

```

EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF
EEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF
EEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF

```

[illegible]

```

EEEEEEEEEE DDDDDDDD FFFFFFFF TTTTTTTTTT EEEEEEEEE RRRRRRRR MM MM
EEEEEEEEEE DDDDDDDD FFFFFFFF TTTTTTTTTT EEEEEEEEE RRRRRRRR MM MM
EE DD DD FF TT EE RR RR MMMM MMMM
EE DD DD FF TT EE RR RR MMMM MMMM
EE DD DD FF TT EE RR RR MM MM MM
EE DD DD FF TT EE RR RR MM MM MM
EEEEEEEEEE DD DD FFFFFFFF TT TT EEEEEEEEE RRRRRRRR MM MM
EEEEEEEEEE DD DD FFFFFFFF TT TT EEEEEEEEE RRRRRRRR MM MM
EE DD DD FF TT TT EE EE RR RR MM MM
EE DD DD FF TT TT EE EE RR RR MM MM
EE DD DD FF TT TT EE EE RR RR MM MM
EE DD DD FF TT TT EE EE RR RR MM MM
EEEEEEEEEE DDDDDDDD FF TT EEEEEEEEEEE RR RR MM MM
EEEEEEEEEE DDDDDDDD FF TT EEEEEEEEEEE RR RR MM MM

```

....
....
....
....

```

LL I I I I I SSSSSSSS
LL I I I I I SSSSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LL II SSSSSS
LL II SSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LLLLLLLLLLLL I I I I I SSSSSSSS
LLLLLLLLLLLL I I I I I SSSSSSSS

