

IDENTIFICATION

-----

PRODUCT CODE: MAINDEC-08-DHTMC-A-D  
PRODUCT NAME: TMS-E DRIVE FUNCTION TIMER  
DATE CREATED: JULY 16, 1973  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHOR: PATRICK COYNE / LEONARD E. BEYERSDORFER

THE MATERIAL IN THIS DOCUMENT IS FOR INFORMATION PURPOSES ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OF SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

COPYRIGHT (C) 1973

DIGITAL EQUIPMENT CORPORATION  
MAYNARD, MASS. 01754

~~TMDRFF~~ TMTIM



NOTE

\*\*\*\*

THERE ARE SIX DIAGNOSTIC PROGRAMS ASSOCIATED WITH THE TMB-E DEC MAGTAPE CONTROL AND ITS TRANSPORT SYSTEM, ALTHOUGH PHYSICALLY SEPARATE, THESE PROGRAMS MUST BE TREATED AS A LARGE INTEGRATED TEST, AND TO ENSURE PROPER SYSTEM OPERATION, THESE TESTS MUST BE EXECUTED IN THE ORDER DELINEATED BELOW.

IF A GIVEN TEST SHOULD FAIL AND IT APPEARS THAT A FIX HAS BEEN FOUND, ALL PROGRAMS MUST ONCE AGAIN BE RUN, ONLY WHEN ALL TESTS HAVE RUN WITHOUT ANY UNACCEPTABLE ERRORS CAN THE TMB-E SYSTEM BE CONSIDERED UP.

TMB-E DIAGNOSTIC PROGRAMS' ORDER OF EXECUTION

-----

1. TMB-E CONTROL TEST PART 1 (MAINDEC-08-DHTMA)
2. TMB-E CONTROL TEST PART 2 (MAINDEC-08-DHTMB)
3. TMB-E DRIVE FUNCTION TIMER (MAINDEC-08-DHTMC)
4. TMB-E DATA RELIABILITY 9 TRACK (MAINDEC-08-DHTMD)
5. TMB-E DATA RELIABILITY 7 TRACK (MAINDEC-08-DHTME)
6. TMB-E RANDOM EXERCISER (MAINDEC-08-DHTMF)

## TABLE OF CONTENTS

-----

1,	ABSTRACT
2,	REQUIREMENTS
2,1	HARDWARE
2,2	MEMORY
2,3	PRELIMINARY PROGRAMS
3,	PROGRAM LOADING PROCEDURE
4,	PROGRAM STARTING PROCEDURE
5,	STANDARD TEST PROCEDURE
5,1	DRIVE SELECTION
5,2	TEST PROCEDURE
6,	PROGRAM CONTROLS
7,	ERRORS
7,1	TIME LIMITS SPECIFICATION
8,	RESTRICTIONS
9,	EXECUTION TIME
10,	PROGRAM DESCRIPTION
11,	LISTING

1. ABSTRACT  
-----

THE TMBE DRIVE FUNCTION TIMER ASSISTS IN THE TESTING OF THE TMBE CONTROL UNIT AND TU10 TAPE UNIT. SELECTED OPERATIONS ARE EXECUTED, TIMED, AND THE TIMES ARE THEN PRINTED (IN MILLISECONDS). THERE IS NO LIMIT OR ERROR TESTING FACILITIES IN THE PROGRAM, THE DECISION ON THE VALIDITY OF TIMES MEASURED MUST BE MADE BY THE OPERATOR. ANY CONFIGURATION OF UP TO 8 TU10 TAPE UNITS (7 AND 0 CHANNEL) MAY BE SELECTED.

2. REQUIREMENTS  
-----

2.1 HARDWARE  
-----

PDP-8/E, 8/M, 8/F  
TELETYPE OR COMPATIBLE DEVICE (TTY)  
TMB-E DECMAGTAPE CONTROL  
TU10 MASTER/SLAVE TRANSPORT SYSTEM

2.2 MEMORY  
-----

THIS PROGRAM REQUIRES 4K OF MEMORY AND MAY RESIDE IN ANY MEMORY FIELD.

2.3 PRELIMINARY PROGRAMS  
-----

ALL PROCESSOR/MEMORY DIAGNOSTICS  
TMB-E CONTROL TEST PART 1  
TMB-E CONTROL TEST PART 2

3. PROGRAM LOADING PROCEDURE  
-----

LOAD THE PROGRAM INTO ANY DESIRED MEMORY FIELD USING THE STANDARD BINARY LOADER TECHNIQUE.

4. PROGRAM STARTING PROCEDURE  
-----

- A. LOAD ADDRESS 0200
- B. LOAD THE EXTENDED ADDRESS WITH THE PROGRAM FIELD.
- C. CLEAR SWITCHES
- D. DEPRESS CLEAR, THEN CONTINUE

E. THE PROGRAM WILL PRINT ITS TITLE AND MAINDEC NUMBER, THEN ASK FOR DRIVE SELECTION, PRIOR TO MAKING DRIVE SELECTION, GO TO THE STANDARD TEST PROCEDURE IN PARAGRAPH 5.

NOTE! THE PROGRAM MAY BE RESTARTED AT ANY TIME AT ADDRESS 0201. IN THIS CASE THE PROGRAM BYPASSES PRINTING ITS TITLE AND IMMEDIATELY ASKS FOR DRIVE SELECTION,

## 5. STANDARD TEST PROCEDURE

USE OF THE STANDARD TEST PROCEDURE ENSURES PROPER TMB-E/TUI0 CHECKOUT, INFORMATION REGARDING THE TIME LIMIT SPECIFICATIONS FOR THE TAPE OPERATIONS TIMED BY THIS PROGRAM ARE GIVEN IN PARAGRAPH 7.1. NO ERROR DETECTION OR DECISIONS REGARDING PROPER FUNCTION TIMES ARE CARRIED OUT BY THE PROGRAM.

### 5.1 DRIVE SELECTION

TO SPECIFY THE DRIVE(S) TO BE TESTED, CARRY OUT THE FOLLOWING STEPS:

- A. MOUNT A SPARE REEL OF INDUSTRY COMPATIBLE MAGNETIC TAPE ON ALL DRIVES TO BE SELECTED WITH THE WRITE ENABLE RING IN PLACE (WRITE ENABLED). LOAD THE TAPES AND POSITION AT BOT, ASSIGN DRIVE NUMBERS IN THE FOLLOWING SEQUENCE! 01711;612;513;4) AND THEN SWITCH ON LINE.
- B. START OR RESTART THE PROGRAM AS DESCRIBED IN PARAGRAPH 4.
- C. RESPOND TO "SELECT DRIVES" BY SETTING SR BIT N = 1 FOR EACH DRIVE TO BE TESTED (DRIVE N). THEN DEPRESS CONTINUE.
- D. RESPOND TO "SELECT 7 AND/OR 9 TRACK OPERATION" BY SETTING SR BIT N = 1 ONLY FOR THOSE SELECTED DRIVES WHICH ARE 9 TRACK UNITS. THEN DEPRESS CONTINUE.

### 5.2 TEST PROCEDURE

ACCOMPLISH THE FOLLOWING STEPS FOR ALL DRIVES IN THE SYSTEM UNDER TEST.

- A. SELECT ALL DRIVES IN THE SYSTEM UNDER TEST AS DESCRIBED IN PARAGRAPH 5.1 ABOVE.
- B. AS SOON AS TRACK SELECTION IS COMPLETE, THE PROGRAM WILL START TIMING THE VARIOUS TAPE OPERATIONS AND INDICATING ALL TIMES ON THE TTY.
- C. REPEAT THE ENTIRE TIMING PROCEDURE 2 TIMES.
- D. COMPARE THE RESULTS AS PRINTED ON THE TTY WITH THE LIMITS SPECIFIED IN PARAGRAPH 7.1. ADDITIONAL INFORMATION IS

GIVEN IN THE PROGRAM DESCRIPTION IN PARAGRAPH 10.

6. PROGRAM CONTROLS

THERE ARE NO SWITCH REGISTER OPTIONS ASSOCIATED WITH THIS PROGRAM.

7. ERRORS

THERE IS NO ERROR DETECTION INCLUDED IN THIS PROGRAM. ALL DECISIONS MUST BE MADE BY THE USER BY COMPARING THE FUNCTION TIMES PRINTED ON THE TTY WITH THOSE TIMES GIVEN IN THE TABLE BELOW.

7.1 TIME LIMITS SPECIFICATION

THE TABLE BELOW LISTS THE TIME LIMITS IN THE SAME FORMAT AS THEY ARE PRINTED ON THE TTY. TIMES LISTED UNDER UNIT 0 ARE 9 TRACK TIMES, AND THOSE UNDER UNIT 1 ARE 7 TRACK TIMES. ALL TIMES ARE IN MILLISECONDS.

FUNCTION	(9TRACK)	+/-	(7TRACK)	+/-
800 BPI	UNIT0		UNIT1	
WR FM BOT DELAY	185	15	185	15
WRITE SHUTDOWN	2.7	0.5	2.7	0.5
WRITE START	8.9	1.0	12.6	1.0
WR NONSTOP GAP	11.5	2.0	14.5	2.0
BKSP SHDWN*SDWN	15	5	19	5
READ SHUTDOWN	1.8	0.5	1.8	0.5
WRITE XIRG	95	10	95	10
LAST CHR TO CUR	0.26	0.1	0.26	0.1
RD FM BOT DELAY	185	15	185	15
SPCE SHDWN*SDWN	14	5	14	5
WRITE EOF	100	10	105	10
ER TO EF SF TME	100	10	105	10
WR TO ERASE HEAD	12	5	12	5
1 INCH DATA TIME	22	1	22	1

GAP1  
GAP2  
GAP3  
GAP4  
GAP5  
GAP6  
GAP7  
GAP8

GAPSI 8>7>6>5>4>1; 1-2<1.7; 2>3

556 BPI	UNIT1	+/-	(7TRACK)
WR FM BOT DELAY	185	15	
WRITE SHUTDOWN	2.7	0.5	
LAST CHR TO CUR	0.35	0.1	
BKSP SHDWN*SDWN	19	5	
READ SHUTDOWN	1.8	0.5	
1 INCH DATA TIME	22	1	

	UNIT1	+/- (7TRACK)
200 BPI	185	15
WR FM BOT DELAY	2.7	0.5
WRITE SHUTDOWN	1.05	0.2
LAST CHR TO CUR	19	5
BKSP SHOWN*SDWN	1.8	0.5
READ SHUTDOWN	22	1
1 INCH DATA TIME		

8. RESTRICTIONS  
-----

NONE.

9. EXECUTION TIME  
-----

EXECUTION TIME VARIES DEPENDING ON THE NUMBER AND TYPES OF DRIVES BEING TESTED.

10. PROGRAM DESCRIPTION  
-----

10.1 WRITE FROM BOT DELAY

THIS IS THE TIME NECESSARY TO MOVE BEGINNING OF TAPE (BOT) MARKER APPROXIMATELY 6 INCHES PAST THE WRITE HEAD. THE FIRST RECORD ON TAPE MUST BE WRITTEN AT LEAST 3" AWAY FROM THE BOT MARKER.

PROCEDURE TO MEASURE TIME!

- A. REWIND TO BOT.
- B. INITIALIZE WC & CA LOCATIONS.
- C. LOAD COMMAND REGISTER WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- D. LOAD FUNCTION WITH WRITE & GO BITS.
- E. MONITOR CA REGISTER UNTIL IT INDICATES FIRST WORD HAS BEEN TAKEN.
- F. THE TIME FROM "GO" UNTIL FIRST WORD IS OUTPUT IS "WRITE FROM BOT DELAY."

10.2 WRITE SHUTDOWN

THIS IS THE TIME NECESSARY TO CONTINUE MOVING TAPE AFTER A RECORD IS WRITTEN SO THAT THE PROPER INTERRECORD GAP WILL EXIST BETWEEN RECORDS.

PROCEDURE TO MEASURE TIME!

- A. REWIND TO BOT.
- B. INITIALIZE WC & CA LOCATIONS TO WRITE ONE



200(10) WORD RECORD FROM BOT,

- C. LOAD COMMAND REGISTER WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- D. LOAD FUNCTION WITH WRITE & GO BITS.
- E. AFTER MTF SETS INDICATING THE END OF RECORD, INITIATE A CONTINUOUS WRITE AND TIME TILL THE FIRST WORD IS TRANSFERRED FROM MEMORY TO THE TMB/E CONTROL (OCCURS APPROXIMATELY ON THE LEADING EDGE OF SDWN).

### 10.3 WRITE START

THE TIME NECESSARY FOR TAPE TO ACCELERATE TO FULL SPEED + GUARANTEE A 1/2" INTERRECORD GAP.

PROCEDURE TO MEASURE TIME:

SAME AS "WRITE FROM BOT" EXCEPT NOW WE ARE NOT AT BOT.

- A. INITIALIZE WC & CA REGISTER.
- B. LOAD COMMAND WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- C. LOAD FUNCTION WITH WRITE & GO BITS.
- D. MONITOR CA FOR ITS SECOND INCREMENT.
- E. THE TIME FROM "GO" UNTIL 2ND INCREMENT OF CA IS APPROXIMATELY EQUAL TO "WRITE START".

### 10.4 WRITE NONSTOP GAP

WRITE NONSTOP GAP IS EQUIVALENT TO THE SUM OF "WRITE SHUTDOWN" & "WRITE START" AND IS THE TIME NECESSARY TO INSURE THAT THE INTERRECORD GAP WILL BE AT LEAST 1/2" WHEN WRITING NON-STOP.

