

IDENTIFICATION

PRODUCT CODE: MAINDEC-Ø8-DHTDA-A-D
 REPLACES: MAINDEC-8E-D3AB-D

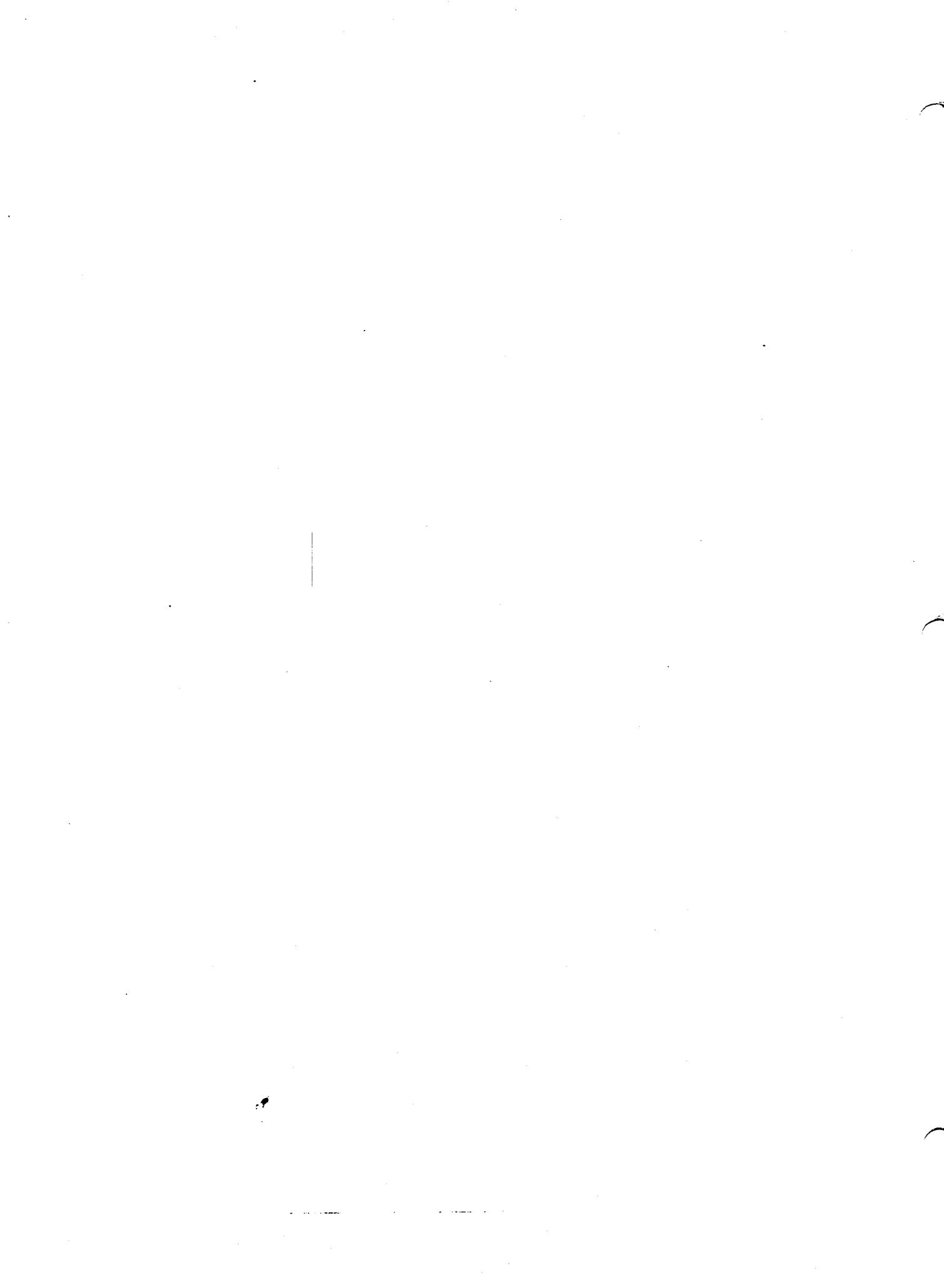
PRODUCT NAME: TD8E DECTAPE DIAGNOSTIC

DATE CREATED: NOVEMBER 1, 1972

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: E. STEINBERGER/B. HANSEN

COPYRIGHT © 1971, 1972
DIGITAL EQUIPMENT CORPORATION



1, ABSTRACT

TD8E DECTAPE DIAGNOSTIC IS A PROGRAM WHICH HAS BEEN WRITTEN TO CHECKOUT AND TEST TD8E DECTAPE CONTROLS WITH TU56 DECTAPE TRANSPORTS. THE PROGRAM TESTS THE BASIC FUNCTIONS OF THE CONTROL (IOT SKIPS, DATA TRANSFERS, ETC) AS WELL AS CHECKING THE ABILITY TO READ AND WRITE ON DECTAPE.

2, REQUIREMENTS

2,1 EQUIPMENT

PDP-8E
TD8E DECTAPE CONTROL
TU56 DECTAPE TRANSPORT (AT LEAST ONE)
ALL NECESSARY CABLES AND MODULES

2,2 STORAGE

THE PROGRAM OCCUPIES MEMORY FROM LOCATION 20 TO LOCATION 7177 AND USES LOCATIONS 7200 TO 7577 AS DATA BUFFER AREA.

2,3 PRELIMINARY PROGRAMS

NONE

#, LOADING PROCEDURE

3,1 METHOD

THE PROGRAM IS LOADED USING THE STANDARD BINARY LOADER TECHNIQUE.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

THE FOLLOWING IS A TABLE OF CONTROL SWITCH SETTINGS AND THEIR ACTION UPON THE PROGRAM:

SR	STATE	ACTION
0	1	LOOP ON CURRENT SUBTEST
	0	DON'T LOOP
1	1	LOOP ON CURRENT TEST
	0	DON'T LOOP
2	1	LOOP ON CONTROL TESTS
	0	DON'T LOOP
3	1	DON'T PRINT ERRORS
	0	PRINT ERRORS
4	1	DON'T HALT ON ERRORS
	0	HALT ON ERROR
5	1	
	0	
6	1	
	0	
7	1	
	0	
8	1	
	0	
9	1	
	0	
10	1	
	0	
11	1	SINGLE UNIT TRANSPORT
	0	DUAL UNIT TRANSPORT

4.2 STARTING ADDRESSES

0200	OPERATOR INTERVENTION TESTS
0201	CONTROL AND DATA TRANSFER TESTS
2100	SEARCH AND FIND ALL BLOCK NUMBERS
2200	DISPLAY BLOCK NUMBERS IN AC
2237	ROUTINE TO ROCK DECTAPE 0 (TIME DEPENDENT ON SWITCH REGISTER)
2400	READ AND CHECK THE MARK TRACK FROM ENDZONE TO ENDZONE
7200	IDT MODIFICATION PROGRAM

4.3 PROGRAM AND/OR OPERATOR ACTION

4.3.1 TO TEST "SELECT ERROR" AND "WRITE LOCK OUT"

4.3.1.1 DUAL TRANSPORTS

- A) SET SWITCH REGISTER TO 0200
- B) ON THE TRANSPORTS, SET ONE TRANSPORT TO UNIT 0, ON-LINE, WRITE LOCK; SET THE OTHER TRANSPORT TO UNIT 1, OFF-LINE.
- C) DEPRESS "LOAD ADDRESS", THEN "CLEAR", THEN "CONTINUE". THE PROGRAM SHOULD TYPE "OK"
- D) REVERSE THE ROLES OF THE TWO TRANSPORTS AND REPEAT STEP C.
- E) SET BOTH TRANSPORTS TO UNIT 1, ON-LINE; DEPRESS "LOAD ADDRESS", THEN "CLEAR", THEN "CONTINUE". THE PROGRAM SHOULD INDICATE NO UNIT 0 SELECTED
- F) PROCEED TO 4.3.2

4.3.1.2 SINGLE TRANSPORT

- A) SET SWITCH REGISTER TO 0200
- B) ON THE TRANSPORT, SET TO UNIT 0, ON-LINE, WRITE LOCK
- C) DEPRESS "LOAD", THEN "CLEAR", THEN "CONTINUE". THE PROGRAM SHOULD TYPE "OK"
- D) PROCEED TO 4.3.2

4.3.2 TO TEST CONTROL AND ABILITY TO PERFORM DATA TRANSFERS

- A) SET SWITCH REGISTER TO 0201, DEPRESS "LOAD ADDRESS"

- B) SET SWITCH REGISTER PER 4,1, SET SR11 IF ONLY ONE TRANSPORT EXISTS OR ONLY ONE TRANSPORT IS TO BE TESTED,
- C) MOUNT A STANDARD PDP-8 DECTAPE (2702 BLOCKS, 201 WORDS PER BLOCK) ON EACH TRANSPORT TO BE TESTED WITH THE TAPES WRAPPED AT LEAST 2 TURNS ON EACH TAKE UP REEL, RESPECTIVELY,
- D) SET A TRANSPORT TO UNIT 0, ON-LINE, WRITE ENABLE; SET THE OTHER TRANSPORT (IF IT EXISTS OR IS TO BE TESTED) TO UNIT 1, ON -LINE, WRITE ENABLE,
- E) DEPRESS "CLEAR", THEN "CONTINUE", THE PROGRAM WILL PERFORM THE BASIC CONTROL TESTS ON THE TD8E, AND, IF SR2 IS A 0, PROCEED TO MOVE TAPE AND PERFORM DATA TRANSFERS TO AND FROM TAPE, CHECKING THE RESULTS;

4,3,3

TO MODIFY THE TD8E IOT SET TO HANDLE A CONTROL FOR UNITS OTHER THAN 0 AND 1,

- A) SET SWITCH REGISTER TO 7200, DEPRESS "LOAD ADDRESS"
- B) SET SWITCH REGISTER BITS 6, 7 AND 8 TO DEVICE SELECTOR BITS 6, 7, AND 8 OF THE CONTROL TO BE TESTED (4, 5, 6 OR 7)
- C) DEPRESS "CLEAR", THEN "CONTINUE", THE PROGRAM WILL MODIFY ALL TD8E IOT'S TO HANDLE THE SELECTED CONTROL,
- D) PERFORM ALL TESTS INDICATED IN 4,3,1 AND 4,3,2 ABOVE FOR THE SELECTED CONTROL SUBSTITUTING UNIT 2, 4 OR 6 FOR UNIT 0 AND UNIT 3, 5 OR 7 FOR UNIT 1 ABOVE,
- E) CAUTION- THE CODE TO CHANGE THE IOT'S IS IN THE DATA BUFFER AREA FOR THE DATA TRANSFER TESTS AND WILL BE DESTROYED WHEN THAT PORTION OF THE PROGRAM IS RUN, AN OVERLAY TAPE IS PROVIDED TO ALLOW THIS CODE TO BE READ BACK INTO MEMORY FOR RE-EXECUTION, MAINDEC-00-DHYDA-A-PB2

5, OPERATING PROCEDURE
-----5,1 OPERATIONAL SWITCH SETTINGS

SEE 4,1

5,2 SUBROUTINE ABSTRACTS

NONE

5,3 PROGRAM AND/OR OPERATOR ACTION

SEE 4,3

5,3,1 IF PROBLEMS ARE SUSPECTED IN THE CONTROL WHEN READING THE TIMING TRACK OFF OF DECTAPE INCLUDING SINGLE LINE FLAG AND QUAD LINE FLAG, A ROCK TAPE ROUTINE HAS BEEN PROVIDED AT LOCATION 2237 TO ALLOW SCOPING OF SINGLE LINE FLAG, QUAD LINE FLAG, UP-TO-SPEED, ETC.

(2237)

- A) SET SWITCH REGISTER TO 2237, DEPRESS "LOAD ADDRESS"
- B) SET SWITCH REGISTER TO 0070, DEPRESS "CLEAR" THEN "CONTINUE", THE DECTAPE ON UNIT 0 SHOULD START ROCKING.
- C) MODIFY SWITCH REGISTER SETTING TO INCREASE OR DECREASE "ROCK" PERIOD.
- D) CAUTION-IF THE NUMBER IN THE SWITCH REGISTER IS TOO SMALL, THE DECTAPE TRANSPORT WILL NOT GET UP TO SPEED BEFORE IT TURNS AROUND.

5,3,2 A ROUTINE HAS BEEN PROVIDED AT LOCATION 2100 TO ALLOW A DECTAPE TO BE RUN FROM ENDZONE TO ENDZONE COMPARING ALL BLOCK NUMBERS; TO RUN THIS ROUTINE, START THE COMPUTER AT LOCATION 2100, THE SR HAS NO AFFECT UPON THE ROUTINE, TO RUN UNIT 1 CHANGE THE CONTENTS OF UNIT (LOCATION 2234) TO 4000, A HALT WILL OCCUR AT LOCATION 2150 IF AN ERROR OCCURS, THE CONTENTS OF THE AC EQUALS THE BLOCK THAT WAS BEING SEARCHED FOR, PRESS "CONT" AND THE PROGRAM WILL HALT AT LOCATION 2153 WITH THE AC EQUAL TO THE BLOCK THAT WAS FOUND, PRESS "CONT" AGAIN, THE PROGRAM WILL RECYCLE UNTIL ANOTHER ERROR IS FOUND.

(2100)

5,3,3 A ROUTINE HAS BEEN PROVIDED AT LOCATION 2200 TO ALLOW A DECTAPE TO BE RUN FROM ENDZONE TO ENDZONE WITH THE BLOCK NUMBERS DISPLAYED IN THE AC, TO RUN THIS ROUTINE, START THE COMPUTER AT LOCATION 2200, THE SR HAS NO AFFECT UPON THE ROUTINE, TO RUN UNIT 1, CHANGE THE CONTENTS OF UNIT (LOCATION 2234) TO 4000, NO ERRORS ARE DETECTED.

(2200)

5,3,4 A ROUTINE HAS BEEN PROVIDED AT LOCATION 2400 TO ALLOW A DECTAPE TO BE RUN FROM ENDZONE TO ENDZONE WITH THE MARK TRACK FORMAT BEING CHECKED, AN ERROR HALT WILL OCCUR IF AN ERROR IS DETECTED, TO RUN THIS ROUTINE (ONLY ON UNIT 0), START THE COMPUTER AT LOCATION 2400, THE SR HAS NO AFFECT UPON THE ROUTINE.

(2400)

6, ERRORS

6,1 ERROR HALTS AND DESCRIPTION

MOST ERROR HALTS ARE PRECEDED BY AN ERROR TYPEOUT (UNLESS SR3 IS A 1). IF NO ERROR TYPEOUT OCCURS, CONSULT THE LISTING FOR THE CAUSE OF THE ERROR.

6,2 ERROR RECOVERY

MOST ERRORS (EXCEPT DATA ERRORS) CAN BE ISCOPED BY SETTING SR0 TO 1 AND DEPRESSING "CONTINUE"

DATA ERRORS CANNOT BE ISCOPED, BUT DATA TRANSFERS CAN BE CONTINUED BY DEPRESSING "CONTINUE".

6,3 IF TAPE RUNS OFF THE END

NORMALLY, TAPE WILL NOT RUN OFF THE END OF THE REEL UNLESS THE PROGRAM IS IN A ISCOPE LOOP OR A SELECT ERROR OCCURS WHEN A TAPE IS MOVING (THE OPERATOR SETTING BOTH TAPE UNITS TO THE SAME NUMBER DURING THE DATA TRANSFER TESTS).

IF TAPE RUNS OFF THE END AND THE PROGRAM HANGS AROUND LOCATION 0621, CHECK THE ABILITY TO READ THE TIMING TRACK INTO THE T08E CONTROL AND THE CIRCUITS RELATING TO THE TIMING PULSE GENERATOR.

IF TAPE RUNS OFF THE END AND THE PROGRAM HANGS AROUND LOCATION 1466, CHECK THE ABILITY TO READ THE MARK TRACK INTO THE T08E CONTROL AND THE CIRCUITS RELATING TO THE MARK TRACK REGISTER.

7, RESTRICTIONS

7,1 STARTING RESTRICTIONS

NONE IF PARAGRAPH 4,3 IS PROPERLY FOLLOWED.

7,2 OPERATING RESTRICTIONS

NONE IF PARAGRAPH 4,3 IS PROPERLY FOLLOWED.

8: MISCELLANEOUS
-----8.1 EXECUTION TIME - \approx 20 MIN (Transport SA=201)

THE EXECUTION TIME OF THE CONTROL TESTS IS NORMALLY LESS THAN 1 MINUTE, DEPENDING UPON THE POSITION OF TAPE ON UNIT 0;

THE EXECUTION TIME OF THE DATA TESTS DEPENDS ON WHETHER ONE OR TWO TRANSPORTS IS BEING EXERCISED, PASS "N" COMPLETE WILL BE PRINTED ON THE TELEPRINTER AFTER ALL DATA PATTERNS HAVE BEEN EXERCISED ONCE, (NORMALLY LESS THAN 1 HOUR PER PASS);

20 min/Transport.

9: PROGRAM DESCRIPTION
-----9.1 DATA REGISTER TEST (SA=0201)

IN THIS TEST THE DATA REGISTER IS CHECKED FOR ITS ABILITY TO BE LOADED AND READ; FIRST THE COMPLEMENT OF THE DATA TO BE CHECKED IS LOADED INTO THE DATA REGISTER, THEN THE DATA ITSELF IS LOADED INTO THE REGISTER; THIS IS DONE TO CHECK THAT ALL BITS CAN BE LOADED TO A 1 FROM A 0 AND TO A 0 FROM A 1; THE DATA IS THEN READ INTO THE AC AND CHECKED FOR ERRORS; AN INCREMENT PATTERN IS USED;

9.2 COMMAND REGISTER TEST (SA=0236)

IN THIS TEST THE COMMAND REGISTER IS CHECKED FOR ITS ABILITY TO BE LOADED AND READ; DATA IS LOADED INTO THE COMMAND REGISTER THEN READ IN THE AC AND CHECKED FOR ERRORS; A 400 INCREMENT PATTERN IS USED, THE STOP/GO BIT IS MASKED OUT;

9.3 INITIALIZE TEST (SA=0305)

THIS TESTS CHECKS THAT "CAP" CLEARS THE COMMAND REGISTER; THE C,R, IS LOADED WITH 6400, THEN "CAP" IS ISSUED; THE C,R, IS THEN READ AND CHECKED TO CONTAIN 0;

9.4 CHECK SDLC, SOLD, SDRG, AND SDRD AND AC CLEAR (SA=0400)

THIS TEST CHECKS THE AC CLEAR FUNCTION OF THE SDLC, SOLD, SDRG AND SDRD INSTRUCTION; THIS IS DONE BY SETTING THE AC TO 7777; THEN ISSUING THE APPROPRIATE IOT (ONE AT A TIME) AND CHECK TO SEE IF THE AC DID OR DID NOT CLEAR (SOLD DOES NOT CLEAR THE AC, THE OTHERS DO);

9.5 CHECK SINGLE LINE SKIP INSTRUCTION AND LOGIC TEST (\$A=0600)

THIS TEST CHECKS THE SINGLE LINE FLAG LOGIC AND SKIP INSTRUCTION, IN PARTICULAR IT TESTS: SINGLE LINE FLAG CLEAR AFTER A "CAF"; SINGLE LINE FLAG SETS; SDSS DOES NOT CLEAR SINGLE LINE FLAG; CAF CLEARS SINGLE LINE FLAG; SOLD CLEARS SINGLE LINE FLAG; SDRG CLEARS SINGLE LINE FLAG; SDRD CLEARS SINGLE LINE FLAG; SDST, SDSQ, AND SDLC DOES NOT CLEAR SINGLE LINE FLAG.

9.6 CHECK QUAD LINE SKIP INSTRUCTION AND LOGIC TEST (\$A=1024)

THIS TEST CHECKS THE QUAD LINE FLAG LOGIC AND SKIP INSTRUCTION; IN PARTICULAR IT TESTS: QUAD LINE FLAG CLEAR AFTER A "CAF"; QUAD LINE FLAG SETS AT PROPER TIME; SDSQ DOES NOT CLEAR QUAD LINE FLAG; CAF CLEARS QUAD LINE FLAG; SOLD CLEARS QUAD LINE FLAG; SDRG CLEARS QUAD LINE FLAG; SDRD CLEARS QUAD LINE FLAG; SDST, SDSQ, AND SDLC DOES NOT CLEAR QUAD LINE FLAG; ALL QUAD LINE FLAG COUNTER FLIP/FLOPS GET CLEARED (BY SOLD);

9.7 CHECK TIMING ERROR SKIP INSTRUCTION AND LOGIC TEST (\$A=1315)

THIS TEST CHECKS THE TIMING ERROR LOGIC AND SKIP INSTRUCTION; IN PARTICULAR IT TESTS: TIMING ERROR CLEAR AFTER A "CAF"; TIMING ERROR SETS IN READ MODE (SDSQ SKIPS); SDST DOES NOT CLEAR TIMING ERROR; CAF CLEARS TIMING ERROR; TIMING ERROR STATUS BIT CAN BE READ INTO AC BY SDRG; SDLC CLEARS TIMING ERROR; TIMING ERROR SETS IN WRITE MODE (PERFORMED AT REVERSE ENDZONE AT BEGINNING OF TAPE); TIMING ERROR STATUS CLEARS "WRITE"; SDRG SDRD SOLD ISSUED AT THE WRONG TIME SETS TIMING ERROR.

