SATURN SOFTWARE LIMITED

PRESENTS

>>>> X-RAY <<<<

RELEASE 2.0

BY

Ralph Deane

&

Jack Brown

Distributed by:

SATURN SOFTWARE LIMITED
POST OFFICE BOX 397
NEW WESTMINSTER, BRITISH COLUMBIA
V3L 4Y7, CANADA

DISCLAIMER

SATURN SOFTWARE LIMITED and the authors make no warranties, either expressed or implied, with respect to this manual, or with respect to the software it describes, or its quality, performance, or suitability for any particular application. In no event will SATURN SOFTWARE LIMITED or the authors be liable for any direct, indirect, incidental, or consequential damages resulting from any defects in the manual or software supplied.

While this software package is now operational of the author's 32K SYM-1/KTM-2 system, it should be expected that the purchaser may have to provide custom I/O drivers to match his/her particular terminal and printer configuration.

COPYRIGHT NOTICE

X-RAY 2.0 COPYRIGHT (C) January 1, 1982

BY SATURN SOFTWARE LIMITED (ALL RIGHTS RESERVED)

This manual describing X-RAY, and the accompanying cassette containing the X-RAY object code and source files, are copyrighted, and are provided for the personal use and enjoyment of the original purchaser only. All rights are reserved. Reproduction by any means whatsoever, without the prior written consent of SATURN SOFTWARE LIMITED, is strictly prohibited. The original purchaser is, however, permitted to make backup copies of the cassette software to protect against accidental loss or erasure. The use of X-RAY for the promotion of sales of microcomputer hardware and equipment is strictly prohibited without the prior written consent of SATURN SOFTWARE LIMITED.

Address all communications to:

John W. Brown, President
SATURN SOFTWARE LIMITED
POST OFFICE BOX 397
NEW WESTMINSTER, BRITISH COLUMBIA
V3L 4Y7, CANADA

Introduction

Extended RAE (X-RAY) is a program designed to extend and enhance Synertek's RAE-1 ROM(s). It adds a very fine line editor, much needed by RAE, and also adds more editing commands. This is all integrated in such a way as to appear to be part of RAE. X-RAY makes developing a program much easier as it fills many deficient areas in RAE.

Equipment Required

X-RAY requires the following minimum hardware in order to execute properly :

1. a SYM-1 computer with the RAE-1 ROM(s) installed

2. at least 8K of RAM, which is sufficent to hold the X-RAY code and provide a decent text file area.

3. a KTM -2/80 or equivalent terminal. Conversely, any terminal can be used but some line editing features are lost or diminished.

4. (optional) a printer attached to the SYM-1 20 ma current loop.

5. (disk version) a HDE minidisk system with FODS V3.1 at $7380. X-RAY is set up for a two drive system and will have to be modified for more or less drives.

System Startup

The X-RAY system is entered at the NEW.COLD address. This will initalize the necessary pointers and values required by a cold start. Disk versions will also turn off the drives and set the default drive to number 2.

The accompanying cassette contains in the following ordrer:

a) Three copies to the cassette version object ID=C0
b) Three copies of the HDE disk version object ID=D0
c) Two sets of the X-RAY source files F10, F20, F30, and F40.

Cassette based users can get started with .L2 C0 (ret) followed by .G 200 when the succesful load is complete. Disk based users must use .L2 D0 (ret) and then wait for the cassette version of the object to be by-passed. The object can then be saved to disk from FODS with:

```
ENT 1/$0200$0BDD=%XRY (ret)
```

A warm start can be done by entering at the address NEW.WARM. This will do a normal RAE warm start but will also reinitalize the cassette ports so that motor control is still enabled.

After either a cold or warm start you enter the RAE editor program. The system status data is printed out, along with the '>' prompt. At this point you are now in the X-RAY line editor.

Line Editor

The X-RAY line editor started life as the Super Terminal Patch by Jack Brown (Saturn Softnews Vol 1 No 3). It has since been modified and expanded to improve its features.

This line editor uses control codes to move the cursor to various points in the input line, for insertions and/or deletions. For this reason, a terminal with an addressable cursor is desirable. However, as will be seen later, almost any terminal can be used with only minor changes.

The control codes used by the line editor, together with their actions, is documented in the following :

Control-A

Move the cursor to the start of the input line. This does not clear or cancel the current line contents.

Control-B

Insert a special blank character into the line buffer and then do a carriage return. This is used to insert a blank line into a file.

Control-C

Cancel and exit auto line mode. This is equivelent to the standard sequence :

```
>// <ret>
>AU <ret>
```

used by RAE.

Control-D

Change the direction of the scrolling editor to down (smaller line numbers) and then do a carriage return.

Control-H

Back up the cursor one position on the screen and in the line buffer. This does not affect the contents of the line. If the cursor is already at the start of the line, it will wrap around and appear at the end of the line.

Control-I (TAB)

Move the cursor forward one position in the buffer and on the screen, updating if necessary. The line contents are not affected. If the cursor is at the end of the line, it will wrap around and appear at the start.

Control-J (LineFeed)

Put the system into auto line mode with a step size of 1. If linefeed is typed at the end of a line (instead of carriage return) the line contents will be processed before the auto line prompt.

Control-M (Carriage Return)

Send the buffer contents to the command processor (X-RAY and RAE) for processing. The line is truncated at the current cursor position.

Control-P

Toggle the RAE hardcopy flag on and off. The line count is set to 4 and the page count to 0, avoiding a page number until the second page is printed.

Control-S

Escape to the SYM monitor. To return to X-RAY type : G <ret> . Monitor output will be echoed to the printer if the hardcopy flag is set.

Control-T

Toggle the cassette motors on and off. This is identical to the RAE Control-T except that the characters are not echoed to the screen.

Control-U

Change the direction of the scrolling editor to up (larger line numbers) and then do a carriage return.

Control-X

Cancel the current line of input. This will also cancel the scrolling editor. Note that control is not returned to RAE, only to the start of the line editor.

Control-Y

Exit to the SYM monitor for one command. Upon completion of the command control is return to the line editor.

Control-Z

Move the cursor from its current position to the end of the line.

The screen is updated if necessary.

Escape (ESC or ASCII $1B)

Escape is always followed by another character. It will move the cursor through the line and stop just after the first occurance of that character.

Delete (DEL or ASCII $5F or $7F)

Delete the character to the left of the cursor and close up the space in the buffer. The screen is not updated.

All other control codes are ignored. All printable characters are inserted into the line at the current cursor position. The screen is not updated after an insertion, so it may appear that you are overtyping existing characters. A Control-Z will update the screen after an insertion.

For those of you who do not have a terminal with an addressable cursor, the line editor can still be used with minor changes. The location addressed by the label KTMFLG holds a flag which indicates the type of terminal being used. If KTMFLG is set to $01 then an addressable cursor terminal is assumed. When KTMFLG is set to $00 then a dumb (non-addressable cursor) terminal is being used. In the dumb terminal mode the line editor is slightly different. The Control-H command is inactive and is ignored if input, and the Delete command will output a '®' instead of backing up the cursor. The other control codes execute the same as previously described.

New Commands

The X-RAY system does more than just add a line editor to RAE. It also adds a command processor and eight new commands. The commands use RAE format and are indistinguishable from the existing commands. These new commands fill some large gaps in RAE's editing function and will make file preparation much easier.

The new commands, their two letter codes, and description are as follows :

ADDRESS (AD)

used in the form : AD line#

This command will output the starting address of the line which is numbered 'line#'. If no such line exists, the command is ignored.

APPEND (AP)

used in the form : AP line#

This is the scrolling editor. It is used to step through a file line by line, all the while remaining in line edit mode. APPEND will search

for the line numbered 'line#' and, if found, will dump it to the line buffer. At this point you can edit the line contents using the line editor. When you are finished editing, you enter carriage return, line feed, Control-B, Control-D or Control-U to terminate the line. The action of each terminator is :

Carriage Return - this is the normal terminator. It re-enters the line into the file and continues scrolling in the preset direction. The default direction is set to up - ie. increasing line numbers.

Control-D - this is used to change the scrolling direction to down - ie. decreasing line numbers. The line currently in the buffer is re-entered into the file.

Control-U - this will change the scrolling direction to up. The current line buffer contents are re-entered into the file.

Line Feed - this re-enters the current line into the file and then enters auto line mode with a step size of 1. At this point you are no longer in the APPEND command but are in normal auto line entry mode. New lines are now created and entered into the file. Auto mode is terminated by a Control-C, at which time you re-enter APPEND mode with no change in the previous direction.

After entering the terminator, the next line in sequence is dumped to the line buffer and the action is repeated. The scrolling edit is terminated when you :

a. reach the end of the file (scrolling up).
b. reach the start of the file (scrolling down).
c. enter a Control-X.

A Control-X cancels APPEND mode without altering the contents of the current line.

EXECUTE (EX)

used in the form : EX addr

This command is used to switch input from the keyboard to a RAE file. The file, which is located starting at 'addr', contains commands for RAE to execute. Each command must be entered on a separte line of the file. Input will continue from the file, errors or not, until the file end is reached. Input then resumes from the keyboard.

FILE (FL)

used in the form : FL addr

This command is used to reinstate a text file. The new file is assumed to start at 'addr' and the text-start pointer is set to that address. The file is then searched to find the end address, which is stored in the end-of-text pointer. This will recover any RAE file in memory. This command will not recover an accidentally cleared file.

RESTORE (RE)

used in the form : RE

This command is used to recover a file after a cold start or an accidental CLEAR. It will remove the EOF marker that RAE put at the start of the file and will set the end-of-text pointer to the proper address. It is assumed that the file starts at the current text start address. This command will only work properly if the first line in the file has only one character.

SAVE (SA)

used in the form : SA id addr1 addr2

This command is used to save object code onto tape. It is equivalent to the SYM monitor command :

```
.S2 id,addr1,addr2
```

The tape startup delay is reset to the proper value by SA so that it is possible to read the tape thus produced. This was not always the case when exiting RAE to save code using S2.

SORT (SO)

used in the form : SO

The SORT command is used to do a alphabetical name sort on the label file. It is used after Pass 1 of an assembly to produce a sorted symbol table at the end of Pass 2. SORT uses the relocating buffer area as temporary storage so the buffer pointer must point to usable memory. The sort routine is a modified "hypersort" originally written by J. CYR and published in RAE NOTES issue number 3.

TAPE (TA)

used in the form : TA 1 or TA 2

This command allows you to change the speed of the SYM cassette tape interface. TA 2 sets the speed to twice normal (SYM x 2) while TA 1 resets it to normal (SYM x 1). The other commands affected by the speed up/down are : GET PUT SAVE and the monitor commands L2 and S2. Some SYM systems do not work reliably at the faster speed and some tweeking of the speed values may be needed. The high speed values are set in the routine addressed by the label SPD2.

As well as adding these new commands, X-RAY also modifies the RAE line edit command EDIT. If the form of EDIT is :

```
ED line#
```

then the X-RAY version is executed. The other version of EDIT is unchanged. The new ED command will dump the line numbered 'line#' to the input buffer, where you can edit it using the line editor commands. The line is re-entered into the file by a carriage return.

Other Features

The X-RAY system has other features besides those already described. Some of these are :

1. All I/O is vectored through a series of jumps at the start of the program. For those with non-standard I/O devices these vectors should be changed to point to your routines. Do not change the SYM I/O vectors, as X-RAY with not run if you do.

2. A printer interface is built-in to X-RAY. This assumes a 2400 baud device attached to the SYM current loop, with the ready flag on bit 0 of VIA #1 Reg B.

3. The baud rate of the terminal and printer can be changed by changing the value of PBAUD (printer) and KBAUD (terminal). The default values are 2400 and 4800 baud, respectively.

4. In the cassette version, the continue-on-tape (.CT) patch is built-in and is initalized by either a warm or cold start.

5. The start and end addresses of the text and label files can easily be changed for new cold start values. The values are in a table at D.TAB.

6. The disk version uses a modified version of Tom Getty's interface. This interface implements the LOD, ENT and DC commands, as well as a continue-on-disk feature for the assembler. A description is as follows :

LOD - used in the form : LOD drive# name

This will load the file called 'name' from the drive numbered 'drive#'. If no drive# is specified then drive 2 is accessed. If no name is provided a 30 error occurs. If the file excedes the current text file boundry a 31 error occurs. When 'name' is preceded by a '+' the file is appended to the current file already in memory.
Examples :

```
LOD 1 FILE     -drive# 1
LOD 2 TEXT     -drive# 2
LOD HDEX       -drive# 2
LOD 1 +FILE    -append to current file
```

ENT - used in the form : ENT drive# name

This is used to save the current file in memory onto 'drive#' with a file name of 'name'. If no drive# is specified then drive 2 is selected. If 'name' is omitted a 30 error occurs.
Examples :

```
   ENT 1 FILE     -drive# 1
   ENT 2 TEXT     -drive# 2
   ENT HDEX       -drive# 2
```

DC - used in the form :   DC command-string

This command is used to execute FODS disk commands. The entire command-string is passed to FODS for processing. If no command-string is supplied a 32 error occurs.

Examples :

```
   DC DEL 2/:TEXT
   DC DIR 2
   DC FRE 1
   DC ENT 2/$1000$210F=%OBJ
```

Please note that all FODS commands, except CPY, will return to X-RAY. CPY will cause a return to FODS as it modifies the break vector. A Control-C exit from FODS and a X-RAY warm start are required to return properly.

Continue-on-disk

The .CT operation in the assembler has been modified to work with both tape and disk. If the form used is :

```
    .CT
```

then it is assumed to be a continue-on-tape call and the file is read from the cassette. If the form is :

```
    .CT drive# name
or  .CT name
```

it is assumed to be a disk call. The file called 'name' is read from disk off of drive number 'drive#' or, in the second case, drive 2.

Extending X-RAY

The command processor portion of X-RAY is easily expanded to include more RAE-style commands. This section will show the procedure necessary to add a new command to X-RAY and will detail some useful built-in subroutines.

The new command to be shown here is a variable tone alarm. It is based on the shift register in VIA #3 ($A800). It is necessary to have an amplifier hooked to VIA #3's CB2 line (AA-5) for this alarm to sound.

The name of this command will be ALARM (AL) and it will be used in the form : AL freq where 'freq' is a value between 1 and 255 ($01 to $FF). This value will determine the alarm frequency. Once the alarm command is executed it will continue to sound until a key is typed on the keyboard. The source program for ALARM is :

```
ALARM LDA #$00
      STA $A808
      LDA #$10
      STA $A80B ;TURN ON
      LDA #$0F
      STA $A80A ;SHIFT VALUE
      JSR GET.N ;GET FREQ
      LDA *PROC>ADDRS
      STA $A808
      JSR INTCHR ;WAIT FOR KEY
      LDA #$00
      STA $A80B ;TURN OFF
      JMP PRMT.OT
```

A few notes :

1. All command routines must end with a jump to either PRMT.OT or NEW.HOT. NEW.HOT should be used if the stack has been messed up by the command routine. Otherwise use PRMT.OT.

2. When a command routine is executed the Y register points to the next 'field' in the input buffer, CRT. If Y >= $50 the end of the line has been reached.

