

TSV 05

TSV05 DATA REL
CVTSEB0

COPYRIGHT (c) 1982-83
AH-T179B-MC
FICHE 01 OF 01

APR 1984
digital
Made In USA

1000000	1000000
1000000	1000000
1000000	1000000
1000000	1000000

.REM_

IDENTIFICATION

PRODUCT CODE: AC-T1788-MC
PRODUCT NAME: CVTSEBO TSV05 DATA RELIABILITY
PRODUCT DATE: 25-APR-82
MAINTAINER: CSS/PPG DIAGNOSTICS
AUTHOR: DICK GORDON

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

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

COPYRIGHT (C) 1982,1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

USER DOCUMENTATION TABLE OF CONTENTS

GLOSSARY

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

1.1.1 FUNCTIONAL DESCRIPTION

1.1.2 STRUCTURE OF PROGRAM

1.1.3 MEMORY MAP

1.1.4 DIAGNOSTIC INFORMATION

1.1.4.1 SCOPE

1.1.4.2 ERROR RECOVERY

1.1.4.3 WRITE ERROR RECOVERY

1.1.4.3.1

MEDIA/OPERATION

SELECTIVE WRITE-ERROR-RECOVERY

1.1.4.3.2

OPERATIONAL WRITE-ERROR-RECOVERY

1.1.4.4 DIAGNOSTIC TIMING ADJUSTMENT

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE REQUIREMENTS

1.2.2 SOFTWARE REQUIREMENTS

1.3 RELATED DOCUMENTS AND STANDARDS

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

1.5 ASSUMPTIONS

2.0 OPERATING INSTRUCTIONS

2.1 HARDWARE PARAMETERS

2.2 SOFTWARE PARAMETERS

2.2.1 TSV05 COMMAND LIST

2.2.2 DATA PATTERNS

2.3 EXAMPLES OF SOFTWARE PARAMETER DIALOGUE

2.3.1 BASIC FUNCTION AND DATA RELIABILITY
WITH ALL ERROR REPORTING ENABLED

2.3.2 SCOPE LOOP SET UP IN BASIC FUNCTIONS

2.3.3 SCOPE LOOP SET UP IN DATA RELIABILITY

.PAGE

2.4 EXECUTION TIMES

105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161

2.4.1 SYSTEM CONFIGURATION
2.4.2 TEST EXECUTION TIMES

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

3.1.1 ERROR #1 - COMMAND PACKET ADDRESS IS NOT ON A
MODULO 4 BOUNDARY
3.1.2 ERROR #2 - TS05 NOT READY
3.1.3 ERROR #3 - NO RESPONSE ERRORS
3.1.4 ERROR #4 - NO INTERRUPT ERROR
3.1.5 SPECIAL CONDITION ERRORS
3.1.5.1 ERROR #5 - TCC0, UNDEFINED SPECIAL CONDITION
3.1.5.2 ERROR #6 - TCC1, ATTENTION CONDITION
3.1.5.3 ERROR #7 - TCC2, TAPE STATUS ALERT
3.1.5.4 ERROR #8 - TCC3, FUNCTION REJECT
3.1.5.5 ERROR #9 - TCC4, RECOVERABLE ERROR
3.1.5.6 ERROR #10 - TCC5, RECOVERABLE ERROR
3.1.5.7 ERROR #11 - TCC6, UNRECOVERABLE ERROR
3.1.5.8 ERROR #12 - TCC7, FATAL SUBSYSTEM ERROR
3.1.6 ERROR #13 - RFC NON-ZERO ERROR
3.1.7 ERROR #14 - RETRY LIMIT EXCEEDED
3.1.8 ERROR #15 - TOO MANY INTERRUPTS
3.1.9 ERROR #16 - CAPSTAN RUNAWAY
3.1.10 ERROR #17 - DATA COMPARE ERRORS

3.2 ERROR HALTS

4.0 PERFORMANCE REPORT

5.0 TEST SUMMARIES

5.1 TEST 1 - BASIC FUNCTIONS
5.2 TEST 2 - DATA RELIABILITY
5.3 TEST 3 - WRITE COMPATABILITY/WRITE UTILITY
5.4 TEST 4 - READ COMPATABILITY/READ UTILITY
5.5 TEST 5 - RANDOM/OPERATOR SELECTED COMMAND SEQUENCE

6.0 DEVICE INFORMATION

6.1 GENERAL
6.2 Q-BUS INTERFACE SPECIFICATIONS
6.3 BIT DEFINITIONS FOR TSV05/TS05 REGISTERS
6.3.1 TSV05/TS05 REGISTER SUMMARY
6.3.2 TSV05 STATUS REGISTER (TSSR)
6.3.2.1 TSV05 EXTENDED DATA BUFFER REGISTER (TSD8X)
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)
6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1)

162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218

6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)
6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3)
6.3.7 EXTENDED STATUS REGISTER 4 (XSTAT4)

7.0 DIAGNOSTIC HISTORY

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

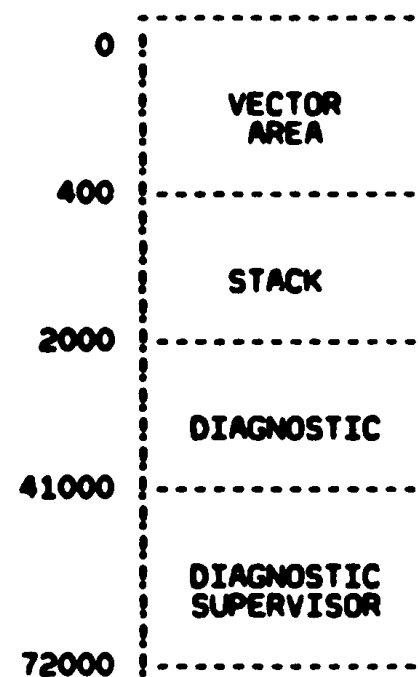
1.1.1 FUNCTIONAL DESCRIPTION

THIS PROGRAM CAN BE USED AS A BASIC FUNCTION TEST, A DATA RELIABILITY TEST, OR A COMPATIBILITY TEST.

1.1.2 STRUCTURE OF PROGRAM

THIS DIAGNOSTIC IS A SINGLE PROGRAM FROM THE STANDPOINT OF THE DIAGNOSTIC USER, BUT IT CONTAINS A CONTROL MODULE RELEASED INDEPENDENTLY AS A DIAGNOSTIC SUPERVISOR.

1.1.3 MEMORY MAP



276
277
278
279
280
281
282
283
284
285
286
287
288
G SUSPECTED
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
ABORTED.
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332

OPERATIONAL ONES.

ALGORITHM

A WRITE RETRY SUBROUTINE IS CALLED BY THE RECOVERABLE ERROR SUBROUTINE WHICH IS ENTERED UPON DETECTION OF A WRITE RECOVERABLE ERROR. THE WRITE RETRY SUBROUTINE ATTEMPTS TO REWRITE THE RECORD IN SAME SPOT ON TAPE 4 TIMES.

IF ALL 4 REPEATS ARE GOOD, THE RECORD IS CONSIDERED AS RECOVERED AND A RECOVERABLE WRITE ERROR IS LOGGED AT THAT RECORD NUMBER.

IF ANY OF THE 4 REWRITE ATTEMPTS FAIL, THE ROUTINE WILL ERASE THE BAD RECORD, AND LOG BAD SPOT AT THAT RECORD NUMBER, THE ROUTINE WILL THEN ATTEMPT TO WRITE THE RECORD AGAIN 3 INCHES FURTHER DOWN TAPE AND RETRY THIS SEQUENCE 4 TIMES, FOR UP TO 4 REPEATS EACH.

IF A RECORD CANNOT BE WRITTEN WITHOUT RECOVERABLE ERRORS AFTER 4 RETRIES, THEN THE ROUTINE WILL ERASE THE RECORD AND REPORT RETRY FAILED ON BAD SPOT.

THE RECOVERABLE ERROR SUBROUTINE THEN CONTINUES TO CALL THE WRITE RETRY SUBROUTINE, WHICH REISSUES THE GROUP OF 4 RETRIES, UNTIL THE RECORD IS RECOVERED OR 20 BAD SPOTS HAVE BEEN LOGGED.

TWENTY (20) BAD SPOTS MAXIMUM ARE ALLOWED PER BOT TO EOT PASS OF TAPE. WHEN 20 BAD SPOTS HAVE BEEN LOGGED, WHETHER ON THE SAME RECORD NUMBER OR NOT, TAPE IS CONSIDERED DEFECTIVE; A BAD TAPE OVERFLOW MESSAGE IS PRINTED AND THE UNIT IS REWOUND, THEN DROPPED.

DURING THE RECOVERY PROCESS, IT IS NECESSARY TO PERFORM SEVERAL TAPE POSITIONING OPERATIONS: SPACE REVERSE, ERASE. IF A POSITION ERROR IS DETECTED IN THE STATUS WORD DURING THOSE OPERATIONS, THEN THE RECOVERY ATTEMPT IS AN APPROPRIATE UNRECOVERABLE ERROR MESSAGE IS PRINTED AND THE UNIT IS DROPPED.

ALL BADLY WRITTEN RECORDS LOGGED WITH RECOVERABLE ERRORS ARE ERASED UNTIL RECOVERED, INCLUDING THE RECORD AT THE 20TH BAD SPOT, SO THAT ALL RECORDS LEFT ON TAPE ARE KNOWN GOOD WRITTEN RECORDS.

BAD SPOTS ARE ERASED WITH ERASE GAPS FROM 3 TO 12 INCHES PER RETRY GROUP. UP TO 20 FEET OF ERASE GAP COULD RESULT WHEN RETRYING TO RECOVER A SINGLE RECORD. THAT LONG STRETCH OF BAD TAPE WOULD THEN BE LOGGED WITH 20 BAD SPOTS AT SAME RECORD NUMBER AND THE TAPE CONSIDERED DEFECTIVE.

BAD SPOTS REPORTS

IF THE PRINTING OF RECOVERABLE ERRORS IS ENABLED, THE BAD SPOTS ON TAPE ARE IDENTIFIED AS THEY ARE DETECTED. SINCE THE BAD RECORDS ARE ERASED UNTIL RECOVERED, THE BAD SPOT ACTUALLY PRECEDES THE RECORD NUMBER THAT IDENTIFIES IT. THE NUMBER OF REPEATS AND RETRIES ATTEMPTED IS PRINTED, FROM WHICH THE LENGTH OF ERASE GAPS CAN BE DETERMINED: APPROXIMATELY 3 INCHES PER RETRY.

THE STATISTICAL REPORT PRINTED AT THE END OF TEST 2 OR UPON A "PRINT" REQUEST, CONTAINS A SUMMARY OF THE BAD SPOTS LOGGED ON THE CURRENT PASS OF TAPE. IN THAT REPORT, ALL COUNTS ARE CUMULATIVE FROM PASS TO PASS, EXCEPT FOR THE NUMBER OF BAD SPOTS; IT RELATES TO A "BOT TO EOT TAPE PASS" ONLY. FOR THIS PURPOSE, A "TAPE PASS" IS A WRITE PASS FROM BOT TO EOT, OR FROM

333
334
LL THE TESTS REQUESTED

BOT TO WHERE THE DIAGNOSTIC IS HALTED BEFORE REACHING EOT.
DON'T CONFUSE THIS WITH A PASS BY THE SUPERVISOR WHICH IS DEFINED AS A RUN THROUGH A
ON ALL UNITS SELECTED. THOSE PASSES ARE IDENTIFIED AS "PASS" AND "EOP".

335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389

THE NUMBER OF WRITE RETRIES, CUMULATIVE FROM PASS TO PASS, IS A GLOBAL
COUNT OF HOW MANY TIMES THE GROUP OF 4 RETRIES HAS BEEN CALLED.

THE NUMBER OF WRITE RECOVERABLE ERRORS EXCLUDES BAD TAPE SPOTS
AND REFLECTS THE SPECIFICATIONS OF THE HARDWARE UNDER TEST.

TO CLEAR CUMULATIVE COUNTS, ANSWER 'Y' TO: CLEAR COUNTERS (L) Y ?.
THE BAD TAPE SPOTS COUNT IS THEN CLEARED WHEN WRITING THE TAPE FROM BOT.

IF TEST 2 IS HALTED, THEN RESTARTED OR CONTINUED, THE RECORD COUNT
IS RESET TO ZERO AND THE BAD SPOT ID SHALL FOLLOW THAT RESET COUNT.

SINCE ALL WRITTEN RECORDS ARE KNOWN GOOD, THE READ ERRORS CAN
BE ATTRIBUTED TO TRANSIENT NOISE, TRANSIENT ELECTRICAL MALFUNCTIONS,
OR CONTAMINANTS ON TAPE AS OPPOSED TO TAPE DEFECTS.

THE SAME RECORDS MUST BE WRITTEN FROM TAPE PASS TO TAPE PASS
FOR THE BAD SPOTS ID TO REMAIN CONSISTENT IN THOSE TAPE PASSES.

EXAMPLE OF A PRINT OUT FOR A BAD SPOT ON TAPE:

CVTSE SFT ERR 00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 0 PASS: 1 RECORD: 6
PREVIOUS CMD WAS WRT
CMDPKT TSBA RFC TSSR TCC
100205 002406 000000 100210 4
026600
000000
003107
XST0 XST1 XST2 XST3 XST4
000350 000002 100400 000000 000000
SUSPECT BAD SPOT AFTER 1 RETRY, 2 REPEAT
SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 3 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 4 RETRY, 3 REPEAT
RETRY FAILED ON BAD SPOT...ERASED!
SUSPECT BAD SPOT AFTER 1 RETRY, 1 REPEAT
SUSPECT BAD SPOT AFTER 2 RETRY, 1 REPEAT

CVTSE SFT ERR 00009 ON UNIT 00 TST 002 SUB 000 PC: 012100
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 0 PASS: 1 RECORD:10210
PREVIOUS CMD WAS WRT
CMDPKT TSBA RFC TSSR TCC
100205 002406 000000 100210 4
026600
000000
004000
XST0 XST1 XST2 XST3 XST4
000350 000002 100010 000000 000000
RECOVERED ON RETRY # 1

↑C
DR>PRI

UNIT 0 PASS: 1 RECORD:10210
 BYTES WRITTEN 0,272,279,691
 BYTES READ REV 0,301,123,654
 BYTES READ REV 0,301,120,381

	WRT	RDR	RDF
RECOVERABLE ERRORS	1	0	0
UNRECOVERABLE ERRORS	0	0	0
WRITE RETRIES	3		

2 BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:

SPEC COND	HARD	FATAL	COMPARE
6	6		
2	0	0	0

DR>

THIS EXAMPLE SHOWS:

RECORD 6 RECOVERED ON 2ND RETRY GROUP
 THE 2 BAD SPOTS RESIDE IN A 18 INCH ERASE GAP BETWEEN RECORDS 5 AND 6
 RECORD 10210 RECOVERED ON 1ST RETRY OF 4 GOOD REPEATS
 3 WRITE GROUP RETRIES ATTEMPTED, RESULTING IN:
 1 RECOVERABLE WRT ERR FROM RECORD 10210
 2 BAD SPOTS BETWEEN RECORDS 5 AND 6

1.1.4.3.2 OPERATIONAL WRITE-ERROR-RECOVERY ALGORITHM

WHEN THIS ALGORITHM IS SELECTED, THE TSV05 WRITE RETRY COMMAND IS ISSUED UP TO 16 TIMES OR UNTIL RECORD IS RECOVERED, ON A WRITE RECOVERABLE ERROR. THE WRITE RETRY COMMAND CONSISTS OF A SPACE REVERSE OVER THE BAD RECORD, THEN AN ERASE OF 3 INCHES OF TAPE AND REWRITE OF THE RECORD. THAT COMPOSITE COMMAND DOES NOT ALLOW THE DETECTION OF BAD SPOTS ON TAPE. THEREFORE NO BAD TAPE SPOTS STATUS IS PRINTED.

IF RECORD CANNOT BE RECOVERED AFTER 16 WRITE RETRY COMMANDS, A RETRY LIMIT EXCEEDED IS FLAGGED AND UNIT IS DROPPED.

1.1.4.4 DIAGNOSTIC TIMING ADJUSTMENT

A NUMBER OF SUPERVISOR TIMING DELAY MACROS, KNOWN AS WATCH DOG DELAYS, ARE CALLED BY THE DIAGNOSTIC TO WAIT FOR VARIOUS COMMANDS COMPLETION. THESE DELAYS ARE NOT CALIBRATED AND SIMPLY EXPANDS INTO AN INLINE NESTED LOOP PAIR. THE COUNT FOR THE OUTER LOOP COMES FROM THE VARIABLE ARGUMENT SUPPLIED BY THE DELAY CALLS. THE COUNT FOR THE INNER LOOP COMES FROM THE FIXED "HEADER" ELEMENT "LIDLY". AS THE DIAGNOSTIC IS RUN ON DIFFERENT CPU'S, THESE DELAYS WILL VARY IN LENGTH WITH MEMORY SPEED.

IF TIME-OUT OCCURS WHEN NO APPARENT MALFUNCTIONS IN THE TAPE UNIT IS EVIDENT, ALL TIMINGS OF THE DIAGNOSTIC MAY BE ADJUSTED TO MATCH MEMORY SPEED AND NOT RESULT IN TIME-OUTS, BY PATCHING

447 THAT FIXED DELAY ELEMENT "L#DLY".

448 A PRESET COUNT OF 500 RESIDES AT "L#DLY" IN LOCATION 2116 OF THE
449 "HEADER" SECTION.
450

451 1.2 SYSTEM REQUIREMENTS
452 -----
453

454 1.2.1 HARDWARE REQUIREMENTS
455 -----
456

457 PDP-11/23 PROCESSOR WITH 32K OR MORE OF MEMORY
458 CONSOLE DEVICE (VT52,LA36,ETC.)
459 PROGRAM LOAD DEVICE
460 TSV05/TS05
461

462 1.2.2 SOFTWARE REQUIREMENTS
463 -----
464

465 DIAGNOSTIC SUPERVISOR
466
467

468 1.3 RELATED DOCUMENTS AND STANDARDS
469 -----
470

471 DIGITAL EQUIPMENT CORPORATION DOCUMENTS:
472

- 473 1. CIOPMA0 XXDP+ PROGRAMMER'S MANUAL; DOCUMENT NUMBER AC-S296A-AC
474 DATE: 14 JULY 1980.
475
476 2. TSV05 TRANSPORT SUBSYSTEM USER'S GUIDE; DOCUMENT NUMBER EK-TSV05-UG-001
477 DATE: AUGUST 1982
478
479 3. TSV05 TRANSPORT SUBSYSTEM TECHNICAL MANUAL; DOCUMENT NUMBER EK-TSV05-TM-001
480 DATE: AUGUST 1982
481
482 4. TSV05 TRANSPORT SUBSYSTEM INSTALLATION MANUAL; DOCUMENT NUMBER EK-TSV05-IN-001
483 DATE: AUGUST 1982
484
485
486
487
488
489
490

491 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
492 -----
493

494 ORDER OF HOST CPU DIAGNOSTIC USAGE:
495

- 496 1) CONTROL LOGIC PROGRAM - ALL TESTS.
497

498 (VTSA,VTSB,VTSC,VTSD)
499

- 500 2) DATA RELIABILITY PROGRAM:
501

- 502 A) BASIC FUNCTION TEST.
503 B) DATA RELIABILITY TEST.

504
505
506
507
508
509
510
511
512
513
514

1.5 ASSUMPTIONS

THE HARDWARE OTHER THAN THE SUBSYSTEM BEING TESTED IS ASSUMED TO WORK
PROPERLY. FALSE ERRORS MAY BE REPORTED IF THE PROCESSOR, MEMORY, ETC.,
DO NOT FUNCTION PROPERLY.
VTSA, VTSB, VTSC, AND VTSD HAVE ALL SUCCESSFULLY RUN WITHOUT ERRORS.

2.0 OPERATING INSTRUCTIONS

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

COMMANDS

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

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

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

OPERATOR COMMANDS

THE TSV05 DIAGNOSTIC IS A PDP-11/23 DIAGNOSTIC SUPERVISOR COMPATIBLE
PROGRAM. ALL LOADING AND RUNTIME INSTRUCTIONS CAN BE REFERENCED IN THE
PDP-11 PROGRAMMER'S MANUAL "CIGPM40 XXDP+ PROGRAMMERS MANUAL, NUMBER
AC-S296A-AC. THE USER ENTRY IS IN QUOTES.

BOOT THE DIAGNOSTIC XXDP MEDIA

```
CHMDLBO XXDP+ DL MONITOR 28K
BOOTED VIA UNIT 0
ENTER DATE (DD-MMM-YR): " ENTER DATE OR JUST <CR> "
RESTART ADDRESS: 153726
50 HZ? N " <CR> "
LSI? N " Y<CR> "
THIS IS XXDP+. TYPE "H" OR "H/L" FOR DETAILS
R VTSEA0
VTSEA0BINDRS LOADED
DIAG. RUN-TIME SERVICES REV D. APR 79
```

CVTSE-A-0
TSV05 DATA RELIABILITY
UNIT IS TSV05

SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

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

EXAMPLE OF SWITCH USAGE:

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

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

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

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

FLAGS

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

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

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

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

/FLAGS:LOE:IER:BOE

2.1 HARDWARE PARAMETERS

ON A "N" RESPONSE TO "CHANGE HW?", THE DIAG SHALL RUN ASSUMING ONE UNIT AT TSDB = 172520 WITH A VECTOR = 224 AND DRIVE=0.

ON A "Y" RESPONSE TO "CHANGE HW?" QUESTION, THEN THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONSE.

687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743

TSDB ADDRESS (0) 172520 ?

VECTOR (0) 224 ?

SELECT DRIVE 0-1 (0) ?

THE VALIDITY OF THESE PARAMETERS CAN BE CHECKED BEFORE RUNNING THE TESTS BY SETTING THE FLAG "ADR" ON A STA, RES OR CON COMMAND. THE SO CALLED AUTO DROP CODE SHALL THEN BE EXECUTED AFTER THE INIT CODE AND BEFORE THE HARDWARE TESTS ARE RUN. THAT CODE FIRST TESTS THE ADDRESS OF THE TSDB(S). IF NO RESPONSE, IT DROPS THE UNIT(S) IMMEDIATELY WITH THE FOLLOWING MESSAGE:

BUS TRAP AT XXXXXX (XXXXXX = TSDB AD)
INTERFACE BAD OR NOT SET TO ABOVE ADDRESS.

ON A RESPONSE FROM THE INTERFACE, THE UNITS THAT ARE NOT READY OR NOT ON-LINE ARE DROPPED IMMEDIATELY. THE HARDWARE TESTS SHALL THEN BE RUN ON RESPONDING UNITS.

IF THE "ADR" FLAG IS NOT SET, THE READY AND OFF-LINE STATUS OF THE DRIVE IS CHECKED. A MESSAGE SHALL BE PRINTED EVERY SO OFTEN TO WARN THE OPERATOR OF DRIVES BEING NOT READY OR OFF-LINE. THESE DRIVES SHALL BE DROPPED AFTER A REASONABLE AMOUNT OF TIME.

2.2 SOFTWARE PARAMETERS

THE FOLLOWING QUESTIONS ARE ASKED WHEN ONE ANSWERS YES TO THE CHANGE SOFTWARE QUESTION ON A START, RESTART, OR CONTINUE. THEY ALLOW FLEXABILITY IN THE WAY THE PROGRAM BEHAVES.

CLEAR COUNTERS (L) Y ?

RESET RANDOM VARIABLES (L) N ?

PRINT RECOVERABLE ERRORS (L) N ?

HALT AFTER EACH CMD (L) N ?

INHIBIT RECOVERY (L) N ?

BAD TAPE SPOT DETECTION (L) Y ?

DISABLE INTERRUPTS (L) N ?

INHIBIT RFC ERROR REPORTS (L) N ?

CHANGE CMD SEQUENCE (L) N ? (SEE NOTE1:)

DEFAULT SWITCH SETTINGS (L) Y ?

100IPS (L) N ?

WRITE BUFFERING (L) N ?

744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800

READ BUFFERING (L) N ?

ANSWERING NO TO THE DEFAULT SWITCH QUESTION WILL CAUSE THE 100 IPS QUESTION TO BE ASKED.

ANSWERING YES TO THE 100 IPS QUESTION WILL INHIBIT THE LAST TWO QUESTIONS.

ANSWERING NO TO THE 100 IPS QUESTION WILL CAUSE THE WRITE BUFFERING QUESTION TO BE ASKED.

ANSWERING YES TO THE WRITE BUFFERING QUESTION WILL INHIBIT THE LAST QUESTION.

ANSWERING NO TO THE WRITE BUFFERING QUESTION WILL CAUSE THE READ BUFFERING QUESTION TO BE ASKED.

NOTE1: THIS QUESTION SHOULD BE ANSWERED (N) UNLESS AN OPERATOR SELECTED SEQUENCE IS TO BE EXECUTED. IF THIS QUESTION WAS ANSWERED Y, THE FOLLOWING QUESTIONS MUST BE ANSWERED OR DEFAULTED WITH A <CR> ONLY:

CHARACTERISTICS CODE (0) 40 ?	(0,20,40,200) (OCTAL)
CMD/2 (D) 13 ?	(1-27) (DECIMAL)
BRF COUNT (D) 1 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 1 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/3 (D) 4 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/4 (D) 3 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/5 (D) 2 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/6 (D) 13 ?	(1-27) (DECIMAL)
BRF COUNT (D) 1 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 1 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/7 (D) 27 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)
CMD/8 (D) 27 ?	(1-27) (DECIMAL)
BRF COUNT (D) 2048 ?	(1-2K) (DECIMAL)
# OF OPERATIONS (D) 32000 ?	(1-32K) (DECIMAL)
PATTERN (D) 7 ?	(0-8) (DECIMAL)

NOTE: THE PROGRAM AUTOMATICALLY INSERTS A CHARACTERISTIC CODE OF 40 AS THE FIRST COMMAND IN THE SEQUENCE TABLE. IF A DIFFERENT CHARACTERISTIC IS DESIRED, THE OPERATOR SHOULD

ENTER THAT CHARACTERISTIC CODE. A TOTAL OF 7 COMMANDS MAY BE ENTERED IN ADDITION TO THE SET CHARACTERISTICS COMMAND. IF THE OPERATOR WISHES TO USE LESS THAN 7 COMMANDS, AN END COMMAND MUST BE ENTERED AND THEN A CONTROL Z (↑Z) CAN BE ENTERED TO TERMINATE SOFTWARE DIALOGUE.

2.2.1 COMMAND LIST FOR USE IN SOFTWARE DIALOGUE.

CODE	COMMAND	DESCRIPTION
1	DRI	DRIVE INITIATE.
2	RDF	READ FORWARD.
3	RDR	READ REVERSE.
4	WRT	WRITE.
5	WTV	WRITE/VERIFY. IE. WRITE N RECORDS; READ REVERSE AND CHECK N RECORDS OF DATA; READ FORWARD AND CHECK N RECORDS.
6	SRF	SPACE RECORDS FORWARD.
7	SRR	SPACE RECORDS REVERSE.
8	RNR	READ NEXT REVERSE, IE. SPACE FWD, READ REV.
9	RNF	READ NEXT FORWARD, IE. READ FWD, SPACE REV.
10	RPF	READ PREVIOUS FWD, IE. SPACE REV, READ FWD.
11	RPR	READ PREVIOUS REV, IE. READ REV, SPACE FWD.
12	WRR	WRITE RETRY.
13	RWD	REWIND.
14	MBR	MESSAGE BUFFER RELEASE.
15	WTH	WRITE TAPE MARK.
16	WTR	WRITE TAPE MARK RETRY.
17	SFF	SPACE FILES FORWARD.
18	SFR	SPACE FILES REVERSE.
19	GES	GET EXTENDED STATUS.
20	ERS	ERASE 3 INCHES OF TAPE.
21	UNL	UNLOAD.
22	CLN	CLEAN TAPE
23	SCH	SET DEVICE CHARACTERISTIC. WHERE BRF=200, 40, 20, 0. 200 = ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT) 40 = ENABLE ATTENTION INTERRUPTS. 20 = ENABLE MESSAGE BUFFER RELEASE INTERRUPTS. SEE TSV05/TS05 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
25	JMP	JUMP TO THE NTH COMMAND IN THE COMMAND SEQUENCE TABLE, WHERE N IS DEFINED IN THE BRF FIELD. THE NUMBER OF JUMPS IS ENTERED IN THE # OF OPERATIONS FIELD
26	DLY	DELAY "N" MILLISECONDS WHERE N IS DEFINED IN THE # OF OPERATIONS.
27	END	END OF COMMAND SEQUENCE.

2.2.2 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE.

PATTERN #	DESCRIPTION.
0	INCREMENTING PATTERN. 0 - 377.
1	ALL "1"'S PATTERN.
2	ALL "0"'S PATTERN.
3	"1" BIT WALKING FROM R TO L IN A FIELD OF "0"'S.
4	"0" BIT WALKING FROM R TO L IF A FIELD OF "1"'S.
5	ALTERNATING "1" AND "0" BITS WITH ALTERNATE BYTES COMPLIMENTED.
6	ALTERNATING BYTES OF 000 AND 377.

E2

CVTSEBO TSV05 DATA RELIABILITY MACRO M1113 30-NOV-83 10:17

SEQ 017

858
859

7
8

RANDOM DATA PATTERN.
NO PATTERN GENERATION.

2.3 EXAMPLES OF SOFTWARE DIALOGUE

CHANGE HW (L) ?

#UNITS (D) ?

TSDB ADDRESS (O) 172520 ?

VECTOR (O) 224 ?

SELECT DRIVE 0-1 (O) ?

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

CHANGE SW (L) ?

2.3.1 BASIC FUNCTION AND DATA RELIABILITY WITH ALL ERROR REPORTING ENABLED

- A) RECEIVE PROMPT (DR>)
- B) ENTER STA/TES:1-2<CR>
- C) ANSWER HARDWARE QUESTIONS.
- D) PROCEED WITH THE FOLLOWING DIALOGUE:

CHANGE SW (L) ? Y<CR>

CLEAR COUNTERS (L) N ? Y<CR>

RESET RANDOM VARIABLES (L) N ? <CR>

PRINT RECOVERABLE ERRORS (L) N ? Y<CR>

HALT AFTER EACH CMD (L) N ? <CR>

INHIBIT RECOVERY (L) N ? <CR>

BAD TAPE SPOT DETECTION (L) Y ? <CR>

DISABLE INTERRUPTS (L) N ? <CR>

INHIBIT RFC ERROR REPORT (L) N ? <CR>

CHANGE CMD SEQUENCE (L) N ? <CR>

DEFAULT SWITCH SETTINGS (L) Y ? <CR>

2.3.2 TO SET UP A SCOPE LOOP FOR A FAILURE IN BASIC FUNCTIONS.

- A) RECEIVE PROMPT (DR>)
- B) ENTER STA/TES:1/FLA:LOE:IER:ISR:IDU<CR>
- C) ANSWER HARDWARE QUESTIONS.
- D) PROCEED WITH THE FOLLOWING DIALOGUE:

CHANGE SW (L) ? Y<CR>

CLEAR COUNTERS (L) N ? Y<CR>

RESET RANDOM VARIABLES (L) N ? N<CR>

PRINT RECOVERABLE ERRORS (L) N ? N<CR>

HALT AFTER EACH CMD (L) N ? N<CR>

INHIBIT RECOVERY (L) N ? N<CR>

BAD TAPE SPOT DETECTION (L) Y ? N<CR>

DISABLE INTERRUPTS (L) N ? N<CR>

918 INHIBIT RFC ERROR REPORT (L) N ? Y<CR>
 919 CHANGE CMD SEQUENCE (L) N ? N<CR>
 920 DEFAULT SWITCH SETTINGS (L) Y ? <CR>

2.3.3 TO SET UP A SCOPE LOOP FOR A FAILURE IN DATA RELIABILITY

- A) RECEIVE PROMPT (DR>)
 B) ENTER STA/TES:5/FLA:IER:ISR:IDU/EOP:1000<CR>
 C) ANSWER HARDWARE QUESTIONS.
 D) PROCEED WITH THE FOLLOWING DIALOGUE:

CHANGE SW (L) ?	Y<CR>	
CLEAR COUNTERS (L) N ?	Y<CR>	
RESET RANDOM VARIABLES (L) N ?	N<CR>	
PRINT RECOVERABLE ERRORS (L) N ?	N<CR>	
HALT AFTER EACH CMD (L) N ?	N<CR>	
INHIBIT RECOVERY (L) N ?	N<CR>	
BAD TAPE SPOT DETECTION (L) Y ?	N<CR>	
DISABLE INTERRUPTS (L) N ?	Y<CR>	
INHIBIT RFC ERROR REPORT (L) N ?	Y<CR>	
CHANGE CMD SEQUENCE (L) N ?	Y<CR>	
CHARACTERISTICS CODE (O) 40 ?	40<CR>	
CMD/2 (D) 5 ?	13<CR>	(REWIND)
BRF COUNT (D) 2048 ?	1<CR>	
# OF OPERATIONS (D) 10 ?	1<CR>	
PATTERN (D) 7 ?	1<CR>	
CMD/3 (D) 5 ?	4<CR>	(WRITE)
BPF (D) 2048 ?	1000<CR>	
# OF OPERATIONS (D) 10 ?	10000<CR>	
PATTERN (D) 7 ?	1<CR>	
CMD/4 (D) 5 ?	27<CR>	(END)
BRF (D) 2048 ?	<+Z>	

2.4 EXECUTION TIMES

2.4.1 SYSTEM CONFIGURATION

PDP11/23
 MOS MEMORY
 LA36
 TSV05/TS05

2.4.2 TEST EXECUTION TIMES (2400 FT. TAPE)

TEST 1 - BASIC FUNCTIONS - 30 SECONDS PER PASS.
 TEST 2 - DATA RELIABILITY - 45 MINUTES PER PASS.
 TEST 3 - WRITE COMPATABILITY - 20 MINUTES PER PASS.
 TEST 4 - READ COMPATABILITY - 20 MINUTES PER PASS.
 TEST 5 - RANDOM/OPERATOR SELECTED SEQUENCE -20 MINUTES PER PASS.

NOTE: ALL EXECUTION TIMES ARE SHOWN FOR ONE DRIVE OPERATION.

3.0 ERROR INFORMATION

3.1 ERROR REPORTING

ALL ERROR REPORTS EXCEPT FOR ERRORS #1 AND #17 INCLUDE A DUMP
OF THE FOLLOWING INFORMATION:

ERROR #, TEST #, SUBTEST #, PROGRAM COUNTER, UNIT #,
COMMAND, PREVIOUS COMMAND, PASS COUNT, # OF
RECORDS FROM BOT, RECORD READ COUNT, THE COMMAND PACKET, TSSR,
TCC, TSBA, RFC, AND THE EXTENDED STATUS REGISTERS
(SEE 2.3.14.1 FOR LIST OF COMMANDS).

STANDARD ERROR REPORT FORMAT:

```
CVTSE SFT ERR XXXXX TST XXX SUB XXX PC: XXXXXX
(ASCII ERROR MESSAGE)
XXX CMD FAILED - UNIT X PASS: XXXXX RECORD: XXXXX
PREVIOUS CMD WAS XXX * RECORD READ: XXXXX *
CMDPKT TSBA RFC TSSR TCC
XXXXXX XXXXXX XXXXXX XXXXXX X
XXXXXX
XXXXXX
XXXXXX
XST0 XST1 XST2 XST3 XST4
XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX
```

* CAUTION *

INTERPRET THAT "RECORD READ" COUNT WITH CAUTION.
IF VERY DIFFERENT FROM RECORD COUNT TRACKED BY THE DIAGNOSTIC,
TAPE POSITION IS NOT NECESSARELY LOST. ERRORS IN READING THAT
RECORD MIGHT HAVE CAUSED RECORD COUNT TO BE ERRONEOUSLY
READ FROM TAPE.
IN TEST 2, IF DIAGNOSTIC IS RESTARTED OR CONTINUED, RECORD COUNT
IS RESET TO ZERO ALTHOUGH THE TAPE IS NOT REWOUND. THIS IS
NECESSARY BECAUSE THERE IS NO ACCURATE WAY TO DETERMINE
ON WHAT RECORD COUNT OF WHICH UNIT THE DIAGNOSTIC WAS HALTED
BEFORE RESTARTING OR CONTINUING.
IT IS SUGGESTED THAT A "PRINT" BE REQUESTED WHEN HALTING DIAG
TO GET A PRINT OF THE RECORD COUNT WHEN HALTED.

EXAMPLE OF AN ERROR REPORT:

```
CVTSE SFT ERR 00009 TST 002 SUB 000 PC: 010606
RECOVERABLE ERROR
WRT CMD FAILED - UNIT 2 PASS: 2 RECORD: 254
PREVIOUS CMD WAS WRT
CMDPKT TSBA RFC TSSR TCC
100005 002324 000000 100210 4
051766
000000
```

1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088

000371

XST0	XST1	XST2	XST3	XST4
000350	000002	100004	000000	040055

3.1.1 ERROR #1 - COMMAND PACKET ADDRESS NOT ON A MODULO 4 BOUNDARY:

IF THIS ERROR IS REPORTED, THE PROGRAM DID NOT LOAD PROPERLY. THIS IS A SYSTEM FATAL ERROR AND THE PROGRAM MUST BE RELOADED TO CORRECT IT.

3.1.2 ERROR #2 - TS05 NOT READY:

BEFORE ANY COMMAND IS ISSUED TO THE TS05, THE SUBSYSTEM READY BIT IN THE TSSR IS CHECKED. IF THE SSR IS NOT SET, THE PROGRAM REPORTS THE NOT READY ERROR. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE UNLESS THE IDU OPTION IS USED.

3.1.3 ERROR #3 - NO RESPONSE ERROR:

ONCE THE TSDB IS LOADED, THE TS05 HAS ONE MILLISECOND TO RESPOND OR THE PROGRAM REPORTS A NO RESPONSE ERROR. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST SEQUENCE UNLESS THE IDU OPTION IS USED.

3.1.4 ERROR #4 - NO INTERRUPT ERROR:

COMMAND WAS ISSUED AND NO INTERRUPT RECEIVED. THE PROGRAM REPORTS THAT NO INTERRUPT OCCURRED. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5 SPECIAL CONDITION ERRORS:

IF, DURING EXECUTION, AN INCIDENT OCCURS FORCING THE TSSR SPECIAL CONDITION BIT TO SET, THE PROGRAM WILL SELECT ONE OF 8 ERROR HANDLING ROUTINES, DEPENDING ON THE TERMINATION CLASS CODE.

THE TERMINATION CLASS CODES IN THE TSSR ARE PROCESSED AS FOLLOWS WHEN SPECIAL CONDITION IS SET:

3.1.5.1 ERROR #5 - TERMINATION CLASS CODE 0, UNDEFINED SPECIAL CONDITION

THE ERROR IS REPORTED, A HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.

3.1.5.2 ERROR #6 - TERMINATION CLASS CODE 1, ATTENTION CONDITION

THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE

1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145

SUCH AS GOING OFFLINE OR COMING ONLINE. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.3 ERROR #7 - TERMINATION CLASS CODE 2, TAPE STATUS ALERT

A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE TO THE PROGRAM. BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, EOT. ACTION TAKEN DEPENDS ON THE TEST BEING EXECUTED. IF THE CONDITION IS UNEXPECTED, THE ERROR IS REPORTED AND A HARD ERROR IS LOGGED. THE PROGRAM PROCEEDS NORMALLY.

3.1.5.4 ERROR #8 - TERMINATION CLASS CODE 3, FUNCTION REJECT

THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST ARE RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA. THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.5 ERROR #9 - TERMINATION CLASS CODE 4, RECOVERABLE ERROR

TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN THE FUNCTION WAS INITIATED. RECOVERY PROCEDURE IS TO LOG THE ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND. IF RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, RETRY LIMIT EXCEEDED IS REPORTED AS DESCRIBED IN ERROR #14 BELOW.

3.1.5.6 ERROR #10 - TERMINATION CLASS CODE 5, RECOVERABLE ERROR

TAPE POSITION HAS NOT CHANGED. RECOVERY PROCEDURE IS TO LOG THE ERROR AND RE-ISSUE THE ORIGINAL COMMAND. IF RETRY LIMIT IS REACHED BEFORE THE ERROR IS RECOVERED, RETRY LIMIT EXCEEDED IS REPORTED AS DESCRIBED IN ERROR #14 BELOW.

3.1.5.7 ERROR #11 - TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR

TAPE POSITION HAS BEEN LOST. THE ONLY VALID RECOVERY PROCEDURE IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR SEQUENCE NUMBERS. IF DENSITY CHECK IS SET THIS DIAGNOSTIC WILL REWIND AND RETRY THE COMMAND, OTHERWISE THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

3.1.5.8 ERROR #12 - TERMINATION CLASS CODE 7, FATAL SUBSYSTEM ERROR

THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE. REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR. THE DEVICE WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.

1146 3.1.6 ERROR #13 - RFC NON-ZERO ERROR:
1147
1148 IF, AFTER EXECUTION, THE RESIDUAL FRAME COUNT IS NON-ZERO, THE
1149 ERROR IS REPORTED AND A HARD ERROR IS LOGGED. THE PROGRAM THEN
1150 PROCEEDS NORMALLY. THE REPORTING AND LOGGING OF THESE ERRORS
1151 IS OPTIONAL.
1152
1153
1154 3.1.7 ERROR #14 - RETRY LIMIT EXCEEDED:
1155
1156 ON A WRITE COMMAND THIS IS A FATAL DEVICE ERROR AND THE DEVICE
1157 WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1158
1159 ON A READ COMMAND THIS ERROR IS LOGGED AS A HARD ERROR AND
1160 THE PROGRAM PROCEEDS NORMALLY.
1161
1162
1163 3.1.8 ERROR #15 - TOO MANY INTERRUPTS:
1164
1165 IF MORE THAN ONE INTERRUPT OCCURS PER COMMAND, THIS ERROR IS REPORTED.
1166 THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1167 THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1168
1169
1170 3.1.9 ERROR #16 - CAPSTAN RUNAWAY:
1171
1172 CAPSTAN DID NOT STOP WITHIN ACCEPTABLE WINDOW AFTER LAST
1173 COMMAND. THE PROGRAM WILL ISSUE A GET STATUS COMMAND BEFORE REPORTING
1174 THE ERROR SO THAT THE DEAD TRACK FIELD IN EXTENDED STATUS REGISTER 2
1175 WILL CONTAIN THE TACH COUNT WHEN THE TAPE STOPPED.
1176 THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1177 THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1178
1179
1180 3.1.10 ERROR #17 - DATA COMPARE ERROR:
1181
1182 IF A DATA VALIDATION ERROR OCCURS DURING A WRITE/VERIFY COMMAND,
1183 THE PROGRAM PRINTS WHAT THE DATA SHOULD HAVE BEEN AND WHAT THE
1184 DATA WAS, AND PRINTS THE BYTE AND RECORD NUMBER THE ERROR OCCURRED
1185 ON. ONLY THE FIRST 10 BYTES IN ERROR PER RECORD ARE PRINTED.
1186 THE TOTAL # OF BYTES IN ERROR PER RECORD IS ALSO PRINTED. A
1187 HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.
1188
1189
1190 3.2 ERROR HALTS
1191 -----
1192
1193 ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION
1194 WITH /FLAG:HOE. THERE ARE NO OTHER HALTS.
1195
1196 4.0 PERFORMANCE REPORT
1197 -----
1198
1199 UNIT X PASS:XXXXX RECORD:XXXXX
1200 BYTES WRITTEN XXX,XXX,XXX,XXX
1201 BYTES READ REV XXX,XXX,XXX,XXX
1202 BYTES READ FWD XXX,XXX,XXX,XXX

	WRT	RDR	RDF
RECOVERABLE ERRORS	XXXXX	XXXXX	XXXXX
UNRECOVERABLE ERRORS	XXXXX	XXXXX	XXXXX

SPEC COND	HARD	FATAL	COMPARE
XXXXX	XXXXX	XXXXX	XXXXX

5.0 TEST SUMMARIES

5.1 TEST 1 -

BASIC FUNCTIONS.

EXECUTES AND VERIFIES CORRECT COMPLETION OF ALL TS05 FUNCTIONS.

SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.

- ♦ SET CHARACTERISTIC 200.
- ♦ DRIVE INITIATE.
- ♦ SET CHARACTERISTIC 20.
- ♦ GET STATUS
- ♦ SET CHARACTERISTIC 40.
- ♦ PRINT TS05 MICROCODE LEVEL (PASS 1 ONLY)

SUBTEST 2 - REWIND.

- ♦ REWIND.
- ♦ REWIND AT BOT.

SUBTEST 3 - WRITE/VERIFY.

- ♦ WRITE/VERIFY PATTERN 1.
- ♦ WRITE/VERIFY PATTERN 2.
- ♦ WRITE/VERIFY PATTERN 3.
- ♦ WRITE/VERIFY PATTERN 4.
- ♦ WRITE/VERIFY PATTERN 5.
- ♦ WRITE/VERIFY PATTERN 6.
- ♦ WRITE/VERIFY PATTERN 0.

SUBTEST 4 - WRITE TAPE MARK, ERASE.

