

TSV 05

TSV05 DATA REL
CVTSECO

COPYRIGHT (c) 1982-84
AH-T179C-MC
FICHE 01 OF 01

APR 1985
digital
Made In USA

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

.REM

IDENTIFICATION

PRODUCT ID: AC T178C-MC

PRODUCT TITLE: CVTSECO TSV05 DATA REL

PRODUCT DATE: 4 JUN 84

MAINTAINER: CSS/PGG 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, 1984 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL
DECPDP
DECUSUNIBUS
DECTAPE

MASSBUS

48 USER DOCUMENTATION TABLE OF CONTENTS

49 GLOSSARY

50 1.0 GENERAL INFORMATION

51 1.1 PROGRAM ABSTRACT

52 1.1.1 FUNCTIONAL DESCRIPTION

53 1.1.2 STRUCTURE OF PROGRAM

54 1.1.3 MEMORY MAP

55 1.1.4 DIAGNOSTIC INFORMATION

56 1.1.4.1 SCOPE

57 1.1.4.2 ERROR RECOVERY

58 1.1.4.3 WRITE ERROR RECOVERY

59 1.1.4.3.1

60 MEDIA/OPERATIONAL

61 SELECTIVE WRITE-ERROR-RECOVERY

62 1.1.4.3.2

63 OPERATIONAL WRITE ERROR-RECOVERY

64 1.1.4.4 DIAGNOSTIC TIMING ADJUSTMENT

65 1.2 SYSTEM REQUIREMENTS

66 1.2.1 HARDWARE REQUIREMENTS

67 1.2.2 SOFTWARE REQUIREMENTS

68 1.3 RELATED DOCUMENTS AND STANDARDS

69 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

70 1.5 ASSUMPTIONS

71 2.0 OPERATING INSTRUCTIONS

72 2.1 HARDWARE PARAMETERS

73 2.2 SOFTWARE PARAMETERS

74 2.2.1 TSV05 COMMAND LIST

75 2.2.2 DATA PATTERNS

76 2.3 EXAMPLES OF SOFTWARE PARAMETER DIALOGUE

77 2.3.1 BASIC FUNCTION AND DATA RELIABILITY
78 WITH ALL ERROR REPORTING ENABLED

79 2.3.2 SCOPE LOOP SET UP IN BASIC FUNCTIONS

80 2.3.3 SCOPE LOOP SET UP IN DATA RELIABILITY

81 .PAGE

82 2.4 EXECUTION TIMES

48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104

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 (TSDBX)
6.3.3 EXTENDED STATUS REGISTER 0 (XSTAT0)
6.3.4 EXTENDED STATUS REGISTER 1 (XSTAT1)

162 6.3.5 EXTENDED STATUS REGISTER 2 (XSTAT2)
163 6.3.6 EXTENDED STATUS REGISTER 3 (XSTAT3)
164 6.3.7 EXTENDED STATUS REGISTER 4 (XSTAT4)
165

7.0 DIAGNOSTIC HISTORY

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

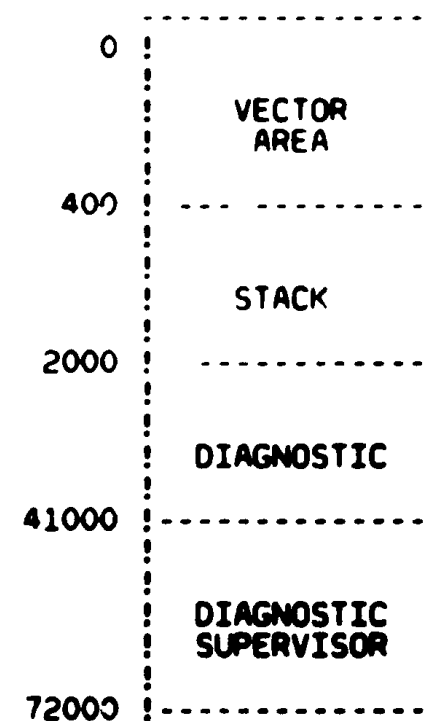
1.1.1 FUNCTIONAL DESCRIPTION

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



FREE MEMORY SPACE FOR WR/RD BFRS OR OTHER PURPOSES IS ALLOCATED BY THE SUPERVISOR ON REQUEST OR CHOSEN BY PROGRAMMER TO RESIDE BETWEEN THE DIAG AND THE SUPERVISOR.

1.1.4.1 SCOPE

THERE ARE 5 TESTS IN THIS PROGRAM:

- TEST 1 - BASIC FUNCTIONS.
TEST 2 - DATA RELIABILITY.
TEST 3 - WRITE COMPATABILITY/WRITE UTILITY.
TEST 4 - READ COMPATABILITY/READ UTILITY.
TEST 5 - RANDOM/OPERATOR SELECTED SEQUENCE UTILITY.

1.1.4.2 ERROR RECOVERY

ERROR RECOVERY IS PERFORMED ON READ, WRITE AND WRITE TAPE MARK FUNCTIONS UNLESS ERROR RECOVERY IS INHIBITED BY THE OPERATOR AT START UP TIME. THE READ FORWARD/READ REVERSE RETRY LIMIT IS 16 (8 IN THE SAME DIRECTION AND 8 IN THE OPPOSITE DIRECTION). FOR MORE DETAILED INFORMATION ON ERROR RECOVERY PROCEDURES, REFER TO SECTION 3.0 (ERROR REPORTING) OF THIS LISTING.

1.1.4.3 WRITE ERROR RECOVERY

THERE ARE 2 . SELECTABLE WRITE-ERROR-RECOVERY ALGORITHMS USED BY THIS DIAGNOSTIC:

1. MEDIA/OPERATIONAL SELECTIVE ALGORITHM
2. OPERATIONAL ALGORITHM

BY DEFAULT THE DIAGNOSTIC SELECTS THE FIRST ALGORITHM TO IDENTIFY MEDIA RELATED WRITE ERRORS FROM OPERATIONAL ONES.

TO SELECT THE SECOND ALGORITHM:

- ANSWER 'Y' TO CHANGE SW (L) ?
ANSWER 'N' TO BAD TAPE SPOT DETECTION (L) Y ?

IF ERROR RECOVERY IS INHIBITED, THE LATTER QUESTION IS NOT ASKED
AND BOTH ALGORITHMS ARE BYPASSED.

1.1.4.3.1 MEDIA/OPERATIONAL SELECTIVE WRITE-ERROR RECOVERY ALGORITHM

SCOPE

THIS ALGORITHM IDENTIFIES MEDIA RELATED WRITE ERRORS FROM

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

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

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
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 "L\$DLY". 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
452
453 1.2 SYSTEM REQUIREMENTS
454 -----
455

456
457 1.2.1 HARDWARE REQUIREMENTS
458 -----
459

460 PDP-11/23 PROCESSOR WITH 32K OR MORE OF MEMORY
461 CONSOLE DEVICE (VT52,LA36,ETC.)
462 PROGRAM LOAD DEVICE
463 TSV05/TS05
464

465
466 1.2.2 SOFTWARE REQUIREMENTS
467 -----
468

469 DIAGNOSTIC SUPERVISOR
470

471
472
473 1.3 RELATED DOCUMENTS AND STANDARDS
474 -----
475

476 DIGITAL EQUIPMENT CORPORATION DOCUMENTS:
477

- 478 1. CIQPM40 XXDP+ PROGRAMMER'S MANUAL; DOCUMENT NUMBER AC-S296A AC
479 DATE: 14 JULY 1980.
480
481 2. TSV05 TRANSPORT SUBSYSTEM USER'S GUIDE; DOCUMENT NUMBER EK-TSV05-UG-001
482 DATE: AUGUST 1982
483
484 3. TSV05 TRANSPORT SUBSYSTEM TECHNICAL MANUAL; DOCUMENT NUMBER EK-TSV05-TM-001
485 DATE: AUGUST 1982
486
487 4. TSV05 TRANSPORT SUBSYSTEM INSTALLATION MANUAL; DOCUMENT NUMBER EK-TSV05 IN-001
488 DATE: AUGUST 1982
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 (VTSA,VTSB,VTSC,VTSD)
498
499 2) DATA RELIABILITY PROGRAM:
500 A) BASIC FUNCTION TEST.
501 B) DATA RELIABILITY TEST.
502
503

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 SUCESSFULLY 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 (CH015).

COMMANDS

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

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP. MONITOR (XXDP. OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME)
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC)
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 "CIOPMAO XXDP. PROGRAMMER'S MANUAL, NUMBER
AC-S296A-AC. THE USER ENTRY IS IN QUOTES.

ROOT 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 "M" OR "H/L" FOR DETAILS
R VTSEA0
VTSEA0BJNDORS 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 FLAG 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.

TSDB ADDRESS (0) 172520 ?

VECTOR (0) 224 ?

SELECT DRIVE 0-1 (0) ?

B.

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 ?

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
801			
802			
803			
804	1	DRI	DRIVE INITIATE.
805	2	RDF	READ FORWARD.
806	3	RDR	READ REVERSE.
807	4	WRT	WRITE.
808	5	WTV	WRITE/VERIFY. IE. WRITE N RECORDS, READ REVERSE AND CHECK
809			N RECORDS OF DATA, READ FORWARD AND CHECK N RECORDS.
810	6	SRF	SPACE RECORDS FORWARD.
811	7	SRR	SPACE RECORDS REVERSE.
812	8	RNR	READ NEXT REVERSE, IE. SPACE FWD, READ REV.
813	9	RNF	READ NEXT FORWARD, IE. READ FWD, SPACE REV.
814	10	RPF	READ PREVIOUS FWD, IE. SPACE REV, READ FWD.
815	11	RPR	READ PREVIOUS REV, IE. READ REV, SPACE FWD.
816	12	WRR	WRITE RETRY.
817	13	RWD	REWIND.
818	14	MBR	MESSAGE BUFFER RELEASE.
819	15	WTM	WRITE TAPE MARK.
820	16	WTR	WRITE TAPE MARK RETRY.
821	17	SFF	SPACE FILES FORWARD.
822	18	SFR	SPACE FILES REVERSE.
823	19	GES	GET EXTENDED STATUS.
824	20	ERS	ERASE 3 INCHES OF TAPE.
825	21	UNL	UNLOAD.
826	22	CLN	CLEAN TAPE
827	23	SCH	SET DEVICE CHARACTERISTIC. WHERE BRF=200, 40, 20, 0.
828			200 = ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT)
829			40 = ENABLE ATTENTION INTERRUPTS.
830			20 = ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
831			SEE TSV05/TS05 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
832	25	JMP	JUMP TO THE NTH COMMAND IN THE COMMAND SEQUENCE
833			TABLE, WHERE N IS DEFINED IN THE BRF FIELD.
834			THE NUMBER OF JUMPS IS ENTERED IN THE # OF OPERATIONS FIELD
835	26	DLY	DELAY "N" MILLISECONDS WHERE N IS DEFINED IN
836			THE # OF OPERATIONS.
837	27	END	END OF COMMAND SEQUENCE.
838			
839			

2.2.2 DATA PATTERN LIST FOR USE IN SOFTWARE DIALOGUE.

	PATTERN #	DESCRIPTION.
840		
841		
842		
843		
844	0	INCREMENTING PATTERN. 0 - 377.
845	1	ALL "1"'S PATTERN.
846	2	ALL "0"'S PATTERN.
847	3	"1" BIT WALKING FROM R TO L IN A FIELD OF "0"'S.
848	4	"0" BIT WALKING FROM R TO L IF A FIELD OF "1"'S.
849	5	ALTERNATING "1" AND "0" BITS WITH ALTERNATE BYTES COMPLIMENTED.
850	6	ALTERNATING BYTES OF 000 AND 377.
851	7	RANDOM DATA PATTERN.
852	8	NO PATTERN GENERATION.

2.3 EXAMPLES OF SOFTWARE DIALOGUE

CHANGE HW (L) ?

#UNITS (D) ?

TSDB ADDRESS (U) 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>

911 INHIBIT RFC ERROR REPORT (L) N ? Y<CR>
 912 CHANGE CMD SEQUENCE (L) N ? N<CR>
 913 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:

922 CHANGE SW (L) ? Y<CR>
 923 CLEAR COUNTERS (L) N ? Y<CR>
 924 RESET RANDOM VARIABLES (L) N ? N<CR>
 925 PRINT RECOVERABLE ERRORS (L) N ? N<CR>
 926 HALT AFTER EACH CMD (L) N ? N<CR>
 927 INHIBIT RECOVERY (L) N ? N<CR>
 928 BAD TAPE SPOT DETECTION (L) Y ? N<CR>
 929 DISABLE INTERRUPTS (L) N ? Y<CR>
 930 INHIBIT RFC ERROR REPORT (L) N ? Y<CR>
 931 CHANGE CMD SEQUENCE (L) N ? Y<CR>
 932 CHARACTERISTICS CODE (0) 40 ? 40<CR>
 933 CMD/2 (0) 5 ? 13<CR> (REWIND)
 934 BRP COUNT (0) 2048 ? 1<CR>
 935 # OF OPERATIONS (0) 10 ? 1<CR>
 936 PATTERN (0) 7 ? 1<CR>
 937 CMD/3 (0) 5 ? 4<CR> (WRITE)
 938 BRP (0) 2048 ? 1000<CR>
 939 # OF OPERATIONS (0) 10 ? 10000<CR>
 940 PATTERN (0) 7 ? 1<CR>
 941 CMD/4 (0) 5 ? 27<CR> (END)
 942 BRP (0) 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
```

1025
1026
1027
1028
1029
1030
1031
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

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

1082
1083
1084
1085
1086
1087
1088
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

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.

1139 3.1.6 ERROR #13 - RFC NON ZERO ERROR:
1140 IF, AFTER EXECUTION, THE RESIDUAL FRAME COUNT IS NON ZERO, THE
1141 ERROR IS REPORTED AND A HARD ERROR IS LOGGED. THE PROGRAM THEN
1142 PROCEEDS NORMALLY. THE REPORTING AND LOGGING OF THESE ERRORS
1143 IS OPTIONAL.
1144
1145
1146 3.1.7 ERROR #14 - RETRY LIMIT EXCEEDED:
1147 ON A WRITE COMMAND THIS IS A FATAL DEVICE ERROR AND THE DEVICE
1148 WILL BE DROPPED FROM THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1149
1150 ON A READ COMMAND THIS ERROR IS LOGGED AS A HARD ERROR AND
1151 THE PROGRAM PROCEEDS NORMALLY.
1152
1153
1154 3.1.8 ERROR #15 - TOO MANY INTERRUPTS:
1155 IF MORE THAN ONE INTERRUPT OCCURS PER COMMAND, THIS ERROR IS REPORTED.
1156 THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1157 THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1158
1159
1160 3.1.9 ERROR #16 - CAPSTAN RUNAWAY:
1161 CAPSTAN DID NOT STOP WITHIN ACCEPTABLE WINDOW AFTER LAST
1162 COMMAND. THE PROGRAM WILL ISSUE A GET STATUS COMMAND BEFORE REPORTING
1163 THE ERROR SO THAT THE DEAD TRACK FIELD IN EXTENDED STATUS REGISTER 2
1164 WILL CONTAIN THE TACH COUNT WHEN THE TAPE STOPPED.
1165 THIS IS A FATAL DEVICE ERROR AND THE DEVICE WILL BE DROPPED FROM
1166 THE TEST CYCLE UNLESS THE IDU OPTION IS USED.
1167
1168
1169 3.1.10 ERROR #17 - DATA COMPARE ERROR:
1170 IF A DATA VALIDATION ERROR OCCURS DURING A WRITE/VERIFY COMMAND,
1171 THE PROGRAM PRINTS WHAT THE DATA SHOULD HAVE BEEN AND WHAT THE
1172 DATA WAS, AND PRINTS THE BYTE AND RECORD NUMBER THE ERROR OCCURRED
1173 ON. ONLY THE FIRST 10 BYTES IN ERROR PER RECORD ARE PRINTED.
1174 THE TOTAL # OF BYTES IN ERROR PER RECORD IS ALSO PRINTED. A
1175 HARD ERROR IS LOGGED AND THE PROGRAM PROCEEDS NORMALLY.
1176
1177
1178 3.2 ERROR HALTS
1179 -----
1180 ERROR HALTS ARE SUPPORTED PER DESCRIBED IN THE PREVIOUS SECTION
1181 WITH /FLAG:MOE. THERE ARE NO OTHER HALTS.
1182
1183 4.0 PERFORMANCE REPORT
1184 -----
1185
1186 UNIT X PASS:XXXXX RECORD:XXXXX
1187 BYTES WRITTEN XXX,XXX,XXX,XXX
1188 BYTES READ REV XXX,XXX,XXX,XXX
1189 BYTES READ FWD XXX,XXX,XXX,XXX
1190
1191
1192
1193
1194
1195

1196					
1197					
1198					
1199					
1200					
1201					
1202					
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					

	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 TSV05 FUNCTIONS.

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

- SET CHARACTERISTIC 200.
- DRIVE INITIATE.
- SET CHARACTERISTIC 20.
- GET STATUS
- SET CHARACTERISTIC 40.
- PRINT TSV05 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.

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

• 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.

1294
1295
1296
1297
1298
1299
1300
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

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 TSV05 TAPE SUBSYSTEM CONSISTS OF A TSV05 Q-BUS CONTROLLER CONNECTED TO A TSV05 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 TSV05 SUBSYSTEM TO FETCH THE WORD(S) WITHIN THE COMMAND PACKET. THE WORDS WITHIN THE COMMAND PACKET ARE:

1351
1352
1353
1354
1355
1356
1357
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

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 TSO5 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 TSOB 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 .BSYSTEM INIT.
- G. COMMAND PACKETS MUST RESIDE ON DIVIDE BY FOUR MEMORY BOUNDARIES (AS OPPOSED TO DIVIDE BY 2 OR WORD BOUNDARIES).

6.2 Q BUS INTERFACE SPECIFICATIONS

1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394

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	THK	RLS	LET	RLI	MLE	NEF	ILC	ILA	MOT	OML	IE	VCK	PED	MLK	BOT	EOT
XST1	DLT	0	COR	0	0	0	0	RBP	0	0	0	0	0	0	UNC	0
XST2	OPH	RCE	0	0	0	MCF	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 TSD8 AND SUB-SYSTEM READY (SSR) IS NOT SET. NOTE THAT THIS BIT CAUSES SPECIAL CONDITION BUT NO TERMINATION CLASS (IN FACT, THE TSV05 NEVER SEES THIS ERROR) BECAUSE ON A SYSTEM WITH NO BUGS, THIS BIT MAY COME UP ON AN ATTENTION MESSAGE. IF ATTN8 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 TSV05 NEEDS A MESSAGE BUFFER ADDRESS. THIS BIT IS CLEARED DURING THE SET CHARACTERISTICS

1511				COMMAND (IF A GOOD ADDRESS WAS GIVEN).
1512				
1513	09	A17	S	BUS ADDRESS BIT 17. A17 AND A16 (BIT 08) TRACK
1514				THE VALUES OF BITS 17 AND 16 OF THE TSBA
1515				REGISTER. LOADED FROM TSDB BITS 01-00 WHEN TSDB
1516				IS WRITTEN.
1517				
1518	08	A16	S	BUS ADDRESS BIT 16. SEE A17 (BIT 09).
1519				
1520	07	SSR	S	SUB-SYSTEM READY. WHEN SET, INDICATES THAT THE
1521				TSV05/TS05 SUBSYSTEM IS NOT BUSY AND IS READY TO
1522				ACCEPT A NEW COMMAND POINTER.
1523				
1524	06	OFL	S,1,3	OFF-LINE. WHEN SET, INDICATES THAT THE TS05 IS
1525				OFF-LINE AND UNAVAILABLE FOR ANY TAPE MOTION
1526				COMMANDS. THIS BIT CAN CAUSE A TERMINATION CLASS
1527				OF 1 (ON ATTN INTERRUPT) OR 3 (RESULTS IN NEF).
1528				
1529	05	FC1	7	FATAL TERMINATION CLASS 01. FC1 AND FCO (BIT
1530				04) ARE USED TO INDICATE THE TYPE OF FATAL
1531				ERROR WHICH HAS OCCURRED ON THE TS05. THESE
1532				BITS ARE VALID ONLY WHEN SC IS SET AND THE
1533				TERMINATION CLASS CODE BITS ARE ALL SET (111).
1534				
1535	04	FC0	7	FATAL TERMINATION CLASS 00. SEE FC1 (BIT 05).
1536				
1537	03	TC2	S	TERMINATION CLASS BIT 02. THIS BIT, ALONG WITH
1538				THE TC1 AND TCO BITS, ACT AS AN OFFSET VALUE
1539				WHENEVER AN ERROR OR EXCEPTION CONDITION OCCURS
1540				ON A COMMAND. EACH OF THE EIGHT POSSIBLE
1541				VALUES OF THIS FIELD REPRESENT A PARTICULAR
1542				CLASS OF ERRORS OR EXCEPTIONS. THE CONDITIONS
1543				IN EACH CLASS HAVE SIMILAR SIGNIFICANCE AND, AS
1544				APPLICABLE, RECOVERY PROCEDURES. THE CODE
1545				PROVIDED IN THIS FIELD IS EXPECTED TO BE
1546				UTILIZED AS AN OFFSET INTO A DISPATCH TABLE FOR
1547				HANDLING OF THE CONDITION.
1548				
1549	02	TC1	S	TERMINATION CLASS BIT 01. SEE TC2 (BIT 03).
1550				
1551	01	TC0	S	TERMINATION CLASS BIT 00. SEE TC2 (BIT 03).
1552				
1553	00	-		NOT USED. (ALWAYS A 0)

Q-BUS ADDRESS + 2 - WRITE ONLY

SUBSYSTEM INITIALIZE

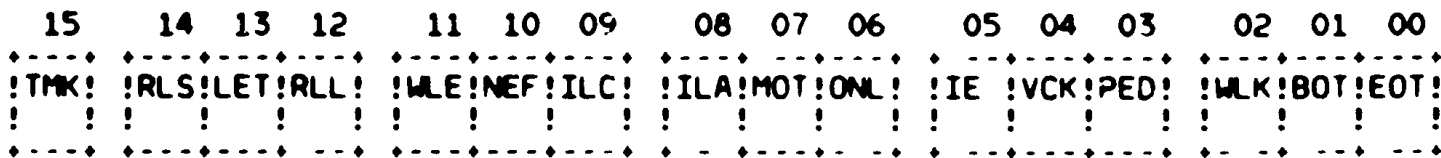
6.3.2.1 TSV05 EXTENDED DATA BUFFER REGISTER (TSDBX)



(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)



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.

1619				
1620				
1621	13	LET	2	LOGICAL END OF TAPE. SET ONLY ON THE SKIP TAPE
1622				MARKS COMMAND WHEN EITHER TWO CONTIGUOUS TAPE
1623				MARKS ARE DETECTED OR WHEN MOVING OFF OF BOT
1624				AND THE FIRST RECORD ENCOUNTERED IS A TAPE
1625				MARK. THE SETTING OF THIS BIT WILL NOT OCCUR
1626				UNLESS THIS MODE OF TERMINATION IS ENABLED
1627				THROUGH USE OF THE SET CHARACTERISTICS COMMAND.
1628				
1629	12	RLL	2	RECORD LENGTH LONG. WHEN SET, THIS BIT
1630				INDICATES THAT THE RECORD READ WAS LONGER THAN
1631				THE BYTE COUNT SPECIFIED.
1632				
1633	11	WLE	3,6	WRITE LOCK ERROR. WHEN SET, INDICATES THAT A
1634				WRITE OPERATION WAS ISSUED BUT THE MOUNTED TAPE
1635				DID NOT CONTAIN A WRITE ENABLE RING OR THE WRT
1636				LOCK SWITCH ACTIVATED DURING THE OPERATION.
1637				
1638	10	NEF	3	NON-EXECUTABLE FUNCTION. WHEN SET, INDICATES
1639				THAT THE COMMAND COULD NOT BE EXECUTED DUE TO
1640				ONE OF THE FOLLOWING CONDITIONS:
1641				- THE COMMAND SPECIFIED REVERSE TAPE
1642				DIRECTION BUT THE TAPE WAS ALREADY
1643				POSITIONED AT BOT.
1644				- THE ISSUING OF ANY MOTION COMMAND EXCEPT

1646
1647
1648
1649
1650
1651
1652
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

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.

1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731

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 TS05.
14	-	-	NOT USED. (ALWAYS A 0)
13	COR	S	CORRECTABLE DATA. CORRECTABLE DATA ERROR HAS BEEN ENCOUNTERED.
12-09			RESERVED (ALWAYS A 0)
08	RPB	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.

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)

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MICRO DIAGNOSTIC ERROR CODE								0	OPI	REV	TRF	DCK	0	0	RIB

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)

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
HSP	RCE	0	0	0	0	0	0	WRITE RETRY COUNT							

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

1822	15	HSP	S	HIGH SPEED. WHEN SET, INDICATES THAT THE TRANSPORT IS
1823				OPERATING IN HIGH SPEED MODE.(100IPS) WHEN CLEAR, THE
1824				TRANSPORT IS OPERATING IN LOW SPEED MODE.(25IPS)
1825				
1826	14	RCE	6	RETRY COUNT EXCEEDED. WHEN SET, INDICATES THAT THE CONTROLLER
1827				WAS BUFFERING WRITE DATA AND COULD NOT SUCCESSFULLY OUTPUT
1828				THE BUFFERED RECORD WITHIN THE SPECIFIED NUMBER OF RETRIES.
1829				CAUSES TAPE POSITION LOST TERMINATION.
1830				
1831	13-8			RESERVED (ALWAYS A 0)
1832				
1833	7 0	WRC	S	WRITE RECOUNT COUNT STATISTIC. THIS FIELD INDICATES, WHEN
1834				THE CONTROLLER IS BUFFERING WRITE DATA RECORDS, THE TOTAL
1835				NUMBER OF CONTROLLER INITIATED RETRIES PERFORMED IN ORDER
1836				TO WRITE THE PREVIOUS BUFFERED RECORD. THIS COUNT IS CLEARED
1837				AFTER IT IS DISPLAYED.
1838				
1839				
1840				

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.

REVISION C - JUN 1984
MINOR CHANGES FOR "ORION" CPU

1858

```

13      .TITLE PROGRAM HEADER AND TABLES
14      .SBTTL PROGRAM HEADER
43
45      .ENABL ABS,AMA
46      .      2000
48      .BGNMOD
49
50      ;**
51      ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
52      ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
53      ;
54
55      .PTR BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU
56
64
65      .HEADER CVTSE,C,0,5000,1
        L$NAME::      ;DIAGNOSTIC NAME
        .ASCII /C/
        .ASCII /V/
        .ASCII /T/
        .ASCII /S/
        .ASCII /E/
        .BYTE 0
        .BYTE 0
        .BYTE 0
        L$REV::      ;REVISION LEVEL
        .ASCII /C/
        L$DEPO::      ;0
        .ASCII /0/
        L$UNIT::      ;NUMBER OF UNITS
        .WORD 0
        L$TIME::      ;LONGEST TEST TIME
        .WORD 5000
        L$MPCP::      ;PTR. TO M.W. QUES.
        .WORD L$HARD
        L$SPCP::      ;PTR. TO S.W. QUES.
        .WORD L$SOFT
        L$MPTP::      ;PTR. TO DEF. M.W. PTABLE
        .WORD L$HW
        L$SPTP::      ;PTR. TO S.W. PTABLE
        .WORD L$SW
        L$LADP::      ;DIAG. END ADDRESS
        .WORD L$LAST
        L$STA::      ;RESERVED FOR APT STATS
        .WORD 0
        L$CO::
        .WORD 0
        L$DTYP::      ;DIAGNOSTIC TYPE
        .WORD 1
        L$APT::      ;APT EXPANSION
        .WORD 0
        L$DTP::      ;PTR. TO DISPATCH TABLE
        .WORD L$DISPATCH
        L$PRIO::      ;DIAGNOSTIC RUN PRIORITY
        .WORD 0
        L$ENVI:      ;FLAGS DESCRIBE HOW IT WAS SETUP

```

PROGRAM HEADER AND TABLE
PROGRAM HEADER

MACRO M1113 14 JUN 74 18-32

SEQ 0039

002044	000000			.WORD	0
002046		L\$EXP::	: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 FLAGS	.WORD	0
002052	007000			.WORD	0
002054	000000			.WORD	0
002056		L\$JPC::			
002056	000000			.WORD	0
002060		L\$DEVP::	: POINTER TO DEVICE TYPE LIST	.WORD	L\$DVTYP
002067	002164			.WORD	L\$DVTYP
002062		L\$REPP::	:PTR. TO REPORT CODE	.WORD	L\$RPT
002062	017570			.WORD	L\$RPT
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\$AU
002072		L\$DUT::	:PTR. TO DROP UNIT CODE	.WORD	L\$DU
002072	024000			.WORD	L\$DU
002074		L\$LUN::	:LUN FOR EXERCISERS TO FILL	.WORD	0
002074	000000			.WORD	0
002076		L\$DESP::	:POINTER TO DIAG. DESCRIPTION	.WORD	L\$DESC
002076	002136			.WORD	L\$DESC
002100		L\$LOAD::	:GENERATE SPECIAL AUTOLOAD EMT	EMT	E\$LOAD
002100	104035				
002102		L\$ETP::	:POINTER TO ERRIBL	.WORD	0
002102	000000			.WORD	0
002104		L\$ICP::	:PTR. TO INIT CODE	.WORD	L\$INIT
002104	021324			.WORD	L\$INIT
002106		L\$CCP::	:PTR. TO CLEAN-UP CODE	.WORD	L\$CLEAN
002106	023736			.WORD	L\$CLEAN
002110		L\$ACP::	:PTR. TO AUTO CODE	.WORD	L\$AUTO
002110	023314			.WORD	L\$AUTO
002112		L\$PRT::	:PTR. TO PROTECT TABLE	.WORD	L\$PROT
002112	021316			.WORD	L\$PROT
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

PROGRAM HEADER AND TABLES
DISPATCH TABLE

MACRO M1113 14 JUN 84 18:32

SEQ 0040

```

73      .SBTTL DISPATCH TABLE
74
75      ;**
76      ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
77      ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
78      ;--
79
80      DISPATCH 5
      002122 000005      .WORD 5
      002124      .WORD T1
      002126 024156      .WORD T2
      002128 025644      .WORD T3
      002130 026500      .WORD T4
      002132 026674      .WORD T5
      002134 027054
81
88
89      .SBTTL DESCRIPTIVE TEXT
90
91      ;**
92      ; 2 LINES OF TEXT PRINTED TO THE OPERATOR TO IDENTIFY THE DIAGNOSTIC AND THE DEVICE UNDER TES
93      ;--
94
95      DESCRIPT <DATA RELIABILITY TEST>
      002136      L$DESC::
      002138      104      101      124      .ASCIIZ /DATA RELIABILITY TE
96
      002141      101      040      122
      002144      105      114      111
      002147      101      102      111
      002152      114      111      124
      002155      131      040      124
      002160      105      123      124
      002163      000
      002164      DEVTYP <TSV05>      .EVEN
      002166      L$DVTYP::
      002168      124      123      126      .ASCIIZ /TSV05/
      002170      060      065      000      .EVEN

```

PROGRAM HEADER AND TABLES
DEFAULT HARDWARE P TABLE

MACRO M1113 14 JUN 84 18:32

SEQ 0041

```

98      .SBTTL  DEFAULT HARDWARE P TABLE
99
100     ;**
101     ; THE DEFAULT HARDWARE P TABLE CONTAINS DEFAULT VALUES OF
102     ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
103     ; IS IDENTICAL TO THE STRUCTURE OF THE RUN TIME P TABLE.
104     ;
105
106     002172      BGNMW  DFPTBL
107     002172      000003
108     002174
109     002174      172520      .WORD  L10000-L$HW/2
110     002176      000224      .WORD  172520      ;TSD8 ADDRESS.
111     002200      000000      .WORD  224        ;VECTOR ADDRESS.
112                                     .WORD  0        ;DRIVE #0 FOR DEFAULT
113     002202      ENDMW
114     002202      L10000:
115
116
117     .SBTTL  SOFTWARE P-TABLE
118
119     ;**
120     ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
121     ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
122     ;--
123
124     002202      BGNSW  SFPTBL
125     002202      000051
126     002204
127     002204
128
129     002204      001      CLRFLG:: .BYTE  1      ;CLEAR COUNTERS FLAG.
130     002205      000      RRANV:: .BYTE  0      ;RESET RANDOM VARIABLES EACH PASS FLAG.
131     002206      000      MAE:: .BYTE  0      ;HALT AFTER EACH COMMAND FLAG.
132     002207      000      ERCVER:: .BYTE  0      ;ENABLE RECOVERABLE ERROR PRINTS FLAG.
133     002210      001      BADTSW:: .BYTE  1      ;BAD TAPE SWITCH TO REWRITE ON SAME SPOT & DETECT BAD TAPE
134     002211      000      DINT:: .BYTE  0      ;SPARE
135     002212      000      IREC:: .BYTE  0      ;DISABLE INTERRUPTS FLAG.
136     002213      000      CHGFLG:: .BYTE  0      ;INHIBIT ERROR RECOVERY FLAG.
137     002214      000      PIRE:: .BYTE  0      ;CHANGE CMD SEQ TABLE FLAG.
138     002215      000      CHAR:: CH.EAI      ;SPARE.
139     002216      000      CMDD:: .WORD  13.      ;INHIBIT RESIDUAL FRAMECOUNT ERROR REPORT FLAG.
140     002217      000      CMDD:: .WORD  1      ;SPARE.
141     002220      000040      CMDD:: .WORD  1      ;CHARACTERISTICS CODE (DEFAULT = 40).
142     002222      000015      CMDD:: .WORD  1      ;COMMAND 2 (DEFAULT = REWIND).
143     002224      000001      CMDD:: .WORD  1      ;BYTE COUNT
144     002226      000001      CMDD:: .WORD  1      ;NUMBER OF OPERATIONS
145     002230      000007      CMDD:: .WORD  RANP      ;PATTERN
146     002232      000004      CMDD:: .WORD  4      ;COMMAND 3 (DEFAULT = WRITE)
147     002234      004000      CMDD:: .WORD  DATCNT      ;BYTE COUNT (DEFAULT = MAX BUFFER SIZE).
148     002236      076400      CMDD:: .WORD  32000.      ;NUMBER OF OPERATIONS (DEFAULT = 32000).
149     002240      000007      CMDD:: .WORD  RANP      ;PATTERN (DEFAULT = RANDOM).
150     002242      000003      CMDD:: .WORD  3      ;COMMAND 4 (DEFAULT = READ REV).

```

PROGRAM HEADER AND TABLES
SOFTWARE P TABLE

MACRO M1113 14 JUN 84 18:32

SEQ 0042

```

154 002244 004000 .WORD DATCNT
155 002246 076400 .WORD 32000.
156 002250 000007 .WORD RANP
157 002252 000002 .WORD 2
158 002254 004000 .WORD DATCNT
159 002256 076400 .WORD 32000.
160 002260 000007 .WORD RANP
161 002262 000015 .WORD 13.
162 002264 000001 .WORD 1
163 002266 000001 .WORD 1
164 002270 000007 .WORD RANP
165 002272 000033 .WORD 27.
166 002274 004000 .WORD DATCNT
167 002276 076400 .WORD 32000.
168 002300 000007 .WORD RANP
169 002302 000033 .WORD 27.
170 002304 004000 .WORD DATCNT
171 002306 076400 .WORD 32000.
172 002310 000007 .WORD RANP
173 002312 000001 TS1MD:: .WORD 1
174 002314 000000 RDBUF:: .WORD 0
175 002316 000000 WTBUF:: .WORD 0
176 002320 000000 HSSW:: .WORD 0
177 002322 000000 EXTFEA:: .WORD 0
178 002324 000000 BENBSW:: .WORD 0
179
180 002326 .ENDSW
    002326 L10001:
181
182 002326 .ENDMOD

```

```

;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

```

```

195
196
197
206
207 002326
208
209
210
211
212
213
214 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 14 JUN 84 18:32
GLOBAL EQUATES SECTION

SEQ 0044

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 MOE== 100000

; 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.

215
223
224
225
226
227
228
229
230
231
232
233

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL EQUATES SECTION

SEQ 0045

```

235
236      ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE TSSR REGISTERS.
237
238      100000      TS.SC==100000      ;SPECIAL CONDITION BIT.
239      040000      TS.UPE==40C00      ;UNIBUS PARITY ERROR
240      020000      TS.SPE==20000      ;SERIAL BUS PARITY ERROR.
241      010000      TS.RMR==10000      ;REGISTER MODIFICATION REFUSED.
242      004000      TS.NXM==4000      ;NON-EXISTENT MEMORY.
243      002000      TS.NBA==2000      ;NEED BUFFER ADDRESS.
244      001000      TS.A17==1000      ;BUS ADDRESS BIT 17.
245      000400      TS.A16==400      ;BUS ADDRESS BIT 16.
246      000200      TS.SSR==200      ;UNIT READY BIT.
247      000100      TS.OFL==100      ;OFF LINE.
248      177717      TSC.FCC==177717      ;FATAL CLASS CODE MASK.
249      177761      TSC.TCC==177761      ;TERMINATION CLASS CODE MASK.
250
251      ;THE FOLLOWING ARE BIT DEFINITIONS FOR THE COMMAND WORD
252
253      100000      ACK.C==100000      ;ACKNOWLEDGE BIT
254      040000      CVC.C==40000      ;CLEAR VOLUME CHECK.
255      020000      OPP.C==20000      ;OPPOSITE BIT
256      010000      SWB.C==10000      ;SWAP BYTE BIT
257      004000      MOD.C3==4000      ;MODE BIT 3
258      004000      BRFC.C==4000      ;BYTE/RECORD/FILE COUNT FLAG BIT. NOT USED
259      ;BY TS05 BUT USED INTERNALLY BY THIS PROGRAM ONLY.
260      002000      MOD.C2==2000      ;MODE BIT 2
261      001000      MOD.C1==1000      ;MODE BIT 1
262      000400      MOD.C0==400      ;MODE BIT 0
263      000200      IE.C==200      ;INTERRUPT ENABLE
264      000100      FMT.C1==100      ;FORMAT BIT 1
265      000100      VFY.C==100      ;WRITE VERIFY FLAG BIT. INTERNAL USE ONLY.
266      ;NOT USED BY TS05.
267      000040      FMT.C0==40      ;FORMAT BIT 0.
268      000040      JMP.C==40      ;JUMP BIT-TO DIRECT THIS PROGRAM TO JUMP TO
269      ;A CERTAIN LOCATION IN THE COMMAND SEQUENCE
270      ;TABLE. INTERNAL USE ONLY.
271      000020      CMD.C4==20      ;COMMAND BIT 4
272      000020      DLY.C==20      ;INSERT DELAY. INTERNAL USE ONLY.
273      000010      CMD.C3==10      ;COMMAND BIT 3
274      000004      CMD.C2==4      ;COMMAND BIT 2
275      000002      CMD.C1==2      ;COMMAND BIT 1
276      000001      CMD.C0==1      ;COMMAND BIT 0
277
278      ;BIT DEFINITIONS FOR DEVICE CHARACTERISTICS.
279
280      000200      CH.ESS==200      ;ENABLE SKIP TAPE MARKS STOP (STOP AT LOGICAL EOT).
281      000040      CH.EAI==40      ;ENABLE ATTENTION INTERRUPTS.
282      000020      CH.ERI==20      ;ENABLE MESSAGE BUFFER RELEASE INTERRUPTS.
283      000040      DFTSCH==CH.EAI      ;DEFAULT CHARACTERISTICS CODE.

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL EQUATES SECTION

SEQ 0046

```

285
286      ;BIT DEFINITIONS FOR EXTENDED CONTROL WORD
287
288      000040      EF.HSS==40      ;ENABLE HIGH SPEED SELECT
289      000030      EF.RWB==30      ;ENABLE BOTH READ & WRITE BUFFERING
290      000020      EF.RBO==20      ;ENABLE READ BUFFERING ONLY
291
292      ;THE FOLLOWING INDICATES THE RELATIVE POSITIONS OF THE STATUS WORDS
293      ;IN THE MESSAGE BUFFER.
294
295      000004      MS.RFC==4      ;RESIDUAL FRAME COUNT.
296      000006      MS.XS0==6      ;EXT STATUS REG 0
297      000010      MS.XS1==10     ;EXT STATUS REG 1
298      000012      MS.XS2==12     ;EXT STATUS REG 2
299      000014      MS.XS3==14     ;EXT STATUS REG 3
300      000016      MS.XS4==16     ;EXT STATUS REG 4
301
302      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 0.
303
304      100000      X0.TMK==100000 ;TAPE MARK.
305      040000      X0.RLS==40000  ;RECORD LENGTH SHORT.
306      020000      X0.LET==20000  ;LOGICAL EOT.
307      010000      X0.RLL==10000  ;RECORD LENGTH LONG.
308      000100      X0.ONL==100    ;ON LINE BIT.
309      000004      X0.WLK==4      ;WRITE LOCK BIT
310      000002      X0.BOT==2      ;BOT BIT.
311      000001      X0.EOT==1      ;EOT BIT.
312
313      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 2.
314
315      100000      X2.OPM==100000  ;OPERATION IN PROGRESS, TAPE MOVING
316      000200      X2.EFE==200    ;EXTENDED FEATURES ENABLED
317      000100      X2.BFE==100    ;BUFFERING ENABLED
318
319      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 3.
320
321      000010      X3.DCK==10      ;DENSITY CHECK.
322      157400      X3.RNY==157400  ;CAPSTAN RUNAWAY UDIAG ERROR CODE.
323
324      ;THE FOLLOWING ARE BIT DEFINITIONS FOR EXTENDED STATUS REGISTER 4.
325
326      100000      X4.HSS==100000  ;HIGH SPEED SWITCH INDICATING 100IPS
327      040000      X4.RCE==40000   ;RETRY COUNT EXCEEDED
328
329      ;THE FOLLOWING DEFINITIONS SHOW THE RELATIVE POSITIONS OF THE COMMAND
330      ;PACKET ENTRIES.
331
332
333      000000      CP.CMD==0      ;CMDPKT.0==TS05 COMMAND.
334      000002      CP.ADL==2      ;CMDPKT.2==BUFFER ADDRESS LOW.
335      000004      CP.ADH==4      ;CMDPKT.4==BUFFER ADDRESS HIGH.
336      000006      CP.CNT==6      ;CKDPKT.6==BYTE/FILE/RECORD COUNT

```