PROCEDURE TO MEASURE TIME:

- A. INITIALIZE WC & CA LOCATIONS
- B. LOAD COMMAND WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- C. LOAD FUNCTION;WRITE, GO BITS.
- D. WAIT FOR "JOB DONE."
- E. ISSUE "CLF."
- F. RE-INITIALIZE WC & CA LOCATIONS.
- G. LOAD FUNCTION;WRITE, GO BITS.
- H. TIME FROM 2ND "GO" UNTIL FIRST WORD OF 2ND WRITE IS OUTPUT IS "WRITE NONSTOP GAP."

10.5 BACKSPACE SHUTDOWN + SETTLE DOWN DELAY

THIS IS THE LENGTH OF TIME NECESSARY TO GUARANTEE THAT IF A WRITE OPERATION FOLLOWS A BACKSPACE THE TAPE WILL BE POSITIONED SUCH THAT ALL PREVIOUS DATA IS IN FRONT OF THE WRITE AND ERASE HEADS AND WILL BE ERASED.

PROCEDURE TO MEASURE TIME I

RECORD IS USED WHICH WAS JUST WRITTEN IN PREVIOUS TEST "WRITE NONSTOP GAP",

- A. LOAD COMMAND WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- B. SET WC TO BACKSPACE 1 RECORD.
- C. ISSUE BACKSPACE, GO.
- D. AFTER JOB DONE BECOMES A ONE INDICATING THE BEGINNING OF THE RECORD MONITOR TUR UNTIL IT BECOMES A "1", THE TIME FROM JOB DONE UNTIL TUR IS "BACKSPACE SHUTDOWN + SETTLEDOWN DELAY,"

10.6 READ SHUTDOWN

THIS IS THE AMOUNT OF TIME NECESSARY TO CONTINUE MOVING TAPE AFTER A RECORD IS READ SO THAT THERE IS ENOUGH GAP FOR TAPE TO BE FULLY ACCELERATED IF A READ IS FOLLOWED BY A BACKSPACE. "READ SHUTDOWN" MUST ALSO BE LESS THAN "WRITE SHUTDOWN" TO GUARANTEE THAT THE WRITE AND ERASE HEADS WILL BE POSITIONED SUCH THAT ALL PREVIOUS DATA IS IN FRONT OF THE HEADS AND WILL BE ERASED IF A WRITE FOLLOWS A READ. IN ADDITION WHEN A WRITE FOLLOWS A READ THE INTERRECORD GAP MUST STILL BE AT LEAST 1/2".

PROCEDURE TO MEASURE TIME I

- A. RECORD USED IN "BACKSPACE SHUTDOWN" IS READ.
- B. INITIALIZE WC & CA REGISTERS.
- C. ISSUE READ FUNCTION, 800 BPI, GO.
- D. AFTER MTF SETS INDICATING THE END OF RECORD, INITIATE A CONTINUOUS WRITE AND TIME TILL THE FIRST WORD IS TRANSFERRED FROM MEMORY TO THE TMR/E CONTROL (OCCURS APPROXIMATELY ON THE LEADING EDGE OF SDWN).

10.7 WRITE XIRG

THIS IS THE TIME NECESSARY TO CAUSE THE GENERATION OF AN INTERRECORD GAP THAT IS AT

LEAST 3" LONG AS COMPARED WITH THE NORMAL 3/8" GAP. THE PURPOSE IS TO ELIMINATE WRITE ERRORS THAT MAY BE CAUSED BY A DEFECTIVE AREA ON TAPE. NORMALLY ONE REWRITE WITH XIRG WOULD BE SUFFICIENT IF IT ISN'T THE PROCEDURE WOULD BE TO REPEAT THE "BACKSPACE-REWRITE WITH XIRG" SEQUENCE UNTIL A RECORD IS WRITTEN WITHOUT ERRORS. EACH SUCCESSIVE REWRITE WOULD ADD 3" TO THE INTERRECORD GAP UNTIL "GOOD" TAPE WAS REACHED.

PROCEDURE TO MEASURE TIME:

- A. TAPE IS NOT BOT.
- B. INITIALIZE WC & CA REGISTERS.
- C. ISSUE WRITE WITH XIRG FUNCTION, 800 BPI, GO.
- D. MONITOR CA TO DETERMINE WHEN FIRST WORD IS OUTPUT.
- E. THE TIME FROM "GO" UNTIL FIRST WORD IS OUTPUT IS "WRITE WITH XIRG".

#### 10.8 LAST CHARACTER TO CU READY

THIS IS THE AMOUNT OF TIME IT TAKES FOR THE CONTROL TO SENSE 3 MISSING WORDS ON TAPE (END OF RECORD) UNTIL "CU READY" BECOMES A "1."

PROCEDURE TO MEASURE TIME:

- A. PROGRAM USES SAME RECORD THAT WAS WRITTEN DURING "WRITE XIRG."
- B. LOAD COMMAND WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- C. INITIALIZE WC TO SPACE REVERSE 1 RECORD.
- D. WAIT FOR JOB DONE.
- E. REINITIALIZE WC & CA LOCATIONS.
- F. LOAD COMMAND WITH DRIVE SELECT, 800 BPI, ODD PARITY.
- G. ISSUE READ FUNCTION, GO.
- H. WAIT UNTIL WC=0 AND THEN MONITOR "CU READY" UNTIL IT BECOMES A "1".
- J. THE TIME FROM WC=0 UNTIL "CU READY" = 1 IS "LAST CHARACTER TO CU READY".

#### 10.9 READ FROM BOT

THE FIRST RECORD WRITTEN ON TAPE IS SUPPOSED

TO BE AT LEAST 6" FROM THE BOT MARKER,  
IN THE EVENT THAT THIS CONDITION WASN'T MET  
IT IS STILL DESIREABLE TO READ THE RECORD,  
READ FROM BOT IS THE TIME FROM WHEN A READ  
FUNCTION IS ISSUED UNTIL THE FIRST WORD IS  
OUTPUT,

PROCEDURE TO MEASURE TIME!

- A. THE RECORD THAT WAS WRITTEN JUST OFF BOT DURING "WRITE START" IS USED.
- B. TAPE IS REWOUND TO BOT.
- C. INITIALIZE WC & CA REGISTERS.
- D. ISSUE READ FUNCTION, 800 BPI, GO.
- E. MONITOR CA TO DETERMINE WHEN FIRST WORD IS OUTPUT.
- F. THE TIME FROM "GO" UNTIL FIRST BREAK IS "READ FROM BOT,"

10.10 SPACE SHUTDOWN + SETTLEDOWN DELAY

SPACE SHUTDOWN IS THE AMOUNT OF TIME NECESSARY TO CONTINUE MOVING TAPE AFTER A RECORD IS SPACED OVER IN THE FORWARD DIRECTION FOR THE SAME REASONS AS "READ SHUTDOWN",

PROCEDURE TO MEASURE TIME!

- A. SAME RECORD IS USED AS IN T3 WRITE START.
- B. INITIALIZE WC TO SPACE FORWARD 1 RECORD.
- C. ISSUE SPACE FORWARD FUNCTION, 800 BPI, GO.
- D. MONITOR JOB DONE UNTIL IT IS =1.
- E. TIME FROM JOB DONE =1 UNTIL TUR IS "SPACE SHUTDOWN + SETTLEDOWN DELAY",

10.11 WRITE EOF

TO WRITE AN END OF FILE MARK IT IS NECESSARY FOR TAPE TO MOVE 3 INCHES BEFORE WRITING A RECORD WITH EXTENDED INTERRECORD GAP, HOWEVER, AN EOF MARK CORRESPONDS TO A 1 WORD RECORD, THE TIME SHOULD BE SLIGHTLY LARGER THAN "WRITE XIRG",

PROCEDURE TO MEASURE TIME!

- A. TAPE UNIT IS REWOUND TO BOT.
- B. INITIALIZE WC & CA REGISTERS.

- C. ISSUE WRITE, 800 BPI, GO.
- D. WAIT FOR JOB DONE.
- E. ISSUE CLF.
- F. ISSUE WRITE EOF FUNCTION, 800 BPI, GO.
- G. WAIT FOR JOB DONE #1
- H. THE TIME FROM "GO" UNTIL "JOB DONE" IS "WRITE EOF."

10.12 EOR TO EOF TIME

THIS IS THE TIME NEEDED TO MOVE TAPE FROM THE END OF A RECORD TO AN END OF FILE MARK WRITTEN AFTER IT. THE PROCEDURE USED TURNS OUT TO BE A TEST OF THE WRITE AND ERASE HEAD POLARITIES. IF THE TIME PRINTED IS EQUAL TO ZERO IT IS AN INDICATION THAT THE EOF WAS NOT FOUND WHEN "JOB DONE" BECAME A ONE.

THIS COULD INDICATE ONE OR MORE OF THE FOLLOWING PROBLEMS:

1. ERASE HEAD POLARITY REVERSED.
2. ERASE HEAD CURRENT NOT SUFFICIENT TO FULLY SATURATE TAPE.
3. ONE OR MORE OF WRITE HEAD TRACKS POLARITY REVERSED.
4. ONE OR MORE SENSITIVE READ AMPLIFIERS.
5. WRITE EOF FUNCTION DIDN'T REALLY WRITE AN EOF MARK.

OTHERWISE "EOR TO EOF SPACE TIME" SHOULD BE SLIGHTLY LARGER THAN "WRITE EOF."

PROCEDURE TO MEASURE TIME:

- A. A RECORD AND EOF WAS PREVIOUSLY WRITTEN FROM BOT FOR "WRITE EOF".
- B. TAPE IS REWOUND TO BOT.
- C. REWRITE RECORD OVER PREVIOUSLY WRITTEN RECORD.
- D. BACKSPACE OVER RECORD JUST WRITTEN.
- E. SET WC REGISTER TO SPACE 2 RECORDS.
- F. ISSUE SPACE FORWARD FUNCTION, SET GO.
- G. WAIT FOR WC REGISTER TO INDICATE THAT FIRST RECORD HAS BEEN SPACED OVER THEN MONITOR "CU READY" UNTIL IT BECOMES A

"1", AFTER "CU READY" CHECK TO SEE IF "EOF"  
IS A 1 IN STATUS REGISTER, IF EOF IS NOT  
SET THEN ZERO TIME COUNTER.

H. TIME FROM WC=1 UNTIL "CU READY" IS  
"EOR TO EOF SPACE TIME."

### 10.13 WRITE TO ERASE HEAD

THE "WRITE TO ERASE HEAD" TEST INSURES THAT THE  
TAPE IN FRONT OF THE WRITE HEAD IS ERASED  
DURING A WRITE OPERATION.

#### PROCEDURE TO MEASURE TIME

- A. A LONG RECORD HAS BEEN WRITTEN FROM BOT. SAME  
RECORD THAT WAS USED TO TIME "WRITE FROM BOT  
DELAY"
- B. TAPE IS REWOUND TO BOT.
- C. WC REGISTER IS INITIALIZED FOR A 3 WORD RECORD  
CA REGISTER IS INITIALIZED TO 3775.
- D. LOAD COMMAND REGISTER WITH DRIVE SELECT,  
800 BPI, ODD PARITY.
- E. LOAD FUNCTION REGISTER WITH WRITE & GO BITS.
- F. MONITOR CA UNTIL IT IS 3777 INDICATING 2 WORDS  
HAVE BEEN OUTPUT, THEN ISSUE CLT WHICH  
STOPS ALL DATA TRANSFERS & CAUSES THE DRIVE  
TO SHUTDOWN.
- G. REWIND TO BOT.
- H. INITIALIZE CA TO 3775
- I. ISSUE READ FUNCTION, 800 BPI, SET GO.
- J. MONITOR CA UNTIL IT IS 3777 THEN  
GO TO TIMING ROUTINE AND TIME UNTIL CA=4000  
THIS TIME WILL INDICATE THE DISTANCE BETWEEN  
THE LAST WORD OF THE NEW DATA AND THE  
FIRST WORD OF THE OLD DATA WHICH IS ALSO  
THE AMOUNT OF TAPE THAT WAS ERASED BY  
THE ERASE HEAD DURING THE WRITE OPERATION.

### 10.14 ONE INCH DATA TIME

ONE INCH OF DATA, 800 BPI (ALSO 556 & 200 IF  
7 CHANNEL UNIT), IS WRITTEN AND TIMED TO  
DETERMINE IF TAPE IS MOVING AT PROPER SPEED.

#### PROCEDURE TO MEASURE TIME

- A. INITIALIZE WC & CA REGISTERS. CA(3776) WC(=2000)

- B. ISSUE WRITE FUNCTION, 800, 556 OR 200 BPI, GO,
- C. WAIT FOR CURRENT MEMORY ADDRESS REGISTER TO INDICATE FIRST WORD IS TAKEN AND THEN MONITOR WORD COUNT UNTIL EQUAL TO ZERO,
- D. TIME FROM FIRST WORD OUTPUT UNTIL WORD COUNT=0 IS "ONE INCH DATA TIME."

#### 10.15 GAP CONSISTENCY

FOR PROPER OPERATION THE INTERRECORD GAPS ON TAPE MUST ALWAYS BE AT LEAST 1/2 OF AN INCH. THIS WILL ALLOW DATA WRITTEN USING ONE TAPE UNIT TO BE READ ON ANOTHER TAPE UNIT WHEN THE START/STOP CHARACTERISTICS OF EACH UNIT ARE DIFFERENT. THE MINIMUM GAP SIZE OF 1/2 AN INCH IS GENERATED WHEN A WRITE FOLLOWS A READ. ALL OTHER GAPS SHOULD BE LARGER DEPENDING ON HOW THEY WERE WRITTEN.

