

IDENTIFICATION

PRODUCT CODE: MAINDEL-08-DHRKC-H-0
PRODUCT NAME: RKBE/RKBL DATA RELIABILITY PROGRAM
DATE RELEASED: FEBRUARY, 1977
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: JOHN VROBEL/WILLIAM HEAVEY

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

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

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

COPYRIGHT (C) 1972, 1975, 1976, 1977 BY DIGITAL EQUIPMENT CORPORATION

TABLE OF CONTENTS

1. ABSTRACT
2. RESTRICTIONS
 - 2.1 HARDWARE
 - 2.2 PROGRAM STORAGE
 - 2.3 PRELIMINARY PROGRAMS
 - 2.4 EXECUTION TIME
3. SWITCH REGISTER SETTINGS
4. OPERATOR AND/OR PROGRAM ACTION
 - 4.1 STANDARD TEST PROCEDURE
 - 4.2 RK05J DRIVE CARTRIDGE MOUNTING PROCEDURE
 - 4.3 RK05F DRIVE MOUNTING PROCEDURE
 - 4.4 RK0E/RK0L DATA RELIABILITY (ACCEPT MODE)
 - 4.5 RK0E DATA RELIABILITY (MANUAL INTERVENTION MODE)
 - 4.6 CHANGE PROGRAM IOT CODES
5. ERRORS
 - 5.1 USEFUL INFORMATION
 - 5.2 ERROR HALTS
 - 5.3 ERROR TYPEOUTS
 - 5.4 ERROR RECOVERY AND ERROR DISCONNECT
 - 5.5 STATUS COMPLETE TYPEOUT AND PASS COMPLETE DISCONNECT
 - 5.6 TYPICAL ERROR TYPEOUTS
6. RESTRICTIONS
7. TROUBLE SHOOTING INFORMATION
8. PROGRAM DESCRIPTION (ACCEPT MODE)
9. CONSOLE PACKAGE ADDENDUM
10. APT-8 HOOKS
11. PROGRAM LISTING

1.

ABSTRACT

THE RK8E/RK8L DATA RELIABILITY PROGRAM IS DESIGNED PRIMARILY AS AN ACCEPTANCE TEST TO VERIFY DISK DATA TRANSFERS WITHIN THE DISK SYSTEM.

THE "ACCEPT MODE" OF OPERATION VERIFIES THE CAPABILITY OF TRANSFERRING A TOTAL 3×10^9 BITS OF DATA TO AND FROM EACH INDIVIDUAL DISK DRIVE ON THE DISK SYSTEM.

THE "MANUAL INTERVENTION MODE" IS AVAILABLE AS A HARDWARE DEBUGGING AID TO ALLOW THE OPERATOR TO SELECT DATA PATTERNS, TRANSFER LENGTHS, AND ADDRESSING.

(NOTE: LOCATION 0 CONTAINS REVISION LEVEL (IN ASCII) OF PROGRAM ON PROGRAM LOAD).

2.

RESTRICTIONS

THE RK8L CONTROL, WHICH CAN CONTROL UP TO 8 DRIVES, WILL NOT RUN WITH THE DW8E BUS ADAPTER. THE REASON FOR THIS STATEMENT IS THAT THE RK8L CONTROL USES IOTS FOR EXTENDED DRIVES 4-7 WHICH IS NOT AVAILABLE ON THE DW8E.

2.1

HARDWARE

- A. PDP-8/A, 8/E, 8/F, OR 8/M COMPUTER OR OTHER FAMILY OF 8 COMPATIBLE COMPUTER WITH NECESSARY DW8E BUS ADAPTER.
- B. AT LEAST 4K OF READ/WRITE MEMORY. AT LEAST 8K OF MEMORY IS NECESSARY FOR OPERATION OF THE CONSOLE PACKAGE.
- C. ASR-33 TELETYPE OR EQUIVALENT
- D. RK8E OR RK8L DISK CONTROL
- E. RK05J OR RK05F DISK DRIVE(S)
- F. FORMATTED 2200 DPI-16 SECTOR PACK(S).

NOTE: THE RK05F DISK DRIVE IS CONSIDERED AS TWO SEPARATE UNITS. WHEN ANSWERING ALL QUESTIONS THE SEPARATE DRIVES MUST BE SPECIFIED. DSK07, DSK17, DSK27, ETC.

2.2

PROGRAM STORAGE

THE PROGRAM OCCUPIES OR UTILIZES LOCATION 0000 TO LOCATION 7577 OF FIELD 0. ALL EXTENDED MEMORY LOCATIONS, IF AVAILABLE, ARE UTILIZED FOR TESTING.

2.3

PRELIMINARY PROGRAMS

THIS PROGRAM REQUIRES A FORMATTED CARTRIDGE ON ALL DRIVES TO BE TESTED.

ALL BASIC AND EXTENDED MEMORY DIAGNOSTICS SHOULD BE RUN PRIOR TO RUNNING THIS PROGRAM.

RK8E CONTROL: RUN THE RK8E DISKLESS CONTROL TEST AND THE RK8E/RK8L DISK FORMATTER IF THIS DIAGNOSTIC FAILS TO OPERATE PROPERLY.

RK8L CONTROL: RUN THE RK8L INSTRUCTION TEST AND THE RK8E/RK8L FORMATTER IF THIS DIAGNOSTIC FAILS TO OPERATE PROPERLY.

2.4 EXECUTION TIME *****

THE PROGRAM EXECUTION TIME (I.E., PASSING 3×10^9 BITS OF DATA ON A DISK DRIVE), IS APPROX. 4 HOURS PER DISK DRIVE ON A 4K MEMORY SYSTEM OR APPROX. 3.5 HOURS PER DISK DRIVE ON SYSTEMS WITH EXTENDED MEMORY.

3. SWITCH REGISTER SETTINGS *****

SWR0#1 LOOP ON WRITE SEQUENCE.
SWR1#1 LOOP ON READ SEQUENCE.
SWR2#1 INHIBIT ALL ERROR TYPEOUTS
SWR3#1 TYPE "STATUS-COMplete" REPORT.
SWR4#1 PROGRAM STOP ON HALT.
SWR5#1 DRIVE DISCONNECT AFTER PASS COMPLETION.
SWR6#1 PERFORM ONLY "OVERLAP SEEKS", DO NOT EXECUTE DATA BREAKS.

4. OPERATOR AND/OR PROGRAM ACTION *****

4.1 STANDARD TEST PROCEDURE *****

- A. START AS SPECIFIED THROUGHOUT THIS DOCUMENTATION IS KEY CLEAR AND THEN KEY CONTINUE ON PDP8/E, PDP8/M, AND PDP8/F COMPUTERS.
- B. LOAD THE PROGRAM INTO MEMORY FIELD 0 USING THE STANDARD BINARY LOADER TECHNIQUE.
- C. IF IT IS DESIRED TO CHANGE THE IOT CODES WITHIN THE PROGRAM, FOLLOW THE PROCEDURE IN SECTION 4.6.
- D. RUN THE ACCEPTANCE MODE OF DATA RELIABILITY WITH ALL DRIVES AND MEMORY AVAILABLE BY FOLLOWING THE PROCEDURE

IN SECTION 4.4.

- E. THE MANUAL INTERVENTION MODE, SECTION 4.5, MAY BE USED FOR TROUBLE SHOOTING, IF DESIRED.
- F. IF POSSIBLE SWR4#1 SHOULD ALWAYS BE USED TO STOP THE PROGRAM.
- G. IF THE PROGRAM HAS BEEN STOPPED DUE TO SWR4#1, THE PROGRAM CAN BE RESTARTED, AND THE INITIAL STARTUP QUESTIONS BYPASSED, BY USING 0205 AS THE RESTART ADDRESS.
- H. FOR THE ABSOLUTE LOCATIONS OF ALL KNOWN HALTS IN THIS PROGRAM, ACCESS PAGE 1-22 OF THE PROGRAM LISTING.

4.2

RK05J DRIVE CARTRIDGE MOUNTING PROCEDURE

THE FOLLOWING IS THE CORRECT CARTRIDGE MOUNTING PROCEDURE FOR THE RK05J DISK DRIVE. ANY DEVIATION ENCOUNTERED DURING THIS PROCEDURE WILL BE CONSIDERED AN ERROR CONDITION.

- A. SET SWITCH LABELED "RUN/LOAD" TO THE "LOAD" POSITION.
- B. TURN AC POWER TO DISK DRIVE ON.
- C. VERIFY THAT THE LIGHT LABELED "PWR" IS ON.
- D. WAIT FOR THE LIGHT LABELED "LOAD" TO COME ON.
- E. VERIFY THAT THE LIGHTS LABELED "RDY", "ON CYL", "FAULT", "WT", AND "RD" ARE OFF.
- F. OPEN ACCESS DOOR.
- G. INSERT CARTRIDGE.
- H. CLOSE ACCESS DOOR.
- I. SET SWITCH LABELED "RUN/LOAD" TO THE "RUN" POSITION.
- J. WAIT FOR THE LIGHTS LABELED "RDY" AND "ON CYL" TO COME ON.
- K. TOGGLE SWITCH LABELED "WT PROT" AND VERIFY THAT THE LIGHT LABELED "WT PROT" GOES ON AND OFF.
- L. TOGGLE SWITCH LABELED "WT PROT" UNTIL THE LIGHT LABELED "WT PROT" IS OFF.
- M. VERIFY THAT LIGHTS LABELED "FAULT", "WT", "RD", AND "LOAD" ARE OFF.

4.3

RK05F DRIVE SETUP PROCEDURE

THE FOLLOWING IS THE CORRECT SETUP PROCEDURE

FOR THE RK05F DISK DRIVE, ANY DEVIATION ENCOUNTERED DURING THIS PROCEDURE WILL BE CONSIDERED AN ERROR CONDITION.

- A. SET SWITCH LABELED "RUN/LOAD" TO THE "LOAD" POSITION.
- B. TURN AC POWER ON.
- C. VERIFY THAT LIGHT LABELED "PWR" IS ON.
- D. WAIT FOR LIGHT LABELED "LOAD" TO COME ON.
- E. VERIFY THAT LIGHTS LABELED "RDY", "ON CYL", "FAULT", "WT", AND "RD" ARE OFF.
- F. SET SWITCH LABELED "RUN/LOAD" TO THE "RUN" POSITION.
- G. WAIT FOR LIGHTS LABELED "RDY" AND "ON CYL" TO COME ON.
- H. TOGGLE SWITCH LABELED "WT PROT" AND VERIFY THAT THE LIGHT LABELED "WT PROT" GOES ON AND OFF.
- I. TOGGLE SWITCH LABELED "WT PROT" UNTIL LIGHT LABELED "WT PROT" GOES OFF.
- J. VERIFY THAT LIGHTS LABELED "FAULT", "WT", "RD", AND "LOAD" ARE OFF.

4.4

RK08E/RK08L DATA RELIABILITY (ACCEPT MODE)

- A. MAKE READY ALL DRIVES TO BE TESTED USING THE RK05J DRIVE CARTRIDGE MOUNTING PROCEDURE SECTION 4.2 OR THE RK05F DRIVE PROCEDURE IN SECTION 4.3.
- B. SET SWITCH LABELED "RUN/LOAD" TO THE "LOAD" POSITION ON ALL DRIVES NOT BEING TESTED.
- C. VERIFY THAT AC POWER IS ON, ON ALL DRIVES NOT BEING TESTED.
- D. SET THE SWITCH REGISTER TO 0200 AND PRESS LOAD ADDRESS.
- E. SET THE SWITCH REGISTER TO 0000 AND PRESS START.
- F. THE OPERATOR MAY SET SWRS=1 IF IT IS DESIRED TO HAVE THE PROGRAM AUTOMATICALLY DISCONNECT EACH DISK DRIVE AS EACH MAKE THEIR PASS COMPLETION. (NOTE: IF SWRS=0, ALL DISK DRIVES WILL CONTINUE TO RUN AFTER THEIR PASS COMPLETION)
- G. THE TTY WILL PRINT THE FOLLOWING PROGRAM NAME AND QUESTION,

RK08E/RK08L DATA RELIABILITY
EXTENDED R/W MEMORY (0-7)?

THE OPERATOR SHOULD THEN TYPE THE AMOUNT OF EXTENDED READ/ WRITE MEMORY BANKS NUMBERED SEQUENTIALLY FROM BANK 0, AS INDICATED BY THE TTY QUESTION.
- H. THE TTY WILL PRINT THE FOLLOWING QUESTION(S), ASKING THE

DESIRED DISK DRIVE(S) TO BE USED IN TESTING.

EXERCISE DISK07
EXERCISE DISK17
EXERCISE DISK27
EXERCISE DISK37
EXERCISE DISK47
EXERCISE DISK57
EXERCISE DISK67
EXERCISE DISK77

FOR THE QUESTION(S) ABOVE, TYPE Y FOR YES, IF IT IS DESIRED TO TEST THE DISK DRIVE IN QUESTION, OTHERWISE, TYPE N FOR NO.

I. THE TTY WILL PRINT THE FOLLOWING QUESTION.

ACCEPT MODE?

THE OPERATOR SHOULD THEN TYPE Y FOR YES TO RUN THE ACCEPTANCE MODE OF OPERATION.

J. THE TTY WILL PRINT THE FOLLOWING QUESTION.

ARE YOU SURE?

IF THE OPERATOR IS CERTAIN OF THE AMOUNT OF MEMORY, THE DISK DRIVE(S) SELECTED, AND THE MODE OF OPERATION, TYPE Y FOR YES. TYPING N FOR NO WILL RESULT IN A REPEAT OF ALL MESSAGES AND QUESTIONS ENCOUNTERED THUS FAR.

K. THE PROGRAM SHOULD START TESTING THE DISK DRIVE(S) AND MEMORY SELECTED.

L. THE "STATUS=COMPLETE" TYPEOUT SHOULD OCCUR UPON PASS COMPLETION OF EACH DISK DRIVE. ALL OTHER TYPEOUTS OR HALTS WILL BE CONSIDERED AS AN ERROR CONDITION. SEE SECTION 5.5 FOR "STATUS=COMPLETE" TYPEOUT.

M. A SUCCESSFUL PASS COMPLETE ON A DISK DRIVE WILL BE CONSIDERED AS NO "HARD" ERRORS AND NO MORE THAN ONE (1) "SOFT" ERROR PER PASS COMPLETE.

N. IF ANY ERRORS DO OCCUR, THE OPERATOR SHOULD ACCESS SECTION 5 IN THIS DOCUMENTATION.

4.5

RK08E/RK08L DATA RELIABILITY (MANUAL INTERVENTION MODE)

THE MANUAL INTERVENTION MODE IS AVAILABLE AS A TROUBLE SHOOTING AID AND SHOULD ONLY BE USED FOR SUCH PURPOSES, IF DESIRED.

A. MAKE READY ALL DISK DRIVES TO BE TESTED USING THE RK05J DRIVE CARTRIDGE MOUNTING PROCEDURE SETION 4.2. OR THE RK05F DRIVE PROCEDURE SECTION 4.3.

B. SET SWITCH LABELED "RUN/LOAD" TO THE "LOAD" POSITION ON

ALL DRIVES NOT BEING TESTED.

- C. VERIFY THAT AC POWER IS ON, ON ALL DRIVES NOT BEING TESTED.
- D. SET THE SWITCH REGISTER TO 0200 AND PRESS LOAD ADDRESS.
- E. SET THE SWITCH REGISTER TO 0000 AND PRESS START.
- F. THE TTY WILL PRINT THE FOLLOWING PROGRAM NAME AND QUESTION.

RK0E/RK0L DATA RELIABILITY
EXTENDED R/W MEMORY (0-7)?

THE OPERATOR SHOULD THEN TYPE THE AMOUNT OF EXTENDED READ/
WRITE MEMORY BANKS NUMBERED SEQUENTIALLY FROM BANK 0, AS
INDICATED BY THE TTY QUESTION.

- G. THE TTY WILL PRINT THE FOLLOWING QUESTION(S), ASKING THE DESIRED DISK DRIVE(S) TO BE USED IN TESTING.

EXERCISE DISK0?
EXERCISE DISK1?
EXERCISE DISK2?
EXERCISE DISK3?
EXERCISE DISK4?
EXERCISE DISK5?
EXERCISE DISK6?
EXERCISE DISK7?

FOR THE QUESTION(S) ABOVE, TYPE Y FOR YES, IF IT IS DESIRED
TO TEST THE DISK DRIVE IN QUESTION, OTHERWISE, TYPE N FOR
NO.

- H. THE TTY WILL PRINT THE FOLLOWING QUESTION.

ACCEPT MODE?

THE OPERATOR SHOULD THEN TYPE N FOR NO TO RUN THE MANUAL
INTERVENTION MODE OF OPERATION.

- I. THE TTY WILL THEN PRINT THE FOLLOWING QUESTION, ASKING IF THE OPERATOR DESIRES TO SELECT A CONSTANT MEMORY FIELD, RATHER THAN THE NORMAL RANDOM FIELD SELECTION.

FIELD?

IF THE OPERATOR DESIRES TO SELECT A CONSTANT FIELD,
TYPE Y FOR YES, OTHERWISE, TYPE N FOR NO. IF Y WAS TYPED THE
TTY WILL SPACE OUT ONCE AND WAIT FOR THE OPERATOR TO TYPE
THE DESIRED FIELD IN OCTAL (0-7).

- J. THE TTY WILL PRINT THE FOLLOWING QUESTION, ASKING IF THE OPERATOR DESIRES TO SELECT A CONSTANT TRACK, RATHER THAN THE NORMAL RANDOM TRACK SELECTION.

TRACK?

IF THE OPERATOR DESIRES TO SELECT A CONSTANT TRACK, TYPE Y FOR YES, OTHERWISE, N FOR NO. IF Y WAS TYPED, THE TTY WILL SPACE OUT ONCE AND WAIT FOR THE OPERATOR TO INPUT THE DESIRED TRACK ADDRESS (00000-14537).

- K. THE TTY WILL PRINT THE FOLLOWING QUESTION, ASKING IF THE OPERATOR DESIRES TO SELECT HALF BLOCK OR FULL BLOCK TRANSFERS, RATHER THAN THE NORMAL RANDOM SELECTION.

BLOCK LENGTH?

IF THE OPERATOR DESIRES TO CHANGE THE BLOCK LENGTH, TYPE Y FOR YES, OTHERWISE, N FOR NO. IF Y WAS TYPED THE TTY WILL SPACE OUT ONCE AND WAIT FOR THE OPERATOR TO TYPE THE BLOCK LENGTH DESIRED (0=256 WORD BLOCK OR 1=128 WORD BLOCK).

- L. THE TTY WILL PRINT THE FOLLOWING QUESTION, ASKING IF THE OPERATOR DESIRES TO SELECT A CONSTANT NUMBER OF SECTORS TO BE TRANSFERRED, RATHER THAN THE NORMAL RANDOM SECTOR SELECTION.

EXTRA SECTORS?

IF THE OPERATOR DESIRES TO SELECT A CONSTANT AMOUNT OF SECTORS, TYPE Y FOR YES, OTHERWISE, N FOR NO. IF Y WAS TYPED THE TTY WILL SPACE OUT ONCE, AND WAIT FOR THE OPERATOR TO TYPE IN THE EXTRA SECTORS DESIRED (00-17). (NOTE: IF THE FIELD AND THE BLOCK LENGTH PREVIOUSLY SELECTED WAS 0, THE AMOUNT OF EXTRA SECTORS WILL BE LIMITED TO 07. OTHERWISE THE MAXIMUM AMOUNT IS LIMITED TO 17.)

- M. THE TTY WILL PRINT THE FOLLOWING QUESTION, ASKING IF THE OPERATOR DESIRES TO SELECT A DATA PATTERN, RATHER THAN NORMAL RANDOM DATA SELECTION.

DATA?

IF THE OPERATOR DESIRES TO SELECT A DATA PATTERN, TYPE Y FOR YES, OTHERWISE, N FOR NO. IF Y WAS TYPED, THE TTY WILL DO A "CRLF" AND WAIT FOR THE OPERATOR TO TYPE IN 12 OCTAL DATA WORDS TO BE USED IN TESTING.

- N. THE TTY WILL PRINT THE FOLLOWING QUESTION.

ARE YOU SURE?

IF THE OPERATOR IS CERTAIN OF THE INFORMATION SELECTED, TYPE Y FOR YES, TYPING N FOR NO WILL RESULT IN A REPEAT OF ALL MESSAGES AND QUESTIONS ENCOUNTERED THUS FAR.

- P. THE PROGRAM SHOULD START EXECUTING THE OPERATIONS SELECTED.

- R. IF ERRORS ARE ENCOUNTERED, ACCESS SECTION 5 IN THIS DOCUMENTATION.

4.6

CHANGE PROGRAM DEVICE IOT CODES

THE PROGRAM NORMALLY RECOGNIZES DEVICE IOT CODE X74X. TO CHANGE THE DEVICE IOT CODES WITHIN THE PROGRAM:

- A. SET THE SWITCH REGISTER TO 0204 AND PRESS LOAD ADDRESS.
- B. SET THE SWITCH REGISTER TO 0000, SET SWITCH REGISTER BITS 3-8 TO THE DESIRED DEVICE IOT CODE, AND PRESS START.
- C. THE PROGRAM WILL CHANGE THE DEVICE IOT CODES WITHIN THE PROGRAM AND THEN HALT.
- D. PRESSING KEY CONTINUE WILL START THE PROGRAM AT LOCATION 0200 (SEE SECTIONS 4.4 OR 4.5 FOR OPERATION INSTRUCTIONS).

5. ERRORS

5.1 USEFUL INFORMATION

ALL STATUS ERRORS WILL BE REPORTED AS STATUS ERRORS. ALL DATA ERRORS WILL BE REPORTED AS DISK DATA ERRORS.

WHEN DATA IS BEING READ OFF THE DISK AND A CRC ERROR OCCURRES THE PROGRAM WILL REPORT THE ERROR AS A READ STATUS ERROR. THE PROGRAM WILL THEN CHECK THE DATA READ FOR DATA ERRORS. IF DATA ERRORS EXIST THEY WILL BE REPORTED AS DISK DATA ERRORS.

5.2 ERROR HALTS

ERROR HALTS FOR WHICH THERE ARE NO ERROR TYPEOUTS ARE LISTED AND DEFINED AS FOLLOWS.

BIGSTP	MASTER ERROR HALT FOR ALL OF THE FOLLOWING ERROR STOPS. WHEN THE COMPUTER HALTS THE AC REGISTER WILL INDICATE THE PC OF THE FAILING ERROR STOP.
INTER1	NO DISK INTERRUPT
ERHLT0	SKIP TMAP FOR IOT "DLSC"
ERHLT2	SKIP TMAP FOR IOT "DCLR"
ERHLT3	SKIP TMAP FOR IOT "DLAG"
ERHLT5	SKIP TMAP FOR IOT "DRST"
ERHLT6	SKIP TMAP FOR IOT "DLDC"
BADHLT	CHECKSUM FAILED BUT WORD-BY-WORD COMPARE WORKED

NODSKS NO DISKS AVAILABLE TO RUN
FLDHLT PROGRAM WILL ONLY RUN IN FIELD 0

FOR THE ABSOLUTE LOCATIONS OF THE HALTS LISTED ABOVE,
ACCESS PAGE 1-22 OF THE PROGRAM LISTING.

5.3 ERROR TYPEOUTS

WHEN AN ERROR OCCURRES THE PROGRAM WILL PRINT AN
"ERROR HEADER" WHICH WILL SPECIFY THE PARTICULAR TYPE
OF ERROR FOUND AT THE TIME OF THE FAILURE.

POSSIBLE "ERROR HEADERS" ARE AS FOLLOWS.

WRITE STATUS ERROR
READ STATUS ERROR
DISK DATA ERROR
RECALIBRATE STATUS ERROR

AFTER THE "ERROR HEADER" MENTIONED ABOVE IS TYPED, THE
PROGRAM WILL PRINT THE FOLLOWING ERROR INFORMATION
FOUND AT THE TIME OF THE FAILURE, PERTAINING TO THE
FAILURE. POSSIBLE TYPEOUTS ARE AS FOLLOWS.

PC: PROGRAM LOCATION OF THE ACTUAL FAILURE.
ST: CONTENTS OF THE STATUS REGISTER.
EX: EXTENDED DRIVE BIT
CM: SOFTWARE COMMAND REGISTER.
IA: INITIAL SOFTWARE DISK ADDRESS REGISTER OR THE
CYLINDER, SURFACE, AND SECTOR BITS.
DA: FINAL SOFTWARE DISK ADDRESS REGISTER OR THE
CYLINDER, SURFACE, AND SECTOR BITS.
CA: SOFTWARE INITIAL CURRENT ADDRESS
WC: SOFTWARE INITIAL WORD COUNT
FW: SOFTWARE FINAL WORD COUNT
AS: SECTOR IN ERROR ON THE PARTICULAR CYLINDER
AND SURFACE IN QUESTION,
WA: WORD ADDRESS WITHIN THE SECTOR IN ERROR
AD: BREAK ADDRESS OF DATA BREAK IN COMPUTER.
DG: EXPECTED DATA
DB: DATA FOUND DURING DATA BREAK.

5,4

ERROR RECOVERY AND ERROR DISCONNECT

WHEN A READ, WRITE, OR DISK DATA ERROR OCCURS (SEE SECTION 5.3), THE PROGRAM WILL TRY TO REPEAT THE FAILING SEQUENCE FOUR (4) TIMES. IF THE ERROR HAS OCCURRED FOUR (4) TIMES SIMULTANEOUSLY, THE ERROR WILL BE CONSIDERED AS A NON-RECOVERABLE ERROR, THE "ERROR HEADER" WILL BE CHANGED TO INDICATE "NON-RECOVERABLE" ERROR, ANOTHER DISK ADDRESS WILL BE SELECTED FOR TESTING. IF A "SOFT" ERROR SHOULD OCCUR ON A TRACK, THE PROGRAM WILL RETRY THE READ SEQUENCE (64) TIMES BEFORE SELECTING ANOTHER TRACK FOR TESTING. (NOTE: THIS 64 RETRY ON "SOFT" ERRORS WILL BE TERMINATED ON A "HARD" ERROR).

POSSIBLE NON-RECOVERABLE ERROR HEADERS ARE AS FOLLOWS.

NON-RECOVERABLE READ STATUS ERROR
NON-RECOVERABLE WRITE STATUS ERROR
NON-RECOVERABLE DISK DATA ERROR

IF A NON-RECOVERABLE READ OR WRITE ERROR SHOULD OCCUR, THE DISK IN QUESTION WILL THEN BE RECALIBRATED (RESTORED TO CYLINDER 0). IF THE RECALIBRATE SEQUENCE FAILS, THE DISK DRIVE IN ERROR WILL BE DISCONNECTED BY THE PROGRAM AND NO LONGER BE TESTED.

THE FOLLOWING "DISCONNECT" AND "STATUS-COMPLETE" TYPEOUTS SHOULD OCCUR.