Once the command routine has been written, it is time to add the command name to the command table. The command table is located at the address CMD and contains name and routine address pairs for all of X-RAY's commands. The table terminates with a $00. A new command is added to X-RAY by inserting its two character name and address pair into the table. The name of our alarm command is AL and its address is ALARM. The new command would now look like :

```
CMD .BY 'ED'
    .SI EDIT
       .
       .
    .BY 'SA'
    .SI SAVE
    .BY 'AL'
    .BY ALARM
    .BY $00
```

This is all that is required to interface a new command to X-RAY.

The routines which are useful in adding new commands are :

GET.N - this subroutine will get the number in the field addressed by Y into PROC>ADDRS (low) and PROC>ADDRS+1 (high). The number may be decimal or hex.

ASC.SP>DEC - get the line number from the field pointed to by Y. Register X must be set to $08 for the line number to be put into FIRST, FIRST+1 and be set to $0A for it to go into LAST, LAST+1.

SET.F2 - get the address of the line whose number is in FIRST, FIRST+1 and place it in SCRAT0, SCRAT0+1. If the flags Z=0 or C=1 then the line was not found.

PAS.SPCS - increment Y until it points to the next non-blank character in the line.

NXT.FLD - increment Y until it points to the next field in the line. If Y >= $50 the end of the line has been reached.