9.8 CHECK UP TO SPEED CIRCUITRY TEST (\$A=1400)

THIS TEST CHECKS THE UP-TO-SPEED CIRCUITRY TO FUNCTION PROPERLY WHEN CERTAIN COMMANDS ARE GIVEN TO THE DECTAPE CONTROL, THE CHECK IS PERFORMED VIA THE FEATURE OF THE UP-TO-SPEED CIRCUITRY CLEARING THE MARK TRACK REGISTER WHEN THE UP-TO-SPEED DELAY STARTS TIMING OUT, THE COMMANDS ISSUED ARE: STOP TO GO; GO TO STOP; REVERSE TO FORWARD; FORWARD TO REVERSE; UNIT 0 TO UNIT 1; UNIT 1 TO UNIT 0 (ONLY IF UNIT 1 EXISTS - SR11 SET TO 1)

9,9 ROUTINE TO SEARCH AND FIND ALL BLOCK NUMBERS (SA=2100)

THIS ROUTINE RUNS TAPE FROM ENDBONE TO ENDBONE COMPARING ALL
 BLOCK NUMBERS,

9,10 DISPLAY BLOCK NUMBER ROUTINE (SA=2200)

THIS ROUTINE RUNS TAPE FROM ENDBONE TO ENDBONE DISPLAYING
 THE CURRENT BLOCK NUMBER IN THE AC,

9,11 ROUTINE TO ROCK DECTAPE 0 (SA=2237)

THIS ROUTINE ROCKS DECTAPE 0 FOR A DISTANCE DETERMINED
 BY THE CONTENTS OF THE SWITCH REGISTER; THIS ROUTINE
 CAN BE USED TO CHECK "UP TO SPEED", SINGLE LINE FLAG,
 AND QUAD LINE FLAG LOGIC,

9,12 ROUTINE TO RUN DECTAPE FROM ENDBONE TO ENDBONE AND CHECK

THE MARK TRACK IN BLOCKS (SA=2400)

THIS ROUTINE RUNS DECTAPE 0 FROM ENDBONE TO ENDBONE AND
 CHECKS THE CONTENTS OF THE MARK TRACK ON TAPE IN THE
 BLOCKS ON TAPE,

9,13 CHECK SELECT ERROR CIRCUITRY TEST (SA=2500, 2600)

THIS TEST CHECKS THE "SELECT ERROR" CIRCUITRY OF THE
 TDBE CONTROL UNIT 0 IS "ON-LINE", UNIT 1 IS "OFF-LINE"
 OR NO-EXISTANT; FUNCTIONS CHECKED ARE: "SELECT ERROR"
 STATUS FROM UNIT 1; "SELECT ERROR" PREVENTING "WRITE"
 FROM SETTING; NO "SELECT ERROR" FROM UNIT 0,

9,14 CHECK WRITE LOCK OUT CIRCUITRY TEST (SA=2673)

THIS TEST CHECKS THE "WRITE LOCK OUT" CIRCUITRY OF THE
 TDBE CONTROL, UNIT 0 IS "ON-LINE" AND "WRITE LOCKED",
 FUNCTIONS CHECKED ARE: "WRITE-LOCK" STATUS FROM UNIT 0;
 WRITE LOCK STATUS PREVENTING "WRITE FROM SETTING,

"OK" IS PRINTED ON THE TELEPRINTER AFTER THE TWO TESTS
 DESCRIBED IN 9,14 AND 9,15 ABOVE ARE COMPLETED,

9:15

DATA TRANSFER TEST (SAB3000)

DATA TRANSFER TESTS IS A SERIES OF ROUTINES WHICH CHECK THE READ - WRITE - SEARCH CAPABILITIES OF THE T08E CONTROL; EIGHT BASIC DATA PATTERNS ARE USED FOR DATA TRANSFER, THESE ARE: A BUFFER FULL OF 0'S; A BUFFER FULL OF -1'S; A BUFFER FULL OF 2525'S; A BUFFER FULL OF THE DATA PATTERN 2225, 5522, 2535, REPEATED; A BUFFER FULL OF INCREMENT BY 1 DATA PATTERN; A BUFFER FULL OF DECREMENT BY 1 DATA PATTERN; A BUFFER FULL OF 6161'S; A BUFFER FULL OF 3434'S.

DATA TRANSFERS ARE PERFORMED IN BOTH THE FORWARD AND REVERSE DIRECTION; DATA IS WRITTEN IN THE FORWARD DIRECTION, FIRST INTO BLOCK 0, THE SEQUENCE OF OPERATIONS IS: WRITE DATA IN THE FORWARD DIRECTION; READ DATA IN THE FORWARD DIRECTION; CHECK CHECKSUM AND DATA; READ DATA IN THE REVERSE DIRECTION; CHECK CHECKSUM ONLY; THIS SEQUENCE IS REPEATED EVERY 100 BLOCKS (BLOCK 0, 100, 200, 300, ETC) UP TO AND INCLUDING BLOCK 2700, IF IT IS DESIRED TO GO A DIFFERENT NUMBER OF BLOCKS FORWARD CHANGE LOCATION 3154 TO THE DESIRED NUMBER OF BLOCKS;

DATA IS THEN WRITTEN IN THE REVERSE DIRECTION, FIRST INTO BLOCK 2701, THE SEQUENCE OF OPERATIONS IS: WRITE DATA IN THE REVERSE DIRECTION; READ DATA IN REVERSE DIRECTION; CHECK CHECKSUM AND DATA; READ DATA IN THE FORWARD DIRECTION; CHECK CHECKSUM ONLY; THIS SEQUENCE IS REPEATED EVERY 100 BLOCKS (BLOCK 2701, 2801, 2901, 2401, ETC) DOWN TO AND INCLUDING BLOCK 1, IF IT IS DESIRED TO GO A DIFFERENT NUMBER OF BLOCKS REVERSE CHANGE LOCATION 3146 TO THE 2'S COMPLEMENT OF THE DESIRED NUMBER OF BLOCKS;

AFTER UNIT 0 HAS BEEN COMPLETELY TRAVERSED ONCE (FORWARD AND BACKWARD), UNIT 1 WILL BE RUN, IF AVAILABLE, THE PROGRAM WILL THEN PROCEED TO THE NEXT DATA PATTERN AND UNIT 0 AGAIN, AFTER ALL 8 DATA PATTERNS HAVE BEEN EXERCISED ON BOTH UNITS, THE PROGRAM WILL PRINT "PASS 'N' COMPLETE" ON THE TELEPRINTER, THEN PROCEED BACK TO THE FIRST DATA PATTERN,

/TDBE DIAGNOSTIC
/
/COPYRIGHT 1971
/DIGITAL EQUIP, CORP.
/MAYNARD, MASS,

/DECTAPE COMMANDS

6771 SDSS=6771 /SKIP ON SINGLE LINE FLAG
6772 SDST=6772 /SKIP ON TIMING ERROR
6773 SDSD=6773 /SKIP ON QUADRUPLE LINE FLAG
6774 SOLC=6774 /LOAD COMMAND REGISTER
6775 SLD=6775 /LOAD DATA REGISTER, CLEAR FLAG
6776 SDRC=6776 /READ COMMAND REGISTER AND MARK TRACK, CLEAR FLAG
6777 SDRD=6777 /READ DATA REGISTER, CLEAR FLAG

6887 /NEW PDP-0E INSTRUCTIONS /CLEAR ALL FLAG (GENERATE INITIALIZE)
CAF=6887

8817 =17

8817 8888 AUTO, 0
8828 8888 OUT, 0
8821 8888 IN, 0
8822 8888 CNTR1, 0
8823 8888 CNTR2, 0
8824 8888 GOOD, 0
8825 8888 HEAD1, 0
8826 8888 HEAD2, 0
8827 8888 BLK, 0
8838 8888 FILPNT, 0
/SWITCH OPTIONS:
/SR0(1) LOOP ON CURRENT SUBTEST
/SR1(1) LOOP ON CURRENT TEST
/SR2(1) LOOP ON CONTROL TESTS
/SR3(1) DON'T PRINT ERRORS
/SR4(1) DON'T HALT ON ERRORS
/SR11(1) SINGLE UNIT TRANSPORT (S-ONLY)
8831 8888 TYPE, 0
8832 8846 TLS
8833 8841 TSP
8834 8833 JMP ,=1
8835 8842 TCF
8836 7288 CLA
8837 8431 JMP I TYPE
8848 8888 CRLF, 0
8841 1177 TAD (215
8842 4831 JMS TYPE
8843 1176 TAD (212
8844 4831 JMS TYPE
8845 5448 JMP I CRLF
8846 8888 LOOP1, 0
8847 7684 LAS

8858 7884 RAL
8851 7788 SNA CLA
8852 2846 IS2 LOOP1
8853 5446 JMP I LOOP1

8854 7778 M18, -18
8855 8888 BLKTRY, 0
8856 8888 DISBL, 0
8857 8888 DISDA, 0
8868 8888 BLKCN, 0

8861 8888 BLKREV, 0
8862 1175 TAD (3888
8863 1774 TAD UNIT
8864 8774 IOT172, SOLC
8865 4773 JMS RDQUAD
8866 4773 JMS RDQUAD
8867 5461 JMP I BLKREV

8878 8888 BLKEND, 0
8871 6771 IOT173, SDSS
8872 5871 JMP ,=1
8873 6776 IOT174, SDRC
8874 8172 AND (77
8875 1171 TAD (=82
8876 7648 SEA CLA
8877 5871 JMP ,=6
8888 5478 JMP I BLKEND

8181 8888 BLKSER, 0
8182 6771 IOT175, SDSS
8183 5182 JMP ,=1
8184 6777 IOT176, SDRD
8185 3857 DCA DISDA
8186 6776 IOT177, SDRC
8187 8172 AND (77
8188 1178 TAD (=26
8189 7648 SEA CLA
8190 5182 JMP ,=18
8191 5581 JMP I BLKSER

8200 8200 PAGE
8200 5777 JMP SELECT /GO TO OPERATOR INTERVENTION TESTS FIRST
/Routine TO CHECK THE LOADING AND READING OF THE DATA REGISTER

8201 7388 DATREG, CLA CLL
8202 3828 DCA OUT /START WITH 8
8203 1376 TAD (HESS1
8204 3825 DCA HEAD1
8205 1828 TAD OUT
8206 7848 CHA
8207 8775 IOT1, SLD /LOAD DATA REGISTER WITH
8210 7288 CLA /COMPLEMENT OF DATA
8211 1828 TAD OUT

0212	6775	10T2,	SDLD		/LOAD DATA REGISTER WITH DATA
0213	7200		CLA		
0214	6777	10T3,	SDRD		/READ DATA REGISTER
0215	3021		DCA	IN	
0216	7604		LAS		
0217	7710		SPA	CLA	/LOOP?
0220	5210		JMP	DATREG+7	/YES
0221	1021		TAD	IN	/COMPARE DATA IN
0222	7041		CLA		
0223	1020		TAD	OUT	/WITH DATA SENT OUT
0224	7650		SNA	CLA	/EQUAL?
0225	5232		JMP	DATLUP	/YES
0226	4775		JMS	ERROR1	
0227	7604		LAS		
0230	7710		SPA	CLA	/LOOP?
0231	5210		JMP	DATREG+7	/YES
0232	2020	DATLUP,	ISE	OUT	/INCREMENT NUMBER TO BE SENT
0233	5005		JMP	DATREG+4	/GO BACK TO DO NEXT NUMBER
0234	4046		JMS	LOOP1	
0235	5201		JMP	DATREG	

/ROUTINE TO CHECK THE LOADING AND READING OF THE COMMAND REGISTER

0236	7300	CONREG,	CLA	CLL	
0237	3020		DCA	OUT	/START WITH 0
0240	1374		TAD	(MESS2	
0241	3025		DCA	HEAD1	
0242	1020		TAD	OUT	
0243	0373		AND	(6400	
0244	6774	10T4,	SDLC		/LOAD COMMAND REGISTER WITH DATA
0245	7200		CLA		
0246	6776	10T5,	SDRC		/READ COMMAND REGISTER
0247	0372		AND	(7400	/MASK TO C,R, BITS
0250	3021		DCA	IN	/AND STORE
0251	7604		LAS		
0252	7710		SPA	CLA	/LOOP
0253	5242		JMP	CONREG+4	/YES
0254	1020		TAD	OUT	/GET GOOD WORD
0255	0373		AND	(6400	/MASK OUT 00 BIT
0256	7041		CLA		
0257	1021		TAD	IN	/COMPARE IT WITH WORD IN
0260	7450		SNA	CLA	/BITS OK?
0261	5246		JMP	CLOOP	/YES
0262	4775		JMS	ERROR1	
0263	7604		LAS		
0264	7710		SPA	CLA	
0265	5242		JMP	CONREG+4	
0266	1020	CLOOP,	TAD	OUT	
0267	1371		TAD	(400	
0270	7450		SNA		
0271	5303		JMP	INITST=2	
0272	3020		DCA	OUT	
0273	7604		LAS		
0274	7010		RAR		/MOVE SINGLE UNIT BIT INTO LINK

74 LD Command 2
RES

7300

76 RB COMMAND
PFF

75 LD Command SW

002

007 Clear
Command Reg

0275	7620	SNL	CLA		/SINGLE UNIT
0276	5242	JMP	CONREG+4		/NO
0277	7010	RAR			
0300	1020	TAD		OUT	/YES, WORKING
0301	7640	SEA	CLA	/ON 2ND UNIT?	
0302	5242	JMP	CONREG+4		/NO
0303	4046	JMS	LOOP1		
0304	5236	JMP	CONREG		
0305	7300	INITST,	CLA	CLL	/TEST INIT TO CLEAR CR
0306	1370		TAD	(MESS3	
0307	3025		DCA	HEAD1	
0310	1367		TAD	(MESS4	
0311	3026		DCA	HEAD2	
0312	1373		TAD	(6400	
0313	6774	10T6,	SDLC		/LOAD CR WITH 74
0314	6007		CAF		/CLEAR CR
0315	7604		LAS		
0316	7710		SPA	CLA	/LOOP?
0317	5303		JMP	INITST	/YES
0320	6776	10T7,	SDRC		/READ CR
0321	0372		AND	(7400	
0322	7650		SNA	CLA	/CR BITS 0?
0323	5330		JMP	,+3	/YES, OK
0324	4766		JMS	ERROR2	/NO, ERROR, INIT (CAF) DID NOT CLEAR CR
0325	7604		LAS		
0326	7710		SPA	CLA	/LOOP?
0327	5303		JMP	INITST	/YES
0330	4046		JMS	LOOP1	
0331	5303		JMP	INITST	
0332	3771		JMP	CHKCLA	
0366	0337				
0367	3034				
0370	3044				
0371	0400				
0372	7400				
0373	6400				
0374	3021				
0375	0476				
0376	3000				
0377	2600				
	0400				

/CHECK SDLC, SDLD, SDRC, SDRD TO CLEAR AC AT PROPER TIME (OR NOT AT ALL)

0400	7300	CHKCLA,	CLA	CLL	
0401	1377		TAD	(MESS5	
0402	3025		DCA	HEAD1	
0403	1376		TAD	(MESS6	
0404	3026		DCA	HEAD2	
0405	1167	SDLC,	TAD	(6777	/SET AC TO 6777
0406	6774	10T8,	SDLC		
0407	7650		SNA	CLA	/OID SDLC CLEAR AC (AC SHOULD CLEAR)?
0410	5215		JMP	,+3	/YES

```

0411 7604 LAS /NO,ERROR
0412 7710 SPA CLA /LOOP?
0413 5205 JMP CSOLC /NO
0414 4337 JMS ERROR2 /ERROR
0415 7604 LAS
0416 7710 SPA CLA /LOOP?
0417 5205 JMP CSOLC /YES
0420 1375 TAD (MESS7
0421 3026 DCA HEAD2
0422 7300 CSORC, CLA CLL
0423 6774 IOT9, SOLC /LOAD COMMAND REGISTER WITH B
0424 7240 CLA CMA /SET AC TO ALL 1'S
0425 6776 IOT10, SDRC /READ COMMAND REGISTER
0426 7690 SNA CLA /ALL ZERO'S (AC SHOULD CLEAR BEFORE READING)?
0427 5234 JMP ,+5 /YES
0430 7604 LAS /NO,ERROR
0431 7710 SPA CLA /LOOP?
0432 5222 JMP CSORC /YES
0433 4337 JMS ERROR2 /ERROR
0434 7604 LAS
0435 7710 SPA CLA /LOOP?
0436 5222 JMP CSORC /YES
0437 1374 TAD (MESS8
0440 3026 DCA HEAD2
0441 7240 CSOLD, CLA CMA /SET AC TO ALL 1'S
0442 6775 IOT11, SOLD /LOAD DATA REGISTER
0443 7640 SZA CLA /DID SOLD CLEAR AC (AC SHOULD NOT CLEAR)?
0444 5251 JMP ,+5 /NO, ALL OR
0445 7604 LAS /YES,ERROR
0446 7710 SPA CLA /LOOP?
0447 5241 JMP CSOLD /YES
0450 4337 JMS ERROR2 /ERROR
0451 7604 LAS
0452 7710 SPA CLA /LOOP?
0453 5241 JMP CSOLD /YES
0454 1373 TAD (MESS9
0455 3026 DCA HEAD2

0456 7300 CSORD, CLA CLL
0457 6775 IOT12, SOLD /LOAD REGISTER WITH B
0460 7240 CLA CMA /SET AC TO ALL 1'S
0461 6777 IOT13, SORD /READ DATA REGISTER
0462 7650 SNA CLA /ALL ZERO'S(AC SHOULD CLEAR BEFORE READING)?
0463 5270 JMP ,+5 /YES
0464 7604 LAS /NO,ERROR
0465 7710 SPA CLA /LOOP?
0466 5256 JMP CSORD /YES
0467 4337 JMS ERROR2 /ERROR
0470 7604 LAS
0471 7710 SPA CLA /LOOP?
0472 5256 JMP CSORD /YES
0473 4046 JMS LOOP1
0474 5200 JMP CHKCLA
0475 5772' JMP SINGLE

```

/ERROR HANDLER ROUTINE=DATA WORD TYPEOUTS

```

0476 0000 ERROR1, 0
0477 7604 LAS /GET SR
0500 0371 AND (400 /MASK TO TYPEOUT BIT
0501 7640 SZA CLA /TYPE OUT ERROR?
0502 5325 JMP ERR1MT /NO
0503 4040 JMS CRLF /YES
0504 1025 TAD HEAD1
0505 7490 SNA /TYPE OUT HEADER?
0506 5315 JMP ,+7 /NO
0507 4770' JMS MESSAGE /YES, PRINT HEADER
0510 3025 DCA HEAD1
0511 4040 JMS CRLF /CRLF
0512 1307 TAD (FORMT1 /PRINT REST OF FORMAT
0513 4770' JMS MESSAGE
0514 4040 JMS CRLF /CRLF
0515 1020 TAD OUT /PRINT GOOD DATA
0516 0366 AND (6400
0517 4765' JMS OPRINT
0520 1344 TAD (240 /SPACE
0521 4031 JMS TYPE
0522 1021 TAD IN /PRINT BAD DATA
0523 4765' JMS OPRINT
0524 4040 JMS CRLF /CRLF
0525 7604 ERR1MT, LAS /GET SR
0526 0363 AND (200 /MASK TO HALT BIT
0527 7650 SNA CLA /STOP?
0530 7402 E1HLT, HLT /NO
0531 5676 JMP ! ERROR1 /EXIT

0532 0717 FORMT1, TEXT "GOOD BAD"
0533 1704
0534 4002
0535 0104
0536 0000

```

/ERROR HANDLER - NO DATA WORD TYPEOUTS

```

0537 0000 ERROR2, 0
0540 7604 LAS /GET SR
0541 0371 AND (400 /MASK TO TYPEOUT BIT
0542 7640 SZA CLA /TYPE OUT ERROR?
0543 5356 JMP ERR2MT /NO
0544 4040 JMS CRLF /YES
0545 1025 TAD HEAD1
0546 7490 SNA /TYPE OUT HEADER
0547 5353 JMP ,+4 /NO
0550 4770' JMS MESSAGE /YES
0551 3025 DCA HEAD1
0552 4040 JMS CRLF
0553 1026 TAD HEAD2 /TYPE OUT ERROR MESSAGE
0554 4770' JMS MESSAGE
0555 4040 JMS CRLF
0556 7604 ERR2MT, LAS /GET SR

```

```

/TDSE DIAGNOSTIC
0557 0363 AND 1200 /MASK TO HALT BIT
0560 7690 SNA CLA /STOP?
0561 7402 HLT /YES
0562 5737 JMP I ERROR2

0563 0200
0564 0240
0565 2316
0566 6400
0567 0532
0570 2264
0571 0400
0572 0600
0573 5161
0574 5191
0575 5136
0576 5123
0577 5076
0600
PAGE