- ♦ WRITE TAPE MARK.
- ♦ WRITE 10 RECORDS
- ♦ ERASE 10 TIMES
- ♦ WRITE TAPE MARK.
- ♦ WRITE TAPE MARK RETRY.

SUBTEST 5 - SPACE FILES.

- ♦ SPACE 2 FILES REVERSE.
- ♦ SPACE 2 FILES FORWARD.
- ♦ SPACE 2 FILES REVERSE.
- ♦ SPACE 2 FILES FORWARD.

SUBTEST 6 - SPACE RECORDS.

- ♦ REWIND.
- ♦ SPACE 7 RECORDS FORWARD.
- ♦ SPACE 7 RECORDS REVERSE.
- ♦ SPACE 7 RECORDS FORWARD.

1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259

1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299

♦ SPACE 7 RECORDS REVERSE.

SUBTEST 7 - WRITE RETRY.

♦ REWIND.
♦ WRITE DATA.
♦ WRITE RETRY.

SUBTEST 8 - READ REV RETRY.

♦ READ REVERSE.
♦ READ NEXT REVERSE.
♦ READ NEXT FORWARD.

SUBTEST 9 - READ FWD RETRY.

♦ READ FORWARD.
♦ READ PREVIOUS FORWARD.
♦ READ PREVIOUS REVERSE.

SUBTEST 10 - CLEAN.

♦ CLEAN.
♦ REWIND.

SUBTEST 11 - WRITE/VERIFY SWAPPED DATA BYTES.

♦ WRITE/VERIFY EVEN LENGTH (RECORD 1).
♦ WRITE/VERIFY ODD LENGTH (RECORD 2).
♦ SET DATA BYTE SWAP.
♦ WRITE/VERIFY EVEN LENGTH (RECORD 3).
♦ WRITE/VERIFY ODD LENGTH (RECORD 4).
♦ CLEAR DATA BYTE SWAP.

SUBTEST 12 - READ SWAPPED DATA BYTES.

♦ READ REV RECORD 4.
♦ READ REV RECORD 3.
♦ SET DATA BYTE SWAP.
♦ READ REV RECORD 2.
♦ READ REV RECORD 1.
♦ READ FWD RECORD 1.
♦ READ FWD RECORD 2.
♦ CLEAR DATA BYTE SWAP.
♦ READ FWD RECORD 3.
♦ READ FWD RECORD 4.

1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357

5.2 TEST 2 - DATA RELIABILITY.

1. THE TAPE IS INITIATED WITH THE FOLLOWING COMMANDS:
SET CHARACTERISTIC 40
REWIND
WRITE 64 RECORDS OF RANDOM LENGTH AND DATA
2. WRITE AND READ COMMANDS ARE SELECTED AT RANDOM AND ARE EXECUTED A RANDOM NUMBER OF TIMES WITH RANDOM LENGTHS AND RANDOM PATTERN UNTIL END OF TAPE IS REACHED.
3. AT THE END OF EACH PASS, A REWIND COMMAND IS ISSUED AND A PERFORMANCE REPORT IS PRINTED.

NOTE: IF A RESTART COMMAND IS USED TO INITIATE TEST 1, THE INITIAL REWIND COMMAND IS NOT ISSUED.

5.3 TEST 3 - WRITE COMPATABILITY/WRITE UTILITY.

REWINDS AND WRITES RECORDS OF RANDOM LENGTHS AND RANDOM DATA FROM BOT TO EOT.

5.4 TEST 4 - READ COMPATABILITY/READ UTILITY.

REWINDS AND READS ENTIRE TAPE, FORWARD AND REVERSE.

5.5 TEST 5 - RANDOM/OPERATOR SELECTED COMMAND SEQUENCE.

A DEFAULT SEQUENCE OF REWIND/WRITE/READ REV/READ FWD/REWIND OF ENTIRE TAPE IS EXECUTED WITH RANDOM PATTERN AND RECORD LENGTH OF 2048 BYTES. OPERATOR CAN ENTER SEQUENCE OF COMMANDS UP TO SEVEN IF THEY DON'T WANT DEFAULT SEQUENCE.

6.0 DEVICE INFORMATION TABLES

6.1 GENERAL

THE TS05 TAPE SUBSYSTEM CONSISTS OF A TSV05 Q-BUS CONTROLLER CONNECTED TO A TS05 DRIVE. FROM A SOFTWARE VIEWPOINT THIS CONFIGURATION IS UNIQUE (FOR A Q-BUS DEVICE) IN A NUMBER OF WAYS:

- A. ONLY ONE REGISTER MAY BE WRITTEN - TSDB (TAPE SYSTEM DATA BUFFER).
- B. TWO REGISTERS MAY BE READ - TSSR AND TSBA (TAPE SYSTEM STATUS REGISTER AND TAPE SYSTEM BUS ADDRESS REGISTER).
- C. COMMANDS ARE NOT WRITTEN TO THE DRIVE; RATHER, COMMAND POINTERS ARE WRITTEN WHICH POINT TO COMMAND PACKETS SOMEWHERE IN CPU MEMORY. THE COMMAND POINTER IS USED BY THE TS05 SUBSYSTEM TO FETCH THE WORD(S) WITHIN THE COMMAND PACKET. THE WORDS WITHIN THE COMMAND PACKET ARE:

1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389

1. COMMAND WORD
 2. LOW ORDER BUFFER ADDRESS
 3. HIGH ORDER BUFFER ADDRESS
 4. BYTE COUNT
- D. THE TSSR CONTAINS ALL THE INFORMATION WHICH WILL BE NECESSARY TO DETERMINE WHETHER:
1. THE DRIVE IS READY TO ACCEPT ANOTHER COMMAND,
 2. THE PREVIOUS COMMAND WAS EXECUTED WITHOUT ERROR.
- IF EITHER OF THE ABOVE CONDITIONS IS UNTRUE AT "JOB DONE" OR "COMMAND INITIATION" TIME, IT MAY BE NECESSARY TO GET THE EXTENDED STATUS REGISTERS TO DETERMINE WHAT ACTION IS TO BE TAKEN AND/OR LOG THE ERROR INFORMATION.
- E. EXTENDED STATUS REGISTERS ARE NOT READ DIRECTLY FROM DRIVE REGISTERS; RATHER, A "GET STATUS" COMMAND IS ISSUED WHICH WILL CAUSE THE TS05 TO TRANSFER EXTENDED STATUS INFORMATION TO THE MEMORY AREA POINTED TO BY THE BUFFER ADDRESS OF THE "GET STATUS" COMMAND. THERE ARE FIVE EXTENDED STATUS REGISTERS. SEE .3.
- F. THE TSDB MUST BE WRITTEN WITH A DATO INSTRUCTION TO PROPERLY WRITE THE COMMAND POINTER. A DATOB WILL CAUSE A MAINTENANCE FUNCTION. A DATO TO THE TSSR WILL CAUSE SUBSYSTEM INIT.
- G. COMMAND PACKETS MUST RESIDE ON DIVIDE BY FOUR MEMORY BOUNDARIES (AS OPPOSED TO DIVIDE BY 2 OR WORD BOUNDARIES) .

1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401

6.2 Q-BUS INTERFACE SPECIFICATIONS

TSV05/ TS05 -----	INT. VECTOR -----	UNIBUS ADDRESS -----	REGISTER -----
FIRST	224	772520 772522	TSBA/TSDB TSSR

6.3 BIT DEFINITIONS FOR TSV05/TS05 REGISTERS

6.3.1 TSV05/TS05 REGISTER SUMMARY

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	
(R/O) TSBA	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00	
(W/O) TSD8	P15	P14	P13	P12	P11	P10	P09	P08	P07	P06	P05	P04	P03	P02	P17	P16	
(R/O) TSSR	SC	0	SCE	RMR	NXM	NBA	A17	A16	SSR	OFL	FC1	FC0	TC2	TC1	TC0	0	
(W/O) TSD8X	BT	0	0	0	P21	P20	P19	P18	(TSD8X EXISTS ONLY WHEN ENABLED BY THE EXTENDED FEATURES SWITCH ON THE M7196)								
XST0	TMK	RLS	LET	RLI	MLE	NEF	ILC	ILA	MOT	ONL	IE	VCK	PED	WLK	BOT	EOT	
XST1	DLT	0	COR	0	0	0	0	RBP	0	0	0	0	0	0	UNC	0	
XST2	OPM	RCE	0	0	0	WCF	0	0	RL7	RL6	RL5	RL4	RL3	RL2	RL1	RL0	
XST3	MICRO DIAGNOSTIC ERROR CODE									0	OPI	REV	TRF	DCK	0	0	RIB
XST4	HSP	RCE	0	0	0	0	0	0	WRITE RETRY COUNT								

TERMINATION CLASS CODES (TSSR TC0-TC2):

- 0 - NORMAL TERMINATION
- 1 - ATTENTION CONDITION
- 2 - TAPE STATUS ALERT
- 3 - FUNCTION REJECT
- 4 - RECOVERABLE ERROR - TAPE POSITION - ONE RECORD
DOWN TAPE FROM START OF FUNCTION
- 5 - RECOVERABLE ERROR - TAPE NOT MOVED
- 6 - UNRECOVERABLE ERROR - TAPE POSITION LOST
- 7 - FATAL CONTROLLER ERROR

FATAL CLASS CODES (TSSR FC0-FC1):

- 0 - MICRO DIAGNOSTIC FAILURE. SEE ERROR CODE BYTE (XST3) FOR FAILED FUNCTION.
- 1 - RESERVED
- 2 - NOT USED
- 3 - RESERVED FOR FUTURE USE ALWAYS READ AS A 0

6.3.2 TSV05 STATUS REGISTER (TSSR)

Q-BUS ADDRESS + 2 - READ ONLY

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
!SC!	!0!	!SCE!	!RMR!	!NXM!	!NBA!	!A17!	!A16!	!SSR!	!OFL!	!FC1!	!FC0!	!TC2!	!TC1!	!TC0!	!0!

BIT	NAME	TCC	DEFINITION
15	SC	S	SPECIAL CONDITION. WHEN SET, INDICATES THAT THE LAST COMMAND DID NOT COMPLETE WITHOUT INCIDENT. SPECIFICALLY, EITHER AN ERROR WAS DETECTED OR AN EXCEPTION CONDITION OCCURRED. EXCEPTION CONDITIONS CAN BE TAPE MARKS ON READ COMMANDS, REVERSE MOTION AND AT BOT, EOT WHILE WRITING, ETC. MAY ALSO BE SET BY THE ERROR BITS CONTAINED IN THE TSSR REGISTER: SCE, RMR, AND NXM. THE TERMINATION CLASS BITS ARE SOMETHING OTHER THAN 0 (UNLESS RMR IS THE ONLY ERROR - SEE RMR).
14	-	-	RESERVED (ALWAYS A 0)
13	SCE	FC0	SANITY CHECK ERROR-SETS WHEN THE CONTROLLER DETECTS AN ABNORMAL CONDITION WITHIN ITSELF DURING EXECUTION OF IT'S FUNCTIONS AND THE PROBLEM IS SERIOUS ENOUGH THAT A MESSAGE PACKET IS NOT STORED.
12	RMR	S	REGISTER MODIFICATION REFUSED. SET BY THE TSV05 WHEN A COMMAND POINTER IS LOADED INTO TSOB AND SUB-SYSTEM READY (SSR) IS NOT SET. NOTE THAT THIS BIT CAUSES SPECIAL CONDITION BUT NO TERMINATION CLASS (IN FACT, THE TSO5 NEVER SEES THIS ERROR) BECAUSE ON A SYSTEM WITH NO BUGS, THIS BIT MAY COME UP ON AN ATTENTION MESSAGE. IF ATTNS ARE NOT ENABLED, THIS BIT COMING UP IS AN INDICATION OF EITHER A FATAL CONTROLLER ERROR OR A SOFTWARE BUG.
11	NXM	4/5	NON-EXISTENT MEMORY. SET BY THE TSV05 WHEN TRYING TO TRANSFER TO OR FROM A MEMORY LOCATION WHICH DOES NOT EXIST. MAY OCCUR WHEN FETCHING THE COMMAND PACKET, FETCHING OR STORING DATA, OR STORING THE MESSAGE PACKET.
10	NBA	S	NEED BUFFER ADDRESS. WHEN SET, INDICATES THAT THE TSO5 NEEDS A MESSAGE BUFFER ADDRESS. THIS BIT IS CLEARED DURING THE SET CHARACTERISTICS

1518				COMMAND (IF A GOOD ADDRESS WAS GIVEN).
1519				
1520	09	A17	S	BUS ADDRESS BIT 17. A17 AND A16 (BIT 08) TRACK
1521				THE VALUES OF BITS 17 AND 16 OF THE TSBA
1522				REGISTER. LOADED FROM TSDB BITS 01-00 WHEN TSDB
1523				IS WRITTEN.
1524				
1525	08	A16	S	BUS ADDRESS BIT 16. SEE A17 (BIT 09).
1526				
1527	07	SSR	S	SUB-SYSTEM READY. WHEN SET, INDICATES THAT THE
1528				TSV05/TS05 SUBSYSTEM IS NOT BUSY AND IS READY TO
1529				ACCEPT A NEW COMMAND POINTER.
1530				
1531	06	OFL	S,1,3	OFF-LINE. WHEN SET, INDICATES THAT THE TS05 IS
1532				OFF-LINE AND UNAVAILABLE FOR ANY TAPE MOTION
1533				COMMANDS. THIS BIT CAN CAUSE A TERMINATION CLASS
1534				OF 1 (ON ATTN INTERRUPT) OR 3 (RESULTS IN NEF).
1535				
1536	05	FC1	7	FATAL TERMINATION CLASS 01. FC1 AND FC0 (BIT
1537				04) ARE USED TO INDICATE THE TYPE OF FATAL
1538				ERROR WHICH HAS OCCURRED ON THE TS05. THESE
1539				BITS ARE VALID ONLY WHEN SC IS SET AND THE
1540				TERMINATION CLASS CODE BITS ARE ALL SET (111).
1541				
1542	04	FC0	7	FATAL TERMINATION CLASS 00. SEE FC1 (BIT 05).
1543				
1544	03	TC2	S	TERMINATION CLASS BIT 02. THIS BIT, ALONG WITH
1545				THE TC1 AND TC0 BITS, ACT AS AN OFFSET VALUE
1546				WHENEVER AN ERROR OR EXCEPTION CONDITION OCCURS
1547				ON A COMMAND. EACH OF THE EIGHT POSSIBLE
1548				VALUES OF THIS FIELD REPRESENT A PARTICULAR
1549				CLASS OF ERRORS OR EXCEPTIONS. THE CONDITIONS
1550				IN EACH CLASS HAVE SIMILAR SIGNIFICANCE AND, AS
1551				APPLICABLE, RECOVERY PROCEDURES. THE CODE
1552				PROVIDED IN THIS FIELD IS EXPECTED TO BE
1553				UTILIZED AS AN OFFSET INTO A DISPATCH TABLE FOR
1554				HANDLING OF THE CONDITION.
1555				
1556	02	TC1	S	TERMINATION CLASS BIT 01. SEE TC2 (BIT 03).
1557				
1558	01	TC0	S	TERMINATION CLASS BIT 00. SEE TC2 (BIT 03).
1559				
1560	00	-	-	NOT USED. (ALWAYS A 0)
1561				
1562				
1563				
1564				
1565				
1566				
1567				

Q-BUS ADDRESS + 2 - WRITE ONLY

SUBSYSTEM INITIALIZE

6.3.2.1 TSV05 EXTENDED DATA BUFFER REGISTER (TSDBX)

```

      +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+
      |BT| |0| |0| |0| |P21|P20|P19| |P18|
TSDBX +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+

```

(TSDBX EXISTS ONLY WHEN
ENABLED BY THE EXTENDED
FEATURES SWITCH ON THE M7196)

BIT	NAME	TCC	DEFINITION
15	BT	-	BOOT COMMAND BIT. WHEN WRITTEN TO A 1, WITH SSR=1, CAUSES THE TAPE TO BE REWOUND TO BOT, THE FIRST TAPE RECORD TO BE SKIPPED, AND THE SECOND RECORD TO BE LOADED INTO CPU MEMORY SPACE STARTING AT LOCATION 0.
14-12			RESERVED (ALWAYS A 0)
11-08	P<21:18>		COMMAND POINTER BITS 21-18. WHEN THE TSDBX IS WRITTEN AND SSR=1, THE DATA IS LOADED INTO BITS 21-18 OF THE INTERNAL TSBA REGISTER.
07-00			RESERVED (ALWAYS A 0)

6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)

```

      15  14  13  12  11  10  09  08  07  06  05  04  03  02  01  00
      +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+
      |TMK| |RLS| |LET| |RL| |WLE| |NEF| |ILC| |ILA| |MOT| |ONL| |IE| |VCK| |PED| |WLK| |BOT| |EOT|
      +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+ +---+

```

BIT	NAME	TCC	DEFINITION
15	TMK	S,2	TAPE MARK DETECTED. SET WHENEVER A TAPE MARK WAS DETECTED DURING A READ, SPACE, OR SKIP COMMAND AND AS A RESULT OF THE WRITE TAPE MARK OR WITE TAPE MARK RETRY COMMANDS.
14	RLS	2	RECORD LENGTH SHORT. THIS BIT INDICATES THAT EITHER THE RECORD'S LENGTH WAS SHORTER THAN THE BYTE COUNT ON READ OPERATIONS, A SPACE RECORD OPERATION ENCOUNTERED A TAPE MARK OR BOT BEFORE THE POSITION COUNT WAS EXHAUSTED, OR A SKIP TAPE MARKS COMMAND WAS TERMINATED BY ENCOUNTERING BOT OR A DOUBLE TAPE MARK (IF THAT OPERATIONAL MODE IS ENABLED, SEE LET) PRIOR TO EXHAUSTING THE POSITION COUNTER.

1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651

13 LET 2

LOGICAL END OF TAPE. SET ONLY ON THE SKIP TAPE MARKS COMMAND WHEN EITHER TWO CONTIGUOUS TAPE MARKS ARE DETECTED OR WHEN MOVING OFF OF BOT AND THE FIRST RECORD ENCOUNTERED IS A TAPE MARK. THE SETTING OF THIS BIT WILL NOT OCCUR UNLESS THIS MODE OF TERMINATION IS ENABLED THROUGH USE OF THE SET CHARACTERISTICS COMMAND.

12 RLL 2

RECORD LENGTH LONG. WHEN SET, THIS BIT INDICATES THAT THE RECORD READ WAS LONGER THAN THE BYTE COUNT SPECIFIED.

11 WLE 3,6

WRITE LOCK ERROR. WHEN SET, INDICATES THAT A WRITE OPERATION WAS ISSUED BUT THE MOUNTED TAPE DID NOT CONTAIN A WRITE ENABLE RING OR THE WRT LOCK SWITCH ACTIVATED DURING THE OPERATION.

10 NEF 3

NON-EXECUTABLE FUNCTION. WHEN SET, INDICATES THAT THE COMMAND COULD NOT BE EXECUTED DUE TO ONE OF THE FOLLOWING CONDITIONS:

- THE COMMAND SPECIFIED REVERSE TAPE DIRECTION BUT THE TAPE WAS ALREADY POSITIONED AT BOT.
- THE ISSUING OF ANY MOTION COMMAND EXCEPT

1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701

WHEN THE VOLUME CHECK BIT IS SET.
- ANY COMMAND, EXCEPT GET STATUS OR DRIVE INITIALIZE, WHEN THE TS05 IS OFF-LINE.
- ANY WRITE COMMAND WHEN THE TAPE DOES NOT CONTAIN A WRITE ENABLE RING (WRITE LOCK STATUS - WLS).

09	ILC	3	ILLEGAL COMMAND. SET WHEN A COMMAND IS ISSUED AND EITHER ITS COMMAND FIELD OR ITS COMMAND MODE FIELD CONTAINS CODES WHICH ARE NOT SUPPORTED BY THE TS05.
08	ILA	3	ILLEGAL ADDRESS. (MORE THAN 18 BITS OR ODD WHEN AN EVEN ADDRESS IS REQUIRED.)
07	MOT	S	TAPE IS MOVING.
06	ONL	S	ON LINE. WHEN SET, INDICATES THAT THE TS05 IS ON-LINE AND OPERABLE.
05	IE	S	INTERRUPT ENABLE. REFLECTS THE STATE OF THE INTERRUPT ENABLE BIT SUPPLIED ON THE LAST COMMAND.
04	VCK	S	VOLUME CHECK. WHEN SET, INDICATES THAT THE DRIVE HAS BEEN EITHER POWERED DOWN OR TURNED OFF-LINE. CLEARED BY THE CLEAR VOLUME CHECK (CVC) BIT IN THE COMMAND HEADER WORD. THIS BIT CAN CAUSE A TERMINATION CLASS OF 3.
03	PED	S	PHASE ENCODED DRIVE. ALWAYS SET, INDICATES THAT THE TS05 IS CAPABLE OF READING AND WRITING ONLY 1600 BPI PHASE ENCODED DATA.
02	WLK	S,3	WRITE LOCKED. WHEN SET, INDICATES THAT THE MOUNTED REEL OF TAPE DOES NOT HAVE A WRITE-ENABLE RING INSTALLED. THE TAPE IS, THEREFORE, WRITE PROTECTED.
01	BOT	S,3	BEGINNING OF TAPE. WHEN SET, INDICATES THAT THE TAPE IS POSITIONED AT THE LOAD POINT AS DENOTED BY THE BOT REFLECTIVE STRIP ON THE TAPE.
00	EOT	S,2	END OF TAPE. THIS BIT IS SET WHENEVER THE TAPE IS POSITIONED AT OR BEYOND THE END OF TAPE REFLECTIVE STRIP.

6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1)

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DLT	0	COR	0	0	0	0	RBP	0	0	0	0	0	0	UNC	0

BIT	NAME	TCC	DEFINITION
15	DLT	4	DATA LATE. SET WHEN THE FIFO IS FULL ON A READ OR EMPTY ON A WRITE. THESE CONDITIONS OCCUR WHENEVER THE Q-BUS LATENCY EXCEEDS THE DATA TRANSFER RATE OF THE TLOS.
14	-	-	NOT USED. (ALWAYS A 0)
13	COR	5	CORRECTABLE DATA. CORRECTABLE DATA ERROR HAS BEEN ENCOUNTERED.
12-09			RESERVED (ALWAYS A 0)
08	RBP	4	READ BUS PARITY ERROR. SET WHEN CONTROLLER DETECTS A PARITY ERROR ON THE READ DATA LINES OF THE TRANSPORT BUS.
07-02 & 00			RESERVED (ALWAYS A 0)
01	UNC	4	UNCORRECTABLE DATA ERROR.

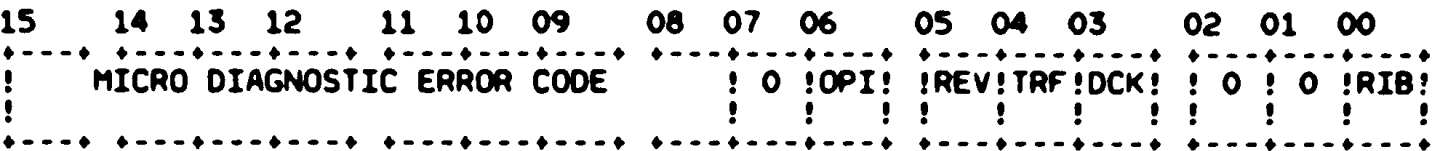
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770

6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
OPM	RCE	0	0	0	WCF	0	0	RL7	RL6	RL5	RL4	RL3	RL2	RL1	RL0

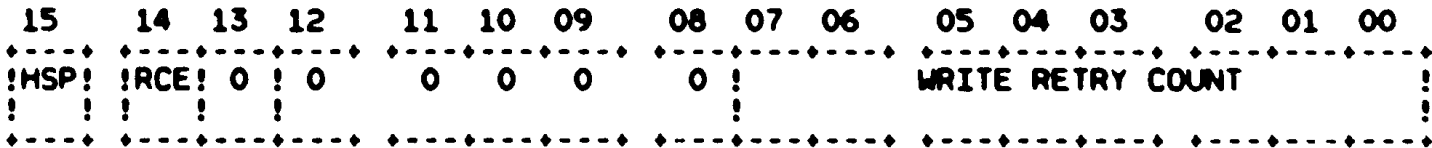
BIT	NAME	TCC	DEFINITION
---	----	---	-----
15	OPM	S	OPERATION IN PROGRESS. (TAPE MOVING)
14	RCE	7,F2	RAM CHECKSUM ERROR. CAUSES FATAL CLASS 2 BECAUSE THE ERROR MIGHT HAVE OCCURRED DURING THE TRANSMISSION OF THE MESSAGE PACKET.
13-11			RESERVED (ALWAYS A 0)
10	WCF	7	WRITE CLOCK FAILURE. SET DURING A WRITE TO INDICATE THAT THE FIFO IS NOT BEING EMPTIED BY THE TRANSPORT.
09-08			RESERVED (ALWAYS A 0)
07-00	RL	-	REVISION LEVEL.
	7-0		

6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3)



BIT	NAME	TCC	DEFINITION
15 TO 08			MICRO DIAGNOSTIC ERROR CODE. (SEE LIST OF CODES BELOW).
07			RESERVED (ALWAYS A 0)
06	OPI	6	OPERATION INCOMPLETE. SET WHEN A READ, SPACE, OR SKIP OPERATION HAS MOVED 25 FEET OF TAPE WITHOUT DETECTING ANY DATA ON THE TAPE.
05	REV	S	DIRECTION OF CURRENT OPERATION WAS REVERSE (BUT IS 0 IF REWIND OR FORWARD)
04	-	-	RESERVED (ALWAYS A 0)
03	DCK	S,6	DENSITY CHECK. SET WHEN A PE IDENTIFICATION BURST (IDB) WAS NOT DETECTED WHEN MOVING OFF OF BOT.
02-01			RESERVED (ALWAYS A 0)
00	RIB	2	REVERSE INTO BOT. A READ, SPACE, OR SKIP COMMAND ALREADY IN PROGRESS HAS ENCOUNTERED THE BOT MARKER WHEN MOVING TAPE IN THE REVERSE DIRECTION. TAPE MOTION WILL BE HALTED AT BOT.

6.3.7 EXTENDED STATUS REGISTER 4 (XSTAT4)



BIT	NAME	TCC	DEFINITION
-----	------	-----	------------

1829	15	HSP	S	HIGH SPEED. WHEN SET, INDICATES THAT THE TRANSPORT IS
1830				OPERATING IN HIGH SPEED MODE.(100IPS) WHEN CLEAR, THE
1831				TRANSPORT IS OPERATING IN LOW SPEED MODE.(25IPS)
1832				
1833	14	RCE	6	RETRY COUNT EXCEEDED. WHEN SET, INDICATES THAT THE CONTROLLER
1834				WAS BUFFERING WRITE DATA AND COULD NOT SUCCESSFULLY OUTPUT
1835				THE BUFFERED RECORD WITHIN THE SPECIFIED NUMBER OF RETRIES.
1836				CAUSES TAPE POSITION LOST TERMINATION.
1837				
1838	13-8		-	RESERVED (ALWAYS A 0)
1839				
1840	7-0	WRC	S	WRITE RECOUNT COUNT STATISTIC. THIS FIELD INDICATES, WHEN
1841				THE CONTROLLER IS BUFFERING WRITE DATA RECORDS, THE TOTAL
1842				NUMBER OF CONTROLLER INITIATED RETRIES PERFORMED IN ORDER
1843				TO WRITE THE PREVIOUS BUFFERED RECORD. THIS COUNT IS CLEARED
1844				AFTER IT IS DISPLAYED.
1845				
1846				
1847				
1848				
1849				
1850				
1851				
1852				
1853				
1854				
1855				
1856				
1857				
1858				
1859				
1860				
1861				

7.0 DIAGNOSTIC HISTORY

- REVISION A - MAR 1982
- MODIFIED CZTSHC FROM TS11 FOR TSV05
REVISION B - APR 1983
- UPDATED THE DIAGNOSTIC TO SUPPLY THE CORRECT
RECORD NUMBER DURING EXECUTION OF TEST #2.
REF: CHMIELECKI TO MITCHELL "TSV05 DATA
RELIABILITY PROBLEM"; 21-JAN-83.

```

1
13      .TITLE PROGRAM HEADER AND TABLES
14      .SBTTL PROGRAM HEADER
43
45 000000      .ENABL ABS,AMA
46      .      2000
48 002000      BGNMOD
49
50
51      ;**
52      ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
53      ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
54      ;--
55 002000      POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU
56
64
65 002000      HEADER CVTSE,B,0,5000,1
002000      L$NAME::      ;DIAGNOSTIC NAME
002000      103
002001      126
002002      124
002003      123
002004      105
002005      000
002006      000
002007      000
002010      L$REV::      ;REVISION LEVEL
002010      102
002011      L$DEPO::      ;0
002011      060
002012      L$UNIT::      ;NUMBER OF UNITS
002012      000000
002014      L$TIML::      ;LONGEST TEST TIME
002014      005000
002016      L$HPCP::      ;POINTER TO H.W. QUES.
002016      027754
002020      L$SPCP::      ;POINTER TO S.W. QUES.
002020      030062
002022      L$HPTP::      ;PTR. TO DEF. H.W. PTABLE
002022      002174
002024      L$SPTP::      ;PTR. TO S.W. PTABLE
002024      002204
002026      L$LADP::      ;DIAG. END ADDRESS
002026      032004
002030      L$STA::      ;RESERVED FOR APT STATS
002030      000000
002032      L$CO::
002032      000000
002034      L$DTYP::      ;DIAGNOSTIC TYPE
002034      000001
002036      L$APT::      ;APT EXPANSION
002036      000000
002040      L$DTP::      ;PTR. TO DISPATCH TABLE
002040      002124
002042      L$PRIO::      ;DIAGNOSTIC RUN PRIORITY
002042      000000
002044      L$ENVI::      ;FLAGS DESCRIBE HOW IT WAS SETUP

```

.ASCII	/C/
.ASCII	/V/
.ASCII	/T/
.ASCII	/S/
.ASCII	/E/
.BYTE	0
.BYTE	0
.BYTE	0
.ASCII	/B/
.ASCII	/O/
.WORD	0
.WORD	5000
.WORD	L\$HARD
.WORD	L\$SOFT
.WORD	L\$HW
.WORD	L\$SW
.WORD	L\$LAST
.WORD	0
.WORD	0
.WORD	1
.WORD	0
.WORD	L\$DISPATCH
.WORD	0

PROGRAM HEADER AND TABLES
PROGRAM HEADER

MACRO M1113 30-NOV-83 10:17

SEQ 040

002044	000000			.WORD	0
002046		L\$EXP1::	;EXPANSION WORD	.WORD	0
002046	000000			.WORD	0
002050		L\$HREV::	;SVC REV AND EDIT #	.BYTE	C\$REVISION
002050	003			.BYTE	C\$EDIT
002051	003				
002052		L\$EF::	;DIAG. EVENT FLAG	.WORD	0
002052	000000			.WORD	0
002054	000000				
002056		L\$SPC::		.WORD	0
002056	000000			.WORD	0
002060		L\$DEVP::	; POINTER TO DEVICE TYPE LIST	.WORD	L\$DVTYP
002060	002164			.WORD	L\$RPT
002062		L\$REPP::	;PTR. TO REPORT CODE	.WORD	0
002062	017570			.WORD	0
002064		L\$EXP4::		.WORD	0
002064	000000			.WORD	0
002066		L\$EXP5::		.WORD	0
002066	000000			.WORD	0
002070		L\$AUT::	;PTR. TO ADD UNIT CODE	.WORD	L\$AU
002070	024052			.WORD	L\$DU
002072		L\$DUT::	;PTR. TO DROP UNIT CODE	.WORD	0
002072	024000			.WORD	L\$DESC
002074		L\$LUN::	;LUN FOR EXERCISERS TO FILL	.WORD	0
002074	000000			.WORD	L\$DESC
002076		L\$DESP::	;POINTER TO DIAG. DESCRIPTION	.WORD	L\$LOAD
002076	002136			.WORD	0
002100		L\$LOAD::	;GENERATE SPECIAL AUTOLOAD EMT	EMT	E\$LOAD
002100	104035			.WORD	0
002102		L\$ETP::	;POINTER TO ERR_TBL	.WORD	0
002102	000000			.WORD	L\$INIT
002104		L\$ICP::	;PTR. TO INIT CODE	.WORD	L\$CLEAN
002104	021324			.WORD	L\$AUTO
002106		L\$CCP::	;PTR. TO CLEAN-UP CODE	.WORD	L\$PROT
002106	023736			.WORD	0
002110		L\$ACP::	;PTR. TO AUTO CODE	.WORD	0
002110	023314			.WORD	0
002112		L\$PRT::	;PTR. TO PROTECT TABLE	.WORD	0
002112	021316			.WORD	0
002114		L\$TEST::	;TEST NUMBER	.WORD	0
002114	000000			.WORD	0
002116		L\$DLY::	;DELAY COUNT	.WORD	0
002116	000000			.WORD	0
002120		L\$HIME::	;PTR. TO HIGH MEM	.WORD	0
002120	000000			.WORD	0
66					
72		.SBTTL	DISPATCH TABLE		
73					
74		;			
75		;	THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.		
76		;	IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.		
77		;			
78					
79	002122		DISPATCH 5	.WORD	5
	002122	000005		.WORD	T1
	002124		L\$DISPATCH::		
	002124	024156			

PROGRAM HEADER AND TABLES
DISPATCH TABLE

MACRO M1113 30-NOV-83 10:17

SEQ 041

	002126	025644				.WORD	T2
	002130	026500				.WORD	T3
	002132	026674				.WORD	T4
	002134	027054				.WORD	T5
80							
87							
88							
89							
90							
91							
T							
92							
93							
94	002136						
	002136						
ST/	002136	104	101	124		.ASCIZ	/DATA RELIABILITY TE
	002141	101	040	122			
	002144	105	114	111			
	002147	101	102	111			
	002152	114	111	124			
	002155	131	040	124			
	002160	105	123	124			
	002163	000					
95	002164					.EVEN	
	002164						
	002164	124	123	126		.ASCIZ	/TSV05/
	002167	060	065	000		.EVEN	
96							
97							
98							
99							
100							
101							
102							
103							
104	002172						
	002172	000003				.WORD	L10000-L1MM/2
	002174						
	002174						
105							
10							
107	002174	172520			.WORD	172520	;TSDB ADDRESS.
108	002176	000224			.WORD	224	;VECTOR ADDRESS.
109	002200	000000			.WORD	0	;DRIVE #0 FOR DEFAULT
110							
111	002202						
	002202						
112							
113							
114							
115							
116							
117							
118							
119	002202						
	002202	000051				.WORD	L10001-L1SW/2

002204
002204
120
127 002204 001
128 002205 000
129 002206 000
130 002207 000
131 002210 001
132 002211 000
133 002212 000
134 002213 000
135 002214 000
136 002215 000
137 002216 000
138 002217 000
139 002220 000040
140 002222 000015
141 002224 000001
142 002226 000001
143 002230 000007
144 002232 000004
145 002234 004000
146 002236 076400
147 002240 000007
148 002242 000003
149 002244 004000
150 002246 076400
151 002250 000007
152 002252 000002
153 002254 004000
154 002256 076400
155 002260 000007
156 002262 000015
157 002264 000001
158 002266 000001
159 002270 000007
160 002272 000033
161 002274 004000
162 002276 076400
163 002300 000007
164 002302 000033
165 002304 004000
166 002306 076400
167 002310 000007
168 002312 000001
169 002314 000000
170 002316 000000
171 002320 000000
172 002322 000000
173 002324 000000
174
175 002326
002326
176
177 002326

L1SW::
SFPTBL::
CLRFLG:: .BYTE 1
RRANV:: .BYTE 0
MAE:: .BYTE 0
ERCVER:: .BYTE 0
BADTSW:: .BYTE 1
DINT:: .BYTE 0
IREC:: .BYTE 0
CHGFLG:: .BYTE 0
PIRE:: .BYTE 0
CHAR:: CH.EAI
CHDD:: .WORD 13.
RAMP
DATCNT
32000.
RAMP
3
DATCNT
32000.
RAMP
2
DATCNT
32000.
RAMP
13.
1
RAMP
27.
DATCNT
32000.
RAMP
27.
DATCNT
32000.
RAMP
1
RDBUF:: .WORD 0
WTBUF:: .WORD 0
HSSW:: .WORD 0
EXTFEA:: .WORD 0
BENBSW:: .WORD 0
ENDSW
L10001:
ENDMOD

!CLEAR COUNTERS FLAG.
!RESET RANDOM VARIABLES EACH PASS FLAG.
!HALT AFTER EACH COMMAND FLAG.
!ENABLE RECOVERABLE ERROR PRINTS FLAG.
!BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETECT BAD TAPE
!SPARE
!DISABLE INTERRUPTS FLAG.
!INHIBIT ERROR RECOVERY FLAG.
!CHANGE CMD SEQ TABLE FLAG.
!SPARE.
!INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
!SPARE.
!CHARACTERISTICS CODE (DEFAULT = 40).
!COMMAND 2 (DEFAULT = REWIND).
!BYTE COUNT
!NUMBER OF OPERATIONS
!PATTERN
!COMMAND 3 (DEFAULT = WRITE)
!BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
!NUMBER OF OPERATIONS (DEFAULT = 32000).
!PATTERN (DEFAULT = RANDOM).
!COMMAND 4 (DEFAULT = READ REV).
!BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
!NUMBER OF OPERATIONS (DEFAULT = 32,000).
!PATTERN (DEFAULT = RANDOM).
!COMMAND 5 (DEFAULT = READ FWD).
!BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
!NUMBER OF OPERATIONS (DEFAULT = 32,000).
!PATTERN (DEFAULT = RANDOM).
!COMMAND 6 (DEFAULT = REWIND).
!BYTE COUNT
!NUMBER OF OPERATIONS
!PATTERN
!END OF CMD SEQ TABLE CODE (DEF) OR CMD 7
!BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
!NUMBER OF OPERATIONS (DEFAULT = 32000).
!PATTERN (DEFAULT = RANDOM).
!END OF CMD SEQ TABLE CODE (DEF) OR CMD 8
!BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
!NUMBER OF OPERATIONS (DEFAULT = 32000).
!PATTERN (DEFAULT = RANDOM).
!DEFAULT SWITCH SETTING
!ENABLE READ BUFFERING
!ENABLE WRITE BUFFERING
!RUN AT 100IPS SWITCH
!EXTENDED FEATURES SOFTWARE SW 0=OFF,1=ON
!BUFFER ENABLE SOFTWARE SW 0=OFF,1=ON

PROGRAM HEADER AND TABLES
SOFTWARE P-TABLE

MACRO M1113 30-NOV-83 10:17

E4

SEQ 043

190
191
192
201
202 002326
203
204
205
206
207
208
209 002326

.TITLE GLOBAL AREAS
.SBTTL GLOBAL EQUATES SECTION

BGNMOD

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

EQUALS

! 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

000340	PRI07--	340
000300	PRI06--	300

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL EQUATES SECTION

SEQ 044

```

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

```

210
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250

REGISTER USAGE.

```

      ;
      ; R0 - PASSES PARAMETERS TO/FROM DIAGNOSTIC SUPERVISOR.
      ; R1 - COMMAND SEQUENCE TABLE POINTER.
      ; R2 - GENERAL PURPOSE REGISTER.
      ; R3 - GENERAL PURPOSE REGISTER.
      ; R4 - GENERAL PURPOSE REGISTER.
      ; R5 - CURRENT LOGICAL DEVICE NUMBER X 2.
      ; R6 - STACK POINTER.
      ; R7 - PROGRAM COUNTER.

```

THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.

```

100000      TS.SC==100000      ;SPECIAL CONDITION BIT.
040000      TS.UPE==40000      ;UNIBUS PARITY ERROR
020000      TS.SPE==20000      ;SERIAL BUS PARITY ERROR.
010000      TS.RMR==10000      ;REGISTER MODIFICATION REFUSED.
004000      TS.NXM==4000       ;NON-EXISTENT MEMORY.
002000      TS.NBA==2000       ;NEED BUFFER ADDRESS.
001000      TS.A17==1000       ;BUS ADDRESS BIT 17.
000400      TS.A16==400        ;BUS ADDRESS BIT 16.
000200      TS.SSR==200        ;UNIT READY BIT.
000100      TS.OFL==100        ;OFF LINE.
177717      TSC.FCC==177717    ;FATAL CLASS CODE MASK.
177761      TSC.TCC==177761    ;TERMINATION CLASS CODE MASK.