A study of the existing X-RAY command routines will show examples of the usage of these routines.

```
>

>PA
```

```
0001            .LS
0010 ;
0020 ;*************************************
0030 ;**                                 **
0040 ;**  EXTENDED ENHANCEMENTS (XRAE)   **
0050 ;**  FOR SYNERTEK ASM/TED 1.0       **
0060 ;**                                 **
0070 ;**     COPYRIGHT (C) 1982          **
0080 ;**    SATURN SOFTWARE LIMITED      **
0090 ;**                                 **
0100 ;**       DISK FILE XRAY1           **
0110 ;**       FEBRUARY 17, 1982         **
0120 ;**                                 **
0130 ;*************************************
0140 ;
0150 ; CONDITIONAL ASSEMBLY CONTROL VARIABLE
0160 ; SET FODS EQUAL TO ONE FOR HDE DISK VERSION
0170 ; SET FODS EQUAL TO ZERO FOR CASSETTE VERSION
0180 ;
0190 FODS       .DE $0001      ;SET FOR DISK VERSION
0200 ;
0210 TOPMEM     .DE $6000      ;HIGH MEMORY
0220 ORIGIN     .DE $200       ;START OF PROGRAM
0230 STEXT      .DE END.PGM    ;START OF TEXT FILE
0240 ETEXT      .DE TOPMEM-$2003    ;END OF TEXT FILE
0250 SLABEL     .DE TOPMEM-$2000    ;START LABELS
0260 ELABEL     .DE TOPMEM-$0103    ;END OF LABELS
0270 BUFFER     .DE TOPMEM-$0100    ;RELOCATING BUFFER
0280 ;
0290            .BA ORIGIN
0300            .OS
0310            .CE
0320 ;
0330 ; HDE DISK EQUATES
0340 ;
0350            IFE FODS-1
0360 EOT        .DE $00A8      ;END FODS FILE
0370 TXBUF      .DE $7280      ;FODS TEXT BUFFER
0380 SET2       .DE $7634      ;SELECT DRIVE 2
0390 DSKRW      .DE $765D      ;DRIVE CONTROL
0400 CMDINT     .DE $79B3      ;COMMAND INTERPRETER
0410 FODBRK     .DE $7AF0 ;
0420 DISCC.VEC  .DE $00EC      ;DISK COMMAND VECTOR
0430 DISC1      .DE $00F0      ;DISK OUT VECTOR
0440 DISC2      .DE $00F2      ;DISK IN VECTOR
0450            ***
0460 ;
0470 ; PAGE ZERO DEFINITIONS
0480 ;
0490 CTROLYVEC  .DE $0000      ;USER CONTROL Y VECTOR
0500 USERVEC    .DE $0003      ;USER COMMAND VECTOR
0510 ;
0520 EXPTR      .DE $00FA      ;START OF SCRATCH PAD AREA
0530 UPDFLG     .DE $009E      ;UPDATE LINE FLAG
0540 YMAX       .DE $009F      ;LINE LENGTH REACHED
0550 SCRATCH    .DE $00B0
```

```
0560 LEN         .DE $00B1     ;PROTECTED POINTER
0570 APP         .DE $00B2     ;APPEND FLAG
0580 DIREC       .DE $00B3     ;DIRECTION OF SCROLL
0590 EXFLG       .DE $00B4     ;EXECUTE FILE FLAG
0600 NUMFLG      .DE $00B5     ;SKIP LINE# FLAG
0610 ;
0620 PRTVEC      .DE $00B6     ;USER PRINTER VECTOR
0630 SAVEYY      .DE $00C7     ;Y STORAGE AREA
0640 PUREC       .DE $00C8     ;INPUT BUFFER LOCATION
0650 PROC>ADDRS .DE $00D1     ;NUMBER STORAGE FROM RAE
0660 TPRES       .DE $00D3     ;PRESENT END OF TEXT
0670 ERRORS      .DE $00DB     ;NUMBER OF ERRORS
0680 SCRATO      .DE $00DD     ;TEMPORARY STORAGE
0690 SUP.OUT     .DE $00E3     ;SUPPRESS OUTPUT FLAG
0700 DISCI       .DE $00EE     ;TAPE/DISK INPUT FLAG
0710 DISCO       .DE $00EF     ;TAPE/DISK OUTPUT FLAG
0720 DISCI.VEC   .DE $00F6     ;TAPE/DISK INPUT VECTOR
0730 CURNT       .DE $00FC     ;CURRENT LABEL POINTER
0740 NXT.PTR     .DE $00FE     ;NEXT LABEL POINTER
0750 ;
0760 CRTI        .DE %10000000         ;CRT IN BIT
0770 TTYI        .DE %01000000         ;TTY IN BIT
0780 TTYO        .DE %00100000         ;TTY OUT BIT
0790 CRTO        .DE %00010000         ;CRT OUT BIT
0800 ;
0810 VIADRB      .DE $A000     ;DATA REGISTER
0820 VIADDB      .DE $A002     ;DATA DIR REGISTER
0830 HSBDRY      .DE $A632     ;HIGH SPEED BOUNDRY
0840 TAPET1      .DE $A635     ;TAPE VALUE
0850 TAPET2      .DE $A63C     ;TAPE VALUE
0860 EAL         .DE $A64A     ;END ADDRESS
0870 EAH         .DE $A64B
0880 SAL         .DE $A64C     ;START ADDRESS
0890 SAH         .DE $A64D
0900 ID          .DE $A64E     ;ID NUMBER
0910 TAPDEL      .DE $A630     ;TAPE DELAY
0920 SDBYT       .DE $A651     ;SPEED BYTE
0930 TECHO       .DE $A653     ;MON ECHO FLAG
0940 TOUTFL      .DE $A654     ;TERM OUTPUT FLAG
0950 INVEC       .DE $A660     ;TERMINAL INPUT LINK
0960 OUTVEC      .DE $A663     ;TERMINAL OUTPUT LINK
0970 UBRKVC      .DE $A676
0980 ;
0990 ;RAE PAGE ONE VARIABLES
1000 ;
1010 TXST        .DE $0100     ;START OF TEXT FILE
1020 TXEN        .DE $0102     ;END OF TEXT FILE
1030 STST        .DE $0104     ;LABEL FILE ADDRESS
1040 STEN        .DE $0106     ;END OF LABEL FILE
1050 FIRST       .DE $0108     ;FIRST LINE#
1060 LAST        .DE $010A     ;SECOND LINE#
1070 INCBY       .DE $010C     ;INCREMENT FOR AUTO
1080 MANU        .DE $010E     ;MANUSCRIPT FLAG
1090 FORMAT      .DE $010F     ;FORMAT FLAG
1100 FILE.NO     .DE $0110     ;CURRENT FILE NUMBER
1110 HEX/DEC     .DE $0111     ;HEX/DECIMAL FLAG
1120 PASS        .DE $0113     ;ASM PASS 1/2 FLAG
1130 CON.TAPE    .DE $0114     ;.CT FLAG
```

```
1140 AUTO        .DE $0115     ;AUTO LINE PROMPT FLAG
1150 ADDPAD      .DE $011A     ;NUMBER OF CHARACTERS
1160 PRINT/CTL   .DE $011F     ;HARD COPY FLAG
1170 LINE/CNT    .DE $0120     ;CURRENT LINE COUNT
1180 PAGE/NUM    .DE $0121     ;CURRENT PAGE NUMBER
1190 TSTART      .DE $0124     ;TAPE START ADDRESS
1200 CRT         .DE $0135     ;CRT BUFFER
1210 ;
1220 ;SUPER MON ROUTINES
1230 ;
1240 USRENT      .DE $8035     ;USER ENTRY TO MON
1250 SAVER       .DE $8188     ;SAVE REGS ON STACK
1260 RESXAF      .DE $81B8     ;RESTORE ALL BUT R(A)
1270 RESXF       .DE $81BE     ;RESTORE ALL BUT FLAGS
1280 NIBASC      .DE $8309 ;
1290 ACCESS      .DE $8B86     ;UNPROTECT SYS RAM
1300 B.EEP       .DE $8972     ;BEEP THE BEEPER
1310 SPACE       .DE $8342     ;SPACE SUBROUTINE
1320 T.OUT       .DE $8AA0     ;OUTPUT BYTE TO TERM
1330 INT.CHR     .DE $8A58     ;INPUT BYTE TO TERMINAL
1340 GETCOM      .DE $80FF     ;MONITOR GET COMMAND
1350 DISPAT      .DE $814A     ;DISPATCH COMMAND
1360 ERMSG       .DE $8171     ;MONITOR ERROR ROUTINE
1370 DUMPT       .DE $8E87     ;TAPE WRITE
1380 ;
1390 ;RAE ENTRY POINTS AND SUBROUTINES
1400 ;
1410 ERROR       .DE $B00E     ;ERROR VECTOR
1420 RAE.COLD    .DE $B04B     ;COLD ENTRY
1430 RAE.HOT     .DE $B05E     ;RAE WARM WITHOUT MESSAGE
1440 CL.TXT      .DE $B096     ;CLEAR TEXT FILE
1450 RAE.WARM    .DE $B0AE     ;RAE WARM ENTRY POINT
1460 PAS.SPCS    .DE $B502     ;SKIP SPACES
1470 NXT.FLD     .DE $B4FF     ;SKIP TO NEXT FIELD
1480 ERROO       .DE $B44E     ;ERROR ROUTINE
1490 ASC.SP>DEC .DE $B2E6     ;GET LINE #
1500 SET.F2      .DE $B214     ;FIND ADDRESS OF LINE
1510 OUT.BYT     .DE $E3E2     ;OUTPUT A AS HEX BYTE
1520 CRT.IN      .DE $B5BB     ;OLD FILL BUFFER ROUTINE
1530 CL.STAB     .DE $BF37     ;CLEAR SYMBOL TABLE
1540 MRK.END     .DE $E05F     ;MARK END
1550 PR.LAB.S    .DE $E24A     ;GET NUMBER VALUE
1560 TAPE1.OFF   .DE $E318     ;TURN TAPE1 OFF
1570 TAPE1.ON    .DE $E32A     ;TAPE 1 ON
1580 CRLF        .DE $E3CA     ;RAE'S CR LF
1590 TOG1        .DE $EC59     ;TOGGLE MOTOR 1
1600 TOGO        .DE $EC64     ;TOGGLE MOTOR 0
1610 TLOAD       .DE $EF68     ;ENTRY FOR TAPE LOAD
1620 TAPE.FIN    .DE $EF80     ;TAPE FINISH
1630 ;
1640 ;CONTROL CHARACTERS
1650 ;
1660 DEL         .DE $7F       ;DELETE CHARACTER
1670 ALT         .DE $5F       ;ALTMODE DELETE
1680 BKSL        .DE $5C       ;BACK SLASH FOR TTYOUT
1690 BEL         .DE $07       ;BELL CHARACTER
1700 BS          .DE $08       ;BACKSPACE
1710 SPC         .DE $20       ;SPACE CHARACTER
```

```
                    1720 ;
                    1730 ;
                    1740 ;START OF SUPER TERMINAL CONTROL PATCH
                    1750 ;
                    1760 ;NOTE NEW STARTUP VECTORS
                    1770 ;
0200- 4C 18 02      1780 NEW.COLD      JMP COLD       ;NEW STARTUP ROUTINE
0203- 4C 60 02      1790 NEW.WARM      JMP WARM       ;NEW WARMUP ROUTINE
0206- 4C 5E B0      1800 NEW.HOT       JMP RAE.HOT    ;NO MESSAGE RETURN
0209- 4C 58 8A      1810 INTCHR        JMP INT.CHR    ;INPUT CHARACTER
020C- 4C A0 8A      1820 TOUT          JMP T.OUT      ;OUTPUT CHARACTER
020F- 4C DB 02      1830 TTYOUT        JMP TTY.OUT    ;HARDCOPY OUTPUT
0212- 4C 72 89      1840 BEEP          JMP B.EEP      ;BEEPER ROUTINE
                    1850 ;
                    1860 ;PRINTER/TERMINAL BAUD RATES
                    1870 ;
0215- 06            1880 PBAUD         .BY $06        ;VALUE FOR 2400 BAUD
0216- 01            1890 KBAUD         .BY $01        ;VALUE FOR 4800 BAUD
0217- 01            1900 KTMFLG        .BY $01        ;1=KTM 0=DUMB TERMINAL
                    1910 ;
                    1920 ;NEW COLD START
                    1930 ;
0218- 20 86 8B      1940 COLD          JSR ACCESS     ;TO SYSTEM RAM
021B- A2 FF         1950               LDX #$FF       ;INITIALIZE STACK
021D- 9A            1960               TXS
021E- E8            1970               INX  ;R(X)=0
021F- 86 EF         1980               STX *DISCO     ;RESET FLAGS
0221- 8E 13 01      1990               STX PASS       ;SET FOR PASS 1
0224- 86 E3         2000               STX *SUP.OUT           ;ENABLE RAE'S ECHO
0226- 8E 1F 01      2010               STX PRINT/CTL          ;HARD COPY CLEAR
0229- E8            2020               INX            ;R(X)=1
022A- 86 EE         2030               STX *DISCI     ;SET FOR .CT PATCH
                    2040 ;
022C- 20 F9 02      2050               JSR KTM.I/O    ;ASSUME 4800 BAUD KTM
022F- 20 6C 02      2060               JSR CAS.INIT   :INITIALIZE 2ND CASSETTE
0232- 20 78 02      2070               JSR SET.VEC    ;SET ALL VECTORS
0235- 20 B1 02      2080               JSR D.PARM     ;SET DEFAULT PARAMETERS
0238- A2 00         2090               LDX #$00       ;REQ'D BY COLD
023A- A9 01         2100               LDA #$01       ;TO SET FLAGS
023C- 8D 20 01      2110               STA LINE/CNT
023F- 8D 21 01      2120               STA PAGE/NUM
0242- 8D 0F 01      2130               STA FORMAT
0245- 85 DB         2140               STA *ERRORS
0247- 8E 0C 01      2150               STX INCBY
024A- 8E 0D 01      2160               STX INCBY+1
024D- 8E 0E 01      2170               STX MANU
0250- A9 0C         2180               LDA #$0C
0252- 20 0C 02      2190               JSR TOUT
0255- 20 12 02      2200               JSR BEEP
0258- A0 00         2210               LDY #$00
025A- 20 5D 06      2220               JSR MESSUB
025D- 4C 4B B0      2230               JMP RAE.COLD
                    2240 ;
                    2250 ;
                    2260 ;NEW WARM START
                    2270 ;
0260- 20 86 8B      2280 WARM          JSR ACCESS     ;TO SYSTEM RAM
0263- 20 6C 02      2290               JSR CAS.INIT   :INITIALIZE AUX CASSETTE
```

```
0266- 20 78 02  2300               JSR SET.VEC    ;RESET ALL VECTORS
0269- 4C AE B0  2310               JMP RAE.WARM   :OUR WARM ENTRY
                2320 ;
                2330 ;SUBROUTINE TO INITIALIZE SECOND CASSETTE
                2340 ;
026C- AD 02 A0  2350 CAS.INIT      LDA VIADDB     ;GET CURRENT STATE
026F- 09 80     2360               ORA #$80       ;SET BIT 7
0271- 8D 02 A0  2370               STA VIADDB     ;FOR OUTPUT
0274- 20 18 E3  2380               JSR TAPE1.OFF  :NOW TURN IT OFF
0277- 60        2390               RTS
                2400 ;
                2410 ;SUBROUTINE TO INITIALIZE ALL VECTORS
                2420 ;
0278- A9 60     2430 SET.VEC       LDA #$60       ;DISABLE USER CALL
027A- 85 03     2440               STA *USERVEC   :BY SETTING AN RTS
027C- A9 4C     2450               LDA #$4C       ;SET JUMP FOR
027E- 85 00     2460               STA *CTROLYVEC :USER CONTROL Y
0280- 85 B6     2470               STA *PRTVEC    ;USER PRINTER VECTOR
                2480 ;
0282- A2 0F     2490               LDX #L,TTYOUT          ;SET PRINT VECTOR
0284- A9 02     2500               LDA #H,TTYOUT
0286- 86 B7     2510               STX *PRTVEC+1
0288- 85 B8     2520               STA *PRTVEC+2
                2530 ;
028A- A2 0E     2540               LDX #L,CTROLY1         ;SET USER CONTROL Y
028C- A9 05     2550               LDA #H,CTROLY1         ;VECTOR
028E- 86 01     2560               STX *CTROLYVEC+1
0290- 85 02     2570               STA *CTROLYVEC+2
                2580 ;
0292- 20 2E 03  2590               JSR RAE.I/O
                2600 ;
0295- A2 2B     2610               LDX #L,@C.CHK
0297- A9 0A     2620               LDA #H,@C.CHK
0299- 8E 76 A6  2630               STX UBRKVC
029C- 8D 77 A6  2640               STA UBRKVC+1
                2650 ;
029F- A2 3C     2660               LDX #L,CON.DSK
02A1- A9 0A     2670               LDA #H,CON.DSK
02A3- 86 F6     2680               STX *DISCI.VEC
02A5- 85 F7     2690               STA *DISCI.VEC+1
                2700 ;
02A7- 20 39 0A  2710               JSR DSK.SET    ;SET DISK VECTORS
                2720 ;
02AA- A9 00     2730               LDA #$00       ;CLEAR FLAGS
02AC- 85 B2     2740               STA *APP       ;NOT SCROLLING
02AE- 85 B4     2750               STA *EXFLG     ;NOT EXECUTE FILE
02B0- 60        2760               RTS
                2770 ;
                2780 ;SUBROUTINE TO SET DEFAULT EXTENTS
                2790 ;
02B1- A2 07     2800 D.PARM        LDX #$07
02B3- BD D1 02  2810 D.PARM1       LDA D.TAB,X    ;MOVE IN NEW
02B6- 9D 00 01  2820               STA TXST,X     ;DEFAULT VALUES
02B9- CA        2830               DEX
02BA- 10 F7     2840               BPL D.PARM1
02BC- AD D9 02  2850               LDA D.TAB+$08
02BF- 85 C8     2860               STA *PUREC
02C1- AD DA 02  2870               LDA D.TAB+$09
```

```
02C4- 85 C9    2880               STA *PUREC+1
02C6- E8       2890               INX           ;R(X)=0
02C7- 20 96 B0 2900               JSR CL.TXT    ;CLEAR TEXT FILE
02CA- 20 37 BF 2910               JSR CL.STAB   ;CLEAR LABEL FILE
02CD- 20 5F E0 2920               JSR MRK.END   ;SET END OF FILE
