RXV21/RX02

FORMATTER PROGRAM

CZRXEA0

AH-E623A-MC

FICHE 1 OF 1

DEC 1978

digital

MADE IN USA

```
                        ;<PRUCHA>RX2FMT.P11.7, 22-AUG-78 13:28:06, EDIT BY SOUSA
                        .REM    &
```

IDENTIFICATION

PRODUCT CODE: AC-E662A-MC

PRODUCT NAME: CZRXEA0 RX02 FMTR PROG

DATE CREATED: 25 JUL 78

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: L.S. PRUCHA

TABLE OF CONTENTS

1.0 ABSTRACT

THIS PROGRAM IS INTENDED TO FORMAT A FLOPPY DISK TO EITHER SINGLE OR DOUBLE DENSITY ON A DRIVE CAPABLE OF SETTING DENSITY ON A FLOPPY DISK. IT WILL RUN UNDER APT, BUT WILL RUN IN A USER MODE WITH SEVERAL SWITCHABLE OPTIONS TO ENABLE A TESTING AND DEBUGGING.

2.0 REQUIREMENTS

2.1 HARDWARE

1. ANY PDP-11 PROCESSOR
2. RX02,XX SUBSYSTEM

2.2 STORAGE

THIS PROGRAM REQUIRES AT LEAST 4K WORDS OF CORE. IF LOADING VIA XXDP MORE STORAGE WILL BE NEEDED FOR THE XXDP MONITOR.

3.0 LOADING PROCEDURE

USE OF STANDARD BINARY LOADING PROCEDURE OR DOWN LINE LOAD VIA APT.

4.0 STARTING PROCEDURE

1. USER MODE - START AT LOCATION 200(8) TO INITIALIZE REGISTERS.
   - START AT LOCATION 210(8) TO RESTART TEST.
   - START AT LOCATION 220(8) TO ENTER DEBUG MODE.
2. APT MODE - ALWAYS START AT LOCATION 200(8).

5.0 SOFTWARE SWITCH SETTINGS

1. USER MODE - THE SWITCH REGISTER IS SETUP FOR THE USER BY THE PROGRAM VIA USER DIALOG.

SW15: 1 = HALT ON ERROR
0 = CONTINUE

SW14: 1 = EXTENDED ERROR REPORTS
0 = NORMAL ERROR REPORTS

SW13: 1 = INHIBIT ERROR REPORTS
0 = PRINT ERRORS

SW12: 1 = BUS INITIALIZE ON ERROR IF LOOPING
0 = NO BUS INITIALIZE ON ERROR

SW09: 1 = LOOP ON ERROR
0 = CONTINUE

--> THESE SWITCH SETTINGS ARE AVAILABLE, BUT ARE INTENDED FOR HARDWARE DEBUG ONLY.

****** SOFTWARE SWITCH REGISTER ******

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
SWR	HOE	EXT ER	IER	BUS INT			LOE	DBL DEN	CRC ALL						SY1 SEL	SY0 SEL

THESE BITS ARE SET BY PROGRAM INTERFACE TO USER - THEY ARE SHOWN HERE TO AID IN PROGRAM MAINTENANCE.

2. APT MODE - THE APT SWITCH REGISTER '$SWREG' MUST BE SET UP AS FOLLOWS: SY0SEL=1 (BIT0) AND DBLDEN=1 (BIT8) FOR SETTING TO DOUBLE DENSITY. IF CRC CHECK ON ALL TRACKS IS DESIRED THEN SET BIT CRCALL=1 (BIT7) SEE SWITCH REGISTER ABOVE.

6.0 OPERATION

THE PROGRAM OPERATION IN ''USER MODE'' IS SIMPLE. AFTER INITIAL START THE PROGRAM WILL ASK THE OPERATOR IF HELP IS WANTED AND WILL TYPE OUT A SHORT DESCRIPTION OF THE PROGRAM IF ANSWERED YES. THE PROGRAM WILL ALSO ASK IF A FULL DISKETTE SCAN IS WANTED TO VERIFY CRC. IF ADDRESS MODIFICATION IS SELECTED THEN THE PROGRAM WILL THEN ASK THE OPERATOR TO ENTER THE BUS ADDRESS AND VECTOR ADDRESS.

6.1 CONDITIONS OF OPERATION

THE PROGRAM EXPECTS TO RUN ON AN RX02 INTERFACE AND RX02 DRIVE SUBSYSTEM THAT HAVE PASSED ALL DIAGNOSTIC TESTS. ERRORS ARE REPORTED, BUT PRINTOUTS ARE NOT EXTENSIVE ENOUGH TO DIAGNOSE HARDWARE FAILURES.

7.0 PROGRAM DESCRIPTION

THE PROGRAM IS ORGANIZED AS FOLLOWS:

```
    START
    DETERMINE IF LSI PROCESSOR
    IF NOT APT MODE, CALL OPERATOR INTERFACE
    SET UP REGISTERS
    CALL SYSTEM SCHEDULER
    IF NOT APT MODE-CALL OUTPUT SYSTEM DONE (GET FURTHER OPERATOR INSTRUCTIONS)
    PRINT END OF PASS
    INCREMENT PASS COUNTER
    CHECK IF ACT MODE
    BUS RESET
    IF IDLE MODE SET THEN IDLE IN LOOP
    JUMP BACK TO ''CALL SYSTEM SCHEDULER''
```

```
                                7.1     FLOW
                                        ----
                                        BEGINROUTINE (MOD PROGRAM START + CONTROL)
                                        :  INITIALIZE STACK POINTER
                                        :  IF R0 IS FALSE
                                        :  :  THEN
                                        :  :    PRINT PROGRAM IDENTIFICATION MSG
                                        :  :    IF APT_MODE IS TRUE
                                        :  :    :  THEN
                                        :  :    :    GET APT SWITCH REGISTER
                                        :  :    :    IF APT_PROCESSOR_Q-BUS TRUE
                                        :  :    :    :  THEN
                                        :  :    :    :    SET LSI PROC PRI
                                        :  :    :    :  ELSE
                                        :  :    :    :    SET PDP11 PROC PRI
                                        :  :    :    ENDIF
                                        :  :    :    GET APT VECTOR
                                        :  :    :    SET SYSTEM VECTOR
                                        :  :    :    GET APT BASE ADDRESS OF UNIT UNDER TEST
                                        :  :    :    CALL ADDRESSING TEST
                                        :  :    :  ELSE
                                        :  :    :    SETUP BUS TRAP FOR LSI PROC TEST
                                        :  :    :    SET PROCESSOR STATUS WORD
                                        :  :    :    IF LSI_PROCESSOR_STATUS_WORD_TRAP TRUE
                                        :  :    :    :  THEN
                                        :  :    :    :    SET LSI PROCESSOR WORD STATUS
                                        :  :    :    :    SET LSI PROCESSOR FLAG
                                        :  :    :    ENDIF
                                        :  :    :    CALL OPERATOR INTERFACE (MOD 1.0)
                                        :  :    :    SET VECTOR INTERRUPT ADDRESS
                                        :  :    ENDIF
                                        :  :    SETUP DENSITY PER SWITCH REGISTER
                                        :  :    CLEAR TABLES + FLAGS
                                        :  :    CALL MOD-TEST ADDRESS
                                        :  ENDIF
                                        :  CLEAR PASS COUNTER
                                        :  BGNDO
                                        :  :  CALL SYSTEM SCHEDULER (MOD 2.0)
                                        :  :  IF APT_MODE FALSE
                                        :  :  :  THEN
                                        :  :  :    CALL OUTPUT DONE (MOD 3.0)
                                        :  :  ENDIF
                                        :  :  SET UP/PRINT END OF PASS + PASS COUNT
                                        :  :  INCREMENT PASS COUNTER
                                        :  :  IF ACT_MODE TRUE
                                        :  :  :  THEN
                                        :  :  :    RESET
                                        :  :  :    CALL ACT HOOKS
                                        :  :  ENDIF
                                        :  :  IF SYSTEM_FLAG=IDLE
                                        :  :  :  THEN
                                        :  :  :    BEGINDO
                                        :  :  :    ENDDO
                                        :  :  ENDIF
                                        :  ENDDO
                                        ENDROUTINE
```

7.2 PROGRAM FUNCTIONAL DESCRIPTIONS

THE PROGRAM WILL ACCOMPLISH THE FOLLOWING FUNCTIONAL ELEMENTS:

1. PROGRAM INITIALIZE SELECTED SYSTEMS + CHECK STATUS

2. READ MAINTENANCE STATUS ON SELECTED DRIVES + CHECK STATUS

3. SET DENSITY ON ALL SELECTED DRIVES + CHECK STATUS

4. CHECK CRC ON TRACKS #76 AND #0 OR ALL TRACKS, IF SELECTED BY USER. REPORT STATUS, IF ANY ERRORS.

5. REPORT WHEN DONE.

8.0 DEVICE REGISTERS

CODE		FUNCTION
0	=	FILL BUFFER
1	=	EMPTY BUFFER
2	=	WRITE SECTOR
3	=	READ SECTOR
4	=	SET DENSITY (RX02) ** BE CAREFUL TAKES 15 SECONDS **
5	=	READ STATUS ** MAINTENANCE MODE **
6	=	WRITE SECTOR WITH DELISTED DATA
7	=	READ ERROR CODE

***** RX02 REGISTER BITS *****

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
RXCS:	ERR	INT	XM	XM	RX2			SID	DEN	TR	IE	DON	DRV	FUN	FUN	FUN	GO
WC:	WORD COUNT																
BA:	BASE ADDRESS																
RXES:	X	X	X	X	NXM	WC OVF	SID #1	DRV #1	DRV RDY	DEL DAT	DSK DEN	DEN ERR	AC LOW	INT DON	SID RDY	CRC ERR	
RXDB:	DATA BUFFER																