#### PROCEDURE TO MEASURE TIME!

- A. A TOTAL OF NINE RECORDS ARE WRITTEN ON TAPE (FROM BOT) UTILIZING DIFFERENT SEQUENCES TO GENERATE THE INTERRECORD GAPS.
- B. THE TAPE IS REWOUND TO BOT.
- C. INITIALIZE WORD COUNT AND CURRENT ADDRESS REGISTERS.
- D. ISSUE READ FUNCTION, 800 BPI, SET "GO".
- E. WAIT FOR "CU READY" TO BECOME A 1 THEN REPEAT STEP C AND RESET "GO" TO CONTINUE NONSTOP.
- F. MONITOR CA REGISTER TO DETERMINE WHEN FIRST WORD IS INPUT.
- G. THE TIME FROM WHEN "GO" IS RESET UNTIL THE FIRST WORD IS INPUT WILL REFLECT THE SIZE OF THE GAP.
- H. STEPS E,F ARE REPEATED UNTIL ALL 8 GAPS ARE MEASURED.

#### PROGRAM SEQUENCE FOR EACH GAP!

- GAP1 WRITE FOLLOWED BY A WRITE (NONSTOP),
- GAP2 WRITE FOLLOWED BY A WRITE (START/STOP),
- GAP3 READ FOLLOWED BY A WRITE (START/STOP),
- GAP4 WRITE-BACKSPACE FOLLOWED BY A WRITE (START/STOP),
- GAP5 SAME AS GAP4 EXCEPT WRITE-BACKSPACE REPEATED 2 TIMES.

GAP6 SAME AS GAP4 EXCEPT WRITE=BACKSPACE REPEATED 3 TIMES;  
GAP7 SAME AS GAP4 EXCEPT WRITE=BACKSPACE REPEATED 4 TIMES;  
GAP8 SAME AS GAP4 EXCEPT WRITE=BACKSPACE REPEATED 5 TIMES;  
GAP LENGTHS SHOULD REFLECT THE FOLLOWING RELATIONSHIP:  
8>7>6>5>4>1, 1-2<1,7, 2>3

10.16 FUNCTIONS AT 556 BPI

ALL OF THE PREVIOUS TESTS USED THE DENSITY OF 800 BPI.  
IF A 7 CHANNEL DRIVE IS SELECTED IT IS USEFUL TO RUN SEVERAL OF  
THE TESTS AGAIN USING DENSITY OF 556 BPI, REFERENCE  
THE PROPER PARAGRAPHS FOR A DESCRIPTION OF EACH TEST.

10.17 FUNCTIONS AT 200 BPI.

SAME AS ABOVE,  
REFERENCE 8.16, "FUNCTIONS AT 556 BPI".

11. LISTING (ATTACHED)

-----



/TM8E DRIVE FUNCTION TIMER MAINDEC-08-DHTMC-A-L  
 /COPYRIGHT 1972, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS,

//INSTRUCTION EQUALITIES//

6701	LWCR=6701	/LOAD WORD COUNT, CLR AC,
6702	CWCR=6702	/CLEAR WORD COUNT,
6703	LCAR=6703	/LOAD CURRENT ADDRESS, CLR AC,
6704	CCAR=6704	/CLEAR CURRENT ADDRESS,
6705	LWCR=6705	/LOAD COMMAND REGISTER, CLR AC,
6706	LFGR=6706	/LOAD FUNCTION REGISTER, CLR AC,
6707	LDBR=6707	/LOAD DATA BUFFER, CLR AC,
6711	RWCR=6711	/READ WORD COUNT,
6712	CT=6712	/CLEAR TRANSPORT,
6713	RCAR=6713	/READ CURRENT ADDRESS,
6714	RMSR=6714	/READ MAIN STATUS,
6715	RCMR=6715	/READ COMMAND REGISTER,
6716	RFSR=6716	/READ FUNCTION & STATUS 1 REGISTERS,
6717	RDBR=6717	/READ DATA BUFFER,
6721	SKEF=6721	/SKIP IF ERROR FLAG IS SET,
6722	SKCB=6722	/SKIP IF CONTROL IS NOT BUSY,
6723	SKJD=6723	/SKIP ON JOB DONE,
6724	SKTR=6724	/SKIP ON TUR,
6725	CLF=6725	/CLR THE CONTROLLER & TRANSPORT MASTER IF TUR, /IF NOT, CLR MYTF, EP & STATUS REGISTERS, /CHECK FOR DATA LATE ERROR, /SET BREAK REQUEST FOR ONE DATA BREAK,
6726	CKDL=6726	
6727	SBRM=6727	
4497	TIMER=JMS I	XTIME

0010	0010	*10
0010	0000	INDX10, 0
0012	0012	*12
0012	0000	INDX12, 0
0020	0020	*20
0020	0000	DRV1A0, 0000
0021	1000	1000
0022	2000	2000
0023	3000	3000
0024	4000	4000
0025	5000	5000
0026	6000	6000
0027	7000	7000
0030	0000	ALLTU, 0
0031	0000	0
0032	0000	0
0033	0000	0
0034	0000	0
0035	0000	0
0036	0000	0
0037	0000	0
0040	0000	TU7, 0

0041	0000	0
0042	0000	0
0043	0000	0
0044	0000	0
0045	0000	0
0046	0000	0
0047	0000	0
0050	7777	ONCE, 7777
0051	5440	XCRLF, CRLF
0052	0000	NUMPNT, 0
0053	0000	TBLALL, 0
0054	0000	TBLTU7, 0
0055	5325	XHEAD, ALLMSG
0056	5345	XHEAD1, TU7MSG
0057	5011	XTIME, TIME
0060	0000	CNTALL, 0
0061	0000	CNT7, 0
0062	0020	DRVPT, DRV1A0
0063	5200	XRWND, RWIND
0064	5216	XRWND7, RWIND7
0065	5000	XWCCA, WCCA
0066	0402	ODD000, 0402
0067	7777	TITLE, 7777
0070	0000	DRVSEL, 0
0071	0000	OUT, 0
0072	5264	SELALL, ALLTU7
0073	5304	SELTU7, ALLTU7
0074	5711	XMESS, MESSAGE
0075	5425	XPRINT, PRINT
0076	5501	XDEC, DEQPRT
0077	0000	VFB20, 0
0100	0000	VFBUPR, 0
0101	5416	XOTY, OTY
0102	7774	KM4, 7774
0103	0000	HDIWND, 0
0104	0000	PRNCI, 0
0105	5132	NULL, XNULL
0106	0000	TEMP1, 0
0107	0000	MIKCT, 0
0110	0000	MILICT, 0
0111	7634	NCPU, -144
0112	0144	PCPU, 144
0113	4100	K4100, 4100
0114	7777	K7777, 7777
0115	7100	K7100, 7100
0116	2100	K2100, 2100
0117	4500	K4500, 4500
0120	6100	K6100, 6100
0121	5100	K5100, 5100
0122	1100	K1100, 1100
0123	7776	K7776, 7776
0124	0001	K0001, 0001
0125	0100	K0100, 0100
0126	4001	K4001, 4001

```

0127 1000 K1000, 1000
0130 0212 K212, 0212
0131 0215 K215, 0215
0132 7777 KM1, -1
0133 7776 KM2, -2
0134 7775 KM3, -3
0135 7770 KM10, -10
0136 5616 XBAK, BAKSPA
0137 5600 XRITE1, RITE1
0140 5633 XREAD, READ
0141 5631 XREAD1, READ1
0142 0400 K0400, 0400
0143 0000 XBPI, 0
0144 2214 LINK10, T10
0145 5704 XEND, FINI
0146 5670 XREAD2, READ2
0147 0000 INCH, 0
    
```

```

0150 0260 NUMTBL, 260
0151 0261 261
0152 0262 262
0153 0263 263
0154 0264 264
0155 0265 265
0156 0266 266
0157 0267 267
    
```

```

0160 0000 ALLTBL, 0
0161 0000 0
0162 0000 0
0163 0000 0
0164 0000 0
0165 0000 0
0166 0000 0
0167 0000 0
    
```

```

0170 0000 TU7TBL, 0
0171 0000 0
0172 0000 0
0173 0000 0
0174 0000 0
0175 0000 0
0176 0000 0
0177 0000 0
    
```

```

0200 0200 *200
0201 7610 START, SKP CLA
0201 5206 RESTART, JMP SELECT
0202 4474 MSG1, JMS I XMESS
0203 6000 MSG1
0204 4474 MSG1, JMS I XMESS
0205 6017 MSG2
    
```

/BEGINNING OF HEADER.

```

//ROUTINE FOR TRANSPORT SELECTION AND
//7 OR 9 TRACK OPERATION,
//
0206 4474 SELECT, JMS I XMESS //SET SWITCHES TO SELECT TRANSPORTS,
0207 6033 MSG3
0210 7402 HLT
0211 7604 LAS //GET SWITCHES FOR TRANSPORT SELECTION,
0212 3344 DCA DRVSAV //SAVE SWITCHES,
0213 4474 JMS I XMESS //SET SWITCHES FOR 7 OR 9 TRACK OPERATION,
0214 6044 MSG4
0215 7402 HLT
0216 7604 LAS //GET SWITCHES FOR 7 OR 9 TRACK SELECTION,
0217 3343 DCA TRKSAV //SAVE SWITCHES,
0220 3060 DCA CNTALL //INITIALIZE COUNTERS,
0221 3061 DCA CNT7
0222 1377 TAD (DRVTAB //SETUP POINTER
0223 3062 DCA DRVPNT //FOR DRIVE TABLE,
0224 1376 TAD (ALLTU //SETUP POINTER FOR TABLE WHICH WILL HOLD
0225 3340 DCA XALL //ALL TU-10'S TO BE TESTED,
0226 1375 TAD (TU7 //SETUP POINTER FOR TABLE WHICH WILL HOLD ONLY
0227 3341 DCA XTU7 //7 TRACK TU-10'S TO BE TESTED,
0230 1374 TAD (NUMTBL
0231 3052 DCA NUMPNT
0232 1373 TAD (ALLTBL
0233 3053 DCA TBLALL
0234 1372 TAD (TU7TBL
0235 3054 DCA TBLTU7
0236 1345 TAD K7770 //SETUP COUNTER TO EXIT THIS ROUTINE WHEN ALL
0237 3342 DCA DRVCNT //BITS OF SELECT WORDS HAVE BEEN CHECKED,
0240 1344 CHKWRD, TAD DRVSAV //PICKUP DRIVE SELECT WORD,
0241 7700 SMA CLA //MINUS AC INDICATES DRIVE IS SELECTED,
0242 5204 JMP UPDATE //THIS DRIVE NOT SELECTED UPDATE ALL POINTERS,
0243 1402 TAD I DRVPNT //SAVE DRIVE SELECTED IN TABLE CONTAINING
0244 3740 DCA I XALL //ALL DRIVES TO BE TESTED,
0245 1402 TAD I NUMPNT
0246 3493 DCA I TBLALL
0247 2060 ISB CNTALL //COUNTER FOR SELECT ALL DRIVES TABLE,
0250 1343 TAD TRKSAV //PICKUP TRACK SELECT WORD,
0251 7710 SPM CLA //MINUS AC INDICATES 9 TRACK OPERATION,
0252 5202 JMP UPDATE-2 //9 TRACK SELECTED, EXIT AND UPDATE POINTERS,
0253 1402 TAD I DRVPNT //7 TRACK SELECTED, SAVE IT IN 7 TRACK
0254 3741 DCA I XTU7 //SELECT TABLE,
0255 1402 TAD I NUMPNT
0256 3494 DCA I TBLTU7
0257 2061 ISB CNT7 //KEEP COUNT OF ALL 7 TRACK DRIVES SELECTED,
0260 2054 ISB TBLTU7 //UPDATE ALL TABLE POINTERS,
0261 2341 ISB XTU7
0262 2053 ISB TBLALL
0263 2340 ISB XALL
0264 2342 UPDATE, ISB DRVCNT //CHECKED ALL BITS OF SELECT WORDS?
0265 7410 SKP //NO, CONTINUE CHECKING
0266 5303 JMP ENBCHK //YES, BEGIN TESTING SELECTED DRIVES,
0267 2042 ISB DRVPNT
0270 2052 ISB NUMPNT
    
```

```

0271 7100 CLL
0272 1344 TAD DRVSAV /GET DRIVE SELECT WORD;
0273 7004 RAL /SETUP NEXT BIT TO BE CHECKED.
0274 3244 DCA DRVSAV /SAVE IT.
0275 7100 CLL
0276 1344 TAD TRKSAV /GET TRACK SELECT WORD;
0277 7004 RAL /SETUP NEXT BIT TO BE CHECKED.
0300 3343 DCA TRKSAV /SAVE IT.
0301 7100 CLL
0302 5240 JHP CHKHRD /DO IT ALL AGAIN;
0303 1061 ENDCHK, TAD CNT7 /CHECK TO SEE IF ANY DRIVES
0304 7040 SEA CLA /HAVE BEEN SELECTED.
0305 5311 JHP ,=4
0306 1060 TAD CNTALL
0307 7050 SNA CLA
0310 5206 JHP SELECT /NO DRIVES SELECTED, RETURN TO START.
0311 1371 TAD (3767 /FILL WRITE BUFFER WITH ONES.
0312 3346 DCA ONES
0313 1370 TAD (=630
0314 3347 DCA ONESCT
0315 7040 CMA
0316 3050 DCA ONCE
0317 7040 CMA
0320 3746 DCA I ONES
0321 2346 ISR ONES
0322 2347 ISR ONESCT
0323 5317 JHP ,=4
0324 1367 TAD (6377
0325 3010 DCA INOX10
0326 4474 JMS I XMESS
0327 6067 MSG5
0330 4474 JMS I XMESS
0331 6102 MSG7
0332 4455 JMS I XHEAD
0333 4451 JMS I XCRLF
0334 3565 DCA I NULL /ZERO TIME COUNTERS.
0335 3110 DCA MILICT
0336 5737 JMP I LINK1 /BEGIN TESTING.
0337 0400 LINK1, T1
    
```

```

0340 0000 XALL, ALLTU
0341 0040 XTU7, TU7
0342 0000 DRVCT, 0
0343 0000 TRKSAV, 0
0344 0000 DRVSAV, 0
0345 7770 K7770, 7770
0346 0000 ONES, 0
0347 0000 ONESCT, 0
    
```

```

0367 6377
0370 7100
0371 3707
0372 0170
    
```

```

0373 0160
0374 0150
0375 0040
0376 0032
0377 0020
0400 0400
    
```