02D0- 60       2930               RTS
               2940 ;
               2950 ;SET THE FOLLOWING TO YOUR OWN VALUES
               2960 ;
02D1- DD 0B    2970 D.TAB         .SE STEXT     ;START OF TEXT
02D3- FD 3F    2980               .SE ETEXT     ;END OF TEXT
02D5- 00 40    2990               .SE SLABEL    ;START OF SYMBOLS
02D7- FD 5E    3000               .SE ELABEL    ;END OF SYMBOLS
02D9- 00 5F    3010               .SE BUFFER    ;BUFFER
               3020 ;
               3030 ;
               3040 ;OUTPUT TO PRINTER
               3050 ;
02DB- 20 86 8B 3060 TTY.OUT       JSR ACCESS    ;TO SYSTEM RAM
02DE- 48       3070               PHA           ;SAVE CHARACTER
02DF- 20 05 03 3080               JSR PRT.I/O   ;USING CURRENT LOOP
02E2- AD 00 A0 3090 BUSY          LDA VIADRB    ;PRINTER DTR HANGS
02E5- 29 01    3100               AND #$01      ;ON BIT ZERO
02E7- F0 F9    3110               BEQ BUSY      ;WAIT TILL READY
02E9- 68       3120               PLA           ;RECOVER ACCUMULATOR
02EA- 48       3130               PHA           ;AND RESAVE IT ON STACK
02EB- C9 08    3140               CMP #BS       ;IS IT A BACKSPACE
02ED- D0 02    3150               BNE TTY1      ;JUMP IF NOT
02EF- A9 5C    3160               LDA #BKSL     ;REPLACE WITH ®
02F1- 20 0C 02 3170 TTY1          JSR TOUT      ;SENT CHARACTER
02F4- 20 F9 02 3180               JSR KTM.I/O   ;RESET FOR TERMINAL
02F7- 68       3190               PLA           ;RESTORE CHARACTER
02F8- 60       3200               RTS
               3210 ;
               3220 ;SUBROUTINE TO SET KTM I/O
               3230 ;
02F9- A9 90    3240 KTM.I/O       LDA #CRTI+CRTO :GET THE RIGHT BITS
02FB- 8D 54 A6 3250               STA TOUTFL    ;AND LET MON KNOW
02FE- AD 16 02 3260               LDA KBAUD     ;KTM BAUD RATE
0301- 8D 51 A6 3270               STA SDBYT
0304- 60       3280               RTS
               3290 ;
               3300 ;SUBROUTINE TO SET PRINTER I/O
               3310 ;
0305- A9 20    3320 PRT.I/O       LDA #TTYO
0307- 8D 54 A6 3330               STA TOUTFL
030A- AD 15 02 3340               LDA PBAUD
030D- 8D 51 A6 3350               STA SDBYT
0310- 60       3360               RTS
               3370 ;
               3380 ;SUBROUTINE TO SET OUTPUT VECTOR SO THAT
               3390 ;MONITOR WILL ECHO TO PRINTER IF TURNED ON
               3400 ;
0311- 20 86 8B 3410 MON.I/O       JSR ACCESS
0314- A9 00    3420               LDA #$00
0316- 8D 53 A6 3430               STA TECHO
0319- A9 D7    3440               LDA #L,ECHAR.VEC :SET FOR MON
031B- 8D 64 A6 3450               STA OUTVEC+1
```

```
031E- A9 03    3460               LDA #H,ECHAR.VEC
0320- 8D 65 A6 3470               STA OUTVEC+2
0323- A9 4B    3480               LDA #L,MGETCH
0325- 8D 61 A6 3490               STA INVEC+1
0328- A9 03    3500               LDA #H,MGETCH
032A- 8D 62 A6 3510               STA INVEC+2
032D- 60       3520               RTS
               3530 ;
               3540 ;SUBROUTINE TO RESET I/O FOR RAE PATCH
               3550 ;
032E- 20 86 8B 3560 RAE.I/O       JSR ACCESS
0331- A9 00    3570               LDA #$00
0333- 8D 53 A6 3580               STA TECHO
0336- A9 64    3590               LDA #L,CRTIN
0338- 8D 61 A6 3600               STA INVEC+1
033B- A9 03    3610               LDA #H,CRTIN
033D- 8D 62 A6 3620               STA INVEC+2
0340- A9 9F    3630               LDA #L,WRT.HOOK
0342- 8D 64 A6 3640               STA OUTVEC+1
0345- A9 03    3650               LDA #H,WRT.HOOK
0347- 8D 65 A6 3660               STA OUTVEC+2
034A- 60       3670               RTS
               3680 ;
               3690 ;
               3700 ; CHARACTER FETCH FOR MONITOR SO PRINTER PATCH
               3710 ; WORKS PROPERLY
               3720 ;
034B- 20 88 81 3730 MGETCH        JSR SAVER
034E- 20 57 03 3740               JSR GETCH
0351- 20 63 A6 3750               JSR OUTVEC
0354- 4C B8 81 3760               JMP RESXAF
               3770 ;
               3780 ;INPUT FROM KEYBOARD
               3790 ;
0357- A5 B4    3800 GETCH         LDA *EXFLG    ;TEST EXECUTE FILE FLAG
0359- F0 03    3810               BEQ GETCH1    ;NOT FROM FILE
035B- 4C 42 08 3820               JMP EXECT     ;GET CHAR FROM FILE
035E- 20 09 02 3830 GETCH1        JSR INTCHR    ;FROM KEYBOARD
0361- 29 7F    3840               AND #$7F
0363- 60       3850               RTS
               3860 ;
               3870 ;SUBROUTINE TO ECHO CONSOLE INPUT TO PRINTER
               3880 ;WHEN HARD COPY IS SET AND ALSO TO MAKE
               3890 ;KTM DELETE KEY CLEAN A CHARACTER OF THE SCREEN
               3900 ;
0364- 20 57 03 3910 CRTIN         JSR GETCH     ;FETCH A CHARACTER
0367- C9 7F    3920               CMP #DEL      ;IF A DEL CHARACTER
0369- D0 02    3930               BNE CRTIN1
036B- A9 08    3940               LDA #BS       ;THEN REPLACE WITH BS
036D- C9 5F    3950 CRTIN1        CMP #ALT      ;IF A ALT CHARACTER
036F- D0 02    3960               BNE CRTIN2
0371- A9 08    3970               LDA #BS       ;THEN REPLACE WITH BS
0373- AE 1F 01 3980 CRTIN2        LDX PRINT/CTL :HARD COPY ACTIVE?
0376- F0 03    3990               BEQ CRTIN3
0378- 20 0F 02 4000               JSR TTYOUT    ;IF SO SEND TO PRINTER
037B- 60       4010 CRTIN3        RTS
               4020 ;
               4030 ;SUBROUTINE TO OUTPUT CHARACTER TO SCREEN
```

```
                4040 ;A BACKSPACE WILL CLEAN CHARACTER OFF SCREEN
                4050 ;
037C- C9 08     4060 CRTOUT     CMP #BS       ;REPLACE BACKSPACE
037E- F0 09     4070            BEQ DO.BS
0380- C9 07     4080            CMP #BEL
0382- D0 18     4090            BNE SENDCHR
0384- 20 12 02  4100            JSR BEEP
0387- D0 13     4110            BNE SENDCHR
0389- AE 17 02  4120 DO.BS      LDX KTMFLG    ;TEST FOR DUMB
038C- D0 04     4130            BNE NOBSL     ;NO
038E- A9 5C     4140            LDA #BKSL     ;OUTPUT ®
0390- D0 0A     4150            BNE SENDCHR
0392- 20 0C 02  4160 NOBSL      JSR TOUT
0395- A9 20     4170            LDA #SPC
0397- 20 42 83  4180            JSR SPACE
039A- A9 08     4190 BCKSP      LDA #BS
039C- 4C 0C 02  4200 SENDCHR    JMP TOUT
                4210 ;
039F- C9 3E     4220 WRT.HOOK   CMP #'>       ;START OF NEW LINE?
03A1- F0 03     4230            BEQ CHK.STK   ;MAYBE -  CHECK STACK
03A3- 4C 7C 03  4240            JMP CRTOUT
                4250 ;
03A6- BA        4260 CHK.STK    TSX           ;CHECK IF WE CAME
03A7- BD 09 01  4270            LDA TXST+9,X :FROM OLD BUFFER
03AA- C9 B5     4280            CMP #H,CRT.IN+6 :CHECK HI BYTE
03AC- D0 0A     4290            BNE NOT.YET
03AE- BD 08 01  4300            LDA TXST+8,X
03B1- C9 C1     4310            CMP #L,CRT.IN+6 :BETTER CHECK LO TOO
03B3- D0 03     4320            BNE NOT.YET
03B5- 4C BD 03  4330            JMP NEW.CRT
                4340 ;
03B8- A9 3E     4350 NOT.YET    LDA #'>       ;REPLACE THE PROMPTER
03BA- 4C 0C 02  4360            JMP TOUT      ;AND SEND IT ALONG
                4370 ;
03BD- A2 08     4380 NEW.CRT    LDX #$08      ;PULL JUNK OFF STACK
03BF- 68        4390 FIX.STK    PLA
03C0- CA        4400            DEX
03C1- 10 FC     4410            BPL FIX.STK
03C3- AD 15 01  4420 PRMT.OT    LDA AUTO      ;AUTO LINE PROMPTS?
03C6- D0 07     4430            BNE PRMT.OUT
03C8- A5 B2     4440            LDA *APP      ;CHECK IF SCROLLING
03CA- F0 03     4450            BEQ PRMT.OUT          ;HERE IF NOT
03CC- 4C B1 07  4460            JMP PART2     ;OUTPUT NEXT LINE
03CF- A9 3E     4470 PRMT.OUT   LDA #'>       ;RESEND THE PROMPT
03D1- 20 0C 02  4480            JSR TOUT
03D4- 4C E2 03  4490            JMP FILBUF
                4500 ;
                4510 ;THIS ROUTINE REQUIRED SO THAT PATCHING ROUTINE DOES
                4520 ;DOES NOT EAT ITSELF AS WILL HAPPEN IF WE TRY TO USE
                4530 ;RAE'S OR MON'S OUTPUT VECTORS.
                4540 ;
03D7- AE 1F 01  4550 ECHAR.VEC  LDX PRINT/CTL :HARD COPY SET?
03DA- F0 03     4560            BEQ ECHAR.VEC1 :BRANCH IF NOT
03DC- 20 0F 02  4570            JSR TTYOUT
03DF- 4C 0C 02  4580 ECHAR.VEC1 JMP TOUT
                4590 ;
                4600            .CT XRAY2
```

```
                0010 ;
                0020 ;****************************************
                0030 ;**                                    **
                0040 ;**  EXTENDED ENHANCEMENTS (XRAE)      **
                0050 ;**  FOR SYNERTEK ASM/TED 1.0          **
                0060 ;**                                    **
                0070 ;**      COPYRIGHT (C) 1982            **
                0080 ;**     SATURN SOFTWARE LIMITED        **
                0090 ;**                                    **
                0100 ;**       DISK FILE XRAY2              **
                0110 ;**       FEBRUARY 17, 1982            **
                0120 ;**                                    **
                0130 ;****************************************
                0140 ;
                0150 ;THIS IS THE ACTUAL START OF THE NEW FILL BUFFER ROUTIN
                0160 ;IT IS ESSENTIALLY THE SAME AS THE ONE IN EXTENTED SYM
                0170 ;BASIC AND THE ONE IN SYM-FORTH.
                0180 ;
                0190
03E2- 20 52 06  0200 FILBUF     JSR ZERBUF    ;CLEAR BUFFER
                0210
03E5- A0 00     0220 STBUFF     LDY #$00      ;POINT TO START
03E7- 84 9F     0230            STY *YMAX
03E9- 84 B1     0240            STY *LEN
03EB- B8        0250 NEXCHR     CLV  ;BRANCH ALWAYS FLAG
                0260
03EC- 20 57 03  0270            JSR GETCH
03EF- C9 20     0280            CMP #SPC      ;IF CONTROL CHAR
03F1- 90 03     0290            BCC CTROLA    ;THEN PROCESS IT
03F3- 4C 62 05  0300            JMP CHAR.     ;ELSE PROCESS CHARACTER
                0310
03F6- C9 01     0320 CTROLA     CMP #$01      ;SKIP TO START
03F8- D0 23     0330            BNE CTROLC    ;TRY FOR ᵕC
03FA- C4 B1     0340            CPY *LEN      ;IF ALREADY AT START
03FC- D0 03     0350            BNE BEEP11
03FE- 4C A0 04  0360            JMP BEEP1
0401- 20 40 06  0370 BEEP11     JSR CRLFP
0404- 20 09 04  0380            JSR KTA
0407- 50 E2     0390            BVC NEXCHR
0409- AD 17 02  0400 KTA        LDA KTMFLG
040C- F0 09     0410            BEQ DUMA
040E- 88        0420 CTROLA1    DEY           ;BACK UP TO START
040F- 20 3B 06  0430            JSR BCKSPC
0412- C4 B1     0440            CPY *LEN
0414- D0 F8     0450            BNE CTROLA1
0416- 60        0460            RTS
0417- 20 CA E3  0470 DUMA       JSR CRLF
041A- A4 B1     0480            LDY *LEN
041C- 60        0490            RTS
                0500 ;
041D- C9 03     0510 CTROLC     CMP #$03      ;TURN OFF AUTO?
041F- D0 0E     0520            BNE CTROLH    ;TRY FOR BS
0421- A2 00     0530            LDX #$00      ;GET A ZERO
0423- 8E 0C 01  0540            STX INCBY
0426- 8E 0D 01  0550            STX INCBY+1
0429- 8E 15 01  0560            STX AUTO
```

```
042C- 4C 06 02  0570          JMP NEW.HOT
                0580 ;
042F- C9 08     0590 CTROLH   CMP #BS      ;BACK SPACE?
0431- D0 0D     0600          BNE CTROLI   ;NO
0433- AE 17 02  0610          LDX KTMFLG   ;IGNORE IF DUMB SET
0436- F0 B3     0620          BEQ NEXCHR
0438- C4 B1     0630          CPY *LEN     ;YES- AT START ALREADY?
043A- D0 03     0640          BNE CTROLH1  ;DO A BACK SPACE
043C- 4C 24 05  0650          JMP CTROLZ1
043F- 88        0660 CTROLH1  DEY          ;FALL THROUGH WITH BS
                0670
0440- C9 09     0680 CTROLI   CMP #$09     ;HORZ TAB?
0442- D0 0A     0690          BNE CTROLJ   ;TRY FOR LINE FEED
0444- C0 50     0700          CPY #$50     ;YES- AT END ALREADY?
0446- B0 B2     0710          BCS CTROLA+4          ;YES
0448- B9 35 01  0720          LDA CRT,Y    ;NO- DUMP CHAR THEN
044B- F0 AD     0730          BEQ CTROLA+4          ;BUT NOT IF ITS A NULL
044D- C8        0740          INY
                0750 ;
044E- C9 0A     0760 CTROLJ   CMP #$0A     ;AUTO WITH STEP 1?
0450- D0 0F     0770          BNE CTRLU
0452- A2 00     0780          LDX #$00
0454- 8E 0D 01  0790          STX INCBY+1
0457- E8        0800          INX
0458- 8E 0C 01  0810          STX INCBY
045B- 8E 15 01  0820          STX AUTO
045E- 4C 79 04  0830          JMP CTRLM
                0840 ;
0461- C9 15     0850 CTRLU    CMP #$15     ;CONTROL-U ?
0463- D0 06     0860          BNE CTRLD    ;NO
0465- A9 01     0870          LDA #$01     ;SET DIRECTION TO UP
0467- 85 B3     0880          STA *DIREC
0469- D0 0E     0890          BNE CTRLM    ;DO CRLF
                0900 ;
046B- C9 04     0910 CTRLD    CMP #$04     ;CONTROL-D ?
046D- D0 06     0920          BNE CTROLM   ;NO
046F- A9 00     0930          LDA #$00     ;SET DIRECTION TO DOWN
0471- 85 B3     0940          STA *DIREC
0473- F0 04     0950          BEQ CTRLM
                0960         ;
0475- C9 0D     0970 CTROLM   CMP #$0D     ;RETURN?
0477- D0 2D     0980          BNE CTROLP   ;NO
0479- C4 B1     0990 CTRLM    CPY *LEN
047B- D0 0A     1000          BNE CTRM     ;NO
047D- A5 B2     1010          LDA *APP     ;SCROLLING ?
047F- F0 06     1020          BEQ CTRM     ;NO
0481- A9 A0     1030          LDA #$A0     ;INSERT BLANK
0483- 99 35 01  1040          STA CRT,Y
0486- C8        1050          INY
0487- 20 AB 05  1060 CTRM     JSR EEOF
048A- A9 20     1070          LDA #SPC
048C- 99 35 01  1080          STA CRT,Y
048F- C8        1090          INY
0490- 8C 1A 01  1100          STY ADDPAD
0493- 99 35 01  1110 CTLM.L   STA CRT,Y
0496- C8        1120          INY
0497- C0 55     1130          CPY #$55
0499- D0 F8     1140          BNE CTLM.L
```

```
049B- A2 00     1150          LDX #$00     ;SET R(X) TO ZERO
