6383                                     ;
6384                                     ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6385                                     ;
6386                                     ;*****
6387                                     ;
6388 037430 004737 010752                JSR    PC,WRTCHR                 ;ISSUE WRITE CHARACTERISTICS
6389 037434 103407                        BCS    24$                       ;BR, IF COMMAND ISSUED OK
6390 037436 005237 002212                INC    FATFLG                    ;BUMP COUNT
6394 037442 010001                        MOV    R0,R1                     ;SAVE CONTENTS OF TSSR
6395 037444                                ERRHRD ERRNO,WRTMSG,SFIMSG        ;WRITE CHARACTERISTICS FAILED
                                           TRAP  C$ERHRD
                                           .WORD 426
                                           .WORD WRTMSG
                                           .WORD SFIMSG
6396 037454                                24$: CKLOOP                      ;LOOP IF SELECTED
6397 037454 104406                        TRAP  C$CLP1
6398                                     ;
6399                                     ;*****
6400                                     ;
6401                                     ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6402                                     ;
6403                                     ;*****
6404 037456 004737 011104                JSR    PC,REWIND                 ;CALL TAPE REWIND COMMAND
6405 037462 103407                        BCS    30$                       ;BR, IF NO PROBLEM
6406 037464 010001                        MOV    R0,R1                     ;SAVE TSSR
6407 037466 005237 002212                INC    FATFLG                    ;BUMP COUNT
6411 037472                                ERRHRD ERRNO,T24RWN,PKTSSR      ;REWIND NOT ACCEPTED
                                           TRAP  C$ERHRD
                                           .WORD 427
                                           .WORD T24RWN
                                           .WORD PKTSSR
6412 037502                                30$: CKLOOP                      ;LOOP IF SELECTED
6413 037502 104406                        TRAP  C$CLP1
6414                                     ;
6415                                     ;*****
6416                                     ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6417
6418
6419
6420 037504 013701 044750      MOV      T24BFR+6,R1      ;PICK UP XSTO
6421 037510 010102              MOV      R1,R2           ;SET UP EXPECTED
6422 037512 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
6423 037516 020102              CMP      R1,R2           ;DOES EXP = REC'D
6424 037520 001406              BEQ      40$             ;BR, IF EQUAL (OK)
6425 037522 005237 002212      INC      FATFLG          ;BUMP COUNT
6429 037526              ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    428
                                .WORD    T24BOT
                                .WORD    EXPREC
                                TRAP      C$CLP1
6430 037536 40$: CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
6431 037540 012703 001000      MOV      @512,R3         ;RECORD SIZE
6432 037544 013737 003114 045052  MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
6433
6434
6435
6436
6437
6438
6439
6440 037552 012737 140005 045050      MOV      @140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
6441 037560 012704 045050      MOV      @T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
6442 037564
6443 037564 010337 045056      65$: MOV      R3,T24SZ     ;SET UP RECORD SIZE IN PACKET
6444 037570 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
6445 037574 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
6446 037600 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
6447 037604 012702 000200      MOV      @SSR,R2        ;SET UP EXPECTED
6448 037610 020102              CMP      R1,R2          ;ARE THEY EQUAL
6449 037612 001406              BEQ      75$            ;BR, IF OK
6450 037614 005237 002212      INC      FATFLG          ;BUMP COUNT
6454 037620              ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C$ERHRD
                                .WORD    429
                                .WORD    WRTErr
                                .WORD    PKTSSR
                                TRAP      C$CLP1
6455 037630 75$: CKLOOP              ;LOOP IF SELECTED
6456 037632 120$:
6457
6458
6459
6460
6461
6462
6463
6464 037632 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
6465 037636 103407              BCS     130$            ;BR, IF NO PROBLEM
6466 037640 010001              MOV      R0,R1          ;SAVE TSSR
6467 037642 005237 002212      INC      FATFLG          ;BUMP COUNT
6471 037646              ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    429
                                .WORD    T24RWN
                                .WORD    PKTSSR
                                TRAP      C$CLP1

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

037650 000656 .WORD 430
037652 046206 .WORD T24RWN
037654 012136 .WORD PKTSSR
6472 037656 104406 130$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
037656 104406
6473
6474 ;*****
6475 ;
6476 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6477 ;
6478 ;*****
6479
6480 037660 013701 044750 MOV T24BFR+6,R1 ;PICK UP XSTO
6481 037664 010102 MOV R1,R2 ;SET UP EXPECTED
6482 037666 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
6483 037672 020102 CMP R1,R2 ;DOES EXP = REC'D
6484 037674 001406 BEQ 140$ ;BR, IF EQUAL (OK)
6485 037676 005237 002212 INC FATFLG ;BUMP COUNT
6489 037702 ERRHRD ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
037702 104456 TRAP C$ERHRD
037704 000657 .WORD 431
037706 045723 .WORD T24BOT
037710 015564 .WORD EXPREC
6490 037712 104406 140$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
037712 104406
6491 037714 012703 000400 MOV #256.,R3 ;RECORD SIZE
6492 037720 013737 003114 045052 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
6493
6494 ;*****
6495 ;
6496 ;READ DATA,ACK,CVC=1 COMMAND
6497 ;
6498 ;*****
6499
6500 037726 012737 140001 045050 MOV #140001,T24PK3 ;READ DATA,ACK,CVC=1 COMMAND
6501 037734 012704 045050 MOV #T24PK3 R4 ;SET UP R4 WITH PACKET ADDRESS
6502 037740 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6503 037744 010465 000000 MOV R4,TSD8(R5) ;ISSUE COMMAND
6504 037750 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6505 037754 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6506 037760 012702 100204 MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
6507 037764 020102 CMP R1,R2 ;ARE THEY EQUAL
6508 037766 001406 BEQ 170$ ;BR, IF OK
6509 037770 005237 002212 INC FATFLG ;BUMP COUNT
6513 037774 ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
037774 104456 TRAP C$ERHRD
037776 000660 .WORD 432
040000 047254 .WORD T24TRL
040002 012136 .WORD PKTSSR
6514 040004 104406 170$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
040004 104406
6515
6516 ;*****
6517 ;
6518 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6519 ;
6520 ;*****

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

040112 000662 .WORD 434
040114 003650 .WORD SFIERR
040116 012124 .WORD SFIMSG
6575 040120
6576 040120 013737 002172 044740 20$: MOV UNITN,T24DSW ;SET DRIVE NUMBER IN PACKET
6577 040126 012704 044720 MOV #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
6578
6579 ;*****
6580 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6581 ;
6582 ;*****
6583
6584
6585 040132 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
6586 040136 103407 BCS 24$ ;BR. IF COMMAND ISSUED OK
6587 040140 005237 002212 INC FATFLG ;BUMP COUNT
6591 040144 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
6592 040146 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
040146 104456 TRAP C#ERHRD
040150 000663 .WORD 435
040152 005054 .WORD WRTMSG
040154 012124 .WORD SFIMSG
6593 040156 24$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
040156 104406
6594
6595 ;*****
6596 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6597 ;
6598 ;*****
6599
6600
6601 040160 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6602 040164 103407 BCS 30$ ;BR. IF NO PROBLEM
6603 040166 010001 MOV RO,R1 ;SAVE TSSR
6604 040170 005237 002212 INC FATFLG ;BUMP COUNT
6608 040174 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
040174 104456 TRAP C#ERHRD
040176 000664 .WORD 436
040200 046206 .WORD T24RWN
040202 012136 .WORD PKTSSR
6609 040204 30$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
040204 104406
6610 040206 012703 000400 MOV #256.,R3 ;RECORD SIZE
6611 040212 013737 003114 045052 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS
6612
6613 ;*****
6614 ;WRITE DATA,ACK,CVC=1 COMMAND
6615 ;
6616 ;*****
6617
6618
6619 040220 012737 140005 045050 MOV #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
6620 040226 012704 045050 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6621 040232 65$:
6622 040232 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6623 040236 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6624 040242 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET

```

## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6625 040246 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
6626 040252 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED
6627 040256 020102              CMP      R1,R2          ;ARE THEY EQUAL
6628 040260 001406              BEQ      75#            ;BR, IF OK
6629 040262 005237 002212      INC      FATFLG         ;BUMP COUNT
6633 040266              ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERHRD
                                .WORD    437
                                .WORD    WRTErr
                                .WORD    PKTSSR
                                TRAP      C#CLP1
        040266 104456
        040270 000665
        040272 005111
        040274 012136
6634 040276              75#:   CKLOOP          ;LOOP IF SELECTED
        040276 104406
6635 040300              120#:
6636
6637 ;*****
6638 ;
6639 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6640 ;
6641 ;*****
6642
6643 040300 004737 011104      JSR      PC.REWIND      ;CALL TAPE REWIND COMMAND
6644 040304 103407              BCS      130#          ;BR, IF NO PROBLEM
6645 040306 010001              MOV      R0,R1         ;SAVE TSSR
6646 040310 005237 002212      INC      FATFLG         ;BUMP COUNT
6650 040314              ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    438
                                .WORD    T24RWN
                                .WORD    PKTSSR
                                TRAP      C#CLP1
        040314 104456
        040316 000666
        040320 046206
        040322 012136
6651 040324              130#:  CKLOOP          ;LOOP IF SELECTED
        040324 104406
6652 040326 012703 001000      MOV      #512.,R3      ;RECORD SIZE
6653 040332 013737 003114 045052      MOV      FREE,T24RB    ;STARTING READ BUFFER ADDRESS
6654
6655 ;*****
6656 ;
6657 ;READ DATA,ACK,CVC=1 COMMAND
6658 ;
6659 ;*****
6660
6661 040340 012737 140001 045050      MOV      #140001,T24PK3 ;READ DATA,ACK,CVC=1 COMMAND
6662 040346 012704 045050      165#:  MOV      #T24PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
6663 040352 010337 045056      MOV      R3,T24SZ      ;SET UP RECORD SIZE IN PACKET
6664 040356 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE COMMAND
6665 040362 004737 016340      JSR      PC.WAITF      ;WAIT FOR SSR TO SET
6666 040366 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
6667 040372 012702 100204      MOV      #SSR!SC!BIT2,R2 ;SET UP EXPECTED
6668 040376 020102              CMP      R1,R2        ;ARE THEY EQUAL
6669 040400 001406              BEQ      170#         ;BR, IF OK
6670 040402 005237 002212      INC      FATFLG         ;BUMP COUNT
6674 040406              ERRHRD  ERRNO,T24TRL,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERHRD
                                .WORD    439
                                .WORD    T24TRL
                                .WORD    EXPREC
                                TRAP      C#CLP1
        040406 104456
        040410 000667
        040412 047254
        040414 015564
6675 040416              170#:  CKLOOP          ;LOOP IF SELECTED
        040416 104406

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

                                T4.6:
040522                                TRAP          C:DSUB
040522 104402
6728 040524 004737 047544          JSR          PC,T24RT3      ;SET UP OTHER COMMAND PACKET
6729 040530 004737 047410          JSR          PC,T24REST    ;SET COMMAND PACKET
6730 040534 004737 047502          JSR          PC,T24RT2    ;SET UP OTHER COMMAND PACKET
6731
6732          ;*****
6733          ;
6734          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6735          ;
6736          ;*****
6737
6738 040540 004737 016064          JSR          PC,SOFINIT    ;DO INITIALIZE ON CONTROLLER
6739 040544 103407                    BCS          20#          ;BR IF INIT WAS OK
6740 040546 005237 002212          INC          FATFLG       ;BUMP COUNT
6744 040552 010001                    MOV          R0,R1        ;CONTENTS OF TSSR REGISTER
6745 040554                    ERRDF        ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP          C:ERDF
                                .WORD        442
                                .WORD        SFIERR
                                .WORD        SFIMSG
040554 104455
040556 C00672
040560 003650
040562 012124
6746 040564                    20#:
6747 040564 013737 002172 044740    MOV          UNITN,T24DSW  ;SET DRIVE NUMBER IN PACKET
6748 040572 012704 044720    MOV          #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
6749
6750          ;*****
6751          ;
6752          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6753          ;
6754          ;*****
6755
6756 040576 004737 010752          JSR          PC,WRTCHR    ;ISSUE WRITE CHARACTERISTICS
6757 040602 103407                    BCS          24#          ;BR. IF COMMAND ISSUED OK
6758 040604 005237 002212          INC          FATFLG       ;BUMP COUNT
6762 040610 010001                    MOV          R0,R1        ;SAVE CONTENTS OF TSSR
6763 040612                    ERRHRD        ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP          C:ERHRD
                                .WORD        443
                                .WORD        WRTMSG
                                .WORD        SFIMSG
040612 104456
040614 000673
040616 005054
040620 012124
6764 040622                    24#:
040622 104406                    CKLOOP                ;LOOP IF SELECTED
                                TRAP          C:CLP1
6765
6766          ;*****
6767          ;
6768          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6769          ;
6770          ;*****
6771
6772 040624 004737 011104          JSR          PC,REWIND    ;CALL TAPE REWIND COMMAND
6773 040630 103407                    BCS          30#          ;BR. IF NO PROBLEM
6774 040632 010001                    MOV          R0,R1        ;SAVE TSSR
6775 040634 005237 002212          INC          FATFLG       ;BUMP COUNT
6779 040640                    ERRHRD        ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP          C:ERHRD
                                .WORD        444
                                .WORD        T24RWN
040640 104456
040642 000674
040644 046206
    
```



## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

040646 012136
6780 040650 104406 30#: CKLOOP ;LOOP IF SELECTED .WORD PKTSSR
040650 104406 ;TRAP C:CLP1
6781 040652 012703 000400 MOV #256.,R3 ;RECORD SIZE
6782 040656 013737 003114 045052 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS
6783
6784 ;*****
6785 ;WRITE DATA,ACK,CVC-1 COMMAND
6786 ;
6787 ;*****
6788
6789
6790 040664 012737 140005 045050 MOV #140005,T24PK3 ;WRITE DATA,ACK,CVC-1 COMMAND
6791 040672 012704 045050 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6792 040676
6793 040676 010300 65#: MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
6794 040700 004737 017512 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
6795 040704 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6796 040710 C10465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6797 040714 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6798 040720 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6799 040724 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6800 040730 020102 CMP R1,R2 ;ARE THEY EQUAL
6801 040732 001406 BEQ 75# ;BR, IF OK
6802 040734 005237 002212 INC FATFLG ;BUMP COUNT
6806 040740 ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
040740 104456 TRAP C:ERHRD
040742 000675 .WORD 445
040744 005111 .WORD WRERR
040746 012136 .WORD PKTSSR
6807 040750 75#: CKLOOP ;LOOP IF SELECTED .WORD
040750 104406 ;TRAP C:CLP1
6808 040752 005723 TST (R3)^ ;BUMP RECORD SIZE
6809 040754 022703 000414 CMP #268.,R3 ;END OF RECORD YET
6810 040760 001346 BNE 65# ;BR, IF MORE RECORDS TO WRITE
6811 040762 80#: CKLOOP ;LOOP IF SELECTED .WORD
040762 104406 ;TRAP C:CLP1
6812 040764 005743 TST -(R3) ;SET BACK TO 512.
6813 040766 013737 003114 045052 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
6814
6815 ;*****
6816 ;READ REVERSE DATA,ACK COMMAND
6817 ;
6818 ;*****
6819
6820
6821 040774 012737 100401 045050 MOV #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
6822 041002 012704 045050 165#: MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
6823 041006 010337 045056 MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
6824 041012 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
6825 041016 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
6826 041022 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6827 041026 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
6828 041032 020102 CMP R1,R2 ;ARE THEY EQUAL
6829 041034 001406 BEQ 170# ;BR, IF OK
6830 041036 005237 002212 INC FATFLG ;BUMP COUNT
6834 041042 ERRHRD ERRNO,T24WDC,PKTSSR ;TSSR INCORRECT AFTER READ DATA

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

041042 104456
041044 000676
041046 046536
041050 012136
6835 041052 1701: CKLOOP ;LOOP IF SELECTED TRAP C1ERHRD
041052 104406 ;GET BUFFER ADDRESS TRAP C1CLP1
6836 041054 013702 003114 MOV FREE,R2 ;CURRENT RECORD SIZE
6837 041060 010304 MOV R3,R4 ;FIRST LOCATION IN BUFFER
6838 041062 162704 000400 SUB #256.,R4 ;SET POINTER TO FRAME (WORD)
6839 041066 060204 1731: ADD R2,R4 ;CHECK DATA READ (R3=DATA ALSO)
6840 041070 021403 CMP (R4),R3 ;BR, IF ALL IS WELL
6841 041072 001410 BEQ 1801 ;RECD DATA
6842 041074 011401 MOV (R4),R1 ;EXPECTED DATA
6843 041076 010302 MOV R3,R2 ;BUMP COUNT
6844 041100 005237 002212 INC FATFLG ;DATA READ NOT - WRITTEN
6848 041104 ERRHRD ERRNO,T24DTA,EXPREC TRAP C1ERHRD
041104 104456 .WORD 447
041106 000677 .WORD T24DTA
041110 C45770 .WORD EXPREC
041112 015564
6849 041114 1801: CKLOOP ;LOOP IF SELECTED TRAP C1CLP1
041114 104406 ;BUMP TO NEXT LOCATION
6850 041116 005724 TST (R4)+ ;GET RID OF BASE ADDRESS
6851 041120 160204 SUB R2,R4 ;END OF RECORD YET
6852 041122 020403 CMP R4,R3 ;BR, IF NOT AT END OF RECORD
6853 041124 001360 BNE 1731 ;BUMP RECORD SIZE
6854 041126 005743 TST -(R3) ;END OF RECORD YET
6855 041130 022703 000400 CMP #256.,R3 ;BR, IF MORE RECORDS TO WRITE
6856 041134 001322 BNE 1651 ;LOOP IF SELECTED
6857 041136 1901: CKLOOP TRAP C1CLP1
041136 104406 ;>>>>>>>>>> END SUBTEST >>>>>>>>>
6858 041140 ENDSUB L10060: TRAP C1ESUB
041140 104403 ;IS ERROR COUNT AT 25
6859 041142 023727 002212 000017 CMP FATFLG,#15. ;BR, IF LESS THAN 25
6860 041150 103402 BLO 9991 ;TRY TO DROP THE UNIT
6861 041152 004737 017272 JSR PC,CKDROP
6862 041156 9991:
6863 ;
6864 ;
6865 ;
6866 ;TEST 4. SUBTEST 7
6867 ;
6868 ;VERIFIES THAT READ DATA COMMANDS WITH CVC=1 AND THE
6869 ;SWAP BYTES (SMB) BIT SET OPERATES PROPERLY. THE TEST
6870 ;SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2.
6871 ;THE RESULTS, EXCEPT FOR RAM CONTENTS, SHOULD BE THE SAME.
6872 ;
6873 ;
6874 ;
6875 ;
6876 ;- BGNSUB ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
041156 104402 T4.7: TRAP C1SUB
041156 004737 047544 JSR PC,T24RT3 ;SET UP OTHER COMMAND PACKET
6877 041160 004737 047410 JSR PC,T24REST ;SET COMMAND PACKET
6878 041164 004737 047502 JSR PC,T24RT2 ;SET UP OTHER COMMAND PACKET
6879 041170

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6880
6881 ;*****
6882 ;
6883 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6884 ;
6885 ;*****
6886
6887 041174 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
6888 041200 103407 BCS 20$ ;BR IF INIT WAS OK
6889 041202 005237 002212 INC FATFLG ;BUMP COUNT
6893 041206 010001 MOV R0,R1 ;CONTENTS OF TSSR REGISTER
6894 041210 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
        041210 104455 TRAP C$ERDF
        041212 000700 .WORD 448
        041214 003650 .WORD SFIERR
        041216 012124 .WORD SFIMSG
6895 041220
6896 041220 013737 002172 044740 20$: MOV UNITN,T24DSW ;SET DRIVE NUMBER IN PACKET
6897 041226 C12704 044720 MOV $T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
6898
6899 ;*****
6900 ;
6901 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6902 ;
6903 ;*****
6904
6905 041232 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
6906 041236 103407 BCS 24$ ;BR, IF COMMAND ISSUED OK
6907 041240 005237 002212 INC FATFLG ;BUMP COUNT
6911 041244 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
6912 041246 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
        041246 104456 TRAP C$ERHRD
        041250 000701 .WORD 449
        041252 005054 .WORD WRTMSG
        041254 012124 .WORD SFIMSG
6913 041256
        041256 104406 24$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
6914
6915 ;*****
6916 ;
6917 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6918 ;
6919 ;*****
6920
6921 041260 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
6922 041264 103407 BCS 30$ ;BR, IF NO PROBLEM
6923 041266 010001 MOV R0,R1 ;SAVE TSSR
6924 041270 005237 002212 INC FATFLG ;BUMP COUNT
6928 041274 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
        041274 104456 TRAP C$ERHRD
        041276 000702 .WORD 450
        041300 046206 .WORD T24RWN
        041302 012136 .WORD PKTSSR
6929 041304
        041304 104406 30$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
6930 041306 012703 000400 MOV #256,R3 ;RECORD SIZE
6931 041312 013737 003114 045052 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS
    
```

## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

6932
6933 ;*****
6934 ;
6935 ;WRITE DATA,ACK,CVC=1,SMB COMMAND
6936 ;
6937 ;*****
6938
6939 041320 012737 150005 045050      MOV      #150005,T24PK3      ;WRITE DATA,ACK,CVC=1,SMB COMMAND
6940 041326 012704 045050      MOV      #T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
6941 041332 658:      MOV      R3,R0             ;SET PATTERN IN CORRECT REGISTER
6942 041332 010300      JSR      PC,FILLMEM        ;FILL MEMORY WITH RECORD SIZE
6943 041334 004737 017512      MOV      R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
6944 041340 010337 045056      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
6945 041344 010465 000000      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
6946 041350 004737 016340      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
6947 041354 016501 000002      MOV      #SSR,R2         ;SET UP EXPECTED
6948 041360 012702 000200      CMP      R1,R2            ;ARE THEY EQUAL
6949 041364 020102      BEQ      758              ;BR, IF OK
6950 041366 C01406      INC      FATFLG           ;BUMP COUNT
6951 041370 005237 002212      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
6952 041374      TRAP    C:ERHRD
6953 041374 104456      .WORD   451
6954 041376 000703      .WORD   WRERR
6955 041400 005111      .WORD   PKTSSR
6956 041402 012136
6957 041404 758:      CKLOOP                    ;LOOP IF SELECTED
6958 041404 104406      TRAP    C:CLP1
6959 041406 005723      TST     (R3).             ;BUMP RECORD SIZE
6960 041410 022703 000414      CMP     #268..R3         ;END OF RECORD YET
6961 041414 001346      BNE     658              ;BR, IF MORE RECORDS TO WRITE
6962 041416 808:      CKLOOP                    ;LOOP IF SELECTED
6963 041416 104406      TRAP    C:CLP1
6964 041420 005743      TST     -(R3)            ;SET RECORD SIZE BACK TO 512.
6965 041422 013737 003114 045052      MOV     FREE,T24RB       ;STARTING READ BUFFER ADDRESS
6966 ;*****
6967 ;
6968 ;READ REVERSE DATA,ACK,SMB COMMAND
6969 ;
6970 ;*****
6971 041430 012737 110401 045050      MOV     #110401,T24PK3    ;READ REVERSE DATA,ACK,SMB COMMAND
6972 041436 012704 045050      MOV     #T24PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
6973 041442 010337 045056      MOV     R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
6974 041446 010465 000000      MOV     R4,TSD8(R5)      ;ISSUE COMMAND
6975 041452 004737 016340      JSR     PC,WAITF         ;WAIT FOR SSR TO SET
6976 041456 016501 000002      MOV     TSSR(R5),R1      ;GET TSSR CONTENTS
6977 041462 012702 000200      MOV     #SSR,R2         ;SET UP EXPECTED
6978 041466 020102      CMP     R1,R2            ;ARE THEY EQUAL
6979 041470 001406      BEQ     1708             ;BR, IF OK
6980 041472 005237 002212      INC     FATFLG           ;BUMP COUNT
6981 041476 000703      ERRHRD  ERRNO,T24WDC,EXPREC ;TSSR INCORRECT AFTER READ DATA
6982 041476 104456      TRAP    C:ERHRD
6983 041500 000704      .WORD   452
6984 041502 046536      .WORD   T24WDC
6985 041504 015564      .WORD   EXPREC
6986 041506 1708:      CKLOOP                    ;LOOP IF SELECTED

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

041506 104406
6985 041510 013702 003114      MOV    FREE,R2              ;GET BUFFER ADDRESS
6986 041514 010304              MOV    R3,R4               ;CURRENT RECORD SIZE
6987 041516 162704 000400      SUB    #256.,R4           ;FIRST LOCATION IN BUFFER
6988 041522 060204 173$:      ADD    R2,R4               ;SET POINTER TO FRAME (WORD)
6989 041524 021403              CMP    (R4),R3            ;CHECK DATA READ (R3=DATA ALSO)
6990 041526 001410              BEQ    180$               ;BR, IF ALL IS WELL
6991 041530 011401              MOV    (R4),R1           ;RECD DATA
6992 041532 010302              MOV    R3,R2             ;EXPECTED DATA
6993 041534 005237 002212      INC    FATFLG             ;BUMP COUNT
6997 041540 005237 002212      ERHRD  ERRNO,T24DTA,EXP  ;DATA READ NOT = WRITTEN
041540 104456                      TRAP  C#ERHRD
041542 000705                      .WORD 453
041544 045770                      .WORD T24DTA
041546 015564                      .WORD EXPREC
6998 041550 180$:      CKLOOP                    ;LOOP IF SELECTED
041550 104406                      TRAP  C#CLP1
6999 041552 005724              TST    (R4).              ;BUMP TO NEXT LOCATION
7000 041554 160204              SUB    R2,R4               ;GET RID OF BASE ADDRESS
7001 041556 020403              CMP    R4,R3             ;END OF RECORD YET
7002 041560 001360              BNE   173$               ;BR, IF NOT AT END OF RECORD
7003 041562 005743              TST    -(R3)              ;BUMP RECORD SIZE
7004 041564 022703 000400      CMP    #256.,R3           ;END OF RECORD YET
7005 041570 001322              BNE   165$               ;BR, IF MORE RECORDS TO WRITE
7006 041572 190$:      CKLOOP                    ;LOOP IF SELECTED
041572 104406                      TRAP  C#CLP1
7007 041574 000017      ENDSUB                      ;>>>>>>>>>> END SUBTEST >>>>>>>>>>>
041574 104403                      L10061:
7008 041576 023727 002212 000017  CMP    FATFLG,#15.        ;IS ERROR COUNT AT 25
7009 041604 103402              BLO   999$                ;BR, IF LESS THAN 25
7010 041606 004737 017272      JSR   PC,CKDROP           ;TRY TO DROP THE UNIT
7011 041612 999$:
7012 ;*
7013 ;
7014 ;TEST 4, SUBTEST 8
7015 ;
7016 ;VERIFIES THAT A READ REVERSE COMMAND READING A RECORD
7017 ;LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
7018 ;STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG
7019 ; (RLL) BIT SET.
7020 ;
7021 ;
7022 ;-
7023 ;
7024 041612 BGNSUB                      ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>
041612 104402                      T4.8:
7025 041614 004737 047544      JSR   PC,T24RT3           ;SET UP OTHER COMMAND PACKET
7026 041620 004737 047410      JSR   PC,T24REST        ;SET COMMAND PACKET
7027 041624 004737 047502      JSR   PC,T24RT2        ;SET UP OTHER COMMAND PACKET
7028 ;
7029 ;*****
7030 ;
7031 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7032 ;
7033 ;*****

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7034
7035 041630 004737 016064          JSR    PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
7036 041634 103407                  BCS    20$                 ;BR IF INIT WAS OK
7037 041636 005237 002212          INC    FATFLG             ;BUMP COUNT
7041 041642 010001                  MOV    R0,R1              ;CONTENTS OF TSSR REGISTER
7042 041644                  ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C$ERDF
                                .WORD   454
                                .WORD   SFIERR
                                .WORD   SFIMSG
7043 041654
7044 041654 013737 002172 044740 20$: MOV    UNITN,T24DSW        ;SET DRIVE NUMBER 7 I PACKET
7045 041662 012704 044720          MOV    #T24PACKET,R4     ;SUBROUTINE NEEDS PACKET ADDRESS
7046
7047          ;*****
7048          ;
7049          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7050          ;
7051          ;*****
7052
7053 041666 004737 010752          JSR    PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
7054 041672 103407                  BCS    24$                 ;BR, IF COMMAND ISSUED OK
7055 041674 005237 002212          INC    FATFLG             ;BUMP COUNT
7059 041700 010001                  MOV    R0,R1              ;SAVE CONTENTS OF TSSR
7060 041702                  ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C$ERHRD
                                .WORD   455
                                .WORD   WRTMSG
                                .WORD   SFIMSG
7061 041712                  24$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    C$CLP1
7062
7063          ;*****
7064          ;
7065          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7066          ;
7067          ;*****
7068
7069 041714 004737 011104          JSR    PC,REWIND         ;CALL TAPE REWIND COMMAND
7070 041720 004737 016426          JSR    PC,CHKTSSR        ;SEE HOW TSSR IS
7071 041724 103407                  BCS    30$                 ;BR, IF NO PROBLEM
7072 041726 010001                  MOV    R0,R1              ;SAVE TSSR
7073 041730 005237 002212          INC    FATFLG             ;BUMP COUNT
7077 041734                  ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD   456
                                .WORD   T24RWN
                                .WORD   PKTSSR
7078 041744                  30$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    C$CLP1
7079 041746 012703 001000          MOV    #512.,R3          ;RECORD SIZE
7080 041752 013737 003114 045052  MOV    FREE,T24RB        ;STARTING WRITE BUFFER ADDRESS
7081
7082          ;*****
7083          ;
7084          ;WRITE DATA,ACK,CVC=1 COMMAND
7085          ;

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7086 ;*****
7087 ;
7088 041760 012737 140005 045050      MOV      #140005,T24PK3      ;WRITE DATA,ACK,CVC=1 COMMAND
7089 041766 012704 045050              MOV      #T24PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
7090 041772              65$:
7091 041772 010337 045056              MOV      R3,T24SZ         ;SET UP RECORD SIZE IN PACKET
7092 041776 010465 000000              MOV      R4,TSDB(R5)     ;ISSUE COMMAND
7093 042002 004737 016340              JSR      PC,WAITF        ;WAIT FOR SSR TO SET
7094 042006 016501 000002              MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
7095 042012 012702 000200              MOV      #SSR,R2        ;SET UP EXPECTED
7096 042016 020102              CMP      R1,R2          ;ARE THEY EQUAL
7097 042020 001406              BEQ      75$            ;BR, IF OK
7098 042022 005237 002212              INC      FATFLG         ;BUMP COUNT
7102 042026              ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
              TRAP      C#ERHRD
              .WORD    457
              .WORD    WRTErr
              .WORD    PKTSSR
7103 042036              75$:  CKLOOP              ;LOOP IF SELECTED
              TRAP      C#CLP1
7104 042040 012703 000400              MOV      #256.,R3        ;SIZE OF RECORD
7105 042044 013737 003114 045052      MOV      FREE,T24RB     ;STARTING READ BUFFER ADDRESS
7106 ;
7107 ;*****
7108 ;
7109 ;READ DATA,ACK COMMAND
7110 ;
7111 ;*****
7112 ;
7113 042052 012737 100401 045050      MOV      #100401,T24PK3  ;READ DATA,ACK COMMAND
7114 042060 012704 045050      165$:  MOV      #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
7115 042064 010337 045056              MOV      R3,T24SZ       ;SET UP RECORD SIZE IN PACKET
7116 042070 010465 000000              MOV      R4,TSDB(R5)   ;ISSUE COMMAND
7117 042074 004737 016340              JSR      PC,WAITF      ;WAIT FOR SSR TO SET
7118 042100 016501 000002              MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
7119 042104 012702 100204              MOV      #SSR!SC!BIT2,R2 ;SET UP EXPECTED
7120 042110 020102              CMP      R1,R2         ;ARE THEY EQUAL
7121 042112 001406              BEQ      170$          ;BR, IF OK
7122 042114 005237 002212              INC      FATFLG        ;BUMP COUNT
7126 042120              ERRHRD  ERRNO,T24TRL,EXPREC ;TSSR INCORRECT AFTER READ DATA
              TRAP      C#ERHRD
              .WORD    458
              .WORD    T24TRL
              .WORD    EXPREC
7127 042130              170$:  CKLOOP              ;LOOP IF SELECTED
              TRAP      L#CLP1
7128 ;
7129 ;*****
7130 ;
7131 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7132 ;
7133 ;*****
7134 ;
7135 042132 013701 044750              MOV      T24BFR+6,R1    ;GET MESSAGE BUFFER (XSTO)
7136 042136 010102              MOV      R1,R2         ;SET UP EXPECTED
7137 042140 052702 010000              BIS      #BIT12,R2     ;SET THE RLL BIT IN EXPECTED
7138 042144 020102              CMP      R1,R2         ;ARE THEY EQUAL
    
```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7139 042146 001406          BEQ      180$          ;BR, IF EQUAL (ALL IS WELL)
7140 042150 005237 002212  INC      FATFLG          ;BUMP COUNT
7144 042154          ERRHRD  ERRNO,T24LON,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
              042154 104456          TRAP      C#ERHRD
              042156 000713          .WORD    459
              042160 047022          .WORD    T24LON
              042162 015564          .WORD    EXPREC
7145 042164          180$: CKLOOP
              042164 104406          TRAP      C#CLP1
7146 042166          ENDSUB          ;>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>>
              042166          L10062: TRAP      C#ESUB
7147 042170 023727 002212 000017  CMP      FATFLG,#15.     ;IS ERROR COUNT AT 25
7148 042176 103402          BLO      999$           ;BR, IF LESS THAN 25
7149 042200 004737 017272          JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
7150 042204          999$:
7151          ;+
7152          ;
7153          ;
7154          ;TEST 4, SUBTEST 9
7155          ;
7156          ;VERIFIES THAT A READ REVERSE COMMAND SPECIFYING A DATA
7157          ;BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH
7158          ;THE PROPER ERROR STATUS WITHOUT MOVING TAPE
7159          ;-
7160
7161 042204          BGNSUB          ;>>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>>
              042204          T4.9: TRAP      C#BSUB
7162 042206 005737 003126          TST      NXMFLG          ;DO WE HAVE IT?
7163 042212 001002          BNE      10$           ;BR, IF ENOUGH
7164 042214 000137 042670          JMP      180$          ;SKIP THIS TEST IF NOT
7165 042220 004737 047544          10$: JSR      PC,T24RT3    ;SET UP OTHER COMMAND PACKET
7166 042224 004737 047410          JSR      PC,T24REST    ;SET COMMAND PACKET
7167 042230 004737 047502          JSR      PC,T24RT2    ;SET UP OTHER COMMAND PACKET
7168
7169          ;*****
7170          ;
7171          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7172          ;
7173          ;*****
7174
7175 042234 004737 016064          JSR      PC,SOFINIT    ;DO INITIALIZE ON CONTROLLER
7176 042240 103407          BCS      20$           ;BR IF INIT WAS OK
7177 042242 005237 002212          INC      FATFLG          ;BUMP COUNT
7181 042246 010001          MOV      RO,R1          ;CONTENTS OF TSSR REGISTER
7182 042250          ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
              042250 104455          TRAP      C#ERDF
              042252 000714          .WORD    460
              042254 003650          .WORD    SFIERR
              042256 012124          .WORD    SFIMSG
7183 042260          20$:
7184 042260 013737 002172 044740          MOV      UNITN,T24DSW  ;SET DRIVE NUMBER IN PACKET
7185 042266 012704 044720          MOV      @T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
7186
7187          ;*****
7188          ;

```



## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7189 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7190 ;
7191 ;*****
7192
7193 042272 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
7194 042276 103407 BCS 24# ;BR, IF COMMAND ISSUED OK
7195 042300 005237 002212 INC FATFLG ;BUMP COUNT
7199 042304 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
7200 042306 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
; TRAP C#ERHRD
; .WORD 461
; .WORD WRTMSG
; .WORD SFMSG
7201 042316 24#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
7202 042316 104406
7203 ;*****
7204 ;
7205 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7206 ;
7207 ;*****
7208
7209
7210 042320 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES SW SWITCH
7211 042324 001024 BNE 27# ;BR IF SWITCH IS ON
7212 042326 112737 000200 045061 MOVB #200,T24BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
7213 042334 112737 000010 045060 MOVB #10,T24BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
7214 042342 012704 045030 MOV #T24PK2,R4 ;WRITE SUBSYS MEM PACKET
7215 042346 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
7216 042352 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
7217 042356 103407 BCS 28# ;BR, IF NO ERROR
7218 042360 010001 MOV R0,R1 ;ERROR, SAVE TSSR
7219 042362 005237 002212 INC FATFLG ;BUMP COUNT
7223 042366 ERRHRD ERRNO,T22SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
; TRAP C#ERHRD
; .WORD 462
; .WORD T22SSR
; .WORD PKTSSR
7224 042376 27#:
7225 042376 28#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
7226 042376 104406
7227
7228
7229 042400 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
7230 042404 004737 016426 JSR PC,CHKTSSR ;SEE HOW TSSR IS
7231 042410 103407 BCS 30# ;BR, IF NO PROBLEM
7232 042412 010001 MOV R0,R1 ;SAVE TSSR
7233 042414 005237 002212 INC FATFLG ;BUMP COUNT
7237 042420 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
; TRAP C#ERHRD
; .WORD 463
; .WORD T24RWN
; .WORD PKTSSR
7238 042430 30#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
7239 042432 012703 000005 MOV #5.,R3 ;NUMBER OF RECORDS

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7240 042436 013737 003114 045052      MOV      FREE,T24RB      ;STARTING WRITE BUFFER ADDRESS
7241      ;*****
7242      ;
7243      ;WRITE DATA,ACK,CVC=1 COMMAND
7244      ;
7245      ;*****
7246
7247 042444 012737 140005 045050      MOV      #140005,T24PK3  ;WRITE DATA,ACK,CVC=1 COMMAND
7248 042452 012704 045050      MOV      #T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
7249 042456      65$:
7250 042456 012737 000256 045056      MOV      #256,T24SZ     ;SET UP RECORD SIZE IN PACKET
7251 042464 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
7252 042470 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
7253 042474 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
7254 042500 012702 000200      MOV      #SSR,R2       ;SET UP EXPECTED
7255 042504 020102      CMP      R1,R2         ;ARE THEY EQUAL
7256 042506 001406      BEQ      75$           ;BR, IF OK
7257 042510 005237 002212      INC      FATFLG        ;BUMP COUNT
7261 042514      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      042514 104456      TRAP      C$ERHRD
      042516 000720      .WORD    464
      042520 005111      .WORD    WRERR
      042522 012136      .WORD    PKTSSR
7262 042524      75$:  CKLOOP           ;LOOP IF SELECTED
      042524 104406      TRAP      C$CLP1
7263 042526 005303      DEC      R3            ;BUMP DOWN RECORD COUNTER
7264 042530 001352      BNE      65$          ;BR, IF NOT AT 5 RECORDS YET
7265 042532 012703 000400      MOV      #256.,R3     ;RECORD SIZE
7266 042536 012701 160000      MOV      #160000,R1    ;NXM LOW ADDRESS START
7267 042542 012702 177776      MOV      #177776,R2   ;LIMIT CHECK FOR NXM (HIGHEST)
7268 042546 004737 016466      JSR      PC, NXM      ;LOOK FOR NXM ADDRESS
7269 042552 103046      BCC      180$         ;BR, IF NON FOUND
7270 042554 010137 003130      MOV      R1,NXML0     ;SET ADDRESS UP FOR TEST
7271 042560 013737 003130 045052      MOV      NXML0,T24RB   ;STARTING READ BUFFER ADDRESS
7272 042566 005037 045054      CLR      T24RB+2      ;SET TO 16 BIT ADDRESSING
7273
7274      ;*****
7275      ;
7276      ;READ DATA,ACK COMMAND
7277      ;
7278      ;*****
7279
7280 042572 012737 100001 045050      MOV      #100001,T24PK3 ;READ DATA,ACK COMMAND
7281 042600 012704 045050      MOV      #T24PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
7282 042604 012737 000400 045056      MOV      #256.,T24SZ   ;SET UP RECORD SIZE IN PACKET
7283 042612 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE COMMAND
7284 042616 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
7285 042622 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
7286 042626 012702 104210      MOV      #SSR!NXM!SC!BIT3,R2 ;SET UP EXPECTED
7287 042632 020102      CMP      R1,R2        ;ARE THEY EQUAL
7288 042634 001414      BEQ      170$         ;BR, IF OK
7289 042636 005237 045054      INC      T24RB+2      ;SET TO NEXT HIGHER ADDRESSING MODE
7290 042642 023727 045054 000004      CMP      T24RB+2,#4    ;DID WE OVERFLOW INTO 19 BITS
7291 042650 001353      BNE      165$         ;BR, IF STILL IN 16-18 BITS RANGE
7292 042652 005237 002212      INC      FATFLG        ;BUMP COUNT
7296 042656      ERRHRD  ERRNO,T24NXM,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      042656 104456      TRAP      C$ERHRD

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

042660 000721                                .WORD  465
042662 045311                                .WORD  T24NXM
042664 012136                                .WORD  PKTSSR
7297 042666 170#: CKLOOP                        ;LOOP IF SELECTED                TRAP  C#CLP1
042666 104406
7298 042670 180#:
7299 042670  ENDSUB                          ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
042670 104403                                L10063:                          TRAP  C#ESUB
7300 042672 023727 002212 000017            CMP    FATFLG,#15.                ;IS ERROR COUNT AT 25
7301 042700 103402                                BLO   999#                        ;BR, IF LESS THAN 25
7302 042702 004737 017272                    JSR    PC,CKDROP                    ;TRY TO DROP THE UNIT
7303 042706 999#:
7304
7305
7306
7307 ;TEST 4. SUBTEST 10
7308 ;
7309 ;VERIFIES THAT ILLEGAL MODE-FIELD CODES IN THE READ
7310 ;COMMAND CAUSE A FUNCTION REJECT TERMINATION WITH THE
7311 ;ILLEGAL COMMAND (ILC) ERROR BIT SET.
7312 ;
7313 ;
7314 ;
7315 ;-
7316
7317 042706  BGNSUB                          ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
042706                                T4.10:
042706 104402                                TRAP  C#BSUB
7318 042710 004737 047544                    JSR    PC,T24RT3                    ;SET COMMAND PACKET UP CLEAR
7319 042714 004737 047410                    JSR    PC,T24REST                    ;SET COMMAND PACKET
7320 042720 004737 047502                    JSR    PC,T24RT2                    ;SET UP OTHER COMMAND PACKET
7321
7322 ;*****
7323 ;
7324 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7325 ;
7326 ;*****
7327
7328 042724 004737 016064                    JSR    PC,SOFINIT                    ;DO INITIALIZE ON CONTROLLER
7329 042730 103407                                BCS   20#                            ;BR IF INIT WAS OK
7330 042732 005237 002212                    INC    FATFLG                        ;BUMP COUNT
7334 042736 010001                                MOV    R0,R1                          ;CONTENTS OF TSSR REGISTER
7335 042740 042740 104455                    ERDF   ERRNO,SFIERR,SFIMSG           ;FATAL ERROR TSSR WAS NOT OK
042740 104455                                TRAP  C#ERDF
042742 000722                                .WORD  466
042744 003650                                .WORD  SFIERR
042746 012124                                .WORD  SFIMSG
7336 042750 20#:
7337 042750 013737 002172 044740            MOV    UNITN,T24DSW                    ;SET UP DRIVE NUMBER
7338 042756 012704 044720                    MOV    #T24PACKET,R4                 ;SUBROUTINE NEEDS PACKET ADDRESS
7339
7340 ;*****
7341 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7342 ;
7343 ;
7344 ;*****

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7396 043134      ENDSUB                      ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
      043134                                  L10064:
      043134 104403                               TRAP          C#ESUB
7397 043136 023727 002212 000017      CMP          FATFLG.#15.    ;IS ERROR COUNT AT 25
7398 043144 103402                               BLO          999#          ;BR, IF LESS THAN 25
7399 043146 004737 017272       JSR          PC,CKDROP     ;TRY TO DROP THE UNIT
7400 043152      999#:
7401
7402
7403
7404      ;*
7405      ;TEST 4, SUBTEST 11
7406      ;
7407      ;VERIFIES THAT ILLEGAL BUFFER ADDRESSES CAUSE A
7408      ;FUNCTION REJECT TERMINATION WITH ILLEGAL ADDRESS
7409      ;(ILA) ERROR BIT SET.
7410
7411
7412      ;-
7413
7414 043152      BGNSUB                      ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
      043152                                  T4.11:
      043152 104402                               TRAP          C#BSUB
7415 043154 004737 047544       JSR          PC,T24RT3    ;SET COMMAND PACKET UP CLEAR
7416 043160 004737 047410       JSR          PC,T24REST   ;SET COMMAND PACKET
7417 043164 004737 047502       JSR          PC,T24RT2    ;SET UP OTHER COMMAND PACKET
7418
7419      ;*****
7420      ;
7421      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7422      ;
7423      ;*****
7424
7425 043170 004737 016064       JSR          PC,SOFINIT   ;DO INITIALIZE ON CONTROLLER
7426 043174 103407                               BCS          20#          ;BR IF INIT WAS OK
7427 043176 005237 002212       INC          FATFLG      ;BUMP COUNT
7431 043202 010001                               MOV          R0,R1        ;CONTENTS OF TSSR REGISTER
7432 043204                               ERRDF        ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      043204 104455                               TRAP          C#ERDF
      043206 000726                               .WORD        470
      043210 003650                               .WORD        SFIERR
      043212 012124                               .WORD        SFIMSG
7433 043214      20#:
7434 043214 013737 002172 044740     MOV          UNITN,T24DSW ;SET UP DRIVE NUMBER
7435 043222 012704 044720     MOV          #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
7436
7437      ;*****
7438      ;
7439      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7440      ;
7441      ;*****
7442
7443 043226 004737 010752       JSR          PC,WRTCHR    ;ISSUE WRITE CHARACTERISTICS
7444 043232 103407                               BCS          24#          ;BR, IF COMMAND ISSUED OK
7445 043234 005237 002212       INC          FATFLG      ;BUMP COUNT
7449 043240 010001                               MOV          R0,R1        ;SAVE CONTENTS OF TSSR
7450 043242                               ERRHRD        ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED

```









TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7602 043710           BGNSUB                ;>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>
      043710           T4.13:
      043710 104402           TRAP          C:BSUB
7603 043712 004737 047544     JSR      PC.T24RT3           ;SET COMMAND PACKET UP CLEAR
7604 043716 004737 047410     JSR      PC.T24REST        ;SET COMMAND PACKET
7605 043722 004737 047502     JSR      PC.T24RT2        ;SET UP OTHER COMMAND PACKET
7606
7607 ;*****
7608 ;
7609 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7610 ;
7611 ;*****
7612
7613 043726 004737 016064     JSR      PC.SOFINIT        ;DO INITIALIZE ON CONTROLLER
7614 043732 103407           BCS      20$              ;BR IF INIT WAS OK
7615 043734 005237 002212     INC      FATFLG           ;BUMP COUNT
7619 043740 010001           MOV      R0,R1            ;CONTENTS OF TSSR REGISTER
7620 043742           ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      043742 104455           TRAP          C:ERDF
      043744 000735           .WORD       477
      043746 003650           .WORD       SFIERR
      043750 012124           .WORD       SFIMSG
7621 043752
7622 043752 013737 002172 044740 20$: MOV      UNITN,T24DSW      ;SET UP DRIVE NUMBER
7623 043760 012704 044720     MOV      @T24PACKET,R4   ;SUBROUTINE NEEDS PACKET ADDRESS
7624
7625 ;*****
7626 ;
7627 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
7628 ;
7629 ;*****
7630
7631 043764 004737 010752     JSR      PC.WRTPHR       ;ISSUE WRITE CHARACTERISTICS
7632 043770 103407           BCS      24$              ;BR, IF COMMAND ISSUED OK
7633 043772 005237 002212     INC      FATFLG           ;BUMP COUNT
7637 043776 010001           MOV      R0,R1            ;SAVE CONTENTS OF TSSR
7638 044000           ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTISC FAILED
      044000 104456           TRAP          C:ERHRD
      044002 000736           .WORD       478
      044004 005054           .WORD       WRTMSG
      044006 012124           .WORD       SFIMSG
7639 044010           24$: CKLOOP                ;LOOP IF SELECTED
      044010 104406           TRAP          C:CLP1
7640
7641 ;*****
7642 ;
7643 ;ISSUE REWIND COMMAND TO SELECTED APE DRIVE
7644 ;
7645 ;*****
7646
7647 044012 004737 011104     JSR      PC.REWIND       ;CALL TAPE REWIND COMMAND
7648 044016 004737 016426     JSR      PC.CKTTSSR      ;SEE HOW TSSR IS
7649 044022 103407           BCS      30$              ;BR, IF NO PROBLEM
7650 044024 010001           MOV      R0,R1            ;SAVE TSSR
7651 044026 005237 002212     INC      FATFLG           ;BUMP COUNT
7655 044032           ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      044032 104456           TRAP          C:ERHRD

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

044034 000737 .WORD 479
044036 046206 .WORD T24RWN
044040 012136 .WORD PKTSSR
7656 044042 30$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
044042 104406
7657 044044 012703 000400 MOV #256.,R3 ;RECORD SIZE
7658 044050 013737 003114 045052 MOV FREE,T24RB ;STARTINC WRITE BUFFER ADDRESS
7659
7660 ;*****
7661 ;
7662 ;READ REVERSE DATA,ACK COMMAND
7663 ;
7664 ;*****
7665
7666 044056 012737 100401 045050 MOV #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
7667 044064 012704 045050 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7668 044070
7669 044070 010337 045056 65$: MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
7670 044074 C10465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
7671 044100 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
7672 044104 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
7673 044110 012702 100206 MOV #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
7674 044114 020102 CMP R1,R2 ;ARE THEY EQUAL
7675 044116 001406 BEQ 75$ ;BR, IF OK
7676 044120 005237 002212 INC FATFLG ;BUMP COUNT
7680 044124 ERRHRD ERRNO,T24WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
044124 104456 TRAP C$ERHRD
044126 000740 .WORD 480
044130 045651 .WORD T24WDE
044132 012136 .WORD PKTSSR
7681 044134 75$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
044134 104406
7682
7683 ;*****
7684 ;
7685 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7686 ;
7687 ;*****
7688
7689 044136 013701 044750 MOV T24BFR+6,R1 ;GET MESSAGE BUFFER
7690 044142 010102 MOV R1,R2 ;SET UP EXPECTED
7691 044144 052702 002000 BIS #BIT10,R2 ;SET THE NEF BIT IN EXPECTED
7692 044150 020102 CMP R1,R2 ;ARE THEY EQUAL
7693 044152 001406 BEQ 180$ ;BR, IF EQUAL (ALL IS WELL)
7694 044154 005237 002212 INC FATFLG ;BUMP COUNT
7698 044160 ERRHRD ERRNO,T24NEF,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
044160 104456 TRAP C$ERHRD
044162 000741 .WORD 481
044164 045100 .WORD T24NEF
044166 015564 .WORD EXPREC
7699 044170 180$: CKLOOP TRAP C$CLP1
044170 104406
7700 044172 ENDSUB ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
044172 L10067: TRAP C$ESUB
044172 104403
7701 044174 023727 002212 000017 CMP FATFLG,#15. ;IS ERROR COUNT AT 25
7702 044202 103402 BLO 999$ ;BR, IF LESS THAN 25

```



TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

044310 104406                                TRAP    C$CLP1
7756
7757      ;*****
7758      ;
7759      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7760      ;
7761      ;*****
7762
7763 044312 004737 011104      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7764 044316 004737 016426      JSR      PC,CHKTSSR    ;SEE HOW TSSR IS
7765 044322 103407              BCS      30$           ;BR, IF NO PROBLEM
7766 044324 010001              MOV      R0,R1         ;SAVE TSSR
7767 044326 005237 002212      INC      FATFLG        ;BUMP COUNT
7771 044332              ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C$ERHRD
                                .WORD  484
                                .WORD  T24RWN
                                .WORD  PKTSSR
044332 104456
044334 000744
044336 046206
044340 012136
7772 044342              30$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
7773 044344 012703 000400      MOV      #256.,R3     ;RECORD SIZE
7774 044350 013737 003114 045052  MOV      FREE,T24RB   ;STARTING WRITE BUFFER ADDRESS
7775
7776      ;*****
7777      ;
7778      ;WRITE DATA,ACK,CVC=1 COMMAND
7779      ;
7780      ;*****
7781
7782 044356 012737 140005 045050      MOV      #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
7783 044364 012704 045050      MOV      #T24PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
7784 044370
7785 044370 010337 045056      65$:   MOV      R3,T24SZ   ;SET UP RECORD SIZE IN PACKET
7786 044374 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
7787 044400 004737 016340      JSR      PC,WAITF     ;WAIT FOR SSR TO SET
7788 044404 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
7789 044410 012702 000200      MOV      #SSR,R2     ;SET UP EXPECTED
7790 044414 020102              CMP      R1,R2        ;ARE THEY EQUAL
7791 044416 001406              BEQ      75$         ;BR, IF OK
7792 044420 005237 002212      INC      FATFLG        ;BUMP COUNT
7796 044424              ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C$ERHRD
                                .WORD  485
                                .WORD  WRERR
                                .WORD  PKTSSR
044424 104456
044426 000745
044430 005111
044432 012136
7797 044434              75$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP    C$CLP1
044434 104406
7798 044436 012703 000400      MOV      #256.,R3     ;RECORD SIZE
7799 044442 013737 003114 045052  MOV      FREE,T24RB   ;STARTING READ BUFFER ADDRESS
7800
7801      ;*****
7802      ;
7803      ;READ REVERSE DATA,ACK COMMAND
7804      ;
7805      ;*****
7806
7807 044450 012737 100401 045050      MOV      #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7808 044456 012704 045050          165#:  MOV    @T24PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
7809 044462 010337 045056          MOV    R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
7810 044466 010465 000000          MOV    R4,TSD8(R5)       ;ISSUE COMMAND
7811 044472 004737 016340          JSR    PC,WAITF          ;WAIT FOR SSR TO SET
7812 044476 016501 000002          MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
7813 044502 012702 000200          MOV    @SSR,R2          ;SET UP EXPECTED
7814 044506 020102                   CMP    R1,R2             ;ARE THEY EQUAL
7815 044510 001406                   BEQ    170#              ;BR, IF OK
7816 044512 005237 002212          INC    FATFLG           ;BUMP COUNT
7820 044516                   ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C#ERHRD
                                .WORD   486
                                .WORD   T24TRL
                                .WORD   PKTSSR
7821 044526                   170#:  CKLOOP           ;LOOP IF SELECTED
                                TRAP    C#CLP1
7822 044530 012703 000400          MOV    @256.,R3         ;RECORD SIZE
7823 044534 013737 003114 045052   MOV    FREE,T24RB       ;STARTING READ BUFFER ADDRESS
7824
7825 ;*****
7826 ;
7827 ;READ REVERSE DATA,ACK COMMAND
7828 ;
7829 ;*****
7830
7831 044542 012737 100401 045050   MOV    @100401,T24PK3   ;READ REVERSE DATA,ACK COMMAND
7832 044550 012704 045050          195#:  MOV    @T24PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
7833 044554 010337 045056          MOV    R3,T24SZ          ;SET UP RECORD SIZE IN PACKET
7834 044560 010465 000000          MOV    R4,TSD8(R5)       ;ISSUE COMMAND
7835 044564 004737 016340          JSR    PC,WAITF          ;WAIT FOR SSR TO SET
7836 044570 016501 000002          MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
7837 044574 012702 100204          MOV    @SSR!SC!BIT2,R2  ;SET UP EXPECTED
7838 044600 020102                   CMP    R1,R2             ;ARE THEY EQUAL
7839 044602 001406                   BEQ    200#              ;BR, IF OK
7840 044604 005237 002212          INC    FATFLG           ;BUMP COUNT
7844 044610                   ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP    C#ERHRD
                                .WORD   487
                                .WORD   T24TRL
                                .WORD   PKTSSR
7845 044620                   200#:  CKLOOP           ;LOOP IF SELECTED
                                TRAP    C#CLP1
7846 044622 013701 044756          MOV    T24BFR+14,R1     ;GET MESSAGE BUFFER (XST3)
7847 044626 010102                   MOV    R1,R2             ;SET UP EXPECTED
7848 044630 052702 000001          BIS    @BIT0,R2         ;SET THE RIB BIT IN EXPECTED
7849 044634 020102                   CMP    R1,R2             ;ARE THEY EQUAL
7850 044636 001406                   BEQ    210#              ;BR, IF EQUAL (ALL IS WELL)
7851 044640 005237 002212          INC    FATFLG           ;BUMP COUNT
7855 044644                   ERRHRD ERRNO,T24LOR,EXPREC ;THE RIB BIT WAS NOT SET IN XST0
                                TRAP    C#ERHRD
                                .WORD   488
                                .WORD   T24LOR
                                .WORD   EXPREC
7856 044654                   210#:  CKLOOP           ;LOOP IF SELECTED
                                TRAP    C#CLP1
7857 044656                   210#:  ENDSUB           ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
                                L10070:
044656

```

## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

044656 104403
7858 044660 023727 002212 000017      CMP      FATFLG,#15.      TRAP      C#ESUB
7859 044666 103402                      BLO      999#           ;IS ERROR COUNT AT 25
7860 044670 004737 017272              JSR      PC,CKDOP      ;BR, IF LESS THAN 25
7861 044674                      999#           ;TRY TO DROP THE UNIT
7862                      ;
7863                      ;
7864                      ;
7865 044674 004737 016546              JSR      PC,TSTLOOP    ;DO WE NEED TO ITERATE TEST
7866 044700 103002                      BCC      163#         ;BR, IF NO LOOP REQUIRED
7867 044702 000137 035376              JMP      T24LOOP      ;EXECUTE AGAIN
7868 044706                      163#           ;
7869 044706                      EXIT      TST         ;ALL DONE THIS TEST
044706 104432                      TRAP      C#EXIT
044710 002664                      .WORD    L10052-.

