

Micro Fiche Scan

Name of device(s) tested:

TU80

Test description:

TU80 FRONT END PRTB

MAINDEC Number or Package Identifier (after SEP 1977):

CZTUXA0

Fiche Document Part Number:

AH-T333A-MC

Fiche preparation date unknown, using copyright year:

1983

Image resolution:

1-bit black&white, compressed for minimal file size

COPYRIGHT (C) 1983 by d|i|g|i|t|a|l

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 2
USER DOCUMENTATION

.REMI

IDENTIFICATION

PRODUCT ID: AC-T332A-MC
PRODUCT TITLE: CZTUXAO TUBO FRONT END PRT B
PRODUCT DATE: 23 - MARCH - 1983
MAINTAINER: TAPE DIAGNOSTIC ENGINEERING
AUTHOR: DICE SYSTEMS, INC.

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

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

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

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

TABLE OF CONTENTS

ABSTRACT

CHAPTER 1 - REQUIREMENTS

- 1.1 EQUIPMENT
- 1.2 MEMORY STORAGE
- 1.3 PRELIMINARY PROGRAMS

CHAPTER 2 - LOADING AND STARTING PROCEDURE

- 2.1 ACT11 OPERATION

CHAPTER 3 - SWITCH SETTINGS

CHAPTER 4 - ERRORS

- 4.1 ERROR TYPEOUT FORMAT (HARDWARE)
- 4.2 ERROR TYPEOUT FORMAT (FUNCTION OUT OF RANGE)

CHAPTER 5 - SUBROUTINE ABSTRACTS

CHAPTER 6 - MISCELLANIOUS

- 6.1 STACK POINTER
- 6.2 EXECUTION TIME

CHAPTER 7 - PROGRAM DESCRIPTION

- 7.1 FUNCTION TIME DOCUMENT
- 7.2 TEST SEQUENCE / RELATED ADJUSTMENTS / ASSOCIATED HARDWARE
- 7.3 SUBTEST DESCRIPTIONS

ABSTRACT

1.0 ABSTRACT

THIS IS A PDP-11 RESIDENT DIAGNOSTIC WHICH CHECKS THE FUNCTIONALITY OF A TUBO MAGTAPE SUBSYSTEM WHILE CONNECTED TO A PDP-11 SYSTEM. THE PROGRAM PROVIDES ERROR MESSAGES WHICH IDENTIFY FAILING FUNCTIONS THAT AID IN THE REPAIR OF THE DEVICE. REFERENCE THE FOLLOWING DIGITAL EQUIPMENT DOCUMENTS:

1. ENGINEERING SPECIFICATION FOR TUBO MAGTAPE CONTROLLER; DOCUMENT NUMBER: YM-C194D-022; REVISION NUMBER 2; DATE: 28-JUL-81.
2. ENGINEERING SPECIFICATION FOR TUBO DIAGNOSTIC PACKAGE; DOCUMENT NUMBER: YM-C194F-00; REVISION NUMBER 0; DATE: 2-SEP-81.
3. ENGINEERING SPECIFICATION FOR TUBO MAGTAPE SUBSYSTEM; DOCUMENT NUMBER: YM-C194S-02; REVISION NUMBER 3; DATE: 10-JUN-81.
4. CIQPMAD XXDP+ PROGRAMMER'S MANUAL; DOCUMENT NUMBER AC-S296A-AC DATE: 14 JULY 1980.

HARDWARE, SOFTWARE REQUIREMENTS AND PREREQUISITES

2.0 HARDWARE, SOFTWARE REQUIREMENTS AND PREREQUISITES

2.1 HARDWARE REQUIREMENTS

PDP-11 FAMILY PROCESSOR WITH 32K WORDS OF MEMORY
TUBO MAGTAPE SUBSYSTEM (DRIVE AND CONTROLLER)
CAUTION:DIAGNOSTIC REQUIRES 32K WORDS OF MEMORY
(28K USEABLE I.E. 4K FOR I/O PAGE)

2.2 OPTIONAL HARDWARE:

UP TO 4 TUBO CONTROLLERS PER PDP-11 UP TO 1 DRIVES PER CONTROLLER

2.3 SOFTWARE REQUIREMENTS

PDP-11 DIAGNOSTIC SUPERVISOR (HSAADO.SYS)
PDP-11 DIAGNOSTIC LOADER/MONITOR (XXDP+)

2.4 PREREQUISITES

FUNCTIONAL PDP-11 FAMILY CENTRAL PROCESSOR AND MEMORY
FUNCTIONAL CONSOLE TERMINAL
FUNCTIONAL STANDALONE DIAGNOSTIC SUPERVISOR

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 6
 USER DOCUMENTATION

OPERATING INSTRUCTIONS - OPERATOR COMMANDS

3.0 OPERATING INSTRUCTIONS

3.1 OPERATOR COMMANDS

THE TUBO DIAGNOSTIC IS A PDP-11 DIAGNOSTIC SUPERVISOR COMPATIBLE PROGRAM. ALL LOADING AND RUN TIME INSTRUCTIONS CAN BE REFERENCED IN THE PDP-11 PROGRAMMER'S MANUAL 'CIQPMAO XXDP+ PROGRAMMER'S MANUAL, NUMBER AC-S296A-AC.

BOOT THE DIAGNOSTIC XXDP MEDIA

CHMDLBO XXDP+ DL MONITOR 28K
 BOOTED VIA UNIT 0

ENTER DATE (DD-MMM-YY): 29-JAN-82
 RESTART ADDRESS: 153726
 50 HZ ? N <CR>
 LSI ? N Y
 THIS IS XXDP+ TYPE 'H' OR 'H/L' FOR DETAILS

R CZTUXAO

CZTUXABINDRS LOADED
 DIAG. RUN-TIME SERVICES REV. D APR 79
 CZTUX-A-0
 ****TUBO LOGIC DIAGNOSTIC****
 UNIT IS TUBO
 DR>
 DRS>START/FLAG:PNT:HOE

THE ABOVE COMMANDS WILL START THE DIAGNOSTIC. THE COMMAND HAS TWO SWITCHES ON WHICH ARE 'PRINT EACH TEST NBR AS EXECUTED' AND 'HALT ON ERROR'.

3.2 HARDWARE PARAMETERS

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 NOT RUN. IT WILL GIVE THE MESSAGE 'NO UNIT'. A 'Y' IS REQUIRED AND AT LEAST A '1' IS REQUIRED AT THE "# UNITS (D)?" QUESTION.

TSBA/TSDB = 172522, 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

OPERATING INSTRUCTIONS

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 M7454 CONTROLLERS
PRESENT TO BE TESTED>

UNIT 0

DEVICE ADDRESS (O) 172522 ? <ENTER THE ADDRESS OF THE
TSSR 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.

3.3 SOFTWARE PARAMETERS

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.>

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 8
USER DOCUMENTATION

OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS

4.0 OPERATING INSTRUCTIONS - SAMPLE PRINTOUTS

4.1 SUCCESSFUL RUN EXAMPLE (PDP-11)

TST: 001 FIFO EXERCISER TEST
TST: 002 INITIALIZE #4 TEST
TST: 003 OFF-LINE REJECT AND REWIND TEST
TST: 004 BASIC WRITE DATA TEST
TST: 005 BASIC READ DATA (FORWARD AND REVERSE) TEST
TST: 006 MANUAL INTERVENTION TEST
TST: 007 CONFIGURATION TYPEOUT TEST
TST: 008 SCOPE LOOPS TEST

0 ERRORS

NOTE: PROGRAM NOW STARTS OVER AGAIN AT TEST 1

OPERATING INSTRUCTIONS - SAMPLE ERROR MESSAGES

5.0 OPERATING INSTRUCTIONS - SAMPLE ERROR MESSAGES

ERROR MESSAGE EXAMPLE 1

TST: 001 FIFO EXERCISER TEST
CZTUX 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

CZTUX 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

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

PROGRAM RUN TIMES

6.0 PROGRAM RUN TIMES

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

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

TEST NUMBER	N/I SECS.	ITER SECS	DEF SECS.
1	1	1	0
2	1	1	0
3	1	1	0
4	1	1	0
5	1	1	0
6	N/A		
7	N/A		
8	N/A		

THE TIMES REQUIRED TO RUN TESTS 1 THROUGH 5 IN ONE COMMAND:

Q.V.	2 MINS 19 SECONDS
DEFAULT	11 MINS 35 SECONDS

7.0 TEST SUMMARIES

7.1 TEST 1 - FIFO EXERCISER

 * NOTE: IF THIS TEST DETECTS AN ERROR REPLACE THE TUBO'S *
 * CL TOLLER (M7454) *

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

7.1.1 TEST 1, SUBTEST 1:-

THIS TEST VERIFIES, BY USING THE READ STATUS SELECT CODE, THAT
 THE FIFO STATUS IS IN THE CORRECT INITIAL STATE AFTER THE
 CONTROLLER IS INITIALIZED (INPUT READY TRUE, OUTPUT READY
 AND DATA IN MISS FALSE). THESE STATUS SIGNALS ARE CHECKED
 BY THE CONTROLLER'S SELF-TEST SEQUENCE, SO THIS SUBTEST IS
 ACTUALLY MORE OF A PARTIAL CHECK OF THE READ STATUS FUNCTION
 THAN THE FIFO STATUS.

7.1.2 TEST 1, SUBTEST 2:-

THIS SUBTEST VERIFIES THE ABILITY OF THE FIFO TO CORRECTLY
 PASS A DATA BYTE FROM INPUT TO OUTPUT. FOR EACH OF THE 256
 DATA VALUES (0-377 OCTAL) THE FOLLOWING IS DONE:

1. INITIAL FIFO STATUS IS CHECKED
2. THE WRITE FIFO FUNCTION, SPECIFYING A COUNT OF ONE
 BYTE TO BE WRITTEN, IS EXECUTED.
3. READ STATUS IS EXECUTED AND FIFO STATUS IS CHECKED.
4. READ FIFO IS EXECUTED AND THE DATA AND FINAL STATUS IS
 CHECKED.

7.1.3 TEST 1, SUBTEST 3:-

THIS SUBTEST VERIFIES THE ABILITY OF THE FIFO TO CORRECTLY
 PASS MULTIPLE DATA BYTES FROM INPUT TO OUTPUT. THE FOLLOWING
 SEQUENCE IS DONE WITH VARIOUS DATA PATTERNS AND BYTE COUNTS
 FROM 2 TO 64.

1. INITIAL FIFO STATUS IS CHECKED.
2. THE WRITE FIFO FUNCTION.
3. READ STATUS IS EXECUTED AND FIFO STATUS IS CHECKED.
4. READ FIFO IS EXECUTED AND THE DATA AND FINAL STATUS
 IS CHECKED.

7.1.4 TEST 1, SUBTEST 4:-

THIS SUBTEST VERIFIES THAT READING THE FIFO WHEN IT IS EMPTY
 CAUSES THE LAST WORD (ILW) STATUS TO ASSERT.

7.1.5 TEST 1. SUBTEST 5:-

THIS SUBTEST VERIFIES THAT WRITING 64. BYTES INTO FIFO, WITHOUT READING ANY OUT, CAUSES THE INPUT READY STATUS TO NEGATE. THE SUBTEST THEN VERIFIES THAT WRITING A 65TH BYTE INTO FIFO CAUSES THE DATA IN MISS STATUS TO ASSERT. NEXT IT IS VERIFIED THAT THE ORIGINAL 64 BYTES CAN BE READ OUT CORRECTLY AND THAT THE DATA HAS NOT BEEN CORRUPTED.

7.1.6 TEST 1 SUBTEST 6:-

THIS SUBTEST VERIFIES THAT THE RESET FIFO FUNCTION WITHIN THE WRITE MISCELLANIOUS CONTROL 1 FUNCTION INITIALIZES THE FIFO TO THE CORRECT INITIAL STATUS. THE FOLLOWING STEPS ARE PERFORMED:

1. RESET AN ALREADY INITIALIZED FIFO AND CHECK FOR PROPER STATUS.
2. WRITE A VARYING NUMBER OF BYTES (1-65.) INTO THE FIFO AND VERIFY THAT AFTER EACH BLOCK OF BYTES IS WRITTEN THE FIFO CAN BE RESET TO ITS INITIAL STATE.

7.2 TEST 2 - INITIALIZE #4

 * NOTE: IF THIS TEST DETECTS AN ERROR REPLACE THE TUBO'S *
 * CONTROLLER (M7454) *

THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITONS.

7.3 TEST 3 - OFF LINE REJECT AND REWIND TEST

THIS TEST VERIFIES BASIC TAPE MOTION COMMAND DECODING AND BASIC OPERATION OF THE REWIND POSITIONING COMMAND. IT DOES NOT NECESSARILY DEMONSTRATE THAT THE TRANSPORT AN 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 IN THE NORMAL COURSE OF THEIR TEST SEQUENCES. THE TEST CONSISTS OF THE FOLLOWING THREE SUBTESTS:

7.3.1 TEST 3. SUBTEST 1:-

THIS SUBTEST VERIFIES THAT ALL TAPE MOTION COMMANDS (WITH

VALID MODE CODES) TERMINATE WITH FUNCTION REJECT AND OFF LINE STATUS WHEN THE TAPE TRANSPORT IS OFF-LINE.

7.3.2 TEST 3, SUBTEST 2:-

THIS SUBTEST VERIFIES THAT A REWIND COMMAND, WITH THE CLEAR VOLUME CHECK (CVC) BIT CLEAR IS REJECTED IF THE VOLUME CHECK (VCK) FLAG IS SET.

7.3.3 TEST 3, SUBTEST 3:-

THIS SUBTEST VERIFIES THAT A REWIND COMMAND WITH A CVC=1 CLEARS VCK AND RETURNS PROPER STATUS IN THE MESSAGE BUFFER.

7.4 TEST 4 - BASIC WRITE DATA TEST

* NOTE: THIS TEST MUST HAVE A GOOD MAGTAPE IN THE DRIVE ANY *
* TAPE ERRORS WILL BE DISPLAYED AS TAPE STATUS ALERT *

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

7.4.1 TEST 4, SUBTEST 1:-

THIS SUBTEST VERIFIES THAT A WRITE COMMAND (ANY VALID MODE CODE) WITH THE CLEAR VOLUME CHECK (CVC) BIT CLEAR IS REJECTED IF THE VOLUME CHECK (VCK) FLAG IS SET. ALL VALID MODE CODES ARE CHECKED (WRITE DATA, WRITE RETRY).

7.4.2 TEST 4, SUBTEST 2:-

THIS SUBTEST VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT CLEAR OPERATES PROPERLY. THE BYTE COUNT (RECORD SIZE) VARIES FROM 20 THROUGH 64K IN VARYING INCREMENTS (DEPENDING ON WHETHER OR NOT THE DIAGNOSTIC IS RUNNING ON THE LONG VERIFICATION MODE). THE TAPE IS NOT REWOUND BETWEEN SUCCESSIVE RECORDS BUT IS REWOUND AFTER THE FINAL RECORD IS WRITTEN. AN INCREMENTING COUNT PATTERN IS SUPPLIED IN THE DATA BUFFER. AFTER EACH BLOCK IS WRITTEN, THE TTSR AND THE TSBA REGISTERS AND THE MESSAGE BUFFERS ARE CHECKED.

? 4.3 TEST 4, SUBTEST 3:-

THIS SUBTEST VERIFIES THAT WRITE DATA COMMANDS WITH CVC=1 AND

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 11-3
 USER DOCUMENTATION

THE SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2. THE RESULTS SHOULD BE THE SAME.

7.4.4 TEST 4, SUBTEST 4:-

THIS SUBTEST VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE.

7.4.5 TEST 4, SUBTEST 5:-

THIS SUBTEST VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS WITHOUT MOVING TAPE. THIS TEST IS SKIPPED IF NONEXISTANT MEMORY CAN NOT BE ADDRESSED.

7.4.6 TEST 4, SUBTEST 6:-

THIS SUBTEST VERIFIES THAT A WRITE DATA COMMAND SPECIFYING A DATA BUFFER IN STARTING IN EXISTANT MEMORY BUT RUNNING INTO NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS. A LARGE ENOUGH RECORD SIZE IS SPECIFIED SUCH THAT TAPE IS ACTUALLY MOVED AND WRITTEN.

7.5 TEST 5 - BASIC READ DATA TEST (FORWARD AND REVERSE)

 * NOTE: THIS TAPE MUST HAVE A GOOD MAGTAPE IN THE DRIVE *
 * ANY TAPE ERRORS WILL BE DISPLAYED AS TAPE STATUS ALERT *

THIS TEST VERIFIES THAT THE READ FORWARD AND READ REVERSE COMMANDS OPERATE PROPERLY. VARIOUS COMBINATIONS OF ODD AND EVEN DATA BUFFER BOUNDARIES, RECORD SIZES AND BYTE SWAP CONTROL VARIABLES ARE USED. THE TEST 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, AND NONEXISTANT DATA BUFFER ADDRESSES.

7.5.1 TEST 5, SUBTEST 1:-

THIS SUBTEST VERIFIES THAT THE READ FORWARD COMMAND WITH SWB=0 OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF TEST RECORDS VARYING IN LENGTH AND DATA

CONTENT. THE TAPE IS THEN REWOUND AGAIN AND THE RECORD READ SEQUENTIALLY AND RESULTS (STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON EACH READ FORWARD COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

7.5.2 TEST 5, SUBTEST 2:-

THIS SUBTEST VERIFIES THAT THE RAD DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2. THE RESULTS, EXCEPT FOR RAM CONTENTS SHOULD BE THE SAME.

7.5.3 TEST 5, SUBTEST 3:-

THIS SUBTEST VERIFIES THAT A READ FORWARD COMMAND READING A RECORD LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH (RLL) BIT SET.

7.5.4 TEST 5, SUBTEST 4:-

THIS SUBTEST VERIFIES THAT A READ FORWARD COMMAND READING A RECORD SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT (RLS) BIT SET. IT IS VERIFIED THAT THE RESIDUAL BYTE COUNT (RBPCR) IN THE MESSAGE BUFFER CONTAINS THE PROPER NONZERO VALUE (E.G. THE DIFFERENCE BETWEEN THE ACTUAL BYTE COUNT AND THE ACTUAL RECORD LENGTH).

7.5.5 TEST 5, SUBTEST 5:-

OPERATES PROPERLY. THE TAPE IS FIRST REWOUND AND THEN WRITTEN WITH A SERIES OF TEST RECORDS VARYING IN LENGTH AND DATA CONTENT. THE TAPE IS THEN READ IN REVERSE SEQUENTIALLY AND THE RESULTS (STATUS, DATA, ETC.) VERIFIED. THE BYTE COUNT ON EACH READ REVERSE COMMAND IS SET TO THE LENGTH OF THE EXPECTED RECORD, SO NO EXCEPTIONAL CONDITIONS SHOULD OCCUR.

7.5.6 TEST 5, SUBTEST 6:-

THIS SUBTEST VERIFIES THAT THE READ DATA COMMANDS WITH CVC=1 AND THE SWAP BYTES (SWB) BIT SET OPERATES PROPERLY. THE TEST SEQUENCE IS IDENTICAL TO THAT USED IN SUBTEST 2. THE RESULTS EXCEPT FOR RAM CONTENTS SHOULD BE THE SAME.

7.5.7 TEST 5, SUBTEST 7:-

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND, READING A RECORD LONGER THAN THE SPECIFIED BYTE COUNT, CAUSES A TAPE STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG (RLL) BIT SET.

7.5.8 TEST 5, SUBTEST 8:-

THIS SUBTEST VERIFIES THAT A READ REVERS COMMAND SPECIFYING A DATA BUFFER STARTING IN NONEXISTANT MEMORY TERMINATES WITH THE PROPER ERROR STATUS WITHOUT MOVING THE TAPE.

7.5.9 TEST 5, SUBTEST 9:-

THIS SUBTEST VERIFIES THAT ILLEGAL BUFFER ADDRESSES CAUSE A FUNCTION REJECT TERMINATION WITH ILLEGAL ADDRESS (ILA) ERROR BIT SET.

7.5.10 TEST 5, SUBTEST 10:-

THIS SUBTEST VERIFIES THAT A DATA BUFFER ADDRESS, REFERENCING NONEXISTANT MEMORY, CAUSES RECOVERABLE ERROR TERMINATION (TC=4), WITH THE NXM BIT SET IN THE TSSR, AND THAT THE TAPE IS ULTIMATELY POSITIONED PROPERLY.

7.5.11 TEST 5, SUBTEST 11:-

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE TAPE IS AT BOT RESULTS IN A FUNCTION REJECT TERMINATION WITH THE NONEXECUTABLE FUNCTION (NEF) ERROR BIT SET.

7.5.12 TEST 5, SUBTEST 12:-

THIS SUBTEST VERIFIES THAT A READ REVERSE COMMAND ISSUED WHILE THE TAPE IS POSITIONED BEFORE THE FIRST RECORD ON TAPE (BUT NOT AT BOT) RESULTS IN TAPE STATUS ALERT.

7.6 TEST 6 - MANUAL INTERVENTION

THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A 'TEST') THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF THE SUBSYSTEM THAT CAN NOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE. THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR ARE BY TYPING <CTRL-C> OR BY SELECTING CODE 3. SELECTION CODES AND SUBROUTINES ARE:

CODE	ROUTINE
0	HELP. PRINTS THIS MENU
1	OFFLINE/ONLINE ATTENTION TEST
2	WRITE-PROTECT TEST

3 EXIT (RETURN TO SUPERVISOR)

EACH MENU ITEM CORRESPONDS TO A SUBTEST AS FOLLOWS:

PRINTS OUT THE MENU ON THE CONSOLE TERMINAL.

THIS ROUTINE INITIALIZES THE CONTROLLER, ISSUES A WRITE CHARACTERISTICS COMMAND TO ENABLE ATTENTION INTERRUPTS, ISSUES A MESSAGE BUFFER RELEASE COMMAND, PRINTS A MESSAGE ON THE CONSOLE TERMINAL INSTRUCTING THE OPERATOR TO TOGGLE THE ON-LINE SWITCH ON THE TRANSPORT, THEN WAITS FOR AN ATTENTION INTERRUPT. EACH TIME THE TRANSPORT TRANSITIONS FROM ON-LINE TO OFF-LINE OR VICE VERSA, AN ATTENTION INTERRUPT SHOULD BE GENERATED. THE PROGRAM WILL REPORT THE INTERRUPT AND THE CURRENT STATE OF THE TRANSPORT. THE OPERATOR SHOULD VERIFY THAT THE REPORTED STATE MATCHES THE STATE INDICATED BY THE LED ON THE FRONT PANEL OF THE TRANSPORT. IN ADDITION, WHEN THE TRANSPORT IS PLACED OFF-LINE, THE PROGRAM ISSUES A SEQUENCE OF TAPE MOTION COMMANDS (READ, WRITE, POSITION, ETC.) AND VERIFIES THAT, FOR EACH COMMAND, FUNCTION REJECT TERMINATION RESULTS, ALONG WITH THE NON-EXECUTABLE FUNCTION (NEF) ERROR BIT BEING SET.

THIS ROUTINE INSTRUCTS THE OPERATOR TO MOUNT A SCRATCH TAPE REEL THAT DOES NOT HAVE A WRITE ENABLE RING INSTALLED, THEN WAITS FOR THE OPERATOR TO RESPOND THAT THIS HAS BEEN ACCOMPLISHED. UPON THE RESPONSE, THE PROGRAM VERIFIES THAT THE TRANSPORT SHOWS A WRITE-PROTECTED STATUS, THEN ATTEMPTS TO WRITE DATA ON THE TAPE AND EXPECTS THE APPROPRIATE ERROR TERMINATION INDICATING THAT THE WRITE FUNCTION COULD NOT BE PERFORMED BECAUSE THE REEL IS WRITE PROTECTED. IF THE APPROPRIATE TERMINATION IS NOT RECEIVED, AN ERROR IS REPORTED.

7.8 TEST 8 - CONFIGURATION TYPEOUT

THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL THE CONFIGURATION OF THE M7454 MODULE AND THE TUBO SUBSYSTEM. SPECIFICALLY, THE FOLLOWING INFORMATION IS PRINTED:

1. MICROCODE REVISION LEVEL OF THE M7454.
2. NUMBER OF TAPE TRANSPORTS CONNECTED TO THE CONTROLLER.
3. UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED) OF EACH CONNECTED TRANSPORT.

THE OPERATOR IS EXPECTED TO READ THE PRINTOUT AND VERIFY THAT IT MATCHES THE ACTUAL CONFIGURATION AT HAND. IF, FOR EXAMPLE, THE PROGRAM INDICATES THAT IT "SEES" TWO TRANSPORTS CONNECTED WHEN IN FACT ONLY ONE IS PRESENT, THE OPERATOR MUST INTERPRET THIS AS AN ERROR AND ATTEMPT TO FIND THE CAUSE (BAD CABLE, FAULTY UNIT SELECT DECODING IN THE TRANSPORT, ETC.). [SINCE THE CONTROLLER CAN ONLY ACCESS UNIT 0 IF IT IS IN "STANDARD" MODE, THE PROGRAM WILL FORCE THE MODULE INTO EXTENDED MODE VIA THE WRITE SUBSYSTEM MEMORY COMMAND IN ORDER TO SCAN FOR CONNECTED TRANSPORTS.]

THIS ROUTINE, WHEN ITS ACTIONS ARE COMPLETED, WILL EXIT BACK TO

THE DIAGNOSTIC SUPERVISOR SO THAT IF ADDITIONAL UNITS (CONTROLLERS) ARE SELECTED (E.G. FROM THE INITIAL STARTUP DIALOGUE), THE ROUTINE WILL BE REENTERED SO THAT THEIR CONFIGURATIONS CAN BE PRINTED.

7.8 TEST 8 - SCOPE LOOPS

THIS IS A STANDALONE ROUTINE PROVIDING A NUMBER OF TIGHT "SCOPE LOOPS" USEFUL FOR DEBUGGING BASIC REGISTER ACCESS PROBLEMS WITH THE M7454 MODULE. THESE SCOPE LOOPS CAN BE USED WHEN THE NORMAL "LOOP ON ERROR" OR "LOOP ON TEST (SUBTEST)" FACILITIES DON'T SEEM TO ALLOW THE OPERATOR TO ZERO IN A PROBLEM IN THE EARLY TESTS (I.E. THE HARDWARE MAY NOT BE RESPONDING TO A REGISTER ACCESS, CAUSING A BUS ERROR TRAP, EVEN THOUGH THE DEVICE ADDRESS SELECTED BY THE PROGRAM MATCHES THE THE CONFIGURATION SET UP IN THE HARDWARE DIP SWITCHES). THE FOLLOWING MENU OF SCOPE LOOPS IS AVAILABLE:

CODE	SCOPE LOOP
0	HELP. PRINT THIS MENU
1	TSBA READ ACCESS
2	TSSR READ ACCESS
3	INITIALIZE (TSSR WRITE ACCESS)
4	TSDB HIGH BYTE WRITE ACCESS
5	TSDB LOW BYTE WRITE ACCESS
6	TSDB MAINTENANCE-MODE WORD WRITE ACCESS
7	EXIT (RETURN TO SUPERVISOR)

FOR SCOPE LOOPS THAT WRITE INTO REGISTERS, THE PROGRAM PROMPTS THE OPERATOR FOR THE DATA TO BE WRITTEN. TYPING <RETURN> CAUSES AN EXIT FROM THE SCOPE LOOP BACK TO THE MENU.

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 12
PROGRAM HEADER

```

836          .SBTTL PROGRAM HEADER
837
843          .MCALL SVC
844 000000   SVC          ; INITIALIZE SUPERVISOR MACROS
845          .ENABLE LC
846          .NLIST BEX,CND
852 000000   .ENABL AMA,ABS
853          . = 2000
854 002000   BGNMOD TUV2A
      002000   TUV2A::

855
856          :++
857          : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
858          : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
859          :--
860
861
862 002000   POINTER BGNSW,BGNSFT,BGNAU,BGNDU,BGNRPT,BGNSETUP
863 002000   HEADER CZTUX,A,0,655.,0
      002000   LSNAME::          ;DIAGNOSTIC NAME
      002000   103          .ASCII /C/
      002001   132          .ASCII /Z/
      002002   124          .ASCII /T/
      002003   125          .ASCII /U/
      002004   130          .ASCII /X/
      002005   000          .BYTE 0
      002006   000          .BYTE 0
      002007   000          .BYTE 0
      002010   LSREV::          ;REVISION LEVEL
      002010   101          .ASCII /A/
      002011   LSDEPO::          ;0
      002011   060          .ASCII /O/
      002012   LSUNIT::          ;NUMBER OF UNITS
      002012   000001   .WORD TSPTHV
      002014   LSTIML::          ;LONGEST TEST TIME
      002014   001217   .WORD 655.
      002016   LSHPCP::          ;POINTER TO H.W. QUES.
      002016   066646   .WORD LSHARD
      002020   LSSPCP::          ;POINTER TO S.W. QUES.
      002020   067006   .WORD LSSOFT
      002022   LSHPTP::          ;PTR. TO DEF. H.W. PTABLE
      002022   002124   .WORD LSHW
      002024   LSSPTP::          ;PTR. TO S.W. PTABLE
      002024   002134   .WORD LSSW
      002026   LSLADP::          ;DIAG. END ADDRESS
      002026   067220   .WORD LSLAST
      002030   LSSTA::          ;RESERVED FOR APT STATS
      002030   000000   .WORD 0
      002032   LSCO::          .WORD 0
      002032   000000   .WORD 0
      002034   LSDTYP::          ;DIAGNOSTIC TYPE
      002034   000000   .WORD 0
      002036   LSAPT::          ;APT EXPANSION
      002036   000000   .WORD 0
      002040   LSDTP::          ;PTR. TO DISPATCH TABLE
      002040   067174   .WORD LSDISPATCH
      002042   LSPRIO::          ;DIAGNOSTIC RUN PRIORITY

```

CZTUXAO TUBO FRONT END PRT B
PROGRAM HEADER

MACRO M1200 29-MAR-83 13:32 PAGE 12-1

002042	000000	LSENVI::	.WORD	0	
002044					;FLAGS DESCRIBE HOW IT WAS SETUP
002044	000000	LSEXP1::	.WORD	0	
002046					;EXPANSION WORD
002046	000000	LSMREV::	.WORD	0	
002050					;SVC REV AND EDIT #
002050	003		.BYTE	CSREVISION	
002051	003		.BYTE	CSREDIT	
002052		LSEF::			;DIAG. EVENT FLAGS
002052	000000		.WORD	0	
002054	000000		.WORD	0	
002056		LSSPC::			
002056	000000		.WORD	0	
002060		LSDEVP::			; POINTER TO DEVICE TYPE LIST
002060	003334		.WORD	LSDVTYP	
002062		LSREPP::			;PTR. TO REPORT CODE
002062	023046		.WORD	LSRPT	
002064		LSEXP4::			
002064	000000		.WORD	0	
002066		LSEXP5::			
002066	000000		.WORD	0	
002070		LSAUT::			;PTR. TO ADD UNIT CODE
002070	022544		.WORD	LSAU	
002072		LSDUT::			;PTR. TO DROP UNIT CODE
002072	022642		.WORD	LSDU	
002074		LSLUN::			;LUN FOR EXERCISERS TO FILL
002074	000000		.WORD	0	
002076		LSDESP::			;POINTER TO DIAG. DESCRIPTION
002076	003342		.WORD	LSDESC	
002100		LSLOAD::			;GENERATE SPECIAL AUTOLOAD EMT
002100	104035		EMT	ESLOAD	
002102		LSETP::			;POINTER TO ERR_TBL
002102	000000		.WORD	0	
002104		LSICP::			;PTR. TO INIT CODE
002104	021746		.WORD	LSINIT	
002106		LSCCP::			;PTR. TO CLEAN-UP CODE
002106	023024		.WORD	LSCLEAN	
002110		LSACP::			;PTR. TO AUTO CODE
002110	022750		.WORD	LSAUTO	
002112		LSPRT::			;PTR. TO PROTECT TABLE
002112	021736		.WORD	LSPROT	
002114		LSTEST::			;TEST NUMBER
002114	000000		.WORD	0	
002116		LSDLY::			;DELAY COUNT
002116	000000		.WORD	0	
002120		LSHIME::			;PTR. TO HIGH MEM
002120	000000		.WORD	0	

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 13
DEFAULT HARDWARE P-TABLE

.SBTTL DEFAULT HARDWARE P-TABLE

865
866
867
868
869
870
871
872 002122
002122 000003
002124
002124
873
874 002124 172522
875 002126 000224
876 002130 000240
877 002132
002132

:++
: 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-LSHW/2

LSHW::
DFPTBL::

.WORD 172522 ; 2ND (OF 2) REGISTERS.
.WORD 224 ; INTERRUPT VECTOR
.WORD PRI05 ; INTERRUPT PRIORITY.
ENDHW

L10000:

.SBTTL SOFTWARE P-TABLE

879
 880
 881
 882
 883
 884
 885 002132
 002132 000004
 002134
 002134
 886
 887 002134 000000
 888 002136 000000
 889
 890
 891 002140 000031
 892 002142 000310
 893 002144
 002144
 894

```

:++
: THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
: PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
:--
      BGNSW  SFPTBL
      .WORD  L10001-L8SW/2
L8SW::
SFPTBL::

TRANSTST:: .WORD 0 ;ENABLE RAM DUMP
NOITS:: .WORD 0 ;INHIBIT ITERATION OPTION.
; ... 0 = ITERATE.
; ...NZ = INHIBIT ITERATE.
LERRMAX:: .WORD 25. ; LOCAL (PER TEST) ERROR LIMIT
GERRMAX:: .WORD 200. ; GLOBAL (PER UNIT) ERROR LIMIT
      ENDSW
L10001:

```

897
904
909
915
916
917
918
919
920
921
922
923
924
928 002144

.SBTTL GLOBAL EQUATES SECTION

.SBTTL GLOBAL EQUATES SECTION

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

EQUALS ; GET STANDARD EQUATES.

: BIT DIFINITIONS

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

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

: EVENT FLAG DEFINITIONS
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

000040	EF.START== 32.	: START COMMAND WAS ISSUED
000037	EF.RESTART== 31.	: RESTART COMMAND WAS ISSUED
000036	EF.CONTINUE== 30.	: CONTINUE COMMAND WAS ISSUED
000035	EF.NEW== 29.	: A NEW PASS HAS BEEN STARTED
000034	EF.PWR== 28.	: A POWER-FAIL/POWER-UP OCCURRED

: PRIORITY LEVEL DEFINITIONS

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 16-1
 GLOBAL EQUATES SECTION

```

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

```

```

: OPERATOR FLAG BITS
:

```

```

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

```

929
 930 002144

```

KT11 ; DEFINE MEMORY MANAGEMENT REGISTERS

```

```

.SBTTL 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

```

CZTUXAO TUBO FRONT END PRT 8
MEMORY MANAGEMENT DEFINITIONS

MACRO M1200 29-MAR-83 13:32 PAGE 16-2

```
UIPAR0= 177640
UIPAR1= 177642
UIPAR2= 177644
UIPAR3= 177646
UIPAR4= 177650
UIPAR5= 177652
UIPAR6= 177654
UIPAR7= 177656
  .IF NB
  ;*USER 'D' PAGE ADDRESS REGISTERS
  UDPAR0= 177660
  UDPAR1= 177662
  UDPAR2= 177664
  UDPAR3= 177666
  UDPAR4= 177670
  UDPAR5= 177672
  UDPAR6= 177674
  UDPAR7= 177676
  .ENDC
  .ENDC
  .IF NB
  ;*SUPERVISOR 'I' PAGE DESCRIPTOR REGISTERS
  SIPDR0= 172200
  SIPDR1= 172202
  SIPDR2= 172204
  SIPDR3= 172206
  SIPDR4= 172210
  SIPDR5= 172212
  SIPDR6= 172214
  SIPDR7= 172216
  .IF NB
  ;*SUPERVISOR 'D' PAGE DESCRIPTOR REGISTERS
  SDPDR0= 172220
  SDPDR1= 172222
  SDPDR2= 172224
  SDPDR3= 172226
  SDPDR4= 172230
  SDPDR5= 172232
  SDPDR6= 172234
  SDPDR7= 172236
  .ENDC
  ;*SUPERVISOR 'I' PAGE ADDRESS REGISTERS
  SIPAR0= 172240
  SIPAR1= 172242
  SIPAR2= 172244
  SIPAR3= 172246
  SIPAR4= 172250
  SIPAR5= 172252
  SIPAR6= 172254
  SIPAR7= 172256
  .IF NB
  ;*SUPERVISOR 'D' PAGE ADDRESS REGISTERS
  SDPAR0= 172260
  SDPAR1= 172262
  SDPAR2= 172264
  SDPAR3= 172266
  SDPAR4= 172270
```

CZTUXAO TUBO FRONT END PRT B
MEMORY MANAGEMENT DEFINITIONS

MACRO M1200 29-MAR-83 13:32 PAGE 16-3

```

SDPAR5= 172272
SDPAR6= 172274
SDPAR7= 172276
.ENDC
.ENDC
;*KERNEL 'I' PAGE DESCRIPTOR REGISTERS
172300 KIPDR0= 172300
172302 KIPDR1= 172302
172304 KIPDR2= 172304
172306 KIPDR3= 172306
172310 KIPDR4= 172310
172312 KIPDR5= 172312
172314 KIPDR6= 172314
172316 KIPDR7= 172316
.IF NB
;*KERNEL 'D' PAGE DESCRIPTOR REGISTERS
KDPDR0= 172320
KDPDR1= 172322
KDPDR2= 172324
KDPDR3= 172326
KDPDR4= 172330
KDPDR5= 172332
KDPDR6= 172334
KDPDR7= 172336
.ENDC
;*KERNEL 'I' PAGE ADDRESS REGISTERS
172340 KIPAR0= 172340
172342 KIPAR1= 172342
172344 KIPAR2= 172344
172346 KIPAR3= 172346
172350 KIPAR4= 172350
172352 KIPAR5= 172352
172354 KIPAR6= 172354
172356 KIPAR7= 172356
.IF NB
;*KERNEL 'D' PAGE ADDRESS REGISTERS
KDPAR0= 172360
KDPAR1= 172362
KDPAR2= 172364
KDPAR3= 172366
KDPAR4= 172370
KDPAR5= 172372
KDPAR6= 172374
KDPAR7= 172376
.ENDC

```

```

935                                     .SBTTL TUBO REGISTER AND PACKET DEFINITIONS
936
937                                     :
938                                     : SOME GENERAL EQUATES.
939                                     :
940
941         000004         ERRVEC==         4         : POINTER TO ERROR VECTOR FOR BUS TIME OUT.
942         000060         TTIVEC==        60         : INTERRUPT VECTOR FOR CONSOLE INPUT
943         177560         TTICSR==       177560         : BUS ADDRESS OF CONSOLE INPUT
944         177562         TTIBFR==       177562         : CONSOLE INPUT DATA BUFFER
945
946                                     :+
947                                     :BIT DEFINITIONS FOR TSSR REGISTER
948                                     :-
949
950         100000         SC=          BIT15         :SPECIAL CONDITION
951         040000         BIE=          BIT14         :BUS INTERFACE ERROR
952         020000         SCE=          BIT13         :SANITY CHECK ERROR
953         010000         RMR=          BIT12         :MODIFICATION REFUSED
954         004000         NXM=          BIT11         :NONEXISTANT MEMORY ERROR
955         002000         NBA=          BIT10         :NEED BUFFER ADDR. SS
956         001400         HIADDR= BIT9!BIT8         :EXTENDED ADDRESS BITS
957         000200         SSR=          BIT7          :SUB SYSTEM READY
958         000100         OFL=          BIT6          :OFF LINE BIT
959         000060         FATERR= BIT4!BIT5         :FATAL TERMINATION ERROR CODES
960         000016         TERCLS= BIT3!BIT2!BIT1     :TERMINATION CODES
961
962                                     :+
963                                     :BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0
964                                     : (XST0)
965                                     :
966                                     :-
967
968
969
970         100000         XSOTMK= BIT15         :TAPE MARK DETECTED
971         040000         XSORLS= BIT14         :RECORD LENGTH SHORT
972         020000         XSOLET= BIT13         :LOGICAL END OF TAPE
973         010000         XSORLL= BIT12         :RECORD LENGTH LONG
974         004000         XSOWLE= BIT11         :WRITE LOCK ERROR
975         002000         XSONEF= BIT10         :NON EXECUTABLE FUNCTION
976         001000         XSOILC= BIT9         :ILLEGAL COMMAND
977         000400         XSOILA= BIT8         :ILLEGAL ADDRESS
978         000200         XSOMOT= BIT7         :TAPE IN MOTION
979         000100         XSOONL= BIT6         :TRANSPORT ON LINE
980         000040         XSOIE= BIT5         :INTERRUPT ENABLE
981         000020         XSOVCK= BIT4         :VOLUME CHECK BIT
982         000010         XSOPED= BIT3         :PHASE ENCODED DRIVE
983         000004         XSOWLK= BIT2         :WRITE LOCKED
984         000002         XSOBOT= BIT1         :BEGINNING OF TAPE
985         000001         XS0EOT= BIT0         :END OF TAPE
986
987                                     :+
988                                     :BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 1
989                                     : (XST1)
990                                     :
991                                     :-
    
```

CZTUXAO TUBO FRCNT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 17-1
TUBO REGISTER AND PACKET DEFINITIONS

```

992      100000      X1.DLT = BIT15      ;DATA LATE
993      040000      X1.SPARE= BIT14      ;NOT USED
994      020000      X1.COR = BIT13      ;CORRECTABLE DATA ERROR
995      017375      X1.MBZ = BIT12+BIT11+BIT10+BIT9+BIT7+BIT6+BIT5+BIT4+BIT3+BIT2+BIT0 ;ALWAYS 0
996      000400      X1.RBP = BIT8      ;READ BUS PARITY ERROR
997      000002      X1.UNC = BIT1      ;UNCORRECTABLE DATA OR HARD ERROR
998
999      ;+
1000     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2
1001     ;(XST2)
1002     ;-
1003     100000      X2.OPM = BIT15      ;OPERATION IN PROGRESS (TAPE MOVING)
1004     040000      X2.RCE = BIT14      ;RAM CHECKSUM ERROR
1005     035400      X2.SP/RE= BIT13+BIT12+BIT11+BIT9+BIT8 ;NOT USED BY TUBO (ALWAYS=0)
1006     002000      X2.WCF = BIT10      ;WRITE CLOCK FAILURE (FIFO NOT EMPTIED BY TRANSPORT)
1007     000200      X2.EXTF = BIT7      ;IF WRITE CHAR CMD THEN = EXTENDED FEATURES ENABLED
1008     000100      X2.BUFE = BIT6      ;IF WRITE CHAR CMD THEN = BUFFERING ENABLED
1009     000077      X2.REV = 000077    ;IF WRITE CHAR CMD THEN = MICROCODE REVISION LEVEL
1010     000007      X2.UNIT = BIT2+BIT1+BIT0 ;IF GET STATUS THEN = CURRENTLY SELECTED UNIT NO.
1011
1012     ;+
1013     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3
1014     ;(XST3)
1015     ;-
1016     177400      X3.MDE = 177400    ;MICRO-DIAGNOSTIC ERROR CODE
1017     000200      X3.SPARE= BIT7      ;NOT USED BY TUBO
1018     000100      X3.OPI = BIT6      ;OPERATION INCOMPLETE
1019     000040      X3.REV = BIT5      ;REVERSE
1020     000020      X3.TRF = BIT4      ;TRANSPORT RESPONSE FAILURE
1021     000010      X3.DCK = BIT3      ;DENSITY CHECK
1022     000006      X3.MBZ =BIT2+BIT1    ;NOT USED ALWAYS 0
1023     000001      X3.RIB = BIT0      ;REVERSE INTO BOT
1024
1025     ;+
1026     ;BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4
1027     ;(XST4)
1028     ;-
1029     100000      X4.HSP = BIT15      ;HIGH SPEED
1030     040000      X4.RCE = BIT14      ;RETRY COUNT EXCEEDED
1031     020000      X4.TSM = BIT13      ;TRANSPORT SPECIAL MODE
1032     017400      X4.MBZ = BIT12+BIT11+BIT10+BIT9+BIT8 ;NOT USED ALWAYS 0
1033     000377      X4.WRC = 000377    ;WRITE RETRY COUNT FIELD
1034
1035     ;+
1036     ;TSSR TERMINATION CODES (BIT 0-2)
1037     ;-
1038
1039
1040
1041
1042     000006      TSREJ= 3*2          ;COMMAND REJECTED
1043     000006      UNREC= 6          ;UNRECOVERABLE ERROR
1044
1045     ;+
1046     ;DEVICE REGISTER OFFSETS
1047
1048

```


CZTUXAG TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 17-2
 TUBO REGISTER AND PACKET DEFINITIONS

```

1049      :-
1050
1051      177776      TSBA== -2
1052      177776      TSBAL== -2
1053      177776      TSDB== -2      ;TSDB/TSBA REGISTER
1054      177776      TSDBL== -2     ;TSDB/TSBA REGISTER
1055      177777      TSBAH== -1
1056      177777      TSDBH== -1     ;TSDB/TSBA REGISTER HIGH BYTE
1057      000000      TSSR== 0      ;TSSR REGISTER
1058      000001      TSSRH== 1     ;TSSR REGISTER HIGH BYTE
1059
1060      :-+
1061      : TSDB ADDRESS BIT DEFINITIONS
1062      :-
1063      000003      A1716 = BIT1+BIT0      ;ADDRESS BITS 17:16 ARE IN 1:0
1064
1065      :-+
1066      : COMMAND DEFINITIONS
1067      :-
1068      000017      P.GETSTAT      = 17      ;GET STATUS
1069      000013      P.INIT        = 13      ;INITIALIZE
1070      000012      P.CONTROL     = 12      ;CONTROL COMMANDS
1071      000011      P.FORMAT      = 11      ;FORMAT
1072      000010      P.POSITION    = 10      ;POSITION
1073      000006      P.WRTSUB      = 6       ;SUBSYSTEM WRITE
1074      000005      P.WRITE       = 5       ;WRITE
1075      000004      P.WRTCHAR     = 4       ;WRITE CHARACTERISTICS
1076      000001      P.READ        = 1       ;READ
1077
1078      :-+
1079      : COMMAND PACKET HEADER WORD BIT DEFINITIONS
1080      :-
1081      100000      P.ACK          = BIT15     ;BUFFER AVAIL FOR CONTROLLER
1082      040000      P.CVC          = BIT14     ;CLEAR VOLUME CHECK
1083      020000      P.OPP          = BIT13     ;REVERSE SEQUENCE OF DATA BITS
1084      010000      P.SWB          = BIT12     ;SWAP BYTES IN MEMORY
1085      007400      P.MODE         = BIT11!BIT10!BIT9!BIT8 ;EXTENDED COMMAND MODE FIELD
1086      000200      P.IE           = BIT7      ;INTERRUPT ENABLE
1087      000140      P.FMT= BIT6!BIT5 ;PACKET HEADER TYPE (ALWAYS=0)
1088      000037      P.CMD          = 37      ;MAJOR COMMAND FIELD
1089
1090      :-+
1091      : CONTROL COMMAND MODE CODES
1092      :-
1093      000000      PC.RELEASE     = 0*256.   ;RELEASE BUFFER
1094      000400      PC.REWIND     = 1*256.   ;REWIND
1095      001000      PC.NOOP       = 2*256.   ;NO-OP
1096      002000      PC.IEREW     = 4*256.   ;REWIND IMMEDIATE INTERRUPT
1097      002400      PC.ERASE      = 5*256.   ;SECURITY ERASE
1098
1099      :-+
1100      : CONTROLLER RAM DEFINITIONS
1101      :-
1101      000167      RMCHBEG = 167      ;CHARACTERISTICS IO DATA BEGIN RAM ADDRESS
1102      000200      RMCHEND = 200     ;CHARACTERISTICS IO DATA END RAM ADDRESS
1103      000020      RMPKTBEG= 20      ;COMMAND PACKET BEGIN RAM ADDRESS
1104      000027      RMPKTEND= 27     ;COMMAND PACKET END RAM ADDRESS
1105      000104      RMSGBEG= 104     ;MESSAGE BUFFER BEGIN RAM ADDRESS
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 17-3
 TUBO REGISTER AND PACKET DEFINITIONS

```

1106      000117      RMMSGEND= 117      ;MESSAGE BUFFER END RAM ADDRESS
1107      :+
1108      :
1109      :REGISTER DEFINITIONS IN THE MESSAGE BUFFER
1110      :
1111      :-
1112
1113      000006      XST0== 6      ;EXTENDED STATUS REGISTER 0 (WORD 4)
1114      000010      XST1== 8.      ;EXTENDED STATUS REGISTER 1 (WORD 5)
1115      000012      XST2== 10.      ;EXTENDED STATUS REGISTER 2 (WORD 6)
1116      000014      XST3== 12.      ;EXTENDED STATUS REGISTER 3 (WORD 7)
1117      000016      XST4== 14.      ;EXTENDED STATUS REGISTER 4 (WORD 8)
1118
1119
1120      :+
1121      :
1122      :OFFSETS TO WORD LOCATIONS IN PACKET DEFINITIONS
1123      :
1124      :-
1125
1126      000002      PKLOW = 2      ;LOW ORDER CHARACTERISTIC DATA POINTER
1127      000004      PKHI = 4      ;HIGH ORDER CHARACTERISTIC DATA POINTER
1128      000006      PKBCNT = 6      ;NUMBER OF BYTES IN DATA PACKET
1129
1130      000010      EXBCNT=10      ;NUMBER OF BYTES IN EXTENDED DATA PACKET
1131
1132      :+
1133      :DATA PACKET OFFSETS FOR WRITE SUBSYSTEM COMMAND
1134      :-
1135      000000      BSELO = 0      ;BYTE 0
1136      000001      BSEL1 = 1      ;BYTE 1
1137      000002      SEL2 = 2      ;WORD 2
1138      000004      SELDATA = 4      ;WORD 3
1139
1140      :+
1141      :BSELO SELECT CODES FOR WRITE SUBSYSTEM COMMAND
1142      :-
1143      000000      PW.NOP = 0      ;NO-OP
1144      000001      PW.RDRAM = 1      ;READ RAM
1145      000002      PW.WTRAM = 2      ;WRITE RAM
1146      000003      PW.RFIFO = 3      ;READ FIFO
1147      000004      PW.WFIFO = 4      ;WRITE FIFO
1148      000005      PW.RDSTAT = 5      ;READ STATUS
1149      000006      PW.WCTL = 6      ;WRITE TAPE CONTROL
1150      000007      PW.WFMT = 7      ;WRITE TAPE FORMAT
1151      000010      PW.WMISC = 10      ;WRITE MISCELLANEOUS
1152      000011      PW.WNPR = 11      ;WRITE NPR CONTROL
1153      000020      PW.D22 = 20      ;DO MICROTEST 22
1154      000021      PW.D11 = 21      ;DO MICROTEST 11
1155      000022      PW.D13 = 22      ;DO MICROTEST 13
1156      000023      PW.NO1311 = 23      ;DISABLE MICROTEST 11 AND 13
1157      000024      PW.RDEXT = 24      ;READ EXT. TAPE STATUS (NOT SUPPORTED BY ALL TRANSP)
1158
1159      :+
1160      :BSEL1 CODES FOR WRITE TAPE CONTROL
1161      :-
1162      000200      WC.IFAD = BIT7      ;IFAD - FORMATTER ADDRESS

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 17-4
TUBO REGISTER AND PACKET DEFINITIONS

1163	000100	WC.I0TAD	= BIT6	:ITADO	- TRANSPORT ADDRESS BIT 0
1164	000040	WC.I1TAD	= BIT5	:ITAD1	- TRANSPORT ADDRESS BIT 1
1165	000020	WC.I5RESV	= BIT4	:IRESV5	- RESERVED #5
1166	000010	WC.IREW	= BIT3	:IREW	- REWIND
1167	000004	WC.IRWU	= BIT2	:IRWU	- REWIND AND UNLOAD
1168	000002	WC.IFEN	= BIT1	:IFEN	- FORMATTER ENABLE
1169	000001	WC.IGO	= BIT0	:GO	
1170					
1171		:+			
1172		:BSEL1 CODES FOR WRITE FORMAT			
1173		:-			
1174	000200	WF.IHISP	= BIT7	:IHISP	- HIGH SPEED
1175	000100	WF.IWRT	= BIT6	:IWRT	- WRITE
1176	000040	WF.IREV	= BIT5	:IREV	- REVERSE
1177	000020	WF.IWFM	= BIT4	:IWFM	- WRITE FILE MARK
1178	000010	WF.IEDIT	= BIT3	:IEDIT	- EDIT
1179	000004	WF.IERASE	= BIT2	:IERASE	- ERASE
1180	000002	WF.I3RESV	= BIT1	:IRESV3	- RESERVED #3
1181	000001	WF.I4RESV	= BIT0	:IRESV4	- RESERVED #4
1182					
1183					
1184		:+			
1185		:BSEL1 CODES FOR WRITE MISCELLANEOUS SUBCOMMAND			
1186		:-			
1187	000200	MS.EXT	= BIT7	:INVERT SENSE OF EXTENDED FEATURES SWITCH	
1188	000020	MS.RSFIFO	= BIT4	:RESET FIFO AND INPUT PARITY ERRORR	
1189	000010	MS.RSTAPE	= BIT3	:RESET TAPE STATUS IN 2 FLIP-FLOPS	
1190	000006	MS.ATTN	= BIT2:BIT1	:ATTENTION TRIGGER FIELD	
1191	000001	MS.RSD	= BIT0	:RESET TIMER A,B THEN DELAY TIMES IN SEL2	
1192		:+			
1193		: MS.ATTN SUBCODES			
1194		:-			
1195	000000	MSA.NOP	= 0*2	:NO-OP (NOTHING TRIGGERED)	
1196	000002	MSA.VOL	= 1*2	:SIMULATE ON-LINE/OFF-LINE TRANSITION	
1197	000004	MSA.NRAM	= 2*2	:FORCE NON-FATAL RAM ERROR (FORCES ERRCODE 54)	
1198	000006	MSA.FRAME	= 3*2	:FORCE FATAL RAM ERROR (CAUSES SCE TO SET)	
1199		:+			
1200		: WRITE SUBSYSTEM WRITE NPR BSEL1 BIT DEFINITIONS			
1201		:-			
1202	000200	NP.IR	= BIT7	:INTERRUPT REQUEST (0-1 TRANSITION)	
1203	000100	NP.OUT	= BIT6	:TAPE DATA DIRECTION OUT (0= IN)	
1204	000040	NP.LOOP	= BIT5	:ENABLE TRANSPORT LOOPBACK	
1205	000020	NP.WRP	= BIT4	:WRITE CORRECT PARITY (SET=0 TO WRITE WRONG)	
1206		:+			
1207		: READ STATUS MESSAGE BUFFER BIT DEFINITIONS			
1208		:-			
1209					
1210	000200	S2.DIM	= BIT7	:WORD #9 BYTE 2 DATA IN MISS	
1211	000100	S2.ILW	= BIT6	: ILW H	
1212	000040	S2.OURDY	= BIT5	: OUT RDY H	
1213	000020	S2.INRDY	= BIT4	: IN RDY H	
1214	000010	S2.ATIMR	= BIT3	: TIMER A FLAG H	
1215	000004	S2.BTIMR	= BIT2	: TIMER B FLAG H	
1216	000003	S2.UNDEF	= BIT1:BIT0	: (UNDEFINED)	
1217	100000	S1.PARIN	= BIT15	:WORD #8 BYTE 1 PARIN H	
1218	000000	S1.I2RESV	= BIT14	: IRESV2	
1219	020000	S1.I1RESV	= BIT13	: IRESV1	

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 17-5
 TUBO REGISTER AND PACKET DEFINITIONS

1220	010000	S1.IEOT	= BIT12	:	IEOT L
1221	004000	S1.IIDENT	= BIT11	:	IIDENT H
1222	002000	S1.ICER	= BIT10	:	ICER H
1223	001000	S1.IFMK	= BIT9	:	IFMK H
1224	000400	S1.IHER	= BIT8	:	IHER H
1225	000200	SO.ISPEED	= BIT7	:	WORD #8 BYTE 0 ISPEED H
1226	000100	SO.IRDY	= BIT6	:	IRDY L
1227	000040	SO.IONL	= BIT5	:	IONL L
1228	000020	SO.ILDLP	= BIT4	:	ILDLP L
1229	000010	SO.IDBY	= BIT3	:	IDBY L
1230	000004	SO.IRWD	= BIT2	:	IRWD L
1231	000002	SO.IFBY	= BIT1	:	IFBY L
1232	000001	SO.IFPT	= BIT0	:	IFPT L
1233		:		:	
1234		:	SPECIAL KEYBOARD STUFF FOR MOVER PROGRAM	:	
1235	177560	TKS	=177560	:	:KEYBOARD STATUS REGISTER
1236	177562	TKB	=177562	:	:KEYBOARD DATA REGISTER
1237	177564	TPS	=177564	:	:CONSOLE PRINTER STATUS REGISTER
1238	177566	TPB	=177566	:	:CONSOLE PRINTER DATA REGISTER
1239	007776	HIMEM	=007776	:	:HIGH MEMORY MASK VALUE
1240		:	CONTROLLER DEFINITIONS	:	
1241	174400	CSR	=174400	:	:STATUS AND CONTROL REGISTER
1242	174402	BAR	=174402	:	:DL ADDRESS REGISTER
1243	174404	DAR	=174404	:	:PLATTER ADDRESS
1244	174406	MPR	=174406	:	:MULTIPURPOSE REGISTER
1245		:		:	
1246		:	CONTROLLER COMMANDS	:	
1247		:		:	
1248	000004	DLGETS	=4	:	:GET STATUS COMMAND
1249	000006	SEEK	=6	:	:SEEK TRACK AND HEAD SELECT
1250	000010	DLRDHD	=10	:	:READ SECTOR HEADER
1251	000014	READ	=14	:	:READ COMMAND
1252	000016	DLRDNH	=16	:	:READ SECTOR NO HEADER CHECK
1253		:		:	
1254	000001	READY	=1	:	:DRIVE READY BIT IN STATUS REG.
1255	000013	DLSR	=13	:	:STATUS AND RESET
1256	177730	DLERR	=177730	:	:MASK FOR COVER OPEN
1257	000006	DLUN	=6	:	:HEADS UNLOADED
1258	000177	DLCYL	=000177	:	:MASK FOR CYLINDER ADDRESS
1259	100200	DLDNER	=100200	:	:DONE SET OR ERROR SET BITS
1260		:		:	
1261	177560	TTICSR	= 177560	:	:KEYBOARD INPUT STATUS
1262	177562	TTIBFR	= 177562	:	:KEYBOARD DATA REGISTER
1263	177564	TTOCSR	= 177564	:	:CONSOLE PRINTER STATUS REGISTER
1264	177566	TTOBFR	= 177566	:	:CONSOLE PRINTER DATA REGISTER
1265					

CZTUXAO TUBO FRONT END Pr
SPECIAL MACROS AND OPDEFS.

MACRO M1200 29-MAR-83 13:32 PAGE 18

```

1267             .SBTTL SPECIAL MACROS AND OPDEFS.
1268
1269
1270             :+
1271             :SAVE GENERAL REGS 1 TO 5
1272             :-
1273
1274             .MACRO SAVREG
1275             JSR   R5,REGSAV
1276             .ENDM
1277
1278             :+
1279             : MACRO TO FORCE AN ERROR
1280             :-
1281             .MACRO FORCERROR TAG,NOTSSR
1282             .NLIST
1283             .IIF NDF LISTALL, .NLIST
1284             .LIST
1285             .IF B NOTSSR
1286             MOV   TSSR(R5),R1           ;READ TSSR
1287             .ENDC
1288             MOV   FORCER,FORCER       ;IS FORCER SET? (LEAVE C BIT ALONE)
1289             BNE   TAG                 ;BR IF YES
1290             .NLIST
1291             .IIF NDF LISTALL, .LIST
1292             .LIST
1293             .ENDM
1294
1295             :+
1296             : MACRO TO FORCE AN EXIT TO AVOID SECTION ITERATIONS
1297             : WILL EXIT TO A LABEL IF FORCER IS NEGATIVE
1298             : SO TO FORCE ERRORS AND EXIT ON 1 ERROR SET
1299             : FORCER TO 17777
1300             : TO FORCE ERRORS AND ITERATIONS SET FORCER TO 1.
1301             :-
1302             .MACRO FORCEEXIT TAG
1303             .NLIST
1304             .IIF NDF LISTALL, .NLIST
1305             .LIST
1306             MOV   FORCER,FORCER       ;IS FORCER NEGATIVE?
1307             BMI   TAG                 ;BR IF YES
1308             .NLIST
1309             .IIF NDF LISTALL, .LIST
1310             .LIST
1311             .ENDM
1312             :+
1313             : MACRO TO INCREMENT ERROR COUNTS
1314             :-
1315             .MACRO NEXT.ERRNO
1316             .NLIST
1317             :::: .IIF NDF LISTALL, .NLIST
1318             ERRNO=ERRNO+1
1319             :::: .IIF NDF LISTALL, .LIST
1320             .LIST
1321             .ENDM
1322
1323             :+

```

CZTUXAO TUBO FRONT END PRT B
SPECIAL MACROS AND OPDEFS.

