

IDENTIFICATION

PRODUCT CODE: MAINDEC-08-DHRKD-D-D
PRODUCT NAME: RK8E/RK8L DISK FORMATTER PROGRAM
DATE RELEASED: FEBRUARY, 1977
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: JOHN VMOBEL/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, 1976, 1977 BY DIGITAL EQUIPMENT CORPORATION

TABLE OF CONTENTS

1.	ABSTRACT
2.	RESTRICTIONS
2.1	HARDWARE
2.2	PROGRAM STORAGE
3.	PRELIMINARY PROGRAMS
4.	OPERATOR AND/OR PROGRAM ACTION
4.1	STANDARD TEST PROCEDURE
4.2	RK05J DRIVE CARTRIDGE MOUNTING PROCEDURE
4.3	RK05F DRIVE SETUP PROCEDURE
4.4	FORMAT PROGRAM
4.5	SWITCH REGISTER SETTINGS
5.	ERRORS
6.	PROGRAM DISCRIPTION
7.	APT-8 HOOKS
8.	PROGRAM LISTING
9.	CONSOLE PACKAGE ADDENDUM

1. ABSTRACT

THE RK8E/RK8L DISK FORMATTER PROGRAM IS DESIGNED TO WRITE AND CHECK THE FORMAT OF THE COMPLETE DISK CARTRIDGE.

ONLY STANDARD DEC SURFACE FORMAT IS AVAILABLE (I.E. SECTORS NUMBERED IN THE NORMAL NUMERICAL SEQUENCE 0, 1, 2, 3, 4, 5, ETC.).

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 IOT0 FOR EXTENDED DRIVES 4-7 WHICH IS NOT AVAILABLE ON THE DW8E.

2.1 HARDWARE

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

NOTE: THE RK05F'S DRIVE IS CONSIDERED AS TWO SEPARATE UNITS. WHEN ANSWERING ALL QUESTIONS EACH SEPARATE UNIT MUST BE SPECIFIED: DSK0?, DSK1?, DSK2?, ETC.

2.2 PROGRAM STORAGE

THE PROGRAM UTILIZES OR OCCUPIES LOCATIONS 0000 TO 4177 OF THE CURRENT FIELD.

3. PRELIMINARY PROGRAMS

THE FOLLOWING PROGRAMS SHOULD BE RUN IF THE FORMATTER PROGRAM FAILS TO OPERATE CORRECTLY:

ALL BASIC AND EXTENDED MEMORY DIAGNOSTICS

FOR THE RK8E CONTROL, RUN THE RK8E DISKLESS CONTROL TEST AND THE RK8E DRIVE CONTROL TEST.

FOR THE RK8L CONTROL, RUN THE RK8L INSTRUCTION TEST.

4. OPERATOR AND/OR PROGRAM ACTION

4.1 STANDARD TEST PROCEDURE

-
- A. LOAD THE PROGRAM INTO ANY R/W MEMORY BANK USING THE STANDARD BINARY LOADER TECHNIQUE.
 - B. TO RUN THE FORMATTEN PROGRAM, FOLLOW THE PROCEDURE IN SECTION 4.4.

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 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. OPEN ACCESS DOOR.
- G. INSERT CARTRIDGE.
- H. CLOSE ACCESS DOOR.
- I. SET SWITCH LABELED "RUN/LOAD" TO THE "RUN" POSITION.
- J. WAIT FOR 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 LIGHT LABELED "WT PROT" GOES OFF.
- M. VERIFY THAT LIGHTS LABELED "FAULT", "WT", "RD", AND "LOAD" ARE OFF.

4.3 RK05F DRIVE SETUP PROCEDURE

THE FOLLOWING IS THE CORRECT DRIVE 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 FORMAT PROGRAM

- A. MAKE READY ALL DRIVES TO BE FORMATTED:
 - FOR RK05J DRIVES USE THE RK05 DRIVE MOUNTING PROCEDURE REFER TO SECTION 4.2.
 - FOR RK05F DRIVES USE THE RK05 DRIVE SETUP PROCEDURE REFER TO SECTION 4.3.
- B. SET SWITCH LABELED "RUN/LOAD" TO THE "LOAD" POSITION ON ALL DRIVES NOT BEING FORMATTED.
- C. SET THE SWITCH REGISTER TO 0200 AND PRESS LOAD ADDRESS.
- D. SET THE SWITCH REGISTER TO 0000.
- E. IF IT IS DESIRED TO CHANGE THE IOT DEVICE CODES WITHIN THE PROGRAM (THEY ARE NORMALLY X74X), SET SWITCH REGISTER BIT 11 TO A "1".
- F. IF CHANGE IOT CODES WAS SELECTED, SET SWITCH REGISTER BITS 3 TO 8 TO THE DESIRED IOT DEVICE CODE.
- G. PRESS KEY START (KEY START IS KEY CLEAR AND THEN KEY CONTINUE ON A PDP8/E, PDP8/F, OR PDP8/M). IF SELECTING A PDP8/A (PRESS INIT AND THEN PRESS RUN). IF SELECTED, ALL IOT DEVICE CODES WITHIN THE PROGRAM WILL BE CHANGED. THE TTY WILL TYPE THE FOLLOWING PROGRAM NAME, INFORMATION, AND QUESTION.

RK8E/RK8L DISK FORMATTER PROGRAM

FOR ALL QUESTIONS ANSWER Y FOR YES OR N FOR NO,
FORMAT DISK 0?

- H. IF THE OPERATOR DESIRES TO FORMAT DISK 0, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 1?

- I. IF THE OPERATOR DESIRES TO FORMAT DISK 1, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 2?

- J. IF THE OPERATOR DESIRES TO FORMAT DISK 2, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 3?

- K. IF THE OPERATOR DESIRES TO FORMAT DISK 3, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 4?

- L. IF THE OPERATOR DESIRES TO FORMAT DISK 4, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 5?

- M. IF THE OPERATOR DESIRES TO FORMAT DISK 5, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 6?

- N. IF THE OPERATOR DESIRES TO FORMAT DISK 6, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

FORMAT DISK 7?

- O. IF THE OPERATOR DESIRES TO FORMAT DISK 7, TYPE Y FOR YES,
OTHERWISE, N FOR NO, ON THE TTY KEYBOARD. THE FOLLOWING
QUESTION WILL THEN BE TYPED ON THE TTY.

ARE YOU SURE?

- P. TYPING N FOR NO WILL RESULT IN REPEATING ALL THE PREVIOUS
QUESTIONS. TYPING Y FOR YES, WILL RESULT IN EXECUTION
OF THE OPERATION SELECTED.

- Q. PROGRAM EXECUTION IS APROX. 90 SECONDS PER DISK DRIVE.
AFTER ALL DISKS SELECTED HAVE BEEN FORMATTED AND CHECKED
THE TTY WILL TYPE THE FOLLOWING PASS COMPLETE MESSAGE AND

QUESTION.

RK8E/RK8L DISK FORMATTER PASS COMPLETE
FORMAT SAME DISK(S) AGAIN?

R. IF THE OPERATOR DESIRES TO REPEAT THE OPERATION SELECTED,
TYPE Y FOR YES. TYPING N FOR NO WILL RESULT IN A REPEAT
OF THE INITIAL START-UP QUESTIONS.

4,5 SWITCH REGISTER SETTINGS

SWR11=0 DO NOT CHANGE IOT DEVICE CODES

SWR11=1 CHANGE IOT DEVICE CODES

SWR3=8 DESIRED IOT DEVICE CODE.

5. ERRORS

WHEN A RECOVERABLE ERROR OCCURS THE TTY WILL PRINT
AN "ERROR HEADER" AND ERROR INFORMATION PERTAINING
TO THE FAILURE.

POSSIBLE ERROR HEADERS ARE AS FOLLOWS.

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

AFTER THE ERROR HEADER MENTIONED ABOVE IS TYPED THE TTY
WILL PRINT SOME OF THE FOLLOWING ERROR INFORMATION PER-
TAINING TO THE FAILURE.

PC: PROGRAM LOCATION OF FAILURE

GD: EXPECTED INFORMATION

EX: EXTENDED DRIVE BIT

CM: SOFTWARE COMMAND REGISTER

ST: CONTENTS OF STATUS REGISTER

DA: SOFTWARE CYLINDER, SURFACE, AND SECTOR REGISTER

CA: INITIAL CURRENT ADDRESS

AD: ADDRESS OF DATA BREAK

DT: DATA FOUND DURING DATA BREAK

AFTER THE ERROR INFORMATION IS TYPED THE TTY WILL TYPE ONE
OF THE FOLLOWING QUESTIONS ASKING THE ERROR RECOVERY DESIRED.

A. IF THE ERROR WAS A RECALIBRATE ERROR THE FOLLOWING QUESTION

WILL BE TYPED.

TRY TO RECALIBRATE SAME DISK AGAIN?

TYPING A Y FOR YES WILL RESULT IN A REPEAT OF THE RE-CALIBRATE SEQUENCE ON THE DISK IN ERROR, TYPING N FOR NO WILL RESULT IN PROGRESSING TO THE NEXT AVAILABLE DISK.

B. IF THE ERROR WAS A WRITE ERROR THE FOLLOWING QUESTION WILL BE TYPED.

TRY TO FORMAT SAME CYLINDER AGAIN?

TYPING Y FOR YES WILL RESULT IN A REPEAT OF THE WRITE SEQUENCE ON THE CURRENT CYLINDER, TYPING N FOR NO WILL RESULT IN PROGRESSING TO THE NEXT SEQUENTIAL CYLINDER.

C. IF THE ERROR WAS A READ OR CHECK ERROR THE FOLLOWING QUESTION WILL BE TYPED.

TRY TO CHECK SAME CYLINDER AGAIN?

TYPING A Y FOR YES WILL RESULT IN A REPEAT IN THE READ AND CHECK SEQUENCE ON THE CURRENT CYLINDER, TYPING A N FOR NO WILL RESULT IN PROGRESSING TO THE NEXT SEQUENTIAL CYLINDER.

6. PROGRAM DISCRIPTION

THE FORMATTING IS ACTUALLY A FUNCTION OF THE RK8E OR RK8L CONTROL AND DRIVE LOGIC. THE PROGRAM SIMPLY WRITES DATA ON EVERY SECTOR IN THE "WRITE ALL" MODE, THEN CHECKS THE DATA IN SUCH A WAY IN THE "READ DATA" MODE AS TO VERIFY THAT THE HEADER WORDS WRITTEN ON EVERY SECTOR ARE ALSO CORRECT. THE "READ DATA MODE" AUTOMATICALLY PERFORMS A CHECK HEADER FUNCTION.

THE FIRST TWO WORDS OF EVERY SECTOR ARE SET TO THE ABSOLUTE DISK ADDRESS (I.E. COMMAND REGISTER BITS 9-11 AND CYLINDER, SURFACE, AND SECTOR BITS 0-11, RESPECTIVELY) AND THE REMAINDER OF THE DATA AREA TO ALL ZEROS WHEN THE DATA IS WRITTEN. ONLY THE FIRST TWO WORDS OF EVERY SECTOR (I.E. THE ADDRESSING INFORMATION) ARE CHECKED WHEN DATA IS READ IN THE "READ DATA" MODE.

7. APT-8 HOOKS

7.1 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.

7.2 SETUP

ONLY HARDWARE CONFIGURATION WORD 2, ADDRESS 22, NEED BE ESTABLISHED. THE FOLLOWING INFORMATION MUST BE INDICATED:

1. SINGLE OR MULTIPLE DRIVE TESTING.
2. DRIVE OR DRIVES TO BE TESTED.
3. DIAGNOSTIC RUNNING UNDER APT-8.

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 5 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 INDEPENDENTLY.

THE PROGRAM ALLOWS ONLY DRIVES ZERO (0) THROUGH SEVEN (7) TO BE TESTED AT THIS TIME.

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.

7.3 APT-8 INTERFACES

7.3.1. 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 THE SLOWER BUS MEMORY.

7.3.2. ERRORS

ONLY THE ERROR PC IS REPORTED TO APT-8 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.

8. PROGRAM LISTING

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 DESCRIBED IN SECTIONS 1 THROUGH 8 OF THIS DOCUMENT.

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 INITIALIZE FOR A ACTIVE CONSOLE PACKAGE.

9.3 INITIALIZATION

FOR A ACTIVE CONSOLE PACKAGE

- 1.) SET LOCATION 21 BIT0=0 TO INDICATE USE OF 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 TO 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 LOUP FOR A CONTINUE. THE ONLY WAY TO CONTINUE WILL BE TO TYPE A CONTROL G, R OR C . THIS IS A NONPRINTING CHARACTER.

CONTROL G

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 PROGRAM PASS COMPLETE AS DESCRIBED IN SECTION 4.4 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 4.5 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 LOCATIONS CAN BE CHANGED TO MEET THE SPECIFIC
NEED FOR MODIFICATION OF THE DIAGNOSTIC.