7870
7871                      ;*
7872                      ;LOCAL STORAGE FOR THIS TEST
7873                      ;-
7875 044712                      .BLKB   10-<.-TSV2&7>
7877 044720 T24PACKET:                      ;COMMAND PACKET FOR TEST
7878 044720 100204                      .WORD   100204        ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK
7879 044722 044730                      .WORD   T24DATA      ;ADDRESS OF CHARACTERISTICS BLOCK
7880 044724 000000                      .WORD   0
7881 044726 000012                      .WORD   10.          ;STARTING VALUE OF BLOCK SIZE
7882 044730 T24DATA:                      ;CHARACTERISTICS DATA BLOCK
7883 044730 044742                      .WORD   T24BFR       ;ADDRESS OF MESSAGE BUFFER
7884 044732 000000                      .WORD   0
7885 044734 000024                      .WORD   20.          ;LENGTH OF MESSAGE BUFFER
7886 044736 000000                      .WORD   0
7887 044740 000000 T24DSW: .WORD 0           ;DRIVE SELECTION BITS 2-0
7888 044742 T24BFR: .BLKW 25.         ;MESSAGE BUFFER
7889                      ;
7890                      ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
7891                      ;
7893 045024                      .BLKB   10-<.-TSV2&7>
7895 045030 T24PK2:                      ;WRITE SUB SYS MEM COMMAND, IE AND ACK
7896 045030 100206                      .WORD   100206        ;ADDRESS OF SELECT BLOCK DATA
7897 045032 045060                      .WORD   T24BF2
7898 045034 000000                      .WORD   0
7899 045036 000006                      .WORD   6.           ;SIZE OF DATA PACKET
7900
7902 045040                      .BLKB   10-<.-TSV2&7>
7904 045050 T24PK3:                      ;READ COMMAND, IE AND ACK
7905 045050 100205                      .WORD   100205
7906 045052 T24RB:                      ;ADDRESS OF WRITE BUFFER
7907 045052 003114 T24WB: .WORD FREE
7908 045054 000000                      .WORD   0
7909 045056 000000 T24SZ: .WORD 0           ;SIZE OF BUFFER (EXTENT)
7910                      .EVEN
7911                      ;
7912                      ;
7913                      ;
7914 045060 T24BF2:                      ;BSEL0 AREA
7915 045060 010 T24BS0: .BYTE 10
7916 045061 200 T24BS1: .BYTE 200
7917 045062 000000 T24S2: .WORD 0           ;SEL 2 AREA

```

## TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7918 045064 000000      T24S3: .WORD 0          ;DATA AREA
7919                    ;
7920                    ;
7921                    .EVEN
7922                    ;TAPE MOTION PACKET COMMAND VALUES
7923
7924 045066 100005      T24RN: .WORD 100005     ;READ DATA (NEXT)
7925 045070 100405      T24WDR: .WORD 100405     ;READ DATA RETRY
7926 045072 102005      T24CON: .WORD 102005     ;WRITE CONTINUOUS
7927 045074 177777      .WORD 177777           ;END OF DATA
7928 045076 000000      T24DLY: .WORD 0          ;DELAY STORAGE AREA
7929
7930
7931                    ;*
7932                    ;LOCAL TEXT MESSAGES FOR TEST
7933                    ;-
7934
7935 045100      116      105      106      T24NEF: .ASCIZ 'NEF Not Set After NON-EXECUTABLE FUNCTION'
7936 045152      122      111      102      T24LOR: .ASCIZ 'RIB Not Set After READ REVERSE Into BOT'
7937 045222      124      123      123      T24WDG: .ASCIZ 'TSSR Not Correct After Illegal Buffer Address Bits Set'
7938 045311      124      123      123      T24NXM: .ASCIZ 'TSSR Not Correct After NXM Memory Address In Packet'
7939 045375      124      123      123      T24WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
7940 045452      111      154      154      T24ILA: .ASCIZ 'Illegal Address Bits, Failed To Set ILA Bit In XST0'
7941 045536      111      154      154      T24LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XST0'
7942 045617      122      105      101      T24SSR: .ASCIZ 'READ COMMAND Not Accepted'
7943 045651      124      123      123      T24WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
7944 045723      124      141      160      T24BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
7945 045770      104      141      164      T24DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
7946 046056      122      105      101      T24EOT: .ASCIZ 'READ DATA OVER EOT GAVE NO TAPE STATUS ALERT'
7947 046133      124      123      123      T24TH: .ASCIZ 'TSSR Not Correct After READ COMMAND Reject'
7948 046206      122      145      167      T24RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
7949 046255      122      101      115      T24RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
7950 046330      124      123      123      T24AM3: .ASCIZ 'TSSR Init. Failed After READ COMMAND'
7951 046375      104      162      151      T24OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
7952 046450      124      123      123      T24WDD: .ASCIZ 'TSSR Not Correct After READ DATA Command, SWB Bit Set'
7953 046536      124      123      123      T24WDC: .ASCIZ 'TSSR Not Correct After READ DATA Command'
7954 046607      103      126      103      T24VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
7955 046662      124      123      102      T24BA: .ASCIZ 'TSBA Not Correct After READ DATA Command'
7956 046733      127      122      111      T24WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
7957 047022      122      145      141      T24LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XST0'
7958 047104      122      145      141      T24LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XST0'
7959 047166      122      145      163      T24PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
7960 047254      122      145      141      T24TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
7961 047342      102      141      163      TST24ID: .ASCIZ 'Basic Read Data (Forward and Reverse)'
7962                    .EVEN
7963                    ;*
7964                    ;
7965                    ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
7966                    ;WRITE SUBSYSTEM MEMORY COMMAND
7967                    ;
7968                    ;-
7969
7970 047410      T24REST:
7971 047410      SAVREG          ;SAVE THE REGISTERS
7972 047414      012701 044720  MOV #T24PACKET,R1      ;START OF THE PACKET
7973 047420      012721 100004  MOV #100004,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK.
7974 047424      012721 044730  MOV #T24DATA,(R1)+      ;ADDRESS OF CHARAISTICS DATA BLOCK

```

TEST 4: BASIC READ DATA (FORWARD AND REVERSE)

```

7975 047430 005021          CLR      (R1)+          ;EXTENDED ADDRESS
7976 047432 012721 000012  MOV      #10.,(R1)+     ;SIZE OF DATA BLOCK IN BYTES
7977 047436 012721 044742  MOV      #T24BFR,(R1)+ ;ADDRESS OF MESSAGE BUFFER
7978 047442 005021          CLR      (R1)+
7979 047444 012721 000024  MOV      #20.,(R1)+     ;LENGTH OF MESSAGE BUFFER
7980 047450 005021          CLR      (R1)+
7981 047452 012711 000000  MOV      #0,(R1)        ;SELECT DRIVE ZERO
7982 047456 012702 000030  MOV      #24.,R2        ;NUMBER OF LOCATIONS TO BE CLEARED
7983 047462 012762 177777 044742 64#  MOV      #177777,T24BFR(R2) ;ALL ONES TO MESSAGE BUFFER
7984 047470 005742          TST      -(R2)          ;NEXT LOCATION
7985 047472 022702 000000  CMP      #0,R2          ;CHECK FOR END OF LOOP
7986 047476 001371          BNE      64#            ;KEEP GOING UNTIL DONE
7987 047500 000207          RTS      PC            ;RETURN
7988
7989

```

```

7990 047502          T24RT2:
7991 047502          SAVREG          ;SAVE THE REGISTERS
7992 047506 012701 045030  MOV      #T24PK2,R1     ;START OF THE PACKET
7993 047512 012721 100206  MOV      #100206,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK. IE
7994 047516 012721 045060  MOV      #T24BF2,(R1)+ ;ADDRESS OF DATA BLOCK
7995 047522 005021          CLR      (R1)+          ;EXTENDED ADDRESS
7996 047524 012721 000006  MOV      #6.,(R1)+     ;SIZE OF DATA BLOCK IN BYTES
7997 047530 005021          CLR      (R1)+
7998 047532 012701 045060  MOV      #T24BF2,R1     ;POINT TO DATA SEL AREA
7999 047536 005021          CLR      (R1)+
8000 047540 005011          CLR      (R1)
8001 047542 000207          RTS      PC            ;RETURN
8002 047544

```

```

8003 047544          T24RT3:
8004 047550 012701 045050  SAVREG          ;SAVE THE REGISTERS
8005 047554 012721 000000  MOV      #T24PK3,R1     ;START OF THE PACKET
8006 047560 012721 000000  MOV      #0,(R1)+      ;CLEAR AREA OUT
8007 047564 005021          MOV      #0,(R1)+      ;ADDRESS OF DATA BLOCK
8008 047566 012711 000000  CLR      (R1)+          ;EXTENDED ADDRESS
8009 047572 000207          MOV      #0,(R1)        ;SIZE OF DATA BLOCK IN BYTES
8010 047574          RTS      PC            ;RETURN
      ENDTST

```

L10052: TRAP C#ETST

8011 .SBTTL TEST 5: SPACE RECORDS

```

8012 ;*
8013 ;
8014 ; THIS TEST VERIFIES THAT THE SPACE RECORDS FORWARD AND SPACE
8015 ; RECORDS REVERSE POSITION COMMANDS OPERATE PROPERLY WHEN SPACING
8016 ; OVER NORMAL DATA RECORDS. OPERATION WHEN SPACING OVER TAPE MARKS
8017 ; IS VERIFIED IN A SUBSEQUENT TEST. THE BASIC WRITE DATA TEST
8018 ; SHOULD HAVE BEEN RUN SUCCESSFULLY FOR THIS TEST TO PRODUCE MEANINGFUL
8019 ; RESULTS. THIS TEST CONSISTS OF A SERIES OF SUBTESTS. IN EACH
8020 ; OF THE SUBTESTS, THE TAPE IS ENTIRELY WRITTEN WITH RECORDS
8021 ; OF VARYING SIZES AND DATA CONTENT; THE FIRST 4 BYTES OF EACH
8022 ; RECORD INDICATE THAT RECORD'S RELATIVE POSITION ON TAPE. AFTER
8023 ; EACH SPACING OPERATION, THE TAPE POSITION IS VERIFIED BY READING
8024 ; THE NEXT OR PREVIOUS RECORD AND COMPARING THE POSITION DATA WITH
8025 ; THE EXPECTED RESULT.
8026 ;
8027 ;
8028 ;
8029 ; THE TEST CONSISTS OF THE FOLLOWING 8 SUBTESTS

```



TEST 5: SPACE RECORDS

```

8030
8031
8032
8033
8034 047576          BGNTST
      047576
8035 047576 012737 006446 002170      MOV    #EPRT2,EPRTSW      ;SECONDARY ERROR MESSAGE
8040 047604 004737 017364              JSR    PC,KTOFF          ;DON'T NEED LOTS OF MEMORY
8041 047610 012700 056320              MOV    #TST25ID,R0      ;ASCII MESSAGE TO IDENTIFY TEST
8042 047614 004737 016600              JSR    PC,TSTSETUP      ;DO INITIAL TEST SETUP
8043 047620 012737 000005 002206      MOV    #5,LOOPCNT      ;PERFORM 5 ITERATIONS
8044
8045
8046
8047
8048
8049
8050
8051
8052
8053
8054
8055
8056 047626          T25LOOP:
8057 047626          BGNSUB
      047626
      104402
8058 047630 004737 056336              JSR    PC,T25REST      ;SET COMMAND PACKET
8059 047634 005037 055150              CLR    T25CNT          ;CLEAR THE RECORD COUNTER AREA
8060 047640 004737 056430              JSR    PC,T25RT2       ;SET UP OTHER COMMAND PACKET
8061 047644 004737 056472              JSR    PC,T25RT3       ;SET UP OTHER COMMAND PACKET
8062 047650 012737 176750 055152      MOV    #65000.,T25CLY  ;SET UP LOOP COUNTER
8063
8064
8065
8066
8067
8068
8069
8070 047656 004737 016064          5$:   JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
8071 047662 103427              BCS    10$             ;BR IF INIT WAS OK
8072 047664              DELAY  250            ;DELAY IF REQUIRED
      047664 012727 000250              MOV    #250,(PC)+
      047670 000000              .WORD 0
      047672 013727 002116              MOV    L#DLY,(PC)+
      047676 000000              .WORD 0
      047700 005367 177772              DEC    -6(PC)
      047704 001375              BNE    -.4
      047706 005367 177756              DEC    -22(PC)
      047712 001367              BNE    .-20
8073 047714 005337 055152          DEC    T25DLY          ;DEC DELAY COUNTER
8074 047720 001356              BNE    5$             ;BR. IF LOOP IS REQUIRED
8075 047722 005237 002212          INC    FATFLG          ;BUMP COUNT
8079 047726 016501 000002          MOV    TSSR(R5),R1    ;CONTENTS OF TSSR REGISTER
8080 047732          ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      047732 104455              TRAP   C#ERDF
      047734 000765              .WORD 501

```

TEST 5: SPACE RECORDS

```

047736 003650 .WORD SFIERR
047740 012124 .WORD SFIMSG
8081 047742
8082 047742 013737 002172 055010 10#: MOV UNITN,T25DSW ;SET UP DRIVE NUMBER
8083 047750 012704 054770 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
8084
8085 ;*****
8086 ;
8087 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8088 ;
8089 ;*****
8090
8091 047754 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
8092 047760 103407 BCS 15# ;BR, IF COMMAND ISSUED OK
8093 047762 005237 002212 INC FATFLG ;BUMP COUNT
8097 047766 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
8098 047770 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
047770 104456 TRAP C#ERHRD
047772 000766 .WORD 502
047774 005054 .WORD WRTMSG
047776 012124 .WORD SFIMSG
8099
8100 ;*****
8101 ;
8102 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8103 ;
8104 ;*****
8105
8106 050000 15#: CKLOOP
050000 104406 TRAP C#CLP1
8107 050002 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
8108 050006 103407 BCS 30# ;BR, IF NO PROBLEM
8109 050010 010001 MOV RO,R1 ;SAVE TSSR
8110 050012 005237 002212 INC FATFLG ;BUMP COUNT
8114 050016 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
050016 104456 TRAP C#ERHRD
050020 000767 .WORD 503
050022 056125 .WORD T25RWN
050024 012136 .WORD PKTSSR
8115 050026 30#: CKLOOP ;LOOP IF SELECTED
050026 104406 TRAP C#CLP1
8116
8117 ;*****
8118 ;
8119 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8120 ;
8121 ;*****
8122
8123 050030 013701 055020 MOV T258FR+6,R1 ;PICK UP XST0
8124 050034 010102 MOV R1,R2 ;SET UP EXPECTED
8125 050036 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
8126 050042 020102 CMP R1,R2 ;DOES EXP = REC'D
8127 050044 001406 BEQ 40# ;BR, IF EQUAL (OK)
8128 050046 005237 002212 INC FATFLG ;BUMP COUNT
8132 050052 ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
050052 104456 TRAP C#ERHRD
050054 000770 .WORD 504

```

TEST 5: SPACE RECORDS

```

050056 055315 .WORD T2580T
050060 015564 .WORD EXPREC
8133 050062 40#: CKLOOP ;LOOP IF SELECTED TRAP C:CLP1
050062 104406 ;RECORD SIZE
8134 050064 012703 000400 MOV #256.,R3 ;STARTING WRITE BUFFER ADDRESS
8135 050070 013737 003114 055122 MOV FREE,T258B
;*****
;WRITE DATA,ACK,CVC=1 COMMAND
;*****
8143 050076 012737 140005 055120 MOV #140005,T25PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
8144 050104 012704 055120 MOV #T25PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
8145 050110 65#: MOV R3,T25SZ ;SET UP RECORD SIZE IN PACKET
8146 050110 010337 055126 MOV T25CNT,BFREE ;LOAD UP RECORD COUNTER IN WRT BUFFER
8147 050114 013777 055150 132772 ADD #1,T25CNT ;GET READY FOR NEXT RECORD
8148 050122 C62737 000001 055150 MOV R4,TSDB(R5) ;ISSUE COMMAND
8149 050130 010465 000000 JSR PC,WAIF ;WAIT FOR SSR TO SET
8150 050134 004737 016340 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
8151 050140 016501 000002 MOV #SSR,R2 ;SET UP EXPECTED
8152 050144 012702 000200 CMP R1,R2 ;ARE THEY EQUAL
8153 050150 020102 BEQ 75# ;BR, IF OK
8154 050152 001411 BIT #BIT2,R1 ;CHECK FOR TAPE STATUS ALERT
8155 050154 032701 000004 BNE 120# ;BR, IF TSA IS SET (SUSPECT IS EOT)
8156 050160 001014 INC FATFLG ;BUMP COUNT
8157 050162 005237 002212 ERRHRD ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
8161 050166 104456 TRAP C:ERHRD
050170 000771 .WORD 505
050172 005111 .WORD WRTERR
050174 012136 .WORD PKTSSR
8162 050176 75#: CKLOOP ;LOOP IF SELECTED TRAP C:CLP1
050176 104406 ;BUMP RECORD SIZE
8163 050200 005203 INC R3 ;END OF RECORD YET
8164 050202 022703 001000 CMP #512.,R3 ;BR, IF MORE RECORDS TO WRITE
8165 050206 001340 BNE 65# ;ENOUGH RECORDS
8166 050210 000415 BR 125#
8167 050212 120#:
;*****
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;*****
8175 050212 013701 055020 MOV T258FR+6,R1 ;QUICK CHECK FOR EOT SET
8176 050216 010102 MOV R1,R2 ;SET UP EXPECTED
8177 050220 052702 000001 BIS #BIT0,R2 ;SET THE EOT BIT XSTO
8178 050224 020102 CMP R1,R2 ;IS THE EOT BIT SET IN XSTO
8179 050226 001406 BEQ 125# ;BR, IF SET (GOOD)
8180 050230 005237 002212 INC FATFLG ;BUMP COUNT
8184 050234 005237 002212 ERRDF ERRNO,T25NET,EXPREC ;DEVICE FATAL NOT EOT FOUND ETC.
050234 104455 TRAP C:ERDF
050236 000772 .WORD 506
050240 055451 .WORD T25NET

```

TEST 5: SPACE RECORDS

```

8185 050242 015564 .WORD EXPREC
8186 050244
8187
8188
8189
8190
8191
8192
8193 050244 004737 011104 JSR PC.REWIND ;CALL TAPE REWIND COMMAND
8194 050250 103407 BCS 1304 ;BR, IF NO PROBLEM
8195 050252 010001 MOV R0,R1 ;SAVE TSSR
8196 050254 005237 002212 INC FATFLG ;BUMP COUNT
8200 050260 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
      050260 104456 TRAP C1ERRRD
      050262 000773 .WORD 507
      050264 056125 .WORD T25RWN
      050266 012136 .WORD PKTSSR
8201 050270 1304: CKLOOP ;LOOP IF SELECTED TRAP C1CLP1
      050270 104406
8202 050272 012737 000007 055010 MOV #7,T25DSW ;SET UP DRIVE NUMBER
8203 050300 012704 054770 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
8204
8205
8206
8207
8208
8209
8210
8211 050304 004737 010752 JSR PC.WRTCHR ;ISSUE WRITE CHARACTERISTICS
8212 050310 103407 BCS 1404 ;BR, IF COMMAND ISSUED OK
8213 050312 005237 002212 INC FATFLG ;BUMP COUNT
8217 050316 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
8218 050320 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
      050320 104456 TRAP C1ERRRD
      050322 000774 .WORD 508
      050324 005054 .WORD WRTMSG
      050326 012124 .WORD SFMSG
8219 050330 1404: CKLOOP ;SCOPE LOOP TRAP C1CLP1
      050330 104406
8220 050332 005737 002216 TST EXTFEA ;CHECK FOR EXTENDED FEATURES
8221 050336 001044 ONE 1604 ;BR IF SWITCH IS ON
8222
8223 050340 112737 000200 055131 MOVB #200,T25BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
8224 050346 112737 000010 055130 MOVB #10,T25BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
8225 050354 012704 055100 MOV #T25PK2,R4 ;WRITE SUBSYS MEM PACKET
8226 050360 010465 000000 MOV R4,T5DB(R5) ;ISSUE COMMAND
8227 050364 004737 016426 JSR PC.CKTTSSR ;WAIT FOR SSR
8228 050370 103407 BCS 1504 ;BR, IF NO ERROR
8229 050372 010001 MOV R0,R1 ;ERROR, SAVE TSSR
8230 050374 005237 002212 INC FATFLG ;BUMP COUNT
8234 050400 ERRHRD ERRNO,T25SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
      050400 104456 TRAP C1ERRRD
      050402 000775 .WORD 509
      050404 055154 .WORD T25SSR
      050406 012136 .WORD PKTSSR
8235 050410 1504: CKLOOP ;LOOP IF SELECTED

```

TEST 5: SPACE RECORDS

```

050410 104406
8236 050412 012737 000007 055010          MOV      #7,T25DSW          ;SET UP DRIVE NUMBER
8237 050420 012704 054770          MOV      #T25PACKET,R4     ;SUBROUTINE NEEDS PACKET ADDRESS
8238
8239          ;*****
8240          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8241          ;*****
8242
8243
8244
8245 050424 004737 010752          JSR      PC,WRTCHR         ;ISSUE WRITE CHARACTERISTICS
8246 050430 103407          BCS      160$             ;BR, IF COMMAND ISSUED OK
8247 050432 005237 002212          INC      FATFLG           ;BUMP COUNT
8251 050436 010001          MOV      R0,R1           ;SAVE CONTENTS OF TSSR
8252 050440          ERRHRD  ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTICSC FAILED
          104456          TRAP    C#ERHRD
          050442 000776          .WORD   510
          050444 005054          .WORD   WRTMSG
          050446 C12124          .WORD   SFMSG
8253 050450          160$:  CKLOOP           ;SCOPE LOOP
          050450 104406          TRAP    C#CLP1
8254 050452 016501 000002          MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
8255 050456 032701 000100          BIT      #OFL,R1         ;CHECK FOR THE OFFLINE BIT SET
8256 050462 001006          BNE     170$             ;BR, IF OFFLINE (GOOD)
8257 050464 005237 002212          INC      FATFLG           ;BUMP COUNT
8261 050470          ERRDF  ERRNO,T25OFL,SFMSG ;OFF LINE SHOULD HAVE BEEN SET (BAD)
          050470 104455          TRAP    C#ERDF
          050472 000777          .WORD   511
          050474 056174          .WORD   T25OFL
          050476 012124          .WORD   SFMSG
8262 050500          170$:  CKLOOP           ;LOOP IF SELECTED
          050500 104406          TRAP    C#CLP1
8263
8264          ;*****
8265          ;SPACE FORWARD COMMAND IN PLACE
8266          ;*****
8267
8268
8269
8270 050502 012737 140010 055120 180$:  MOV      #140010,T25PK3    ;SPACE FORWARD COMMAND IN PLACE
8271 050510 012737 000001 055122  MOV      #1,T25R8         ;NUMBER OF RECORDS TO SPACE
8272 050516 012704 055120          MOV      #T25PK3,R4      ;R4 = POINTER TO PACKET
8273 050522 010465 000000          MOV      R4,T25DB(R5)    ;ISSUE COMMAND
8274 050526 004737 016340          JSR      PC,WAITF        ;WAIT FOR SSR TO SET
8275 050532 016501 000002          MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
8276 050536 012702 100306          MOV      #SSR!SC!OFL!BIT1!BIT2,R2 ;SET UP EXPECTED
8277 050542 020102          CMP      R1,R2           ;ARE THEY EQUAL
8278 050544 001406          BEQ     190$             ;BR, IF OK ESP. FUNCTION REJECT
8279 050546 005237 002212          INC      FATFLG           ;BUMP COUNT
8283 050552          ERRHRD  ERRNO,T25TM,PKTSSR ;TSSR INCORRECT AFTER TAPE MOTION CMD
          104456          TRAP    C#ERHRD
          050554 001000          .WORD   512
          050556 055362          .WORD   T25TM
          050560 012136          .WORD   PKTSSR
8284 050562          190$:  CKLOOP           ;LOOP IF SELECTED
          050562 104406          TRAP    C#CLP1
8285 050564          ENDSUB                ;>>>>>>>>>> END SUBTEST >>>>>>>>>>

```



TEST 5: SPACE RECORDS

```

050676 005054 .WORD WRTMSG
050700 012124 .WORD SFIMSG
8339
8340 ;*****
8341 ;
8342 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8343 ;
8344 ;*****
8345
8346 050702 15#: CKLOOP TRAP C#CLP1
050702 104406 ;CALL TAPE REWIND COMMAND
8347 050704 004737 011104 JSR PC,REWIND ;BR, IF NO PROBLEM
8348 050710 103407 BCS 30# ;SAVE TSSR
8349 050712 010001 MOV R0,R1 ;BUMP COUNT
8350 050714 005237 002212 INC FATFLG ;REWIND NOT ACCEPTED
8354 050720 ERRHRD ERRNO,T25RWN,PKTSSR
050720 104456 TRAP C#ERHRD
050722 001003 .WORD 515
050724 C56125 .WORD T25RWN
050726 012136 .WORD PKTSSR
8355 050730 30#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
050730 104406 ;CHECK FOR EXTENDED FEATURES SW SWITCH
8356 050732 005737 002216 140#: TST EXTFEA ;BR IF SWITCH IS ON
8357 050736 001044 BNE 160#
8358
8359 050740 112737 000200 055131 MOVB #200,T25BS1 ;WRITE MISCELLANEOUS CONT/READ STATUS
8360 050746 112737 000010 055130 MOVB #10,T25BS0 ;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)
8361 050754 012704 055100 MOV #T25PK2,R4 ;WRITE SUBSYS MEM PACKET
8362 050760 010465 000000 MOV R4,T5DB(R5) ;ISSUE COMMAND
8363 050764 004737 016426 JSR PC,CHKTSSR ;WAIT FOR SSR
8364 050770 103407 BCS 150# ;BR, IF NO ERROR
8365 050772 010001 MOV R0,R1 ;ERROR, SAVE TSSR
8366 050774 005237 002212 INC FATFLG ;BUMP COUNT
8370 051000 ERRHRD ERRNO,T25SSR,PKTSSR ;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS
051000 104456 TRAP C#ERHRD
051002 001004 .WORD 516
051004 055154 .WORD T25SSR
051006 012136 .WORD PKTSSR
8371 051010 150#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
051010 104406 ;SET UP DRIVE NUMBER
8372 051012 012737 000007 055010 MOV #7,T25DSW ;SUBROUTINE NEEDS PACKET ADDRESS
8373 051020 012704 054770 MOV #T25PACKET,R4
8374
8375 ;*****
8376 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8377 ;
8378 ;*****
8379
8380
8381 051024 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
8382 051030 103407 BCS 160# ;BR, IF COMMAND ISSUED OK
8383 051032 005237 002212 INC FATFLG ;BUMP COUNT
8387 051036 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
8388 051040 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
051040 104456 TRAP C#ERHRD
051042 001005 .WORD 517
051044 005054 .WORD WRTMSG

```





TEST 5: SPACE RECORDS

T5.3:

```

051202
051202 104402
8438 051204 004737 056336 JSR PC,T25REST ;SET COMMAND PACKET TRAP C#BSUB
8439 051210 004737 056430 JSR PC,T25RT2 ;SET UP OTHER COMMAND PACKET
8440 051214 004737 056472 JSR PC,T25RT3 ;SET UP OTHER COMMAND PACKET
8441
8442 ;*****
8443 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
8444 ;
8445 ;*****
8446
8447
8448 051220 004737 016064 JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
8449 051224 103407 BCS 10# ;BR IF INIT WAS OK
8450 051226 005237 002212 INC FATFLG ;BUMP COUNT
8454 051232 010001 MOV RO,R1 ;CONTENTS OF TSSR REGISTER
8455 051234 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
051234 104455 TRAP C#ERDF
051236 C01010 .WORD 520
051240 003650 .WORD SFIERR
051242 012124 .WORD SFIMSG
8456 051244 013737 002172 055010 10#: MOV UNITN,T25DSW ;SET UP DRIVE NUMBER
8457
8458 051252 012704 054770 MOV #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
8459
8460 ;*****
8461 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8462 ;
8463 ;*****
8464
8465
8466 051256 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
8467 051262 103407 BCS 15# ;BR, IF COMMAND ISSUED OK
8468 051264 005237 002212 INC FATFLG ;BUMP COUNT
8472 051270 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
8473 051272 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICSC FAILED
051272 104456 TRAP C#ERHRD
051274 001011 .WORD 521
051276 005054 .WORD WRTMSG
051300 012124 .WORD SFIMSG
8474
8475 ;*****
8476 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8477 ;
8478 ;*****
8479
8480
8481 15#: CKLOOP TRAP C#CLP1
051302 104406
8482 051304 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
8483 051310 103407 BCS 30# ;BR, IF NO PROBLEM
8484 051312 010001 MOV RO,R1 ;SAVE TSSR
8485 051314 005237 002212 INC FATFLG ;BUMP COUNT
8489 051320 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
051320 104456 TRAP C#ERHRD
051322 001012 .WORD 522
051324 056125 .WORD T25RWN
    
```

TEST 5: SPACE RECORDS

```

8490 051326 012136
      051330 104406
      051330 104406
8491
8492
8493
8494
8495
8496
8497
8498 051332 013701 055020
8499 051336 010102
8500 051340 052702 000002
8501 051344 020102
8502 051346 001406
8503 051350 005237 002212
8507 051354
      051354 104456
      051356 001013
      051360 055315
      051362 015564
8508 051364
      051364 104406
8509 051366 012737 000001 055122
8510
8511
8512
8513
8514
8515
8516
8517 051374 012737 140010 055120
8518 051402 012704 055120
8519 051406
8520 051406 010465 000000
8521 051412 004737 016340
8522 051416 016501 000002
8523 051422 012702 000200
8524 051426 020102
8525 051430 001411
8526 051432 032701 000004
8527 051436 001006
8528 051440 005237 002212
8532 051444
      051444 104456
      051446 001014
      051450 055235
      051452 015564
8533 051454
      051454 104406
8534 051456
8535
8536
8537
8538
8539
8540

```

```

30$:  CKLOOP                ;LOOP IF SELECTED                .WORD  PKTSSR
                                           TRAP  C#CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;
;*****
      MOV     T25BFR+6,R1      ;PICK UP XSTO
      MOV     R1,R2           ;SET UP EXPECTED
      BIS     #BIT1,R2        ;SET BOT BIT IN EXPECTED
      CMP     R1,R2           ;DOES EXP = REC'D
      BEQ     40$             ;BR, IF EQUAL (OK)
      INC     FATFLG          ;BUMP COUNT
      ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                           TRAP  C#ERHRD
                                           .WORD  523
                                           .WORD  T25BOT
                                           .WORD  EXPREC
40$:  CKLOOP                ;LOOP IF SELECTED                .WORD  PKTSSR
                                           TRAP  C#CLP1
      MOV     #000001,T25RB   ;NUMBER OF RECORDS TO SPACE OVER
;*****
;
;SPACE FORWARD,ACK,CVC-1 COMMAND
;
;*****
      MOV     #140010,T25PK3  ;SPACE FORWARD,ACK,CVC-1 COMMAND
      MOV     #T25PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
65$:  MOV     R4,TSDB(R5)     ;ISSUE COMMAND
      JSR     PC,WAITF        ;WAIT FOR SSR TO SET
      MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
      MOV     #SSR,R2        ;SET UP EXPECTED
      CMP     R1,R2          ;ARE THEY EQUAL
      BEQ     75$            ;BR, IF OK
      BIT     #BIT2,R1       ;CHECK FOR TAPE STATUS ALERT
      BNE     75$            ;BR, IF TSA IS SET (SUSPECT IS EOT)
      INC     FATFLG          ;BUMP COUNT
      ERRHRD ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                           TRAP  C#ERHRD
                                           .WORD  524
                                           .WORD  T25WDE
                                           .WORD  EXPREC
75$:  CKLOOP                ;LOOP IF SELECTED                .WORD  PKTSSR
                                           TRAP  C#CLP1
120$:
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;
;*****

```

TEST 5: SPACE RECORDS

```

8541
8542 051456 013701 055020          MOV     T25BFR+6,R1       ;QUICK CHECK FOR BOT SET
8543 051462 010102                MOV     R1,R2           ;SET UP EXPECTED
8544 051464 042702 000002          BIC     #8BIT1,R2       ;CLEAR THE BOT BIT (XSTO)
8545 051470 020102                CMP     R1,R2           ;IS THE EOT BIT SET IN XSTO
8546 051472 001406                BEQ     125$            ;BR, IF SET (GOOD)
8547 051474 005237 002212          INC     FATFLG          ;BUMP COUNT
8551 051500                ERRHRD  ERRNO,T25BNC,EXPREC ;BOT NOT CLEARED AFTER SPACE FROM BOT
                                TRAP     C#ERHRD
                                .WORD   525
                                .WORD   T25BNC
                                .WORD   EXPREC
                                TRAP     C#CLP1
8552 051510                125$:  CKLOOP
8553 051512 104406                JSR     PC,T25RT3       ;CLEAN UP PACKET
8554 051516 004737 056472          MOV     #257.,T25SZ    ;SET THE CORRECT SIZE UP
8555
8556 ;*****
8557 ;
8558 ;READ DATA COMMAND IN PLACE
8559 ;
8560 ;*****
8561
8562 051524 012737 140001 055120          MOV     #140001,T25PK3 ;READ DATA COMMAND IN PLACE
8563 051532 013737 003114 055122          MOV     FREE,T25RB     ;READ BUFFER ADDRESS TO PACKET
8564 051540 012704 055120          MOV     #T25PK3,R4    ;R4 = POINTER TO PACKET
8565 051544 010465 000000          MOV     R4,TSDB(R5)   ;ISSUE COMMAND
8566 051550 004737 016340          JSR     PC,WAITF       ;WAIT FOR SSR TO SET
8567 051554 016501 000002          MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
8568 051560 012702 000200          MOV     #SSR,R2       ;SET UP EXPECTED
8569 051564 020102                CMP     R1,R2          ;ARE THEY EQUAL
8570 051566 001406                BEQ     190$            ;BR, IF OK ESP. FUNCTION REJECT
8571 051570 005237 002212          INC     FATFLG          ;BUMP COUNT
8575 051574                ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA CMD
                                TRAP     C#ERHRD
                                .WORD   526
                                .WORD   RDERR
                                .WORD   PKTSSR
8576 051604                190$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP     C#CLP1
8577 051606 017701 131302          MOV     @FREE,R1       ;GET FIRST WORD FROM BUFFER
8578 051612 012702 000001          MOV     #1,R2         ;SET UP EXPECTED
8579 051616 020102                CMP     R1,R2          ;WAS RECORD NUMBERED 1
8580 051620 001406                BEQ     200$            ;BR, IF CORRECT RECORD
8581 051622 005237 002212          INC     FATFLG          ;BUMP COUNT
8585 051626                ERRHRD  ERRNO,T25WNG,EXPREC ;SHOULD HAVE BEEN RECORD NUMBER 1
                                TRAP     C#ERHRD
                                .WORD   527
                                .WORD   T25WNG
                                .WORD   EXPREC
8586 051636                200$:  CKLOOP
8587 051640                ENDSUB                ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
                                L10074:
                                TRAP     C#CLP1
                                TRAP     C#ESUB
8588 051642 023727 002212 000017          CMP     FATFLG,#15.    ;IS ERROR COUNT AT 25
8589 051650 103402                BLO     999$           ;BR, IF LESS THAN 25

```



TEST 5: SPACE RECORDS

```

8642 ;*****
8643 ;
8644 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8645 ;
8646 ;*****
8647
8648 051760 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
8649 051764 103407                BCS    30$                ;BR, IF NO PROBLEM
8650 051766 010001                MOV    R0,R1              ;SAVE TSSR
8651 051770 005237 002212        INC    FATFLG             ;BUMP COUNT
8655 051774                ERRHRD  ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C#ERHRD
                                .WORD   530
                                .WORD   T25RWN
                                .WORD   PKTSSR
8656 052004 104406          30$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1
8657 ;*****
8658 ;
8659 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8660 ;
8661 ;*****
8662 ;
8663
8664 052006 013701 055020          MOV    T25BFR+6,R1        ;PICK UP XST0
8665 052012 010102                MOV    R1,R2              ;SET UP EXPECTED
8666 052014 052702 000002        BIS    #BIT1,R2           ;SET BOT BIT IN EXPECTED
8667 052020 020102                CMP    R1,R2              ;DOES EXP = REC'D
8668 052022 001406                BEQ    40$                ;BR, IF EQUAL (OK)
8669 052024 005237 002212        INC    FATFLG             ;BUMP COUNT
8673 052030                ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C#ERHRD
                                .WORD   531
                                .WORD   T25BOT
                                .WORD   EXPREC
8674 052040 104406          40$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1
8675 ;*****
8676 ;
8677 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
8678 ;BIT 15 SETS DIRECTION - 0=FORWARD  1=REVERSE
8679 ;
8680 ;*****
8681 ;
8682
8683 052042 012703 000001          MOV    #000001,R3        ;NUMBER OF RECORDS TO SPACE FORWARD
8684 052046 004737 010556        JSR    PC,SPACE          ;CALL SPACE COMMAND
8685 052052 103410                BCS    50$                ;CHECK FOR ERROR
8686 052054 016501 000002        MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
8687 052060 005237 002212        INC    FATFLG             ;BUMP COUNT
8691 052064                ERRHRD  ERRNO,T25WDE,SFFMSG ;SPACE FORWARD FAILED
                                TRAP    C#ERHRD
                                .WORD   532
                                .WORD   T25WDE
                                .WORD   SFFMSG
8692 052074 104406          50$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1

```

TEST 5: SPACE RECORDS

```

8693 052076 012737 000001 055122      MOV      #1,T25RB      ;NUMBER OF RECORDS TO SPACE OVER
8694
8695      ;*****
8696      ;
8697      ;SPACE REVERSE,ACK,CVC=1 COMMAND
8698      ;
8699      ;*****
8700
8701 052104 012737 140410 055120      MOV      #140410,T25PK3 ;SPACE REVERSE,ACK,CVC=1 COMMAND
8702 052112 012704 055120      MOV      #T25PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
8703 052116      65$:
8704 052116 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE COMMAND
8705 052122 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
8706 052126 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
8707 052132 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
8708 052136 020102      CMP      R1,R2        ;ARE THEY EQUAL
8709 052140 001406      BEQ      75$          ;BR, IF OK
8710 052142 005237 002212      INC      FATFLG       ;BUMP COUNT
8714 052146      ERRHRD  ERRNO,T25WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      052146 104456      TRAP      C#ERHRD
      052150 001025      .WORD    533
      052152 055235      .WORD    T25WDE
      052154 012136      .WORD    PKTSSR
8715 052156      75$:  CKLOOP      ;LOOP IF SELECTED      TRAP      C#CLP1
      052156 104406
8716 052160      120$:
8717 052160 012703 000400      MOV      #256.,R3     ;RECORD SIZE
8718 052164 013737 003114 055122      MOV      FREE,T25RB   ;STARTING READ BUFFER ADDRESS
8719
8720      ;*****
8721      ;
8722      ;READ DATA,ACK,CVC=1 COMMAND
8723      ;
8724      ;*****
8725
8726 052172 012737 140001 055120      MOV      #140001,T25PK3 ;READ DATA,ACK,CVC=1 COMMAND
8727 052200 012704 055120      165$:  MOV      #T25PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
8728 052204 010337 055126      MOV      R3,T25SZ     ;SET UP RECORD SIZE IN PACKET
8729 052210 010465 000000      MOV      R4,TSDB(R5)  ;ISSUE COMMAND
8730 052214 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
8731 052220 016501 000002      MOV      TSSR(R5),R1  ;GET TSSR CONTENTS
8732 052224 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
8733 052230 020102      CMP      R1,R2        ;ARE THEY EQUAL
8734 052232 001406      BEQ      170$         ;BR, IF OK
8735 052234 005237 002212      INC      FATFLG       ;BUMP COUNT
8739 052240      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      052240 104456      TRAP      C#ERHRD
      052242 001026      .WORD    534
      052244 005204      .WORD    RDERR
      052246 012136      .WORD    PKTSSR
8740 052250      170$:  CKLOOP      ;LOOP IF SELECTED      TRAP      C#CLP1
      052250 104406
8741 052252 017701 130636      MOV      @FREE,R1     ;GET FIRST WORD FROM BUFFER
8742 052256 012702 000000      MOV      #0,R2        ;SET UP EXPECTED
8743 052262 020102      CMP      R1,R2        ;WAS RECORD NUMBERED 1
8744 052264 001406      BEQ      200$         ;BR, IF CORRECT RECORD
8745 052266 005237 002212      INC      FATFLG       ;BUMP COUNT

```



## TEST 5: SPACE RECORDS

```

8789 052426 016501 000002      MOV      TSSR(R5),R1      ;CONTENTS OF TSSR REGISTER
8790 052432 005237 002212      INC      FATFLG          ;BUMP COUNT
8794 052436      ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
      052436 104455      TRAP      C#ERDF
      052440 001030      .WORD    536
      052442 003650      .WORD    SFIERR
      052444 012124      .WORD    SFIMSG
8795 052446      20#:      MOV      UNITN,T25DSW      ;SET UP UNIT NUMBER
8796 052446 013737 002172 055010      MOV      @T25PACKET,R4    ;SUBROUTINE NEEDS PACKET ADDRESS
8797 052454 012704 054770
8798
8799      ;*****
8800      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8801      ;
8802      ;*****
8803
8804      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
8805 052460 004737 010752      BCS      25#              ;BR, IF COMMAND ISSUED OK
8806 052464 103407      INC      FATFLG          ;BUMP COUNT
8807 052466 005237 002212      MOV      R0,R1           ;SAVE CONTENTS OF TSSR
8811 052472 010001      ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
8812 052474      TRAP      C#ERHRD
      052474 104456      .WORD    537
      052476 001031      .WORD    WRTMSG
      052500 005054      .WORD    SFIMSG
      052502 012124
8813 052504      25#:      CKLOOP          ;LOOP IF SELECTED
      052504 104406      TRAP      C#CLP1
8814
8815      ;*****
8816      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8817      ;
8818      ;*****
8819
8820      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
8821 052506 004737 011104      BCS      30#              ;BR, IF NO PROBLEM
8822 052512 103407      MOV      R0,R1           ;SAVE TSSR
8823 052514 010001      INC      FATFLG          ;BUMP COUNT
8824 052516 005237 002212      ERRHRD  ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
8828 052522      TRAP      C#ERHRD
      052522 104456      .WORD    538
      052524 001032      .WORD    T25RWN
      052526 056125      .WORD    PKTSSR
      052530 012136
8829 052532      30#:      CKLOOP          ;LOOP IF SELECTED
      052532 104406      TRAP      L#CLP1
8830 052534 013701 055146      MOV      T25CN2,R1      ;NUMBER OF RECORDS ON TAPE
8831 052540 012702 177776      MOV      @65534..R2     ;MAX IT CAN SPACE OVER
8832 052544 020201      CMP      R2,R1           ;WHICH VALUE CAN WE USE
8833 052546 003002      BGT      46#              ;BR, IF # WRITTEN > 64K
8834 052550 010103      MOV      R1,R3           ;# WRITTEN CAN BE USED
8835 052552 000401      BR       47#              ;MOVE ON
8836 052554 010203      46#:      MOV      R2,R3      ;USE MAX NUMBER
8837 052556 162703 000001      47#:      SUB      @1,R3      ;DON'T GO ALL THE WAY YET
8838 052562 010337 055122      MOV      R3,T25R8      ;NUMBER OF RECORDS TO SPACE OVER
8839
8840      ;*****