```

THE FOLLOWING ARE BIT DEFINITIONS FOR THE COMMAND WORD

```

100000      ACK.C==100000      ;ACKNOWLEDGE BIT
040000      CVC.C==40000       ;CLEAR VOLUME CHECK.
020000      OPP.C==20000       ;OPPOSITE BIT
010000      SWB.C==10000       ;SWAP BYTE BIT

```

GLOBAL AREAS
GLOBAL EQUATES SECTION

SEQMACRO M1113 30-NOV-83 10:17

SEQ 045

251	004000	MOD.C3==4000	;MODE BIT 3
252	004000	BRF.C==4000	;BYTE/RECORD/FILE COUNT FLAG BIT. NOT USED
253			;BY TS05 BUT USED INTERNALLY BY THIS PROGRAM ONLY.
254	002000	MOD.C2==2000	;MODE BIT 2
255	001000	MOD.C1==1000	;MODE BIT 1
256	000400	MOD.C0==400	;MODE BIT 0
257	000200	IE.C==200	;INTERRUPT ENABLE
258	000100	FMT.C1==100	;FORMAT BIT 1
259	000100	VFY.C==100	;WRITE VERIFY FLAG BIT. INTERNAL USE ONLY.
260			;NOT USED BY TS05.
261	000040	FMT.C0==40	;FORMAT BIT 0.
262	000040	JMP.C==40	;JUMP BIT-TO DIRECT THIS PROGRAM TO JUMP TO
263			;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
264			;TABLE. INTERNAL USE ONLY.
265	000020	CMD.C4==20	;COMMAND BIT 4
266	000020	DLY.C==20	;INSERT DELAY. INTERNAL USE ONLY.
267	000010	CMD.C3==10	;COMMAND BIT 3
268	000004	CMD.C2==4	;COMMAND BIT 2
269	000002	CMD.C1==2	;COMMAND BIT 1
270	000001	CMD.C0==1	;COMMAND BIT 0
271			
272			;BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.
273			
274	000200	CH.ESS==200	;ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT).
275	000040	CH.EAI==40	;ENABLE ATTENTION INTERRUPTS.
276	000020	CH.ERI==20	;ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
277	000040	DFTSCH==CH.EAI	;DEFAULT CHARACTERISTICS CODE.
278			
279			;BIT DEFINITIONS FOR EXTENDED CONTROL WORD
280			
281	000040	EF.HSS==40	;ENABLE HIGH SPEED SELECT
282	000030	EF.RWB==30	;ENABLE BOTH READ & WRITE BUFFERING
283	000020	EF.RBO==20	;ENABLE READ BUFFERING ONLY
284			
285			;THE FOLLOWING INDICATES THE RELATIVE POSITIONS OF THE STATUS WORDS
286			;IN THE MESSAGE BUFFER.
287			
288	000004	MS.RFC==4	;RESIDUAL FRAME COUNT.
289	000006	MS.XS0==6	;EXT STATUS REG 0
290	000010	MS.XS1==10	;EXT STATUS REG 1
291	000012	MS.XS2==12	;EXT STATUS REG 2
292	000014	MS.XS3==14	;EXT STATUS REG 3
293	000016	MS.XS4==16	;EXT STATUS REG 4
294			
295			;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0.
296			
297	100000	X0.TMK==100000	;TAPE MARK.
298	040000	X0.RLS==40000	;RECORD LENGTH SHORT.
299	020000	X0.LET==20000	;LOGICAL EOT.
300	010000	X0.RLL==10000	;RECORD LENGTH LONG.
301	000100	X0.ONL==100	;ON LINE BIT.
302	000004	X0.WLK==4	;WRITE LOCK BIT
303	000002	X0.BOT==2	;BOT BIT.
304	000001	X0.EOT==1	;EOT BIT.
305			
306			;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2.
307			

GLOBAL AREAS
GLOBAL EQUATES SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 046

```

308      100000      X2.OPM==100000      ;OPERATION IN PROGRESS, TAPE MOVING
309      000200      X2.EFE==200        ;EXTENDED FEATURES ENABLED
310      000100      X2.BFE==100        ;BUFFERING ENABLED
311
312      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3.
313
314      000010      X3.DCK==10          ;DENSITY CHECK.
315      157400      X3.RNY==157400      ;CAPSTAN RUNAWAY UDIAG ERROR CODE.
316
317      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4.
318
319      100000      X4.HSS==100000      ;HIGH SPEED SWITCH INDICATING 100IPS
320      040000      X4.RCE== 40000      ;RETRY COUNT EXCEEDED
321
322      ;THE FOLLOWING DEFINITIONS SHOW THE RELATIVE POSITIONS OF THE COMMAND
323      ;PACKET ENTRIES.
324
325
326      000000      CP.CMD==0           ;CMDPKT.0==TS05 COMMAND.
327      000002      CP.ADL==2          ;CMDPKT.2==BUFFER ADDRESS LOW.
328      000004      CP.ADH==4          ;CMDPKT.4==BUFFER ADDRESS HIGH.
329      000006      CP.CNT==6          ;CKDPKT.6==BYTE/FILE/RECORD COUNT
330
331      ;MISCELLANEOUS DEFINITIONS.
332
333      000340      INTPRI==PRI07        ;PRIORITY TO BE USED IN INTERRUPT STATE.
334      000012      SCHCNT==12          ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
335      ;BUFFER LENGTH. (EVEN #)
336      000020      MSGCNT==20          ;MESSAGE BUFFER LENGTH IN BYTES. (EVEN #)
337      000020      DIACNT==20          ;DIAGNOSTIC COMMAND BUFFER EXTENT.
338      004000      DATCNT==2048.        ;MAXIMUM RECORD LENGTH IN BYTES.
339      ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSURE
340      ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPER.
341      177740      RNOPSC==177740      ;RANDOM # OF OPERATIONS MASK.
342      000007      RANP==7             ;CODE TO SELECT RANDOM PATTERN.
343      000020      RRECL==16.          ;READ RECOVERY ATTEMPT LIMIT.
344      000020      WRECL==16.          ;WRITE RECOVERY ATTEMPT LIMIT.
345      153624      RANBC==153624      ;CONSTANT USED TO RESET RANDOM # GENERATOR BASE.
346      032561      RANSC==32561        ;CONSTANT USED TO RESET RANDOM # SAVE LOCATION.
347      177774      NINUSE==177774     ;NOT IN USE CODE FOR DEVICE STATE TABLE.
348      177740      NCMD.C==ACK.C!CVC.C!OPP.C!SWB.C!MOD.C3!MOD.C2!MOD.C1!MOD.C0!IE.C!FMT.C1!FMT.CO
349      ;NOT "COMMAND" BITS.
350
351      ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TS05 COMMAND.
352
353      100013      DRI==  ACK.C!CMD.C3!CMD.C1!CMD.CO
354      ;DRIVE INIT.
355
356      104001      RDF==  ACK.C!BRF.C!CMD.CO
357      ;READ FORWARD
358
359      104401      RDR==  ACK.C!BRF.C!MOD.CO!CMD.CO
360      ;READ REVERSE
361
362      104005      WRT==  ACK.C!BRF.C!CMD.CO!CMD.C2
363      ;WRITE COMMAND
364

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL EQUATES SECTION

SEQ 047

365	104105	WTV==	ACK.C!BRF.C!VFY.C!CMD.CO!CMD.C2 ;WRITE VERIFY
366			
367			
368	104010	SRF==	ACK.C!BRF.C!CMD.C3 ;SPACE RECORD FORWARD
369			
370			
371	104410	SRR==	ACK.C!BRF.C!MOD.CO!CMD.C3 ;SPACE RECORD REVERSE
372			
373			
374	105401	RNR==	ACK.C!BRF.C!MOD.C1!MOD.CO!CMD.CO ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVERSE
375			
376			
377	125401	RNF==	ACK.C!BRF.C!OPP.C!MOD.C1!MOD.CO!CMD.CO ;READ REV RETRY2 - REREAD NEXT FORWARD, IE. READ FORWARD, SPACE REVERSE
378			
379			
380	105001	RPF==	ACK.C!BRF.C!MOD.C1!CMD.CO ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, READ FORWARD
381			
382			
383	125001	RPR==	ACK.C!BRF.C!OPP.C!MOD.C1!CMD.CO ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SPACE FORWARD
384			
385			
386	105005	WRR==	ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.CO ;WRITE RETRY
387			
388			
389	102010	RWD==	ACK.C!MOD.C2!CMD.C3 ;REWIND COMMAND
390			
391			
392	100012	MBR==	ACK.C!CMD.C3!CMD.C1 ;MESSAGE BUFFER RELEASE
393			
394			
395	100011	WTM==	ACK.C!CMD.C3!CMD.CO ;WRITE TAPE MARK.
396			
397			
398	101011	WTR==	ACK.C!MOD.C1!CMD.C3!CMD.CO ;WRITE TAPE MARK RETRY.
399			
400			
401	105010	SFF==	ACK.C!BRF.C!MOD.C1!CMD.C3 ;SPACE FILE FORWARD
402			
403			
404	105410	SFR==	ACK.C!BRF.C!MOD.CO!MOD.C1!CMD.C3 ;SPACE FILE REVERSE
405			
406			
407	100017	GES==	ACK.C!CMD.CO!CMD.C1!CMD.C2!CMD.C3 ;GET EXTENDED STATUS
408			
409			
410	100411	ERS==	ACK.C!MOD.CO!CMD.C3!CMD.CO ;ERASE 3 INCHES OF TAPE
411			
412			
413	100412	UNL==	ACK.C!MOD.CO!CMD.C3!CMD.C1 ;UNLOAD COMMAND
414			
415			
416	101012	CLN==	ACK.C!MOD.C1!CMD.C3!CMD.C1 ;ERASE TAPE.
417			
418			
419	140004	SCH==	ACK.C!CVC.C!CMD.C2 ;SET DEVICE CHARACTERISTICS.
420			
421	140006	WSM==	ACK.C!CVC.C!CMD.C2!CMD.C1 ;WRITE SUB-SYS MEM

GLOBAL AREAS
GLOBAL EQUATES SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 048

```

422
423      100006      DIA==  ACK.C!CMD.C2!CMD.C1      ;DIAGNOSTICS.
424
425      000040      JMP==  JMP.C                    ;JUMP TO "N"TH COMMAND
426
427      000020      DLY==  DLY.C                    ;DELAY "N" MS.
428
429      177777      END==  177777                    ;END OF COMMAND SEQUENCES
430
431      .SBTTL  GLOBAL DATA SECTION
432      ;**
433      ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
434      ; IN MORE THAN ONE TEST.
435      ;--
436
437
438
439      ;      COMMAND PACKET.
440
441      .          =      <..+3>&177774      ;MUST BE ON MOD 4 BOUNDARY.
442      CNDPKT:: 0      ;1ST WORD IS TS05 COMMAND.
443      0          ;2ND WORD IS THE BUFFER LOW ADDRESS.
444      0          ;3RD WORD IS THE BUFFER HIGH ADDRESS.
445      0          ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.
446
447
448
449      ;      GET STATUS COMMAND PACKET.
450
451      .          =      <..+3>&177774      ;MUST BE ON MOD 4 BOUNDARY.
452      GSCPK:: .WORD  GES
453
454      ;      MESSAGE BUFFER RELEASE COMMAND PACKET.
455
456      .          =      <..+3>&177774      ;MUST BE ON MOD 4 BOUNDARY.
457      BRCPK:: .WORD  MBR
458
459
460
461      ;      REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)
462
463      .          =      <..+3>&177774      ;MUST BE ON MOD 4 BOUNDARY.
464      RWCpk:: .WORD  RWD
465      1          ;
466
467      ;      WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.
468      MSGPKT:: .BLKW 8.
469      ;1ST WORD:: MESSAGE TYPE.
470      ;2ND WORD:: DATA FIELD LENGTH.
471      ;3RD WORD:: RESIDUAL FRAME COUNT.
472      ;4TH WORD:: XSTAT0
473      ;5TH WORD:: XSTAT1
474      ;6TH WORD:: XSTAT2
475      ;7TH WORD:: XSTAT3
476
477
478

```

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 049

```

479                                     ;8TH WORD:: XSTAT4
480                                     ;
481                                     ; MESSAGE PACKETS.
482 002374 MSGPK0:: .BLKW 8.                ;MESSAGE PACKET FOR DEVICE #0
483 002414 MSGPK1:: .BLKW 8.                ;MESSAGE PACKET FOR DEVICE #1
484 002434 MSGPK2:: .BLKW 8.                ;MESSAGE PACKET FOR DEVICE #2
485 002454 MSGPK3:: .BLKW 8.                ;MESSAGE PACKET FOR DEVICE #3
486
487                                     ; SET CHARACTERISTIC BLOCK.
488
489 002474 002374 SCHBK:: MSGPK0            ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXECUTE ROUTINE).
490 002476 000000 0                        ;2ND WORD:: MSGPKT ADDR HI.
491 002500 000020 MSGCNT                    ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
492 002502 000040 CH.EAI                    ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP ROUTINE).
493 002504 000000 0                        ;5TH WORD:: MSP & BUFFER CONTROL ON EXT'D FEATURES
494
495                                     ; WRITE SUB-SYSTEM MEMORY CHARACTERISTIC BLOCK.
496
497
498 002506 000000 WSMBK:: 0                  ;1ST WORD:: SEL 0
499 002510 000000 0                        ;2ND WORD:: SEL 2
500 002512 000000 0                        ;3RD WORD:: SEL 4
501
502                                     ; TS05 REGISTER ADDRESSES.
503
504 002514 TSDB:: .BLKW 4                    ;TS05 DATA BUFFER ADDRESSES.
505 002524 TSSR:: .BLKW 4                    ;TS05 STATUS REGISTER ADDRESSES.
506 002534 TSVCT:: .BLKW 4                  ;TS05 VECTOR ADDRESSES.
507 002514 TSBA==TSDB                       ;DATA BUFFER ADDRESS REGISTER.
508
509                                     ; ADDRESSES OF MESSAGE PACKETS.
510
511
512 002544 002374 MSGPKA:: MSGPK0            ;DEVICE 0.
513 002546 002414 MSGPK1                    ;DEVICE 1.
514 002550 002434 MSGPK2                    ;DEVICE 2.
515 002552 002454 MSGPK3                    ;DEVICE 3.
516
517                                     ; ADDRESSES OF INTERRUPT HANDLING ROUTINES.
518
519 002554 010034 TS5INT:: TS5IN0             ;DEVICE 0.
520 002556 010042 TS5IN1                     ;DEVICE 1.
521 002560 010050 TS5IN2                     ;DEVICE 2.
522 002562 010056 TS5IN3                     ;DEVICE 3.
523
524                                     ; TS05 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
525
526 002564 000000 TS5CL:: 0                  ;DEVICE 0
527 002566 000000 0                        ;DEVICE 1
528 002570 000000 0                        ;DEVICE 2
529 002572 000000 0                        ;DEVICE 3
530
531                                     ; TS05 EXT. FEA & BUF. ENA SW'S, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
532
533 002574 000000 TS5SW:: 0                  ;DEVICE 0
534 002576 000000 0                        ;DEVICE 1
535 002600 000000 0                        ;DEVICE 2

```

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 050

```

536 002602 000000          0          ;DEVICE 3
537
538          ;          UNIT NUMBERS OF ALL DEVICES BEING TESTED(1-4).
539          ;          WHEN DEVICE IS NOT IN USE, IT'S LOCATION WILL = -3.
540          ;          RS WILL ALWAYS CONTAIN THE PRESENT LOGICAL UNIT NUMBER X 2.
541
542 002604 177774  DEVTBL:: .WORD  NINUSE
543 002606 177774          .WORD  NINUSE
544 002610 177774          .WORD  NINUSE
545 002612 177774          .WORD  NINUSE
546 002614 177777          .WORD  END
547
548
549          ;          BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
550          ;          "WRTY" TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST
551
552 002616 003046  BTADDR:: BT0
553 002620 003120          BT1
554 002622 003172          BT2
555 002624 003244          BT3
556          ;          COUNTER AREA.
557
558          002626  CNTBGN=.
559 002626          WRBC:: .BLKW  20          ;BYTES WRITTEN.
560 002666          RRBC:: .BLKW  20          ;BYTES READ REV.
561 002726          RFBC:: .BLKW  20          ;BYTES READ FWD.
562 002766          WRREC:: .BLKW  4          ;RECOVERABLE WRITE ERRORS.
563 002776          WRUNR:: .BLKW  4          ;UNRECOVERABLE WRITE ERRORS.
564 003006          RRREC:: .BLKW  4          ;RECOVERABLE READ REV ERRORS.
565 003016          RRUNR:: .BLKW  4          ;UNRECOVERABLE READ REV ERRORS.
566 003026          RFREC:: .BLKW  4          ;RECOVERABLE READ FWD ERRORS.
567 003036          RFUNR:: .BLKW  4          ;UNRECOVERABLE READ FWD ERRORS.
568 003046          BT0:: .BLKW  21.          ;UNIT 0 BAT TAPE SPOTS LOG
569 003120          BT1:: .BLKW  21.          ;UNIT 1 BAT TAPE SPOTS LOG
570 003172          BT2:: .BLKW  21.          ;UNIT 2 BAT TAPE SPOTS LOG
571 003244          BT3:: .BLKW  21.          ;UNIT 3 BAT TAPE SPOTS LOG
572 003316          WRTYCT:: .BLKW  4          ;WRITE RETRY COUNTER
573 003326          PASCNT:: .BLKW  4          ;PASS COUNT.
574 003336          SCCNT:: .BLKW  4          ;SPECIAL CONDITION COUNT.
575 003346          VFYCNT:: .BLKW  4          ;COUNT OF TS05 DATA COMPARE ERRORS.
576 003356          HROCNT:: .BLKW  4          ;COUNT OF HARD ERRORS.
577 003366          FTLCNT:: .BLKW  4          ;COUNT OF FATAL ERRORS.
578          003376  CNTEND=.          ;END OF STATISTICAL COUNTERS.
579 003376          RECCNT:: .BLKW  4          ;NUMBER OF RECORDS FROM BOT: CLEARED ON REWIND
580          ;          AND WHEN RESTARTING OR CONTINUING TEST 2.
581          000550  CNTLEN= CNTEND-CNTBGN          ;LENGTH OF STATISTICAL COUNTER AREA.
582
583
584          ;          THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
585          ;          USED BY THE PROGRAM.
586
587 003406 000000  DATAWT:: .WORD  0          ;WRITE BUFFER ADDRESS.
588          003406  DIABLK= DATAWT          ;WRITE BUFFER ALSO USED FOR DIAG CMD.
589 003410 000000  DATARD:: .WORD  0          ;READ BUFFER ADDRESS.
590 003412 000000  NCNT:: .WORD  0          ;STORAGE FOR VALUE OF N.
591 003414 000000  NCNT1:: .WORD  0          ;TEMP STORAGE FOR VALUE OF N.
592 003416 000000  BRFCNT:: .WORD  0          ;STORAGE FOR BPCR VALUE.

```

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 051

```

593 003420 177777      CMDWRD:: .WORD  END      ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
594 003422 177777      CHMSAV:: .WORD  END      ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVERY
595 003424 177777      PCMDWD:: .WORD  END      ;CONTAINS PREVIOUS COMMAND WORD.
596 003426 000000      CMDLG:: .WORD  0          ;CURRENT COMMAND LOGGING CODE.
597 003430 000000      LENMSK:: .WORD  0          ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
598 003432 153624      RANB:: .WORD  153624      ;RANDOM # GENERATOR BASE.
599 003434 032561      RANS:: .WORD  32561       ;RANDOM # SAVE LOCATION.
600 003436 000000      TIME1:: .WORD  0          ;TIME COUNT 1.
601 003440 000000      TIME2:: .WORD  0          ;TIME COUNT 2.
602 003442 000000      JLOOP:: .WORD  0          ;JMP COMMAND LOOP COUNT.
603 003444 000000      JLOC:: .WORD  0          ;JMP COMMAND LOCATION COUNT.
604 003446 000000      PATERN:: .WORD  0          ;PATTERN SELECT CODE.
605 003450 000000      CTCC:: .WORD  0          ;CURRENT TERMINATION CLASS CODE.
606 003452 000000      RSSAVE:: .WORD  0          ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
607 003454 000000      TSSREG:: .WORD  0          ;CURRENT STATUS REGISTER.
608 003456 000000      WTMFLG:: .WORD  0          ;WRITE TAPE MARK FLAG
609
610      ;      ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
611      ;      AFTER EACH COMMAND IS COMPLETED.
612
613      BGNFLG=.
614 003460 000000      RETRYC:: .WORD  0          ;# OF RECOVERY ATTEMPTS EXECUTED.
615 003462 000      RPTCNT:: .BYTE  0          ;WRITE REPEAT ON SAME SPOT CNTR: 4 PER WRITE RETRY
616 003463 000      WRTYFG:: .BYTE  0          ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
617 003464 000      WRTYER:: .BYTE  0          ;WRITE RETRY ON SAME SPOT ERROR FLAG
618 003465 000      RECLOG:: .BYTE  0          ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
619 003466 000      ERLOG:: .BYTE  0          ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS RECORD.
620 003467 000      RWERR:: .BYTE  0          ;READ/WRITE ERROR HAS OCCURED.
621 003470 000      UNREC:: .BYTE  0          ;UNRECOVERABLE ERROR HAS OCCURED.
622 003471 000      ERRREC:: .BYTE  0          ;ERROR RECOVERY MODE.
623      .EVEN
624      003472      ENDERF=.
625
626
627      ;      ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.
628
629 003472      INTFLG:: .BLKW  4          ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
630 003502      EOTFLG:: .BLKW  4          ;EOT/BOT FLAGS FOR EACH DEVICE (XSTATO).
631 003512 000000      BTPT:: .WORD  0          ;BAD TAPE SPOT POINTER TO BTO-BT3 VIA BTADDR
632 003514 000      EXPBOT:: .BYTE  0          ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
633 003515 000      RANDOM:: .BYTE  0          ;RANDOM EVERYTHING FLAG.
634 003516 000      VFYFLG:: .BYTE  0          ;SET DURING WRITE/VERIFY COMMAND.
635 003517 000      RPTFLG:: .BYTE  0          ;PERFORMANCE REPORT HAS BEEN REQUESTED.
636 003520 000      SWBFLG:: .BYTE  0          ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZERO.
637 003521 000      IRE:: .BYTE  0          ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
638 003522 000      DROPED:: .BYTE  0          ;CURRENT UNIT HAS BEEN DROPPED
639 003523 000      T1SWB:: .BYTE  0          ;TEST1 SWAP BYTES FLAG
640 003524 000      ALLEOT:: .BYTE  0          ;ALL UNITS @ EOT FLAG
641 003525 000      ERSFLG:: .BYTE  0          ;ERASE FLAG: DO ERASE AFTER A SPACE REV TO DELETE
642      ;      BADLY WRITTEN RECORD. 1 TO 4 ERASES LEAVING
643      ;      A 3 TO 12 INCH GAP MAY RESULT.
644      .EVEN
645      003526      ENDFLG=.
646
647      ;      ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.
648
649 003526 000      STAFLG:: .BYTE  0          ;START FLAG - SET BY INIT CODE IF STARTING.

```

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 052

```

650 003527      000      PWRFLG:: .BYTE 0      ;POWER FAILURE FLAG - SET ONLY DURING INIT.
651 003530      000      TRAPD4:: .BYTE 0      ;TRAPED AT 4 FLAG
652 003531      000      MISCFG:: .BYTE 0      ;MISCELLANEOUS FLAG
653 003532      000000    TSUNT:: .WORD 0      ;NUMBER OF THE UNIT UNDER TEST PLUS HSSP&BUF
654 003534      000000    TSNP:: .WORD 0      ;FOR PRINT OUT UNIT # ONLY
655
656      ;      OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
657      ;      SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST
658
659 003536      000000    OPFLAG:: .WORD 0      ;READ ONLY OPERATOR FLAG WORD
660      .EVEN
661
662      ;THE FOLLOWING IS THE COMMAND SEQUENCE TABLE. THE TABLE
663      ;HAS DEFAULT VALUES AT PROGRAM LOAD AS SHOWN. THESE VALUES
664      ;CAN BE UPDATED BY A TEST OR BY OPERATOR INPUT.
665
666 003540      140004      CMDSEQ:: .WORD SCH      ;SET CHARACTERISTICS.
667 003542      000040      .WORD CH.EAI
668 003544      000001      .WORD 1
669 003546      000000      .WORD 0
670 003550      102010      CMDSE2:: .WORD RWD      ;REWIND.
671 003552      000001      .WORD 1      ;BYTE COUNT.
672 003554      000001      .WORD 1      ;ONCE.
673 003556      000007      .WORD RANP      ;PATTERN.
674 003560      104005      .WORD WRT      ;WRITE.
675 003562      004000      .WORD DATCNT      ;MAX BUFFER LENGTH.
676 003564      076400      .WORD 32000.      ;32,000 RECORDS.
677 003566      000007      .WORD RANP      ;RANDOM PATTERN.
678 003570      104401      .WORD RDR      ;READ REV.
679 003572      004000      .WORD DATCNT      ;MAX BUFFER LENGTH.
680 003574      076400      .WORD 32000.      ;32,000 RECORDS
681 003576      000007      .WORD RANP      ;RANDOM PATTERN.
682 003600      104001      .WORD RDF      ;READ FWD.
683 003602      004000      .WORD DATCNT      ;MAX BUFFER LENGTH.
684 003604      076400      .WORD 32000.      ;32,000 RECORDS.
685 003606      000007      .WORD RANP      ;RANDOM PATTERN.
686 003610      102010      .WORD RWD      ;REWIND.
687 003612      000001      .WORD 1      ;BYTE COUNT.
688 003614      000001      .WORD 1      ;ONCE.
689 003616      000007      .WORD RANP      ;PATTERN.
690 003620      .BLKW 40.      ;EXTENSION TO DOUBLE BUFFER SIZE
691 003740      177777      SEQEND:: .WORD END      ;SOFT END OF SEQUENCE TABLE.
692 003742      177777      .WORD END
693 003744      177777      .WORD END
694 003746      177777      .WORD END
695 003750      177777      .WORD END      ;HARD END OF SEQUENCE TABLE.
696
697      ;THE FOLLOWING IS THE TS05 COMMAND TABLE
698 003752      100013      CMDTBL:: .WORD DRI      ;DRIVE INIT.
699 003754      104001      .WORD RDF      ;READ FORWARD.
700 003756      104401      .WORD RDR      ;READ REVERSE.
701 003760      104005      .WORD WRT      ;WRITE
702 003762      104105      .WORD WTV      ;WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND
703      ;CHECK DATA ON ALL RECORDS, RDF AND
704      ;CHECK DATA ON ALL RECORDS.)
705 003764      104010      .WORD SRF      ;SPACE "N" RECORDS FORWARD.
706 003766      104410      .WORD SRR      ;SPACE "N" RECORDS REVERSE.

```

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 053

707	003770	105401	.WORD	RNR	!READ NEXT REVERSE. I.E., SPACE FWD, READ REVERSE.
708	003772	125401	.WORD	RNF	!READ NEXT FORWARD. I.E., READ FORWARD, SPACE REVERSE.
709	003774	105001	.WORD	RPF	!READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWARD
710	003776	125001	.WORD	RPR	!READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWARD
711	004000	105005	.WORD	WRR	!WRITE RETRY.
712	004002	102010	.WORD	RWD	!REWIND.
713	004004	100012	.WORD	MBR	!MESSAGE BUFFER RELEASE
714	004006	100011	.WORD	WTH	!WRITE TAPE MARK
715	004010	101011	.WORD	WTR	!WRITE TAPE MARK RETRY.
716	004012	105010	.WORD	SFF	!SPACE "N" FILES FORWARD.
717	004014	105410	.WORD	SFR	!SPACE "N" FILES REVERSE.
718	004016	100017	.WORD	GES	!GET EXTENDED STATUS.
719	004020	100411	.WORD	ERS	!ERASE 3 INCHES OF TAPE.
720	004022	100412	.WORD	UNL	!REWIND AND UNLOAD.
721	004024	101012	.WORD	CLN	!CLEAR TAPE.
722	004026	140004	.WORD	SCH	!SET CHARACTERISTICS.
723	004030	100006	.WORD	DIA	!DIAGNOSTIC COMMAND.
724	004032	000040	.WORD	JMP	!JUMP TO THE NTH COMMAND IN THE SEQUENCE.
725	004034	000020	.WORD	DLY	!DELAY "N" MS.
726	004036	177777	.WORD	END	!END OF COMMAND TABLE

THE FOLLOWING TABLE CONTAINS THE ASCII FOR EACH COMMAND.

730	004040	104	122	111	CHDASC:: .ASCII /DRI/	!DRIVE INIT.
731	004043	122	104	106	.ASCII /RDF/	!READ FORWARD.
732	004046	122	104	122	.ASCII /RDR/	!READ REVERSE.
733	004051	127	122	124	.ASCII /WRT/	!WRITE
734	004054	127	124	126	.ASCII /WTV/	!WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND CHECK DATA
735						!ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECORDS.)
736	004057	123	122	106	.ASCII /SRF/	!SPACE "N" RECORDS FORWARD.
737	004062	123	122	122	.ASCII /SRR/	!SPACE "N" RECORDS REVERSE.
738	004065	122	116	122	.ASCII /RNR/	!READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE.
739	004070	122	116	106	.ASCII /RNF/	!READ NEXT FORWARD. I.E., READ FORWARD, SPACE REVERSE.
740	004073	122	120	106	.ASCII /RPF/	!READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWARD
741	004076	122	120	122	.ASCII /RPR/	!READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWARD
742	004101	127	122	122	.ASCII /WRR/	!WRITE RETRY.
743	004104	122	127	104	.ASCII /RWD/	!REWIND.
744	004107	115	102	122	.ASCII /MBR/	!MESSAGE BUFFER RELEASE
745	004112	127	124	115	.ASCII /WTH/	!WRITE TAPE MARK
746	004115	127	124	122	.ASCII /WTR/	!WRITE TAPE MARK RETRY.
747	004120	123	106	106	.ASCII /SFF/	!SPACE "N" FILES FORWARD.
748	004123	123	106	122	.ASCII /SFR/	!SPACE "N" FILES REVERSE.
749	004126	107	105	123	.ASCII /GES/	!GET EXTENDED STATUS.
750	004131	105	122	123	.ASCII /ERS/	!ERASE 3 INCHES OF TAPE.
751	004134	125	116	114	.ASCII /UNL/	!REWIND AND UNLOAD.
752	004137	103	114	116	.ASCII /CLN/	!CLEAN TAPE.
753	004142	123	103	110	.ASCII /SCH/	!SET CHARACTERISTICS. WHERE BRF=200, 40, 20, 0.
754						!SEE TSV05/TS05 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
755	004145	104	111	101	.ASCII /DIA/	!DIAGNOSTICS. SEE TSV05/TS05 PROGRAMMING SPECIFICATION
756						!FOR DESCRIPTION. ODT MUST BE USED TO LOAD DIAGNOSTIC DATA
757						!INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
758	004150	112	115	120	.ASCII /JMP/	!JUMP TO THE NTH COMMAND IN THE COMMAND
759						!SEQUENCE TABLE, WHERE N IS DEFINED IN
760						!THE # OF OPERATIONS.
761	004153	104	114	131	.ASCII /DLY/	!DELAY "N" MS, WHERE N IS DEFINED IN
762						!THE # OF OPERATIONS.
763	004156	105	116	104	.ASCII /END/	!END OF COMMAND SEQUENCE.

GLOBAL AREAS
GLOBAL DATA SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 054

```

764 .EVEN
765
766
767
768 .SBTTL GLOBAL TEXT SECTION
769
770
771 ***
772 ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
773 ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
774 ; MORE THAN ONE TEST.
775 ; --
776
783 ;
784 ; FORMAT STATEMENTS USED IN PRINT CALLS
785 ;
786
787 .NLIST BEX
788
789 004162 045 116 045 CODELM:: .ASCIZ /%AUNIT %D1%A TS05 CODE LEVEL %03%N/
790 004231 045 116 045 SWSET:: .ASCIZ /%AUNIT %D1%A TS05 SWITCH SETTINGS %03%N/
791 .EVEN
792 004306 130 130 130 HALTH:: .ASCIZ /XXX CMD - TYPE <CR> TO CONTINUE/
793 004346 103 115 104 CHDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
794 .EVEN
795 004430 104 101 124 WTVRM:: .ASCIZ /DATA COMPARE ERROR/
796 004453 116 117 040 TOERM:: .ASCIZ /NO TSV05 RESPONSE/
797 004475 115 116 104 SCERM:: .ASCIZ /UNDEFINED SPEC COND/
798 004521 122 106 103 RFCERM:: .ASCIZ /RFC NON ZERO/
799 004536 124 123 126 NSSRM:: .ASCIZ /TSV05 NOT READY/
800 004556 122 105 124 RLEXM:: .ASCIZ /RETRY LIMIT EXCEEDED/
801 004603 104 122 111 ATTNM:: .ASCIZ /DRIVE OFF LINE/
802 004622 106 125 116 FUNRM:: .ASCIZ /FUNCTION REJECT/
803 004642 106 101 124 FATSM:: .ASCIZ /FATAL SUBSYSTEM ERROR/
804 004670 116 117 040 NOINTM:: .ASCIZ /NO INTERRUPT/
805 004705 124 101 120 TSAM:: .ASCIZ /TAPE STATUS ALERT/
806 004727 124 117 117 TOOMM:: .ASCIZ /TOO MANY INTERRUPTS/
807 004753 103 101 120 RNYM:: .ASCIZ /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
808 005017 122 105 103 RERM:: .ASCIZ /RECOVERABLE ERROR/
809 005041 125 116 122 URERM:: .ASCIZ /UNRECOVERABLE ERROR/
810 005065 045 116 045 DROPDM:: .ASCIZ /%A DROPPED UNIT %D1%N/
811 005114 045 116 045 AUDRPM:: .ASCIZ /%A ALL UNITS DROPPED%N/
812 005146 045 116 045 AUDRUN:: .ASCIZ /%A DIAGNOSTIC ONLY SUPPORTS ONE CONTROLLER%N/
813 005226 045 116 045 DTAER2:: .ASCIZ "%A%BYTE,%D4%S2%AS,%B%S2%AS/B,%B%N"
814 005275 045 104 064 DTAER3:: .ASCIZ "%D4%A BYTES IN ERROR OUT OF %D4%N"
815 005337 045 101 116 DTAER4:: .ASCIZ /%A NO DATA READ%N/
816 005360 045 101 122 DTAER5:: .ASCIZ /%A RECORD TOO LONG: >%D4%A BYTES%N/
817 005422 045 101 122 NURTY1:: .ASCIZ /%A RECOVERED ON RETRY %D2%N/
818 005456 045 101 104 OFLINM:: .ASCIZ /%A DRIVE %D1%A OFF LINE%N/
819 005507 045 101 107 GETSTM:: .ASCIZ /%A GET STATUS CMD RESULTS:%N/
820 005543 045 116 045 NODEV:: .ASCII /%A BUS TRAP AT %06%N/
821 005570 045 101 111 .ASCIZ /%A INTERFACE BAD OR TS08 NOT SET TO ABOVE ADDRESS%N/
822 005653 040 052 052 UNILK:: .ASCIZ / *****TAPE IS WRITE-LOCKED AND WILL CAUSE ERRORS*****/
823 005741 045 116 000 CRLF:: .ASCIZ /%N/
824 005744 045 116 045 CRLFSP:: .ASCIZ /%NS7/
825 .LIST BEX
826 .EVEN

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL ERROR REPORT SECTION

SEQ 055

```

827      .SBTTL  GLOBAL ERROR REPORT SECTION
828
829      ;**
830      ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX CALLS
831      ; THAT ARE USED IN MORE THAN ONE TEST.  IT ALSO INCLUDES THE ASCII MESSAGES
832      ; THAT ARE USED BY THE PRINTB AND PRINTX CALLS..
833      ;--
834
835
836      BGNMSG  DTAERM
837      DTAERM:
838      DATERM: PRINTB  #STAER1,TSNP,PASCNT(R5),RECCNT(R5)
839
840      MOV      RECCNT(R5),-(SP)
841      MOV      PASCNT(R5),-(SP)
842      MOV      TSNP, -(SP)
843      MOV      #STAER1, -(SP)
844      MOV      #4, -(SP)
845      MOV      SP,R0
846      TRAP     C#PNTB
847      ADD      #12,SP
848
849      PRINTB   #STAER7
850
851      MOV      #STAER7, -(SP)
852      MOV      #1, -(SP)
853      MOV      SP,R0
854      TRAP     C#PNTB
855      ADD      #4,SP
856
857      MOV      R2,RECREC      ;SAVE R2
858      MOV      R3,TIME1      ;SAVE R3
859      MOV      R4,TIME2      ;SAVE R4
860      JSR      PC,RECTAP      ;RETRIEVE RECORD READ
861      MOV      RECREC,R2      ;RESTORE R2
862      MOV      R3,RECREC      ;SAVE RECORD READ
863      MOV      TIME1,R3      ;RESTORE R3
864      MOV      TIME2,R4      ;RESTORE R4
865      PRINTB   #STAER6,RECREC ;PRINT RECORD READ
866
867      MOV      RECREC, -(SP)
868      MOV      #STAER6, -(SP)
869      MOV      #2, -(SP)
870      MOV      SP,R0
871      TRAP     C#PNTB
872      ADD      #6,SP
873
874      EXIT     MSG
875
876      .WORD    J1JMP
877      .WORD    L10002-2-.
878
879      .EVEN
880
881      ENDMMSG
882      L10002:
883
884      TRAP     C#MSG
885
886      BGNMSG  STAERM
887      STAERM:
888      STAERM: PRINTB  #STAER1,TSNP,PASCNT(R5),RECCNT(R5)
889
890      MOV      RECCNT(R5),-(SP)
891      MOV      PASCNT(R5),-(SP)
892      MOV      TSNP, -(SP)
893      MOV      #STAER1, -(SP)

```

GLOBAL AREAS
GLOBAL ERROR REPORT SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 056

006140	012746	000004			MOV	#4,-(SP)
006144	010600				MOV	SP,R0
006146	104414				TRAP	C#PNTB
006150	062706	000012			ADD	#12,SP
860 006154			PRINTB	#STAER7		
006154	012746	006530			MOV	#STAER7,-(SP)
006160	012746	000001			MOV	#1,-(SP)
006164	010600				MOV	SP,R0
006166	104414				TRAP	C#PNTB
006170	062706	000004			ADD	#4,SP
861 006174	016702	174130	MOV	CMDPKT,R2		
862 006200	042702	177740	BIC	#177740,R2		
863 006204	005302		DEC	R2		;IF CMD IS A READ
864 006206	005702		TST	R2		
865 006210	001016		BNE	500004		
866 006212	004767	001646	JSR	PC,RECTAP		;THEN RETRIEVE
867 006216	010367	001642	MOV	R3,RECTAP		;AND
868 006222			PRINTB	#STAER6,RECRED		;TYPE RECORD READ
006222	016746	000640			MOV	RECRED,-(SP)
006226	012746	006562			MOV	#STAER6,-(SP)
006232	012746	000002			MOV	#2,-(SP)
006236	010600				MOV	SP,R0
006240	104414				TRAP	C#PNTB
006242	062706	000006			ADD	#6,SP
869 006246			500004: PRINTX	#STAER2		
006246	012746	006616			MOV	#STAER2,-(SP)
006252	012746	000001			MOV	#1,-(SP)
006256	010600				MOV	SP,R0
006260	104415				TRAP	C#PNTX
006262	062706	000004			ADD	#4,SP
870 006266			PRINTX	#STAER3,CMDPKT,8TSD8(R5),MSGPKT+MS.RFC,TSSREG,CTCC		
006266	016746	175156			MOV	CTCC,-(SP)
006272	016746	175156			MOV	TSSREG,-(SP)
006276	016746	174056			MOV	MSGPKT+MS.RFC,-(SP)
006302	017546	002514			MOV	8TSD8(R5),-(SP)
006306	016746	174016			MOV	CMDPKT,-(SP)
006312	012746	006675			MOV	#STAER3,-(SP)
006316	012746	000006			MOV	#6,-(SP)
006322	010600				MOV	SP,R0
006324	104415				TRAP	C#PNTX
006326	062706	000016			ADD	#16,SP
871 006332			PRINTX	#STAER4,CMDPKT+2,CMDPKT+4,CMDPKT+6		
006332	016746	174000			MOV	CMDPKT+6,-(SP)
006336	016746	173772			MOV	CMDPKT+4,-(SP)
006342	016746	173764			MOV	CMDPKT+2,-(SP)
006346	012746	006733			MOV	#STAER4,-(SP)
006352	012746	000004			MOV	#4,-(SP)
006356	010600				MOV	SP,R0
006360	104415				TRAP	C#PNTX
006362	062706	000012			ADD	#12,SP
872 006366			PRINTX	#STAER5,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3,MSGPKT+MS.XS		
006366	016746	174000			MOV	MSGPKT+MS.XS4,-(SP)
006372	016746	173772			MOV	MSGPKT+MS.XS3,-(SP)
006376	016746	173764			MOV	MSGPKT+MS.XS2,-(SP)
006402	016746	173756			MOV	MSGPKT+MS.XS1,-(SP)
006406	016746	173750			MOV	MSGPKT+MS.XS0,-(SP)
006412	012746	006753			MOV	#STAER5,-(SP)

```

006416 012746 000006
006422 010600
006424 104415
006426 062706 000016
873 006432 EXIT MSG
006432 000167
006434 000432
874
875
876 006436 045 101 130 STAER1: .NLIST BEX
006436 045 101 130 STAER1: .ASCIZ /#AXXX CMD FAILED - UNIT #D1#S3#APASS:#D5#S3#ARECORD:#D5#N/
006436 045 101 130 STAER1: .EVEN
006530 045 101 120 STAER7: .ASCIZ /#APREVIOUS CMD WAS XXX /
006562 045 123 061 STAER6: .ASCIZ /#S11#A* RECORD READ:#D3#A */
006616 045 116 045 STAER2: .ASCIZ /#N#ACMDPKT#S2#ATSBA#S4#ARFC#S5#ATSSR#S3#ATCC#N/
006675 045 117 066 STAER3: .ASCIZ /#06#S2#06#S2#06#S2#06#S2#D1#N/
006733 045 117 066 STAER4: .ASCII /#06#N/
006740 045 117 066 STAER4: .ASCII /#06#N/
006745 045 117 066 STAER4: .ASCIZ /#06#N/
006753 045 101 130 STAER5: .ASCII /#AXST0#S4#AXST1#S4#AXST2#S4#AXST3#S4#AXST4#N/
007027 045 117 066 STAER5: .ASCIZ /#06#S2#06#S2#06#S2#06#S2#06#N/
007027 045 117 066 STAER5: .LIST BEX
007066 000000 RECRED: .EVEN
007066 000000 RECRED: .WORD 0 ;RECORD READ FROM TAPE
007070 007070 ENDMMSG
007070 104423 L10003:
007070 104423 TRAP C#MSG
892
893 .SBTTL GLOBAL SUBROUTINES SECTION
894
895
896 ;**
897 ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
898 ; THAT ARE USED IN MORE THAN ONE TEST.
899 ;--
900
901
902
903
904 ;*
905 ;
906 ;ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
907 ;BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
908 ;THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
909 ;DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
910 ;
911 ;INPUTS:
912 ;
913 ; R5 CURRENT UNIT NUMBER
914 ;
915 ;OUTPUTS:
916 ;
917 ; R0 CONTENTS OF TSSR, IF ERROR
918 ; CARRY SET IF INIT WAS OKAY
919 ; CLEAR IF FATAL ERROR
920 ;
921 ;CALLING SEQUENCE:
922 ; JSR PC,FIRSTU

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 058

```

922      ;      JSR      PC,SOFINIT
923      ;      BCS      CONTINUE
924      ;      ERDF      ;REPORT FATAL ERROR
925      ;
926      ;-
927
928 007072      SOFINIT::
929
930 007072 012775 000000 002524      MOV      #0,BTSSR(R5)      ; (SAVREG) SAVE THE REGISTERS
931 007100 004767 003574      JSR      PC,WSSR      ; DO THE INIT.
932 007104 012703 000550      MOV      #360.,R3      ;WAIT FOR UNIT TO BE READY
933 007110 004767 000070      2$:      JSR      PC,WAITF      ; WAIT FOR SSR
934 007114 103416      BCS      3$
935 007116      DELAY      250
          MOV      #250,(PC)+
          .WORD      0
          MOV      L#DLY,(PC)+
          .WORD      0
          DEC      -6(PC)
          BNE      -4
          DEC      -22(PC)
          BNE      -20
936 007146 005303      DEC      R3
937 007150 001357      BNE      2$
938 007152 017500 002524      3$:      MOV      BTSSR(R5),R0      ;GET THE TSSR REGISTER
939 007156 010004      MOV      R0,R4      ;TSSR CONTENTS
940 007160 042704 176277      BIC      #C<TS.A17!TS.A16!TS.OFL>,R4
941 007164 052704 002200      BIS      #TS.SSR!TS.NBA,R4      ;R4 HAS EXPECTED CONTENTS
942 007170 020400      CMP      R4,R0      ;ONLY EXPECTED BITS SET ?
943 007172 001402      BEQ      5$      ;BRANCH IF OKAY
944 007174 000241      CLC      ;CLEAR THE CARRY FOR ERROR
945 007176 000401      BR      10$      ;GO TO EXIT
946 007200 000261      5$:      SEC      ;SET THE CARRY BIT
947 007202 000207      10$:      RTS      PC      ;RETURN TO CALLER
948
949      ;
950      ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
951
952      ; INPUTS:
953
954      ;      R5      CURRENT UNIT NUMBER
955
956      ;
957      ; OUTPUTS:
958
959      ;      R0      CONTENTS OF LAST TSSR READ
960      ;      CARRY    SET - READY BIT SET
961      ;      CLR      CLR - TIMEOUT WAITING FOR READY
962
963 007204      WAITF:: BREAK      ; DO A SUPVSR BREAK FIRST.
964 007204 104422      TRAP      C#BRK
965 007206 012746 005670      2$:      MOV      #3000.,-(SP)      ; 300 MSEC TIMER.
966 007212 017500 002524      MOV      BTSSR(R5),R0      ;READ THE TSSR REGISTER
967 007216 105700      TSTB      R0      ;TEST FOR READY BIT SET
968 007220 100420      BHI      3$      ; EXIT ON STOP FLAG.
          MOV      #25,(PC)+
          .WORD      0
          DEC      -6(PC)
          BNE      -4
          DEC      -22(PC)
          BNE      -20
          DELAY      25

```


GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 060

```

1019 007326 032763 000100 000012      BIT      #X2.BFE,MS.XS2(R3)      ;BUFFER ENABLE SWITCH SET
1020 007334 001402                      BEQ      50#                ;BR, IF SWITCH NOT SET
1021 007336 005267 172762              INC      BENBSW              ;SET SOFTWARE SWITCH FOR ENABLED
1022 007342                      50#:
1023 007342 000261                      55#:      SEC                ;SET CARRY NO TROUBLE
1024 007344 000401                      BR      70#                ;EXIT
1025 007346 000241                      60#:      CLC                ;CARRY CLEAR = ERROR
1026 007350 017500 002524              70#:      MOV      @TSSR(R5),R0      ;RETURN TSSR CONTENTS
1027 007354 000207                      RTS      PC                ;RETURN
1028
1029
1030
1031      ;*
1032      ;ROUTINE TO CHECK WRITE LOCK CONDITION
1033      ;INPUT:
1034      ;
1035      ;      R4      ADDRESS OF COMMAND PACKET
1036      ;      R5      CURRENT UNIT NUMBER
1037      ;
1038      ;-
1039 007356      WLKCHK:
1040 007356 010475 002514              10#:      MOV      R4,@TSD8(R5)      ;SEND OUT COMMAND
1041 007362 004767 177616              JSR      PC,WAITF              ;WAIT FOR SSR
1042 007366 103401                      BCS      40#                ;BR, IF SSR IS SET AND OK
1043 007370 000420                      BR      60#                ;BR IF TROUBLE CARRY = CLEAR
1044 007372 005724                      40#:      TST      (R4)+          ;STEP IT
1045 007374 011402                      MOV      (R4),R2              ;POINT TO WRT CHARA DATA PACKET
1046 007376 011203                      MOV      (R2),R3              ;GET ADDRESS OF MESSAGE BUFFER
1047 007400 032763 000004 000006      BIT      #X0.WLK,MS.XS0(R3)      ;IS UNIT WRITE LOCKED?
1048 007406 001407                      BEQ      55#                ;NO,PROCEED WITH TESTING
1049 007410                      ERRHRD 1,UNIWLK              ;TAPE IS WRITE LOCKED
1050      007410 104456                      TRAP      C$ERRHRD
1051      007412 000001                      .WORD    1
1052      007414 005653                      .WORD    UNIWLK
1053      007416 000000                      .WORD    0
1054 007420 004767 007554              JSR      PC,DROPU              ;DROP IT
1055 007424 000402                      BR      60#                ;EXIT WITH CARRY=0
1056 007426 000261                      55#:      SEC                ;SET CARRY NO TROUBLE
1057 007430 000401                      BR      70#                ;EXIT
1058 007432 000241                      60#:      CLC                ;CARRY CLEAR = ERROR
1059 007434                      70#:
1060                      RTS      PC                ;RETURN
1061
1062      ;*
1063      ;ROUTINE TO ISSUE A WRITE CHARACTERISTICS COMMAND
1064      ;INPUT:
1065      ;
1066      ;      R4      ADDRESS OF COMMAND PACKET
1067      ;      R5      CURRENT UNIT NUMBER
1068      ;      REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
1069      ;OUTPUT:
1070      ;
1071      ;      R0      TSSR CONTENTS

```