3637 IS THE LOCATION SET FOR THE NUMBER OF
FILLER CHARACTERS AFTER A CRLF SET TO FOUR (4)

/RK8E/RK8L DISK FORMATTER PROGRAM: MD=08-DHRKD=0
/MAINDEC=08-DHMKD=0=0

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	DLOC=6746		/LOAD COMMAND REGISTER
6747	DMAN=6747		/LOAD MAINTENANCE
/			
4446	LDSC=JMS I	XXLASC	
4430	IOTCHN=JMS I	XCMANG	
4431	LOUTRK=JMS I	XWRTRK	
4432	REDOSK=JMS I	XRDTRK	
4433	RECAL=JMS I	XRESTR	
4434	RECEIV=JMS I	XWAIT	
4435	KILBUF=JMS I	XKLBUF	
4437	ERROR=JMS I	XERRO	
4440	RSTAT=JMS I	XRST	
4444	LDADD=JMS I	XLDAO	
4441	DSKSKP=JMS I	XSDKP	
4442	LDCMD=JMS I	XLDCM	
4443	LDCUR=JMS I	XLUCA	
4445	CLRALL=JMS I	XCLDR	
4447	PRNTER=JMS I	XPHN	
4450	OCTEL=JMS I	XFROCT	
4451	TWOCT=JMS I	XTOCT	
4436	TYPE=JMS I	XPRINT	
4452	CRLF=JMS I	XCRLF	
4424	APT8=JMS I	XAPT8	
4425	TIME=JMS I	XTIME	
4427	TICK=JMS I	XTICK	
4426	KAERRO=JMS I	XAERRO	
/			
0000	*0		
/			
0000	0304	304	/REV D
0001	5001	5001	
0002	0002	0002	
0003	0003	0003	
/			
0010	*10		
/			
0010	0000	AUTO10, 0	
/			
0011	0000	AUTO11, 0	
/			
0020	*20		
/			
0020	0000	0000	/PSEUDO SWITCH REGISTER
0021	4000	4000	/CONTRUL WORD 1
0022	0000	0000	/CONTRUL WORD 2

0023	0000	0000	/RESERVED
0024	1125	XAPT8, APT8	
0025	1557	XTIME, KTIME	
0026	1600	XAERRO, AERRO	
0027	1530	XTICK, KTICK	
0030	1463	XCMANG, CMANG	
0031	0600	XWRTRK, WRTRK	
0032	1000	XRDTRK, REDTRK	
0033	1400	XRESTR, RESTOR	
0034	1327	XWAIT, WAIT	
0035	0752	XKLBUF, KLBUF	
0036	1312	XPRINT, PRINT	
0037	0436	XERRO, ERRO	
0040	0671	XRST, RST	
0041	0740	XSDKP, SKP	
0042	0720	XLDCM, LDCM	
0043	0700	XLUCA, LUCA	
0044	0711	XLOAD, LOAD	
0045	0745	XCLDR, CLDR	
0046	0733	XLASC, LASC	
0047	1252	XPHN, PHN	
0050	1227	XFROCT, FROCT	
0051	1200	XTOCT, TOCT	
0052	1215	XCRLF, UPONE	
0053	2201	XLOTRK, LOTRK	
0054	2200	XHTRK, HTRK	
0055	2200	BGNBUF, WRKBUF	
0056	0000	AMOUNT, 0	
0057	0000	SWITCH, 0	
0060	0003	K0003, 0003	
0061	0004	K4, 4	
0062	0007	K0007, 0007	
0063	0040	K0040, 0040	
0064	7465	M313, -313	
0065	0277	K0277, 0277	
0066	0200	K0200, 0200	
0067	0260	K0260, 0260	
0070	4000	K4000, 4000	
0071	7735	K7735, 7735	
0072	7760	K7760, 7760	
0073	0400	K0400, 400	
0074	0037	K0037, 0037	
0075	6201	KCDF, CDF	
0076	7774	M4, -4	
0077	7770	M10, -10	
0100	0000	DRIVNO, 0	
0101	0000	CHAR, 0	
0102	0000	LOWAD, 0	
0103	0000	HIGHAD, 0	
0104	0000	TRKCNT, 0	
0105	0000	DSKCNT, 0	
0106	0000	SBCNT1, 0	
0107	0000	STCNT1, 0	
0110	0000	STCNT2, 0	
0111	0000	STCNT3, 0	

```

0112 0000 TCNTR1, 0
0113 0000 TCNTR2, 0
0114 0000 TCNTR3, 0
0115 0000 TCNTR4, 0
0116 0000 TCNTR5, 0
/
0117 0000 DOREG2, 0
0120 0000 EX0IT, 0
0121 0000 CMREG, 0
0122 0000 STREG, 0
0123 0000 DAREG, 0
0124 0000 CAREG, 0
0125 0000 ADREG, 0
0126 0000 DTREG, 0
0127 0263 BGNTRT, FRMDSK
0130 0000 HOMEHA, 0
0131 0000 DATCNT, 0
0132 7776 CLKCNT, =2
/
0133 1623 XMOVE, MOVE
0134 0000 LOC0ED, 0
0135 0424 XEND, ENDTST
0136 0000 SOFT, 0
0137 0140 ADPOT1, DSK0A
0140 0000 DSK0A, 0
0141 0000 DSK1A, 0
0142 0000 DSK2A, 0
0143 0000 DSK3A, 0
0144 0000 DSK4A, 0
0145 0000 DSK5A, 0
0146 0000 DSK6A, 0
0147 0000 DSK7A, 0
0150 0151 ADPOT2, DSK0B
0151 0000 DSK0B, 0
0152 0000 DSK1B, 0
0153 0000 DSK2B, 0
0154 0000 DSK3B, 0
0155 0000 DSK4B, 0
0156 0000 DSK5B, 0
0157 0000 DSK6B, 0
0160 0000 DSK7B, 0
0161 0000 PCOUNT, 0
/
0200 0200 /USED ONLY IF ON APT
/
0200 6224 BGN, RIF
0201 3130 DCA HOMEHA
0202 1130 TAD HOMEHA
0203 1075 TAD K0DF /MAKE HOMEDF
0204 3205 DCA ,+1
0205 7402 HLT /MAKE DF=IF
/
/NOH TEST FOR APT SYSTEM
/IF ON APT TERMINAL MESSAGES ARE SKIP
/TO AVOID TIMING PROBLEMS WITH THE SYSTEM
0206 4424 APT8A /TEST FOR APT SYSTEM

```

```

0207 4777 JMS XC8PSW /GET SR=
0210 4430 IDTCHN /CHANGE DEVICE TO SWR3=0
0211 4452 CRLF
0212 4452 CRLF
0213 4447 PRNTR /PRINT "RK8E/RK8L DISK FORMATTER PROGRAM"
0214 2045 MES1 /MESSAGE 1 POINTER
0215 4452 CRLF
0216 4447 PRNTR /PRINT "FOR ALL QUESTIONS"
0217 2066 MES2 /MESSAGE POINTER 2
0220 1077 ALLAGN, TAD M10
0221 3107 DCA STCNT1 /COUNTER FOR AMOUNT OF DISKS
0222 3134 DCA LOC0ED
0223 3110 DCA STCNT2
0224 4452 SAMAGN, CRLF
0225 4447 PRNTR /PRINT "FORMAT DISK ? "
0226 2117 MES3 /MESSAGE POINTER 3
0227 1110 TAD STCNT2
0230 1067 TAD K0260
0231 4436 TYPE /TYPE DISK NUMBER
0232 1065 QUES1, TAD K0277
0233 4436 TYPE /TYPE ?
0234 1137 TAD ADPOT1
0235 1110 TAD STCNT2
0236 3111 DCA STCNT3
0237 4434 RECEIV /WAIT FOR CHARACTER
0240 5244 JMP NOTDSK /NO NOT THIS DISK
0241 5232 JMP QUES1 /NEITHER YES OR NO
0242 2134 WABDSK, ISZ LOC0ED
0243 7340 CLA CLL CHA
0244 3511 NOTDSK, DCA I STCNT3 /YES, WAS CLEAR DISK POINTER
0245 2110 ISZ STCNT2 /UPDATE POINTER
0246 2107 ISZ STCNT1 /COUNT DISKS
0247 5224 JMP SAMAGN /ASK ABOUT NEXT
/
0250 4452 DONE, CRLF
0251 4447 PRNTR /PRINT "ARE YOU SURE ?"
0252 2126 MES4 /MESSAGE POINTER 4
0253 4434 RECEIV /WAIT FOR CHARACTER
0254 5220 JMP ALLAGN /NO, START ALL OVER
0255 5250 JMP DONE /NEITHER TYPE ?
0256 1134 TAD LOC0ED
0257 7041 CIA
0260 7450 BNA /ANY DISKS
0261 5200 JMP BGN /NO, OPERATOR ERROR
0262 3134 DCA LOC0ED /YES, AMOUNT LOCATED
/
/FIRST RECALIBRATE AND FORMAT IN WRITE ALL MODE
/ALL DISK DRIVES SELECTED BY OPERATOR,, MAKE THE FIRST
/TWO WORDS OF EVERY DISK SECTOR EQUAL TO THE
/ABSOLUTE DISK ADDRESS.
/
0263 4533 FRMDSK, JMS I XMOVE /MOVE DISK POINTERS
0264 1134 TAD LOC0ED
0265 3056 DCA AMOUNT
0266 1056 TAD AMOUNT

```



```

0430 4447          PRNTER          /PRINT "TRY SAME SEQUENCE"
0431 2135          MESS
0432 4434          RECEIV          /WAIT FOR INPUT FROM OPERATOR
0433 5775'        JMP          ALLAGN          /NO, ASK AGAIN
0434 5227'        JMP          *-3
0435 5776'        JMP          FRMDSK          /TRY SAME SEQUENCE

/
/
/SUBROUTINE FOR "ERRORS," SCOPE LOOPS, AND
/ERROR TYPEOUTS,
/
0436 0000          ERNO, 0
0437 7301          CLA CLL IAC
0440 1236          TAD          ERNO          /GET PC STORED
0441 3344          DCA          RETRNI          /STORE FOR RETURN
0442 4426          KAERRO          /NOTIFY APT OF ERROR IS NEEDED
0443 4452          CRLF
0444 4452          CRLF
0445 1636          TAD I          ERNO          /GET TEXT POINTER
0446 0062          AND          K0007          /MASK 9=11
0447 1352          TAD          HEDTAD          /MAKE ERROR HEADER TAD
0450 3251          DCA          *-1
0451 7402          HLT
0452 3254          DCA          *-2          /MODIFIED HEADER TAD
0453 4447          PRNTER          /MODIFIED HEADER POINTER
0454 7402          HLT
0455 4452          CRLF
0456 4447          PRNTER          /PRINT PC:
0457 1642          TEXP
0460 1236          TAD          ERNO          /GET PC POINTER
0461 4450          OCTEL          /PRINT PC STORED
0462 1636          TAD I          ERNO          /GET TEXT POINTER
0463 7104          CLL RAL
0464 7420          SNL
0465 5274          JMP          NTGD          /NOT GD: REGISTER
0466 3236          DCA          ERNO
0467 4447          PRNTER          /PRINT GD:
0470 1644          TEXGD
0471 1117          TAD          GONEG2
0472 4450          OCTEL          /PRINT FOUR OCTAL
0473 7610          SKP CLA
0474 3236          DCA          ERNO
0475 4447          PRNTER
0476 1646          TEXEX
0477 1120          TAD          EXBIT
0500 7640          SZA CLA
0501 7001          IAC
0502 4450          OCTEL
0503 1345          TAD          XTEXT
0504 3350          DCA          PCNTR2
0505 1346          TAD          XREG
0506 3010          UCA          AUTO10
0507 1357          TAD          K7771
0510 3347          DCA          PCNTR1          /COUNTER FOR # OF HEADS
0511 7344          CLA CLL CMA RAL

```

```

0512 3351          OCA          PCNTR3
0513 1236          STRAUT, TAD          ERNO          /GET TEXT POINTER
0514 7500          SMA
0515 5336          JMP          NOTEX          /NOT THIS ONE
0516 7104          CLL RAL
0517 3236          OCA          ERNO
0520 1350          TAD          PCNTR2          /GET TEXT MESSAGE POINTER
0521 2350          ISZ          PCNTR2
0522 2350          ISZ          PCNTR2
0523 3325          DCA          *-2          /STORE FOR PRNTER
0524 4447          PRNTER          /PRINT XX:
0525 7402          HLT          /MODIFIED TEXT POINTER
0526 1410          TAD I          AUTO10
0527 4450          OCTEL          /PRINT FOUR OCTAL
0530 2351          ISZ          PCNTR3
0531 7610          SKP CLA
0532 4452          CRLF
0533 2347          AGAIN, ISZ          PCNTR1
0534 5313          JMP          STRAUT
0535 5744          JMP I          RETRNI          /CHECK FOR NEXT XX:
0536 7104          NOTEX, CLL RAL          /RETURN TO QUESTION
0537 3236          DCA          ERNO
0540 2350          ISZ          PCNTR2
0541 2350          ISZ          PCNTR2
0542 2010          ISZ          AUTO10
0543 5333          JMP          AGAIN

/
RETRN., 0
0545 1650          XTEXT, TEXCH
0546 0120          XREG, EXBIT
0547 0000          PCNTR1, 0
0550 0000          PCNTR2, 0
0551 0000          PCNTR3, 0
0552 1353          HEDTAD, TAD          HEDLST
0553 1664          HEDLST, ERTX1
0554 1675          ERTX2
0555 1705          ERTX3
0556 1717          ERTX4
0557 7771          K7771, 7771

/
0575 0220          PAGE
0576 0263
0577 0347          /
0600 0000          /ROUTINE TO FORMAT CYLINDER
0601 7330          /MAKE FIRST TWO WORDS OF EVERY SECTOR
0602 3117          /EQUAL TO DISK ADDRESS.
0603 4435          /
0604 1071          WRTTRK, 0
0605 3112          CLA CLL CML RAR
0606 3117          DCA          GDREG2          /SETUP COMPARE REGISTER
0607 4435          KILBUF          /CLEAR BUFFER
0608 1071          TAD          K7735          /AMOUNT OF SECTORS TO DO
0609 3112          DCA          TCNTR1          /SETUP COUNTER