```



TEST 5: SPACE RECORDS

```

8841
8842      ;SPACE FORWARD,ACK,CVC=1 COMMAND
8843      ;
8844      ;*****
8845
8846 052566 012737 140010 055120      MOV      @140010,T25PK3      ;SPACE FORWARD,ACK,CVC=1 COMMAND
8847 052574 012704 055120      MOV      @T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
8848 052600
8849 052600 013737 055146 055152 65#:  MOV      T25CN2,T25DLY      ;NUMBER OF RECORDS USED AS DELAY COUNTER
8850 052606 010465 000000      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
8851 052612 004737 016340 67#:  JSR      PC,WAITF          ;WAIT FOR SSR TO SET
8852 052616 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
8853 052622 012702 000200      MOV      @SSR,R2          ;SET UP EXPECTED
8854 052626 020102      CMP      R1,R2            ;ARE THEY EQUAL
8855 052630 001425      BEQ      75#              ;BR, IF OK
8856 052632      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
8857 052662 005337 055152      DEC      T25DLY           ;BUMP DOWN COUNTER
8858 052666 001351      BNE      67#              ;BR, IF NOT AT END OF DELAY
8859 052670 005237 002212      INC      FATFLG           ;BUMP COUNT
8863 052674      ERRHRD  ERRNO,T25WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      TRAP    C#ERRRD
      .WORD    539
      .WORD    T25WDE
      .WORD    PKTSSR
8864 052704 104456 75#:  CKLOOP                    ;LOOP IF SELECTED
      TRAP    C#CLP1
8865 052706 012703 010000      MOV      @4096,R3         ;RECORD SIZE
8866 052712 013737 003114 055122      MOV      FREE,T25RB       ;STARTING READ BUFFER ADDRESS
8867
8868      ;*****
8869      ;READ DATA,ACK COMMAND
8870      ;
8871      ;*****
8872
8873
8874 052720 012737 100001 055120 165#: MOV      @100001,T25PK3     ;READ DATA,ACK COMMAND
8875 052726 012704 055120      MOV      @T25PK3,R4       ;SET UP R4 WITH PACKET ADDRESS
8876 052732 010337 055126      MOV      R3,T25SZ         ;SET UP RECORD SIZE IN PACKET
8877 052736 010465 000000      MOV      R4,TSD8(R5)     ;ISSUE COMMAND
8878 052742 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
8879 052746 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
8880 052752 012702 000200      MOV      @SSR,R2         ;SET UP EXPECTED
8881 052756 020102      CMP      R1,R2           ;ARE THEY EQUAL
8882 052760 001411      BEQ      170#            ;BR, IF OK
8883 052762 032701 000004      B.L     @212,R1          ;CHECK FOR TAPE STATUS ALERT
8884 052766 001006      BNE      170#            ;IF SET ALL IS WELL
8885 052770 005237 002212      INC      FATFLG           ;BUMP COUNT
8889 052774      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      TRAP    C#ERRRD
      .WORD    539
      .WORD    T25WDE
      .WORD    PKTSSR
052774 104456

```



TEST 5: SPACE RECORDS

```

053126 000000
053130 013727 002116
053134 000000
053136 005367 177772
053142 001375
053144 005367 177756
053150 001367
8936 053152 005337 055152
8937 053156 001356
8938 053160 016501 000002
8939 053164 005237 002212
8943 053170
053170 104455
053172 001036
053174 003650
053176 012124
8944 053200 013737 002172 055010 20: MOV UNITN,T25DSW ;SET UP UNIT NUMBER
8945
8946 053206 C12704 054770 MOV @T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
8947
8948 ;*****
8949 ;
8950 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
8951 ;
8952 ;*****
8953
8954 053212 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
8955 053216 103407 BCS 25: ;BR, IF COMMAND ISSUED OK
8956 053220 005237 002212 INC FATFLG ;BUMP COUNT
8960 053224 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
8961 053226 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
053226 104456 TRAP C+ERHRD
053230 001037 .WORD 543
053232 005054 .WORD WRTMSG
053234 012124 .WORD SFMSG
8962 053236 25: CKLOOP ;LOOP IF SELECTED
053236 104406 TRAP C+CLP1
8963
8964 ;*****
8965 ;
8966 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8967 ;
8968 ;*****
8969
8970 053240 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
8971 053244 103407 BCS 30: ;BR, IF NO PROBLEM
8972 053246 010001 MOV RO,R1 ;SAVE TSSR
8973 053250 005237 002212 INC FATFLG ;BUMP COUNT
8977 053254 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
053254 104456 TRAP C+ERHRD
053256 001040 .WORD 544
053260 056125 .WORD T25RWN
053262 012136 .WORD PKTSSR
8978 053264 30: CKLOOP ;LOOP IF SELECTED
053264 104406 TRAP C+CLP1
8979
8980 ;*****

```



TEST 5: SPACE RECORDS

```

053460 001042 .WORD 546
053462 055235 .WORD T25WDE
053464 015564 .WORD EXPREC
9030 053466 104406 50$: CKLOOP TRAP C#CLP1
053466 104406
9031 053470 013701 055146 MOV T25CN2,R1 ;NUMBER OF RECORDS ON TAPE
9032 053474 012702 177776 MOV #65534.,R2 ;MAX IT CAN SPACE OVER
9033 053500 020201 CMP R2,R1 ;WHICH VALUE CAN WE USE
9034 053502 003002 BGT 55$ ;BR, IF # WRITTEN > 64K
9035 053504 010103 MOV R1,R3 ;# WRITTEN CAN BE USED
9036 053506 000401 BR 60$ ;MOVE ON
9037 053510 010203 55$: MOV R2,R3 ;USE MAX NUMBER
9038 053512 162703 000001 60$: SUB #1,R3 ;DON'T GO ALL THE WAY YET
9039 053516 010337 055122 MOV R3,T25RB ;NUMBER OF RECORDS TO SPACE OVER
9040
9041 ;*****
9042 ;
9043 ;SPACE REVERSE,ACK,CVC=1 COMMAND
9044 ;
9045 ;*****
9046
9047 053522 012737 140410 055120 MOV #140410,T25PK3 ;SPACE REVERSE,ACK,CVC=1 COMMAND
9048 053530 012704 055120 MOV #T25PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
9049 053534 010465 000000 MOV R4,T5DB(R5) ;ISSUE COMMAND
9050 053540 013737 055146 055152 MOV T25CN2,T25DLY ;SET UP COUNTER
9051 053546 004737 016340 70$: JSR PC,WAITF ;WAIT FOR SSR TO SET
9052 053552 016501 000002 MOV T5SR(R5),R1 ;GET T5SR CONTENTS
9053 053556 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
9054 053562 020102 CMP R1,R2 ;ARE THEY EQUAL
9055 053564 001425 BEQ 75$ ;BR, IF OK
9056 053566 DELAY 250 ;WAIT ABOUT .25 SECONDS
053566 012727 000250 MOV #250,(PC)+
053572 000000 .WORD 0
053574 013727 002116 MOV L#DLY,(PC)+
053600 000000 .WORD 0
053602 005367 177772 DEC -6(PC)
053606 001375 BNE .-4
053610 005367 177756 DEC -22(PC)
053614 001367 BNE .-20
9057 053616 005337 055152 DEC T25DLY ;BUMP COUNTER DOWN
9058 053622 001351 BNE 70$ ;BR, IF COUNTER HASN'T EXPIRED
9059 053624 005237 002212 INC FATFLG ;BUMP COUNT
9063 053630 ERHRD ERRNO,T25WDE,EXPREC ;T5SR INCORRECT AFTER READ DATA
053630 104456 TRAP C#ERHRD
053632 001043 .WORD 547
053634 055235 .WORD T25WDE
053636 015564 .WORD EXPREC
9064 053640 75$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
053640 104406
9065 053642 012703 010000 MOV #4096.,R3 ;RECORD SIZE
9066 053646 013737 003114 055122 MOV FREE,T25RB ;STARTING READ BUFFER ADDRESS
9067
9068 ;*****
9069 ;
9070 ;READ DATA,ACK COMMAND
9071 ;
9072 ;*****

```





TEST 5: SPACE RECORDS

```

9174
9175 ;*****
9176 ;
9177 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9178 ;
9179 ;*****
9180
9181 054152 013701 055020      MOV      T25BFR+6,R1      ;PICK UP XSTO
9182 054156 010102      MOV      R1,R2          ;SET UP EXPECTED
9183 054160 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
9184 054164 020102      CMP      R1,R2          ;DOES EXP = REC'D
9185 054166 001406      BEQ      40$           ;BR, IF EQUAL (OK)
9186 054170 005237 002212      INC      FATFLG        ;BUMP COUNT
9187 054174      ERRHRD  ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
9188      054174 104456      TRAP     C#ERHRD
9189      054176 001051      .WORD   553
9190      054200 055315      .WORD   T25BOT
9191      054202 015564      .WORD   EXPREC
9191 054204      40$:  CKLOOP          ;LOOP IF SELECTED
9192 054206 012737 000001 055122      MOV      #1,T25RB      ;NUMBER OF RECORDS TO SPACE OVER
9193
9194 ;*****
9195 ;
9196 ;SPACE REVERSE,ACK COMMAND
9197 ;
9198 ;*****
9199
9200 054214 012737 100410 055120      MOV      #100410,T25PK3 ;SPACE REVERSE,ACK COMMAND
9201 054222 012704 055120      MOV      #T25PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
9202 054226
9203 054226 010465 000000      65$:  MOV      R4,T5DB(R5) ;ISSUE COMMAND
9204 054232 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9205 054236 016501 000002      MOV      T5SR(R5),R1   ;GET T5SR CONTENTS
9206 054242 012702 100206      MOV      #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
9207 054246 020102      CMP      R1,R2          ;ARE THEY EQUAL
9208 054250 001406      BEQ      75$           ;BR, IF OK
9209 054252 005237 002212      INC      FATFLG        ;BUMP COUNT
9210 054256      ERRHRD  ERRNO,T25WDE,PKTSSR ;T5SR INCORRECT AFTER READ DATA
9211      054256 104456      TRAP     C#ERHRD
9212      054260 001052      .WORD   554
9213      054262 055235      .WORD   T25WDE
9214      054264 012136      .WORD   PKTSSR
9214 054266      75$:  CKLOOP          ;LOOP IF SELECTED
9215      054266 104406      TRAP     C#CLP1
9216
9217 ;*****
9218 ;
9219 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9220 ;
9221 ;*****
9222 054270 013701 055020      MOV      T25BFR+6,R1   ;GET XSTO STATUS WORD
9223 054274 010102      MOV      R1,R2          ;SET UP EXPECTED
9224 054276 052702 002000      BIS      #BIT10,R2     ;SET THE NEF BIT
9225 054302 020102      CMP      R1,R2          ;ARE THEY EQUAL
9226 054304 001406      BEQ      170$          ;BR, IF EQUAL (GOOD)

```



TEST 5: SPACE RECORDS

```

9227 054306 005237 002212      INC      FATFLG      ;BUMP COUNT
9231 054312                ERRHRD  ERRNO,T25NEF,EXPREC ;NEF SHOULD BE SET
      054312 104456                TRAP      C#ERHRD
      054314 001053                .WORD    555
      054316 055763                .WORD    T25NEF
      054320 015564                .WORD    EXPREC
9232 054322 170#: CKLOOP                TRAP      C#CLP1
      054322 104406                L10100:
9233 054324                ENDSUB                TRAP      C#ESUB
      054324 104403                ;IS ERROR COUNT AT 25
9234 054326 023727 002212 000017     CMP      FATFLG,#15. ;BR, IF LESS THAN 25
9235 054334 103402                BLO     999#         ;TRY TO DROP THE UNIT
9236 054336 004737 017272                JSR     PC,CKDROP
9237 054342 999#:
9238
9239
9240
9241
9242
9243
9244
9245
9246
9247
9248
9249
9250
9251
9252 054342                ;*
      054342                ;TEST 5, SUBTEST 8
      054342 104402                ;VERIFIES THAT A SPACE RECORDS REVERSE COMMAND THAT
      054344 004737 056336          JSR     PC,T25REST ;CAUSES THE TAPE TO RUN INTO BOT (WITH THE TAPE NOT
9254 054350 004737 056430          JSR     PC,T25RT2 ;INITIALLY AT BOT) CAUSES A TAPE STATUS ALERT
9255 054354 004737 056472          JSR     PC,T25RT3 ;TERMINATION AND SETS THE REVERSE INTO BOT (RIB)
9256
9257
9258
9259
9260
9261
9262
9263 054360 004737 016064          JSR     PC,SOFINIT ;STATUS BIT
9264 054364 103407                BCS     20#         ;*****
9265 054366 005237 002212          INC     FATFLG      ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
9269 054372 010001                MOV     R0,R1      ;*****
9270 054374                ERRDF   ERRNO,SFIERR,SFIMSG ;DO INITIALIZE ON CONTROLLER
      054374 104455                .WORD    556         ;BR IF INIT WAS OK
      054376 001054                .WORD    SFIMSG    ;BUMP COUNT
      054400 003650                .WORD    T25NEF    ;CONTENTS OF TSSR REGISTER
      054402 012124                .WORD    T25RT2    ;FATAL ERROR TSSR WAS NOT OK
9271 054404 013737 002172 055010 20#: MOV     UNITN,T25DSW ;SET UP UNIT NUMBER
9272
9273 054412 012704 054770          MOV     #T25PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
9274
9275
9276

```

TEST 5: SPACE RECORDS

```

9277 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
9278 ;
9279 ;*****
9280
9281 054416 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
9282 054422 103407 BCS 25$ ;BR, IF COMMAND ISSUED OK
9283 054424 005237 002212 INC FATFLG ;BUMP COUNT
9287 054430 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
9288 054432 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
054432 104456 TRAP C#ERHRD
054434 001055 .WORD 557
054436 005054 .WORD WRTMSG
054440 012124 .WORD SFMSG
9289 054442 25$: CKLOOP ;LOOP IF SELECTED
054442 104406 TRAP C#CLP1
9290
9291 ;*****
9292 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9293 ;
9294 ;*****
9295
9296
9297 054444 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9298 054450 103407 BCS 30$ ;BR, IF NO PROBLEM
9299 054452 010001 MOV RO,R1 ;SAVE TSSR
9300 054454 005237 002212 INC FATFLG ;BUMP COUNT
9304 054460 ERRHRD ERRNO,T25RWN,PKTSSR ;REWIND NOT ACCEPTED
054460 104456 TRAP C#ERHRD
054462 001056 .WORD 558
054464 056125 .WORD T25RWN
054466 012136 .WORD PKTSSR
9305 054470 30$: CKLOOP ;LOOP IF SELECTED
054470 104406 TRAP C#CLP1
9306
9307 ;*****
9308 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9309 ;
9310 ;*****
9311
9312
9313 054472 013701 055020 MOV T25BFR+6,R1 ;PICK UP XSTO
9314 054476 010102 MOV R1,R2 ;SET UP EXPECTED
9315 054500 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
9316 054504 020102 CMP R1,R2 ;DOES EXP = REC'D
9317 054506 001406 BEQ 40$ ;BR, IF EQUAL (OK)
9318 054510 005237 002212 INC FATFLG ;BUMP COUNT
9322 054514 ERRHRD ERRNO,T25BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
054514 104456 TRAP C#ERHRD
054516 001057 .WORD 559
054520 055315 .WORD T25BOT
054522 015564 .WORD EXPREC
9323 054524 40$: CKLOOP
054524 104406 TRAP C#CLP1
9324 054526 012737 000001 055122 MOV #1,T25RB ;NUMBER OF RECORDS TO SPACE OVER
9325
9326 ;*****
9327 ;

```

TEST 5: SPACE RECORDS

```

9328 ;SPACE FORWARD,IE,ACK,CVC=1 COMMAND
9329 ;
9330 ;*****
9331
9332 054534 012737 140210 055120      MOV      #140210,T25PK3      ;SPACE FORWARD,IE,ACK,CVC=1 COMMAND
9333 054542 012704 055120              MOV      #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
9334 054546 010465 000000              MOV      R4,TSDB(R5)       ;ISSUE COMMAND
9335 054552 004737 016340              JSR      PC,WAITF          ;WAIT FOR SSR TO SET
9336 054556 016501 000002              MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9337 054562 012702 000200              MOV      #SSR,R2          ;SET UP EXPECTED
9338 054566 020102                      CMP      R1,R2             ;ARE THEY EQUAL
9339 054570 001406                      BEQ      75#               ;BR, IF OK
9340 054572 005237 002212              INC      FATFLG            ;BUMP COUNT
9344 054576                      ERRHRD  ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERHRD
                                .WORD    560
                                .WORD    T25WDE
                                .WORD    EXPREC
                                75# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
9345 054606 104406                      MOV      #20,T25RB        ;NUMBER OF RECORDS TO SPACE OVER
9346 054610 012737 000020 055122
9347 ;*****
9348 ;
9349 ;SPACE REVERSE,IE,ACK COMMAND
9350 ;
9351 ;*****
9352
9353
9354 054616 012737 100610 055120      MOV      #100610,T25PK3    ;SPACE REVERSE,IE,ACK COMMAND
9355 054624 012704 055120              MOV      #T25PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
9356 054630 010465 000000              MOV      R4,TSDB(R5)       ;ISSUE COMMAND
9357 054634 004737 016340              JSR      PC,WAITF          ;WAIT FOR SSR TO SET
9358 054640 016501 000002              MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
9359 054644 012702 100204              MOV      #SSR!BIT2!SC,R2  ;SET UP EXPECTED
9360 054650 020102                      CMP      R1,R2             ;ARE THEY EQUAL
9361 054652 001406                      BEQ      175#              ;BR, IF OK
9362 054654 005237 002212              INC      FATFLG            ;BUMP COUNT
9366 054660                      ERRHRD  ERRNO,T25WDE,EXPREC ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERHRD
                                .WORD    561
                                .WORD    T25WDE
                                .WORD    EXPREC
                                175# :  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
9367 054670 104406                      MOV      T25BFR+14,R1     ;GET XST3 STATUS WORD
9368 054672 013701 055026              MOV      R1,R2             ;SET UP EXPECTED
9369 054676 010102                      BIS      #BIT0,R2         ;SET THE RIB BIT
9370 054700 052702 000001              CMP      R1,R2             ;ARE THEY EQUAL
9371 054704 020102                      BEQ      180#              ;BR, IF EQUAL (GOOD)
9372 054706 001406                      INC      FATFLG            ;BUMP COUNT
9373 054710 005237 002212              ERRHRD  ERRNO,T25NEF,EXPREC ;NEF SHOULD BE SET
9377 054714                      TRAP      C#ERHRD
                                .WORD    562
                                .WORD    T25NEF
                                .WORD    EXPREC
                                180# :  CKLOOP
9378 054724 104406                      TRAP      C#CLP1

```

TEST 5: SPACE RECORDS

```
9379 054726          ENDSUB           ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
      054726          L10101:           TRAP      C#ESUB
      054726 104403           ;IS ERROR COUNT AT 25
9380 054730 023727 002212 000017    CMP      FATFLG,#15.           ;BR, IF LESS THAN 25
9381 054736 103402           JSR      PC,CKDROP           ;TRY TO DROP THE UNIT
9382 054740 004737 017272           999#:
9383 054744           ;
9384           ;
9385           ;
9386           ;
9387 054744 004737 016546           JSR      PC,TSTLOOP           ;DO WE NEED TO ITERATE TEST
9388 054750 103002           BCC     193#                ;BR, IF NO LOOP REQUIRED
9389 054752 000137 047626           JMP      T25LOOP            ;EXECUTE AGAIN
9390 054756           193#:
9391 054756           EXIT      TST                ;ALL DONE THIS TEST
      054756 104432           TRAP      C#EXIT
      054760 001542           .WORD    L10071-.
9392
9393           ;*
9394           ;LOCAL STORAGE FOR THIS TEST
9395           ;-
9397 054762           .BLKB   10-<.-TSV2&7>
9399 054770           T25PACKET:
9400 054770 100004           .WORD   100004           ;COMMAND PACKET FOR TEST
9401 054772 055000           .WORD   T25DATA         ;WRITE CHARACTERISTICS COMMAND, WITH ACK
9402 054774 000000           .WORD   0                ;ADDRESS OF CHARACTERISTICS BLOCK
9403 054776 000010           .WORD   8.              ;STARTING VALUE OF BLOCK SIZE
9404 055000           T25DATA:                 ;CHARACTERISTICS DATA BLOCK
9405 055000           .WORD   T25BFR           ;ADDRESS OF MESSAGE BUFFER
9406 055002 000000           .WORD   0
9407 055004 000012           .WORD   10.             ;LENGTH OF MESSAGE BUFFER
9408 055006 000000           .WORD   0
9409 055010 000000           T25DSW: .WORD   0        ;SELECT DRIVE ZERO
9410 055012           T25BFR: .BLKW   25.     ;MESSAGE BUFFER
9411
9412           ;WRITE SUBSYSTEM MEMORY COMMAND PACKET
9413           ;
9415 055074           .BLKB   10-<.-TSV2&7>
9417 055100           T25PK2:
9418 055100 100006           .WORD   100006           ;WRITE SUB SYS MEM COMMAND, AND ACK
9419 055102 055130           .WORD   T25BF2          ;ADDRESS OF SELECT BLOCK DATA
9420 055104 000000           .WORD   0
9421 055106 000006           .WORD   6.              ;SIZE OF DATA PACKET
9422
9424 055110           .BLKB   10-<.-TSV2&7>
9426 055120           T25PK3:
9427 055120 100005           .WORD   100005           ;READ COMMAND, AND ACK
9428 055122           T25RB:
9429 055122 003114           T25WB: .WORD   FREE       ;ADDRESS OF WRITE BUFFER
9430 055124 000000           .WORD   0
9431 055126 000000           T25SZ: .WORD   0        ;SIZE OF BUFFER (EXTENT)
9432           .EVEN
9433           ;
9434           ;
9435           ;
9436 055130           T25BF2:
9437 055130 010           T25BS0: .BYTE   10       ;BSELO AREA
```

## TEST 5: SPACE RECORDS

```

9438 055131      200          T25BS1: .BYTE   200          ;BSEL1 AREA
9439 055132 000000          T25S2:  .WORD    0          ;SEL 2 AREA
9440 055134 000000          T25S3:  .WORD    0          ;DATA AREA
9441
9442
9443
9444
9445
9446 055136 100005          T25RN:  .WORD  100005        ;READ DATA (NEXT)
9447 055140 100405          T25WDR: .WORD  100405        ;READ DATA RETRY
9448 055142 102005          T25CON: .WORD  102005        ;WRITE CONTINOUS
9449 055144 177777          .WORD  177777        ;END OF DATA
9450
9451 055146 000000          T25CN2: .WORD    0          ;COUNTER FOR RECORDS
9452 055150 000000          T25CNT: .WORD    0          ;COUNTER FOR RECORDS
9453 055152 000000          T25DLY: .WORD    0          ;COUNTER FOR RECORDS
9454
9455
9456
9457
9458
9459 055154      127      122      111  T25SSR: .ASCIZ  'WRITE SUBSYSTEM Miscellaneous Read Status Failed'
9460 055235      124      123      123  T25WDE: .ASCIZ  'TSSR Not Correct After POSITION (SPACE) Command'
9461 055315      124      141      160  T25BOT: .ASCIZ  'Tape Not At BOT After REWIND Command'
9462 055362      124      123      123  T25TM:  .ASCIZ  'TSSR Not Correct After POSITION (Space Command) Reject'
9463 055451      127      162      151  T25NET: .ASCIZ  'Write Tape, Status Alert, But No EOT Sensed'
9464 055525      123      160      141  T25WNG: .ASCIZ  'Space Forward Failed To Position On Correct Record'
9465 055610      123      160      141  T25BNC: .ASCIZ  'Space Forward, From BOT, Failed To Clear BOT Indication'
9466 055700      123      160      141  T25WNH: .ASCIZ  'Space Reverse Failed To Position On Correct Record'
9467 055763      123      160      141  T25NEF: .ASCIZ  'Space Reverse, At BOT, Failed To Set NEF (XST0)'
9468 056043      123      160      141  T25RIB: .ASCIZ  'Space Reverse, Into BOT, Failed To Set RIB (XST3)'
9469 056125      122      145      167  T25RWN: .ASCIZ  'Rewind (POSITION) Command Not Accepted'
9470 056174      104      162      151  T25OFL: .ASCIZ  'Drive 7 Select Failed To Set "OFL" In TSSR'
9471 056247      124      123      123  T25WDC: .ASCIZ  'TSSR Not Correct After READ DATA Command'
9472 056320      123      160      141  TST25ID: .ASCIZ  'Space Records'
9473
9474
9475
9476
9477
9478
9479
9480
9481 056336          T25REST:
9482 056336          SAVREG
9483 056342 012701 054770          MOV      @T25PACKET,R1          ;SAVE THE REGISTERS
9484 056346 012721 100004          MOV      @100004,(R1)+         ;START OF THE PACKET
9485 056352 012721 055000          MOV      @T25DATA,(R1)+       ;WRITE SUBSYSTEM MEM. WITH ACK
9486 056356 005021          CLR      (R1)+                 ;ADDRESS OF CHARAISTICS DATA BLOCK
9487 056360 012721 000012          MOV      @10.,(R1)+           ;EXTENDED ADDRESS
9488 056364 012721 055012          MOV      @T25BFR,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
9489 056370 005021          CLR      (R1)+                 ;ADDRESS OF MESSAGE BUFFER
9490 056372 012721 000024          MOV      @20.,(R1)+           ;LENGTH OF MESSAGE BUFFER
9491 056376 005021          CLR      (R1)+
9492 056400 012711 000000          MOV      @0,(R1)              ;SELECT DRIVE ZERO
9493 056404 012702 000030          MOV      @24.,R2              ;NUMBER OF LOCATIONS TO BE CLEARED
9494 056410 012762 177777 055012 64+: MOV      @177777,T25BFR(R2)     ;ALL ONES TO MESSAGE BUFFER

```

TEST 5: SPACE RECORDS

```

9495 056416 005742
9496 056420 022702 000000
9497 056424 001371
9498 056426 000207
9499
9500 056430
9501 056430
9502 056434 012701 055100
9503 056440 012721 100006
9504 056444 012721 055130
9505 056450 005021
9506 056452 012721 000006
9507 056456 005021
9508 056460 012701 055130
9509 056464 005021
9510 056466 005011
9511 056470 000207
9512 056472
9513 056472
9514 056476 012701 055120
9515 056502 012721 000000
9516 056506 012721 000000
9517 056512 005021
9518 056514 012721 000000
9519 056520 000207
9520 056522
    056522
    056522 104401
9521
9522
9523
9524
9525
9526
9527
9528
9529
9530
9531
9532
9533
9534
9535
9536
9537
9538
9539
9540 056524
    056524
9541 056524 012737 006446 002170
9542 056532 004737 017364
9547 056536 012700 075707
9548 056542 004737 016600
9549 056546 012737 000005 002206
9550 056554 004737 022206
9551 056560 005037 003126
9552 056564 005037 073156
    
```

```

TST      -(R2)           ;NEXT LOCATION
CMP      #0,R2          ;IS R2 AT ZERO YET
BNE      64$           ;KEEP GOING UNTIL DONE
RTS      PC             ;RETURN

T25RT2:
SAVREG                    ;SAVE THE REGISTERS
MOV      #T25PK2,R1      ;START OF THE PACKET
MOV      #100006,(R1)+   ;WRITE SUBSYSTEM MEM. WITH ACK.
MOV      #T25BF2,(R1)+  ;ADDRESS OF DATA BLOCK
CLR      (R1)+           ;EXTENDED ADDRESS
MOV      #6,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
CLR      (R1)+
MOV      #T25BF2,R1     ;POINT TO DATA SEL AREA
CLR      (R1)+
CLR      (R1)+
RTS      PC             ;RETURN

T25RT3:
SAVREG                    ;SAVE THE REGISTERS
MOV      #T25PK3,R1      ;START OF THE PACKET
MOV      #0,(R1)+       ;WRITE SUBSYSTEM MEM. WITH ACK.
MOV      #0,(R1)+       ;ADDRESS OF DATA BLOCK
CLR      (R1)+           ;EXTENDED ADDRESS
MOV      #0,(R1)+       ;SIZE OF DATA BLOCK IN BYTES
RTS      PC             ;RETURN

L10071: TRAP C#ETST
    
```

.SBTTL 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)
; CONRTOL ARE USED. ALSO TESTED ARE PROPER TERMINATIONS UN
; EXCEPTIONAL OR ERROR CONDITIONS: RECORD LENGTH LONG, RECORD
; LENGTH SHORT, READ REVERSE AT BOT, ILLEGAL DATA BUFFER ADDRESSES,
; AND DATA BUFFERS IN NONEXISTENT MEMORY.
;
;
; THE TEST CONSISTS OF THE FOLLOWING 15 SUBTESTS
;
;
; -
    
```

```

BGNTST
T6::
MOV      #EPRT2,EPRTSW   ;SECONDARY ERROR MESSAGE
JSR      PC,KTOFF        ;DON'T NEED KT11
MOV      #TST26ID,RO     ;ASCII MESSAGE TO IDENTIFY TEST
JSR      PC,TSTS'TUP     ;DO INITIAL TEST SETUP
MOV      #5,LOOPCNT      ;PERFORM 5 ITERATIONS
JSR      PC,MEMCK        ;CHECK FOR MEMORY
CLR      NXMFLG          ;SET FLAG
CLR      T26CNT          ;CLEAR TAPE RECORD COUNTER
    
```

TEST 6: REREADS

```

9553 ;*
9554 ;
9555 ;TEST 6, SUBTEST 1
9556 ;
9557 ;
9558 ;VERIFIES THAT THE REREAD PREVIOUS COMMAND WITH OPP=0
9559 ;AND SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST
9560 ;REWOUND AND THEN WRITTEN WITH A SERIES OF TEST
9561 ;RECORDS VARYING IN LENGTH AND DATA CONTENT. THE TAPE
9562 ;IS THEN REWOUND AGAIN. FOR EACH TEST RECORD, THE
9563 ;TAPE IS SPACED FORWARD ONE RECORD AND A REREAD
9564 ;PREVIOUS COMMAND ISSUED. RESULTS (STATUS, DATA,
9565 ;ETC.) ARE VERIFIED. THE BYTE COUNT ON EACH REREAD
9566 ;PREVIOUS COMMAND IS SET TO THE LENGTH OF THE EXPECTED
9567 ;RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.
9568 ;
9569 ;
9570 ;-
9571 ;
9572 056570 T26LOOP:
9573 ;
9574 056570          BGNSUB              ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
9575          056570          T6.1:
9576          056570 104402          TRAP      C#BSUB
9577 056572 004737 075720          JSR      PC,T26REST          ;SET COMMAND PACKET
9578 056576 004737 076012          JSR      PC,T26RT2          ;SET UP OTHER COMMAND PACKET
9579 056602 004737 076054          JSR      PC,T26RT3          ;SET UP OTHER COMMAND PACKET
9580 056606 012737 176750 073164    MOV      #65000.,T26DLY      ;SET UP DELAY COUNTER
9581 ;*****
9582 ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
9583 ;
9584 ;*****
9585 ;
9586 056614 004737 016064    10$:    JSR      PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
9587 056620 103426          BCS      20$          ;BR IF INIT WAS OK
9588 056622          DELAY  250          ;DELAY FOR A REWIND TO FINISH
9589          056622 012727 000250          MOV      #250.(PC)+
9590          056626 000000          .WORD  0
9591          056630 013727 002116          MOV      L#DLY.(PC)+
9592          056634 000000          .WORD  0
9593          056636 005367 177772          DEC      -6(PC)
9594          056642 001375          BNE      -4
9595          056644 005367 177756          DEC      -22(PC)
9596          056650 001367          BNE      -20
9597 056652 005337 073164          DEC      T26DLY          ;DEC COUNTER
9598 056656 001356          BNE      10$          ;BR, IF DELAY NOT READY
9599 056660 005237 002212          INC      FATFLG          ;BUMP COUNT
9600 056664 010001          MOV      R0,R1          ;CONTENTS OF TSSR REGISTER
9601 056666          ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
9602          056666          TRAP      C#ERDF
9603          056670 001131          .WORD  601
9604          056672 003650          .WORD  SFIERR
9605          056674 012124          .WORD  SFIMSG
9606 9597 056676          20$:    MOV      UNITN,T26DSW          ;SET UP UNIT NUMBER
9607 9598 056676 013737 002172 073020

```

TEST 6: REREADS

```

9599 056704 012704 073000      MOV      #T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
9600
9601      ;*****
9602      ;
9603      ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
9604      ;
9605      ;*****
9606
9607 056710 004737 010752      JSR      PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
9608 056714 103407              BCS     26$                ;BR, IF COMMAND ISSUED OK
9609 056716 005237 002212      INC     FATFLG             ;BUMP COUNT
9613 056722 010001              MOV     R0,R1              ;SAVE CONTENTS OF TSSR
9614 056724              ERRHRD  ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
          056724 104456              TRAP   C$ERHRD
          056726 001132              .WORD 602
          056730 005054              .WORD WRTMSG
          056732 012124              .WORD SFMSG
9615 056734              26$:   CKLOOP              ;LOOP IF SELECTED
          056734 104406              TRAP   C$CLP1
9616
9617      ;*****
9618      ;
9619      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9620      ;
9621      ;*****
9622
9623 056736 004737 011104      JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
9624 056742 103413              BCS     30$                ;BR, IF NO PROBLEM
9625 056744 016501 000002      MOV     TSSR(R5),R1        ;GET TSSR
9626 056750 012702 000200      MOV     #SSR,R2           ;SET UP EXPECTED TSSR
9627 056754 010004              MOV     R0,R4              ;PACKET ADDRESS SET UP
9628 056756 005237 002212      INC     FATFLG             ;BUMP COUNT
9632 056762              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
          056762 104456              TRAP   C$ERHRD
          056764 001133              .WORD 603
          056766 074464              .WORD T26RWN
          056770 012136              .WORD PKTSSR
9633 056772              30$:   CKLOOP              ;LOOP IF SELECTED
          056772 104406              TRAP   C$CLP1
9634
9635      ;*****
9636      ;
9637      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9638      ;
9639      ;*****
9640
9641 056774 013701 073030      MOV     T26BFR+6,R1        ;PICK UP XSTO
9642 057000 010102              MOV     R1,R2              ;SET UP EXPECTED
9643 057002 052702 000002      BIS     #BIT1,R2           ;SET BOT BIT IN EXPECTED
9644 057006 020102              CMP     R1,R2              ;DOES EXP = REC'D
9645 057010 001406              BEQ     40$                ;BR, IF EQUAL (OK)
9646 057012 005237 002212      INC     FATFLG             ;BUMP COUNT
9650 057016              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          057016 104456              TRAP   C$ERHRD
          057020 001134              .WORD 604
          057022 074175              .WORD T26BOT
          057024 015564              .WORD EXPREC

```



TEST 6: REREADS

```

9651 057026          40$:  CKLOOP                ;LOOP IF SELECTED
      057026 104406          ;RECORD SIZE TRAP C#CLP1
9652 057030 012703 000400  MOV #256.,R3
9653 057034 013737 003114 073132  MOV FREE,T26R8 ;STARTING WRITE BUFFER ADDRESS
9654
9655 ;*****
9656 ;
9657 ;WRITE DATA,ACK,CVC=1 COMMAND
9658 ;
9659 ;*****
9660
9661 057042 012737 140005 073130  MOV #140005,T26PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
9662 057050 012704 073130  MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
9663 057054          65$:
9664 057054 010300          MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
9665 057056 004737 017512  JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
9666 057062 010337 073136  MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
9667 057066 010465 000000  MOV R4,TSD8(R5) ;ISSUE COMMAND
9668 057072 004737 016340  JSR PC,WAITF ;WAIT FOR SSR TO SET
9669 057076 016501 000002  MOV TSSR(R5),R1 ;GET TSSR CONTENTS
9670 057102 012702 000200  MOV #SSR,R2 ;SET UP EXPECTED
9671 057106 020102          CMP R1,R2 ;ARE THEY EQUAL
9672 057110 001406          BEQ 75$ ;BR, IF OK
9673 057112 005237 002212  INC FATFLG ;BUMP COUNT
9677 057116          ERRHRD ERRNO,WRTErr,EXPREC ;TSSR INCORRECT AFTER WRITE DATA
      057116 104456          TRAP C#ERHRD
      057120 001135          .WORD 605
      057122 005111          .WORD WRTErr
      057124 015564          .WORD EXPREC
9678 057126          75$:  CKLOOP                ;LOOP IF SELECTED
      057126 104406          TRAP C#CLP1
9679 057130 005723          TST (R3)+ ;BUMP RECORD SIZE
9680 057132 022703 000414  CMP #268.,R3 ;END OF RECORD YET
9681 057136 001346          BNE 65$ ;BR, IF MORE RECORDS TO WRITE
9682 057140          80$:  CKLOOP                ;LOOP IF SELECTED
      057140 104406          TRAP C#CLP1
9683 057142          120$:
9684 ;*****
9685 ;
9686 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9687 ;
9688 ;*****
9689
9690
9691 057142 004737 011104  JSR PC,REWIND ;CALL TAPE REWIND COMMAND
9692 057146 103413          BCS 130$ ;BR, IF NO PROBLEM
9693 057150 016501 000002  MOV TSSR(R5),R1 ;GET TSSR
9694 057154 012702 000200  MOV #SSR,R2 ;SET UP EXPECTED TSSR
9695 057160 010004          MOV R0,R4 ;PACKET ADDRESS SET UP
9696 057162 005237 002212  INC FATFLG ;BUMP COUNT
9700 057166          ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      057166 104456          TRAP C#ERHRD
      057170 001136          .WORD 606
      057172 074464          .WORD T26RWN
      057174 012136          .WORD PKTSSR
9701 057176          130$:  CKLOOP                ;LOOP IF SELECTED
      057176 104406          TRAP C#CLP1
    
```

TEST 6: REREADS

```

9702
9703 ;*****
9704 ;
9705 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9706 ;
9707 ;*****
9708
9709 057200 013701 073030      MOV      T26BFR+6,R1      ;PICK UP XSTO
9710 057204 010102      MOV      R1,R2          ;SET UP EXPECTED
9711 057206 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
9712 057212 020102      CMP      R1,R2          ;DOES EXP = REC'D
9713 057214 001406      BEQ      140$           ;BR, IF EQUAL (OK)
9714 057216 005237 002212      INC      FATFLG         ;BUMP COUNT
9718 057222      ERRHRD  ERRNO,T26BOT,PKTSSR ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    607
                                .WORD    T26BOT
                                .WORD    PKTSSR
                                057222 104456
                                057224 001137
                                057226 074175
                                057230 012136
9719 057232      140$:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                057232 104406
9720 057234 012737 000400 073162      MOV      @256.,T26RSZ    ;SET RECORD SIZE
9721
9722 ;*****
9723 ;
9724 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9725 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9726 ;
9727 ;*****
9728
9729 057242 012703 000001      145$:  MOV      @1,R3      ;SPACE ONE RECORD PARAMETER
9730 057246 004737 010556      JSR      PC,SPACE        ;CALL SPACE ROUTINE
9731 057252 103412      BCS      150$           ;BR, IF NO PROBLEM WITH SPACE COMMAND
9732 057254 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR
9733 057260 012702 000200      MOV      @SSR,R2        ;SET UP EXPECTED TSSR
9734 057264 005237 002212      INC      FATFLG         ;BUMP COUNT
9738 057270      ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
                                TRAP      C$ERHRD
                                .WORD    608
                                .WORD    T26SC
                                .WORD    EXPREC
                                057270 104456
                                057272 001140
                                057274 073577
                                057276 015564
9739 057300      150$:  CKLOOP          ;RECORD SIZE
                                TRAP      C$CLP1
                                057300 104406
9740 057302 013703 073162      MOV      T26RSZ,R3      ;RECORD SIZE
9741 057306 013737 003114 073132      MOV      FREE,T26RB     ;STARTING READ BUFFER ADDRESS
9742
9743 ;*****
9744 ;
9745 ;REREREAD DATA,CVC=1,ACK COMMAND
9746 ;
9747 ;*****
9748
9749 057314 012737 141001 073130      165$:  MOV      @141001,T26PK3 ;REREREAD DATA,CVC=1,ACK COMMAND
9750 057322 012704 073130      MOV      @T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
9751 057326 010337 073136      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
9752 057332 010465 000000      MOV      R4,TSDB(R5)   ;ISSUE COMMAND
9753 057336 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9754 057342 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR CONTENTS

```





## TEST 6: REREADS

```

9860 057630          ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      057630 104456
      057632 001145
      057634 074464
      057636 012136
9861 057640          30$:  CKLOOP                      ;LOOP IF SELECTED
      057640 104406
9862
9863 ;*****
9864 ;
9865 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
9866 ;
9867 ;*****
9868
9869 057642 013701 073030      MOV      T26BFR+6,R1      ;PICK UP XSTO
9870 057646 010102          MOV      R1,R2           ;SET UP EXPECTED
9871 057650 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
9872 057654 020102          CMP      R1,R2           ;DOES EXP = REC'D
9873 057656 C01406          BEQ      40$           ;BR, IF EQUAL (OK)
9874 057660 005237 002212      INC      FATFLG         ;BUMP COUNT
9878 057664          ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      057664 104456
      057666 001146
      057670 074175
      057672 015564
9879 057674          40$:  CKLOOP                      ;LOOP IF SELECTED
      057674 104406
9880 057676 012703 000400      MOV      #256.,R3       ;RECORD SIZE
9881 057702 013737 003114 073132  MOV      FREE,T26RB     ;STARTING WRITE BUFFER ADDRESS
9882
9883 ;*****
9884 ;
9885 ;WRITE DATA,ACK,SWB COMMAND
9886 ;
9887 ;*****
9888
9889 057710 012737 110005 073130      MOV      #110005,T26PK3 ;WRITE DATA,ACK,SWB COMMAND
9890 057716 012704 073130          MOV      #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
9891 057722
9892 057722 010300          65$:  MOV      R3,R0     ;SET PATTERN IN CORRECT REGISTER
9893 057724 004737 017512      JSR      PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
9894 057730 010337 073136      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
9895 057734 010465 000000      MOV      R4,TSD8(R5)   ;ISSUE COMMAND
9896 057740 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
9897 057744 016501 000002      MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
9898 057750 012702 000200      MOV      #SSR,R2       ;SET UP EXPECTED
9899 057754 020102          CMP      R1,R2         ;ARE THEY EQUAL
9900 057756 001406          BEQ      75$           ;BR, IF OK
9901 057760 005237 002212      INC      FATFLG         ;BUMP COUNT
9905 057764          ERRHRD  ERRNO,WRTErr,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
      057764 104456
      057766 001147
      057770 005111
      057772 012136
9906 057774          75$:  CKLOOP                      ;LOOP IF SELECTED
      057774 104406
9907 057776 005723          TST      (R3)+         ;BUMP RECORD SIZE

```

## TEST 6: REREADS

```

9908 060000 022703 000414          CMP      #268.,R3          ;END OF RECORD YET
9909 060004 001346          BNE      65#              ;BR, IF MORE RECORDS TO WRITE
9910 060006 104406          80#:    CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
9911 060010 104406          120#:
9912
9913          ;*****
9914          ;
9915          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
9916          ;
9917          ;*****
9918
9919 060010 004737 011104          JSR      PC.REWIND        ;CALL TAPE REWIND COMMAND
9920 060014 103413          BCS      130#            ;BR, IF NO PROBLEM
9921 060016 016501 000002          MOV      TSSR(R5),R1     ;GET TSSR
9922 060022 012702 000200          MOV      #SSR,R2        ;SET UP EXPECTED TSSR
9923 060026 010004          MOV      R0,R4          ;PACKET ADDRESS SET UP
9924 060030 005237 002212          INC      FATFLG         ;BUMP COUNT
9928 060034          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    616
                                .WORD    T26RWN
                                .WORD    PKTSSR
                                TRAP      C#CLP1
9929 060044 104406          130#:    CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
9930
9931          ;*****
9932          ;
9933          ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
9934          ;
9935          ;*****
9936
9937 060046 013701 073030          MOV      T26BFR+6,R1     ;PICK UP XST0
9938 060052 010102          MOV      R1,R2          ;SET UP EXPECTED
9939 060054 052702 000002          BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
9940 060060 020102          CMP      R1,R2          ;DOES EXP = REC'D
9941 060062 001406          BEQ      140#            ;BR, IF EQUAL (OK)
9942 060064 005237 002212          INC      FATFLG         ;BUMP COUNT
9946 060070          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    617
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
9947 060100 104406          140#:    CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
9948 060102 012737 000400 073162          MOV      #256.,T26RSZ   ;SET UP RECORD SIZE
9949
9950          ;*****
9951          ;
9952          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
9953          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
9954          ;
9955          ;*****
9956
9957 060110 012703 000001          145#:    MOV      #1,R3    ;SPACE ONE RECORD PARAMETER
9958 060114 004737 010556          JSR      PC.SPACE       ;CALL SPACE ROUTINE
9959 060120 103412          BCS      150#            ;BR, IF NO PROBLEM WITH SPACE COMMAND

```

TEST 6: REREADS

```

9960 060122 016501 000002          MOV    TSSR(R5),R1          ;GET TSSR
9961 060126 012702 000200          MOV    #SSR,R2            ;SET UP EXPECTED TSSR
9962 060132 005237 002212          INC    FATFLG             ;BUMP COUNT
9966 060136          ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
          060136 104456          TRAP    C#ERHRD
          060140 001152          .WORD   618
          060142 073577          .WORD   T26SC
          060144 015564          .WORD   EXPREC
9967          060146          150#:  CKLOOP
          060146 104406          TRAP    C#CLP1
9968 060150 013703 073162          MOV    T26RSZ,R3          ;RECORD SIZE
9969 060154 013737 003114 073132  MOV    FREE,T26RB         ;STARTING READ BUFFER ADDRESS
9970
9971          ;*****
9972          ;
9973          ;REREAD DATA,CVC=1,ACK,SWB COMMAND
9974          ;
9975          ;*****
9976
9977 060162 012737 151001 073130  MOV    #151001,T26PK3     ;REREAD DATA,CVC=1,ACK,SWB COMMAND
9978 060170 012704 073130 165#:  MOV    #T26PK3,R4         ;SET UP R4 WITH PACKET ADDRESS
9979 060174 010337 073136          MOV    R3,T26SZ           ;SET UP RECORD SIZE IN PACKET
9980 060200 010465 000000          MOV    R4,TSDB(R5)        ;ISSUE COMMAND
9981 060204 004737 016340          JSR    PC,WAITF           ;WAIT FOR SSR TO SET
9982 060210 016501 000002          MOV    TSSR(R5),R1        ;GET TSSR CONTENTS
9983 060214 012702 000200          MOV    #SSR,R2            ;SET UP EXPECTED
9984 060220 020102          CMP    R1,R2              ;ARE THEY EQUAL
9985 060222 001406          BEQ    170#               ;BR, IF OK
9986 060224 005237 002212          INC    FATFLG             ;BUMP COUNT
9990 060230          ERRHRD  ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
          060230 104456          TRAP    C#ERHRD
          060232 001153          .WORD   619
          060234 075020          .WORD   T26WDC
          060236 012136          .WORD   PKTSSR
9991 060240          170#:  CKLOOP            ;LOOP IF SELECTED
          060240 104406          TRAP    C#CLP1
9992 060242 013702 003114          MOV    FREE,R2            ;CURRENT BUFFER ADDRESS TO R2
9993 060246 010304          MOV    R3,R4              ;CURRENT RECORD SIZE
9994 060250 162704 000400          SUB    #256.,R4           ;FIRST LOCATION IN BUFFER
9995 060254 060204 173#:  ADD    R2,R4              ;SET UP POINTER
9996 060256 021403          CMP    (R4),R3            ;CHECK DATA READ (R3=DATA ALSO)
9997 060260 001410          BEQ    180#               ;BR, IF ALL IS WELL
9998 060262 011401          MOV    (R4),R1            ;RECD DATA
9999 060264 010302          MOV    R3,R2              ;EXPECTED DATA
10000 060266 005237 002212          INC    FATFLG             ;BUMP COUNT
10004 060272          ERRHRD  ERRNO,T26DTA,EXPREC ;DATA READ NOT = WRITTEN
          060272 104456          TRAP    C#ERHRD
          060274 001154          .WORD   620
          060276 074242          .WORD   T26DTA
          060300 015564          .WORD   EXPREC
10005          060302          180#:  CKLOOP            ;LOOP IF SELECTED
          060302 104406          TRAP    C#CLP1
10006 060304 005724          TST    (R4),              ;BUMP TO NEXT LOCATION
10007 060306 160204          SUB    R2,R4              ;CORRECT RECORDS SIZE VALUE
10008 060310 020403          CMP    R4,R3              ;END OF RECORD YET
10009 060312 001360          BNE    173#               ;BR, IF NOT AT END OF RECORD
10010 060314 005723          TST    (R3),              ;BUMP RECORD SIZE

```





## TEST 6: REREADS

```

10063
10064
10065
10066 060372 004737 016064      JSR    PC,SOFINIT      ;DO INITIALIZE ON CONTROLLER
10067 060376 103407              BCS    20#             ;BR IF INIT WAS OK
10068 060400 005237 002212      INC    FATFLG          ;BUMP COUNT
10072 060404 010001              MOV    R0,R1           ;CONTENTS OF TSSR REGISTER
10073 060406              ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C#ERDF
                                .WORD   621
                                .WORD   SFIERR
                                .WORD   SFIMSG
10074 060416 013737 002172 073020 20#:  MOV    UNITN,T26DSW    ;SET UP UNIT NUMBER
10075
10076 060424 012704 073000      MOV    #T26PACKET,R4  ;SUBROUTINE NEEDS PACKET ADDRESS
10077
10078
10079
10080
10081
10082
10083
10084 060430 004737 010752      JSR    PC,WRTCHR       ;ISSUE WRITE CHARACTERISTICS
10085 060434 103407              BCS    26#             ;BR, IF COMMAND ISSUED OK
10086 060436 005237 002212      INC    FATFLG          ;BUMP COUNT
10090 060442 010001              MOV    R0,R1           ;SAVE CONTENTS OF TSSR
10091 060444              ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C#ERHRD
                                .WORD   622
                                .WORD   WRTMSG
                                .WORD   SFIMSG
10092 060454              26#:  CKLOOP           ;LOOP IF SELECTED
                                TRAP    C#CLP1
10093
10094
10095
10096
10097
10098
10099
10100 060456 004737 011104      JSR    PC,REWIND       ;CALL TAPE REWIND COMMAND
10101 060462 103413              BCS    30#             ;BR, IF NO PROBLEM
10102 060464 016501 000002      MOV    TSSR(R5),R1     ;GET TSSR
10103 060470 012702 000200      MOV    #SSR,R2         ;SET UP EXPECTED TSSR
10104 060474 010004              MOV    R0,R4           ;PACKET ADDRESS SET UP
10105 060476 005237 002212      INC    FATFLG          ;BUMP COUNT
10109 060502              ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C#ERHRD
                                .WORD   623
                                .WORD   T26RWN
                                .WORD   PKTSSR
10110 060512              30#:  CKLOOP           ;LOOP IF SELECTED
                                TRAP    C#CLP1
10111
10112
10113
10114
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)

```

TEST 6: REREADS