RECALIBRATE ERROR DISCONNECT!
DISK X DISCONNECTED!
DSK HARD SOFT COMP
X 0030 0010 0001
X 0240 5670 0001

IF ALL DISKS ON THE SYSTEM HAVE BEEN DISCONNECTED DO TO RECALIBRATE ERRORS THE FOLLOWING TYPEOUT WILL OCCUR AND THE PROGRAM WILL HALT.

DISK SYSTEM SHUT DOWN, NO DISKS TO RUN!

5,5

STATUS-COMPLETE TYPEOUT AND PASS COMPLETE DISCONNECT

ALL ERRORS AND PASS COMPLETES ARE TALLIED BY THE PROGRAM PER DISK DRIVE.

THE FOLLOWING IS AN EXAMPLE OF THE "STATUS-COMPLETE" TYPEOUT THAT WILL OCCUR WHEN SWR3=1 INDICATING TYPE THIS REPORT, A PASS COMPLETE OCCURS ON A DRIVE UNDER TEST, OR A DRIVE IS DISCONNECTED DO TO A RECALIBRATE ERROR.

DSK HARD SOFT COMP
X XXXX XXXX XXXX
X XXXX XXXX XXXX
X XXXX XXXX XXXX

X XXXX XXXX XXXX

THE TYPEOUT AS MENTIONED ABOVE IS DESCRIBED AS FOLLOWS.

DSK DISK DRIVE IN QUESTION.

HARD ALL ERRORS OTHER THAN THAT DEFINED AS A SOFT ERROR.

SOFT A READ CRC STATUS ERROR WITH BAD DATA PER TRANSFER WITH RECOVERY POSSIBLE WITHIN FOUR (4) RETRYs. (NOTE: FOUR (4) CONSECUTIVE RETRYs WILL BE CONSIDERED AS A NON-RECOVERABLE ERROR OR A "HARD" ERROR).

COMP PASS COMPLETES. <3 X 10(9) BITS>

IF SWR5=1 INDICATING "DISCONNECT ON PASS COMPLETION", AND A DISK DRIVE UNDER TEST MAKES A PASS COMPLETION, THE FOLLOWING TYPEOUT WILL OCCUR AND THE DRIVE WILL BE DISCONNECTED.

DISK X PASS COMPLETE!
DISK X DISCONNECTED!
DSK HARD SOFT COMP
X XXXX XXXX XXXX
X XXXX XXXX XXXX

IF SWR5=0 INDICATING DON'T "DISCONNECT ON PASS COMPLETION", AND A DISK DRIVE UNDER TEST MAKES A PASS COMPLETION, THE FOLLOWING TYPEOUT WILL OCCUR AND THE DRIVE WILL CONTINUE TO RUN.

DISK X PASS COMPLETE!
DSK HARD SOFT COMP
X XXXX XXXX XXXX
X XXXX XXXX XXXX

IF SWR5=1 AND ALL DRIVES HAVE MADE THEIR PASS COMPLETION AND HAVE BEEN DISCONNECTED, THE FOLLOWING TYPEOUT WILL OCCUR AND THE COMPUTER WILL HALT.

DISK SYSTEM SHUT DOWN, NO DISKS TO RUN!

5.6

TYPICAL ERROR TYPEOUTS

THE FOLLOWING IS AN EXAMPLE OF AN "ERROR HEADER" AND ERROR TYPEOUT THAT COULD HAVE OCCURRED ON A WRITE STATUS ERROR. (NOTE CRC IN THE STATUS INDICATOR "ST!")

WRITE STATUS ERROR
PC12371 ST14010 EX10001 CM14000 IA10001 DA10002
CA13600 WC17000 FW10000

THE FOLLOWING IS AN EXAMPLE OF AN ERROR TYPEOUT THAT COULD HAVE OCCURRED IF THE STATUS REGISTER FAILED ON A SEEK ONLY FUNCTION.

SEEK STATUS ERROR
PC:12076 ST:14002 EX:10001 CM:3000 DA:14007

THE FOLLOWING IS A TYPICAL EXAMPLE OF AN "ERROR HEADER"
AND ERROR TYPEOUT THAT COULD HAVE OCCURRED ON A DISK
DATA ERROR. (NOTE: ADDITION DATA ERRORS IN BUFFER)

DISK DATA ERROR
PC:1674 ST:14010 EX:10001 CM:1432 IA:1035 DA:1021
CA:10001 WC:15000 FW:17400
AS:10015 WA:10007 AD:10010 DG:10537 DB:10536
AS:10015 WA:10077 AD:10100 DG:17777 DB:17776
AS:10016 WA:10002 AD:10405 DG:16167 DB:16166

6. RESTRICTIONS

ALL DISK DRIVES SHOULD BE SET TO THE LOAD POSITION
THAT ARE NOT BEING TESTED.

7. TROUBLE SHOOTING INFORMATION

IOT	FUNCTION
---	-----
6740 DLSC	LOAD SECTOR COUNTER AND EXTENDED DRIVE BIT FOR RK6L.
AC	
--	
0-3	LOAD THE DESIRED AMOUNT OF SECTORS TO BE TRANSFERRED WITH THE BINARY VALUE IN AC BITS 0-3.
4	EXTENDED DRIVE BIT FOR DRIVES 4-7
6741 DSKP	"SKIP" SKIP IF TRANSFER DONE FLAG OR ERROR FLAG IS SET.
6742 DCLR	"CLEAR" FUNCTION IS REGULATED BY AC BITS 10 AND 11. THE AC IS THEN CLEARED.
AC10	AC11
----	----
0	0
	CLEAR THE AC AND STATUS REGISTER.
0	1
	CLEAR THE AC, CONTROL, AND MAJOR REGISTERS. THIS INSTRUCTION WILL STOP THE CONTROL EVEN IF IT IS WRITING A HEADER. THIS IS THE ONLY INSTRUCTION THAT CLEARS MAINTENANCE MODE.

1	0	CLEAR AC, RECALIBRATE DISK DRIVE, AND CLEAR STATUS REGISTER.
6743	DLA6	"LOAD DISK ADDRESS AND GO" LOAD THE DISK CYLINDER, SURFACE, AND SECTOR FROM THE AC, CLEAR THE AC, AND DO THE COMMAND IN THE COMMAND REGISTER.
AC	--	
0-6		CYLINDER
7		SURFACE (1=UPPER) (0=LOWER)
8-11		SECTOR
6744	DLCA	"LOAD CURRENT ADDRESS" LOAD THE CURRENT ADDRESS FROM AC. THE AC IS THEN CLEARED.
AC	--	
0-11		CURRENT ADDRESS
6745	DRST	"READ STATUS" CLEAR THE AC AND READ THE CONTENTS OF THE STATUS REGISTER INTO THE AC.
AC	--	
0		TRANSFER DONE
1		READY TO SEEK, READ, OR WRITE.
2		NOT USED
3		SEEK FAIL
4		DISK FILE READY
5		CONTROL BUSY ERROR
6		TIME OUT ERROR
7		WRITE LOCK ERROR
8		CRC ERROR
9		DATA RATE ERROR
10		DRIVE STATUS ERROR
11		CYLINDER ADDRESS ERROR
6746	DLDC	"LOAD COMMAND" LOAD THE COMMAND REGISTER FROM AC, CLEAR THE AC, AND CLEAR THE STATUS REGISTER.
AC	--	
0-2#0		READ DATA
0-2#1		READ ALL
0-2#2		WRITE LOCK
0-2#3		SEEK ONLY
0-2#4		WRITE DATA

0-205	WRITE ALL
0-206	NOT USED
0-207	NOT USED
3	ENABLE INTERRUPT
4	ENABLE SET TRANSFER DONE ON SEEK DONE
5	HALF BLOCK 120 WORDS
6	EXTENDED MEMORY ADDRESS
7	EXTENDED MEMORY ADDRESS
8	EXTENDED MEMORY ADDRESS
9	UNIT SELECT
10	UNIT SELECT
11	EXTENDED CYLINDER ADDRESS

6747 DMAN

"MAINTENANCE IOT" LOAD THE MAINTENANCE REGISTER FROM THE AC. THE FUNCTION IS REGULATED BY THE AC BITS. MAINTENANCE MODE CAN ONLY BE CLEARED BY DCLK "CLEAR CONTROL".

AC
--

0	ENTER MAINTENANCE MODE
1	ENABLE SHIFT TO LOWER BUFFER
2	AC BIT 10, CRC REGISTER, AND THE LOWER DATA BUFFER ARE CONNECTED AS A SHIFT REGISTER. AC BIT 10 DATA SHIFTS TO THE CRC, THE CRC SHIFTS TO THE LOWER DATA BUFFER.
3	SHIFT COMMAND REGISTER TO THE LOWER DATA BUFFER.
4	SHIFT THE SURFACE AND SECTOR REGISTER TO THE LOWER DATA BUFFER.
5	SHIFT AC 10 DATA TO THE UPPER DATA BUFFER. THE UPPER BUFFER SHOULD SINK IN THE SILO WHEN FULL.
6	ONE SINGLE CYCLE BREAK REQUEST. DIRECTION IS REGULATED BY FUNCTION IN THE COMMAND REGISTER.
7	CLEAR AC THEN READ THE LOWER DATA BUFFER TO THE AC.
8	NOT USED.
9	NOT USED.
10	USED AS DATA WITH OTHER BITS IN THE MAINTENANCE MODE.
11	NOT USED

8. PROGRAM DESCRIPTION (ACCEPT MODE)

THE FOLLOWING IS BRIEF DESCRIPTION OF THE STEPS TAKEN BY THE PROGRAM WHEN RUNNING THE ACCEPT MODE.

A. A RANDOM FIELD IS GENERATED. IF FIELD GENERATED IS A NON-EXISTING FIELD, THE MAXIMUM FIELD AVAILABLE WILL BE USED.

- B. A RANDOM BLOCK LENGTH IS GENERATED (128 OR 256 WORD SECTORS).
- C. A RANDOM AMOUNT OF SEQUENTIAL SECTORS TO TRANSFER IS GENERATED. IF THE FIELD PREVIOUSLY SELECTED WAS AN EXTENDED FIELD OR IF HALF BLOCK TRANSFERS WERE SELECTED (128 WORD SECTORS), THE AMOUNT OF SECTORS WILL BE LIMITED TO 17(8). IF THE FIELD SELECTED WAS FIELD 0 AND IF FULL BLOCK TRANSFERS WERE SELECTED (256 WORD SECTORS), THE AMOUNT OF SECTORS WILL BE LIMITED TO 7(8).
- D. A RANDOM STARTING SECTOR WILL BE GENERATED. THE RANDOM AMOUNT OF EXTRA SECTORS PREVIOUSLY GENERATED WILL BE ADDED TO THIS STARTING SECTOR, DETERMINING THE ACTUAL LENGTH OF THE DATA TRANSFER. IF THE STARTING SECTOR WAS 14 AND THE AMOUNT OF EXTRA SECTORS WAS 6, SECTORS 14, 15, 16, 17, 00, 01, AND 02 WILL BE USED FOR TRANSFERING DATA.
- E. AN INITIAL SOFTWARE WORD COUNT WILL BE CALCULATED.
- F. AN INITIAL RANDOM CURRENT ADDRESS WILL BE GENERATED. IF THE FIELD PREVIOUSLY GENERATED WAS FIELD 0, THE CURRENT ADDRESS WILL BE LIMITED WITHIN THE END OF THE PROGRAM +4000 LOCATIONS.
- G. THE BUFFER SELECTED WILL BE FILLED WITH RANDOM DATA, CHECKSUMMED, AND THE CHECKSUM SAVED. (NOTE: BUFFER IS DEPENDENT ON FIELD, WORD COUNT, BLOCK LENGTH, AND CURRENT ADDRESS PREVIOUSLY SELECTED.)
- H. THE PROGRAM WILL THEN POLE THE DISK DRIVES. DRIVE SELECTION IS SEQUENTIAL, THAT IS DISK0, DISK1, DISK2, ETC.
- I. DATA WILL BE WRITTEN ON THE SELECTED DISK DRIVE TO COMPLETE THE SEEK OPERATION USING THE RANDOM PARAMETERS GENERATED ABOVE. AS DATA IS WRITTEN, A BACK GROUND PROGRAM WILL CLEAR THE BUFFER AREA ALREADY WRITTEN ON THE DISK.
- J. WHEN THE WRITE AND CLEAR IS COMPLETE, DATA WILL BE READ OFF THE CURRENT DRIVE INTO THE BUFFER AREA. AS DATA IS READ, A BACK GROUND PROGRAM WILL CHECKSUM THE BUFFER INFORMATION ALREADY READ OFF THE DISK.
- K. WHEN THE READ AND CHECKSUM IS COMPLETE, THE CHECKSUM FOUND WILL BE COMPARED TO THE CHECKSUM SAVED PREVIOUS TO THE WRITE OPERATION. IF CHECKSUMS DO NOT COMPARE OR IF A CRC ERROR HAS OCCURRED, A WORD BY WORD COMPARE WILL BE MADE TO DETERMINE AND TYPE OUT THE BAD DATA FOUND.
- L. STEPS A-H WILL BE REPEATED AND THE DRIVE POLE WILL BE STARTED AT THE CURRENT DRIVE +1.
- M. FOR ALL POSSIBLE ERRORS, SEE SECTION 5 IN THIS DOCUMENT.

9.

CONSOLE PACKAGE ADDENDUM

9.1. DESCRIPTION

THE CONSOLE PACKAGE HAS BEEN ADDED TO THIS DIAGNOSTIC TO ALLOW THE PROGRAM TO RUN WITH NO HARDWARE SWITCH REGISTER AND TO HAVE COMMUNICATIONS WITH THE DIAGNOSTIC VIA A TERMINAL. THE DIAGNOSTIC CAN BE RUN IN TWO MODES WITH THE CONSOLE PACKAGE . 1) RUNNING WITH THE CONSOLE PACKAGE ACTIVE - THIS ALLOWS THE OPERATOR CONTROL OF THE DIAGNOSTIC THROUGH THE TERMINAL. THE DIAGNOSTIC WILL ASK FOR THE VALUE OF THE PSEUDO SWITCH REGISTER, BEFORE CONTINUING WITH EXECUTION OF THE DIAGNOSTIC. ALL ERRORS AND PASS COMPLETES WILL BE PRINTED AT THE TERMINAL. NO HALTS WILL BE EXECUTED. 2) CONSOLE PACKAGE NOT ACTIVE-THIS WILL RESULT IN THE NORMAL STANDALONE OPERATION OF THE PROGRAM AS DISCRIBED IN SECTIONS 1 THROUGH 8 OF THIS DUCUMENT,

9.2 RESTRICTIONS

- 1) RUNNING THE CONSOLE PACKAGE REQUIRES THAT THE PSEUDO SWITCH REGISTER BE USED.
- 2) ONCE RUNNING THE CONSOLE PACKAGE NONACTIVE AND NOW DESIRE TO RUN IT ACTIVE, ONE MUST RELOAD THE DIAGNOSTIC AND INITILIZE FOR A ACTIVE CONSOLE PACKAGE.

9.3 INITIALIZATION

FOR A ACTIVE CONSOLE PACKAGE

- 1.) SET LOCATION 21 BIT0=0 TO INDICATE USE PSEUDO SWITCH REGISTER.
- 2.) SET LOCATION 22 BIT3=1 TO INDICATE CONSOLE PACKAGE ACTIVE.

FOR A NON ACTIVE CONSOLE PACKAGE

- 1.) SET LOCATION 21 BIT0=1 TO INDICATE NOT TO USE PSEUDO SWITCH REGISTER, BUT TU USE HARDWARE SWITCHES.
- 2.) SET LOCATION 22 BIT3=0 TO INDICATE CONSOLE PACKAGE NOT ACTIVE.

9.4 CONTROL CHARACTERS

CONTROL CHARACTERS ARE USED TO GIVE THE OPERATOR THE

ABILITY TO PERFORM THE FOLLOWING FUNCTIONS.
NOTE: THE PROGRAM WILL RESPOND TO THE CONTROL
CHARACTER IN FIVE (5) SECONDS OR LESS.

CONTROL C

THIS WILL START THE LOADER THAT IS
IN LOCATION 7600.

CONTROL R

THIS WILL RESTART THE PROGRAM AND
REASK THE SWITCH REGISTER QUESTION AS
DESCRIBED IN SECTION 9.6.

CONTROL E

THIS WILL CONTINUE THE PROGRAM FROM
AN ERROR IF ALLOWED BY THE DIAGNOSTIC
OR FROM A WAITING STATEMENT.

CONTROL L

THIS WILL SWITCH THE TERMINAL MESSAGES
FROM THE DISPLAY TO A LINE PRINTER.
TO RESTORE THE MESSAGES ON THE TERMINAL
CONTROL L MUST BE TYPED AGAIN. IF
NO PRINTER IS AVAILABLE AND CONTROL L
IS TYPED THE RESULT WILL BE THAT THE
CONSOLE PACKAGE WILL WAIT FOR CONTROL C OR R.
THE CONTROL L WILL OUTPUT TO THE LINE
PRINTER AND THE PROGRAM WILL
ATTEMPT TO CONTINUE AS IF A CONTROL E
WAS TYPED IN.

CONTROL D

THIS WILL ALLOW THE ABILITY TO CHANGE
THE SWITCH REGISTER DURING PROGRAM
OPERATION. TYPING THIS CHARACTER WILL RESULT
IN AN INTERIGATION OF THE SWITCH REGISTER
QUESTION AS DESCRIBED IN SECTION 9.6.

CONTROL S

THIS WILL STOP PROGRAM EXECUTION AND WAIT IN A
LOOP FOR A CONTINUE. THE ONLY WAY TO CONTINUE
WILL BE TO TYPE A CONTROL Q, R OR C .
THIS IS A NONPRINTING CHARACTER.

CONTROL Q

THIS IS TO CONTINUE A PROGRAM AFTER A CONTROL
S IS TYPED. THIS IS A NONPRINTING CHARACTER.

9.5

WAITING MESSAGE

THE WAITING MESSAGE IS USED TO ALLOW THE OPERATOR TIME
TO MAKE A DECISION AS TO WHAT CONTROL CHARACTER
TO TYPE. THIS MESSAGE MAY APPEAR AT THE END
OF PASS MESSAGE IF THE HALT ON PASS BIT IS SET. THE CONTROL
CHARACTERS MAY NOW BE USED TO PERFORM THE NEEDED FUNCTION.

THE WAITING MESSAGE MAY BE PRINTED AFTER A ERROR MESSAGE IF THE HALT ON ERROR BIT IS SET. HERE AGAIN THE CONTROL CHARACTERS MAY BE USED. THE WAITING MESSAGE MAY BE PRINTED IF OPERATOR INTERVENTION IS REQUIRED.

9.6 SWITCH REGISTER MESSAGE

THIS MESSAGE IS USED TO SETUP THE PSEUDO SWITCH REGISTER BEFORE PROGRAM EXECUTION TAKES PLACE. THE SWITCH REGISTER IS SETUP WHEN THE FOURTH CHARACTER IS ENTERED OR A CARRIAGE RETURN IS TYPED

SR=0000 4000

UNDER SCORING INDICATES OPERATOR RESPONSE

9.7 END OF PASS

THE NORMAL PASS COMPLETE TIMEOUT AS DESCRIBED IN SECTION 5.5 IS USED.

9.8 ERRORS

THE STANDARD ERROR REPORTS AS DESCRIBED IN SECTION 5 OF THIS DOCUMENT WILL BE USED.

9.9 SWITCH REGISTER SETTINGS

THE STANDARD SWITCH SETTINGS AS DESCRIBED IN SECTION 3 OF THIS DOCUMENT WILL BE USED.

9.10 PARAMETER CONTROL WORDS

THE CONSOLE PACKAGE USES THE LOCATIONS 20 21 22 FOR THE FOLLOWING PURPOSES.

LOCATION 20
PSEUDO SWITCH REGISTER

LOCATION 21
HARDWARE IDENTIFIER 1

LOCATION 22

HARDWARE IDENTIFIER 2

LOCATION 0021

BIT ---	OCTAL VALUE -----	FUNCTION WHEN 0 -----	FUNCTION WHEN 1 -----
0	4000	USE PSEUDO SWITCHES	USE HARDWARE SWITCHES
1	2000	NO OPTION 1	HAS OPTION 1
2	1000	NO OPTION 2	HAS OPTION 2
3	400	NO 8A SIMULATOR	HAS 8A SIMULATOR
4	200	NO OPTION SIMULATOR	HAS OPTION SIMULATOR
5	100	NOT ON 8A XOR	ON 8A XOR
6	40	NOT PDP8-E TYPE CPU	PDP8-E TYPE CPU

7-11

8A MEMORY SIZE EX. 1K=00
2K=01
7K=06

32K=31

LOCATION 0022

BIT ---	OCTAL VALUE -----	FUNCTION WHEN 0 -----	FUNCTION WHEN 1 -----
0	4000	NOT ON ACT8A LINE	ON ACT 8A LINE
1	2000	NOT ON ACT 8E LINE	ON ACT 8E LINE
2	1000	NOT YET DEFINED	
3	400	DEACTIVE CONSOLE PACKAGE	ACTIVE CONSOLE PACKAGE

9.11

LOCATION CHANGES

THE FOLLOWING FIELD 1 LOCATIONS CAN BE CHANGED TO MEET THE SPECIFIC NEED FOR MODIFICATION OF THE DIAGNOSTIC.

0246

IS THE LOCATION FOR THE VALUE OF THE NUMBER OF PROGRAM PASSES NEED TO PRINT THE END OF PASS MESSAGE.

1037

IS THE LOCATION SET FOR THE NUMBER OF FILLER CHARACTERS AFTER A CRLF SET TO FOUR (4)

10.

APT-8 HOOKS

10.1 DESCRIPTION

TWO INTERFACES HAVE BEEN PROVIDED WHICH ALLOW THIS DIAGNOSTIC TO RUN UNDER THE STANDARD APT-8 SYSTEM. THESE INTERFACES ARE:

1. TIMING INTERFACE
2. ERROR INTERFACE

EACH WILL BE EXPLAINED IN DETAIL.

10.2 SETUP

THE FOLLOWING INFORMATION MUST BE INDICATED DURING THE INITIAL PROGRAM START UP.

1. SINGLE OR MULTIPLE DRIVE TESTING.
2. DRIVE OR DRIVES TO BE TESTED.
3. DIAGNOSTIC RUNNING UNDER APT-8.
4. THE AMOUNT OF MEMORY IN 1K INCREMENTS.

IF SINGLE DRIVE TESTING BIT 5 OF ADDRESS 22 MUST BE SET TO A ONE (1) WITH BITS 6-11 CONTAINING THE DRIVE TO BE TESTED. IF MULTIPLE DRIVES ARE TO BE DONE BIT MUST BE SET TO A ZERO (0) AND BIT 6-11 CONTAINING THE HIGHEST NUMBER DRIVE TO BE TESTED. WHEN MULTIPLE DRIVE TESTING ONLY A SPECIFIC NUMBER OF DRIVES CAN BE INDICATED, THE PROGRAM ASSUMES THE DRIVES ARE TO BE DONE BEGINNING WITH DRIVE ZERO (0) AND FINISHING WITH THE HIGHEST DRIVE INDICATED. IF MULTIPLE DRIVES OTHER THAN CONSECUTIVELY NUMBERED DRIVES BEGINNING WITH DRIVE ZERO (0) ARE TO BE DONE, THEY MUST BE DONE AS SINGLE DRIVES AND TESTED INDEPENDANTLY.

THE PROGRAM ALLOWS DRIVES 0-7 TO BE SELECTED. USER SHOULD NOT EXCEED 0-3 DRIVES FOR THE MK8E CONTROL.

BIT ZERO OF ADDRESS 22 MUST BE SET TO A ONE TO INDICATE THAT THE PROGRAM WILL RUN UNDER APT-8.

NOTE: IT SHOULD BE NOTED AT THIS TIME THAT WHILE RUNNING UNDER APT-8 THE HARDWARE SWITCH REGISTER IS INOPERATIVE. ONLY THE HALT AND SINGLE STEP SWITCH WILL EFFECT THE PROGRAM RUN.

AMOUNT OF MEMORY IN 1K INCREMENTS SHOULD BE STORED IN BITS 7-11 OF LOCATION 21. AN ADDITION OF 1 TO THE NUMBER OF BITS IN 7-11 INCREASES MEMORY SIZE BY 1K. EX. 4K=3,8K=7. REMEMBER TO RETAIN STATUS OF BITS WHEN MODIFYING LOCATION 21.

APT-8 INTERFACES:

10.3.1. TIMING

APT-8 IS NOTIFIED OF PROGRAM RUN BETWEEN .2 SEC AND 2.0 SEC ON A 1.2 MICROSECOND MEMORY CYCLE. THIS WILL ALLOW THE DIAGNOSTIC TO RUN WITHOUT CAUSING AN APT-8 TIMEOUT ERROR IF THE DIAGNOSTIC IS TO BE RUN ON ON THE SLOWER MOS MEMORY.

10.3.2. ERRORS

ONLY THE DRIVE IN ERROR IS REPORTED TO APT-8 SYSTEM.
SYSTEM. ERRORS WHICH CAUSE A PROGRAMMED HALT CAUSE A TIMEOUT
ERROR. IF A PROGRAMMED HALT SHOULD OCCUR, THE ERROR PC WILL
APPEAR IN THE AC ON THE DEVICE UNDER TEST. PROGRAMMED HALTS
ARE EXPLAINED EARLIER IN THIS DOCUMENT.