9.0 LISTING INDEX

```
                              9.1     LISTING
                                      -------

                              &.
                              .TITLE  CZRXEA0 RX02 FMTR PROG
                              ;RX02-APT
                              ;14-MARCH-78
                              ;L. PRUCHA
                              .ENABLE ABS,AMA
                              .LIST   ME
                              .NLIST  BEX,CND,MC,MD

                              ;***** PROGRAM EQUIVALENTS *****

        000040                        DNBIT   =       40              ;DONE BIT--------<CSR>
        000200                        TRBIT   =       200             ;TR BIT----------<CSR>
        000400                        DENBIT  =       400             ;DENSITY BIT-----<CSR>
        001000                        SIDE1   =       1000            ;SIDE #1 BIT-----<ESR> & <CSR>
        000200                        DRVRDY  =       200             ;DRIVE READY BIT-<ESR>
        000010                        ACLOW   =       10              ;AC LOW BIT------<ESR>
        000004                        INITDN  =       4               ;INITIALIZE DONE BIT-<ESR>
        000002                        SIDRDY  =       2               ;SIDE READY BIT--<ESR>
        000400                        DRIVE1  =       400             ;DRIVE #1 BIT----<ESR>
        000020                        DENERR  =       BIT4            ;DENSITY ERROR---<ESR>
        000040                        DRVDEN  =       BIT5            ;DRIVE DENSITY---<ESR>
        000001                        CRCERR  =       BIT0            ;CRC ERROR BIT---<ESR>
        100000                        ERRBIT  =       100000          ;ERROR BIT-------<CSR>
        000020                        DRV1    =       20              ;DRIVE 1---------<CSR>
        040000                        RXINIT  =       40000           ;RXINIT BIT------<CSR>
        004000                        RX2BIT  =       4000            ;RX02 BIT--------<CSR>
        000001                        APTENV  =       1               ;APT ENVIRONMENT TEST WORD
        000040                        APTCSP  =       40              ;APT CONSOLE SUPRESSION WORD
        000200                        CRCALL  =       SW7             ;CRC ALL TRACKS & SECTORS FLAG
        000006                        RDSEC   =       6               ;READ SECTOR COMMAND CODE (LEFT SHIFTED)
        000012                        RDMNST  =       12              ;READ MAINT STATUS COMMAND CODE (LS)
        000010                        SETDEN  =       10              ;SET DENSITY COMMAND CODE (LS)
        000400                        DBLDEN  =       BIT8            ;DOUBLE DENSITY BIT
        000001                        X=1
        000001                        N=1
```

```
                                .SBTTL  BASIC DEFINITIONS
                                ;*INITIAL ADDRESS OF THE STACK POINTER *** 1000 ***
001000                          STACK=  1000
                                .EQUIV  EMT,ERROR       ;;BASIC DEFINITION OF ERROR CALL
                                .EQUIV  IOT,SCOPE       ;;BASIC DEFINITION OF SCOPE CALL

                                ;*MISCELLANEOUS DEFINITIONS
000011                          HT=     11              ;;CODE FOR HORIZONTAL TAB
000012                          LF=     12              ;;CODE FOR LINE FEED
000015                          CR=     15              ;;CODE FOR CARRIAGE RETURN
000200                          CRLF=   200             ;;CODE FOR CARRIAGE RETURN-LINE FEED
177776                          PS=     177776          ;;PROCESSOR STATUS WORD
                                .EQUIV  PS,PSW
177774                          STKLMT= 177774          ;;STACK LIMIT REGISTER
177772                          PIRQ=   177772          ;;PROGRAM INTERRUPT REQUEST REGISTER
177570                          DSWR=   177570          ;;HARDWARE SWITCH REGISTER
177570                          DDISP=  177570          ;;HARDWARE DISPLAY REGISTER

                                ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                          R0=     %0              ;;GENERAL REGISTER
000001                          R1=     %1              ;;GENERAL REGISTER
000002                          R2=     %2              ;;GENERAL REGISTER
000003                          R3=     %3              ;;GENERAL REGISTER
000004                          R4=     %4              ;;GENERAL REGISTER
000005                          R5=     %5              ;;GENERAL REGISTER
000006                          R6=     %6              ;;GENERAL REGISTER
000007                          R7=     %7              ;;GENERAL REGISTER
000006                          SP=     %6              ;;STACK POINTER
000007                          PC=     %7              ;;PROGRAM COUNTER

                                ;*PRIORITY LEVEL DEFINITIONS
000000                          PR0=    0               ;;PRIORITY LEVEL 0
000040                          PR1=    40              ;;PRIORITY LEVEL 1
000100                          PR2=    100             ;;PRIORITY LEVEL 2
000140                          PR3=    140             ;;PRIORITY LEVEL 3
000200                          PR4=    200             ;;PRIORITY LEVEL 4
000240                          PR5=    240             ;;PRIORITY LEVEL 5
000300                          PR6=    300             ;;PRIORITY LEVEL 6
000340                          PR7=    340             ;;PRIORITY LEVEL 7

                                ;*''SWITCH REGISTER'' SWITCH DEFINITIONS
100000                          SW15=   100000
040000                          SW14=   40000
020000                          SW13=   20000
010000                          SW12=   10000
004000                          SW11=   4000
002000                          SW10=   2000
001000                          SW09=   1000
000400                          SW08=   400
000200                          SW07=   200
000100                          SW06=   100
000040                          SW05=   40
000020                          SW04=   20
000010                          SW03=   10
```

```
 (1)        000004          SW02=   4
 (1)        000002          SW01=   2
 (1)        000001          SW00=   1
 (1)                        .EQUIV  SW09,SW9
 (1)                        .EQUIV  SW08,SW8
 (1)                        .EQUIV  SW07,SW7
 (1)                        .EQUIV  SW06,SW6
 (1)                        .EQUIV  SW05,SW5
 (1)                        .EQUIV  SW04,SW4
 (1)                        .EQUIV  SW03,SW3
 (1)                        .EQUIV  SW02,SW2
 (1)                        .EQUIV  SW01,SW1
 (1)                        .EQUIV  SW00,SW0
 (1)
 (1)                        ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
 (1)        100000          BIT15=  100000
 (1)        040000          BIT14=  40000
 (1)        020000          BIT13=  20000
 (1)        010000          BIT12=  10000
 (1)        004000          BIT11=  4000
 (1)        002000          BIT10=  2000
 (1)        001000          BIT09=  1000
 (1)        000400          BIT08=  400
 (1)        000200          BIT07=  200
 (1)        000100          BIT06=  100
 (1)        000040          BIT05=  40
 (1)        000020          BIT04=  20
 (1)        000010          BIT03=  10
 (1)        000004          BIT02=  4
 (1)        000002          BIT01=  2
 (1)        000001          BIT00=  1
 (1)                        .EQUIV  BIT09,BIT9
 (1)                        .EQUIV  BIT08,BIT8
 (1)                        .EQUIV  BIT07,BIT7
 (1)                        .EQUIV  BIT06,BIT6
 (1)                        .EQUIV  BIT05,BIT5
 (1)                        .EQUIV  BIT04,BIT4
 (1)                        .EQUIV  BIT03,BIT3
 (1)                        .EQUIV  BIT02,BIT2
 (1)                        .EQUIV  BIT01,BIT1
 (1)                        .EQUIV  BIT00,BIT0
 (1)
 (1)                        ;*BASIC "CPU" TRAP VECTOR ADDRESSES
 (1)        000004          ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
 (1)        000010          RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
 (1)        000014          TBITVEC=14              ;;"T" BIT
 (1)        000014          TRTVEC= 14              ;;TRACE TRAP
 (1)        000014          BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
 (1)        000020          IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
 (1)        000024          PWRVEC= 24              ;;POWER FAIL
 (1)        000030          EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
 (1)        000034          TRAPVEC=34              ;;"TRAP" TRAP
 (1)        000060          TKVEC=  60              ;;TTY KEYBOARD VECTOR
 (1)        000064          TPVEC=  64              ;;TTY PRINTER VECTOR
 (1)        000240          PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR
 6043
```

```
                                  ;***** TRAP CATCHERS *****

              000000                      .=0
      000000  000000                      .WORD   0
      000002  000000                      .WORD   0               ;UNASSIGNED TRAP
      000004  000000                      .WORD   0
      000006  000000                      .WORD   0               ;TIME OUT, BUS TRAP
      000010  000000                      .WORD   0
      000012  000000                      .WORD   0               ;RESERVED INSTRUCTION
      000014  000000                      .WORD   0
      000016  000000                      .WORD   0               ;TRACE TRAP
      000020  000000                      .WORD   0
      000022  000000                      .WORD   0
      000024  000000                      .WORD   0
      000026  000000                      .WORD   0
      000030  000000                      .WORD   0
      000032  000000                      .WORD   0
      000034  000000                      .WORD   0
      000036  000000                      .WORD   0
      000040  000000                      .WORD   0
      000042  000000                      .WORD   0
      000044  000000                      .WORD   0



                                  .SBTTL  ACT11 HOOKS
 (1)
 (2)                              ;;*****************************************************************
 (1)                              ;HOOKS REQUIRED BY ACT11
 (1)          000046                      $SVPC=.                 ;SAVE PC
 (1)          000046                      .=46
 (1)  000046  001422                      $ENDAD                  ;;1)SET LOC.46 TO ADDRESS OF $ENDAD
 (1)          000052                      .=52
 (1)  000052  000000                      .WORD   0               ;;2)SET LOC.52 TO ZERO
 (1)          000046                      .=$SVPC                 ;; RESTORE PC
```

```
                                    ;***** TTY INTERRUPT VECTOR *****

         000060                           .=60
000060   012420                           TTINT                         ;TTY INTERRUPT HEADER ADDRESS
000062   000000                           0

                                    ;***** START ADDRESS *****

         000200                           .=200
000200   005000                           CLR      R0
000202   000137   001000                  JMP      START                ;PROGRAM START

         000210                           .=210
000210   012700   000001                  MOV      #1,R0                ;SET NO HEADER FLAG
000214   000137   001000                  JMP      START

         000220                           .=220
000220   005237   000516                  INC      DBGFLG               ;SET DEBUG FLAG
000224   005000                           CLR      R0                   ;SET SO HEADER WILL PRINT
000226   000137   001000                  JMP      START

                                    ;***** RX02 INTERRUPT VECTORS *****

         000264                           .=264
000264   012404                           RXINT0                        ;RX02 INTERRUPT HANDLER #0 ADDRESS
000266   000340                           340

         000270                           .=270
000270   012412                           RXINT1                        ;RX02 INTERRUPT HANDLER #1 ADDRESS
000272   000340                           340

         000300                           .=300

                                    ;***** CONSTANTS *****

000300   177560                     TKS:    177560                      ;TTY READER STATUS
000302   177562                     TKB:    177562                      ;TTY READ BUFFER
000304   177564                     TPS:    177564                      ;TTY PUNCH STATUS
000306   177566                     TPB:    177566                      ;TTY PUNCH BUFFER


                                    ;***** INITIAL CONSTANTS *****

000310   000264                     VECT0:  264                         ;INTERRUPT VECTOR SYS #0
000312   000270                     VECT1:  270                         ;INTERRUPT VECTOR SYS #1
000314   177170                     REGS0:  177170                      ;STARTING REGISTER ADDRESS SYS #0
000316   177200                     REGS1:  177200                      ;STARTING REGISTER ADDRESS SYS #1
000320   000004                     BTRP:   4                           ;BUS TRAP ADDRESS
000322   000006                     BTRP2:  6                           ;BUS TRAP RRIORITY LEVEL
000324   000401                     SWREG:  401                         ;SOFTWARE SWITCH REGISTER - PRESET-> DEN=DBL,SYS0
```

```
 6123
 6124            000400                          .=400
 6125            000401                  ASWREG=401
 6126            000264                  AVECT1=264                      ;APT VECTOR 264
 6127            177170                  ABASE=177170                    ;APT UNIBUS ADDRESS
 6128                                    .SBTTL  APT MAILBOX-ETABLE
  (1)
  (2)                                   ;;**************************************************************************
  (1)                                   .EVEN
  (1)   000400                          $MAIL:                          ;;APT MAILBOX
  (1)   000400  000000                  $MSGTY: .WORD   AMSGTY   ;;MESSAGE TYPE CODE
  (1)   000402  000000                  $FATAL: .WORD   AFATAL   ;;FATAL ERROR NUMBER
  (1)   000404  000000                  $TESTN: .WORD   ATESTN   ;;TEST NUMBER
  (1)   000406  000000                  $PASS:  .WORD   APASS    ;;PASS COUNT
  (1)   000410  000000                  $DEVCT: .WORD   ADEVCT   ;;DEVICE COUNT
  (1)   000412  000000                  $UNIT:  .WORD   AUNIT    ;;I/O UNIT NUMBER
  (1)   000414  000000                  $MSGAD: .WORD   AMSGAD   ;;MESSAGE ADDRESS
  (1)   000416  000000                  $MSGLG: .WORD   AMSGLG   ;;MESSAGE LENGTH
  (1)   000420                          $ETABLE:                        ;;APT ENVIRONMENT TABLE
  (1)   000420     000                  $ENV:   .BYTE   AENV     ;;ENVIRONMENT BYTE
  (1)   000421     000                  $ENVM:  .BYTE   AENVM    ;;ENVIRONMENT MODE BITS
  (1)   000422  000401                  $SWREG: .WORD   ASWREG   ;;APT SWITCH REGISTER
  (1)   000424  000000                  $USWR:  .WORD   AUSWR    ;;USER SWITCHES
  (1)   000426  000000                  $CPUOP: .WORD   ACPUOP   ;;CPU TYPE,OPTIONS
  (1)                                   ;*                              BITS 15-11=CPU TYPE
  (1)                                   ;*                                      11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
  (1)                                   ;*                                      11/70=06,PDQ=07,Q=10
  (1)                                   ;*                              BIT 10=REAL TIME CLOCK
  (1)                                   ;*                              BIT  9=FLOATING POINT PROCESSOR
  (1)                                   ;*                              BIT  8=MEMORY MANAGEMENT
  (1)   000430     000                  $MAMS1: .BYTE   AMAMS1   ;;HIGH ADDRESS,M.S. BYTE
  (1)   000431     000                  $MTYP1: .BYTE   AMTYP1   ;;MEM. TYPE,BLK#1
  (1)                                   ;*                              MEM.TYPE BYTE   --  (HIGH BYTE)
  (1)                                   ;*                                      900 NSEC CORE=001
  (1)                                   ;*                                      300 NSEC BIPOLAR=002
  (1)                                   ;*                                      500 NSEC MOS=003
  (1)   000432  000000                  $MADR1: .WORD   AMADR1   ;;HIGH ADDRESS,BLK#1
  (1)                                   ;*                                MEM.LAST ADDR.=3 BYTES,THIS WORD AND LOW OF ''TYPE'' ABOVE
  (1)   000434     000                  $MAMS2: .BYTE   AMAMS2   ;;HIGH ADDRESS,M.S. BYTE
  (1)   000435     000                  $MTYP2: .BYTE   AMTYP2   ;;MEM.TYPE,BLK#2
  (1)   000436  000000                  $MADR2: .WORD   AMADR2   ;;MEM.LAST ADDRESS,BLK#2
  (1)   000440     000                  $MAMS3: .BYTE   AMAMS3   ;;HIGH ADDRESS,M.S.BYTE
  (1)   000441     000                  $MTYP3: .BYTE   AMTYP3   ;;MEM.TYPE,BLK#3
  (1)   000442  000000                  $MADR3: .WORD   AMADR3   ;;MEM.LAST ADDRESS,BLK#3
  (1)   000444     000                  $MAMS4: .BYTE   AMAMS4   ;;HIGH ADDRESS,M.S.BYTE
  (1)   000445     000                  $MTYP4: .BYTE   AMTYP4   ;;MEM.TYPE,BLK#4
  (1)   000446  000000                  $MADR4: .WORD   AMADR4   ;;MEM.LAST ADDRESS,BLK#4
  (1)   000450  000264                  $VECT1: .WORD   AVECT1   ;;INTERRUPT VECTOR#1,BUS PRIORITY#1
  (1)   000452  000000                  $VECT2: .WORD   AVECT2   ;;INTERRUPT VECTOR#2BUS PRIORITY#2
  (1)   000454  177170                  $BASE:  .WORD   ABASE    ;;BASE ADDRESS OF EQUIPMENT UNDER TEST
  (1)   000456                          $ETEND:
  (1)                                   .MEXIT
```

```
 6130                                  .SBTTL  APT PARAMETER BLOCK
  (1)
  (2)                                  ;;**********************************************************************
  (1)                                  ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
  (2)                                  ;;**********************************************************************
  (1)           000456                         .$X=.    ;;SAVE CURRENT LOCATION
  (1)           000024                         .=24     ;;SET POWER FAIL TO POINT TO START OF PROGRAM
  (1)  000024   000200                         200      ;;FOR APT START UP
  (1)           000044                         .=44     ;;POINT TO APT INDIRECT ADDRESS PNTR.
  (1)  000044   000456                         $APTHDR  ;;POINT TO APT HEADER BLOCK
  (1)           000456                         .=.$X    ;;RESET LOCATION COUNTER
  (2)                                  ;;**********************************************************************
  (1)                                  ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
  (1)                                  ;INTERFACE SPEC.
  (1)
  (1)  000456                          $APTHD:
  (1)  000456   000000                 $HIBTS: .WORD   0        ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
  (1)  000460   000400                 $MBADR: .WORD   $MAIL    ;;ADDRESS OF APT MAILBOX (BITS 0-15)
  (1)  000462   000120                 $TSTM:  .WORD   120      ;;RUN TIM OF LONGEST TEST
  (1)  000464   000120                 $PASTM: .WORD   120      ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
  (1)  000466   000120                 $UNITM: .WORD   120      ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
  (1)  000470   000027                         .WORD   $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
```

```
 6132                                 ;***** FLAGS AND COUNTERS *****
 6133
 6134  000472  177170                 RXCSAD: 177170                  ;RX02 CSR ADDRESS
 6135  000474  177172                 RXDBAD: 177172                  ;RX02 DBR ADDRESS
 6136  000476  000264                 RXVECT: 264                     ;RX02 VECTOR ADDRESS
 6137  000500  000001                 SECTR:  1                       ;SECTOR ADDRESS
 6138  000502  000000                 TRACK:  0                       ;TRACK ADDRESS
 6139  000504  000324                 SWR:    #SWREG                  ;SWITCH REG ADDRESS
 6140  000506  000000                 DENS:   0                       ;DENSITY WORD
 6141  000510  000000                 REGS:   0                       ;REGISTER ADDRESS BUFFER
 6142  000512  000000                 VECT:   0                       ;VECTOR ADDRESS BUFFER
 6143  000514  000000                 INITPG: 0                       ;PROGRAM INITIALIZE FLAG
 6144  000516  000000                 DBGFLG: 0                       ;DEBUG FLAG
 6145  000520  000000                 LSIFLG: 0                       ;LSI OR SWITCHLESS PROCESSOR FLAG
 6146  000522  000000                 PCNTR:  0                       ;PASS COUNTER
 6147  000524  000000                 TOB:    0                       ;TYPE OUTPUT BUFFER
 6148  000526  000000                 TIB:    0                       ;TYPE INPUT BUFFER
 6149  000530  000000                 ANSWER: 0                       ;ANSWER TO QUESTION BUFFER
 6150  000532  000000                 TTWAIT: 0                       ;TTY WAIT FLAG
 6151  000534  000000                 CSR:    0                       ;RXCS TEMP STORE
 6152  000536  000000                 ESR:    0                       ;RXES TEMP STORE
 6153  000540  000000                 WDOT:   0                       ;WORD FOR OUTPUT
 6154  000542  000000                 CMD:    0                       ;RX COMMAND
 6155  000544  000000                 CSRMSK: 0                       ;RXCS MASK WORD
 6156  000546  000000                 CSRCMP: 0                       ;RXCS COMPARE WORD
 6157  000550  000000                 ESRMSK: 0                       ;RXES MASK WORD
 6158  000552  000000                 ESRCMP: 0                       ;RXES COMPARE WORD
 6159  000554  000000                 TSTERR: 0                       ;TEST ERROR FLAG
 6160  000556  000000                 S0WMLT: 0                       ;WATCHDOG MULTIPLER SYSTEM#0
 6161  000560  000000                 S1WMLT: 0                       ;WATCHDOG MULTIPLER SYSTEM#1
 6162  000562  000000                 TEMP1:  0                       ;
 6163  000564  000000                 TEMP2:  0                       ;
 6164  000566  000000                 TEMP3:  0                       ;
 6165  000570  000000                 BTRPFL: 0                       ;BUS TRAP FLAG
 6166  000572  000000                 SYSMSG: 0                       ;SYSTEM MESSAGE BUFFER
 6167  000574  000000                 SYSFLG: 0                       ;SYSTEM FLAGS
 6168  000576  000000                 STFLG:  0                       ;  - START PROGRAM FLAG
 6169  000600  000000                 TMSGAD: 0                       ;MESSAGE ADDRESS-HEADER
 6170  000602  000000                 HDRFLG: 0                       ;HEADER FLAG
 6171  000604  000000                 ERRTYP: 0                       ;ERROR TYPE
 6172  000606  000000                 SYSERR: 0                       ;SYSTEM ERROR
 6173  000610  000000                 SUC:    0                       ;SYSTEM UNDER CONTROL
 6174  000612  000000                 S0AV:   0                       ;SYSTEM#0 AVAILABLE
 6175  000614  000000                 S1AV:   0                       ;SYSTEM#1 AVAILABLE
 6176  000616  000000                 S0RDY:  0                       ;SYSTEM#0 READY
 6177  000620  000000                 S1RDY:  0                       ;SYSTEM#1 READY
 6178  000622  000000                 S0CMD:  0                       ;SYSTEM#0 COMMAND WORD
 6179  000624  000000                 S1CMD:  0                       ;SYSTEM#1 COMMAND WORD
 6180  000626  000000                 S0DN:   0                       ;SYSTEM#0 DONE FLAG
 6181  000630  000000                 S1DN:   0                       ;SYSTEM#1 DONE FLAG
 6182  000632  000000                 TTITFG: 0                       ;TTY INTERRUPT FLAG
 6183  000634  000000                 ENDFTB: 0                       ;*** DO NOT REMOVE THIS LABEL ***
```

```
                    001000                          .=1000
                                                    .SBTTL  MODULE 0.0 - PROGRAM START AND CONTROL
                                                    ;-----------------------------------------------------------------

001000  012706  001000          START:  MOV     #1000,SP        ;SET STACK POINTER
001004  005037  000516                  CLR     DBGFLG          ;CLEAR DEBUG FLAG
001010  005037  000406                  CLR     $PASS           ;CLEAR APT PASS COUNTER
001014  005037  000570                  CLR     BTRPFL          ;CLEAR BUS TRAP FLAG
001020  005700                  IA00:   TST     R0              ;IF PRINT HEADER
001022  001115                          BNE     EA00            ;INDICATED, THEN
001024  012704  014040                  MOV     #MSG3,R4        ;SETUP TITLE MSG
001030  004737  013232                  JSR     PC,TTOUT        ;PRINT TITLE
001034  132737  000001  000420  IB00:   BITB    #APTENV,$ENV    ;IF APT MODE                    [APT]
001042  001441                          BEQ     LB00            ;TRUE, THEN                     [APT]
001044  122737  000100  000427  IC00:   CMPB    #100,$CPUOP+1   ;IF CPU IS                      [APT]
001052  001006                          BNE     LC00            ;LSI/Q-BUS, THEN                [APT]
001054  012737  000001  000520          MOV     #1,LSIFLG       ;SET LSI PROCESSOR FLAG         [APT]
001062  106427  000200                  MTPS    #200            ;SET LSI PRIORITY               [APT]
001066  000403                          BR      EC00            ;                               [APT]
001070  012777  000340  176700  LC00:   MOV     #PR7,@PSW       ;SET PDP-11 PRIORITY            [APT]
001076  012737  000422  000504  EC00:   MOV     #$SWREG,SWR     ;SET SOFTWARE SWITCH ADDRESS    [APT]
001104  113700  000450                  MOVB    $VECT1,R0       ;GET VECTOR FROM ETABLE         [APT]
001110  042700  177400                  BIC     #177400,R0      ;CLEAR TOP BYTE R0              [APT]
001114  012720  012404                  MOV     #RXINT0,(R0)+   ;LOAD INTERRUPT ADR IN VECTOR   [APT]
001120  012710  000340                  MOV     #340,(R0)       ;LOAD PRIORITY                  [APT]
001124  013737  000454  000472          MOV     $BASE,RXCSAD    ;SET REGISTER BASE FROM ETABLE  [APT]
001132  004737  002204                  CALL    ADTST           ;CALL ADDRESSING TST            [APT]
001136  013737  000472  000314          MOV     RXCSAD,REGS0    ;SET SYS#0 BUS ADDR             [APT]
001144  000444                          BR      EA00            ;BR TO END 'A'                  [APT]
001146  042777  000100  177124  LB00:   BIC     #BIT6,@TKS      ;DISABLE TTY KEYBOARD INTERRUPTS [USER]
001154  012777  001200  177136          MOV     #LSITRP,@BTRP   ;SET BUS TRAP FOR LSI TRAP      [USER]
001162  012777  000200  177132          MOV     #200,@BTRP2     ;SET UP PROCESSOR PRI-BUS TRAP  [USER]
001170  012777  000340  176600  ID00:   MOV     #340,@PSW       ;IF PROCESSOR IS                [USER]
001176  000405                          BR      ED00            ;AN LSI, THEN                   [USER]
001200  012737  000001  000520  LSITRP: MOV     #1,LSIFLG       ;SET LSI PROCESSOR FLAG         [USER]
001206  106427  000200                  MTPS    #200            ;SET LSI PROCESSOR PRIORITY     [USER]
001212  005077  177102          ED00:   CLR     @BTRP           ;RESET BUS TRAP                 [USER]
001216  005077  177100                  CLR     @BTRP2          ;RESET BUS TRAP                 [USER]
001222  004737  001456                  CALL    OPINT           ;CALL GET USER INPUT (MOD 1.0)  [USER]
001226  013705  000310                  MOV     VECT0,R5        ;GET SYS#0 VECTOR               [USER]
001232  012725  012404                  MOV     #RXINT0,(R5)+   ;SET SYSTEM#0 VECTOR            [USER]
001236  012715  000340                  MOV     #PR7,(R5)       ;SET PROCESSOR PRI              [USER]
001242  013705  000312                  MOV     VECT1,R5        ;GET SYS#1 VECTOR               [USER]
001246  012725  012412                  MOV     #RXINT1,(R5)+   ;SET SYSTEM#1 VECTOR            [USER]
001252  012715  000340                  MOV     #PR7,(R5)       ;SET PROCESSOR PRI              [USER]
001256  005037  000522          EA00:   CLR     PCNTR           ;CLEAR PASS COUNTER
001262  032777  000400  177214  IJ00:   BIT     #BIT8,@SWR      ;IF SWITCH REG HAS DOUBLE DENSITY BIT
001270  001404                          BEQ     LJ00            ;SET, THEN
001272  012737  000400  000506          MOV     #DBLDEN,DENS    ;SETUP DENSITY=DOUBLE
001300  000402                          BR      BE00            ;BR TO BEGIN DO 'E'
001302  005037  000506          LJ00:   CLR     DENS            ;SETUP DENSITY=SINGLE
001306  012702  000524          BE00:   MOV     #TOB,R2         ;SET BEGINNING OF TABLE
001312  012700  000634                  MOV     #ENDFTB,R0      ;SET END OF TABLE
001316  005022                  BF00:   CLR     (R2)+           ;CLEAR TABLE OF FLAGS + COUNTERS
001320  020200                          CMP     R2,R0           ;DO UNTIL
001322  103775                          BLO     BF00            ;END OF TABLES DONE
```

```
 6241  001324  005237  000514                    INC     INITPG          ;SET PROGRAM INITIALIZE FLAG
 6242  001330  004737  002514                    CALL    SYSCHD          ;GO FORMAT SCHEDULED SYSTEMS    (MOD 2.0)
 6243  001334  132737  000001  000420  IG00:     BITB    #APTENV,$ENV    ;IF APT MODE
 6244  001342  001002                            BNE     EG00            ;FALSE, THEN
 6245  001344  004737  007676                    CALL    OTSYDN          ;CALL OUTPUT SYSTEM DONE        (MOD 3.0)
 6246  001350  012704  014164          EG00:     MOV     #MSG6,R4        ;SET END OF PASS MSG
 6247  001354  004737  013232                    JSR     PC,TTOUT        ;PRINT END OF PASS
 6248  001360  005237  000532                    INC     TTWAIT          ;SET TTY WAIT FLAG
 6249  001364  013703  000522                    MOV     PCNTR,R3
 6250  001370  004737  013346                    JSR     PC,OCTP         ;PRINT PASS NUMBER
 6251  001374  005237  000522                    INC     PCNTR           ;BUMP PASS COUNTER
 6252  001400  013737  000522  000406            MOV     PCNTR,$PASS     ;BUMP APT PASS COUNTER
 6253  001406  013700  000042          IH00:     MOV     @#42,R0         ;IF ACT
 6254  001412  001407                            BEQ     II00            ;MODE, THEN
 6255  001414  005037  000042                    CLR     @#42            ;ABORT THIS PROGRAM
 6256  001420  000005                            RESET
 6257  001422  004710                  $ENDAD:   JSR     PC,(R0)         ;ACT HOOKS
 6258  001424  000240                            NOP                     ; *  DO NOT
 6259  001426  000240                            NOP                     ;       REMOVE
 6260  001430  000240                            NOP                     ;            THESE NOP'S *
 6261  001432  032737  000100  000574  II00:     BIT     #100,SYSFLG     ;IF IDLE MODE
 6262  001440  001401                            BEQ     EI00            ;SET, THEN
 6263  001442  000777                  1$:       BR      1$              ;IDLE LOOP
 6264  001444  000005                  EI00:     RESET                   ;
 6265  001446  005037  000532                    CLR     TTWAIT          ;CLEAR TTY WAIT FLAG
 6266  001452  000137  001306          END00:    JMP     BE00            ;DO NEXT PASS
 6267                                            ;-----------------------------------------------------------------
```

```
                                .SBTTL  MODULE 1.0 - OPERATOR INTERFACE

                                ;BEGINROUTINE (MODULE 1.0 - GET USER INPUT)
                                ;   SET SOFTWARE SWITCH REG ADDRESS
                                ;   SETUP & PRINT 'HELP?'
                                ;   CALL GET ANSWER
                                ;   IF ANSWER=YES                                                    [A]
                                ;   :   THEN
                                ;   :      SETUP & PRINT HELP FILE
                                ;   ENDIF
                                ;   SETUP & PRINT 'SET DISKETTE TO SINGLE DENSITY? (Y OR N)'
                                ;   CALL GET ANSWER
                                ;   IF ANSWER=YES                                                    [B]
                                ;   :   THEN
                                ;   :      SET DENS & SOFT SWITCH REG DENSITY BIT=SINGLE DENSITY
                                ;   :   ELSE
                                ;   :      SET DENS & SOFT SWITCH REG DENSITY BIT=DOUBLE DENSITY
                                ;   ENDIF
                                ;   SETUP & PRINT 'VERIFY DISKETTE CRC (ALL TRACKS)? (Y OR N)'
                                ;   CALL GET ANSWER
                                ;   IF ANSWER=YES                                                    [C]
                                ;   :   THEN
                                ;   :      SET SOFT SWITCH REG=CRC ALL BIT
                                ;   :   ELSE
                                ;   :      CLEAR SOFT SWITCH REG=CRC ALL BIT
                                ;   ENDIF
                                ;   SET <SUC>=0
                                ;   SETUP & PRINT 'FLOPPY SYS #0 ADDRESS CHANGE? (Y OR N)'
                                ;   CALL GET ANSWER
                                ;   IF ANSWER=YES                                                    [D]
                                ;   :   THEN
                                ;   :      SET YES ANSWER FLAG
                                ;   :   ELSE
                                ;   :      CLEAR YES ANSWER FLAG
                                ;   ENDIF
                                ;   CALL GET SYSTEM BUS ADDRESS
                                ;   CALL CHECK SYSTEM BUS ADDRESS (SEE IF BOOT SYSTEM)
                                ;   SET SOFT SWITCH REG=SYS #0 AVAILABLE
                                ;   SETUP & PRINT 'IS SECOND FLOPPY SYSTEM AVAILABLE? (Y OR N)'
                                ;   CALL GET ANSWER
                                ;   IF ANSWER=YES                                                    [E]
                                ;   :   THEN
                                ;   :      SET <SUC>=1
                                ;   :      SET YES ANSWER FLAG
                                ;   :      CALL GET SYSTEM BUS ADDRESS
                                ;   :      CALL CHECK SYSTEM BUS ADDRESS (SEE IF BOT MEDIA)
                                ;   :      SET SOFT SWITCH REG=SYS #1 AVAILABLE
                                ;   ENDIF
                                ;   IF DEBUG_FLAG=SET                                                [F]
                                ;   :   THEN
                                ;   :      CALL GET SOFT SWITCH REG
                                ;   ENDIF
                                ;ENDROUTINE
```

```
                                    ;------------------------------------------------------------------------
001456 012737 000324 000504  OPINT: MOV     #SWREG,SWR      ;SET SOFTSWITCH REG ADR
001464 012704 015726                MOV     #MSG43,R4       ;SETUP MSG 'HELP?''
001470 004737 013232                CALL    TTOUT           ;PRINT
001474 004737 012624                CALL    GETANS          ;CALL GET OPERATOR ANSWER
001500 122737 000131 000530  IA10:  CMPB    #'Y,ANSWER      ;IF ANSWER
001506 001004                       BNE     EA10            ;IS YES, THEN
001510 012704 015300                MOV     #MSG39,R4       ;SETUP MSG --> HELP FILE
001514 004737 013232                CALL    TTOUT           ;PRINT
001520 012704 016045         EA10:  MOV     #MSG45,R4       ;SETUP MSG ''SET DISKETTE TO SINGLE DENSITY: (Y OR N)?''
001524 004737 013232                CALL    TTOUT           ;PRINT
001530 004737 012624                CALL    GETANS          ;CALL GET OPERATOR ANSWER
001534 122737 000131 000530  IB10:  CMPB    #'Y,ANSWER      ;IF ANSWER
001542 001004                       BNE     LB10            ;IS YES, (SET SINGLE DENSITY), THEN
001544 042777 000400 176732         BIC     #DENBIT,@SWR    ;CLEAR DENSITY BIT IN SOFT SWITCH REG
001552 000403                       BR      EB10            ;BR TO END 'B'
001554 052777 000400 176722  LB10:  BIS     #DENBIT,@SWR    ;SET DENSITY BIT IN SOFT SWITCH REG
001562 012704 016125         EB10:  MOV     #MSG46,R4       ;SETUP MSG 'VERIFY DISKETTE CRC (ALL TRACKS): (Y OR N)?''
001566 004737 013232                CALL    TTOUT           ;PRINT
001572 004737 012624                CALL    GETANS          ;CALL GET OPERATOR ANSWER
001576 122737 000131 000530  IC10:  CMPB    #'Y,ANSWER      ;IF ANSWER
001604 001004                       BNE     LC10            ;IS YES, THEN
001606 052777 000200 176670         BIS     #CRCALL,@SWR    ;SET CRC ALL FLAG IN SOFT SWITCH REG
001614 000403                       BR      EC10            ;BR TO END 'C'
001616 042777 000200 176660  LC10:  BIC     #CRCALL,@SWR    ;CLEAR CRC ALL FLAG IN SOFT SWITCH REG
001624 005037 000610         EC10:  CLR     SUC             ;SET <SUC> = 0
001630 012704 016206                MOV     #MSG47,R4       ;SETUP MSG ''FLOPPY SYS 0 ADDRESS MODIFICATION (Y OR N)?''
001634 004737 013232                CALL    TTOUT           ;PRINT
001640 004737 012624                CALL    GETANS          ;CALL GET OPERATOR ANSWER
001644 122737 000131 000530  ID10:  CMPB    #'Y,ANSWER      ;IF ANSWER
001652 001002                       BNE     LD10            ;IS YES, THEN
001654 005202                       INC     R2              ;SET YES ANSWER FLAG
001656 000401                       BR      ED10            ;BR TO END 'D'
001660 005002                LD10:  CLR     R2              ;CLEAR YES ANSWER FLAG
001662 004737 001766         ED10:  CALL    GTSYAD          ;CALL GET SYSTEM BUS ADR & TEST
001666 004737 002422                CALL    CKSYAD          ;CALL CHECK SYSTEM ADR - MEDIA PROTECT
001672 052777 000001 176604         BIS     #1,@SWR         ;SET SYSTEM#0 AVAIL
001700 012704 016271                MOV     #MSG48,R4       ;SETUP MSG ''SECOND FLOPPY SUBSYSTEM (Y OR N)?''
001704 004737 013232                CALL    TTOUT           ;PRINT
001710 004737 012624                CALL    GETANS          ;CALL GET OPERATOR ANSWER
001714 122737 000131 000530  IE10:  CMPB    #'Y,ANSWER      ;IF ANSWER
001722 001013                       BNE     IF10            ;IS YES THEN
001724 012737 000001 000610         MOV     #1,SUC          ;SET <SUC> = 1
001732 005202                       INC     R2              ;SET YES ANSWER FLAG
001734 004737 001766                CALL    GTSYAD          ;CALL GET SYSTEM BUS ADR & TEST
001740 004737 002422                CALL    CKSYAD          ;CALL CHECK SYSTEM ADR - MEDIA PROTECT
001744 052777 000002 176532         BIS     #2,@SWR         ;SET SYSTEM#1 AVAIL
001752 005737 000516         IF10:  TST     DBGFLG          ;IF DEBUG FLAG
001756 001402                       BEQ     X10             ;SET, THEN
001760 004737 002346                CALL    GTSWR           ;GET SOFTWARE SWITCH REG
001764 000207                X10:   RETURN                  ;RETURN
                                    ;------------------------------------------------------------------------
```

```
                                .SBTTL  MODULE 1.1 - GET SYSTEM BUS ADDRESS & TEST

                                ;BEGIN ROUTINE  (MODULE 1.1.  GET SYSTEM BUS ADDRESS & TEST)
                                ;   BEGIN DO
                                ;    :   GET <SUC> REGISTER & VECTOR ADDRESSES
                                ;    :   SETUP & PRINT BUS ADDRESS
                                ;    :   GET NEW BUS ADDRESS
                                ;    :   SET UP BUS ADDRESS
                                ;    :   SETUP & PRINT VECTOR ADDRESS
                                ;    :   GET NEW VECTOR
                                ;    :   SETUP VECTOR
                                ;    :   CLEAR BUS-TRAP-FLAG
                                ;    :   CALL ADDRESS TEST
                                ;   DO UNTIL BUS_TRAP_FLAG=0
                                ;   SET <SUC> REGISTER & VECTOR ADDRESSES
                                ;ENDROUTINE
                                ;-----------------------------------------------------------------------

001766  000240                  GTSYAD: NOP                             ;
001770  000240                  BA11:   NOP                             ;
001772  004737  010126                  CALL    SSUCOF                  ;CALL SET <SUC> OFFSET
001776  016137  000314  000510          MOV     REGS0(R1),REGS          ;GET <SUC> REGISTER ADR
002004  016137  000310  000512          MOV     VECT0(R1),VECT          ;GET <SUC> VECTOR ADR
002012  005702                  IB11:   TST     R2                      ;IF YES FLAG
002014  001450                          BEQ     EB11                    ;SET, THEN
002016  004737  011132                  CALL    PRTSYS                  ;CALL PRINT SYSTEM IDENT
002022  012704  014072                  MOV     #MSG4,R4                ;USER MODE
002026  004737  013232                  JSR     PC,TTOUT                ;REQUEST REGISTER ADDRESS
002032  013703  000510                  MOV     REGS,R3                 ;
002036  004737  013346                  JSR     PC,OCTP                 ;PRINT CURRENT ADDRESS
002042  012705  000510                  MOV     #REGS,R5                ;SET ADDRESS SAVE LOC
002046  012701  000006                  MOV     #6,R1                   ;SET SIZE OF RESPONSE
002052  012702  177500                  MOV     #177500,R2              ;SET UPPER LIMIT
002056  012703  177100                  MOV     #177100,R3              ;SET LOWER LIMIT
002062  004737  012652                  JSR     PC,TTR                  ;GO GET RESPONSE
002066  012704  014127                  MOV     #MSG5,R4                ;
002072  004737  013232                  JSR     PC,TTOUT                ;REQUEST VECTOR
002076  013703  000512                  MOV     VECT,R3                 ;
002102  004737  013346                  JSR     PC,OCTP                 ;PRINT CURRENT VECTOR
002106  012705  000512                  MOV     #VECT,R5                ;SET ADDRESS SAVE LOC
002112  012701  000003                  MOV     #3,R1                   ;SET SIZE OF RESPONSE
002116  012702  000300                  MOV     #300,R2                 ;SET UPPER LIMIT
002122  012703  000250                  MOV     #250,R3                 ;SET LOWER LIMIT
002126  004737  012652                  JSR     PC,TTR                  ;GO GET RESPONSE
002132  013700  000512                  MOV     VECT,R0                 ;GET VECTOR
002136  005037  000570          EB11:   CLR     BTRPFL                  ;CLEAR BUS TRAP FLAG
002142  013737  000510  000472          MOV     REGS,RXCSAD             ;SET REGISTER ADR
002150  004737  002204                  CALL    ADTST                   ;CALL ADDRESSING TEST
002154  005737  000570          UA11:   TST     BTRPFL                  ;DO UNTIL BUS TRAP FLAG
002160  001303                          BNE     BA11                    ;EQUALS 0
002162  004737  010126                  CALL    SSUCOF                  ;CALL SET <SUC> OFFSET
002166  013761  000510  000314          MOV     REGS,REGS0(R1)          ;SET <SUC> REGISTER ADR
002174  013761  000512  000310          MOV     VECT,VECT0(R1)          ;SET <SUC> VECTOR ADR
002202  000207                  X11:    RETURN                          ;RETURN
                                ;-----------------------------------------------------------------------
```

```
                                        .SBTTL  ADDRESSING TEST
                                        ;------------------------------------------------------------------

     002204  000005                     ADTST:  RESET                           ;INITIALIZE BUS
     002206  012737  014367  000600             MOV     #MSG12,TMSGAD           ;SET TEST MSG HEADER
     002214  012777  012474  176076             MOV     #TRAP,@BTRP             ;SET TRAP HANDLER ADR
     002222  012777  000340  176072             MOV     #340,@BTRP2             ;
     002230  013701  000472                     MOV     RXCSAD,R1               ;GET CSR ADDRESS
     002234  005711                     AD1:    TST     (R1)                    ;REF CS REG
     002236  000240                             NOP                             ;IF ADDRESS IS BAD, BUS TRAP WILL OCCUR
     002240  062701  000002                     ADD     #2,R1                   ;SET ADDRESS OF DB
     002244  005711                             TST     (R1)                    ;REF DB REG
     002246  000240                             NOP                             ;
     002250  005737  000570                     TST     BTRPFL                  ;IF INITIAL ADDRESS TEST
     002254  001427                             BEQ     AD2                     ;DID NOT PASS, THEN
     002256  032777  020000  176220             BIT     #BIT13,@SWR             ;IF INHIBIT ERROR REPORT
     002264  001004                             BNE     2$                      ;NOT SET THEN
     002266  012704  014326                     MOV     #MSG11,R4               ;SET UNIT NOT RESPONDING
     002272  004737  013232                     JSR     PC,TTOUT                ;PRINT MSG
     002276  000240                     2$:     NOP                             ;
     002300  004737  012262                     JSR     PC,BSINIT               ;CHECK SWR FOR BUS INITIALIZE
 (1) 002304  032777  001000  176172             BIT     #SW09,@SWR              ;CHECK FOR LOOP ON ERROR
 (1) 002312  001350                             BNE     AD1                     ;GO TO LOOP ERROR
 (1) 002314  004737  012240                     JSR     PC,ERRSET
 (2) 002320  000001                             .WORD   #1                      ;FATAL ERR # 1 - UNIT NOT RESPONDING TO ADDRESS
 (1) 002322  005777  176156                     TST     @SWR                    ;CHECK FOR HALT ON ERROR
 (1) 002326  100001                             BPL     1$                      ;HALT IF SET
 (1) 002330  000000                             HALT                            ;<UNIT NOT RESPONDING TO ADDRESS>
 (1) 002332  000240                     1$:     NOP
     002334  005077  175760             AD2:    CLR     @BTRP                   ;CLEAR BUS TRAP
     002340  005077  175756                     CLR     @BTRP2                  ;   HANDLER
     002344  000207                     ADTSTX: RTS     PC                      ;RETURN
                                        ;------------------------------------------------------------------
```

```
                                        .SBTTL  MODULE 1.2 - GET SOFTWARE SWITCH REGISTER
                                        ;-----------------------------------------------------------------------
002346  012704  014002        GTSWR:  MOV     #MSG2,R4        ;REQUEST SWITCH REG MSG
002352  004737  013232                JSR     PC,TTOUT        ;
002356  013703  000324                MOV     SWREG,R3        ;SET CURRENT SWITCH REG
002362  004737  013334                JSR     PC,OCTPE        ;PRINT CURRENT SWITCH REG
002366  012737  000324  000504        MOV     #SWREG,SWR      ;SET SOFTWARE SWITCH REG ADR
002374  012705  000324                MOV     #SWREG,R5       ;SET ADDRESS OF SWITCH REG
002400  012701  000006                MOV     #6,R1           ;SET SIZE OR RESPONSE
002404  012702  177777                MOV     #177777,R2      ;SET UPPER LIMIT
002410  012703  000000                MOV     #0,R3           ;SET LOWER LIMIT
002414  004737  012652                JSR     PC,TTR          ;GO GET RESPONSE
002420  000207            X12:        RETURN                  ;RETURN
                                        ;-----------------------------------------------------------------------


                                        .SBTTL  MODULE 1.3 - CHECK SYSTEM ADDRESS

                                        ;BEGINROUTINE (MODULE 1.3 - CHECK SYSTEM ADDRESS)
                                        ;  IF LOAD MEDIA PROTECT BYTE=RX02                              [A]
                                        ;  :  THEN
                                        ;  :    IF <SUC> ADDRESS=RX02 STANDARD ADDRESS                  [B]
                                        ;  :    :  THEN
                                        ;  :    :    CALL SYSTEM IDENTIFICATION
                                        ;  :    :    SETUP & PRINT 'REMOVE XXDP MEDIA FROM THIS SYSTEM'
                                        ;  :    :    SETUP & PRINT '-TYPE 'CR' WHEN READY'
                                        ;  :    :    BEGINDO                                            [C]
                                        ;  :    :    :  CALL TTY INPUT-GET TTY CHARACTER
                                        ;  :    :    DOUNTIL CHARACTER='CR'
                                        ;  :    :    CLEAR MEDIA PROTECT BYTE
                                        ;  :    ENDIF
                                        ;  ENDIF
                                        ;ENDROUTINE
                                        ;-----------------------------------------------------------------------

002422  000240            CKSYAD: NOP                     ;
002424  122737  000015  000041  IA13:   CMPB    #15,@#41        ;IF RX02
002432  001026                        BNE     EA13            ;WAS LOAD MEDIA, THEN
002434  022761  177170  000314  IB13:   CMP     #177170,REGS0(R1);IF SYSTEM ADDRESS
002442  001022                        BNE     EA13            ;SET = RX02 STANDARD ADDRESS, THEN
002444  004737  011132                CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
002450  012704  015140                MOV     #MSG34,R4       ;SET MSG-> 'REMOVE XXDP MEDIA FROM BOOT SYSTEM'
(1) 002454  004737  013232                CALL    TTOUT           ;PRINT MSG
002460  012704  017065                MOV     #MSG58,R4       ;SET MSG-> '-TYPE 'CR' WHEN READY'
(1) 002464  004737  013232                CALL    TTOUT           ;PRINT MSG
002470  004737  013156        BC13:   CALL    TTIN            ;CALL TTY INPUT-GET CHARACTER
002474  122737  000015  000526        CMPB    #15,TIB         ;DO UNTIL
002502  001372                        BNE     BC13            ;CHARACTER = 'CR'
002504  105037  000041                CLRB    @#41            ;CLEAR RX02 BOOT MEDIA PROTECTION
002510  000240            EA13:   NOP                     ;
002512  000207            X13:    RETURN                  ;RETURN


                                        ;-----------------------------------------------------------------------
```

```
                                .SBTTL  MODULE 2.0 - SYSTEM SCHEDULER

                                ;BEGINROUTINE (MOD 2.0 SYSTEM SCHEDULER)
                                ;  SET SYS#0 & SYS#1 READY BITS
                                ;  SET SYS#0 SEL BIT
                                ;  SET <SUC> = 0
                                ;  SET SYS0AVL = ALL
                                ;  CALL INITIALIZE CHECK (MOD 2.1)
                                ;  IF SWR = SYS#1 SELECTED                                          [A]
                                ;  :    THEN
                                ;  :       SET <SUC>=1
                                ;  :       SET SYS1AVL = ALL
                                ;  :       CALL INITIALIZE CHECK (MOD 2.1)
                                ;  ENDIF
                                ;  IF NO SYSTEM AVAIL                                               [B]
                                ;  :    THEN
                                ;  :       SETUP SYSMSG 'NO SYSTEM AVAILABLE''
                                ;  :    ELSE
                                ;  :      BEGINDO                                                   [C]
                                ;  :      :  IF SYS0 SELECTED                                       [D]
                                ;  :      :  :   THEN
                                ;  :      :  :      IF SY0_DONE=NOTSET                             [H]
                                ;  :      :  :      :  THEN
                                ;  :      :  :      :       SET <SUC> = 0
                                ;  :      :  :      :       IF SYS0 RDY SET                        [E]
                                ;  :      :  :      :       :   THEN
                                ;  :      :  :      :       :       CALL  SYSTEM DRIVER (MOD 2.2)
                                ;  :      :  :      :       :   ELSE
                                ;  :      :  :      :       :       CALL WATCH DOG SYS#0 (MOD 2.4)
                                ;  :      :  :      :       ENDIF
                                ;  :      :  :      ENDIF
                                ;  :      :  ENDIF
                                ;  :      :  IF SYS1 SELECTED                                       [F]
                                ;  :      :  :  THEN
                                ;  :      :  :      IF SY1_DSONE=NOTSET                            [I]