```

10115
10116
10117
10118 060514 013701 073030      MOV      T26BFR+6,R1      ;PICK UP XSTO
10119 060520 010102              MOV      R1,R2           ;SET UP EXPECTED
10120 060522 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
10121 060526 020102              CMP      R1,R2           ;DOES EXP = REC'D
10122 060530 001406              BEQ      40#             ;BR, IF EQUAL (OK)
10123 060532 005237 002212      INC      FATFLG          ;BUMP COUNT
10127 060536              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERRRD
                                .WORD    624
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
                                .WORD    624
                                .WORD    T26BOT
                                .WORD    EXPREC
10128 060546              40# :   CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
10129 060550 012703 000400      MOV      @256.,R3        ;RECORD SIZE
10130 060554 013737 003114 073132  MOV      FREE,T26RB      ;STARTING WRITE BUFFER ADDRESS
10131
10132
10133
10134
10135
10136
10137
10138 060562 012737 140005 073130      MOV      @140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10139 060570 012704 073130      MOV      @T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
10140 060574
10141 060574 010300      65# :   MOV      R3,R0        ;SET PATTERN IN CORRECT REGISTER
10142 060576 004737 017512      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
10143 060602 010337 073136      MOV      R3,T26SZ        ;SET UP RECORD SIZE IN PACKET
10144 060606 013777 073156 122300  MOV      T26CNT,@FREE    ;MOVE TAPE RECORD NUMBER TO BUFFER
10145 060614 062737 000001 073156  ADD      @1,T26CNT       ;NUMBER READY FOR NEXT RECORD
10146 060622 010465 000000      MOV      R4,TSD8(R5)    ;ISSUE COMMAND
10147 060626 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
10148 060632 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
10149 060636 012702 000200      MOV      @SSR,R2        ;SET UP EXPECTED
10150 060642 020102              CMP      R1,R2           ;ARE THEY EQUAL
10151 060644 001406              BEQ      75#             ;BR, IF OK
10152 060646 005237 002212      INC      FATFLG          ;BUMP COUNT
10156 060652              ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERRRD
                                .WORD    625
                                .WORD    WRERR
                                .WORD    PKTSSR
                                TRAP      C#CLP1
                                .WORD    625
                                .WORD    WRERR
                                .WORD    PKTSSR
10157 060662              75# :   CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
10158 060664 104406              TST      (R3).           ;BUMP THE RECORD SIZE
10159 060666 005723 000414      CMP      @268.,R3       ;MAXIMUM SIZE YET
10160 060672 001401              BEQ      120#           ;BR, IF AT END OF WRITE SEQUENCE
10161 060674 000737              BR       65#            ;WRITE MORE RECORDS
10162 060676
10163 060676 005037 073156      120# :   CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
10164
10165
10166
10167
;*****
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE

```

TEST 6: REREADS

```

10168
10169
10170
10171 060702 004737 011104
10172 060706 103413
10173 060710 016501 000002
10174 060714 012702 000200
10175 060720 010004
10176 060722 005237 002212
10180 060726
      060726 104456
      060730 001162
      060732 074464
      060734 012136
10181 060736
      060736 104406
10182
10183
10184
10185
10186
10187
10188
10189 060740 013701 073030
10190 060744 010102
10191 060746 052702 000002
10192 060752 020102
10193 060754 001406
10194 060756 005237 002212
10198 060762
      060762 104456
      060764 001163
      060766 074175
      060770 015564
10199 060772
      060772 104406
10200
10201
10202
10203
10204
10205
10206
10207
10208 060774 012703 000001
10209 061000 004737 010556
10210 061004 012703 000400
10211 061010 013737 003114 073132
10212
10213
10214
10215
10216
10217
10218
10219 061016 012737 161001 073130
10220 061024 012704 073130

```

```

;*****
;
;CALL TAPE REWIND COMMAND
;BR, IF NO PROBLEM
;GET TSSR
;SET UP EXPECTED TSSR
;PACKET ADDRESS SET UP
;BUMP COUNT
;REWIND NOT ACCEPTED
TRAP C#ERHRD
.WORD 626
.WORD T26RWN
.WORD PKTSSR
130#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;
;*****
MOV T26BFR+6,R1 ;PICK UP XSTO
MOV R1,R2 ;SET UP EXPECTED
BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
CMP R1,R2 ;DOES EXP = REC'D
BEQ 140# ;BR, IF EQUAL (OK)
INC FATFLG ;BUMP COUNT
ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
TRAP C#ERHRD
.WORD 627
.WORD T26BOT
.WORD EXPREC
140#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
;*****
;
;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
;
;*****
MOV #1,R3 ;SPACE 1 RECORD FORWARD
JSR PC,SPACE ;SPACE CALL
MOV #256.,R3 ;RECORD SIZE
150#: MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
;*****
;
;REREAD DATA,CVC=1,ACK, OPP COMMAND
;
;*****
165#: MOV #161001,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS

```

## TEST 6: REREADS

```

10221 061030 010337 073136        MOV      R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
10222 061034 010465 000000        MOV      R4,TSDB(R5)     ;ISSUE COMMAND
10223 061040 004737 016340        JSR      PC,WAITF        ;WAIT FOR SSR TO SET
10224 061044 016501 000002        MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
10225 061050 012702 000200        MOV      @SSR,R2        ;SET UP EXPECTED
10226 061054 020102                CMP      R1,R2          ;ARE THEY EQUAL
10227 061056 001406                BEQ      170$           ;BR, IF OK
10228 061060 005237 002212        INC      FATFLG         ;BUMP COUNT
10232 061064                ERRHRD  ERRNO,T26RRG,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C#ERHRD
                                .WORD    628
                                .WORD    T26RRG
                                .WORD    PKTSSR
                                TRAP      C#CLP1
                                TRAP      C#CLP1
        10233 061074                170$: CKLOOP                ;LOOP IF SELECTED
        10234 061076 104406                TST      (R3),         ;BUMP RECORD SIZE
10235 061100 062737 000001 073156   ADD      @1,T26CNT       ;BUMP TAPE RECORD COUNTER
10236
10237 ;*****
10238 ;
10239 ;READ DATA, CVC=1, ACK COMMAND
10240 ;
10241 ;*****
10242
10243 061106 012737 140001 073130        MOV      @140001,T26PK3 ;READ DATA, CVC=1, ACK COMMAND
10244 061114 010337 073136        MOV      R3,T26SZ        ;SET SIZE INTO PACKET
10245 061120 010465 000000        MOV      R4,TSDB(R5)     ;ISSUE READ DATA COMMAND
10246 061124 004737 016340        JSR      PC,WAITF        ;WAIT FOR SSR
10247 061130 016501 000002        MOV      TSSR(R5),R1     ;PICK UP THE TSSR
10248 061134 012702 000200        MOV      @SSR,R2        ;SET UP EXPECTED
10249 061140 020102                CMP      R1,R2          ;IS THE TSSR OK
10250 061142 001406                BEQ      195$           ;BR, IF TSSR OK (GOOD)
10251 061144 005237 002212        INC      FATFLG         ;BUMP COUNT
10255 061150                ERRHRD  ERRNO,RDERR,PKTSSR ;READ DATA COMMAND FAILED
                                TRAP      C#ERHRD
                                .WORD    629
                                .WORD    RDERR
                                .WORD    PKTSSR
                                TRAP      C#CLP1
                                TRAP      C#CLP1
        10256 061160                195$: CKLOOP                ;LOOP IF SELECTED
        10257 061162 104406                MOV      @FREE,R1        ;FIRST WORD FROM READ BUFFER
10258 061166 013702 073156        MOV      T26CNT,R2      ;SET UP EXPECTED
10259 061172 020102                CMP      R1,R2          ;IS TAPE POSITION CORRECT
10260 061174 001406                BEQ      197$           ;KEEP GOING POSITION OK
10261 061176 005237 002212        INC      FATFLG         ;BUMP COUNT
10265 061202                ERRHRD  ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
                                TRAP      C#ERHRD
                                .WORD    630
                                .WORD    T26WNG
                                .WORD    EXPREC
                                TRAP      C#CLP1
                                TRAP      C#CLP1
        10266 061212                197$: CKLOOP                ;LOOP IF SELECTED
        10267 061214 104406                CMP      @266.,R3       ;AT MAX SIZE YET
10268 061220 022703 000412        BEQ      200$           ;BR, IF AT END OF THE SUBTEST
10269 061222 000672                BR       150$           ;KEEP GOING MORE RECORDS
10270 061224                200$:
10271 061224                ENDSUB                  ;>>>>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>>>>

```



TEST 6: REREADS

```

10324 061330 005237 002212      INC      FATFLG      ;BUMP COUNT
10328 061334 010001              MOV      R0,R1      ;SAVE CONTENTS OF TSSR
10329 061336              ERRHRD  ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
      061336 104456              TRAP      C#ERHRD
      061340 001170              .WORD    632
      061342 005054              .WORD    WRTMSG
      061344 012124              .WORD    SFMSG
10330 061346              26$:    CKLOOP      ;LOOP IF SELECTED
      061346 104406              TRAP      C#CLP1
10331
10332      ;*****
10333      ;
10334      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10335      ;
10336      ;*****
10337
10338 061350 004737 011104      JSR      PC,REWIND  ;CALL TAPE REWIND COMMAND
10339 061354 103413      BCS     30$        ;BR, IF NO PRGBLEM
10340 061356 C16501 000002      MOV     TSSR(R5),R1 ;GET TSSR
10341 061362 012702 000200      MOV     #SSR,R2    ;SET UP EXPECTED TSSR
10342 061366 010004      MOV     R0,R4      ;PACKET ADDRESS SET UP
10343 061370 005237 002212      INC     FATFLG     ;BUMP COUNT
10347 061374              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      061374 104456              TRAP      C#ERHRD
      061376 001171              .WORD    633
      061400 074464              .WORD    T26RWN
      061402 012136              .WORD    PKTSSR
10348 061404              30$:    CKLOOP      ;LOOP IF SELECTED
      061404 104406              TRAP      C#CLP1
10349
10350      ;*****
10351      ;
10352      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10353      ;
10354      ;*****
10355
10356 061406 013701 073030      MOV     T26BFR+6,R1 ;PICK UP XSTO
10357 061412 010102      MOV     R1,R2      ;SET UP EXPECTED
10358 061414 052702 000002      BIS     #BIT1,R2   ;SET BOT BIT IN EXPECTED
10359 061420 020102      CMP     R1,R2      ;DOES EXP = REC'D
10360 061422 001406      BEQ     40$        ;BR, IF EQUAL (OK)
10361 061424 005237 002212      INC     FATFLG     ;BUMP COUNT
10365 061430              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      061430 104456              TRAP      C#ERHRD
      061432 001172              .WORD    634
      061434 074175              .WORD    T26BOT
      061436 015564              .WORD    EXPREC
10366 061440              40$:    CKLOOP      ;LOOP IF SELECTED
      061440 104406              TRAP      C#CLP1
10367 061442 012703 000400      MOV     #256,R3    ;RECORD SIZE
10368 061446 013737 003114 073132      MOV     FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
10369
10370      ;*****
10371      ;
10372      ;WRITE DATA,CVC=1,ACK COMMAND
10373      ;
10374      ;*****

```

TEST 6: REREADS

```

10375
10376 061454 012737 140005 073130      MOV      #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
10377 061462 012704 073130      MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
10378 061466
10379 061466 010300      65#:    MOV      R3,R0          ;SET PATTERN IN CORRECT REGISTER
10380 061470 004737 017512      JSR      PC,FILLMEM        ;FILL MEMORY WITH RECORD SIZE
10381 061474 010337 073136      MOV      R3,T26SZ         ;SET UP RECORD SIZE IN PACKET
10382 061500 013777 073156      MOV      T26CNT,#FREE     ;MOVE TAPE RECORD NUMBER TO BUFFER
10383 061506 062737 000001 073156      ADD      #1,T26CNT        ;NUMBER READY FOR NEXT RECORD
10384 061514 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
10385 061520 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
10386 061524 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
10387 061530 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED
10388 061534 020102      CMP      R1,R2           ;ARE THEY EQUAL
10389 061536 001406      BEQ      75#             ;BR, IF OK
10390 061540 005237 002212      INC      FATFLG          ;BUMP COUNT
10394 061544      ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C#ERHRD
                                .WORD     635
                                .WORD     WRTErr
                                .WORD     PKTSSR
                                TRAP      C#CLP1
10395 061554      75#:    CKLOOP          ;LOOP IF SELECTED
                                .WORD     635
                                .WORD     WRTErr
                                .WORD     PKTSSR
                                TRAP      C#CLP1
10396 061554 104406      TST      (R3)+           ;BUMP THE RECORD SIZE
10397 061556 005723      CMP      #266.,R3       ;MAXIMUM SIZE YET
10398 061560 022703 000412      BEQ      120#           ;BR, IF AT END OF WRITE SEQUENCE
10399 061564 001401      BR       65#            ;WRITE MORE RECORDS
10400 061570
10401 061570 005037 073156      120#:   CLR      T26CNT      ;SET RECORD COUNTER BACK TO ZERO
10402
10403      ;*****
10404      ;
10405      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10406      ;
10407      ;*****
10408
10409 061574 004737 011104      JSR      PC,REWIND        ;CALL TAPE REWIND COMMAND
10410 061600 103413      BCS     130#            ;BR, IF NO PROBLEM
10411 061602 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
10412 061606 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED TSSR
10413 061612 010004      MOV      R0,R4          ;PACKET ADDRESS SET UP
10414 061614 005237 002212      INC      FATFLG          ;BUMP COUNT
10418 061620      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD     636
                                .WORD     T26RWN
                                .WORD     PKTSSR
10419 061630      130#:   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
10420
10421      ;*****
10422      ;
10423      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10424      ;
10425      ;*****
10426
10427 061632 013701 073030      MOV      T26BFR+6,R1     ;PICK UP XSTO

```

## TEST 6: REREADS

```

10428 061636 010102          MOV      R1,R2          ;SET UP EXPECTED
10429 061640 052702 000002  BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
10430 061644 020102          CMP      R1,R2          ;DOES EXP = REC'D
10431 061646 001406          BEQ      140#          ;BR, IF EQUAL (OK)
10432 061650 005237 002212  INC      FATFLG        ;BUMP COUNT
10436 061654          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    637
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
10437 061664          140# : CKLOOP          ;LOOP IF SELECTED
061664 104406          ;*****
;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
;*****
10446 061666 012705 000001      MOV      #1,R3          ;SET UP SPACE FORWARD 1
10447 061672 004737 010556  JSR      PC,SPACE      ;ISSUE SPACE COMMAND
10448 061676 012703 000400      MOV      #256.,R3      ;RECORD SIZE
10449 061702 013737 003114 073132 150# : MOV      FREE,T26RB   ;STARTING READ BUFFER ADDRESS
;*****
;REREAD DATA,CVC=1,ACK, OPP COMMAND
;*****
10457 061710 012737 171001 073130 165# : MOV      #171001,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
10458 061716 012704 073130      MOV      #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
10459 061722 010337 073136      MOV      R3,T26SZ      ;SET UP RECORD SIZE IN PACKET
10460 061726 010465 000000      MOV      R4,T5DB(R5)   ;ISSUE COMMAND
10461 061732 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
10462 061736 016501 000002      MOV      T5SR(R5),R1   ;GET T5SR CONTENTS
10463 061742 012702 000200      MOV      #SSR,R2       ;SET UP EXPECTED
10464 061746 020102          CMP      R1,R2         ;ARE THEY EQUAL
10465 061750 001406          BEQ      170#          ;BR, IF OK
10466 061752 005237 002212  INC      FATFLG        ;BUMP COUNT
10470 061756          ERRHRD  ERRNO,T26RRF,PKTSSR ;T5SR INCORRECT AFTER REREAD DATA
                                TRAP      C#ERHRD
                                .WORD    638
                                .WORD    T26RRF
                                .WORD    PKTSSR
                                TRAP      C#CLP1
10471 061766          170# : CKLOOP          ;LOOP IF SELECTED
061766 104406          ;*****
10472 061770 017701 121120      MOV      #FREE,R1      ;FIRST WORD FROM READ BUFFER
10473 061774 013702 073156      MOV      T26CNT,R2     ;SET UP EXPECTED
10474 062000 000302          SWAB     R2            ;SWAP BYTES IN EXPECTED
10475 062002 020102          CMP      R1,R2         ;IS TAPE POSITION CORRECT
10476 062004 001406          BEQ      190#          ;KEEP GOING POSITION OK
10477 062006 005237 002212  INC      FATFLG        ;BUMP COUNT
10481 062012          ERRHRD  ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
                                TRAP      C#ERHRD
                                .WORD    639
062012 104456
062014 001177

```



TEST 6: REREADS

```

10482 062016 073166          .WORD  T26WNG
      062020 015564          .WORD  EXPREC
10482 062022 104406          TRAP   C#CLP1
      062022 005723          ;NEXT RECORD SIZE
10483 062024 062737 000001 073156  TST    (R3)+      ;BUMP TAPE RECORD COUNTER
10484 062026 000001 073156  ADD    #1,T26CNT
10485
10486 ;*****
10487 ;
10488 ;READ DATA, CVC=1, ACK COMMAND
10489 ;
10490 ;*****
10491
10492 062034 012737 140001 073130  MOV    #140001,T26PK3      ;READ DATA, CVC=1, ACK COMMAND
10493 062042 010337 073136  MOV    R3,T26SZ           ;SET SIZE INTO PACKET
10494 062046 010465 000000  MOV    R4,TSDB(R5)        ;ISSUE READ DATA COMMAND
10495 062052 004737 016340  JSR    PC,WAITF           ;WAIT FOR SSR
10496 062056 016501 000002  MOV    TSSR(R5),R1        ;PICK UP THE TSSR
10497 062062 C12702 000200  MOV    #SSR,R2           ;SET UP EXPECTED
10498 062066 020102  CMP    R1,R2             ;IS THE TSSR OK
10499 062070 001406  BEQ    215#              ;BR, IF TSSR OK (GOOD)
10500 062072 005237 002212  INC    FATFLG             ;BUMP COUNT
10504 062076 104456  ERRHRD ERRNO,T26RDF,PKTSSR ;READ DATA COMMAND FAILED
      062076 001200          TRAP   C#ERHRD
      062100 073336          .WORD  640
      062102 012136          .WORD  T26RDF
      062104          .WORD  PKTSSR
10505 062106 104406          ;LOOP IF SELECTED
      062106 121000          TRAP   C#CLP1
10506 062110 013702 073156  MOV    %FREE,R1          ;FIRST WORD FROM READ BUFFER
10507 062114 020102  MOV    T26CNT,R2        ;SET UP EXPECTED
10508 062120 001406  CMP    R1,R2           ;IS TAPE POSITION CORRECT
10509 062122 005237 002212  BEQ    217#             ;KEEP GOING POSITION OK
10510 062124 005237 002212  INC    FATFLG           ;BUMP COUNT
10514 062130 104456  ERRHRD ERRNO,T26WNG,EXPREC ;TAPE POSITION INCORRECT
      062130 001201          TRAP   C#ERHRD
      062132 073166          .WORD  641
      062134 015564          .WORD  T26WNG
      062136          .WORD  EXPREC
10515 062140 104406          ;AT MAX SIZE YET
      062140 000410          TRAP   C#CLP1
10516 062142 001401  CMP    #264.,R3        ;BR, IF AT END OF THE SUBTEST
10517 062146 000654  BEQ    220#             ;KEEP GOING MORE RECORDS
10518 062150  BR     150#
10519 062152
10520 062152  ENDSUB              ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
      062152 104403          L10106:
10521 062154 023727 002212 000017  CMP    FATFLG,#15.        ;IS ERROR COUNT AT 25
10522 062162 103402  BLO    999#            ;BR, IF LESS THAN 25
10523 062164 004737 017272  JSR    PC,CKDROP         ;TRY TO DROP THE UNIT
10524 062170 999#:
10525
10526 ;*
10527 ;
10528 ;TEST 6, SUBTEST 5
10529 ;

```



TEST 6: REREADS

```

10582 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
10583 ;
10584 ;*****
10585
10586 062272 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
10587 062276 103413 BCS 30$ ;BR, IF NO PROBLEM
10588 062300 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
10589 062304 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
10590 062310 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
10591 062312 005237 002212 INC FATFLG ;BUMP COUNT
10595 062316 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP C#ERHRD
                                .WORD 644
                                .WORD T26RWN
                                .WORD PKTSSR
                                TRAP C#CLP1
062316 104456
062320 001204
062322 074464
062324 012136
10596 062326 30$: CKLOOP ;LOOP IF SELECTED
062326 104406
10597 ;*****
10598 ;
10599 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
10600 ;
10601 ;*****
10602
10603
10604 062330 013701 073030 MOV T26BFR+6,R1 ;PICK UP XST0
10605 062334 010102 MOV R1,R2 ;SET UP EXPECTED
10606 062336 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
10607 062342 020102 CMP R1,R2 ;DOES EXP = REC'D
10608 062344 001406 BEQ 40$ ;BR, IF EQUAL (OK)
10609 062346 005237 002212 INC FATFLG ;BUMP COUNT
10613 062352 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP C#ERHRD
                                .WORD 645
                                .WORD T26BOT
                                .WORD EXPREC
                                TRAP C#CLP1
062352 104456
062354 001205
062356 074175
062360 015564
10614 062362 40$: CKLOOP ;LOOP IF SELECTED
062362 104406
10615 062364 012703 001000 MOV #512.,R3 ;RECORD SIZE
10616 062370 013737 003114 073132 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
10617 ;*****
10618 ;
10619 ;WRITE DATA,CVC=1,ACK COMMAND
10620 ;
10621 ;*****
10622
10623
10624 062376 012737 140005 073130 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10625 062404 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
10626 062410 65$:
10627 062410 010337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
10628 062414 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
10629 062420 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
10630 062424 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
10631 062430 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
10632 062434 020102 CMP R1,R2 ;ARE THEY EQUAL
10633 062436 001406 BEQ 75$ ;BR, IF OK
10634 062440 005237 002212 INC FATFLG ;BUMP COUNT

```

TEST 6: REREADS

```

10638 062444          ERRHRD  ERRNO,WRERR,PKTSSR      ;TSSR INCORRECT AFTER WRITE DATA
      062444 104456          TRAP  C#ERHRD
      062446 001206          .WORD 646
      062450 005111          .WORD WRERR
      062452 012136          .WORD PKTSSR
10639 062454          75$:  CKLOOP                    ;LOOP IF SELECTED
      062454 104406          TRAP  C#CLP1
10640 062456 005303          DEC  R3                    ;SET RECORD SIZE TO 511.
10641 062460 013737 003114 073132  MOV  FREE,T26RB          ;STARTING READ BUFFER ADDRESS
10642
10643 ;*****
10644 ;
10645 ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
10646 ;
10647 ;*****
10648
10649 062466 012737 161001 073130  MOV  #16100,T26PK3      ;REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
10650 062474 012704 073130 165$:  MOV  #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
10651 062500 C10337 073136  MOV  R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
10652 062504 010465 000000  MOV  R4,TSDB(R5)      ;ISSUE COMMAND
10653 062510 004737 016340  JSR  PC,WAITF         ;WAIT FOR SSR TO SET
10654 062514 016501 000002  MOV  TSSR(R5),R1      ;GET TSSR CONTENTS
10655 062520 012702 100204  MOV  #SSR!SC!BIT2,R2 ;SET UP EXPECTED
10656 062524 020102  CMP  R1,R2            ;ARE THEY EQUAL
10657 062526 001406  BEQ  170$           ;BR, IF OK
10658 062530 005237 002212  INC  FATFLG          ;BUMP COUNT
10662 062534          ERRHRD  ERRNO,T26TRL,PKTSSR      ;TSSR INCORRECT AFTER REREAD DATA
      062534 104456          TRAP  C#ERHRD
      062536 001207          .WORD 647
      062540 075542          .WORD T26TRL
      062542 012136          .WORD PKTSSR
10663 062544          170$:  CKLOOP                    ;LOOP IF SELECTED
      062544 104406          TRAP  C#CLP1
10664
10665 ;*****
10666 ;
10667 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10668 ;
10669 ;*****
10670
10671 062546 013701 073030  MOV  T26BFR+6,R1      ;GET MESSAGE BUFFER
10672 062552 010102  MOV  R1,R2            ;SET UP EXPECTED
10673 062554 052702 010000  BIS  #BIT12,R2       ;SET THE RLL BIT IN EXPECTED
10674 062560 020102  CMP  R1,R2            ;ARE THEY EQUAL
10675 062562 001406  BEQ  180$           ;BR, IF EQUAL (ALL IS WELL)
10676 062564 005237 002212  INC  FATFLG          ;BUMP COUNT
10680 062570          ERRHRD  ERRNO,T26LON,EXPREC      ;THE RLL BIT WAS NOT SET IN XSTO
      062570 104456          TRAP  C#ERHRD
      062572 001210          .WORD 648
      062574 075310          .WORD T26LON
      062576 015564          .WORD EXPREC
10681 062600          180$:  CKLOOP                    ;LOOP IF SELECTED
      062600 104406          TRAP  C#CLP1
10682 062602 012703 000777          MOV  #511.,R3          ;SET RECORD SIZE
10683 062606 013737 003114 073132  MOV  FREE,T26RB          ;STARTING READ BUFFER ADDRESS
10684
10685 ;*****

```



## TEST 6: REREADS

```

10737          ;SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE
10738          ;RESIDUAL BYTE COUNTER (RBPCR) IN THE MESSAGE BUFFER
10739          ;CONTAINS THE PROPER NONZERO VALUE (E.G., THE
10740          ;DIFFERENCE BETWEEN THE ORIGINAL BYTE COUNT AND THE
10741          ;ACTUAL RECORD LENGTH). RESULTS ARE VERIFIED FOR BOTH
10742          ;STATES OF OPP (0 AND 1).
10743          ;
10744          ;
10745          ;
10746          ;-
10747 062746      BGNSUB                ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>
        062746      T6.6:
        062746      104402      TRAP      C#BSUB
10748 062750      004737      075720      JSR      PC.T26REST      ;SET COMMAND PACKET
10749 062754      004737      076012      JSR      PC.T26RT2      ;SET UP OTHER COMMAND PACKET
10750 062760      004737      076054      JSR      PC.T26RT3      ;SET UP OTHER COMMAND PACKET
10751          ;
10752          ;*****
10753          ;
10754          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
10755          ;
10756          ;*****
10757          ;
10758 062764      004737      016064      JSR      PC.SOFINIT      ;DO INITIALIZE ON CONTROLLER
10759 062770      103407      BCS      20#      ;BR IF INIT WAS OK
10760 062772      005237      002212      INC      FATFLG      ;BUMP COUNT
10764 062776      010001      MOV      R0,R1      ;CONTENTS OF TSSR REGISTER
10765 063000      ERRDF      ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
        063000      104455      TRAP      C#ERDF
        063002      001213      .WORD    651
        063004      003650      .WORD    SFIERR
        063006      012124      .WORD    SFIMSG
10766 063010      013737      002172      073020 20#:  MOV      UNITN,T26DSW      ;SET UP UNIT NUMBER
10767          ;
10768 063016      012704      073000      MOV      @T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
10769          ;
10770          ;*****
10771          ;
10772          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
10773          ;
10774          ;*****
10775          ;
10776 063022      004737      010752      JSR      PC.WRTCHR      ;ISSUE WRITE CHARACTERISTICS
10777 063026      103407      BCS      26#      ;BR, IF COMMAND ISSUED OK
10778 063030      005237      002212      INC      FATFLG      ;BUMP COUNT
10782 063034      010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
10783 063036      ERRHRD      ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
        063036      104456      TRAP      C#ERHRD
        063040      001214      .WORD    652
        063042      005054      .WORD    WRTMSG
        063044      012124      .WORD    SFIMSG
10784 063046      104406      26#:  CKLOOP      ;LOOP IF SELECTED
        063046      TRAP      C#CLP1
10785          ;
10786          ;*****
10787          ;
10788          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE

```

## TEST 6: REREADS

```

10789
10790 ;
10791 ;*****
10792 063050 004737 011104 JSR PC.REWIND ;CALL TAPE REWIND COMMAND
10793 063054 103413 BCS 300 ;BR, IF NO PROBLEM
10794 063056 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
10795 063062 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
10796 063066 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
10797 063070 005237 002212 INC FATFLG ;BUMP COUNT
10801 063074 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      063074 104456 TRAP C0ERHRD
      063076 001215 .WORD 653
      063100 074464 .WORD T26RWN
      063102 012136 .WORD PKTSSR
10802 063104 300: CKLOOP ;LOOP IF SELECTED TRAP C0CLP1
      063104 104406
10803
10804 ;*****
10805 ;
10806 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
10807 ;
10808 ;*****
10809
10810 063106 013701 073030 MOV T26FR+6,R1 ;PICK UP XST0
10811 063112 010102 MOV R1,R2 ;SET UP EXPECTED
10812 063114 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
10813 063120 020102 CMP R1,R2 ;DOES EXP = REC'D
10814 063122 001406 BEQ 400 ;BR, IF EQUAL (OK)
10815 063124 005237 002212 INC FATFLG ;BUMP COUNT
10819 063130 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      063130 104456 TRAP C0ERHRD
      063132 001216 .WORD 654
      063134 074175 .WORD T26BOT
      063136 015564 .WORD EXPREC
10820 063140 400: CKLOOP ;LOOP IF SELECTED TRAP C0CLP1
      063140 104406
10821 063142 012703 000400 MOV #256,R3 ;RECORD SIZE
10822 063146 013737 003114 073132 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
10823
10824 ;*****
10825 ;
10826 ;WRITE DATA,CVC=1,ACK COMMAND
10827 ;
10828 ;*****
10829
10830 063154 012737 140005 073130 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
10831 063162 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
10832 063166 650:
10833 063166 010337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
10834 063172 010465 C00000 MOV R4,TSD0(R5) ;ISSUE COMMAND
10835 063176 004737 016340 JSR PC.WAITF ;WAIT FOR SSR TO SET
10836 063202 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
10837 063206 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
10838 063212 020102 CMP R1,R2 ;ARE THEY EQUAL
10839 063214 001406 BEQ 750 ;BR, IF OK
10840 063216 005237 002212 INC FATFLG ;BUMP COUNT
10844 063222 ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA

```

TEST 6: REREADS

```

063222 104456 TRAP C#ERHRD
063224 001217 .WORD 655
063226 005111 .WORD WRTERR
063230 012136 .WORD PKTSSR
10845 063232 75#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
063232 104406
10846 063234 012703 001000 MOV #512.,R3 ;RECORD SIZE
10847 063240 013737 003114 073132 MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
10848
10849 ;*****
10850 ;
10851 ;REREAD PREVIOUS,ACK,CVC=1,OPP=1
10852 ;
10853 ;*****
10854
10855 063246 012737 161001 073130 MOV #161001,T26PK3 ;REREAD PREVIOUS,ACK,CVC=1,OPP=1
10856 063254 012704 073130 165#: MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
10857 063260 010337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
10858 063264 C10465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
10859 063270 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
10860 063274 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
10861 063300 012702 100204 MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
10862 063304 020102 CMP R1,R2 ;ARE THEY EQUAL
10863 063306 001406 BEQ 170# ;BR, IF OK
10864 063310 005237 002212 INC FATFLG ;BUMP COUNT
10868 063314 ERRHRD ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
063314 104456 TRAP C#ERHRD
063316 001220 .WORD 656
063320 075542 .WORD T26TRL
063322 012136 .WORD PKTSSR
10869 063324 170#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
063324 104406
10870
10871 ;*****
10872 ;
10873 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
10874 ;
10875 ;*****
10876
10877 063326 013701 073030 MOV T26BFR+6,R1 ;GET MESSAGE BUFFER
10878 063332 010102 MOV R1,R2 ;SET UP EXPECTED
10879 063334 052702 040000 BIS #BIT14,R2 ;SET THE RLS BIT IN EXPECTED
10880 063340 020102 CMP R1,R2 ;ARE THEY EQUAL
10881 063342 001406 BEQ 180# ;BR, IF EQUAL (ALL IS WELL)
10882 063344 005237 002212 INC FATFLG ;BUMP COUNT
10886 063350 ERRHRD ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XST0
063350 104456 TRAP C#ERHRD
063352 001221 .WORD 657
063354 075372 .WORD T26LOP
063356 015564 .WORD EXPREC
10887 063360 180#: CKLOOP TRAP C#CLP1
063360 104406
10888 063362 013701 073026 MOV T26BFR+4,R1 ;PICK UP RESIDUAL BYTE COUNTER
10889 063366 012702 000400 MOV #256.,R2 ;THIS SHOULD BE THE DIFFERENCE
10890 063372 020102 CMP R1,R2 ;IS THE DIFFERENCE CORRECT
10891 063374 001406 BEQ 190# ;BR, IF CORRECT
10892 063376 005237 002212 INC FATFLG ;BUMP COUNT

```



TEST 6: REREADS

```

10896 063402          ERRHRD  ERRNO,T26PBP,EXPREC      ;RBPCR NOT CORRECT          TRAP  C#ERHRD
      063402 104456          .WORD 658
      063404 001222          .WORD T26PBP
      063406 075454          .WORD EXPREC
      063410 015564
10897 063412          190#: CKLOOP                      ;LOOP IF SELECTED          TRAP  C#CLP1
      063412 104406
10898 063414 001000      MOV    #512.,R3          ;RECORD SIZE
10899 063420 013737 003114 073132      MOV    FREE,T26RB      ;STARTING READ BUFFER ADDRESS
10900
10901      ;*****
10902      ;
10903      ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
10904      ;
10905      ;*****
10906
10907 063426 012737 141001 073130      MOV    #141001,T26PK3  ;REREAD PREVIOUS,ACK,CVC=1,OPP=0
10908 063434 012704 073130      MOV    #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
10909 063440 C10337 073136      MOV    R3,T26SZ       ;SET UP RECORD SIZE IN PACKET
10910 063444 010465 000000      MOV    R4,TSDB(R5)    ;ISSUE COMMAND
10911 063450 004737 016340      JSR    PC,WAITF       ;WAIT FOR SSR TO SET
10912 063454 016501 000002      MOV    TSSR(R5),R1    ;GET TSSR CONTENTS
10913 063460 012702 100204      MOV    #SSR!SC!BIT2,R2 ;SET UP EXPECTED
10914 063464 020102      CMP    R1,R2          ;ARE THEY EQUAL
10915 063466 001406      BEQ    270#           ;BR, IF OK
10916 063470 005237 002212      INC    FATFLG         ;BUMP COUNT
10920 063474          ERRHRD  ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      063474 104456          TRAP  C#ERHRD
      063476 001223          .WORD 659
      063500 075542          .WORD T26TRL
      063502 012136          .WORD PKTSSR
10921 063504          270#: CKLOOP                      ;LOOP IF SELECTED          TRAP  C#CLP1
      063504 104406
10922
10923      ;*****
10924      ;
10925      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
10926      ;
10927      ;*****
10928
10929 063506 013701 073030      MOV    T26BFR+6,R1    ;GET MESSAGE BUFFER
10930 063512 010102      MOV    R1,R2          ;SET UP EXPECTED
10931 063514 052702 040000      BIS    #BIT14,R2     ;SET THE RLS BIT IN EXPECTED
10932 063520 020102      CMP    R1,R2          ;ARE THEY EQUAL
10933 063522 001406      BEQ    280#           ;BR, IF EQUAL (ALL IS WELL)
10934 063524 005237 002212      INC    FATFLG         ;BUMP COUNT
10938 063530          ERRHRD  ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
      063530 104456          TRAP  C#ERHRD
      063532 001224          .WORD 660
      063534 075372          .WORD T26LOP
      063536 015564          .WORD EXPREC
10939 063540          280#: CKLOOP                      ;LOOP IF SELECTED          TRAP  C#CLP1
      063540 104406
10940 063542 013701 073026      MOV    T26BFR+4,R1    ;PICK UP RESIDUAL BYTE COUNTER
10941 063546 012702 000400      MOV    #256.,R2      ;THIS SHOULD BE THE DIFFERENCE
10942 063552 020102      CMP    R1,R2          ;IS THE DIFFERENCE CORRECT
10943 063554 001406      BEQ    290#           ;BR, IF CORRECT

```



TEST 6: REREADS

```

10994 063652 013737 002172 073020 20:  MOV    UNITN,T26DSW          ;SET UP UNIT NUMBER
10995
10996 063660 012704 073000          MOV    #T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
10997
10998 ;*****
10999 ;
11000 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
11001 ;
11002 ;*****
11003
11004 063664 004737 010752          JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
11005 063670 103407          BCS    26:                ;BR, IF COMMAND ISSUED OK
11006 063672 005237 002212          INC    FATFLG            ;BUMP COUNT
11010 063676 010001          MOV    R0,R1             ;SAVE CONTENTS OF TSSR
11011 063700          ERRHRD  ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C#ERHRD
                                .WORD   662
                                .WORD   WRTMSG
                                .WORD   SFMSG
                                063700 104456
                                063702 001226
                                063704 005054
                                063706 C12124
11012 063710          26:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C#CLP1
                                063710 104406
11013
11014 ;*****
11015 ;
11016 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11017 ;
11018 ;*****
11019
11020 063712 004737 011104          JSR    PC,REWIND         ;CALL TAPE REWIND COMMAND
11021 063716 103413          BCS    30:                ;BR, IF NO PROBLEM
11022 063720 016501 000002          MOV    TSSR(R5),R1      ;GET TSSR
11023 063724 012702 000200          MOV    #SSR,R2          ;SET UP EXPECTED TSSR
11024 063730 010004          MOV    R0,R4             ;PACKET ADDRESS SET UP
11025 063732 005237 002212          INC    FATFLG            ;BUMP COUNT
11029 063736          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C#ERHRD
                                .WORD   663
                                .WORD   T26RWN
                                .WORD   PKTSSR
                                063736 104456
                                063740 001227
                                063742 074464
                                063744 012136
11030 063746          30:  CKLOOP              ;LOOP IF SELECTED
                                TRAP    C#CLP1
                                063746 104406
11031
11032 ;*****
11033 ;
11034 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11035 ;
11036 ;*****
11037
11038 063750 013701 073030          MOV    T26BFR+6,R1      ;PICK UP XSTO
11039 063754 010102          MOV    R1,R2             ;SET UP EXPECTED
11040 063756 052702 000002          BIS    #BIT1,R2         ;SET BOT BIT IN EXPECTED
11041 063762 020102          CMP    R1,R2             ;DOES EXP = REC'D
11042 063764 001406          BEQ    40:                ;BR, IF EQUAL (OK)
11043 063766 005237 002212          INC    FATFLG            ;BUMP COUNT
11047 063772          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C#ERHRD
                                .WORD   664
                                063772 104456
                                063774 001230

```

## TEST 6: REREADS

```

063776 074175 .WORD T26BOT
064000 015564 .WORD EXPREC
11048 064002 40#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
064002 104406 ;RECORD SIZE
11049 064004 012703 000400 MOV #256.,R3 ;STARTING WRITE BUFFER ADDRESS
11050 064010 013737 003114 073132 MOV FREE,T26RB
11051 ;*****
11052 ;
11053 ;WRITE DATA,CVC=1,ACK COMMAND
11054 ;
11055 ;*****
11056 ;
11057 ;
11058 064016 012737 140005 073130 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
11059 064024 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
11060 064030 65#:
11061 064030 010300 MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
11062 064032 004737 017512 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
11063 064036 C10337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
11064 064042 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
11065 064046 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
11066 064052 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
11067 064056 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
11068 064062 020102 CMP R1,R2 ;ARE THEY EQUAL
11069 064064 001406 BEQ 75# ;BR, IF OK
11070 064066 005237 002212 INC FATFLG ;BUMP COUNT
11074 064072 ERRHRD ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFIER WRITE DATA
064072 104456 TRAP C#ERHRD
064074 001231 .WORD 665
064076 005111 .WORD WRERR
064100 012136 .WORD PKTSSR
11075 064102 75#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
064102 104406 ;BUMP RECORD SIZE
11076 064104 005723 ;END OF RECORD YET
11077 064106 022703 000414 CMP #268.,R3 ;BR, IF MORE RECORDS TO WRITE
11078 064112 001346 BNE 65# ;LOOP IF SELECTED TRAP C#CLP1
11079 064114 80#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
064114 104406 ;
11080 064116 120#:
11081 ;*****
11082 ;
11083 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11084 ;
11085 ;*****
11086 ;
11087 ;
11088 064116 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
11089 064122 103413 BCS 130# ;BR, IF NO PROBLEM
11090 064124 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
11091 064130 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
11092 064134 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
11093 064136 005237 002212 INC FATFLG ;BUMP COUNT
11097 064142 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED TRAP C#ERHRD
064142 104456 .WORD 666
064144 001232 .WORD T26RWN
064146 074464 .WORD PKTSSR
064150 012136

```

TEST 6: REREADS

```

11098 064152 104406      130$:  CKLOOP                ;LOOP IF SELECTED          TRAP  C#CLP1
      064152
11099
11100 ;*****
11101 ;
11102 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11103 ;
11104 ;*****
11105
11106 064154 013701 073030      MOV      T26BFR+6,R1      ;PICK UP XSTO
11107 064160 010102      MOV      R1,R2           ;SET UP EXPECTED
11108 064162 052702 000002      BIS      #BIT1,R2        ;SET BOT BIT IN EXPECTED
11109 064166 020102      CMP      R1,R2           ;DOES EXP = REC'D
11110 064170 001406      BEQ      140$            ;BR, IF EQUAL (OK)
11111 064172 005237 002212      INC      FATFLG          ;BUMP COUNT
11115 064176      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      064176 104456      TRAP  C#ERHRD
      064200 001233      .WORD 667
      064202 C74175      .WORD T26BOT
      064204 015564      .WORD EXPREC
11116 064206      140$:  CKLOOP                ;LOOP IF SELECTED          TRAP  C#CLP1
      064206 104406
11117 064210 012737 000400 073162      MOV      #256.,T26RSZ    ;STORE START RECORD SIZE
11118 064216 000420      BR       150$            ;SKIP THE SPACE THIS TIME
11119
11120 ;*****
11121 ;
11122 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
11123 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
11124 ;
11125 ;*****
11126
11127 064220 012703 000001      145$:  MOV      #1,R3      ;SPACE ONE RECORD PARAMETER
11128 064224 004737 010556      JSR      PC,SPACE        ;CALL SPACE ROUTINE
11129 064230 103413      BCS     150$            ;BR, IF NO PROBLEM WITH SPACE COMMAND
11130 064232 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
11131 064236 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED TSSR
11132 064242 010004      MOV      R0,R4          ;PACKET ADDRESS SET UP
11133 064244 005237 002212      INC      FATFLG          ;BUMP COUNT
11137 064250      ERRHRD  ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
      064250 104456      TRAP  C#ERHRD
      064252 001234      .WORD 668
      064254 073577      .WORD T26SC
      064256 015564      .WORD EXPREC
11138 064260      150$:  CKLOOP                ;LOOP IF SELECTED          TRAP  C#CLP1
      064260 104406
11139 064262 013703 073162      MOV      T26RSZ,R3       ;RECORD SIZE
11140 064266 013737 003114 073132      MOV      FREE,T26RB      ;STARTING READ BUFFER ADDRESS
11141
11142 ;*****
11143 ;
11144 ;REREREAD DATA,CVC=1,ACK COMMAND
11145 ;
11146 ;*****
11147
11148 064274 012737 141401 073130      165$:  MOV      #141401,T26PK3 ;REREREAD DATA,CVC=1,ACK COMMAND
11149 064302 012704 073130      MOV      #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS

```





## TEST 6: REREADS

```

11252 064572 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR
11253 064576 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED TSSR
11254 064602 010004              MOV      R0,R4          ;PACKET ADDRESS SET UP
11255 064604 005237 002212      INC      FATFLG         ;BUMP COUNT
11259 064610              ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
          064610 104456              TRAP      C#ERHRD
          064612 001241              .WORD    673
          064614 074464              .WORD    T26RWN
          064616 012136              .WORD    PKTSSR
11260 064620 104406      30$:   CKLOOP              ;LOOP IF SELECTED
          064620 104406              TRAP      C#CLP1
11261
11262      ;*****
11263      ;
11264      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11265      ;
11266      ;*****
11267
11268 064622 013701 073030      MOV      T26BFR+6,R1    ;PICK UP XSTO
11269 064626 010102      MOV      R1,R2          ;SET UP EXPECTED
11270 064630 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
11271 064634 020102      CMP      R1,R2          ;DOES EXP = REC'D
11272 064636 001406      BEQ      40$           ;BR, IF EQUAL (OK)
11273 064640 005237 002212      INC      FATFLG         ;BUMP COUNT
11277 064644              ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          064644 104456              TRAP      C#ERHRD
          064646 001242              .WORD    674
          064650 074175              .WORD    T26BOT
          064652 015564              .WORD    EXPREC
11278 064654 104406      40$:   CKLOOP              ;LOOP IF SELECTED
          064654 104406              TRAP      C#CLP1
11279 064656 012703 000400      MOV      #256.,R3       ;RECORD SIZE
11280 064662 013737 003114 073132      MOV      FREE,T26RB     ;STARTING WRITE BUFFER ADDRESS
11281
11282      ;*****
11283      ;
11284      ;WRITE DATA,CVC=1,ACK,SWB COMMAND
11285      ;
11286      ;*****
11287
11288 064670 012737 150005 073130      MOV      #150005,T26PK3 ;WRITE DATA,CVC=1,ACK,SWB COMMAND
11289 064676 012704 073130      MOV      #T26PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
11290 064702
11291 064702 010300      65$:   MOV      R3,R0        ;SET PATTERN IN CORRECT REGISTER
11292 064704 004737 017512      JSR      PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
11293 064710 010337 073136      MOV      R3,T26SZ       ;SET UP RECORD SIZE IN PACKET
11294 064714 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
11295 064720 004737 016340      JSR      PC,WAITF       ;WAIT FOR SSR TO SET
11296 064724 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
11297 064730 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED
11298 064734 020102      CMP      R1,R2          ;ARE THEY EQUAL
11299 064736 001406      BEQ      75$           ;BR, IF OK
11300 064740 005237 002212      INC      FATFLG         ;BUMP COUNT
11304 064744              ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
          064744 104456              TRAP      C#ERHRD
          064746 001243              .WORD    675
          064750 005111              .WORD    WRERR

```



TEST 6: REREADS

```

11305 064752 012136
11305 064754 104406
11306 064756 005723
11307 064760 022703 000414
11308 064764 001346
11309 064766 104406
11310 064770
11311
11312
11313
11314
11315
11316
11317
11318 064770 004737 011104
11319 064774 103413
11320 064776 C16501 000002
11321 065002 012702 000200
11322 065006 010004
11323 065010 005237 002212
11327 065014
11328 065024 104406
11329
11330
11331
11332
11333
11334
11335
11336 065026 013701 073030
11337 065032 010102
11338 065034 052702 000002
11339 065040 020102
11340 065042 001406
11341 065044 005237 002212
11345 065050
11346 065060 104406
11347 065062 012737 000400 073162
11348 065070 000420
11349
11350
11351
11352
11353
11354

75: CKLOOP ;LOOP IF SELECTED .WORD PKTSSR
; BUMP RECORD SIZE TRAP C#CLP1
TST (R3)+ ;END OF RECORD YET
CMP #268.,R3 ;BR, IF MORE RECORDS TO WRITE
BNE 65: ;LOOP IF SELECTED TRAP C#CLP1

80: CKLOOP

120:
;*****
;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
;*****
JSR PC.REWIND ;CALL TAPE REWIND COMMAND
BCS 130: ;BR, IF NO PROBLEM
MOV TSSR(R5),R1 ;GET TSSR
MOV #SSR,R2 ;SET UP EXPECTED TSSR
MOV R0,R4 ;PACKET ADDRESS SET UP
INC FATFLG ;BUMP COUNT
ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED TRAP C#ERHRD
;*****
;*****
130: CKLOOP ;LOOP IF SELECTED .WORD 676
;*****
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
;*****
MOV T26BFR+6,R1 ;PICK UP XST0
MOV R1,R2 ;SET UP EXPECTED
BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
CMP R1,R2 ;DOES EXP = REC'D
BEQ 140: ;BR, IF EQUAL (OK)
INC FATFLG ;BUMP COUNT
ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND TRAP C#ERHRD
;*****
;*****
140: CKLOOP ;LOOP IF SELECTED .WORD 12680T
; START RECORD SIZE TRAP C#CLP1
BR 150: ;SKIP SAPCE THIS TIME
;*****
;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
;BIT 15 SETS DIRECTION - 0-FORWARD 1-REVERSE
;*****

```

TEST 6: REREADS

```

11355 ;*****
11356
11357 065072 012703 000001 145: MOV #1,R3 ;SPACE ONE RECORD PARAMETER
11358 065076 004737 010556 JSR PC,SPACE ;CALL SPACE ROUTINE
11359 065102 103413 BCS 150: ;BR, IF NO PROBLEM WITH SPACE COMMAND
11360 065104 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
11361 065110 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
11362 065114 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
11363 065116 005237 002212 INC FATFLG ;BUMP COUNT
11367 065122 ERRHRD ERRNO,T26SC,EXPREC ;POSITION (SPACE RECORDS) FAILED
                                TRAP C#ERHRD
                                .WORD 678
                                .WORD T26SC
                                .WORD EXPREC
                                TRAP C#CLP1
11368 065132 150: CKLOOP
                                TRAP C#CLP1
11369 065134 104406 MOV T26RSZ,R3 ;RECORD SIZE
11370 065140 013703 073162 MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
11371 013737 003114 073132
11372 ;*****
11373 ;
11374 ;REREAD DATA,ACK,CVC=1,SWB COMMAND
11375 ;
11376 ;*****
11377
11378 065146 012737 151401 073130 165: MOV #151401,T26PK3 ;REREAD DATA,ACK,CVC=1,SWB COMMAND
11379 065154 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
11380 065160 010337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
11381 065164 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
11382 065170 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
11383 065174 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
11384 065200 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
11385 065204 020102 CMP R1,R2 ;ARE THEY EQUAL
11386 065206 001406 BEQ 170: ;BR, IF OK
11387 065210 005237 002212 INC FATFLG ;BUMP COUNT
11391 065214 ERRHRD ERRNO,T26WDC,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP C#ERHRD
                                .WORD 679
                                .WORD T26WDC
                                .WORD PKTSSR
11392 065224 170: CKLOOP ;LOOP IF SELECTED
                                TRAP C#CLP1
11393 065226 104406 MOV FREE,R2 ;CURRENT BUFFER ADDRESS TO R2
11394 065232 013702 003114 MOV R3,R4 ;CURRENT RECORD SIZE
11395 065234 162704 000400 SUB #256.,R4 ;FIRST LOCATION IN BUFFER
11396 065240 060204 173: ADD R2,R4 ;SET UP POINTER
11397 065242 021403 CMP (R4),R3 ;CHECK DATA READ (R3=DATA ALSO)
11398 065244 001410 BEQ 180: ;BR, IF ALL IS WELL
11399 065246 011401 MOV (R4),R1 ;RECD DATA
11400 065250 010302 MOV R3,R2 ;EXPECTED DATA
11401 065252 005237 002212 INC FATFLG ;BUMP COUNT
11405 065256 ERRHRD ERRNO,T26DTA,EXPREC ;DATA READ NOT - WRITTEN
                                TRAP C#ERHRD
                                .WORD 680
                                .WORD T26DTA
                                .WORD EXPREC
11406 065266 180: CKLOOP ;LOOP IF SELECTED

```



TEST 6: REREADS