```

338
339      ;MISCELLANEOUS DEFINITIONS.
340
341      000340      INTPRI==PRI07      ;PRIORITY TO BE USED IN INTERRUPT STATE.
342      000012      SCHCNT==12        ;ARBITRARY BYTE LENGTH FOR CHARACTERISTIC
343                                     ;BUFFER LENGTH. (EVEN #)
344      000020      MSGCNT==20        ;MESSAGE BUFFER LENGTH IN BYTES. (EVEN #)
345      000020      DIACNT==20        ;DIAGNOSTIC COMMAND BUFFER EXTENT.
346      004000      DATCNT==2048.     ;MAXIMUM RECORD LENGTH IN BYTES.
347                                     ;THIS COUNT SHOULD BE A MULTIPLE OF 256 TO INSURE
348                                     ;PROPER READ/WRITE BUFFER ALLOCATION BY THE SUPER.
349      177740      RNOPSC==177740    ;RANDOM # OF OPERATIONS MASK.
350      000007      RANP==7           ;CODE TO SELECT RANDOM PATTERN.
351      000020      RRECL==16.        ;READ RECOVERY ATTEMPT LIMIT.
352      000020      WRECL==16.        ;WRITE RECOVERY ATTEMPT LIMIT.
353      153624      RANBC==153624     ;CONSTANT USED TO RESET RANDOM # GENERATOR BASE.
354      032561      RANSC==32561      ;CONSTANT USED TO RESET RANDOM # SAVE LOCATION.
355      177774      NINUSE==177774    ;NOT IN USE CODE FOR DEVICE STATE TABLE.
356      177740      NCMD.C==ACK.C!CVC.C!OPP.C!SWB.C!MOD.C3!MOD.C2!MOD.C1!MOD.CO!IE.C!FMT.C1!FMT.CO
357                                     ;NOT "COMMAND" BITS.
358
359      ;THE FOLLOWING DEFINES THE COMMAND WORD FOR EACH TS05 COMMAND.
360
361      100013      DRI== ACK.C!CMD.C3!CMD.C1!CMD.CO
362                                     ;DRIVE INIT.
363
364      104001      PDF== ACK.C!BRF.C!CMD.CO
365                                     ;READ FORWARD
366
367      104401      RDR== ACK.C!BRF.C!MOD.CO!CMD.CO
368                                     ;READ REVERSE
369
370      104005      WRT== ACK.C!BRF.C!CMD.CO!CMD.C2
371                                     ;WRITE COMMAND
372
373      104105      WTV== ACK.C!BRF.C!VFY.C!CMD.CO!CMD.C2
374                                     ;WRITE VERIFY
375
376      104010      SRF== ACK.C!BRF.C!CMD.C3
377                                     ;SPACE RECORD FORWARD
378
379      104410      SRR== ACK.C!BRF.C!MOD.CO!CMD.C3
380                                     ;SPACE RECORD REVERSE
381
382      105401      RNR== ACK.C!BRF.C!MOD.C1!MOD.CO!CMD.CO
383                                     ;READ REV RETRY1 - REREAD NEXT REVERSE, IE. SPACE FWD, READ REVERSE
384
385      125401      RNF== ACK.C!BRF.C!OPP.C!MOD.C1!MOD.CO!CMD.CO
386                                     ;READ REV RETRY2 - REREAD NEXT FORWARD, IE. READ FORWARD, SPACE REVERSE
387
388      105001      RPF== ACK.C!BRF.C!MOD.C1!CMD.CO
389                                     ;READ FWD RETRY1 - REREAD PREVIOUS FORWARD, IE. SPACE REVERSE, READ FORWARD
390
391      125001      RPR== ACK.C!BRF.C!OPP.C!MOD.C1!CMD.CO
392                                     ;READ FWD RETRY2 - REREAD PREVIOUS REVERSE, IE. READ REVERSE, SPACE FORWARD
393
394      105005      WRR== ACK.C!MOD.C1!BRF.C!CMD.C2!CMD.CO

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL EQUATES SECTION

SEQ 0048

395				;WRITE RETRY
396				
397	102010	RWD==	ACK.C!MOD.C2!CMD.C3	
398				;REWIND COMMAND
399				
400	100012	MBR==	ACK.C!CMD.C3!CMD.C1	
401				;MESSAGE BUFFER RELEASE
402				
403	100011	WTM==	ACK.C!CMD.C3!CMD.C0	
404				;WRITE TAPE MARK.
405				
406	101011	WTR==	ACK.C!MOD.C1!CMD.C3!CMD.C0	
407				;WRITE TAPE MARK RETRY.
408				
409	105010	SFF==	ACK.C!BRF.C!MOD.C1!CMD.C3	
410				;SPACE FILE FORWARD
411				
412	105410	SFR==	ACK.C!BRF.C!MOD.C0!MOD.C1!CMD.C3	
413				;SPACE FILE REVERSE
414				
415	100017	GES==	ACK.C!CMD.C0!CMD.C1!CMD.C2!CMD.C3	
416				;GET EXTENDED STATUS
417				
418	100411	ERS==	ACK.C!MOD.C0!CMD.C3!CMD.C0	
419				;ERASE 3 INCHES OF TAPE
420				
421	100412	UNL==	ACK.C!MOD.C0!CMD.C3!CMD.C1	
422				;UNLOAD COMMAND
423				
424	101012	CLN==	ACK.C!MOD.C1!CMD.C3!CMD.C1	
425				;ERASE TAPE.
426				
427	140004	SCH==	ACK.C!CVC.C!CMD.C2	;SET DEVICE CHARACTERISTICS.
428				
429	140006	WSM==	ACK.C!CVC.C!CMD.C2!CMD.C1	;WRITE SUB-SYS MEM
430				
431	100006	DIA==	ACK.C!CMD.C2!CMD.C1	;DIAGNOSTICS.
432				
433	000040	JMP==	JMP.C	;JUMP TO "N"TH COMMAND
434				
435	000020	DLY==	DLY.C	;DELAY "N" MS.
436				
437	177777	END==	177777	;END OF COMMAND SEQUENCES

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL DATA SECTION

SEQ 0049

```

439 .SBTTL GLOBAL DATA SECTION
440 ;**
441 ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
442 ; IN MORE THAN ONE TEST.
443 ;--
444 ; COMMAND PACKET.
445 ; = <..+3>&177774 ;MUST BE ON MOD 4 BOUNDARY.
446 002330 000000 CNDPKT:: 0 ;1ST WORD IS TS05 COMMAND.
447 002332 000000 0 ;2ND WORD IS THE BUFFER LOW ADDRESS.
448 002334 000000 0 ;3RD WORD IS THE BUFFER HIGH ADDRESS.
449 002336 000000 0 ;4TH WORD IS THE BYTE/RECORD/FILE COUNT.
450
451 ; GET STATUS COMMAND PACKET.
452
453 ; = <..+3>&177774 ;MUST BE ON MOD 4 BOUNDARY.
454 002340 100017 GSCP:: .WORD GES
455
456 ; MESSAGE BUFFER RELEASE COMMAND PACKET.
457
458 ; = <..+3>&177774 ;MUST BE ON MOD 4 BOUNDARY.
459 002344 100012 BRCPK:: .WORD MBR
460
461 ; REWIND COMMAND PACKET (USED IN ERROR RECOVERY ONLY)
462
463 ; = <..+3>&177774 ;MUST BE ON MOD 4 BOUNDARY.
464 002350
465 002350 102010 RWCPK:: .WORD RWD
466 002352 000001 .WORD 1
467
468 ; WORK AREA FOR ANALYSIS OF MESSAGE PACKET CONTENTS.
469
470 002354 MSGPKT:: .BLKW 8. ;1ST WORD:: MESSAGE TYPE.
471 ;2ND WORD:: DATA FIELD LENGTH.
472 ;3RD WORD:: RESIDUAL FRAME COUNT.
473 ;4TH WORD:: XSTAT0
474 ;5TH WORD:: XSTAT1
475 ;6TH WORD:: XSTAT2
476 ;7TH WORD:: XSTAT3
477 ;8TH WORD:: XSTAT4
478
479 ; MESSAGE PACKETS.
480 002374 MSGPK0:: .BLKW 8. ;MESSAGE PACKET FOR DEVICE #0
481 002414 MSGPK1:: .BLKW 8. ;MESSAGE PACKET FOR DEVICE #1
482 002434 MSGPK2:: .BLKW 8. ;MESSAGE PACKET FOR DEVICE #2
483 002454 MSGPK3:: .BLKW 8. ;MESSAGE PACKET FOR DEVICE #3
484
485 ; SET CHARACTERISTIC BLOCK.
486 002474 002374 SCHBK:: MSGPK0 ;1ST WORD:: MSGPKT ADDR LO(SET UP BY EXECUTE ROUTINE).
487 002476 000000 0 ;2ND WORD:: MSGPKT ADDR HI.
488 002500 000020 MSGCNT ;3RD WORD:: MSG BUFFER LENGTH (BYTES)
489 002502 000040 CH.EAI ;4TH WORD:: CHARACTERISTICS WORD(SET BY SETUP ROUTINE).
490 002504 000000 0 ;5TH WORD:: MSP & BUFFER CONTROL ON EXT'D FEATURES
491
492 ; WRITE SUB-SYSTEM MEMORY CHARACTERISTIC BLOCK.
493 002506 000000 WSMBK:: 0 ;1ST WORD:: SEL 0
494 002510 000000 0 ;2ND WORD:: SEL 2
495 002512 000000 0 ;3RD WORD:: SEL 4

```

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL DATA SECTION

SEQ 0050

```

496
497
498 002514
499 002524
500 002534
501      002514
502
503
504 002544 002374
505 002546 002414
506 002550 002434
507 002552 002454
508
509
510 002554 010034
511 002556 010042
512 002560 010050
513 002562 010056
514
515
516 002564 000000
517 002566 000000
518 002570 000000
519 002572 000000
520
521
522 002574 000000
523 002576 000000
524 002600 000000
525 002602 000000
526
527
528
529
530 002604 177774
531 002606 177774
532 002610 177774
533 002612 177774
534 002614 177777
535
536
537
538 002616 003046
539 002620 003120
540 002622 003172
541 002624 003244

;      TS05 REGISTER ADDRESSES.
TSDB : .BLKW 4 ;TS05 DATA BUFFER ADDRESSES.
TSSR.: .BLKW 4 ;TS05 STATUS REGISTER ADDRESSES.
TSVCT.: .BLKW 4 ;TS05 VECTOR ADDRESSES.
TSBA==TSDB ;DATA BUFFER ADDRESS REGISTER.

;      ADDRESSES OF MESSAGE PACKETS.
MSGPKA:: MSGPK0 ;DEVICE 0.
          MSGPK1 ;DEVICE 1.
          MSGPK2 ;DEVICE 2.
          MSGPK3 ;DEVICE 3.

;      ADDRESSES OF INTERRUPT HANDLING ROUTINES.
TSSINT:: TSSINO ;DEVICE 0.
          TSSIN1 ;DEVICE 1.
          TSSIN2 ;DEVICE 2.
          TSSIN3 ;DEVICE 3.

;      TS05 CODE LEVELS, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
TSSCL:: 0 ;DEVICE 0
        0 ;DEVICE 1
        0 ;DEVICE 2
        0 ;DEVICE 3

;      TS05 EXT. FEA & BUF. ENA SW'S, WILL BE STORED AFTER SCH CMD IN BASIC FUNCTION TEST
TSSSW:: 0 ;DEVICE 0
        0 ;DEVICE 1
        0 ;DEVICE 2
        0 ;DEVICE 3

;      UNIT NUMBERS OF ALL DEVICES BEING TESTED(1-4).
;      WHEN DEVICE IS NOT IN USE, IT'S LOCATION WILL = 3.
;      RS WILL ALWAYS CONTAIN THE PRESENT LOGICAL UNIT NUMBER X 2.
DEVTBL:: .WORD NINUSE
          .WORD NINUSE
          .WORD NINUSE
          .WORD NINUSE
          .WORD END

;      BAD TAPE TABLE POINTER: USED BY WRITE RETRY ROUTINE
;      "WRTY" TO LOG BAD TAPE SPOTS ON UNITS UNDER TEST
BTADDR:: BT0
          BT1
          BT2
          BT3

```

```

543      ;          COUNTER AREA.
544
545      002626      CNTBGN=.
546      002626      WRBC:: .BLKW 20      ;BYTES WRITTEN.
547      002666      RRBC:: .BLKW 20      ;BYTES READ REV.
548      002726      RFBC:: .BLKW 20      ;BYTES READ FWD.
549      002766      WRREC:: .BLKW 4      ;RECOVERABLE WRITE ERRORS.
550      002776      WRUNR:: .BLKW 4      ;UNRECOVERABLE WRITE ERRORS.
551      003006      RRREC:: .BLKW 4      ;RECOVERABLE READ REV ERRORS.
552      003016      RRUNR:: .BLKW 4      ;UNRECOVERABLE READ REV ERRORS.
553      003026      RFREC:: .BLKW 4      ;RECOVERABLE READ FWD ERRORS.
554      003036      RFUNR:: .BLKW 4      ;UNRECOVERABLE READ FWD ERRORS.
555      003046      BT0:: .BLKW 21      ;UNIT 0 BAT TAPE SPOTS LOG
556      003120      BT1:: .BLKW 21      ;UNIT 1 BAT TAPE SPOTS LOG
557      003172      BT2:: .BLKW 21      ;UNIT 2 BAT TAPE SPOTS LOG
558      003244      BT3:: .BLKW 21      ;UNIT 3 BAT TAPE SPOTS LOG
559      003316      WRTYCT:: .BLKW 4      ;WRITE RETRY COUNTER
560      003326      PASCNT:: .BLKW 4      ;PASS COUNT.
561      003336      SCCNT:: .BLKW 4      ;SPECIAL CONDITION COUNT.
562      003346      VFYCNT:: .BLKW 4      ;COUNT OF TS05 DATA COMPARE ERRORS.
563      003356      HRCNT:: .BLKW 4      ;COUNT OF HARD ERRORS.
564      003366      FTLCNT:: .BLKW 4      ;COUNT OF FATAL ERRORS.
565      003376      CNTEND=.      ;END OF STATISTICAL COUNTERS.
566      003376      RECCNT:: .BLKW 4      ;NUMBER OF RECORDS FROM BOT: CLEARED ON REWIND
567      ;          ;AND WHEN RESTARTING OR CONTINUING TEST 2.
568      000550      CNTLEN==CNTEND-CNTBGN      ;LENGTH OF STATISTICAL COUNTER AREA.
569
570
571      ;          THE FOLLOWING ARE THE DEFINITIONS OF VARIABLES
572      ;          USED BY THE PROGRAM.
573
574      003406      000000      DATAW:: .WORD 0      ;WRITE BUFFER ADDRESS.
575      003406      DIABLK==DATAW      ;WRITE BUFFER ALSO USED FOR DIAG CMD.
576      003410      000000      DATARD:: .WORD 0      ;READ BUFFER ADDRESS.
577      003412      000000      NCNT:: .WORD 0      ;STORAGE FOR VALUE OF N.
578      003414      000000      NCNT1:: .WORD 0      ;TEMP STORAGE FOR VALUE OF N.
579      003416      000000      BRCNT:: .WORD 0      ;STORAGE FOR BPCR VALUE.
580      003420      177777      CMDWRD:: .WORD END      ;CONTAINS COMMAND WORD BEING EXECUTED PRESENTLY.
581      003422      177777      CMDSAV:: .WORD END      ;SAVE LOCATION FOR CMD WORD DURING ERROR RECOVERY
582      003424      177777      PCMDWD:: .WORD END      ;CONTAINS PREVIOUS COMMAND WORD.
583      003426      000000      CMDLG:: .WORD 0      ;CURRENT COMMAND LOGGING CODE.
584      003430      000000      LENMSK:: .WORD 0      ;RANDOM WRITE LENGTH MASK, TO BE SET UP BY TESTS
585      003432      153624      RANB:: .WORD 153624      ;RANDOM # GENERATOR BASE.
586      003434      032561      RANS:: .WORD 32561      ;RANDOM # SAVE LOCATION.
587      003436      000000      TIME1:: .WORD 0      ;TIME COUNT 1.
588      003440      000000      TIME2:: .WORD 0      ;TIME COUNT 2.
589      003442      000000      JLOOP:: .WORD 0      ;JMP COMMAND LOOP COUNT.
590      003444      000000      JLOC:: .WORD 0      ;JMP COMMAND LOCATION COUNT.
591      003446      000000      PATERN:: .WORD 0      ;PATTERN SELECT CODE.
592      003450      000000      CTCC:: .WORD 0      ;CURRENT TERMINATION CLASS CODE.
593      003452      000000      RSSAVE:: .WORD 0      ;LOCATION FOR SAVING CURRENT DEVICE POINTER.
594      003454      000000      TSSREG:: .WORD 0      ;CURRENT STATUS REGISTER.
595      003456      000000      WTMFLG:: .WORD 0      ;WRITE TAPE MARK FLAG

```

```

59*      .      ERROR FLAG AREA, THESE FLAGS ARE CLEARED DURING INITIALIZATION AND
598      .      AFTER EACH COMMAND IS COMPLETED.
599
600      003460      BGNFLG*.
601 003460 000000      RETRYC:: .WORD 0      ;# OF RECOVERY ATTEMPTS EXECUTED.
602 003462      000      RPTCNT:: .BYTE 0      ;WRITE REPEAT ON SAME SPOT CNTR. 4 PER WRITE RETRY
603 003463      000      WRTYFG:: .BYTE 0      ;WRITE RETRY ON SAME SPOT IN PROGRESS FLAG
604 003464      000      WRTYER:: .BYTE 0      ;WRITE RETRY ON SAME SPOT ERROR FLAG
605 003465      000      RECLOG:: .BYTE 0      ;RECORD COUNT HAS BEEN UPDATED FOR THIS RECORD.
606 003466      000      ERLOG:: .BYTE 0      ;DATA BYTES AND ERRORS HAVE BEEN LOGGED FOR THIS RECORD.
607 003467      000      RMERR:: .BYTE 0      ;READ/WRITE ERROR HAS OCCURED.
608 003470      000      UNREC:: .BYTE 0      ;UNRECOVERABLE ERROR HAS OCCURED.
609 003471      000      ERRREC:: .BYTE 0      ;ERROR RECOVERY MODE.
610      .EVEN
611      003472      ENDFLG*.
612
613
614      :      ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED DURING INITIALIZATION.
615
616 003472      INTFLG:: .BLKW 4      ;INTERRUPT OCCURRED FLAGS FOR EACH DEVICE.
617 003502      EOTFLG:: .BLKW 4      ;EOT/BOT FLAGS FOR EACH DEVICE (XSTATO).
618 003512 000000      BTPT:: .WORD 0      ;BAD TAPE SPOT POINTER TO BT0-BT3 VIA BTADDR
619 003514      000      EXPBOT:: .BYTE 0      ;BOT IS EXPECTED, DO NOT ABORT ON BOT/FUNC RTI.
620 003515      000      RANDOM:: .BYTE 0      ;RANDOM EVERYTHING FLAG.
621 003516      000      VFYFLG:: .BYTE 0      ;SET DURING WRITE/VERIFY COMMAND.
622 003517      000      RPTFLG:: .BYTE 0      ;PERFORMANCE REPORT HAS BEEN REQUESTED.
623 003520      000      SWBFLG:: .BYTE 0      ;ENABLES SWAP BYTE FUNCTION WHEN NOT EQUAL TO ZERO.
624 003521      000      IRE:: .BYTE 0      ;INHIBIT RESIDUAL FRAME COUNT ERROR REPORT.
625 003522      000      DROPED:: .BYTE 0      ;CURRENT UNIT HAS BEEN DROPPED
626 003523      000      T1SWB:: .BYTE 0      ;TEST1 SWAP BYTES FLAG
627 003524      000      ALLEOT:: .BYTE 0      ;ALL UNITS @ EOT FLAG
628 003525      000      ERSFLG:: .BYTE 0      ;ERASE FLAG: DO ERASE AFTER A SPACE REV TO DELETE
629      .EVEN      ;BADLY WRITTEN RECORD. 1 TO 4 ERASES LEAVING
630      .EVEN      ;A 3 TO 12 INCH GAP MAY RESULT.
631
632      003526      ENDFLG*.
633
634      :      ADDITIONAL FLAGS, THESE FLAGS ARE CLEARED ONLY AFTER BEING CHECKED.
635
636 003526      000      STAFLG:: .BYTE 0      ;START FLAG - SET BY INIT CODE IF STARTING.
637 003527      000      PWRFLG:: .BYTE 0      ;POWER FAILURE FLAG - SET ONLY DURING INIT.
638 003530      000      TRAPD4:: .BYTE 0      ;TRAPED AT 4 FLAG
639 003531      000      MISCFG:: .BYTE 0      ;MISCELLANEOUS FLAG
640 003532 000000      TSUNT:: .WORD 0      ;NUMBER OF THE UNIT UNDER TEST PLUS HSSP&BUF
641 003534 000000      TSNP:: .WORD 0      ;FOR PRINT OUT UNIT # ONLY
642
643      :      OPERATOR FLAG SETTINGS PASSED BY DIAG. SUPERVISOR IN A 16 BIT WORD
644      :      SEE GLOBAL EQUATES SECTION FOR FLAG BIT LIST
645
646 003536 000000      OPFLAG:: .WORD 0      ;READ ONLY OPERATOR FLAG WORD
647      .EVEN

```

```

649
650
651
652
653 003540 140004
654 003542 000040
655 003544 000001
656 003546 000000
657 003550 102010
658 003552 000001
659 003554 000001
660 003556 000007
661 003560 104005
662 003562 004000
663 003564 076400
664 003566 000007
665 003570 104401
666 003572 004000
667 003574 076400
668 003576 000007
669 003600 104001
670 003602 004000
671 003604 076400
672 003606 000007
673 003610 102010
674 003612 000001
675 003614 000001
676 003616 000007
677 003620
678 003740 177777
679 003742 177777
680 003744 177777
681 003746 177777
682 003750 177777
683
684
685 003752 100013
686 003754 104001
687 003756 104401
688 003760 104005
689 003762 104105
690
691
692 003764 104010
693 003766 104410
694 003770 105401
695 003772 125401
696 003774 105001
697 003776 125001
698 004000 105005
699 004002 102010
700 004004 100012
701 004006 100011
702 004010 101011
703 004012 105010
704 004014 105410
705 004016 100017

;THE FOLLOWING IS THE COMMAND SEQUENCE TABLE. THE TABLE
;HAS DEFAULT VALUES AT PROGRAM LOAD AS SHOWN. THESE VALUES
;CAN BE UPDATED BY A TEST OR BY OPERATOR INPUT.

CMOSEQ:: .WORD SCH ;SET CHARACTERISTICS.
         .WORD CH.EAI
         .WORD 1
         .WORD 0
CMOSEQ2:: .WORD RWD ;REWIND.
         .WORD 1 ;BYTE COUNT.
         .WORD 1 ;ONCE.
         .WORD RANP ;PATTERN.
         .WORD WRT ;WRITE.
         .WORD DATCNT ;MAX BUFFER LENGTH.
         .WORD 32000. ;32,000 RECORDS.
         .WORD RANP ;RANDOM PATTERN.
         .WORD RDR ;READ REV.
         .WORD DATCNT ;MAX BUFFER LENGTH.
         .WORD 32000. ;32,000 RECORDS
         .WORD RANP ;RANDOM PATTERN.
         .WORD RDF ;READ FWD.
         .WORD DATCNT ;MAX BUFFER LENGTH.
         .WORD 32000. ;32,000 RECORDS.
         .WORD RANP ;RANDOM PATTERN.
         .WORD RWD ;REWIND.
         .WORD 1 ;BYTE COUNT.
         .WORD 1 ;ONCE.
         .WORD RANP ;PATTERN.
         .WORD BLKW 40. ;EXTENSION TO DOUBLE BUFFER SIZE
SEGENO:: .WORD END ;SOFT END OF SEQUENCE TABLE.
         .WORD END
         .WORD END
         .WORD END
         .WORD END ;HARD END OF SEQUENCE TABLE.

;THE FOLLOWING IS THE TS05 COMMAND TABLE

CMOTBL:: .WORD DRI ;DRIVE INIT.
         .WORD RDF ;READ FORWARD.
         .WORD RDR ;READ REVERSE.
         .WORD WRT ;WRITE
         .WORD WTV ;WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND
         ;CHECK DATA ON ALL RECORDS, RDF AND
         ;CHECK DATA ON ALL RECORDS.)
         .WORD SRF ;SPACE "N" RECORDS FORWARD.
         .WORD SRR ;SPACE "N" RECORDS REVERSE.
         .WORD RNR ;READ NEXT REVERSE. I.E., SPACE FWD, READ REVERSE.
         .WORD RNF ;READ NEXT FORWARD. I.E., READ FORWARD, SPACE REVERSE.
         .WORD RPF ;READ PREVIOUS FORWARD. I.E., SPACE REVERSE, READ FORWARD
         .WORD RPR ;READ PREVIOUS REVERSE. I.E., READ REVERSE, SPACE FORWARD
         .WORD WRR ;WRITE RETRY.
         .WORD RWD ;REWIND.
         .WORD MBR ;MESSAGE BUFFER RELEASE
         .WORD WTM ;WRITE TAPE MARK
         .WORD WTR ;WRITE TAPE MARK RETRY.
         .WORD SFF ;SPACE "N" FILES FORWARD.
         .WORD SFR ;SPACE "N" FILES REVERSE.
         .WORD GES ;GET EXTENDED STATUS.

```

706	004020	100411		.WORD	ERS	;ERASE 3 INCHES OF TAPE.
707	004022	100412		.WORD	UNL	;REWIND AND UNLOAD.
708	004024	101012		.WORD	CLN	;CLEAR TAPE.
709	004026	140004		.WORD	SCH	;SET CHARACTERISTICS.
710	004030	100006		.WORD	DIA	;DIAGNOSTIC COMMAND.
711	004032	000040		.WORD	JMP	;JUMP TO THE NTH COMMAND IN THE SEQUENCE.
712	004034	000020		.WORD	DLY	;DELAY "N" MS.
713	004036	177777		.WORD	END	;END OF COMMAND TABLE
714						
715						
716						
717	004040	104	122	111	CMDASC:: .ASCII /DRI/	;DRIVE INIT.
718	004043	122	104	106	.ASCII /RDF/	;READ FORWARD.
719	004046	122	104	122	.ASCII /RDR/	;READ REVERSE.
720	004051	127	122	124	.ASCII /WRT/	;WRITE
721	004054	127	124	126	.ASCII /WTV/	;WRITE/VERIFY. (WRITE ALL RECORDS, RDR AND CHECK DATA
722						;ON ALL RECORDS, RDF AND CHECK DATA ON ALL RECORDS.)
723	004057	123	122	106	.ASCII /SRF/	;SPACE "N" RECORDS FORWARD.
724	004062	123	122	122	.ASCII /SRR/	;SPACE "N" RECORDS REVERSE.
725	004065	122	116	122	.ASCII /RNR/	;READ NEXT REVERSE. I.E., SPACE FWD READ REVERSE.
726	004070	122	116	106	.ASCII /RNF/	;READ NEXT FORWARD. I.E., READ FORWARD, SPACE REVERSE.
727	004073	122	120	106	.ASCII /RPF/	;READ PREVIOUS FORWARD. IE., SPACE REVERSE, READ FORWARD
728	004076	122	120	122	.ASCII /RPR/	;READ PREVIOUS REVERSE. IE., READ REVERSE, SPACE FORWARD
729	004101	127	122	122	.ASCII /WRR/	;WRITE RETRY.
730	004104	122	127	104	.ASCII /RWD/	;REWIND.
731	004107	115	102	122	.ASCII /MBR/	;MESSAGE BUFFER RELEASE
732	004112	127	124	115	.ASCII /WTH/	;WRITE TAPE MARK
733	004115	127	124	122	.ASCII /WTR/	;WRITE TAPE MARK RETRY.
734	004120	123	106	106	.ASCII /SFF/	;SPACE "N" FILES FORWARD.
735	004123	123	106	122	.ASCII /SFR/	;SPACE "N" FILES REVERSE.
736	004126	107	105	123	.ASCII /GES/	;GET EXTENDED STATUS.
737	004131	105	122	123	.ASCII /ERS/	;ERASE 3 INCHES OF TAPE.
738	004134	125	116	114	.ASCII /UNL/	;REWIND AND UNLOAD.
739	004137	103	114	116	.ASCII /CLN/	;CLEAN TAPE.
740	004142	123	103	110	.ASCII /SCH/	;SET CHARACTERISTICS. WHERE BRF=200, 40, 20, 0.
741						;SEE TSV05/TS05 PROGRAMMING SPECIFICATION FOR DESCRIPTION.
742	004145	104	111	101	.ASCII /DIA/	;DIAGNOSTICS. SEE TSV05/TS05 PROGRAMMING SPECIFICATION
743						;FOR DESCRIPTION. ODT MUST BE USED TO LOAD DIAGNOSTIC DATA
744						;INTO THE WRITE BUFFER BEFORE THIS CMD IS ISSUED.
745	004150	112	115	120	.ASCII /JMP/	;JUMP TO THE NTH COMMAND IN THE COMMAND
746						;SEQUENCE TABLE, WHERE N IS DEFINED IN
747						;THE # OF OPERATIONS.
748	004153	104	114	131	.ASCII /DLY/	;DELAY "N" MS, WHERE N IS DEFINED IN
749						;THE # OF OPERATIONS.
750	004156	105	116	104	.ASCII /END/	;END OF COMMAND SEQUENCE.
751					.EVEN	
752						

```

754          .SBTTL GLOBAL TEXT SECTION
755
756          ;**
757          ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
758          ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
759          ; MORE THAN ONE TEST.
760          ;
761
762
763          ;
764          ; FORMAT STATEMENTS USED IN PRINT CALLS
765          ;
766
767          .NLIST BEX
768
769          004162      045      116      045 CODELM:: .ASCIZ /%N%AUNIT %D1%A TS05 CODE LEVEL %03%N%N/
770          004231      045      116      045 SWSET:: .ASCIZ /%N%AUNIT %D1%A TS05 SWITCH SETTINGS %03%N%N/
771
772          .EVEN
773          004306      130      130      130 HALTM:: .ASCIZ /XXX CMD - TYPE <CR> TO CONTINUE/
774          004346      103      115      104 CMDPKM:: .ASCIZ /CMD PACKET ADR NOT ON MODULO 4 BOUNDARY: RELOAD!/
775
776          .EVEN
777          004430      104      101      124 WTVRM:: .ASCIZ /DATA COMPARE ERROR/
778          004453      116      117      040 TOERM:: .ASCIZ /NO TSV05 RESPONSE/
779          004475      125      116      104 SCERM:: .ASCIZ /UNDEFINED SPEC COND/
780          004521      122      106      103 RFCERM:: .ASCIZ /RFC NON ZERO/
781          004536      124      123      126 NSSRM:: .ASCIZ /TSV05 NOT READY/
782          004556      122      105      124 RLEXM:: .ASCIZ /RETRY LIMIT EXCEEDED/
783          004603      104      122      111 ATTNM:: .ASCIZ /DRIVE OFF LINE/
784          004622      106      125      116 FUNRM:: .ASCIZ /FUNCTION REJECT/
785          004642      106      101      124 FATSM:: .ASCIZ /FATAL SUBSYSTEM ERROR/
786          004670      116      117      040 NOINTM:: .ASCIZ /NO INTERRUPT/
787          004705      124      101      120 TSAM:: .ASCIZ /TAPE STATUS ALERT/
788          004727      124      117      117 TOOMM:: .ASCIZ /TOO MANY INTERRUPTS/
789          004753      103      101      120 RNYM:: .ASCIZ /CAPSTAN RUNAWAY-GET STATUS RESULTS:/
790          005017      122      105      103 RERM:: .ASCIZ /RECOVERABLE ERROR/
791          005041      125      116      122 URERM:: .ASCIZ /UNRECOVERABLE ERROR/
792          005065      045      116      045 DROPDM:: .ASCIZ /%N%ADROPPED UNIT %D1%N/
793          005114      045      116      045 AUDRPM:: .ASCIZ /%N%AALL UNITS DROPPED%N%N/
794          005146      045      116      045 AUDRUN:: .ASCIZ /%N%ADIAGNOSTIC ONLY SUPPORTS ONE CONTROLLER%N%N/
795          005226      045      116      045 DTAER2:: .ASCIZ "%N%ABYTE:%D4%S2%AWAS:%B8%S2%AS/B:%B8%N"
796          005275      045      104      064 DTAER3:: .ASCIZ "%D4%A BYTES IN ERROR OUT OF %D4%N"
797          005337      045      101      116 DTAER4:: .ASCIZ /%N%AND DATA READ%N/
798          005360      045      101      122 DTAER5:: .ASCIZ /%N%ARECORD TOO LONG: >%D4%A BYTES%N/
799          005422      045      101      122 NURTY1:: .ASCIZ /%N%ARECOVERED ON RETRY %D2%N/
800          005456      045      101      104 OFLINM:: .ASCIZ /%N%ADRIVE %D1%A OFF LINE%N/
801          005507      045      101      107 GETSTM:: .ASCIZ /%N%AGET STATUS CMD RESULTS:%N/
802          005543      045      116      045 NODEV:: .ASCII /%N%ABUS TRAP AT %06%N/
803          005570      045      101      111 .ASCIZ /%N%INTERFACE BAD OR TSDB NOT SET TO ABOVE ADDRESS%N/
804          005653      040      052      052 UNILK:: .ASCIZ / *****TAPE IS WRITE-LOCKED AND WILL CAUSE ERRORS*****/
805          005741      045      116      000 CRLF:: .ASCIZ /%N/
806          005744      045      116      045 CRLFSP:: .ASCIZ /%N%S7/
807
808          .LIST BEX
809          .EVEN
810
811
812

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL ERROR REPORT SECTION

SEQ 0056

```

814      .SBTTL  GLOBAL ERROR REPORT SECTION
815
816      ;**
817      ; THE GLOBAL ERROR REPORT SECTION CONTAINS THE PRINTB AND PRINTX CALLS
818      ; THAT ARE USED IN MORE THAN ONE TEST.  IT ALSO INCLUDES THE ASCII MESSAGES
819      ; THAT ARE USED BY THE PRINTB AND PRINTX CALLS..
820      ;
821
822
823      BGNMSG  DTAERM
      DTAERM:
829      DATERM:PRINTB  #STAER1,TSNP,PASCNT(R5),RECCNT(R5)
      MOV      RECCNT(R5), (SP)
      MOV      PASCNT(R5), -(SP)
      MOV      TSNP, (SP)
      MOV      #STAER1, -(SP)
      MOV      #4, -(SP)
      MOV      SP,R0
      TRAP     C$PNTB
      ADD      #12,SP
830      PRINTB  #STAER
      MOV      #STAER7, -(SP)
      MOV      #1, -(SP)
      MOV      SP,R0
      TRAP     C$PNTB
      ADD      #4,SP
831      MOV      R2,RECRED      ;SAVE R2
832      MOV      R3,TIME1      ;SAVE R3
833      MOV      R4,TIME2      ;SAVE R4
834      JSR      PC,RECTAP      ;RETRIEVE RECORD READ
835      MOV      RECRED,R2      ;RESTORE R2
836      MOV      R3,RECRED      ;SAVE RECORD READ
837      MOV      TIME1,R3      ;RESTORE R3
838      MOV      TIME2,R4      ;RESTORE R4
839      PRINTB  #STAER6,RECRED ;PRINT RECORD READ
      MOV      RECRED, -(SP)
      MOV      #STAER6, -(SP)
      MOV      #2, -(SP)
      MOV      SP,R0
      TRAP     C$PNTB
      ADD      #6,SP
840      EXIT      MSG
      .WORD     J$JMP
      .WORD     L10002-2
841      .EVEN
842
843      ENDMMSG
      L10002:
      TRAP     C$MSG
844
845      BGNMSG  STAERM
      STAERM:
846      STAERM:PRINTB  #STAER1,TSNP,PASCNT(R5),RECCNT(R5)
      MOV      RECCNT(R5), -(SP)
      MOV      PASCNT(R5), -(SP)
      MOV      TSNP, -(SP)
      MOV      #STAER1, -(SP)
      MOV      016546 003376
      MOV      016546 003326
      MOV      016746 175400
      MOV      012746 006436

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL ERROR REPORT SECTION

SEQ 0057

006140	012746	000004		MOV	#4, (SP)
006144	010600			MOV	SP, R0
006146	104414			TRAP	C#PNTB
006150	062706	000012		ADD	#12, SP
847 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
848 006174	016702	174130	MOV CMDPKT, R2		
349 006200	042702	177740	BIC #177740, R2		
850 006204	005302		DEC R2		; IF CMD IS A READ
851 006206	005702		TST R2		
852 006210	001016		BNE 500004		
853 006212	004767	001646	JSR PC, RECTAP		; THEN RETRIEVE
854 006216	010367	001642	MOV R3, RECTAP		; AND
855 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
856 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
857 006266			PRINTX #STAER3, CMDPKT, @TSD8(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	@TSD8(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
858 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
859 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)

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL ERROR REPORT SECTION

SEQ 0058

```

006416 012746 000006
006422 010600
006424 104415
006426 062706 000016
860 006432          EXIT MSG
006432 000167
006434 000432
861
862
863 006436      045      101      130 STAER1: .ASCIZ /#AXXX CMD FAILED - UNIT #D1#S3#APASS:#D5#S3#ARECORD:#D5#N/
864
865 006530      045      101      120 STAER7: .ASCIZ /#APREVIOUS CMD WAS XXX /
866 006562      045      123      061 STAER6: .ASCIZ /#S11#A* RECORD READ:#D5#A */
867 006616      045      116      045 STAER2: .ASCIZ /#N#ACMDPKT#S2#ATSBA#S4#ARFC#S5#ATSSR#S3#ATCC#N/
868 006675      045      117      066 STAER3: .ASCIZ /#06#S2#06#S2#06#S2#06#S2#D1#N/
869 006733      045      117      066 STAER4: .ASCII /#06#N/
870 006740      045      117      066
871 006745      045      117      066
872 006753      045      101      130 STAER5: .ASCII /#AXST0#S4#AXST1#S4#AXST2#S4#AXST3#S4#AXST4#N/
873 007027      045      117      066
874
875
876 007066      000000      RECRED: .WORD 0 ;RECORD READ FROM TAPE
877
878 007070
007070
007070 104423
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908

```

.SBTTL GLOBAL SUBROUTINES SECTION
 ;**
 ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
 ; THAT ARE USED IN MORE THAN ONE TEST.
 ;--
 ;*
 ;
 ;ROUTINE TO DO A SOFT INITIALIZE OF THE CONTROLLER
 ;BY WRITING INTO THE TSSR REGISTER. AFTER THE INIT,
 ;THE TSSR REGISTER IS TESTED FOR ERRORS. ANY ERRORS
 ;DETECTED SHOULD BE TREATED AS DEVICE FATAL ERRORS.
 ;
 ;INPUTS:
 ;
 ; R5 CURRENT UNIT NUMBER
 ;
 ;
 ;OUTPUTS:
 ;
 ; R0 CONTENTS OF TSSR, IF ERROR
 ; CARRY SET IF INIT WAS OKAY
 ; CLEAR IF FATAL ERROR
 ;
 ;CALLING SEQUENCE:
 ; JSR PC,FIRSTU

MOV #6, (SP)
 MOV SP,R0
 TRAP C#PNTX
 ADD #16,SP
 .WORD J\$JMP
 .WORD L10003 2 .

TRAP C#MSG

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0059

```

909      ;      JSR      PC,SOFINIT
910      ;      BCS      CONTINUE
911      ;      ERRDF      ;REPORT FATAL ERROR
912      ;
913      ;
914      ;
915 007072      SOFINIT::
916      ;      ; (SAVREG) SAVE THE REGISTERS
917 007072 012775 000000 002524      MOV      #0, @TSSR(R5)      ; DO THE INIT.
918 007100 004767 003574      JSR      PC,WSSR      ;WAIT FOR UNIT TO BE READY
919 007104 012703 000550      MOV      #360.,R3
920 007110 004767 000070      2$:      JSR      PC,WAITF      ; WAIT FOR SSR
921 007114 103416      BCS      3$
922 007116      DELAY      250
          MOV      #250,(PC)+
          .WORD      0
          MOV      L$DLY,(PC)+
          .WORD      0
          DEC      -6(PC)
          BNE      -4
          DEC      -22(PC)
          BNE      -20
923 007146 005303      DEC      R3
924 007150 001357      BNE      2$
925 007152 017500 002524      3$:      MOV      @TSSR(R5),R0      ;GET THE TSSR REGISTER
926 007156 010004      MOV      R0,R4      ;TSSR CONTENTS
927 007160 042704 176277      BIC      #1<TS.A17!TS.A16!TS.OFL>,R4
928 007164 052704 002200      BIS      #TS.SSR!TS.NBA,R4      ;R4 HAS EXPECTED CONTENTS
929 007170 020400      CMP      R4,R0      ;ONLY EXPECTED BITS SET ?
930 007172 001402      BEQ      5$      ;BRANCH IF OKAY
931 007174 000241      CLC      ;CLEAR THE CARRY FOR ERROR
932 007176 000401      BR      10$      ;GO TO EXIT
933 007200 000261      5$:      SEC      ;SET THE CARRY BIT
934 007202 000207      10$:      RTS      PC      ;RETURN TO CALLER
935
936      ;
937      ; SUBROUTINE TO WAIT FOR THE SUBSYSTEM READY FLAG
938      ;
939      ; INPUTS:
940      ;
941      ;      R5      CURRENT UNIT NUMBER
942      ;
943      ;
944      ; OUTPUTS:
945      ;
946      ;      R0      CONTENTS OF LAST TSSR READ
947      ;      CARRY    SET - READY BIT SET
948      ;      CLR      CLR - TIMEOUT WAITING FOR READY
949      ;
950 007204      WAITF:: BREAK      ; DO A SUPVSR BREAK FIRST.
          TRAP      C$BRK
          007204 104422
951 007206 012746 005670      MOV      #3000.,-(SP)      ; 300 MSEC TIMER.
952 007212 017500 002524      2$:      MOV      @TSSR(R5),R0      ;READ THE TSSR REGISTER
953 007216 105700      TSTB      R0      ;TEST FOR READY BIT SET
954 007220 100420      BMI      3$      ; EXIT ON STOP FLAG.
955 007222      DELAY      25      ; WAIT
          MOV      #25,(PC)+
          007222 012727 000025

```



```

1006 007326 032763 000100 000012      BIT      @X2.BFE,MS.XS2(R3)      ;BUFFER ENABLE SWITCH SET
1007 007334 001402                      BEQ      50$              ;BR, IF SWITCH NOT SET
1008 007336 005267 172762              INC      BENBSW          ;SET SOFTWARE SWITCH FOR ENABLED
1009 007342                      50$:
1010 007342 000261                      55$: SEC                  ;SET CARRY NO TROUBLE
1011 007344 000401                      BR      70$              ;EXIT
1012 007346 000241                      60$: CLC                  ;CARRY CLEAR = ERROR
1013 007350 017500 002524              70$: MOV      @TSSR(R5),R0      ;RETURN TSSR CONTENTS
1014 007354 000207                      RTS      PC                ;RETURN
1015
1016      ;+
1017      ;
1018      ;ROUTINE TO CHECK WRITE LOCK CONDITION
1019      ;
1020      ;INPUT:
1021      ;
1022      ;      R4      ADDRESS OF COMMAND PACKET
1023      ;      R5      CURRENT UNIT NUMBER
1024      ;
1025      ;-
1026 007356      WLKCHK::
1027 007356 010475 002514              10$: MOV      R4,@TSD8(R5)      ;SEND C T COMMAND
1028 007362 004767 177616              JSR      PC,WAITF          ;WAIT FOR SSR
1029 007366 103401                      BCS      40$              ;BR, IF SSR IS SET AND OK
1030 007370 000420                      BR      60$              ;BR IF TROUBLE CARRY = CLEAR
1031 007372 005724                      40$: TST      (R4)+          ;STEP IT
1032 007374 011402                      MOV      (R4),R2          ;POINT TO WRT CHARA DATA PACKET
1033 007376 011203                      MOV      (R2),R3          ;GET ADDRESS OF MESSAGE BUFFER
1034 007400 032763 000004 000006      BIT      @X0.WLK,MS.XS0(R3)      ;IS UNIT WRITE LOCKED?
1035 007406 001407                      BEQ      55$              ;NO,PROCEED WITH TESTING
1036 007410                      ERRHRD 1,UNIWLK          ;TAPE IS WRITE LOCKED
1037 007410 104456                      TRAP      C$ERRHRD
1038 007412 000001                      .WORD    1
1039 007414 005653                      .WORD    UNIWLK
1040 007416 000000                      .WORD    0
1041 007420 004767 007554              JSR      PC,DROPU          ;DROP IT
1042 007424 000402                      BR      60$              ;EXIT WITH CARRY=0
1043 007426 000261                      55$: SEC                  ;SET CARRY NO TROUBLE
1044 007430 000401                      BR      70$              ;EXIT
1045 007432 000241                      60$: CLC                  ;CARRY CLEAR = ERROR
1046 007434 000207                      70$: RTS      PC                ;RETURN
1047
1048      ;+
1049      ;
1050      ;ROUTINE TO ISSUE A WRITE CHARACTERISTICS COMMAND
1051      ;
1052      ;INPUT:
1053      ;
1054      ;      R4      ADDRESS OF COMMAND PACKET
1055      ;      R5      CURRENT UNIT NUMBER
1056      ;      REQUIRES A CALL TO SOFINIT BE DONE PREVIOUSLY
1057      ;
1058      ;OUTPUT:
1059      ;
1060      ;      R0      TSSR CONTENTS

```

```

1059      ;      CARRY SET WRITE CHARACTERISTICS COMMAND OK
1060      ;      CLR  WRITE CHARACTERISTICS FAILED
1061      ;
1062      ;IMPLICIT OUTPUT:
1063      ;
1064      ;
1065      ;
1066      ;SIDE EFFECTS:
1067      ;
1068      ;
1069      ;-
1070
1071 007436      WRCHR::
1072 007436 010475 002514 10$: MOV R4, @TSDB(R5) ;SEND OUT COMMAND
1073 007442 004767 177536      JSR PC, WAITF ;WAIT FOR SSR
1074 007446 103401      BCS 50$ ;BR, IF SSR IS SET AND OK
1075 007450 000402      BR 60$ ;BR IF TROUBLE CARRY = CLEAR
1076 007452      50$: SEC ;SET CARRY NO TROUBLE
1077 007452 000261      BR 70$ ;EXIT
1078 007454 000401      60$: CLC ;CARRY CLEAR = ERROR
1079 007456 000241      70$: MOV @TSSR(R5), R0 ;RETURN TSSR CONTENTS
1080 007460 017500 002524      RTS PC ;RETURN
1081 007464 000207
1082
1083
1084      ;+
1085      ;
1086      ;ROUTINE TO DO SET UP OF RUNNING CONDITIONS
1087      ;
1088      ;INPUTS:
1089      ;
1090      ;      R5      CURRENT UNIT NUMBER
1091      ;
1092      ;
1093      ;OUTPUTS:
1094      ;
1095      ;
1096      ;CALLING SEQUENCE:
1097      ;      JSR PC, FIRSTU
1098      ;      JSR PC, SOFINIT
1099      ;      BCS CONTINUE
1100      ;      ERDF ;REPORT FATAL ERROR
1101      ;      JSR PC, MDSET
1102      ;
1103      ;-
1104
1105 007466      MDSET:: BREAK ; DO A SUPVSR BREAK FIRST.
1106 007466 104422      JSR PC, SETDEF ;RESTORE DEFAULT TRAP C$BRK
1107 007470 004767 000272      JSR PC, WLKCHK ;CHECK WRITE LOCK
1108 007474 004767 177656      BCS 1$ ;C=1 IS O.K.
1109 007500 103416      DELAY 1 ;WAIT
1110 007502      MOV #1, (PC)+
1111 007506 000000      .WORD 0
1112 007510 016727 172402      MOV L$DLY, (PC)+
1113 007514 000000      .WORD 0
1114 007516 005367 177772      DEC -6(PC)

```

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
------	---------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

```

1162 007732 012767 002506 172372      MOV    @WSMBK,CMDPKT.CP.ADL      ;MSG BUF ADDR
1163 007740 012767 000006 172370      MOV    @6,CMDPKT.CP.CNT      ;BYTE COUNT
1164 007746 012767 100010 172532      MOV    @100010,WSMBK      ;INVERT THE SWITCH
1165 007754 012704 002330              MOV    @CMDPKT,R4              ;
1166 007760 004767 177452              JSR    PC,WRTCHR      ;DO IT
1167 007764 000207                    RTS    PC      ;RETURN
1168
1169
1170      ; SUBROUTINE TO SETUP DEFAULT SET CHAR CMD
1171      ;
1172      ; INPUTS:
1173      ;
1174      ;
1175      ; OUTPUTS:
1176      ;
1177      ;
1178      ; R4      ADDRESS OF COMMAND PACKET
1179
1180      SETDEF::
1181 007766 012701 140004              MOV    @SCH,R1      ;WRITE CHAR CMD
1182 007772 010167 172332              MOV    R1,CMDPKT.CP.CMD      ;SET UP COMMAND
1183 007776 012767 002474 172326      MOV    @SCHBY,CMDPKT.CP.ADL      ;SET UP ADR LO TO POINT TO MSG BUF(MSGPK0)
1184 010004 012767 000012 172324      MOV    @SCHCNT,CMDPKT.CP.CNT      ;SET BUFFER EXTENT
1185 010012 012767 000040 172462      MOV    @DFTSCH,SCHBK*6      ;STORE CHARACTERISTIC CODE IN SCH BLOCK.
1186 010020 016767 173506 172456      MOV    TSUNT,SCHBK*10      ;UNIT #
1187 010026 012704 002330              MOV    @CMDPKT,R4      ;ADDRESS OF CMD PACKET
1188 010032 000207                    RTS    PC      ;RETURN
1189
1190
1191      ; MODULES TO HANDLE TS05 INTERRUPTS.
1192
1193
1194 010034      BGNSRV    TSSINO
1195 010034      TSSINO::
1196 010034 005267 173432      INC    INTFLG      ;SET INTERRUPT OCCURRED FLAG.
1197 010040      ENDSRV
1198 010040      L10004:
1199 010040 000002                                RTI
1200
1201 010042      BGNSRV    TSSIN1
1202 010042      TSSIN1::
1203 010042 005267 173426      INC    INTFLG*2      ;SET INTERRUPT OCCURRED FLAG.
1204 010046      ENDSRV
1205 010046      L10005:
1206 010046 000002                                RTI
1207
1208 010050      BGNSRV    TSSIN2
1209 010050      TSSIN2::
1210 010050 005267 173422      INC    INTFLG*4      ;SET INTERRUPT OCCURRED FLAG.
1211 010054      ENDSRV
1212 010054      L10006:
1213 010054 000002                                RTI
1214
1215 010056      BGNSRV    TSSIN3
1216 010056      TSSIN3::
1217 010056 005267 173416      INC    INTFLG*6      ;SET INTERRUPT OCCURRED FLAG.
1218 010062      ENDSRV

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0065

```

010062          000002          110007:
010062          000002          RTI
1209
1210          :      SUBROUTINE TO RETRIEVE RECORD COUNT READ FROM TAPE FOR ERROR
1211          :      PRINTS.
1212          :      INPUTS:
1213          :      OUTPUTS: R3 = RECORD COUNT READ
1214          :      REGISTERS: R2, R3, R4
1215          :      CALLS:
1216
1217 010064 032767 000400 173326 RECTAP: BIT      @MOD.CO.CMDWRD          ;READ REV FETCH
1218 010072 001430          BEQ      50001$
1219 010074 016702 172260          MOV      MSGPKT.MS.RFC,R2          ;FIND LAST READ AD.
1220 010100 066702 173304          ADD      DATARD,R2
1221 010104 032702 000001          BIT      @BIT00,R2          ;ODD AD., REASSEMBLE
1222 010110 001417          BEQ      50002$
1223 010112 005202          INC      R2          ;REC COUNT STARTING
1224 010114 111203          MOV8     (R2),R3          ;WITH UPPER BYTE FETCH
1225 010116 142703 177400          BICB     @177400,R3
1226 010122 000303          SWAB     R3
1227 010124 005302          DEC      R2          ;LET R2 := R2 - #1          ;LOWER BYTE AD.
1228 010126 105767 173366          TSTB     SWBFLG          ;IFB SWBFLG NE #0 THEN
1229 010132 001401          BEQ      50003$
1230 010134 005302          DEC      R2          ;LET R2 := R2 - #1          ;LOWER BYTE AD. ON SWAP
1231
1232          50003$:
1233 010136 111204          MOV8     (R2),R4          ;FETCH LOWER BYTE
1234 010140 142704 177400          BICB     @177400,R4
1235 010144 050403          BIS      R4,R3
1236 010146 000401          BR       50004$
1237 010150          50002$:
1238 010150 011203          MOV      (R2),R3          ;LET R3 := (R2)          ;EVEN AD. FETCH
1239 010152          50004$:
1240 010152 000402          BR       50005$
1241 010154          50001$:
1242 010154 017703 173230          MOV      @DATARD,R3          ;LET R3 := @DATARD          ;READ FWD FETCH
1243
1244          50005$:
1245 010160 000207          RTS      PC
1246
1247          :      SUBROUTINE TO STORE A SET CHARACTERISTIC COMMAND AS
1248          :      THE FIRST ENTRY IN THE SEQUENCE TABLE.
1249          :      INPUTS:
1250          :      OUTPUTS:
1251          :      REGISTERS:
1252          :      CALLS:
1253
1254          SETCH::
1255 010162 012701 003540          MOV      @CMDSEQ,R1          ;INIT CMD SEQUENCE TABLE POINTER.
1256 010166 012721 140004          MOV      @SCH,(R1)+          ;THIS CODE SETS UP A SET CHARACTERISTIC
1257 010172 012721 000040          MOV      @DFTSCH,(R1)+          ;COMMAND AS THE FIRST COMMAND IN THE
1258 010176 012721 000001          MOV      @1,(R1)+          ;SEQUENCE TABLE.
1259 010202 005721          TST      (R1)+          ;SKIP PATTERN LOCATION.
1260 010204 000207          RTS      PC
1261
1262          :      SUBROUTINE TO STORE A REWIND COMMAND IN THE SEQUENCE TABLE
1263          :      INPUTS:

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SF0 0066

```

1264      ;      OUTPUTS:
1265      ;      REGISTERS:
1266      ;      CALLS:
1267
1268 010206 012721 102010      SETRW:: MOV      @RWD,(R1).      ;CMD = REWIND.
1269 010212 012721 000001      MOV      #1,(R1).      ;BRF.
1270 010216 012721 000001      MOV      #1,(R1).      ;# OF OPERATIONS.
1271 010222 005721      TST      (R1).      ;SKIP PATTERN.
1272 010224 000207      RTS      PC      ;RETURN
1273
1274      ;      SUBROUTINE TO EXECUTE ALL COMMANDS IN THE SEQUENCE TABLE ON ALL
1275      ;      DEVICES.
1276      ;      INPUTS:
1277      ;      OUTPUTS:      R2 = TERMINATION INDICATOR (0=END OF TABLE,1=EOT)
1278      ;      REGISTERS:
1279      ;      CALLS:      CMDAC,SETUP,EXSUB,CKHAE,NEXTU,FIRSTU,VFYDAT.
1280
1281 010226 012701 003540      EXALL:: MOV      @CMDSEQ,R1      ;INIT SEQUENCE TABLE POINTER.
1282 010232      500068:      CMP      (R1),#END      ;WHILE THERE ARE CMDS IN THE SEQUENCE TABLE.
1283 010232 021127 177777      BEQ      500078
1284 010236 001530      JSR      PC,SETUP      ;GO SETUP THE COMMAND BLOCK.
1285 010240 004767 000726      500104: BREAK      ;DO A SUPVSR BREAK FIRST.
1286 010244      010244 104422      TRAP      C$BRK
1287 010246 026767 173140 173140      CMP      NCNT,NCNT1      ;WHILE THERE ARE RECORDS REMAINING:
1288 010254 002116      BGE      500118
1289 010256 004767 000602      JSR      PC,CMDAC      ;STORE CMD ASCII IN ERROR MESSAGE.
1290 010262 105767 173227      TSTB      RANDOM      ;IF IN RANDOM MODE:
1291 010266 001435      BEQ      500128
1292 010270 026727 173124 104005      CMP      CMDWRD,@WRT      ;IF CMD IS A WRITE THEN:
1293 010276 001031      BNE      500138
1294 010300 105767 173212      TSTB      VFYFLG      ;IF DATA IS NOT TO BE VERIFIED THEN:
1295 010304 001026      BNE      500148
1296 010306 066767 173122 173116      ADD      RANS,RANB      ;LET RANB := RANB + RANS ;GENERATE
1297 010314 066767 173112 173112      ADD      RANB,RANS      ;LET RANS := RANS + RANB ;RANDOM
1298 010322 016767 173106 173066      MOV      RANS,BRFCNT      ;LET BRFCNT := RANS ;LENGTH
1299 010330 046767 173074 173060      BIC      LENMSK,BRFCNT      ;MASK RANDOM LENGTH.
1300 010336 026727 173054 000022      CMF      BRFCNT,#18.      ;DO NOT ALLOW BYTE COUNT OF LESS THAN 18
1301 010344 002003      BGE      500158
1302 010346 012767 000022 173042      MOV      #18.,BRFCNT      ;CHANGE COUNT OF 0-17 TO 18.
1303
1304 010354      500158:
1305 010354 016767 173036 171754      MOV      BRFCNT,CMDPKT+CP.CNT      ;MOVE BRF TO CMD PACKET.
1306
1307 010362      500148:
1308
1309 010362      500138:
1310
1311 010362      500128:
1312 010362 004767 000136      JSR      PC,EXSUB      ;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
1313 010366 004767 007106      JSR      PC,CKHAE      ;CHECK HALT AFTER EACH CMD FLAG.
1314 010372 012702 000001      MOV      #1,R2      ;LET R2 := #1 ;SET ALL UNITS AT BOT/EOT.
1315 010376 004767 006500      JSR      PC,FIRSTU      ;FIND FIRST UNIT.
1316
1317 010402      500168:
1318 010402 026527 002604 177777      CMP      DEVTBL(R5),#END      ;WHILE THERE ARE MORE UNITS:
1319 010410 001426      BEQ      500178

```

```

1320 010412 032767 000400 173000      BIT      #MOD.CO,CMDWRD      ;IF CMD IS REVERSE THEN:
1321 010420 001406                      BEQ      50020$
1322 010422 032765 000002 003502      BIT      @X0.BOT,EOTFLG(R5)      ;IF NOT AT BOT THEN:
1323 010430 001001                      BNE      50021$
1324 010432 005002                      CLR      R2              ;LET R2 := #0 ;CLEAR EOT/BOT FLAG.
1325
1326 010434                      50021$:
1327 010434 000411                      BR      50022$              ;ELSE IF CMD IS NOT REVERSE:
1328 010436                      50020$:
1329 010436 032765 000001 003502      BIT      @X0.EOT,EOTFLG(R5)
1330 010444 001404                      BEQ      50023$
1331 010446 032767 000001 172744      BIT      #CMD.CO,CMDWRD
1332 010454 001001                      BNE      50024$
1333 010456                      50023$:
1334
1335 010456 005002                      CLR      R2              ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
1336
1337 010460                      50024$:
1338
1339 010460                      50022$:
1340 010460 004767 006464                      JSR      PC,NEXTU              ;FIND NEXT UNIT
1341 010464 000746                      BR      50016$
1342 010466                      50017$:
1343 010466 020227 000001                      CMP      R2,#1              ;IF ALL UNIT ARE AT EOT/BOT THEN:
1344 010472 001001                      BNE      50025$
1345 010474 000412                      BR      EXARTN              ;RETURN WITH R2 = #1.
1346
1347 010476                      50025$:
1348 010476 005267 172710                      INC      NCNT              ;LET NCNT := NCNT + #1      ;UPDATE RECORD COUNT.
1349 010502 016767 172712 172714                      MOV      CMDWRD,PCMDWD      ;SAVE PREVIOUS COMMAND WORD.
1350
1351 010510 000655                      BR      50010$
1352 010512                      50011$:
1353 010512 004767 005350                      JSR      PC,VFYDAT              ;IF LAST CMD WAS A WRITE VERIFY, THEN GO
1354
1355
1356 010516 000645                      BR      50006$
1357 010520                      50007$:
1358 010520 005002                      CLR      R2              ;LET R2 := #0      ;SET NORMAL RETURN INDICATOR.
1359 010522 000207                      EXARTN: RTS PC              ;RETURN.
1360
1361
1362
1363      ;      SUBROUTINE TO ISSUE COMMAND TO ALL DEVICES, WAIT FOR
1364      ;      ALL INTERRUPTS, AND CHECK ALL STATUS.
1365      ;      INPUTS:
1366      ;      OUTPUTS:
1367      ;      REGISTERS:
1368      ;      CALLS:      EXCUTE,GOWAIT,NEXTU,FIRSTU.
1369
1370 010524 004767 006352                      EXSUB:: JSR      PC,FIRSTU              ;SET UP FOR FIRST UNIT.
1371 010530                      50026$:
1372 010530 026527 002604 177777                      CMP      DEVTBL(R5),#END      ;WHILE THERE ARE MORE DEVICES:
1373 010536 001465                      BEQ      50027$
1374 010540 032767 000400 172652                      BIT      #MOD.CO,CMDWRD      ;IF CMD IS REVERSE THEN:
1375 010546 001421                      BEQ      50030$
1376 010550 032765 000002 003502                      BIT      @X0.BOT,EOTFLG(R5)      ;IF NOT AT BOT

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0068

1377	010556	001014			BNE	50031:		
1378	010560	032765	000001	003502	BIT	0X0.EOT,EOTFLG(R5)	;BUT IF AT EOT	
1379	010566	001406			BEQ	50032:		
1380	010570	105767	172730		TSTB	ALLEOT	;AND ALL OTHERS AT EOT	
1381	010574	001402			BEQ	50033:		
1382	010576	004767	001252		JSR	PC,EXECUTE	;THEN EXECUTE REV CMD	
1383							;IF NOT ALL AT EOT, FREEZE UNIT(S) AT EOT	
1384	010602				50033:			
1385	010602	000402			BR	50034:	;IF NOT AT BOT AND	
1386	010604				50032:			
1387	010604	004767	001244		JSR	PC,EXECUTE	;NOT AT EOT, EXEC REV CMD	
1388								
1389	010610				50034:			
1390								
1391	010610				50031:			
1392	010610	000435			BR	50035:	;ELSE IF CMD IS NOT REVERSE:	
1393	010612				50030:			
1394	010612	026727	172610	000002	CMP	CMDLG,#2		
1395	010620	001011			BNE	50036:		
1396	010622	032765	000002	003502	BIT	0X0.BOT,EOTFLG(R5)		
1397	010630	001405			BEQ	50036:		
1398							;CLEAR BAD SPOT COUNTS WHEN WRITING FROM BOT	
1399	010632	016567	002616	172652	MOV	BTADDR(R5),BTPT	;LET BTPT := BTADDR(R5)	
1400	010640	005077	172646		CLR	BTPT	;LET BTPT := #0	
1401								
1402	010644				50036:			
1403	010644	032765	000001	003502	BIT	0X0.EOT,EOTFLG(R5)		
1404	010652	001404			BEQ	50037:		
1405	010654	032767	000001	172536	BIT	0CMD.CO,CMDWRD		
1406	010662	001003			BNE	50040:		
1407	010664				50037:			
1408							;IF NOT AT EOT OR NOT A MOTION CMD THEN:	
1409	010664	004767	001164		JSR	PC,EXECUTE	;ISSUE CMD TO TS05	
1410								
1411	010670	000405			BR	50041:		
1412	010672				50040:			
1413	010672	105767	172626		TSTB	ALLEOT	;IFB ALLEOT NE #0 THEN	
1414	010676	001402			BEQ	50042:		
1415	010700	004767	001150		JSR	PC,EXECUTE		
1416								
1417	010704				50042:			
1418								
1419	010704				50041:			
1420								
1421	010704				50035:			
1422	010704	004767	006240		JSR	PC,NEXTU	;FIND NEXT UNIT IN TEST CYCLE.	
1423								
1424	010710	000707			BR	50026:		
1425	010712				50027:			
1426	010712	105767	172601		TSTB	RPTFLG	;IF REPORT HAS BEEN REQUESTED THEN:	
1427	010716	001403			BEQ	50043:		
1428	010720	105067	172573		CLRB	RPTFLG	;CLR THE FLAG,	
1429	010724				DORPT		;PRINT THE PERFORMANCE REPORT.	
1430	010726						TRAP	C\$DRPT
1431	010726	004767	006150		50043:			
1432	010732				JSR	PC,FIRSTU	;SET UP FOR FIRST UNIT.	
					50044:			

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
------	---------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0070

```

1490 011064 016704 172330 CMDAC:: MOV CMDWRD,R4;LET R4 := CMDWRD ;R4 = CMD BINARY.
1491 011070 004767 000042 JSR PC,GCMDA ;GET CMD ASCII.
1492 011074 112367 175340 MOVB (R3)+,STAER1+2 ;MOVE CMD ASCII
1493 011100 112367 175335 MOVB (R3)+,STAER1+3 ;
1494 011104 111367 175332 MOVB (R3),STAER1+4 ;INTO MSG.
1495 011110 016704 172310 MOV PCMDWD,R4 ;R4 = PREVIOUS CMD BINARY.
1496 011114 004767 000016 JSR PC,GCMDA ;GET CMD ASCII.
1497 011120 112367 175430 MOVB (R3)+,STAER7+24 ;MOVE CMD ASCII
1498 011124 112367 175425 MOVB (R3)+,STAER7+25 ;
1499 011130 111367 175422 MOVB (R3),STAER7+26 ;INTO MSG.
1500 011134 000207 RTS PC ;RETURN. GO EXECUTE NEXT FUNCTION.
1501
1502
1503 ; SUBROUTINE TO FIND THE ASCII EQUIVALENT OF THE COMMAND IN R4.
1504 ; ADDRESS OF ASCII 1ST WORD IS RETURNED IN R3.
1505 ; INPUTS: R4 = PRESENT COMMAND WORD.
1506 ; OUTPUTS: R3 = ADDRESS OF PRESENT COMMAND ASCII.
1507 ; REGISTERS:
1508 ; CALLS:
1509
1510 011136 005003 GCMDA:: CLR R3;LET R3 := #0 ;INIT CMD TBL POINTER.
1511 011140 500604: CMP CMDTBL(R3),R4 ;UNTIL CURRENT CMD IS FOUND:
1512 011140 026304 003752 BEQ 500614
1513 011144 001403 ADD #2,R3 ;LET R3 := R3 + #2 ;SEARCH CMD TABLE.
1514 011146 062703 000002 BR 500604
1515 011152 000772
1516 011154 500614: MOV R3,R4 ;LET R4 := R3
1517 011154 010304 ASR R3 ;POINT TO ASCII FOR THAT COMMAND
1518 011156 006203 NOP
1519 011160 000240 ADD R4,R3
1520 011162 060403 ADD #CMDASC,R3
1521 011164 062703 004040 RTS PC ;RETURN.
1522 011170 000207
1523
1524 ; THIS SUBROUTINE LOADS THE TS05 COMMAND PACKET FROM ONE
1525 ; ENTRY IN THE SEQUENCE TABLE.
1526 ; INPUTS:
1527 ; OUTPUTS:
1528 ; REGISTERS: R2, R3.
1529 ; CALLS: GENPAT.
1530
1531 011172 005067 172230 SETUP:: CLR CMDLG ;CLR CMD LOGGING CODE(DISABLES LOGGING)
1532 011176 012167 171126 MOV (R1)+,CMDPKT ;LOAD THE COMMAND WORD.
1533 011202 011167 171130 MOV (R1),CMDPKT+CP.CNT ;LOAD THE BYTE/RECORD/FILE COUNT.
1534 011206 011167 172204 MOV (R1),BRFCNT ;SAVE BRFCNT FOR THIS COMMAND.
1535 011212 016702 171112 MOV CMDPKT,R2 ;GET CMD.
1536 011216 042702 177740 BIC #NCMD.C,R2 ;CLR ALL BUT CMD BITS.
1537 011222 010203 MOV R2,R3 ;SAVE IT TWICE.
1538 011224 162703 000010 SUB #CMD.C3,R3 ;POSITION COMMAND?
1539 011230 001003 BNE 24 ;BR IF NOT.
1540 011232 011167 171074 MOV (R1),CMDPKT+2 ;MOVE BPCR IN 2ND PKT WORD FOR POSITION CMD.
1541 011236 000464 BR 34
1542 011240 026727 171064 100011 24: CMP CMDPKT,#WTM ;IF CMD IS A WRITE TAPE MARK THEN:
1543 011246 001003 BNE 500624
1544 011250 012767 000002 172150 MOV #2,CMDLG ;WTM LOGGING CODE IS 2.
1545
1546 011256 500624:

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0071

1547	011256	010203			MOV	R2,R3		
1548	011260	162703	000001		SUB	#CMD.C0,R3		;IS IT A READ?
1549	011264	001017			BNE	1\$		;BR IF NOT.
1550	011266	016767	172116	171036	MOV	DATARD,CMDPKT+CP.ADL		;IF SO, LOAD THE BUFFER ADDR.
1551	011274	032767	000400	171026	BIT	#MOD.C0,CMDPKT		;IF CMD IS A READ REV THEN:
1552	011302	001404			BEQ	50063\$		
1553	011304	012767	000004	172114	MOV	#4,CMDLG		;LOGGING CODE IS 4.
1554								;ELSE - IF CMD IS A READ FWD:
1555	011312	000403			BR	50064\$		
1556	011314							
1557	011314	012767	000006	172104	MOV	#6,CMDLG		;LOGGING CODE IS 6.
1558								
1559	011322							
1560	011322	000432			BR	3\$		;CONTINUE.
1561	011324	010203			MOV	R2,R3		;IS IT
1562	011326	162703	000004		SUB	#CMD.C2,R3		;A SET CHARACTERISTICS CMD?
1563	011332	001014			BNE	4\$		;BR IF NOT.
1564	011334	012767	002474	170770	MOV	#SCHBK,CMDPKT+CP.ADL		;SET UP ADR LO FOR SET CHAR.
1565	011342	012767	000012	170766	MOV	#SCHCNT,CMDPKT+CP.CNT		;SET BUFFER EXTENT
1566	011350	011167	171126		MOV	(R1),SCHBK+6		;STORE CHARACTERISTIC CODE IN SCH BLOCK.
1567	011354	016767	172152	171122	MOV	TSUNT,SCHBK+10		;UNIT #
1568	011362	000412			BR	3\$		;CONTINUE.
1569	011364	010203			MOV	R2,R3		;IS IT
1570	011366	162703	000006		SUB	#CMD.C1!CMD.C2,R3		;A DIAGNOSTIC (DIA) CMD?
1571	011372	001006			BNE	3\$		;BR IF NOT.
1572	011374	012767	000020	170734	MOV	#DIACNT,CMDPKT+CP.CNT		;LOAD BUFFER EXTENT.
1573	011402	012767	003406	170722	MOV	#DIABLK,CMDPKT+CP.ADL		;LOAD BUFFER ADR LOW.
1574	011410	005721			TST	(R1)+		;POINT TO N (NUMBER OF TIMES TO EXECUTE THIS INS
1575	011412	012167	171776		MOV	(R1)+,NCNT1		;SAVE NUMBER OF OPERATIONS
1576	011416	005067	171770		CLR	NCNT		;CLEAR OPERATION COUNTER.
1577	011422	012167	172020		MOV	(R1)+,PATERN		;SAVE PATTERN CODE FOR CURRENT CMD.
1578	011426	010203			MOV	R2,R3		;IS IT
1579	011430	162703	000005		SUB	#CMD.C0!CMD.C2,R3		;A WRITE?
1580	011434	001010			BNE	5\$		;BR IF NOT.
1581	011436	016767	171744	170666	MOV	DATAWT,CMDPKT+CP.ADL		;LOAD WRITE BUFFER LO ORDER.
1582	011444	004767	000106		JSR	PC,GENPAT		;GO GENERATE THE WRITE PATTERN.
1583	011450	012767	000002	171750	MOV	#2,CMDLG		;WRITE LOGGING CODE IS 2.
1584	011456	032767	000100	170644	BIT	#VFY.C,CMDPKT		;IF DATA VERIFICATION IS REQUIRED:
1585	011464	001407			BEQ	50065\$		
1586	011466	112767	000001	172022	MOVB	#1,VFYFLG		;SET VERIFY FLAG.
1587	011474	042767	000100	170626	BIC	#VFY.C,CMDPKT		;CLEAR VERIFY BIT(NOT USED BY HARDWARE).
1588								;IF DATA VERIFICATION IS NOT REQUIRED:
1589	011502	000402			BR	50066\$		
1590	011504							
1591	011504	105067	172006		CLRB	VFYFLG		;CLR VERIFY FLAG.
1592								
1593	011510							
1594	011510	016767	171704	171706	MOV	CMDWRD,PCMDWD		;SAVE PREVIOUS CMD WORD.
1595	011516	016767	170606	171674	MOV	CMDPKT,CMDWRD		;SAVE PRESENT CMD WORD.
1596	011524	105767	171770		TSTB	SWBFLG		;IF SWAP BYTES IS ENABLED:
1597	011530	001403			BEQ	50067\$		
1598	011532	052767	010000	170570	BIS	#SWB.C,CMDPKT		;SET SWAP BIT IN COMMAND.
1599								
1600	011540							
1601	011540	042767	004000	170562	BIC	#BRF.C,CMDPKT		;CLR BRF BIT (INTERNAL ONLY).
1602	011546	016767	170556	171646	MOV	CMDPKT,CMDSAV		;SAVE 1ST WORD OF COMMAND PACKET.
1603	011554	000207			RTS	PC		;RETURN.

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0072

```

1604
1605      ;      THIS SUBROUTINE SETS UP AND CALLS THE APPROPRIATE SUBROUTINE TO GENERATE
1606      ;      THE DESIRED PATTERN FOR THE WRITE AND WRITE/VERIFY COMMANDS.
1607      ;      INPUTS:
1608      ;      OUTPUTS:
1609      ;      REGISTERS:      R2, R3, R4.
1610      ;      CALLS:      PATRO  PATR7
1611
1612 011556 016703 171664  GENPAT: MOV    PATERN,R3      ;SETUP PATTERN ROUTINE POINTER
1613 011562 006303      ASL      R3
1614 011564 016704 171626  MOV    BRFCNT,R4      ;SET LENGTH OF WRITE BFR
1615 011570 005204      INC      R4
1616 011572 042704 000001  BIC     #1,R4      ;ROUNDED UP TO NEXT WORD
1617 011576 162704 000002  SUB     #2,R4      ;WITH FIRST WORD RESERVED
1618 011602 016702 171600  MOV    DATAW,R2      ;FOR RECORD COUNT
1619 011606 062702 000002  ADD     #2,R2
1620 011612 004773 011620  JSR     PC,@PATTBL(R3) ;GO GENERATE THE APPROPRIATE PATTERN.
1621 011616 000207      RTS      PC      ;RETURN TO SETUP SUBROUTINE.
1622
1623      ;TS05 WRITE PATTERN LOOKUP TABLE. USED TO JSR TO THE
1624      ;CORRECT DATA PATTERN GENERATING ROUTINE.
1625
1626 011620 011642  PATTBL: PATRO
1627 011622 011700      PATR1
1628 011624 011720      PATR2
1629 011626 011730      PATR3
1630 011630 011754      PATR4
1631 011632 011766      PATR5
1632 011634 012000      PATR6
1633 011636 012020      PATR7
1634 011640 012052      PATR8
1635
1636      ;INCREMENTING PATTERN. 0 377.
1637
1638 011642 012703 000400  PATRO: MOV    #400,R3;LET R3 := #400
1639 011646 162704 000002  1$: SUB     #2,R4;LET R4 := R4 - #2 ;DECREMENT WORD COUNT.
1640 011652 100411      BMI     2$      ;BR IF DONE.
1641 011654 010322      MOV     R3,(R2)+ ;STORE DATA WORD.
1642 011656 062703 001002  ADD     #1002,R3 ;UPDATE PATTERN.
1643 011662 020327 001000  CMP     R3,#1000 ;IF PATTERN HAS WRAPPED AROUND THEN:
1644 011666 001002      BNE     50070$
1645 011670 012703 000400  MOV     #400,R3 ;INIT THE PATTERN AGAIN.
1646
1647 011674      50070$: BR      1$      ;DO IT AGAIN.
1648 011674 000764
1649
1650 011676 000207  2$: RTS      PC      ;RETURN.
1651
1652      ;ALL ONE'S PATTERN.
1653
1654 011700 012703 177777  PATR1: MOV    #1,R3 ;ALL ONES PATTERN;.
1655 011704 162704 000002  ZROPAT: SUB     #2,R4 ;DECREMENT BYTE COUNT.
1656 011710 100402      BMI     1$      ;DONE?,BR IF YES.
1657 011712 010322      MOV     R3,(R2)+ ;IF NOT LOAD NEXT BYTE WITH PATTERN.
1658 011714 000773      BR      ZROPAT ;DO IT AGAIN.
1659
1660 011716 000207  1$: RTS      PC      ;RETURN.

```

GLOBAL AREAS MACRO M1113 14-JUN-84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0073

```

1661
1662 ;ALL ZEROES PATTERN.
1663
1664 011720 005003 PATR2:: CLR R3 ;CLR PATTERN REGISTER.
1665 011722 004767 177756 JSR PC,ZROPAT ;GO GENERATE IT.
1666 011726 000207 RTS PC ;RETURN.
1667
1668 ;ONE BIT WALKING FROM R TO L IN A FIELD OF ZEROES.
1669
1670 011730 012703 000401 PATR3:: MOV #401,R3 ;INIT PATTERN REGISTER.
1671 011734 162704 000002 WLKZRO: SUB #2,R4;LET R4 := R4 - #2 ;DECREMENT WORD COUNT.
1672 011740 100404 BMI 1$ ;BR IF DONE.
1673 011742 010322 MOV R3,(R2)+ ;LOAD DATA.
1674 011744 006303 ASL R3 ;SHIFT PATTERN.
1675 011746 005503 ADC R3 ;ADD CARRY BACK INTO PATTERN.
1676 011750 000771 BR WLKZRO ;DO IT AGAIN.
1677 011752 000207 1$: RTS PC ;RETURN.
1678
1679 ;ZERO BIT WALKING FROM R TO L IN A FIELD OF 1'S.
1680
1681 011754 012703 177376 PATR4:: MOV #177376,R3 ;INIT PATTERN REGISTER.
1682 011760 004767 177750 JSR PC,WLKZRO ;GO GENERATE ;IT.
1683 011764 000207 RTS PC ;RETURN.
1684
1685 ;ALTERNATING ONE AND ZERO BITS WITH ALTERNATE BYTES
1686 ;COMPLEMENTED.
1687
1688 011766 012703 125125 PATR5:: MOV #125125,R3 ;INIT PATTERN REGISTER.
1689 011772 004767 177706 JSR PC,ZROPAT ;GO GENERATE IT.
1690 011776 000207 RTS PC ;RETURN.
1691
1692 ;ALTERNATING BYTES OF 000 AND 377.
1693
1694 012000 012703 177400 PATR6:: MOV #177400,R3 ;INIT PATTERN REGISTER.
1695 012004 162704 000002 1$: SUB #2,R4 ;DECREMENT WORD COUNT.
1696 012010 100402 BMI 2$ ;BR IF DONE.
1697 012012 010322 MOV R3,(R2)+ ;LOAD DATA.
1698 012014 000773 BR 1$ ;DO IT AGAIN.
1699 012016 000207 2$: RTS PC ;RETURN.
1700
1701 ;RANDOM PATTERN GENERATOR
1702
1703 012020 162704 000002 PATR7:: SUB #2,R4 ;DECREMENT WORD COUNT
1704 012024 100411 BMI GIT ;BR IF DONE.
1705 012026 066767 171402 171376 ADD RANS,RANB
1706 012034 066767 171372 171372 ADD RANB,RANS ;GET NEW #.
1707 012042 016722 171366 MOV RANS,(R2)+ ;SAVE #.
1708 012046 000764 BR PATR7 ;CONTINUE.
1709 012050 000207 GIT: RTS PC ;RETURN
1710
1711 ; NO PATTERN GENERATION.
1712
1713 012052 000207 PATR8:: RTS PC ;RETURN.
1714
1715 ; THIS SUBROUTINE INITIATES TS05 COMMAND EXECUTION
1716 ; AND CHECKS FOR TS05 RESPONSE.
1717 ; INPUTS:

```

```

1718 ; OUTPUTS:
1719 ; REGISTERS: R2, R3.
1720 ; CALLS: DROPU, MOVMSG, FIRSTU, NEXTU, WSSR.
1721
1722 012054 012767 177777 171354 EXCUTE: MOV #1, TIME1 ;INIT TIMEOUT COUNTER.
1723 012062 50071$: REPEAT ;WAIT -
1724 012062 005367 171350 DEC TIME1 ;UPDATE TIMEOUT COUNTER.
1725 012066 005767 171344 TST TIME1 ;IF TIMED OUT:
1726 012072 001011 BNE 50072$
1727 012074 004767 000634 JSR PC, MOVMSG ;MOVE CURRENT PACKET MSG.
1728 012100 ERRDF 2, NSSRM, STAERM ;REPORT TS05 NOT READY
      012100 104455
      012102 000002
      012104 004536
      012106 006120
1729 012110 004767 005064 JSR PC, DROPU ;DROP THE UNIT.
1730 012114 000522 BR EXCRTN ;RETURN.
1731
1732 012116 50072$: BIT #TS, SSR, @TSSR(R5) ;WAIT UNTIL DEVICE IS READY.
1733 012116 032775 000200 002524 BEQ 50071$
1734 012124 001756 CMP CMDWRD, #SCH ;IF WE ARE DOING A SET CHAR CMD THEN:
1735 012126 026727 171266 140004 BNE 50073$
1736 012134 001022 MOV R5, RSSAVE ;SAVE CURRENT DEVICE POINTER.
1737 012136 010567 171310 JSR PC, FIRSTU ;FIND FIRST UNIT.
1738 012142 004767 004734
1739 012146 50074$: CMP DEVTBL(R5), #END ;WHILE DEVTBL(R5) NE #END DO
1740 012146 026527 002604 177777 BEQ 50075$
1741 012154 001405 JSR PC, WSSR ;WAIT FOR UNIT READY OR TIME OUT,
1742 012156 004767 000516 JSR PC, NEXTU ;FIND NEXT UNIT.
1743 012162 004767 004762
1744
1745 012166 000767 BR 50074$
1746 012170 50075$: MOV RSSAVE, R5 ;RESTORE CURRENT DEVICE POINTER.
1747 012170 016705 171256 MOV MSGPKA(R5), SCHBK ;SET UP ADR OF MSG PKT IN SCH BLOCK.
1748 012174 016567 002544 170272
1749
1750 012202 50073$: MOV MSGPKA(R5), R3 ;ADR OF THIS UNIT'S MSG PACKET.
1751 012202 016503 002544 CLR R2 ;CLR COUNTER.
1752 012206 005002
1753 012210 50076$: CMP R2, #MSGCNT ;WHILE THERE ARE MORE LOCATIONS:
1754 012210 020227 000020 BEQ 50077$
1755 012214 001405 MOV #1, (R3)+ ;INIT THE MSG PACKET WITH ALL 1'S
1756 012216 012723 177777 ADD #2, R2 ;UPDATE COUNTER.
1757 012222 062702 000002
1758
1759 012226 000770 BR 50076$
1760 012230 50077$: TSTB DINT ;ARE INTERRUPTS DISABLED.
1761 012230 105767 167756 BNE 1$ ;BR IF YES.
1762 012234 001023 CMPB INTFLG(R5), #1 ;IF MORE THAN ONE INTERRUPT HAS OCCURED:
1763 012236 126527 003472 000001 BLE 50100$
1764 012244 003412 MOV @TSSR(R5), TSSREG ;FREEZE THE CURRENT STATUS REG FOR PRINT
1765 012246 017567 002524 171200 ERRDF 15, TOOMM, STAERM ;REPORT TOO MANY INTERRUPTS.
1766 012254
      012254 104455
      012256 000017
      012260 004727
      012262 006120

```

```

TRAP C$ERDF
.WORD 2
.WORD NSSRM
.WORD STAERM

```

```

TRAP C$ERDF
.WORD 15
.WORD TOOMM
.WORD STAERM

```

166

```

1767 012264 004767 004710      JSR      PC,DROPU      ;DROP THE UNIT
1768 012270 000434      BR        EXCRTN      ;RETURN  UNIT HAS BEEN DROPPED.
1769
1770 012272      50100$:
1771 012272 005065 003472      CLR      INTFLG(R5)      ;CLR INTERRUPT FLAG FOR THIS DEV.
1772 012276 052767 000200 170024  BIS      #IE.C,CMDPKT      ;SET INT ENABLE BIT.
1773 012304 105767 171161 1$:  TSTB      ERRREC,IFB ERRREC EQ #0 THEN ;IF NOT RETRYING
1774 012310 001005      BNE      50101$
1775 012312 005265 003376      INC      RECCNT(R5)      ;LET RECCNT(R5) := RECCNT(R5) + #1
1776 012316 016577 003376 171062  MOV      RECCNT(R5),@DATAWT ;THEN UPDATE REC COUNT TO WRITE IT ON TAPE
1777
1778 012324      50101$:
1779 012324 012775 002330 002514  MOV      #CMDPKT,@TSDB(R5)      ;LOAD TSDB WITH CMDPKT ADDRESS
1780                                ;THIS INITIATES COMMAND EXECUTION.
1781 012332 032775 000200 002524  BIT      #TS.SSR,@TSSR(R5)      ;IF READY DID NOT DROP THEN:
1782 012340 001410      BEQ      50102$
1783 012342 004767 000366      JSR      PC,MOVMSG      ;MOVE CURRENT MESSAGE PACKET TO COMMON.
1784 012346      ERRDF      3,TOERM,STAERM      ;REPORT NO TS05 RESPONSE.
                                TRAP      C$ERDF
                                .WORD      3
                                .WORD      TOERM
                                .WORD      STAERM
1785 012356 004767 004616      JSR      PC,DROPU      ;DROP THE UNIT
1786
1787 012362      50102$:
1788 012362 000207      EXCRTN: RTS      PC      ;RETURN.
1789
1790      ; THIS SUBROUTINE WAITS FOR THE TS05 INERRUPT OR DONE BIT TO SET AND ALLOWS THE
1791      ; OPERATOR TO TRANSFER CONTROL TO THE SUPERVISOR.
1792      ; UPON APPEARANCE OF THE INTERRUPT OR DONE, CHECK TSSR FOR STATUS ERRORS,
1793      ; LOG BYTES AND ERRORS AND PERFORM ERROR RECOVERY IF NESSASARY.
1794      ; INPUTS:
1795      ; OUTPUTS:
1796      ; REGISTERS:      R2, R3.
1797      ; CALLS:      DROPU, MOVMSG, RECUD, CHKERR, LOG, CLRERR.
1798
1799 012364 012767 177777 171044  GOWAIT:: MOV      #-1,TIME1      ;INIT TIME OUT COUNTER.
1800 012372      50103$: ;REPEAT
1801 012372      BREAK      ;REPEAT UNTIL INTERRUPT OCCURES:
                                ;GO TO THE SUPER TO ALLOW TTY INPUT.
                                TRAP      C$BRK
1802 012374 104422      CMP      CMDWRD,@RWD      ;IF COMMAND WAS REWIND THEN:
1803 012402 026727 171020 102010  BNE      50104$
1804 012404      DELAY      10.      ;WAIT EXTRA MSECS EACH LOOP.
                                MOV      #10.,(PC)+
                                .WORD      0
                                MOV      L$DLY,(PC)+
                                .WORD      0
                                DEC      -6(PC)
                                BNE      -4
                                DEC      22(PC)
                                BNE      . 20
                                012404 012727 000012
                                012410 000000
                                012412 016727 167500
                                012416 000000
                                012420 005367 177772
                                012424 001375
                                012426 005367 177756
                                012432 001367
1805 012434      50104$:
1806 012434 026727 170760 105010  CMP      CMDWRD,#SFF      ;IF CMDWRD EQ #SFF OR CMDWRD EQ #SFR THEN
1807 012442 001404      BFQ      50105$
1808 012444 026727 170750 105410  CMP      CMDWRD,#SFR
1809 012452 001014      BNE      50106$
1810 012454      50105$:

```

```

1811 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$DLY,(PC).
      012466 000000          .WORD      0
      012470 005367 177772          DEC      6(PC)
      012474 001375          BNE      -4
      012476 005367 177756          DEC      -22(PC)
      012502 001367          BNE      -20
1812 012504          50106$:
1813 012504 105767 167502          TSTB     DINT          ;IF INTERRUPTS ARE ENABLED.
1814 012510 001003          BNE      50107$
1815 012512 016502 003472          MOV      INTFLG(R5),R2 ;FETCH INTERRUPT OCCURRED FLAG.
1816
1817 012516 000406          BR      50110$
1818 012520          50107$:
1819 012520 012703 000200          MOV      #TS.SSR,R3 ;SET UP A MASK FOR THE DONE BIT.
1820 012524 005103          COM      R3
1821 012526 017502 002524          MOV      @TSSR(R5),R2 ;FETCH DONE BIT.
1822 012532 040302          BIC      R3,R2
1823
1824 012534          50110$:
1825 012534 005367 170676          DEC      TIME1 ;UPDATE TIMEOUT COUNTER.
1826 012540 005702          TST      R2 ;REPEAT UNTIL INTERRUPT OR READY OCCURES.
1827 012542 001003          BNE      50111$
1828 012544 005767 170666          TST      TIME1
1829 012550 001310          BNE      50103$
1830 012552          50111$:
1831 012552 005767 170660          TST      TIME1 ;IF TIME OUT HAS OCCURRED:
1832 012556 001022          BNE      50112$
1833 012560 016577 003376 170620          MOV      RECCNT(R5),@DATAWT
1834 012566 005377 170614          DEC      @DATAWT
1835 012572 004767 000136          JSR      PC,MOVMSG ;MOVE CURRENT MSG PACKET TO COMMON AREA.
1836 012576          ERRDF 4,NOINTM,SYAERM ;REPORT NO INTERRUPT.
      012576 104455          TRAP      C$ERDF
      012600 000004          .WORD      4
      012602 004670          .WORD      NOINTM
      012604 006120          .WORD      STAERM
1837 012606 004767 004366          JSR      PC,DROPU ;DROP THE UNIT.
1838 012612 012703 003472          MOV      #ENDERF,R3 ;LET R3 := #ENDERF
1839 012616 004767 000042          JSR      PC,CLRERR ;CLEAR ALL ERROR FLAGS
1840
1841 012622 000417          BR      50113$
1842 012624          50112$:
1843 012624 004767 000104          JSR      PC,MOVMSG ;MOVE CURRENT MSG. PACKET TO COMMON AREA.
1844 012630 004767 000164          JSR      PC,RECU ;UPDATE THE RECORD COUNT.
1845 012634 004767 000350          JSR      PC,CHKERR ;CHECK FOR STATUS ERRORS.
1846 012640 105767 170617          TSTB     WRTYFG ;IFB WRTYFG EQ #0 THEN
1847 012644 001006          BNE      50114$
1848 012646 004767 002714          JSR      PC,LOG ;LOG BYTES AND ERRORS.
1849 012652 012703 003472          MOV      #ENDERF,R3 ;LET R3 := #ENDERF
1850 012656 004767 000002          JSR      PC,CLRERR ;CLEAR ALL ERROR FLAGS
1851
1852 012662          50114$:
1853
1854 012662          50113$:
1855 012662 000207          RTS      PC ;RETURN IF DONE.

```

```

1856
1857      ; SUBROUTINE TO CLEAR FLAGS.
1858      ; INPUTS:      R3 = LWA TO BE CLEARED * 2.
1859      ; OUTPUTS:
1860      ; REGISTERS:    R2
1861      ; CALLS:
1862
1863 012664 012702 003460 CLRERR: MOV  @BGNFLG,R2      ;LET R2 := @BGNFLG
1864 012670 501154: REPEAT
1865 012670 005022      CLR      (R2).      ;LET (R2). := #0
1866 012672 020203      CMP      R2,R3      ;UNTIL R2 EQ R3
1867 012674 001375      BNE      501154
1868 012676 000207      RTS      PC
1869
1870
1871      ; SUBROUTINE TO WAIT UNTIL CURRENT UNIT IS READY OR UNTIL TIME OUT.
1872      ; INPUTS:
1873      ; OUTPUTS:
1874      ; REGISTERS:
1875      ; CALLS:
1876
1877 012700 WSSR:
1878 012700 012767 177777 170530 MOV  @-1,TIME1      ;INIT TIMEOUT COUNTER.
1879 012706 501164: REPEAT UNTIL DEV READY OR TIMEOUT:
1880 012706 BREAK      ;BREAK TO THE SUPERVISOR.
1881 012710 005367 170522      DEC      TIME1      ;UPDATE TIMEOUT COUNTER.
1882 012714 032775 000200 002524 BIT  @TS.SSR,@TSSR(R5) ;UNTIL @TS.SSR SET IN @TSSR(R5) OR TIME1 EQ #0
1883 012722 001003      BNE      501174
1884 012724 005767 170506      TST      TIME1
1885 012730 001366      BNE      501164
1886 012732 501174:
1887 012732 000207      RTS      PC      ;RETURN.
1888
1889
1890
1891      ; SUBROUTINE TO MOVE THE CURRENT MESSAGE PACKET TO THE COMMON AREA AND
1892      ; TO UPDATE THE CURRENT TERMINATION CLASS CODE.
1893      ; INPUTS:
1894      ; OUTPUTS:
1895      ; REGISTERS:    R2, R3.
1896      ; CALLS:
1897
1898 012734 017567 002524 170512 MOVMSG: MOV  @TSSR(R5),TSSREG      ;FREEZE THE STATUS REG CONTENTS
1899 012742 016702 170506      MOV      TSSREG,R2      ;EXTRACT THE TERMINATION CLASS CODE.
1900 012746 042702 177761      BIC      @TSC.TCC,R2
1901 012752 010267 170472      MOV      R2,CTCC      ;AND SAVE IT
1902 012756 006267 170466      ASR      CTCC
1903 012762 016503 002544      MOV      MSGPKA(R5),R3      ;ADR OF THIS DEVICE'S MSG.
1904 012766 005002      CLR      R2      ;CLR COUNTER.
1905 012770 501204:
1906 012770 020227 000020      CMP      R2,@MSGCNT      ;WHILE THERE ARE MORE LOCATIONS:
1907 012774 001405      BEQ      501214
1908 012776 012362 002354      MOV      (R3),MSGPKT(R2)      ;MOVE MSG TO COMMON AREA
1909 013002 062702 0000C2      ADD      @2,R2      ;UPDATE COUNTER.
1910
1911 013006 000770      BR      501204

```

```

1912 013010
1913 013010 016767 167346 170464
1914 013016 000207
1915
1916
1917
1918
1919
1920
1921
1922 013020 105767 170441
1923 013024 001070
1924 013026 005365 003376
1925 013032 032767 000001 170410
1926 013040 001057
1927 013042 032767 100000 167316
1928 013050 001453
1929 013052 105267 170407
1930 013056 026727 170336 102010
1931 013064 001003
1932 013066 005065 003376
1933
1934 013072 000442
1935 013074
1936 013074 032767 004000 170316
1937 013102 001436
1938 013104 032767 000400 170306
1939 013112 001007
1940 013114 032767 000400 170302
1941 013122 001002
1942 013124 005265 003376
1943
1944 013130
1945
1946 013130 000423
1947 013132
1948 013132 032767 000400 170264
1949 013140 001417
1950 013142 032765 000002 003502
1951 013150 001013
1952 013152 105767 170313
1953 013156 001406
1954 013160 105767 170332
1955 013164 001403
1956 013166 105767 170273
1957 013172 001002
1958 013174 005365 003376
1959 013200
1960
1961 013200
1962
1963 013200
1964
1965 013200
1966
1967
1968 013200

50121$:
MOV MSGPKT+MS.XSO,EOTFLG ;MOVE XSTATO TO EOT FLAG.
RTS PC

;
; SUBROUTINE TO ADJUST THE RECORD COUNT.
; INPUTS:
; OUTPUTS:
; REGISTERS:
; CALLS:

RECUD:: TSTB RECLOG ;IF RECORD HAS NOT BEEN LOGGED:
BNE 50122$
DEC RECCNT(R5) ;LET RECCNT(R5) := RECCNT(R5) #1
BIT #BITO,CTCC ;IF TAPE MOVED
BNE 50123$
BIT #X2.OPM,MSGPKT+MS.XS2
BEQ 50123$
INCB RECLOG ;SET RECORD LOGGED.
CMP CMDWRD,#RWD ;IF THIS IS A REWIND CMD:
BNE 50124$
CLR RECCNT(R5) ;CLEAR RECORD COUNT.

BR 50125$

50124$:
BIT #BRF.C,CMDWRD ;IF BRF USED, UPDATE RECORD COUNT.
BEQ 50126$
BIT #MOD.CO,CMDWRD ;IF A FORWARD CMD:
BNE 50127$
BIT #MOD.CO,PCMDWD ;IF PREV CMD WAS A FWD ALSO:
BNE 50130$
INC RECCNT(R5) ;INCREMENT RECORD COUNT.

BR 50131$

50130$:
;IF REVERSE CMD:

BR 50131$

50127$:
BIT #MOD.CO,PCMDWD ;IF PREVIOUS CMD WAS A REV ALSO:
BEQ 50132$
BIT #XO.BOT,EOTFLG(R5) ;WHEN NOT AT BOT THEN
BNE 50133$
TSTB ERRREC ;CHECK THE ERROR RETRY INDICATOR
BEQ 2$ ;BR, IF WE ARE NOT NOW IN ERROR RETRY
TSTB VFYFLG ;CHECK THE WRITE VERIFY INDICATOR
BEQ 2$ ;BR, IF WE ARE NOT IN WRT/VFY MODE
TSTB RECLOG ;CHECK IF THIS RECORD HAS BEEN COUNTED
BNE 10$ ;BR, IF HAVE ALREADY BUMPED RECORD CNTR.
DEC RECCNT(R5) ;DECREMENT RECORD COUNT.

2$:
10$:

50133$:

50132$:

50131$:

50126$:

```

```

1969
1970 013200 501251:
1971
1972 013200 501231:
1973 013200 016577 003376 170200 MOV RECCNT(R5),@DATAWT ;LET @DATAWT := RECCNT(R5)
1974
1975 013206 501221:
1976 013206 000207 RTS PC ;RETURN.
1977
1978 ; THIS IS THE ERROR CHECK SUBROUTINE. AFTER INTERRUPT THIS
1979 ; SUBROUTINE IS CALLED TO CHECK THE TSOS STATUS.
1980 ; IF SPECIAL COND IS SET THEN THE TCC HANDLING SUBROUTINE IS ENTERED.
1981 ; IF THE RFC IS NON ZERO FOR A COMMAND REQUIRING A BPCR,
1982 ; THEN AN ERROR RFC IS REPORTED.
1983 ; INPUTS:
1984 ; OUTPUTS:
1985 ; REGISTERS: R2, R4.
1986 ; CALLS: TCC0-TCC7.
1987
1988 013210 032767 100000 170236 CHKERR: BIT @TS.SC,TSSREG ;IF SPECIAL COND STATUS IS SET THEN:
1989 013216 001441 BEQ 501341
1990 013220 026727 170224 000002 CMP CTCC,02 ;IF TCC IS NOT 2 THEN:
1991 013226 001405 BEQ 501351
1992 013230 105767 170235 TSTB ERRREC ;IF NOT IN ERROR RECOVERY:
1993 013234 001002 BNE 501361
1994 013236 005265 003336 INC SCCNT(R5) ;INC SC COUNTER.
1995
1996 013242 501361:
1997
1998 013242 501351:
1999 013242 032767 004000 170204 BIT @TS.NXM,TSSREG ;WHEN NON-EXISTANT MEMO
2000 013250 001004 BNE 501371
2001 013252 032767 040000 170174 BIT @TS.UPE,TSSREG
2002 013260 001412 BEQ 501401
2003 013262 501371:
2004 013262 032767 100000 167076 BIT @X2.OPM,MSGPKT+MS.XS2 ;AND TAPE NOT MOVED
2005 013270 001003 BNE 501411
2006 013272 012702 000005 MOV @5,R2 ;SET TCC5 INDEX
2007
2008 013276 000402 BR 501421
2009 013300 501411:
2010 013300 012702 000004 MOV @4,R2 ;TAPE MOVED, SET TCC4 INDEX
2011
2012 013304 501421:
2013
2014 013304 000402 BR 501431
2015 013306 501401:
2016 013306 016702 170136 MOV CTCC,R2 ;SET DETECTED TCC INDEX
2017
2018 013312 501431:
2019 013312 006302 ASL R2 ;CURRENT TCC X 2.
2020 013314 004772 013414 JSR PC,@TCCRA(R2) ;GO TO THE TCC HANDLING SUBROUTINE.
2021
2022 013320 000426 BR 501441
2023 013322 501341:
2024 013322 032767 004000 170070 BIT @BRF.C,CMDWRD ;IF BRF IS USED IN THIS CMD THEN:
2025 013330 001422 BEQ 501451

```

```

2026 013332 005767 167022          TST      MSGPKT+MS,RFC          ;IF THERE IS AN RFC THEN:
2027 013336 001417          BEQ      501464
2028 013340 105767 170151          TSTB     RANDOM          ;IFB RANDOM EQ 00 ORB VFYFLG NE 00 THEN
2029 013344 001403          BEQ      501474
2030 013346 105767 170144          TSTB     VFYFLG
2031 013352 001411          BEQ      501504
2032 013354          501474:
2033
2034 013354 105767 170141          TSTB     IRE          ;IF NOT IN RANDOM OR IF CMD IS WT.:
2035 013360 001006          BNE      501514          ;IF RFC ERROR REPORTS ARE ALLOWED:
2036 013362 005265 003356          INC      HRDCNT(R5)          ;UPDATE HARD ERROR COUNT
2037 013366          ERRHRD 13,RFCERM,STAERM          ;REPORT RFC ERROR
          013366 104456          TRAP      C$ERHRD
          013370 000015          .WORD    13
          013372 004521          .WORD    RFCERM
          013374 006120          .WORD    STAERM

2038
2039 013376          501514:
2040
2041 013376          501504:
2042
2043 013376          501464:
2044
2045 013376          501454:
2046
2047 013376          501444:
2048 013376 105767 170065          TSTB     RWERR          ;IF A READ/WRITE ERROR HAS OCCURRED THEN:
2049 013402 001403          BEQ      501524
2050 013404 016767 170012 166716          MOV      CMDSAV,CMDPKT          ;RESTORE CMD PACKET AFTER ERROR RECOV.
2051
2052 013412          501524:
2053 013412 000207          RTS      PC          ;RETURN.
2054
2055          ; ADDRESSES OF TCC HANDLING ROUTINES FOR TERMINATION CLASS CODES 0 7.
2056
2057 013414 013434          TCCRA: TCC0
2058 013416 013452          TCC1
2059 013420 013470          TCC2
2060 013422 013600          TCC3
2061 013424 013616          TCC4
2062 013426 014232          TCC5
2063 013430 014330          TCC6
2064 013432 014472          TCC7
2065
2066          ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 0, UNDEFINED SPECIAL
2067          ; CONDITION ERROR.
2068          ; INPUTS:
2069          ; OUTPUTS:
2070          ; REGISTERS:
2071          ; CALLS:
2072
2073 013434 005265 003356          TCC0:: INC      HRDCNT(R5)          ;UPDATE HARD ERROR COUNT.
2074 013440          ERRHRD 5,SCERM,STAERM          ;REPORT SPECIAL CONDITION ERROR.
          013440 104456          TRAP      C$ERHRD
          013442 000005          .WORD    5
          013444 004475          .WORD    SCERM
          013446 006120          .WORD    STAERM

```

```

2075 013450 000207      RTS PC                      ;RETURN.
2076
2077
2078      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 1, ATTENTION CONDITION.
2079      ;      THIS TCC INDICATES THAT THE DRIVE HAS UNDERGONE A STATUS CHANGE
2080      ;      SUCH AS GOING OFFLINE OR COMING ONLINE.
2081      ;      INPUTS:
2082      ;      OUTPUTS:
2083      ;      REGISTERS:      R2,R4
2084      ;      CALLS:          DROPU
2085
2086 013452      TCC1:: ERRDF 6,ATTNM,STAERM      ;REPORT ATTENTION-UNIT OFF LINE.
      013452      104455                      TRAP      C:ERDF
      013454      000006                      .WORD      6
      013456      004603                      .WORD      ATNM
      013460      006120                      .WORD      STAERM
2087 013462 004767 003512      JSR      PC,DROPU      ;DROP THE UNIT.
2088 013466 000207      RTS      PC                  ;RETURN.
2089
2090      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 2, TAPE STATUS ALERT.
2091      ;      A STATUS CONDITION HAS BEEN ENCOUNTERED THAT MAY HAVE SIGNIFICANCE
2092      ;      TO THE PROGRAM. BITS OF INTEREST INCLUDE TMK, RLS, LET, RLL, BOT, EOT.
2093      ;      INPUTS:
2094      ;      OUTPUTS:
2095      ;      REGISTERS:
2096      ;      CALLS:
2097
2098 013470 032767 000002 166664      TCC2:: BIT      @X0.BOT,MSGPKT+MS.XSO
2099 013476 001404                      BEQ      501534
2100 013500 105767 170010      TSTB      EXPBOT
2101 013504 001401                      BEQ      501534
2102
2103 013506 000433                      BR      TC2RTN      ;IF AT BOT AND BOT IS EXPECTED:
2104                      ;RETURN-TCC2 CAUSED BY EXPECTED BOT.
2105 013510                      501534:
2106 013510 032767 170002 166644      BIT      @X0.RLS!X0.RLL!X0.TMK!X0.LET!X0.BOT,MSGPKT+MS.XSO
2107                      ;IF @X0.RLS!X0.RLL!X0.TMK!X0.LET!X0.BOT SETIN MSGPKT+MS.XSO THEN
2108
2109 013516 001427                      BEQ      501544
2110
2111 013520 105767 167771      TSTB      RANDOM      ;IF TCC2 CAUSED BY ANYTHING BUT EOT:
2112 013524 001403                      BEQ      501554      ;IFB RANDOM EQ #0 ORB VFYFLG NE #0 THEN
2113 013526 105767 167764      TSTB      VFYFLG
2114 013532 001421                      BEQ      501564
2115 013534                      501554:
2116
2117 013534 105767 167761      TSTB      IRE
2118 013540 001016                      BNE      501574      ;IF NOT IN RANDOM OR IF CMD IS WTV:
2119 013542 105767 167723      TSTB      ERRREC      ;IF RFC ERROR REPORTS ARE ALLOWED:
2120 013546 001403                      BEQ      501604      ;IF WE ARE IN ERROR RECOVERY THEN:
2121 013550 105267 167714      INCB      UNREC      ;SET UNRECOVERABLE FLAG FOR LOG.
2122                      ;ELSE - IF NOT IN ERROR RECOVERY:
2123 013554 000402                      BR      501614
2124 013556                      501604:
2125 013556 005265 003336      INC      SCCNT(R5)      ;INCREMENT THE SPEC COND COUNTER.
2126
2127 013562                      501614:

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0082

```

2128 013562 005265 003356      INC      HRDCNT(R5)      ;UPDATE HARD ERROR COUNT.
2129 013566      ERRHRD  7,TSAM,STAERM      ;REPORT TAPE STATUS ALERT.
      013566 104456      TRAP      C$ERHRD
      013570 000007      .WORD      7
      013572 004705      .WORD      TSAM
      013574 006120      .WORD      STAERM

2130
2131 013576      501574:
2132
2133 013576      501564:
2134
2135 013576      501544:
2136
2137 013576 000207      TC2RTN:  RTS   PC      ;RETURN.
2138
2139
2140      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 3, FUNCTION REJECT.
2141      ;      THE SPECIFIED FUNCTION WAS NOT INITIATED. BITS OF INTEREST ARE
2142      ;      RMR, OFL, VCK, BOT, ILC, WLE, ILA, AND NBA.
2143      ;      INPUTS:
2144      ;      OUTPUTS:
2145      ;      REGISTERS:      R2,R4
2146      ;      CALLS:      DROPU
2147
2148 013600      TCC3::  ERRDF  8,FUNRM,STAERM      ;REPORT FUNCTION REJECT.
      013600 104455      TRAP      C$ERDF
      013602 000010      .WORD      8
      013604 004622      .WORD      FUNRM
      013606 006120      .WORD      STAERM

2149 013610 004767 003364      JSR      PC,DROPU      ;DROP THE UNIT.
2150 013614 000207      RTS      PC      ;RETURN.
2151
2152      ;      SUBROUTINE TO HANDLE TERMINATION CLASS CODE 4, RECOVERABLE ERROR.
2153      ;      TAPE POSITION IS ONE RECORD BEYOND WHAT ITS POSITION WAS WHEN
2154      ;      THE FUNCTION WAS INITIATED. RECOVERY PROCEDURE IS TO LOG THE
2155      ;      ERROR AND ISSUE THE APPROPRIATE RETRY COMMAND.
2156      ;      2 WRITE-ERROR-RECOVERY ALGORITHMS CAN BE SELECTED:
2157      ;      THE FIRST ONE, VIA BADTSW SWITCH, DOES DETECT BAD SPOTS ON TAPE.
2158      ;      IT CALLS A WRITE RETRY SUBR UNTIL THE RECORD IS RECOVERED
2159      ;      OR 20 BAD SPOTS HAVE BEEN LOGGED. ON REACHING 20 BAD
2160      ;      SPOTS LOGGED, A BAD TAPE OVERFLOW MSG IS PRINTED AND THE
2161      ;      UNIT DROPPED.
2162      ;      THE SECOND ALGORITHM ISSUES THE TS05 WRITE RETRY COMMAND
2163      ;      UP TO 16 TIMES BEFORE DROPPING THE UNIT OR PROCEEDING
2164      ;      WITH THE NEXT RECORD ON RECOVERY.
2165      ;      INPUTS:
2166      ;      OUTPUTS:
2167      ;      REGISTERS:      R2,R4.
2168      ;      CALLS:      RTLE, EXCUTE, GOWAIT, DROPU, WRTY
2169
2170 013616 026727 167604 000002      TCC4::  CMP      CMDLG,#2      ;IF CMDLG EQ #2 ANDB BADTSW NE #0 THEN
2171 013624 001125      BNE      501624
2172 013626 105767 166356      TSTB      BADTSW
2173 013632 001522      BEQ      501624
2174 013634 105767 167631      TSTB      ERRREC      ;IFB ERRREC EQ #0 ANDB ERCVER NE #0 THEN
2175 013640 001007      BNE      501634
2176 013642 105767 166341      TSTB      ERCVER

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SFO 0083

2177	013646	001404		BEQ	501634			
2178	013650			ERRSOFT	9, RERM, STAERM			
	013650	104457					TRAP	CERSOFT
	013652	000011					.WORD	9
	013654	005017					.WORD	RERM
	013656	006120					.WORD	STAERM
2179								
2180	013660			501634:				
2181	013660	105767	166327	TSTB	IREF	; IFB IREF EQ #0 THEN		
2182	013664	001102		BNE	501644			
2183	013666	105267	167577	INCB	ERRREC	; RETRY FLAG FOR EXECUTE SUBR: DON'T UPDATE REC CN		
2184	013672	105267	167566	INCB	WRTYER	; REWRITE ERROR FLAG FOR WRTY SUBR		
2185	013676	105767	167561	TSTB	WRTYFG	; FIRST RETRY ON THIS RECORD: SUBSEQUENT		
2186	013702	001072		BNE	501654			
2187						; RETRIES WITH TCC4 ERRORS BY-PASS THIS SECTION		
2188	013704	016767	167510	MOV	CMDWRD, WTYWRD	; SAVE WRITE COMMAND PACKET		
2189	013712	016767	166412	MOV	CMDPKT, WTYCMD			
2190	013720	016767	166412	MOV	CMDPKT, CP.CNT, WTYBRF			
2191	013726	105267	167535	INCB	RWERR	; LOG SUBR FLAG: COUNT WRT ERRORS		
2192	013732	105267	167525	INCB	WRTYFG	; RETRY IN PROGRESS FLAG		
2193								
2194	013736			501664:	; REPEAT			
2195	013736	005265	003316	INC	WRTYCT(R5)	; COUNT GLOBAL WRITE RETRIES		
2196	013742	005067	167512	CLR	RETRYC	; CLEAR # OF RETRIES PER RECORD		
2197	013746	105067	167510	CLRB	RPTCNT	; CLEAR # OF REPEATS		
2198	013752	004767	000660	JSR	PC, WRTY	; CALL WRITE RETRY		
2199	013756	105767	167502	TSTB	WRTYER	; REPEAT RETRIES ON SAME RECORD		
2200	013762	001404		BEQ	501674			
2201	013764	027727	167522	CMP	BTPT, #40.			
2202	013772	103761		BLO	501664			
2203	013774			501674:				
2204						; UNTIL RECOVERED OR 20 BAD SPOTS		
2205	013774	027727	167512	CMP	BTPT, #40.	; WHEN 20 BAD SPOTS LOGGED		
2206	014002	103423		BLO	501704			
2207	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	C\$PNTB
	014020	062706	000004				ADD	#4, SP
2208	014024	004767	001266	JSR	PC, BORERS	; ERASE BAD RECORD		
2209	014030	005365	003376	DEC	RECCNT(R5)			
2210	014034	004767	003140	JSR	PC, DROPU	; DROP UNIT		
2211	014040	005065	003376	CLR	RECCNT(R5)			
2212	014044	012775	002350	MOV	RWCPK, BTSD(R5)	; REWIND UNIT		
2213								
2214	014052			501704:				
2215	014052	105067	167405	CLRB	WRTYFG	; RETRY COMPLETE FLAG		
2216	014056	105267	167447	INCB	MISCFG	; DO NOT HALT ON THIS CMD FLG		
2217	014062	016767	001020	MOV	WTYWRD, PCMDWD	; RESTORE ORIGINAL WRT CMD AFTER RECOVERY		
2218								
2219	014070			501654:				
2220								
2221	014070	000402		BR	501714			
2222	014072			501644:				
2223	014072	105267	167372	INCB	UNREC	; LET UNREC :B= UNREC + #1 ;		
2224								

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0084

```

2225 014076          50171$:
2226
2227 014076 000454          BR      50172$
2228 014100          50162$:
2229 014100 004767 000404      JSR      PC,RTLE          ;CHECK FOR RETRY LIMIT EXCEEDED.
2230 014104 026727 167316 000002      CMP      CMDLG,02          ;IF READ CMD THEN:
2231 014112 003411          BLE      50173$
2232 014114 012702 000020      MOV      @RRECL,R2          ;R2=READ RETRY COUNT LIMIT / 2
2233 014120 006202          ASR      R2
2234 014122 026702 167332      CMP      RETRYC,R2          ;IF RETRY COUNT IS MORE THAN HALF LIMIT:
2235 014126 002403          BLT      50174$
2236 014130 052767 020000 166172      BIS      @OPP.C,CMDPKT          ;SET OPPOSITE BIT FOR RETRY2.
2237
2238 014136          50174$:
2239
2240 014136          50173$:
2241 014136 005767 167316      TST      RETRYC          ;IF THIS IS THE ORIGINAL ERROR THEN:
2242 014142 001007          BNE      50175$
2243 014144 105767 166037      TSTB     ERVER
2244 014150 001404          BEQ      50175$
2245 014152          ERRSOFT 9,RERM,STAERM          ;REPORT RECOVERABLE ERROR
2246          014152 104457          TRAP      C$ERSOFT
2247          014154 000011          .WORD     9
2248          014156 005017          .WORD     RERM
2249          014160 006120          .WORD     STAERM
2250
2251          50175$:          ;PROVIDED OPERATOR HAS ENABLED THE REPORT
2252 014162          INC      RETRYC          ;UPDATE RETRY COUNT.
2253 014166 005267 167272          BIS      @MOD.C1,CMDPKT          ;SET RETRY BIT IN CMD PACKET.
2254 014174 105767 166013          TSTB     IREC          ;IF ERROR RECOVERY ENABLED:
2255 014200 001011          BNE      50176$
2256 014202 105267 167263          INCB     ERRREC          ;SET ERROR RECOVERY FLAG.
2257 014206 012602          MOV      (SP)+,R2          ;POP 2 RTN ADRS FROM STACK.
2258 014210 012602          MOV      (SP)+,R2
2259 014212 004767 175636          JSR      PC,EXCUTE          ;GO EXECUTE THE RETRY COMMAND.
2260 014216 000167 176142          JMP      GOWAIT          ;GO WAIT FOR INTERRUPT + CHECK STATUS.
2261          50176$:          ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
2262 014222 000402          BR      50177$
2263 014224          50176$:
2264 014224 105267 167240          INCB     UNREC          ;SET UNRECOVERABLE ERROR FLAG.
2265
2266          50177$:
2267          50172$:
2268          RTS PC          ;RETURN
2269
2270          ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 5, RECOVERABLE ERROR.
2271          ; TAPE POSITION HAS NOT CHANGED. RECOVERY PROCEDURE IS TO LOG THE
2272          ; ERROR AND RE-ISSUE THE ORIGINAL COMMAND.
2273          ; INPUTS:
2274          ; OUTPUTS:
2275          ; REGISTERS: R2,R4.
2276          ; CALLS: RTLE, EXCUTE, GOWAIT, DROPU.
2277 014232 004767 000252      TCC5:: JSR      PC,RTLE          ;CHECK FOR RETRY LIMIT EXCEEDED
2278 014236 005767 167216      TST      RETRYC          ;IF THIS IS THE ORIGINAL ERROR THEN:
2279 014242 001004          BNE      50200$

```

```

2278 014244 ERRSOFT 10, RERM, STAERM ;REPORT RECOVERABLE ERROR.
      014244 104457
      014246 000012
      014250 005017
      014252 006120
2279 014254 50200$:
2280 014254 005267 167200 INC RETRYC ;UPDATE RETRY COUNTER.
2281 014260 105767 165727 TSTB IREC ;IF ERROR RECOVERY IS ENABLED:
2282 014264 001016 BNE 50201$
2283 014266 105267 167177 INCB ERRREC ;SET ERROR RECOVERY FLAG.
2284 014272 005265 003376 INC RECCNT(R5) ;UPDATE REC COUNT
2285 014276 016577 003376 167102 MOV RECCNT(R5), @DATAWT ;AND INSERT IT INTO WRT BFR
2286 014304 012602 MOV (SP)+, R2 ;POP 2 RTN ADRS FROM STACK.
2287 014306 012602 MOV (SP)+, R2
2288 014310 004767 175540 JSR PC, EXCUTE ;GO RE-ISSUE THE COMMAND.
2289 014314 000167 176044 JMP GOWAIT ;GO WAIT FOR INTERRUPT + CHECK STATUS.
2290 ;ELSE IF ERROR RECOVERY IS NOT ENABLED:
2291 014320 000402 BR 50202$
2292 014322 50201$:
2293 014322 105267 167142 INCB UNREC ;SET UNRECOVERABLE ERROR FLAG.
2294
2295 014326 50202$:
2296 014326 000207 RTS PC ;RETURN.
2297
2298
2299 ;
2300 ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 6, UNRECOVERABLE ERROR.
2301 ; TAPE POSITION HAS BEEN LOST. THE ONLY VALID RECOVERY PROCEDURE
2302 ; IS TO REWIND AND START OVER AT BOT UNLESS THE TAPE HAS LABELS OR
2303 ; SEQUENCE NUMBERS. THIS DIAGNOSTIC WILL REWIND AND RETRY THE
2304 ; COMMAND ONLY IF DENSITY CHECK IS SET, OTHERWISE THE UNIT WILL BE
2305 ; DROPPED FROM THE TEST SEQUENCE.
2306 ; INPUTS:
2307 ; OUTPUTS:
2308 ; REGISTERS: R2, R4
2309 ; CALLS: RTLE, WSSR, EXCUTE, GOWAIT, DROPU
2310 014330 036767 163454 166032 TCC6:: BIT X3.DCK, MSGPKT+MS.XS3; IF X3.DCK NOTSET IN MSGPKT+MS.XS3 THEN
2311 014336 001016 BNE 50203$
2312 ;IF THERE IS NO DENSITY CHECK THEN:
2313 014340 005767 167062 TST CMDLG ;IF CMD IS A READ OR WRITE THEN:
2314 014344 001404 BEQ 50204$
2315 014346 105267 167115 INCB RWERR ;SET RD/WR ERROR FLAG.
2316 014352 105267 167112 INCB UNREC ;SET UNRECOVERABLE ERROR FLAG.
2317
2318 014356 50204$:
2319 014356 ERRDF 11, URERM, STAERM ;REPORT UNRECOVERABLE ERROR.
      014356 104455
      014360 000013
      014362 005041
      014364 006120
2320 014366 004767 002606 JSR PC, DROPU ;REPORT ERROR + DROP UNIT.
2321 ;ELSE-IF THERE IS DENSITY CHECK:
2322 014372 000436 BR 50205$
2323 014374 50203$:
2324 014374 004767 000110 JSR PC, RTLE ;CHECK FOR RETRY LIMIT EXCEEDED.
2325 014400 005767 167054 TST RETRYC ;IF THIS IS THE ORIGINAL ERROR THEN:
2326 014404 001004 BNE 50206$

```

```

TRAP C$ERSOFT
.WORD 10
.WORD RERM
.WORD STAERM

```

```

TRAP C$ERDF
.WORD 11
.WORD URERM
.WORD STAERM

```

```

2327 014406 ERRSOF 11,URERM,STAERM ;REPORT DENSITY CHECK ERROR
      014406 104457 TRAP C$ERSOF 11
      014410 000013 .WORD 11
      014412 005041 .WORD URERM
      014414 006120 .WORD STAERM

2328
2329 014416 50206$: INC RETRYC ;UPDATE RETRY COUNT.
2330 014416 005267 167036 TSTB IRE ;IF ERROR RECOVERY IS ENABLED THEN:
2331 014422 105767 167073 BNE 50207$
2332 014426 001016 INCB ERRREC ;SET ERROR RECOVERY FLAG,
2333 014430 105267 167035 MOV @RWCPK,@TSDB(R5) ;ISSUE A REWIND COMMAND,
2334 014434 012775 002350 002514 JSR PC,WSSR ;WAIT FOR SUBSYSTEM READY,
2335 014442 004767 176232 MOV (SP)+,R2 ;POP 2 RTN ADRS FROM STACK.
2336 014446 012602 MOV (SP)+,R2
2337 014450 012602 JSR PC,EXCUTE ;REISSUE THE COMMAND,
2338 014452 004767 175376 JMP GOWAIT ;WAIT FOR INTERRUPT
2339 014456 000167 175702 BR 50210$ ;ELSE-IF ERR REC DISABLED:
2340
2341 014462 000402 50207$: INCB UNREC ;SET UNRECOVERABLE ERROR FLAG.
2342 014464
2343 014464 105267 167000 50210$:
2344
2345 014470 50205$:
2346
2347 014470
2348 014470 000207 RTS PC ;RETURN
2349
2350 ; SUBROUTINE TO HANDLE TERMINATION CLASS CODE 7, FATAL SUBSYSTEM
2351 ; ERROR. THE SUBSYSTEM IS INCAPABLE OF PROPERLY PERFORMING
2352 ; COMMANDS OR AT LEAST ITS INTEGRITY IS SERIOUSLY QUESTIONABLE.
2353 ; REFER TO THE FATAL CLASS CODE FIELD IN THE TSSR REGISTER FOR
2354 ; ADDITIONAL INFORMATION ON THE TYPE OF FATAL ERROR.
2355 ; INPUTS:
2356 ; OUTPUTS:
2357 ; REGISTERS: R2, R4
2358 ; CALLS:
2359
2360 014472 TCC7:: ERRDF 12,FATSM,STAERM ;REPORT FATAL SUBSYSTEM ERROR.
      014472 104455 TRAP C$ERDF
      014474 000014 .WORD 12
      014476 004642 .WORD FATSM
      014500 006120 .WORD STAERM
2361 014502 004767 002472 JSR PC,DROPU ;DROP THE UNIT.
2362 014506 000207 RTS PC ;RETURN.
2363
2364
2365 ; SUBROUTINE TO CHECK FOR RETRY LIMIT EXCEEDED. PRINTS ERROR MESSAGE
2366 ; IF EXCEEDED AND DROP UNIT UNLESS COMMAND IS A READ.
2367 ; INPUTS:
2368 ; OUTPUTS:
2369 ; REGISTERS: R2, R4.
2370 ; CALLS: DROPU
2371
2372 014510 005767 166712 RTLE:: TST CMDLG ;IF CMD IS NOT A READ OR WRITE THEN:
2373 014514 001010 BNE 50211$
2374 014516 004767 104455 ERRDF 11,URERM,STAERM ;REPORT UNRECOVERABLE ERROR.
      014516 104455 TRAP C$ERDF

```

Line	Address	Offset	Label	Instruction	Comments	Trap	Condition
	014520	000013				.WORD	11
	014522	005041				.WORD	URERM
	014524	006120				.WORD	STAERM
2375	014526	004767	002446	JSR	PC,DROPU		
2376	014532	012602		MOV	(SP)+,R2		
2377	014534	000437		BR	RTLRTN		
2378							
2379	014536						
2380	014536	105267	166725	INCB	RWERR		
2381	014542	026727	166660	CMP	CMDLG,#2		
2382	014550	001016		BNE	50212\$		
2383	014552	026727	166702	CMP	RETRYC,#WRECL		
2384	014560	001011		BNE	50213\$		
2385	014562	105267	166702	INCB	UNREC		
2386	014566			ERRDF	14,RLEXM,STAERM		
	014566	104455				TRAP	C\$ERDF
	014570	000016				.WORD	14
	014572	004556				.WORD	RLEXM
	014574	006120				.WORD	STAERM
2387	014576	004767	002376	JSR	PC,DROPU		
2388	014602	012602		MOV	(SP)+,R2		
2389	014604						
2390							
2391	014604	000413		BR	50214\$		
2392	014606						
2393	014606	026727	166646	CMP	RETRYC,#RRECL		
2394	014614	001007		BNE	50215\$		
2395	014616	105267	166646	INCB	UNREC		
2396	014622			ERRHRD	14,RLEXM,STAERM		
	014622	104456				TRAP	C\$ERHRD
	014624	000016				.WORD	14
	014626	004556				.WORD	RLEXM
	014630	006120				.WORD	STAERM
2397	014632	012602		MOV	(SP)+,R2		
2398	014634						
2399							
2400	014634						
2401	014634	000207		RTLRTN: RTS	PC		
2402							
2403							
2404							
2405							
2406							
2407							
2408							
2409							
2410							
2411							
2412							
2413							
2414							
2415							
2416							
2417							
2418							
2419	014636			WRTY::			
2420							

167

```

2421 014636      50217$:
2422              ;BEGIN REPEAT              ;REPEAT
2423
2424 014636      50221$:
2425 014636      004767 000454      JSR      PC,BORERS      ;BACKSPACE/ERASE ONE RECORD
2426 014642      105067 166616      CLR      WRTYER      ;CLEAR WRITE RETRY ERROR
2427 014646      004767 000620      JSR      PC,REWRT      ;REWRITE RECORD ON SAME SPOT
2428 014652      105267 166604      INCB     RPTCNT      ;COUNT REPEATS
2429 014656      126727 166600 000004  CMPB     RPTCNT,#4      ;LIMIT: 4 REPEATS OR RECOVERED
2430 014664      001403              BEQ      50222$
2431 014666      105767 166572      TSTB     WRTYER
2432 014672      001761              BEQ      50221$
2433 014674
2434              50222$:
2435              ;END REPEAT
2436 014674      005267 166560      50220$:
2437 014700      105767 166560      INC      RETRYC      ;COUNT RETRIES
2438 014704      001001              TSTB     WRTYER
2439 014706      000457              BNE      50223$
2440              BR      50216$
2441              ;EXIT RETRY LOOP IF RECOVERED
2442 014710      105767 165273      50223$:
2443 014714      001415              TSTB     ERCVER      ;IFB ERCVER NE #0 THEN
2444 014716              BEQ      50225$
2445 014716      005046              PRINTB   #BTMSG1,RETRYC,<B,RPTCNT> ;PRINT SUSPECTED BAD SPOT
2446 014720      156716 166536              CLR      -(SP)
2447 014724      016746 166530              BISB     RPTCNT,(SP)
2448 014730      012746 015112              MOV      RETRYC,-(SP)
2449 014734      012746 000003              MOV      #BTMSG1,-(SP)
2450 014740      010600              MOV      #3,-(SP)
2451 014742      104414              MOV      SP,R0
2452 014744      062706 000010              TRAP     C#PNTB
2453              ADD      #10,SP
2454 014750      026727 166504 000001 50225$:
2455 014756      001021              CMP      RETRYC,#1      ;ON FIRST RETRY, LOGG BAD SPOT
2456 014760      016567 002616 166524  BNE      50226$
2457 014766      017704 166520      MOV      BTADDR(R5),BTPT      ;BTPT IS BOTH THE BAD SPOT COUNTER
2458 014772      062704 000002              MOV      #BTPT,R4      ;AND THE LOGGING INDEX
2459 014776      010477 166510      ADD      #2,R4
2460 015002      020427 000050      MOV      R4,#BTPT
2461 015006      101005              CMP      R4,#40      ;IF R4 LOS #40. THEN
2462 015010      016703 166476      BHI      50227$
2463 015014      060304              MOV      BTPT,R3      ;STORE FIRST 20 BAD SPOTS
2464 015016      016514 003376      ADD      R3,R4      ;LET R4 := R4 + R3
2465              MOV      RECCNT(R5),(R4) ;LET (R4) := RECCNT(R5)
2466
2467 015022      50227$:
2468
2469 015022      50226$:
2470 015026      105267 166477      INCB     ERSFLG      ;ERASE FLAG TO ERASE BAD RECORD
2471 015026      105067 166435      CLR      RWERR      ;CANCELL "LCG" ERROR FLAG ON FAILING RET
2472 015032      105067 166424      CLR      RPTCNT      ;CLEAR REPEAT COUNT FOR NEXT RETRY
2473
2474 015036      50224$:
2475 015036      026727 166416 000004  CMP      RETRYC,#4      ;LIMIT: 4 RETRIES
2476 015044      001274              BNE      50217$
2477
2478 015046      50216$:
2479              ;END RETRY

```

```

2470 015046 105767 166412      TSTB   WRTYER ;IFB WRTYER NE #0 THEN
2471 015052 001413      BEQ     50230$
2472 015054 105767 165127      TSTB   ERCVER ;IFB ERCVER NE #0 THEN
2473 015060 001410      BEQ     50231$
2474 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

2475
2476 015102      50231$:
2477
2478 015102      50230$:
2479 015102 000207      RTS     PC
2480
2481 015104 000000      WTYCMD: .WORD 0 ;STORAGE FOR WRITE CMD WHILE RETRYING
2482 015106 000000      WTYWRD: .WORD 0 ;STORAGE FOR WRITE CMD WORD WHILE RETRYING
2483 015110 000000      WTYBRF: .WORD 0 ;STORAGE FOR WRITE BPCR WHILE RETRYING
2484
2485 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
2486 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
2487 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 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0090

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	

2488
2489
2490
2491
2492
2493
2494
2495
2496

.EVEN

2497 015316 016767 166076 166100 BORERS::MOV CMDWRD,PCMDWD ;SET COMMAND TO SPACE REV

2498 015324 012767 104410 166066 MOV #SRR,CMDWRD ;LET CMDWRD := #SRR ;

2499 015332 016767 166062 164770 MOV CMDWRD,CMDPKT ;LET CMDPKT := CMDWRD CLR.BY #BRF.C ;

2500 015340 042767 004000 164762 BIC #BRF.C,CMDPKT ;

2501 015346 016767 164756 166046 MOV CMDPKT,CMDSAV ;LET CMDSAV := CMDPKT ;

2502 015354 012767 000001 164750 MOV #1,CMDPKT.CP.ADL ;LET CMDPKT.CP.ADL := #1 ;

2503 015362 005067 166040 CLR CMDLG ;LET CMDLG := #0 ;

2504 015366 004767 173472 JSR PC,CMDAC ;

2505 015372 004767 174456 JSR PC,EXCUTE ;

2506 015376 004767 174762 JSR PC,GOWAIT ;

2507 015402 004767 002072 JSR PC,CKHAE ;

2508 015406 105767 166113 TSTB ERSFLG ;WHEN ERASE FLAG IS SET, DO ERASE

2509 015412 001426 BEQ 502321 ;

2510 015414 016767 166000 166002 MOV CMDWRD,PCMDWD ;LET PCMDWD := CMDWRD ;

2511 015422 012767 100411 165770 MOV #ERS,CMDWRD ;LET CMDWRD := #ERS ;

2512 015430 016767 165764 164672 MOV CMDWRD,CMDPKT ;LET CMDPKT := CMDWRD ;

2513 015436 016767 164666 165756 MOV CMDPKT,CMDSAV ;LET CMDSAV := CMDPKT ;

2514 015444 004767 173414 JSR PC,CMDAC ;

2515 015450 004767 174400 JSR PC,EXCUTE ;

2516 015454 004767 174704 JSR PC,GOWAIT ;

2517 015460 004767 002014 JSR PC,CKHAE ;

2518 015464 105067 166035 CLRB ERSFLG ;LET ERSFLG := #0

2519
2520 015470 502321:
2521 015470 000207 RTS PC

2522
2523
2524

; SUBR TO REWRITE A BADLY WRITTEN RECORD

2525 015472 016767 165722 165724 REWRT::MOV CMDWRD,PCMDWD ;RESTORE WRITE COMMAND PACKET

2526 015500 016767 177402 165712 MOV WTYWRD,CMDWRD ;LET CMDWRD := WTYWRD ;

2527 015506 016767 177372 164614 MOV WTYCMD,CMDPKT ;LET CMDPKT := WTYCMD ;

2528 015514 016767 164610 165700 MOV CMDPKT,CMDSAV ;LET CMDSAV := CMDPKT ;

2529 015522 016767 165660 164602 MOV DATAWT,CMDPKT.CP.ADL ;LET CMDPKT.CP.ADL := DATAWT ;

2530 015530 016767 177354 164600 MOV WTYBRF,CMDPKT.CP.CNT ;LET CMDPKT.CP.CNT := WTYBRF ;

2531 015536 012767 000002 165662 MOV #2,CMDLG ;LET CMDLG := #2 ;

2532 015544 004767 173314 JSR PC,CMDAC ;

2533 015550 004767 174300 JSR PC,EXCUTE ;RE-WRITE RECORD

2534 015554 004767 174604 JSR PC,GOWAIT ;

2535 015560 004767 001714 JSR PC,CKHAE ;

2536 015564 000207 RTS PC

```

2537
2538
2539
2540
2541
2542
2543
2544
2545 015566 105767 165674
2546 015572 001126
2547 015574 105267 165666
2548 015600 016704 165622
2549 015604 005704
2550 015606 001520
2551 015610 162704 000002
2552 015614 010502
2553 015616 066402 016052
2554 015622 062702 002626
2555 015626 066712 165564
2556 015632 026767 164522 165556
2557 015640 101002
2558 015642 166712 164512
2559
2560 015646
2561 015646 010203
2562 015650 062703 000010
2563
2564 015654
2565 015654 021227 001747
2566 015660 003404
2567 015662 162712 001750
2568 015666 005213
2569
2570 015670 000771
2571 015672
2572 015672 010302
2573 015674 062702 000010
2574 015700
2575 015700 021327 001747
2576 015704 003404
2577 015706 162713 001750
2578 015712 005212
2579
2580 015714 000771
2581 015716
2582 015716 010203
2583 015720 062703 000010
2584 015724
2585 015724 021227 001747
2586 015730 003404
2587 015732 162712 001750
2588 015736 005213
2589
2590 015740 000771
2591 015742
2592 015742 105767 165521
2593 015746 001440

```

: SUBROUTINE TO LOG BYTES READ/WITTEN.
 : ALSO UPDATES READ/WRITE ERROR COUNTERS.
 : INPUTS:
 : OUTPUTS:
 : REGISTERS: R2, R3, R4.
 : CALLS:

```

LOG:: TSTB ERLOG ;IF DATA AND ERRORS HAVE NOT BEEN LOGGED THEN:
      BNE 50233$
      INCB ERLOG ;SET LOG DONE FLAG.
      MOV CMDLG,R4 ;GET CURRENT CMD LOGGING CODE.
      TST R4 ;IF THERE IS A CODE THEN:
      BEQ 50234$
      SUB #2,R4 ;ADJUST THE CODE FOR TABLE INDEX.
      MOV R5,R2 ;R2 = ADR OF BYTE COUNT LSW
      ADD BINC(R4),R2
      ADD @CNTBGN,R2
      ADD BRFCNT,(R2) ;ADD BRFCNT TO LSW.
      CMP MSGPKT*MS.RFC,BRFCNT ;IF THE RFC IS LOWER OR THE SAME AS BRFC THEN
      BHI 50235$
      SUB MSGPKT*MS.RFC,(R2) ;SUBTRACT RFC FROM EXPECTED BRFC.

50235$: MOV R2,R3 ;R3 = ADR OF 2ND WORD.
      ADD #10,R3

50236$: ;WHILE (R2) GT #999. DO
      CMP (R2),#999.
      BLE 50237$
      SUB #1000.,(R2) ;UPDATE BYTE COUNT
      INC (R3) ;LET (R3) := (R3) + #1 ;2ND WORD.
      BR 50236$

50237$: MOV R3,R2 ;LET R2 := R3 + #10 ;R2 = ADR OF 3RD WORD.
      ADD #10,R2

50240$: ;WHILE (R3) GT #999. DO
      CMP (R3),#999.
      BLE 50241$
      SUB #1000.,(R3) ;UPDATE BYTE COUNT
      INC (R2) ;LET (R2) := (R2) + #1 ;3RD WORD.
      BR 50240$

50241$: MOV R2,R3 ;LET R3 := R2 + #10 ;R3 = ADR OF 4TH WORD.
      ADD #10,R3

50242$: ;WHILE (R2) GT #999. DO
      CMP (R2),#999.
      BLE 50243$
      SUB #1000.,(R2) ;UPDATE BYTE COUNT
      INC (R3) ;LET (R3) := (R3) + #1 ;4TH WORD.
      BR 50242$

50243$: TSTB RWERR ;IF R/W ERROR, UPDATE ERROR COUNT.
      BEQ 50244$

```

```

2594 015750 010502      MOV      R5,R2                ;R2 = ADR OF COUNTER.
2595 015752 066402 016060  ADD      EINC(R4),R2
2596 015756 062702 002766  ADD      @WRREC,R2
2597 015762 105767 165502  TSTB     UNREC                ;IS THE ERROR UNRECOVERABLE?
2598 015766 001404      BEQ      502451
2599 015770 062702 000010  ADD      #10,R2                ;YES, POINT TO NEXT COUNTER.
2600 015774 005212      INC      (R2)                ;UPDATE THE ERROR COUNTER
2601                                     ;ELSE - IF ERROR IS RECOVERABLE:
2602 015776 000424      BR       502461
2603 016000      502451:  INC      (R2)                ;UPDATE THE ERROR COUNTER
2604 016000 005212      TSTB     IREC                ;IF ERROR RECOVERY IS ENABLD:
2605 016002 105767 164205  BNE      502471
2606 016006 001020      TSTB     DROPED                ;IF UNIT HAS NOT BEEN DROPPED:
2607 016010 105767 165506  BNE      502501
2608 016014 001015      TSTB     ERCVER
2609 016016 105767 164165  BEQ      502501
2610 016022 001412      PRINTB   @NURTY1,RETRYC        ;PRINT # OF RETRIES TO RECOVER
2611 016024      MOV      RETRYC, (SP)
      016024 016746 165430      MOV      @NURTY1, (SP)
      016030 012746 005422      MOV      #2, -(SP)
      016034 012746 000002      MOV      SP,R0
      016040 010600      TRAP     C$PNTB
      016042 104414      ADD      #6,SP
      016044 062706 000006      ;PROVIDED PRINT HAS BEEN ENABLED

2612                                     ;
2613 016050      502501:
2614                                     ;
2615 016050      502471:
2616                                     ;
2617                                     ;
2618 016050      502461:
2619                                     ;
2620 016050      502441:
2621                                     ;
2622 016050      502341:
2623                                     ;
2624 016050      502331:
2625 016050 000207      RTS      PC
2626                                     ;
2627                                     ; INDEXES TO BYTE COUNTERS.
2628 016052 000000      BINC:    0                ;WRITE.
2629 016054 000040      40                ;READ REV.
2630 016056 000100      100               ;READ FWD.
2631                                     ; INDEXES TO READ/WRITE ERROR COUNTERS.
2632 016060 000000      EINC:    0                ;WRITE.
2633 016062 000020      20                ;READ REV.
2634 016064 000040      40                ;READ FWD.
2635                                     ;
2636                                     ; IF A WRITE/VERIFY COMMAND IS ISSUED, CONTROL IS THEN
2637                                     ; TRANSFERRED TO THIS SUBROUTINE TO READ REVERSE, CHECK DATA.
2638                                     ; READ FORWARD, CHECK DATA, THEN CONTINUE TO NEXT COMMAND.
2639                                     ; INPUTS:
2640                                     ; OUTPUTS:
2641                                     ; REGISTERS:
2642                                     ; CALLS:          VFEXC.
2643
2644 016066 105767 165424  VFYDAT:;TSTB   VFYFLG        ;IF DATA IS TO BE VERIFIED:

```

```

2645 016072 001426      BEQ      502514
2646 016074 016767 165320 165322      MOV      CMDWRD,PCMDWD      ;SAVE THE PREVIOUS COMMAND WORD.
2647 016102 012767 104401 165310      MOV      @RDR,CMDWRD      ;COMMAND IS READ REV.
2648 016110 012767 000004 165310      MOV      #4,CMDLG      ;SET UP CMD LOGGING INDEX.
2649 016116 004767 000030      JSR      PC,VFEXC      ;GO READ ALL THE RECORDS REV.
2650 016122 016767 165272 165274      MOV      CMDWRD,PCMDWD      ;SAVE THE PREVIOUS COMMAND WORD.
2651 016130 012767 104001 165262      MOV      @RDF,CMDWRD      ;COMMAND IS READ FWD.
2652 016136 012767 000006 165262      MOV      #6,CMDLG      ;SET UP CMD LOGGING INDEX.
2653 016144 004767 000002      JSR      PC,VFEXC      ;GO READ ALL RECORDS FWD.
2654
2655 016150      502514:
2656 016150 000207      RTS      PC      ;RETURN.
2657
2658
2659
2660
2661      ;      SUBROUTINE TO EXECUTE THE READ AND VERIFY, FORWARD OR REVERSE.
2662      ;      INPUTS:
2663      ;      OUTPUTS:
2664      ;      REGISTERS:      R2
2665      ;      CALLS:      CMDAC, FIRSTU, VFISU, NEXTU, CKHAE.
2666
2667 016152 016767 165242 164150 VFEXC:: MOV      CMDWRD,CMDPKT      ;COMMAND PACKET = READ REV OR FWD.
2668 016160 042767 004000 164142      BIC      @BRF.C,CMDPKT
2669 016166 105767 165326      TSTB     SWBFLG      ;IF BYTES ARE TO BE SWAPPED:
2670 016172 001403      BEQ      502524
2671 016174 052767 010000 164126      BIS      @SWB.C,CMDPKT      ;SET SWAB BIT IN CMD PACKET.
2672
2673 016202      502524:
2674 016202 016767 164122 165212      MOV      CMDPKT,CMDSAV      ;SAVE COMMAND PACKET 1ST WORD.
2675 016210 016767 165174 164114      MOV      DATARD,CMDPKT+CP.ADL      ;SAVE BUFFER START ADDRESS.
2676 016216 005067 165170      CLR      NCNT      ;CLEAR NUMBER OF OPERATIONS.
2677
2678 016222      502534: ;WHILE NCNT LT NCNT1 DO      ;WHILE THERE ARE RECORDS REMAINING:
2679 016222 026767 165164 165164      CMP      NCNT,NCNT1
2680 016230 002062      BGE      502544
2681 016232 004767 172626      JSR      PC,CMDAC      ;STORE CMD ASCII IN ERROR MSG.
2682 016236 004767 000640      JSR      PC,FIRSTU      ;SET UP FOR FIRST UNIT.
2683 016242      502554: ;WHILE DEVTBL(R5) NE #END DO ;WHILE THERE ARE DEVICES REMAINING:
2684 016242 026527 002604 177777      CMP      DEVTBL(R5),#END
2685 016250 001442      BEQ      502564
2686 016252 032767 000400 165140      BIT      @MOD.CO,CMDWRD      ;IF CMD IS REVERSE THEN:
2687 016260 001421      BEQ      502574
2688 016262 032765 000002 003502      BIT      @X0.BOT,EOTFLG(R5)      ;IF NOT AT BOT
2689 016270 001014      BNE      502604
2690 016272 032765 000001 003502      BIT      @X0.EOT,EOTFLG(R5)      ;BUT IF AT EOT
2691 016300 001406      BEQ      502614
2692 016302 105767 165216      TSTB     ALLEOT      ;AND ALL OTHERS AT EOT
2693 016306 001402      BEQ      502624
2694 016310 004767 000064      JSR      PC,VFISU      ;THEN READ VERIFY
2695      ;IF NOT ALL AT EOT, FREEZE UNIT(S)
2696 016314      502624:
2697      ;IF NOT AT BOT AND
2698 016314 000402      BR      502634
2699 016316      502614:
2700 016316 004767 000056      JSR      PC,VFISU      ;NOT AT EOT, READ VFY
2701

```

```

2702 016322      502634:
2703
2704 016322      502604:
2705 016322 000412      BR      502644      ;ELSE IF CMD IS NOT REVERSE:
2706 016324      502574:
2707 016324 032765 000001 003502      BIT      @X0.EOT,EOTFLG(R5)
2708 016332 001404      BEQ      502654
2709 016334 032767 000001 165056      BIT      @CMD.CO,CMDWRD
2710 016342 001002      BNE      502664
2711 016344      502654:
2712      ;IF NOT AT EOT OR NOT A MOTION CMD THEN:
2713 016344 004767 000030      JSR      PC,VFISU      ;ISSUE CMD, CHECK STATUS AND DATA.
2714
2715 016350      502664:
2716
2717 016350      502644:
2718 016350 004767 000574      JSR      PC,NEXTU      ;GO FIND THE NEXT UNIT.
2719
2720 016354 000732      BR      502554
2721 016356      502564:
2722 016356 004767 001116      JSR      PC,CKHAE      ;CHECK FOR HALT AFTER EACH CMD.
2723 016362 005267 165024      INC      NCNT      ;UPDATE THE RECORD COUNT.
2724 016366 016767 165026 165030      MOV      CMDWRD,PCMDWD      ;SAVE PREVIOUS COMMAND WORD.
2725
2726 016374 000712      BR      502534
2727 016376      502544:
2728 016376 000207      RTS      PC      ;RETURN.
2729
2730      ; SUBROUTINE TO ISSUE COMMAND, AWAIT INTERRUPT,
2731      ; CHECK STATUS, CHECK DATA.
2732      ; INPUTS:
2733      ; OUTPUTS:
2734      ; REGISTERS:      R2
2735      ; CALLS:      EXECUTE, GOWAIT, CKDATA.
2736
2737 016400 016702 165004      VFISU:: MOV      DATARD,R2      ;INIT READ BUFFER POINTER.
2738 016404 062702 000010      ADD      @B.,R2
2739 016410      502674:      ;WHILE R2 NE DATARD DO      ;UNTIL 8 BYTES HAVE BEEN SET.
2740 016410 020267 164774      CMP      R2,DATARD
2741 016414 001403      BEQ      502704
2742 016416 012742 177777      MOV      @-1,-(R2)      ;INIT READ BUFFER.
2743
2744 016422 000772      BR      502674
2745 016424      502704:
2746 016424 004767 173424      JSR      PC,EXECUTE      ;GO EXECUTE THE COMMAND.
2747 016430 105767 165066      TSTB     DROPED      ;IF UNIT HAS NOT BEEN DROPPED THEN:
2748 016434 001002      BNE      502714
2749 016436 004767 173722      JSR      PC,GOWAIT      ;GO WAIT FOR DONE BIT.
2750
2751 016442      502714:
2752 016442 105767 165054      TSTB     DROPED      ;IF UNIT HAS NOT BEEN DROPPED THEN:
2753 016446 001006      BNE      502724
2754 016450 032765 000002 003502      BIT      @X0.BOT,EOTFLG(R5)      ;WHEN NOT REVERSED INTO BOT, THEN
2755 016456 001002      BNE      502734
2756 016460 004767 000002      JSR      PC,CKDATA      ;GO VERIFY DATA.
2757
2758 016464      502734:

```

```

2759
2760 016464
2761 016464 000207
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771 016466 016703 164724
2772 016472 166703 163662
2773 016476 005703
2774 016500 001015
2775 016502
    016502 104456
    016504 000021
    016506 004430
    016510 005752
2776 016512
    016512 012746 005337
    016516 012746 000001
    016522 010600
    016524 104414
    016526 062706 000004
2777 016532 000560
2778 016534
2779 016534 020367 164656
2780 016540 101417
2781 016542
    016542 104456
    016544 000021
    016546 004430
    016550 005752
2782 016552
    016552 016746 163560
    016556 012746 005360
    016562 012746 000002
    016566 010600
    016570 104414
    016572 062706 000006
2783
2784 016576 000536
2785 016600
2786 016600 010367 000272
2787 016604 005367 000266
2788 016610 005067 000264
2789 016614 005002
2790 016616 016703 164564
2791 016622 016704 164562
2792 016626 105767 164671
2793 016632 001401
2794 016634 000313
2795
2796 016636

502724: RTS PC

; SUBROUTINE TO COMPARE DATA BETWEEN READ AND WRITE BUFFERS
; AND PRINT ERROR MESSAGE ON MISCOMPARE.
; INPUTS:
; OUTPUTS:
; REGISTERS: R2, R3, R4.
; CALLS: GCMDB

CKDATA: MOV BRCNT,R3 ; COMPUTE REC LENGTH READ
        SUB MSGPKT+MS.RFC,R3
        TST R3 ; WHEN NO DATA RECEIVED
        BNE 502744
        ERHRD 17,WTVERM,DTAERM ; PRINT ERROR AND EXIT
                                TRAP C$ERHRD
                                .WORD 17
                                .WORD WTVERM
                                .WORD DTAERM

        PRINTB #DTAER4 ; COMPARE ROUTINE
                                MOV #DTAER4,-(SP)
                                MOV #1,-(SP)
                                MOV SP,R0
                                TRAP C$PNTB
                                ADD #4,SP

BR 502754

502744: CMP R3,BRCNT ; WHEN REC READ IS LONGER
        BLOS 502764
        ERHRD 17,WTVERM,DTAERM ; THAN EXPECTED, PRINT
                                TRAP C$ERHRD
                                .WORD 17
                                .WORD WTVERM
                                .WORD DTAERM

        PRINTB #DTAERS,CMDPKT+CP.CNT ; AN ERROR MESSAGE
                                MOV CMDPKT+CP.CNT,-(SP)
                                MOV #DTAERS,-(SP)
                                MOV #2,-(SP)
                                MOV SP,R0
                                TRAP C$PNTB
                                ADD #6,SP

; AND EXIT ROUTINE
BR 502774

502764: MOV R3,CKDCNT ; SAVE VERIFICATION LENGTH - 1.
        DEC CKDCNT
        CLR CKDFF ; CLEAR # OF BYTES IN ERROR COUNTER.
        CLR R2 ; INIT BYTE COUNTER
        MOV DATAWT,R3 ; GET WRITE BUFFER ADDRESS.
        MOV DATARD,R4 ; GET READ BUFFER ADDRESS.
        TSTB T1SWB ; WHEN RUNNING TEST1 SUB 12.
        BEQ 503004
        SWAB (R3) ; SWAP FIRST WORD OF WRT BFR
                                ; WHICH CONTAINS THE RECORD COUNT

503004:

```

```

2797                                ;REPEAT                                ;REPEAT UNTIL ALL DATA IS COMPARED:
2798 016636 50301$: CMP R2,CKDCNT                                ;IF THIS IS THE LAST BYTE THEN:
2799 016636 020267 000234 BNE 50302$                                ;IF BYTE SWAPPING IS ENABLED THEN:
2800 016642 001011 TSTB SWBFLG                                ;IF RECORD LENGTH IS ODD THEN:
2801 016644 105767 164650 BEQ 50303$                                ;LAST BYTE WILL BE IN
2802 016650 001406 BIT #BIT00,CKDCNT                                ;THE UPPER BYTE.
2803 016652 032767 000001 000216 BNE 50304$
2804 016660 001002 TSTB (R3)+
2805 016662 105723 TSTB (R4)+
2806 016664 105724
2807
2808 016666 50304$:
2809
2810 016666 50303$:
2811
2812 016666 50302$:
2813 016666 121314 CMPB (R3),(R4)                                ;ARE THEY EQUAL.
2814 016670 001452 BEQ 3$                                ;BR IF SO.
2815 016672 005767 000202 TST CKDFF                                ;1 ST TIME THRU?
2816 016676 001010 BNE 2$                                ;BR IF NOT.
2817 016700 005265 003346 INC VFYCNTR(R5)                                ;INC THE VERIFY ERROR COUNTER.
2818 016704 005265 003356 INC HRCNTR(R5)                                ;INC THE HARD ERROR COUNT.
2819 016710 ERRHRD 17,WTVERM,DTAERM                                ;REPORT WRITE/VERIFY ERROR.
2820 016710 104456 TRAP C$ERRHRD
2821 016712 000021 .WORD 17
2822 016714 004430 .WORD WTVERM
2823 016716 005752 .WORD DTAERM
2824 016720 005267 000154 2$: INC CKDFF;LET CKDFF := CKDFF + #1 ;INCREMENT # OF BYTES IN ERROR.
2825 016724 111467 164506 MOVB (R4),TIME1                                ;SAVE WAS DATA FOR TYP0UT.
2826 016730 042767 177400 164500 BIC #177400,TIME1                                ;CLEAR GARBAGE.
2827 016736 111367 164476 MOVB (R3),TIME2                                ;SAVE SHOULD BE DATA FOR TYP0UT.
2828 016742 042767 177400 164470 BIC #177400,TIME2                                ;CLEAR GARBAGE.
2829 016750 026727 000124 000013 CMP CKDFF,#11.                                ;IF ERROR BYTE COUNT IS LESS THAN 11:
2830 016756 002017 BGE 50305$
2831 016760 PRINTX #DTAER2,R2,<B,TIME1>,<B,TIME2>;PRINT ACTUAL & EXPECTED DATA
2832 016760 005046 CLR -(SP)
2833 016762 156716 164452 BISB TIME2,(SP)
2834 016766 005046 CLR -(SP)
2835 016770 156716 164442 BISB TIME1,(SP)
2836 016774 010246 MOV R2,-(SP)
2837 016776 012746 005226 MOV #DTAER2,-(SP)
2838 017002 012746 000004 MOV #4,-(SP)
2839 017006 010600 MOV SP,R0
2840 017010 104415 TRAP C$PNTX
2841 017012 062706 000012 ADD #12,SP
2842 017016 50305$:
2843
2844 3$: TSTB (R3)+                                ;UPDATE WRITE BUFFER ADDRESS.
2845 017020 105724 TSTB (R4)+                                ;UPDATE READ BUFFER ADDRESS.
2846 017022 105722 TSTB (R2)+                                ;UPDATE BYTE COUNTER.
2847 017024 020267 000046 CMP R2,CKDCNT                                ;END OF DATA COMPARE REPEAT LOOP.
2848 017030 003702 BLE 50301$
2849 017032 005267 000040 INC CKDCNT                                ;CKDCNT EQUALS RECORD LENGTH.
2850 017036 005767 000036 TST CKDFF                                ;IF COMPARE ERROR HAS OCCURED THEN:
2851 017042 001414 BEQ 50306$
2852 017044 PRINTB #DTAER3,CKDFF,CKDCNT                                ;PRINT # OF BYTES IN ERROR.
2853 017046 016746 000026 MOV CKDCNT,-(SP)

```

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0097

```

017050 016746 000024
017054 012746 005275
017060 012746 000003
017064 010600
017066 104414
017070 062706 000010

2839
2840 017074 50306$:
2841
2842 017074 50277$:
2843
2844 017074 50275$:
2845 017074 000207 RTS PC ; OTHERWISE, RETURN.
2846
2847 017076 000000 CKDCNT: .WORD 0 ; # OF BYTES TO BE VERIFIED -1.
2848 017100 000000 CKDFF: .WORD 0 ; # OF BYTES IN ERROR COUNTER.
2849
2850 ; SUBROUTINE TO FIND THE FIRST DEVICE IN THE TEST SEQUENCE.
2851 ;
2852 ; INPUTS:
2853 ; OUTPUTS:
2854 ; REGISTERS:
2855 ; CALLS:
2856 017102 105067 164414 FIRSTU:: CLRB DROPED ; CLR UNIT DROPPED FLAG
2857 017106 005005 CLR R5 ; CLR DEVICE POINTER.
2858 017110 026527 002604 177774 50307$: CMP DEVTBL(R5),#NINUSE ; WHILE DEVICES ARE NOT IN USE:
2859 017116 001003 BNE 50310$
2860 017120 062705 000002 ADD #2,R5 ; LET R5 := R5 + #2 ; POINT TO NEXT DEVICE.
2861 017124 000771 BR 50307$
2862 017126 50310$:
2863 017126 026527 002604 177777 CMP DEVTBL(R5),#END ; IF ALL UNITS HAVE BEEN DROPPED THEN:
2864 017134 001001 BNE 50311$
2865 017136 DOCLN ; DO CLEAN CODE AND TERMINATE PASS.
2866 017136 104444 TRAP C#DOCLN
2867
2868 017140 016567 002604 162726 50311$: MOV DEVTBL(R5),L#LUN ; SET UNIT # IN "HEADER" FOR ERROR REPORT
2869 017146 000207 RTS PC ; RETURN WITH 1ST DEVICE IN R5.
2870
2871
2872 ; SUBROUTINE TO FIND THE NEXT UNIT IN THE TEST CYCLE.
2873 ;
2874 ; INPUTS:
2875 ; OUTPUTS:
2876 ; REGISTERS:
2877 ; CALLS:
2878 017150 105067 164346 NEXTU:: CLRB DROPED ; CLR UNIT DROPPED FLAG
2879 ; REPEAT ; REPEAT UNTIL THE NEXT DEVICE IS FOUND.
2880 017154 50312$:
2881 017154 062705 000002 ADD #2,R5 ; UPDATE DEVICE TABLE POINTER.
2882 017160 026527 002604 177774 CMP DEVTBL(R5),#NINUSE ; UNTIL DEVTBL(R5) NE #NINUSE
2883 017166 001772 BEQ 50312$
2884 017170 016567 002604 162676 MOV DEVTBL(R5),L#LUN ; SET UNIT # IN "HEADER" FOR ERROR REPORT
2885 017176 000207 RTS PC ; RETURN.
2886
2887
2888 ; SUBROUTINE TO DROP A DEVICE FROM THE TEST SEQUENCE.

```

```

2889      ;      INPUTS:
2890      ;      OUTPUTS:
2891      ;      REGISTERS:
2892      ;      CALLS:      MOVMSG, PRXST, LOG
2893
2894 017200 005265 003366      DROPU:: INC      FTLCNT(R5)      ;INCREMENT THE FATAL ERROR COUNT.
2895 017204 016704 163160      MOV      MSGPKT+MS.XS3,R4      ;GET UDIAG ERROR CODE FROM XSTAT3.
2896 017210 042704 000377      BIC      #377,R4
2897 017214 016503 002544      MOV      MSGPKA(R5),R3      ;ADR OF THIS UNIT'S MSG PACKET.
2898 017220 005002      CLR      R2      ;LET R2 := #0      ;CLR COUNTER.
2899 017222      50313$: WHILE R2 NE #MSGCNT D0      ;WHILE THERE ARE MORE LOCATIONS:
2900 017222 020227 000020      CMP      R2,#MSGCNT
2901 017226 001405      BEQ      50314$
2902 017230 012723 177777      MOV      #1,(R3)+      ;INIT THE MSG PACKET WITH ALL 1 S
2903 017234 062702 000002      ADD      #2,R2      ;LET R2 := R2 + #2      ;UPDATE COUNTER.
2904
2905 017240 000770      BR      50313$
2906 017242      50314$: MOV      #GSCPK,#TSDB(R5)      ;INITIATE A GET STATUS COMMAND.
2907 017242 012775 002340 002514      JSR      PC,WSSR      ;WAIT A WHILE FOR SSR=1
2908 017250 004767 173424      JSR      PC,MOVMSG      ;MOVE MSG PACKET TO COMMON AREA.
2909 017254 004767 173454      CMP      R4,#X3.RNY      ;IF WE HAVE A CAPSTAN RUNAWAY THEN:
2910 017260 020427 157400      BNE      50315$
2911 017264 001005      ERDF      16,RNYM,STAERM      ;REPORT CAPSTAN RUNAWAY WITH TACH CNT.
2912 017266      TRAP      C#ERDF
2913      .WORD      16
2914      .WORD      RNYM
2915      .WORD      STAERM
2916      ;ELSE-IF NOT A RUNAWAY:
2917      BR      50316$
2918 017276 000402      50315$: JSR      PC,PRXST      ;PRINT EXTENDED STATUS REGISTERS.
2919 017300 004767 000106      50316$: TSTB      RECLOG      ;IF THE RECORD HAS BEEN LOGGED THEN:
2920 017304 105767 164155      BEQ      50317$
2921 017310 001404      INCB      DROPED      ;SET UNIT DROPPED FLAG.
2922 017312 105267 164204      JSR      PC,LOG      ;LOG DATA BYTES + RD/WR ERRORS.
2923 017316 004767 176244
2924 017322      50317$: DORPT      ;PRINT PERFORMANCE REPORT
2925 017322      TRAP      C#DRPT
2926 017322 104424      DROPUA: TST      PASCNT(R5)      ;IF PASCNT(R5) NE #0 THEN
2927 017324 005765 003326      BEQ      50320$
2928 017330 001402      DEC      PASCNT(R5)      ;LET PASCNT(R5) := PASCNT(R5) - #1
2929 017332 005365 003326      50320$: MOV      TSMP,DROPN      ;SAVE # OF UNIT TO BE DROPPED.
2930 017336      MOV      TSMP,R0      ;R0=LOGICAL DEVICE NUMBER
2931 017336 016767 164172 000044      DODU      R0      ;DROP THE UNIT
2932 017344 016700 164164      TRAP      C#DODU
2933 017350      ;EXEC BGNDU-ENDDU CODE IF IDU = 0
2934 017350 104451
2935
2936 017352 026527 002604 177774      CMP      DEVTBL(R5),#NINUSE      ;IF UNIT NOT DROPPED
2937 017360 001410      BEQ      50321$
2938 017362 105767 162625      TSTB      IREC      ;IF RECOVERY IS ENABLED THEN:
2939 017366 001005      BNE      50322$

```

GLOBAL AREAS MACRO M1113 14 JUN 84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0099

```

2940 017370 000240      NOP
2941 017372 000240      NOP
2942 017374 000240      NOP
2943 017376 105267 164124      INCB      STAFLG      ;SET START FLAG TO ENABLE REWIND.
2944
2945 017402      50322$:
2946
2947 017402      50321$:
2948 017402 105267 164114      DRORTN: INCB      DROPED      ;SET UNIT DROPPED FLAG.
2949 017406 000207      RTS      PC      ;RETURN.
2950
2951 017410 000000      DROPN: .WORD 0      ;# OF UNIT TO BE DROPPED
2952
2953      ;      SUBROUTINE TO PRINT EXTENDED STATUS REGISTERS.
2954      ;      INPUTS:
2955      ;      OUTPUTS:
2956      ;      REGISTERS:
2957      ;      CALLS:
2958
2959 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$PNTX
      017426 062706 000004      ADD      #4,SP
4 2960 017432      PRINTX #STAERS,MSGPKT+MS.XS0,MSGPKT+MS.XS1,MSGPKT+MS.XS2,MSGPKT+MS.XS3,MSGPKT+MS.XS
      017432 016746 162734      MOV      MSGPKT+MS.XS4,-(SP)
      017436 016746 162726      MOV      MSGPKT+MS.XS3,-(SP)
      017442 016746 162720      MOV      MSGPKT+MS.XS2,-(SP)
      017446 016746 162712      MOV      MSGPKT+MS.XS1,-(SP)
      017452 016746 162704      MOV      MSGPKT+MS.XS0,-(SP)
      017456 012746 006753      MOV      #STAERS,-(SP)
      017462 012746 000006      MOV      #6,-(SP)
      017466 010600      MOV      SP,R0
      017470 104415      TRAP      C$PNTX
      017472 062706 000016      ADD      #16,SP
2961 017476 000207      RTS PC
2962
2963      ;      SUBROUTINE TO HALT AFTER EACH COMMAND.
2964      ;      INPUTS:
2965      ;      OUTPUTS:
2966      ;      REGISTERS:      R3, R4
2967      ;      CALLS:
2968
2969 017500 105767 162502      CKHAE:: TSTB      HAE,IFB HAE NE #0 THEN      ;IF HALT FLAG IS SET:
2970 017504 001430      BEQ      50323$
2971 017506 105767 164017      TSTB      MISCFG      ;IFB MISCFG EQ #0 THEN      ;
2972 017512 001023      BNE      50324$
2973 017514      MANUAL      ;IS MANUAL INTERVENTION ALLOWED?
      017514 104450      TRAP      C$MANI
2974 017516      BNCOMPLETE CKHRTN      ;BR IF NOT.
      017516 103023      BCC      CKHRTN
2975 017520 016704 163674      MOV      CMDWRD,R4      ;LET R4 := CMDWRD
2976 017524 004767 171406      JSR      PC,GCMDA      ;FETCH ADR OF CMD ASCII.
2977 017530 112367 164552      MOV      (R3)+,HALTM      ;MOVE CMD ASCII
2978 017534 112367 164547      MOV      (R3)+,HALTM+1      ;LET HALTM+1 := (R3)+
2979 017540 111367 164544      MOV      (R3),HALTM+2      ;INTO MESSAGE.

```

GLOBAL AREAS MACRO M1113 14 JUN-84 18:32
GLOBAL SUBROUTINES SECTION

SEQ 0100

```

2980 017544          GMANIL HALTM,TIME1,1,YES      ;HALT  WAIT FOR AN OEPRAOTR INPUT.
      017544 104443          TRAP      C$GMAN
      017546 000404          BR        10000$
      017550 003436          .WORD    TIME1
      017552 000130          .WORD    T$CODE
      017554 004306          .WORD    HALTM
      017556 000001          .WORD    1
      017560          10000$:
2981 017560          10000$:
2982
2983 017560 000402          BR        50325$
2984 017562          50324$:
2985 017562 105067 163743          CLR8      MISCFG      ;LET MISCFG :B= #0      ;
2986
2987 017566          50325$:
2988
2989 017566          50323$:
2990 017566 000207          CKHRTN: RTS      PC      ;RETURN
2991                          .EVEN
2992
2993 017570          ENDMOD
2994
2995
2996          .TITLE MISCELLANEOUS SECTIONS
2997          .SBTTL  REPORT CODING SECTION
2998
2999
3000
3001          ;**
3002          ; THE REPORT CODING SECTION CONTAINS THE
3003          ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
3004          ;--
3005 017570          BGNRPT
      017570          L$RPT::
3006 017570 010567 163656          MOV      R5,R$SAVE          ;SAVE CURRENT DEVICE POINTER.
3007 017574 004767 177302          JSR      PC,FIRSTU          ;FIND THE FIRST UNIT.
3008 017600          50326$: ;WHILE DEVTBL(R5) NE #END DO      ;WHILE THERE ARE MORE DEVICES:
3009 017600 026527 002604 177777          CMP      DEVTBL(R5),#END
3010 017606 001562          BEQ      50327$
3011 017610          PRINTS      @RPT1A,DEVTBL(R5),PASCNT(R5),RECCNT(R5)
      017610 016546 003376          MOV      RECCNT(R5),-(SP)
      017614 016546 003326          MOV      PASCNT(R5),-(SP)
      017620 016546 002604          MOV      DEVTBL(R5),-(SP)
      017624 012746 020432          MOV      @RPT1A, -(SP)
      017630 012746 000004          MOV      #4, -(SP)
      017634 010600          MOV      SP,R0
      017636 104416          TRAP      C$PNTS
      017640 062706 000012          ADD      #12,SP
3012 017644          PRINTS      @RPT1B,WRBC+30(R5),WRBC+20(R5),WRBC+10(R5),WRBC(R5)
      017644 016546 002626          MOV      WRBC(R5),-(SP)
      017650 016546 002636          MOV      WRBC+10(R5),-(SP)
      017654 016546 002646          MOV      WRBC+20(R5),-(SP)
      017660 016546 002656          MOV      WRBC+30(R5),-(SP)
      017664 012746 020507          MOV      @RPT1B, -(SP)
      017670 012746 000005          MOV      #5, -(SP)
      017674 C10600          MOV      SP,R0
      017676 104416          TRAP      C$PNTS

```

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1113 14-JUN-84 18:32

SEQ 0101

3013	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
3014	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	002756				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
3015	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	002766				MOV	WRREC(R5),-(SP)
	020020	012746	020735				MOV	#RPT1F, -(SP)
	020024	012746	000004				MOV	#4, -(SP)
	020030	010600					MOV	SP,R0
	020032	104416					TRAP	C#PNTS
3016	020034	062706	000012				ADD	#12,SP
	020040			PRINTS	#RPT1G,WRUNR(R5),RRUNR(R5),RFUNR(R5)		MOV	RFUNR(R5),-(SP)
	020044	016546	003036				MOV	RRUNR(R5),-(SP)
	020050	016546	002776				MOV	WRUNR(R5),-(SP)
	020054	012746	021006				MOV	#RPT1G, -(SP)
	020060	012746	000004				MOV	#4, -(SP)
	020064	010600					MOV	SP,R0
	020066	104416					TRAP	C#PNTS
	020070	062706	000012				ADD	#12,SP
3017	020074	105767	162110	TSTB	BADTSW	;IFB BADTSW NE #0 THEN		
3018	020100	001402		BEQ	503304			
3019	020102	004767	000056		JSR PC,BTRPT	;GO PRINT BAD TAPE SPOTS WHEN ENABLED		
3020								
3021	020106			503304:				
3022	020106				PRINTS	#RPT1I,SCCNT(R5),HRDCNT(R5),FTLCNT(R5),VFYCNT(R5)		
	020106	016546	003346				MOV	VFYCNT(R5),-(SP)
	020112	016546	003366				MOV	FTLCNT(R5),-(SP)
	020116	016546	003356				MOV	HRDCNT(R5),-(SP)
	020122	016546	003336				MOV	SCCNT(R5),-(SP)
	020126	012746	021203				MOV	#RPT1I, -(SP)
	020132	012746	000005				MOV	#5, -(SP)
	020136	010600					MOV	SP,R0
	020140	104416					TRAP	C#PNTS
3023	020142	062706	000014				ADD	#14,SP
3024	020146	004767	176776	JSR	PC,NEXTU	;FIND THE NEXT UNIT.		
3025	020152	000612		BR	503264			

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1113 14 JUN-84 18:32

SEQ 0102

```

3026 020154      50327$:
3027 020154      016705 163272      MOV      R5SAVE,R5          ;RESTORE CURRENT DEVICE POINTER.
3028 020160      020160 000167      EXIT      RPT
      020162 001130      .WORD      J$JMP
      .WORD      L10010 2 .

3029
3030      ;      SUBR TO PRINT BAD TAPES SPOTS DURING THE REPORT PRINTS
3031      ;      WRITE RETRIES: CUMULATIVE COUNT
3032      ;      BAD TAPE SPOTS: COUNT PER TAPE PASS ONLY, NOT CUMULATIVE.
3033      ;      COUNT OF RECOVERABLE WRITE ERRORS EXCLUDES BAD TAPE SPOTS.
3034
3035 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$PNTS
      020204 062706 000006      ADD      @6,SP
3036 020210 016567 002616 163274      MOV      BTADDR(R5),BTPT      ;BTPT IS BOTH THE BAD TAPE SPOT COUNTER
3037 020216 017703 163270      MOV      @BTPT,R3      ;AND THE LOGGING INDEX
3038 020222 006203      ASR      R3
3039 020224      PRINTS  @RPT1J,R3      ;PRINT # OF BAD TAPE SPOTS
      020224 010346      MOV      R3,-(SP)
      020226 012746 021107      MOV      @RPT1J,-(SP)
      020232 012746 000002      MOV      @2,-(SP)
      020236 010600      MOV      SP,R0
      020240 104416      TRAP      C$PNTS
      020242 062706 000006      ADD      @6,SP
3040 020246 005703      TST      R3      ;PRINT RECORD # IF BAD SPOTS DETECTED
3041 020250 001457      BEQ      50331$
3042 020252 020327 000024      CMP      R3,@20.      ;IF R3 HI @20. THEN
3043 020256 101402      BLOS     50332$
3044 020260 012703 000024      MOV      @20.,R3      ;20 BAD SPOTS IS THE LIMIT
3045
3046 020264      50332$:
3047 020264      PRINTS      @CRLFSP      ;
      020264 012746 005744      MOV      @CRLFSP,-(SP)
      020270 012746 000001      MOV      @1,-(SP)
      020274 010600      MOV      SP,R0
      020276 104416      TRAP      C$PNTS
      020300 062706 000004      ADD      @4,SP
3048 020304 016704 163202      MOV      BTPT,R4      ;LET R4 := BTPT + @2 ;FETCH A BAD SPOT ID
3049 020310 062704 000002      ADD      @2,R4
3050 020314 005002      CLR      R2      ;R2 = PRINT COUNT PER LINE: 10 MAX
3051 020316      50333$:      ;REPEAT
3052 020316      PRINTS      @RPT1K,(R4)      ;PRINT A BAD SPOT ID
      020316 011446      MOV      (R4),-(SP)
      020320 012746 021174      MOV      @RPT1K,-(SP)
      020324 012746 000002      MOV      @2,-(SP)
      020330 010600      MOV      SP,R0
      020332 104416      TRAP      C$PNTS
      020334 062706 000006      ADD      @6,SP
3053 020340 005202      INC      R2      ;LET R2 := R2 + @1 ;COUNT PRINTS
3054 020342 062704 000002      ADD      @2,R4      ;LET R4 := R4 + @2 ;NEXT
3055 020346 020227 000012      CMP      R2,@10.      ;IF R2 EQ @10. THEN
3056 020352 001014      BNE      50334$
3057 020354      PRINTS  @CRLFSP      ;GO TO NEXT PRINT LINE PAST 10 PRINTS

```

MISCELLANEOUS SECTIONS MACRO M1113 14 JUN-84 18:32
REPORT CODING SECTION

SEQ 0103

```

020354 012746 005744
020360 012746 000001
020364 010600
020366 104416
020370 062706 000004
3058 020374 162703 000012
3059 020400 162702 000012
3060
3061 020404
3062 020404 020203
3063 020406 001343
3064
3065 020410
3066 020410
020410 012746 005741
020414 012746 000001
020420 010600
020422 104416
020424 062706 000004
3067 020430 000207
3068
3069
3070 020432 045 116 045 RPT1A: .LIST BEX
3071 020507 045 101 102 RPT1B: .ASCIZ /#N#N#AUNIT #D1#S3#APASS:#D5#S3#ARECORD:#D5#N/
3072 020560 045 101 102 RPT1C: .ASCIZ /#ABYTES WRITTEN #D3#A,#Z3#A,#Z3#A,#Z3#N/
3073 020631 045 101 102 RPT1D: .ASCIZ /#ABYTES READ REV #D3#A,#Z3#A,#Z3#A,#Z3#N/
3074 020701 045 123 062 RPT1E: .ASCIZ /#S23#AURT#S4#ARDR#S4#ARDF#N/
3075 020735 045 101 122 RPT1F: .ASCIZ /#ARECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
3076 021006 045 101 125 RPT1G: .ASCIZ /#AUNRECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
3077 021057 045 101 127 RPT1H: .ASCIZ /#AWRITE RETRIES#S8#D5#N/
3078 021107 045 116 045 RPT1I: .ASCIZ /#N#D2#A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
3079 021174 045 104 065 RPT1J: .ASCIZ /#D5#S1/
3080 021203 045 101 123 RPT1K: .ASCIZ /#S3#D5#S3#D5#S3#D5#S3#D5#N#N/
3081 021257 045 123 063
3082
3083
3084
3085 021314
021314
021314 104425
3086
3087
3088
3089
3090
3091
3092
3093
3094 021316
021316
3095
3096 021316 000000
3097 021320 177777
3098 021322 177777
LLER
3099 021324
3100
3101

```

MOV #CRLFSP, -(SP)
 MOV #1, (SP)
 MOV SP, R0
 TRAP C#PNTS
 ADD #4, SP
 SUB #10., R3 ;LET R3 := R3 #10. ;ADJUST BAD SPOT COUNT
 SUB #10., R2 ;LET R2 := R2 #10. ;ADJUST PRINT COUNT
 ;
 50334\$: CMP R2, R3 ;UNTIL R2 EQ R3 ;LIMIT: # OF BAD SPOTS
 BNE 50333\$
 ;
 50331\$: PRINTS #CRLF ;
 ;
 MOV #CRLF, (SP)
 MOV #1, -(SP)
 MOV SP, R0
 TRAP C#PNTS
 ADD #4, SP
 RTS PC
 .LIST BEX
 RPT1A: .ASCIZ /#N#N#AUNIT #D1#S3#APASS:#D5#S3#ARECORD:#D5#N/
 RPT1B: .ASCIZ /#ABYTES WRITTEN #D3#A,#Z3#A,#Z3#A,#Z3#N/
 RPT1C: .ASCIZ /#ABYTES READ REV #D3#A,#Z3#A,#Z3#A,#Z3#N/
 RPT1D: .ASCIZ /#ABYTES READ FWD #D3#A,#Z3#A,#Z3#A,#Z3#N/
 RPT1E: .ASCIZ /#S23#AURT#S4#ARDR#S4#ARDF#N/
 RPT1F: .ASCIZ /#ARECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
 RPT1G: .ASCIZ /#AUNRECOVERABLE ERRORS #D5#S2#D5#S2#D5#N/
 RPT1H: .ASCIZ /#AWRITE RETRIES#S8#D5#N/
 RPT1I: .ASCIZ /#N#D2#A BAD SPOTS THIS TAPE PASS PRECEDING RECORD #:/
 RPT1J: .ASCIZ /#D5#S1/
 RPT1K: .ASCIZ /#S3#D5#S3#D5#S3#D5#S3#D5#N#N/
 .LIST BEX
 .EVEN
 ENDRPT
 L10010:
 TRAP C#RPT
 .SBTTL LOAD DEVICE PROTECTION TABLE
 ;**
 ;TABLE FOR SUPERVISOR TO IDENTIFY THE P-TBL FOR THE LOAD DEV
 ;THE SUPERVISOR USES THE TBL TO WARN THE OPERATOR WHEN HE TRIES TO TEST THE LOAD DEV
 ;
 BGNPROT
 L\$PROT::
 .WORD 0 ;P-TBL OFFSET OF TSOB
 .WORD 1 ;P-TBL OFFSET OF MASS BUS UNIT #: -1 = NOT A MASS BUS DE
 .WORD -1 ;P-TBL OFFSET OF DRVIVE #: 1 = NONE, THREE DRIVES PER CONTR
 ENDPROT
 .SBTTL INITIALIZE SECTION

```

3102
3103
3104      ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
3105      ; AT THE BEGINNING OF EACH PASS.
3106      ; -
3107
3108      BGNINIT
3109      L$INIT::
3110      021324 032727 000003 002330 INIT10: BIT    #BIT0:BIT1, #CMDPKT    ;IF CMD PACKET IS NOT ON MODULO 4 BOUNDARY:
3111      021332 001421      BEQ      50335$
3112      021334      ERRSF 1,CMDPKM      ;PRINT ERROR MSG.
3113      021334 104454      TRAP      C$ERSF
3114      021336 000001      .WORD    1
3115      021340 004346      .WORD    CMDPKM
3116      021342 000000      .WORD    0
3117      021344      DELAY 200.      ;GO TO SUPERVISOR, WAIT 2 SECONDS.
3118      021344 012727 000310      MOV      #200.,(PC).
3119      021350 000000      .WORD    0
3120      021352 016727 160540      MOV      L$DLY,(PC).
3121      021356 000000      .WORD    0
3122      021360 005367 177772      DEC      -6(PC)
3123      021364 001375      BNE      -4
3124      021366 005367 177756      DEC      -22(PC)
3125      021372 001367      BNE      -20
3126      021374 000753      BR      INIT10
3127
3128      50335$:
3129
3130      TSTB    CLRFLG      ;IF CLR COUNTERS FLAG SET:
3131      BEQ     50336$
3132      CLR     CLRFLG      ;INIT CLR FLAG.
3133      CLR     R2          ;LET R2 := #0
3134      50337$: WHILE R2 NE #CNTLEN DO
3135      CMP     R2,#CNTLEN
3136      BEQ     50340$
3137      CLR     WRBC(R2)    ;CLR ALL STATISTICAL COUNTERS.
3138      ADD     #2,R2      ;LET R2 := R2 + #2
3139      BR      50337$
3140
3141      50340$:
3142
3143      50336$:
3144
3145      TSTB    RRANV      ;IF RESET RANDOM VARIABLE FLAG IS SET THEN:
3146      BEQ     50341$
3147      MOV     #RANBC,RANB      ;RESET RANDOM BASE #.
3148      MOV     #RANSC,RANS      ;RESET RANDOM SAVE LOCATION.
3149
3150      50341$:
3151      READEF  #EF.START      ;READ START COMMAND EVENT FLAG.
3152      MOV     #EF.START,RO
3153      TRAP    C$REFG
3154      BNCOMPLETE      INIT15      ;BRANCH IF NOT STARTING.
3155      INCB     STAFLG      ;SET START COMMAND FLAG.

```

MISCELLANEOUS SECTIONS
INITIALIZE SECTION

MACRO M1113 14 JUN 84 18:32

SEQ 0105

```

3143 021470 012705 000006      MOV    #6,R5      ;LET R5 := #6
3144 021474      503421: ;REPEAT
3145 021474 012765 177774 002604  MOV    #NINUSE,DEVTBL(R5) ;INITIATE UNIT NUMBER TABLE
3146 021502 162705 000002      SUB    #2,R5      ;LET R5 := R5 - #2 ;BY STORING NOT IN USE IN EACH LOCATION.
3147 021506 005705      TST    R5      ;UNTIL R5 EQ #0
3148 021510 001371      BNE    503421
3149 021512 022767 000001 160272  CMP    #1,L#UNIT ;ONLY ONE UNIT ALLOWED
3150 021520 001425      BEQ    50341 ;OK
3151 021522      PRINTF    #AUDRUN ;TELL THE MAN

      MOV    #AUDRUN,(SP)
      MOV    #1,(SP)
      MOV    SP,R0
      TRAP    C#PNTF
      ADD    #4,SP

3152 021542      DELAY    25 ;WAIT
      MOV    #25,(PC)
      .WORD    0
      MOV    L#DLY,(PC)
      .WORD    0
      DEC    -6(PC)
      BNE    -4
      DEC    -22(PC)
      BNE    -20

3153 021572      DOCLN ;ABORT
      TRAP    C#DOCLN

3154 021574 016705 160212 50341: MOV    L#UNIT,R5 ;LET R5 := L#UNIT SHIFT 1
3155 021600 006305      ASL    R5
3156 021602      503431: ;REPEAT ;STORE ALL UNIT
3157 021602 162705 000002      SUB    #2,R5      ;LET R5 := R5 - #2 ;NUMBERS IN DEVTBL.
3158 021606 010565 002604      MOV    R5,DEVTBL(R5) ;LET DEVTBL(R5) := R5 SHIFT 1
3159 021612 006265 002604      ASR    DEVTBL(R5)
3160 021616 005705      TST    R5      ;UNTIL R5 EQ #0
3161 021620 001370      BNE    503431
3162
3163 021622      INIT15: READEF #EF.PWR ;HAS THERE BE A POWER FAILURE?
      MOV    #EF.PWR,R0
      TRAP    C#REFG
      021622 012700 000034
      021626 104447
      BNCOMplete INIT16 ;BRANCH IF NOT.
      BCC    INIT16
3164 021630      INCB    STAF LG ;IF SO - SET THE START FLAG.
      INCB    PWRFLG ;IF SO - SET THE POWER FAIL FLAG.
3165 021632 105267 161670
3166 021636 105267 161665
3167
3168 021642      INIT16: RFLAGS OPFLAG ;READ AND STORE FLAGS SET BY OPERATOR
      TRAP    C#RFLA
      MOV    R0,OPFLAG
      021642 104421
      021644 010067 161666
3169 021650 005003      CLR    R3      ;LET R3 := #0 ;CLEAR EVENT FLAG
3170 021652 105767 161651      TSTB    PWRFLG ;IF POWER FAIL HAS NOT OCCURRED THEN:
3171 021656 001020      BNE    503441
3172 021660      READEF #EF.NEW ;UPDATE PASS COUNT WHEN
      MOV    #EF.NEW,R0
      TRAP    C#REFG
      021660 012700 000035
      021664 104447
3173 021666 103014      BCC    503451 ;SUPERVISOR IS IN NEW PASS
3174 021670 105767 161632      TSTB    STAF LG ;AND DIAG WAS NEITHER STARTED
3175 021674 001010      BNE    503461
3176 021676      READEF #EF.RES ;NOR
      MOV    #EF.RES,R0
      TRAP    C#REFG
      021676 012700 000037
      021702 104447

```

```

3177 021704 103402      BCS      503474      ;IFCOND CC THEN      ;RESTARTED
3178 021706 005103      COM      R3          ;LET R3 := COMP R3 ;DO IT
3179
3180 021710 000401      BR        503504
3181 021712      503474:      INC      R3          ;SET 1ST PASS IF NEW PASS AND
3182 021712 005203      ;RESTARTING
3183
3184 021714      503504:
3185
3186 021714 000401      BR        503514
3187 021716      503464:      INC      R3          ;SET 1ST PASS IF NEW PASS AND
3188 021716 005203      ;STARTING
3189
3190 021720      503514:      ;DO NOT UPDATE IT ON CONTINUE
3191
3192 021720      503454:      ;OR ON POWER FAIL
3193
3194 021720      503444:      JSR      PC,FIRSTU      ;INIT DEVICE POINTER.
3195 021720 004767 175156      CLR      R2          ;LET R2 := #0      ;INIT DEVICE COUNTER.
3196 021724 005002
3197 021726      503524:      ;WHILE DEVTBL(R5) NE #END DO
3198 021726 026527 002604 177777      CMP      DEVTBL(R5),#END
3199 021734 001456      BEQ      503534
3200 021736 005202      INC      R2          ;LET R2 := R2 + #1
3201 021740 010500      MOV      R5,R0      ;LET R0 := R5 SHIFT -1
3202 021742 006200      ASR      R0
3203 021744      GPHARD      R0,R0      ;GET HARDWARE P TABLE FROM SUPER.
3204 021744 104442      TRAP      C:GPHRD
3205 021746 103044      BCC      503544      ;IFCOND CS THEN
3206 021750 011065 002514      MOV      (R0),TSDB(R5)      ;SAVE TSDB ADDRESS.
3207 021754 012065 002524      MOV      (R0)+,TSSR(R5)      ;SAVE TSSR ADDRESS.
3208 021760 062765 000002 002524      ADD      #2,TSSR(R5)
3209 021766 012065 002534      MOV      (R0)+,TSVCT(R5)      ;SAVE INTERRUPT VECTOR ADDRESS.
3210 021772 011065 003532      MOV      (R0)+,TSUNT(R5)      ;SAVE NUMBER OF DRIVE
3211 021776 011067 161532      MOV      (R0),TSNP      ;SAVE FOR PRINT OUT'S
3212 022002      SETVEC      TSVCT(R5),TSSINT(R5),#INTPRI
3213 022002 012746 000340      MOV      #INTPRI,-(SP)
3214 022006 016546 002554      MOV      TSSINT(R5),-(SP)
3215 022012 016546 002534      MOV      TSVCT(R5),-(SP)
3216 022016 012746 000003      MOV      #3,-(SP)
3217 022022 104437      TRAP      C:SVEC
3218 022024 062706 000010      ADD      #10,SP
3219
3220 022030 005065 003472      CLR      INTFLG(R5)      ;SET UP INTERRUPT PROCESSING CONDITIONS.
3221 022034 005703      TST      R3          ;CLEAR INTERRUPT FLAGS.
3222 022036 001410      BEQ      503554      ;ACTUAL PASSCOUNT UPDATE PER R3
3223 022040 005703      TST      R3
3224 022042 002003      BGE      503564      ;IF R3 LT #0 THEN
3225 022044 005265 003326      INC      PASCNT(R5)      ;LET PASCNT(R5) := PASCNT(R5) + #1
3226
3227 022050 000403      BR        503574
3228 022052      503564:      MOV      #1,PASCNT(R5)      ;LET PASCNT(R5) := #1
3229 022052 012765 000001 003326
3230
3231 022060      503574:
3232
3233 022060      503554:

```

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

MISCELLANEOUS SECTIONS MACRO M1113 14 JUN-84 18:32
INITIALIZE SECTION

SEQ 0108

```

3261 022260      50365$:
3262 022260 005267 161152      INC      TIME1
3263 022264      50365$:
3264 022264 026727 161146 000025      CMP      TIME1,#25
3265 022272 003134      BGT      4$
3266 022274 012775 002340 002514      MOV      @GSCPK,@TSDB(R5)
3267 022302      DELAY      25
                                ;AND GET UNITS STATUS
                                ;WAIT
                                MOV      #25,(PC)+
                                .WORD    0
                                MOV      L$DLY,(PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE      . 4
                                DEC      -22(PC)
                                BNE      . -20
3268 022332      CLRVEC      #4
                                ;CLEAR VECTOR AT 4
                                MOV      #4,R0
                                TRAP      C$CVEC
3269 022340 105767 161164      TSTB      TRAPD4
3270 022344 001423      BEQ      2$
3271 022346 005265 003366      INC      FTLCNT(R5)
3272 022352      PRINTF      @NODEV,TSSR(R5)
                                ;IFB TRAPD4 NE #0 THEN
                                ;LET FTLCNT(R5) := FTLCNT(R5) + #1
                                ;PRINT ERROR
                                MOV      TSSR(R5),-(SP)
                                MOV      @NODEV, -(SP)
                                MOV      #2, -(SP)
                                MOV      SP,R0
                                TRAP      C$PNTF
                                ADD      #6,SP
3273 022376 016567 002604 175004      MOV      DEVTBL(R5),DROPN
3274 022404 010500      MOV      R5,R0
3275 022406 006200      ASR      R0
3276 022410      DODU      R0
                                ;SAVE # OF UNIT TO BE DROPPED.
                                ;R0=LOGICAL DEVICE NUMBER
                                ;DROP THE UNIT
                                TRAP      C$DODU
3277 022412      DOCLN
                                ; EXEC BGNDU-ENDDU CODE IF IDU = 0
                                ;DO CLEAN EABORT
                                TRAP      C$DCLN
3279 022414 105067 161110      2$:
3280 022420      CLRB      TRAPD4
3281 022420      SETVEC      #4,@TRAP4,@INTPRI
                                ;CLEAR TRAP FLAG
                                ;SET VECTOR 4,PRIORITY #6
                                MOV      @INTPRI, -(SP)
                                MOV      @TRAP4, -(SP)
                                MOV      #4, -(SP)
                                MOV      #3, -(SP)
                                TRAP      C$SVEC
                                ADD      #10,SP
3282 022446 005775 002524      TST      @TSSR(R5)
3283 022452      DELAY      25
                                ;CHECK FOR ADDRESS
                                ;WAIT
                                MOV      #25,(PC)+
                                .WORD    0
                                MOV      L$DLY,(PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE      . -4
                                DEC      -22(PC)
                                BNE      . 20
3284 022502      CLRVEC      #4
                                ;CLEAR VECTOR AT 4
                                MOV      #4,R0

```

```

;IFB TRAPD4 NE #0 THEN          TRAP      C$CVEC
;LET FTLCNT(R5) := FTLCNT(R5) + #1
;PRINT ERROR
                                MOV      TSSR(R5), -(SP)
                                MOV      #NODEV, (SP)
                                MOV      #2, -(SP)
                                MOV      SP, R0
                                TRAP      C$PNTF
                                ADD      #6, SP
;SAVE # OF UNIT TO BE DROPPED.
;R0=LOGICAL DEVICE NUMBER

;DROP THE UNIT
                                TRAP      C$DODU
; EXEC BGNDU-ENDDU CODE IF IDU = 0
;DO CLEAN EABORT
                                TRAP      C$DCLN

;SET UNIT NUMBER

                                MOV      #25, (PC)+
                                .WORD    0
                                MOV      L$DLY, (PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE      .-4
                                DEC      -22(PC)
                                BNE      .-20
;AND GET UNITS STATUS
;WAIT
                                MOV      #25, (PC)+
                                .WORD    0
                                MOV      L$DLY, (PC)+
                                .WORD    0
                                DEC      -6(PC)
                                BNE      .-4
                                DEC      -22(PC)
                                BNE      .-20
;IF #TS.SSR SETIN @TSSR(R5) THEN
;IF #TS.OFL NOTSETIN @TSSR(R5) THEN
;EXIT COUNTER WHEN UNIT ON LINE

;PRINT UNIT OFF LINE EVERY 10 SEC
                                MOV      TSNP, -(SP)
                                MOV      #OFLIM, (SP)
                                MOV      #2, -(SP)
                                MOV      SP, R0
                                TRAP      C$PNTF
                                ADD      #6, SP

```

MISCELLANEOUS SECTIONS MACRO M1113 14-JUN-84 18:32
INITIALIZE SECTION

SEQ 0110

```

3311
3312 022732      50372$:
3313
3314 022732      000412      BR      50373$
3315 022734      50370$:
3316 022734      PRINTF #NRDYM,DEVTBL(R5)
      022734      016546      002604      MOV      DEVTBL(R5),-(SP)
      022740      012746      023700      MOV      #NRDYM, -(SP)
      022744      012746      000002      MOV      #2, -(SP)
      022750      01060C      MOV      SP,R0
      022752      104417      TRAP     C$PNTF
      022754      062706      000006      ADD      #6, SP

3317
3318 022760
3319 022760      012767      000001      160452      50373$:      MOV      #1,TIME2      ;INCR TIME2 FROM #1 TO #13 BY #1
3320 022766      000402      BR      50374$
3321 022770      50375$:
3322 022770      005267      160444      INC      TIME2
3323 022774      50374$:
3324 022774      026727      160440      000013      CMP      TIME2,#13
3325 023002      003C16      BGT      50376$
3326 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      L$DLY,(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

3327 023034      BREAK      ;ALLOW TERMINAL INTERRUPT
      023034      104422      TRAP     C$BRK
3328 023036      000754      BR      50375$
3329 023040      50376$:
3330 023040      000167      177214      JMP      50366$
3331 023044      50367$:
3332 023044      50364$:
3333 023044      CLRVEC      #4      ;CLEAR VECTOR AT 4
      023044      012700      000004      MOV      #4,R0
      023050      104436      TRAP     C$CVEC

3334 023052      026727      160360      000025      CMP      TIME1,#25
3335 023060      003404      BLE      50377$      ;IF OFF LINE FOR 3.5 MINUTES
3336 023062      004767      167646      JSR PC,MOVMSG      ;GET MESSAGE PACKET
3337 023066      004767      170360      JSR PC,TCC1      ;PRINT ERROR AND DROP OFF LINE UNIT
3338
3339 023072      50377$:
3340
3341 023072      004767      174052      JSR PC,NEXTU      ;REPEAT UNTIL ON LINE OR TIMED OUT.
3342
3343 023076      000167      177100      JMP      50362$      ;SET UP FOR NEXT UNIT.
3344
3345 023102      50363$:
3346 023102      50361$:
3347 023102      105767      160421      TSTB      PWRFLG      ;IFB PWRFLG EQ #0 THEN
3348 023106      001026      BNE      50400$
3349 023110      MEMORY DATAWT      ;REQUEST MEMORY FROM SUPER FOR RD/WR BUFFERS.
      023110      104431      TRAP     C$MEM

```

MISCELLANEOUS SECTIONS MACRO M1113 14-JUN-84 18:32
INITIALIZE SECTION

SEQ 0111

```

3350 023112 010067 160270
3351 023116 016767 160264 160264
3352 023124 062767 004000 160256
3353 023132 027727 160250 004000
3354 023140 002011
3355 023142
3356 023142 012746 023210
3357 023146 012746 000001
3358 023152 010600
3359 023154 104417
3360 023156 062706 000004
3361 023162
3362 023162 104444
3363 023164
3364 023164
3365 023164
3366 023164 105067 157024
3367 023170 012703 003526
3368 023174 004767 167464
3369 023200 105067 160323
3370 023204
3371 023204 104432
3372 023206 000104
3373 023210 045 101 106
3374 023213 122 105 105
3375 023216 040 115 105
3376 023221 115 117 040
3377 023224 124 117 117
3378 023227 040 123 115
3379 023232 101 114 114
3380 023235 040 106 117
3381 023240 122 040 122
3382 023243 104 055 127
3383 023246 122 040 102
3384 023251 106 122 123
3385 023254 045 116
3386 023256 045 101 122
3387 023261 105 055 114
3388 023264 117 101 104
3389 023267 040 111 116
3390 023272 040 114 101
3391 023275 122 107 105
3392 023300 122 040 115
3393 023303 105 115 117
3394 023306 045 116 000
3395 023312
3396 023312 104411
3397 023312
3398 023312
3399 023312

```

MOV DATAWT,DATARD ;SET RD BFR ADDRESS

ADD #DATCNT,DATARD

CMP #DATAWT,#DATCNT ;WHEN NOT ENOUGH FREE MEMO AVAILABLE

BGE 50401\$

PRINTF #MEMOM ;WARN OPERATOR

MOV #MEMOM,-(SP)

MOV #1,(SP)

MOV SP,R0

TRAP C\$PNTF

ADD #4,SP

DOCLN ;AND ABORT PASS

TRAP C\$DCLN

;DIAG MUST BE RE-LOADED IN A CPU WITH LARGER MEMO

50401\$:

50400\$:

CLRB CHGFLG ;CLR CHANGE CMD SEQ TBL FLAG.

MOV #ENDFLG,R3 ;LET R3 := #ENDFLG

JSR PC,CLRERR ;CLEAR ALL FLAGS.

CLRB PWRFLG ;CLEAR THE POWER FAIL FLAG.

EXIT INIT

TRAP C\$EXIT

.WORD L10012-

MEMOM: .ASCII /#AFREE MEMO TOO SMALL FOR RD-WR BFRS#N/

.ASCIZ /#ARE-LOAD IN LARGER MEMO#N/

.EVEN

ENDINIT

L10012:

TRAP C\$INIT

.SBTTL AUTO DROP SECTION

;

MISCELLANEOUS SECTIONS MACRO M1113 14 JUN 84 18:32
 AUTO DROP SECTION

SEQ 0112

```

3376 ;SECTION EXECUTED AFTER THE INIT CODE WHEN "ADR" FLAG IS SET BY OPERATOR
3377 ;SECTION CHECKS FOR A VALID INTERFACE LOCATION. DROPS UNIT IF NO RESPONSE
3378 ;FROM INTERFACE
3379 ;
3380
3381 023314 BGNAUTO
023314 L$AUTO::
3382
3383 023314 004767 173562 JSR PC,FIRSTU ;FIND FIRST UNIT
3384 023320 50402$: ;WHILE DEVTBL(R5) NE #END DO ;
3385 023320 026527 002604 177777 CMP DEVTBL(R5),#END
3386 023326 001525 BEQ 50403$
3387 023330 105067 160174 CLRB TRAPD4 ;LET TRAPD4 :B= #0 ;
3388 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$SVEC
023356 062706 000010 ADD #10,SP
3389 023362 017502 002514 MOV #TSD8(R5),R2 ;ADDRESS TS05 INTERFACE
3390 023366 CLRVEC #4 ;CLEAR VECTOR AT 4
023366 012700 000004 MOV #4,R0
023372 104436 TRAP C$CVEC
3391 023374 105767 160130 TSTB TRAPD4 ;IFB TRAPD4 NE #0 THEN
3392 023400 001423 BEQ 50404$
3393 023402 005265 003366 INC FTLCNT(R5) ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3394 023406 PRINTF #AUTODM,TSD8(R5) ;PRINT ERROR
023406 016546 002514 MOV TSD8(R5),-(SP)
023412 012746 023604 MOV #AUTODM,-(SP)
023416 012746 000002 MOV #2,-(SP)
023422 010600 MOV SP,R0
023424 104417 TRAP C$PNTF
023426 062706 000006 ADD #6,SP
3395 023432 016567 002604 173750 MOV DEVTBL(R5),DROPN ;SAVE # OF UNIT TO BE DROPPED.
3396 023440 010500 MOV R5,R0 ;R0=LOGICAL DEVICE NUMBER
3397 023442 006200 ASR R0
3398 023444 DODU R0 ;DROP THE UNIT: EXEC BGNDU-ENDDU CODE IF IDU = 0
023444 104451 TRAP C$DODU
3399
3400 023446 000452 BR 50405$
3401 023450 50404$:
3402 023450 012775 002340 002514 MOV #GSCPK,#TSD8(R5) ;SEND GET STATUS COMMAND
3403 023456 004767 167216 JSR PC,WSSR ;WAIT
3404 023462 032775 000200 002524 BIT #TS.SSR,#TSSR(R5) ;IF #TS.SSR SETIN #TSSR(R5) THEN
3405 023470 001423 BEQ 50406$
3406 023472 032775 000100 002524 BIT #TS.OFL,#TSSR(R5) ;IF #TS.OFL SETIN #TSSR(R5) THEN
3407 023500 001416 BEQ 50407$
3408 023502 005265 003366 INC FTLCNT(R5) ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3409 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
3410 023532 004767 173566 JSR PC,DROPUA

```

MISCELLANEOUS SECTIONS MACRO M1113 14-JUN-84 18:32
 AUTO DROP SECTION

SEQ 0113

```

3411
3412 023536          50407$:
3413
3414 023536 000416          BR      50410$
3415 023540          50406$:
3416 023540 005265 003366          INC      FTLCNT(R5)          ;LET FTLCNT(R5) := FTLCNT(R5) + #1
3417 023544          PRINTF #NRDYM,DEVTBL(R5)
      023544 016546 002604          MOV      DEVTBL(R5),-(SP)
      023550 012746 023700          MOV      #NRDYM,(SP)
      023554 012746 000002          MOV      #2,-(SP)
      023560 010600          MOV      SP,R0
      023562 104417          TRAP     C$PNTF
      023564 062706 000006          ADD      #6,SP
3418 023570 004767 173530          JSR PC,DROP UA
3419
3420 023574          50410$:
3421
3422 023574          50405$:
3423 023574 004767 173350          JSR PC,NEXTU
3424
3425 023600 000647          BR      50402$
3426 023602          50403$:
3427
3428 023602          ENDAUTO
      023602          L10013:
      023602 104461          TRAP     C$AUTO
3429
3430 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
3431 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
3432 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
3433          .EVEN

```

MISCELLANEOUS SECTIONS MACRO M1113 14 JUN 84 18:32
 AUTO DROP SECTION

SEQ 0114

```

3434
3435      ;      DEVICE BUS TRAP HANDLER
3436      ;      OUTPUT: TRAPD4 BYTE 1: TRAPED AT 4
3437      ;      0: NO TRAP
3438
3439 023730 105267 157574      TRAP4:: INCB      TRAPD4;LET TRAPD4 :B= TRAPD4 + #1
3440 023734 000002      RTI
3441
3442
3443      .SBTTL  CLEANUP CODING SECTION
3444
3445      ;+
3446      ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
3447      ; AT THE END OF EACH PASS.
3448      ;
3449
3450 023736      BGNCLN
3451 023736      L$CLEAN::
3452 023736 004767 173140      JSR      PC,FIRSTU      ;FIND FIRST UNIT.
3453 023742      50411$: ;WHILE DEVTBL(R5) NE #END DO
3454 023742 026527 002604 177777      CMP      DEVTBL(R5),#END
3455 023750 001410      BEQ      50412$
3456 023752 004767 166722      JSR PC,WSSR      ;WAIT FOR UNIT READY OR TIMEOUT,
3457 023756      CLRVEC      TSVCT(R5)      ;RELEASE INTERRUPT VECTORS FOR ALL DEV.
3458 023756 016500 002534      MOV      TSVCT(R5),R0
3459 023762 104436      TRAP      C$CVEC
3460 023764 004767 173160      JSR      PC,NEXTU      ;FIND NEXT UNIT.
3461 023770 000764      BR      50411$
3462 023772      50412$:
3463 023772      EXIT      CLN
3464 023772 104432      TRAP      C$EXIT
3465 023774 000002      .WORD      L10014-.
3466 023776      .EVEN
3467 023776      ENDCLN
3468 023776      L10014:
3469 023776 104412      TRAP      C$CLEAN
3470
3471      .SBTTL  DROP UNIT SECTION
3472
3473      ;+
3474      ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3475      ; TO NO LONGER BE TESTED. THAT CODE SHALL BE EXECUTED WHEN DODU
3476      ; MACRO IS CALLED WHILE IDU FL 3 IS NOT SET BY OPERATOR
3477      ;--
3478 024000      BGN DU
3479 024000      L$DU::
3480 024000 010005      MOV      R0,R5      ;R5 = LOGICAL DEVICE NUMBER X 2.
3481 024002 006305      ASL      R5
3482 024004 012765 177774 002604      MOV      #NINUSE,DEVTBL(R5)      ;SET NOT IN USE FLAG FOR THE DEVICE.
3483 024012 016500 002534      CLRVEC TSVCT(R5)      ;RELEASE THE INTERRUPT VECTOR.
3484 024012 016500 002534      MOV      TSVCT(R5),R0

```

MISCELLANEOUS SECTIONS MACRO M1113 14 JUN-84 18:32
DROP UNIT SECTION

SEQ 0115

```

3482 024016 104436
024020          PRINTF #DROPDM,DROPN          ;PRINT DROP DEVICE MESSAGE
024020 016746 173364
024024 012746 005065
024030 012746 000002
024034 010600
024036 104417
024040 062706 000006
3483 024044          EXIT    DU
024044 000167
024046 000000
3484          .EVEN
3485
3486 024050          ENDDU
024050          L10015:
024050 104453          TRAP    C$DU

3487          .SBTTL  ADD UNIT SECTION
3488
3489
3490          ;**
3491          ; THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3492          ; TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
3493          ; "EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
3494          ;--
3495
3496 024052          BGNAU
024052          L$AU::
3497 024052 010005          MOV    R0,R5          ;R5 = LOGICAL DEVICE NUMBER X 2.
3498 024054 006305          ASL    R5
3499 024056 010065 002604          MOV    R0,DEVTBL(R5)          ;STORE UNIT # IN DEVICE TABLE.
3500 024062          GPHARD R0,R0          ;GET HARDWARE P TABLE FROM SUPER.
024062 104442          TRAP    C$GPHRD
3501 024064 011065 002514          MOV    (R0),TSDB(R5)          ;SAVE TSDB ADDRESS.
3502 024070 012065 002524          MOV    (R0)+,TSSR(R5)          ;SAVE TSSR ADDRESS.
3503 024074 062765 000002 002524          ADD    #2,TSSR(R5)
3504 024102 011065 002534          MOV    (R0),TSVCT(R5)          ;SAVE INTERRUPT VECTOR ADDRESS.
3505 024106 011065 003532          MOV    (R0),TSUNT(R5)          ;SAVE NUMBER OF DRIVE
3506 024112 011067 157416          MOV    (R0),TSNP          ;SAVE FOR PRINT OUT'S
3507 024116          SETVEC TSVCT(R5),TSSINT(R5),#INTPRI
024116 012746 000340          MOV    #INTPRI,-(SP)
024122 016546 002554          MOV    TSSINT(R5),-(SP)
024126 016546 002534          MOV    TSVCT(R5),-(SP)
024132 012746 000003          MOV    #3,-(SP)
024136 104437          TRAP    C$SVEC
024140 062706 000010          ADD    #10,SP
3508          ;SET UP INTERRUPT PROCESSING CONDITIONS.
3509 024144 005065 003472          CLR    INTFLG(R5)          ;CLEAR INTERRUPT FLAGS.
3510
3511 024150          EXIT    AU
024150 000167
024152 000000
3512          .EVEN
3513
3514
3515 024154          ENDAU
024154          L10016:
024154 104452          TRAP    C$AU

```

```

3516
3517
3518
3519      .TITLE HARDWARE TESTS
3520
3521      .SBTTL TEST 1: BASIC FUNCTIONS.
3522
3523      ;**
3524      ; TEST TO EXECUTE ALL TS05 FUNCTIONS.
3525      ;--
3526
3527 024156      BGNMOD
3528
3529 024156      BGNTST
3530 024156
3531 024156 105067 157333      CLRB      RANDOM      ;CLR THE RANDOM OPERATIONS FLAG.
3532 024162 105067 157326      CLRB      EXPBOT      ;CLR EXPECT BOT FLAG.
3533
3534 024166      BGNSUB      ;SUBTEST 1 - SET CHAR, DRIVE INIT, GET STATUS.
3535 024166 104402      T1.1:      TRAP      C#BSUB
3536 024170 004767 172706      JSR      PC,FIRSTU      ;FIND THE FIRST UNIT.
3537 024174 004767 162672      JSR      PC,SOFINIT      ;INIT DEVICE
3538 024200 103404      BCS      11$
3539 024202      ERRDF      2,NSSRM,STAERM      ;REPORT TS05 NOT READY
3540 024202 104455      TRAP      C#ERDF
3541 024204 000002      .WORD      2
3542 024206 004536      .WORD      NSSRM
3543 024210 006120      .WORD      STAERM
3544
3545 024212 004767 163250      11$: JSR      PC,MDSET      ;GO DO SETUP'S
3546 024216 012702 025074      MOV      #BFSEQ0,R2      ;ADR OF CMD SEQ.
3547 024222 004767 000622      JSR      PC,BFSEQ      ;SET UP CMD SEQ.
3548 024226 004767 163774      JSR      PC,EXALL      ;EXECUTE CMD SEQ ON ALL DEVICES.
3549 024232 004767 172644      JSR      PC,FIRSTU      ;FIND THE FIRST UNIT.
3550 024236      50413$: ;WHILE DEVTBL(R5) NE #END DO      ;WHILE THERE ARE MORE DEVICES:
3551 024236 026527 002604 177777      CMP      DEVTBL(R5),#END
3552 024244 001451      BEQ      50414$
3553 024246 016502 002544      MOV      MSGPKA(R5),R2      ;GET MSG PACKET ADR,
3554 024252 062702 000012      ADD      #12,R2      ;LET R2 := R2 + #12      ;GET XSTAT2 ADR,
3555 024256 011265 002564      MOV      (R2),TS5CL(R5)      ;STORE CODE LEVEL FROM DTR BYTE,
3556 024262 042765 177700 002564      BIC      #177700,TS5CL(R5)
3557 024270 011265 002574      MOV      (R2),TS5SW(R5)      ;STORE SWITCH SETTINGS
3558 024274 042765 177477 002574      BIC      #177477,TS5SW(R5)
3559 024302      PRINTF      #CODELM,DEVTBL(R5),TS5CL(R5)
3560 024302 016546 002564      MOV      TS5CL(R5),-(SP)
3561 024306 016546 002604      MOV      DEVTBL(R5),-(SP)
3562 024312 012746 004162      MOV      #CODELM, -(SP)
3563 024316 012746 000003      MOV      #3, -(SP)
3564 024322 010600      MOV      SP,R0
3565 024324 104417      TRAP      C#PNTF
3566 024326 062706 000010      ADD      #10,SP
3567
3568      ;PRINT THE TS05 MICROCODE LEVEL.
3569 024332      PRINTF      #SWSET,DEVTBL(R5),TS5SW(R5)
3570 024332 016546 002574      MOV      TS5SW(R5),-(SP)

```

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0117

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
3558						
3559	024362		504154:			;PRINT THE TS05 SWITCH SETTINGS.
3560	024362	004767	172562	JSR	PC, NEXTU	;FIND NEXT UNIT.
3561						
3562	024366	000723		BR	504134	
3563	024370		504144:			
3564						
3565	024370			ENDSUB		
	024370		L10020:			
	024370	104403				TRAP C#ESUB
3566						
3567	024372			BGNSUB		;SUBTEST 2 REWIND.
	024372		T1.2:			
	024372	104402				TRAP C#BSUB
3568						
3569	024374	012702	025146	MOV	#BFSEQ1, R2	;ADR OF CMD SEQ.
3570	024400	004767	000444	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3571	024404	004767	163616	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3572	024410	105067	157112	CLRB	STAFLG	;CLEAR START FLAG
3573	024414			ENDSUB		
	024414		L10021:			
	024414	104403				TRAP C#ESUB
3574						
3575	024416			BGNSUB		SUBTEST 3 - WRITE/VERIFY.
	024416		T1.3:			
	024416	104402				TRAP C#BSUB
3576						
3577	024420	012702	025160	MOV	#BFSEQ2, R2	;ADR OF CMD SEQ.
3578	024424	004767	000420	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3579	024430	004767	163572	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3580	024434			ENDSUB		
	024434		L10022:			
	024434	104403				TRAP C#ESUB
3581						
3582	024436			BGNSUB		;SUBTEST 4 - WRITE TAPE MARK, ERASE.
	024436		T1.4:			
	024436	104402				TRAP C#BSUB
3583						
3584	024440	012702	025252	MOV	#BFSEQ3, R2	;ADR OF CMD SEQ.
3585	024444	004767	000400	JSR	PC, BFSEQ	;SET UP CMD SEQ.
3586	024450	004767	163552	JSR	PC, EXALL	;EXECUTE CMD SEQ ON ALL DEVICES.
3587	024454			ENDSUB		
	024454		L10023:			
	024454	104403				TRAP C#ESUB
3588						
3589	024456			BGNSUB		;SUBTEST 5 - SPACE FILES.
	024456		T1.5:			
	024456	104402				TRAP C#BSUB
3590						
3591	024460	012702	025324	MOV	#BFSEQ4, R2	;ADR OF CMD SEQ.
3592	024464	004767	000360	JSR	PC, BFSEQ	;SET UP CMD SEQ.

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0118

3593	024470	004767	163532		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3594	024474			L10024:	ENDSUB			
	024474							
	024474	104403						TRAP C1ESUB
3595								
3596	024476			T1.6:	BGNSUB			;SUBTEST 6 SPACE RECORDS.
	024476							
	024476	104402						TRAP C1BSUB
3597								
3598	024500	012702	025366		MOV	#BFSEQ5,R2		;ADR OF CMD SEQ.
3599	024504	004767	000340		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3600	024510	004767	163512		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3601	024514				ENDSUB			
	024514			L10025:				
	024514	104403						TRAP C1ESUB
3602								
3603	024516			T1.7:	BGNSUB			;SUBTEST 7 - WRITE RETRY.
	024516							
	024516	104402						TRAP C1BSUB
3604								
3605	024520	012702	025440		MOV	#BFSEQ6,R2		;ADR OF CMD SEQ.
3606	024524	004767	000320		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3607	024530	004767	163472		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3608	024534				ENDSUB			
	024534			L10026:				
	024534	104403						TRAP C1ESUB
3609								
3610	024536			T1.8:	BGNSUB			;SUBTEST 8 - READ REV RETRY.
	024536							
	024536	104402						TRAP C1BSUB
3611								
3612	024540	012702	025512		MOV	#BFSEQ7,R2		;ADR OF CMD SEQ.
3613	024544	004767	000300		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3614	024550	004767	163452		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3615	024554				ENDSUB			
	024554			L10027:				
	024554	104403						TRAP C1ESUB
3616								
3617	024556			T1.9:	BGNSUB			;SUBTEST 9 - READ FWD RETRY.
	024556							
	024556	104402						TRAP C1BSUB
3618								
3619	024560	012702	025544		MOV	#BFSEQ8,R2		;ADR OF CMD SEQ.
3620	024564	004767	000260		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3621	024570	004767	163432		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3622	024574				ENDSUB			
	024574			L10030:				
	024574	104403						TRAP C1ESUB
3623								
3624	024576			T1.10:	BGNSUB			;SUBTEST 10- CLEAN.
	024576							
	024576	104402						TRAP C1BSUB
3625								
3626	024600	012702	025576		MOV	#BFSEQ9,R2		;ADR OF CMD SEQ.
3627	024604	004767	000240		JSR	PC,BFSEQ		;SET UP CMD SEQ.
3628	024610	004767	163412		JSR	PC,EXALL		;EXECUTE CMD SEQ ON ALL DEVICES.
3629	024614				ENDSUB			

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0119

024614				L10031:			
(24614	104403					TRAP	C#ESUB
3630							
3631	024616			T1.11:	BGNSUB	;SUBTEST 11	WTV SWAPPED DATA BYTES.
	024616						
	024616	104402				TRAP	C#BSUB
3632	024620	012702	025620		MOV	#BFSE10,R2	;ADR OF CMD SEQ.
3633	024624	004767	000220		JSR	PC,BFSEQ	;SET UP CMD SEQ.
3634	024630	004767	163372		JSR	PC,EXALL	;WRITE/VERIFY RECORDS 1 AND 2.
3635	024634	112767	000001	156656	MOVB	#1,SMBFLG	;ENABLE BYTE SWAPPING.
3636	024642	004767	163360		JSR	PC,EXALL	;WRITE/VERIFY RECORDS 3 AND 4.
3637	024646	105067	156646		CLRB	SMBFLG	;DISABLE BYTE SWAPPING.
3638	024652				ENDSUB		
	024652			L10032:			
	024652	104403				TRAP	C#ESUB
3639	024654	016702	156526		MOV	DATAWT,R2	;INIT WRITE BUFFER POINTER.
3640	024660	062702	000012		ADD	#10.,R2	
3641	024664			504161:	;WHILE R2 NE DATAWT DO	;UNTIL 10 BYTES HAVE BEEN SWAPPED.	
3642	024664	020267	156516		CMP	R2,DATAWT	
3643	024670	001402			BEQ	504171	
3644	024672	000342			SWAB	-(R2)	;SWAP DATA BYTES IN WRITE BUFFER.
3645							
3646	024674	000773			BR	504161	
3647	024676			504171:			
3648	024676	105267	156621		INCB	T1SWB	;SET T1 SWAP BYTES FLAG FOR "CKDATA" SUBR
3649							
3650	024702				BGNSUB	;SUBTEST 12 - READ SWAPPED DATA BYTES.	
	024702			T1.12:			
	024702	104402				TRAP	C#BSUB
3651	024704	012767	104401	156506	MOV	#RDR,CMDWRD	;CMD IS READ REV.
3652	024712	004767	171234		JSR	PC,VFEXC	;VERIFY ODD LENGTH SWAP (RECORD 4).
3653	024716	012767	000012	155412	MOV	#12,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 10.
3654	024724	004767	171222		JSR	PC,VFEXC	;VERIFY EVEN LENGTH SWAP (RECORD 3).
3655	024730	112767	000001	156562	MOVB	#1,SMBFLG	;ENABLE BYTE SWAPPING.
3656	024736	012767	000011	155372	MOV	#11,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 9.
3657	024744	004767	171202		JSR	PC,VFEXC	;VERIFY ODD LENGTH SWAP (RECORD 2).
3658	024750	012767	000012	155360	MOV	#12,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 10.
3659	024756	004767	171170		JSR	PC,VFEXC	;VERIFY EVEN LENGTH SWAP (RECORD 1).
3660	024762	012767	104001	156430	MOV	#RDF,CMDWRD	;CMD IS READ FWD.
3661	024770	004767	171156		JSR	PC,VFEXC	;VERIFY EVEN LENGTH SWAP (RECORD 1).
3662	024774	012767	000011	155334	MOV	#11,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 9.
3663	025002	004767	171144		JSR	PC,VFEXC	;VERIFY ODD LENGTH SWAP (RECORD 2).
3664	025006	105067	156506		CLRB	SMBFLG	;DISABLE BYTE SWAPPING.
3665	025012	012767	000012	155316	MOV	#12,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 10.
3666	025020	004767	171126		JSR	PC,VFEXC	;VERIFY EVEN LENGTH SWAP (RECORD 3).
3667	025024	012767	000011	155304	MOV	#11,CMDPKT.CP.CNT	;CHANGE BYTE COUNT TO 9.
3668	025032	004767	171114		JSR	PC,VFEXC	;VERIFY ODD LENGTH SWAP (RECORD 4).
3669							
3670	025036				ENDSUB		
	025036			L10033:			
	025036	104403				TRAP	C#ESUB
3671							
3672	025040	105067	156457		CLRB	T1SWB	;CLEAR T1 SWAP BYTES FLAG
3673							
3674	025044				EXIT	TST	
	025044	104432				TRAP	C#EXIT
	025046	000574				.WORD	L10017

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0120

```

3675
3676      ;      SUBROUTINE TO MOVE A COMMAND SEQUENCE TO THE SEQUENCE TABLE.
3677      ;      INPUTS:      R2 = FWA OF COMMAND SEQUENCE.
3678      ;      OUTPUTS:
3679      ;      REGISTERS:
3680      ;      CALLS:
3681
3682 025050 012701 003540      BFSEQ:: MOV      #CMDSEQ,R1      ;INIT SEQ TABLE ADDRESS.
3683 025054      504201: ;WHILE (R2) NE #END DO      ;WHILE THERE ARE MORE COMMANDS:
3684 025054      CMP      (R2),#END
3685 025060      BEQ      504211
3686 025062      MOV      (R2)+,(R1)+      ;MOVE COMMANDS TO SEQ TABLE.
3687
3688 025064 000773      BR      504201
3689 025066      504211:
3690 025066 012711 177777      MOV      #END,(R1)      ;STORE END OF SEQUENCE CODE.
3691 025072 000207      RTS      PC      ;RETURN.
3692
3693
3694      ;      BASIC FUNCTION COMMAND SEQUENCE
3695
3696 025074 140004      BFSEQ0: .WORD      SCH      ;SET CHAR. 200.      (1)
3697 025076 000200      200
3698 025100 000001      1
3699 025102 000000      0
3700 025104 100013      DRI      ;DRIVE INIT.      (2)
3701 025106 000001      1
3702 025110 000001      1
3703 025112 000000      0
3704 025114 140004      SCH      ;SET CHAR. 20      (3)
3705 025116 000020      20
3706 025120 000001      1
3707 025122 000000      0
3708 025124 100017      GES      ;GET STATUS.      (4)
3709 025126 000001      1
3710 025130 000001      1
3711 025132 000000      0
3712 025134 140004      SCH      ;SET CHAR. 40.      (5)
3713 025136 000040      40
3714 025140 000001      1
3715 025142 000000      0
3716 025144 177777      .WORD      END
3717
3718 025146 102010      BFSEQ1:      RWD      ;REWIND TWICE.      (6)
3719 025150 000001      1
3720 025152 000002      2
3721 025154 000000      0
3722 025156 177777      .WORD      END
3723
3724 025160 104105      BFSEQ2:      WTV      ;WRITE/VERIFY PAT 1.      (7)
3725 025162 004000      DATCNT
3726 025164 000001      1
3727 025166 000001      1
3728 025170 104105      WTV      ;WTV PAT 2.      (8)
3729 025172 004000      DATCNT
3730 025174 000001      1
3731 025176 000002      2

```

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0121

3732	025200	104105	WTV	;WTV PAT 3.	(9)
3733	025202	004000	DATCNT		
3734	025204	000001	1		
3735	025206	000003	3		
3736	025210	104105	WTV	;WTV PAT 4.	(10)
3737	025212	004000	DATCNT		
3738	025214	000001	1		
3739	025216	000004	4		
3740	025220	104105	WTV	;WTV PAT 5.	(11)
3741	025222	004000	DATCNT		
3742	025224	000001	1		
3743	025226	000005	5		
3744	025230	104105	WTV	;WTV PAT 6.	(12)
3745	025232	004000	DATCNT		
3746	025234	000001	1		
3747	025236	000006	6		
3748	025240	104105	WTV	;WTV PAT 0.	(13)
3749	025242	004000	DATCNT		
3750	025244	000001	1		
3751	025246	000000	0		
3752	025250	177777	.WORD	END	
3753					
3754	025252	100011	BFSEQ3: WTM	;WRITE TAPE MARK.	(14)
3755	025254	000001	1		
3756	025256	000001	1		
3757	025260	000000	0		
3758	025262	104005	WRT	;WRITE 10 RECORDS.	(15)
3759	025264	004000	DATCNT		
3760	025266	000010	10		
3761	025270	000001	1		
3762	025272	100411	ERS	;ERASE 10 TIMES.	(16)
3763	025274	000001	1		
3764	025276	000010	10		
3765	025300	000000	0		
3766	025302	100011	WTM	;WRITE TAPE MARK.	(17)
3767	025304	000001	1		
3768	025306	000001	1		
3769	025310	000000	0		
3770	025312	101011	WTR	;WTM RETRY	(18)
3771	025314	000001	1		
3772	025316	000001	1		
3773	025320	000000	0		
3774	025322	177777	.WORD	END	
3775					
3776	025324	105410	BFSEQ4: SFR	;SPACE 2 FILES REV.	(19)
3777	025326	000002	2		
3778	025330	000001	1		
3779	025332	000000	0		
3780	025334	105010	SFF	;SPACE 2 FILES FWD.	(20)
3781	025336	000002	2		
3782	025340	000001	1		
3783	025342	000000	0		
3784	025344	105410	SFR	;SPACE 2 FILES REV.	(21)
3785	025346	000001	1		
3786	025350	000002	2		
3787	025352	000000	0		
3788	025354	105010	SFF	;SPACE 2 FILES FWD.	(22)

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0122

3789	025356	000001	1		
3790	025360	000002	2		
3791	025362	000000	0		
3792	025364	177777	END		
3793					
3794	025366	102010	BFSEQ5:	RWD	;REWIND. (23)
3795	025370	000001	1		
3796	025372	000001	1		
3797	025374	000000	0		
3798	025376	104010	SRF		;SPACE 7 RECORDS FWD. (24)
3799	025400	000007	7		
3800	025402	000001	1		
3801	025404	000000	0		
3802	025406	104410	SRR		;SPACE 7 RECORDS REV. (25)
3803	025410	000007	7		
3804	025412	000001	1		
3805	025414	000000	0		
3806	025416	104010	SRF		;SPACE 7 RECORDS FWD. (26)
3807	025420	000001	1		
3808	025422	000007	7		
3809	025424	000000	0		
3810	025426	104410	SRR		;SPACE 7 RECORDS REV. (27)
3811	025430	000001	1		
3812	025432	000007	7		
3813	025434	000000	0		
3814	025436	177777	END		
3815					
3816	025440	102010	BFSEQ6:	RWD	;REWIND. (28)
3817	025442	000001	1		
3818	025444	000001	1		
3819	025446	000000	0		
3820	025450	104005	WRT		;WRITE. (29)
3821	025452	004000	DATCNT		
3822	025454	000001	1		
3823	025456	000001	1		
3824	025460	105005	WRR		;WRITE RETRY. (30)
3825	025462	004000	DATCNT		
3826	025464	000001	1		
3827	025466	000001	1		
3828	025470	100011	WTH		;WRITE TAPE MARK.
3829	025472	000001	1		
3830	025474	000001	1		
3831	025476	000000	0		
3832	025500	105410	SFR		;SPACE 1 FILE REV.
3833	025502	000001	1		
3834	025504	000001	1		
3835	025506	000000	0		
3836	025510	177777	END		
3837					
3838	025512	104401	BFSEQ7:	RDR	;READ REV. (31)
3839	025514	004000	DATCNT		
3840	025516	000001	1		
3841	025520	000001	1		
3842	025522	105401	RNR		;READ NEXT REV. (32)
3843	025524	004000	DATCNT		
3844	025526	000001	1		
3845	025530	000001	1		

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 1: BASIC FUNCTIONS.

SEQ 0123

```

3846 025532 125401          RNF          ;READ NEXT FWD.          (33)
3847 025534 004000          DATCNT
3848 025536 000001          1
3849 025540 000001          1
3850 025542 177777          .WORD      END
3851
3852 025544 104001          BFSEQ8:      RDF          ;READ FWD.          (34)
3853 025546 004000          DATCNT
3854 025550 000001          1
3855 025552 000001          1
3856 025554 105001          RPF          ;READ PREVIOUS FWD.      (35)
3857 025556 004000          DATCNT
3858 025560 000001          1
3859 025562 000001          1
3860 025564 125001          RPR          ;READ PREVIOUS REV.      (36)
3861 025566 004000          DATCNT
3862 025570 000001          1
3863 025572 000001          1
3864 025574 177777          .WORD      END
3865
3866 025576 101012          BFSEQ9: .WORD      CLN          ;CLEAN.          (37)
3867 025600 000001          1
3868 025602 000001          1
3869 025604 000000          0
3870 025606 102010          RWD          ;REWIND          (38)
3871 025610 000001          1
3872 025612 000001          1
3873 025614 000000          0
3874 025616 177777          .WORD      END          ;END OF SEQUENCE.
3875
3876 025620 104105          BFSE10:      WTV          ;WRITE/VERIFY EVEN LENGTH. (39)
3877 025622 000012          12
3878 025624 000001          1
3879 025626 000000          0
3880 025630 104105          WTV          ;WRITE/VERIFY ODD LENGTH.  (40)
3881 025632 000011          11
3882 025634 000001          1
3883 025636 000000          0
3884 025640 177777          .WORD      END
3885          .EVEN
3886
3887 025642          ENDTST
          L10017:
          025642
          025642 104401          TRAP      C$ETST
3888
3889          .SBTTL  TEST 2: DATA RELIABILITY.
3890
3891          ;++
3892          ; TEST TO CHECK THE DATA RELIABILITY OF THE TS05.
3893          ;--
3894 025644          BGNTST
          025644          T2::
3895
3896 025644 112767 000001 155643      MOVB      #1,RANDOM          ;SET THE RANDOM OPERATIONS FLAG.
3897 025652 105067 155636          CLR      EXPBOT          ;CLEAR EXPECT BOT FLAG.
3898 025656 005067 155574          CLR      WTMFLG          ;CLEAR WRITE TAPE MARK FLAG
3899 025662 004767 171214          JSR      PC,FIRSTU          ;FIND THE FIRST UNIT.

```

3900	025666	004767	161200		JSR	PC,SOFINIT		;INIT DEVICE	
3901	025672	103404			BCS	11:			
3902	025674				ERRDF	2,NSSRM,STAERM		;REPORT TS05 NOT READY	
	025674	104455							TRAP C:ERDF
	025676	000002							.WORD 2
	025700	004536							.WORD NSSRM
	025702	006120							.WORD STAERM
3903									
3904	025704	004767	161556	11:	JSR	PC,MDSET		;GO DO SETUP'S	
3905	025710	012702	004000		MOV	#DATCNT,R2		;SET UP THE RECORD LENGTH MASK.	
3906	025714	005302			DEC	R2			
3907	025716	010267	155506		MOV	R2,LENMSK		;ALLOW MAXIMUM BUFFER.	
3908	025722	005167	155502		COM	LENMSK			
3909	025726	004767	162230		JSR	PC,SETCH		;CMD 1 = SET CHARACTERISTIC.	
3910	025732	105767	155570		TSTB	STAFLG ;IFB STAFLG NE #0 THEN		;IF STARTING THEN:	
3911	025736	001417			BEQ	50424:			
3912	025740	004767	162242		JSR	PC,SETRW		;CMD2=REWIND	
3913	025744	105067	155556		CLRB	STAFLG ;LET STAFLG :B= #0		;CLR START FLAG.	
3914									
3915	025750			50422:					
3916	025750	012721	104105		MOV	#WTV,(R1)+			
3917	025754	012721	004000		MOV	#DATCNT,(R1)+			
3918	025760	012702	177740		MOV	#RNOPSC,R2			
3919	025764	005102			COM	R2			
3920	025766	010221			MOV	R2,(R1)+			
3921	025770	012721	000007		MOV	#RANP,(R1)+			
3922									
3923	025774			50423:	BREAK			; DO A SUPVSR BREAK FIRST.	
	025774	104422						TRAP C:BRK	
3924									
3925	025776			50424:				;FILL SEQ TBL WITH RANDOM CMDs.	
3926	025776	020127	003740		CMP	R1,#SEQEND			
3927	026002	002012			BGE	50425:			
3928	026004	066767	155422	155422	ADD	RANB,RANS		;LET RANS := RANS + RANB	
3929	026012	016702	155416		MOV	RANS,R2			
3930	026016	042702	177741		BIC	#177741,R2			
3931	026022	004772	026160		JSR	PC,#RANCMD(R2)		;SET UP A RANDOM CMD + BRK.	
3932									
3933	026026	000763			BR	50424:			
3934	026030			50425:					
3935	026030	012711	177777		MOV	#END,(R1)		;STORE END OF SEQUENCE CODE IN TABLE.	
3936	026034	004767	162166		JSR	PC,EXALL		;GO EXECUTE ALL CMDs IN SEQUENCE TABLE.	
3937									
3938	026040	012701	003540		MOV	#CMDSEQ,R1		;INIT CMD SEQ TBL POINTER.	
3939	026044	005702			TST	R2		;REPEAT UNTIL EOT IS REACHED	
3940	026046	001752			BEQ	50423:			
3941	026050	105267	155450		INCB	ALLEOT		;FLAG ALL UNITS @ EOT	
3942	026054	000240			NOP				
3943	026056	000240			NOP				
3944	026060	000240			NOP				
3945	026062	004767	001546		JSR	PC,TSWEOT		;WRITE ONE RECORD BEYOND EOT ON ALL UNITS	
3946								;SO THAT SHORTER READ STOP DISTANCE	
3947								;SHALL POSITION HEAD IN CLEAN IRG GAP	
3948								;READ REV THAT EXTRA REC TO RE-POSITION THE TAPE	
3949	026066	004767	000126		JSR	PC,RANRD		;SET UP READ REV/FWD CMDs.	
3950	026072	012767	177740	155444	MOV	#RNOPSC,CMDSEQ+4		;# OF RECORDS FOR READ REV.	
3951	026100	005167	155440		COM	CMDSEQ+4			

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 2: DATA RELIABILITY.

SEQ 0125

```

3952 026104 016767 155434 155442      MOV      CMDSEQ+4,CMDSEQ+14      ;# OF RECORDS FOR READ FORWARD.
3953 026112 012711 177777              MOV      #END,(R1)          ;STORE END OF SEQUENCE CODE IN SEQ TABLE.
3954 026116 004767 162104              JSR      PC,EXALL          ;GO EXECUTE READ REV/FWD OF LAST N RECORDS.
3955 026122 105067 155376              CLR      ALLEOT          ;CLEAR ALL UNITS @ EOT FLAG
3956 026126 112767 000001 155363      MOV      #1,RPTFLG        ;REQUEST PERFORMANCE REPORT DURING REWIND.
3957 026134 012701 003540              MOV      #CMDSEQ,R1       ;INIT SEQ TBL POINTER,
3958 026140 004767 162042              JSR      PC,SETRW         ;STORE REWIND IN SEQ TBL,
3959 026144 012711 177777              MOV      #END,(R1)       ;STORE END IN SEQ TBL,
3960 026150 004767 162052              JSR      PC,EXALL        ;EXECUTE REWIND CMD ON ALL UNITS
3961
3962 026154              EXIT      TST
      026154 104432
      026156 000320
                                     TRAP      C$EXIT
                                     .WORD     L10034-.

3963
3964      ;      ADDRESSES OF SUBROUTINES USED TO SET UP RANDOM OPERATIONS IN
3965      ;      THE DATA RELIABILITY TEST.
3966
3967 026160 026356      RANCMD: RANWV          ;WRITE
3968 026162 026332      RANWR          ;WRITE.
3969 026164 026332      RANWR          ;WRITE.
3970 026166 026332      RANWR          ;WRITE.
3971 026170 026332      RANWR          ;WRITE.
3972 026172 026332      RANWR          ;WRITE.
3973 026174 026332      RANWR          ;WRITE.
3974 026176 026332      RANWR          ;WRITE.
3975 026200 026220      RANRD          ;READ.
3976 026202 026220      RANRD          ;READ.
3977 026204 026220      RANRD          ;READ.
3978 026206 026220      RANRD          ;READ.
3979 026210 026220      RANRD          ;READ.
3980 026212 026220      RANRD          ;READ.
3981 026214 026220      RANRD          ;READ.
3982 026216 026220      RANRD          ;READ.
3983
3984
3985      ;      SUBROUTINE TO SET UP READ COMMANDS IN SEQUENCE TABLE.
3986      ;      INPUTS:
3987      ;      OUTPUTS:
3988      ;      REGISTERS:      R2
3989      ;      CALLS:
3990
3991 026220 005767 155232      RANRD:: TST      WTMFLG          ;WAS LAST CMD A WRITE?
3992 026224 001406          BEQ      1$          ;NO,GO AHEAD
3993 026226 004767 000136          JSR      PC,RAWTM        ;YES PUT DOWN TAPE MARK
3994 026232 004767 000160          JSR      PC,RASFR        ;AND SPACE FILE REV
3995 026236 005067 155214          CLR      WTMFLG          ;THEN CLEAR THE FLAG
3996 026242 020127 003740      1$:  CMP      R1,#SEQEND
3997 026246 002030          BGE      2$
3998 026250 012721 104401          MOV      #RDR,(R1)+      ;STORE READ REV CMD.
3999 026254 012721 004000          MOV      #DATCNT,(R1)+    ;SET BRJ TO MAX FOR READ RANDOM LENGTHS.
4000 026260 066767 155150 155144      ADD      RANS,RANB      ;LET RANB := RANB + RANS
4001 026266 016702 155140          MOV      RANB,R2      ;LET R2 := RANB CLR.BY #RNOPSC
4002 026272 042702 177740          BIC      #RNOPSC,R2
4003 026276 010221          MOV      R2,(R1)+      ;SET RANDOM # OF OPERATIONS.
4004 026300 012721 000007          MOV      #RANP,(R1)+    ;RANDOM PATTERN.
4005 026304 020127 003740          CMP      R1,#SEQEND
4006 026310 002007          BGE      2$

```

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 2: DATA RELIABILITY.

SEQ 0126

```

4007 026312 012721 104001      MOV    #RDF,(R1)+    ;STORE READ FWD CMD.
4008 026316 012721 004000      MOV    #DATCNT,(R1)+ ;SET BRF TO MAX TO READ RANDOM LENGTHS.
4009 026322 010221              MOV    R2,(R1)+    ;SET RANDOM # OF OPERATIONS.
4010 026324 012721 000007      MOV    #RANP,(R1)+    ;RANDOM PATTERN.
4011 026330 000207      2$:    RTS PC
4012
4013      ;      SUBROUTINE TO SET UP A WRITE COMMAND IN THE SEQUENCE TABLE,
4014      ;      THEN A WRITE TAPE MARK AND SPACE FILE REVERSE.
4015      ;
4016      ;      INPUTS:
4017      ;      OUTPUTS:
4018      ;      REGISTERS:
4019      ;      CALLS:
4020
4021 026332 012721 104005      RANWR:: MOV    #WRT,(R1)+    ;STORE WRITE CMD.
4022 026336 004767 000102      JSR PC,RANWR    ;STORE BRF, # OF OPERATIONS, PATTERN.
4023 026342 005767 155110      TST     WTMFLG    ;LAST CMD A WRT?
4024 026346 001002              BNE     1$        ;YES,RETURN
4025 026350 005267 155102      INC     WTMFLG    ;NO,SET THE FLAG
4026 026354 000207      1$:    RTS PC
4027
4028
4029      ;      SUBROUTINE TO SET UP A WRITE/VERIFY COMMAND IN THE SEQUENCE TABLE.
4030      ;      INPUTS:
4031      ;      OUTPUTS:
4032      ;      REGISTERS:
4033      ;      CALLS:
4034
4035 026356 012721 104105      RANWV:: MOV    #WTV,(R1)+    ;STORE WRITE/VERIFY CMD.
4036 026362 004767 000056      JSR PC,RANWV    ;STORE BRF, # OF OPERATIONS, PATTERN.
4037 026366 000207      RTS     PC
4038
4039
4040      ;      SUBROUTINE TO SET UP A WRITE TAPE MARK IN THE SEQUENCE TABLE.
4041      ;      INPUTS:
4042      ;      OUTPUTS:
4043      ;      REGISTERS:
4044      ;      CALLS:
4045
4046 026370 020127 003740      RAWTM:: CMP     R1,#SEQEND
4047 026374 002007              BGE     1$
4048 026376 012721 100011      MOV     #WTM,(R1)+    ;STORE WRITE TAPE MARK CMD.
4049 026402 012721 000001      MOV     #1,(R1)+    ;BRF
4050 026406 012721 000001      MOV     #1,(R1)+    ;# OF OPERATIONS
4051 026412 005721              TST     (R1)+    ;SKIP PATTERNS
4052 026414 000207      1$:    RTS PC
4053
4054      ;      SUBROUTINE TO SET UP A SPACE FILE REVERSE IN THE SEQUENCE TABLE.
4055      ;      INPUTS:
4056      ;      OUTPUTS:
4057      ;      REGISTERS:
4058      ;      CALLS:
4059
4060 026416 020127 003740      RASFR:: CMP     R1,#SEQEND
4061 026422 002007              BGE     1$
4062 026424 012721 105410      MOV     #SFR,(R1)+    ;STORE SPACE FILE REVERSE
4063 026430 012721 000001      MOV     #1,(R1)+    ;BRF

```

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 2: DATA RELIABILITY.

SEQ 0127

```

4064 026434 012721 000001      MOV    #1,(R1)+      ;# OF OPERATIONS
4065 026440 005721             TST    (R1)+      ;SKIP PATTERNS
4066 026442 000207             11$: RTS    PC
4067
4068
4069 :                           SUBROUTINE TO STORE BR# , # OF OPERATIONS, PATTERN IN COMMAND
4070 :                           SEQUENCE TABLE FOR WRITE AND WRITE/VERIFY COMMANDS.
4071 :                           INPUTS:
4072 :                           OUTPUTS:
4073 :                           REGISTERS:      R2
4074 :                           CALLS:
4075
4076 026444 012721 004000      RANW:: MOV    #DATCNT,(R1)+      ;SET BR# TO MAX FOR PATTERN GENERATION.
4077                                     ;RANDOM BR# WILL BE GENERATED FOR EACH RECORD.
4078 026450 066767 154760 154754  ADD    RANS,RAHB      ;LET RANB := RANB + RANS
4079 026456 016702 154750      MOV    RANB,R2      ;LET R2 := RANB CLR.BY #RNOPSC
4080 026462 042702 177740      BIC    #RNOPSC,R2
4081 026466 010221             MOV    R2,(R1)+      ;SET RANDOM # OF OPERATIONS.
4082 026470 012721 000007      MOV    #RANP,(R1)+      ;RANDOM PATTERN.
4083 026474 000207             RTS    PC      ;RETURN.
4084
4085 .EVEN
4086
4087 026476             ENDTST
      026476             L10034:
      026476 104401             TRAP    C#ETST
4088
4089 .SBTTL TEST 3: WRITE COMPATABILITY/WRITE UTILITY.
4090
4091 :++
4092 : TEST TO WRITE RECORDS FROM BOT TO EOT.
4093 :--
4094
4095 026500             BGNTST
      026500             T3::
4096
4097 026500 112767 000001 155007  MOVB    #1,RANDOM      ;SET THE RANDOM OPERATIONS FLAG.
4098 026506 105067 155002      CLRB    EXPBOT ;LET EXPBOT :B= #0      ;CLEAR EXPECT BOT FLAG.
4099
4100 026512 004767 170364      JSR     PC,FIRSTU      ;FIND THE FIRST UNIT.
4101 026516 004767 160350      JSR     PC,SOFINIT      ;INIT DEVICE
4102 026522 103404             BCS     11$
4103 026524             ERDF    2,NSSRM,STAERM      ;REPORT TS05 NOT READY
      026524 104455             TRAP    C#ERDF
      026526 000002             .WORD   2
      026530 004536             .WORD   NSSRM
      026532 006120             .WORD   STAERM
4104
4105 026534 004767 160726      11$: JSR     PC,M0SET      ;GO DO SETUP'S
4106 026540 012702 004000      MOV     #DATCNT,R2      ;SET UP THE RECORD LENGTH MASK.
4107 026544 005302             DEC     R2
4108 026546 010267 154656      MOV     R2,LENMSK      ;ALLOW MAXIMUM BUFFER.
4109 026552 005167 154652      COM     LENMSK
4110 026556 004767 161400      JSR     PC,SETCH      ;CMD 1 = SET CHARACTERISTIC.
4111 026562 004767 161420      JSR     PC,SETRW      ;CMD2=REWIND
4112 026566 105067 154734      CLRB    STAFLG ;LET STAFLG :B= #0      ;CLEAR START FLAG
4113 026572             50426$: BREAK ; DO A SUPVSR BREAK FIRST.

```

HARDWARE TESTS MACRO M1113 14-JUN-84 18:32
TEST 3: WRITE COMPATABILITY/WRITE UTILITY.

SEQ 0128

026572	104422				TRAP	C\$BRK
4114						
4115	026574			50427\$:		
4116	026574	020127	003740	CMP	R1,#SEQEND	;WHILE THERE IS MORE ROOM IN SEQ TABLE:
4117	026600	002003		BGE	50430\$	
4118	026602	004767	177524	JSR	PC,RANWR	;STORE A WRITE CMD IN SEQUENCE TABLE.
4119	026606	000772		BR	50427\$	
4120	026610			50430\$:		
4121	026610	012711	177777	MOV	#END,(R1)	;STORE END OF SEQUENCE CODE IN TABLE.
4122	026614	004767	161406	JSR	PC,EXALL	;EXECUTE ALL CMDs IN SEQ TBL ON UNITS.
4123	026620	012701	003540	MOV	#CMOSEQ,R1	;INIT SEQ TBL POINTER,
4124	026624	005702		TST	R2	;REPEAT UNTIL EOT IS REACHED
4125	026626	001761		BEQ	50426\$	
4126	026630	105267	154670	INCB	ALLEOT	;SET ALL UNITS @ EOT FLAG
4127	026634	000240		NOP		
4128	026636	000240		NOP		
4129	026640	000240		NOP		
4130	026642	004767	000766	JSR	PC,TSWEOT	;WRITE ONE RECORD BEYOND EOT ON ALL UNITS
4131						;SO THAT SHORTER READ STOP DISTANCE
4132						;SHALL POSITION HEAD IN CLEAN IRG GAP
4133						;READ REV THAT EXTRA REC TO RE-POSITION TAPE
4134	026646	105067	154652	CLRB	ALLEOT	;CLEAR ALL UNITS @ EOT FLAG
4135	026652	004767	161330	JSR	PC,SETRW	;STORE REWIND IN SEQ TBL.
4136	026656	012711	177777	MOV	#END,(R1)	;STORE END IN SEQ TBL.
4137	026662	004767	161340	JSR	PC,EXALL	;EXECUTE REWIND CMD ON ALL UNITS
4138						
4139						
4140	026666			EXIT	TST	
	026666	104432				TRAP
	026670	000002				.WORD
4141						C\$EXIT
4142						L10035-.
4143				.EVEN		
4144	026672			ENDTST		
	026672			L10035:		
	026672	104401				TRAP
4145						C\$ETST
4146						
4147				.SBTTL	TEST 4: READ COMPATABILITY/READ UTILITY.	
4148						
4149				;		
4150				;	TEST TO READ ENTIRE TAPE FORWARD AND REVERSE.	
4151				;		
4152						
4153	026674			BGNTST		
	026674			T4::		
4154						
4155	026674	112767	000001	MOVB	#1,RANDOM	;SET THE RANDOM OPERATIONS FLAG.
4156	026702	112767	000001	MOVB	#1,EXPBOT	;SET EXPECT BOT FLAG.
4157						
4158	026710	004767	170166	JSR	PC,FIRSTU	;FIND THE FIRST UNIT.
4159	026714	004767	160152	JSR	PC,SOFINIT	;INIT DEVICE
4160	026720	103404		BCS	11\$	
4161	026722			ERRDF	2,NSSRM,STAERM	;REPORT TS05 NOT READY
	026722	104455				TRAP
	026724	000002				.WORD
	026726	004536				.WORD
						C\$ERDF
						2
						NSSRM

HARDWARE TESTS MACRO M1113 14 JUN-84 18:32
TEST 4: READ COMPATABILITY/READ UTILITY.

SEQ 0129

```

026730 006120                                .WORD STAERM
4162
4163 026732 004767 160530      11$: JSR    PC,MOSET      ;GO DO SETUP'S
4164 026736 004767 161220      JSR PC,SETCH      ;CMD 1 = SET CHARACTERISTIC.
4165 026742 004767 161240      JSR PC,SETRW      ;CMD2=REWIND.
4166 026746 105067 154554      CLRB   STAFLG ;LET STAFLG :B= 40 ;CLEAR START FLAG
4167 026752 012721 104001      MOV    #RDF,(R1)+    ;CMD3 = READ FORWARD.
4168 026756 012721 004000      MOV    #DATCNT,(R1)+  ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
4169 026762 012721 077777      MOV    #77777,(R1)+  ;SET RECORD COUNT TO MAX FOR WHOLE TAPE.
4170 026766 012721 000007      MOV    #RAMP,(R1)    ;PATTERN = RANDOM.
4171 026772 012711 177777      MOV    #END,(R1)    ;STORE END OF SEQUENCE CODE IN TABLE.
4172 026776 004767 161224      JSR    PC,EXALL      ;EXECUTE ALL CMDS IN SEQ TBL ON ALL UNITS.
4173 027002 105267 154516      INCB   ALLEOT      ;FLAG TO ALLOW ALL UNITS AT EOT TO READ REV
4174 027006 012701 003540      MOV    #CMDSEQ,R1    ;INIT CMD SEQ TBL POINTER.
4175 027012 012721 104401      MOV    #RDR,(R1)+    ;CMD1 = READ REVERSE.
4176 027016 012721 004000      MOV    #DATCNT,(R1)+  ;SET LENGTH TO MAX FOR UNKNOWN LENGTHS.
4177 027022 012721 077777      MOV    #77777,(R1)+  ;RECORD COUNT = MAX FOR WHOLE TAPE.
4178 027026 012721 000007      MOV    #RAMP,(R1)+  ;PATTERN = RANDOM.
4179 027032 012711 177777      MOV    #END,(R1)    ;STORE END OF SEQUENCE CODE IN TABLE.
4180 027036 004767 161164      JSR    PC,EXALL      ;GO EXECUTE READ REV. OF ENTIRE TAPE.
4181 027042 105067 154456      CLRB   ALLEOT      ;CLEAR ALL UNITS @ EOT FLAG
4182
4183 027046      EXIT    TST
      027046 104432
      027050 000002                                TRAP    C$EXIT
                                                    .WORD    L10036-.
4184
4185      .EVEN
4186
4187 027052      ENDTST
      027052
      027052 104401      L10036:                                TRAP    C$ETST
4188
4189      .SBTTL TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
4190
4191      ;**
4192      ; TEST TO EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.
4193      ;--
4194
4195 027054      BGNSTST
      027054      TS::
4196
4197 027054 105067 154435      CLRB   RANDOM      ;CLEAR RANDOM MODE FLAG.
4198 027060 112767 000001 154426      MOVB   #1,EXPBOT    ;SET EXPECT BOT FLAG.
4199
4200 027066 004767 170010      JSR    PC,FIRSTU    ;FIND THE FIRST UNIT.
4201 027072 004767 157774      JSR    PC,SOFINIT    ;INIT DEVICE
4202 027076 103404      BCS     11$
4203 027100      ERDF    2,NSSRM,STAERM    ;REPORT TS05 NOT READY
      027100 104455
      027102 000002                                TRAP    C$ERDF
      027104 004536                                .WORD    2
      027106 006120                                .WORD    NSSRM
                                                    .WORD    STAERM
4204
4205 027110 004767 160352      11$: JSR    PC,MOSET      ;GO DO SETUP'S
4206 027114 116767 153076 154377      MOVB   PIRE,IRE    ;MOVE INHIBIT RFC ERROR REPORT FLAG.
4207 027122 004767 161034      JSR PC,SETCH      ;CMD 1 = SET CHARACTERISTIC.
4208 027126 016767 153066 154406      MOV    CHAR,CMDSEQ+2 ;MOVE CHAR CODE FROM P TBL TO SEQ TBL.

```

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 0130

4209	027134	012702	002222	MOV	#CMD0,R2	;R2 POINTS TO CMD0 IN SOFT P TABLE.
4210	027140	004767	000446	JSR	PC,PTCMD5	;MOVE CMD 2 FROM P TBL TO SEQ TBL.
4211	027144	004767	000442	JSR	PC,PTCMD5	;MOVE CMD 3 FROM P TBL TO SEQ TBL.
4212	027150	004767	000436	JSR	PC,PTCMD5	;MOVE CMD 4 FROM P TBL TO SEQ TBL.
4213	027154	004767	000432	JSR	PC,PTCMD5	;MOVE CMD 5 FROM P TBL TO SEQ TBL.
4214	027160	004767	000426	JSR	PC,PTCMD5	;MOVE CMD 6 FROM P TBL TO SEQ TBL.
4215	027164	004767	000422	JSR	PC,PTCMD5	;MOVE CMD 7 FROM P TBL TO SEQ TBL.
4216	027170	004767	000416	JSR	PC,PTCMD5	;MOVE END CMD FROM P TBL TO SEQ TBL.
4217	027174	005067	154242	CLR	JLOOP	;CLEAR JMP CMD LOOP COUNT.
4218	027200	105067	154322	CLRB	STAFLG	;CLEAR START FLAG
4219	027204	012701	003540	MOV	#CMDSEQ,R1	;INIT SEQUENCE TABLE POINTER.
4220	027210			3\$:	;WHILE (R1) NE #END DO	;WHILE THERE ARE CMD5 LEFT IN SEQUENCE TBL:
4221	027210			50431\$:		
4222	027210	021127	177777	CMP	(R1),#END	
4223	027214	001574		BEQ	50432\$	
4224	027216	022711	000040	CMP	#JMP.C,(R1)	;IS THIS A JUMP CMD?
4225	027222	001024		BNE	6\$	;BR IF NOT.
4226	027224	062701	000002	ADD	#2,R1 ;LET R1 := R1 + #2	;POINT TO BR.
4227	027230	012167	154210	MOV	(R1)+,JLOC	;SAVE BR (LOCATION).
4228	027234	022167	154202	CMP	(R1)+,JLOOP	;HAS LOOP COUNT BE SATISFIED?
4229	027240	001003		BNE	1\$	;IF NOT, JMP AGAIN.
4230	027242	062701	000002	ADD	#2,R1	;IF SO, ADJUST SEQ POUNTER
4231	027246	000760		BR	3\$	;AND GO TO NEXT COMMAND.
4232	027250	005267	154166	1\$:	INC JLOOP	;UPDATE THE LOOP COUNT.
4233	027254	012701	003540	MOV	#CMDSEQ,R1	;INIT CMD SEQ TABLE POINTER.
4234	027260	005367	154160	2\$:	DEC JLOC	;DECR LOCATION COUNTER.
4235	027264	001751		BEQ	3\$	;IF THIS IS THE RIGHT LOCATION TO JMP TO, GO SET
4236	027266	062701	000010	ADD	#10,R1	;IF NOT, UPDATE SEQ POINTER TO NEXT CMD.
4237	027272	000772		BR	2\$	;DO IT AGAIN.
4238						
4239	027274	022711	000020	6\$:	CMP #DLY.C,(R1)	;DELAY?
4240	027300	001026		BNE	4\$	;BR IF NOT.
4241	027302	062701	000004	ADD	#4,R1	;R1 = LOCATION OF N COUNT.
4242	027306	011167	154126	MOV	(R1),TIME2	;SAVE N COUNT.
4243	027312			7\$:	DELAY 1	;GO TO SUPER WAIT 1 MSEC.
	027312	012727	000001			
	027316	000000				MOV #1,(PC)+
	027320	016727	152572			.WORD 0
	027324	000000				MOV L\$DLY,(PC)+
	027326	005367	177772			.WORD 0
	027332	001375				DEC -6(PC)
	027334	005367	177756			BNE .-4
	027340	001367				DEC -22(PC)
						BNE .-20
4244	027342	005367	154072	DEC	TIME2	
4245	027346	001361		BNE	7\$	
4246	027350	062701	000004	ADD	#4,R1 ;LET R1 := R1 + #4	;POINT TO NEXT CMD.
4247	027354	000715		BR	3\$	;GO CHECK NEXT CMD.
4248	027356	004767	161610	4\$:	JSR PC,SETUP	;GO SETUP THE COMMAND BLOCK.
4249	027362			50433\$:	;WHILE NCNT LT NCNT1 DO	;WHILE THERE ARE RECORDS REMAINING:
4250	027362	026767	154024	154024	CMP NCNT,NCNT1	
4251	027370	002103		BGE	50434\$	
4252	027372	004767	161466	JSR	PC,CMDAC	;STORE CMD ASCII IN ERROR MSG.
4253	027376	004767	161122	JSR	PC,EXSUB	;ISSUE CMD TO ALL,AWAIT INTS,CHECK STATUS.
4254	027402	026727	154012	100017	CMP CMDWRD,#GES	;IF CMD IS GET STATUS THEN:
4255	027410	001002		BNE	50435\$	
4256	027412	004767	167774		JSR PC,PRXST	;PRINT EXTENDED STATUS REGISTERS.
4257						

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 0131

4258 027416				50435:		
4259 027416	004767	170056			JSR PC,CKHAE	;CHECK HALT AFTER EACH CMD FLAG.
4260 027422	012702	000001			MOV #1,R2	;SET ALL UNITS AT BOT/EOT.
4261 027426	004767	167450			JSR PC,FIRSTU	;FIND FIRST UNIT.
4262 027432				50436:	WHILE DEVTBL(R5) NE #END DO	;WHILE THERE ARE MORE UNITS:
4263 027432	026527	002604	177777		CMP DEVTBL(R5),#END	
4264 027440	001426				BEQ 50437	
4265 027442	032767	000400	153750		BIT #MOD.CO,CMDWRD	;IF CMD IS REVERSE THEN:
4266 027450	001406				BEQ 50440	
4267 027452	032765	000002	003502		BIT #X0.BOT,EOTFLG(R5)	;IF NOT AT BOT THEN:
4268 027460	001001				BNE 50441	
4269 027462	005002				CLR R2	;CLEAR EOT/BOT FLAG.
4270						
4271 027464				50441:		
4272 027464	000411				BR 50442	;ELSE IF CMD IS NOT REVERSE:
4273 027466				50440:		
4274 027466	032765	000001	003502		BIT #X0.EOT,EOTFLG(R5)	
4275 027474	001404				BEQ 50443	
4276 027476	032767	000001	153714		BIT #CMD.CO,CMDWRD	
4277 027504	001001				BNE 50444	
4278 027506				50443:		
4279						
4280 027506	005002				CLR R2	;IF NOT AT EOT OR NOT A MOTION CMD THEN: ;LET R2 := #0 ;CLEAR EOT/BOT FLAG.
4281						
4282 027510				50444:		
4283						
4284 027510				50442:		
4285 027510	004767	167434			JSR PC,NEXTU	;FIND NEXT UNIT
4286						
4287 027514	000746				BR 50436	
4288 027516				50437:		
4289 027516	020227	000001			CMP R2,#1	;IF ALL UNIT ARE AT EOT/BOT THEN:
4290 027522	001016				BNE 50445	
4291 027524	016767	153662	153662		MOV NCNT,NCNT1	;FORCE TERMINATION OF COMMAND.
4292 027532	005267	153656			INC NCNT1	
4293 027536	105267	153762			INCB ALLEOT	;FLAG ALL UNITS AT EOT/BOT TO ALLOW VERIFY OF D
4294 027542	026727	153660	000002		CMP CMDLG,#2	;WHEN WRITING IS CURRENT COMMAND
4295 027550	001002				BNE 50446	
4296 027552	004767	000056			JSR PC,TSWEOT	;GO WRITE/READ REV ONE RECORD BEYOND EOT
4297						
4298 027556				50446:		
4299						
4300 027556	000402				BR 50447	
4301 027560				50445:		
4302 027560	105067	153740			CLRB ALLEOT	;WHEN NOT ALL SEOT, CLEAR FLAG
4303						
4304 027564				50447:		
4305 027564	005267	153622			INC NCNT	;UPDATE RECORD COUNT.
4306 027570	016767	153624	153626		MOV CMDWRD,PCMDWD	;SAVE PREVIOUS COMMAND WORD.
4307						
4308 027576	000671				BR 50433	
4309 027600				50434:		
4310 027600	004767	166262			JSR PC,VFYDAT	;IF LAST CMD WAS A WRITE VERIFY, THEN GO ;VERIFY THE LAST N RECORDS OF DATA.
4311						
4312						
4313 027604	000601				BR 50431	
4314 027606				50432:		

HARDWARE TESTS MACRO M1113 14 JUN 84 18:32
TEST 5: EXECUTE OPERATOR SELECTED COMMAND SEQUENCE.

SEQ 0132

```

4315
4316 027606          EXIT    TST
      027606 104432
      027610 000140          TRAP    C$EXIT
                                .WORD    L10037 .
4317
4318      ;          SUBROUTINE TO MOVE A COMMAND FROM THE SOFTWARE P TABLE TO
4319      ;          THE COMMAND SEQUENCE TABLE.
4320      ;          INPUTS:          R2 = POINTER TO SOFT "P" TABLE
4321      ;          OUTPUTS:
4322      ;          REGISTERS:      R3.
4323      ;          CALLS:
4324
4325 027612 012203      PTCMD5: MOV    (R2)+,R3          ;R3 = COMMAND TABLE INDEX.
4326 027614 005303      DEC    R3
4327 027616 006303      ASL    R3
4328 027620 016321 003752  MOV    CMDTBL(R3),(R1)+      ;MOVE COMMAND WORD.
4329 027624 012221      MOV    (R2)+,(R1)+      ;MOVE # OF BYTES.
4330 027626 012221      MOV    (R2)+,(R1)+      ;MOVE # OF OPERATIONS.
4331 027630 012221      MOV    (R2)+,(R1)+      ;MOVE PATTERN CODE.
4332 027632 000207      RTS PC
4333
4334      ;          SUBROUTINE TO WRITE THEN READ REVERSE ONE RECORD BEYOND EOT
4335      ;          INPUTS:
4336      ;          OUTPUTS:
4337      ;          REGISTERS:
4338      ;          CALLS:      CMDAC,EXSUB,CKHAE
4339
4340 027634 000240      TSWEOT: NOP
4341 027636 000240      NOP
4342 027640 004767 160660  JSR PC,EXSUB      ;WRITE ONE RECORD BEYOND EOT
4343 027644 004767 167630  JSR PC,CKHAE      ;SO THAT READ SHORTER STOP DISTANCE
4344
4345 027650 012700 000002  ;SHALL POSITION HEAD IN CLEAN IRG GAP
4346 027654 016767 153540 153542 14:  MOV    #2,R0      ;SET UP COUNTER FOR EOT
4347 027662 012767 104401 153530  MOV    CMDWRD,PCMDWD      ;LET PCMDWD := CMDWRD ;REPOSITION TAPE
4348 027670 012767 000004 153530  MOV    #RDR,CMDWRD      ;LET CMDWRD := #RDR ;BEFORE EXTRA RECORD
4349 027676 016767 153516 152424  MOV    #4,CMDLG      ;BY READING REVERSE
4350 027704 042767 004000 152416  MOV    CMDWRD,CMDPKT      ;LET CMDPKT := CMDWRD CLR.BY #BRF.C
4351 027712 016767 152412 153502  BIC    #BRF.C,CMDPKT
4352 027720 016767 153464 152404  MOV    CMDPKT,CMDSAV      ;LET CMDSAV := CMDPKT ;THAT RECORD TO ALLOW
4353 027726 004767 161132      MOV    DATARD,CMDPKT+CP.ADL      ;NEXT COMMAND IN THE
4354 027732 004767 160566      JSR PC,CMDAC      ;TABLE TO BE EXECUTED
4355 027736 004767 167536      JSR PC,EXSUB
4356 027742 005300      JSR PC,CKHAE
4357 027744 001343      DEC    R0      ;FOUND EOT YET?
4358 027746 000207      BNE    14      ;NO,KEEP GOING
4359      RTS PC      ;YES,RETURN
4360
4361      .EVEN
4362 027750          ENDTST
      027750          L10037:
      027750 104401          TRAP    C$ETS*
4363
4364 027752          ENDMOD
4365
4366      .TITLE PARAMETER CODING
4367

```

011

```

4368 .SBTTL  HARDWARE PARAMETER CODING SECTION
4369
4370 027752      BGNMOD
4371
4372 ;**
4373 ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
4374 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
4375 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
4376 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
4377 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
4378 ; WITH THE OPERATOR.
4379 ;--
4380
4381 027752      BGNHRD
4382 027752      000042
4383 027754      L$HARD::
4384 027754      GPRMA  TSSADR,0,0,160010,177564,YES
4385 027754      GPRMD  TSSVCT,2,0,777,60,776,YES
4386 030010      EXIT HRD
4387 030010      024004
4388
4389 030012      124      123      104  TSSADR: .NLIST  BEX
4390 030027      126      105      103  TSSVCT: .ASCIZ  /TSDB ADDRESS/
4391 030036      123      105      114  TSSUNT: .ASCIZ  /VECTOR/
4392
4393
4394
4395 030060      .LIST  BEX
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405

```

```

4406      ; -
4407
4408 030060      BGNSFT
      030060 000302
      030062
4409 030062      L$SOFT::
      030062 000130      GPRML CLRM,0,1,YES
      030064 030666
      030066 000001
4410 030070      GPRML RRVM,0,400,YES
      030070 000130
      030072 030705
      030074 000400
4411 030076      GPRML RCVERM,2,400,YES
      030076 001130
      030100 031010
      030102 000400
4412 030104      GPRML HAEM,2,1,YES
      030104 001130
      030106 030734
      030110 000001
4413 030112      GPRML IRECH,6,400,YES
      030112 003130
      030114 031064
      030116 000400
4414 030120      XFERT NEXTSP
      030120 004024
4415 030122      GPRML BADTM,4,1,YES
      030122 002130
      030124 030760
      030126 000001
4416 030130      NEXTSP: GPRML DINTM,6,1,YES
      030130 003130
      030132 031041
      030134 000001
4417 030136      GPRML IREM,12,1,YES
      030136 005130
      030140 031131
      030142 000001
4418 030144      GPRML CHGM,10,1,YES
      030144 004130
      030146 031105
      030150 000001
4419 030152      XFERF ENDSP1
      030152 127044
4420 030154      GPRMD CHARM,14,0,377,0,777,YES
      030154 006032
      030156 031162
      030160 000377
      030162 000000
      030164 000777
4421 030166      GPRMD CMD2M,16,D,37,1,33,YES
      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 RRVM
 .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 BADTM
 .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

4422	030200		GPRMD	BPCRM,20,D,-1,1,DATCNT,YES		.WORD	T\$CODE
	030200	010052				.WORD	BPCRM
	030202	031215				.WORD	-1
	030204	177777				.WORD	T\$LOLIM
	030206	000001				.WORD	T\$HILIM
	030210	004000					
4423	030212		GPRMD	NUMBM,22,D,-1,1,77777,YES		.WORD	T\$CODE
	030212	011052				.WORD	NUMBM
	030214	031227				.WORD	1
	030216	177777				.WORD	T\$LOLIM
	030220	000001				.WORD	T\$HILIM
	030222	077777					
4424	030224		GPRMD	PATTM,24,D,17,0,10,YES		.WORD	T\$CODE
	030224	012052				.WORD	PATTM
	030226	031247				.WORD	17
	030230	000017				.WORD	T\$LOLIM
	030232	000000				.WORD	T\$HILIM
	030234	000010					
4425	030236		GPRMD	CMD3M,26,D,37,1,33,YES		.WORD	T\$CODE
	030236	013052				.WORD	CMD3M
	030240	031356				.WORD	37
	030242	000037				.WORD	T\$LOLIM
	030244	000001				.WORD	T\$HILIM
	030246	000033					
4426	030250		GPRMD	BPCRM,30,D,-1,1,DATCNT,YES		.WORD	T\$CODE
	030250	014052				.WORD	BPCRM
	030252	031215				.WORD	-1
	030254	177777				.WORD	T\$LOLIM
	030256	000001				.WORD	T\$HILIM
	030260	004000					
4427	030262		GPRMD	NUMBM,32,D,-1,1,77777,YES		.WORD	T\$CODE
	030262	015052				.WORD	NUMBM
	030264	031227				.WORD	-1
	030266	177777				.WORD	T\$LOLIM
	030270	000001				.WORD	T\$HILIM
	030272	077777					
4428	030274		GPRMD	PATTM,34,D,17,0,10,YES		.WORD	T\$CODE
	030274	016052				.WORD	PATTM
	030276	031247				.WORD	17
	030300	000017				.WORD	T\$LOLIM
	030302	000000				.WORD	T\$HILIM
	030304	000010					
4429	030306		GPRMD	CMD4M,36,D,37,1,33,YES		.WORD	T\$CODE
	030306	017052				.WORD	CMD4M
	030310	031364				.WORD	37
	030312	000037				.WORD	T\$LOLIM
	030314	000001				.WORD	T\$HILIM
	030316	000033					
4430	030320		GPRMD	BPCRM,40,D,-1,1,DATCNT,YES		.WORD	T\$CODE
	030320	020052				.WORD	BPCRM
	030322	031215				.WORD	-1
	030324	177777				.WORD	T\$LOLIM
	030326	000001				.WORD	T\$HILIM
	030330	004000					
4431	030332		GPRMD	NUMBM,42,D,-1,1,77777,YES		.WORD	T\$CODE
	030332	021052				.WORD	NUMBM
	030334	031227					

	030336	177777			.WORD	1
	030340	000001			.WORD	T\$LOLIM
	030342	077777			.WORD	T\$HILIM
4432	030344		GPRMD	PATTM,44,D,17,0,10,YES		
	030344	022052			.WORD	T\$CODE
	030346	031247			.WORD	PATTM
	030350	000017			.WORD	17
	030352	000000			.WORD	T\$LOLIM
	030354	000010			.WORD	T\$HILIM
4433	030356		GPRMD	CMD5M,46,D,37,1,33,YES		
	030356	023052			.WORD	T\$CODE
	030360	031372			.WORD	CMD5M
	030362	000037			.WORD	37
	030364	000001			.WORD	T\$LOLIM
	030366	000033			.WORD	T\$HILIM
4434	030370		GPRMD	BPCRM,50,D,-1,1,DATCNT,YES		
	030370	024052			.WORD	T\$CODE
	030372	031215			.WORD	BPCRM
	030374	177777			.WORD	-1
	030376	000001			.WORD	T\$LOLIM
	030400	004000			.WORD	T\$HILIM
4435	030402		GPRMD	NUMBM,52,D,-1,1,77777,YES		
	030402	025052			.WORD	T\$CODE
	030404	031227			.WORD	NUMBM
	030406	177777			.WORD	-1
	030410	000001			.WORD	T\$LOLIM
	030412	077777			.WORD	T\$HILIM
4436	030414		GPRMD	PATTM,54,D,17,0,10,YES		
	030414	026052			.WORD	T\$CODE
	030416	031247			.WORD	PATTM
	030420	000017			.WORD	17
	030422	000000			.WORD	T\$LOLIM
	030424	000010			.WORD	T\$HILIM
4437	030426		XFER	ENDSP2		
	030426	002004			.WORD	T\$CODE
4438	030430		ENDSP1: XFER	ENDSP3		
	030430	076004			.WORD	T\$CODE
4439	030432		ENDSP2: GPRMD	CMD6M,56,D,37,1,33,YES		
	030432	027052			.WORD	T\$CODE
	030434	031400			.WORD	CMD6M
	030436	000037			.WORD	37
	030440	000001			.WORD	T\$LOLIM
	030442	000033			.WORD	T\$HILIM
4440	030444		GPRMD	BPCRM,60,D, 1,1,DATCNT,YES		
	030444	030052			.WORD	T\$CODE
	030446	031215			.WORD	BPCRM
	030450	177777			.WORD	-1
	030452	000001			.WORD	T\$LOLIM
	030454	004000			.WORD	T\$HILIM
4441	030456		GPRMD	NUMBM,62,D,-1,1,77777,YES		
	030456	031052			.WORD	T\$CODE
	030460	031227			.WORD	NUMBM
	030462	177777			.WORD	-1
	030464	000001			.WORD	T\$LOLIM
	030466	077777			.WORD	T\$HILIM
4442	030470		GPRMD	PATTM,64,D,17,0,10,YES		
	030470	032052			.WORD	T\$CODE

PARAMETER CODING MACRO M1113 14-JUN-84 18:32
SOFTWARE PARAMETER CODING SECTION

SEQ 0137

	030472	031247		.WORD	PATM
	030474	000017		.WORD	17
	030476	000000		.WORD	T\$LOLIM
	030500	000010		.WORD	T\$HILIM
4443	030502		GPRMD CMD7M,66,D,37,1,33,YES		
	030502	033052		.WORD	T\$CODE
	030504	031406		.WORD	CMD7M
	030506	000037		.WORD	37
	030510	000001		.WORD	T\$LOLIM
	030512	000033		.WORD	T\$HILIM
4444	030514		GPRMD BPCRM,70,D, 1,1,DATCNT,YES		
	030514	034052		.WORD	T\$CODE
	030516	031215		.WORD	BPCRM
	030520	177777		.WORD	-1
	030522	000001		.WORD	T\$LOLIM
	030524	004000		.WORD	T\$HILIM
4445	030526		GPRMD NUMBM,72,D,-1,1,77777,YES		
	030526	035052		.WORD	T\$CODE
	030530	031227		.WORD	NUMBM
	030532	177777		.WORD	-1
	030534	000001		.WORD	T\$LOLIM
	030536	077777		.WORD	T\$HILIM
4446	030540		GPRMD PATM,74,D,17,0,10,YES		
	030540	036052		.WORD	T\$CODE
	030542	031247		.WORD	PATM
	030544	000017		.WORD	17
	030546	000000		.WORD	T\$LOLIM
	030550	000010		.WORD	T\$HILIM
4447	030552		GPRMD CMD8M,76,D,37,1,33,YES		
	030552	037052		.WORD	T\$CODE
	030554	031414		.WORD	CMD8M
	030556	000037		.WORD	37
	030560	000001		.WORD	T\$LOLIM
	030562	000033		.WORD	T\$HILIM
4448	030564		GPRMD BPCRM,100,D,-1,1,DATCNT,YES		
	030564	040052		.WORD	T\$CODE
	030566	031215		.WORD	BPCRM
	030570	177777		.WORD	-1
	030572	000001		.WORD	T\$LOLIM
	030574	004000		.WORD	T\$HILIM
4449	030576		GPRMD NUMBM,102,D, 1,1,77777,YES		
	030576	041052		.WORD	T\$CODE
	030600	031227		.WORD	NUMBM
	030602	177777		.WORD	-1
	030604	000001		.WORD	T\$LOLIM
	030606	077777		.WORD	T\$HILIM
4450	030610		GPRMD PATM,104,D,17,0,10,YES		
	030610	042052		.WORD	T\$CODE
	030612	031247		.WORD	PATM
	030614	000017		.WORD	17
	030616	000000		.WORD	T\$LOLIM
	030620	000010		.WORD	T\$HILIM
4451	030622		XFER ENDSP		
	030622	022004		.WORD	T\$CODE
4452	030624		ENDSP3: GPRML TSMD,106,1,YES		
	030624	043130		.WORD	T\$CODE
	030626	031257		.WORD	TSMD

	030630	000001								.WORD	1
4453	030632					XFERT	ENDSP				
	030632	016024								.WORD	T\$CODE
4454	030634					GPRML	FAST,114,1,YES				
	030634	046130								.WORD	T\$CODE
	030636	031347								.WORD	FAST
	030640	000001								.WORD	1
4455	030642					XFERT	ENDSP4				
	030642	011024								.WORD	T\$CODE
4456	030644					GPRML	WTBF,112,1,YES				
	030644	045130								.WORD	T\$CODE
	030646	031327								.WORD	WTBF
	030650	000001								.WORD	1
4457	030652					XFERT	ENDSP				
	030652	006024								.WORD	T\$CODE
4458	030654					GPRML	RDBF,110,1,YES				
	030654	044130								.WORD	T\$CODE
	030656	031310								.WORD	RDBF
	030660	000001								.WORD	1
4459	030662					ENDSP5: XFER	ENDSP				
	030662	002004								.WORD	T\$CODE
4460	030664					ENDSP4: XFER	ENDSP				
	030664	001004								.WORD	T\$CODE
4461	030666					ENDSP:					
4462	030666					ENDSF T					
	030666					L10041:				.EVEN	
4463						.EVEN					
4464											
4465						.NLIST	BEX				
4466	030666	103	114	105	CLRM:	.ASCIZ	/CLEAR COUNTERS/				
4467	030705	122	105	123	RRVM:	.ASCIZ	/RESET RANDOM VARIABLES/				
4468	030734	110	101	114	HAEM:	.ASCIZ	/HALT AFTER EACH CMD/				
4469	030760	102	101	104	BADTM:	.ASCIZ	/BAD TAPE SPOT DETECTION/				
4470	031010	120	122	111	RCVERM:	.ASCIZ	/PRINT RECOVERABLE ERRORS/				
4471	031041	104	111	123	DINTM:	.ASCIZ	/DISABLE INTERRUPTS/				
4472	031064	111	116	110	IRECH:	.ASCIZ	/INHIBIT RECOVERY/				
4473	031105	103	110	101	CHGM:	.ASCIZ	/CHANGE CMD SEQUENCE/				
4474	031131	111	116	110	IREH:	.ASCIZ	/INHIBIT RFC ERROR REPORT/				
4475	031162	103	110	101	CHARM:	.ASCIZ	/CHARACTERISTICS CODE/				
4476	031207	103	115	104	CMD2M:	.ASCIZ	"CMD/2"				
4477	031215	102	122	106	BPCRM:	.ASCIZ	/BRF COUNT/				
4478	031227	043	040	117	NUMBM:	.ASCIZ	/# OF OPERATIONS/				
4479	031247	120	101	124	PATTM:	.ASCIZ	/PATTERN/				
4480	031257	104	105	106	TSMD:	.ASCIZ	/DEFAULT SWITCH SETTINGS?/				
4481	031310	122	105	101	RDBF:	.ASCIZ	/READ BUFFERING/				
4482	031327	127	122	111	-WTBF:	.ASCIZ	/WRITE BUFFERING/				
4483	031347	061	060	060	FAST:	.ASCIZ	/100IPS/				
4484					.LIST	BEX					
4485					.EVEN						
4486											
4487					.NLIST	BEX					
4488	031356	103	115	104	CMD3M:	.ASCIZ	"CMD/3"				
4489	031364	103	115	104	CMD4M:	.ASCIZ	"CMD/4"				

PARAMETER CODING MACRO M1113 14-JUN-84 18:32
SOFTWARE PARAMETER CODING SECTION

SEQ 0139

```

4493 031414      103      115      104  CMD8M: .ASCIZ  "CMD/8"
4494                                     .LIST  BEX
4495                                     .EVEN
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506 031422      PATCH:: .BLKW   64.
4507
4509              032000      . = .!377+1
4511
4512 032000      LASTAD
                                     .EVEN
                                     .WORD   0
                                     .WORD   0
4513 032004      L$LAST::
4514              ENDMOD
4515
4516              .SBTTL  HARD CODED P-TBL
4517
4518              ;++
4519              ;DIAG IS PRE-PARAMETERIZED PER TBL
4520              ;--
4521 032004      BGNSETUP 1
4522 032004      BGNPTAB
                                     .WORD   0
                                     .WORD  L10044 ./2-1
4523 032010      172522
4524 032012      000224
4525 032014      000000
4526 032016      0
4527 032016      L10044:  ENDPTAB
4528              ENDSETUP
4529              .END

```

PARAMETER CODING
SYMBOL TABLE

MACRO M1113 14-JUN-84 18:32

SEQ 0140

ACK.C = 100000 G	BTADDR 002616 G	CP.CNT= 000006 G	C#TPRI= 000013	EXPBOT 003514 G
ADR = 000020 G	BTMSG1 015112	CRLF 005741 G	DATARD 003410 G	EXSUB 010524 G
ALLEOT 003524 G	BTMSG2 015177	CRLFSP 005744 G	DATAWT 003406 G	EXTFEA 002322 G
ASSEMB= 000010	BTMSG3 015247	CTCC 003450 G	DATCNT= 004000 G	E#END = 002100
ATTNM 004603 G	BTPT 003512 G	CVC.C = 040000 G	DATERM 005752 G	E#LOAD= 000035
AUDRPM 005114 G	BTRPT 020164 G	C#AU = 000052	DEVTBL 002604 G	FAST 031347
AUDRUN 005146 G	BT0 003046 G	C#AUTO= 000061	DFPTBL 002174 G	FATSM 004642 G
AUTODM 023604	BT1 003120 G	C#BRK = 000022	DFTSCH= 000040 G	FIRSTU 017102 G
BADTM 030760	BT2 003172 G	C#BSEG= 000004	DIA = 100006 G	FMT.CO= 000040 G
BADTSW 002210 G	BT3 003244 G	C#BSUB= 000002	DIABLK= 003406 G	FMT.C1= 000100 G
BENBSW 002324 G	CHAR 002220 G	C#CEFG= 000045	DIACNT= 000020 G	FTLCNT 003366 G
BFSEQ 025050 G	CHARM 031162	C#CLCK= 000062	DIAGMC= 000000	FUNRM 004622 G
BFSEQ0 025074	CHGFLG 002214 G	C#CLEA= 000012	DINT 002212 G	F#AU = 000015
BFSEQ1 025146	CHGM 031105	C#CLOS= 000035	DINTM 031041	F#AUTO= 000020
BFSEQ2 025160	CHKERR 013210 G	C#CLP1= 000006	DLY = 000020 G	F#BGN = 000040
BFSEQ3 025252	CH.EAI= 000040 G	C#CVEC= 000036	DLY.C = 000020 G	F#CLEA= 000007
BFSEQ4 025324	CH.ERI= 000020 G	C#DCLN= 000044	DRI = 100013 G	F#DU = 000016
BFSEQ5 025366	CH.ESS= 000200 G	C#DODU= 000051	DROPM 005065 G	F#END = 000041
BFSEQ6 025440	CKDATA 016466 G	C#DRPT= 000024	DROPEO 003522 G	F#HARD= 000004
BFSEQ7 025512	CKDCNT 017076	C#DU = 000053	DROPN 017410	F#HW = 000013
BFSEQ8 025544	CKDFF 017100	C#EDIT= 000003	DROPU 017200 G	F#INIT= 000006
BFSEQ9 025576	CKHAE 017500 G	C#ERDF= 000055	DROPUA 017324	F#JMP = 000050
BFSE10 025620	CKHRTN 017566	C#ERHR= 000056	DRORTN 017402	F#MOD = 000000
BGNFLG= 003460	CLN = 101012 G	C#ERRO= 000060	DTAERM 005752 G	F#MSG = 000011
BINC 016052	CLRERR 012664 G	C#ERSF= 000054	DTAER2 005226 G	F#PROT= 000021
BIT0 = 000001 G	CLRFLG 002204 G	C#ERSO= 000057	DTAER3 005275 G	F#PWR = 000017
BIT00 = 000001 G	CLRM 030666	C#ESCA= 000010	DTAER4 005337 G	F#RPT = 000012
BIT01 = 000002 G	CMDAC 011064 G	C#ESEG= 000005	DTAER5 005360 G	F#SEG = 000003
BIT02 = 000004 G	CMDASC 004040 G	C#ESUB= 000003	EF.CON= 000036 G	F#SOFT= 000005
BIT03 = 000010 G	CMDD 002222 G	C#ETST= 000001	EF.HSS= 000040 G	F#SRV = 000010
BIT04 = 000020 G	CMDLG 003426 G	C#EXIT= 000032	EF.NEW= 000035 G	F#SUB = 000002
BIT05 = 000040 G	CMDPKM 004346 G	C#GETB= 000026	EF.PWR= 000034 G	F#SW = 000014
BIT06 = 000100 G	CMDPKT 002330 G	C#GETW= 000027	EF.RBO= 000020 G	F#TEST= 000001
BIT07 = 000200 G	CMDSAV 003422 G	C#GMAN= 000043	EF.RES= 000037 G	GCMOA 011136 G
BIT08 = 000400 G	CMDSEQ 003540 G	C#GPHR= 000042	EF.RWB= 000030 G	GENPAT 011556 G
BIT09 = 001000 G	CMDSE2 003550 G	C#GPLO= 000030	EF.STA= 000040 G	GES = 100017 G
BIT1 = 000002 G	CMDTBL 003752 G	C#GPRI= 000040	EINC 016060	GETSTM 005507 G
BIT10 = 002000 G	CMDWRD 003420 G	C#INIT= 000011	END = 177777 G	GIT 012050
BIT11 = 004000 G	CMD.CO= 000001 G	C#INLP= 000020	ENDERF= 003472	GOWAIT 012364 G
BIT12 = 010000 G	CMD.C1= 000002 G	C#MANI= 000050	ENDFLC= 003526	GSCPK 002340 G
BIT13 = 020000 G	CMD.C2= 000004 G	C#MEM = 000031	ENDSP 030666	G#CNT0= 000200
BIT14 = 040000 G	CMD.C3= 000010 G	C#MSG = 000023	ENDSP1 030430	G#DELM= 000372
BIT15 = 100000 G	CMD.C4= 000020 G	C#OPEN= 000034	ENDSP2 030432	G#DISP= 000003
BIT2 = 000004 G	CMD2M 031207	C#PNTB= 000014	ENDSP3 030624	G#EXCP= 000400
BIT3 = 000010 G	CMD3M 031356	C#PNTF= 000017	ENDSP4 030664	G#HILI= 000002
BIT4 = 000020 G	CMD4M 031364	C#PNTS= 000016	ENDSP5 030662	G#LOLI= 000001
BIT5 = 000040 G	CMD5M 031372	C#PNTX= 000015	EOTFLG 003502 G	G#NO = 000000
BIT6 = 000100 G	CMD6M 031400	C#QIO = 000377	ERCVER 002207 G	G#OFFS= 000400
BIT7 = 000200 G	CMD7M 031406	C#RDBU= 000007	ERLOG 003466 G	G#OF SI= 000376
BIT8 = 000400 G	CMD8M 031414	C#REFG= 000047	ERRREC 003471 G	G#PRMA= 000001
BIT9 = 001000 G	CNTBGN= 002626	C#RESE= 000033	ERS = 100411 G	G#PRMD= 000002
BOE = 000400 G	CNTEND= 003376	C#REVI= 000003	ERSFLG 003525 G	G#PRML= 000000
BORERS 015316 G	CNTLEN= 000550 G	C#RFLA= 000021	EVL = 000004 G	G#RADA= 000140
BPCRM 031215	CODELM 004162 G	C#RPT = 000025	EXALL 010226 G	G#RADB= 000000
BRCPK 002344 G	CP.ADH= 000004 G	C#SEFG= 000046	EXARTN 010522	G#RADD= 000040
BRFCNT 003416 G	CP.ADL= 000002 G	C#SPRI= 000041	EXCRTN 012362	G#RADL= 000120
BRF.C = 004000 G	CP.CMD= 000000 G	C#SVEC= 000037	EXCUTE 012054 G	G#RADO= 000020

PARAMETER CODING
SYMBOL TABLE

MACRO M1113 14 JUN-84 18:32

SEQ 0141

G\$XFER=	000004	L\$CCP	002106	G	L10012	023312	NSSRM	004536	G	RAWTM	026370	G
G\$YES =	000010	L\$CLEA	023736	G	L10013	023602	NUMBM	031227		RCVERM	031010	
HAE	002206	L\$CO	002032	G	L10014	023776	NURTY1	005422	G	RDBF	031310	
HAEM	030734	L\$DEPO	002011	G	L10015	024050	OFLINM	005456	G	RDBUF	002314	G
HALTM	004306	L\$DESC	002136	G	L10016	024154	ONEFIL =	000001		RDF	= 104001	G
HELP	= 000000	L\$DESP	002076	G	L10017	025642	OPFLAG	003536	G	RDR	= 104401	G
HOE	= 100000	L\$DEVP	002060	G	L10020	024370	OPP.C =	020000	G	RECCNT	003376	G
HRDCNT	003356	L\$DISP	002124	G	L10021	024414	0\$APTS=	000000		RECLOG	003465	G
HSSW	002320	L\$DLY	002116	G	L10022	024434	0\$AU =	000001		RECRED	007066	
IBE	= 010000	L\$DTP	002040	G	L10023	024454	0\$BGNR=	000001		RECTAP	010064	G
IDU	= 000040	L\$DTYP	002034	G	L10024	024474	0\$BGNS=	000001		RECUD	013020	G
IER	= 020000	L\$DU	024000	G	L10025	024514	0\$DU =	000001		TERM	005017	G
IE.C =	000200	L\$DUT	002072	G	L10026	024534	0\$ERRT=	000000		RETRYC	003460	G
INIT10	021324	L\$DVTY	002164	G	L10027	024554	0\$GNSW=	000001		REWRT	015472	G
INIT15	021622	L\$EF	002052	G	L10030	024574	0\$POIN=	000001		RFBC	002726	G
INIT16	021642	L\$ENVI	002044	G	L10031	024614	0\$SETU=	000000		RFCERM	004521	G
INTFLG	003472	L\$ETP	002102	G	L10032	024652	PASCNT	003326	G	RFREC	003026	G
INTPRI=	000340	L\$EXP1	002046	G	L10033	025036	PATCH	031422	G	RFUNR	003036	G
INVRT	007724	L\$EXP4	002064	G	L10034	026476	PATERN	003446	G	RLEXM	004556	G
IRE	003521	L\$EXP5	002066	G	L10035	026672	PATRO	011642	G	RNF	= 125401	G
IREC	002213	L\$HARD	027754	G	L10036	027052	PATR1	011700	G	RNOPSC=	177740	G
IRECH	031064	L\$HIME	002120	G	L10037	027750	PATR2	011720	G	RNR	= 105401	G
IREM	031131	L\$HPCP	002016	G	L10040	030060	PATR3	011730	G	RNYM	004757	G
ISR	= 000100	L\$HPTP	002022	G	L10041	030666	PATR4	011754	G	RPF	= 105001	G
IXE	= 004000	L\$HW	002174	G	L10042	032010	PATR5	011766	G	RPR	= 125001	G
I\$AU =	000041	L\$ICP	002104	G	L10044	032016	PATR6	012000	G	RPTCNT	003462	G
I\$AUTO=	000041	L\$INIT	021324	G	MBR	= 100012	PATR7	012020	G	RPTFLG	003517	G
I\$CLN =	000041	L\$LADP	002026	G	MDSET	007466	PATR8	012052	G	RPT1A	020432	
I\$DU =	000041	L\$LAST	032004	G	MEMOM	023210	PATTBL	011620		RPT1B	020507	
I\$HRD =	000041	L\$LOAD	002100	G	MISCFG	003531	PATTH	031247		RPT1C	020560	
I\$INIT=	000041	L\$LUN	002074	G	MOD.CO=	000400	PCMDWD	003424	G	RPT1D	020631	
I\$MOD =	000041	L\$MREV	002050	G	MOD.C1=	001000	PIRE	002216	G	RPT1E	021057	
I\$MSG =	000041	L\$NAME	002000	G	MOD.C2=	002000	PNT	= 001000	G	RPT1F	020735	
I\$PROT=	000040	L\$PRIO	002042	G	MOD.C3=	004000	PRI	= 002000	G	RPT1G	021006	
I\$PTAB=	000041	L\$PROT	021316	G	MOVMSG	012734	PRI00	= 000000	G	RPT1H	021203	
I\$PWR =	000041	L\$PRT	002112	G	MSGCNT=	000020	PRI01	= 000040	G	RPT1J	021107	
I\$RPT =	000041	L\$REPP	002062	G	MSGPKA	002544	PRI02	= 000100	G	RPT1K	021174	
I\$SEG =	000041	L\$REV	002010	G	MSGPKT	002354	PRI03	= 000140	G	RRANV	002205	G
I\$SETU=	000041	L\$RPT	017570	G	MSGPK0	002374	PRI04	= 000200	G	RRBC	002666	G
I\$SFT =	000041	L\$SOFT	030062	G	MSGPK1	002414	PRI05	= 000240	G	RRECL =	000020	G
I\$SRV =	000041	L\$SPC	002056	G	MSGPK2	002434	PRI06	= 000300	G	RRREC	003006	G
I\$SUB =	000041	L\$SPCP	002020	G	MSGPK3	002454	PRI07	= 000340	G	RRUNR	003016	G
I\$TST =	000041	L\$SPTP	002024	G	MS.RFC=	000004	PRXST	017412	G	RRVM	030705	
JLOC	003444	L\$STA	002030	G	MS.XS0=	000006	PTCHDS	027612	G	RTLE	014510	G
JLOOP	003442	L\$SW	002204	G	MS.XS1=	000010	PWRFLG	003527	G	RTLRTN	014634	
JMP	= 000040	L\$TEST	002114	G	MS.XS2=	000012	RANB	003432	G	RWCPK	002350	G
JMP.C =	000040	L\$TIML	002014	G	MS.XS3=	000014	RANBC	= 153624	G	RWD	= 102010	G
J\$JMP =	000167	L\$UNIT	002012	G	MS.XS4=	000016	RANCMD	026160		RWERR	003467	G
LENMSK	003430	L10000	002202		NCMD.C=	177740	RANDOM	003515	G	RSSAVE	003452	G
LOE	= 040000	L10001	002326		NCNT	003412	RANP	= 000007	G	SCCNT	003336	G
LOG	015566	L10002	006116		NCNT1	003414	RANRD	026220	G	SCERM	004475	G
LOT	= 000010	L10003	007070		NEXTSP	030130	RANS	003434	G	SCH	= 140004	G
L\$ACP	002110	L10004	010040		NEXTU	017150	RANSC	= 032561	G	SCHBK	002474	G
L\$APT	002036	L10005	010046		NINUSE=	177774	RANW	026444	G	SCHCNT=	000012	G
L\$AU	024052	L10006	010054		NGDEV	005543	RANWR	026332	G	SEQEND	003740	G
L\$AUT	002070	L10007	010062		NOINTM	004670	RANWV	026356	G	SETCH	010162	G
L\$AUTO	023314	L10010	021314		NRDYM	023700	RASFR	026416	G	SETDEF	007766	G

PARAMETER CODING
SYMBOL TABLE

MACRO M1113 14 JUN-84 18:32

SEQ 0142

SETRW	010206	G	TRAP4	023730	G	T\$LAST=	000001	T1	024156	G	WRTY	014636	G
SETUP	011172	G	TSAM	004705	G	T\$LOLI=	000000	T1SWB	003523	G	WRTYCT	003316	G
SFF	= 105010	G	TSBA	= 002514	G	T\$LSYM=	010000	T1.1	024166		WRTYER	003464	G
SFPTBL	002204	G	TSC.FC=	177717	G	T\$LTNO=	000005	T1.10	024576		WRTYFG	003463	G
SFR	= 105410	G	TSC.TC=	177761	G	T\$NEST=	177777	T1.11	024616		WRUNR	002776	G
SOFINI	007072	G	TSDB	002514	G	T\$NS0 =	000000	T1.12	024702		WSM	= 140006	G
SRF	= 104010	G	TSMD	031257		T\$NS1 =	000005	T1.2	024372		WSMBK	002506	G
SRR	= 104410	G	TSSP	003534	G	T\$NS2 =	000002	T1.3	024416		WSSR	012700	G
STAERM	006120	G	TSSR	002524	G	T\$PCNT=	000000	T1.4	024436		WTBF	031327	
STAER1	006436		TSSREG	003454	G	T\$PTAB=	010043	T1.5	024456		WTBUF	002316	G
STAER2	006616		TSUNT	003532	G	T\$PTHV=	000001	T1.6	024476		WTM	= 100011	G
STAER3	006675		TSVCT	002534	G	T\$PTNU=	000001	T1.7	024516		WTMFLG	003456	G
STAER4	006733		TS.A16=	000400	G	T\$SAVL=	177777	T1.8	024536		WTR	= 101011	G
STAER5	006753		TS.A17=	001000	G	T\$SEGL=	177777	T1.9	024556		WTV	= 104105	G
STAER6	006562		TS.NBA=	002000	G	T\$SIZE=	000005	T2	025644	G	WTVRM	004430	G
STAER7	006530		TS.NXM=	004000	G	T\$SUBN=	000000	T3	026500	G	WTYBRF	015110	
STAFLG	003526	G	TS.OFL=	000100	G	T\$TAGL=	177777	T4	026674	G	WTYCMD	015104	
SVCGBL=	000000		TS.RMR=	010000	G	T\$TAGN=	010045	T5	027054	G	WTYWRD	015106	
SVCINS=	000001		TS.SC =	100000	G	T\$TEMP=	000000	T5WEOT	027634	G	X\$ALWA=	000000	
SVCSUB=	000000		TS.SPE=	020000	G	T\$TEST=	000005	UAM	= 000200	G	X\$FALS=	000040	
SVCTAG=	000000		TS.SSR=	000200	G	T\$TSTM=	177777	UNIWLK	005653		X\$OFFS=	000400	
SVCTST=	000000		TS.UPE=	040000	G	T\$TSTS=	000001	UNL	= 100412	G	X\$TRUE=	000020	
SWBFLG	003520	G	TS1MD	002312	G	T\$AU =	010016	UNREC	003470	G	X0.BOT=	000002	G
SWB.C =	010000	G	TS5ADR	030012		T\$AUT=	010013	URERM	005041	G	X0.EOT=	000001	G
SWSET	004231	G	TS5CL	002564	G	T\$CLE=	010014	VFEXC	016152	G	X0.LET=	020000	G
S\$LSYM=	010000		TS5INT	002554	G	T\$DAT=	010044	VFISU	016400	G	X0.ONL=	000100	G
TCCRA	013414		TS5INO	010034	G	T\$DU =	010015	VFYCNT	003346	G	X0.RLL=	010000	G
TCC0	013434	G	TS5IN1	010042	G	T\$HAR=	010040	VFYDAT	016066	G	X0.RLS=	040000	G
TCC1	013452	G	TS5IN2	010050	G	T\$HW =	010000	VFYFLG	003516	G	X0.TMK=	100000	G
TCC2	013470	G	TS5IN3	010056	G	T\$INI=	010012	VFY.C =	000100	G	X0.WLK=	000004	G
TCC3	013600	G	TS5SW	002574	G	T\$MSG=	010003	WAITF	007204	G	X2.BFE=	000100	G
TCC4	013616	G	TS5UNT	030036		T\$PC =	000001	WLKCHK	007356	G	X2.EFE=	000200	G
TCC5	014232	G	T\$ARGC=	000003		T\$PRO=	010011	WLKZRO	011734		X2.OPM=	100000	G
TCC6	014330	G	T\$CODE=	001004		T\$PTA=	010043	WRBC	002626	G	X3.DCK=	000010	G
TCC7	014472	G	T\$ERRN=	000002		T\$RPT=	010010	WRECL =	000020	G	X3.RNY=	157400	G
TC2RTN	013576		T\$EXCP=	000000		T\$SOF=	010041	WRR	= 105005	G	X4.HSS=	100000	G
TIME1	003436	G	T\$FLAG=	000041		T\$SRV=	010007	WRREC	002766	G	X4.RCE=	040000	G
TIME2	003440	G	T\$FREE=	032016		T\$SUB=	010033	WRT	= 104005	G	ZROPAT	011704	
TOERM	004453	G	T\$GMAN=	000000		T\$SW =	010001	WRTCHK	007270	G	\$LSTIN=	000001	
TOOMM	004727	G	T\$HILI=	000010		T\$TES=	010037	WRTCHR	007436	G	\$LSTTA=	000001	
TRAPD4	003530	G											

. ABS. 032016 000
000000 001
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 26207 WORDS (103 PAGES)
DYNAMIC MEMORY: 20614 WORDS (79 PAGES)
ELAPSED TIME: 00:17:01
CVTSEC,CVTSEC/-SP-SVC/ML,TSV1E,CVTSEC.SRC