                                ;  :      :  :      :  THEN
                                ;  :      :  :      :       SET <SUC> = 1
                                ;  :      :  :      :       IF SYS1 RDY SET                        [G]
                                ;  :      :  :      :       :  THEN
                                ;  :      :  :      :       :     CALL SYSTEM DRIVER (MOD 2.2)
                                ;  :      :  :      :       :  ELSE
                                ;  :      :  :      :       :     CALL WATCH DOG SYS#1 (MOD 2.5)
                                ;  :      :  :      :       ENDIF
                                ;  :      :  :      ENDIF
                                ;  :      :  ENDIF
                                ;  :      :  CLEAR PROGRAM INITIALIZE FLAG
                                ;  :      :  CALL UPDATE SYSTEM STATUS (MOD 2.3)
                                ;  :      DO UNTIL ALL SYSTEMS DONE OR NO SYSTEM AVAIL
                                ;  :      SETUP SYSTEM MSG
                                ;  ENDIF
                                ;  PRINT PRINT SYSTEM MSG
                                ;ENDROUTINE
```

```
                                        ;------------------------------------------------------------------------

002514  005237  000616          SYSCHD: INC     S0RDY           ;SET SYS0 RDY
002520  005237  000620                  INC     S1RDY           ;SET SYS1 RDY
002524  052777  000001  175752          BIS     #1,@SWR         ;SET SYS1 SELECT IN SOFT SWITCH REG
002532  005037  000610                  CLR     SUC             ;SET SYS UNDER CONTROL=#0
002536  012737  000017  000612          MOV     #17,S0AV        ;SET SYSTEM#0 AVAIL=ALL
002544  004737  003002                  CALL    INITCK          ;CALL INITIALIZE CHECK          (MOD 2.1)
002550  032777  000002  175726  IA20:   BIT     #2,@SWR         ;IF SYS #1
002556  001410                          BEQ     IB20            ;SELECTED, THEN
002560  012737  000001  000610          MOV     #1,SUC          ;SET SYS UNDER CONTROL=#1
002566  012737  000017  000614          MOV     #17,S1AV        ;SET SYSTEM#1 AVAIL=ALL
002574  004737  003002                  CALL    INITCK          ;CALL INITIALIZE CHECK          (MOD 2.1)
002600  005737  000612          IB20:   TST     S0AV            ;IF SYSTEM#0
002604  001007                          BNE     ID20            ;NOT AVAIL AND
002606  005737  000614                  TST     S1AV            ;IF SYSTEM#1
002612  001004                          BNE     ID20            ;NOT AVAIL, THEN
002614  012704  016357                  MOV     #MSG49,R4       ;SET MSG='NO SYS AVAIL TO THIS PROG''
002620  000465                          BR      EB20            ;BR TO END 'B'
002622  000240                  BC20:   NOP                     ;BEGIN DO UNTIL 'C'
002624  032777  000001  175652  ID20:   BIT     #1,@SWR         ;IF SYS #0
002632  001421                          BEQ     IF20            ;SELECTED, THEN
002634  005737  000626          IH20:   TST     S0DN            ;IF SYS0 DONE
002640  001016                          BNE     IF20            ;NOT SET, THEN
002642  005037  000610                  CLR     SUC             ;SET SYSTEM UNDER CONTROL=SYS #0
002646  005737  000616          IE20:   TST     S0RDY           ;IF SYS#0 READY FLAG
002652  001407                          BEQ     LE20            ;IS SET, THEN
002654  005037  000616                  CLR     S0RDY           ;CLEAR SYS #0 READY
002660  005237  007670                  INC     WATIN0          ;SET WATCHDOG INIT #0 FLAG
002664  004737  003272                  CALL    SYSDVR          ;CALL SYS DRIVER                (MOD 2.2)
002670  000402                          BR      IF20            ;BR TO IF 'F'
002672  004737  007474          LE20:   CALL    WATCH0          ;CALL WATCH DOG SYS#0           (MOD 2.4)
002676  032777  000002  175600  IF20:   BIT     #2,@SWR         ;IF SYS #1
002704  001422                          BEQ     EF20            ;SELECTED, THEN
002706  005737  000630          II20:   TST     S1DN            ;IF SYS1 DONE
002712  001017                          BNE     EF20            ;NOT SET, THEN
002714  012737  000001  000610          MOV     #1,SUC          ;SET SYSTEM UNDER CONTROL=SYS #1
002722  005737  000620          IG20:   TST     S1RDY           ;IF SYS #1 READY FLAG
002726  001407                          BEQ     LG20            ;IS SET, THEN
002730  005037  000620                  CLR     S1RDY           ;CLEAR SYS #1 READY
002734  005237  007672                  INC     WATIN1          ;SET WATCHDOG INIT #1 FLAG
002740  004737  003272                  CALL    SYSDVR          ;CALL SYS DRIVER                (MOD 2.2)
002744  000402                          BR      EF20            ;BR OT END 'G'
002746  004737  007572          LG20:   CALL    WATCH1          ;CALL WATCH DOG SYS #1          (MOD 2.5)
002752  005037  000514          EF20:   CLR     INITPG          ;CLEAR PROGRAM INITIALIZE FLAG
002756  004737  006570                  CALL    SYSTAT          ;CALL SYSTEM STATUS             (MOD 2.3)
002762  005737  000574          DC20:   TST     SYSFLG          ;DO UNTIL SYSTEM FLAG
002766  001715                          BEQ     BC20            ;SET (EITHER SYSTEM DONE OR NO SYS AVAIL)
002770  013704  000572                  MOV     SYSMSG,R4       ;SETUP SYSTEM MSG
002774  004737  013232          EB20:   CALL    TTOUT           ;PRINT SYSTEM MSG
003000  000207                  X20:    RETURN                  ;RETURN
                                        ;------------------------------------------------------------------------
```

```
.SBTTL  MODULE 2.1 - INITIALIZE CHECK

;BEGINROUTINE (MOD 2.1 - INITIALIZE CHECK
;   CLEAR INITIALIZE_DROP FLAG
;   SET <SUC> ADDRESS & OFFSET
;   SET <SUC> COMMAND=INITIALIZE
;   INITIALIZE SYSTEM AT ADDRESS
;   SETUP DELAY FOR DONE
;   CALL DELAY ROUTINE
;   SET <SUC> OFFSET
;   IF <SUC> CSR = NO DONE BIT SET                                      [A]
;   :  THEN
;   :     SET INITIALIZE_DROP FLAG
;   :     CALL PRINT SYSTEM IDENTIFICATION
;   :     SET APT ERROR #
;   :  ELSE
;   :     IF <SUC> CSR=ERR BIT SET                                      [B]
;   :     :  THEN
;   :     :     SET INITIALIZE_DROP FLAG
;   :     :     CALL PRINT SYSTEM IDENTIFICATION
;   :     :     SETUP & PRINT ''ERROR BIT SET AFTER INITIALIZE''
;   :     :     IF <SUC> ESR = AC LOW BIT SET                           [C]
;   :     :     :  THEN
;   :     :     :     SETUP & PRINT ''AC-LOW-IS FLOPPY SYSTEM POWERED UP?''
;   :     :     ENDIF
;   :     :     IF <SUC> ESR = INITIALIZE DONE NOT SET                  [D]
;   :     :     :   THEN
;   :     :     :      SETUP & PRINT ''INITIALIZE NOT DONE - RUN DIAG UNLESS ERROR''
;   :     :     ENDIF
;   :     :     SET APT ERROR #
;   :     ENDIF
;   :     IF <SUC> CSR NOT=DOUBLE DENSITY TYPE SYSTEM                   [E]
;   :     :  THEN
;   :     :    SET INITIALIZE_DROP FLAG
;   :     :    SETUP & PRINT ''THIS SYSTEM NOT CAPABLE OF DOUBLE DEN OPS''
;   :     ENDIF
;   ENDIF
;   IF INITIALIZE_DROP FLAG SET                                         [F]
;   :  THEN
;   :     CLEAR <SUC> AVAIL
;   ENDIF
;ENDROUTINE
```

```
 6661                                   ;-----------------------------------------------------------------------
 6662
 6663  003002  005037  003270          INITCK: CLR     INITDP          ;CLEAR INITIALIZE DROP FLAG
 6664  003006  004737  010076                  CALL    SSUCAD          ;CALL SET <SUC> ADDRESS
 6665  003012  012761  040000  000622          MOV     #RXINIT,SOCMD(R1);SET <SUC> COMMAND = INITIALIZE
 6666  003020  012777  040000  175444          MOV     #RXINIT,@RXCSAD ;INITIALIZE SYSTEM AT <SUC> CSR ADDRESS
 6667  003026  013737  000472  010230          MOV     RXCSAD,CSRADR   ;SET ADDRESS FOR DELAY TEST
 6668  003034  012737  000040  010226          MOV     #DNBIT,RDYWD    ;SET DONE BIT FOR DELAY TEST
 6669  003042  012737  000015  010222          MOV     #15,RYDX        ;SET DELAY MULTIPLIER
 6670  003050  004737  010136                  CALL    DELAY           ;DELAY FOR DONE BIT
 6671  003054  004737  010126                  CALL    SSUCOF          ;CALL SET <SUC> OFFSET
 6672  003060  032777  000040  175404  IA21:   BIT     #DNBIT,@RXCSAD  ;IF <SUC> CSR DONE BIT
 6673  003066  001014                          BNE     IB21            ;NOT SET, THEN
 6674  003070  005237  003270                  INC     INITDP          ;SET INITIALIZE DROP FLAG
 6675  003074  004737  011132                  CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
 6676  003100  012704  016436                  MOV     #MSG52,R4       ;SET MSG-> 'NO DONE BIT AFTER INITIALIZE''
   (1) 003104  004737  013232                  CALL    TTOUT           ;PRINT MSG
 6677  003110  004737  012240                  JSR     PC,ERRSET
   (2) 003114  000002                          .WORD   #2              ;FATAL ERR # 2 - NO DONE BIT AFTER INITIALIZE
 6678  003116  000453                          BR      EA21            ;BR TO MOD EXIT
 6679  003120  032777  100000  175344  IB21:   BIT     #ERRBIT,@RXCSAD ;IF <SUC> CSR ERROR BIT
 6680  003126  001433                          BEQ     IE21            ;SET, THEN
 6681  003130  005237  003270                  INC     INITDP          ;SET INITIALIZE DROP FLAG
 6682  003134  004737  011132                  CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
 6683  003140  012704  014776                  MOV     #MSG29,R4       ;SET MSG-> 'ERROR BIT SET AFTER INITIALIZE''
   (1) 003144  004737  013232                  CALL    TTOUT           ;PRINT MSG
 6684  003150  032777  000010  175316  IC21:   BIT     #ACLOW,@RXDBAD  ;IF <SUC> ESR AC LOW BIT
 6685  003156  001404                          BEQ     ID21            ;IS SET THEN
 6686  003160  012704  016530                  MOV     #MSG54,R4       ;SET MSG-> ''AC LOW ERROR-IS DISK SYSTEM POWERED UP?''
   (1) 003164  004737  013232                  CALL    TTOUT           ;PRINT MSG
 6687  003170  032777  000004  175276  ID21:   BIT     #INITDN,@RXDBAD ;IF <SUC> ESR INITIALIZE DONE BIT
 6688  003176  001004                          BNE     ED21            ;NOT SET, THEN
 6689  003200  012704  015752                  MOV     #MSG44,R4       ;SET MSG-> ''INITIALIZE NOT DONE-RUN DIAG UNLESS ERR OBVI
   (1) 003204  004737  013232                  CALL    TTOUT           ;PRINT MSG
 6690  003210                          ED21:
   (1) 003210  004737  012240                  JSR     PC,ERRSET
   (2) 003214  000003                          .WORD   #3              ;FATAL ERR # 3 - ERROR BIT SET ON INITIALIZE
 6691  003216  032777  004000  175246  IE21:   BIT     #RX2BIT,@RXCSAD ;IF <SUC>-DOUBLE DENSITY SYSTEM TYPE
 6692  003224  001010                          BNE     EA21            ;BIT NOT SET, THEN
 6693  003226  005237  003270                  INC     INITDP          ;SET INITIALIZE DROP FLAG
 6694  003232  004737  011132                  CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
 6695  003236  012704  014610                  MOV     #MSG23,R4       ;SET MSG-> 'THIS SYS NOT CAPABLE OF DOUBLE DEN OPS''
   (1) 003242  004737  013232                  CALL    TTOUT           ;PRINT MSG
 6696  003246  012761  000017  000612  EA21:   MOV     #17,SOAV(R1)    ;SET <SUC> SYSTEM AVAIL
 6697  003254  005737  003270          IF21:   TST     INITDP          ;IF INITIALIZE DROP FLAG
 6698  003260  001402                          BEQ     X21             ;SET, THEN
 6699  003262  005061  000612                  CLR     SOAV(R1)        ;CLEAR <SUC> SYSTEM AVAILABLE
 6700  003266  000207                  X21:    RETURN                  ;RETURN
 6701                                   ;-----------------------------------------------------------------------
 6702  003270  000000                  INITDP: 0                       ;INITIALIZE DROP FLAG
 6703                                   ;-----------------------------------------------------------------------
```

```
                                .SBTTL  MODULE 2.2 - SYSTEM DRIVER

                                ;BEGINROUTINE (MOD 2.2 SYSTEM DRIVER)
                                ;   CLEAR WATCHDOG INITIALIZE FLAG FOR <SUC>
                                ;   CALL ERROR CHECK ROUTINE (MOD 2.2.5)
                                ;   IF THIS_SYSTEM_AVAILABLE NOT = 0                                          [A]
                                ;   :   THEN
                                ;   :     BEGINDO                                                             [C]
                                ;   :     :   SET <SUC> DRIVE_DONE=GET NEXT COMMAND FLAG
                                ;   :     :   CLEAR GET NEXT COMMAND FLAG
                                ;   :     :   CALL GET NEXT COMMAND (MOD 2.2.1)
                                ;   :     :   IF COMMAND = DONE                                               [D]
                                ;   :     :   :   THEN
                                ;   :     :   :     SET SUC->SYS_DONE_FLAG
                                ;   :     :   :   ELSE
                                ;   :     :   :     GET SYSTEM UNDER CONTROL OFFSET
                                ;   :     :   :     IF COMMAND = DRIVE #1 SELECTED                            [E]
                                ;   :     :   :     :   THEN
                                ;   :     :   :     :     IF SUC-> DRIVE #1 AVAIL                             [F]
                                ;   :     :   :     :     :   THEN
                                ;   :     :   :     :     :     IF COMMAND=SIDE #1 SELECTED                   [G]
                                ;   :     :   :     :     :     :   THEN
                                ;   :     :   :     :     :     :       IF SUC-> SIDE#1_AVAIL NOT SET       [H]
                                ;   :     :   :     :     :     :       :   THEN
                                ;   :     :   :     :     :     :       :     SET GET NEXT COMMAND FLAG
                                ;   :     :   :     :     :     :       ENDIF
                                ;   :     :   :     :     :     :   ENDIF
                                ;   :     :   :     :     :     : ELSE
                                ;   :     :   :     :     :     :   SET GET NEXT COMMAND FLAG
                                ;   :     :   :     :     :   ENDIF
                                ;   :     :   :     :     : ELSE
                                ;   :     :   :     :     :   IF SUC-> DRIVE #0_AVIAL SET                   [I]
                                ;   :     :   :     :     :   :   THEN
                                ;   :     :   :     :     :   :     IF COMMAND = SIDE #1 SELECTED             [J]
                                ;   :     :   :     :     :   :     : THEN
                                ;   :     :   :     :     :   :     :   IF SUC-> SIDE #1_AVIAL NOT SET      [K]
                                ;   :     :   :     :     :   :     :   : THEN
                                ;   :     :   :     :     :   :     :   :   SET GET NEXT COMMAND FLAG
                                ;   :     :   :     :     :   :     :   ENDIF
                                ;   :     :   :     :     :   :     ENDIF
                                ;   :     :   :     :     :   :   ELSE
                                ;   :     :   :     :     :   :     SET GET NEXT COMMAND FLAG
                                ;   :     :   :     :     :   ENDIF
                                ;   :     :   :     :   ENDIF
                                ;   :     :   :   ENDIF
                                ;   :     :   ENDIF
                                ;   :     DOUNTIL GET_NEXT_COMMAND=0
                                ;   :     IF COMMAND NOT = DONE                                               [L]
                                ;   :     :   THEN
                                ;   :     :     IF COMMAND = READ SECTOR                                      [M]
                                ;   :     :     :   THEN
                                ;   :     :     :     CALL TRK & SECTOR (MOD 2.2.2)
                                ;   :     :     ENDIF
                                ;   :     :     CALL DRVR (MOD 2.2.3)
                                ;   :     ENDIF
                                ;   ENDIF
                                ;ENDROUTINE
```

```
                                        ;-----------------------------------------------------------------
003272  053737  000514  004360  SYSDVR: BIS     INITPG,INITTS     ;PASS INITIALIZE FLAG TO TRACK & SECTOR MOD
003300  005037  003556                  CLR     GTCMD             ;CLEAR GET COMMAND FLAG
003304  004737  010126                  CALL    SSUCOF            ;CALL SET <SUC> OFFSET
003310  005261  007670                  INC     WATINO(R1)        ;INCREMENT SYSTEM WATCH DOG INITIALIZE
003314  004737  005164                  CALL    ERRCHK            ;CALL ERROR CHECK (MOD 2.2.5)
003320  004737  010126                  CALL    SSUCOF            ;CALL SET <SUC> OFFSET
003324  005761  000612          IA22:   TST     SOAV(R1)          ;IF SYSTEM AVAILABLE
003330  001511                          BEQ     X22               ;NOT = 0, THEN
003332  053761  003556  003560  BC22:   BIS     GTCMD,SODVDN(R1);SET <SUC> DRIVE DONE =GET COMMAND FLAG
003340  005037  003556                  CLR     GTCMD             ;CLEAR GET COMMAND FLAG
003344  004737  003564                  CALL    GETCMD            ;CALL GET COMMAND (MOD 2.2.1)
003350  005737  000542          ID22:   TST     CMD               ;IF COMMAND
003354  002005                          BGE     LD22              ;EQUALS DONE (=-1), THEN
003356  004737  010126                  CALL    SSUCOF            ;CALL SET <SUC> OFFSET
003362  005261  000626                  INC     SODN(R1)          ;SET SYSTEM DONE FLAG
003366  000451                          BR      UC22              ;BR TO UNTIL 'C'
003370  004737  010126          LD22:   CALL    SSUCOF            ;CALL SET <SUC> OFFSET
003374  032737  000020  000542  IE22:   BIT     #DRV1,CMD         ;IF COMMAND
003402  001422                          BEQ     II22              ;HAS DRIVE #1 SELECTED
003404  032761  000012  000612  IF22:   BIT     #12,SOAV(R1)      ;IF SYSTEM DRIVE #1 AVAILABLE
003412  001413                          BEQ     LF22              ;SET, THEN
003414  032737  001000  000542  IG22:   BIT     #SIDE1,CMD        ;IF COMMAND
003422  001433                          BEQ     UC22              ;HAS SIDE # SELECTED
003424  032761  000010  000612  IH22:   BIT     #10,SOAV(R1)      ;IF <SUC> SIDE #1 AVAILABLE
003432  001027                          BNE     UC22              ;NOT SET, THEN
003434  005237  003556                  INC     GTCMD             ;SET GET COMMAND FLAG
003440  000424                          BR      UC22              ;BR TO ENDIF 'E'
003442  005237  003556          LF22:   INC     GTCMD             ;ELSE SET GET COMMAND FLAG
003446  000421                          BR      UC22              ;BR TO END IF 'E'
003450  032761  000005  000612  II22:   BIT     #5,SOAV(R1)       ;IF SYSTEM DRIVE #0 AVAILABLE
003456  001413                          BEQ     LI22              ;SET, THEN
003460  032737  001000  000542  IJ22:   BIT     #SIDE1,CMD        ;IF COMMAND
003466  001411                          BEQ     UC22              ;HAS SIDE #1 SELECTED, THEN
003470  032761  000004  000612  IK22:   BIT     #4,SOAV(R1)       ;IF SYSTEM SIDE #1 AVAILABLE
003476  001005                          BNE     UC22              ;NOT SET, THEN
003500  005237  003556                  INC     GTCMD             ;SET GET COMMAND FLAG
003504  000402                          BR      UC22              ;BR TO ENDIF 'E'
003506  005237  003556          LI22:   INC     GTCMD             ;SET GET COMMAND FLAG
003512  005737  003556          UC22:   TST     GTCMD             ;DO UNTIL GET NEXT CMD FLAG
003516  001305                          BNE     BC22              ;EQUALS 0
003520  005737  000542          IL22:   TST     CMD               ;IF COMMAND
003524  100413                          BMI     X22               ;NOT=DONE(-1), THEN
003526  013702  000542                  MOV     CMD,R2            ;GET COMMAND TO TEST
003532  042702  177761                  BIC     #177761,R2        ;CLEAR ALL BUT ACTUAL COMMAND
003536  022702  000006          IM22:   CMP     #RDSEC,R2         ;IF COMMAND
003542  001002                          BNE     EM22              ;SET TO READ SECTOR, THEN
003544  004737  004016                  CALL    TRKSEC            ;CALL TRACK & SECTOR UPDATE (MOD 2.2.2)
003550  004737  004402          EM22:   CALL    DRVR              ;CALL UNIT DRIVER (MOD 2.2.3)
003554  000207                  X22:    RETURN                    ;RETURN TO CALLER
                                        ;-----------------------------------------------------------------
003556  000000                  GTCMD:  0                         ;GET COMMAND FLAG
003560  000000                  SODVDN: 0                         ;SYSTEM#0 DRIVE DONE FLAG
003562  000000                  S1DVDN: 0                         ;SYSTEM#1 DRIVE DONE FLAG
                                        ;-----------------------------------------------------------------
```

```
                        .SBTTL  MODULE 2.2.1 - GET COMMAND

                        ;BEGINROUTINE (MOD 2.2.1 - GET COMMAND)
                        ;  INITIALIZE SYSTEM COMMAND TABLE POINTERS, IF INIT SET
                        ;  GET SYSTEM UNDER CONTROL <SUC> COMMAND TABLE POINTER.
                        ;  GET COMMAND AT POINTER
                        ;  IF COMMAND = DONE                                              [A]
                        ;  :  THEN
                        ;  :     RESET <SUC> COMMAND POINTER
                        ;  :  ELSE
                        ;  :     IF COMMAND = READ SECTOR                                 [B]
                        ;  :     :  THEN
                        ;  :     :     GET <SUC> DRIVE DONE FLAG
                        ;  :     :     IF DRIVE_DONE_FLAG SET                           [C]
                        ;  :     :     :  THEN
                        ;  :     :     :     CLEAR <SUC> DRIVE DONE FLAG
                        ;  :     :     :     IF <SUC> NEW COMMAND FLAG NOT SET        [E]
                        ;  :     :     :     :  THEN
                        ;  :     :     :     :     INCREMENT <SUC> COMMAND TABLE POINTER
                        ;  :     :     :     :     GET COMMAND AT POINTER
                        ;  :     :     :     :     IF COMMAND = DONE                [D]
                        ;  :     :     :     :     :  THEN
                        ;  :     :     :     :     :     RESET <SUC> COMMAND TABLE POINTER
                        ;  :     :     :     :     ENDIF
                        ;  :     :     :     ENDIF
                        ;  :     :     :     CLEAR <SUC> NEW COMMAND FLAG
                        ;  :     :     ENDIF
                        ;  :     :  ELSE
                        ;  :     :     INCREMENT <SUC> COMMAND TABLE POINTER
                        ;  :     :     SET <SUC> NEW COMMAND FLAG
                        ;  :     ENDIF
                        ;  ENDIF
                        ;  CLEAR <SUC> DRIVE DONE FLAG
                        ;  SET SELECTED DENSITY ONTO COMMAND
                        ;ENDROUTINE
```

```
                                          ;--------------------------------------------------------------------
003564  005737  000514          GETCMD: TST     INITPG          ;IF INITIALIZE
003570  001404                          BEQ     1$              ;SET, THEN
003572  005037  003762                  CLR     SY0TPT          ;CLEAR SYS #0 TABLE PTR
003576  005037  003764                  CLR     SY1TPT          ;CLEAR SYS #1 TABLE PTR
003602  004737  010126          1$:     CALL    SSUCOF          ;CALL SET <SUC> OFFSET
003606  016102  003762                  MOV     SY0TPT(R1),R2   ;GET <SUC> CMD TABLE PTR
003612  016237  003772  000542          MOV     CMDTBL(R2),CMD  ;GET COMMAND AT POINTER
003620  013703  000542                  MOV     CMD,R3          ;SAVE COMMAND TEMP #3
003624  005703                  IA221:  TST     R3              ;IF COMMAND
003626  002003                          BGE     LA221           ;EQUALS 'DONE'', THEN
003630  005061  003762                  CLR     SY0TPT(R1)      ;RESET <SUC> CMD TABLE PTR
003634  000444                          BR      EA221           ;BR TO EXIT MOD
003636  042703  177761          LA221:  BIC     #177761,R3      ;CLEAR ALL BUT COMMAND
003642  022703  000006          IB221:  CMP     #6,R3           ;IF COMMAND
003646  001032                          BNE     LB221           ;EQUALS 'READ SECTOR'', THEN
003650  016104  003560                  MOV     S0DVDN(R1),R4   ;GET <SUC> DRIVE DONE FLAG
003654  005704                  IC221:  TST     R4              ;IF <SUC> DRIVE DONE FLAG
003656  001433                          BEQ     EA221           ;SET,THEN
003660  005061  003560                  CLR     S0DVDN(R1)      ;CLEAR <SUC> DRIVE DONE FLAG
003664  005761  003766          IE221:  TST     S0CMDF(R1)      ;IF <SUC> NEW COMMAND FLAG
003670  001016                          BNE     EE221           ;NOT SET, THEN
003672  062761  000002  003762          ADD     #2,SY0TPT(R1)   ;INCREMENT <SUC> CMD TABLE PTR
003700  016102  003762                  MOV     SY0TPT(R1),R2   ;GET <SUC> CMD TABLE PTR
003704  016237  003772  000542          MOV     CMDTBL(R2),CMD  ;GET COMMAND AT POINTER
003712  013703  000542                  MOV     CMD,R3          ;SAVE COMMAND IN TEMP#3
003716  005703                  ID221:  TST     R3              ;IF COMMAND
003720  002012                          BGE     EA221           ;EQUALS 'DONE'', THEN
003722  005061  003762                  CLR     SY0TPT(R1)      ;RESET CMD TABLE POINTER
003726  005061  003766          EE221:  CLR     S0CMDF(R1)      ;CLEAR <SUC> NEW COMMAND FLAG
003732  000405                          BR      EA221           ;BR TO EXIT MOD
003734  062761  000002  003762  LB221:  ADD     #2,SY0TPT(R1)   ;INCREMENT <SUC> CMD TABLE PTR
003742  005261  003766                  INC     S0CMDF(R1)      ;SET <SUC> NEW COMMAND FLAG
003746  053737  000506  000542  EA221:  BIS     DENS,CMD        ;SET SELECTED DENSITY
003754  005061  003560                  CLR     S0DVDN(R1)      ;CLEAR <SUC> DRIVE DONE FLAG
003760  000207                  X221:   RETURN                  ;RETURN

                                ;--------------------------------------------------------------------
003762  000000                  SY0TPT: 0                       ;SYSTEM#0 COMMAND TABLE POINTER
003764  000000                  SY1TPT: 0                       ;SYSTEM#1 COMMAND TABLE POINTER
003766  000000                  S0CMDF: 0                       ;SYS#0 NEW COMMAND FLAG
003770  000000                  S1CMDF: 0                       ;SYS#1 NEW COMMAND FLAG
                                ;--------------------------------------------------------------------

                                                                ;DRV#/SIDE#/COMMAND                /PTR
                                                                ;----/-----/-----------------------/-------
003772  000113                  CMDTBL: .WORD   0113            ; 0  /  0  / READ MAINT STATUS     /0
003774  000133                          .WORD   0133            ; 1  /  0  / READ MAINT STATUS     /2
003776  000111                          .WORD   0111            ; 0  /  0  / SET DENSITY           /4
004000  000131                          .WORD   0131            ; 1  /  0  / SET DENSITY           /6
004002  000107                          .WORD   0107            ; 0  /  0  / READ SECTOR           /10
004004  000127                          .WORD   0127            ; 1  /  0  / READ SECTOR           /12
004006  001107                          .WORD   1107            ; 0  /  1  / READ SECTOR           /14
004010  001127                          .WORD   1127            ; 1  /  1  / READ SECTOR           /16
004012  177777                          .WORD   -1              ;DONE TERMINATOR                   /20
004014  177777                          .WORD   -1              ;DONE TERMINATOR                   /22
                                ;--------------------------------------------------------------------
```

```
                        .SBTTL  MODULE 2.2.2 - TRACK & SECTOR UPDATE

                        ;BEGINROUTINE (MOD 2.2.2 TRACK & SECTOR UPDATE)
                        ;  INITIALIZE MODULE, IF INITIALIZE PROGRAM SET
                        ;  IF SOFT_SWITCH_REG->CRC ALL TRACKS SET                                    [A]
                        ;  :  THEN
                        ;  :    IF <SUC> DRIVE SECTOR DONE                                           [B]
                        ;  :    :  THEN
                        ;  :    :    IF <SUC> DRIVE TRACK DONE                                       [C]
                        ;  :    :    :  THEN
                        ;  :    :    :     CLEAR <SUC> DRIVE TRACK DONE
                        ;  :    :    :     SET <SUC> DRIVE TRACK = 76
                        ;  :    :    :     SET <SUC> DRIVE DONE
                        ;  :    :    :  ELSE
                        ;  :    :    :     DECREMENT <SUC> DRIVE TRACK
                        ;  :    :    :     SET TRACK=<SUC> DRIVE TRACK
                        ;  :    :    :     IF <SUC> DRIVE TRACK=0                                  [D]
                        ;  :    :    :     :  THEN
                        ;  :    :    :     :     SET <SUC> DRIVE TRACK DONE
                        ;  :    :    :     ENDIF
                        ;  :    :    :     SET SECTOR = <SUC> DRIVE SECTOR
                        ;  :    :    :  ENDIF
                        ;  :    :    :  CLEAR <SUC> DRIVE SECTOR DONE
                        ;  :    :    ELSE
                        ;  :    :    :  SET <SUC> DRIVE SECTOR=INTERLEAVE + <SUC> DRIVE SECTOR
                        ;  :    :    :  SET SECTOR=<SUC> DRIVE SECTOR
                        ;  :    :    :  IF <SUC> DRIVE SECTOR=26                                    [E]
                        ;  :    :    :  :  THEN
                        ;  :    :    :  :     SET <SUC> DRIVE SECTOR DONE FLAG
                        ;  :    :    :  :     SET <SUC> DRIVE SECTOR=1
                        ;  :    :    :  :  ELSE
                        ;  :    :    :  :     IF SECTOR >26                                        [F]
                        ;  :    :    :  :     :  THEN
                        ;  :    :    :  :     :     SET <SUC> DRIVE SECTOR & SECTOR =2
                        ;  :    :    :  :     ENDIF
                        ;  :    :    :  ENDIF
                        ;  :    :    :  SET TRACK = <SUC> DRIVE TRACK
                        ;  :    :  ENDIF
                        ;  :  ELSE
                        ;  :    IF <SUC> DRIVE TRACK DONE FLAG SET                                   [G]
                        ;  :    :  THEN
                        ;  :    :     SET TRACK=0
                        ;  :    :     SET <SUC> DRIVE TRACK=0
                        ;  :    :     CLEAR <SUC> DRIVE TRACK DONE
                        ;  :    :     SET <SUC> DRIVE DONE
                        ;  :    :  ELSE
                        ;  :    :     SET TRACK =76
                        ;  :    :     SET <SUC> DRIVE TRACK DONE FLAG
                        ;  :    ENDIF
                        ;  :    SET <SUC> DRIVE SECTOR=1
                        ;  :    SET SECTOR= <SUC> DRIVE SECTOR
                        ;  ENDIF
                        ;ENDROUTINE
```

```
 6965                                  ;--------------------------------------------------------------------------------
 6966 004016  005737  004360           TRKSEC: TST     INITTS          ;IF INITIALIZE PROGRAM
```

```
 6968  004022  001426                        BEQ     1$                  ;SET, THEN
 6969  004024  005037  004360                CLR     INITTS              ;CLEAR INITIALIZE FLAG
 6970  004030  005037  004362                CLR     DVTKDN              ;CLEAR SYS#0 DRIVE TRK DONE
 6971  004034  005037  004364                CLR     DVTKDN+2            ;CLEAR SYS#1 DRIVE TRK DONE
 6972  004040  005037  004366                CLR     DVSCDN              ;CLEAR SYS#0 DRIVE SEC DONE
 6973  004044  005037  004370                CLR     DVSCDN+2            ;CLEAR SYS#1 DRIVE SEC DONE
 6974  004050  012737  000114  004376        MOV     #76.,DVTRK          ;SET SYS#0 DRIVE TRK = 76
 6975  004056  012737  000114  004400        MOV     #76.,DVTRK+2        ;SET SYS#1 DRIVE TRK = 76
 6976  004064  012737  000001  004372        MOV     #1,DVSEC            ;SET SYSTEM#0 DRVSEC=1
 6977  004072  012737  000001  004374        MOV     #1,DVSEC+2          ;SET SYSTEM#1 DRVSEC=1
 6978  004100  004737  010126          1$:   CALL    SSUCOF              ;CALL SET <SUC> OFFSET
 6979  004104  032777  000200  174372  IA222:BIT     #CRCALL,@SWR        ;IF SWR -> CRC ALL TRACKS
 6980  004112  001474                        BEQ     IG222               ;SET, THEN
 6981  004114  005761  004366          IB222:TST     DVSCDN(R1)          ;IF <SUC> DRIVE SECTOR DONE FLAG
 6982  004120  001433                        BEQ     LB222               ;SET, THEN
 6983  004122  005761  004362          IC222:TST     DVTKDN(R1)          ;IF <SUC> DRIVE TRACK DONE FLAG
 6984  004126  001410                        BEQ     LC222               ;SET, THEN
 6985  004130  005061  004362                CLR     DVTKDN(R1)          ;CLEAR <SUC> DRIVE TRACK DONE FLAG
 6986  004134  012761  000114  004376        MOV     #76.,DVTRK(R1)      ;SET <SUC> DRIVE TRACK =76
 6987  004142  005261  003560                INC     SODVDN(R1)          ;SET <SUC> DRIVE DONE FLAG
 6988  004146  000415                        BR      EC222               ;BR TO ENDIF 'C'
 6989  004150  016137  004376  000502  LC222:MOV     DVTRK(R1),TRACK     ;SET TRACK = <SUC> DRIVE TRACK
 6990  004156  005361  004376                DEC     DVTRK(R1)           ;DECREMENT <SUC> DRIVE TRACK
 6991  004162  005761  004376          ID222:TST     DVTRK(R1)           ;IF <SUC> DRIVE TRACK
 6992  004166  001002                        BNE     ED222               ;EQUALS ZERO, THEN
 6993  004170  005261  004362                INC     DVTKDN(R1)          ;SET <SUC> DRIVE TRACK DONE FLAG
 6994  004174  012761  000001  004372  ED222:MOV     #1,DVSEC(R1)        ;SET <SUC> DRIVE SECTOR=1
 6995  004202  005061  004366          EC222:CLR     DVSCDN(R1)          ;CLEAR <SUC> DRIVE SECTOR DONE FLAG
 6996  004206  000463                        BR      X222                ;BR TO MOD EXIT
 6997  004210  062761  000002  004372  LB222:ADD     #INTLV,DVSEC(R1)    ;SET <SUC> DRV SEC=INTERLEAVE + OLD DRV SEC
 6998  004216  016137  004372  000500        MOV     DVSEC(R1),SECTR     ;SET SECTOR=<SUC> DRIVE SECTOR
 6999  004224  026127  004372  000032  IE222:CMP     DVSEC(R1),#26.      ;IF <SUC> DRVE SECTOR
 7000  004232  001006                        BNE     IF222               ;EQUALS 26, THEN
 7001  004234  005261  004366                INC     DVSCDN(R1)          ;SET <SUC> DRIVE SECTOR DONE FLAG
 7002  004240  012761  000001  004372        MOV     #1,DVSEC(R1)        ;SET <SUC> DRIVE SECTOR=1
 7003  004246  000412                        BR      EE222               ;BR TO END 'E'
 7004  004250  023727  000500  000032  IF222:CMP     SECTR,#26.          ;IF SECTOR
 7005  004256  103406                        BLO     EE222               ;>26, THEN
 7006  004260  012737  000002  000500        MOV     #2,SECTR            ;SET SECTOR=2
 7007  004266  012761  000002  004372        MOV     #2,DVSEC(R1)        ;SET <SUC> DRIVE SECTOR = 2
 7008  004274  016137  004376  000502  EE222:MOV     DVTRK(R1),TRACK     ;SET TRACK = <SUC> DRIVE TRACK
 7009  004302  000425                        BR      X222                ;BR TO MOD EXIT
 7010  004304  005761  004362          IG222:TST     DVTKDN(R1)          ;IF <SUC> DRIVE TRACK DONE FLAG
 7011  004310  001407                        BEQ     LG222               ;SET, THEN
 7012  004312  005061  004362                CLR     DVTKDN(R1)          ;CLEAR <SUC> DRIVE TRACK DONE FLAG
 7013  004316  005037  000502                CLR     TRACK               ;SET TRACK=0
 7014  004322  005261  003560                INC     SODVDN(R1)          ;SET <SUC> DRIVE DONE FLAG
 7015  004326  000405                        BR      EG222               ;BR TO ENDIF 'G'
 7016  004330  012737  000114  000502  LG222:MOV     #76.,TRACK          ;SET TRACK=76
 7017  004336  005261  004362                INC     DVTKDN(R1)          ;SET <SUC> DRIVE TRACK DONE FLAG
 7018  004342  012761  000001  004372  EG222:MOV     #1,DVSEC(R1)        ;SET <SUC> DRIVE SECTOR=1
 7019  004350  016137  004372  000500        MOV     DVSEC(R1),SECTR     ;SET SECTOR=<SUC> DRIVE SECTOR
 7020  004356  000207                  X222: RETURN                      ;RETURN - MOD EXIT
 7021                                        ;-----------------------------------------------------------------------
```

```
                                ;------------------------------------------------------------------------
                000002                  INTLV   =  2

                                ;------------------------------------------------------------------------
        004360  000000          INITTS: 0                               ;INITIALIZE TRACK & SECTOR FLAG
                                ;------------------------------------------------------------------------

        004362  000000          DVTKDN: .WORD   0                       ;DRIVE TRACK DONE  (DVTKDN)     SYS #0
        004364  000000                  .WORD   0                       ;DRIVE TRACK DONE  (DVTKDN)     SYS #1
        004366  000000          DVSCDN: .WORD   0                       ;DRIVE SECTOR DONE (DVSCDN)     SYS #0
        004370  000000                  .WORD   0                       ;DRIVE SECTOR DONE (DVSCDN)     SYS #1
        004372  000001          DVSEC:  .WORD   1                       ;DRIVE SECTOR      (DVSEC)      SYS #0
        004374  000001                  .WORD   1                       ;DRIVE SECTOR      (DVSEC)      SYS #1
        004376  000114          DVTRK:  .WORD   76.                     ;DRIVE TRACK       (DVTRK)      SYS #0
        004400  000114                  .WORD   76.                     ;DRIVE TRACK       (DVTRK)      SYS #1

                                ;------------------------------------------------------------------------
```

```
 7041                                   .SBTTL  MODULE 2.2.3 - DRIVE DRIVER
 7042
 7043                                   ;BEGINROUTINE (MOD 2.2.3 - DRIVE DRIVER)
 7044                                   ;   CALL SET <SUC> ADDRESS AND OFFSET
 7045                                   ;   SAVE COMMAND IN <SUC> COMMAND
 7046                                   ;   IF COMMAND=SET DENSITY
 7047                                   ;   :   THEN
 7048                                   ;   :     SET <SUC> WATCHDOG MULTIPLIER LIMIT=400
 7049                                   ;   :   ELSE
 7050                                   ;   :     SET <SUC> WATCHDOG MULTIPLIER LIMIT=20
 7051                                   ;   ENDIF
 7052                                   ;   SETUP SECTOR FOR PRINT
 7053                                   ;   SETUP TRACK FOR PRINT
 7054                                   ;   SETUP SECTOR ADR FOR OUTPUT
 7055                                   ;   SETUP TRACK ADR FOR OUTPUT
 7056                                   ;   SETUP COMMAND FOR OUTPUT
 7057                                   ;   CALL SET PRIORITY HIGH
 7058                                   ;   CALL OUTPUT COMMAND
 7059                                   ;   CALL SET PRIORITY LOW
 7060                                   ;ENDROUTINE
 7061
 7062                                   ;-----------------------------------------------------------------------
 7063
 7064  004402  004737  010076           DRVR:   CALL    SSUCAD          ;CALL SET <SUC> ADDRESS & OFFSET
 7065  004406  013761  000542  000622           MOV     CMD,SOCMD(R1)   ;SAVE COMMAND IN <SUC> COMMAND
 7066  004414  013702  000542                   MOV     CMD,R2          ;GET COMMAND
 7067  004420  042702  177761                   BIC     #177761,R2      ;CLEAR ALL BUT COMMAND
 7068  004424  022702  000010           IA223:  CMP     #SETDEN,R2      ;IF COMMAND
 7069  004430  001004                           BNE     LA223           ;EQUALS ''SET DENSITY''
 7070  004432  012761  000400  000556           MOV     #400,SOWMLT(R1) ;SET <SUC> WATCHDOG MULT LIMIT=400
 7071  004440  000403                           BR      EA223           ;BR TO END 'A'
 7072  004442  012761  000020  000556  LA223:  MOV     #20,SOWMLT(R1)  ;SET <SUC> WATCHDOG MULT LIMIT=20
 7073  004450  013761  000500  011310  EA223:  MOV     SECTR,SOSEC(R1) ;SETUP SECTOR FOR PRINT
 7074  004456  013761  000502  011304           MOV     TRACK,SOTRK(R1) ;SETUP TRACK FOR PRINT
 7075  004464  013737  000500  005114           MOV     SECTR,SECADR    ;SETUP SECTOR ADR FOR OUTPUT
 7076  004472  013737  000502  005112           MOV     TRACK,TRKADR    ;SETUP TRACK ADR FOR OUTPUT
 7077  004500  013737  000542  000540           MOV     CMD,WDOT        ;SETUP COMMAND FOR OUTPUT
 7078  004506  004737  010020                   CALL    PRIHI           ;CALL SET PRIORITY HIGH
 7079  004512  004737  004524                   CALL    OUTCMD          ;CALL OUTPUT COMMAND
 7080  004516  004737  010034                   CALL    PRILO           ;CALL SET PRIORITY LOW
 7081  004522  000207                   X223:   RETURN                  ;RETURN
 7082                                   ;-----------------------------------------------------------------------
```

```
                                .SBTTL  MODULE 2.2.3.1 - OUTPUT DRIVE COMMAND
                                ;-----------------------------------------------------------------------
004524 012737 000040 010226 OUTCMD: MOV     #DNBIT,RDYWD    ;READY TEST WD          (PASS TO 2.2.3.1.1)
004532 013737 000540 005102         MOV     WDOT,WRDS       ;WORD FOR OUTPUT        (PASS TO 2.2.3.1.1)
004540 013737 000472 005104         MOV     RXCSAD,ADRS     ;ADDRESS OF OUTPUT      (PASS TO 2.2.3.1.1)
004546 004737 005124                JSR     PC,OUTSWD       ;OUTPUT FUNCTION WD (FW) DO 2.2.3.1.1)
004552 032737 000010 000540 IB2231: BIT     #10,WDOT        ;THEN, IF FUNCTION IS
004560 001043                       BNE     IC2231          ;'READ, WRITE, FILL, EMPTY'' (FW BIT #3=0)
004562 032737 000004 000540 IH2231: BIT     #4,WDOT         ;AND THEN IF FUNCTION IS
004570 001047                       BNE     LH2231          ;'EMPTY, FILL'' (FW BIT#2=0)
004572 012737 000200 010226         MOV     #TRBIT,RDYWD    ;THEN SET OUTPUT READY TEST WORD (PASS TO 2.2.3.1.1)
004600 013737 005122 005102         MOV     WDCT,WRDS       ;AND SET WORD FOR OUTPUT        (PASS TO 2.2.3.1.1)
004606 013737 000474 005104         MOV     RXDBAD,ADRS     ;AND SET ADDRESS OF OUTPUT      (PASS TO 2.2.3.1.1)
004614 004737 005124                JSR     PC,OUTSWD       ;OUTPUT BASE ADDRESS WORD       DO 2.2.3.1.1
004620 032737 000002 000540 IK2231: BIT     #2,WDOT         ;IF 'FILL'' (FW BIT#1=0)
004626 001004                       BNE     LK2231          ;THEN
004630 013737 005116 005102         MOV     BAFILL,WRDS     ;SET DATA FILL BUFFER ADR       (PASS TO 2.2.3.1.1)
004636 000403                       BR      EK2231          ;BR TO END IF 'K'
004640 013737 005120 005102 LK2231: MOV     BAEMPT,WRDS     ;SET DATA EMPTY BUFFER ADR.     (PASS TO 2.2.3.1.1)
004646 012737 000200 010226 EK2231: MOV     #TRBIT,RDYWD    ;SET OUTPUT READY TEST WORD     (PASS TO 2.2.3.1.1)
004654 013737 000474 005104         MOV     RXDBAD,ADRS     ;ADDRESS OF OUTPUT              (PASS TO 2.2.3.1.1)
004662 004737 005124                JSR     PC,OUTSWD       ;OUTPUT WORD COUNT WORD         DO 2.2.3.1.1
004666 000444                       BR      EH2231          ;BRANCH TO END IF 'H'
004670 032737 000004 000540 IC2231: BIT     #4,WDOT         ;IF FUNCTION WORD IS
004676 001455                       BEQ     IE2231          ;'WRITE D.D'' OR 'READ E.C'' (FW BIT #2=1)
004700 032737 000002 000540 ID2231: BIT     #2,WDOT         ;THEN, IF FUNCTION IS
004706 001035                       BNE     LD2231          ;'WRITE D.D'', THEN (FW BIT#1=0)
004710 012737 000200 010226 LH2231: MOV     #TRBIT,RDYWD    ;SET OUTPUT READY TEST WORD
004716 013737 005114 005102         MOV     SECADR,WRDS     ;MOVE TRACK AND SECTOR ADDRESS
004724 042737 177700 005102         BIC     #177700,WRDS    ;FORMAT TO SECTOR ADDRESS
004732 013737 000474 005104         MOV     RXDBAD,ADRS     ;ADDRESS OF OUTPUT
004740 004737 005124                JSR     PC,OUTSWD       ;OUTPUT SECTOR ADDRESS
004744 013737 005112 005102         MOV     TRKADR,WRDS     ;MOVE TRACK AND SECTOR ADDRESS
004752 042737 177600 005102         BIC     #177600,WRDS    ;FORMAT TRACK ADDRESS
004760 012737 000200 010226         MOV     #TRBIT,RDYWD    ;SET OUTPUT READY TEST WORD
004766 013737 000474 005104         MOV     RXDBAD,ADRS     ;ADDRESS OF OUTPUT
004774 004737 005124                JSR     PC,OUTSWD       ;OUTPUT TRACK ADDRESS
005000 000437               EH2231: BR      END2231         ;ENDIF H -DONE
005002 012737 000200 010226 LD2231: MOV     #TRBIT,RDYWD    ;SET READY WD TO TR MODE
005010 012737 017446 005102         MOV     #XER,WRDS       ;EXT ERR. CODE TABLE ADD
005016 013737 000474 005104         MOV     RXDBAD,ADRS     ;ADDRESS OF OUTPUT, RXDB
005024 004737 005124                JSR     PC,OUTSWD       ;O/P BASE ADD FOR ERR. CODE
005030 000423                       BR      END2231         ;DONE
005032 032737 000002 000540 IE2231: BIT     #2,WDOT         ;IF FUNCTION IS
005040 001404                       BEQ     LE2231          ;'READ STATUS'' (FW BIT#1=1)
005042 012737 000001 005106 TE2231: MOV     #1,ERSTAT       ;THEN-SET ERR STATUS FLAG
005050 000413                       BR      END2231         ;DONE
005052 012737 000200 010226 LE2231: MOV     #TRBIT,RDYWD    ;SET OUTPUT READY TEST WD
005060 013737 005110 005102         MOV     VALWD,WRDS      ;VALIDATION WORD
005066 013737 000474 005104         MOV     RXDBAD,ADRS     ;ADDRESS OF OUTPUT, RXDB
005074 004737 005124                JSR     PC,OUTSWD       ;OUTPUT VALIDATION WORD
005100 000207               END2231: RTS    PC              ;RETURN TO MOD 2.3
                                ;-----------------------------------------------------------------------
```

```
                                         ;------------------------------------------------------------------------
005102  000000                           WRDS:   0                              ;MODULE 2.2.3.1.1 OUTPUT WORD
005104  000000                           ADRS:   0                              ;MODULE 2.2.3.1.1 OUTPUT ADDRESS
005106  000000                           ERSTAT: 0                              ;MODULE 0.0 ERR STATUS READ FLAG
005110  000111                           VALWD:  111                            ;VALIDATION WD (SET DENS-ASCII "I")
005112  000000                           TRKADR: 0                              ;TRACK ADDRESS
005114  000000                           SECADR: 0                              ;SECTOR ADDRESS
005116  000000                           BAFILL: 0                              ;BASE ADDRESS FILL BUFFER
005120  000000                           BAEMPT: 0                              ;BASE ADDRESS EMPTY BUFFER
005122  000000                           WDCT:   0                              ;WORD COUNT
                                         ;MOD 2.2.3.1 ----- END MODULE -------------------------------------------

                                         .SBTTL  MODULE 2.2.3.1.1 - OUTPUT SINGLE WORD
                                         ;------------------------------------------------------------------------

005124  013737  000472  010230           OUTSWD: MOV     RXCSAD,CSRADR          ;SET C&S REG ADR
005132  013737  010226  010226                   MOV     RDYWD,RDYWD            ;OUTPUT READY WORD (PASS TO DELAY)
005140  004737  010136                           JSR     PC,DELAY               ;DELAY FOR READY   DO DELAY
005144  033777  010226  173320                   BIT     RDYWD,@RXCSAD          ;IF READY,
005152  001403                                   BEQ     X22311                 ;THEN
005154  013777  005102  177722                   MOV     WRDS,@ADRS             ;MOV WORD TO ADDRESS
005162  000207                           X22311: RTS     PC                     ;RETURN
                                         ;MOD 2.2.3.1.1 ----- END MODULE -----------------------------------------
```

```
7162                                    .SBTTL  MODULE 2.2.5 - ERROR CHECK
7163
7164                                    ;BEGINROUTINE (MOD 2.2.5 - ERROR CHECK)
7165                                    ;  CLEAR PRINT HEADER FLAG & ERROR FLAG
7166                                    ;  CALL SET <SUC> ADDRESS & OFFSET
7167                                    ;  GET <SUC> COMMAND (FROM LAST OPERATION)
7168                                    ;  IF <SUC> COMMAND=READ MAINT STATUS                                  [A]
7169                                    ;  :  THEN
7170                                    ;  :    SET COMMAND HEADER='READ MAINT STATUS''
7171                                    ;  :    SETUP CSR & ESR CHECK BITS & DON'T CARE BITS
7172                                    ;  :    CALL CSR ERROR CHECK & CALL ESR ERROR CHECK
7173                                    ;  :    SET ERROR FLAG, IF ANY ERRORS
7174                                    ;  :    IF ERROR_FLAG SET                                            [B]
7175                                    ;  :    :  THEN
7176                                    ;  :    :    SET ERROR=<READ MAINT STATUS COMMAND ERROR>
7177                                    ;  :    :    CALL DRIVES DROP                    (MOD 2.2.5.1)
7178                                    ;  :    :    CALL LOOP                           (MOD 2.2.5.4)
7179                                    ;  :    :  ELSE
7180                                    ;  :    :    CALL DRIVES AVAILABLE CHECK         (MOD 2.2.5.2)
7181                                    ;  :    ENDIF
7182                                    ;  :  ELSE
7183                                    ;  :    IF <SUC> COMMAND=SET DENSITY                                 [C]
7184                                    ;  :    :  THEN
7185                                    ;  :    :    SET COMMAND HEADER='SET DENSITY ERROR''
7186                                    ;  :    :    CALL SET TEST BITS                  (MOD 2.2.5.3)
7187                                    ;  :    :    CALL CSR ERROR CHECK & CALL ESR ERROR CHECK
7188                                    ;  :    :    SET ERROR FLAG, IF ANY ERRORS
7189                                    ;  :    :    IF ERROR_FLAG SET                                        [D]
7190                                    ;  :    :    :  THEN
7191                                    ;  :    :    :    SET ERROR=<SET DENSITY COMMAND ERROR>
7192                                    ;  :    :    :    CALL DRIVES DROP                (MOD 2.2.5.1)
7193                                    ;  :    :    :    CALL LOOP                       (MOD 2.2.5.4)
7194                                    ;  :    :    ENDIF
7195                                    ;  :    :  ELSE
7196                                    ;  :    :    IF <SUC> COMMAND=READ SECTOR                             [F]
7197                                    ;  :    :    :  THEN
7198                                    ;  :    :    :    SET COMMAND HEADER='READ SECTOR COMMAND ERROR''
7199                                    ;  :    :    :    CALL SET TEST BITS              (MOD 2.2.5.3)
7200                                    ;  :    :    :    CALL CSR ERROR CHECK & CALL ESR ERROR CHECK
7201                                    ;  :    :    :    SET ERROR FLAG, IF ANY ERRORS
7202                                    ;  :    :    :    IF ERROR_FLAG SET                                    [E]
7203                                    ;  :    :    :    :  THEN
7204                                    ;  :    :    :    :    IF RXDB=CRC ERROR                              [G]
7205                                    ;  :    :    :    :    :  THEN
7206                                    ;  :    :    :    :    :    CALL TRACK & SECTOR IDENT
7207                                    ;  :    :    :    :    ENDIF
7208                                    ;  :    :    :    :    SET ERROR=<READ SECTOR COMMAND ERROR>
7209                                    ;  :    :    :    :    CALL DRIVES DROP            (MOD 2.2.5.1)
7210                                    ;  :    :    :    :    CALL LOOP                   (MOD 2.2.5.4)
7211                                    ;  :    :    :    ENDIF
7212                                    ;  :    :    ENDIF
7213                                    ;  :    ENDIF
7214                                    ;  ENDIF
7215                                    ;ENDROUTINE
```

```
                                          ;------------------------------------------------------------------

005164  005037  000602           ERRCHK: CLR     HDRFLG            ;CLEAR HEADER FLAG
005170  005037  005526                   CLR     ERRFLG            ;CLEAR ERROR FLAGS