MACRO M1200 29-MAR-83 13:32 PAGE 18-1

1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347

:MACRO TO PERFORM XOR

:-

```
.MACRO XOR A,B
MOV A,-(SP)
BIC B,(SP)
BIC A,B
BIS (SP)+,B
.ENDM
```

000000

```
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.
```

CZTUXAO TUBO FRONT END PRT B
GLOBAL DATA SECTION

MACRO M1200 29-MAR-83 13:32 PAGE 19

.SBTTL GLOBAL DATA SECTION

```

1349
1350
1351      :++
1352      :THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1353      :IN MORE THAN ONE TEST.
1354      :--
1355
1356      :
1357      :THE FOLLOWING DATA ARE SET FOR EACH UNIT AT INIT TIME.
1358      :SINGLE UNIT DEFAULTS (LISTED) ARE IN THE DEFAULT P-TABLE.
1359      :
1360 002146 000000  EPRTSW::      .WORD  0      ;PRINT SWITCH
1361 002150 000000  UNITN::      .WORD  0      ;UNIT # UNDER TEST.
1362 002152 000000  QVP::       .WORD  0      ;QUICK VERIFY FLAG.
1363 002154 000000  CSRADDR::   .WORD  0      ;ADDRESS OF CSR FOR CURRENT DEVICE
1364 002156 000224  IVEC::      .WORD  224    ;INTERRUPT VECTOR
1365 002160 000200  IPRI::      .WORD  PRI04  ;INTERRUPT PRIORITY.
1366 002162 000000  TSTCNT::    .WORD  0      ;NUMBER OF TESTS RUN IN THIS PASS
1367 002164 000000  LOOPCNT::   .WORD  0      ;REMAINING ITERATION COUNT FOR TEST
1368 002166 000000  DEVCNT::    .WORD  0      ;NUMBER OF DEVICE UNDER TEST
1369 002170 000000  FATFLG::    .WORD  0      ;SET IF FATAL ERROR IS DETECTED IN TEST
1370 002172 000000  INTRECV::   .WORD  0      ;SET IF TAPE INTERRUPT WAS RECEIVED
1371 002174 000000  BENBSW::    .WORD  0      ;BUFFER ENABLE SWITCH SW 0=OFF;1=ON
1372 002176 000000  EXPD::      .WORD  0      ;EXPECTED RAM DATA FOR PRAMPKT ROUTINE
1373 002200 000000  RECV::      .WORD  0      ;RECEIVED RAM DATA FOR PRAMPKT ROUTINE
1374 002202 000000  ERRHI::     .WORD  0      ;HIGH ADDRESS MEMORY ERROR
1375 002204 000000  ERRLO::     .WORD  0      ;LOW ADDRESS MEMORY ERROR
1376 002206 000000  RAMDATA::   .BLKB  16.    ;DATA READ FROM RAM PACKET OR MESSAGE BUF AREA
1377 002246 000000  RAMSIZ::    .WORD  0      ;RAM DATA SIZE FOR PRAMPKT ROUTINE
1378 002250 000000  RCVHIADD::  .WORD  0      ;RECEIVED BUFFER HIGH ADDRESS
1379 002252 000000  RCVLOADD::  .WORD  0      ;RECEIVED BUFFER LOW ADDRESS
1380 002254 000000  COUNT::     .WORD  0      ;TEST COUNT PATTERN
1381 002256 000000  DATA::     .WORD  0      ;TEST DATA
1382 002260 000000  TSTFLAG::   .WORD  0      ;TEST FLAG WORD
1383 002262 000000  TSTPTR::    .WORD  0      ;TSTBLK POINTER
1384 002264 000000  PRMNO::     .WORD  0      ;PRINT ROUTINE TEMP
1385 002266 000000  EXPMSG::    .BLKB  100.   ;EXPECTED MESSAGE BUFFER DATA
1386 002432 000000  RECMSG::    .BLKB  100.   ;RECEIVED MESSAGE BUFFER DATA
1387 002576 000000  TMPBFR::    .BLKB  80.    ;TEMPORARY STORAGE FOR PRINT
1388 002716 000000  MESBFA::    .WORD  0      ;STORES ADDRESS OF MESSAGE BUFFER FOR ERR PRT

```

CZTUXAO TU80 FRONT END PRT B
TSTBLK - TEST DATA TABLE

MACRO M1200 29-MAR-83 13:32 PAGE 20

.SBTTL TSTBLK - TEST DATA TABLE

1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405

.*
: 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
:-

1406 002720
1407 002720 000000
1408 002722 177777
1409 002724 000001
1410 002726 000002
1411 002730 000004
1412 002732 000010
1413 002734 000020
1414 002736 000040
1415 002740 000100
1416 002742 000200
1417 002744 000400
1418 002746 001000
1419 002750 002000
1420 002752 004000
1421 002754 010000
1422 002756 020000
1423 002760 040000
1424 002762 100000
1425 002764 177776
1426 002766 177775
1427 002770 177773
1428 002772 177767
1429 002774 177757
1430 002776 177737
1431 003000 177677
1432 003002 177577
1433 003004 177377
1434 003006 176777
1435 003010 175777
1436 003012 173777
1437 003014 167777
1438 003016 157777
1439 003020 137777
1440 003022 077777
1441 003024 125252
1442 003026 052525
1443 003030

TSTBLK::

.WORD 0
.WORD 177777
.WORD BIT0
.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
.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
.WORD 052525

: ALL ZEROS
: ALL ONES
: DATA FOR WALKING ONES

: DATA FOR WALKING ZEROS

: ALTERNATING ONES, ZEROS
: ALTERNATING ONES, ZERO OPPOSITE FROM ABOVE

TBLEND==.


```

1445          .SBTTL GLOBAL ENVIRONMENT STORAGE
1446          ;
1447          ;STORAGE FOR DEVICE REGISTERS
1448          ;
1449 003030 000000 100000 000000 DUMMY: 0,100000,0,U          ;DUMMY DEVICE REGISTERS...
1450 003040 000000 000000 000000          0,0,0,0,0,0,0,0          ;...FOR MULTI-UNIT CHECKOUT.
1451
1452
1453
1454 003060 000000          DUFLG::          .WORD 0          ;'DROPPED UNIT' FLAG.
1455          ;INHIBITS CODE IN 'CLEAN-UP'.
1456 003062 000000          NODEV::          .WORD 0          ;FLAG TO SAY NO DEVICE.
1457
1458 003064 000000          TEMP1::          .WORD 0          ;SOME TEMP LOCATIONS.
1459 003066 000000          TEMP2::          .WORD 0
1460 003070 000000          XXCOMM::          .WORD 0          ;XXDP+ COMM BLOCK POINTER.
1461 003072 000000          FREE::          .WORD 0          ;1ST FREE MEMORY ADDRESS...
1462 003074 000000          FRESIZ::          .WORD 0          ;...AND SIZE (IN WORDS).
1463 003076 000000          FREEHI:          .WORD 0          ;LAST WORD IN FREE SPACE
1464 003100 000000          KTFLG::          .WORD 0          ;KT11, MEM AVAIL FLAG -
1465          ;- .WORD 0 = <24K OR NO KT -
1466          ;- NZ = >24K AND KT.
1467 003102 000000          KTENABLE::          .WORD 0          ;SET BY TEST ROUTINES TO FLAG >28K UNDER TEST
1468 003104 002000          PST32W::          .WORD 2000          ;32W BLOCK ADDRESS FOR 32K START
1469 003106 000000          SIFLAG::          .WORD 0
1470 003110 000000          BADDAT::          .WORD 0          ;ACTUAL DATA
1471 003112 000000          GDDAT::          .WORD 0          ;EXPECTED DATA
1472 003114 000000          LOOPFL::          .WORD 0
1473 003116          CTAB::          ;CONFIGURATION TABLES.
1474 003116 000000          CTABM::          .WORD 0          ;CONFIG WORK.
1475 003120          .WORD 0
1476 003122          .WORD 0
1477 003124          .WORD 0
1478 003126 177777          .WORD -1          ;END OF MEM TABLE.
1479 003130          CTABE::
1480          ;ERROR STATISTICS TABLE (1 WORD PER UNIT), 64 UNITS MAX:
1481          ;
1482          ;          0          =          UNIT NOT TESTED
1483          ;          100000          =          UNIT ONLINE, NO ERRORS
1484          ;          10XXXX          =          UNIT ONLINE, ENCOUNTERED XXXX ERRORS
1485          ;          160000          =          UNIT DROPPED, NON-EXISTENT DEVICE REGISTER
1486          ;          160001          =          UNIT DROPPED, NOT IDLE AT START
1487          ;          14XXXX          =          UNIT DROPPED, ENCOUNTERED XXXX ERRORS
1488          ;
1489 003130          ERTABL:          .BLKW 64.
1490 003330 000000          ERTABE:          .WORD 0
1491
1492 003332 000000          SKIPT:          .WORD 0          ;1=SKIP SUBTEST 0=NO SKIP OF SUBTEST

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 22
GLOBAL TEXT MESSAGES

.SBTTL GLOBAL TEXT MESSAGES

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

:+
: NAMES OF DEVICES SUPPORTED
:--

1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507 003334
003334
003334 124 125 070

DEV TYP <TUBO>
LSDVTYP::
.ASCIZ /TUBO/
.EVEN

1508
1509
1510
1511
1512 003342
003342
003342 103 132 124

:+
: TEST DESCRIPTION
:--
DESCRIP <CZTUXAO TUBO FRONT END PRT B>
LSDDESC::
.ASCIZ /CZTUXAO TUBO FRONT END PRT B/
.EVEN

1513
1514
1515
1516

:+
: BIT TO ASCII CONVERSION FOR TSSR REGISTER
:--

1517 003400 003440 003443 003447
1518 003420 003501 003505 003511
1519 003440 123 103 000
1520 003443 102 111 105
1521 003447 123 103 105
1522 003453 122 115 122
1523 003457 116 130 115
1524 003463 116 102 101
1525 003467 102 111 124
1526 003474 102 111 124
1527 003501 123 123 122
1528 003505 117 106 114
1529 003511 102 111 124
1530 003516 102 111 124
1531 003523 102 111 124
1532 003530 102 111 124
1533 003535 102 111 124
1534 003542 102 111 124
1535
1536 003550 124 123 123
1537 003603 124 123 123
1538 003636 040 040 116
1539 003675 045 101 040
1540 003716 045 101 040
1541 003756 045 101 040
1542 004015 045 116 045
1543 004021 040 040 125
1544 004050 040 040 111

TSSRBIT:: .WORD 1\$,2\$,3\$,4\$,5\$,6\$,7\$,8\$
 .WORD 9\$,10\$,11\$,12\$,13\$,14\$,15\$,16\$
1\$: .ASCIZ 'SC'
2\$: .ASCIZ 'BIE'
3\$: .ASCIZ 'SCE'
4\$: .ASCIZ 'RMR'
5\$: .ASCIZ 'NXM'
6\$: .ASCIZ 'NBA'
7\$: .ASCIZ 'BIT9'
8\$: .ASCIZ 'BIT8'
9\$: .ASCIZ 'SSR'
10\$: .ASCIZ 'OFL'
11\$: .ASCIZ 'BIT5'
12\$: .ASCIZ 'BIT4'
13\$: .ASCIZ 'BIT3'
14\$: .ASCIZ 'BIT2'
15\$: .ASCIZ 'BIT1'
16\$: .ASCIZ 'BIT0'
.EVEN
SFIERR: .ASCIZ 'TSSR ERROR AFTER SOFT INIT'
SFHERR: .ASCIZ 'TSSR ERROR AFTER BUS RESET'
NXR: .ASCIZ / NON-EXISTANT DEVICE REGISTER/
NXRX: .ASCIZ /XA ADDRESS: X06/
TSSX: .ASCIZ /XA TSBA,TSSR EXP'D: X06XA,X06XN/
.ASCIZ /XA TSBA,TSSR REC'D: X06XA,X06/
FUSI: .ASCIZ /XNXA/
USI: .ASCIZ / UNEXPECTED INTERRUPT/
NSI: .ASCIZ / INTERRUPT EXPECTED, NOT RECEIVED/

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 22-1
 GLOBAL TEXT MESSAGES

1545	004113	045	116	045	FNOINTR:	.ASCII /XNZA/
1546	004117	040	040	116	NOINTR:	.ASCIZ / NO INTERRUPT WAS GENERATED/
1547	004154	040	040	111	IFault:	.ASCIZ / INTERRUPT FAULT/
1548	004176	045	101	040	INTX:	.ASCIZ /%A CPU PC: X06XA TSBA: X06/
1549	004233	040	040	042	NOINIT:	.ASCIZ / 'BUS-INIT' DIDN'T INITIALIZE CONTROLLER/
1550	004305	040	040	042	NSINIT:	.ASCIZ / 'SOFT-INIT' DIDN'T INITIALIZE THE DPU/
1551	004355	040	040	042	BRINIT:	.ASCIZ / 'BUS-RESET' DIDN'T INITIALIZE THE DPU/
1552						
1553	004425	000			NUL:	.ASCIZ //
1554	004426	045	116	000	NULCR:	.ASCIZ /XN/
1555	004431	045	101	040	EXPGOT:	.ASCIZ /%A EXP'D: X06XA, REC'D: X06/
1556	004465	045	116	045	EXPGT2:	.ASCIZ /XNZA EXP'D: X06XA, X06XNZA REC'D: X0XA, X06/
1557	004541	045	101	040	DUAD12:	.ASCIZ /%A REG(W) WRITTEN TO: X06XA REG(R) READ: EXP'D: X06XA, REC'D: X06/
1558	004643	122	101	115	PKTRAM:	.ASCIZ 'RAM Contents Do Not Match Packet Sent'
1559	004711	040	040	103	SCME:	.ASCIZ / CONFIG DOESN'T MATCH MFG. MASTER/
1560	004754	127	122	111	WRTMSG:	.ASCIZ 'WRITE CHARACTERISTICS Failed'
1561	005011	124	123	123	WRTERR:	.ASCIZ 'TSSR Incorrect After WRITE Command, More Bits Set Than SSR'
1562	005104	174	123	123	RDERR:	.ASCIZ 'TSSR Incorrect After READ Command, More Bits Set Than SSR'
1563						.EVEN
1564						
1565						
1566						

CZTUXAO TUBO FRONT END PRT B
GLOBAL ERROR REPORT SECTION

MACRO M1200 29-MAR-83 13:32 PAGE 23

.SBTTL GLOBAL ERROR REPORT SECTION

1568
1569
1570
1571
1572
1573
1574
1575

```

:++
: 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.
:--
    
```

1576 005176
005176
1577 005176
005176 013746 003062
005202 012746 003675
005206 012746 000002
005212 010600
005214 104415
005216 062706 000006
1578 005222 004737 005230
1579 005226
005226
005226 104423

```

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$PNTX
ADD #6,SP
JSR PC,EXTEND ; PRINT EXTENSION IF REQUIRED.
ENDMSG
L10002:
TRAP C$MSG
    
```

1580
1581
1582
1583
1584
1585

```

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

1586 005230 005727
1587 005232 000000
1588 005234 001402
1589 005236 004777 177770
1590 005242
005242 012746 004426
005246 012746 000001
005252 010600
005254 104415
005256 062706 000004
1591 005262 000207

```

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
    
```

CZTUXAO TUBG FRONT END PRT B
PRITSSR - PRINT TSSR CONTENTS

MACRO M1200 29-MAR-83 13:32 PAGE 25

.SBTTL PRITSSR - PRINT TSSR CONTENTS

1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611

```

:ROUTINE TO DISPLAY THE CONTENTS, AND BIT DEFINITIONS, OF
:THE TSSR REGISTER. THIS ROUTINE IS NORMALLY CALLED ONLY
:BY A MESSAGE PRINTING ROUTINE

```

:INPUTS:

R1 CONTENTS OF TSSR

:SUBORDINATE ROUTINES:

CHKAMB CHECK FOR AMBIGUOUS CONTENTS

PRITSSR:

1612	005264			SAVREG		:SAVE GENERAL REGISTERS
1613	005264			MOV R1,R4		:SAVE THE TSSR CONTENTS
1614	C05270	010104		PRINTB #TSSRFOR,R4		:PRINT THE CONTENTS OF TSSR
1615	005272			MOV R4,-(SP)		
	005272	010446		MOV #TSSRFOR,-(SP)		
	005274	012746	006105	MOV #2,-(SP)		
	005300	012746	000002	MOV SP,R0		
	005304	010600		TRAP CSPNTB		
	005306	104414		ADD #6,SP		
	005310	062706	000006	MOV R4,R0		:GET TSSR BACK FOR CHKAMB
1616	005314	010400		JSR PC,CHKAMB		:ARE CONTENTS AMBIGUOUS ?
1617	005316	004737	016700	BCS 5\$:BRANCH IF NOT
1618	005322	103410		PRINTX #AMBTSSR		:SHOW CONTENTS ARE AMBIGUOUS
1619	005324			MOV #AMBTSSR,-(SP)		
	005324	012746	006325	MOV #1,-(SP)		
	005330	012746	000001	MOV SP,R0		
	005334	010600		TRAP CSPNTX		
	005336	104415		ADD #4,SP		
	005340	062706	000004	5\$: MOV R4,R3		:CONTENTS OF TSSR
1620	005344	010403		BIC #HIADDR!FATERR!TERCLS,R3		:CLEAR ALL MULTIPLE BIT FIELDS
1621	005346	042703	001476	BEQ 20\$:NO BITS ARE SET
1622	005352	001434		MOV #TMPBFR,R2		:TEMPORARY ASCII BUFFER
1623	005354	012702	002576	MOV #TSSRBIT,R1		:ASCII EQUIVALENT OF BITS
1624	005360	012701	003400	10\$: TST R3		:REMAINING BITS TO CONVERT
1625	005364	005703		BEQ 15\$:BRANCH WHEN ALL ARE DONE
1626	005366	001413		CLC		:CLEAR CARRY FOR SHIFT
1627	005370	000241		ROL R3		:SHIFT NEXT BIT TO CARRY
1628	005372	006103		BCC 13\$:BRANCH IF BIT NOT SET
1629	005374	103006		MOV (R1),R0		:POINTER TO BIT DEFINITION
1630	005376	011100		11\$: MOV (R0)+,(R2)+		:MOVE ASCII TO BUFFER
1631	005400	112022		BNE 11\$:MOVE ALL BITS
1632	005402	001376		MOV #',-1(R2)		:INSERT A COMMA TO TERMINATE
1633	005404	112762	000054 177777	13\$: TST (R1)+		:POINT TO NEXT DESCRIPTION
1634	005412	005721		BR 10\$:GET THE REMAINING BITS
1635	005414	005763		15\$: CLRB -(R2)		:TERMINATE THE LINE
1636	005416	105042		PRINTX #TSSDEF,#TMPBFR		:PRINT THE BIT DEFINITIONS
1637	005420			MOV #TMPBFR,-(SP)		
	005420	012746	002576	MOV #TSSDEF,-(SP)		
	005424	012746	006276			

CZTUXAO TU80 FRONT END PRT B
PRITSSR - PRINT TSSR CONTENTS

MACRO M1200 29-MAR-83 13:32 PAGE 25-1

005430	012746	000002			MOV	#2,-(SP)		
005434	010600				MOV	SP,R0		
005436	104415				TRAP	CSPNTX		
005440	062706	000006			ADD	#5,SP		
1638								
1639	005444	010403		208:	MOV	R4,R3	:GET THE TSSR CONTENTS	
1640	005446	042703	177761		BIC	#^CTERCLS,R3	:CLEAR ALL BUT TERMINATION	
1641	005452	016303	006366		MOV	TCOCOD(R3),R3	:GET THE TERMINATION CODE MEANING	
1642	005456				PRINTX	#TCOASC,R3	:PRINT THE TERMINATION CODE	
	005456	010346			MOV	R3,-(SP)		
	005460	012746	006166		MOV	#TCOASC,-(SP)		
	005464	012746	000002		MOV	#2,-(SP)		
	005470	010600			MOV	SP,R0		
	005472	104415			TRAP	CSPNTX		
	005474	062706	000006		ADD	#6,SP		
1643	005500	010403			MOV	R4,R3	:TSSR CONTENTS AGAIN	
1644	005502	042703	177717		BIC	#^CFATERR,R3	:CLEAR ALL BUT FATAL TERMINATION	
1645	005506	001421			BEQ	258	:DON'T PRINT IF ZERO	
1646	005510	006203			ASR	R3		
1647	005512	006203			ASR	R3		
1648	005514	006203			ASR	R3	:ALINE TERMINATION CODE FOR INDEX	
1649	005516	016303	006726		MOV	TSFCOD(R3),R3	:GET THE FATAL TERMINATION CODE	
1650	005522				PRINTX	#TFCASC,R3	:PRINT THE FATAL TERMINATION CODE	
	005522	010346			MOV	R3,-(SP)		
	005524	012746	006227		MOV	#TFCASC,-(SP)		
	005530	012746	000002		MOV	#2,-(SP)		
	005534	010600			MOV	SP,R0		
	005536	104415			TRAP	CSPNTX		
	005540	062706	000006		ADD	#6,SP		
1651	005544	012737	000031	002170	MOV	#25.,FATFLG	:DROP THIS UNIT AFTER ERROR	
1652	005552	010403			MOV	R4,R3	:GET TSSR CONTENTS	
1653	005554	042703	176377		BIC	#^CHIADDR,R3	:CLEAR ALL BUT EXTENDED ADDRESS	
1654	005560	001411			BEQ	308	:DON'T PRINT IF ZERO	
1655	005562				PRINTX	#TEXASC,R3	:PRINT THE EXTENDED ADDRESS BITS	
	005562	010346			MOV	R3,-(SP)		
	005564	012746	006125		MOV	#TEXASC,-(SP)		
	005570	012746	000002		MOV	#2,-(SP)		
	005574	010600			MOV	SP,R0		
	005576	104415			TRAP	CSPNTX		
	005600	062706	000006		ADD	#6,SP		
1656	005604	022704	100210		308:	CMP	#100210,R4	:CHECK FOR MEDIA ERROR
1657	005610	001003			BNE	318	:BR, IF PROBABLY NOT TAPE ERROR	
1658	005612	012737	006014	002146	MOV	#EPRT3,EPRTSW	: 'PROBABLY MEDIA RELETED ERROR - BAD TAPE'	
1659	005620	005737	002146		318:	TST	EPRTSW	:CHECK FOR THE SWITCH EMPTY
1660	005624	001003			BNE	3108	:BR, IF SWITCH IS NOT EMPTY	
1661	005626	012737	005755	002146	MOV	#EPRT1,EPRTSW	:SET SWITCH TO DEFAULT	
1662	005634	013737	002146	005644	3108:	MOV	EPRTSW,328+2	:PUT REAL SWITCHABLE MESSAGE IN PLACE
1663	005642				328:	PRINTB	#EPRT1	:PRINT THE ERROR MESSAGE
	005642	012746	005755		MOV	#EPRT1,-(SP)		
	005646	012746	000001		MOV	#1,-(SP)		
	005652	010600			MOV	SP,R0		
	005654	104414			TRAP	CSPNTB		
	005656	062706	000004		ADD	#4,SP		
1664	005662	012737	005755	002146	MOV	#EPRT1,EPRTSW	:RESET TO NORMAL ERROR POINTER	
1665	005670	090207			RTS	PC	:RETURN TO CALLER	
1666	005672	045	116	045	EPRT2:	.ASCIZ	'ZXZA *****CHECK M7454, CABLES AND TRANSPORT*****XS'	
1667	005755	045	116	045	EPRT1:	.ASCIZ	'ZXZA *****REPLACE M7454*****XS'	

CZTUXAO TU80 FRONT END PRT B
 PRITSSR - PRINT TSSR CONTENTS

MACRO M1200 29-MAR-83 13:32 PAGE 25-2

1668	006014	045	116	045	EFRT3:	.ASCIZ	'XNZA *****POSSIBLE MEDIA RELATED ERROR - BAD TAPE*****XS'
1669							
1670	006105	045	116	045	TSSRFOR:	.ASCIZ	'XNZA TSSR = X06'
1671	006125	045	116	045	TEXASC:	.ASCIZ	'XNZA Extended Address Bits = X06'
1672	006166	045	116	045	TCOASC:	.ASCIZ	'XNZA Termination Class Code = XT'
1673	006227	045	116	045	TFCASC:	.ASCIZ	'XNZA Fatal Termination Class Code = XT'
1674	006276	045	116	045	TSSDEF:	.ASCIZ	'XNZA TSSR Bits Set: XT'
1675	006325	045	116	045	AMBTSSR:	.ASCIZ	'XNZA TSSR Contents Are Ambiguous'
1676						.EVEN	
1677	006366	006406	006431	006457	TCOCOD:	.WORD	1S,2S,3S,4S,5S,6S,7S,8S
1678	006406	116	157	162	1S:	.ASCIZ	'Normal Termination'
1679	006431	124	145	162	2S:	.ASCIZ	'Termination Condition'
1680	006457	124	141	160	3S:	.ASCIZ	'Tape Status Alert'
1681	006501	106	165	156	4S:	.ASCIZ	'Function Reject'
1682	006521	122	145	143	5S:	.ASCIZ	'Recoverable Error - Tape Position One Record Down'
1683	006603	122	145	143	6S:	.ASCIZ	'Recoverable Error - Tape Was Not Moved'
1684	006652	125	156	162	7S:	.ASCIZ	'Unrecoverable Error'
1685	006676	106	141	164	8S:	.ASCIZ	'Fatal Contruller Error'
1686						.EVEN	
1687							
1688	006726	006736	006772	007003	TSFCOD:	.WORD	1S,2S,3S,4S
1689	006736	111	156	164	1S:	.ASCIZ	'Internal Diagnostic Failure'
1690	006772	122	145	163	2S:	.ASCIZ	'Reserved'
1691	007003	102	165	163	3S:	.ASCIZ	'Bus Interface or Sanity Check Error'
1692	007047	122	145	163	4S:	.ASCIZ	'Reserved'
1693						.EVEN	

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 26
 PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET

```

1695          .SBTTL PRIPKT - PRINT THE ADDRESS/CONTENTS OF COMMAND PACKET
1698          :THIS ROUTINE PRINTS THE ADDRESS AND CONTENTS OF A COMMAND PACKET.
1699          :THIS ROUTINE IS NORMALLY ONLY CALLED FROM A PRINT ROUTINE.
1700
1701          :INPUT:
1703          :      R0      NUMBER OF WORDS IN PACKET
1704          :      R3      HIGH ORDER COMMAND PACKET ADDRESS
1705          :      R4      ADDRESS OF COMMAND PACKET
1707          :      NOTE:   R3 IS IGNORED IF THE KTENABLE FLAG IS CLEAR.
1710 007060   PRIPKT::
1711 007060   SAVREG                ;SAVE THE REGISTERS
1712 007064   010005                ;SAVE NO. OF WORDS IN PACKET
1713 007066   005737 003102         ;ABOVE 28K UNDER TEST?
1714 007072   001001                ;BR IF YES
1715 007074   005003                ;SET HIGH ORDER ADDRESS TO 0
1716 007076   010301                ;COPY HIGH ORDER ADDRESS
1717 007100   010400                ;GET LOWER ADDRESS
1718 007102   006100                ;SHIFT BIT 15 INTO C BIT
1719 007104   006101                ;AND INTO HIGH ORDER.
1720 007106   PRINTB #PKTADD,R1,R4 ;PRINT PACKET ADDRESS
          007106   010446           MOV     R4,-(SP)
          007110   010146           MOV     R1,-(SP)
          007112   012746 007264     MOV     #PKTADD,-(SP)
          007116   012746 000003     MOV     #3,-(SP)
          007122   010600           MOV     SP,R0
          007124   104414           TRAP   CSPNTB
          007126   062706 000010     ADD     #10,SP
1721 007132   010300                15$:   MOV     R3,R0                ;GET HIGH ORDER ADDRESS
1722 007134   001404                BEQ     20$                ;BR IF NOT ABOVE 28K.
1723 007136   010401                MOV     R4,R1                ;GET LOW ORDER ADDRESS
1724 007140   004737 020252         JSR     PC,SETMAP           ;SETUP PAR6 MAPPING FOR 18 BIT ADDRESS
1725 007144   010004                MOV     R0,R4                ;GET RETURNED PAR6 ADDRESS BIAS
1726 007146   005001                20$:   CLR     R1                ;SAVE WORD NUMBER
1727 007150   012402                25$:   MOV     (R4)+,R2           ;GET PACKET CONTENTS
1728 007152   PRINTB #PKTFRM,R1,R2 ;PRINT THE DATA
          007152   010246           MOV     R2,-(SP)
          007154   010146           MOV     R1,-(SP)
          007156   012746 007226     MOV     #PKTFRM,-(SP)
          007162   012746 000003     MOV     #3,-(SP)
          007166   010600           MOV     SP,R0
          007170   104414           TRAP   CSPNTB
          007172   062706 000010     ADD     #10,SP
1729 007176   005201                INC     R1                ;NEXT WORD NUMBER
1730 007200   020105                CMP     R1,R5                ;DONE ALL PACKET WORDS?
1731 007202   002762                BLT     25$                ;LOOP TILL ALL DONE
1732 007204   PRINTB #PKTNEW        ;JUST A COUPLE NEW LINES
          007204   012746 007321     MOV     #PKTNEW,-(SP)
          007210   012746 000001     MOV     #1,-(SP)
          007214   010600           MOV     SP,R0
          007216   104414           TRAP   CSPNTB
          007220   062706 000004     ADD     #4,SP
1733 007224   000207                RTS     PC                ;RETURN
1735 007226   045 116 045 PKTFRM: .ASCIZ 'XNZX Packet Word #XD1XA = X06'
1736 007264   045 116 045 PKTADD: .ASCIZ 'XNZX Packet Address = X01X05'
1738 007321   045 116 045 PKTNEW: .ASCIZ 'XNZX '
1739          .EVEN

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 27
 PRIBXOR - PRINT EXPD, RECV AND XOR BYTE

1742
 1743
 1744
 1745
 1746
 1747
 1748
 1749
 1750
 1751
 1752
 1753
 1754
 1755
 1756
 1757
 1758
 1759
 1760
 1761
 1762
 1763
 1764
 1765
 1766
 1767
 1768
 1769
 1770
 1771
 1772
 1773
 1774

.SBTTL PRIBXOR - PRINT EXPD, RECV AND XOR BYTE

```

: +
: PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE DATA BYTE
: THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
: INPUTS:
:     R1     RECEIVED DATA
:     R2     EXPECTED DATA
: OUTPUT:
:     R0     XOR OF EXPECTED/RECEIVED DATA
: -
    
```

```

PRIBXOR::
    SAVREG                ;SAVE THE REGISTERS
    MOV R2,R3              ;EXPECTED DATA
    XOR R1,R3              ;FORM THE EXCLUSIVE OR
    MOV #^C<377>,R0       ;BYTE MASK
    BIC R0,R1              ;SAVE LOW BYTE RECV
    BIC R0,R2              ;SAVE LOW BYTE EXPD
    BIC R0,R3              ;SAVE LOW BYTE XOR
    PRINTB #XORBFOR,R2,R1,R3 ;PRINT THE MESSAGE
    MOV R3,-(SP)
    MOV R1,-(SP)
    MOV R2,-(SP)
    MOV #XORBFOR,-(SP)
    MOV #4,-(SP)
    MOV SP,R0
    TRAP CSPNTB
    ADD #12,SP
    MOV R3,R0              ;R0 HAS XOR ON RETURN
    RTS PC                 ;RETURN TO CALLER

045 XORBFOR: .ASCIZ 'XNXA EXPD: X03XA RECV: X03XA XOR: X03'
.EVEN
    
```

```

007332
007332 010203
007336 012700 177400
007354 040001
007356 040002
007360 040003
007362 010346
007364 010146
007366 010246 007414
007370 012746 000004
007374 012746 000004
007400 010600
007402 104414
007404 062706 000012
007410 010300
007412 000207
    
```

CZTUXAO TUGO FRONT END PRT B MACPO M1200 29-MAR-83 13:32 PAGE 28
PRI XOR - PRINT EXPD, RECV AND XOR

.SBTTL PRI XOR - PRINT EXPD, RECV AND XOR

1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793

:+
:PRINT EXPECTED DATA, RECEIVED DATA, AND XOR OF THE TWO
:THIS ROUTINE IS NORMALLY CALLED ONLY FOR PRINT ROUTINES.
:INPUTS:
:R1 RECEIVED DATA
:R2 EXPECTED DATA
:OUTPUT:
:R0 XOR OF EXPECTED/RECEIVED DATA
:-

1794 007462
1795 007462
1796 007466 010203
1797 007470
1798 007500
007500 010346
007502 010146
007504 010246
007506 012746 007532
007512 012746 000004
007516 010600
007520 104414
007522 062706 000012
1799 007526 010300
1800 007530 000207
1801
1802 007532 045 116 045
1803

PRI XOR::
SAVREG ;SAVE THE REGISTERS
MOV R2,R3 ;EXPECTED DATA
XOR R1,R3 ;FORM THE EXCLUSIVE OR
PRINTB #XORFOR,R2,R1,R3 ;PRINT THE MESSAGE
MOV R3,-(SP)
MOV R1,-(SP)
MOV R2,-(SP)
MOV #XORFOR,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #12,SP
MOV R3,R0 ;R0 HAS XOR ON RETURN
RTS PC ;RETURN TO CALLER
XORFOR: .ASCIZ 'X%X EXPD: X06XA RECV: X06XA XOR: X06'
.EVEN

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 29
 PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT

1805 .SBTTL PRIEQU - PRINT BIT NUMBERS AS ASCII EQUIVALENT

1806
 1807 :+
 1808 :ROUTINE TO CONVERT BIT VALUES TO ASCII AND PRINT THE STRING
 1809 :THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
 1810

1811 :INPUTS:
 1812 :
 1813 : R0 OCTAL VALUE TO CONVERT
 1814 : R1 TABLE OF POINTERS TO ASCII EQUIVALENT
 1815 :
 1816 :-
 1817

1818
 1819 007600 PRIEQU:
 1820 007600 SAVREG ;SAVE THE REGISTERS
 1821 007604 000207 RTS PC ;RETURN TO CALLER
 1822
 1823
 1824

1825 .SBTTL PRIRAM - PRINT RAM ADDRESS
 1826 :+
 1827 :PRINT CONTROLLER RAM ADDRESS.
 1828 :THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
 1829

1830 :INPUTS:
 1831 :
 1832 : R4 RAM ADDRESS
 1833 :
 1834 :-
 1835

1836
 1837 007606 PRIRAM:
 1838 007606 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
 1839 007612 PRINTB #RAMFOR,R4 ;PRINT RAM ADDRESS IN ERROR
 007612 010446 MOV R4,-(SP)
 007614 012746 007636 MOV #RAMFOR,-(SP)
 007620 012746 000002 MOV #2,-(SP)
 007624 010600 MOV SP,R0
 007626 104414 TRAP CSPNTB
 007630 062706 000006 ADD #6,SP
 1840 007634 000207 RTS PC ;RETURN

1841
 1842 007636 045 116 045 RAMFOR: .ASCIZ 'XNZA CONTROLLER RAM ADDRESS = X06'
 1843 .EVEN
 1844

1845 .SBTTL PRIADD - PRINT MEMORY ERROR ADDRESS
 1846 :+
 1847 :PRINT MEMORY ADDRESS
 1848 :THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
 1849

1850 :IMPLICIT INPUTS
 1851 :
 1852 : ERRHI - HIGH ORDER ADDRESS
 1853 : ERRLO - LOW ORDER ADDRESS
 1854 :
 1855

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 29-1
PRIADD - PRINT MEMORY ERROR ADDRESS

```

1856
1857
1858 007700
1859 007700
1860 007704 013700 002202
1861 007710 013701 002204
1862 007714 010102
1863 007716 006101
1864 007720 006100
1865 007722
      007722 010246
      007724 010046
      007726 012746 007750
      007732 012746 000003
      007736 010600
      007740 104414
      007742 062706 000010
1866 007746 000207
      SAVREG
      MOV ERRHI,R0 ;SAVE R1-R5 UNTIL NEXT RETURN
      MOV ERRLO,R1 ;GET HIGH ADDRESS
      MOV R1,R2 ;GET LOW ADDRESS
      ROL R1 ;COPY LOW ADDRESS
      ROL R0 ;SHIFT BIT 15 TO C BIT
      PRINTB #PRIAD,R0,R2 ;SHIFT INTO HIGH ORDER
      MOV R2,-(SP) ;PRINT MEMORY ADDRESS IN ERROR
      MOV R0,-(SP)
      MOV #PRIAD,-(SP)
      MOV #3,-(SP)
      MOV SP,R0
      TRAP CSPNTB
      ADD #10,SP
      RTS PC ;RETURN

```

```

1867
1868 007750 045 116 045 PRIAD: .ASCIZ 'XNZA MEMORY ERROR ADDRESS = X01X05'
1869 .EVEN

```

```

1870
1871 ;+
1872 ;PRINT MEMORY ADDRESS
1873 ;THIS ROUTINE IS NORMALLY CALLED ONLY FROM PRINT ROUTINES.
1874 ; IMPLICIT INPUTS
1875 : ERRHI - HIGH ORDER ADDRESS
1876 : ERRLO - LOW ORDER ADDRESS

```

```

1877 010014
1878 010014
1879 010020 013700 002202
1880 010024 013701 002204
1881 010030 010102
1882 010032 006101
1883 010034 006100
1884 010036
      010036 010246
      010040 010046
      010042 012746 010064
      010046 012746 000003
      010052 010600
      010054 104414
      010056 062706 000010
1885 010062 000207
      SAVREG
      MOV ERRHI,R0 ;SAVE R1-R5 UNTIL NEXT RETURN
      MOV ERRLO,R1 ;GET HIGH ADDRESS
      MOV R1,R2 ;GET LOW ADDRESS
      ROL R1 ;COPY LOW ADDRESS
      ROL R0 ;SHIFT BIT 15 TO C BIT
      PRINTB #PRITD,R0,R2 ;SHIFT INTO HIGH ORDER
      MOV R2,-(SP) ;PRINT MEMORY ADDRESS IN ERROR
      MOV R0,-(SP)
      MOV #PRITD,-(SP)
      MOV #3,-(SP)
      MOV SP,R0
      TRAP CSPNTB
      ADD #10,SP
      RTS PC ;RETURN

```

```

1886 010064 045 116 045 PRITD: .ASCIZ 'XNZA MEMORY TEST ADDRESS = X01X05'
1887 .EVEN

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 30
SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND

```

1889                                     .SBTTL SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND
1890
1891 :ROUTINE TO ISSUE A SPACE RECORDS
1892 :COMMAND (FORWARD OR REVERSE)
1893 :INPUT:
1894 R3      NUMBER OF RECORDS TO BE SPACED OVER
1895         BIT15 CONTROLS DIRECTION
1896         BIT15 = 0 IS FORWARD
1897         BIT15 = 1 IS REVERSE
1898 R5      FIRST DEVICE UNIBUS ADDRESS
1899         REQUIRES A WRITE CHARACTERISTICS DONE PREVIOUSLY
1900 :OUTPUT:
1901 CARRY   SET - SPACE RECORDS COMMAND OK
1902         CLR - SPACE RECORDS FAILED
1903 R0      THE CONTENTS OF R4 IS MOVED TO R0
1904 :IMPLICIT OUTPUT:
1905         TAPE HAS BEEN MOVED
1906 :SIDE EFFECTS:
1907 :-
1908 SPACE::
1909     SAVREG                               ;SAVE THE GENERAL REGISTERS
1910     MOV #500.,SDELAY                     ;SET UP DELAY
1911     MOV #140010,80$                      ;SET UP COMMAND, SPACE FORWARD
1912     TST R3                               ;CHECK FOR DIRECTION
1913     BMI 5$                               ;BR, IF REVERSE INDICATED
1914     MOV R3,90$                           ;LOAD UP NUMBER OF RECORDS TO SPACE
1915     BR 10$                               ;GO DO COMMAND
1916     BIC #BIT15,R3                        ;CLEAR DIRECTION BIT
1917     MOV R3,90$                           ;LOAD UP NUMBER OF RECORDS TO SPACE
1918     BIS #BIT8,80$                        ;SET REVERSE BIT IN COMMAND PACKET
1919     MOV #80$,R4                           ;SET UP R4 WITH PACKET ADDRESS
1920     MOV R4,TSDB(R5)                      ;SEND OUT COMMAND
1921     JSR PC,WAITF                          ;WAIT FOR SSR
1922     BCS 20$                              ;BR, IF SSR IS SET AND OK
1923     DELAY 250                             ;DELAY ABOUT .25 SECONDS
1924     MOV #250,(PC)+
1925     .WORD 0
1926     MOV LSDLY,(PC)+
1927     .WORD 0
1928     DEC -6(PC)
1929     BNE -4
1930     DEC -22(PC)
1931     BNE -20
1932     DEC SDELAY                           ;BUMP DELAY COUNTER DOWN
1933     BNE 15$                              ;BR, IF MORE DELAY
1934     BR 60$                               ;BR IF TROUBLE CARRY = CLEAR
1935     MOV TSSR(R5),R1                      ;READ TSSR
1936     MOV #SSR,R2                          ;SET UP EXPECTED
1937     CMP R2,R1                            ;ARE THEY OK
1938     BEQ 40$                              ;BR, IF EQUAL = OK
1939     BR 60$                               ;TROUBLE EXIT
1940     SEC                                  ;SET CARRY NO TROUBLE
1941     BR 70$                               ;EXIT
1942     CLC                                  ;CARRY CLEAR = ERROR
1943     MOV R4,R0                             ;PASS PACKET ADDRESS
1944     RTS PC                                ;RETURN

```

CZTUXAD TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 30-1
SPACE - SPACE RECORDS (FORWARD AND REVERSE) COMMAND

1938			:			
1939			:	PACKET FOR SPACE COMMAND		
1940			:			
1942	010304		:	.BLKB	10-<.-TUV2A&7>	
1944			:			
1945			:	COMMAND WORD		
1946	010310	000000	80\$:	.WORD		
1947			:	NUMBER OF RECORDS TO BE SPACED OVER WORD		
1948	010312	000000	90\$:	.WORD		
1949	010314	000000		.WORD		
1950	010316	000000		.WORD		
1951	010320	000000	SDELAY:	.WORD	0	:DELAY COUNTER
1952				.EVEN		

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 31
 WRTCHR - WRITE CHARACTERISTICS COMMAND

.SBTTL WRTCHR - WRITE CHARACTERISTICS COMMAND

1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009

010322
010322 005037 002174
010326 010465 177776
010336 004737 017220
010342 103401
010344 000423
010346 016501 000000
010352 012702 000200
010356 032701 000100
010362 001402
010364 052702 000100
010370 020201
010372 001401
010374 000407
010376 062704 000010
010402 011403
010404 010337 002716
010410 000261
010412 G00401
010414 000241
010416 016500 000000
010422 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:
:   BENBSW = BUFFER ENABLE SWITCH ON OR OFF
:SIDE EFFECTS:
:-
    
```

```

WRTCHR::
:SAVE THE GENERAL REGISTERS
:SAVE THE GENERAL REGISTERS
:CLR BENBSW :CLEAR BUFFER ENABLE SWITCH
: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
: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$: ADD #8.,R4 :POINT TO WRT CHARA DATA PACKET
:MOV (R4),R3 :GET ADDRESS OF MESSAGE BUFFER
:MOV R3,MESBFA :STORE FOR PRINT ROUTINES
:SEC :SET CARRY NO TROUBLE
:BR 70$ :EXIT
:60$: CLC :CARRY CLEAR = ERROR
:70$: MOV TSSR(R5),R0 :RETURN TSSR CONTENTS
:RTS PC :RETURN
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 32
 REWIND - POSITION TAPE (REWIND) COMMAND

.SBTTL REWIND - POSITION TAPE (REWIND) COMMAND

2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039 010424
2040 010424
2041 010430 012704 010520
2042 010434 010465 177776
2043 010440 012703 000550
2044 010444 004737 017104
2045 010450 103417
2046 010452
010452 012727 000372
010456 000000
010460 013727 002116
010464 000000
010466 005367 177772
010472 001375
010474 005367 177756
010500 001367
2047 010502 005303
2048 010504 001357
2049 010506 000241
2050 010510 010400
2051 010512 000207
2053 010514
2055 010520
2056 010520 102010
2057 010522 000000

```

:~
:~
:~ THIS ROUTINE WILL REWIND THE SELECTED TAPE.
:~
:~ CAUTION: THE ROUTINE DOES NOT WAIT FOR BOT
:~ TO ARRIVE. ALSO THE CALLER MUST CHECK FOR
:~ SSR TO SET IN THE TSSR
:~
:~ CALLING SEQUENCE:
:~
:~ DO A SOFT INIT
:~ DO A WRITE CHARACTERISTICS
:~ JSR PC,REWIND
:~
:~ INPUT:
:~
:~ R5 FIRST DEVICE UNIBUS ADDRESS
:~
:~ OUTPUT
:~
:~ R0 THE CONTENTS OF R4 IS PASSED TO R0
:~
:~ REWIND::
:~ SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
:~ MOV #RWPACK,R4 ;GET PACKET ADDRESS
:~ MOV R4,TSDB(R5) ;SEND PACKET ADDRESS TO EXECUTE
:~ MOV #360.,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND
10$: JSR PC,WAITF ;WAIT FOR SSR TO SET
:~ BCS 20$ ;LEAVE WHEN SSR IS SET
:~ DELAY 250. ;WAIT FOR .25 SECONDS
:~ MOV #250.,(PC)+
:~ .WORD 0
:~ MOV L$DLY,(PC)+
:~ .WORD 0
:~ DEC -6(PC)
:~ BNE -.4
:~ DEC -22(PC)
:~ BNE .-20
:~ DEC R3 ;BUMP COUNTER DOWN
:~ BNE 10$ ;KEEP GOING
:~ CLC ;CLEAR CARRY TO SET ERROR
20$: MOV R4,R0 ;PASS THE PACKET ADDRESS
:~ RTS PC ;RETURN
:~ .BLKB 10-<.-TUV2A&7>
RWPACK: .WORD 102010 ;POSITION COMMAND (REWIND)
:~ .WORD 0 ;NOT USED
    
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 33
 CKRAM - COMPARE RAM TO I/O PACKET

2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087 010524
2088 010524
2089 010530 012701 002206
2090 010534 012702 000020
2091 010540 005003
2092 010542 004737 017220
2093 010546 004737 017220
2094 010552 110265 177777
2095 010556 004737 017220
2096 010562 116511 177776
2097 010566 122124
2098 010570 001401
2099 010572 005203
2100 010574 005202
2101 010576 020227 000027
2102 010602 003761
2103 010604 005703
2104 010606 001402
2105 010610 000241
2106 010612 000401
2107 010614 000261
2108 010616 012737 000010 002246
2109 010624 000207
2110

.SBTTL CKRAM - COMPARE RAM TO I/O PACKET

```

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

```

```

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

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 34
 RAMER - READ AND DISPLAY SELECTED RAM

2112
 2113
 2114
 2115
 2116
 2117
 2118
 2119
 2120
 2121
 2122
 2123
 2124
 2125
 2126
 2127
 2128
 2129
 2130
 2131

.SBTTL RAMER - READ AND DISPLAY SELECTED RAM

```

:ROUTINE TO READ THE SELECTED RAM LOCATIONS
:INPUT:
:
:   R5      FIRST DEVICE UNIBUS ADDRESS
:   CONSOLE WILL ALSO BE PRINTED TO
:
:IMPLICIT OUTPUT:
:
:   THE TABLE RAMDATA IS FILLED WITH THE
:   DATA HELD IN RAM.
:
:SIDE EFFECTS:
:
:-
    
```

2132 010626
 2133 010626
 2134 010632 013705 011012
 2135 010636 012701 002206
 2136 010642 013702 011010
 2137 010646 013703 002246
 2138 010652 004737 017220
 2139 010656 110265 177777
 2140 010662 004737 017220
 2141 010666 116521 177776
 2142 010672 062702 000001
 2143 010676 077313
 2144 010700 013704 002246
 2145 010704 013702 011010
 2146 010710 060204
 2147 010712 162704 000001
 2148 010716
 010716 010446
 010720 010246
 010722 012746 011014
 010726 012746 000003
 010732 010600
 010734 104415
 010736 062706 000010
 2149 010742 012701 002206
 2150 010746 013703 002246
 2151 010752 005004
 2152 010754 112104
 2153 010756 042704 177400
 2154 010762
 010762 010446
 010764 012746 011065
 010770 012746 000002
 010774 010600
 010776 104415
 011000 062706 000006
 2155 011004 077316

```

RAMER::
      SAVREG
      MOV      RAMR5H,R5      ;SAVE THE GENERAL REGISTERS
      MOV      #RAMDATA,R1   ;RESET R5 TO FIRST DEVICE REGISTER
      MOV      RAMHLD,R2     ;ADDRESS TO SAVE THE RAM DATA
      MOV      RAMSIZ,R3     ;BYTE ADDRESS OF THE FIRST RAM DATA
      MOV      RAMSIZ,R3     ;SET THE SIZE OF THE READ UP
10$:   JSR      PC,CHKTSSR    ;WAIT FOR THE SSR TO SET
      MOV      R2,TSDBH(R5)  ;SELECT NEXT RAM ADDRESS
      JSR      PC,CHKTSSR    ;WAIT FOR SSR TO SET
      MOV      TSBAL(R5),(R1)+ ;READ THE RAM DATA
20$:   ADD      #1,R2        ;ADDRESS OF THE NEXT RAM LOCATION
      SOB      R3,10$       ;NUMBER OF LOCATIONS COUNTER
      MOV      RAMSIZ,R4     ;GET THE RAM SIZE
      MOV      RAMHLD,R2     ;GET THE STARTING RAM ADDRESS
      ADD      R2,R4        ;CALCULATE THE END ADDRESS
      SUB      #1,R4        ;CORRECT VALUE OF PRINTOUT
      PRINTX  #RAMIOP,R2,R4 ;RAM ADDRESS = 10 - 17, ETC.
      MOV      R4,-(SP)
      MOV      R2,-(SP)
      MOV      #RAMIOP,-(SP)
      MOV      #3,-(SP)
      MOV      SP,R0
      TRAP    C$PNTX
      ADD      #10,SP
      MOV      #RAMDATA,R1   ;ADDRESS OF WHERE RAM DATA IS
      MOV      RAMSIZ,R3     ;THE SIZE OF THE RAM FIELD READ
30$:   CLR      R4          ;NO EXTRA DATA LEFT OVER
      MOV      (R1)+,R4      ;PICK UP BYTE OF RAM DATA
      BIC      #177400,R4    ;GET RID OF SIGN EXTEND
      PRINTX  #RAMPD,R4     ;'010 211 111 222 377 000 123 134 ETC.'"
      MOV      R4,-(SP)
      MOV      #RAMPD,-(SP)
      MOV      #2,-(SP)
      MOV      SP,R0
      TRAP    C$PNTX
      ADD      #6,SP
      SOB      R3,30$       ;LOOP UNTIL ALL PRINTED
    
```


CZTUXAG TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 35
CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA

2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215

011100
011100
011104 012701 002206
011110 012702 000167
011114 005003
011116 004737 017220
011122 004737 017220
011126 110265 177777
011132 004737 017220
011136 116511 177776
011142 122124
011144 001401
011146 005203
011150 005202
011152 012737 000010 002246
011160 020227 000176
011164 003756
011166 005703
011170 001402
011172 000241
011174 000401
011176 000261
011200 000207

```

.SBTTL CKRAM2 - COMPARE RAM TO I/O CHARACTERISTICS DATA
:ROUTINE TO READ THE FIRST 8 OR 10 BYTES FROM RAM
:MEMORY AND COMPARE THIS DATA TO A CHARACTERISTICS DATA BLOCK.
:INPUT:
R4 ADDRESS OF THE CHARACTERISTICS DATA
R5 FIRST DEVICE UNIBUS ADDRESS
:OUTPUT:
CARRY SET - RAM MATCHES PACKET
CLR - RAM DOES NOT MATCH PACKET
:IMPLICIT OUTPUT:
THE TABLE RAMDATA IS FILLED WITH THE
DATA HELD IN RAM.
RAMSIZ IS SET TO 8. OR 10. FOR PRAMPKT ROUTINE
:SIDE EFFECTS:
-
CKRAM2::
SAVREG ;SAVE THE GENERAL REGISTERS
MOV #RAMDATA,R1 ;ADDRESS TO SAVE THE RAM DATA
MOV #RMCHBEG,R2 ;BYTE ADDRESS OF FIRST RAM DATA
CLR R3 ;CLEAR THE ERROR FLAG
JSR PC,CHKTSSR ;WAIT FOR SSR
10$: JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
MOVB R2,TSDBH(R5) ;SELECT NEXT RAM ADDRESS
JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
MOVB TSBAL(R5),(R1) ;READ THE RAM DATA
CMPB (R1)+,(R4)+ ;COMPARE TO EXPECTED
BEQ 20$ ;BRANCH IF OK
INC R3 ;SET ERROR FLAG
20$: INC R2 ;ADDRESS OF NEXT RAM LOCATION
MOV #8,RAMSIZ ;ASSUME NORMAL NOT SET
CMP R2,#RMCHEND-2 ;REACHED END YET ?
BLE 10$ ;BRANCH TILL ALL READ
27$: TST R3 ;WAS AN ERROR FOUND ?
BEQ 30$ ;BRANCH IF NOT
CLC ;CLEAR CARRY TO SHOW ERROR
BR 50$ ;AND EXIT
30$: SEC ;SHOW GOOD COMPARE
50$: RTS PC ;RETURN

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 36
 CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS

2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271

011202
011202
011206 010037 002250
011212 010137 002252
011216 005737 003102
011222 001403
011224 004737 020252
011230 010001
011232 005004
011234 005003
011236 010205
011240 011264 002266
011244 011164 002432
011250 022221
011252 001401
011254 005203
011256 062704 000002
011262 020427 000014
011266 003764
011270 032765 000200 000012
011276 001403
011300 020427 000016
011304 003755
011306 005703
011310 001402
011312 000241
011314 000401
011316 000261
011320 000207

```

.SBTTL CKMSG - COMPARE WRITE CHAR. MESSAGE BUFFERS
+
:ROUTINE TO COMPARE A WRITE CHARACTERISTICS EXPD AND RECV
:BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
:ERROR PRINT ROUTINES.
:INPUT:
:      R0      RECV MESSAGE BUFFER HIGH ORDER ADDRESS
:      R1      RECV MESSAGE BUFFER LOW ORDER ADDRESS
:      R2      EXPD MESSAGE BUFFER ADDRESS
:OUTPUT:
:      CARRY   SET - MESSAGE BUFFERS MATCH
:             CLR -MESSAGE BUFFERS DON'T MATCH
:IMPLICIT OUTPUT:
:      EXPMSG  BUFFER IS SET TO EXPD DATA
:      RECVMSG BUFFER IS SET TO RECV DATA
:      RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
:      RCVLOADD SET TO LOW ORDER ADDRESS OF RECV
-
CKMSG::
SAVREG          :SAVE R1-R5 UNTIL NEXT RETURN
MOV R0,RCVHIADD :SAVE RECV HIGH ADDRESS
MOV R1,RCVLOAD  :SAVE RECV LOW ADDRESS
TST KENABLE    :TESTING ABOVE 28K?
BEQ 10$        :BR IF NO
JSR PC,SETMAP  :RETURN ADDRESS BIASED TO PAR6 IN R0
MOV R0,R1      :GET RETURNED ADDRESS BIASED TO PAR6
10$: CLR R4     :WORD IN BUFFER
     CLR R3     :CLEAR ERROR SEEN FLAG
     MOV R2,R5  :GET EXPD BUFFER ADDRESS
15$: MOV (R2),EXPMSG(R4) :SAVE EXPD FOR ERROR REPORT
     MOV (R1),RECVMSG(R4) :SAVE RECV FOR ERROR REPORT
     CMP (R2)+,(R1)+    :EXPD EQUAL RECV?
     BEQ 25$        :BR IF YES
     INC R3        :SET ERROR SEEN FLAG
25$: ADD #2,R4     :POINT TO NEXT WORD ADDRESS
     CMP R4,#14    :DONE FIRST 7 WORDS?
     BLE 15$       :BR IF NO
     BIT #X2.EXTF,XST2(R5) :IS EXTENDED FEATURES SET IN EXPD?
     BEQ 50$       :BR IF NO
     CMP R4,#16    :DONE EXTENDED FEATURES WORD?
     BLE 15$       :BR IF NO
50$: TST R3       :ANY ERRORS SEEN?
     BEQ 55$       :BR IF NO
     CLC          :SET FAILURE
     BR 60$       :
55$: SEC          :SET SUCCESS
60$: RTS PC      :RETURN
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 37
 CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2273 .SBTTL CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS
2274
2275
2276 :ROUTINE TO COMPARE AN EXPECTED AND RECEIVED MESSAGE
2277 :BUFFER. THE EXPECTED AND RECEIVED BUFFERS ARE STORED FOR
2278 :ERROR PRINT ROUTINES.
2279
2280 :INPUT:
2281
2282 R0 RECV MESSAGE BUFFER HIGH ORDER ADDRESS
2283 R1 RECV MESSAGE BUFFER LOW ORDER ADDRESS
2284 R2 EXPD MESSAGE BUFFER ADDRESS
2285 R3 NUMBER OF BYTES TO COMPARE
2286
2287 :OUTPUT:
2288
2289 CARRY SET - MESSAGE BUFFERS MATCH
2290 CLR - MESSAGE BUFFERS DON'T MATCH
2291
2292 :IMPLICIT OUTPUT:
2293
2294 EXPMSG BUFFER IS SET TO EXPD DATA
2295 RECMSG BUFFER IS SET TO RECV DATA
2296 RCVHIADD SET TO HIGH ORDER ADDRESS OF RECV
2297 RCVLOADD SET TO LOW ORDER ADDRESS OF RECV
2298
2299
    
```

```

2300 011322 CKMSG2::
2301 011322 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2302 011326 020327 000144 CMP R3,#RECMSG-EXPMSG;@D IS COUNT ABOVE MAX ALLOWED?
2303 011332 003412 BLE 5$ ;@D BR IF NO
2304 011334 012703 000144 MOV #RECMSG-EXPMSG,R3;@D
2305 011340 PRINTF #DEBUGMSG ;@D
    011340 012746 011454 MOV #DEBUGMSG,-(SP)
    011344 012746 000001 MOV #1,-(SP)
    011350 010600 MOV SP,R0
    011352 104417 TRAP CSPNTF
    011354 062706 000004 ADD #4,SP
2306 011360 010037 002250 5$: MOV R0,RCVHIADD ;SAVE RECV HIGH ADDRESS
2307 011364 010137 002252 MOV R1,RCVLOAD ;SAVE RECV LOW ADDRESS
2308 011370 005737 003102 TST KENABLE ;TESTING ABOVE 28K?
2309 011374 001403 BEQ 10$ ;BR IF NO
2310 011376 004737 020252 JSR PC,SETMAP ;RETURN ADDRESS BIASED TO PAR6 IN R0
2311 011402 010001 MOV R0,# ;GET RETURNED ADDRESS BIASED TO PAR6
2312 011404 005004 10$: CLR R4 ;WORD IN BUFFER
2313 011406 005005 CLR R5 ;CLEAR ERROR SEEN FLAG
2314 011410 111264 002266 15$: MOVB (R2),EXPMSG(R4) ;SAVE EXPD FOR ERROR REPORT
2315 011414 111164 002432 MOVB (R1),RECMSG(R4) ;SAVE RECV FOR ERROR REPORT
2316 011420 122221 CMPB (R2)+,(R1)+ ;EXPD EQUAL RECV?
2317 011422 001401 BEQ 25$ ;BR IF YES
2318 011424 005205 INC R5 ;SET ERROR SEEN FLAG
2319 011426 062704 000001 25$: ADD #1,R4 ;POINT TO NEXT BYTE
2320 011432 02C4G3 CMP R4,R3 ;DONE ALL BYTES?
2321 011434 00200? BGE 50$ ;BR IF YES
2322 011436 000764 BR 15$ ;DO NEXT BYTE
2323 011440 005705 50$: TST R5 ;ANY ERRORS SEEN?
2324 011442 001402 BEQ 55$ ;BR IF NO
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 37-1
CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2325 011444 000241          CLC          ;SET FAILURE
2326 011446 000401          BR          60$          ;
2327 011450 000261          55$: SEC          ;SET SUCCESS
2328 011452 000207          60$: RTS          P^          ;RETURN
2329
2330 011454 120 122 117 DEBUGMSG: .ASCIZ 'PROGRAM INTERNAL ERROR -CKMSG2 MESSAGE BUFFER EXCEEDED-';000
2331 011544 045 116 045 FERCM: .ASCII /XNZA ***/
2332 011555 040 040 124 ERCM: .ASCIZ / TSSR ERROR CODE REC'D = /
2333 011610 056 056 056 SIMSG: .ASCIZ /... AFTER DOING SOFT INIT/
2334 011643 124 105 123 TINERR: .ASCIZ /TEST: .../
2335 .EVEN

```

CZTUXAG TUBO FRONT END PRT @ MACRO M1200 27-MAR-83 13:32 PAGE 38
CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387

011656
011656
011656 004737 005264
011662 004737 020136
011666
011666
011666 104423
011670
011670
011670 004737 005264
011674 012700 000004
011700 004737 007060
011704 013700 002716
011710 005001
011712 004737 014052
011716
011716 104423
011716 104423

:+
:PRINT ROUTINE TO FATAL SOFT INIT ERRORS
:INPUT:
:R1 CONTENTS OF TSSR AT ERROR
:SIDE EFFECTS:
:EXECUTES DROP UNIT TO CEASE TESTING
:-
BGNMSG SFMSG
SFMSG::
JSR PC,PRITSSR ;PRINT CONTENTS OF TSSR REGISTER
JSR PC,CKDROP ;DROP UNIT, IF ALLOWED
ENDMSG
L10003:
TRAP CSMSG

:+
:PRINT ROUTINE TO PRINT THE CONTENTS OF
:TSSR AND A COMMAND PACKET OTHER THAN GET STATUS COMMAND PACKET.
:INPUTS:
:R1 TSSR CONTENTS
:R4 ADDRESS OF COMMAND PACKET
:-

BGNMSG PKTSSR
PKTSSR::
JSR PC,PRITSSR ;PRINT THE CONTENTS OF TSSR REGISTER
MOV #4,R0 ;NO. OF WORDS IN PACKET
JSR PC,PRIPKT ;PRINT THE CONTENTS OF COMMAND PACKET
MOV MESBFA,R0 ;ADDRESS OF MESSAGE BUFFER
CLR R1 ;ASSUME NO HIGH MEMORY
JSR PC,PMESS ;PRINT THE MESSAGE BUFFER ALSO
ENDMSG
L10004:
TRAP CSMSG

:+
:PRINT ROUTINE TO PRINT THE CONTENTS OF
:TSSR AND A GET STATUS COMMAND PACKET.
:INPUTS:
:R1 TSSR CONTENTS
:R4 ADDRESS OF COMMAND PACKET
:-

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 38-1
 CKMSG2 - COMPARE EXPD RECV MESSAGE BUFFERS

```

2388
2389 011720          BGNMSG  PKTGETS
      011720          PKTGETS::
2390 011720 004737 005264      JSR      PC,PRITSSR      ;PRINT THE CONTENTS OF TSSR REGISTER
2391 011724 012700 000002      MOV      #2,R0          ;NO. OF WORDS IN GET STATUS PACKET
2392 011730 004737 007060      JSR      PC,PRIPKT      ;PRINT THE CONTENTS OF COMMAND PACKET
2393 011734          ENDMSG
      011734          L10005:
      011734 104423      TRAP      CSMSG

2394
2395
2396          ;+
2397          ;PRINT TSSR ERRORS FOR INITIALIZATION TESTS
2398
2399          ;INPUTS:
2400          ;
2401          ;      R1      TSSR CONTENTS
2402          ;      R4      ADDRESS OF COMMAND PACKET
2403          ;-
2404
2405 011736          BGNMSG  SFFMSG
      011736          SFFMSG::
2406 011736 004737 005264      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR REGISTER
2407 011742          ENDMSG
      011742          L10006:
      011742 104423      TRAP      CSMSG

2408
2409
2410          .SBTTL  PKTMES - PRINT TSSR AND MESSAGE BUFFER
2411          ;+
2412          ;
2413          ;PRINT ROUTINE TO PRINT THE CONTENTS OF TSSR AND MESSAGE
2414          ;BUFFER FOR ERROR REPORTS
2415
2416          ;INPUTS:
2417          ;
2418          ;      R1      CONTENTS OF TSSR
2419          ;      R2      LOW ORDER MESSAGE BUFFER
2420          ;      R3      HIGH ORDER MESSAGE BUFFER ADDRESS
2421          ;      NOTE: R3 IS IGNORED IF KTENABLE FLAG IS CLEAR
2422          ;-
2423 011744          BGNMSG  PKTMES
      011744          PKTMES::
2424 011744 004737 005264      JSR      PC,PRITSSR      ;PRINT CONTENTS OF TSSR
2425 011750 010200          MOV      R2,R0          ;LOW ORDER ADDRESS
2426 011752 010301          MOV      R3,R1          ;HIGH ORDER ADDRESS
2427 011754 004737 014052      JSR      PC,PRMESS      ;PRINT THE MESSAGE BUFFER
2428 011760          ENDMSG
      011760          L10007:
      011760 104423      TRAP      CSMSG

2429
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 39
ADDSSR - PRINT TEST ADDRESS AND TSSR

2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443 011762
011762
2444 011762 004737 010014
2445 011766 016501 000000
2446 011772 004737 005264
2447 011776
011776
011776 104423
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462 012000
012000
2463 012000 012700 000007
2464 012004 004737 015406
2465 012010
012010
012010 104423
2466
2467

```
.SBTTL ADDSSR - PRINT TEST ADDRESS AND TSSR
:
:PRINT ROUTINE TO PRINT THE CONTENTS OF
:TSSR AND A MEMORY TEST ADDRESS
:
:INPUTS:
:
:      R5      FIRST DEVICE UNIBUS ADDRESS
:      ERRHI   HIGH ORDER MEMORY TEST ADDRESS
:      ERRLO   LOW ORDER MEMORY TEST ADDRESS
:
:
:      BGNMSG  ADDSSR
ADDSSR::
:      JSR     PC,PRITADD      ;PRINT MEMORY TEST ADDRESS
:      MOV     TSSR(R5),R1     ;GET CURRENT TSSR
:      JSR     PC,PRITSSR     ;PRINT THE CONTENTS OF TSSR REGISTER
:      ENDMSG
L10010:
:      TRAP    C$MSG
```

```
.SBTTL MSGEXP - PRINT WRITE CHAR. EXPD-RCV MESSAGE BUFFERS
:
:PRINT ROUTINE TO PRINT WRITE CHARACTERISTIC MESSAGE BUFFER
:
:IMPLICIT INPUTS:
:
:      EXPMSG  - EXPECTED MESSAGE BUFFER
:      RECMSG  - RECEIVED MESSAGE BUFFER
:      RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
:      RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
:
:
:      BGNMSG  MSGEXP
MSGEXP::
:      MOV     #7,R0          ;ASSUME NO EXT FEATURES
:      JSR     PC,PRMSGEXP   ;PRINT EXPD/RCV MESSAGE BUFFERS
:      ENDMSG
L10011:
:      TRAP    C$MSG
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 40
FIFEXP - PRINT FIFO EXP/RECV DATA

```

2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481 012012
      012012
2482 012012
      012012 010146
      012014 012746 012064
      012020 012746 000002
      012024 010600
      012026 104415
      012030 062706 000006
2483 012034
      012034 012746 012133
      012040 012746 000001
      012044 010600
      012046 104415
      012050 062706 000004
2484 012054 010100
2485 012056 004737 015756
2486 012062
      012062
      012062 104423
2487 012064 045 116
2488 012133 045 116
2489
2490

```

```

.SBTTL FIFEXP - PRINT FIFO EXP/REC V DATA
:
:PRINT ROUTINE TO PRINT FIFO EXP/REC V DATA
:
:      R1      - BYTE COUNT
:
:IMPLICIT INPUTS:
:
:      EXPMSG - EXPECTED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
:      RECMSG - RECEIVED MESSAGE BUFFER (CONTAINS FIFO DATA ONLY)
:
:      BGNMSG FIFEXP
FIFEXP::
:PRINTX #FIF1MSG,R1      ;PRINT BYTES TRANSFERRED
:MOV R1,-(SP)
:MOV #FIF1MSG,-(SP)
:MOV #2,-(SP)
:MOV SP,R0
:TRAP C$PNTX
:ADD #6,SP
:PRINTX #FIF2MSG      ;PRINT HEADER MSG
:MOV #FIF2MSG,-(SP)
:MOV #1,-(SP)
:MOV SP,R0
:TRAP C$PNTX
:ADD #4,SP
:MOV R1,R0      ;GET BYTE COUNT
:JSR PC,PRBYTEXP ;PRINT FIFO BYTES IN ERROR
:ENDMSG
L10012:
:TRAP C$MSG
:ASCIZ 'XNXA NUMBER OF BYTES TRANSFERRED = X02'
:ASCIZ 'XNXA FIFO DATA BYTES IN ERROR:'
.EVEN

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 41
MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS

2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505 012172
012172
2506 012172 012701 012234
2507 012176 012100
2508 012200 001410
2509 012202
012202 010046
012204 012746 000001
012210 010600
012212 104415
012214 062706 000004
2510 012220 000766
2511 012222 012700 000012
2512 012226 004737 015406
2513 012232
012232
012232 104423
2514
2515 012234 012252 012314 012405
2516 012252 045 116 045
2517 012314 045 116 045
2518 012405 045 116 045
2519 012476 045 116 045
2520 012567 045 116 045
2521 012631 045 116 045
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538 012706
012706
2539 012706 012701 012750

```

.SBTTL MSGSTAT - PRINT STATUS HEADER AND MESSAGE BUFFERS
:
:PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
:
:IMPLICIT INPUTS:
:
:EXPMSG - EXPECTED MESSAGE BUFFER
:RECMSG - RECEIVED MESSAGE BUFFER
:RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
:RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
:
:
:BGMSG MSGSTAT
MSGSTAT::
10$: MOV #STATCOD,R1 ;ASCII ADDRESS TABLE
MOV (R1)+,R0 ;DONE ALL MSG LINES?
BEQ 20$ ;BR IF YES
PRINTX R0 ;PRINT STATUS BIT NAMES
MOV R0,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP
BR 10$ ;DO ANOTHER MSG LINE
20$: MOV #10.,R0 ;NUMBER OF WORDS IN A READ STATUS BUFFER
JSR PC,PRMSGEXP ;PRINT EXPD/RCV MESSAGE BUFFERS
ENDMSG
L10013: TRAP C$MSG
STATCOD: .WORD 1$,2$,3$,4$,5$,6$,0
1$: .ASCIZ 'XNZA Tape Bus Signals in Word #8:'
2$: .ASCIZ 'XNZA PARERR<15> IEOT <12> IFMK <9> IRDY<6> IRWD<2>'
3$: .ASCIZ 'XNZA IRESV2<14> IIDENT<11> IHER <8> IONL<5> IFBY<1>'
4$: .ASCIZ 'XNZA IRESV1<13> ICER <10> ISPEED<7> ILDP<4> IFPT<0>'
5$: .ASCIZ 'XNZA Tape Bus Signals in Word #9:'
6$: .ASCIZ 'XNZA DATMIS<7> ILW<6> OUTRDY<5> INRDY<4>'
.EVEN

```

```

.SBTTL MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS
:
:PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
:
:IMPLICIT INPUTS:
:
:EXPMSG - EXPECTED MESSAGE BUFFER
:RECMSG - RECEIVED MESSAGE BUFFER
:RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
:RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
:
:
:BGMSG MSGLOOP
MSGLOOP::
MOV #LOOPCOD,R1 ;ASCII ADDRESS TABLE

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 41-1
MSGLOOP - PRINT LOOPBACK HEADER AND MESSAGE BUFFERS

```

2540 012712 012100          10$:  MOV      (R1)+,R0          ;DONE ALL MSG LINES?
2541 012714 001410          BEQ      20$          ;BR IF YES
2542 012716          PRINTX  R0          ;PRINT STATUS BIT NAMES
      012716 010046          MOV      R0,-(SP)
      012720 012746 000001  MOV      #1,-(SP)
      012724 010600          MOV      SP,R0
      012726 104415          TRAP    C$PNTX
      012730 062706 000004  ADD      #4,SP
2543 012734 000766          BR       10$          ;DO ANOTHER MSG LINE
2544 012736 012700 000012  20$:  MOV      #10,R0        ;NUMBER OF WORDS IN A READ STATUS BUFFER
2545 012742 004737 015406  JSR     PC,PRMSGEXP   ;PRINT EXPD/RECV MESSAGE BUFFERS
2546 012746          ENDMMSG
      012746          L10014:
      012746 104423          TRAP    C$MSG
2547
2548 012750 012770 013043 013142 LOOPCOD: .WORD 1$,2$,3$,4$,5$,6$,7$,0
2549 012770          045 116 045 1$: .ASCIZ 'XNZA Tape Bus Loopback Signals in Word #8:'
2550 013043          045 116 045 2$: .ASCIZ 'XNZA PARERR<15> IRESV2<14> IRESV1<13>'
2551 013142          045 116 045 3$: .ASCIZ 'XNZA IHISP=>IEOT<12> IWRT=>IIDENT<11> IREV =>ICER <10>'
2552 013241          045 116 045 4$: .ASCIZ 'XNZA IWFM =>IFMK<09> IEDIT=>IHER <08> IFAD =>ISPEED<07>'
2553 013340          045 116 045 5$: .ASCIZ 'XNZA ITADO=>IRDY<06> ITAD1=>IONL <05> IERASE=>ILDLP <04>'
2554 013437          045 116 045 6$: .ASCIZ 'XNZA IREW =>IDBY<03> IRWU =>IRWD <02> IFEN =>IFBY <01>'
2555 013536          045 116 045 7$: .ASCIZ 'XNZA IGO =>IFPT<00>'
2556          .EVEN
2557

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 42
 MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER

```

2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572 013564
      013564
2573 013564 012700 000012
2574 013570 004737 015406
2575 013574
      013574
      013574 104423

2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593 013576
      013576
2594 013576 004737 007700
2595 013602 013701 002176
2596 013606 013702 002200
2597 013612 004737 007462
2598 013616
      013616
      013616 104423

2599
    
```

```

.SBTTL MSGSUB - PRINT WRITE SUBSYSTEM MESSAGE BUFFER
:
:PRINT ROUTINE TO PRINT MESSAGE BUFFER EXPD/RCV
:
:IMPLICIT INPUTS:
:
:   EXPMSG - EXPECTED MESSAGE BUFFER
:   RECMG  - RECEIVED MESSAGE BUFFER
:   RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
:   RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
:
:
:   BGNMSG  MSGSUB
MSGSUB::
:   MOV     #10.,R0      ;SIZE OF WRITE SUBSYSTEM BUFFER
:   JSR     PC,PRMSGEXP ;PRINT EXPD/RCV MESSAGE BUFFERS
:   ENDMSG
L10015:
:   TRAP    CSMSG
    
```

```

.SBTTL MEMADD - PRINT MEMORY ADDRESS DATA ERROR
:
:PRINT ROUTINE TO PRINT MEMORY ADDRESS DATA COMPARE ERROR
:
:IMPLICIT INPUTS:
:
:   ERRHI  - MEMORY ERROR HIGH ORDER ADDRESS
:   ERRLO  - MEMORY ERROR LOW ORDER ADDRESS
:   EXP    - EXPECTED DATA
:   RECV   - RECEIVED DATA
:
:
:   BGNMSG  MEMADD
MEMADD::
:   JSR     PC,PRIADD   ;PRINT MEMORY ADDRESS IN ERROR
:   MOV     EXPD,R1     ;GET EXPD DATA
:   MOV     RECV,R2     ;GET RECEIVED DATA
:   JSR     PC,PRIXOR   ;PRINT EXPD/RCV
:   ENDMSG
L10016:
:   TRAP    CSMSG
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 43
 PRAMPKT - PRINT RAM AND PACKET DATA