11. PROGRAM LISTING

```

/RK8E/RK8L DATA RELIABILITY PROGRAM: MU=88-DHRKC=M
/MAINDEC=88-DHRKC=M-L
/COPYRIGHT 1972,1975,1976,1977 DIGITAL EQUIP. CORP.
/MAYNAD, MASS. 01754
/
0001 / FIELD 1
/
/CONSOL SRC=V2-R8= CONSOLE PACKAGE
/
/THE PROGRAM SHOULD CHECK FOR A CONTROL CHARACTER FROM THE TERMINAL
/EVERY FIVE(5) SECONDS OR SOONER.

/Locations THAT NEED TO BE SET UP FOR USING THE CONSOLE PACKAGE.

/CNTVAL IN XCBPASS THIS LOCATION DETERMINES THE NUMBER OF
/PROGRAM COMPLETIONS THAT ARE NEEDED BEFORE THE PASS MESSAGE IS TYPED
/THE VALUE SHOULD PUT THE PASS MESSAGE OUT IN THE RANGE OF 1 TO 5 MINUTES.
/THIS SHOULD BE A POSITIVE NUMBER.

/CBSTRT THIS IS FOUND IN CNTRL ROUTINE CONTROL R PART
/IT IS THE RETURN WHEN CONTROL R IS ENTERED (RESTART PROGRAM)
/THE RETURN JUMPS TO XDO8H WHICH CONTAINS CBSTRT SO PUT THE LABEL CBSTRT
/WHERE YOU WANT TO RESTART THE PROGRAM.

/SETUP1 IN XCBERR THIS IS THE MASK BIT FOR HALT ON ERROR
/PLACE THE CORRECT BIT IN THIS LOCATION FOR HALTING ON ERRORS.

/SETUP2 IN XCBPASS THIS IS THE MASK FOR HALT A END OF PASS.

/THE CALL TABLE IS A CONDITIONAL ASSEMBLY.
/TO ASSEMBLE THE CALL REMOVE THE / BEFORE CONSOL=0.
/IN COMBINING THE CONSOL PACKAGE TO A DIAGNOSTIC.
/THE CALL TABLE IS TO BE AT THE BEGINNING OF A PROGRAM.
    
```

```

0000 CONSOL=0
0001 PSKF= 0001
0002 PCLF= 0002
0003 PSKE= 0003
0004 PSTD= 0004
0005 PSIE= 0005
0006 STF= 0004
0007 ACL= 7701
0008 CAF= 0007
0009 MQL= 7421
0010 MQA= 7501
/
0020 /*20
/
0020 0000 F15WR, 0
0021 4000 F10P1, 4000
    
```

```

0022 0000 F10P2, 0
/
IFDEF CONSOL <

0024 *24

4424 C0PASS= JMS I .
0024 0200 XC0PAS /C0 PASS COMPLETION ROUTINE
4425 C0CKSW= JMS I .
0025 0262 XC0SW /CHECK SW REG SETTING
4426 C0TTYI= JMS I .
0026 0272 XC0TTY /FETCH CONSOL CHAR
4427 C0CNTR= JMS I .
0027 0400 XC0CNT /CHECK FOR CONTROL CHAR
4430 C0PRNT= JMS I .
0030 0303 XC0PNT /C0 PRINT A BUFFER
4431 C0SWIT= JMS I .
0031 0656 XC0PSW /SET UP PSEUDO SW, REG
4432 C0OCTA= JMS I .
0032 1000 XC0OCT /CONVERT TO ASCII AND PRINT
4433 C0CRLF= JMS I .
0033 1023 XC0CRL /DO A CARRIAGE RETURN+ LINE FEED
4434 C0ECHO= JMS I .
0034 1043 XC0ECH /CHECK INPUT CHAR
4435 C0TYPE= JMS I .
0035 1077 XC0TYP /C0 PRINT ONE CHAR
4436 C0ERR= JMS I .
0036 1207 XC0ERR /C0 ERROR HANDLER
4437 C0INQU= JMS I .
0037 0635 XC0ING /LOOK FOR OPERATOR INTERVENTION
4440 C0CKPA= JMS I .
0040 1041 XC0CKP /CHECK IF CONTROL CHAR
4441 C0PAUS= JMS I .
0041 0337 XC0PAU /IF CONSOL PACKAGE RETURN CALL PLUS ONE
/IF NOT USING CONSOL REPLACE CALL WITH
/A HLT AND THEN GO TO THE HALT

/*****
/*20 /PSEUDO SWITCH REGISTER
/*21 /HARDWARE INDICATORS
/4000=USE FRONT PANEL SWITCH REGISTER
/0000=USE THE PSEUDO SWITCH REGISTER LOC,20
/*22 /SYSTEM CONFIGURATION
/400=CONSOL PACKAGE SET ACTIVE
/000=CONSOL PACKAGE SET DEACTIVE
/*23 /RESERVED FOR FUTURE USE
/
0200 /*200
/
/*****
/C0PASS
    
```

```

/THIS IS CALLED AT THE END OF EACH PROGRAM COMPLETION
/THE VALUE OF** CNTVAL** WILL BE DETERMINED BY THE TIME IT TAKES
/THE PROGRAM TO COMPLETE THIS MANY CPASS TO BE IN THE 1 TO 4 MINUTE
/RANGE
/
/      CPASS=JMS      XCSPAS
/EX. OF CALL      CPASS
/
/      JMP      HLT      /HALT IF NON CONSOL PACKAGE
/              START1   /CONTINUE RUNNING THIS PROGRAM

```

```

/RETURN TO LOCATION CALL PLUS ONE WITH THE AC=0 IF NON CONSOL PACKAGE AND HLT
/IF CONTINUE TO RUN THEN RETURN TO CALL PLUS2 AC=0
/THE LOCATION SETUP2 IS THE MASK BIT FOR THE HALT AT END OF PASS
/CHECK THAT IT IS CORRECT FOR THE CURRENT PROGRAM

```

/CALLS USED BY XCSPAS ARE CHKCLA-XC8CNLF-XC8OCTA-XC88W-XC8PNT-XC8INO-

```

0200 0000      XCSPAS, 0
0201 7200      CLA
0202 4777*     JMS      CHKCLA      /IS WORD 22 BIT 3 ACTIVE CONSOLE?
0203 5212     JMP      DOPACK      /IS CLASSIC
0204 4776*     JMS      C8GET      /GET REGISTERS.
0205 4262     JMS      XC88W      /DEACTIVE CONSOL CHECK SR SETTING
0206 0375     AND      4000      /FOR HALT ON END OF CPASS
0207 7640     SZA      CLA      /1= HALT 0 CONTINUE
0210 5600     JMP      I      XCSPAS /GO TO HALT
0211 5230     JMP      C8BY1     /CONTINUE ON RUNNING PROGRAM
0212 4232     DOPACK, JMS      CKCOUT   /CLASS CHECK CPASS COUNT
0213 5230     JMP      C8BY1     /CPASS COUNT NOT DONE REDO PROGRAM
0214 2250     ISZ      PASCNT     /CPASS COUNT DONE SET CPASS COUNT
0215 4774*     JMS      XC8CRLF    /
0216 4303     JMS      XC8PNT     /CPHNT BUFFER
0217 0253     MESPAS
0220 1250     TAD      PASCNT     /
0221 4773*     JMS      XC8OCTA    /GET NUMBER
0222 4774*     JMS      XC8CRLF    /CONVERT IT TO ASCII
0223 4776*     JMS      C8GET      /DO A CARRIAGE RETURN
0224 4262     JMS      XC88W      /GET REGISTERS.
0225 0375     SETUP2, AND      4000   /CHECK A HALT AT END OF CPASS
0226 7640     SZA      CLA      /MASK BIT
0227 4772*     JMS      XC8INO     /HALT #1 NO SKIP CONTINUE #0
0230 2200     C8BY1, ISZ      XC88W   /STOP PROGRAM EXECUTION-LOOK FOR INPUT
0231 5600     JMP      I      XCSPAS /BUMP RETURN
0232 0000     CKCOUT, 0
0233 1251     TAD      DOSET      /CHECK IF SET UP NEEDED
0234 7640     SZA      CLA      /0=SET UP CPASS COUNT VALUE
/1=CPASS COUNT VALUE OK
0235 5242     JMP      NOSET      /CPASS COUNT VALUE OK
0236 1252     TAD      CNTVAL     /SET COUNT VALUE FOR THIS PROG
0237 7040     CMA
0240 3247     DCA      DOCNT     /SET TO NEGATIVE
0241 2251     ISZ      DOSET      /STORE IN HERE
0242 2247     NOSET, ISZ      DOCNT  /INDICATE VALUE SET UP
0243 5230     JMP      C8BY1     /COUNT THE NUMBER OF PASSES
/EXIT FOR ANOTHER PASS

```

```

0244 3251     DCA      DOSET      /SET TO C8PNT CPASS
0245 2232     ISZ      CKCOUT     /BUMP RETURN FOR
0246 5632     JMP      I      CKCOUT /CPASS C8TYPE OUT
0247 0000     DOCNT, 0
0250 0000     PASCNT, 0
0251 0000     DOSET, 0
0252 0000     CNTVAL, 0
0253 0410     MESPAS, TEXT      "DMRKC PASS "
0254 2213
0255 0310
0256 4040
0257 2001
0260 2323
0261 4000

```

/*****

/C8CKSW

```

/THIS ROUTINE CAN BE USED INPLACE OF A READ THE SWITCHES LAS.
/ROUTINE THAT WILL CHECK WHERE TO READ THE
/08 SWITCHES FROM IE, FROM PANEL OR PSEUDO SWITCH REGISTER
/THE SELECTION IS DETERMINED BY THE STATE OF BIT 0 IN LOCATION 21.

```

```

/C8CKSW=      JMS XC88W
/EX.      JMS XC88W      /READ THE C8SWIT REGISTER
/              /RETURN WITH THE CONTENTS OF SWITCH REGISTER

```

/RETURN TO NEXT LOCATION FOLLOWING CALL WITH THE AC= TO VALUE OF C8SWIT SETTING

/CALLS USED ARE-XC8CKPA-

```

0262 0000      XC88W, 0
0263 4771*     JMS      XC8CKPA    /GO CHECK THE IF ANY CONTRL
0264 7000     NOP
0265 1021     TAD      21      /GET MU FUN INDICATOR
0266 7710     SPA      CLA      /CHECK IF FROM PANEL 4000
0267 7614     TAD      7614     /00 LAS AND SKIP GET FROM PANEL WITH LAS
0270 1020     TAD      20      /PSEUDU SWITCH
0271 5662     JMP      I      XC88W  /EXIT WITH STATUS BIT IN AC.

```

/*****

/C8TTYI

```

/THIS ROUTINE WILL LOOK FOR A INPUT FROM THE TERMINAL
/AND REMOVE ANY PARITY BITS, THEN MAKE IT 8 BIT ASCII.
/
/      C8TTYI= JMS XC8TTYI
/EX.      JMS XC8TTYI      /READ CHAR FROM THE CONSOL DEVICE
/              /RETURN TO CALL PLUS ONE AC CONTAINS THE CHAR

```

/CALLS USED-NONE- BUT C0CHAR IS OFF PAGE AND IN ROUTINE CALLED XC0ECHO

```

/
/
0272 0000 XC0TTY, 0
0273 0031 KSF /LOOK FOR KEYBOARD FLAG
0274 5273 JMP ,=1
0275 0036 KRB /GET CHAR
0276 0370 AND (177 /MASK FOR 7 BITS
0277 1367 TAD (200 /ADD THE EIGHTH BIT
0300 3766 DCA C0CHAR /STORE IT
0301 1766 TAD C0CHAR
0302 5672 JMP I XC0TTY /EXIT

```

/C0PRT

/THIS ROUTINE WILL TYPE THE CONTENTS OF THE C0 PRINT BUFFER, THE LOCATION /OF THE BUFFER WILL BE IN THE ADDR FOLLOWING THE CALL, PRINTING OF THE BUFFER /WILL STOP WHEN A 00 CHAR IS DETECTED, CHAFACTERS ARE PACKED 2 PER WORD.

/ C0PRT= JMS XC0PNT

/EX. JMS XC0PNT /C0PRT THE CONTENTS OF THE FOLLOWING BUFFER
MESS77 /LOCATION OF C0PRT BUFFER

/C0PRT WILL USE THE LOCATION FOLLOWING THE CALL AS THE POINTER FOR THE /C0PRT ROUTINE, RETURN TO CALL PLUS TWO WITH AC= 0

/CALLS USED ARE-XC0TYPE-XC0PNT

```

0303 0000 XC0PNT, 0
0304 7300 CLA CLL
0305 1703 TAD I XC0PNT /GET C0PRT BUFFERS STARTING LOCATION
0306 3336 DCA PT0TOR /STORE IN PT0TOR
0307 2303 ISZ XC0PNT /BUMP RETURN
0310 1736 C0D01, TAD I PT0TOR /GET DATA WORD
0311 0365 AND (7700 /MASK FOR LEFT BYTE
0312 7450 SNA /CHECK IF 00 TERMINATE
0313 5703 JMP I XC0PNT /EXIT
0314 7500 SMA /IS AC MINUS
0315 7020 CML /MAKE CHAR A 300 AFTER ROTATE
0316 7001 IAC /MAKE CHAR A 200 AFTER ROTATE
0317 7012 RTR
0320 7012 RTR
0321 7012 RTR /PUT CHAR IN BITS 4-11 MAKE IT 8 BIT ASCII
0322 4704 JMS XC0TYPE /C0PNT IT ON CONSOLE
0323 1736 TAD I PT0TOR /GET DATA WORD

```

```

0324 0363 AND (0077 /MASK FOR RIGHT BYTE
0325 7430 SNA /CHECK IF 00 TERMINATOR
0326 5703 JMP I XC0PNT //EXIT
0327 1362 TAD (3740 /ADD FUDGE FACTOR TO DETERMINE IF 200
0330 7500 SMA /OR 300 IS TO BE ADD TO CHAR
0331 1361 TAD (100 /ADD 100
0332 1360 TAD (240 /ADD 200
0333 4704 JMS XC0TYPE /C0TYPE ONLY BITS 4-11
0334 2336 ISZ PT0TOR /BUMP POINTER FOR NEXT WORD
0335 5310 JMP C0D01 /DO AGAIN
0336 0000 PT0TOR, 0 /STOP FOR C0PRT BUFFER

```

/C0PAUS
/THIS ROUTINE WILL CHECK IF THE CONSOL PACKAGE IS ACTIVE, IF ACTIVE /IT WILL RETURN TO CALL PLUS ONE AC= 0, AND DO THAT INSTRUCTION, /IF THE CONSOL PACKAGE IS NOT ACTIVE THE CALL WILL BE REPLACED /WITH A 7402 HALT AND THEN RETURN TO THE HALT.

/ C0PAUS= JMS XC0PAU

/EX. JMS XC0PAUS /CHECK IF ON ACTIVE CONSOL IF NOT HALT HERE
ANYTHING /RETURN HERE IF ON ACTIVE CONSOL

/CALLS USED ARE-CHKCLA-

```

0337 0000 XC0PAU, 0
0340 7300 CLA CLL
0341 4777 JMS CHKCLA /CHECK LOC 22 BIT 3 CONSOLE BIT
0342 5350 JMP C0D03 /GO DO CONSOL PART RETURN CALL+1
0343 7040 CMA /DEACTIVE CONSOLE PACKAGE PUT HLT IN CALL
0344 1337 TAD XC0PAU /GET CURRENT RETURN ADDR
0345 3337 DCA XC0PAU /SET UP RETURN
0346 1357 TAD (7402 /GET CODE FOR HLT
0347 3737 DCA I XC0PAU /PUT HLT IN CALL LOCATION
0350 5737 C0D03, JMP I XC0PAU /GO TO HALT OR RETURN TO NEXT LOCATION

```

```

0357 7402
0360 0240
0361 0100
0362 3740
0363 0077
0364 1077
0365 7700
0366 1075
0367 0200
0370 0177
0371 1041

```

0372 0635
0373 1000
0374 1023
0375 0400
0376 0624
0377 1200
0400 0400

PAGE

/C0CNTM

/THIS ROUTINE WILL CHECK FOR THE PRESENCE OF CONTROL CHARACTERS

/IT WILL CHECK FOR THE FOLLOWING CHAR C-R-U-L-S

/ C0CNTM= JMS XC0CNT

/EX.

JMS XC0CNTR
JMP ANYTHING
JMP ANYTHING

/CHECK FOR CONTROL CHARACTER

/LOC FOLLOWING CALL IS FOR CONTINUING THE PROGRAM

/LOC. IS FOR RETURN IF INMODE SET AND NOT CNTRL CHAR

/RETURN IS TO CALL PLUS ONE IF CONTINUE

/RETURN IS TO CALL PLUS TWO IF INMODE SET AND NOT CONTROL CHAR

/RETURN IS TO CALL PLUS TWO IF INMODE IS NOT SET AND NO

/CONTROL CHAR +, THIS WILL PRINT THE CHARACTER AND A ?

/CLEAR THE AC AND RETURN CALL+2.

/CALLS USED ARE=CHKCLA=XC0TYPE=XC0CRLF=C0GET=UPARON=XC0TYI=XC0PSW=

/

/

/

0400 0000
0401 3777
0402 4776
0403 5206
0404 1777
0405 5600
0406 6004
0407 3775
0410 7501
0411 3774
0412 3255
0413 1257
0414 3256
0415 1656
0416 7450
0417 5226
0420 1773
0421 7650
0422 5243
0423 2255
0424 2256
0425 5215
0426 1772
0427 7640
0430 5240

XC0CNT, 0

DCA ACSAVE
JMS CHKCLA
JMP +3
TAD ACSAVE
JMP I XC0CNT
GTF
DCA FLSAVE
MGA
DCA MGSAVE
DCA INDEXA
TAD XTABLA
DCA GETDAT
REDDA, TAD I GETDAT
SNA
JMP DONEA
TAD C0CHAR
SNA CLA
JMP GOITA
ISZ INDEXA
ISZ GETDAT
REDDA
DONEA, TAD INMODE
SZA CLA
JMP EXITA

/SAVE THE AC
/CHECK LOC,22 BITS FOR CONSOLE BIT
/ON ACTIVE CONSOLE
/DEACTIVE CONSOLEGET AC FOR RETURN
/EXIT NOT UN ACTIVE CONSOLE
/SAVE THE MQ
/SET DISPLACEMENT INTO TABLE 0
/GET ADDRS OF TABLE A
/CONTAINS POINTER TO CONTROL CHAR
/GET CONTROL CHAR FROM TABLE
/CHECK FOR A 0 END OF TABLE
/END OF TABLE NO CONTROL CHAR
/COMPARE CHAR TO CONTROL CHAR
/0 IF MATCH
/MATCH
/NO MATCH NOT END OF TABLE REDD
/BUMP INDEX FOR EXIT WHEN CONTROL FOUND
/BUMP GETDAT FOR COMPARE OF NEXT CNTRL CHAR.
/CHECK IF PROGRAM EXPECTS CHAR
/1=CHAR EXPECTED 0= NO CHAR EXPECTED
/CHAR EXPECTED

0431 1773
0432 4771
0433 1370
0434 4771
0435 4767
0436 2200
0437 5600
0440 2200
0441 1773
0442 5600
0443 1773
0444 1366
0445 3773
0446 1260
0447 1255
0450 3254
0451 1654
0452 3254
0453 5654
0454 0000
0455 0000
0456 0000
0457 0461
0460 0471
0461 7575
0462 7564
0463 7557
0464 7556
0465 7555
0466 7573
0467 7574
0470 0000

TAD C0CHAR
JMS XC0TYPE
TAD (277)
JMS XC0TYPE
JMS XC0CRLF
ISZ XC0CNT
JMP I XC0CNT
REDDA, ISZ XC0CNT
TAD C0CHAR
JMP I XC0CNT
TAD C0CHAR
TAD (100)
DCA C0CHAR
TAD XTABLA
TAD INDEXA
DCA G0TOA
TAD I G0TOA
DCA G0TOA
JMP I G0TOA
G0TOA, 0000
INDEXA, 0000
GETDAT, 0000
XTABLA, TABLA
XTABLA, TABLB
TABLA, 7575
7564
7557
7556
7555
7573
7574
0000
TABLB, CNTRL0
CNTRL1
CNTRL2
CNTRL3
CNTRL4
CNTRL5
CNTRL6

/GET CHAR= NOT CONTROL+ NOT EXPECTED
/C0PNTM CHAR
/GET CODE FOR "2"
/0UMP RETURN
/EXIT CALL+2
/0UMP RETURN FOR MAIN PROGRAM CHECK OF CHAR
/PUT CHAR IN AC.
/EXIT
/GET THE CONTENTS OF CHAR
/ADD 100 TO FORM A GOOD ASCII CHARACTER
/RESTORE COFFECT CHAR
/GET START OF TABLE B
/GET NUM P \ INTO TABLE
/STORE IT
/GET THE ROUTINE STARTTING ADDRESS
/STORE IT IN HERE
/GOTO CONTROL CHAR ROUTINE
/ADD OF CNTRL ROUTINE TO EXECUTE
/DISPLACEMENT INTO CNTRL TABLE
/LOCATION OF ADDRS OF CONTROL CHAR.
/ADDNS OF TABLEA
/ADDNS OF TABLEB
/CNTRL C BACK TO MONITOR 203
/CNTRL L SWITCH ERROR PRINTTING DEVICE 214
/CNTRL Q START DISPLAYING CHAR, AGAIN 221
/CNTRL R BACK TO BEGINNING OF PROGRAM 222
/CNTRL S STOP SENDING CHAR TO DISPLAY WAIT FOR CNTRL Q 223
/CNTRL E CONTINUE WITH PROGRAM 205
/CONTROL D CHANGE SWITCH REGISTER ON FLY

0471 0551
0472 0537
0473 0500
0474 0511
0475 0521
0476 0545
0477 0600

TABLB, CNTRL0
CNTRL1
CNTRL2
CNTRL3
CNTRL4
CNTRL5
CNTRL6

/CONTROL Q
/START SENDING CHAR. TO THE DISPLAY
/THIS WILL RETURN CONTROL TO CALL THAT WAS SET BY
/THE CALL FOR CONTROL S.

0500 3772
0501 1335
0502 7640
0503 5306
0504 4765
0505 5600
0506 3335
0507 4765
0510 5736

CNTRL0, DCA INMODE
TAD C0SETS
SZA CLA
JMP BYMETR
JMS C0GET
JMP I XC0CNTM
BYMETR, DCA C0SETS
JMS C0GET
JMP I C0METR

/SET SUFT FLAG FOR UNEXPECTED CHAR
/CHECK IF CONTROL S TYPED IN
/CONTROL S TYPED IN
/NO CONTROL S TYPED PREVIOUSLY
/LEAVE VIA CNTR ENTRY ADDRESS
/CLEAR THE SUFT FLAG
/RESTORE REGISTERS
/EXIT TO ADDRESS SET BY CONTROL S

```

/
/CONTRL R
/GO TO THE QUESTION C08SWIT
0511 3764' CNTRLR, DCA TTYLPT /CLEAR THE TYPE FLAG SET TO TTY
0512 3335 DCA C08ETS /CLEAR SOFT FLAG FOR CNTRL S
0513 3772' DCA INMODE
0514 4763' JMS UPAROW /PRINT THE " AND C08CHAR
0515 3762' C08Y4, DCA C08SWT /CLEAR FLAG FOR CNTRL D OR R
0516 6203 CIF C0F 0
0517 5720 JMP I XDUSW /GO TO ADUMS OF C08SWIT
0520 0200 XDUSW, BGN /D08W IS LABEL FOR C08SWIT QUESTION
/
/CONTRL S
/STOP SENDING CHAR. TO DISPLAY UNTIL A "Q" IS RECEIVED
/
0521 1335 CNTRLR, TAD C08ETS /IF1 DO NOT STORE IN C08RETR
0522 7640 SZA CLA
0523 5327 JMP C0D07 /DONT SET UP C08RETR
0524 7001 IAC /MAKE RETURN CALL PLUS 2
0525 1200 TAD XC0CNT /GET RETURN FOR THIS CALL
0526 3336 DCA C08RETR /STONE IT HERE FOR USE BE CNTRL Q
0527 2335 C0D07, I0Z C08ETS /SET FLAG TO SAVE CALL
0530 4761' JMS XC0TTYI /LOOK FOR THE INPUT
0531 4765' JMS C0GET /GEI REGISTERS
0532 4200 JMS XC0CNTR /CHECK FOR THE CONTROL CHAR
0533 7200 CLA
0534 5321 JMP CNTRLR /IF NOT A CNTRL Q W C REASK
0535 0000 C08ETS, 0
0536 0000 C08RETR, 0
/
/SWITCH OUTPUT FROM ONE OUTPUT DEVICE TO ANOTHER- THE TWO OUTPUTS ARE THE
/CONSOLE AND THE PRINTER WITH DEVICE CODE 66.
/
0537 1764' CNTRLR, TAD TTYLPT /GET PRESENT C08SWIT INDICATOR
0540 7000 CMA /COMPLEMENT IT
0541 3764' DCA TTYLPT /STON NEW C08SWIT
0542 4763' JMS UPAROW /C08PNT " AND CHAR ON NEW DEVICE
0543 4765' JMS C0GET /RESTORE THE REGISTERS
0544 5000 JMP I XC0CNT /EXIT
/
/CONTRL E
/CONTINUE RUNNING FROM A INQUIRE OR ERROR
/
0545 4763' CNTRLR, JMS UPAROW /PRINT THE CONTROL CHAR
0546 3762' DCA C08SWT /CLEAR ENTRY FLAG.
0547 4765' JMS C0GET /GET THE REGISTERS
0550 5000 JMP I XC0CNT /RETURN TO CALL PLUS ONE
/
/CONTRL C

```

```

/RETURN TO MONITOR CONTROL C
0551 3764' CNTRLR, DCA TTYLPT /CLEAR THE LPT FLAG TO PRINT ON DISPLAY
0552 3762' DCA C08SWT /CLEAR ENTRY FLAG.
0553 4763' JMS UPAROW /C08PNT " AND LETTER IN CHAR
0554 6203 C0F CIP /GO TO 0 FLD
0555 6007 CAF /CLEAR THE WORLD
0556 5760 JMP I (7600) /GO TO DIAGNOSTIC MONITOR
/*****
/
/
0560 7600
0561 0272
0562 0745
0563 0615
0564 1121
0565 0624
0566 0100
0567 1023
0570 0277
0571 1077
0572 1076
0573 1075
0574 1346
0575 1347
0576 1200
0577 1345
0600 PAGE
/
/CONTRL D
/CHANGE THE SWITCH REGISTER ANYTIME CNTRL D AND RETURN TO
/THE PROGRAM RUNNING.
/
0600 4215 CNTRLR, JMS UPAROW
0601 1213 TAD C08ETD /CHECK IF THE RETURN ADDR IS SAFE
0602 7640 SZA CLA
0603 5207 JMP C0D011 /DO NOT CHANGE THE RETURN ADDR
0604 1777' TAD XC0CNT /GET THE RETURN ADDR AND SAVE IT
0605 3214 DCA C08ETD /SAVE THE RETURN HERE
0606 2213 I0Z C08ETD /INDICATE RETURN SAVED DONT DISTROY
0607 4256 JMS XC0PSW /GU CHANGE THE SWITCH REGISTER
0610 3213 DCA C08ETD /CLEAR THE FLAG
0611 4224 JMS C0GET /RESTORE THE AC MU LINK ETC
0612 5614 JMP I C08ETD /RETURN TO THE PROGRAM
/
0613 0000 C08ETD, 0
0614 0000 C08RETR, 0
/
/THIS WILL TYPE A UP ARROW AND THE CHAN IN C08CHAR.
0615 0000 UPAROW, 0 /C08PNT THE "" AND THE CHAN C08TYPED IN

```

```

0616 1376      TAD      (336      /CODE FOR "
0617 4775*    JMS      XC0TYPE
0620 1774*    TAD      C0CHAR      /C0TYPE THE CHAR
0621 4775*    JMS      XC0TYPE
0622 4773*    JMS      XC0CRLF
0623 3615      JMP I    UPAROH      /EXIT

```

```

0624 0000      C0GET, 0
0625 7200      CLA      /
0626 1772*    TAD      M0SAVE
0627 7421*    MQL      /RESTORE M0
0630 1771*    TAD      FL0SAVE
0631 7004      RAL      /RESTORE THE LINK
0632 7200      CLA      /
0633 1770*    TAD      AC0SAVE
0634 3624      JMP I    C0GET      /GET THE REGISTERS

```

```

/C0INQU
/C0INQU ROUTINE WILL PRINT A WAITING
/AND THE PROGRAM IS EXPECTING A CONTROL CHAR INPUT
/IF CONTINUE FROM CONTROL CHAR RETURN IS CALL PLUS ONE
/IF NO CONTROL CHAR ENTERED THEN WAITING IS REPRINTED
/AND PROGRAM WAITS FOR A CONTROL CHAR AGAIN.

/      C0INQU =      JMS XC0ING

/EX.   JMS      XC0ING      /C0 WILL PRINT A WAITING AND WAIT FOR INPUT
/      OD ANYTHING      /RETURN IS CALL PLUS ONE AC = 0 CONTINUE

/CALLS USED ARE=CHKCLA=XC0PNT=XC0TYI=C0GET=XC0CNTR=