005174  004737  010076                   CALL    SSUCAD            ;CALL SET <SUC> ADDRESS & OFFSET
005200  016102  000622                   MOV     SOCMD(R1),R2      ;GET <SUC> LAST COMMAND
005204  042702  177761                   BIC     #177761,R2        ;CLEAR ALL BUT COMMAND
005210  022702  000012           IA225:  CMP     #RDMNST,R2        ;IF COMMAND
005214  001047                           BNE     IC225             ;EQUALS 'READ MAINT STATUS'', THEN
005216  012737  017346  000600           MOV     #MSGCD5,TMSGAD    ;SETUP COMMAND HEADER
005224  012737  000040  000546           MOV     #DNBIT,CSRCMP     ;SET DONE BIT CK-CSR COMPARE
005232  052737  004000  000546           BIS     #RX2BIT,CSRCMP    ;SET DOUBLE DENSITY BIT CK-CSR COMPARE
005240  012737  173537  000544           MOV     #173537,CSRMSK    ;SET CSR DON'T CARE BITS
005246  005037  000552                   CLR     ESRCMP            ;SET DRV RDY-ESR COMPARE
005252  012737  171662  000550           MOV     #171662,ESRMSK    ;SET-ESR DON'T CARE BITS
005260  004737  010404                   CALL    CSRCHK            ;CALL CSR ERROR CHECK
005264  010037  005526                   MOV     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005270  004737  010624                   CALL    ESRCHK            ;CALL ESR ERROR CHECK
005274  050037  005526                   BIS     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005300  005737  005526           IB225:  TST     ERRFLG            ;IF ERROR FLAG
005304  001410                           BEQ     LB225             ;SET, THEN
005306  004737  012240                   JSR     PC,ERRSET
005312  000004                           .WORD   #4                ;FATAL ERR # 4 - READ MAINT STATUS COMMAND ERROR
005314  004737  005530                   CALL    DRVDRP            ;CALL DRIVES DROP         (MOD 2.2.5.1)
005320  004737  006406                   CALL    LOOP              ;CALL LOOPING MODULE      (MOD 2.2.5.4)
005324  000476                           BR      EA225             ;BR TO MOD EXIT
005326  004737  006040           LB225:  CALL    DRVAVL            ;CALL DRIVES AVAIL CHK   (MOD 2.2.5.2)
005332  000473                           BR      EA225             ;BR TO MOD EXIT
005334  022702  000010           IC225:  CMP     #SETDEN,R2        ;IF COMMAND
005340  001030                           BNE     IF225             ;EQUALS 'SET DENSITY'', THEN
005342  012737  017311  000600           MOV     #MSGCD4,TMSGAD    ;SETUP COMMAND HEADER
005350  004737  006266                   CALL    STSTBT            ;CALL SET TEST BITS       (MOD 2.2.5.3)
005354  004737  010404                   CALL    CSRCHK            ;CALL CSR ERROR CHECK
005360  010037  005526                   MOV     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005364  004737  010624                   CALL    ESRCHK            ;CALL ESR ERROR CHECK
005370  050037  005526                   BIS     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005374  005737  005526           ID225:  TST     ERRFLG            ;IF ERROR FLAG
005400  001450                           BEQ     EA225             ;SET, THEN
005402  004737  012240                   JSR     PC,ERRSET
005406  000005                           .WORD   #5                ;FATAL ERR # 5 - SET DENSITY COMMAND ERROR
005410  004737  005530                   CALL    DRVDRP            ;CALL DRIVES DROP         (MOD 2.2.5.1)
005414  004737  006406                   CALL    LOOP              ;CALL LOOPING MODULE      (MOD 2.2.5.4)
005420  000440                           BR      EA225             ;
005422  022702  000006           IF225:  CMP     #RDSEC,R2         ;IF COMMAND
005426  001035                           BNE     EA225             ;EQUALS 'READ SECTOR'', THEN
005430  012737  017254  000600           MOV     #MSGCD3,TMSGAD    ;SET COMMAND HEADER
005436  004737  006266                   CALL    STSTBT            ;CALL SET TEST BITS       (MOD 2.2.5.3)
005442  004737  010404                   CALL    CSRCHK            ;CALL CSR ERROR CHECK
005446  010037  005526                   MOV     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005452  004737  010624                   CALL    ESRCHK            ;CALL ESR ERROR CHECK
005456  050037  005526                   BIS     R0,ERRFLG         ;SET ERROR FLAG, IF ANY ERRORS
005462  005737  005526           IE225:  TST     ERRFLG            ;IF ERROR FLAG
005466  001415                           BEQ     EA225             ;SET, THEN
005470  032777  000001  172776   IG225:  BIT     #CRCERR,@RXDBAD   ;IF CRC ERROR
005476  001402                           BEQ     EG225             ;SET, THEN
005500  004737  011232                   CALL    PRTKSC            ;CALL PRINT TRACK/SECTOR
```

```
 7271  005504                          EG225:
  (1)  005504  004737  012240                JSR     PC,ERRSET
  (2)  005510  000006                        .WORD   #6              ;FATAL ERR # 6 - READ SECTOR COMMAND ERROR
 7272  005512  004737  005530                CALL    DRVDRP          ;CALL DRIVES DROP       (MOD 2.2.5.1)
 7273  005516  004737  006406                CALL    LOOP            ;CALL LOOPING MODULE    (MOD 2.2.5.4)
 7274  005522  000240                EA225:  NOP                     ;
 7275  005524  000207                X225:   RETURN                  ;RETURN
 7276                                ;-----------------------------------------------------------------------
 7277  005526  000000                ERRFLG: 0                       ;ERROR FLAG
 7278                                ;-----------------------------------------------------------------------
```

```
                        .SBTTL  MODULE 2.2.5.1 - DRIVE DROP

                        ;BEGINROUTINE (MOD 2.2.5.1 - DRIVE DROP)
                        ;  CLEAR DROP_DRIVE CONTROL WORD
                        ;  CLEAR PRINT_SIDE_FLAG
                        ;  CALL SET <SUC> OFFSET
                        ;  IF <SUC> COMMAND=DRIVE #1 SELECTED                    [A]
                        ;  :  THEN
                        ;  :    IF <SUC> COMMAND=SIDE #1 SELECTED                [B]
                        ;  :    :  THEN
                        ;  :    :    IF <SUC> SYS_AVAIL=SIDE #1                  [C]
                        ;  :    :    :  THEN
                        ;  :    :    :    SETUP PRINT SIDE=#1, DRIVE=#1
                        ;  :    :    :    SET DROP_DRIVE=SIDE 1/DRIVE 1
                        ;  :    :    :    SET PRINT SIDE FLAG
                        ;  :    :    ENDIF
                        ;  :    :  ELSE
                        ;  :    :    SET DROP _DRIVE=SIDE 0/DRIVE 1
                        ;  :    :    IF <SUC> SYS_AVAIL=SIDE #1                  [D]
                        ;  :    :    :  THEN
                        ;  :    :    :    SETUP PRINT SIDE=#0, DRIVE=#1
                        ;  :    :    :    SET PRINT_SIDE_FLAG
                        ;  :    :    :  ELSE
                        ;  :    :    :    SETUP PRINT DRIVE=#1
                        ;  :    :    ENDIF
                        ;  :    ENDIF
                        ;  :  ELSE
                        ;  :    IF <SUC> COMMAND=SIDE #1 SELECTED                [E]
                        ;  :    :  THEN
                        ;  :    :    IF <SUC> SYS_AVAIL=SIDE #1                  [F]
                        ;  :    :    :  THEN
                        ;  :    :    :    SETUP PRINT SIDE=#1, DRIVE=#0
                        ;  :    :    :    SET DROP_DRIVE=SIDE 1/DRIVE 0
                        ;  :    :    ENDIF
                        ;  :    :  ELSE
                        ;  :    :    SET DROP_DRIVE=SIDE 0/ DRIVE 0
                        ;  :    :    IF <SUC> SYS_AVAIL=SIDE #1                  [G]
                        ;  :    :    :  THEN
                        ;  :    :    :    SETUP PRINT SIDE=#0, DRIVE=#0
                        ;  :    :    :    SET PRINT_SIDE_FLAG
                        ;  :    :    :  ELSE
                        ;  :    :    :    SETUP PRINT DRIVE=#0
                        ;  :    :    ENDIF
                        ;  :    ENDIF
                        ;  ENDIF
                        ;  IF DROP_DRIVE NOT=0                                   [H]
                        ;  :  THEN
                        ;  :    CALL PRINT SYSTEM IDENTIFICATION
                        ;  :    CALL PRINT DRIVE IDENTIFICATION
                        ;  :    DROP DRIVES SET IN DROP_DRIVE FROM <SUC> AVAIL
                        ;  ENDIF
                        ;ENDROUTINE
```

```
7333                                    ;-----------------------------------------------------------------------
7334
7335  005530  005037  006036            DRVDRP: CLR     DROPDV          ;CLEAR DROP DRIVES CONTROL WORD
7336  005534  005037  011230                    CLR     PTSIDF          ;CLEAR PRINT SIDE FLAG
7337  005540  004737  010126                    CALL    SSUCOF          ;CALL SET <SUC> OFFSET
7338  005544  032761  000020  000622    IA2251: BIT     #DRV1,SOCMD(R1) ;IF <SUC> COMMAND=DRIVE #1
7339  005552  001447                            BEQ     IE2251          ;SELECTED, THEN
7340  005554  032761  001000  000622    IB2251: BIT     #SIDE1,SOCMD(R1);IF <SUC> COMMAND=SIDE #1
7341  005562  001420                            BEQ     LB2251          ;SELECTED, THEN
7342  005564  032761  000010  000612    IC2251: BIT     #10,SOAV(R1)    ;IF <SUC> SYS AVAIL=SIDE #1
7343  005572  001502                            BEQ     IH2251          ;SET, THEN
7344  005574  012737  000001  011226            MOV     #1,SIDE         ;SETUP PRINT SIDE=#1
7345  005602  012737  000001  011224            MOV     #1,DRIVE        ;SETUP PRINT DRIVE=#1
7346  005610  012737  000010  006036            MOV     #10,DROPDV      ;SET DROP DRIVE=SIDE 1/DRIVE 1
7347  005616  005237  011230                    INC     PTSIDF          ;SET PRINT SIDE FLAG
7348  005622  000466                            BR      IH2251          ;BR TO IF 'H'
7349  005624  012737  000002  006036    LB2251: MOV     #2,DROPDV       ;SET DROP DRIVE=SIDE 0/DRIVE 1
7350  005632  032761  000010  000612    ID2251: BIT     #10,SOAV(R1)    ;IF <SUC> SYS AVAIL=SIDE #1
7351  005640  001410                            BEQ     LD2251          ;SET, THEN
7352  005642  005037  011226                    CLR     SIDE            ;SETUP PRINT SIDE=#0
7353  005646  012737  000001  011224            MOV     #1,DRIVE        ;SETUP PRINT DRIVE=#1
7354  005654  005237  011230                    INC     PTSIDF          ;SET PRINT SIDE FLAG
7355  005660  000447                            BR      IH2251          ;BR TO IF 'H'
7356  005662  012737  000001  011224    LD2251: MOV     #1,DRIVE        ;SETUP PRINT DRIVE=#1
7357  005670  000443                            BR      IH2251          ;BR TO IF 'H'
7358  005672  032761  001000  000622    IE2251: BIT     #SIDE1,SOCMD(R1);IF <SUC> COMMAND=SIDE #1
7359  005700  001417                            BEQ     LE2251          ;SELECTED, THEN
7360  005702  032761  000001  000612    IF2251: BIT     #1,SOAV(R1)     ;IF <SUC> AVAIL=SIDE #1/DRIVE #0
7361  005710  001433                            BEQ     IH2251          ;SET, THEN
7362  005712  012737  000001  011226            MOV     #1,SIDE         ;SETUP PRINT SIDE=#1
7363  005720  005037  011224                    CLR     DRIVE           ;SETUP PRINT DRIVE=#0
7364  005724  012737  000004  006036            MOV     #4,DROPDV       ;SET DROP DRIVE=SIDE 1/DRIVE 0
7365  005732  005237  011230                    INC     PTSIDF          ;SET PRINT SIDE FLAG
7366  005736  000420                            BR      IH2251          ;BR TO IF 'H'
7367  005740  012737  000001  006036    LE2251: MOV     #1,DROPDV       ;SET DROP DRIVE=SIDE 0/DRIVE 0
7368  005746  032761  000004  000612    IG2251: BIT     #4,SOAV(R1)     ;IF <SUC> SYS AVAIL=SIDE #1/DRIVE #0
7369  005754  001407                            BEQ     LG2251          ;SET, THEN
7370  005756  005037  011226                    CLR     SIDE            ;SETUP PRINT SIDE=#0
7371  005762  005037  011224                    CLR     DRIVE           ;SETUP PRINT DRIVE=#0
7372  005766  005237  011230                    INC     PTSIDF          ;SET PRINT SIDE FLAG
7373  005772  000402                            BR      IH2251          ;BR TO IF 'H'
7374  005774  005037  011224    LG2251: CLR     DRIVE           ;SETUP PRINT DRIVE=#0
7375  006000  005737  006036    IH2251: TST     DROPDV          ;IF DROP DRIVE CONTROL WORD
7376  006004  001413                            BEQ     X2251           ;SET, THEN
7377  006006  004737  011132                    CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
7378  006012  004737  011154                    CALL    PRTDRV          ;CALL PRINT DRIVE IDENT
7379  006016  043761  006036  000612          BIC     DROPDV,SOAV(R1) ;DROP DRIVES SET IN <SUC> AVAIL
7380  006024  012704  015121                    MOV     #MSG33,R4       ;SET MSG-> 'DROPPED''
 (1)  006030  004737  013232                    CALL    TTOUT           ;PRINT MSG
7381  006034  000207                    X2251:  RETURN                  ;RETURN
7382                                    ;-----------------------------------------------------------------------
7383  006036  000000                    DROPDV: 0                       ;DROP DRIVE CONTROL WORD
7384                                    ;-----------------------------------------------------------------------
```

```
7386                                    .SBTTL  MODULE 2.2.5.2 - DRIVES AVAILABLE CHECK
7387
7388                                    ;BEGINROUTINE (MOD 2.2.5.2 - DRIVES AVAILABLE CHECK)
7389                                    ;  CLEAR DROP AVAILABLE DRIVES CONTROL
7390                                    ;  IF <SUC> COMMAND=DRIVE #1 SELECTED                          [A]
7391                                    ;  :  THEN
7392                                    ;  :    IF RXES=DRIVE NOT RDY                                  [B]
7393                                    ;  :    :  THEN
7394                                    ;  :    :    IF <SUC> AVAIL=DRIVE #1 SET                      [C]
7395                                    ;  :    :    :  THEN
7396                                    ;  :    :    :    SET DRIVE=1
7397                                    ;  :    :    :    SET DROP_AVAIL_DRIVE=#1
7398                                    ;  :    :    ENDIF
7399                                    ;  :    ENDIF
7400                                    ;  :    IF RXES=SIDE NOT RDY                                   [D]
7401                                    ;  :    :  THEN
7402                                    ;  :    :    DROP <SUC> AVAIL-SIDE #1/DRIVE #1
7403                                    ;  :    ENDIF
7404                                    ;  :  ELSE
7405                                    ;  :    IF RXES=DRIVE NOT RDY                                  [E]
7406                                    ;  :    :  THEN
7407                                    ;  :    :    IF <SUC> AVAIL=DRIVE #0 SET                      [F]
7408                                    ;  :    :    :  THEN
7409                                    ;  :    :    :    SET DRIVE=0
7410                                    ;  :    :    :    SET DROP_AVAIL_DRIVE=#0
7411                                    ;  :    :    ENDIF
7412                                    ;  :    ENDIF
7413                                    ;  :    IF RXES=SIDE NOT RDY                                   [G]
7414                                    ;  :    :  THEN
7415                                    ;  :    :    DROP <SUC> AVAIL-SIDE #1/DRIVE #1
7416                                    ;  :    ENDIF
7417                                    ;  ENDIF
7418                                    ;  IF DROP_AVAIL_DRIVE NOT=0                                    [H]
7419                                    ;  :  THEN
7420                                    ;  :    CALL PRINT SYSTEM IDENT
7421                                    ;  :    SETUP PRINT 'DRIVE NOT READY-IS DISK IN DRIVE &''
7422                                    ;  :    CALL PRINT SYSTEM IDENT
7423                                    ;  :    CLEAR SIDE PRINT FLAG
7424                                    ;  :    CALL PRINT DRIVE IDENT
7425                                    ;  :    DROP THE DRIVE SET IN DROP_AVAIL_DRIVE IN <SUC> AVAIL
7426                                    ;  :    SETUP & PRINT 'DROPPED''
7427                                    ;  ENDIF
7428                                    ;ENDROUTINE
```

```
                                          ;---------------------------------------------------------------------------
                                          
        006040  005037  006264            DRVAVL: CLR     DROPAV              ;CLEAR DROP AVAIL DRIVE
        006044  004737  010076                    CALL    SSUCAD              ;CALL SET <SUC> ADDRESS & OFFSET
        006050  032761  000020  000622    IA2252: BIT     #DRV1,SOCMD(R1)     ;IF <SUC> COMMAND
        006056  001426                            BEQ     IE2252              ;HAS DRIVE #1 SELECTED, THEN
        006060  032777  000200  172406    IB2252: BIT     #DRVRDY,@RXDBAD     ;IF RXES
        006066  001012                            BNE     ID2252              ;DRIVE RDY NOT SET, THEN
        006070  032761  000002  000612    IC2252: BIT     #2,SOAV(R1)         ;IF <SUC> DRIVE #1
        006076  001406                            BEQ     ID2252              ;AVAILABLE, THEN
        006100  012737  000001  011224            MOV     #1,DRIVE            ;SET DRIVE=1 FOR PRINT
        006106  012737  000002  006264            MOV     #2,DROPAV           ;SET DROP AVAIL SIDE #0, DRIVE #!
        006114  032777  000002  172352    ID2252: BIT     #SIDRDY,@RXDBAD     ;IF RXES
        006122  001030                            BNE     IH2252              ;SIDE RDY NOT SET, THEN
        006124  042761  000010  000612            BIC     #10,SOAV(R1)        ;DROP SIDE #1, DRIVE #1
        006132  000424                            BR      IH2252              ;BR TO IF 'H'
        006134  032777  000200  172332    IE2252: BIT     #DRVRDY,@RXDBAD     ;IF RXES
        006142  001011                            BNE     IG2252              ;DRIVE RDY NOT SET, THEN
        006144  032761  000001  000612    IF2252: BIT     #1,SOAV(R1)         ;IF <SUC> DRIVE #0
        006152  001405                            BEQ     IG2252              ;AVAILABLE, THEN
        006154  005037  011224                    CLR     DRIVE               ;SET DRIVE=0
        006160  012737  000001  006264            MOV     #1,DROPAV           ;SET DROP AVAIL SIDE #0, DRIVE #0
        006166  032777  000002  172300    IG2252: BIT     #SIDRDY,@RXDBAD     ;IF RXES
        006174  001003                            BNE     IH2252              ;SIDE RDY NOT SET, THEN
        006176  042761  000004  000612            BIC     #4,SOAV(R1)         ;DROP SIDE #1, DRIVE #0
        006204  005737  006264            IH2252: TST     DROPAV              ;IF DROP AVAIL SIDE/DRIVE
        006210  001423                            BEQ     EH2252              ;SET, THEN
        006212  004737  011132                    CALL    PRTSYS              ;CALL PRINT SYSTEM IDENT
        006216  012704  016613                    MOV     #MSG55,R4           ;SET MSG-> 'DRIVE NOT RDY-IS DISK IN DRIVE & DOOR CLOSED
   (1)  006222  004737  013232                    CALL    TTOUT               ;PRINT MSG
        006226  004737  011132                    CALL    PRTSYS              ;CALL PRINT SYSTEM IDENT
        006232  005037  011230                    CLR     PTSIDF              ;CLEAR PRINT SIDE FLAG
        006236  004737  011154                    CALL    PRTDRV              ;CALL PRINT DRIVE/SIDE IDENT
        006242  043761  006264  000612            BIC     DROPAV,SOAV(R1)     ;DROP DRIVE SET -> IN <SUC> AVAIL
        006250  012704  015121                    MOV     #MSG33,R4           ;SET MSG-> 'DROPPED'
   (1)  006254  004737  013232                    CALL    TTOUT               ;PRINT MSG
        006260  000240                    EH2252: NOP                         ;
        006262  000207                    X2252:  RETURN                      ;RETURN
                                          ;---------------------------------------------------------------------------
        006264  000000                    DROPAV: 0                           ;DROP AVAIL DRIVES/SIDES CONTROL
                                          ;---------------------------------------------------------------------------
                                          ;
                                          ;
                                          ;       SYSTEM AVAILABLE WORD
                                          ;
                                          ;       +---+---+---+---+
                                          ;       ! 10! 4 ! 2 ! 1 !   <-- SIDE & DRIVE BIT AVAILABLE
                                          ;       +---+---+---+---+
                                          ;       ! 11! 10! 01! 00!   <-- SIDE & DRIVE CODE
                                          ;       +---+---+---+---+
                                          ;       ! SD! SD! SD! SD!
                                          ;                     ^^
                                          ;                     !!...DRIVE
                                          ;                     !....SIDE
```

```
                                .SBTTL  MODULE 2.2.5.3 - SET TEST BITS IN CSR & ESR

                                ;BEGINROUTINE (MOD 2.2.5.3 - SET TEST BITS IN CSR & ESR)
                                ;   SETUP CSR_COMPARE=DONE BIT
                                ;   SETUP CSR_COMPARE=DOUBLE DENSITY BIT
                                ;   GET <SUC> LAST COMMAND
                                ;   SAVE ALL READ/WRITE BITS
                                ;   SETUP CSR_COMPARE=SET ALL READ/WRITE BITS
                                ;   SETUP CSR_MASK=DON'T CARE BITS
                                ;   CLEAR ALL BUT SIDE BIT FROM <SUC> LAST COMMAND
                                ;   SETUP ESR_COMPARE=SIDE BIT AS SET FROM <SUC> LAST CMD
                                ;   IF <SUC> LAST COMMAND=DRIVE#1 SELECTED                    [A]
                                ;   :  THEN
                                ;   :     SETUP ESR_COMAPRE=DRIVE#1
                                ;   ENDIF
                                ;   IF <SUC> LAST COMMAND=DOUBLE DENSITY BIT SET              [B]
                                ;   :  THEN
                                ;   :     SETUP ESR_COMPARE=DRIVE DOUBLE DENSITY BIT SET
                                ;   ENDIF
                                ;   SETUP ESR_MASK=DON'T CARE BITS
                                ;ENDROUTINE
                                ;-------------------------------------------------------------------------

006266  012737  000040  000546 STSTBT: MOV     #DNBIT,CSRCMP   ;SET DONE BIT CK-CSR MODULE
006274  052737  004000  000546         BIS     #RX2BIT,CSRCMP  ;SET DOUBLE DENSITY BIT CK-CSR COMPARE
006302  016103  000622                 MOV     SOCMD(R1),R3    ;SET <SUC> COMMAND
006306  042703  176257                 BIC     #176257,R3      ;SAVE ALL R/W BITS
006312  050337  000546                 BIS     R3,CSRCMP       ;SET R/W BITS FOR-CSR COMPARE
006316  012737  070017  000544         MOV     #70017,CSRMSK   ;SET-CSR DON'T CARE BIT MASK
006324  016103  000622                 MOV     SOCMD(R1),R3    ;GET <SUC> COMMAND
006330  042703  176777                 BIC     #^CSIDE1,R3     ;CLEAR ALL BIT SIDE BIT
006334  010337  000552                 MOV     R3,ESRCMP       ;SET SIDE BIT (IF SELECTED)-ESR COMAPRE
006340  032761  000020  000622 IA2253: BIT     #DRV1,SOCMD(R1) ;IF DRIVE #1
006346  001403                 BEQ     IB2253          ;SELECTED IN COMMAND, THEN
006350  052737  000400  000552         BIS     #DRIVE1,ESRCMP  ;SET DRIVE #1 BIT-ESR COMPARE
006356  032761  000400  000622 IB2253: BIT     #DENBIT,SOCMD(R1);IF DOUBLE DENSITY
006364  001403                 BEQ     EB2253          ;SELECTED IN COMMAND, THEN
006366  052737  000040  000552         BIS     #DRVDEN,ESRCMP  ;SET DRIVE DENSITY BIT-ESR COMPARE
006374  012737  170000  000550 EB2253: MOV     #170000,ESRMSK  ;SET ESR DON'T CARE BIT MASK
006402  000240                 NOP                     ;
006404  000207         X2253:  RETURN                  ;RETURN
                                ;-------------------------------------------------------------------------
```

```
                                        .SBTTL  MODULE 2.2.5.4 - LOOPING MODULE
                                        ;-------------------------------------------------------------------

      006406  000240                    LOOP:   NOP                             ;
      006410  005737  000516                    TST     DBGFLG                  ;IF DEBUG FLAG
      006414  001464                            BEQ     X2254                   ;SET, THEN
      006416  032777  101000  172060            BIT     #101000,@SWR            ;IF HALT ON ERR OR LOOP ON ERR
      006424  001460                            BEQ     X2254                   ;SET, THEN
      006426  005037  000602                    CLR     HDRFLG                  ;CLEAR HEADER FLAG
      006432  012737  017417  000600            MOV     #MSGLP,TMSGAD           ;SET MODULE MESSAGE
      006440  016137  000622  000542            MOV     SOCMD(R1),CMD           ;GET <SUC> COMMAND
      006446  013737  000542  000540    LOOP1:  MOV     CMD,WDOT                ;PASS COMMAND TO OUTPUT MODULE
      006454  004737  010404                    CALL    CSRCHK                  ;CALL CSR CHECK
      006460  005700                            TST     R0                      ;IF RXCS ERROR
      006462  001416                            BEQ     LOOP2                   ;SET, THEN
      006464  004737  012262                    JSR     PC,BSINIT               ;CHECK SWR FOR BUS INITIALIZE
  (1) 006470  032777  001000  172006            BIT     #SW09,@SWR              ;CHECK FOR LOOP ON ERROR
  (1) 006476  001363                            BNE     LOOP1                   ;GO TO LOOP ERROR
  (1) 006500  004737  012240                    JSR     PC,ERRSET
  (2) 006504  000007                            .WORD   #7                      ;FATAL ERR # 7 - LOOPING MOD-CSR ERR
  (1) 006506  005777  171772                    TST     @SWR                    ;CHECK FOR HALT ON ERROR
  (1) 006512  100001                            BPL     1$                      ;HALT IF SET
  (1) 006514  000000                            HALT                            ;<LOOPING MOD-CSR ERR>
  (1) 006516  000240                    1$:     NOP
      006520  004737  010624            LOOP2:  CALL    ESRCHK                  ;CALL ESR CHECK
      006524  005700                            TST     R0                      ;IF RXES ERROR
      006526  001416                            BEQ     LOOP3                   ;SET, THEN
      006530  004737  012262                    JSR     PC,BSINIT               ;CHECK SWR FOR BUS INITIALIZE
  (1) 006534  032777  001000  171742            BIT     #SW09,@SWR              ;CHECK FOR LOOP ON ERROR
  (1) 006542  001341                            BNE     LOOP1                   ;GO TO LOOP ERROR
  (1) 006544  004737  012240                    JSR     PC,ERRSET
  (2) 006550  000010                            .WORD   #10                     ;FATAL ERR # 10 - LOOPING MOD-ESR ERR
  (1) 006552  005777  171726                    TST     @SWR                    ;CHECK FOR HALT ON ERROR
  (1) 006556  100001                            BPL     1$                      ;HALT IF SET
  (1) 006560  000000                            HALT                            ;<LOOPING MOD-ESR ERR>
  (1) 006562  000240                    1$:     NOP
      006564  000240                    LOOP3:  NOP                             ;
      006566  000207                    X2254:  RETURN                          ;RETURN
                                        ;-------------------------------------------------------------------
```

```
7549                                    .SBTTL  MODULE 2.3 - SYSTEM STATUS
7550
7551                                    ;BEGINROUTINE (MOD 2.3 - SYSTEM STATUS)
7552                                    ;  IF SYSTEM#0_AVAIL=0 & SYSTEM#0=SELECTED                          [A]
7553                                    ;  :  THEN
7554                                    ;  :    SET <SUC>=0
7555                                    ;  :    CALL DROP SYSTEM                                (MOD 2.3.2)
7556                                    ;  ENDIF
7557                                    ;  IF SYSTEM#1_AVAIL=0 & SYSTEM#1=SELECTED                          [B]
7558                                    ;  :  THEN
7559                                    ;  :    SET <SUC>=1
7560                                    ;  :    CALL DROP SYSTEM                                (MOD 2.3.2)
7561                                    ;  ENDIF
7562                                    ;  IF SYSTEM#0 & SYSTEM#1=NOT SELECTED                              [C]
7563                                    ;  :  THEN
7564                                    ;  :    SET SYSTEM FLAG
7565                                    ;  :    SET MSG 'NO SYSTEM AVAILABLE TO FORMATER''
7566                                    ;  :  ELSE
7567                                    ;  :    IF SYSTEM#1=SELECTED                                      [D]
7568                                    ;  :    :  THEN
7569                                    ;  :    :    IF SYSTEM#1_DONE=1                                   [E]
7570                                    ;  :    :    :  THEN
7571                                    ;  :    :    :    IF SYSTEM#0 SELECTED                           [F]
7572                                    ;  :    :    :    :  THEN
7573                                    ;  :    :    :    :    IF SYSTEM#0_DONE=1                         [G]
7574                                    ;  :    :    :    :    :  THEN
7575                                    ;  :    :    :    :    :    SET SYSTEM FLAG
7576                                    ;  :    :    :    :    :    CALL PRINT SYSTEM STATUS (MOD 2.3.1)
7577                                    ;  :    :    :    :    :    CLEAR SYSTEM#0 & SYSTEM#1 DONE
7578                                    ;  :    :    :    :    :    SETUP FORMATTER MSG
7579                                    ;  :    :    :    :    :  ELSE
7580                                    ;  :    :    :    :    :    CALL SYSTEM TIMEOUT CHECK (MOD 2.3.3)
7581                                    ;  :    :    :    :    :  ENDIF
7582                                    ;  :    :    :    :    ELSE
7583                                    ;  :    :    :    :    :  SET SYSTEM FLAG
7584                                    ;  :    :    :    :    :  CALL PRINT SYSTEM STATUS      (MOD 2.3.1)
7585                                    ;  :    :    :    :    :  CLEAR SYSTEM#1 DONE
7586                                    ;  :    :    :    :    :  SETUP FORMATTER MSG
7587                                    ;  :    :    :    :    ENDIF
7588                                    ;  :    :    :    :  ELSE
7589                                    ;  :    :    :    :    CALL SYSTEM TIMEOUT CHECK          (MOD 2.3.3)
7590                                    ;  :    :    :    ENDIF
7591                                    ;  :    :    :  ELSE
7592                                    ;  :    :    :    IF SYSTEM#0_DONE=1                                [H]
7593                                    ;  :    :    :    :  THEN
7594                                    ;  :    :    :    :    SET SYSTEM FLAG
7595                                    ;  :    :    :    :    CALL PRINT SYSTEM STATUS          (MOD 2.3.1)
7596                                    ;  :    :    :    :    CLEAR SYSTEM#0 DONE
7597                                    ;  :    :    :    :    SETUP FORMATTER MSG
7598                                    ;  :    :    :    :  ELSE
7599                                    ;  :    :    :    :    CALL SYSTEM TIMEOUT CHECK          (MOD 2.3.3)
7600                                    ;  :    :    :    ENDIF
7601                                    ;  :    :  ENDIF
7602                                    ;  ENDIF
7603                                    ;ENDROUTINE
```

```
                                          ;------------------------------------------------------------------------

006570  000240                            SYSTAT: NOP                          ;
006572  005737  000612                    IA23:   TST     S0AV                 ;IF SYSTEM#0 AVAIL
006576  001010                                    BNE     IB23                 ;EQUALS ZERO AND
006600  032777  000001  171676                    BIT     #1,@SWR              ;SYSTEM#0
006606  001404                                    BEQ     IB23                 ;SELECTED, THEN
006610  005037  000610                            CLR     SUC                  ;SET <SUC>=0
006614  004737  007264                            CALL    SYSDRP               ;CALL SYSTEM DROP       (MOD 2.3.2)
006620  005737  000614                    IB23:   TST     S1AV                 ;IF SYSTEM#1 AVAIL
006624  001011                                    BNE     IC23                 ;EQUALS ZERO AND
006626  032777  000002  171650                    BIT     #2,@SWR              ;SYSTEM#1
006634  001405                                    BEQ     IC23                 ;SELECTED THEN
006636  012737  000001  000610                    MOV     #1,SUC               ;SET <SUC>=1
006644  004737  007264                            CALL    SYSDRP               ;CALL SYSTEM DROP       (MOD 2.3.2)
006650  032777  000003  171626    IC23:   BIT     #3,@SWR              ;IF SYS#0 & SYS#1
006656  001006                                    BNE     ID23                 ;NOT SELECTED OR DESELECTED, THEN
006660  005237  000574                            INC     SYSFLG               ;SET SYSTEM FLAG
006664  012737  014206  000572                    MOV     #MSG8,SYSMSG         ;SET SYSTEM MSG 'NO SYSTEMS TO FORMAT''
006672  000471                                    BR      EC23                 ;BR TO END 'C'
006674  032777  000002  171602    ID23:   BIT     #2,@SWR              ;IF SYSTEM#1
006702  001446                                    BEQ     IH23                 ;SELECTED, THEN
006704  005737  000630                    IE23:   TST     S1DN                 ;IF SYSTEM#1 DONE
006710  001440                                    BEQ     LE23                 ;SET, THEN
006712  032777  000001  171564    IF23:   BIT     #1,@SWR              ;IF SYSTEM#0
006720  001422                                    BEQ     LF23                 ;SELECTED, THEN
006722  005737  000626                    IG23:   TST     S0DN                 ;IF SYSTEM#0 DONE
006726  001414                                    BEQ     LG23                 ;SET, THEN
006730  005237  000574                            INC     SYSFLG               ;SET SYSTEM FLAG
006734  004737  007062                            CALL    PTSYST               ;CALL PRINT SYSTEM STATUS       (MOD 2.3.1)
006740  005037  000626                            CLR     S0DN                 ;CLEAR SYSTEM#0 DONE
006744  005037  000630                            CLR     S1DN                 ;CLEAR SYSTEM#1 DONE
006750  012737  014246  000572                    MOV     #MSG9,SYSMSG         ;SET SYSTEM MSG 'FORMAT COMPLETED''
006756  000437                                    BR      EC23                 ;BR TO END 'C'
006760  004737  007342                    LG23:   CALL    SYTMCK               ;CALL SYSTEM TIMEOUT CHECK      (MOD 2.3.3)
006764  000434                                    BR      EC23                 ;BR TO END 'C'
006766  005237  000574                    LF23:   INC     SYSFLG               ;SET SYSTEM FLAG
006772  004737  007062                            CALL    PTSYST               ;CALL PRINT SYSTEM STATUS       (MOD 2.3.1)
006776  005037  000630                            CLR     S1DN                 ;CLEAR SYSTEM#1 DONE
007002  012737  014246  000572                    MOV     #MSG9,SYSMSG         ;SET SYSTEM MSG 'FORMAT COMPLETED''
007010  000422                                    BR      EC23                 ;BR TO END 'C'
007012  004737  007342                    LE23:   CALL    SYTMCK               ;CALL SYSTEM TIMEOUT CHECK      (MOD 2.3.3)
007016  000417                                    BR      EC23                 ;BR TO END 'C'
007020  005737  000626                    IH23:   TST     S0DN                 ;IF SYSTEM#0 DONE
007024  001412                                    BEQ     LH23                 ;SET, THEN
007026  005237  000574                            INC     SYSFLG               ;SET SYSTEM FLAG
007032  004737  007062                            CALL    PTSYST               ;CALL PRINT SYSTEM STATUS       (MOD 2.3.1)
007036  005037  000626                            CLR     S0DN                 ;CLEAR SYSTEM#0 DONE
007042  012737  014246  000572                    MOV     #MSG9,SYSMSG         ;SET SYSTEM MSG 'FORMAT COMPLETED''
007050  000402                                    BR      EC23                 ;BR TO END 'C'
007052  004737  007342                    LH23:   CALL    SYTMCK               ;CALL SYSTEM TIMEOUT CHECK      (MOD 2.3.3)
007056  000240                            EC23:   NOP                          ;
007060  000207                            X23:    RETURN                       ;RETURN
                                          ;------------------------------------------------------------------------
```

```
 7660                                   .SBTTL  MODULE 2.3.1 - PRINT SYSTEM STATUS
 7661
 7662                                   ;BEGINROUTINE (MOD 2.3.1 - PRINT SYSTEM STATUS)
 7663                                   ;  SET <SUC>=0
 7664                                   ;  BEGINDO
 7665                                   ;  :  INITIALIZE DRIVE_PTR=1, DRIVE_CTR=0                        [A]
 7666                                   ;  :  IF <SUC> DONE=SET                                          [B]
 7667                                   ;  :  :  THEN
 7668                                   ;  :  :    PRINT ''SYSTEM: <SUC> FORMAT DONE ON FOLLOWING:''
 7669                                   ;  :  :    BEGINDO                                               [C]
 7670                                   ;  :  :    :  IF <SUC> AVAIL=BIT SET AT SYSTEM AVAIL PTR         [D]
 7671                                   ;  :  :    :  :  THEN
 7672                                   ;  :  :    :  :    IF <SUC> AVAIL=EITHER SIDE AVAIL            [E]
 7673                                   ;  :  :    :  :    :  THEN
 7674                                   ;  :  :    :  :    :    SET SIDE FLAG
 7675                                   ;  :  :    :  :    :  ELSE
 7676                                   ;  :  :    :  :    :    CLEAR SIDE FLAG
 7677                                   ;  :  :    :  :    ENDIF
 7678                                   ;  :  :    :  :    SET DRIVE #
 7679                                   ;  :  :    :  :    SET SIDE #
 7680                                   ;  :  :    :  :    CALL PRINT DRIVES IDENT
 7681                                   ;  :  :    :  ENDIF
 7682                                   ;  :  :    :  INCREMENT DRIVE COUNTER
 7683                                   ;  :  :    :  SHIFT LEFT DRIVE POINTER
 7684                                   ;  :  :    :  DO UNTIL DRIVE_COUNTER=1
 7685                                   ;  :  :    ENDDO
 7686                                   ;  :  ENDIF
 7687                                   ;  :  INCREMENT <SUC> '
 7688                                   ;  :  DO UNTIL <SUC>=2
 7689                                   ;  ENDDO
 7690                                   ;ENDROUTINE
```

```
 7692                                   ;-----------------------------------------------------------------------
 7693
 7694  007062                           PTSYST:
  (1)  007062  012704  014452                    MOV     #MSG14,R4       ;SET MSG-> 'FORMAT DONE ON FOLLOWING'
  (1)  007066  004737  013232                    CALL    TTOUT           ;PRINT MSG
 7695  007072  005037  000610                    CLR     SUC             ;SET <SUC>=0
 7696  007076  012737  000001  007262  BA231:    MOV     #1,DRVPTR       ;INITIALIZE DRIVE POINTER=1
 7697  007104  005037  007260                    CLR     DRVCNT          ;INITIALIZE DRIVE COUNTER=0
 7698  007110  004737  010126                    CALL    SSUCOF          ;CALL SET <SUC> OFFSET
 7699  007114  005761  000626          IB231:    TST     SODN(R1)        ;IF <SUC> DONE
 7700  007120  001450                            BEQ     EB231           ;SET, THEN
 7701  007122  004737  011132                    CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
 7702  007126  000240                  BC231:    NOP                     ;
 7703  007130  033761  007262  000612  ID231:    BIT     DRVPTR,SOAV(R1) ;IF DRIVE POINTER=<SUC> AVAIL BIT
 7704  007136  001431                            BEQ     ED231           ;SET, THEN
 7705  007140  032761  000014  000612  IE231:    BIT     #14,SOAV(R1)    ;IF EITHER SIDE IN <SUC> AVAIL
 7706  007146  001403                            BEQ     LE231           ;SET, THEN
 7707  007150  005237  011230                    INC     PTSIDF          ;SET PRINT SIDE FLAG
 7708  007154  000402                            BR      EE231           ;BR TO END 'E'
 7709  007156  005037  011230          LE231:    CLR     PTSIDF          ;CLEAR PRINT SIDE FLAG
 7710  007162  013737  007260  011224  EE231:    MOV     DRVCNT,DRIVE    ;GET DRIVE COUNT
 7711  007170  042737  000002  011224            BIC     #2,DRIVE        ;CLEAR SIDE BIT (DRIVE=DRIVE#) PRINT
 7712  007176  013737  007260  011226            MOV     DRVCNT,SIDE     ;GET DRIVE COUNT
 7713  007204  006237  011226                    ASR     SIDE            ;SHIFT IT RIGHT TO SHOW SIDE BIT
 7714  007210  042737  177776  011226            BIC     #177776,SIDE    ;CLEAR ANY JUNK BITS
 7715  007216  004737  011154                    CALL    PRTDRV          ;CALL PRINT DRIVE IDENT
 7716  007222  005237  007260          ED231:    INC     DRVCNT          ;INCREMENT DRIVE COUNTER
 7717  007226  006337  007262                    ASL     DRVPTR          ;SHIFT LEFT DRIVE POINTER
 7718  007232  022737  000004  007260  UC231:    CMP     #4,DRVCNT       ;DO UNTIL DRIVE COUNT
 7719  007240  001332                            BNE     BC231           ;EQUALS 4
 7720  007242  005237  000610          EB231:    INC     SUC             ;INCREMENT <SUC> TO NEXT SYSTEM
 7721  007246  022737  000002  000610  UA231:    CMP     #2,SUC          ;DO UNTIL <SUC>
 7722  007254  001310                            BNE     BA231           ;EQUALS 2
 7723  007256  000207                  X231:     RETURN                  ;RETURN
 7724                                   ;-----------------------------------------------------------------------
 7725  007260  000000                  DRVCNT:   0                       ;DRIVE COUNTER
 7726  007262  000000                  DRVPTR:   0                       ;DRIVE POINTER
 7727                                   ;-----------------------------------------------------------------------
```

```
 7729                                   .SBTTL  MODULE 2.3.2 - SYSTEM DROP
 7730
 7731                                   ;BEGINROUTINE (MOD 2.3.2 SYSTEM DROP)
 7732                                   ;     SETUP & PRINT MSG ''SYSTEM''
 7733                                   ;     SETUP & PRINT SYSTEM#
 7734                                   ;     IF SUC = 0
 7735                                   ;     :     THEN
 7736                                   ;     :        DESELECT SYSTEM#0
 7737                                   ;     :     ELSE
 7738                                   ;     :        DESELECT SYSTEM#1
 7739                                   ;     ENDIF
 7740                                   ;     SETUP & PRINT MSG 'NO DRIVES AVAIL - DROPPED''
 7741                                   ;ENDROUTINE
 7742
 7743                                   ;-------------------------------------------------------------------------
 7744
 7745  007264  012704  016425           SYSDRP: MOV     #MSG51,R4       ;SETUP MSG ''SYSTEM:''
 7746  007270  004737  013232                   CALL    TTOUT           ;PRINT IT!
 7747  007274  013703  000610                   MOV     SUC,R3          ;SET SYSTE #
 7748  007300  004737  013346                   CALL    OCTP            ;PRINT IT!
 7749  007304  005737  000610           IA232:  TST     SUC             ;IF SYSTEM UNDER CONTROL
 7750  007310  001004                           BNE     LA232           ;IS ZERO, THEN
 7751  007312  042777  000001  171164           BIC     #1,@SWR         ;DESELECT SYSTEM ZERO
 7752  007320  000403                           BR      EA232           ;BR TO ENDIF 'A'
 7753  007322  042777  000002  171154   LA232:  BIC     #2,@SWR         ;ELSE DESELECT SYSTEM ONE
 7754  007330  012704  016475           EA232:  MOV     #MSG53,R4       ;SETUP MSG 'DROPPED FROM FORMATTER''
 7755  007334  004737  013232                   CALL    TTOUT           ;PRINT IT!
 7756  007340  000207                   X232:   RETURN                  ;RETURN
 7757                                   ;-------------------------------------------------------------------------
```

```
                                .SBTTL  MODULE 2.3.3 - SYSTEM TIMEOUT CHECK

                                ;BEGINROUTINE   (MOD 2.3.3 - SYSTEM TIMEOUT CHECK)
                                ;  IF SYSTEM_ERROR NOT=0                                [A]
                                ;  :  THEN
                                ;  :    IF SYSTEM ERROR=SYS#0 TIME OUT ERROR          [B]
                                ;  :    :  THEN
                                ;  :    :    SET <SUC>=0
                                ;  :    :    CALL SYSTEM IDENT
                                ;  :    :    SETUP & PRINT TIME OUT ERROR
                                ;  :    :    CLEAR SYS #0 AVAILABLE WORD
                                ;  :    ENDIF
                                ;  :    IF SYSTEM ERROR=SYS#1 TIMEOUT ERROR           [C]
                                ;  :    :  THEN
                                ;  :    :    SET <SUC>=1
                                ;  :    :    CALL SYSTEM IDENT
                                ;  :    :    SETUP & PRINT TIMEOUT ERROR
                                ;  :    :    CLEAR SYS#1 AVAILABLE WORD
                                ;  :    ENDIF
                                ;  ENDIF
                                ;ENDROUTINE
                                ;-------------------------------------------------------------------------

007342  000240          SYTMCK: NOP                     ;
007344  005737  000606  IA233:  TST     SYSERR          ;IF SYSTEM ERROR
007350  001446                  BEQ     EA233           ;NOT = 0, THEN
007352  032737  000010  000606  IB233:  BIT     #10,SYSERR      ;IF SYSTEM ERROR
007360  001417                  BEQ     IC233           ;EQUALS SYS#0 TIME OUT ERR
007362  005037  000610          CLR     SUC             ;SET <SUC> = 0
007366  004737  011132          CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
007372  012704  014272          MOV     #MSG10,R4       ;SET MSG-> ''INTERRUPT DID NOT OCCUR''
(1)     007376  004737  013232  CALL    TTOUT           ;PRINT MSG
007402  012704  015702          MOV     #MSG41,R4       ;SET MSG-> ''-TIME OUT ERROR''
(1)     007406  004737  013232  CALL    TTOUT           ;PRINT MSG
007412  005037  000612          CLR     S0AV            ;CLEAR SYS#0 AVIALABLE
007416  000423                  BR      EA233           ;BR TO END 'A'
007420  032737  000020  000606  IC233:  BIT     #20,SYSERR      ;IF SYSTEM ERROR
007426  001417                  BEQ     EA233           ;EQUALS SYS#1 TIME OUT ERR , THEN
007430  012737  000001  000610          MOV     #1,SUC          ;SET <SUC> = 1
007436  004737  011132          CALL    PRTSYS          ;CALL PRINT SYSTEM IDENT
007442  012704  014272          MOV     #MSG10,R4       ;SET MSG-> ''INTERRUPT DID NOT OCCUR''
(1)     007446  004737  013232  CALL    TTOUT           ;PRINT MSG
007452  012704  015702          MOV     #MSG41,R4       ;SET MSG-> ''-TIME OUT ERROR''
(1)     007456  004737  013232  CALL    TTOUT           ;PRINT MSG
007462  005037  000614          CLR     S1AV            ;CLEAR SYS#1 AVIALABLE
007466  005037  000606  EA233:  CLR     SYSERR          ;CLEAR SYSTEM ERROR
007472  000207                  RETURN                  ;RETURN
                                ;-------------------------------------------------------------------------
```

```
                                          .SBTTL  MODULE 2.4 - WATCH DOG SYSTEM#0

                                          ;------------------------------------------------------------------
007474  005737  007670                    WATCH0: TST     WATIN0          ;IF WATCH DOG INIT0
007500  001406                                    BEQ     1$              ;NOT = 0, THEN
007502  005037  007670                            CLR     WATIN0          ;CLEAR WATCHDOG INIT #0 FLAG
007506  005037  007566                            CLR     WCNTR0          ;CLEAR WATCH DOG COUNTER #0
007512  005037  007570                            CLR     WMLT0           ;CLEAR WATCH DOG MULTIPLIER #0
007516  005237  007566                    1$:     INC     WCNTR0          ;INCREMENT WATCH DOG COUNTER #0
007522  023737  007566  007674                    CMP     WCNTR0,WCNTMX   ;IF COUNTER
007530  103415                                    BLO     XWAT0           ;IS = TO COUNT MAX
007532  005037  007566                            CLR     WCNTR0          ;CLEAR WATCH DOG COUNTER #0
007536  005237  007570                            INC     WMLT0           ;INCREMENT WATCH DOG MULTIPIER #0
007542  023737  007570  000556                    CMP     WMLT0,S0WMLT    ;IF MULTIPLIER
007550  103405                                    BLO     XWAT0           ;IS = TO MULTIPLIER MAX
007552  005037  007570                            CLR     WMLT0           ;CLEAR WATCH DOG MULTIPLIER #0
007556  052737  000010  000606                    BIS     #10,SYSERR      ;SET SYSTEM ERR = TIME OUT
007564  000207                            XWAT0:  RETURN                  ;RETURN

                                          ;------------------------------------------------------------------
007566  000000                            WCNTR0: 0                       ;WATCH DOG COUNTER #0
007570  000000                            WMLT0:  0                       ;WATCH DOG MULTIPLIER #0
                                          ;------------------------------------------------------------------



                                          .SBTTL  MODULE 2.5 - WATCH DOG SYSTEM#1

                                          ;------------------------------------------------------------------
007572  005737  007672                    WATCH1: TST     WATIN1          ;IF WATCH DOG INIT1
007576  001406                                    BEQ     1$              ;NOT = 1, THEN
007600  005037  007672                            CLR     WATIN1          ;CLEAR WATCH DOG INIT #1 FLAG
007604  005037  007664                            CLR     WCNTR1          ;CLEAR WATCH DOG COUNTER #1
007610  005037  007666                            CLR     WMLT1           ;CLEAR WATCH DOG MULTIPLIER #1
007614  005237  007664                    1$:     INC     WCNTR1          ;INCREMENT WATCH DOG COUNTER #1
007620  023737  007664  007674                    CMP     WCNTR1,WCNTMX   ;IF COUNTER
007626  103415                                    BLO     XWAT1           ;IS = TO COUNT MAX
007630  005037  007664                            CLR     WCNTR1          ;CLEAR WATCH DOG COUNTER #1
007634  005237  007666                            INC     WMLT1           ;INCREMENT WATCH DOG MULTIPIER #1
007640  023737  007666  000560                    CMP     WMLT1,S1WMLT    ;IF MULTIPLIER
007646  103405                                    BLO     XWAT1           ;IS = TO MULTIPLIER MAX
007650  005037  007666                            CLR     WMLT1           ;CLEAR WATCH DOG MULTIPLIER #1
007654  052737  000020  000606                    BIS     #20,SYSERR      ;SET SYSTEM ERR = TIME OUT
007662  000207                            XWAT1:  RETURN                  ;RETURN

                                          ;------------------------------------------------------------------
007664  000000                            WCNTR1: 0                       ;WATCH DOG COUNTER #1
007666  000000                            WMLT1:  0                       ;WATCH DOG MULTIPLIER #1

                                          ;------------------------------------------------------------------
007670  000000                            WATIN0: 0                       ;WATCH DOG INITIALIZE FLAG #0
007672  000000                            WATIN1: 0                       ;WATCH DOG INITIALIZE FLAG #1
007674  177777                            WCNTMX: -1                      ;WATCH DOGS MAX COUNT LIMIT
                                          ;------------------------------------------------------------------