```

1072      CARRY SET - WRITE CHARACTERISTICS COMMAND OK
1073      CLR - WRITE CHARACTERISTICS FAILED
1074
1075      ;IMPLICIT OUTPUT:
1076
1077
1078
1079      ;SIDE EFFECTS:
1080
1081
1082      ;-
1083
1084      WRTCHR::
1085      10$: MOV R4,BTSD8(R5)      ;SEND OUT COMMAND
1086      JSR PC,WAITF      ;WAIT FOR SSR
1087      BCS 50$      ;BR, IF SSR IS SET AND OK
1088      BR 60$      ;BR IF TROUBLE CARRY = CLEAR
1089
1090      50$: SEC      ;SET CARRY NO TROUBLE
1091      BR 70$      ;EXIT
1092      60$: CLC      ;CARRY CLEAR = ERROR
1093      70$: MOV BTSSR(R5),R0      ;RETURN TSSR CONTENTS
1094      RTS PC      ;RETURN
1095
1096
1097
1098      ;*
1099      ;ROUTINE TO DO SET UP OF RUNNING CONDITIONS
1100
1101      ;INPUTS:
1102
1103      R5      CURRENT UNIT NUMBER
1104
1105
1106      ;OUTPUTS:
1107
1108
1109      ;CALLING SEQUENCE:
1110      JSR PC,FIRSTU
1111      JSR PC,SOFINIT
1112      BCS CONTINUE
1113      ERROF      ;REPORT FATAL ERROR
1114      JSR PC,MDSET
1115
1116      ;-
1117
1118      MDSET:: BREAK      ; DO A SUPVSR BREAK FIRST.
1119      JSR PC,SETDEF      TRAP C$BRK
1120      JSR PC,WLKCHK      ;RESTORE DEFAULT
1121      BCS 1$      ;CHECK WRITE LOCK
1122      DELAY 1      ;C=1 IS O.K.
1123      ;WAIT
1124
1125      MOV #1,(PC)+
1126      .WORD 0
1127      MOV L$DLY,(PC)+
1128      .WORD 0
1129      DEC -6(PC)

```

	007522	001375					BNE	-4
	007524	005367	177756				DEC	-22(PC)
	007530	001367					BNE	-20
1123	007532				BREAK	,BREAK TO SUPER		
	007532	104422					TRAP	C#BRK
1124	007534				DOCLN	,DO CLEAN AND ABORT	TRAP	C#DCLN
	007534	104444						
1125	007536	005767	172550	1#:	TST TS1MD	,RUN IN DEFAULT MODE?		
1126	007542	001064			BNE 10#	,YES,RETURN		
1127	007544	004767	000216		JSR PC,SETDEF	,RESTORE DEFAULT		
1128	007550	004767	177514		JSR PC,WRTCHK	,GO DO SWITCH CHECK		
1129	007554	005767	172540		TST HSSW	,DO WE RUN AT 100IPS?		
1130	007560	001415			BEQ 3#	,NO		
1131	007562	052767	000040	173742	BIS #EF.HSS,TSUNT	,YES,SET THE BIT		
1132	007570	005767	172526		TST EXTFEA	,ARE WE SET?		
1133	007574	001002			BNE 2#	,YES		
1134	007576	004767	000122		JSR PC,INVRT	,INVERT THE SWITCH		
1135	007602	004767	000160	2#:	JSR PC,SETDEF	,NOW SET THE MODES		
1136	007606	004767	177624		JSR PC,WRTCHR	,DO IT		
1137	007612	000443			BR 11#			
1138	007614	005767	172476	3#:	TST WTBUF	,RUN WITH WRITE BUFFERING?		
1139	007620	001415			BEQ 5#	,NO		
1140	007622	052767	000030	173702	BIS #EF.RWB,TSUNT	,YES SET THE BITS		
1141	007630	005767	172466		TST EXTFEA	,ARE WE SET?		
1142	007634	001002			BNE 4#	,YES		
1143	007636	004767	000062		JSR PC,INVRT	,INVERT THE SWITCH		
1144	007642	004767	000120	4#:	JSR PC,SETDEF	,NOW SET THE MODES		
1145	007646	004767	177564		JSR PC,WRTCHR	,DO IT		
1146	007652	000423			BR 11#			
1147	007654	005767	172434	5#:	TST RDBUF	,RUN WITH READ BUFFERING?		
1148	007660	001415			BEQ 10#	,NO		
1149	007662	052767	000020	173642	BIS #EF.RBO,TSUNT	,YES SET THE BITS		
1150	007670	005767	172426		TST EXTFEA	,ARE WE SET?		
1151	007674	001002			BNE 6#	,YES		
1152	007676	004767	000022		JSR PC,INVRT	,INVERT THE SWITCH		
1153	007702	004767	000060	6#:	JSR PC,SETDEF	,NOW SET THE MODES		
1154	007706	004767	177524		JSR PC,WRTCHR	,DO IT		
1155	007712	000403			BR 11#			
1156								
1157	007714	016767	173612	172562	10#:	MOV TSUNT,SCHBK+10	,AND UNIT #	
1158								
1159	007722	000207			11#:	RTS PC	,RETURN	
1160								
1161								
1162								
1163								
1164								
1165								
1166								
1167								
1168								
1169								
1170								
1171								
1172								
1173	007724				INVRT::			
1174	007724	012767	140006	172376	MOV #WSM,CMDPKT+CP.CMD	,WRT SUB-SYS MEM		

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 063

```

1175 007732 012767 002506 172372      MOV      #WSMBK,CMDPKT+CP.ADL      ;MSG BUF ADDR
1176 007740 012767 000006 172370      MOV      #6,CMDPKT+CP.CNT      ;BYTE COUNT
1177 007746 012767 100010 172532      MOV      #100010,WSMBK      ;INVERT THE SWITCH
1178 007754 012704 002330              MOV      #CMDPKT,R4              ;
1179 007760 004767 177452              JSR      PC,WRTCHR      ;DO IT
1180 007764 000207                    RTS      PC      ;RETURN
1181
1182
1183      ; SUBROUTINE TO SETUP DEFAULT SET CHAR CMD
1184      ;
1185      ; INPUTS:
1186      ;
1187      ;
1188      ;
1189      ; OUTPUTS:
1190      ;
1191      ;      R4      ADDRESS OF COMMAND PACKET
1192
1193 007766      SETDEF::
1194 007766      012701 140004      MOV      #SCH,R1      ;WRITE CHAR CMD
1195 007772      010167 172332      MOV      R1,CMDPKT+CP.CMD      ;SET UP COMMAND
1196 007776      012767 002474 172326      MOV      #SCHBK,CMDPKT+CP.ADL      ;SET UP ADR LO TO POINT TO MSG BUF(MSGPK0)
1197 010004      012767 000012 172324      MOV      #SCHCNT,CMDPKT+CP.CNT      ;SET BUFFER EXTENT
1198 010012      012767 000040 172462      MOV      #DFTSCH,SCHBK+6      ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
1199 010020      016767 173506 172456      MOV      TSUNT,SCHBK+10      ;UNIT #
1200 010026      012704 002330      MOV      #CMDPKT,R4      ;ADDRESS OF CMD PACKET
1201 010032 000207                    RTS      PC      ;RETURN
1202
1203
1204      ;      MODULES TO HANDLE TS05 INTERRUPTS.
1205
1206
1207 010034      BGNSRV TS5IN0
1208 010034      TS5IN0::
1209 010034 005267 173432      INC      INTFLG      ;SET INTERRUPT OCCURRED FLAG.
1210 010040      ENDSRV
1211 010040      L10004:
1212 010040 000002                                RTI
1213
1214 010042      BGNSRV TS5IN1
1215 010042      TS5IN1::
1216 010042 005267 173426      INC      INTFLG+2      ;SET INTERRUPT OCCURRED FLAG.
1217 010046      ENDSRV
1218 010046      L10005:
1219 010046 000002                                RTI
1220
1221 010050      BGNSRV TS5IN2
1222 010050      TS5IN2::
1223 010050 005267 173422      INC      INTFLG+4      ;SET INTERRUPT OCCURRED FLAG.
1224 010054      ENDSRV
1225 010054      L10006:
1226 010054 000002                                RTI
1227
1228 010056      BGNSRV TS5IN3
1229 010056      TS5IN3::
1230 010056 005267 173416      INC      INTFLG+6      ;SET INTERRUPT OCCURRED FLAG.
1231 010062      ENDSRV

```

```

010062          L10007:
010062 000002          RTI

1222
1223          ; SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
1224          ; PRINTS.
1225          ; INPUTS:
1226          ; OUTPUTS: R3 = RECORD COUNT READ
1227          ; REGISTERS: R2, R3, R4
1228          ; CALLS:
1229
1230 010064 032767 000400 173326 RECTAP: BIT    #MOD.CO,CMDWRD      ;READ REV FETCH
1231 010072 001430          BEQ    50001#
1232 010074 016702 172260          MOV    MSGPKT+MS.RFC,R2      ;FIND LAST READ AD.
1233 010100 066702 173304          ADD    DATARD,R2
1234 010104 032702 000001          BIT    #BIT00,R2            ;ODD AD., REASSEMBLE
1235 010110 001417          BEQ    50002#
1236 010112 005202          INC     R2                          ;REC COUNT STARTING
1237 010114 111203          MOVB   (R2),R3                     ;WITH UPPER BYTE FETCH
1238 010116 142703 177400          BICB   #177400,R3
1239 010122 000303          SWAB   R3
1240 010124 005302          DEC     R2                          ;LET R2 := R2 - #1      ;LOWER BYTE AD.
1241 010126 105767 173366          TSTB   SWBFLG                ;IFB SWBFLG NE #0 THEN
1242 010132 001401          BEQ    50003#
1243 010134 005302          DEC     R2                          ;LET R2 := R2 - #1      ;LOWER BYTE AD. ON SWAP
1244
1245 010136          50003#:
1246 010136 111204          MOVB   (R2),R4                      ;FETCH LOWER BYTE
1247 010140 142704 177400          BICB   #177400,R4
1248 010144 050403          BIS     R4,R3
1249 010146 000401          BR      50004#
1250 010150          50002#:
1251 010150 011203          MOV     (R2),R3                    ;LET R3 := (R2)      ;EVEN AD. FETCH
1252 010152          50004#:
1253 010152 000402          BR      50005#
1254 010154          50001#:
1255 010154 017703 173230          MOV     @DATARD,R3          ;LET R3 := @DATARD      ;READ FWD FETCH
1256
1257 010160          50005#:
1258 010160 000207          RTS     PC
1259
1260          ; SUBROUTINE TO STORE A SET CHARACTERISTIC COMMAND AS
1261          ; THE FIRST ENTRY IN THE SEQUENCE TABLE.
1262          ; INPUTS:
1263          ; OUTPUTS:
1264          ; REGISTERS:
1265          ; CALLS:
1266
1267 010162          SETCH::
1268 010162 012701 003540          MOV     #CMDSEQ,R1          ;INIT CMD SEQUENCE TABLE POINTER.
1269 010166 012721 140004          MOV     #SCH,(R1)+          ;THIS CODE SETS UP A SET CHARACTERISTIC
1270 010172 012721 000040          MOV     #DFTSCH,(R1)+        ;COMMAND AS THE FIRST COMMAND IN THE
1271 010176 012721 000001          MOV     #1,(R1)+          ;SEQUENCE TABLE.
1272 010202 005721          TST     (R1)+          ;SKIP PATTERN LOCATION.
1273 010204 000207          RTS     PC
1274
1275          ; SUBROUTINE TO STORE A REWIND COMMAND IN THE SEQUENCE TABLE
1276          ; INPUTS:

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 065

```

1277      ;      OUTPUTS:
1278      ;      REGISTERS:
1279      ;      CALLS:
1280
1281 010206 012721 102010      SETRW:: MOV      @RWD,(R1)+      ;CMD = REWIND.
1282 010212 012721 000001      MOV      @1,(R1)+      ;BRF.
1283 010216 012721 000001      MOV      @1,(R1)+      ;# OF OPERATIONS.
1284 010222 005721      TST      (R1)+      ;SKIP PATTERN.
1285 010224 000207      RTS      PC      ;RETURN
1286
1287      ;      SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
1288      ;      DEVICES.
1289      ;      INPUTS:
1290      ;      OUTPUTS:      R2 = TERMINATION INDICATOR (0=END OF TABLE,1=EOT)
1291      ;      REGISTERS:
1292      ;      CALLS:      CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.
1293
1294 010226 012701 003540      EXALL:: MOV      @CMDSEQ,R1      ;INIT SEQUENCE TABLE POINTER.
1295 010232      50006::      CMP      (R1),@END      ;WHILE THERE ARE CMDS IN THE SEQUENCE TABLE.
1296 010232 021127 177777      BEQ      50007:
1297 010236 001530      JSR      PC,SETUP      ;GO SETUP THE COMMAND BLOCK.
1298 010240 004767 000726      50010:: BREAK      ; DO A SUPVSR BREAK FIRST.
1299 010244      104422      TRAP      C#BRK
1300 010246 026767 173140 173140      CMP      NCNT,NCNT1      ;WHILE THERE ARE RECORDS REMAINING:
1301 010254 002116      BGE      50011:
1302 010256 004767 000602      JSR      PC,CMDAC      ;STORE CMD ASCII IN ERROR MESSAGE.
1303 010262 105767 173227      TSTB      RANDOM      ;IF IN RANDOM MODE:
1304 010266 001435      BEQ      50012:
1305 010270 026727 173124 104005      CMP      CMDWRD,@WRT      ;IF CMD IS A WRITE THEN:
1306 010276 001031      BNE      50013:
1307 010300 105767 173212      TSTB      VFYFLG      ;IF DATA IS NOT TO BE VERIFIED THEN:
1308 010304 001026      BNE      50014:
1309 010306 066767 173122 173116      ADD      RANS,RANS      ;LET RANS := RANS + RANS ;GENERATE
1310 010314 066767 173112 173112      ADD      RANS,RANS      ;LET RANS := RANS + RANS ;RANDOM
1311 010322 016767 173106 173066      MOV      RANS,BRFCNT      ;LET BRFCNT := RANS ;LENGTH
1312 010330 046767 173074 173060      BIC      LENMSK,BRFCNT      ;MASK RANDOM LENGTH.
1313 010336 026727 173054 000022      CMP      BRFCNT,@18.      ;DO NOT ALLOW BYTE COUNT OF LESS THAN 18
1314 010344 002003      BGE      50015:
1315 010346 012767 000022 173042      MOV      @18.,BRFCNT      ;CHANGE COUNT OF 0-17 TO 18.
1316
1317 010354      50015::
1318 010354 016767 173036 171754      MOV      BRFCNT,CMDPKT+CP.CNT      ;MOVE BRF TO CMD PACKET.
1319
1320 010362      50014::
1321
1322 010362      50013:
1323
1324 010362      50012:
1325 010362 004767 000136      JSR      PC,EXSUB      ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
1326 010366 004767 007106      JSR      PC,CKHAE      ;CHECK HALT AFTER EACH CMD FLAG.
1327 010372 012702 000001      MOV      @1,R2      ;LET R2 := @1 ;SET ALL UNITS AT BOT/EOT.
1328 010376 004767 006500      JSR      PC,FIRSTU      ;FIND FIRST UNIT.
1329
1330 010402      50016::
1331 010402 026527 002604 177777      CMP      DEVTBL(R5),@END      ;WHILE THERE ARE MORE UNITS:
1332 010410 001426      BEQ      50017:

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 066

```

1333 010412 032767 000400 173000      BIT      #MOD.CO,CMDWRD      ;IF CMD IS REVERSE THEN:
1334 010420 001406                    BEQ      500201
1335 010422 032765 000002 003502      BIT      #X0.BOT,EOTFLG(R5)      ;IF NOT AT BOT THEN:
1336 010430 001001                    BNE      500211
1337 010432 005002                    CLR      R2                      ;LET R2 := #0 ;CLEAR EOT/BOT FLAG.
1338
1339 010434                    500211:
1340 010434 000411                    BR      500221                      ;ELSE IF CMD IS NOT REVERSE:
1341 010436                    500201:
1342 010436 032765 000001 003502      BIT      #X0.EOT,EOTFLG(R5)
1343 010444 001404                    BEQ      500231
1344 010446 032767 000001 172744      BIT      #CMD.CO,CMDWRD
1345 010454 001001                    BNE      500241
1346 010456                    500231:
1347
1348 010456 005002                    CLR      R2                      ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
1349
1350 010460                    500241:
1351
1352 010460                    500221:
1353 010460 004767 006464      JSR      PC,NEXTU                      ;FIND NEXT UNIT
1354 010464 000746                    BR      500161
1355 010466                    500171:
1356 010466 020227 000001      CMP      R2,#1                      ;IF ALL UNIT ARE AT EOT/BOT THEN:
1357 010472 001001                    BNE      500251
1358 010474 000412      BR      EXARTN                      ;RETURN WITH R2 = #1.
1359
1360 010476                    500251:
1361 010476 005267 172710      INC      NCNT                      ;LET NCNT := NCNT + #1 ;UPDATE RECORD COUNT.
1362 010502 016767 172712 172714      MOV      CMDWRD,PCMDWD      ;SAVE PREVIOUS COMMAND WORD.
1363
1364 010510 000655                    BR      500101
1365 010512                    500111:
1366 010512 004767 005350      JSR      PC,VFYDAT                      ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
1367
1368
1369 010516 000645                    BR      500061
1370 010520                    500071:
1371 010520 005002      CLR      R2                      ;LET R2 := #0 ;SET NORMAL RETURN INDICATOR.
1372 010522 000207      EXARTN: RTS PC                      ;RETURN.
1373
1374
1375
1376      ; SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES. WAIT FOR
1377      ; ALL INTERRUPTS, AND CHECK ALL STATUS.
1378      ; INPUTS:
1379      ; OUTPUTS:
1380      ; REGISTERS:
1381      ; CALLS:      EXCUTE,GOWAIT,NEXTU,FIRSTU.
1382
1383 010524 004767 006352      EXSUB:: JSR      PC,FIRSTU                      ;SET UP FOR FIRST UNIT.
1384 010530                    500261:
1385 010530 026527 002604 177777      CMP      DEVTBL(R5),#END      ;WHILE THERE ARE MORE DEVICES:
1386 010536 001465                    BEQ      500271
1387 010540 032767 000400 172652      BIT      #MOD.CO,CMDWRD      ;IF CMD IS REVERSE THEN:
1388 010546 001421                    BEQ      500301
1389 010550 032765 000002 003502      BIT      #X0.BOT,EOTFLG(R5)      ;IF NOT AT BOT

```

1390	010556	001014			BNE	50031:		
1391	010560	032765	000001	003502	BIT	50031:	#X0.EOT,EOTFLG(R5)	;BUT IF AT EOT
1392	010566	001406			BEQ	50032:		
1393	010570	105767	172730		TSTB	50032:	ALLEOT	;AND ALL OTHERS AT EOT
1394	010574	001402			BEQ	50033:		
1395	010576	004767	001252		JSR	50033:	PC,EXECUTE	;THEN EXECUTE REV CMD
1396								;IF NOT ALL AT EOT, FREEZE UNIT(S) AT EOT
1397	010602				50033:			
1398	010602	000402			BR	50034:		;IF NOT AT BOT AND
1399	010604				50032:			
1400	010604	004767	001244		JSR	50032:	PC,EXECUTE	;NOT AT EOT, EXEC REV CMD
1401								
1402	010610				50034:			
1403								
1404	010610				50031:			
1405	010610	000435			BR	50035:		;ELSE IF CMD IS NOT REVERSE:
1406	010612				50030:			
1407	010612	026727	172610	000002	CMPL	50030:	CMDLG,#2	
1408	010620	001011			BNE	50036:		
1409	010622	032765	000002	003502	BIT	50036:	#X0.BOT,EOTFLG(R5)	
1410	010630	001405			BEQ	50036:		
1411								;CLEAR BAD SPOT COUNTS WHEN WRITING FROM BOT
1412	010632	016567	002616	172652	MOV		BTADDR(R5),BTPT	;LET BTPT := BTADDR(R5)
1413	010640	005077	172646		CLR		8BTPT	;LET 8BTPT := #0
1414								
1415	010644				50036:			
1416	010644	032765	000001	003502	BIT	50036:	#X0.EOT,EOTFLG(R5)	
1417	010652	001404			BEQ	50037:		
1418	010654	032767	000001	172536	BIT	50037:	#CMD.CO,CMDWRD	
1419	010662	001003			BNE	50040:	50040:	
1420	010664				50037:			
1421								;IF NOT AT EOT OR NOT A MOTION CMD THEN:
1422	010664	004767	001164		JSR	50037:	PC,EXECUTE	;ISSUE CMD TO TS05
1423								
1424	010670	000405			BR	50041:		
1425	010672				50040:			
1426	010672	105767	172626		TSTB	50040:	ALLEOT	;IFB ALLEOT NE #0 THEN
1427	010676	001402			BEQ	50042:		
1428	010700	004767	001150		JSR	50042:	PC,EXECUTE	
1429								
1430	010704				50042:			
1431								
1432	010704				50041:			
1433								
1434	010704				50035:			
1435	010704	004767	006240		JSR	50035:	PC,NEXTU	;FIND NEXT UNIT IN TEST CYCLE.
1436								
1437	010710	000707			BR	50027:	50026:	
1438	010712				50027:			
1439	010712	105767	172601		TSTB	50027:	RPTFLG	;IF REPORT HAS BEEN REQUESTED THEN:
1440	010716	001403			BEQ	50043:		
1441	010720	105067	172573		CLRB	50043:	RPTFLG	;CLR THE FLAG,
1442	010724				DORPT	50043:		;PRINT THE PERFORMANCE REPORT.
1443	010726	104424			50043:			TRAP C:DRPT
1444	010726	004767	006150		JSR	50044:	PC,FIRSTU	;SET UP FOR FIRST UNIT.
1445	010732				50044:			

1446	010732	026527	002604	177777	CMP	DEVTBL(R5),#END	;WHILE THERE ARE MORE DEVICES:
1447	010740	001450			BEQ	500458	
1448	010742	032767	000400	172450	BIT	#MOD.CO,CMDWRD	;IF CMD IS REVERSE THEN:
1449	010750	001421			BEQ	500468	
1450	010752	032765	000002	003502	BIT	#X0.BOT,EOTFLG(R5)	;IF NOT AT BOT
1451	010760	001014			BNE	500478	
1452	010762	032765	000001	003502	BIT	#X0.EOT,EOTFLG(R5)	;BUT IF AT EOT
1453	010770	001406			BEQ	500508	
1454	010772	105767	172526		TSTB	ALLEOT	;AND ALL OTHERS AT EOT
1455	010776	001402			BEQ	500518	
1456	011000	004767	001360		JSR	PC,GOWAIT	;THEN WAIT FOR CMD END
1457							;IF NOT ALL AT EOT, DO NOT WAIT
1458	011004			500518;			
1459							;NOT AT BOT, AND NOT AT EOT
1460	011004	000402			BR	500528	
1461	011006			500508;			
1462	011006	004767	001352		JSR	PC,GOWAIT	;WAIT FOR INT,CHECK STAT
1463							
1464	011012			500528;			
1465							
1466							
1467	011012			500478;			
1468	011012	000420			BR	500538	;ELSE IF CMD IS FORWARD:
1469	011014			500468;			
1470	011014	032765	000001	003502	BIT	#X0.EOT,EOTFLG(R5)	
1471	011022	001404			BEQ	500548	
1472	011024	032767	000001	172366	BIT	#CMD.CO,CMDWRD	
1473	011032	001003			BNE	500558	
1474	011034			500548;			
1475							;IF NOT AT EOT OR NOT A MOTION CMD THEN:
1476	011034	004767	001324		JSR	PC,GOWAIT	;WAIT FOR INT,CHECK STATUS.
1477							
1478	011040	000405			BR	500568	
1479	011042			500558;			
1480	011042	105767	172456		TSTB	ALLEOT	;IFB ALLEOT NE #0 THEN
1481	011046	001402			BEQ	500578	
1482	011050	004767	001310		JSR	PC,GOWAIT	
1483							
1484	011054			500578;			
1485							
1486	011054			500568;			
1487							
1488	011054			500538;			
1489	011054	004767	006070		JSR	PC,NEXTU	;FIND NEXT UNIT IN TEST CYCLE.
1490							
1491	011060	000724			BR	500448	
1492	011062			500458;			
1493	011062	000207			RTS	PC	;RETURN.
1494							
1495				:	THIS SUBROUTINE STORES THE ASCII FOR THE CURRENT COMMAND AND PREVIOUS		
1496				:	COMMAND IN THE STANDARD ERROR MESSAGE. ON ENTRY LOCATION CMDWRD		
1497				:	CONTAINS CURRENT CMD AND LOCATION PCMDWD CONTAINS PREVIOUS CMD.		
1498				:	INPUTS;		
1499				:	OUTPUTS;		
1500				:	REGISTERS; R3, R4.		
1501				:	CALLS; GCMDA		
1502				:			

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 069

```

1503 011064 016704 172330      CMDAC:: MOV      CMDWRD,R4,LET R4 := CMDWRD      ,R4 = CMD BINARY.
1504 011070 004767 000042      JSR      PC,GCMDA      ,GET CMD ASCII.
1505 011074 112367 175340      MOV      (R3)+,STAER1+2      ,MOVE CMD ASCII
1506 011100 112367 175335      MOV      (R3)+,STAER1+3      ,
1507 011104 111367 175332      MOV      (R3),STAER1+4      ,INTO MSG.
1508 011110 016704 172310      MOV      PCMDWD,R4      ,R4 = PREVIOUS CMD BINARY.
1509 011114 004767 000016      JSR      PC,GCMDA      ,GET CMD ASCII.
1510 011120 112367 175430      MOV      (R3)+,STAER7+24      ,MOVE CMD ASCII
1511 011124 112367 175425      MOV      (R3)+,STAER7+25      ,
1512 011130 111367 175422      MOV      (R3),STAER7+26      ,INTO MSG.
1513 011134 000207      RTS      PC      ,RETURN. GO EXECUTE NEXT FUNCTION.
1514
1515
1516      ;      SUBROUTINE TO FIND THE ASCII EQUIVALENT OF THE COMMAND IN R4.
1517      ;      ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
1518      ;      INPUTS:      R4 = PRESENT COMMAND WORD.
1519      ;      OUTPUTS:      R3 = ADDRESS OF PRESENT COMMAND ASCII.
1520      ;      REGISTERS:
1521      ;      CALLS:
1522
1523 011136 005003      GCMDA:: CLR      R3,LET R3 := #0      ,INIT CMD TBL POINTER.
1524 011140 500601:      CMP      CMDTBL(R3),R4      ,UNTIL CURRENT CMD IS FOUND:
1525 011140 026304 003752      BEQ      500611
1526 011144 001403      ADD      #2,R3      ,LET R3 := R3 + #2      ,SEARCH CMD TABLE.
1527 011146 062703 000002      BR      500601
1528 011152 000772
1529 011154 500611:      MOV      R3,R4      ,LET R4 := R3
1530 011154 010304      ASR      R3      ,POINT TO ASCII FOR THAT COMMAND
1531 011156 006203      NOP
1532 011160 000240      ADD      R4,R3
1533 011162 060403      ADD      #CMDASC,R3
1534 011164 062703 004040      RTS      PC      ,RETURN.
1535 011170 000207
1536
1537      ;      THIS SUBROUTINE LOADS THE TS05 COMMAND PACKET FROM ONE
1538      ;      ENTRY IN THE SEQUENCE TABLE.
1539      ;      INPUTS:
1540      ;      OUTPUTS:
1541      ;      REGISTERS:      R2, R3.
1542      ;      CALLS:      GENPAT.
1543
1544 011172 005067 172230      SETUP:: CLR      CMDLG      ,CLR CMD LOGGING CODE(DISABLES LOGGING)
1545 011176 012167 171126      MOV      (R1)+,CMDPKT      ,LOAD THE COMMAND WORD.
1546 011202 011167 171130      MOV      (R1),CMDPKT+CP.CNT      ,LOAD THE BYTE/RECORD/FILE COUNT.
1547 011206 011167 172204      MOV      (R1),BRFCNT      ,SAVE BRFCNT FOR THIS COMMAND.
1548 011212 016702 171112      MOV      CMDPKT,R2      ,GET CMD.
1549 011216 042702 177740      BIC      #NLMID.C,R2      ,CLR ALL BUT CMD BITS.
1550 011222 010203      MOV      R2,R3      ,SAVE IT TWICE.
1551 011224 162703 000010      SUB      #CMD.C3,R3      ,POSITION COMMAND?
1552 011230 001003      BNE      21      ,BR IF NOT.
1553 011232 011167 171074      MOV      (R1),CMDPKT+2      ,MOVE BPCR IN 2ND PKT WORD FOR POSITION CMD.
1554 011236 000464      BR      31
1555 011240 026727 171064 100011 21:      CMP      CMDPKT,#WTH      ,IF CMD IS A WRITE TAPE MARK THEN:
1556 011246 001003      BNE      500621      ,WTH LOGGING CODE IS 2.
1557 011250 012767 000002 172150      MOV      #2,CMDLG
1558
1559 011256      500621:

```

1560	011256	010203			MOV	R2,R3		
1561	011260	162703	000001		SUB	#CMD.CO,R3		;IS IT A READ?
1562	011264	001017			BNE	1:		;BR IF NOT.
1563	011266	016767	172116	171036	MOV	DATARD,CMDPKT+CP.ADL		;IF SO, LOAD THE BUFFER ADDR.
1564	011274	032767	000400	171026	BIT	#MOD.CO,CMDPKT		;IF CMD IS A READ REV THEN:
1565	011302	001404			BEQ	50063:		
1566	011304	012767	000004	172114	MOV	#4,CMDLG		;LOGGING CODE IS 4.
1567								;ELSE - IF CMD IS A READ FWD:
1568	011312	000403			BR	50064:		
1569	011314							
1570	011314	012767	000006	172104	MOV	#6,CMDLG		;LOGGING CODE IS 6.
1571								
1572	011322							
1573	011322	000432			BR	3:		;CONTINUE.
1574	011324	010203			MOV	R2,R3		;IS IT
1575	011326	162703	000004		SUB	#CMD.C2,R3		;A SET CHARACTERISTICS CMD?
1576	011332	001014			BNE	4:		;BR IF NOT.
1577	011334	012767	002474	170770	MOV	#SCHBK,CMDPKT+CP.ADL		;SET UP ADR LO FOR SET CHAR.
1578	011342	012767	000012	170766	MOV	#SCHCNT,CMDPKT+CP.CNT		;SET BUFFER EXTENT
1579	011350	011167	171126		MOV	(R1),SCHBK+6		;STORE CHARACTERISTIC CODE IN SCH BLOCK.
1580	011354	016767	172152	171122	MOV	TSUNT,SCHBK+10		;UNIT #
1581	011362	000412			BR	3:		;CONTINUE.
1582	011364	010203			MOV	R2,R3		;IS IT
1583	011366	162703	000006		SUB	#CMD.C1!CMD.C2,R3		;A DIAGNOSTIC (DIA) CMD?
1584	011372	001006			BNE	3:		;BR IF NOT.
1585	011374	012767	000020	170734	MOV	#DIACNT,CMDPKT+CP.CNT		;LOAD BUFFER EXTENT.
1586	011402	012767	003406	170722	MOV	#DIABLK,CMDPKT+CP.ADL		;LOAD BUFFER ADR LOW.
1587	011410	005721			TST	(R1),		;POINT TO N (NUMBER OF TIMES TO EXECUTE THIS INS
1588	011412	012167	171776		MOV	(R1),NCNT1		;SAVE NUMBER OF OPERATIONS
1589	011416	005067	171770		CLR	NCNT		;CLEAR OPERATION COUNTER.
1590	011422	012167	172020		MOV	(R1),PATERN		;SAVE PATTERN CODE FOR CURRENT CMD.
1591	011426	010203			MOV	R2,R3		;IS IT
1592	011430	162703	000005		SUB	#CMD.CO!CMD.C2,R3		;A WRITE?
1593	011434	001010			BNE	5:		;BR IF NOT.
1594	011436	016767	171744	170666	MOV	DATAWT,CMDPKT+CP.ADL		;LOAD WRITE BUFFER LO ORDER.
1595	011444	004767	000106		JSR	PC,GENPAT		;GO GENERATE THE WRITE PATTERN.
1596	011450	012767	000002	171750	MOV	#2,CMDLG		;WRITE LOGGING CODE IS 2.
1597	011456	032767	000100	170644	BIT	#VFY.C,CMDPKT		;IF DATA VERIFICATION IS REQUIRED:
1598	011464	001407			BEQ	50065:		
1599	011466	112767	000001	172022	MOVB	#1,VFYFLG		;SET VERIFY FLAG.
1600	011474	042767	000100	170626	BIC	#VFY.C,CMDPKT		;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
1601								;IF DATA VERIFICATION IS NOT REQUIRED:
1602	011502	000402			BR	50066:		
1603	011504							
1604	011504	105067	172006		CLRB	VFYFLG		;CLR VERIFY FLAG.
1605								
1606	011510							
1607	011510	016767	171704	171706	MOV	CMDWRD,PCMDWD		;SAVE PREVIOUS CMD WORD.
1608	011516	016767	170606	171674	MOV	CMDPKT,CMDWRD		;SAVE PRESENT CMD WORD.
1609	011524	105767	171770		TSTB	SWBFLG		;IF SWAP BYTES IS ENABLED:
1610	011530	001403			BEQ	50067:		
1611	011532	052767	010000	170570	BIS	#SWB.C,CMDPKT		;SET SWAP BIT IN COMMAND.
1612								
1613	011540							
1614	011540	042767	004000	170562	BIC	#BRF.C,CMDPKT		;CLR BRF BIT (INTERNAL ONLY).
1615	011546	016767	170556	171646	MOV	CMDPKT,CMDSAV		;SAVE 1ST WORD OF COMMAND PACKET.
1616	011554	000207			RTS	PC		;RETURN.

```

1617
1618
1619
1620
1621
1622
1623
1624
1625 011556 016703 171664
1626 011562 006303
1627 011564 016704 171626
1628 011570 005204
1629 011572 042704 000001
1630 011576 162704 000002
1631 011602 016702 171600
1632 011606 062702 000002
1633 011612 004773 011620
1634 011616 000207
1635
1636
1637
1638
1639 011620 011642
1640 011622 011700
1641 011624 011720
1642 011626 011730
1643 011630 011754
1644 011632 011766
1645 011634 012000
1646 011636 012020
1647 011640 012052
1648
1649
1650
1651 011642 012703 000400
1652 011646 162704 000002
1653 011652 100411
1654 011654 010322
1655 011656 062703 001002
1656 011662 020327 001000
1657 011666 001002
1658 011670 012703 000400
1659
1660 011674
1661 011674 000764
1662
1663 011676 000207
1664
1665
1666
1667 011700 012703 177777
1668 011704 162704 000002
1669 011710 100402
1670 011712 010322
1671 011714 000773
1672
1673 011716 000207

;      THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERATE
;      THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
;      INPUTS:
;      OUTPUTS:
;      REGISTERS:      R2, R3, R4.
;      CALLS:          PATR0 - PATR7

GENPAT:  MOV     PATERN,R3      ;SETUP PATTERN ROUTINE POINTER
        ASL     R3
        MOV     BRFCNT,R4      ;SET LENGTH OF WRITE BFR
        INC     R4
        BIC     #1,R4          ;ROUNDED UP TO NEXT WORD
        SUB     #2,R4          ;WITH FIRST WORD RESERVED
        MOV     DATAW,R2      ;FOR RECORD COUNT
        ADD     #2,R2
        JSR     PC,8PATTBL(R3) ;GO GENERATE THE APPROPRIATE PATTERN.
        RTS     PC             ;RETURN TO SETUP SUBROUTINE.

;TS05 WRITE PATTERN LOOKUP TABLE. USED TO JSR TO THE
;CORRECT DATA PATTERN GENERATING ROUTINE.

PATTBL: PATR0
        PATR1
        PATR2
        PATR3
        PATR4
        PATR5
        PATR6
        PATR7
        PATR8

;INCREMENTING PATTERN. 0 - 377.

PATR0:  MOV     #400,R3;LET R3 := #400
14:     SUB     #2,R4;LET R4 := R4 - #2 ;DECREMENT WORD COUNT.
        BMI     24          ;BR IF DONE.
        MOV     R3,(R2)+     ;STORE DATA WORD.
        ADD     #1002,R3     ;UPDATE PATTERN.
        CMP     R3,#1000     ;IF PATTERN HAS WRAPPED AROUND THEN:
        BNE     500704
        MOV     #400,R3      ;INIT THE PATTERN AGAIN.

500704: BR      14           ;DO IT AGAIN.

24:     RTS     PC           ;RETURN.

;ALL ONE'S PATTERN.

PATR1:  MOV     #-1,R3
ZROPAT: SUB     #2,R4
        BMI     14          ;DONE?,BR IF YES.
        MOV     R3,(R2)+     ;IF NOT LOAD NEXT BYTE WITH PATTERN.
        BR      ZROPAT      ;DO IT AGAIN.

14:     RTS     PC           ;RETURN.

```

```

1674
1675
1676
1677 011720 005003
1678 011722 004767 177756
1679 011726 000207
1680
1681
1682
1683 011730 012703 000401
1684 011734 162704 000002
1685 011740 100404
1686 011742 010322
1687 011744 006303
1688 011746 005503
1689 011750 000771
1690 011752 000207
1691
1692
1693
1694 011754 012703 177376
1695 011760 004767 177750
1696 011764 000207
1697
1698
1699
1700
1701 011766 012703 125125
1702 011772 004767 177706
1703 011776 000207
1704
1705
1706
1707 012000 012703 177400
1708 012004 162704 000002
1709 012010 100402
1710 012012 010322
1711 012014 000773
1712 012016 000207
1713
1714
1715
1716 012020 162704 000002
1717 012024 100411
1718 012026 066767 171402 171376
1719 012034 066767 171372 171372
1720 012042 016722 171366
1721 012046 000764
1722 012050 000207
1723
1724
1725
1726 012052 000207
1727
1728
1729
1730

```

;ALL ZEROES PATTERN.
 PATR2:: CLR R3 ;CLR PATTERN REGISTER.
 JSR PC,ZROPAT ;GO GENERATE IT.
 RTS PC ;RETURN.
 ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
 PATR3:: MOV #401,R3 ;INIT PATTERN REGISTER.
 WLKZRO: SUB #2,R4;LET R4 := R4 - #2 ;DECREMENT WORD COUNT.
 BMI 1# ;BR IF DONE.
 MOV R3,(R2)+ ;LOAD DATA.
 ASL R3 ;SHIFT PATTERN.
 ADC R3 ;ADD CARRY BACK INTO PATTERN.
 BR WLKZRO ;DO IT AGAIN.
 1#: RTS PC ;RETURN.
 ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.
 PATR4:: MOV #177376,R3 ;INIT PATTERN REGISTER.
 JSR PC,WLKZRO ;GO GENERATE IT.
 RTS PC ;RETURN.
 ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
 ;COMPLEMENTED.
 PATR5:: MOV #125125,R3 ;INIT PATTERN REGISTER.
 JSR PC,ZROPAT ;GO GENERATE IT.
 RTS PC ;RETURN.
 ;ALTERNATING BYTES OF 000 AND 377.
 PATR6:: MOV #177400,R3 ;INIT PATTERN REGISTER.
 1#: SUB #2,R4 ;DECREMENT WORD COUNT.
 BMI 2# ;BR IF DONE.
 MOV R3,(R2)+ ;LOAD DATA.
 BR 1# ;DO IT AGAIN.
 2#: RTS PC ;RETURN.
 ;RANDOM PATTERN GENERATOR
 PATR7:: SUB #2,R4 ;DECREMENT WORD COUNT
 BMI GIT ;BR IF DONE.
 ADD RANS,RANB ;GET NEW #.
 ADD RANB,RANS ;SAVE #.
 MOV RANS,(R2)+ ;CONTINUE.
 BR PATR7 ;RETURN
 GIT: RTS PC
 ; NO PATTERN GENERATION.
 PATR8:: RTS PC ;RETURN.
 ; THIS SUBROUTINE INITIATES TS05 COMMAND EXECUTION
 ; AND CHECKS FOR TS05 RESPONSE.
 ; INPUTS:

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 073

```

1731      ;      OUTPUTS:
1732      ;      REGISTERS:      R2, R3.
1733      ;      CALLS:      DROP, MOVMSG, FIRSTU, NEXTU, WSSR.
1734
1735 012054 012767 177777 171354 EXCUTE:: MOV      #-1, TIME1      ;INIT TIMEOUT COUNTER.
1736 012062      500714: ;REPEAT      ;WAIT -
1737 012062 005367 171350      DEC      TIME1      ;UPDATE TIMEOUT COUNTER.
1738 012066 005767 171344      TST      TIME1      ;IF TIMED OUT:
1739 012072 001011      BNE      500724
1740 012074 004767 000634      JSR      PC, MOVMSG      ;MOVE CURRENT PACKET MSG.
1741 012100      ERRDF      2, NSSRM, STAERM      ;REPORT TS05 NOT READY
1742      012100 104455      TRAP      C:ERDF
1743      012102 000002      .WORD      2
1744      012104 004536      .WORD      NSSRM
1745      012106 006120      .WORD      STAERM
1746 012110 004767 005064      JSR      PC, DROP
1747 012114 000522      BR      EXCRN      ;DROP THE UNIT.
1748      ;RETURN.
1749
1750 012116      500724:      BIT      #TS.SSR, @TSSR(R5)      ;WAIT UNTIL DEVICE IS READY.
1751 012116 032775 000200 002524      BEQ      500714
1752 012124 001756      CMP      CMDWRD, #SCH      ;IF WE ARE DOING A SET CHAR CMD THEN:
1753 012126 026727 171266 140004      BNE      500734
1754 012134 001022      MOV      R5, RSSAVE      ;SAVE CURRENT DEVICE POINTER.
1755 012136 010567 171310      JSR      PC, FIRSTU      ;FIND FIRST UNIT.
1756 012142 004767 004734      500744:      CMP      DEVTBL(R5), #END      ;WHILE DEVTBL(R5) NE #END DO
1757 012146      BEQ      500754
1758 012146 026527 002604 177777      JSR      PC, WSSR      ;WAIT FOR UNIT READY OR TIME OUT.
1759 012154 001405      JSR      PC, NEXTU      ;FIND NEXT UNIT.
1760 012156 004767 000516
1761 012162 004767 004762      BR      500744
1762
1763 012166 000767      500754:      MOV      RSSAVE, R5      ;RESTORE CURRENT DEVICE POINTER.
1764 012170      MOV      MSGPKA(R5), SCHBK      ;SET UP ADR OF MSG PKT IN SCH BLOCK.
1765 012174 016705 171256
1766 012174 016567 002544 170272      500734:      MOV      MSGPKA(R5), R3      ;ADR OF THIS UNIT'S MSG PACKET.
1767 012202      CLR      R2      ;CLR COUNTER.
1768 012202 016503 002544      500764:      CMP      R2, #MSGCNT      ;WHILE THERE ARE MORE LOCATIONS:
1769 012206 005002      BEQ      500774
1770 012210 020227 000020      MOV      #-1, (R3)      ;INIT THE MSG PACKET WITH ALL 1'S
1771 012214 001405      ADD      #2, R2      ;UPDATE COUNTER.
1772 012216 012723 177777
1773 012222 062702 000002      BR      500764
1774
1775 012226 000770      500774:      TSTB      DINT      ;ARE INTERRUPTS DISABLED.
1776 012230      BNE      14      ;BR IF YES.
1777 012234 105767 167756      CMPB      INTFLG(R5), #1      ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
1778 012236 001023      BLE      501004
1779 012244 126527 003472 000001      MOV      @TSSR(R5), TSSREG      ;FREEZE THE CURRENT STATUS REG FOR PRINT
1780 012246 003412      ERRDF      15, TOOMM, STAERM      ;REPORT TOO MANY INTERRUPTS.
1781 012254      TRAP      C:ERDF
1782      012254 104455      .WORD      15
1783      012256 000017      .WORD      TOOMM
1784      012260 004727      .WORD      STAERM
1785      012262 006120

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 074

```

1780 012264 004767 004710      JSR    PC,DROPU      ;DROP THE UNIT
1781 012270 000434              BR      EXCRTN      ;RETURN - UNIT HAS BEEN DROPPED.
1782
1783 012272                    501001:
1784 012272 005065 003472      CLR    INTFLG(R5)      ;CLR INTERRUPT FLAG FOR THIS DEV.
1785 012276 052767 000200 170024  BIS    #IE.C,CMDPKT      ;SET INT ENABLE BIT.
1786 012304 105767 171161      11:    TSTB    ERRREC,IFB ERRREC EQ #0 THEN ;IF NOT RETRYING
1787 012310 001005              BNE    501011
1788 012312 005265 003376      INC     RECCNT(R5)      ;LET RECCNT(R5) := RECCNT(R5) + #1
1789 012316 016577 003376 171062  MOV     RECCNT(R5),@DATAWT ;THEN UPDATE REC COUNT TO WRITE IT ON TAPE
1790
1791 012324                    501011:
1792 012324 012775 002330 002514  MOV     #CMDPKT,@TSDB(R5) ;LOAD TSDB WITH CMDPKT ADDRESS
1793                                ;THIS INITIATES COMMAND EXECUTION.
1794 012332 032775 000200 002524  BIT     #TS.SSR,@TSSR(R5) ;IF READY DID NOT DROP THEN:
1795 012340 001410              BEQ     501021
1796 012342 004767 000366      JSR     PC,MOVMSG      ;MOVE CURRENT MESSAGE PACKET TO COMMON.
1797 012346              ERRDF    3,TOERM,STAERM      ;REPORT NO TS05 RESPONSE.
                                TRAP    C#ERDF
                                .WORD   3
                                .WORD   TOERM
                                .WORD   STAERM
1798 012356 004767 004616      JSR     PC,DROPU      ;DROP THE UNIT
1799
1800 012362                    501021:
1801 012362 000207      EXCRTN: RTS     PC      ;RETURN.
1802
1803      ; THIS SUBROUTINE WAITS FOR THE TS05 INERRUPT OR DONE BIT TO SET AND ALLOWS THE
1804      ; OPERATOR TO TRANSFER CONPOL TO THE SUPERVISOR.
1805      ; UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
1806      ; LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NECESSARY.
1807      ; INPUTS:
1808      ; OUTPUTS:
1809      ; REGISTERS: R2, R3.
1810      ; CALLS: DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.
1811
1812 012364 012767 177777 171044  GOWAIT: MOV     #-1,TIME1      ;INIT TIME OUT COUNTER.
1813 012372                    501031: ;REPEAT UNTIL INTERRUPT OCCURES:
1814 012372              BREAK      ;GO TO THE SUPER TO ALLOW TTY INPUT.
                                TRAP    C#BRK
1815 012374 026727 171020 102010  CMP     CMDWRD,@RWD      ;IF COMMAND WAS REWIND THEN:
1816 012402 001014              BNE    501041
1817 012404              DELAY     10.      ;WAIT EXTRA MSECS EACH LOOP.
                                MOV     #10.,(PC)+
                                .WORD   0
                                MOV     L#DLY,(PC)+
                                .WORD   0
                                DEC     -6(PC)
                                BNE     -.4
                                DEC     -22(PC)
                                BNE     -.20
1818 012434                    501041:
1819 012434 026727 170760 105010  CMP     CMDWRD,#SFF      ;IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
1820 012442 001404              BEQ     501051
1821 012444 026727 170750 105410  CMP     CMDWRD,#SFR
1822 012452 001014              BNE    501061
1823 012454                    501051:

```

GLOBAL AREAS
GLOBAL SUBROUTINES SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 075

```

1824 012454          DELAY 12.          ;ADD DELAY FOR SPACE TAPE MARK COMMANDS
      012454 012727 000014              MOV #12..(PC)+
      012460 000000                      .WORD 0
      012462 016727 167430              MOV L$DLY,(PC)+
      012466 000000                      .WORD 0
      012470 005367 177772              DEC -6(PC)
      012474 001375                      BNE -4
      012476 005367 177756              DEC -22(PC)
      012502 001367                      BNE -20

1825 012504          50106:
1826 012504 105767 167502              TSTB DINT          ;IF INTERRUPTS ARE ENABLED.
1827 012510 001003                      BNE 50107:
1828 012512 016502 003472              MOV INTFLG(R5),R2 ;FETCH INTERRUPT OCCURRED FLAG.
1829
1830 012516 000406                      BR 50110:
1831 012520          50107:
1832 012520 012703 000200              MOV #TS.SSR,R3 ;SET UP A MASK FOR THE DONE BIT.
1833 012524 005103                      COM R3
1834 012526 017502 002524              MOV #TSSR(R5),R2 ;FETCH DONE BIT.
1835 012532 040302                      BIC R3,R2
1836
1837 012534          50110:
1838 012534 005367 170676              DEC TIME1 ;UPDATE TIMEOUT COUNTER.
1839 012540 005702                      TST R2 ;REPEAT UNTIL INTERRUPT OR READY OCCURES.
1840 012542 001003                      BNE 50111:
1841 012544 005767 170666              TST TIME1
1842 012550 001310                      BNE 50103:
1843 012552          50111:
1844 012552 005767 170660              TST TIME1 ;IF TIME OUT HAS OCCURRED:
1845 012556 001022                      BNE 50112:
1846 012560 016577 003376 170620      MOV RECCNT(R5),R0DATAWT
1847 012566 005377 170614              DEC R0DATAWT
1848 012572 004767 000136              JSR PC,MOVMSG ;MOVE CURRENT MSG PACKET TO COMMON AREA.
1849 012576          ERRDF 4,NOINTM,STAERM ;REPORT NO INTERRUPT.
      012576 104455                      TRAP C$ERDF
      012600 000004                      .WORD 4
      012602 004670                      .WORD NOINTM
      012604 006120                      .WORD STAERM

1850 012606 004767 004366              JSR PC,DROPU ;DROP THE UNIT.
1851 012612 012703 003472              MOV #ENDERF,R3 ;LET R3 := #ENDERF
1852 012616 004767 000042              JSR PC,CLRERR ;CLEAR ALL ERROR FLAGS
1853
1854 012622 000417                      BR 50113:
1855 012624          50112:
1856 012624 004767 000104              JSR PC,MOVMSG ;MOVE CURRENT MSG. PACKET TO COMMON AREA.
1857 012630 004767 000164              JSR PC,RECUO ;UPDATE THE RECORD COUNT.
1858 012634 004767 000350              JSR PC,CHKERR ;CHECK FOR STATUS ERRORS.
1859 012640 105767 170617              TSTB WRTYFG ;IFB WRTYFG EQ #0 THEN
1860 012644 001006                      BNE 50114:
1861 012646 004767 002714              JSR PC,LOG ;LOG BYTES AND ERRORS.
1862 012652 012703 003472              MOV #ENDERF,R3 ;LET R3 := #ENDERF
1863 012656 004767 000002              JSR PC,CLRERR ;CLEAR ALL ERROR FLAGS
1864
1865 012662          50114:
1866
1867 012662          50113:
1868 012662 000207          RTS PC ;RETURN IF DONE.