```

```

0606 3113      DCA      TCNTR2      /STARTING WITH 0
0607 1072      TAD      K7760      /STOPPER
0610 3114      DCA      TCNTR3      /SECTOR COUNTER POINTER STOP
0611 1113      LODR1,  TAD      TCNTR2
0612 0074      AND      K0037      /MASK SECTOR BITS
0613 1102      TAD      LOWAD      /ADD IN CYLINDER
0614 3453      OCA I    XLOTRK      /SETUP TRACK WORD IN BUFFER
0615 1120      TAD      EXBIT      /ADD IN EXTENDED BIT
0616 1103      TAD      HIGHAD      /ADD IN DRIVE NUMBER
0617 1100      TAD      DRIVNO      /SETUP TRACK WORD IN BUFFER
0620 3454      OCA I    XMITRK
0621 1454      TAD I    XMITRK
0622 0270      AND      K7577
0623 1130      TAD      HOMEMA      /CURRENT FIELD
0624 1267      TAD      K5000      /FUNCTION WRITE ALL
0625 4442      LOCMD      /LOAD COMMAND
0626 1120      TAD      EXBIT
0627 4446      LD8C      /LOAD EXTENDED DRIVE BIT
0630 7200      CLA      /CLEAR EXTENDED DRIVE BIT
0631 1055      TAD      BGNBUF
0632 4443      LOCUR      /LOAD CURRENT ADDRESS
0633 1453      TAD I    XLOTRK
0634 4444      LDA00      /LOAD TRACK AND GO
0635 4441      DSKSKP      /SKIP ON FLAG
0636 5235      JMP      =-1      /WAIT FOR FLAG
0637 4440      RDSTAT      /READ STATUS
0640 1070      TAD      K4000
0641 7640      SZA CLA      /WAS STATUS 0?
0642 5254      JMP      LOUER      /ERROR, STATUS ON WRITE ALL
0643 2113      ISZ      TCNTR2
0644 2114      ISZ      TCNTR3      /COUNT FIRST REVOLUTION
0645 7610      SKP CLA      /STILL IN FIRST REV.
0646 3113      DCA      TCNTR2      /SETUP FOR SECTOR "1"
0647 2113      ISZ      TCNTR2
0650 2112      ISZ      TCNTR1      /UPDATE SECTOR COUNTER
0651 5211      JMP      LODR1      /TRY NEXT SECTOR
0652 2200      ISZ      WRITRK
0653 5600      JMP I    WRITRK      /THIS CYLINDER DONE
0654 4437      LOUER,  ERROR      /ENRDR, STATUS
0655 3602      3602      /TEXT POINTER

/
0656 4433      RECAL      /CLEAR CONTROL AND DRIVE
0657 5600      JMP I    WRITRK      /TO NEXT DISK
0660 4452      CRLF
0661 4407      PHNTEH      /PRINT "TRY SAME AGAIN"
0662 1734      ERMES1
0663 4434      RECEIV
0664 5252      JMP      LOUER=2      /WAIT FOR YES OR NO
0665 5200      JMP      =0      /WAS A NO TRY SAME CYLINDER
0666 5201      JMP      WRITRK +1      /WAS NEITHER ASK AGAIN
0667 5000      K5000, 5000      /YES, TRY NEXT
0670 7577      K7577, 7577
/
/
/SUBROUTINE TO READ STATUS REGISTER

```

```

/
0671 0000      RDST,  0
0672 6745      IO15,  DRST      /HEAD STATUS IOT
0673 7410      SKP
0674 4777*     ERHLT5, JMS      XCERR      /SKIP TRAP ERROR.
0675 3122      DCA      STMEG      /SAVE RESULTS
0676 1122      TAD      STMEG
0677 5671      JMP I    RDST      /EXIT

/
/SUBROUTINE TO LOAD CURRENT ADDRESS REGISTER
/
0700 0000      LDCA,  0
0701 3125      DCA      ADREG      /SAVE IN ADDRESS
0702 1125      TAD      ADREG
0703 3124      DCA      CAREG      /SETUP INITIAL CURRENT ADDRESS
0704 1125      TAD      ADREG
0705 6744      IO14,  ULCA      /LOAD CURRENT ADDRESS IOT
0706 5700      JMP I    LOCA      /EXIT
0707 4777*     ERHLT4, JMS      XCERR      /SKIP TRAP ERROR.
0710 5307      JMP      =-1

/
/
/SUBROUTINE TO LOAD TRACK ADDRESS REGISTER
/
0711 0000      LDAD,  0
0712 3123      DCA      DAREG      /SAVE OUTBOUND DATA
0713 1123      TAD      DAREG
0714 6743      IO13,  DLAG      /LOAD DISK ADDRESS REGISTER
0715 5711      JMP I    LOAD      /EXIT
0716 4777*     ERHLT3, JMS      XCERR      /SKIP TRAP ERROR.
0717 5316      JMP      =-1

/
/
/SUBROUTINE TO LOAD COMMAND REGISTER
/
0720 0000      LOCM,  0
0721 3121      DCA      CMREG      /SAVE OUTBOUND DATA
0722 3776*     OCA      INMODE
0723 4775*     JMS      XCCKP      /CHECK FOR CONTROL CHARACTERS.
0724 7200      CLA
0725 7200      CLA
0726 1121      TAD      CMREG
0727 6746      IO16,  DL8C      /LOAD COMMAND REGISTER
0730 5720      JMP I    LOCM      /EXIT
0731 4777*     ERHLT6, JMS      XCERR      /SKIP TRAP ERROR.
0732 5331      JMP      =-1

/
/
/SUBROUTINE ISSUE "DLSC"
0733 0000      XL8C,  0
0734 6740      IO10,  DL8C
0735 5733      JMP I    XL8C
0736 4777*     ERHLT8, JMS      XCERR
0737 5336      JMP      =-1

```

```

/SUBROUTINE TO ISSUE "DSKP" DISK SKIP IOT
/
0740 0000   SDKP, 0
0741 0741   IOT1, DSKP           /DISK SKIP IOT
0742 7410   SKP             /DID NOT SKIP
0743 2340   ISZ           SDKP
0744 5740   JMP I           SDKP           /EXIT
/
/SUBROUTINE TO ISSUE "DCLR" CLEAR IOT
/
0745 0000   CLDR, 0
0746 0742   IOT2, DCLR           /DCLR "CLEAR IOT"
0747 5745   JMP I           CLDR           /EXIT
0750 4777*  ERHLT2, JMS          XCERR      /SKIP TRAP ERROR.
0751 5350   JMP             .-1
/
/ROUTINE TO ZERO WORK BUFFER
/
0752 0000   KLBUF, 0
0753 7340   CLA CLL CMA
0754 1055   TAD           BGNBUF           /START OF BUFFER =1
0755 3010   DCA           AUTO10          /SETUP AUTO INDEX
0756 1364   TAD           K7400
0757 3131   DCA           DATCNT          /SETUP COUNTER
0760 3410   DCA I          AUTO10          /CLEAR BUFFER
0761 2131   ISZ           DATCNT          /UPDATE COUNTER
0762 5360   JMP             .-2           /NOT ALL CLEARED YET
0763 5752   JMP I           KLBUF          /BUFFER CLEARED
0764 7400   K7400, 7400
/
0775 3641
0776 3676
0777 4007
1000
PAGE
/
/ROUTINE TO READ AND CHECK A CYLINDER
/

```

```

1000 0000   REDTRK, 0
1001 1071   TAD           K7735
1002 3112   DCA           TCNTR1          /AMOUNT OF SECTORS TO DO
1003 3113   DCA           TCNTR2          /STARTING WITH 0
1004 1072   TAD           K7760
1005 3114   DCA           TCNTR3
1006 4435   KILBUF           /CLEAR BUFFER
1007 7340   CHKRI, CLA CLL CMA
1010 3136   DCA           SOFT           /SETUP SOFT ERROR FLAG
1011 1055   TAD           BGNBUF
1012 4443   LOCUR           /LOAD CURRENT ADDRESS
1013 1103   TAD           HIGHAD          /EXTENDED CYLINDER BIT
1014 1100   TAD           URIVNO          /CURRENT DRIVE
1015 1130   TAD           HOMEMA          /CURRENT FIELD
1016 4442   LOCMD           /LOAD COMMAND
1017 1120   TAD           EXBIT           /LOAD EXTENDED DRIVE BIT
1020 4446   LOSC

```

```

1021 7200   CLA
1022 1113   TAD           TCNTR2          /CLEAR EXTENDED DRIVE BIT
1023 0074   AND           K0037
1024 1102   TAD           LOWAD           /MASK SECTOR BITS OFF
1025 4444   LDADD           /ADD IN OTHER DISK ADDRESS
1026 4441   DSXSKP          /LOAD AND GO
1027 5226   JMP             .-1           /DISK SKIP IOT
1030 4440   RDSTAT          /WAIT FOR FLAG
1031 1070   TAD           K4000           /READ STATUS
1032 7650   SNA CLA          /ADD IN FUDGE FACTOR
1033 5241   JMP           STOK           /SKIP IF ERROR
1034 1122   TAD           STREG          /STATUS O.K.
1035 0777*  AND           K0010          /GET STATUS READ
1036 7650   SNA CLA          /WAS IT A CRC
1037 5306   JMP           STAER          /NO, JUST A HARD ERROR
1040 3136   DCA           SOFT           /CLEAR SOFT ERROR FLAG
1041 1121   STAOK, TAD        CMREG          /GET LAST COMMAND
1042 0062   AND           K0007
1043 1120   TAD           EXBIT           /ADD EXTENDED DRIVE BIT
1044 7041   CIA
1045 1434   TAD I          XHITRK          /GET WORD READ FROM DISK
1046 7650   SNA CLA          /SKIP IF ERROR
1047 5256   JMP           FRSTOK          /FIRST WORD O.K.
1050 1454   TAD I          XHITRK          /GET WORD
1051 3126   DCA           DTREG          /SETUP ERROR PRINTER
1052 1121   TAD           CMREG
1053 0062   AND           K0007
1054 3117   DCA           GDREG2          /SETUP GOOD FOR PRINTER
1055 5303   JMP           DATER          /NO, DATA ERROR
1056 1453   FRSTOK, TAD I     XLUTRK          /GET WORD READ
1057 7041   CIA
1060 1123   TAD           DANEG          /COMPARE TO GOOD
1061 7650   SNA CLA          /SKIP IF ERROR
1062 5271   JMP           DATOK          /WORD O.K.
1063 2125   ISZ           ADREG          /SETUP ERROR PRINTER
1064 1123   TAD           UAREG
1065 3117   DCA           GDREG2          /SETUP GOOD WORD FOR PRINTER
1066 1453   TAD I          XLUTRK          /GET WORD READ
1067 3126   DCA           DTREG          /SETUP FOR PRINTER
1070 5303   JMP           DATER          /DATA ERROR
1071 1136   DATOK, TAD        SOFT          /GET SOFT ERROR FLAG
1072 7650   SNA CLA          /WAS IT CLEAR
1073 5306   JMP           STAER          /YES, STATUS ERROR
1074 1113   TAD           TCNTR2
1075 1060   TAD           K0003           /ADVANCE 3 SECTORS
1076 3113   DCA           TCNTR2
1077 2114   ISZ           TCNTR3
1100 5207   JMP           CMKR1           /MORE TO FORMAT
1101 2200   ISZ           REDTRK
1102 5000   JMP I          REDTRK          /EXIT, O.K.
1103 1776*  DATER, TAD        K7741
1104 3313   DCA           TCHKT          /SETUP TEXT POINTER
1105 5312   JMP           CMKER          /ERROR
1106 1775*  STAER, TAD        K3000
1107 3313   DCA           TCHKT          /SETUP TEXT POINTER

```

```

1110 7330          CLA CLL CML RAR
1111 3117          DCA GONEG2
1112 4437          CHKR, ERROR /SETUP GOOD STATUS PRINTER
1113 0000          TCHK, 0 /ERROR, READ DATA
1114 4433          RECAL /MODIFIED TEXT POINTER
1115 5000          JMP I REDTRK /CLEAR CONTROL AND DRIVE
1116 4452          CRLF /TO NEXT DISK
1117 4447          PRNTR /PRINT "TRY SAME AGAIN"
1120 2000          ERME33
1121 4434          RECEIV
1122 5301          JMP DATER -2 /CHECK NEXT
1123 5316          JMP ,=5 /RE-PRINT
1124 5201          JMP MEDTRK +1 /TRY SAME AGAIN

/THIS ROUTINE WILL TEST FOR APT AND NOP CONSOLE
/PACKAGE IF NEED BE
/
1125 0000          APT8, 0
1126 1022          TAD 22
1127 7700          SNA CLA
1130 5725          JMP I APT8
1131 1022          TAD 22
1132 0373          AND K7377 /ON APT, NOP CONSOLE PACKAGE
1133 3022          DCA 22
1134 1022          TAD 22
1135 0062          AND K0007 /ISOLATE DRIVE NUMBER OR
/NUMBER OF DRIVES TO BE DONE

1136 3107          DCA STCNT1
1137 1022          TAD 22
1140 0774          AND K0100
1141 7650          SNA CLA /SINGLE DRIVE TESTING
1142 5353          JMP MULDSK /NO, SEVERAL TO DO
1143 1137          TAD ADPOT1 /GET DISK POINTER
1144 1107          TAD STCNT1 /ESTABLISH DRIVE TO DO
1145 3107          DCA STCNT1
1146 7340          CLL CLA CMA /=-1
1147 3507          DCA I STCNT1
1150 7340          CLL CLA CMA /ONE DISK TO DO
1151 3134          DCA LOCBED
1152 5527          JMP I 0GNTST
1153 1107          MULDSK, TAD STCNT1 /UNIVE TO BE DONE
1154 7040          CMA
1155 3107          DCA STCNT1
1156 1137          TAD ADPOT1 /GET DISK POINTER
1157 1110          TAD STCNT2 /ESTABLISH DRIVE TO BE DONE
1160 3111          DCA STCNT3
1161 2134          ISZ LOCBED
1162 7340          CLL CLA CMA
1163 3511          DCA I STCNT3 /DO THIS DRIVE
1164 2110          ISZ STCNT2
1165 2107          ISZ STCNT1
1166 5356          JMP MULDSK+3 /MORE TO DO
1167 1134          TAD LOCBED
1170 7041          CIA
1171 3134          DCA LOCBED /NUMBER TO BE DONE