```

```

0635 0000      XC0ING, 0
0636 7300      CLA CLL
0637 4767*    JMS      CHKCLA      /CHECK LOC 22 BIT 3 CONSOLE BIT
0640 7410      SKP
0641 3635      JMP I    XC0ING      /ACTIVE CONSOLE PACKAGE
0642 4766*    JMS      XC0PNT      /NOT CONSOLE LEAVE
0643 0651      WATHEB
0644 4765*    JMS      XC0TTYI      /INQUIRE WAITING
0645 4224      JMS      C0GET      /GET CHARACTER
0646 4777*    JMS      XC0CNTR      /CHECK IF CONTROL CHARACTER
0647 3635      JMP I    XC0ING      /EXIT AND CONTINUE
0650 5236      JMP      XC0ING*1      /REASK
0651 2701      WATHEB, TEXT "WAITING "
0652 1124
0653 1116

```

```

0654 0740
0655 0000

```

```

/C0SWIT
/ROUTINE WILL CHECK IF CONSOLE IS ACTIVE IF IT IS ACTIVE DISPLAY
/SW QUESTION. IN NOT ACTIVE IT WILL NOT PRINT THE SW QUESTION BUT
/RETURN TO CALL PLUS ONE AC=0.
/C0SWIT WILL SET UP THE PSEUDO SWITCH
/REGISTER WITH THE NEW DATA ENTERED

/      C0SWIT =      JMS XC0PSW

/EX.   JMS      XC0PSW      /SET UP PSEUDO C0SWIT REGISTER IF
/      /ON THE CONSOLE PACKAGE, RETURN IS CALL PLUS ONE AC = 0

/CALLS USED ARE=CHKCLA=XC0PSW=XC0PNT=XC0CTA=XC0TYPE=

```

```

0656 0000      XC0PSW, 0
0657 4767*    JMS      CHKCLA      /CHECK LOC 22 BIT 3 CONSOLE BIT
0660 7410      SKP
0661 3656      JMP I    XC0PSW      /ACTIVE CONSOLE
/DEACTIVE CONSOLE PACKAGE
/RETURN WITHOUT ASKING PSEUDO SWITCH
0662 1345      TAD      C0SWST      /IS THE SOFT FLAG SET FOR SWITCH?
0663 7640      SZA CLA
0664 3764*    JMP      C0BY4      /SKIP IF ONE ENTRY AT A TIME OK
0665 2345      ISZ      C0SWST      /SECOND ENTRY WITH OUT A EXIT GO TO SW QUESTION
0666 4766*    JMS      XC0PNT      /FIRST ENTRY SET FLAG
0667 0747      MESA
0670 1020      TAD      20
0671 4763*    JMS      XC0CTA      /GET CONTENTS OF SW
/CONVERT IT TO ASCII
0672 1362      TAD      (40
0673 4775*    JMS      XC0TYPE      /GET SPACE
0674 2761*    ISZ      INMODE      /SET FLAG FOR CHAR EXECTED
0675 4760*    JMS      XC0ECHO      /LOOK FOR INPUT
0676 4315      JMS      TSTCHA      /NOT CONTROL TEST IT IS LEGAL
0677 1774*    TAD      C0CHAR      /STORE NEW CHAR IN SW REG
0700 3020      DCA      20

0701 1357      TAD      (=3
0702 3346      DCA      TPCNT
0703 4760*    JMS      XC0ECHO      /GET NEXT CHAR
0704 4315      JMS      TSTCHA      /CHECK IF CR+ GOOD CHAR
0705 1020      TAD      20
0706 7106      RTL CLL
0707 7004      RAL
0710 1774*    TAD      C0CHAR      /GET CHAR+ ADD IT TO PREVIOUS CONTENTS
0711 3020      DCA      20
0712 2346      ISZ      TPCNT
0713 5303      JMP      GETCH1      /BUMP COUNT
0714 5342      JMP      ENDIT      /JMP BACK+ GET NEXT CHAR
/END 4 CHAR C0TYPED IN

```

```

0715 0000 TSTCHA, 0
0716 7041 CIA /CMPL CHAR IN AC
0717 1356 TAD (215 /TEST IF IT IS A CARRIAGE RETURN
0720 7650 SNA CLA /SKIP IN NOT CR,
0721 5342 JMP ENDIT /HAB CARRIAGE RETURN
0722 1774 TAD C0CHAR /NOT CR, GET CHAR
0723 1355 TAD (-260 /CHECK IF IT IS IN RANGE
0724 7710 SPA CLA /IF NOT POSITIVE CBERR CHAR SMALLER THEN 260
0725 5336 JMP ERR1 /CBERR= CHAR TOO SMALL
0726 1774 TAD C0CHAR /GET CHAR
0727 1354 TAD (-270 /GET A=270+ CHECK IF IT IS LARGER THEN 7
0730 7700 SNA CLA /SKIP IF LESS THEN 7
0731 5336 JMP ERR1 /CBERR ON CHAR NOT IN RANGE
0732 1774 TAD C0CHAR /GET CHAR
0733 0353 AND (7 /MASK FOR RIGHT BYTE
0734 3774 DCA C0CHAR /STORE IN CHAR
/GET CHAR IN AC
/EXIT
ERR1, TAD (277 /CBPNT
/7
/7
/EXIT+ ASK AGAIN
/DO A UR LF
/DO A UR LF
DCA C0BNS7 /CLEAR THE PSH ENTRY FLAG
JMP I XC0PSW /EXIT ROUTINE
C0BNS7, 0

0746 0000 THPCNT, 0
0747 2322 MESA, TEXT "SR="
0750 7540
0751 0000

```

```

0752 0277
0753 0007
0754 7510
0755 7520
0756 0215
0757 7775
0760 1063
0761 1076
0762 0040
0763 1000
0764 0515
0765 0272
0766 0303
0767 1200
0770 1345
0771 1347
0772 1346
0773 1023
0774 1075
0775 1077
0776 0336
0777 0400

```

```

1000 PAGE
/C00CTA
/OCTAL TO ASCII CONVERSION
/THIS ROUTINE WILL TAKE THE OCTAL NUMBER IN THE AC AND CONVERT IT TO ASCII
/THE RESULT WILL BE PRINTED ON THE CONSOLE TERMINAL
/ C00CTA= JMS XC0OCT
/
/EX. JMS XC0OCTA /AC CONTAINS NUMBER TO BE CHANGE
/ RETURN IS TO CALL PLUS ONE AC=0
/
/CALLS USED ARE=XC0TYPE=

1000 0000 XC0OCT, 0
1001 7106 CLL RTL
1002 7006 RTL /POSITION THE FIRST CHAR FOR PRINTING
1003 3221 DCA C0TMP1 /SAVE CORRECT POSITIONED WORD HERE
1004 1377 TAD (-4
1005 3222 DCA C0CKP /STORE COUNTER IN HERE
1006 1221 TAD C0TMP1 /GET FIRST NUMBER
1007 0376 AND (0007 /MASK
1010 1375 TAD (260 /ADD THE PRINT CONSTANT
1011 4277 JMS XC0TYPE /TYPE THE NUMBER
1012 1221 TAD C0TMP1 /
1013 7006 RTL
1014 7004 RAL /PUT NEXT NUMBER IN POSITION
1015 3221 DCA C0TMP1 /STORE IT
1016 2222 ISZ C0CKP /DONE YET WITH FOUR NUMBERS
1017 5206 JMP C0D04 /NOT YET DO MORE
1020 5600 JMP I XC0OCT /DONE WITH FOUR
1021 0000 C0TMP1, 0
1022 0000 C0CKP, 0

/*****
/C0CRLF
/C0TYPE CR AND LF WITH FILLERS FOLLOWING EACH LF AND CR
/
/ C0CRLF= JMS XC0CRLF
/EX. JMS XC0CRLF /CBPNT A CR AND LF WITH FILL
/ /RETURN TO CALL PLUS ONE AC =0
/CALLS USED ARE=XC0TYPE=

1023 0000 XC0CRLF, 0
1024 7300 CLA CLL
1025 1374 TAD (215 /GET CODE FOR CR
1026 4277 JMS XC0TYPE
1027 1237 TAD FILLER
1030 7040 CMA
1031 3240 DCA FILCNT /STORE FILLER IN HERE

```

```

1032 1373 TAD C212 /GET CODE FOR LF
1033 4277 C000R, JMS XC8TYPE
1034 2240 ISZ FILCNT /CHECK ON FILLER CHAR
1035 5233 JMP C000R /TYPE A NON PRINTING CHAR
1036 5623 JMP I XC8CRL /EXIT
1037 0004 FILLER, 0004 /FILLER SET FOR 4 CHAR
1040 0000 FILCNT, 0 /COUNTER FOR FILL

```

```

//*****
/C8CKPA
/THIS ROUTINE WILL CHECK IF A CHARACTER WAS ENTERED FROM THE
/TERMINAL. IF THE FLAG IS SET AND THE CONSOLE PACKAGE IS
/ACTIVE A CHECK IS MADE TO DETERMINE IF IT IS A CONTROL CHAR.
/IF IT WAS A CONTROL CHAR THEN ITS CONTROL FUNCTION IS PERFORMED.
/IF NOT A CONTROL CHARACTER OR A CONTROL E-D-L-D- IT WILL DO
/THE CONTROL FUNCTION AND RETURN TO CALL PLUS 2.
/A NON CONTROL CHARACTER WILL BE PRINTED AND A "2" IT WILL RETURN TO
/CALL PLUS 2.
/IF NO FLAG IS SET OR THE CONSOL IS NOT ACTIVE THE RETURN IS TO
/CALL PLUS 1.

```

```

/ C8CKPA= JMS XC8CKP

```

```

/EX. JMS XC8CKPA /CALL TO CHECK IF CONTROL CHAR SET
/ ANYTHING(SKIP) /RETURN IF NOT FLAG OR NOT CONSOLE ACTIVE
/ ANYTHING(JMP EXIT SKIP CHAIN) /RETURN IF NOT CONTROL OR CONTINUE CONTROL

```

```

/CALLS USED ARE-XC8TTYI-XC8CNTN-C0GET-

```

```

1041 0000 XC8CKP, 0
1042 3772 DCA ACBAVE /SAVE THE AC
1043 0004 GTF /SAVE THE FLAG
1044 3771 DCA FLBAVE /SAVE THE FLAG
1045 7501 MQA /PUT MQ IN AC
1046 3770 DCA MQBAVE /BACK THE MQ
1047 0031 KSF /CHECK THE KEYBOARD FLAG
1050 5201 JMP C0BY3 /EXIT TO CALL PLUS 1
1051 4707 JMS CHKCLA /CHECK LOC 20 BIT 3 CONSOLE BIT
1052 7410 SKP /ACTIVE CONSOLE PACKAGE
1053 5201 JMP C0BY3 /EXIT TO CALL PLUS 1
1054 4706 JMS XC8TTYI /GET THE CHAR
1055 4705 JMS C0GET /GET THE FLAG
1056 4704 JMS XC8CNTN /CHECK IF CONTROL CHAR.
1057 7000 NOP /RETURN IF A CONTINUE CHAR.
1060 2241 ISZ XC8CKP /BUMP RETURN FOR CALL PLUS 2
1061 4705 C0BY3, JMS C0GET /GET REGISTERS
1062 5641 JMP I XC8CKP /BAY GOOD BY

```

```

//*****

```

```

/C8ECHO
/THIS ROUTINE WILL LOOK FOR A CHAR FROM THE KEYBOARD. STORE IT IN LOCATION CHAR
/CHECK IF IT WAS A CONTROL CHARACTER- SET INMODE- PRINT CHARACTER

```

```

/ C8ECHO = JMS XC8ECHO
/EX. JMS XC8ECHO /LOOK FOR CONSOLE CHAR C0PRT IT
/RETURN CALL PLUS ONE AC 0 CHAR C0TYPED IN

```

```

/CALLS USED ARE-XC8TTYI-XC8CNTN-C0GET-XC8ECHO-XC8TYPE

```

```

/
1063 0000 XC8ECHO, 0
1064 4706 JMS XC8TTYI /WAIT FOR CHAR FROM KEYBOARD
1065 4705 JMS C0GET /RESTORE THE REGISTERS
1066 2276 ISZ INMODE /SET INMODE IDENTIFYING THIS AS A EXPECTED CHAR
1067 4704 JMS XC8CNTN /GO CHECK IF IT IS A CONTROL CHAR
1070 5063 JMP I XC8ECHO /WAS A CONTROL CHAR- CONTINUE RUNNING
1071 4277 JMS XC8TYPE /NOT A CONTROL CHAR C0PRT IT
1072 3276 DCA INMODE /CLEAR FLAG THAT CHAR EXPECTED
1073 1275 TAD C0CHAR /GET CHAR IN AC
1074 5063 JMP I XC8ECHO /EXIT
1075 0000 C0CHAR, 0
1076 0000 INMODE, 0

```

```

//*****

```

```

/C8TYPE
/THIS ROUTINE WILL C0PRT ON THE CONSOLE OR THE LPT WITH DEVICE CODE 66.
/
/ C8TYPE= JMS XC8TYP

```

```

/EX. JMS XC8TYPE /C0PRT THE CHAR IN THE AC.
/RETURN CALL PLUS ONE AC =0000
/DO NOT CLEAR THE LINK IN THIS ROUTINE NEEDED BYC0CT

```

```

/CALLS USED ARE-C0HANG-XC8CNTN-XC8PNT-XC8CMLP-XC8INQU-

```

```

1077 0000 XC8TYP, 0
1100 3320 DCA PNTBUF /STORE CHAN
1101 1321 TAD TTYLPT /CHECK 0TTY 7777=LPT
1102 7640 SZA CLA
1103 5312 JMP XDULPT /OO OUT PUT ON LPT
1104 1320 TAD PNTBUF
1105 0046 TLA
1106 0041 TLF
1107 3306 JMP =-1
1110 0042 TCF
1111 5316 JMP C0BY5
1112 1320 XDULPT, TAD PNTBUF /GET CHAR
1113 0046 PSTB PCLF /C0PMT IT
1114 4322 JMS C0HANG /CHECK KEYBOARD IF HUNG
1115 0042 PCLF /CLEAR THE FLAG
1116 7600 C0BY5, 7600 /CLEAR THE AC

```



```

1117 5677      JMP I  XC8TYP      /EXIT
1120 0000      PNTBUF, 0
1121 0000      TTYLPT, 0

1122 0000      C0HANG, 0
1123 7200      CLA
1124 1316      TAD      C0BYS      /GET CONSTANT 7600
1125 3320      DCA      PNTBUF     /PNTBUF IS NOW A COUNTER
1126 6661      PSKF
1127 7410      SKP
1130 5722      JMP I  C0HANG     /SKIP UN PINTER DONE
1131 2345      ISZ      C0CONT     /NOT DONE YET
1132 5326      JMP      =-4         /SAB FLAG DONE
1133 2320      ISZ      PNTBUF     /FIRST COUNTER FAST ONE
1134 5331      JMP      =-5         /CHECK IF FLAG SET YET
1135 1764*     TAD      XC0CNTR    /MADE 4096 COUNTS ON FAST COUNTER
1136 3322      DCA      C0HANG     /KEEP IT UP FOR 5 SEC
1137 3321      DCA      TTYLPT    /GET THE RETURN ADDRESS IN CONTROL
1140 4763*     JMS      XC0PNT    /SAVE IT IN HANG
1141 1146      MESHANG
1142 4223      JMS      XC0CRLF   /ALLOW PRINTING ON TTY
1143 4762*     JMS      XC0INGU   /LPT ERROR
1144 5722      JMP I  C0HANG     /PRINT WAITING
1145 0000      C0CONT, 0      /CONTINUE TO SAVE ADDRESS
1146 1420      MESHANG,TEXT  "LPT ERROR" /COUNTER FOR TIMER
1147 2440
1150 0522
1151 2217
1152 2200

1162 0635
1163 0303
1164 0400
1165 0624
1166 0272
1167 1200
1170 1346
1171 1347
1172 1345
1173 0212
1174 0215
1175 0260
1176 0007
1177 7774
1200

```

PAGE
/*****
/*****
/THIS ROUTINE WILL CHECK LOCATION 22 THE HARD WARE CONFIG WORD.
/TO SEE IF THE CONSOLE BIT 3 (400) IS SET IF SET THEN RETURN
/TO CALL PLUS TWO FOR A ACTIVE CONSOLE PACKAGE AC#0
/IF NOT SET THEN TO CALL PLUS ONE FOR A DEACTIVE CONSOLE PACKAGE.

```

1200 0000      CHKCLA, 0
1201 7200      CLA
1202 1022      TAD      22          /GET THE CONTENTA OF LOCATION 22
1203 0377      AND      (400)       /MASK FOR BIT 3 (400)
1204 7650      SNA CLA
1205 2200      ISZ      CHKCLA     /
1206 5600      JMP I  CHKCLA     /ACTIVE CONSOLE PACKAGE RETURN
1207 0000      /CALL PLUS ONE (1) FOR ACTIVE
1208 0000      /DEACTIVE CONSOLE PACKAGE RETURN
1209 0000      /CALL PLUS TWO (2)

/COBERR
/THIS ROUTINE WILL DETERMINE WHAT TO DO WHEN A COBERR IS ENCOUNTERED
/WILL CHECK IF CLASSIC SYSTEM, WILL CHECK COBERR REGISTERS.
/      COBERR= JMS XC0ERR
/EX.  JMS  XC0ERR      /GO TO COBERR CALL IF NOT CONSOLE
/      /RETURN IS CALL PLUS ONE AC #0000

/CALLS USED ARE=CHKCLA=XC0CRLF=XC0BYS=XC0INGU=XC0PNT=XC0OCTA=

1207 0000      XC0ERR, 0
1210 6002      IOP
1211 3345      DCA      ACSAVE     /SAVE AC
1212 6004      GTF
1213 3347      DCA      PLSAVE     /SAVE THE FLAGS
1214 7501      MGA
1215 3346      DCA      MGSAVE     /SAVE THE MQ
1216 7340      CLA CLL CMA      /SUBTRACT A 1 FOR TRUE LOCATION
1217 1207      TAD      XC0ERR     /GET RETURN LOCATION
1220 3344      DCA      PCSAVE     /SAVE ADD OF COBERR CALL
1221 6201      CDF
1222 7340      CLA CLL CMA
1223 1776      TAD I  (CLASIK)    /GET REAL PC.
1224 3316      DCA      REALPC     /SAVE IT.
1225 6211      CDF
1226 4200      JMS      CHKCLA     /CHECK LOC.22 BIT 3 CONSOLE BIT
1227 7410      SKP
1230 5270      JMP      NTCLAS    /ACTIVE CONSOLE PACKAGE
1231 4775*     JMS      C0GET     /NOT CLASSIC SYSTEM
1232 4774*     JMS      XC0BYS    /GET REGISTERS.
1233 0373      SETUP1, AND  (0000) /CHECK SWITCH REG FOR BIT THAT INDICATES
1234 7640      /NO ERROR MESSAGE
1235 5262      SZA CLA          /MASK FOR BIT FOR NO ERROR PRINTING
1236 4772*     JMP      C0U010    /IF THIS ERROR MESSAGE IS TO ALWAYS
1237 4771*     JMS      XC0CRLF   /BE PRINTED LEAVE AND VALUE AT 0000
1240 1320      ERRMES          /SKIP IF BIT IS 0 PRINT ERROR MESSAGE
1241 4771*     JMS      XC0PNT    /DO NOT PRINT
1242 1330      MESPC          /PRINT THE ERROR MESSAGE
1243 1316      TAD      REALPC     /PRINT THE PC STATEMENT
1244 4770*     JMS      XC0OCTA   /GET PC
1245 4771*     JMS      XC0PNT    /CONVERT 4 DIGIT PC TO ASCII
1246 1333      MESAC          /PRINT THE AC MESS

```

```

1247 1345      TAD      ACSAVE
1250 4770'    JMS      XCBOCTA
1251 4771'    JMS      XCOPNT
1252 1336      MESMQ
1253 1346      TAD      MQSAVE
1254 4770'    JMS      XCBOCTA
1255 4771'    JMS      XCOPNT
1256 1341      MESFL
1257 1347      TAD      FLSAVE
1260 4770'    JMS      XCBOCTA
1261 4772'    JMS      XCOCRLF
1262 4775'    C00010, JMS     C0GET
1263 4774'    JMS      XCOSW
1264 7610      SKP     CLA
1265 5300      JMP     C0BY2
1266 4767'    JMS      XCINQ
1267 5300      JMP     C0BY2
1270 4775'    NTCLAS, JMS     C0GET
1271 4774'    JMS      XCOSW

1272 7610      SKP     CLA
1273 5607      JMP     I      XCERR
1274 1366      TAD      (7402
1275 3744      DCA     I      PCSAVE
1276 4775'    JMS      C0GET
1277 5744      JMP     I      PCSAVE
1300 4775'    C0BY2,  JMS     C0GET
1301 5607      JMP     I      XCERR

/
/
1302 7402      ROUINS, HLT
1303 7000      NOP
1304 3317      DCA     MYAC
1305 6201      CDF     0
1306 1020      TAD     SWR
1307 3765      DCA     I      (SWR)
1310 1776      TAD     I      (CLASIK)
1311 3315      DCA     CLRTRN
1312 1317      TAD     MYAC
1313 6202      CIF     0
1314 5715      JMP     I      CLRTRN

/
1315 0000      CLRTRN, 0
1316 0000      REALPC, 0
1317 0000      MYAC, 0

/
1320 0410      ERNME, TEXT "DHRKCH FAILED "
1321 2213
1322 0310
1323 4040
1324 0601
1325 1114
1326 0504
1327 4000
1330 4040      MESPC, TEXT " PC1"

```

```

1331 2003
1332 7200
1333 4040      MESAC, TEXT " AC1"
1334 0103
1335 7200
1336 4040      MESMQ, TEXT " MQ1"
1337 1521
1340 7200
1341 4040      MESFL, TEXT " FL1"
1342 0614
1343 7200
1344 7777      PCSAVE, 7777
1345 7777      ACSAVE, 7777
1346 7777      MQSAVE, 7777
1347 7777      FLSAVE, 7777

/
1365 0020
1366 7402
1367 0035
1370 1000
1371 0303
1372 1023
1373 0000
1374 0262
1375 0024
1376 1514
1377 0400
0000      FIELD 0

```

0000	00000000	00000000	11101111	11111111	11000000	00000000	00000000	00000000
0100	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
0200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0300	11111111	11111111	11111111	11111111	11111111	10000001	11111111	11111111
0400	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0500	11111111	11111111	11111111	11111111	11111111	11111110	11111111	11111111
0600	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
0700	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1100	11111111	11111111	11111111	11111111	11111111	11100000	00111111	11111111
1200	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
1300	11111111	11111111	11111111	11111111	11111111	00000000	00000111	11111111

1400
1500

1600
1700

2000
2100

2200
2300

2400
2500

2600
2700

3000
3100

3200
3300

3400
3500

3600
3700

4000
4100

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

```

/
/NOTES:LOCATION 0 CONTAINS THE REVISION
/LEVEL (IN ASCII) ON PROGRAM LOAD.
/
/ALL KNOWN HALTS
/
1400 0556 ERHLT0 /SKIP TRAP DLSC
1401 0563 ERHLT2 /SKIP TRAP DCLR
1402 2561 ERHLT3 /SKIP TRAP DLAG
1403 2544 ERHLT5 /SKIP TRAP DRST
1404 0547 ERHLT6 /SKIP TRAP DLDC
1405 3130 INTER1 /NO DISK INTERRUPT
1406 2362 INTER2 /UNDEFINED INTERRUPT
1407 0206 FLDHLT /PROGRAM WILL ONLY RUN IN FIELD 0
1410 2702 NODSK0 /NO DISKS AVAILABLE TO RUN
1411 0603 STPHLT /PROGRAM STOP FROM SWR0=1
1412 2755 CMHHLT /IOT CHANGE HALT
1413 1707 BAHHLT /COMPUTER MUST BE DOWN, CHECKSUM FAILED
/
1414 3136 BIGSTP /BUT WORD-BY-WORD COMPARE WORKED.
/STOP FOR ALL ERROR HALTS.
/
6740 DLSC=6740 /LOAD SECTOR COUNTER
6741 DSKP=6741 /SKIP UN TRANSFER DONE OR ERROR
6742 DCLR=6742 /CLEAR DISK CONTROL LOGIC
6743 DLAG=6743 /LOAD ADDRESS AND GO
6744 DLCA=6744 /LOAD CURRENT ADDRESS
6745 DRST=6745 /READ STATUS REGISTER
6746 DLDC=6746 /LOAD COMMAND REGISTER
/
4406 LAB=JMS I XLAS
4407 CLASIC=JMS I XCLAS
4427 RANDAT=JMS I XRNWRD
4430 DISCON=JMS I XDUMP
4431 SPACE=JMS I XSPAC
4432 ONEIN=JMS I XOCT1
4433 FORIN=JMS I XOCT4
4434 SETGEN=JMS I XSTGEN
/
4435 SETFLD=JMS I XSTFLD
4437 YESNO=JMS I XCHKYN
/
4436 SELCHK=JMS I XCKPOT
4440 RANGEN=JMS I XRNODM
4442 REBRAN=JMS I XRBRAN
4441 DISKGO=JMS I XDSKGO
4443 RECAL=JMS I XRESTR
4444 RECEIV=JMS I XWAIT
4446 ERROR=JMS I XERRO
4447 ROSTAT=JMS I XROST
4453 LDADD=JMS I XL0AD
4450 DSKBKP=JMS I XSKBP

```

```

4451 LDCM=JMS I XLDCM
4452 LOCUR=JMS I XL0CA
4454 CLRALL=JMS I XCLDR
4455 PRNTER=JMS I XPRN
4456 OCTEL=JMS I XPROCT
4445 TYPE=JMS I XPRINT
4457 CRLF=JMS I XCRLF
4426 GENDAT=JMS I XGN0AT
4424 CHK22=JMS I XCMK22
4425 KTICK=JMS I XKTCK
/
0000 *0
/
0000 0310 310 /REVISION "H" INTERRUPT SERVICE RETURN
0001 5001 5001 /OCA SAVAC SAVE AC AT INT.
0002 0002 0002 /RAL SHIFT LINK AT TIME OF INT.
0003 0003 0003 /OCA SVLNK SAVE LINK AT TIME OF INT.
0004 0004 0004 /JMP I 5 RETURN TO INT. SERVICE
0005 0005 0005 /RETURN POINTER
/
0006 1546 XLAS, MYLAS
0007 1514 XCLAS, CLASIK
/
0010 *10
/
0010 0000 AUTO10, 0
/
0011 0000 AUTO11, 0
/
0012 0000 AUTO12, 0
/
0013 0004 K0004, 0004
0014 0070 K0070, 0070
0015 0100 K0100, 0100
0016 0200 K0200, 0200
0017 0400 K0400, 0400
/
0020 *20
/
0020 0000 SWK, 0
0021 4000 OP1, 4000
0022 0000 OP2, 0
/
0023 2156 KAERRO, AERROR
0024 0523 XCMK22, CMK22
0025 1154 XKTCK, KXTICK
0026 1737 XGN0AT, GN0AT
0027 2600 XRNWRD, RNWRD
0030 2637 XDUMP, DUMP
0031 1506 XSPAC, SPAC
0032 2400 XOCT1, OCT1
0033 2430 XOCT4, OCT4
0034 1753 XSTGEN, STGEN
0035 2703 XSTFLD, STFLD