```

2601 .SBTTL PRAMPKT - PRINT RAM AND PACKET DATA
2602
2603
2604 :PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
2605 :WHEN THE RAM DATA DOES NOT MATCH.
2606
2607 :INPUTS:
2608
2609 : R4 POINTER TO COMMAND PACKET
2610
2611 :IMPLICIT INPUTS:
2612
2613 : RAMDATA DATA AS READ FROM THE RAM
2614 : RAMSIZ NUMBER OF BYTES IN PACKET
2615 : IF RAMSIZ=0 THEN DEFAULT TO 8.
2616
2617 :IMPLICIT OUTPUTS:
2618
2619 : RAMSIZ SET TO 0
2620
2621 :-
2622 PRAMPKT:
2623 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2624 MOV #RAMDATA,R1 ;DATA FROM THE RAM
2625 CLR R2 ;INIT BYTE NUMBER
2626 5$: CMPB (R1)+,(R4)+ ;COMPARE EXPECTED, RECEIVED
2627 BNE 7$ ;BR IF NO MATCH
2628 7$: MOVB -1(R1),R5 ;GET RECV RAM DATA
2629 MOVB -1(R4),R3 ;GET EXPD PACKET DATA
2630 XOR R5,R3 ;XOR EXPD/RECV
2631 BIC #177400,R3 ;LOW BYTE ONLY
2632 MOVB -1(R1),RECV ;GET RECEIVED RAM DATA
2633 MOVB -1(R4),EXPD ;GET EXPECTED RAM DATA
2634 PRINTB #RAMASC,R2,RECV,EXPD,R3
      MOV R3,-(SP)
      MOV EXPD,-(SP)
      MOV RECV,-(SP)
      MOV R2,-(SP)
      MOV #RAMASC,-(SP)
      MOV #5,-(SP)
      MOV SP,R0
      TRAP C$PNTB
      ADD #14,SP
2635 10$: INC R2 ;UPDATE BYTE COUNT
2636 TST RAMSIZ ;DEFAULT TO 8.?
2637 BEQ 15$ ;BR IF YES
2638 CMP R2,RAMSIZ ;DONE ALL BYTES?
2639 BLE 5$ ;BR IF NO
2640 BR 25$
2641 15$: CMP R2,#8. ;DONE DEFAULT NUMBER OF BYTES?
2642 20$: BLT 5$ ;BR IF NO
2643 25$: CLR RAMSIZ ;SET DEFAULT RAMSIZ
2644 RTS PC ;RETURN
2645
2646 045 116 045 RAMASC: .ASCIZ 'XNZX BYTE: XD2XA RAM: X03XA Packet: X03XA XOR:X03'
2647 .EVEN
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 44
 PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

```

2649 .SBTTL PRMESS - PRINT CONTENTS OF MESSAGE BUFFER
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666 014052
2667 014052
2668 014056 010537 011012
2669 014062 010005
2670 014064 005737 003102
2671 014070 001001
2672 014072 005001
2673 014074 010103
2674 014076 006100
2675 014100 006101
2676 014102
    014102 010546
    014104 010146
    014106 012746 014677
    014112 012746 000003
    014116 010600
    014120 104415
    014122 062706 000010
2677 014126 022715 177777
2678 014132 001010
2679 014134
    014134 012746 014617
    014140 012746 000001
    014144 010600
    014146 104415
    014150 062706 000004
2680 014154
    014154 012746 014744
    014160 012746 000001
    014164 010600
    014166 104415
    014170 062706 000004
2681 014174 005004
2682 014176 010501
2683 014200 010300
2684 014202 001403
2685 014204 004737 020252
2686 014210 010005
2687 014212
2688 014212

```

```

      +
      : THIS ROUTINE PRINTS THE CONTENTS OF
      : THE 7 WORD MESSAGE BUFFER RETURNED BY THE
      : TUBO.
      : INPUT:
      :
      : R0      LOW ORDER ADDRESS OF MESSAGE BUFFER
      : R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
      : NOTE: R1 IS IGNORED IF KENABLE FLAG IS CLEAR
      :
      : THIS ROUTINE IS NORMALLY CALLED FROM A PRINT ROUTINE
      :
      : -
PRMESS:
      SAVREG          ;SAVE THE REGISTERS
      MOV            R5,RAMR5H ;SAVE DEVICE REGISTER POINTER
      MOV            R0,R5    ;SAVE LOW ORDER ADDRESS
      TST            KENABLE  ;ADDRESS ABOVE 28K?
      BNE            10$     ;BR IF YES
      CLR            R1      ;SET HIGH ORDER ADDRESS TO 0
10$:  MOV            R1,R3    ;SAVE HIGH ORDER ADDRESS
      ROL            R0      ;SHIFT BIT15 TO C BIT
      ROL            R1      ;SHIFT TO HIGH ORDER FOR PRINTOUT
      PRINTX        #PROASC,R1,R5 ;PRINT MESSAGE BUFFER ADDRESS
      MOV            R5,-(SP)
      MOV            R1,-(SP)
      MOV            #PROASC,-(SP)
      MOV            #3,-(SP)
      MOV            SP,R0
      TRAP          CSPNTX
      ADD            #10,SP
      CMP            #177777,(R5) ;MESSAGE BUFFER FULL OF ONES
      BNE            15$     ;BR IF BUFFER IS PROBABLY OKAY
      PRINTX        #MESBFN  ;'MESSAGE BUFFER PROBABLY NOT VALID'
      MOV            #MESBFN,-(SP)
      MOV            #1,-(SP)
      MOV            SP,R0
      TRAP          CSPNTX
15$:  ADD            #4,SP
      PRINTX        #PRIASC  ;PRINT HEADER FOR CONTENTS
      MOV            #PRIASC,-(SP)
      MOV            #1,-(SP)
      MOV            SP,R0
      TRAP          CSPNTX
      ADD            #4,SP
      CLR            R4      ;NUMBER OF THE NEXT WORD
      MOV            R5,R1   ;COPY LOW ORDER ADDRESS
      MOV            R3,R0   ;COPY HIGH ORDER ADDRESS
      BEQ            20$     ;BR IF NOT ABOVE 28K
      JSR            PC,SETMAP ;SETUP PAR ADDRESS IN R0
      MOV            R0,R5   ;GET PAR FORMAT ADDRESS ABOVE 28K
20$:  PRINTX        #MESHEA,(R5)+ ;PRINT 'MESSAGE BUFFER HEADER ='

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 44-1
 PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

	014212	012546		MOV	(R5)+,-(SP)	
	014214	012746	015002	MOV	#MESHEA,-(SP)	
	014220	012746	000002	MOV	#2,-(SP)	
	014224	010600		MOV	SP,RO	
	014226	104415		TRAP	CSPNTX	
	014230	062706	000006	ADD	#6,SP	
2689	014234			PRINTX	#DATAFL,(R5)+	;PRINT 'DATA FIELD LENGTH ='
	014234	012546		MOV	(R5)+,-(SP)	
	014236	012746	015047	MOV	#DATAFL,-(SP)	
	014242	012746	000002	MOV	#2,-(SP)	
	014246	010600		MOV	SP,RO	
	014250	104415		TRAP	CSPNTX	
	014252	062706	000006	ADD	#6,SP	
2690	014256			PRINTX	#RBPORA,(R5)+	;PRINT 'RESIDUAL BYTE COUNTER ='
	014256	012546		MOV	(R5)+,-(SP)	
	014260	012746	015114	MOV	#RBPORA,-(SP)	
	014264	012746	000002	MOV	#2,-(SP)	
	014270	010600		MOV	SP,RO	
	014272	104415		TRAP	CSPNTX	
	014274	062706	000006	ADD	#6,SP	
2691	014300			PRINTX	#XSOCN,(R5)+	;PRINT 'XSTAT0 CONTENTS ='
	014300	012546		MOV	(R5)+,-(SP)	
	014302	012746	015161	MOV	#XSOCN,-(SP)	
	014306	012746	000002	MOV	#2,-(SP)	
	014312	010600		MOV	SP,RO	
	014314	104415		TRAP	CSPNTX	
	014316	062706	000006	ADD	#6,SP	
2692	014322			PRINTX	#XS1CN,(R5)+	;PRINT 'XSTAT1 CONTENTS ='
	014322	012546		MOV	(R5)+,-(SP)	
	014324	012746	015226	MOV	#XS1CN,-(SP)	
	014330	012746	000002	MOV	#2,-(SP)	
	014334	010600		MOV	SP,RO	
	014336	104415		TRAP	CSPNTX	
	014340	062706	000006	ADD	#6,SP	
2693	014344			PRINTX	#XS2CN,(R5)+	;PRINT 'XSTAT2 CONTENTS ='
	014344	012546		MOV	(R5)+,-(SP)	
	014346	012746	015273	MOV	#XS2CN,-(SP)	
	014352	012746	000002	MOV	#2,-(SP)	
	014356	010600		MOV	SP,RO	
	014360	104415		TRAP	CSPNTX	
	014362	062706	000006	ADD	#6,SP	
2694	014366			PRINTX	#XS3CN,(R5)+	;PRINT 'XSTAT3 CONTENTS ='
	014366	012546		MOV	(R5)+,-(SP)	
	014370	012746	015340	MOV	#XS3CN,-(SP)	
	014374	012746	000002	MOV	#2,-(SP)	
	014400	010600		MOV	SP,RO	
	014402	104415		TRAP	CSPNTX	
	014404	062706	000006	ADD	#6,SP	
2695	014410	022737	000001	CMP	#1,TRANSTST	;CHECK FOR DUMP
2696	014416	001042		BNE	506	;BR, IF NO DUMP REQUIRED
2697	014420			PRINTX	#RAMFHR	
	014420	012746	014526	MOV	#RAMFHR,-(SP)	
	014424	012746	000001	MOV	#1,-(SP)	
	014430	010600		MOV	SP,RO	
	014432	104415		TRAP	CSPNTX	
	014434	062706	000004	ADD	#4,SP	
2698	014440	012737	000010	MOV	#8,RAMSIZ	;RAM FIELD IS 8 BYTES LONG

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 44-2
 PRMESS - PRINT CONTENTS OF MESSAGE BUFFER

```

2699 014446 012737 000020 011010      MOV      #20,RAMHLD      ;FIELD STARTS AT 20 OCTAL (10 HEX)
2700 014454 004737 010626                JSR      PC,RAMER      ;READ AND PRINT THEM
2701 014460 012737 000040 011010      MOV      #40,RAMHLD      ;FIELD STARTS AT 40 OCTAL (20 HEX)
2702 014466 004737 010626                JSR      PC,RAMER      ;READ AND PRINT THEM
2703 014472 012737 000060 011010      MOV      #60,RAMHLD      ;FIELD STARTS AT 60 OCTAL (30 HEX)
2704 014500 004737 010626                JSR      PC,RAMER      ;READ AND PRINT THEM
2705 014504 012737 000020 002246      MOV      #16.,RAMSIZ     ;RAM FIELD IS SIXTEEN BYTES LONG
2706 014512 012737 000100 011010      MOV      #100,RAMHLD     ;FIELD STARTS AT 100 OCTAL (40 HEX)
2707 014520 004737 010626                JSR      PC,RAMER      ;READ AND PRINT THEM
2708 014524 000207                SOS:      RTS          PC          ;RETURN
2709 014526      045      116      045  RAMFHR: .ASCIZ  'XNZA ***** SPECIAL M7454 RAM MEMORY DUMP *****'
2710 014617      045      116      045  MESBFN: .ASCIZ  'XNZA MESSAGE BUFFER CONTENTS PROBABLY NOT VALID'
2711 014677      045      116      045  PROASC: .ASCIZ  'XNZA Message Buffer Address = X01X05'
2712 014744      045      116      045  PRIASC: .ASCIZ  'XNZA Message Buffer Contents:'
2713
2714 015002      045      116      045  MESHEA: .ASCIZ  'XNZA Message Buffer Header          = X06'
2715 015047      045      116      045  DATAFL: .ASCIZ  'XNZA Data Field Length              = X06'
2716 015114      045      116      045  RBPCRA: .ASCIZ  'XNZA Residual Byte Counter          = X06'
2717 015161      045      116      045  XSOCON: .ASCIZ  'XNZA XSTAT0 Contents                = X06'
2718 015226      045      116      045  XS1CON: .ASCIZ  'XNZA XSTAT1 Contents                = X06'
2719 015273      045      116      045  XS2CON: .ASCIZ  'XNZA XSTAT2 Contents                = X06'
2720 015340      045      116      045  XS3CON: .ASCIZ  'XNZA XSTAT3 Contents                = X06'
2721                .EVEN

```

CZTUXAO TURD FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 45
PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS

```

2723 .SBTTL PRMSGEXP - PRINT EXPD/RCV MESSAGE BUFFERS
2724 :+[B
2725 :ROUTINE TO PRINT EXPECTED AND RECEIVED MESSAGE BUFFERS
2726 RO - NUMBER OF WORDS IN BUFFER
2727 :IMPLICIT INPUTS:
2728 EXPMSG - EXPECTED MESSAGE BUFFER
2729 RECMMSG - RECEIVED MESSAGE BUFFER
2730 RCVHIADD- RECEIVED MESSAGE BUFFER HIGH ORDER ADDRESS
2731 RCVLOADD- RECEIVED MESSAGE BUFFER LOW ORDER ADDRESS
2732 :-
2733 PRMSGEXP::
2734 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2735 MOV RO,R5 ;SAVE NUMBER OF WORDS
2736 MOV RCVLOADD,RO ;GET RCV LOW ADDRESS
2737 MOV RO,R4 ;COPY LOW ADDRESS
2738 MOV RCVHIADD,R1 ;GET RCV HIGH ADDRESS
2739 ROL RO ;SHIFT BIT15 TO C BIT
2740 ROL R1 ;SHIFT TO HIGH ORDER FOR PRINTOUT
2741 PRINTX #PRMSGO,R1,R4 ;PRINT MESSAGE BUFFER ADDRESS
2742 MOV R4,-(SP)
2743 MOV R1,-(SP)
2744 MOV #PRMSGO,-(SP)
2745 MOV #3,-(SP)
2746 MOV SP,RO
2747 TRAP CSPNTX
2748 ADD #10,SP
2749 PRINTX #PRMSG1 ;PRINT HEADER FOR CONTENTS
2750 MOV #PRMSG1,-(SP)
2751 MOV #1,-(SP)
2752 MOV SP,RO
2753 TRAP CSPNTX
2754 ADD #4,SP
2755 CLR R4 ;NUMBER OF THE CURRENT WORD
2756 MOV #EXPMSG,R1 ;GET EXPD BUFFER ADDRESS
2757 MOV #RCMMSG,R2 ;GET RCV BUFFER ADDRESS
2758 MOV (R1),RO ;GET EXPD
2759 MOV (R2),R3 ;GET RCV
2760 XOR RO,R3 ;XOR EXPD/RCV
2761 PRINTX #PRMSG2,R4,(R1)+,(R2)+,R3
2762 MOV R3,-(SP)
2763 MOV (R2)+,-(SP)
2764 MOV (R1)+,-(SP)
2765 MOV R4,-(SP)
2766 MOV #PRMSG2,-(SP)
2767 MOV #5,-(SP)
2768 MOV SP,RO
2769 TRAP CSPNTX
2770 ADD #14,SP
2771 INC R4 ;NUMBER OF THE NEXT
2772 CMP R4,R5 ;DONE ALL YET?
2773 BGE 50$ ;BR IF YES
2774 BR 20$ ;DO ANOTHER
2775 50$: RTS ;RETURN
2776 PC
2777 20$: PRMSGO: .ASCII 'XNZA Message Buffer Address = X01X05'
2778 PRMSG1: .ASCII 'XNZA Message Buffer Contents:'
2779 PRMSG2: .ASCII 'XNZA WORD #XD2XA EXPD: X06XA RCV: X06XA XOR: X06'
2780 .EVEN

```

TUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 46
 PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER

```

2760 .SBTTL PRBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER
2761
2762
2763 ROUTINE TO PRINT ERROR BYTES IN MESSAGE BUFFERS
2764 ONLY THE FIRST 8 ERRORS ENCOUNTERED ARE PRINTED DUE TO SCREEN SPACE
2765
2766 R0 - NUMBER OF BYTES IN BUFFER
2767
2768 IMPLICIT INPUTS:
2769
2770 EXPMSG - EXPECTED MESSAGE BUFFER
2771 RECMG - RECEIVED MESSAGE BUFFER
2772
2773 PRBYTEXP::
2774 SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
2775 MOV R0,R5 ;SAVE NUMBER OF BYTES
2776 CLR PRMNO ;INIT ERROR COUNT
2777 CLR R4 ;NUMBER OF THE CURRENT BYTE
2778 MOV #EXPMSG,R1 ;GET EXPD BUFFER ADDRESS
2779 MOV #RECMG,R2 ;GET RECV BUFFER ADDRESS
2780 20$: MOVB (R1),R0 ;GET EXPD BYTE
2781 BIC #^C<377>,R0 ;CLEAR UPPER BYTE
2782 MOVB R0,PRBEXP ;SAVE FOR ERROR REPORT
2783 MOVB (R2),R3 ;GET RECV BYTE
2784 BIC #^C<377>,R3 ;CLEAR UPPER BYTE
2785 MOVB R3,PRBREC ;FOR ERROR REPORT
2786 XOR R0,R3 ;XOR EXPD/RECV
2787 CMPB (R1)+,(R2)+ ;EXPD = RECV?
2788 BEQ 30$ ;BR IF YES
2789 INC PRMNO ;UPDATE ERROR COUNT
2790 2790 016046 023727 002264 000010 CMP PRMNO,#8. ;PRINTED 8?
2791 BHI 30$ ;BR IF YES
2792 27$: PRINTX #PRMSG,R4,PRBEXP,PRBREC,R3
2793 MOV R3,-(SP)
2794 MOV PRBREC,-(SP)
2795 MOV PRBEXP,-(SP)
2796 MOV R4,-(SP)
2797 MOV #PRMSG,-(SP)
2798 MOV #5,-(SP)
2799 MOV SP,R0
2800 016104 104415 TRAP CSPNTX
2801 016106 062706 000014 ADD #14,SP
2802 2793 016112 FORCEEXIT 50$ ;20D
2803 2794 016122 000404 BR 35$ ;20D
2804 2795 016124 30$: FORCERROR 27$,NOTSSR ;20D
2805 2796 016124 35$: ;20D
2806 2797 016134 ;20D
2807 2798 016134 005204 INC R4 ;NUMBER OF THE NEXT
2808 2799 016136 020405 CMP R4,R5 ;DONE ALL YET?
2809 2800 016140 002001 BGE 50$ ;BR IF YES
2810 2801 016142 000717 BR 20$ ;DO ANOTHER
2811 2802 016144 50$: PRINTX #PRBTOT,PRMNO ;PRINT TOTAL ERROR COUNT
2812 016144 013746 002264 MOV PRMNO,-(SP)
2813 016150 012746 016257 MOV #PRBTOT,-(SP)
2814 016154 012746 000002 MOV #2,-(SP)
2815 016160 010600 MOV SP,R0
2816 016162 104415 TRAP CSPNTX
  
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 46-1
PPBYTEXP - PRINT ERROR BYTES IN EXP/REC MESSAGE BUFFER

```

2803 016164 062706 000006          ADD      #6,SP
2804 016170 000207          RTS      PC          ;RETURN
2805 016172      045      116      045 PRBMSG: .ASCIZ  'XNZA  BYTE #XD2XA  EXPD: X03XA  RECV: X03XA  XOR: X03'
2806 016257      045      116      045 PRBTOT: .ASCIZ  'XNZA  NUMBER OF BYTES IN ERROR = XD2'
2807                                .EVEN
2808 016324 000000          PRBEXP: .WORD  0          ;EXPD
2809 016326 000000          PRBREC: .WORD  0          ;RECV
2810
2811          ;+
2812          ;PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2813          ;INPUTS:
2814          ;
2815          ;
2816          ;
2817          ;
2818          ;
2819          ;
2820          ;-
2821 016330          BGNMSG  EXPREC
2822 016330 004737 007462          EXPREC:: JSR      PC,PRIXOR      ;PRINT THE DATA
2823 016334          ENDMSG
          L10017: TRAP      CSMSG
          016334 104423

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 47
EXPBREC - PRINT EXPD/RECV BYTE DATA

2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838 016336
016336
2839 016336 004737 007332
2840 016342
016342
016342 104423

```
.SBTTL EXPBREC - PRINT EXPD/RECV BYTE DATA
:+
:PRINT ROUTINE TO DISPLAY BYTE EXPD/RECV DATA
:INPUTS:
:      R1      RECEIVED DATA BYTE
:      R2      EXPECTED DATA BYTE
:-
      BGNMSG EXPBREC
EXPBREC::
      JSR      PC,PRIBXOR      ;PRINT THE DATA
      ENDMSG
L10020:
      TRAP    CSMSG
```

2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865 016344
016344
2866 016344 004737 013620
2867 016350
016350
016350 104423

```
.SBTTL RAMERR - PRINT RAM AND PACKET DATA
:+
:PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
:INPUTS:
:      R4      POINTER TO COMMAND PACKET
:IMPLICIT INPUTS:
:      RAMDATA DATA AS READ FROM THE RAM
:      RAMSIZ  NUMBER OF BYTES IN PACKET
:              IF RAMSIZ=0 THEN DEFAULT TO 8.
:IMPLICIT OUTPUTS:
:      RAMSIZ  SET TO 0
:-
      BGNMSG RAMERR
RAMERR::
      JSR      PC,PRAMPKT      ;PRINT RAM/PACKET DATA
      ENDMSG
L10021:
      TRAP    CSMSG
```

2868
2869
2870
2871
2872
2873
2874
2875

```
.SBTTL RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA
:+
:PRINT ROUTINE TO DISPLAY RAM/PACKET DATA
:INPUTS:
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 47-1
 RAMTADD - PRINT TEST ADDRESS, RAM AND PACKET DATA

```

2876
2877          R4      POINTER TO COMMAND PACKET
2878
2879      :IMPLICIT INPUTS:
2880
2881          RAMDATA   DATA AS READ FROM THE RAM
2882          RAMSIZ    NUMBER OF BYTES IN PACKET
2883                  IF RAMSIZ=0 THEN DEFAULT TO 8.
2884          ERRHI     HIGH ORDER TEST ADDRESS
2885          ERRLO     LOW ORDER TEST ADDRESS
2886
2887      :IMPLICIT OUTPUTS:
2888
2889          RAMSIZ    SET TO 0
2890      :-
2891
2892      BGNMSG  RAMTADD
2893      RAMTADD::
2894      JSR     PC,PRITADD      ;PRINT TEST ADDRESS
2895      JSR     PC,PRAMPKT     ;PRINT RAM/PACKET DATA
2896      ENDMSG
2897
2898      L10022:
2899      TRAP    CSMSG
2900
2901      .SBTTL  RAMEXP - PRINT RAM EXPD/RECV DATA
2902      :+
2903      :PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2904      :INPUTS:
2905
2906          R1      RECEIVED DATA
2907          R2      EXPECTED DATA
2908          R4      CONTROLLER RAM ADDRESS
2909      :-
2910
2911      BGNMSG  RAMEXP
2912      RAMEXP::
2913      BIC     #'C<377>,R1    ;SAVE EXPD RAM DATA BYTE
2914      BIC     #'C<377>,R2    ;SAVE EXPD RAM DATA BYTE
2915      JSR     PC,PRIRAM      ;PRINT THE RAM ADDRESS
2916      JSR     PC,PR'XOR      ;PRINT THE DATA
2917      ENDMSG
2918
2919      L10023:
2920      TRAP    CSMSG
2921
2922      .SBTTL  TIMEXP - PRINT TIMER A,B AND EXP/REC
2923      :+
2924      :PRINT ROUTINE TO DISPLAY EXPD/RECV DATA
2925      :AND TIMER A,B HEADER MESSAGE
2926      :INPUTS:
2927
2928          R1      RECEIVED DATA
2929          R2      EXPECTED DATA
    
```

```

2892 016352
    016352
2893 016352 004737 010014
2894 016356 004737 013620
2895 016362
    016362
    016362 104423
    
```

```

2910 016364
    016364
2911 016364 042701 177400
2912 016370 042702 177400
2913 016374 004737 007606
2914 016400 004737 007462
2915 016404
    016404
    016404 104423
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 47-2
TIMEXP - PRINT TIMER A,B AND EXP/REC

```

2927
2928
2929 016406
      016406
2930 016406
      016406 012746 016434
      016412 012746 000001
      016416 010600
      016420 104415
      016422 062706 000004
2931 016426 004737 007462
2932 016432
      016432
      016432 104423
2933
2934
2935 016434 045 116 045
2936

```

```

:-
      BGNMSG  TIMEXP
TIMEXP::
      PRINTX  #TIMSGO      ;PRINT HEADER
      MOV     #TIMSGO,-(SP)
      MOV     #1,-(SP)
      MOV     SP,R0
      TRAP   C$PNTX
      ADD    #4,SP
      JSR   PC,PRIXOR      ;PRINT THE DATA
      ENDMSG
L10024:
      TRAP   C$MSG
TIMSGO: .ASCIZ 'XNXA TIMER A STATUS IS IN BIT 3XNXA TIMER B STATUS IS IN BIT 2'
      .EVEN

```


CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 48
BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS

.SBTTL BADSSR - PRINT TSSR ERRORS ON DATA TRANSFERS

2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950

:+
:PRINT ROUTINE FOR TSSR ERRORS ON DATA TRANSFERS
:INPUTS:
: R1 CONTENTS OF TSSR
: R2 DATA WRITTEN (8 BITS)
:-

2951 016534
016534
2952 016534 010246
2953 016536 042702 177400
2954 016542
016542 010246
016544 012746 016574
016550 012746 000002
016554 010600
016556 104414
016560 062706 000006
2955 016564 012602
2956 016566 004737 005264
2957 016572
016572
016572 104423
2958 016574 045 116 0'5

BGNMSG BADSSR
BADSSR::
MOV R2,-(SP) ;SAVE DATA TRANSFERRED
BIC #177400,R2 ;GET JUST ONE BYTE
PRINTB #XFERASC,R2
MOV R2,-(SP)
MOV #XFERASC,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP
MOV (SP)+,R2 ;RESTORE R2
JSR PC,PRITSSR ;DECODE TSSR CONTENTS
ENDMSG
L10025:
TRAP C\$MSG
XFERASC: .ASCIZ 'XNZA Data Transferred = X03'

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 49
 SOFINIT - SOFT INITIALIZE OF CONTROLLER

```

2960                                     .SBTTL SOFINIT - SOFT INITIALIZE OF CONTROLLER
2961
2962                                     :+
2963                                     :
2964                                     :ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
2965                                     :BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
2966                                     :THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
2967                                     :DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
2968
2969                                     :INPUTS:
2970                                     :
2971                                     :       R5      ADDRESS OF FIRST REGISTER
2972
2973                                     :OUTPUTS:
2974                                     :
2975                                     :       R0      CONTENTS OF TSSR, IF ERROR
2976                                     :       CARRY   SET IF INIT WAS OKAY
2977                                     :               CLEAR IF FATAL ERROR
2978
2979                                     :CALLING SEQUENCE:
2980                                     :
2981                                     :       MOV     #ADDRESS,R5
2982                                     :       JSR     PC,SOFINIT
2983                                     :       BCS    CONTINUE
2984                                     :       ERRDF          :REPORT FATAL ERROR
2985
2986                                     :-
2987
2988 016630 SOFINIT::
2989 016630 SAVREG          : SAVE THE REGISTERS
2990 016634 012765 000000 000000 MOV     #0,TSSR(R5)      : DO THE INIT.
2991 016642 004737 017104 JSR     PC,WAITF        : WAIT FOR SSR
2992 016646 016500 000000 MOV     TSSR(R5),R0    :GET THE TSSR REGISTER
2993 016652 010004 MOV     R0,R4          :TSSR CONTENTS
2994 016654 042704 176277 BIC     #^C<HIADDR!OFL>,R4
2995 016660 052704 002200 BIS     #SSR!NBA,R4   :R4 HAS EXPECTED CONTENTS
2996 016664 020400 CMP     R4,R0          :ONLY EXPECTED BITS SET ?
2997 016666 001402 BEQ     5$          :BRANCH IF OKAY
2998 016670 000241 CLC          :CLEAR THE CARRY FOR ERROR
2999 016672 000401 BR     10$         :GO TO EXIT
3000 016674 000261 5$: SEC          :SET THE CARRY BIT
3001 016676 000207 10$: RTS     PC    :RETURN TO CALLER

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 50
 CHKAMB - CHECK TSSR FOR AMBIGUITY

.SBT ' CHKAMB - CHECK TSSR FOR AMBIGUITY

3003
 3004
 3005
 3006
 3007
 3008
 3009
 3010
 3011
 3012
 3013
 3014
 3015
 3016
 3017
 3018
 3019
 3020
 3021
 3022
 3023 016700
 3024 016700
 3025 016704 010004
 3026 016706 032700 100000
 3027 016712 001004
 3028 016714 032700 174077
 3029 016720 001023
 3030 016722 000424
 3031 016724 032700 000200
 3032 016730 001011
 3033 016732 032700 000040
 3034 016736 001414
 3035 016740 042704 177761
 3036 016744 020427 000016
 3037 016750 001007
 3038 016752 000410
 3039 016754 032700 000040
 3040 016760 001405
 3041 016762 032700 000006
 3042 016766 001002
 3043 016770 000241
 3044 016772 000401
 3045 016774 000261
 3046 016776 000207
 3047

:+
 : THIS ROUTINE TESTS THE CONTENTS OF THE TSSR REGISTER
 : FOR AMBIGUITY

: INPUT:

RO CONTENTS OF TSSR

: OUTPUT:

RO CONTENTS OF TSSR

CARRY SET - NO AMBIGUITY
 CLR - AMBIGUOUS CONTENTS

:-
 CHKAMB:

```

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

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 51
 ENAINT,DSBINT - ENABLE/DISABLE INTERRUPTS

```

3049          .SBTTL ENAINT,DSBINT - ENABLE/DISABLE INTERRUPTS
3050          ;
3051          ; DEFAULT DISPLAY INTERRUPT HANDLERS.
3052          ; IF DISPLAY TIME-OUT, REPORT DEV FATAL, AND ABORT PASS.
3053          ; OTHERWISE, SAVE D*J REGISTERS AND DISMISS.
3054          ;
3055          ;
3056          ; BIT DEFINITIONS FOR "INTMASK" AND "INTFLAG" BYTES:
3057          ;
3058          ; IOKCKIN=BIT7          ; DON'T CHECK FOR BAD INTERRUPTS -- TEST WILL.
3059          ; IOKSTP=BIT0         ; EXPECT "STOP" INTERRUPT.
3060          ;
3061          ; INTERRUPT MASK -- SAYS EXPECTING INTERRUPTS
3062          INTMASK: .BYTE 0
3063          ; INTERRUPT FLAG -- SAYS WE GOT ONE (IF POSITIVE)
3064          INTFLAG: .BYTE 0
3065          ;
3066          ; SAVED INTERRUPT VECTOR:
3067          INTVEC: .WORD 0
3068          ; SAVE CPU PC
3069          INTCPC: .WORD 0
3070          ;
3071          ; SUBROUTINE TO ENABLE INTERRUPTS:
3072          ENAINT: MOV     R0,-(SP)          ;SAVE R0
3073                   MOV     IVEC,R0        ;GET POINTER TO VECTORS
3074                   MOV     #INTR,(R0)+    ;SET UP INTERRUPT VECTOR
3075                   MOV     #PRI07,(R0)+
3076                   MOV     (SP)+,R0      ;RESTORE R0
3077                   MOV     (SP),-(SP)
3078                   MOV     #0,2(SP)      ;SET CPU TO LEVEL 0
3079                   RTI
3080          ;
3081          ; SUBROUTINE TO DISABLE INTERRUPTS (RAISE PRIORITY TO LEVEL 7)
3082          DSBINT: MOV     (SP),-(SP)
3083                   MOV     #PRI07,2(SP)
3084                   RTI
3085
000200
000001
017000 000
017001 000
017002 000000
017004 000000
017006 010046
017010 013700 002156
017014 012720 017052
017020 012720 000340
017024 012600
017026 011646
017030 012766 000000 000002
017036 000002
017040 011646
017042 012766 000340 000002
017050 0C0002

```

CZTUXAO TUBO FRONT END PRT B
INTR - INTERRUPT HANDLERS

MACRO M1200 29-MAR-83 13:32 PAGE 52

```

3087          .SBTTL  INTR  - INTERRUPT HANDLERS
3088
3089 017052    BGNSRV  INTR          :DEFINE INTERRUPT ENTRY
3090 017052    INTR::
3091 017052 012737 000001 002172    MOV      #1,INTRECV      :SET FLAG TO SHOW INTERRUPT RECEIVED
3092 017060 105037 017001          CLRB     INTFLAG        :CLEAR FLAG TO SAY WE GOT INTERRUPT
3093 017064 132737 000001 017000    BITB    #IOKSTP,INTMASK :EXPECTING STOP INTERRUPT?
3094 017072 001003          BNE     1$              :BR IF YES
3095 017074 152737 000001 017001    BISB    #IOKSTP,INTFLAG ;NO. SET THE ERROR FLAG.
3096
3097          :SAVE REGISTERS, MSG BUFFER, ETC.
3098 017102    1$:
3099 017102          ENDSRV
3100 017102 000002    L10026:
3101          RTI

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 53
 WAITF - WAIT FOR SUBSYSTEM READY

```

3102 .SETTL WAITF - WAIT FOR SUBSYSTEM READY
3103
3104 : SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
3105 :
3106 : INPUTS:
3107 :
3108 : R5 ADDRESS OF FIRST DEVICE REGISTER
3109 :
3110 : OUTPUTS:
3111 :
3112 : R0 CONTENTS OF LAST TSSR READ
3113 : CARRY SET - READY BIT SET
3114 : CLR - TIMEOUT WAITING FOR READY
3115 :
3116 WAITF:: BREAK ; DO A SUPVSR BREAK FIRST.
          TRAP CSBRK
3117 017104 104422 010000 MOV #10000,-(SP) ;BIG MSEC TIMER
3118 017112 012727 000001 DELAY 1 ;DELAY 100US
          017112 012727 000001 MOV #1,(PC)+
          017116 000000 .WORD 0
          017120 013727 002116 MOV LSDLY,(PC)+
          017124 000000 .WORD 0
          017126 005367 177772 DEC -6(PC)
          017132 001375 BNE -4
          017134 005367 177756 DEC -22(PC)
          017140 001367 BNE -20
3119 017142 016500 000000 2$: MOV TSSR(R5),R0 ;READ THE TSSR REGISTER
3120 017146 105700 TSTB R0 ;TEST FOR READY BIT SET
3121
3122 017150 100420 BMI 3$ ; EXIT ON STOP FLAG.
3123 017152 012727 000001 .FLAY 1 ; WAIT 100 USEC
          017152 012727 000001 MOV #1,(PC)+
          017156 000000 .WORD 0
          017160 013727 002116 MOV LSDLY,(PC)+
          017164 000000 .WORD 0
          017166 005367 177772 DEC -6(PC)
          017172 001375 BNE -4
          017174 005367 177756 DEC -22(PC)
          017200 001367 BNE -20
3124 017202 005316 DEC (SP) ;REDUCE DELAY COUNT
3125 017204 001356 BNE 2$ ;RETRY UNTIL TIMER EXPIRES
3126 017206 000241 CLC ; C = 0, CONTROLLER STILL RUNNING...
3127 017210 000401 BR 4$ ;...OR HUNG-UP AFTER 300 MSEC.
3128 017212 000261 3$: SEC ; C = 1, CONTROLLER IS STOPPED.
3129 017214 005326 4$: DEC (SP)+ ;RESTORE STACK WITHOUT CHANGING CARRY BIT
3130 017216 000207 RTS PC
    
```

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 54
 CHKTSSR - CHECK TSSR FOR READY

.SBTTL CHKTSSR - CHECK TSSR FOR READY

3132
 3133
 3134
 3135
 3136
 3137
 3138
 3139
 3140
 3141
 3142
 3143
 3144
 3145
 3146
 3147
 3148
 3149
 3150

```

: +
: THIS ROUTINE WAITS FOR READY IN THE TSSR
: AND TESTS FOR AMBIGUOUS BIT SETTINGS IN TSSR.
: INPUT:
:         R5      ADDRESS OF CSR REGISTERS
: OUTPUT:
:         R0      CONTENTS OF TSSR
:         CARRY   SET - OKAY
:                CLR - NOT READY AMBIGUOUS, OR SC SET
: -
    
```

3151 017220
 3152 017220 004737 017104
 3153 017224 103014
 3154 017226 004737 016700
 3155 017232 103006
 3156 017234 032700 100000
 3157 017240 001405
 3158 017242 032700 074000
 3159 017246 001402
 3160 017250 000241
 3161 017252 000401
 3162 017254 000261
 3163 017256 000207

```

CHKTSSR:
        JSR      PC, WAITF      ;WAIT FOR READY
        BCC      20$           ;BRANCH IF TIME OUT
        JSR      PC, CHKAMB     ;TSSR AMBIGUOUS?
        BCC      10$           ;BR IF YES
        BIT      #SC, R0        ;SPECIAL CONDITION SET?
        BEQ      15$           ;BR IF NO
        BIT      #<SCE!BIE!RMR!NXM>, R0 ;ANY ERROR BITS SET?
        BEQ      15$           ;BR IF NO
10$:    CLC                    ;SET FAILURE
        BR       20$
15$:    SEC                    ;SET SUCCESS
20$:    RTS      PC           ;RETURN TO CALLER
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 55
 XNXM - CHECK FOR NONEXISTENT MEMORY

```

3165 .SBTTL XNXM - CHECK FOR NONEXISTENT MEMORY
3166
3167 :+
3168 : ROUTINE TO TEST FOR A NEXM IN THE RANGE (R1) THRU (R2).
3169 : ON RETURN, IF 'C' = 1, (R1) = NEXM ADDRESS.
3170 : 'C' = 0, ALL ADDRESSES OK.
3171 :CALL: MOV ADR1,R1
3172 : MOV ADR2,R2
3173 : JSR PC,NXM
3174 : RETURN ;TEST 'C' AND PROCEED.
3175
3176 017260 012737 017312 000004 XNXM: MOV #2$,2#4 ; SET BUSERR VECTOR.
3177 017266 012737 000200 000006 MOV #PR104,2#6
3178 017274 005003 CLR R3 ;FLAG.
3179 017276 005711 1$: TST (R1) ;TEST THE ADDRESS(ES).
3180 ;IF ANY TRAP, CONTINUE AT 2$.
3181 017300 020102 CMP R1,R2 ;OTHERWISE, CONTINUE HERE.
3182 017302 001407 BEQ 3$ ;BR IF FINISHED (NO NEXM'S).
3183 017304 062701 000002 ADD #2,R1 ;SET NEXT ADDRESS...
3184 017310 000772 BR 1$ ;...AND CONTINUE.
3185
3186 017312 005103 2$: COM R3 ;GOT ONE, SET FLAG...
3187 017314 012716 017322 MOV #3$, (SP)
3188 017320 000002 RTI ;...AND DISMISS INTERRUPT...
3189 017322 3$: CLRVEC #4 ;...AND GIVE BACK THE VECTOR.
3190 017322 012700 000004 MOV #4,R0
3191 017322 104436 TRAP CSCVEC
3192 017330 005703 TST R3 ;DID WE CATCH ONE ??
3193 017332 001401 BEQ .+4 ;NO, 'C' = 0, SKIP NEXT.
3194 017334 000261 SEC ;YES, 'C' = 1, (R1) = NEXM ADDR.
3195 017336 000207 RTS PC
3196
3197
3198
3199
3200 .SBTTL TSTLOOP - CHECK ITERATION COUNT
3201 :+
3202 : SUBROUTINE TO EXECUTE TEST ITERATIONS.
3203 : EXIT WITH 'C' SET IF LOOPS ALLOWED AND LOOP COUNT NON-ZERO.
3204 : LOOP COUNTER IS SET BY 'BEGIN.TEST' MACRO.
3205 :CALL: LOOPTO ARG
3206
3207 TSTLOOP::
3208 TST NOITS ; ITERATIONS INHIBITED?
3209 BNE 1$ ; YES.
3210 TST QVP ; NO.
3211 BMI 1$ ;LOOPS DISALLOWED IN QUICK PASS.
3212 DEC LOOPCNT ; BUMP LOOP COUNTER.
3213 BNE 2$
3214 1$: CLC ;LOOP DISALLOWED, OR DONE.
3215 BR 3$
3216 2$: SEC ;LOOP ENABLED.
3217 3$: RTS PC
  
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 56
 TSTLOOP - CHECK ITERATION COUNT

3218
 3219
 3220
 3221
 3222
 3223
 3224
 3225
 3226
 3227
 3228
 3229
 3230
 3231
 3232
 3233
 3234
 3235
 3236
 3237
 3238
 3239
 3240
 3241
 3242
 3243
 3244
 3245

.SBTTL TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS

```

: PRINT THE NUMBER AND NAME OF EACH TEST AS WE GO ALONG.
: INCREMENT 'TESTK' TO INDICATE THE NUMBER OF TESTS
: IN THE CURRENT RUN SEQUENCE.
: CLEAR THE ERROR COUNTER AND SIGNATURE EXTENSION FLAGS.
: INPUT:
:         RO        POINTER TO TEST ID ASCIZ STRING
: OUTPUT:
:         R5        ADDRESS OF FIRST DEVICE REGISTER
: IMPLICIT OUTPUTS:
:         TSTCNT    UPDATED TO COUNT TESTS PERFORMED SINCE START OR RESTART
: SIDE EFFECTS:
:         INTERRUPT LEVEL IS RASIED TO LEVEL OF
:         THE DEVICE UNDER TEST
: -

```

```

3246 017372
3247 017372 010046
3248 017374 005037 003106
3249 017400 005037 017640
3250 017404 005037 005232
3251 017410 105037 017000
3252 017414 013700 002150
3253 017420 006300
3254 017422 005737 003062
3255 017426 001430
3256 017430 100010
3257 017432 052760 160000 003130
3258 017440
      017440 104455
      017442 000001
      017444 003636
      017446 005176
3259 017450 000407
3260 017452 052760 160001 003130 38:
3261 017460
      017460 104455
      017462 000002
      017464 004233
      017466 000000
3262 017470 012737 177777 003060 28:
3263 017476
      017476 013700 002150
      017502 104451
3264 017504

```

```

TSTSETUP::
      MOV      RO,-(SP)          ;SAVE THE TEST ID MESSAGE
      CLR      SIFLAG           ; CLEAR "SOFT INIT" FLAG
      CLR      ERR              ; CLEAR LOCAL ERROR COUNTER.
      CLR      EXTA            ; CLEAR ERROR EXTENSION FLAG.
      CLRB     INTMASK         ; CLEAR INTERRUPT MASK (CHECK ERROR)
      MOV      UNITN,RO        ; GET THE UNIT NUMBER,
      ASL      RO              ; ... AND MAKE IT A WORD OFFSET.
      TST      NODEV           ; DID STARTUP FIND THE DEVICE?
      BEQ      48              ; BR IF YES
      BPL      38              ; BR IF NOT IDLE
      BIS      #160000,ERTABL(RO) ; FLAG ERROR IN THE ERROR TABLE
      ERDF     1,NXR,NXRERR    ; NO DEVICE HERE -- PRINT IT
      TRAP    CSERDF
      .WORD   1
      .WORD   NXR
      .WORD   NXRERR
      BR      28
      BIS      #160001,ERTABL(RO) ; FLAG ERROR IN THE ERROR TABLE
      ERDF     2,NOINIT       ; DEVICE NOT IDLE
      TRAP    CSERDF
      .WORD   2
      .WORD   NOINIT
      .WORD   0
      MOV      #-1,DUFLG      ; DROP THE UNIT
      DODU     UNITN
      MOV      UNITN,RO
      TRAP    CSODU
      DOCLN
      ; ABORT THE PASS

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 56-1
 TSTSETUP - PRINT TEST NAME AND INIT ERROR COUNTS

3265	017504	104444				TRAP	CSDCLM		
	017506	00C423				BR	58		
3266									
3267	017510			48:		RFLAGS	R0		: GET THE OPERATOR FLAGS.
	017510	104421				TRAP	CSR+LA		
3268	017512	032700	001000			BIT	#PNT,R0		: PRINT THE TEST NUMBERS?
3269	017516	001412				BEQ	18		: BR IF NO
3270	017520	011600				MOV	(SP),R0		:GET THE ID MESSAGE
3271	017522					PRINTF	#TNAM,R0		:DISPLAY THE TEST ID
	017522	010046				MOV	R0,-(SP)		
	017524	012746	017566			MOV	#TNAM,-(SP)		
	017530	012746	000002			MOV	#2,-(SP)		
	017534	010600				MOV	SP,R0		
	017536	104417				TRAP	CSPNTF		
	017540	062706	000006			ADD	#6,SP		
3272	017544	005237	002162		18:	INC	TSTCNT		: BUMP TEST COUNTER.
3273	017550					SETPRI	IPRI		:PRIORITY THAT OF DEVICE
	017550	013700	002160			MOV	IPRI,R0		
	017554	104441				TRAP	CSSPRI		
3274	017556	005726			58:	TST	(SP)+		:FIX UP THE STACK
3275	017560	013705	002154			MOV	CSRADDR,R5		: ADDRESS OF TSV REGISTERS ON UNIBUS
3276	017564	000207				RTS	PC		
3277	017566	045	123	045	TNAM:	.ASCIZ	'%SXTXA Test'		
3278						.EVEN			

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 57
TSTEND - PRINT ERRORS RECEIVED

```

3280
3281
3282
3283
3284
3285 017602
      017602 104421
3286 017604 030027 020000
3287 017610 001412
3288 017612
      017612 013746 017640
      017616 012746 017642
      017622 012746 000002
      017626 010600
      017630 104417
      017632 062706 000006
3289 017636 000207
3290
3291 017640 000000
3292 017642 045 101 040
3293 017661 105 122 122
3294
3295
3296
3297
3298
3299
3300 017726 005237 017640
3301 017732 010046
3302 017734 013700 002150
3303 017740 006300
3304 017742 062700 003130
3305 017746 005210
3306 017750 032710 007777
3307 017754 001001
3308 017756 005310
3309 017760 012600
3310 017762 000207
3311
3312 017764 010046
3313 017766 013700 002150
3314 017772 006300
3315 017774 016000 003130
3316 020000 042700 170000
3317 020004 020037 002142
3318 020010 103004
3319 020012 023737 017640 002140
3320 020020 103417
3321 020022
      020022 104421
3322 020024 032700 000040
3323 020030 001013
3324 020032 012737 177777 003060
3325 020040
      020040 104455
      020042 000004
      020044 017661

```

```

.SBTTL TSTEND - PRINT ERRORS RECEIVED
:
: AT END OF EACH TEST, PRINT THE NUMBER OF ERRORS RECEIVED
: IF NORMAL ERROR REPORTING IS DISABLED (FLA:IER).
:
TSTEND: RFLAGS RO
        TRAP CSRFLA
        BIT RO,#IER
        BEQ 1$ ; BR IF "IER" NOT SET.
        PRINTF #ESUM,ERRK ; PRINT ERROR COUNT.
        MOV ERRK,-(SP)
        MOV #ESUM,-(SP)
        MOV #2,-(SP)
        MOV SP,RO
        TRAP C$PNTF
        ADD #6,SP
1$: RTS PC

ERRK: 0 ; LOCAL ERROR COUNT.
ESUM: .ASCIZ /%A %DXA ERRORS/
EMAXDU: .ASCIZ /ERROR LIMIT REACHED -- DROPPING UNIT/
        .EVEN

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

CKEMAX: MOV RO,-(SP) ; SAVE RO
        MOV UNITN,RO ; GET UNIT NUMBER
        ASL RO ; ... AND MAKE IT A WORD OFFSET
        MOV ERTABL(RO),RO ; GET ERROR TABLE ENTRY
        BIC #17000,RO ; EXTRACT ERROR COUNT FIELD
        CMP RO,GERRMAX ; IS GLOBAL LIMIT EXCEEDED FOR THIS UNIT?
        BHIS 1$ ; BR IF YES
        CMP ERRK,LERRMAX ; IS LOCAL LIMIT EXCEEDED FOR THIS TEST?
        BLO 2$ ; BR IF NO
1$: RFLAGS RO ; GET OPERATOR FLAGS
        TRAP CSRFLA
        BIT #IDU,RO ; IS DROPPING INHIBITED?
        BNE 2$ ; BR IF YES.
        MOV #-1,DUFLG ; NO -- DROP THE UNIT
        ERDF 4,EMAXDU
        TRAP C$ERDF
        .WORD 4
        .WORD EMAXDU

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 57-1
INCRK - INCREMENT LOCAL ERROR COUNT

```

3326 020046 000000
      020050
      020050 013700 002150
      020054 104451
3327 020056
      020056 104444
3328 020060 012600
3329 020062 000207
3330
3331
3332
3333
3334
3335
3336
3337 020064
3338 020064
3339 020070 013701 002150
3340 020074 006301
3341 020076 062761 000001 003130
3342 020104 005237 002170
3343 020110 023727 002170 000031
3344 020116 002406
3345 020120
      020120 104421
3346 020122 032700 040000
3347 020126 001002
3348 020130 004737 020136
3349 020134 000207
3350
3351
3352

```

```

      .WORD 0
      DODU UNITN
      MOV UNITN,RO
      TRAP CSDODU
      DOCLN
      TRAP CSDCLN
2$: MOV (SP)+,RO ; RESTORE RO
      RTS PC ; RETURN TO CALLER
      .SBTTL FATCHK - INC FATAL ERRORS AND CHECK FOR LIMIT

```

```

      :+
      :
      : CHECK FATAL COUNTER, AFTER INC, FOR MORE THAN 25
      : ERRORS AND IF OVER CALL UNIT DROP ROUTINE
      :-

```

```

FATCHK:
      SAVREG
      MOV UNITN,R1 ; BETTER SAVE THE REGISTERS
      ASL R1 ; PICK UP THE UNIT NUMBER
      ADD #1,ERTABL(R1) ; MAKE IT INTO A BYTE OFFSET
      INC FATFLG ; ADD 1 TO THE PROPER UNIT'S ERROR COUNTER
      CMP FATFLG,#25. ; BUMP FATAL ERROR COUNTER
      BLT 9$ ; CHECK AGAINST 25
      RFLAGS RO ; BR, IF LESS THAN 25 ERRORS
      TRAP CSRFLA ; READ THE FLAGS INTO RO
      BIT #BIT14,RO ; BR, IF LOOP ON ERROR IS SET
      BNE 9$ ; OTHERWISE NEVER BE ABLE TO SCOPE ETC.
      JSR PC,CKDROP ; DROP UNIT IF ALLOWED
9$: RTS PC ; RETURN ETC.

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 58
 CKDROP - CHECK IF UNIT SHOULD BE DROPPED

```

3354 .SBTTL CKDROP - CHECK IF UNIT SHOULD BE DROPPED
3355 ;+
3356 ; CHECK IF UNIT SHOULD BE DROPPED
3357 ; -
3358 020136 010046 CKDROP: MOV RO,-(SP)
3359 020140 FORCERROR 1$,NOTSSR
3360 020150 RFLAGS RO
      020150 104421 TRAP CSRFLA
3361 020152 032700 000040 BIT #IDU,RO
3362 020156 001010 BNE 1$
3363 020160 011600 MOV (SP),RO
3364 020162 012737 177777 003060 MOV #-1,DUFLG
3365 020170 DODU UNITN
      020170 013700 002150 MOV UNITN RO
      020174 104451 TRAP CSDOFJ
3366 020176 DOCLN ;ABORT THE PASS
      020176 104444 TRAP CSDCLN
3367 020200 012600 1$: MOV (SP)+,RO
3368 020202 000207 RTS PC
3369
3370
3371
3372
3373 .SBTTL CONFIG - DETERMINE CONFIGURATION OF SYSTEM
3374 ;
3375 ; SUBROUTINE - DETERMINE CONFIGURATION OF TUBO SYSTEM.
3376 ;
3377 020204 CONFIG: JSR PC,SOFINIT
3378 020204 00473 016630 RTS PC
3379 020210 000207
3380
3381
3382

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 59
KTON,KTOFF - ENABLE/DISABLE MEMORY MANAGEMENT

```

3384 .SBTTL KTON,KTOFF - ENABLE/DISABLE MEMORY MANAGEMENT
3385 ;
3386 ; SUBROUTINE - ENABLE MEM MGT.
3387 ;
3388 020212 005737 003100 KTON: TST KTFLG ; GOT KI?
3389 020216 001403 BEQ 1$ ; NO.
3390 020220 012737 C00001 177572 MOV #1,SRO ; YES. ENABLE KT11.
3391 020226 000207 1$: RTS PC
3392
3393
3394
3395 ;
3396 ; SUBROUTINE - DISABLE MEM MGT.
3397 ;
3398 020230 005737 003100 KTOFF: TST KTFLG ; GOT KT11?
3399 020234 001405 BEQ 1$ ; NO.
3400 020236 000240 NOP
3401 020240 000240 NOP
3402 020242 012737 000000 177572 MOV #0,SRO ; DISABLE KT.
3403 020250 000207 1$: RTS PC
3404
3405

```

CZTUXAO TUBO FRONT END PRT B
SETMAP - SETUP PAR6 MAPPING

MACRO M1200 29-MAR-83 13:32 PAGE 60

.SBTTL SETMAP - SETUP PAR6 MAPPING

3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446

:+

: THIS ROUTINE SETS UP KERNEL PAR6 TP HANDLE
: AN 18 BIT ADDRESS. THE OFFSET INTO THE PAGE
: IS RETURNED BIASED TO PAR6.

: INPUTS:

RO HIGH ORDER ADDRESS BITS
R1 LOW ORDER ADDRESS BITS

: OUTPUTS:

RO 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 RO ;CONVERT WORD ADDRESS TO 32W BLOCKS
ROR R1 ;MAKE IT DOUBLE PRECISION
.ENDR
BIC #177,R1 ;ALINE FOR LOWER 4K BOUNDARY
CMP R1,KTF LG ;HIGHER THAN EXISTING MEMORY?
BHS 10\$;BR IF YES
MOV R1,@#KIPAR6 ;SETUP MAPPING REGISTER PAR6
BIC #160000,R2 ;SETUP DISPLACEMENT IN PAGE
ADD #140000,R2 ;ADD IN PAR6 BIAS
MOV R2,RO ;RETURN IN RO
SEC ;SET SUCCESS
BR 15\$;
10\$: CLC ;SET FAILURE
15\$: RTS PC ;RETURN

020252
020252 005737 003100
020262 001433
020264 010102
000006
020316 042701 000177
020322 020137 003100
020326 103011
020330 010137 172354
020334 042702 160000
020340 062702 140000
020344 010200
020346 000261
020350 000401
020352 000241
020354 000207

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 61
 FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN

3448
 3449
 3450
 3451
 3452
 3453
 3454
 3455
 3456
 3457
 3458
 3459
 3460
 3461
 3462
 3463
 3464
 3465
 3466
 3467
 3468
 3469
 3470
 3471
 3472
 3473
 3474
 3475
 3476
 3477
 3478
 3479
 3480
 3481
 3482
 3483
 3484
 3485
 3486
 3487
 3488
 3489
 3490
 3491
 3492
 3493
 3494

020356
 020356 004737 020230
 020366 010003
 020370 013701 003072
 020374 013702 003074
 020400 010321
 020402 005302
 020404 003375
 020406 005737 003100
 020412 001452
 020414 004737 020212
 020420 005000
 020422 013701 003104
 000006
 020472 004737 020252
 020476 010320
 020500 020027 160000
 020504 103774
 020506 162700 020000
 020512 062737 000200 172354
 020520 023737 172354 003100
 020526 001402
 020530 000137 020476
 020534 004737 020230
 020540 000207

```

.SBTTL FILLMEM - FILL MEMORY WITH BACKGROUND PATTERN
+
F'LL MEMORY WITH A BACKGROUND PATTERN
:
: INPUTS:
:
:   RO = BACKGROUND PATTERN
:   FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
:   KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
:
: OUTPUTS:
:
:   NONE
:
-
FILLMEM:
: SAVREG ;SAVE R1-R5 UNTIL NEXT RETURN
: JSR PC,KTOFF ;DISABLE KT.
: MOV RO,R3 ;COPY TEST PATTERN
: MOV FREE,R1 ;GET FIRST FREE LOCATION
: MOV FRESIZ,R2 ;SIZE OF FREE SPACE BELOW 28K.
10$: MOV R3,(R1)+ ;STORE A BACKGROUND WORD
: DEC R2 ;DONE ALL MEMORY IN FREE SPACE?
: BGT 10$ ;BR IF NO
: TST KTFLG ; GOT KT?
: BEQ 55$ ; NO. GET OUT.
: JSR PC,KTON ; YES. ENABLE KT.
: CLR RO ;HIGH ORDER ADDRESS START
: MOV PST32W,R1 ;GET >28K START ADDRESS (IN 32W BLOCKS)
: .REPT 6
: CLC ;CLEAR C BIT
: ROL R1 ;CONVERT BLOCKS TO WORDS
: ROL RO ;MAKE IT DOUBLE PRECISION
: .ENDR
30$: JSR PC,SETMAP ;SETUP PAR6 MAPPING REGISTER
: MOV R3,(R0)+ ;STORE TEST PATTERN IN >28K ADDRESS
: CMP RO,#160000 ;END OF PAR6 MAPPING AREA?
: BLO 30$ ;BR IF NO
: SUB #20000,RO ;BACKUP INTO PAR6 MAPPING BEGIN
: ADD #200,@#KIPAR6 ;POINT TO NEXT 4K BLOCK ^28K.
: CMP @#KIPAR6,KTFLG ;END OF MEMORY?
: BEQ 50$ ;BR IF YES
: JMP 30$ ;KEEP GOING ON ETC.
50$: JSR PC,KTOFF ; DISABLE KT.
55$: RTS PC
    
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 62
 CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN

```

3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518 020542
3519 020542
3520 020546 010003
3521 020550 004737 020230
3522 020554 013701 003072
3523 020560 013702 003074
3524 020564 020311
3525 020566 001411
3526 020570 010137 002204
3527 020574 005037 002202
3528 020600 010337 002176
3529 020604 011137 002200
3530 020610 000474
3531 020612 005721
3532 020614 005302
3533 020616 003362
3534 020620 005737 003100
3535 020624 001472
3536 020626 004737 020212
3537 020632 005000
3538 020634 013701 003104
3539 000006
3540
3541
3542
3543 020670 042701 000177
3544 020674 010046
3545 020676 010146
3546 020700 004737 020252
3547 020704 010004
3548 020706 012601
3549 020710 012600
3550 020712 020314
3551 020714 001411
3552 020716 010037 002202
    
```

```

.SBTTL CMPMEM - COMPARE MEMORY TO BACKGROUND PATTERN
*
COMPARE MEMORY WITH A BACKGROUND PATTERN
INPUTS:
    RO = BACKGROUND PATTERN
    FREE = FIRST LOCATION AVAILABLE TO DIAGNOSTIC
    KTFLG = SET TO HIGHEST MEMORY LOCATION IF > 28K.
OUTPUTS:
    CARRY - SET IF NO ERROR
    CARRY - CLR IF ERROR
IMPLICIT OUTPUTS:
    ERRHI - ERROR HIGH ADDRESS
    ERRLO - ERROR LOW ADDRESS
    EXPD  - EXPECTED DATA
    RECV  - RECEIVED DATA
CMPMEM:
    SAVREG                ;SAVE R1-R5 UNTIL NEXT RETURN
    MOV RO,R3              ;COPY TEST PATTERN
    JSR PC,KTOFF           ;DISABLE KT.
    MOV FREE,R1            ;GET FIRST FREE LOCATION
    MOV FRESIZ,R2         ;SIZE OF FREE SPACE BELOW 28K.
10$: CMP R3,(R1)          ;FREE SPACE LOCATION EQUAL TO EXPD?
    BEQ 15$               ;BR IF YES
    MOV R1,ERRLO          ;SAVE ADDRESS IN ERROR
    CLR ERRHI             ;NO HIGH ADDRESS
    MOV R3,EXPD           ;SAVE EXPD FOR ERROR REPORT
    MOV (R1),RECV        ;SAVE RECV FOR ERROR REPORT
    BR 50$
15$: TST (R1)+           ;POINT TO NEXT ADDRESS
    DEC R2                ;DONE ALL MEMORY IN FREE SPACE?
    BGT 10$              ;BR IF NO
    TST KTFLG             ; GOT KT?
    BEQ 55$              ; NO. GET OUT.
    JSR PC,KTON           ; YES. ENABLE KT.
    CLR RO                ;HIGH ORDER ADDRESS START
    MOV PST32W,R1        ;GET >28K START ADDRESS (IN 32W BLOCKS)
    .REPT 6
    ROL R1                ;CONVERT BLOCKS TO WORDS
    ROL R0                ;MAKE IT DOUBLE PRECISION
    .ENDR
    BIC #177,R1           ;ALINE 4K BOUNDARY
    MOV RO,-(SP)          ;SAVE HIGH ORDER
    MOV R1,-(SP)          ;SAVE LOW ORDER
    JSR PC,SETMAP        ;SETUP PAR6 MAPPING REGISTER
    MOV RO,R4            ;COPY ADDRESS BIASED TO PAR6
    MOV (SP)+,R1         ;RESTORE LOW ORDER IN NON PAR6 FORMAT
    MOV (SP)+,R0         ;RESTORE HIGH ORDER IN NON PAR6 FORMAT
30$: CMP R3,(R4)         ;ABOVE 28K LOCATION EQUAL EXPD?
    BEQ 32$              ;BR IF YES
    MOV RO,ERRHI         ;SAVE HIGH ORDER IN ERROR
    
```


CZTUXAO TUBO FRONT END PRT B
REGSAV - SAVE R1-R5 ON STACK

MACRO M1200 29-MAR-83 13:32 PAGE 63

3574		
3575		
3576		
3577		
3578		
3579		
3580		
3581		
3582		
3583		
3584		
3585		
3586		
3587		
3588		
3589		
3590		
3591		
3592		
3593		
3594	021022	
3595	021022	
	021022	104422
3596	021024	010446
3597	021026	010346
3598	021030	010246
3599	021032	010146
3600	021034	010546
3601	021036	016605
3602	021042	004736
3603	021044	012601
3604	021046	012602
3605	021050	012603
3606	021052	012604
3607	021054	012605
3608	021056	
	021056	104422
3609	021060	000207
3610		

```

.SBTTL REGSAV - SAVE R1-R5 ON STACK
:
:ROUTINE TO
:SAVE R1 THROUGH R5 ON THE STACK
:
:CALLING SEQUENCE:
:
:       JSR      R5,REGSAV
:
:THIS IS A COOROUTINE WHICH TRANSFER CONTROL BACK TO
:THE CALLING ROUTINE. AT THE END OF THE CALLING ROUTINE,
:THE RTS PC RETURNS CONTROL TO THIS ROUTINE TO RESTORE
:REGISTERS.
:
:THIS ROUTINE SHOULD ONLY BE CALLED FROM ROUTINES WHICH ARE
:CALLED VIA A JSR PC INSTRUCTION
:
:-
REGSAV:
      BREAK
      TRAP      C$BRK           ;LOOK FOR CNTL C
      MOV       R4,-(SP)
      MOV       R3,-(SP)
      MOV       R2,-(SP)
      MOV       R1,-(SP)
      MOV       R5,-(SP)
      MOV       10.(SP),R5
      JSR       PC,@(SP)+
      MOV       (SP)+,R1
      MOV       (SP)+,R2
      MOV       (SP)+,R3
      MOV       (SP)+,R4
      MOV       (SP)+,R5
      BREAK
      TRAP      C$BRK
      RTS       PC           ;LOOK FOR CNTL C

```

000012

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 64
 GETPAT - GET 8 BIT PATTERN FROM OPERATOR

```

3612          .SBTTL  GETPAT  - GET 8 BIT PATTERN FROM OPERATOR
3613          :+
3614          :ROUTINE TO REQUEST AN 8 BIT DATA PATTERN FROM THE OPERATOR
3615          :INPUTS:
3616          :
3617          :      NONE.
3618          :
3619          :OUTPUTS:
3620          :
3621          :      RO      OCTAL NUMBER FROM THE OPERATOR
3622          :
3623          :CALLING SEQUENCE:
3624          :
3625          :      JSR      PC,GETPAT
3626          :
3627          :-
3628
3629
3630
3631 021062     GETPAT::
3632 021062     1$:      SAVREG          ;SAVE THE GENERAL REGISTERS
3633 021066     GMANID  DATASC,PATDAT,0,377,0,377,NO
3634 021066     TRAP    CSGMAN
3635 021070     BR      10000$
3636 021072     .WORD  PATDAT
3637 021074     .WORD  TSCODE
3638 021076     .WORD  DATASC
3639 021100     .WORD  377
3640 021102     .WORD  TSLOLIM
3641 021104     .WORD  TSHILIM
3642 021106     10000$:
3643 021106     BNCOMPLETE      1$      ;RETRY IF ERROR
3644 021110     BCC      1$
3645 021114     MOV      PATDAT,RO      ;DATA PATTERN FROM OPERATOR
3646 021116     RTS      PC            ;RETURN TO CALLER
3647
3648          :+
3649          :LOCAL DATA AREA
3650          :-
3651
3652 021116     000000     PATDAT: .WORD  0      ;TEMPORARY STORAGE FOR DATA
3653 021120     105      116      124  DATASC: .ASCIZ 'ENTER DATA PATTERN'
3654
  
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 65
 GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE

```

3646 .SBTTL GETSEL - ISSUE MENU AND GET OPERATOR RESPONSE
3647
3648 :+
3649 :ROUTINE TO ISSUE A MENU AND GET
3650 :THE OPERATOR'S RESPONSE.
3651 :INPUTS:
3652 : R0 ADDRESS OF ASCIZ STRING OF MENU
3653 : R1 MAXIMUM ALLOWABLE OPERATOR RESPONSE
3654 :OUTPUTS:
3655 : R0 NUMBER OF THE OPERATOR'S SELECTION
3656 GETSEL::
3657 SAVREG ;SAVE GENERAL REGISTERS
3658 MOV R0,R2 ;SAVE THE MENU ADDRESS
3659 1$: MOV R2,R3 ;START OF MENU STRING
3660 2$: TST (R3) ;END OF ASCII ?
3661 BEQ 3$ ;BRANCH IF ALL LINES DISPLAYED
3662 PRINTF #SELASC,(R3)+ ;DISPLAY THE MENU
      MOV (R3)+,-(SP)
      MOV #SELASC,-(SP)
      MOV #2,-(SP)
      MOV SP,R0
      TRAP CSPNTF
      ADD #6,SP
      BR 2$
3663 3$: GMANID MENASC,MENRES,D,-1,0,-1,NO
3664 TRAP CSGMAN
      BR 10001$
      .WORD MENRES
      .WORD TSCODE
      .WORD MENASC
      .WORD -1
      .WORD TSLOLIM
      .WORD TSHILIM
10001$:
3665 BNCOMPLETE 1$ ;RETRY IF ERROR
      BCC 1$
3666 MOV MENRES,R0 ;GET THE OPERATOR'S REPLY
3667 CMP R0,R1 ;COMPARE TO MAXIMUM ALLOWED
3668 BLOS 5$ ;BRANCH IF OK
3669 PRINTF #MENERR ;DISPLAY ERROR MESSAGE
      MOV #MENERR,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP CSPNTF
      ADD #4,SP
      BR 1$ ;RETRY
3670 5$: RTS PC ;RETURN TO CALLER
3671 045 MENERR: .ASCIZ 'XNZA *** Menu Selection Too Large ***'
3672 045 SELASC: .ASCIZ 'XNXT'
3673 164 MENASC: .ASCIZ 'Enter Menu Selection: '
3674 .EVEN
3675 MENRES: .WORD 0
3676

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 66
CHKMAN - CHECK MANUAL INTERVENTION LEGALITY

3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699