```

```

1869
1870      ; SUBROUTINE TO CLEAR FLAGS.
1871      ; INPUTS:      R3 = LWA TO BE CLEARED + 2.
1872      ; OUTPUTS:
1873      ; REGISTERS:   R2
1874      ; CALLS:
1875
1876 012664 012702 003460 CLRERR:: MOV    #BGNFLG,R2      ;LET R2 := #BGNFLG
1877 012670 501154: REPEAT
1878 012670 005022      CLR    (R2)+      ;LET (R2)+ := #0
1879 012672 020203      CMP     R2,R3      ;UNTIL R2 EQ R3
1880 012674 001375      BNE     501154
1881 012676 000207      RTS      PC
1882
1883
1884      ; SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
1885      ; INPUTS:
1886      ; OUTPUTS:
1887      ; REGISTERS:
1888      ; CALLS:
1889
1890 012700
1891 012700 012767 177777 170530 WSSR:: MOV     #-1,TIME1      ;INIT TIMEOUT COUNTER.
1892 012706 501164: REPEAT UNTIL DEV READY OR TIMEOUT:
1893 012706 BREAK      ;BREAK TO THE SUPERVISOR.
1894 012710 104422 TRAP     C#BRK
1895 012714 005367 170522 DEC     TIME1      ;UPDATE TIMEOUT COUNTER.
1896 012722 032775 000200 002524 BIT     #TS.SSR,#TSSR(R5) ;UNTIL #TS.SSR SETIN #TSSR(R5) OR TIME1 EQ #0
1897 012724 001003 BNE     501174
1898 012730 005767 170506 TST     TIME1
1899 012732 001366 BNE     501164
1900 012732 000207 501174: RTS      PC      ;RETURN.
1901
1902
1903
1904      ; SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
1905      ; TO UPDATE THE CURRENT TERMINATION CLASS CODE.
1906      ; INPUTS:
1907      ; OUTPUTS:
1908      ; REGISTERS:   R2, R3.
1909      ; CALLS:
1910
1911 012734 017567 002524 170512 MOVMSG: MOV    #TSSR(R5),TSSREG      ;FREEZE THE STATUS REG CONTENTS
1912 012742 016702 170506 MOV    TSSREG,R2      ;EXTRACT THE TERMINATION CLASS CODE.
1913 012746 042702 177761 BIC    #TSC.TCC,R2
1914 012752 010267 170472 MOV    R2,CTCC      ;AND SAVE IT
1915 012756 006267 170466 ASR    CTCC
1916 012762 016503 002544 MOV    MSGPKA(R5),R3      ;ADR OF THIS DEVICE'S MSG.
1917 012766 005002 CLR    R2      ;CLR COUNTER.
1918 012770 501204:
1919 012770 020227 000020 CMP    R2,#MSGCNT      ;WHILE THERE ARE MORE LOCATIONS:
1920 012774 001405 BEQ    501214
1921 012776 012362 002354 MOV    (R3)+,MSGPKT(R2) ;MOVE MSG TO COMMON AREA.
1922 013002 062702 000002 ADD    #2,R2      ;UPDATE COUNTER.
1923
1924 013006 000770 BR     501204

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 077

```

1925 013010
1926 013010 016767 167346 170464 501211: MOV MSGPKT+MS.X50,EOTFLG ;MOVE XSTATO TO EOT FLAG.
1927 013016 000207 RTS PC
1928
1929 ; SUBROUTINE TO ADJUST THE RECORD COUNT.
1930 ; INPUTS:
1931 ; OUTPUTS:
1932 ; REGISTERS:
1933 ; CALLS:
1934
1935 013020 105767 170441 RECUD:: TSTB RECLOG ;IF RECORD HAS NOT BEEN LOGGED:
1936 013024 001070 BNE 501221
1937 013026 005365 003376 DEC RECCNT(R5) ;LET RECCNT(R5) := RECCNT(R5) - #1
1938 013032 032767 000001 170410 BIT #BIT0,CTCC ;IF TAPE MOVED
1939 013040 001057 BNE 501231
1940 013042 032767 100000 167316 BIT #X2.OPM,MSGPKT+MS.X52
1941 013050 001453 BEQ 501231
1942 013052 105267 170407 INCB RECLOG ;SET RECORD LOGGED.
1943 013056 026727 170336 102010 CMP CMDWRD,#RWD ;IF THIS IS A REWIND CMD:
1944 013064 001003 BNE 501241
1945 013066 005065 003376 CLR RECCNT(R5) ;CLEAR RECORD COUNT.
1946
1947 013072 000442 BR 501251
1948 013074
1949 013074 032767 004000 170316 501241: BIT #BRF.C,CMDWRD ;IF BRF USED, UPDATE RECORD COUNT.
1950 013102 001436 BEQ 501261
1951 013104 032767 000400 170306 BIT #MOD.CO,CMDWRD ;IF A FORWARD CMD:
1952 013112 001007 BNE 501271
1953 013114 032767 000400 170302 BIT #MOD.CO,PCMDWD ;IF PREV CMD WAS A FWD ALSO:
1954 013122 001002 BNE 501301
1955 013124 005265 003376 INC RECCNT(R5) ;INCREMENT RECORD COUNT.
1956
1957 013130 501301:
1958
1959 013130 000423 BR 501311 ;IF REVERSE CMD:
1960 013132
1961 013132 032767 000400 170264 501271: BIT #MOD.CO,PCMDWD ;IF PREVIOUS CMD WAS A REV ALSO:
1962 013140 001417 BEQ 501321
1963 013142 032765 000002 003502 BIT #X0.BOT,EOTFLG(R5) ;WHEN NOT AT BOT THEN
1964 013150 001013 BNE 501331
1965 013152 105767 170313 TSTB ERRREC ;CHECK THE ERROR RETRY INDICATOR
1966 013156 001406 BEQ 21 ;BR, IF WE ARE NOT NOW IN ERROR RETRY
1967 013160 105767 170332 TSTB VFYFLG ;CHECK THE WRITE VERIFY INDICATOR
1968 013164 001403 BEQ 21 ;BR, IF WE ARE NOT IN WRT/VFY MODE
1969 013166 105767 170273 TSTB RECLOG ;CHECK IF THIS RECORD HAS BEEN COUNTED
1970 013172 001002 BNE 101 ;BR, IF HAVE ALREADY BUMPED RECORD CNTR.
1971 013174 005365 003376 21: DEC RECCNT(R5) ;DECREMENT RECORD COUNT.
1972 013200 101:
1973
1974 013200 501331:
1975
1976 013200 501321:
1977
1978 013200 501311:
1979
1980
1981 013200 501261:

```

```

1982
1983 013200 501254:
1984
1985 013200 501234:
1986 013200 016577 003376 170200 MOV RECCNT(R5),SDATAWT ;LET SDATAWT := RECCNT(R5)
1987
1988 013206 501224:
1989 013206 000207 RTS PC ;RETURN.
1990
1991 ; THIS IS THE ERROR CHECK SUBROUTINE. AFTER INTERRUPT THIS
1992 ; SUBROUTINE IS CALLED TO CHECK THE TS05 STATUS.
1993 ; IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
1994 ; IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
1995 ; THEN AN ERROR RFC IS REPORTED,
1996 ; INPUTS:
1997 ; OUTPUTS:
1998 ; REGISTERS: R2, R4.
1999 ; CALLS: TCC0-TCC7.
2000
2001 013210 032767 100000 170236 CHKERR: BIT #TS.SC,TSSREG ;IF SPECIAL COND STATUS IS SET THEN:
2002 013216 001441 BEQ 501344
2003 013220 026727 170224 000002 CMP CTCC,#2 ;IF TCC IS NOT 2 THEN:
2004 013226 001405 BEQ 501354
2005 013230 105767 170235 TSTB ERRREC ;IF NOT IN ERROR RECOVERY:
2006 013234 001002 BNE 501364
2007 013236 005265 003336 INC SCCNT(R5) ;INC SC COUNTER.
2008
2009 013242 501364:
2010
2011 013242 501354:
2012 013242 032767 004000 170204 BIT #TS.NXM,TSSREG ;WHEN NON-EXISTANT MEMO
2013 013250 001004 BNE 501374
2014 013252 032767 040000 170174 BIT #TS.UPE,TSSREG
2015 013260 001412 BEQ 501404
2016 013262 501374:
2017 013262 032767 100000 167076 BIT #X2.OPM,MSGPKT+MS.XS2 ;AND TAPE NOT MOVED
2018 013270 001003 BNE 501414
2019 013272 012702 000005 MOV #5,R2 ;SET TCC5 INDEX
2020
2021 013276 000402 BR 501424
2022 013300 501414:
2023 013300 012702 000004 MOVL #4,R2 ;TAPE MOVED, SET TCC4 INDEX
2024
2025 013304 501424:
2026
2027 013304 000402 BR 501434
2028 013306 501404:
2029 013306 016702 170136 MOV CTCC,R2 ;SET DETECTED TCC INDEX
2030
2031 013312 501434:
2032 013312 006302 ASL R2 ;CURRENT TCC X 2.
2033 013314 004772 013414 JSR PC,BTCCRA(R2) ;GO TO THE TCC HANDLING SUBROUTINE.
2034
2035 013320 000426 BR 501444
2036 013322 501344:
2037 013322 032767 004000 170070 BIT #BRF.C,CMDWRD ;IF BRF IS USED IN THIS CMD THEN:
2038 013330 001422 BEQ 501454

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 079

```

2039 013332 005767 167022      TST      MSGPKT+MS.RFC      ;IF THERE IS AN RFC THEN:
2040 013336 001417      BEQ      501460
2041 013340 105767 170151      TSTB     RANDOM      ;IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
2042 013344 001403      BEQ      501470
2043 013346 105767 170144      TSTB     VFYFLG
2044 013352 001411      BEQ      501500
2045 013354      501470:
2046
2047 013354 105767 170141      TSTB     IRE      ;IF NOT IN RANDOM OR IF CMD IS WTV:
2048 013360 001006      BNE      501510      ;IF RFC ERROR REPORTS ARE ALLOWED:
2049 013362 005265 003356      INC      HRDCNT(R5)      ;UPDATE HARD ERROR COUNT
2050 013366      ERRMRD 13,RFCERM,STAERM      ;REPORT RFC ERROR
      013366 104456      TRAP      C1ERMRD
      013370 000015      .WORD    13
      013372 004521      .WORD    RFCERM
      013374 006120      .WORD    STAERM

2051
2052 013376      501510:
2053
2054 013376      501500:
2055
2056 013376      501460:
2057
2058 013376      501450:
2059
2060 013376      501440:
2061 013376 105767 170065      TSTB     RWERR      ;IF A READ/WRITE ERROR HAS OCCURRED THEN:
2062 013402 001403      BEQ      501520
2063 013404 016767 170012 166716      MOV     CMDSAV,CMDPKT      ;RESTORE CMD PACKET AFTER ERROR RECOV.
2064
2065 013412      501520:
2066 013412 000207      RTS      PC      ;RETURN.
2067
2068      ;      ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 - 7.
2069
2070 013414 013434      TCCRA: TCC0
2071 013416 013452      TCC1
2072 013420 013470      TCC2
2073 013422 013600      TCC3
2074 013424 013616      TCC4
2075 013426 014232      TCC5
2076 013430 014330      TCC6
2077 013432 014472      TCC7
2078
2079      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
2080      ;      CONDITION ERROR.
2081      ;      INPUTS:
2082      ;      OUTPUTS:
2083      ;      REGISTERS:
2084      ;      CALLS:
2085
2086 013434 005265 003356      TCC0:: INC      HRDCNT(R5)      ;UPDATE HARD ERROR COUNT.
2087 013440      ERRMRD 5,SCERM,STAERM      ;REPORT SPECIAL CONDITION ERROR.
      013440 104456      TRAP      C1ERMRD
      013442 000005      .WORD    5
      013444 004475      .WORD    SCERM
      013446 006120      .WORD    STAERM

```

```

2088 013450 000207          RTS PC                      ;RETURN.
2089
2090
2091          ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
2092          ;      THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
2093          ;      SUCH AS GOING OFFLINE OR COMING ONLINE.
2094          ;      INPUTS:
2095          ;      OUTPUTS:
2096          ;      REGISTERS:      R2,R4
2097          ;      CALLS:          DROPU
2098
2099 013452          TCC1::  ERRDF   6,ATTNM,STAERM          ;REPORT ATTENTION-UNIT OFF LINE.
013452          104455          TRAP      C:ERDF
013454          000006          .WORD    6
013456          004603          .WORD    ATTNM
013460          006120          .WORD    STAERM
2100 013462 004767 003512    JSR      PC,DROPU          ;DROP THE UNIT.
2101 013466 000207          RTS      PC                  ;RETURN.
2102
2103          ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 2, TAPE STATUS ALERT.
2104          ;      A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
2105          ;      TO THE PROGRAM. BITS OF INTEREST INCLUDE THK, RLS, LET, RLL, BOT, EOT.
2106          ;      INPUTS:
2107          ;      OUTPUTS:
2108          ;      REGISTERS:
2109          ;      CALLS:
2110
2111 013470 032767 000002 166664 TCC2::  BIT      @X0.BOT,MSGPKT+MS,XS0
2112 013476 001404          BEQ      501530
2113 013500 105767 170010    TSTB     EXPBOT
2114 013504 001401          BEQ      501530
2115
2116 013506 000433          BR       TC2RTN                ;IF AT BOT AND BOT IS EXPECTED:
2117                                     ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
2118 013510          501530:
2119 013510 032767 170002 166644 BIT      @X0.RLS!X0.RLL!X0.THK!X0.LET!X0.BOT,MSGPKT+MS,XS0
2120                                     ;IF @X0.RLS!X0.RLL!X0.THK!X0.LET!X0.BOT SETIN MSGPKT+MS,XS0 THEN
2121
2122 013516 001427          BEQ      501540
2123
2124 013520 105767 167771    TSTB     RANDOM                ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
2125 013524 001403          BEQ      501550                ;IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
2126 013526 105767 167764    TSTB     VFYFLG
2127 013532 001421          BEQ      501560
2128 013534          501550:
2129
2130 013534 105767 167761    TSTB     IRE                    ;IF NOT IN RANDOM OR IF CMD IS WTV:
2131 013540 001016          BNE      501570                ;IF RFC ERROR REPORTS ARE ALLOWED:
2132 013542 105767 167723    TSTB     ERRREC
2133 013546 001403          BEQ      501600                ;IF WE ARE IN ERROR RECOVERY THEN:
2134 013550 105267 167714    INCB     UNREC                ;SET UNRECOVERABLE FLAG FOR LOG.
2135                                     ;ELSE - IF NOT IN ERROR RECOVERY:
2136 013554 000402          BR       501610
2137 013556          501600:
2138 013556 005265 003336    INC      SCCNT(R5)            ;INCREMENT THE SPEC COND COUNTER.
2139
2140 013562          501610:

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 081

```

2141 013562 005265 003356      INC      HRDCNT(R5)      ;UPDATE HARD ERROR COUNT.
2142 013566      EPRHRD    7,TSAM,STAERM      ;REPORT TAPE STATUS ALERT.
      013566      104456      TRAP      C0ERHRD
      013570      000007      .WORD    7
      013572      004705      .WORD    TSAM
      013574      006120      .WORD    STAERM

2143
2144 013576      501570:
2145
2146 013576      501560:
2147
2148 013576      501540:
2149
2150 013576 000207      TC2RTN:   RTS    PC      ;RETURN.
2151
2152
2153      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 3, FUNCTION REJECT.
2154      ;      THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST ARE
2155      ;      RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
2156      ;      INPUTS:
2157      ;      OUTPUTS:
2158      ;      REGISTERS:      R2,R4
2159      ;      CALLS:          DROPU
2160
2161 013600      TCC3::  ERD0F    8,FUNRM,STAERM      ;REPORT FUNCTION REJECT.
      013600      104455      TRAP      C0ERD0F
      013602      000010      .WORD    8
      013604      004622      .WORD    FUNRM
      013606      006120      .WORD    STAERM

2162 013610 004767 003364      JSR      PC,DROPU      ;DROP THE UNIT.
2163 013614 000207      RTS      PC      ;RETURN.
2164
2165      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 4, RECOVERABLE ERROR.
2166      ;      TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN
2167      ;      THE FUNCTION WAS INITIATED. RECOVERY PROCEDURE IS TO LOG THE
2168      ;      ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND.
2169      ;      2 WRITE-ERROR-RECOVERY ALGORITHMS CAN BE SELECTED:
2170      ;      THE FIRST ONE, VIA BADTSW SWITCH, DOES DETECT BAD SPOTS ON TAPE.
2171      ;      IT CALLS A WRITE RETRY SUBR UNTIL THE RECORD IS RECOVERED
2172      ;      OR 20 BAD SPOTS HAVE BEEN LOGGED. ON REACHING 20 BAD
2173      ;      SPOTS LOGGED, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
2174      ;      UNIT DROPPED.
2175      ;      THE SECOND ALGORITHM ISSUES THE TS05 WRITE RETRY COMMAND
2176      ;      UP TO 16 TIMES BEFORE DROPPING THE UNIT OR PROCEEDING
2177      ;      WITH THE NEXT RECORD ON RECOVERY.
2178      ;      INPUTS:
2179      ;      OUTPUTS:
2180      ;      REGISTERS:      R2,R4.
2181      ;      CALLS:          RTLE, EXCUTE, GOWAIT, DROPU, WRTY
2182
2183 013616 026727 167604 000002 TCC4::  CMP      CMDLG,#2      ;IF CMDLG EQ #2 AND8 BADTSW NE #0 THEN
2184 013624 001125      BNE      501620
2185 013626 105767 166356      TSTB     BADTSW
2186 013632 001522      BEQ      501620
2187 013634 105767 167631      TSTB     ERRREC      ;IF8 ERRREC EQ #0 AND8 ERCVER NE #0 THEN
2188 013640 001007      BNE      501630
2189 013642 105767 166341      TSTB     ERCVER

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 082

```

2190 013646 001404      BEQ      501631
2191 013650      ERRSOFT 9, RERM, STAERM ,
      013650 104457
      013652 000011      TRAP    C1ERSOFT
      013654 005017      .WORD   9
      013656 006120      .WORD   RERM
      .WORD   STALRM

2192
2193 013660
2194 013660 105767 166327      501631:  TSTB    IREC      ;IFB IREC EQ #0 THEN
2195 013664 001102      BNE      501641
2196 013666 105267 167577      INCB    ERRREC    ;RETRY FLAG FOR EXECUTE SUBR; DON'T UPDATE REC CN
2197 013672 105267 167566      INCB    WRTYER    ;REWRITE ERROR FLAG FOR WRTY SUBR
2198 013676 105767 167561      TSTB    WRTYFG    ;FIRST RETRY ON THIS RECORD; SUBSEQUENT
2199 013702 001072      BNE      501651
2200
      .RETRIES WITH TCC4 ERRORS BY-PASS THIS SECTION
2201 013704 016767 167510 001174      MOV     CMDWRD, WTYWRD ;SAVE WRITE COMMAND PACKET
2202 013712 016767 166412 001164      MOV     CMDPKT, WTYCMD
2203 013720 016767 166412 001162      MOV     CMDPKT, CP.CNT, WTYBRF
2204 013726 105267 167535      INCB    RWERR      ;LOG SUBR FLAG: COUNT WRT ERRORS
2205 013732 105267 167525      INCB    WRTYFG    ;RETRY IN PROGRESS FLAG
2206
2207 013736      501661:  REPEAT
2208 013736 005265 003316      INC      WRTYCT(R5) ;COUNT GLOBAL WRITE RETRIES
2209 013742 005067 167512      CLR      RETRYC    ;CLEAR # OF RETRIES PER RECORD
2210 013746 105067 167510      CLRB    RPTCNT    ;CLEAR # OF REPEATS
2211 013752 004767 000660      JSR     PC, WRTY ;CALL WRITE RETRY
2212 013756 105767 167502      TSTB    WRTYER    ;REPEAT RETRIES ON SAME RECORD
2213 013762 001404
2214 013764 027727 167522 000050      BEQ     501671
2215 013772 103761      CMP     #BTPT, #40.
2216 013774      BLO      501661
2217
      501671:
2218 013774 027727 167512 000050      CMP     #BTPT, #40. ;UNTIL RECOVERED OR 20 BAD SPOTS
2219 014002 103423      BLO      501701 ;WHEN 20 BAD SPOTS LOGGED
2220 014004      PRINTB  #BTMSG2 ;PRINT BAD TAPE OVERFLOW MSG
      014004 012746 015177      MOV     #BTMSG2, -(SP)
      014010 012746 000001      MOV     #1, -(SP)
      014014 010600      MOV     SP, R0
      014016 104414      TRAP    C1PNTB
      014020 062706 000004      ADD     #4, SP

2221 014024 004767 001266      JSR     PC, BORERS ;ERASE BAD RECORD
2222 014030 005365 003376      DEC     RECCNT(R5)
2223 014034 004767 003140      JSR     PC, DROPU ;DROP UNIT
2224 014040 005065 003376      CLR     RECCNT(R5)
2225 014044 012775 002350 002514      MOV     #RWCPK, #TSDB(R5) ;REWIND UNIT
2226
2227 014052      501701:
2228 014052 105067 167405      CLRB    WRTYFG    ;RETRY COMPLETE FLAG
2229 014056 105267 167447      INCB    MISCFG    ;DO NOT HALT ON THIS CMD FLG
2230 014062 016767 001020 167334      MOV     WTYWRD, PCMDWD ;RESTORE ORIGINAL WRT CMD AFTER RECOVERY
2231
2232 014070      501651:
2233
2234 014070 000402      BR      501711
2235 014072      501641:
2236 014072 105267 167372      INCB    UNREC      ;LET UNREC :B= UNREC * #1
2237

```

```

2238 014076          501714:
2239
2240 014076 000454          501624: BR      501724
2241 014100
2242 014100 004767 000404    JSR      PC,RTLE          ;CHECK FOR RETRY LIMIT EXCEEDED.
2243 014104 026727 167316 000002    CMP      CMDLG,#2          ;IF READ CMD THEN:
2244 014112 003411          BLE      501734
2245 014114 012702 000020    MOV      @RRECL,R2          ;R2=READ RETRY COUNT LIMIT / 2
2246 014120 006202          ASR      R2
2247 014122 026702 167332    CMP      RETRYC,R2          ;IF RETRY COUNT IS MORE THAN HALF LIMIT:
2248 014126 002403          BLT      501744
2249 014130 052767 020000 166172    BIS      @OPP.C,CMDPKT          ;SET OPPOSITE BIT FOR RETRY2.
2250
2251 014136          501744:
2252
2253 014136          501734:
2254 014136 005767 167316    TST      RETRYC          ;IF THIS IS THE ORIGINAL ERROR THEN:
2255 014142 001007          BNE      501754
2256 014144 105767 166037    TSTB     ERVER
2257 014150 001404          BEQ      501754
2258 014152          ERRSOFT 9,RERM,STAERM ;REPORT RECOVERABLE ERROR
2259          014152 104457          TRAP     C#ERSOFT
2260          014154 000011          .WORD    9
2261          014156 005017          .WORD    RERM
2262          014160 006120          .WORD    STAERM
2263
2264 014162          ;PROVIDED OPERATOR HAS ENABLED THE REPORT
2265 014162 005267 167272    501754: INC      RETRYC          ;UPDATE RETRY COUNT.
2266 014166 052767 001000 166134    BIS      @MOD.C1,CMDPKT ;SET RETRY BIT IN CMD PACKET.
2267 014174 105767 166013    TSTB     IREC          ;IF ERROR RECOVERY ENABLED:
2268 014200 001011          BNE      501764
2269 014202 105267 167263    INCB     ERRREC          ;SET ERROR RECOVERY FLAG.
2270 014206 012602          MOV      (SP)+,R2          ;POP 2 RTN ADRS FROM STACK.
2271 014210 012602          MOV      (SP)+,R2
2272 014212 004767 175636    JSR      PC,EXECUTE
2273 014216 000167 176142    JMP      GOWAIT          ;GO EXECUTE THE RETRY COMMAND.
2274          ;GO WAIT FOR INTERRUPT + CHECK STATUS.
2275          ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
2276
2277 014222 000402          BR      501774
2278 014224          501764: INCB     UNREC          ;SET UNRECOVERABLE ERROR FLAG.
2279 014224 105267 167240
2280
2281 014230          501774:
2282
2283 014230          501724:
2284 014230 000207          RTS PC          ;RETURN
2285
2286          ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
2287          ; TAPE POSITION HAS NOT CHANGED. RECOVERY PROCEDURE IS TO LOG THE
2288          ; ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
2289          ; INPUTS:
2290          ; OUTPUTS:
2291          ; REGISTERS: R2,R4.
2292          ; CALLS: RTLE, EXECUTE, GOWAIT, DROPV.
2293
2294 014232 004767 000252    TCC5: JSR      PC,RTLE          ;CHECK FOR RETRY LIMIT EXCEEDED
2295 014236 005767 167216    TST      RETRYC          ;IF THIS IS THE ORIGINAL ERROR THEN:
2296 014242 001004          BNE      502004

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 084

```

2291 014244 ERRSOF 10,RERM,STAERM ;REPORT RECOVERABLE ERROR.
      014244 104457
      014246 000012
      014250 005017
      014252 006120
2292 014254
2293 014254 005267 167200
2294 014260 105767 165727
2295 014264 001016
2296 014266 105267 167177
2297 014272 005265 003376
2298 014276 016577 003376 167102
2299 014304 012602
2300 014306 012602
2301 014310 004767 175540
2302 014314 000167 176044
2303
2304 014320 000402
2305 014322
2306 014322 105267 167142
2307
2308 014326
2309 014326 000207
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323 014330 036767 163454 166032
2324 014336 001016
2325
2326 014340 005767 167062
2327 014344 001404
2328 014346 105267 167115
2329 014352 105267 167112
2330
2331 014356
2332 014356
      014356 104455
      014360 000013
      014362 005041
      014364 006120
2333 014366 004767 002606
2334
2335 014372 000436
2336 014374
2337 014374 004767 000110
2338 014400 005767 167054
2339 014404 001004

      502001:
      INC RETRYC ;UPDATE RETRY COUNTER.
      TSTB IREC ;IF ERROR RECOVERY IS ENABLED:
      BNE 502011
      INCB ERRREC ;SET ERROR RECOVERY FLAG.
      INC RECCNT(R5) ;UPDATE REC COUNT
      MOV RECCNT(R5),B0DATAWT ;AND INSERT IT INTO WRT BFR
      MOV (SP)+,R2 ;POP 2 RTN ADRS FROM STACK.
      MOV (SP)+,R2
      JSR PC,EXCUTE ;GO RE-ISSUE THE COMMAND.
      JMP GOWAIT ;GO WAIT FOR INTERRUPT + CHECK STATUS.
                  ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
      BR 502021
      502011:
      INCB UNREC ;SET UNRECOVERABLE ERROR FLAG.
      502021:
      RTS PC ;RETURN.

      ;
      ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR.
      ; TAPE POSITION HAS BEEN LOST. THE ONLY VALID RECOVERY PROCEDURE
      ; IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR
      ; SEQUENCE NUMBERS. THIS DIAGNOSTIC WILL REWIND AND RETRY THE
      ; COMMAND ONLY IF DENSITY CHECK IS SET, OTHERWISE THE UNIT WILL BE
      ; DROPPED FROM THE TEST SEQUENCE.
      ;
      ; INPUTS:
      ; OUTPUTS:
      ; REGISTERS: R2, R4
      ; CALLS: RTLE, WSSR, EXCUTE, GOWAIT, DROPU

      TCC6::
      BIT X3.DCK,MSGPKT+MS.XS3;IF X3.DCK NOTSETIN MSGPKT+MS.XS3 THEN
      BNE 502031
                  ;IF THERE IS NO DENSITY CHECK THEN:
                  ;IF CMD IS A READ OR WRITE THEN:
      TST CMDLG
      BEQ 502041
      INCB RWERR ;SET RD/WR ERROR FLAG.
      INCB UNREC ;SET UNRECOVERABLE ERROR FLAG.
      502041:
      ERROF 11,URERM,STAERM ;REPORT UNRECOVERABLE ERROR.
                  TRAP C1ERDF
                  .WORD 11
                  .WORD URERM
                  .WORD STAERM
      JSR PC,DROPU ;REPORT ERROR + DROP UNIT.
                  ;ELSE-IF THERE IS DENSITY CHECK:
      BR 502051
      502031:
      JSR PC,RTLE ;CHECK FOR RETRY LIMIT EXCEEDED.
      TST RETRYC ;IF THIS IS THE ORIGINAL ERROR THEN:
      BNE 502061

```

```

2340 014406          ERRSOF 11,URERM,STAERM          ;REPORT DENSITY CHECK ERROR
          014406 104457                                TRAP      C#ERSOF 11
          014410 000013                                .WORD
          014412 005041                                .WORD      URERM
          014414 006120                                .WORD      STAERM

2341
2342 014416          50206: INC      RETRYC          ;UPDATE RETRY COUNT.
2343 014416 005267 167036      TSTB      IRE          ;IF ERROR RECOVERY IS ENABLED THEN:
2344 014422 105767 167073      BNE      50207:
2345 014426 001016          INCB      ERRREC          ;SET ERROR RECOVERY FLAG.
2346 014430 105267 167035      MOV      @RWCPK,@TSDB(R5) ;ISSUE A REWIND COMMAND.
2347 014434 012775 002350 002514 JSR      PC,WSSR          ;WAIT FOR SUBSYSTEM READY.
2348 014442 004767 176232      MOV      (SP)+,R2      ;POP 2 RTN ADRS FROM STACK.
2349 014446 012602          MOV      (SP)+,R2
2350 014450 012602          JSR      PC,EXCUTE
2351 014452 004767 175376      JMP      GOWAIT
2352 014456 000167 175702      BR      50210:
2353
2354 014462 000402          50207: INCB      UNREC          ;SET UNRECOVERABLE ERROR FLAG.
2355 014464
2356 014464 105267 167000      50210:
2357
2358 014470          50205: RTS      PC          ;RETURN
2359
2360 014470
2361 014470 000207
2362
2363      ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 7, FATAL SUBSYSTEM
2364      ; ERROR. THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING
2365      ; COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE.
2366      ; REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR
2367      ; ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR.
2368      ; INPUTS:
2369      ; OUTPUTS:
2370      ; REGISTERS:      R2, R4
2371      ; CALLS:
2372
2373 014472          TCC7:: ERRDF 12,FATSM,STAERM          ;REPORT FATAL SUBSYSTEM ERROR.
          014472 104455                                TRAP      C#ERDF 12
          014474 000014                                .WORD      FATSM
          014476 004642                                .WORD      STAERM
          014500 006120
2374 014502 004767 002472      JSR      PC,DROPU          ;DROP THE UNIT.
2375 014506 000207      RTS      PC          ;RETURN.
2376
2377
2378      ; SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED. PRINTS ERROR MESSAGE
2379      ; IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
2380      ; INPUTS:
2381      ; OUTPUTS:
2382      ; REGISTERS:      R2, R4.
2383      ; CALLS:      DROPU
2384
2385 014510 005767 166712      RTLE:: TST      CMDLG          ;IF CMD IS NOT A READ OR WRITE THEN:
2386 014514 001010          BNE      50211:
2387 014516          ERRDF 11,URERM,STAERM ;REPORT UNRECOVERABLE ERROR.
          014516 104455                                TRAP      C#ERDF

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 086

```

014520 000013
014522 005041
014524 006120
2388 014526 004767 002446 JSR PC,DROPU ;DROP THE UNIT.
2389 014532 012602 MOV (SP)+,R2 ;POP RTN ADRS FROM STACK.
2390 014534 000437 BR RTLRTN ;AND RETURN.
2391
2392 014536 502111: INCB RWERR ;SET READ/WRITE ERROR FLAG.
2393 014536 105267 166725 CMP CMDLG,#2 ;IF CMD IS A WRT OR WTM:
2394 014542 026727 166660 000002 BNE 502121
2395 014550 001016 CMP RETRYC,#WRECL ;IF RETRY COUNT HAS REACHED LIMIT:
2396 014552 026727 166702 000020 BNE 502131
2397 014560 001011 INCB UNREC ;SET UNRECOVERABLE FLAG
2398 014562 105267 166702 ERDF 14,RLEXM,STAERM ;REPORT RETRY LIMIT EXCEEDED.
2399 014566 104455 TRAP C#ERDF
014570 000016 .WORD 14
014572 004556 .WORD RLEXM
014574 006120 .WORD STAERM
2400 014576 004767 002376 JSR PC,DROPU ;DROP THE UNIT.
2401 014602 012602 MOV (SP)+,R2 ;POP 2 RTN ADRS FROM STACK.
2402 014604 502131: ;ELSE - CMD IS A READ:
2403
2404 014604 000413 BR 502141
2405 014606 502121: CMP RETRYC,#RRECL ;IF RETRY COUNT HAS REACHED LIMIT:
2406 014606 026727 166646 000020 BNE 502151
2407 014614 001007 INCB UNREC ;SET UNRECOVERABLE FLAG
2408 014616 105267 166646 ERHRD 14,RLEXM,STAERM ;REPORT RECOVERABLE ERROR.
2409 014622 104456 TRAP C#ERHRD
014622 000016 .WORD 14
014624 004556 .WORD RLEXM
014626 006120 .WORD STAERM
014630 0012602
2410 014632 012602 MOV (SP)+,R2 ;POP 2 RTN ADRS FROM STACK.
2411 014634 502151:
2412
2413 014634 502141: RTLRTN: RTS PC ;RETURN
2414 014634 000207
2415
2416 ; SUBR TO REWRITE A BAD, BUT RECOVERABLE WRITTEN RECORD.
2417 ; REWRITE RECORD ON SAME SPOT; REPEAT 4 TIMES.
2418 ; IF ALL 4 REPEATS GOOD, RECORD IS RECOVERED
2419 ; AND A RECOVERABLE WRITE ERROR IS LOGGED.
2420 ; IF ANY OF 4 REPEATS BAD, ERASE BAD RECORD, LOG SUSPECTED
2421 ; BAD SPOT, RETRY AGAIN. RETRY 4 TIMES, UP TO 4 REPEATS EACH.
2422 ; IF RECORD NOT GOOD AFTER 4 RETRIES, ERASE IT, EXIT WITH
2423 ; ERROR FLAG WRTYER SET, PRINTING RETRY FAILED.
2424 ; THIS ALL SCHEME IS REENTERED 20 TIMES MAX, IE 20 BAD
2425 ; SPOTS MAX ARE ALLOWED.
2426 ;
2427 ; INPUTS:
2428 ; OUTPUTS:
2429 ; REGISTERS: R3,R4
2430 ; CALLS: BORERS, REWRT
2431
2432 014636 WRTY:: ;BEGIN RETRY ;REPEAT
2433

```

```

2434 014636          502174:          ;BEGIN REPEAT          ;REPEAT
2435
2436
2437 014636          502214:          ;BACKSPACE/ERASE ONE RECORD
2438 014636 004767 000454          JSR      PC,BORERS          ;CLEAR WRITE RETRY ERROR
2439 014642 105067 166616          CLR      WRTYER          ;REWRITE RECORD ON SAME SPOT
2440 014646 004767 000620          JSR      PC,REWRT          ;COUNT REPEATS
2441 014652 105267 166604          INCB     RPTCNT          ;LIMIT: 4 REPEATS OR RECOVERED
2442 014656 126727 166600 000004  CMPB     RPTCNT,#4
2443 014664 001403          BEQ      502224
2444 014666 105767 166572          TSTB     WRTYER
2445 014672 001761          BEQ      502214
2446 014674
2447          502224:          ;END REPEAT
2448 014674          502204:          ;COUNT RETRIES
2449 014674 005267 166560          INC      RETRYC
2450 014700 105767 166560          TSTB     WRTYER
2451 014704 001001          BNE      502234
2452 014706 000457          BR       502164          ;EXIT RETRY LOOP IF RECOVERED
2453
2454 014710          502234:          ;IFB ERCVER NE #0 THEN
2455 014710 105767 165273          TSTB     ERCVER
2456 014714 001415          BEQ      502254
2457 014716          PRINTB  #BTMSG1,RETRYC,<B,RPTCNT>          ;PRINT SUSPECTED BAD SPOT
2458 014716 005046          CLR      -(SP)
2459 014720 156716 166536          BISB     RPTCNT,(SP)
2460 014724 016746 166530          MOV      RETRYC, -(SP)
2461 014730 012746 015112          MOV      #BTMSG1, -(SP)
2462 014734 012746 000003          MOV      #3, -(SP)
2463 014740 010600          MOV      SP,R0
2464 014742 104414          TRAP     C#PNTB
2465 014744 062706 000010          ADD      #10,SP
2466 014750          502254:          ;ON FIRST RETRY, LOGG BAD SPOT
2467 014750 026727 166504 000001  CMP      RETRYC,#1
2468 014756 001021          BNE      502264
2469 014760 016567 002616 166524  MOV      BTADDR(R5),BTPT
2470 014766 017704 166520          MOV      #BTPT,R4
2471 014772 062704 000002          ADD      #2,R4
2472 014776 010477 166510          MOV      R4,#BTPT
2473 015002 020427 000050          CMP      R4,#40
2474 015006 101005          BHI      502274          ;IF R4 LOS #40. THEN
2475 015010 016703 166476          MOV      BTPT,R3
2476 015014 060304          ADD      R3,R4
2477 015016 016514 003376          MOV      RECCNT(R5),(R4)          ;STORE FIRST 20 BAD SPOTS
2478
2479 015022          502274:          ;LET R4 := R4 + R3
2480 015022          502264:          ;LET (R4) := RECCNT(R5)
2481 015026 105067 166435          INCB     ERSFLG          ;ERASE FLAG TO ERASE BAD RECORD
2482 015032 105067 166424          CLR      RWERR          ;CANCELL "LOG" ERROR FLAG ON FAILING RET
2483          CLR      RPTCNT          ;CLEAR REPEAT COUNT FOR NEXT RETRY
2484 015036          502244:          ;LIMIT: 4 RETRIES
2485 015036 026727 166416 000004  CMP      RETRYC,#4
2486 015044 001274          BNE      502174
2487
2488 015046          502164:          ;END RETRY

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 088

```

2483 015046 105767 166412          TSTB  WRTYER  ,IFB WRTYER NE #0 THEN
2484 015052 001413          BEQ    502301
2485 015054 105767 165127          TSTB  ERCVER  ,IFB ERCVER NE #0 THEN
2486 015060 001410          BEQ    502311
2487 015062          PRINTB  #BTMSG3          ,PRINT RETRY FAILED
      015062 012746 015247          MOV    #BTMSG3,-(SP)
      015066 012746 000001          MOV    #1,-(SP)
      015072 010600          MOV    SP,R0
      015074 104414          TRAP   C#PNTB
      015076 062706 000004          ADD    #4,SP