```

```

0036 2060 XCKPOT, CKPOT
0037 2035 XCMKYN, CMKYN
0040 1715 XRANDOM, RANDOM
0041 2200 XD8KGO, D8KGO
0042 1761 XRSRAN, RSRAN
0043 3052 XRESTR, RESTOR
0044 2000 XWAIT, WAIT
0045 2620 XPRINT, PRINT
0046 1200 XERRO, ERRO
0047 2541 XROST, ROST
0050 0751 X8UKP, 8UKP
0051 0542 XLUOM, LCOM
0052 2550 XLCCA, LCCA
0053 2554 XLOAD, LOAD
0054 0560 XCLDR, CLDR
0055 1450 XPRN, PRN
0056 1426 XFROCT, FROCT
0057 1414 XCRLF, UPONE
0060 0000 AMOUNT, 0
0061 0001 K0001, 0001
0062 0003 K0003, 0003
0063 0006 K0006, 0006
0064 0007 K0007, 0007
0065 0010 K0010, 0010
0066 0017 K0017, 0017
0067 0260 K0260, 0260
0070 0277 K0277, 0277
0071 0770 A0770, 0770
0072 7007 A7007, 7007
0073 4000 K4000, 4000
0074 4100 K4100, 4100
0075 1000 K1000, 1000
0076 1777 K1777, 1777
0077 7700 K7700, 7700
0100 7760 K7760, 7760
0101 7777 K7777, 7777
0102 0077 K0077, 0077
0103 6201 KCDF, CDF
0104 7400 K7400, 7400
/
0105 7764 M12, -12
/
0106 7774 M4, -4
0107 7770 M10, -10
0110 7775 K7775, 7775
/
0111 0000 TRASH1, 0
0112 0000 TRASH2, 0
0113 0000 TRASH3, 0
0114 0000 UPDATE, 0
0115 0000 POLDSK, 0

```

```

0116 0000 OPMTAL, 0
0117 0000 BUFTAL, 0
0120 0000 PCNEG, 0
0121 0000 STNEG, 0
0122 0000 EXREG, 0
0123 0000 CMREG, 0
0124 0000 INTDA, 0
0125 0000 DAREG, 0
0126 0000 CAREG, 0
0127 0000 WCNEG, 0
0130 0000 FWNEG, 0
0131 0000 ASNEG, 0
0132 0000 WAREG, 0
0133 0000 ADREG, 0
0134 0000 DGREG, 0
0135 0000 DBNEG, 0
0136 0000 INTCH, 0
0137 0000 STATRY, 0
0140 0000 DATTRY, 0
0141 0000 CMKSAV, 0
0142 0000 FNDUSM, 0
0143 0000 MAXFLD, 0
0144 7607 MAXTIM, 7607
0145 3240 MAXTRK, 3240
0146 3600 BGNBUF, 8TRBUF
0147 0000 CONSEC, 0
0150 7777 CLKCNT, -1
/
0151 0756 DATPOT, DAT1
0152 3522 TIRPOT, 00TM1
0153 3537 STAPOT, 00HRD-3
0154 3512 RUNPUT, 00K00
/
0135 0000 CRCNT, 0
0156 0000 CRCFLG, 0
0137 0000 DATFLG, 0
0160 0000 SPFLD, 0
0161 0000 SPTRK1, 0
0162 0000 SPTRK2, 0
0163 0000 SPSEC, 0
0164 0000 SPNLK, 0
0165 0000 ERFLG, 0
0166 0000 SAVAC, 0
0167 0000 SVLNC, 0
0170 0000 FINTIM, 0
0171 0000 TRYCNT, 0
0172 3213 XTEXT, TEXPC
0173 3142 PRNDAT, TYPDAT
0174 0000 SAVCM, 0
0175 0000 CLNBAK, 0
/
0176 3131 BGLT, 01GHT
0200 *200
/
/

```

```

/START OF PROGRAM BY OPERATOR:
/AT 0200, TTY INTERMIGATION:
/AT 0201, CHANGE IOT DEVICE CODES:
/AT 0202, RESTART AT SEEK ROUTINE:
/
0200 4777* BGN, JMS APTS /TO REGULAR TEST
0201 5776* JMP CHANG /CHANGE IOT ROUTINE
0202 5775* JMP RUN
0203 3156 DCA CRCFLG /CLEAR CRC FLAG
0204 6224 RIF
0205 7440 SZA /FIELD 0777?
0206 4576 FLUMLT, JMS I BGMLT /WILL ONLY RUN IN FIELD 0777?
0207 1103 TAD KCDF
0210 3211 DCA ,+1
0211 7402 MLT /MAKE DF=IF
/
/SETUP INTERRUPT SERVICE:
/
0212 1362 TAD ACUCA
0213 3001 DCA 1 /SETUP AC DCA
0214 1250 TAD KRUT
0215 3002 DCA 2 /SETUP ROTATE LINK
0216 1361 TAD LNKDCA
0217 3003 DCA 3 /SETUP SAVE LINK
0220 1360 TAD K5405
0221 3004 DCA 4 /SETUP JMP RETURN
0222 1363 TAD BRKRET
0223 3005 DCA 5 /RETURN POINTER
/
/CLEAR DATA INFORMATION TABLE
/AT END OF PROGRAM:
/
0224 1077 STRTEX, TAD K7700
0225 3111 DCA TRASH1 /CLEAR COUNTER
0226 1774* TAD RANJMS
0227 3773* DCA SWDAT /SET INSTRUCTION SWITCH
0230 7340 CLA CLL CMA
0231 1152 TAD TAMPDT
0232 3010 DCA AUTO10 /LOCATION POINTER
0233 3410 DCA I AUTO10 /CLEAR
0234 2111 ISZ TRASH1
0235 5233 JMP ,+2 /MORE TO CLEAR
0236 3157 DCA DATFLG
0237 5775* SKPNOP, JMP RUN
/
/PRINT PROGRAM NAME AND
/ASK OPERATOR ABOUT AMOUNT
/OF MEMORY:
/
0240 4457 CRLF
0241 4455 PRNTER /PRINT "RKBE/RKBL DATA RELIABILITY"
0242 3307 MES1
0243 4455 PRNTER /PRINT "AMOUNT OF MEMORY"
0244 3346 MESS

```

```

0245 4432 ONEIN
0246 0070 0070 /RECEIVE ONE OCTAL
0247 5243 JMP ,+4 /LIMITS
0250 7004 KRUT, RAL /INPUT ERROR
0251 7006 RTL
0252 7040 CMA
0253 3143 DCA MAXFLD /COMPLEMENT
0254 4772* JMS CLAFLD /MAXIMUM FIELD POINTER
0255 3111 ALLAGN, DCA TRASH1 /CHECK FOR CLASSIC.
0256 1107 TAD M10
0257 3112 DCA TRASH2
0260 3060 DCA AMOUNT /A FEW POINTERS
/
/ASK OPERATOR ABOUT DISK(S) TO TEST:
/
0261 1111 NEXT, TAD TRASH1
0262 1154 TAD RUNPDT
0263 3113 DCA TRASH3 /SAVE RUN POINTER
0264 4455 PRNTER /PRINT "EXERCISE"
0265 3325 MES2
0266 7340 CLA CLL CMA
0267 4455 PRNTER /PRINT "DISK"
0270 3332 MES3
0271 1067 TAD K0260
0272 1111 TAD TRASH1 /ADD IN DISK NUMBER
0273 4445 TYPE /TYPE DISK NUMBER
0274 1070 TAD K0277
0275 4445 TYPE /TYPE ?
0276 4444 RECEIV /RECEIVE KEY INPUT
0277 4437 YESNO /WAS IT YES OR NO
0300 5255 JMP ALLAGN /NEITHER
0301 5304 JMP ,+3 /WAS A NO
0302 2060 ISZ AMOUNT /AMOUNT OF DISK FOUND
0303 7340 CLA CLL CMA /AC TO 7777 FOR EXISTING DISK
0304 3513 DCA I TRASH3 /SETUP RUN POINTER
0305 2111 ISZ TRASH1
0306 2112 ISZ TRASH2
0307 5261 JMP NEXT /ASK ABOUT NEXT DISK
/
/
/ASK IF ACCEPT MODE:
/
0310 1060 TAD AMOUNT /GET AMOUNT FOUND
0311 7050 SNA CLA /WERE ANY FOUND
0312 5224 JMP STRTEX /OPERATOR ERROR NO DISK INPUT
0313 4455 PRNTER /PRINT "ACCEPT MODE?"
0314 3363 MES6
0315 4444 RECEIV /RECEIVE INPUT
0316 4437 YESNO /YES OR NO???
0317 5313 JMP ,+4 /NEITHER ALL AGAIN
0320 7610 SKP CLA /MANUAL TEST
0321 5771* JMP ASKSUR /ASK "ARE YOU SURE"
/
/
/IF ACCEPT MODE, INTERAGATE

```

```

/ABOUT CONSTANT FIELD1
/
0322 4455 MANUAL, PRNTER /PRINT "FIELD?"
0323 3404 MESS
0324 4444 RECEIV /RECEIVE Y OR N
0325 4437 YESNO /CHECK FOR Y OR N
0326 5322 JMP MANUAL /NEITHER Y OR N
0327 5345 JMP ASKNX1 /HAS A N, ASK ABOUT NEXT
0330 4431 SPACE /SPACE OUT ONE
0331 4432 ONEIN /GET 1 OCTAL
0332 0070 0070 /LIMITS
0333 5322 JMP MANUAL /INPUT ERROR ASK AGAIN
0334 7104 CLL RAL
0335 7006 RTL
0336 3100 DCA SPFLD /SAVE INPUT
0337 1100 TAD SPFLD
0340 1143 TAD MAXFLD /COMPARE TO MAXIMUM
0341 7700 SMA CLA /O.K.?
0342 5322 JMP MANUAL /INPUT ERROR
0343 7340 CLA CLL CMA
0344 3770 DCA FLOFLG /SETUP FIELD FLAG
/
/INTERIGATE ABOUT CONSTANT TRACK1
/
0345 4455 ASKNX1, PRNTER /PRINT "TRACK?"
0346 3410 MESS
0347 4444 RECEIV /RECEIVE Y OR N
0350 4437 YESNO /CHECK FOR Y OR N
0351 5345 JMP ASKNX1 /ERROR, ASK AGAIN
0352 5707 JMP ASKNX2 /N, ASK ABOUT NEXT
0353 4431 SPACE
0354 4432 ONEIN /RECEIVE 1 IN OCTAL
0355 0010 0010 /LIMITS
0356 5345 JMP ASKNX1 /ERROR, ASK AGAIN
0357 5706 JMP SAVE1 /TU SAVE SOME ROOM,
/
0360 5405 K5405, 5405
0361 3107 LNKDCA, DCA SVLNK
0362 3106 ACUCA, DCA SAVAC
0363 2304 BRKRET, RETURN
/
0366 0400
0367 0406
0370 3572
0371 0513
0372 1404
0373 2601
0374 0522
0375 0600
0376 2730
0377 2070
0400 PAGE
/
/

```

```

/INTERIGATE ABOUT CONSTANT
/BLOCK LENGTH1
/
0400 3101 SAVE1, DCA SPTRK1 /SAVE EXTENDED TRACK BIT
0401 4433 FORIN /GET FOUR IN OCTAL.
0402 5777 JMP ASKNX1 /ERROR, ASK AGAIN
0403 3102 DCA SPTRK2 /SAVE CYL., SURFACE, AND SECTOR
0404 7340 CLA CLL CMA
0405 3770 DCA TRKFLG /SETUP TRACK FLAG
/
0406 4455 ASKNX2, PRNTER /PRINT "BLOCK LENGTH?"
0407 3424 MESS1
0410 4444 RECEIV /RECEIVE INPUT
0411 4437 YESNO /CHECK FOR Y OR N
0412 5206 JMP ASKNX2 /ERROR, ASK AGAIN
0413 5225 JMP ASKNX3 /N, ASK ABOUT NEXT
0414 4431 SPACE /Y, SPACE OUT 1
0415 4432 ONEIN /RECEIVE 1 IN OCTAL
0416 0010 0010 /LIMITS
0417 5206 JMP ASKNX2 /ERROR, ASK AGAIN
0420 7640 SZA CLA /SET HALF BLOCK?
0421 7340 CLA CLL CMA /YES
0422 3104 DCA SPBLK /SETUP BLOCK NUMBER
0423 7340 CLA CLL CMA
0424 3775 DCA MLFFLG /SETUP BLOCK FLAG
/
/INTERIGATE ABOUT CONSTANT
/SECTORS1
/
0425 4455 ASKNX3, PRNTER /PRINT "EXTRA SECTORS?"
0426 3414 MESS10
0427 4444 RECEIV /RECEIVE INPUT
0430 4437 YESNO /CHECK FOR Y OR N
0431 5225 JMP ASKNX3 /INPUT ERROR
0432 5204 JMP ASKNX5 /N, ASK ABOUT NEXT
0433 4431 SPACE /SPACE OUT 1
0434 4432 ONEIN /RECEIVE 1 IN OCTAL
0435 0010 0010 /LIMITS
0436 5225 JMP ASKNX3 /ERROR, ASK AGAIN
0437 7104 CLL RAL
0440 7006 RTL
0441 3103 DCA SPSEC /SAVE IT
0442 4432 ONEIN /RECEIVE 1 IN OCTAL
0443 0070 0070 /LIMITS
0444 5225 JMP ASKNX3 /INPUT ERROR, ASK AGAIN
0445 1103 TAD SPSEC /ADD IN LAST
0446 3103 DCA SPSEC /SAVE ALL
0447 1104 TAD SPBLK
0450 7640 SZA CLA /BLOCK LENGTH 0?????
0451 5254 JMP ,+5 /NO LIMIT IS 17.
0452 1100 TAD SPFLD
0453 7640 SZA CLA /FIELD 0?????
0454 1065 TAD K0010 /LIMIT IS 17.
0455 1064 TAD K0007

```

```

0456 7140      CLL CMA
0457 1163      TAD      SPSEC      /COMPARE SECTOR INPUT;
0460 7630      S2L CLA      /IN LIMITS???
0461 5225      JMP      ASKNX3      /NO, INPUT ERROR
0462 7340      CLA CLL CMA
0463 3774      DCA      SECFLG      /SETUP SECTOR FLAG
//
//
//INTERIGATE ABOUT "OPERATOR
//SELECT DATA"
0464 4455      ASKNX5, PRNTER      /PRINT "DATA"
0465 3433      MES13
0466 1322      TAD      RANJMB
0467 3773      DCA      SWDAT      /SET INSTRUCTION SWITCH
0470 4444      RECEIV      /RECEIVE INPUT
0471 4437      YESNO      /Y OR N
0472 5264      JMP      ASKNX5      /ERROR, ASK AGAIN
0473 5313      JMP      ASKSUR      /ASK "ARE YOU SURE"
0474 1346      TAD      KSKP
0475 3773      DCA      SWDAT      /SET INSTRUCTION SWITCH
0476 1105      TAD      M12
0477 3111      DCA      TRASH1      /SETUP WORD COUNTER
0500 7340      CLA CLL CMA
0501 1151      TAD      DATPOT      /GET POT POINTER
0502 3010      DCA      AUTO10
0503 4457      CRLF
0504 4433      FORIN
0505 5264      JMP      ASKNX5      /RECEIVE 4 IN OCTAL
0506 3410      DCA I      AUTO10      /INPUT ERROR, ASK AGAIN
0507 2111      ISZ      TRASH1      /SAVE DATA
0510 5303      JMP      ,+5      /UPDATE COUNTER
0511 7340      CLA CLL CMA      /GET NEXT
0512 3157      DCA      DATFLG      /SETUP DATA FLAG
//
//ASK IF HE'S SURE;
0513 4455      ASKSUR, PRNTER      /PRINT "ARE YOU SURE"
0514 3436      MES14
0515 4444      RECEIV      /GET INPUT
0516 4437      YESNO      /Y OR N
0517 5313      JMP      ASKSUR      /INPUT ERROR
0520 3772      JMP      STRTEX      /ALL AGAIN
0521 5771      JMP      RUN      /START DATA TESTING
0522 4426      RANJMB, GENDAT
//
//THIS ROUTINE TESTS FOR BEING ON APT,
//IF ON APT RETURN IS PLUS ONE, IF NOT RETURN IS PLUS TWO.
0523 0000      CHEK22, 0
0524 1022      TAD      22
0525 7700      SNA CLA      /ON APT?
0526 2323      ISZ      CHEK22      /NO, UPDATE RETURN.
0527 5723      JMP I      CHEK22      /AND RETURN.

```

```

//
//ROUTINE TO NOTIFY APT.
0530 0000      KTIME, 0
0531 4424      CHK22
0532 7410      SKP
0533 5730      JMP I      KTIME      /ON APT.
0534 6002      IOF      /NOT ON APT.GO ABOUT NORMAL RUN.
0535 6201      CDF      00      /TURN INTERRUPT SYSTEM OFF
//
//DATA FIELD SHOULD ALWAYS
//BE ZERO IN PROGRAM RUN.
//CHANGED TO CURRENT DATA FIELD.
0536 6272      CIF      70
0537 4741      JMS I      K6500
0540 5730      JMP I      KTIME      /RETURN,
//
0541 6500      K6500, 6500
//
//SUBROUTINE TO LOAD COMMAND REGISTER
0542 0000      LDCM, 0
0543 3123      DCA      CMREG
0544 1123      TAD      CMREG
0545 6746      IOT0, DLDC
0546 7610      KSKP, SKP CLA      /LOAD COMMAND REGISTER
0547 4576      ERMLT0, JMS I      BGMLT
0550 1122      TAD      EXREG
0551 7110      CLL RAM
0552 7630      S2L CLA
0553 1016      TAD      K0200
0554 6740      IOT0, DL0C
0555 7610      SKP CLA      /LOAD EXT. DRIVE
0556 4576      ERMLT0, JMS I      BGMLT      /SKIP TRAP IOT0
0557 5742      JMP I      LDCM      /EXIT
//
//SUBROUTINE TO ISSUE "DCLR" CLEAR IOT
0560 0000      CLDR, 0
0561 6742      IOT2, DCLR      /DCLR "CLEAR IOT"
0562 5760      JMP I      CLDR      /EXIT
0563 4576      ERMLT2, JMS I      BGMLT      /ENRDR SKIP TRAP
//
0571 0600
0572 0224
0573 2601
0574 3574
0575 3575
0576 3573
0577 0345
0600
PAGE
//
//SETUP ADDRESSING, COMMAND,
//AND DATA PARAMETERS;

```



```

/MAKE FIELD1
/
0600 4406 RUN, LAS
0601 0016 AND K0200
0602 7640 SZA CLA
0603 7402 STPHLT, HLT
0604 1777* TAD FLDPLG
0605 7650 SNA CLA
0606 5211 JMP ,+3
0607 1160 TAD SPFLD
0610 5230 JMP HNPLD
0611 7301 CLA CLL IAC
0612 1143 TAD MAXFLD
0613 7650 SNA CLA
0614 5230 JMP HNPLD
0615 4400 RANGEN
0616 0014 AND K0070
0617 7450 SNA
0620 5230 JMP HNPLD
0621 3136 DCA INTCH
0622 1136 TAD INTCH
0623 1143 TAD MAXFLD
0624 7710 SPA CLA
0625 5231 JMP RNPLD+1
0626 1143 TAD MAXFLD
0627 7000 CMA
0630 3136 RNPLD, DCA INTCH
/MAKE BLOCK LENGTH1
/
0631 1776* TAD MLFFLG
0632 7650 SNA CLA
0633 4400 RANGEN
0634 1164 TAD SPBLK
0635 0015 AND K0100
0636 1136 TAD INTCH
0637 3136 DCA INTCH
0640 1136 TAD INTCH
0641 0015 AND K0100
0642 7640 SZA CLA
0643 1016 TAD K0200
0644 1104 TAD K7400
0645 3112 DCA TRASH2
0646 1112 TAD TRASH2
0647 7041 CIA
0650 3114 DCA UPDATE
0651 1136 TAD INTCH
0652 0350 AND AB170
0653 7640 SZA CLA
0654 1065 TAD K0010
0655 1064 TAD K0007
0656 3111 DCA TRASH1
/MAKE AMOUNT OF SECTORS
/TO TRANSFER

```

```

/GET THE SWITCHES.
/MASK HLT SW.
/TIME TO HALT?
/HLT FROM SWR4#1.
/GET FIELD FLAG
/WAS IT SET?
/NO, USE RANDOM FIELD
/YES, GET OPERATOR FIELD
/GO
/GET MAXIMUM FIELD POINTER
/ANY FIELDS THERE
/NO EXTENDED FIELDS TO USE
/YES, GET A RANDOM FIELD
/MASK
/COULD BE 0
/WAS DON'T HAVE TO CHECK LIMITS
/SAVE FIELD FOUND
/ADD IN MAXIMUM FIELD POINTER
/IN LIMITS????
/YES, USE IT
/NO, USE MAXIMUM IN THE MACHINE

```

```

/
0657 1775* TAD SECFLG
0660 7650 SNA CLA
0661 4400 RANGEN
0662 1163 TAD SPSEC
0663 0111 AND TRASH1
0664 3147 DCA CONSEC
0665 1147 TAD CONSEC
0666 7040 CMA
0667 3111 DCA TRASH1
/MAKE WORD COUNT1
/
0670 1112 TAD TRASH2
0671 2111 ISZ TRASH1
0672 5270 JMP ,+2
0673 3127 DCA WCREG
/MAKE CURRENT ADDRESS1
/
0674 4400 RANGEN
0675 3126 DCA CAREG
0676 1136 TAD INTCH
0677 0014 AND K0070
0700 7640 SZA CLA
0701 5317 JMP FILLUP
0702 1146 TAD BGNBUF
0703 7100 CMA CLL
0704 1126 TAD CAREG
0705 7620 SNA CLA
0706 5315 JMP CONCUR
0707 1127 TAD WCREG
0710 7041 CIA
0711 1126 TAD CAREG
0712 1016 TAD K0200
0713 7630 SZA CLA
0714 5317 JMP FILLUP
0715 1146 CONCUR, TAD BGNBUF
0716 3126 DCA CAREG
/ROUTINE TO FILL AND CHECK SUM BUFFER
/
0717 4425 FILLUP, KTKIC
0720 4434 BETGEN
0721 1106 TAD M4
0722 3137 DCA STATRY
0723 4435 REPIII, SETFLD
0724 3325 DCA ,+1
0725 7402 HLT
0726 3141 DCA CHKSAV
0727 4427 NEWRD, RANDAT
0730 3111 UCA
0731 1111 TAD TRASH1
0732 3411 DCA I AUTO11
0733 7100 CLL

```

```

/GET SECTOR FLAG
/WAS IT SET???
/USE RANDOM
/GET OPERATOR INPUT
/MASK OUT
/SAVE
/CONSECUTIVE TO DO
/COMPUTE INITIAL WC
/UPDATE BY BUILDER
/INITIAL WORD COUNT ****
/GENERATE RANDOM CA
/SAVE IT
/MASK FIELD BITS
/EXTENDED FIELD???
/INITIAL CA O.K.****
/GRATER THAN PROGRAM+1
/NO, USE CONSTANT VALUE
/GET WORD COUNT
/ADD IN CA
/WITHIN BOUNDS????
/YES, INITIAL CA O.K.****
/NO, USE PROGRAM+1
/SAVE IT
/NOTIFY APT IF NEED BE.
/SETUP AND SAVE GENERATER
/SETUP TRY COUNTER
/FIELD+ BUPTAL+ AUTO 11+ 12
/FIELD TO BUFFER IN AC
/BUF TO BUFFER
/STANT WITH 0
/GENERATE DATA
/SAVE OUTPUT WORD
/GET BACK WORD
/STONE IN BUFFER