```

/CHECK SINGLE LINE SKIP INSTRUCTION AND LOGIC

```

0600 7300 SINGLE, CLA CLL
0601 1377 TAD (MESS10
0602 3025 DCA HEAD1
0603 1376 TAD (MESS11
0604 3026 DCA HEAD2
0605 6774 SDLC /CLEAR ALL FLAG INITIALLY
0606 6771 10T14, SDSS /SKIP ON SINGLE LINE
0607 7410 SKP
0610 4775' JMS ERROR2
0611 1374 TAD (1000 /ERROR, SDSS SHOULD NOT HAVE SKIPPED
0612 6774 10T15, SOLC /LOAD COMMAND REGISTER WITH US,FND,GO,READ
0613 1373 TAD (MESS12
0614 3026 DCA HEAD2

0615 7200 SINGLE, CLA
0616 1372 TAD (=1000 /SET UP FOR
0617 3022 DCA CNTR1 /A DELAY
0620 3023 DCA CNTR2 /OP ABOUT 1 SECOND
0621 6771 10T16, SDSS /SINGLE LINE FLAG UP YET?
0622 7410 SKP /NO
0623 5234 JMS SINGLE08 /YES
0624 2023 ISB CNTR2 /NO, COUNT
0625 5221 JMP =4
0626 2022 ISB CNTR1 /DELAY OVER?
0627 5221 JMP =6 /NO
0630 4775' JMS ERROR2 /YES, NO SINGLE LINE FLAG, OR SDSS DOES NOT SKIP
0631 7604 LAR
0632 7710 SPA CLA /LOOP?
0633 5200 JMS SINGLE /YES
0634 1371 TAD (MESS13
0635 3026 DCA HEAD2

```

/TDSE DIAGNOSTIC PAL10 V141 19-OCT-78 11189 PAGE 167

```

0636 6771 SINGLE, SDSS /FLAG STILL UP?
0637 4775' JMS ERROR2 /SINGLE LINE FLAG CLEARED BY SDSS
0640 7604 LAR
0641 7710 SPA CLA /LOOP?
0642 5236 JMS SINGLE1 /YES
0643 1370 TAD (MESS14
0644 3026 DCA HEAD2
0645 6771 SINGLE2, SDSS /WAIT FOR SINGLE LINE FLAG
0646 5245 JMS =5
0647 6007 CAF /CLEAR FLAG WITH CAF
0650 7604 LAR
0651 7710 SPA CLA /LOOP?
0652 5245 JMS SINGLE2 /YES
0653 6771 10T17, SDSS /DID FLAG CLEAR?
0654 5241 JMS SINGLE3=4 /YES
0655 4775' JMS ERROR2 /NO, SINGLE LINE FLAG NOT CLEARED BY CAF
0656 7604 LAR
0657 7710 SPA CLA /LOOP?
0660 5245 JMS SINGLE2 /YES
0661 1374 TAD (1000
0662 6774 10T18, SDLC /LOAD COMMAND REGISTER AGAIN

0663 1367 TAD (MESS15
0664 3026 DCA HEAD2
0665 6771 SINGLE3, SDSS /WAIT FOR SINGLE LINE FLAG
0666 5265 JMS =5
0667 7200 CLA
0670 6775 10T19, SLDL /ISSUE SLDL TO CLEAR SINGLE LINE FLAG
0671 7604 LAR
0672 7710 SPA CLA /LOOP?
0673 5265 JMS SINGLE3 /YES
0674 6771 10T20, SDSS /FLAG STILL UP?
0675 5302 JMS SINGLE4=8 /NO
0676 4775' JMS ERROR2 /YES, ERROR, SINGLE LINE FLAG NOT CLEARED BY SLDL
0677 7604 LAR
0680 7710 SPA CLA /LOOP?
0681 5265 JMS SINGLE3
0682 1366 TAD (MESS16
0683 3026 DCA HEAD2
0684 6771 SINGLE4, SDSS /WAIT FOR SINGLE LINE FLAG
0685 5304 JMS =5
0686 7200 CLA
0687 6776 10T21, SDRG /ISSUE SDRG TO CLEAR SINGLE LINE FLAG
0688 7604 LAR
0689 7710 SPA CLA /LOOP?
0690 5304 JMS SINGLE4 /YES
0691 6771 10T22, SDSS /FLAG CLEARED?
0692 5321 JMS SINGLE5=8 /YES
0693 4775' JMS ERROR2 /NO, ERROR, SINGLE LINE FLAG NOT CLEARED BY SDRG
0694 7604 LAR
0695 7710 SPA CLA /LOOP?
0696 5304 JMS SINGLE4 /YES
0697 1365 TAD (MESS17
0698 3026 DCA HEAD2

```



```

0723 6771 SING5, SDSS
0724 5323 JMP
0725 7200 CLA
0726 6777 IOT23, SDRD
0727 7604 LAS
0730 7710 SPA CLA
0731 5323 JMP SING5
0732 6771 IOT24, SDSS
0733 5774 JMP SING6*2
0734 4775 JMS ERROR2
0735 7604 LAS
0736 7710 SPA CLA
0737 5323 JMP SING5
0740 5774 JMP SING6*2
0765 5416
0766 5373
0767 5350
0770 5326
0771 5385
0772 7000
0773 5253
0774 1000
0775 0537
0776 5222
0777 5174
1000

```

PAGE

```

1000 1377 TAD (MESS10
1001 3026 DCA HEAD2
1002 6771 SING6, SDSS
1003 5202 JMP
1004 6772 IOT25, SDST
1005 7000 NOP
1006 6773 IOT26, SDSQ
1007 7200 CLA
1010 1376 TAD (1000
1011 6774 IOT27, SDLC
1012 6771 IOT28, SDSS
1013 7410 SKP
1014 5221 JMP SING7
1015 4775 JMS ERROR2
1016 7604 LAS
1017 7710 SPA CLA
1020 5202 JMP SING6
1021 6774 SING7, SDLC
1022 4044 JMS LOOP1
1023 5774 JMP SINGLE

```

/CHECK QUAD LINE SKIP INSTRUCTION AND LOGIC

```

1024 7300 QUAD, CLA CLL
1025 1373 TAD (MESS19

```

```

1026 3025 DCA HEAD1
1027 1372 TAD (MESS20
1030 3026 DCA HEAD2
1031 6774 IOT29, SDLC
1032 6773 SDSQ
1033 7410 SKP
1034 4775 JMS ERROR2
1035 1376 TAD (1000
1036 6774 IOT30, SDLC
1037 1371 TAD (MESS21
1040 3026 DCA HEAD2
1041 6777 QUAD0, SDRD
1042 6771 IOT31, SDSS
1043 9242 JMP
1044 6773 IOT32, SDSQ
1045 5250 JMP
1046 4775 JMS ERROR2
1047 5274 JMP
1050 6771 IOT33, SDSS
1051 7410 SKP
1052 5250 JMP
1053 6773 IOT34, SDSQ
1054 5257 JMP
1055 4775 JMS ERROR2
1056 5274 JMP
1057 6771 IOT35, SDSS
1060 5257 JMP
1061 6773 IOT36, SDSQ
1062 5245 JMP
1063 4775 JMS ERROR2
1064 5274 JMP
1065 1370 TAD (MESS22
1066 3026 DCA HEAD2
1067 6771 IOT37, SDSS
1070 7410 SKP
1071 5287 JMP
1072 6773 IOT38, SDSQ
1073 4775 JMS ERROR2
1074 7604 QBLUP, LAS
1075 7710 SPA CLA
1076 5235 JMP
1077 1367 TAD QUAD2*4
1080 3026 DCA HEAD2 (MESS23
1081 6773 QUAD1, SDSQ
1082 4775 JMS ERROR2
1083 7604 LAS
1084 7710 SPA CLA
1085 5381 JMP QUAD1
1086 1366 TAD (MESS24
1087 3026 DCA HEAD2
1090 6773 QUAD2, SDSQ
1091 5310 JMP
1092 6007 CAF
1093 7604 LAS

```

```

1114 7710 SPA CLA /LOOP?
1115 5310 JMP QUAD2 /YES
1116 6773 10T39, S0S0 /DID FLAG CLEAR?
1117 5324 JMP QUAD3=4 /YES
1118 4775' JMS ERROR2 /NO, QUAD LINE FLAG NOT CLEARED BY CAF
1121 7604 LAS
1122 7710 SPA CLA /LOOP?
1123 5310 JMP QUAD2 /YES
1124 1376 TAD (1000
1125 6774 10T40, S0LC /LOAD COMMAND REGISTER AGAIN
1126 1345 TAD (MESS25
1127 3026 DCA HEAD2
1130 6773 QUA03, S0S0 /WAIT FOR QUAD FLAG
1131 5330 JMP ,=I
1132 7200 CLA
1133 6775 10T41, S0LD /ISSUE S0LD TO CLEAR QUAD FLAG
1134 7604 LAS
1135 7710 SPA CLA /LOOP?
1136 5330 JMP QUAD3 /YES
1137 6773 10T42, S0S0 /FLAG STILL UP?
1138 5764' JMP QUAD4=2 /NO
1141 4775' JMS ERROR2 /YES, ERROR, QUAD FLAG NOT CLEARED BY S0LD
1142 7604 LAS
1143 7710 SPA CLA /LOOP?
1144 5330 JMP QUAD3 /YES
1145 5764' JMP QUAD4=2
1164 1200
1165 5656
1166 5635
1167 5615
1170 5564
1171 5546
1172 5516
1173 5471
1174 0600
1175 0537
1176 1000
1177 5441
1200

```

PAGE

```

1200 1377 TAD (MESS26
1201 3026 DCA HEAD2
1202 6773 QUA04, S0S0 /WAIT FOR QUAD FLAG
1203 5202 JMP ,=I
1204 7200 CLA
1205 6776 10T43, S0RC /ISSUE S0RC TO CLEAR QUAD FLAG
1206 7604 LAS
1207 7710 SPA CLA /LOOP?
1210 5202 JMP QUAD4 /YES
1211 4773 10T44, S0S0 /FLAG CLEARED?
1212 5217 JMP QUAD5=2 /YES
1213 4776' JMS ERROR2 /NO, ERROR, QUAD FLAG NOT CLEARED BY S0RC
1214 7604 LAS
1215 7710 SPA CLA /LOOP?

```

```

1216 5202 JMP QUAD4 /YES
1217 1375 TAD (MESS27
1220 3026 QUA05, DCA HEAD2
1221 6773 S0S0 /WAIT FOR QUAD FLAG
1222 5221 JMP ,=I
1223 7200 CLA
1224 6777 10T45, S0RD /ISSUE S0RD TO CLEAR QUAD LINE FLAG
1225 7604 LAS
1226 7710 SPA CLA /LOOP?
1227 5221 JMP QUAD5 /YES
1230 6773 10T46, S0S0 /FLAG CLEARED?
1231 5236 JMP QUAD6=2 /YES
1232 4776' JMS ERROR2 /NO, ERROR, QUAD FLAG NOT CLEARED BY S0RD
1233 7604 LAS
1234 7710 SPA CLA /LOOP?
1235 5221 JMP QUAD5 /YES
1236 1374 TAD (MESS28
1237 3026 DCA HEAD2
1238 6773 QUA06, S0S0 /WAIT FOR QUAD FLAG
1241 5240 JMP ,=I
1242 6772 10T47, S0ST /ISSUE S0ST
1243 7000 NOP
1244 6771 10T48, S0SS /S0SS
1245 7000 NOP
1246 7200 CLA
1247 1373 TAD (1000
1250 6774 10T49, S0LC /AND S0LC
1251 7604 LAS
1252 7710 SPA CLA /LOOP?
1253 5240 JMP QUAD6 /YES
1254 6773 10T50, S0S0 /DID STYT, S0SS, OR S0LC CLEAR FLAG?
1255 7410 SKP /YES
1256 5243 JMP QUAD7=2 /NO
1257 4776' JMS ERROR2 /ERROR, S0ST, S0SS, OR S0LC CLEARED QUAD FLAG
1260 7604 LAS
1261 7710 SPA CLA /LOOP?
1262 5240 JMP QUAD6 /YES
1263 1372 TAD (MESS29
1264 3026 DCA HEAD2
1265 7300 QUA07, CLA CLL /SET LOOP COUNT TO=2
1266 1371 TAD (=2
1267 3022 DCA CNTR1
1270 6775 10T51, S0LD /CLEAR QUAD FLAG PLIP/PLOPS
1271 6771 10T52, S0SS /WAIT FOR SINGLE LINE
1272 5271 JMP ,=I /TO COME
1273 6771 10T53, S0SS /GO AWAY
1274 7410 SKP
1275 5274 JMP ,=I
1276 6771 10T54, S0SS /AND COME AGAIN
1277 5276 JMP ,=I /
1300 2022 198 CNTR1 /TWICE THRU?
1301 5270 JMP QUAD7=3 /NO
1302 7604 LAS /YES

```

```

1303 7710 SPA CLA /LOOP?
1304 5265 JHP QUAD7 /YES
1305 6773 10T55, SDSC /NO, IS QUAD FLAG UP?
1306 7410 SKP /NO
1307 4776' JMS ERROR2 /YES, ERROR QUAD FLAG COUNTER FLIP/FLOPS NOT CLEARED
1310 7604 LAS /BY S0LD
1311 7710 SPA CLA /LOOP?
1312 5245 JHP QUAD7 /YES
1313 4046 JMS LOOP1
1314 5770' JHP QUAD

```

/CHECK TIMING ERROR SKIP INSTRUCTION AND LOGIC

```

1315 7300 TIMING, CLA CLL
1316 1367 TAD (MESS30
1317 3025 DCA HEAD1
1320 1366 TAD (MESS31
1321 3026 DCA HEAD2
1322 6774 SDLC /CLEAR ALL FLAGS INITIALLY
1323 1373 TAD (1000
1324 6774 10T50, SDLC /LOAD COMMAND REGISTER WITH 00,FWD,GO,READ
1325 6771 10T57, SDSS /WAIT FOR SINGLE
1326 5325 JHP /LINE FLAG
1327 6772 10T50, SDST =1 /SKIP ON TIMING ERROR
1330 7410 SKP
1331 4776' JMS ERROR2 /ERROR, SDST SHOULD NOT HAVE SKIPPED
1332 1365 TAD (MESS32
1333 3026 DCA HEAD2
1334 6773 TIME0, SDSC /WAIT FOR QUAD FLAG
1335 5334 JHP =1
1336 7200 CLA
1337 3022 DCA CNTR1
1340 2022 1SE CNTR1 /WAIT A WHILE SO THAT TIMING ERROR
1341 5340 JHP =1 /CAN SET
1342 6772 10T59, SDST =1 /TIMING ERROR SET?
1343 7410 SKP /NO
1344 5351 JHP TIME1=2 /YES
1345 4776' JMS ERROR2 /ERROR, TIMING ERROR NOT SET IN READ MODE
1346 7604 LAS
1347 7710 SPA CLA /LOOP?
1350 5334 JHP TIME0 /YES
1351 1364 TAD (MESS33
1352 3026 DCA HEAD2
1353 6772 TIME1, SDST /TIMING ERROR STILL SET?
1354 4776' JMS ERROR2 /TIMING ERROR CLEARED BY SDST
1355 7604 LAS
1356 7710 SPA CLA /LOOP?
1357 5353 JHP TIME1 /YES
1360 5763' JHP TIME2=2
1363 1400
1364 6140
1365 6101
1366 6052
1367 6026

```

```

1370 1024
1371 7776
1372 5773
1373 1000
1374 5744
1375 5722
1376 0537
1377 5700
1400

```

PAGE

```

1400 1377 TAD (MESS34
1401 3026 DCA HEAD2
1402 6772 TIME2, SDST /WAIT FOR TIMING ERROR
1403 5202 JHP =1
1404 6007 CAF /CLEAR FLAG WITH CAF
1405 7604 LAS
1406 7710 SPA CLA /LOOP?
1407 5202 JHP TIME2 /YES
1410 6772 10T60, SDST /DID FLAG CLEAR?
1411 5216 JHP TIME3=4
1412 4776' JMS ERROR2 /NO, TIMING ERROR NOT CLEARED BY CAF
1413 7604 LAS
1414 7710 SPA CLA /LOOP?
1415 5202 JHP TIME2 /YES
1416 1375 TAD (1000
1417 6774 10T61, SDLC /LOAD COMMAND REGISTER AGAIN

1420 1374 TAD (MESS35
1421 3026 DCA HEAD2
1422 6772 TIME3, SDST /WAIT FOR TIMING ERROR
1423 5222 JHP =1
1424 6776 10T62, SDRC /READ DECI:PE COMMAND REGISTER FOR STATUS
1425 3021 DCA IN /SAVE
1426 7604 LAS
1427 7710 SPA CLA /LOOP?
1430 5222 JHP TIME3 /YES
1431 1021 TAD IN /GET STATUS BACK AGAIN
1432 0373 AND (100 /MASK TO BIT 5
1433 7448 SEA /TIMING ERROR STATUS SET?
1434 5241 JHP TIME4=4 /YES, OK
1435 4776' JMS ERROR2 /NO, ERROR, TIMING ERROR STATUS NOT SET
1436 7604 LAS
1437 7710 SPA CLA /LOOP?
1440 5222 JHP TIME3 /YES
1441 1375 TAD (1000
1442 6774 10T63, SDLC /LOAD COMMAND REGISTER AGAIN
1443 1372 TAD (MESS36
1444 3026 DCA HEAD2
1445 6772 TIME4, SDST /WAIT FOR TIMING ERROR
1446 5245 JHP =1
1447 6774 10T64, SDLC /CLEAR FLAG WITH SDLC
1450 7604 LAS
1451 7710 SPA CLA /LOOP?
1452 5245 JHP TIME4 /YES

```

/TDBE DIAGNOSTIC

PAL10 V141 19OCT-72

11189 PAGE 1614

1433 6772
1454 5281
1455 4776'
1456 7684
1457 7718
1468 5243

1461 1371
1462 3826
1463 7388
1464 1378
1465 6774
1466 6771
1467 5266
1478 6776
1471 8367
1472 1366
1473 7648
1474 5266
1475 6776
1476 8365
1477 1344
1588 6774
1581 3822
1582 2822
1583 9382
1584 6772
1585 4776'
1586 1363
1587 3826
1518 6776
1511 8364
1512 7648
1513 4776'
1514 7684
1515 7718
1516 5263

10T65, SDST
JMP TIME3=2
JMS ERROR2
LAS
SPA CLA
JMP TIME4

TAD (MESS37
DCA HEAD2
TIME5, CLA CLL (3888
TAD
10T66, SDLC
10T67, SDSS
JMP =1
10T68, SDRC
AND (77
TAD (=22
SEA CLA
JMP =6
10T69, SDRC (7888
AND (488
TAD
10T70, SDLC
DCA CNTR1
ISE CNTR1
JMP =1
10T71, SDST
JMS ERROR2
TAD (MESS38
DCA HEAD2
SDRC (488
AND
SEA CLA
JMS ERROR2
LAS
SPA CLA
JMP TIME5

/DID FLAG CLEAR?
/YES
/NO, TIMING ERROR NOT CLEARED BY SOLC
/LOOP?