2488
2489 015102          502311:
2490
2491 015102          502301:
2492 015102 000207          RTS    PC
2493
2494 015104 000000          WTYCMD: .WORD 0          ,STORAGE FOR WRITE CMD WHILE RETRYING
2495 015106 000000          WTYWRD: .WORD 0          ,STORAGE FOR WRITE CMD WORD WHILE RETRYING
2496 015110 000000          WTYBRF: .WORD 0          ,STORAGE FOR WRITE BPCR WHILE RETRYING
2497
2498 015112 045 101 123 BTMSG1: .ASCIZ /#ASUSPECT BAD SPOT AFTER #D1#A RETRY, #D1#A REPEAT#N/
      015115 125 123 120
      015120 105 103 124
      015123 040 102 101
      015126 104 040 123
      015131 120 117 124
      015134 040 101 106
      015137 124 105 122
      015142 040 045 104
      015145 061 045 101
      015150 040 122 105
      015153 124 122 131
      015156 054 040 045
      015161 104 061 045
      015164 101 040 122
      015167 105 120 105
      015172 101 124 045
      015175 116 000
2499 015177 045 116 045 BTMSG2: .ASCIZ /#N#ABAD TAPE OVERFLOW: CHANGE TAPE!#N#N/
      015202 101 102 101
      015205 104 040 124
      015210 101 120 105
      015213 040 117 126
      015216 105 122 106
      015221 114 117 127
      015224 072 040 103
      015227 110 101 116
      015232 107 105 040
      015235 124 101 120
      015240 105 041 045
      015243 116 045 116
      015246 000
2500 015247 045 101 122 BTMSG3: .ASCIZ /#ARETRY FAILED ON BAD SPOT...ERASED!#N/
      015252 105 124 122
      015255 131 040 106
      015260 101 111 114
      015263 105 104 040

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 089

015266	117	116	040	
015271	102	101	104	
015274	040	123	120	
015277	117	124	056	
015302	056	056	105	
015305	122	101	123	
015310	105	104	041	
015313	045	116	000	

2501 .EVEN
2502
2503
2504
2505
2506
2507
2508
2509

```

SUBR TO BACKSPACE ONE RECORD
IF THE ERASE FLAG IS SET, THEN ERASE THAT RECORD
INPUTS:      ERSFLG 1 = DO ERASE
OUTPUTS:
REGISTERS:
CALLS:      EXECUTE, GOWAIT, CKHAE

```

2510 015316 016767 166076 166100 BORERS: MOV CMDWRD,PCMDWD ;SET COMMAND TO SPACE REV
2511 015324 012767 104410 166066 MOV #SRR,CMDWRD ;LET CMDWRD := #SRR
2512 015332 016767 166062 164770 MOV CMDWRD,CMDPKT ;LET CMDPKT := CMDWRD CLR.BY #BRF.C
2513 015340 042767 004000 164762 BIC #BRF.C,CMDPKT
2514 015346 016767 164756 166046 MOV CMDPKT,CMSAV ;LET CMSAV := CMDPKT
2515 015354 012767 000001 164750 MOV #1,CMDPKT+CP.ADL ;LET CMDPKT+CP.ADL := #1
2516 015362 005067 166040 CLR CMDLG ;LET CMDLG := #0
2517 015366 004767 173472 JSR PC,CMDAC
2518 015372 004767 174456 JSR PC,EXECUTE
2519 015376 004767 174762 JSR PC,GOWAIT
2520 015402 004767 002072 JSR PC,CKHAE
2521 015406 105767 166113 TSTB ERSFLG ;WHEN ERASE FLAG IS SET, DO ERASE
2522 015412 001426 BEQ 502324
2523 015414 016767 166000 166002 MOV CMDWRD,PCMDWD ;LET PCMDWD := CMDWRD
2524 015422 012767 100411 165770 MOV #ERS,CMDWRD ;LET CMDWRD := #ERS
2525 015430 016767 165764 164672 MOV CMDWRD,CMDPKT ;LET CMDPKT := CMDWRD
2526 015436 016767 164666 165756 MOV CMDPKT,CMSAV ;LET CMSAV := CMDPKT
2527 015444 004767 173414 JSR PC,CMDAC
2528 015450 004767 174400 JSR PC,EXECUTE
2529 015454 004767 174704 JSR PC,GOWAIT
2530 015460 004767 002014 JSR PC,CKHAE
2531 015464 105067 166035 CLRB ERSFLG ;LET ERSFLG := #0
2532
2533 015470 502324:
2534 015470 000207 RTS PC
2535
2536
2537
2538 015472 016767 165722 165724 REWRT: MOV CMDWRD,PCMDWD ;RESTORE WRITE COMMAND PACKET
2539 015500 016767 177402 165712 MOV WTYWRD,CMDWRD ;LET CMDWRD := WTYWRD
2540 015506 016767 177372 164614 MOV WTYCMD,CMDPKT ;LET CMDPKT := WTYCMD
2541 015514 016767 164610 165700 MOV CMDPKT,CMSAV ;LET CMSAV := CMDPKT
2542 015522 016767 165660 164602 MOV DATAWT,CMDPKT+CP.ADL ;LET CMDPKT+CP.ADL := DATAWT
2543 015530 016767 177354 164600 MOV WTYBRF,CMDPKT+CP.CNT ;LET CMDPKT+CP.CNT := WTYBRF
2544 015536 012767 000002 165662 MOV #2,CMDLG ;LET CMDLG := #2
2545 015544 004767 173314 JSR PC,CMDAC
2546 015550 004767 174300 JSR PC,EXECUTE ;RE-WRITE RECORD
2547 015554 004767 174604 JSR PC,GOWAIT
2548 015560 004767 001714 JSR PC,CKHAE
2549 015564 000207 RTS PC

```

2550
2551      ; SUBROUTINE TO LOG BYTES READ/WITTEN.
2552      ; ALSO UPDATES READ/WRITE ERROR COUNTERS.
2553      ; INPUTS:
2554      ; OUTPUTS:
2555      ; REGISTERS:      R2, R3, R4.
2556      ; CALLS:
2557
2558 015566 105767 165674      LOG::  TSTB      ERLOG      ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED THEN:
2559 015572 001126          BNE      50233$
2560 015574 105267 165666      INCB      ERLOG      ;SET LOG DONE FLAG.
2561 015600 016704 165622      MOV      CMDLG,R4      ;GET CURRENT CMD LOGGING CODE.
2562 015604 005704          TST      R4      ;IF THERE IS A CODE THEN:
2563 015606 001520          BEQ      50234$
2564 015610 162704 000002      SUB      #2,R4      ;ADJUST THE CODE FOR TABLE INDEX.
2565 015614 010502          MOV      R5,R2      ;R2 = ADR OF BYTE COUNT LSW.
2566 015616 066402 016052      ADD      BINC(R4),R2
2567 015622 062702 002626      ADD      #CNTBGN,R2
2568 015626 066712 165564      ADD      BRFCNT,(R2)      ;ADD BRFCNT TO LSW.
2569 015632 026767 164522 165556      CMP      MSGPKT+MS.RFC,BRFCNT      ;IF THE RFC IS LOWER OR THE SAME AS BRFC THEN
2570 015640 101002          BHI      50235$
2571 015642 166712 164512      SUB      MSGPKT+MS.RFC,(R2)      ;SUBTRACT RFC FROM EXPECTED BRFC.
2572
2573 015646          50235$:
2574 015646 010203          MOV      R2,R3      ;R3 = ADR OF 2ND WORD.
2575 015650 062703 000010      ADD      #10,R3
2576
2577 015654          50236$:      ;WHILE (R2) GT #999. DO
2578 015654 021227 001747      CMP      (R2),#999.
2579 015660 003404          BLE      50237$
2580 015662 162712 001750      SUB      #1000..(R2)      ;UPDATE BYTE COUNT
2581 015666 005213          INC      (R3)      ;LET (R3) := (R3) + #1      ;2ND WORD.
2582
2583 015670 000771          BR      50236$
2584 015672
2585 015672 010302          50237$:      MOV      R3,R2      ;LET R2 := R3 + #10      ;R2 = ADR OF 3RD WORD.
2586 015674 062702 000010      ADD      #10,R2
2587 015700          50240$:      ;WHILE (R3) GT #999. DO
2588 015700 021327 001747      CMP      (R3),#999.
2589 015704 003404          BLE      50241$
2590 015706 162713 001750      SUB      #1000..(R3)      ;UPDATE BYTE COUNT
2591 015712 005212          INC      (R2)      ;LET (R2) := (R2) + #1      ;3RD WORD.
2592
2593 015714 000771          BR      50240$
2594 015716
2595 015716 010203          50241$:      MOV      R2,R3      ;LET R3 := R2 + #10      ;R3 = ADR OF 4TH WORD.
2596 015720 062703 000010      ADD      #10,R3
2597 015724          50242$:      ;WHILE (R2) GT #999. DO
2598 015724 021227 001747      CMP      (R2),#999.
2599 015730 003404          BLE      50243$
2600 015732 162712 001750      SUB      #1000..(R2)      ;UPDATE BYTE COUNT
2601 015736 005213          INC      (R3)      ;LET (R3) := (R3) + #1      ;4TH WORD.
2602
2603 015740 000771          BR      50242$
2604 015742
2605 015742 105767 165521      50243$:      TSTB      RWERR      ;IF R/W ERROR, UPDATE ERROR COUNT.
2606 015746 001440          BEQ      50244$

```

```

2607 015750 010502      MOV      R5,R2              ;R2 = ADR OF COUNTER.
2608 015752 066402 016060  ADD      EINC(R4),R2
2609 015756 062702 002766  ADD      @WRREC,R2
2610 015762 105767 165502  TSTB    UNREC          ;IS THE ERROR UNRECOVERABLE?
2611 015766 001404      BEQ      50245#
2612 015770 062702 000010  ADD      @10,R2        ;YES, POINT TO NEXT COUNTER.
2613 015774 005212      INC      (R2)          ;UPDATE THE ERROR COUNTER
2614                                ;ELSE - IF ERROR IS RECOVERABLE:
2615 015776 000424      BR       50246#
2616 016000      50245# : INC      (R2)          ;UPDATE THE ERROR COUNTER
2617 016000 005212      TSTB    IREC          ;IF ERROR RECOVERY IS ENABLED:
2618 016002 105767 164205  BNE      50247#
2619 016006 001020      TSTB    DROPED        ;IF UNIT HAS NOT BEEN DROPPED:
2620 016010 105767 165506  BNE      50250#
2621 016014 001015      TSTB    ERCVER
2622 016016 105767 164165  BEQ      50250#
2623 016022 001412      PRINTB  @NURTY1,RETRYC ;PRINT # OF RETRIES TO RECOVER
2624 016024      MOV      RETRYC,-(SP)
      016024 016746 165430  MOV      @NURTY1,-(SP)
      016030 012746 005422  MOV      @2,-(SP)
      016034 012746 000002  MOV      SP,R0
      016040 010600      MOV      SP,R0
      016042 104414      TRAP     C#PNTB
      016044 062706 000006  ADD      @6,SP
2625                                ;PROVIDED PRINT HAS BEEN ENABLED
2626 016050      50250# :
2627      50247# :
2628 016050
2629
2630
2631 016050      50246# :
2632
2633 016050      50244# :
2634
2635 016050      50234# :
2636
2637 016050      50233# :
2638 016050 000207      RTS      PC
2639
2640      ; INDEXES TO BYTE COUNTERS.
2641 016052 000000      BINC:    0
2642 016054 000040      40
2643 016056 000100      100
2644      ; INDEXES TO READ/WRITE ERROR COUNTERS.
2645 016060 000000      EINC:    0
2646 016062 000020      20
2647 016064 000040      40
2648
2649      ; IF A WRITE/VERIFY COMMAND IS ISSUED, CONTROL IS THEN
2650      ; TRANSFERRED TO THIS SUBROUTINE TO READ REVERSE, CHECK DATA,
2651      ; READ FORWARD, CHECK DATA, THEN CONTINUE TO NEXT COMMAND.
2652      ; INPUTS:
2653      ; OUTPUTS:
2654      ; REGISTERS:
2655      ; CALLS:      VFEXC.
2656
2657 016066 105767 165424  VFYDAT:;TSTB    VFYFLG          ;IF DATA IS TO BE VERIFIED:

```

```

2658 016072 001426 BEQ 502511
2659 016074 016767 165320 165322 MOV CMDWRD,PCMDWD ;SAVE THE PREVIOUS COMMAND WORD.
2660 016102 012767 104401 165310 MOV @RDR,CMDWRD ;COMMAND IS READ REV.
2661 016110 012767 000004 165310 MOV #4,CMDLG ;SET UP CMD LOGGING INDEX.
2662 016116 004767 000030 JSR PC,VFEXC ;GO READ ALL THE RECORDS REV.
2663 016122 016767 165272 165274 MOV CMDWRD,PCMDWD ;SAVE THE PREVIOUS COMMAND WORD.
2664 016130 012767 104001 165262 MOV @RDF,CMDWRD ;COMMAND IS READ FWD.
2665 016136 012767 000006 165262 MOV #6,CMDLG ;SET UP CMD LOGGING INDEX.
2666 016144 004767 000002 JSR PC,VFEXC ;GO READ ALL RECORDS FWD.
2667
2668 016150 502511:
2669 016150 000207 RTS PC ;RETURN.
2670
2671
2672
2673
2674 ; SUBROUTINE TO EXECUTE THE READ AND VERIFY, FORWARD OR REVERSE.
2675 ; INPUTS:
2676 ; OUTPUTS:
2677 ; REGISTERS: R2
2678 ; CALLS: CMDAC, FIRSTU, VFISU, NEXTU, CKHAE.
2679
2680 016152 016767 165242 164150 VFEXC:: MOV CMDWRD,CMDPKT ;COMMAND PACKET - READ REV OR FWD.
2681 016160 042767 004000 164142 BIC @BRF.C,CMDPKT
2682 016166 105767 165326 TSTB SWBFLG ;IF BYTES ARE TO BE SWAPPED:
2683 016172 001403 BEQ 502521
2684 016174 052767 010000 164126 BIS @SWB.C,CMDPKT ;SET SWAB BIT IN CMD PACKET.
2685
2686 016202 502521:
2687 016202 016767 164122 165212 MOV CMDPKT,CMDSAV ;SAVE COMMAND PACKET 1ST WORD.
2688 016210 016767 165174 164114 MOV DATARD,CMDPKT+CP.ADL ;SAVE BUFFER START ADDRESS.
2689 016216 005067 165170 CLR NCNT ;CLEAR NUMBER OF OPERATIONS.
2690
2691 016222 502531: ;WHILE NCNT LT NCNT1 DO ;WHILE THERE ARE RECORDS REMAINING:
2692 016222 026767 165164 165164 CMP NCNT,NCNT1
2693 016230 002062 BGE 502541
2694 016232 004767 172626 JSR PC,CMDAC ;STORE CMD ASCII IN ERROR MSG.
2695 016236 004767 000640 JSR PC,FIRSTU ;SET UP FOR FIRST UNIT.
2696 016242 502551: ;WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE DEVICES REMAINING:
2697 016242 026527 002604 177777 CMP DEVTBL(R5),#END
2698 016250 001442 BEQ 502561
2699 016252 032767 000400 165140 BIT @MOD.CO,CMDWRD ;IF CMD IS REVERSE THEN:
2700 016260 001421 BEQ 502571
2701 016262 032765 000002 003502 BIT @X0.BOT,EOTFLG(R5) ;IF NOT AT BOT
2702 016270 001014 BNE 502601 ;BUT IF AT EOT
2703 016272 032765 000001 003502 BIT @X0.EOT,EOTFLG(R5)
2704 016300 001406 BEQ 502611
2705 016302 105767 165216 TSTB ALLEOT ;AND ALL OTHERS AT EOT
2706 016306 001402 BEQ 502621
2707 016310 004767 000064 JSR PC,VFISU ;THEN READ VERIFY
2708 ;IF NOT ALL AT EOT, FREEZE UNIT(S)
2709 016314 502621: ;IF NOT AT BOT AND
2710
2711 016314 000402 BR 502631
2712 016316 502611:
2713 016316 004767 000056 JSR PC,VFISU ;NOT AT EOT, READ VFY
2714

```

```

2715 016322      502630:
2716
2717 016322      502600:
2718 016322      000412      BR      502640      ;ELSE IF CMD IS NOT REVERSE:
2719 016324
2720 016324      032765      000001      003502      502570:
2721 016332      001404      BIT      @X0.EOT,EOTFLG(R5)
2722 016334      032767      000001      165056      BEQ      502650
2723 016342      001002      BIT      @CMD.CO,CMDWRD
2724 016344      502650:      BNE      502660
2725
2726 016344      004767      000030      JSR      PC,VFISU      ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
2727
2728 016350      502660:      ;ISSUE CMD, CHECK STATUS AND DATA.
2729
2730 016350      502640:
2731 016350      004767      000574      JSR      PC,NEXTU      ;GO FIND THE NEXT UNIT.
2732
2733 016354      000732      BR      502550
2734 016356      502560:
2735 016356      004767      001116      JSR      PC,CKHAE      ;CHECK FOR HALT AFTER EACH CMD.
2736 016362      005267      165024      INC      NCNT      ;UPDATE THE RECORD COUNT.
2737 016366      016767      165026      165030      MOV      CMDWRD,PCMDWD      ;SAVE PREVIOUS COMMAND WORD.
2738
2739 016374      000712      BR      502530
2740 016376      502540:
2741 016376      000207      RTS      PC      ;RETURN.
2742
2743      ;      SUBROUTINE TO ISSUE COMMAND, AWAIT INTERRUPT,
2744      ;      CHECK STATUS, CHECK DATA.
2745      ;      INPUTS:
2746      ;      OUTPUTS:
2747      ;      REGISTERS:      R2
2748      ;      CALLS:      EXECUTE, GOWAIT, CKDATA.
2749
2750 016400      016702      165004      VFISU:: MOV      DATARD,R2      ;INIT READ BUFFER POINTER.
2751 016404      062702      000010      ADD      @B.,R2
2752 016410      502670:      ;WHILE R2 NE DATARD DO      ;UNTIL 8 BYTES HAVE BEEN SET.
2753 016410      020267      164774      CMP      R2,DATARD
2754 016414      001403      BEQ      502700
2755 016416      012742      177777      MOV      @-1,-(R2)      ;INIT READ BUFFER.
2756
2757 016422      000772      BR      502670
2758 016424      502700:
2759 016424      004767      173424      JSR      PC,EXECUTE      ;GO EXECUTE THE COMMAND.
2760 016430      105767      165066      TSTB     DROPED      ;IF UNIT HAS NOT BEEN DROPPED THEN:
2761 016434      001002      BNE      502710
2762 016436      004767      173722      JSR      PC,GOWAIT      ;GO WAIT FOR DONE BIT.
2763
2764 016442      502710:
2765 016442      105767      165054      TSTB     DROPED      ;IF UNIT HAS NOT BEEN DROPPED THEN:
2766 016446      001006      BNE      502720
2767 016450      032765      000002      003502      BIT      @X0.BOT,EOTFLG(R5)      ;WHEN NOT REVERSED INTO BOT, THEN
2768 016456      001002      BNE      502730
2769 016460      004767      000002      JSR      PC,CKDATA      ;GO VERIFY DATA.
2770
2771 016464      502730:

```

```

2772
2773 016464      502720: RTS      PC
2774 016464      000207
2775
2776
2777      ; SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
2778      ; AND PRINT ERROR MESSAGE ON MISCOMPARE.
2779      ; INPUTS:
2780      ; OUTPUTS:
2781      ; REGISTERS:      R2, R3, R4.
2782      ; CALLS:          GCMDA
2783
2784 016466      016703      164724      CKDATA: MOV      BRFCNT,R3          ;COMPUTE REC LENGTH READ
2785 016472      166703      163662      SUB      MSGPKT+MS.RFC,R3
2786 016476      005703      TST      R3          ;WHEN NO DATA RECEIVED
2787 016500      001015      BNE      502740
2788 016502      ERRHRD      17,WTVERM,DTAERM          ;PRINT ERROR AND EXIT
2789      016502      104456      TRAP      C1ERHRD
2790      016504      000021      .WORD      17
2791      016506      004430      .WORD      WTVERM
2792      016510      005752      .WORD      DTAERM
2793      016512      PRINTB      #DTAER4          ;COMPARE ROUTINE
2794      016512      012746      005337      MOV      #DTAER4,-(SP)
2795      016516      012746      000001      MOV      #1,-(SP)
2796      016522      010600      MOV      SP,R0
2797      016524      104414      TRAP      C1PNTB
2798      016526      062706      000004      ADD      #4,SP
2799 016532      000560      BR      502750
2800 016534      502740: CMP      R3,BRFCNT          ;WHEN REC READ IS LONGER
2801 016534      020367      164656      BLOS     502760
2802 016540      101417      ERRHRD      17,WTVERM,DTAERM          ;THAN EXPECTED. PRINT
2803      016542      104456      TRAP      C1ERHRD
2804      016544      000021      .WORD      17
2805      016546      004430      .WORD      WTVERM
2806      016550      005752      .WORD      DTAERM
2807 016552      PRINTB      #DTAERS,CMDPKT+CP.CNT          ;AN ERROR MESSAGE
2808      016552      016746      163560      MOV      CMDPKT+CP.CNT,-(SP)
2809      016556      012746      005360      MOV      #DTAERS,-(SP)
2810      016562      012746      000002      MOV      #2,-(SP)
2811      016566      010600      MOV      SP,R0
2812      016570      104414      TRAP      C1PNTB
2813      016572      062706      000006      ADD      #6,SP
2814 016576      000536      BR      502770          ;AND EXIT ROUTINE
2815 016600      502760: MOV      R3,CKDCNT          ;SAVE VERIFICATION LENGTH - 1.
2816 016600      010367      000272      DEC      CKDCNT
2817 016604      005367      000266      CLR      CKDFF
2818 016610      005067      000264      CLR      R2
2819 016614      005002      MOV      DATAW,R3
2820 016616      016703      164564      MOV      DATARD,R4
2821 016622      016704      164562      TSTC     T1SWB
2822 016626      105767      164671      BEQ      503000
2823 016632      001401      SWAB      (R3)
2824 016634      000313      ;SWAP FIRST WORD OF WRT BFR
2825 016636      503000: ;WHICH CONTAINS THE RECORD COUNT

```

```

2810                                ;REPEAT
2811 016636 503014:                ;REPEAT UNTIL ALL DATA IS COMPARED:
2812 016636 020267 000234          ;IF THIS IS THE LAST BYTE THEN:
2813 016642 001011                ;IF BYTE SWAPPING IS ENABLED THEN:
2814 016644 105767 164650          ;IF RECORD LENGTH IS ODD THEN:
2815 016650 001406                ;LAST BYTE WILL BE IN
2816 016652 032767 000001 000216  ;THE UPPER BYTE.
2817 016660 001002                ;
2818 016662 105723                ;
2819 016664 105724                ;
2820                                ;
2821 016666 503044:                ;
2822                                ;
2823 016666 503034:                ;
2824                                ;
2825 016666 503024:                ;
2826 016666 121314                ;ARE THEY EQUAL.
2827 016670 001452                ;BR IF SO.
2828 016672 005767 000202          ;1 ST TIME THRU?
2829 016676 001010                ;BR IF NOT.
2830 016700 005265 003346          ;INC THE VERIFY ERROR COUNTER.
2831 016704 005265 003356          ;INC THE HARD ERROR COUNT.
2832 016710                ;REPORT WRITE/VERIFY ERROR.
      016710 104456                TRAP C1ERHRD
      016712 000021                .WORD 17
      016714 004430                .WORD WTVRM
      016716 005752                .WORD DTAERM
2833 016720 005267 000154          24: INC CKDFF,LET CKDFF := CKDFF + #1 ;INCREMENT # OF BYTES IN ERROR.
2834 016724 111467 164506          MOV (R4),TIME1 ;SAVE WAS DATA FOR TYP0UT.
2835 016730 042767 177400 164500  BIC #177400,TIME1 ;CLEAR GARBAGE.
2836 016736 111367 164476          MOV (R3),TIME2 ;SAVE SHOULD BE DATA FOR TYP0UT.
2837 016742 042767 177400 164470  BIC #177400,TIME2 ;CLEAR GARBAGE.
2838 016750 026727 000124 000013  CMP CKDFF,#11. ;IF ERROR BYTE COUNT IS LESS THAN 11:
2839 016756 002017                BGE 503054
2840 016760                PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>;PRINT ACTUAL & EXPECTED DATA
      016760 005046                CLR -(SP)
      016762 156716 164452          BISB TIME2,(SP)
      016766 005046                CLR -(SP)
      016770 156716 164442          BISB TIME1,(SP)
      016774 010246                MOV R2,-(SP)
      016776 012746 005226          MOV #DTAER2,-(SP)
      017002 012746 000004          MOV #4,-(SP)
      017006 010600                MOV SP,R0
      017010 104415                TRAP C1PNTX
      017012 062706 000012          ADD #12,SP
2841 017016 503054:                ;
2842                                ;
2843 017016 105723                ;UPDATE WRITE BUFFER ADDRESS.
2844 017020 105724                ;UPDATE READ BUFFER ADDRESS.
2845 017022 105722                ;UPDATE BYTE COUNTER.
2846 017024 020267 000046          ;END OF DATA COMPARE REPEAT LOOP.
2847 017030 003702                ;
2848 017032 005267 000040          ;CKDCNT EQUALS RECORD LENGTH.
2849 017036 005767 000036          ;IF COMPARE ERROR HAS OCCURED THEN:
2850 017042 001414                ;
2851 017044 016746 000026          ;PRINT # OF BYTES IN ERROR.
      017044                MOV CKDCNT,-(SP)

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 096

```

017050 016746 000024
017054 012746 005275
017060 012746 000003
017064 010600
017066 104414
017070 062706 000010
2852
2853 017074 503064:
2854
2855 017074 502774:
2856
2857 017074 502754:
2858 017074 000207 RTS PC ; OTHERWISE, RETURN.
2859
2860 017076 000000 CKDCNT: .WORD 0 ; # OF BYTES TO BE VERIFIED -1.
2861 017100 000000 CKDFF: .WORD 0 ; # OF BYTES IN ERROR COUNTER.
2862
2863 ; SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
2864 ; INPUTS:
2865 ; OUTPUTS:
2866 ; REGISTERS:
2867 ; CALLS:
2868
2869 017102 105067 164414 FIRSTU:: CLRB DROPED ; CLR UNIT DROPPED FLAG
2870 017106 005005 CLR R5 ; CLR DEVICE POINTER.
2871 017110 026527 002604 177774 503074: CMP DEVTBL(R5),#NINUSE ; WHILE DEVICES ARE NOT IN USE:
2872 017116 001003 BNE 503104
2873 017120 062705 000002 ADD #2,R5 ; LET R5 := R5 + #2 ; POINT TO NEXT DEVICE.
2874 017124 000771 BR 503074
2875 017126 503104:
2876 017126 026527 002604 177777 CMP DEVTBL(R5),#END ; IF ALL UNITS HAVE BEEN DROPPED THEN:
2877 017134 001001 BNE 503114 ; DO CLEAN CODE AND TERMINATE PASS.
2878 017136 104444 DOCLN TRAP C#DCLN
2879
2880 017140 503114:
2881 017140 016567 002604 162726 MOV DEVTBL(R5),L#LUN ; SET UNIT # IN "HEADER" FOR ERROR REPORT
2882 017146 000207 RTS PC ; RETURN WITH 1ST DEVICE IN R5.
2883
2884 ; SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
2885 ; INPUTS:
2886 ; OUTPUTS:
2887 ; REGISTERS:
2888 ; CALLS:
2889
2890
2891 017150 105067 164346 NEXTU:: CLRB DROPED ; CLR UNIT DROPPED FLAG
2892 017154 503124: ; REPEAT UNTIL THE NEXT DEVICE IS FOUND.
2893 017154 062705 000002 ADD #2,R5 ; UPDATE DEVICE TABLE POINTER.
2894 017160 026527 002604 177774 CMP DEVTBL(R5),#NINUSE ; UNTIL DEVTBL(R5) NE #NINUSE
2895 017166 001772 BEQ 503124
2896 017170 016567 002604 162676 MOV DEVTBL(R5),L#LUN ; SET UNIT # IN "HEADER" FOR ERROR REPORT
2897 017176 000207 RTS PC ; RETURN.
2898
2899
2900
2901 ; SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.

```

```

2902      ; INPUTS:
2903      ; OUTPUTS:
2904      ; REGISTERS:
2905      ; CALLS:      MOVMSG, PRXST, LOG
2906
2907 017200 005265 003366      DROPU:: INC      FTLCNT(R5)      ;INCREMENT THE FATAL ERROR COUNT.
2908 017204 016704 163160      MOV      MSGPKT+MS.XS3,R4      ;GET UDIAG ERROR CODE FROM XSTAT3.
2909 017210 042704 000377      BIC      #377,R4
2910 017214 016503 002544      MOV      MSGPKA(R5),R3      ;ADR OF THIS UNIT'S MSG PACKET.
2911 017220 005002      CLR      R2      ;LET R2 := #0      ;CLR COUNTER.
2912 017222      50313: WHILE R2 NE #MSGCNT DO      ;WHILE THERE ARE MORE LOCATIONS:
2913 017222 020227 000020      CMP      R2,#MSGCNT
2914 017226 001405      BEQ      50314:
2915 017230 012723 177777      MOV      #-1,(R3)+      ;INIT THE MSG PACKET WITH ALL 1'S
2916 017234 062702 000002      ADD      #2,R2      ;LET R2 := R2 + #2      ;UPDATE COUNTER.
2917
2918 017240 000770      BR      50313:
2919 017242      50314: MOV      #GSCPK,#TSD8(R5)      ;INITIATE A GET STATUS COMMAND.
2920 017242 012775 002340 002514      JSR      PC,WSSR      ;WAIT A WHILE FOR SSR=1
2921 017250 004767 173424      JSR      PC,MOVMSG      ;MOVE MSG PACKET TO COMMON AREA.
2922 017254 004767 173454      CMP      R4,#X3.RNY      ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
2923 017260 020427 157400      BNE      50315:
2924 017264 001005      ERDF      16,RNYM,STAERM      ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
2925 017266      TRAP      C#ERDF
2926 017266 104455      .WORD      16
2927 017270 000020      .WORD      RNYM
2928 017272 004753      .WORD      STAERM
2929 017274 006120
2926 017276 000402      BR      50316:      ;ELSE-IF NOT A RUNAWAY:
2927 017300      50315: JSR      PC,PRXST      ;PRINT EXTENDED STATUS REGISTERS.
2928 017300 004767 000106
2929 017304      50316: TSTB      RECLOG      ;IF THE RECORD HAS BEEN LOGGED THEN:
2930 017304 105767 164155      BEQ      50317:
2931 017310 001404      INCB      DROPE      ;SET UNIT DROPPED FLAG.
2932 017312 105267 164204      JSR      PC,LOG      ;LOG DATA BYTES + RD/WR ERRORS.
2933 017316 004767 176244
2934 017322      50317: DORPT      ;PRINT PERFORMANCE REPORT
2935 017322 104424      TRAP      C#DRPT
2936 017324 005765 003326      DROPUA: TST      PASCNT(R5)      ;IF PASCNT(R5) NE #0 THEN
2937 017330 001402      BEQ      50320:
2938 017332 005365 003326      DEC      PASCNT(R5)      ;LET PASCNT(R5) := PASCNT(R5) - #1
2939 017336      50320: MOV      TSNP,DROPN      ;SAVE # OF UNIT TO BE DROPPED.
2940 017336 016767 164172 000044      MOV      TSNP,RO      ;RO=LOGICAL DEVICE NUMBER
2941 017344 016700 164164      DODU      RO      ;DROP THE UNIT
2942 017350      TRAP      C#DODU
2943 017350 104451      ;EXEC BGNDU-ENDDU CODE IF IDU = 0
2944 017352 026527 002604 177774      CMP      DEVTBL(R5),#NINUSE      ;IF UNIT NOT DROPPED
2945 017360 001410      BEQ      50321:
2946 017362 105767 162625      TSTB      IREC
2947 017366 001005      BNE      50322:
2948
2949
2950
2951
2952

```

```

2953 017370 000240      NOP
2954 017372 000240      NOP
2955 017374 000240      NOP
2956 017376 105267 164124 INCB   STAFLG      ;SET START FLAG TO ENABLE REWIND.
2957
2958 017402      503221:
2959
2960 017402      503211:
2961 017402 105267 164114 DRORTN: INCB   DROPE      ;SET UNIT DROPPED FLAG.
2962 017406 000207      RTS    PC      ;RETURN.
2963
2964 017410 000000      DROPN: .WORD 0      ;# OF UNIT TO BE DROPPED
2965
2966      ; SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
2967      ; INPUTS:
2968      ; OUTPUTS:
2969      ; REGISTERS:
2970      ; CALLS:
2971
2972 017412      PRXST:: PRINTX #GETSTM
      017412 012746 005507      MOV    #GETSTM, -(SP)
      017416 012746 000001      MOV    #1, -(SP)
      017422 010600      MOV    SP, R0
      017424 104415      TRAP   C#PNTX
      017426 062706 000004      ADD    #4, SP
2973 017432      PRINTX #STAERS,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3,MSGPKT+MS.XS
      017432 016746 162734      MOV    MSGPKT+MS.XS4, -(SP)
      017436 016746 162726      MOV    MSGPKT+MS.XS3, -(SP)
      017442 016746 162720      MOV    MSGPKT+MS.XS2, -(SP)
      017446 016746 162712      MOV    MSGPKT+MS.XS1, -(SP)
      017452 016746 162704      MOV    MSGPKT+MS.XS0, -(SP)
      017456 012746 006753      MOV    #STAERS, -(SP)
      017462 012746 000006      MOV    #6, -(SP)
      017466 010600      MOV    SP, R0
      017470 104415      TRAP   C#PNTX
      017472 062706 000016      ADD    #16, SP
2974 017476 000207      RTS PC
2975
2976      ; SUBROUTINE TO HALT AFTER EACH COMMAND.
2977      ; INPUTS:
2978      ; OUTPUTS:
2979      ; REGISTERS: R3, R4
2980      ; CALLS:
2981
2982 017500 105767 162502      CKHAE:: TSTB   HAE,IFB HAE NE #0 THEN      ;IF HALT FLAG IS SET:
2983 017504 001430      BEQ    503231
2984 017506 105767 164017      TSTB   MISCFG      ;IFB MISCFG EQ #0 THEN
2985 017512 001023      BNE    503241
2986 017514      MANUAL      ;IS MANUAL INTERVENTION ALLOWED?
      017514 104450      TRAP   C#MANI
2987 017516      BNCOMPLETE CKHRTN      ;OR IF NOT.
      017516 103023      BCC    CKHRTN
2988 017520 016704 163674      MOV    CMDWRD,R4      ;LET R4 := CMDWRD
2989 017524 004767 171406      JSR    PC,GCHDA      ;FETCH ADR OF CMD ASCII.
2990 017530 112367 164552      MOVB   (R3)+,HALTM      ;MOVE CMD ASCII
2991 017534 112367 164547      MOVB   (R3)+,HALTM+1      ;LET HALTM+1 := (R3)+
2992 017540 111367 164544      MOVB   (R3),HALTM+2      ;INTO MESSAGE.

```

GLOBAL AREAS MACRO M1113 30-NOV-83 10:17
GLOBAL SUBROUTINES SECTION

SEQ 099

```

2993 017544          GMANIL HALTM,TIME1,1,YES      ;HALT - WAIT FOR AN OEPATOR INPUT.
      017544 104443
      017546 000404
      017550 003436
      017552 000130
      017554 004306
      017556 000001
      017560
2994 017560          100001:
2995 017560          100001:
2996 017560 000402
2997 017562          503241: BR      503251
2998 017562 105067 163743
2999 017562          503241: CLRB   MISCFCG      ;LET MISCFCG :B= #0
3000 017566          503251:
3001 017566          503231:
3002 017566 000207
3003 017566          CKHRTN: RTS      PC          ;RETURN
3004 017566          .EVEN
3005 017566          ENDMOD
3006 017570
3007
3008
3009 .TITLE MISCELLANEOUS SECTIONS
3010 .SBTTL REPORT CODING SECTION
3011
3012
3013
3014
3015
3016
3017
3018 017570          BGNRPT
      017570
3019 017570 010567 163656
3020 017574 004767 177302
3021 017600
3022 017600 026527 002604 177777
3023 017606 001562
3024 017610
      017610 016546 003376
      017614 016546 003326
      017620 016546 002604
      017624 012746 020432
      017630 012746 000004
      017634 010600
      017636 104416
      017640 062706 000012
3025 017644          PRINTS      #RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
      017644 016546 002626
      017650 016546 002636
      017654 016546 002646
      017660 016546 002656
      017664 012746 020507
      017670 012746 000005
      017674 010600
      017676 104416

      L#RPT::
      MOV      R5,RSSAVE          ;SAVE CURRENT DEVICE POINTER.
      JSR      PC,FIRSTU          ;FIND THE FIRST UNIT.
      503261: WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
      CMP      DEVTBL(R5),#END
      BEQ      503271
      PRINTS      #RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
      MOV      RECCNT(R5),-(SP)
      MOV      PASCNT(R5),-(SP)
      MOV      DEVTBL(R5),-(SP)
      MOV      #RPT1A, -(SP)
      MOV      #4, -(SP)
      MOV      SP,R0
      TRAP     C#PNTS
      ADD      #12,SP
      MOV      WRBC(R5),-(SP)
      MOV      WRBC+10(R5),-(SP)
      MOV      WRBC+20(R5),-(SP)
      MOV      WRBC+30(R5),-(SP)
      MOV      #RPT1B, -(SP)
      MOV      #5, -(SP)
      MOV      SP,R0
      TRAP     C#PNTS

```

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 100

3026	017700	062706	000014				ADD	#14,SP
	017704			PRINTS	#RPT1C,RRBC+30(R5),RRBC+20(R5),RRBC+10(R5),RRBC(R5)		MOV	RRBC(R5),-(SP)
	017710	016546	002666				MOV	RRBC+10(R5),-(SP)
	017714	016546	002706				MOV	RRBC+20(R5),-(SP)
	017720	016546	002716				MOV	RRBC+30(R5),-(SP)
	017724	012746	020560				MOV	#RPT1C,-(SP)
	017730	012746	000005				MOV	#5,-(SP)
	017734	010600					MOV	SP,R0
	017736	104416					TRAP	C#PNTS
3027	017740	062706	000014				ADD	#14,SP
	017744			PRINTS	#RPT1D,RFBC+30(R5),RFBC+20(R5),RFBC+10(R5),RFBC(R5)		MOV	RFBC(R5),-(SP)
	017750	016546	002726				MOV	RFBC+10(R5),-(SP)
	017754	016546	002736				MOV	RFBC+20(R5),-(SP)
	017760	016546	002746				MOV	RFBC+30(R5),-(SP)
	017764	012746	020631				MOV	#RPT1D,-(SP)
	017770	012746	000005				MOV	#5,-(SP)
	017774	010600					MOV	SP,R0
	017776	104416					TRAP	C#PNTS
3028	020000	062706	000014				ADD	#14,SP
	020004			PRINTS	#RPT1F,WRREC(R5),RRREC(R5),RFREC(R5)		MOV	RFREC(R5),-(SP)
	020010	016546	003026				MOV	RRREC(R5),-(SP)
	020014	016546	003006				MOV	WRREC(R5),-(SP)
	020020	016546	002766				MOV	#RPT1F,-(SP)
	020024	012746	020735				MOV	#4,-(SP)
	020030	012746	000004				MOV	SP,R0
	020032	010600					TRAP	C#PNTS
	020034	104416					ADD	#12,SP
3029	020040	062706	000012					
	020040			PRINTS	#RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)		MOV	RFUNR(R5),-(SP)
	020044	016546	003036				MOV	RRUNR(R5),-(SP)
	020048	016546	003016				MOV	WRUNR(R5),-(SP)
	020050	016546	002776				MOV	#RPT1G,-(SP)
	020054	012746	021006				MOV	#4,-(SP)
	020060	012746	000004				MOV	SP,R0
	020064	010600					TRAP	C#PNTS
	020066	104416					ADD	#12,SP
3030	020070	062706	000012					
3031	020074	105767	162110	TSTB	BADTSW	,IFB BADTSW NE #0 THEN		
3032	020100	001402		BEQ	503304			
3033	020102	004767	000056	JSR	PC,BTRPT	,GO PRINT BAD TAPE SPOTS WHEN ENABLED		
3034	020106							
3035	020106			503304:				
	020106			PRINTS	#RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)		MOV	VFYCNT(R5),-(SP)
	020112	016546	003346				MOV	FTLCNT(R5),-(SP)
	020116	016546	003366				MOV	HRDCNT(R5),-(SP)
	020122	016546	003356				MOV	SCCNT(R5),-(SP)
	020126	016546	003336				MOV	#RPT1I,-(SP)
	020132	012746	021203				MOV	#5,-(SP)
	020136	012746	000005				MOV	SP,R0
	020140	010600					TRAP	C#PNTS
	020142	104416					ADD	#14,SP
3036	020146	062706	000014					
3037	020146	004767	176776	JSR	PC,NEXTU	,FIND THE NEXT UNIT.		
3038	020152	000612		BR	503264			

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
REPORT CODING SECTION

SEQ 101

```

3039 020154      503274:
3040 020154 016705 163272      MOV    R5SAVE,R5      ;RESTORE CURRENT DEVICE POINTER.
3041 020160      EXIT      RPT
      020160 000167      .WORD    J$JMP
      020162 001130      .WORD    L10010-2-.

3042
3043      ;      SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
3044      ;      WRITE RETRIES: CUMULATIVE COUNT
3045      ;      BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
3046      ;      COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
3047
3048 020164      BTRPT:: PRINTS  #RPT1E,WRTYCT(R5)      ;PRINT GLOBAL WRITE RETRY COUNT
      020164 016546 003316      MOV    WRTYCT(R5),-(SP)
      020170 012746 021057      MOV    #RPT1E, -(SP)
      020174 012746 000002      MOV    #2, -(SP)
      020200 010600      MOV    SP,R0
      020202 104416      TRAP    C$PNTS
      020204 062706 000006      ADD    #6,SP
3049 020210 016567 002616 163274      MOV    BTADDR(R5),BTPT      ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
3050 020216 017703 163270      MOV    #BTPT,R3      ;AND THE LOGGING INDEX
3051 020222 006203      ASR    R3
3052 020224      PRINTS  #RPT1J,R3      ;PRINT # OF BAD TAPE SPOTS
      020224 010346      MOV    R3, -(SP)
      020226 012746 021107      MOV    #RPT1J, -(SP)
      020232 012746 000002      MOV    #2, -(SP)
      020236 010600      MOV    SP,R0
      020240 104416      TRAP    C$PNTS
      020242 062706 000006      ADD    #6,SP
3053 020246 005703      TST    R3      ;PRINT RECORD # IF BAD SPOTS DETECTED
3054 020250 001457      BEQ    503314
3055 020252 020327 000024      CMP    R3,#20.      ;IF R3 HI #20. THEN
3056 020256 101402      BLOS   503324
3057 020260 012703 000024      MOV    #20.,R3      ;20 BAD SPOTS IS THE LIMIT
3058
3059 020264      503324:
3060 020264      PRINTS  #CRLFSP
      020264 012746 005744      MOV    #CRLFSP, -(SP)
      020270 012746 000001      MOV    #1, -(SP)
      020274 010600      MOV    SP,R0
      020276 104416      TRAP    C$PNTS
      020300 062706 000004      ADD    #4,SP
3061 020304 016704 163202      MOV    BTPT,R4      ;LET R4 := BTPT * #2 ;FETCH A BAD SPOT ID
3062 020310 062704 000002      ADD    #2,R4
3063 020314 005002      CLR    R2      ;R2 = PRINT COUNT PER LINE: 10 MAX
3064 020316      503334: ;REPEAT
3065 020316      PRINTS  #RPT1K,(R4)      ;PRINT A BAD SPOT ID
      020316 011446      MOV    (R4), -(SP)
      020320 012746 021174      MOV    #RPT1K, -(SP)
      020324 012746 000002      MOV    #2, -(SP)
      020330 010600      MOV    SP,R0
      020332 104416      TRAP    C$PNTS
      020334 062706 000006      ADD    #6,SP
3066 020340 005202      INC    R2      ;LET R2 := R2 + #1 ;COUNT PRINTS
3067 020342 062704 000002      ADD    #2,R4      ;LET R4 := R4 + #2 ;NEXT
3068 020346 020227 000012      CMP    R2,#10.      ;IF R2 EQ #10. THEN
3069 020352 001014      BNE    503344
3070 020354      PRINTS  #CRLFSP      ;GO TO NEXT PRINT LINE PAST 10 PRINTS

```

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 102

```

020354 012746 005744
020360 012746 000001
020364 010600
020366 104416
020370 062706 000004
3071 020374 162703 000012
3072 020400 162702 000012
3073
3074 020404
3075 020404 020203
3076 020406 001343
3077
3078 020410
3079 020410
020410 012746 005741
020414 012746 000001
020420 010600
020422 104416
020424 062706 000004
3080 020430 000207
3081
3082
3083 020432 045 116 045 RPT1A: .NLIST BEX
3084 020507 045 101 102 RPT1B: .ASCIZ /#N#AUNIT #D1#S3#APASS:#D5#S3#ARECORD:#D5#N/
3085 020560 045 101 102 RPT1C: .ASCIZ /#BYTES WRITTEN #D3#A,#Z3#A,#Z3#A,#Z3#N/
3086 020631 045 101 102 RPT1D: .ASCIZ /#BYTES READ REV #D3#A,#Z3#A,#Z3#A,#Z3#N/
3087 020701 045 123 062 RPT1E: .ASCIZ /#BYTES READ FWD #D3#A,#Z3#A,#Z3#A,#Z3#N/
3088 020735 045 101 122 RPT1F: .ASCIZ /#S23#AMRT#S4#ARDR#S4#ARDF#N/
3089 021006 045 101 125 RPT1G: .ASCIZ /#ARECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
3090 021057 045 101 127 RPT1H: .ASCIZ /#AUNRECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
3091 021107 045 116 045 RPT1I: .ASCIZ /#WRITE RETRIES#S0#D5#N/
3092 021174 045 104 065 RPT1J: .ASCIZ /#N#D2#A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
3093 021203 045 101 123 RPT1K: .ASCIZ /#D5#S1/
3094 021257 045 123 063 RPT1L: .ASCIZ /#ASPEC COND#S3#AHARD#S3#AFATAL#S3#ACOMPARE#N"
3095 .ASCIZ /#S3#D5#S3#D5#S3#D5#S3#D5#N#N/
3096 .LIST BEX
3097 .EVEN
3098 021314
021314 ENDRPT
021314 104425 L10010: TRAP C#RPT

3099
3100 .SBTTL LOAD DEVICE PROTECTION TABLE
3101
3102
3103
3104
3105
3106
3107 021316
021316 BGNPROT
L#PROT::
3108
3109 021316 000000 .WORD 0 ;P-TBL OFFSET OF TSDB
3110 021320 177777 .WORD -1 ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MASS BUS DE
3111 021322 177777 .WORD -1 ;P-TBL OFFSET OF DRIVE #: -1 = NONE, THREE DRIVES PER CONTRO
LLER
3112 021324
3113
3114 .SBTTL INITIALIZE SECTION

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
INITIALIZE SECTION

SEQ 103

```

3115
3116
3117      ;
3118      ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
3119      ; AT THE BEGINNING OF EACH PASS.
3120      ;--
3121      021324      BGNINIT
3122      021324      L$INIT::
3123      021324      032727      000003      002330      INIT10: BIT      #BIT0!BIT1,#CMDPKT      ;IF CMD PACKET IS NOT ON MODULO 4 BOUNDARY:
3124      021332      001421      BEQ      50335$
3125      021334      ERRSF      1,CMDPKM      ;PRINT ERROR MSG.
3126      021344      012727      000310      DELAY      200.      ;GO TO SUPERVISOR, WAIT 2 SECONDS.
3127      021374      000753      BR      INIT10
3128
3129      50335$:
3130
3131      021376      105767      160602      TSTB      CLRFLG      ;IF CLR COUNTERS FLAG SET:
3132      021402      001413      BEQ      50336$
3133      021404      105067      160574      CLR      CLRFLG      ;INIT CLR FLAG.
3134      021410      005002      CLR      R2
3135      021412      50337$: WHILE R2 NE #CNTLEN DO
3136      021412      020227      000550      CMP      R2,#CNTLEN
3137      021416      001405      BEQ      50340$
3138      021420      005062      002626      CLR      WRBC(R2)      ;CLR ALL STATISTICAL COUNTERS.
3139      021424      062702      000002      ADD      #2,R2      ;LET R2 := R2 + #2
3140
3141      021430      000770      BR      50337$
3142      021432      50340$:
3143
3144      021432      50336$:
3145
3146      021432      105767      160547      TSTB      RRANV      ;IF RESET RANDOM VARIABLE FLAG IS SET THEN:
3147      021436      001406      BEQ      50341$
3148      021440      012767      153624      161764      MOV      #RANBC,RANB      ;RESET RANDOM BASE #.
3149
3150      021446      012767      032561      161760      MOV      #RANSC,RANS      ;RESET RANDOM SAVE LOCATION.
3151
3152      021454      50341$:
3153      021454      012700      000040      READEF      #EF.START      ;READ START COMMAND EVENT FLAG.
3154      021462      103057      BNCOMPLETE      INIT15      ;BRANCH IF NOT STARTING.
3155      021464      105267      162036      INCB      STAFLG      ;SET START COMMAND FLAG.