```

.SBTTL  CHKMAN - CHECK MANUAL INTERVENTION LEGALITY
:ROUTINE TO TEST FOR MANUAL INTERVENTION LEGALITY.
:INPUT:
      NONE.
:OUTPUT:
      CARRY  0      MANUAL INTERVENTION NOT ALLOWED
      CARRY  1      MANUAL INTERVENTION IS OK
:SIDE EFFECTS:
      A MESSAGE IS DISPLAYED WARNING THAT TEST IS
      NOT EXECUTED IF MANUAL INTERVENTION IS NOT
      ALLOWED.

```

```

3700 021366
3701 021366
3702 021372 104450
3703 021374 103411
3704 021376 012746 021422
      021402 012746 000001
      021406 010600
      021410 104417
      021412 062706 000004
3705 021416 000241
3706 021420 000207
3707
3708 021422 045 116 045
3709

```

```

CHKMAN::
      SAVREG          ;SAVE THE REGISTERS
      MANUAL          ;SEE IF MANUAL INTERVENTION OK
      TRAP  CSMANI
      BCOMPLETE 1$   ;BRANCH IF ALLOWED
      BCS  1$
      PRINTF #NOMAN ;PRINT THE WARNING MESSAGE
      MOV  #NOMAN,-(SP)
      MOV  #1,-(SP)
      MOV  SP,R0
      TRAP CSPNTF
      ADD  #4,SP
      CLC          ;CLEAR CARRY FOR ERROR
      RTS  PC      ;RETURN
1$:
NOMAN: .ASCIZ 'XNZA *** Manual Intervention not Allowed - Test Aborted ***'
      .even

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 67
 ENVIRM - SETUP FREE DIAGNOSTIC SPACE

```

3711                                     .SBTTL ENVIRM - SETUP FREE DIAGNOSTIC SPACE
3712                                     :
3713                                     : SUBROUTINE TO SET-UP VARIOUS ENVIRONMENTAL PARAMETERS.
3714                                     :
3715 021516                               ENVIRM: MEMORY R0
      021516 104431                       TRAP      CSMEM
3716 021520 010037 003072                 MOV       R0,FREE      ; GET 1ST FREE ADDRESS...
3717 021524 062737 000002 003072         ADD       #2,FREE
3718 021532 011037 003074                 MOV       (R0),FRESIZ ; ...AND WORD COUNT.
3719 021536 162737 000004 003074         SUB       #4,FRESIZ
3720 021544 013702 002012                 MOV       LBUNIT,R2   ; GET NUMBER OF UNITS
3721 021550 162737 000007 003074 108:   SUB       #7,FRESIZ   ; TAKE AWAY 7 WORDS PER UNIT
3722 021556 005302                       DEC       R2
3723 021560 001373                       BNE      108
3724 021562 013700 003072                 MOV       FREE,R0     ;GET FIRST FREE ADDRESS
3725 021566 063700 003074                 ADD       FRESIZ,R0   ;POINT TO LAST FREE ADDRESS
3726 021572 162700 000002                 SUB       #2,R0       ;BACKUP 1 WORD
3727 021576 010037 003076                 MOV       R0,FREEHI   ;STORE LAST FREE ADDRESS
3728 021602 000207                       RTS      PC           ;RETURN
3729
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 68
KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS

```

3731                                     .SBTTL KTINIT - SETUP KT11 MEMORY MANAGEMENT REGISTERS
3732                                     :+
3733                                     :ROUTINE TJ INIT *T-11
3734                                     :-
3735
3736
3737
3738 021604                               KTINIT:
3739 021604 005037 003100                 CLR      KTFLG                : INIT >28K MEMORY FLAG
3740 021610 005037 003102                 CLR      KTENABLE           : INIT TEST >28K FLAG
3741 021614 023727 002120 001577         CMP      LSHIME,#1577       : GOT ENOUGH MEMORY (>28K)?
3742 021622 101444                       BLOS     9S                 : NO.
3743 021624 013700 000004                 MOV      @ERRVEC,R0        : SAVE OLD ERR VEC PTR.
3744 021630 012737 021722 000004         MOV      #28,@ERRVEC       : SET ERR VEC PTR.
3745 021636 005737 177572                 TST     @SRO               : GOT KT11?
3746 021642 000240                       NOP                        : (TRAP IF NO).
3747 021644 013737 002120 003100         MOV      LSHIME,KTFLG     : YES. SET KT FLAG.
3748 021652 042737 000177 003100         BIC     #177,KTFLG        :
3749 021660 010037 000004                 MOV      R0,@ERRVEC       : RESTORE OLD ERR VEC PTR.
3750 021664 005000                       CLR      R0                : R0 = AR DATA.
3751 021666 012701 172340                 MOV      #KIPAR0,R1       : R1 = KI REGS PTR.
3752 021672 012761 077406 177740 1S:    MOV      #77406,-40(R1)    : SET DESCRIPTOR REG.
3753 021700 010021                       MOV      R0,(R1)+         : SET KIPAR REG.
3754 021702 062700 000200                 ADD     #200,R0           : BUMP AR DATA BY '4K'.
3755 021706 020027 002000                 CMP     R0,#2000          : AT 'I/O'?
3756 021712 001367                       BNE     1S                 : NO.
3757 021714 012741 177600                 MOV     #177600,-(R1)     : YES. SET KTPAR7 FOR I/O.
3758 021720 000405                       BR      9S                 :
3759
3760 021722 012716 021730                 2S:   MOV     #6S,(SP)      : SET UP RETURN
3761 021726 000002                       RTI                          : RTI TO NEXT LOCATION
3762
3763 021730 010037 000004                 6S:   MOV     R0,@ERRVEC   : RESTORE OLD ERR VEC PTR.
3764
3765 021734 000207                 9S:   RTS     PC
3780
3781
3782 021736                               BGNPROT
3783 021736 177777 177777 177777         LSPROT:: .WORD -1,-1,-1,-1 :NO DEVICE PROTECTION REQUIRED.
3784 021746                               ENDPROT

```


3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825
3826

021746
021746
021746
021746 012737 005755 002146
021754 005037 003106
021760 005037 003102
021764 005037 002246
021770
021770 012700 000036
021774 104447
021776
021776 103023
022000 023737 002150 002012
022006 103073
022010 005737 003060
022014 100475
022016 013701 002150
022022 006301
022024 005761 003130
022030 001521
022032 032761 040000 003130
022040 001063
022042
022042 104432
022044 000430
022046
022046 012700 000035
022052 104447
022054
022054 103055
022056
022056 012700 000040
022062 104447
022064
022064 103404
022066
022066 012700 000037
022072 104447
022074
022074 103034
022076
022076
022076 104433
022100 005037 002162
022104 005037 002170

```

.SBTTL INITIALIZE SECTION
:++
:THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
:AT THE BEGINNING OF EACH PASS.
:
:IF "START" OR "RESTART", SET QUICK-PASS FLAG AND BUS-INIT.
:IF "CONTINUE", NOTHING IS REQUIRED.
:
:--
:
:INSERT TEMPORARY JUMP TO ODT
:
:
      BGNINIT
LSINIT::
40$:
      MOV      #EPR1,EPR1SW ;SET UP PRIMARY MESSAGE FOR REPLACEMENT
      CLR      SIFLAG       ;CLEAR "SOFT INIT" FLAG
      CLR      KTENABLE     ;CLEAR TEST ABOVE 2BK FLAG
      CLR      RAMSIZ       ;CLEAR RAM SIZE FOR RAMERR ROUTINE
      REDEF    #EF.CONTINUE
      MOV      #EF.CONTINUE,R0
      TRAP    CSREFG
      BNCOMPLETE 1$
      BCC     1$
      CMP     UNITN,LSUNIT ;UNIT IN RANGE?
      BHIS   4$           ;BR IF NO.
      TST    DUFLG       ;DROPPED UNIT?
      BMI    NXTU        ;BR IF YES
      MOV    UNITN,R1
      ASL   R1
      TST   ERTABL(R1)
      BEQ   SETU
      BIT   #BIT14,ERTABL(R1) ;DROPPED?
      BNE   NXTU
      EXIT  INIT          ;DO NOTHING IF "CONTINUE".
      TRAP CSEXIT
      .WORD L10030-.
1$:
      REDEF   #EF.NEW
      MOV    #EF.NEW,R0
      TRAP  CSREFG
      BNCOMPLETE NXTU    ;TAKE NEXT UNIT IF NOT NEW PASS.
      BCC   NXTU
      REDEF #EF.START
      MOV  #EF.START,R0
      TRAP CSREFG
      BNCOMPLETE 2$
      BCS  2$
      REDEF #EF.RESTART
      MOV  #EF.RESTART,R0
      TRAP CSREFG
      BNCOMPLETE 31$
      BCC  31$
2$:
      BRESET
      TRAP CSRESET
      CLR  TSTCNT ;NUMBER OF TESTS RUN IN PASS
      CLR  FATFLG ;RESET FLAG TO ZERO "FATAL ERRORS"

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 70-1
INITIALIZE SECTION

```

3827 022110 000406          BR      19$          ;BR, IF THE FLAG IS NOT SET
3828                                     ;(NO DEBUGGER ETC.)
3829 022112 012746 000340    MOV     #340,-(SP)
3830 022116 012746 022132    MOV     #20$,-(SP)          ;RETURN TO DEBUGGER
3831 022122 000137 062046    JMP     0.ODI              ;:ENTER THE DEBUGGER
3832 022126 005037 003332    19$:   CLR     SKIPT          ;CLEAR THE SUBTEST "SKIPPER"
3833 022132                                     20$:
3834 022132 012737 177777 002152    MOV     #-1,QVP            ;...QUICK VERIFY...
3835 022140 004737 021516    JSR     PC,ENVIRN          ;SET ENVIRONMENT.
3836 022144 004737 021604    JSR     PC,KTINIT         ;INITIALIZE KT MEMORY MANAGEMENT
3837 022150 012700 003130    MOV     #ERTABL,RO
3838 022154 005020 003330    30$:   CLR     (RO)+          ;CLEAR THE ERROR TABLE
3839 022156 020027 003330    CMP     RO,#ERTABE
3840 022162 103774          BLO     30$
3841 022164 000404          BR      4$
3842 022166 005037 002152    31$:   CLR     QVP
3843 022172 000137 022242    JMP     PASRPT            ;GO REPORT THE STATUS
3844
3845 022176                                     4$:
3846 022176 012737 177777 002150    NEWPAS: MOV    #-1,UNITN          ;INIT UNIT NUMBER...
3847 022204 005037 002166          CLR     DEVCNT            ;CLEAR COUNT OF DEVICES RUNNING
3848 022210                                     NXTU:
3849 022212 104422          TRAP    CSBRK
3850 022216 005237 002150          INC     UNITN              ;...AND SET NEXT UNIT NUMBER.
3851 022224 103423          CMP     UNITN,LSUNIT
3852 022226 012737 177777 003060    BLO     SETU
3853 022234 000401          MOV     #-1,DUFLG
3854 022236          BR      11$
3855 022240 104444          DOCLN  CSDECLN            ;ABORT, NO MORE UNITS.
3856 022242          TRAP    CSDCLN
3857 022242 023727 002012 000001    11$:   NOP
3858 022250 101752          PASRPT:
3859 022252 005737 002166          CMP     LSUNIT,#1          ;HOW MANY UNITS SELECTED?
3860 022256 001747          BLOS   NEWPAS              ;BR IF ONLY 1
3861 022260          TST    DEVCNT              ;ARE ANY STILL RUNNING?
3862 022262 032700 000100          BEQ    NEWPAS              ;BR IF NO
3863 022266 001343          RFLAGS RO
3864          TRAP    CSRFLA
3865 022270          BIT   #ISR,RC              ;SHOULD WE PRINT STATISTICS
3866 022272 104424 000741          BNE    NEWPAS              ;BR IF NO
3867 022274          DORPT CSDRPT
3868          TRAP    NEWPAS
3869 022274          BR      10$
3870 022274 013700 002150    SETU:  GPHARD UNITN,RO          ;GET UNIT N P-TABLE POINTER.
3871 022300 104442          MOV     UNITN,RO
3872 022302 103342          TRAP    C$GPHRD
3873 022304 005037 003060    BNCOMPLETE NXTU          ;BR IF UNIT NOT AVAILABLE.
3874 022310 005237 002166    BCC    NXTU
3875 022314 012001          CLR     DUFLG              ;CLEAR "DROPPED" FLAG.
3876 022316 010137 002154    INC     DEVCNT
3877          MOV     (RO)+,R1          ;GET 1ST REGISTER ADDRESS.
3878          MOV     R1,CSRADDR      ;ADDRESS OF REGISTERS OF UNIT UNDER TEST
3879          MOV     (RO)+,R1          ;GET VECTOR ADDRESS.

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 70-2
INITIALIZE SECTION

```

3877 022324 011002          MOV      (R0),R2          ;GET INTERRUPT PRIORITY
3878 022326 010237 002160  MOV      R2,IPRI        ;SET INYERRUPT PRIORITY.
3879 022332 010137 002156  MOV      R1,IVEC        ;SET INTERRUPT VECTOR POINTER...
3880 022336 012721 017052  MOV      #!NTR,(R1)+    ;...VECTOR...
3881 022342 010221          MOV      R2,(R1)+      ;...AND PRIORITY.
3882
3883 022344          1$:
3884          :          TST      QVP          ;1ST PASS ??
3885          :          BEQ      SS          ;NO, SKIP THE PASS 1 STUFF.
3886
3887          :
3888          :          ;1ST PASS, CHECK THAT DEVICE ADDRESSES ARE VALID, AND
3889          :          ;THAT THE DISPLAY STATUS IS PROPERLY INITIALIZED.
3890          :
3891 022344 013701 002150          MOV      UNITN,R1
3892 022350 006301          ASL      R1
3893 022352 052761 100000 003130  BIS      #BIT15,ERTABL(R1) ;SAY DEVICE RUNNING
3894 022360 005037 005232          CLR      EXTA          ;CLEAR ERROR EXTENSION FLAG.
3895 022364 023727 002012 000001  CMP      LSUNIT,#1     ;ARE WE TESTING MULTIPLE UNITS?
3896 022372 101416          BLOS    10$           ;BR IF NO.
3897 022374          RFLAGS  RO          ;YES -- GET OPERATOR FLAGS.
          022374 104421          TRAP    CSRFLA
3898 022376 032700 001000          BIT      #PNT,RO      ;SHOULD WE PRINT UNIT #?
3899 022402 001412          BEQ     10$           ;BR IF NOT.
3900 022404          PRINTF #PUNIT,UNITN ;PRINT THE UNIT #
          022404 013746 002150          MOV      UNITN,-(SP)
          022410 012746 022476          MOV      #PUNIT,-(SP)
          022414 012746 000002          MOV      #2,-(SP)
          022420 010600          MOV      SP,RO
          022422 104417          TRAP    C$PNTF
          022424 062706 000006          ADD     #6,SP
3901 022430          10$:
3902 022430 005037 003062          CLR      NODEV
3903 022434 013701 002154          MOV      CSRADDR,R1   ;ADDRESS OF FIRST REGISTER
3904 022440 010102          MOV      R1,R2        ;START OF REGISTERS
3905 022442 062702 000000          ADD     #TSSR,R2     ;ADDRESS OF TSSR REGISTER
3906 022446 004737 017260          JSR     PC,XNXM       ;TEST BOTH CONTROLLER REGISTERS...
3907 022452 103005          BCC     2$           ;...AND BR IF ALL OK.
3908 022454 010137 003062          MOV     R1,NODEV     ;FLAG DEVICE AS NON-EXISTENT
3909 022460 012737 177777 003060  MOV     #-1,DUFLG    ;DROP THIS UNIT.
3910 022466          2$:
3911          :
3912          :          ;FINALLY, SET CPU PRIORITY AND WE'RE DONE.
3913          :
3914 022466          5$:          SETPRI  #PRI00         ;ENABLE INTERRUPTS.
          022466 012700 000000          MOV     #PRI00,RO
          022472 104441          TRAP   C$SPRI
3915 022474          ENDINIT
          022474          L10030:
          022474 104411          TRAP   C$INIT
3916
3917 022476          045      116      045  PUNIT: .ASCIZ /XNXNZA***** TESTING UNIT XD2ZA *****/
3918          .EVEN

```

LZTUXAO TUBO FRONT END PRT B
ADD AND DROP UNITS SECTIONS

MACRO M1200 29-MAR-83 13:32 PAGE 71

.SBTTL ADD AND DROP UNITS SECTIONS

3920
3921
3922
3923
3924
3925
3926
3927 022544
022544
3928 022544 010001
3929 022546 006301
3930 022550 052761 100000 003130
3931 022556 042761 040000 003130
3932 022564
022564 010046
022566 012746 022612
022572 012746 000002
022576 010600
022600 104417
022602 062706 000006
3933 022606
022606 000167
022610 000026
3934 022612 045 116 045 1\$:
3935
3936
3937 022640
022640
022640 104452
3938
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948
3949 022642
022642
3950 022642 012737 177777 003060
3951 022650 010001
3952 022652 006301
3953 022654 052761 140000 003130
3954 022662 000240 000240 000240
3955 022670
022670 010046
022672 012746 022716
022676 012746 000002
022702 010600
022704 104417
022706 062706 000006
3956 022712
022712 000167
022714 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
LSAU::
      MOV      RO,R1          ; GET UNIT TO BE ADDED (RO)
      ASL      R1              ; MAKE IT A WORD INDEX
      BIS      #10000,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     CSPNTF
      ADD      #6,SP
      EXIT     AU
      .WORD    JSJMP
      .WORD    L10031-2-.
      .ASCIZ   /XNZA UNIT XDZA ADDED/
      .EVEN

      ENDAU          ; UNUSED.
L10031:
      TRAP     CSAU

:++
: 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
LSDU::
      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     CSPNTF
      ADD      #6,SP
      EXIT     DU
      .WORD    JSJMP
      .WORD    L10032-2-.

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 71-1
 ADD AND DROP UNITS SECTIONS

3957	022716	045	116	045	1\$:	.ASCIZ /XNZA UNIT XDZA DROPPED/ .EVEN ENDDU	
3958							
3959	022746				L10032:	TRAP CS\$DU	
	022746	104453					
3960					:++		
3961					: AUTO-DROP CODE SECTION.		
3962					:--		
3963	022750				LSAUTO::	BGNAUTO	
	022750						
3964	022750	012703	000550		10\$:	MOV #360.,R3 ;ENOUGH TIME FOR 2400' REEL TO REWIND	
3965	022754	004737	017104			JSR PC,WAITF ;WAIT FOR SSR TO SET	
3966	022760	103420				BCS 20\$;LEAVE WHEN SSR IS SET	
3967	022762					DELAY 250. ;WAIT FOR .25 SECONDS	
	022762	012727	000372			MOV #250.,(PC)+	
	022766	000000				.WORD 0	
	022770	013727	002116			MOV LSDLY,(PC)+	
	022774	000000				.WORD 0	
	022776	005367	177772			DEC -6(PC)	
	023002	001375				BNE -.4	
	023004	005367	177756			DEC -22(PC)	
	023010	001367				BNE .-20	
3968	023012	005303				DEC R3 ;BUMP COUNTER DOWN	
3969	023014	001357				BNE 10\$;KEEP GOING	
3970	023016	004737	020136			JSR PC,CKDROP ;TRY AND DROP UNIT	
3971	023022				20\$:	ENDAUTO ;UNUSED.	
3972	023022				L10033:		
	023022	104461				TRAP CSAUTO	

.SBTTL CLEAN-UP AND REPORT CODING SECTIONS

3974
3975
3976
3977
3978
3979
3980
3981 023024
023024
3982 023024 005737 003060
3983 023030 100405
3984
3985
3986 023032 012765 000000 000000
3987 023040 004737 017104
3988 023044
3989 023044
023044
023044 104412
3990
3991
3992
3993
3994 023046
023046
3995 023046
023046 012746 023310
023052 012746 000001
023056 010600
023060 104416
023062 062706 000004
3996 023066 010246
3997 023070 010346
3998 023072 010446
3999 023074 012704 003130
4000 023100 005003
4001 023102 011402
4002 023104 001467
4003 023106 100066
4004 023110 032702 040000
4005 023114 001015
4006 023116 042702 170000
4007 023122
023122 010246
023124 010346
023126 012746 023345
023132 012746 000003
023136 010600
023140 104416
023142 062706 000010
4008 023146 000446
4009 023150 020227 160000
4010 023154 001012
4011 023156
023156 010346
023160 012746 023415
023164 012746 000002

```

:++
: THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS
: EXECUTED AT THE END OF EACH PASS (OR SUB-PASS).
: USE TO RETURN DEVICE UNDER TEST TO A NEUTRAL STATE.
:--
      BGNCLN
L$CLEAN::
      TST      DUFLG      ;'DROPPED' FLAG IS SET ON...
      BMI      1$        ;...AND GROSS CONTROLLER FAULT...
                          ;...DON'T TRY TO XCT CLEANUP CODE.
      MOV      #0,TSSR(R5) ;DO SOFT INIT
      JSR      PC,WAITF
1$:
2$:      ENDCLN
L10034:      TRAP      C$CLEAN

:++
: THE REPORT CODING SECTION CONTAINS THE
: 'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.
:--
      BGNRPT
LSRPT::
      PRINTS   #DEVSUM
      MOV      #DEVSUM,-(SP)
      MOV      #1,-(SP)
      MOV      SP,R0
      TRAP     C$PNTS
      ADD      #4,SP
      MOV      R2,-(SP)
      MOV      R3,-(SP)
      MOV      R4,-(SP)
      MOV      #ERTABL,R4      ; GET START OF ERROR TABLE.
      CLR      R3              ; CLEAR UNIT NUMBER
1$:      MOV      (R4),R2      ; GET ERROR TABLE ENTRY & TEST IT.
      BEQ      4$              ; ZERO IF UNIT NOT RUN
      BPL      4$
      BIT      #BIT14,R2      ; WAS UNIT DROPPED?
      BNE      2$              ; BR IF YES
      BIC      #^C7777,R2    ; GET ERROR COUNT FIELD
      PRINTS   #DEVONL,R3,R2 ; PRINT
      MOV      R2,-(SP)
      MOV      R3,-(SP)
      MOV      #DEVONL,-(SP)
      MOV      #3,-(SP)
      MOV      SP,R0
      TRAP     C$PNTS
      ADD      #10,SP
      BR       4$
2$:      CMP      R2,#160000   ; WAS UNIT NON-EXISTENT?
      BNE      3$              ; BR IF NO
      PRINTS   #DEVNKR,R3
      MOV      R3,-(SP)
      MOV      #DEVNKR,-(SP)
      MOV      #2,-(SP)

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 72-1
CLEAN-UP AND REPORT CODING SECTIONS

023170	010600			MOV	SP,R0		
023172	104416			TRAP	CSPNTS		
023174	062706	0C0006		AF)	#6,SP		
4012 023200	000431			BR	4\$		
4013 023202	020227	160001	3\$:	CMP	R2,#i60C01	:	WAS UNIT NOT READY AT STARTUP?
4014 023206	001012			BNE	30\$:	BR IF NO.
4015 023210				PRINTS	#DEVNRD,R3		
023210	010346			MOV	R3,-(SP)		
023212	012746	023477		MOV	#DEVNRD,-(SP)		
023216	012746	000002		MOV	#2,-(SP)		
023222	010600			MOV	SP,R0		
023224	104416			TRAP	CSPNTS		
023226	062706	000006		ADD	#6,SP		
4016 023232	000414			BR	4\$		
4017 023234	042702	170000	30\$:	BIC	#^C7777,R2		
4018 023240				PRINTS	#DEVDR0,R3,R2		
023240	010246			MOV	R2,-(SP)		
023242	010346			MOV	R3,-(SP)		
023244	012746	023560		MOV	#DEVDR0,-(SP)		
023250	012746	000003		MOV	#3,-(SP)		
023254	010600			MOV	SP,R0		
023256	104416			TRAP	CSPNTS		
023260	062706	000010		ADD	#10,SP		
4019 023264	062704	000002	4\$:	ADD	#2,R4		
4020 023270	005203			INC	R3		
4021 023272	020427	003330		CMP	R4,#ERTABE		
4022 023276	103701			BLO	1\$		
4023 023300	012604			MOV	(SP)+,R4		
4024 023302	012603			MOV	(SP)+,R3		
4025 023304	012602			MOV	(SP)+,R2		
4026 023306				ENDRPT		:	UNUSED.
023306			L10G35:				
023306	104425			TRAP	CSRPT		
4027							
4028							
4029 023310	045	116	045	DEVSUM:	.ASCIZ /%N%ADEVICE STATUS SUMMARY:%N/		
4030 023345	045	101	040	DEVONL:	.ASCIZ /%A UNIT %D3%A ONLINE, ERRORS = %D%N/		
4031 023415	045	101	040	DEVNXR:	.ASCIZ /%A UNIT %D3%A DROPPED, NON-EXISTENT REGISTER%N/		
4032 023477	045	101	040	DEVNRD:	.ASCIZ /%A UNIT %D3%A DROPPED, NOT READY AT STARTUP%N/		
4033 023560	045	101	040	DEVDR0:	.ASCIZ /%A UNIT %D3%A DROPPED, ERRORS = %D%N/		
4034				.EVEN			

4062
4063
4064
4065
4066
4067
4068
4069
4070
4071
4072
4073
4074
4075
4076
4077
4078
4079
4080
4081
4082
4083
4084 023630
023630
4085 023630 005037 002170
4086 023634 005037 003100
4091 023640 012700 031216
4092 023644 004737 017372
4093 023650 012737 000002 002164
4094 023656 004737 020230
4095 023662 005037 003102
4096
4097
4098
4099
4100
4101
4102
4103 023666
023666
023666 104402
4104 023670 012737 014544 033126
4105
4106
4107
4108
4109
4110
4111
4112 023676 004737 016630
4113 023702 103424
4114 023704
023704 012727 000250
023710 000000
023712 013727 002116
023716 000000
023720 005367 177772

.SBTTL TEST 1: FIFO EXERCISER

TEST DESCRIPTION:

This test uses the Write Subsystem Memory command to verify the controller's FIFO and associated status and control logic.

TEST STEPS:

REPEAT FOR LOOPCNT

BEGIN

- Do Subtest 1 - FIFO Initialize status test
- Do Subtest 2 - FIFO Write Single Byte test
- Do Subtest 3 - FIFO Write Multiple Bytes test
- Do Subtest 4 - FIFO Verify ILW Status test
- Do Subtest 5 - FIFO Input Ready test
- Do Subtest 6 - FIFO Verify Reset FIFO test

END

BGNTST

```

CLR FATFLG ;CLEAR FATAL ERROR FLAG
CLR KTF LG ;HOLD OF KT11
MOV #TST17ID,RO ;ASCII MESSAGE TO IDENTIFY TEST
JSR PC,TSTSETUP ;DO INITIAL TEST SETUP
MOV #2.,LOOPCNT ;PERFORM 2 ITERATIONS
JSR PC,KTOFF ;SHUT OFF MEMORY MANAGEMENT
CLR KTENABLE ;REALLY SHUT DOWN KT-11

```

THIS CODE CHECKS FOR THE DRIVE TO BE AT BOT VIA A REWIND COMMAND

BGNSUB

```

MOV #6500.,T17DLY ;SET UP LOOP COUNTER

```

```

*****
:ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF THERE'S AN ERROR
*****

```

```

SS: JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
BCS 108 ;BR IF INIT WAS OK
DELAY 250 ;DELAY A WHILE
MOV #250,(PC)+
.WORD 0
MOV LSDLY,(PC)+
.WORD 0
DEC -6(PC)

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 76-1
TEST 1: FIFO EXERCISER

```

023724 001375
023726 005367 177756
023732 001367
4115 023734 004737 020064 JSR PC,FATCHK ;INCREMENT AND CHECK FOR MORE THAN 25 ERRORS
4116 023740 016501 000000 MOV TSSR(R5),R1 ;CONTENTS OF THE TSSR REGISTER
4117 023744 ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
023744 104455 TRAP CSCRDF
023746 000144 .WORD 100
023750 003550 .WORD SFIERR
023752 011656 .WORD SFIMSG
4118 023754 10$:
4119
4120 023754 012704 032640 MOV #T17PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
4121
4122 :*****
4123 :WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
4124 :*****
4125
4126
4127
4128 023760 004737 010322 JSR PC,WRTCHR ;ISSUE WRITE CHARACTERISTICS
4129 023764 103407 BCS 15$ ;BR IF COMMAND ISSUED IS OK
4130 023766 004737 020064 JSR PC,FATCHK ;INCREMENT AND CHECK FOR MORE THAN 25 ERRORS
4131 023772 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4135 023774 ERRHRD ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
023774 104456 TRAP CSERHRD
023776 000145 .WORD 101
024000 004754 .WORD WRTMSG
024002 011656 .WORD SFIMSG
4136
4137
4138
4139 :*****
4140 :ISSUE A REWIND COMMAND TO THE SELECTED TAPE DRIVE
4141 :*****
4142
4143
4144
4145 024004 004737 010424 15$: JSR PC,REWIND ;CALL TAPE REWIND COMMAND
4146 024010 103412 BCS 30$ ;BR IF NO PROBLEM
4147 024012 005337 033126 DEC T17DLY ;DEC DELAY COUNTER
4148 024016 001327 BNE 5$ ;BR IF LOOP IS REQUIRED
4149 024020 010001 MOV R0,R1 ;SAVE TSSR
4150 024022 004737 020064 JSR PC,FATCHK ;INCREMENT AND CHECK FOR MORE THAN 25 ERRORS
4154 024026 ERRHRD ERRNO,T17RWN,PKTSSR ;REWIND NOT ACCEPTED
024026 104456 TRAP CSERHRD
024030 000146 .WORD 102
024032 031235 .WORD T17RWN
024034 011670 .WORD PKTSSR
4155 024036 30$: CKLOOP ;LOOP IF SELECTED
024036 104406 TRAP CSCLP1
4156 024040 104406
4157
4158 024042 000137 030762 JMP TMPEND ;TEMP JUMP OVER FIFO TESTS
4159 024046 T17LOOP:

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 77
 TEST 1: SUBTEST 1: FIFO INITIALIZE STATUS TEST

4161
4162
4163
4164
4165
4166
4167
4168
4169
4170
4171
4172
4173
4174
4175
4176
4177
4178
4179
4180
4181
4182
4183
4184
4185
4186
4187
4188 024046
4189 024046 004737 016630
4190 024052 103405
4191 024054 010001
4192 024056
024056 104455
024060 000146
024062 003550
024064 011656
4193
4194 024066 005037 002170
4195 024072 012704 032640
4196 024076 004737 010322
4197 024102 103407
4198 024104 010001
4199 024106
4200 024106
024106 104455
024110 000147
024112 031263
024114 011670
4201 024116 004737 020064
4202 024122
024122 104406
4203
4204
4205 024124 004737 032422
4206 024130 012704 033010
4207 024134 010465 177776
4208 024140 004737 017220

```
.SBTTL TEST 1: SUBTEST 1: FIFO INITIALIZE STATUS TEST
**
TEST 1: SUBTEST 1:
SUBTEST DESCRIPTION:
    This test verifies, by using the Read Status select code,
    that the FIFO status is in the correct initial state after
    the controller is initialized (Input Ready TRUE,
    Output Ready and Data In Miss FALSE). These status
    signals are checked by the controller's self-test
    sequence, so this subtest is actually more of a partial
    check of the Read Status function than the FIFO status.
TEST STEPS:
BEGIN
    Write to TSSR to soft initialize
    Do a WRITE CHARACTERISTICS to setup a message buffer
    Do a WRITE SUBSYSTEM Read Status
    If Input Ready NOT=1 Then Print Error
    If Output Ready NOT=0 Then Print Error
    If Data In Miss NOT=0 Then Print Error
END
--
:
:      Write to TSSR register to soft initialize the controller
5$:
    JSR    PC,SOFINIT           ;WRITE TO TSSR TO SOFT INITIALIZE
    BCS    10$                 ;BR IF SOFT INIT OKAY
    MOV    R0,R1               ;SAVE CONTENTS OF TSSR
    ERDF   ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
                                TRAP    CSERDF
                                .WORD   102
                                .WORD   SFIERR
                                .WORD   SFIMSG
:
:      Do a WRITE CHARACTERISTICS to setup a message buffer
10$:
    CLR    FATFLG              ;CLEAR FATAL ERROR FLAG
    MOV    #T17PACKET,R4      ;GET THE ADDRESS OF COMMAND PACKET
    JSR    PC,WRTCHR           ;DO WRITE CHARACTERISTICS COMMAND
    BCS    50$                 ;BR IF CARRY SET (GOOD RETURN)
    MOV    R0,R1               ;SAVE CONTENTS OF TSSR
    NEXT.ERRNO
42$:
    ERDF   ERRNO,T17SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                TRAP    CSERDF
                                .WORD   103
                                .WORD   T17SSR
                                .WORD   PKTSSR
:
50$:
    JSR    PC,FATCHK           ;INC AND CHECK FOR MORE THAN 25 ERRORS
    CKLOOP                          ;LOOP ON ERROR, IF FLAG SET
                                TRAP    CSCLP1
:
:      Do a Write Subsystem READ STATUS
:
    JSR    PC,T17SRD           ;SETUP PACKET FOR READ STATUS
    MOV    #T17PK2,R4         ;GET WRITE SUBSYSTEM COMMAND PACKET
    MOV    R4,TSDB(R5)        ;SET THE PACKET ADDRESS TO EXECUTE
    JSR    PC,CHKTSSR         ;WAIT FOR SSR TO SET
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 77-1
 TEST 1: SUBTEST 1: FIFO INITIALIZE STATUS TEST

```

4209 024144 103407          BCS      70$          ;BR IF CARRY SET (GOOD RETURN)
4210 024146 010001          MOV      RO,R1        ;SAVE CONTENTS OF TSSR
4211 024150                NEXT.ERRNO
4212 024150                ERRDF   E^RNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                                62$:
                                024150 104455          TRAP    CSERDF
                                024152 000150          .WORD  104
                                024154 031364          .WORD  T173SSR
                                024156 011670          .WORD  PKTSSR
4213 024160 004737 020064    JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
4214 024164                CKLOOP              ;LOOP ON ERROR, IF FLAG SET
                                024164 104406          TRAP    CSCLP1
4215 :                      ; Set WORDS 0-7 of expd message buffer = to rcv since not testing
4216 024166 004737 032604    JSR      PC,T17SETEXP ;SET WORDS 0-7 EXPD=RCV
4217 024172 012701 031012    MOV      #T17EXSTA,R1 ;GET EXPECTED READ STATUS
4218 024176 012702 032702    MOV      #T17BFSTA,R2 ;GET RCV READ STATUS
4219 024202 012221          MOV      (R2)+,(R1)+  ;SET EXPD WORD #8 = RCV TEMP
4220 024204 011211          MOV      (R2),(R1)    ;SET EXPD WORD #9 = RCV TEMP
4221 024206 052711 000020    BIS      #S2.INRDY,(R1) ;SET EXP INPUT READY= TRUE
4222 024212 042711 000040    BIC      #S2.OTRDY,(R1) ;SET EXP OUTPUT READY= FALSE
4223 024216 042711 000200    BIC      #S2.DIM,(R1)  ;SET EXP DATA IN MISS = FALSE
4224 :                      ; If Input Ready NOT=1 then Print Error
4225 :                      ; If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error
4226 024222 005000          CLR      RO          ;HIGH RCV ADDRESS FOR CKMSG2
4227 024224 012701 032662    MOV      #T17BFR,R1  ;LOW RCV ADDRESS FOR CKMSG2
4228 024230 012702 030772    MOV      #T17EXP,R2  ;EXPD ADDRESS
4229 024234 012703 000024    MOV      #20.,R3     ;NUMBER OF BYTES TO COMPARE
4230 024240 004737 011322    JSR      PC,CKMSG2   ;EXPD EQUAL RCV?
4231 024244 103404          BCS      90$          ;BR IF YES
4232 024246                NEXT.ERRNO
4233 024246                ERRHRD  ERRNO,T171CMP,MSGSTAT ;REPORT ERROR
                                024246 104456          TRAP    CSERHRD
                                024250 000151          .WORD  105
                                024252 031603          .WORD  T171CMP
                                024254 012172          .WORD  MSGSTAT
4234 024256                CKLOOP              ;LOOP ON ERROR, IF FLAG SET
                                024256 104406          TRAP    CSCLP1
4235 :                      ;
4236 024260                ENDSUB              ;////////// END SUBTEST ////////////
                                024260                L10037:
                                024260 104403          TRAP    CSesub
4237 :                      ;
4238 024262 005737 002170    TST      FATFLG      ;ANY FATAL ERRORS ?
4239 024266 001402          BEQ      160$        ;BRANCH IF NOT
4240 024270 004737 020136    JSR      PC,CKDROP   ;TRY TO DROP THE UNIT
4241 024274                160$:
4242

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 78
TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

4244
4245
4246
4247
4248
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295

.SBTTL TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

..**
: TEST 1: SUBTEST 2:

: SUBTEST DESCRIPTION:

: This subtest verifies the ability of the FIFO to correctly
: pass a single data byte from input to output. For each
: of 256 data values (0-377 octal) the following is done:
: 1. Initial FIFO status is checked
: 2. The Write FIFO function, specifying a count of
: one byte to be written is executed.
: 3. Read Status is executed and FIFO status is checked.
: 4. Read FIFO is executed and the data and final status
: is checked.

: TEST STEPS:

: BEGIN

: Write to TSSR to soft initialize
: Do a WRITE CHARACTERISTICS to setup a message buffer
: Do a Write Subsystem READ STATUS
: If Input Ready NOT=1 Then Print Error
: If Output Ready NOT=0 Then Print Error
: If Data In Miss NOT=0 Then Print Error

: REPEAT FOR DATA FROM 0 TO 377 OCTAL

: BEGIN

: Do a Write Subsystem WRITE NPR to set tape direction out
: Do a Write Subsystem WRITE FIFO with byte count equal to 1
: Do a Write Subsystem READ STATUS
: If Input Ready NOT=1 Then Print Error
: If Output Ready NOT=1 Then Print Error
: If Data In Miss NOT=0 Then Print Error
: Do Write Subsystem READ FIFO with byte count equal to 1
: If Data read from FIFO NOT= to Data sent Then Print Error
: Do a Write Subsystem READ STATUS
: If Input Ready NOT=1 Then Print Error
: If Output Ready NOT=0 Then Print Error
: If Data In Miss NOT=0 Then Print Error

: END
: END

:-- BGNSUB ://////////////// BEGIN SUBTEST //////////////////
: T1.2: TRAP CSBSUB

: Write to TSSR register to soft initialize the controller

: 5s:

: JSR PC,SOFINIT :WRITE TO TSSR TO SOFT INITIALIZE
: BCS 105 :BR IF SOFT INI, OKAY
: MOV R0,R1 :SAVE CONTENTS OF TSSR
: ERRDF ERRNO,SFIERR,SFIMSG :DEVICE FATAL DURING INIT
: TRAP CSERDF
: .WORD 105
: .WORD SF1ERR

024274
024274 104402
024276
024276 004737 016630
024302 103405
024304 010001
024306 104455
024310 000151
024312 003550

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 78-2
 TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

```

4337
4338
4339 024510 012737 000000 002256 : Repeat for DATA from 0 to 377
4340 024516 100$: MOV #0,DATA ;GET FIRST DATA
4341 : ;REPEAT LABEL
4342 024516 012700 000100 : Do a Write Subsystem WRITE NPR to set tape direction out
4343 024522 004737 032464 MOV #NP.OUI,R0 ;SET TAPE DIRECTION OUT
4344 024526 012704 033010 JSR PC,T17SNPR ;SETUP T17PK2 FOR WRITE NPR
4345 024532 010465 177776 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4346 024536 004737 017220 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4347 024542 103407 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4348 024544 010001 BCS 105$ ;BR IF CARRY SET (GOOD RETURN)
4349 024546 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4350 024546 102$: NEXT.ERRNO ERRDF ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      024546 104455 TRAP CSERDF
      024550 000155 .WORD 109
      024552 031431 .WORD T174SSR
      024554 011670 .WORD PKTSSR
4351 024556 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4352 024562 105$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      024562 104406 TRAP CSCLP1
4353 : Do a Write Subsystem WRITE FIFO with byte count equal to 1
4354 024564 012700 000001 MOV #1,R0 ;WRITE 1 BYTE
4355 024570 012701 002256 MOV #DATA,R1 ;FIFO WRITE DATA ADDRESS
4356 024574 004737 032510 JSR PC,T17WFIF ;SETUP T17PK2 FOR WRITE FIFO
4357 024600 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4358 024604 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4359 024610 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4360 024614 103407 BCS 110$ ;BR IF CARRY SET (GOOD RETURN)
4361 024616 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4362 024620 NEXT.ERRNO
4363 024620 107$: ERRDF ERRNO,T175SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      024620 104455 TRAP CSERDF
      024622 000156 .WORD 110
      024624 031474 .WORD T175SSR
      024626 011670 .WORD PKTSSR
4364 024630 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4365 024634 110$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      024634 104406 TRAP CSCLP1
4366
4367 : Do a Write Subsystem READ STATUS
4368 024636 004737 032422 JSR PC,T17SRD ;SETUP PACKET FOR READ STATUS
4369 024642 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4370 024646 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4371 024652 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4372 024656 103407 BCS 120$ ;BR IF CARRY SET (GOOD RETURN)
4373 024660 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4374 024662 NEXT.ERRNO
4375 024662 112$: ERRDF ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      024662 104455 TRAP CSERDF
      024664 000157 .WORD 111
      024666 031364 .WORD T173SSR
      024670 011670 .WORD PKTSSR
4376 024672 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4377 024676 120$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      024676 104406 TRAP CSCLP1
4378 : Set WORDS 0-7 of expd message buffer = to rcv since not testing

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 78-3
 TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

```

4379 024700 004737 032604 JSR PC,T17SETEXP ;SET WORDS 0-7 EXPD=RECV
4380 024704 012701 031012 MOV #T17EXSTA,R1 ;GET EXPECTED READ STATUS
4381 024710 012702 032702 MOV #T17BFSTA,R2 ;GET RECV READ STATUS
4382 024714 012221 MOV (#2)+,(R1)+ ;SET EXPD WORD #8 = RECV TEMP
4383 024716 011211 MOV (R2),(R1) ;SET EXPD WORD #9 = RECV TEMP
4384 024720 052711 000020 BIS #S2.INRDY,(R1) ;SET EXP INPUT READY= 1
4385 024724 052711 000040 BIS #S2.OTRDY,(R1) ;SET EXP OUTPUT READY= 1
4386 024730 042711 000200 BIC #S2.DIM,(R1) ;SET EXP DATA IN MISS = 0
4387 : If Input ReadyB NOT=1 then Print Error
4388 : If Output Ready NOT=1 or Data in Miss NOT=0 Then Print Error
4389 024734 005000 CLR RO ;HIGH RECV ADDRESS FOR CKMSG2
4390 024736 012701 032662 MOV #T17BFR,R1 ;LOW RECV ADDRESS FOR CKMSG2
4391 024742 012702 030772 MOV #T17EXP,R2 ;EXPD ADDRESS
4392 024746 012703 000024 MOV #20.,R3 ;NUMBER OF BYTES TO COMPARE
4393 024752 004737 011322 JSR PC,CKMSG2 ;EXPD EQUAL RECV?
4394 024756 103404 BCS 140$ ;BR IF YES
4395 024760 NEXT.ERRNO
4396 024760 132$: ERRHRD ERRNO,T173CMP,MSGSTAT ;REPORT ERROR
      024760 104456 TRAP CSERHRD
      024762 000160 .WORD 112
      024764 031761 .WORD T173CMP
      024766 012172 .WORD MSGSTAT
4397 024770 140$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      024770 104406 TRAP C$CLP1
4398 :
4399 : Do Write Subsystem READ FIFO with byte count equal to 1
4400 024772 012700 000001 MOV #1,RO ;SET READ BYTE COUNT
4401 024776 004737 032544 JSR PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
4402 025002 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4403 025006 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4404 025012 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4405 025016 103407 BCS 150$ ;BR IF CARRY SET (GOOD RETURN)
4406 025020 010001 MOV RO,R1 ;SAVE CONTENTS OF TSSR
4407 025022 NEXT.ERRNO
4408 025022 142$: ERRDF ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      025022 104455 TRAP CSERDF
      025024 000161 .WORD 113
      025026 031540 .WORD T176SSR
      025030 011670 .WORD PKTSSR
4409 025032 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4410 025036 150$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      025036 104406 TRAP C$CLP1
4411 : Set WORDS 0-7 of expd message buffer = to recv since not testing
4412 025040 004737 032604 JSR PC,T17SETEXP ;SET WORDS 0-7 EXPD=RECV
4413 025044 012701 031012 MOV #T17EXSTA,R1 ;GET EXPECTED READ STATUS
4414 025050 012702 032702 MOV #T17BFSTA,R2 ;GET RECV READ STATUS
4415 025054 013721 002256 MOV DATA,(R1)+ ;SET EXPD WORD #8 = COUNT DATA
4416 025060 011211 MOV (R2),(R1) ;SET EXPD WORD #9 = RECV (NOT TESTING)
4417 : If Data read from FIFO NOT= to Data sent Then Print Error
4418 : The data is in WORD #8 of the message buffer
4419 025062 005000 CLR RO ;HIGH RECV ADDRESS FOR CKMSG2
4420 025064 012701 032662 MOV #T17BFR,R1 ;LOW RECV ADDRESS FOR CKMSG2
4421 025070 012702 030772 MOV #T17EXP,R2 ;EXPD ADDRESS
4422 025074 012703 000022 MOV #18.,R3 ;NUMBER OF BYTES TO COMPARE
4423 025100 004737 011322 JSR PC,CKMSG2 ;EXPD EQUAL RECV?
4424 025104 103404 BCS 160$ ;BR IF YES
4425 025106 NEXT.ERRNO

```

C7TUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 78-4
 TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

```

4426 025106          152$:  ERRHRD  ERRNO,T172CMP,MSGSUB  ;REPORT ERROR
      025106 104456
      025110 000162
      025112 031665
      025114 013564
      025116 104406
4427 025116          160$:  CKLOOP                ;LOOP ON ERROR, IF FLAG SET
      025116 104406
4429 025116          ;
4430 025120 004737 032422
4431 025124 012704 033010
4432 025130 010465 177776
4433 025134 004737 017220
4434 025140 103407
4435 025142 010001
4436 025144
4437 025144          162$:  ERRDF  ERRNO,T173SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      025144 104455
      025146 000163
      025150 031364
      025152 011670
4438 025154 004737 020064
4439 025160          170$:  JSR  PC,FATCHK                ;INC AND CHECK FOR MORE THAN 25 ERRORS
      025160 104406
4440 025160          ;
4441 025162 004737 032604
4442 025166 012701 031012
4443 025172 012702 032702
4444 025176 012221
4445 025200 011211
4446 025202 052711 000020
4447 025206 042711 000040
4448 025212 042711 000200
4449 025212
4450 025212
4451 025216 005000
4452 025220 012701 032662
4453 025224 012702 030772
4454 025230 012703 000024
4455 025234 004737 011322
4456 025240 103404
4457 025242
4458 025242          172$:  ERRHRD  ERRNO,T174CMP,MSGSTAT  ;REPORT ERROR
      025242 104456
      025244 000164
      025246 032045
      025250 012172
4459 025252          180$:  CKLOOP                ;LOOP ON ERROR, IF FLAG SET
      025252 104406
4460 025254          FORCEEXIT 205$
4461 025264 005237 002256
4462 025270 023727 002256 000377
4463 025276 101002
4464 025300 000137 024516
4465 025304          205$:
4467 025304          ENDSUB

```

```

;////////////////////////////////// END SUBTEST ////////////////////////////////////

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 78-5
TEST 1: SUBTEST 2: FIFO WRITE SINGLE BYTE TEST

025304			
025304	104403		
4469	025306	005737	002170
4470	025312	001402	
4471	025314	004737	020136
4472	025320		

260\$:

TST	FATFLG
BEQ	260\$
JSR	PC,LKDROP

L10040:	TRAP	CSESUB
:ANY FATAL ERRORS ?		
:BRANCH IF NOT		
:TRY TO DROP THE UNIT		

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 79
TEST 1: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

4475
4476
4477
4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4520
4521
4522
4523
4524
4525
4526
4527
4528
4529

.SBTTL TEST 1: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

TEST 1: SUBTEST 3:

SUBTEST DESCRIPTION:

This subtest verifies the ability of the FIFO to correctly pass a multiple data bytes from input to output. The following sequence is done with various data patterns and byte counts from 2 to 64.

1. Initial FIFO status is checked
2. The Write FIFO function.
3. Read Status is executed and FIFO status is checked.
4. Read FIFO is executed and the data and final status is checked.

TEST STEPS:

BEGIN

Write to TSSR to soft initialize
 Do a WRITE CHARACTERISTICS to setup a message buffer
 Do a Write Subsystem READ STATUS
 If Input Ready NOT=1 Then Print Error
 If Output Ready NOT=0 Then Print Error
 If Data In Miss NOT=0 Then Print Error
 If Last Word NOT=0 Then Print Error
 REPEAT FOR DATA 0 TO 377, 377 TO 0, FLOATING 1'S,0'S AND ALL 1'S/0'S
 REPEAT FOR BYTE COUNT 2 TO 64 DECIMAL

BEGIN

Do a Write Subsystem WRITE NPR to set tape direction out
 Do a Write Subsystem WRITE FIFO
 Do a Write Subsystem READ STATUS
 If Input Ready NOT=1 Then Print Error
 If Output Ready NOT=1 Then Print Error
 If Data In Miss NOT=0 Then Print Error
 If Last Word NOT=0 Then Print Error
 Do Write Subsystem READ FIFO
 If Data read from FIFO NOT= to Data sent Then Print Error
 Do a Write Subsystem READ STATUS
 If Input Ready NOT=1 Then Print Error
 If Output Ready NOT=0 Then Print Error
 If Data In Miss NOT=0 Then Print Error
 If Last Word NOT=0 Then Print Error

END

END

BGNSUB

////////// BEGIN SUBTEST //////////
T1.3:

TRAP CSBSUB

025320 104402

Write to TSSR register to soft initialize the controller

5s:

JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
 BCS 10S ;BR IF SOFT INIT OKAY
 MOV R0,R1 ;SAVE CONTENTS OF TSSR
 ERRDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 79-1
 TEST 1: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

025332	104455					TRAP	C\$ERDF
025334	000164					.WORD	116
025336	003550					.WORD	SFIERR
025340	011656					.WORD	SFIMSG
4530				:	Do a WRITE CHARACTERISTICS to setup a message buffer		
4531	025342	005037	002170	10\$:	CLR FATFLG	:	CLEAR FATAL ERROR FLAG
4532	025346	012704	032640		MOV #T17PACKET,R4	:	GET THE ADDRESS OF COMMAND PACKET
4533	025352	004737	010322		JSR PC,WRTCHR	:	DO WRITE CHARACTERISTICS COMMAND
4534	025356	103407			BCS 50\$:	BR IF CARRY SET (GOOD RETURN)
4535	025360	010301			MOV R0,R1	:	SAVE CONTENTS OF TSSR
4536	025362				NEXT.ERRNO		
4537	025362			42\$:	ERRDF ERRNO,T17SSR,PKTSSR	:	DEVICE FATAL SSR FAILED TO SET
	025362	104455				TRAP	C\$ERDF
	025364	000165				.WORD	117
	025366	031263				.WORD	T17SSR
	025370	011670				.WORD	PKTSSR
4538	025372	004737	020064		JSR PC,FATCHK	:	INC AND CHECK FOR MORE THAN 25 ERRORS
4539	025376			50\$:	CKLOOP	:	LOOP ON ERROR, IF FLAG SET
	025376	104406				TRAP	C\$CLP1
4540				:	Do a Write Subsystem READ STATUS		
4541	025400	004737	032422		JSR PC,T17SRD	:	SETUP PACKET FOR READ STATUS
4542	025404	012704	033010		MOV #T17PK2,R4	:	GET WRITE SUBSYSTEM COMMAND PACKET
4543	025410	010465	177776		MOV R4,TSDB(R5)	:	SET THE PACKET ADDRESS TO EXECUTE
4544	025414	004737	017220		JSR PC,CHKTSSR	:	WAIT FOR SSR TO SET
4545	025420	103407			BCS 70\$:	BR IF CARRY SET (GOOD RETURN)
4546	025422	010001			MOV R0,R1	:	SAVE CONTENTS OF TSSR
4547	025424				NEXT.ERRNO		
4548	025424			62\$:	ERRDF ERRNO,T173SSR,PKTSSR	:	DEVICE FATAL SSR FAILED TO SET
	025424	104455				TRAP	C\$ERDF
	025426	000166				.WORD	118
	025430	031364				.WORD	T173SSR
	025432	011670				.WORD	PKTSSR
4549	025434	004737	020064		JSR PC,FATCHK	:	INC AND CHECK FOR MORE THAN 25 ERRORS
4550	025440			70\$:	CKLOOP	:	LOOP ON ERROR, IF FLAG SET
	025440	104406				TRAP	C\$CLP1
4551				:	Set WORDS 0-7 of expd message buffer = to recv since not testing		
4552	025442	004737	032604		JSR PC,T17SETEXP	:	SET WORDS 0-7 EXPD=RECV
4553	025446	012701	031012		MOV #T17EXSTA,R1	:	GET EXPECTED READ STATUS
4554	025452	012702	032702		MOV #T17BFSTA,R2	:	GET RECV READ STATUS
4555	025456	012221			MOV (R2)+,(R1)+	:	SET EXPD WORD #8 = RECV TEMP
4556	025460	011211			MOV (R2),(R1)	:	SET EXPD WORD #9 = RECV TEMP
4557	025462	052711	000020		BIS #S2.INRDY,(R1)	:	SET EXP INPUT READY= 1
4558	025466	042711	000040		BIC #S2.OTRDY,(R1)	:	SET EXP OUTPUT READY= 0
4559	025472	042711	000200		BIC #S2.DIM,(R1)	:	SET EXP DATA IN MISS = 0
4560	025476	042711	000100		BIC #S2.ILW,(R1)	:	SET EXP LAST WORD (ILW)=0
4561				:	If Input Ready NOT=1 then Print Error		
4562				:	If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error		
4563				:	If Last Word NOT=0 Then Print Error		
4564	025502	005000			CLR R0	:	HIGH RECV ADDRESS FOR CKMSG2
4565	025504	012701	032662		MOV #T17BFR,R1	:	LOW RECV ADDRESS FOR CKMSG2
4566	025510	012702	030772		MOV #T17EXP,R2	:	EXPD ADDRESS
4567	025514	012703	000024		MOV #20.,R3	:	NUMBER OF BYTES TO COMPARE
4568	025520	004737	011322		JSR PC,CKMSG2	:	EXPD EQUAL RECV?
4569	025524	103404			BCS 90\$:	BR IF YES
4570	025526				NEXT.ERRNO		
4571	025526			82\$:	ERRHRD ERRNO,T171CMP,MSGSTAT	:	REPORT ERROR
	025526	104456				TRAP	C\$ERHRD

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 79-2
TEST 1: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

```

025530 000167 .WORD 119
025532 031603 .WORD T171CMP
025534 012172 .WORD MSGSTAT
4572 025536 90$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
025536 104406 TRAP C$CLP1
4573
4574
4575
4576 ; REPEAT FOR BYTE COUNT 2 TO 64 DECIMAL
4577 ; TSTFLAG =1 FOR INCREMENT TEST PATTERN
4578 ; =2 FOR DECREMENT TEST PATTERN
4579 ; =3 FOR TSTBLK TABLE PATTERN
4580 025540 012737 000001 002260 MOV #1,TSTFLAG ;TEST PATTERN FLAG
4581 025546 95$: MOV #2,COUNT ;GET FIRST BYTE COUNT
4582 025546 012737 000002 002254 100$:
4583 025554 ; Do a Write Subsystem WRITE NPR to set tape direction out
4584 ; MOV #NP.OUT,R0 ;SET TAPE DIRECTION OUT
4585 025554 012700 000100 JSR PC,T17SNPR ;SETUP T17PK2 FOR WRITE NPR
4586 025560 004737 032464 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4587 025564 012704 033010 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4588 025570 010465 177776 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4589 025574 004737 017220 BCS 105$ ;BR IF CARRY SET (GOOD RETURN)
4590 025600 103407 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4591 025602 010001 NEXT.ERRNO
4592 025604 102$: ERRDF ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
4593 025604 TRAP C$SERDF
025604 104455 .WORD 120
025606 000170 .WORD T174SSR
025610 031431 .WORD PKTSSR
025612 011670
4594 025614 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4595 025620 105$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
025620 104406 TRAP C$CLP1
4596 ; Do a Write Subsystem WRITE FIFO
4597 025622 004737 032564 JSR PC,T17CLEXP ;CLEAR EXPD BUFFER
4598 025626 012701 031114 MOV #T17WFDATA,R1 ;EXPD WRITE FIFO DATA BUFFER
4599 025632 013702 002254 MOV COUNT,R2 ;TEST PATTERN SIZE
4600 025636 022737 000001 002260 CMP #1,TSTFLAG ;INCREMENT PATTERN THIS TIME THRU?
4601 025644 001005 BNE 115$ ;BR IF NO
4602 025646 005000 CLR R0 ;INCREMENT TEST PATTERN
4603 025650 110021 110$: MOV R0,(R1)+ ;STORE INCREMENT TEST BYTE
4604 025652 005200 INC R0 ;SET NEXT PATTERN
4605 025654 005302 DEC R2 ;DONE?
4606 025656 003374 BGT 110$ ;BR IF NO
4607 025660 022737 000002 002260 115$: CMP #2,TSTFLAG ;DECREMENT PATTERN THIS TIME THRU?
4608 025666 001006 BNE 125$ ;BR IF NO
4609 025670 012700 000377 MOV #377,R0 ;DECREMENT TEST PATTERN
4610 025674 110021 120$: MOV R0,(R1)+ ;STORE DECREMENT TEST BYTE
4611 025676 005300 DEC R0 ;SET NEXT PATTERN
4612 025700 005302 DEC R2 ;DONE?
4613 025702 003374 BGT 120$ ;BR IF NO
4614 025704 022737 000003 002260 125$: CMP #3,TSTFLAG ;TSTBLK PATTERNS THIS TIME THRU?
4615 025712 001005 BNE 135$ ;BR IF NO
4616 025714 012700 002720 MOV #TSTBLK,R0 ;FLOAT 1'S/O'S ETC. TEST TABLE
4617 025720 112021 130$: MOV R0,(R0)+,(R1)+ ;STORE A TSTBLK BYTE
4618 025722 005302 DEC R2 ;DONE?
4619 025724 003375 BGT 130$ ;BR IF NO

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 79-4
TEST 1: SUBTEST 3: FIFO WRITE MULTIPLE BYTES TEST

```

026130 000173 .WORD 123
026132 031761 .WORD T173CMP
026134 012172 .WORD MSGSTAT
4666 026136 170$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
026136 104406 TRAP C$CLP1
4667
4668 ; Do Write Subsystem READ FIFO
4669 026140 013700 002254 MOV COUNT,R0 ;SET READ BYTE COUNT
4670 026144 004737 032544 JSR PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
4671 026150 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4672 026154 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4673 026160 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4674 026164 103407 BCS 180$ ;BR IF CARRY SET (GOOD RETURN)
4675 026166 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4676 026170
4677 026170 172$: ERRDF ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
026170 104455 TRAP C$ERDF
026172 000174 .WORD 124
026174 031540 .WORD T176SSR
026176 011670 .WORD PKTSSR
4678 026200 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4679 026204 180$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
026204 104406 TRAP C$CLP1
4680
4681 ; If Data read from FIFO NOT= to Data sent Then Print Error
4682 026206 005000 CLR R0 ;HIGH RECV ADDRESS FOR CKMSG2
4683 026210 012702 031114 MOV #T17WFDATA,R2 ;GET EXPECTED ADDRESS FOR CKMSG2
4684 026214 012701 032702 MOV #T17BFSTA,R1 ;GET RECEIVED ADDRESS FOR CKMSG2
4685 026220 013703 002254 MOV COUNT,R3 ;NUMBER OF BYTES TO COMPARE
4686 026224 004737 011322 JSR PC,CKMSG2 ;EXPD EQUAL RECV?
4687 026230 103406 BCS 200$ ;BR IF YES
4688 026232
4689 026232 013701 002254 192$: MOV COUNT,R1 ;GET BYTE COUNT
4690 026236 ERRHRD ERRNO,T175CMP,FIFEXP ;REPORT ERROR
026236 104456 TRAP C$ERHRD
026240 000175 .WORD 125
026242 032130 .WORD T175CMP
026244 012012 .WORD FIFEXP
4691 026246 200$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
026246 104406 TRAP C$CLP1
4692
4693 ; Do a Write Subsystem READ STATUS
4694 026250 004737 032422 JSR PC,T17SRD ;SETUP PACKET FOR READ STATUS
4695 026254 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4696 026260 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4697 026264 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
4698 026270 103407 BCS 220$ ;BR IF CARRY SET (GOOD RETURN)
4699 026272 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
4700 026274
4701 026274 212$: ERRDF ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
026274 104455 TRAP C$ERDF
026276 000176 .WORD 126
026300 031364 .WORD T173SSR
026302 011670 .WORD PKTSSR
4702 026304 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4703 026310 220$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
026310 104406 TRAP C$CLP1

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 79-5
 TEST 1: SUBTEST 3. FIFO WRITE MULTIPLE BYTES TEST

```

4704          :      Set WORDS 0-7 of expd message buffer = to recv since not testing
4705 026312 004737 032604      JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RECV
4706 026316 012701 031012      MOV      #T17EXSTA,R1      ;GET EXPECTED READ STATUS
4707 026322 012702 032702      MOV      #T17BFSTA,R2      ;GET RECV READ STATUS
4708 026326 012221              MOV      (R2)+,(R1)+      ;SET EXPD WORD #8 = RECV TEMP
4709 026330 011211              MOV      (R2),(R1)        ;SET EXPD WORD #9 = RECV TEMP
4710 026332 052711 000020      BIS      #S2.INRDY,(R1)    ;SET EXP INPUT READY= 1
4711 026336 042711 000040      BIC      #S2.OTRDY,(R1)    ;SET EXP OUTPUT READY= 0
4712 026342 042711 000200      BIC      #S2.DIM,(R1)     ;SET EXP DATA IN MISS = 0
4713 026346 042711 000100      BIC      #S2.ILW,(R1)     ;SET EXP LAST WORD (ILW)=0
4714          :
4715          :      If Input Ready NOT=1 then Print Error
4716 026352 005000              CLR      R0                ;HIGH RECV ADDRESS FOR CKMSG2
4717 026354 012701 032662      MOV      #T17BFR,R1        ;LOW RECV ADDRESS FOR CKMSG2
4718 026360 012702 030772      MOV      #T17EXP,R2        ;EXPD ADDRESS
4719 026364 012703 000024      MOV      #20.,R3          ;NUMBER OF BYTES TO COMPARE
4720 026370 004737 011322      JSR      PC,CKMSG2         ;EXPD EQUAL RECV?
4721 026374 103404              BCS      240$              ;BR IF YES
4722 026376
4723 026376 232$: ERRHRD  ERRNO,T174CMP,MSGSTAT ;REPORT ERROR
      026376 104456              TRAP      C$ERHRD
      026400 000177              .WORD    127
      026402 032045              .WORD    T174CMP
      026404 012172              .WORD    MSGSTAT
4724 026406 240$: CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      026406 104406              TRAP      C$CLP1
4725 026410          FORCEEXIT      250$      ;END
4726 026420 005237 002254      INC      COUNT             ;GET NEXT BYTE COUNT
4727 026424 023727 002254 000077  CMP      COUNT,#77         ;DONE 0 TO 77
4728 026432 101002              BHI      250$              ;BR IF YES
4729 026434 000137 025554      JMP      100$              ;DO ANOTHER BYTE COUNT
4730 026440 005237 002260 250$: INC      TSTFLAG           ;GET NEXT TEST PATTERN CODE
4731 026444 023727 002260 000003  CMP      TSTFLAG,#3        ;DONE INC,DEC,TSTBLK PATTERNS?
4732 026452 101002              BHI      255$              ;BR IF YES
4733 026454 000137 025546      JMP      95$               ;DO ANOTHER TEST PATTERN
4734 026460 255$:
4735 026460          ENDSUB          ;////////// END SUBTEST //////////
      026460          L10041:
      026460 104403              TRAP      C$ESUB
4736
4737 026462 005737 002170      TST      FATFLG           ;ANY FATAL ERRORS ?
4738 026466 001402              BEQ      260$              ;BRANCH IF NOT
4739 026470 004737 020136      JSR      PC,CKDROP         ;TRY TO DROP THE UNIT
4740 026474 260$:
4741
4742
4743
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 80
 TEST 1: SUBTEST 4; FIFO VERIFY ILW STATUS

```

4745          .SBTTL TEST 1: SUBTEST 4; FIFO Verify ILW Status
4746
4747
4748          :++
4749          : TEST 1: SUBTEST 4;
4750          : SUBTEST DESCRIPTION:
4751          :
4752          : This subtest verifies that reading the FIFO when it is
4753          : empty causes the Last Word (ILW) status to assert.
4754          :
4755          : TEST STEPS:
4756          :
4757          : BEGIN
4758          : Write to TSSR to soft initialize
4759          : Do Write Subsystem READ FIFO with byte count equal to 1
4760          : Do a Write Subsystem READ STATUS
4761          : If Input Ready NOT=1 Then Print Error
4762          : If Output Ready NOT=0 Then Print Error
4763          : If Data In Miss NOT=0 Then Print Error
4764          : If Last Word (ILW) NOT=1 Then Print Error
4765          :
4766          : END
4767          :--
4768          :
4769          :
4770          :
4771          :
4772          :
4773          :
4774          :
4775          :
4776          :
4777          :
4778          :
4779          :
4780          :
4781          :
4782          :
4783          :
4784          :
4785          :
4786          :
4787          :
4788          :
4789          :
4790          :

```

026474				BGNSUB	////////// BEGIN SUBTEST //////////
026474					T1.4: TRAP CSBSUB
026474	104402				
026476				58:	Write to TSSR register to soft initialize the controller
026476	004737	016630		JSR	PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
026502	103405			BCS	10\$;BR IF SOFT INIT OKAY
026504	010001			MOV	RO,R1 ;SAVE CONTENTS OF TSSR
026506				ERRDF	ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
026506	104455				TRAP CSERDF
026510	000177				.WORD 127
026512	003550				.WORD SFIERR
026514	011656				.WORD SFIMSG
026516	005037	002170		108:	Do a WRITE CHARACTERISTICS to setup a message buffer
026522	012704	032640		CLR	FATFLG ;CLEAR FATAL ERROR FLAG
026526	004737	010322		MOV	#T17PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
026532	103407			JSR	PC,WRTCHR ;DO WRITE CHARACTERISTICS COMMAND
026534	010001			BCS	50\$;BR IF CARRY SET (GOOD RETURN)
026536				MOV	RO,R1 ;SAVE CONTENTS OF TSSR
026536				NEXT.ERRNO	
026536	104455			428:	ERRDF ERRNO,T17SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
026540	000200				TRAP CSERDF
026542	031263				.WORD 128
026544	011670				.WORD T17SSR
026546	004737	020064			.WORD PKTSSR
026552				508:	JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
026552	104406			CKLOOP	;LOOP ON ERROR, IF FLAG SET
					TRAP CSCLP1
026554	012700	000001		:	Do Write Subsystem READ FIFO with byte count equal to 1
026560	004737	032544		MOV	#1,RO ;SET READ BYTE COUNT
026564	012704	033010		JSR	PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
026570	010465	177776		MOV	#T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
				MOV	R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 80-1
 TEST 1: SUBTEST 4; FJFO VERIFY ILW STATUS