/GET TAPE MOVING BACKWARD
/WAIT FOR END ZONE

/SET "WRITE"

/WAIT A WHILE
/TIMING ERROR?
/NO, ERROR

/YES, READ STATUS
/"WRITE" CLEARED
/NO, ERROR
/LOOP?
/YES

1917 1362
1928 3826
1921 7388
1922 1375
1923 6774
1924 6771
1925 9324
1926 1361
1927 3823
1938 3822
1931 6776
1932 6777
1933 6775
1934 2822
1935 9331

TAD (MESS39
DCA HEAD2
TIME6, CLA CLL (1888
TAD
10T73, SDLC
10T74, SDSS
JMP =1
TAD (=8
DCA CNTR2
DCA CNTR1
10T75, SDRC
10T76, SDRD
10T77, SDLD
ISE CNTR1
JMP =4

/SET UNIT 8 RUNNING FORWARD
/WAIT FOR "UP TO SPEED"

/ISSUE MANY SDRC,SDRD,SOLC'S

/TDBE DIAGNOSTIC

PAL10 V141 19OCT-72

11189 PAGE 1615

1936 2823
1937 9331
1948 7684
1941 7718
1942 5327
1943 6772
1944 4776'
1945 7684
1946 7718
1947 5321
1958 4848
1951 5768'
1952 5757'
1957 1688
1968 1315
1961 7773
1962 6324
1963 6276
1964 8488
1965 7888
1966 7756
1967 8877
1978 3888
1971 6252
1972 6231
1973 8188
1974 6177
1975 1888
1976 8537
1977 6157
1688

ISE CNTR2
JMP =6
LAS
SPA CLA
JMP TIME6+6
10T78, SDST
JMS ERROR2
LAS
SPA CLA
JMP TIME6
JMS LOOP1
JMP TIMING
JMP UTSHRK

/LOOP?
/YES
/TIMING ERROR?
/NO, ERROR
/LOOP?
/YES

PAGE

/CHECK UP TO SPEED CIRCUITRY TO CLEAR MARK TRACK WINDOW

1688 7388
1681 1377
1682 3823
1683 1376
1684 3826
1685 6774
1686 1375
1687 6774
1618 7684
1611 7718
1612 5288
1613 6776
1614 8374
1615 7448
1616 4773'
1617 7684
1628 7718
1621 5288

UTSHRK, CLA CLL
TAD (MESS43
DCA HEAD1
TAD (MESS44
DCA HEAD2
10T82, SDLC
TAD (1888
10T83, SDLC
LAS
SPA CLA
JMP UTSHRK
10T84, SDRC
AND (77
SEA
JMS ERROR2
LAS
SPA CLA
JMP UTSHRK

/CLEAR STOP/GO BIT

/SET STOP/GO BIT

/LOOP?
/YES
/READ MARK TRACK
/ZERO?
/NO, ERROR
/LOOP?
/YES

1622	1372	TAD	(MESS45	
1623	3826	DCA	HEAD2	
1624	7388	UTSHK1, CLA CLL		
1625	1375	TAD	(1888	/SET STOP/80 BIT,
1626	6774	10T85, SDLC		
1627	6771	10T86, SDSS		/SINGLE LINE FLAG?
1630	5227	JMP	,=1	/NO
1631	6776	10T87, SORC		/YES, READ MARK TRACK
1632	8374	AND	(77	
1633	7658	SNA CLA		/ZERO?
1634	5227	JMP	,=5	/YES, TRY AGAIN
1635	6774	10T88, SDLC		/CLEAR STOP/GO BIT
1636	7684	LAS		
1637	7718	SPA CLA		/LOOP?
1640	5224	JMP	UTSHK1	/YES
1641	6776	10T89, SORC		/READ MARK TRACK
1642	8374	AND	(77	
1643	7448	SEA		/ZERO?
1644	4773	JMS	ERROR2	/NO, ERROR
1645	7684	LAS		
1646	7718	SPA CLA		/LOOP?
1647	5224	JMP	UTSHK1	/YES
1650	1371	TAD	(MESS46	
1651	3826	DCA	HEAD2	
1652	7388	UTSHK2, CLA CLL		
1653	1378	TAD	(3888	/SET STOP/80 AND PWD/REV
1654	6774	10T90, SDLC		
1655	6771	10T91, SDSS		
1656	5255	JMP	,=1	
1657	6776	10T92, SORC		
1660	8374	AND	(77	
1661	7658	SNA CLA		
1662	5255	JMP	,=5	
1663	1375	TAD	(1888	/CLEAR PWD/REV (BIT1)
1664	6774	10T93, SDLC		
1665	7684	LAS		
1666	7718	SPA CLA		
1667	5252	JMP	UTSHK2	
1670	6776	10T94, SORC		
1671	8374	AND	(77	
1672	7448	SEA		/MARK TRACK ZERO?
1673	4773	JMS	ERROR2	/NO, ERROR
1674	7684	LAS		
1675	7718	SPA CLA		
1676	5252	JMP	UTSHK2	
1677	1367	TAD	(MESS47	
1788	3826	DCA	HEAD2	
1781	7388	UTSHK3, CLA CLL		
1782	1375	TAD	(1888	/SET STOP/80, CLEAR PWD/REV (BIT1)
1783	6774	10T95, SDLC		
1784	6771	10T96, SDSS		
1785	5384	JMP	,=1	
1786	6776	10T97, SORC		

1787	8374	AND	(77	
1718	7658	SNA CLA		
1711	5384	JMP	,=5	
1712	1378	TAD	(3888	/SET PWD/REV (BIT 1)
1713	6774	10T98, SDLC		
1714	7684	LAS		
1715	7718	SPA CLA		
1716	5381	JMP	UTSHK3	
1717	6776	10T99, SORC		
1720	8374	AND	(77	
1721	7448	SEA		/MARK TRACK ZERO?
1722	4773	JMS	ERROR2	/NO, ERROR
1723	7684	LAS		
1724	7718	SPA CLA		
1725	5381	JMP	UTSHK3	
1726	5766	JMP	UTSHK4=2	
1766	2888			
1767	6517			
1770	3888			
1771	6466			
1772	6441			
1773	8537			
1774	8877			
1775	1888			
1776	6415			
1777	6352			
	2888	PAGE		
2888	1377	TAD	(MESS48	
2881	3826	DCA	HEAD2	
2882	7388	UTSHK4, CLA CLL		
2883	1376	TAD	(1888	/SET STOP/80, CLEAR UNIT (BIT8)
2884	6774	10T100, SDLC		
2885	6771	10T101, SDSS		
2886	5285	JMP	,=1	
2887	6776	10T102, SORC		
2818	8375	AND	(77	
2811	7698	SNA CLA		
2812	5285	JMP	,=5	
2813	1374	TAD	(5827	/SET UNIT (BIT8)
2814	6774	10T103, SDLC		
2815	7684	LAS		
2816	7718	SPA CLA		
2817	5282	JMP	UTSHK4	
2828	6776	10T104, SORC		
2821	8375	AND	(77	
2822	7448	SEA		/MARK TRACK 8?
2823	4773	JMS	ERROR2	/NO
2824	7684	LAS		
2825	7718	SPA CLA		
2826	5282	JMP	UTSHK4	
2827	7684	LAS		
2838	7818	RAR		
2831	7638	SEL CLA		/IS THERE A UNIT??

```

2832 5262          JMP      UTSHK6          /NO

2833 1372          TAD      (MESS49
2834 3826          DCA      HEAD2
2835 7388          UTSHK5, CLA CLL
2836 1374          TAD      (5888          /SET STOP/GO, UNIT (BIT8)
2837 6774          IOT105, SOLC
2840 6771          IOT106, SDSS
2841 9248          JMP      ,=1
2842 6776          IOT107, SDRC
2843 8375          AND      (77
2844 7658          SNA CLA
2845 9248          JMP      ,=5
2846 1376          TAD      (1888          /CLEAR UNIT (BIT2)
2847 6774          IOT108, SOLC
2850 7684          LAS
2851 7718          SPA CLA
2852 5235          JMP      UTSHK5
2853 6776          IOT109, SDRC
2854 8375          AND      (77
2855 7448          SEA
2856 4773'        JMS     ERROR2          /MARK TRACK ZERO?
2857 7684          LAS
2858 7718          SPA CLA
2861 5235          JMP      UTSHK5
2862 4846          UTSHK6, JMS     LOOP1
2863 9771'        JMP      UTSHRK
2864 1378          TAD      (4888
2865 6774          IOT110, SOLC          /STOP UNIT 1 IF MOVING
2866 7684          LAS
2867 7886          RTL
2870 7718          SPA CLA
2871 5767'        JMP      DAYREC
2872 5766'        JMP      XFER

/ROUTINE TO SEARCH AND FIND ALL BLOCK NUMBERS
/THE RIGHT HAND REEL MUST HAVE AT LEAST FOUR TURNS OF TAPE ON IT

2100 2100          *2100
2100 7388          BLKCH, CLA      CLL
2101 3896          DCA      DISBL
2102 1166          TAD      (=8782
2103 3868          DCA      BLKCN
2104 4861          JMS     BLKREV
2105 4878          JMS     BLKEND
2106 1165          TAD      (1888
2107 1765'        TAD      UNIT
2110 6774          IOT171, SOLC
2111 4764'        JMS     RDSUAD
2112 4764'        JMS     RDSUAD
2113 4181          FBLKCT, JMS     BLKSER
2114 1857          TAD      DISDA
2115 7841          CIA
2116 1856          TAD      DISBL
2117 7648          SEA      CLA
2120 5346          JMP      BLKERR          /BLOCKS DIDN'T COMPARE
    
```

```

2121 2856          ISE     DISBL
2122 2868          ISE     BLKCN
2123 5313          JMP     FBLKCT
2124 4878          JMS     BLKEND
2125 1164          TAD      (=2781
2126 3896          DCA      DISBL
2127 1166          TAD      (=8782
2128 3868          DCA      BLKCN
2131 4861          JMS     BLKREV
2132 4181          RBLKCT, JMS     BLKSER
2133 1857          TAD      DISDA
2134 7841          CIA
2135 1856          TAD      DISBL
2136 7648          SEA      CLA
2137 5346          JMP     BLKERR
2140 7848          CMA
2141 1856          TAD      DISBL
2142 3856          DCA      DISBL
2143 2868          ISE     BLKCN
2144 5332          JMP     RBLKCT
2145 5388          JMP     BLKCH

2146 7388          BLKERR, CLA     CLL
2147 1856          TAD      DISBL
2150 7482          HLT
2151 7288          CLA
2152 1857          TAD      DISDA
2153 7482          HLT
2154 5388          JMP     BLKCH          /AC=THE BLOCK NUMBER THAT WAS BEING SEARCHED FOR
                          /AC=THE BLOCK NUMBER THAT WAS FOUND
                          /RETURN TO START OF ROUTINE

2164 4787
2165 2234
2166 3888
2167 8281
2170 4888
2171 1688
2172 6684
2173 8537
2174 9888
2175 8877
2176 1888
2177 6558
2200 2288          PAGE

/TAPE 2
/ROUTINE TO RUN FROM END ZONE TO END ZONE
/AND DISPLAY THE CURRENT BLOCK NUMBER IN THE AC

2200 7388          DBLOCK, CLA CLL
2201 3233          DCA      DISBLK          /ZERO DISBLK
2202 1377          TAD      (3888
2203 1234          TAD      UNIT
2204 6774          IOT111, SOLC          /LOAD CONTROL WITH UNIT REV GO READ
    
```

```

2205 7300
2206 6771 DISLUP, CLA CLL
2207 5286 JMP SDSS /WAIT FOR SINGLE LINE FLAG
2210 7300 CLA CLL
2211 6777 IOT112, SDRD /READ DATA BUFFER
2212 3236 DCA DISDAT /AND SAVE
2213 6776 IOT113, SDRD /READ MARK TRACK
2214 8376 AND (77
2215 1375 TAD (=26
2216 7448 SEA
2217 5224 JMP OISEND /BLOCK NUMBER?
2220 1236 TAD DISDAT /NO, CHECK FOR END ZONE
2221 2233 ISE DISBLK /YES, DISPLAY BLOCK NUMBER
2222 5221 JMP (=4
2223 5286 JMP DISLUP (=4
2224 1374 OISEND, TAD SEA CLA /END ZONE?
2225 7648 JMP DISLUP /NO, LOOP
2226 5286 IOT114, SDRD /YES, EXTRACT DIRECTION BIT
2227 6776 RTL /AND COMPLEMENT
2230 7806 CML RTR
2231 7832 JMP DISLUP=2 /GO LOAD INTO CONTROL
2232 5284
2233 8888 DISBLK, 0
2234 8888 UNIT, 0
2235 8888 DISTRK, 0
2236 8888 DISDAT, 0
/ROUTINE TO ROCK DECTAPE UNIT 0
/FOR A DISTANCE DETERMINED BY ACS

2237 7300 ROCK, CLA CLL
2240 1373 TAD (1888
2241 6774 IOT115, SOLC
2242 7684 LAS
2243 7848 CMA
2244 3881 DCA 1
2245 2888 ISE 0
2246 5245 JMP (=4
2247 2881 ISE 1
2248 5245 JMP (=3
2251 7888 NOP
2252 1377 TAD (3888
2253 6774 IOT116, SOLC
2254 7684 LAS
2255 7848 CMA
2256 3881 DCA 1
2257 2888 ISE 0
2260 5257 JMP (=4
2261 2881 ISE 1
2262 5257 JMP (=3
2263 5248 JMP ROCK+1
2264 8888 MESSAGE, 0
2265 3315 DCA MPNTR
2266 1715 TAD I MPNTR
2267 7812 RTR
2270 7812 RTR
    
```

```

2271 7812 RTR
2272 8376 AND (77
2273 7458 SNA
2274 5664 JMP I MESSAGE
2275 1372 TAD (=48
2276 7518 SPA
2277 1371 TAD (188
2300 1370 TAD (248
2301 4831 JMS TYPE
2302 1715 TAD I MPNTR
2303 8376 AND (77
2304 7458 SNA
2305 5664 JMP I MESSAGE
2306 1372 TAD (=48
2307 7518 SPA
2310 1371 TAD (188
2311 1370 TAD (248
2312 4831 JMS TYPE
2313 2315 ISE MPNTR
2314 5266 JMP MESSAGE+2
2315 8888 MPNTR, 0
2316 8888 OPRINT, 0
2317 3348 DCA ONUMB
2320 1367 TAD (=4
2321 3341 DCA OCNT
2322 1348 TAD ONUMB
2323 7884 RAL
2324 7884 OPLOOP, RAL
2325 7884 RTL
2326 3348 DCA ONUMB
2327 1348 TAD ONUMB
2330 8366 AND (7
2331 1365 TAD (=268
2332 4831 JMS TYPE
2333 1348 TAD ONUMB
2334 2341 ISE OCNT
2335 5324 JMP OPLOOP
2336 7288 CLA
2337 5716 JMP I OPRINT
2340 8888 ONUMB, 0
2341 8888 OCNT, 0
2365 8268
2366 8887
2367 7774
2370 8248
2371 8188
2372 7748
2373 1888
2374 8884
2375 7752
2376 8877
2377 3888
2488
    
```

2400	7300	BLKCHK, CLA CLL		
2401	1377	TAD	(3000)	/START TAPE MOVING BACKWARD
2402	6774	10T117, SDLC		
2403	4315	JMS	RD6HRK	/WAIT FOR WINDOW TO OPEN
2404	4307	END2, JMS	RD1HRK	/READ BACK MARK TRACK
2405	1374	TAD	(=22)	
2406	7640	SZA CLA		/ENDZONE?
2407	9204	JMP	,=3	/NO
2410	6776	10T118, SDRG		/TURN AROUND
2411	7006	RTL		
2412	7032	CML RTR		
2413	6774	10T119, SDLC		
2414	4315	JMS	RD6HRK	/WAIT FOR WINDOW TO OPEN
2415	4307	JMS	RD1HRK	/READ MARK TRACK
2416	1375	TAD	(=26)	
2417	7650	SNA CLA		/BLOCK NUMBER?
2420	9236	JMP	RVGARD	/YES, GO CHECK REVERSE GUARD
2421	9215	JMP	,=4	/NO, LOOK AGAIN
2422	4315	FNDEXP, JMS	RD6HRK	/READ MARK TRACK
2423	1374	TAD	(=25)	
2424	7440	SZA		/EXPAND CODE?
2425	7402	HLT		/NO, ERROR
2426	4315	BLKHRK, JMS	RD6HRK	/READ MARK TRACK
2427	1375	TAD	(=26)	
2430	7450	SNA		/BLOCK NUMBER?
2431	9236	JMP	RVGARD	/YES, GO CHECK REVERSE GUARD
2432	7001	IAC		/NO
2433	7440	SZA		/EXPAND CODE?
2434	7402	HLT		/NO, UNKNOWN
2435	9204	JMP	END2	/YES, EXPAND CODE, GO LOOK FOR ENDZONE
2436	4315	RVGARD, JMS	RD6HRK	/GET MARK TRACK
2437	1373	TAD	(=32)	
2440	7440	SZA		/REVERSE GUARD?
2441	7402	HLT		/NO, ERROR
2442	1372	TAD	(=4)	/SET UP
2443	3000	DCA	B	/FOR 4 MARKS
2444	4315	JMS	RD6HRK	/GET MARK TRACK
2445	1371	LOCK, TAD	(=10)	
2446	7440	SZA		/LOCK, REV CHKSM, REV FINAL, REV PRE=FINAL?
2447	7402	HLT		/NO, ERROR
2450	2000	ISE	B	
2451	9244	JMP	,=5	
2452	1370	DATA, TAD	(=122)	/SET UP
2453	3000	DCA	B	/FOR 82 MARKS
2454	4315	JMS	RD6HRK	/GET MARK TRACK
2455	1367	TAD	(=70)	
2456	7440	SZA		/DATA MARK?
2457	7402	HLT		/NO, ERROR
2460	2000	ISE	B	
2461	9254	JMP	,=5	
2462	1372	PREFIN, TAD	(=4)	/SET UP
2463	3000	DCA	B	/FOR 4 MARKS
2464	4315	JMS	RD6HRK	/GET MARK TRACK
2465	1366	TAD	(=73)	
2466	7440	SZA		/PREFINAL, FINAL, CHKSM, REVLOCK?

2467	7402	HLT		/NO, ERROR
2470	2000	ISE	B	
2471	9204	JMP	,=5	
2472	4315	GUARD, JMS	RD6HRK	/GET MARK TRACK
2473	1360	TAD	(=51)	
2474	7440	SZA		/GUARD?
2475	7402	HLT		/NO, ERROR
2476	4315	REVBK, JMS	RD6HRK	/GET MARK TRACK
2477	1364	TAD	(=45)	
2500	7440	SZA		/REVERSE BLOCK NUMBER?
2501	7402	HLT		/NO, ERROR
2502	4315	REVEXP, JMS	RD6HRK	/GET MARK TRACK
2503	1374	TAD	(=25)	
2504	7440	SZA		/REVERSE EXPAND CODE?
2505	7402	HLT		/NO, ERROR
2506	9222	JMP	FNDEXP	
2507	0000	/READ 1 SHIFT OF MARK TRACK SUBROUTINE		
2510	6771	RD1HRK, B		
2511	9310	10T120, S0SS		
2512	6776	JMP	,=1	
2513	0363	10T121, SDRG		
2514	9707	AND	(77)	
		JMP I	RD1HRK	
		/READ 0 SHIFTS OF MARK TRACK SUBROUTINE		
2515	0000	RD6HRK, B		
2516	1362	TAD	(=6)	
2517	3307	DCA	RD1HRK	
2520	6771	10T122, S0SS		
2521	9320	JMP	,=1	
2522	6776	10T123, SDRG		
2523	2307	ISE	RD1HRK	
2524	9320	JMP	,=4	
2525	0363	AND	(77)	
2526	9715	JMP I	RD6HRK	
2527	7772			
2528	0077			
2529	7733			
2530	7727			
2531	7705			
2532	7710			
2533	7656			
2534	7770			
2535	7774			
2536	7746			
2537	7753			
2538	7752			
2539	7756			
2540	3000			
2541	2000			

PAGE

/CHECK SELECT ERROR STATUS BIT AND ABILITY TO CLEAR "WRITE"
 /UNIT 1 IS "OFF-LINE" OR NON-EXISTANT
 /UNIT 0 IS "ON LINE" AND "WRITE LOCKED"
 SELECT, CLA CLL

2601	1377	TAD	(MESS50	
2602	3025	DCA	HEAD1	
2603	1376	TAD	(MESS51	
2604	3026	DCA	HEAD2	
2605	6774	10T124, SDLC		
2606	6772	10T125, SDST		/IS TIMING ERROR SET?
2607	7410	SKP		
2610	4775	JMS	ERROR2	/YES, ERROR
2611	1374	TAD	(MESS52	
2612	3026	DCA	HEAD2	
2613	1373	TAD	(4000	
2614	6774	10T126, SDLC		/SET UNIT BIT TO 1
2615	7200	CLA		
2616	6776	10T127, SDRC		/READ STATUS
2617	3021	DCA	IN	/SAVE
2620	7604	LAS		
2621	7710	SPA CLA		/LOOP?
2622	5200	JMP	SELECT	/YES
2623	1021	TAD	IN	
2624	0372	AND	(100	
2625	7650	SNA CLA		/SELECT ERROR?
2626	4775	JMS	ERROR2	/NO, ERROR
2627	7604	LAS		
2630	7710	SPA CLA		/LOOP?
2631	5200	JMP	SELECT	/YES
2632	1371	TAD	(MESS53	
2633	3026	DCA	HEAD2	
2634	1370	SELECT1, TAD	(4000	
2635	6774	10T128, SDLC		/TRY TO SET "WRITE"
2636	7604	LAS		
2637	7710	SPA CLA		/LOOP?
2640	5234	JMP	SELECT1	/YES
2641	6776	10T129, SDRC		/READ STATUS
2642	3021	DCA	IN	/SAVE
2643	1021	TAD	IN	
2644	0367	AND	(400	
2645	7640	SEA CLA		/WRITE SET?
2646	4775	JMS	ERROR2	/YES, ERROR
2647	7604	LAS		
2650	7710	SPA CLA		/LOOP?
2651	5234	JMP	SELECT1	/YES
2652	1366	TAD	(MESS63	
2653	3026	DCA	HEAD2	
2654	6774	SELECT2, SDLC		/SELECT UNIT 0
2655	7604	LAS		
2656	7710	SPA CLA		/LOOP?
2657	5254	JMP	SELECT2	/YES
2660	6776	10T130, SDRC		/READ STATUS
2661	3021	DCA	IN	/SAVE
2662	1021	TAD	IN	
2663	0372	AND	(100	
2664	7640	SEA CLA		/SELECT ERROR?
2665	4775	JMS	ERROR2	/YES
2666	7604	LAS		
2667	7710	SPA CLA		/LOOP?