049D- 4C 69 06  1160          JMP CMD.PR   ;PROCESS COMMANDS
                1170 ;
04A0- 20 12 02  1180 BEEP1    JSR BEEP
04A3- 4C EB 03  1190 NEXCHR1  JMP NEXCHR
                1200 ;
04A6- C9 10     1210 CTROLP   CMP #$10     ;TOGGLE PRINTER ON?
04A8- D0 14     1220          BNE CTROLS
04AA- A2 04     1230          LDX #$04     ;START ON LINE 4
04AC- AD 1F 01  1240          LDA PRINT/CTL :GET CURRENT STATE
04AF- F0 05     1250          BEQ CTROLP1
04B1- A2 00     1260          LDX #$00
04B3- 8E 21 01  1270          STX PAGE/NUM :ZERO PAGE COUNT
04B6- 8E 1F 01  1280 CTROLP1  STX PRINT/CTL :SET NEW STATE
04B9- 8E 20 01  1290          STX LINE/CNT :SET LINE COUNT
04BC- 50 E5     1300          BVC NEXCHR1
                1310 ;
04BE- C9 13     1320 CTROLS   CMP #$13     ;MONITOR JUMP?
04C0- D0 0C     1330          BNE CTROLT   ;NO
04C2- 20 11 03  1340          JSR MON.I/O :SET I/O FOR MONITOR
04C5- 20 35 80  1350          JSR USRENT
04C8- 20 2E 03  1360          JSR RAE.I/O
04CB- 4C 06 02  1370          JMP NEW.HOT
                1380
04CE- C9 14     1390 CTROLT   CMP #$14     ;TOGGLE CASSETTE?
04D0- D0 15     1400          BNE CTROLX   ;NO
04D2- 20 57 03  1410          JSR GETCH
04D5- C9 30     1420          CMP #$30     ;CASSETE 0?
04D7- D0 05     1430          BNE CTROLT1  ;NO-
04D9- 20 64 EC  1440          JSR TOG0     ;YES- DO IT TO IT
04DC- 50 C5     1450          BVC NEXCHR1
04DE- C9 31     1460 CTROLT1  CMP #$31     ;CASSETTE 1 THEN?
04E0- D0 BE     1470          BNE BEEP1    ;NO BEEP THE OPERATOR
04E2- 20 59 EC  1480          JSR TOG1     ;YES DO
04E5- 50 BC     1490          BVC NEXCHR1
04E7- C9 18     1500 CTROLX   CMP #$18     ;CANCEL LINE?
04E9- D0 1F     1510          BNE CTROLY   ;NO
04EB- A2 00     1520          LDX #$00
04ED- A5 B2     1530          LDA *APP
04EF- F0 07     1540          BEQ CTROLX0
04F1- 86 B2     1550          STX *APP     ;CLEAR APPEND MODE
04F3- 86 B1     1560          STX *LEN
04F5- 4C 79 04  1570          JMP CTRLM
04F8- 86 B1     1580 CTROLX0  STX *LEN
04FA- C4 B1     1590          CPY *LEN
04FC- F0 03     1600          BEQ CTROLX1
04FE- 20 09 04  1610          JSR KTA
0501- 20 AB 05  1620 CTROLX1  JSR EEOF
0504- 20 40 06  1630          JSR CRLFP    ;ECHO IF PRINTER ON
0507- 4C E2 03  1640          JMP FILBUF
                1650 ;
050A- C9 19     1660 CTROLY   CMP #$19     ;CONTROL Y?
050C- D0 12     1670          BNE CTROLZ   ;NO
050E- 20 11 03  1680 CTROLY1  JSR MON.I/O :SET FOR MONITOR
0511- 20 02 81  1690          JSR GETCOM+3
0514- 20 4A 81  1700          JSR DISPAT
0517- 20 71 81  1710          JSR ERMSG
051A- 20 2E 03  1720          JSR RAE.I/O
```

```
051D- 4C 06 02  1730              JMP NEW.HOT
                1740 ;
0520- C9 1A     1750 CTROLZ       CMP #$1A      ;SKIP TO END?
0522- D0 10     1760              BNE ESCAPE    ;NO
0524- B9 35 01  1770 CTROLZ1      LDA CRT,Y     ;YES
0527- F0 06     1780              BEQ CTROLZ2
0529- 20 D7 03  1790              JSR ECHAR.VEC
052C- C8        1800              INY
052D- 50 F5     1810              BVC CTROLZ1
052F- 20 AB 05  1820 CTROLZ2      JSR EEOF
0532- 50 1A     1830              BVC NEXCHR2
                1840
0534- C9 1B     1850 ESCAPE       CMP #$1B      ;SKIP TO CHARACTER?
0536- D0 19     1860              BNE ECHO      ;NO
0538- 20 57 03  1870              JSR GETCH
053B- 85 B0     1880              STA *SCRATCH
053D- B9 35 01  1890 ESCAPE1      LDA CRT,Y
0540- F0 ED     1900              BEQ CTROLZ2
0542- 20 D7 03  1910              JSR ECHAR.VEC
0545- C8        1920              INY
0546- C5 B0     1930              CMP *SCRATCH
0548- F0 04     1940              BEQ NEXCHR2
054A- C0 50     1950              CPY #$50
054C- 90 EF     1960              BCC ESCAPE1
054E- 4C A3 04  1970 NEXCHR2      JMP NEXCHR1
                1980
0551- C9 08     1990 ECHO         CMP #BS       ;CNTRL-H ?
0553- F0 08     2000              BEQ ECHO1     ;ECHO IT
0555- C9 02     2010              CMP #$02      ;CNTRL-B ?
0557- F0 25     2020              BEQ ACCEPT1   ;BLANK LINE
0559- C9 20     2030              CMP #SPC      ;PRINTABLE ?
055B- 90 F1     2040              BCC NEXCHR2   ;NO-IGNORE
055D- 20 D7 03  2050 ECHO1        JSR ECHAR.VEC
0560- 50 EC     2060              BVC NEXCHR2
                2070
0562- C9 5F     2080 CHAR.        CMP #$5F      ;DELETE?
0564- F0 04     2090              BEQ DELETE
0566- C9 7F     2100              CMP #$7F      ;DELETE?
0568- D0 0F     2110              BNE ACCEPT
                2120
056A- C4 B1     2130 DELETE       CPY *LEN      ;ALREADY AT START?
056C- F0 08     2140              BEQ BEEP2
056E- 20 1F 06  2150              JSR DELCHR
0571- 20 03 06  2160              JSR MOVDWN
0574- 50 D8     2170              BVC NEXCHR2
0576- 4C A0 04  2180 BEEP2        JMP BEEP1
                2190
0579- 20 C7 05  2200 ACCEPT       JSR INSERT
057C- 50 D0     2210              BVC NEXCHR2
                2220 ;
057E- A9 A0     2230 ACCEPT1      LDA #$A0      ;INSERT BLANK
0580- 20 C7 05  2240              JSR INSERT
0583- 4C 79 04  2250              JMP CTRLM     ;DO CRLF
                2260 ;
                2270 ;
0586- AD 17 02  2280 UPDATE       LDA KTMFLG
0589- F0 1B     2290              BEQ UPDATE4
058B- 84 B0     2300              STY *SCRATCH
```

```
058D- B9 35 01  2310 UPDATE1      LDA CRT,Y
0590- D0 02     2320              BNE UPDATE2
0592- A9 20     2330              LDA #SPC
0594- 20 0C 02  2340 UPDATE2      JSR TOUT
0597- C8        2350              INY
0598- C4 9F     2360              CPY *YMAX
059A- 90 F1     2370              BCC UPDATE1
059C- A9 08     2380 UPDATE3      LDA #BS
059E- 20 0C 02  2390              JSR TOUT
05A1- 88        2400              DEY
05A2- C4 B0     2410              CPY *SCRATCH
05A4- D0 F6     2420              BNE UPDATE3
05A6- A9 80     2430 UPDATE4      LDA #$80
05A8- 85 9E     2440              STA *UPDFLG
05AA- 60        2450              RTS
                2460 ;
05AB- AD 17 02  2470 EEOF         LDA KTMFLG
05AE- F0 16     2480              BEQ EEOF3
05B0- 84 B0     2490              STY *SCRATCH
05B2- A9 20     2500 EEOF1        LDA #SPC
05B4- 20 0C 02  2510              JSR TOUT
05B7- C8        2520              INY
05B8- C4 9F     2530              CPY *YMAX     ;DON'T ERASE TOO MUCH!
05BA- 90 F6     2540              BCC EEOF1
05BC- A9 08     2550 EEOF2        LDA #BS
05BE- 20 0C 02  2560              JSR TOUT
05C1- 88        2570              DEY
05C2- C4 B0     2580              CPY *SCRATCH
05C4- D0 F6     2590              BNE EEOF2
                2600 ;
05C6- 60        2610 EEOF3        RTS
                2620
05C7- 48        2630 INSERT       PHA  ;SAVE CHARACTER
05C8- 98        2640              TYA  ;SAVE Y TOO
05C9- 48        2650              PHA
05CA- A2 01     2660              LDX #$01      ;FIND BUFFER END
05CC- B9 35 01  2670 INSERT1      LDA CRT,Y
05CF- F0 04     2680              BEQ INSERT2
05D1- C8        2690              INY
05D2- E8        2700              INX
05D3- 50 F7     2710              BVC INSERT1
                2720
05D5- C4 9F     2730 INSERT2      CPY *YMAX
05D7- 90 04     2740              BCC INSERT20            ;UPDATE YMAX
05D9- C8        2750              INY
05DA- 84 9F     2760              STY *YMAX     ;IF NECESSARY
05DC- 88        2770              DEY
05DD- C0 50     2780 INSERT20     CPY #$50      ;PASSED END?
05DF- 90 08     2790              BCC INSERT3   ;NOPE
05E1- 68        2800              PLA  ;YES- RESTORE AND BEEP
05E2- A8        2810              TAY
05E3- 68        2820              PLA
05E4- 20 12 02  2830              JSR BEEP
05E7- 50 16     2840              BVC INSERT4
                2850
05E9- B9 35 01  2860 INSERT3      LDA CRT,Y
05EC- C8        2870              INY
05ED- 99 35 01  2880              STA CRT,Y
```

```
05F0- 88        2890          DEY
05F1- 88        2900          DEY
05F2- CA        2910          DEX
05F3- D0 F4     2920          BNE INSERT3
05F5- 68        2930          PLA
05F6- A8        2940          TAY
05F7- 68        2950          PLA
05F8- 99 35 01  2960          STA CRT,Y
05FB- 20 D7 03  2970          JSR ECHAR.VEC
05FE- C8        2980          INY
05FF- 20 86 05  2990 INSERT4  JSR UPDATE
0602- 60        3000          RTS
                3010
0603- 98        3020 MOVDWN   TYA
0604- 48        3030          PHA  ;SAVE Y
0605- B9 35 01  3040 MOVDWN1  LDA CRT,Y
0608- F0 08     3050          BEQ MOVDWN2
060A- 88        3060          DEY
060B- 99 35 01  3070          STA CRT,Y
060E- C8        3080          INY
060F- C8        3090          INY
0610- 50 F3     3100          BVC MOVDWN1
0612- 88        3110 MOVDWN2  DEY
0613- 99 35 01  3120          STA CRT,Y
0616- 68        3130          PLA  ;RESTORE Y
0617- A8        3140          TAY
0618- 88        3150          DEY
0619- 20 86 05  3160          JSR UPDATE
061C- C6 9F     3170          DEC *YMAX
061E- 60        3180          RTS
                3190 ;
061F- AE 1F 01  3200 DELCHR   LDX PRINT/CTL        ;ECHO BACK SL
0622- F0 05     3210          BEQ DELCHR1
0624- A9 5C     3220          LDA #BKSL
0626- 20 0F 02  3230          JSR TTYOUT
0629- AE 17 02  3240 DELCHR1  LDX KTMFLG
062C- D0 05     3250          BNE DELCR
062E- A9 5C     3260          LDA #BKSL
0630- 4C 0C 02  3270          JMP TOUT
0633- 20 3B 06  3280 DELCR    JSR BCKSPC
0636- A9 20     3290          LDA #SPC
0638- 20 0C 02  3300          JSR TOUT
063B- A9 08     3310 BCKSPC   LDA #BS
063D- 4C 0C 02  3320          JMP TOUT
                3330 ;
                3340 ;ECHO CRLF TO PRINTER IF PRINT/CTL IS SET
                3350 ;
0640- AE 1F 01  3360 CRLFP    LDX PRINT/CTL :IF PRINTER OFF
0643- F0 0C     3370          BEQ CRLFP1    ;THEN RETURN
0645- 48        3380          PHA           ;SAVE R(A) ON STACK
0646- A9 0D     3390          LDA #$0D      ;ECHO CARRIAGE RETURN
0648- 20 0F 02  3400          JSR TTYOUT
064B- A9 0A     3410          LDA #$0A      ;ECHO LINE FEED
064D- 20 0F 02  3420          JSR TTYOUT
0650- 68        3430          PLA           ;RESTORE R(A)
0651- 60        3440 CRLFP1   RTS
                3450 ;
                3460 ;SUBROUTINE TO CLEAR INPUT BUFFER TO NULLS
```

```
                3470 ;
0652- A0 55     3480 ZERBUF     LDY #$55
0654- A9 00     3490            LDA #$00
0656- 99 35 01  3500 ZERBUF1    STA CRT,Y
0659- 88        3510            DEY
065A- 10 FA     3520            BPL ZERBUF1
065C- 60        3530            RTS
                3540 ;
065D- B9 DB 09  3550 MESSUB     LDA TEXT,Y
0660- F0 06     3560            BEQ MESSUB1
0662- 20 D7 03  3570            JSR ECHAR.VEC
0665- C8        3580            INY
0666- D0 F5     3590            BNE MESSUB
0668- 60        3600 MESSUB1    RTS
                3610 ;
                3620            .CT XRAY3


                0010 ;
                0020 ;****************************************
                0030 ;**                                    **
                0040 ;**  EXTENDED ENHANCEMENTS (XRAE)      **
                0050 ;**  FOR SYNERTEK ASM/TED 1.0          **
                0060 ;**                                    **
                0070 ;**      COPYRIGHT (C) 1982            **
                0080 ;**     SATURN SOFTWARE LIMITED        **
                0090 ;**                                    **
                0100 ;**        DISK FILE XRAY3             **
                0110 ;**       FEBRUARY 17,1982             **
                0120 ;**                                    **
                0130 ;****************************************
                0140 ;
                0150 ; THIS IS THE START OF THE NEW COMMAND PROCESSOR
                0160 ;
0669- 20 CA E3  0170 CMD.PR     JSR CRLF       ;DO CRLF
066C- AD 15 01  0180            LDA AUTO       ;CHECK AUTO FLAG
066F- D0 37     0190            BNE NOCM       ;INSERT IF SET
0671- A5 B2     0200            LDA *APP
0673- F0 07     0210            BEQ NXC0
0675- A5 9E     0220            LDA *UPDFLG
0677- D0 2F     0230            BNE NOCM       ;INSERT IF SET
0679- 4C C3 03  0240            JMP PRMT.OT
067C- A2 00     0250 NXC0       LDX #$00
067E- A0 00     0260 NXC        LDY #$00       ;START OF LINE
0680- 20 02 B5  0270            JSR PAS.SPCS          ;SKIP SPACES
0683- C0 50     0280            CPY #$50       ;LINE EMPTY ?
0685- B0 21     0290            BCS NOCM       ;YES
0687- BD BD 06  0300            LDA CMD,X      ;TEST AGAINST
068A- F0 1C     0310            BEQ NOCM       ;COMMAND TABLE
068C- D9 35 01  0320            CMP CRT,Y
068F- F0 07     0330            BEQ NXTLT
0691- 09 20     0340            ORA #$20
0693- D9 35 01  0350            CMP CRT,Y
0696- D0 0A     0360            BNE NXT.D
0698- B9 36 01  0370 NXTLT      LDA CRT+1,Y
069B- 29 DF     0380            AND #$DF
069D- DD BE 06  0390            CMP CMD+1,X
```

```
06A0- F0 09      0400          BEQ FOUD
06A2- E8         0410 NXT.D    INX   ;NEXT COMMAND
06A3- E8         0420          INX
06A4- E8         0430          INX
06A5- E8         0440          INX
06A6- D0 D6      0450          BNE NXC
06A8- A2 00      0460 NOCM     LDX #$00     ;CAN'T FIND
06AA- 60         0470          RTS   ;LET RAE PROCESS
06AB- BD BF 06   0480 FOUD     LDA CMD+2,X  ;GET ROUTINE ADDRESS
06AE- 8D DD 00   0490          STA SCRATO
06B1- BD C0 06   0500          LDA CMD+3,X
06B4- 8D DE 00   0510          STA SCRATO+1
06B7- 20 FF B4   0520          JSR NXT.FLD  ;SKIP TO NEXT FIELD
06BA- 6C DD 00   0530          JMP (SCRATO)        ;EXECUTE COMMAND
                 0540 ;
                 0550 ;COMMAND TABLE
                 0560 ;