```



```
0000 1      .TITLE EDF$TERM - MACRO-32 SUPPORT ROUTINES FOR EDF.
0000 2      .IDENT 'V04-000'
0000 3      : ++
0000 4      : *****
0000 5      : *
0000 6      : *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 7      : *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 8      : *  ALL RIGHTS RESERVED.
0000 9      : *
0000 10     : *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 11     : *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 12     : *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 13     : *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 14     : *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 15     : *  TRANSFERRED.
0000 16     : *
0000 17     : *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 18     : *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 19     : *  CORPORATION.
0000 20     : *
0000 21     : *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 22     : *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 23     : *
0000 24     : *****
0000 25     : *****
0000 26     :
0000 27     : FACILITY:          VAX/VMS FDL EDITOR (EDF Facility)
0000 28     :
0000 29     : ABSTRACT:         MACRO-32 subroutines for EDF
0000 30     :
0000 31     : ENVIRONMENT:       NATIVE/USER MODE
0000 32     :
0000 33     : AUTHOR:           Ken F. Henderson Jr.
0000 34     :
0000 35     : CREATION DATE:    7-Apr-1981
0000 36     :
0000 37     : MODIFIED BY:
0000 38     :          V03-002 KFH0002      Ken Henderson      9 Aug 1983
0000 39     :          Remove clear of wrap bit.
0000 40     :
0000 41     :          V03-001 KFH0001      Ken Henderson      14 Apr 1983
0000 42     :          Return EDF$DEVCLASS in R0 if not
0000 43     :          terminal, rather than calling
0000 44     :          LIB$STOP. Also, clear ITSM_WRAP bit
0000 45     :          in the SETMODE.
0000 46     :
0000 47     : --
```

```
0000 49 $$$DEF
0000 50 $DIBDEF
0000 51 $IODEF
0000 52 $DCDEF
0000 53 $TTDEF
0000 54 $TT2DEF
0000 55
00000000 56 .PSECT EDF$TERM_DATABASE, LONG
0000 57 : ++
0000 58 :
0000 59 : This is the static database for the module.
0000 60 : It contains the necessary data blocks for the QIO services.
0000 61 :
0000 62 :
0000 63 : --
0000 64
4E 49 24 53 59 53 00000008'010E0000' 0000 65 TTY: .ASCID /SYSS$INPUT:/
3A 54 55 50 000E
0012 66
00000014 0012 67 TTYCHN: .BLKW 1
00000015 0014 68 TTYIN: .BLKB 1
0015 69
0015 70 .ALIGN LONG
0018 71
00000020 0018 72 TTYIOSB: .BLKW 4
0000001A 0020 73 TERMSPEED = TTYIOSB + 2
0020 74
00000038 0020 75 EDF$OLDMODE:: .BLKW 12
00000021 0038 76 TERMNUM = EDF$OLDMODE + 1
00000024 0038 77 TERMCHR = EDF$OLDMODE + 4
00000028 0038 78 EXTRABITS = EDF$OLDMODE + 8
0038 79
00000050 0038 80 EDF$NEWMODE:: .BLKW 12
0000003C 0050 81 NEWCHR = EDF$NEWMODE + 4
0000003F 0050 82 TERMPAGE = EDF$NEWMODE + 7
0050 83
000000C4 0050 84 DIB: .BLKB DIB$K_LENGTH
0000 0074 00C4 85 DIB_DESC: .WORD DIB$K_LENGTH, 0
00000050' 00C8 86 .ADDRESS DIB
00CC 87
00CC 88 .ALIGN LONG
00CC 89
00CC 90 EXITBLOCK:
00000000 00CC 91 .LONG 0 ; exit control block
00000118' 00D0 92 .ADDRESS EDF$RESET_TERM ; system uses this for pointer
00000001 00D4 93 .LONG 1 ; address of exit handler (to reset
000000DC' 00D8 94 .ADDRESS STATUS ; number of args for handler
000000E0 00DC 95 STATUS: .BLKL 1 ; address to receive status code
; status code
```



```
00000000 97 .PSECT EDF$TERM_CODE, LONG
0000 98 : ++
0000 99 :
0000 100 : EDF$TERM_SETUP -- Set up the terminal and get its speed.
0000 101 :
0000 102 : This routine queries the terminal driver to see what's the
0000 103 : speed of the terminal we're talking to.
0000 104 : It also checks to make sure that indeed the input is a terminal,
0000 105 : and it also sets the page length to 16 for hardcopy terminals.
0000 106 : It also declares an exit handler and sets up a control/C AST.
0000 107 :
0000 108 : CALLING SEQUENCE:
0000 109 :
0000 110 : TERMINAL_SPEED := EDF$TERM_SETUP;
0000 111 :
0000 112 :
0000 113 : --
0000 114 :
0000 115 : .ENTRY EDF$TERM_SETUP, ^M<>
0002 116 :
0002 117 : $ASSIGN S -
0002 118 : DEVNAM = TTY, -
0002 119 : CHAN = TTYCHN
0017 120 :
0017 121 : BLBS R0, 7$
001A 122 : BRW ERROR
001D 123 :
001D 124 7$: $GETCHN S -
001D 125 : CHAN = TTYCHN, -
001D 126 : PRIBUF = DIB_DESC
0037 127 :
0037 128 : BLBS R0, 9$
003A 129 : BRW ERROR
003D 130 9$: CMPB DIB+DIB$B_DEVCLASS, #DC$ TERM
0045 131 : BEQL PAST
0047 132 :
0047 133 : MOVL #EDF$_DEVCLASS, R0 ; INDICATE IT'S NOT A TERMINAL
004E 134 : BRW OUT
0051 135 :
0051 136 PAST: $QIOW S -
0051 137 : CHAN = TTYCHN, -
0051 138 : FUNC = #IOS$ SENSEMODE, -
0051 139 : IOSB = TTYIOSB, -
0051 140 : P1 = EDF$OLDMODE
0078 141 :
0078 142 : MOVQ EDF$OLDMODE, EDF$NEWMODE ; COPY EDF$OLDMODE INTO EDF$NEWMODE
0083 143 : MOVQ EDF$OLDMODE+8, EDF$NEWMODE+8
008E 144 : MOVQ EDF$OLDMODE+16, EDF$NEWMODE+16
0099 145 :
0099 146 : BBS #TT$V_SCOPE, TERMCHR, - ; SKIP DOWN IF IT'S A SCOPE
00A0 147 : VIDEO
00A1 148 :
00A1 149 : MOVB #16, TERMPAGE ; SET PAGE = 16 LINES
00A8 150 : ; HARDCOPY TERMINALS ARE SPECIAL
00A8 151 :
00A8 152 VIDEO:
00A8 153 : $QIOW S -
```

```
00A8 154          CHAN = TTYCHN,-
00A8 155          FUNC = #IOS SETMODE,-
00A8 156          P1    = EDF$NEWMODE
00CB 157
00CB 158          $DCLEXH_S    DESBLK=EXITBLOCK      ; DECLARE THE EXIT HANDLER
00D8 159          CALLS #0,    EDF$CTRLCAST        ; SETUP ^C AST ROUTINE
00D8 160
00DF 161          MOVZBL TERMSPEED,    R0
50 000000F0'EF 00 FB 00D8 162 OUT: RET
00DF 163 ERROR:
00E6 164 $EXIT_S R0
00E7 165
00E7 166
```



```
00F0 168 : ++
00F0 169 :
00F0 170 : EDF$CTRLCAST -- Set up ^C ast.
00F0 171 :
00F0 172 : This routine sets up another ^C ast.
00F0 173 :
00F0 174 : CALLING SEQUENCE:
00F0 175 :
00F0 176 : EDF$CTRLCAST;
00F0 177 :
00F0 178 : --
00F0 179 :
0000 00F0 180 .ENTRY EDF$CTRLCAST,^M<>
00F2 181 $QIOW_S -
00F2 182 CHAN = TTYCHN,- ; SET UP THE CONTROL/C AST
00F2 183 FUNC = #IOS_SETMODE!IOSM_CTRLCAST,-
00F2 184 P1 = CTRLCAST_ROUTINE
00F2 185
04 0117 186
0117 187 RET
```

```
0118 189 : ++
0118 190 :
0118 191 : EDF$RESET_TERM -- Reset TTDRV to its original mode.
0118 192 :
0118 193 : This routine returns TTDRV to it start mode, plus it calls
0118 194 : EDF$RESET_SCROLL to put the scrolling back to full screen.
0118 195 :
0118 196 : CALLING SEQUENCE:
0118 197 :
0118 198 : EDF$RESET_TERM;
0118 199 :
0118 200 : --
0000 0118 201 :
0118 202 : .ENTRY EDF$RESET_TERM,^M<>
011A 203 : $QIOW_S -
011A 204 :     CHAN = TTYCHN,-
011A 205 :     FUNC = #IOS$ SETMODE,-
011A 206 :     P1   = EDF$OLDMODE
013D 207 :
00000000'EF 00 FB 013D 208 : CALLS #0, EDF$RESET_SCROLL ; RESET VT100'S SCROLL REGION
04 0144 209 : RET
0144 210 :
```



```
0145 212 : ++
0145 213 :
0145 214 : CTRLCAST_ROUTINE -- Catch Control/C when typed by the user.
0145 215 :
0145 216 : This routine does a SYS$EXIT to call the exit handler before leaving.
0145 217 :
0145 218 : CALLING SEQUENCE:
0145 219 :
0145 220 : called as an AST routine
0145 221 :
0145 222 : --
0145 223 :
0000 0145 224 : .ENTRY CTRLCAST_ROUTINE,^M<>
0147 225 :
0147 226 : $EXIT_S ; invoke image rundown, which calls the exit handler
0150 227 :
04 0150 228 : RET ; never gets here - we hope!
0151 229 :
0151 230 : .END
```