2670	5254	JMP	SELECT2	/YES
2671	4046	JMS	LOOP1	
2672	5200	JMP	SELECT	
		/CHECK WRITE LOCK OUT STATUS BIT AND ABILITY TO CLEAR "WRITE"		
		/UNIT 0 IS "WRITE=LOCKED"		
2673	7300	WL0, CLA	CLL	
2674	1365	TAD	(MESS54	
2675	3025	DCA	HEAD1	
2676	1364	TAD	(MESS55	
2677	3026	DCA	HEAD2	
2700	6774	SDLC		
2701	6776	10T131, SDRC		/READ STATUS
2702	3021	DCA	IN	/SAVE
2703	7604	LAS		
2704	7710	SPA CLA		/LOOP?
2705	5273	JMP	WL0	/YES
2706	1021	TAD	IN	
2707	0363	AND	(200	
2710	7650	SNA CLA		/WRITE LOCK OUT BIT SET?
2711	4775	JMS	ERROR2	/NO, ERROR
2712	7604	LAS		
2713	7710	SPA CLA		/LOOP?
2714	5273	JMP	WL0	/YES
2715	1362	TAD	(MESS56	
2716	3026	DCA	HEAD2	
2717	1367	WL1, TAD	(400	
2720	6774	10T132, SDLC		/TRY TO SET "WRITE"
2721	7604	LAS		
2722	7710	SPA CLA		/LOOP?
2723	5317	JMP	WL1	/YES
2724	6776	10T133, SDRC		/READ STATUS
2725	3021	DCA	IN	/SAVE
2726	1021	TAD	IN	
2727	0367	AND	(400	
2730	7640	SEA CLA		/WRITE SET?
2731	4775	JMS	ERROR2	/YES, ERROR
2732	7604	LAS		
2733	7710	SPA CLA		/LOOP?
2734	5317	JMP	WL1	/YES
2735	4046	JMS	LOOP1	
2736	5273	JMP	WL0	
2737	1361	TAD	(OK	
2740	4760	JMS	MESSAGE	
2741	4040	JMS	CRLF	
2742	4002	MLT		
2743	9342	JMP	OK	
2744	1713	OK, TEXT	"OK"	
2745	8000			
2700	2264			
2701	2744			
2702	6767			
2703	0200			
2704	6744			
2705	6730			

2766 7107
 2767 8409
 2770 4409
 2771 6706
 2772 8108
 2773 4808
 2774 6664
 2775 8537
 2776 6693
 2777 6648
 3888

PAGE

/TDBE READ-WRITE AND SEARCH TEST PROGRAM
 /COPYRIGHT 1971, DIGITAL EQUIPMENT CORP., MAYNARD, MASS, 01754

7200
 7400

BUFF1=7200
 BUFF2=7400

/ROUTINE TO CHECK DATA TRANSFERS ON TAPE

3888	7308	XFER,	CLA	CLL		
3881	3821		DCA		IN	
3882	1377		TAD		(FILL8	
3883	4224		JMS		WREAD	/FILL A BUFFER, THEN WRITE AND READ 818
3884	1376		TAD		(FILL1	
3885	4224		JMS		WREAD	/FILL A BUFFER, THEN WRITE AND READ 018
3886	1375		TAD		(FILL25	
3887	4224		JMS		WREAD	/FILL A BUFFER, THEN WRITE AND READ 2525
3889	1374		TAD		(FILPAT	
3811	4224		JMS		WREAD	/FILL A BUFFER, THEN WRITE AND READ 2225, /5522,2555
3812	1373		TAD		(FILINC	/INCREMENT PATTERN
3813	4224		JMS		WREAD	
3814	1372		TAD		(FILDEC	/DECREMENT PATTERN
3815	4224		JMS		WREAD	
3816	1371		TAD		(SPEC1	/6161
3817	4224		JMS		WREAD	
3820	1370		TAD		(SPEC2	/3434
3821	4224		JMS		WREAD	
3822	4747		JMS		PARGMT	
3823	5282		JMP		XFER=2	

/ROUTINE TO WRITE AND READ BACK AND COMPARE EVERY 160TH BLOCK ON TAPE

3824	8888	WREAD,	B			
3825	3838		DCA		FILPNT	
3826	3766		DCA		SUNIT	
3827	1365		TAD		(=888	
3830	4438		JMS		FILPNT	/FILL BUFF1 WITH DATA
3831	7200		BUFF1			
3832	3827		DCA		BLK	/ZERO BLOCK NUMBER
3833	1364		TAD		(MESS59	
3834	3825		DCA		HEAD1	
3835	1363	WRBL1,	TAD		(MESS60	
3836	3826		DCA		HEAD2	
3837	1827		TAD		BLK	

3840 4762'
 3841 7200
 3842 7400
 3843 1361
 3844 3826
 3845 1827
 3846 4760'
 3847 7400
 3848 7577
 3851 1365
 3852 4757'
 3853 7200
 3854 7401
 3855 1356
 3856 3826
 3857 1827
 3860 4755'
 3861 7400
 3862 7577
 3863 1827

		JMS		WRITE		/WRITE ONTO TAPE
		BUFF1				
		=200				
		TAD		(MESS59		
		DCA		HEAD2		
		TAD		BLK		
		JMS		READ		/READ BACK INTO MEMORY
		BUFF2				
		=201				
		TAD		(=888		
		JMS		COMPAR		/COMPARE DATA
		BUFF1				
		BUFF2=1				
		TAD		(MESS60		
		DCA		HEAD2		
		TAD		BLK		
		JMS		READR		/READ BACK BACKWARDS
		BUFF2				
		=201				
		TAD		BLK		/BUMP BLOCK NUMBER

3864 1354
 3865 3827
 3866 1827
 3867 1353
 3870 7718
 3871 5235
 3872 1352
 3873 3827
 3874 1351
 3875 3825
 3876 1350
 3877 3826
 3180 1827
 3181 4747'
 3182 7200
 3183 7400
 3184 1356
 3185 3826
 3186 1827
 3187 4755'
 3188 7400
 3189 7577
 3112 1365
 3113 4757'
 3114 7200
 3115 7401
 3116 1361
 3117 3826
 3120 1827
 3121 4760'
 3122 7400
 3123 7577
 3124 1827
 3125 1366

		TAD		(100		
		DCA		BLK		
		TAD		BLK		
		TAD		(=2701		
		JMP	CLA			
		JMP		WRBL1		/SET BLOCK NUMBER TO 2701
		TAD		(2701		
		DCA		BLK		
		TAD		(MESS61		
		DCA		HEAD1		
		TAD		(MESS62		
		DCA		HEAD2		
		TAD		BLK		
		JMS		WRITER		/WRITE ONTO TAPE BACKWARDS
		BUFF1				
		=200				
		TAD		(MESS60		
		DCA		HEAD2		
		TAD		BLK		
		JMS		READR		/READ BACK INTO MEMORY BACKWARDS
		BUFF2				
		=201				
		TAD		(=888		
		JMS		COMPAR		/COMPARE DATA
		BUFF1				
		BUFF2=1				
		TAD		(MESS59		
		DCA		HEAD2		
		TAD		BLK		
		JMS		READ		/READ BACK FORWARD
		BUFF2				
		=201				
		TAD		BLK		/BUMP BLOCK NUMBER
		TAD		(=888		

3126	3027	DCA	BLK	
3127	1027	TAD	BLK	
3130	7700	SMA	CLA	
3131	5276	JMP	WRRL2	
3132	7604	LAB		
3133	7810	RAR		
3134	7630	SZL	CLA	/TWO UNITS?
3135	5024	JMP I	WREAD	/NO
3136	1766	TAD	SUNIT	/YES, INCREMENT
3137	1345	TAD	(4000	/UNIT
3140	7430	SMA		
3141	5024	JMP I	WREAD	/EXIT IF BACK TO ZERO
3142	3766	DCA	SUNIT	/STORE BACK IF NO=ZERO
3143	5232	JMP	WRRL1-3	/LOOP
3145	4000			
3146	7700			
3147	3671			
3150	7075			
3151	7042			
3150	2701			
3153	7077			
3154	0100			
3155	3714			
3156	7070			
3157	3200			
3160	4600			
3161	7037			
3162	4472			
3163	7025			
3164	7012			
3165	7000			
3166	4471			
3167	3477			
3170	3461			
3171	3443			
3172	3421			
3173	3400			
3174	3276			
3175	3260			
3176	3243			
3177	3227			
	3200			

PAGE

```

/SUBROUTINE TO COMPARE TWO DATA BUFFERS, INDICATE AN ERROR
/CALLING SEQUENCE:
/ TAD (=-N /MINUS (219) NUMBER OF WORDS TO COMPARE
/ JMS COMPAR /CALL SUBROUTINE
/ CADD /1ST ADDRESS OF GOOD DATA
/ TADD /1ST ADDRESS OF TEST DATA
/ RETURN HERE WHEN DONE

```

3200	0000	COMPAR,	0	
3201	3224	DCA	CCNTR	
3202	1600	TAD I	COMPAR	

3203	3225	DCA	GPNTR	
3204	2200	ISE	COMPAR	
3205	1600	TAD I	COMPAR	
3206	3226	DCA	TPNTR	
3207	2200	ISE	COMPAR	
3210	1377	TAD	(DATMS	
3211	3776	DCA	DATMD	
3212	1625	CONLUP, TAD I	GPNTR	
3213	7041	CIA		
3214	1626	TAD I	TPNTR	
3215	7640	SZA	CLA	
3216	4775	JMS	DATERR	
3217	2225	ISE	GPNTR	
3220	2220	ISE	TPNTR	
3221	2224	ISE	CCNTR	
3222	5212	JMP	CONLUP	
3223	5600	JMP I	COMPAR	
3224	0000	CCNTR,	0	
3225	0000	GPNTR,	0	
3226	0000	TPNTR,	0	

```

/SUBROUTINE TO FILL MEMORY WITH ZEROS
/CALLING SEQUENCE:
/ TAD (=-N /MINUS (219) NUMBER OF WORDS TO FILL
/ JMS FILL0 /CALL SUBROUTINE
/ ADDR /1ST ADDRESS TO FILL

```

3227	0000	FILL0,	0	
3230	3241	DCA	FILL0C	
3231	1627	TAD I	FILL0	
3232	3242	DCA	FILL0P	
3233	2227	ISE	FILL0	
3234	3642	DCA I	FILL0P	
3235	2242	ISE	FILL0P	
3236	2241	ISE	FILL0C	
3237	5234	JMP	=S	
3240	5627	JMP I	FILL0	
3241	0000	FILL0C,	0	
3242	0000	FILL0P,	0	

```

/SUBROUTINE TO FILL MEMORY WITH 01 (7777)
/CALLING SEQUENCE:
/ TAD (=-N /MINUS (219) NUMBER OF WORDS TO FILL
/ JMS FILL1 /CALL SUBROUTINE
/ ADDR /1ST ADDRESS TO FILL

```

3243	0000	FILL1,	0	
3244	3296	DCA	FILL1C	
3245	1643	TAD I	FILL1	
3246	3297	DCA	FILL1P	
3247	2243	ISE	FILL1	
3250	7240	CLA	CMA	
3251	3657	DCA I	FILL1P	
3252	2297	ISE	FILL1P	

```

3293 2296      ISE  FILL1C
3294 2298      JMP  ,=4
3295 2443      JMP I  FILL1
3296 0000      FILL1C, 0
3297 0000      FILL1P, 0
/SUBROUTINE TO FILL MEMORY WITH 2525
/CALLING SEQUENCE)
/      TAD  (=N          /MINUS (218) NUMBER OF WORDS TO FILL
/      JMS  FILL25      /CALL SUBROUTINE
/      ADDR          /1ST ADDRESS TO FILL

3200 0000      FILL25, 0
3201 3273      DCA  FILL2C
3202 1040      TAD I  FILL25
3203 3275      DCA  FILL2P
3204 2260      ISE  FILL25
3205 1274      TAD  FILL2K
3206 3675      DCA I  FILL2P
3207 2275      ISE  FILL2P
3208 2273      ISE  FILL2C
3209 2273      JMP  ,=4
3210 2273      JMP I  FILL25
3211 2273      FILL2C, 0
3212 2273      FILL2P, 2525
3213 2273      FILL2K, 0

```

```

/SUBROUTINE TO FILL MEMORY WITH 2225,5522,2555
/CALLING SEQUENCE)
/      TAD  (=N          /MINUS (215) NUMBER OF WORDS TO FILL
/      JMS  FILPAT      /CALL SUBROUTINE
/      ADDR          /1ST ADDRESS TO FILL

```

```

3276 0000      FILPAT, 0
3277 3323      DCA  FILLC1
3278 1476      TAD I  FILPAT
3279 3321      DCA  FILLP1
3280 2276      ISE  FILPAT
3281 1325      FILPL1, TAD  FILTP
3282 3322      DCA  FILLP2
3283 1331      TAD  FILTC
3284 3324      DCA  FILLC2
3285 1722      FILPL2, TAD I  FILLP2
3286 3721      DCA I  FILLP1
3287 2321      ISE  FILLP1
3288 2323      ISE  FILLC1
3289 7410      SKP
3290 5676      JMP I  FILPAT
3291 2322      ISE  FILLP2
3292 2324      ISE  FILLC2
3293 5307      JMP  FILLP2
3294 5303      JMP  FILPL1
3295 0000      FILLP1, 0
3296 0000      FILLP2, 0
3297 0000      FILLC1, 0

```

```

3324 0000      FILLC2, 0
3325 3326      FILTP, ,=1
3326 2225      FILTP, 2225
3327 5522      FILTP, 5522
3328 2555      FILTP, 2555
3329 7775      FILTC, FILTP-FILTC+1
3330 4000
3331 4000
3332 4000
3333 4042      PAGE
3334 3400

```

```

/SUBROUTINE TO FILL MEMORY WITH AN INCREMENT PATTERN
/CALLING SEQUENCE)
/      TAD  (=N          /MINUS (215) NUMBER OF WORDS TO FILL
/      JMS  FILING      /CALL SUBROUTINE
/      ADDR          /1ST ADDRESS TO FILL

```

```

3400 0000      FILING, 0
3401 3216      DCA  FILICT
3402 1600      TAD I  FILING
3403 3217      DCA  FILIPT
3404 2200      ISE  FILING
3405 3220      DCA  FILIDT
3406 1220      TAD  FILIDT
3407 3617      DCA I  FILIPT
3408 2220      ISE  FILIDT
3409 7000      NOP
3410 2217      ISE  FILIPT
3411 2216      ISE  FILICT
3412 5206      JMP  ,=6
3413 5600      JMP I  FILING
3414 0000      FILICT, 0
3415 0000      FILIPT, 0
3416 0000      FILIDT, 0
3417 0000
3418 0000

```

```

/SUBROUTINE TO FILL MEMORY WITH A DECREMENT PATTERN
/CALLING SEQUENCE)
/      TAD  (=N          /MINUS (215) NUMBER OF WORDS TO FILL
/      JMS  FILDEC      /CALL SUBROUTINE
/      ADDR          /1ST ADDRESS TO FILL

```

```

3421 0000      FILDEC, 0
3422 3240      DCA  FILOCT
3423 1621      TAD I  FILODEC
3424 3241      DCA  FILODT
3425 2221      ISE  FILODEC
3426 3242      DCA  FILODT
3427 1242      TAD  FILODT
3428 3641      DCA I  FILODT
3429 7040      CMA
3430 1242      TAD  FILODT
3431 2241      ISE  FILODT
3432 2240      ISE  FILOCT
3433 5226      JMP  ,=7
3434 7200      CLA

```

3437 3621
3440 0000
3441 0000
3442 0000

JMP I FILDEC
FILDCT, B
FILDPT, B
FILDDT, B

/SUBROUTINE TO FILL MEMORY WITH 4161

/CALLING SEQUENCE1

/ TAD (N /MINUS (219) NUMBER OF WORDS TO FILL
/ JMS SPEC1 /CALL SUBROUTINE
/ ADDR /1ST ADDRESS TO FILL

3443 0000
3444 3296
3445 1643
3446 3297
3447 2243
3450 1200
3451 3697
3452 2297
3453 2296
3454 5200
3455 3643
3456 0000
3457 0000
3460 6161

SPEC1, B
DCA DCA SPEC1
TAD I TAD I SPEC1
DCA DCA SPEPT
ISE ISE SPEC1
TAD TAD SPEC1D
DCA I DCA I SPEPT
ISE ISE SPEPT
ISE ISE SPECT
JMP I,=4
JMP I SPEC1
SPECT, B
SPEPT, B
SPEC1D, 4161

/SUBROUTINE TO FILL MEMORY WITH 3434

/CALLING SEQUENCE1

/ TAD (N /MINUS (218) NUMBER OF WORDS TO FILL
/ JMS SPEC2 /CALL SUBROUTINE
/ ADDR /1ST ADDRESS TO FILL

3461 0000
3462 3274
3463 1641
3464 3275
3465 2261
3466 1276
3467 3675
3470 2275
3471 2274
3472 5266
3473 3661
3474 0000
3475 0000
3476 3434
3477 0000
3500 4040
3501 1377
3502 4776
3503 2021
3504 7000
3505 1021
3506 4775
3507 1374

SPEC2, B
DCA DCA SPECT
TAD I TAD I SPEC2
DCA DCA SPEPT
ISE ISE SPEC2
TAD TAD SPEC2D
DCA I DCA I SPEPT
ISE ISE SPEPT
ISE ISE SPECT
JMP I,=4
JMP I SPEC2
SPECT, B
SPEPT, B
SPEC2D, 3434
PASCNT, B
JMS CRLF
TAD (PASS
JMS MESSAGE
ISE IN
NOP
TAD IN
JMS DPRINT
TAD (COMP

3510 4776
3511 4040
3512 3677
3513 2001
3514 2323
3515 4000
3516 4003
3517 1715
3520 2014
3521 0324
3522 0300
3574 3516
3575 2316
3576 2244
3577 3513
3600

JMS MESSAGE
JMS CRLF
JMP I PASCNT
PASS, TEXT "PASS "
COMP, TEXT " COMPLETE"
PAGE

/REVERSE SEARCH SUBROUTINE

3600 0000
3601 3270
3602 1034
3603 3095
3604 1377
3605 1776
3606 6774
3607 0776
3610 0375
3611 7640
3612 9563
3613 4774
3614 4774
3615 6771
3616 7410
3617 6777
3620 6771
3621 5220
3622 6776
3623 2373
3624 1372
3625 7490
3626 5240
3627 1371
3630 7640
3631 5215
3632 6776
3633 7006
3634 7032
3635 2095
3636 5206
3637 5261
3640 6776
3641 7006

RSEARCH, B
DCA DCA RSLOOK
TAD TAD M10 /SET P A COUNT OF 10 TIMES
DCA DCA BLKTRY /TO SEARCH FOR A BLOCK
TAD TAD (1000
TAD TAD SUNIT
RSRCHB, SDLC
IOT134, SDRC
AND (100
SEA CLA
JMP I (SELEARN
JMS R0QUAD
JMS R0QUAD
RSRCH1, SDSS
SKP
IOT135, SDRO
IOT136, SDSS
JMP I,=1
IOT136, SDRC
AND (77
TAD TAD (=26 /BLOCK MARK
SNA
JMP RSRCH2 /YES
TAD TAD (4 /END ZONE
SEA CLA
JMP RSRCH1 /NO,GO READ AGAIN
IOT13A, SDRC
RTL /READ THE C.R,
CML RTR /SET THE DIRECTION BIT IN LINK
ISE BLKTRY /INCREMENT IT FOR TURN AROUND
JMP RSRCHB
JMP RSRCHB
RSRCH2, SDRC
RTL /COULDN'T FIND BLOCK AFTER 8 TRIES

```

3642 6777      10Y137, SDRD      /READ THE BLOCK NUMBER
3643 7041      CIA
3644 1270      TAD      RLOCK
3645 7450      SNA
3646 5265      JMP      RLOCSD
3647 7041      CIA
3650 7420      SNL
3651 1371      TAD      (4
3652 7630      SBL CLA
3653 5215      JMP      RSRCH1
3654 6776      RETURN, SDRC
3655 7056      RTL
3656 7032      CML RTR
3657 2055      ISE      BLKTRY
3658 5206      JMP      RSRCH0
3661 7200      CLA
3662 1270      TAD      RLOCK
3663 7402      MLY
3664 5263      JMP      /ACQTHE BLOCK THAT IT WAS LOOKING FOR
3665 7630      RLOCSD, SEL CLA /BUT FAILED TO FIND AFTER 10 TRIES;
3666 5215      JMP      RSRCH1
3667 5600      JMP I RSRCH
3670 0000      RLOCK, 0
/WRITE REVERSE SUBROUTINE

3671 0000      WRITER, 0
3672 3770      DCA      WCNT
3673 1691      TAD I WRITER
3674 3767      DCA I WADDR
3675 2371      ISE      WRITER
3676 1691      TAD I WRITER
3677 3766      DCA WCOUNT
3678 1271      TAD      WRITER
3681 7001      IAG
3682 3769      DCA      WRITE
3683 4764      JMS      CSUMRY      /CALCULATE THE CHECKSUM
3684 0025      25
3685 7177      BUFP1=1
3686 7000      =200
3687 4763      JMS      SDCXR
3688 4331      JMS      WRYLCK      /CHECK FOR WRITE LOCK OUT
3689 1770      TAD      WCNT
3690 4200      JMS      RSRCH
3691 5762      JMP      WRITE1
/READ REVERSE SUBROUTINE

3714 0000      READR, 0
3715 3761      DCA      RCNT
3716 1714      TAD I READR
3717 3760      DCA RADDR
3718 2314      ISE      READR
3719 1714      TAD I READR
3720 3757      DCA RCOUNT
3721 1314      TAD      READR
3722 7001      IAG
3723 1314      TAD      READR
3724 7001      IAG