```

```

3156 021470 012705 000006      MOV      #6,R5      ;LET R5 := #6
3157 021474      50342: ;REPEAT
3158 021474 012765 177774 002604  MOV      #NINUSE,DEVTBL(R5) ;INITIATE UNIT NUMBER TABLE
3159 021502 162705 000002      SUB      #2,R5      ;LET R5 := R5 - #2
3160 021506 005705      TST      R5      ;UNTIL R5 EQ #0
3161 021510 001371      BNE      50342:
3162 021512 022767 000001 160272  CMP      #1,L#UNIT      ;ONLY ONE UNIT ALLOWED
3163 021520 001425      BEQ      5034:      ;OK
3164 021522      PRINTF      #AUDRUN      ;TELL THE MAN
      021522 012746 005146      MOV      #AUDRUN,-(SP)
      021526 012746 000001      MOV      #1,-(SP)
      021532 010600      MOV      SP,R0
      021534 104417      TRAP      C#PNTF
      021536 062706 000004      ADD      #4,SP
3165 021542      DELAY      25      ;WAIT
      021542 012727 000025      MOV      #25,(PC)+
      021546 000000      .WORD      0
      021550 016727 160342      MOV      L#DLY,(PC)+
      021554 000000      .WORD      0
      021556 005367 177772      DEC      -6(PC)
      021562 001375      BNE      -4
      021564 005367 177756      DEC      -22(PC)
      021570 001367      BNE      -20
3166 021572      DOCLN      ;ABORT
      021572 104444      TRAP      C#DCLN
3167 021574 016705 160212 50344: MOV      L#UNIT,R5      ;LET R5 := L#UNIT SHIFT 1
3168 021600 006305      ASL      R5
3169 021602      50343: ;REPEAT      ;STORE ALL UNIT
3170 021602 162705 000002      SUB      #2,R5      ;LET R5 := R5 - #2 ;NUMBERS IN DEVTBL.
3171 021606 010565 002604      MOV      R5,DEVTBL(R5) ;LET DEVTBL(R5) := R5 SHIFT -1
3172 021612 006265 002604      ASR      DEVTBL(R5)
3173 021616 005705      TST      R5      ;UNTIL R5 EQ #0
3174 021620 001370      BNE      50343:
3175
3176 021622      INIT15: REDEF      #EF.PWR      ;HAS THERE BE A POWER FAILURE?
      021622 012700 000034      MOV      #EF.PWR,R0
      021626 104447      TRAP      C#REFG
3177 021630      BNCOMPLETE INIT16      ;BRANCH IF NOT.
      021630 103004      BCC      INIT16
3178 021632 105267 161670      INCB      STAFLG      ;IF SO - SET THE START FLAG.
3179 021636 105267 161665      INCB      PWRFLG      ;IF SO - SET THE POWER FAIL FLAG.
3180
3181 021642      INIT16: RFLAGS      OPFLAG      ;READ AND STORE FLAGS SET BY OPERATOR
      021642 104421      TRAP      C#RFLA
      021644 010067 161666      MOV      R0,OPFLAG
3182 021650 005003      CLR      R3      ;LET R3 := #0      ;CLEAR EVENT FLAG
3183 021652 105767 161651      TSTB      PWRFLG      ;IF POWER FAIL HAS NOT OCCURRED THEN:
3184 021656 001020      BNE      50344:
3185 021660      READDEF      #EF.NEW      ;UPDATE PASS COUNT WHEN
      021660 012700 000035      MOV      #EF.NEW,R0
      021664 104447      TRAP      C#REFG
3186 021666 103014      BCC      50345:      ;SUPERVISOR IS IN NEW PASS
3187 021670 105767 161632      TSTB      STAFLG      ;AND DIAG WAS NEITHER STARTED
3188 021674 001010      BNE      50346:
3189 021676      READDEF      #EF.RES      ;NOR
      021676 012700 000037      MOV      #EF.RES,R0
      021702 104447      TRAP      C#REFG

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
INITIALIZE SECTION

SEQ 105

```

3190 021704 103402      BCS      503470      ;IFCOND CC THEN      ;RESTARTED
3191 021706 005103      COM      R3          ;LET R3 := COMP R3 ;DO IT
3192
3193 021710 000401      BR        503500
3194 021712      503470:      INC      R3          ;SET 1ST PASS IF NEW PASS AND
3195 021712 005203      ;RESTARTING
3196
3197 021714      503500:
3198
3199 021714 000401      BR        503510
3200 021716      503460:      INC      R3          ;SET 1ST PASS IF NEW PASS AND
3201 021716 005203      ;STARTING
3202
3203 021720      503510:
3204
3205 021720      503450:      ;DO NOT UPDATE IT ON CONTINUE
3206
3207 021720      503440:      ;OR ON POWER FAIL
3208 021720 004767 175156      JSR      PC,FIRSTU      ;INIT DEVICE POINTER.
3209 021724 005002      CLR      R2          ;LET R2 := #0      ;INIT DEVICE COUNTER.
3210 021726      503520:      ;WHILE DEVTBL(R5) NE #END DO
3211 021726 026527 002604 177777      CMP      DEVTBL(R5),#END DO
3212 021734 001456      BEQ      503530
3213 021736 005202      INC      R2          ;LET R2 := R2 + #1
3214 021740 010500      MOV      R5,R0      ;LET R0 := R5 SHIFT -1
3215 021742 006200      ASR      R0
3216 021744      GPHARD      R0,R0          ;GET HARDWARE P TABLE FROM SUPER.
3217 021744 104442      ;TRAP      C#GPHRD
3218 021746 103044      BCC      503540      ;IFCOND CS THEN
3219 021750 011065 002514      MOV      (R0),TSDB(R5)      ;SAVE TSDB ADDRESS.
3220 021754 012065 002524      MOV      (R0),TSSR(R5)      ;SAVE TSSR ADDRESS.
3221 021760 062765 000002 002524      ADD      #2,TSSR(R5)
3222 021766 012065 002534      MOV      (R0),TSVCT(R5)      ;SAVE INTERRUPT VECTOR ADDRESS.
3223 021772 011065 003532      MOV      (R0),TSUNT(R5)      ;SAVE NUMBER OF DRIVE
3224 021776 011067 161532      MOV      (R0),TSNP      ;SAVE FOR PRINT OUT'S
3225 022002      SETVEC      TSVCT(R5),TSSINT(R5),#INTPRI
3226 022002 012746 000340      MOV      #INTPRI,-(SP)
3227 022006 016546 002554      MOV      TSSINT(R5),-(SP)
3228 022012 016546 002534      MOV      TSVCT(R5),-(SP)
3229 022016 012746 000003      MOV      #3,-(SP)
3230 022022 104437      TRAP      C#SVEC
3231 022024 062706 000010      ADD      #10,SP
3232
3233 022030 005065 003472      CLR      INTFLG(R5)      ;SET UP INTERRUPT PROCESSING CONDITIONS.
3234 022034 005703      TST      R3          ;CLEAR INTERRUPT FLAGS.
3235 022036 001410      REQ      503550      ;ACTUAL PASSCOUNT UPDATE PER R3
3236 022040 005703      TST      R3
3237 022042 002003      BGE      503560      ;IF R3 LT #0 THEN
3238 022044 005265 003326      INC      PASCNT(R5)      ;LET PASCNT(R5) := PASCNT(R5) + #1
3239
3240 022050 000403      BR        503570
3241 022052      503560:      MOV      #1,PASCNT(R5)      ;LET PASCNT(R5) := #1
3242
3243 022052 012765 000001 003326      503570:
3244
3245 022060      503550:

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
INITIALIZE SECTION

SEQ 106

```

3240
3241 022060          50354: CLR    RECCNT(R5)          ;CLEAR RECORD COUNT
3242 022060 005065 003376      JSR    PC,NEXTU          ;DO IT FOR ALL DEVICES.
3243 022064 004767 175060
3244
3245 022070 000716          BR      50352:
3246 022072          50353:
3247
3248 022072 005702          TST    R2                  ;IF THERE ARE NO UNITS:
3249 022074 001026          BNE    50360:              ;PRINT ALL UNITS DROPPED.
3250 022076          PRINTF #AUDRPM
      022076 012746 005114          MOV    #AUDRPM,-(SP)
      022102 012746 000001          MOV    #1,-(SP)
      022106 010600          MOV    SP,R0
      022110 104417          TRAP   C#PNTF
      022112 062706 000004          ADD    #4,SP
3251 022116          DELAY 200.          ;GO TO SUPERVISOR, WAIT 2 SECONDS.
      022116 012727 000310          MOV    #200.,(PC)+
      022122 000000          .WORD   0
      022124 016727 157766          MOV    L#DLY,(PC)+
      022130 000000          .WORD   0
      022132 005367 177772          DEC    -6(PC)
      022136 11375          BNE     -4
      022140 05367 177756          DEC    -22(PC)
      022144 001367          BNE     -20
3252 022146          BREAK              ;GO TO SUPERVISOR, CHECK TTY.
      022146 104422          TRAP   C#BRK
3253 022150          DOCLN              ;DO CLEAN CODE + ABORT PASS.
      022150 104444          TRAP   C#DCLN
3254
3255 022152          50360:
3256
3257
3258 022152          SETPRI #PRI00          ;LOWER CPU PRIORITY TO 0
      022152 012700 000000          MOV    #PRI00,R0
      022156 104441          TRAP   C#SPRI
3259 022160 105767 160027          ;IF ERROR RECOVERY IS ENABLED
3260 022164 001033
3261 022166 032767 000020 161342      TSTB   IREC
3262 022174 001027          BNE     1:
3263 022176 004767 174700          BIT     #ADR,OPFLAG
3264 022202          BNE     1:
3265 022202 026527 002604 177777      JSR    PC,FIRSTU
3266 022210 001421          50362: CMP     DEVTBL(R5),#END
3267 022212 105067 161312          BEQ     1:
3268 022216          CLR      TRAPD4
      022216 012746 000340          SETVEC  #4,#TRAP4,#INTPRI
      022222 012746 023730          ;CLEAR TRAP FLAG
      022226 012746 000004          ;SET VECTOR 4,PRIORITY 06
      022232 012746 000003          MOV    #INTPRI,-(SP)
      022236 104437          MOV    #TRAP4,-(SP)
      022240 062706 000010          MOV    #4,-(SP)
      022244 012767 000001 161164      MOV    #3,-(SP)
3269 022244 012767 000001 161164      TRAP   C#SVEC
3270 022252 000404          ADD    #10,SP
3271 022252 000404
3272 022254 000167 000622          ;START 3.5 MINUTE COUNTER
3273          ;INCR TIME1 FROM #1 TO #25 BY #1
      022254 000167 000622          MOV    #1,TIME1
      022254 000167 000622          BR     50365:
      022254 000167 000622          JMP    50363:

```

MISCELLANEOUS SECTIONS
INITIALIZE SECTION

MACRO M1113 30-NOV-83 10:17

SEQ 107

3274	022260			503661:					
3275	022260	005267	161152		INC	TIME1			
3276	022264			503651:					
3277	022264	026727	161146	000025	CMP	TIME1,#25			
3278	022272	003134			BGT	41			
3279	022274	012775	002340	002514	MOV	#GSCPK,#TSDB(R5)		AND GET UNITS STATUS	
3280	022302				DELAY	25		WAIT	
	022302	012727	000025					MOV	#25,(PC)+
	022306	000000						.WORD	0
	022310	016727	157602					MOV	L#DLY,(PC)+
	022314	000000						.WORD	0
	022316	005367	177772					DEC	-6(PC)
	022322	001375						BNE	.-4
	022324	005367	177756					DEC	-22(PC)
	022330	001367						BNE	.-20
3281	022332				CLRVEC	#4		CLER VECTOR AT 4	
	022332	012700	000004					MOV	#4,R0
	022336	104436						TRAP	C#CVEC
3282	022340	105767	161164		TSTB	TRAPD4		IFB TRAPD4 NE #0 THEN	
3283	022344	001423			BEQ	21			
3284	022346	005265	003366		INC	FTLCNT(R5)		LET FTLCNT(R5) := FTLCNT(R5) + #1	
3285	022352				PRINTF	#NODEV,TSSR(R5)		PRINT ERROR	
	022352	016546	002524					MOV	TSSR(R5),-(SP)
	022356	012746	005543					MOV	#NODEV,-(SP)
	022362	012746	000002					MOV	#2,-(SP)
	022366	010600						MOV	SP,R0
	022370	104417						TRAP	C#PNTF
	022372	062706	000006					ADD	#6,SP
3286	022376	016567	002604	175004	MOV	DEVTBL(R5),DROPN		SAVE # OF UNIT TO BE DROPPED.	
3287	022404	010500			MOV	R5,R0		R0=LOGICAL DEVICE NUMBER	
3288	022406	006200			ASR	R0			
3289	022410				ODD!	R0		DROP THE UNIT	
	022410	104451						TRAP	C#ODDU
3290								EXEC BGNDU-ENDDU CODE IF IDU = 0	
3291	022412				DOCLN			DO CLEAN ABORT	
	022412	104444						TRAP	C#DCLN
3292									
3293	022414	105067	161110	21:	CLRB	TRAPD4		CLEAR TRAP FLAG	
3294	022420				SETVEC	#4,#TRAP4,#INTPRI		SET VECTOR 4,PRIORITY #6	
	022420	012746	000340					MOV	#INTPRI,-(SP)
	022424	012746	023730					MOV	#TRAP4,-(SP)
	022430	012746	000004					MOV	#4,-(SP)
	022434	012746	000003					MOV	#3,-(SP)
	022440	104437						TRAP	C#SVEC
	022442	062706	000010					ADD	#10,SP
3295	022446	005775	002524		TST	#TSSR(R5)		CHECK FOR ADDRESS	
3296	022452				DELAY	25		WAIT	
	022452	012727	000025					MOV	#25,(PC)+
	022456	000000						.WORD	0
	022460	016727	157432					MOV	L#DLY,(PC)+
	022464	000000						.WORD	0
	022466	005367	177772					DEC	-6(PC)
	022472	001375						BNE	.-4
	022474	005367	177756					DEC	-22(PC)
	022500	001367						BNE	.-20
3297	022502				CLRVEC	#4		CLER VECTOR AT 4	
	022502	012700	000004					MOV	#4,R0

Address	Offset	Hex	Dec	Label	Instruction	Comment
022506	104436					
3298	022510	105767	161014		TSTB TRAPD4	;IFB TRAPD4 NE #0 THEN TRAP C#CVEC
3299	022514	001424			BEQ 3#	
3300	022516	005265	003366		INC FTLCNT(R5)	;LET FTLCNT(R5) := FTLCNT(R5) + #1
3301	022522				PRINTF #NODEV,TSSR(R5)	;PRINT ERROR
	022522	016546	002524			MOV TSSR(R5),-(SP)
	022526	012746	005543			MOV #NODEV,-(SP)
	022532	012746	000002			MOV #2,-(SP)
	022536	010600				MOV SP,R0
	022540	104417				TRAP C#PNTF
	022542	062706	000006			ADD #6,SP
3302	022546	016567	002604	174634	MOV DEVTBL(R5),DROPN	;SAVE # OF UNIT TO BE DROPPED.
3303	022554	010500			MOV R5,R0	;R0=LOGICAL DEVICE NUMBER
3304	022556	006200			ASR R0	
3305	022560				DODU R0	;DROP THE UNIT
	022560	104451				TRAP C#DODU
3306						; EXEC BGNDU-ENDDU CODE IF IDU = 0
3307	022562				DOCLN	;DO CLEAN &ABORT
	022562	104444				TRAP C#DCLN
3308						
3309	022564	003127		4#:	BGT 50367#	
3310						
3311	022566	004767	165174	3#:	JSR PC,SETDEF	;SET UNIT NUMBER
3312	022572	010475	002514		MOV R4,@TSDB(R5)	
3313	022576				DELAY 25	
	022576	012727	000025			MOV #25,(PC)+
	022602	000000				.WORD 0
	022604	016727	157306			MOV L#DLY,(PC)+
	022610	000000				.WORD 0
	022612	005367	177772			DEC -6(PC)
	022616	001375				BNE .-4
	022620	005367	177756			DEC -22(PC)
	022624	001367				BNE .-20
3314	022626	012775	002340	002514	MOV #GSCP,K,@TSDB(R5)	;AND GET UNITS STATUS
3315	022634				DELAY 25	;WAIT
	022634	012727	000025			MOV #25,(PC)+
	022640	000000				.WORD 0
	022642	016727	157250			MOV L#DLY,(PC)+
	022646	000000				.WORD 0
	022650	005367	177772			DEC -6(PC)
	022654	001375				BNE .-4
	022656	005367	177756			DEC -22(PC)
	022662	001367				BNE .-20
3316	022664	032775	000200	002524	BIT #TS.SSR,@TSSR(R5)	;IF #TS.SSR SETIN @TSSR(R5) THEN
3317	022672	001420			BEQ 50370#	
3318	022674	032775	000100	002524	BIT #TS.OFL,@TSSR(R5)	;IF #TS.OFL NOTSETIN @TSSR(R5) THEN
3319	022702	001001			BNE 50371#	
3320	022704	000457			BR 50364#	;EXIT COUNTER WHEN UNIT ON LINE
3321						
3322	022706			50371#:		
3323	022706				PRINTF #OFLINM,TSNP	;PRINT UNIT OFF LINE EVERY 10 SEC
	022706	016746	160622			MOV TSNP,-(SP)
	022712	012746	005456			MOV #OFLINM,-(SP)
	022716	012746	000002			MOV #2,-(SP)
	022722	010600				MOV SP,R0
	022724	104417				TRAP C#PNTF
	022726	062706	000006			ADD #6,SP

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
INITIALIZE SECTION

SEQ 109

```

3324
3325 022732          503724:
3326
3327 022732 000412          503734: BR      503734
3328 022734          503704:          PRINTF #NRDYM,DEVTBL(R5)
3329 022734          MOV      DEVTBL(R5),-(SP)
      022734 016546 002604      MOV      #NRDYM, -(SP)
      022740 012746 023700      MOV      #2, -(SP)
      022744 012746 000002      MOV      SP,R0
      022750 010600      TRAP    C1PNTF
      022752 104417      ADD     #6,SP
      022754 062706 000006
3330
3331 022760          503734:          ;INCR TIME2 FROM #1 TO #13 BY #1
3332 022760 012767 000001 160452      MOV      #1,TIME2
3333 022766 000402          BR      503744
3334 022770          503754:          ;
3335 022770 005267 160444          INC      TIME2
3336 022774          503744:          ;
3337 022774 026727 160440 000013      CMP      TIME2,#13
3338 023002 003016          BGT      503764
3339 023004          DELAY    100.          ;WAIT FOR UNIT TO BE SET ON-LINE
      023004 012727 000144          MOV      #100.,(PC)+
      023010 000000          .WORD    0
      023012 016727 157100          MOV      L1DLY,(PC)+
      023016 000000          .WORD    0
      023020 005367 177772          DEC      -6(PC)
      023024 001375          BNE      .-4
      023026 005367 177756          DEC      -22(PC)
      023032 001367          BNE      .-20
3340 023034          BREAK          ;ALLOW TERMINAL INTERRUPT
      023034 104422          TRAP    C1BRK
3341 023036 000754          BR      503754
3342 023040          503764:          ;
3343 023040 000167 177214          JMP      503664
3344 023044          503674:          ;
3345 023044          503644:          ;CLEAR VECTOR AT 4
3346 023044          CLRVEC #4          MOV      #4,R0
      023044 012700 000004          TRAP    C1CVEC
      023050 104436
3347 023052 026727 160360 000025      CMP      TIME1,#25
3348 023060 003404          BLE      503774          ;IF OFF LINE FOR 3.5 MINUTES
3349 023062 004767 167646          JSR PC,MOVMSG          ;GET MESSAGE PACKET
3350 023066 004767 170360          JSR PC,TCC1          ;PRINT ERROR AND DROP OFF LINE UNIT
3351
3352 023072          503774:          ;REPEAT UNTIL ON LINE OR TIMED OUT.
3353
3354 023072 004767 174052          JSR PC,NEXTU          ;SET UP FOR NEXT UNIT.
3355
3356 023076 000167 177100          JMP      503624
3357
3358 023102          503634:          ;
3359 023102          503614:          ;
3360 023102 105767 160421          TSTB    PWRFLG          ;IFB PWRFLG EQ #0 THEN
3361 023106 001026          BNE      504004
3362 023110          MEMORY DATAWT          ;REQUEST MEMORY FROM SUPER FOR RD/WR BUFFERS.
      023110 104431          TRAP    C1MEM

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
INITIALIZE SECTION

SEQ 110

```

3363 023112 010067 160270
3364 023116 016767 160264 160264
3365 023124 062767 004000 160256
3366 023132 027727 160250 004000
3367 023140 002011
3367 023142 012746 023210
3367 023146 012746 000001
3367 023152 010600
3367 023154 104417
3367 023156 062706 000004
3368 023162
3368 023162 104444
3369
3370 023164 504011:
3371
3372 023164 504001:
3373
3374 023164 105067 157024
3375 023170 012703 003526
3376 023174 004767 167464
3377 023200 105067 160323
3378
3379 023204
3379 023204 104432
3379 023206 000104
3380 023210 045 101 106 MEMOM: .ASCII /%AFREE MEMO TOO SMALL FOR RD-WR BFRS%N/
3380 023213 122 105 105
3380 023216 040 115 105
3380 023221 115 117 040
3380 023224 124 117 117
3380 023227 040 123 115
3380 023232 101 114 114
3380 023235 040 106 117
3380 023240 122 040 122
3380 023243 104 055 127
3380 023246 122 040 102
3380 023251 106 122 123
3380 023254 045 116
3381 023256 045 101 122 .ASCIZ /%ARE-LOAD IN LARGER MEMOM%N/
3381 023261 105 055 114
3381 023264 117 101 104
3381 023267 040 111 116
3381 023272 040 114 101
3381 023275 122 107 105
3381 023300 122 040 115
3381 023303 105 115 117
3381 023306 045 116 000
3382
3383
3384 023312
3384 023312
3384 023312 104411
3385
3386
3387
3388

          .EVEN
          ENDINIT
L10012:
          TRAP C:INIT

          .SBTTL AUTO DROP SECTION
          ***

```

```

3389      ;SECTION EXECUTED AFTER THE INIT CODE WHEN "ADR" FLAG IS SET BY OPERATOR
3390      ;SECTION CHEKS FOR A VALID INTERFACE LOCATION.  DROPS UNIT IF NO RESPONSE
3391      ;FROM INTERFACE
3392      ;--
3393
3394 023314      BGNAUTO
023314      L$AUTO::
3395
3396 023314 004767 173562      JSR PC,FIRSTU      ;FIND FIRST UNIT
3397 023320      504024: ;WHILE DEVTBL(R5) NE #END DO      ;
3398 023320 026527 002604 177777      CMP      DEVTBL(R5),#END
3399 023326 001525      BEQ      504034
3400 023330 105067 160174      CLRB      TRAPD4      ;LET TRAPD4 :B= #0
3401 023334      SETVEC #4,#TRAP4,#INTPRI      ;SET VECTOR 4
023334 012746 000340      MOV      #INTPRI,-(SP)
023340 012746 023730      MOV      #TRAP4,-(SP)
023344 012746 000004      MOV      #4,-(SP)
023350 012746 000003      MOV      #3,-(SP)
023354 104437      TRAP      C$SVEC
023356 062706 000010      ADD      #10,SP
3402 023362 017502 002514      MOV      @TSDB(R5),R2      ;ADDRESS TS05 INTERFACE
3403 023366      CLRVEC #4      ;CLEAR VECTOR AT 4
023366 012700 000004      MOV      #4,R0
023372 104436      TRAP      C$CVEC
3404 023374 105767 160130      TSTB      TRAPD4      ;IFB TRAPD4 NE #0 THEN
3405 023400 001423      BEQ      504044
3406 023402 005265 003366      INC      FTLCNT(R5)      ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3407 023406      PRINTF #AUTODM,TSDB(R5)      ;PRINT ERROR
023406 016546 002514      MOV      TSDB(R5),-(SP)
023412 012746 023604      MOV      #AUTODM,-(SP)
023416 012746 000002      MOV      #2,-(SP)
023422 010600      MOV      SP,R0
023424 104417      TRAP      C$PNTF
023426 062706 000006      ADD      #6,SP
3408 023432 016567 002604 173750      MOV      DEVTBL(R5),DROPN      ;SAVE # OF UNIT TO BE DROPPED.
3409 023440 010500      MOV      R5,R0      ;R0=LOGICAL DEVICE NUMBER
3410 023442 006200      ASR      R0
3411 023444      DODU      R0      ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
023444 104451      TRAP      C$DODU
3412
3413 023446 000452      BR      504054
3414 023450      504044:
3415 023450 012775 002340 002514      MOV      #GSCP, @TSDB(R5)      ;SEND GET STATUS COMMAND
3416 023456 004767 167216      JSR      PC,WSSR      ;WAIT
3417 023462 032775 000200 002524      BIT      #TS.SSR,@TSSR(R5)      ;IF #TS.SSR SETIN @TSSR(R5) THEN
3418 023470 001423      BEQ      504064
3419 023472 032775 000100 002524      BIT      #TS.OFL,@TSSR(R5)      ;IF #TS.OFL SETIN @TSSR(R5) THEN
3420 023500 001416      BEQ      504074
3421 023502 005265 003366      INC      FTLCNT(R5)      ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3422 023506      PRINTF #OFLINM,TSNP
023506 016746 160022      MOV      TSNP,-(SP)
023512 012746 005456      MOV      #OFLINM,-(SP)
023516 012746 000002      MOV      #2,-(SP)
023522 010600      MOV      SP,R0
023524 104417      TRAP      C$PNTF
023526 062706 000006      ADD      #6,SP
3423 023532 004767 173566      JSR PC,DROPUA

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
 AUTO DROP SECTION

SEQ 112

```

3424
3425 023536          50407:
3426
3427 023536 000416          BR      50410:
3428 023540          50406:
3429 023540 005265 003366          INC      FTLCNT(R5)      ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3430 023544          PRINTF #NRDYM,DEVTBL(R5)
      023544 016546 002604          MOV      DEVTBL(R5),-(SP)
      023550 012746 023700          MOV      #NRDYM, -(SP)
      023554 012746 000002          MOV      #2, -(SP)
      023560 010600          MOV      SP,R0
      023562 104417          TRAP     C#PNTF
      023564 052706 000006          ADD      #6,SP
3431 023570 004767 173530          JSR PC,DROPUA
3432
3433 023574          50410:
3434
3435 023574          50405:
3436 023574 004767 173350          JSR PC,NEXTU
3437
3438 023600 000647          BR      50402:
3439 023602          50403:
3440
3441 023602          ENDAUTO
      023602          L10013:
      023602 104461          TRAP     C#AUTO
3442
3443 023604 045 101 102 AUTODM: .ASCII /#ABUS TRAP AT #06#N/
      023607 125 123 040
      023612 124 122 101
      023615 120 040 101
      023620 124 040 045
      023623 117 066 045
      023626 116
3444 023627 045 101 111 .ASCIZ /#AINTERFACE BAD OR NOT SET TO ABOVE AD#N/
      023632 116 124 105
      023635 122 106 101
      023640 103 105 040
      023643 102 101 104
      023646 040 117 122
      023651 040 116 117
      023654 124 040 123
      023657 105 124 040
      023662 124 117 040
      023665 101 102 117
      023670 126 105 040
      023673 101 104 045
      023676 116 000
3445 023700 045 101 125 NRDYM: .ASCIZ /#AUNIT #D1#A NOT RDY#N/
      023703 116 111 124
      023706 040 045 104
      023711 061 045 101
      023714 040 116 117
      023717 124 040 122
      023722 104 131 045
      023725 116 000
3446 .EVEN

```

```

3447
3448      ;      DEVICE BUS TRAP HANDLER
3449      ;      OUTPUT: TRAPD4 BYTE  1: TRAPED AT 4
3450      ;      ;      0: NO TRAP
3451
3452 023730 105267 157574      TRAP4:: INCB      TRAPD4,LET TRAPD4 :B= TRAPD4 + #1
3453 023734 000002      RTI
3454
3455      .SBTTL  CLEANUP CODING SECTION
3456
3457      ;**
3458      ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
3459      ; AT THE END OF EACH PASS.
3460      ;--
3461
3462
3463 023736      BGNCLN
3464 023736      L$CLEAN::
3465
3465 023736 004767 173140      JSR      PC,FIRSTU      ;FIND FIRST UNIT.
3466 023742      504110: ;WHILE DEVTBL(R5) NE #END DO
3467 023742 026527 002604 177777      CMP      DEVTBL(R5),#END
3468 023750 001410      BEQ      504120
3469 023752 004767 166722      JSR PC,WSSR      ;WAIT FOR UNIT READY OR TIMEOUT.
3470 023756      CLRVEC      TSVCT(R5)      ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
3471 023764 004767 173160      JSR      PC,NEXTU      ;FIND NEXT UNIT.
3472
3473 023770 000764      BR      504110
3474 023772      504120:
3475
3476 023772      EXIT      CLN
3477
3478      .EVEN
3479 023776      ENDCLN
3480
3481      L10014:
3482
3483      TRAP      C$EXIT
3484      .WORD      L10014-.
3485
3486      TRAP      C$CLEAN
3487
3488
3489 024000      .SBTTL  DROP UNIT SECTION
3490 024000
3491
3491 024000 010005      ;**
3492 024002 006305      ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3493 024004 012765 177774 002604      ; TO NO LONGER BE TESTED. THAT CODE SHALL BE EXECUTED WHEN DODU
3494 024012 016500 002534      ; MACRO IS CALLED WHILE IDU FLAG IS NOT SET BY OPERATOR
3495      ;--
3496
3497      BGNDU
3498      L$DU::
3499
3500 024000 010005      MOV      R0,R5      ;R5 = LOGICAL DEVICE NUMBER X 2.
3501 024002 006305      ASL      R5
3502 024004 012765 177774 002604      MOV      #NINUSE,DEVTBL(R5)      ;SET NOT IN USE FLAG FOR THE DEVICE.
3503 024012 016500 002534      CLRVEC TSVCT(R5)      ;RELEASE THE INTERRUPT VECTOR.
3504      MOV      TSVCT(R5),R0

```

MISCELLANEOUS SECTIONS MACRO M1113 30-NOV-83 10:17
DROP UNIT SECTION

SEQ 114

```

3495 024016 104436
024020 016746 173364
024024 012746 005065
024030 012746 000002
024034 010600
024036 104417
024040 062706 000006
3496 024044
024044 000167
024046 000000

3497
3498
3499 024050
024050
024050 104453

3500
3501
3502
3503
3504
3505
3506
3507
3508
3509 024052
024052
3510 024052 010005
3511 024054 006305
3512 024056 010065 002604
3513 024062
024062 104442
3514 024064 011065 002514
3515 024070 012065 002524
3516 024074 062765 000002 002524
3517 024102 011065 002534
3518 024106 011065 003532
3519 024112 011067 157416
3520 024116
024116 012746 000340
024122 016546 002554
024126 016546 002534
024132 012746 000003
024136 104437
024140 062706 000010

3521
3522 024144 005065 003472
3523
3524 024150
024150 000167
024152 000000

3525
3526
3527
3528 024154
024154
024154 104452

        PRINTF  #DROPM,DROPN          ;PRINT DROP DEVICE MESSAGE
        EXIT    DU
        .EVEN
        ENDDU
L10015:
        .SBTTL  ADD UNIT SECTION
        ;++
        ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
        ; TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
        ; "EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
        ;--
        BGNAU
L$AU::
        MOV     R0,R5          ;R5 = LOGICAL DEVICE NUMBER X 2.
        ASL     R5
        MOV     R0,DEVTBL(R5)  ;STORE UNIT # IN DEVICE TABLE.
        GPHARD  R0,R0          ;GET HARDWARE P TABLE FROM SUPER.
                                TRAP    C$GPHRD
        MOV     (R0),TSDB(R5)  ;SAVE TSDB ADDRESS.
        MOV     (R0),TSSR(R5)  ;SAVE TSSR ADDRESS.
        ADD     #2,TSSR(R5)
        MOV     (R0),TSVCT(R5) ;SAVE INTERRUPT VECTOR ADDRESS.
        MOV     (R0),TSUNT(R5) ;SAVE NUMBER OF DRIVE
        MOV     (R0),TSNP      ;SAVE FOR PRINT OUT'S
        SETVEC  TSVCT(R5),TS5INT(R5),#INTPRI
                                MOV     #INTPRI,-(SP)
                                MOV     TS5INT(R5),-(SP)
                                MOV     TSVCT(R5),-(SP)
                                MOV     #3,-(SP)
                                TRAP    C$SVEC
                                ADD     #10,SP
                                ;SET UP INTERRUPT PROCESSING CONDITIONS.
                                ;CLEAR INTERRUPT FLAGS.
        CLR     INTFLG(R5)
        EXIT    AU
        .EVEN
        ENDAU
L10016:
        .WORD   J$JMP
        .WORD   L10016-2-.
                                TRAP    C$AU

```

```

3529
3530
3531
3532 .TITLE HARDWARE TESTS
3533
3534 .SBTTL TEST 1: BASIC FUNCTIONS.
3535
3536 ***
3537 ; TEST TO EXECUTE ALL TS05 FUNCTIONS.
3538 ;--
3539
3540 024156 BGNMOD
3541
3542 024156 BGNTST
3543 024156 T1::
3544 024156 105067 157333 CLR8 RANDOM ;CLR THE RANDOM OPERATIONS FLAG.
3545 024162 105067 157326 CLR8 EXPBOT ;CLR EXPECT BOT FLAG.
3546
3547 024166 BGNSUB ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
3548 024166 104402 T1.1: TRAP C#BSUB
3549 024170 004767 172706 JSR PC,FIRSTU ;FIND THE FIRST UNIT.
3550 024174 004767 162672 JSR PC,SOFINIT ;INIT DEVICE
3551 024200 103404 BCS 11#
3552 024202 104455 ERRDF 2,NSSRM,STAERM ;REPORT TS05 NOT READY
3553 024204 000002 TRAP C#ERRDF
3554 024206 004536 .WORD 2
3555 024210 006120 .WORD NSSRM
3556 024212 004767 163250 11#: JSR PC,MSET ;GO DO SETUP'S
3557 024216 012702 025074 MOV #BFSEQ0,R2 ;ADR OF CMD SEQ.
3558 024222 004767 000622 JSR PC,BFSEQ ;SET UP CMD SEQ.
3559 024226 004767 163774 JSR PC,EXALL ;EXECUTE CMD SEQ ON ALL DEVICES.
3560 024232 004767 172644 JSR PC,FIRSTU ;FIND THE FIRST UNIT.
3561 024236 026527 002604 177777 50413#: ;WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE MORE DEVICES:
3562 024244 001451 BEQ 50414#
3563 024246 016502 002544 MOV MSGPKA(R5),R2 ;GET MSG PACKET ADR.
3564 024252 062702 000012 ADD #12,R2 ;LET R2 := R2 + #12 ;GET XSTAT2 ADR.
3565 024256 011265 002564 MOV (R2),TS5CL(R5) ;STORE CODE LEVEL FROM DTR BYTE.
3566 024262 042765 177700 002564 BIC #177700,TS5CL(R5)
3567 024270 011265 002574 MOV (R2),TS5SW(R5) ;STORE SWITCH SETTINGS
3568 024274 042765 177477 002574 BIC #177477,TS5SW(R5)
3569 024302 PRINTF #CODELM,DEVTBL(R5),TS5CL(R5)
3570 024332 PRINTF #SWSET,DEVTBL(R5),TS5SW(R5) ;PRINT THE TS05 MICROCODE LEVEL.
3571 024332 016546 002574 MOV TS5SW(R5),-(SP)

```

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 116

024336	016546	002604			MOV	DEVTBL(R5), -(SP)
024342	012746	004231			MOV	#SWSET, -(SP)
024346	012746	000003			MOV	#3, -(SP)
024352	010600				MOV	SP, R0
024354	104417				TRAP	C#PNTF
024356	062706	000010			ADD	#10, SP
3571						
3572	024362		504154:			;PRINT THE TS05 SWITCH SETTINGS.
3573	024362	004767	172562	JSR	PC, NEXTU	;FIND NEXT UNIT.
3574						
3575	024366	000723		BR	504134	
3576	024370		504144:			
3577						
3578	024370			ENDSUB		
	024370		L10020:			
	024370	104403				TRAP C#ESUB
3579						
3580	024372			BGNSUB		;SUBTEST 2 - REWIND.
	024372		T1.2:			
	024372	104402				TRAP C#BSUB
3581						
3582	024374	012702	025146	MOV	#BFSEQ1, R2	;ADR OF CMD SEQ.
3583	024400	004767	000444	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3584	024404	004767	163616	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3585	024410	105067	157112	CLRB	STAFLG	;CLEAR START FLAG
3586	024414			ENDSUB		
	024414		L10021:			
	024414	104403				TRAP C#ESUB
3587						
3588	024416			BGNSUB		;SUBTEST 3 - WRITE/VERIFY.
	024416		T1.3:			
	024416	104402				TRAP C#BSUB
3589						
3590	024420	012702	025160	MOV	#BFSEQ2, R2	;ADR OF CMD SEQ.
3591	024424	004767	000420	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3592	024430	004767	163572	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3593	024434			ENDSUB		
	024434		L10022:			
	024434	104403				TRAP C#ESUB
3594						
3595	024436			BGNSUB		;SUBTEST 4 - WRITE TAPE MARK, ERASE.
	024436		T1.4:			
	024436	104402				TRAP C#BSUB
3596						
3597	024440	012702	025252	MOV	#BFSEQ3, R2	;ADR OF CMD SEQ.
3598	024444	004767	000400	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3599	024450	004767	163552	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3600	024454			ENDSUB		
	024454		L10023:			
	024454	104403				TRAP C#ESUB
3601						
3602	024456			BGNSUB		;SUBTEST 5 - SPACE FILES.
	024456		T1.5:			
	024456	104402				TRAP C#BSUB
3603						
3604	024460	012702	025324	MOV	#BFSEQ4, R2	;ADR OF CMD SEQ.
3605	024464	004767	000360	JSR	PC, BFSEQ	;SET UP CMD SEQ.