06BD- 45 44      0570 CMD      .BY 'ED'     ;EDIT
06BF- F3 06      0580          .SI EDIT
06C1- 41 44      0590          .BY 'AD'     ;ADDRESS
06C3- 75 07      0600          .SI ADR
06C5- 41 50      0610          .BY 'AP'     ;APPEND
06C7- 96 07      0620          .SI APPEND
06C9- 45 58      0630          .BY 'EX'     ;EXECUTE
06CB- 2E 08      0640          .SI EXEC
06CD- 46 4C      0650          .BY 'FL'     ;FILE
06CF- 9B 08      0660          .SI FILE
06D1- 54 41      0670          .BY 'TA'     ;TAPE SPEED
06D3- D0 08      0680          .SI TAPE
06D5- 53 4F      0690          .BY 'SO'     ;HYPERSORT
06D7- FF 08      0700          .SI SORT
06D9- 53 41      0710          .BY 'SA'     ;SAVE
06DB- A3 09      0720          .SI SAVE
06DD- 52 45      0730          .BY 'RE'     ;RESTORE FILE
06DF- 85 08      0740          .SI RESTORE
                 0750 ;
                 0760 ; LEAVE ROOM FOR THREE MORE COMMANDS
                 0770 ;
06E1- 00 00 00   0780          .BY $00 $00 $00 $00
06E4- 00
06E5- 00 00 00   0790          .BY $00 $00 $00 $00
06E8- 00
06E9- 00 00 00   0800          .BY $00 $00 $00 $00
06EC- 00
06ED- 00         0810          .BY $00      ;END OF TABLE
                 0820 ;
06EE- A9 ED      0830 EDERR    LDA #$ED     ;ERROR IN COMMAND
06F0- 4C 4E B4   0840          JMP ERROO
                 0850 ;
06F3- B9 35 01   0860 EDIT     LDA CRT,Y    ;TEST FOR LINE #
06F6- C9 30      0870          CMP #$30
06F8- 90 AE      0880          BCC NOCM     ;NO
06FA- C9 3A      0890          CMP #$3A
06FC- B0 AA      0900          BCS NOCM     ;NO-LET RAE PROCESS
06FE- A2 08      0910 EDIT1    LDX #$08
0700- 20 E6 B2   0920          JSR ASC.SP>DEC      ;GET LINE #
0703- 20 14 B2   0930          JSR SET.F2   ;GET LINE ADDRESS
0706- F0 51      0940          BEQ TOLAR    ;NOT IN FILE
```

```
0708- B0 4F      0950          BCS TOLAR    ;NO IN FILE
070A- 20 52 06   0960          JSR ZERBUF   ;CLEAR BUFFER
070D- A2 00      0970          LDX #$00
070F- AD 09 01   0980          LDA FIRST+1  ;PUT LINE # INTO
0712- 20 60 07   0990          JSR NUM.BUF  ;BUFFER
0715- AD 08 01   1000          LDA FIRST
0718- 20 60 07   1010          JSR NUM.BUF
071B- A0 02      1020          LDY #$02
071D- B1 DD      1030 NEW.CR   LDA (SCRATO),Y        ;MOVE LINE
071F- 30 07      1040          BMI ENDLN    ;TO BUFFER
0721- 9D 35 01   1050          STA CRT,X
0724- E8         1060          INX
0725- C8         1070          INY
0726- D0 F5      1080          BNE NEW.CR
0728- 29 7F      1090 ENDLN    AND #$7F
072A- 9D 35 01   1100          STA CRT,X
072D- A9 04      1110          LDA #$04     ;PROTECT LINE #
072F- 85 B1      1120          STA *LEN
0731- C8         1130          INY
0732- C8         1131          INY
0733- C8         1132          INY
0734- C8         1133          INY
0735- C8         1140          INY
0736- 84 9F      1150          STY *YMAX
0738- A0 00      1160          LDY #$00
073A- 84 9E      1170          STY *UPDFLG
073C- 8C 15 01   1180          STY AUTO     ;CLEAR AUTO MODE
073F- 8C 0C 01   1190          STY INCBY
0742- 8C 0D 01   1200          STY INCBY+1
0745- B9 35 01   1210 DO.NUM   LDA CRT,Y
0748- 20 D7 03   1220          JSR ECHAR.VEC
074B- C8         1230          INY
074C- C4 B1      1240          CPY *LEN
074E- 90 F5      1250          BCC DO.NUM
0750- A9 5D      1260          LDA #']
0752- 20 D7 03   1270          JSR ECHAR.VEC
0755- B8         1280          CLV
0756- 4C 24 05   1290          JMP CTROLZ1  ;START TO EDIT
                 1300 ;
0759- A9 00      1310 TOLAR    LDA #$00     ;CLEAR SCROLLING
075B- 85 B2      1320          STA *APP
075D- 4C C3 03   1330          JMP PRMT.OT  ;IGNORE COMMAND
                 1340 ;
0760- 48         1350 NUM.BUF  PHA   ;INSERT LINE #
0761- 4A         1360          LSR A        ;INTO BUFFER
0762- 4A         1370          LSR A
0763- 4A         1380          LSR A
0764- 4A         1390          LSR A
0765- 09 30      1400          ORA #$30
0767- 9D 35 01   1410          STA CRT,X
076A- E8         1420          INX
076B- 68         1430          PLA
076C- 29 0F      1440          AND #$0F
076E- 09 30      1450          ORA #$30
0770- 9D 35 01   1460          STA CRT,X
0773- E8         1470          INX
0774- 60         1480          RTS
                 1490 ;
```

```
0775- A2 08     1500 ADR        LDX #$08
0777- 20 E6 B2  1510            JSR ASC.SP>DEC       ;GET LINE #
077A- A9 FF     1520            LDA #$FF
077C- 8D 0B 01  1530            STA LAST+1
077F- 20 14 B2  1540            JSR SET.F2   ;GET LINE ADDRESS
0782- F0 0C     1550            BEQ NOFUD    ;NOT THERE
0784- B0 0A     1560            BCS NOFUD    ;NOT THERE
0786- A5 DE     1570            LDA *SCRATO+1        ;PRINT ADDRESS
0788- 20 E2 E3  1580            JSR OUT.BYT
078B- A5 DD     1590            LDA *SCRATO
078D- 20 E2 E3  1600            JSR OUT.BYT
0790- 20 CA E3  1610 NOFUD      JSR CRLF
0793- 4C C3 03  1620            JMP PRMT.OT
                1630 ;
0796- A9 01     1640 APPEND     LDA #$01     ;SET APPEND FLAG
0798- 85 B2     1650            STA *APP
079A- 85 B3     1660            STA *DIREC   ;DIRECTION IS UP
079C- B9 35 01  1670            LDA CRT,Y    ;TEST FOR LINE #
079F- C9 30     1680            CMP #$30
07A1- 90 07     1690            BCC APERR    ;NO
07A3- C9 3A     1700            CMP #$3A
07A5- B0 03     1710            BCS APERR    ;NO
07A7- 4C FE 06  1720            JMP EDIT1    ;EDIT LINE
07AA- A9 00     1730 APERR      LDA #$00
07AC- 85 B2     1731            STA *APP
07AE- 4C EE 06  1732            JMP EDERR
                1740 ;
07B1- 20 52 06  1750 PART2      JSR ZERBUF   ;CLEAR BUFFER
07B4- AD 15 01  1760            LDA AUTO     ;TEST FOR AUTO MODE
07B7- F0 03     1770            BEQ PART3    ;NO
07B9- 4C E2 03  1780            JMP FILBUF   ;CONTINUE IN AUTO MODE
07BC- A5 B3     1790 PART3      LDA *DIREC
07BE- F0 37     1800            BEQ DOWN     ;DOWN
07C0- A0 00     1810            LDY #$00
07C2- A5 9E     1820            LDA *UPDFLG
07C4- F0 06     1830            BEQ ANO1
07C6- B1 DD     1840 ANO        LDA (SCRATO),Y
07C8- C9 A0     1850            CMP #$A0     ;TEST END OF LINE
07CA- B0 08     1860            BCS NEXT     ;YES
07CC- E6 DD     1870 ANO1       INC *SCRATO  ;NEXT CHAR
07CE- D0 F6     1880            BNE ANO
07D0- E6 DE     1890            INC *SCRATO+1
07D2- D0 F2     1900            BNE ANO
07D4- A0 02     1910 NEXT       LDY #$02     ;LINE # INTO
07D6- A9 00     1920            LDA #$00
07D8- 85 9E     1930            STA *UPDFLG
07DA- A2 00     1940 NEXT1      LDX #$00     ;BUFFER
07DC- B1 DD     1950            LDA (SCRATO),Y
07DE- 20 60 07  1960            JSR NUM.BUF  ;HIGH BYTE
07E1- 88        1970            DEY
07E2- B1 DD     1980            LDA (SCRATO),Y
07E4- 20 60 07  1990            JSR NUM.BUF  ;LOW BYTE
07E7- C8        2000            INY  ;START OF TEXT
07E8- C8        2010            INY
07E9- B1 DD     2020            LDA (SCRATO),Y       ;TEST END OF FILE
07EB- F0 03     2030            BEQ END.FIL  ;YES
07ED- 4C 1D 07  2040            JMP NEW.CR   ;EDIT LINE
07F0- A9 00     2050 END.FIL    LDA #$00     ;STOP SCROLLING
```

```
07F2- 85 B2     2060            STA *APP
07F4- 4C E2 03  2070            JMP FILBUF   ;INPUT FROM KEYBOARD
07F7- 20 21 08  2080 DOWN       JSR FIL.BOT  ;TEST BOTTOM OF FILE
07FA- F0 F4     2090            BEQ END.FIL  ;YES
07FC- A9 02     2100            LDA #$02
07FE- 85 B1     2110            STA *LEN
0800- 38        2120 SUBL       SEC
0801- A5 DD     2130            LDA *SCRATO
0803- E5 B1     2140            SBC *LEN     ;NEXT CHAR DOWN
0805- 85 DD     2150            STA *SCRATO
0807- B0 02     2160            BCS TSTL
0809- C6 DE     2170            DEC *SCRATO+1
080B- A9 01     2180 TSTL       LDA #$01
080D- 85 B1     2190            STA *LEN
080F- A0 01     2200            LDY #$01
0811- 20 21 08  2210            JSR FIL.BOT  ;START OF FILE ?
0814- F0 C4     2220            BEQ NEXT1    ;YES
0816- A0 00     2230            LDY #$00
0818- B1 DD     2240            LDA (SCRATO),Y       ;START OF LINE ?
081A- C9 A0     2250            CMP #$A0
081C- 90 E2     2260            BCC SUBL     ;NO
081E- 4C D4 07  2270            JMP NEXT     ;EDIT LINE
                2280 ;
0821- A5 DE     2290 FIL.BOT    LDA *SCRATO+1        ;TEST FOR START
0823- CD 01 01  2300            CMP TXST+1   ;OF FILE
0826- D0 05     2310            BNE LARG
0828- A5 DD     2320            LDA *SCRATO
082A- CD 00 01  2330            CMP TXST
082D- 60        2340 LARG       RTS
                2350 ;
082E- 20 76 08  2360 EXEC       JSR GET.N    ;GET ADDRESS
0831- A9 01     2370            LDA #$01     ;SET EXECUTE FLAG
0833- 85 B4     2380            STA *EXFLG
0835- 85 B5     2390            STA *NUMFLG
0837- A5 D1     2400            LDA *PROC>ADDRS      ;SET START OF FILE
0839- 85 FA     2410            STA *EXPTR
083B- A5 D2     2420            LDA *PROC>ADDRS+1
083D- 85 FB     2430            STA *EXPTR+1
083F- 4C C3 03  2440            JMP PRMT.OT  ;FETCH CHARACTERS
                2450 ;
0842- 84 C7     2460 EXECT      STY *SAVEYY
0844- A5 B5     2470            LDA *NUMFLG  ;LINE # TO SKIP ?
0846- F0 12     2480            BEQ NOSKP    ;NO
0848- 18        2490            CLC
0849- A5 FA     2500            LDA *EXPTR   ;SKIP LINE #
084B- 69 02     2510            ADC #$02
084D- 85 FA     2520            STA *EXPTR
084F- 90 02     2530            BCC GETCR
0851- E6 FB     2540            INC *EXPTR+1
0853- A9 00     2550 GETCR      LDA #$00     ;CLEAR FLAG
0855- 85 B5     2560            STA *NUMFLG
0857- A9 0D     2570            LDA #$0D     ;END OF LINE
0859- 60        2580            RTS
085A- A0 00     2590 NOSKP      LDY #$00
085C- B1 FA     2600            LDA (EXPTR),Y        ;GET NEXT CHAR
085E- F0 0F     2610            BEQ FINEX    ;END OF FILE
0860- 10 02     2620            BPL NOED     ;NOT END OF LINE
0862- E6 B5     2630            INC *NUMFLG  ;SET FLAG
```

```
0864- 29 7F     2640 NOED       AND #$7F
0866- E6 FA     2650            INC *EXPTR      ;NEXT CHAR POINTER
0868- D0 02     2660            BNE RTSE
086A- E6 FB     2670            INC *EXPTR+1
086C- A4 C7     2680 RTSE       LDY *SAVEYY
086E- 60        2690            RTS
086F- A9 00     2700 FINEX      LDA #$00        ;CLEAR EXECUTE FLAG
0871- 85 B4     2710            STA *EXFLG
0873- 4C 06 02  2720            JMP NEW.HOT     ;RESTART RAE
                2730 ;
0876- A2 00     2740 GET.N      LDX #$00        ;HEX MODE
0878- C0 50     2750            CPY #$50        ;END OF LINE ?
087A- B0 06     2760            BCS N.PAR       ;YES
087C- 8E 11 01  2770            STX HEX/DEC
087F- 4C 4A E2  2780            JMP PR.LAB.S            ;GET NUMBER
0882- 4C EE 06  2790 N.PAR      JMP EDERR
                2800 ;
0885- AD 00 01  2810 RESTORE    LDA TXST
0888- 85 D1     2820            STA *PROC>ADDRS
088A- AD 01 01  2830            LDA TXST+1
088D- 85 D2     2840            STA *PROC>ADDRS+1
088F- A0 02     2850            LDY #$02
0891- B1 D1     2860            LDA (PROC>ADDRS),Y
0893- D0 1F     2870            BNE NXTL1
0895- A9 A0     2880            LDA #$A0
0897- 91 D1     2890            STA (PROC>ADDRS),Y
0899- D0 19     2900            BNE NXTL1
                2910 ;
089B- 20 76 08  2920 FILE       JSR GET.N       ;GET ADDRESS
089E- A5 D1     2930            LDA *PROC>ADDRS         ;SET FILE START
08A0- 8D 00 01  2940            STA TXST
08A3- A5 D2     2950            LDA *PROC>ADDRS+1
08A5- 8D 01 01  2960            STA TXST+1
08A8- 4C B4 08  2970            JMP NXTL1
08AB- 20 BE 08  2980 NXTL       JSR IPADDS
08AE- 20 BE 08  2990            JSR IPADDS
08B1- 20 BE 08  3000 NXTL2      JSR IPADDS
08B4- A0 02     3010 NXTL1      LDY #$02
08B6- B1 D1     3020            LDA (PROC>ADDRS),Y
08B8- F0 0B     3030            BEQ ENDFIL
08BA- 30 EF     3040            BMI NXTL
08BC- 10 F3     3050            BPL NXTL2
                3060 ;
08BE- E6 D1     3070 IPADDS     INC *PROC>ADDRS
08C0- D0 02     3080            BNE IPADDS1
08C2- E6 D2     3090            INC *PROC>ADDRS+1
08C4- 60        3100 IPADDS1    RTS
                3110 ;
08C5- A5 D1     3120 ENDFIL     LDA *PROC>ADDRS         ;SET FILE END
08C7- 85 D3     3130            STA *TPRES
08C9- A5 D2     3140            LDA *PROC>ADDRS+1
08CB- 85 D4     3150            STA *TPRES+1
08CD- 4C AE B0  3160            JMP RAE.WARM            ;OUTPUT FILE INFO
                3170 ;
08D0- C0 50     3180 TAPE       CPY #$50        ;NO PARAMETERS ?
08D2- B0 0E     3190            BCS TERR        ;YES
08D4- 20 86 8B  3200            JSR ACCESS
08D7- B9 35 01  3210            LDA CRT,Y
```

```
08DA- C9 31     3220            CMP #$31        ; 1 ?
08DC- F0 07     3230            BEQ SPD1        ;YES
08DE- C9 32     3240            CMP #$32        ; 2 ?
08E0- F0 15     3250            BEQ SPD2        ;YES
08E2- 4C EE 06  3260 TERR       JMP EDERR
08E5- A9 46     3270 SPD1       LDA #$46        ;SET TO SYM x 1 SPEED
08E7- A2 33     3280            LDX #$33
08E9- A0 5A     3290            LDY #$5A
08EB- 8D 32 A6  3300 STORE      STA HSBDRY
08EE- 8E 35 A6  3310            STX TAPET1
08F1- 8C 3C A6  3320            STY TAPET2
08F4- 4C C3 03  3330            JMP PRMT.OT
08F7- A9 23     3340 SPD2       LDA #$23        ;SET TO SYM x 2 SPEED
08F9- A2 19     3350            LDX #$19
08FB- A0 2D     3360            LDY #$2D
08FD- D0 EC     3370            BNE STORE
                3380 ;
08FF- A2 00     3390 SORT       LDX #$00        ;CLEAR EXCH FLAG
0901- AD 04 01  3400            LDA STST        ;START AT BEGINNING
0904- 85 FE     3410            STA *NXT.PTR