```

```

3725 3756      DCA      READ
3726 1701      TAD      RCNT
3727 4200      JMS      RSRCH
3728 5755      JMP      READ1

3731 0000      WRYLCK, 0      /ROUTINE TO CHECK FOR WRITE LOCKOUT
3732 1776      TAD      SUNIT
3733 6774      10Y151, SDCG
3734 6776      10Y15A, SDRC
3735 0304      AND      (200
3736 7640      SZA CLA
3737 5562      JMP I CHROERR
3738 5731      JMP I WRYLCK

3754 0200
3755 4612
3756 4000
3757 4660
3758 4697
3761 4696
3762 4512
3763 4714
3764 4303
3765 4472
3766 4545
3767 4544
3770 4470
3771 0004
3772 7752
3773 0077
3774 4707
3775 0100
3776 4471
3777 1000
4000      PAGE

```

/DATA ERROR HANDLER

```

4000 0000      DATERR, 0
4001 7604      LAB
4002 0377      AND      (400
4003 7640      SZA CLA
4004 5233      JMP      DATHLT=3
4005 1240      TAD      DATNO
4006 7650      SNA CLA
4007 5220      JMP      DATNUM
4008 4200      JMS      HEADTP
4009 1240      TAD      DATNO
4010 4776      JMS      MESSAGE
4011 3240      DCA      DATNO
4012 4040      JMS      CRUF
4013 1375      TAD      (FORMT1
4014 4776      JMS      MESSAGE

```

```

4017 4040      JMS      CRLF
4020 1774'    DATNUM, TAD      CPNTR
4021 3241      DCA      DAPPNT
4022 1041      TAD I    DAPPNT
4023 4773'    JMS      OPRINT
4024 1372      TAD      (240
4025 4031      JMS      TYPE
4026 1771'    TAD      TPNTR
4027 3241      DCA      DAPPNT
4030 1041      TAD I    DAPPNT
4031 4773'    JMS      OPRINT
4032 4040      JMS      CRLF
4033 7004      LBS
4034 0370      AND      (200
4035 7050      SNA CLA
4036 7402      DATHLT, HLT
4037 5000      JMP I    DAYERR
4040 0000      DATHD, 0
4041 0000      DATPNT, 0
4042 0401      DATHE, TEXT  "DATA ERROR"
4043 2401
4044 4005
4045 2222
4046 1722
4047 0000

```

/SUBROUTINE TO TYPE OUT HEADER FOR DATA TESTS

```

4050 0000      HEADTP, 0
4051 4040      JMS      CRLF
4052 1347      TAD      (UMESS
4053 4776'    JMS      MESSAGE
4054 1372      TAD      (240
4055 4031      JMS      TYPE
4056 6776      IOT139, SDRC
4057 7710      SPA CLA
4060 7001      IAC
4061 1344      TAD      (200
4062 4031      JMS      TYPE
4063 4040      JMS      CRLF
4064 1365      TAD      (UMESS
4065 4776'    JMS      MESSAGE
4066 1372      TAD      (240
4067 4031      JMS      TYPE
4070 1027      TAD      BLK
4071 4773'    JMS      OPRINT
4072 4040      JMS      CRLF
4073 1025      TAD      HEAD1
4074 4776'    JMS      MESSAGE
4075 4040      JMS      CRLF
4076 1026      TAD      HEAD2
4077 4776'    JMS      MESSAGE
4100 4040      JMS      CRLF
4101 5050      JMP I    HEADTP
4102 2910      UMESS, TEXT  "UNITY"
4103 1124

```

```

4104 0000      BMESS, TEXT  "BLOCK"
4105 0214
4106 1703
4107 1300

```

/CHECKSUM ERROR HANDLER

```

4110 0000      CHKERR, 0
4111 3331      DCA      CHKDAT
4112 6776      IOT140, SDRC      /STOP TAPE
4113 0364      AND      (4000
4114 6774      IOT141, SOLC
4115 4250      JMS      HEADTP
4116 1343      TAD      (CHKMES
4117 4776'    JMS      MESSAGE
4120 4040      JMS      CRLF
4121 7004      LBS
4122 0370      AND      (200
4123 7640      SZA CLA
4124 5710      JMP I    CHKERR
4125 1331      TAD      CHKDAT
4126 7402      CHKHLT, HLT
4127 7200      CLA
4130 5710      JMP I    CHKERR
4131 0000      CHKDAT, 0
4132 0310      CHKMES, TEXT  "CHECKSUM ERROR"
4133 0503
4134 1323
4135 2915
4136 4005
4137 2222
4140 1722
4141 0000

```

PAGE

/WRITE LOCK OUT ERROR

```

4200 4777'    WR0ERR, JMS      HEADTP
4201 6776      IOT142, SDRC      /STOP TAPE

```

```

4202 0376 AND (4000
4203 0774 10T143, SOLC
4204 1375 TAO (WRDMES
4205 4774' JMS MESSAGE
4206 4040 JMS CRLF
4207 7004 LAB (200
4210 0373 AND
4211 7000 SNA CLA
4212 7402 WROHLT, HLT
4213 5772' JMP WRELS+0

4214 2316 WRDMES, TEXT "UNIT WRITE LOCKED"
4215 1114
4216 4027
4217 2211
4220 2403
4221 4014
4222 1703
4223 1305
4224 0400

```

/SELECT ERROR HANDLER

```

4225 4777' SELERR, JMS HEADTP
4226 1371 TAO (SELMES
4227 4774' JMS MESSAGE
4230 4040 JMS CRLF
4231 7004 LAB (200
4232 0373 AND
4233 7000 SNA CLA
4234 7402 SELHLT, HLT
4235 5772' JMP WRELS+0

4236 2309 SELMES, TEXT "SELECT ERROR"
4237 1405
4240 0304
4241 4005
4242 2222
4243 1722
4244 0000

```

/TIMING ERROR HANDLER

```

4245 0000 TYMERR, B
4246 6776 10T144, SRCQ (4000 /STOP TAPE
4247 0376 AND
4250 6774 10T145, SOLC
4251 4777' JMS HEADTP
4252 1370 TAO (TYMERR
4253 4774' JMS MESSAGE
4254 4040 JMS CRLF
4255 7004 LAB (200
4256 0373 AND

```

```

4257 7000 SNA CLA
4260 7402 TYMHLT, HLT
4261 5747' JMP WRELS+0
4262 2411 TYMERR, TEXT "TIMING ERROR"
4263 1311
4264 1607
4265 4005
4266 2222
4267 1722
4270 0000

```

/SUBROUTINE TO CLEAR WRITE AFTER QUAD LINE FLAG

```

4271 0000 CLRNT, B
4272 6773 10T140, SRCQ /WAIT FOR QUAD LINE FLAG
4273 5272 JMP ,=1
4274 6772 10T154, SRCQ /TIMING ERROR
4275 7010 SKP CLA /NO
4276 4161 JMS (TYMERR /YES
4277 6776 10T149, SRCQ /READ THE COMMAND REGISTER
4300 0346 AND (7000 /MASK OFF WRITE BIT
4301 6774 10T170, SOLC /LOAD THE COMMAND REGISTER
4302 5071 JMP ! CLRNT /EXIT

4303 0000 CSUMRT, B
4304 1703 TAO ! CSUMRT
4305 3769' DCA CHKSUM
4306 2303 IRR CSUMRT
4307 1703 TAO ! CSUMRT
4310 3017 DCA AUTO
4311 2303 IRR CSUMRT
4312 1703 TAO ! CSUMRT
4313 3322 DCA XXX
4314 2303 IRR CSUMRT
4315 1417 TAO ! AUTO
4316 4764' JMS SBOXOR
4317 2322 IRR XXX
4320 5315 JMP ,=0
4321 5703 JMP ! CSUMRT
4322 0000 XXX, B

4323 0000 CHKCHK, B
4324 4303 JMS CSUMRT
4325 0000 B
4326 7377 BUFF2=1
4327 7379 =203
4330 1769' TAO CHKSUM
4331 7040 CMA
4332 0363 AND (77
4333 7440 SBA
4334 4960 JMS ! (CHKERR /CHECK SUM ERROR
4335 5723 JMP ! CHKCHK /RETURN

4363 0077
4364 4714

```


4365 4744
4366 7888
4367 3887
4370 4242
4371 4236
4372 3837
4373 8288
4374 2244
4375 4214
4376 4888
4377 4888
4488

PAGE

/T08=EA READ=WRITE=AND=SEARCH SUBROUTINES
/COPYRIGHT 1971, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS, 01754
/DECTAPE COMMANDS

6771 S0SS=6771 /SKIP ON SINGLE LINE FLAG
6772 S0RT=6772 /SKIP ON TIMING ERROR
6773 S0SQ=6773 /SKIP ON QUADRUPLE LINE FLAG
6774 S0LC=6774 /LOAD COMMAND REGISTER
6775 S0LD=6775 /LOAD DATA REGISTER, CLEAR FLAGS
6776 S0RC=6776 /READ COMMAND REGISTER AND MARK TRACK, CLEAR FLAG
6777 S0RD=6777 /READ DATA REGISTER, CLEAR FLAGS

/SEARCH SUBROUTINE
/SUBROUTINE IS ENTERED WITH THE NUMBER OF THE DESIRED BLOCK IN THE AC
/PROGRAM WILL EXIT WITH TAPE MOVING IN THE FORWARD DIRECTION
/UNIT BIT IS IN SUNIT, BIT 0, BITS 1 TO 11 ARE 8

4488 8888 SEARCH, 8
4481 3278 DCA SLOOK /SAVE BLOCK NUMBER
4482 1894 TAD M18 /SET UP A COUNT OF 18
4483 3895 DCA BLNTRY /TO SEARCH FOR A BLOCK,
4484 1377 TAO (3888 / PUT IN MOTION BACKWARD
4485 1271 TAO SUNIT
4486 6774 SRCHB, S0LC /LOAD CONTROL WITH UNIT, REV, GO, READ
4487 6776 IOT146, S0RC /READ STATUS
4488 8376 AND (188
4489 7448 SZA CLA /SELECT ERROR?
4490 5963 JMP I (SELERR /YES
4491 4775' JMS RDBUAD /DELAY TO ASSURE
4492 4775' JMS RDBUAD /MARK WINDOW OPEN
4493 6771 SRCH1, S0SS /SINGLE LINE FLAG
4494 7418 SKP /NO
4495 6777 IOT147, S0RD /YES
4496 6771 IOT148, S0SS /SKIP ON SINGLE LINE FLAG
4497 5228 JMP =1
4498 6776 IOT149, S0RC /READ MARK TRACK AND COMMAND REGISTER
4499 8374 AND (77 /MASK TO MARK TRACK BITS
4500 1373 TAO (=26 /BLOCK MARK ?
4501 7488 SNA
4502 5248 JMP SRCH2 /YES, GO READ THE BLOCK NUMBER
4503 1372 TAO (4 /END ZONE ?

4438 7448 SZA CLA
4439 5215 JMP SRCH1 /NO, GO GET NEXT WORD
4440 6776 IOT144, S0RC /READ THE COMMAND REG;
4441 7886 RTL
4442 7832 CHL RTR /TURN THE TAPE AROUND
4443 2895 ISE BLNTRY /8 TRIES ?
4444 5286 JMP SRCHB /NO, TRY AGAIN
4445 5281 JMP BADBLK /YES, CAN NOT FIND BLOCK
4446 6776 SRCH2, S0RC /READ COMMAND REGISTER
4447 7886 RTL /MOVE DIRECTION BIT INTO THE LINK
4448 6777 IOT150, S0RD /GET BLOCK NUMBER FOUND
4449 7841 CIA
4450 1278 TAD SLOOK /COMBINE WITH BLOCK LOOKED FOR
4451 7498 SNA /CURRENT BLOCK?
4452 5263 JMP LOGGED /YES, CHECK DIRECTION
4453 7841 CIA /NO, TAKE 2'S COMPLEMENT
4454 7428 SNA /LINK IS 1 IF BACKWARD AND NOT AT OR LOWER THAN BLOCK
4455 1371 TAD (2 /ADD TWO TO ENABLE TURN AROUND
4456 7438 SEL CLA /TURN AROUND (3 BEYOND)?
4457 5215 JMP SRCH1 /NO, DON'T TURN AROUND
4458 6776 IOT158, S0RC /READ THE COMMAND REGISTER
4459 7886 RTL /MOVE THE DIRECTION BIT INTO LINK
4460 7832 CHL RTR /COMPLEMENT THE DIRECTION BIT
4461 2895 ISE BLNTRY /8 TRIES ?
4462 5286 JMP SRCHB /NO, GO SEARCH AGAIN
4463 7288 BADBLK, CLA
4464 1278 TAD SLOOK
4465 7482 WLT /AC=THE BLOCK BEING SEARCHED FOR BUT FAILED
4466 5263 JMP =1 /TO FIND AFTER 8 TRIES
4467 7628 LOGGED, SNA CLA /FOUND BLOCK FORWARD?
4468 5215 JMP SRCH1 /NO
4469 5688 JMP I SEARCH /YES, EXIT
4470 8888 SLOOK, 8 /BLOCK NUMBER LOOKED FOR
4471 8888 SUNIT, 8 /CURRENT UNIT

/WRITE SUBROUTINE
/CALLING SEQUENCE)
/ TAO (BLKNO /FIRST BLOCK TO BE WRITTEN INTO
/ JMS WRITE /CALL SUBROUTINE
/ ADDRESS /ADDRESS OF FIRST DATA WORD
/ =N /MINUS (2'S) NUMBER OF WORDS TO TRANSFER
/ /RETURN HERE
/128 WORDS PER BLOCK WILL BE WRITTEN FROM MEMORY

4472 8888 WRITE, 8
4473 3278 DCA WCNT /SAVE BLOCK NUMBER
4474 1672 TAO I WRITE
4475 3344 DCA WADDR /SAVE ADDRESS
4476 2272 ISE WRITE
4477 1672 TAO I WRITE
4478 3344 DCA WCOUNT /SAVE WORD COUNT
4479 2272 ISE WRITE
4480 4778' JMS CSUMRT
4481 8825 25

```

4984 7177      DUFF1=1
4985 7888      -288
4986 4787'     JMS      SBCXOR
4987 4788'     JMS      WRTLCK /CHECK FOR WRITE LOCKOUT
4918 1278      TAO      WCNT
4911 4888      JMS      SEARCH /FIND BLOCK
4912 4789'     WRITE1, JMS  REVGRD /WAIT FOR REVERSE GUARD
4913 4779'     JMS      RDQUAD /DELAY TWO-THIRDS THRU LOCK
4914 6776      IOT152, SDRG
4915 1364      TAO      (488
4916 6774      IOT153, SDCG /LOAD CONTROL WITH UNIT, PWD, CO, WRITE
4917 1363      TAO      (25
4928 4782'     JMS      WRQUAD /WRITE REVERSE CHECKSUM
4921 1744      WRITE2, TAO I WADDR /GET THE DATA WORD
4922 2344      ISB      WADDR /INCREMENT ADDRESS
4923 7888      NOP      /SAFETY NOP
4924 4782'     JMS      WRQUAD /WRITE DATA WORD ON TAPE
4925 2345      ISB      WCOUNT /WORD 1289
4926 5381      JMP      /NO
4927 4782'     JMS      WRQUAD /YES WRITE A B (WORD 129)
4938 1761'     TAO      CHKSUM
4931 7848      CMA
4932 8374      AND      (77
4933 7186      RTL CLL
4934 7886      RTL
4935 7886      RTL
4936 4782'     JMS      WRQUAD /WRITE CHECKSUM
4937 4768'     JMS      CLRNT /WAIT FOR CHECKSUM TO BE WRITTEN,CLEAR "WRITE"
4948 6776      IOT155, SDRG
4941 8387      AND      (4888
4942 6774      IOT156, SDCG /STOP TAPE
4943 5672      JMP I WRITE
4944 8888      WCNT=BLOCK /BLOCK NUMBER, ALSO BLOCK DATA COUNTER
4945 8888      WADDR, B /WORD ADDRESS
4945 8888      WCOUNT, B /WORD COUNT

4957 8888
4958 4271
4959 4744
4962 4781
4963 8825
4964 8488
4965 4661
4966 3731
4967 4714
4978 4383
4971 8882
4972 8884
4973 7792
4974 8877
4975 4787
4976 8188
4977 3888
4688
    
```

PAGE

```

/READ SUBROUTINE
/CALLING SEQUENCE:
/ TAO (BLKNO /FIRST BLOCK TO BE READ FROM
/ JMS READ /CALL SUBROUTINE
/ ADDRESS /ADDRESS FOR FIRST DATA WORD
/ -N /MINUS (219) NUMBER OF WORDS TO TRANSFER
/ /RETURN HERE
/128 WORDS PER BLOCK WILL BE READ INTO MEMORY

4688 8888      READ, B
4681 3296      DCA      RCNT /SAVE BLOCK NUMBER
4682 1688      TAO I READ
4683 3297      DCA      RADDR /SAVE ADDRESS
4684 2288      ISB      READ
4685 1688      TAO I READ
4686 3288      DCA      RCOUNT /SAVE WORD COUNT
4687 2288      ISB      READ
4618 1896      TAO      RCNT
4611 4777'     JMS      SEARCH /FIND BLOCK
4612 6771      READ1, SDBS /WAIT FOR REVERSE GUARD
4613 5212      JMP      /NO
4614 6776      IOT15A, SDRG /READ THE MARK TRACK
4615 8376      AND      (77
4616 1375      TAO      (32
4617 7488      SNA /REVERSE GUARD
4628 5225      JMP      /YES, EXIT
4621 1374      TAO      (18 /NO
4622 7648      SRA      CLA /END ZONE ?
4623 5212      JMP      READ1 /NO
4624 5274      JMP      IOT162 /YES STOP TAPE
4625 4387      JMS      RDQUAD /WAIT FOR
4626 4387      JMS      RDQUAD /REVERSE CHECKSUM
4627 4387      JMS      RDQUAD /HASK
4638 8376      AND      (77
4631 7418      SKP /STORE THE WORD
4632 4387      READ2, JMS  RDQUAD /GET DATA WORD
4633 3687      DCA I RADDR
4634 2287      ISB      RADDR
4635 7888      NOP      /SAFETY NOP
4636 2288      ISB      RCOUNT /128 DATA WORDS?
4637 5292      JMP      READ2 /NO
4648 4387      JMS      RDQUAD /YES, SET WORD 129
4641 3687      DCA I RADDR /STORE IT
4642 2287      ISB      RADDR
4643 4387      JMS      RDQUAD /GET FORWARD CHECKSUM

4644 8373      AND      (7788
4645 3687      DCA I RADDR
4646 6772      IOT157, SDBS
4647 7418      SKP
4688 4961      JMS I CTYERR /TIMING ERROR
4681 6776      IOT158, SDRG
4682 8372      AND      (4888
    
```

```

4653 6774 IOT159, SOLC /STOP TAPE
4654 4771 JMS CHKCHK /CALCULATE AND CHECK CHECK SUM
4655 5600 JMP I READ
4656 0000 RCNT, 0 /BLOCK NUMBER, ALSO BLOCK DATA COUNTER
4657 0000 RADDR, 0 /WORD ADDRESS
4658 0000 RCOUNT, 0 /WORD COUNT

/WAIT FOR REVERSE GUARD SUBROUTINE
4661 0000 REVGRD, 0
4662 6771 IOT160, SOS5 /WAIT FOR MARK TRACK CHANGE
4663 5242 JMP I=I
4664 6776 IOT161, SDRC /READ MARK TRACK
4665 0376 AND (77
4666 1375 TAD (=32
4667 7499 SNA /REVERSE GUARD?
4668 5661 JMP I REVGRD /YES, EXIT?
4671 1374 TAD (10 /NO
4672 7648 SZA CLA /END EDNE?
4673 5242 JMP REVGRD+1 /NO
4674 6776 IOT162, SDRC /YES, STOP TAPE
4675 0372 AND (4000
4676 6774 IOT163, SOLC
4677 7402 WLI
4678 5277 JMP I=I /NON-RECOVERABLE ERROR, PROGRAM
/FOUND ENDZONE WHILE LOOKING FOR REV GRD
/BLOCK PROBABLY ABOVE 2771

/WRITE A "QUAD WORD" (12 BIT WORD) SUBROUTINE
4781 0000 WRQUAD, 0
4782 6773 IOT164, SDSQ /WAIT FOR NEXT QUAD FLAG
4783 5302 JMP I=I
4784 6775 IOT165, SOLD /LOAD DATA BUFFER TO WRITE ON TAPE
4785 7600 M0200A, CLA+400 /CLEAR AC
4786 5701 JMP I WRQUAD /EXIT

/READ A "QUAD WORD" (12 BIT WORD) SUBROUTINE
4787 0000 RDQUAD, 0
4788 6773 IOT166, SDSQ /WAIT FOR QUAD FLAG
4789 5310 JMP I=I
4790 6777 IOT167, SDRD /READ DATA BUFFER, CLEAR FLAG
4791 5707 JMP I RDQUAD

/SIXBIT COMPLEMENT XOR SUBROUTINE
/SUBROUTINE IS ENTERED WITH DATA WORD TO BE XORED IN AC
/TWO 6-BIT COMPLEMENT XORS WILL TAKE PLACE TO LOCATION CHKSUM
/WITH THE RESULT IN CHKSUM

4714 0000 SBCKOR, 0
4715 7040 DCA /COMPLEMENT WORD
4716 3349 DCA SBWORD /AND SAVE
4717 1345 TAD SBWORD
4718 0344 AND CHKSUM
4719 7041 CIA
4720 7104 CLL RAL
4721 1345 TAD SBWORD
4722 1344 TAD CHKSUM

```

```

4725 3344 DCA CHKSUM
4726 1345 TAD SBWORD
4727 7112 RTR CLL, RTR, RTR

4730 7012 DCA SBWORD
4731 7012 TAD SBWORD
4732 3349 AND CHKSUM
4733 1345 CIA
4734 0344 CLL RAL
4735 7041 TAD SBWORD
4736 7104 TAD CHKSUM
4737 1345 AND (77
4738 1344 DCA CHKSUM
4739 0376 DCA CHKSUM
4740 3344 JMP I SBCKOR
4741 0714 CHKSUM, 0
4742 0000 SBWORD,
4743 4745

4771 4323
4772 4000
4773 7700
4774 0010
4775 7746
4776 0077
4777 4400
5000

PAGE

/MESSAGES
MESS1, TEXT "LOAD AND READ DATA REGISTER ERROR"

5000 1417
5001 0104
5002 4001
5003 1604
5004 4022
5005 0501
5006 0440
5007 0401
5010 2401
5011 4022
5012 0507
5013 1123
5014 2405
5015 2240
5016 0522
5017 2217
5020 2200
5021 1417 MESS2, TEXT "LOAD AND READ COMMAND REGISTER ERROR"
5022 0104
5023 4001
5024 1604
5025 4022
5026 0501
5027 0440

```

9038 0317
 9031 1515
 9032 0116
 9033 0440
 9034 2205
 9035 0711
 9036 2324
 9037 0522
 9040 4005
 9041 2222
 9042 1722
 9043 0000
 9044 1116
 9045 1124
 9046 1101
 9047 1411
 9050 3205
 9051 4024
 9052 0523
 9053 2400
 9054 0301
 9055 0640
 9056 0411
 9057 0440
 9060 1017
 9061 2440
 9062 0314
 9063 0501
 9064 2240
 9065 0317
 9066 1515
 9067 0116
 9070 0440
 9071 2205
 9072 0711
 9073 2324
 9074 0522
 9075 0000
 9076 0310
 9077 0503
 9100 1340
 9101 2304
 9102 1403
 9103 3440
 9104 2304
 9105 1404
 9106 3440
 9107 2304
 9110 2203
 9111 3440
 9112 2304
 9113 2204
 9114 4001
 9115 1004
 9116 4001

MESS3, TEXT "INITIALIZE TEST"

MESS4, TEXT "CAF DID NOT CLEAR COMMAND REGISTER"

MESS5, TEXT "CHECK SOLC, SLD, SRC, SDRD AND AC CLEAR"

9117 0340
 9120 0314
 9121 0501
 9122 2200
 9123 2304
 9124 1403
 9125 4004
 9126 1104
 9127 4016
 9130 1724
 9131 4003
 9132 1405
 9133 0122
 9134 4001
 9135 0300
 9136 2304
 9137 2203
 9140 4004
 9141 1104
 9142 4016
 9143 1724
 9144 4003
 9149 1405
 9146 0122
 9147 4001
 9150 0300
 9151 2304
 9152 1404
 9153 4003
 9154 1405
 9155 0122
 9156 0504
 9157 4001
 9160 0300
 9161 2304
 9162 2204
 9163 4004
 9164 1104
 9165 4016
 9166 1724
 9167 4003
 9170 1405
 9171 0122
 9172 4001
 9173 0300
 9174 2311
 9175 1007
 9176 1405
 9177 4014
 9200 1116
 9201 0540
 9202 0614
 9203 0107
 9204 4023
 9205 1311

MESS6, TEXT "SOLC DID NOT CLEAR AC"

MESS7, TEXT "SRC DID NOT CLEAR AC"

MESS8, TEXT "SOLD CLEARED AC"

MESS9, TEXT "SDRD DID NOT CLEAR AC"

MESS10, TEXT "SINGLE LINE FLAG SKIP INSTRUCTION AND LOGIC"

9206 2040
 9207 1116
 9210 2324
 9211 2225
 9212 0324
 9213 1117
 9214 1040
 9215 0116
 9216 0440
 9217 1417
 9220 0711
 9221 0300
 9222 2311
 9223 1007
 9224 1405
 9225 4014
 9226 1116
 9227 0540
 9230 0014
 9231 0107
 9232 4001
 9233 1427
 9234 0131
 9235 2340
 9236 2305
 9237 2440
 9240 1722
 9241 4003
 9242 0423
 9243 2340
 9244 0114
 9245 2701
 9246 3123
 9247 4023
 9250 1311
 9251 2023
 9252 0000
 9253 2311
 9254 1007
 9255 1405
 9256 4014
 9257 1116
 9260 0540
 9261 0614
 9262 0107
 9263 4004
 9264 1705
 9265 2340
 9266 1017
 9267 2440
 9270 2305
 9271 2440
 9272 1722
 9273 4023
 9274 0423

MESS11, TEXT "SINGLE LINE FLAG ALWAYS SET OR S055 ALWAYS SKIPS"

MESS12, TEXT "SINGLE LINE FLAG DOES NOT SET OR S055 DOES NOT SKIP"

9275 2340
 9276 0417
 9277 0923
 9300 4016
 9301 1724
 9302 4023
 9303 1311
 9304 2000
 9305 2311
 9306 1007
 9307 1405
 9310 4014
 9311 1116
 9312 0540
 9313 0014
 9314 0107
 9315 4003
 9316 1405
 9317 0122
 9320 0504
 9321 4002
 9322 3140
 9323 2304
 9324 2323
 9325 0000
 9326 2311
 9327 1007
 9330 1405
 9331 4014
 9332 1116
 9333 0540
 9334 0014
 9335 0107
 9336 4016
 9337 1724
 9340 4003
 9341 1405
 9342 0122
 9343 0504
 9344 4002
 9345 3140
 9346 0301
 9347 0600
 9350 2311
 9351 1007
 9352 1405
 9353 4014
 9354 1116
 9355 0540
 9356 0014
 9357 0107
 9360 4016
 9361 1724
 9362 4003
 9363 1405

MESS13, TEXT "SINGLE LINE FLAG CLEARED BY S059"

MESS14, TEXT "SINGLE LINE FLAG NOT CLEARED BY CAP"

MESS15, TEXT "SINGLE LINE FLAG NOT CLEARED BY S0L0N"

5364 0122
 5365 0504
 5366 4002
 5367 3140
 5370 2304
 5371 1404
 5372 0000
 5373 2311
 5374 1407
 5375 1409
 5376 4014
 5377 1110
 5400 0940
 5401 0014
 5402 0107
 5403 4016
 5404 1724
 5405 4003
 5406 1409
 5407 0122
 5410 0504
 5411 4002
 5412 3140
 5413 2304
 5414 2203
 5415 0000
 5416 2311
 5417 1407
 5420 1409
 5421 4014
 5422 1110
 5423 0940
 5424 0014
 5425 0107
 5426 4016
 5427 1724
 5430 4003
 5431 1409
 5432 0122
 5433 0504
 5434 4002
 5435 3140
 5436 2304
 5437 2204
 5440 0000
 5441 2311
 5442 1407
 5443 1409
 5444 4014
 5445 1110
 5446 0940
 5447 0014
 5450 0107
 5451 4003
 5452 1409

MESS16, TEXT "SINGLE LINE FLAG NOT CLEARED BY SDRD"

MESS17, TEXT "SINGLE LINE FLAG NOT CLEARED BY SDRD"

MESS18, TEXT "SINGLE LINE FLAG CLEARED BY SOST, SDBS, OR S DLC"

5453 0122
 5454 0504
 5455 4002
 5456 3140
 5457 2304
 5460 2304
 5461 5440
 5462 2304
 5463 2301
 5464 5440
 5465 1722
 5466 4003
 5467 0014
 5470 0300
 5471 2109
 5472 0104
 5473 4014
 5474 1110
 5475 0940
 5476 0014
 5477 0107
 5500 4003
 5501 1311
 5502 2040
 5503 1110
 5504 2304
 5505 2225
 5506 0304
 5507 1117
 5510 1040
 5511 0110
 5512 0440
 5513 1417
 5514 0711
 5515 0300
 5516 2109
 5517 0104
 5520 4014
 5521 1110
 5522 0940
 5523 0014
 5524 0107
 5525 4001
 5526 1407
 5527 0131
 5530 2340
 5531 2309
 5532 2440
 5533 1722
 5534 4003
 5535 4003
 5536 0140
 5537 0114
 5540 2701
 5541 3103

MESS19, TEXT "QUAD LINE FLAG SKIP INSTRUCTION AND LOGIC"

MESS20, TEXT "QUAD LINE FLAG ALWAYS SET OR SDBS ALWAYS SKIPS"

5542	4023		
5543	1311		
5544	2023		
5545	0000		
5546	2125	MESS21, TEXT	"QUAD LINE FLAG SET TOO SOON"
5547	0104		
5550	4014		
5551	1116		
5552	0940		
5553	0014		
5554	0107		
5555	4023		
5556	0524		
5557	4024		
5560	1717		
5561	4023		
5562	1717		
5563	1400		
5564	2125	MESS22, TEXT	"QUAD LINE FLAG DOES NOT SET OR S030 DOES NOT SKIP"
5565	0104		
5566	4014		
5567	1116		
5570	0940		
5571	0014		
5572	0107		
5573	4004		
5574	1705		
5575	2340		
5576	1617		
5577	2440		
5600	2305		
5601	2440		
5602	1722		
5603	4023		
5604	0423		
5605	2140		
5606	0417		
5607	0523		
5610	4016		
5611	1724		
5612	4023		
5613	1311		
5614	2000		
5615	2125	MESS23, TEXT	"QUAD LINE FLAG CLEARED BY S050"
5616	0104		
5617	4014		
5620	1116		
5621	0940		
5622	0014		
5623	0107		
5624	4003		
5625	1405		
5626	0122		
5627	0504		
5630	4002		

5631	3140		
5632	2304		
5633	2321		
5634	0000		
5635	2125	MESS24, TEXT	"QUAD LINE FLAG NOT CLEARED BY CAP"
5636	0104		
5637	4014		
5640	1116		
5641	0940		
5642	0014		
5643	0107		
5644	4016		
5645	1724		
5646	4003		
5647	1405		
5650	0122		
5651	0504		
5652	4002		
5653	3140		
5654	0301		
5655	0400		
5656	2125	MESS25, TEXT	"QUAD LINE FLAG NOT CLEARED BY S0LD"
5657	0104		
5660	4014		
5661	1116		
5662	0940		
5663	0014		
5664	0107		
5665	4016		
5666	1724		
5667	4003		
5670	1405		
5671	0122		
5672	0504		
5673	4002		
5674	3140		
5675	2304		
5676	1404		
5677	0000		
5700	2125	MESS26, TEXT	"QUAD LINE FLAG NOT CLEARED BY SDRD"
5701	0104		
5702	4014		
5703	1116		
5704	0940		
5705	0014		
5706	0107		
5707	4016		
5710	1724		
5711	4003		
5712	1405		
5713	0122		
5714	0504		
5715	4002		
5716	3140		
5717	2304		

5720 2203
 5721 0000
 5722 2125
 5723 0104
 5724 4014
 5725 1116
 5726 0940
 5727 0014
 5730 0107
 5731 4016
 5732 1724
 5733 4003
 5734 1405
 5735 0122
 5736 0904
 5737 4002
 5740 3140
 5741 2304
 5742 2204
 5743 0000
 5744 2125
 5745 0104
 5746 4014
 5747 1116
 5750 0940
 5751 0014
 5752 0107
 5753 4003
 5754 1405
 5755 0122
 5756 0904
 5757 4002
 5760 3140
 5761 2304
 5762 2304
 5763 5440
 5764 2304
 5765 2323
 5766 5440
 5767 1722
 5770 4023
 5771 0414
 5772 0300
 5773 2125
 5774 0104
 5775 4014
 5776 1116
 5777 0940
 0000 0014
 0001 0107
 0002 4003
 0003 1725
 0004 1024
 0005 0922
 0006 4006

MESS27, TEXT "QUAD LINE FLAG NOT CLEARED BY SORD"

MESS28, TEXT "QUAD LINE FLAG CLEARED BY SOST, S0SS, OR WOLC"

MESS29, TEXT "QUAD LINE FLAG COUNTER FLIP/FLOP NOT PROPERLY CLEARED"

0007 1411
 0010 2057
 0011 0014
 0012 1720
 0013 4016
 0014 1724
 0015 4020
 0016 2217
 0017 2005
 0020 2214
 0021 3140
 0022 0314
 0023 0901
 0024 2205
 0025 0400

MESS30, TEXT "TIMING ERROR SKIP INSTRUCTION AND LOGIC"

0026 2411
 0027 1911
 0030 1407
 0031 4005
 0032 2222
 0033 1722
 0034 4023
 0035 1311
 0036 2040
 0037 1116
 0040 2324
 0041 2225
 0042 0324
 0043 1117
 0044 1040
 0045 0116
 0046 0440
 0047 1417
 0050 0711
 0051 0300
 0052 2411
 0053 1911
 0054 1407
 0055 4005
 0056 2222
 0057 1722
 0060 4001
 0061 1427
 0062 0131
 0063 2340
 0064 2305
 0065 2440
 0066 1722
 0067 4023
 0070 0423
 0071 2440
 0072 0114

MESS31, TEXT "TIMING ERROR ALWAYS SET OR SOST ALWAYS SKIP"

6073	2781		
6074	3123		
6075	4023		
6076	1381		
6077	2023		
6100	0000		
6101	2411	MESS32, TEXT	"TIMING ERROR DOES NOT SET IN READ MODE OR SOST DOES NOT SKIP"
6102	1911		
6103	1607		
6104	4009		
6105	2222		
6106	1722		
6107	4004		
6110	1709		
6111	2340		
6112	1617		
6113	2440		
6114	2305		
6115	2440		
6116	1116		
6117	4022		
6120	0901		
6121	0440		
6122	1917		
6123	0409		
6124	4017		
6125	2440		
6126	2304		
6127	2304		
6130	4004		
6131	1709		
6132	2340		
6133	1617		
6134	2440		
6139	2313		
6136	1120		
6137	0000		
6140	2411	MESS33, TEXT	"TIMING ERROR CLEARED BY SOST"
6141	1911		
6142	1607		
6143	4009		
6144	2222		
6145	1722		
6146	4003		
6147	1409		
6150	0122		
6151	0904		
6152	4002		
6153	3140		
6154	2304		
6155	2304		
6156	0000		
6157	2411	MESS34, TEXT	"TIMING ERROR NOT CLEARED BY CAF"
6160	1911		
6161	1607		

6162	4009		
6163	2222		
6164	1722		
6165	4016		
6166	1724		
6167	4003		
6170	1409		
6171	0122		
6172	0904		
6173	4002		
6174	3140		
6175	0301		
6176	0600		
6177	2411	MESS35, TEXT	"TIMING ERROR STATUS BIT NOT SET IN COMMAND REGISTER"
6200	1911		
6201	1607		
6202	4009		
6203	2222		
6204	1722		
6205	4003		
6206	2401		
6207	2405		
6210	2340		
6211	0211		
6212	2440		
6213	1617		
6214	2440		
6215	2305		
6216	2440		
6217	1116		
6220	4003		
6221	1715		
6222	1901		
6223	1604		
6224	4022		
6225	0907		
6226	1123		
6227	2405		
6230	2200		
6231	2411	MESS36, TEXT	"TIMING ERROR NOT CLEARED BY SDC"
6232	1911		
6233	1607		
6234	4009		
6235	2222		
6236	1722		
6237	4016		
6240	1724		
6241	4003		
6242	1409		
6243	0122		
6244	0904		
6245	4002		
6246	3140		
6247	2304		
6250	1403		

6251 0000
 6252 2411
 6253 1511
 6254 1607
 6255 4000
 6256 2222
 6257 1722
 6200 4004
 6261 1700
 6262 2340
 6263 1617
 6264 2440
 6265 2300
 6266 2440
 6267 1110
 6270 4027
 6271 2211
 6272 2400
 6273 4010
 6274 1704
 6275 0000
 6276 2411
 6277 1911
 6300 1607
 6301 4000
 6302 2222
 6303 1722
 6304 4004
 6305 1700
 6306 2340
 6307 1617
 6310 2440
 6311 0314
 6312 0001
 6313 2240
 6314 2722
 6315 1124
 6316 0040
 6317 0614
 6320 1120
 6321 0706
 6322 1417
 6323 2000
 6324 2411
 6325 1911
 6326 1607
 6327 4000
 6330 2202
 6331 1722
 6332 4016
 6333 1724
 6334 4000
 6335 0004
 6336 4002
 6337 3140

MESS37, TEXT "TIMING ERROR DOES NOT SET IN WRITE MODE"

MESS38, TEXT "TIMING ERROR DOES NOT CLEAR WRITE FLIP/FLOP"

MESS39, TEXT "TIMING ERROR NOT SET BY \$DR0 \$DRD, OR \$DLC"

6340 2304
 6341 2200
 6342 4000
 6343 0422
 6344 0404
 6345 4017
 6346 2240
 6347 2304
 6350 1400
 6351 0000
 6352 2000
 6353 4024
 6354 1740
 6355 2000
 6356 0000
 6357 0440
 6360 0311
 6361 2200
 6362 2011
 6363 2422
 6364 3140
 6365 0310
 6366 0000
 6367 1340
 6370 2000
 6371 1114
 6372 0740
 6373 4700
 6374 1400
 6375 0122
 6376 4010
 6377 0122
 6400 1340
 6401 2422
 6402 0100
 6403 1340
 6404 2000
 6405 0711
 6406 2324
 6407 0022
 6410 4740
 6411 0400
 6412 0124
 6413 2022
 6414 0000
 6415 1901
 6416 2210
 6417 4024
 6420 2201
 6421 0310
 6422 4022
 6423 0007
 6424 1100
 6425 2400
 6426 2240

MESS43, TEXT "UP TO SPEED CIRCUITRY CHECK USING 'CLEAR MARK TRACK REGISTER' FEATURE"

MESS44, TEXT "MARK TRACK REGISTER NOT CLEARED BY 'GO'"

6427 1617
 6430 2440
 6431 0314
 6432 0501
 6433 2205
 6434 0440
 6435 0231
 6436 4047
 6437 0717
 6440 4700
 6441 1501
 6442 2213
 6443 4024
 6444 2201
 6445 0313
 6446 4022
 6447 0507
 6450 1123
 6451 2405
 6452 2240
 6453 1617
 6454 2440
 6455 0314
 6456 0501
 6457 2205
 6460 0440
 6461 0231
 6462 4047
 6463 2324
 6464 1700
 6465 4700
 6466 1501
 6467 2213
 6470 4024
 6471 2201
 6472 0313
 6473 4022
 6474 0507
 6475 1123
 6476 2405
 6477 2240
 6500 1617
 6501 2440
 6502 0314
 6503 0501
 6504 2205
 6505 0440
 6506 0231
 6507 4047
 6510 2205
 6511 2647
 6512 4024
 6513 1740
 6514 4700
 6515 2704

MESS45, TEXT "MARK TRACK REGISTER NOT CLEARED BY /STOP/"

MESS46, TEXT "MARK TRACK REGISTER NOT CLEARED BY /REV/ TO /PMD/"

6516 4700
 6517 1501
 6520 2213
 6521 4024
 6522 2201
 6523 0313
 6524 4022
 6525 0507
 6526 1123
 6527 2405
 6530 2240
 6531 1617
 6532 2440
 6533 0314
 6534 0501
 6535 2205
 6536 0440
 6537 0231
 6540 4047
 6541 0627
 6542 0447
 6543 4024
 6544 1740
 6545 4722
 6546 0526
 6547 4700
 6550 1501
 6551 2213
 6552 4024
 6553 2201
 6554 0313
 6555 4022
 6556 0507
 6557 1123
 6560 2405
 6561 2240
 6562 1617
 6563 2440
 6564 0314
 6565 0501
 6566 2205
 6567 0440
 6570 0231
 6571 4047
 6572 2516
 6573 1124
 6574 4060
 6575 4740
 6576 2417
 6577 4047
 6600 2916
 6601 1124
 6602 4061
 6603 4700
 6604 1501