```

```

0730 1111      TAD   TRASH1      /GET BACK WORD
0735 1141      TAD   CNKSAV      /ADD IN LAST
0736 7430      SZL                   /LINK SET??
0737 7001      IAC                   /ADD IT IN
0740 3141      DCA   CNKSAV      /SAVE FOR NEXT
0741 2117      ISZ   BUPTAL      /UPDATE BUFFER TALLY
0742 5327      JMP   NEWRD      /MORE WORDS TO GO
0743 6201      CDF   @
0744 1165      TAD   ERPLG
0745 7650      SNA   CLA          /ENRRR FLAG SET????
0746 5774*     JMP   POLNEX      /POLE DRIVES
0747 5773*     JMP   RENRT      /YES, MUST BE A WRITE ERROR

0750 0170      /
                A0170, 0170
                /
                /SUBROUTINE TO ISSUE "D8KP" DISK SKIP IOT
                /
0751 0000      S0KP, @
0752 6741      IOT1, D8KP          /DISK SKIP IOT
0753 7410      SKP                   /DID NOT SKIP
0754 2351      ISZ   S0KP
0755 5751      JMP I  S0KP          /EXIT

                /PLACE FOR DATA IN MANUAL MODE
                /
0756 0000      DAT1, 0000
0757 0000      DAT2, 0000
0760 0000      DAT3, 0000
0761 0000      DAT4, 0000
0762 0000      DAT5, 0000
0763 0000      DAT6, 0000
0764 0000      DAT7, 0000
0765 0000      DAT8, 0000
0766 0000      DAT9, 0000
0767 0000      DAT10, 0000
0770 0000      DAT11, 0000
0771 0000      DAT12, 0000

                /
0773 1047
0774 1000
0775 3574
0776 3575
0777 3572
                PAGE
                /
                /ROUTINE TO SELECT DRIVE NO.
                /SEQUENTIAL SELECTION 0,1,2,3,0,1,ETC.
                /
1000 2115      POLNEX, ISZ   POLDISK /UPDATE POLE POINTER
1001 1115      TAD   POLDISK      /GET POINTER
1002 4436      SELCHK
1003 5200      JMP   POLNEX      /CHECK IF DISK ON SYSTEM,
1004 1115      TAD   POLDISK      /NO, TRY NEXT DRIVE
1005 7112      CLL   RTN
1006 0061      AND   K0001
    
```

```

1007 3122      DCA   EXREG          /SET EXT, DRIVE BIT

                /
                /DRIVE COMPLETED, START
                /WRITE SEQUENCE
                /SELECT DISK ADDRESS.
                /
1010 1115      GOTIT, TAD   POLDISK /GET DRIVE NO.
1011 0062      AND   K0003          /MASK
1012 7104      CLL   RAL          /MOVE TO 9=10.
1013 1136      TAD   INTCH        /ADD IN OTHER.
1014 3136      DCA   INTCH        /SAVE INITIAL COMMAND.
1015 1777*     TAD   TRKFLG       /GET TRACK FLAG
1016 7650      SNA   CLA          /WAS IT SET????
1017 4440      RANGEN          /GET RANDOM DA.
1020 1141      TAD   SPTRK1       /GET OPERATOR CONSTANT INPUT.
1021 0061      AND   K0001          /MASK EXT. BIT.
1022 1136      TAD   INTCH        /ADD IN OTHER.
1023 3136      DCA   INTCH        /SAVE COMPLETE INITIAL COMMAND.
1024 1777*     TAD   TRKFLG       /GET TRACK FLAG
1025 7650      SNA   CLA          /WAS IT SET????
1026 4440      RANGEN          /USE RANDOM
1027 1162      TAD   SPTRK2       /GET INPUT
1030 3124      DCA   INTDA        /SAVE INITIAL DA.
1031 1777*     TAD   TRKFLG
1032 7640      SZA   CLA          /INPUT BY OPERATOR?
1033 5247      JMP   RENRT        /LET HIM FAIL??
1034 1136      TAD   INTCH
1035 7010      RAR
1036 7620      SNA   CLA          /EXT BIT SET?
1037 5247      JMP   RENRT        /NO, DON'T LIMIT DA.
1040 1145      TAD   MAXTRK
1041 1124      TAD   INTDA
1042 7630      SZA   CLA          /BEOND MAXIMUM LIMIT?
1043 5247      JMP   RENRT        /NO, DONT LIMIT.
1044 1124      TAD   INTDA
1045 7040      CMA
1046 3124      DCA   INTDA        /YES, SET TO LEGAL LIMIT.

                /
                /WRITE INFORMATION
                /CLEAR BUFFER ON THE FLY
                /
1047 4441      RENRT, DISKGO
1050 4400      4400
1051 5263      JMP   REREAD      /GO WRITE
1052 7340      CLA   CLL   CMA    /WRITE DATA POINTER
1053 3165      DCA   ERPLG      /WRITE O.K.
1054 4442      RESRAN          /SET WRITE ERROR FLAG
1055 2137      ISZ   STATRY      /RESET GENERATOR
1056 5776*     JMP   REFILL      /UPDATE WRITE RE=TRY
                /
                /CHECK FOR LOOP ON WRITE
                /
1057 4406      LAS
1060 7710      TRYTIM, SPA   CLA   /GET SWITCH @
1061 5775*     JMP   REFILL-2     /LOOP ON WRITE????
                /YES, TRY WRITE AGAIN
    
```

```

1062 5351 JMP STRREL /RESMORE ALL UNIVES ON ERROR
1063 1200 RENEAD, TAD TRYTIM
1064 3171 DCA TRYCNT /SETUP FOR SOFT ERROR RETRY
1065 3165 DCA ERFLG /CLEAR ERROR FLAG
1066 1106 TAD M4
1067 3137 DCA STATRY /SETUP TRY COUNTER
1070 1106 TAD M4
1071 3140 DCA DATTRY /SETUP TRY COUNTER
1072 3155 DCA CMCNT /CLEAR CRC COUNTER!!!!

/READ INFORMATION/
/CHECK BUFFER ON THE FLY/

1073 4441 RDTRY, DISKGO /READ DATA
1074 0400 0400 /READ DATA POINTER
1075 7610 SKP CLA /DATA READ O.K.
1076 5305 JMP RDSTA /STATUS ERROR
1077 3155 DCA CRCNT /CLEAR CRC COUNTER/

/CHECK DATA ON NO STATUS ERRORS/

1100 4774* JMS DTCHK /CHECK DATA
1101 5324 JMP RENUN /DATA O.K.
1102 2140 ISZ DATTRY /UPDATE READ RE-TRY
1103 5273 JMP RDTRY /TRY AGAIN
1104 5323 JMP RERUN-1 /TRY TO SEEK IT
1105 1121 RDSTA, TAD STNEG /GET STATUS READ
1106 0065 AND K0010 /MASK CRC
1107 7450 SNA /CRC ERROR????
1110 5320 JMP UPTRY /NO, TRY READ AGAIN
1111 3156 DCA CRCFLG /YES, SET FLAG
1112 2155 ISZ CRCNT /UPDATE CRC POINTER

/CHECK DATA AFTER CRC ERROR/

1113 4774* JMS DTCHK /CHECK DATA
1114 7610 SKP CLA /IS A HARD ERROR/
1115 7340 CLA CLL CMA /SET RETRY COUNTER/
1116 3165 DCA ERFLG /SETUP FOR 64 RETRYS IF AC=7777
1117 7410 SKP /CHECK ON RETRY!!!!
1120 3155 UPTRY, DCA CRCNT
1121 2137 ISZ STATRY /UPDATE TRY POINTER
1122 5273 JMP RDTRY /TRY AGAIN
1123 3165 DCA ERFLG /IS A HARD ERROR
1124 3155 RENUN, DCA CRCNT /CLEAR CRC COUNT
1125 3156 DCA CRCFLG /CLEAR CRC FLAG
1126 4773* JMS CKTIM /CHECK TIME POINTERS
1127 1105 TAD ERFLG
1130 7650 SNA CLA /IS IT 64 RETRYS FOR SOFT ERROR?
1131 5334 JMP .+3 /NO DON'T BOTHER
1132 2171 ISZ TRYCNT /YES, UPDATE RETRY COUNTER
1133 5266 JMP RENEAD+3 /TRY AGAIN

/CHECK FOR LOOP ON READ/

```

```

1134 4406 LAS /GET SWITCH 1
1135 7104 CLL RAL
1136 7710 SPA CLA /LOOP????
1137 5263 JMP RENEAD /YES, LOOP
1140 1137 TAD STATRY /TEST FOR HARD ERROR
1141 7650 SNA CLA
1142 5351 JMP STRREL /YES
1143 3165 DCA ERFLG /CLEAR ERROR FLAG

/CHECK FOR TYPE STATUS
/REPORT/

1144 4406 LAS
1145 0017 AND K0400 /MASK
1146 7640 SZA CLA /TYPE STATUS REPORT????
1147 4772* JMS TPSTA /YES
1150 5771* JMP RUN /DU NEXT DRIVE

/RESTORE DRIVE AFTER ERROR/

1151 1123 STRREL, TAD CMREG /GET DRIVE NO.
1152 4443 RECAL /RESTORE
1153 5771* JMP RUN /START NEXT DRIVE

/Routine TO DETERMINE IF TIMING NEEDS TO BE FOR APT SYSTEM,

1154 0000 XKTICK, 0
1155 4424 CHK22 /TEST FOR APT
1156 7410 SKP
1157 5754 JMP I XKTICK /NO, RETURN TO NORMAL RUN
1160 6201 CDF 0
1161 2150 ISZ CLKCNT /LONG COUNTER FOR APT
1162 5366 JMP EXTICK /NORMAL RETURN
1163 1100 TAD K7760 /INIT COUNTER
1164 3150 DCA CLKCNT
1165 4770* JMS KTIME /NOTIFY APT
1166 6201 EXTICK, CDF 0
1167 5754 JMP I XKTICK
1170 0530
1171 0600
1172 3000
1173 2450
1174 1600
1175 0721
1176 0723
1177 3573

PAGE
/ENROR HANDLER/
/UPDATE "SOFT" ON "HARD" TALLYS/
/PRINT ERROR TEXT AND DATA/
/CHECK INHIBIT ENROR SW/

1200 0000 ERNO, 0
1201 7001 IAC /UPDATE AC FLAG

```

```

1202 3374          DCA  PCNTR2          /SAVE NON-RECOVERABLE POINTER;
/COMPUTE WAY TO "HARD"/"SOFT" TALLYS;
1203 1377          TAD  K7773
1204 3375          DCA  PCNTR3          /LINE COUNTER
1205 1123          TAD  CMREG          /GET LAST COMMAND
1206 0063          AND  K0006          /MASK DRIVE NUMBER
1207 7170          CLL  CML  CMA  RAR
1210 3373          DCA  PCNTR1          /SETUP COUNTER
1211 1062          TAD  K0003
1212 2373          ISZ  PCNTR1
1213 5211          JMP  0=2          /COMPUTE WAY TO BUFFER
1214 1153          TAD  STAPOT
1215 3373          DCA  PCNTR1          /PINTER TO BUFFER

/DETERMINE IF ERROR IS "HARD" OR "SOFT";
1216 1156          TAD  CRCFLG          /GET CRC FLAG
1217 7650          SNA  CLA          /CRC ERROR????
1220 5251          JMP  NTSOFT
1221 1600          TAD  I  ERRO          /NO, WAS DEFINITLY A HARD ERROR;
1222 7650          SNA  CLA          /GET ERROR POINTER;
1223 5255          JMP  NTERR          /WAS IT FIRST TIME?
1224 1125          TAD  DAREG          /NO ERROR, ADDITIONAL CRC DATA;
1225 0066          AND  K0017          /COMPARE FAILING SECTOR TO
1226 7041          CIA          /SECTOR WHERE DATA ERROR
1227 1131          TAD  ASREG          /OCCURRED;
1230 7640          SZA  CLA          /SAME SECTOR?
1231 5251          JMP  NTSOFT          /NO, "HARD" ERROR
1232 7340          CLA  CLL  CMA
1233 1155          TAD  CRCCNT          /GET CRC COUNTER
1234 7450          SNA          /WAS THIS FIRST POSSIBLE "SOFT"?
1235 5245          JMP  SOFT          /YES, UPDATE "SOFT" TALLY;
1236 1110          TAD  K7775          /CHECK IF NONRECOVERABLE "SOFT";
1237 7650          SNA  CLA          /WAS IT?
1240 2373          ISZ  PCNTR1          /NO, DUMP "SOFT" TALLY;
1241 1773          TAD  I  PCNTR1          /OTHERWISE DUMP "HARD" TALLY;
1242 7440          SZA          /DUNT GO BACK WARDS!!!!!!
1243 1101          TAD  K7777          /DUMP APPROPRIATE TALLY!!
1244 5254          JMP  NTERR=1          /DUMP IT;
1245 1101          SOFT, TAD  K7777
1246 1773          TAD  I  PCNTR1          /REDUCE HARD ERROR COUNT
1247 3773          DCA  I  PCNTR1
1250 2373          ISZ  PCNTR1          /YES, UPDATE POINTER
1251 1101          NTSOFT, TAD  K7777
1252 2773          ISZ  I  PCNTR1          /UPDATE ERROR COUNT
1253 7610          SKP  CLA
1254 3773          DCA  I  PCNTR1          /HOLD AT 7777

/CHECK INHIBIT SW;
1255 4423          NTERR, JMS  I  KAERRO          /REPORT ERROR TO APT IF REQUIRED
1256 4406          LAR
1257 7106          CLL  RTL
    
```

```

1260 7710          SPA  CLA          /INHIBIT ERROR????
1261 5356          JMP  ERROEX+1          /YES

/CHECK FOR NO HEADER ON SECOND DATA ERROR;
1262 1600          DOMEAD, TAD  I  ERRO          /GET TEXT POINTER
1263 7650          SNA  CLA          /DATA ERROR?
1264 5355          JMP  ERROEX          /EXIT

/TYPE ERROR MESSAGE;
1265 4457          CRLF
1266 4457          CRLF
1267 1374          TAD  PCNTR2          /GET NON-RECOV. FLAG
1270 7640          SZA  CLA          /WAS IT SET
1271 5275          JMP  0=4          /NO DON'T TYPE IT
1272 7340          CLA  CLL  CMA
1273 4455          PRNTER          /PRINT "NON-RECOVERABLE "
1274 3335          MES4
1275 1600          TAD  I  ERRO          /GET TEXT POINTER;
1276 1376          TAD  HEDTAD          /MAKE ERROR HEADER POINTEN;
1277 3120          DCA  PCNEG          /SAVE POINTER;
1300 1520          TAD  I  PCNEG          /GET CORRECT TEXT;
1301 3304          DCA  0=3
1302 7340          CLA  CLL  CMA
1303 4455          PRNTER          /PRINT HEADER
1304 7402          HLT
1305 7340          CLA  CLL  CMA
1306 4455          PRNTER          /PRINT "ERROR"
1307 3303          MES0
1310 4457          CRLF
1311 1200          TAD  ERRO
1312 3120          DCA  PCNEG          /SAVE PC
1313 2200          ISZ  ERRO
1314 1600          TAD  I  ERRO
1315 3371          DCA  ESAVE
1316 2200          ISZ  ERRO          /UPDATE FOR RETURN
1317 1172          TAD  XTXT
1320 3374          DCA  PCNTR2
1321 1372          TAD  XREG
1322 3010          DCA  AUTD10
1323 1105          TAD  M12
1324 3373          DCA  PCNTR1          /COUNTER FOR # OF HEADS
1325 1371          STHAUT, TAD  ESAVE          /GET TEXT POINTER
1326 7500          SNA
1327 5363          JMP  NOTEX          /NUT THIS ONE
1330 7104          CLL  RAL
1331 3371          DCA  ESAVE
1332 2375          ISZ  PCNTR3          /UPDATE LINE FILL COUNTER
1333 7610          SKP  CLA          /NO CRLF
1334 4457          CRLF
1335 1374          TAD  PCNTR2          /GET TEXT MESSAGE POINTER
1336 2374          ISZ  PCNTR2
1337 2374          ISZ  PCNTR2
1340 3343          DCA  0=3          /STORE FOR PRNTER
    
```

```

1341 7340      CLA CLL CMA
1342 4455      PRNTR
1343 7402      HLT                /PRINT XX1
1344 1410      TAD I      AUTO10  /MODIFIED TEXT POINTER
1345 4456      OCTEL
1346 2373      AGAIN, ISZ      PCNTR1  /PRINT FOUR OCTAL
1347 5325      JMP      STMAUT
1350 1520      TAD I      PCNEG
1351 1106      TAD      M4
1352 7650      SNA CLA
1353 4573      JMS I      PRNDAT  /FIRST DATA ERROR?
1354 5360      JMP      .+4       /YES, PRINT DATA
1355 4573      ERROEX, JMS I      PRNDAT
1356 2200      ISZ      ERMO
1357 2200      ISZ      ERMO
1360 7301      CLA CLL IAC
1361 4454      CLRALL
1362 5600      JMP I      ERMO
1363 7104      NOTEX,  CLL RAL
1364 3371      DCA      ESAVE
1365 2374      ISZ      PCNTR2
1366 2374      ISZ      PCNTR2
1367 2010      ISZ      AUTO10
1370 5346      JMP      AGAIN

/
1371 0000      ESAVE,  0
1372 0117      XREG,  PCREG=1
1373 0000      PCNTR1, 0
1374 0000      PCNTR2, 0
1375 0000      PCNTR3, 0
1376 1377      HEDTAD, BUFPNT=1
1377 7773      K7773, 7773

/
1400 1400      PAGE
/
/POINTERS FOR TEXT INFORMATION/
/
1400 3247      BUFPNT, ERTX1
1401 3255      ERTX2
1402 3264      ERTX3
1403 3276      ERTX4

/
/ROUTINE TO CHECK FOR CLASSIC AND LIMIT
/TRANSFERS TO FIELD 0 IF AVAILABLE.
/
1404 0000      CLAFLD, 0
1405 1022      TAD      22
1406 0017      AND      K0400
1407 7650      SNA CLA
1410 5604      JMP I      CLAFLD  /ON CLASSIC?
1411 7340      CLA CLL CMA      /NO.
1412 3143      DCA      MAXFLD
1413 5604      JMP I      CLAFLD  /LIMIT TO FIELD 0.

/
/ROUTINE TO DO CRLF

```

```

1414 0000      UPONE,  0
1415 7300      CLA CLL
1416 1224      TAD      K0215
1417 4445      TYPE
1420 1225      TAD      K0212
1421 4445      TYPE
1422 4445      TYPE
1423 5614      JMP I      UPONE

/
1424 0215      K0215, 0215
1425 0212      K0212, 0212

/
/ROUTINE TO PRINT FOUR OCTAL
/
1426 0000      FROCT,  0
1427 7006      RTL
1430 7006      RTL
1431 3214      DCA      UPONE
1432 1106      TAD      M4
1433 3250      DCA      PRN
1434 1214      TAD      UPONE
1435 0064      AND      K0007
1436 1067      TAD      K0260
1437 4445      TYPE
1440 1214      TAD      UPONE
1441 7006      RTL
1442 7004      RAL
1443 3214      DCA      UPONE
1444 2250      ISZ      PRN
1445 5234      JMP      .+11
1446 4431      SPACE
1447 5626      JMP I      FROCT

/
/SUBROUTINE TO PRINT TEXT
/
1450 0000      PRN,  0
1451 7650      SNA CLA
1452 4457      CRLF
1453 1650      TAD I      PRN
1454 2250      ISZ      PRN
1455 3226      DCA      FROCT

/TYPE CRLF
/YES!!!!
/GET POINTER

1456 7300      MKPRN,  CLA CLL
1457 1626      TAD I      FROCT
1460 0077      AND      K7700
1461 7450      SNA
1462 5304      JMP      EXIT
1463 7500      SNA
1464 7020      CML
1465 7001      IAC
1466 7012      RTR
1467 7012      RTR

```

```

1470 7012      RTR
1471 4445      TYPE
1472 1626      TAD I   FRUCT
1473 0102      AND     K0077
1474 7450      SNA
1475 5304      JMP     EXIT
1476 1313      TAD     K3740
1477 7500      SNA
1500 1074      TAD     K4100
1501 4431      SPACE                   /SPACE OUT 1
1502 2226      ISZ     FRUCT
1503 5256      JMP     MRPRN           /MORE TO PRINT
1504 7300      EXIT,   CLA CLL
1505 5650      JMP I   PRN
/
/ROUTINE TO SPACE OUT 1
/
1506 0000      SPAC,   0
1507 1312      TAD     K0240
1510 4445      TYPE
1511 5706      JMP I   SPAC
/
1512 0240      K0240, 240
1513 3740      K3740, 3740
/
/THIS ROUTINE WILL BE A SKIP INSTRUCTION FOR SYSTEMS WITHOUT CLASSIC
/OTHERWISE IT WILL EXECUTE THE NEXT INSTRUCTION IN FIELD 0 AND THEN
/SKIP THE INSTRUCTION AFTER THAT ONE.
/
1514 0000      CLASIK, 0
1515 3345      DCA     SAVEAC           /SAVE CURRENT AC.
1516 1714      TAD I   CLASIK         /GET INSTRUCTION TO EXECUTE,
1517 3344      DCA     ROUTHMP        /SAVE IT.
1520 2314      ISZ     CLASIK
1521 1022      TAD     UP2
1522 0017      AND     K0400
1523 7640      SZA CLA                   /ARE WE ON CLASSIC?
1524 5327      JMP     ,+J             /YES.
1525 1345      TAD     SAVEAC         /NO, THEN
1526 5714      JMP I   CLASIK         /EXIT.
1527 2314      ISZ     CLASIK
1530 6211      CDF     10
1531 1020      TAD     SWR
1532 3777      DCA I   (SWR)           /SAVE SWITCH REGISTER.
1533 1021      TAD     UP1
1534 3776      DCA I   (DP1)          /SAVE CONTROL 1.
1535 1022      TAD     OP2
1536 3775      DCA I   (OP2)
1537 1344      TAD     ROUTHMP
1540 3774      DCA I   (ROUTINS)      /SAVE ROUTINE IN FIELD 1.
1541 1345      TAD     SAVEAC         /GET BACK AC.
1542 6212      CDF     10
1543 5774      JMP I   (ROUTINS)      /GO AND EXECUTE INSTRUCTION,
/
1544 0000      ROUTHMP, 0

```

```

1545 0000      SAVEAC, 0
/ROUTINE TO GET THE SWITCHES,
/
1546 0000      MYLAS, 0
1547 4407      CLASIK                   /CHECK FOR CLASSIC.
1550 4425      COCKSW                   /GET SWITCHES.
1551 7604      7604
1552 5746      JMP I   MYLAS           /EXIT.
/
/ROUTINE TO RESET REGISTERS FOR ERROR PRINTER
/
1553 0000      SETREG, 0
1554 1073      TAD     K4000           /GET STATUS
1555 3121      DCA     STREG           /SAVE FOR ERROR PRINTER
1556 7340      CLA CLL CMA           /DECREASE BY 1
1557 1111      TAD     TRASH1         /GET SECTOR POINTER
1560 0066      AND     K0017
1561 1112      TAD     TRASH2         /ADD IN ADDRESS
1562 3125      DCA     DAREG           /SAVE FOR ERROR PRINTER
1563 1170      TAD     FIXTIM         /CHECK IF FIRST SECTOR?
1564 7640      SZA CLA                   /IF 0, DON'T UPDATE COMMAND!
1565 5753      JMP I   SETREG         /NO, DON'T!
1566 1174      TAD     SAVCM          /GET COMMAND REG.
1567 3123      DCA     CMREG         /SAVE FOR ERROR PRINTER
1570 5753      JMP I   SETREG         /RETURN
/
1574 1302
1575 0022
1576 0021
1577 0020
1600 1600      PAGE
/ROUTINE TO CHECK DATA READ
/
1600 0000      DTCHK, 0
1601 1156      TAD     CRCFLG         /GET CRC FLAG
1602 7640      SZA CLA                   /WAS IT SET?
1603 5212      JMP     WRDCHK         /YES, THEN WORD BY WORD CHECK!!!
1604 1142      TAD     FNDUSUM        /GET CHECK SUM FOUND
1605 7041      CIA
1606 1141      TAD     CHRSAV         /COMPARE TO GOOD VALUE SAVED
1607 7650      SNA CLA                   /WERE THEY THE SAME
1610 5600      JMP I   DTCHK         /YES, DATA O.K.
1611 7340      CLA CLL CMA
1612 3406      WRDCHK, DCA I XEMRO     /SETUP CHECKSUM ERROR FLAG
1613 1123      TAD     CMREG
/
1614 0015      AND     K0100
1615 7640      SZA CLA                   /HALF BLOCK SET??
1616 1016      TAD     K0200         /YES!
1617 1104      TAD     K7400
1620 3112      DCA     TRASH2
1621 1112      TAD     TRASH2
1622 7040      CMA

```

```

1623 3314      DCA  MSKER
1624 7340      CLA CLL CMA
1625 3142      DCA  FNDSUM
1626 4442      RESRAN
1627 1130      TAD  FWREG
1630 4435      SETFLD
1631 3246      DCA  GOCDF
1632 1112      TAD  TRASH2
1633 3361      DCA  RSRAN
1634 1124      TAD  INTDA
1635 3353      DCA  STGEN
1636 1361      DTR1, TAD  RSRAN
1637 0314      AND  MSKER
1640 3132      DCA  WAREG
1641 1353      TAD  STGEN
1642 0066      AND  K0017
1643 3131      DCA  ASREG
1644 4427      RANDAT
1645 3134      DCA  DGREG
1646 7402      GOCDF, HLT/CDP
1647 1411      TAD I  AUTO11
1650 6201      CDF  0
1651 3135      DCA  DBNEG
1652 1011      TAD  AUTO11
1653 3133      DCA  ADREG
1654 1135      TAD  DBREG
1655 7041      CIA
1656 1134      TAD  DGREG
1657 7650      SNA CLA
1660 5272      JMP  NOERR
1661 2142      ISZ  FNDSUM
1662 5310      JMP  NTRKRS
1663 1156      TAD  CRCFLG
1664 7650      SNA CLA
1665 1140      TAD  DATTRY
1666 2200      ISZ  DTCHK
1667 4446      ERROR
1670 0004      0004
1671 7760      7760
1672 2361      NOERR, ISZ  RSRAN
1673 5300      JMP  +5
1674 2353      ISZ  STGEN
1675 7000      NOP
1676 1112      TAD  TRASH2
1677 3361      DCA  RSRAN
1678 2117      ISZ  BUPTAL
1679 5236      JMP  DTN1
1682 1446      TAD I  XENRO
1683 7650      SNA CLA
1684 3155      DCA  CRCCNT
1685 2446      ISZ I  XENRO
1686 5600      JMP I  DTCHK
1687 4576      BADHLT, JMS I  BGHLT
1688 4446      NTRKRS, ERROR
1689 0000      0000
1690 0000      0000
1691 0000      0000
1692 0000      0000
1693 0000      0000
1694 0000      0000
1695 0000      0000
1696 0000      0000
1697 0000      0000
1698 0000      0000
1699 0000      0000
1700 0000      0000
1701 0000      0000
1702 0000      0000
1703 0000      0000
1704 0000      0000
1705 0000      0000
1706 0000      0000
1707 0000      0000
1708 0000      0000
1709 0000      0000
1710 0000      0000
1711 0000      0000
1712 0000      0000
1713 5272      /
1714 0000      /MSKER, 0
1715 0000      /ROUTINE TO GENERATE RANDOM NUMBERS
1716 0000      /
1717 0000      RANDOM, 0
1718 7301      CLA CLL IAC
1719 1373      TAD  RAD1
1720 1374      TAD  RAD2
1721 1375      TAD  RAD3
1722 3373      DCA  RAD1
1723 7004      RAL
1724 1373      TAD  RAD1
1725 1374      TAD  RAD2
1726 1375      TAD  RAD3
1727 3374      DCA  RAD2
1728 7004      RAL
1729 1373      TAD  RAD1
1730 1374      TAD  RAD2
1731 1375      TAD  RAD3
1732 3375      DCA  RAD3
1733 1375      TAD  RAD3
1734 3375      DCA  RAD3
1735 1375      TAD  RAD3
1736 5715      JMP I  RANDOM
1737 0000      /EXIT, RANDOM NUMBER IN AC
1738 0000      /
1739 0000      /GENERATOR FOR RANDOM DATA
1740 0000      /
1741 0000      GNDAT, 0
1742 7301      CLA CLL IAC
1743 1367      TAD  RAN1
1744 1370      TAD  RAN2
1745 7106      CLL RTL
1746 3367      DCA  RAN1
1747 1370      TAD  RAN2
1748 7012      RTR
1749 1367      TAD  RAN1
1750 3370      DCA  RAN2
1751 1370      TAD  RAN2
1752 5737      JMP I  GNDAT
1753 0000      /
1754 0000      /ROUTINE TO SAVE RANDOM GENERATOR
1755 0000      /
1756 0000      STGEN, 0
1757 1367      TAD  RAN1
1758 3371      DCA  SAV1
1759 1370      TAD  RAN2
1760 3372      DCA  SAV2
1761 5753      JMP I  STGEN
1762 0000      /
1763 0000      /ROUTINE TO RESET RANDOM GENERATOR
1764 0000      /
1765 0000      RSRAN, 0
1766 1371      TAD  SAV1
1767 3367      DCA  RAN1

```

/SET FIRST TIME FLAG
/NO, SETUP RANDOM GENERATOR
/GET FINAL WC
/GET AUTO11+ BUPTAL+ FIELD
/SAVE FIELD CDF

/GENERATE DATA
/SAVE GOOD DATA POINTER
/CDF TO BUFFER FIELD
/GET BAD DATA WORD
/NUMB DF
/SAVE BAD WORD
/GET ADDRESS
/SAVE FOR PRINTER
/GET DATA READ