0906- AD 05 01  3420            LDA STST+1
0909- 85 FF     3430            STA *NXT.PTR+1
090B- A0 02     3440            LDY #$02        ;TEST FOR EMPTY
090D- B1 FE     3450            LDA (NXT.PTR),Y
090F- F0 1B     3460            BEQ SORT.EXIT           ;YES
0911- A5 FE     3470 NXT.LABEL  LDA *NXT.PTR            ;MAKE NEXT LABEL
0913- 85 FC     3480            STA *CURNT      ;CURRENT LABEL
0915- A5 FF     3490            LDA *NXT.PTR+1
0917- 85 FD     3500            STA *CURNT+1
0919- A0 01     3510            LDY #$01        ;FIND NEXT LABEL
091B- C8        3520 FIND.NXT   INY
091C- B1 FC     3530            LDA (CURNT),Y
091E- 10 FB     3540            BPL FIND.NXT
0920- 20 96 09  3550            JSR CALC.NXT            ;CHECK FOR END OF FIL
0923- A0 02     3560            LDY #$02
0925- B1 FE     3570            LDA (NXT.PTR),Y
0927- D0 07     3580            BNE COMPARE1
0929- 8A        3590            TXA  ;TEST FOR NEXT PASS
092A- D0 D3     3600            BNE SORT        ;YES
092C- 4C C3 03  3610 SORT.EXIT  JMP PRMT.OT
092F- C8        3620 COMPARE    INY  ;COMPARE CURRENT LABEL
0930- B1 FC     3630 COMPARE1   LDA (CURNT),Y           ;WITH NEXT LABEL
0932- 51 FE     3640            EOR (NXT.PTR),Y         ;AND EXCHANGE IF
0934- 30 0A     3650            BMI END.LABEL           ;NECESSARY
0936- B1 FE     3660            LDA (NXT.PTR),Y
0938- D1 FC     3670            CMP (CURNT),Y
093A- 90 1A     3680            BCC EXCHANGE
093C- D0 D3     3690            BNE NXT.LABEL
093E- F0 EF     3700            BEQ COMPARE
0940- B1 FE     3710 END.LABEL  LDA (NXT.PTR),Y         ;END OF LABEL
0942- 10 0A     3720            BPL END.CURNT
0944- 29 7F     3730            AND #$7F        ;END OF NEXT LABEL
0946- D1 FC     3740            CMP (CURNT),Y
0948- F0 0C     3750            BEQ EXCHANGE
094A- 90 0A     3760            BCC EXCHANGE
094C- B0 C3     3770            BCS NXT.LABEL
094E- 09 80     3780 END.CURNT  ORA #$80        ;END OF CURRENT LABEL
0950- D1 FC     3790            CMP (CURNT),Y
```

```
0952- F0 BD     3800          BEQ NXT.LABEL
0954- B0 BB     3810          BCS NXT.LABEL
0956- A0 00     3820 EXCHANGE LDY #$00      ;EXCHANGE LABELS
0958- B1 FC     3830          LDA (CURNT),Y
095A- 91 C8     3840          STA (PUREC),Y
095C- C8        3850          INY
095D- B1 FC     3860          LDA (CURNT),Y
095F- 91 C8     3870          STA (PUREC),Y
0961- C8        3880 CL1      INY
0962- B1 FC     3890          LDA (CURNT),Y
0964- 91 C8     3900          STA (PUREC),Y
0966- 10 F9     3910          BPL CL1
0968- A0 00     3920          LDY #$00
096A- B1 FE     3930          LDA (NXT.PTR),Y
096C- 91 FC     3940          STA (CURNT),Y
096E- C8        3950          INY
096F- B1 FE     3960          LDA (NXT.PTR),Y
0971- 91 FC     3970          STA (CURNT),Y
0973- C8        3980 CL2      INY
0974- B1 FE     3990          LDA (NXT.PTR),Y
0976- 91 FC     4000          STA (CURNT),Y
0978- 10 F9     4010          BPL CL2
097A- 20 96 09  4020          JSR CALC.NXT
097D- A0 00     4030          LDY #$00
097F- B1 C8     4040          LDA (PUREC),Y
0981- 91 FE     4050          STA (NXT.PTR),Y
0983- C8        4060          INY
0984- B1 C8     4070          LDA (PUREC),Y
0986- 91 FE     4080          STA (NXT.PTR),Y
0988- C8        4090 CL3      INY
0989- B1 C8     4100          LDA (PUREC),Y
098B- 91 FE     4110          STA (NXT.PTR),Y
098D- 10 F9     4120          BPL CL3
098F- E8        4130          INX  ;SET EXCH FLAG
0990- D0 01     4140          BNE NXT.LAB
0992- E8        4150          INX
0993- 4C 11 09  4160 NXT.LAB  JMP NXT.LABEL
0996- 98        4170 CALC.NXT TYA  ;CALCULATE VALUE OF
0997- 38        4180          SEC  ;NEXT POINTER FROM VALUE
0998- 65 FC     4190          ADC *CURNT    ;OF CURRENT POINTER
099A- 85 FE     4200          STA *NXT.PTR          ;AND Y
099C- A5 FD     4210          LDA *CURNT+1
099E- 69 00     4220          ADC #$00
09A0- 85 FF     4230          STA *NXT.PTR+1
09A2- 60        4240          RTS
                4250 ;
09A3- 20 86 8B  4260 SAVE     JSR ACCESS
09A6- 20 76 08  4270          JSR GET.N     ;GET ID
09A9- A5 D1     4280          LDA *PROC>ADDRS
09AB- 8D 4E A6  4290          STA ID
09AE- 20 02 B5  4300          JSR PAS.SPCS
09B1- 20 76 08  4310          JSR GET.N     ;GET ADDR1
09B4- A5 D1     4320          LDA *PROC>ADDRS
09B6- 8D 4C A6  4330          STA SAL
09B9- A5 D2     4340          LDA *PROC>ADDRS+1
09BB- 8D 4D A6  4350          STA SAH
09BE- 20 02 B5  4360          JSR PAS.SPCS
09C1- 20 76 08  4370          JSR GET.N     ;GET ADDR2
```

```
09C4- A5 D1     4380          LDA *PROC>ADDRS
09C6- 8D 4A A6  4390          STA EAL
09C9- A5 D2     4400          LDA *PROC>ADDRS+1
09CB- 8D 4B A6  4410          STA EAH
09CE- A9 04     4420          LDA #$04
09D0- 8D 30 A6  4430          STA TAPDEL    ;SET DELAY
09D3- A0 80     4440          LDY #$80
09D5- 20 87 8E  4450          JSR DUMPT     ;WRITE TO TAPE
09D8- 4C C3 03  4460          JMP PRMT.OT
                4470 ;
09DB- 58 2D 52  4480 TEXT     .BY 'X-RAY 2.0  BY  J. BROWN & R. DEANE'
09DE- 41 59 20
09E1- 32 2E 30
09E4- 20 20 42
09E7- 59 20 20
09EA- 4A 2E 20
09ED- 42 52 4F
09F0- 57 4E 20
09F3- 26 20 52
09F6- 2E 20 44
09F9- 45 41 4E
09FC- 45
09FD- 0D 0A 0A  4490          .BY $0D $0A $0A $0A
0A00- 0A
0A01- 43 4F 50  4500          .BY 'COPYRIGHT 1982 SATURN SOFTWARE LIMI
0A04- 59 52 49
0A07- 47 48 54
0A0A- 20 31 39
0A0D- 38 32 20
0A10- 53 41 54
0A13- 55 52 4E
0A16- 20 53 4F
0A19- 46 54 57
0A1C- 41 52 45
0A1F- 20 4C 49
0A22- 4D 49 54
0A25- 45 44
0A27- 0D 0A 0A  4510          .BY $0D $0A $0A $00
0A2A- 00
                4520 ;
                4530          .CT XRAY4


                0010 ;
                0020 ;***************************************
                0030 ;**                                   **
                0040 ;**  EXTENDED ENHANCEMENTS (XRAE)     **
                0050 ;**  FOR SYNERTEK ASM/TED 1.0         **
                0060 ;**                                   **
                0070 ;**      COPYRIGHT (C) 1982           **
                0080 ;**     SATURN SOFTWARE LIMITED       **
                0090 ;**                                   **
                0100 ;**       DISK FILE XRAY4             **
                0110 ;**       FEBRUARY 17, 1982           **
                0120 ;**                                   **
                0130 ;***************************************
                0140 ;
```

```
                 0150 ; ©C.CHK is entered whenever a BRK instruction occurs.
                 0160 ; If the break was due to a CTRL-C in RAE we exit to
                 0170 ; SUPERMON, otherwise it must have come from FODS, so
                 0180 ; we stay in RAE.
                 0190
0A2B- BA         0200 ©C.CHK       TSX
0A2C- BD 03 01   0210              LDA $103,X     ;GET PC,HIGH
0A2F- C9 B0      0220              CMP #$B0       ;BRK FROM RAE?
0A31- D0 03      0230              BNE ©C.CHK1    ;NO, SO STAY IN RAE
0A33- 4C C2 04   0240              JMP CTROLS+4            ;EXIT TO MONITOR
                 0250              IFE FODS
                 0260 ©C.CHK1      JMP CTROLS+$0A
                 0270              ***
                 0280              IFE FODS-1
0A36- 4C 8F 0A   0290 ©C.CHK1      JMP EN4
                 0300              ***
                 0310 ;
                 0320 ; SET DISK VECTORS, TAPE VERSION VECTORS THROUGH
                 0330 ; HERE BUT DOES NOTHING.
                 0340 ;
                 0350              IFE FODS
                 0360 DSK.SET      RTS
                 0370              NOP
                 0380              NOP
                 0390              ***
                 0400              IFE FODS-1
0A39- 4C 4A 0A   0410 DSK.SET      JMP SET.EM
                 0420              ***
                 0430 ;
                 0440 ; THIS IS THE COMMON .CT PATCH VECTOR
                 0450 ;
                 0460              IFE FODS
                 0470 CON.DSK      JMP TAPE.DD
                 0480              ***
                 0490              IFE FODS-1
0A3C- 4C 19 0B   0500 CON.DSK      JMP CON.DSK1
                 0510              ***
                 0520 ;
0A3F- 20 2A E3   0530 TAPE.DD      JSR TAPE1.ON            ;TURN TAPE BACK ON
0A42- A9 00      0540              LDA #0
0A44- 8D 10 01   0550              STA FILE.NO
0A47- 4C 68 EF   0560              JMP TLOAD
                 0570 ;
                 0580 ; THIS ROUTINE SETS RAE'S ENter LOad and DCommand VECTORS
                 0590 ;
                 0600              IFE FODS-1
                 0610 ;
0A4A- A2 65      0620 SET.EM       LDX #L,ENTER
0A4C- A9 0A      0630              LDA #H,ENTER
0A4E- 86 F0      0640              STX *DISC1
0A50- 85 F1      0650              STA *DISC1+1
                 0660 ;
0A52- A2 98      0670              LDX #L,LOAD
0A54- A9 0A      0680              LDA #H,LOAD
```

```
0A56- 86 F2      0690              STX *DISC2
0A58- 85 F3      0700              STA *DISC2+1
                 0710 ;
0A5A- A2 ED      0720              LDX #L,DISK.CMD
0A5C- A9 0A      0730              LDA #H,DISK.CMD
0A5E- 86 EC      0740              STX *DISCC.VEC
0A60- 85 ED      0750              STA *DISCC.VEC+1
                 0760 ;
0A62- 4C 4F 0B   0770              JMP RST.DSK   ; RESET DRIVES
                 0780 ;
                 0790 ;
                 0800 ; START OF DISK INTERFACE
                 0810 ;
                 0820 ; ENTER is the entry point on an ENTER command.  All RA
                 0830 ; disk files are prefaced with an ":".  An error code o
                 0840 ; 30 means no filename was supplied.
                 0850
                 0860
0A65- C0 50      0870 ENTER        CPY #$50      ;IF Y=50 FILENAME IS MISSING
0A67- D0 05      0880              BNE EN1       ;Y POINTS AT FILENAME
0A69- A2 30      0890              LDX #$30      ;ERR 30=NO FILENAME
0A6B- 6C 0E B0   0900              JMP (ERROR)   ;PRINT ERROR MESSAGE
                 0910
0A6E- A2 00      0920 EN1          LDX #0
0A70- 86 EF      0930              STX *DISCO    ;RE-ENABLE TAPE OUTPUT
                 0940
                 0950 ; Build command string: ENT $s.adr$e.adr=:name
                 0960 ; Put it in FODS buffer and let FODS do the work
                 0970
0A72- BD A7 0B   0980 EN2          LDA SAVE.D,X          ;GET NEXT CMD CHARACTER
0A75- F0 06      0990              BEQ EN3       ;ARE WE DONE?
0A77- 9D 80 72   1000              STA TXBUF,X   ;MOVE TO FODS BUFFER
0A7A- E8         1010              INX
0A7B- D0 F5      1020              BNE EN2
0A7D- 20 B3 0B   1030 EN3          JSR DRIVE?    ;CHECK FOR DRIVE#
0A80- 20 8F 0B   1040              JSR RAE.STR   ;TELL FODS WHERE FILE STARTS
0A83- A9 24      1050              LDA #'$
0A85- 9D 80 72   1060              STA TXBUF,X
0A88- E8         1070              INX
0A89- 20 9C 0B   1080              JSR RAE.END   ;TELL FODS WHERE FILE ENDS
0A8C- 20 6D 0B   1090              JSR NAM.CMD   ;GO PUT NAME IN AND DO CMD
                 1100
0A8F- 20 4F 0B   1110 EN4          JSR RST.DSK
0A92- 20 2E 03   1120              JSR RAE.I/O
0A95- 4C 5E B0   1130              JMP RAE.HOT   ;AND GO BACK TO RAE
                 1140
                 1150
                 1160 ; LOAD is the entry point on a LOAD command.  The file
                 1170 ; specified is down-loaded into the current text file.
                 1180 ; If the file exceeds the current text file boundary an
                 1190 ; error 31 results.  The entire file has been brought
                 1200 ; in, but the upper limit should be set to accommodate
                 1210 ; the whole file before preceding.  If the filename is
                 1220 ; preceded with a "+" the file will be appended to the
                 1230 ; current file.  If no filename is specified, an error
                 1240 ; 30 results.
                 1250
0A98- C0 50      1260 LOAD         CPY #$50
```

```
0A9A- D0 05    1270            BNE LO1      ;Y POINTS AT FILENAME
0A9C- A2 30    1280            LDX #$30     ;ERR 30=NO FILENAME
0A9E- 6C 0E B0 1290            JMP (ERROR)  ;PRINT ERROR MESSAGE
               1300
0AA1- 20 BD 0A 1310 LO1        JSR DSK.LOD  ;BRING IN FILE
               1320
0AA4- CD 03 01 1330            CMP TXEN+1   ;TEST FOR
0AA7- 90 E6    1340            BCC EN4      ;TEXT FILE
0AA9- D0 07    1350            BNE LO6      ;OVERFLOW
0AAB- AD 02 01 1360            LDA TXEN     ;HIGH BYTES EQUAL,
0AAE- C5 D3    1370            CMP *TPRES   ;SO MUST TEST
0AB0- B0 DD    1380            BCS EN4      ;LOW BYTES
               1390
0AB2- 20 4F 0B 1400 LO6        JSR RST.DSK
0AB5- 20 2E 03 1410            JSR RAE.I/O
0AB8- A2 31    1420            LDX #$31     ;ERR 31=OVERFLOW OF THE
0ABA- 6C 0E B0 1430            JMP (ERROR)  ;TEXT FILE ON DISK LOAD
               1440
               1450
               1460 ; Build command string: LOD $s.adr=:name
               1470 ; Put it in FODS buffer and let FODS do the work
               1480
0ABD- BD AD 0B 1490 DSK.LOD    LDA LOD,X    ;GET NEXT CMD CHARACTER
0AC0- F0 06    1500            BEQ LO3      ;ARE WE DONE?
0AC2- 9D 80 72 1510            STA TXBUF,X  ;MOVE TO FODS BUFFER
0AC5- E8       1520            INX
0AC6- D0 F5    1530            BNE DSK.LOD
0AC8- 20 B3 0B 1540 LO3        JSR DRIVE?   ;CHECK FOR DRIVE#
0ACB- B9 35 01 1550            LDA CRT,Y    ;GET 1ST CHARACTER IN NAME STRING
0ACE- C9 2B    1560            CMP #'+      ;IS IT TO BE APPENDED?
0AD0- D0 06    1570            BNE LO4      ;IF NOT, SKIP
0AD2- C8       1580            INY  ;BYPASS THE "+"