```

4791 026574 004737 017220      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
4792 026600 103407              BCS      150$           ;BR IF CARRY SET (GOOD RETURN)
4793 026602 010001              MOV      RO,R1         ;SAVE CONTENTS OF TSSR
4794 026604              NEXT.ERRNO
4795 026604              142$:  ERRDF  ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      026604 104455              TRAP    CSERDF
      026606 000201              .WORD  129
      026610 031540              .WORD  T176SSR
      026612 011670              .WORD  PKTSSR
4796 026614 004737 020064      JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
4797 026620              150$:  CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      026620 104406              TRAP    CSCLP1
4798
4799
      ; Do a Write Subsystem READ STATUS
4800 026622 004737 032422      JSR      PC,T17SRD     ;SETUP PACKET FOR READ STATUS
4801 026626 012704 033010      MOV      #T17PK2,R4   ;GET WRITE SUBSYSTEM COMMAND PACKET
4802 026632 010465 177776      MOV      R4,TSDB(R5)  ;SET THE PACKET ADDRESS TO EXECUTE
4803 026636 004737 017220      JSR      PC,CHKTSSR   ;WAIT FOR SSR TO SET
4804 026642 103407              BCS      170$         ;BR IF CARRY SET (GOOD RETURN)
4805 026644 010001              MOV      RO,R1         ;SAVE CONTENTS OF TSSR
4806 026646              162$:  ERRDF  ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      026646 104455              TRAP    CSERDF
      026650 000202              .WORD  130
      026652 031364              .WORD  T173SSR
      026654 011670              .WORD  PKTSSR
4808 026656 004737 020064      JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
4809 026662              170$:  CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      026662 104406              TRAP    CSCLP1
4810
      ; Set WORDS 0-7 of expd message buffer = to recv since not testing
4811 026664 004737 032604      JSR      PC,T17SETEXP  ;SET WORDS 0-7 EXPD=RECV
4812 026670 012701 031012      MOV      #T17EXSTA,R1 ;GET EXPECTED READ STATUS
4813 026674 012702 032702      MOV      #T17BFSTA,R2 ;GET RECV READ STATUS
4814 026700 012221              MOV      (R2)+,(R1)+  ;SET EXPD WORD #8 = RECV TEMP
4815 026702 011211              MOV      (R2),(R1)    ;SET EXPD WORD #9 = RECV TEMP
4816 026704 052711 000020      BIS      #S2.INRDY,(R1) ;SET EXP INPUT READY= 1
4817 026710 042711 000040      BIC      #S2.OUTRDY,(R1) ;SET EXP OUTPUT READY= 0
4818 026714 042711 000200      BIC      #S2.DIM,(R1)  ;SET EXP DATA IN MISS = 0
4819 026720 052711 000100      BIS      #S2.ILW,(R1) ;SET EXP LAST WORD (ILW)=1
4820
      ; If Input Ready NOT=1 then Print Error
      ; If Output Ready NOT=0 or Data in Miss NOT=0 Then Print Error
      ; If Las. Word (ILW) NOT=1 Then Print Error
4823 026724 005000              CLR      RO           ;HIGH RECV ADDRESS FOR CKMSG2
4824 026726 012701 032662      MOV      #T17BFR,R1   ;LOW RECV ADDRESS FOR CKMSG2
4825 026732 012702 030772      MOV      #T17EXP,R2   ;EXPD ADDRESS
4826 026736 012703 000024      MOV      #20,,R3     ;NUMBER OF BYTES TO COMPARE
4827 026742 004737 011322      JSR      PC,CKMSG2    ;EXPD EQUAL RECV?
4828 026746 103404              BCS      180$         ;BR IF YES
4829 026750              172$:  NEXT.ERRNO
4830 026750              ERRHRD  ERRNO,T176CMP,MSGSTAT ;REPORT ERROR
      026750 104456              TRAP    CSERHRD
      026752 000203              .WORD  131
      026754 032204              .WORD  T176CMP
      026756 012172              .WORD  MSGSTAT
4831 026760              180$:  CKLOOP          ;LOOP ON ERROR, IF FLAG SET
      026760 104406              TRAP    CSCLP1
4832

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 80-2
TEST 1: SUBTEST 4; FIFO VERIFY ILW STATUS

4833 026762
026762
026762 104403

ENDSUB

;/!!!!!!!!!!!! END SUBTEST !!!!!!!!!!!!!
L10042: TRAP CSESUB

4834
4835 026764 005737 002170
4836 026770 001402
4837 026772 004737 020136
4838 026776
4839
4840

260\$:

TST FAT:LG
BEQ 260\$
JSR PC,CKDROP

;ANY FATAL ERRORS ?
;BRANCH IF NOT
;TRY TO DROP THE UNIT

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 81
 TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

4842
4843
4844
4845
4846
4847
4848
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4870
4871
4872
4873
4874
4875
4876
4877
4878
4879

.SBTTL TEST 1: SUBTEST 5: FIFO Verify Input Ready

++
TEST 1: SUBTEST 5:

SUBTEST DESCRIPTION:

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

TEST STEPS:

BEGIN

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

END

--
BGNSUB ;////////// BEGIN SUBTEST ///////////
T1.5: TRAP CSBSUB

4880 026776
026776 104402
4881
4882
4883 027000
4884 027000 004737 016630
4885 027004 103405
4886 027006 010001
4887 027010
027010 104455
027012 000203
027014 003550
027016 011656
4888
4889 027020 005037 002170
4890 027024 012704 032640
4891 027030 004737 010322
4892 027034 103407

5s: Write to TSSR register to soft initialize the controller
 JSR PC,SOFINIT ;WRITE TO TSSR TO SOFT INITIALIZE
 BCS 10\$;BR IF SOFT INIT OKAY
 MOV R0,R1 ;SAVE CONTENTS OF TSSR
 ERRDF ERRNO,SFIERR,SFIMSG ;DEVICE FATAL DURING INIT
 TRAP CSERDF
 .WORD 131
 .WORD SFIERR
 .WORD SFIMSG
 10s: Do a WRITE CHARACTERISTICS to setup a message buffer
 CLR FATFLG ;CLEAR FATAL ERROR FLAG
 MOV #T17PACKET,R4 ;GET THE ADDRESS OF COMMAND PACKET
 JSR PC,WRTCHR ;DO WRITE CHARACTERISTICS COMMAND
 BCS 50\$;BR IF CARRY SET (GOOD RETURN)

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 81-1
 TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

```

4893 027036 010001      MOV      R0,R1      ;SAVE CONTENTS OF TSSR
4894 027040      NEXT.ERRNO
4895 027040      42$:  ERRDF  ERRNO,T17SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      027040 104455      TRAP      C$ERDF
      027042 000204      .WORD    132
      027044 031263      .WORD    T17SSR
      027046 011670      .WORD    PKTSSR
4896 027050 004737 020064 JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4897 027054      50$:  CKLOOP ;LOOP ON ERROR, IF FLAG SET
      027054 104406      TRAP      C$CLP1

4898
4899 ; Do a Write Subsystem WRITE NPR to set tape direction out
4900 027056 012700 000100 100$: MOV      #NP.OUT,R0 ;SET TAPE DIRECTION OUT
4901 027062 004737 032464 JSR      PC,T17SNPR ;SETUP T17PK2 FOR WRITE NPR
4902 027066 012704 033010 MOV      #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4903 027072 010465 177776 MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4904 027076 004737 017220 JSR      PC,CHKTSSR ;WAIT FOR SSR TO SET
4905 027102 103407 BCS     105$ ;BR IF CARRY SET (GOOD RETURN)
4906 027104 010001 MOV      R0,R1 ;SAVE CONTENTS OF TSSR
4907 027106      NEXT.ERRNO
4908 027106      102$: ERRDF  ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      027106 104455      TRAP      C$ERDF
      027110 000205      .WORD    133
      027112 031431      .WORD    T174SSR
      027114 011670      .WORD    PKTSSR
4909 027116 004737 020064 105$: JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4910 027122      CKLOOP ;LOOP ON ERROR, IF FLAG SET
      027122 104406      TRAP      C$CLP1

4911
4912 ; Do a Write Subsystem WRITE FIFO 64. bytes incrementing pattern
4913 027124 012737 000100 002254 MOV      #64.,COUNT ;WRITE 64 BYTES
4914 027132 012701 031114 MOV      #T17WFDATA,R1 ;EXPD WRITE FIFO DATA BUFFER
4915 027136 012702 000100 MOV      #64.,R2 ;TEST PATTERN SIZE
4916 027142 005000 CLR      R0 ;INCREMENT TEST PATTERN
4917 027144 110021 110$: MOVB   R0,(R1)+ ;STORE INCREMENT TEST BYTE
4918 027146 005200 INC      R0 ;SET NEXT PATTERN
4919 027150 005302 DFC     R2 ;DONE?
4920 027152 003374 BGT     110$ ;BR IF NO
4921 027154 013700 002254 MOV      COUNT,R0 ;FIFO BYTE COUNT
4922 027160 012701 031114 MOV      #T17WFDATA,R1 ;FIFO WRITE DATA ADDRESS
4923 027164 004737 032510 JSR      PC,T17WFIF ;SETUP T17PK2 FOR WRITE FIFO
4924 027170 012704 033010 MOV      #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
4925 027174 010465 177776 MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
4926 027200 004737 017220 JSR      PC,CHKTSSR ;WAIT FOR SSR TO SET
4927 027204 103407 BCS     150$ ;BR IF CARRY SET (GOOD RETURN)
4928 027206 010001 MOV      R0,R1 ;SAVE CONTENTS OF TSSR
4929 027210      NEXT.ERRNO
4930 027210      142$: ERRDF  ERRNO,T175SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      027210 104455      TRAP      C$ERDF
      027212 000206      .WORD    134
      027214 031474      .WORD    T175SSR
      027216 011670      .WORD    PKTSSR
4931 027220 004737 020064 150$: JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
4932 027224      CKLOOP ;LOOP ON ERROR, IF FLAG SET
      027224 104406      TRAP      C$CLP1

4933
4934 ; Do a Write Subsystem READ STATUS

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 81-2
 TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

```

4935      :      If Input Ready NOT=0      Then Print Error
4936      :      If Output Ready NOT=1      Then Print Error
4937      :      If Data In Miss NOT=0      Then Print Error
4938 027226 004737 032422      JSR      PC,T17SRD      ;SETUP PACKET FOR READ STATUS
4939 027232 012704 033010      MOV      #T1/PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
4940 027236 010465 177776      MOV      R4,TSDB(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
4941 027242 004737 017220      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
4942 027246 103407      BCS      160$      ;BR IF CARRY SET (GOOD RETURN)
4943 027250 010001      MOV      RO,R1      ;SAVE CONTENTS OF TSSR
4944 027252      NEXT.ERRNO
4945 027252      157$: ERRDF  ERRNO,T173SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      027252 104455      TRAP      CSERDF
      027254 000207      .WORD      135
      027256 031364      .WORD      T173SSR
      027260 011670      .WORD      PKTSSR
4946 027262 004737 020064      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
4947 027266      160$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      027266 104406      TRAP      CSCLP1
4948      :      Set WORDS 0-7 of expd message buffer = to recv since not testing
4949 027270 004737 032604      JSR      PC,T17SETEXP      ;SET WORDS 0-7 EXPD=RECV
4950 027274 012701 031012      MOV      #T17EXSTA,R1      ;GET EXPECTED READ STATUS
4951 027300 012702 032702      MOV      #T17BFSTA,R2      ;GET RECV READ STATUS
4952 027304 012221      MOV      (R2)+,(R1)+      ;SET EXPD WORD #8 = RECV TEMP
4953 027306 011211      MOV      (R2),(R1)      ;SET EXPD WORD #9 = RECV TEMP
4954 027310 042711 000020      BIC      #S2.INRDY,(R1)      ;SET EXP INPUT READY= 0
4955 027314 052711 000040      BIS      #S2.OURDY,(R1)      ;SET EXP OUTPUT READY= 1
4956 027320 042711 000200      BIC      #S2.DIM,(R1)      ;SET EXP DATA IN MISS = 0
4957 027324 005000      CLR      RO      ;HIGH RECV ADDRESS FOR CKMSG2
4958 027326 012701 032662      MOV      #T17BFR,R1      ;LOW RECV ADDRESS FOR CKMSG2
4959 027332 012702 030772      MOV      #T17EXP,R2      ;EXPD ADDRESS
4960 027336 012703 000024      MOV      #20.,R3      ;NUMBER OF BYTES TO COMPARE
4961 027342 004737 011322      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
4962 027346 103404      BCS      170$      ;BR IF YES
4963 027350      NEXT.ERRNO
4964 027350      162$: FRRHRD  ERRNO,T173CMP,MSGSTAT      ;REPORT ERROR
      027350 104456      TRAP      CSERHRD
      027352 000210      .WORD      136
      027354 031761      .WORD      T173CMP
      027356 012172      .WORD      MSGSTAT
4965 027360      170$: CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      027360 104406      TRAP      CSCLP1
4966
4967
4968      :      Do a Write Subsystem WRITE FIFO 1 byte for a total of 65. written
4969 027362 012700 000001      MOV      #1,RO      ;FIFO BYTE COUNT
4970 027366 012701 031114      MOV      #T17WFDATA,R1      ;FIFO WRITE DATA ADDRESS
4971 027372 004737 032510      JSR      PC,T17WFIF      ;SETUP T17PK2 FOR WRITE FIFO
4972 027376 012704 033010      MOV      #T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
4973 027402 010465 177776      MOV      R4,TSDB(R5)      ;SET THE PACKET ADDRESS TO EXECUTE
4974 027406 004737 017220      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
4975 027412 103407      BCS      180$      ;BR IF CARRY SET (GOOD RETURN)
4976 027414 010001      MOV      RO,R1      ;SAVE CONTENTS OF TSSR
4977 027416      NEXT.ERRNO
4978 027416      172$: ERRDF  ERRNO,T175SSR,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      027416 104455      TRAP      CSERDF
      027420 000211      .WORD      137
      027422 031474      .WORD      T175SSR

```

CZTUXAO TUBO FRONT END PRT B HACRO M1200 29-MAR-83 13:32 PAGE 81-3
 TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

```

4979 027424 011670
4980 027426 004737 020064
4981 027432 104406
4982 027432 104406
4983
4984
4985
4986 027434 004737 032422
4987 027440 012704 033010
4988 027444 010465 177776
4989 027450 004737 017220
4990 027454 103407
4991 027456 010001
4992 027460
4993 027460
      027460 104455
      027462 000212
      027464 031364
      027466 011670
4994 027470 004737 020064
4995 027474
      027474 104406
4996
4997 027476 004737 032604
4998 027502 012701 031012
4999 027506 012702 032702
5000 027512 012221
5001 027514 011211
5002 027516 042711 000020
5003 027522 052711 000040
5004 027526 052711 000200
5005 027532 005000
5006 027534 012701 032662
5007 027540 012702 030772
5008 027544 012703 000024
5009 027550 004737 011322
5010 027554 103404
5011 027556
5012 027556
      027556 104456
      027560 000213
      027562 031761
      027564 012172
5013 027566
      027566 104406
5014
5015 027570 013700 002254
5016 027574 004737 032544
5017 02.600 012704 033010
5018 027604 010465 177776
5019 027610 004737 017220
5020 027614 103407
5021 027616 010001
5022 027620
5023 027620

```

```

180$: JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
      CKLOOP ;LOOP ON ERROR, IF FLAG SET
      TRAP C$CLP1

: Do a Write Subsystem READ STATUS
: If Input Ready NOT=0 Then Print Error
: If Output Ready NOT=1 Then Print Error
: If Data In Miss NOT=1 Then Print Error
JSR PC,T17SRD ;SETUP PACKET FOR READ STATUS
MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
BCS 190$ ;BR IF CARRY SET (GOOD RETURN)
MOV R0,R1 ;SAVE CONTENTS OF TSSR
NEXT.ERRNO
187$: ERRDF ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
      TRAP CSERDF
      .WORD 138
      .WORD T173SSR
      .WORD PKTSSR
190$: JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
      CKLOOP ;LOOP ON ERROR, IF FLAG SET
      TRAP C$CLP1

: Set WORDS 0-7 of expd message buffer = to recv since not testing
JSR PC,T17SETEXP ;SET WORDS 0-7 EXPD=RECV
MOV #T17EXSTA,R1 ;GET EXPECTED READ STATUS
MOV #T17BFSTA,R2 ;GET RECV READ STATUS
MOV (R2)+,(R1)+ ;SET EXPD WORD #8 = RECV TEMP
MOV (R2),(R1) ;SET EXPD WORD #9 = RECV TEMP
BIC #S2.INRDY,(R1) ;SET EXP INPUT READY= 0
BIS #S2.OUTRDY,(R1) ;SET EXP OUTPUT READY= 1
BIS #S2.DIM,(R1) ;SET EXP DATA IN MISS = 1
CLR R0 ;HIGH RECV ADDRESS FOR CKMSG2
MOV #T17BFR,R1 ;LOW RECV ADDRESS FOR CKMSG2
MOV #T17EXP,R2 ;EXPD ADDRESS
MOV #20.,R3 ;NUMBER OF BYTES TO COMPARE
JSR PC,CKMSG2 ;EXPD EQUAL RECV?
BCS 200$ ;BR IF YES
NEXT.ERRNO
192$: ERRHRD ERRNO,T173CMP,MSGSTAT ;REPORT ERROR
      TRAP CSERHRD
      .WORD 139
      .WORD T173CMP
      .WORD MSGSTAT
200$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
      TRAP C$CLP1

: Do Write Subsystem READ FIFO
MOV COUNT,R0 ;SET READ BYTE COUNT
JSR PC,T17RFIF ;SETUP T17PK2 FOR READ FIFO
MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
BCS 220$ ;BR IF CARRY SET (GOOD RETURN)
MOV R0,R1 ;SAVE CONTENTS OF TSSR
NEXT.ERRNO
212$: ERRDF ERRNO,T176SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET

```

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 81-4
 TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

```

027620 104455 TRAP CSERDF
027622 000214 .WORD 140
027624 031540 .WORD T176SSR
027626 011670 .WORD PKTSSR
5024 027630 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5025 027634 104406 220$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
027634 104406 TRAP CSCLP1
5026
5027 ; If Data read from FIFO NOT= to Data sent Then Print Error
5028 027636 005000 CLR R0 ;HIGH RECV ADDRESS FOR CKMSG2
5029 027640 012702 031114 MOV #T17WFDATA,R2 ;GET EXPECTED ADDRESS FOR CKMSG2
5030 027644 012701 032702 MOV #T17BFSTA,R1 ;GET RECEIVED ADDRESS FOR CKMSG2
5031 027650 013703 002254 MOV COUNT,R3 ;NUMBER OF BYTES TO COMPARE
5032 027654 004737 011322 JSR PC,CKMSG2 ;EXPD EQUAL RECV?
5033 027660 103406 BCS 240$ ;BR IF YES
5034 027662
5035 027662 013701 002254 232$: MOV COUNT,R1 ;GET BYTE COUNT
5036 027666 ERRHRD ERRNO,T175CMP,FIFEXP ;REPORT ERROR
027666 104456 TRAP CSERHRD
027670 000215 .WORD 141
027672 032130 .WORD T175CMP
027674 012012 .WORD FIFEXP
5037 027676 104406 240$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
027676 104406 TRAP CSCLP1
5038
5039 ; Do a Write Subsystem READ STATUS
5040 ; If Input Ready NOT=1 Then Print Error
5041 ; If Output Ready NOT=0 Then Print Error
5042 ; If Data In Miss NOT=1 Then Print Error
5043 027700 004737 032422 JSR PC,T17SRD ;SETUP PACKET FOR READ STATUS
5044 027704 012704 033010 MOV #T17PK2,R4 ;GET WRITE SUBSYSTEM COMMAND PACKET
5045 027710 010465 177776 MOV R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
5046 027714 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
5047 027720 103407 BCS 260$ ;BR IF CARRY SET (GOOD RETURN)
5048 027722 010001 MOV R0,R1 ;SAVE CONTENTS OF TSSR
5049 027724
5050 027724 104455 252$: ERRDF ERRNO,T173SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
027724 104455 TRAP CSERDF
027726 000216 .WORD 142
027730 031364 .WORD T173SSR
027732 011670 .WORD PKTSSR
5051 027734 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
5052 027740 104406 260$: CKLOOP ;LOOP ON ERROR, IF FLAG SET
027740 104406 TRAP CSCLP1
5053 ; Set WORDS 0-7 of expd message buffer = to recv since not testing
5054 027742 004737 032604 JSR PC,T17SETEXP ;SET WORDS 0-7 EXPD=RECV
5055 027746 012701 031012 MOV #T17EXSTA,R1 ;GET EXPECTED READ STATUS
5056 027752 012702 032702 MOV #T17BFSTA,R2 ;GET RECV READ STATUS
5057 027756 012221 MOV (R2)+,(R1)+ ;SET EXPD WORD #8 = RECV TEMP
5058 027760 011211 MOV (R2),(R1) ;SET EXPD WORD #9 = RECV TEMP
5059 027762 052711 000020 BIS #S2.INRDY,(R1) ;SET EXP INPUT READY= 1
5060 027766 042711 000040 BIC #S2.OURDY,(R1) ;SET EXP OUTPUT READY= 0
5061 027772 052711 000200 BIS #S2.DIM,(R1) ;SET EXP DATA IN MISS = 1
5062 027776 005000 CLR R0 ;HIGH RECV ADDRESS FOR CKMSG2
5063 030000 012701 032662 MOV #T17BFR,R1 ;LOW RECV ADDRESS FOR CKMSG2
5064 030004 012702 030772 MOV #T17EXP,R2 ;EXPD ADDRESS
5065 030010 012703 000024 MOV #20.,R3 ;NUMBER OF BYTES TO COMPARE
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 81-5
TEST 1: SUBTEST 5: FIFO VERIFY INPUT READY

```

5066 030014 004737 011322      JSR    PC,CKMSG2      ;EXPD EQUAL RECV?
5067 030020 103404              BCS    280$          ;BR IF YES
5068 030022                      NEXT.ERRNO
5069 030022 272$:  ERRHRD  EPRNO,T174CMP,MSGSTAT ;REPORT ERROR
      030022 104456                      TRAP  CSERMRD
      030024 000217                      .WORD 143
      030026 032045                      .WORD T174CMP
      030030 012172                      .WORD MSGSTAT
5070 030032 280$:  CKLOOP              ;LOOP ON ERROR, IF FLAG SET
      030032 104406                      TRAP  CSCLP1
5071
5072 030034                      ENDSUB              .////////// END SUBTEST //////////
      030034                      L10043:
      030034 104403                      TRAP  CSesub
5073
5074 030036 005737 002170      TST    FATFLG        ;ANY FATAL ERRORS ?
5075 030042 001402              BEQ    300$          ;BRANCH IF NOT
5076 030044 004737 020136      JSR    PC,CKDROP    ;TRY TO DROP THE UNIT
5077 030050 300$:
5078
5079
5080

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 82-1
 TEST 1: SUBTEST 6; FIFO VERIFY RESET FIFO TEST

```

5133 030110 010001      MOV      RO,R1      ;SAVE CONTENTS OF TSSR
5134 030112      NEXT.ERRNO
5135 030112      42$:  ERRDF  ERRNO,T17SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      030112 104455      TRAP  C$ERDF
      030114 000220      .WORD 144
      030116 031263      .WORD T17SSR
      030120 011670      .WORD PKTSSR
5136 030122 004737 020064 JSR      PC,FATCHK  ;INC AND CHECK FOR MORE THAN 25 ERRORS
5137 030126      50$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      030126 104406      TRAP  C$CLP1
5138 :
5139 030130 004737 032442 JSR      PC,T17RSFIF ;SETUP PKT FOR WRITE MISC RESET FIFO
5140 030134 012704 033010 MOV      #T17PK2,R4  ;GET WRITE SUBSYSTEM COMMAND PACKET
5141 030140 010465 177776 MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
5142 030144 004737 017220 JSR      PC,CHKTSSR  ;WAIT FOR SSR TO SET
5143 030150 103407      BCS      70$        ;BR IF CARRY SET (GOOD RETURN)
5144 030152 010001      MOV      RO,R1      ;SAVE CONTENTS OF TSSR
5145 030154      NEXT.ERRNO
5146 030154      62$:  ERRDF  ERRNO,T172SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      030154 104455      TRAP  C$ERDF
      030156 000221      .WORD 145
      030160 031320      .WORD T172SSR
      030162 011670      .WORD PKTSSR
5147 030164 004737 020064 JSR      PC,FATCHK  ;INC AND CHECK FOR MORE THAN 25 ERRORS
5148 030170      70$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      030170 104406      TRAP  C$CLP1
5149 :
5150 :
5151 :
5152 :
      Do a Write Subsystem READ STATUS
      If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
      signals NOT=0 Then Print Error
5153 030172 004737 032422 JSR      PC,T17SRD   ;SETUP PACKET FOR READ STATUS
5154 030176 012704 033010 MOV      #T17PK2,R4  ;GET WRITE SUBSYSTEM COMMAND PACKET
5155 030202 010465 177776 MOV      R4,TSDB(R5) ;SET THE PACKET ADDRESS TO EXECUTE
5156 030206 004737 017220 JSR      PC,CHKTSSR  ;WAIT FOR SSR TO SET
5157 030212 103407      BCS      80$        ;BR IF CARRY SET (GOOD RETURN)
5158 030214 010001      MOV      RO,R1      ;SAVE CONTENTS OF TSSR
5159 030216      NEXT.ERRNO
5160 030216      77$:  ERRDF  ERRNO,T173SSR,PKTSSR  ;DEVICE FATAL SSR FAILED TO SET
      030216 104455      TRAP  C$ERDF
      030220 000222      .WORD 146
      030222 031364      .WORD T173SSR
      030224 011670      .WORD PKTSSR
5161 030226 004737 020064 JSR      PC,FATCHK  ;INC AND CHECK FOR MORE THAN 25 ERRORS
5162 030232      80$:  CKLOOP  ;LOOP ON ERROR, IF FLAG SET
      030232 104406      TRAP  C$CLP1
5163 030234 004737 032604 JSR      PC,T17SETEXP ;SET WORDS 0-7 EXPD=RECV (NOT TESTING)
5164 030240 012701 031012 MOV      #T17EXSTA,R1 ;GET EXPECTED READ STATUS
5165 030244 012702 032702 MOV      #T17BFSTA,R2 ;GET RECV READ STATUS
5166 030250 011211      MOV      (R2),(R1)   ;SET EXPD WORD #8 = RECV TEMP
5167 030252 042711 002000 BIC      #S1.ICER,(R1) ;SET EXPD ICER =0
5168 030256 042711 001000 BIC      #S1.IFMK,(R1) ;SET EXPD IFMK =0
5169 030262 042711 000400 BIC      #S1.IHER,(R1) ;SET EXPD IHER =0
5170 030266 016261 000002 000002 MOV      2(R2),2(R1) ;SET EXPD WORD #9 = RECV (NOT TESTING)
5171 030274 005000      CLR      RO        ;HIGH RECV ADDRESS FOR CKMSG2
5172 030276 012701 032662 MOV      #T17BFR,R1  ;LOW RECV ADDRESS FOR CKMSG2
5173 030302 012702 030772 MOV      #T17EXP,R2  ;EXPD ADDRESS
5174 030306 012703 000024 MOV      #20.,R3    ;NUMBER OF BYTES TO COMPARE

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 82-2
 TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

```

5175 030312 004737 011322      JSR      PC,CKMSG2      ;EXPD EQUAL RECV?
5176 030316 103404              BCS      100$           ;BR IF YES
5177 030320                      NEXT.ERRNO
5178 030320 92$:      ERRHRD  ERRNO,T177CMP,MSGSTAT ;REPORT ERROR
                    030320 104456              TRAP      CSERHRD
                    030322 000223              .WORD    147
                    030324 032312              .WORD    T177CMP
                    030326 012172              .WORD    MSGSTAT
5179 030330 100$:      CKLOOP              ;LOOP ON ERROR, IF FLAG SET
                    030330 104406              TRAP      CSCLP1
5180
5181      ;      Do a Write Subsystem WRITE NPR to set tape direction out
5182 030332 012700 000100      MOV      #NP.OUT,R0      ;SET TAPE DIRECTION OUT
5183 030336 004737 032464      JSR      PC,T17SNPR      ;SETUP T17PK2 FOR WRITE NPR
5184 030342 012704 033010      MOV      #T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5185 030346 010465 177776      MOV      R4,TSDB(R5)     ;SET THE PACKET ADDRESS TO EXECUTE
5186 030352 004737 017220      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5187 030356 103407              BCS      120$           ;BR IF CARRY SET (GOOD RETURN)
5188 030360 010001              MOV      R0,R1          ;SAVE CONTENTS OF TSSR
5189 030362                      NEXT.ERRNO
5190 030362 112$:      ERRDF  ERRNO,T174SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                    030362 104455              TRAP      CSERDF
                    030364 000224              .WORD    148
                    030366 031431              .WORD    T174SSR
                    030370 011670              .WORD    PKTSSR
5191 030372 004737 020064      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
5192 030376 120$:      CKLOOP              ;LOOP ON ERROR, IF FLAG SET
                    030376 104406              TRAP      CSCLP1
5193
5194      ;      Setup incrementing pattern in FIFO data buffer
5195 030400 012701 031012      MOV      #T17EXSTA,R1    ;EXPD WRITE FIFO DATA BUFFER
5196 030404 012702 000100      MOV      #64.,R2         ;TEST PATTERN SIZE
5197 030410 005000              CLR      R0              ;INCREMENT TEST PATTERN
5198 030412 110021 130$:      MOV      R0,(R1)+        ;STORE INCREMENT TEST BYTE
5199 030414 005200              INC      R0              ;SET NEXT PATTERN
5200 030416 005302              DEC      R2              ;DONE?
5201 030420 003374              BGT      130$           ;BR IF NO
5202
5203      ; REPEAT FOR BYTE COUNT 1 TO 65.
5204 030422 012737 000001 002254 ;      MOV      #1,COUNT    ;GET FIRST BYTE COUNT
5205      ;      Do a Write Subsystem WRITE FIFO with the current byte count
5206 030430 150$:      ;REPEAT LOOP LABEL
5207 030430 013700 002254      MOV      COUNT,R0        ;FIFO BYTE COUNT
5208 030434 012701 031012      MOV      #T17EXSTA,R1    ;FIFO WRITE DATA ADDRESS
5209 030440 004737 032510      JSR      PC,T17WFIF      ;SETUP T17PK2 FOR WRITE FIFO
5210 030444 012704 033010      MOV      #T17PK2,R4      ;GET WRITE SUBSYSTEM COMMAND PACKET
5211 030450 010465 177776      MOV      R4,TSDB(R5)     ;SET THE PACKET ADDRESS TO EXECUTE
5212 030454 004737 017220      JSR      PC,CHKTSSR      ;WAIT FOR SSR TO SET
5213 030460 103407              BCS      160$           ;BR IF CARRY SET (GOOD RETURN)
5214 030462 010001              MOV      R0,R1          ;SAVE CONTENTS OF TSSR
5215 030464                      NEXT.ERRNO
5216 030464 152$:      ERRDF  ERRNO,T175SSR,PKTSSR ;DEVICE FATAL SSR FAILED TO SET
                    030464 104455              TRAP      CSERDF
                    030466 000225              .WORD    149
                    030470 031474              .WORD    T175SSR
                    030472 011670              .WORD    PKTSSR
5217 030474 004737 020064      JSR      PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS

```

CZTUXAO TUBO FRONT FND PRT B MACRO M1200 29-MAR-83 13:32 PAGE 82-3
 TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

```

5218 030500          160$: CKLOOP          :LOOP ON ERROR, IF FLAG SET
      030500 104406          TRAP      C$CLP1
5219
5220          : Do a Write Subsystem Write Misc to Reset FIFO
5221 030502 004737 032442      JSR      PC,I17RSFIF      :SETUP PKT FOR WRITE MISC RESET FIFO
5222 030506 012704 033010      MOV      #T17PK2,R4      :GET WRITE SUBSYSTEM COMMAND PACKET
5223 030512 010465 177776      MOV      R4,TSDB(R5)     :SET THE PACKET ADDRESS TO EXECUTE
5224 030516 004737 017220      JSR      PC,CHKTSSR      :WAIT FOR SSR TO SET
5225 030522 103407          BCS      170$           :BR IF CARRY SET (GOOD RETURN)
5226 030524 010001          MOV      R0,R1           :SAVE CONTENTS OF TSSR
5227 030526          NEXT.ERRNO
5228 030526          162$: ERRDF      ERRNO,T172SSR,PKTSSR :DEVICE FATAL SSR FAILED TO SET
      030526 104455          TRAP      C$SERDF
      030530 000226          .WORD      150
      030532 031320          .WORD      T172SSR
      030534 011670          .WORD      PKTSSR
5229 030536 004737 020064      JSR      PC,FATCHK      :INC AND CHECK FOR MORE THAN 25 ERRORS
5230 030542          170$: CKLOOP          :LOOP ON ERROR, IF FLAG SET
      030542 104406          TRAP      C$CLP1
5231
5232          : Do a Write Subsystem READ STATUS
5233          : If all Tape Status 2 (ICER,IFMK,IHER) flip-flop
5234          : signals NOT=0 Then Print Error
5235 030544 004737 032422      JSR      PC,T17SRD      :SETUP PACKET FOR READ STATUS
5236 030550 012704 033010      MOV      #T17PK2,R4      :GET WRITE SUBSYSTEM COMMAND PACKET
5237 030554 010465 177776      MOV      R4,TSDB(R5)     :SET THE PACKET ADDRESS TO EXECUTE
5238 030560 004737 017220      JSR      PC,CHKTSSR      :WAIT FOR SSR TO SET
5239 030564 103407          BCS      180$           :BR IF CARRY SET (GOOD RETURN)
5240 030566 010001          MOV      R0,R1           :SAVE CONTENTS OF TSSR
5241 030570          NEXT.ERRNO
5242 030570          177$: ERRDF      ERRNO,T173SSR,PKTSSR :DEVICE FATAL JSR FAILED TO SET
      030570 104455          TRAP      C$SERDF
      030572 000227          .WORD      151
      030574 031364          .WORD      T173SSR
      030576 011670          .WORD      PKTSSR
5243 030600 004737 020064      JSR      PC,FATCHK      :INC AND CHECK FOR MORE THAN 25 ERRORS
5244 030604          180$: CKLOOP          :LOOP ON ERROR, IF FLAG SET
      030604 104406          TRAP      C$CLP1
5245 030606 004737 032604      JSR      PC,T17SETEXP     :SET WORDS 0-7 EXPD=RECV (NOT TESTING)
5246 030612 012701 031012      MOV      #T17EXSTA,R1    :GET EXPECTED READ STATUS
5247 030616 012702 032702      MOV      #T17BFSTA,R2    :GET RECV READ STATUS
5248 030622 011211          MOV      (R2),(R1)       :SET EXPD WORD #8 = RECV TEMP
5249 030624 042711 002000      BIC      #S1.ICER,(R1)   :SET EXPD ICER =0
5250 030630 042711 001000      BIC      #S1.IFMK,(R1)   :SET EXPD IFMK =0
5251 030634 042711 000400      BIC      #S1.IHER,(R1)   :SET EXPD IHER =0
5252 030640 016261 000002 000002 MOV      2(R2),2(R1)     :SET EXPD WORD #9 = RECV (NOT TESTING)
5253 030646 005000          CLR      R0             :HIGH RECV ADDRESS FOR CKMSG2
5254 030650 012701 032662      MOV      #T17BFR,R1     :LOW RECV ADDRESS FOR CKMSG2
5255 030654 012702 030772      MOV      #T17EXP,R2     :EXPD ADDRESS
5256 030660 012703 000024      MOV      #20,,R3        :NUMBER OF BYTES TO COMPARE
5257 030664 004737 011322      JSR      PC,CKMSG2      :EXPD EQUAL RECV?
      030670 103404          BCS      200$           :BR IF YES
      030672          NEXT.ERRNO
      030672          192$: ERRHRD      ERRNO,T177CMP,MSGSTAT :REPORT ERROR
      030672 104456          TRAP      C$SERHRD
      030674 000230          .WORD      152
      030676 032312          .WORD      T177CMP

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 82-4
 TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

```

5261 030700 012172
      030702 104406
5262
5263
5264 030704
5265 030704
5266 030714 005237 002254
5267 030720 023727 002254 000101
5268 030726 101002
5269 030730 000137 030430
5270 030734
5271
5272 030734
      030734
      030734 104403
5273
5274 030736 005737 002170
5275 030742 001402
5276 030744 004737 020136
5277 030750 004737 017340
5278 030754 103002
5279 030756 000137 024046
5280 030762
5281 030762
5282
5283 030762
      030762 104432
      030764 002144
5284
5285
    
```

```

200$: CKLOOP
      :LOOP ON ERROR, IF FLAG SET
      TRAP C$CLP1

250$:
    FORCEEXIT 260$
    INC COUNT
    CMP COUNT,#65.
    BHI 260$
    JMP 150$

260$:
    ENDSUB
      :////////// END SUBTEST //////////
      L10044: TRAP C$ESUB

300$:
    TST FATFLG
    BEQ 300$
    JSR PC,CKDROP
    JSR PC,TSTLOOP
    BCC 305$
    JMP T17LOOP
      :ANY FATAL ERRORS ?
      :BRANCH IF NOT
      :TRY TO DROP THE UNIT
      :DO ITERATIONS?
      :BR IF NO
      :LOOP UNTIL ITERATIONS DONE

305$:
    T$PEND:
      :### TEMPORARY END OF TEST

EXIT TST
      :////////// EXIT TEST //////////
      TRAP C$EXIT
      .WORD L10036-.
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 83
 TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

```

5287
5288
5289
5290
5291
5292 030766
5293
5294 030766 377
5295 030767 037
5296 030770 360
5297 030771 000
5298
5299 030772
5300 030772 000000
5301 030774 000000
5302 030776 000000
5303 031000 000000
5304 031002 000000
5305 031004 000000
5306 031006 000000
5307 031010 000000
5308 031012
5309 031114
5310
5311 031114
5312
5313
5314
5315
5316
5317 031216 106 111 106 TST17ID: .ASCIZ 'FIFO Exerciser'
5318
5319 031235 122 105 127 T17RW: .ASCIZ 'REWIND Command Failed'
5320 031263 127 122 111 T17SSR: .ASCIZ 'WRITE CHARACTERISTICS Failed'
5321 031320 127 122 111 T172SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Misc) Failed'
5322 031364 127 122 111 T173SSR: .ASCIZ 'WRITE SUBSYSTEM (Read Status) Failed'
5323 031431 127 122 111 T174SSR: .ASCIZ 'WRITE SUBSYSTEM (Write Mpr) Failed'
5324 031474 127 122 111 T175SSR: .ASCIZ 'WRITE SUBSYSTEM (Write FIFO) Failed'
5325 031540 127 122 111 T176SSR: .ASCIZ 'WRITE SUBSYSTEM (Read FIFO) Failed'
5326 031603 106 111 106 T171CMP: .ASCIZ 'FIFO Status in WORD #9 Incorrect after Initialize'
5327 031665 122 145 141 T172CMP: .ASCIZ 'Read FIFO Data not equal to Write FIFO , Data is in WORD #8'
5328 031761 106 111 106 T173CMP: .ASCIZ 'FIFO Status (In WORD #9) Incorrect after WRITE FIFO'
5329 032045 106 111 106 T174CMP: .ASCIZ 'FIFO Status (In WORD #9) Incorrect after READ FIFO'
5330 032130 122 145 141 T175CMP: .ASCIZ 'Read FIFO Data not equal to Write FIFO Data'
5331 032204 106 111 106 T176CMP: .ASCIZ 'FIFO Status (In WORD #9) Incorrect after READ FIFO from an Empty FIFO'
5332 032312 106 111 106 T177CMP: .ASCIZ 'FIFO Status (In WORD #9) Incorrect after RESET FIFO'
5333
5334
5335
5336
5337
5338 032376
5339 032376
5340 032402 012701 032662
5341 032406 012702 000120
5342 032412 105021
5343 032414 005302

;+
; LOCAL STORAGE FOR THIS TEST
;-

T17MSK:
; MASK OF UNTESTED BITS IN READ STATUS BYTES
; UNTESTED BITS ARE SET TO 1
; BYTE 0 MASK
; BYTE 1 MASK (PARERR, IRESV2, IRESV1)
; BYTE 2 (TIMER A, TIMER B, UNDEFINED<1:0>)
; MAKE IT EVEN
.BYTE ^C<000>
.BYTE ^C<340>
.BYTE ^C<017>
.BYTE 0

T17EXP:
; BEGIN EXPECTED DATA BUFFER
; MESSAGE TYPE
; DATA FIELD LENGTH
; RBPCR
; XST0
; XST1
; XST2
; XST3
; XST4 (ALWAYS PRESENT FOR WRITE SUB.)
; EXPECTED READ STATUS AND WRITE FIFO DATA
; END EXPECTED DATA BUFFER
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.WORD 0
.T17EXSTA: .BLKB 66.
T17EXEND:

T17WFDATA: .BLKB 66.
; WRITE FIFO EXPECTED DATA BUFFER

;+
; LOCAL TEXT MESSAGES FOR TEST
;-

T17CLRBUF:
; CLEAR MESSAGE BUFFER
; SAVE R1-R5 UNTIL NEXT RETURN
; GET MESSAGE BUFFER ADDRESS
; SIZE OF MESSAGE BUFFER IN BYTES
; CLEAR A BYTE
; DONE?
SAVREG
MOV #T17BFR,R1
MOV #T17BEND-T17BFR,R2
108: CLRB (R1)+
DEC R2
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 83-1
 TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

```

5344 032416 003375          BGT      10$          :BR IF NO
5345 032420 000207          RTS      PC          :RETURN
5346
5347
5348      :+
5348      : SETUP T17PK2 PACKET FOR READ STATUS
5349      :-
5350      T17SRD:
5351 032422 004737 032376      JSR      PC,T17CLRBUF      :CLEAR MESSAGE BUFFER
5352 032426 012700 033020      MOV      #T17DT2,R0      :WRITE SUBSYSTEM DATA BUFFER
5353 032432 112720 000005      MOVB     #PW.RDSTATUS,(R0)+ :STORE READ STATUS COMMAND IN BSELO
5354 032436 105010          CLRB     (R0)            :CLEAR BSEL1
5355 032440 000207          RTS      PC          :RETURN
5356
5357
5358      :+
5358      : SETUP T17PK2 PACKET FOR WRITE MISC RESET FIFO
5359      :-
5360      T17RSFIF:
5361 032442 004737 032376      JSR      PC,T17CLRBUF      :CLEAR MESSAGE BUFFER
5362 032446 012700 033020      MOV      #T17DT2,R0      :WRITE SUBSYSTEM DATA BUFFER
5363 032452 112720 000010      MOVB     #PW.WMISC,(R0)+   :STORE WRITE MISCELLANEOUS IN BSELO
5364 032456 112710 000030      MOVB     #MS.RSFIF!MS.RSTAP,(R0) :STORE BSEL1 CLEAR FIFO CODES
5365 032462 000207          RTS      PC          :RETURN
5366
5367
5368      :+
5368      : SETUP T17PK2 PACKET FOR WRITE NPR
5369      : INPUT:
5370      : RO CONTAINS BSEL1 NPR DATA
5371      :
5372      : SETS NP.WRP SINCE IF 0 IT WRITES WRONG PARITY.
5373      :
5374      :-
5375      T17SNPR:
5376 032464 004737 032376      JSR      PC,T17CLRBUF      :CLEAR MESSAGE BUFFER
5377 032470 012701 033020      MOV      #T17DT2,R1      :WRITE SUBSYSTEM DATA BUFFER
5378 032474 112721 000011      MOVB     #PW.WNPR,(R1)+   :STORE WRITE NPR IN BSELO
5379 032500 052700 000020      BIS      #NP.WRP,R0      :DON'T WRITE WRONG PARITY
5380 032504 110011          MOVB     R0,(R1)         :STORE NPR DATA IN BSEL1
5381 032506 000207          RTS      PC          :RETURN
5382
5383
5384      :+
5384      : SETUP T17PK2 PACKET FOR WRITE FIFO
5385      : INPUT:
5386      : RO CONTAINS BYTE COUNT
5387      : R1 CONTAINS DATA PATTERN BLOCK ADDRESS
5388      :
5389      :-
5390      T17WFIF:
5391 032510          SAVREG          :SAVE R1-R5 UNTIL NEXT RETURN
5392 032514 004737 032376      JSR      PC,T17CLRBUF      :CLEAR MESSAGE BUFFER
5393 032520 012702 033020      MOV      #T17DT2,R2      :WRITE SUBSYSTEM DATA BUFFER
5394 032524 112722 000004      MOVB     #PW.WFIFO,(R2)+   :STORE WRITE FIFO IN BSELO
5395 032530 110022          MOVB     R0,(R2)+         :STORE BYTE COUNT IN BSEL1
5396 032532 005022          CLR      (R2)+           :CLEAR SEL2 (UNUSED)
5397 032534 112122 10$:      MOVB     (R1)+,(R2)+       :STORE DATA PATTERN BYTE
5398 032536 005300          DEC      R0              :DONE ALL BYTES?
5399 032540 003375          BGT      10$            :BR IF NO
5400 032542 000207          RTS      PC          :RETURN
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 83-2
 TEST 1: SUBTEST 6; FIFO VERIFY RESET FIFO TEST

```

5401
5402
5403      :+
5404      : SETUP T17PK2 PACKET FOR READ FIFO
5405      :
5406      : INPUT:
5407      :      RO CONTAINS SEL2 BYTE COUNT
5408      :
5408 032544 T17RFIF:
5409 032544 004737 032376      JSR      PC,T17CLRBUF      ;CLEAR MESSAGE BUFFER
5410 032550 012701 033020      MOV      #T17DT2,R1      ;WRITE SUBSYSTEM DATA BUFFER
5411 032554 112721 000003      MOVB    #PW.RFIFO,(R1)+ ;STORE READ FIFO IN BSELO
5412 032560 110021      MOVB    RO,(R1)+      ;STORE BYTE COUNT IN BSEL1
5413 032562 000207      RTS      PC      ;RETURN
5414
5415      :+
5416      : CLEAR EXPECTED DATA MESSAGE BUFFER
5417      :
5417 032564 T17CLEXP:
5418 032564 012701 030772      MOV      #T17EXP,R1      ;GET EXPD ADDRESS
5419 032570 012700 000122      MOV      #T17XEND-T17EXP,RO ;GET EXPD SIZE
5420 032574 105021 10$:      CLRB    (R1)+      ;CLEAR A BYTE
5421 032576 005300      DEC      RO      ;DONE?
5422 032600 003375      BGT      10$      ;BR IF NO
5423 032602 000207      RTS      PC      ;RETURN
5424
5425      :+
5426      :Set WORDS 0-7 of expd message buffer = to recv since not testing
5427      :
5428 032604 T17SETEXP:
5429 032604 012702 030772      MOV      #T17EXP,R2      ;GET EXPD
5430 032610 012703 032662      MOV      #T17BFR,R3      ;GET READ STATUS RECV BUFFER
5431 032614 012700 000010      MOV      #8,RO      ;SET WORDS 0-7 EXP=RECV
5432 032620 012322 5$:      MOV      (R3)+,(R2)+      ;SFT EXPD=RECV
5433 032622 005300      DEC      RO      ;DONE WORDS 0-7 WORDS?
5434 032624 003375      BGT      5$      ;BR IF NO
5435 032626 000207      RTS      PC      ;RETURN
5436
5438 032630      .BLKB  10-<.-TUV2A&7>
5440
5441      :
5442      :WRITE CHARACTERISTICS COMMAND PACKET
5443      :
5443 032640 T17PACKET:
5444 032640 100004      .WORD  100004      ;COMMAND PACKET FOR TEST
5445 032642 032650      .WORD  T17DATA      ;WRITE CHARACTERISTICS COMMAND, WITH ACK
5446 032644 000000      .WORD  0      ;ADDRESS OF CHARACTERISTICS BLOCK
5447 032646 000012      .WORD  10.      ;MINIMUM MESSAGE PACKET SIZE
5448
5449 032650 T17DATA:
5450 032650 032662      .WORD  T17BFR      ;CHARACTERISTICS DATA BLOCK
5451 032652 000000      .WORD  0      ;ADDRESS OF MESSAGE BUFFER
5452 032654 000024      .WORD  20.      ;LENGTH OF MESSAGE BUFFER
5453 032656 000000      .WORD  0      ;ESS,ENB,EAI,ERI
5454 032660 000000      .WORD  0      ;EXTENDED FEATURES UNIT NO. ETC.
5455
5456
5457      :MESSAGE BUFFER FOR ALL TEST 6 COMMANDS
5458
5459 032662 T17BFR:      ;BEGIN MESSAGE BUFFER
    
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 83-3
TEST 1: SUBTEST 6: FIFO VERIFY RESET FIFO TEST

5460 032662 000000
5461 032664 000000
5462 032666 000000
5463 032670 000000
5464 032672 000000
5465 032674 000000
5466 032676 000000
5467 032700 000000
5468 032702
5469 033002
5470
5471
5472
5474 033002
5476 033010
5477 033010 100006
5478 033012 033020
5479 033014 000000
5480 033016 000012
5481
5482 033020
5483 033020 000
5484 033021 000
5485 033022 000000
5486 033024
5487 033126 000000
5488
5489 033130
033130
033130 104401
5490

.WORD 0 ;MESSAGE TYPE
.WORD 0 ;DATA FIELD LENGTH
.WORD 0 ;RBPCR
.WORD 0 ;XST0
.WORD 0 ;XST1
.WORD 0 ;XST2
.WORD 0 ;XST3
.WORD 0 ;XST4 (ALWAYS PRESENT FOR WRITE SUBSYSTEM
T17BFSTA: .BLKB 64. ;READ STATUS AND WRITE FIFO BUFFER
T17BEND: ;END OF MESSAGE BUFFER
:WRITE SUBSYSTEM READ STATUS COMMAND PACKET
:.
T17PK2: .BLKB 10-<.-TUV2A&7>
.WORD P.WRTSUB!P.ACK ;WRITE SUBSYSTEM WITH ACK
.WORD T17DT2 ;LOW ADDRESS OF DATA BLOCK
.WORD 0 ;HIGH ADDRESS OF DATA BLOCK
.WORD 10. ;MINIMUM MESSAGE PACKET SIZE
T17DT2: ;DATA BLOCK
.BYTE 0 ;BSEL0
.BYTE 0 ;BSEL1
.WORD 0 ;SEL2
.BLKB 66. ;WRITE FIFO DATA OUTPUT BUFFER
T17DLY: .WORD 0 ;HOLDS DELAY VALUE
ENDTST

L10036: TRAP CSETST

CZTUXAO TUBO FRONT END PRT B
TEST 2: INITIALIZE #2 TEST

MACRO M1200 29-MAR-83 13:32 PAGE 84

5492
5493
5494
5495
5496
5497
5498
5499

.SBTTL TEST 2: INITIALIZE #2 TEST

:+
:THIS TEST VERIFIES THAT WRITING INTO THE TSSR RETURNS THE
:CONTROLLER TO ITS INITIALIZED STATE FROM VARIOUS CONDITIONS
:-

BGNTST

T2::

5500 033132 005037 002170
5501 033136 005037 003100
5502 033142 012737 005755 002146

CLR FATFLG ;CLEAR FATAL ERROR FLAG
CLR KTFLG ;HOLD OFF KT11
MOV #EPRT1,EPRTSW ;SET UP PRIMARY ERROR MESSAGE

5503
5504
5505
5506
5507
5508
5509

:TEST 1
:-

5515 033150 004737 017040
5516 033154 012700 034006
5517 033160 004737 017372
5518 033164 012737 000002 002164

JSR PC,DSBINT ;DISABLE INTERRUPTS
MOV #TST21ID,R0 ;ASCII MESSAGE TO IDENTIFY TEST
JSR PC,TSTSETUP ;DO INITIAL TEST SETUP
MOV #2,LOOPCNT ;PERFORM 2 ITERATIONS

T21LOOP:

5519 033172
5520 033172 004737 034030
5521 033176 004737 034120

JSR PC,T21REST ;SET COMMAND PACKET
JSR PC,T21RT2 ;SET UP OTHER COMMAND PACKET

5522
5523
5524
5525
5526
5527

:*****
:ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
:*****

5528
5529 033202 012737 176750 033566
5530 033210 004737 016630
5531 033214 103426
5532 033216

MOV #65000.,T21DLY ;SET DELAY ROUTINE
11\$: JSR PC,SOFINIT ;DO INITIALIZE ON CONTROLLER
BCS 20\$;BR IF INIT WAS OK
DELAY 250 ;DELAY FOR A REWIND TO FINISH

033216 012727 000250
033222 000000
033224 013727 002116
033230 000000
033232 005367 177772
033236 001375
033240 005367 177756
033244 001367

MOV #250,(PC)+
.WORD 0
MOV LSDLY,(PC)+
.WORD 0
DEC -6(PC)
BNE -4
DEC -22(PC)
BNE -20

5533 033246 005337 033566
5534 033252 001356
5535 033254 004737 020064
5539 033260 010001
5540 033262

DEC T21DLY ;BUMP COUNTER DOWN
BNE 11\$;BR, IF MORE TIME TO GO
JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
MOV R0,R1 ;CONTENTS OF TSSR REGISTER
ERRDF ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK

033262 104455
033264 000311
033266 003550
033270 011656

TRAP CSERDF
.WORD 201
.WORD SFIERR
.WORD SFIMSG

5541 033272
5542 033272 012704 033440

20\$:

MOV #T21PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS

CZTUXAO TUBO FRONT END PRT B
TEST 2: INITIALIZE #2 TEST

MACRO M1200 29-MAR-83 13:32 PAGE 84-1

```

5543
5544
5545
5546
5547
5548
5549
5550 033276 004737 010322      JSR      PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
5551 033302 103407              BCS      23$           ;BR, IF COMMAND ISSUED OK
5552 033304 004737 020064      JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
5556 033310 010001              MOV      R0,R1         ;SAVE CONTENTS OF TSSR
5557 033312              ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
                                TRAP      C$ERHRD
                                .WORD    202
                                .WORD    WRTMSG
                                .WORD    SFIMSG
5558 033322              23$:
5559 033322 012765 000000 000000      MOV      #0,TSSR(R5)   ;ISSUE A SOFT INITIALIZE
5560 033330 004737 017104              JSR      PC,WAITF      ;WAIT FOR JUST THE SSR BIT TO SET
5561 033334 016501 000000      MOV      TSSR(R5),R1   ;READ THE TSSR BACK
5562 033340 010102              MOV      R1,R2         ;WORK REGISTER
5563 033342 042702 176377      BIC      #^C<HIADDR>,R2 ;CLEAR OUT OTHER BITS
5564 033346 052702 002200      BIS      #SSR!NBA,R2  ;SOME OF THE BITS THAT SHOULD BE SET
5565 033352 032701 000100      BIT      #OFL,R1      ;IS OFF LINE BIT SET
5566 033356 001012              BNE      38$           ;BR, IF DRIVE IS OFF LINE
5567 033360 020102              35$:
5568 033362 001406              CMP      R1,R2         ;EXPECTED (R2) = RECEIVED (R1)
5569 033364 004737 020064      BEQ      37$           ;BK, IF THEY ARE EQUAL (OK)
5573 033370              JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
                                ERRHRD  ERRNO,T21AM3,EXPREC ;'ERROR TRYING TO INIT AFTER WRITE MISC.
                                TRAP      C$ERHRD
                                .WORD    203
                                .WORD    T21AM3
                                .WORD    EXPREC
5574 033400              37$:
5575 033402 000406              CKLOOP              ;LOOP IF SELECTED
                                TRAP      C$CLP1
5576 033404              BR      40$           ;SKIP OVER OFF-LINE STUFF
5580 033404              38$:
5581 033404 104455              ERRDF  ERRNO,T21OFL,EXPREC ;DRIVE IS OFF LINE
                                TRAP      C$ERDF
                                .WORD    204
                                .WORD    T21OFL
                                .WORD    EXPREC
5581 033414 004737 020136      JSR      PC,CKDROP     ;TRY AND DROP UNIT
5582 033420 004737 017340      JSR      PC,TSTLOOP    ;DO WE NEED TO ITERATE TEST
5583 033424 103002              BCC      63$           ;BR, IF NO LOOP REQUIRED
5584 033426 000137 033172      JMP      T21LOOP       ;EXECUTE AGAIN
5585 033432              63$:
5586 033432 104432              EXIT      TST         ;ALL DONE THIS TEST
                                TRAP      C$EXIT
                                .WORD    L10045-

```

CZTUXAO TUBO FRONT END PRT B
 TEST 2: INITIALIZE #2 TEST

MACRO M1200 29-MAR-83 13:32 PAGE 85

```

5587
5588
5589
5591 033436
5593 033440
5594 033440 100004
5595 033442 033450
5596 033444 000000
5597 033446 000012
5598 033450
5599 033450 033460
5600 033452 000000
5601 033454 000024
5602 033456 000000
5603 033460
5604
5605
5606
5608 033542
5610 033550
5611 033550 100206
5612 033552 033560
5613 033554 000000
5614 033556 000006
5615
5616
5617 033560
5618 033560 000
5619 033561 000
5620 033562 000000
5621 033564 000000
5622 033566 000000
5623
5624
5625

;+
;LOCAL STORAGE FOR THIS TEST
;-
      .BLKB 10-<.-TUV2A&7>
T21PACKET:
      .WORD 100004
      .WORD T21DATA
      .WORD 0
      .WORD 10.
T21DATA:
      .WORD T21BFR
      .WORD 0
      .WORD 20.
      .WORD 0
T21BFR: .BLKW 25.

;WRITE SUBSYSTEM MEMORY COMMAND PACKET
;
      .BLKB 10-<.-TUV2A&7>
T21PK2:
      .WORD 100206
      .WORD T21BF2
      .WORD 0
      .WORD 6.

      .EVEN
T21BF2:
T21BS0: .BYTE 0
T21BS1: .BYTE 0
T21S2: .WORD 0
T21S3: .WORD 0
T21DLY: .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

;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
;DELAY COUNTER
  
```

```

5627
5628
5629          ;+
5630          ;LOCAL TEXT MESSAGES FOR TEST
5631          ;-
5632 033570    127    122    111  T21SSR: .ASCIZ  'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
5633 033665    124    123    123  T21AM3: .ASCIZ  'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
5634 033765    104    162    151  T21OFL: .ASCIZ  'Drive is OFFLINE'
5635 034006    111    156    151  TST21ID: .ASCIZ  'Init'alization #2'
5636          .EVEN
5637          ;+
5638          ;
5639          ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
5640          ;WRITE SUBSYSTEM MEMORY COMMAND
5641          ;
5642          ;-
5643
5644 034030    T21REST:
5645 034030          SAVREG          ;SAVE THE REGISTERS
5646 034034    012701  033440    MOV          #T21PACKET,R1      ;START OF THE PACKET
5647 034040    012721  100004    MOV          #100004,(R1)+      ;WRITE SUBSYSTEM MEM. WITH ACK,
5648 034044    012721  033450    MOV          #T21DATA,(R1)+    ;ADDRESS OF CHARAIST:CS DATA BLOCK
5649 034050    005021          CLR          (R1)+            ;EXTENDED ADDRESS
5650 034052    012721  000010    MOV          #8.,(R1)+        ;SIZE OF DATA BLOCK IN BYTES
5651 034056    012721  033460    MOV          #T21BFR,(R1)+    ;ADDRESS OF MESSAGE BUFFER
5652 034062    005021          CLR          (R1)+
5653 034064    012721  000024    MOV          #20.,(R1)+      ;LENGTH OF MESSAGE BUFFER
5654 034070    005021          CLR          (R1)+
5655 034072    005011          CLR          (R1)
5656 034074    012702  000020    MOV          #20,R2          ;NUMBER OF LOCATIONS TO BE CLEARED
5657 034100    012762  177777  033460  64$:  MOV          #177777,T21BFR(R2) ;ALL ONES TO MESSAGE BUFFER
5658 034106    005742          TST          -(R2)           ;NEXT LOCATION
5659 034110    020227  000000    CMP          R2,#0           ;CHECK R2 FOR ZERO
5660 034114    001371          BNE          64$            ;BR, IF NOT AT ZERO YET
5661 034116    000207          RTS          PC             ;RETURN
5662 034120
5663 034120    T21RT2:
5664 034124    012701  033550    SAVREG          ;SAVE THE REGISTERS
5665 034130    012721  100206    MOV          #T21PK2,R1      ;START OF THE PACKET
5666 034134    012721  033560    MOV          #100206,(R1)+    ;WRITE SUBSYSTEM MEM. WITH ACK, IE
5667 034140    005021          MOV          #T21BF2,(R1)+    ;ADDRESS OF DATA BLOCK
5668 034142    012721  000006    CLR          (R1)+            ;EXTENDED ADDRESS
5669 034146    005021          MOV          #6.,(R1)+        ;SIZE OF DATA BLOCK IN BYTES
5670 034150    012701  033560    MOV          #T21BF2,R1      ;ADDRESS OF DATA FOR WRT SUB SYS MEM
5671 034154    005021          CLR          (R1)+
5672 034156    005011          CLR          (R1)
5673 034160    000207          RTS          PC             ;RETURN
5674 034162          ENDTST
          034162    104401
          L10045: TRAP C$ETST

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 87
TEST 3: OFF-LINE AND REJECT REWIND

.SBTTL TEST 3: OFF-LINE AND REJECT REWIND

5676
5677
5678
5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5697
5698
5699
5700
5701
5702

:+
:
: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. THE TEST CONSISTS OF THE FOLLOWING ONE SUBTEST
:
:-

BGNTST

034164
034164
034164 005037 002170
034170 005037 003100
034174 012737 005755 002146
034202 004737 017040
034206 012700 035322
034212 004737 017372
034216 012737 000002 002164

CLR FAYFLG
CLR KTFLG
MOV #EPRT1,EPRTSW
JSR PC,DSBINT
MOV #TST22ID,RO
JSR PC,TSTSETUP
MOV #2,LOOPCNT

T3::
:CLEAR FATAL ERROR FLAG
:HOLD OFF KT11
:SET UP PRIMARY ERROR MESSAGE
:DISABLE INTERRUPTS
:ASCII MESSAGE TO IDENTIFY TEST
:DO INITIAL TEST SETUP
:PERFORM 2 ITERATIONS

:+
:
:

CZTUXAO TUBO FRONT END FRT B MACRO M1200 29-MAR-83 13:32 PAGE 88-1
TEST 3: OFF-LINE AND REJECT REWIND

5757 034350 004737 020064
5761 034354
034354 104456
034356 000457
034360 035174
034362 016330
5762 034364
034364 104406
5763 034366 012703 034512
5764 034372 016301 000006
5765 034376 010102
5766 034400 042702 000020
5767 034404 020102
5768 034406 001406
5769 034410 004737 020064
5773 034414
034414 104456
034416 000460
034420 035247
034422 016330
5774 034424
5775 034424
034424
034424 104403
5776 034426 023727 002170 000031
5777 034434 002402
5778 034436 004737 020136
5779 034442
5780 034442 004737 017340
5781 034446 103002
5782 034450 000137 034224
5783 034454
034454 104432
034456 001036

80\$: CKLOOP

JSR PC,FATCHK
ERRHRD ERRNO,T22RWJ,EXPREC

MOV #T22BFR,R3
MOV XST0(R3),R1
MOV R1,R2
BIC #BIT4,R2
CMP R1,R2
BEQ 90\$
JSR PC,FATCHK
ERRHRD ERRNO,T22VCK,EXPREC

:INC AND CHECK FOR MORE THAN 25 ERRORS
:TSSR INCORRECT AFTER TAPE MOTION CMD
TRAP C\$ERHRD
.WORD 303
.WORD T22RWJ
.WORD EXPREC

:LOOP IF SELECTED
TRAP C\$CLP1
:POINTER TO MESSAGE BUFFER
:PICK UP XST0 FROM MESSAGE BUFFER
:SET UP EXPECTED
:VCK SHOULD BE CLEAR
:ARE THEY EQUAL
:BR, IF OK (GOOD)
:INC AND CHECK FOR MORE THAN 25 ERRORS
:VCK WASN'T CLEAR (BAD)
TRAP C\$ERHRD
.WORD 304
.WORD T22VCK
.WORD EXPREC

90\$: ENDSUB

CMP FATFLG,#25.
BLT 999\$
JSR PC,CKDROP

JSR PC,TSTLOOP
BCC 163\$
JMP T22LOOP
163\$: EXIT TST

:>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
L10047:
TRAP C\$ESUB
: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 L10046-

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 89
 TEST 3: OFF-LINE AND REJECT REWIND

5785			;		
5786			;	LOCAL STORAGE FOR THIS TEST	
5787			;		
5789	034460			.BLKB	10-<.-TUV2AB7>
5791	034470		T22PACKET:		:COMMAND PACKET FOR TEST
5792	034470	100204		.WORD	100204
5793	034472	034500		.WORD	T22DATA
5794	034474	000000		.WORD	0
5795	034476	000012		.WORD	10.
5796	034500		T22DATA:		:STARTING VALUE OF BLOCK SIZE
5797	034500	034512		.WORD	T22BFR
5798	034502	000000		.WORD	0
5799	034504	000024		.WORD	20.
5800	034506	000000		.WORD	0
5801	034510	000007		.WORD	7
5802	034512		T22BFR:	.BLKB	25.
5803			;		
5804			;	WRITE SUBSYSTEM MEMORY COMMAND PACKET	
5805			;		
5807	034574			.BLKB	10-<.-TUV2AB7>
5809	034600		T22PK2:		
5810	034600	100206		.WORD	100206
5811	034602	034610		.WORD	T22BF2
5812	034604	000000		.WORD	0
5813	034606	000006		.WORD	6.
5814					
5815				.EVEN	
5816	034610		T22BF2:		
5817	034610	000	T22BS0:	.BYTE	0
5818	034611	000	T22BS1:	.BYTE	0
5819	034612	000000	T22S2:	.WORD	0
5820	034614	000000	T22S3:	.WORD	0
5821			;		
5822			;		
5823				.EVEN	
5824			;	TAPE MOTION PACKET COMMAND VALUES	
5825	034616	100201	T22RD:	.WORD	100201
5826	034620	100205	T22WRT:	.WORD	100205
5827	034622	100210	T22POS:	.WORD	100210
5828	034624	100211	T22FOR:	.WORD	100211
5829	034626	177777		.WORD	177777
5830					
5831					

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 90
 TEST 3: OFF-LINE AND REJECT REWIND

```

5833
5834
5835          ;+
5836          ;LOCAL TEXT MESSAGES FOR TEST
5837          ;-
5838 034630    127    122    111 T22SSR: .ASCIZ 'WRITE MISCELLANEOUS CONTROL/READ STATUS Command Not Accepted'
5839 034725    124    123    123 T22AM3: .ASCIZ 'TSSR Init. Failed After WRITE MISCELLANEOUS CONRTOL/READ STATUS'
5840 035025    104    162    151 T22OFL: .ASCIZ 'Drive 7 Select Failed To Set 'OFL' In TSSR'
5841 035100    124    123    123 T22TM: .ASCIZ 'TSSR Incorrect After Tape Motion Command To Off-Line Device'
5842 035174    124    123    123 T22RWJ: .ASCIZ 'TSSR Not Correct After REWIND With VCK Set'
5843 035247    103    126    103 T22VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
5844 035322    117    146    146 T22ID:  .ASCIZ 'Off-Line And Reject Rewind'
5845          .EVEN
5846          ;+
5847          ;
5848          ;ROUTINE TO RESTORE COMMAND PACKET TO START-UP (DEFAULT) VALUES
5849          ;WRITE SUBSYSTEM MEMORY COMMAND
5850          ;
5851          ;-
5852
5853          T22REST:
5854          SAVREG          ;SAVE THE REGISTERS
5855          MOV #T22PACKET,R1 ;START OF THE PACKET
5856          MOV #100204,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK, IE
5857          MOV #T22DATA,(R1)+ ;ADDRESS OF CHARAISTICS DATA BLOCK
5858          CLR (R1)+          ;EXTENDED ADDRESS
5859          MOV #10.,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
5860          MOV #T22BFR,(R1)+   ;ADDRESS OF MESSAGE BUFFER
5861          CLR (R1)+
5862          MOV #20.,(R1)+      ;LENGTH OF MESSAGE BUFFER
5863          CLR (R1)+
5864          MOV #7,(R1)         ;SELECT DRIVE SEVEN
5865          MOV #20,R2          ;NUMBER OF LOCATIONS TO BE CLEARED
5866          MOV #177777,T22BFR(R2) ;ALL ONES TO MESSAGE BUFFER
5867          TST -(R2)          ;BUMP R2 DOWN
5868          CMP R2,#0          ;IS R2 AT ZERO YET
5869          BNE 64$           ;KEEP GOING UNTIL DONE
5870          RTS PC           ;RETURN
5871
5872          T22RT2:
5873          SAVREG          ;SAVE THE REGISTERS
5874          MOV #T22PK2,R1    ;START OF THE PACKET
5875          MOV #100206,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK, IE
5876          MOV #T22BF2,(R1)+ ;ADDRESS OF DATA BLOCK
5877          CLR (R1)+          ;EXTENDED ADDRESS
5878          MOV #6.,(R1)+      ;SIZE OF DATA BLOCK IN BYTES
5879          MOV #T22BF2,R1    ;POINT TO DATA SEL AREA
5880          CLR (R1)+
5881          CLR (R1)
5882          CLR (R1)          ;LAST LOC TO BE CLEARED
5883          RTS PC           ;RETURN
5884          ENDTST
          035514    104401
          035514
  
```

L10046.