```

11458 065346 004737 076012          JSR    PC,T26RT2          ;SET UP OTHER COMMAND PACKET
11459 065352 004737 076054          JSR    PC,T26RT3          ;SET UP OTHER COMMAND PACKET
11460
11461          ;*****
11462          ;
11463          ;ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
11464          ;
11465          ;*****
11466
11467 065356 004737 016064          JSR    PC,SOFINIT          ;DO INITIALIZE ON CONTROLLER
11468 065362 103407                   BCS    20#                 ;BR IF INIT WAS OK
11469 065364 005237 002212          INC    FATFLG              ;BUMP COUNT
11473 065370 010001                   MOV    RO,R1               ;CONTENTS OF TSSR REGISTER
11474 065372                   ERROF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
                                TRAP    C0ERDF
                                .WORD   681
                                .WORD   SFIERR
                                .WORD   SFIMSG
11475 065402 C13737 002172 073020 20# :  MOV    UNITN,T26DSW          ;SET UP UNIT NUMBER
11476
11477 065410 012704 073000          MOV    @T26PACKET,R4       ;SUBROUTINE NEEDS PACKET ADDRESS
11478
11479          ;*****
11480          ;
11481          ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
11482          ;
11483          ;*****
11484
11485 065414 004737 010752          JSR    PC,WRTCHR           ;ISSUE WRITE CHARACTERISTICS
11486 065420 103407                   BCS    26#                 ;BR, IF COMMAND ISSUED OK
11487 065422 005237 002212          INC    FATFLG              ;BUMP COUNT
11491 065426 010001                   MOV    RO,R1               ;SAVE CONTENTS OF TSSR
11492 065430                   ERROF  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP    C0ERROF
                                .WORD   682
                                .WORD   WRTMSG
                                .WORD   SFIMSG
11493 065440 26# :  CKLOOP                   ;LOOP IF SELECTED
11494 065440 104406                   TRAP    C0CLP1
11495
11496          ;*****
11497          ;
11498          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11499          ;
11500          ;*****
11501 065442 004737 011104          JSR    PC,REWIND           ;CALL TAPE REWIND COMMAND
11502 065446 103413                   BCS    30#                 ;BR, IF NO PROBLEM
11503 065450 016501 000002          MOV    TSSR(R5),R1         ;GET TSSR
11504 065454 012702 000200          MOV    @SSR,R2            ;SET UP EXPECTED TSSR
11505 065460 010004                   MOV    RO,R4              ;PACKET ADDRESS SET UP
11506 065462 005237 002212          INC    FATFLG              ;BUMP COUNT
11510 065466                   ERROF  ERRNO,T26RMN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP    C0ERROF
                                .WORD   683
                                .WORD   T26RMN
                                .WORD   PKTSSR
065466 104456
065470 001253
065472 074464
065474 012136

```

TEST 6: REREADS

```

11511 065476 104406 304: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
11512 065476 104406
11513 ;*****
11514 ;
11515 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11516 ;
11517 ;*****
11518
11519 065500 013701 073030 MOV T26BFR+6,R1 ;PICK UP XSTO
11520 065504 010102 MOV R1,R2 ;SET UP EXPECTED
11521 065506 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
11522 065512 020102 CMP R1,R2 ;DOES EXP = REC'D
11523 065514 001406 BEQ 404 ;BR, IF EQUAL (OK)
11524 065516 005237 002212 INC FATFLG ;BUMP COUNT
11528 065522 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
065522 104456 TRAP C#ERHRD
065524 001254 .WORD 684
065526 C74175 .WORD T26BOT
065530 015564 .WORD EXPREC
11529 065532 404: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
065532 104406
11530 065534 012703 000400 MOV #256.,R3 ;RECORD SIZE
11531 065540 013737 003114 073132 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
11532 ;*****
11533 ;
11534 ;WRITE DATA,CVC=1,ACK COMMAND
11535 ;
11536 ;*****
11537
11538
11539 065546 012737 140005 073130 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
11540 065554 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
11541 065560
11542 065560 010337 073136 654: MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
11543 065564 013777 073156 115322 MOV T26CNT,#FREE ;MOVE TAPE RECORD NUMBER TO BUFFER
11544 065572 062737 000001 073156 ADD #1,T26CNT ;NUMBER READY FOR NEXT RECORD
11545 065600 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
11546 065604 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
11547 065610 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
11548 065614 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
11549 065620 020102 CMP R1,R2 ;ARE THEY EQUAL
11550 065622 001406 BEQ 754 ;BR, IF OK
11551 065624 005237 002212 INC FATFLG ;BUMP COUNT
11555 065630 ERRHRD ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
065630 104456 TRAP C#ERHRD
065632 001255 .WORD 685
065634 005111 .WORD WRTErr
065636 012136 .WORD PKTSSR
11556 065640 754: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
065640 104406
11557 065642 005723 TST (R3)+ ;BUMP THE RECORD SIZE
11558 065644 022703 000414 CMP #268.,R3 ;MAXIMUM SIZE YET
11559 065650 001401 BEQ 1204 ;BR, IF AT END OF WRITE SEQUENCE
11560 065652 000742 BR 654 ;WRITE MORE RECORDS
11561 065654
11562 065654 005037 073156 1204: CLR T26CNT ;SET RECORD COUNTER BACK TO ZERO
    
```

TEST 6: REREADS

```

11563
11564 ;*****
11565 ;
11566 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11567 ;
11568 ;*****
11569
11570 065660 004737 011104          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
11571 065664 103411                BCS      130#              ;BR, IF NO PROBLEM
11572 065666 016501 000002          MOV      TSSR(R5),R1       ;GET TSSR
11573 065672 010004                MOV      R0,R4            ;PACKET ADDRESS SET UP
11574 065674 005237 002212          INC      FATFLG           ;BUMP COUNT
11578 065700                ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    686
                                .WORD    T26RWN
                                .WORD    PKTSSR
                                TRAP      C#CLP1
11579 065710 104456                130# :  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                .WORD    104406
11580
11581 ;*****
11582 ;
11583 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11584 ;
11585 ;*****
11586
11587 065712 013701 073030          MOV      T268FR+6,R1      ;PICK UP XSTO
11588 065716 010102                MOV      R1,R2            ;SET UP EXPECTED
11589 065720 052702 000002          BIS      #BIT1,R2        ;SET BOT BIT IN EXPECTED
11590 065724 020102                CMP      R1,R2            ;DOES EXP = REC'D
11591 065726 001406                BEQ      135#              ;BR, IF EQUAL (OK)
11592 065730 005237 002212          INC      FATFLG           ;BUMP COUNT
11596 065734                ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    687
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
11597 065744                135# :  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                .WORD    104406
11598 065746 012737 000400 073162          MOV      #256.,T26RSZ     ;STARTING RECORD SIZE
11599 065754 000420                BR       140#              ;SKIP OVER THE SAPCE THIS TIME
11600
11601 ;*****
11602 ;
11603 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
11604 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
11605 ;
11606 ;*****
11607
11608 065756 012703 000001          132# :  MOV      #000001,R3 ;SET UP SPACE COMMAND (1 FORWARD)
11609 065762 004737 010556          JSR      PC,SPACE         ;CALL SPACE ROUTINE
11610 065766 103413                BCS      140#              ;BR, IF NO TROUBLE
11611 065770 016501 000002          MOV      TSSR(R5),R1     ;GET TSSR
11612 065774 012702 000200          MOV      #SSR,R2         ;SET UP EXPECTED TSSR
11613 066000 010004                MOV      R0,R4            ;PACKET ADDRESS SET UP
11614 066002 005237 002212          INC      FATFLG           ;BUMP COUNT
11618 066006                ERRHRD  ERRNO,T26SC,PKTSSR ;SPACE (FORWARD) FAILED

```







TEST 6: REREADS

```

066310 005054 .WORD WRTMSG
066312 012124 .WORD SFIMSG
11720 066314 104406 26#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066314 104406
11721
11722 ;*****
11723 ;
11724 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11725 ;
11726 ;*****
11727
11728 066316 004737 011104 JSR PC.REWIND ;CALL TAPE REWIND COMMAND
11729 066322 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
11730 066326 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
11731 066332 103407 BCS 30# ;BR, IF NO PROBLEM
11732 066334 010004 MOV RO,R4 ;PACKET ADDRESS SET UP
11733 066336 005237 002212 INC FATFLG ;BUMP COUNT
11737 066342 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
066342 104456 TRAP C#ERHRD
066344 001265 .WORD 693
066346 074464 .WORD T26RWN
066350 012136 .WORD PKTSSR
11738 066352 104406 30#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066352 104406
11739
11740 ;*****
11741 ;
11742 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11743 ;
11744 ;*****
11745
11746 066354 013701 073030 MOV T26BFR+6,R1 ;PICK UP XSTO
11747 066360 010102 MOV R1,R2 ;SET UP EXPECTED
11748 066362 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
11749 066366 020102 CMP R1,R2 ;DOES EXP = REC'D
11750 066370 001406 BEQ 40# ;BR, IF EQUAL (OK)
11751 066372 005237 002212 INC FATFLG ;BUMP COUNT
11755 066376 ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
066376 104456 TRAP C#ERHRD
066400 001266 .WORD 694
066402 074175 .WORD T26BOT
066404 015564 .WORD EXPREC
11756 066406 104406 40#: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066406 104406
11757 066410 012703 000400 MOV #256.,R3 ;RECORD SIZE
11758 066414 013737 003114 073132 MOV FREE,T26RB ;STARTING WRITE BUFFER ADDRESS
11759
11760 ;*****
11761 ;
11762 ;WRITE DATA,CVC=1,ACK COMMAND
11763 ;
11764 ;*****
11765
11766 066422 012737 140005 073130 MOV #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
11767 066430 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
11768 066434
11769 066434 010337 073136 65#: MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET

```

TEST 6: REREADS

```

11770 066440 013777 073156 114446      MOV      T26CNT,0FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
11771 066446 062737 000001 073156      ADD      #1,T26CNT        ;NUMBER READY FOR NEXT RECORD
11772 066454 010465 000000      MOV      R4,TSDB(R5)      ;ISSUE COMMAND
11773 066460 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
11774 066464 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
11775 066470 012702 000200      MOV      #SSR,R2         ;SET UP EXPECTED
11776 066474 020102      CMP      R1,R2           ;ARE THEY EQUAL
11777 066476 001406      BEQ      75#             ;BR, IF OK
11778 066500 005237 002212      INC      FATFLG          ;BUMP COUNT
11782 066504      ERRHRD  ERRNO,WRterr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERHRD
                                .WORD    695
                                .WORD    WRterr
                                .WORD    PKTSSR
11783 066514      75# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                .WORD    104406
11784 066516 005723      TST      (R3)+           ;BUMP THE RECORD SIZE
11785 066520 022703 000414      CMP      #268.,R3       ;MAXIMUM SIZE YET
11786 066524 001401      BEQ      120#           ;BR, IF AT END OF WRITE SEQUENCE
11787 066526 000742      BR       65#            ;WRITE MORE RECORDS
11788 066530      120# :
11789 066530 005037 073156      CLR      T26CNT          ;SET RECORD COUNTER BACK TO ZERO
11790
11791      ;*****
11792      ;
11793      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11794      ;
11795      ;*****
11796
11797 066534 004737 011104      JSR      PC,REWIND       ;CALL TAPE REWIND COMMAND
11798 066540 103411      BCS      130#           ;BR, IF NO PROBLEM
11799 066542 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
11800 066546 010004      MOV      R0,R4          ;PACKET ADDRESS SET UP
11801 066550 005237 002212      INC      FATFLG          ;BUMP COUNT
11805 066554      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    696
                                .WORD    T26RWN
                                .WORD    PKTSSR
11806 066564      130# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
                                .WORD    104406
11807
11808      ;*****
11809      ;
11810      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11811      ;
11812      ;*****
11813
11814 066566 013701 073030      MOV      T26BFR+6,R1     ;PICK UP XSTO
11815 066572 010102      MOV      R1,R2          ;SET UP EXPECTED
11816 066574 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
11817 066600 020102      CMP      R1,R2          ;DOES EXP = REC'D
11818 066602 001406      BEQ      135#           ;BR, IF EQUAL (OK)
11819 066604 005237 002212      INC      FATFLG          ;BUMP COUNT
11823 066610      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    697
066610 104456
066612 001271

```

TEST 6: REREADS

```

066614 074175 .WORD T26BOT
066616 015564 .WORD EXPREC
11824 066620 104406 000400 073162 135: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066620 104406 ;START RECORD SIZE
11825 066622 012737 000400 073162 MOV #256.,T26RSZ ;SKIP OVER SPACE
11826 066630 000420 BR 140#
11827
11828 ;*****
11829 ;
11830 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
11831 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
11832 ;
11833 ;*****
11834
11835 066632 012703 000001 136: MOV #000001,R3 ;SET UP SPACE COMMAND (1 FORWARD)
11836 066636 004737 010556 JSR PC,SPACE ;CALL SPACE ROUTINE
11837 066642 103413 BCS 140# ;BR, IF NO TROUBLE
11838 066644 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
11839 066650 C12702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
11840 066654 010004 MOV R0,R4 ;PACKET ADDRESS SET UP
11841 066656 005237 002212 INC FATFLG ;BUMP COUNT
11845 066662 ERRHRD ERRNO,T26SC,PKTSSR ;SPACE (FORWARD) FAILED
066662 104456 TRAP C#ERHRD
066664 001272 .WORD 698
066666 073577 .WORD T26SC
066670 012136 .WORD PKTSSR
11846 066672 104406 073162 140: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066672 104406 ;RECORD SIZE
11847 066674 013703 073162 MOV T26RSZ,R3 ;STARTING READ BUFFER ADDRESS
11848 066700 013737 003114 073132 150: MOV FREE,T26RB
11849
11850 ;*****
11851 ;
11852 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
11853 ;
11854 ;*****
11855
11856 066706 012737 161401 073130 165: MOV #161401,T26PK3 ;REREAD DATA,CVC=1,ACK, OPP COMMAND
11857 066714 012704 073130 MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
11858 066720 010337 073136 MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
11859 066724 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
11860 066730 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
11861 066734 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
11862 066740 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
11863 066744 020102 CMP R1,R2 ;ARE THEY EQUAL
11864 066746 001406 BEQ 170# ;BR, IF OK
11865 066750 005237 002212 INC FATFLG ;BUMP COUNT
11869 066754 ERRHRD ERRNO,T26RRF,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
066754 104456 TRAP C#ERHRD
066756 001273 .WORD 699
066760 073405 .WORD T26RRF
066762 012136 .WORD PKTSSR
11870 066764 104406 114122 170: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
066764 104406 ;FIRST WORD FROM READ BUFFER
11871 066766 017701 114122 MOV #FREE,R1 ;SET UP EXPECTED
11872 066772 013702 073156 MOV T26CNT,R2 ;IS TAPE POSITION CORRECT
11873 066776 020102 CMP R1,R2

```



TEST 6: REREADS

```

067116 104455                                TRAP    C#ERDF
067120 001275                                .WORD  701
067122 003650                                .WORD  SFIERR
067124 012124                                .WORD  SFIMSG
11928 067126 013737 002172 073020 20#:  MOV    UNITN,T26DSW          ;SET UP UNIT NUMBER
11929                                     ;
11930 067134 012704 073000                   MOV    #T26PACKET,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
11931                                     ;
11932                                     ;*****
11933                                     ;
11934                                     ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTPHR)
11935                                     ;
11936                                     ;*****
11937                                     ;
11938 067140 004737 010752                   JSR    PC,WRTPHR          ;ISSUE WRITE CHARACTERISTICS
11939 067144 103407                           BCS    26#                ;BR, IF COMMAND ISSUED OK
11940 067146 005237 002212                   INC    FATFLG             ;BUMP COUNT
11944 067152 010001                           MOV    R0,R1              ;SAVE CONTENTS OF TSSR
11945 067154                               ERRHRD  ERRNO,WRTPHR,SFIMSG ;WRITE CHARACTERISTICS FAILED
067154 104456                                TRAP    C#ERRRD
067156 001276                                .WORD  702
067160 005054                                .WORD  WRTPHR
067162 012124                                .WORD  SFIMSG
11946 067164                               26#:  CKLOOP                ;LOOP IF SELECTED
067164 104406                                TRAP    C#CLP1
11947                                     ;
11948                                     ;*****
11949                                     ;
11950                                     ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
11951                                     ;
11952                                     ;*****
11953                                     ;
11954 067166 004737 011104                   JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
11955 067172 016501 000002                   MOV    TSSR(R5),R1       ;GET TSSR
11956 067176 012702 000200                   MOV    #SSR,R2           ;SET UP EXPECTED TSSR
11957 067202 103407                           BCS    30#                ;BR, IF NO PROBLEM
11958 067204 010004                           MOV    R0,R4              ;PACKET ADDRESS SET UP
11959 067206 005237 002212                   INC    FATFLG             ;BUMP COUNT
11963 067212                               ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
067212 104456                                TRAP    C#ERRRD
067214 001277                                .WORD  703
067216 074464                                .WORD  T26RWN
067220 012136                                .WORD  PKTSSR
11964 067222                               30#:  CKLOOP                ;LOOP IF SELECTED
067222 104406                                TRAP    C#CLP1
11965                                     ;
11966                                     ;*****
11967                                     ;
11968                                     ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
11969                                     ;
11970                                     ;*****
11971                                     ;
11972 067224 013701 073030                   MOV    T26BFR+6,R1       ;PICK UP XSTO
11973 067230 010102                           MOV    R1,R2              ;SET UP EXPECTED
11974 067232 052702 000002                   BIS    #BIT1,R2          ;SET BOT BIT IN EXPECTED
11975 067236 020102                           CMP    R1,R2              ;DOES EXP = REC'D
11976 067240 001406                           BEQ    40#                ;BR, IF EQUAL (OK)

```

TEST 6: REREADS

```

11977 067242 005237 002212      INC    FATFLG      ;BUMP COUNT
11981 067246      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      067246 104456      TRAP    C#ERHRD
      067250 001300      .WORD  704
      067252 074175      .WORD  T26BOT
      067254 015564      .WORD  EXPREC
11982      40#:  CKLOOP      ;LOOP IF SELECTED      TRAP    C#CLP1
      067256 104406
11983 067260 012703 001000      MOV    #512.,R3      ;RECORD SIZE
11984 067264 013737 003114 073132      MOV    FREE,T26RB    ;STARTING WRITE BUFFER ADDRESS
11985
11986      ;*****
11987      ;
11988      ;WRITE DATA,CVC=1,ACK COMMAND
11989      ;
11990      ;*****
11991
11992 067272 012737 140005 073130      MOV    #140005,T26PK3 ;WRITE DATA,CVC=1,ACK COMMAND
11993 067300 C12704 073130      MOV    #T26PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
11994 067304
11995 067304 010337 073136      65#:  MOV    R3,T26SZ    ;SET UP RECORD SIZE IN PACKET
11996 067310 010465 000000      MOV    R4,TSD8(R5)   ;ISSUE COMMAND
11997 067314 004737 016340      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
11998 067320 016501 000002      MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
11999 067324 012702 000200      MOV    #SSR,R2       ;SET UP EXPECTED
12000 067330 020102      CMP    R1,R2         ;ARE THEY EQUAL
12001 067332 001406      BEQ    75#          ;BR, IF OK
12002 067334 005237 002212      INC    FATFLG      ;BUMP COUNT
12006 067340      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      067340 104456      TRAP    C#ERHRD
      067342 001301      .WORD  705
      067344 005111      .WORD  WRERR
      067346 012136      .WORD  PKTSSR
12007 067350      75#:  CKLOOP      ;LOOP IF SELECTED      TRAP    C#CLP1
      067350 104406
12008
12009      ;*****
12010      ;
12011      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12012      ;
12013      ;*****
12014
12015 067352 004737 011104      JSR    PC,REWIND     ;CALL TAPE REWIND COMMAND
12016 067356 016501 000002      MOV    TSSR(R5),R1   ;GET TSSR
12017 067362 012702 000200      MOV    #SSR,R2       ;SET UP EXPECTED TSSR
12018 067366 103407      BCS    130#         ;BR, IF NO PROBLEM
12019 067370 010004      MOV    R0,R4         ;PACKET ADDRESS SET UP
12020 067372 005237 002212      INC    FATFLG      ;BUMP COUNT
12024 067376      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
      067376 104456      TRAP    C#ERHRD
      067400 001302      .WORD  706
      067402 074464      .WORD  T26RWN
      067404 012136      .WORD  PKTSSR
12025 067406      130#: CKLOOP      ;LOOP IF SELECTED      TRAP    C#CLP1
      067406 104406
12026
12027      ;*****

```

TEST 6: REREADS

```

12028
12029 ; READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12030 ;
12031 ;*****
12032
12033 067410 013701 073030      MOV      T26BFR+6,R1      ; PICK UP XSTO
12034 067414 010102      MOV      R1,R2           ; SET UP EXPECTED
12035 067416 052702 000002      BIS      #BIT1,R2        ; SET BOT BIT IN EXPECTED
12036 067422 020102      CMP      R1,R2           ; DOES EXP = REC'D
12037 067424 001406      BEQ      140#           ; BR, IF EQUAL (OK)
12038 067426 005237 002212      INC      FATFLG          ; BUMP COUNT
12042 067432      ERRHRD  ERRNO,T26BOT,EXPREC ; TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    707
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C#CLP1
12043 067442      140# : CKLOOP      ; LOOP IF SELECTED
                                .WORD    707
                                .WORD    T26BOT
                                .WORD    EXPREC
12044 067444 104406      DEC      R3              ; SET RECORD SIZE TO 511.
12045 067446 005303 003114 073132      MOV      FREE,T26RB      ; STARTING READ BUFFER ADDRESS
12046
12047 ;*****
12048 ;
12049 ; REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
12050 ;
12051 ;*****
12052
12053 067454 012737 161401 073130      MOV      #161401,T26PK3  ; REREAD DATA,CVC=1,ACK,OPP=1 COMMAND
12054 067462 012704 073130      165# : MOV      #T26PK3,R4    ; SET UP R4 WITH PACKET ADDRESS
12055 067466 010337 073136      MOV      R3,T26SZ        ; SET UP RECORD SIZE IN PACKET
12056 067472 010465 000000      MOV      R4,TSD8(R5)     ; ISSUE COMMAND
12057 067476 004737 016340      JSR      PC,WAITF        ; WAIT FOR SSR TO SET
12058 067502 016501 000002      MOV      TSSR(R5),R1     ; GET TSSR CONTENTS
12059 067506 012702 100204      MOV      #SSR!SC!BIT2,R2 ; SET UP EXPECTED
12060 067512 020102      CMP      R1,R2           ; ARE THEY EQUAL
12061 067514 001406      BEQ      170#           ; BR, IF OK
12062 067516 005237 002212      INC      FATFLG          ; BUMP COUNT
12066 067522      ERRHRD  ERRNO,T26TRL,PKTSSR ; TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C#ERHRD
                                .WORD    708
                                .WORD    T26TRL
                                .WORD    PKTSSR
                                TRAP      C#CLP1
12067 067532      170# : CKLOOP      ; LOOP IF SELECTED
                                .WORD    708
                                .WORD    T26TRL
                                .WORD    PKTSSR
12068
12069 ;*****
12070 ;
12071 ; READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12072 ;
12073 ;*****
12074
12075 067534 013701 073030      MOV      T26BFR+6,R1     ; GET MESSAGE BUFFER
12076 067540 010102      MOV      R1,R2           ; SET UP EXPECTED
12077 067542 052702 010000      BIS      #BIT12,R2       ; SET THE RLL BIT IN EXPECTED
12078 067546 020102      CMP      R1,R2           ; ARE THEY EQUAL
12079 067550 001406      BEQ      180#           ; BR, IF EQUAL (ALL IS WELL)
12080 067552 005237 002212      INC      FATFLG          ; BUMP COUNT

```







TEST 6: REREADS

```

12187 070024          ERRHRD  ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTIC FAILED
      070024 104456          TRAP                  C#ERHRD
      070026 001311          .WORD                  713
      070030 005054          .WORD                  WRTMSG
      070032 012124          .WORD                  SFIMSG
12188 070034          26$:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C#CLP1
      070034 104406
12189
12190 ;*****
12191 ;
12192 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12193 ;
12194 ;*****
12195
12196 070036 004737 011104      JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
12197 070042 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR
12198 070046 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED TSSR
12199 070052 103407             BCS     30$                ;BR, IF NO PROBLEM
12200 070054 C10004             MOV      R0,R4             ;PACKET ADDRESS SET UP
12201 070056 005237 002212      INC      FATFLG            ;BUMP COUNT
12205 070062          ERRHRD  ERRNO,T26RWN,PKTSSR      ;REWIND NOT ACCEPTED
      070062 104456          TRAP                  C#ERHRD
      070064 001312          .WORD                  714
      070066 074464          .WORD                  T26RWN
      070070 012136          .WORD                  PKTSSR
12206 070072          30$:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C#CLP1
      070072 104406
12207
12208 ;*****
12209 ;
12210 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12211 ;
12212 ;*****
12213
12214 070074 013701 073030      MOV      T26BFR+6,R1       ;PICK UP XSTO
12215 070100 010102             MOV      R1,R2             ;SET UP EXPECTED
12216 070102 052702 000002      BIS     #BIT1,R2          ;SET BOT BIT IN EXPECTED
12217 070106 020102             CMP      R1,R2             ;DOES EXP = REC'D
12218 070110 001406             BEQ     40$                ;BR, IF EQUAL (OK)
12219 070112 005237 002212      INC      FATFLG            ;BUMP COUNT
12223 070116          ERRHRD  ERRNO,T26BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      070116 104456          TRAP                  C#ERHRD
      070120 001313          .WORD                  715
      070122 074175          .WORD                  T26BOT
      070124 015564          .WORD                  EXPREC
12224 070126          40$:  CKLOOP                    ;LOOP IF SELECTED          TRAP  C#CLP1
      070126 104406
12225 070130 012703 000400      MOV      #256.,R3         ;RECORD SIZE
12226 070134 013737 003114 073132  MOV      FREE,T26RB        ;STARTING WRITE BUFFER ADDRESS
12227
12228 ;*****
12229 ;
12230 ;WRITE DATA,CVC=1,ACK COMMAND
12231 ;
12232 ;*****
12233
12234 070142 012737 140005 073130  MOV      #140005,T26PK3    ;WRITE DATA,CVC=1,ACK COMMAND

```

TEST 6: REREADS

```

12235 070150 012704 073130          MOV    #T26PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
12236 070154          65#:    MOV    R3,T26SZ          ;SET UP RECORD SIZE IN PACKET
12237 070154 010337 073136          MOV    R4,TSDB(R5)       ;ISSUE COMMAND
12238 070160 010465 000000          JSR    PC,WAITF          ;WAIT FOR SSR TO SET
12239 070164 004737 016340          MOV    TSSR(R5),R1      ;GET TSSR CONTENTS
12240 070170 016501 000002          MOV    #SSR,R2          ;SET UP EXPECTED
12241 070174 012702 000200          CMP    R1,R2            ;ARE THEY EQUAL
12242 070200 020102          BEQ    75#              ;BR, IF OK
12243 070202 001406          INC    FATFLG           ;BUMP COUNT
12244 070204 005237 002212          ERRHRD ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
12248 070210          TRAP  C#ERRRD
12248 070210 104456          .WORD 716
12248 070212 001314          .WORD WRTERR
12248 070214 005111          .WORD PKTSSR
12248 070216 012136
12249 070220          75#:    CKLOOP          ;LOOP IF SELECTED
12249 070220 104406          TRAP  C#CLP1
12250 070222          120#:
12251
12252 ;*****\*****
12253 ;
12254 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12255 ;
12256 ;*****\*****
12257
12258 070222 004737 011104          JSR    PC,REWIND        ;CALL TAPE REWIND COMMAND
12259 070226 016501 000002          MOV    TSSR(R5),R1      ;GET TSSR
12260 070232 012702 000200          MOV    #SSR,R2          ;SET UP EXPECTED TSSR
12261 070236 103407          BCS    130#             ;BR, IF NO PROBLEM
12262 070240 010004          MOV    R0,R4            ;PACKET ADDRESS SET UP
12263 070242 005237 002212          INC    FATFLG           ;BUMP COUNT
12267 070246          ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
12267 070246 104456          TRAP  C#ERRRD
12267 070250 001315          .WORD 717
12267 070252 074464          .WORD T26RWN
12267 070254 012136          .WORD PKTSSR
12268 070256          130#:    CKLOOP          ;LOOP IF SELECTED
12268 070256 104406          TRAP  C#CLP1
12269
12270 ;*****\*****
12271 ;
12272 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12273 ;
12274 ;*****\*****
12275
12276 070260 013701 073030          MOV    T26BFR*6,R1      ;PICK UP XSTO
12277 070264 010102          MOV    R1,R2            ;SET UP EXPECTED
12278 070266 052702 000002          BIS    #BIT1,R2         ;SET BOT BIT IN EXPECTED
12279 070272 020102          CMP    R1,R2            ;DOES EXP = REC'D
12280 070274 001406          BEQ    135#             ;BR, IF EQUAL (OK)
12281 070276 005237 002212          INC    FATFLG           ;BUMP COUNT
12285 070302          ERRHRD ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
12285 070302 104456          TRAP  C#ERRRD
12285 070304 001316          .WORD 718
12285 070306 074175          .WORD T26BOT
12285 070310 015564          .WORD EXPREC
12286 070312          135#:    CKLOOP          ;LOOP IF SELECTED

```

TEST 6: REREADS

```

12287 070312 104406
12288 070314 012703 001000
12289 070320 013737 003114 073132
12290
12291
12292
12293
12294
12295
12296 070326 012737 161401 073130
12297 070334 012704 073130 165:
12298 070340 010337 073136
12299 070344 010465 000000
12300 070350 004737 016340
12301 070354 016501 000002
12302 070360 012702 100204
12303 070364 020102
12304 070366 001406
12305 070370 005237 002212
12309 070374
070374 104456
070376 001317
070400 075542
070402 012136
12310 070404
070404 104406
12311
12312
12313
12314
12315
12316
12317
12318 070406 013701 073030
12319 070412 010102
12320 070414 052702 040000
12321 070420 020102
12322 070422 001406
12323 070424 005237 002212
12327 070430
070430 104456
070432 001320
070434 075372
070436 015564
12328 070440
070440 104406
12329 070442 013701 073026
12330 070446 012702 000400
12331 070452 020102
12332 070454 001405
12336 070460
070460 104456
070462 001320
070464 075454
070466 015564
12337 070470

```

```

;*****
;
;REREAD NEXT,ACK,CVC=1,OPP=1
;*****
;REREAD NEXT,ACK,CVC=1,OPP=1
;SET UP R4 WITH PACKET ADDRESS
;SET UP RECORD SIZE IN PACKET
;ISSUE COMMAND
;WAIT FOR SSR TO SET
;GET TSSR CONTENTS
;SET UP EXPECTED
;ARE THEY EQUAL
;BR, IF OK
;BUMP COUNT
;TSSR INCORRECT AFTER READ DATA
TRAP C:ERHRD
.WORD 719
.WORD T26TRL
.WORD PKTSSR
;LOOP IF SELECTED
TRAP C:CLP1
;*****
;
;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
;*****
;GET MESSAGE BUFFER
;SET UP EXPECTED
;SET THE RLS BIT IN EXPECTED
;ARE THEY EQUAL
;BR, IF EQUAL (ALL IS WELL)
;BUMP COUNT
;THE RLL BIT WAS NOT SET IN XSTO
TRAP C:ERHRD
.WORD 720
.WORD T26LOP
.WORD EXPREC
;PICK UP RESIDUAL BYTE COUNTER
;THIS SHOULD BE THE DIFFERENCE
;IS THE DIFFERENCE CORRECT
;BR, IF CORRECT
;RBPCR NOT CORRECT
TRAP C:ERHRD
.WORD 720
.WORD T26PBP
.WORD EXPREC
;LOOP IF SELECTED

```

TEST 6: REREADS

```

12338 070470 104406          TRAP C#CLP1
12338 070472 012703 001000          MOV #512.,R3 ;RECORD SIZE
12339 070476 013737 003114 073132  MOV FREE,T26RB ;STARTING READ BUFFER ADDRESS
12340
12341 ;*****
12342 ;
12343 ;REREAD NEXT,ACK,CVC=1,OPP=0
12344 ;
12345 ;*****
12346
12347 070504 012737 141401 073130  MOV #141401,T26PK3 ;REREAD NEXT,ACK,CVC=1,OPP=0
12348 070512 012704 073130  MOV #T26PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
12349 070516 010337 073136  MOV R3,T26SZ ;SET UP RECORD SIZE IN PACKET
12350 070522 010465 000000  MOV R4,TSDB(R5) ;ISSUE COMMAND
12351 070526 004737 016340  JSR PC,WAITF ;WAIT FOR SSR TO SET
12352 070532 016501 000002  MOV TSSR(R5),R1 ;GET TSSR CONTENTS
12353 070536 012702 100204  MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
12354 070542 020102  CMP R1,R2 ;ARE THEY EQUAL
12355 070544 C01406  BEQ 270# ;BR, IF OK
12356 070546 005237 002212  INC FATFLG ;BUMP COUNT
12360 070552  ERRHRD ERRNO,T26TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
12360 070552 104456          TRAP C#ERHRD
12360 070554 001321          .WORD 721
12360 070556 075542          .WORD T26TRL
12360 070560 012136          .WORD PKTSSR
12361 070562 270#: CKLOOP ;LOOP IF SELECTED
12361 070562 104406          TRAP C#CLP1
12362
12363 ;*****
12364 ;
12365 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12366 ;
12367 ;*****
12368
12369 070564 013701 073030  MOV T26BFR+6,R1 ;GET MESSAGE BUFFER
12370 070570 010102  MOV R1,R2 ;SET UP EXPECTED
12371 070572 052702 040000  BIS #BIT14,R2 ;SET THE RLS BIT IN EXPECTED
12372 070576 020102  CMP R1,R2 ;ARE THEY EQUAL
12373 070600 001406  BEQ 280# ;BR, IF EQUAL (ALL IS WELL)
12374 070602 005237 002212  INC FATFLG ;BUMP COUNT
12378 070606  ERRHRD ERRNO,T26LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
12378 070606 104456          TRAP C#ERHRD
12378 070610 001322          .WORD 722
12378 070612 075372          .WORD T26LOP
12378 070614 015564          .WORD EXPREC
12379 070616 280#: CKLOOP
12379 070616 104406          TRAP C#CLP1
12380 070620 013701 073026  MOV T26BFR+4,R1 ;PICK UP RESIDUAL BYTE COUNTER
12381 070624 012702 000400  MOV #256.,R2 ;THIS SHOULD BE THE DIFFERENCE
12382 070630 020102  CMP R1,R2 ;IS THE DIFFERENCE CORRECT
12383 070632 001405  BEQ 290# ;BR, IF CORRECT
12387 070636  ERRHRD ERRNO,T26PBP,EXPREC ;RBPB NOT CORRECT
12387 070636 104456          TRAP C#ERHRD
12387 070640 001322          .WORD 722
12387 070642 075454          .WORD T26PBP
12387 070644 015564          .WORD EXPREC
12388 070646 290#: CKLOOP ;LOOP IF SELECTED

```



TEST 6: REREADS

```

12440 070764 103407          BCS      26#          ;BR, IF COMMAND ISSUED OK
12441 070766 005237 002212  INC      FATFLG      ;BUMP COUNT
12445 070772 010001          MOV      R0,R1       ;SAVE CONTENTS OF TSSR
12446 070774          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP      C#ERHRD
                                .WORD    724
                                .WORD    WRTMSG
                                .WORD    SFIMSG
                                TRAP      C#CLP1
12447 071004 104406 26# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
12448
12449 ;*****
12450 ;
12451 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12452 ;
12453 ;*****
12454
12455 071006 004737 021252      JSR      PC,INVERT    ;INVERT THE EXTENDED FEATURES SWITCH
12456 071012 004737 011104      JSR      PC,REWIND    ;CALL TAPE REWIND COMMAND
12457 071016 103411          BCS      30#          ;BR, IF NO PROBLEM
12458 071020 016501 000002      MOV      TSSR(R5),R1 ;GET TSSR
12459 071024 010004          MOV      R0,R4       ;PACKET ADDRESS SET UP
12460 071026 005237 002212  INC      FATFLG      ;BUMP COUNT
12464 071032          ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    725
                                .WORD    T26RWN
                                .WORD    PKTSSR
12465 071042 104406 30# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
12466
12467 ;*****
12468 ;
12469 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12470 ;
12471 ;*****
12472
12473 071044 013701 073030      MOV      T26BFR+6,R1 ;PICK UP XSTO
12474 071050 010102          MOV      R1,R2       ;SET UP EXPECTED
12475 071052 052702 000002      BIS      #BIT1,R2    ;SET BOT BIT IN EXPECTED
12476 071056 020102          CMP      R1,R2       ;DOES EXP = REC'D
12477 071060 001406          BEQ      40#          ;BR, IF EQUAL (OK)
12478 071062 005237 002212  INC      FATFLG      ;BUMP COUNT
12482 071066          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    726
                                .WORD    T26BOT
                                .WORD    EXPREC
12483 071076 104406 003114 073132 40# :   CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
12484 071100 013737          MOV      FREE,T26RB  ;STARTING WRITE BUFFER ADDRESS
12485
12486 ;*****
12487 ;
12488 ;WRITE DATA,CVC=1,ACK COMMAND
12489 ;
12490 ;*****

```

TEST 6: REREADS

```

12491
12492 071106 012737 140005 073130      MOV      #140005,T26PK3      ;WRITE DATA,CVC=1,ACK COMMAND
12493 071114 012704 073130      MOV      #T26PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
12494 071120 012737 000400 073136 65#:  MOV      #256.,T26SZ        ;SET UP RECORD SIZE IN PACKET
12495 071126 013777 073156 111760  MOV      T26CNT,#FREE      ;MOVE TAPE RECORD NUMBER TO BUFFER
12496 071134 062737 000001 073156  ADD      #1,T26CNT         ;NUMBER READY FOR NEXT RECORD
12497 071142 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
12498 071146 004737 016340      JSR      PC,WAITF          ;WAIT FOR SSR TO SET
12499 071152 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
12500 071156 012702 000200      MOV      #SSR,R2          ;SET UP EXPECTED
12501 071162 020102      CMP      R1,R2            ;ARE THEY EQUAL
12502 071164 001406      BEQ      75#              ;BR, IF OK
12503 071166 005237 002212      INC      FATFLG           ;BUMP COUNT
12507 071172      ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      C#ERHRD
                                .WORD    727
                                .WORD    WRTErr
                                .WORD    PKTSSR
12508 071202      75#:  CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
12509 071204 022737 000013 073156  CMP      #11.,T26CNT       ;CHECK NUMBER OF RECORDS WRITTEN
12510 071212 001401      BEQ      120#             ;BR, IF AT END OF WRITE SEQUENCE
12511 071214 000741      BR       65#              ;WRITE MORE RECORDS
12512 071216      120#: CLR      NXMHI              ;SET TO 16 BIT ADDRESS
12513 071216 005037 003132      125#: CLR      T26CNT       ;SET RECORD COUNTER BACK TO ZERO
12514 071222
12515 071222 005037 073156
12516
12517 ;*****
12518 ;
12519 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12520 ;
12521 ;*****
12522
12523 071226 004737 011104      JSR      PC,REWIND        ;CALL TAPE REWIND COMMAND
12524 071232 103411      BCS     130#             ;BR, IF NO PROBLEM
12525 071234 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR
12526 071240 010004      MOV      R0,R4           ;PACKET ADDRESS SET UP
12527 071242 005237 002212      INC      FATFLG         ;BUMP COUNT
12531 071246      ERRHRD  ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C#ERHRD
                                .WORD    728
                                .WORD    T26RWN
                                .WORD    PKTSSR
12532 071256      130#: CKLOOP              ;LOOP IF SELECTED
                                TRAP      L#CLP1
12533 071256 104406
12534 ;*****
12535 ;
12536 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12537 ;
12538 ;*****
12539
12540 071260 013701 073030      MOV      T26BFR+6,R1     ;PICK UP XSTO
12541 071264 010102      MOV      R1,R2          ;SET UP EXPECTED
12542 071266 052702 000002      BIS     #BIT1,R2         ;SET BOT BIT IN EXPECTED
12543 071272 020102      CMP     R1,R2           ;DOES EXP = REC'D

```



TEST 6: REREADS

```

12544 071274 001406          BEQ      140#          ;BR, IF EQUAL (OK)
12545 071276 005237 002212  INC      FATFLG      ;BUMP COUNT
12549 071302          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    729
                                .WORD    T26BOT
                                .WORD    EXPREC
12550 071312          140# :  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
12551 071314 012703 073146      MOV      #T26RN,R3    ;COMMAND BUFFER ADDRESS
12552 071320 013737 003130 073132 150# :  MOV      NXML0,T26RB  ;STARTING READ BUFFER ADDRESS
12553 071326 013737 003132 073134  MOV      NXMHI,T26RB+2 ;SET UP HIGH ORDER ADDRESS BITS
12554
12555 ;*****
12556 ;
12557 ;REREAD DATA,IE,ACK, OPP COMMAND
12558 ;
12559 ;*****
12560
12561 071334 011337 073130      MOV      (R3),T26PK3  ;REREAD DATA,IE,ACK, OPP COMMAND
12562 071340 012704 073130      MOV      #T26PK3,R4  ;SET UP R4 WITH PACKET ADDRESS
12563 071344 012737 000400 073136  MOV      #256.,T26SZ  ;SET UP RECORD SIZE IN PACKET
12564 071352 010465 000000      MOV      R4,TSD8(R5) ;ISSUE COMMAND
12565 071356 004737 016340      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
12566 071362 016501 000002      MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
12567 071366 012702 104210      MOV      #SSR!NXM!SC!BIT3,R2 ;SET UP EXPECTED
12568 071372 020102          CMP      R1,R2       ;ARE THEY EQUAL
12569 071374 001422          BEQ      170#        ;BR, IF OK
12570 071376 031327 001000      BIT      (R3),#BIT9  ;CHECK FOR A READ COMMAND
12571 071402 001403          BEQ      168#        ;BR, IF IT WAS A READ COMMAND
12572 071404 030127 000002      BIT      R1,#BIT1   ;WAS BIT1 SET
12573 071410 001014          BNE      170#        ;BR, IF REREAD AND BIT1 SET
12574 071412
12575 071412 005237 003132      168# :  INC      NXMHI       ;BUMP TO NEXT ADDRESS RANGE
12576 071416 023727 003132 000004  CMP      NXMHI,#4    ;CHECK FOR OVERFLOW
12577 071424 001276          BNE      125#        ;BR, IF MORE BITS TO GO
12578 071426 005237 002212      INC      FATFLG      ;BUMP COUNT
12582 071432          ERRHRD  ERRNO,T26RRF,PKTSSR ;TSSR INCORRECT AFTER REREAD DATA
                                TRAP      C#ERHRD
                                .WORD    730
                                .WORD    T26RRF
                                .WORD    PKTSSR
12583 071442          170# :  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
12584
12585 ;*****
12586 ;
12587 ;READ DATA, ACK,CVC-1 COMMAND
12588 ;
12589 ;*****
12590
12591 071444 012737 140001 073130      MOV      #140001,T26PK3 ;READ DATA, ACK,CVC-1 COMMAND
12592 071452 012737 000400 073136  MOV      #256.,T26SZ  ;SET SIZE INTO PACKET
12593 071460 005037 073134      CLR      T26RB+2     ;CLEAR OUT HIGH ADDRESS BITS
12594 071464 013737 003114 073132  MOV      FREE,T26RB  ;GIVE READ A GOOD BUFFER
12595 071472 010465 000000      MOV      R4,TSD8(R5) ;ISSUE READ DATA COMMAND
12596 071476 004737 016340      JSR      PC,WAITF    ;WAIT FOR SSR

```

TEST 6: REREADS

```

12597 071502 016501 000002      MOV      TSSR(R5),R1      ;PICK UP THE TSSR
12598 071506 012702 000200      MOV      #SSR,R2        ;SET UP EXPECTED
12599 071512 020102              CMP      R1,R2          ;IS THE TSSR OK
12600 071514 001406              BEQ      180$           ;BR, IF TSSR OK (GOOD)
12601 071516 005237 002212      INC      FATFLG         ;BUMP COUNT
12605 071522              ERRHRD  ERRNO, RDERR, PKTSSR ;READ DATA COMMAND FAILED
                                TRAP
                                .WORD  C$ERHRD
                                .WORD  731
                                .WORD  PDERR
                                .WORD  PKTSSR
12606 071532              180$: CKLOOP           ;LOOP IF SELECTED
                                TRAP  C$CLP1
12607 071534 017701 111354      MOV      #FREE,R1       ;FIRST WORD FROM READ BUFFER
12608 071540 012702 000001      MOV      #1,R2          ;SET UP EXPECTED
12609 071544 020102              CMP      R1,R2          ;IS TAPE POSITION CORRECT
12610 071546 001406              BEQ      190$           ;KEEP GOING POSITION OK
12611 071550 005237 002212      INC      FATFLG         ;BUMP COUNT
12615 071554              ERRHRD  ERRNO, T26WNG, EXPREC ;TAPE POSITION INCORRECT
                                TRAP  C$ERHRD
                                .WORD  732
                                .WORD  T26WNG
                                .WORD  EXPREC
12616 071564              190$: CKLOOP           ;LOOP IF SELECTED
                                TRAP  C$CLP1
12617
12618 ;*****
12619 ;
12620 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12621 ;
12622 ;*****
12623
12624 071566 004737 011104      JSR      PC,REWIND       ;CALL TAPE REWIND COMMAND
12625 071572 103411              BCS      194$           ;BR, IF NO PROBLEM
12626 071574 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR
12627 071600 010004              MOV      R0,R4          ;PACKET ADDRESS SET UP
12628 071602 005237 002212      INC      FATFLG         ;BUMP COUNT
12632 071606              ERRHRD  ERRNO, T26RWN, PKTSSR ;REWIND NOT ACCEPTED
                                TRAP  C$ERHRD
                                .WORD  733
                                .WORD  T26RWN
                                .WORD  PKTSSR
12633 071616              194$: CKLOOP           ;LOOP IF SELECTED
                                TRAP  C$CLP1
12634
12635 ;*****
12636 ;
12637 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12638 ;
12639 ;*****
12640
12641 071620 013701 073030      MOV      T26BFR+6,R1    ;PICK UP XSTO
12642 071624 010102              MOV      R1,R2          ;SET UP EXPECTED
12643 071626 052702 000002      BIS      #BIT1,R2       ;SET BOT BIT IN EXPECTED
12644 071632 020102              CMP      R1,R2          ;DOES EXP = REC'D
12645 071634 001406              BEQ      196$           ;BR, IF EQUAL (OK)
12646 071636 005237 002212      INC      FATFLG         ;BUMP COUNT
12650 071642              ERRHRD  ERRNO, T26BOT, EXPREC ;TAPE NOT AT BOT AFTER REWIND

```





TEST 6: REREADS

```

12751 072060 013701 073030      MOV      T26BFR+6,R1      ;PICK UP XSTO
12752 072064 010102      MOV      R1,R2           ;SET UP EXPECTED
12753 072066 052702 000002      BIS      @BIT1,R2        ;SET BOT BIT IN EXPECTED
12754 072072 020102      CMP      R1,R2           ;DOES EXP = REC'D
12755 072074 001406      BEQ      40$             ;BR, IF EQUAL (OK)
12756 072076 005237 002212      INC      FATFLG          ;BUMP COUNT
12760 072102      ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD    738
                                .WORD    T26BOT
                                .WORD    EXPREC
                                TRAP      C$CLP1
12761 072112      40$:   CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
12762 072114 012737 000400 073136      MOV      @256.,T26SZ     ;SET UP RECORD SIZE IN PACKET
12763 072122 013737 003114 073132      MOV      FREE,T26RB      ;ADDRESS OF READ BUFFER
12764 072130 005703      TST      R3              ;CHECK NUMBER OF TIMES THROUGH HERE
12765 072132 001404      BEQ      50$             ;BR, IF FIRST TIME THROUGH HERE
12766
12767      ;*****
12768      ;
12769      ;REREAD,CVC=1,ACK COMMAND
12770      ;
12771      ;*****
12772
12773 072134 012737 161001 073130      MOV      @161001.T26PK3 ;REREAD,CVC=1,ACK COMMAND
12774 072142 000403      BR       55$             ;SKIP NEXT COMMAND
12775
12776      ;*****
12777      ;
12778      ;REREAD,ACK COMMAND
12779      ;
12780      ;*****
12781
12782 072144 012737 141001 073130 50$:   MOV      @141001.T26PK3 ;REREAD,ACK COMMAND
12783 072152 012704 073130 55$:   MOV      @T26PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
12784 072156 010465 000000 65$:   MOV      R4,TSDB(R5)     ;ISSUE COMMAND
12785 072156 004737 016340      JSR      PC,WAITF        ;WAIT FOR SSR TO SET
12786 072162 016501 000002      MOV      TSSR(R5),R1     ;GET TSSR CONTENTS
12787 072166 012702 100206      MOV      @SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
12788 072172 020102      CMP      R1,R2           ;ARE THEY EQUAL
12789 072176 001406      BEQ      75$             ;BR, IF OK
12790 072200 005237 002212      INC      FATFLG          ;BUMP COUNT
12791 072202 001406      ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
12792 072206      TRAP      C$ERHRD
                                .WORD    739
                                .WORD    T26WDE
                                .WORD    PKTSSR
                                TRAP      C$CLP1
12796 072206 104456
12797 072210 001343
12797 072212 074123
12797 072214 012136
12797 072216 104406      75$:   CKLOOP           ;LOOP IF SELECTED
                                TRAP      C$CLP1
12798
12799      ;*****
12800      ;
12801      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12802      ;
12803      ;*****

```



TEST 6: REREADS