0AD3- 20 9C 0B 1590            JSR RAE.END  ;TELL FODS WHERE ATTACH POINT IS
0AD6- D0 03    1600            BNE LO5
0AD8- 20 8F 0B 1610 LO4        JSR RAE.STR  ;TELL FODS WHERE TO BEGIN FILE
0ADB- 20 6D 0B 1620 LO5        JSR NAM.CMD  ;GO PUT IN NAME AND DO CMD
               1630
               1640 ; Set EOT pointer and test for overflow
               1650
0ADE- A9 00    1660            LDA #0
0AE0- A0 02    1670            LDY #2
0AE2- 91 A8    1680            STA (EOT),Y  ;FLAG EOT FOR RAE
0AE4- A5 A8    1690            LDA *EOT     ;TELL RAE
0AE6- 85 D3    1700            STA *TPRES   ;WHERE THE
0AE8- A5 A9    1710            LDA *EOT+1   ;END OF THE
0AEA- 85 D4    1720            STA *TPRES+1         ;FILE IS
0AEC- 60       1730            RTS
               1740
               1750
               1760 ; DISK.CMD is the entry point on a DC command.  If no
               1770 ; command string is supplied an error 32 results.
               1780
0AED- C0 50    1790 DISK.CMD   CPY #$50
0AEF- D0 05    1800            BNE DC1      ;Y POINTS AT COMMAND STRING
0AF1- A2 32    1810            LDX #$32     ;ERR 32=NO COMMAND STRING
0AF3- 6C 0E B0 1820            JMP (ERROR)  ;PRINT ERROR MESSAGE
               1830
0AF6- A2 00    1840 DC1        LDX #0
```

```
0AF8- B9 35 01 1850 DC2        LDA CRT,Y    ;GET NEXT CMD CHARACTER
0AFB- 9D 80 72 1860            STA TXBUF,X  ;PUT IT IN FODS BUFFER
0AFE- E8       1870            INX
0AFF- C8       1880            INY
0B00- C9 20    1890            CMP #$20     ;IS IT A BLANK?
0B02- D0 F4    1900            BNE DC2      ;NO, KEEP GOING
0B04- B9 35 01 1910            LDA CRT,Y    ;GET CHARACTER AFTER SPACE
0B07- C9 20    1920            CMP #$20     ;ANOTHER ONE?
0B09- D0 ED    1930            BNE DC2      ;NOPE, STILL MORE
0B0B- A9 0D    1940            LDA #$D      ;PUT IN THE <CR>
0B0D- 9D 7F 72 1950            STA TXBUF-1,X
0B10- 20 11 03 1960            JSR MON.I/O
0B13- 20 B3 79 1970            JSR CMDINT   ;LET FODS DEAL WITH IT
0B16- 4C 8F 0A 1980 DC3        JMP EN4      ;GO FINISH UP
               1990
               2000
0B19- 20 18 E3 2010 CON.DSK1   JSR TAPE1.OFF
0B1C- AD 14 01 2020            LDA CON.TAPE
0B1F- C9 0F    2030            CMP #$F
0B21- D0 1C    2040            BNE TAPE.D   ;REALLY A TAPE CALL
               2050
0B23- AD 25 01 2060            LDA TSTART+1
0B26- C9 01    2070            CMP #1
0B28- D0 1E    2080            BNE CT.OUT   ;IGNORE 2ND CALL
0B2A- 20 88 81 2090            JSR SAVER
0B2D- A0 0F    2100            LDY #$F
0B2F- B9 35 01 2110 NAM.CHK    LDA CRT,Y
0B32- C9 3B    2120            CMP #';
0B34- F0 09    2130            BEQ TAPE.D   ;LINE COMMENT - IGNORE IT
0B36- C9 20    2140            CMP #$20
0B38- D0 08    2150            BNE NAM.FND  ;WE FOUND A FILE NAME
0B3A- C8       2160            INY
0B3B- C0 50    2170            CPY #$50
0B3D- D0 F0    2180            BNE NAM.CHK
0B3F- 4C 3F 0A 2190 TAPE.D     JMP TAPE.DD
               2200
               2210
0B42- 20 BD 0A 2220 NAM.FND    JSR DSK.LOD
0B45- 20 2E 03 2230            JSR RAE.I/O
0B48- 20 4F 0B 2240 CT.OUT     JSR RST.DSK
0B4B- 18       2250            CLC
0B4C- 4C 80 EF 2260            JMP TAPE.FIN
               2270
               2280
               2290
               2300 ; RESET DRIVES OFF AND DEFAULT TO DRIVE 2
               2310
0B4F- A9 20    2320 RST.DSK    LDA #$20     ;DRIVES OFF CODE
0B51- 20 5D 76 2330            JSR DSKRW    ;GO TURN 'EM OFF
0B54- 20 34 76 2340            JSR SET2     ;DEFAULT IS DRIVE 2
0B57- 60       2350            RTS
               2360
               2370
               2380 ; CONVERT A TO 2 HEX DIGITS & PUT 'EM IN TXBUF
               2390
0B58- 48       2400 HEX.ASC    PHA
0B59- 4A       2410            LSR A
0B5A- 4A       2420            LSR A
```

```
0B5B- 4A        2430            LSR A
0B5C- 4A        2440            LSR A
0B5D- 20 09 83  2450            JSR NIBASC
0B60- 9D 80 72  2460            STA TXBUF,X
0B63- E8        2470            INX
0B64- 68        2480            PLA
0B65- 20 09 83  2490            JSR NIBASC
0B68- 9D 80 72  2500            STA TXBUF,X
0B6B- E8        2510            INX
0B6C- 60        2520            RTS
                2530
                2540
                2550 ; PUT NAME IN FODS BUFFER & PASS COMMAND TO FODS
                2560
0B6D- A9 3D     2570 NAM.CMD    LDA #'=
0B6F- 9D 80 72  2580            STA TXBUF,X
0B72- E8        2590            INX
0B73- A9 3A     2600            LDA #':      ;ALL RAE FILES BEGIN WITH ":"
0B75- 9D 80 72  2610            STA TXBUF,X
0B78- B9 35 01  2620 MOV.NAM    LDA CRT,Y    ;MOVE FILENAME INTO FODS BUFFEI
0B7B- E8        2630            INX
0B7C- C8        2640            INY
0B7D- 9D 80 72  2650            STA TXBUF,X
0B80- C9 20     2660            CMP #$20
0B82- D0 F4     2670            BNE MOV.NAM
0B84- A9 0D     2680            LDA #$D
0B86- 9D 80 72  2690            STA TXBUF,X  ;PUT IN A <CR>
0B89- 20 11 03  2700            JSR MON.I/O
0B8C- 4C B3 79  2710            JMP CMDINT   ;AND LET FODS FIGURE IT OUT
                2720
                2730
                2740 ; TELL FODS WHERE RAE TEXT AREA BEGINS
                2750
0B8F- AD 01 01  2760 RAE.STR    LDA TXST+1   ;CONVERT RAE
0B92- 20 58 0B  2770            JSR HEX.ASC  ;START ADDRESS
0B95- AD 00 01  2780            LDA TXST     ;TO ASCII
0B98- 20 58 0B  2790            JSR HEX.ASC  ;AND PUT IN FODS BUFFER
0B9B- 60        2800            RTS
                2810
                2820
                2830 ; TELL FODS WHERE RAE TEXT AREA ENDS
                2840
0B9C- A5 D4     2850 RAE.END    LDA *TPRES+1          ;CONVERT RAE
0B9E- 20 58 0B  2860            JSR HEX.ASC  ;END ADDRESS
0BA1- A5 D3     2870            LDA *TPRES   ;TO ASCII
0BA3- 20 58 0B  2880            JSR HEX.ASC  ;AND PUT IN FODS BUFFER
0BA6- 60        2890            RTS
                2900 ;
0BA7- 45 4E 54  2910 SAVE.D     .BY 'ENT $' $00
0BAA- 20 24 00
0BAD- 4C 4F 44  2920 LOD        .BY 'LOD $' $00
0BB0- 20 24 00
                2930 ;
0BB3- B9 35 01  2940 DRIVE?     LDA CRT,Y    ;DRIVE# ?
0BB6- C9 31     2950            CMP #$31     ; 1 ?
0BB8- F0 05     2960            BEQ DRIV     ;YES
0BBA- C9 32     2970            CMP #$32     ; 2 ?
0BBC- F0 01     2980            BEQ DRIV     ;YES
```

```
0BBE- 60        2990 NODRIV     RTS
0BBF- B9 36 01  3000 DRIV       LDA CRT+1,Y  ;FOLLOWED BY A
0BC2- C9 20     3010            CMP #$20     ;SPACE ?
0BC4- D0 F8     3020            BNE NODRIV   ;NO-NOT DRIVE#
0BC6- CA        3030            DEX
0BC7- B9 35 01  3040            LDA CRT,Y
0BCA- 9D 80 72  3050            STA TXBUF,X  ;ADD NUMBER
0BCD- E8        3060            INX
0BCE- A9 2F     3070            LDA #'/      ;AND '/'
0BD0- 9D 80 72  3080            STA TXBUF,X
0BD3- E8        3090            INX
0BD4- A9 24     3100            LDA #'$      ;PLUS '$'
0BD6- 9D 80 72  3110            STA TXBUF,X
0BD9- E8        3120            INX
0BDA- 4C FF B4  3130            JMP NXT.FLD  ;SKIP TO NAME
                3140            ***
                3150 ;
                3160 END.PGM    .EN
```

```
LABEL FILE:  [ / = EXTERNAL ]


/ACCESS=8B86          /ADDPAD=011A          /ALT=005F
/APP=00B2             /ASC.SP>DEC=B2E6      /AUTO=0115
/B.EEP=8972           /BEL=0007             /BKSL=005C
/BS=0008              /BUFFER=5F00          /CL.STAB=BF37
/CL.TXT=B096          /CMDINT=79B3          /CON.TAPE=0114
/CRLF=E3CA            /CRT=0135             /CRT.IN=B5BB
/CRTI=0080            /CRTO=0010            /CTROLYVEC=0000
/CURNT=00FC           /DEL=007F             /DIREC=00B3
/DISC1=00F0           /DISC2=00F2           /DISCC.VEC=00EC
/DISCI=00EE           /DISCI.VEC=00F6       /DISCO=00EF
/DISPAT=814A          /DSKRW=765D           /DUMPT=8E87
/EAH=A64B             /EAL=A64A             /ELABEL=5EFD
/EOT=00A8             /ERMSG=8171           /ERROO=B44E
/ERROR=B00E           /ERRORS=00DB          /ETEXT=3FFD
/EXFLG=00B4           /EXPTR=00FA           /FILE.NO=0110
/FIRST=0108           /FODBRK=7AF0          /FODS=0001
/FORMAT=010F          /GETCOM=80FF          /HEX/DEC=0111
/HSBDRY=A632          /ID=A64E              /INCBY=010C
/INT.CHR=8A58         /INVEC=A660           /LAST=010A
/LEN=00B1             /LINE/CNT=0120        /MANU=010E
/MRK.END=E05F         /NIBASC=8309          /NUMFLG=00B5
/NXT.FLD=B4FF         /NXT.PTR=00FE         /ORIGIN=0200
/OUT.BYT=E3E2         /OUTVEC=A663          /PAGE/NUM=0121
/PAS.SPCS=B502        /PASS=0113            /PR.LAB.S=E24A
/PRINT/CTL=011F       /PROC>ADDRS=00D1      /PRTVEC=00B6
/PUREC=00C8           /RAE.COLD=B04B        /RAE.HOT=B05E
/RAE.WARM=B0AE        /RESXAF=81B8          /RESXF=81BE
/SAH=A64D             /SAL=A64C             /SAVER=8188
/SAVEYY=00C7          /SCRAT0=00DD          /SCRATCH=00B0
/SDBYT=A651           /SET.F2=B214          /SET2=7634
/SLABEL=4000          /SPACE=8342           /SPC=0020
/STEN=0106            /STEXT=0BDD           /STST=0104
/SUP.OUT=00E3         /T.OUT=8AA0           /TAPDEL=A630
/TAPE.FIN=EF80        /TAPE1.OFF=E318       /TAPE1.ON=E32A
/TAPET1=A635          /TAPET2=A63C          /TECHO=A653
```

```
/TLOAD=EF68          /TOG0=EC64           /TOG1=EC59
/TOPMEM=6000         /TOUTFL=A654         /TPRES=00D3
/TSTART=0124         /TTYI=0040           /TTYO=0020
/TXBUF=7280          /TXEN=0102           /TXST=0100
/UBRKVC=A676         /UPDFLG=009E         /USERVEC=0003
/USRENT=8035         /VIADDB=A002         /VIADRB=A000
/YMAX=009F           ACCEPT=0579          ACCEPT1=057E
ADR=0775             ANO=07C6             ANO1=07CC
APERR=07AA           APPEND=0796          BCKSP=039A
BCKSPC=063B          BEEP=0212            BEEP1=04A0
BEEP11=0401          BEEP2=0576           BUSY=02E2
CALC.NXT=0996        CAS.INIT=026C        CHAR.=0562
CHK.STK=03A6         CL1=0961             CL2=0973
CL3=0988             CMD=06BD             CMD.PR=0669
COLD=0218            COMPARE=092F         COMPARE1=0930
CON.DSK=0A3C         CON.DSK1=0B19        CRLFP=0640
CRLFP1=0651          CRTIN=0364           CRTIN1=036D
CRTIN2=0373          CRTIN3=037B          CRTOUT=037C
CT.OUT=0B48          CTLM.L=0493          CTRLD=046B
CTRLM=0479           CTRLU=0461           CTRM=0487
CTROLA=03F6          CTROLA1=040E         CTROLC=041D
CTROLH=042F          CTROLH1=043F         CTROLI=0440
CTROLJ=044E          CTROLM=0475          CTROLP=04A6
CTROLP1=04B6         CTROLS=04BE          CTROLT=04CE
CTROLT1=04DE         CTROLX=04E7          CTROLX0=04F8
CTROLX1=0501         CTROLY=050A          CTROLY1=050E
CTROLZ=0520          CTROLZ1=0524         CTROLZ2=052F
D.PARM=02B1          D.PARM1=02B3         D.TAB=02D1
DC1=0AF6             DC2=0AF8             DC3=0B16
DELCHR=061F          DELCHR1=0629         DELCR=0633
DELETE=056A          DISK.CMD=0AED        DO.BS=0389
DO.NUM=0745          DOWN=07F7            DRIV=0BBF
DRIVE?=0BB3          DSK.LOD=0ABD         DSK.SET=0A39
DUMA=0417            ECHAR.VEC=03D7       ECHAR.VEC1=03DF
ECHO=0551            ECHO1=055D           EDERR=06EE
EDIT=06F3            EDIT1=06FE           EEOF=05AB
EEOF1=05B2           EEOF2=05BC           EEOF3=05C6
EN1=0A6E             EN2=0A72             EN3=0A7D
EN4=0A8F             END.CURNT=094E       END.FIL=07F0
END.LABEL=0940       END.PGM=0BDD         ENDFIL=08C5
ENDLN=0728           ENTER=0A65           ESCAPE=0534
ESCAPE1=053D         EXCHANGE=0956        EXEC=082E
EXECT=0842           FIL.BOT=0821         FILBUF=03E2
FILE=089B            FIND.NXT=091B        FINEX=086F
FIX.STK=03BF         FOUD=06AB            GET.N=0876
GETCH=0357           GETCH1=035E          GETCR=0853
HEX.ASC=0B58         INSERT=05C7          INSERT1=05CC
INSERT2=05D5         INSERT20=05DD        INSERT3=05E9
INSERT4=05FF         INTCHR=0209          IPADDS=08BE
IPADDS1=08C4         KBAUD=0216           KTA=0409
KTM.I/O=02F9         KTMFLG=0217          LARG=082D
LO1=0AA1             LO3=0AC8             LO4=0AD8
LO5=0ADB             LO6=0AB2             LOAD=0A98
LOD=0BAD             MESSUB=065D          MESSUB1=0668
MGETCH=034B          MON.I/O=0311         MOV.NAM=0B78
MOVDWN=0603          MOVDWN1=0605         MOVDWN2=0612
N.PAR=0882           NAM.CHK=0B2F         NAM.CMD=0B6D
NAM.FND=0B42         NEW.COLD=0200        NEW.CR=071D
```

```
NEW.CRT=03BD         NEW.HOT=0206         NEW.WARM=0203
NEXCHR=03EB          NEXCHR1=04A3         NEXCHR2=054E
NEXT=07D4            NEXT1=07DA           NOBSL=0392
NOCM=06A8            NODRIV=0BBE          NOED=0864
NOFUD=0790           NOSKP=085A           NOT.YET=03B8
NUM.BUF=0760         NXC=067E             NXC0=067C
NXT.D=06A2           NXT.LAB=0993         NXT.LABEL=0911
NXTL=08AB            NXTL1=08B4           NXTL2=08B1
NXTLT=0698           PART2=07B1           PART3=07BC
PBAUD=0215           PRMT.OT=03C3         PRMT.OUT=03CF
PRT.I/O=0305         RAE.END=0B9C         RAE.I/O=032E
RAE.STR=0B8F         RESTORE=0885         RST.DSK=0B4F
RTSE=086C            SAVE=09A3            SAVE.D=0BA7
SENDCHR=039C         SET.EM=0A4A          SET.VEC=0278
SORT=08FF            SORT.EXIT=092C       SPD1=08E5
SPD2=08F7            STBUFF=03E5          STORE=08EB
SUBL=0800            TAPE=08D0            TAPE.D=0B3F
TAPE.DD=0A3F         TERR=08E2            TEXT=09DB
TOLAR=0759           TOUT=020C            TSTL=080B
TTY.OUT=02DB         TTY1=02F1            TTYOUT=020F
UPDATE=0586          UPDATE1=058D         UPDATE2=0594
UPDATE3=059C         UPDATE4=05A6         WARM=0260
WRT.HOOK=039F        ZERBUF=0652          ZERBUF1=0656
©C.CHK=0A2B          ©C.CHK1=0A36
//0000,0BDD,0BDD
>SE

2000-5FFC  6000-7EFC  7F00
3902  6AEE
//

>
```