TRAP CBETSI

CZTUXAO TUBO FRONT END PRT B
TEST 4: BASIC WRITE DATA

MACRO M1200 29-MAR-83 13:32 PAGE 91

.SBTTL TEST 4: BASIC WRITE DATA

5886
5887
5888
5889
5890
5891
5892
5893
5894
5895
5896
5897
5898
5899

:+
:
: 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.
:
: THE TEST CONSISTS OF THE FOLLOWING 5 SUBTESTS
:
:
:
:-

5900 035516

BGNTST

035516

5901 035516 005037 002170

CLR FATFLG

T4::
: CLEAR FATAL ERROR FLAG

5902 035522 005037 003100

CLR KFLG

: HOLD OFF KT11

5903 035526 012737 005755 002146

MOV #EPRT1,EPRTSW

: SET UP PRIMARY ERROR MESSAGE

5904 035534 005037 003102

CLR KTENABLE

: TURN OFF KT11

5905 035540 004737 020230

JSR PC,KTOFF

: TURN OFF KT11

5910 035544 004737 017040

JSR PC,DSBINT

: DISABLE INTERRUPTS

5911 035550 012700 041251

MOV #TST23ID,RO

: ASCII MESSAGE TO IDENTIFY TEST

5912 035554 004737 017372

JSR PC,TSTSETUP

: DO INITIAL TEST SETUP

5913 035560 012737 000001 002164

MOV #1,LOOPCNT

: PERFORM 1 ITERATIONS

5914
5915

:+
:
:

CZTUXAO TUBO FRONT END PRT B
TEST 4: BASIC WRITE DATA

MACRO M1200 29-MAR-83 13:32 PAGE 92-1

```

5974 035662 010001          MOV      R0,R1          ;SAVE CONTENTS OF TSSR
5975 035664          ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTICS FAILED
      035664 104456          TRAP      CSERHRD
      035666 000622          .WORD    402
      035670 004754          .WORD    WRTMSG
      035672 011656          .WORD    SFIMSG
5976 035674          23$:   CKLOOP          ;LOOP IF SELECTED
      035674 104406          TRAP      CSCLP1
5977
5978      ;*****
5979      ;
5980      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
5981      ;
5982      ;*****
5983
5984 035676 004737 010424          JSR      PC,REWIND      ;CALL THE TAPE REWIND
5985 035702 012703 000024          MOV      #20.,R3       ;STARTING RECORD SIZE
5986 035706 013737 003072 040002 65$:   MOV      FREE,T23WB     ;STARTING WRITE BUFFER ADDRESS
5987
5988      ;*****
5989      ;
5990      ;WRITE DATA,CVC=1,ACK COMMAND
5991      ;
5992      ;*****
5993
5994 035714 012737 140005 040000          MOV      #140005,T23PK3 ;WRITE DATA,CVC=1,ACK COMMAND
5995 035722 012704 040000          MOV      #T23PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
5996 035726 010300          MOV      R3,R0         ;SET PATTERN IN CORRECT REGISTER
5997 035730 004737 020356          JSR      PC,FILLMEM     ;FILL MEMORY WITH RECORD SIZE
5998 035734 010337 040006          MOV      R3,T23SZ      ;SET UP RECORD SIZE IN PACKET
5999 035740 010465 177776          MOV      R4,TSDB(R5)   ;ISSUE COMMAND
6000 035744 004737 017104          JSR      PC,WAITF      ;WAIT FOR SSR TO SET
6001 035750 016501 000000          MOV      TSSR(R5),R1   ;GET TSSR CONTENTS
6002 035754 012702 000200          MOV      #SSR,R2      ;SET UP EXPECTED
6003 035760 020102          CMP      R1,R2         ;ARE THEY EQUAL
6004 035762 001406          BEQ      80$           ;BR, IF OK
6005 035764 004737 020064          JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
6010 035770          ERRHRD  ERRNO,WRTERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      035770 104456          TRAP      CSERHRD
      035772 000623          .WORD    403
      035774 005011          .WORD    WRTERR
      035776 011670          .WORD    PKTSSR
6011 036000          80$:   CKLOOP          ;LOOP IF SELECTED
      036000 104406          TRAP      CSCLP1
6012 036002 016501 177776          MOV      TSBA(R5),R1   ;GET TSBA CONTENTS
6013 036006 012702 040000          MOV      #T23PK3,R2   ;SET UP EXPECTED
6014 036012 020102          85$:   CMP      R1,R2     ;ARE THEY EQUAL
6015 036014 001406          BEQ      90$           ;BR, IF TSBA IS CORRECT
6016 036016 004737 020064          JSR      PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
6020 036022          ERRHRD  ERRNO,T23BA,EXPREC ;TSBA WAS NOT CORRECT AFTER WRITE DATA
      036022 104456          TRAP      CSERHRD
      036024 000624          .WORD    404
      036026 041110          .WORD    T23BA
      036030 016330          .WORD    EXPREC
6021 036032          90$:   CKLOOP          ;LOOP IF SELECTED
      036032 104406          TRAP      CSCLP1
6022 036034 062703 001750          115$:  ADD      #1000.,R3 ;NEXT RECORD SIZE/DATA PATTERN

```

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:52 PAGE 92-2
TEST 4: BASIC WRITE DATA

```

6023 036040 020337 040010          CMP    R3,T23RSZ          ;IS R3 OVER MAX RECORD SIZE
6024 036044 002005                 BGE    1208              ;IF RECORD SIZE IS TOO BIG QUIT
6025 036046 020327 177776          CMP    R3,#65534.       ;END OF SUBTEST MAX RECORD SIZE
6026 036052 001402                 BEQ    1208              ;BR, IF COMPLETED
6027 036054 000137 035706          JMP    658               ;DO MORE RECORDS
6028 036060                       1208:
6029 036060 004737 041422          JSR    PC,T23RT3        ;RESTORE PACKET
6030 036064 004737 041360          JSR    PC,T23RT2        ;CLEAN UP PACKET
6031 036070 004737 010424          JSR    PC,REWIND        ;ISSUE REWIND COMMAND WITH WAIT
6032 036074 103407                 BCS    1308             ;BR, IF TSSR IS OK (GOOD)
6033 036076 010001                 MOV    R0,R1            ;SAVE TSSR CONTENTS
6034 036100 004737 020064          JSR    PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
6038 036104                   ERRHRD  ERRNO,T23RWN,PKTSSR ;TSSR IS INCORRECT AFTER REWIND
                                TRAP    C$ERHRD
                                .WORD   405
                                .WORD   T23RWN
                                .WORD   PKTSSR
6039 036114                       1308:
6040 036114                   ENDSUB                  ;>>>>>>>>>> END SUBTEST >>>>>>>>>>
                                L10051:
                                TRAP    C$ESUB
6041 036116 104403 023727 002170 000031  CMP    FATFLG,#25.      ;IS ERROR COUNT AT 25
6042 036124 002402                 BLT    9998              ;BR, IF LESS THAN 25
6043 036126 004737 020136          JSR    PC,CKDROP        ;TRY TO DROP THE UNIT
6044 036132                       9998:

```


CZTUXAO TUBO FRONT END PRT B
TEST 4: BASIC WRITE DATA

MACRO M1200 29-MAR-83 13:32 PAGE 93-1

```

6099          :WRITE DATA,CVC=1,ACK,SWB COMMAND
6100          :
6101          :*****
6102          :
6103 036232 012737 150005 040000      MOV      #150005,T23PK3      :WRITE DATA,CVC=1,ACK,SWB COMMAND
6104 036240 012704 040000              MOV      #T23PK3,R4        :SET UP R4 WITH PACKET ADDRESS
6105 036244 010300              MOV      R3,R0            :SET PATTERN IN CORRECT REGISTER
6106 036246 004737 020356              JSR      PC,FILLMEM        :FILL MEMORY WITH RECORD SIZE
6107 036252 010337 040006              MOV      R3,T23SZ         :SET UP RECORD SIZE IN PACKET
6108 036256 010465 177776              MOV      R4,TSDB(R5)      :ISSUE COMMAND
6109 036262 004737 017104              JSR      PC,WAITF         :WAIT FOR SSR TO SET
6110 036266 016501 000000              MOV      TSSR(R5),R1     :GET TSSR CONTENTS
6111 036272 012702 000200              MOV      #SSR,R2         :SET UP EXPECTED
6112 036276 020102              CMP      R1,R2           :ARE THEY EQUAL
6113 036300 001406              BEQ      80$             :BR, IF OK
6114 036302 004737 020064              JSR      PC,FATCHK        :INC AND CHECK FOR MORE THAN 25 ERRORS
6118 036306              ERRHRD  ERRNO,WRterr,PKTSSR :TSSR INCORRECT AFTER WRITE DATA
                                TRAP  C$ERHRD
                                .WORD 408
                                .WORD WRterr
                                .WORD PKTSSR
6119 036316              80$:  CKLOOP          :LOOP IF SELECTED
                                TRAP  C$CLP1
                                .WORD 409
                                .WORD T23BA
                                .WORD EXPREC
6120 036320 016501 177776              MOV      TSBA(R5),R1     :GET TSBA CONTENTS
6121 036324 012702 040000              MOV      #T23PK3,R2     :SET UP EXPECTED
6122 036330 020102              85$:  CMP      R1,R2     :ARE THEY EQUAL
6123 036332 001406              BEQ      90$             :BR, IF TSBA IS CORRECT
6124 036334 004737 020064              JSR      PC,FATCHK        :INC AND CHECK FOR MORE THAN 25 ERRORS
6128 036340              ERRHRD  ERRNO,T23BA,EXPREC :TSBA WAS NOT CORRECT AFTER WRITE DATA
                                TRAP  C$ERHRD
                                .WORD 409
                                .WORD T23BA
                                .WORD EXPREC
6129 036350              90$:  CKLOOP          :LOOP IF SELECTED
                                TRAP  C$CLP1
6130 036352 020327 007376              CMP      R3,#7376        :ONLY CHECK RAM UNTIL ITS FULL
6131 036356 002057              BGE      115$            :IT WRAPS AROUND ETC.
6132 036360 004737 041360              JSR      PC,T23RT2        :MAKE SURE PACKET AND DATA ARE CLEAN
6133 036364 012737 000400 040014      MOV      #256,T23S2      :STARTING RAM ADDRESS
6134 036372 112737 000000 040012      MOVB     #0,T23BS0       :STOP INTERNAL TUV05 DIAGNOSTICS
6135 036400 112737 000000 040013      MOVB     #0,T23BS1       :SIZE OF RAM READ
6136 036406 012704 037760              MOV      #T23PK2,R4     :SET R4 WITH PACKET ADDRESS
6137 036412 010465 177776              MOV      R4,TSDB(R5)    :ISSUE WRITE SUB SYS MEM COMMAND
6138 036416 004737 017220              JSR      PC,CHKTSSR      :CHECK TSSR AND WAIT FOR SSR TO SET
6139 036422 103407              BCS     92$             :BR, IF NO ERRORS IN TSSR
6140 036424 010001              MOV      R0,R1           :SAVE TSSR
6141 036426 004737 020064              JSR      PC,FATCHK        :INC AND CHECK FOR MORE THAN 25 ERRORS
6145 036432              ERRHRD  ERRNO,T23WSS,PKTSSR :TSSR BAD AFTER WRITE SUB SYS MEM
                                TRAP  C$ERHRD
                                .WORD 410
                                .WORD T23WSS
                                .WORD PKTSSR
6146 036442              92$:  CKLOOP          :LOOP IF SELECTED
                                TRAP  C$CLP1
                                .WORD 411
                                .WORD T23WSS
6147 036444 004737 041360              JSR      PC,T23RT2        :MAKE SURE PACKET AND DATA ARE CLEAN
6148 036450 012737 000400 040014      MOV      #256,T23S2      :STARTING RAM ADDRESS
6149 036456 112737 000001 040012      MOVB     #1,T23BS0       :READ RAM COMMAND FOR WRITE SUB SYS M.

```



```

6183          :+
6184          :
6185          :TEST 4, SUBTEST 3
6186          :
6187          :VERIFIES THAT A WRITE COMMAND WITH AN ILLEGAL MODE
6188          :FIELD OR AN ILLEGAL BUFFER ADDRESS IS REJECTED WITH
6189          :THE PROPER ERROR STATUS AND THAT TAPE DOES NOT MOVE
6190          :
6191          :-
6192          :
6193          :
6194 036610      BGNSUB                :>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
        036610      T4.3:
        036610 104402      TRAP      CSBSUB
6195 036612 004737 041266      JSR      PC,T23REST      :SET COMMAND PACKET
6196 036616 004737 041422      JSR      PC,T23RT3      :RESTORE PACKET
6197 036622 004737 041360      JSR      PC,T23RT2      :SET UP OTHER COMMAND PACKET
6198          :
6199          :*****
6200          :ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
6201          :
6202          :*****
6203          :
6204          :
6205 036626 004737 016630      JSR      PC,SOFINIT      :DO INITIALIZE ON CONTROLLER
6206 036632 103407      BCS      20$      :BR IF INIT WAS OK
6207 036634 004737 020064      JSR      PC,FATCHK      :INC AND CHECK FOR MORE THAN 25 ERRORS
6211 036640 010001      MOV      R0,R1      :CONTENTS OF TSSR REGISTER
6212 036642      ERRDF      ERRNO,SFIERR,SFIMSG      :FATAL ERROR TSSR WAS NOT OK
        036642 104455      TRAP      CSERDF
        036644 000635      .WORD      413
        036646 003550      .WORD      SFIERR
        036650 011656      .WORD      SFIMSG
6213 036652      20$:
6214 036652 012704 037650      MOV      #T23PACKET,R4      :SUBROUTINE NEEDS PACKET ADDRESS
6215          :
6216          :*****
6217          :
6218          :WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
6219          :
6220          :*****
6221          :
6222 036656 004737 010322      JSR      PC,WRTCHR      :ISSUE WRITE CHARACTERISTICS
6223 036662 103407      BCS      23$      :BR, IF COMMAND ISSUED OK
6224 036664 004737 020064      JSR      PC,FATCHK      :INC AND CHECK FOR MORE THAN 25 ERRORS
6228 036670 010001      MOV      R0,R1      :SAVE CONTENTS OF TSSR
6229 036672      ERRHRD      ERRNO,WRTMSG,SFIMSG      :WRITE CHARACTERISTICS FAILED
        036672 104456      TRAP      CCERHRD
        036674 000636      .WORD      414
        036676 004754      .WORD      WRTMSG
        036700 011656      .WORD      SFIMSG
6230          :
6231          :*****
6232          :
6233          :WRITE DATA, ACK, ILLEGAL BITS
6234          :
6235          :*****

```


CZTUXAO TUBO FRONT END PRT B
TEST 4: BASIC WRITE DATA

MACRO M1200 29 MAR-83 13:32 PAGE 95-1

```

6313                                     ;WRITE DATA, ACK, CVC=1
6314                                     ;.....
6315                                     ;.....
6316                                     ;.....
6317                                     ;.....
6318 037122 005037 050216          23$: CLR   T24DLY          ;SET EXTENDED ADDRESS BITS TO 0
6319 037126 012737 140005    040000 25$: MOV   #140005,T23PK3 ;WRITE DATA, ACK, CVC=1
6320 037134 012701 160000          MOV   #160000,R1 ;START POSSIBLE NXM ADDRESS
6321 037140 012702 177776          MOV   #177776,R2 ;END POSSIBLE NXM ADDRESS
6322 037144 004737 017260          JSR   PC,XNXM    ;CALL NXM FINDER ROUTINE
6323 037150 103402              BCS   76$        ;BR IF NXM ADDRESS FOUND
6324 037152 000137 037266          JMP   90$        ;JMP OVER CAN'T FIND NXM
6325 037156 010137 040002          76$: MOV   R1,T23WB ;SET UP WRITE BUFFER ADDRESS
6326 037162 013737 050216    040004 MOV   T24DLY,T23WB+2 ;HIGH ORDER ADDRESS BITS
6327 037170 012737 000100    040006 MOV   #64.,T23SZ  ;SET UP BUFFER SIZE
6328 037176 012704 040000          MOV   #T23PK3,R4 ;R4 = POINTER TO PACKET
6329 037202 010465 177776          MOV   R4,T5DB(R5) ;ISSUE COMMAND
6330 037206 004737 017104          JSR   PC,WAITF   ;WAIT FOR SSR TO SET
6331 037212 016501 000000          MOV   T5SR(R5),R1 ;GET T5SR CONTENTS
6332 037216 012702 104210          MOV   #SC!NXM!5SR!BIT3,R2 ;SET UP EXPECTED
6333 037222 020102              CMP   R1,R2     ;ARE THEY EQUAL
6334 037224 001417              BEQ   80$        ;BR, IF OK ESP. FUNCTION REJECT
6335 037226 062737 000001    050216 ADD   #1,T24DLY ;LOOK AT NEXT EXTENDED BITS WORTH OF MEM
6336 037234 022737 000004    050216 CMP   #4,T24DLY ;TOO MUCH MEMORY YET
6337 037242 001402              BEQ   168$       ;BR, IF OVER 18 BIT ADDRESS
6338 037244 000137 037126          JMP   25$        ;TRY AGAIN (NEXT BUNCH OF MEMORY)
6339 037250 004737 020064          168$: JSR   PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6343 037254                 ERRHRD  ERRNO,T23TM,PKT5SR ;T5SR INCORRECT AFTER WRITE COMMAND
6343 037254 104456                 TRAP   C5ERHRD
6343 037256 000642                 .WORD  418
6343 037260 040226                 .WORD  T23TM
6343 037262 011670                 .WORD  PKT5SR
6344 037264          80$:  CKLOOP           ;LOOP IF SELECTED
6344 037264 104406                 TRAP   C5CLP1
6345 037266          90$:
6346 037266          ENDSUB           ;>>>>>>>>>> END SUBTEST >>>>>>>>>
6346 037266          L10054:
6347 037270 023727 002170    000031 CMP   FATFLG,#25. ;IS ERROR COUNT AT 25
6348 037276 002402          999$: BLT   999$       ;BR, IF LESS THAN 25
6349 037300 004737 020136          JSR   PC,CKDROP ;TRY TO DROP THE UNIT
6350 037304

```


CZTUXAO TUBO FRONT END PRT B
TEST 4: BASIC WRITE DATA

MACRO M1200 29-MAR-83 13:32 PAGE 96-1

```

037406 011656 .WORD SFMSG
6406
6407 :*****
6408 :
6409 :WRITE DATA, ACK,CVL=1
6410 :
6411 :*****
6412
6413 037410 012737 000000 040004 23$: MOV #0,T23WB+2 ;HIGH ORDER ADDRESS BITS ETC.
6414 037416 012737 140005 040000 2'S: MOV #140005,T23PK3 ;WRITE DATA, ACK,CVC=1
6415 037424 013701 003076 MOV FREEH1,R1 ;HIGHEST ADDRESS
6416 037430 162701 000100 SUB #100,R1 ;SET ADDRESS A LITTLE LOWER
6417 037434 010137 040002 MOV R1,T23WB ;LOAD INTO THE PACKET
6418 037440 012737 175000 040006 MOV #64000.,T23SZ ;SET UP BUFFER SIZE (64K BYTES)
6419 037446 012704 040000 MOV #T23PK3,R4 ;R4 = POINTER TO PACKET
6420 037452 010465 177776 MOV R4,TSDB(R5) ;ISSUE COMMAND
6421 037456 004737 017104 JSR PC,WAITF ;WAIT FOR SSR TO SET
6422 037462 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
6423 037466 012702 104210 MOV #SC!NXM!SSR!BIT3,R2 ;SET UP EXPECTED
6424 037472 020102 CMP R1,R2 ;ARE THEY EQUAL
6425 037474 001415 BEQ 80$ ;BR, IF OK ESP. FUNCTION REJECT
6426 037476 062737 000001 040004 ADD #1,T23WB+2 ;START CUTTING THE HIGH ADDRESS BITS DOWN
6427 037504 022737 000004 040004 CMP #4,T23WB+2
6428 037512 001341 BNE 24$ ;IF NOT AT ZERO, KEEP TRYING
6429 037514 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6433 037520 ERRHRD ERRNO,T23TMA,PKTSSR ;TSSR INCORRECT AFTER WRITE COMMAND
        037520 104456 TRAP CSERHRD
        037522 000645 .WORD 421
        037524 040315 .WORD T23TMA
        037526 011670 .WORD PKTSSR
6434 037530 80$: CKLOOP ;LOOP IF SELECTED TRAP C$CLP1
        037530 104406
6435 037532 004737 041360 JSR PC,T23RT2 ;CLEAN UP PACKET
6436 037536 004737 041422 JSR PC,T23RT3 ;RESTORE PACKET
6437
6438 :*****
6439 :
6440 : ISSUE A CALL TO REWIND ROUTINE WHICH WILL WAIT QUITE SOME TIME FOR SSR
6441 :
6442 :*****
6443 037542 004737 010424 JSR PC,REWIND ;CALL THE TAPE REWIND
6444
6445
6446
6447 037546 012737 102010 037760 MOV #102010,T23PK2 ;REWIND (POSITION) COMMAND
6448 037554 012704 037760 MOV #T23PK2,R4 ;LOAD R4 WITH PACKET ADDRESS
6449 037560 010465 177776 MOV R4,TSDB(R5) ;ISSUE REWIND COMMAND
6450 037564 004737 017220 JSR PC,CHKTSSR ;WAIT FOR SSR TO SET
6451 037570 103407 BCS 85$ ;BR, IF TSSR IS OK (GOOD)
6452 037572 010001 MOV R0,R1 ;SAVE TSSR CONTENTS
6453 037574 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6457 037600 ERRHRD ERRNO,T23RWV,PKTSSR ;TSSR IS INCORRECT AFTER REWIND
        037600 104456 TRAP CSERHRD
        037602 000646 .WORD 422
        037604 040401 .WORD T23RWV
        037606 011670 .WORD PKTSSR
6458 037610 85$:

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 96-2
TEST 4: BASIC WRITE DATA

```

6459 037610          SKIP:
6460 037610          1308:
6461 037610          ENDSUB          :>>>>>>>>>> END SUBTEST >>>>>>>>>>
      037610          L10055:
      037610 104403          TRAP      CSESUB
6462 037612 023727 002170 000031    CMP      FATFLG,#25.      ;IS ERROR COUNT AT 25
6463 037620 002402          BLT      9998             ;BR, IF LESS THAN 25
6464 037622 004737 020136          JSR      PC,CKDROP       ;TRY TO DROP THE UNIT
6465 037626          9998:
6466 037626 004737 017340          JSR      PC,TSTLOOP     ;DO WE NEED TO ITERATE TEST
6467 037632 103002          BCC      1638             ;BR, IF NO LOOP REQUIRED
6468 037634 000137 035566          JMP      T23LOOP       ;EXECUTE AGAIN
6469 037640          1638:
6470 037640          EXIT      TST          ;ALL DONE THIS TEST
      037640 104432          TRAP      CSEXIT
      037642 001604          .WORD    L10050-.
```


CZTUXAO TUBO FRONT END PRT B
TEST : BASIC WRITE DATA

MACRO M1200 29-MAR-83 13:32 PAGE 97

6472
6473
6474
6476 037644
6478 037650
6479 037650 100004
6480 037652 037660
6481 037654 000000
6482 037656 000010
6483 037660
6484 037660 037670
6485 037662 000000
6486 037664 000012
6487 037666 000000
6488 037670
6489
6490
6491
6493 037752
6495 037760
6496 037760 100006
6497 037762 040012
6498 037764 000000
6499 037766 000006
6500
6502 037770
6504 040000
6505 040000 100005
6506 040002 000000
6507 040004 000000
6508 040006 000000
6509
6510
6511 040010 000000
6512
6513
6514 040012
6515 040012 010
6516 040013 200
6517 040014 000000
6518 040016 000000
6519
6520
6521
6522
6523
6524 040020 100005
6525 040022 100405
6526 040024 102005
6527 040026 177777
6528
6529

```

:
: LOCAL STORAGE FOR THIS TEST
:
:
T23PACKET: .BLKB 10-<.-TUV2AB7>
           .WORD 100004
           .WORD T23DATA
           .WORD 0
           .WORD 8.
T23DATA:  .WORD T23BFR
           .WORD 0
           .WORD 10.
           .WORD 0
T23BFR:   .BLKW 25.
:
: WRITE SUBSYSTEM MEMORY COMMAND PACKET
:
T23PK2:   .BLKB 10-<.-TUV2AB7>
           .WORD 100006
           .WORD T23BF2
           .WORD 0
           .WORD 6.
T23PK3:   .BLKB 10-<.-TUV2AB7>
           .WORD 100005
T23WB:    .WORD 0
           .WORD 0
T23SZ:    .WORD 0
           .EVEN
T23RSZ:   .WORD 0
:
T23BF2:
T23BS0:   .BYTE 10
T23BS1:   .BYTE 200
T23S2:    .WORD 0
T23S3:    .WORD 0
:
           .EVEN
: TAPE MOTION PACKET COMMAND VALUES
T23WD:    .WORD 100005
T23WDR:   .WORD 100405
T23CON:   .WORD 102005
           .WORD 177777

```

```

: 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

: MESSAGE BUFFER

: WRITE SUB SYS MEM COMMAND, AND ACK
: ADDRESS OF SELECT BLOCK DATA

: SIZE OF DATA PACKET

: WRITE COMMAND, AND ACK
: ADDRESS OF WRITE BUFFER

: SIZE OF BUFFER (EXTENT)

: LARGEST TAPE RECORD IN BYTES

: BSEL0 AREA
: BSEL1 AREA
: SEL 2 AREA
: DATA AREA

: WRITE DATA (NEXT)
: WRITE DATA RETRY
: WRITE CONTINUOUS
: END OF DATA

```

6531
 6532
 6533
 6534
 6535
 6536 040030
 6537 040063
 6538 040150
 6539 040226
 6540 040315
 6541 040401
 6542 040450
 6543 040523
 6544 040571
 6545 040644
 6546 040733
 6547 041035
 6548 041110
 6549 041162
 6550 041251
 6551
 6552
 6553
 6554
 6555
 6556
 6557
 6558
 6559 041266
 6560 041266
 6561 041272
 6562 041276
 6563 041302
 6564 041306
 6565 041310
 6566 041314
 6567 041320
 6568 041322
 6569 041326
 6570 041330
 6571 041334
 6572 041340
 6573 041346
 6574 041350
 6575 041354
 6576 041356
 6577
 6578
 6579 041360
 6580 041360
 6581 041364
 6582 041370
 6583 041374
 6584 041400
 6585 041402
 6586 041406
 6587 041412

127 122 111
 105 117 124
 127 122 111
 124 123 123
 124 123 123
 122 145 167
 122 101 115
 124 123 123
 104 162 151
 124 123 123
 124 123 123
 103 126 103
 124 123 102
 127 122 111
 102 141 163
 012701 037650
 012721 100004
 012721 037660
 005021
 012721 000012
 012721 037670
 005021
 012721 000024
 005021
 012711 000000
 012702 000030
 012762 177777 037670 64\$
 005742
 020227 000000
 001371
 000207
 012701 037760
 012721 100006
 012721 040012
 005021
 012721 000006
 012701 040012
 005021

:+
 :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 Due To NXM'
 T23TMA: .ASCIZ 'TSSR Not Correct After WRITE To Non-Existent Memory'
 T23RUN: .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)'
 T23ID: .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
 MOV #177777,T23BFR(R2) ;ALL ONES TO MESSAGE BUFFER
 TST -(R2) ;BUMP DOWN TO NEXT LOCATION
 CMP R2,#0 ;R2 AT ZERO YET
 BNE 64\$;KEEP GOING UNTIL DONE
 RTS PC ;RETURN

T23RT2:
 SAVREG ;SAVE THE REGISTERS
 MOV #T23PK2,R1 ;START OF THE PACKET
 MOV #100006,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK
 MOV #T23BF2,(R1)+ ;ADDRESS OF DATA BLOCK
 CLR (R1)+ ;EXTENDED ADDRESS
 MOV #6,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
 MOV #T23BF2,R1 ;POINT TO DATA SEL AREA
 CLR (R1)+

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 98-1
TEST 4: BASIC WRITE DATA

6588	041414	005021	
6589	041416	005011	
6590	041420	000207	
6591	041422		
6592	041422		
6593	041426	012701	040000
6594	041432	012721	100005
6595	041436	005021	
6596	041440	005021	
6597	041442	005011	
6598	041444	000207	
6599	041446		
	041443		
	041446	104401	

T23RT3:

```

CLR (R1)+
CLR (R1)
RTS PC
SAVREG
MOV #T23PK3,R1
MOV #100005,(R1)+
CLR (R1)+
CLR (R1)+
CLR (R1)
RTS PC
ENDTST

```

:RETURN

```

:SAVE THE REGISTERS
:START OF THE PACKET
:WRITE TAPE. WITH ACK
:ADDRESS OF DATA BLOCK
:EXTENDED ADDRESS
:SIZE OF DATA BLOCK
:RETURN

```

L10050:

TRAP CSETST

CZTUXAO TU80 FRONT END PRT B MACRC M1200 29-MAR-83 13:32 PAGE 100
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

6602
6603
6604
6605
6606
6607
6608
6609
6610
6611
6612
6613
6614
6615
6616
6617
6618
6619
6620
6621
6622
6623
6624
6625
6626
6627
6632
6633
6634
6635
6636
6637

.SBTTL TEST 5: 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 12 SUBTESTS

:-
BGNTST

CLR FATFLG ; CLEAR FATAL ERROR FLAG
CLR KTFLG ; HOLD OFF KT11
MOV #EPRT1,EPRTSW ; SET UP PRIMARY ERROR MESSAGE
CLR KTENABLE ; TURN OFF KT11
JSR PC,KTOFF ; TURN KT11 OFF
MOV #TST24ID,RO ; ASCII MESSAGE TO IDENTIFY TEST
JSR PC,TSTSETUP ; DO INITIAL TEST SETUP
MOV #1,LOOPCNT ; PERFORM 1 ITERATIONS

:+
T24LOOP:
:

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 101-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

6695 041612          ERRHRD  ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTIC FAILED
      041612 104456          TRAP          CSERHRD
      041614 000766          .WORD          502
      041616 004754          .WORD          WRTMSG
      041620 011656          .WORD          SFIMSG
6696 041622          24$:  CKLOOP                      ;LOOP IF SELECTED
      041622 104406          TRAP          CSCLP1
6697
6698
6699
6700
6701
6702
6703
6704 041624 004737 010424      JSR      PC,REWIND          ;CALL TAPE REWIND COMMAND
6705 041630 103407          BCS      30$              ;BR, IF NO PROBLEM
6706 041632 010001          MOV      R0,R1             ;SAVE TSSR
6707 041634 004737 020064      JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
6711 041640          ERRHRD  ERRNO,T24RWN,PKTSSR      ;REWIND NOT ACCEPTED
      041640 104456          TRAP          CSERHRD
      041642 000767          .WORD          503
      041644 051326          .WORD          T24RWN
      041646 011670          .WORD          PKTSSR
6712 041650          30$:  CKLOOP                      ;LOOP IF SELECTED
      041650 104406          TRAP          CSCLP1
6713
6714
6715
6716
6717
6718
6719
6720 041652 013701 050066      MOV      T24BFR+6,R1      ;PICK UP XSTO
6721 041656 010102          MOV      R1,R2             ;SET UP EXPECTED
6722 041660 052702 000002      BIS      #BIT1,R2         ;SET BOT BIT IN EXPECTED
6723 041664 020102          CMP      R1,R2             ;DOES EXP = REC'D
6724 041666 001406          BEQ      40$              ;BR, IF EQUAL (OK)
6725 041670 004737 020064      JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
6729 041674          ERRHRD  ERRNO,T24BOT,EXPREC      ;TAPE NOT AT BOT AFTER REWIND
      041674 104456          TRAP          CSERHRD
      041676 000770          .WORD          504
      041700 051043          .WORD          T24BOT
      041702 016330          .WORD          EXPREC
6730 041704          40$:  CKLOOP                      ;LOOP IF SELECTED
      041704 104406          TRAP          CSCLP1
6731 041706 012703 000400      MOV      #256.,R3         ;RECORD SIZE
6732 041712 013737 003072 050172  MOV      FREE,T24RB       ;STARTING WRITE BUFFER ADDRESS
6733
6734
6735
6736
6737
6738
6739
6740 041720 012737 140005 050170  MOV      #140005,T24PK3   ;WRITE DATA,CVC=1,ACK COMMAND
6741 041726 012704 050170      MOV      #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6742 041732          65$:

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 101-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

6743 041732 010300      MOV      R3,R0      ;SET PATTERN IN CORRECT REGISTER
6744 041734 004737 020356 JSR      PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
6745 041740 010337 050176 MOV      R3,T24SZ   ;SET UP RECORD SIZE IN PACKET
6746 041744 010465 177776 MOV      R4,TSSDB(R5) ;ISSUE COMMAND
6747 041750 004737 017104 JSR      PC,WAITF   ;WAIT FOR SSR TO SET
6748 041754 016501 000000 MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
6749 041760 012702 000200 MOV      #SSR,R2    ;SET UP EXPECTED
6750 041764 020102      CMP      R1,R2     ;ARE THEY EQUAL
6751 041766 001406      BEQ      75$      ;BR, IF OK
6752 041770 004737 020064 JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6756      ;SOFT ERROR, REALLY CHECKING THE
6757      ;READ COMMAND
6758 041774      ERRSFT ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
        041774 104457      TRAP      CSERSOFT
        041776 000771      .WORD    505
        042000 005011      .WORD    WPTERR
        042002 011670      .WORD    PKTSSR
6759 042004      75$:   CKLOOP      ;LOOP IF SELECTED
        042004 104406      TRAP      CSCLP1
6760 042006 005723      TST      (R3)+     ;BUMP RECORD SIZE
6761 042010 022703 000414  CMP      #268.,R3 ;END OF RECORD YET
6762 042014 001346      BNE      65$      ;BR, IF MORE RECORDS TO WRITE
6763 042016      80$:   CKLOOP      ;LOOP IF SELECTED
        042016 104406      TRAP      CSCLP1
6764 042020      120$:
6765      ;*****
6766      ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6767      ;*****
6768      ;*****
6769      ;*****
6770      ;*****
6771      ;*****
6772 042020 012703 000012  MOV      #10.,R3   ;SPECIAL MULTIPLE REWIND
6773 042024 004737 010424  JSR      PC,REWIND ;ISSUE REWIND COMMAND
6774 042030 103407      BCS      130$     ;BR, IF NO PROBLEM
6775 042032 010001      MOV      R0,R1    ;SAVE TSSR
6776 042034 004737 020064  JSR      PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
6780 042040      ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
        042040 104456      TRAP      CSERHRD
        042042 000772      .WORD    506
        042044 051326      .WORD    T24RWN
        042046 011670      .WORD    PKTSSR
6781 042050      130$:   CKLOOP      ;LOOP IF SELECTED
        042050 104406      TRAP      CSCLP1
6782      ;*****
6783      ;*****
6784      ;READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
6785      ;*****
6786      ;*****
6787      ;*****
6788      ;*****
6789 042052 013701 050066  MOV      T24BFR+6,R1 ;PICK UP XST0
6790 042056 010102      MOV      R1,R2    ;SET UP EXPECTED
6791 042060 052702 000002  BIS      #BIT1,R2 ;SET BOT BIT IN EXPECTED
6792 042064 020102      CMP      R1,R2   ;DOES EXP = REC'D
6793 042066 001407      BEQ      140$    ;BR, IF EQUAL (OK)
6794 042070 077323      SOB      R3,125$ ;DO ANOTHER REWIND BEFORE REPORTING ERROR

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 101-3
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

6795 042072 004737 020064      JSR    PC,FATCHK      ;INC AND CHECK FOR MOPE THAN 25 ERRORS
6799 042076      ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
      042076 104456      TRAP   C$ERHRD
      042100 000773      .WORD  507
      042102 051043      .WORD  T24BOT
      042104 016330      .WORD  EXPREC
6800 042106      140$:  CKLOOP      ;LOOP IF SELECTED      TRAP   C$CLP1
      042106 104406      ;RECORD SIZE
6801 042110 012703 000400      MOV    #256.,R3      ;STARTING READ BUFFER ADDRESS
6802 042114 013737 003072 050172      MOV
6803
6804      ;*****
6805      ;
6806      ;READ DATA,CVC=1,ACK COMMAND
6807      ;
6808      ;*****
6809 042122 012737 140001 050170      MOV    #140001,T24PK3 ;READ DATA,CVC=1,ACK COMMAND
6810 042130 012704 050170 165$:  MOV    #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
6811 042134 010337 050176      MOV    R3,T24SZ      ;SET UP RECORD SIZE IN PACKET
6812 042140 010465 177776      MOV    R4,TSDB(R5)    ;ISSUE COMMAND
6813 042144 004737 017104      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
6814 042150 016501 000000      MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
6815 042154 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
6816 042160 020102      CMP    R1,R2        ;ARE THEY EQUAL
6817 042162 001406      BEQ    170$         ;BR, IF OK
6818 042164 004737 020064      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
6822 042170      ERRHRD  ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      042170 104456      TRAP   C$ERHRD
      042172 000774      .WORD  508
      042174 005104      .WORD  RDERR
      042176 011670      .WORD  PKTSSR
6823 042200      170$:  CKLOOP      ;LOOP IF SELECTED      TRAP   C$CLP1
      042200 104406      ;GET BUFFER ADDRESS
6824 042202 013702 003072      MOV    FREE,R2      ;CURRENT RECORD SIZE
6825 042206 010304      MOV    R3,R4        ;FIRST LOCATION IN BUFFER
6826 042210 162704 000400      SUB    #256.,R4     ;GET LOCATION IN BUFFER (ADDRESS)
6827 042214 060204 173$:  ADD    R2,R4        ;CHECK DATA READ (R3=DATA ALSO)
6828 042216 021403      CMP    (R4),R3     ;BR, IF ALL IS WELL
6829 042220 001410      BEQ    180$         ;RECD DATA
6830 042222 011401      MOV    (R4),R1     ;EXPECTED DATA
6831 042224 010302      MOV    R3,R2        ;INC AND CHECK FOR MORE THAN 25 ERRORS
6832 042226 004737 020064      JSR    PC,FATCHK      ;DATA READ NOT = WRITTEN
6836 042232      ERRHRD  ERRNO,T24DTA,EXPREC ;
      042232 104456      TRAP   C$ERHRD
      042234 000775      .WORD  509
      042236 051110      .WORD  T24DTA
      042240 016330      .WORD  EXPREC
6837 042242      180$:  CKLOOP      ;LOOP IF SELECTED      TRAP   C$CLP1
      042242 104406      ;BUMP TO NEXT LOCATION
6838 042244 005724      TST    (R4)+       ;GET BACK TO CORRECT SIZE
6839 042246 100204      SUB    R2,R4        ;END OF RECORD YET
6840 042250 020403      CMP    R4,R3        ;BR, IF NOT AT END OF RECORD
6841 042252 001360      BNE    173$        ;BUMP RECORD SIZE
6842 042254 005723      TST    (R3)+       ;END OF RECORD YET
6843 042256 022703 000412      CMP    #266.,R3     ;BR, IF MORE RECORDS TO WRITE
6844 042262 001322      BNE    165$        ;LOOP IF SELECTED
6845 042264      190$:  CKLOOP
    
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 102-1
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

6904          :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6905          :
6906          :*****
6907          :
6908 042400 004737 010424          JSR    PC,REWIND          :CALL TAPE REWIND COMMAND
6909 042404 103407          BCS    30,                :BR, IF NO PROBLEM
6910 042406 010001          MOV    R0,R1              :SAVE TSSR
6911 042410 004737 020064          JSR    PC,FATCHK          :INC AND CHECK FOR MORE THAN 25 EPRORS
6915 042414          ERRHRD  ERRNO,T24RWN,PKTSSR :REWIND NOT ACCEPTED
        042414 104456          TRAP   CSERHRD
        042416 001000          .WORD  512
        042420 051326          .WORD  T24RWN
        042422 011670          .WORD  PKTSSR
6916 042424          30$:   CKLOOP          :LOOP IF SELECTED          TRAP   CSCLP1
        042424 104406
6917          :
6918          :*****
6919          :
6920          :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6921          :
6922          :*****
6923          :
6924 042426 013701 050066          MOV    T24BFR+6,R1       :PICK UP XSTO
6925 042432 010102          MOV    R1,R2              :SET UP EXPECTED
6926 042434 052702 000002          BIS    #BIT1,R2          :SET BOT BIT IN EXPECTED
6927 042440 020102          CMP    R1,R2              :DOES EXP = REC'D
6928 042442 001406          BEQ    40$,              :BK, IF EQUAL (OK)
6929 042444 004737 020064          JSR    PC,FATCHK          :INC AND CHECK FOR MORE THAN 25 ERRORS
6933 042450          ERRHRD  ERRNO,T24BOT,EXPREC :TAPE NOT AT BOT AFTER REWIND
        042450 104456          TRAP   CSERHRD
        042452 001001          .WORD  513
        042454 051043          .WORD  T24BOT
        042456 016330          .WORD  EXPREC
6934 042460          40$:   CKLOOP          :LOOP IF SELECTED          TRAP   CSCLP1
        042460 104406
6935 042462 012703 000400          MOV    #256.,R3           :RECORD SIZE
6936 042466 013737 003072 050172  MOV    FREE,T24RB         :STARTING WRITE BUFFER ADDRESS
6937          :
6938          :*****
6939          :
6940          :WRITE DATA,ACK,SWB,CVC=1 COMMAND
6941          :
6942          :*****
6943          :
6944 042474 012737 150005 050170          MOV    #150005,T24PK3     :WRITE DATA,ACK,SWB,CVC=1 COMMAND
6945 042502 012704 050170          MOV    #T24PK3,R4        :SET UP R4 WITH PACKET ADDRESS
6946 042506          65$:
6947 042506 010300          MOV    R3,R0              :SET PATTERN IN CORRECT REGISTER
6948 042510 004737 020356          JSR    PC,FILLMEM         :FILL MEMORY WITH RECORD SIZE
6949 042514 010337 050176          MOV    R3,T24SZ          :SET UP RECORD SIZE IN PACKET
6950 042520 010465 177776          MOV    R4,TSDB(R5)       :ISSUE COMMAND
6951 042524 004737 017104          JSR    PC,WAITF          :WAIT FOR SSR TO SET
6952 042530 016501 000000          MOV    TSSR(R5),R1       :GET TSSR CONTENTS
6953 042534 012702 000200          MOV    #SSR,R2           :SET UP EXPECTED
6954 042540 020102          CMP    R1,R2              :ARE THEY EQUAL
6955 042542 001406          BEQ    75$,              :BR, IF OK
6956 042544 004737 020064          JSR    PC,FATCHK          :INC AND CHECK FOR MORE THAN 25 ERRORS
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 102-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

6960                                     :SOFT ERROR, REALLY CHECKING THE
6961                                     :READ COMMAND
6962 042550 ERRSOF ERRNO,WRERR,PKTSSR    :TSSR INCORRECT AFTER WRITE DATA
      042550 104457                      TRAP CSERSOFT
      042552 001002                      .WORD 514
      042554 005011                      .WORD WRERR
      042556 011670                      .WORD PKTSSR
6963 042560 758: CKLOOP                  :LOOP IF SELECTED                      TRAP CSCLP1
      042560 104406
6964 042562 005723                      :BUMP RECORD SIZE
6965 042564 022703 000414              CMP #268.,R3                          :END OF RECORD YET
6966 042570 001346                      BNE 658                               :BR, IF MORE RECORDS TO WRITE
6967 042572 808: CKLOOP                  :LOOP IF SELECTED                      TRAP CSCLP1
      042572 104406
6968 042574 1208:
6969                                     :*****
6970                                     :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
6971                                     :*****
6972
6973
6974
6975
6976 042574 004737 010424              JSR PC,REWIND                        :CALL TAPE REWIND COMMAND
6977 042600 103407                      BCS 1308                             :BR, IF NO PROBLEM
6978 042602 010001                      MOV R0,R1                            :SAVE TSSR
6979 042604 004737 020064              JSR PC,FATCHK                        :INC AND CHECK FOR MORE THAN 25 ERRORS
6983 042610 ERRHRD ERRNO,T24RWN,EXPREC :REWIND NOT ACCEPTED
      042610 104456                      TRAP CSERHRD
      042612 001003                      .WORD 515
      042614 051326                      .WORD T24RWN
      042616 016330                      .WORD EXPREC
6984 042620 1308: CKLOOP                  :LOOP IF SELECTED                      TRAP CSCLP1
      042620 104406
6985
6986                                     :*****
6987                                     :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
6988                                     :*****
6989
6990
6991
6992 042622 013701 050066              MOV T24BFR+6,R1                      :PICK UP XSTO
6993 042626 010102                      MOV R1,R2                            :SET UP EXPECTED
6994 042630 052702 000002              BIS #BIT1,R2                         :SET BOT BIT IN EXPECTED
6995 042634 020102                      CMP R1,R2                            :DOES EXP = REC'D
6996 042636 001406                      BEQ 1408                             :BR, IF EQUAL (OK)
6997 042640 004737 020064              JSR PC,FATCHK                        :INC AND CHECK FOR MORE THAN 25 ERRORS
7001 042644 ERRHRD ERRNO,T24BOT,EXPREC :TAPE NOT AT BOT AFTER REWIND
      042644 104456                      TRAP CSERHRD
      042646 001004                      .WORD 516
      042650 051043                      .WORD T24BOT
      042652 016330                      .WORD EXPREC
7002 042654 1408: CKLOOP                  :LOOP IF SELECTED                      TRAP CSCLP1
      042654 104406
7003 042656 012703 000400              MOV #256.,R3                          :RECORD SIZE
7004 042662 013737 003072 050172      MOV FREE,T24RB                       :STARTING READ BUFFER ADDRESS
7005
7006                                     :*****

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 102-3
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```
7007
7008 ;READ DATA,IE,ACK,SWB COMMAND
7009 ;
7010 ;.....
7011
7012 042670 012737 110001 050170      MOV #110001,T24PK3      ;READ DATA,IE,ACK,SWB COMMAND
7013 042676 012704 050170      165$: MOV #T24PK3,R4      ;SET UP R4 WITH PACKET ADDRESS
7014 042702 010337 050176      MOV R3,T24SZ           ;SET UP RECORD SIZE IN PACKET
7015 042706 010465 177776      MOV R4,TSDDB(R5)       ;ISSUE COMMAND
7016 042712 004737 017104      JSR PC,WAITF           ;WAIT FOR SSR TO SET
7017 042716 016501 000000      MOV TSSR(R5),R1       ;GET TSSR CONTENTS
7018 042722 012702 000200      MOV #SSR,R2           ;SET UP EXPECTED
7019 042726 020102              CMP R1,R2              ;ARE THEY EQUAL
7020 042730 001406              BEQ 170$               ;BR, IF OK
7021 042732 004737 020064      JSR PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
7025 042736              ERRHRD ERRNO,RDERR,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                              TRAP CSERHRD
                              .WORD 517
                              .WORD RDERR
                              .WORD PKTSSR
7026 042746              170$: CKLOOP          ;LOOP IF SELECTED
                              TRAP CSCLP1
7027 042750 013702 003072      MOV FREE,R2           ;GET BUFFER ADDRESS
7028 042754 010304              MOV R3,R4              ;CURRENT RECORD SIZE
7029 042756 162704 000400      SUB #256.,R4          ;FIRST LOCATION IN BUFFER
7030 042762 060204      173$: ADD R2,R4              ;GET LOCATION IN BUFFER (ADDRESS)
7031 042764 021403      CMP (R4),R3           ;CHECK DATA READ (R3=DATA ALSO)
7032 042766 001410      BEQ 180$              ;BR, IF ALL IS WELL
7033 042770 011401      MOV (R4),R1           ;RECD DATA
7034 042772 010302      MOV R3,R2              ;EXPECTED DATA
7035 042774 004737 020064      JSR PC,FATCHK         ;INC AND CHECK FOR MORE THAN 25 ERRORS
7039 043000              ERRHRD ERRNO,T24DTA,EXPREC ;DATA READ NOT = WRITTEN
                              TRAP CSERHRD
                              .WORD 518
                              .WORD T24DTA
                              .WORD EXPREC
7040 043010              180$: CKLOOP          ;LOOP IF SELECTED
                              TRAP CSCLP1
7041 043012 005724              TST (R4)+              ;BUMP TO NEXT LOCATION
7042 043014 160204              SUB R2,R4              ;SET SIZE TO CORRECT VALUE
7043 043016 020403      CMP R4,R3              ;END OF RECORD YET
7044 043020 001360              BNE 173$              ;BR, IF NOT AT END OF RECORD
7045 043022 005723              TST (R3)+              ;BUMP RECORD SIZE
7046 043024 022703 000412      CMP #266.,R3          ;END OF RECORD YET
7047 043030 001322              BNE 165$              ;BR, IF MORE RECORDS TO WRITE
7048 043032              190$: CKLOOP          ;LOOP IF SELECTED
7049 043034              ENDSUB               ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>
                              TRAP CSCLP1
                              L10060:
                              TRAP CSesub
7050 043036 023727 002170 0^0031      CMP FATFLG,#25.       ;IS ERROR COUNT AT 25
7051 043044 002402              BLT 999$               ;BR, IF LESS THAN 25
7052 043046 004737 020136      JSR PC,CKDROP         ;TRY TO DROP THE UNIT
7053 043052      999$:
```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 103-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7107      :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7108      :
7109      :*****
7110      :
7111 043146 004737 010424      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7112 043152 103407      BCS      30$           ;BR, IF NO PROBLEM
7113 043154 010001      MOV      R0,R1        ;SAVE TSSR
7114 043156 004737 020064      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7118 043162      ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
          043162 104456      TRAP      CSERHRD
          043164 001011      .WORD    521
          043166 051326      .WORD    T24RWN
          043170 011670      .WORD    PKTSSR
7119 043172      30$:      CKLOOP           ;LOOP IF SELECTED
          043172 104406      TRAP      CSCLP1
7120      :
7121      :*****
7122      :
7123      :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)
7124      :
7125      :*****
7126      :
7127 043174 013701 050066      MOV      T24BFR+6,R1  ;PICK UP XSTO
7128 043200 010102      MOV      R1,R2        ;SET UP EXPECTED
7129 043202 052702 000002      BIS      #BIT1,R2    ;SET BOT BIT IN EXPECTED
7130 043206 020102      CMP      R1,R2        ;DOES EXP = REC'D
7131 043210 001406      BEQ      40$         ;BK, IF EQUAL (OK)
7132 043212 004737 020064      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7136 043216      ERRHRD  ERRNO,T24BOT,EXPREC ;TAPE NOT AT BOT AFTER REWIND
          043216 104456      TRAP      CSERHRD
          043220 001012      .WORD    522
          043222 051043      .WORD    T24BOT
          043224 016330      .WORD    EXPREC
7137 043226      40$:      CKLOOP           ;LOOP IF SELECTED
          043226 104406      TRAP      CSCLP1
7138 043230 012703 001000      MOV      #512.,R3    ;RECORD SIZE
7139 043234 013737 003072 050172  MOV      FREE,T24RB  ;STARTING WRITE BUFFER ADDRESS
7140      :
7141      :*****
7142      :
7143      :WRITE DATA,ACK,CVC=1 COMMAND
7144      :
7145      :*****
7146      :
7147 043242 012737 140005 050170  MOV      #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
7148 043250 012704 050170      MOV      #T24PK3,R4  ;SET UP R4 WITH PACKET ADDRESS
7149 043254      65$:
7150 043254 010337 050176      MOV      R3,T24SZ    ;SET UP RECORD SIZE IN PACKET
7151 043260 010465 177776      MOV      R4,TSDB(R5) ;ISSUE COMMAND
7152 043264 004737 017104      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
7153 043270 016501 000000      MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
7154 043274 012702 000200      MOV      #SSR,R2    ;SET UP EXPECTED
7155 043300 020102      CMP      R1,R2        ;ARE THEY EQUAL
7156 043302 001406      BEQ      75$         ;BR, IF OK
7157 043304 004737 020064      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7161      :SOFT ERROR, REALLY CHECKING THE
7162      :READ DATA COMMAND

```

CZTUXAJ TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 103-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7163 043310          ERRSOFT ERRNO,WRERR,PKTSSR      :TSSR INCORRECT AFTER WRITE DATA
      043310 104457          TRAP          CSERSOFT
      043312 001013          .WORD          523
      043314 005011          .WORD          WRERR
      043316 011670          .WORD          PKTSSR
7164 043320          758:   CKLOOP                  :LOOP IF SELECTED          TRAP          CSCLP1
      043320 104406
7165 043322          1208:
7166
7167          :*****
7168          :
7169          :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7170          :
7171          :*****
7172
7173 043322 004737 010424      JSR      PC,REWIND          :CALL TAPE REWIND COMMAND
7174 043326 103407          BCS      1308              :BR, IF NO PROBLEM
7175 043330 010001          MOV      R0,R1             :SAVE TSSR
7176 043332 004737 020064      JSR      PC,FATCHK        :INC AND CHECK FOR MORE THAN 25 ERRORS
7180 043336          ERRHRD  ERRNO,T24RWN,PKTSSR      :REWIND NOT ACCEPTED
      043336 104456          TRAP          CSERHRD
      043340 001014          .WORD          524
      043342 051326          .WORD          T24RWN
      043344 011670          .WORD          PKTSSR
7181 043346          1308:   CKLOOP                  :LOOP IF SELECTED          TRAP          CSCLP1
      043346 104406
7182
7183          :*****
7184          :
7185          :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7186          :
7187          :*****
7188
7189 043350 013701 050066      MOV      T24BFR+6,R1      :PICK UP XST0
7190 043354 010102          MOV      R1,R2             :SET UP EXPECTED
7191 043356 052702 000002      BIS      #BIT1,R2        :SET BOT BIT IN EXPECTED
7192 043362 020102          CMP      R1,R2             :DOES EXP = REC'D
7193 043364 001406          BEQ      1408              :BR, IF EQUAL (OK)
7194 043366 004737 020064      JSR      PC,FATCHK        :INC AND CHECK FOR MORE THAN 25 ERRORS
7198 043372          ERRHRD  ERRNO,T24BOT,EXPREC      :TAPE NOT AT BOT AFTER REWIND
      043372 104456          TRAP          CSERHRD
      043374 001015          .WORD          525
      043376 051043          .WORD          T24BOT
      043400 016330          .WORD          EXPREC
7199 043402          1408:   CKLOOP                  :LOOP IF SELECTED          TRAP          CSCLP1
      043402 104406
7200 043404 012703 000400      MOV      #256.,R3         :RECORD SIZE
7201 043410 013737 003072 050172  MOV      FREE,T24RB       :STARTING READ BUFFER ADDRESS
7202
7203          :*****
7204          :
7205          :READ DATA,ACK,CVC=1 COMMAND
7206          :
7207          :*****
7208
7209 043416 012737 140001 050170  MOV      #140001,T24PK3   :READ DATA,ACK,CVC=1 COMMAND
7210 043424 012704 050170      1658:  MOV      #T24PK3,R4   :SET UP R4 WITH PACKET ADDRESS

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 104
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7248
7249
7250
7251
7252
7253
7254
7255
7256
7257
7258
7259
7260
7261
7262
7263
7264
7265 043546          BGNSUB                  ;>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>>
       043546                      T5.4:
       043546 104402                JSR     PC,T24RT3        ;SET UP OTHER COMMAND PACKET   TRAP   CSBSUB
7266 043550 004737 052664          JSR     PC,T24REST      ;SET COMMAND PACKET
7267 043554 004737 052530          JSR     PC,T24RT2      ;SET UP OTHER COMMAND PACKET
7268 043560 004737 052622
7269
7270
7271
7272
7273
7274
7275
7276 043564 004737 016630          JSR     PC,SOFINIT     ;DO INITIALIZE ON CONTROLLER
7277 043570 103407                BCS     20$            ;BR IF INIT WAS OK
7278 043572 004737 020064          JSR     PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
7282 043576 010001                MOV     R0,R1         ;CONTENTS OF TSSR REGISTER
7283 043600                ERRDF  ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
       043600 104455                TRAP   CSERDF
       043602 001020                .WORD  528
       043604 003550                .WORD  SFIERR
       043606 011656                .WORD  SFIMSG
7284 043610                20$:
7285 043610 012704 050040          MOV     #T24PACKET,R4 ;SUBROUTINE NEEDS PACKET ADDRESS
7286
7287
7288
7289
7290
7291
7292
7293 043614 004737 010322          JSR     PC,WRTCHR     ;ISSUE WRITE CHARACTERISTICS
7294 043620 103407                BCS     24$            ;BR, IF COMMAND ISSUED OK
7295 043622 004737 020064          JSR     PC,FATCHK     ;INC AND CHECK FOR MORE THAN 25 ERRORS
7299 043626 010001                MOV     R0,R1         ;SAVE CONTENTS OF TSSR
7300 043630                ERHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
       043630 104456                TRAP   CSERHRD
       043632 001021                .WORD  529
       043634 004754                .WORD  WRTMSG
       043636 011656                .WORD  SFIMSG

```

:*
:
:TEST 5, SUBTEST 4
:
:VERIFIES THAT A READ FORWARD COMMAND READING A RECORD
:SHORTER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
:STATUS ALERT TERMINATION WITH THE RECORD LENGTH SHORT
:(RLS) BITS SET. IT IS VERIFIED THAT THE RESIDUAL BYTE
:COUNT (RBPCR) IN THE MESSAGE BUFFER CONTAINS THE
:PROPER NONZERO VALUE (E.G. THE DIFFERENCE BETWEEN
:THE ORIGINAL BYTE COUNT AND THE ACTUAL RECORD
:LENGTH).
:
:~
:
:~

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 104-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7301 043640          24$:  CKLOOP                :LOOP IF SELECTED
      043640 104406                                TRAP  CSCLP1
7302
7303      :*****
7304      :
7305      :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7306      :
7307      :*****
7308
7309 043642 004737 010424      JSR    PC,REWIND      :CALL TAPE REWIND COMMAND
7310 043646 103407      BCS    30$           :BR, IF NO PROBLEM
7311 043650 010001      MOV    R0,R1         :SAVE TSSR
7312 043652 004737 020064      JSR    PC,FATCHK    :INC AND CHECK FOR MORE THAN 25 ERRORS
7316 043656      ERRHRD  ERRNO,T24RW,PKTSSR  :REWIND NOT ACCEPTED
      043656 104456                                TRAP  CSERHRD
      043660 001022                                .WORD 530
      043662 051326                                .WORD T24RW
      043664 011670                                .WORD PKTSSR
7317 043666          30$:  CKLOOP                :LOOP IF SELECTED
      043666 104406                                TRAP  CSCLP1
7318 043670 012703 000400      MOV    #256.,R3     :RECORD SIZE
7319 043674 013737 003072 050172  MOV    FREE,T24RB   :STARTING WRITE BUFFER ADDRESS
7320
7321      :*****
7322      :
7323      :WRITE DATA,ACK,CVC=1 COMMAND
7324      :
7325      :*****
7326
7327 043702 012737 140005 050170  MOV    #140005,T24PK3 :WRITE DATA,ACK,CVC=1 COMMAND
7328 043710 012704 050170      MOV    #T24PK3,R4   :SET UP R4 WITH PACKET ADDRESS
7329 043714
7330 043714 010337 050176          65$:  MOV    R3,T24SZ     :SET UP RECORD SIZE IN PACKET
7331 043720 010465 177776      MOV    R4,TSDB(R5)  :ISSUE COMMAND
7332 043724 004737 017104      JSR    PC,WAITF     :WAIT FOR SSR TO SET
7333 043730 016501 000000      MOV    TSSR(R5),R1  :GET TSSR CONTENTS
7334 043734 012702 000200      MOV    #SSR,R2      :SET UP EXPECTED
7335 043740 020102      CMP    R1,R2        :ARE THEY EQUAL
7336 043742 001406      BEQ    75$         :BR, IF OK
7337 043744 004737 020064      JSR    PC,FATCHK    :INC AND CHECK FOR MORE THAN 25 ERRORS
7341      :SOFT ERROR, REALLY CHECKING THE
7342      :READ DATA COMMAND
7343      ERRSOFT ERRNO,WRERR,PKTSSR  :TSSR INCORRECT AFTER WRITE DATA
      043750                                TRAP  CSERSOFT
      043750 104457                                .WORD 531
      043752 001023                                .WORD WRERR
      043754 005011                                .WORD PKTSSR
      043756 011670
7344 043760          75$:  CKLOOP                :LOOP IF SELECTED
      043760 104406                                TRAP  CSCLP1
7345 043762          120$:
7346
7347      :*****
7348      :
7349      :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7350      :
7351      :*****
7352

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 104-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7353 043762 004737 010424      JSR      PC,REWIND      :CALL TAPE REWIND COMMAND
7354 043766 103407                BCS      130$          :BR, IF NO PROBLEM
7355 043770 010001                MOV      R0,R1         :SAVE TSSR
7356 043772 004737 020064      JSR      PC,FATCHK     :INC AND CHECK FOR MORE THAN 25 ERRORS
7360 043776                ERRHRD   ERRNO,T24RWN,PKTSSR :REWIND NOT ACCEPTED
                                TRAP      CSERHRD
                                .WORD     532
                                .WORD     T24RWN
                                .WORD     PKTSSR
7361 044006                130$:   CKLOOP                :LOOP IF SELECTED
                                TRAP      CSCLP1
                                .WORD     533
7362 044010 012703 001000      MOV      #512.,R3      :RECORD SIZE
7363 044014 013737 003072 050172  MOV      FREE,T24RB    :STARTING READ BUFFER ADDRESS
7364
7365      :*****
7366      :READ DATA,ACK,CVC=1 COMMAND
7367      :*****
7368
7369
7370
7371 044022 012737 140001 050170      MOV      #140001,T24PK3 :READ DATA,ACK,CVC=1 COMMAND
7372 044030 012704 050170      165$:   MOV      #T24PK3,R4  :SET UP R4 WITH PACKET ADDRESS
7373 044034 010337 050176      MOV      R3,T24SZ      :SET UP RECORD SIZE IN PACKET
7374 044040 010465 177776      MOV      R4,TSDB(R5)   :ISSUE COMMAND
7375 044044 004737 017104      JSR      PC,WAITF      :WAIT FOR SSR TO SET
7376 044050 016501 000000      MOV      TSSR(R5),R1   :GET TSSR CONTENTS
7377 044054 012702 100204      MOV      #SSR!SC!BIT2,R2 :SET UP EXPECTED
7378 044060 020102                CMP      R1,R2         :ARE THEY EQUAL
7379 044062 001406                BEQ      170$          :BR, IF OK
7380 044064 004737 020064      JSR      PC,FATCHK     :INC AND CHECK FOR MORE THAN 25 ERRORS
7384 044070                ERRHRD   ERRNO,T24TRL,EXPREC :TSSR INCORRECT AFTER READ DATA
                                TRAP      CSERHRD
                                .WORD     533
                                .WORD     T24TRL
                                .WORD     EXPREC
7385 044100                170$:   CKLOOP                :LOOP IF SELECTED
                                TRAP      CSCLP1
                                .WORD     534
7386
7387      :*****
7388      :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
7389      :*****
7390
7391
7392
7393 044102 013701 050066      MOV      T24BFR+6,R1   :GET MESSAGE BUFFER
7394 044106 010102                MOV      R1,R2         :SET UP EXPECTED
7395 044110 052702 040000      BIS      #BIT14,R2     :SET THE RLS BIT IN EXPECTED
7396 044114 020102                CMP      R1,R2         :ARE THEY EQUAL
7397 044116 001406                BEQ      180$          :BR, IF EQUAL (ALL IS WELL)
7398 044120 004737 020064      JSR      PC,FATCHK     :INC AND CHECK FOR MORE THAN 25 ERRORS
7402 044124                ERRHRD   ERRNO,T24LOP,EXPREC :THE RLL BIT WAS NOT SET IN XST0
                                TRAP      CSERHRD
                                .WORD     534
                                .WORD     T24LOP
                                .WORD     EXPREC
7403 044134                180$:
7404 044134 013701 050064      MOV      T24BFR+4,R1   :PICK UP RESIDUAL BYTE COUNTER

```

CZTUXAO TUBO FROM: END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 104-3
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

7405	044140	012702	000400	MOV	#256,R2	;THIS SHOULD BE THE DIFFERENCE	
7406	044144	020102		CMP	R1,R2	;IS THE DIFFERENCE CORRECT	
7407	04414E	001406		BEQ	1908	;BR, IF CORRECT	
7408	044150	004737	020064	JSR	PC,FATCHK	;INC AND CHECK FOR MORE THAN 25 ERRORS	
7412	044154			ERRHRD	ERRNO,T24PBP,EXPREC	;RBPDR NOT CORRECT	
	044154	104456				TRAP CSERHRD	
	044156	001027				.WORD 535	
	044160	052306				.WORD T24PBP	
	044162	016330				.WORD EXPREC	
7413	044164		1908:	CKLOOP		;LOOP IF SELECTED	
	044164	104406				TRAP CSCLP1	
7414	044166			ENDSUB		;>>>>>>>>> END SUBTEST >>>>>>>>>>	
	044166					L10062:	
	044166	104403				TRAP CSesub	
7415	044170	023727	002170	000031	CMP	FATFLG,#25.	;IS ERROR COUNT AT 25
7416	044176	002402			BLT	9998	;BR, IF LESS THAN 25
7417	044200	004737	020136		JSR	PC,CKDROP	;TRY TO DROP THE UNIT
7418	044204		9998:				

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 105-1
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7472 044310 004737 020064      JSR    PC,FATCHK      ;INC AND CHECK FOR MORE THAN 25 ERRORS
7476 044314      ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      044314 104456      TRAP    CSERHRD
      044316 001032      .WORD  538
      044320 051326      .WORD  T24RWN
      044322 011670      .WORD  PKTSSR
7477 044324      30$:   CKLOOP      ;LOOP IF SELECTED      TRAP    CSCLP1
      044324 104406
7478 044326 012703 000400      MOV    #256.,R3      ;RECORD SIZE
7479 044332 013737 003072 050172      MOV    FREE,T24RB    ;STARTING WRITE BUFFER ADDRESS
7480      ;*****
7481      ;WRITE DATA,ACK,CVC=1 COMMAND
7482      ;*****
7483      ;*****
7484      ;*****
7485 044340 012737 140005 050170      MOV    #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
7486 044346 012704 050170      MOV    #T24PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
7487 044352 010300      65$:   MOV    R3,R0      ;SET PATTERN IN CORRECT REGISTER
7488 044354 004737 020356      JSR    PC,FILLMEM    ;FILL MEMORY WITH RECORD SIZE
7489 044360 010337 050176      MOV    R3,T24SZ      ;SET UP RECORD SIZE IN PACKET
7490 044364 010465 177776      MOV    R4,TSDB(R5)   ;ISSUE COMMAND
7491 044370 004737 017104      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
7492 044374 016501 000000      MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
7493 044400 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
7494 044404 020102      CMP    R1,R2         ;ARE THEY EQUAL
7495 044406 001406      BEQ    755          ;BR, IF OK
7496 044410 004737 020064      JSR    PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7500      ;SOFT ERROR, REALLY CHECKING THE
7501      ;READ DATA COMMAND
7502 044414      ERRSOFT ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      044414 104457      TRAP    CSERSOFT
      044416 001033      .WORD  539
      044420 005011      .WORD  WRTErr
      044422 011670      .WORD  PKTSSR
7503 044424      75$:   CKLOOP      ;LOOP IF SELECTED      TRAP    CSCLP1
      044424 104406
7504 044426 005723      TST    (R3)+         ;BUMP RECORD SIZE
7505 044430 022703 000414      CMP    #268.,R3     ;END OF RECORD YET
7506 044434 001346      BNE    65$          ;BR, IF MORE RECORDS TO WRITE
7507 044436      80$:   CKLOOP      ;LOOP IF SELECTED      TRAP    CSCLP1
      044436 104406
7508 044440 005743      TST    -(R3)        ;SET BACK TO 512.
7509 044442 013737 003072 050172      MOV    FREE,T24RB    ;STARTING READ BUFFER ADDRESS
7510      ;*****
7511      ;READ REVERSE DATA,ACK COMMAND
7512      ;*****
7513      ;*****
7514      ;*****
7515 044450 012737 100401 050170      MOV    #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
7516 044456 012704 050170      165$:  MOV    #T24PK3,R4    ;SET UP R4 WITH PACKET ADDRESS
7517 044462 010337 050176      MOV    R3,T24SZ      ;SET UP RECORD SIZE IN PACKET
7518 044466 010465 177776      MOV    R4,TSDB(R5)   ;ISSUE COMMAND
7519 044472 004737 017104      JSR    PC,WAITF      ;WAIT FOR SSR TO SET
7520 044476 016501 000000      MOV    TSSR(R5),R1   ;GET TSSR CONTENTS
7521 044502 012702 000200      MOV    #SSR,R2      ;SET UP EXPECTED
7522 044506 020102      CMP    R1,R2         ;ARE THEY EQUAL
7523 044510 001406      BEQ    170$        ;BR, IF OK

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 106-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7610 ;ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7611 :
7612 :*****
7613 :
7614 044726 004737 010424 JSR PC,REWIND ;CALL TAPE REWIND COMMAND
7615 044732 103407 BCS 30$ ;BR, IF NO PROBLEM
7616 044734 010001 MOV R0,R1 ;SAVE TSSR
7617 044736 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7621 044742 ERRHRD ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
      044742 104456 TRAP CSERHRI
      044744 001040 .WORD 544
      044746 051326 .WORD T24RWN
      044750 011670 .WORD PKTSSR
7622 044752 30$: CKLOOP ;LOOP IF SELECTED TRAP CSCLP1
      044752 104406
7623 044754 012703 000400 MOV #256.,R3 ;RECORD SIZE
7624 044760 013737 003072 050172 MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS
7625 :
7626 :*****
7627 :
7628 :WRITE DATA,ACK,CVC=1,SWB COMMAND
7629 :
7630 :*****
7631 :
7632 044766 012737 150005 050170 MOV #150005,T24PK3 ;WRITE DATA,ACK,CVC=1,SWB COMMAND
7633 044774 012704 050170 MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7634 045000 65$:
7635 045000 010300 MOV R3,R0 ;SET PATTERN IN CORRECT REGISTER
7636 045002 004737 020356 JSR PC,FILLMEM ;FILL MEMORY WITH RECORD SIZE
7637 045006 010337 050176 MOV R3,T24S2 ;SET UP RECORD SIZE IN PACKET
7638 045012 010465 177776 MOV #4,T24S2 ;ISSUE COMMAND
7639 045016 004737 017104 JSR PC,WAITF ;WAIT FOR SSR TO SET
7640 045022 016501 000000 MOV TSSR(R5),R1 ;GET TSSR CONTENTS
7641 045026 012702 000200 MOV #SSR,R2 ;SET UP EXPECTED
7642 045032 020102 CMP R1,R2 ;ARE THEY EQUAL
7643 045034 001406 BEQ 75$ ;BR, IF OK
7644 045036 004737 020064 JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7648 ;SOFT ERROR, REALLY CHECKING THE
7649 ;READ DATA COMMAND
7650 ERRSOFT ERRNO,WRERR,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
      045042 104457 TRAP CSERSOFT
      045044 001041 .WORD 545
      045046 005011 .WORD WRERR
      045050 011670 .WORD PKTSSR
7651 045052 75$: CKLOOP ;LOOP IF SELECTED TRAP CSCLP1
      045052 104406
7652 045054 005723 TST (R3)+ ;BUMP RECORD SIZE
7653 045056 022703 000414 CMP #268.,R3 ;END OF RECORD YET
7654 045062 001346 BNE 65$ ;BR, IF MORE RECORDS TO WRITE
7655 045064 80$: CKLOOP ;LOOP IF SELECTED TRAP CSCLP1
      045064 104406
7656 045066 005743 TST -(R3) ;SET RECORD SIZE BACK TO 512.
7657 045070 013737 003072 050172 MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS
7658 :
7659 :*****
7660 :
7661 :READ REVERSE DATA,ACK,SWB COMMAND

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 106-2

TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7662
7663
7664
7665 045076 012737 110401 050170
7666 045104 012704 050170
7667 045110 010337 050176
7668 045114 010465 177776
7669 045120 004737 017104
7670 045124 016501 000000
7671 045130 012702 000200
7672 045134 020102
7673 045136 001406
7674 045140 004737 020064
7678 045144
    045144 104456
    045146 001042
    045150 051656
    045152 016330
7679 045154
    045154 104406
7680 045156 013702 003072
7681 045162 010304
7682 045164 162704 000400
7683 045170 060204
7684 045172 021403
7685 045174 001410
7686 045176 011401
7687 045200 010302
7688 045202 004737 020064
7692 045206
    045206 104456
    045210 001043
    045212 051110
    045214 016330
7693 045216
    045216 104406
7694 045220 005724
7695 045222 160204
7696 045224 020403
7697 045226 001360
7698 045230 005743
7699 045232 022703 000400
7700 045236 001322
7701 045240
    045240 104406
7702 045242
    045242
    045242 104403
7703 045244 023727 002170 000031
7704 045252 002402
7705 045254 004737 020136
7706 045260