```

```
                        .SBTTL  MODULE 3.0 - OUTPUT SYSTEM DONE

                        ;BEGINROUTINE (MOD 3.0 - OUTPUT SYSTEM DONE)
                        ;  SETUP & PRINT ''DO YOU WANT TO FORMAT MORE DISKETTES?''
                        ;  CALL GET ANSWER
                        ;  IF ANSWER=YES                                                    [A]
                        ;  :  THEN
                        ;  :    SETUP & PRINT ''REMOVE FORMATTED DISKETTS & INSERT DISKETTS TO BE FORMATTED''
                        ;  :    SETUP & PRINT ''-->TYPE ''CR'' WHEN READY''
                        ;  :    BGNDO                                                      [B]
                        ;  :    :  GET OPERATOR RESPONSE-CALL TTY INPUT
                        ;  :    :  DO UNTIL TTY CTR=''CR''
                        ;  :    ENDDO
                        ;  :  ELSE
                        ;  :    SETUP & PRINT ''FORMATTER DONE-RESTART MONITOR, UPDATE PROGRAM-->TYPE CTRL-C TO S
                        ;  :    SET SYSTEM FLAG=IDLE
                        ;  :    ENABLE TTY INTERRUPTS & SET TTY GO BIT
                        ;  ENDIF
                        ;ENDROUTINE

                        ;-------------------------------------------------------------------------

007676                  OTSYDN:
(1) 007676 012704 016703        MOV     #MSG56,R4       ;SET MSG-> DO YOU WANT TO FORMAT MORE DISKETTES?
(1) 007702 004737 013232        CALL    TTOUT           ;PRINT MSG
007706 004737 012624            CALL    GETANS          ;CALL GET OPERATOR ANSWER
007712 122737 000131 000530 IA30: CMPB  #'Y,ANSWER      ;IF ANSWER
007720 001022                   BNE     LA30            ;IS YES, THEN
007722 012704 016766            MOV     #MSG57,R4       ;SET MSG-> REMOVE FORMATTED DISKETTES & INSERT DISKETTES
(1) 007726 004737 013232        CALL    TTOUT           ;PRINT MSG
007732 012704 017065            MOV     #MSG58,R4       ;SET MSG-> TYPE ''CR'' WHEN READY
(1) 007736 004737 013232        CALL    TTOUT           ;PRINT MSG
007742 004737 013156        BB30: CALL  TTIN            ;CALL TTY INPUT
007746 122737 000015 000526     CMPB    #15,TIB         ;DO UNTIL
007754 001372                   BNE     BB30            ;CHARACTER=''CR''
007756 012737 000002 000574     MOV     #2,SYSFLG       ;SET SYSTEM FLAG=RESTART
007764 000414                   BR      X30             ;BR TO MODULE EXIT
007766                  LA30:
(1) 007766 012704 017117        MOV     #MSG59,R4       ;SET MSG-> FORMATTER DONE-RESTART MONITOR, UPDATE PROGRA
(1) 007772 004737 013232        CALL    TTOUT           ;PRINT MSG
007776 012737 000100 000574     MOV     #100,SYSFLG     ;SET SYSTEM FLAG=IDLE
010004 052777 000100 170266     BIS     #BIT6,@TKS      ;SET TTY KEYBOARD INTERRUPT BIT
010012 005277 170262            INC     @TKS            ;SET TTY KEYBOARD GO BIT
010016 000207               X30: RETURN                 ;RETURN
                        ;-------------------------------------------------------------------------
```

```
                                        .SBTTL  MODULE U.PRIHI - SET PROCESSOR PRIORITY HIGH
                                        ;---------------------------------------------------------------
010020  012737  000340  010074  PRIHI:  MOV     #PR7,NEWPRI     ;SETUP FOR PROCESSOR PRI LEV=7
010026  004737  010050                  CALL    SETPRI          ;SET PROCESSOR FLAG
010032  000207                          RETURN                  ;RETURN
                                        ;---------------------------------------------------------------
                                        .SBTTL  MODULE U.PRILO - SET PROCESSOR PRIORITY LOW
                                        ;---------------------------------------------------------------
010034  012737  000000  010074  PRILO:  MOV     #PR0,NEWPRI     ;SETUP FOR PROCESSOR PRI LEV=0
010042  004737  010050                  CALL    SETPRI          ;SET PROCESSOR PRI
010046  000207                          RETURN                  ;RETURN
                                        ;---------------------------------------------------------------
                                        .SBTTL  MODULE U.PROPRI - SET PROCESSOR PRI
                                        ;---------------------------------------------------------------
010050  005737  000520          SETPRI: TST     LSIFLG          ;IF PROCESSOR IS
010054  001403                          BEQ     1$              ;LSI, THEN
010056  106437  010074                  MTPS    NEWPRI          ;SET PROCESSOR PRI
010062  000403                          BR      SETPIX          ;BR TO END
010064  013777  010074  167704  1$:     MOV     NEWPRI,@PSW     ;SET PROCESSOR PRI
010072  000207          SETPIX: RTS     PC                      ;RETURN
                                        ;---------------------------------------------------------------
010074  000000          NEWPRI: 0                               ;NEW PROCESSOR PRIORITY
                                        ;---------------------------------------------------------------
                                        .SBTTL  MODULE U.SADR   - SET SYSTEM BUS ADDRESS
                                        ;---------------------------------------------------------------
010076  004737  010126          SSUCAD: CALL    SSUCOF          ;CALL SET <SUC> OFFSET
010102  016137  000314  000472          MOV     REGS0(R1),RXCSAD;SET NEW SYSTEM ADDRESS
010110  016137  000314  000474          MOV     REGS0(R1),RXDBAD;SET NEW SYSTEM DB ADDR
010116  062737  000002  000474          ADD     #2,RXDBAD       ;BUMP ADDRESS
010124  000207                          RETURN                  ;RETURN
                                        ;---------------------------------------------------------------
                                        .SBTTL  MODULE U.SUCO   - SET SYSTEM UNDER CONTROL OFFSET
                                        ;---------------------------------------------------------------
010126  013701  000610          SSUCOF: MOV     SUC,R1          ;GET SYSTEM UNDER CONTROL
010132  006301                          ASL     R1              ;DOUBLE IT! FOR WORD OFFSET ADDRESSING
010134  000207                          RETURN                  ;RETURN
                                        ;---------------------------------------------------------------
```

```
 7941                                   .SBTTL  MODULE U.DL - DELAY FOR ''TR'' OR ''DONE''
 7942                                   ;-------------------------------------------------------------------
 7943  010136  013704  010222           DELAY:  MOV     RYDX,R4          ;SET READY DELAY MULT
 7944  010142  013703  010224           BDAUDL: MOV     RYDLY,R3         ;SET READY DELAY
 7945  010146  033777  010226  000054   BDBUDL: BIT     RDYWD,@CSRADR    ;IF READY
 7946  010154  001021                           BNE     XUDL             ;EQUAL TO ''1'' ,THEN BR TO EXIT
 7947  010156  005303                           DEC     R3               ;ELSE DECREMENT DELAY
 7948  010160  001372                           BNE     BDBUDL           ;DO UNTIL R3=0
 7949  010162  012737  000007  000524           MOV     #7,TOB           ;SET UP BELL
 7950  010170  004737  013700                   JSR     PC,TOG           ;PRINT BELL
 7951  010174  005304                           DEC     R4               ;DECREMENT DELAY MULT.
 7952  010176  001361                           BNE     BDAUDL           ;DO UNTIL R4=0
 7953  010200  052737  040000  000606           BIS     #40000,SYSERR    ;SET TIME OUT ERR
 7954  010206  013737  010226  010402           MOV     RDYWD,TOERBT     ;SET T.O. WORD
 7955  010214  004737  010232                   JSR     PC,TOPRT         ;GO PRINT TIMEOUT ERRORS
 7956  010220  000207                   XUDL:   RTS     PC               ;RETURN TO CALLING MOD
 7957                                   ;-------------------------------------------------------------------
 7958  010222  000015                   RYDX:   15                       ;READY MULTIPLIER
 7959  010224  100000                   RYDLY:  100000                   ;READY DELAY
 7960  010226  000000                   RDYWD:  0                        ;READY WORD - TEST FOR DEVICE READY
 7961  010230  000000                   CSRADR: 0                        ;C&S REG OF UNIT- WAITING FOR
 7962                                   ;--------------- END MODULE ----------------------------------------
 7963
 7964                                   .SBTTL  TIME OUT ERROR PRINT
 7965                                   ;-------------------------------------------------------------------
 7966  010232  032777  020000  170244   TOPRT:  BIT     #BIT13,@SWR      ;IF INHIBIT ERROR RESULTS
```

```
010240  001057                           BNE     TOPRTX          ;IS NOT SET, THEN
010242  004737  011042                   JSR     PC,TSHDCK       ;CALL TEST HEADER CHECK
010246  032737  000040  010402  1$:      BIT     #DNBIT,TOERBT   ;IF READY WORD WAS SET TO
010254  001405                           BEQ     2$              ;TEST DONE BIT, THEN
010256  012704  015031                   MOV     #MSG30,R4       ;SET DONE NOT SET MSG
010262  004737  013232                   JSR     PC,TTOUT        ;PRINT MSG
010266  000404                           BR      3$              ;GO PRINT MSG
010270  012704  015055          2$:      MOV     #MSG31,R4       ;SET TR BIT NOT SET MSG
010274  004737  013232                   JSR     PC,TTOUT        ;PRINT MSG
010300  012704  015702          3$:      MOV     #MSG41,R4       ;SET TIME OUT MSG
010304  004737  013232                   JSR     PC,TTOUT        ;PRINT MSG
010310  032777  040000  170166           BIT     #SW14,@SWR      ;IF EXTENDED ERROR REPORTS
010316  001430                           BEQ     TOPRTX          ;SET, THEN
010320  012704  013742                   MOV     #MSG1,R4        ;SET REG MSG
010324  004737  013232                   JSR     PC,TTOUT        ;PRINT MSG
010330  017703  170136                   MOV     @RXCSAD,R3      ;SET TO PRINT RXCS
010334  004737  013334                   JSR     PC,OCTPE        ;PRINT
010340  012704  014564                   MOV     #MSG21,R4       ;SET SPACES MSG
010344  004737  013232                   JSR     PC,TTOUT        ;PRINT
010350  017703  170120                   MOV     @RXDBAD,R3      ;SET TO PRINT RXES
010354  004737  013334                   JSR     PC,OCTPE        ;PRINT
010360  012704  014564                   MOV     #MSG21,R4       ;SET SPACES
010364  004737  013232                   JSR     PC,TTOUT        ;PRINT
010370  013703  000542                   MOV     CMD,R3          ;SET COMMAND
010374  004737  013346                   JSR     PC,OCTP         ;PRINT COMMAND
010400  000207                  TOPRTX:  RTS     PC              ;RETURN
                                ;-----------------------------------------------------------------
010402  000000                  TOERBT:  0
                                ;-----------------------------------------------------------------
```

```
 7998                                   .SBTTL  RXCS ERROR CHECK
 7999                                   ;-------------------------------------------------------------------------
 8000
 8001  010404  017701  170062          CSRCHK: MOV     @RXCSAD,R1      ;GET RXCS
 8002  010410  043701  000544                  BIC     CSRMSK,R1       ;MASK OFF BITS DON'T CARE ABOUT
 8003  010414  023701  000546                  CMP     CSRCMP,R1       ;IF RXCS CONTAINS
 8004  010420  001471                        BEQ     5$              ;ERRORS, THEN
 8005  010422  010137  011540          1$:     MOV     R1,BADWRD       ;SET BAD WORD
 8006  010426  013737  000546  011536        MOV     CSRCMP,CMPWRD   ;SET COMPARE WORD
 8007  010434  012737  011544  011542        MOV     #CMSGTB,TABADR  ;SET MSG TABLE ADDRESS
 8008  010442  012737  000004  011534        MOV     #4,BITOFF       ;SET # BITS TO OFFSET WORD
 8009  010450  012737  000014  011532        MOV     #12.,BITLIM     ;SET # BITS TO CHECK
 8010  010456  032777  020000  170020        BIT     #SW13,@SWR      ;IF INHIBIT ERROR REPORTS
 8011  010464  001044                        BNE     4$              ;NOT SET, THEN
 8012  010466  004737  011042                CALL    TSHDCK          ;CALL TEST HEADER CHECK
 8013  010472  012704  014722          2$:     MOV     #MSG26,R4       ;SET RXCS MSG HEADER
 8014  010476  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
 8015  010502  032777  040000  167774        BIT     #SW14,@SWR      ;IF EXTENDED ERROR REPORTS
 8016  010510  001430                        BEQ     3$              ;SET, THEN
 8017  010512  012704  014411                MOV     #MSG13,R4       ;SET REG FORMAT MSG
 8018  010516  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
 8019  010522  013703  000546                MOV     CSRCMP,R3       ;SET GOOD RXCS
 8020  010526  004737  013334                JSR     PC,OCTPE        ;PRINT GOOD RXCS
 8021  010532  012704  014564                MOV     #MSG21,R4       ;SET SPACES MSG
 8022  010536  004737  013232                JSR     PC,TTOUT        ;PRINT SPACES
 8023  010542  013703  011540                MOV     BADWRD,R3       ;SET COMPARED RXCS
 8024  010546  004737  013334                JSR     PC,OCTPE        ;PRINT COMPARED RXCS
 8025  010552  012704  014564                MOV     #MSG21,R4       ;SET SPACES MSG
 8026  010556  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
 8027  010562  017703  167704                MOV     @RXCSAD,R3      ;SET FULL RXCS
 8028  010566  004737  013334                JSR     PC,OCTPE        ;PRINT FULL RXCS
 8029  010572  004737  011314          3$:     JSR     PC,CKBITS       ;REPORT BAD BITS
 8030  010576  012700  177777          4$:     MOV     #-1,R0          ;SET ERR
 8031  010602  000405                        BR      CSRCKX          ;BR TO END
 8032  010604  005000                5$:     CLR     R0              ;CLEAR ERRORS
 8033  010606  053700  000606                BIS     SYSERR,R0       ;SET ANY SYSTEM ERRORS
 8034  010612  005037  000606                CLR     SYSERR          ;CLEAR SYS ERROR
 8035  010616  050037  000554          CSRCKX: BIS     R0,TSTERR       ;SET TEST ERROR FLAG, IF ERRORS
 8036  010622  000207                        RTS     PC              ;RETURN
 8037                                   ;-------------------------------------------------------------------------
```

```
8039                                    .SBTTL  RXES ERROR CHECK
8040                                    ;-----------------------------------------------------------------------
8041
8042  010624  017701  167644            ESRCHK: MOV     @RXDBAD,R1      ;GET RXES
8043  010630  043701  000550                    BIC     ESRMSK,R1       ;MASK OFF BITS DON'T CARE ABOUT
8044  010634  023701  000552                    CMP     ESRCMP,R1       ;IF RXES CONTAINS
8045  010640  001470                            BEQ     5$              ;ERRORS, THEN
8046  010642  010137  011540            1$:     MOV     R1,BADWRD       ;SET BAD WORD
8047  010646  013737  000552  011536            MOV     ESRCMP,CMPWRD   ;SET COMPARE WORD
8048  010654  012737  011574  011542            MOV     #EMSGTB,TABADR  ;SET MSG TABLE ADR
8049  010662  005037  011534                    CLR     BITOFF          ;SET BIT OFFSET
8050  010666  012737  000014  011532            MOV     #12.,BITLIM     ;SET # BITS TO CHECK
8051  010674  032777  020000  167602            BIT     #SW13,@SWR      ;IF INHIBIT ERROR REPORTS
8052  010702  001044                            BNE     4$              ;NOT SET, THEN
8053  010704  004737  011042                    CALL    TSHDCK          ;CALL TEST HEADER CHECK
8054  010710  012704  014736            2$:     MOV     #MSG27,R4       ;SET RXES MSG HEADER
8055  010714  004737  013232                    JSR     PC,TTOUT        ;PRINT MSG
8056  010720  032777  040000  167556            BIT     #SW14,@SWR      ;IF EXTENDED ERROR REPORTS
8057  010726  001430                            BEQ     3$              ;SET, THEN
8058  010730  012704  014411                    MOV     #MSG13,R4       ;SET REG FORMAT MSG
8059  010734  004737  013232                    JSR     PC,TTOUT        ;PRINT MSG
8060  010740  013703  000552                    MOV     ESRCMP,R3       ;SET GOOD RXES
8061  010744  004737  013334                    JSR     PC,OCTPE        ;PRINT GOOD RXES
8062  010750  012704  014564                    MOV     #MSG21,R4       ;SET SPACES MSG
8063  010754  004737  013232                    JSR     PC,TTOUT        ;PRINT SPACES
8064  010760  013703  011540                    MOV     BADWRD,R3       ;SET COMPARED RXES
8065  010764  004737  013334                    JSR     PC,OCTPE        ;PRINT COMPARED RXCS
8066  010770  012704  014564                    MOV     #MSG21,R4       ;SET SPACES MSG
8067  010774  004737  013232                    JSR     PC,TTOUT        ;PRINT MSG
8068  011000  017703  167470                    MOV     @RXDBAD,R3      ;SET FULL RXCS
8069  011004  004737  013334                    JSR     PC,OCTPE        ;PRINT FULL RXCS
8070  011010  004737  011314            3$:     JSR     PC,CKBITS       ;REPORT BAD BITS
8071  011014  012700  177777            4$:     MOV     #-1,R0          ;SET ERR
8072  011020  000405                            BR      ESRCKX          ;BR TO END
8073  011022  005000                    5$:     CLR     R0              ;CLEAR ERRORS
8074  011024  053700  000606                    BIS     SYSERR,R0       ;SET ANY SYSTEM ERRORS
8075  011030  005037  000606                    CLR     SYSERR          ;CLEAR SYSTEM ERRORS
8076  011034  050037  000554            ESRCKX: BIS     R0,TSTERR       ;SET TEST ERROR FLAG, IF ERRORS
8077  011040  000207                            RTS     PC              ;RETURN
8078                                    ;-----------------------------------------------------------------------
```

```
 8080                                   .SBTTL  TEST HEADER CHECK & PRINT
 8081                                   ;-------------------------------------------------------------------------
 8082
 8083  011042  005737  000602          TSHDCK: TST     HDRFLG          ;IF HEADER FLAG
 8084  011046  001030                          BNE     ENDTSH          ;NOT SET, THEN
 8085  011050  005237  000602                  INC     HDRFLG          ;SET TEST HEADER FLAG
 8086  011054  013704  000600                  MOV     TMSGAD,R4       ;SET TEST MSG #
 8087  011060  004737  013232                  JSR     PC,TTOUT        ;PRINT TEST MSG
 8088  011064  004737  010126                  CALL    SSUCOF          ;CALL SET <SUC> OFFSET
 8089  011070  004737  011132                  CALL    PRTSYS          ;CALL PRINT SYSTEM IDENTIFICATION
 8090  011074  032761  000020  000622          BIT     #DRV1,SOCMD(R1) ;IF DRIVE #1
 8091  011102  001404                          BEQ     1$              ;SELECTED, THEN
 8092  011104  012737  000001  011224          MOV     #1,DRIVE        ;SET DRIVE=#1
 8093  011112  000402                          BR      2$              ;BR TO 2$
 8094  011114  005037  011224          1$:     CLR     DRIVE           ;SET DRIVE=#0
 8095  011120  005037  011230          2$:     CLR     PTSIDF          ;CLEAR PRINT SIDE FLAG
 8096  011124  004737  011154                  CALL    PRTDRV          ;CALL PRINT DRIVE
 8097  011130  000207                  ENDTSH: RTS     PC              ;RETURN
 8098                                   ;-------------------------------------------------------------------------
```

```
                        MODULE U.PRTSYS - PRINT SYSTEM IDENTIFICATION                                   SEQ 0064

 8100
 8101
 8102  011132
  (1)  011132  012704  016425          .SBTTL  MODULE U.PRTSYS - PRINT SYSTEM IDENTIFICATION
  (1)  011136  004737  013232          ;-----------------------------------------------------------------
 8103  011142  013703  000610          PRTSYS:
 8104  011146  004737  013346                  MOV     #MSG51,R4       ;SET MSG-> ''SYSTEM:''
 8105  011152  000207                          CALL    TTOUT           ;PRINT MSG
 8106                                          MOV     SUC,R3          ;SET SYSTEM# FOR PRINT
 8107                                          CALL    OCTP            ;PRINT IT!
 8108                                          RETURN                  ;RETURN
 8109                                  ;-----------------------------------------------------------------
 8110
 8111  011154                          .SBTTL  MODULE U.PRTDRV - PRINT DRIVE IDENTIFICATION
  (1)  011154  012704  014547          ;-----------------------------------------------------------------
  (1)  011160  004737  013232          PRTDRV:
 8112  011164  013703  011224                  MOV     #MSG17,R4       ;SET MSG-> 'DRIVE: ''
 8113  011170  004737  013346                  CALL    TTOUT           ;PRINT MSG
 8114  011174  005737  011230                  MOV     DRIVE,R3        ;GET DRIVE #
 8115  011200  001410                          CALL    OCTP            ;PRINT IT!
 8116  011202  012704  014556                  TST     PTSIDF          ;IF SIDE FLAG
  (1)  011206  004737  013232                  BEQ     XUPTDV          ;SET, THEN
 8117  011212  013703  011226                  MOV     #MSG20,R4       ;SET MSG-> ''SIDE: ''
 8118  011216  004737  013346                  CALL    TTOUT           ;PRINT MSG
 8119  011222  000207                          MOV     SIDE,R3         ;GET SIDE #
 8120                                          CALL    OCTP            ;PRINT IT!
 8121  011224  000000                  XUPTDV: RETURN                  ;RETURN
 8122  011226  000000                  ;-----------------------------------------------------------------
 8123  011230  000000                  DRIVE:  0                       ;DRIVE # TO PRINT
 8124                                  SIDE:   0                       ;SIDE # TO PRINT
 8125                                  PTSIDF: 0                       ;SIDE PRINT CONTROL FLAG
 8126                                  ;-----------------------------------------------------------------
 8127
 8128                                  .SBTTL  MODULE U.PRTKSC - PRINT TRACK & SECTOR ERROR IDENT
 8129  011232                          ;-----------------------------------------------------------------
  (1)  011232  012704  014570          PRTKSC:
  (1)  011236  004737  013232                  MOV     #MSG22,R4       ;SET MSG-> ''CRC ERROR-''
 8130  011242  012704  014506                  CALL    TTOUT           ;PRINT MSG
  (1)  011246  004737  013232                  MOV     #MSG15,R4       ;SET MSG-> ''TRACK=''
 8131  011252  016103  011304                  CALL    TTOUT           ;PRINT MSG
 8132  011256  004737  013346                  MOV     S0TRK(R1),R3    ;GET TRACK #
 8133  011262  012704  014526                  CALL    OCTP            ;PRINT IT!
  (1)  011266  004737  013232                  MOV     #MSG16,R4       ;SET MSG-> ''SECTOR=''
 8134  011272  016103  011310                  CALL    TTOUT           ;PRINT MSG
 8135  011276  004737  013346                  MOV     S0SEC(R1),R3    ;GET SECTOR #
 8136  011302  000207                          CALL    OCTP            ;PRINT IT!
 8137                                          RETURN                  ;RETURN
 8138  011304  000000                  ;-----------------------------------------------------------------
 8139  011306  000000                  S0TRK:  0                       ;SYS#0 TRACK
 8140  011310  000000                  S1TRK:  0                       ;SYS#1 TRACK
 8141  011312  000000                  S0SEC:  0                       ;SYS#0 SECTOR
 8142                                  S1SEC:  0                       ;SYS#1 SECTOR
                                       ;-----------------------------------------------------------------
```

```
                                        .SBTTL  CHECK BITS SET & NOT SET
                                        ;---------------------------------------------------------------------

011314  005037  011524        CKBITS: CLR     BITPAS          ;CLEAR BIT PASS COUNT
011320  012737  012222  011530        MOV     #SETMSG,BITMSG  ;SET, SET BITS MSG ADR
011326  013700  011536                MOV     CMPWRD,R0       ;GET COMPARE WORD
011332  013701  011540                MOV     BADWRD,R1       ;GET BAD WORD
011336  040001                        BIC     R0,R1           ;R1 = BITS THAT SHOULDN'T BE SET
011340  005100                        COM     R0              ;COMPLIMENT COMPARE WORD
011342  053700  011540                BIS     BADWRD,R0       ;SET BITS
011346  005100                        COM     R0              ;R0 = BITS THAT SHOULD BE SET
011350  005737  011534                TST     BITOFF          ;IF BIT OFFSET
011354  001411                        BEQ     2$              ;NOT=0, THEN
011356  005337  011534        1$:     DEC     BITOFF          ;
011362  000241                        CLC                     ;CLEAR CARRY
011364  006000                        ROR     R0              ;
011366  000241                        CLC                     ;CLEAR CARRY
011370  006001                        ROR     R1              ;
011372  005737  011534                TST     BITOFF          ;IF BIT OFFSET
011376  001367                        BNE     1$              ;EQUALS 0, THEN
011400  005037  011526        2$:     CLR     BITCNT          ;CLEAR BIT COUNTER
011404  032701  000001        3$:     BIT     #1,R1           ;
011410  001420                        BEQ     4$              ;
011412  012704  015210                MOV     #MSG35,R4       ;SET UP "-" MSG
011416  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
011422  013704  011526                MOV     BITCNT,R4       ;GET BIT COUNT
011426  006304                        ASL     R4              ;DOUBLE FOR WORD ADDRESSING
011430  063704  011542                ADD     TABADR,R4       ;ADD TABLE ADDRESS
011434  011404                        MOV     (R4),R4         ;SET MSG TO PRINT
011436  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
011442  013704  011530                MOV     BITMSG,R4       ;SET SET BITS MSG
011446  004737  013232                JSR     PC,TTOUT        ;PRINT MSG
011452  005237  011526        4$:     INC     BITCNT          ;INCREMENT BIT# COUNTER
011456  000241                        CLC                     ;CLEAR CARRY BIT
011460  006001                        ROR     R1              ;SHIFT NEXT BIT FOR TEST
011462  023737  011532  011526        CMP     BITLIM,BITCNT   ;IF ALL BITS SPECIFIED
011470  101345                        BHI     3$              ;DONE, THEN
011472  005037  011526                CLR     BITCNT          ;RESET BIT COUNT
011476  005737  011524                TST     BITPAS          ;
011502  001007                        BNE     5$              ;
011504  005237  011524                INC     BITPAS          ;SET BITPASS TO GET OUT NEXT PASS
011510  012737  012200  011530        MOV     #NSMSG,BITMSG   ;SET NOT SET BITS MSG ADR
011516  010001                        MOV     R0,R1           ;GET NOT SET BITS FOR TEST
011520  000731                        BR      3$              ;DO NOT SET BITS
011522  000207                5$:     RTS     PC              ;RETURN
                                        ;---------------------------------------------------------------------
011524  000000                BITPAS: 0
011526  000000                BITCNT: 0
011530  000000                BITMSG: 0
011532  000000                BITLIM: 0
011534  000000                BITOFF: 0
011536  000000                CMPWRD: 0
011540  000000                BADWRD: 0
011542  000000                TABADR: 0
                                        ;---------------------------------------------------------------------
```

```
 8200
 8201  011544  011624                  ;-------------------------------------------------------------------
 8202  011546  011634                  CMSGTB: .WORD   CMSG4
 8203  011550  011641                          .WORD   CMSG5
 8204  011552  011662                          .WORD   CMSG6
 8205  011554  011665                          .WORD   CMSG7
 8206  011556  011675                          .WORD   CMSG8
 8207  011560  011704                          .WORD   CMSG9
 8208  011562  011713                          .WORD   CMSG10
 8209  011564  011750                          .WORD   CMSG11
 8210  011566  011750                          .WORD   CMSG12
 8211  011570  014202                          .WORD   CMSG12
 8212  011572  011770                          .WORD   MSG7
 8213                                          .WORD   CMSG15
 8214  011574  011777                  ;-------------------------------------------------------------------
 8215  011576  012003                  EMSGTB: .WORD   EMSG0
 8216  011600  012014                          .WORD   EMSG1
 8217  011602  012026                          .WORD   EMSG2
 8218  011604  012035                          .WORD   EMSG3
 8219  011606  012051                          .WORD   EMSG4
 8220  011610  012066                          .WORD   EMSG5
 8221  011612  012103                          .WORD   EMSG6
 8222  011614  012115                          .WORD   EMSG7
 8223  011616  012125                          .WORD   EMSG8
 8224  011620  012134                          .WORD   EMSG9
 8225  011622  012156                          .WORD   EMSG10
 8226                                          .WORD   EMSG11
 8227  011624  051104  053111  020105    ;-------------------------------------------------------------------
 8228  011634  047504  042516     043    CMSG4:  .ASCII  /DRIVE 1#/
 8229  011641     111  052116  051105    CMSG5:  .ASCII  /DONE#/
 8230  011662  051124     043            CMSG6:  .ASCII  /INTERRUPT ENABLE#/
 8231  011665     104  047105  044523    CMSG7:  .ASCII  /TR#/
 8232  011675     123  042111  020105    CMSG8:  .ASCII  /DENSITY#/
 8233  011704  044502  020124  030061    CMSG9:  .ASCII  /SIDE 1#/
 8234  011713     104  052517  046102    CMSG10: .ASCII  /BIT 10#/
 8235  011750  054105  042524  042116    CMSG11: .ASCII  /DOUBLE DENSITY FLOPPY SYSTEM#/
 8236  011770  051105  047522  020122    CMSG12: .ASCII  /EXTENDED MEMORY#/
 8237                                    CMSG15: .ASCII  /ERROR #/
 8238  011777     103  041522     043    ;-------------------------------------------------------------------
 8239  012003     123  042111  020105    EMSG0:  .ASCII  /CRC#/
 8240  012014  047111  052111  042040    EMSG1:  .ASCII  /SIDE RDY#/
 8241  012026  041501  046040  053517    EMSG2:  .ASCII  /INIT DONE#/
 8242  012035     104  047105  044523    EMSG3:  .ASCII  /AC LOW#/
 8243  012051     104  051511  020113    EMSG4:  .ASCII  /DENSITY ERR#/
 8244  012066  042504  042514  042524    EMSG5:  .ASCII  /DISK DENSITY#/
 8245  012103     104  044522  042526    EMSG6:  .ASCII  /DELETED DATA#/
 8246  012115     104  044522  042526    EMSG7:  .ASCII  /DRIVE RDY#/
 8247  012125     123  042111  020105    EMSG8:  .ASCII  /DRIVE 1#/
 8248  012134  047527  042122  041440    EMSG9:  .ASCII  /SIDE 1#/
 8249  012156  047516  020116  054105    EMSG10: .ASCII  /WORD CNT OVERFLOW#/
 8250                                    EMSG11: .ASCII  /NON EXISTENT MEM.#/
 8251  012200  041040  052111  047055    ;-------------------------------------------------------------------
 8252  012222  041040  052111  051455    NSMSG:  .ASCII  / BIT-NOT SET ERR #/
 8253                                    SETMSG: .ASCII  / BIT-SET ERR #/
 8254                                            .EVEN
 8255                                    ;-------------------------------------------------------------------
```

```
 8257                                   .SBTTL  ERROR SET SUBROUTINE
 8258                                   ;-----------------------------------------------------------------------
 8259
 8260  012240  017637  000000  000402  ERRSET: MOV     @(SP),$FATAL    ;MOVE ERR# TO $FATAL MAILBOX
 8261  012246  012737  000001  000400           MOV     #1,$MSGTY       ;SET MAIL BOX FLAG
 8262  012254  062716  000002                   ADD     #2,(SP)         ;ADJUST STACK RETURN ADDRESS
 8263  012260  000207                   ERSETX: RTS     PC              ;RETURN
 8264                                   ;-----------------------------------------------------------------------
 8265
 8266                                   .SBTTL  BUS INITIALIZE SUBROUTINE
 8267                                   ;-----------------------------------------------------------------------
 8268
 8269  012262  032777  041000  166214  BSINIT: BIT     #41000,@SWR     ;IF LOOP
 8270  012270  001407                           BEQ     1$              ;IS SET, THEN
 8271  012272  004737  011042                   CALL    TSHDCK          ;PRINT TEST HEADER
 8272  012276  032777  010000  166200           BIT     #SW12,@SWR      ;IF BUS INITIALIZE
 8273  012304  001401                           BEQ     1$              ;BIT SET, THEN
 8274  012306  000005                           RESET                   ;INITIALIZE BUS
 8275  012310  005737  012402           1$:     TST     LPPRT           ;IF LOOP MESSAGE
 8276  012314  001026                           BNE     5$              ;NOT PRINTED, THEN
 8277  012316  032777  001000  166160  2$:     BIT     #SW09,@SWR      ;IF LOOP ON
 8278  012324  001405                           BEQ     3$              ;ERROR, THEN
 8279  012326  012704  014752                   MOV     #MSG28,R4       ;SET LOOP ON ERROR MSG
 8280  012332  004737  013232                   CALL    TTOUT           ;PRINT MSG
 8281  012336  000407                           BR      4$              ;
 8282  012340  005777  166140           3$:     TST     @SWR            ;IF HALT ON ERROR
 8283  012344  100012                           BPL     5$              ;IS SET, THEN
 8284  012346  012704  015223                   MOV     #MSG37,R4       ;SET HALT ON ERROR MSG
 8285  012352  004737  013232                   CALL    TTOUT           ;PRINT MSG
 8286  012356  011603                   4$:     MOV     (SP),R3         ;GET RETURN ADR
 8287  012360  062703  000014                   ADD     #14,R3          ;ADJ. ADDRESS FOR ERR NO.
 8288  012364  011303                           MOV     (R3),R3         ;GET ERR NO.
 8289  012366  004737  013346                   CALL    OCTP            ;PRINT ERR #
 8290  012372  012737  000001  012402  5$:     MOV     #1,LPPRT        ;SET LOOP MSG PRINTED FLAG
 8291  012400  000207                   BSINTX: RTS     PC              ;RETURN
 8292                                   ;-----------------------------------------------------------------------
 8293  012402  000000                   LPPRT:  0                       ;LOOP FLAG
 8294                                   ;-----------------------------------------------------------------------
```

```
 8296                                   .SBTTL  RX02 INTERRUPT HANDLER #0
 8297                                   ;---------------------------------------------------------------
 8298
 8299  012404  005237  000616           RXINT0: INC     S0RDY           ;INCREMENT SYS #0 READY
 8300  012410  000002                         RTI                     ;RETURN TO CALLER
 8301                                   ;---------------------------------------------------------------
 8302
 8303                                   .SBTTL  RX02 INTERRUPT HANDLER #1
 8304                                   ;---------------------------------------------------------------
 8305
 8306  012412  005237  000620           RXINT1: INC     S1RDY           ;INCREMENT SYS #1 READY
 8307  012416  000002                         RTI                     ;RETURN TO CALLER
 8308                                   ;---------------------------------------------------------------
 8309                                   .SBTTL  TTY INTERRUPT HANDLER
 8310                                   ;---------------------------------------------------------------
 8311
 8312  012420  004737  013156           TTINT:  CALL    TTIN            ;CALL TTY INPUT ROUTINE
 8313  012424  113737  000526  000632           MOVB    TIB,TTITFG      ;SAVE TTY INPUT CHARACTER
 8314  012432  022737  000003  000632           CMP     #3,TTITFG       ;IF TTY INTERRUPT
 8315  012440  001007                         BNE     1$              ;WAS CTRL C, THEN
 8316  012442  012704  014715                 MOV     #MSG25,R4       ;SET MSG-> "^C"
  (1)  012446  004737  013232                 CALL    TTOUT           ;PRINT MSG
 8317  012452  005000                         CLR     R0              ;
 8318  012454  000137  001000                 JMP     START           ;JUMP TO START OVER
 8319  012460  052777  000100  165612  1$:    BIS     #BIT6,@TKS      ;SET TTY INTERRURT BIT
 8320  012466  005277  165606                 INC     @TKS            ;SET TTY GO BIT
 8321  012472  000002                         RTI
 8322                                   ;---------------------------------------------------------------
 8323
```

```
                                        .SBTTL  BUS ADDRESS TRAP HANDLER
                                        ;-------------------------------------------------------------------
012474  005237  000570          TRAP:   INC     BTRPFL          ;SET BUS TRAP FLAG
012500  032777  020000  165776          BIT     #20000,@SWR     ;SEE IF SHOULD PRINT ERRORS
012506  001020                          BNE     TRAP2           ;IF NOT, BR
012510  005737  000602                  TST     HDRFLG          ;SEE IF DONE HEADER
012514  001006                          BNE     TRAP1           ;IF SO, BR
012516  005237  000602                  INC     HDRFLG          ;ELSE SET HEADER FLAG
012522  013704  000600                  MOV     TMSGAD,R4
012526  004737  013232                  JSR     PC,TTOUT        ;PRINT HEADER
012532  012704  014701          TRAP1:  MOV     #MSG24,R4
012536  004737  013232                  JSR     PC,TTOUT        ;PRINT ERROR
012542  010103                          MOV     R1,R3
012544  004737  013346                  JSR     PC,OCTP         ;PRINT ADDRESS OF TRAP
012550  004737  012570          TRAP2:  JSR     PC,APTER
012554  000055                          .WORD   '55             ;BUS TRAP ERROR
012556  005777  165722                  TST     @SWR            ;SEE IF HALT ON ERROR
012562  100001                          BPL     TRAPX           ;IF NOT, BR
012564  000000                          HALT
012566  000002          TRAPX:  RTI                             ;RETURN FROM INTERRUPT
                                        ;-------------------------------------------------------------------

                                        .SBTTL  APT ERROR HANDLER
                                        ;-------------------------------------------------------------------
012570  132737  000001  000420  APTER:  BITB    #APTENV,$ENV    ;IF IN
012576  001407                          BEQ     2$              ;APT MODE, THEN
012600  017637  000000  000402          MOV     @(SP),$FATAL    ;MOVE ERR # TO $FATAL MAILBOX
012606  012737  000001  000400          MOV     #1,$MSGTY       ;SET MAIL BOX FLAG
012614  000777                  1$:     BR      1$              ;APT ERROR LOOP
012616  062716  000002          2$:     ADD     #2,(SP)         ;
012622  000207                          RTS     PC              ;RETURN
                                        ;-------------------------------------------------------------------

                                        .SBTTL  MODULE U.OPRANS - SETUP & GET  OPERATOR ANSWERS
                                        ;-------------------------------------------------------------------
012624  012737  000116  012646  GETANS: MOV     #'N,ANSDEF     ;SET DEFAULT ANSWER-'NO'
012632  012737  000131  012650          MOV     #'Y,ANSCHG      ;SET CHANGE ANSWER-'YES'
012640  004737  013036                  CALL    TTAR            ;GET ANSWER-CALL TTY ANSWER
012644  000207                          RETURN                  ;RETURN
                                        ;-------------------------------------------------------------------
012646  000000          ANSDEF: 0                               ;DEFAULT ANSWER BUFFER
012650  000000          ANSCHG: 0                               ;CHANGE ANSWER BUFFER
                                        ;-------------------------------------------------------------------
```

```
8372
8373                                    .SBTTL  TTY ENTRY SUBROUTINE
8374                                    ;-----------------------------------------------------------------------
8375
8376                                    ;THIS SUBROUTINE IS USED BY THE TEST CONDITION ENTRY ROUTINE TO READ
8377                                    ;THE RESPONSE ENTERED AT THE TTY AND CHECK THEM FOR LEGALITY AND LIMITS.
8378                                    ;ALL RESPONSE MUST BE TYPED IN OCTAL (0 - 7) AND MUST FALL WITHIN THE
8379                                    ;LIMITS SET BY THE CALLING ROUTINE.  IF AN ENTRY IS ILLEGAL OR OUTSIDE
8380                                    ;THE LIMITS, A QUESTION MARK (?) IS TYPED AND THE RESPONSE MAY BE RE-
8381                                    ;ENTERED.  ENTRIES MAY NOT EXCEED SIX (6) CHARACTERS AND MAY BE TERMIN-
8382                                    ;ATED AT LESS THAN SIX BY TYPING A CARRIAGE RETURN.
8383
8384  012652  005037  000562          TTR:    CLR     TEMP1           ;CLEAR FIRST CHARACTER FLAG
8385  012656  005000                          CLR     R0
8386  012660  004737  013156          TTR0:   JSR     PC,TTIN         ;GO READ CHARACTER
8387  012664  122737  000015  000526          CMPB    #15,TIB         ;SEE IF CR
8388  012672  001005                          BNE     TTR1            ;IF NOT, BR
8389  012674  005737  000562                  TST     TEMP1           ;SEE IF FIRST CHARACTER
8390  012700  001446                          BEQ     TTR5            ;IF SO, BR
8391  012702  000137  012774                  JMP     TTR2            ;ELSE, GO LOAD VALUE
8392  012706  122737  000060  000526  TTR1:   CMPB    #60,TIB         ;SEE IF CHAR IS LESS THAN 0
8393  012714  101402                          BLOS    TTR1A           ;IF NOT, BR
8394  012716  000137  013020                  JMP     TINER           ;ELSE, GO TO ERROR
8395  012722  122737  000070  000526  TTR1A:  CMPB    #70,TIB         ;SEE IF CHAR IS GREATER THAN 7
8396  012730  101002                          BHI     TTR1B           ;IF NOT, BR
8397  012732  000137  013020                  JMP     TINER           ;ELSE, GO TO ERROR
8398  012736  005237  000562          TTR1B:  INC     TEMP1           ;SET FIRST CHARACTER FLAG
8399  012742  000241                          CLC
8400  012744  006100                          ROL     R0
8401  012746  000241                          CLC
8402  012750  006100                          ROL     R0              ;SHIFT 3 LEFT
8403  012752  000241                          CLC
8404  012754  006100                          ROL     R0
8405  012756  042737  177770  000526          BIC     #177770,TIB     ;STRIP ASCII
8406  012764  053700  000526                  BIS     TIB,R0          ;LOAD CHARACTER
8407  012770  005301                          DEC     R1              ;SEE IF DONE
8408  012772  001332                          BNE     TTR0            ;IF NOT, BR
8409  012774  020002          TTR2:   CMP     R0,R2           ;SEE IF EXCEEDED MAXIMUM LIMIT
8410  012776  101402                          BLOS    TTR3            ;IF NOT, BR
8411  013000  000137  013020                  JMP     TINER           ;ELSE, GO TO ERROR
8412  013004  020300          TTR3:   CMP     R3,R0           ;SEE IF BELOW MINIMUM LIMIT
8413  013006  101402                          BLOS    TTR4            ;IF NOT, BR
8414  013010  000137  013020                  JMP     TINER           ;ELSE, GO TO ERROR
8415  013014  010015          TTR4:   MOV     R0,(R5)         ;LOAD VALUE
8416  013016  000207          TTR5:   RTS     PC              ;EXIT
8417                                    ;-----------------------------------------------------------------------
8418
8419
8420                                    .SBTTL  TTY ENTRY ERROR SUBROUTINE
8421                                    ;-----------------------------------------------------------------------
8422
8423  013020  012704  014202  TINER:   MOV     #MSG7,R4
8424  013024  004737  013232          JSR     PC,TTOUT        ;PRINT?
8425  013030  162716  000020          SUB     #20,(SP)        ;RESET SP TO START OF VALUE ROUTINE
8426  013034  000207                  RTS     PC              ;REDO VALUE ENTRY
8427                                    ;-----------------------------------------------------------------------
```

```

                                        .SBTTL  TTY ANSWER ENTRY SUBROUTINE
                                        ;-----------------------------------------------------------------------
                                        ;THIS SUBROUTINE IS USED BY THE TEST CONDITION ENTRY ROUTINE TO
                                        ;READ THE RESPONSE ENTERED AT THE TTY AND CHECK THEM FOR LEGALITY.
                                        ;ALL RESPONSES MUST BE EQUAL TO  RETURN THE DEFAULT ANSWER, CHANGED ANSWER
                                        ;OR A CARRIAGE RETURN. IF ENTRY IS ILLEGAL A QUESTION MARK IS TYPED AND
                                        ;THE RESPONSE MAY BE REENTERED.
                                        ;
013036  105037  000530                  TTAR:   CLRB    ANSWER          ;CLEAR ANSWER
013042  005000                                  CLR     R0              ;
013044  004737  013156                          CALL    TTIN            ;GO READ  A CHARACTER
013050  122737  000032  000526                  CMPB    #32,TIB         ;IF ''^L''
013056  001003                                  BNE     TTAR1           ;THEN
013060  005237  000576                          INC     STFLG           ;SET START PROGRAM
013064  000433                                  BR      XTTAR           ;BR TO END
013066  122737  000015  000526          TTAR1:  CMPB    #15,TIB         ;IF ''CR''
013074  001004                                  BNE     TTAR2           ;THEN
013076  113737  012646  000530                  MOVB    ANSDEF,ANSWER   ;SET DEFAULT ANSWER
013104  000423                                  BR      XTTAR           ;BR TO END
013106  042737  000040  000526          TTAR2:  BIC     #40,TIB         ;CLEAR LOWER CASE BIT
013114  123737  012650  000526                  CMPB    ANSCHG,TIB      ;IF ANSWER NOT OK WITH CHANGE ANS
013122  001411                                  BEQ     TTAR3           ;THEN
013124  123737  012646  000526                  CMPB    ANSDEF,TIB      ;IF ASNWER NOT OK WITH DEFAULT ANS
013132  001405                                  BEQ     TTAR3           ;THEN
013134  012704  014202                          MOV     #MSG7,R4        ;SETUP MSG ''?''
013140  004737  013232                          CALL    TTOUT           ;PRINT MSG
013144  000734                                  BR      TTAR            ;START OVER
013146  113737  000526  000530          TTAR3:  MOVB    TIB,ANSWER      ;SAVE ANSWER
013154  000207                          XTTAR:  RETURN                  ;RETURN
                                        ;-----------------------------------------------------------------------


                                        .SBTTL  TTY READ SUBROUTINE
                                        ;-----------------------------------------------------------------------

013156  005077  165116                  TTIN:   CLR     @TKS
013162  005037  000526                          CLR     TIB
013166  005277  165106                          INC     @TKS
013172  105777  165102                  TTIN1:  TSTB    @TKS
013176  100375                                  BPL     TTIN1
013200  017737  165076  000526                  MOV     @TKB,TIB
013206  105777  165072                  TTIN2:  TSTB    @TPS
013212  100375                                  BPL     TTIN2
013214  113777  000526  165064                  MOVB    TIB,@TPB
013222  042737  000200  000526                  BIC     #200,TIB                ;STRIP OFF TOP BIT OF BYTE
013230  000207                                  RTS     PC
                                        ;-----------------------------------------------------------------------
```

```
 8476                                   .SBTTL  TTY ASCII OUTPUT SUBROUTINE
 8477                                   ;---------------------------------------------------------------
 8478
 8479  013232  112437  000524           TTOUT:  MOVB    (R4)+,TOB
 8480  013236  122737  000043  000524           CMPB    #43,TOB         ;IF TOB= ''#''
 8481  013244  001432                           BEQ     TCEX
 8482  013246  122737  000045  000524           CMPB    #45,TOB         ;IF TOB= ''%''
 8483  013254  001403                           BEQ     TCRLF
 8484  013256  004737  013700                   JSR     PC,TOG
 8485  013262  000763                           BR      TTOUT
 8486  013264  112737  000015  000524  TCRLF:  MOVB    #15,TOB         ;SET TOB= ''CR''
 8487  013272  004737  013700                   JSR     PC,TOG
 8488  013276  012703  000001                   MOV     #1,R3
 8489  013302  005037  000524           TCRLFA: CLR     TOB
 8490  013306  004737  013700                   JSR     PC,TOG
 8491  013312  005303                           DEC     R3
 8492  013314  001372                           BNE     TCRLFA          ;DO FILLERS
 8493  013316  112737  000012  000524           MOVB    #12,TOB         ;SET TOB= ''LF''
 8494  013324  004737  013700                   JSR     PC,TOG
 8495  013330  000740                           BR      TTOUT
 8496  013332  000207                   TCEX:   RTS     PC              ;RETURN
 8497                                   ;---------------------------------------------------------------
```

```
 8499                                   .SBTTL  OCTAL OUTPUT SUBROUTINE
 8500                                   ;-------------------------------------------------------------------------
 8501
 8502  013334  012737  000001  013570  OCTPE:  MOV     #1,OFL
 8503  013342  010304                          MOV     R3,R4
 8504  013344  000410                          BR      OCTP0
 8505  013346  005037  013570          OCTP:   CLR     OFL             ;CLEAR FLAG FOR LEADING ZERO
 8506  013352  010304                  OCTPE1: MOV     R3,R4           ;SEE IF NUMBER IS ZERO
 8507  013354  001004                          BNE     OCTP0           ;IF NOT ZERO, BR
 8508  013356  004737  013550                  JSR     PC,OCTPG1       ;ELSE PRINT ZERO
 8509  013362  000137  013512                  JMP     OCTP3           ;SPACE AND EXIT
 8510  013366  032704  100000          OCTP0:  BIT     #100000,R4      ;SEE IF MSG=1
 8511  013372  001406                          BEQ     OCTP1           ;IF NOT, BR
 8512  013374  012704  000001                  MOV     #1,R4
 8513  013400  004737  013526                  JSR     PC,OCTPG        ;PRINT 1
 8514  013404  000137  013416                  JMP     OCTP2
 8515  013410  005004                  OCTP1:  CLR     R4
 8516  013412  004737  013526                  JSR     PC,OCTPG        ;PRINT 0
 8517  013416  010304                  OCTP2:  MOV     R3,R4
 8518  013420  006004                          ROR     R4
 8519  013422  006004                          ROR     R4
 8520  013424  006004                          ROR     R4
 8521  013426  006004                          ROR     R4              ;POSITION DIGIT
 8522  013430  000304                          SWAB    R4
 8523  013432  004737  013526                  JSR     PC,OCTPG        ;PRINT DIGIT 2
 8524  013436  010304                          MOV     R3,R4
 8525  013440  006004                          ROR     R4
 8526  013442  000304                          SWAB    R4
 8527  013444  004737  013526                  JSR     PC,OCTPG        ;PRINT DIGIT 3
 8528  013450  010304                          MOV     R3,R4
 8529  013452  006104                          ROL     R4
 8530  013454  006104                          ROL     R4
 8531  013456  000304                          SWAB    R4
 8532  013460  004737  013526                  JSR     PC,OCTPG        ;PRINT DIGIT 4
 8533  013464  010304                          MOV     R3,R4
 8534  013466  006004                          ROR     R4
 8535  013470  006004                          ROR     R4
 8536  013472  006004                          ROR     R4
 8537  013474  004737  013526                  JSR     PC,OCTPG        ;PRINT DIGIT 5
 8538  013500  005237  013570                  INC     OFL             ;SET FLAG TO PRINT LSD
 8539  013504  010304                          MOV     R3,R4
 8540  013506  004737  013526                  JSR     PC,OCTPG        ;PRINT DIGIT 6
 8541  013512  012737  000240  000524  OCTP3:  MOV     #240,TOB        ;PRINT SPACE
 8542  013520  004737  013700                  JSR     PC,TOG          ;EXIT
 8543  013524  000207                  OCTPX:  RTS     PC              ;EXIT
 8544                                   ;-------------------------------------------------------------------------
```

```
 8546
 8547  013526  042704  177770          OCTPG:  BIC     #177770,R4
 8548  013532  001004                          BNE     OCTPG0
 8549  013534  005737  013570                  TST     OFL
 8550  013540  001001                          BNE     OCTPG0
 8551  013542  000207                          RTS     PC
 8552  013544  005237  013570          OCTPG0: INC     OFL
 8553  013550  052704  000260          OCTPG1: BIS     #260,R4
 8554  013554  010437  000524                  MOV     R4,TOB
 8555  013560  004737  013700                  JSR     PC,TOG
 8556  013564  010304                          MOV     R3,R4
 8557  013566  000207                  OCTPGX: RTS     PC
 8558
 8559  013570  000000                  OFL:    0                       ;FIRST CHAR FLAG
 8560
```

```
8562                                    .SBTTL  DATA CHARACTER OUTPUT SUBROUTINE
8563                                    ;---------------------------------------------------------------
8564
8565
8566  013572  005037  000524            DOUT:   CLR     TOB
8567  013576  012704  000010                    MOV     #10,R4          ;SET NUMBER TO PRINT
8568  013602  110337  000524                    MOVB    R3,TOB
8569  013606  105777  164472            DOUT1:  TSTB    @TPS
8570  013612  100375                            BPL     DOUT1
8571  013614  132737  000200  000524          BITB    #200,TOB
8572  013622  001404                            BEQ     DOUT2
8573  013624  012777  000061  164454          MOV     #061,@TPB
8574  013632  000403                            BR      DOUT3
8575  013634  012777  000060  164444  DOUT2:  MOV     #060,@TPB
8576  013642  006137  000524            DOUT3:  ROL     TOB
8577  013646  005304                            DEC     R4
8578  013650  001356                            BNE     DOUT1
8579  013652  000207                            RTS     PC
8580                                    ;---------------------------------------------------------------
8581  013654  013703  000566            DOUTD:  MOV     TEMP3,R3
8582  013660  000303                            SWAB    R3
8583  013662  004737  013572                    JSR     PC,DOUT
8584  013666  013703  000566                    MOV     TEMP3,R3
8585  013672  004737  013572                    JSR     PC,DOUT
8586  013676  000207                            RTS     PC
8587                                    ;---------------------------------------------------------------
8588
8589                                    .SBTTL  TTY OUTPUT
8590                                    ;---------------------------------------------------------------
8591  013700  132737  000040  000421  TOG:    BITB    #APTCSP,$ENVM   ;SEE IF CONSOLE SUPPRESS MODE
8592  013706  001014                            BNE     TEX             ;IF SO, BR
8593  013710  105777  164370                    TSTB    @TPS
8594  013714  100371                            BPL     TOG
8595  013716  113777  000524  164362          MOVB    TOB,@TPB
8596  013724  005737  000532                    TST     TTWAIT          ;IS TTY WAIT FLAG SET
8597  013730  001403                            BEQ     TEX             ;THEN
8598  013732  105777  164346            1$:     TSTB    @TPS            ;WAIT FOR CHARTER
8599  013736  100375                            BPL     1$              ;TO BE PRINTED
8600  013740  000207                    TEX:    RTS     PC
8601                                    ;---------------------------------------------------------------
```

```
                                         .SBTTL  MESSAGE TABLE
                                         ;------------------------------------------------------------------------

013742  020045  020040  054122  MSG1:    .ASCII  /%   RXCS       RXDB        CMD%    #/
014002  051445  043117  053524  MSG2:    .ASCII  /%SOFTWARE SWITCH REG (OCT) = #/
014040  022445  055103  054122  MSG3:    .ASCII  /%%CZRXEAO RX02 FMTR PROG%#/
014072  020045  020040  020040  MSG4:    .ASCII  /%       RXCS ADDRESS (OCT) = #/
014127     045  020040  020040  MSG5:    .ASCII  /%             VECTOR (OCT) = #/
014164  042445  042116  047440  MSG6:    .ASCII  /%END OF PASS #/
014202  037440  021440          MSG7:    .ASCII  / ? #/
014206  047045  020117  054523  MSG8:    .ASCII  /%NO SYSTEM AVAILABLE TO FORMAT #/
014246  022445  047506  046522  MSG9:    .ASCII  /%%FORMAT COMPLETED #/
014272  020045  044440  052116  MSG10:   .ASCII  /%  INTERRUPT DID NOT OCCUR #/
014326  052445  044516  020124  MSG11:   .ASCII  /%UNIT NOT RESPONDING TO ADDRESS #/
014367     045  042101  051104  MSG12:   .ASCII  /%ADDRESSING TEST #/
014411     045  020040  054105  MSG13:   .ASCII  /%  EXPECT    RCVD       ACTUAL%  #/
014452  022445  047506  046522  MSG14:   .ASCII  /%%FORMAT DONE ON FOLLOWING #/
014506  020040  051124  041501  MSG15:   .ASCII  /  TRACK (OCT) =#/
014526  020040  042523  052103  MSG16:   .ASCII  /  SECTOR (OCT) =#/
014547     104  044522  042526  MSG17:   .ASCII  /DRIVE:#/
014556  044523  042504  021472  MSG20:   .ASCII  /SIDE:#/
014564  020040  021440          MSG21:   .ASCII  /   #/
014570  020045  051103  020103  MSG22:   .ASCII  /% CRC ERROR -> #/
014610  020045  044124  051511  MSG23:   .ASCII  /% THIS SYSTEM NOT CAPABLE OF DOUBLE DENSITY OPERATIONS! #/
014701     045  052502  020123  MSG24:   .ASCII  /%BUS TRAP: #/
014715     045  041536  021440  MSG25:   .ASCII  /%^C #/
014722  020045  054122  051503  MSG26:   .ASCII  /% RXCS ERR #/
014736  020045  054122  051505  MSG27:   .ASCII  /% RXES ERR #/
014752  020040  046055  047517  MSG28:   .ASCII  /  -LOOPING ON ERR: #/
014776  042445  051122  051117  MSG29:   .ASCII  /%ERROR BIT SET AFTER INIT #/
015031     045  042040  047117  MSG30:   .ASCII  /% DONE BIT NOT SET #/
015055     045  052040  020122  MSG31:   .ASCII  /% TR BIT NOT SET #/
015077     045  020040  020040  MSG32:   .ASCII  /%                = #/
015121     040  026440  020076  MSG33:   .ASCII  /  -> DROPPED  #/
015140  026445  051076  046505  MSG34:   .ASCII  /%->REMOVE XXDP MEDIA FROM THIS SYSTEM!!#/
015210  020045  020040  020055  MSG35:   .ASCII  /%    - #/
015217     045  037040     043  MSG36:   .ASCII  /% >#/
015223     040  026440  040510  MSG37:   .ASCII  /  -HALT ON ERR: #/
015244  042445  052116  051105  MSG38:   .ASCII  /%ENTER CONDITIONS IN OCTAL #/
015300  052045  044510  020123  MSG39:   .ASCII  /%THIS PROGRAM FORMATS DISKETTES TO SINGLE OR DOUBLE(DEFAULT) DENSITY/
015404  047445  020116  047502  MSG40:   .ASCII  /%ON BOTH DRIVES OF A FLOPPY DISK SUBSYSTEM CAPABLE OF DOUBLE DENSITY/
015510  047445  042520  040522           .ASCII  /%OPERATIONS. A CRC VERIFY IS PERFORMED ON TRACKS: 0 & 76 (DEFAULT) OR/
015615     045  046101  020114           .ASCII  /%ALL TRACKS OF THE FORMATTED DISKETTE, IF SELECTED.%#/
015702  052055  046511  020105  MSG41:   .ASCII  /-TIME OUT ERROR #/
015723     045  021440          MSG42:   .ASCII  /% #/
015726  044045  046105  037520  MSG43:   .ASCII  /%HELP? (Y OR N) N  #/
015752  004445  047111  052111  MSG44:   .ASCII  /%      INITIALIZE NOT DONE - RUN DIAGNOSTICS UNLESS OBVIOUS ERR#/
016045     045  042523  020124  MSG45:   .ASCII  /%SET DISKETTE TO SINGLE DENSITY?    (Y OR N) N  #/
016125     045  042526  044522  MSG46:   .ASCII  /%VERIFY DISKETTE CRC (ALL TRACKS)?  (Y OR N) N  #/
016206  043045  047514  050120  MSG47:   .ASCII  /%FLOPPY DISK SYSTEM: 0 ADDRESS CHANGE? (Y OR N) N #/
016271     045  051511  040440  MSG48:   .ASCII  /%IS ANOTHER FLOPPY DISK SYSTEM AVAILABLE? (Y OR N) N #/
016357     045  047516  051440  MSG49:   .ASCII  /%NO SYSTEM AVAILABLE TO FORMATTER #/
016422  004445     011          MSG50:   .ASCII  /%            /
016425     045  054523  052123  MSG51:   .ASCII  /%SYSTEM:#/
016436  047045  020117  047504  MSG52:   .ASCII  /%NO DONE BIT AFTER INITIALIZE #/
016475     040  037055  051104  MSG53:   .ASCII  / ->DROPPED FROM FORMATTER%#/
```

```
 016530 004445  041501  046040  MSG54:  .ASCII  /%       AC LOW ERROR - IS FLOPPY DISK SYSTEM POWERED UP?#/
 016613    045  042011  044522  MSG55:  .ASCII  /%       DRIVE NOT READY - IS DISKETTE IN DRIVE & DOOR CLOSED?#/
 016703    045  047504  054440  MSG56:  .ASCII  /%DO YOU WANT TO FORMAT MORE DISKETTES? (Y OR N) N #/
 016766 051045  046505  053117  MSG57:  .ASCII  /%REMOVE FORMATTED DISKETTES & INSERT DISKETTES TO BE FORMATTED#/
 017065    045  026455  020076  MSG58:  .ASCII  /%--> TYPE ''CR'' WHEN READY#/
 017117    045  047506  046522  MSG59:  .ASCII  /%FORMATTER DONE-RESTART MONITOR OR UPDATE PROGRAM-->TYPE CTRL C TO/
 017221    045  052123  051101          .ASCII  /%START THIS PROGRAM AGAIN #/
                                ;------------------------------------------------------------------------


                                .SBTTL  TEST HEADERS
                                ;------------------------------------------------------------------------

 017254 022445  037055  042522  MSGCD3: .ASCII  /%%->READ SECTOR COMMAND ERR #/
 017311    045  026445  051476  MSGCD4: .ASCII  /%%->SET DENSITY COMMAND ERR #/
 017346 022445  037055  042522  MSGCD5: .ASCII  /%%->READ MAINTENANCE STATUS COMMAND ERR #/
 017417    045  026445  037055  MSGLP:  .ASCII  /%%-->LOOP MODULE-ERR #/
        017446                          .EVEN
                                ;------------------------------------------------------------------------


                                .SBTTL  ERROR CODE STORAGE
                                ;------------------------------------------------------------------------
 017446    000                  XER:    .BYTE   0                       ;ERR CODE - EXTENDED
 017447    000                          .BYTE   0                       ;
 017450    000                          .BYTE   0                       ;
 017451    000                          .BYTE   0                       ;
 017452    000                          .BYTE   0                       ;
 017453    000                          .BYTE   0                       ;
 017454    000                          .BYTE   0                       ;
 017455    000                          .BYTE   0                       ;
                                ;------------------------------------------------------------------------



                                .SBTTL  PATCH AREA
                                ;------------------------------------------------------------------------

 017456 000000                  PATCH:  0
        017660                          .=.+200
 017660 000000                  LASTAD: 0                               ;LAST ADDRESS NOT TO EXCEED 17776
                                ;------------------------------------------------------------------------

        000001                  .END
```

```
ABASE = 177170           6127#   6128
ACDW1 = 000000           6128
ACDW2 = 000000           6128
ACLOW = 000010           5988#   6684
ACPUOP= 000000           6128
ADDW0 = 000000           6128
ADDW1 = 000000           6128
ADDW10= 000000           6128
ADDW11= 000000           6128
ADDW12= 000000           6128
ADDW13= 000000           6128
ADDW14= 000000           6128
ADDW15= 000000           6128
ADDW2 = 000000           6128
ADDW3 = 000000           6128
ADDW4 = 000000           6128
ADDW5 = 000000           6128
ADDW6 = 000000           6128
ADDW7 = 000000           6128
ADDW8 = 000000           6128
ADDW9 = 000000           6128
ADEVCT= 000000           6128
ADEVM = 000000           6128
ADRS    005104           7088*   7096*   7104*   7114*   7119*   7124*   7133*   7140#   7158*
ADTST   002204           6211*   6423*   6435#
ADTSTX  002344           6455#
AD1     002234           6440#   6452
AD2     002334           6446    6453#
AENV  = 000000           6128
AENVM = 000000           6128
AFATAL= 000000           6128
AMADR1= 000000           6128
AMADR2= 000000           6128
AMADR3= 000000           6128
AMADR4= 000000           6128
AMAMS1= 000000           6128
AMAMS2= 000000           6128
AMAMS3= 000000           6128
AMAMS4= 000000           6128
AMSGAD= 000000           6128
AMSGLG= 000000           6128
AMSGTY= 000000           6128
AMTYP1= 000000           6128
AMTYP2= 000000           6128
AMTYP3= 000000           6128
AMTYP4= 000000           6128
ANSCHG  012650           8364*   8369#   8449
ANSDEF  012646           8363*   8368#   8446    8451
ANSWER  000530           6149#   6328    6335    6343    6352    6363    7878    8437*   8446*   8456*
APASS = 000000           6128
APRIOR= 000000           6128
APTCSP= 000040           6000#   8591
APTENV= 000001           5999#   6197    6243    8351
APTER   012570           8340    8351#
ASWREG= 000401           6125#   6128
ATESTN= 000000           6128
```

```
AUNIT = 000000           6128
AUSWR = 000000           6128
AVECT1= 000264           6126#   6128
AVECT2= 000000           6128
BADWRD  011540           8005*   8023    8046*   8064    8150    8153    8196#
BAEMPT  005120           7102    7146#
BAFILL  005116           7100    7145#
BA11    001770           6395#   6425
BA231   007076           7696#   7722
BB30    007742           7882#   7884
BC13    002470           6503#   6505
BC20    002622           6584#   6612
BC22    003332           6771#   6802
BC231   007126           7702#   7719
BDAUDL  010142           7944#   7952
BDBUDL  010146           7945#   7948
BE00    001306           6234    6236#   6266
BF00    001316           6238#   6240
BITCNT  011526           8164*   8169    8176*   8179    8181*   8191#
BITLIM  011532           8009*   8050*   8179    8193#
BITMSG  011530           8148*   8174    8185*   8192#
BITOFF  011534           8008*   8049*   8155    8157*   8162    8194#
BITPAS  011524           8147*   8182    8184*   8190#
BIT0  = 000001           5994    6042#
BIT00 = 000001           6042#
BIT01 = 000002           6042#
BIT02 = 000004           6042#
BIT03 = 000010           6042#
BIT04 = 000020           6042#
BIT05 = 000040           6042#
BIT06 = 000100           6042#
BIT07 = 000200           6042#
BIT08 = 000400           6042#
BIT09 = 001000           6042#
BIT1  = 000002           6042#
BIT10 = 002000           6042#
BIT11 = 004000           6042#
BIT12 = 010000           6042#
BIT13 = 020000           6042#   6447    7966
BIT14 = 040000           6042#
BIT15 = 100000           6042#
BIT2  = 000004           6042#
BIT3  = 000010           6042#
BIT4  = 000020           5992    6042#
BIT5  = 000040           5993    6042#
BIT6  = 000100           6042#   6214    7889    8319
BIT7  = 000200           6042#
BIT8  = 000400           6005    6042#   6231
BIT9  = 001000           6042#
BPTVEC= 000014           6042#
BSINIT  012262           6452    7540    7544    8269#
BSINTX  012400           8291#
BTRP    000320           6118#   6215*   6221*   6437*   6453*
BTRPFL  000570           6165#   6192*   6421*   6424    6445    8328*
BTRP2   000322           6119#   6216*   6222*   6438*   6454*
CKBITS  011314           8029    8070    8147#
```

```
CKSYAD  002422          6358*  6368*  6495#
CMD     000542          6154#  6774   6780   6784   6794   6803   6805   6861*  6862   6878*  6879   6887*  7055
                        7066   7077   7535*  7536   7991
CMDTBL  003772          6861   6878   6899#
CMPWRD  011536          8006*  8047*  8149   8195#
CMSGTB  011544          8007   8201#
CMSG10  011704          8207   8233#
CMSG11  011713          8208   8234#
CMSG12  011750          8209   8210   8235#
CMSG15  011770          8212   8236#
CMSG4   011624          8201   8227#
CMSG5   011634          8202   8228#
CMSG6   011641          8203   8229#
CMSG7   011662          8204   8230#
CMSG8   011665          8205   8231#
CMSG9   011675          8206   8232#
CR    = 000015          6042#
CRCALL= 000200          6001#  6345   6347   6979
CRCERR= 000001          5994#  7268
CRLF  = 000200          6042#
CSR     000534          6151#
CSRADR  010230          6667*  7153*  7945   7961#
CSRCHK  010404          7232*  7248*  7262*  7537*  8001#
CSRCKX  010616          8031   8035#
CSRCMP  000546          6156#  7227*  7228*  7505*  7506*  7509*  8003   8006   8019
CSRMSK  000544          6155#  7229*  7510*  8002
DBGFLG  000516          6088*  6144#  6190*  6370   7529
DBLDEN= 000400          6005#  6233
DC20    002762          6611#
DDISP = 177570          6042#
DELAY   010136          6670*  7155   7943#
DENBIT= 000400          5985#  6337   6339   7517
DENERR= 000020          5992#
DENS    000506          6140#  6233*  6235*  6887
DNBIT = 000040          5983#  6668   6672   7086   7227   7505   7970
DOUT    013572          8566#  8583   8585
DOUTD   013654          8581#
DOUT1   013606          8569#  8570   8578
DOUT2   013634          8572   8575#
DOUT3   013642          8574   8576#
DRIVE   011224          7345*  7353*  7356*  7363*  7371*  7374*  7440*  7450*  7710*  7711*  8092*  8094*  8112
                        8121#
DRIVE1= 000400          5991#  7516
DROPAV  006264          7432*  7441*  7451*  7455   7462   7467#
DROPDV  006036          7335*  7346*  7349*  7364*  7367*  7375   7379   7383#
DRVAVL  006040          7242*  7432#
DRVCNT  007260          7697*  7710   7712   7716*  7718   7725#
DRVDEN= 000040          5993#  7519
DRVDRP  005530          7239*  7255*  7272*  7335#
DRVPTR  007262          7696*  7703   7717*  7726#
DRVR    004402          6810*  7064#
DRVRDY= 000200          5987#  7436   7446
DRV1  = 000020          5996#  6780   7338   7434   7514   8090
DSWR  = 177570          6042#
DVSCDN  004366          6972*  6973*  6981   6995*  7001*  7032#
DVSEC   004372          6976*  6977*  6994*  6997*  6998   6999   7002*  7007*  7018*  7019   7034#
```