```

```

1172 5527          JMP I 0GNTST
1173 7377          K7377, 7377
1174 1556
1175 1326
1176 1325
1177 1324
1200 1200          PAGE
/
/SUBROUTINE TO PRINT TWO OCTAL
/
1200 0000          TOCT, 0
1201 3106          DCA 0BCNT1 /SAVE AC
1202 1106          TAD 0BCNT1
1203 7010          HAR
1204 7012          NTR
1205 0062          AND K0007
1206 1067          TAD K0260
1207 4436          TYPE /PRINT FIRST BYTE
1210 1106          TAD 0BCNT1
1211 0062          AND K0007
1212 1067          TAD K0260
1213 4436          TYPE /PRINT SECOND BIT
1214 5000          JMP I TOCT /EXIT

/
/
/ROUTINE TO DO CRLF
/
1215 0000          UPONE, 0
1216 7300          CLA CLL
1217 1225          TAD K0215
1220 4436          TYPE
1221 1226          TAD K0212
1222 4436          TYPE
1223 4436          TYPE /TYPE ONE NULL
1224 5615          JMP I UPONE

/
1225 0215          K0215, 0215
1226 0212          K0212, 0212
/
/ROUTINE TO PRINT FOUR OCTAL
/
1227 0000          PROCT, 0
1230 7006          RTL
1231 7006          RTL
1232 3215          DCA UPONE
1233 1076          TAD M4
1234 3200          DCA TOCT
1235 1215          TAD UPONE
1236 0062          AND K0007
1237 1067          TAD K0260
1240 4436          TYPE
1241 1215          TAD UPONE
1242 7006          RTL
1243 7004          RAL

```

```

1244 3215      DCA  UPONE
1245 2200      ISZ  TOCT
1246 5235      JMP  ,=-11
1247 1321      YAD  K0240
1250 4436      TYPE
1251 5627      JMP I  FROCT

```

```

/
/SUBROUTINE TO PRINT TEXT
/

```

```

1252 0000      PRN,  0
1253 7300      CLA CLL
1254 1652      TAD I  PRN          /GET POINTER
1255 2252      ISZ  PRN
1256 3227      DCA  FROCT
1257 1627      TAD I  FROCT
1260 0322      AND  K7700
1261 7450      SNA
1262 5306      JMP  EXIT
1263 7500      SMA
1264 7020      CML
1265 7001      IAC
1266 7012      RTR
1267 7012      RTR
1270 7012      RTR
1271 4436      TYPE
1272 1627      TAD I  FROCT
1273 0323      AND  K0077
1274 7450      SNA
1275 5306      JMP  EXIT
1276 1311      YAD  K3740
1277 7500      SMA
1300 1310      TAD  K4100
1301 1321      TAD  K0240
1302 4436      TYPE
1303 2227      ISZ  FROCT
1304 7300      CLA CLL
1305 5257      JMP  PRN+5
1306 7300      EXIT, CLA CLL
1307 5652      JMP I  PRN

```

```

/
1310 4100      K4100, 4100
1311 3740      K3740, 3740
/

```

```

/ROUTINE TO TYPE
/

```

```

1312 0000      PRINT, 0
1313 6046      TLS
1314 6041      T8F
1315 5314      JMP  ,=-1
1316 6042      TCF
1317 7200      CLA
1320 5712      JMP I  PHINT
1321 0240      K0240, 0240
1322 7700      K7700, 7700

```

```

1323 0077      K0077, 0077
1324 0010      K0010, 10
1325 7741      K7741, 7741
1326 3600      K3600, 3600

```

```

/ROUTINE TO WAIT FOR KEY FROM OPERATOR
/

```

```

1327 0000      WAIT, 0
1330 7300      CLA CLL
1331 6032      KCC
1332 6031      K8F
1333 5332      JMP  ,=-1
1334 6036      KR0
1335 6046      T8S
1336 6041      T8F
1337 5336      JMP  ,=-1
1340 0370      AND  K0177
1341 1066      TAD  K0200
1342 3101      DCA  CHAR
1343 1101      TAD  CHAR
1344 3777*     DCA  C0CHAR
1345 2776*     ISZ  INMODE
1346 4775*     JMS  XC0CNT          /CHECK FOR CONTROL CHARACTERS.
1347 7200      CLA
1350 7200      CLA
1351 3776*     DCA  INMODE
1352 6032      KCC
1353 6042      TCF
1354 1101      TAD  CHAR
1355 7041      CIA
1356 1371      TAD  K0316
1357 7650      SNA CLA
1360 5727      JMP I  WAIT          /WAS IT A NO
1361 2327      ISZ  WAIT          /YES
1362 1101      TAD  CHAR          /UPDATE RETURN POINTER
1363 7041      CIA
1364 1372      TAD  K0331
1365 7650      SNA CLA
1366 2327      ISZ  WAIT          /WAS IT A YES
1367 5727      JMP I  WAIT          /WAS A YES
1370 0177      K0177, 0177          /WAS NEITHER
1371 0316      K0316, 0316
1372 0331      K0331, 0331
/

```

```

1375 3200
1376 3676
1377 3675
1400 1400

```

```

PAGE
/
/ROUTINE TO RECALIBRATE SELECTED DRIVE
/

```

```

1400 0000      RESTOR, 0
1401 7301      CLA CLL IAC          /ENABLE CLEAR CONTROL
1402 4445      CLRALL          /CLEAR CONTROL
1403 1100      TAD  DRIVNO          /CURRENT DRIVE

```



```

1404 1130      TAD  HOMEHA      /CURRENT FIELD
1405 4442      LDCMD          /LOAD COMMAND
1406 1120      TAD  EXBIT           /
1407 4446      LOSC           /LOAD EXTENDED DRIVE BIT
1410 7330      CLA CLL CML RAR    /MAYBE EXPECTED STATUS
1411 3117      DCA  GDMEG2       /SETUP COMPARE REGISTER
1412 7326      CLA CLL CML RTL    /ENABLE RECALIBRATE BIT
1413 4445      CLRALL          /"RECALIBRATE"
1414 4441      DSKSKP         /DISK SKIP IOT
1415 5214      JMP  .-1          /WAIT FOR FIRST DONE FLAG
1416 4440      ROSTAT         /READ STATUS
1417 1327      TAD  K2000        /
1420 7450      SNA           /WAS IT BUSY AND DONE
1421 5225      JMP  RESTA        /YES, THEN ITS O.K.
1422 1327      TAD  K2000        /NO, THEN IT MUST BE JUST DONE
1423 7640      SZA CLA          /WAS IT JUST DONE
1424 5243      JMP  RESTER       /NO, ERROR
1425 4445      RESTA, CLRALL     /CLEAR STATUS
1426 1066      TAD  K0200        /ENABLE SET SECOND DONE FLAG
1427 1121      TAD  CMREG        /ORIGINAL COMMAND
1430 4442      LDCMD          /LOAD COMMAND
1431 4441      DSKSKP         /DISK SKIP IOT
1432 5231      JMP  .-1          /WAIT FOR SECOND DONE
1433 4440      ROSTAT         /READ STATUS
1434 1070      TAD  K4000        /
1435 7640      SZA CLA          /WAS IT ONLY DONE FLAG
1436 5243      JMP  RESTER       /NO, ERROR STATUS
1437 7301      CLA CLL IAC       /ENABLE CLEAR CONTROL
1440 4445      CLRALL          /CLEAR CONTROL
1441 2200      ISZ  RESTOR       /UPDATE FOR GOOD RECALIBRATE
1442 5600      JMP I  RESTOR     /RETURN
1443 4437      RESTER, ERROR     /ERROR, STATUS
1444 3603      JMP I  RESTOR     /TEXT POINTER

```

```

1445 4452      /
1446 4447      CRLF  PRNTER      /PRINT "TRY RECALIBRATE"
1447 1756      ERMES2          /
1450 4454      RECEIV         /WAIT FOR INPUT
1451 5254      JMP  .+5          /TRY NEXT EXISTING DISK
1452 5245      JMP  .+5          /
1453 5201      JMP  RESTOR +1    /TRY AGAIN
1454 7301      CLA CLL IAC       /
1455 1056      TAD  AMOUNT        /GET AMOUNT ON SYSTEM
1456 7450      SNA           /WAS THERE ONLY 1 LEFT
1457 5535      JMP I  XEND        /LAST DISK
1460 3056      DCA  AMOUNT        /MORE TO GO BUT CLEAR THIS ONE
1461 3516      DCA I  TCNTRS      /CLEAR DISK POINTER
1462 5600      JMP I  RESTOR     /TRY NEXT ONE

```

```

/
/ROUTINE TO CHANGE DEVICE CODES

```

```

1463 0000      /
1464 4777*     CHANG, 0          /
1465 7010      JMS  XC8SW        /GET SWITCH REGISTER BITS,
RAR

```

```

1466 7620      SNL CLA          /CHANGE DEVICE CODES?
1467 5663      JMP I  CHANG       /NO.
1470 4777*     JMS  XC8SW        /GET SWITCHES,
1471 0313      AND  A0770        /
1472 3314      DCA  CSAVE1       /SAVE DESIRED
1473 1316      TAD  CCNTR1       /
1474 3315      DCA  CSAVE2       /
1475 1317      TAD  CHNPOT       /
1476 3200      DCA  RESTOR       /
1477 1600      CHANGH, TAO I RESTOR /GET ADDRESS POINTER
1500 3311      DCA  KWAIT        /
1501 1711      TAO I  KWAIT       /GET OLD CODE
1502 0312      AND  A7007        /MASK
1503 1314      TAD  CSAVE1       /ADD IN DESIRED
1504 3711      DCA I  KWAIT       /STORE DESIRED DEVICE CODE
1505 2200      ISZ  RESTOR       /UPDATE POINTER
1506 2315      ISZ  CSAVE2       /UPDATE CHANGE COUNTER
1507 5277      JMP  CHANGR       /
1510 5663      JMP I  CHANG       /EXIT TO PROGRAM.

```

```

/
1511 0000      KWAIT, 0          /
1512 7007      A7007, 7007       /
1513 0770      A0770, 0770       /
1514 0000      CSAVE1, 0         /
1515 0000      CSAVE2, 0         /
1516 7771      CCNTR1, 7771      /
1517 1520      CHNPOT, CHNPOT +1 /
1520 0734      IOT0             /
1521 0741      IOT1             /
1522 0746      IOT2             /
1523 0714      IOT3             /
1524 0705      IOT4             /
1525 0672      IOT5             /
1526 0727      IOT6             /
1527 2000      K2000, 2000       /

```

```

/THIS ROUTINE WILL GENERATE TIMING IF NEEDED BY THE APT SYSTEM

```

```

1530 0000      /
1531 7300      /KTICK, 0         /
1532 1022      CLL  CLA          /
1533 0070      TAO  22           /GET HARDWARE CONFIGURATION
1534 7650      AND  K4000        /
1535 5351      SNA  CLA          /ON APT?
1536 1730      JMP  EXTICK       /NO
1537 3353      TAD I  KTICK       /GET TIMING VALUE
1540 2132      DCA  COUNT        /ESTABLISH TIME
1541 5351      ISZ  CLKCNT       /
1542 1353      JMP  EXTICK       /RETURN
1543 3132      TAD  COUNT        /GET VALUE OF COUNTER
1544 2354      DCA  CLKCNT       /STORE IT
1545 5351      ISZ  CNT          /TIMING NEED BE DONE?
1546 4425      JMP  EXTICK       /
1547 1355      TIME             /
1550 3354      TAD  KCNT         /TIMING VALUE
DCA  CNT          /INIT SECOND COUNTER

```

```

1551 2330 EXTICK, ISZ KTICK /MOVE BEYOND TIMING VALUE
1552 5730 JMP I KTICK

1553 0000 COUNT, 0
1554 7776 CNT, -2
1555 7776 KCNT, -2
1556 0100 K0100, 0100
/
/
/ROUTINE TO NOTIFY APT OF USE IF REQUIRED
/

1557 0000 KTIME, 0
1560 0002 IOF /DISABLE INTERRUPTS
1561 0214 RUF /GET PRESENT DATA FIELD
1562 1075 TAD KCUF
1563 3364 DCA ,+1 /ESTABLISHES CURRENT DATA FIELD
1564 7402 HLT
1565 6272 CIF 70 /FIELD 7, LOCATION OF UV FROM
1566 4771 JMS I K6500
1567 7300 CLL CLA
1570 5757 JMP I KTIME

1571 6500 K6500, 6500
/

1577 3062 PAGE
1600 1600
/
/
/THIS ROUTINE WILL NOTIFY APT OF AN ERROR AND SEND PC TO
/APT SYSTEM. ALL ERRORS WILL RESULT IN PROGRAM HLT AND A TIME OUT ON
/APT. APT WILL TAKE OVER FROM THERE.

/

1600 0000 AENRU, 0
1601 0002 IOF /DISABLE INTERRUPTS
1602 7200 CLA
1603 1022 TAD 22 /CHECK FOR APT SYSTEM
1604 7700 SMA CLA
1605 5600 JMP I AENRU /RETURN NOT ON APT
1606 1621 TAD I KERRO /GET PC
1607 3222 DCA SAVPC
1610 0214 RDF /GET CURRENT DATA FIELD
1611 1075 TAD KCUF
1612 3214 DCA ,+2
1613 1222 TAD SAVPC
1614 7402 HLT
1615 6272 CIF 70 /REPLACED WILL CURRENT DATA FIELD
1616 5620 JMP I K6520 /CHANGE IF FOR APT RETURN TO FIELD 7
1617 7402 HLT /NOTIFIES APT OF ERROR

/

1620 6520 K6520, 6520
1621 0436 KERRO, ERRO
1622 0000 SAVPC, 0
/
/

```

```

/ROUTINE TO MOVE DISK POINTERS
/

1623 0000 MOVE, 0
1624 1237 TAD ADPT1
1625 5010 DCA AUTO10

1626 1240 TAD ADPT2
1627 3011 DCA AUTO11
1630 1077 TAD M10
1631 3241 DCA MCNTR1
1632 1410 TAD I AUTO10 /FROM HERE
1633 3411 DCA I AUTO11 /TO THERE
1634 2241 ISZ MCNTR1 /4 POINTERS
1635 5232 JMP ,+3
1636 5623 JMP I MOVE

/

1637 0137 ADPT1, DSK0A =1
1640 0150 ADPT2, DSK0B =1
1641 0000 MCNTR1, 0
/
/

1642 2003 TEXPC, TEXT "PC:"
1643 7200
1644 0704 TEXGO, TEXT "GO:"
1645 7200
1646 0530 TEXEX, TEXT "EX:"
1647 7200
1650 0315 TEXCM, TEXT "CM:"
1651 7200
1652 2324 TEXST, TEXT "ST:"
1653 7200
1654 0401 TEXDA, TEXT "DA:"
1655 7200
1656 0301 TEXCA, TEXT "CA:"
1657 7200
1660 0104 TEXAD, TEXT "AD:"
1661 7200
1662 0424 TEXDT, TEXT "DT:"
1663 7200

/

1664 2205 ERTX1, TEXT "READ STATUS ERROR"
1665 0104
1666 4023
1667 2401
1670 2425
1671 2340
1672 0522
1673 2217
1674 2200
1675 0411 ERTX2, TEXT "DISK DATA ERROR"
1676 2313
1677 4004
1700 0124
1701 0140
1702 0522