```

:
:.....
165\$: MOV #110401,T24PK3 :READ REVERSE DATA,ACK,SUB COMMAND
MOV #T24PK3,R4 :SET UP R4 WITH PACKET ADDRESS
MOV R3,T24SZ :SET UP RECORD SIZE IN PACKET
MOV R4,TSDDB(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 170\$:BR, IF OK
JSR PC,FATCHK :INC AND CHECK FOR MORE THAN 25 ERRORS
ERRHRD ERRNO,T24WDC,EXPREC :TSSR INCORRECT AFTER READ DATA
TRAP C\$ERHRD
WORD 546
WORD T24WDC
WORD EXPREC
170\$: CKLOOP :LOOP IF SELECTED
TRAP C\$CLP1
MOV FREE,R2 :GET BUFFER ADDRESS
MOV R3,R4 :CURRENT RECORD SIZE
SUB #256.,R4 :FIRST LOCATION IN BUFFER
173\$: ADD R2,R4 :SET POINTER TO FRAME (WORD)
CMP (R4),R3 :CHECK DATA READ (R3=DATA ALSO)
BEQ 180\$:BR, IF ALL IS WELL
MOV (R4),R1 :RECD DATA
MOV R3,R2 :EXPECTED DATA
JSR PC,FATCHK :INC AND CHECK FOR MORE THAN 25 ERRORS
ERRHRD ERRNO,T24DTA,EXPREC :DATA READ NOT = WRITTEN
TRAP C\$ERHRD
WORD 547
WORD T24DTA
WORD EXPREC
180\$: CKLOOP :LOOP IF SELECTED
TRAP C\$CLP1
TST (R4)+ :BUMP TO NEXT LOCATION
SUB R2,R4 :GET RID OF BASE ADDRESS
CMP R4,R3 :END OF RECORD YET
BNE 173\$:BR, IF NOT AT END OF RECORD
TST -(R3) :BUMP RECORD SIZE
CMP #256.,R3 :END OF RECORD YET
BNE 165\$:BR, IF MORE RECORDS TO WRITE
190\$: CKLOOP :LOOP IF SELECTED
TRAP C\$CLP1
ENDSUB :>>>>>>>>> END SUBTEST >>>>>>>>>
L10064:
TRAP C\$ESUB
7703: CMP FATFLG,#25. :IS ERROR COUNT AT 25
BLT 999\$:BR, IF LESS THAN 25
JSR PC,CKDROP :TRY TO DROP THE UNIT
999\$:

```

7708
7709
7710      :+
7711      :TEST 5, SUBTEST 7
7712      :
7713      :VERIFIES THAT A READ REVERSE COMMAND READING A RECORD
7714      :LONGER THAN THE SPECIFIED BYTE COUNT CAUSES TAPE
7715      :STATUS ALERT TERMINATION WITH THE RECORD LENGTH LONG
7716      : (RLL) BIT SET.
7717      :
7718      :-
7719
7720      045260      BGNSUB                :>>>>>>>>>>>> BEGIN SUBTEST >>>>>>>>>>>>
          045260
          045260 104402
7721      045262 004737 052664          JSR      PC,T24RT3          ;SET UP OTHER COMMAND PACKET   TRAP    CSBSUB
7722      045266 004737 052530          JSR      PC,T24REST        ;SET COMMAND PACKET
7723      045272 004737 052622          JSR      PC,T24RT2        ;SET UP OTHER COMMAND PACKET
7724
7725      :*****
7726      :
7727      :ISSUE CONTROLLER "SOFT" INITIALIZE - CARRY BIT CLEAR IF ERROR
7728      :
7729      :*****
7730
7731      045276 004737 016630          JSR      PC,SOFINIT       ;DO INITIALIZE ON CONTRLLER
7732      045302 103407                    BCS      20$              ;BR IF INIT WAS OK
7733      045304 004737 020064          JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7737      045310 010001                    MOV      R0,R1            ;CONTENTS OF TSSR REGISTER
7738      045312                    ERRDF   ERRNO,SFIERR,SFIMSG ;FATAL ERROR TSSR WAS NOT OK
          045312 104455
          045314 001044                    TRAP    CSERDF 548
          045316 003550                    .WORD  SFIERR
          045320 011656                    .WORD  SFIMSG
7739      045322
7740      045322 012704 050040      20$:  MOV      #T24PACKET,R4          ;SUBROUTINE NEEDS PACKET ADDRESS
7741
7742      :*****
7743      :
7744      :WRITE CHARACTERISTICS COMMAND (CALL TO WRTCHR)
7745      :
7746      :*****
7747
7748      045326 004737 010322          JSR      PC,WRTCHR        ;ISSUE WRITE CHARACTERISTICS
7749      045332 103407                    BCS      24$              ;BR, IF COMMAND ISSUED OK
7750      045334 004737 020064          JSR      PC,FATCHK        ;INC AND CHECK FOR MORE THAN 25 ERRORS
7754      045340 010001                    MOV      R0,R1            ;SAVE CONTENTS OF TSSR
7755      045342                    ERRHRD  ERRNO,WRTMSG,SFIMSG ;WRITE CHARACTERISTIC FAILED
          045342 104456                    TRAP    CSERHRD 549
          045344 001045                    .WORD  WRTMSG
          045346 004754                    .WORD  SFIMSG
          045350 011656                    .WORD
7756      045352
          045352 104406      24$:  CKLOOP          ;LOOP IF SELECTED          TRAP    CSCLP1
7757
7758      :*****
7759
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 107-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7760      :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
7761      :
7762      :*****
7763      :
7764 045354 004737 010424      JSR      PC,REWIND      :CALL TAPE REWIND COMMAND
7765 045360 004737 017220      JSR      PC,CHKTSSR     :SEE HOW TSSR IS
7766 045364 103407              BCS      30$            :BR, IF NO PROBLEM
7767 045366 010001              MOV      R0,R1         :SAVE TSSR
7768 045370 004737 020064      JSR      PC,FATCHK     :INC AND CHECK FOR MORE THAN 25 ERRORS
7772 045374 004737 020064      ERRHRD  ERRNO,T24RWN,PKTSSR :REWIND NOT ACCEPTED
                                TRAP      CSERHRD
                                .WORD     550
                                .WORD     T24RWN
                                .WORD     PKTSSR
                                TRAP      CSCLP1
7773 045404 104406      30$:    CKLOOP              :LOOP IF SELECTED
7774 045406 012703 001000      MOV      #512.,R3      :RECORD SIZE
7775 045412 013737 003072 050172  MOV      FREE,T24RB    :STARTING WRITE BUFFER ADDRESS
7776      :
7777      :*****
7778      :
7779      :WRITE DATA,ACK,CVC=1 COMMAND
7780      :
7781      :*****
7782      :
7783 045420 012737 140005 050170  MOV      #140005,T24PK3 :WRITE DATA,ACK,CVC=1 COMMAND
7784 045426 012704 050170      MOV      #T24PK3,R4    :SET UP R4 WITH PACKET ADDRESS
7785 045432 010337 050176      65$:    MOV      R3,T24SZ    :SET UP RECORD SIZE IN PACKET
7786 045432 010337 050176      MOV      R4,TSDB(R5)   :ISSUE COMMAND
7787 045436 010465 177776      JSR      PC,WAITF      :WAIT FOR SSR TO SET
7788 045442 004737 017104      MOV      TSSR(R5),R1   :GET TSSR CONTENTS
7789 045446 016501 000000      MOV      #SSR,R2      :SET UP EXPECTED
7790 045452 012702 000200      CMP      R1,R2        :ARE THEY EQUAL
7791 045456 020102              BEQ      75$           :BR, IF OK
7792 045460 001406              JSR      PC,FATCHK     :INC AND CHECK FOR MORE THAN 25 ERRORS
7793 045462 004737 020064      ERRSOFT ERRNO,WRTErr,PKTSSR :SOFT ERROR, REALLY CHECKING THE
                                :READ DATA COMMAND
                                :TSSR INCORRECT AFTER WRITE DATA
                                TRAP      CSERSOFT
                                .WORD     551
                                .WORD     WRTErr
                                .WORD     PKTSSR
7799 045466 104457      75$:    CKLOOP              :LOOP IF SELECTED
                                TRAP      CSCLP1
7800 045476 104406      75$:    CKLOOP              :LOOP IF SELECTED
7801 045500 012703 000400      MOV      #256.,R3     :SIZE OF RECORD
7802 045504 013737 003072 050172  MOV      FREE,T24RB    :STARTING READ BUFFER ADDRESS
7803      :
7804      :*****
7805      :
7806      :READ DATA,ACK COMMAND
7807      :
7808      :*****
7809      :
7810 045512 012737 100401 050170  MOV      #100401,T24PK3 :READ DATA,ACK COMMAND
7811 045520 012704 050170      165$:  MOV      #T24PK3,R4    :SET UP R4 WITH PACKET ADDRESS
7812 045524 010337 050176      MOV      R3,T24SZ    :SET UP RECORD SIZE IN PACKET

```

CZTUXAO TUBO FRONT END PRT B MACRO M120G 29-MAR-83 13:32 PAGE 107-2
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

7813	045530	010465	177776	MOV	R4,TSDB(R5)	:ISSUE COMMAND				
7814	045534	004737	017104	JSR	PC,WAITF	:WAIT FOR SSR TO SET				
7815	045540	016501	00C000	MOV	TSSR(R5),R1	:GET TSSR CONTENTS				
7816	045544	012702	100204	MOV	#SSR!SC!BIT2,R2	:SET UP EXPECTED				
7817	045550	020102		CMP	R1,R2	:ARE THEY EQUAL				
7818	045552	001406		BEQ	170\$:BR, IF OK				
7819	045554	004737	020064	JSR	PC,FATCHK	:INC AND CHECK FOR MORE THAN 25 ERRORS				
7823	045560			ERRHRD	ERRNO,T24TRL,EXPREC	:TSSR INCORRECT AFTER READ DATA				
	045560	104456					TRAP	C\$ERHRD		
	045562	001050					.WORD	552		
	045564	052374					.WORD	T24TRL		
	045566	016330					.WORD	EXPREC		
7824	045570			170\$:	CKLOOP	:LOOP IF SELECTED				
	045570	104406					TRAP	C\$CLP1		
7825										
7826										
7827										
7828										
7829										
7830										
7831										
7832	045572	013701	050066	MOV	T24BfR+6,R1	:GET MESSAGE BUFFER (XSTO)				
7833	045576	010102		MOV	R1,R2	:SET UP EXPECTED				
7834	045600	052702	010000	BIS	#BIT1?,R2	:SET THE RLL BIT IN EXPECTED				
7835	045604	020102		CMP	R1,R2	:ARE THEY EQUAL				
7836	045606	001406		BEQ	180\$:BR, IF EQUAL (ALL IS WELL)				
7837	045610	004737	020064	JSR	PC,FATCHK	:INC AND CHECK FOR MORE THAN 25 ERRORS				
7841	045614			ERRHRD	ERRNO,T24LON,EXPREC	:THE RLL BIT WAS NOT SET IN XSTO				
	045614	104456					TRAP	C\$ERHRD		
	045616	001051					.WORD	553		
	045620	052142					.WORD	T24LON		
	045622	016330					.WORD	EXPREC		
7842	045624			180\$:						
7843	045624				ENDSUB	:>>>>>>>>> END SUBTEST >>>>>>>>>				
	045624					L10065:				
	045624	104403					TRAP	C\$ESUB		
7844	045626	023727	002170	000031	CMP	FATFLG,#25.	:IS ERROR COUNT AT 25			
7845	045634	002402			BLT	999\$:BR, IF LESS THAN 25			
7846	045636	004737	020136		JSR	PC,CKDROP	:TRY TO DROP THE UNIT			
7847	045642			999\$:						
7848				:	*					

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 108-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

7902
7903
7904
7905
7906
7907
7908
7909 045756 004737 010424      JSR      PC,REWIND      ;CALL TAPE REWIND COMMAND
7910 045762 004737 017220      JSR      PC,CHKTSSR    ;SEE HOW TSSR IS
7911 045766 103407                BCS      30$           ;BR, IF NO PROBLEM
7912 045770 010001                MOV      R0,R1        ;SAVE TSSR
7913 045772 004737 020064      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7917 045776                ERRHRD  ERRNO,T24RWN,PKTSSR ;REWIND NOT ACCEPTED
                                TRAP      CSERHRD
                                .WORD     556
                                .WORD     T24RWN
                                .WORD     PKTSSR
7918 046006 104406      30$:  CKLOOP                ;LOOP IF SELECTED      TRAP      CSCLP1
                                .WORD     556
                                .WORD     T24RWN
                                .WORD     PKTSSR
7919 046010 012703 000400      MOV      #256.,R3     ;RECORD SIZE
7920 046014 013737 003072 050172  MOV      FREE,T24RB   ;STARTING WRITE BUFFER ADDRESS
7921
7922
7923
7924
7925
7926
7927
7928 046022 012737 140005 050170      MOV      #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
7929 046030 012704 050170      MOV      #T24PK3,R4   ;SET UP R4 WITH PACKET ADDRESS
7930 046034                65$:
7931 046034 010337 050176      MOV      R3,T24SZ    ;SET UP RECORD SIZE IN PACKET
7932 046040 010465 177776      MOV      R4,TSDB(R5) ;ISSUE COMMAND
7933 046044 004737 017104      JSR      PC,WAITF    ;WAIT FOR SSR TO SET
7934 046050 016501 000000      MOV      TSSR(R5),R1 ;GET TSSR CONTENTS
7935 046054 012702 000200      MOV      #SSR,R2    ;SET UP EXPECTED
7936 046060 020102                CMP      R1,R2      ;ARE THEY EQUAL
7937 046062 001406                BEQ      75$        ;BR, IF OK
7938 046064 004737 020064      JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
7942 046070                ERRSOFT ERRNO,WRTErr,PKTSSR ;TSSR INCORRECT AFTER WRITE DATA
                                TRAP      CSERSOFT
                                .WORD     557
                                .WORD     WRTErr
                                .WORD     PKTSSR
7943 046100 104406      75$:  CKLOOP                ;LOOP IF SELECTED      TRAP      CSCLP1
                                .WORD     557
                                .WORD     WRTErr
                                .WORD     PKTSSR
7944 046102 012703 000400      MOV      #256.,R3     ;RECORD SIZE
7945 046106 012701 160000      MOV      #160000,R1   ;START POSSIBLE NXM ADDRESS
7946 046112 012702 177776      MOV      #177776,R2   ;END POSSIBLE NXM ADDRESS
7947 046116 004737 017260      JSR      PC,XXM      ;CALL NXM FINDER ROUTINE
7948 046122 103402                BCS      76$        ;BR IF NXM ADDRESS FOUND
7949 046124 000137 046300      JMP      180$        ;JMP OVER CAN'T FIND NXM
7950 046130 010137 050172      76$:  MOV      R1,T24RB   ;STARTING READ REVERSE BUFFER ADDRESS
7951
7952
7953
7954

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 108-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

7955				:			
7956				:	*****		
7957				:			
7958	046134	013737	050216		050174	MOV	T24DLY,T24RB+2 ;GET BITS 16 AND 17
7959	046142	012737	100401		050170	MOV	#100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
7960	046150	012704	050170	165\$:		MOV	#T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
7961	046154	012737	000400		050176	MOV	#256.,T24SZ ;SET UP RECORD SIZE IN PACKET
7962	046162	010465	177776			MOV	R4,TSD8(R5) ;ISSUE COMMAND
7963	046166	004737	017104			JSR	PC,WAITF ;WAIT FOR SSR TO SET
7964	046172	016501	000000			MOV	TSSR(R5),R1 ;GET TSSR CONTENTS
7965	046176	012702	104210			MOV	#SSR!NXM!SC!BIT3,R2 ;SET UP EXPECTED
7966	046202	020102				COMP	R1,R2 ;ARE THEY EQUAL
7967	046204	001417				BEQ	170\$;BR, IF OK
7968	046206	062737	000001		050216	ADD	#1,T24DLY ;NEXT BUNCH OF MEMORY
7969	046214	022737	000004		050216	COMP	#4,T24DLY ;TOO MUCH MEMORY
7970	046222	001402				BEQ	168\$;BR IF OVER
7971	046224	000137	045664			JMP	10\$;TRY AGAIN
7972	046230	004737	02C064	168\$:		JSR	PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7976	046234					ERRHRD	ERRNO,T24NXM,PKTSSR ;TSSR INCORRECT AFTER READ DATA
	046234	104456					TRAP CSERHRD
	046236	001056					.WORD 558
	046240	050431					.WORD T24NXM
	046242	011670					.WORD PKTSSR
7977	046244			170\$:	CKLOOP		;LOOP IF SELECTED
	046244	104406					TRAP CSLLP1
7978							
7979				:	*****		
7980				:			
7981				:	READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XSTO)		
7982				:			
7983				:	*****		
7984							
7985	046246	013701	050066			MOV	T24BFR+6,R1 ;GET MESSAGE BUFFER
7986	046252	010102				MOV	R1,R2 ;SET UP EXPECTED
7987	046254	052702	040000			BIS	#BIT14,R2 ;SET THE RLS BIT IN EXPECTED
7988	046260	020102				COMP	R1,R2 ;ARE THEY EQUAL
7989	046262	001406				BEQ	180\$;BR, IF EQUAL (ALL IS WELL)
7990	046264	004737	020064			JSR	PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
7994	046270					ERRHRD	ERRNO,T24LOP,EXPREC ;THE RLL BIT WAS NOT SET IN XSTO
	046270	104456					TRAP CSERHRD
	046272	001057					.WORD 559
	046274	052224					.WORD T24LOP
	046276	016330					.WORD EXPREC
7995	046300			180\$:			
7996	046300				ENDSUB		>>>>>>>>> END SUBTEST >>>>>>>>>
	046300						L10066:
	046300	104403					TRAP CSesub
7997	046302	023727	002170		000031	COMP	FATFLG,#25. ;IS ERROR COUNT AT 25
7998	046310	002402				BLT	999\$;BR, IF LESS THAN 25
7999	046312	004737	020136			JSR	PC,CKDROP ;TRY TO DROP THE UNIT
8000	046315			999\$:			

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 109-1 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8054      :*****
8055      :
8056      :LEGAL MODE,ACK,CVC=1,READ COMMAND
8057      :
8058      :*****
8059
8060 046426 012737 140001 050170      MOV      #140001,T24PK3      ;LEGAL MODE,ACK,CVC=1,READ COMMAND
8061 046434 012704 050170              MOV      #T24PK3,R4          ;SET UP R4 WITH PACKET ADDRESS
8062 046440 012737 000400 050176      MOV      #256.,T24SZ        ;SET UP RECORD SIZE IN PACKET
8063 046446 010465 177776              MOV      R4,TSDB(R5)        ;ISSUE COMMAND
8064 046452 004737 017104              JSR      PC,WAITF          ;WAIT FOR SSR!BIT1!BIT2 TO SET
8065 046456 016501 000000              MOV      TSSR(R5),R1        ;GET TSSR CONTENTS
8066 046462 012702 100206              MOV      #SSR!SC!BIT1!BIT2,R2 ;SET UP EXPECTED
8067 046466 020102                    CMP      R1,R2              ;ARE THEY EQUAL
8068 046470 001406                    BEQ      75$                ;BR, IF OK
8069 046472 004737 020064              JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
8073 046476                                ERRHRD  ERRNO,T24WDG,PKTSSR ;TSSR INCORRECT AFTER READ DATA
                                TRAP      CSERHRD
                                .WORD    562
                                .WORD    T24WDG
                                .WORD    PKTSSR
8074 046506      75$:      CKLOOP                    ;LOOP IF SELECTED
                                TRAP      C$CLP1
                                .WORD    104406
8075
8076      :*****
8077      :
8078      :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8079      :
8080      :*****
8081
8082 046510 013701 050066      MOV      T24BFR+6,R1        ;GET MESSAGE BUFFER
8083 046514 010102                    MOV      R1,R2              ;SET UP EXPECTED
8084 046516 052702 000400      BIS      #BIT8,R2           ;SET THE ILA BIT IN EXPECTED
8085 046522 020102                    CMP      R1,R2              ;ARE THEY EQUAL
8086 046524 001406                    BEQ      180$               ;BR, IF EQUAL (ALL IS WELL)
8087 046526 004737 020064              JSR      PC,FATCHK          ;INC AND CHECK FOR MORE THAN 25 ERRORS
8091 046532                                ERRHRD  ERRNO,T24ILA,EXPREC ;THE ILA BIT WAS NOT SET IN XST0
                                TRAP      CSERHRD
                                .WORD    563
                                .WORD    T24ILA
                                .WORD    EXPREC
8092 046542      180$:
8093 046542                                ENDSUB                      ;>>>>>>>>>>>> END SUBTEST >>>>>>>>>>>>
                                L10067:
                                TRAP      C$ESUB
                                .WORD    104403
8094 046544 023727 002170 000031      CMP      FATFLG,#25.        ;IS ERROR COUNT AT 25
8095 046552 002402                    BLT     999$                ;BR, IF LESS THAN 25
8096 046554 004737 020136              JSR      PC,CKDROP          ;TRY TO DROP THE UNIT
8097 046560      999$:

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 110-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8152 046672          24$:  CKLOOP                :LOOP IF SELECTED
      046672  104406
8153 046674  012701  160000          MOV      #160000,R1      ;START POSSIBLE NXM ADDRESS
8154 046700  012702  177776          MOV      #177776,R2      ;END POSSIBLE NXM ADDRESS
8155 046704  004737  017260          JSR      PC,NXM         ;CALL NXM FINDER ROUTINE
8156 046710  103402          BCS      76$           ;BR IF WE FOUND NXM
8157 046712  000137  047034          JMP      180$          ;JUMP OVER, CAN'T FIND NXM
8158 046716  010137  050172          76$:  MOV      R1,T24RB    ;STARTING READ BUFFER ADDRESS
8159
8160  :*****
8161  :READ, ACK, CVC=1, COMMAND
8162  :*****
8163
8164
8165 046722  013737  050216  050174          MOV      T24DLY,T24RB+2 ;MEMORY BITS 16 AND 17
8166 046730  012737  140001  050170          MOV      #140001,T24PK3 ;READ, ACK, CVC=1, COMMAND
8167 046736  012704  050170          MOV      #T24PK3,R4     ;SET UP R4 WITH PACKET ADDRESS
8168 046742  012737  000400  050176          MOV      #256.,T24SZ    ;SET UP RECORD SIZE IN PACKET
8169 046750  010465  177776          MOV      R4,TSDB(R5)    ;ISSUE COMMAND
8170 046754  004737  017104          JSR      PC,WAITF       ;WAIT FOR SSR!BIT1!BIT2 TO SET
8171 046760  016501  000000          MOV      TSSR(R5),R1    ;GET TSSR CONTENTS
8172 046764  012702  104210          MOV      #SSR!SC!NXM!BIT3,R2 ;SET UP EXPECTED
8173 046770  020102          CMP      R1,R2         ;ARE THEY EQUAL
8174 046772  001417          BEQ      75$           ;BR, IF OK
8175 046774  062737  000001  050216          ADD      #1,T24DLY      ;NEXT BUNCH OF MEMORY
8176 047002  022737  000004  050216          CMP      #4,T24DLY      ;TOO MUCH MEMORY
8177 047010  001402          BEO      50$           ;END OF LINE
8178 047012  000137  046602          JMP      10$           ;TRY AGAIN
8179 047016  004737  020064          50$:  JSR      PC,FATCHK    ;INC AND CHECK FOR MORE THAN 25 ERRORS
8183 047022          ERRHRD  ERRNO,T24NXM,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      047022  104456
      047024  001066          TRAP   CSERHRD
      047026  050431          .WORD  566
      047030  011670          .WORD  T24NXM
      047032          .WORD  PKTSSR
8184 047032          75$:  CKLOOP                :LOOP IF SELECTED
      047032  104406
8185 047034          80$:
8186 047034          180$:  ENDSUB
      047034
      047034  104403
8187 047036  023727  002170  000031          CMP      FATFLG,#25.    ;IS ERROR COUNT AT 25
8188 047044  002402          BLT     999$           ;BR, IF LESS THAN 25
8189 047046  004737  020136          JSR      PC,CKDROP      ;TRY TO DROP THE UNIT
8190 047052          999$:

```


CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 111-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8244          :ISSUE REWIND COMMAND TO SELECTED TAPE DRIVE
8245          :
8246          :*****
8247          :
8248 047146 004737 010424          JSR      PC,REWIND          :CALL TAPE REWIND COMMAND
8249 047152 004737 017220          JSR      PC,CHKTSSR        :SEE HOW TSSR IS
8250 047156 103407                   BCS      30$                :BR, IF NO PROBLEM
8251 047160 010001                   MOV      R0,R1              :SAVE TSSR
8252 047162 004737 020064          JSR      PC,FATCHK         :INC AND CHECK FOR MORE THAN 25 ERRORS
8256 047166                   ERRHRD  ERRNO,T24RWN,PKTSSR :REWIND NOT ACCEPTED
                                TRAP  CSERHRD
                                .WORD 569
                                .WORD T24RWN
                                .WORD PKTSSR
      047166 104456
      047170 001071
      047172 051326
      047174 011670
8257 047176          30$:  CKLOOP          :LOOP IF SELECTED          TRAP  CSCLP1
      047176 104406
8258 047200 012703 000400          MOV      #256,R3          :RECORD SIZE
8259 047204 013737 003072 050172  MOV      FREE,T24RB       :STARTING WRITE BUFFER ADDRESS
8260
8261          :*****
8262          :
8263          :READ REVERSE DATA,ACK COMMAND
8264          :
8265          :*****
8266          :
8267 047212 012737 100401 050170  MOV      #100401,T24PK3   :READ REVERSE DATA,ACK COMMAND
8268 047220 012704 050170          MOV      #T24PK3,R4      :SET UP R4 WITH PACKET ADDRESS
8269 047224          65$:
8270 047224 010337 050176          MOV      R3,T24SZ        :SET UP RECORD SIZE IN PACKET
8271 047230 010465 177776          MOV      R4,TSDB(R5)     :ISSUE COMMAND
8272 047234 004737 017104          JSR      PC,WAITF        :WAIT FOR SSR TO SET
8273 047240 016501 000000          MOV      TSSR(R5),R1     :GET TSSR CONTENTS
8274 047244 012702 100206          MOV      #SSR!SC!BIT1!BIT2,R2 :SET UP EXPECTED
8275 047250 020102                   CMP      R1,R2           :ARE THEY EQUAL
8276 047252 001406                   BEQ      75$             :BR, IF OK
8277 047254 004737 020064          JSR      PC,FATCHK         :INC AND CHECK FOR MORE THAN 25 ERRORS
8281 047260                   ERRHRD  ERRNO,T24WDE,PKTSSR :TSSR INCORRECT AFTER READ DATA
                                TRAP  CSERHRD
                                .WORD 570
                                .WORD T24WDE
                                .WORD PKTSSR
      047260 104456
      047262 001072
      047264 050771
      047266 011670
8282 047270          75$.  CKLOOP          :LOOP IF SELECTED          TRAP  CSCLP1
      047270 104406
8283
8284          :*****
8285          :
8286          :READ MESSAGE BUFFER EXTENDED STATUS REGISTER ZERO (XST0)
8287          :
8288          :*****
8289          :
8290 047272 013701 050066          MOV      T24BFR+6,R1     :GET MESSAGE BUFFER
8291 047276 010102                   MOV      R1,R2           :SET UP EXPECTED
8292 047300 052702 002000          BIS      #BIT10,R2       :SET THE NEF BIT IN EXPECTED
8293 047304 020102                   CMP      R1,R2           :ARE THEY EQUAL
8294 047306 001406                   BEQ      180$            :BR, IF EQUAL (ALL IS WELL)
8295 047310 004737 020064          JSR      PC,FATCHK         :INC AND CHECK FOR MORE THAN 25 ERRORS
8299 047314                   ERRHRD  ERRNO,T24NEF,EXPREC :THE RLL BIT WAS NOT SET IN XST0

```


CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 112-1
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8359 047464 011670
      047466 104406
      047466 104406
8360 047470 012703 000400
8361 047474 013737 003072 050172
8362
8363
8364
8365
8366
8367 047502 012737 140005 050170
8368 047510 012704 050170
8369 047514 010337 050176
8370 047520 010465 177776
8371 047524 004737 017104
8372 047530 016501 000000
8373 047534 012702 000200
8374 047540 020102
8375 047542 001406
8376 047544 004737 020064
8380
8381
8382 047550
      047550 104457
      047552 001077
      047554 005011
      047556 011670
8383 047560
      047560 104406
8384 047562 012703 000400
8385 047566 013737 003072 050172
8386
8387
8388
8389
8390
8391 047574 012737 100401 050170
8392 047602 012704 050170
8393 047606 010337 050176
8394 047612 010465 177776
8395 047616 004737 017104
8396 047622 016501 000000
8397 047626 012702 000200
8398 047632 020102
8399 047634 001406
8400 047636 004737 020064
8404 047642
      047642 104456
      047644 001100
      047646 052374
      047650 011670
8405 047652
      047652 104406
8406 047654 012703 000400
8407 047660 013737 003072 050172
8408
8409

```

30\$: CKLOOP ;LOOP IF SELECTED .WORD PKTSSR
;RECORD SIZE TRAP C\$CLP1
MOV #256.,R3 ;RECORD SIZE
MOV FREE,T24RB ;STARTING WRITE BUFFER ADDRESS

:WRITE DATA,ACK,CVC=1 COMMAND

MOV #140005,T24PK3 ;WRITE DATA,ACK,CVC=1 COMMAND
MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
65\$: MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
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
JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
;SOFT ERROR, REALLY CHECKING THE
;READ REVERSE DATA COMMAND
ERRSOFT ERRNO,WRterr,PKTSSR ;TSSR INCORRECT AFTER READ DATA
TRAP CSERSOFT
. WORD 575
. WORD WRterr
. WORD PKTSSR

75\$: CKLOOP ;LOOP IF SELECTED TRAP C\$CLP1
;RECORD SIZE
MOV #256.,R3 ;RECORD SIZE
MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS

:READ REVERSE DATA,ACK COMMAND

MOV #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
165\$: MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
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 170\$;BR, IF OK
JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
TRAP CSERHRD
. WORD 576
. WORD T24TRL
. WORD PKTSSR

170\$: CKLOOP ;LOOP IF SELECTED TRAP C\$CLP1
;RECORD SIZE
MOV #256.,R3 ;RECORD SIZE
MOV FREE,T24RB ;STARTING READ BUFFER ADDRESS

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 112-2
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8410 ;READ REVERSE DATA,ACK COMMAND
8411 :
8412 :*****
8413 047666 012737 100401 050170 195$: MOV #100401,T24PK3 ;READ REVERSE DATA,ACK COMMAND
8414 047674 012704 050170 195$: MOV #T24PK3,R4 ;SET UP R4 WITH PACKET ADDRESS
8415 047700 010337 050176 195$: MOV R3,T24SZ ;SET UP RECORD SIZE IN PACKET
8416 047704 010465 177776 195$: MOV R4,TSDR(R5) ;ISSUE COMMAND
8417 047710 004737 017104 195$: JSR PC,WAITF ;WAIT FOR SSR TO SET
8418 047714 016501 000000 195$: MOV TSSR(R5),R1 ;GET TSSR CONTENTS
8419 047720 012702 100204 195$: MOV #SSR!SC!BIT2,R2 ;SET UP EXPECTED
8420 047724 020102 195$: CMP R1,R2 ;ARE THEY EQUAL
8421 047726 001406 195$: BEQ 200$ ;BR, IF OK
8422 047730 004737 020064 195$: JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8426 047734 : ERRHRD ERRNO,T24TRL,PKTSSR ;TSSR INCORRECT AFTER READ DATA
      047734 104456 TRAP C$ERHRD
      047736 001101 .WORD 577
      047740 052374 .WORD T24TRL
      047742 011670 .WORD PKTSSR
8427 047744 200$: CKLOOP ;LOOP IF SELECTED
      047744 104406 TRAP C$CLP1
8428 047746 013701 050074 200$: MOV T24BFR+14,R1 ;GET MESSAGE BUFFER (XST3)
8429 047752 010102 200$: MOV R1,R2 ;SET UP EXPECTED
8430 047754 052702 000001 200$: BIS #BIT0,R2 ;SET THE RIB BIT IN EXPECTED
8431 047760 020102 200$: CMP R1,R2 ;ARE THEY EQUAL
8432 047762 001406 200$: BEQ 210$ ;BR, IF EQUAL (ALL IS WELL)
8433 047764 004737 020064 200$: JSR PC,FATCHK ;INC AND CHECK FOR MORE THAN 25 ERRORS
8437 047770 : ERRHRD ERRNO,T24LOR,EXPREC ;THE RIB BIT WAS NOT SET IN XST0
      047770 104456 TRAP C$ERHRD
      047772 001102 .WORD 578
      047774 050272 .WORD T24LOR
      047776 016330 .WORD EXPREC
8438 050000 210$: ENDSUB ;>>>>>>>>>> END SUBTEST >>>>>>>>>
      050000 L10072:
      050000 TRAP C$ESUB
8439 050002 023727 002170 000031 210$: CMP FATFLG,#25. ;IS ERROR COUNT AT 25
8440 050010 002402 210$: BLT 999$ ;BR, IF LESS THAN 25
8441 050012 004737 020136 210$: JSR PC,CKDROP ;TRY TO DROP THE UNIT
8442 050016 999$:
8443 050016 004737 017340 999$: JSR PC,TSTLOOP ;DO WE NEED TO ITERATE TEST
8444 050022 103002 999$: BCC 163$ ;BR, IF NO LOOP REQUIRED
8445 050024 000137 041514 999$: JMP T24LOOP ;EXECUTE AGAIN
8446 050030 163$:
8447 050030 : EXIT TST ;ALL DONE T1'S TEST
      050030 104432 TRAP C$EXIT
      050032 002662 .WORD L10056-.
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 113
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8449
8450
8451
8453 050034
8455 050040
8456 050040 100204
8457 050042 050050
8458 050044 000000
8459 050046 000012
8460 050050
8461 050050 050060
8462 050052 000000
8463 050054 000024
8464 050056 000000
8465 050060
8466
8467
8468
8470 050142
8472 050150
8473 050150 100206
8474 050152 050200
8475 050154 000000
8476 050156 000006
8477
8479 050160
8481 050170
8482 050170 100205
8483 050172
8484 050172 003072
8485 050174 000000
8486 050176 000000
8487
8488
8489
8490
8491 050200
8492 050200 010
8493 050201 200
8494 050202 000000
8495 050204 000000
8496
8497
8498
8499
8500
8501 050206 100005
8502 050210 100405
8503 050212 102005
8504 050214 177777
8505 050216 000000
8506
8507

:+
:LOCAL STORAGE FOR THIS TEST
:-
      .BLKB 10-<.-TUV2A&7>
T24PACKET:
      .WORD 100204
      .WORD T24DATA
      .WORD 0
      .WORD 10.
T24DATA:
      .WORD T24BFR
      .WORD 0
      .WORD 20.
      .WORD 0
T24BFR: .BLKW 25.
:WRITE SUBSYSTEM MEMORY COMMAND PACKET
      .BLKB 10-<.-TUV2A&7>
T24PK2:
      .WORD 100206
      .WORD T24BF2
      .WORD 0
      .WORD 6.
      .BLKB 10-<.-TUV2A&7>
T24PK3:
      .WORD 100205
T24RB:
T24WB: .WORD FREE
      .WORD 0
T24SZ: .WORD 0
      .EVEN
:
:
:
T24BF2:
T24BS0: .BYTE 10
T24BS1: .BYTE 200
T24S2: .WORD 0
T24S3: .WORD 0
:
:
      .EVEN
:TAPE MOTION PACKET COMMAND VALUES
T24RN: .WORD 100005
T24WDR: .WORD 100405
T24CON: .WORD 102005
      .WORD 177777
T24DLY: .WORD 0

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

:STARTING VALUE OF BLOCK SIZE
:CHARACTERISTICS DATA BLOCK
:ADDRESS OF MESSAGE BUFFER

:LENGTH OF MESSAGE BUFFER

:MESSAGE BUFFER

:WRITE SUB SYS MEM COMMAND, IE AND ACK
:ADDRESS OF SELECT BLOCK DATA

:SIZE OF DATA PACKET

:READ COMMAND, IE AND ACK

:ADDRESS OF WRITE BUFFER

:SIZE OF BUFFER (EXTENT)

:BSELO AREA
:BSEL1 AREA
:SEL 2 AREA
:DATA AREA

:READ DATA (NEXT)
:READ DATA RETRY
:WRITE CONTINGUS
:END OF DATA
:DELAY STORAGE AREA

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 114
TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

```

8509
8510
8511
8512
8513
8514 050220 116 105 106 T24NEF: .ASCIZ 'NEF Not Set After NON-EXECUTABLE FUNCTION'
8515 050272 122 111 102 T24LOR: .ASCIZ 'RIB Not Set After READ REVERSE Into BOT'
8516 050342 124 123 123 T24WDG: .ASCIZ 'TSSR Not Correct After Illegal Buffer Address Bits Set'
8517 050431 124 123 123 T24NXM: .ASCIZ 'TSSR Not Correct After NXM Memory Address In Packet'
8518 050515 124 123 123 T24WDF: .ASCIZ 'TSSR Not Correct After Illegal Mode Bits Set'
8519 050572 111 154 154 T24ILA: .ASCIZ 'Illegal Address Bits. Failed To Set ILA Bit In XSTO'
8520 050656 111 154 154 T24LOQ: .ASCIZ 'Illegal Mode Bits, Failed To Set ILC Bit In XSTO'
8521 050737 122 105 101 T24SSR: .ASCIZ 'READ COMMAND Not Accepted'
8522 050771 124 123 123 T24WDE: .ASCIZ 'TSSR Not Correct After WRITE DATA Command'
8523 051043 124 141 160 T24BOT: .ASCIZ 'Tape Not At BOT After REWIND Command'
8524 051110 104 141 164 T24DTA: .ASCIZ 'Data Written To Tape Not Equal To Data Read From Tape'
8525 051176 122 105 101 T24EOT: .ASCIZ 'READ DATA OVER EOT GAVE NO TAPE STATUS ALERT'
8526 051253 124 123 123 T24TM: .ASCIZ 'TSSR Not Correct After READ COMMAND Reject'
8527 051326 122 145 167 T24RUN: .ASCIZ 'Rewind (POSITION) Command Not Accepted'
8528 051375 122 101 115 T24RNC: .ASCIZ 'RAM Error, Correct Data Pattern Not In Ram'
8529 051450 124 123 123 T24AM3: .ASCIZ 'TSSR Init. Failed After READ COMMAND'
8530 051515 104 162 151 T24OFL: .ASCIZ 'Drive 7 Select Failed To Set 'DFL' In TSSR'
8531 051570 124 123 123 T24WDD: .ASCIZ 'TSSR Not Correct After READ DATA Command, SWB Bit Set'
8532 051656 124 123 123 T24WDC: .ASCIZ 'TSSR Not Correct After READ DATA Command'
8533 051727 103 126 103 T24VCK: .ASCIZ 'CVC Set, Didn't Reset VCK In Message Buffer'
8534 052002 124 123 102 T24BA: .ASCIZ 'TSBA Not Correct After READ DATA Command'
8535 052053 127 122 111 T24WSS: .ASCIZ 'WRITE SUBSYSTEM MEMORY Command Not Accepted (RAM Read)'
8536 052142 122 145 141 T24LON: .ASCIZ 'Reading Long Record Failed To Set RLL Bit In XSTO'
8537 052224 122 145 141 T24LOP: .ASCIZ 'Reading Long Record Failed To Set RLS Bit In XSTO'
8538 052306 122 145 163 T24PBP: .ASCIZ 'Residual Byte Count Incorrect After Short Record Read'
8539 052374 122 145 141 T24TRL: .ASCIZ 'Reading Long Record Failed To Give Tape Status Alert'
8540 052462 102 141 163 TST24ID: .ASCIZ 'Basic Read Data (Forward and Reverse)'
8541 .EVEN
8542
8543
8544
8545
8546
8547
8548
8549 052530 T24REST:
8550 052530 SAVREG ;SAVE THE REGISTERS
8551 052534 012701 050040 MOV #T24PACKET,R1 ;START OF THE PACKET
8552 052540 012721 100004 MOV #100004,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK,
8553 052544 012721 050050 MOV #T24DATA,(R1)+ ;ADDRESS OF CHARAISTICS DATA BLOCK
8554 052550 005021 CLR (R1)+ ;EXTENDED ADDRESS
8555 052552 012721 000012 MOV #10,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
8556 052556 012721 050060 MOV #T24BFR,(R1)+ ;ADDRESS OF MESSAGE BUFFER
8557 052562 005021 CLR (R1)+
8558 052564 012721 000024 MOV #20,(R1)+ ;LENGTH OF MESSAGE BUFFER
8559 052570 005021 CLR (R1)+
8560 052572 012711 000000 MOV #0,(R1) ;SELECT DRIVE ZERO
8561 052576 012702 000030 MOV #24,R2 ;NUMBER OF LOCATIONS TO BE CLEARED
8562 052602 012762 177777 050060 64$: MOV #177777,T24BFR(R2) ;ALL ONES TO MESSAGE BUFFER
8563 052610 005742 TST -(R2) ;NEXT LOCATION
8564 052612 022702 000000 CMP #0,R2 ;CHECK FOR END OF LOOP
8565 052616 001371 BNE 64$ ;KEEP GOING UNTIL DONE

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 114-1
 TEST 5: BASIC READ DATA (FORWARD AND REVERSE)

8566	052620	000207		RTS	PC		:RETURN
8567							
8568							
8569	052622			T24RT2:			
8570	052622			SAVREG			:SAVE THE REGISTERS
8571	052626	012701	050150	MOV	#T24PK2,R1		:START OF THE PACKET
8572	052632	012721	100206	MOV	#100206,(R1)+		:WRITE SUBSYSTEM MEM. WITH ACK, IE
8573	052636	012721	050200	MOV	#T24BF2,(R1)+		:ADDRESS OF DATA BLOCK
8574	052642	005021		CLR	(R1)+		:EXTENDED ADDRESS
8575	052644	012721	000006	MOV	#6,(R1)+		:SIZE OF DATA BLOCK IN BYTES
8576	052650	005021		CLR	(R1)+		
8577	052652	012701	050200	MOV	#T24BF2,R1		:POINT TO DATA SEL AREA
8578	052656	005021		CLR	(R1)+		
8579	052660	005011		CLR	(R1)		
8580	052662	000207		RTS	PC		:RETURN
8581	052664			T24RT3:			
8582	052664			SAVREG			:SAVE THE REGISTERS
8583	052670	012701	050170	MOV	#T24PK3,R1		:START OF THE PACKET
8584	052674	012721	000000	MOV	#0,(R1)+		:CLEAR AREA OUT
8585	052700	012721	000000	MOV	#0,(R1)+		:ADDRESS OF DATA BLOCK
8586	052704	005021		CLR	(R1)+		:EXTENDED ADDRESS
8587	052706	012711	000000	MOV	#0,(R1)		:SIZE OF DATA BLOCK IN BYTES
8588	052712	000207		RTS	PC		:RETURN
8589	052714			ENDTST			
	052714						L10056:
	052714	104401					TRAP CSETST

CZTUXAO TUBO FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115

.SBTTL TEST 6: MANUAL INTERVENTION

8591
8592
8593
8594
8595
8596
8597
8598
8599
8600
8601
8602
8603
8604
8605
8606
8607
8608
8609
8610
8611
8612
8613
8614
8615
8616
8617
8618
8619
8620
8621
8622
8623
8624
8625
8626
8627
8628
8629
8630
8631
8632
8633
8634
8635
8636
8637
8638
8639
8640
8641
8642

:THE MANUAL INTERVENTION TEST IS A STANDALONE ROUTINE (NOT REALLY A "TEST")
:THAT ALLOWS THE OPERATOR TO CHECK OUT VARIOUS ELEMENTS AND FUNCTIONS OF
:THE SUBSYSTEM THAT CANNOT BE MANIPULATED BY THE PROGRAM ALONE. WHEN
:THIS ROUTINE IS STARTED, IT FIRST PRINTS OUT A MENU OF SELECTABLE
:SUBTESTS AND THEN WAITS FOR THE OPERATOR TO TYPE IN A SELECTION CODE.
:THE ONLY WAYS TO EXIT THIS ROUTINE AND RETURN TO THE DIAGNOSTIC SUPERVISOR
:ARE BY TYPING <CTRL-C> OR SELECTING CODE 3.
:SELECTION CODES AND SUBROUTINES ARE:

CODE	ROUTINE
0	HELP. PRINTS THIS MENU.
1	REWIND AND UNLOAD COMMAND TEST
2	WRITE-PROTECT TEST
3	EXIT (RETURN TO SUPERVISOR)

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

:PRINTS OUT THE MENU ON THE CONSOLE TERMINAL.

:THIS ROUTINE INSTRUCTS THE OPERATOR TO PLACE THE DRIVE ON-LINE
:AND AT OR BEYOND BOT. THE TEST WILL THEN ISSUE THE REWIND AND
:UNLOAD COMMAND. IT WILL ALSO TELL THE OPERATOR IS THE DRIVE
:ENDED UP ON-LINE OR OFF-LINE.

:THIS ROUTINE INSTRUCTS THE OPERATOR TO MOUNT A SCRATCH
:TAPE REEL THAT DOES NOT HAVE A WRITE-ENABLE RING INSTALLED, THEN
:WAITS FOR THE OPERATOR TO RESPOND THAT THIS HAS BEEN ACCOMPLISHED.
:UPON THE RESPONSE, THE PROGRAM VERIFIES THAT THE TRANSPORT SHOWS
:A WRITE-PROTECTED STATUS, THEN ATTEMPTS TO WRITE DATA ON THE
:TAPE AND EXPECTS THE APPROPRIATE ERROR TERMINATION INDICATING THAT
:THE WRITE FUNCTION COULD NOT BE PERFORMED BECAUSE THE REEL IS
:WRITE-PROTECTED. IF THE APPROPRIATE TERMINATION IS NOT RECEIVED,
:AN ERROR IS REPORTED.

8643	052716		
	052716		
8644	052716	005037	002170
8645	052722	005037	003100
8650	052726	005737	002162

BGNTST

CLR	FATFLG
CLR	KTFLG
TST	TSTCNT

T6::
:CLEAR FATAL ERROR FLAG
:HOLD OF KT11
:IS THIS THE FIRST TEST

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 115-1
TEST 6: MANUAL INTERVENTION

```

8651 052732 001403                BEQ     21$
8652 052734 012700 054550         MOV     #T38NE,RO
8653 052740 000402                BR
8654 052742                    21$:
8655 052742 012700 055526         MOV     #T38ID,RO
8656 052746 004737 017372         JSR    PC,TSTSETUP
8657 052752 004737 021366         JSR    PC,CHKMAN
8658 052756 103402                BCS    19$
8659 052760 000137 053744         JMP    64$
8660 052764 022737 000001 002162 19$:  CMP     #1,TSTCNT
8661 052772 001402                BEQ     22$
8662 052774 000137 053744         JMP    64$
8663 053000                    22$:
8667 053000 005037 002170         CLR    FATFLG
8668 053004 012737 176750 053756 2$:   MOV     #65000.,T38DLY
8669 053012 004737 016630         JSR    PC,SOFINIT
8670 053016 103427                BCS    23$
8671 053020 010001                MOV     RO,R1
8672 053022 032701 000200         BIT    #SSR,R1
8673 053026 001023                BNE    23$
8674 053030                DELAY  250
                                MOV     #250,(PC)+
                                .WORD  0
                                MOV     L$DLY,(PC)+
                                .WORD  0
                                DEC     -6(PC)
                                BNE    -.4
                                DEC     -22(PC)
                                BNE    .-20
8675 053060 005337 053756         DEC    T38DLY
8676 053064 001352                BNE    5$
8677 053066                ERRDF  ERRNO,SFIERR,SFIMSG
                                :BUMP COUNTER DOWN
                                :BR, IF MORE TIME LEFT
                                :REPORT FATAL ERROR
                                TRAP    CSERDF
                                .WORD  601
                                .WORD  SFIERR
                                .WORD  SFIMSG
                                MOV     #MIMENU,RO
                                MOV     #6,R1
                                JSR    PC,GETSEL
                                MOV     RO,R4
                                ASL    R4
                                JMP    @6$(R4)
8678 053076 012700 055552         :MENU OF MANUAL INTERVENTIONS
8679 053102 012701 000006         :MAXIMUM ALLOWED SELECTION
8680 053106 004737 021144         :GO GET THE OPERATORS SELECTION
8681 053112 010004                 :GET NUMBER FROM ROUTINE
8682 053114 006304                 :CONVERT TO WORD OFFSET
8683 053116 000174 053122         :JUMP TO PROPER LOOP
8684 053122 053000         6$:   .WORD  2$
8685 053124 053132                 :RETYPE THE MENU
8686 053126 053464                 : 3 REWIND AND UNLOAD COMMAND TEST
8687 053130 053744                 : 4 WRITE PROTECT
8688                               : 6 LEAVE THE TEST
8689
8690 053132                20$:   PRINTF #T38MS4
                                :TELL'EM WHAT TO DO
                                MOV     #T38MS4,-(SP)
                                MOV     #1,-(SP)
                                MOV     SP,RO
                                TRAP    CS$PNTF
                                ADD     #4,SP
8691 053132 012746 055144         :DO SOFT INIT OF CONTROLLER
8692 053132 012746 000001         :BR IF SOFT INIT = OK
8693 053142 010600                 :SAVE CONTENTS OF ISSR
8694 053144 104417
8695 053146 062706 000004
8696 053152 004737 016630         JSR    PC,SOFINIT
8697 053156 103405                BCS    300$
8698 053160 010001                MOV     RO,R1

```


CZTUXAO TUBO FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-2

```

8697 053162          ERRDF  ERRNO,SFIERR,SFIMSG      ;DEVICE FATAL ERROR DURING INIT
      053162 104455          TRAP  C$ERDF
      053164 001132          .WORD 602
      053166 003550          .WORD SFIERR
      053170 011656          .WORD SFIMSG

8698 053172          300$:
8699 053172 012704 054470  MOV  #T38PK2,R4      ;SUBROUTINE NEEDS PACKET ADDRESS
8700 053176 004737 010322  JSR  PC,WRTCHR      ;ISSUE WRITE CHARACTERISTICS
8701 053202 103405          BCS  310$          ;BR, IF COMMAND ISSUED OK
8705 053204 010001          MOV  R0,R1          ;SAVE CONTENTS OF TSSR
8706 053206          ERRHRD ERRNO,WRTMSG,SFIMSG      ;WRITE CHARACTERISTIC FAILED
      053206 104456          TRAP  C$ERHRD
      053210 001133          .WORD 603
      053212 004754          .WORD WRTMSG
      053214 011656          .WORD SFIMSG

8707 053216          310$:
8708 053216 012737 000144 053756  MOV  #100.,T38DLY   ;SET UP FOR A LONG WAIT
8709 053224 036527 000000 000100 311$:  BIT  TSSR(R5),#OFL ;IS DRIVE OFF-LINE
8710 053232 001431          BEQ  315$          ;BR, IF DRIVE IS ON-LINE
8711 053234          DELAY 250          ;DELAY ABOUT .25 SEC
      053234 012727 000250          MOV  #250,(PC)+
      053240 000000          .WORD 0
      053242 013727 002116          MOV  L$DLY,(PC)+
      053246 000000          .WORD 0
      053250 005367 177772          DEC  -6(PC)
      053254 001375          BNE  -.4
      053256 005367 177756          DEC  -22(PC)
      053262 001367          BNE  -.20

8712 053264 005337 053756          DEC  T38DLY
8713 053270 001355          BNE  311$
8714 053272          PRINTF #T38OFL      ;BUMP LONG DELAY COUNTER DOWN
      053272 012746 055110          MOV  #T38OFL,-(SP)
      053276 012746 000001          MOV  #1,-(SP)
      053302 010600          MOV  SP,R0
      053304 104417          TRAP C$PNTF
      053306 062706 000004          ADD  #4,SP
8715 053312 000137 053152          JMP  222$          ;STAY HERE FOREVER, WITH MESSAGE
8716 053316          315$:  PRINTF #T38M$5      ;'DRIVE SHOULD NOW REWIND AND GO OFL'
      053316 012746 055223          MOV  #T38M$5,-(SP)
      053322 012746 000001          MOV  #1,-(SP)
      053326 010600          MOV  SP,R0
      053330 104417          TRAP $PNTF
      053332 062706 000004          ADD  #4,SP
8717 053336 012704 054520          MOV  #T38PK3,R4      ;SET UP NEW PACKET FOR REWIND/RELEASE
8718 053342 010465 177776          MOV  R4,T$DB(R5)    ;REWIND RELEASE,ACK,CVC=1 CMD
8719 053346 004737 017104          JSR  PC,WAITF      ;WAIT FOR SSR TO SET
8720 053352 016501 000000          MOV  TSSR(R5),R1   ;GET TSSR STATUS
8721 053356 012702 000300          320$:  MOV  #SSR!OFL,R2   ;SET UP EXPECTED
8722 053362 020201          CMP  R2,R1          ;IS EVERYTHING OK
8723 053364 001404          BEQ  350$          ;BR, IF ALL IS WELL
8727 053366          ERRHRD ERRNO,T38SST,PKTSSR      ;DEVICE FATAL SSR FAILED TO SET
      053366 104456          TRAP  C$ERHRD
      053370 001134          .WORD 604
      053372 055300          .WORD T38SST
      053374 011670          .WORD PKTSSR

8728 053376          350$:  CKLOOP      ;LOOP ON ERROR, IF FLAG SET
      053376 104406          TRAP  C$CLP1

```

CZTUXAO TUBO FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-3

8729	053400	016501	000000	MOV	TSSR(R5),R1	:READ TSSR STATUS	
8730	053404	032701	000100	BIT	#OFL,R1	:CHECK THE OFF-LINE BIT	
8731	053410	001011		BNE	380\$:BR, IF DRIVE IS OFF-LINE	
8732	053412			PRINTF	#T38ONL	: "DRIVE IS NOW ON-LINE"	
	053412	012746	055046				MOV #T38ONL,-(SP)
	053416	012746	000001				MOV #1,-(SP)
	053422	010600					MOV SP,RC
	053424	104417					TRAP C\$PNTF
	053426	062706	000004				ADD #4,SP
8733	053432	000410		BR	390\$:ALMOST DONE	
8734	053434			380\$: PRINTF	#T38OFL	: "DRIVE IS NOW OFF-LINE"	
	053434	012746	055110				MOV #T38OFL,-(SP)
	053440	012746	000001				MOV #1,-(SP)
	053444	010600					MOV SP,RO
	053446	104417					TRAP C\$PNTF
	053450	062706	000004				ADD #4,SP
8735	053454	005037	002172	390\$: CLR	INTRECV	:CLEAR INTERRUPT FLAG	
8736	053460	000137	053000	JMP	2\$:TRY AGAIN	
8737	053464			25\$: GMANIL	T38MSG,T38DAT,-1,NO	:WAIT FOR OPERATOR TO MOUNT TAPE	
	053464	104443					TRAP C\$GMAN
	053466	000404					BR 10000\$
	053470	056024					.WORD T38DAT
	053472	000120					.WORD T\$CODE
	053474	055430					.WORD T38MSG
	053476	177777					.WORD -1
	053500						
8738	053500			BNCOMPLETE	25\$:RETRY IF ERROR	10000\$:
	053500	103371					BCC 25\$
8739	053502	005737	056024	TST	T38DAT	:DID OPERATOR SAY 'YES' ?	
8740	053506	001002		BNE	27\$:BRANCH IF YES	
8741	053510	000137	053000	JMP	2\$:RETURN TO MAIN MENU	
8742	053514			27\$: JSR	PC,SOFINIT	:DO SOFT INIT OF CONTROLLER	
8743	053514	004737	016630	BCS	400\$:BR IF SOFT INIT = OK	
8744	053520	103405		MOV	RO,R1	:SAVE CONTENTS OF TSSR	
8748	053522	010001		ERRDF	ERRNO,SFIERR,SFIMSG	:DEVICE FATAL ERROR DURING INIT	
8749	053524						TRAP C\$ERDF
	053524	104455					.WORD 605
	053526	001135					.WORD SFIERR
	053530	003550					.WORD SFIMSG
	053532	011656					
8750	053534			400\$: CKLOOP		:LOOP IF SELECTED	
	053534	104406					TRAP C\$CLP1
8751	053536	012704	054470	MOV	#T38PK2,R4	:SUBROUTINE NEEDS PACKET ADDRESS	
8752	053542	004737	010322	JSR	PC,WRTCHR	:ISSUE WRITE CHARACTERISTICS	
8753	053546	103405		BCS	410\$:BR, IF COMMAND ISSUED OK	
8757	053550	010001		MOV	RO,R1	:SAVE CONTENTS OF TSSR	
8758	053552			ERRHRD	ERRNO,WRTMSG,SFIMSG	:WRITE CHARACTERISTIC FAILED	
	053552	104456					TRAP C\$ERHRD
	053554	001136					.WORD 606
	053556	004754					.WORD WRTMSG
	053560	011656					.WORD SFIMSG
8759	053562			410\$: CKLOOP		:LOOP IF SELECTED	
	053562	104406					TRAP C\$CLP1
8760	053564	013701	054014	MOV	T38BFR+6,R1	:PICK UP XSTO CONTENTS	
8761	053570	010102		MOV	R1,R2	:SET UP EXPECTED	
8762	053572	052702	000004	BIS	#BIT2,R2	:SET UP THE WRITE LOCKED BIT	
8763	053576	020102		CMP	R1,R2	:ARE THEY CORRECT	

CZTUXAO TU80 FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-4

```

8764 053600 001406          BEQ      430$
8768 053602          ERRHRD  ERRNO,T38WRL,EXPREC
      053602 104456
      053604 001137
      053606 054711
      053610 016330
8769 053612 000137 053000          JMP      2$
8770 053616          430$: CKLOOP
      053616 104406
8771 053620          PRINTF  #T38WOK
      053620 012746 055470
      053624 012746 000001
      053630 010600
      053632 104417
      053634 062706 000004
8772 053640 017737 127226 054542 435$: MOV      @FREE,T38WR
8773 053646 012704 054540          MOV      #T38PK4,R4
8774 053652 010465 177776          MOV      R4,TSDB(R5)
8775 053656 004737 017104          JSR      PC,WAITF
8776 053662 016501 000000          MOV      TSSR(R5),R1
8777 053666 012702 100206          MOV      #SC!SSR!BIT1!BIT2,R2
8778 053672 020102          CMP      R1,R2
8779 053674 001404          BEQ      440$
8783 053676          ERRHRD  ERRNO,T38WRT,PKTSSR
      053676 104456
      053700 001140
      053702 054625
      053704 011670
8784 053706          440$: CKLOOP
      053706 104406
8785 053710 013701 054014          MOV      T38BFR+6,R1
8786 053714 010102          MOV      R1,R2
8787 053716 052702 004000          BIS      #BIT11,R2
8788 053722 020102          CMP      R1,R2
8789 053724 001404          BEQ      450$
8793 053726          ERRHRD  ERRNO,T38WLE,EXPREC
      053726 104456
      053730 001141
      053732 054752
      053734 016330
8794 053736          450$: CKLOOP
      053736 104406
8795 053740 000137 053000          JMP      2$
8796
8797
8798
8799 053744          63$:
8800 053744          64$: EXIT  TST
      053744 104432
      053746 002570
8801
8802
8803
8804
8805
8806
8807

```

```

;BR, IF ALL IS WELL (OK)
;'WRITE LOCKED BIT IS NOT SET ETC.'
      TRAP  CSERHRD
      .WORD 607
      .WORD T38WRL
      .WORD EXPREC
;BECAUSE OF ERROR GO BACK TO MENU
;LOOP IF SELECTED
      TRAP  CSCLP1
;'PRINT 'DRIVE IS WRITE PROTECTED'
      MOV  #T38WOK,-(SP)
      MOV  #1,-(SP)
      MOV  SP,R0
      TRAP CSPTNF
      ADD  #4,SP
;SET UP WRITE BUFFER ADDRESS
;GET PACKET ADDRESS
;SET THE PACKET ADDRESS
;WAIT FOR SSR TO SET
;GET TSSR
;SET UP EXPECTED
;ARE THEY EQUAL (CORRECT)
;BR, IF CORRECT STATUS
;'TSSR INCORRECT AFTER WRITE COMMAND
      TRAP  CSERHRD
      .WORD 608
      .WORD T38WRT
      .WORD PKTSSR
;LOOP ON ERROR, IF FLAG SET
      TRAP  CSCLP1
;READ XSTO CONTENTS
;SET UP EXPECTED
;SET THE WRITE LOCK ERROR BIT (XSTO)
;WAS THE BIT SET
;BR, IF IT WAS (GOOD)
;'WRITE LOCK ERROR BIT NOT SET'
      TRAP  CSERHRD
      .WORD 609
      .WORD T38WLE
      .WORD EXPREC
;LOOP IF SELECTED
      TRAP  CSCLP1
;GO BACK TO MENU
;LEAVE TEST
      TRAP  CSEXIT
      .WORD L10073-.