DVTKDN	004362	6970*	6971*	6983	6985*	6993*	7010	7012*	7017*	7030#
DVTRK	004378	6974*	6975*	6986*	6989	6990*	6991	7008	7036#	
EA00	001256	6194	6213	6230#						
EA10	001520	6329	6332#							
EA13	002510	6497	6499	6507#						
EA21	003246	6678	6692	6696#						
EA221	003746	6866	6872	6881	6884	6887#				
EA223	004450	7071	7073#							
EA225	005522	7241	7243	7253	7257	7259	7267	7274#		
EA232	007330	7752	7754#							
EA233	007466	7784	7792	7794	7800#					
EB10	001562	6338	6340#							
EB11	002136	6400	6421#							
EB20	002774	6583	6614#							
EB2253	006374	7518	7520#							
EB231	007242	7700	7720#							
EC00	001076	6203	6205#							
EC10	001624	6346	6348#							
EC222	004202	6988	6995#							
EC23	007056	7624	7638	7640	7645	7647	7654	7656#		
ED00	001212	6218	6221#							
ED10	001662	6355	6357#							
ED21	003210	6688	6690#							
ED222	004174	6992	6994#							
ED231	007222	7704	7716#							
EE221	003726	6875	6883#							
EE222	004274	7003	7005	7008#						
EE231	007162	7708	7710#							
EF20	002752	6598	6600	6607	6609#					
EG00	001350	6244	6246#							
EG222	004342	7015	7018#							
EG225	005504	7269	7271#							
EH2231	005000	7106	7121#							
EH2252	006260	7456	7464#							
EI00	001444	6262	6264#							
EK2231	004646	7101	7103#							
EMSGTB	011574	8048	8214#							
EMSG0	011777	8214	8238#							
EMSG1	012003	8215	8239#							
EMSG10	012134	8224	8248#							
EMSG11	012156	8225	8249#							
EMSG2	012014	8216	8240#							
EMSG3	012026	8217	8241#							
EMSG4	012035	8218	8242#							
EMSG5	012051	8219	8243#							
EMSG6	012066	8220	8244#							
EMSG7	012103	8221	8245#							
EMSG8	012115	8222	8246#							
EMSG9	012125	8223	8247#							
EMTVEC=	000030	6042#								
EM22	003550	6808	6810#							
ENDFTB	000634	6183#	6237							
ENDTSH	011130	8084	8097#							
END00	001452	6266#								
END223	005100	7121	7126	7130	7135#					
ERRBIT=	100000	5995#	6679							

```
ERRCHK  005164          6767*  7219#
ERRFLG  005526          7220*  7233*  7235*  7236   7249*  7251*  7252   7263*  7265*  7266   7277#
ERRSET  012240          6452   6677   6690   7238   7254   7271   7540   7544   8260#
ERRTYP  000604          6171#
ERRVEC= 000004          6042#
ERSETX  012260          8263#
ERSTAT  005106          7129*  7141#
ESR     000536          6152#
ESRCHK  010624          7234*  7250*  7264*  7541*  8042#
ESRCKX  011034          8072   8076#
ESRCMP  000552          6158#  7230*  7513*  7516*  7519*  8044   8047   8060
ESRMSK  000550          6157#  7231*  7520*  8043
GETANS  012624          6327*  6334*  6342*  6351*  6362*  7877*  8363#
GETCMD  003564          6773*  6855#
GTCMD   003556          6764*  6771   6772*  6788*  6790*  6798*  6800*  6801   6813#
GTSWR   002346          6372*  6462#
GTSYAD  001766          6357*  6367*  6394#
HDRFLG  000602          6170#  7219*  7533*  8083   8085*  8331   8333*
HT    = 000011          6042#
IA00    001020          6193#
IA10    001500          6328#
IA13    002424          6496#
IA20    002550          6573#
IA21    003060          6672#
IA22    003324          6769#
IA221   003624          6863#
IA222   004104          6979#
IA223   004424          7068#
IA225   005210          7224#
IA2251  005544          7338#
IA2252  006050          7434#
IA2253  006340          7514#
IA23    006572          7608#
IA232   007304          7749#
IA233   007344          7783#
IA30    007712          7878#
IB00    001034          6197#
IB10    001534          6335#
IB11    002012          6399#
IB13    002434          6498#
IB20    002600          6574   6578#
IB21    003120          6673   6679#
IB221   003642          6868#
IB222   004114          6981#
IB2231  004552          7090#
IB225   005300          7236#
IB2251  005554          7340#
IB2252  006060          7436#
IB2253  006356          7515   7517#
IB23    006620          7609   7611   7614#
IB231   007114          7699#
IB233   007352          7785#
IC00    001044          6199#
IC10    001576          6343#
IC21    003150          6684#
IC221   003654          6871#
```