```

1703 2217
 1704 2200
 1705 2722 ERTX3, TEXT "WHITE STATUS ERROR"
 1706 1124
 1707 0540
 1710 2324
 1711 0124
 1712 2523
 1713 4005
 1714 2222
 1715 1722
 1716 0000
 1717 2205 ERTX4, TEXT "RECALIBRATE STATUS ERROR"
 1720 0301
 1721 1411
 1722 0222
 1723 0124
 1724 0540
 1725 2324
 1726 0124
 1727 2523
 1730 4005
 1731 2222
 1732 1722
 1733 0000

/
 1734 2422 ERMES1, TEXT "TRY TO FORMAT SAME CYLINDER AGAIN?"
 1735 3140
 1736 2417
 1737 4006
 1740 1722
 1741 1501
 1742 2440
 1743 2301
 1744 1505
 1745 4003
 1746 3114
 1747 1116
 1750 0405
 1751 2240
 1752 0107
 1753 0111
 1754 1677
 1755 0000
 1756 2422 ERMES2, TEXT "TRY TO RECALIBRATE SAME DISK AGAIN?"
 1757 3140
 1760 2417
 1761 4022
 1762 0503
 1763 0114
 1764 1102
 1765 2201
 1766 2405
 1767 4023
 1770 0115

1771 0540
 1772 0411
 1773 2313
 1774 4001
 1775 0701
 1776 1116
 1777 7700
 2000 2422 ERMES3, TEXT "TRY TO CHECK SAME CYLINDER AGAIN?"
 2001 3140
 2002 2417
 2003 4003
 2004 1005
 2005 0313
 2006 4023
 2007 0115
 2010 0540
 2011 0331
 2012 1411
 2013 1604
 2014 0522
 2015 4001
 2016 0701
 2017 1116
 2020 7700
 /
 2021 2213 TEXEND, TEXT "RK8E/RK8L DISK FORMATTER PASS COMPLETE"
 2022 7005
 2023 5722
 2024 1370
 2025 1440
 2026 0411
 2027 2313
 2030 4006
 2031 1722
 2032 1501
 2033 2424
 2034 0522
 2035 4020
 2036 0123
 2037 2340
 2040 0317
 2041 1520
 2042 1405
 2043 2405
 2044 0000
 2045 2213 MES1, TEXT "RK8E/RK8L DISK FORMATTER PROGRAM"
 2046 7005
 2047 5722
 2050 1370
 2051 1440
 2052 0411
 2053 2313
 2054 4006
 2055 1722
 2056 1501

```

2057 2424
2060 0522
2061 4020
2062 2217
2063 0722
2064 0115
2065 0000
2066 0617    MES2,  TEXT "FOR ALL QUESTIONS, ANSWER Y FOR YES OR N FOR NO."
2067 2240
2070 0114
2071 1440
2072 2125
2073 0523
2074 2411
2075 1716
2076 2354
2077 4001
2100 1623
2101 2705
2102 2240
2103 3140
2104 0617
2105 2240
2106 3105
2107 2340
2110 1722
2111 4016
2112 4006
2113 1722
2114 4016
2115 1756
2116 0000
2117 0617    MES3,  TEXT "FORMAT DISK "
2120 2215
2121 0124
2122 4004
2123 1123
2124 1340
2125 0000
2126 0122    MES4,  TEXT "ARE YOU SURE?"
2127 0540
2130 3117
2131 2540
2132 2325
2133 2205
2134 7700
2135 0617    MES5,  TEXT "FORMAT SAME DISK(S) AGAIN?"
2136 2215
2137 0124
2140 4023
2141 0115
2142 0540
2143 0411
2144 2313
2145 5023

```

```

2146 5140
2147 0107
2150 0111
2151 1677
2152 0000

/
2200 PAGE
/
2200 WRKBUF=.
/
2200 HITRK=.
2201 LDIRK=. +1
/
2577 ENDBUF=. +377
/

/CONSOL SRC =V2=R0= CONSOLE PACKAGE

/LAS= CALL C0CASH UR JMS XC0SW
/THIS WILL READ THE SWITCH REGISTER FROM THE PLACE SPECIFIED
/BY LOCATION 2W BIT 0.

/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 XC0PASS THIS LOCATION DETERMINDS 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.

/C0STR1 THIS IS FOUND IN CNTML ROUTINE CONTROL R PART
/IT IS THE RETURN WHEN CONTROL R IS ENTERED (RESTART PROGRAM)
/THE RETURN JUMPS TO XD0SW WHICH CONTAINS C0STR1 SO PUT THE LABEL C0STR1
/WHERE YOU WANT TO RESTART THE PROGRAM.

/SETUP1 IN XC0ERR THIS IS THE MASK BIT FOR HALT ON ERROR
/PLACE THE CORRECT BIT IN THIS LOCATION FOR HALTING ON ERRORS.

/SETUP2 IN XC0PASS 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.

/CONSOL=0
6661 PSKF= 6661
6662 PCLF= 6662

```

```

6663 PSKE= 6663
6664 PSTB= 6664
6665 PSIE= 6665
6004 GTF= 6004
7701 ACL= 7701
6007 CAF= 6007
7421 MQL= 7421
7501 MQA= 7501
/
3000 *3000
/
/*****
/CBPASS
/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 CBPASS TO BE IN THE 1 TO 4 MINUTE
/RANGE
/ CBPASS*JMS XCBPAS
/EX. OP CALL CBPASS
/ HLT /HALT IF NON CONSOL PACKAGE
/ JMP 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 XCBPAS ARE CHKCLA-XC0CLF-XC0OCTA-XC0SW-XC0PNT-XC0ING-

```

3000 0000 XCBPAS, 0
3001 7200 CLA
3002 4777 JMS CHKCLA /IS WORD 22 BIT 3 ACTIVE CONSOLE?
3003 5212 JMP DOPACK /IS CLASSIC
3004 4776 JMS C0GET /GET THE REGISTERS.
3005 4262 JMS XC0SW /DEACTIVE CONSOL CHECK SR SETTING
3006 0375 AND 400 /FOR HALT ON END OF CBPASS
3007 7640 SZA CLA /I= HALT 0 CONTINUE
3010 5600 JMP I XCBPAS /GO TO HALT
3011 5230 JMP C0BY1 /CONTINUE ON RUNNING PROGRAM
3012 4232 DOPACK, JMS CKCOUT /CLASS CHECK CBPASS COUNT
3013 5230 JMP C0BY1 /CBPASS COUNT NOT DONE REDO PROGRAM
3014 2250 ISZ PASCNT /CBPASS COUNT DONE SET CBPASS COUNT
3015 4774 JMS XC0CLF
3016 4303 JMS XC0PNT /CBPNT BUFFER
3017 3053 MESPAS
3020 1250 TAD PASCNT /GET NUMBER
3021 4773 JMS XC0OCTA /CONVERT IT TO ASCII
3022 4774 JMS XC0CLF /DO A LARRIAGE RETURN
3023 4776 JMS C0GET /GET THE REGISTERS.
3024 4262 JMS XC0SW /CHECK A HALT AT END OF CBPASS
3025 0375 SETUP2, AND 400 /MASK BIT
3026 7640 SZA CLA /HALT =1 NO SKIP CONTINUE =0
3027 4772 JMS XC0ING /STOP PROGRAM EXECUTION-LOOK FOR INPUT

```

```

3030 2200 C0BY1, ISZ XCBPAS /BUMP RETURN
3031 5600 JMP I XCBPAS
3032 0000 CKCOUT, 0
3033 1251 TAD D0SET /CHECK IF SET UP NEEDED
3034 7640 SZA CLA /0=SET UP CBPASS COUNT VALUE
/1=CBPASS COUNT VALUE OK
/CBPASS COUNT VALUE ON
/GET COUNT VALUE FOR THIS PROG
/SET TO NEGATIVE
/STORE IN MEME
/INDICATE VALUE SET UP
/COUNT THE NUMBER OF PASSES
/EXIT FOR ANOTHER PASS
/SET TO C0PNT CBPASS
/BUMP RETURN FOR
/CBPASS C0TYPE OUT
3047 0000 D0CNT, 0
3050 0000 PASCNT, 0
3051 0000 D0SET, 0
3052 0000 CNTVAL, 0
3053 0410 MESPAS, TEXT "DHRKD PASS "
3054 2213
3055 0404
3056 4040
3057 2001
3060 2323
3061 4000

```

/*****

/C0CKSW
/THIS ROUTINE CAN BE USED INPLACE OF A READ THE SWITCHES LAS,
/ROUTINE THAT WILL CHECK WHERE TO READ THE
/C0 SWITCHES FROM IE. FROM PANEL OR PSEUDO SWITCH REGISTER
/THE SELECTION IS DETERMINED BY THE STATE OF BIT 0 IN LOCATION 21.

```

/C0CKSW= JMS XC0SW /READ THE C0SWIT REGISTER
/EX. JMS XC0SW /RETURN WITH THE CONTENTS OF SWITCH REGISTER

```

/RETURN TO NEXT LOCATION FOLLOWING CALL WITH THE AC= TO VALUE OF C0SWIT SETTING

/CALLS USED ARE-XC0CKPA-