PAGE

//TIME WRITE FROM BOT DELAY//

```

0400 7300 T1, CLA CLL
0401 1377 TAD (OUT1 /SETUP T1 EXIT LOCATION.
0402 3071 DCA OUT
0403 4463 JMS I XRNND /REWIND ALL SELECTED DRIVES TO BOT.
0404 4463 RET1, JMS I XWCCA /INITIALIZE WG, CA LOCATIONS.
0405 3770 JMS I XWCCA
0406 7470 =310 /NO
0407 4472 JMS I SELALL /GET DRIVE SELECT WORD;
0410 1066 TAD ODO000 /COMBINE DRIVE SELECT, 000 BPI, 000 PARITY
0411 6705 LGMR /LOAD COMMAND.
0412 1113 TAD K4100
0413 6720 LPSR /LOAD FUNCTION/ WRITE, GO BITS;
0414 4457 TIMER /TIME "BOT DELAY."
0415 6713 R0AR /NOT TO BE USED BY TIMER.
0416 7710 SPA CLA /CONDITION TO REMAIN IN TIMING LOOP.
0417 6724 SKTR
0420 5217 JHP ,=1 /WAIT FOR PUR.
0421 6725 CLP
0422 5204 JMP RET1 /GO AGAIN ON NEXT DRIVE.
0423 4474 OUT1, JMS I XMESS /MESSAGE ROUTINE
0424 6112 MSG1
0425 1060 TAD CNTALL
0426 3104 DCA PRNCT
0427 4475 JMS I XPRINT
    
```

//TIME WRITE SHUTDOWN//

```

0430 7300 T2, CLA CLL
0431 1377 TAD (OUT2 /SETUP T2 EXIT LOCATION.
0432 3071 DCA OUT
0433 4463 JMS I XRNND /REWIND ALL SELECTED DRIVES TO BOT.
0434 4463 RET2, JMS I XWCCA /INITIALIZE WG, CA LOCATIONS.
0435 3770 JMS I XWCCA
0436 7470 =310 /NO
0437 4472 JMS I SELALL /GET DRIVE SELECT WORD;
0440 1066 TAD ODO000 /COMBINE DRIVE SELECT, 000 BPI, 000 PARITY.
0441 6705 LGMR /LOAD COMMAND.
0442 1113 TAD K4100
0443 6706 LPSR /LOAD FUNCTION/ WRITE, GO BITS;
0444 6723 SKJD
0445 5244 JMP ,=1 /WAIT FOR JOB DONE.
0446 6725 CLP /CLEAR FLAGS
0447 4775 JMS I (HATONT /CONTINUOUS WRITE
0450 5234 JMP RET2 /MESSAGE ROUTINE
0451 4474 OUT2, JMS I XMESS
    
```

21

```

0452 6123      MSGT2
0453 1060      TAD      CNTALL
0454 3104      DCA      PRNCT
0455 4475      JMS I   XPRINT
0456 5657      JMP I   LINK3
0457 0600      LINK3, T3

0575 5400
0576 0451
0577 0423
0600 0600

PAGE
//TIME WRITE START//
//
0600 7300      T3,   CLA CLL
0601 1377      TAD      (OUT3      /SETUP T3 EXIT LOCATION.
0602 3071      DCA      OUT          /INITIALIZE WC + CA LOCATIONS,
0603 4465      RET3,  JMS I   XWCCA      /CA
0604 3776      DCA      3776        /WC
0605 7470      DCA      -310        /GET DRIVE SELECT WORD;
0606 4472      JMS I   SELALL      /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY.
0607 1066      TAD      ODD800      /LOAD COMMAND.
0610 6705      LCMR      K4100
0611 1113      TAD     
0612 6706      LFGR      /LOAD FUNCTION; WRITE, GO BITS;
0613 4457      TIMER      /TIME "WRITE START,"
0614 6713      RCAR      /EXIT TIMER ON
0615 7710      SPA CLA      /CA=4000,
0616 6724      SKTR
0617 5216      JMP      .-1        /WAIT FOR TUR.
0620 6725      CLF
0621 5203      JMP      RET3        /DO AGAIN ON NEXT DRIVE.

0622 4474      OUT3,  JMS I   XMESS      /MESSAGE ROUTINE
0623 6134      MSGT3
0624 1060      TAD      CNTALL
0625 3104      DCA      PRNCT
0626 4475      JMS I   XPRINT

//TIME WRITE NONSTOP GAP//
//
0627 7300      T4,   CLA CLL
0630 1376      TAD      (OUT4      /SETUP T4 EXIT LOCATION.
0631 3071      DCA      OUT          /INITIALIZE WC + CA LOCATIONS,
0632 4465      RET4,  JMS I   XWCCA      /CA
0633 3776      DCA      3776        /WC
0634 7470      DCA      -310        /GET DRIVE SELECT WORD;
0635 4472      JMS I   SELALL      /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY.
0636 1066      TAD      ODD800      /LOAD COMMAND.
0637 6705      LCMR      K4100
0640 1113      TAD     
0641 6706      LFGR      /LOAD FUNCTION; WRITE, GO BITS;
0642 6723      SKJD
0643 5242      JMP      .-1        /WAIT FOR JOB DONE.
0644 6725      CLF

```

```

0645 4465      JMS I   XWCCA      /RE-INITIALIZE WC + CA LOCATIONS;
0646 3776      DCA      3776        /CA
0647 7470      DCA      -310        /WC
0650 1375      TAD      (3          /RESET TIME COUNTER WITH
0651 3505      DCA I   NULL        /DURATION OF THIS SETUP PROCEDURE;
0652 1113      TAD      K4100
0653 6706      LFGR      /LOAD FUNCTION; WRITE, GO BITS;
0654 4457      TIMER      /TIME "NONSTOP GAP,"
0655 6713      RCAR      /EXIT TIMER ON
0656 7710      SPA CLA      /CA=4000,
0657 6723      SKJD
0660 5257      JMP      .-1        /WAIT FOR JOB DONE.
0661 6725      CLF
0662 5232      JMP      RET4        /MESSAGE ROUTINE.
0663 4474      OUT4,  JMS I   XMESS
0664 6145      MSGT4
0665 1060      TAD      CNTALL
0666 3104      DCA      PRNCT
0667 4475      JMS I   XPRINT

//TIME BACKSPACE SHUTDOWN + SETTLE DOWN DELAY//
//
0670 7300      T5,   CLA CLL
0671 1374      TAD      (OUT5      /SETUP T5 EXIT LOCATION.
0672 3071      DCA      OUT          /INITIALIZE WC + CA LOCATIONS,
0673 4472      RET5,  JMS I   SELALL      /GET DRIVE SELECT WORD;
0674 1066      TAD      ODD800      /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY.
0675 6705      LCMR      /LOAD COMMAND.
0676 1114      TAD      K7777
0677 6701      LHCR      /LOAD WC = 0.
0678 1115      TAD      K7100
0681 6706      LFGR      /LOAD FUNCTION; SPACE REVERSE, GO BITS;
0682 6723      SKJD
0683 5302      JMP      .-1        /WAIT FOR JOB DONE.
0684 4457      TIMER      /TIME "BACKSPACE SHUTDOWN + SETTLE DOWN DELAY",
0685 6724      SKTR      /EXIT TIMER ON TUR.
0686 7610      SKP CLA      /USED BY TIMER;
0687 6725      CLF
0688 5273      JMP      RET5        /DO AGAIN ON NEXT DRIVE.
0689 4474      OUT5,  JMS I   XMESS      /MESSAGE ROUTINE.
0692 6196      MSGT5
0693 1060      TAD      CNTALL
0694 3104      DCA      PRNCT
0695 4475      JMS I   XPRINT

//TIME READ SHUTDOWN//
//
0716 7300      T6,   CLA CLL
0717 1373      TAD      (OUT6      /SETUP T6 EXIT LOCATION.
0720 3071      DCA      OUT          /INITIALIZE WC + CA LOCATIONS,
0721 4465      RET6,  JMS I   XWCCA      /CA
0722 3777      DCA      3777        /WC
0723 7470      DCA      -310        /GET DRIVE SELECT WORD;
0724 4472      JMS I   SELALL

```

```

0725 1066 TAD 000888 /COMBINE DRIVE SELECT, 888 BPI, ODD PARITY,
0726 6705 LCMR /LOAD COMMAND.
0727 1116 TAD K2100
0730 6706 LFGR /LOAD FUNCTION; READ, 60 BITS,
0731 6723 SKJD
0732 5331 JMP ,=1 /WAIT FOR JOB DONE;
0733 6725 CLF /CLEAR FLAGS
0734 4772 JMS I (WRTCNT /WRITE CONTINUOUS
0735 5321 JMP RET6 /DO NEXT DRIVE
0736 4474 OUT9, JMS I XMESS /MESSAGE ROUTINE.
0737 6167 MSGT6
0740 1060 TAD CNTALL
0741 3104 DCA PRNCT
0742 4475 JMS I XPRINT
0743 5744 JMP I LINK7
0744 1000 LINK7, T7

0772 5400
0773 0736
0774 0711
0775 0003
0776 0663
0777 0622
1000

```

PAGE

//TIME WRITE XIRG//

//

```

1000 7300 T7, CLA CLL /SETUP T7 EXIT LOCATION.
1001 1377 TAD (OUT7
1002 3071 DCA OUT
1003 4465 JMS I XWCCA /INITIALISE WC + CA LOCATIONS.
1004 3776 3776 /CA
1005 7470 =310 /WC
1006 4472 JMS I SELALL /GET DRIVE SELECT WORD;
1007 1066 TAD 000888 /COMBINE DRIVE SELECT, 888 BPI, ODD PARITY,
1010 6705 LCMR /LOAD COMMAND.
1011 1117 TAD K4900
1012 6706 LFGR /LOAD FUNCTION; WRITE XIRG, 60 BITS.
1013 4497 TIMER /TIME "WRITE XIRG".
1014 6713 RCAR /EXIT TIMER ON
1015 7710 SPA CLA /CA=4000.
1016 6724 SKTR
1017 5216 JMP ,=1 /WAIT FOR PUR.
1020 6725 CLF
1021 5203 JMS I RET7 /DO AGAIN ON NEXT DRIVE.
1022 4474 OUT7, JMS I XMESS /MESSAGE ROUTINE
1023 6200 MSGT7
1024 1060 TAD CNTALL
1025 3104 DCA PRNCT
1026 4475 JMS I XPRINT

```

//TIME LAST CHARACTER TO CU READY//

//

```

1027 7300 T8, CLA CLL /SETUP T8 EXIT LOCATION.
1030 1376 TAD (OUT8

```

```

1031 3071 RET8, DCA OUT /GET DRIVE SELECT WORD;
1032 4472 JMS I SELALL /COMBINE DRIVE SELECT, 888 BPI, ODD PARITY,
1033 1066 TAD 000888 /SAVE THEM;
1034 3106 DCA TEMP1
1035 1106 TAD TEMP1
1036 6705 LCMR /LOAD COMMAND.
1037 1114 TAD K7777
1040 6701 LCMR /LOAD WC = -1.
1041 1115 TAD K7100
1042 6706 LFGR /LOAD FUNCTION; SPACE REVERSE, GO;
1043 6723 SKJD
1044 5243 JMP ,=1 /WAIT FOR JOB DONE.
1045 6725 CLF
1046 4465 JMS I XWCCA /INITIALISE WC & CA LOCATIONS.
1047 3776 3776 /CA
1050 7470 =310 /WC
1051 1106 TAD TEMP1 /GET DRIVE, 888 BPI, ODD PARITY.
1052 6705 LCMR /LOAD COMMAND.
1053 1116 TAD K2100
1054 6706 LFGR /LOAD FUNCTION; READ, 60.
1055 6711 RCAR /READ WC.
1056 7640 SPA CLA /WC = 0T
1057 5235 JMP ,=2 /NO, WAIT FOR IT.
1060 4487 TIMER /YES, TIME "LAST CHARACTER TO CU READY;"
1061 6722 SKGB /EXIT TIMER ON CU READY.
1062 7610 SKP CLA
1063 6724 SKTR
1064 5263 JMP ,=1 /WAIT FOR PUR.
1065 6725 CLF
1066 5232 JMS I RET8 /DO AGAIN ON NEXT DRIVE.
1067 4474 OUT9, JMS I XMESS /MESSAGE ROUTINE
1070 6211 MSGT8
1071 1060 TAD CNTALL
1072 3104 DCA PRNCT
1073 4475 JMS I XPRINT

```

//TIME READ FROM BOT//

//

```

1074 7300 T9, CLA CLL /SETUP T9 EXIT LOCATION.
1075 1375 TAD (OUT9
1076 3071 DCA OUT
1077 4463 JMS I XRWND /REWIND ALL SELECTED DRIVES TO BOT;
1078 4465 JMS I XWCCA /INITIALISE WC & CA LOCATIONS.
1081 3777 3777 /CA
1082 7470 =310 /WC
1083 4472 JMS I SELALL /GET DRIVE SELECT WORD;
1084 1066 TAD 000888 /COMBINE DRIVE SELECT, 888 BPI, ODD PARITY,
1085 6705 LCMR /LOAD COMMAND.
1086 1116 TAD K2100
1087 6706 LFGR /LOAD FUNCTION; READ, 60 BITS.
1088 4497 TIMER /TIME "READ FROM BOT."
1089 6713 RCAR /EXIT TIMER ON CA=4000;
1092 7710 SPA CLA
1093 6724 SKTR
1094 5313 JMP ,=1 /WAIT FOR PUR