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 117

3606	024470	004767	163532		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3607	024474			L10024:	ENDSUB			
	024474							
	024474	104403						TRAP C#ESUB
3608								
3609	024476			T1.6:	BGNSUB			;SUBTEST 6 - SPACE RECORDS.
	024476							
	024476	104402						TRAP C#BSUB
3610								
3611	024500	012702	025366		MOV	#BFSEQ5,R2		;ADR OF CMD SEQ.
3612	024504	004767	000340		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3613	024510	004767	163512		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3614	024514			L10025:	ENDSUB			
	024514							
	024514	104403						TRAP C#ESUB
3615								
3616	024516			T1.7:	BGNSUB			;SUBTEST 7 - WRITE RETRY.
	024516							
	024516	104402						TRAP C#BSUB
3617								
3618	024520	012702	025440		MOV	#BFSEQ6,R2		;ADR OF CMD SEQ.
3619	024524	004767	000320		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3620	024530	004767	163472		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3621	024534			L10026:	ENDSUB			
	024534							
	024534	104403						TRAP C#ESUB
3622								
3623	024536			T1.8:	BGNSUB			;SUBTEST 8 - READ REV RETRY.
	024536							
	024536	104402						TRAP C#BSUB
3624								
3625	024540	012702	025512		MOV	#BFSEQ7,R2		;ADR OF CMD SEQ.
3626	024544	004767	000300		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3627	024550	004767	163452		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3628	024554			L10027:	ENDSUB			
	024554							
	024554	104403						TRAP C#ESUB
3629								
3630	024556			T1.9:	BGNSUB			;SUBTEST 9 - READ FWD RETRY.
	024556							
	024556	104402						TRAP C#BSUB
3631								
3632	024560	012702	025544		MOV	#BFSEQ8,R2		;ADR OF CMD SEQ.
3633	024564	004767	000260		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3634	024570	004767	163432		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3635	024574			L10030:	ENDSUB			
	024574							
	024574	104403						TRAP C#ESUB
3636								
3637	024576			T1.10:	BGNSUB			;SUBTEST 10- CLEAN.
	024576							
	024576	104402						TRAP C#BSUB
3638								
3639	024600	012702	025576		MOV	#BFSEQ9,R2		;ADR OF CMD SEQ.
3640	024604	004767	000240		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3641	024610	004767	163412		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3642	024614				ENDSUB			

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 118

```

024614          L10031:
024614 104403          TRAP      CIESUB
3643
3644 024616          BGNSUB          ;SUBTEST 11 - WTV SWAPPED DATA BYTES.
024616          T1.11:          ;ADR OF CMD SEQ.
024616 104402          ;SET UP CMD SEQ.
3645 024620 012702 025620          MOV      #BFSE10,R2          ;WRITE/VERIFY RECORDS 1 AND 2.
3646 024624 004767 000220          JSR      PC,BFSEQ          ;ENABLE BYTE SWAPPING.
3647 024630 004767 163372          JSR      PC,EXALL          ;WRITE/VERIFY RECORDS 3 AND 4.
3648 024634 112767 000001 156656          MOVB   #1,SMBFLG          ;DISABLE BYTE SWAPPING.
3649 024642 004767 163360          JSR      PC,EXALL
3650 024646 105067 156646          CLRB    SMBFLG
3651 024652          ENDSUB
024652          L10032:
024652 104403          TRAP      CIESUB
3652 024654 016702 156526          MOV      DATAW,R2          ;INIT WRITE BUFFER POINTER.
3653 024660 062702 000012          ADD      #10.,R2
3654 024664          504161: ;WHILE R2 NE DATAW DO ;UNTIL 10 BYTES HAVE BEEN SWAPPED.
3655 024664 020267 156516          CMP      R2,DATAW
3656 024670 001402          BEQ      504171
3657 024672 000342          SWAB   -(R2)          ;SWAP DATA BYTES IN WRITE BUFFER.
3658
3659 024674 000773          BR      504161
3660 024676          504171:
3661 024676 105267 156621          INCB    T1SMB          ;SET T1 SWAP BYTES FLAG FOR "CKDATA" SUBR
3662
3663 024702          BGNSUB          ;SUBTEST 12 - READ SWAPPED DATA BYTES.
024702          T1.12:          ;CHD IS READ REV.
024702 104402          TRAP      CIESUB
3664 024704 012767 104401 156506          MOV      #ORDR,CHDWRD          ;VERIFY ODD LENGTH SWAP (RECORD 4).
3665 024712 004767 171234          JSR      PC,VFEXC          ;CHANGE BYTE COUNT TO 10.
3666 024716 012767 000012 155412          MOV      #12,CHDPKT+CP.CNT          ;VERIFY EVEN LENGTH SWAP (RECORD 3).
3667 024724 004767 171222          JSR      PC,VFEXC          ;ENABLE BYTE SWAPPING.
3668 024730 112767 000001 156562          MOVB   #1,SMBFLG          ;CHANGE BYTE COUNT TO 9.
3669 024736 012767 000011 155372          MOV      #11,CHDPKT+CP.CNT          ;VERIFY ODD LENGTH SWAP (RECORD 2).
3670 024744 004767 171202          JSR      PC,VFEXC          ;CHANGE BYTE COUNT TO 10.
3671 024750 012767 000012 155360          MOV      #12,CHDPKT+CP.CNT          ;VERIFY EVEN LENGTH SWAP (RECORD 1).
3672 024756 004767 171170          JSR      PC,VFEXC          ;CHD IS READ FWD.
3673 024762 012767 104001 156430          MOV      #ORDF,CHDWRD          ;VERIFY EVEN LENGTH SWAP (RECORD 1).
3674 024770 004767 171156          JSR      PC,VFEXC          ;CHANGE BYTE COUNT TO 9.
3675 024774 012767 000011 155334          MOV      #11,CHDPKT+CP.CNT          ;VERIFY ODD LENGTH SWAP (RECORD 2).
3676 025002 004767 171144          JSR      PC,VFEXC          ;DISABLE BYTE SWAPPING.
3677 025006 105067 156506          CLRB    SMBFLG          ;CHANGE BYTE COUNT TO 10.
3678 025012 012767 000012 155316          MOV      #12,CHDPKT+CP.CNT          ;VERIFY EVEN LENGTH SWAP (RECORD 3).
3679 025020 004767 171126          JSR      PC,VFEXC          ;CHANGE BYTE COUNT TO 9.
3680 025024 012767 000011 155304          MOV      #11,CHDPKT+CP.CNT          ;VERIFY ODD LENGTH SWAP (RECORD 4).
3681 025032 004767 171114          JSR      PC,VFEXC
3682
3683 025036          ENDSUB
025036          L10033:
025036 104403          TRAP      CIESUB
3684
3685 025040 105067 156457          CLRB    T1SMB          ;CLEAR T1 SWAP BYTES FLAG
3686
3687 025044          EXIT    TST
025044 104432          TRAP      CEXIT
025046 000574          .WORD    L10017-.
```

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 119

```

3688
3689
3690
3691
3692
3693
3694
3695 025050 012701 003540      BFSEQ:: MOV      #CMDSEQ,R1      ;INIT SEQ TABLE ADDRESS.
3696 025054      504201: ;WHILE (R2) NE #END DO      ;WHILE THERE ARE MORE COMMANDS:
3697 025054 021227 177777      CMP      (R2),#END
3698 025060 001402      BEQ      504211
3699 025062 012221      MOV      (R2)+,(R1)+      ;MOVE COMMANDS TO SEQ TABLE.
3700
3701 025064 000773      BR      504201
3702 025066      504211:
3703 025066 012711 177777      MOV      #END,(R1)      ;STORE END OF SEQUENCE CODE.
3704 025072 000207      RTS      PC      ;RETURN.
3705
3706
3707
3708
3709 025074 140004      BFSEQ0: .WORD      SCH      ;SET CHAR. 200.      (1)
3710 025076 000200      200
3711 025100 000001      1
3712 025102 000000      0
3713 025104 100013      DRI      ;DRIVE INIT.      (2)
3714 025106 000001      1
3715 025110 000001      1
3716 025112 000000      0
3717 025114 140004      SCH      ;SET CHAR. 20      (3)
3718 025116 000020      20
3719 025120 000001      1
3720 025122 000000      0
3721 025124 100017      GES      ;GET STATUS.      (4)
3722 025126 000001      1
3723 025130 000001      1
3724 025132 000000      0
3725 025134 140004      SCH      ;SET CHAR. 40.      (5)
3726 025136 000040      40
3727 025140 000001      1
3728 025142 000000      0
3729 025144 177777      .WORD      END
3730
3731 025146 102010      BFSEQ1:      RWD      ;REWIND TWICE.      (6)
3732 025150 000001      1
3733 025152 000002      2
3734 025154 000000      0
3735 025156 177777      .WORD      END
3736
3737 025160 104105      BFSEQ2:      WTV      ;WRITE/VERIFY PAT 1.      (7)
3738 025162 004000      DATCNT
3739 025164 000001      1
3740 025166 000001      1
3741 025170 104105      WTV      ;WTV PAT 2.      (8)
3742 025172 004000      DATCNT
3743 025174 000001      1
3744 025176 000002      2

```

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 120

3745	025200	104105	WTV	WTV PAT 3.	(9)
3746	025202	004000	DATCNT		
3747	025204	000001	1		
3748	025206	000003	3		
3749	025210	104105	WTV	WTV PAT 4.	(10)
3750	025212	004000	DATCNT		
3751	025214	000001	1		
3752	025216	000004	4		
3753	025220	104105	WTV	WTV PAT 5.	(11)
3754	025222	004000	DATCNT		
3755	025224	000001	1		
3756	025226	000005	5		
3757	025230	104105	WTV	WTV PAT 6.	(12)
3758	025232	004000	DATCNT		
3759	025234	000001	1		
3760	025236	000006	6		
3761	025240	104105	WTV	WTV PAT 0.	(13)
3762	025242	004000	DATCNT		
3763	025244	000001	1		
3764	025246	000000	0		
3765	025250	177777	END		
3766			.WORD		
3767	025252	100011	BFSEQ3: WTM	WRITE TAPE MARK.	(14)
3768	025254	000001	1		
3769	025256	000001	1		
3770	025260	000000	0		
3771	025262	104005	WRT	WRITE 10 RECORDS.	(15)
3772	025264	004000	DATCNT		
3773	025266	000010	10		
3774	025270	000001	1		
3775	025272	100411	ERS	ERASE 10 TIMES.	(16)
3776	025274	000001	1		
3777	025276	000010	10		
3778	025300	000000	0		
3779	025302	100011	WTM	WRITE TAPE MARK.	(17)
3780	025304	000001	1		
3781	025306	000001	1		
3782	025310	000000	0		
3783	025312	101011	WTR	WTM RETRY	(18)
3784	025314	000001	1		
3785	025316	000001	1		
3786	025320	000000	0		
3787	025322	177777	END		
3788			.WORD		
3789	025324	105410	BFSEQ4: SFR	SPACE 2 FILES REV.	(19)
3790	025326	000002	2		
3791	025330	000001	1		
3792	025332	000000	0		
3793	025334	105010	SFF	SPACE 2 FILES FWD.	(20)
3794	025336	000002	2		
3795	025340	000001	1		
3796	025342	000000	0		
3797	025344	105410	SFR	SPACE 2 FILES REV.	(21)
3798	025346	000001	1		
3799	025350	000002	2		
3800	025352	000000	0		
3801	025354	105010	SFF	SPACE 2 FILES FWD.	(22)

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 121

3802	025356	000001		1		
3803	025360	000002		2		
3804	025362	000000		0		
3805	025364	177777	.WORD	END		
3806						
3807	025366	102010	BFSEQ5:	RWD	;REWIND.	(23)
3808	025370	000001		1		
3809	025372	000001		1		
3810	025374	000000		0		
3811	025376	104010		SRF	;SPACE 7 RECORDS FWD.	(24)
3812	025400	000007		7		
3813	025402	000001		1		
3814	025404	000000		0		
3815	025406	104410		SRR	;SPACE 7 RECORDS REV.	(25)
3816	025410	000007		7		
3817	025412	000001		1		
3818	025414	000000		0		
3819	025416	104010		SRF	;SPACE 7 RECORDS FWD.	(26)
3820	025420	000001		1		
3821	025422	000007		7		
3822	025424	000000		0		
3823	025426	104410		SRR	;SPACE 7 RECORDS REV.	(27)
3824	025430	000001		1		
3825	025432	000007		7		
3826	025434	000000		0		
3827	025436	177777	.WORD	END		
3828						
3829	025440	102010	BFSEQ6:	RWD	;REWIND.	(28)
3830	025442	000001		1		
3831	025444	000001		1		
3832	025446	000000		0		
3833	025450	104005		WRT	;WRITE.	(29)
3834	025452	004000		DATCNT		
3835	025454	000001		1		
3836	025456	000001		1		
3837	025460	105005		WRR	;WRITE RETRY.	(30)
3838	025462	004000		DATCNT		
3839	025464	000001		1		
3840	025466	000001		1		
3841	025470	100011		WTH	;WRITE TAPE MARK.	
3842	025472	000001		1		
3843	025474	000001		1		
3844	025476	000000		0		
3845	025500	105410		SFR	;SPACE 1 FILE REV.	
3846	025502	000001		1		
3847	025504	000001		1		
3848	025506	000000		0		
3849	025510	177777	.WORD	END		
3850						
3851	025512	104401	BFSEQ7:	RDR	;READ REV.	(31)
3852	025514	004000		DATCNT		
3853	025516	000001		1		
3854	025520	000001		1		
3855	025522	105401		RNR	;READ NEXT REV.	(32)
3856	025524	004000		DATCNT		
3857	025526	000001		1		
3858	025530	000001		1		

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 1: BASIC FUNCTIONS.

SEQ 122

3859	025532	125401		RNF		;READ NEXT FWD.	(33)
3860	025534	004000		DATCNT			
3861	025536	000001		1			
3862	025540	000001		1			
3863	025542	177777	.WORD	END			
3864							
3865	025544	104001	BFSEQ8:	RDF		;READ FWD.	(34)
3866	025546	004000		DATCNT			
3867	025550	000001		1			
3868	025552	000001		1			
3869	025554	105001		RPF		;READ PREVIOUS FWD.	(35)
3870	025556	004000		DATCNT			
3871	025560	000001		1			
3872	025562	000001		1			
3873	025564	125001		RPR		;READ PREVIOUS REV.	(36)
3874	025566	004000		DATCNT			
3875	025570	000001		1			
3876	025572	000001		1			
3877	025574	177777	.WORD	END			
3878							
3879	025576	101012	BFSEQ9: .WORD	CLN		;CLEAN.	(37)
3880	025600	000001		1			
3881	025602	000001		1			
3882	025604	000000		0			
3883	025606	102010		RWD		;REWIND	(38)
3884	025610	000001		1			
3885	025612	000001		1			
3886	025614	000000		0			
3887	025616	177777	.WORD	END		;END OF SEQUENCE.	
3888							
3889	025620	104105	BFSEQ10:	WTV		;WRITE/VERIFY EVEN LENGTH.	(39)
3890	025622	000012		12			
3891	025624	000001		1			
3892	025626	000000		0			
3893	025630	104105		WTV		;WRITE/VERIFY ODD LENGTH.	(40)
3894	025632	000011		11			
3895	025634	000001		1			
3896	025636	000000		0			
3897	025640	177777	.WORD	END			
3898			.EVEN				
3899							
3900	025642		ENDTST				
	025642		L10017:				
	025642	104401				TRAP	C#ETST
3901							
3902			.SBTTL	TEST 2: DATA RELIABILITY.			
3903							
3904			---				
3905			;	TEST TO CHECK THE DATA RELIABILITY OF THE TS05.			
3906			---				
3907	025644		RGNTST				
	025644		T2::				
3908							
3909	025644	112767	000001	155643	MOVB	#1,RANDOM	;SET THE RANDOM OPERATIONS FLAG.
3910	025652	105067	155636		CLRB	EXPBOT	;CLEAR EXPECT BOT FLAG.
3911	025656	005067	155574		CLR	WTFMFLG	;CLEAR WRITE TAPE MARK FLAG
3912	025662	004767	171214		JSR	PC,FIRSTU	;FIND THE FIRST UNIT.

3913	025666	004767	161200		JSR	PC,SOFINIT		;INIT DEVICE
3914	025672	103404			BCS	110		
3915	025674				ERRDF	2,NSSRM,STAERM		;REPORT TS05 NOT READY
	025674	104455						TRAP C1ERDF
	025676	000002						.WORD 2
	025700	004536						.WORD NSSRM
	025702	006120						.WORD STAERM
3916								
3917	025704	004767	161556	110:	JSR	PC,MDSET		;GO DO SETUP'S
3918	025710	012702	004000		MOV	#DATCNT,R2		;SET UP THE RECORD LENGTH MASK,
3919	025714	005302			DEC	R2		
3920	025716	010267	155506		MOV	R2,LENMSK		;ALLOW MAXIMUM BUFFER.
3921	025722	005167	155502		COM	LENMSK		
3922	025726	004767	162230		JSR	PC,SETCH		;CMD 1 = SET CHARACTERISTIC.
3923	025732	105767	155570		TSTB	STAFLG ;IFB STAFLG NE #0 THEN		;IF STARTING THEN:
3924	025736	001417			BEQ	504240		
3925	025740	004767	162242		JSR	PC,SETRW		;CMD2=REWIND
3926	025744	105067	155556		CLRB	STAFLG ;LET STAFLG :B= #0		;CLR START FLAG.
3927								
3928	025750			504220:				
3929	025750	012721	104105		MOV	#MTV,(R1)+		
3930	025754	012721	004000		MOV	#DATCNT,(R1)+		
3931	025760	012702	177740		MOV	#RNOPSC,R2		
3932	025764	005102			COM	R2		
3933	025766	010221			MOV	R2,(R1)+		
3934	025770	012721	000007		MOV	#RANP,(R1)+		
3935								
3936	025774			504230:	BREAK			; DO A SUPVSR BREAK FIRST.
	025774	104422						TRAP C1BRK
3937								
3938	025776			504240:				;FILL SEQ TBL WITH RANDOM CMDS.
3939	025776	020127	003740		CMP	R1,#SEQEND		
3940	026002	002012			BGE	504250		
3941	026004	066767	155422	155422	ADD	RANB,RANS		;LET RANS := RANS + RANB
3942	026012	016702	155416		MOV	RANS,R2		
3943	026016	042702	177741		BIC	#177741,R2		
3944	026022	004772	026160		JSR	PC,BRANCMD(R2)		;SET UP A RANDOM CMD + BRF.
3945								
3946	026026	000763			BR	504240		
3947	026030			504250:				
3948	026030	012711	177777		MOV	#END,(R1)		;STORE END OF SEQUENCE CODE IN TABLE.
3949	026034	004767	162166		JSR	PC,EXALL		;GO EXECUTE ALL CMDS IN SEQUENCE TABLE.
3950								
3951	026040	012701	003540		MOV	#CMDSEQ,R1		;INIT CMD SEQ TBL POINTER.
3952	026044	005702			TST	R2		;REPEAT UNTIL EOT IS REACHED
3953	026046	001752			BEQ	504230		
3954	026050	105267	155450		INCB	ALLEOT		;FLAG ALL UNITS @ EOT
3955	026054	000240			NOP			
3956	026056	000240			NOP			
3957	026060	000240			NOP			
3958	026062	004767	001546		JSR	PC,TSWEOT		;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
3959								;SO THAT SHORTER READ STOP DISTANCE
3960								;SMALL POSITION HEAD IN CLEAN IRG GAP
3961								;READ REV THAT EXTRA REC TO RE-POSITION THE TAPE
3962	026066	004767	000126		JSR	PC,RANRD		;SET UP READ REV/FWD CMDS.
3963	026072	012767	177740	155444	MOV	#RNOPSC,CMDSEQ+4		;# OF RECORDS FOR READ REV.
3964	026100	005167	155440		COM	CMDSEQ+4		

```

3965 026104 016767 155434 155442      MOV      CMDSEQ,4,CMDSEQ+14      ;# OF RECORDS FOR READ FORWARD.
3966 026112 012711 177777              MOV      #END,(R1)          ;STORE END OF SEQUENCE CODE IN SEQ TABLE.
3967 026116 004767 162104              JSR      PC,EXALL          ;GO EXECUTE READ REV/FWD OF LAST N RECORDS.
3968 026122 105067 155376              CLR      ALLEOT          ;CLEAR ALL UNITS & EOT FLAG
3969 026126 112767 000001 155363      MOV      #1,RPTFLG        ;REQUEST PERFORMANCE REPORT DURING REWIND.
3970 026134 012701 003540              MOV      #CMDSEQ,R1      ;INIT SEQ TBL POINTER,
3971 026140 004767 162042              JSR      PC,SETRW        ;STORE REWIND IN SEQ TBL,
3972 026144 012711 177777              MOV      #END,(R1)      ;STORE END IN SEQ TBL,
3973 026150 004767 162052              JSR      PC,EXALL        ;EXECUTE REWIND CMD ON ALL UNITS
3974
3975 026154              EXIT      TST
      026154 104432
      026156 000320
                                     TRAP      C#EXIT
                                     .WORD      L10034-.

3976
3977
3978
3979
3980 026160 026332      RANCMD: RANWR      ;WRITE
3981 026162 026356      RANWR      ;WRITE.
3982 026164 026332      RANWR      ;WRITE.
3983 026166 026332      RANWR      ;WRITE.
3984 026170 026332      RANWR      ;WRITE.
3985 026172 026332      RANWR      ;WRITE.
3986 026174 026332      RANWR      ;WRITE.
3987 026176 026332      RANWR      ;WRITE.
3988 026200 026220      RANRD      ;READ.
3989 026202 026220      RANRD      ;READ.
3990 026204 026220      RANRD      ;READ.
3991 026206 026220      RANRD      ;READ.
3992 026210 026220      RANRD      ;READ.
3993 026212 026220      RANRD      ;READ.
3994 026214 026220      RANRD      ;READ.
3995 026216 026220      RANRD      ;READ.
3996
3997
3998
3999
4000
4001
4002
4003
4004 026220 005767 155232      RANRD:: TST      WTMFLG      ;WAS LAST CMD A WRITE?
4005 026224 001406              BEQ      1#              ;NO,GO AHEAD
4006 026226 004767 000136      JSR      PC,RAWTH      ;YES PUT DOWN TAPE MARK
4007 026232 004767 000160      JSR      PC,RASFR      ;AND SPACE FILE REV
4008 026236 005067 155214      CLR      WTMFLG      ;THEN CLEAR THE FLAG
4009 026242 020127 003740      1#:  CMP      R1,#SEQEND
4010 026246 002030              BGE      2#
4011 026250 012721 104401      MOV      #RDR,(R1)+      ;STORE READ REV CMD.
4012 026254 012721 004000      MOV      #DATCNT,(R1)+  ;SET BR# TO MAX FOR READ RANDOM LENGTH;
4013 026260 066767 155150 155144      ADD      RANS,RANB      ;LET RANB := RANB + RANS
4014 026266 016702 155140      MOV      RANB,R2        ;LET R2 := RANB CLR.BY #RNOPSC
4015 026272 042702 177740      BIC      #RNOPSC,R2
4016 026276 010221              MOV      R2,(R1)+      ;SET RANDOM # OF OPERATIONS.
4017 026300 012721 000007      MOV      #RANP,(R1)+    ;RANDOM PATTERN.
4018 026304 020127 003740      CMP      R1,#SEQEND
4019 026310 002007              BGE      2#

```

```

4020 026312 012721 104001      MOV    #RDF,(R1)+    ;STORE READ FWD CMD.
4021 026316 012721 004000      MOV    #DATCNT,(R1)+  ;SET BR# TO MAX TO READ RANDOM LENGTHS.
4022 026322 010221              MOV    R2,(R1)+    ;SET RANDOM # OF OPERATIONS.
4023 026324 012721 000007      MOV    #RANP,(R1)+  ;RANDOM PATTERN.
4024 026330 000207      2#:    RTS PC
4025
4026      ;          SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE,
4027      ;          THEN A WRITE TAPE MARK AND SPACE FILE REVERSE.
4028      ;
4029      ;          INPUTS:
4030      ;          OUTPUTS:
4031      ;          REGISTERS:
4032      ;          CALLS:
4033
4034 026332 012721 104005      RANWR:: MOV    #WRT,(R1)+    ;STORE WRITE CMD.
4035 026336 004767 000102      JSR PC,RANW    ;STORE BR# , # OF OPERATIONS, PATTERN.
4036 026342 005767 155110      TST     WTMFLG    ;LAST CMD A WRT?
4037 026346 001002              BNE     1#        ;YES,RETURN
4038 026350 005267 155102      INC     WTMFLG    ;NO,SET THE FLAG
4039 026354 000207      1#:    RTS PC
4040
4041      ;          SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
4042      ;          INPUTS:
4043      ;          OUTPUTS:
4044      ;          REGISTERS:
4045      ;          CALLS:
4046
4047
4048 026356 012721 104105      RANWV:: MOV    #WTV,(R1)+    ;STORE WRITE/VERIFY CMD.
4049 026362 004767 000056      JSR PC,RANW    ;STORE BR# , # OF OPERATIONS, PATTERN.
4050 026366 000207      RTS     PC
4051
4052      ;          SUBROUTINE TO SET UP A WRITE TAPE MARK IN THE SEQUENCE TABLE.
4053      ;          INPUTS:
4054      ;          OUTPUTS:
4055      ;          REGISTERS:
4056      ;          CALLS:
4057
4058
4059 026370 020127 003740      RAWTH:: CMP     R1,#SEQEND
4060 026374 002007              BGE     1#
4061 026376 012721 100011      MOV     #WTH,(R1)+    ;STORE WRITE TAPE MARK CMD.
4062 026402 012721 000001      MOV     #1,(R1)+    ;BR#
4063 026406 012721 000001      MOV     #1,(R1)+    ;# OF OPERATIONS
4064 026412 005721              TST     (R1)+    ;SKIP PATTERNS
4065 026414 000207      1#:    RTS PC
4066
4067      ;          SUBROUTINE TO SET UP A SPACE FILE REVERSE IN THE SEQUENCE TABLE.
4068      ;          INPUTS:
4069      ;          OUTPUTS:
4070      ;          REGISTERS:
4071      ;          CALLS:
4072
4073 026416 020127 003740      RASFR:: CMP     R1,#SEQEND
4074 026422 002007              BGE     1#
4075 026424 012721 105410      MOV     #SFR,(R1)+    ;STORE SPACE FILE REVERSE
4076 026430 012721 000001      MOV     #1,(R1)+    ;BR#

```

```

4077 026434 012721 000001      MOV    #1,(R1)+      ;# OF OPERATIONS
4078 026440 005721              TST    (R1)+      ;SKIP PATTERNS
4079 026442 000207      11:    RTS    PC
4080
4081
4082      ;
4083      ; SUBROUTINE TO STORE BRf, # OF OPERATIONS, PATTERN IN COMMAND
4084      ; SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
4085      ; INPUTS:
4086      ; OUTPUTS:
4087      ; REGISTERS:      R2
4088      ; CALLS:
4089 026444 012721 004000      RANW:: MOV    #DATCNT,(R1)+      ;SET BRf TO MAX FOR PATTERN GENERATION.
4090                                ;RANDOM BRf WILL BE GENERATED FOR EACH RECORD.
4091 026450 066767 154760 154754  ADD    RANS,RANB      ;LET RANB := RANB + RANS
4092 026456 016702 154750      MOV    RANB,R2      ;LET R2 := RANB CLR.BY #RNOPSC
4093 026462 042702 177740      BIC    #RNOPSC,R2
4094 026466 010221              MOV    R2,(R1)+      ;SET RANDOM # OF OPERATIONS.
4095 026470 012721 000007      MOV    #RANP,(R1)+      ;RANDOM PATTERN.
4096 026474 000207      RTS    PC      ;RETURN.
4097
4098      .EVEN
4099
4100 026476              ENDTST
4101 026476 104401      L10034:
4102                                TRAP    C#ETST
4103
4104      .SBTTL TEST 3: WRITE COMPATABILITY/WRITE UTILITY.
4105      ;
4106      ; TEST TO WRITE RECORDS FROM BOT TO EOT.
4107      ;
4108 026500              BGNTST
4109 026500      T3::
4110 026500 112767 000001 155007  MOVB    #1,RANDOM      ;SET THE RANDOM OPERATIONS FLAG.
4111 026506 105067 155002      CLRB    EXPBOT ;LET EXPBOT :B= #0      ;CLEAR EXPECT BOT FLAG.
4112
4113 026512 004767 170364      JSR     PC,FIRSTU      ;FIND THE FIRST UNIT.
4114 026516 004767 160350      JSR     PC,SOFINIT      ;INIT DEVICE
4115 026522 103404              BCS     11#
4116 026524              ERROF    2,NSSRM,STAERM      ;REPORT TS05 NOT READY
4117 026524 104455              TRAP    C#ERDF
4118 026526 000002              .WORD    2
4119 026530 004536              .WORD    NSSRM
4120 026532 006120              .WORD    STAERM
4121
4122 026534 004767 160726      111:  JSR     PC,M0SET      ;GO DO SETUP'S
4123 026540 012702 004000      MOV     #DATCNT,R2      ;SET UP THE RECORD LENGTH MASK.
4124 026544 005302      DEC     R2
4125 026546 010267 154656      MOV     R2,LENMSK      ;ALLOW MAXIMUM BUFFER.
4126 026552 005167 154652      COM     LENMSK
4127 026556 004767 161400      JSR     PC,SETCH      ;CMD 1 = SET CHARACTERISTIC.
4128 026562 004767 161420      JSR     PC,SETRW      ;CMD2=REWIND
4129 026566 105067 154734      CLRB    STAFLG ;LET STAFLG :B= #0      ;CLEAR START FLAG
4130 026572              504261: BREAK      ; DO A SUPVSR BREAK FIRST.

```

Line	Address	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418	Op419	Op420	Op421	Op422	Op423	Op424	Op425	Op426	Op427	Op428	Op429	Op430	Op431	Op432	Op433	Op434	Op435	Op436	Op437	Op438	Op439	Op440	Op441	Op442	Op443	Op444	Op445	Op446	Op447	Op448	Op449	Op450	Op451	Op452	Op453	Op454	Op455	Op456	Op457	Op458	Op459	Op460	Op461	Op462	Op463	Op464	Op46
------	---------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------

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

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 129

4222	027134	0127 2	002222	MOV	#CMD0,R2	;R2 POINTS TO CMD2 IN SOFT P TABLE.
4223	027140	004767	000446	JSR	PC,PTCMD5	;MOVE CMD 2 FROM P TBL TO SEQ TBL.
4224	027144	004767	000442	JSR	PC,PTCMD5	;MOVE CMD 3 FROM P TBL TO SEQ TBL.
4225	027150	004767	000436	JSR	PC,PTCMD5	;MOVE CMD 4 FROM P TBL TO SEQ TBL.
4226	027154	004767	000432	JSR	PC,PTCMD5	;MOVE CMD 5 FROM P TBL TO SEQ TBL.
4227	027160	004767	000426	JSR	PC,PTCMD5	;MOVE CMD 6 FROM P TBL TO SEQ TBL.
4228	027164	004767	000422	JSR	PC,PTCMD5	;MOVE CMD 7 FROM P TBL TO SEQ TBL.
4229	027170	004767	000416	JSR	PC,PTCMD5	;MOVE END CMD FROM P TBL TO SEQ TBL.
4230	027174	005067	154242	CLR	JLOOP	;CLEAR JMP CMD LOOP COUNT.
4231	027200	105067	154322	CLRB	STAFLG	;CLEAR START FLAG
4232	027204	012701	003540	MOV	#CMDSEQ,R1	;INIT SEQUENCE TABLE POINTER.
4233	027210			34:	WHILE (R1) NE #END DO	;WHILE THERE ARE CMDS LEFT IN SEQUENCE TBL:
4234	027210			504314:		
4235	027210	021127	177777	CMP	(R1),#END	
4236	027214	001574		BEG	504324	
4237	027216	022711	000040	CMP	#JMP.C,(R1)	;IS THIS A JUMP CMD?
4238	027222	001024		BNE	64	;BR IF NOT.
4239	027224	062701	000002	ADD	#2,R1	;POINT TO BR.
4240	027230	012167	154240	MOV	(R1)+,JLOC	;SAVE BR (LOCATION).
4241	027234	022167	154202	CMP	(R1)+,JLOOP	;HAS LOOP COUNT BE SATISFIED?
4242	027240	001003		BNE	14	;IF NOT, JMP AGAIN.
4243	027242	062701	000002	ADD	#2,R1	;IF SO, ADJUST SEQ POINTER
4244	027246	000760		BR	34	;AND GO TO NEXT COMMAND.
4245	027250	005267	154166	14:	INC	JLOOP
4246	027254	012701	003540	MOV	#CMDSEQ,R1	;UPDATE THE LOOP COUNT.
4247	027260	005367	154160	24:	DEC	JLOC
4248	027264	001751		BEG	34	;INIT CMD SEQ TABLE POINTER.
4249	027266	062701	000010	ADD	#10,R1	;DECR LOCATION COUNTER.
4250	027272	000772		BR	24	;IF THIS IS THE RIGHT LOCATION TO JMP TO, GO SET
4251						;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
4252	027274	022711	000020	64:	CMP	#DLY.C,(R1)
4253	027300	001026		BNE	44	;DO IT AGAIN.
4254	027302	062701	000004	ADD	#4,R1	;DELAY?
4255	027306	011167	154126	MOV	(R1),TIME2	;BR IF NOT.
4256	027312			74:	DELAY	1
	027312	012727	000001			;R1 = LOCATION OF N COUNT.
	027316	000000				;SAVE N COUNT.
	027320	016727	152572			;GO TO SUPER-WAIT 1 MSEC.
	027324	000000				
	027326	005367	177772			MOV
	027332	001375				.WORD
	027334	005367	177756			MOV
	027340	001367				.WORD
4257	027342	005367	154072			DEC
4258	027346	001361				BNE
4259	027350	062701	000004			DEC
4260	027354	000715				BNE
4261	027356	004767	161610			DEC
4262	027362					BNE
4263	027362	026767	154024			BNE
4264	027370	002103				BNE
4265	027372	004767	161466			BNE
4266	027376	004767	161122			BNE
4267	027402	026727	154012			BNE
4268	027410	001002				BNE
4269	027412	004767	167774			BNE
4270						BNE

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 130

4271 027416				50435:		
4272 027416	004767	170056			JSR PC,CKHAE	;CHECK HALT AFTER EACH CMD FLAG.
4273 027422	012702	000001			MOV #1,R2	;SET ALL UNITS AT BOT/EOT.
4274 027426	004767	167450			JSR PC,FIRSTU	;FIND FIRST UNIT.
4275 027432				50436:	WHILE DEVTBL(R5) NE #END DO	;WHILE THERE ARE MORE UNITS:
4276 027432	026527	002604	177777		DEVTBL(R5),#END	
4277 027440	001426				50437:	
4278 027442	032767	000400	153750		#MOD.CO,CMDWRD	;IF CMD IS REVERSE THEN:
4279 027450	001406				50440:	
4280 027452	032765	000002	003502		#X0.BOT,EOTFLG(R5)	;IF NOT AT BOT THEN:
4281 027460	001001				50441:	
4282 027462	005002				R2	;CLEAR EOT/BOT FLAG.
4283						
4284 027464				50441:		
4285 027464	000411				BR 50442:	;ELSE IF CMD IS NOT REVERSE:
4286 027466				50440:		
4287 027466	032765	000001	003502		BIT #X0.EOT,EOTFLG(R5)	
4288 027474	001404				50443:	
4289 027476	032767	000001	153714		BEQ 50443:	
4290 027504	001001				BIT #CMD.CO,CMDWRD	
4291 027506					50444:	
4292						
4293 027506	005002				CLR R2	;IF NOT AT EOT OR NOT A MOTION CMD THEN: ;LET R2 := #0 ;CLEAR EOT/BOT FLAG.
4294						
4295 027510				50444:		
4296						
4297 027510				50442:		
4298 027510	004767	167434			JSR PC,NEXTU	;FIND NEXT UNIT
4299						
4300 027514	000746				BR 50436:	
4301 027516				50437:		
4302 027516	020227	000001			R2,#1	;IF ALL UNIT ARE AT EOT/BOT THEN:
4303 027522	001016				50445:	
4304 027524	016767	153662	153662		MOV NCNT,NCNT1	;FORCE TERMINATION OF COMMAND.
4305 027532	005267	153656			INC NCNT1	
4306 027536	105267	153762			INCB ALLEOT	;FLAG ALL UNITS AT EOT/BOT TO ALLOW VERIFY OF D
4307 027542	026727	153660	000002		CMP CMDLG,#2	;WHEN WRITING IS CURRENT COMMAND
4308 027550	001002				50446:	
4309 027552	004767	000056			JSR PC,TSWEOT	;GO WRITE/READ REV ONE RECORD BEYOND EOT
4310						
4311 027556				50446:		
4312						
4313 027556	000402				BR 50447:	
4314 027560				50445:		
4315 027560	105067	153740			CLRB ALLEOT	;WHEN NOT ALL SEOT, CLEAR FLAG
4316						
4317 027564				50447:		
4318 027564	005267	153622			INC NCNT	;UPDATE RECORD COUNT.
4319 027570	016767	153624	153626		MOV CMDWRD,PCMDWD	;SAVE PREVIOUS COMMAND WORD.
4320						
4321 027576	000671				BR 50433:	
4322 027600				50434:		
4323 027600	004767	166262			JSR PC,VFYDAT	;IF LAST CMD WAS A WRITE VERIFY, THEN GO ;VERIFY THE LAST N RECORDS OF DATA.
4324						
4325						
4326 027604	000601				BR 50431:	
4327 027606				50432:		

HARDWARE TESTS MACRO M1113 30-NOV-83 10:17
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 131

```

4328
4329 027606          EXIT    TST
      027606 104432
      027610 000140          TRAP    C#EXIT
                                .WORD  L10037-.

4330
4331      ;      SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
4332      ;      THE COMMAND SEQUENCE TABLE.
4333      ;      INPUTS:          R2 = POINTER TO SOFT "P" TABLE
4334      ;      OUTPUTS:
4335      ;      REGISTERS:      R3.
4336      ;      CALLS:
4337
4338 027612 012203      PTCMD5:;MOV    (R2)+,R3          ;R3 = COMMAND TABLE INDEX.
4339 027614 005303      DEC    R3
4340 027616 006303      ASL    R3
4341 027620 016321 003752  MOV    CMDTBL(R3),(R1)+      ;MOVE COMMAND WORD.
4342 027624 012221      MOV    (R2)+,(R1)+      ;MOVE # OF BYTES.
4343 027626 012221      MOV    (R2)+,(R1)+      ;MOVE # OF OPERATIONS.
4344 027630 012221      MOV    (R2)+,(R1)+      ;MOVE PATTERN CODE.
4345 027632 000207      RTS    PC
4346
4347      ;      SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
4348      ;      INPUTS:
4349      ;      OUTPUTS:
4350      ;      REGISTERS:
4351      ;      CALLS:      CMDAC,EXSUB,CKHAE
4352
4353 027634 000240      TSMEOT:; NOP
4354 027636 000240      NOP
4355 027640 004767 160660      JSR    PC,EXSUB          ;WRITE ONE RECORD BEYOND EOT
4356 027644 004767 167630      JSR    PC,CKHAE          ;SO THAT READ SHORTER STOP DISTANCE
4357
4358 027650 012700 000002      MOV    #2,R0          ;SHALL POSITION HEAD IN CLEAN IRG GAP
4359 027654 016767 153540 153542 11:  MOV    CMDWRD,PCMDWD      ;SET UP COUNTER FOR EOT
4360 027662 012767 104401 153530      MOV    #ORDR,CMDWRD      ;LET PCMDWD := CMDWRD ;REPOSITION TAPE
4361 027670 012767 000004 153530      MOV    #4,CMDLG          ;LET CMDWRD := #ORDR ;BEFORE EXTRA RECORD
4362 027676 016767 153516 152424      MOV    CMDWRD,CMDPKT      ;BY READING REVERSE
4363 027704 042767 004000 152416      BIC    #BRF.C,CMDPKT      ;LET CMDPKT := CMDWRD CLR.BY #BRF.C
4364 027712 016767 152412 153502      MOV    CMDPKT,CMDSAV      ;LET CMDSAV := CMDPKT ;THAT RECORD TO ALLOW
4365 027720 016767 153464 152404      MOV    DATARD,CMDPKT+CP.ADL ;NEXT COMMAND IN THE
4366 027726 004767 161132      JSR    PC,CMDAC          ;TABLE TO BE EXECUTED
4367 027732 004767 160566      JSR    PC,EXSUB
4368 027736 004767 167536      JSR    PC,CKHAE
4369 027742 005300      DEC    R0          ;FOUND EOT YET?
4370 027744 001343      BNE    11          ;NO,KEEP GOING
4371 027746 000207      RTS    PC          ;YES,RETURN
4372
4373      .EVEN
4374
4375 027750          ENDTST
      027750
      027750 104401      L10037:
                                TRAP    C#ETST

4376
4377 027752          ENDMOD
4378
4379      .TITLE PARAMETER CODING
4380

```

```

4381      .SBTTL  HARDWARE PARAMETER CODING SECTION
4382
4383 027752      BGNMOD
4384
4385      ***
4386      ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
4387      ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
4388      ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
4389      ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
4390      ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
4391      ; WITH THE OPERATOR.
4392      ;---
4393
4394 027752      BGNHRD
4395 027752      000042
4396 027754      L$HARD::
4397 027754      GPRMA  TSSADR,0,0,160010,177564,YES
4398 027754      000031
4399 027756      030012
4400 027760      160010
4401 027762      177564
4402 027764      GPRMD  TSSVCT,2,0,777,60,776,YES
4403 027764      001032
4404 027766      030027
4405 027770      000777
4406 027772      000060
4407 027774      000776
4408 027776      GPRMD  TSSUNT,4,0,1,0,1,NO
4409 027776      002022
4410 030000      030036
4411 030002      000001
4412 030004      000000
4413 030006      000001
4414 030010      EXIT HRD
4415 030010      024004
4416
4417 030012      124      123      104  TSSADR: .NLIST  BEX
4418 030027      126      105      103  TSSVCT: .ASCIZ  /TSDB ADDRESS/
4419 030036      123      105      114  TSSUNT: .ASCIZ  /VECTOR/
4420      .LIST  BEX
4421      .EVEN
4422
4423 030060      ENDRD
4424
4425 030060      L10040:
4426
4427      .SBTTL  SOFTWARE PARAMETER CODING SECTION
4428
4429      ***
4430      ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
4431      ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
4432      ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
4433      ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
4434      ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
4435      ; WITH THE OPERATOR.

```

.WORD L10040-L\$HARD/2

.WORD T\$CODE
 .WORD TSSADR
 .WORD T\$LOLIM
 .WORD T\$HILIM

.WORD T\$CODE
 .WORD TSSVCT
 .WORD 777
 .WORD T\$LOLIM
 .WORD T\$HILIM

.WORD T\$CODE
 .WORD TSSUNT
 .WORD 1
 .WORD T\$LOLIM
 .WORD T\$HILIM

.WORD T\$CODE

.EVEN

PARAMETER CODING MACRO M1113 30-NOV-83 10:17
SOFTWARE PARAMETER CODING SECTION

SEQ 133

```

4419      1--
4420
4421      030060      BGNSFT
          030060      000302
          030062      L#SOFT::
4422      030062      GPRML      CLRM,0,1,YES
          030062      000130
          030064      030666
          030066      000001
          030070      GPRML      RRVN,0,400,YES
4423      030070      000130
          030072      030705
          030074      000400
          030076      GPRML      RCVERM,2,400,YES
4424      030076      001130
          030100      031010
          030102      000400
          030104      GPRML      HAEM,2,1,YES
4425      030104      001130
          030106      030734
          030110      000001
          030112      GPRML      IRECM,6,400,YES
4426      030112      003130
          030114      031064
          030116      000400
          030120      XFERT      NEXTSP
4427      030120      004024
          030122      GPRML      BADTH,4,1,YES
4428      030122      002130
          030124      030760
          030126      000001
          030130      NEXTSP: GPRML      DINTM,6,1,YES
4429      030130      003130
          030132      031041
          030134      000001
          030136      GPRML      IREM,12,1,YES
4430      030136      005130
          030140      031131
          030142      000001
          030144      GPRML      CHGM,10,1,YES
4431      030144      004130
          030146      031105
          030150      000001
          030152      XFERT      ENDSP1
4432      030152      127044
          030154      GPRMD      CHARM,14,0,377,0,777,YES
4433      030154      006032
          030156      031162
          030160      000377
          030162      000000
          030164      000777
          030166      GPRMD      CMD2M,16,D,37,1,33,YES
4434      030166      007052
          030170      031207
          030172      000037
          030174      000001
          030176      000033

```

.WORD L10041-L#SOFT/2

.WORD T#CODE
.WORD CLRM
.WORD 1

.WORD T#CODE
.WORD RRVN
.WORD 400

.WORD T#CODE
.WORD RCVERM
.WORD 400

.WORD T#CODE
.WORD HAEM
.WORD 1

.WORD T#CODE
.WORD IRECH
.WORD 400

.WORD T#CODE

.WORD T#CODE
.WORD BADTH
.WORD 1

.WORD T#CODE
.WORD DINTM
.WORD 1

.WORD T#CODE
.WORD IREM
.WORD 1

.WORD T#CODE
.WORD CHGM
.WORD 1

.WORD T#CODE

.WORD T#CODE
.WORD CHARM
.WORD 377
.WORD T#LOLIM
.WORD T#HILIM

.WORD T#CODE
.WORD CMD2M
.WORD 37
.WORD T#LOLIM
.WORD T#HILIM

4435	030200		GPRMD	BPCRM,20,D,-1,1,DATCNT,YES		
	030200	010052			.WORD	T#CODE
	030202	031215			.WORD	BPCRM
	030204	177777			.WORD	-1
	030206	000001			.WORD	T#LOLIM
	030210	004000			.WORD	T#HILIM
4436	030212		GPRMD	NUMBM,22,D,-1,1,77777,YES		
	030212	011052			.WORD	T#CODE
	030214	031227			.WORD	NUMBM
	030216	177777			.WORD	-1
	030220	000001			.WORD	T#LOLIM
	030222	077777			.WORD	T#HILIM
4437	030224		GPRMD	PATTH,24,D,17,0,10,YES		
	030224	012052			.WORD	T#CODE
	030226	031247			.WORD	PATTH
	030230	000017			.WORD	17
	030232	000000			.WORD	T#LOLIM
	030234	000010			.WORD	T#HILIM
4438	030236		GPRMD	CMD3M,26,D,37,1,33,YES		
	030236	013052			.WORD	T#CODE
	030240	031356			.WORD	CMD3M
	030242	000037			.WORD	37
	030244	000001			.WORD	T#LOLIM
	030246	000033			.WORD	T#HILIM
4439	030250		GPRMD	BPCRM,30,D,-1,1,DATCNT,YES		
	030250	014052			.WORD	T#CODE
	030252	031215			.WORD	BPCRM
	030254	177777			.WORD	-1
	030256	000001			.WORD	T#LOLIM
	030260	004000			.WORD	T#HILIM
4440	030262		GPRMD	NUMBM,32,D,-1,1,77777,YES		
	030262	015052			.WORD	T#CODE
	030264	031227			.WORD	NUMBM
	030266	177777			.WORD	-1
	030270	000001			.WORD	T#LOLIM
	030272	077777			.WORD	T#HILIM
4441	030274		GPRMD	PATTH,34,D,17,0,10,YES		
	030274	016052			.WORD	T#CODE
	030276	031247			.WORD	PATTH
	030300	000017			.WORD	17
	030302	000000			.WORD	T#LOLIM
	030304	000010			.WORD	T#HILIM
4442	030306		GPRMD	CMD4M,36,D,37,1,33,YES		
	030306	017052			.WORD	T#CODE
	030310	031364			.WORD	CMD4M
	030312	000037			.WORD	37
	030314	000001			.WORD	T#LOLIM
	030316	000033			.WORD	T#HILIM
4443	030320		GPRMD	BPCRM,40,D,-1,1,DATCNT,YES		
	030320	020052			.WORD	T#CODE
	030322	031215			.WORD	BPCRM
	030324	177777			.WORD	-1
	030326	000001			.WORD	T#LOLIM
	030330	004000			.WORD	T#HILIM
4444	030332		GPRMD	NUMBM,42,D,-1,1,77777,YES		
	030332	021052			.WORD	T#CODE
	030334	031227			.WORD	NUMBM