```

3062 0000 XC0SW, 0
3063 4771 JMS XC0CKPA /GO CHECK THE IF ANY CONTRL
3064 7000 NOP
3065 1021 TAD 21 /GET WD FOR INDICATOR
3066 7710 SPA CLA /CHECK IF FROM PANEL 4000
3067 7614 7614 /DO LAS AND SKIP GET FROM PANEL WITH LAS

```

3070 1020 TAD 20 /PSEUDO SWITCH
 3071 5662 JMP I XC8SW /EXIT WITH STATUS BIT IN AC.

 /C8TTY
 /THIS ROUTINE WILL LOOK FOR A INPUT FROM THE TERMINAL
 /AND REMOVE ANY PARITY BITS, THEN MAKE IT 8 BIT ASCII.
 / C8TTY= JMS XC8TTY
 /EX. JMS XC8TTY /READ CHAR FROM THE CONSOL DEVICE
 / / /RETURN TO CALL PLUS ONE AC CONTAINS THE CHAR

/CALLS USED =NONE= BUT C8CHAR IS OFF PAGE AND IN ROUTINE CALLED XC8ECHO

3072 0000 XC8TTY, 0
 3073 6031 KSF /LOOK FOR KEYBOARD FLAG
 3074 5273 JMP ,=1
 3075 6036 KR0 /GET CHAR
 3076 0370 AND (177 /MASK FOR 7 BITS
 3077 1367 TAD (200 /ADD THE EIGHTH BIT
 3100 3766* DCA C8CHAR /STORE IT
 3101 1766* TAD C8CHAR
 3102 5672 JMP I XC8TTY /EXIT

 /C8PRNT
 /THIS ROUTINE WILL TYPE THE CONTENTS OF THE C8 PRINT BUFFER. THE LOCATION
 /OF THE BUFFER WILL BE IN THE ADDR8 FOLLOWING THE CALL. PRINTING OF THE BUFFER
 /WILL STOP WHEN A 00 CHAR IS DETECTED. CHARACTERS ARE PACKED 2 PER WORD.

/ C8PRNT= JMS XC8PNT
 /EX. JMS XC8PNT /C8PNT THE CONTENTS OF THE FOLLOWING BUFFER
 / MESSAGE /LOCATION OF C8PRNT BUFFER

/C8PNT WILL USE THE LOCATION FOLLOWING THE CALL AS THE POINTER FOR THE
 /C8PNT ROUTINE, RETURN TO CALL PLUS TWO WITH AC= 0
 /CALLS USED ARE=XC8TYPE=XC8PNT

3103 0000 XC8PNT, 0
 3104 7300 CLA CLL
 3105 1703 TAD I XC8PNT /GET C8PNT BUFFERS STARTING LOCATION
 3106 3336 OCA PTSTOR /STORE IN PTSTOR

3107 2303 ISZ XC8PNT /BUMP RETURN
 3110 1736 C8D01, TAD I PTSTOR /GET DATA WORD
 3111 8365 AND (7700 /MASK FOR LEFT BYTE
 3112 7450 SNA /CHECK IF 00 TERMINATE
 3113 5703 JMP I XC8PNT /EXIT
 3114 7500 SHA /IS AC MINUS
 3115 7020 CML /MAKE CHAR A 300 AFTER ROTATE
 3116 7001 IAC /MAKE CHAR A 200 AFTER ROTATE
 3117 7012 RTR
 3120 7012 RTR
 3121 7012 RTR
 3122 4760* JMS XC8TYPE /PUT CHAR IN BITS 4-11 MAKE IT 8 BIT ASCII
 3123 1736 TAD I PTSTOR /C8PNT IT ON CONSOLE
 3124 8363 AND (0077 /GET DATA WORD
 3125 7450 SNA /MASK FOR RIGHT BYTE
 3126 5703 JMP I XC8PNT /CHECK IF 00 TERMINATOR
 3127 1362 TAD (3740 //EXIT
 3130 7500 SHA /ADD FUDGE FACTOR TO DETERMINE IF 200
 3131 1361 TAD (100 /OR 300 IS TO BE ADD TO CHAR
 3132 1360 TAD (240 /ADD 200
 3133 4760* JMS XC8TYPE /C8TYPE ONLY BITS 4-11
 3134 2336 ISZ PTSTOR /BUMP POINTER FOR NEXT WORD
 3135 5310 JMP C8D01 /DO AGAIN
 3136 0000 PTSTOR, 0 /STOR FOR C8PRNT BUFFER

 /C8PAUS
 /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.

/ C8PAUS= JMS XC8PAU
 /EX. JMS XC8PAUS /CHECK IF ON ACTIVE CONSOL IF NOT HALT HERE
 / ANYTHING /RETURN HERE IF ON ACTIVE CONSOL

/CALLS USED ARE =CHKCLA=

3137 0000 XC8PAU, 0
 3140 7300 CLA CLL
 3141 4777* JMS CHKCLA /CHECK LOC 22 BIT 3 CONSOLE BIT
 3142 5350 JMP C8D03 /GO DO CONSOL PART RETURN CALL +1
 3143 7040 CMA /DEACTIVATE CONSOL PACKAGE PUT HLT IN CALL
 3144 1337 TAD XC8PAU /GET CORRECT RETURN ADDR8
 3145 3337 OCA XC8PAU /SET UP RETURN
 3146 1357 TAD (7402 /GET CODE FOR HLT
 3147 3737 DCA I XC8PAU /PUT HALT IN CALL LOCATION
 3150 5737 C8D03, JMP I XC8PAU /GO TO HALT OR RETURN TO NEXT LOCATION

3157 7402
 3160 0240
 3161 0100
 3162 3740
 3163 0877
 3164 3677
 3165 7700
 3166 3675
 3167 0200
 3170 0177
 3171 3641
 3172 3435
 3173 3600
 3174 3623
 3175 0400
 3176 3424
 3177 4000
 3200

PAGE

/*****

/C0CNTX

/THIS ROUTINE WILL CHECK FOR THE PRESENCE OF CONTROL CHARACTERS
 /IT WILL CHECK FOR THE FOLLOWING CHAR C-R-Q-L-S
 / C0CNTX= JMS XC0CNT

/EX. JMS XC0CNTX /CHECK FOR CONTROL CHARACTER
 / JMP ANYTHING /LOC FOLLOWING CALL IS FOR CONTINUING THE PROGRAM
 / JMP ANYTHING /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=UPANOW=XC0TYI=XC0PSW-
 /
 /

3200 0000 XC0CNT, 0
 3201 3777* DCA ACSAVE /SAVE THE AC
 3202 4776* JMS CHKCLA /CHECK LOC.22 BITS FOR CONSOLE BIT
 3203 5206 JMP .+3 /ON ACTIVE CONSOLE
 3204 1777* TAD ACSAVE /DEACTIVE CONSOLEGET AC FOR RETURN
 3205 5600 JMP I XC0CNT /EXIT NOT ON ACTIVE CONSOLE
 3206 0004 GTF
 3207 3775* DCA FLSAVE
 3210 7501 MGA
 3211 3774* DCA MQSAVE /SAVE THE MQ
 3212 3255 DCA INDEXA /SET DISPLACEMENT INTO TABLE B
 3213 1257 TAD XTABLA /GET ADDRS OF TABLE A

3214 3256 DCA GETDAT /CONTAINS POINTER TO CONTROL CHAR
 3215 1656 REDDA, TAD I GETDAT /GET CONTROL CHAR FROM TABLE
 3216 7450 SNA /CHECK FOR A 0 END OF TABLE
 3217 5226 JMP UDNEA /END OF TABLE NO CONTROL CHAR
 3220 1773* TAD C0CHAR /COMPARE CHAR TO CONTROL CHAR
 3221 7650 SNA CLA /0 IF MATCH
 3222 5243 JMP GOITA /MATCH
 3223 2255 ISZ INDEXA /NO MATCH NOT END OF TABLE REDD
 3224 2256 ISZ GETDAT /BUMP INDEX FOR EXIT WHEN CONTROL FOUND
 3225 5215 JMP REDDA /BUMP GETDAT FOR COMPARE OF NEXT CNTRL CHAR.
 3226 1772* DNEA, TAD INMODE /CHECK IF PROGRAM EXPECTS CHAR
 3227 7640 SZA CLA /1=CHAR EXPECTED 0= NO CHAR EXPECTED
 3230 5240 JMP EXITA /CHAR EXPECTED
 3231 1773* TAD C0CHAR /GET CHAR = NOT CONTROL + NOT EXPECTED
 3232 4771* JMS XC0TYPE /C0PHNT CHAR
 3233 1370 TAD (277 /GET CODE FOR "?"
 3234 4771* JMS XC0TYPE
 3235 4767* JMS XC0CRLF
 3236 2200 ISZ XC0CNT /BUMP RETURN
 3237 5600 JMP I XC0CNT /EXIT CALL+2
 3240 2200 EXITA, ISZ XC0CNT /BUMP RETURN FOR MAIN PROGRAM CHECK OF CHAR
 3241 1773* TAD C0CHAR /PUT CHAR IN AC.
 3242 5600 JMP I XC0CNT /EXIT
 3243 1773* GOITA, TAD C0CHAR /GET THE CONTENTS OF CHAR
 3244 1366 TAD (100 /ADD 100 TO FORM A GOOD ASCII CHARACTER
 3245 3773* DCA C0CHAR /RESTORE CORRECT CHAR
 3246 1260 TAD XTABLB /GET START OF TABLE B
 3247 1255 TAD INDEXA /GET NOW FAR INTO TABLE
 3250 3254 DCA G0TOA /STORE IT
 3251 1654 TAD I G0TOA /GET THE ROUTINE STARTTING ADDRESS
 3252 3254 DCA G0TOA /STORE IT IN HERE
 3253 5654 JMP I G0TOA /GOTO CONTROL CHAR ROUTINE
 3254 0000 G0TOA, 0000 /ADD UP CNTRL ROUTINE TO EXECUTE
 3255 0000 INDEXA, 0000 /DISPLACEMENT INTO CNTRL TABLE
 3256 0000 GETDAT, 0000 /LOCATION OF ADDRS OF CONTROL CHAR.
 3257 3261 XTABLA, TABLA /ADDMS OF TABLE A
 3260 3271 XTABLB, TABLB /ADDMS OF TABLE B
 3261 7575 TABLA, 7575 /CNTRL C BACK TO MONITOR 203
 3262 7564 7564 /CNTRL L SWITCH ERROR PRINTING DEVICE 214
 3263 7557 7557 /CNTRL Q START DISPLAYING CHAR. AGAIN 221
 3264 7556 7556 /CNTRL R BACK TO BEGINNING OF PROGRAM 222
 3265 7555 7555 /CNTRL S STOP SENDING CHAR TO DISPLAY WAIT FOR CNTRL Q 223
 3266 7573 7573 /CNTRL E CONTINUE WITH PROGRAM 205
 3267 7574 7574 /CONTROL D CHANGE SWITCH REGISTER ON FLY
 3270 0000 0000
 3271 3307 TABLB, CNTRL C
 3272 3336 CNTRL L
 3273 3300 CNTRL Q
 3274 3311 CNTRL R
 3275 3320 CNTRL S
 3276 3344 CNTRL E
 3277 3400 CNTRL D
 /
 /CONTROL Q

```

/START SENDING CHAR. TO THE DISPLAY
/THIS WILL RETURN CONTROL TO CALL THAT WAS SET BY
/THE CALL FOR CONTROL S.
/
3300 3772* CNTRLQ, DCA INMODE /SET SUFT FLAG FOR UNEXPECTED CHAN
3301 1334 TAD C0SETS /CHECK IF CONTROL S TYPED IN
3302 7640 SZA CLA
3303 5306 JMP BYMETR /CONTROL S TYPED IN
3304 4765* JMS C0GET /NO CONTROL S TYPED PREVIOUSLY
3305 5600 JMP I XC0CNTR /LEAVE VIA CNTR ENTRY ADDRESS
3306 3334 BYMETR, DCA C0SETS /CLEAR THE SUFT FLAG
3307 4765* JMS C0GET /RESTORE REGISTERS
3310 5735 JMP I C0RETR /EXIT TO ADDRESS SET BY CONTROL S
/
/CONTROL R
/GO TO THE QUESTION C0SWIT
3311 3764* CNTRLR, DCA TTYLPT /CLEAR THE TYPE FLAG SET TO TTY
3312 3334 DCA C0SETS /CLEAR SOFT FLAG FOR CNTRL S
3313 3772* DCA INMODE
3314 4763* JMS UPAROW /PRINT THE " AND C0CHAR
3315 3762* C0BY4, DCA C0BWSY /CLEAR FLAG FOR CNTRL D OR N
3316 5717 JMP I X0D5W /GO TO ADDR5 OF C0SWIT
3317 0200 X0D5W, BGN /0D5W IS LABEL FOR C0SWIT QUESTION
/
/CONTROL S
/STOP SENDING CHAR. TO DISPLAY UNTIL A "U IS RECEIVED
/
3320 1334 CNTRL5, TAD C0SETS /IF! DU NOT STORE IN C0RETR
3321 7640 SZA CLA
3322 5326 JMP C0D07 /DONT SET UP C0RETR
3323 7001 IAC /MAKE RETURN CALL PLUS 2
3324 1200 TAD XC0CNT /GET RETURN FOR THIS CALL
3325 3335 DCA C0RETR /STORE IT HERE FOR USE BE CNTRL W
3326 2334 C0D07, ISZ C0SETS /SET FLAG TO SAVE CALL
3327 4761* JMS XC0TTYI /LOOK FOR THE INPUT
3330 4765* JMS C0GET /GET REGISTERS
3331 4200 JMS XC0CNTR /CHECK FOR THE CONTROL CHAR
3332 7200 CLA
3333 5320 JMP CNTRL5 /IF NOT A CNTRL Q R C REASK
3334 0000 C0SETS, 0
3335 0000 C0RETR, 0
/
/SWITCH OUTPUT FROM ONE OUTPUT DEVICE TO ANOTHER - THE TWO OUTPUTS ARE THE
/CONSOLE AND THE PRINTER WITH DEVICE CODE 06.
/
3336 1764* CNTRL6, TAD TTYLPT /GET PRESENT C0SWIT INDICATOR
3337 7040 CMA /COMPLEMENT IT
3340 3764* DCA TTYLPT /STOR NEW C0SWIT
3341 4763* JMS UPAROW /C0PHNT " AND CHAR ON NEW DEVICE
3342 4765* JMS C0GET /RESTORE THE REGISTERS
3343 5600 JMP I XC0CNT /EXIT

```

```

/CONTROL E
/CONTINUE RUNNING FROM A INQUIRE OR ERROR
/
3344 4763* CNTRLE, JMS UPAROW /PRINT THE CONTROL CHAR
3345 4765* JMS C0GET /GET THE REGISTERS
3346 5600 JMP I XC0CNT /RETURN TO CALL PLUS ONE
/
/CONTROL C
/RETURN TO MONITOR CONTROL C
3347 3764* CNTRLC, DCA TTYLPT /CLEAR THE LPT FLAG TO PRINT ON DISPLAY
3350 4763* JMS UPAROW /C0PHNT " AND LETTER IN CHAR
3351 6203 CDF CIP /GO TO 0 PLD
3352 6007 CAF /CLEAR THE WORLD
3353 5760 JMP I L7600 /GO TO DIAGNOSTIC MONITOR
/*****
/
/
3360 7600
3361 3072
3362 3545
3363 3415
3364 3721
3365 3424
3366 0100
3367 3023
3370 0277
3371 3677
3372 3676
3373 3675
3374 4123
3375 4124
3376 4000
3377 4122
3400 PAGE
/
/CONTROL D
/CHANGE THE SWITCH REGISTER ANYTIME CNTRL D AND RETURN TO
/THE PROGRAM RUNNING.
3400 4215 CNTRLD, JMS UPAROW
3401 1213 TAD C0SEY0 /CHECK IF THE RETURN ADDR5 IS SAFE
3402 7640 SZA CLA
3403 5207 JMP C0D011 /DO NOT CHANGE THE RETURN ADDR5
3404 1777* TAD XC0CNT /GET THE RETURN ADDR5 AND SAVE IT
3405 3214 DCA C0SEY0 /SAVE THE RETURN HERE
3406 2213 ISZ C0SEY0 /INDICATE RETURN SAVED DONT DESTROY
3407 4256 C0D011, JMS XC0PSW /GO CHANGE THE SWITCH REGISTER
3410 3213 DCA C0SEY0 /CLEAR THE FLAG
3411 4224 JMS C0GET /RESTORE THE AC MQ LINK ETC

```



```

3412 5614      /      JMP I  C0RETD      /RETURN TO THE PROGRAM
3413 0000      C0RETD, 0
3414 0000      C0RETD, 0

      /THIS WILL TYPE A UP ARROW AND THE CHAN IN C0CHAR.