```

072334 104455 TRAP C#ERDF
072336 001345 .WORD 741
072340 003650 .WORD SFIERR
072342 012124 .WORD SFIMSG
12858 072344 013737 002172 073020 20$: MOV UNITN,T26DSW ;SET UP UNIT NUMBER
12859
12860 072352 012704 073000 MOV #T26PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
12861
12862 ;*****
12863 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
12864 ;
12865 ;*****
12866
12867
12868 072356 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
12869 072362 103407 BCS 25$ ;BR, IF COMMAND ISSUED OK
12870 072364 005237 002212 INC FATFLG ;BUMP COUNT
12874 072370 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
12875 072372 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
072372 104456 TRAP C#ERHRD
072374 001346 .WORD 742
072376 005054 .WORD WRTMSG
072400 012124 .WORD SFIMSG
12876 072402 25$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
072402 104406
12877
12878 ;*****
12879 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
12880 ;
12881 ;*****
12882
12883
12884 072404 004737 011104 26$: JSR PC,REWIND ;CALL TAPE REWIND COMMAND
12885 072410 016501 000002 MOV TSSR(R5),R1 ;GET TSSR
12886 072414 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED TSSR
12887 072420 103407 BCS 30$ ;BR, IF NO PROBLEM
12888 072422 010004 MOV RO,R4 ;PACKET ADDRESS SET UP
12889 072424 005237 002212 INC FATFLG ;BUMP COUNT
12893 072430 ERRHRD ERRNO,T26RWN,PKTSSR ;REWIND NOT ACCEPTED
072430 104456 TRAP C#ERHRD
072432 001347 .WORD 743
072434 074464 .WORD T26RWN
072436 012136 .WORD PKTSSR
12894 072440 30$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
072440 104406
12895
12896 ;*****
12897 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
12898 ;
12899 ;*****
12900
12901
12902 072442 013701 073030 MOV T26BFR+6,R1 ;PICK UP XSTO
12903 072446 010102 MOV R1,R2 ;SET UP EXPECTED
12904 072450 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
12905 072454 020102 CMP R1,R2 ;DOES EXP = REC'D
12906 072456 001406 BEQ 40$ ;BR, IF EQUAL (OK)

```

TEST 6: REREADS

```

12907 072460 005237 002212          INC    FATFLG          ;BUMP COUNT
12911 072464          ERRHRD  ERRNO,T26BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          072464 104456          TRAP    C#ERHRD
          072466 001350          .WORD  744
          072470 074175          .WORD  T26BOT
          072472 015564          .WORD  EXPREC
12912 072474          40#:   CKLOOP          TRAP    C#CLP1
          072474 104406
12913
12914          ;*****
12915          ;
12916          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
12917          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
12918          ;
12919          ;*****
12920
12921 072476 012703 000001          MOV    #000001,R3          ;SET UP SPACE FORWARD 1 RECORD
12922 072502 004737 010556          JSR    PC,SPACE          ;ISSUE SPACE COMMAND
12923 072506 103411          BCS    75#              ;BR, IF OK
12924 072510 016501 000002          MOV    TSSR(R5),R1       ;GET STATUS DATA
12925 072514 010004          MOV    R0,R4            ;GET PACKET ADDRESS
12926 072516 005237 002212          INC    FATFLG          ;BUMP COUNT
12930 072522          ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          072522 104456          TRAP    C#ERHRD
          072524 001351          .WORD  745
          072526 074123          .WORD  T26WDE
          072530 012136          .WORD  PKTSSR
12931 072532          75#:   CKLOOP          ;LOOP IF SELECTED
          072532 104406          TRAP    C#CLP1
12932
12933          ;*****
12934          ;
12935          ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
12936          ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
12937          ;
12938          ;*****
12939
12940 072534 012703 100001          MOV    #100001,R3       ;SET SPACE REVERSE 1 RECORD
12941 072540 004737 010556          JSR    PC,SPACE          ;ISSUE COMMAND
12942 072544 103411          BCS    175#            ;GO ON IF ALL IS WELL
12943 072546 016501 000002          MOV    TSSR(R5),R1       ;GET TSSR CONTENTS
12944 072552 010004          MOV    R0,R4            ;SET UP EXPECTED (PACKET CONTENTS)
12945 072554 005237 002212          INC    FATFLG          ;BUMP COUNT
12949 072560          ERRHRD  ERRNO,T26WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
          072560 104456          TRAP    C#ERHRD
          072562 001352          .WORD  746
          072564 074123          .WORD  T26WDE
          072566 012136          .WORD  PKTSSR
12950 072570          175#:  CKLOOP          ;LOOP IF SELECTED
          072570 104406          TRAP    C#CLP1
12951 072572 013737 003114 073132          MOV    FREE,T26RB       ;ADDRESS OF BUFFER
12952 072600 005737 073160          TST    T26CNU          ;CHECK FOR TIMES THROUGH HERE
12953 072604 001404          BEQ    176#            ;BR, IF FIRST TIME THROUGH
12954
12955          ;*****
12956          ;
12957          ;REREAD (PREVIOUS),IE,ACK,OPP=1 CMD.

```





## TEST 6: REREADS

```

13009 072764
13010 072764
      072764 104432
      072766 003116
13011
13012
13013
13014
13016 072770
13018 073000
13019 073000 014004
13020 073002 073010
13021 073004 000000
13022 073006 000012
13023 073010
13024 073010 073022
13025 073012 000000
13026 073014 000024
13027 073016 C00000
13028 073020 000000
13029 073022
13030
13031
13032
13034 073104
13036 073110
13037 073110 100006
13038 073112 073140
13039 073114 000000
13040 073116 000006
13041
13043 073120
13045 073130
13046 073130 140005
13047 073132
13048 073132 003114
13049 073134 000000
13050 073136 000000
13051
13052
13053
13054
13055 073140
13056 073140 010
13057 073141 200
13058 073142 000000
13059 073144 000000
13060
13061
13062
13063
13064
13065 073146 140001
13066 073150 141401
13067 073152 161401
13068 073154 177777
13069

```

```

1634:  EXIT  TST
; ALL DONE THIS TEST
      TRAP  CEXIT
      .WORD L10102-
;
; LOCAL STORAGE FOR THIS TEST
;
; T26PACKET:
; .BLKB  10-<.-TSV2&7>
; .WORD  14004
; .WORD  T26DATA
; .WORD  0
; .WORD  10.
; T26DATA:
; .WORD  T26BFR
; .WORD  0
; .WORD  20.
; .WORD  0
; T26DSW: .WORD  0
; T26BFR: .BLKW  25.
;
; WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
; T26PK2:
; .BLKB  10-<.-TSV2&7>
; .WORD  100006
; .WORD  T26BF2
; .WORD  0
; .WORD  6.
;
; T26PK3:
; .BLKB  10-<.-TSV2&7>
; .WORD  140005
; T26RB:
; T26WB: .WORD  FREE
; .WORD  0
; T26SZ: .WORD  0
; .EVEN
;
; T26BF2:
; T26BS0: .BYTE  10
; T26BS1: .BYTE  200
; T26S2: .WORD  0
; T26S3: .WORD  0
;
; .EVEN
; TAPE MOTION PACKET COMMAND VALUES
; T26RN: .WORD  140001
; .WORD  141401
; .WORD  161401
; .WORD  177777
;
; COMMAND PACKET FOR TEST
; WRITE CHARACTERISTICS COMMAND, WITH CVC=1. ACK
; ADDRESS OF CHARACTERISTICS BLOCK
;
; STARTING VALUE OF BLOCK SIZE
; CHARACTERISTICS DATA BLOCK
; ADDRESS OF MESSAGE BUFFER
;
; LENGTH OF MESSAGE BUFFER
;
; SELECT DRIVE 0
; MESSAGE BUFFER
;
; WRITE SUB SYS MEM COMMAND, AND ACK
; ADDRESS OF SELECT BLOCK DATA
;
; SIZE OF DATA PACKET
;
; REREAD COMMAND, CVC=1 AND ACK
; ADDRESS OF WRITE BUFFER
;
; SIZE OF BUFFER (EXTENT)
;
; BSEL0 AREA
; BSEL1 AREA
; SEL 2 AREA
; DATA AREA
;
; READ DATA
; REREAD NEXT OPP=0
; REREAD NEXT OPP=1
; END OF DATA

```

## TEST 6: REREADS

```

13070
13071 073156 000000      T26CNT: .WORD 0      ;TAPE RECORD COUNTER STORAGE AREA
13072 073160 000000      T26CNU: .WORD 0      ;TAPE RECORD COUNTER STORAGE AREA
13073
13074 073162 000000      T26RSZ: .WORD 0      ;RECORD STORAGE SIZE AREA
13075
13076 073164 000000      T26DLY: .WORD 0      ;DELAY COUNTER AREA
13077
13078
13079
13080      ;*
13081      ;LOCAL TEXT MESSAGES FOR TEST
13082      ;-
13083
13084 073166      124      141      160      T26WNG: .ASCIZ 'Tape Position Incorrect After REREAD Previous (OPP=1)'
13085 073254      122      105      122      T26NEF: .ASCIZ 'REREAD PREVIOUS, At BOT, failed To Set NEF (XSTO)'
13086 073336      124      123      123      T26RDF: .ASCIZ 'TSSR Incorrect After READ DATA Command'
13087 073405      122      105      122      T26RRF: .ASCIZ 'REREAD Previous (Space Reverse, Read Forward) Command Failed'
13088 073502      122      105      122      T26RRG: .ASCIZ 'REREAD Previous (Read Reverse, Space Forward) Command Failed'
13089 073577      120      117      123      T26SC: .ASCIZ 'POSITION (Space Command) Failed, TSSR Not Correct'
13090 073661      122      111      102      T26LOR: .ASCIZ 'RIB NOT SET AFTER READ REVERSE INTO BOT'
13091 073731      124      123      123      T26WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
13092 074006      111      154      154      T26LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
13093 074067      122      105      122      T26SSR: .ASCIZ 'REREAD COMMAND Not Accepted'
13094 074123      124      123      123      T26WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
13095 074175      124      141      160      T26BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
13096 074242      104      141      164      T26DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
13097 074330      122      105      122      T26EOT: .ASCIZ 'REREAD DATA OVER EOT GAVE NO TAPE STATUS ALERT'
13098 074407      124      123      123      T26TM: .ASCIZ 'TSSR Not Correct After REREAD COMMAND Reject'
13099 074464      122      145      167      T26RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
13100 074533      122      101      115      T26RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
13101 074606      124      123      123      T26AM3: .ASCIZ 'TSSR Init. Failed After REREAD COMMAND'
13102 074655      104      162      151      T26OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
13103 074730      124      123      123      T26WDD: .ASCIZ 'TSSR Not Correct After REREAD DATA Command, SWB Bit Set'
13104 075020      124      123      123      T26WDC: .ASCIZ 'TSSR Not Correct After REREAD DATA Command'
13105 075073      103      126      103      T26VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
13106 075146      124      123      102      T268A: .ASCIZ 'TSBA Not Correct After REREAD DATA Command'
13107 075221      127      122      111      T26WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
13108 075310      122      145      141      T26LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XSTO'
13109 075372      122      145      141      T26LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XSTO'
13110 075454      122      145      163      T26PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
13111 075542      122      145      141      T26TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
13112 075630      104      141      164      T26NEQ: .ASCIZ 'Data REREAD From Tape Not Correct, After SWB=1'
13113 075707      122      145      162      TST26ID: .ASCIZ 'Rereads'
13114
13115
13116
13117      ;*
13118      ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
13119      ;WRITE SUBSYSTEM MEMORY COMMAND
13120      ;-
13121
13122 075720      T26REST:
13123 075720      SAVREG      ;SAVE THE REGISTERS
13124 075724      012701 073000      MOV      #T26PACKET,R1      ;START OF THE PACKET
13125 075730      012721 140004      MOV      #140004,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK, CVC=1
13126 075734      012721 073010      MOV      #T26DATA,(R1)+      ;ADDRESS OF CHARAISTICS DATA BLOCK

```

## TEST 6: REREADS

```

13127 075740 005021          CLR      (R1)+          ;EXTENDED ADDRESS
13128 075742 012721 000012  MOV      #10.,(R1)+    ;SIZE OF DATA BLOCK IN BYTES
13129 075746 012721 073022  MOV      #T26BFR,(R1)+;ADDRESS OF MESSAGE BUFFER
13130 075752 005021          CLR      (R1)+
13131 075754 012721 000024  MOV      #20.,(R1)+    ;LENGTH OF MESSAGE BUFFER
13132 075760 005021          CLR      (R1)+
13133 075762 012711 000000  MOV      #0,(R1)       ;SELECT DRIVE ZERO (0)
13134 075766 012702 000030  MOV      #24.,R2       ;NUMBER OF LOCATIONS TO BE CLEARED
13135 075772 012762 177777 073022 64#: MOV      #177777.T26BFR(R2);ALL ONES TO MESSAGE BUFFER
13136 076000 005742          TST      -(R2)         ;NEXT LOCATION
13137 076002 020227 000000  CMP      R2,#0         ;CHECK FOR END OF LOOP
13138 076006 001371          BNE      64#          ;KEEP GOING UNTIL DONE
13139 076010 000207          RTS      PC           ;RETURN
13140
13141
13142 076012          T26RT2:
13143 076012          SAVREG
13144 076016 012701 073110  MOV      #T26PK2,R1    ;SAVE THE REGISTERS
13145 076022 012721 140006  MOV      #140006.(R1)+;START OF THE PACKET
13146 076026 012721 073140  MOV      #T26BF2,(R1)+;WRITE SUBSYSTEM MEM. WITH ACK,CVC-1.
13147 076032 005021          CLR      (R1)+        ;ADDRESS OF DATA BLOCK
13148 076034 012721 000006  MOV      #6.,(R1)+    ;EXTENDED ADDRESS
13149 076040 005021          CLR      (R1)+        ;SIZE OF DATA BLOCK IN BYTES
13150 076042 012701 073140  MOV      #T26BF2,R1    ;POINT TO DATA SEL AREA
13151 076046 005021          CLR      (R1)+
13152 076050 005011          CLR      (R1)
13153 076052 000207          RTS      PC           ;RETURN
13154 076054          T26RT3:
13155 076054          SAVREG
13156 076060 012701 073130  MOV      #T26PK3,R1    ;SAVE THE REGISTERS
13157 076064 012721 000000  MOV      #0,(R1)+     ;START OF THE PACKET
13158 076070 012721 000000  MOV      #0,(R1)+     ;WRITE SUBSYSTEM MEM. WITH ACK.
13159 076074 005021          CLR      (R1)+        ;ADDRESS OF DATA BLOCK
13160 076076 012711 000000  MOV      #0,(R1)+    ;EXTENDED ADDRESS
13161 076102 000207          RTS      PC           ;SIZE OF DATA BLOCK IN BYTES
13162 076104          ENDTST          ;RETURN
076104          L10102: TRAP      C#ETST
076104 104401
13163
13164          .SBTTL TEST 7: WRITE DATA RETRY
13165          ;+
13166          ;
13167          ;THIS TEST VERIFIES PROPER OPERATION OF THE WRITE DATA RETRY
13168          ;COMMAND (SPACE REVERSE, ERASE, WRITE DATA)
13169          ;
13170          ;
13171          ;THE TEST CONSISTS OF THE FOLLOWING 5 SUBTESTS
13172          ;
13173          ;
13174          ;
13175          ;-
13176 076106          BGNTST
076106
13177 076106 012737 006446 002170  MOV      #EPRT2,EPRTSW ;SECONDARY ERROR MESSAGE
13178 076114 005037 003124          CLR      KTENABLE    ;TURN OFF KT11
13179 076120 004737 017364          JSR      PC,KTOFF     ;TURN KT11 BACK OFF IF THERE
13184 076124 012700 105743          MOV      #TST27ID,RO  ;ASCII MESSAGE TO IDENTIFY TEST

```



TEST 7: WRITE DATA RETRY

```

13231 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
13232 ;
13233 ;*****
13234
13235 076272 004737 010752 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
13236 076276 103407 BCS 25# ;BR, IF COMMAND ISSUED OK
13237 076300 005237 002212 INC FATFLG ;BUMP COUNT
13241 076304 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
13242 076306 ERRHRD ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTICS FAILED
; TRAP C#ERHRD
; .WORD 702
; .WORD WRTMSG
; .WORD SFMSG
13243 076316 104406 25#: CKLOOP ;LOOP IF SELECTED
; TRAP C#CLP1
13244
13245 ;*****
13246 ;
13247 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
13248 ;
13249 ;*****
13250
13251 076320 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
13252 076324 103407 BCS 30# ;BR, IF NO PROBLEM
13253 076326 010004 MOV R0,R4 ;SET UP REWIND PACKET ADDRESS
13254 076330 005237 002212 INC FATFLG ;BUMP COUNT
13258 076334 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
; TRAP C#ERHRD
; .WORD 703
; .WORD T27RWN
; .WORD PKTSSR
13259 076344 104406 30#: CKLOOP ;LOOP IF SELECTED
; TRAP C#CLP1
13260
13261 ;*****
13262 ;
13263 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERC (XSTO)
13264 ;
13265 ;*****
13266
13267 076346 013701 102660 MOV T27BFR+6,R1 ;PICK UP XSTO
13268 076352 010102 MOV R1,R2 ;SET UP EXPECTED
13269 076354 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
13270 076360 020102 CMP R1,R2 ;DOES EXP = REC'D
13271 076362 001406 BEQ 40# ;BR, IF EQUAL (OK)
13272 076364 005237 002212 INC FATFLG ;BUMP COUNT
13276 076370 ERRHRD ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
; TRAP C#ERHRD
; .WORD 704
; .WORD T27BOT
; .WORD EXPREC
13277 076400 104406 40#: CKLOOP ;LOOP IF SELECTED
; TRAP C#CLP1
13278 076402 012737 000400 102766 MOV #256.,T27SZ ;SET UP RECORD SIZE
13279 076410 013737 003114 102762 MOV FREE,T27WB ;ADDRESS OF WRITE BUFFER
13280
13281 ;*****

```

TEST 7: WRITE DATA RETRY

```

13282
13283 ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
13284 ;
13285 ;*****
13286
13287 076416 012737 141005 102760      MOV      #141005,T27PK3      ;WRITE DATA RETRY,ACK,CVC=1 COMMAND
13288 076424 012704 102760      MOV      #T27PK3,R4        ;SET UP R4 WITH PACKET ADDRESS
13289 076430 010465 000000      MOV      R4,TSDB(R5)       ;ISSUE COMMAND
13290 076434 004737 016340      JSR      PC,WAITF         ;WAIT FOR SSR TO SET
13291 076440 016501 000002      MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
13292 076444 012702 100206      MOV      #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
13293 076450 020102              CMP      R1,R2             ;ARE THEY EQUAL
13294 076452 001406              BEQ      75#               ;BR, IF OK
13295 076454 005237 002212      INC      FATFLG            ;BUMP COUNT
13299 076460              ERRHRD  ERRNO,T27WDE,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      C#ERHRD
                                .WORD    705
                                .WORD    T27WDE
                                .WORD    PKTSSR
                                TRAP      C#CLP1
13300 076470 104406      75# :   CKLOOP              ;LOOP IF SELECTED
                                TRAP      C#CLP1
13301
13302 ;*****
13303 ;
13304 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13305 ;
13306 ;*****
13307
13308 076472 013701 102660      MOV      T27BFR+6,R1       ;GET XSTO STATUS WORD
13309 076476 010102              MOV      R1,R2             ;SET UP EXPECTED
13310 076500 052702 002000      BIS      #BIT10,R2         ;SET THE NEF BIT
13311 076504 020102              CMP      R1,R2             ;ARE THEY EQUAL
13312 076506 001406              BEQ      170#              ;BR, IF EQUAL (GOOD)
13313 076510 005237 002212      INC      FATFLG            ;BUMP COUNT
13317 076514              ERRHRD  ERRNO,T27NEF,EXPREC ;NEF SHOULD BE SET
                                TRAP      C#ERHRD
                                .WORD    706
                                .WORD    T27NEF
                                .WORD    EXPREC
                                TRAP      C#CLP1
13318 076524 104406      170# :  CKLOOP
13319 076526              ENDSUB
                                L10123:  TRAP      C#ESUB
13320 076530 023727 002212 000017      CMP      FATFLG,#15.       ;IS ERROR COUNT AT 25
13321 076536 103402              BLO     999#               ;BR, IF LESS THAN 25
13322 076540 004737 017272      JSR      PC,CKDROP         ;TRY TO DROP THE UNIT
13323 076544              999# :
13324
13325 ;
13326 ;
13327 ;TEST 7. SUBTEST 2
13328 ;
13329 ;VERIFIES THAT WRITE DATA RETRY COMMAND ISSUED WHILE
13330 ;THE TAPE IS POSITIONED BEFORE THE FIRST RECORD ON
13331 ;TAPE (BUT NOT AT BOT) RESULTS IN TAPE STATUS ALERT
13332 ;TERMINATION, WITH THE REVERSE INTO BOT (RIB) STATUS

```





TEST 7: WRITE DATA RETRY

```

13385 076656 016501 000002      MOV     TSSR(R5),R1      ;GET TSSR CONTENTS
13386 076662 005237 002212      INC     FATFLG          ;BUMP COUNT
13390 076666      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      076666 104456      TRAP   C#ERHRD
      076670 001305      .WORD 709
      076672 104165      .WORD T27RWN
      076674 012136      .WORD PKTSSR
13391 076676      26#:  CKLOOP          ;LOOP IF SELECTED      TRAP   C#CLP1
      076676 104406
13392 076700 012703 000400      MOV     #256.,R3      ;STARTING RECORD SIZE
13393 076704 013737 003114 102762      MOV     FREE,T27WB    ;STARTING WRITE BUFFER ADDRESS
13394
13395      ;*****
13396      ;
13397      ;WRITE DATA,CVC=1,ACK COMMAND
13398      ;
13399      ;*****
13400
13401 076712 012737 140005 102760      MOV     #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
13402 076720 012704 102760      MOV     #T27PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
13403 076724 010337 102766      MOV     R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
13404 076730 010465 000000      MOV     R4,TSD8(R5)   ;ISSUE COMMAND
13405 076734 004737 016340      JSR    PC,WAITF       ;WAIT FOR SSR TO SET
13406 076740 016501 000002      MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
13407 076744 012702 000200      MOV     #SSR,R2      ;SET UP EXPECTED
13408 076750 020102      CMP     R1,R2         ;ARE THEY EQUAL
13409 076752 001406      BEQ    28#           ;BR, IF OK
13410 076754 005237 002212      INC     FATFLG        ;BUMP COUNT
13414 076760      ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      076760 104456      TRAP   C#ERHRD
      076762 001306      .WORD 710
      076764 005111      .WORD WRERR
      076766 012136      .WORD PKTSSR
13415 076770      28#:  CKLOOP          ;LOOP IF SELECTED      TRAP   C#CLP1
      076770 104406
13416
13417      ;*****
13418      ;
13419      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
13420      ;
13421      ;*****
13422
13423 076772 004737 011104      JSR    PC,REWIND     ;CALL TAPE REWIND COMMAND
13424 076776 103411      BCS    30#           ;BR, IF NO PROBLEM
13425 077000 016501 000002      MOV     TSSR(R5),R1   ;GET TSSR CONTENTS
13426 077004 010004      MOV     R0,R4         ;SET UP REWIND PACKET ADDRESS
13427 077006 005237 002212      INC     FATFLG        ;BUMP COUNT
13431 077012      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      077012 104456      TRAP   C#ERHRD
      077014 001307      .WORD 711
      077016 104165      .WORD T27RWN
      077020 012136      .WORD PKTSSR
13432 077022      30#:  CKLOOP          ;LOOP IF SELECTED      TRAP   C#CLP1
      077022 104406
13433
13434      ;*****
13435      ;

```

## TEST 7: WRITE DATA RETRY

```

13436 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13437 ;
13438 ;*****
13439
13440 077024 013701 102660          MOV     T27BFR+6,R1      ;PICK UP XSTO
13441 077030 010102          MOV     R1,R2           ;SET UP EXPECTED
13442 077032 052702 000002      BIS     @BIT1,R2        ;SET BOT BIT IN EXPECTED
13443 077036 020102          CMP     R1,R2           ;DOES EXP = REC'D
13444 077040 001406          BEQ    40$             ;BR, IF EQUAL (OK)
13445 077042 005237 002212      INC     FATFLG          ;BUMP COUNT
13449 077046          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP     C$ERHRD
                                .WORD    712
                                .WORD    T27BOT
                                .WORD    EXPREC
                                TRAP     C$CLP1
077046 104456
077050 001310
077052 103661
077054 015564
13450 077056          40$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP     C$CLP1
077056 104406
13451 ;*****
13452 ;
13453 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
13454 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
13455 ;
13456 ;*****
13457
13458
13459 077060 012703 000001          MOV     @1,R3           ;PARAMETER SPACE FORWARD 1 RECORD
13460 077064 004737 010556      JSR    PC,SPACE        ;CALL SPACE RECORDS ROUTINE
13461 077070 103413          BCS    50$             ;BR, IF NO ERRORS
13462 077072 016501 000002      MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
13463 077076 012702 000200      MOV     @SSR,R2        ;SET UP EXPECTED
13464 077102 010004          MOV     R0,R4           ;SET UP REWIND PACKET ADDRESS
13465 077104 005237 002212      INC     FATFLG          ;BUMP COUNT
13469 077110          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE RECORDS COMMAND FAILED
                                TRAP     C$ERHRD
                                .WORD    713
                                .WORD    T27SCF
                                .WORD    PKTSSR
                                TRAP     C$CLP1
077110 104456
077112 001311
077114 105427
077116 012136
13470 077120          50$:   CKLOOP          ;LOOP IF SELECTED
                                TRAP     C$CLP1
077120 104406
13471 ;*****
13472 ;
13473 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
13474 ;BIT 15 SETS DIRECTION - 0=FORWARD 1=REVERSE
13475 ;
13476 ;*****
13477
13478
13479 077122 012703 100001          MOV     @100001,R3     ;PARAMETER SPACE REVERSE 1 RECORD
13480 077126 004737 010556      JSR    PC,SPACE        ;CALL SPACE RECORDS ROUTINE
13481 077132 103413          BCS    60$             ;BR, IF NO ERRORS
13482 077134 016501 000002      MOV     TSSR(R5),R1    ;GET TSSR CONTENTS
13483 077140 012702 000200      MOV     @SSR,R2        ;SET UP EXPECTED
13484 077144 010004          MOV     R0,R4           ;SET UP REWIND PACKET ADDRESS
13485 077146 005237 002212      INC     FATFLG          ;BUMP COUNT
13489 077152          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE RECORDS COMMAND FAILED
                                TRAP     C$ERHRD
                                .WORD    714
                                .WORD    714
077152 104456
077154 001312

```





TEST 7: WRITE DATA RETRY

```

13590 077440 005237 002212      INC      FATFLG      ;BUMP COUNT
13594 077444      ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      077444 104456      TRAP      C#ERHRD
      077446 001317      .WORD    719
      077450 104165      .WORD    T27RWN
      077452 012136      .WORD    PKTSSR
13595 077454      30#:   CKLOOP      ;LOOP IF SELECTED      TRAP      C#CLP1
      077454 104406
13596
13597      ;*****
13598      ;
13599      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13600      ;
13601      ;*****
13602
13603 077456 013701 102660      MOV      T27BFR+6,R1      ;PICK UP XSTO
13604 077462 010102      MOV      R1,R2      ;SET UP EXPECTED
13605 077464 052702 000002      BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
13606 077470 C20102      CMP      R1,R2      ;DOES EXP = REC'D
13607 077472 001406      BEQ      40#      ;BR, IF EQUAL (OK)
13608 077474 005237 002212      INC      FATFLG      ;BUMP COUNT
13612 J77500      ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      077500 104456      TRAP      C#ERHRD
      077502 001320      .WORD    720
      077504 103661      .WORD    T27BOT
      077506 015564      .WORD    EXPREC
13613 077510      40#:   CKLOOP      ;LOOP IF SELECTED      TRAP      C#CLP1
      077510 104406
13614 077512 012703 000024      MOV      #20.,R3      ;STARTING RECORD SIZE
13615 077516 013737 003114 102762      MOV      FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
13616
13617      ;*****
13618      ;
13619      ;WRITE DATA,CVC=1,ACK COMMAND
13620      ;
13621      ;*****
13622
13623 077524 012737 140005 102760 65#:   MOV      #140005,T27PK3      ;WRITE DATA,CVC=1,ACK COMMAND
13624 077532 0127 4 102760      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
13625 077536 010300      MOV      R3,R0      ;SET PATTERN IN CORRECT REGISTER
13626 077540 004737 017512      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
13627 077544 010337 102766      MOV      R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
13628 077550 010465 000000      MOV      R4,TSD8(R5)      ;ISSUE COMMAND
13629 077554 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
13630 077560 016501 000002      MOV      TSSR(R5),R1      ;GET TSSR CONTENTS
13631 077564 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
13632 077570 0 0102      CMP      R1,R2      ;ARE THEY EQUAL
13633 077572 001406      BEQ      80#      ;BR, IF OK
13634 077574 005237 002212      INC      FATFLG      ;BUMP COUNT
13638 077600      ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      077600 104456      TRAP      C#ERHRD
      077602 001321      .WORD    721
      077604 005111      .WORD    WRTErr
      077606 012136      .WORD    PKTSSR
13639 077610      80#:   CKLOOP      ;LOOP IF SELECTED      TRAP      C#CLP1
      077610 104406
13640

```

TEST 7: WRITE DATA RETRY

```

13641 ;*****
13642 ;
13643 ;WRITE DATA RETRY,CVC-1,ACK COMMAND
13644 ;
13645 ;*****
13646
13647 077612 012737 141005 102760      MOV      #141005,T27PK3      ;WRITE DATA RETRY,CVC-1,ACK COMMAND
13648 077620 010465 000000                MOV      R4,TSDB(R5)        ;ISSUE COMMAND
13649 077624 004737 016340                JSR      PC,WAITF           ;WAIT FOR SSR TO SET
13650 077630 016501 000002                MOV      TSSR(R5),R1       ;GET TSSR CONTENTS
13651 077634 012702 000200                MOV      #SSR,R2          ;SET UP EXPECTED
13652 077640 020102                CMP      R1,R2             ;ARE THEY EQUAL
13653 077642 001406                BEQ      90$               ;BR, IF OK
13654 077644 005237 002212                INC      FATFLG            ;BUMP COUNT
13658 077650                ERRHRD  ERRNO,T27WRF,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA RETRY
                                TRAP      C$ERHRD
                                .WORD    722
                                .WORD    T27WRF
                                .WORD    PKTSSR
                                90$:  CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
13659 077660                TST      (R3)              ;BUMP RECORD SIZE COUNTER
                                CMP      R3,#40.            ;AT 40 SIZE YET
13660 077662 005723 000050                BNE      65$               ;BR, IF MORE RECORDS TO WRITE
13661 077664 020327
13662 0776 0 001315
13663 ;*****
13664 ;
13665 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
13666 ;
13667 ;*****
13668
13669
13670 077672 004737 011104                JSR      PC,REWIND         ;CALL TAPE REWIND COMMAND
13671 077676 103407                BCS      230$             ;BR, IF NO PROBLEM
13672 077700 010001                MOV      R0,R1            ;SAVE TSSR
13673 077702 005237 002212                INC      FATFLG            ;BUMP COUNT
13677 077706                ERRHRD  ERRNO,T27RWN,EXPREC ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD    723
                                .WORD    T27RWN
                                .WORD    EXPREC
                                230$: CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
13678 077716 104406
13679 ;*****
13680 ;
13681 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13682 ;
13683 ;*****
13684
13685
13686 077720 013701 102660                MOV      T27BFR+6,R1      ;PICK UP XSTO
13687 077724 010102                MOV      R1,R2            ;SET UP EXPECTED
13688 077726 052702 000002                BIS      #BIT1,R2         ;SET BOT BIT IN EXPECTED
13689 077732 020102                CMP      R1,R2            ;DOES EXP = REC'D
13690 077734 001406                BEQ      240$             ;BR, IF EQUAL (OK)
13691 077736 005237 002212                INC      FATFLG            ;BUMP COUNT
13695 077742                ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                077742 104456

```

TEST 7: WRITE DATA RETRY

```

077744 001324 .WORD 724
077746 103661 .WORD T27BOT
077750 015564 .WORD EXPREC
13696 077752 240$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
077752 104406 ;STARTING RECORD SIZE
13697 077754 012703 000024 MOV #20.,R3 ;STARTING READ BUFFER ADDRESS
13698 077760 013737 003114 102762 MOV FREE,T27RB
13699 ;*****
13700 ;READ DATA,ACK COMMAND
13701 ;
13702 ;*****
13703
13704
13705
13706 077766 012737 100001 102760 265$: MOV #100001,T27PK3 ;READ DATA,ACK COMMAND
13707 077774 012704 102760 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
13708 100000 010337 102766 MOV R3,T27SZ ;SET UP RECORD SIZE IN PACKET
13709 100004 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
13710 100010 C04737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
13711 100014 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
13712 100020 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
13713 100024 020102 CMP R1,R2 ;ARE THEY EQUAL
13714 100026 001406 BEQ 280$ ;BR, IF OK
13715 100030 005237 002212 INC FATFLG ;BUMP COUNT
13719 100034 ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
100034 104456 TRAP C#ERHRD
100036 001325 .WORD 725
100040 005204 .WORD RDERR
100042 012136 .WORD PKTSSR
13720 100044 280$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
100044 104406 ;GET BUFFER ADDRESS
13721 100046 013702 003114 MOV FREE,R2 ;GET RECORD SIZE
13722 100052 010304 MOV R3,R4 ;POINT BACK TO 1ST RECORD
13723 100054 162704 000024 SUB #20.,R4 ;POINT TO 1ST LOC IN BUFFER
13724 100060 060204 285$: ADD R2,R4 ;DATA WRITTEN = READ
13725 100062 021403 CMP (R4),R3 ;BR, IF DATA OK (GOOD)
13726 100064 001410 BEQ 290$ ;PICK UP BAD DATA
13727 100066 011401 MOV (R4),R1 ;SET UP EXPECTED
13728 100070 010302 MOV R3,R2 ;BUMP COUNT
13729 100072 005237 002212 INC FATFLG ;DATA IN BUFFER NOT CORRECT
13733 100076 ERRHRD ERRNO,T27DTA,EXPREC TRAP C#ERHRD
100076 104456 .WORD 726
100100 001326 .WORD T27DTA
100102 105646 .WORD EXPREC
100104 015564
13734 100106 290$: CKLOOP ;LOOP IF SELECTED TRAP C#CLP1
100106 104406 ;BUMP TO NEXT ADDRESS
13735 100110 005724 TST (R4)+ ;BACK TO RECORD SIZE
13736 100112 160204 SUB R2,R4 ;AT END OF RECORD YET
13737 100114 020403 CMP R4,R3 ;BR, IF MORE DATA TO CHECK
13738 100116 001360 BNE 285$ ;BUMP RECORD SIZE
13739 100120 005723 TST (R3)+ ;DONE YET
13740 100122 020327 000050 CMP R3,#40. ;BR, IF NOT DONE YET (MORE READS)
13741 100126 001317 BNE 265$ ;LOOP IF SELECTED
13742 100130 300$: CKLOOP TRAP C#CLP1
100130 104406
13743 100132 330$:

```





TEST 7: WRITE DATA RETRY

```

13788 ;*****
13789 ;
13790 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
13791 ;
13792 ;*****
13793
13794 100270 004737 010752          JSR      PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
13795 100274 103407                BCS      23$                ;BR, IF COMMAND ISSUED OK
13796 100276 005237 002212          INC      FATFLG             ;BUMP COUNT
13800 100302 010001                MOV      R0,R1              ;SAVE CONTENTS OF TSSR
13801 100304                ERRHRD   ERRNO,WRTMSG,SFMSG ;WRITE CHARACTERISTICS FAILED
                                TRAP      C$ERHRD
                                .WORD     728
                                .WORD     WRTMSG
                                .WORD     SFMSG
                                100304 104456
                                100306 001330
                                100310 005054
                                100312 012124
13802 100314                23$:   CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                100314 104406

13803 ;*****
13804 ;
13805 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
13806 ;
13807 ;*****
13808
13809
13810 100316 004737 011104          JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
13811 100322 103411                BCS      30$                ;BR, IF NO PROBLEM
13812 100324 016501 000002          MOV      TSSR(R5),R1        ;GET TSSR CONTENTS
13813 100330 010004                MOV      R0,R4              ;GET PACKET ADDRESS
13814 100332 005237 002212          INC      FATFLG             ;BUMP COUNT
13818 100336                ERRHRD   ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      C$ERHRD
                                .WORD     729
                                .WORD     T27RWN
                                .WORD     PKTSSR
                                100336 104456
                                100340 001331
                                100342 104165
                                100344 012136
13819 100346                30$:   CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                100346 104406

13820 ;*****
13821 ;
13822 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13823 ;
13824 ;*****
13825
13826
13827 100350 013701 102660          MOV      T27BFR+6,R1        ;PICK UP XSTO
13828 100354 010102                MOV      R1,R2              ;SET UP EXPECTED
13829 100356 052702 000002          BIS      #BIT1,R2           ;SET BOT BIT IN EXPECTED
13830 100362 020102                CMP      R1,R2              ;DOES EXP = REC'D
13831 100364 001406                BEQ      40$                ;BR, IF EQUAL (OK)
13832 100366 005237 002212          INC      FATFLG             ;BUMP COUNT
13836 100372                ERRHRD   ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C$ERHRD
                                .WORD     730
                                .WORD     T27BOT
                                .WORD     EXPREC
                                100372 104456
                                100374 001332
                                100376 103661
                                100400 015564
13837 100402                40$:   CKLOOP                ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                100402 104406
13838 100404 012703 000024          MOV      #20.,R3           ;STARTING RECORD SIZE

```

## TEST 7: WRITE DATA RETRY

```

13839 100410 013737 003114 102762      MOV      FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
13840
13841      ;*****
13842      ;
13843      ;WRITE DATA,CVC=1,ACK COMMAND
13844      ;
13845      ;*****
13846
13847 100416 012737 140005 102760 65$:  MOV      #140005,T27PK3      ;WRITE DATA,CVC=1,ACK COMMAND
13848 100424 012704 102760      MOV      #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
13849 100430 010300      MOV      R3,R0          ;SET PATTERN IN CORRECT REGISTER
13850 100432 004737 017512      JSR      PC,FILLMEM      ;FILL MEMORY WITH RECORD SIZE
13851 100436 010337 102766      MOV      R3,T27SZ      ;SET UP RECORD SIZE IN PACKET
13852 100442 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
13853 100446 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
13854 100452 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
13855 100456 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
13856 100462 020102      CMP      R1,R2          ;ARE THEY EQUAL
13857 100464 C01406      BEQ      80$           ;BR, IF OK
13858 100466 005237 002212      INC      FATFLG        ;BUMP COUNT
13862 100472      ERRHRD  ERRNO,WRterr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      100472 104456      TRAP    C#ERHRD
      100474 001333      .WORD  731
      100476 005111      .WORD  WRterr
      100500 012136      .WORD  PKTSSR
13863 100502 80$:  CKLOOP      ;LOOP IF SELECTED      TRAP    C#CLP1
      100502 104406
13864
13865      ;*****
13866      ;
13867      ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
13868      ;
13869      ;*****
13870
13871 100504 012737 111005 102760      MOV      #111005,T27PK3 ;WRITE DATA RETRY,ACK,SWB=1 COMMAND
13872 100512 010465 000000      MOV      R4,TSDB(R5)    ;ISSUE COMMAND
13873 100516 004737 016340      JSR      PC,WAITF      ;WAIT FOR SSR TO SET
13874 100522 016501 000002      MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
13875 100526 012702 000200      MOV      #SSR,R2      ;SET UP EXPECTED
13876 100532 020102      CMP      R1,R2          ;ARE THEY EQUAL
13877 100534 001406      BEQ      90$           ;BR, IF OK
13878 100536 005237 002212      INC      FATFLG        ;BUMP COUNT
13882 100542      ERRHRD  ERRNO,T27WRF,EXPREC ;TSSR INCORRECT AFTER WRITE DATA RETRY
      100542 104456      TRAP    C#ERHRD
      100544 001334      .WORD  732
      100546 105566      .WORD  T27WRF
      100550 015564      .WORD  EXPREC
13883 100552 90$:  CKLOOP      ;LOOP IF SELECTED      TRAP    C#CLP1
      100552 104406
13884 100554 005723      TST     (R3)+          ;BUMP RECORD SIZE COUNTER
13885 100556 020327 000050      CMP     R3,#40        ;AT 40 SIZE YET
13886 100562 001315      BNE     65$           ;BR, IF MORE RECORDS TO WRITE
13887
13888      ;*****
13889      ;
13890      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
13891      ;

```

TEST 7: WRITE DATA RETRY

```

13892 ;*****
13893
13894 100564 004737 011104 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
13895 100570 103411 BCS 230# ;BR, IF NO PROBLEM
13896 100572 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
13897 100576 010004 MOV R0,R4 ;GET PACKET ADDRESS
13898 100600 005237 002212 INC FATFLG ;BUMP COUNT
13902 100604 ERRHRD ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
; TRAP C#ERHRD
; .WORD 733
; .WORD T27RWN
; .WORD PKTSSR
13903 100614 230#: CKLOOP ;LOOP IF SELECTED
100614 104406 ; TRAP C#CLP1
13904 ;*****
13905 ;
13906 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
13907 ;
13908 ;*****
13909 ;
13910
13911 100616 013701 102660 MOV T27BFR+6,R1 ;PICK UP XSTO
13912 100622 010102 MOV R1,R2 ;SET UP EXPECTED
13913 100624 052702 000002 BIS #BIT1,R2 ;SET BOT BIT IN EXPECTED
13914 100630 020102 CMP R1,R2 ;DOES EXP = REC'D
13915 100632 001406 BEQ 240# ;BR, IF EQUAL (OK)
13916 100634 005237 002212 INC FATFLG ;BUMP COUNT
13920 100640 ERRHRD ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
; TRAP C#ERHRD
; .WORD 734
; .WORD T27BOT
; .WORD EXPREC
13921 100650 240#: CKLOOP ;LOOP IF SELECTED
100650 104406 ; TRAP C#CLP1
13922 100652 012703 000024 MOV #20.,R3 ;STARTING RECORD SIZE
13923 100656 013737 003114 102762 MOV FREE,T27RB ;STARTING READ BUFFER ADDRESS
13924 ;*****
13925 ;
13926 ;READ DATA,ACK COMMAND
13927 ;
13928 ;*****
13929 ;
13930
13931 100664 012737 100001 102760 265#: MOV #100001,T27PK3 ;READ DATA,ACK COMMAND
13932 100672 012704 102760 MOV #T27PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
13933 100676 010337 102766 MOV R3,T27SZ ;SET UP RECORD SIZE IN PACKET
13934 100702 010465 000000 MOV R4,TSDB(R5) ;ISSUE COMMAND
13935 100706 004737 016340 JSR PC,WAITF ;WAIT FOR SSR TO SET
13936 100712 016501 000002 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
13937 100716 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
13938 100722 020102 CMP R1,R2 ;ARE THEY EQUAL
13939 100724 001406 BEQ 280# ;BR, IF OK
13940 100726 005237 002212 INC FATFLG ;BUMP COUNT
13944 100732 ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
; TRAP C#ERHRD
; .WORD 735
; .WORD RDERR
100732 104456
100734 001337
100736 005204

```





TEST 7: WRITE DATA RETRY

```

14041
14042 ;*****
14043 ;
14044 ;WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
14045 ;
14046 ;*****
14047
14048 101206 004737 010752          JSR    PC,WRTCHR          ;ISSUE WRITE CHARACTERISTICS
14049 101212 103407                BCS    23#                ;BR, IF COMMAND ISSUED OK
14050 101214 005237 002212          INC    FATFLG             ;BUMP COUNT
14054 101220 010001                MOV    R0,R1              ;SAVE CONTENTS OF TSSR
14055 101222                ERRHRD  ERRNO,WRTMSG,SFMSG  ;WRITE CHARACTERISTIC FAILED
                                TRAP    C#ERHRD
                                .WORD   738
                                .WORD   WRTMSG
                                .WORD   SFMSG
                                101222 104456
                                101224 001342
                                101226 005054
                                101230 012124
14056 101232                23#:   CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1
                                101232 104406
14057
14058 ;*****
14059 ;
14060 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
14061 ;
14062 ;*****
14063
14064 101234 004737 011104          JSR    PC,REWIND          ;CALL TAPE REWIND COMMAND
14065 101240 103411                BCS    30#                ;BR, IF NO PROBLEM
14066 101242 016501 000002          MOV    TSSR(R5),R1        ;GET TSSR CONTENTS
14067 101246 010004                MOV    R0,R4              ;GET PACKET ADDRESS
14068 101250 005237 002212          INC    FATFLG             ;BUMP COUNT
14072 101254                ERRHRD  ERRNO,T27RWN,PKTSSR  ;REWIND NOT ACCEPTED
                                TRAP    C#ERHRD
                                .WORD   739
                                .WORD   T27RWN
                                .WORD   PKTSSR
                                101254 104456
                                101256 001343
                                101260 104165
                                101262 012136
14073 101264                30#:   CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1
                                101264 104406
14074
14075 ;*****
14076 ;
14077 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
14078 ;
14079 ;*****
14080
14081 101266 013701 102660          MOV    T27BFR+6,R1        ;PICK UP XSTO
14082 101272 010102                MOV    R1,R2              ;SET UP EXPECTED
14083 101274 052702 000002          BIS    #BIT1,R2           ;SET BOT BIT IN EXPECTED
14084 101300 020102                CMP    R1,R2              ;DOES EXP = REC'D
14085 101302 001406                BEQ    40#                ;BR, IF EQUAL (OK)
14086 101304 005237 002212          INC    FATFLG             ;BUMP COUNT
14090 101310                ERRHRD  ERRNO,T27BOT,EXPREC  ;TAPE NOT AT BOT AFTER REWIND
                                TRAP    C#ERHRD
                                .WORD   740
                                .WORD   T27BOT
                                .WORD   EXPREC
                                101310 104456
                                101312 001344
                                101314 103661
                                101316 015564
14091 101320                40#:   CKLOOP                ;LOOP IF SELECTED
                                TRAP    C#CLP1
                                101320 104406

```

## TEST 7: WRITE DATA RETRY

```

14092 101322 012703 000144          MOV    #100.,R3          ;NUMBER OF RECORDS TO BE WRITTEN
14093 101326 013737 003114 102762  MOV    FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
14094
14095 ;*****
14096 ;
14097 ;WRITE DATA,ACK,CVC-1 COMMAND
14098 ;
14099 ;*****
14100
14101 101334 012737 140005 102760 65#:  MOV    #140005,T27PK3  ;WRITE DATA,ACK,CVC-1 COMMAND
14102 101342 012704 102760          MOV    #T27PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
14103 101346 012737 000024 102766  MOV    #20.,T27SZ      ;SET UP RECORD SIZE IN PACKET
14104 101354 010465 000000          MOV    R4,TSD8(R5)     ;ISSUE COMMAND
14105 101360 004737 016340          JSR    PC,WAITF        ;WAIT FOR SSR TO SET
14106 101364 016501 000002          MOV    TSSR(R5),R1     ;GET TSSR CONTENTS
14107 101370 012702 000200          MOV    #SSR,R2        ;SET UP EXPECTED
14108 101374 020102          CMP    R1,R2          ;ARE THEY EQUAL
14109 101376 001406          BEQ   70#            ;BR, IF OK
14110 101400 C05237 002212          INC   FATFLG         ;BUMP COUNT
14111 101404          ERRHRD  ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
14112 101404 104456          TRAP  C#ERRHRD
14113 101406 001345          .WORD 741
14114 101410 005111          .WORD WRTErr
14115 101412 012136          .WORD PKTSSR
14115 101414          70#:  CKLOOP          ;LOOP IF SELECTED          TRAP  C#CLP1
14116 101414 104406          DEC   R3              ;DEC RECORD COUNTER
14117 101416 005303          BNE  65#             ;BR, IF MORE RECORDS TO WRITE
14118 101420 001345
14119
14120 ;*****
14121 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
14122 ;
14123 ;*****
14124
14125 101422 004737 011104          JSR    PC,REWIND      ;CALL TAPE REWIND COMMAND
14126 101426 103411          BCS   130#           ;BR, IF NO PROBLEM
14127 101430 016501 000002          MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
14128 101434 010004          MOV    R0,R4          ;GET PACKET ADDRESS
14129 101436 005237 002212          INC   FATFLG         ;BUMP COUNT
14130 101442          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
14131 101442 104456          TRAP  C#ERRHRD
14132 101444 001346          .WORD 742
14133 101446 104165          .WORD T27RWN
14134 101450 012136          .WORD PKTSSR
14134 101452          130#: CKLOOP          ;LOOP IF SELECTED          TRAP  C#CLP1
14135 101452 104406
14136
14137 ;*****
14138 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
14139 ;
14140 ;*****
14141
14142 101454 013701 102660          MOV    T27BFR+6,R1   ;PICK UP XSTO
14143 101460 010102          MOV    R1,R2         ;SET UP EXPECTED
14144 101462 052702 000002          BIS   #BIT1,R2       ;SET BOT BIT IN EXPECTED

```

TEST 7: WRITE DATA RETRY