```

```

1115 6725      CLF
1116 5300      JMP RET9     /DO AGAIN ON NEXT DRIVE
1117 4474      OUT9, JMS I XMESS /MESSAGE ROUTINE.
1120 6222      MSGT9
1121 1060      TAD CNTALL
1122 3104      DCA PRNCT
1123 4475      JMS I XPRINT

//TIME SPACE SHUTDOWN + SETTLE DOWN DELAY//
//
1124 7300      T10,  CLA CLL (OUT10 /SETUP T10 EXIT LOCATION,
1125 1374      TAD OUT
1126 3071      DCA OUT
1127 1114      RET10, TAD K7777
1130 6701      LWGR /LOAD WC = 1.
1131 4472      JMS I SELALL /SET DRIVE SELECT WORD;
1132 1066      TAD ODD0000 /COMBINE DRIVE SELECT, 000 BPI, ODD PARITY.
1133 6705      LCMR /LOAD COMMAND
1134 1120      TAD K6100
1135 6706      LFGR /LOAD FUNCTION; SPACE FORWARD, GO BITS,
1136 6723      SKJD
1137 5336      JMP .=-1 /WAIT FOR JOB DONE;
1140 4457      TIMER /TIME "SPACE SHUTDOWN + SETTLEDOWN DELAY,"
1141 6724      SKTR /EXIT TIMER ON TUR.
1142 7610      SKP CLA
1143 6725      CLF
1144 5327      JMP RET10 /DO AGAIN ON NEXT DRIVE.
1145 4474      OUT10, JMS I XMESS /MESSAGE ROUTINE.
1146 6233      MSGT10
1147 1060      TAD CNTALL
1148 3104      DCA PRNCT
1149 4475      JMS I XPRINT
1152 5753      JMP I LINK11
1153 1200      LINK11, T11

1174 1145
1175 1117
1176 1067
1177 1022
1200      PAGE

```

//TIME WRITE EOF//

```

1200 7300      T11,  CLA CLL (OUT11 /SETUP T6 EXIT LOCATION,
1201 1377      TAD (OUT11
1202 3071      DCA OUT
1203 4463      JMS I XRWND /REWIND ALL SELECTED DRIVES TO BOT;
1204 4465      RET11, JMS I XWCCA /INITIALIZE WC + CA LOCATIONS,
1205 3776      3776 /CA
1206 0000      0000 /WC
1207 4472      JMS I SELALL /SET DRIVE SELECT WORD;
1210 1066      TAD ODD0000 /COMBINE DRIVE SELECT, 000 BPI, ODD PARITY.
1211 3106      DCA TEMP1
1212 1106      TAD TEMP1
1213 6705      LCMR /LOAD COMMAND.

```

```

1214 1113      TAD K4100
1215 6706      LFGR /LOAD FUNCTION; WRITE, GO BITS.
1216 6723      SKJD
1217 5216      JMP .=-1 /WAIT FOR JOB DONE;
1220 6725      CLF
1221 1121      TAD K5100
1222 6706      LFGR /LOAD FUNCTION; WRITE EOF, GO BITS;
1223 4497      TIMER /TIME "WRITE EOF",
1224 6723      SKJD
1225 7610      SKP CLA
1226 6725      CLF
1227 5204      JMP RET11 /DO AGAIN ON NEXT DRIVE.
1228 4474      OUT11, JMS I XMESS /MESSAGE ROUTINE.
1231 6244      MSGT11
1232 1060      TAD CNTALL
1233 3104      DCA PRNCT
1234 4475      JMS I XPRINT

```

//TIME EOR TO EOF SPACE TIME//

```

1235 7300      T12,  CLA CLL (OUT12 /SETUP T7 EXIT LOCATION
1236 1376      TAD (OUT12
1237 3071      DCA OUT
1240 4463      RET12, JMS I XRWND /REWIND ALL SELECTED DRIVES TO BOT;
1241 4465      JMS I XWCCA /INITIALIZE WC + CA LOCATIONS,
1242 3776      3776 /CA
1243 0000      0000 /WC
1244 4472      JMS I SELALL /SET DRIVE SELECT WORD;
1245 1066      TAD ODD0000 /COMBINE DRIVE SELECT, 000 BPI, ODD PARITY.
1246 3106      DCA TEMP1
1247 1106      TAD TEMP1
1250 6705      LCMR /LOAD COMMAND.
1251 1113      TAD K4100
1252 6706      LFGR /LOAD FUNCTION; WRITE, GO BITS;
1253 6723      SKJD
1254 5253      JMP .=-1 /WAIT FOR JOB DONE;
1255 6725      CLF
1256 1106      TAD TEMP1
1257 6705      LCMR
1260 1122      TAD K1100
1261 6706      LFGR /LOAD FUNCTION, REWIND, GO BITS.
1262 6723      SKJD
1263 5262      JMP .=-1
1264 6725      CLF
1265 1106      TAD TEMP1
1266 6705      LCMR /LOAD COMMAND.
1267 1123      TAD K7776 /-2
1270 6701      LWGR /LOAD WC.
1271 1120      TAD K6100
1272 6706      LFGR /LOAD FUNCTION; SPACE FORWARD, GO BITS;
1273 6711      RWGR
1274 1124      TAD K0001
1275 7640      SEA CLA /CHECK FOR WC=1
/INDICATING FIRST RECORD HAS BEEN SPACED OVER.

```

```

1276 5273      JMP      ,=3
1277 4487      TIMER
1300 6723      SKJD
1301 7610      SKP CLA
1302 6714      RM8R
1303 8125      AND      K0100
1304 7650      SNA CLA
1305 5317      JMP      ERR12
1306 6725      CLF
1307 5241      JMP      RET12
                /TIME "EOR TO EOF SPACE TIME",
                /EXIT TIMER ON JOB DONE.

1310 4474      OUT12, JMS I  XMESS
1311 6255      MSGT12
1312 1060      TAD
1313 3104      DCA  CNTALL
1314 4475      JMS I  XPRINT
1315 5716      JMP I  LINK13
                /CHECK EOF BIT.
                /EOF = 1?
                /YES, SAVE EOF TIME.
                /NO, ZERO TIME COUNTER.
                /GO AGAIN ON NEXT DRIVE.
                /MESSAGE ROUTINE.

1316 1400      LINK13, TAD
1317 1010      ERR12, TAD  10
1320 1375      TAD  (-2
1321 3010      DCA  10
1322 3410      DCA I 10
1323 3410      DCA I 10
1324 6725      CLF
1325 5241      JMP      RET12

1375 7776
1376 1310
1377 1230
                PAGE
                1400

```

```

//TIME WRITE TO ERASE HEAD//
//
1400 7300      T13,  CLA CLL
1401 1377      TAD  (OUT13
1402 3071      DCA  OUT
                /SETUP TO EXIT LOCATION.
1403 4463      JMS I  XRWND
1404 4465      RET13, JMS I  XWCCA
                /REWIND ALL SELECTED DRIVES TO BOT;
                /INITIALIZE WC & CA LOCATIONS.
1405 3775      3775
                /CA
                /WC
1406 7775      =3
                /SET DRIVE SELECT WORD;
1407 4472      JMS I  SELALL
                /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY;
1410 1066      TAD  000000
                /SAVE IT.
1411 3106      DCA  TEMP1
1412 1106      TAD  TEMP1
                /LOAD COMMAND.
1413 6705      LCMR
                /LOAD COMMAND.
1414 1113      TAD  K4100
                /LOAD FUNCTION; WRITE, GO BITS.
1415 6706      LFGR
                /CA = 3777 INDICATES 2ND WORD SENT.
1416 6713      RCAR
1417 1126      TAD  K4001
                /2ND WORD SENT?
1418 7640      SEA CLA
                /NO, GO AGAIN.
1421 5216      JMP      ,=3
                /YES, ISSUE CLEAR ALL.
1422 6712      CLT
1423 1106      TAD  TEMP1

```

```

1424 6705      LCMR
                /LOAD COMMAND
1425 1122      TAD  K1100
                /LOAD FUNCTION; REWIND, GO BITS.
1426 6706      LFGR
                /CHECK FOR BOT;
1427 6714      RM8R
                /CHECK FOR BOT;
1430 8127      AND      K1000
1431 7650      SNA CLA
1432 5227      JMP      ,=5
1433 6725      CLF
                /INITIALIZE WC & CA LOCATIONS
1434 4465      JMS I  XWCCA
                /CA
1435 3776      3776
                /WC
1436 7775      =3
                /LOAD COMMAND.
1437 1106      TAD  TEMP1
                /LOAD COMMAND.
1440 6705      LCMR
                /LOAD COMMAND.
1441 1116      TAD  K2100
                /LOAD FUNCTION; READ, GO BITS.
1442 6706      LFGR
                /CA=3777 INDICATES 2ND WORD READ
1443 6713      RCAR
                /2ND WORD READ?
1444 1126      TAD  K4001
                /NO, WAIT;
1445 7640      SEA CLA
                /YES, TIME "WRITE TO ERASE HEAD."
1446 5243      JMP      ,=3
                /EXIT TIMER ON CA=4000;
1447 4487      TIMER
1450 6713      RCAR
1451 7710      SPA CLA
1452 6723      SKJD
1453 5252      JMP      ,=1
1454 6725      CLF
                /GO AGAIN ON NEXT DRIVE.
1455 5204      JMS I  RET13
                /MESSAGE ROUTINE.
1456 4474      OUT13, JMS I  XMESS
1457 6266      MSGT13
1460 1060      TAD  CNTALL
1461 3104      DCA  PRNCT
1462 4475      JMS I  XPRINT

```

```

//ONE INCH DATA TIME//
//
1463 7300      T14,  CLA CLL
1464 1376      TAD  (OUT14
1465 3071      DCA  OUT
                /SETUP TO EXIT LOCATION.
1466 4463      JMS I  XRWND
1467 4465      RET14, JMS I  XWCCA
                /REWIND ALL SELECTED DRIVES TO BOT;
                /INITIALIZE WC & CA LOCATIONS.
1470 3776      3776
                /CA
1471 7160      =020
                /WC
1472 4472      JMS I  SELALL
                /SET DRIVE SELECT WORD;
1473 1066      TAD  000000
                /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY
1474 6705      LCMR
                /LOAD COMMAND.
1475 1113      TAD  K4100
                /LOAD FUNCTION; WRITE, GO BITS.
1476 6706      LFGR
                /MINUS CA INDICATES FIRST WORD TAKEN.
1477 6713      RCAR
1480 7700      SNA CLA
1481 5277      JMP      ,=2
                /TIME "800 BPI DATATIME".
1482 4487      TIMER
                /EXIT TIMER ON WC=0.
1483 6711      RHQR
1484 7650      SNA CLA
1485 6724      SKTR
1486 5305      JMP      ,=1
                /WAIT FOR ?UR.

```

```

1507 6725 CLF
1510 9267 JMS RET14 /DO AGAIN ON NEXT DRIVE.
1511 4474 OUT14, JMS I XMESS /MESSAGE ROUTINE.
1512 6277 MSET14
1513 1060 TAD CNTALL
1514 3104 DCA PRNCT
1515 4475 JMS I XPRINT
1516 5717 JMP I LINKGP
1517 1600 LINKGP, GAPS
    
```

```

1576 1511
1577 1456
1600 1600
    
```