3415 0000      UPAROW, 0      /C0PMT THE "" AND THE CHAR C0TYPED IN
3416 1376      TAD      (336      /CODE FOR "
3417 4775*     JMS      XC8TYPE
3420 1774*     TAD      C0CHAR      /C0TYPE THE CHAR
3421 4775*     JMS      XC8TYPE
3422 4773*     JMS      XC8CRLF
3423 5615      JMP I  UPAROW      /EXIT
    
```

```

/*****

3424 0000      C0GET, 0
3425 7200      CLA
3426 1772*     TAD      M0SAVE
3427 7421*     MQL      /RESTORE M0
3430 1771*     TAD      FLSAVE
3431 7004      RAL      /RESTORE THE LINK
3432 7200      CLA
3433 1770*     TAD      ACSAVE      /RESTORE THE AC
3434 5624      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 XC8INQ

/EX.  JMS      XC8INQ      /C0 WILL PRINT A WAITING AND WAIT FOR INPUT
/      DO ANYTHING      /RETURN IS CALL PLUS ONE AC = 0 CONTINUE

/CALLS USED ARE =CHKCLA-XC8PNT-XC8TYI-C0GET-XC8CNTR=
    
```

```

3435 0000      XC8INQ, 0
3436 7300      CLA CLL
3437 4767*     JMS      CHKCLA      /CHECK LOC 22 BIT 3 CONSOLE BIT
3440 7410      SKP
3441 5635      JMP I  XC8INQ      /NOT CONSOLE LEAVE
    
```

```

3442 4766*     JMS      XC8PNT
3443 3451      WATMES
3444 4765*     JMS      XC8TTYI      /INQUIRE WAITING
3445 4224      JMS      C0GET      /GET CHARACTER
3446 4777*     JMS      XC8CNTR      /CHECK IF CONTROL CHARACTER
3447 5635      JMP I  XC8INQ      /EXIT AND CONTINUE
3450 5236      JMP      XC8INQ+1      /REASK
3451 2701      WATMES, TEXT "WAITING "
3452 1124
3453 1116
3454 0740
3455 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 XC8PSW

/EX.  JMS      XC8PSW      /SET UP PSEUDO C0SWIT REGISTER IF
/ON THE CONSOLE PACKAGE. RETURN IS CALL PLUS ONE AC = 0

/CALLS USED ARE =CHKCLA-XC8PSW-XC8PNT-XC8OCTA-XC8TYPE=
    
```

```

3456 0000      XC8PSW, 0
3457 4767*     JMS      CHKCLA      /CHECK LOC 22 BIT 3 CONSOLE BIT
3460 7410      SKP
3461 5656      JMP I  XC8PSW      /DEACTIVE CONSOLE PACKAGE
3462 1345      TAD      C0SWST      /RETURN WITHOUT ASKING PSEUDO SWITCH
3463 7640      SZA CLA      /IS THE SWPT FLAG SET FOR SWITCH?
3464 5764*     JMP      C0BY4      /SKIP IF ONE ENTRY AT A TIME OK
3465 2345      ISZ      C0SWST      /SECOND ENTRY WITH OUT A EXIT GO TO SW QUESTION
3466 4766*     C0NDPS, JMS      XC8PNT      /FIRST ENTRY SET FLAG
3467 3547      MESA
3470 1020      TAD      20      /C0PMT SR=
3471 4763*     JMS      XC8OCTA      /GET CONTENTS OF SW
3472 1362      TAD      (40      /CONVERT IT TO ASCII
3473 4775*     JMS      XC8TYPE      /GET SPACE
3474 2761*     ISZ      INMODE      /SET FLAG FOR CHAR EXECUTED
3475 4760*     JMS      XC8ECHO      /LOOK FOR INPUT
3476 4315      JMS      TSTCHA      /NOT CONTROL TEST IT IS LEGAL
3477 1774*     TAD      C0CHAR      /STORE NEW CHAR IN SW REG
3500 3020      DCA      20

3501 1357      TAD      (=3      /GET A MINUS 3
3502 3346      DCA      TPCNT      /STORE IN TEMP COUNT
    
```

```

3503 4760' GETCH1, JMS XC8ECHO /GET NEXT CHAR
3504 4315 JMS TSTCHA /CHECK IF CR + GOOD CHAR
3505 1020 TAD 20 /GET C8SWT REGISTER
3506 7106 RTL CLL /ROTATE IT LEFT 3 PLACES
3507 7004 RAL
3510 1774' TAD C8CHAR /GET CHAR + ADD IT TO PREVIOUS CONTENTS
3511 3020 DCA 20 /SAVE NEW CONTENTS
3512 2346 ISZ TPCNT /BUMP COUNT
3513 5303 JMP GETCH1 /JMP BACK + GET NEXT CHAR
3514 5342 JMP ENUIT /END 4 CHAR C8TYPED IN
3515 0000 TSTCHA, 0
3516 7041 CIA /CMPL CHAR IN AC
3517 1356 TAD (215 /TEST IF IT IS A CARRIAGE RETURN
3520 7650 SNA CLA /SKIP IN NOT CR.
3521 5342 JMP ENUIT /WAS CARRIAGE RETURN
3522 1774' TAD C8CHAR /NOT CM, GET CHAR
3523 1355 TAD (=260 /CHECK IF IT IS IN RANGE
3524 7710 SPA CLA /IF NOT POSITIVE C8ERR CHAR SMALLER THEN 260
3525 5336 JMP ERR1 /C8EMR = CHAN TOO SMALL
3526 1774' TAD C8CHAR /GET CHAR
3527 1354 TAD (=270 /GET A =270 + CHECK IF IT IS LARGER THEN 7
3530 7700 SNA CLA /SKIP IF LESS THEN 7
3531 5336 JMP ERR1 /C8EMR ON CHAR NOT IN RANGE
3532 1774' TAD C8CHAR /GET CHAR
3533 0353 AND (7 /MASK FOR RIGHT BYTE
3534 3774' DCA C8CHAR /STORE IN CHAR
/GET CHAR IN AC
3535 5715 JMP I TSTCHA /EXIT
3536 1352 ERR1, TAD (277 /CBPHNT
3537 4775' JMS XC8TYPE /?
3540 4773' JMS XC8CRLF /
3541 5206 JMP C8DPS /EXIT + ASK AGAIN
3542 4773' ENUIT, JMS XC8CRLF /DO A CR LF
3543 3345 DCA C8SWST /CLEAR THE PSW ENTRY FLAG
3544 5656 JMP I XC8PSW /EXIT ROUTINE
3545 0000 C8SWST, 0
3546 0000 TPCNT, 0
3547 2322 MESA, TEXT "SH= "
3550 7540
3551 0000

```

```

3552 0277
3553 0007
3554 7510
3555 7520
3556 0215
3557 7775
3560 3663
3561 3676
3562 0040
3563 3600
3564 3315
3565 3072

```

```

3566 3103
3567 4000
3570 4122
3571 4124
3572 4123
3573 3623
3574 3675
3575 3677
3576 0336
3577 3200
3600 PAGE
3600 /C8OCTA
/DOCTAL 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
/ C8OCTA= JMS XC8OCT
/
/EX, JMS XC8OCTA /AC CONTAINS NUMBER TO BE CHANGE
/ RETURN IS TO CALL PLUS ONE AC=0
/
/CALLS USED ARE =XC8TYPE=
3600 0000 XC8OCT, 0
3601 7106 CLL RTL
3602 7006 RTL /POSITION THE FIRST CHAR FOR PRINTING
3603 3221 DCA C8TMP1 /SAVE CORRECT POSITIONED WORD HERE
3604 1377 TAD (=4
3605 3222 DCA C8CKP /STORE COUNTER IN HERE
3606 1221 C8OCT, TAD C8TMP1 /GET FIRST NUMBER
3607 0376 AND (0007 /MASK
3610 1375 TAD (260 /ADD THE PRINT CONSTANT
3611 4277 JMS XC8TYPE /TYPE THE NUMBER
3612 1221 TAD C8TMP1 /
3613 7006 RTL
3614 7004 RAL /PUT NEXT NUMBER IN POSITION
3615 3221 DCA C8TMP1 /STORE IT
3616 2222 ISZ C8CKP /DONE YET WITH FOUR NUMBERS
3617 5206 JMP C8OCT /NOT YET DO MORE
3620 5600 JMP I XC8OCT /DONE WITH FOUR
3621 0000 C8TMP1, 0
3622 0000 C8CKP, 0
/*****
/C8CRLF
/C8TYPE CR AND LF WITH FILLERS FOLLOWING EACH LF AND CR
/
/ C8CRLF= JMS XC8CRLF
/EX, JMS XC8CRLF /C8PMNT A CR AND LF WITH FILL
/ /RETURN TO CALL PLUS ONE AC =0

```

/CALLS USED ARE =XC8TYPE=

```

3623 0000 XC8CRLF,0
3624 7300 CLA CLL
3625 1374 TAD (215 /GET CODE FOR CR
3626 4277 JMS XC8TYPE
3627 1237 TAD FILLER
3630 7000 CMA
3631 3240 OCA FILCNT /STONE FILLER IN HERE
3632 1373 TAD (212 /GET CODE FOR LF
3633 4277 C8D02, JMS XC8TYPE
3634 2240 ISZ FILCNT /CHECK ON FILLER CHAR
3635 5233 JMP C8D02 /TYPE A NON PRINTING CHAR
3636 5623 JMP I XC8CRLF /EXIT
3637 0004 FILLER, 0004 /FILLER SET FOR 4 CHAR
3640 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=O= IT WILL DO
/THE CONTROL FUNCTION AND RETURN TO CALL PLUS 2.
/A NON CONTROL CHARACTER WILL BE PRINTED AND A "?" IT WILL RETURN TO
/CALL PLUS 2.
/IF NO FLAG IS SET OR THE CONSUL 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=XC8CNTR=C8GET=

```

3641 0000 XC8CKP, 0
3642 3772 DCA ACSAVE /SAVE THE AC
3643 0004 GTF /SAVE THE FLAGS
3644 3771 OCA FLSAVE /SAVE THE FLAGS
3645 7501 MQA /PUT MQ IN AC
3646 3770 OCA MMSAVE /SAVE THE MQ
3647 0031 KSF /CHECK THE KEYBOARD FLAG
3650 5201 JMP C8BY3 /EXIT TO CALL PLUS 1
3651 4767 JMS CHKCLA /CHECK LOC 22 BIT 3 CONSOLE BIT
3652 7410 SKP /ACTIVE CONSOLE PACKAGE

```

```

3653 5201 JMP C8BY3 /EXIT TO CALL PLUS 1
3654 4766 JMS XC8TTYI /GET THE CHAR
3655 4765 JMS C8GET /GET THE FLAGS
3656 4764 JMS XC8CNTR /CHECK IF CONTROL CHAR.
3657 7000 NOP /RETURN IF A CONTINUE CHAR.
3660 2241 ISZ XC8CKP /BUMP RETURN FOR CALL PLUS 2
3661 4765 C8BY3, JMS C8GET /GET REGISTERS
3662 5641 JMP I XC8CKP /SAY 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 & PRINT IT
/RETURN CALL PLUS ONE AC = CHAR C8TYPED IN

```

/CALLS USED ARE =XC8TTYI=XC8CNTR=C8GET=XC8ECH=XC8TYPE=

```

/
XC8ECH, 0
3663 0000 JMS XC8TTYI /WAIT FOR CHAR FROM KEYBOARD
3664 4766 JMS C8GET /RESTORE THE REGISTERS
3665 4765 ISZ INMODE /SET INMODE IDENTIFYING THIS AS A EXPECTED CHAR
3666 2276 JMS XC8CNTR /GO CHECK IF IT IS A CONTROL CHAR
3667 4764 JMP I XC8ECH /WAS A CONTROL CHAR - CONTINUE RUNNING
3670 5663 JMS XC8TYPE /NOT A CONTROL CHAR & PRINT IT
3671 4277 OCA INMODE /CLEAR FLAG THAT CHAR EXPECTED
3672 3276 TAD C8CHAR /GET CHAR IN AC
3673 1275 JMP I XC8ECH /EXIT
3674 5663 C8CHAR, 0
3675 0000 INMODE, 0
3676 0000

```

```

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

```

```

/EX. JMS XC8TYPE /PRINT THE CHAR IN THE AC.
/RETURN CALL PLUS ONE AC = 0000
/DO NOT CLEAR THE LINK IN THIS ROUTINE NEEDED BY C8OCT

```

/CALLS USED ARE =C8HANG=XC8CNTR=XC8PNT=XC8CRLF=XC8INQU=

```

3677 0000 XC8TYP, 0
3700 5320 DCA PNTBUF /STONE CHAR
3701 1321 TAD TTYLPT /CHECK O=TTY 7777=LPT
3702 7640 BZA CLA
3703 5312 JMP XDOLPT /DO OUT PUT ON LPT
3704 1320 TAD PNTBUF

```

```

3705 6046      TLS
3706 6041      TSF
3707 5306      JMP      =-1
3710 6042      TCF
3711 5316      JMP      C0BY5
3712 1320      XDULPT, TAD      PNTBUF      /GET CHAR
3713 6666      PSTB      PCLF      /CBPNT IT
3714 4322      JMS      C0HANG      /CHECK KEYBOARD IF HUNG
3715 6662      PCLF      /CLEAR THE FLAG
3716 7600      C0BY5, 7600      /CLEAR THE AC
3717 5677      JMP I      XC0TYP      /EXIT
3720 0000      PNTBUF, 0
3721 0000      TTYLPT, 0

3722 0000      C0HANG, 0
3723 7200      CLA
3724 1316      TAD      C0BY5      /GET CONSTANT 7600
3725 3320      DCA      PNTBUF      /PNTBUF IS NOW A COUNTER
3726 6661      PSKF      /SKIP ON PINTER DONE
3727 7410      SRP      /NOT DONE YET
3730 5722      JMP I      C0HANG      /SAM FLAG DONE
3731 2345      ISZ      C0CONT      /FIRST COUNTER FAST ONE
3732 5326      JMP      =-4      /CHECK IF FLAG SET YET
3733 2320      ISZ      PNTBUF      /MADE 4096 COUNTS ON FAST COUNTER
3734 5331      JMP      =-5      /KEEP IT UP FOR 5 SEC
3735 1764      TAD      XC0CNTR      /GET THE RETURN ADDRESS IN CONTROL
3736 3322      DCA      C0HANG      /SAVE IT IN HANG
3737 3321      DCA      TTYLPT      /ALLOW PRINTING ON TTY
3740 4763      JMS      XC0PNT
3741 3746      MESHANG
3742 4223      JMS      XC0CRLF      /LPT ERROR
3743 4762      JMS      XC0INGU      /PRINT WAITING
3744 5722      JMP I      C0HANG      /CONTINUE TO SAVE ADDRESS
3745 0000      C0CONT, 0      /COUNTER FOR TIMER
3746 1420      MESHANG,TEXT "LPT ERROR"
3747 2440
3750 0522
3751 2217
3752 2200

3762 3435
3763 3103
3764 5200
3765 3424
3766 3072
3767 4300
3770 4123
3771 4124
3772 4122
3773 0212
3774 0215
3775 0260
3776 0007
3777 7774