Variable	Value	Mode	Priority
\$\$T1	= 00000001		
CTRLCAST_ROUTINE	00000145	RG	03
DC\$ TERM	= 00000042		
DIB	00000050	R	02
DIB\$B_DEVCLASS	= 00000004		
DIB\$K_LENGTH	= 00000074		
DIB DESC	000000C4	R	02
EDF\$CTRLCAST	000000F0	RG	03
EDF\$NEWMODE	00000038	RG	02
EDF\$OLDMODE	00000020	RG	02
EDF\$RESET_SCROLL	*****	X	03
EDF\$RESET_TERM	00000118	RG	03
EDF\$TERM_SETUP	00000000	RG	03
EDF\$ DEVCLASS	*****	X	03
ERROR	000000E7	R	03
EXITBLOCK	000000CC	R	02
EXTRABITS	= 00000028	R	02
IOSM_CTRLCAST	= 00000100		
IOS_SENSEMODE	= 00000027		
IOS_SETMODE	= 00000023		
NEWCHR	= 0000003C	R	02
OUT	000000E6	R	03
PAST	00000051	R	03
STATUS	000000DC	R	02
SYSS\$ASSIGN	*****	GX	03
SYSS\$DCLEXH	*****	GX	03
SYSS\$EXIT	*****	GX	03
SYSS\$GETCHN	*****	GX	03
SYSS\$QIOW	*****	GX	03
TERMCHR	= 00000024	R	02
TERMMUM	= 00000021	R	02
TERMPAGE	= 0000003F	R	02
TERMSPEED	= 0000001A	R	02
TT\$V_SCOPE	= 0000000C		
TTY	00000000	R	02
TTYCHN	00000012	R	02
TTYIN	00000014	R	02
TTYIOSB	00000018	R	02
VIDEO	000000A8	R	03

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
EDF\$TERM_DATABASE	000000E0 (224.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG
EDF\$TERM_CODE	00000151 (337.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
-----	-----	-----	-----
Initialization	33	00:00:00.13	00:00:00.72
Command processing	123	00:00:00.41	00:00:01.96
Pass 1	318	00:00:10.75	00:00:26.42
Symbol table sort	0	00:00:01.91	00:00:04.11
Pass 2	56	00:00:01.90	00:00:03.96
Symbol table output	6	00:00:00.08	00:00:00.08
Psect synopsis output	2	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	541	00:00:15.22	00:00:37.29

The working set limit was 1200 pages.
61132 bytes (120 pages) of virtual memory were used to buffer the intermediate code.
There were 70 pages of symbol table space allocated to hold 1211 non-local and 2 local symbols.
230 source lines were read in Pass 1, producing 27 object records in Pass 2.
22 pages of virtual memory were used to define 21 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
-----	-----
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	18

1350 GETS were required to define 18 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LISS:EDF\$TERM/OBJ=OBJ\$:EDF\$TERM MSRC\$:EDF\$TERM/UPDATE=(ENH\$:EDF\$TERM)

0128

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY