

TM02 TU16/TE16

BASIC FUNCTION TEST
CZTUBG0

AH-9452G-MC

COPYRIGHT © 74-77

FICHE 1 OF 1

JAN 1978

digital

MADE IN USA

REM 8

IDENTIFICATION

PRODUCT CODE: AC-9451G-MC
PRODUCT TITLE CZTUBGO TMO2 - TU16/TE16 BSC FC
DATE CREATED 15 AUGUST 1977
REVISED 11 NOV 1977 BY CLEM WALSH
MAINTAINER DIAGNOSTIC ENGINEERING

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

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A
LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE
TERMS OF SUCH LICENSE

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

COPYRIGHT (C) 1974-1977 BY DIGITAL EQUIPMENT CORPORATION

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

TABLE OF CONTENTS

PARAGRAPH	SUBJECT	PAGE
1.	ABSTRACT	1
2	REQUIREMENTS	1
3	LOADING PROCEEDURE	1
4	STARTING PROCEEDURE	1
5	SWITCH SETTINGS	2
6	ERROR PRINTOUTS	3
7	OPERATION	4
8	TEST DESCRIPTION	5
9	LISTING	

34
35
36
37
38
39
40
41
42
43
44
45
46

(PAGE 1)

1 ABSTRACT

THIS PROGRAM IS INTENDED TO TEST ALL OF THE BASIC
FUNCTIONAL LEVEL OPERATIONS OF THE TMO2-TU16/TE16 MAG
TAPE SYSTEM. ALL FUNCTIONS, WRITE, READ, SPACE,
ERASE, REWIND, ETC; WILL BE TESTED. IN ADDITION TO THE
TMO2-TU16/TE16 TESTS, THE RH WILL BE TESTED SEPARATELY IN
SO FAR AS IT IS POSSIBLE TO SEPARATE THE RH FROM THE
TMO2-TU16/TE16 ITSELF

2 REQUIREMENTS (HARDWARE)

- A ANY PDP-11 PROCESSOR - WITH OR WITHOUT A HARDWARE SWITCH REGISTER
- B 8K OF CORE
- C CONSOLE TTY
- D TMO2 MAGTAPE CONTROLLER
- E MASS BUS CONTROLLER
- F TU16 OR TE16 MAG TAPE TRANSPORT

3 LOADING PROCEEDURE

USE STANDARD BINARY LOADING PROCEEDURE

4 STARTING PROCEEDURE

****SOFTWARE SWITCH REGISTER IS LOCATED AT LOC 176(REFER TO SECTION 5 FOR MORE
THERE ARE TWO (2) STARTING ADDRESSES THAT MAYBE USED 200(8) AND 210(8)

A 200(8) STARTING AT THIS ADDRESS WILL CAUSE THE PROGRAM
IDENTIFICATION TO BE PRINTED FOLLOWED BY
REQUESTS FOR THE VARIOUS PARAMETERS NEEDED
BY THE PROGRAM

B 210(8) THIS ADDRESS IS INTENDED FOR USE AS A RESTART ONLY
AND WILL USE THE CURRENT PARAMETER VALUES

91 4 1 SAMPLE START AT 200(8). OPERATOR RESPONSES ARE IN PARENS.
92
93 ***SWR=XXXXXX NEW= WILL BE PRINTED FIRST IF SOFTWARE SWITCH REGISTER IS SELEC
94 (REFER TO SECTION 5 FOR OPERATOR ACTION)
95 TMO2-TU16/TE16 BASIC FUNCTION TEST
96 ENTER CONDITIONS IN OCTAL
97
98 REGISTER START: 172440 (CR)
99 VECTOR: 224 (CR)
100 DRIVE NUMBER: 0 (3)
101 SLAVE NUMBER 0 (6) SERIAL NO 200
102 RH11 OR RH70 (0)
103 RH ONLY (0)
104 NRZ ONLY (1) -NRZ (NON-RETURN-TO ZERO) IS THE METHOD OF RECORDING
105 ON MAGNETIC TAPE
106
107 THIS EXAMPLE SLOWS THE PROGRAM START USING THE RH11
108 ADDRESS (CS1) OF 172440, AN INTERRUPT VECTOR OF 224,
109 DRIVE NUMBER 3, AND SLAVE NUMBER 6, NRZ ONLY
110 NOTE THAT THE CURRENT VALUES FOR EACH PARAMETER IS
111 PRINTED AND MAY OR NOT BE CHANGED
112
113 ***IF THE SOFTWARE SWITCH REGISTER IS SELECTED THE FIRST TYPE OUT WILL BE
114 AS FOLLOWS SWR=XXXXXX NEW=
115 THIS WILL BE TYPED OUT BEFORE THE HEADER MESSAGE (REFER TO SECTION 5 FOR
116 A MORE DETAIL DESCRIPTION FOR OPERATOR ACTION)
117

(PAGE 2)

5 CONSOLE SWITCH SETTING

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED

CONTROL

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC 176) FROM THE TTY THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G < G>, THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC 176 AT SELECTED POINTS WITHIN THE PROGRAM, I. E. SCOPE ROUTINE AND AFTER
- 2) THE MACHINE WILL THEN TYPE SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER)
- 3) AFTER THE ''NEW='' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
 - A) TYPE A NUMBER TO BE LOADED INTO LOC 176 FOLLOWED BY A <CR> (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED
 - B) IF A CONTROL U < U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2

156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182

ALL SWITCHES EXCEPT 5-9 ARE USED AND THE NORMAL, OR DEFAULT,
RUN IS DONE WITH ALL SWITCHES SET TO ZERO (0)
ALL HARDWARE SWITCHES ARE DYNAMIC, AND MAY BE CHANGED AT ANY TIME
***BUT, THE SOFTWARE SWITCH REGISTER CAN ONLY BE LOADED DYNAMICALLY
AS STATED ABOVE UNDER CONTROL HEADING

SW15(100000).	1=HALT ON ERROR 0=CONTINUE
SW14(040000).	1=LOOP ON ERROR (SCOPE RH TESTS ONLY) 0=CONTINUE
SW13(020000).	1=DO NOT PRINT ERRORS 0=PRINT ALL ERRORS
SW12(010000)	1=INHIBIT ITERATION 0=DO ALL ITERATIONS PER TEST
SW11(004000)	1=CONTINUOUS CYCLE 0=HALT AT END OF PASS
SW10(002000).	1=HALT AT END OF CURRENT TEST 0=CONTINUE
SW9-5	N/A
SW4-0	SELECT TEST NUMBER 00=ALL TESTS

THE USE OF SW4-0 IS TO ALLOW SELECTION AND CONTINUOUS
EXECUTION OF ANY TEST. THE TEST SELECTION MAY BE CHANGED AT
ANY TIME, HOWEVER IT IS ADVISABLE TO USE SW10 TO STOP THE
PROGRAM AT THE END OF THE CURRENT TEST BEFORE CHANGING NUMBER.

(PAGE 3)

6 ERROR PRINTOUTS

THE ERROR PRINTOUTS FOR EACH TEST WILL APPEAR IN THE SAME
GENERAL FORMAT. THE FIRST LINE WILL ALWAYS SHOW THE TEST NUMBER AND ITS
TITLE THE SECOND LINE WILL BE AN EXPLANATION OF THE ERROR THE
FOLLOWING LINES WILL SHOW THE APPROPRIATE REGISTER OR ADDRESS
VALUES THAT ARE APPLICABLE TO THE INDIVIDUAL TEST

EXAMPLES

- 1 THIS EXAMPLE SHOWS A TYPICAL ERROR PRINTOUT FOR THE WR TE
READ TEST A WRITE CRC ERROR OCCURRED ON SLAVE 6

FT13 WRITE-READ TEST
WRITE ERROR NRZ

CS1	WC	BA	FC	CS2	DS	ER	TC
144260	000000	015650	000000	000103	150600	100000	101306

- 2 THIS EXAMPLE SHOWS A TYPICAL SPACE ERROR
THE FC IS NOT ZERO AT THE END OF THE OPERATION

FT14 SPACE TEST
SPACE REVERSE ERROR NRZ

CS1	WC	BA	FC	CS2	DS	ER	TC
144230	177700	017162	177740	000114	150600	001000	161700

- 3 THIS EXAMPLE SHOWS A SPACE OPERATION WHICH RESULTED IN
INCORRECT POSITIONING SHOULD BE AT RECORD 20, IS AT RECORD 22

FT14 SPACE TEST
POSITION ERROR
REVERSE ERROR EXPT 20 RCVD 22

(PAGE 4)

7 OPERATION

THE PROCEEDURES FOR OPERATING THIS PROGRAM ARE QUITE SIMPLE
AND REQUIRE ONLY A FEW STEP

- 1 LOAD ADDRESS 200 OR 210
- 2 SET SWITCHES FOR DESIRED TEST CYCLE
****REFER TO SECTION 5 FOR DYNAMIC LOADING
OF SOFTWARE SWITCH REGISTER ***
- 3 PRESS START
- 4 ENTER APPROPRIATE RESPONES TO THE TTY REQUESTS

ALL HARDWARE SWITCHES ARE DYNAMIC AND MAY BE CHANGED AT ANY
TIME THE NORMAL, OR DEFAULT, OPERATING SEQUENCE IS ALL
SWITCHES DOWN (ZERO) THE END OF EACH PASS IS NOTED BY A
MESSAGE STATING END OF PASS AND THE NUMBER OF THAT PASS
*****FOR THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER REFER TO SECTION 5

SINGLE TEST SELECTION (SWO-SW4)

WHEN SWO-4 ARE SET TO ZERO (00) THE SCHEDULAR WILL
EXECUTE ALL OF THE TESTS IN SEQUENCE (1-24) IF SWO-4 IS
SET TO SOME SPECIFIC TEST NUMBER (1-24) THAT PARTICULAR TEST
WILL BE EXECUTED CONTINUOUSLY ANY TEST MAY BE SINGLE
SELECTED IN ANY ORDER, HOWEVER, THE BEST WAY TO AFFECT
THE CHANGE IS TO USE SW10 TO HALT THE CURRENT TEST, THEN
CHANGE NUMBER AND PRESS CONTINUE

RH11 OR RH70 OPTION

A ONE RESPONSE IS FOR THE RH70,
A ZERO RESPONSE IS FOR THE RH11

RH ONLY OPTION

BY RESPONDING TO THE REQUEST (RH ONLY) WITH
A ONE (1), ONLY THE TESTS WHICH ARE POINTED TO
THE RH (TESTS 1 - 10) W LL BE EXECUTED IN
EACH PASS

(PAGE 5)

8 TEST DESCRIPTION

THE FOLLOWING IS A LIST OF ALL TESTS IN THEIR PROPER SEQUENCE
A BASIC DESCRIPTION OF EACH TEST IS PROVIDED TO AID IN UNDERSTANDING
OF THE ERROR MESSAGES ASSOCIATED WITH EACH ONE

A RH TESTS: THE FIRST TEN (10) TESTS WILL PERFORM BASIC RH
OPERATIONS AS FAR AS IS POSSIBLE WITHOUT REQUIRING
THE TMO2-TU16/TE16 ITSELF. (SEE RH ONLY OPTION, PAR 7)

FT1 RH ADDRESSING: THIS TEST WILL ASSURE THAT THE
RH WILL RESPOND WITHOUT CAUSING A BUS
TRAP TO ALL TMO2 REGISTER ADDRESS
IN SEQUENCE STARTING AT THE ADDRESS
OF CS1 ENTERED BY THE OPERATOR.

FT2 RH REGISTER BITS READ/WRITE THIS TEST WILL ASSURE THAT
ALL BITS OF THE RH WRITE/READ REGISTERS
CAN BE SET AND RESET

FT3 RH INITIALIZE THIS TEST WILL ASSURE THAT A RH INITIALIZE
(BIT 5 OF CS2=1) WILL INDEED CLEAR
THE RH ERRORS

* FT4 SILO TEST 1 THIS TEST WILL ASSURE THAT A READ FROM
AN EMPTY SILO WILL CAUSE DLT TO SET

* FT5 SILO TEST 2 THIS TEST WILL ASSURE THAT BOTH THE
IR AND OR BITS WILL CORRECTLY RESPOND
TO LOADING OF THE SILO WITH ALL ZEROS
AND THEN A WORD OF ALL ONES

* FT6 SILO TEST 3 THIS TEST WILL WRITE AND THEN READ
THE ENTIRE SILO TO ASSURE THAT DATA CAN
BE PROPERLY FILLED AND READ ALSO THE
PROPER STATUS OF IR AND OR ARE CHECKED

* FT7 SILO TEST 4 THIS TEST WILL ASSURE PROPER RH11
RESPONSE TO SILO OVERFLOW

* FT10 SILO TEST 5 THIS TEST WILL ASSURE SILO RESET
BY RH11 INITIALIZE

*** NOTE SILO TESTS (FT4-FT10) ARE FOR THE RH11 ONLY ***

(PAGE 6)

B. TMO2-TU16/TE16 BASIC FUNCTIONS: THE FOLLOWING FOURTEEN (14) TESTS WILL ASSURE OPERATION OF THE MAG TAPE BASIC FUNCTIONS.

FT11 NOP TEST. THIS TEST WILL ASSURE THAT THE NOP FUNCTION EXECUTES WITH NO ERROR

FT12 REWIND TEST. THIS TEST WILL ASSURE THAT THE REWIND FUNCTION WILL POSITION THE TAPE TO BOT WITH NO ERROR.

- 1 ISSUE A REWIND COMMAND
- 2 AWAIT PIP RESET (MOTION STOPPED)
3. ASSURE THAT NO ERROR OCCURED
- 4 END

FT13 WRITE/READ TEST THIS TEST WILL ASSURE THAT THE UNIT UNDER TEST CAN WRITE AND READ IN ALL DENSITIES (FOR BOTH PE AND NRZ)

- 1 REWIND TO BOT
- 2 WRITE 100 RECORDS
 - A, ALL ONES DATA
 - B, 200 FRAMES
 - C, 200 BPI, ODD
- 3 CHECK FOR ERRORS ON EACH RECORD
- 4 READ REVERSE THEN FORWARD ALL 100 RECORDS
- 5 CHECK FOR ERRORS ON EACH RECORD
- 6 REPEAT STEPS 2 THRU 5 FOR 556, 800, 1600 BPI
- 7 END

DATA RELATED ERRORS (PARITY ERROR, CRC ERROR, ETC) ARE IGNORED T
DATA READ IS NOT CHECKED, ONLY THE FUNCTION IS TESTED, NOT THE M

FT14 SPACE TEST THIS TEST WILL ASSURE THAT PROPER POSITIONING IS MAINTAINED BY BOTH SPACE FORWARD AND REVERSE

- 1 REWIND TO BOT
- 2 WRITE 100 RECORDS
 - A EACH RECORD IS ONE FRAME LARGER THAN THE LAST
- 3 EACH RECORD IS ERROR CHECKED
- 4 DATA RELATED ERRORS ARE IGNORED
- 5 NOW SPACE REVERSE 77 RECORDS AND READ REVERSE 1, THE FRAME COUNT SHOULD BE 100
- 6 THIS IS THE SIZE OF THE FIRST RECORD
- 7 NOW SPACE FORWARD 76 RECORDS AND READ FORWARD 1, THE FRAME COUNT SHOULD BE 177
- 8 THIS IS THE SIZE OF THE NEXT TO LAST RECORD
- 9 CONTINUE THE SPACE AND READ (DECREMENTING THE RECORD UNTIL ALL POSITIONS HAVE BEEN CHECKED IF POSITION IS
- 10 REPEAT STEPS 1 THRU 7 FOR PE

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12 40^{L 1} PAGE 11
CZTUBG P11 05-DEC-77 12 39

SEQ 0011

365
366
367

9 END

(PAGE 7)

368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423

FT15. ERASE TEST.

THIS TEST WILL ASSURE THAT THE ERASE
FUNCTION WILL INDEED ERASE TAPES

1. REWIND TO BOT
2. ISSUE 200 ERRASE COMMANDS.
3. ASSURE NO ERRORS FOR EACH COMMAND
4. REWIND TO BOT.
5. ISSUE A READ FORWARD COMMAND.
6. THE TAPE SHOULD MOVE FORWARD UNTIL
STOPPED BY OPI (APPROX 25 FT)
7. ASSURE NO ERRORS OTHER THAN OPI.
8. END

FT16 TAPE MARK WRITE/READ THIS TEST WILL ASSURE THAT
A TAPE MARK CAN BE WRITTEN AND READ
IN BOTH PE AND NRZ

1. REWIND TO BOT
2. ISSUE A WRITE TAPE MARK COMMAND
3. ASSURE NO ERRORS
4. ASSURE THAT TAPE MARK STATUS IS SET
IN DRIVE STATUS (BIT 2)
5. READ REVERSE.
6. ASSURE THAT TAPE MARK IS SET
7. ASSURE THAT NO ERRORS OTHER THAN FCE OCCURED.
8. READ FORWARD
9. REPEAT STEPS 6 AND 7
10. REPEAT STEPS 1 THRU 9 FOR PE
11. END

FT17 TAPE MARK SPACE TEST THIS TEST WILL ASSURE THAT
SPACING WILL BE TERMINATED BY RECOGNITION
OF TAPE MARK BOTH IN PE AND NRZ

1. REWIND TO BOT
2. WRITE THE FOLLOWING PATTERN OF
TAPE MARKS AND DATA RECORDS

TM 20 RECS TM 40 RECS TM 60 RECS TM 100 RECS TM

3. ASSURE NO ERRORS
4. ASSURE THAT TAPE MARK STATUS IS SET FOR TM WRITES
5. NOW SPACE REVERSE 200 RECORDS
6. THE SPACE OPERATION SHOULD STOP ON EACH
TAPE MARK IT FINDS THEREFOR 5 SPACE
COMMANDS ARE ISSUED TO COVER THE ENTIRE
PATTERN WRITTEN ON TAPE
BOT SHOULD NEVER BE REACHED AND THE
FRAME COUNT WILL REFELCT
THE NUMBER OF RECORDS BETWEEN
TAPE MARKS
7. REPEAT STEP 6 IN THE FORWARD DIRECTION
8. ASSURE NO ERRORS OTHER THAN FCE
9. REPEAT STEPS 1 THRU 8 FOR PE

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12 40^{N 1} PAGE 13
CZTUBG P11 05-DEC-77 12 39

SEQ 0013

424
425

10 END

426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481

(PAGE 8)

FT20 WRITE CHECK TEST: BOTH WRITE CHECK FORWARD AND REVERSE ARE
TESTED IN BOTH PE AND NRZ

1. REWIND TO BOT
2. WRITE A 400 FRAME RECORD USING DATA
PATTERN 3 (125125).
3. ASSURE NO ERRORS OCCURED
4. ISSUE A REVERSE WRITE CHECK COMMAND.
5. ASSURE NO ERRORS OCCURED
6. REPEAT STEP 5 FOR A FORWARD WRITE CHECK.
7. REPEAT STEPS 1 THRU 6 FOR PE
8. END

FT21 ERASE HEAD TEST THIS TEST WILL ASSURE THAT THE
ERASE HEAD ITSELF IS OPERATING

1. REWIND TO BOT
2. WRITE 2 RECORDS OF 800(10) FRAMES
EACH EACH RECORD WILL BE 1 INCH
OF TAPE DATA IS NOT ALL ONES
3. REWIND TO BOT
4. NOW WRITE A 400(10) FRAME RECORD
THIS RECORD WILL BE ONE HALF
INCH OF TAPE THE ERASE HEAD
SHOULD CLEAR THE REMAINDER OF THE
FIRST RECORD (ONE HALF INCH)
5. REWIND TO BOT.
6. NOW READ THE SHORT FIRST RECORD
IT SHOULD BE 400(10) FRAMES
7. NOW READ THE SECOND RECORD
IT SHOULD BE STILL 800(10)
FRAMES
8. IF THE SECOND RECORD IS TOO LONG,
THE ERASE HEAD DID NOT FUNCTION OR
IT IS IN THE WRONG POLARITY
10. END

FT22 BUFFERED COMMAND THIS TEST WILL ASSURE THAT THE
TMO2 WILL ACCEPT AND EXECUTE ANOTHER
COMMAND WHILE ITS SELECTED SLAVE IS REWINDING

1. REWIND TO BOT
2. ISSUE 3 LONG WRITE COMMANDS TO ASSURE
BEING OFF BOT
3. ISSUE A REWIND COMMAND
4. AS SOON AS DRIVE READY BECOMES SET,
ISSUE ANOTHER WRITE COMMAND
5. THE NEXT DRIVE READY SHOULD BE AFTER THE
TAPE HAS REACHED BOT AND EXECUTED THE
BUFFERED WRITE COMMAND
6. ASSURE NO ERRORS OCCURED
7. END

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12 40^{C 2} PAGE 15
CZTUBG P11 05-DEC-77 12 39

SEQ 0015

482

483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513

(PAGE 9)

FT23 READ IN PRESET THIS TEST WILL ASSURE THAT UNIT 0
IS REWOUND AND SET TO 800 BPI NORMAL
(ONLY IF SLAVE 0 IS SELECTED).

1. ISSUE A WRITE COMMAND TO ASSURE
BEING OFF BOT
2. ISSUE THE READ-IN PRESET COMMAND
3. AWAIT MOTION STOP.
4. ASSURE THAT BOT WAS REACHED
5. ASSURE THAT THE TAPE CONTROL REGISTER
IS SET TO 800 BPI, NORMAL, ODD
6. END

(THIS TEST IS ONLY PERFORMED IF THE SELECTED SLAVE IS ZERO (0).

FT24 REWIND OFF LINE THIS TEST WILL ASSURE
THAT THE UNIT WILL REWIND AND
GO OFF LINE (NOT IF IN CONTINUOUS CYCLE)

1. ISSUE THE REWIND OFF-LINE COMMAND
2. ASSURE THAT MOL (BIT 12 OF DRIVE STATUS)
IS RESET INDICATING THE UNIT WENT OFF LINE
3. END

(THIS TEST IS NOT PERFORMED WHEN CONTINUOUS CYCLE OPERATION IS S

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12 40^{E 2} PAGE 17
CZTUBG P11 05-DEC-77 12 39

SEQ 0017

514
515

9

LISTING

```

590      ,REGISTER EQUIVS*****
591
592      000000      RO=%0
593      000001      R1=%1
594      000002      R2=%2
595      000003      R3=%3
596      000004      R4=%4
597      000005      R5=%5
598      000006      SP=%6
599      000007      PC=%7
600
601      ,TRAP CATCHERS*****
602
603      000000      =0
604      000200      REPT 200
605
606      +2
607      HALT
608      ENDR
609
610      ,*****
611      ,ACT11 HOOKS
612      ,*****
613
614      001000      $SVPC=
615
616      000040      000040      =40
617      000040      000      DRIVE BYTE 0 ,DRIVE # FOR XXDP LOAD MED UM
618
619      000041      000041      =41
620      000041      000      MEDIUM. BYTE 0 ,XXDP LOAD MEDIUM
621
622
623      000042      000042      =42
624      000042      000000      WORD 0 ,LOCATION INDICATOR - AUTOM/MAN MODE
625
626
627      000046      000046      =46
628      000046      003334      WORD SENDAD ,SET TO SENDAD IN SEOP
629
630
631      000052      000052      =52
632      000052      000000      WORD 0 ,CHARACTERISTICS OF PROGRAM
633
634      001000      =5SVPC ,RESTORE PC
635
636      ,*****
637
638
639      ,TTY INTERRUPT VECTOR*****
640
641      000060      000060      =60
642      000060      012570      TTINT ,TTY INTERRUPT HEADER ADDRESS
643      000062      000000      0
644
645      ,SOFTWARE SWITCH REGISTER*****

```

646					
647		000176		=176	
648	000176	000000		SWREG 0	, SOFTWARE SWITCH REGISTER
649					
650					
651					, *****
652					
653					, THIS PROGRAM SUPPORTS THE SOFTWARE SWITCH REGISTER LOC 176
654					, REFER TO SECTION 5 OF DOCUMENT FOR DESCRIPTION
655					
656					, *****
657					, START ADDRESS*****
658					
659		000200		=200	
660	000200	005000		CLR R0	
661	000202	000167	001372	JMP START	, PROGRAM START
662					
663		000210		=210	
664	000210	000240		NOP	
665	000212	012700	000C01	MOV #1, R0	, SET NO HEADER FLAG
666	000216	000167	001356	JMP START	
667					
668					, TMO2 INTERRUPT VECTOR*****
669					
670		000224		=224	
671	000224	012554		MTINT	, TAPE INTERRUPT HANDLER ADDRESS
672	000226	000240		340	
673					
674					
675		000600		=600	
676					, MASS BUS REGISTER EQUIVS*****
677					
678	000600	172440	C1	172440	
679	000602	172442	WC	172442	
680	000604	172444	BA	172444	
681	000606	172446	FC	172446	
682	000610	172450	CS	172450	
683	000612	172452	DS	172452	
684	000614	172454	EP	172454	
685	000616	172456	AS	172456	
686	000620	172460	CC	172460	
687	000622	172462	DB	172462	
688	000624	172464	MR	172464	
689	000626	172466	DT	172466	
690	000630	172470	SN	172470	
691	000632	172472	TC	172472	
692	000634	172474	BAE	172474	
693					
694					, CONSTANTS*****
695					
696	000636	177776	PSW	177776	, PROCESSOR STATUS
697	000640	177570	SWR	177570	, SWITCH REGISTER
698	000642	177560	TKS	177560	, TTY READER STATUS
699	000644	177562	TKB	177562	, TTY READ BUFFER
700	000646	177564	TPS	177564	, TTY PUNCH STATUS
701	000650	177566	TPB	177566	, TTY PUNCH BUFFER

702	000652	177777	SERNUM	177777	, SERIAL NUMBER
703	000654	000011	DRVTP	011	, DRIVE TYPE
704	000656	000010	ITAMT	10	, ITERATION AMOUNT
705	000660	000224	VECT	224	, INTERRUPT VECTOR(RH)
706	000662	172440	REGS	172440	, STARTING REGISTER ADDRESS
707	000664	000004	BTRP	4	, BUS TRAP ADDRESS
708	000666	000006	BTRP2	6	, BUS TRAP PRIORITY LEVEL 7

```
709
710 , *****
711 , ACT11 MODE INDICATORS
712 , *****
713
714 000670 000000 AUTOM WORD 0 ;AUTOMATIC MODE INDICATOR
715 000672 000 ACT11M BYTE 0 ;ACT11 AUTO MODE INDICATOR
716 000673 000 XXDPM BYTE 0 ;XXDP AUTO MODE INDICATOR
717 000674 000 ADUMPM BYTE 0 ;ACT11 DUMPM INDICATOR
718 000675 000 XDUMPM BYTE 0 ;XXDP DUMP MODE INDICATOR
719
720 , *****
```

721 , FLAGS AND COUNTERS*****

722			
723	000676	000000	TOB 0
724	000700	000000	TIB 0
725	000702	000000	RH17F 0
726	000704	000000	HDRFL 0
727	000706	000000	EMADDR 0
728	000710	000000	DRVN 0
729	000712	000000	SLVN 0
730	000714	000000	BADDR 0
731	000716	000000	FCNT 0
732	000720	000000	WCNT 0
733	000722	000000	RCNT 0
734	000724	000000	ERRP 0
735	000726	000000	ERRP1 0
736	000730	000000	RRO 0
737	000732	000000	RFD 0
738	000734	000000	RDYDX 0
739	000736	000000	OPDYX 0
740	000740	000000	SCNT 0
741	000742	000000	PFLG 0
742	000744	000000	RTRN 0
743	000746	000000	ERADD 0
744	000750	000000	TEMP1 0
745	000752	000000	TEMP2 0
746	000754	000000	TEMP3 0
747	000756	000000	STMSK 0
748	000760	000000	ITCNT 0
749	000762	000000	DSAV 0
750	000764	000000	SAV1 0
751	000766	000000	SAV2 0
752	000770	000000	SAV3 0
753	000772	000000	SCOLP 0
754	000774	000000	ITRLP 0
755	000776	000000	EXFL 0
756	001000	000000	PEXFL 0
757	001002	000000	STFLG 0
758	001004	000000	LTADD 0
759	001006	000000	FUN 0
760	001010	000000	SERFL 0
761	001012	000000	CRCNT 0
762	001014	000000	UDES 0
763	001016	000000	PATRN 0
764	001020	000000	RHTF 0
765	001022	000000	NRZOF 0
766	001024	000000	RHOF 0
767	001026	000000	PCNTR 0
768	001030	000000	TEMPST 0
769	001032	000000	COUNT 0
770	001034	000000	RDSW 0
771	001036	000000	PAFLG 0

772 , DATA PATTERN GENERATORS*****

773			
774			
775	001040	000000	DATBL 0
776	001042	012324	DATAC DAT1 , ALL ONE BITS

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12:40^{M 2} PAGE 25
CZTUBG P11 05-DEC-77 12 39

SEQ 0025

777	001044	012346	DATA1	DAT2	, ALL ZERO BITS
778	001046	012354	DATA2	DAT3	, ALTERNATING ONE/ZERO BITS
779	001050	012364	DATA3	DAT4	, ALL BITS 0-377

```
780
781
782          , LOGIC TEST ENTRY TABLE*****
783          TSTTBL 0
784          0
785          FT1
786          FT1
787          FT2
788          FT2
789          FT3
790          FT3
791          FT4
792          FT4
793          FT5
794          FT5
795          FT6
796          FT6
797          FT7
798          FT7
799          FT10
800          FT10
801          FT11
802          FT11
803          FT12
804          FT12
805          FT13
806          FT13
807          FT14
808          FT14
809          FT15
810          FT15
811          FT16
812          FT16
813          FT17
814          FT17
815          FT20
816          FT20
817          FT21
818          FT21
819          FT22
820          FT22
821          FT23
822          FT23
823          FT24
824          FT24
825          0
826          0
827          0
828          0
```

```

829          001600          . =1600
830                                     , PROGRAM START AND HOUSEKEEPING*****
831
832 001600 000240          START  NOP
833 001602 005067 177230      CLR      PAFLG          ; INIT PASS FLAG
834 001606 005067 177214      CLR      PCNTR         ; INIT PASS COUNTER
835 001612 012777 000340 177016 MOV      #340, @PSW      ; SET PRIORITY
836 001620 012706 000500      MOV      #500, SP        ; SET STACK POINTER
837
838                                     *****
839                                     DIAGNOSTIC SETUP FOR EXECUTION
840                                     UNDER ACT11.
841                                     *****
842
843 001624 004767 012434      JSR      PC, CKMODE      ; CHECK FOR MODE OF OPERATION
844 001630 005767 177034      TST      AUTOM          ; IS IT AUTOMATIC MODE
845 001634 001001          BNE      1$              ; BRANCH - IF YES
846 001636 000412          BR       SUSWR           ; CHECK SWR IN DUMPM - IF NOT
847 001640 032737 020000 000052 1$ BIT      #20000, @#52 ; SET UP FOR MANUAL INTERVENTION?
848 001646 001406          BEQ      SUSWR           ; BRANCH - IF NO
849 001650 012704 014420      MOV      #MSGC, R4      ; GET MESSAGE
850 001654 004767 011274      JSR      PC, TTOUT      ; TYPE MESSAGE
851 001660 000167 012500      JMP      ABORT          ; AND ABORT THE PROGRAM
852
853                                     *****
854
855 001664 013746 000006      SUSWR MOV      @#6, -(SP)    ; SAVE VECTORS
856 001670 013746 000004      MOV      @#4, -(SP)
857 001674 012737 001714 000004 MOV      #1$ , @#4      ; SET UP FOR TIMEOUT
858 001702 022777 177777 176730 CMP      #-1, @SWR    ; REFERENCE HARDWARE SWITCH REGISTER
859 001710 001402          BEQ      2$
860 001712 000404          BR       3$
861 001714 022626          CMP      (SP)+, (SP)+      ; ADJUST STACK
862 001716 012767 000176 176714 2$ MOV      #SWREG, SWR ; POINT TO SOFTWARE SWITCH REG
863 001724 012637 000004      MOV      (SP)+, @#4    ; RESTORE VECTORS
864 001730 012637 000006      MOV      (SP)+, @#6
865 001734 023727 000640 000176 CMP      @#SWR, #SWREG ; IS SOFTWARE REG USED
866 001742 001002          BNE      4$              ; BRANCH IF NO
867 001744 004767 012052      JSR      PC, CNTLU      ; ALLOW SOFTWARE SWITCH REGISTER TO BE CHANGED
868
869 001750          4$
870
871                                     *****
872                                     IF IN ACT11 MODE INHIBIT TYPING PROGRAM
873                                     IDENTIFICATION AND MANUAL INTERVENTION
874                                     *****
875
876 001750 005767 176716      TST      ACT11M          ; CHECK FOR ACT11 MODE
877 001754 001104          BNE      ST              ; BRANCH - IF ACT11
878                                     *****
879
880 001756 005700          TST      RO              ; SEE IF PRINT HEADER
881 001760 001402          BEQ      STOR            ; IF SO BR
882 001762 000167 000764      JMP      ST4          ; ELSE SKIP
883 001766 012704 014624      MOV      #MSG3, R4
884 001772 004767 011156      JSR      PC, TTOUT      ; PRINT TITLE
  
```



```

885 001776 012704 014726      STOB  MOV  #MSG4,R4
886 002002 004767 011146      JSR   PC,TTOUT      ;REQUEST REGISTER ADDRESS
887 002006 016703 176650      MOV   REGS,R3
888 002012 004767 011302      JSR   PC,OCTP       ;PRINT CURRENT ADDRESS
889 002016 012705 000662      MOV   #REGS,R5      ;SET ADDRESS SAVE LOC
890 002022 012701 000006      MOV   #6,R1         ;SET SIZE OF RESPONSE
891 002026 012702 176400      MOV   #176400,R2    ;SET UPPER LIMIT
892 002032 012703 172300      MOV   #172300,R3    ;SET LOWER LIMIT
893 002036 004767 010656      JSR   PC,TTR        ;GO GET RESPONSE
894 002042 012704 014751      MOV   #MSG5,R4
895 002046 004767 011102      JSR   PC,TTOUT      ;REQUEST VECTOR
896 002052 016703 176602      MOV   VECT,R3
897 002056 004767 011236      JSR   PC,OCTP       ;PRINT CURRENT VECTOR
898 002062 012705 000660      MOV   #VECT,R5      ;SET ADDRESS SAVE LOC
899 002066 012701 000003      MOV   #3,R1         ;SET SIZE OF RESPONSE
900 002072 012702 000224      MOV   #224,R2       ;SET UPPER LIMIT
901 002076 012703 000150      MOV   #150,R3      ;SET LOWER LIMIT
902 002102 004767 010612      JSR   PC,TTR        ;GO GET RESPONSE
903 002106 016700 176546      MOV   VECT,R0       ;GET VECTOR
904 002112 012720 012554      MOV   #MTINT,(R0)+  ;LOAD INTERRUPT ADDRESS IN VECTOR
905 002116 012710 000340      MOV   #340,(R0)    ;LOAD PRIORITY
906 002122 016700 176534      MOV   REGS,R0       ;GET START OF REGS
907 002126 012701 000016      MOV   #16,R1        ;SET NUMBER OF REGS
908 002132 012702 000600      MOV   #C1,R2        ;GET START OF TABLE
909 002136 010022 000002      STO   MOV   R0,(R2)+ ;BUILD TABLE
910 002140 062700 000002      ADD   #2,R0         ;BUMP ADDRESS
911 002144 005301              DEC   R1              ;SEE IF DONE
912 002146 001373              BNE   STO           ;IF NOT BR
913 002150 012702 000676      MOV   #TOB,R2
914 002154 012700 000054      MOV   #54,R0
915 002160 005022              ST1   CLR   (R2)+      ;CLEAR FLAGS + COUNTERS
916 002162 005300              DEC   R0
917 002164 001375              BNE   ST1
918 002166 012767 000001 176624 ST   MOV   #1,RHTF      ;SET ADDRESS TEST FLAG
919 002174 000167 000650      JMP   TSRH         ;GO DO INITIAL ADDRESS TEST PASS
920
921 002200              ST1A
922
923 002200 005767 176464      /      *****
924 002204 001017              TST   AUTOM      ;CHECK FOR AUTOMATIC MODE
925 002204 001017              BNE   15          ;BRANCH - IF YES
926 002204 001017              *****
927
928 002206 012704 015032      MOV   #MSG10,R4
929 002212 004767 010736      JSR   PC,TTOUT      ;REQUEST DRIVE NUMBER
930 002216 012705 000710      MOV   #DRVN,R5      ;SET ADDRESS OF DRIVE NUMBER SAVE
931 002222 012701 000001      MOV   #1,R1         ;SET SIZE OF RESPONSE
932 002226 012702 000007      MOV   #7,R2         ;SET UPPER LIMIT
933 002232 012703 000000      MOV   #0,R3         ;SET LOWER LIMIT
934 002236 004767 010456      JSR   PC,TTR        ;GO GET RESPONSE
935 002242 000434              BR    CONT1       ;EXIT
936
937 002244              *****
938 002244              AUTOMATICALLY SIZE FOR DRIVES
939 002244              *****
940 002244              15

```

```

941 002244 012767 177777 176436      MOV      #-1, DRVN      ; INIT DRIVE #
942 002252 012767 177777 176432  NXTDRV  MOV      #-1, SLVN      ; INIT SLAVE #
943 002260 012777 000040 176322  1$      MOV      #40, @CS      ; INIT CONTROLLER
944 002266 005267 176416              INC      DRVN      ; STEP DRIVE #
945 002272 022767 000010 176410      CMP      #10, DRVN      ; ALL DRIVES TESTED?
946 002300 001002              BNE      2$      ; BRANCH - IF NOT
947 002302 000167 000764              JMP      TEND0      ; EXIT
948 002306 016777 176376 176274  2$      MOV      DRVN, @CS      ; LOAD DRIVE #
949 002314 005777 176260              TST      @C1      ; ACCESS DRIVE
950 002320 032777 010000 176262      BIT      #10000, @CS      ; NON-EXISTANT DRIVE?
951 002326 001354              BNE      1$      ; BRANCH - IF YES (NED=1)
952 002330 000167 000106              JMP      NXTSLV      ; EXIT TO SIZE FOR SLAVES
953
954
955
956 002334 012777 000040 176246  CONT1  MOV      #40, @CS      ; SET INIT
957 002342 056777 176342 176240      BIS      DRVN, @CS      ; SET DRIVE NUMBER
958 002350 005777 176224              TST      @C1      ; ACCESS DRIVE
959 002354 032777 010000 176226      BIT      #10000, @CS      ; SEE IF NED
960 002362 001405              BEQ      ST2      ; IF NOT. BR
961 002364 012704 015764              MOV      #MSG41, R4
962 002370 004767 010560              JSR      PC, TTOUT      ; PRINT NOT AVAIL
963 002374 000701              BR       ST1A      ; REDO DRIVE REQUEST
964
965 002376              ST2
966
967
968 002376 005767 176266      TST      AUTOM      ; CHECK FOR AUTOMATIC MODE
969 002402 001017              BNE      1$      ; BRANCH - IF YES
970
971
972 002404 012704 015052      MOV      #MSG11, R4
973 002410 004767 010540              JSR      PC, TTOUT      ; REQUEST SLAVE NUMBER
974 002414 012705 000712      MOV      #SLVN, R5      ; SET ADDRESS OF SLAVE SAVE
975 002420 012701 000001      MOV      #1, R1      ; SET SIZE OF RESPONSE
976 002424 012702 000007      MOV      #7, R2      ; SET UPPER LIMIT
977 002430 012703 000000      MOV      #0, R3      ; SET LOWER LIMIT
978 002434 004767 010260              JSR      PC, TTR      ; GO GET RESPONSE
979 002440 000432              BR       CONT2      ; AND EXIT
980
981
982
983
984 002442              1$
985 002442 005267 176244      NXTSLV  INC      SLVN      ; STEP SLAVE #
986 002446 001010              BNE      1$      ; BRANCH - IF NOT SLAVE 0
987 002450 005767 176234              TST      DRVN      ; DRIVE 0?
988 002454 001005              BNE      1$      ; BRANCH - IF NOT
989 002456 105767 176211      TSTB     XXDPM      ; CHAIN MODE?
990 002462 001402              BEQ      1$      ; BRANCH - IF NOT
991 002464 005267 176222      INC      SLVN      ; STEP TO NEXT SLAVE
992 002470 022767 000010 176214  1$      CMP      #10, SLVN      ; ALL SLAVES TESTED?
993 002476 001665              BEQ      NXTDRV      ; BRANCH - IF YES
994 002500 016777 176206 176124      MOV      SLVN, @TC      ; LOAD SLAVE UNIT #
995 002506 032777 002000 176112      BIT      #2000, @DT      ; SLAVE PRESENT
996 002514 001752              BEQ      NXTSLV      ; BRANCH - IF NOT (SPR=C)

```



```

997 002516 032777 140000 176102      BIT    #140000, @DT      ; IS DRIVE A TAPE UNIT
998 002524 001746                      BEQ    NXTSLV        ; BRANCH - IF NOT
999
1000                                     *****
1001
1002 002526 012777 000040 176054 CONT2 MOV    #40, @CS        ; INIT
1003 002534 056777 176150 176046      BIS    DRVN, @CS      ; SET DRIVE NUMBER
1004 002542 016777 176144 176062      MOV    SLVN, @TC      ; LOAD SLAVE NUMBER
1005 002550 032777 002000 176050      BIT    #2000, @DT      ; SEE IF SLAVE PRESENT
1006 002556 001005                      BNE    ST3            ; IF SO BR
1007 002560 012704 016005                      MOV    #MSG42, R4
1008 002564 004767 010364                      JSR    PC, TTOUT      ; PRINT NON-EXIST SLAVE
1009 002570 000702                      BR     ST2            ; REDO SLAVE REQUEST
1010 002572 012704 016026      ST3  MOV    #MSG43, R4
1011 002576 004767 010352                      JSR    PC, TTOUT      ; PRINT SERIAL NUMBER TAG
1012 002602 017703 176022                      MOV    @SN, R3
1013 002606 004767 011034                      JSR    PC, SNPT      ; PRINT SERIAL NUMBER
1014
1015                                     *****
1016 002612 005767 176052      TST     AUTOM          ; CHECK FOR AUTOMATIC MODE
1017 002616 001057                      BNE    TSCD          ; BRANCH - IF YES
1018                                     *****
1019
1020 002620 012704 016610      MOV    #MSG61, R4
1021 002624 004767 010324                      JSR    PC, TTOUT      ; REQUEST RH11 OR RH70
1022 002630 012705 000702      MOV    #RH17F, R5      ; GET ADDRESS OF FLAG
1023 002634 012701 000001      MOV    #1, R1        ; SET SIZE OF RESPONSE
1024 002640 012702 000001      MOV    #1, R2        ; SET UPPER LIMIT
1025 002644 012703 000000      MOV    #0, R3        ; SET LOWER LIMIT
1026 002650 004767 010044                      JSR    PC, TTR        ; GET RESPONSE
1027 002654 012704 016630      MOV    #MSG62, R4
1028 002660 004767 010270                      JSR    PC, TTOUT      ; REQUEST RH11 ONLY RESPONSE
1029 002664 012705 001024      MOV    #FHOF, R5      ; SET FLAG ADDRESS
1030 002670 012701 000001      MOV    #1, R1        ; SET SIZE OF RESPONSE
1031 002674 012702 000001      MOV    #1, R2        ; SET UPPER LIMIT
1032 002700 012703 000000      MOV    #0, R3        ; SET LOWER LIMIT
1033 002704 004767 010010                      JSR    PC, TTR        ; GO GET RESPONSE
1034 002710 005767 176110      TST     RHJF          ; SEE IF RH11 ONLY
1035 002714 001016                      BNE    ST4            ; IF SO BR
1036 002716 012704 016501      MOV    #MSG55, R4
1037 002722 004767 010226                      JSR    PC, TTOUT      ; REQUEST NRZ ONLY RESPONSE
1038 002726 012705 001022      MOV    #NRZOF, R5      ; SET FLAG ADDRESS
1039 002732 012701 000001      MOV    #1, R1        ; SET SIZE OF RESPONSE
1040 002736 012702 000001      MOV    #1, R2        ; SET UPPER LIMIT
1041 002742 012703 000000      MOV    #0, R3        ; SET LOWER LIMIT
1042 002746 004767 007746                      JSR    PC, TTR        ; GO GET RESPONSE
1043 002752 005067 176050      ST4  CLR    PCNTR      ; CLEAR PASS COUNTER

```

```

1044                                     ,TEST SCHEDULAR*****
1045
1046 002756 000240          TSCD  NOP
1047 002760 005067 176016      CLR      STFLG      ,CLEAR SINGLE TEST FLAG
1048
1049                                     *****
1050
1051 002764 005067 175712          CLR      RH17F      ,SET RH11 INDICATOR
1052 002770 013746 000004          MOV      @#4,-(SP)    ,SAVE ERROR TRAP AND VECTORS
1053 002774 013746 000006          MOV      @#6,-(SP)    ,SAVE PRIORITY
1054 003000 016737 000020 000004  MOV      15,@#4      ,SET TIME OUT
1055 003006 005037 000006          CLR      @#6        ,SET LOW PRIORITY
1056 003012 005777 175616          TST      @BAE      ,REFERENCE BAE REGISTER
1057 003016 012767 000001 175656  MOV      #1,RH17F    ,SET RH70 INDICATOR
1058 003024 012637 000006 15     MOV      (SP)+,@#6    ,RESTORE ERROR TYPE
1059 003030 012637 000004          MOV      (SP)+,@#4
1060
1061                                     *****
1062
1063 003034 017700 175600          MOV      @SWR,R0
1064 003040 042700 177740          BIC      #177740,R0
1065 003044 005700          TST      R0
1066 003046 001055          BNE      STSCD      ,GO SELECT SINGLE TEST
1067 003050 012767 001052 175726  TSRH  MOV      #TSTTBL,LTADD
1068 003056 062767 000004 175720  TSCD0 ADD      #4,LTADD
1069 003064 016767 175714 175702  MOV      LTADD,ITPLP
1070 003072 062767 000002 175674  ADD      #2,ITRLP      ,SET ITEPATION ADDRESS
1071 003100 005777 175700          TST      @LTADD
1072 003104 001002          BNE      TSCD1
1073 003106 000167 000144          JMP      TEND      ,GO TO END ROUTINE
1074 003112 000240          TSCD1 NOP
1075 003114 005067 175636          CLR      STMSK
1076 003120 005067 175600          CLR      ERFP
1077 003124 005067 175554          CLR      HDRFL      ,CLEAR PRINT HEADER FLAG
1078 003130 017700 175650          MOV      @LTADD,R0      ,SET POINTER TO TEST
1079 003134 000110          JMP      (R0)      ,GO TO TEST
1080 003136 000240          TSCD2 NOP
1081 003140 032777 002000 175472  BIT      #2000,@SWR      ,SEE IF HALT ON TEST
1082 003146 001401          BEQ      TSCD3      ,IF NOT BR
1083 003150 000000          HALT
1084 003152 004767 010572          TSCD3 JSR      PC,CKSWR      ,CHECK FOR CNTL G
1085 003156 000240          NOP
1086 003160 005767 175616          TST      STFLG      ,SE IF SINGLE TEST
1087 003164 001734          BEQ      TSCD0      ,IF NOT BR
1088 003166 017700 175446          MOV      @SWR,R0
1089 003172 042700 177740          BIC      #177740,R0      ,MASK TEST NUMBER
1090 003176 005700          TST      R0      ,SEE IF RETURN TO ALL
1091 003200 001666          BEQ      TSCD      ,IF SO BR
1092 003202 000240          STSCD NOP
1093 003204 012767 000001 175570  MOV      #1,STFLG      ,SET SINGLE TEST FLAG
1094 003212 022700 000025          CMP      #25,R0      ,SEE IF EXCEEDED TESTS
1095 003216 003417          BLE      TEND      ,IF SO BR
1096 003220 000241          CLC
1097 003222 006100          ROL      R0
1098 003224 006100          ROL      R0      ,SET TABLE MODIFIER
1099 003226 012767 001052 175550  MOV      #TSTTBL,LTADD

```

```

1100 003234 060067 175544      ADD    RO,LTADD      ,SET TEST POINTER
1101 003240 016767 175540 175526  MOV    LTADD,ITRLP
1102 003246 062767 000002 175520  ADD    #2,ITRLP      ,SET ITERATION POINTER
1103 003254 000716          BR      TSCD1
1104
1105          ,      *****
1106
1107 003256 000240      TEND    NOP
1108 003260 005767 175404      TST     AUTOM          ,CHECK FOR AUTO MODE
1109 003264 001402      BEQ     TEND0      ,BRANCH - IF NOT
1110 003266 000167 177150      JMP     NXTSLV      ,GET ANOTHER SLAVE DEVICE
1111 003272          TEND0
1112
1113          ,      *****
1114
1115 003272 012704 014764      MOV     #MSG6,R4
1116 003276 004767 007652      JSR     PC,TTOUT      ,PRINT END OF PASS
1117 003302 016703 175520      MOV     PCNTR,R3
1118 003306 004767 010006      JSR     PC,OCTP      ,PRINT PASS NUMBER
1119
1120          ,      *****
1121          ,      AUTOMATIC MODE END OF PASS
1122          ,      *****
1123
1124 003312 005767 175520      TST     PAFLG          ,PASS INDICATOR SET?
1125 003316 001002          BNE     35          ,BRANCH - IF SET
1126 003320 005267 175512      INC     PAFLG          ,SET PASS INDICATOR
1127 003324 013704 000042      MOV     @#42,R4      ,CONTENTS OF 42 TO R4
1128 003330 001405          BEQ     HERE          ,BRANCH - IF NOT AUTO MODE
1129 003332 000005          RESET      ,CLEAR THE WORLD
1130 003334 004714      SENDAD  JSR     PC,(R4)      ,RETURN TO MONITOR
1131 003336 000240          NOP
1132 003340 000240          NOP
1133 003342 000240          NOP
1134 003344          HERE
1135 003344 005767 175320      TST     AUTOM          ,CHECK FOR AUTOMATIC MODE
1136 003350 001005          BNE     TENDX      ,BRANCH - IF YES
1137 003352 032777 004000 175260  BIT     #4000,@SWR      ,SEE IF HALT ON PASS
1138 003360 001001          BNE     TENDX      ,IF NOT BR
1139 003362 000000          HALT
1140          TENDX
1141 003364 005767 175300      TST     AUTOM          ,CHECK FOR AUTO MODE
1142 003370 001402          BEQ     15          ,BRANCH - IF NOT
1143 003372 000167 000004          JMP     EXIT          ,RESTART
1144 003376 004767 010346      JSR     PC,CASWR      ,CHECK FOR CNTL G
1145 003402 005267 175420      INC     PCNTR          ,BUMP PASS COUNTER
1146 003406 000167 177344      JMP     TSCD          ,RESTART
1147
1148          ,      *****
  
```

```

1149
1150
1151
1152 003412 012767 016643 175266 FT1 MOV #MSFT1,EMADDR ;SET HEADER
1153 003420 012777 012600 175236 MOV #TRAP,2BTRP ;SET TRAP HANDLER ADDRESS
1154 003426 012777 000340 175232 MOV #340,2BTRP2
1155 003434 012700 000016 MOV #16,R0 ;SET NUMBER OF REGISTER
1156 003440 016701 175134 MOV C1,R1 ;GET FIRST ADDRESS (CS1)
1157 003444 005711 FT1A TST (R1) ;REFERENCE REGISTER
1158 003446 000240 NOP ;IF ADDRESS IS BAD, BUS TRAP WILL OCCUR
1159 003450 005300 FT1B DEC R0 ;SEE IF DONE ALL
1160 003452 001403 BEQ FT1X ;IF SO BR
1161 003454 062701 000002 ADD #2,R1 ;BUMP ADDRESS POINTER
1162 003460 000771 BR FT1A ;CONTINUE
1163 003462 012777 000006 175174 FT1X MOV #6,2BTRF ;RESET TRAP CATCHER
1164 003470 005767 175324 TST RHTF ;SEE IF INITIAL ADDRESS TEST PASS
1165 003474 001404 BEQ FT1XX ;IF NOT BR
1166 003476 005067 175316 CLR RHTF ;CLEAR FLAG
1167 003502 000167 176472 JMP ST1A ;RETURN
1168 003506 000167 177424 FT1XX JMP TSCD2 ;RETURN TO SCHEDULAR
  
```

```

1169
1170 .RM REGISTER BITS READ/WRITE*****
1171
1172 003512 012767 016670 175166 FT2 MOV #MSFT2,EMADDR ,SET TEST HEADER
1173 003520 012701 177777 MOV #-1,R1 ,SET ALL ONES PATTERN
1174 003524 004767 006776 FT2A JSR PC,INIT1 ,GO INIT
1175 003530 016700 175046 MOV WC,R0 ,GET ADDRESS OF WORD COUNT
1176 003534 010102 MOV R1,R2 ,SET EXPT REGISTER BIT PATTERN
1177 003536 010110 MOV R1,(R0) ,LOAD PATTERN
1178 003540 021002 CMP (R0),R2 ,SEE IF EXPT=RCVD
1179 003542 001410 BEQ FT2B ,IF SO BR
1180 003544 012767 015312 175174 MOV #MSG25,ERADD ,SET CODE
1181 003552 012767 003524 175212 MOV #FT2A,SCOLP ,SET SCOPE
1182 003560 004767 000116 JSR PC,FT2ER ,GO DO ERROR
1183 003564 016700 175014 FT2B MOV BA,R0 ,GET ADDRESS OF BUS ADDRESS
1184 003570 010102 MOV R1,R2
1185 003572 042702 000001 BIC #1,R2 ,SET EXPT PATTERN
1186 003576 010110 MOV R1,(R0) ,LOAD PATTERN
1187 003600 020210 CMP R2,(R0) ,SEE IF EXPT=RCVD
1188 003602 001410 BEQ FT2C ,IF SO BR
1189 003604 012767 015320 175134 MOV #MSG26,ERADD ,SET ERROR CODE
1190 003612 012767 003564 175152 MOV #FT2B,SCOLP ,SET SCOPE ADDRESS
1191 003620 004767 000056 JSR PC,FT2ER ,GO DO ERROR
1192 003624 016700 174722 FT2C MOV DB,R0 ,GET ADDRESS OF DATA BUFFER
1193 003630 010102 MOV R1,P2
1194 003632 010110 MOV R1,(R0) ,LOAD PATTERN
1195 003634 012703 004000 MOV #4000,R3
1196 003640 005303 FT2D DEC R3 ,DELAY
1197 003642 001376 BNE FT2D
1198 003644 020210 CMP P2,(R0) ,SEE IF EXPT=RCVD
1199 003646 001410 BEQ FT2E ,IF SO BR
1200 003650 012767 015326 175070 MOV #MSG27,ERADD ,SET ERROR CODE
1201 003656 012767 003624 175106 MOV #FT2C,SCOLP ,SET SCOPE ADDRESS
1202 003664 004767 000012 JSR PC,FT2ER ,GO DO ERROR
1203 003670 005701 FT2E TST R1 ,SEE IF DONE RESET
1204 003672 001454 BEQ FT2X ,IF SO BR
1205 003674 005001 CLR R1 ,SET ZERO PATTERN
1206 003676 000167 177622 JMP FT2A ,DO ZERO BITS
1207 003702 000240 FT2ER NOP
1208 003704 032777 020000 174726 BIT #20000,@SWP ,SEE IF PRINT ERPOP
1209 003712 001034 BNE FT2ERB ,IF NOT BR
1210 003714 005767 174764 TST HDRFL ,SEE IF DONE HEADER
1211 003720 001034 BNE FT2ERA ,IF SO BR
1212 003722 016704 174760 MOV EMADDR,R4
1213 003726 004767 007222 JSR PC,TTOUT ,DO HEADER
1214 003732 012767 000001 174744 FT2ERA MOV #1,HDRFL ,SET FLAG
1215 003740 016704 175002 MOV ERADD,R4
1216 003744 004767 007204 JSR PC,TTOUT ,PRINT ERPOP CODE
1217 003750 012704 015256 MOV #MSG22,R4
1218 003754 004767 007174 JSR PC,TTOUT ,PRINT EXPT TAG
1219 003760 010103 MOV R1,R3
1220 003762 004767 007320 JSR PC,OCTPE ,PRINT EXPT
1221 003766 012704 015266 MOV #MSG23,R4
1222 003772 004767 007156 JSR PC,TTOUT ,PRINT RCVD TAG
1223 003776 011003 MOV (R0),R3
1224 004000 004767 007302 JSR PC,OCTPE ,PRINT RCVD

```

1225	004004	005777	174630	FT2ERB	TST	@SWR	,SEE IF HALT ON ERROR
1226	004010	100001			BPL	FT2ERC	,IF NOT BR
1227	004012	000000			HALT		
1228	004014	004767	006366	FT2ERC	JSR	PC,SCOPE	,GO SEE IF SCOPE ON ERROR
1229	004020	000240			NOP		
1230	004022	000207			RTS	PC	,IF NO SCOPE. CONTINUE TEST
1231	004024	000240		FT2X	NOP		
1232	004026	004767	006416		JSR	PC,ITER	,GC SEE IF ITERATIONS
1233	004032	000167	177100		JMP	TSCD2	,RETURN TO SCHEDULAR


```

1234
1235      *
1236      ,RH INITIALIZE TEST*****
1237 004036 012767 016725 174642 FT3:  MOV  #MSFT3,EMADDR ;SET TEST HEADER
1238 004044 012767 004036 174720      MOV  #FT3,SCOLP
1239 004052 004767 006450      JSR   PC,INIT1 ;GO INIT
1240 004056 052777 020000 174524      BIS  #20000,@CS ;FORCE UPE =1
1241 004064 000240      NOP
1242 004066 004767 006434      JSR   PC,INIT1 ;GO INIT
1243 004072 005777 174502      TST  @C1 ;SEE IF SC IS RESET
1244 004076 100005      BPL  FT3A ;IF SO: BR
1245 004100 012767 015364 174640      MOV  #MSG29,ERADD ;SET ERROR CODE
1246 004106 004767 000060      JSR   PC,FT3ER ;GO DO ERROR
1247 004112 032777 040000 174460 FT3A: BIT  #40000,@C1 ;SEE IF TRE IS RESET
1248 004120 001405      BEQ  FT3B ;IF SO: BR
1249 004122 012767 015413 174616      MOV  #MSG30,ERADD ;SET ERROR CODE
1250 004130 004767 000036      JSR   PC,FT3ER ;GO DO ERROR
1251 004134 017701 174450      FT3B: MOV  @CS,R1 ;GET CS2
1252 004140 042701 000307      BIC  #307,R1 ;MARK IR/OR
1253 004144 005701      TST  R1 ;SEE IF RESET
1254 004146 001405      BEQ  FT3X ;IF SO: BR
1255 004150 012767 015443 174570      MOV  #MSG31,ERADD ;SET ERROR CODE
1256 004156 004767 000010      JSR   PC,FT3ER ;GO DO ERROR
1257 004162 004767 006262      FT3X: JSR   PC,ITER ;GO SEE IF ITERATION
1258 004166 000167 176744      JMP  TSCD2 ;RETURN TO SCHEDULAR
1259 004172 032777 020000 174440 FT3ER: BIT  #20000,@SWR ;SEE IF PRINT ERROR
1260 004200 001015      BNE  FT3ERB ;IF NOT BR
1261 004202 005767 174476      TST  HDRFL ;SEE IF DONE HEADER
1262 004206 001006      BNE  FT3ERA ;IF SO BR
1263 004210 016704 174472      MOV  EMADDR,R4
1264 004214 004767 006734      JSR   PC,TTOUT ;PRINT HEADER
1265 004220 005267 174460      INC  HDRFL
1266 004224 016704 174516      FT3ERA: MOV  ERADD,R4
1267 004230 004767 006720      JSR   PC,TTOUT ;PRINT ERROR CODE
1268 004234 005777 174400      FT3ERB: TST  @SWR ;SEE IF HALT ON ERROR
1269 004240 100001      BPL  FT3ERC ;IF NOT BR
1270 004242 000000      HALT
1271 004244 000240      FT3ERC: NOP
1272 004246 004767 006134      JSR   PC,SCOPE ;GO SEE IF SCOPE
1273 004252 000207      RTS  PC ;IF NOT BR
  
```

```

1274
1275
1276
1277 004254 005767 174422 FT4 TST RH17F
1278 004260 001141 BNE FT5X ; IF RH70. BR
1279 004262 012767 016757 174416 MOV #MSFT4,EMADDR ; SET TEST TEST HEADER
1280 004270 012777 000040 174312 MOV #40,ACS ; INIT
1281 004276 017700 174320 MOV @DB,R0 ; READ DB
1282 004302 005777 174302 TST ACS ; SEE IF DLT IS SET
1283 004306 100013 BPL FT4ER ; IF NOT. BR
1284 004310 005777 174264 TST AC1 ; SEE IF SC IS SET
1285 004314 100014 BPL FT4ERA ; IF NOT. BR
1286 004316 032777 040000 174254 BIT #40000,AC1 ; SEE IF TRE IS SET
1287 004324 001414 BEQ FT4ERB ; IF NOT. BR
1288 004326 004767 006116 FT4X JSR PC,ITER ; GO SEE IF ITERATION
1289 004332 000167 176600 JMP TSCD2 ; RETURN TO SCHEDULAR
1290 004336 012767 015473 174402 FT4ER MOV #MSG32,ERADD ; SET ERROR CODE
1291 004344 000407 BR FT4ERC
1292 004346 012767 015511 174372 FT4ERA MOV #MSG33,ERADD ; SET ERROR CODE
1293 004354 000403 BR FT4ERC
1294 004356 012767 015526 174362 FT4ERB MOV #MSG34,ERADD ; SET ERROR CODE
1295 004364 000240 FT4ERC NOP
1296 004366 012767 004254 174376 MOV #FT4,SCOLP ; SET SCOPE ADDRESS
1297 004374 004767 177572 JSR PC,FT3ER ; GO PRINT ERROR
1298 004400 000752 BR FT4X
  
```



```

1299
1300      ,RH11 SILO TEST 2: IR/OR CHECK*****
1301
1302 004402 005767 174274      FT5  TST  RH17F      ;SEE IF RH70
1303 004406 001066      BNE  FT5X      ;IF SO: BR
1304 004410 012767 017007 174270  MOV  #MSFT5,EMADDR ;SET TEST HEADER
1305 004416 012767 004424 174346  MOV  #FT5A,SCOLP   ;SET SCOPE ADDRESS
1306 004424 004767 006076      FT5A JSR  PC,INIT1     ;GC INIT
1307 004430 032777 000100 174152  BIT   #100,ACS     ;SEE IF IR IS SET
1308 004436 001005      BNE  FT5B      ;IF SO: BR
1309 004440 012767 015544 174300  MOV  #MSG35,ERADD  ;SET ERROR CODE
1310 004446 004767 000122      JSR  PC,FT5ER   ;GO DO ERROR
1311 004452 032777 000200 174130  BIT   #200,ACS     ;SEE IF OR IS RESET
1312 004460 001405      BEQ  FT5C      ;IF SO: BR
1313 004462 012767 015571 174256  MOV  #MSG36,ERADD  ;SET ERROR CODE
1314 004470 004767 000100      JSR  PC,FT5ER   ;GO DO ERROR
1315 004474 012777 000000 174120  MOV  #0,ADB      ;LOAD ZERO INTO SILO
1316 004502 032777 000200 174100  BIT   #200,ACS     ;SEE THAT OR RESET
1317 004510 001405      BEQ  FT5D      ;IF IT DOES BR
1318 004512 012767 015620 174226  MOV  #MSG37,ERADD  ;SET ERROR CODE
1319 004520 004767 000050      JSR  PC,FT5ER   ;GO DO ERROR
1320 004524 012777 177777 174070  MOV  #-1,ADB     ;LOAD SILO WITH -1
1321 004532 012700 004000      MOV  #4000,R0
1322 004536 032777 000200 174044  BIT   #200,ACS     ;SEE IF OR IS SET
1323 004544 001007      BNE  FT5X      ;IF SO: BR
1324 004546 005300      DEC  R0
1325 004550 001372      BNE  FT5E      ;AWAIT OR
1326 004552 012767 015620 174166  MOV  #MSG37,ERADD  ;SET ERROR CODE
1327 004560 004767 000010      JSR  PC,FT5ER   ;GO DO ERROR
1328 004564 004767 005660      FT5X JSR  PC,ITER     ;GO SEE IF ITERATION
1329 004570 000167 176342      JMP  TSCD2    ;RETURN TO SCHEDULAR
1330 004574 004767 177372      FT5ER JSR  PC,FT3ER   ;GO PRINT ERROR
1331 004600 000240      NOP
1332 004602 000207      PTS  PC      ;CONTINUE TEST
  
```

```

1333
1334
1335
1336 004604 005767 174072          FT6  TST  RH17F
1337 004610 001052                BNE  FT6X      ; IF RH70: BR
1338 004612 012767 017037 174066  MOV  #MSFT6, EMADDR ; SET TEST HEADER
1339 004620 012767 004626 174144  MOV  #FT6A, SCOLP   ; SET SCOPE ADDRESS
1340 004626 004767 005674          FT6A JSR  PC, INIT1    ; GC INIT
1341 004632 005000                CLR  R0           ; PRESET DATA
1342 004634 010077 173762          FT6B MOV  R0, @DB      ; LOAD SILO
1343 004640 005200                INC  R0           ; BUMP DATA
1344 004642 022700 000102          CMP  #102, R0     ; SEE IF FILLED ALL
1345 004646 001372                BNE  FT6B         ; IF NOT: BR
1346 004650 032777 000100 173732  BIT  #100, @CS     ; SEE IF IR IS RESET
1347 004656 001405                BEQ  FT6C         ; IF SO: BR
1348 004660 012767 015731 174060  MOV  #MSG40, ERADD  ; SET ERROR CODE
1349 004666 004767 000054          JSR  PC, FT6ER    ; GO DO ERROR
1350 004672 032777 000200 173710  FT6C BIT  #200, @CS     ; SEE IF OR IS SET
1351 004700 001005                BNE  FT6D         ; IF SO: BR
1352 004702 012767 015657 174036  MOV  #MSG38, ERADD  ; SET ERROR CODE
1353 004710 004767 000032          JSR  PC, FT6ER    ; GO DO ERROR
1354 004714 005000                FT6D CLR  R0           ; PRESET DATA
1355 004716 017701 173700          FT6E MOV  @DB, R1      ; READ SILO
1356 004722 020001                CMP  R0, R1       ; SEE IF EXPT=RCVD
1357 004724 001014                BNE  FT6DE        ; IF NOT: BR
1358 004726 005200                INC  R0           ; BUMP DATA
1359 004730 022700 000102          CMP  #102, R0     ; SEE IF DONE ALL
1360 004734 001370                BNE  FT6E         ; IF NOT: BR
1361 004736 004767 005506          FT6X JSR  PC, ITER    ; GO SEE IF ITERATION
1362 004742 000167 176170          JMP  TSCD2       ; RETURN TO SCHEDULAR
1363 004746 000240          FT6ER NOP
1364 004750 004767 177216          JSR  PC, FT3ER    ; GO PRINT ERROR
1365 004754 000207                RTS  PC           ; RETURN
1366 004756 000240          FT6DE NOP
1367 004760 032777 020000 173652  BIT  #20000, @SWR   ; SEE IF PRINT ERROR
1368 004766 001032                BNE  FT6DEB       ; IF NOT: BR
1369 004770 005767 173710          TST  HDRFL        ; SEE IF DONE HEADER
1370 004774 016701 173706          MOV  EMADDR, R1
1371 005000 004767 006150          JSR  PC, TTOUT    ; PRINT HEADER
1372 005004 005267 173674          INC  HDRFL        ; SET FLAG
1373 005010 012704 015711          FT6DEA MOV  #MSG39, R4
1374 005014 004767 006134          JSR  PC, TTOUT    ; PRINT SILO READ ERROR
1375 005020 012704 015256          MOV  #MSG22, R4
1376 005024 004767 006124          JSR  PC, TTOUT    ; PRINT EXPT TAG
1377 005030 010003          MOV  R0, R3
1378 005032 004767 006262          JSR  PC, OCTP     ; PRINT EXPT
1379 005036 012704 015266          MOV  #MSG23, R4
1380 005042 004767 006106          JSR  PC, TTOUT    ; PRINT RCVD TAG
1381 005046 010103          MOV  R1, R3
1382 005050 004767 006244          JSR  PC, OCTP     ; PRINT RCVD
1383 005054 005777 173560          FT6DEB TST  @SWR      ; SEE IF HALT ON ERROR
1384 005060 100001                BPL  FT6DEX      ; IF NOT: BR
1385 005062 000000          HALT
1386 005064 004767 006660          FT6DEX JSR  PC, CKSWR   ; CHECK FOR CNTL G
1387 005070 000207                RTS  PC           ; RETURN TO TEST

```

```
1388
1389
1390
1391 005072 005767 173604 FT7 TST RH17F
1392 005076 001021 BNE FT7X ; IF RH70. BR
1393 005100 012767 017067 173600 MOV #MSFT7,EMADDR ; SET TEST HEADER
1394 005106 012767 005072 173656 MOV #FT7,SCOLP ; SET SCOPE ADDRESS
1395 005114 004767 005406 JSR PC,INIT1 ; GC INIT
1396 005120 012700 000103 MOV #103,R0 ; SET SIZE OF SILO +1
1397 005124 010077 173472 FT7A MOV R0,ADB ; LOAD SILO
1398 005130 005301 DEC R0 ; SEE IF DONE
1399 005132 001374 BNE FT7A ; IF NOT. BR
1400 005134 005777 173450 TST ACS ; SEE IF DLT IS SET
1401 005140 100004 BPL FT7ER ; IF NOT. BR
1402 005142 004767 005302 FT7X JSR PC,ITER ; GO SEE IF ITERATION
1403 005146 000167 175764 JMP TSCD2 ; RETURN TO SCHEDULAR
1404 005152 012767 015473 173566 FT7ER MOV #MSG32,ERADD ; SET ERROR CODE
1405 005160 004767 177006 JSR PC,FT3ER ; GO DO ERROR
1406 005164 000766 BR FT7X
```

```

1407
1408
1409
1410 005166 005767 173510          FT10  TST    RH17F
1411 005172 001034                BNE    FT10X
1412 005174 012767 017117 173504    MOV    #MSFT10,EMADDR ; IF RH70: BR
1413 005202 012767 005166 173562    MOV    #FT10,SCOLP  ; SET TEST HEADER
1414 005210 012777 000040 173372    MOV    #40,ACS      ; SET SCOPE ADDRESS
1415 005216 012700 000004          MOV    #4,RO          ; INITIALIZE
1416 005222 010077 173374          MOV    RO,ADB        ; SET NUMBER OF SILO WRITER
1417 005226 005300                DEC    RO              ; WRITE SILO
1418 005230 001374                BNE    FT10A          ; SEE IF DONE
1419 005232 052777 000040 173350    BIS    #40,ACS      ; IF NOT: BR
1420 005240 012777 177777 173354    MOV    #-1,ADB       ; INITIALIZE
1421 005246 017701 173350          MOV    ADB,R1        ; WRITE SILO
1422 005252 017701 173344          MOV    ADB,R1        ; READ SILO 1
1423 005256 005777 173326          TST    ACS           ; READ SILO 2
1424 005262 100011                BPL    FT10ER        ; SEE IF DLT IS SET
1425 005264 004767 005160          FT10X JSR    PC,ITER      ; IF NOT: BR
1426 005270 005767 173530          TST    RHOF          ; GO SEE IF ITERATION
1427 005274 001402                BEQ    FT10XX        ; SEE IF RH11 ONLY
1428 005276 000167 175754          JMP    TEND        ; IF NOT: BR
1429 005302 000167 175630          FT10XX JMP    TSCD2       ; ELSE GO TO END
1430 005306 012767 015473 173432    FT10ER MOV    #MSG32,ERADD ; RETURN TO SCHEDULAR
1431 005314 004767 176652          JSR    PC,FT3ER    ; SET ERROR CODE
1432 005320 000761                BR     FT10X        ; GO DO ERPOR

```

```

1433                                     ,NOP TEST*****
1434
1435 005322 000240                      FT11 NOP
1436 005324 012767 005322 173440      MOV #FT11, SCOLP , SET SCOPE ADDRESS
1437 005332 004767 005170              JSR PC, INIT1
1438 005336 012767 000300 173450      MOV #300, UDES , SET TC= ALL NRZ, NORM, ODD
1439 005344 012767 177777 173344      MOV #-1, FCNT ; SET FC= ALL OVER
1440 005352 012767 177777 173340      MOV #-1, WCNT ; SET WC= ALL OVER
1441 005360 012767 177777 173326      MOV #-1, BADDR ; SET BA= ALL OVER
1442 005366 012767 000001 173340      MOV #1, RDYDX ; SET DELAY
1443 005374 012767 000001 173334      MOV #1, OPDYX ; SET OP DELAY
1444 005402 012767 000001 173376      MOV #1, FUN ; SET NOP FUNCTIONS CODE
1445 005410 004767 003760              JSR PC, EXEC ; GO EXECUTE COMMAND
1446 005414 000240                      NOP
1447 005416 012767 017150 173262      MOV #MSFT11, EMADDR
1448 005424 004767 004174              JSR PC, ERCHK , GO CHECK REGISTER
1449 005430 004767 005014              JSR PC, ITER , GO SEE IF ITERATIONS
1450 005434 000167 175476              JMP TSCD2 , RETURN TO SCHEDULAR
  
```

```

1451                                     ,REWIND TEST*****
1452
1453 005440 000240                      FT12  NOP
1454 005442 012767 005440 173322      MOV    #FT12, SCOLP
1455 005450 004767 005052                      JSR    PC, INIT1      ;GO INITIALIZE
1456 005454 052777 001700 173150      BIS    #1700, @TC      ;SET TO NRZ, NORMAL
1457 005462 012767 177760 173226      MOV    #-20, FCNT      ;SET FC=20
1458 005470 012767 177770 173222      MOV    #-10, WCNT      ;SET WC=10
1459 005476 012767 017630 173210      MOV    #WDATA, BADDR    ;SET BA=WRITE BUFFER
1460 005504 012767 000007 173274      MOV    #7, FUN        ;SET REWIND OP CODE
1461 005512 004767 003656                      JSR    PC, EXEC        ;GO EXECUTE COMMAND
1462 005516 000240
1463 005520 032777 020000 173064      FT12A  BIT    #20000, @DS
1464 005526 001374                      BNE    FT12A          ;AWAIT PIP
1465 005530 012767 017170 173150      MOV    #MSFT12, EMADDR
1466 005536 004767 004062                      JSR    PC, ERCHK      ;GO CHECK FOR ERROR
1467 005542 004767 004702                      JSR    PC, ITER      ;GO SEE IF ITERATION
1468 005546 000167 175364                      JMP    TSCD2         ;RETURN TO SCHEDULAR
1469

```



```

1470                                     ;WRITE/READ TEST*****
1471
1472 005552 000240          FT13  NOP
1473 005554 012767 000001 173152  MOV    #1,RCYDX
1474 005562 012767 000001 173146  MOV    #1,OPDYX
1475 005570 012767 000100 173124  MOV    #100,RCNT      ;SET RECORD COUNT
1476 005576 012767 017213 173102  MOV    #MSFT13,EMADDR ;SET TEST HEADER
1477 005604 012767 000001 173204  MOV    #1,PATRN
1478 005612 004767 004414          JSR    PC,DSUP      ;SET UP ALL ONES DATA PATTERN
1479 005616 012767 000300 173170  MOV    #300,UDES      ;REWIND TO BOT
1480 005624 004767 003676  FT13A  JSR    PC,RWND      ;SET 200 BPI, NORMAL
1481 005630 012767 177600 173060  MOV    #-200,FCNT     ;SET FC
1482 005636 012767 177700 173054  MOV    #-100,WCNT     ;SET WC
1483 005644 012767 017630 173042  MOV    #WDATA,BADDR   ;SET BA
1484 005652 012767 000061 173126  MOV    #61,FUN        ;SET WRITE OP-CODE
1485 005660 012767 015072 173036  MOV    #MSG12,ERRP
1486 005666 004767 003502  FT13B  JSR    PC,EXEC      ;GO EXECUTE COMMAND
1487 005672 005067 173074          CLR    SCOLP      ;NO SCOPE LOOP
1488 005676 004767 003722          JSR    PC,ERCHK     ;GO CHECK ERROR
1489 005702 005367 173014          DEC    RCNT        ;SEE IF DONE ALL
1490 005706 001367          BNE    FT13B              ;IF NOT: BR
1491 005710 012767 000100 173004  MOV    #100,RCNT      ;SET RECORD COUNT
1492 005716 012767 021342 172770  MOV    #RDATA,BADDR
1493 005724 062767 000200 172762  ADD     #200,BADDR    ;SET BA
1494 005732 012767 000077 173046  MOV    #77,FUN        ;SET READ REVERSE OP-CPDE
1495 005740 012767 015110 172756  MOV    #MSG13,ERRP
1496 005746 004767 003422  FT13C  JSR    PC,EXEC      ;GO EXECUTE COMMAND
1497 005752 004767 003646          JSR    PC,ERCHK     ;GO CHECK ERROR
1498 005756 005367 172740          DEC    RCNT        ;SEE IF READ ALL
1499 005762 001371          BNE    FT13C              ;IF NOT: BR
1500 005764 162767 000200 172722  SUB     #200,BADDR    ;SET BA
1501 005772 012767 000071 173006  MOV    #71,FUN        ;SET READ FORWARD OP-CODE
1502 006000 012767 015135 172716  MOV    #MSG14,ERRP
1503 006006 012767 000100 172706  MOV    #100,RCNT     ;SET RECORD COUNT
1504 006014 004767 003354  FT13D  JSR    PC,EXEC      ;GO EXECUTE COMMAND
1505 006020 004767 003600          JSR    PC,ERCHK     ;GO CHECK ERRORS
1506 006024 005367 172672          DEC    RCNT        ;SEE IF DONE ALL
1507 006030 001371          BNE    FT13D              ;IF NOT: BR
1508 006032 032767 002000 172754  BIT     #2000,UDES    ;SEE IF DONE PE
1509 006040 001017          BNE    FT13X              ;IF SO: BR
1510 006042 062767 000400 172744  ADD     #400,UDES    ;SELECT NEXT DENSITY
1511 006050 032767 002000 172736  BIT     #2000,UDES    ;SEE IF PE
1512 006056 001403          BEQ    FT13E              ;IF NOT: BR
1513 006060 005767 172736          TST    NR20F      ;SEE IF NRZ ONLY
1514 006064 001005          BNE    FT13X              ;IF SO: BR
1515 006066 012767 000100 172626  FT13E  MOV    #100,RCNT   ;RESET RECORD COUNT
1516 006074 000167 177524          JMP    FT13A        ;GO DO NEXT DENSITY
1517 006100 000167 175032  FT13X  JMP    TSCDC      ;RETURN TO SCHEDULAR

```

```

1518                                     ,SPACE TEST*****
1519
1520 006104 000240 FT14 NOP
1521 006106 012767 017242 172572 MOV #MSFT14,EMADDR ,SET TEST HEADER
1522 006114 012767 001700 172672 MOV #1700,UDES ,SET NRZ,NORMAL
1523 006122 004767 003400 FT14A1 JSR PC,RWNO ,GO INITIALIZE
1524 006126 012767 000100 172566 MOV #100,RCNT ,SET NUMBER OF RECORDER
1525 006134 012767 177777 011466 MOV #-1,WDATA ,SET DATA PATTERN
1526 006142 012767 177700 172546 MOV #-100,FCNT ,PRESET FRAME CNT
1527 006150 012767 177740 172542 MOV #-40,WCNT ,PRESET WORD CNT
1528 006156 004767 004344 FT14A JSR PC,INIT1 ,GO REWIND
1529 006162 012767 001000 172546 MOV #1000,OPDYX
1530 006170 012767 040000 172536 MOV #40000,RDYDX
1531 006176 012767 000061 172602 MOV #61,FUN ,SET WRITE OP-CODE
1532 006204 012767 102300 172544 MOV #102300,STMSK ,MASK DATA RELATED ERRORS
1533 006212 052777 000010 172370 BIS #10,ACS ,INHIBIT BUS ADDRESS INCREMENT
1534 006220 004767 003150 JSR PC,EXEC ,GO EXECUTE COMMAND
1535 006224 000240 NOP
1536 006226 012767 016145 172470 MOV #MSG46,ERRP ,SET ERROR CODE
1537 006234 004767 003364 JSR PC,ERCHK ,GO CHECK ERRORS
1538 006240 005767 172544 TST SERFL ,SEE IF ERROR
1539 006244 001402 BEQ FT14A2 ,IF NOT BR
1540 006246 000167 000466 JMP FT14X ,ELSE EXIT
1541 006252 162767 000001 172436 FT14A2 SUB #1,FCNT ,BUMP FC
1542 006260 032767 000001 172430 BIT #1,FCNT ,SEE IF SHOULD BUMP WC
1543 006266 001403 BEQ FT14A3 ,IF NOT BR
1544 006270 162767 000001 172422 SUB #1,WCNT ,BUMP WC
1545 006276 005367 172420 FT14A3 DEC RCNT ,SEE IF DONE ALL
1546 006302 001325 BNE FT14A ,WRITE ALL RECORDS
1547 006304 000240 NOP
1548 006306 012767 000100 172414 MOV #100,RRD ,PRESET RECORD POSITION
1549 006314 012767 000176 172410 MOV #176,RFD
1550 006322 000240 NOP
1551 006324 012767 177701 172406 MOV #-77,SCNT ,SET SPACE AMOUNT
1552 006332 012767 000033 172446 FT14B MOV #33,FUN ,SET OP-CODE SPACE REVERSE
1553 006340 004767 003030 JSR PC,EXEC ,GO EXECUTE COMMAND
1554 006344 012767 016216 172352 MOV #MSG48,ERRP ,SET ERROR CODE
1555 006352 004767 003246 JSR PC,ERCHK ,GO CHECK ERRORS
1556 006356 005767 172426 TST SERFL ,SEE IF ERROR
1557 006362 001166 BNE FT14X ,IF SO BR
1558 006364 004767 000070 JSR PC,FT14RR ,GO READ REVERSE + CHECK DATA
1559 006370 000240 NOP
1560 006372 012767 000031 172406 MOV #31,FUN ,SET SPACE FORWARD OP-CODE
1561 006400 005267 172334 INC SCNT ,SET SPACE AMOUNT
1562 006404 001555 BEQ FT14X ,IF DONE BR
1563 006406 004767 002762 JSR PC,EXEC ,GO EXECUTE COMMAND
1564 006412 012767 016171 172304 MOV #MSG47,ERRP ,SET ERROR CODE
1565 006420 004767 003200 JSR PC,ERCHK ,GO CHECK ERROR
1566 006424 005767 172360 TST SERFL ,SEE IF ERROR FLAG
1567 006430 001143 BNE FT14X ,IF NO BR
1568 006432 004767 000064 JSR PC,FT14RF ,GO READ FORWARD FOR POSITION CHECK
1569 006436 000240 NOP
1570 006440 005267 172274 INC SCNT ,DECREMENT SPACE AMOUNT
1571 006444 001535 BEQ FT14X ,IF DONE BR
1572 006446 005267 172256 INC RRD ,BUMP DATA EXPT
1573 006452 005367 172254 DEC RFD ,BUMP DATA EXPT
  
```


Address	Hex	Dec	Label	Op	Op	Comment
1574	006456	000725		BR	FT14B	
1575	006460	000240		NOP		
1576	006462	012767	021342	MOV	#RDATA, BADDR	; SET BA
1577	006470	012767	000077	MOV	#77, FUN	; SET READ REVERSE OP-CODE
1578	006476	004767	002672	JSR	PC, EXEC	; GO EXECUTE COMMAND
1579	006502	000240		NOP		
1580	006504	016705	172220	MOV	RRO, R5	
1581	006510	020577	172072	CMP	R5, @FC	; SEE IF CORRECT RECORD
1582	006514	001020		BNE	FT14RER	; IF NOT BR
1583	006516	000167	000026	JMP	FT14EC	; GO CLEAR RH11 ERROR BIT
1584	006522	000240		NOP		
1585	006524	012767	000071	MOV	#71, FUN	; SET READ FORWARD OP-CODE
1586	006532	004767	002636	JSR	PC, EXEC	; GO EXECUTE COMMAND
1587	006536	016705	172170	MOV	RFD, R5	
1588	006542	020577	172040	CMP	R5, @FC	; SEE IF CORRECT RECORD
1589	006546	001003		BNE	FT14RER	; IF NOT BR
1590	006550	004767	003752	JSR	PC, INIT1	; CLEAR RH
1591	006554	000207		RTS	PC	; RETURN
1592	006556	000240		NOP		
1593	006560	032777	020000	BIT	#20000, @SWR	; SEE IF PRINT INHIBITED
1594	006566	001060		BNE	FT14R3	; IF SO BR
1595	006570	012704	017242	MOV	#MSFT14, R4	
1596	006574	004767	004354	JSR	PC, TTOUT	; PRINT HEADER
1597	006600	012704	015010	MOV	#MSG9, R4	
1598	006604	004767	004344	JSR	PC, TTOUT	; PRINT ERROR TYPE
1599	006610	012704	015243	MOV	#MSG20, R4	; SET NRZ TAG POINTER
1600	006614	032767	002000	BIT	#2000, UDES	; SEE IF PE
1601	006622	001402		BEQ	FT14R0	; IF NOT BR
1602	006624	012704	015251	MOV	#MSG21, R4	; ELSE SET PE TAG POINTER
1603	006630	004767	004320	JSR	PC, TTOUT	; PRINT TAG
1604	006634	032767	000002	BIT	#2, FUN	; SEE IF READ REVERSE
1605	006642	001003		BNE	FT14R1	; IF SO BR
1606	006644	012704	015223	MOV	#MSG17, R4	
1607	006650	000402		BR	FT14R2	; GO PRINT
1608	006652	012704	015203	MOV	#MSG16, R4	
1609	006656	004767	004272	JSR	PC, TTOUT	; PRINT FRWD/REV
1610	006662	012704	015256	MOV	#MSG22, R4	
1611	006666	004767	004262	JSR	PC, TTOUT	; PRINT EXPT TAG
1612	006672	010503		MOV	R5, R3	
1613	006674	042703	177700	BIC	#177700, R3	; MASK RECORD NUMBER
1614	006700	004767	004414	JSR	PC, OCTP	; PRINT EXPT RECORD NUMBER
1615	006704	012704	015266	MOV	#MSG23, R4	
1616	006710	004767	004240	JSR	PC, TTOUT	; PRINT RCVD TAG
1617	006714	017703	171666	MOV	@FC, R3	
1618	006720	042703	177700	BIC	#177700, R3	; MASK RECORD NUMBER
1619	006724	004767	004370	JSR	PC, OCTP	; PRINT ACTUAL RECORD NUMBER
1620	006730	005777	171704	TST	@SWR	; SEE IF HALT ON ERROP
1621	006734	100001		BPL	FT14X	; IF NOT BR
1622	006736	000000		HALT		
1623	006740	004767	005004	JSR	PC, CKSWR	; CHECK FOR CNTL G
1624	006744	005767	172052	TST	NRZOF	; SEE IF NRZ ONLY
1625	006750	001011		BNE	FT14XX	; IF SO BR
1626	006752	032767				

CZTUBGO TMO2 - TU16/TE16 BSC FC MACY11 30(1046) 05-DEC-77 12 40¹ 4⁴ PAGE 47
CZTUBG P11 05-DEC-77 12 39

SEQ 0047

1620 006774 000167 174136 FT14XX JMP TSCD2 ,RETURN TO SCHEDULAR

```

1631                                     ,ERASE TEST*****
1632
1633 007000 000240                      FT15  NOP
1634 007002 005067 171750              CLR    STMSK
1635 007006 012767 000100 171720      MOV    #100, RDYDX
1636 007014 012767 000010 171714      MOV    #10, OPDYX
1637 007022 012767 017264 171656      MOV    #MSFT15, EMADDR ; SET TEST HEADER
1638 007030 004767 002472              JSR    PC, RWND ; REWIND
1639 007034 012767 021342 171652      MOV    #RDATA, BADDR ; SET BA
1640 007042 012767 001700 171744      MOV    #1700, UDES ; SET NRZ, NORMAL
1641 007050 012767 000025 171730      MOV    #25, FUN ; SET ERASE OP-CODE
1642 007056 012767 000200 171636      MOV    #200, RCNT ; SET TO ERASE 128 TIMES
1643 007064 004767 002304              FT15B JSR    PC, EXEC ; GO EXECUTE COMMAND
1644 007070 012767 016145 171626      MOV    #MSG46, ERPP ; SET ERROR CODE
1645 007076 004767 002522              JSR    PC, ERCHK ; GO CHECK ERRORS
1646 007102 005767 171702              TST    SERFL ; SEE IF ANY ERRORS
1647 007106 001032                      BNE    FT15X ; IF SO EXIT
1648 007110 005367 171606              DEC    RCNT ; SEE IF DONE ERASING
1649 007114 001363                      PNE    FT15B ; IF NOT BR
1650 007116 000240                      NOP
1651 007120 004767 002402              JSR    PC, RWND ; REWIND
1652 007124 012767 177600 171566      MOV    #-200, WCNT ; SET WC
1653 007132 012767 000071 171646      MOV    #71, FUN ; SET READ FORWARD OP-CODE
1654 007140 012767 000040 171566      MOV    #40, RDYDX ; SET DELAY
1655 007146 004767 002222              JSR    PC, EXEC ; GO EXECUTE COMMAND
1656 007152 000240                      NOP
1657 007154 012767 016541 171542      MOV    #MSG60, ERPP ; SET ERROR CODE
1658 007162 012767 020000 171566      MOV    #20000, STMSK
1659 007170 004767 002430              JSR    PC, ERCHK ; GO CHECK ERRORS
1660 007174 000167 173736              FT15X JMP    TSCD2 ; RETURN TO SCHEDULAR
  
```

```

1661                                     ,TAPE MARK WRITE/READ TEST*****
1662
1663 007200 000240 FT16 NOP
1664 007202 012767 000001 171524 MOV #1,RDYDX
1665 007210 012767 001000 171520 MOV #1000,OPDYX
1666 007216 012767 017306 171462 MOV #MSFT16,EMADDR ;SET HEADER
1667 007224 012767 001700 171562 MOV #1700,UDES ;SET TO NRZ,NORMAL,ODD
1668 007232 004767 002270 FT16A JSR PC,RWND ;REWIND
1669 007236 012767 177760 171452 FT16B MOV #-20,FCNT ;FC=20
1670 007244 012767 177770 171446 MOV #-10,WCNT ;WC=10
1671 007252 012767 000027 171526 MOV #27,FUN ;SET WRITE TAPE MARK OP-CODE
1672 007260 004767 002110 JSR PC,EXEC ;GO EXECUTE COMMAND
1673 007264 012767 001000 171464 MOV #1000,STMSK ;SET FOR FCE MASK
1674 007272 012767 015072 171424 MOV #MSG12,ERRP ;SET ERROR CODE
1675 007300 004767 002320 JSR PC,ERCHK ;GO CHECK ERROR
1676 007304 004767 002662 JSR PC,TMCHK ;GO SEE IF TM SET
1677 007310 012767 000077 171470 MOV #77,FUN ;SET USED REVERSE OP-CODE
1678 007316 004767 002052 JSR PC,EXEC ;GO EXECUTE COMMAND
1679 007322 012767 001000 171426 MOV #1000,STMSK ;SET FCE ERROR MASK
1680 007330 012767 015110 171366 MOV #MSG13,ERRP ;SET ERROR CODE
1681 007336 004767 002262 JSR PC,ERCHK ;GO CHECK ERRORS
1682 007342 004767 002624 JSR PC,TMCHK ;GO SEE IF TM SET
1683 007346 012767 000071 171432 MOV #71,FUN ;SET READ FORWARD OP-CODE
1684 007354 004767 002014 JSR PC,EXEC ;GO EXECUTE COMMAND
1685 007360 012767 015135 171336 MOV #MSG14,ERRP ;SET ERROR CODE
1686 007366 004767 002232 JSR PC,ERCHK ;GO CHECK ERRORS
1687 007372 004767 002574 JSR PC,TMCHK ;GO SEE IF TM SET
1688 007376 032767 002000 171410 BIT #2000,UDES ;SEE IF DONE PE
1689 007404 001012 BNE FT16X ;IF SO BR
1690 007406 005767 171410 TST NRZOF ;SEE IF NRZ ONLY
1691 007412 001007 BNE FT16X ;IF SO BR
1692 007414 012767 002300 171372 MOV #2300,UDES ;SET PE, NORMAL
1693 007422 004767 003100 JSR PC,INIT1 ;INITIALIZE
1694 007426 000167 177604 JMP FT16B ;DO IN PE
1695 007432 004767 003012 FT16X JSR PC,ITER ;DO ITERATIONS
1696 007436 000167 173474 JMP TSCD2 ;RETURN TO SCHEDULAP
1697

```

```

1698
1699
1700
1701 007442 005067 171254 FT17 CLR RCNT
1702 007446 012767 017347 171232 MOV #MSFT17,EMADDR ;SET HEADER
1703 007454 012767 001700 171332 MOV #1700, UDES ;SET TO NRZ
1704 007462 004767 002040 FT17A JSR PC,RWND ;REWIND TAPE
1705 007466 012767 000027 171312 FT17B MOV #27,FUN
1706 007474 012767 040000 171232 MOV #40000,RDYDX ;SET DRY DELAY
1707 007502 012767 040000 171226 MOV #40000,OPDYX ;SET OP DELAY
1708 007510 004767 001660 JSR PC,EXEC ;GO WRITE TM
1709 007514 012767 102300 171234 MOV #102300,STMSK ;MASK DATA RELATED ERRORS
1710 007522 012767 015162 171174 MOV #MSG15,ERRP ;SET ERROR TYPE
1711 007530 004767 002070 JSR PC,ERCHK ;GO CHECK ERROR
1712 007534 005767 171250 TST SERFL ;SEE IF ERROR
1713 007540 001144 BNE FT17X ;IF SO BR
1714 007542 004767 002424 JSR PC,TMCHK ;GO SEE IF TM SET
1715 007546 000240 NOP
1716 007550 000240 NOP
1717 007552 032767 000100 171142 BIT #100,RCNT ;SEE IF DONE PATTERN
1718 007560 001046 BNE FT17D ;IF SO BR
1719 007562 062767 000020 171132 ADD #20,RCNT ;ADD 20 TO RECORD COUNT
1720 007570 016767 171126 171152 MOV RCNT,TEMP1 ;SAVE RECORD COUNT
1721 007576 012767 177600 171114 MOV #-200,WCNT ;WC=128
1722 007604 012767 177400 171104 MOV #-400,FCNT ;FC=256
1723 007612 012767 017630 171074 MOV #WDATA,BADDR ;BA=WRITE BUFFER
1724 007620 012767 000061 171160 MOV #61,FUN ;SET WRITE OP CODE
1725 007626 000240 FT17C NOP
1726 007630 000240 NOP
1727 007632 004767 001536 JSR PC,EXEC ;GO WRITE
1728 007636 012767 015072 171060 MOV #MSG12,ERRP ;SET ERROR CODE
1729 007644 012767 102300 171104 MOV #102300,STMSK ;MASK DATA RELATED ERRORS
1730 007652 004767 001746 JSR PC,ERCHK ;GO CHECK ERROR
1731 007656 005767 171126 TST SERFL ;SEE IF ERROR
1732 007662 001073 BNE FT17X ;IF SO BR
1733 007664 005367 171060 DEC TEMP1 ;SEE IF DONE ALL
1734 007670 001356 BNE FT17C ;IF NOT BR
1735 007672 000167 177570 JMP FT17B ;ELSE GO DO TM
1736 007676 000240 FT17D NOP
1737 007700 012767 000033 171100 MOV #33,FUN ;SET SPACE REVERSE
1738 007706 012767 015203 171010 MOV #MSG16,ERRP ;SET ERROR CODE
1739 007714 012767 177600 171016 FT17D1 MOV #-200,SCNT ;SET TO 200 RECORDS
1740 007722 012767 000005 170772 MOV #5,RCNT ;SET NUMBER OF OPS TO DO
1741 007730 004767 002572 FT17E JSR PC,INIT1 ;GO INIT
1742 007734 004767 001434 JSR PC,EXEC ;GO SPACE
1743 007740 012767 001000 171010 MOV #1000,STMSK ;SET ERROR MASK
1744 007746 004767 001652 JSR PC,ERCHK ;GO CHECK ERROR
1745 007752 005767 171032 TST SERFL ;SEE IF ERROR
1746 007756 001035 BNE FT17X ;IF SO BR
1747 007760 004767 002206 JSR PC,TMCHK ;GO SEE IF TM SET
1748 007764 005367 170732 DEC RCNT ;SEE IF DONE SPACES
1749 007770 001357 BNE FT17E ;IF NOT BR
1750 007772 022767 000031 171006 CMP #31,FUN ;SEE IF DONE FORWARD
1751 010000 001410 BEQ FT17F ;IF SO BR
1752 010002 012767 015223 170714 MOV #MSG17,ERRP ;SET ERROR CODE
1753 010010 012767 000031 170770 MOV #31,FUN ;SET TO SPACE FORWARD

```

1754	010016	000167	177672		JMP	FT17D1	, DO FORWARD
1755	010022	032767	002000	170764	FT17F	BIT	#2000, UDES
1756	010030	001010			BNE	FT17X	, IF SO. BR
1757	010032	005767	170764		TST	NR20F	, SEE IF NRZ ONLT
1758	010036	001005			BNE	FT17X	, IF SO BR
1759	010040	012767	002300	170746	MOV	#2300, UDES	, SET TO PE
1760	010046	000167	177410		JMP	FT17A	, GO PE
1761	010052	000167	173060		FT17X	JMP	TSCD2
							, RETURN TO SCHEDULAR


```

1762
1763
1764
1765 010056 000240
1766 010060 012767 017375 170620
1767 010066 004767 001434
1768 010072 012767 000003 170716
1769 010100 004767 002126
1770 010104 012767 017630 170602
1771 010112 012767 177400 170576
1772 010120 012767 177600 170572
1773 010126 012767 001700 170660
1774 010134 012767 000061 170644
1775 010142 004767 001226
1776 010146 012767 016145 170550
1777 010154 004767 001444
1778 010160 005767 170624
1779 010164 001050
1780 010166 012767 015203 170530
1781 010174 012767 000057 170604
1782 010202 062767 000376 170504
1783 010210 004767 001160
1784 010214 004767 001404
1785 010220 012767 015223 170476
1786 010226 012767 000051 170552
1787 010234 162767 000376 170452
1788 010242 004767 001126
1789 010246 004767 001352
1790 010252 032767 002000 170534
1791 010260 001012
1792 010262 005767 170534
1793 010266 001007
1794 010270 012767 002300 170516
1795 010276 004767 002224
1796 010302 000167 177626
1797 010306 004767 002136
1798 010312 000167 172620

, WRITE CHECK TEST*****

FT20 NOP
MOV #MSFT20, EMADDR ; SET HEADER
JSR PC, RWND ; REWIND
MOV #3, PATRN
JSR PC, DSUP ; GO SET PATTERN 3
MOV #WDATA, BADDR ; SET BA
MOV #-400, FCNT ; SET FC
MOV #-200, WCNT ; SET WC
MOV #1700, UDES ; SET NRZ NORMAL
FT20A MOV #61, FUN ; SET WRITE OP CODE
JSR PC, EXEC ; GO WRITE RECORD
MOV #MSG46, ERRP ; SET ERROR CODE
JSR PC, ERCHK ; GO CHECK ERROR
TST SERFL ; SEE IF ERORR
BNE FT20X ; IF SO: BR
MOV #MSG16, ERRP ; SET REVERSE ERROR TAG
MOV #57, FUN ; SET REVERSE WRITE CHECK OP-CODE
ADD #376, BADDR ; SET BA FOR REVERSE CHECK
JSR PC, EXEC ; GO DO REVERSE CHECK
JSR PC, ERCHK ; GO CHECK ERROR
FT20B MOV #MSG17, ERRP ; SET FORWARD TAG
MOV #51, FUN ; SET FORWARD CHECK OP CODE
SUB #376, BADDR ; SET BA FOR FORWARD CHECK
JSR PC, EXEC ; GO DO FORWARD CHECK
JSR PC, ERCHK ; GO CHECK ERROR
FT20C BIT #2000, UDES ; SEE IF DONE PE
BNE FT20X ; IF SO BR
TST NRZOF ; SEE IF NRZ ONLY
BNE FT20X ; IF SO BR
MOV #2300, UDES ; ELSE SET PE
JSR PC, INIT1 ; GO INIT
JMP FT20A ; DO IN PE
FT20X JSR PC, ITER ; DO ITERATIONS
JMP TSCD2 ; RETURN TO SCHEDULAR
  
```

```

1799
1800      ,ERASE HEAD TEST*****
1801
1802 010316 012767 017426 170362 FT21. MOV    #MSFT21,EMADDR ;SET TEST HEADER
1803 010324 004767 001176      JSR    PC,RWND      ;GO REWIND
1804 010330 012767 000003 170460 MOV    #3,PATRN
1805 010336 004767 001670      JSR    PC,DSUP      ;GO SET PATTERN 3
1806 010342 012767 017630 170344 MOV    #WDATA,BADDR ;SET BA=WRITE BUFFER
1807 010350 012767 176340 170340 MOV    #-1440,FCNT   ;SET FC=800
1808 010356 012767 177160 170334 MOV    #-620,WCNT    ;SET WC=400
1809 010364 012767 001700 170422 MOV    #1700,UDES    ;SET NRZ, NORMAL
1810 010372 012767 000061 170406 MOV    #61,FUN       ;SET WRITE OP-CODE
1811 010400 004767 000770      JSR    PC,EXEC      ;GO DO WRITE 1
1812 010404 012767 015072 170312 MOV    #MSG12,ERRP   ;SET ERROR CODE
1813 010412 004767 001206      JSR    PC,ERCHK     ;GO CHECK FOR ERROR
1814 010416 004767 000752      JSR    PC,EXEC      ;YES DO WRITE 2
1815 010422 004767 001176      JSR    PC,ERCHK     ;YES CHECK FOR ERROR
1816 010426 000240      NOP
1817 010430 004767 001072      JSR    PC,RWND      ;GO REWIND
1818 010434 012767 177160 170254 MOV    #-620,FCNT    ;SET FC=400
1819 010442 012767 177470 170250 MOV    #-310,WCNT    ;SET WC=200
1820 010450 004767 000720      JSR    PC,EXEC      ;GO REWRITE RECORD 1-WH TO EH
1821 010454 000240      NOP
1822 010456 004767 001044      JSR    PC,RWND      ;REWIND
1823 010462 012767 021342 170224 MOV    #RDATA,BADDR ;SET BA=READ BUFFER
1824 010470 012767 177160 170220 MOV    #-620,FCNT    ;SET FC=400
1825 010476 012767 177470 170214 MOV    #-310,WCNT    ;SET WC=200
1826 010504 012767 000071 170274 MOV    #71,FUN       ;SET READ OP-CODE
1827 010512 004767 000656      JSR    PC,EXEC      ;GO READ RECORD 1
1828 010516 012767 015135 170200 MOV    #MSG14,ERRP   ;SET ERROR CODE
1829 010524 004767 001074      JSR    PC,ERCHK     ;GO CHECK FOR ERROR
1830 010530 000240      NOP
1831 010532 052777 000010 170050 BIS     #10,ACS       ;INHIBIT BA INCREMENT
1832 010540 012767 176340 170150 MOV    #-1440,FCNT    ;SET FC=800
1833 010546 012767 177160 170144 MOV    #-620,WCNT    ;SET WC=400
1834 010554 004767 000614      JSR    PC,EXEC      ;GO READ RECORD 2
1835 010560 022777 001440 170020 CMP     #1440,AF C    ;SEE IF READ RECORD 2
1836 010566 001423      BEQ     FT21X      ;IF SO BR
1837 010570 022777 001441 170010 CMP     #1441,AF C    ;++F CHECK FOR 801 FRAMES
1838 010576 001417      BEQ     FT21X      ;++F IF SO BR
1839 010600 012767 016112 170140 MOV    #MSG45,ERADD   ;++F SET ERROR CODE
1840 010606 022777 001440 167772 CMP     #1440,AF C    ;++F MORE THAN 801 FRAMES ?
1841 010614 101403      BLOS     1$        ;++F IF SO BR
1842 010616 012767 016043 170122 MOV    #MSG44,ERADD   ;++F SET ERROR CODE
1843 010624 012767 010454 170140 1$ MOV    #FT21A,SCOLP   ;SET SCOPE ADDRESS
1844 010632 004767 173334      JSR    PC,FT3ER     ;GO PRINT ERROR
1845 010636 004767 001606      JSR    PC,ITER     ;GO SEE IF ITERATION
1846 010642 000167 172270      JMP     TSCD2      ;RETURN TO SCHEDULAR
1847
1848

```



```

1849                                     , BUFFERED COMMAND TEST*****
1850
1851 010646 012767 017455 170032 FT22· MOV #MSFT22,EMADDR ; SET TEST HEADER
1852 010654 004767 000646 JSR PC,RWNO ; GO REWIND
1853 010660 012700 000003 MOV #3,RO ; SET NUMBER OF WRITES
1854 010664 012767 001700 170122 MOV #1700,UDS ; SET TO NRZ NORMAL
1855 010672 012767 017630 170014 MOV #WDATA,BADDR ; SET BA=WRITE BUFFER
1856 010700 012767 177000 170010 MOV #-1000,FCNT ; SET FC=1000
1857 010706 012767 177400 170004 MOV #-400,WCNT ; SET WC=400
1858 010714 012767 000061 170064 MOV #61,FUN ; SET WRITE OP-CODE
1859 010722 004767 000446 FT22A JSR PC,EXEC ; GO DO WRITE
1860 010726 005300 DEC RO ; SEE IF DONE ALL
1861 010730 001374 BNE FT22A ; IF NOT BR
1862 010732 000240 NOP
1863 010734 012777 000007 167636 MOV #7,DC1 ; START REWIND
1864 010742 032777 000200 167642 FT22B BIT #200,ADS
1865 010750 001774 BEQ FT22B
1866 010752 004767 001550 JSR PC,INIT1 ; INITIALIZE
1867 010756 012767 000010 167750 MOV #10,RDYDX ; SET LONG READY DELAY
1868 010764 004767 000404 JSR PC,EXEC ; ISSUE BUFFERED WRITE
1869 010770 000240 NOP
1870 010772 012767 016243 167724 MOV #MSG49,ERRP ; SET ERROR CODE
1871 011000 012767 102300 167750 MOV #102300,STMSK ; MARK DATA ERROR
1872 011006 004767 000612 JSR PC,ERCHK ; GO CHECK ERROR
1873 011012 032777 000002 167572 BIT #2,ADS ; SEE IF BOT IS SET
1874 011020 001410 BEQ FT22X ; IF NOT BR
1875 011022 012767 016271 167716 MOV #MSG50,ERADD ; SET ERROR CODE
1876 011030 012767 010646 167734 MOV #FT22,SCOLP
1877 011036 004767 173130 JSR PC,FT3ER ; GO DO ERROR
1878 011042 004767 001402 FT22X JSR PC,ITER ; GO SEE IF ITEPATION
1879 011046 000167 172064 JMP TSCD2 ; RETURN TO SCHEDULAR
1880
1881

```

```

1882                                     ,READ-IN PRESET TEST*****
1883
1884 011052 005767 167634          FT23  TST      SLVN      ,SEE IF SLAVE SELECT=0
1885 011056 001103                BNE      FT23X      ,IF NOT BR
1886 011060 012767 017512 167620  MOV      #MSFT23,EMADDR ,SET TEST HEADER
1887 011066 004767 001434                JSR      PC,INIT1 ,GO INIT
1888 011072 012767 001700 167714  MOV      #1700,UDES ,SET TO NRZ NORMAL
1889 011100 012767 017630 167606  MOV      #WDATA,BADDR ,SET BA=WRITE BUFFER
1890 011106 012767 177400 167602  MOV      #-400,FCNT ,SET FC=400
1891 011114 012767 177600 167576  MOV      #-200,WCNT ,SET WC=200
1892 011122 012767 000061 167656  MOV      #61,FUN ,SET WRITE OP-CODE
1893 011130 004767 000240                JSR      PC,EXEC ,GO DO WRITE
1894 011134 000240                NOP
1895 011136 004767 001364                JSR      PC,INIT1 ,INITIALIZE
1896 011142 012767 000021 167636  MOV      #21,FUN ,SET READ-IN PRESET OP CODE
1897 011150 004767 000220                JSR      PC,EXEC ,GO DO COMMAND
1898 011154 005000                CLR      R0
1899 011156 012703 000004                MOV      #4,R3 ,SET MULT
1900 011162 032777 020C00 167422  FT23A  BIT      #20000,ADS ,SEE IF PIP RESET
1901 011170 001404                BEQ      FT23B ,IF SO BR
1902 011172 005300                DEC      R0
1903 011174 001372                BNE      FT23A ,AWAIT PIP RESET
1904 011176 005303                DEC      R3
1905 011200 001370                BNE      FT23A ,DELAY
1906 011202 032777 000002 167402  FT23B  BIT      #2,ADS ,SEE IF BOT
1907 011210 001010                BNE      FT23C ,IF SO BR
1908 011212 012767 016327 167526  MOV      #MSG51,ERADD ,SET ERROR CODE
1909 011220 012767 011052 167544  MOV      #FT23,SCOLP
1910 011226 004767 172740                JSR      PC,FT3ER ,GO DO ERROR
1911 011232 012701 141000          FT23C  MOV      #141000,R1 ,SET EXPT TC
1912 011236 016700 167370                MOV      TC,R0 ,SET TC ADDRESS
1913 011242 020110                CMP      R1,(R0) ,SEE IF EXPT=PCVD
1914 011244 001410                BEQ      FT23X ,IF SO BR
1915 011246 012767 016363 167472  MOV      #MSG52,ERADD ,SET ERROR CODE
1916 011254 012767 011052 167510  MOV      #FT23,SCOLP ,CLEAR SCOPE ADDRESS
1917 011262 004767 172414                JSR      PC,FT2ER ,GO DO ERROR
1918 011266 000167 171644          FT23X  JMP      TSCD2 ,RETURN TO SCHEDULAR
1919
1920

```

```

1921                                     ,REWIND OFF LINE TEST*****
1922
1923 011272 032777 004000 167340 FT24 BIT #4000, @SWR ,SEE IF IN CONTINUOUS MODE
1924 011300 001033 BNE FT24XX ,IF SO. BR
1925 011302 012767 017545 167376 MOV #MSFT24, EMADDR ,SET TEST HEADER
1926 011310 004767 001212 JSR PC, INIT1 ,GO INITIAIZE
1927 011314 012777 000003 167256 MOV #3, @C1 ,ISSUE REWIND OFF LINE COMMAND
1928 011322 012700 004000 MOV #4000, R0
1929 011326 005300 FT24A DEC R0 ,DELAY
1930 011330 001376 BNE FT24A
1931 011332 032777 010000 167252 BIT #10000, @DS ,SEE IF MOL IS RESET
1932 011340 001407 BEQ FT24X ,IF SO. BR
1933 011342 005067 167424 CLR SCOLP ,ASSURE NO SCOPE
1934 011346 012767 016402 167372 MOV #MSG53, ERADD ,SET ERROR CODE
1935 011354 004767 172612 JSR PC, FT3ER ,GO DO ERROR
1936 011360 012704 016426 FT24X MOV #MSG54, R4
1937 011364 004767 001564 JSR PC, TTOUT ,PRINT ON LINE REQUEST
1938 011370 000167 171542 FT24XX JMP TSCD2 ,RETURN TO SCHEDULAR
1939
1940
  
```

```

1941                                     ;COMMAND EXECUTE SUBROUTINE*****
1942
1943 011374 000240                               EXEC  NOP
1944 011376 056777 167412 167226                BIS    UDES, @TC      , LOAD TAPE CONT
1945 011404 016777 167310 167170                MOV    WCNT, @WC      , LOAD WC
1946 011412 016777 167300 167166                MOV    FCNT, @FC      , LOAD FC
1947 011420 016777 167270 167156                MOV    BADDR, @BA     , LOAD BA
1948 011426 022767 000031 167352                CMP    #31, FUN      , SEE IF SPACE FORWARD
1949 011434 001404                               BEQ    EXECA        , IF SO BR
1950 011436 022767 000033 167342                CMP    #33, FUN      , SEE IF SPACE REVERSE
1951 011444 001003                               BNE    EXECB        , IF NOT BR
1952 011446 016777 167266 167132  EXECA  MOV    SCNT, @FC      , SET SPACE COUNT
1953 011454 000240                               EXECB  NOP
1954 011456 016777 167324 167114                MOV    FUN, @C1       , LOAD OP-CODE + GO
1955 011464 000240                               NOP
1956 011466 016703 167242                MOV    RDYDX, R3     , SET DELAY
1957 011472 005004                               CLR    R4
1958 011474 032777 000200 167110  EXECC  BIT    #200, @DS      , SEE IF DRY
1959 011502 001004                               BNE    EXECX        , IF SO BR
1960 011504 005304                               DEC    R4
1961 011506 001372                               BNE    EXECC
1962 011510 005303                               DEC    R3      , DELAY FOR DRY
1963 011512 001370                               BNE    EXECC
1964 011514 016703 167216  EXECX  MOV    OPDYX, R3
1965 011520 005303  EXECXA  DEC    R3      , DELAY
1966 011522 001376                               BNE    EXECXA
1967 011524 000207  EXECXX  RTS    PC      , RETURN TO CALLER
1968
  
```

```

1969                                     , REWIND SUBROUTINE*****
1970
1971 011526 000240                                RWND  NOP
1972 011530 004767 000772                        JSR    PC, INIT1      , INIT
1973 011534 012777 000007 167036                MOV    #7, @C1      , START REWIND
1974 011542 012700 040000                        MOV    #40000, R0
1975 011546 005300                                RWNDA  DEC    R0
1976 011550 001376                                BNE     RWNDA      , DELAY
1977 011552 032777 020000 167032  RWNDB  BIT    #20000, @DS
1978 011560 001374                                BNE     RWNDB      , AWAIT PIP
1979 011562 032777 000002 167022                BIT    #2, @DS      , SEE IF BOT
1980 011570 001012                                BNE     RWNDX      , IF SO BR
1981 011572 016704 167110                        MOV    EMADDR, R4
1982 011576 004767 001352                        JSR    PC, TTOUT      , PRINT HEADER
1983 011602 012704 014606                        MOV    #MSG2, R4
1984 011606 004767 001342                        JSR    PC, TTOUT      , PRINT REWIND ERROR
1985 011612 000167 171320                        JMP    TSCD2      , RETURN TO SECHEDULAR
1986 011616 004767 000704  RWNDX  JSR    PC, INIT1      , INIT
1987 011622 000207                        RTS    PC      , RETURN TO CALLEP
1988

```

```

1989                                     ,ERROR CHECK SUBROUTINE*****
1990
1991 011624 005067 167160 ERCHK CLR SERFL ,CLEAR FLAG
1992 011630 017767 166756 167124 MOV @DS,DSAV ,SAVE DRIVE STATUS REG STER
1993 011636 032777 040000 166746 BIT #40000,@DS ,SEE IF ERROR
1994 011644 001001 BNE ERPT ,IF SO BR
1995 011646 000207 RTS PC ,RETURN
1996 011650 017704 166740 ERPT MOV @ER,R4 ,GET ERROR REGISTER
1997 011654 032767 002000 167132 BIT #2000,UDES ,SEE IF PE
1998 011662 001403 BEQ ERPTA1 ,IF SO BR
1999 011664 042767 000200 167064 BIC #200,STMSK ,RESET PEF MASK
2000 011672 046704 167060 ERPTA1 BIC STMSK,R4 ,MASK DONT CARE BITS
2001 011676 001530 BEQ ERPTX ,IF NO UNEXPECTED ERRORS BR
2002 011700 012767 000001 167102 ERPTG MOV #1,SERFL ,SET FLAG
2003 011706 032777 020000 166724 B T #20000,@SWR ,SEE IF SHOULD PRINT ERRORS
2004 011714 001115 BNE ERPTD ,IF NOT BR
2005 011716 005767 166762 TST HDRFL ,SEE IF DONE HEADER
2006 011722 001006 BNE ERPTA ,IF SO BR
2007 011724 005267 166754 INC HDRFL ,SET HEADER FLAG
2008 011730 016704 166752 MOV EMADDR,R4
2009 011734 004767 001214 JSR PC,TTOUT ,PRINT HEADER
2010 011740 016704 166760 ERPTA MOV ERRP,R4 ,GET ERROR CODE
2011 011744 001414 BEQ ERPTB ,IF NONE BR
2012 011746 004767 001202 JSR PC,TTOUT ,PRINT ERROR CODE
2013 011752 012704 015243 MOV #MSG20,R4 ,SET NRZ TAG
2014 011756 032777 002000 166646 BIT #2000,@TC ,SEE IF PE
2015 011764 001402 BEQ ERPT1A ,IF NOT BR
2016 011766 012704 015251 MOV #MSG21,R4 ,ELSE SET PE TAG
2017 011772 004767 001156 ERPT1A JSR PC,TTOUT ,PRINT TAG
2018 011776 016704 166724 ERPTB MOV ERRP1,R4 ,SEE IF CODE 2
2019 012002 001402 BEQ ERPTB1 ,IF NOT BR
2020 012004 004767 001144 JSR PC,TTOUT ,PRINT CODE 2
2021 012010 032777 010000 166622 ERPTB1 BIT #10000,@SWP ,SEE IF ITERATION
2022 012016 001010 BNE ERPTC ,IF NOT BR
2023 012020 012704 016515 MOV #MSG56,R4
2024 012024 004767 001124 JSR PC,TTOUT ,PRINT ITER TAG
2025 012030 016703 166724 MOV ITCNT,R3
2026 012034 004767 001260 JSR PC,OCTP ,PRINT ITERATION
2027 012040 012704 014520 EPPTC MOV #MSG1,R4
2028 012044 004767 001104 JSR PC,TTOUT ,PRINT REGISTER TAG
2029 012050 017703 166524 MOV @C1,R3
2030 012054 004767 001226 JSR PC,OCTPE ,PRINT C51
2031 012060 017703 166516 MOV @WC,R3
2032 012064 004767 001216 JSR PC,OCTPE ,PRINT WC
2033 012070 017703 166510 MOV @BA,R3
2034 012074 004767 001206 JSR PC,OCTPE ,PRINT BA
2035 012100 017703 166502 MOV @FC,R3
2036 012104 004767 001176 JSR PC,OCTPE ,PRINT FC
2037 012110 017703 166474 MOV @CS,R3
2038 012114 004767 001166 JSR PC,OCTPE ,PRINT CS2
2039 012120 017703 166466 MOV @DS,R3
2040 012124 004767 001156 JSR PC,OCTPE ,PRINT DS
2041 012130 017703 166460 MOV @ER,R3
2042 012134 004767 001146 JSR PC,OCTPE ,PRINT EP
2043 012140 017703 166466 MOV @TC,R3
2044 012144 004767 001136 JSR PC,OCTPE ,PRINT TC

```


2045	012150	005777	166464	ERPTD	TST	QSWR	, SEE IF HALT ON ERROR
2046	012154	100001			BPL	ERPTX	, IF NOT BR
2047	012156	000000			HALT		
2048	012160	004767	001564	ERPTX	JSR	PC, CKSWR	, CHECK FOR CNTL G
2049	012164	004767	000336		JSR	PC, INIT1	, INIT
2050	012170	000207		ERPTXX	RTS	PC	, RETURN
2051							
2052							

```

2053                                     ,TAPE MARK STATUS CHECK*****
2054
2055 012172 032767 000004 166562 TMCHK BIT #4,DSAV ,SEE IF TM SET
2056 012200 001401 BEQ TMCHK1 ;IF NOT BR
2057 012202 000207 TMCHK0 RTS PC ;ELSE RETURN
2058 012204 005767 166600 TMCHK1 TST SERFL ,SEE IF HAD ERROR
2059 012210 001374 BNE TMCHK0 ;IF SO BR
2060 012212 012767 016525 166506 MOV #MSG57,ERRP1 ,SET ERROR CODE 2
2061 012220 004767 177454 JSR PC,ERPTG ;GO PRINT TM ERROR
2062 012224 005067 166476 CLR ERRP1 ;CLEAR CODE 2 FLAG
2063 012230 000207 RTS PC ;RETURN
2064
2065                                     ,DATA SETUP ROUTINE*****
2066
2067 012232 000240 DSUP NOP
2068 012234 012703 017630 DSO MOV #WDATA,R3 ,R3 = ADDRS OF WRITE BUFFER
2069 012240 016701 166552 MOV PATRN,R1 ,R1 = PATTERN SELECTOR
2070 012244 000241 CLC
2071 012246 006101 ROL R1 ,MAKE PATTERN SELECTOR EVEN
2072 012250 000171 001040 JMP @DATBL(R1) ,GO GENERATE PATTERN
2073 012254 032777 010000 166344 DS1 BIT #10000,@DT ,SEE IF SEVEN TRACK
2074 012262 001410 BEQ DS3 ;IF NOT BR
2075 012264 012702 000640 MOV #640,R2 ;SET BUFFER SIZE
2076 012270 012701 017630 MOV #WDATA,R1 ,SET START OF BUFFER
2077 012274 042721 140300 DS2 BIC #140300,(R1)+ ,MASK FOR 7 CH
2078 012300 005302 DEC R2 ,SEE IF DONE
2079 012302 001374 BNE DS2 ;IF NOT BR
2080 012304 012702 000640 DS3 MOV #640,R2 ,R2=BUFFER SIZE +2
2081 012310 012701 021342 MOV #RDATA,R1 ,R1=READ DATA START
2082 012314 005021 DS4 CLR (R1)+ ,CLEAR BUFFER
2083 012316 005302 DEC R2 ,SEE IF DONE ALL
2084 012320 001375 BNE DS4 ;IF NOT BR
2085 012322 000207 RTS PC ;EXIT
2086
2087                                     ,ALL ONES*****
2088
2089 012324 012701 177777 DAT1 MOV #-1,R1 ,R1=DATA
2090 012330 012702 000640 DAT1A MOV #640,R2 ,R2=WORD COUNT +2
2091 012334 010123 DAT1B MOV R1,(R3)+ ,LOAD BUFFER
2092 012336 005302 DEC R2 ,SEE IF DONE
2093 012340 001375 BNE DAT1B ,IF NOT BR
2094 012342 000167 177706 JMP DS1 ,RETURN
2095
2096                                     ,ALL ZEROS*****
2097
2098 012346 005001 DAT2 CLR R1 ,R1=DATA
2099 012350 000167 177754 JMP DAT1A ,LOAD BUFFER
2100

```


2101						, ONE/ZERO IN ALTERNATING CHARACTERS*****
2102						
2103	012354	012701	125125	DAT3	MOV	#125125,R1 ; R1=DATA
2104	012360	000167	177744		JMP	DAT1A ; LOAD BUFFER
2105						
2106						, ALL BITS 0-377*****
2107						
2108	012364	005001		DAT4	CLR	R1 ; R1=STARTING DATA
2109	012366	012702	001500		MOV	#1500,R2 ; R2=CHARACTER COUNT
2110	012372	110123		DAT4A	MOVB	R1,(R3)+ ; LOAD BUFFER
2111	012374	105201			INCB	R1 ; BUMP DATA
2112	012376	005302			DEC	R2 ; SEE IF DONE
2113	012400	001374			BNE	DAT4A ; IF NOT BR
2114	012402	000167	177646		JMP	DS1 ; RETURN
2115						
2116						
2117						, SCOPE LOOP ON ERROR SUBROUTINE*****
2118						
2119	012406	004767	001336	SCOPE	JSR	PC,CKSWR ; CHECK FOR CNTL G
2120	012412	000240			NOP	
2121	012414	032777	040000 166216		BIT	#40000,@SWR ; SEE IF LOOP ON ERROR
2122	012422	001001			BNE	SCOPE1 ; IF SO BR
2123	012424	000207			RTS	PC ; ELSE EXIT
2124	012426	000240		SCOPE1	NOP	
2125	012430	005767	166336		TST	SCOLP ; SEE IF SCOPE ADDRESS
2126	012434	001001			BNE	SCOPE2 ; IF NOT BR
2127	012436	000207			RTS	PC ; ELSE EXIT
2128	012440	005726		SCOPE2	TST	(SP)+ ; RESET STACK
2129	012442	005726			TST	(SP)+
2130	012444	000177	166322		JMP	@SCOLP ; LOOP ON ERROR
2131						
2132						, TEST ITERATION SUBROUTINE*****
2133						
2134	012450	004767	001274	ITER	JSR	PC,CKSWR ; CHECK FOR CNTL G
2135	012454	000240			NOP	
2136						
2137						*****
2138						CHECK FOR END OF PASS INDICATOR
2139						SET FOR QUICK VERIFY
2140						
2141	012456	005767	166354		TST	PAFLG ; PASS FLAG SET?
2142	012462	001404			BEQ	ITERO ; BRANCH - IF NO
2143						*****
2144						
2145	012464	032777	010000 166146		BIT	#10000,@SWR ; SEE IF ITERATIONS
2146	012472	001403			BEQ	ITER1 ; IF SO BR
2147	012474	005067	166260	ITERO	CLR	ITCNT ; CLEAR ITERATION COUNTER
2148	012500	000207			RTS	PC ; ELSE EXIT
2149	012502	005267	166252	ITER1	INC	ITCNT ; BUMP COUNTER
2150	012506	026767	166246 166142		CMP	ITCNT,ITAMT ; SEE IF DONE ALL
2151	012514	001767			BEQ	ITERO ; IF SO BR
2152	012516	005726			TST	(SP)+ ; RESET STACK
2153	012520	017700	166250		MOV	@ITRLP,RO ; SET ITERATION POINTER
2154	012524	000110			JMP	(RO) ; GO ITERATE
2155						
2156						, INITIALIZE SUBROUTINE*****


```

2164                                     ,MAG TAPE INTERRUPT HANDLER*****
2165
2166 012554 000240          MTINT  NOP
2167 012556 022626          CMP      (SP)+,(SP)+      ,RESET STACK POINTER
2168 012560 000240          NOP
2169 012562 000240          NOP
2170 012564 000177 166154    JMP      @RTN          ,RETURN TO CALLER
2171
2172                                     ,TTY INTERRUPT HANDLER*****
2173
2174 012570 000240          TTINT  NOP
2175 012572 000240          NOP
2176 012574 000240          NOP
2177 012576 000002          RTI
2178
2179                                     ,BUS ADDRESS TRAP HANDLER*****
2180
2181 012600 000240          TRAP   NOP
2182 012602 032777 020000 166030  BIT      #20000,@SWR      ;SEE IF SHOULD PRINT ERRORS
2183 012610 001020          BNE     TRAP2          ,IF NOT: BR
2184 012612 005767 166066    TST      HDRFL          ,SEE IF DONE HEADER
2185 012616 001006          BNE     TRAP1          ,IF SO: BR
2186 012620 005267 166060    INC      HDRFL          ,ELSE SET HEADER FLAG
2187 012624 016704 166056    MOV      EMADDR,R4
2188 012630 004767 000320          JSR      PC,TTOUT      ,PRINT HEADER
2189 012634 012704 015276    TRAP1  MOV      #MSG24,R4
2190 012640 004767 000310          JSR      PC,TTOUT      ;PRINT ERROR
2191 012644 010103          MOV      R1,R3
2192 012646 004767 000446          JSR      PC,OCTP      ,PRINT ADDRESS OF TRAP
2193 012652 005777 165762    TRAP2: TST      @SWR          ;SEE IF HALT ON ERROR
2194 012656 100001          BPL     TRAPX          ;IF NOT: BR
2195 012660 000000          HALT
2196 012662 004767 001062    TRAPX  JSR      PC,CKSWR          ,CHECK FOR CNTL G
2197 012666 022626          CMP      (SP)+,(SP)+      ;RESET STACK
2198 012670 012767 003444 166074    MOV      #FT1A,SCOLP      ,SET SCOPE ADDRESS
2199 012676 004767 177504          JSR      PC,SCOPE      ,GO SEE IF SCOPE LOOP
2200 012702 005767 166112          TST      RHTF          ,SEE IF INITIAL ADDRESS TEST
2201 012706 001402          BEQ     TRAPXX          ,IF NOT BR
2202 012710 000167 167062          JMP      STOB          ,ELSE REDO ADDRESS REQUEST
2203 012714 000167 170530    TRAPXX JMP      FT1B          ,RETURN TO TEST 1
2204

```

```

2205 ,*****
2206 ,TTY ENTRY SUBROUTINE:
2207 ,
2208 ,THIS SUBROUTINE IS USED BY THE TEST CONDITION
2209 ,ENTRY ROUTINE TO READ THE RESPONSE ENTERED
2210 ,AT THE TTY AND CHECK THEM FOR LEGALITY AND
2211 ,LIMITS. ALL RESPONSE MUST BE TYPED IN OCTAL
2212 ,(0-7) AND MUST FALL WITHIN THE LIMITS SET BY
2213 ,THE CALLING ROUTINE
2214 ,IF AN ENTRY IS ILLEGAL OR OUTSIDE THE LIMITS,
2215 ,A QUESTION MARK IS TYPED (?) AND THE RESPONSE
2216 ,MAY BE REENTERED
2217 ,ENTRIES MAY NOT EXCEED SIX (6) CHARACTERS AND
2218 ,MAY BE TERMINATED AT LESS THAN SIX BY TYPING A
2219 ,CARRIAGE RETURN
2220 ,*****
2221
2222 012720 005067 166024 TTR CLR TEMP1 ,CLEAR FIRST CHARACTER FLAG
2223 012724 005000 CLR RO
2224 012726 004767 000152 TTR0 JSR PC,TTIN ,GO READ CHARACTER
2225 012732 122767 000015 165740 CMPB #15,T1B ,SEE IF CR
2226 012740 001005 BNE TTR1 ,IF NOT BR
2227 012742 005767 166002 TST TEMP1 ,SEE IF FIRST CHARACTER
2228 012746 001446 BEQ TTR5 ,IF SO BR
2229 012750 000167 000066 JMP TTR2 ,ELSE GO LOAD VALUE
2230 012754 122767 000060 165716 TTR1 CMPB #60,T1B ,SEE IF CHAR IS LESS THAN 0
2231 012762 101402 BLOS TTR1A ,IF NOT BR
2232 012764 000167 000076 JMP T1NER ,ELSE GO TO ERROR
2233 012770 122767 000070 165702 TTR1A CMPB #70,T1B ,SEE IF CHAR IS GREATER THAN 7
2234 012776 101002 BHI TTR1B ,IF NOT BR
2235 013000 000167 000062 JMP T1NER ,ELSE GO TO ERROR
2236 013004 005267 165740 TTR1B INC TEMP1 ,SET FIRST CHARACTER FLAG
2237 013010 000241 CLC
2238 013012 006100 ROL RO
2239 013014 000241 CLC
2240 013016 006100 ROL RO ,SHIFT 3 LEFT
2241 013020 000241 CLC
2242 013022 006100 ROL RO
2243 013024 042767 177770 165646 BIC #177770,T1B ,STRIP ASCII
2244 013032 056700 165642 BIS T1B,RO ,LOAD CHARACTER
2245 013036 005301 DEC R1 ,SEE IF DONE
2246 013040 001332 BNE TTR0 ,IF NOT BR
2247 013042 020002 TTR2 CMP RO,R2 ,SEE IF EXCEEDED MAXIMUM LIMIT
2248 013044 101402 BLOS TTR3 ,IF NOT BR
2249 013046 000167 000014 JMP T1NER ,ELSE GO TO ERROR
2250 013052 020300 TTR3 CMP R3,RO ,SEE IF BELOW MINIMUM LIMIT
2251 013054 101402 BLOS TTR4 ,IF NOT BR
2252 013056 000167 000004 JMP T1NER ,ELSE GO TO ERROR
2253 013062 010015 TTR4 MOV RO,(R5) ,LOAD VALUE
2254 013064 000207 TTR5 RTS PC ,EXIT
2255

```

```

2256                                     ,TTY ENTRY ERROR SUBROUTINE*****
2257
2258 013066 012704 015002          T1NER  MOV      #MSG7,R4
2259 013072 004767 000056          JSR      PC,TTOUT      ;PRINT?
2260 013076 162716 000020          SUB      #20,(SP)      ;RESET SP TO START OF VALUE ROUTINE
2261 013102 000207                  RTS      PC           ;REDO VALUE ENTRY
2262
2263                                     ,TTY READ SUBROUTINE*****
2264
2265 013104 005077 165532          TTIN   CLR      @TKS
2266 013110 005077 165530          CLR      @TKB
2267 013114 105777 165522          TSTB     @TKS
2268 013120 100375                  BPL      -4
2269 013122 017767 165516 165550      MOV      @TKB,T1B
2270 013130 042767 177600 165542      BIC      #177600,T1B
2271 013136 105777 165504          TTIN2  TSTB     @TPS
2272 013142 100375                  BPL      TTIN2
2273 013144 116777 165530 165476      MOVB     T1B,@TPB
2274 013152 000207                  RTS      PC
2275
2276                                     ,TTY OUTPUT SUBROUTINE*****
2277
2278 013154 112467 165516          TTOUT.  MOVB     (R4)+,TOB
2279 013160 122767 000043 165510      CMPB     #43,TOB
2280 013166 001446                  BEQ      TEX
2281 013170 122767 000045 165500      CMPB     #45,TOB
2282 013176 001403                  BEQ      TCRLF
2283 013200 004767 000064          JSR      PC,TOG
2284 013204 000763                  BR       TTOUT
2285 013206 112767 000015 165462      TCRLF   MOVB     #15,TOB
2286 013214 004767 000050          JSR      PC,TOG
2287 013220 012703 000004          MOV      #4,R3
2288 013224 005067 165446          TCRLFA  CLR      TOB
2289 013230 004767 000034          JSR      PC,TOG
2290 013234 005303                  DEC      R3
2291 013236 001372                  BNE      TCRLFA      ,DO FILLERS
2292 013240 112767 000012 165430      MOVB     #12,TOB
2293 013246 004767 000016          JSR      PC,TOG
2294 013252 105767 165556          TSTB     RDSW
2295 013256 100401                  BMI      15
2296 013260 000735                  BR       TTOUT
2297 013262 005067 165546          15      CLR      RDSW
2298 013266 000406                  BR       TEX
2299 013270 105777 165352          TOG     TSTB     @TPS
2300 013274 100375                  BPL      TOG
2301 013276 116777 165374 165344      MOVB     TOB,@TPB
2302 013304 000207                  TEX     RTS      PC
2303
2304

```

```

2305 ;OCTAL OUTPUT SUBROUTINE*****
2306
2307 013306 012767 000001 000222 OCTPE MOV #1,OFL
2308 013314 010304 MOV R3,R4
2309 013316 000410 BR OCTP0
2310 013320 005067 000212 OCTP CLR OFL ;CLEAR FLAG FOR LEADING ZERO
2311 013324 010304 OCTPE1 MOV R3,R4 ;SEE IF NUMBER IS ZERO
2312 013326 001004 BNE OCTP0 ;IF NOT ZERO, BR
2313 013330 004767 000162 JSR PC,OCTPG1 ;ELSE PRINT ZERO
2314 013334 000167 000120 JMP OCTP3 ;SPACE AND EXIT
2315 013340 032704 100000 OCTP0 BIT #100000,R4 ;SEE IF MSD = 1
2316 013344 001406 BEQ OCTP1 ;IF NOT BR
2317 013346 012704 000001 MOV #1,R4
2318 013352 004767 000116 JSR PC,OCTPG ;PRINT 1
2319 013356 000167 000006 JMP OCTP2
2320 013362 005004 OCTP1 CLR R4
2321 013364 004767 000104 JSR PC,OCTPG ;PRINT 0
2322 013370 010304 OCTP2 MOV R3,R4
2323 013372 006004 ROR R4
2324 013374 006004 ROR R4
2325 013376 006004 ROR R4 ;POSITION DIGIT
2326 013400 006004 ROR R4
2327 013402 000304 SWAB R4
2328 013404 004767 000064 JSR PC,OCTPG ;PRINT DIGIT 2
2329 013410 010304 MOV R3,R4
2330 013412 006004 ROR R4
2331 013414 000304 SWAB R4
2332 013416 004767 000052 JSR PC,OCTPG ;PRINT DIGIT 3
2333 013422 010304 MOV R3,R4
2334 013424 006104 ROL R4
2335 013426 006104 ROL R4
2336 013430 000304 SWAB R4
2337 013432 004767 000036 JSR PC,OCTPG ;PRINT DIGIT 4
2338 013436 010304 MOV R3,R4
2339 013440 006004 ROR R4
2340 013442 006004 ROR R4
2341 013444 006004 ROR R4
2342 013446 004767 000022 JSR PC,OCTPG
2343 013452 010304 MOV R3,R4
2344 013454 004767 000014 JSR PC,OCTPG ;PRINT DIGIT 5
2345 013460 012767 000240 165210 OCTP3 MOV #240,T0B
2346 013466 004767 177576 JSR PC,T0B ;PRINT SPACE
2347 013472 000207 RTS PC ;EXIT
2348 013474 042704 177770 OCTPG BIC #177770,R4
2349 013500 001004 BNE OCTPG0
2350 013502 005767 000030 TST OFL
2351 013506 001001 BNE OCTPG0
2352 013510 000207 RTS PC
2353 013512 005267 000020 OCTPG0 INC OFL
2354 013516 052704 000260 OCTPG1 BIS #260,R4
2355 013522 010467 165150 MOV R4,T0B
2356 013526 004767 177536 JSR PC,T0B
2357 013532 010304 MOV R3,R4
2358 013534 000207 RTS PC
2359 013536 000000 OFL 0 ;FIRST CHAR FLAG
2360

```



```

2361                                     ,DATA CHARACTER OUTPUT SUBROUTINE*****
2362
2363 013540 005067 165132          DOUT CLR      TOB
2364 013544 012704 000010          MOV      #10,R4          ,SET NUMBER TO PRINT
2365 013550 110367 165122          MOV8     R3,TOB
2366 013554 105777 165066          DOUT1  TSTB     @TPS
2367 013560 100375          BPL      DOUT1
2368 013562 132767 000200 165106  BITB     #200,TOB
2369 013570 001404          BEQ      DOUT2
2370 013572 012777 000061 165050  MOV      #061,@TPB
2371 013600 000403          BR       DOUT3
2372 013602 012777 000060 165040  DOUT2  MOV      #060,@TPB
2373 013610 006167 165062          DOUT3  ROL      TOB
2374 013614 005304          DEC      R4
2375 013616 001356          BNE      DOUT1
2376 013620 000207          RTS      PC
2377 013622 016703 165126          DOUTD  MOV      TEMP3,R3
2378 013626 000303          SWAB     R3
2379 013630 004767 177704          JSR      PC,DOUT
2380 013634 016703 165114          MOV      TEMP3,R3
2381 013640 004767 177674          JSR      PC,DOUT
2382 013644 000207          RTS      PC
2383
2384                                     ,TU16/TE16 SERIAL NUMBER PRINT SUBROUTINE*****
2385
2386 013646 010304          SNPT  MOV      R3,R4
2387 013650 000304          SWAB     R4
2388 013652 006004          ROR      R4
2389 013654 006004          ROR      R4
2390 013656 006004          ROR      R4
2391 013660 006004          ROR      R4          ,GET FIRST DIGIT
2392 013662 004767 000036          JSR      PC,SNPG          ,GO PRINT
2393 013666 010304          MOV      R3,R4
2394 013670 000304          SWAB     R4          ,GET SECOND DIGIT
2395 013672 004767 000026          JSR      PC,SNPG          ,GO PRINT
2396 013676 010304          MOV      R3,R4
2397 013700 006004          ROR      R4
2398 013702 006004          ROR      R4
2399 013704 006004          ROR      R4
2400 013706 006004          ROR      R4          ,GET THIRD DIGIT
2401 013710 004767 000010          JSR      PC,SNPG          ,GO PRINT
2402 013714 010304          MOV      R3,R4          ,GET FOURTH DIGIT
2403 013716 004767 000002          JSR      PC,SNPG          ,GO PRINT
2404 013722 000207          RTS      PC          ,EXIT
2405 013724 012767 000260 164744  SNPG  MOV      #260,TOB          ,SET BASE = 0
2406 013732 042704 177760          BIC      #177760,R4          ,MASK DIGIT
2407 013736 050467 164734          BIS      R4,TOB          ,SET ASCII
2408 013742 004767 177322          JSR      PC,TOG          ,TYPE DIGIT
2409 013746 000207          RTS      PC          ,RETURN
2410
2411                                     ,CKSWR ROUTINE THAT ALLOWS THE LOADING OF LOC 176, SWREG*****
2412                                     ,FROM THE TTY AT SELECTED POINTS WITHIN THE PROGRAM*****
2413
2414 013750 022767 000176 164662  CKSWR CMP      #SWREG,SWR          ,SOFTWARE SWITCH REG PRESENT
2415 013756 001041          BNE      OUT          ,NO, GET OUT
2416 013760 105777 164656          TSTB     @TKS          ,YES, WAIT FOR
  
```

2417	013764	100036			BPL	OUT	, READY, GET CHARACTER
2418	013766	017767	164652	164704	MOV	@TKB, TIB	, AND STRIP OFF
2419	013774	042767	177600	164676	BIC	#177600, TIB	, THE GARBAGE
2420	014002	022767	000007	164670	CMP	#7, TIB	, IS IT A ' G>
2421	014010	001024			BNE	OUT	
2422	014012	012704	017601		MOV	#SCNTG, R4	
2423	014016	004767	177132		JSR	PC, TTOUT	
2424	014022	012704	017605		MOV	#MSWR, R4	
2425	014026	004767	177122		JSR	PC, TTOUT	
2426	014032	017703	164602		MOV	@SWR, R3	
2427	014036	004767	177244		JSR	PC, OCTPE	
2428	014042	012704	017614		MOV	#MNEW, R4	
2429	014046	004767	177102		JSR	PC, TTOUT	
2430	014052	005037	001030		CLR	@TEMPST	
2431	014056	004767	000002		JSR	PC, SREAD	, GO READ A LINE
2432	014062	000207			OUT	RTS	, RETURN TO MAIN BODY OF PROGRAM
2433							
2434	014064	005067	164740		SREAD	CLR	TEMPST
2435	014070	012767	000007	164734	MOV	#7, COUNT	
2436	014076	004767	177002		JSR	PC, TTIN	, GO READ A CHARACTER
2437	014102	042767	177600	164570	BIC	#177600, TIB	, STRIP OFF GARBAGE
2438	014110	122767	000025	164562	CMPB	#25, TIB	, IS IT A U?
2439	014116	001002			BNE	25	, BRANCH IF NOT
2440	014120	005726			TST	(SP)+	, POP THE STACK
2441	014122	000737			BR	CNTLU	, START OVER
2442	014124	122767	000015	164546	25	CMPB	#15, TIB
2443	014132	001013			BNE	45	, IS IT A <CR>?
2444	014134	012767	000200	164672	MOV	#200, RDSW	, BRANCH IF NOT
2445	014142	004767	177040		JSR	PC, TCRLF	
2446	014146	022767	000007	164656	CMP	#7, COUNT	, ECHO IT WITH <LF>
2447	014154	001037			BNE	75	, WAS IT FIRST CHARACTER
2448	014156	005726			TST	(SP)+	, CHANGE SWR IF NOT FIRST ONE
2449	014160	000740			BR	OUT	, POP THE STACK
2450	014162	122767	000060	164510	45	CMPB	#60, TIB
2451	014170	003004			BGT	55	, GET OUT
2452	014172	122767	000067	164500	CMPB	#67, TIB	
2453	014200	003005			BGT	65	
2454	014202	012704	017624		55	MOV	#SQUEST, R4
2455	014206	004767	176742		JSR	PC, TTOUT	
2456	014212	000742			BR	35	, START OVER IF NOT LEGAL CHARACTER
2457	014214	006367	164610		65	ASL	TEMPST
2458	014220	006367	164604		ASL	TEMPST	
2459	014224	006367	164600		ASL	TEMPST	
2460	014230	142767	000060	164442	BICB	#60, TIB	, GET NITTY-GRITTY
2461	014236	156767	164436	164564	BISB	TIB, TEMPST	
2462	014244	005367	164562		DEC	COUNT	, ONLY WANT 6 DIGITS
2463	014250	001754			BEQ	55	
2464	014252	000711			BR	15	
2465	014254	016777	164550	164356	75	MOV	TEMPST, @SWR
2466	014262	000735			BR	85	, CHANGE SWITCH REGISTER CONTENTS
2467							


```
2468
2469
2470
2471
2472
2473 014264 005067 164400 CKMODE CLR AUTOM ; INIT AUTO MODE
2474 014270 005737 000042 TST @#42 ; AUTO MODE?
2475 014274 001417 BEQ 2$ ; BRANCH - IF NOT
2476 014276 005267 164366 INC AUTOM ; SET AUTO MODE INDICATORE
2477 014302 023737 000042 000046 CMP @#42,@#46 ; ACT11 MODE?
2478 014310 001403 BEQ 1$ ; BRANCH - IF YES
2479 014312 105267 164355 INCB XXDPM ; INDICATE XXDP AUTO MODE
2480 014316 000421 BR 5$ ; AND EXIT
2481 014320 105267 164346 1$ INCB ACT11M ; INDICATE ACT11 AUTO MODE
2482 014324 052777 104000 164306 BIS #104000,@SWR ; SET FOR CON CYCLE & HALT ON ERROR
2483 014332 000413 BR 5$ ; AND EXIT
2484 014334 105737 000041 2$ TSTB @#41 ; MAN: MODE VIA ACT11/PAPER TAPE?
2485 014340 001003 BNE 3$ ; BRANCH - IF NOT
2486 014342 105267 164326 INCB ADUMPM ; INDICATE MAN MODE VIA ACT11/PAPER TAPE
2487 014346 000402 BR 4$ ; AND EXIT THRU M I
2488 014350 105267 164321 3$ INCB XDUMPM ; INDICATE MANUAL MODE VIA XXDP
2489 014354 052737 020000 000052 4$ BIS #20000,@#52 ; ALLOW MANUAL INTERVENTION
2490 014362 000207 5$ RTS PC ; RETURN
2491
2492
2493
```

```

2494
2495
2496
2497
2498
2499
2500 014364 ABORT
2501 014364 000005 RESET
2502 014366 012704 014476 MOV #MSGD,R4 , INITIALIZE THE WORLD
2503 014372 004767 176556 JSR PC,TTOUT , GET ABORT MESSAGE
2504 014376 105767 164271 TSTB XXDPM , TYPE ABORT MESSAGE
2505 014402 001405 BEQ 1$ , XXDP AUTO MODE?
2506 014404 013700 000042 MOV @#42,R0 , BRANCH - IF NOT
2507 014410 005037 000042 CLR @#42 , GET MONITOR EXIT ADDRESS
2508 014414 004710 JSR PC,(R0) , USE AS ABORT FLAG
2509 014416 000777 BR , EXIT TO XXDP MONITOR
2510
2511

```

DISCONTINUE TESTING FOR
ILLEGAL CONDITIONS

2512						MESSAGE TABLE*****
2513						*****
2514	014420	050045	047522	051107	MSGC	ASCII /%PROGRAM DISABLED ? NO MANUAL INTERVENTION %#/
2515	014426	046501	042040	051511		
2516	014434	041101	042514	020104		
2517	014442	037440	047040	020117		
2518	014450	040515	052516	046101		
2519	014456	044440	052116	051105		
2520	014464	042526	052116	047511		
2521	014472	027116	021445			
2522	014476	050045	047522	051107	MSGD	ASCII /%PROGRAM ABORTED%#/
2523	014504	046501	040440	047502		
2524	014512	052122	042105	021445		
2525						
2526						*****
2527	014520	041445	030523	020040	MSG1	ASCII /%CS1 WC BA FC CS2 /
2528	014526	020040	041527	020040		
2529	014534	020040	041040	020101		
2530	014542	020040	020040	041506		
2531	014550	020040	020040	041440		
2532	014556	031123	020040	020040		
2533	014564	051504	020040	020040		ASCII /DS ER TC%#/
2534	014572	042440	020122	020040		
2535	014600	020040	041524	021445		
2536	014606	051045	053505	047111	MSG2	ASCII /%REWIND ERROR%#/
2537	014614	020104	051105	047522		
2538	014622	021522				
2539	014624	022445	046524	031060	MSG3	ASCII /%TMO2 - TU16 - TE16 BSC FC (CZTUBGO) %/
2540	014632	026440	052040	030525		
2541	014640	020066	020055	042524		
2542	014646	033061	041040	041523		
2543	014654	043040	020103	041450		
2544	014662	052132	041125	030107		
2545	014670	020051	045			
2546	014673	105	052116	051105		ASCII /ENTER CONDITIONS IN OCTAL%#/
2547	014700	041440	047117	044504		
2548	014706	044524	047117	020123		
2549	014714	047111	047440	052103		
2550	014722	046101	021445			
2551	014726	051045	043505	051511	MSG4	ASCII /%REGISTER START = %/
2552	014734	042524	020122	052123		
2553	014742	051101	020124	020075		
2554	014750	043				
2555	014751	045	042526	052103	MSG5	ASCII /%VECTOR = %/
2556	014756	051117	036440	021440		
2557	014764	042445	042116	047440	MSG6	ASCII /%END OF PASS %/
2558	014772	020106	040520	051523		
2559	015000	021440				
2560	015002	037440	021440		MSG7	ASCII / ? %/
2561	015006	021445			MSG8	ASCII /%#/
2562	015010	050045	051517	052111	MSG9	ASCII /%POSITION ERROR %/
2563	015016	047511	020116	051105		
2564	015024	047522	035122	021440		
2565	015032	042045	044522	042526	MSG10	ASCII /%DRIVE NUMBER %/
2566	015040	047040	046525	042502		
2567	015046	035122	021440			

2568	015052	051445	040514	042526	MSG11	ASCII	/%SLAVE NUMBER #/
2569	015060	047040	046525	042502			
2570	015066	035122	021440				
2571	015072	053445	044522	042524	MSG12	ASCII	/%WRITE ERROR #/
2572	015100	042440	051122	051117			
2573	015106	021440					
2574	015110	051045	040505	020104	MSG13	ASCII	/%READ REVERSE ERROR #/
2575	015116	042522	042526	051522			
2576	015124	020105	051105	047522			
2577	015132	020122	043				
2578	015135	045	042522	042101	MSG14	ASCII	/%READ FORWARD ERROR #/
2579	015142	043040	051117	040527			
2580	015150	042122	042440	051122			
2581	015156	051117	021440				
2582	015162	053445	044522	042524	MSG15	ASCII	/%WRITE TM ERROR #/
2583	015170	052040	020115	051105			
2584	015176	047522	020122	043			
2585	015203	045	042522	042526	MSG16	ASCII	/%REVERSE ERROR #/
2586	015210	051522	020105	051105			
2587	015216	047522	020122	043			
2588	015223	045	047506	053522	MSG17	ASCII	/%FORWARD ERROR #/
2589	015230	051101	020104	051105			
2590	015236	047522	020122	043			
2591	015243	040	051116	020132	MSG20	ASCII	/ NRZ #/
2592	015250	043					
2593	015251	040	042520	021440	MSG21	ASCII	/ PE #/
2594	015256	042440	050130	035124	MSG22	ASCII	/ EXPT #/
2595	015264	021440					
2596	015266	051040	053103	035104	MSG23	ASCII	/ RCVD #/
2597	015274	021440					
2598	015276	041045	051525	052040	MSG24	ASCII	/%BUS TRAP #/
2599	015304	040522	035120	021440			
2600	015312	053445	035103	021440	MSG25	ASCII	/%WC #/
2601	015320	041045	035101	021440	MSG26	ASCII	/%BA #/
2602	015326	042045	035102	021440	MSG27	ASCII	/%DB #/
2603	015334	044445	044516	020124	MSG28	ASCII	/%INIT DID NOT CLEAR PH #/
2604	015342	044504	020104	047516			
2605	015350	020124	046103	040505			
2606	015356	020122	044122	021440			
2607	015364	051445	020103	047516	MSG29	ASCII	/%SC NOT RESET BY INIT #/
2608	015372	020124	042522	042523			
2609	015400	020124	054502	044440			
2610	015406	044516	020124	043			
2611	015413	045	051124	020105	MSG30	ASCII	/%TRE NOT RESET BY INIT #/
2612	015420	047516	020124	042522			
2613	015426	042523	020124	054502			
2614	015434	044440	044516	020124			
2615	015442	043					
2616	015443	045	051503	020062	MSG31	ASCII	/%CS2 NOT RESET BY INIT #/
2617	015450	047516	020124	042522			
2618	015456	042523	020124	054502			
2619	015464	044440	044516	020124			
2620	015472	043					
2621	015473	045	046104	020124	MSG32	ASCII	/%DLT NOT SET #/
2622	015500	047516	020124	042523			
2623	015506	020124	043				

2624	015511	045	041523	047040	MSG33	ASCII	/%SC NOT SET #/
2625	015516	052117	051440	052105			
2626	015524	021440					
2627	015526	052045	042522	047040	MSG34	ASCII	/%TRE NOT SET #/
2628	015534	052117	051440	052105			
2629	015542	021440					
2630	015544	044445	020122	047516	MSG35	ASCII	/%IR NOT SET BY INIT #/
2631	015552	020124	042523	020124			
2632	015560	054502	044440	044516			
2633	015566	020124	043				
2634	015571	045	051117	047040	MSG36	ASCII	/%OR NOT RESET BY INIT #/
2635	015576	052117	051040	051505			
2636	015604	052105	041040	020131			
2637	015612	047111	052111	021440			
2638	015620	047445	020122	047516	MSG37	ASCII	/%OR NOT RESET BY 1 SILO ENTRY #/
2639	015626	020124	042522	042523			
2640	015634	020124	054502	030440			
2641	015642	051440	046111	020117			
2642	015650	047105	051124	020131			
2643	015656	043					
2644	015657	045	051117	047040	MSG38	ASCII	/%OR NOT SET BY SILO FULL #/
2645	015664	052117	051440	052105			
2646	015672	041040	020131	044523			
2647	015700	047514	043040	046125			
2648	015706	020114	043				
2649	015711	045	040502	020104	MSG39	ASCII	/%BAD SILO READ #/
2650	015716	044523	047514	051040			
2651	015724	040505	020104	043			
2652	015731	045	051111	047040	MSG40	ASCII	/%IR NOT RESET BY SILO FULL #/
2653	015736	052117	051040	051505			
2654	015744	052105	041040	020131			
2655	015752	044523	047514	043040			
2656	015760	046125	021514				
2657	015764	047040	047117	042455	MSG41	ASCII	/ NON-EXIST DRIVE#.
2658	015772	044530	052123	042040			
2659	016000	044522	042526	043			
2660	016005	040	047516	026516	MSG42	ASCII	/ NON-EXIST SLAVE#.
2661	016012	054105	051511	020124			
2662	016020	046123	053101	021505			
2663	016026	051440	051105	040511	MSG43	ASCII	/ SERIAL NO #/
2664	016034	020114	047516	020072			
2665	016042	043					
2666	016043	045	051105	051501	MSG44	ASCII	/%ERASE HEAD INOPERATIVE
2667	016050	020105	042510	042101			
2668	016056	044440	047516	042520			
2669	016064	040522	044524	042526			
2670	016072	041445	042510	045503		ASCII	/%CHECK POLARITY#.
2671	016100	050040	046117	051101			
2672	016106	052111	021531				
2673	016112	042445	040522	042523	MSG45	ASCII	/%ERASE HEAD POLARITY WRONG#.
2674	016120	044040	040505	020104			
2675	016126	047520	040514	044522			
2676	016134	054524	053440	047522			
2677	016142	043516	043				
2678	016145	045	042523	026524	MSG46	ASCII	/%SET-UP WRITE ERROR#.
2679	016152	050125	053440	044522			

2680	016160	042524	042440	051122			
2681	016166	051117	043				
2682	016171	045	050123	041501	MSG47	ASCII	/%SPACE FORWARD ERROR#
2683	016176	020105	047506	053522			
2684	016204	051101	020104	051105			
2685	016212	047522	021522				
2686	016216	051445	040520	042503	MSG48	ASCII	/%SPACE REVERSE ERROR#
2687	016224	051040	053105	051105			
2688	016232	042523	042440	051122			
2689	016240	051117	043				
2690	016243	045	052502	043106	MSG49	ASCII	/%BUFFERED WRITE ERROR#
2691	016250	051105	042105	053440			
2692	016256	044522	042524	042440			
2693	016264	051122	051117	043			
2694	016271	045	047502	020124	MSG50	ASCII	/%BOT SET AFTER BUFFERED WRITE#
2695	016276	042523	020124	043101			
2696	016304	042524	020122	052502			
2697	016312	043106	051105	042105			
2698	016320	053440	044522	042524			
2699	016326	043					
2700	016327	045	047516	041040	MSG51	ASCII	/%NO BOT FROM READ IN PRESET#
2701	016334	052117	043040	047522			
2702	016342	020115	042522	042101			
2703	016350	044440	020116	051120			
2704	016356	051505	052105	043			
2705	016363	045	041524	044440	MSG52	ASCII	/%TC INCORRECT #
2706	016370	041516	051117	042522			
2707	016376	052103	021440				
2708	016402	051445	040514	042526	MSG53	ASCII	/%SLAVE NOT OFF LINE#
2709	016410	047040	052117	047440			
2710	016416	043106	046040	047111			
2711	016424	021505					
2712	016426	022445	042522	042523	MSG54	ASCII	/%RESET SLAVE TO ON LINE BEFORE CONTINUING#
2713	016434	020124	046123	053101			
2714	016442	020105	047524	047440			
2715	016450	020116	044514	042516			
2716	016456	041040	043105	051117			
2717	016464	020105	047503	052116			
2718	016472	047111	044525	043516			
2719	016500	043					
2720	016501	045	051116	020132	MSG55	ASCII	/%NRZ ONLY #
2721	016506	047117	054514	020072			
2722	016514	043					
2723	016515	040	052111	051105	MSG56	ASCII	/%ITER #
2724	016522	020072	043				
2725	016525	045	046524	047040	MSG57	ASCII	/%TM NOT SET#
2726	016532	052117	051440	052105			
2727	016540	043					
2728	016541	045	044505	044124	MSG60	ASCII	/%EITHER TAPE NOT ERASED OR OPI PROBLEM#
2729	016546	051105	052040	050101			
2730	016554	020105	047516	020124			
2731	016562	051105	051501	042105			
2732	016570	047440	020122	050117			
2733	016576	020111	051120	041117			
2734	016604	042514	021515				
2735	016610	051045	030510	020061	MSG61	ASCII	/%RH11 OR RH70 #

SEQ 0076

```

2736 016616 051117 051040 033510
2737 016624 035060 021440
2738 016630 051045 020110 047117 MSG62 ASC11 /%RH ONLY. #/
2739 016636 054514 020072 043
2740

```

```

2741                                     ,TEST HEADERS*****
2742
2743 016643      045 043045 030524 MSFT1  ASCII  /%%FT1 RH ADDRESSING #/
2744 016650      051072 020110 042101
2745 016656      051104 051505 044523
2746 016664      043516 021440
2747 016670      022445 052106 035062 MSFT2  ASCII  /%%FT2 RH REGISTER BITS TEST #/
2748 016676      044122 051040 043505
2749 016704      051511 042524 020122
2750 016712      044502 051524 052040
2751 016720      051505 020124      043
2752 016725      045 043045 031524 MSFT3  ASCII  /%%FT3 RH INITIALIZE TEST #/
2753 016732      051072 020110 047111
2754 016740      052111 040511 044514
2755 016746      042532 052040 051505
2756 016754      020124      043
2757 016757      045 043045 032124 MSFT4  ASCII  /%%FT4 RH11 SILO TEST 1 #/
2758 016764      051072 030510 020061
2759 016772      044523 047514 052040
2760 017000      051505 020124 020061
2761 017006      043
2762 017007      045 043045 032524 MSFT5  ASCII  /%%FT5 RH11 SILO TEST 2 #/
2763 017014      051072 030510 020061
2764 017022      044523 047514 052040
2765 017030      051505 020124 020062
2766 017036      043
2767 017037      045 043045 033124 MSFT6  ASCII  /%%FT6 RH11 SILO TEST 3 #/
2768 017044      051072 030510 020061
2769 017052      044523 047514 052040
2770 017060      051505 020124 020063
2771 017066      043
2772 017067      045 043045 033524 MSFT7  ASCII  /%%FT7 RH11 SILO TEST 4 #/
2773 017074      051072 030510 020061
2774 017102      044523 047514 052040
2775 017110      051505 020124 020064
2776 017116      043
2777 017117      045 043045 030524 MSFT10  ASCII  /%%FT10 RH11 SILO TEST 5 #/
2778 017124      035060 044122 030461
2779 017132      051440 046111 020117
2780 017140      042524 052123 032440
2781 017146      021440
2782 017150      022445 052106 030461 MSFT11  ASCII  /%%FT11 NOP TEST#/
2783 017156      047072 050117 052040
2784 017164      051505 021524
2785 017170      022445 052106 031061 MSFT12  ASCII  /%%FT12 REWIND TEST#/
2786 017176      051072 053505 047111
2787 017204      020104 042524 052123
2788 017212      043
2789 017213      045 043045 030524 MSFT13  ASCII  /%%FT13 WRITE-READ TEST#/
2790 017220      035063 051127 052111
2791 017226      026505 042522 042101
2792 017234      052040 051505 021524
2793 017242      022445 052106 032061 MSFT14  ASCII  /%%FT14 SPACE TEST#/
2794 017250      051472 040520 042503
2795 017256      052040 051505 021524
2796 017264      022445 052106 032461 MSFT15  ASCII  /%%FT15 ERASE TEST#/
  
```


2797	017272	042472	040522	042523			
2798	017300	052040	051505	021524			
2799	017306	022445	052106	033061	MSFT16	ASCII	/%FT16 TAPE MARK WRITE-READ TEST#
2800	017314	052072	050101	020105			
2801	017322	040515	045522	053440			
2802	017330	044522	042524	051055			
2803	017336	040505	020104	042524			
2804	017344	052123	043				
2805	017347	045	043045	030524	MSFT17	ASCII	/%FT17 TM SPACE TEST #
2806	017354	035067	046524	051440			
2807	017362	040520	042503	052040			
2808	017370	051505	020124	043			
2809	017375	045	043045	031124	MSFT20	ASCII	/%FT20 WRITE CHECK TEST #
2810	017402	035060	051127	052111			
2811	017410	020105	044103	041505			
2812	017416	020113	042524	052123			
2813	017424	021440					
2814	017426	022445	052106	030462	MSFT21	ASCII	/%FT21 ERASE HEAD TEST#
2815	017434	042472	040522	042523			
2816	017442	044040	040505	020104			
2817	017450	042524	052123	043			
2818	017455	045	043045	031124	MSFT22	ASCII	/%FT22 BUFFERED COMMAND TEST#
2819	017462	035062	052502	043106			
2820	017470	051105	042105	041440			
2821	017476	046517	040515	042116			
2822	017504	052040	051505	021524			
2823	017512	022445	052106	031462	MSFT23	ASCII	/%FT23 READ IN PRESET TEST#
2824	017520	051072	040505	020104			
2825	017526	047111	050040	042522			
2826	017534	042523	020124	042524			
2827	017542	052123	043				
2828	017545	045	043045	031124	MSFT24	ASCII	/%FT24 REWIND-OFF LINE TEST#
2829	017552	035064	042522	044527			
2830	017560	042116	047455	043106			
2831	017566	046040	047111	020105			
2832	017574	042524	052123	043			
2833	017601	045	043536	043	\$CNTG	ASCII	/% G#
2834	017605	045	053523	036522	\$MSWR	ASCII	/%SWR= #
2835	017612	021440					
2836	017614	020040	042516	036527	\$MNEW	ASCII	/ NEW= #
2837	017622	021440					
2838	017624	022477	043		\$QUEST	ASCII	/% #
2839							
2840							
2841		017630				EVEN	
2842	017630	000000			WDATA	0	
2843		021342				= +1510	
2844	021342	000000			RDATA	0	
2845							
2846		000001				END	

ABORT	014364	851	2500#												
ACT11M	000672	715#	876	2481*											
ADUMPM	000674	717#	2486*												
AS	000616	685#													
AUTOM	000670	714#	844	924	968	1016	1108	1135	1141	2473*	2476*				
BA	000604	680#	1183	1947*	2033										
BADDR	000714	730#	1441*	1459*	1483*	1492*	1493*	1500*	1576*	1639*	1723*	1770*	1782*	1787*	
		1806*	1823*	1855*	1889*	1947									
BAE	000634	692#	1056												
BTRP	000664	707#	1153*	1163*											
BTRP2	000666	708#	1154*												
CC	000620	686#													
CKMODE	014264	843	2473#												
CKSWR	013750	1084	1144	1386	1623	2048	2119	2134	2196	2414#					
CNTLU	014022	867	2424#	2441											
CONT1	002334	935	956#												
CONT2	002526	979	1002#												
COUNT	001032	769#	2435*	2446	2462*										
CRCNT	001012	761#													
CS	000610	682#	943*	948*	950	956*	957*	959	1002*	1003*	1240*	1251	1280*	1282	
		1307	1311	1316	1322	1346	1350	1400	1414*	1419*	1423	1533*	1831*	2037	
		2159*	2160*												
C1	000600	678#	908	949	958	1156	1243	1247	1284	1286	1863*	1927*	1954*	1973*	
		2029													
DATA0	001042	776#													
DATA1	001044	777#													
DATA2	001046	778#													
DATA3	001050	779#													
DATBL	001040	775#	2072												
DAT1	012324	776	2089#												
DAT1A	012330	2090#	2099	2104											
DAT1B	012334	2091#	2093												
DAT2	012346	777	2098#												
DAT3	012354	778	2103#												
DAT4	012364	779	2108#												
DAT4A	012372	2110#	2113												
DB	000622	687#	1192	1281	1315*	1320*	1342*	1355	1397*	1416*	1420*	1421	1422		
DOUT	013540	2363#	2379	2381											
DOUT0	013622	2377#													
DOUT1	013554	2366#	2367	2375											
DOUT2	013602	2369	2372#												
DOUT3	013610	2371	2373#												
DRIVE	000040	616#													
DRVN	000710	728#	930	941*	944*	945	948	957	987	1003	2160				
DRVTP	000654	703#													
DS	000612	683#	1463	1864	1873	1900	1906	1931	1958	1977	1979	1992	1993	2039	
DSAV	000762	749#	1992*	2055											
DSUP	012232	1478	1769	1805	2067#										
DSO	012234	2068#													
DS1	012254	2073#	2094	2114											
DS2	012274	2077#	2079												
DS3	012304	2074	2080#												
DS4	012314	2082#	2084												
DT	000626	689#	995	997	1005	2073									
EMADDR	000706	727#	1152*	1172*	1212	1237*	1263	1279*	1304*	1338*	1370	1393*	1412*	1447*	
		1465*	1476*	1521*	1637*	1666*	1702*	1766*	1802*	1851*	1886*	1925*	1981	2008	