```

```

4000 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.

4000 0000      CHKCLA, 0
4001 7200      CLA
4002 1022      TAD      22      /GET THE CONTENTA OF LOCATION 22
4003 0377      AND      (400      /MASK FOR BIT 3 (400
4004 7650      SNA CLA      /
4005 2200      ISZ      CHKCLA      /ACTIVE CONSOLE PACKAGE RETURN
/CALL PLUS ONE (1) FOR ACTIVE
4006 5600      JMP I      CHKCLA      /DEACTIVE CONSOLE PACKAGE RETURN
/CALL PLUS TWO (2)

/CBERR
/THIS ROUTINE WILL DETERMINE WHAT TO DO WHEN A CBERR IS ENCOUNTERED
/WILL CHECK IF CLASSIC SYSTEM, WILL CHECK C0SWIT REGISTERS.
/      CBERR= JMS XC0ERR
/Ex. JMS      XC0ERR      /GO TO CBERR CALL IF NOT CONSOLE
/      /RETURN IS CALL PLUS ONE AC =0000

/CALLS USED ARE =CHKCLA=XC0CRLF=XC0SW=XC0INGU=XC0PNT=XC0OCTA=

4007 0000      XC0ERR, 0
4010 6002      IOF
4011 3322      DCA      AC0SAVE      /SAVE AC
4012 6004      GTF
4013 3324      DCA      FL0SAVE      /SAVE THE FLAGS
4014 7501      MOA
4015 3323      OCA
4016 7340      CLA CLL CMA      /SAVE THE MO
4017 1207      TAD      XC0ERR      /SUBTRACT A 1 FOR TRUE LOCATION
4020 3321      DCA      PC0SAVE      /GET RETURN LOCATION
4021 4200      JMS      CHKCLA      /SAVE ADD OF CBERR CALL
4022 7410      SRP      /CHECK LOC.22 BIT 3 CONSOLE BIT
4023 5263      JMP      NTCLAS      /ACTIVE CONSOLE PACKAGE
4024 4776      JMS      C0SET      /NOT CLASSIC SYSTEM
4025 4775      JMS      XC0SW      /GET THE REGISTERS,
/CHECK SWITCH REG FOR BIT THAT INDICATES
/NO ERROR MESSAGE
4026 0374      SETUP1, AND      (0000      /MASK FOR BIT FOR NO ERROR PRINTING
/IF THIS ERROR MESSAGE IS TO ALWAYS
/BE PRINTED LEAVE AND VALUE AT 0000
/SKIP IF BIT IS 0 PRINT ERROR MESSAGE
/DO NOT PRINT

4027 7640      SZA CLA
4030 5255      JMP      C0DD10
4031 4773      JMS      XC0CRLF
4032 4772      JMS      XC0PNT
4033 4075      ERRMES      /PRINT THE ERROR MESSAGE
4034 4772      JMS      XC0PNT

```

```

4035 4105      MESPC      /PRINT THE PC STATEMENT
4036 1321      TAD        PCSAVE
4037 4771'     JMS        XCBOCTA /CONVERT 4 DIGIT PC TO ASCII
4040 4772'     JMS        XCBPNT
4041 4110      MESAC      /PRINT THE AC MESS
4042 1322      TAD        ACSAVE
4043 4771'     JMS        XCBOCTA
4044 4772'     JMS        XCBPNT
4045 4113      MESMQ      /PRINT MQ
4046 1323      TAD        MQSAVE
4047 4771'     JMS        XCBOCTA
4050 4772'     JMS        XCBPNT
4051 4116      MESFL      /PRINT FL
4052 1324      TAD        FLSAVE
4053 4771'     JMS        XCBOCTA
4054 4773'     JMS        XCBCLRF
4055 4776'     C0DU10, JMS  C0GET  /GET THE REGISTERS.
4056 4775'     JMS        XCB0SW /CHECK SWITCH REGISTER
4057 7610      SKP CLA    /SKIP IF BIT 0 SET
4060 5273      JMP        C0BY2  /LEAVE
4061 4770'     JMS        XCBING  /GO TO THE INQUIRE ROUTINE
4062 5273      JMP        C0BY2  /LEAVE
4063 4776'     NTCLAS, JMS  C0GET  /GET THE REGISTERS.
4064 4775'     JMS        XCB0SW /CHECK PSEUDO SWITCH REGISTER
                                /CHECK THE C0SWIT REGISTER
                                /SKIP IF MALT
4065 7610      SKP CLA    /NO MALT CONTINUE
4066 5607      JMP I      XCBERR  /CODE FOR MLT
4067 1367      TAD        C7402  /PUT IT IN CALL LOG.
4070 3721      DCA I      PCSAVE
4071 4776'     JMS        C0GET
4072 5721      JMP I      PCSAVE /EXIT TO CALL AND MALT
4073 4776'     C0BY2, JMS  C0GET  /GET THE REGISTERS
4074 5607      JMP I      XCBERR
4075 0410      ERNMES, TEXT "DMRKDD FAILED "
4076 2213
4077 0404
4100 4040
4101 0601
4102 1114
4103 0504
4104 4000
4105 4040      MESPC, TEXT " PC:"
4106 2003
4107 7200
4110 4040      MESAC, TEXT " AC:"
4111 0103
4112 7200
4113 4040      MESMQ, TEXT " MQ:"
4114 1521
4115 7200
4116 4040      MESFL, TEXT " FL:"
4117 0614
4120 7200
4121 7777      PCSAVE, 7777
4122 7777      ACSAVE, 7777

```

```

4123 7777      MQSAVE, 7777
4124 7777      FLSAVE, 7777

                                $$$

4167 7402
4170 3435
4171 3600
4172 3103
4173 3623
4174 0000
4175 3062
4176 3424
4177 0400

```

```

0000 11110000 11000000 11111111 11111111 11111111 11111111 11111111 11111111
0100 11111111 11111111 11111111 11111111 11111111 11111111 11000000 00000000

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11100111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0500 11111111 11111111 11111111 11111111 11111111 11111111 00000000 00000111

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 11111111 11111111 11111111 11111111 11111111 11110000 00000111

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

1200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11100111

1400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11000001

1600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

2000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
2100 11111111 11111111 11111111 11111111 11111111 11100000 00000000 00000000

2200
2300

2400
2500

2600
2700

3000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3100 11111111 11111111 11111111 11111111 11111111 10000001 11111111 11111111

3200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3300 11111111 11111111 11111111 11111111 11111111 11110000 11111111 11111111

3400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

3600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
3700 11111111 11111111 11111111 11111111 11111111 11100000 00111111 11111111

```

```

4000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
4100 11111111 11111111 11110000 00000000 00000000 00000000 00000001 11111111

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

```

A0770	1513	CHKCLA	4000	DSK5B	0156	INMODE	3676
A7007	1512	CHKDAT	0353	USK6A	0146	IO7K	1734
ACL	7701	CHKDSK	0357	USK6B	0157	IO71	0741
ACSAVE	4122	CHKER	1112	DSK7A	0147	IO72	0746
ADPOT1	0137	CHKR1	1007	DSK7B	0100	IO73	0714
ADPOT2	0150	CHNPOT	1517	DSKCNT	0105	IO74	0705
ADPT1	1637	CKCOUT	3052	DSKP	0741	IO75	0672
ADPT2	1640	CLDR	0745	DSKSKP	4441	IO76	0727
ADREG	0125	CLKCNT	0132	OTREB	0126	IO7CHN	4430
AERRO	1600	CLRALL	4445	ENDBUF	2577	K0003	0060
AGAIN	0533	CMREG	0121	ENDIT	3542	K0007	0062
ALLAGN	0220	CNT	1554	ENDT0T	0424	K0010	1324
AMOUNT	0056	CNTRLC	3347	ERHLT0	0736	K0037	0074
APT8	1125	CNTRLD	3400	ERHLT2	0750	K0040	0063
APT8A	4424	CNTRLE	3344	ERHLT3	0716	K0077	1523
AUTO10	0010	CNTRLL	3336	ERHLT4	0707	K0100	1556
AUTO11	0011	CNTRLQ	3300	ERHLT5	0674	K0177	1370
BGN	0200	CNTRLR	3411	ERHLT6	0731	K0200	0066
BGNBUF	0055	CNTRLB	3320	ERMES1	1734	K0212	1226
BGNTST	0127	CNTVAL	3052	ERMES2	1756	K0215	1225
BYRETR	3306	COUNT	1553	ERMES3	2000	K0240	1321
C0BY1	3030	CRLF	4452	ERR1	3536	K0260	0067
C0BY2	4073	CSAVE1	1514	ERMES	4075	K0277	0065
C0BY3	3661	CSAVE2	1515	ERRU	0436	K0316	1371
C0BY4	3315	DAREG	0123	ERRON	4437	K0331	1372
C0BY5	3716	DATCNT	0131	ERTX1	1664	K0400	0073
C0CHAR	3675	DATER	1103	ERTX2	1675	K2000	1527
C0CKP	3622	DATOK	1071	ERTX3	1705	K3600	1526
C0CONT	3745	DCLR	0742	ERTX4	1717	K3740	1311
C0D01	3110	DLAG	0743	EXBIT	0120	K4	0061
C0D010	4055	DLCA	0744	EXIT	1306	K4000	0070
C0D011	3407	DLDC	0746	EXITA	3240	K4100	1310
C0D02	3633	DLSC	0740	EXITLK	1551	K5000	0067
C0D03	3150	DHAN	0747	FILCNT	3640	K6500	1571
C0D04	3606	DOCNT	3047	FILLER	3637	K6520	1620
C0D07	3326	DONE	0230	FLSAVE	4124	K7377	1173
C0GET	3424	DONEA	3226	FORMAT	0302	K7400	0764
CBHANG	3722	DOPACK	3012	FRMSK	0263	K7577	0670
CBRDP8	3466	D08EY	3031	FROCT	1227	K7700	1322
CBRETD	3414	DRIVNO	0100	FRSTUK	1056	K7735	0071
CBRETR	3335	DMST	0745	GDRG2	0117	K7741	1325
CBSETO	3413	DSK0A	0140	GETCM1	3503	K7760	0072
CBSETS	3334	DSK0B	0131	GETDAT	3236	K7771	0557
CBSHST	3545	DSK1A	0141	GOITA	3243	KAERRO	4426
CBTMP1	3621	DSK1B	0132	GOTDA	3234	KCDF	0075
CAF	6007	DSK2A	0142	GTF	6004	KCNT	1555
CAREG	0124	DSK2B	0133	MEDLST	0333	KERRO	1621
CNTR1	1516	DSK3A	0143	MEDTAD	0332	KILBUF	4435
CHANG	1463	DSK3B	0154	HIGHAD	0103	KLBUF	0752
CHANGR	1477	DSK4A	0144	HITHK	2200	KTICK	1530
CHAR	0101	DSK4B	0135	HOMEMA	0130	KTIME	1557
CHECK	0400	DSK5A	0145	INDEXA	3235	KWAIT	1511

L0AD	0711	PRNTR	4447	TEXT	1632	XRDST	0040
L0ADD	4444	PSIE	6665	TICK	4427	XRDTRK	0032
L0CA	0700	PSKE	6663	TIME	4425	XREG	0546
L0CM	0720	PSKF	6661	TMPCNT	3546	XRESTR	0033
L0CNO	4442	PSTB	6664	TOCT	1200	XSKP	0041
L0CUR	4443	PTSTOR	3136	TRKCNT	0104	XTABLA	3257
L0DC	4446	QUES1	0232	TSTCHA	3515	XTABL8	3260
L0CBED	0134	R0ST	0671	TTYLMT	3721	XTEXT	0545
L0DER	0654	ROSTAT	4440	TNOCI	4451	XTICK	0027
L0DR1	0611	RECAL	4433	TYPE	4436	XTIME	0025
L0DTRK	4451	RECEIV	4434	UPARUW	3415	XTOCT	0051
L0TRK	2201	REDDSK	4432	UPONE	1215	XWAIT	0034
L0WAD	0102	REDOA	3215	WAIT	1527	XWRTRK	0031
M10	0077	REDTRK	1000	WASDSK	0242	XLDSK	0046
M13	0064	RENEW1	0335	WATMES	3431		
Ma	0076	RENEW2	0414	WRKBUF	2200		
MCNTR1	1641	RESTA	1425	WRTOCK	0521		
MES1	2045	RESTER	1443	WRTMCK	0600		
MES2	2066	RESTOR	1400	XAERN0	0026		
MES3	2117	RETRN1	0544	XAPT8	0024		
MES4	2126	SAMAGN	0224	XC0CKP	3641		
MES5	2135	SAVPC	1622	XC0CNT	3200		
MESA	3547	00CNT1	0106	XC0CNL	3623		
MESAC	4110	SKP	0740	XC0CCH	3663		
MESFL	4116	SETUP1	4026	XC0ERR	4007		
MESHAN	3746	SETUP2	3025	XC0ING	3435		
MESHQ	4113	SOFT	0136	XC0OCT	3600		
MESPAS	3053	STAER	1106	XC0PAS	3000		
MESPC	4105	STAQK	1041	XC0PAU	3137		
MOVE	1623	STCNT1	0107	XC0PNT	3103		
HQA	7501	STCNT2	0110	XC0PSW	3436		
HQL	7421	STCNT3	0111	XC0SW	3062		
HQSAVE	4123	STRAUT	0513	XC0TTY	3072		
HULOSK	1153	STREG	0122	XC0TYP	3677		
NEXCHK	0347	SWITCH	0057	XCHANG	0030		
NEXFRM	0276	TABLA	3201	XCLUN	0045		
NOSET	3042	TABL8	3271	XCRFP	0052		
NOTOSK	0244	TCMKT	1113	XDOLPT	3712		
NOTEX	0536	TCNTR1	0112	XDOSW	3317		
NTCLAS	4063	TCNTR2	0113	XEND	0135		
NTGD	0474	TCNTR3	0114	XERRU	0037		
OCTEL	4450	TCNTR4	0115	XFROCT	0050		
PASCNT	3050	TCNTR5	0116	XHITHK	0034		
PCLF	6662	TEXAD	1660	XKLBUF	0033		
PCNTR1	0547	TEXCA	1656	XLDAU	0044		
PCNTR2	0550	TEXCM	1650	XLCA	0043		
PCNTR3	0551	TEXDA	1654	XLDCM	0042		
PCOUNT	0161	TEXOT	1662	XLDSK	0733		
PNSAVE	4121	TEXEND	2021	XLDRK	0053		
PNTBUF	3720	TEXEX	1646	XMOVt	0133		
PRINT	1312	TEXGD	1644	XPRINT	0036		
PRN	1252	TEXPC	1642	XPRN	0047		

ERRURS DETECTED: 0
LINKS GENERATED: 132
RUN-TIME: 4 SECONDS
3K CORE USED