IC222	004122	6983#						
IC2231	004670	7091	7107#					
IC225	005334	7225	7244#					
IC2251	005564	7342#						
IC2252	006070	7438#						
IC23	006650	7615	7617	7620#				
IC233	007420	7786	7793#					
ID00	001170	6217#						
ID10	001644	6352#						
ID20	002624	6579	6581	6585#				
ID21	003170	6685	6687#					
ID22	003350	6774#						
ID221	003716	6880#						
ID222	004162	6991#						
ID2231	004700	7109#						
ID225	005374	7252#						
ID2251	005632	7350#						
ID2252	006114	7437	7439	7442#				
ID23	006674	7621	7625#					
ID231	007130	7703#						
IE10	001714	6363#						
IE20	002646	6590#						
IE21	003216	6680	6691#					
IE22	003374	6780#						
IE221	003664	6874#						
IE222	004224	6999#						
IE2231	005032	7108	7127#					
IE225	005462	7266#						
IE2251	005672	7339	7358#					
IE2252	006134	7435	7446#					
IE23	006704	7627#						
IE231	007140	7705#						
IF10	001752	6364	6370#					
IF20	002676	6586	6588	6595	6597#			
IF21	003254	6697#						
IF22	003404	6782#						
IF222	004250	7000	7004#					
IF225	005422	7245	7258#					
IF2251	005702	7360#						
IF2252	006144	7448#						
IF23	006712	7629#						
IG00	001334	6243#						
IG20	002722	6602#						
IG22	003414	6784#						
IG222	004304	6980	7010#					
IG225	005470	7268#						
IG2251	005746	7368#						
IG2252	006166	7447	7449	7452#				
IG23	006722	7631#						
IH00	001406	6253#						
IH20	002634	6587#						
IH22	003424	6786#						
IH2231	004562	7092#						
IH2251	006000	7343	7348	7355	7357	7361	7366	7373	7375#
IH2252	006204	7443	7445	7453	7455#				
IH23	007020	7626	7648#						

```
II00     001432            6254    6261#
II20     002706            6599#
II22     003450            6781    6792#
IJ00     001262            6231#
IJ22     003460            6794#
IK22     003470            6796#
IK2231   004620            7098#
IL22     003520            6803#
IM22     003536            6807#
INITCK   003002            6572*   6577*   6663#
INITDN=  000004            5989#   6687
INITDP   003270            6663*   6674*   6681*   6693*   6697    6702#
INITPG   000514            6143#   6241*   6609*   6763    6855
INITTS   004360            6763*   6966    6969*   7027#
INTLV =  000002            6997    7024#
IOTVEC=  000020            6042#
LASTAD   017660            8698#
LA221    003636            6864    6867#
LA223    004442            7069    7072#
LA232    007322            7750    7753#
LA30     007766            7879    7887#
LB00     001146            6198    6214#
LB10     001554            6336    6339#
LB221    003734            6869    6885#
LB222    004210            6982    6997#
LB225    005326            7237    7242#
LB2251   005624            7341    7349#
LC00     001070            6200    6204#
LC10     001616            6344    6347#
LC222    004150            6984    6989#
LD10     001660            6353    6356#
LD22     003370            6775    6779#
LD2231   005002            7110    7122#
LD2251   005662            7351    7356#
LE20     002672            6591    6596#
LE2231   005052            7128    7131#
LE2251   005740            7359    7367#
LE23     007012            7628    7646#
LE231    007156            7706    7709#
LF       =  000012         6042#
LF22     003442            6783    6790#
LF23     006766            7630    7641#
LG20     002746            6603    6608#
LG222    004330            7011    7016#
LG2251   005774            7369    7374#
LG23     006760            7632    7639#
LH2231   004710            7093    7111#
LH23     007052            7649    7655#
LI22     003506            6793    6800#
LJ00     001302            6232    6235#
LK2231   004640            7099    7102#
LOOP     006406            7240*   7256*   7273*   7528#
LOOP1    006446            7536#   7540    7544
LOOP2    006520            7539    7541#
LOOP3    006564            7543    7545#
LPPRT    012402            8275    8290*   8293#
```