/COMPARE TO GOOD VALUE
/WERE THEY THE SAME
/YES, NO ERROR
/FIRST TIME PRINT????
/NO, JUST ADDRESS AND DATA
/GET CRC FLAG
/IF SET NO NON-RECOVERABLE.
/NO, GET NON-RECOVERABLE FLAG.
/UPDATE FOR ERROR RETURN
/ENRDR DATA
/PRINTER
/POINTER

/UPDATE BUFFER TALLY
/MORE WORDS TO CHECK
/GET ERROR INDICATOR!
/HAS THERE AN ERROR?
/NO, CLEAR CRC COUNTER
/CHECK FOR COMPUTER ERROR?
/ALL O.K.
/COMPUTER MUST BE DOWN, CHECKSUM
/OTHER ERRORS IN BUFFER

```

1712 0000      0000
1713 5272      JMP  NOERR
1714 0000      /
1715 0000      /MSKER, 0
1716 0000      /ROUTINE TO GENERATE RANDOM NUMBERS
1717 0000      /
1718 0000      RANDOM, 0
1719 7301      CLA CLL IAC
1720 1373      TAD  RAD1
1721 1374      TAD  RAD2
1722 1375      TAD  RAD3
1723 3373      DCA  RAD1
1724 7004      RAL
1725 1373      TAD  RAD1
1726 1374      TAD  RAD2
1727 1375      TAD  RAD3
1728 3374      DCA  RAD2
1729 7004      RAL
1730 1373      TAD  RAD1
1731 1374      TAD  RAD2
1732 1375      TAD  RAD3
1733 3375      DCA  RAD3
1734 1375      TAD  RAD3
1735 3375      DCA  RAD3
1736 5715      JMP I  RANDOM
1737 0000      /EXIT, RANDOM NUMBER IN AC
1738 0000      /
1739 0000      /GENERATOR FOR RANDOM DATA
1740 0000      /
1741 0000      GNDAT, 0
1742 7301      CLA CLL IAC
1743 1367      TAD  RAN1
1744 1370      TAD  RAN2
1745 7106      CLL RTL
1746 3367      DCA  RAN1
1747 1370      TAD  RAN2
1748 7012      RTR
1749 1367      TAD  RAN1
1750 3370      DCA  RAN2
1751 1370      TAD  RAN2
1752 5737      JMP I  GNDAT
1753 0000      /
1754 0000      /ROUTINE TO SAVE RANDOM GENERATOR
1755 0000      /
1756 0000      STGEN, 0
1757 1367      TAD  RAN1
1758 3371      DCA  SAV1
1759 1370      TAD  RAN2
1760 3372      DCA  SAV2
1761 5753      JMP I  STGEN
1762 0000      /
1763 0000      /ROUTINE TO RESET RANDOM GENERATOR
1764 0000      /
1765 0000      RSRAN, 0
1766 1371      TAD  SAV1
1767 3367      DCA  RAN1

```

```

1764 1372 TAD SAV2
1765 3370 DCA RAN2
1766 5761 JMP I RSHAN

1767 1234 RAN1, 1234
1770 5670 RAN2, 5670

1771 0000 SAV1, 0
1772 0000 SAV2, 0
1773 1234 RAD1, 1234
1774 5670 RAD2, 5670
1775 4321 RAD3, 4321

2000 / PAGE
/ROUTINE TO WAIT FOR KEY FROM OPERATOR.

2000 0000 WAIT, 0
2001 6032 KCC
2002 6031 KBF
2003 5202 JMP , -1
2004 6036 KRB
2005 0234 AND K177
2006 1016 TAD K0200
2007 3235 DCA CHKYN /SAVE CHARACTER
2010 1022 TAD 22 /CHECK FOR CLASSIC
2011 0017 AND K0400 /MASK CLASSIC BIT
2012 7650 SNA CLA /CLASSIC=NON ZERO
2013 5226 JMP WAIT1
2014 1235 TAD CHKYN /RESTORE CHAR, FOR CLASSIC
2015 6211 CDF 10
2016 3777* DCA C0CHAR /SAVE CHARACTER,
2017 2776* ISZ INMODE
2020 1777* TAD C0CHAR /GET BACK AC.
2021 6201 CDF 0
2022 4407 CLASIC /CHECK FOR CLASSIC.
2023 4427 C0CNTR /ROUTINE TO EXECUTE.
2024 7000 NOP
2025 7300 CLA CLL /CLEAR CLASSIC AC RETURN
2026 1235 WAIT1, TAD CHKYN /RESTORE CHARACTER
2027 6046 TAD TLR
2030 6041 TBF
2031 5230 JMP , -1
2032 6042 TCF
2033 5600 JMP I WAIT /EXIT

2034 0177 K177, 0177
/ROUTINE TO CHECK FOR YES OR NO

2035 0000 CHKYN, 0
2036 3200 DCA WAIT /SAVE POINTER
2037 1235 TAD CHKYN /GET PC STORED
2040 3260 DCA CHKPOT /SAVE IT

```

```

2041 1200 TAD WAIT
2042 2235 ISZ CHKYN
2043 7041 CIA
2044 1257 TAD K0316
2045 7650 SNA CLA /WAS IT A NO
2046 5635 JMP I CHKYN /YES
2047 1200 TAD WAIT
2050 2235 ISZ CHKYN
2051 7041 CIA
2052 1256 TAD K0331
2053 7650 SNA CLA /WAS IT A YES
2054 5635 JMP I CHKYN /YES
2055 5660 JMP I CHKPOT /WAS NEITHER

/ROUTINE TO CHECK DISK RUN POINTERS

2060 0000 CHKPOT, 0
2061 0064 AND K0007
2062 1154 TAD MUNPOT
2063 3200 DCA WAIT
2064 1600 TAD I WAIT /GET RUN POINTER
2065 7640 SZA CLA /RUN THIS DRIVE
2066 2200 ISZ CHKPOT /NO
2067 5600 JMP I CHKPOT /EXIT

/ROUTINE TO TEST FOR APT AND SET UP APPROPRIATE
/REGISTERS IN UN THE SYSTEM.

2070 0000 APT0, 0
2071 4424 CHK22
2072 5301 JMP , +7 /TEST FOR APT
2073 4407 CLASIC /YES
2074 4431 C0SWIT
2075 7000 NOP
2076 1355 TAD K7000
2077 3775* OCA SKPNOP
2100 5351 JMP EXAPT0 /EXIT
2101 1022 TAD OP2
2102 0354 AND K7577 /NOP CONSOLE PACKAGE
2103 3022 DCA UP2
2104 1355 TAD K7000
2105 3774* DCA MYLAS+3 /NOP SWITCH REGISTER
/NO OPERATOR INTERVENTION ALLOWED

2106 1022 TAD OP2
2107 0064 AND K0007 /GET # OF DRIVES
2110 3111 OCA TRASH1
2111 1022 TAD UP2
2112 0015 AND K0100
2113 7650 SNA CLA /SINGLE DRIVE = NON ZERO AC
2114 5325 JMP M0USKS /NO.

```



```

2115 7301      CLL CLA IAC
2116 3860      DCA      AMOUNT
2117 1111      TAD      TRASH1      /ONLY ONE DRIVE
2120 1154      TAD      KUNPOT      /GET DRIVE NUMBER
2121 3111      DCA      TRASH1
2122 7340      CLL CLA CMA
2123 3511      DCA I   TRASH1      /DU THIS DRIVE
2124 5342      JMP      MENSET
2125 1111      MODSKS, TAD TRASH1
2126 7040      CMA
2127 3112      DCA      TRASH2      /SAVE THE NUMBER OF DRIVES
2130 3111      DCA      TRASH1
2131 1111      TAD      TRASH1
2132 1154      TAD      KUNPOT      /ESTABLISH DRIVE
2133 3113      DCA      TRASH3
2134 7340      CLL CLA CMA
2135 3513      DCA I   TRASH3      /DU THIS DRIVE
2136 2111      ISZ      TRASH1
2137 2000      ISZ      AMOUNT
2140 2112      ISZ      TRASH2      /DONE?
2141 5331      JMP      MODSKS+4      /MORE TO DO
2142 1021      MENSET, TAD  21
2143 7012      RTR
2144 0064      AND      K0007
2145 7104      CLL RAL
2146 7006      RTL
2147 7040      CMA
2150 3143      DCA      MAXFLD      /NEGATIVE AMOUNT OF FIELDS.
2151 2270      EXAPTS, ISZ APT8
2152 2270      ISZ      APT8
2153 5670      JMP I   APT8
/
2154 7377      K7377, 7377
2155 7000      K7000, 7000
/
/THIS ROUTINE WILL NOTIFY APT OF AN ERRUR.
/ONLY THE DRIVE IN ERROR IS ESTABLISHED.
/
2156 0000      AENROR, 0
2157 4424      CHK22
2160 7410      SKP
2161 5756      JMP I   AENROR      /CHECK FOR APT=0,
2162 6002      IOF
2163 7200      CLA
2164 1115      TAD      POLOSK      /DRIVE NUMBER
2165 0064      AND      K0007
2166 6201      CDF 00
2167 6272      CIF 70
2170 5772      JMP I   K6520      /NOTIFY APT
2171 7402      HLT
/
2172 6520      K6520, 6520
2174 1551
2175 0237
2176 1076

```

```

2177 1075      PAGE
2178 2200
/
/ROUTINE TO WRITE OR READ SECTORS SELECTED
/
2200 0000      DSKGO, 0
2201 7340      CLA CLL CMA
2202 3170      DCA      FINTIM      /SETUP FIRST TIME POINTER
2203 3156      DCA      CRCFLG      /CLEAR CRC FLAG
2204 1126      TAD      CAREG      /GET INITIAL CURRENT ADDRESS
2205 4452      LDCUR      /LOAD CURRENT ADDRESS
2206 1127      TAD      WCNEG
2207 3150      DCA      FWNEG      /SETUP FINAL WC
2210 1124      TAD      INTDA      /GET INITIAL STARTING SECTOR
2211 3111      DCA      TRASH1      /SAVE
2212 1124      TAD      INTDA      /GET DISK ADDRESS
2213 0100      AND      K7760      /MASK
2214 3112      DCA      TRASH2      /SAVE
2215 1136      TAD      INTCM      /GET INITIAL COMMAND
2216 1600      TAD I   DSKGO      /GET READ OR WRITE
2217 4451      LDCMD      /LOAD COMMAND
2220 1123      TAD      CMNEG
2221 1075      TAD      K1000
2222 3174      DCA      SAVCM      /MAKE READ ALL OR WRITE ALL
2223 1111      TAD      TRASH1      /SAVE FOR SWITCH TO CONSECUTIVE MODE
2224 0066      AND      K0017      /SECTOR TO DO
2225 1112      TAD      TRASH2      /MASK
2226 4453      LDADD      /ADD TO TRACK
2227 6001      IDN
/
/ROUTINE TO CLEAR OR CHECK SUM BUFFER ON THE FLY
/
2230 3777*     G0BAK, DCA  TIMER2      /CLEAR LONG TIMER
2231 3142      DCA      FNDSUM      /CLEAR SUM CHECK
2232 4435      SETFLD
2233 3254      DCA      CMNCDF      /GET FIELD TO BUFFER
2234 1170      TAD      FINTIM      /SAVE CDF
2235 7650      SNA CLA
2236 5241      JMP      STRHRK      /TIME TO GO
2237 4776*     JMP      TIME
2240 5234      JMP      ==4
2241 1117      STRHRK, TAD BUPTAL      /WAIT FOR FIRST INTERRUPT
2242 7041      CIA
2243 1130      TAD      FWREG      /NOT HERE YET
2244 7450      SNA
2245 5274      JMP      WRKDON      /COMPARE TO SOFTWARE FINAL
2246 7041      CIA
2247 3175      DCA      CLMBAK      /WAIT FOR DISK???
2250 1175      TAD      CLMBAK      /YES!!!!
2251 7041      CIA
2252 1117      TAD      BUPTAL
2253 3117      DCA      BUPTAL
2254 7402      CMNCDF, HLT
2255 1123      TAD      CMNEG      /UPDATE BUFFER TALLY
2256 7700      SMA CLA      /CDF TO BUFFER FIELD
/
/HEAD OR WRITE

```

```

2257 5264      JMP      HASRD      /HAS A HEAD!!
2260 3411      GOCLR, DCA I  AUTO11 /HAS A WRITE, CLEAR BUFFER
2261 2175      ISZ      CLMBAK    /UPDATE TALLY
2262 5260      JMP      GOCLR      /MORE TO CLEAR
2263 5274      JMP      WRKDOON   /DONE WITH SOME
2264 1142      HASRD, TAD      FNDSUM
2265 7100      GOCHK, CLL
2266 1411      TAD I   AUTO11    /GET WORD
2267 7430      SZL
2270 7001      IAC
2271 2175      ISZ      CLMBAK    /UPDATE CLEAR POINTER
2272 5265      JMP      GOCHK      /MORE TO CHECKSUM
2273 3142      DCA      FNDSUM    /SAVE IT
2274 6201      WRKDOON, COF      0
2275 1117      TAD      BUPTAL
2276 7650      SNA CLA
2277 5302      JMP      DSKEK      /LAST WORD DONE????
2300 4776      JMS      TIME      /EXIT
2301 5241      JMP      STRWRK    /TIME AND WAIT
2302 2200      DSKEK, ISZ      DSKGO   /WAIT FOR INT, OR DONE!!!!
2303 5600      JMP I   DSKGO      /EXIT
/
/INTERUPT SERVICE
/
2304 6741      RETURN, DSKP      /DISK SKIP IOT
2305 5353      JMP      NODSKP     /NOT THE DISK
2306 2111      ISZ      TRASH1   /UPDATE SECTOR NUMBER
2307 7000      NOP
2310 1114      TAD      UPUATE     /IT WON'T WORK WITHOUT IT!
2311 1130      TAD      FWREG
2312 3130      DCA      FWREG    /UPDATE WORD COUNT
2313 6745      STATUS, DRST      /READ STATUS
2314 1075      TAD      K4000
2315 7440      SZA
2316 5337      JMP      STATER    /ONLY DONE FLAG?
2317 1130      TAD      FWREG    /STATUS ERROR
2320 7650      SNA CLA
2321 5365      JMP      TRDNE      /LAST TRANSFER?
2322 3170      DCA      FINTIM    /TRANSFER IS DONE
2323 1174      TAD SAVCM    /CLEAR FIRST TIME POINTER
2324 6746      RDLHRL, DLDC      /GET READ OR WRITE COMMAND
2325 1111      TAD      TRASH1   /LOAD COMMAND REGISTER
2326 0066      AND      K0017    /GET SECTOR TO DO
2327 1112      TAD      TRASH2   /MASK OFF
2330 6743      LOGGO, DLAG      /ADD IN TRACK
2331 1167      RETRN, TAD      SVLNK   /LOAD DISK ANFD GO
2332 7110      CLL RAM
2333 1166      TAD      SAVAC    /GET AC
2334 6244      RMF
2335 6001      ION          /RESTORE FIELDS
2336 5400      JMP I   0        /TURN INTERRUPT ON
2337 4775      STATER, JMS      SETREG  /EXIT
2340 1123      TAD      CMREG    /SETUP REGISTERS
2341 7710      SPA CLA
2342 7001      IAC          /WRITE OR READ
/WRIT

```

```

2343 7001      IAC
2344 3347      DCA      ,+3
2345 1137      TAD      STATRY   /MODIFY HEADER POINTER
2346 4446      ERROR      /GET TRY POINTER
2347 0000      B000        /PRINT MESSAGE
2350 7770      7770      /MODIFIED HEADER POINTER
2351 2200      ISZ      DSKGO   /MESSAGE POINTER
2352 5302      JMP      DSKEK   /UPDATE FOR ERROR
2353 3374      NODSKP, DCA      TIMER3 /EXIT
2354 2374      ISZ      TIMER3
2355 5354      JMP      ,=1
2356 4407      CLASIC
2357 4440      COCKPA
2360 7000      NOP
2361 6031      K0F
2362 4576      INTER2, JMS I  BGLT   /KEYBOARD FLAG??
2363 6032      KCC
2364 5331      JMP      RETRN    /ILLEGAL INTERRUPT
2365 4775      TRDNE, JMS      SETREG  /EXIT BACK
2366 3170      DCA      FINTIM    /SETUP REGISTERS
2367 1167      TAD      SVLNK   /CLEAR FIRST TIME POINTER
2370 7110      CLL RAM
2371 1166      TAD      SAVAC    /REPLACE LINK
2372 6244      RMF
2373 5400      JMP I   0        /REPLACE AC
/RESTORE MEMORY FIELDS+ FLAGS
/RETURN TO BACK GROUND
/
2374 0000      TIMER3, 0
/
2375 1553
2376 3123
2377 3141
2400
PAGE
/
/ROUTINE TO GET ONE IN OCTAL
/
2400 0000      OCT1, 0
2401 4444      RECEIV
2402 3354      DCA      LOAD    /RECEIVE
2403 1600      TAD I   OCT1   /SAVE IT
2404 0064      AND      K0007   /GET LIMITS
2405 1067      TAD      K0260  /MASK
2406 7141      CLL CIA
2407 1354      TAD      LOAD    /GET INPUT
2410 7620      SNA CLA
2411 5226      JMP      INERR   /IN LIMITS????
2412 1600      TAD I   OCT1   /NO, ERROR EXIT
2413 0014      AND      K0070  /GET LIMITS
2414 7110      CLL RAM
2415 7012      RTR
2416 1067      TAD      K0260  /MASK
2417 7040      CMA
2420 1354      TAD      LOAD    /GET INPUT
2421 7630      SZA CLA
2422 5226      JMP      INERR   /IN LIMITS????
2423 1354      TAD      LOAD    /NO, ERROR
/GET INPUT

```

```

2424 0064      AND      K0007
2425 2200      ISZ      OCT1      /MASK
2426 2200      INERN, ISZ      OCT1
2427 5000      JMP I   OCT1      /GOOD EXIT
/
/Routine TO RECEIVE FOUR OCTAL
/
2430 0000      OCT4,  0
2431 1106      TAD      M4
2432 3341      DCA      R0ST      /SETUP COUNTER
2433 3350      DCA      LDCA      /START WITH 0
2434 4432      ONEIN
2435 0070      0070
2436 5630      JMP I   OCT4      /RECEIVE ONE OCTAL
2437 1350      TAD      LDCA      /LIMITS
2440 2341      ISZ      R0ST      /ENRUR EXIT
2441 7410      SKP
2442 5246      JMP      .+4      /GET LAST
2443 7004      HAL
2444 7006      RTL
2445 5233      JMP      OCT4+3      /UPDATE COUNTER
2446 2230      ISZ      OCT4
2447 5630      JMP I   OCT4      /EXIT
/EXIT
/EXIT OCTAL IN AC
/
/Routine TO UPDATE AND CHECK FOR PASS COMPLETE
/
2450 0000      CKTIM,  0
2451 1115      TAD      POLDISK
2452 0064      AND      K0007      /SETUP CURRENT DRIVE #
2453 3341      DCA      R0ST      /POINTER
2454 1341      TAD      R0ST
2455 1152      TAD      TIMPOT      /GET TIME POINTER
2456 3354      DCA      LOAD      /SAVE IT
2457 7301      CLA CLL JAC      /ONE FOR 0
2460 1147      TAD      CONSEC      /GET AMOUNT DONE
2461 1754      TAD I  LDAD      /ADD IN AMOUNT COMPLETED SO FAR
2462 3754      DCA I  LDAD
2463 7620      SNL CLA
2464 5650      JMP I   CKTIM      /LINK UP????
2465 4440      RANGEN
2466 3777      DCA      RAN1      /NO, EXIT
2467 4440      RANGEN      /GET RANDOM NUMBER
2470 3776      DCA      RAN2      /RE-PRIME GENERATOR
2471 7100      CLL
2472 1354      TAD      LDAD
2473 1013      TAD      K0004
2474 3354      DCA      LDAD      /SECOND TIME POINTER
2475 2754      ISZ I  LDAD      /UPDATE IT
2476 1754      TAD I  LDAD      /GET COUNT
2477 1144      TAD      MAXTIM      /ADD IN FUDGE FACTOR
2500 7620      SNL CLA      /PASS COMPLETE????
2501 9650      JMP I   CKTIM      /NO, EXIT
2502 3754      DCA I  LDAD      /ZERU SECCUND COUNTER
2503 1341      TAD      R0ST
2504 7040      CMA

```

```

2505 3341      DCA      R0ST      /SETUP COUNTER
2506 1362      TAD      CMPPOT      /ADD IN POINTER
2507 1062      TAD      K0003
2510 2341      ISZ      R0ST      /COMPUTE BUFFER
2511 5307      JMP      .-2
2512 3341      DCA      R0ST      /SAVE ADDRESS POINTER
2513 7340      CLA CLL CMA
2514 2741      ISZ I  R0ST
2515 7610      SKP CLA      /UPDATE PASS COMPLETE POINTER
2516 3741      DCA I  R0ST
2517 4457      CRLF
2520 4455      PRNTER
2521 3477      MES17      /PRINT "DISK"
2522 1115      TAD      POLDISK
2523 0064      AND      K0007      /GET DISK POLE NUMBER
2524 1067      TAD      K0260      /MASK
2525 4445      TYPE
2526 7340      CLA CLL CMA      /TYPE DISK NO.
2527 4455      PRNTER
2530 3502      MES18      /PRINT "PASS COMPLETE"
2531 4406      LAS
2532 0015      AND      K0100      /MASK
2533 7650      SNA CLA      /PASS COMPLETE DISCONNECT????
2534 5337      JMP      .+3      /NO WAY!!!!
2535 4430      DISCON      /DUMP DRIVE
2536 5775      JMP      RUN      /MORE TO TEST!!!!
2537 4774      JMS      TPSTA      /STATUS=COMPLETE TYPEOUT
2540 5650      JMP I   CKTIM      /EXIT
/
/SUBROUTINE TO READ STATUS REGISTER
/
2541 0000      RDST,  0
2542 6745      IOT5,  DRST      /HEAD STATUS IOT
2543 7410      SKP
2544 4576      ERHLT5, JMS I  BGHLT      /SKIP TRAP
2545 3121      DCA      STREG      /SAVE RESULTS
2546 1121      TAD      STREG
2547 5741      JMP I   RDST      /EXIT
/
/SUBROUTINE TO LOAD CURRENT ADDRESS REGISTER
/
2550 0000      LDCA,  0
2551 6744      IOT4,  DLCA      /LOAD CURRENT ADDRESS IOT
2552 4425      KTICK
2553 5750      JMP I   LDCA      /NOTIFY APT
/EXIT
/
/SUBROUTINE TO LOAD TRACK ADDRESS REGISTER
/
2554 0000      LDAD,  0
2555 3125      DCA      DAMEG
2556 1125      TAD      DAMEG
2557 6743      IOT3,  DLAG      /LOAD DISK ADDRESS REGISTER
2560 5754      JMP I   LDAD      /EXIT
2561 4576      ERHLT3, JMS I  BGHLT      /ENRUR SKIP TRAP

```

```

2562 3541 /
2574 3000 / CMPPDT, D0CMP-3
2575 0600 /
2576 1770 /
2577 1767 /
2600 2600 / PAGE
/
/ROUTINE TO GET RANDOM OR OPERATOR DATA
/
2600 0000 RNWRD, 0
2601 7402 SWDAT, HLT /MODIFIED SWITCH
2602 5600 JMP I RNWRD /EXIT
2603 6201 CDF 0 /HOME CDF
2604 1412 TAD I AUTO12 /GET DATA
2605 7402 RECDF, HLT /BUFFER CDF
2606 2116 ISZ OPNTAL /UPDATE TALLY
2607 5600 JMP I RNWRD /EXIT
2610 3220 DCA PRINT /SAVE WORD
2611 1105 TAD M12
2612 3116 DCA OPNTAL /REPLACE TALLY
2613 7340 CLA CLL CMA
2614 1151 TAD DATPOT
2615 3012 DCA AUTO12 /REPLACE AUTO INDEX
2616 1220 TAD PRINT /GET SAVED WORD
2617 5600 JMP I RNWRD /EXIT

/ROUTINE TO TYPE
/
2620 0000 PRINT, 0
2621 3237 DCA DUMP /STORE AC VALUE
2622 4424 CHK22 /SEE IF ON APT
2623 5235 JMP PREXIT /NO, EXIT
2624 1237 TAD DUMP /RETURN AC.
2625 4407 CLASIC /CHECK FOR CLASSIC.
2626 4435 CBTYP /ROUTINE TO EXECUTE.
2627 7410 SKP
2630 5620 JMP I PRINT /EXIT.
2631 6046 TLS
2632 6041 TSB
2633 5232 JMP ,=1
2634 6042 TCF
2635 7200 PREXIT, CLA
2636 5620 JMP I PRINT

/ROUTINE TO DUMP AND REPORT DISK STATUS
/
2637 0000 DUMP, 0
2640 4424 CHK22 /CHECK FOR APT
2641 5637 JMP I DUMP
2642 4455 PRNTER /PRINT "DISK "
2643 3477 MES17
2644 1115 TAD POLDISK
2645 0064 AND K0007 /SETUP CURRENT DRIVE #
2646 3200 DCA RNWRD /SAVE

```

```

2647 1200 TAD RNWRD /GET DISK NUMBER
2650 1067 TAD K0200
2651 4445 TYPE /TYPE DISK NUMBER
2652 7340 CLA CLL CMA
2653 4455 PRNTER /PRINT "DISCONNECTED!"
2654 3445 MES15
2655 4777 JMS TPSTA /TYPE STATUS REPORT
2656 1200 TAD RNWRD
2657 1154 TAD RUNPOT
2660 3200 DCA RNWRD /SAVE POINTER ADDRESS
2661 3600 DCA I RNWRD /CLEAR RUN POINTER
2662 3200 DCA RNWRD
2663 1106 TAD M4
2664 3220 DCA PRINT /CHECK FOR MORE POINTER
2665 1200 TAD RNWRD
2666 4436 SELCHK /CHECK SELECT POINTERS
2667 7610 SKP CLA /DISK NOT HERE
2670 5637 JMP I DUMP /MORE AVAILABLE
2671 2200 ISZ RNWRD
2672 2220 ISZ PRINT /UPDATE POINTERS
2673 5265 JMP ,=6
2674 4457 CRLF
2675 4455 PRNTER /PRINT "DISK"
2676 3477 MES17
2677 7340 CLA CLL CMA
2700 4455 PRNTER /PRINT "SYSTEM DOWN"
2701 3455 MES16
2702 4576 NO08KB, JMS I 0GMLT /ERROR, NO DISK AVAILABLE

/ROUTINE TO SETUP FIELD TO BUFFER+ AUTO11+ BUFFER TALLY
/
2703 0000 STFLD, 0
2704 7041 CIA
2705 1127 TAD WCHEG
2706 3117 DCA BUFTAL
2707 7340 CLA CLL CMA
2710 1126 TAD CAREG /GET INITIAL CA
2711 3011 DCA AUTO11 /SAVE
2712 1137 TAD DATFLG /GET DATA FLAG
2713 7650 SNA CLA /WAS IT SET???
2714 5322 JMP ,+6 /NO, USE REGULAR
2715 1105 TAD M12
2716 3116 DCA OPNTAL /SETUP SPECIAL TALLY
2717 7340 CLA CLL CMA
2720 1151 TAD DATPOT
2721 3012 DCA AUTO12 /SETUP SPECIAL AUTO INDEX
2722 1136 TAD INTCM
2723 0014 AND K0070 /MASK FIELD BITS
2724 1103 TAD KCDF /MAKE BUFFER CDF
2725 3205 OCA RECDF /SETUP SPECIAL CDF
2726 1205 TAD RECDF /GET BACK CDF
2727 5703 JMP I STFLD /EXIT, FIELD IN AC

/ROUTINE TO CHANGE DEVICE IOT CODES
/