```

14145 101466 020102          CMP      R1,R2          ;DOES EXP = REC'D
14146 101470 001406          BEQ      140#          ;BR, IF EQUAL (OK)
14147 101472 005237 002212    INC      FATFLG        ;BUMP COUNT
14151 101476          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          101476 104456          TRAP      CIERHRD
          101500 001347          .WORD    743
          101502 103661          .WORD    T27BOT
          101504 015564          .WORD    EXPREC
14152 101506          140# :  CKLOOP          ;LOOP IF SELECTED
          101506 104406          TRAP      CICLP1
14153 101510 012704 102760    MOV      #T27PK3,R4    ;SET UP PACKET ADDRESS
14154 101514 012737 000010 102762    MOV      #10,T27R8    ;SET UP RECORDS TO SPACE OVER
14155
14156 ;*****
14157 ;
14158 ;ACK,CVC=1,SPACE FORWARD COMMAND
14159 ;
14160 ;*****
14161
14162 101522 012737 140010 102760    MOV      #140010,T27PK3 ;ACK,CVC=1,SPACE FORWARD COMMAND
14163 101530 010465 000000    150# :  MOV      R4,TSDB(R5) ;ISSUE COMMAND
14164 101534 005237 103006    152# :  INC      T27CNT    ;BUMP TIMER
14165 101540          DELAY    1              ;DELAY ABOUT 100US
          101540 012727 000001          MOV      #1,(PC)+
          101544 000000          .WORD    0
          101546 013727 002116          MOV      L0DLY,(PC)+
          101552 000000          .WORD    0
          101554 005367 177772          DEC      -6(PC)
          101560 001375          BNE     -.4
          101562 005367 177756          DEC      -22(PC)
          101566 001367          BNE     .-20
14166 101570 016501 000002    MOV      TSSR(R5),R1   ;GET TSSR
14167 101574 032701 000200    BIT      #BIT7,R1     ;CHECK FOR TSSR'S SSR SET
14168 101600 001755          BEQ      152#         ;KEEP COUNTING UNTIL SET
14169 101602 016501 000002    MOV      TSSR(R5),R1   ;GET STATUS FROM TSSR
14170 101606 012702 000200    MOV      #SSR,R2      ;SET UP EXPECTED
14171 101612 020201          CMP      R2,R1        ;WAS EVERYTHING OK
14172 101614 001406          BEQ      160#         ;BR, IF ALL IS WELL
14173 101616 005237 002212    INC      FATFLG        ;BUMP COUNT
14177 101622          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD DIDN'T WORK OUT
          101622 104456          TRAP      CIERHRD
          101624 001350          .WORD    744
          101626 105427          .WORD    T27SCF
          101630 012136          .WORD    PKTSSR
14178 101632          160# :  CKLOOP          ;LOOP IF SELECTED
          101632 104406          TRAP      LICLP1
14179
14180 ;*****
14181 ;
14182 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
14183 ;
14184 ;*****
14185
14186 101634 004737 011104    JSR      PC.REWIND     ;CALL TAPE REWIND COMMAND
14187 101640 004737 016426    JSR      PC.CKNTSSR   ;SEE HOW TSSR IS
14188 101644 103407          BCS     170#         ;BR, IF NO PROBLEM
14189 101646 010001          MOV      R0,R1        ;SAVE TSSR

```



TEST 7: WRITE DATA RETRY

```

14190 101650 005237 002212          INC  FATFLG          ;BUMP COUNT
14194 101654          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      101654 104456          TRAP  C#ERHRD
      101656 001351          .WORD 745
      101660 104165          .WORD T27RWN
      101662 012136          .WORD PKTSSR
14195 101664          170$: CKLOOP          ;LOOP IF SELECTED
      101664 104406          TRAP  C#CLP1

14196
14197
14198
14199
14200
14201
14202
      ;*****
      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
      ;*****
14203 101666 013701 102660          MOV   T27BFR+6,R1      ;PICK UP XSTO
14204 101672 010102          MOV   R1,R2           ;SET UP EXPECTED
14205 101674 052702 000002          BIS   #BIT1,R2        ;SET BOT BIT IN EXPECTED
14206 101700 C20102          CMP   R1,R2           ;DOES EXP = REC'D
14207 101702 001406          BEQ   175$            ;BR, IF EQUAL (OK)
14208 101704 005237 002212          INC   FATFLG          ;BUMP COUNT
14212 101710          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      101710 104456          TRAP  C#ERHRD
      101712 001352          .WORD 746
      101714 103661          .WORD T27BOT
      101716 015564          .WORD EXPREC
14213 101720          175$: CKLOOP          ;LOOP IF SELECTED
      101720 104406          TRAP  C#CLP1
14214 101722 012703 000144          MOV   #100.,R3        ;STARTING RECORD SIZE
14215 101726 013737 003114 102762 177$: MOV   FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
14216
14217
14218
14219
14220
14221
14222
      ;*****
      ;WRITE DATA,CVC=1,ACK COMMAND
      ;*****
14223 101734 012737 140005 102760          MOV   #140005,T27PK3 ;WRITE DATA,CVC=1,ACK COMMAND
14224 101742 012704 102760          MOV   #T27PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
14225 101746 012737 000024 102766          MOV   #20.,T27SZ     ;SET UP RECORD SIZE IN PACKET
14226 101754 010465 000000          MOV   R4,T27SDB(R5)  ;ISSUE COMMAND
14227 101760 004737 016340          JSR   PC,WAITF        ;WAIT FOR SSR TO SET
14228 101764 016501 000002          MOV   TSSR(R5),R1    ;GET TSSR CONTENTS
14229 101770 012702 000200          MOV   #SSR,R2        ;SET UP EXPECTED
14230 101774 020102          CMP   R1,R2           ;ARE THEY EQUAL
14231 101776 001406          BEQ   180$            ;BR, IF OK
14232 102000 005237 002212          INC   FATFLG          ;BUMP COUNT
14236 102004          ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      102004 104456          TRAP  C#ERHRD
      102006 001353          .WORD 747
      102010 005111          .WORD WRERR
      102012 012136          .WORD PKTSSR
14237 102014          180$: CKLOOP          ;LOOP IF SELECTED
      102014 104406          TRAP  C#CLP1
14238 102016 005303          DEC   R3              ;COUNT NUMBER OF RECORDS
14239 102020 001342          BNE   177$            ;BR, IF MORE RECORDS TO WRITE
14240

```

TEST 7: WRITE DATA RETRY

```

14241 ;*****
14242 ;
14243 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
14244 ;
14245 ;*****
14246
14247 102022 004737 011104      JSR      PC,REWIND      ;ISSUE REWIND
14248 102026 103411          BCS      182$          ;BR, IF ALL IS WELL
14249 102030 010004          MOV      R0,R4         ;GET PACKET ADDRESS
14250 102032 016501 000002    MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
14251 102036 005237 002212    INC      FATFLG        ;BUMP COUNT
14255 102042          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND FAILED
14255 102042 104456          TRAP      C$ERHRD
14255 102044 001354          .WORD    748
14255 102046 104165          .WORD    T27RWN
14255 102050 012136          .WORD    PKTSSR
14256 102052          182$:  CKLOOP          ;SELECT LOOP MAYBE
14256 102052 104406          TRAP      C$CLP1
14257
14258 ;*****
14259 ;
14260 ;ISSUE SPACE RECORDS COMMAND - VALUE IN R3 SETS NUMBER OF RECORDS
14261 ;BIT 15 SETS DIRECTION - 0=FORWARD  1=REVERSE
14262 ;
14263 ;*****
14264
14265 102054 012703 000001    MOV      #1.,R3        ;SPACE 1 RECORD FORWARD
14266 102060 004737 010556    JSR      PC,SPACE      ;ISSUE SPACE COMMAND
14267 102064 103411          BCS      185$          ;BR, IF COMMAND OK
14268 102066 010004          MOV      R0,R4         ;GET PACKET ADDRESS
14269 102070 016501 000002    MOV      TSSR(R5),R1   ;GET TSSR STATUS
14270 102074 005237 002212    INC      FATFLG        ;BUMP COUNT
14274 102100          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD COMMAND FAILED
14274 102100 104456          TRAP      C$ERHRD
14274 102102 001355          .WORD    749
14274 102104 105427          .WORD    T27SCF
14274 102106 012136          .WORD    PKTSSR
14275 102110          185$:  CKLOOP          ;LOOP IF SELECTED
14275 102110 104406          TRAP      C$CLP1
14276 102112 012703 000144    MOV      #100.,R3      ;NUMBER OF RECORDS TO BE WRITTEN
14277 102116 013737 003114 102762  MOV      FREE,T27WB    ;STARTING WRITE BUFFER ADDRESS
14278
14279 ;*****
14280 ;
14281 ;WRITE DATA RETRY,ACK COMMAND
14282 ;
14283 ;*****
14284
14285 102124 012737 101005 102760 190$:  MOV      #101005,T27PK3 ;WRITE DATA RETRY,ACK COMMAND
14286 102132 012704 102760          MOV      #T27PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
14287 102136 012737 000024 102766    MOV      #20.,T27SZ    ;SET UP RECORD SIZE IN PACKET
14288 102144 010465 000000          MOV      R4,TSD8(R5)   ;ISSUE COMMAND
14289 102150 004737 016340          JSR      PC,WAITF      ;WAIT FOR SSR TO SET
14290 102154 016501 000002    MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
14291 102160 012702 000200          MOV      #SSR,R2       ;SET UP EXPECTED
14292 102164 020102          CMP      R1,R2         ;ARE THEY EQUAL
14293 102166 001406          BEQ      200$          ;BR, IF OK

```

TEST 7: WRITE DATA RETRY

```

14294 102170 005237 002212          INC  FATFLG          ;BUMP COUNT
14298 102174          ERRHRD  ERRNO,T27WDC,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      102174 104456          TRAP  C%ERHRD
      102176 001356          .WORD 750
      102200 104521          .WORD T27WDC
      102202 012136          .WORD PKTSSR
14299 102204          200$: CKLOOP          ;LOOP IF SELECTED
      102204 104406          TRAP  C%CLP1
14300 102206 013737 0J3114 102762    MOV  FREE,T27WB      ;STARTING WRITE BUFFER ADDRESS
14301
14302          ;*****
14303          ;
14304          ;WRITE DATA,CVC-1,ACK COMMAND
14305          ;
14306          ;*****
14307
14308 102214 012737 140005 102760    MOV  #140005,T27PK3 ;WRITE DATA,CVC-1,ACK COMMAND
14309 102222 012704 102760    MOV  #T27PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
14310 102226 C12737 000024 102766    MOV  #20.,T27SZ    ;SET UP RECORD SIZE IN PACKET
14311 102234 010465 000000    MOV  R4,TSDB(R5)   ;ISSUE COMMAND
14312 102240 004737 016340    JSR  PC,WAITF      ;WAIT FOR SSR TO SET
14313 102244 016501 000002    MOV  TSSR(R5),R1   ;GET TSSR CONTENTS
14314 102250 012702 000200    MOV  #SSR,R2       ;SET UP EXPECTED
14315 102254 020102          CMP  R1,R2         ;ARE THEY EQUAL
14316 102256 001406          BEQ  210$         ;BR, IF OK
14317 102260 005237 002212          INC  FATFLG          ;BUMP COUNT
14321 102264          ERRHRD  ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      102264 104456          TRAP  C%ERHRD
      102266 001357          .WORD 751
      102270 005111          .WORD WRERR
      102272 012136          .WORD PKTSSR
14322 102274          210$: CKLOOP          ;LOOP IF SELECTED
      102274 104406          TRAP  C%CLP1
14323 102276 005303          DEC  R3            ;BUMP DOWN RECORD COUNTER
14324 102300 001311          BNE  190$         ;BR, IF MORE RECORDS TO WRITE RETRY
14325
14326          ;*****
14327          ;
14328          ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
14329          ;
14330          ;*****
14331
14332 102302 004737 011104    JSR  PC,REWIND     ;CALL TAPE REWIND COMMAND
14333 102306 103411          BCS  230$         ;BR, IF NO PROBLEM
14334 102310 016501 000002    MOV  TSSR(R5),R1   ;GET TSSR CONTENTS
14335 102314 010004          MOV  R0,R4        ;GET PACKET ADDRESS
14336 102316 005237 002212          INC  FATFLG          ;BUMP COUNT
14340 102322          ERRHRD  ERRNO,T27RWN,PKTSSR ;REWIND NOT ACCEPTED
      102322 104456          TRAP  C%ERHRD
      102324 001360          .WORD 752
      102326 104165          .WORD T27RWN
      102330 012136          .WORD PKTSSR
14341 102332          230$: CKLOOP          ;LOOP IF SELECTED
      102332 104406          TRAP  C%CLP1
14342
14343          ;*****
14344          ;

```

TEST 7: WRITE DATA RETRY

```

14345 ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
14346 ;
14347 ;*****
14348
14349 102334 013701 102660      MOV      T27BFR+6,R1      ;PICK UP XSTO
14350 102340 010102          MOV      R1,R2          ;SET UP EXPECTED
14351 102342 052702 000002    BIS      #BIT1,R2      ;SET BOT BIT IN EXPECTED
14352 102346 020102          CMP      R1,R2          ;DOES EXP = REC'D
14353 102350 001406          BEQ      240#          ;BR, IF EQUAL (OK)
14354 102352 005237 002212    INC      FATFLG        ;BUMP COUNT
14358 102356          ERRHRD  ERRNO,T27BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
                                TRAP      C#ERHRD
                                .WORD    753
                                .WORD    T27BOT
                                .WORD    EXPREC
14359 102366          240#:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
14360 102370 012704 102760    MOV      #T27PK3,R4    ;SET UP PACKET ADDRESS
14361 102374 C12737 000010 102762    MOV      #10,T27RB    ;SET UP RECORDS TO SPACE OVER
14362 ;*****
14363 ;
14364 ;ACK,CVC=1,SPACE FORWARD COMMAND
14365 ;
14366 ;*****
14367 ;
14368
14369 102402 012737 140010 102760    MOV      #140010,T27PK3 ;ACK,CVC=1,SPACE FORWARD COMMAND
14370 102410 010465 000000    250#:  MOV      R4,TSDB(R5) ;ISSUE COMMAND
14371 102414 005237 103010    252#:  INC      T27CNU   ;BUMP TIMER
14372 102420          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
14373 102450 016501 000002    MOV      TSSR(R5),R1   ;GET TSSR
14374 102454 032701 000200    BIT      #BIT7,R1     ;CHECK FOR TSSR'S SSR SET
14375 102460 001755          BEQ      252#         ;KEEP COUNTING UNTIL SET
14376 102462 016501 000002    MOV      TSSR(R5),R1   ;GET STATUS FROM TSSR
14377 102466 012702 000200    MOV      #SSR,R2      ;SET UP EXPECTED
14378 102472 020201          CMP      R2,R1        ;WAS EVERYTHING OK
14379 102474 001406          BEQ      260#         ;BR, IF ALL IS WELL
14380 102476 005237 002212    INC      FATFLG        ;BUMP COUNT
14384 102502          ERRHRD  ERRNO,T27SCF,PKTSSR ;SPACE FORWARD DIDN'T WORK OUT
                                TRAP      C#ERHRD
                                .WORD    754
                                .WORD    T27SCF
                                .WORD    PKTSSR
14385 102512          260#:  CKLOOP          ;LOOP IF SELECTED
                                TRAP      C#CLP1
14386 102514 013701 103006    MOV      T27CNT,R1    ;TIME FOR WRITE SPACING
14387 102520 013702 103010    MOV      T27CNU,R2    ;TIME FOR WRITE RETRY SPACING
14388 102524 160102          SUB      R1,R2        ;GET'EM PRETTY CLOSE
14389 102526 160102          SUB      R1,R2        ;GET'EM PRETTY CLOSE

```



## TEST 7: WRITE DATA RETRY

```

14446 102750
14448 102760
14449 102760 100005
14450 102762
14451 102762 003114
14452 102764 000000
14453 102766 000000
14454
14455
14456
14457
14458 102770
14459 102770 010
14460 102771 200
14461 102772 000000
14462 102774 000000
14463
14464
14465
14466
14467
14468 102776 100205
14469 103000 100605
14470 103002 102205
14471 103004 177777
14472
14473
14474 103006 000000
14475 103010 000000
14476 103012 000000
14477
14478
14479
14480
14481
14482
14483
14484 103014 124 141 160
14485 103102 124 123 123
14486 103151 122 105 122
14487 103246 120 117 123
14488 103330 122 111 102
14489 103400 124 123 123
14490 103455 111 154 154
14491 103536 122 105 122
14492 103572 124 123 123
14493 103661 124 141 160
14494 103754 127 122 111
14495 104031 122 105 122
14496 104110 124 123 123
14497 104165 122 145 167
14498 104234 122 101 115
14499 104307 124 123 123
14500 104356 104 162 151
14501 104431 124 123 123
14502 104521 124 123 123
14503 104574 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)
T27WDR: .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'
T27RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
T27RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
T27AM3: .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'

```

TEST 7: WRITE DATA RETRY

14504	104647	124	123	102	T27BA:	.ASCIZ	'TSBA Not Correct After REREAD DATA Command'
14505	104722	127	122	111	T27WSS:	.ASCIZ	'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
14506	105011	122	145	141	T27LON:	.ASCIZ	'Reading Long Record Failed To Set RLL Bit In XST0'
14507	105073	122	145	141	T27LOP:	.ASCIZ	'Reading Long Record Failed To Set RLS Bit In XST0'
14508	105155	122	145	163	T27PBP:	.ASCIZ	'Residual Byte Count Incorrect After Short Record Read'
14509	105243	122	145	141	T27TRL:	.ASCIZ	'Reading Long Record Failed To Give Tape Status Alert'
14510	105331	127	122	111	T27NEF:	.ASCIZ	'WRITE DATA RETRY, At First Record, Failed To Set RIB Bit XST3'
14511	105427	124	123	123	T27SCF:	.ASCIZ	'TSSR Not Correct After SPACE RECORDS Command'
14512	105504	124	123	123	T27TSA:	.ASCIZ	'TSSR Not Correct After WRITE DATA RETRY, Into BOT'
14513	105566	124	123	123	T27WRF:	.ASCIZ	'TSSR Not Correct After WRITE DATA RETRY Command'
14514	105646	104	141	164	T27DTA:	.ASCIZ	'Data Compare Error, Data Read From Tape Not Equal To Written'
14515	105743	127	162	151	TST27ID:	.ASCIZ	'Write Data Retry'
14516						.EVEN	
14517							
14518							
14519							
14520							
14521							
14522							
14523							
14524	105764				T27REST:		
14525	105764				SAVREG		;SAVE THE REGISTERS
14526	105770	012701	102630		MOV	#T27PACKET,R1	;START OF THE PACKET
14527	105774	012721	100004		MOV	#100004,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK,
14528	106000	012721	102640		MOV	#T27DATA,(R1)+	;ADDRESS OF CHARAISTICS DATA BLOCK
14529	106004	005021			CLR	(R1)+	;EXTENDED ADDRESS
14530	106006	012721	000012		MOV	#10..(R1)+	;SIZE OF DATA BLOCK IN BYTES
14531	106012	012721	102652		MOV	#T27BFR,(R1)+	;ADDRESS OF MESSAGE BUFFER
14532	106016	005021			CLR	(R1)+	
14533	106020	012721	000024		MOV	#20..(R1)+	;LENGTH OF MESSAGE BUFFER
14534	106024	005021			CLR	(R1)+	
14535	106026	012711	000000		MOV	#0,(R1)	;SELECT DRIVE ZERO
14536	106032	012702	000030		MOV	#24..R2	;NUMBER OF LOCATIONS TO BE CLEARED
14537	106036	012762	177777	102652	MOV	#177777,T27BFR(R2)	;ALL ONES TO MESSAGE BUFFER
14538	106044	005742			TST	-(R2)	;NEXT LOCATION
14539	106046	022702	000000		CMF	#0,R2	;AT END OF LOOP YET
14540	106052	001371			BNE	64#	;KEEP GOING UNTIL DONE
14541	106054	000207			RTS	PC	;RETURN
14542							
14543							
14544	106056				T27RT2:		
14545	106056				SAVREG		;SAVE THE REGISTERS
14546	106062	012701	102740		MOV	#T27PK2,R1	;START OF THE PACKET
14547	106066	012721	100006		MOV	#100006,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK,
14548	106072	012721	102770		MOV	#T27BF2,(R1)+	;ADDRESS OF DATA BLOCK
14549	106076	005021			CLR	(R1)+	;EXTENDED ADDRESS
14550	106100	012721	000006		MOV	#6..(R1)+	;SIZE OF DATA BLOCK IN BYTES
14551	106104	005021			CLR	(R1)+	
14552	106106	012701	102770		MOV	#T27BF2,R1	;POINT TO DATA SEL AREA
14553	106112	005021			CLR	(R1)+	
14554	106114	005011			CLR	(R1)	
14555	106116	000207			RTS	PC	;RETURN
14556	106120						
14557	106120				T27RT3:		
14558	106124	012701	102760		SAVREG		;SAVE REGISTERS
14559	106130	005021			MOV	#T27PK3,R1	;SET UP POINTER ADDRESS
14560	106132	005021			CLR	(R1)+	;COMMAND SPACE
					CLR	(R1)+	;ADDRESS OF DATA BLOCK





## TEST 8: WRITE/READ TAPE MARK

	106232	104455							TRAP	C#ERDF	
	106234	001441							.WORD	801	
	106236	003650							.WORD	SFIERR	
	106240	012124							.WORD	SFIMSG	
14620	106242	012737	000007	111360	20#:	MOV	#7,T28DSW			;SET UP DRIVE NUMBER	
14621	106250	012704	111340			MOV	#T28PACKET,R4			;SUBROUTINE NEEDS PACKET ADDRESS	
14622	106254	004737	010752			JSR	PC,WRTCHR			;ISSUE WRITE CHARACTERISTICS	
14623	106260	103407				BCS	24#			;BR, IF COMMAND ISSUED OK	
14624	106262	005237	002212			INC	FATFLG			;BUMP COUNT	
14628	106266	010001				MOV	RO,R1			;SAVE CONTENTS OF TSSR	
14629	106270					ERRHRD	ERRNO,WRTMSG,SFIMSG			;WRITE CHARACTERISTISC FAILED	
	106270	104456							TRAP	C#ERHRD	
	106272	001442							.WORD	802	
	106274	005054							.WORD	WRTMSG	
	106276	012124							.WORD	SFIMSG	
14630	106300				24#:	CKLOOP					
	106300	104406								TRAP	C#CLP1
14631	106302	005737	002216			TST	EXTFEA			;CHECK FOR EXTENDED FEATURES SW SWITCH	
14632	106306	C01044				BNE	50#			;BR IF SWITCH IS ON	
14633											
14634	106310	112737	000200	111501		MOV	#200,T28BS1			;WRITE MISCELLANEOUS CONT/READ STATUS	
14635	106316	112737	000010	111500		MOV	#10,T28BS0			;FUNCTION SELECTION BIT (TURN ON EXTFEA HW SWITCH)	
14636	106324	012704	111450			MOV	#T28PK2,R4			;WRITE SUBSYS MEM PACKET	
14637	106330	010465	000000			MOV	R4,TSDB(R5)			;ISSUE COMMAND	
14638	106334	004737	016426			JSR	PC,CHKTSSR			;WAIT FOR SSR	
14639	106340	103407				BCS	30#			;BR, IF NO ERROR	
14640	106342	010001				MOV	RO,R1			;ERROR, SAVE TSSR	
14641	106344	005237	002212			INC	FATFLG			;BUMP COUNT	
14645	106350					ERRHRD	ERRNO,T28SSR,PKTSSR			;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS	
	106350	104456							TRAP	C#ERHRD	
	106352	001443							.WORD	803	
	106354	112175							.WORD	T28SSR	
	106356	012136							.WORD	PKTSSR	
14646	106360				30#:	CKLOOP				;LOOP IF SELECTED	
	106360	104406								TRAP	C#CLP1
14647	106362	012704	111340			MOV	#T28PACKET,R4			;SUBROUTINE NEEDS PACKET ADDRESS	
14648	106366	012737	000007	111360		MOV	#7,T28DSW			;SELECT DRIVE 7	
14649	106374	004737	010752			JSR	PC,WRTCHR			;ISSUE WRITE CHARACTERISTICS	
14650	106400	103407				BCS	50#			;BR, IF COMMAND ISSUED OK	
14651	106402	005237	002212			INC	FATFLG			;BUMP COUNT	
14655	106406	010001				MOV	RO,R1			;SAVE CONTENTS OF TSSR	
14656	106410					ERRHRD	ERRNO,WRTMSG,SFIMSG			;WRITE CHARACTERISTISC FAILED	
	106410	104456							TRAP	C#ERHRD	
	106412	001444							.WORD	804	
	106414	005054							.WORD	WRTMSG	
	106416	012124							.WORD	SFIMSG	
14657	106420				50#:	CKLOOP				;SCOPE LOOP	
	106420	104406								TRAP	C#CLP1
14658	106422	016501	000002			MOV	TSSR(R5),R1			;GET TSSR CONTENTS	
14659	106426	032701	000100			BIT	#OFL,R1			;CHECK FOR THE OFFLINE BIT SET	
14660	106432	001006				BNE	60#			;BR, IF OFFLINE (GOOD)	
14661	106434	005237	002212			INC	FATFLG			;BUMP COUNT	
14665	106440					ERRDF	ERRNO,T28OFL,SFIMSG			;OFF LINE SHOULD HAVE BEEN SET (BAD)	
	106440	104455							TRAP	C#ERDF	
	106442	001445							.WORD	805	
	106444	112530							.WORD	T28OFL	
	106446	012124							.WORD	SFIMSG	





## TEST 8: WRITE/READ TAPE MARK

14763 107034 004737 017272  
 14764 107040  
 14765  
 14766  
 14767  
 14768  
 14769  
 14770  
 14771  
 14772  
 14773  
 14774  
 14775  
 14776  
 14777  
 14778  
 14779  
 14780  
 14781  
 14782  
 14783  
 14784  
 14785  
 14786  
 14787  
 14788  
 14789  
 14790  
 14791  
 14792  
 14793  
 14794  
 14795  
 14796  
 14797  
 14798  
 14799  
 14800  
 14801  
 14802  
 14803  
 14804  
 14805  
 14806  
 14807  
 14808  
 14809  
 14810  
 14811  
 14812  
 14813  
 14814  
 14815  
 14816  
 14817  
 14818  
 14819

```

9991:      JSR      PC,CKDROP                ;TRY TO DROP THE UNIT
;
;
;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 RBPCR (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
  
```



## TEST 8: WRITE/READ TAPE MARK

14867	107226				ERRHRD	ERRNO,T28RWN,PKTSSR		;REWIND NOT ACCEPTED		
	107226	104456							TRAP	C#ERHRD
	107230	001455							.WORD	813
	107232	112461							.WORD	T28RWN
	107234	012136							.WORD	PKTSSR
14868	107236				30#:	CKLOOP		;LOOP IF SELECTED		
	107236	104406							TRAP	C#CLP1
14869	107240	013701	111370			MOV	T28BFR+6,R1	;PICK UP XSTO		
14870	107244	010102				MOV	R1,R2	;SET UP EXPECTED		
14871	107246	052702	000002			BIS	#BIT1,R2	;SET BOT BIT IN EXPECTED		
14872	107252	020102				CMP	R1,R2	;DOES EXP = REC'D		
14873	107254	001406				BEQ	40#	;BR, IF EQUAL (OK)		
14874	107256	005237	002212			INC	FATFLG	;BUMP COUNT		
14878	107262					ERRHRD	ERRNO,T28BOT,EXPREC	;TAPE NOT AT BOT AFTER REWIND		
	107262	104456							TRAP	C#ERHRD
	107264	001456							.WORD	814
	107266	112337							.WORD	T28BOT
	107270	015564							.WORD	EXPREC
14879	107272				40#:	CKLOOP		;LOOP IF SELECTED		
	107272	104406							TRAP	C#CLP1
14880	107274	005737	002216		42#:	TST	EXTFEA	;CHECK FOR EXTENDED FEATURES SW SWITCH		
14881	107300	001024				BNE	50#	;BR IF SWITCH IS ON		
14882	107302	112737	000200	111501		MOVB	#200,T28BS1	;WRITE MISCELLANEOUS CONT/READ STATUS		
14883	107310	112737	000010	111500		MOVB	#10,T28BS0	;FUNC. SEL. BIT (TURN ON EXTFEA SWITCH)		
14884	107316	012704	111450			MOV	#T28PK2,R4	;WRITE SUBSYS MEM PACKET		
14885	107322	010465	000000			MOV	R4,TSDB(R5)	;ISSUE COMMAND		
14886	107326	004737	016426			JSR	PC,CHKTSSR	;WAIT FOR SSR		
14887	107332	103407				BCS	50#	;BR, IF NO ERROR		
14888	107334	010001				MOV	R0,R1	;ERROR, SAVE TSSR		
14889	107336	005237	002212			INC	FATFLG	;BUMP COUNT		
14893	107342					ERRHRD	ERRNO,T28SSR,PKTSSR	;TSSR NOT CORRECT AFTER WRT. MISCELLANEOUS		
	107342	104456							TRAP	C#ERHRD
	107344	001457							.WORD	815
	107346	112175							.WORD	T28SSR
	107350	012136							.WORD	PKTSSR
14894	107352				50#:	CKLOOP		;LOOP IF SELECTED		
	107352	104406							TRAP	C#CLP1
14895	107354	012737	000007	111360		MOV	#7,T28DSW	;SET UP DRIVE NUMBER		
14896	107362	012704	111340			MOV	#T28PACKET,R4	;SUBROUTINE NEEDS PACKET ADDRESS		
14897	107366	004737	010752			JSR	PC,WRTCHR	;ISSUE WRITE CHARACTERISTICS		
14898	107372	103407				BCS	60#	;BR, IF COMMAND ISSUED OK		
14899	107374	005237	002212			INC	FATFLG	;BUMP COUNT		
14903	107400	010001				MOV	R0,R1	;SAVE CONTENTS OF TSSR		
14904	107402					ERRHRD	ERRNO,WRTMSG,SFIMSG	;WRITE CHARACTERISTIC FAILED		
	107402	104456							TRAP	C#ERHRD
	107404	001460							.WORD	816
	107406	005054							.WORD	WRTMSG
	107410	012124							.WORD	SFIMSG
14905	107412				60#:	CKLOOP		;SCOPE LOOP		
	107412	104406							TRAP	C#CLP1
14906	107414	016501	000002			MOV	TSSR(R5),R1	;GET TSSR CONTENTS		
14907	107420	032701	000100			BIT	#OFL,R1	;CHECK FOR THE OFFLINE BIT SET		
14908	107424	001006				BNE	65#	;BR, IF OFFLINE (GOOD)		
14909	107426	005237	002212			INC	FATFLG	;BUMP COUNT		
14913	107432					ERRDF	ERRNO,T28OFL,SFIMSG	;OFF LINE SHOULD HAVE BEEN SET (BAD)		
	107432	104455							TRAP	C#ERDF
	107434	001461							.WORD	817

## TEST 8: WRITE/READ TAPE MARK

	107436	112530						.WORD	T28OFL
	107440	012124						.WORD	SFIMSG
14914	107442			65:	CKLOOP		;LOOP IF SELECTED	TRAP	C:CLP1
	107442	104406							
14915	107444	013737	002172	111360	MOV	UNITN,T28DSW	;SET UP DRIVE NUMBER		
14916	107452	012704	111340		MOV	#T28PACKET,R4	;SUBROUTINE NEEDS PACKET ADDRESS		
14917	107456	004737	010752		JSR	PC,WRTCHR	;ISSUE WRITE CHARACTERISTICS		
14918	107462	103407			BCS	68:	;BR, IF COMMAND ISSUED OK		
14919	107464	005237	002212		INC	FATFLG	;BUMP COUNT		
14923	107470	010001			MOV	R0,R1	;SAVE CONTENTS OF TSSR		
14924	107472				ERRHRD	ERRNO,WRTMSG,SFIMSG	;WRITE CHARACTERISTIC FAILED		
	107472	104456						TRAP	C:ERHRD
	107474	001462						.WORD	818
	107476	005054						.WORD	WRTMSG
	107500	012124						WORD	SFIMSG
14925	107502			68:	CKLOOP		;LOOP IF SELECTED	TRAP	C:CLP1
	107502	104406							
14926	107504	012737	140011	111470	MOV	#140011,T28PK3	;WRITE TAPE MARK,ACK,CVC=1 COMMAND		
14927	107512	C12704	111470		MOV	#T28PK3,R4	;SET UP R4 WITH PACKET ADDRESS		
14928	107516	010465	000000		MOV	R4.TSDB(R5)	;ISSUE COMMAND		
14929	107522	004737	016340		JSR	PC,WAITF	;WAIT FOR SSR TO SET		
14930	107526	016501	000002		MOV	TSSR(R5),R1	;GET TSSR CONTENTS		
14931	107532	012702	000200		MOV	#SSR,R2	;SET UP EXPECTED		
14932	107536	020102			CMP	R1,R2	;ARE THEY EQUAL		
14933	107540	001406			BEQ	70:	;BR, IF OK		
14934	107542	005237	002212		INC	FATFLG	;BUMP COUNT		
14938	107546				ERRHRD	ERRNO,T28WDC,PKTSSR	;TSSR INCORRECT AFTER WRITE TAPE MARK		
	107546	104456						TRAP	C:ERHRD
	107550	001463						.WORD	819
	107552	112603						.WORD	T28WDC
	107554	012136						.WORD	PKTSSR
14939	107556			70:	CKLOOP		;LOOP IF SELECTED	TRAP	C:CLP1
	107556	104406							
14940	107560	013701	111370		MOV	T28BFR+6,R1	;PICK UP XSTO (VCK CHECK)		
14941	107564	010102			MOV	R1,R2	;SET UP EXPECTED		
14942	107566	042702	000020		BIC	#BIT4,R2	;VCK SHOULD BE 0		
14943	107572	020102			CMP	R1,R2	;IS VCK SET CORRECTLY		
14944	107574	001406			BEQ	80:	;BR, IF VCK IS CLEAR		
14945	107576	005237	002212		INC	FATFLG	;BUMP COUNT		
14949	107602				ERRHRD	EPRNO,T28VCK,EXPREC	;VCK WAS NOT CLEAR AFTER CVC=1		
	107602	104456						TRAP	C:ERHRD
	107604	001464						.WORD	820
	107606	112662						.WORD	T28VCK
	107610	015564						.WORD	EXPREC
14950	107612			80:	CKLOOP		;LOOP IF SELECTED	TRAP	C:CLP1
	107612	104406							
14951	107614	013701	111370		MOV	T28BFR+6,R1	;PICK UP XSTO (CHECK TMK)		
14952	107620	010102			MOV	R1,R2	;SET UP EXPECTED		
14953	107622	052702	100000		BIS	#BIT15,R2	;TMK SHOULD BE SET		
14954	107626	020102			CMP	R1,R2	;WAS TMK SET		
14955	107630	001406			BEQ	90:	;BR, IF TMK WAS SET		
14956	107632	005237	002212		INC	FATFLG	;BUMP COUNT		
14960	107636				ERRHRD	ERRNO,T28TMK,EXPREC	;TMK WAS NOT SET AFTER WRT TAPE MARK		
	107636	104456						TRAP	C:ERHRD
	107640	001465						.WORD	821
	107642	112735						.WORD	T28TMK
	107644	015564						.WORD	EXPREC

TEST 8: WRITE/READ TAPE MARK

14961	107646			90:	CKLOOP		:LOOP IF SELECTED		
	107646	104406						TRAP	C:CLP1
14962	107650	004737	011104		JSR	PC,REWIND	:CALL TAPE REWIND COMMAND		
14963	107654	103411			BCS	130:	:BR, IF NO PROBLEM		
14964	107656	010004			MOV	R0,R4	:SAVE PACKET ADDRESS		
14965	107660	016501	000002		MOV	TSSR(R5),R1	:GET TSSR STATUS		
14966	107664	005237	002212		INC	FATFLG	:BUMP COUNT		
14970	107670				ERRHRD	ERRNO,T28RWN,PKTSSR	:REWIND NOT ACCEPTED		
	107670	104456						TRAP	C:ERHRD
	107672	001466						.WORD	822
	107674	112461						.WORD	T28RWN
	107676	012136						.WORD	PKTSSR
14971	107700			130:	CKLOOP		:LOOP IF SELECTED		
	107700	104406						TRAP	C:CLP1
14972	107702	013701	111370		MOV	T28BFR+6,R1	:PICK UP XSTO		
14973	107706	010102			MOV	R1,R2	:SET UP EXPECTED		
14974	107710	052702	000002		BIS	#BIT1,R2	:SET BOT BIT IN EXPECTED		
14975	107714	020102			CMP	R1,R2	:DOES EXP = REC'D		
14976	107716	C01406			BEQ	140:	:BR, IF EQUAL (OK)		
14977	107720	005237	002212		INC	FATFLG	:BUMP COUNT		
14981	107724				ERRHRD	ERRNO,T28BOT,EXPREC	:TAPE NOT AT BOT AFTER REWIND		
	107724	104456						TRAP	C:ERHRD
	107726	001467						.WORD	823
	107730	112337						.WORD	T28BOT
	107732	015564						.WORD	EXPREC
14982	107734			140:	CKLOOP		:LOOP IF SELECTED		
	107734	104406						TRAP	C:CLP1
14983	107736	012703	000012		MOV	#10,R3	:NUMBER OF RECORDS TO WRITE TM		
14984	107742	012737	140011	111470	MOV	#140011,T28PK3	:WRITE TAPE MARK,ACK,CVC-1 COMMAND		
14985	107750	012704	111470		MOV	#T28PK3,R4	:SET UP R4 WITH PACKET ADDRESS		
14986	107754	010465	000000		MOV	R4,TSD8(R5)	:ISSUE COMMAND		
14987	107760	004737	016340		JSR	PC,WAITF	:WAIT FOR SSR TO SET		
14988	107764	016501	000002		MOV	TSSR(R5),R1	:PICK UP TSSR		
14989	107770	012702	000200		MOV	#SSR,R2	:SET UP EXPECTED (SSR ONLY)		
14990	107774	020102			CMP	R1,R2	:WAS STATUS GOOD		
14991	107776	001406			BEQ	165:	:BR, IF TERMINATION WAS GOOD		
14992	110000	005237	002212		INC	FATFLG	:BUMP COUNT		
14996	110004				ERRHRD	ERRNO,T28WDC,PKTSSR	:TSSR NOT CORRECT AFTER WRT TAPE M.		
	110004	104456						TRAP	C:ERHRD
	110006	001470						.WORD	824
	110010	112603						.WORD	T28WDC
	110012	012136						.WORD	PKTSSR
14997	110014			165:	CKLOOP		:LOOP IF SELECTED		
	110014	104406						TRAP	C:CLP1
14998	110016	013701	111370		MOV	T28BFR+6,R1	:PICK UP XSTO		
14999	110022	010102			MOV	R1,R2	:SET UP EXPECTED		
15000	110024	052702	100000		BIS	#BIT15,R2	:SET TMK BIT IN EXPECTED		
15001	110030	020102			CMP	R1,R2	:DOES EXP = REC'D		
15002	110032	001406			BEQ	180:	:BR, IF EQUAL (OK)		
15003	110034	005237	002212		INC	FATFLG	:BUMP COUNT		
15007	110040				ERRHRD	ERRNO,T28TMK,EXPREC	:TMK NOT SET AFTER WRT TAPE MARK		
	110040	104456						TRAP	C:ERHRD
	110042	001471						.WORD	825
	110044	112735						.WORD	T28TMK
	110046	015564						.WORD	EXPREC
15008	110050			180:	CKLOOP		:LOOP IF SELECTED		
	110050	104406						TRAP	C:CLP1



## TEST 8: WRITE/READ TAPE MARK

15009	110052	005303			DEC	R3		;BUMP COUNTER DOWN
15010	110054	001337			BNE	155#		;BR, IF LESS THAN 10 TAPE MARKS
15011	110056	012700	177777		MOV	#177777,R0		;VALUE TO WRITTEN TO MEMORY
15012	110062	004737	017512		JSR	PC,FILLMEM		;FILL MEM WITH ALL ONES
15013	110066	013737	003114	111472	MOV	FREE,T28WB		;STARTING READ BUFFER ADDRESS
15014	110074	012737	140401	111470	MOV	#140401,T28PK3		;READ REVERSE,ACK, COMMAND
15015	110102	012704	111470		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
15016	110106	013737	000024	111476	MOV	20.,T28SZ		;SET UP RECORD SIZE IN PACKET
15017	110114	010465	000000		MOV	R4,TSDB(R5)		;ISSUE COMMAND
15018	110120	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET
15019	110124	016501	000002		MOV	TSSR(R5),R1		;GET TSSR CONTENTS
15020	110130	012702	100204		MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED
15021	110134	020102			CMP	R1,R2		;ARE THEY EQUAL
15022	110136	001406			BEQ	200#		;BR, IF OK
15023	110140	005237	002212		INC	FATFLG		;BUMP COUNT
15027	110144				ERRHRD	ERRNO,T28RDF,PKTSSR		;TSSR INCORRECT AFTER WRITE DATA
	110144	104456					TRAP	C#ERHRD
	110146	001472					.WORD	826
	110150	111674					.WORD	T28RDF
	110152	012136					.WORD	PKTSSR
15028	110154			200#:	CKLOOP			;LOOP IF SELECTED
	110154	104406					TRAP	C#CLP1
15029	110156	013701	111370		MOV	T28BFR+6,R1		;PICK UP XSTO
15030	110162	010102			MOV	R1,R2		;SET UP EXPECTED
15031	110164	052702	100000		BIS	#BIT15,R2		;TMK SHOULD BE SET
15032	110170	020102			CMP	R1,R2		;IS TMK SET
15033	110172	001406			BEQ	210#		;BR, IF TMK WAS SET (GOOD)
15034	110174	005237	002212		INC	FATFLG		;BUMP COUNT
15038	110200				ERRHRD	ERRNO,T28RRM,EXPREC		;TMK NOT SET AFTER READ REV
	110200	104456					TRAP	C#ERHRD
	110202	001473					.WORD	827
	110204	113007					.WORD	T28RRM
	110206	015564					.WORD	EXPREC
15039	110210			210#:	CKLOOP			;LOOP IF SELECTED
	110210	104406					TRAP	C#CLP1
15040	110212	017701	072676		MOV	#FREE,R1		;FIRST LOC IN READ BUFFER
15041	110216	012702	177777		MOV	#177777,R2		;EXPECTED IF NO DATA TRANS.
15042	110222	020102			CMP	R1,R2		;DID ANY DATA GET TRANSFERRED
15043	110224	001406			BEQ	220#		;BR, IF NO DATA TRANS (GOOD)
15044	110226	005237	002212		INC	FATFLG		;BUMP COUNT
15048	110232				ERRHRD	ERRNO,T28DTR,EXPREC		;DATA TRANSFERRED ON READ TAPE MARK
	110232	104456					TRAP	C#ERHRD
	110234	001474					.WORD	828
	110236	113222					.WORD	T28DTR
	110240	015564					.WORD	EXPREC
15049	110242			220#:	CKLOOP			;LOOP IF SELECTED
	110242	104406					TRAP	C#CLP1
15050	110244	012737	100410	111470	MOV	#100410,T28PK3		;SPACE REVERSE,ACK, COMMAND
15051	110252	012737	000001	111472	MOV	#1,T28RB		;NUMBER OF RECORDS TO SPACE BACK
15052	110260	012704	111470		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
15053	110264	010465	000000		MOV	R4,TSDB(R5)		;ISSUE COMMAND
15054	110270	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET
15055	110274	016501	000002		MOV	TSSR(R5),R1		;GET TSSR CONTENTS
15056	110300	012702	100204		MOV	#SSR!SC!BIT2,R2		;SET UP EXPECTED
15057	110304	020102			CMP	R1,R2		;ARE THEY EQUAL
15058	110306	001406			BEQ	222#		;BR, IF OK
15059	110310	005237	002212		INC	FATFLG		;BUMP COUNT

## TEST 8: WRITE/READ TAPE MARK

15063	110314			ERRHRD	ERRNO,T28RDG,PKTSSR		;TSSR INCORRECT AFTER SPACE CMD.		
	110314	104456						TRAP	C#ERHRD
	110316	001475						.WORD	829
	110320	111755						.WORD	T28RDG
	110322	012136						.WORD	PKTSSR
15064	110324			222#:	CKLOOP		;LOOP IF SELECTED		
	110324	104406						TRAP	C#CLP1
15065	110326	013701	111370		MOV	T28BFR+6,R1	;PICK UP XSTO		
15066	110332	010102			MOV	R1,R2	;SET UP EXPECTED		
15067	110334	052702	100000		BIS	#BIT15,R2	;TMK SHOULD BE SET		
15068	110340	020102			CMP	R1,R2	;IS TMK SET		
15069	110342	001406			BEQ	226#	;BR, IF TMK WAS SET (GOOD)		
15070	110344	005237	002212		INC	FATFLG	;BUMP COUNT		
15074	110350				ERRHRD	ERRNO,T28RRN,EXPREC	;TMK NOT SET AFTER SPACE REV		
	110350	104456						TRAP	C#ERHRD
	110352	001476						.WORD	830
	110354	113065						.WORD	T28RRN
	110356	015564						.WORD	EXPREC
15075	110360			226#:	CKLOOP		;LOOP IF SELECTED		
	110360	104406						TRAP	C#CLP1
15076	110362	004737	011104		JSR	PC,REWIND	;CALL TAPE REWIND COMMAND		
15077	110366	103411			BCS	230#	;BR, IF NO PROBLEM		
15078	110370	010004			MOV	R0,R4	;SAVE PACKET ADDRESS		
15079	110372	016501	000002		MOV	TSSR(R5),R1	;GET TSSR		
15080	110376	005237	002212		INC	FATFLG	;BUMP COUNT		
15084	110402				ERRHRD	ERRNO,T28RWN,PKTSSR	;REWIND NOT ACCEPTED		
	110402	104456						TRAP	C#ERHRD
	110404	001477						.WORD	831
	110406	112461						.WORD	T28RWN
	110410	012136						.WORD	PKTSSR
15085	110412			230#:	CKLOOP		;LOOP IF SELECTED		
	110412	104406						TRAP	C#CLP1
15086	110414	013701	111370		MOV	T28BFR+6,R1	;PICK UP XSTO		
15087	110420	010102			MOV	R1,R2	;SET UP EXPECTED		
15088	110422	052702	000002		BIS	#BIT1,R2	;SET BOT BIT IN EXPECTED		
15089	110426	020102			CMP	R1,R2	;DOES EXP = REC'D		
15090	110430	001406			BEQ	240#	;BR, IF EQUAL (OK)		
15091	110432	005237	002212		INC	FATFLG	;BUMP COUNT		
15095	110436				ERRHRD	ERRNO,T28BOT,EXPREC	;TAPE NOT AT BOT AFTER REWIND		
	110436	104456						TRAP	C#ERHRD
	110440	001500						.WORD	832
	110442	112337						.WORD	T28BOT
	110444	015564						.WORD	EXPREC
15096	110446			240#:	CKLOOP		;LOOP IF SELECTED		
	110446	104406						TRAP	C#CLP1
15097	110450	012700	177777		MOV	#177777,R0	;VALUE TO WRITTEN TO MEMORY		
15098	110454	004737	017512		JSR	PC,FILLMEM	;FILL MEM WITH ALL ONES		
15099	110460	013737	003114	111472	MOV	FREE,T28RB	;STARTING READ BUFFER ADDRESS		
15100	110466	012737	100001	111470	MOV	#100001,T28PK3	;READ FORWARD,ACK, COMMAND		
15101	110474	012704	111470		MOV	#T28PK3,R4	;SET UP R4 WITH PACKET ADDRESS		
15102	110500	013737	000024	111476	MOV	20.,T28SZ	;SET UP RECORD SIZE IN PACKET		
15103	110506	010465	000000		MOV	R4,TSD8(R5)	;ISSUE COMMAND		
15104	110512	004737	016340		JSR	PC,WAITF	;WAIT FOR SSR TO SET		
15105	110516	016501	000002		MOV	TSSR(R5),R1	;GET TSSR CONTENTS		
15106	110522	012702	100204		MOV	#SSR!SC!BIT2,R2	;SET UP EXPECTED		
15107	110526	020102			CMP	R1,R2	;ARE THEY EQUAL		
15108	110530	001406			BEQ	245#	;BR, IF OK		

## TEST 8: WRITE/READ TAPE MARK

15109	110532	005237	002212		INC	FATFLG		;BUMP COUNT
15113	110536				ERRHRD	ERRNO,T28WDE,PKTSSR		;TSSR INCORRECT AFTER WRITE DATA
	110536	104456						TRAP C#ERHRD
	110540	001501						.WORD 833
	110542	112246						.WORD T28WDE
	110544	012136						.WORD PKTSSR
15114	110546			245#:	CKLOOP			;LOOP IF SELECTED
	110546	104406						TRAP C#CLP1
15115	110550	013701	111370		MOV	T28BFR+6,R1		;PICK UP XSTO
15116	110554	010102			MOV	R1,R2		;SET UP EXPECTED
15117	110556	052702	100000		BIS	#8BIT15,R2		;TMK SHOULD BE SET
15118	110562	020102			CMP	R1,R2		;IS TMK SET
15119	110564	001406			BEQ	247#		;BR, IF TMK WAS SET (GOOD)
15120	110566	005237	002212		INC	FATFLG		;BUMP COUNT
15124	110572				ERRHRD	ERRNO,T28RRP,EXPREC		;TMK NOT SET AFTER READ REV
	110572	104456						TRAP C#ERHRD
	110574	001502						.WORD 834
	110576	113144						.WORD T28RRP
	110600	C15564						.WORD EXPREC
15125	110602			247#:	CKLOOP			;LOOP IF SELECTED
	110602	104406						TRAP C#CLP1
15126	110604	017701	072304		MOV	#FREE,R1		;FIRST LOC IN READ BUFFER
15127	110610	012702	177777		MOV	#177777,R2		;EXPECTED IF NO DATA TRANS.
15128	110614	020102			CMP	R1,R2		;DID ANY DATA GET TRANSFERRED
15129	110616	001406			BEQ	250#		;BR, IF NO DATA TRANS (GOOD)
15130	110620	005237	002212		INC	FATFLG		;BUMP COUNT
15134	110624				ERRHRD	ERRNO,T28DTR,EXPREC		;DATA TRANSFERRED ON READ TAPE MARK
	110624	104456						TRAP C#ERHRD
	110626	001503						.WORD 835
	110630	113222						.WORD T28DTR
	110632	015564						.WORD EXPREC
15135	110634			250#:	CKLOOP			;LOOP IF SELECTED
	110634	104406						TRAP C#CLP1
15136	110636	012737	100410	111470	MOV	#100410,T28PK3		;SPACE REVERSE,ACK, COMMAND
15137	110644	012737	000005	111472	MOV	#5,T28R8		;NUMBER OF RECORDS TO SPACE BACK
15138	110652	012704	111470		MOV	#T28PK3,R4		;SET UP R4 WITH PACKET ADDRESS
15139	110656	010465	000000		MOV	R4,T28DB(R5)		;ISSUE COMMAND
15140	110662	004737	016340		JSR	PC,WAITF		;WAIT FOR SSR TO SET
15141	110666	016501	000002		MOV	T#SR(R5),R1		;GET TSSR CONTENTS
15142	110672	012702	100204		MOV	#SR#SC#BIT2,R2		;SET UP EXPECTED
15143	110676	020102			CMP	R1,R2		;ARE THEY EQUAL
15144	110700	001406			BEQ	260#		;BR, IF OK
15145	110702	005237	002212		INC	FATFLG		;BUMP COUNT
15149	110706				ERRHRD	ERRNO,T28RDG,PKTSSR		;TSSR INCORRECT AFTER SPACE REV CMD.
	110706	104456						TRAP C#ERHRD
	110710	001504						.WORD 836
	110712	111755						.WORD T28RDG
	110714	012136						.WORD PKTSSR
15150	110716			260#:	CKLOOP			;LOOP IF SELECTED
	110716	104406						TRAP C#CLP1
15151	110720	013701	111370		MOV	T28BFR+6,R1		;PICK UP XSTO
15152	110724	010102			MOV	R1,R2		;SET UP EXPECTED
15153	110726	052702	100000		BIS	#8BIT15,R2		;TMK SHOULD BE SET
15154	110732	020102			CMP	R1,R2		;IS TMK SET
15155	110734	001406			BEQ	270#		;BR, IF TMK WAS SET (GOOD)
15156	110736	005237	002212		INC	FATFLG		;BUMP COUNT
15160	110742				ERRHRD	ERRNO,T28RRN,EXPREC		;TMK NOT SET AFTER READ REV

## TEST 8: WRITE/READ TAPE MARK

	110742	104456							TRAP	C#ERHRD
	110744	001505							.WORD	837
	110746	113065							.WORD	T28RRN
	110750	015564							.WORD	EXPREC
15161	110752			270#:	CKLOOP					;LOOP IF SELECTED
	110752	104406							TRAP	C#CLP1
15162	110754	013701	111366		MOV	T28BFR+4,R1				;PICK UP RESIDUAL BYTE COUNTER
15163	110760	012702	000004		MOV	#4,R2				;SHOULD BE THE DIFFERENCE
15164	110764	020102			CMP	R1,R2				;IS COUNTER CORRECT
15165	110766	001406			BEQ	280#				;BR, IF COUNTER CORRECT
15166	110770	005237	002212		INC	FATFLG				;BUMP COUNT
15170	110774				ERRHRD	ERRNO,T28PBP,EXPREC				;RESIDUAL BYTE COUNTER NOT CORRECT
	110774	104456							TRAP	C#ERHRD
	110776	001506							.WORD	838
	111000	111611							.WORD	T28PBP
	111002	015564							.WORD	EXPREC
15171	111004				280#:	CKLOOP				;LOOP IF SELECTED
	111004	104406							TRAP	C#CLP1
15172	111006	C12737	100410	111470	MOV	#100410,T28PK3				;SPACE REVERSE,ACK, COMMAND
15173	111014	012737	000001	111472	MOV	#1,T28RB				;NUMBER OF RECORDS TO SPACE BACK
15174	111022	012704	111470		MOV	#T28PK3,R4				;SET UP R4 WITH PACKET ADDRESS
15175	111026	010465	000000		MOV	R4,TSDB(R5)				;ISSUE COMMAND
15176	111032	004737	016340		JSR	PC,WAITF				;WAIT FOR SSR TO SET
15177	111036	016501	000002		MOV	TSSR(R5),R1				;GET TSSR CONTENTS
15178	111042	012702	100204		MOV	#SSR!SC!BIT2,R2				;SET UP EXPECTED
15179	111046	020102			CMP	R1,R2				;ARE THEY EQUAL
15180	111050	001406			BEQ	290#				;BR, IF OK
15181	111052	005237	002212		INC	FATFLG				;BUMP COUNT
15185	111056				ERRHRD	ERRNO,T28RDG,PKTSSR				;TSSR INCORRECT AFTER SPACE CMD.
	111056	104456							TRAP	C#ERHRD
	111060	001507							.WORD	839
	111062	111755							.WORD	T28RDG
	111064	012136							.WORD	PKTSSR
15186	111066				290#:	CKLOOP				;LOOP IF SELECTED
	111066	104406							TRAP	C#CLP1
15187	111070	013701	111376		MOV	T28BFR+14,R1				;PICK UP XST3
15188	111074	010102			MOV	R1,R2				;SET UP EXPECTED
15189	111076	052702	000001		BIS	#BIT0,R2				;RIB SHOULD BE SET
15190	111102	020102			CMP	R1,R2				;IS RIB SET
15191	111104	001406			BEQ	300#				;BR, IF RIB WAS SET (GOOD)
15192	111106	005237	002212		INC	FATFLG				;BUMP COUNT
15196	111112				ERRHRD	ERRNO,T28RIB,EXPREC				;RIB NOT SET AFTER READ REV
	111112	104456							TRAP	C#ERHRD
	111114	001510							.WORD	840
	111116	111534							.WORD	T28RIB
	111120	015564							.WORD	EXPREC
15197	111122				300#:	CKLOOP				;LOOP IF SELECTED
	111122	104406							TRAP	C#CLP1
15198	111124	012737	100010	111470	MOV	#100010,T28PK3				;SPACE FORWARD,ACK, COMMAND
15199	111132	012737	000005	111472	MOV	#5,T28RB				;NUMBER OF RECORDS TO SPACE FORM.
15200	111140	012704	111470		MOV	#T28PK3,R4				;SET UP R4 WITH PACKET ADDRESS
15201	111144	010465	000000		MOV	R4,TSDB(R5)				;ISSUE COMMAND
15202	111150	004737	016340		JSR	PC,WAITF				;WAIT FOR SSR TO SET
15203	111154	016501	000002		MOV	TSSR(R5),R1				;GET TSSR CONTENTS
15204	111160	012702	100204		MOV	#SSR!SC!BIT2,R2				;SET UP EXPECTED
15205	111164	020102			CMP	R1,R2				;ARE THEY EQUAL
15206	111166	001406			BEQ	310#				;BR, IF OK

## TEST 8: WRITE/READ TAPE MARK