LSIFLG	000520	6145#	6201*	6219*	7913			
LSITRP	001200	6215	6219#					
MSGCD3	017254	7260	8671#					
MSGCD4	017311	7246	8672#					
MSGCD5	017346	7226	8673#					
MSGLP	017417	7534	8674#					
MSG1	013742	7981	8606#					
MSG10	014272	7789	7797	8615#				
MSG11	014326	6449	8616#					
MSG12	014367	6436	8617#					
MSG13	014411	8017	8058	8618#				
MSG14	014452	7694	8619#					
MSG15	014506	8130	8620#					
MSG16	014526	8133	8621#					
MSG17	014547	8111	8622#					
MSG2	014002	6462	8607#					
MSG20	014556	8116	8623#					
MSG21	014564	7985	7989	8021	8025	8062	8066	8624#
MSG22	014570	8129	8625#					
MSG23	014610	6695	8626#					
MSG24	014701	8336	8627#					
MSG25	014715	8316	8628#					
MSG26	014722	8013	8629#					
MSG27	014736	8054	8630#					
MSG28	014752	8279	8631#					
MSG29	014776	6683	8632#					
MSG3	014040	6195	8608#					
MSG30	015031	7972	8633#					
MSG31	015055	7975	8634#					
MSG32	015077	8635#						
MSG33	015121	7380	7463	8636#				
MSG34	015140	6501	8637#					
MSG35	015210	8167	8638#					
MSG36	015217	8639#						
MSG37	015223	8284	8640#					
MSG38	015244	8641#						
MSG39	015300	6330	8642#					
MSG4	014072	6402	8609#					
MSG40	015404	8643#						
MSG41	015702	7790	7798	7977	8646#			
MSG42	015723	8647#						
MSG43	015726	6325	8648#					
MSG44	015752	6689	8649#					
MSG45	016045	6332	8650#					
MSG46	016125	6340	8651#					
MSG47	016206	6349	8652#					
MSG48	016271	6360	8653#					
MSG49	016357	6582	8654#					
MSG5	014127	6417	8610#					
MSG50	016422	8655#						
MSG51	016425	7745	8102	8656#				
MSG52	016436	6676	8657#					
MSG53	016475	7754	8658#					
MSG54	016530	6686	8659#					
MSG55	016613	7458	8660#					
MSG56	016703	7876	8661#					

```
MSG57   016766            7880    8662#
MSG58   017065            6502    7881    8663#
MSG59   017117            7887    8664#
MSG6    014164            6246    8611#
MSG7    014202            8211    8423    8453    8612#
MSG8    014206            7623    8613#
MSG9    014246            7637    7644    7653    8614#
N     = 000011            6007#   6452#   6677#   6690#   7238#   7254#   7271#   7540#   7544#
NEWPRI  010074            7897*   7905*   7915*   7917    7920#
NSMSG   012200            8185    8251#
OCTP    013346            6250    6405    6414    7748*   7992    8104*   8113*   8118*   8132*   8135*   8289*   8339    8505#
OCTPE   013334            6465    7984    7988    8020    8024    8028    8061    8065    8069    8502#
OCTPE1  013352            8506#
OCTPG   013526            8513    8516    8523    8527    8532    8537    8540    8547#
OCTPGX  013566            8557#
OCTPG0  013544            8548    8550    8552#
OCTPG1  013550            8508    8553#
OCTPX   013524            8543#
OCTP0   013366            8504    8507    8510#
OCTP1   013410            8511    8515#
OCTP2   013416            8514    8517#
OCTP3   013512            8509    8541#
OFL     013570            8502*   8505*   8538*   8549    8552*   8559#
OPINT   001456            6223*   6324#
OTSYDN  007676            6245*   7876#
OUTCMD  004524            7079*   7086#
OUTSWD  005124            7089    7097    7105    7115    7120    7125    7134    7153#
PATCH   017456            8696#
PCNTR   000522            6146#   6230*   6249    6251*   6252
PIRQ  = 177772            6042#
PIRQVE= 000240            6042#
PRIHI   010020            7078*   7897#
PRILO   010034            7080*   7905#
PRTDRV  011154            7378*   7461*   7715*   8096*   8111#
PRTKSC  011232            7270*   8129#
PRTSYS  011132            6401*   6500*   6675*   6682*   6694*   7377*   7457*   7459*   7701*   7788*   7796*   8089*   8102#
PR0   = 000000            6042#   7905
PR1   = 000040            6042#
PR2   = 000100            6042#
PR3   = 000140            6042#
PR4   = 000200            6042#
PR5   = 000240            6042#
PR6   = 000300            6042#
PR7   = 000340            6042#   6204    6226    6229    7897
PS    = 177776            6042#
PSW   = 177776            6042#   6204*   6217*   7917*
PTSIDF  011230            7336*   7347*   7354*   7365*   7372*   7460*   7707*   7709*   8095*   8114    8123#
PTSYST  007062            7634*   7642*   7651*   7694#
PWRVEC= 000024            6042#
RDMNST= 000012            6003#   7224
RDSEC = 000006            6002#   6807    7258
RDYWD   010226            6668*   7086*   7094*   7103*   7111*   7118*   7122*   7131*   7154*   7156    7945    7954    7960#
REGS    000510            6141#   6397*   6404    6406    6422    6427
REGS0   000314            6116#   6212*   6397    6427*   6498    7927    7928
REGS1   000316            6117#
RESVEC= 000010            6042#
```

```
RXCSAD  000472  6134# 6210* 6212  6422* 6439  6666* 6667  6672  6679  6691  7088  7153  7156
                7927* 7983  8001  8027
RXDBAD  000474  6135# 6684  6687  7096  7104  7114  7119  7124  7133  7268  7436  7442  7446
                7452  7928* 7929* 7987  8042  8068
RXINIT= 040000  5997# 6665  6666
RXINT0  012404  6095  6208  6225  8299#
RXINT1  012412  6099  6228  8306#
RXVECT  000476  6136#
RX2BIT= 004000  5998# 6691  7228  7506
RYDLY   010224  7944  7959#
RYDX    010222  6669* 7943  7958#
SECADR  005114  7075* 7112  7144#
SECTR   000500  6137# 6998* 7004  7006* 7019* 7073  7075
SETDEN= 000010  6004# 7068  7244
SETMSG  012222  8148  8252#
SETPIX  010072  7916  7918#
SETPRI  010050  7898* 7906* 7913#
SIDE    011226  7344* 7352* 7362* 7370* 7712* 7713* 7714* 8117  8122#
SIDE1 = 001000  5986# 6784  6794  7340  7358  7512
SIDRDY= 000002  5990# 7442  7452
SSUCAD  010076  6664* 7064* 7221* 7433* 7926#
SSUCOF  010126  6396* 6426* 6671* 6765* 6768* 6776* 6779* 6859* 6978* 7337* 7698* 7926* 7936#
                8088*
STACK = 001000  6042#
START   001000  6081  6085  6090  6189#  8318
STFLG   000576  6168# 8442*
STKLMT= 177774  6042#
STSTBT  006266  7247* 7261* 7505#
SUC     000610  6173# 6348* 6365* 6570* 6575* 6589* 6601* 7612* 7618* 7695* 7720* 7721  7747
                7749  7787* 7795* 7936  8103
SWR     000504  6139# 6205* 6231  6324* 6337* 6339* 6345* 6347* 6359* 6369* 6447  6452  6466*
                6569* 6573  6585  6597  6979  7531  7540  7544  7610  7616  7620  7625  7629
                7751* 7753* 7966  7979  8010  8015  8051  8056  8269  8272  8277  8282  8329
                8342
SWREG   000324  6120# 6139  6324  6464  6466  6467
SW0   = 000001  6042#
SW00  = 000001  6042#
SW01  = 000002  6042#
SW02  = 000004  6042#
SW03  = 000010  6042#
SW04  = 000020  6042#
SW05  = 000040  6042#
SW06  = 000100  6042#
SW07  = 000200  6042#
SW08  = 000400  6042#
SW09  = 001000  6042#  6452  7540  7544  8277
SW1   = 000002  6042#
SW10  = 002000  6042#
SW11  = 004000  6042#
SW12  = 010000  6042#  8272
SW13  = 020000  6042#  8010  8051
SW14  = 040000  6042#  7979  8015  8056
SW15  = 100000  6042#
SW2   = 000004  6042#
SW3   = 000010  6042#
SW4   = 000020  6042#
```

```
SW5     = 000040          6042#
SW6     = 000100          6042#
SW7     = 000200          6001    6042#
SW8     = 000400          6042#
SW9     = 001000          6042#
SYSCHD    002514          6242*   6567#
SYSDRP    007264          7613*   7619*   7745#
SYSDVR    003272          6594*   6606*   6763#
SYSERR    000606          6172#   7783    7785    7793    7800*   7819*   7842*   7953*   8033    8034*   8074    8075*
SYSFLG    000574          6167#   6261    6611    7622*   7633*   7641*   7650*   7885*   7888*
SYSMSG    000572          6166#   6613    7623*   7637*   7644*   7653*
SYSTAT    006570          6610*   7607#
SYTMCK    007342          7639*   7646*   7655*   7782#
SYOTPT    003762          6857*   6860    6865*   6876*   6877    6882*   6885*   6892#
SY1TPT    003764          6858*   6893#
SOAV      000612          6174#   6571*   6578    6696*   6699*   6769    6782    6786    6792    6796    7342    7350    7360
                          7368    7379*   7438    7444*   7448    7454*   7462*   7608    7703    7705    7791*
SOCMD     000622          6178#   6665*   7065*   7222    7338    7340    7358    7434    7507    7511    7514    7517    7535
                          8090
SOCMDF    003766          6874    6883*   6886*   6894#
SODN      000626          6180#   6587    6777*   7631    7635*   7648    7652*   7699
SODVDN    003560          6771*   6814#   6870    6873*   6888*   6987*   7014*
SORDY     000616          6176#   6567*   6590    6592*   8299*
SOSEC     011310          7073*   8134    8140#
SOTRK     011304          7074*   8131    8138#
SOWMLT    000556          6160#   7070*   7072*   7816
S1AV      000614          6175#   6576*   6580    7614    7799*
S1CMD     000624          6179#
S1CMDF    003770          6895#
S1DN      000630          6181#   6599    7627    7636*   7643*
S1DVDN    003562          6815#
S1RDY     000620          6177#   6568*   6602    6604*   8306*
S1SEC     011312          8141#
S1TRK     011306          8139#
S1WMLT    000560          6161#   7839
TABADR    011542          8007*   8048*   8171    8197#
TBITVE=   000014          6042#
TCEX      013332          8481    8496#
TCRLF     013264          8483    8486#
TCRLFA    013302          8489#   8492
TEMP1     000562          6162#   8384*   8389    8398*
TEMP2     000564          6163#
TEMP3     000566          6164#   8581    8584
TEX       013740          8592    8597    8600#
TE2231    005042          7129#
TIB       000526          6148#   6504    7883    8313    8387    8392    8395    8405*   8406    8440    8444    8448*   8449
                          8451    8456    8464*   8468*   8471    8472*
TINER     013020          8394    8397    8411    8414    8423#
TKB       000302          6107#   8468
TKS       000300          6106#   6214*   7889*   7890*   8319*   8320*   8463*   8465*   8466
TKVEC   = 000060          6042#
TMSGAD    000600          6169#   6436*   7226*   7246*   7260*   7534*   8086    8334
TOB       000524          6147#   6236    7949*   8479*   8480    8482    8486*   8489*   8493*   8541*   8554*   8566*   8568*
                          8571    8576*   8595
TOERBT    010402          7954*   7970    7995#
TOG       013700          7950    8484    8487    8490    8494    8542    8555    8591#   8594
```

```
TOPRT    010232          7955    7966#
TOPRTX   010400          7968    7980    7993#
TPB      000306          6109#   8471*   8573*   8575*   8595*
TPS      000304          6108#   8469    8569    8593    8598
TPVEC =  000064          6042#
TRACK    000502          6138#   6989*   7008*   7013*   7016*   7074    7076
TRAP     012474          6437    8328#
TRAPVE=  000034          6042#
TRAPX    012566          8343    8345#
TRAP1    012532          8332    8336#
TRAP2    012550          8330    8340#
TRBIT =  000200          5984#   7094    7103    7111    7118    7122    7131
TRKADR   005112          7076*   7116    7143#
TRKSEC   004016          6809*   6966#
TRTVEC=  000014          6042#
TSHDCK   011042          7969    8012*   8053*   8083#   8271*
TSTERR   000554          6159#   8035*   8076*
TTAR     013036          8365*   8437#   8455
TTAR1    013066          8441    8444#
TTAR2    013106          8445    8448#
TTAR3    013146          8450    8452    8456#
TTIN     013156          6503*   7882*   8312*   8386    8439*   8463#
TTINT    012420          6074    8312#
TTIN1    013172          8466#   8467
TTIN2    013206          8469#   8470
TTITFG   000632          6182#   8313*   8314
TTOUT    013232          6196    6247    6326*   6331*   6333*   6341*   6350*   6361*   6403    6412    6450    6463    6501*
                         6502*   6614*   6676*   6683*   6686*   6689*   6695*   7380*   7458*   7463*   7694*   7746*   7755*
                         7789*   7790*   7797*   7798*   7876*   7880*   7881*   7887*   7973    7976    7978    7982    7986
                         7990    8014    8018    8022    8026    8055    8059    8063    8067    8087    8102*   8111*   8116*
                         8129*   8130*   8133*   8168    8173    8175    8280*   8285*   8316*   8335    8337    8424    8454*
                         8479#   8485    8495
TTR      012652          6410    6419    6471    8384#
TTR0     012660          8386#   8408
TTR1     012706          8388    8392#
TTR1A    012722          8393    8395#
TTR1B    012736          8396    8398#
TTR2     012774          8391    8409#
TTR3     013004          8410    8412#
TTR4     013014          8413    8415#
TTR5     013016          8390    8416#
TTWAIT   000532          6150#   6248*   6265*   8596
UA11     002154          6424#
UA231    007246          7721#
UC22     003512          6778    6785    6787    6789    6791    6795    6797    6799    6801#
UC231    007232          7718#
VALWD    005110          7132    7142#
VECT     000512          6142#   6398*   6413    6415    6420    6428
VECT0    000310          6114#   6224    6398    6428*
VECT1    000312          6115#   6227
WATCH0   007474          6596*   7806#
WATCH1   007572          6608*   7829#
WATIN0   007670          6593*   6766*   7806    7808*   7848#
WATIN1   007672          6605*   7829    7831*   7849#
WCNTMX   007674          7812    7835    7850#
WCNTR0   007566          7809*   7811*   7812    7814*   7822#
```

```
WCNTR1  007664          7832* 7834* 7835  7837* 7845#
WDCT    005122          7095  7147#
WDOT    000540          6153# 7077* 7087  7090  7092  7098  7107  7109  7127  7536*
WMLT0   007570          7810* 7815* 7816  7818* 7823#
WMLT1   007666          7833* 7838* 7839  7841* 7846#
WRDS    005102          7087* 7095* 7100* 7102* 7112* 7113* 7116* 7117* 7123* 7132* 7139# 7158
X     = 000001          6006#
XER     017446          7123  8681#
XTTAR   013154          8443  8447  8457#
XUDL    010220          7946  7956#
XUPTDV  011222          8115  8119#
XWAT0   007564          7813  7817  7820#
XWAT1   007662          7836  7840  7843#
X10     001764          6371  6373#
X11     002202          6429#
X12     002420          6472#
X13     002512          6508#
X20     003000          6615#
X21     003266          6698  6700#
X22     003554          6770  6804  6811#
X221    003760          6889#
X222    004356          6996  7009  7020#
X223    004522          7081#
X22311  005162          7157  7159#
X225    005524          7275#
X2251   006034          7376  7381#
X2252   006262          7465#
X2253   006404          7522#
X2254   006566          7530  7532  7546#
X23     007060          7657#
X231    007256          7723#
X232    007340          7756#
X30     010016          7886  7891#
$APTHD  000456          6130#
$BASE   000454          6128# 6210
$CPUOP  000426          6128# 6199
$DEVCT  000410          6128#
$ENDAD  001422          6069  6257#
$ENV    000420          6128# 6197  6243  8351
$ENVM   000421          6128# 8591
$ETABL  000420          6128#
$ETEND  000456          6128# 6130
$FATAL  000402          6128# 8260* 8353*
$HIBTS  000456          6130#
$MADR1  000432          6128#
$MADR2  000436          6128#
$MADR3  000442          6128#
$MADR4  000446          6128#
$MAIL   000400          6128# 6130
$MAMS1  000430          6128#
$MAMS2  000434          6128#
$MAMS3  000440          6128#
$MAMS4  000444          6128#
$MBADR  000460          6130#
$MSGAD  000414          6128#
$MSGLG  000416          6128#
```

```
$MSGTY   000400            6128#   8261*   8354*
$MTYP1   000431            6128#
$MTYP2   000435            6128#
$MTYP3   000441            6128#
$MTYP4   000445            6128#
$PASS    000406            6128#   6191*   6252*
$PASTM   000464            6130#
$SVPC  = 000046            6069#
$SWREG   000422            6128#   6205
$TESTN   000404            6128#
$TSTM    000462            6130#
$UNIT    000412            6128#
$UNITM   000466            6130#
$USWR    000424            6128#
$VECT1   000450            6128#   6206
$VECT2   000452            6128#
.      = 017662            6046#   6069#   6073#   6079#   6083#   6087#   6094#   6098#   6102#   6124#   6130#   6185#   8675#
                           8697#
.$X    = 000456            6130#
```

```
COMMEN   1526#  6042#
ENDCOM   1538#  6042#
ERNUM    6028#  6452   6677   6690   7238   7254   7271   7540   7544
ERR      6009#  6452   6677   6690   7238   7254   7271   7540   7544
ERROR    6042#
ESCAPE   1654#  6042#
GETPRI   1278#  6042#
GETSWR   1725#  6042#
MULT     4393#  6042#
NEWTST   1585#  6042#
POP      2103#  6041#  6042#
PUSH     2095#  6041#  6042#
REPORT   5352#  6042#
SCOPE    6042#
SETPRI   1246#  6042#
SETUP    1302#  6042#
SKIP     1688#  6042#
SLASH    1478#  6042#
SPACE    6042#
STARS    1447#  6041#  6042#  6069   6128   6130
SWRSU    1416#  6042#
TTYOUT   6035#  6501   6502   6676   6683   6686   6689   6695   7380   7458   7463   7694   7789   7790   7797
         7798   7876   7880   7881   7887   8102   8111   8116   8129   8130   8133   8316
TYPBIN   2039#  6042#
TYPDEC   2009#  6042#
TYPNAM   1779#  6042#
TYPNUM   1976#  6042#
TYPOCS   1929#  6042#
TYPOCT   1892#  6042#
TYPTXT   1846#  6042#
$$ESCA   1667#  6042#
$$NEWT   1621#  6042#
$$SKIP   1701#  6042#
.EQUAT    176#  6042
.HEADE     50#
.KT11     319#
.SETUP   1180#
.SWRHI     92#
.$ACT1   4961#  6041#  6069
.$APTB   5005#  6041#  6128
.$APTH   5261#  6041#  6130
.$APTY   5436#  6041#
.$ASTA   5307#
.$CATC    905#
.$CMTA   1016#  6041#
.$DB2D   4591#
.$DB2O   4714#
.$DIV    4494#
.$EOP    2162#
.$EQUA   6041#
.$ERRO   2643#
.$ERRT   2838#
.$MULT   4431#
.$POWE   4143#
.$RAND   4218#
.$RDDE   3814#
```

```
.$RDOC   3723#
.$READ   3328#
.$R2AZ   4858#
.$SAVE   3889#
.$SB2D   4675#
.$SB2O   4776#
.$SCOP   2397#
.$SIZE   4271#
.$SUPR   4814#
.$TRAP   3991#
.$TYPB   3221#
.$TYPD   3144#
.$TYPE   2925#
.$TYPO   3048#
.$40CA    944#
.1170     498#


. ABS.  017662     000


 ERRORS DETECTED:  0

 CZRXEA,CZRXEA/CRF=CZRXEA.SML,CZRXEA.P11
 RUN-TIME: 16 14 1 SECONDS
 RUN-TIME RATIO: 38/31=1.2
 CORE USED:  35K  (70 PAGES)
```