MESS47, TEXT "MARK TRACK REGISTER NOT CLEARED BY /PMD/ TO /REV/"

MESS48, TEXT "MARK TRACK REGISTER NOT CLEARED BY /UNIT 0/ TO /UNIT 1/"

MESS49, TEXT "MARK TRACK REGISTER NOT CLEARED BY /UNIT 1/ TO /UNIT 0/"

6605 2213
 6606 4824
 6607 2201
 6610 0313
 6611 4822
 6612 0507
 6613 1123
 6614 2405
 6615 2248
 6616 1617
 6617 2448
 6620 0314
 6621 0501
 6622 2205
 6623 0448
 6624 0231
 6625 4847
 6626 2514
 6627 1124
 6630 4801
 6631 4748
 6632 2417
 6633 4847
 6634 2516
 6635 1124
 6636 4848
 6637 4708
 6640 4723
 6641 0514
 6642 0503
 6643 2448
 6644 0522
 6645 2217
 6646 2247
 6647 4824
 6650 0523
 6651 2423
 6652 0808
 6653 2411
 6654 1511
 6655 1607
 6656 4805
 6657 2222
 6660 1722
 6661 4823
 6662 0524
 6663 0808
 6664 1617
 6665 4823
 6666 0514
 6667 0503
 6670 2448
 6671 0522
 6672 2217
 6673 2248

MESS00, TEXT "/SELECT ERROR' TESTS"

MESS01, TEXT "TIMING ERROR SET"

MESS02, TEXT "NO SELECT ERROR STATUS FROM UNIT E"

6674 2324
 6675 0124
 6676 2523
 6677 4806
 6700 2217
 6701 1548
 6702 2516
 6703 1124
 6704 4801
 6705 0808
 6706 4727
 6707 2211
 6710 2405
 6711 4748
 6712 1617
 6713 2448
 6714 0314
 6715 0501
 6716 2205
 6717 0448
 6720 0231
 6721 4823
 6722 0514
 6723 0503
 6724 2448
 6725 0522
 6726 2217
 6727 2208
 6730 4727
 6731 2211
 6732 2405
 6733 4814
 6734 1703
 6735 1348
 6736 1725
 6737 2447
 6740 4824
 6741 0523
 6742 2423
 6743 0808
 6744 1617
 6745 4827
 6746 2211
 6747 2405
 6750 4814
 6751 1703
 6752 1348
 6753 1725
 6754 2448
 6755 2324
 6756 0124
 6757 2523
 6760 4806
 6761 2217
 6762 1548

MESS03, TEXT "WRITE' NOT CLEARED BY SELECT ERROR"

MESS04, TEXT "WRITE LOCK OUT' TESTS"

MESS05, TEXT "NO WRITE LOCK OUT STATUS FROM UNIT E"

6763	2916		
6764	1124		
6765	4868		
6766	0888		
6767	4727	MESS56, TEXT	"WRITE' NOT CLEARED BY WRITE LOCK OUT"
6770	2211		
6771	2485		
6772	4748		
6773	1617		
6774	2448		
6775	0314		
6776	0581		
6777	2285		
7800	0448		
7801	0231		
7802	4827		
7803	2211		
7804	2485		
7805	4814		
7806	1783		
7807	1348		
7810	1723		
7811	2488		
7812	0481	MESS57, TEXT	"DATA WRITTEN FORWARD"
7813	2481		
7814	4827		
7815	2211		
7816	2424		
7817	0516		
7820	4886		
7821	1722		
7822	2781		
7823	2284		
7824	0888		
7825	2722	MESS58, TEXT	"WRITE DATA FORWARD"
7826	1124		
7827	0548		
7830	8481		
7831	2481		
7832	4886		
7833	1722		
7834	2781		
7835	2284		
7836	0888		
7837	2285	MESS59, TEXT	"READ DATA FORWARD"
7840	0184		
7841	4884		
7842	0124		
7843	0148		
7844	0617		
7845	2227		
7846	0122		
7847	0488		
7850	2285	MESS60, TEXT	"READ DATA BACKWARD"
7851	0184		

/TDBE DIAGNOSTIC

PAL18

V141

19-OCT-72

11189

PAGE 1065

7852	4884		
7853	0124		
7854	0148		
7855	0281		
7856	0313		
7857	2781		
7860	2284		
7861	0888		
7862	0481	MESS61, TEXT	"DATA WRITTEN BACKWARD"
7863	2481		
7864	4827		
7865	2211		
7866	2424		
7867	0516		
7870	4882		
7871	0183		
7872	1327		
7873	0122		
7874	0488		
7875	2722	MESS62, TEXT	"WRITE DATA BACKWARD"
7876	1124		
7877	0548		
7100	0481		
7101	2481		
7102	4882		
7103	0183		
7104	1327		
7105	0122		
7106	0488		
7107	1617	MESS63, TEXT	"NO UNIT 0 SELECTED"
7110	4823		
7111	1611		
7112	2448		
7113	0848		
7114	2385		
7115	1485		
7116	0324		
7117	0584		
7120	0888		

7289

PAGE

/ROUTINE TO CHANGE ALL TDBE IOYS IN PROGRAM FOR MULTIPLE UNIT
 /ROUTINE IS STARTED AT LOCATION "MODIFY" WITH AC6,7 AND 8 INDICATING
 /DEVICE SELECTOR BITS 6, 7, AND 8 OF THE CONTROL TO BE TESTED (4,5,6, OR 7)

7280	7604	MODIFY, LAS	
7281	9221	AND	MODMS1
7282	3223	DCA	MODCON
7283	1238	TAD	MODTAB
7284	3224	DCA	MODPT1
7285	1227	TAD	MODS12
7286	3226	DCA	MODCNT
7287	1624	MODLUP, TAD I	MODPT1

7218	3225	DCA	MODPT2
7211	1625	TAD I	MODPT2
7212	8222	AND	MODMS2
7213	1223	TAD	MODCON
7214	3623	OCA I	MODPT2
7215	2224	ISE	MODPT1
7216	2226	ISE	MODCNT
7217	5287	JMP	MODLUP
7228	7482	HLT	
7221	8838	MODMS1,	38
7222	7747	MODMS2,	7747
7223	8888	MODCON,	B
7224	8888	MODPT1,	B
7225	8888	MODPT2,	B
7226	8888	MODCNT,	B
7227	7461	MODSIE,	=MODEND+MODTAB
7238	7231	MODTAB,	+1
7231	8287		10T1
7232	8212		10T2
7233	8214		10T3
7234	8244		10T4
7235	8246		10T5
7236	8313		10T6
7237	8328		10T7
7248	8486		10T8
7241	8423		10T9
7242	8425		10T10
7243	8442		10T11
7244	8437		10T12
7245	8461		10T13
7246	8686		10T14
7247	8612		10T15
7258	8621		10T16
7251	8636		SING1
7252	8645		SING2
7253	8633		10T17
7254	8662		10T18
7255	8665		SING3
7256	8678		10T19
7257	8674		10T20
7268	8784		SING4
7261	8787		10T21
7262	8713		10T22
7263	8723		SING5
7264	8726		10T23
7265	8732		10T24
7266	1882		SING6
7267	1884		10T25
7278	1886		10T26
7271	1811		10T27
7272	1812		10T28
7273	1832		10T29
7274	1836		10T30

7275	1841		QUAD8
7276	1842		10T31
7277	1844		10T32
7388	1888		10T33
7381	1853		10T34
7382	1857		10T35
7383	1861		10T36
7384	1867		10T37
7385	1872		10T38
7386	1181		QUAD1
7387	1118		QUAD2
7318	1116		10T39
7311	1125		10T40
7312	1138		QUAD3
7313	1133		10T41
7314	1137		10T42
7315	1282		QUAD4
7316	1285		10T43
7317	1211		10T44
7328	1221		QUAD5
7321	1224		10T45
7322	1238		10T46
7323	1248		QUAD6
7324	1242		10T47
7325	1244		10T48
7326	1258		10T49
7327	1254		10T50
7338	1278		10T51
7331	1271		10T52
7332	1273		10T53
7333	1276		10T54
7334	1385		10T55
7335	1324		10T56
7336	1325		10T57
7337	1327		10T58
7348	1334		TIME8
7341	1342		10T59
7342	1393		TIME1
7343	1482		TIME2
7344	1418		10T60
7345	1417		10T61
7346	1422		TIME3
7347	1424		10T62
7358	1442		10T63
7351	1445		TIME4
7352	1447		10T64
7353	1453		10T65
7354	1468		10T66
7355	1466		10T67
7356	1478		10T68
7357	1475		10T69
7368	1588		10T70
7361	1584		10T71

7362	1918	10T72
7363	1923	10T73
7364	1924	10T74
7365	1931	10T75
7366	1932	10T76
7367	1933	10T77
7378	1943	10T78
7371	1685	10T82
7372	1687	10T83
7373	1613	10T84
7374	1626	10T85
7375	1627	10T86
7376	1631	10T87
7377	1635	10T88
7488	1641	10T89
7481	1654	10T98
7482	1655	10T91
7483	1657	10T92
7484	1664	10T93
7485	1678	10T94
7486	1783	10T95
7487	1784	10T96
7418	1786	10T97
7411	1713	10T98
7412	1717	10T99
7413	2884	10T108
7414	2885	10T101

7415	2887	10T102
7416	2814	10T103
7417	2828	10T104
7428	2837	10T105
7421	2848	10T106
7422	2842	10T107
7423	2847	10T108
7424	2853	10T109
7425	2865	10T118
7426	2284	10T211
7427	2286	DISLUP
7438	2211	10T112
7431	2213	10T113
7432	2227	10T114
7433	2241	10T115
7434	2253	10T116
7435	2482	10T117
7436	2418	10T118
7437	2413	10T119
7448	2518	10T128
7441	2512	10T121
7442	2528	10T122
7443	2522	10T123
7444	2685	10T124
7445	2686	10T125
7446	2614	10T126

7447	2616	10T127
7458	2635	10T128
7451	2641	10T129
7452	2654	SELCY2
7453	2668	10T138
7454	2781	10T131
7455	2728	10T132
7456	2724	10T133
7457	3686	RSRCH8
7468	3687	10T134

7461	3615	RSRCH1
7462	3617	10T135
7463	3628	10T136
7464	3632	10T13A
7465	3642	10T137
7466	3622	10T138
7467	3648	RSRCH2
7478	3654	RSTURN
7471	4856	10T139
7472	4112	10T148
7473	4114	10T141
7474	4281	10T142
7475	4283	10T143
7476	4246	10T144
7477	4258	10T145
7588	4486	SRCH8
7581	4487	10T146
7582	4415	SRCH1
7583	4417	10T147
7584	4428	10T148
7585	4422	10T149
7586	4432	10T14A
7587	4442	10T158
7518	4454	10T158
7511	4448	SRCH2
7512	3733	10T151
7513	3734	10T15A
7514	4514	10T152
7515	4516	10T153
7516	4274	10T154
7517	4548	10T155
7528	4542	10T156
7521	4646	10T157
7522	4651	10T158
7523	4653	10T159
7524	4642	10T168
7525	4614	10T16A
7526	4612	READ1
7527	4664	10T161
7538	4674	10T162
7531	4676	10T163
7532	4782	10T164
7533	4784	10T165

4000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4300	11111111	11111111	11111111	11111100	00000000	00000000	00011111	11111111	11111111
4400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4500	11111111	11111111	11111111	11111111	11111100	00000000	11111111	11111111	11111111
4600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4700	11111111	11111111	11111111	11111111	11111100	00000000	00000000	01111111	11111111
5000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5300	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5500	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
5700	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6100	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6300	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6500	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
6700	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
7000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
7100	11111111	11111111	10000000	00000000	00000000	00000000	00000000	00000000	00000000
7200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
7300	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
7400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
7500	11111111	11111111	11111111	11111111	11111111	00000000	00000000	00000000	00000000
7600									
7700									

AUTO	0017	DISEND	2284	107102	2007	107149	4422
BADBLK	4461	DIBLUP	2206	107103	2014	10714A	4432
BLK	0027	DIBTRK	2209	107104	2000	10715	0012
BLKCH	2100	DIMLT	0900	107105	2007	107100	4442
BLKCHK	2400	ENDE	2404	107106	2000	107151	3733
BLKCN	0000	ERR1M7	0925	107107	2042	107152	4514
BLKEND	0070	ERR2M7	0956	107108	2047	107153	4516
BLKERR	2146	ERRDR1	0496	107109	2093	107154	4274
BLKHRR	2426	ERRDR2	0907	10711	0442	107155	4540
BLKREY	0061	FBLKCT	2123	107110	2049	107156	4542
BLKSER	0101	FILDCY	3440	107111	2204	107157	4646
BLKTRY	0055	FILDDY	3442	107112	2211	107158	4691
BMESS	4105	FILDEC	3481	107113	2213	107159	4693
BUFF1	7200	FILDPT	3441	107114	2207	10715A	3736
BUFF2	7400	FILICT	3456	107115	2241	10715B	4494
CAP	0007	FILIDT	3400	107116	2293	10710	0021
CCNTR	3224	FILINC	3400	107117	2402	107100	4042
CHKCHK	4323	FILIPY	3447	107118	2410	107101	4044
CHKCLA	0400	FILLS	3227	107119	2413	107102	4074
CHKDAT	4131	FILLPC	3241	10712	0497	107103	4076
CHKERR	4110	FILLSP	3242	107120	2540	107104	4702
CHKHLY	4120	FILL1	3243	107121	2542	107105	4704
CHKHES	4132	FILL1C	3206	107122	2500	107106	4710
CHKSUM	4744	FILL1P	3207	107123	2522	107107	4712
CLOOP	0266	FILL25	3200	107124	2005	107100	4272
CLRWI	4271	FILL2C	3273	107125	2006	107109	4277
CNTR1	0022	FILL2K	3274	107126	2014	10710A	4014
CNTR2	0023	FILL2P	3275	107127	2016	10717	0093
COMLUP	3212	FILLC1	3303	107120	2035	107170	4301
COMP	3916	FILLC2	3304	107129	2041	107171	2110
COMPAR	3200	FILLP1	3321	10713	0461	107172	0044
COMREG	0236	FILLP2	3322	107130	2040	107173	0071
CRLF	0040	FILPAT	3296	107131	2701	107174	0073
CS DLC	0405	FILPL1	3303	107132	2720	107175	0102
CS DLD	0441	FILPL2	3307	107133	2724	107176	0104
CS DRC	0422	FILPNT	0030	107134	3007	107177	0106
CS DRD	0456	FILTC	3331	107135	3017	10710	0042
CSUHRT	4303	FILTP	3325	107136	3020	10710	0070
DATA	2452	FORNT1	0532	107137	3042	1072	0212
DATERR	0000	FWDEXP	2422	107130	3022	10720	0074
DATND	0040	GOOD	0024	107139	4096	10721	0007
DATHLT	0036	GPNT*	3295	10713A	3032	10722	0713
DAYLUP	0232	GUARD	2472	10714	0000	10723	0720
DAYNES	0042	HEAD1	0025	107140	4112	10724	0732
DATNUM	0020	HEAD2	0026	107141	4114	10725	1004
DATPNT	0041	HEADTP	0000	107142	4201	10720	1000
DATREG	0201	IN	0021	107143	4203	10727	1011
DBLOCK	0200	INITSY	0305	107144	4246	10720	1012
DIBL	0296	1071	0207	107145	4250	10720	1032
DIBLK	2293	10710	0425	107146	4407	1073	0214
DISDA	0057	107100	2004	107147	4417	10730	1030
DISDAT	2236	107101	2005	107140	4420	10731	1042

10T32	1844	10T82	1485	MESS34	6157	PASS	3513
10T33	1850	10T83	1487	MESS35	6177	PREFIN	2462
10T34	1853	10T84	1483	MESS36	6231	QBLUP	1874
10T35	1857	10T85	1486	MESS37	6292	QUAD	1824
10T36	1861	10T86	1487	MESS38	6276	QUAD8	1841
10T37	1867	10T87	1431	MESS39	6324	QUAD1	1101
10T38	1872	10T88	1435	MESS4	5854	QUAD2	1118
10T39	1116	10T89	1441	MESS43	6352	QUAD3	1138
10T4	2244	10T9	8423	MESS44	6415	QUAD4	1282
10T48	1125	10T98	1484	MESS45	6441	QUAD5	1221
10T41	1135	10T91	1485	MESS46	6466	QUAD6	1248
10T42	1137	10T92	1437	MESS47	6517	QUAD7	1265
10T43	1289	10T93	1464	MESS48	6558	RADDR	4957
10T44	1211	10T94	1498	MESS49	6584	RBLKCT	2132
10T45	1224	10T95	1783	MESS5	5876	RCNT	4856
10T46	1238	10T96	1784	MESS58	6648	RCOUNT	4868
10T47	1242	10T97	1786	MESS51	6693	RD1HRK	2587
10T48	1244	10T98	1783	MESS52	6664	RD6HRK	2515
10T49	1259	10T99	1717	MESS53	6786	RDDUAD	4787
10T5	2246	LOCKED	4465	MESS54	6758	READ	4888
10T58	1254	LOCK	2445	MESS55	6744	READ1	4812
10T51	1278	LOOP1	8846	MESS56	6767	READ2	4832
10T52	1271	MS288A	4789	MESS57	7812	READR	3714
10T53	1273	MS	8854	MESS58	7825	REVBLK	2476
10T54	1276	MESSAGE	2264	MESS59	7837	REVEXP	2582
10T55	1389	MESS1	5888	MESS6	5183	REVGRD	4661
10T56	1324	MESS18	5174	MESS68	7898	RLOCDD	3669
10T57	1325	MESS11	9282	MESS61	7862	ROCK	2237
10T58	1327	MESS12	9283	MESS62	7875	RSRCH	3888
10T59	1342	MESS13	9385	MESS63	7187	RSL0CK	3478
10T6	8313	MESS14	9386	MESS7	5136	RSRCH8	3886
10T68	1418	MESS15	9389	MESS8	5151	RSRCH1	3815
10T61	1417	MESS16	9373	MESS9	5161	RSRCH2	3848
10T62	1424	MESS17	9416	MODCON	7226	RSTURN	3854
10T63	1442	MESS18	9441	MODCON	7223	RVBARD	2436
10T64	1447	MESS19	9471	MODEND	7347	RXCOR	4714
10T65	1453	MESS2	5821	MODIFY	7288	SBWORD	4749
10T66	1469	MESS28	5516	MODLUP	7287	SOLC	6774
10T67	1466	MESS21	5546	MODH81	7221	SOLD	6779
10T68	1478	MESS22	5544	MODH82	7222	SORC	6776
10T69	1475	MESS23	5615	MODPT1	7224	SDND	6777
10T7	8328	MESS24	5635	MODPT2	7225	SD80	6773
10T78	1588	MESS25	5696	MODDIR	7227	S888	6771
10T71	1584	MESS26	5788	MODTAB	7238	ROST	6772
10T72	1518	MESS27	5722	MPNTR	2319	SEARCH	4488
10T73	1523	MESS28	8744	OGNT	2341	SELCY1	2834
10T74	1524	MESS29	5773	OK	2744	SELCY2	2854
10T75	1531	MESS3	5844	ONUMB	2348	SELCY	2888
10T76	1532	MESS38	6886	OPLOOP	2384	SELENR	4889
10T77	1533	MESS31	6892	OPRINT	2316	SELHLT	4234
10T78	1543	MESS32	6181	OUT	8889	SELHES	4238
10T8	8486	MESS33	6148	PASCNT	3497	SING1	8838

SING2	8645	WRITE2	4321
SING3	8669	WRITER	3671
SING4	8784	WRORER	4288
SING5	8723	WRORLY	4282
SING6	1882	WRQUAD	4781
SING7	1881	WRTLCK	3781
SING8	8615	WRRL1	3885
SINGLE	8688	WRRL2	3876
SLOOK	4478	XPER	3888
SP1CT	3456	XXX	4382
SP1PT	3487		
SP2CT	3474		
SP2PT	3475		
SPEC1	3443		
SPEC1D	3468		
SPEC2	3461		
SPEC2D	3476		
SRCH8	4486		
SRCH1	4415		
SRCH2	4448		
SUNIT	4471		
TIME8	1334		
TIME1	1393		
TIME2	1482		
TIME3	1482		
TIME4	1445		
TIME5	1463		
TIME6	1521		
TIMING	1315		
TPNTR	3226		
TYMERR	4249		
TYMHLT	4288		
TYMHES	4262		
TYPE	8831		
UMESS	4182		
UNIT	2234		
UTSMK1	1684		
UTSMK2	1632		
UTSMK3	1781		
UTSMK4	2882		
UTSMK5	2835		
UTSMK6	2842		
UTSMRK	1688		
WADDR	4344		
WCNT	4478		
WCOUNT	4545		
WLB	2673		
WLI	2717		
WRDMES	4214		
WREAD	3824		
WRITE	4472		
WRITE1	4512		

ERRORS DETECTED: 8

LINKS GENERATED: 147

RUN-TIME: 21 SECONDS

3K CORE USED