```

```

;+
;LOCAL TEXT MESSAGES FOR TEST
;-
;LOCAL STORAGE FOR THIS TEST
;-

```

CZTUXAO TU80 FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-5

```

8808          ;+
8809          ;LOCAL STORAGE FOR THIS TEST
8810          ;-
8811
8812 053750 000000  TTION2:      .WORD    0          ;WORD SET IF SUPERVISOR TTI INTER OFF
8813 053752 000000  TVSAV2:   .WORD    0          ;SAVE TTI VECTOR
8814 053754 000000  TPSAV2:   .WORD    0          ;SAVE TTI PRIORITY
8815
8816 053756 000000  T38DLY:   .WORD    0          ;DELAY COUNTER FOR TEST
8818 053760          .BLKB   10-<.-TUV2A&7>
8820 053770          T38PACKET:  ;COMMAND PACKET FOR TEST
8821 053770 140006          .WORD   140006      ;WRITE SUBSYSTEM MEM. CMD. ACK,CVC=1
8822 053772 054000          .WORD   T38TAD      ;ADDRESS OF CHARACTERISTICS BLOCK
8823 053774 000000          .WORD    0
8824 053776 000012          .WORD   10.         ;STARTING VALUE OF BLOCK SIZE
8825 054000          T38TAD:   ;CHARACTERISTICS DATA BLOCK
8826 054000          T38BS0:   .BYTE    0          ;BSEL0 BYTE
8827 054001          T38BS1:   .BYTE    0          ;BSEL1 BYTE
8828 054002 000000          T38BS2:   .WORD    0          ;BSEL1 WORD
8829 054004 000000          .WORD    0          ;DATA
8830 054006          T38BFR:   .BLKW   150.       ;MESSAGE BUFFER
8831 054462 000000          T38EB:   .WORD          ;END OF BUFFER ADDRESS
8832
8833
8835 054464          .BLKB   10-<.-TUV2A&7>
8837 054470          T38PK2:   ;COMMAND PACKET FOR TEST
8838 054470 140004          .WORD   140004      ;WRITE CHARA. MEM. CMND., ACK,CVC=1
8839 054472 054500          .WORD   T38DTA      ;ADDRESS OF SELECT DATA BLOCK
8840 054474 000000          .WORD    0
8841 054476 000012          .WORD   10.         ;STARTING VALUE OF BLOCK SIZE
8842
8843
8844 054500          T38DTA:   ;SELECT DATA BLOCK
8845 054500 054006          .WORD   T38BFR      ;ADDRESS OF MESSAGE BUFFER
8846 054502 000000          .WORD    0
8847 054504 000400          .WORD   256.       ;LENGTH OF MESSAGE BUFFER
8848 054506 000000          T38EAI:   .WORD    0          ;EAI BIT WORD
8850 054510          .BLKB   10-<.-TUV2A&7>
8852 054520 140412          T38PK3:   .WORD   140412  ;REWIND AND UNLOAD COMMAND
8853 054522 000000          .WORD    0          ;NOT USED
8854 054524 000000          .WORD    0          ;NOT USED
8855 054526 000000          .WORD    0          ;NOT USED
8856 054530 000000          .WORD    0          ;NOT USED
8857
8858          ;WRITE TAPE PACKET
8859          ;
8861 054532          .BLKB   10-<.-TUV2A&7>
8863 054540 140005          T38PK4:   .WORD   140005  ;WRITE, ACK, CVC=1 COMMAND
8864 054542 000000          T38WR:   .WORD    0          ;ADDRESS OF WRITE BUFFER
8865 054544 000000          .WORD    0          ;MORE ADDRESS OF WRITE BUFFER
8866 054546 000400          T38SIZ:   .WORD   256.   ;SIZE OF RECORD
8867
8868
8869
8870
8871          ;+
8872          ;LOCAL TEXT MESSAGES FOR TEST

```

CZTUXAO TUBO FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-6

```

8873      :-
8874
8875
8876
8877
8878
8879 054550    123    164    141  T38NE: .ASCIZ  'Stand-alone Manual Intervention Not Executed'
8880 054625    124    123    123  T38WRT: .ASCIZ  'TSSR Not Correct After WRITE, with WRITE PROTECT On'
8881 054711    127    122    111  T38WRL: .ASCIZ  'WRITE LOCKED Bit Not Set In XST0'
8882 054752    127    122    111  T38WLE: .ASCIZ  'WRITE LOCK ERROR Bit Not Set In XST0, After Attempted WRITE'
8883 055046    045    116    045  T38ONL: .ASCIZ  'ZNXA ERROR Drive Is Still ON-LINE'
8884 055110    045    116    045  T38OFL: .ASCIZ  'ZNXA Drive Is Now OFF-LINE'
8885 055144    045    116    045  T38MS4: .ASCIZ  'ZNXA Set Drive To On-line and At Or Beyond BOT'
8886 055223    045    116    045  T38MS5: .ASCIZ  'ZNXA Drive Should Now Rewind and Go Off-line'
8887 055300    103    157    156  T38SST: .ASCIZ  'Contents Of TSSR Incorrect After REWIND And RELEASE'
8888 055364    045    116    045  T38MS2: .ASCIZ  'ZNXA Type RETURN To Return To Menu'
8889 055430    111    163    040  T38MSG: .ASCIZ  'Is Write-Protected Tape Mounted'
8890 055470    045    116    045  T38WOK: .ASCIZ  'ZNXA Drive Is Write Protected'
8891 055526    115    141    156  T38ID:  .ASCIZ  'Manual Intervention'
8892      .EVEN
8893 055552    055570  055642  055670  MIMENU: .WORD    1$,2$,5$,6$
8894 055562    055760  056023  000000  .WORD    9$,10$,0
8895
8896 055570    012    123    105  1$:    .ASCIZ  '<12>'SELECT OPERATION FROM FOLLOWING OPTIONS:'
8897 055642    012    011    060  2$:    .ASCIZ  '<12>' 0 Display This Menu'
8898 055670    011    061    011  5$:    .ASCIZ  ' 1 Rewind and Unload Command Test'
8899 055732    011    062    011  6$:    .ASCIZ  ' 2 Write Protect Test'
8900 055760    011    063    011  9$:    .ASCIZ  ' 3 Return to Diagnostic Supervisor'
8901 056023    000
8902      .EVEN
8903
8904      :+
8905      :LOCAL STORAGE FOR THIS TEST
8906      :-
8907
8908 056024    000000  T38DAT: .WORD    0 ;LOGICAL RESPONSE TO QUESTION
8909 056026  T38REST:
8910 056026  SAVREG ;SAVE THE REGISTERS
8911 056032  012701  053770  MOV #T38PACKET,R1 ;START OF THE PACKET
8912 056036  012721  140206  MOV #140206,(R1)+ ;WRITE SUBSYSTEM MEM. WITH ACK,CVC=1
8913 056042  012721  054000  MOV #T38TAD,(R1)+ ;ADDRESS OF DATA BLOCK
8914 056046  005021  CLR (R1)+ ;EXTENDED ADDRESS
8915 056050  012721  000006  MOV #6,(R1)+ ;SIZE OF DATA BLOCK IN BYTES
8916 056054  005021  CLR (R1)+ ;CLEAR BSEL0 AND BSEL1
8917 056056  005021  CLR (R1)+ ;CLEAR SEL2
8918 056060  005011  CLR (R1) ;CLEAR DATA AREA
8919 056062  000207  RTS PC ;RETURN
8920
8921
8922      :+
8923      :THIS ROUTINE PRINTS THE CONTENTS OF
8924      :THE 256 BYTE MESSAGE BUFFER RETURNED BY THE
8925      :TUV-05.
8926
8927      :INPUT:
8928
8929

```

CZTUXAO TU80 FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-7

```

8930      :      RO      LOW ORDER ADDRESS OF MESSAGE BUFFER
8931      :      R1      HIGH ORDER ADDRESS OF MESSAGE BUFFER
8932      :      NOTE: R1 IS IGNORED IF KTENABLE FLAG IS CLEAR
8933      :
8934      :
8935      :
8936      :
8937 056064      T38MBP:
8938 056064      SAVREG      ;SAVE THE REGISTERS
8939 056070 010005      MOV      R0,R5      ;SAVE LOW ORDER ADDRESS
8940 056072 005737 003102      TST      KTENABLE      ;ADDRESS ABOVE 28K?
8941 056076 001001      BNE      910$      ;BR IF YES
8942 056100 005001      CLR      R1      ;SET HIGH ORDER ADDRESS TO 0
8943 056102 010103      910$: MOV      R1,R3      ;SAVE HIGH ORDER ADDRESS
8944 056104 006100      ROL      R0      ;SHIFT BIT15 TO C BIT
8945 056106 006101      ROL      R1      ;SHIFT TO HIGH ORDER FOR PRINTOUT
8946 056110      PRINTX   #T38AS0,R1,R5 ;PRINT MESSAGE BUFFER ADDRESS
                                MOV      R5,-(SP)
                                MOV      R1,-(SP)
                                MOV      #T38AS0,-(SP)
                                MOV      #3,-(SP)
                                MOV      SP,R0
                                TRAP     C$PNTX
                                ADD      #10,SP
                                MOV      #T38AS1,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP     C$PNTX
                                ADD      #4,SP
                                8947 056134      PRINTX   #T38AS1      ;PRINT HEADER FOR CONTENTS
                                MOV      #T38AS1,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP     C$PNTX
                                ADD      #4,SP
8948 056154 010501      MOV      R5,R1      ;COPY LOW ORDER ADDRESS
8949 056156 010300      MOV      R3,R0      ;COPY HIGH ORDER ADDRESS
8950 056160 001403      BEQ     913$      ;BR IF NOT ABOVE 28K
8951 056162 004737 020252      JSR     PC,SETMAP  ;SETUP PAR ADDRESS IN R0
8952 056166 010005      MOV      R0,R5      ;GET PAR FORMAT ADDRESS ABOVE 28K
8953 056170 010537 056534      913$: MOV      R5,T38CNT ;HOLD ADDRESS
8954 056174 011504      911$: MOV      (R5),R4   ;GET BUFFER ENTRY
8955 056176 022704 125252      CMP     #125252,R4  ;CHECK FOR NO LOAD CONDITION
8956 056202 001417      BEQ     912$      ;BR, IF BUFFER WASN'T LOADED
8957 056204 010403      MOV      R4,R3      ;MAKE COPY
8958 056206 042704 170377      BIC     #170377,R4  ;ONLY BITS 11,10,9 AND 8 ARE SAVED
8959 056212 000241      CLC      ;CLEAR CARRY
8960 056214 006004      ROR     R4      ;11 TO 10 BIT POSITION
8961 056216 006004      ROR     R4      ;10 TO 9 BIT POSITION
8962 056220 006004      ROR     R4      ;9 TO 8 BIT POSITON
8963 056222 006004      ROR     R4      ;8 TO 7 BIT POSITION
8964 056224 042703 177760      BIC     #177760,R3  ;ONLY BITS 3,2,1 AND 0 ARE SAVED
8965 056230 060403      ADD     R4,R3      ;'OR'EM TOGETHER
8966 056232 010325      MOV     R3,(R5)+   ;PUT BACK IN BUFFER
8967 056234 020527 054462      CMP     R5,#T38EB  ;END OF BUFFER YET
8968 056240 001355      BNE     911$      ;BR, IF NOT AT END YET
8969 056242 013705 056534      912$: MOV     T38CNT,R5 ;PUT ADDRESS BACK
8970 056246 012704 000001      MOV     #1,R4      ;START BYTE NUMBER AT ONE
8971 056252      915$: PRINTX   #T38ASN,R4,(R5)+ ;PRT MEM BUFFER W/NEWLINE
                                MOV     (R5)+,-(SP)
                                MOV     R4,-(SP)
                                MOV     #T38ASN,-(SP)
056252 012546
056254 010446
056256 012746 056510

```

CZTUXAO TUBO FRONT END PRT B
TEST 6: MANUAL INTERVENTION

MACRO M1200 29-MAR-83 13:32 PAGE 115-8

```

056262 012746 000003
056266 010600
056270 104415
056272 062706 000010
8972 056276 005037 056534
8973 056302 000412
8974 056304
920$: CLR T38LNT :CLEAR COUNTER
BR 921$ :SKIP OTHER PRINT
PRINTX #T38ASC,R4,(R5)+ :PRINT THE CONTENTS OF MEMORY BUFFER
056304 012546
056306 010446
056310 012746 056471
056314 012746 000003
056320 010600
056322 104415
056324 062706 000010
8975 056330 005237 056534
8976 056334 005204
8977 056336 020427 000200
8978 056342 003010
8979 056344 023727 056534 000004
8980 056352 001401
8981 056354 000753
8982 056356 005037 056534
8983 056362 000733
8984 056364 000207
50$: RTS
8985
8986 056366 045 116 045 T38AS0: .ASCIZ 'ZNZA Message Buffer Address = X01X05'
8987 056433 045 116 045 T38AS1: .ASCIZ 'ZNZA Message Buffer Contents:'
8988 056471 045 101 040 T38ASC: .ASCIZ 'XA XD4XA: X03'
8989 056510 045 116 045 T38ASN: .ASCIZ 'ZNZA ByteXD4XA: X03'
8990
8991 056534 000000 T38CNT: .WORD :COUNTER FOR PRINT
8992 056536
056536
056536 104401
MOV #3,-(SP)
MOV SP,R0
TRAP CSPNTX
ADD #10,SP
MOV (R5)+,-(SP)
MOV R4,-(SP)
MOV #T38ASC,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP CSPNTX
ADD #10,SP
:NUMBER OF THE NEXT
:DONE ALL YET ?
:BRANCH IF ALL DONE
:DONE FOUR YET
:BR, IF THREE DONE
:KEEP GOING
:CLEAR COUNTER
:PRINT WITH NEW LINE
:RETURN
L10073: TRAP CSETST

```

.SBTTL TEST 7: CONFIGURATION TYPEOUT

8994
8995
8996
8997
8998
8999
9000
9001
9002
9003
9004
9005
9006
9007
9008
9009
9010
9011
9012
9013
9014
9015
9016
9017
9018
9019
9020
9021
9022
9023
9024
9025
9026
9027
9032
9033
9034
9035
9036
9037
9038
9039
9040
9041
9042
9043
9044
9045
9046
9050
9051
9052

056540
056540
056540 005037 002170
056544 005037 003100
056550 005737 002162
056554 001403
056556 012700 057773
056562 000402
056564 012700 060432
056570 004737 017372
056574 004737 021366
056600 103402
056602 000137 057202
056606 022737 000001 002162
056614 001402
056616 000137 057202
056622
056622 004737 016630
056626 103405
056630 010001
056632
056632 104455
056634 001275
056636 003550
056640 011656

:THIS IS A STANDALONE ROUTINE THAT PRINTS OUT ON THE CONSOLE TERMINAL
:THE CONFIGURATION OF THE M7454 MODULE AND TUV05 SUBSYSTEM. SPECIFICALLY,
:THE FOLLOWING INFORMATION IS PRESENTED:
:
:1.0 MICROCODE REVISION LEVEL OF THE M7454,
:2.0 NUMBER OF TAPE TRANSPORT . CONNECTED TO THE CONTROLLER,
:3.0 UNIT SELECT CODE AND STATE (ONLINE/OFFLINE, WRITE ENABLED/PROTECTED)
:OF EACH CONNECTED TRANSPORT.
:
:THE OPERATOR IS EXPECTED TO READ THE PRINTOUT AND VERIFY THAT IT MATCHES
:THE ACTUAL CONFIGURATION AT HAND. IF, FOR EXAMPLE, THE PROGRAM INDICATES
:THAT IT "SEES" TWO TRANSPORTS CONNECTED WHEN IN FACT ONLY ONE IS PRESENT,
:THE OPERATOR MUST INTERPRET THIS AS AN ERROR AND ATTEMPT TO FIND THE
:CAUSE (BAD CABLE, FAULTY UNIT-SELECT DECODING IN THE TRANSPORT, ETC.).
:[SINCE THE CONTROLLER CAN ONLY ACCESS UNIT 0 IF IT IS IN "STANDARD"
:MODE, THE PROGRAM WILL FORCE THE MODULE INTO EXTENDED MODE VIA THE
:WRITE SUBSYSTEM MEMORY COMMAND IN ORDER TO SCAN FOR CONNECTED TRANSPORTS.]
:
:THIS ROUTINE, WHEN ITS ACTIONS ARE COMPLETED, WILL EXIT BACK TO THE
:DIAGNOSTIC SUPERVISOR SO THAT IF ADDITIONAL UNITS (CONTROLLERS) ARE
:SELECTED (E.G., FROM THE INITIAL STARTUP DIALOG), THE ROUTINE WILL BE
:REENTERED SO THAT THEIR CONFIGURATIONS CAN BE PRINTED.

BGNTST

CLR FATFLG
CLR KTFLG
TST TSTCNT
BEQ 10\$
MOV #T39NE,R0
BR 11\$
10\$: MOV #TST39ID,R0
11\$: JSR PC,TSTSETUP
JSR PC,CHKMAN
BCS 15\$
JMP 64\$
15\$: CMP #1,TSTCNT
BEQ 20\$
JMP 64\$
20\$: JSR PC,SOFINIT
BCS 25\$
MOV R0,R1
ERRDF ERRNO,SFIERR,SFIMSG
25\$: CKLOOP

T7: :
:CLEAR FATAL ERROR FLAG
:HOLD OF KT11
:IS THIS FIRST TEST IN SEQUENCE ?
:BR, IF FIRST TEST
:"TEST NOT EXECUTED"
:JUMP OUT OF TEST IF NOT
:TEST ID MESSAGE
:DO THE COMMON SETUP
:IS MANUAL INTERVENTION ALLOWED?
:BR, IF MANUAL INTERVENTION ALLOWED
:JUMP TO OUT IF NOT
:IS THIS THE FIRST TEST IN SEQ
:BR, IF FIRST TEST
:JMP IF IT WASN'T
:DO SOFT INIT OF CONTROLLER
:BR IF SOFT INIT = OK
:SAVE CONTENTS OF TSSR
:DEVICE FATAL ERROR DURING INIT
TRAP C\$ERDF
.WORD 701
.WORD SFIERR
.WORD SFIMSG
:LOOP IF SELECTED

CZTUXAO TUBO FRONT END PRT B
TEST 7: CONFIGURATION TYPEOUT

MACRO M1200 29-MAR-83 13:32 PAGE 116-1

```

9053 056642 104406
9054 056644 012704 057720
9055 056650 004737 010322
9059 056654 103405
9060 056656 010001
          056660 104456
          056662 001276
          056664 004754
          056666 011656
9061 056670          50$: CKLOOP
          056670 104406
9062 056672 013701 057242
9063 056676 042701 177700
9064 056702 010137 060370
9065 056706
          056706 013746 060370
          056712 012746 060313
          056716 012746 000002
          056722 010600
          056724 104415
          056726 062706 000006
9066 056732 004737 016630
9067 056736 103405
9071 056740 010001
9072 056742
          056742 104455
          056744 001277
          056746 003550
          056750 011656
9073 056752          140$: CKLOOP
          056752 104406
9074 056754 012704 057720
9075 056760 004737 010322
9076 056764 103405
9080 056766 010001
9081 056770
          056770 104456
          056772 001300
          056774 004754
          056776 011656
9082 057000          150$: CKLOOP
9083 057002 005037 002150
9084
9085 057006 016501 000000
9086 057012 032701 000100
9087 057016 001414
9088 057020
          057020 013746 002150
          057024 012746 060052
          057030 012746 000002
          057034 010600
          057036 104415
          057040 062706 000006
9089 057044 000137 057162
9090 057050          200$: JMP 63$
          PRINTX #T390N2,UNITN

```

TRAP C\$CLP1
:SUBROUTINE NEEDS PACKET ADDRESS
:ISSUE WRITE CHARACTERISTICS
:BR, IF COMMAND ISSUED OK
:SAVE CONTENTS OF TSSR
:WRITE CHARACTERISTICS FAILED
TRAP C\$ERHRD
.WORD 702
.WORD WRTMSG
.WORD SFIMSG
:LOOP IF SELECTED
TRAP C\$CLP1
:PICK UP THE RES. BYTE CNTR AREA
:ONLY LEAVE MICROCODE REV LEVEL
:LOAD UP REV LEVEL
:'MICROCODE REVISION LEVEL =000xxx'
MOV T39RL,-(SP)
MOV #T39MCL,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTX
ADD #6,SP
:DO SOFT INIT OF CONTROLLER
:BR IF SOFT INIT = OK
:SAVE CONTENTS OF TSSR
:DEVICE FATAL ERROR DURING INIT
TRAP C\$ERDF
.WORD 703
.WORD SFIERR
.WORD SFIMSG
:LOOP IF SELECTED
TRAP C\$CLP1
:SUBROUTINE NEEDS PACKET ADDRESS
:ISSUE WRITE CHARACTERISTICS
:BR, IF COMMAND ISSUED OK
:SAVE CONTENTS OF TSSR
:WRITE CHARACTERISTICS FAILED
TRAP C\$ERHRD
.WORD 704
.WORD WRTMSG
.WORD SFIMSG
:LOOP IF SELECTED
TRAP C\$CLP1
:SET TO DRIVE 0
:GET TSSR STATUS
:CHECK FOR OFF-LINE
:BR, IF DRIVE IS ON-LINE
:'DRIVE NUMBER XX IS OFF-LINE'
MOV UNITN,-(SP)
MOV #T390F2,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTX
ADD #6,SP
:DO NOT TRY TO GET ANYMORE INFO.
:'DRIVE NUMBER XX IS ON-LINE'

CZTUXAO TU80 FRONT END PRT B
TEST 7: CONFIGURATION TYPEOUT

MACRO M1200 29-MAR-83 13:32 PAGE 116-2

057050	013746	002150				MOV	UNITN,-(SP)
057054	012746	060116				MOV	#T39ON2,-(SP)
057060	012746	000002				MOV	#2,-(SP)
057064	010600					MOV	SP,RO
057066	104415					TRAP	CSPNTX
057070	062706	000006				ADD	#6,SP
9091	057074	013701	057244	MOV	T39BFR+6,R1		:READ EXTENDED STATUS (XSTO)
9092	057100	032701	000004	BIT	#BIT2,R1		:IS DRIVE WRITE PROTECTED
9093	057104	001013		BNE	210\$:BR, IF WRITE PROTECTED
9094	057106			PRINTX	#T39WPN,UNITN		: "DRIVE NUMBER IS NOT WRT PRO"
	057106	013746	002150			MOV	UNITN,-(SP)
	057112	012746	060234			MOV	#T39WPN,-(SP)
	057116	012746	000002			MOV	#2,-(SP)
	057122	010600				MOV	SP,RO
	057124	104415				TRAP	CSPNTX
	057126	062706	000006			ADD	#6,SP
9095	057132	000413					
9096	057134			BR	63\$:SKIP OVER
	057134	013746	002150	PRINTX	#T39WRT,UNITN		: "DRIVE NUMBER XX IS WRT PRO"
	057140	012746	060161			MOV	UNITN,-(SP)
	057144	012746	000002			MOV	#T39WRT,-(SP)
	057150	010600				MOV	#2,-(SP)
	057152	104415				MOV	SP,RO
	057154	062706	000006			TRAP	CSPNTX
9097	057160	000400				ADD	#6,SP
9098	057162			BR	63\$:BR, IF NO MORE DRIVES
	057162	012746	057770	PRINTX	#T39NFL		:NEW LINE
	057166	012746	000001			MOV	#T39NFL,-(SP)
	057172	010600				MOV	#1,-(SP)
	057174	104415				MOV	SP,RO
	057176	062706	000004			TRAP	CSPNTX
9099	057202			EXIT	TST		#4,SP
	057202	104432					:EXIT THIS SECTION
	057204	001254				TRAP	CSEXIT
						.WORD	L10074-
9100				:+			
9101				:LOCAL TEXT MESSAGES FOR TEST			
9102				:-			
9103							
9104				:LOCAL STORAGE FOR THIS TEST			
9105				:-			
9106							
9107	057206	000000		T39DLY:	.WORD 0		:DELAY COUNTER FOR TEST
9109	057210				.BLKB 10-<.-TUV2A&7>		
9111	057220			T39PACKET:			:COMMAND PACKET FOR TEST
9112	057220	140006			.WORD 140006		:WRITE SUBSYSTEM MEM. CMD, ACK,CVC=1
9113	057222	057230			.WORD T39TAD		:ADDRESS OF CHARACTERISTICS BLOCK
9114	057224	000000			.WORD 0		
9115	057226	000012			.WORD 10.		:STARTING VALUE OF BLOCK SIZE
9116	057230			T39TAD:			:CHARACTERISTICS DATA BLOCK
9117	057230	000		T39BS0:	.BYTE 0		:BSEL0 BYTE
9118	057231	000		T39BS1:	.BYTE 0		:BSEL1 BYTE
9119	057232	000000		T39BS2:	.WORD 0		:BSEL1 WORD
9120	057234	000000			.WORD 0		:DATA
9121	057236			T39BFR:	.BLKW 150.		:MESSAGE BUFFER
9122							
9123							
9125	057712				.BLKB 10-<.-TUV2A&7>		

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 117-1
 TEST 8: SCOPE LOOPS

060556	001441											.WORD	801
060560	003550											.WORD	SFIERR
060562	011656											.WORD	SFIMSG
9265	060564	012700	061170	5\$:	MOV	#SCMENU,R0							
9266	060570	012701	000010		MOV	#8,R1							
9267	060574	004737	021144		JSR	PC,GETSEL							
9268	060600	005700			TST	R0							
9269	060602	001760			BEQ	2\$							
9270	060604	010004		3\$:	MOV	R0,R4							
9271	060606				SETPRI	#PRI07							
	060606	012700	000340									MOV	#PRI07,R0
	060612	104441										TRAP	CSSPRI
9272	060614	005037	061162		CLR	TTION							
9273	060620	032737	000100	177560	BIT	#100,#TTICSR							
9274	060626	001005			BNE	4\$							
9275	060630	005237	061162		INC	TTION							
9276	060634	052737	000100	177560	BIS	#100,#TTICSR							
9277	060642	012701	000060	4\$:	MOV	#TTIVEC,R1							
9278	060646	011137	061164		MOV	(R1),TVECSAV							
9279	060652	012721	061070		MOV	#60,(R1)+							
9280	060656	011137	061166		MOV	(R1),TPRISAV							
9281	060662	012711	000340		MOV	#PRI07,(R1)							
9282	060666				SETPRI	#PRI00							
	060666	012700	000000									MOV	#PRI00,R0
	060672	104441										TRAP	CSSPRI
9283	060674	006304			ASL	R4							
9284	060676	000174	060702		JMP	#6\$(R4)							
9285	060702	060544		6\$:	.WORD	2\$							
9286	060704	060722			.WORD	10\$							
9287	060706	060730			.WORD	15\$							
9288	060710	060736			.WORD	20\$							
9289	060712	060752			.WORD	25\$							
9290	060714	060766			.WORD	30\$							
9291	060716	061002			.WORD	35\$							
9292	060720	061156			.WORD	65\$							
9293													
9294													
9295	060722			10\$:									
9296	060722	016500	177776	12\$:	MOV	TSBA(R5),R0							
9297	060726	000775			BR	12\$							
9298													
9299	060730			15\$:									
9300	060730	016500	000000	18\$:	MOV	TSSR(R5),R0							
9301	060734	000775			BR	18\$							
9302													
9303													
9304	060736	004737	021062	20\$:	JSR	PC,GETPAT							
9305	060742	010001			MOV	R0,R1							
9306	060744	010165	000000	22\$:	MOV	R1,TSSR(R5)							
9307	060750	000775			BR	22\$							
9308													
9309													
9310	060752			25\$:									
9311	060752	004737	021062		JSR	PC,GETPAT							
9312	060756	010001			MOV	R0,R1							
9313	060760	110165	177777	27\$:	MOVB	R1,TSDBH(R5)							
9314	060764	000775			BR	27\$							

:MENU OF SCOPE LOOP SELECTIONS
 :MAXIMUM ALLOWED SELECTION
 :GO GET THE OPERATORS SELECTION
 :WAS ZERO SPECIFIED ?
 :REPEAT MENU IF YES.
 :SAVE THE MENU SELECTION
 :RAISE THE PRIORITY

:ASSUME INTERRUPTS ARE ENABLED
 :ARE TTI INTERRUPTS ON ?
 :BRANCH IF YES
 :FLAG SET IF INTERRUPTS OFF
 :ENABLE INTERRUPTS
 :START OF TTI VECTORS
 :SAVE THE CURRENT TTI VECTOR
 :SET NEW INTERRUPT ROUTINE
 :SAVE THE VECTOR PRIORITY
 :USE PRIORITY SEVEN
 :LOWER INTERRUPT BR LEVEL

:CONVERT TO WORD OFFSET
 :JUMP TO PROPER LOOP
 :RETYPE THE MENU
 :TSBA READ ACCESS
 :TSSR READ ACCESS
 :TSSR WRITE ACCESS
 :TSDB HIGH BYTE WRITE ACCESS
 :TSDB LOW BYTE WRITE ACCESS
 :TSSR BYTE WRITE (SELF-TEST)
 :LEAVE THE TEST

:READ TSBA REGISTER
 :LOOP UNTIL HALTED

:READ TSSR REGISTER
 :LOOP UNTIL STOPPED

:READ THE DATA PATTERN
 :DATA PATTERN FOR LOOP
 :WRITE DATA TO TSSR
 :LOOP

:READ THE DATA PATTERN
 :DATA PATTERN FOR LOOP
 :WRITE THE DATA TO TSDB, HIGH BYTE
 :LOOP UNTIL STOPPED

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 117-2
 TEST 8: SCOPE LOOPS

```

9315
9316
9317 060766          30$:      JSR      PC,GETPAT      ;READ THE DATA PATTERN
9318 060766 004737 021062      MOV      RO,R1      ;DATA PATTERN FOR LOOP
9319 060772 010001          32$:      MOVB     R1,TSSB(R5)  ;WRITE DATA TO TSSR, LOW BYTE
9320 060774 110165 177776      BR       32$        ;LOOP UNTIL HALTED BY OPERATOR
9321 061000 000775
9322
9323 061002 004737 021062      35$:      JSR      PC,GETPAT      ;PICK UP THE PATTERN FROM OPR
9324 061006 010001          MOV      RO,R1      ;STORE IN R1
9325 061010 110165 000000      37$:      MOVB     R1,TSSR(R5)  ;WRITE BYTE TO TSSR, THIS STARTS MDIAG
9326 061014          DELAY     250        ;WAIT 2500US
          MOV      #250,(PC)+
          .WORD    0
          MOV      LSDLY,(PC)+
          .WORD    0
          DEC     -6(PC)
          BNE     -4
          DEC     -22(PC)
          BNE     -20
9327 061044 036527 000000 000200      BIT      TSSR(R5),#SSR ;CHECK FOR READY SET, IF A TRAP OCCURS
9328                                     ;THE TSSR WAS 'NOT BACK TO THE BUS'
9329                                     ;IN TIME.
9330 061052 001356          BNE      37$        ;BR, IF SSR WAS SET (GOOD) KEEP LOOPING
9331 061054          ERRDF     ERRNO,T40NSR,SFIMSG ;PRINT DEVICE FATAL ERROR MESSAGE
          TRAP     C$ERDF
          .WORD    802
          .WORD    T40NSR
          .WORD    SFIMSG
9332 061054 104455
9333 061056 001442
9334 061060 061754
9335 061062 011656
9336 061064 000137 000200      JMP      200        ;GO TO SUPERVISOR ETC.
9337
9338      ;+
9339      ;PROCESS CONSOLE INTERRUPTS
9340      ;-
9341
9342 061070 010046          60$:      MOV      RO,-(SP)    ;SAVE WORK REGISTER
9343 061072 113700 177562      MOVB     @#TTIBFR,RO ;GET THE OPERATOR INPUT
9344 061076 042700 000200      BIC      #200,RO    ;STRIP OFF PARITY BIT
9345 061102 122700 000015      CMPB     #15,RO    ;IS IT A CARRIAGE RETURN ?
9346 061106 001021          BNE      61$        ;JUST EXIT IF NOT
9347 061110 012766 060544 000002      MOV      #2$,2(SP) ;RETURN TO MASTER MENU
9348 061116 005066 000004      CLR      4(SP)     ;FORCE PRIORITY ZERO
9349 061122 013737 061164 000060      MOV      TVECSAV,@#TTIVEC ;RESTORE SUPERVISOR VECTOR
9350 061130 013737 061166 000062      MOV      TPRISAV,@#TTIVEC+2 ;RESTORE SUPERVISOR PRIORITY
9351 061136 005737 061162          TST      TTION     ;ARE SUPERVISOR INTERRUPTS ENABLED ?
9352 061142 001403          BEQ      61$        ;BRANCH IF YES
9353 061144 042737 000100 177560      BIC      #100,@#TTICSR ;TURN OFF TTI INTERRUPTS
9354 061152 012600          61$:      MOV      (SP)+,RO  ;RESTORE REGISTER
9355 061154 000002          RTI             ;RETURN FROM INTERRUPT
9356
9357 061156          64$:
9358 061156          63$:
9359 061156          65$:      EXIT     TST      ;EXIT THE TEST
          TRAP     C$EXIT
          .WORD    L10075-.
          061156 104432
          061160 000664
9360
    
```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 117-3
TEST 8: SCOPE LOOPS

```

9361
9362          ;+
9363          ;LOCAL STORAGE FOR THIS TEST
9364          ;-
9365 061162 000000  TTION:          .WORD 0          ;WORD SET IF SUPERVISOR TTI INTER OFF
9366 061164 000000  TVECSAV:         .WORD 0          ;SAVE TTI VECTOR
9367 061166 000000  TPRISAV:         .WORD 0          ;SAVE TTI PRIORITY
9368
9369

```

```

9370          ;+
9371          ;MENU FOR OPERATOR INPUT FOR SCOPE LOOPS
9372          ;-
9373

```

```

9374          .EVEN
9375 061170 061220 061273 061321  SCMENU: .WORD 1$,2$,3$,4$,5$,6$
9376 061204 061472 061530 061567  .WORD 7$,8$,10$,11$,12$,0
9377
9378
9379 061220 012 123 105 1$: .ASCIZ <12>'SELECT SCOPE LOOP FROM FOLLOWING OPTIONS:'
9380 061273 012 011 060 2$: .ASCIZ <12>' 0 Display This Menu'
9381 061321 011 061 011 3$: .ASCIZ ' 1 TSBA Read Access'
9382 061345 011 062 011 4$: .ASCIZ ' 2 TSSR Read Access'
9383 061371 011 063 011 5$: .ASCIZ ' 3 Initialize (TSSR Write Access)'
9384 061433 011 064 011 6$: .ASCIZ ' 4 TSDB High Byte Write Access'
9385 061472 011 065 011 7$: .ASCIZ ' 5 TSDB Low Byte Write Access'
9386 061530 011 066 011 8$: .ASCIZ ' 6 TSSR Write Byte (Self Test)'
9387 061567 011 067 011 10$: .ASCIZ ' 7 Return to Diagnostic Supervisor'
9388 061632 000 11$: .ASCIZ ''
9389 061633 124 171 160 12$: .ASCIZ 'Type RETURN To Stop Scope Loops'
9390 061673 123 164 141 T4ONE: .ASCIZ 'Stand-alone Scope Loops Not Executed'
9391 061740 123 143 157 TST40ID: .ASCIZ 'Scope Loops'
9392 061754 123 123 122 T4ONSR: .ASCIZ 'SSR Failed To Set After TSSR Write Byte And 10ms Delay'
9393          .EVEN
9394 062044          ENDTST
          062044
          062044 104401

```

L10075: TRAP CSETST

CZTUXAO TUBO FRONT END PRT B
DISPLAY BREAKPOINT SETTINGS

MACRO M1200 29-MAR-83 13:32 PAGE 140

10469
10474
10480
10481
10482
10483
10484
10485
10486
10487
10488
10489
10490
10491
10492
10493 066644
066644 000015
066646

10494
10495 066646
066646 000031
066650 066700
066652 160000
066654 177776

10496 066656
066656 001031
066660 066727
066662 000000
066664 000776

10497 066666
066666 002032
066670 066753
066672 000340
066674 000000
066676 000007

10498 066700

.SBTTL HARDWARE PARAMETER CODING SECTION

```

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

```

```

BGNHRD
.WORD L10076-LSHARD/2
LSHARD::

```

```

GPRMA HPM1,0,0,160000,177776,YES ;GET TSBA/TSDB REGISTER ADDRESS.
.WORD TSCODE
.WORD HPM1
.WORD TSLOLIM
.WORD TSHILIM
GPRMA HPM2,2,0,0,776,YES ;GET VECTOR ADDRESS.
.WORD TSCODE
.WORD HPM2
.WORD TSLOLIM
.WORD TSHILIM
GPRMD HPM3,4,0,340,0,7,YES ;GET INTERRUPT PRIORITY.
.WORD TSCODE
.WORD HPM3
.WORD 340
.WORD TSLOLIM
.WORD TSHILIM
ENDHRD
.EVEN

```

```

066700
10499 066700 104 105 126 HPM1: .ASCIZ 'DEVICE ADDRESS (TSSR) '
10500 066727 111 116 124 HPM2: .ASCIZ 'INTERRUPT VECTOR '
10501 066753 111 116 124 HPM3: .ASCIZ 'INTERRUPT PRIORITY '
10502 .EVEN
10503

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141
SOFTWARE PARAMETER CODING SECTION

.SBTTL SOFTWARE PARAMETER CODING SECTION

10505
10506
10507
10508
10509
10510
10511
10512
10513
10514

```

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

```

10515 067004
067004 000006
067006
10516 067006
067006 000130
067010 067022
067012 177777
10517 067014
067014 001130
067016 067061
067020 177777

```

BGNSFT
.WORD L10077-LSSOFT/2
LSSOFT::
GPRML SPM1,0,-1,YES ;GET RAM DUMP FLAG
.WORD TSCODE
.WORD SPM1
.WORD -1
GPRML SPM4,2,-1,YES ; GET ITERATION CONTROL.
.WORD TSCODE
.WORD SPM4
.WORD -1
GPRMD SPM6,4,D,7777,0,7777,YES ; GET LOCAL ERROR LIMIT
GPRMD SPM7,6,D,7777,0,7777,YES ; GET GLOBAL ERROR LIMIT
ENDSFT
.EVEN

```

10518
10519
10520 067022
067022

L10077:

10521
10522
10523 067022 105 116 101
10524 067061 111 116 110
10525 067111 120 105 122
10526 067141 120 105 122

```

SPM1: .ASCIZ 'ENABLE M7454 RAM DUMP ON ERROR'
SPM4: .ASCIZ 'INHIBIT ITERATIONS'
SPM6: .ASCIZ 'PER TEST ERROR LIMIT'
SPM7: .ASCIZ 'PER UNIT ERROR LIMIT'
.EVEN

```

10527
10528
10529
10530
10531
10532
10533
10534

```

.SBTTL PATCH AREA
:++
:DISPATCH TABLE
:
: *** MOVE TO FRONT OF PROGRAM FOR RELEASE ***
:--

```

10535 067172
067172 000010
067174
067174 023630
067176 033132
067200 034164
067202 035516
067204 041450
067206 052716
067210 056540
067212 060462

```

DISPATCH TESTNO
.WORD 8
LSDISPATCH::
.WORD T1
.WORD T2
.WORD T3
.WORD T4
.WORD T5
.WORD T6
.WORD T7
.WORD T8

```

10536
10537
10538
10539
10540
10541

```

: FINALLY A GENEROUS PATCH AREA.
:
: AND AN ADJUSTMENT TO ACCOUNT FOR THE "LASTAD BIT7" HACK
: DESCRIBED IN "SUPPRG.MEM" (FOR REV C).

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-1
PATCH AREA

```

10542
10543
10544 067214
10545
10546
10547
10548
10549 067214
      067214 067232
      067216 000005
      067220
10550
10551
10552
10553
10554 067220
10555 067220
      067220 000000
      067222 000003
      067224
10556 067224 172522
10557 067226 000224
10558 067230 000240
10559 067232
      067232
10560 067232
10561
10562      000001

```

```

:
PATCH::
      .EVEN
      .IF      NZ,..2377
      .=.!377+1
      .ENDC
LASTAD      ;SET LAST USED ADDRESS.
      .EVEN
      .WORD T$FREE
      .WORD T$SIZE
L$LAST::
      .SBTTL  HARD CODED P-TABLE
:++
      DIAGNOSTIC IS PRE-PARAMETERIZED PER THIS TABLE
:--
      BGNSETUP      1
      BGNPTAB
      .WORD 0
      .WORD  L10102-./2-1
L10100:
      .WORD      172522
      .WORD      224
      .WORD      PRI05
      ENDPTAB
L10102:
      ENDSETUP
      .END

```

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-2
SYMBOL TABLE

ADSSR	011762	G	CSAU	=	000052	DEBUGM	011454	FATFLG	002170	G	HIMEM	=	007776		
ADR	=	000020	CSAUTO	=	000061	DEVCNT	002166	FERCM	011544	G	HOE	=	100000	G	
AMBTSS	006325		CSBRK	=	000022	DEVDR0	023560	FIFEXP	012012	G	HPM1	066700			
ASSEMB	=	000010	CSBSEG	=	000004	DEVNRD	023477	FIF1MS	012064		HPM2	066727			
A1716	=	000003	CSBSUB	=	000002	DEVNXR	023415	FIF2MS	012133		HPM3	066753			
BADDAT	003110	G	CSCEFG	=	000045	DEVONL	023345	FILLME	020356		IBE	=	010000	G	
BADSSR	016534	G	CSCLCK	=	000062	DEVSUM	023310	FNOINT	004113		IDU	=	000040	G	
BAR	=	174402	CSCLEA	=	000012	DFPTBL	002124	FORCER	002144	G	IER	=	020000	G	
BENBSW	002174	G	CSCL0S	=	000035	DIAGMC	=	000000	FREE	003072	G	IFAUlT	004154		
BIE	=	040000	CSCLP1	=	000006	DLCYL	=	000177	FREEHI	003076	G	INCERK	017726		
BIT0	=	000001	CSVEEC	=	000036	DLNER	=	100200	FRESIZ	003074	G	INTCPC	017004		
BIT00	=	000001	CSDCLN	=	000044	DLERR	=	177730	FUSI	004015		INTFLA	017001		
BIT01	=	000002	CSDODU	=	000051	DLGETS	=	000004	FSAU	=	000015	INTMAS	017000		
BIT02	=	000004	CSDRPT	=	000024	DLRDHD	=	000010	FSAUTO	=	000020	INTR	017052	G	
BIT03	=	000010	CSDU	=	000053	DLRDNH	=	000016	FSBGN	=	000040	INTREC	002172	G	
BIT04	=	000020	CSEDIT	=	000003	DLSR	=	000013	FSCLEA	=	000007	INTVEC	017002		
BIT05	=	000040	CSERDF	=	000055	DLUN	=	000006	FSDU	=	000016	INTX	004176		
BIT06	=	000100	CSERHR	=	000056	DSBINT	017040	FSEND	=	000041	I0KCKI	=	000200		
BIT07	=	000200	CSERRO	=	000060	DUAD12	004541	FSHARD	=	000004	I0KSTP	=	000001		
BIT08	=	000400	CSERSF	=	000054	DUFLG	003060	FSHW	=	000013	IPRI	002160	G		
BIT09	=	001000	CSERSO	=	000057	DUMMY	003030	FSINIT	=	000006	ISR	=	000100	G	
BIT1	=	000002	CSESCA	=	000010	EF.CON	=	000036	FSJMP	=	000050	IVEC	002156	G	
BIT10	=	002000	CSSEEG	=	000005	EF.NEW	=	000035	FSMOD	=	000000	IXE	=	004000	G
BIT11	=	004000	CSSESUB	=	000003	EF.PWR	=	000034	FSMSG	=	000011	ISAU	=	000041	
BIT12	=	010000	CSSETST	=	000001	EF.RES	=	000037	FSPROT	=	000021	ISAUTO	=	000041	
BIT13	=	020000	CS\$EXIT	=	000032	EF.STA	=	000040	FSPWR	=	000017	ISCLN	=	000041	
BIT14	=	040000	CSGETB	=	000026	EMAXDU	017661	F\$RPT	=	000012	ISDU	=	000041		
BIT15	=	100000	CSGETW	=	000027	EN	=	000000	FSSEG	=	000003	ISHRD	=	000041	
BIT2	=	000004	CSGMAN	=	000043	ENAINI	017006	FSSOFT	=	000005	ISINIT	=	000041		
BIT3	=	000010	CSGPHR	=	000042	ENVIRN	021516	FSSRV	=	000010	ISMOD	=	000040		
BIT4	=	000020	CSGPLO	=	000030	EPRTSW	002146	FSSUB	=	000002	ISMSG	=	000041		
BIT5	=	000040	CSGPRI	=	000040	EPRT1	005755	FSSW	=	000014	ISPROT	=	000040		
BIT6	=	000100	CSINIT	=	000011	EPRT2	005672	FSTEST	=	000001	ISPTAB	=	000041		
BIT7	=	000200	CSINLP	=	000020	EPRT3	006014	GDDAT	003112	G	ISPWR	=	000041		
BIT8	=	000400	CSMANI	=	000050	ERRCM	011555	GERRMA	002142	G	ISRPT	=	000041		
BIT9	=	001000	CSMEM	=	000031	ERRHI	002202	GETPAT	021062	G	ISSEG	=	000041		
BOE	=	000400	CSMSG	=	000023	ERRK	017640	GETSEL	021144	G	ISSETU	=	000041		
BRINIT	004355		CSOPEN	=	000034	ERRLO	002204	GSCNTO	=	000200	ISSFT	=	000041		
BSELO	=	000000	CSPNTB	=	000014	ERRNO	=	001442	GSDELM	=	000372	ISSRV	=	000041	
BSEL1	=	000001	CSPNTF	=	000017	ERRVEC	=	000004	GSDISP	=	000003	ISSUB	=	000041	
CHKAMB	016700		CSPNTS	=	000016	ERTABE	003330	GSEXCP	=	000400	ISTST	=	000041		
CHKMAN	021366	G	CSPNTX	=	000015	ERTABL	003130	GSHILI	=	000002	JSJMP	=	000167		
CHKTSS	017220		CSQID	=	000377	ESUM	017642	GSLOLI	=	000001	KIPAR0	=	172340		
CKDROP	020136		CSRDBU	=	000007	EVL	=	000004	GSNO	=	000000	KIPAR1	=	172342	
CKEMAX	017764		CSREFG	=	000047	EXBCNT	=	000010	GSOFFS	=	000400	KIPAR2	=	172344	
CKMSG	011202	G	CSRESE	=	000033	EXPBRE	016336	GSOFSI	=	000376	KIPAR3	=	172346		
CKMSG2	011322	G	CSREVI	=	000003	EXPD	002176	GSPRMA	=	000001	KIPAR4	=	172350		
CKRAM	010524	G	CSRFLA	=	000021	EXPGOT	004431	GSPRMD	=	000002	KIPAR5	=	172352		
CKRAM2	011100	G	CSRPT	=	000025	EXPGT2	004465	GSPRML	=	000000	KIPAR6	=	172354		
CMPIEM	020542		CSSEFG	=	000046	EXPMMSG	002266	GSRADA	=	000140	KIPAR7	=	172356		
CONFIG	020204		CS\$PRI	=	000041	EXPREC	016330	GSRADB	=	000000	KIPDR0	=	172300		
COUNT	002254	G	CS\$VEC	=	000037	EXTA	005232	GSRADD	=	000040	KIPDR1	=	172302		
CSR	=	174400	CSTPRI	=	000013	EXTEND	005230	GSRADL	=	000120	KIPDR2	=	172304		
CSRADD	002154	G	DAR	=	174404	ESEND	=	002100	GSRADO	=	000020	KIPDR3	=	172306	
CTAB	003116	G	DATA	002256	G	ESLOAD	=	000035	GSXFER	=	000004	KIPDR4	=	172310	
CTABE	003130	G	DATAFL	015047		FATCHK	020064	GSYES	=	000010	KIPDR5	=	172312		
CTABM	003116	G	DATASC	021120		FATERR	=	000060	HIADDR	=	001400	KIPDR6	=	172314	

CZTUXAO TU80 FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-3
 SYMBOL TABLE

KIPDR7= 172316	LSSOFT 067006 G	L10061 043530	NXRERR 005176 G	O.G02 063750
KTENAB 003102 G	LSSPC 002056 G	L10062 044166	NXRX 003675	O.HIGH 066546
KTFLG 003100 G	LSSPCP 002020 G	L10063 044614	NXTU 022210	O.LG = 000010
KTINIT 021604	LSSPTP 002024 G	L10064 045242	OFL = 000100	O.LGCH 066141
KTOFF 020230	LSSSTA 002030 G	L10065 045624	ONEFIL= 000001	O.LGDR 062536
KTON 020212	LSSW 002134 G	L10066 046300	OSAPTS= 000000	O.LOW 066544
LERRMA 002140 G	LSTEST 002114 G	L10067 046542	OSAU = 000001	O.MOVE 064250
LISTAL= 000001	LSTIML 002014 G	L10070 047034	OSBGNR= 000001	O.MSK 066542
LOE = 040000 G	LSUNIT 002012 G	L10071 047324	OSBGNS= 000001	O.ODT 062046 G
LOOPCN 002164 G	L10000 002132	L10072 050000	OSDU = 000001	O.OFST 063366
LOOPCO 012750	L10001 002144	L10073 056536	OSERRT= 000000	O.OLD 062764
LOOPFL 003114 G	L10002 005226	L10074 060460	OSGNSW= 000001	O.OP1 062770
LOT = 000010 G	L10003 011666	L10075 062044	OSPOIN= 000001	O.OP2 063034
LSACP 002110 G	L10004 011716	L10076 066700	OSSETU= 000001	O.OP2A 063042
LSAPT 002036 G	L10005 011734	L10077 067022	O.ADR1 066556	O.ORAB 062274
LSAU 022544 G	L10006 011742	L10100 067224	O.ALL 065142	O.ORPC 062252
LSAUT 002070 G	L10007 011760	L10102 067232	O.AS 062636	O.ORRB 062304
LSAUTO 022750 G	L10010 011776	MEMADD 013576 G	O.ASC 066125	O.P 066121
LSCCP 002106 G	L10011 012010	MENASC 021335	O.ASCI 064152	O.PCS 062264
LSCLEA 023024 G	L10012 012062	MENERR 021262	O.BACK 063122	O.PRNT 064410
LSCO 002032 G	L10013 012232	MENRES 021364	O.BALL 065026	O.PROC 063766
LSDEPO 002011 G	L10014 012746	MESBFA 002716 G	O.BD 066126	O.PROM 066134
LSDESC 003342 G	L10015 013574	MESBFN 014617	O.BKP = 000016	O.RALL 063312
LSDESP 002076 G	L10016 013616	MESHEA 015002	O.BKPT 063150	O.RCSR= 177560
LSDEVP 002060 G	L10017 016334	MIMENU 055552	O.BRK 064456	O.RDB = 177562
LSDISP 067174 G	L10020 016342	MMVEC = 000250	O.BW 066106	O.REG 066040
LSDLY 002116 G	L10021 016350	MPR = 174406	O.BYT 062674	O.REGT 062164
LSDTP 002040 G	L10022 016362	MSA.FR= 000006	O.BYT1 062666	O.REM 065312
LSDTYP 002034 G	L10023 016404	MSA.NO= 000000	O.CAD 066110	O.RCB 065246
LSDU 022642 G	L10024 016432	MSA.NR= 000004	O.CADV 065454	O.RSR 065216
LSDUT 002072 G	L10025 016572	MSA.VD= 000002	O.CLGT= 000035	O.RSTT 065406
LSDVTY 003334 G	L10026 017102	MSGEXP 012000 G	O.CLSE 065752	O.S 066117
LSEF 002052 G	L10030 022474	MSGLOO 012706 G	O.COMP 064312	O.SCAN 062430
LSENV1 002044 G	L10031 022640	MSGSTA 012172 G	O.CR 066131	O.SEMI 062630
LSETP 002102 G	L10032 022746	MSGSUB 013564 G	O.CRET 062756	O.SEQ 066124
LSEXP1 002046 G	L10033 023022	MS.ATT= 000006	O.CRLF 066004	O.SNGL 062354
LSEXP4 002064 G	L10034 023044	MS.EXT= 000200	O.CRLS 066020	O.SPAC 065740
LSEXP5 002066 G	L10035 023306	MS.RSD= 000001	O.CSR1 066122	O.STM = 000340
LSHARD 066646 G	L10036 033130	MS.RSF= 000020	O.CSR2 066123	O.SVR 065156
LSHIME 002120 G	L10037 024260	MS.RST= 000010	O.CT 066600	O.SVTT 065360
LSHPCP 002016 G	L10040 025304	NBA = 002000	O.C1 064040	O.SWCH 066550
LSHPTP 002022 G	L10041 026460	NEWPAS 022176	O.DCD 062404	O.T 066120
LSHW 002124 G	L10042 026762	NODEV 003062 G	O.DCDA 062762	O.TBIT 063716
LSICP 002104 G	L10043 030034	NOINIT 004233	O.DCDB 063310	O.TBT = 000020
LSINIT 021746 G	L10044 030734	NOINTR 004117	O.DCD1 062424	O.TCLS 062326
LSLADP 002026 G	L10045 034162	NOITS 002136 G	O.DCD2 062420	O.TCSR= 177564
LSLAST 067220 G	L10046 035514	NOMAN 021422	O.DOT 066112	O.TDB = 177566
LSLOAD 002100 G	L10047 034424	NP.IR = 000200	O.DUMP 064072	O.TL 066176
LSLUN 002074 G	L10050 041446	NP.LOO= 000040	O.EFF 063476	O.TRTC 066206
LSMREV 002050 G	L10051 036114	NP.OUT= 000100	O.ERR 062374	O.TVEC= 000014
LSNAME 002000 G	L10052 036572	NP.WRP= 000020	O.ERR1 063472	O.TYPE 065724
LSPRIO 002042 G	L10053 037000	NSI 004050	O.FCHR 066552	O.UIN 066622
LSPROT 021736 G	L10054 037266	NSINIT 004305	O.FCNT 066554	O.UPC 066536
LSPRT 002112 G	L10055 037610	NUL 004425	O.FTYP 065570	O.UPS 066540
LSREPP 002062 G	L10056 052714	NULCR 004426	O.GET 065636	O.URO 066520
LSREV 002010 G	L10057 042266	NXM = 004000	O.GO 063666	O.USB 066534
LSRPT 023046 G	L10060 043034	NXR 003636	O.G01 063744	O.WB1 062702

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-4
 SYMBOL TABLE

O.WDFG 066116	PST32W 003104 G	REWIND 010424 G	S1.IFM= 001000	TST21I 034006
O.WRD 062652	PUNIT 022476	RMCHBE= 000167	S1.IHE= 000400	TST22I 035322
O.WRD1 062716	PW.D11= 000021	RMCHEN= 000200	S1.IID= 004000	TST23I 041251
O.WSCH 063502	PW.D13= 000022	RMMSGB= 000104	S1.IIR= 020000	TST24I 052462
O.XXX 066114	PW.D22= 000020	RMMSGE= 000117	S1.I2R= 040000	TST39I 060432
PASRPT 022242	PW.NOP= 000000	RMPKTB= 000020	S1.PAR= 100000	TST40I 061740
PATCH 067214 G	PW.NO1= 000023	RMPKTE= 000027	S2.ATI= 000010	TTIBFR= 177562 G
PATDAT 021116	PW.RDE= 000024	RMR = 010000	S2.BTI= 000004	TTICSR= 177560 G
PC.ERA= 002400	PW.RDR= 000001	RWPACK 010520	S2.DIM= 000200	TTION 061162
PC.IER= 002000	PW.RDS= 000005	SC = 100000	S2.ILW= 000100	TTION2 053750
PC.NOO= 001000	PW.RFI= 000003	SCE = 020000	S2.INR= 000020	TTIVEC= 000060 G
PC.REL= 000000	PW.RFI= 000003	SCME 004711	S2.OUT= 000040	TTOBF= 177566
PC.REW= 000400	PV.WCT= 000006	SCMENU 061170	S2.UND= 000003	TTOCSR= 177564
PKBCNT= 000006	PW.WFI= 000004	SDELAY 010320	TBLEND= 003030 G	TUV2A 002000 G
PKHI = 000004	PW.WFM= 000007	SEEK = 000006	TCOASC 006166	TVECSA 061164
PKLOW = 000002	PW.WMI= 000010	SELASC 021330	TCOCOD 006366	TVSAV2 053752
PKTADD 007264	PW.WNP= 000011	SELDAT= 000004	TEMP1 003064 G	TSARGC= 000001
PKTFRM 007226	PW.WTR= 000002	SEL2 = 000002	TEMP2 003066 G	TSCODE= 001130
PKTGET 011720 G	P.ACK = 100000	SETMAP 020252	TERCLS= 000016	TSERRN= 001442
PKTMES 011744 G	P.CMD = 000037	SETU 022274	TESTNO= 000010	TSEXCP= 000000
PKTNEW 007321	P.COMT= 000012	SFFMSG 011736 G	TEXASC 006125	TSFLAG= 000040
PKTRAM 004643 G	P.CVC = 040000	SFHERR 003603	TFCASC 006227	TSFREE= 067232
PKTSSR 011670 G	P.FMT = 000140	SFIERR 003550	TIMEXP 016406 G	TSGMAN= 000000
PNT = 001000 G	P.FORM= 000011	SFIMSG 011656 G	TIMSGO 016434	TSHILI= 000007
PRAMPK 013620	P.GETS= 000017	SFPTBL 002134 G	TINERR 011643	TSLAST= 000001
PRBEXP 016324	P.IE = 000200	SIFLAG 003106 G	TKB = 177562	TSLOLI= 000000
PRBMSG 016172	P.INIT= 000013	SIMSG 011610	TKS = 177560	TSLSYM= 010000
PRBREC 016326	P.MODE= 007400	SKIP 037610	TMPBFR 002576 G	TSLTNO= 000010
PRBTOT 016257	P.OPP = 020000	SKIPT 003332	TMPEND 030762	TSNEST= 000000
PRBYTE 015756 G	P.POSI= 000010	SOFINI 016630 G	TNAM 017566	TSNS0 = 000000
PRI = 002000 G	P.READ= 000001	SPACE 010126 G	TPB = 177566	TSNS1 = 000005
PRIADD 007700	P.SWB = 010000	SPM1 067022	TPRISA 061166	TSNS2 = 000002
PRIAO 007750	P.WRIT= 000005	SPM4 067061	TPS = 177564	TSPCNT= 000000
PRIBXO 007332 G	P.WRTC= 000004	SPM6 067111	TPSAV2 053754	TSPTAB= 010101
PRIEQU 007600	P.WRTS= 000006	SPM7 067141	TRANST 002134 G	TSPTHV= 000001
PRIPKT 007060 G	QVP 002152 G	SRO = 177572	TSBA = 177776 G	TSPTMU= 000001
PRIRAM 007606	RAMASC 013766	SR1 = 177574	TSBAH = 177777 G	TSSAVL= 177777
PRITAD 010014	RAMDAT 002206 G	SR2 = 177576	TSBAL = 177776 G	TSSEGL= 177777
PRITSS 005264	RAMER 010626 G	SR3 = 172516	TSDB = 177776 G	TSSIZE= 000005
PRITO 010064	RAMERR 016344 G	SSR = 000200	TSDBH = 177777 G	TSSUBN= 000000
PRIXOR 007462 G	RAMEXP 016364 G	STATCO 012234	TSDBL = 177776 G	TSTAGL= 177777
PRI00 = 000000 G	RAMFHR 014526	SVCGBL= 000000	TSFCOD 006726	TSTAGN= 010103
PRI01 = 000040 G	RAMFOR 007636	SVCINS= 000000	TSREJ = 000006	TSTEMP= 000011
PRI02 = 000100 G	RAMHLD 011010	SVCSUB= 000001	TSSDEF 006276	TSTEST= 000010
PRI03 = 000140 G	RAMIOP 011014	SVCTAG= 000000	TSSR = 000000 G	TSTSTM= 177777
PRI04 = 000200 G	RAMPD 011065	SVCTST= 000001	TSSRBI 003400 G	TSTSTS= 000001
PRI05 = 000240 G	RAMRSH 011012	SSLSYM= 010000	TSSRFO 006105	TSSAU = 010031
PRI06 = 000300 G	RAMSIZ 002246 G	SO.IDB= 000010	TSSRH = 000001 G	TSSAUT= 010033
PRI07 = 000340 G	RAMTAD 016352 G	SO.IFB= 000002	TSSX 003716	TSSCLE= 010034
PRMESS 014052	RBPCRA 015114	SO.IFP= 000001	TSTBLK 002720 G	TSSDAT= 010102
PRMNO 002264 G	RCVHIA 002250 G	SO.IFD= 000020	TSTCNT 002162 G	TSSDU = 010032
PRMSGE 015406 G	RCVLOA 002252 G	SO.ION= 000040	TSTEND 017602	TSSHAR= 010076
PRMSGO 015566	RDERR 005104	SO.IRD= 000100	TSTFLA 002260 G	TSSHW = 010000
PRMSG1 015633	READ = 000014	SO.ISP= 000004	TSTL00 017340 G	TSSINI= 010030
PRMSG2 015671	READY = 000001	SO.ISP= 000200	TSTPTR 002262 G	TSSMSG= 010025
PROASC 014677	RECMG 002432 G	S1.ICE= 002000	TSTSET 017372 G	TSSPC = 000001
PRIASC 014744	RECV 002200 G	S1.IEO= 010000	TST17I 031216	TSSPRO= 010027
	REGSAV 021022			

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-5
 SYMBOL TABLE

TSSPTA=	010101	T21DLY	033566	T23TM	040226	T3.1	034224	T39TAD	057230
TSSRPT=	010035	T21LOO	033172	T23TMA	040315	T38ASC	056471	T39WPN	060274
TSSSOF=	010077	T21OFL	033765	T23VCK	041035	T38ASN	056510	T39WR	057762
TSSSRV=	010026	T21PAC	033440	T23VCK	041035	T38ASO	056366	T39WRT	060161
TSSSUB=	010072	T21PK2	033550	T23WB	040002	T38AS1	056433	T4	035516 G
TSSSW =	010001	T21RES	034030	T23WD	040020	T38BFR	054006	T4.1	035566
TSSTES=	010075	T21RT2	034120	T23WDC	040733	T38BS0	054000	T4.2	036132
T1	023630 G	T21SSR	033570	T23WDD	040644	T38BS1	054001	T4.3	036610
T1.1	023666	T21S2	033562	T23WDR	040022	T38BS2	054002	T4.4	037016
T1.2	024274	T21S3	033564	T23WSS	041162	T38CNT	056534	T4.5	037304
T1.3	025320	T22AM3	034725	T24AM3	051450	T38DAT	056024	T4ONE	061673
T1.4	026474	T22BFR	034512	T24BA	052002	T38DLY	053756	T4ONSR	061754
T1.5	026776	T22BF2	034610	T24BFR	050060	T38DTA	054500	T5	041450 G
T1.6	030050	T22BS0	034610	T24BF2	050200	T38EAI	054506	T5.1	041514
T17BEN	033002	T22BS1	034611	T24BOT	051043	T38EB	054462	T5.10	046560
T17BFR	032662	T22DAT	034500	T24BS0	050200	T38ID	055526	T5.11	047052
T17BFS	032702	T22FOR	034624	T24BS1	050201	T38MBP	056064	T5.12	047342
T17CLE	032564	T22FOR	034624	T24CON	050212	T36MSG	055430	T5.2	042304
T17CLR	032376	T22LOO	034224	T24DAT	050050	T38MS2	055364	T5.3	043052
T17DAT	032650	T22OFL	035025	T24DLY	050216	T38MS4	055144	T5.4	043546
T17DLY	033126	T22PAC	034470	T24DTA	051110	T38MS5	055223	T5.5	044234
T17DT2	033020	T22PK2	034600	T24EOT	051176	T38NE	054550	T5.6	044632
T17EXE	031114	T22POS	034622	T24ILA	050572	T38OFL	055110	T5.7	045260
T17EXP	030772	T22RD	034616	T24LON	052142	T38ONL	055046	T5.8	045642
T17EXS	031012	T22RES	035356	T24LOO	041514	T38PAC	053770	T5.9	046316
T17LOO	024046	T22RT2	035450	T24LOP	052224	T38PK2	054470	T6	052716 G
T17MSK	030766	T22RWJ	035174	T24LOQ	050656	T38PK3	054520	T7	056540 G
T17PAC	032640	T22SSR	034630	T24LOR	050272	T38PK4	054540	T8	060462 G
T17PK2	033010	T22S2	034612	T24NEF	050220	T38RES	056026	UAM =	000200 G
T17RFI	032544	T22S3	034614	T24NXM	050431	T38SI2	054546	UNITN	002150 G
T17RSF	032442	T22TM	035100	T24OFL	051515	T38SST	055300	UNREC =	000006
T17RSF	032442	T22VCK	035247	T24PAC	050040	T38TAD	054000	USI	004021
T17RWJ	031235	T22WRT	034620	T24PBP	052306	T38WLE	054752	WAITF	017104 G
T17SET	032604	T23AM3	040523	T24PK2	050150	T38WOK	055470	WC.IFA=	002200
T17SNP	032464	T23BA	041110	T24PK3	050170	T38WR	054542	WC.IFE=	000002
T17SRD	032422	T23BFR	037670	T24RB	050172	T38WRL	054711	WC.IG0=	000001
T17SSR	031263	T23BF2	040012	T24RES	052530	T38WRT	054625	WC.IRE=	000010
T17WFD	031114	T23BS0	040012	T24RN	050206	T38BFR	057236	WC.IRW=	000004
T17WFI	032510	T23BS1	040013	T24RNC	051375	T39BS0	057230	WC.IOT=	000100
T171CM	031603	T23CON	040024	T24RT2	052622	T39BS1	057231	WC.IIT=	000040
T172CM	031665	T23DAT	037660	T24RT3	052664	T39BS2	057232	WC.I5R=	000020
T172SS	031320	T23EOT	040150	T24RWN	051326	T39DAT	060372	WF.IED=	000010
T173CM	031761	T23ET	040063	T24SSR	050737	T39DLY	057206	WF.IER=	000004
T173SS	031364	T23LOO	035566	T24SZ	050176	T39DTA	057730	WF.IHI=	000200
T174CM	032045	T23OFL	040571	T24S2	050202	T39EAI	057736	WF.IRE=	000040
T174SS	031431	T23PAC	037650	T24S3	050204	T39MCL	060313	WF.IWF=	000020
T175CM	032130	T23PK2	037760	T24TM	051253	T39NE	057773	WF.IWR=	000100
T175SS	031474	T23PK3	040000	T24TRL	052374	T39NFL	057770	WF.I3R=	000002
T176CM	032204	T23RES	041266	T24VCK	051727	T39OF2	060052	WF.I4R=	000001
T176SS	031540	T23RNC	040450	T24WB	050172	T39ON2	060116	WRTCHR	010322 G
T177CM	032312	T23RSZ	040010	T24WDC	051656	T39PAC	057220	WRTERR	005011
T2	033132 G	T23RT2	041360	T24WDD	051570	T39PK2	057720	WRTMSG	004754
T21AM3	033665	T23RT3	041422	T24WDE	050771	T39PK3	057750	XFERAS	016574
T21BFR	033460	T23RWN	040401	T24WDF	050515	T39PK4	057760	XNXM	017260
T21BF2	033560	T23SSR	040030	T24WDG	050342	T39RES	060374	XORBFO	007414
T21BS0	033560	T23SZ	040006	T24WDR	050210	T39RL	060370	XORFOR	007532
T21BS1	033561	T23S2	040014	T24WSS	052053	T39SLZ	057766	XSIQ =	000006 G
T21DAT	033450	T23S3	040016	T3	034164 G				

CZTUXAO TUBO FRONT END PRT B MACRO M1200 29-MAR-83 13:32 PAGE 141-6
 SYMBOL TABLE

XST1 = 000010 G	XSONEF= 002000	XS3CON 015340	X1.UNC= 000002	X3.MDE= 177400
XST2 = 000012 G	XSOONL= 000100	XXCOMM 003070 G	X2.BUF= 000100	X3.OPI= 000100
XST3 = 000014 G	XSOPED= 000010	XSALWA= 000000	X2.EXT= 000200	X3.REV= 000040
XST4 = 000016 G	XSORLL= 010000	XSFALS= 000040	X2.OPM= 100000	X3.RIB= 000001
XSOBOT= 000002	XSORLS= 040000	XSOFFS= 000400	X2.RCE= 040000	X3.SPA= 000200
XSOCON 015161	XSOTMK= 100000	XSTRUE= 000020	X2.REV= 000077	X3.TRF= 000020
XSOEOT= 000001	XSOVCK= 000020	X1.COR= 020000	X2.SPA= 035400	X4.HSP= 100000
XSOIE = 000040	XSOWLE= 004000	X1.DLT= 100000	X2.UNI= 000007	X4.MBZ= 017400
XSOILA= 000400	XSOWLK= 000004	X1.MBZ= 017375	X2.WCF= 002000	X4.RCE= 040000
XSOILC= 001000	XS1CON 015226	X1.RBP= 000400	X3.DCK= 000010	X4.TSM= 020000
XSOLET= 020000	XS2CON 015273	X1.SPA= 040000	X3.MBZ= 000006	X4.WRC= 000377
XSOMOT= 000200				

. ABS. 067232 000
 000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 36936 WORDS (145 PAGES)
 DYNAMIC MEMORY: 20060 WORDS (77 PAGES)
 ELAPSED TIME: 00:09:15
 CZTUXA.BIC,CZTUXA/-SP=SVC.MLB/ML,CZTUXA.MAC