```

```

2730 4407  CHANG,  CLASSIC
2731 4431          C88WIT
2732 7000          NOP
2733 4406          LAS
2734 0071          AND      A0770
2735 3776          DCA      LDCM
2736 1300          TAD      CHNPOT
2737 3111          DCA      TRASH1
2740 1357          TAD      CCNTR1
2741 3112          DCA      TRASH2
2742 1511          CHANGR, TAD I TRASH1
2743 3113          DCA      TRASH3
2744 1513          TAD I TRASH3
2745 0072          AND      A7007
2746 1776          TAD      LDCM
2747 3513          DCA I TRASH3
2750 2111          ISZ     TRASH1
2751 2112          ISZ     TRASH2
2752 5342          JMP      CHANGR
2753 4407          CLASSIC
2754 4436          CBERR
2755 7402          CHNHLT, HLT
2756 5775          JMP      BGN

/
2757 7765          CCNTR1, 7765
/
2760 2761          CHNPOT, CHNPOT+1
2761 2304          RETURN
2762 2313          STATUS
2763 2324          RDLWRL
2764 2330          LODGO
2765 0554          IOT0
2766 0752          IOT1
2767 0561          IOT2
2770 2557          IOT3
2771 2551          IOT4
2772 2542          IOT5
2773 0545          IOT6

/
2775 0200
2776 0542
2777 3000
3000 3000          PAGE
/
/ROUTINE TO TYPE STATUS REPORT
/
3000 0000          TPSTA, 0
3001 4424          CHK22
3002 5600          JMP I TPSTA
3003 4457          CRLF
3004 4455          PRNTR
3005 3372          MEB7
3006 1107          TAD      MIB
3007 3245          DCA      TSAVE1
/PRINT "DISK HARD SOFT COMP"
/MAXIMUM TO DO

```

```

3010 3246          DCA      TSAVE2
3011 3247          DCA      TSAVE3
3012 1246          CHKRES, TAD TSAVE2
3013 1062          TAD      K0003
3014 3246          DCA      TSAVE2
3015 1246          TAD      TSAVE2
3016 1153          TAD      STAPOT
3017 3251          DCA      TSAVE5
3020 1247          TAD      TSAVE3
3021 4436          SELCHK
3022 5241          JMP      NOTSTA
3023 4457          CRLF
3024 4431          SPACE
3025 1247          TAD      TSAVE3
3026 1067          TAD      K0260
3027 4445          TYPE
3030 4431          SPACE
3031 4431          SPACE
3032 7346          CLA CLL CMA RTL
3033 3250          DCA      TSAVE4
3034 1651          TAD I TSAVE5
3035 4456          OCTEL
3036 2251          ISZ     TSAVE5
3037 2250          ISZ     TSAVE4
3040 5234          JMP      =-4
3041 2247          NOTSTA, ISZ     TSAVE3
3042 2245          ISZ     TSAVE1
3043 3212          JMP      CHKRES
3044 5000          JMP I TPSTA
/UPDATE DRIVE NUMBER
/MORE TO REPORT
/EXIT

/
3045 0000          TSAVE1, 0
3046 0000          TSAVE2, 0
3047 0000          TSAVE3, 0
3050 0000          TSAVE4, 0
3051 0000          TSAVE5, 0
/
/ROUTINE TO RECALIBRATE SELECTED DRIVE
/DISCONNECT DRIVE ON ERROR!
/
3052 0000          RESTOR, 0
3053 0063          AND      K0006
3054 3200          DCA      TPSTA
3055 1077          TAD      K7700
3056 3341          DCA      TIMER2
3057 2340          ISZ     TIMER1
3060 5257          JMP      =-1
3061 2341          ISZ     TIMER2
3062 5257          JMP      =-3
3063 1200          TAD      TPSTA
3064 4451          LDCMD
3065 7326          CLA CLL CML RTL
3066 4454          CLRALL
3067 4450          DSKSKP
3070 5267          JMP      =-1
3071 4447          ROSTAT
/SAVE DRIVE NUMBER
/SETUP COUNTER
/WAIT FOR DISK TO COOL OFF!
/CURRENT DRIVE
/LOAD COMMAND
/ENABLE RECALIBRATE BIT
/"RECALIBRATE"
/DISK SKIP IOT
/WAIT FOR FIRST DONE FLAG
/HEAD STATUS

```

```

3072 7500 SMA /DONE FLAG SET????
3073 5311 JMP REBERR /NO, ERROR
3074 0076 AND K1777 /MASK OTHER ERROR BITS
3075 7640 SZA CLA /ANY SET????
3076 5311 JMP REBERR /YES, ERROR
3077 4454 RESTA, CLRALL /CLEAR STATUS
3100 1016 TAD K0200 /ENABLE SET SECOND DONE FLAG
3101 1200 TAD TPSTA /ORIGINAL COMMAND
3102 4451 LOCMD /LOAD COMMAND
3103 4450 DSKSKP /DISK SKIP IOT
3104 5303 JMP .-1 /WAIT FOR SECOND DONE
3105 4447 RDSTAT /READ STATUS
3106 1073 TAD K4000
3107 7650 SNA CLA /WAS IT ONLY DONE FLAG
3110 5652 JMP I RESTOR /YES, RETURN
3111 7300 RESERR, CLA CLL
3112 4446 ERROR /ERROR
3113 0003
3114 7500
3115 4457 CRLF
3116 4457 CRLF
3117 4455 PRNTER
3120 3174 /PRINT"RECALIBRATE ERROR DISCONNECT"
3121 4430 DISCON
3122 5652 JMP I RESTOR /DISCONNECT DISK
/MORE DISK AVAILABLE
/
/ROUTINE TO TIME AND WAIT
/
3123 0000 TIME, 0
3124 2340 ISZ TIMER1
3125 5725 JMP I TIME /EXIT
3126 2341 ISZ TIMER2
3127 5725 JMP I TIME /EXIT
3130 4576 INTER1, JMS I 00HLT /NO INTERRUPT OCCURRED, I GUESS;
/
/ROUTINE TO COMBINE ERROR HALTS,
/WHEN THE COMPUTER HALTS THE AC
/IT WILL EQUAL THE PC ON THE FAILING
/HALT INSTRUCTION,
/
3131 0000 BIGHLT, 0
3132 7300 CLA CLL
3133 1331 TAD BIGHLT
3134 4407 CLASIC /LOAD AC WITH PC,
3135 4436 CBERR /CHECK FOR CLASSIC,
3136 7402 BIGHTP, HLT /ROUTINE TO EXECUTE,
3137 5332 JMP .-5 /AC=PC,
/ /NON-RECOVERABLE.
/
3140 0000 TIMER1, 0
3141 0000 TIMER2, 0
/
/ROUTINE TO TYPE OUT DATA INFORMATION
/
3142 0000 TYPDAT, 0
3143 4455 PRNTER /PRINT "AS:"

```

```

3144 3235 TEXAS
3145 1131 TAD ASREG
3146 4456 OCTEL
3147 7340 CLA CLL CMA
3150 4455 PRNTER /PRINT "WA:"
3151 3237 TEXWA
3152 1132 TAD WAREG
3153 4456 OCTEL
3154 7340 CLA CLL CMA
3155 4455 PRNTER /PRINT "AD:"
3156 3241 TEXAD
3157 1133 TAD AOREG
3160 4456 OCTEL
3161 7340 CLA CLL CMA
3162 4455 PRNTER /PRINT "DG:"
3163 3243 TEXDG
3164 1134 TAD DGREG
3165 4456 OCTEL
3166 7340 CLA CLL CMA
3167 4455 PRNTER /PRINT "DB:"
3170 3245 TEXDB
3171 1135 TAD DBREG
3172 4456 OCTEL
3173 5742 JMP I TYPDAT
/
3174 2205 MEB19, TEXT "RECALIBRATE ENRRR DISCONNECT;"
3175 0301
3176 1411
3177 0222
3200 0124
3201 0540
3202 0522
3203 2217
3204 2240
3205 0411
3206 2303
3207 1716
3210 1605
3211 0324
3212 4100
/
3213 2003 TEXPC, TEXT "PC:"
3214 7200
3215 2324 TEXST, TEXT "ST:"
3216 7200
3217 0530 TEXEX, TEXT "EX:"
3220 7200
3221 0315 TEXCM, TEXT "CM:"
3222 7200
3223 1101 TEXIA, TEXT "IA:"
3224 7200
3225 0401 TEXDA, TEXT "DA:"
3226 7200
3227 0301 TEXCA, TEXT "CA:"
3230 7200

```

3231	2703	TEXWC, TEXT	"WC1"
3232	7200		
3233	0627	TEXFW, TEXT	"FW1"
3234	7200		
3235	0123	TEXAS, TEXT	"AS1"
3236	7200		
3237	2701	TEXWA, TEXT	"WA1"
3240	7200		
3241	0104	TEXAD, TEXT	"AD1"
3242	7200		
3243	0407	TEXDG, TEXT	"DG1"
3244	7200		
3245	0402	TEXDB, TEXT	"DB1"
3246	7200		
/			
3247	2205	ERTX1, TEXT	"READ STATUS"
3250	0104		
3251	4023		
3252	2401		
3253	2425		
3254	2300		
3255	2722	ERTX2, TEXT	"WRITE STATUS"
3256	1124		
3257	0540		
3260	2324		
3261	0124		
3262	2523		
3263	0000		
3264	2205	ERTX3, TEXT	"RECALIBRATE STATUS"
3265	0301		
3266	1411		
3267	0222		
3270	0124		
3271	0540		
3272	2324		
3273	0124		
3274	2523		
3275	0000		
3276	0411	ERTX4, TEXT	"DISK DATA"
3277	2313		
3300	4004		
3301	0124		
3302	0100		
/			
3303	4005	ME00, TEXT	" ERROR"
3304	2222		
3305	1722		
3306	0000		
3307	2213	ME01, TEXT	"RK0E/RK0L DATA RELIABILITY"
3310	7005		
3311	5722		
3312	1370		
3313	1440		
3314	0401		
3315	2401		

3316	4022		
3317	0514		
3320	1101		
3321	0211		
3322	1411		
3323	2431		
3324	0000		
3325	0530	ME02, TEXT	"EXERCISE"
3326	0522		
3327	0311		
3330	2305		
3331	0000		
3332	4004	ME03, TEXT	" DISK"
3333	1123		
3334	1300		
3335	1617	ME04, TEXT	"NON-RECOVERABLE "
3336	1655		
3337	2205		
3340	0317		
3341	2605		
3342	2201		
3343	0214		
3344	0540		
3345	0000		
3346	0530	ME05, TEXT	"EXTENDED R/W MEMORY(0-7)?"
3347	2405		
3350	1404		
3351	0504		
3352	4022		
3353	5727		
3354	4015		
3355	0515		
3356	1722		
3357	3150		
3360	6055		
3361	6751		
3362	7700		
/			
3363	0103	ME06, TEXT	"ACCEPT MODE?"
3364	0305		
3365	2024		
3366	4015		
3367	1704		
3370	0577		
3371	0000		
3372	0423	ME07, TEXT	"DSK HARD SOFT COMP"
3373	1340		
3374	1001		
3375	2204		
3376	4023		
3377	1706		
3400	2440		
3401	0317		
3402	1520		

3403	0000		
3404	0611	ME08, TEXT	"FIELD?"
3405	0514		
3406	0477		
3407	0000		
3410	2422	ME09, TEXT	"TRACK?"
3411	0103		
3412	1377		
3413	0000		
3414	0530	ME10, TEXT	"EXTRA SECTORS?"
3415	2422		
3416	0140		
3417	2305		
3420	0324		
3421	1722		
3422	2377		
3423	0000		
3424	0214	ME11, TEXT	"BLOCK LENGTH?"
3425	1703		
3426	1340		
3427	1405		
3430	1607		
3431	2410		
3432	7700		
3433	0401	ME13, TEXT	"DATA?"
3434	2401		
3435	7700		
3436	0122	ME14, TEXT	"ARE YOU SURE?"
3437	0540		
3440	3117		
3441	2540		
3442	2325		
3443	2205		
3444	7700		
3445	4004	ME15, TEXT	" DISCONNECTED!"
3446	1123		
3447	0317		
3450	1616		
3451	0503		
3452	2405		
3453	0441		
3454	0000		
3455	2331	ME16, TEXT	"SYSTEM SHUT DOWN, NO DISKS TO RUN!"
3456	2324		
3457	0515		
3460	4033		
3461	1025		
3462	2440		
3463	0417		
3464	2716		
3465	3440		
3466	1617		
3467	4004		
3470	1123		
3471	1323		

3472	4024		
3473	1740		
3474	2225		
3475	1641		
3476	0000		
3477	0411	ME17, TEXT	"DISK "
3500	2313		
3501	4000		
3502	4020	ME18, TEXT	" PASS COMPLETE!"
3503	0123		
3504	2340		
3505	0317		
3506	1520		
3507	1405		
3510	2405		
3511	4100		
		/	
3512	0000	DSK00, 0	
3513	0000	DSK10, 0	
3514	0000	DSK20, 0	
3515	0000	DSK30, 0	
3516	0000	DSK40, 0	
3517	0000	DSK50, 0	
3520	0000	DSK60, 0	
3521	0000	DSK70, 0	
		/	
3522	0000	D0TH1, 0	
3523	0000	D1TH1, 0	
3524	0000	D2TH1, 0	
3525	0000	D3TH1, 0	
3526	0000	D4TH1, 0	
3527	0000	D5TH1, 0	
3530	0000	D6TH1, 0	
3531	0000	D7TH1, 0	
3532	0000	D0TH2, 0	
3533	0000	D1TH2, 0	
3534	0000	D2TH2, 0	
3535	0000	D3TH2, 0	
3536	0000	D4TH2, 0	
3537	0000	D5TH2, 0	
3540	0000	D6TH2, 0	
3541	0000	D7TH2, 0	
		/	
3542	0000	D0HRD, 0	
3543	0000	D0SOF, 0	
3544	0000	D0CMP, 0	
3545	0000	D1HRD, 0	
3546	0000	D1SOF, 0	
3547	0000	D1CMP, 0	
3550	0000	D2HRD, 0	
3551	0000	D2SOF, 0	
3552	0000	D2CMP, 0	

4000
4100

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

A0170	0750	C0GET	0624	CONCUR	0715	DAT3	0760
A0770	0071	C0HANG	1122	CONSEC	0147	DAT4	0761
A7007	0072	C0INQU	4437	CONSOL	0000	DAT5	0762
AC0CA	0362	C0OCTA	4432	CRCENT	0135	DAT6	0763
ACL	7701	C0PASS	4424	CRCPLG	0136	DAT7	0764
ACSAVE	1345	C0PAUS	4441	CRLF	4437	DAT8	0765
ADREG	0133	C0PRNT	4430	D0CMP	3544	DAT9	0766
AERROR	2156	C0RDPS	0606	D0HRU	3542	DATFLG	0137
AGAIN	1346	C0RETO	0614	D0SOP	3543	DATPDT	0151
ALLAGN	0255	C0RETR	0536	D0TM1	3522	DATTRY	0140
AMOUNT	0060	C0SETD	0613	D0TM2	3532	DBREG	0135
APTS	2370	C0SET8	0535	D1CMP	3547	DCLR	6742
ASKNX1	0345	C0SHIT	4431	D1HRU	3545	DGREG	0134
ASKNX2	0406	C0SMT	0745	D1SOP	3546	DISCON	4430
ASKNX3	0425	C0THP1	1021	D1TM1	3523	DISKGO	4441
ASKNX5	0464	C0TTY1	4426	D1TM2	3533	DLAG	6743
ASKSUR	0513	C0TYPE	4435	D2CMP	3552	DLCA	6744
ASREG	0131	CAF	0007	D2HRU	3550	DLDC	6746
AUTO10	0010	CAREG	0126	D2SOP	3551	DLSC	6740
AUTO11	0011	CCNTR1	2757	D2TM1	3524	DOCNT	0247
AUTO12	0012	CHANG	2730	D2TM2	3534	DOHEAD	1262
BADHLT	1707	CHANGR	2742	D3CMP	3555	DONEA	0426
BGHLT	0176	CHEK22	0523	D3HRU	3553	DOPACK	0212
BGN	0200	CHK22	4424	D3SOP	3554	D0SET	0251
BGNBUF	0146	CHKCLA	1200	D3TM1	3525	DRST	6745
BIGHT	3131	CHKPDT	2000	D3TM2	3535	D8K00	3512
BIGSTP	3136	CHKRES	3012	D4CMP	3560	D8K10	3513
BKRRET	0363	CHKSAV	0141	D4HRU	3556	D8K20	3514
BUPPNT	1400	CHKYN	2035	D4SOP	3557	D8K30	3515
BUPFAL	0117	CHNCDF	2254	D4TM1	3526	D8K40	3516
BYRETR	0506	CHNHLT	2755	D4TM2	3536	D8K50	3517
C0BY1	0230	CHNPOT	2760	D5CMP	3563	D8K60	3520
C0BY2	1300	CKCOUT	0232	D5HRU	3561	D8K70	3521
C0BY3	1061	CKTIM	2450	D5SOP	3562	D8KEY	2302
C0BY4	0515	CLAFLD	1404	D5TM1	3527	D8KGO	2200
C0BY5	1116	GLABIC	4407	D5TM2	3537	D8KPF	6741
C0CHAR	1075	CLASIK	1514	D6CMP	3566	D8KSKP	4430
C0CKP	1022	CLDR	0500	D6HRU	3564	DTCHK	1000
C0CKPA	4440	CLKCNT	0190	D6SOP	3565	UTR1	1636
C0CKSW	4425	CLRALL	4454	D6TM1	3530	DUMP	2037
C0CNTR	4427	CLRBAK	0175	D6TM2	3540	ENDIT	0742
C0CNT	1145	CLRTRN	1315	D7CMP	3571	ERFLG	0165
C0CRLF	4433	CMPPOT	2562	D7HRU	3567	ERHLT0	0556
C0D01	0310	CMREG	0123	D7SOP	3570	ERHLT2	0563
C0D010	1262	CNTRLC	0551	D7TM1	3531	ERHLT3	2561
C0D011	0607	CNTRLD	0600	D7TM2	3541	ERHLT5	2544
C0D02	1033	CNTHLE	0545	DAREG	0125	ERHLT6	0547
C0D03	0350	CNTRLL	0537	DAT1	0756	ERR1	0736
C0D04	1006	CNTRLQ	0300	DAT10	0707	ERRMES	1420
C0D07	0527	CNTRLN	0511	DAT11	0770	ERR0	1200
C0ECHO	4434	CNTRL3	0521	DAT12	0771	ERRDEX	1455
C0ERR	4436	CNTRVAL	0252	DAT2	0757	ERROR	4446

ERTX1	3247	K0001	0001	M12	0105	NTHHKS	1710
ERTX2	3255	K0003	0002	M4	0106	OCT1	2400
ERTX3	3264	K0004	0013	MANUAL	0322	OCT4	2430
ERTX4	3276	K0006	0063	MAXFLD	0143	OCTEL	4456
ESAVE	1371	K0007	0004	MAXTIM	0144	ONEIN	4432
EXAPT8	2191	K0010	0005	MAXTRK	0145	OP1	0021
EXIT	1504	K0017	0006	MEMSET	2142	OP2	0022
EXITA	0440	K0070	0014	MES0	3303	OPRTAL	0116
EXREG	0122	K0077	0102	MES1	3307	PASCNT	0250
EXTICK	1106	K0100	0015	MES10	3414	PCLF	6062
F1OP1	0021	K0200	0016	MES11	3424	PCNTR1	1373
F1OP2	0022	K0212	1425	MES13	3433	PCNTR2	1374
F13NR	0020	K0215	1424	MES14	3436	PCNTR3	1375
FILCNT	1040	K0240	1512	MES15	3445	PCREG	0120
FILLER	1037	K0260	0007	MES16	3455	PCSAVE	1344
FILLUP	0717	K0277	0070	MES17	3477	PNTBUF	1120
FIRTIM	0170	K0316	2057	MES10	3502	POLDSK	0115
FLOFLG	3572	K0331	2056	MES19	3174	POLNEX	1000
FLDMLT	0206	K0400	0017	MES2	3325	PREXIT	2635
FLSAVE	1347	K1000	0075	MES3	3332	PRINT	2620
FNDSUM	0142	K177	2034	MES4	3335	PRN	1450
FORIN	4433	K1777	0076	MES5	3346	PRNDAT	0173
PROCT	1426	K3740	1513	MES6	3303	PRNTER	4455
FWREG	0130	K4000	0073	MES7	3372	PSIE	6065
GENDAT	4426	K4100	0074	MES8	3404	PSKE	6063
GETCH1	0703	K5405	0300	MES9	3410	PSKF	6064
GETDAT	0456	K6500	0541	MES4	0747	PSTB	6064
GNDAT	1737	K6520	2172	MESAC	1533	PTSTOR	0336
GDBAK	2230	K7000	2155	MESFL	1341	RAD1	1773
GDCOF	1646	K7377	2154	MESMAN	1146	RAD2	1774
GDCMK	2265	K7400	0104	MESMU	1336	RAD3	1775
GOCLR	2260	K7700	0077	MESPAS	0253	RAN1	1767
GOITA	0443	K7760	0100	MESPC	1330	RAN2	1770
GOTIT	1010	K7773	1377	MODSRS	2125	RANDAT	4427
GOTOA	0454	K7775	0110	MOA	7501	RANDOM	1715
GTF	0004	K7777	0101	MGL	7421	RANGEN	4440
HEDTAD	1376	KAERRO	0023	MQSAVE	1346	RANJMS	0522
HLFFLG	3575	KCDF	0103	MRPRN	1456	RDLWRL	2324
INDEXA	0455	KRDT	0230	MSKER	1714	ROST	2541
INERR	2426	KSKP	0546	MYAC	1317	RDSTA	1105
INMODE	1076	KTICK	4425	MYLAB	1546	RDSTAT	4447
INTCM	0136	KTIME	0530	NEWRO	0727	RDTRY	1073
INTDA	0124	LAB	4406	NEXT	0201	REALPC	1316
INTER1	3130	LOAD	2554	NODSCKP	2353	RECAL	4443
INTER2	2362	LOADD	4453	NODSRS	2702	RECDF	2605
IO70	0554	LOCA	2550	NOERR	1072	RECEIV	4444
IO71	0752	LDCM	0542	NOBET	0242	REDDA	0415
IO72	0561	LDCMD	4451	NOTEX	1303	REFILL	0723
IO73	2557	LDCUR	4452	NOTSTA	3041	REREAD	1063
IO74	2551	LNKCA	0361	NTCLAS	1270	RERUN	1124
IO75	2542	LDDGO	2330	NTERN	1255	RESERR	3111
IO76	0545	M10	0107	NTSOFT	1251	RESRAN	4442

RESTA	3077	TABLB	0471	XCBCNL	1023		
RESTOR	3052	TEXAD	3241	XCBECH	1063		
RETRN	2331	TEXAS	3235	XCBERK	1207		
RETURN	2304	TEXCA	3227	XCBOING	0055		
REWRT	1047	TEXCM	3221	XCBOCT	1000		
RNFLO	0630	TEXDA	3225	XCSPAS	0200		
RNRWD	2400	TEXDB	3245	XCSPAU	0337		
ROUNS	1302	TEXDG	3243	XCSPNT	0303		
ROUTMP	1544	TEXEX	3217	XCSPSW	0056		
RSRAN	1761	TEXFW	3233	XCBSW	0202		
RUN	0600	TEXIA	3223	XCBTY	0272		
RUNPOT	0154	TEXPC	3213	XCBTYP	1077		
SAV1	1771	TEXST	3215	XCHKZ2	0024		
SAV2	1772	TEXWA	3237	XCHKYN	0037		
SAVAC	0166	TEXWC	3231	XCKPUT	0056		
SAVCH	0174	TIME	3123	XCLAS	0007		
SAVE1	0400	TIMER1	3140	XCLUK	0054		
SAVEAC	1545	TIMER2	3141	XCRLP	0057		
SDKP	0751	TIMER3	2374	XDDLPT	1112		
SECFLG	3574	TIMPOT	0152	XDOSW	0520		
SELCHK	4436	THPCNT	0746	XDSKWD	0041		
SETFLD	4435	TPSTA	3000	XDUMP	0030		
SETGEN	4434	TRASH1	0111	XERRU	0046		
SETREG	1553	TRASH2	0112	XFROCT	0056		
SETUP1	1233	TRASH3	0113	XGNDAT	0026		
SETUP2	0225	TRDNE	2305	XKTCK	0025		
SKPNOP	0237	TRKFLG	3573	XKTICK	1154		
SOFT	1245	TRYCNT	0171	XLAB	0006		
SPAC	1506	TRYTIM	1000	XLDAD	0053		
SPACE	4431	TSAVE1	3045	XLDCA	0052		
SPBLK	0164	TSAVE2	3046	XLDCM	0051		
SPFLD	0160	TSAVE3	3047	XOCT1	0032		
SPSEC	0163	TSAVE4	3050	XOCT4	0053		
SPTRK1	0161	TSAVE5	3051	XPRINT	0045		
SPTRK2	0162	TSTCHA	0715	XPRN	0055		
STAPOT	0153	TTYLPT	1121	XRDST	0047		
STATER	2337	TYPDAT	3142	XREG	1372		
STATRY	0137	TYPE	4445	XNESTR	0043		
STATUS	2313	UPAROW	0615	XRNDOU	0040		
STFLD	2703	UPDATE	0114	XRNWKO	0027		
STGEN	1753	UPONE	1414	XRSRAN	0042		
STPHLT	0003	UPTRY	1120	XSDKP	0050		
STRAUT	1325	WAIT	2000	XSPAC	0031		
STRBUF	3600	WAIT1	2026	XSTFLU	0055		
STREG	0121	WAREG	0132	XSTGEN	0054		
STRREL	1151	WASRD	2204	XTABLA	0457		
STRTEX	0224	WATMES	0651	XTABLB	0400		
STRWRK	2241	WCREG	0127	XTEXT	0172		
SVLNK	0167	WRDCHK	1612	XWAIT	0044		
SHDAT	2601	WRKDDN	2274	YESNU	4437		
SHR	0020	XCBCCKP	1041				
TABLA	0461	XCBCNT	0400				

ERRORS DETECTED: 0

LINKS GENERATED: 163

RUN-TIME: 6 SECONDS

3K CORE USED