```

15207 111170 005237 002212           INC    FATFLG           ;BUMP COUNT
15211 111174           ERRHRD  ERRNO,T28RDF,EXPREC ;TSSR INCORRECT AFTER SPACE CMD.
           111174   104456                                     TRAP    C#ERHRD
           111176   001511                                     .WORD  841
           111200   111674                                     .WORD  T28RDF
           111202   015564                                     .WORD  EXPREC
15212 111204           310#:  CKLOOP           ;LOOP IF SELECTED               TRAP    C#CLP1
           111204   104406
15213 111206   013701 111370       MOV    T28BFR+6,R1       ;PICK UP XSTO
15214 111212   010102           MOV    R1,R2            ;SET UP EXPECTED
15215 111214   052702 100000       BIS    #BIT15,R2        ;TMK SHOULD BE SET
15216 111220   020102           CMP    R1,R2            ;IS TMK SET
15217 111222   001406           BEQ    320#             ;BR, IF TMK WAS SET (GOOD)
15218 111224   005237 002212           INC    FATFLG           ;BUMP COUNT
15222 111230           ERRHRD  ERRNO,T28RRP,EXPREC ;TMK NOT SET AFTER READ REV
           111230   104456                                     TRAP    C#ERHRD
           111232   001512                                     .WORD  842
           111234   113144                                     .WORD  T28RRP
           111236   C15564                                     .WORD  EXPREC
15223 111240           320#:  CKLOOP           ;LOOP IF SELECTED               TRAP    C#CLP1
           111240   104406
15224 111242   013701 111366       MOV    T28BFR+4,R1       ;PICK UP RESIDUAL BYTE COUNTER
15225 111246   012702 000004       MOV    #4.,R2           ;SHOULD BE THE DIFFERENCE
15226 111252   020102           CMP    R1,R2            ;IS COUNTER CORRECT
15227 111254   001406           BEQ    330#             ;BR, IF COUNTER CORRECT
15228 111256   005237 002212           INC    FATFLG           ;BUMP COUNT
15232 111262           ERRHRD  ERRNO,T28PBP,EXPREC ;RESIDUAL BYTE COUNTER NOT CORRECT
           111262   104456                                     TRAP    C#ERHRD
           111264   001513                                     .WORD  843
           111266   111611                                     .WORD  T28PBP
           111270   015564                                     .WORD  EXPREC
15233 111272           330#:  CKLOOP           ;LOOP IF SELECTED               TRAP    C#CLP1
           111272   104406
15234 111274           ENDSUB                  ;<<<<<<<<<<<<<<<<<< END SUBTEST >>>>>>>>>>>>>>>>>>>>>>>>>
           111274   104403                                     L10133:
15235 111276   023727 002212 000017       CMP    FATFLG,#15.        ;IS ERROR COUNT AT 25
15236 111304   103402           BLO    999#            ;BR, IF LESS THAN 25
15237 111306   004737 017272           JSR    PC,CKDROP        ;TRY TO DROP THE UNIT
15238 111312           999#:
15239           ;
15240           ;
15241           ;
15242 111312   004737 016546           JSR    PC,TSTLOOP       ;DO WE NEED TO ITERATE TEST
15243 111316   103002           BCC    163#            ;BR, IF NO LOOP REQUIRED
15244 111320   000137 106200           JMP    T28LOOP         ;EXECUTE AGAIN
15245 111324           163#:
15246 111324           EXIT    TST            ;ALL DONE THIS TEST
           111324   104432                                     TRAP    C#EXIT
           111326   002256                                     .WORD  L10130-.
15247
15248           ;*
15249           ;LOCAL STORAGE FOR THIS TEST
15250           ;-
15252 111330           .BLKB 10-<<.-TSV2&7>
15254 111340   T28PACKET:
15255 111340   100004           .WORD 100004           ;COMMAND PACKET FOR TEST
                                     ;WRITE CHARACTERISTICS COMMAND, WITH IE, ACK

```

## TEST 8: WRITE/READ TAPE MARK

15256	111342	111350	.WORD	T28DATA	; ADDRESS OF CHARACTERISTICS BLOCK
15257	111344	000000	.WORD	0	
15258	111346	000012	.WORD	10.	; STARTING VALUE OF BLOCK SIZE
15259	111350				; CHARACTERISTICS DATA BLOCK
15260	111350	111362	T28DATA: .WORD	T28BFR	; ADDRESS OF MESSAGE BUFFER
15261	111352	000000	.WORD	0	
15262	111354	000024	.WORD	20.	; LENGTH OF MESSAGE BUFFER
15263	111356	000000	.WORD	0	
15264	111360	000000	T28DSW: .WORD	0	; SELECT DRIVE 0
15265	111362		T28BFR: .BLKW	25.	; MESSAGE BUFFER
15266					
15267					; WRITE SUBSYSTEM MEMORY COMMAND PACKET
15268					
15270	111444		.BLKB	10-<.-TSV2&7>	
15272	111450		T28PK2: .WORD	100006	; WRITE SUB SYS MEM COMMAND, IE AND ACK
15273	111450	100006	.WORD	T28BF2	; ADDRESS OF SELECT BLOCK DATA
15274	111452	111500	.WORD	0	
15275	111454	000000	.WORD	6.	; SIZE OF DATA PACKET
15276	111456	000006			
15277			.BLKB	10-<.-TSV2&7>	
15279	111460		T28PK3: .WORD	100005	; REREAD COMMAND, AND ACK
15281	111470		.WORD	100005	
15282	111470	100005	T28RB: .WORD	FREE	; ADDRESS OF WRITE BUFFER
15283	111472		.WORD	0	
15284	111472	003114	T28SZ: .WORD	0	; SIZE OF BUFFER (EXTENT)
15285	111474	000000	.EVEN		
15286	111476	000000			
15287					
15288					
15289					
15290					
15291	111500		T28BF2: .BYTE	10	; BSELO AREA
15292	111500	010	T28BS0: .BYTE	200	; BSEL1 AREA
15293	111501	200	T28S1: .WORD	0	; SEL 2 AREA
15294	111502	000000	T28S2: .WORD	0	; DATA AREA
15295	111504	000000	T28S3: .WORD	0	
15296					
15297					
15298			.EVEN		
15299					; TAPE MOTION PACKET COMMAND VALUES
15300					
15301	111506		T28IMV: .WORD	101411	; ILLEGAL MODE BITS TEST DATA
15302	111506	101411	.WORD	102011	
15303	111510	102011	.WORD	103411	
15304	111512	103411	.WORD	177777	
15305	111514	177777	T28RN: .WORD	100011	; WRITE TAPE MARK COMMAND
15306	111516	100011	T28WDR: .WORD	100411	; ERASE COMMAND
15307	111520	100411	T28CON: .WORD	101011	; WRITE TAPE MARK RETRY
15308	111522	101011	.WORD	177777	; END OF DATA
15309	111524	177777			
15310					
15311					
15312	111526	000000	T28CNT: .WORD	0	; TAPE TIMER COUNTER STORAGE AREA
15313	111530	000000	T28CNU: .WORD	0	; TAPE TIMER COUNTER STORAGE AREA
15314	111532	000000	T28DLY: .WORD	0	; DELAY COUNTER
15315			.EVEN		
15316					

## TEST 8: WRITE/READ TAPE MARK

```

15317
15318
15319
15320
15321
15322 111534 124 141 160 T28RIB: .ASCIZ 'Tape Position Not Correct, RIB Should Be Set'
15323 111611 122 145 163 T28PBP: .ASCIZ 'Residual Byte Counter Register (RBPCR) Not Correct'
15324 111674 124 123 123 T28RDF: .ASCIZ 'TSSR Incorrect After READ REVERSE Into TAPE MARK'
15325 111755 124 123 123 T28RDG: .ASCIZ 'TSSR Incorrect After SPACE Command Into TAPE MARK'
15326 112037 124 123 123 T28WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
15327 112114 111 154 154 T28LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
15328 112175 127 122 111 T28SSR: .ASCIZ 'WRITE MISCELLANEOUS Command Not Accepted'
15329 112246 124 123 123 T28WDE: .ASCIZ 'TSSR Not Correct After READ DATA Command, Into TAPE MARK'
15330 112337 124 141 160 T28BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
15331 112404 124 123 123 T28TM: .ASCIZ 'TSSR Not Correct After FORMAT Command Reject'
15332 112461 122 145 167 T28RWN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
15333 112530 104 162 151 T28OFL: .ASCIZ 'Drive 7 Select Failed To Set "OFL" In TSSR'
15334 112603 124 123 123 T28WDC: .ASCIZ 'TSSR Not Correct After WRITE TAPE MARK Command'
15335 112662 103 126 103 T28VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
15336 112735 124 115 113 T28TMK: .ASCIZ 'TMK Not Set After WRITE TAPE MARK Command'
15337 113007 124 115 113 T28RRM: .ASCIZ 'TMK Not Set After READ REVERSE Into TAPE MARK'
15338 113065 124 115 113 T28RRN: .ASCIZ 'TMK Not Set After SPACE REVERSE Into TAPE MARK'
15339 113144 124 115 113 T28RRP: .ASCIZ 'TMK Not Set After READ FORWARD Into TAPE MARK'
15340 113222 104 141 164 T28DTR: .ASCIZ 'Data Transferred On READ REVERSE Into A TAPE MARK'
15341 113304 104 141 164 T28DTA: .ASCIZ 'Data Compare Error, Data Read From Tape Not Equal To Written'
15342 113401 127 162 151 TST28ID: .ASCIZ 'Write/Read Tape Mark'
15343 .EVEN
15344
15345
15346
15347
15348
15349
15350
15351 113426
15352 113426
15353 113432 012701 111340
15354 113436 012721 100004
15355 113442 012721 111350
15356 113446 005021
15357 113450 012721 000012
15358 113454 012721 111362
15359 113460 005021
15360 113462 012721 000024
15361 113466 005021
15362 113470 012711 000000
15363 113474 012702 000030
15364 113500 012762 177777 111362 64:
15365 113506 005742
15366 113510 020227 000000
15367 113514 001371
15368 113516 000207
15369
15370
15371 113520
15372 113520
15373 113524 012701 111450

;*
;LOCAL TEXT MESSAGES FOR TEST
;-

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

T28REST:
SAVREG
MOV #T28PACKET,R1 ;SAVE THE REGISTERS
MOV #100004,(R1)+ ;START OF THE PACKET
MOV #T28DATA,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK,
CLR (R1)+ ;ADDRESS OF CHARAISTICS DATA BLOCK
MOV #10.,(R1)+ ;EXTENDED ADDRESS
MOV #T28BFR,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
CLR (R1)+ ;ADDRESS OF MESSAGE BUFFER
MOV #20.,(R1)+ ;LENGTH OF MESSAGE BUFFER
CLR (R1)+
MOV #0,(R1) ;SELECT DRIVE ZERO
MOV #24.,R2 ;NUMBER OF LOCATIONS TO BE CLEARED
MOV #177777,T28BFR(R2) ;ALL ONES TO MESSAGE BUFFER
TST -(R2) ;NEXT LOCATION
CMP R2,#0 ;CHECK FOR END
BNE 64$ ;KEEP GOING UNTIL DONE
RTS PC ;RETURN

T28RT2:
SAVREG
MOV #T28PK2,R1 ;SAVE THE REGISTERS
;START OF THE PACKET

```

TEST 8: WRITE/READ TAPE MARK

15374	113530	012721	100006	MOV	#100006,(R1)+	;WRITE SUBSYSTEM MEM. WITH ACK.
15375	113534	012721	111500	MOV	#T28BF2,(R1)+	;ADDRESS OF DATA BLOCK
15376	113540	005021		CLR	(R1)+	;EXTENDED ADDRESS
15377	113542	012721	000006	MOV	#6,(R1)+	;SIZE OF DATA BLOCK IN BYTES
15378	113546	005021		CLR	(R1)+	
15379	113550	012701	111500	MOV	#T28BF2,R1	;POINT TO DATA SEL AREA
15380	113554	005021		CLR	(R1)+	
15381	113556	005011		CLR	(R1)	
15382	113560	000207		RTS	PC	;RETURN
15383	113562			T28RT3:		
15384	113562			SAVREG		
15385	113566	012701	111470	MOV	#T28PK3,R1	;GET PACKET ADDRESS
15386	113572	005021		CLR	(R1)+	;CLEAR COMMAND AREA
15387	113574	005021		CLR	(R1)+	;CLEAR ADDRESS AREA
15388	113576	005021		CLR	(R1)+	;CLEAR EXTENDED ADDRESS AREA
15389	113600	005011		CLR	(R1)	;SIZE OF DATA TRANSFER
15390	113602	000207		RTS	PC	;RETURN
15391	113604			ENDTST		
	113604					
	113604	104401				L10130: TRAP C#ETST
15392	113606			ENDMOD		
15393				.TITLE	TSV6 - PARAMETER CODING	
15399						
15404						
15410						
15411	113606			BGNMOD	TSV6	
	113606			TSV6::		
15412				.SBTTL	HARDWARE PARAMETER CODING SECTION	
15413						
15414						
15415						
15416				;	THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS	
15417				;	THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE	
15418				;	MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE	
15419				;	INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE	
15420				;	MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS	
15421				;	WITH THE OPERATOR.	
15422				;		
15423	113606			;		
	113606	000010		---		
	113610			BGNHRD		
				.WORD	L10134-L#HARD/2	
15424				L#HARD::		
15425	113610			GPRMA	HPM1,0,0,160010,177776,YES	;GET TSBA/TSDB REGISTER ADDRESS.
	113610	000031		.WORD	T#CODE	
	113612	113630		.WORD	HPM1	
	113614	160010		.WORD	T#LOLIM	
	113616	177776		.WORD	T#HILIM	
15426	113620			GPRMA	HPM2,2,0,0,776,YES	;GET VECTOR ADDRESS.
	113620	001031		.WORD	T#CODE	
	113622	113664		.WORD	HPM2	
	113624	000000		.WORD	T#LOLIM	
	113626	000776		.WORD	T#HILIM	
15427				;GPRMD	HPM3,4,0,340,0,7,YES	;GET INTERRUPT PRIORITY.
15428	113630			ENDHRD		
				.EVEN		
	113630			L10134:		
15429	113630	104	105	126	HPM1: .ASCIZ 'DEVICE ADDRESS (TSBA/TSDB) '	



HARDWARE PARAMETER CODING SECTION

15430	113664	111	116	124	HPM2:	.ASCIZ	'INTERRUPT VECTOR	'
15431	113710	111	116	124	HPM3:	.ASCIZ	'INTERRUPT PRIORITY	'
15432						.EVEN		

SOFTWARE PARAMETER CODING SECTION

```

15434                                     .SBTTL SOFTWARE PARAMETER CODING SECTION
15435
15436
15437                                     ;**
15438                                     ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
15439                                     ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
15440                                     ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
15441                                     ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
15442                                     ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
15443                                     ; WITH THE OPERATOR.
15444                                     ;--
15444 113740                                BGNSFT
15445 113740 000003                          .WORD L10135-L1SOFT/2
15446 113742
15445 113742                                L1SOFT::
15446 113742 001130                          GPRM1  SPH1.0,-1.YES           ; GET TRANSPORT TEST FLAG.
15447 113744 114000                          GPRM1  SPM4.2,-1.YES         ; GET ITERATION CONTROL.
15448 113746 177777                          .WORD  T%CODE
15449 113750                                .WORD  SPM4
15450                                     .WORD  -1
15451 113750                                GPRM6  SPH6.4,D.7777.0,7777.YES ; GET LOCAL ERROR LIMIT
15452 114000                                GPRM7  SPH7.6,D.7777.0,7777.YES ; GET GLOBAL ERROR LIMIT
15453 114030                                ENDSFT
15454 114060                                .EVEN
15455                                     L10135:
15456 113750                                SPH1:  .ASCIZ 'ENABLE TRANSPORT TESTS ;
15457 114000                                SPM4:  .ASCIZ 'INHIBIT ITERATIONS   ;
15458 114030                                SPM6:  .ASCIZ 'PER TEST ERROR LIMIT ;
15459 114060                                SPM7:  .ASCIZ 'PER UNIT ERROR LIMIT  ;
15460                                     .SBTTL PATCH AREA
15461                                     ;
15462                                     ; FINALLY A GENEROUS PATCH AREA.
15463                                     ;
15464                                     ; AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
15465                                     ; DESCRIBED IN "SUPPRG.MEM" (FOR REV C).
15466                                     ;
15467                                     PATCH::
15468 114110                                .BLKW  32.
15469 114400                                . = .!377.1
15470 114400 000000                          LASTAD                                ;SET LAST USED ADDRESS.
15471 114402 000000                          .EVEN
15472 114404                                .WORD  0
15473 114404                                .WORD  0
15474                                     L1LAST::
15475                                     ENDMOD
15476                                     .END

```

Symbol table

ADDSSR	012216	G	C#AU	=	000052	DEVDR0	024306	FREE	003114	G	INCERK	017134			
ADR	=	000020	C#AUTO	=	000061	DEVNRD	024225	FREEHI	003120		INTCPC	016240			
AMBTSS	006725		C#BRK	=	000022	DEVNXR	024143	FRESIZ	003116	G	INTFLA	016235			
ASSEMB	=	000010	C#BSEG	=	000004	DEVNXL	024073	FUSI	004115		INTMAS	016234			
A1716	=	000003	C#BSUB	=	000002	DEVSUM	024036	F#AU	=	000015	INTR	016306	G		
BADDAT	003146	G	C#CEFG	=	000045	DFPTBL	002146	F#AUTO	=	000020	INTREC	002214	G		
BADSSR	015770	G	C#CLCK	=	000062	DIAGMC	=	000000	F#BGN	=	000040	INTVEC	016236		
BDVPCR	=	177520	C#CLEA	=	000012	DICEC	=	000001	F#CLEA	=	000007	INTX	004276		
BENBSW	002220	G	C#CLOS	=	000035	DSBINT	016274	F#DU	=	000016	INVERT	021252	G		
BIE	=	040000	C#CLP1	=	000006	DUAD12	004641	F#END	=	000041	IOKCKI	=	000200		
BIT0	=	000001	C#CVEC	=	000036	DUFLG	003102	F#HARD	=	000004	IOKSTP	=	000001		
BIT00	=	000001	C#DCLN	=	000044	DUMMY	003052	F#HW	=	000013	IPRI	002202	G		
BIT01	=	000002	C#DODU	=	000051	EF.CON	=	000036	F#INIT	=	000006	ISR	=	000100	G
BIT02	=	000004	C#DRPT	=	000024	EF.NEW	=	000035	F#JMP	=	000050	IVEC	002200	G	
BIT03	=	000010	C#DU	=	000053	EF.PWR	=	000034	F#MOD	=	000000	IXE	=	004000	G
BIT04	=	000020	C#EDIT	=	000003	EF.RES	=	000037	F#MSG	=	000011	I#AU	=	000041	
BIT05	=	000040	C#ERDF	=	000055	EF.STA	=	000040	F#PROT	=	000021	I#AUTO	=	000041	
BIT06	=	000100	C#ERHR	=	000056	EMAXDU	017067	F#PWR	=	000017	I#CLN	=	000041		
BIT07	=	000200	C#ERRO	=	000060	EN	=	000000	F#RPT	=	000012	I#DU	=	000041	
BIT08	=	000400	C#ERSF	=	000054	ENAINI	016242	F#SEG	=	000003	I#HRD	=	000041		
BIT09	=	001000	C#ERSO	=	000057	ENVIRN	020720	F#SOFT	=	000005	I#INIT	=	000041		
BIT1	=	000002	C#ESCA	=	000010	EPRTSW	002170	F#SRV	=	000010	I#MOD	=	000041		
BIT10	=	002000	C#ESEG	=	000005	EPRT1	006356	F#SUB	=	000002	I#MSG	=	000041		
BIT11	=	004000	C#ESUB	=	000003	EPRT2	006446	F#SW	=	000014	I#PROT	=	000040		
BIT12	=	010000	C#ETST	=	000001	ERCM	012023	F#TEST	=	000001	I#PTAB	=	000041		
BIT13	=	020000	C#EXIT	=	000032	ERRHI	002226	GDDAT	003150	G	I#PWR	=	000041		
BIT14	=	040000	C#GETB	=	000026	ERRK	017046	GERRMA	002164	G	I#RPT	=	000041		
BIT15	=	100000	C#GETW	=	000027	ERRLO	002230	GETPAT	020264	G	I#SEG	=	000041		
BIT2	=	000004	C#GMAN	=	000043	ERRNO	=	001513	GETSEL	020346	G	I#SETU	=	000041	
BIT3	=	000010	C#GPHR	=	000042	ERRVEC	=	000004	G#CNT0	=	000200	I#SFT	=	000041	
BIT4	=	000020	C#GPLO	=	000030	ERTABE	003366	G#DELM	=	000372	I#SRV	=	000041		
BIT5	=	000040	C#GPRI	=	000040	ERTABL	003166	G#DISP	=	000003	I#SUB	=	000041		
BIT6	=	000100	C#INIT	=	000011	ESUM	017050	G#EXCP	=	000400	I#TST	=	000041		
BIT7	=	000200	C#INLP	=	000020	EVL	=	000004	G#HILI	=	000002	J#JMP	=	000167	
BIT8	=	000400	C#MANI	=	000050	EXBCNT	=	000010	G#LOLI	=	000001	KIPAR0	=	172340	
BIT9	=	001000	C#MEM	=	000031	EXIT	035204	G#NO	=	000000	KIPAR1	=	172342		
BOE	=	000400	C#MSG	=	000023	EXPBRE	015572	G#OFFS	=	000400	KIPAR2	=	172344		
BRINIT	004455		C#OPEN	=	000034	EXPD	002222	G#OF SI	=	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#PRML	=	000000	KIPAR6	=	172354		
CHKMAN	020570	G	C#PNTX	=	000015	EXPREC	015564	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#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#RFLA	=	000021	FATAL	035304	G#YES	=	000010	KIPDR5	=	172312		
CKRAM2	011314	G	C#RPT	=	000025	FATERR	=	000060	HIADDR	=	001400	KIPDR6	=	172314	
CMDPKT	022170	G	C#SEFG	=	000046	FATFLG	002212	HOE	=	100000	KIPDR7	=	172316		
CMPMEM	017750		C#SPRI	=	000041	FERCM	012012	HPM1	=	113630	KTENAB	=	003124	G	
CONFIG	017340		C#SVEC	=	000037	FIFEXP	012260	HPM2	=	113664	KTFLG	=	003122	G	
COUNT	002300	G	C#TPRI	=	000013	FIF1MS	012332	HPM3	=	113710	KTINIT	=	021100		
CSRADD	002176	G	DATA	=	002302	FIF2MS	012401	IBE	=	010000	KTOFF	=	017364		
CTAB	003154	G	DATASC	=	020322	FILLME	017512	IDU	=	000040	KTON	=	017346		
CTABE	003166	G	DEBUGM	=	011722	FNOINT	004213	IER	=	020000	LERRMA	=	002162	G	
CTABM	003154	G	DEVcnt	=	002210	FORCER	002166	IFault	=	004254	LISTAL	=	000001		

Symbol table

LDE	=	040000	G	L#UNIT	G02012	G	L10071	056522	M8189	005643	PRBEXP	015560				
LOOPCN		002206	G	L10000	002154		L10072	050564	NBA	=	002000	PRBMSG	015426			
LOOPCO		013216		L10001	002166		L10073	051164	NEWPAS		022720	PRBREC	015562			
LOOPFL		003152	G	L10002	005764		L10074	051640	NODEV		003104	PRBTOT	015513			
LOT	=	000J10	G	L10003	012134		L10075	052304	NOINIT		004333	PRBYTE	015212	G		
L#ACP		002110	G	L10004	012152		L10076	053044	NOINTR		004217	PRI	=	002000	G	
L#APT		002036	G	L10005	012170		L10077	054004	NOITS		002160	PRIADD		010250		
L#AU		023262	G	L10006	012176		L10100	054324	NOMAN		020624	PRIAO		010320		
L#AUT		002070	G	L10007	012214		L10101	054726	NOMEM		005456	PRIBXO		007702	G	
L#AUTO		023466	G	L10010	012232		L10102	076104	NP.IR	=	000200	PRIEQU		010150		
L#CCP		002106	G	L10011	012256		L10103	057464	NP.LOO	=	000040	PRIPKT		007460	G	
L#CLEA		023546	G	L10012	012330		L10104	060332	NP.OUT	=	000100	PRIRAM		010156		
L#CO		002032	G	L10013	012500		L10105	061224	NP.WRP	=	000020	PRITAD		010364		
L#DEPO		002011	G	L10014	013214		L10106	062152	NSI		004150	PRITSS		006022		
L#DESC		003400	G	L10015	014042		L10107	062730	NSINIT		004405	PRITO		010446		
L#DESP		002076	G	L10016	014064		L10110	063572	NUL		004525	PRIT1		010511		
L#DEVP		002060	G	L10017	015570		L10111	064444	NULCR		004526	PRI XOR		010032	G	
L#DISP		002124	G	L10020	015576		L10112	065316	NXM	=	004000	PRI00	=	000000	G	
L#DLY		002116	G	L10021	015604		L10113	066172	NXMFLG		003126	PRI01	=	000040	G	
L#DTP		002040	G	L10022	015616		L10114	067046	NXMHI		003132	PRI02	=	000100	G	
L#DTYP		002034	G	L10023	015640		L10115	067716	NXMLO		003130	PRI03	=	000140	G	
L#DU		023360	G	L10024	015666		L10116	070650	NXMTST		022362	PRI04	=	00C200	G	
L#DUT		002072	G	L10025	016026		L10117	071700	NXR		003736	PRI05	=	000240	G	
L#DVTY		003372	G	L10026	016336		L10120	072260	NXRERR		005734	PRI06	=	000300	G	
L#EF		002052	G	L10030	023212		L10121	072734	NXRX		003775	PRI07	=	000340	G	
L#ENVI		002044	G	L10031	023356		L10122	106142	NXTU		022732	PRMESS		014332		
L#ETP		002102	G	L10032	023464		L10123	076526	OFL	=	000100	PRMNO		002310	G	
L#EXP1		002046	G	L10033	023544		L10124	077310	ONEFIL	=	000000	PRMSGE		014642	G	
L#EXP4		002064	G	L10034	023572		L10125	100132	O#APTS	=	000000	PRMSGO		015022		
L#EXP5		002066	G	L10035	024034		L10126	101034	O#AU	=	000001	PRMSG1		015067		
L#HARD		113610	G	L10036	025470		L10127	102564	O#BGNR	=	000001	PRMSG2		015125		
L#HIME		002120	G	L10037	030150		L10130	113604	O#BGNS	=	000001	PROASC		014510		
L#HPCP		002016	G	L10040	026076		L10131	106542	O#DU	=	000001	PRIASC		014555		
L#HPTP		002022	G	L10041	026420		L10132	107022	O#ERRT	=	000000	PST32W		003142	G	
L#HW		002146	G	L10042	027000		L10133	111274	O#GNSW	=	000001	PUNIT		023214		
L#ICP		002104	G	L10043	035330		L10134	113630	O#POIN	=	000001	PW.D11	=	J00021		
L#INIT		022466	G	L10044	030556		L10135	113750	O#SETU	=	000000	PW.D13	=	000022		
L#LADP		002026	G	L10045	031426		MEMADD	014044	PASRPT		022764	PW.D22	=	000020		
L#LAST		114404	G	L10046	032246		MEMCK	022206	PATCH		114110	PW.NOP	=	000000		
L#LOAD		002100	G	L10047	032462		MENASC	020537	PATDAT		020320	PW.NOI	=	000023		
L#LUN		002074	G	L10050	033030		MENERR	020464	PC.ERA	=	002400	PW.RDE	=	000024		
L#MREV		002050	G	L10051	033374		MENRES	020566	PC.IER	=	002000	PW.RDR	=	000001		
L#NAME		002000	G	L10052	047574		MMVEC	=	000250	PC.NOO	=	001000	PW.RDS	=	000005	
L#PRIO		002042	G	L10053	036002		MSA.FR	=	000006	PC.REL	=	000000	PW.RFI	=	000003	
L#PROT		022456	G	L10054	036562		MSA.NO	=	000000	PC.REW	=	000400	PW.WCT	=	000006	
L#PRT		002112	G	L10055	037336		MSA.NR	=	000004	PKBCNT	=	000006	PW.WFI	=	000004	
L#REPP		002062	G	L10056	040040		MSA.VO	=	000002	PKHI	=	J00004	PW.WFM	=	000007	
L#REV		002010	G	L10057	040504		MSGEXP	012234	PKLOW	=	000002	PW.WMI	=	000010		
L#RPT		023574	G	L10060	041140		MSGLOO	013154	PKTADD		007644	PW.WNP	=	000011		
L#SOFT		113742	G	L10061	041574		MSGSTA	012440	PKTFRM		007606	PW.WTR	=	000002		
L#SPC		002056	G	L10062	042166		MSGSUB	014032	PKTGET		012154	P.ACK	=	100000		
L#SPCP		002020	G	L10063	042670		MS.ATT	=	000006	PKTHES		012200	P.CMD	=	000037	
L#SPTP		002024	G	L10064	043134		MS.EXT	=	000200	PKTRAM		004743	P.CONT	=	000012	
L#STA		002030	G	L10065	043406		MS.RSD	=	000001	PKTSSR		012136	P.CVC	=	040000	
L#SW		002156	G	L10066	043672		MS.RSF	=	000020	PNT	=	001000	P.FMT	=	000140	
L#TEST		002114	G	L10067	044172		MS.RST	=	000010	PRAMPK		014066	P.FORM	=	000011	
L#TIML		002014	G	L10070	044656		M8186	005552	PRASC		014613	P.GETS	=	000017		

Symbol table

P.IE	=	000200	SPM6	=	114030	TSREJ	=	000006	T##CLE	=	010034	T22WRT	=	027170
P.INIT	=	000013	SPM7	=	114060	TSSDEF	=	006676	T##DU	=	010032	T23A	=	003134 G
P.MODE	=	007400	SR0	=	177572	TSSR	=	000002 G	T##HAR	=	010134	T23AM3	=	034220
P.OPP	=	020000	SR1	=	177574	TSSRBI	=	003500 G	T##HW	=	010000	T23B	=	003136 G
P.POSI	=	000J10	SR2	=	177576	TSSRFO	=	006505	T##INI	=	010030	T23BA	=	034605
P.READ	=	000001	SR3	=	172516	TSSRH	=	000003 G	T##MSG	=	010025	T23BFR	=	033462
P.SWB	=	010000	SSR	=	000200	TSSX	=	004016	T##PRO	=	010027	T23BF2	=	033602
P.WRIT	=	000005	STATCO	=	012502	TSTBLK	=	002742 G	T##RPT	=	010035	T23BS0	=	033602
P.WRTC	=	000004	SVCGBL	=	000000	TSTCNT	=	002204 G	T##SOF	=	010135	T23BS1	=	033603
P.WRTS	=	000006	SVCINS	=	000000	TSTEND	=	017010	T##SRV	=	010026	T23CHK	=	035142
QVP	=	002174 G	SVCSUB	=	000001	TSTFLA	=	002304 G	T##SUB	=	010133	T23CON	=	033620
RAMASC	=	014246	SVCTAG	=	000000	TSTL00	=	016546 G	T##SW	=	010001	T23DAT	=	033450
RAMDAT	=	002232 G	SVCTST	=	000001	TSTPTR	=	002306 G	T##TES	=	010130	T23DSW	=	033460
RAMERR	=	015600 G	S#LSYM	=	010000	TSTSET	=	016600 G	T1	=	024356 G	T23EOT	=	033744
RAMEXP	=	015620 G	SO.IDB	=	000010	TST21I	=	025314	T2	=	025472 G	T23ET	=	033657
RAMFOR	=	010206	SO.IFB	=	000002	TST22I	=	027757	T2.1	=	025522	T23L00	=	030222
RAMSIZ	=	002272 G	SO.IFP	=	000001	TST23I	=	034746	T2.2	=	026114	T23OFL	=	034266
RAMTAD	=	015606 G	SO.ILD	=	000020	TST24I	=	047342	T2.3	=	026436	T23PAC	=	033440
RCVHIA	=	002274 G	SO.ION	=	000040	TST25I	=	056320	T21AM3	=	025173	T23PK2	=	033550
RCVLOA	=	002276 G	SO.IRD	=	000100	TST26I	=	075707	T21BFR	=	024774	T23PK3	=	033570
RDERR	=	005204	SO.IRW	=	000004	TST27I	=	105743	T21BF2	=	025070	T23RES	=	034762
RECMG	=	002456 G	SO.ISP	=	000200	TST28I	=	113401	T21BS0	=	025070	T23RNC	=	034145
RECV	=	002224 G	S1.ICE	=	002000	TSV2	=	002000 G	T21BS1	=	025071	T23RSZ	=	033600
REGSAV	=	020230	S1.IEO	=	010000	TSV3	=	002166 G	T21DAT	=	024760	T23RT2	=	035054
RETErr	=	005370	S1.IFM	=	001000	TSV4	=	022456 G	T21DLY	=	024772	T23RT3	=	035116
RETRY	=	035206	S1.IHE	=	000400	TSV6	=	113606 G	T21DSW	=	024770	T23RWN	=	034076
REWIND	=	011104 G	S1.IID	=	004000	TSV7	=	024356 G	T21L00	=	024406	T23SSR	=	033624
RMCHBE	=	000167	S1.IIR	=	020000	TTIBFR	=	177562 G	T21OFL	=	025273	T23SZ	=	033576
RMCHEN	=	000200	S1.IZR	=	040000	TTICSR	=	177560 G	T21PAC	=	024750	T23S2	=	033604
RMMSGB	=	000215	S1.PAR	=	100000	TTIVEC	=	000060 G	T21PK2	=	025060	T23S3	=	033606
RMSGE	=	000234	S2.ATI	=	000010	T#ARGC	=	000003	T21RES	=	025336	T23TM	=	034022
RMPKTB	=	000201	S2.BTI	=	000004	T#CODE	=	001130	T21RT2	=	025426	T23TMP	=	033610
RMPKTE	=	000210	S2.DIM	=	000200	T#ERRN	=	001513	T21SSR	=	025076	T23VCK	=	034532
RMR	=	010000	S2.ILW	=	000100	T#EXCP	=	000000	T21S2	=	025072	T23WB	=	033572
RMPACK	=	011200	S2.INR	=	000020	T#FLAG	=	000040	T21S3	=	025074	T23WD	=	033614
SC	=	100000	S2.OUT	=	000040	T#GMAN	=	000000	T22AM3	=	027275	T23WDC	=	034430
SCE	=	020000	S2.UND	=	000003	T#HILI	=	000776	T22BFR	=	027062	T23WDD	=	034341
SCHERR	=	005276	TBLEND	=	003052 G	T#LAST	=	000001	T22BF2	=	027160	T23WDR	=	033616
SCME	=	005011	TCOASC	=	006566	T#LOLI	=	000000	T22BS0	=	027160	T23WRT	=	033612
SDELAY	=	010750	TCOCOD	=	006766	T#LSYM	=	010000	T22BS1	=	027161	T23WSS	=	034657
SELASC	=	020532	TEMP1	=	003106 G	T#LTNO	=	000010	T22DAT	=	027050	T24AM3	=	046330
SELDAT	=	000004	TEMP2	=	003110 G	T#NEST	=	177777	T22FOR	=	027174	T24BA	=	046662
SEL2	=	000002	TERCLS	=	000016	T#NS0	=	000000	T22L00	=	025522	T24BFR	=	044742
SETMAP	=	017406	TESTNO	=	000010	T#NS1	=	000005	T22OFL	=	027375	T24BF2	=	045060
SETU	=	023016	TEXASC	=	006525	T#NS2	=	000002	T22PAC	=	027040	T24BOT	=	045723
SFFMSG	=	012172 G	TFCASC	=	006627	T#PTNU	=	000000	T22PK2	=	027150	T24BS0	=	045060
SFHERR	=	003703	TIMEXP	=	015642 G	T#SAVL	=	177777	T22POS	=	027172	T24BS1	=	045061
SFIERR	=	003650	TJMSGO	=	015670	T#SEGL	=	177777	T22RD	=	027166	T24CON	=	045072
SFIMSG	=	012124 G	TINERR	=	012111	T#SUBN	=	000003	T22RES	=	030012	T24DAT	=	044730
SFPTBL	=	002156 G	TMPBFR	=	002622 G	T#TAGL	=	177777	T22RT2	=	030104	T24DLY	=	045076
SIFLAG	=	003144 G	TNAM	=	016774	T#TAGN	=	010136	T22RWJ	=	027544	T24DSW	=	044740
SIMSG	=	012056	TRANST	=	002156 G	T#TEMP	=	000000	T22SSR	=	027200	T24DTA	=	045770
SKIPT	=	003370	TSBA	=	000000 G	T#TEST	=	000010	T22S2	=	027162	T24EOT	=	046056
SOFINI	=	016064 G	TSBAH	=	000001 G	T#TSTM	=	177777	T22S3	=	027164	T24ILA	=	045452
SPACE	=	010556 G	TSDB	=	000000 G	T#TSTS	=	000001	T22TM	=	027450	T24LON	=	047022
SPM1	=	113750	TSDBH	=	000001 G	T#AU	=	010031	T22VCK	=	027617	T24L00	=	035376
SPM4	=	114000	TSFCOD	=	007326	T#AUT	=	010033	T22WLK	=	027672	T24LOP	=	047104

Symbol table

T24LOQ	045536	T25SSR	055154	T26WDC	075020	T27WDR	103000	T39BFR	021436
T24LOR	045152	T25SZ	055126	T26WDD	074730	T27WNG	103014	T39BS0	021430
T24NEF	045100	T25S2	055132	T26WDE	074123	T27WRF	105566	T39BS1	021431
T24NXM	045311	T25S3	055134	T26WDF	073731	T27WSS	104722	T39BS2	021432
T24OFL	046375	T25TM	055362	T26WNG	073166	T28BFR	111362	T39DLY	021414
T24PAC	044720	T25WB	055122	T26WSS	075221	T28BF2	111500	T39DSW	022140
T24PBP	047166	T25WDC	056247	T27AM3	104307	T28B0T	112337	T39DTA	022130
T24PK2	045030	T25WDE	055235	T27BA	104647	T28BS0	111500	T39EAI	022136
T24PK3	045050	T25WDR	055140	T27BFR	102652	T28BS1	111501	T39PAC	021420
T24RB	045052	T25WNG	055525	T27BF2	102770	T28CNT	111526	T39PK2	022120
T24RES	047410	T25WNH	055700	T27B0T	103661	T28CNU	111530	T39PK3	022150
T24RN	045066	T26AM3	074606	T27BS0	102770	T28CON	111522	T39PK4	022160
T24RNC	046255	T26BA	075146	T27BS1	102771	T28DAT	111350	T39SIZ	022166
T24RT2	047502	T26BFR	073022	T27CNT	103006	T28DLY	111532	T39TAD	021430
T24RT3	047544	T26BF2	073140	T27CNU	103010	T28DSW	111360	T39WR	022162
T24RWN	046206	T26B0T	074175	T27CON	103002	T28DTA	113304	T4	035332 G
T24SSR	045617	T26BS0	073140	T27DAT	102640	T28DTR	113222	T4.1	035376
T24SZ	045056	T26BS1	073141	T27DLY	103012	T28IMV	111506	T4.10	042706
T24S2	045062	T26CNT	073156	T27DSW	102650	T28LO0	106200	T4.11	043152
T24S3	045064	T26CNU	073160	T27DTA	105646	T28LOQ	112114	T4.12	043424
T24TM	046133	T26DAT	073010	T27E0T	104031	T28OFL	112530	T4.13	043710
T24TRL	047254	T26DLY	073164	T27LON	105011	T28PAC	111340	T4.14	044210
T24VCK	046607	T26DSW	073020	T27L00	076146	T28PBP	111611	T4.2	036020
T24WB	045052	T26DTA	074242	T27L0P	105073	T28PK2	111450	T4.3	036600
T24WDC	046536	T26E0T	074330	T27LOQ	103455	T28PK3	111470	T4.4	037354
T24WDD	046450	T26LON	075310	T27LOR	103330	T28RB	111472	T4.5	040056
T24WDE	045651	T26L00	056570	T27NEF	105331	T28RDF	111674	T4.6	040522
T24WDF	045375	T26LOP	075372	T27OFL	104356	T28RDG	111755	T4.7	041156
T24WDG	045222	T26LOQ	074006	T27PAC	102630	T28RES	113426	T4.8	041612
T24WDR	045070	T26LOR	073661	T27PBP	105155	T28RIB	111534	T4.9	042204
T24WSS	046733	T26NEF	073254	T27PK2	102740	T28RN	111516	T5	047576 G
T25BFR	055012	T26NEQ	075630	T27PK3	102760	T28RRM	113007	T5.1	047626
T25BF2	055130	T26OFL	074655	T27RB	102762	T28RRN	113065	T5.2	050602
T25BNC	055610	T26PAC	073000	T27RDF	103102	T28RRP	113144	T5.3	051202
T25B0T	055315	T26PBP	075454	T27RES	105764	T28RT2	113520	T5.4	051656
T25BS0	055130	T26PK2	073110	T27RN	102776	T28RT3	113562	T5.5	052322
T25BS1	055131	T26PK3	073130	T27RNC	104234	T28RWN	112461	T5.6	053062
T25CNT	055150	T26RB	073132	T27RRF	103151	T28SSR	112175	T5.7	054022
T25CN2	055146	T26RDF	073336	T27RT2	106056	T28SZ	111476	T5.8	054342
T25CON	055142	T26RES	075720	T27RT3	106120	T28S2	111502	T6	056524 G
T25DAT	055000	T26RN	073146	T27RWN	104165	T28S3	111504	T6.1	056570
T25DLY	055152	T26RNC	074533	T27SC	103246	T28TM	112404	T6.10	066210
T25DSW	055010	T26RRF	073405	T27SCF	105427	T28THK	112735	T6.11	067064
T25L00	047626	T26RRG	073502	T27SSR	103536	T28VCK	112662	T6.12	067734
T25NEF	055763	T26RSZ	073162	T27SZ	102766	T28WB	111472	T6.13	070666
T25NET	055451	T26RT2	076012	T27S2	102772	T28WDC	112603	T6.14	071716
T25OFL	056174	T26RT3	076054	T27S3	102774	T28WDE	112246	T6.15	072276
T25PAC	054770	T26RWN	074464	T27TIM	103754	T28WDF	112037	T6.2	057502
T25PK2	055100	T26SC	073577	T27TH	104110	T28WDR	111520	T6.3	060350
T25PK3	055120	T26SSR	074067	T27TRL	105243	T3	030152 G	T6.4	061242
T25RB	055122	T26SZ	073136	T27TSA	105504	T3BFLG	003140 G	T6.5	062170
T25RES	056336	T26S2	073142	T27VCK	104574	T3.1	030222	T6.6	062746
T25RIB	056043	T26S3	073144	T27WB	102762	T3.2	030574	T6.7	063610
T25RN	055136	T26TM	074407	T27WDC	104521	T3.3	031444	T6.8	064462
T25RT2	056430	T26TRL	075542	T27WDD	104431	T3.4	032264	T6.9	065334
T25RT3	056472	T26VCK	075073	T27WDE	103572	T3.5	032500	T7	076106 G
T25RWN	056125	T26WB	073132	T27WDF	103400	T3.6	033046	T7.1	076146

Symbol table

T7.2	076544	WC.IOT=	000100	XORFOR	010102	XSOTMK=	100000	X2.RCE=	040000
T7.3	077326	WC.IIT=	000040	XST0	= 000006	XSQVCK=	000020	X2.REV=	000077
T7.4	100150	WC.ISR=	000020	XST1	= 000010	XSOWLE=	004000	X2.SPA=	035400
T7.5	101052	WF.IED=	000010	XST2	= 000012	XSOWLK=	000004	X2.UNI=	000007
T8	106144	WF.IER=	000004	XST3	= 000014	XXCOMM	003112	X2.WCF=	002000
T8.1	106200	WF.IHI=	000200	XST4	= 000016	X#ALWA=	000000	X3.DCK=	000010
T8.2	106560	WF.IRE=	000040	XS0BOT=	000002	X#FALS=	000040	X3.MBZ=	000006
T8.3	107040	WF.IWF=	000020	XS0EOT=	000001	X#OFFS=	000400	X3.MDE=	177400
UAM	= 000200	WF.IWR=	000100	XS0IE	= 000040	X#TRUE=	000020	X3.OPI=	000100
UNITN	002172	WF.I3R=	000002	XS0ILA=	000400	X1.COR=	020000	X3.REV=	000040
UNREC	= 000006	WF.I4R=	000001	XS0ILC=	001000	X1.DLT=	100000	X3.RIB=	000001
USI	004121	WRTCHR	010752	XS0LET=	020000	X1.MBZ=	017375	X3.SPA=	000200
WAITF	016340	WRTERR	005111	XS0MOT=	000200	X1.RBP=	000400	X3.TRF=	000020
WC.IFA=	000200	WRTMSG	005054	XSONEF=	002000	X1.SPA=	040000	X4.HSP=	100000
WC.IFE=	000002	WSMBK	022200	XS0ONL=	000100	X1.UNC=	000002	X4.MBZ=	017400
WC.IG0=	000001	XFERAS	016030	XS0PED=	000010	X2.BUF=	000100	X4.RCE=	040000
WC.IRE=	000010	XNXM	016466	XS0RLL=	010000	X2.EXT=	000200	X4.TSM=	020000
WC.IRW=	000004	XORBFO	007764	XS0RLS=	040000	X2.OPM=	100000	X4.WRC=	000377

. ABS. 114404 000 (RW,I,GBL,ABS,OVR)  
 000000 001 (RW,I,LCL,REL,CON)  
 ABS 000000 002 (RW,I,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 322  
 Work file writes: 308  
 Size of work file: 28912 Words ( 113 Pages)  
 Size of core pool: 19714 Words ( 75 Pages)  
 Operating system: RSX-11M/PLUS (Under VAX/VMS)

Elapsed time: 00:14:25.40  
 CVTSCC.CVTSCC.=SVC/ML.CVTSCC