PAGE

//CREATE GAPS TO BE TIMED IN GAP CONSISTENCY TEST//

```

//
1600 7300 GAPS, CLA CLL
1601 1377 TAD (OUTGAP /SETUP GAPS EXIT LOCATION,
1602 3071 DCA OUT /
1603 4463 JMS I XRWND /REWIND ALL SELECTED DRIVES TO BOT.
1604 1133 RETGAP, TAD KM2
1605 3306 DCA GAP2CT /GAP2 COUNT=-2;
1606 1133 TAD KM2 /GAP5 COUNT=-2;
1607 3307 DCA GAP5CT /GAP6 COUNT=-3;
1610 1134 TAD KM3 /GAP6 COUNT=-3;
1611 3310 DCA GAP6CT /GAP6 COUNT=-3;
1612 1102 TAD KM4 /GAP7 COUNT=-4;
1613 3311 DCA GAP7CT /GAP7 COUNT=-4;
1614 1313 TAD KMB
1615 3312 DCA GAP8CT /GAP8 COUNT=-5;
1616 4465 JMS I XNCCA /INITIALIZE WC, CA LOCATIONS.
1617 3767 JMS I XNCCA /CA
1620 7470 -310 /WC
1621 4492 JMS I SELALL /SET DRIVE SELECT WORD;
1622 1066 TAD ODD800 /COMBINE DRIVE SELECT, 800 BPI, ODD PARITY;
1623 3106 DCA TEMP1 /SAVE IT.
1624 1106 TAD TEMP1
1625 6705 LCMR /LOAD COMMAND.
1626 1113 TAD K4100
1627 6706 LFGR /LOAD FUNCTION; WRITE, GO BITS,
1630 6723 SKJD
1631 5230 JMP
1632 6725 DOGAP2, CLF .=-1 /WAIT FOR JOB DONE;
1633 4465 JMS I XNCCA /WRITE NONSTOP, GAP1 CREATED.
1634 3767 JMS I XNCCA /INITIALIZE WC, CA LOCATIONS.
1635 7470 -310 /CA
1636 1106 TAD TEMP1 /PICKUP DRIVE, BPI, ODD PARITY.
1637 6705 LCMR /LOAD COMMAND.
1640 1113 TAD K4100
1641 6706 LFGR /LOAD FUNCTION; WRITE GO BITS,
1642 6723 SKJD /WAIT FOR JOB DONE;
1643 5242 JMP .=-1
1644 6724 SKTR /WAIT FOR PUR.
1645 5244 JMP .=-1
    
```

```

1646 2306 ISZ GAP2CT /WRITE 3RD RECORD START-STOP
1647 5232 JMP DOGAP2 /TO CREATE GAP2.
1650 6725 CLF
1651 4536 JMS I XBAK /SPACE REVERSE 1 RECORD.
1652 4540 JMS I XREAD /READ RECORD.
1653 4537 JMS I XRITE1 /WRITE 4TH RECORD; 3RD GAP CREATED.
1654 4937 JMS I XRITE1 /START GAP4, WRITE 1 RECORD.
1655 4536 JMS I XBAK /SPACE REVERSE OVER RECORD JUST WRITTEN.
1656 4937 JMS I XRITE1 /WRITE 5TH RECORD; 4TH GAP CREATED.
1657 4537 JMS I XRITE1 /START GAP5, WRITE 1 RECORD.
1660 4936 DOGAP5, JMS I XBAK /SPACE REVERSE OVER RECORD JUST WRITTEN.
1661 4937 JMS I XRITE1 /WRITE 1 RECORD
1662 2307 ISZ GAP5CT /DO XBAK + XRITE 1 MORE TIME
1663 5240 JMP DOGAP5 /TO CREATE 5TH GAP.
1664 4937 JMS I XRITE1 /START GAP6, WRITE 1 RECORD.
1665 4936 DOGAP6, JMS I XBAK /SPACE REVERSE OVER RECORD JUST WRITTEN.
1666 4937 JMS I XRITE1 /WRITE 1 RECORD.
1667 2310 ISZ GAP6CT /DO XBAK + XRITE 2 MORE TIMES
1670 5245 JMP DOGAP6 /TO CREATE 6TH GAP.
1671 4937 JMS I XRITE1 /START GAP7, WRITE 1 RECORD.
1672 4936 DOGAP7, JMS I XBAK /SPACE REVERSE OVER RECORD JUST WRITTEN.
1673 4937 JMS I XRITE1 /WRITE 1 RECORD.
1674 2311 ISZ GAP7CT /DO XBAK + XRITE 3 MORE TIMES.
1675 5272 JMP DOGAP7 /TO CREATE 7TH GAP.
1676 4937 JMS I XRITE1 /START GAP8, WRITE 1 RECORD.
1677 4936 DOGAP8, JMS I XBAK /SPACE REVERSE OVER RECORD JUST WRITTEN.
1678 4937 JMS I XRITE1 /WRITE 1 RECORD
1681 2312 ISZ GAP8CT /DO XBAK + XRITE 4 MORE TIMES
1682 5277 JMP DOGAP8 /TO CREATE 8TH GAP.
1683 5284 OUTGAP, JMP I LINK15 /DO AGAIN ON NEXT DRIVE.
1684 5705 LINK15, T15
1685 2000 LINK15, T15
1706 0000 GAP2CT, 0
1707 0000 GAP5CT, 0
1710 0000 GAP6CT, 0
1711 0000 GAP7CT, 0
1712 0000 GAP8CT, 0
1713 7773 KMB, -5
    
```

```

1777 1704
1800 2000
    
```

PAGE

//NOW READ NONSTOP THE GAPS JUST CREATED  
//TYPE GAP TIME AS EACH GAP IS READ ON ALL DRIVES  
//GAP1 > GAP2, GAP2 APPROXIMATELY 8 GAPS  
//GAP4 THRU GAPS SHOULD GET INCREASINGLY LONGER

```

2000 7300 T15, CLA CLL
2001 1377 TAD (OUT15A /SETUP T15A EXIT LOCATION,
2002 3071 DCA OUT /
2003 1264 TAD M7
2004 3263 DCA GAPCNT
    
```



```

2005 1376 TAD (GAPTBL=1
2006 3012 DCA INDX12
2007 4463 JMS I XRWND
2010 4472 RET15A, JMS I SELALL
2011 1066 TAD ODB800
2012 3106 DCA TEMP1
2013 4846 JMS I XREAD2
2014 4541 JMS I XREAD1
2015 5210 JMP RET15A
2016 4250 OUT15A, JMS GAPOUT
2017 1375 TAD (OUT15B
2020 3071 DCA OUT
2021 4472 RET15B, JMS I SELALL
2022 1066 TAD ODB800
2023 3106 DCA TEMP1
2024 4536 JMS I XBAK
2025 4546 JMS I XREAD2
2026 4541 JMS I XREAD1
2027 5221 JMP RET15B
2030 4250 OUT15B, JMS GAPOUT
2031 2263 ISR GPCNT
2032 5221 JMP RET15B
2033 1061 TAD CNT7
2034 7050 SNA CLA
2035 5545 JMP I XEND
2036 5637 JMP I LINK56
2037 2200 LINK56, SET556

2040 0261 GAPTBL, 0261
2041 0262 0262
2042 0263 0263
2043 0264 0264
2044 0265 0265
2045 0266 0266
2046 0267 0267
2047 0270 0270

2050 0000 GAPOUT, 0
2051 4474 JMS I XMESS
2052 6310 MSGT15
2053 1412 TAD I INDX12
2054 4501 JMS I XOTY
2055 4474 JMS I XMESS
2056 6313 MSGSPC
2057 1060 TAD CNTALL
2060 3104 DCA PRNCT
2061 4475 JMS I XPRINT
2062 5650 JMP I GAPOUT
2063 0000 GPCNT, 0
2064 7771 M7, -7

2175 2030
2176 2037
2177 2016
2200
    
```

PAGE

```

//ROUTINE TO RUN TESTS AT 556 BPI//
2200 7300 SET556, CLA CLL
2201 4474 JMS I XMESS
2202 6321 MSG556
2203 4456 JMS I XHEAD1
2204 4451 JMS I XCRLF
2205 1142 TAD K0400
2206 7001 IAC
2207 3143 DCA XBP1
2210 1213 TAD M426
2211 3147 DCA INCH
2212 5544 JMP I LINK16

2213 7352 M426, -426

//TIME WRITE FROM BOT DELAY AT EITHER 556 OR 200 BPI//
//
2214 7300 T16, CLA CLL
2215 1377 TAD (OUT16
2216 3071 DCA OUT
2217 4464 JMS I XRWND7
2220 4465 RET16, JMS I XWCCA
2221 3776 JMS I XWCCA
2222 7470 -310
2223 4473 JMS I SELTU7
2224 1143 TAD XBP1
2225 6705 LCMR
2226 1113 TAD K4100
2227 6706 LFGP
2230 4457 TIMER
2231 6713 RCAR
2232 7710 SPA CLA
2233 6724 SKTR
2234 5233 JMP .=-1
2235 6725 CLP
2236 5220 JMP RET16
2237 4474 OUT16, JMS I XMESS
2240 6112 MSGT1
2241 1061 TAD CNT7
2242 3104 DCA PRNCT
2243 4475 JMS I XPRINT

//TIME WRITE SHUTDOWN AT
//EITHER 556 OR 200 BPI
//
2244 7300 T17, CLA CLL
2245 1376 TAD (OUT17
2246 3071 DCA OUT
2247 4464 JMS I XRWND7
2250 4465 RET17, JMS I XWCCA
2251 3776 JMS I XWCCA
2252 7470 -310
2253 4473 JMS I SELTU7
    
```

```

0047 1111      TAQ      NCRU      /SUBTRACT MIKECT FROM ORIGINAL CONSTANT.
0050 7040      CMA
0051 1112      TAQ      PCBU      /NOW ADD ORIGINAL CONSTANT TO NUMBER OF
0052 3236      DCA      DIVDND    /ITERATIONS IN MIKECT TO FORM DIVIDEND.
0053 1112      TAQ      PCBU      /NOW SETUP
0054 3237      DCA      DIVSR    /DIVISOR.
0055 4274      JMS      DIVIDE    /ENTER DIVIDE ROUTINE.

0056 0000      DIVDND, 0      /DIVIDEND.
0057 0000      DIVSR, 0      /DIVISOR.
0060 7200      DIVREI, 0      /RETURN HERE FROM DIVIDE ROUTINE.
0061 1103      TAQ      HDIVND   /GET ANSWER.
0062 3273      DCA      SAVANS    /SAVE IT.
0063 1110      TAQ      MHLICT   /GET VALUE IN MILLISECONDS.
0064 3410      DCA I   INBX10  /STORE AWAY.
0065 1273      TAQ      SAVANS    /GET VALUE IN MICRO SECONDS.
0066 3410      DCA I   INBX10  /STORE AWAY.
0067 3107      DCA      MIKECT   /CLEAR TIME COUNTERS.
0070 3110      DCA      MHLICT
0071 3905      DCA I   NULL
0072 5611      JMP I   TIME      /RETURN TO TEST.

0073 0000      SAVANS, 0

/SINGLE PRECISION DIVIDE SUBROUTINE
/

0074 0000      DIVIDE, 0
0075 7100      CLL
0076 3103      DCA      HDIVND   /HIGH ORDER DIVIDEND.
0077 1674      TAQ I   DIVIDE    /FETCH LOW ORDER DIVIDEND.
0080 3326      DCA      LDIVND   /SAVE IT.
0081 2274      ISR      DIVIDE    /UPDATE POINTER.
0082 1674      TAQ I   DIVIDE    /FETCH DIVISOR.
0083 3327      DCA      DIVSR    /SAVE IT.
0084 2274      ISR      DIVIDE
0085 1331      TAQ      KM13
0086 3330      DCA      DIVCNT
0087 5320      JMP      DV2
0088 1103      DVS,   TAQ      HDIVND   /DIVIDE LOOP
0089 7004      RAL
0090 3103      DCA      HDIVND   /DIVIDEND LEFT SHIFT.
0091 1103      TAQ      HDIVND
0092 1327      TAQ      DIVSR    /COMPARE DIVISOR/DIVIDEND.
0093 7430      SEL
0094 3103      DCA      HDIVND   /REMAINDER AFTER SUBTRACT.
0095 7200      CLA
0096 1326      DV2,   TAQ      LDIVND   /QUOTIENT BITS.
0097 7004      RAL      /ENTER HERE.
0098 3326      DCA      LDIVND
0099 2330      ISR      DIVCNT   /DONE 12?
0100 5310      JMP      DVS      /NO, CONTINUE.
0101 5674      JMP I   DIVIDE    /YES, EXIT.

0126 0000      LDIVND, 0
    
```

```

0127 0000      DIVSR, 0
0130 0000      DIVCNT, 0
0131 7765      KM13,  -13
0132 0000      XNULL,  0

0175 0142
0176 7036
0177 7037      PAGE
0200 0000      RWIND, 0      /ROUTINE TO REWIND ALL SELECTED TRANSPORTS TO BOT/
0201 6722      SKGB
0202 5201      JMP      .-1      /WAIT FOR BUR.
0203 6725      CLF
0204 1377      TAQ      (ALLTU   /SETUP DRIVE SELECTION
0205 3070      DCA      DRVSEL   /POINTER FOR ALL DRIVES.
0206 1060      TAQ      CNTALL   /SETUP COUNT
0207 7041      CIA
0210 3261      DCA      REWCNT   /TO INDICATE WHEN ALL
0211 1297      TAQ      EXALL    /DRIVES HAVE BEEN REWOUND.
0212 3236      DCA      GETBAK
0213 1066      TAQ      ODD000   /PICKUP PARITY AND BPI.
0214 3263      DCA      TEMPS   /SAVE THEM.
0215 5233      JMP      RWINDA   /GO AND REWIND SELECTED DRIVES.
0216 0000      RWIND, 0
0217 6722      SKGB
0220 5217      JMP      .-1      /WAIT FOR BUR.
0221 6725      CLF
0222 1376      TAQ      (TU7    /SETUP DRIVE SELECTION
0223 3070      DCA      DRVSEL   /POINTER FOR 7 TRACK DRIVES ONLY.
0224 1061      TAQ      CNT7    /SETUP COUNT TO INDICATE
0225 7041      CIA      /WHEN ALL 7 TRACK DRIVES
0226 3261      DCA      REWCNT   /HAVE BEEN REWOUND.
0227 1260      TAQ      EXTU7
0230 3236      DCA      GETBAK
0231 1143      TAQ      XB#1
0232 3263      DCA      TEMPS   /PICKUP PARITY AND BPI.
0233 1470      RWINDA, TAQ I   DRVSEL   /SAVE THEM.
0234 1263      TAQ      TEMPS   /GET DRIVE SELECTED.
0235 6705      LCMR      /COMBINE WITH PARITY AND BPI.
0236 6724      SKTR      /LOAD COMMAND.
0237 5236      JMP      .-1
0240 6714      RMR
0241 0242      AND      MK1000
0242 7040      SEA CLA
0243 5232      JMP      .+7
0244 1375      TAQ      (1100
0245 6706      LFR
0246 6723      SKJD
0247 6246      JMP      .-1
0250 6724      SKTR
0251 5230      JMP      .-1
0252 6725      CLF      /WAIT FOR BUR.
    
```

```

3253 2070     ISR   DRVSEL
3254 2261     ISR   REWCNT
3255 5233     JMP   RWINDA
3256 7402     GETBAK, HLT

3257 5600     EXALL, JMP I  RWIND
3260 5616     EXTU7, JMP I RWIND7
3261 0000     REWCNT, 0
3262 1000     MK1000, 1000
3263 0000     TEMP3, 0

/Routine used to select both 7 and 9 track drives/
/TO BE TESTED AT 800 BPI/
/
3264 0000     ALTU79, 0
3265 2050     ISR   ONCE
3266 5274     JMP   ,+6
3267 1377     TAD   (ALLTU
3270 3070     DCA   DRVSEL
3271 1060     TAD   CNTALL
3272 7040     CMA
3273 3324     DCA   SELCNT
3274 1470     TAD I  DRVSEL
3275 2070     ISR   DRVSEL
3276 2324     ISR   SELCNT
3277 5684     JMP I  ALTU79
3300 7200     CLA
3301 1114     TAD   K7777
3302 3050     DCA   ONCE
3303 5471     JMP I  OUT

/Routine used to select 7 track drives to be/
/TESTED AT 556 AND 200 BPI/
/
3304 0000     ALLTU7, 0
3305 2050     ISR   ONCE
3306 5314     JMP   ,+6
3307 1376     TAD   (TU7
3310 3070     DCA   DRVSEL
3311 1061     TAD   CNT7
3312 7040     CMA
3313 3324     DCA   SELCNT
3314 1470     TAD I  DRVSEL
3315 2070     ISR   DRVSEL
3316 2324     ISR   SELCNT
3317 5784     JMP I  ALLTU7
3320 7200     CLA
3321 1114     TAD   K7777
3322 3050     DCA   ONCE
3323 5471     JMP I  OUT

3324 0000     SELCNT, 0

```

```

/Routine used to output to TTY both 7 and 9 track/
/DRIVES TESTED AT 800 BPI/
/
3325 0000     ALLMSG, 0
3326 1374     TAD   (ALLTBL
3327 3070     DCA   DRVSEL
3330 1060     TAD   CNTALL
3331 7041     CIA
3332 3324     DCA   SELCNT
3333 4474     JMS I  XMESS
3334 6076     MSG0
3335 1470     TAD I  DRVSEL
3336 4501     JMS I  XOTY
3337 1365     TAD   SPACE
3340 4501     JMS I  XOTY
3341 2070     ISR   DRVSEL
3342 2324     ISR   SELCNT
3343 5333     JMP   ALLRET
3344 5725     JMP I  ALLMSG

/Routine used to output to TTY all 7 track drives/
/TESTED AT 556 AND 200 BPI/
/
3345 0000     TU7MSG, 0
3346 1373     TAD   (TU7TBL
3347 3070     DCA   DRVSEL
3350 1061     TAD   CNT7
3351 7041     CIA
3352 3324     DCA   SELCNT
3353 4474     JMS I  XMESS
3354 6076     MSG0
3355 1470     TAD I  DRVSEL
3356 4501     JMS I  XOTY
3357 1365     TAD   SPACE
3360 4501     JMS I  XOTY
3361 2070     ISR   DRVSEL
3362 2324     ISR   SELCNT
3363 5333     JMP   TU7RET
3364 5745     JMP I  TU7MSG

3365 0240     SPACE, 0240

3373 0170
3374 0160
3375 1100
3376 0040
3377 0030
3400 5400     PAGE

```

```

//ROUTINE TO WRITE CONTINUOUS AND TIME TILL LEADING EDGE OF 80WN.//
//THIS ROUTINE USED FOR TIMING WRITE AND READ SHUTDOWN.//
3400 0000     WRTCNT, 0

```

```

2401 1377 TAD (3 /ADJUST FOR SETUP TIME
2402 3505 DCA I NULL
2403 1113 TAD K4100 /CONTINUOUS WRITE
2404 6706 LFGP
2405 4457 TIMER /TIME TILL FIRST WORD XFERRED OUT OF
2406 6711 RGR /MEMORY TO IMBE (APPROX LEADING EDGE
2407 7640 SEA CLA /OF SOWN;
2410 6723 SKJD /WAIT JOB DONE
2411 5210 JMP .=1
2412 6724 SKIR /WAIT TUR
2413 5212 JMP .=1
2414 6725 CLF
2415 5000 JMP I WRTCNT /CLEAN ALL
/EXIT
    
```

/SINGLE CHARACTER OUTPUT ROUTINE  
/ENTERED WITH CHARACTER IN AC

```

2416 0000 OTY, 0
2417 6046 TLF
2420 6041 TSF
2421 5220 JMP .=1
2422 6042 TCF
2423 7200 CLA
2424 5616 JMP I OTY
    
```

2425 0000 PRINT, 0 /ROUTINE TO OUTPUT ACCUMULATED  
2426 7300 CLA CLL /TIMES FOR ALL DRIVES,

```

2427 1376 TAD (6377
2430 3010 DCA INDX10
2431 1104 TAD PRNCT /SETUP PRINT COUNTER;
2432 7041 CIA
2433 3104 DCA PRNCT
2434 3077 LOPRIN, DCA VFS20
2435 3100 DCA VFSUPR
2436 1410 TAD I INDX10
2437 4476 JMS I XDEC
2440 1277 TAD I K256 /DECIMAL POINT;
2441 4501 JMS I XOTY
2442 1300 TAD I K7775
2443 3077 DCA VFS20
2444 7040 CMA
2445 3100 DCA VFSUPR
2446 1410 TAD I INDX10
2447 4476 JMS I XDEC
2450 1276 TAD I K240 /SPACE;
2451 4501 JMS I XOTY
2452 2104 ISE PRNCT
2453 5234 JMP LOPRIN
2454 4260 JMS CRLF
2455 1376 TAD (6377
2456 3010 DCA INDX10
2457 5625 JMP I PRINT
    
```

```

2460 0000 CRLF, 0
2461 1131 TAD K215 /CARRIAGE RETURN;
2462 6046 TLF
2463 6041 TSF
2464 5263 JMP .=1
2465 6042 TCF
2466 7200 CLA
2467 1130 TAD K212 /LINE FEED;
2470 6046 TLF
2471 6041 TSF
2472 5271 JMP .=1
2473 6042 TCF
2474 7200 CLA
2475 5660 JMP I CRLF
    
```

```

2476 0240 K240, 0240
2477 0256 K250, 0256
2500 7775 K7775, 7775
    
```

/UNSIGNED DECIMAL PRINT  
/CALL WITH NUMBER TO BE TYPED IN AC  
/RETURN TO LOCATION FOLLOWING THE JMS.

```

2501 0000 DECPRT, 0
2502 3363 DCA VALUE /SAVE INPUT;
2503 3364 DCA DIGIT /CLEAR;
2504 3366 DCA VTEMP2
2505 1102 TAD KM4
2506 3365 DCA VTEMP1 /SET COUNTER TO 4;
2507 1396 TAD ADDBEA
2510 3315 DCA ARROW /SET TABLE POINTER;
2511 7410 SKP
2512 3363 DCA VALUE /SAVE
2513 7100 CLL
2514 1363 TAD VALUE
2515 1397 ARROW, TAD TENPHR /SUBTRACT POWER OF TEN;
2516 7430 SEL
2517 2364 ISE DIGIT /DEVELOP BCD DIGIT;
2520 7430 SEL
2521 5312 JMP ARROW-3 /LOOP
2522 7200 CLA /HAVE BCD DIGIT;
2523 1077 TAD VFS20
2524 7050 SNA CLA /SUPPRESS LEADING DIGITS?
2525 5330 JMP .=3 /NO
2526 2077 ISE VFS20 /YES, COUNT AND DO NOT PRINT.
2527 5347 JMP LOOPD+2
2530 1364 TAD DIGIT /SET DIGIT
2531 7490 SNA /IS THIS A ZERO?
2532 5335 JMP .=3 /YES
2533 2366 ISE VTEMP2 /NO
2534 5345 JMP LOOPD
2535 1366 TAD VTEMP2
2536 7640 SEA CLA /HAS A DIGIT BEEN TYPED?
2537 5345 JMP LOOPD /YES;
2540 1100 TAD VFSUPR /NO.
    
```

```

0541 7640          SZA CLA          /SHOULD THIS ONE BE SUPPRESSED?
0542 5345          JMP          LODPD          /NO,
0543 1276          TAO          K240          /YES, TYPE SPACE
0544 5346          JMP          LODPD*1
0545 1367          LOOPD, TAO          K260          /MAKE IT ASCII
0546 4901          JMS I          XOTY          /TYPE DIGIT.
0547 3364          DCA          DIGIT          /CLEAN
0550 2315          ISE          ARROW          /UPDATE POINTER,
0551 2365          ISE          VTEMP1          /DONE ALL FOUR?
0552 5314          JMP          ARROW-1          /NO, CONTINUE.
0553 3100          DCA          VFSUPR
0554 3077          DCA          VFS20
0555 5701          JMP I          DECPRT          /YES, EXIT.

0556 1357          ADDRZA, TAO          TENPWR
0557 6030          TENPWR, =1750          /ONE THOUSAND
0558 7634          =0144          /ONE HUNDRED
0559 7766          =0012          /TEN
0560 7777          =0001          /ONE

0563 0000          VALUE, 0
0564 0000          DIGIT, 0
0565 0000          VTEMP1, 0
0566 0000          VTEMP2, 0
0567 0260          K260, 260
    
```

```

PAGE
//ROUTINE TO WRITE ONE 200(10) WORD RECORD//
//
0600 0000          RITE1, 0
0601 4485          JMS I          XWCCA          /INITIALIZE WC, CA LOCATIONS.
0602 3777          3777          /CA
0603 7470          =310          /NC
0604 1106          TAO          TEMP1          /PICKUP DRIVE, BPI, ODD PARITY,
0605 6705          LCMR          /LOAD COMMAND.
0606 1113          TAO          K4100
0607 6706          LFGR          /LOAD FUNCTION/ WRITE, GO BITS,
0610 6723          SKJD
0611 5210          JMP          .-1          /WAIT FOR JOB DONE.
0612 6724          SKTR
0613 5212          JMP          .-1          /WAIT FOR PUR.
0614 6725          CLP
0615 5600          JMP I          RITE1          /RETURN.

//BACKSPACE ONE RECORD//
//
0616 0000          BAKSPA, 0
0617 1132          TAO          KM1
0620 6701          LCMR          /WC=-1
0621 1106          TAO          TEMP1          /SET DRIVE, BPI, ODD PARITY,
0622 6705          LCMR          /LOAD COMMAND.
0623 1115          TAO          K7100
    
```

```

0624 6706          LFGR          /LOAD FUNCTION/ SPACE REVERSE, GO,
0625 6723          SKJD
0626 5225          JMP          .-1          /WAIT FOR JOB DONE.
0627 6724          SKTR
0630 5227          JMP          .-1          /WAIT FOR JOB DONE.
0631 6725          CLP
0632 5616          JMP I          BAKSPA          /RETURN.

//ROUTINE TO READ 1 RECORD START=STOP//
//
0633 0000          READ, 0
0634 4485          JMS I          XWCCA          /INITIALIZE WC, CA LOCATIONS.
0635 3777          3777          /CA
0636 7470          =310          /NC
0637 1106          TAO          TEMP1          /PICKUP DRIVE, BPI, ODD PARITY,
0640 6705          LCMR          /LOAD COMMAND.
0641 1116          TAO          K2100
0642 6706          LFGR          /LOAD FUNCTION/ READ, GO BITS,
0643 6723          SKJD
0644 5243          JMP          .-1          /WAIT FOR JOB DONE.
0645 6724          SKTR
0646 5245          JMP          .-1          /WAIT FOR PUR.
0647 6725          CLP
0650 5633          JMP I          READ          /RETURN.
    
```

```

//ROUTINE TO READ 1 RECORD AND TIME THE INTERRECORD
//GAP FOLLOWING IT
//
0651 0000          READ1, 0
0652 4485          JMS I          XWCCA          /INITIALIZE WC, CA LOCATIONS.
0653 3777          3777          /CA
0654 7470          =310          /NC
0655 1106          TAO          TEMP1          /PICKUP DRIVE, BPI, ODD PARITY,
0656 6705          LCMR          /LOAD COMMAND.
0657 1116          TAO          K2100
0660 6706          LFGR          /LOAD FUNCTION/ READ, GO BITS,
0661 4497          TIMER          /TIME GAP,
0662 6713          RGR          /EXIT TIMER ON CA=4000.
0663 7710          SPA CLA
0664 6724          SKTR          /RETURN HERE.
0665 5264          JMP          .-1          /WAIT FOR PUR.
0666 6725          CLP
0667 5651          JMP I          READ1          /RETURN.

//ROUTINE TO READ 1 RECORD NONSTOP//
//
0670 0000          READ2, 0
0671 4485          JMS I          XWCCA          /INITIALIZE WC, CA LOCATIONS.
0672 3777          3777          /CA
0673 7470          =310          /NC
0674 1106          TAO          TEMP1          /PICKUP DRIVE, BPI, ODD PARITY,
0675 6705          LCMR          /LOAD COMMAND.
0676 1116          TAO          K2100
0677 6706          LFGR          /LOAD FUNCTION/ READ, GO BITS,
0678 6723          SKJD
    
```

```

9701 5300      JMP      ,=1      /WAIT FOR JOB DONE,
9702 6725      CLF
9703 5670      JMP I   READ2    /RETURN.

                //ENDING ROUTINE//
9704 7300      FINI,   CLA CLL
9705 4474      JMS I   XMESS
9706 6343      MSGEND
9707 5710      JMP I   ,+1
9710 0206      SELECT

/MESSAGE TYPE-OUT ROUTINE
/
9711 0000      MESSAGE, 0
9712 7240      CLA CMA
9713 1711      TAD I   MESSAGE
9714 3011      DCA     11
9715 2311      ISZ   MESSAGE
9716 1411      TAD I   11
9717 3330      DCA     MSRGHT
9720 1330      TAD     MSRGHT
9721 7012      RTR
9722 7012      RTR
9723 7012      RTR
9724 4331      JMS     TYPECH
9725 1330      TAD     MSRGHT
9726 4331      JMS     TYPECH
9727 5316      JMP     MESSAGE+5
9730 0000      MSRGHT, 0
9731 0000      TYPECH, 0
9732 0363      AND     MASK77
9733 7490      SNA
9734 5711      JMP I   MESSAGE
9735 1364      TAD     H40
9736 7510      SPA
9737 5342      JMP     ,+5
9740 1365      TAD     C240
9741 5355      JMP     MTP
9742 7001      IAC
9743 7440      SEA
9744 5347      JMP     ,+5
9745 1366      TAD     C215
9746 5355      JMP     MTP
9747 7001      IAC
9750 7440      SEA
9751 5354      JMP     ,+5
9752 1367      TAD     C212
9753 5355      JMP     MTP
9754 1370      TAD     C336
9755 6046      MTP,   TL8
9756 6041      TSE
9757 5356      JMP     ,=1
9760 6042      TCF
    
```

```

9761 7200      CLA
9762 5731      JMP I   TYPECH
9763 0077      MASK77, 77
9764 7740      H40,   =40
9765 0240      C240, 240
9766 0215      C215, 215
9767 0212      C212, 212
9770 0336      C336, 336

6000           PAGE

0000 3736      MSG1,  TEXT  "=>IMBE DRIVE FUNCTION TIMER=>"
0001 2415
0002 7005
0003 4004
0004 2211
0005 2605
0006 4006
0007 2516
0010 0324
0011 1117
0012 1640
0013 2411
0014 1905
0015 2237
0016 3600

0017 3736      MSG2,  TEXT  "=>MAINDEC=08-DHTMC=A=>"
0020 1901
0021 1116
0022 0405
0023 0355
0024 6070
0025 5504
0026 1024
0027 1503
0030 5501
0031 3736
0032 0000

0033 3736      MSG3,  TEXT  "=>SELECT DRIVES=>"
0034 2305
0035 1405
0036 0324
0037 4004
0040 2211
0041 2605
0042 2337
0043 3600

0044 3736      MSG4,  TEXT  "=>SELECT 7 AND/OR 9 TRACK OPERATION=>"
0045 2305
0046 1405
0047 0324
    
```

0050	4067			
0051	4001			
0052	1604			
0053	5717			
0054	2240			
0055	7140			
0056	2422			
0057	0103			
0060	1340			
0061	1720			
0062	0922			
0063	0124			
0064	1117			
0065	1637			
0066	3600			
0067	3736	MSG5, TEXT	"FUNCTION"	
0070	0625			
0071	1603			
0072	2411			
0073	1716			
0074	3736			
0075	0000			
0076	4040	MSG6, TEXT	" UNIT"	
0077	2514			
0100	1124			
0101	0000			
0102	7060	MSG7, TEXT	"800 BPI "	
0103	4040			
0104	0220			
0105	1140			
0106	4040			
0107	4040			
0110	4040			
0111	4000			
0112	2722	MSG11, TEXT	"NR FM BOT DELAY "	
0113	4000			
0114	1940			
0115	0217			
0116	2440			
0117	0405			
0120	1401			
0121	3140			
0122	0000			
0123	2722	MSG12, TEXT	"WRITE SHUTDOWN "	
0124	1124			
0125	0540			
0126	2310			
0127	2524			
0130	0417			
0131	2716			

0132	4040			
0133	0000			
0134	2722	MSG13, TEXT	"WRITE START "	
0135	1124			
0136	0540			
0137	2324			
0140	0122			
0141	2440			
0142	4040			
0143	4040			
0144	0000			
0145	2722	MSG14, TEXT	"NR NONSTOP GAP "	
0146	4016			
0147	1716			
0150	2324			
0151	1720			
0152	4007			
0153	0120			
0154	4040			
0155	0000			
0156	0213	MSG15, TEXT	"BRSP SHOWN+SDWN "	
0157	2320			
0160	4023			
0161	1004			
0162	2716			
0163	5323			
0164	0427			
0165	1640			
0166	0000			
0167	2205	MSG16, TEXT	"READ SHUTDOWN "	
0170	0104			
0171	4023			
0172	1025			
0173	2404			
0174	1727			
0175	1640			
0176	4040			
0177	0000			
0200	2722	MSG17, TEXT	"WRITE XIRG "	
0201	1124			
0202	0540			
0203	3011			
0204	2207			
0205	4040			
0206	4040			
0207	4040			
0210	0000			
0211	1401	MSG18, TEXT	"LAST CHR TO CUR "	
0212	2324			

```

0213 4003
0214 1022
0215 4024
0216 1740
0217 0325
0220 2240
0221 0000

0222 2204 MSGT9, TEXT "RD FM BOT DELAY "
0223 4006
0224 1540
0225 0217
0226 2440
0227 0405
0230 1401
0231 3140
0232 0000

0233 2320 MSGT10, TEXT "SPCE SHDWN+SDWN "
0234 0305
0235 4023
0236 1004
0237 2716
0240 5323
0241 0427
0242 1640
0243 0000

0244 2722 MSGT11, TEXT "WRITE EOP "
0245 1124
0246 0540
0247 0517
0250 0640
0251 4040
0252 4040
0253 4040
0254 0000

0255 0522 MSGT12, TEXT "ER TO EF SP TME "
0256 4024
0257 1740
0260 0506
0261 4023
0262 2040
0263 2415
0264 0540
0265 0000

0266 2722 MSGT13, TEXT "WR TO ERASE HED "
0267 4024
0270 1740
0271 0522
0272 0123
0273 0540
0274 1005

```

```

0275 0440
0276 0000

0277 6140 MSGT14, TEXT "1 INCH DATA TME "
0300 1116
0301 0310
0302 4004
0303 0124
0304 0140
0305 2415
0306 0540
0307 0000
0310 0701 MSGT15, TEXT "GAP "
0311 2040
0312 0000 MSGSPG, TEXT " "
0313 4040
0314 4040
0315 4040
0316 4040
0317 4040
0320 4000
0321 3736 MSG556, TEXT "++556 BPI "
0322 6565
0323 6640
0324 0220
0325 1140
0326 4040
0327 4040
0330 4040
0331 4000
0332 3736 MSG200, TEXT "++200 BPI "
0333 6260
0334 6040
0335 0220
0336 1140
0337 4040
0340 4040
0341 4040
0342 4000
0343 3736 MSGEND, TEXT "++END OF TIMING-!"
0344 0516
0345 0440
0346 1706
0347 4024
0350 1115
0351 1116
0352 0737
0353 3000

```



```

0000 00000000 10100000 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11111111 11111111 11111111 00000000 00000000 11111111 11111111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 00000000 00000000
0500 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 11111111 11111111 11111111 11111000 00000000 00000000 00000000 00000000

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1100 11111111 11111111 11111111 11111111 11111111 11110000 00000000 00000000 00000000

1200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1300 11111111 11111111 11111111 11111100 00000000 00000000 00000000 00000000 00000000

1400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1500 11111111 11111111 00000000 00000000 00000000 00000000 00000000 00000000 00000000

1600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1700 11111111 11110000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

2000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111000 00000000
2100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

2200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

2400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2500 11110000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

2600
2700

3000
3100

3200
3300

3400
3500

3600
3700

```

```

4000
4100

4200
4300

4400
4500

4600
4700

5000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5100 11111111 11111111 11111111 11111111 11100000 00000000 00000000 00000000 00000000

5200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111100 00000000

5400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 00000000

5600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
5700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 10000000

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 11110000 00000000 00000000 00000000

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

```

ADDRZA	5556	GAP5CT	1707	LINK20	2346	ONESCT	0347
ALLMSG	5325	GAP6CT	1710	LINK3	0457	OTY	5446
ALLRET	5333	GAP7CT	1711	LINK00	2037	OUT	0071
ALLTBL	0160	GAP8CT	1712	LINK7	0744	OUT1	0403
ALLTU	0030	GAPCNT	2063	LINKCM	1577	OUT10	1145
ALLTU7	5304	GAPOUT	2090	LOOP	5525	OUT11	1230
ALTU79	5264	GAPS	1000	LOPRIN	5434	OUT12	1300
ARROW	5515	GAPTBL	2040	LWCR	0701	OUT13	1456
BARSPA	5616	GETBAK	3256	M100	2904	OUT14	1511
C212	5767	HDIVND	0103	M40	3744	OUT15A	2016
C215	5766	INQH	0147	M420	2213	OUT15B	2030
C230	5745	INDX10	0010	M7	2044	OUT16	2237
C336	5770	INDX12	0012	MASK77	3743	OUT17	2205
CCAR	6704	IOIT	5024	MESSAGE	3711	OUT18	2302
CHKWRD	0240	K0001	0124	MIKCT	0107	OUT19	2300
CKUL	6726	K0100	0125	MILICT	0110	OUT2	0481
CLF	6725	K0400	0142	MK1000	3282	OUT20	2400
CLT	6712	K1000	0127	MV01	6000	OUT21	2405
CNT7	0061	K1100	0122	MV02	6017	OUT3	0022
CNTALL	0060	K2100	0116	MV0200	6302	OUT4	0003
CRLF	5400	K212	0130	MV03	6033	OUT5	0711
CWCR	6702	K215	0131	MV04	6024	OUT6	0736
DAIATI	2435	K240	5476	MV05	6067	OUT7	1022
DECPRT	5501	K296	5477	MV0590	6321	OUT8	1067
DIGIT	5564	K200	5567	MV06	6096	OUT9	1117
DIVCNT	5130	K4001	0126	MV07	6102	OUTGAP	1704
DIVDEN	5056	K4100	0113	MV0END	6143	PCPU	0112
DIVIDE	5074	K4500	0117	MV0SP0	6113	PRINT	5425
DIVRET	5060	K5100	0121	MV0T1	6112	PRNCT	0104
DIVSOR	5127	K6100	0120	MV0T10	6233	RCAR	6713
DIVSR	5057	K7100	0115	MV0T11	6244	RCMR	6715
DO400	2503	K7770	0345	MV0T12	6295	RDBR	6717
DO0AP2	1632	K7775	5500	MV0T13	6266	READ	5033
DO0AP5	1660	K7776	0123	MV0T14	6297	READ1	5051
DO0AP6	1665	K7777	0114	MV0T15	6310	READ2	5070
DO0AP7	1672	KM1	0132	MV0T2	6123	RESTAR	0201
DO0AP8	1677	KM10	0135	MV0T3	6134	RET1	0404
DO0VCNT	0342	KM13	5131	MV0T4	6145	RET10	1127
DRVANT	0062	KM2	0133	MV0T5	6156	RET11	1204
DRVSAV	0344	KM3	0134	MV0T6	6167	RET12	1241
DRVSEL	0070	KM4	0102	MV0T7	6200	RET13	1404
DRVTAB	0020	KM5	1713	MV0T8	6211	RET14	1467
DVX	5120	L0AR	6703	MV0T9	6222	RET15A	2010
DV3	5110	L0CR	6705	MVRGHT	3730	RET15B	2021
ENUCHK	0303	L0BR	6707	MTP	3795	RET16	2220
ENDYST	2900	L0IVND	5126	NCPU	0111	RET17	2250
ERRT12	1317	L0BR	6706	NULL	0105	RET18	2275
EXALL	5257	LINK1	0337	NUMPNT	0002	RET19	2342
EXIME	5040	LINK11	1193	NUMTBL	0100	RET2	0434
EXTU7	5260	LINK13	1316	OD0000	0066	RET20	2403
FINI	5704	LINK15	1705	ONCE	0000	RET21	2433
GAP2CT	1706	LINK16	0144	ONES	0346	RET3	0003

RET4	0632	T0LTU7	0084
RET5	0673	TEMP1	0106
RET6	0721	TEMP3	5263
RET7	1003	TENPWR	5597
RET8	1032	TIME	5011
RET9	1100	TIMER	4487
RETGAP	1604	TITLE	0067
RENOCNT	5261	TRKSAV	0343
RFBR	6716	TU7	0040
RTE1	5000	TU7MSG	5345
RMBR	6714	TU7RET	5353
RWCR	6711	TU7TBL	0170
RWIND	5200	TYPECH	5731
RWIND7	5216	UPDATE	0264
RWINDA	5233	VALUE	5563
SAVANS	5073	VFS20	0077
SBNM	6727	VFSUPR	0100
SELALL	0072	VTEMP1	5565
SELCNT	5324	VTEMP2	5566
SELECT	0206	WGCA	5000
SELTU7	0073	WRICNT	5400
SET200	2462	XALL	0340
SET556	2200	XBAK	0136
SK00	6722	XBPI	0143
SK0F	6721	XCLRF	0001
SK1P	5025	XDEC	0076
SK1D	6723	XEND	0145
SKTR	6724	XHEAD	0005
SPACE	5365	XHEAD1	0096
START	0200	XMESS	0074
T1	0400	XNULL	0132
T10	1124	XOITY	0101
T11	1200	XPRINT	0075
T12	1235	XREAD	0140
T13	1400	XREAD1	0141
T14	1463	XREAD2	0146
T15	2000	XRITE1	0137
T16	2214	XRWND	0063
T17	2244	XRWND7	0064
T18	2272	XTIME	0007
T19	2337	XTU7	0341
T2	0430	XWGCA	0065
T20	2400		
T21	2425		
T3	0000		
T4	0027		
T5	0070		
T6	0716		
T7	1000		
T8	1027		
T9	1074		
TBLALL	0093		

ERRORS DETECTED: 0  
LINKS GENERATED: 0  
RUN-TIME: 10 SECONDS  
3K CORE USED

