



```

EEEEEEEEEE  RRRRRRRR  FFFFFFFFFF  000000  UU      UU  TTTTTTTTTT  PPPPPPPP  UU      UU  TTTTTTTTTT
EEEEEEEEEE  RRRRRRRR  FFFFFFFFFF  000000  UU      UU  TTTTTTTTTT  PPPPPPPP  UU      UU  TTTTTTTTTT
EE          RR      RR  FF          00      00  UU      UU  TT          PP      PP  UU      UU  TT          TT
EE          RR      RR  FF          00      00  UU      UU  TT          PP      PP  UU      UU  TT          TT
EE          RR      RR  FF          00      00  UU      UU  TT          PP      PP  UU      UU  TT          TT
EEEEEEEEEE  RRRRRRRR  FFFFFFFF  00      00  UU      UU  TT          P P P P P P P P  UU      UU  TT          TT
EEEEEEEEEE  RRRRRRRR  FFFFFFFF  00      00  UU      UU  TT          P P P P P P P P  UU      UU  TT          TT
EE          RR  RR      FF          00      00  UU      UU  TT          PP          UU      UU  TT          TT
EE          RR  RR      FF          00      00  UU      UU  TT          PP          UU      UU  TT          TT
EE          RR      RR  FF          00      00  UU      UU  TT          PP          UU      UU  TT          TT
EE          RR      RR  FF          00      00  UU      UU  TT          PP          UU      UU  TT          TT
EEEEEEEEEE  RR      RR  FF          00      00  UU      UU  TT          PP          UU      UU  TT          TT
EEEEEEEEEE  RR      RR  FF          000000  UUUUUUUUUU  TT          PP          UUUUUUUUUU  TT          TT          TT          TT
EEEEEEEEEE  RR      RR  FF          000000  UUUUUUUUUU  TT          PP          UUUUUUUUUU  TT          TT          TT          TT

```

.....

.....

.....

.....

```

LL          IIIIII  SSSSSSSS
LL          IIIIII  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
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS

```

ER VO  
.....



```
1 0001 0 MODULE ERFOUTPUT
2 0002 0 (%TITLE 'ERF output routines'
3 0003 0 IDENT = 'V04-000') =
4 0004 0
5 0005 1 BEGIN
6 0006 1
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
12 0012 1 * ALL RIGHTS RESERVED. *
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
19 0019 1 * TRANSFERRED. *
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
23 0023 1 * CORPORATION. *
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 ++
32 0032 1 FACILITY: ERF, Error Log Report Generator
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 This module contains the text outputting routines for ERF.
37 0037 1
38 0038 1 ENVIRONMENT:
39 0039 1
40 0040 1 VAX/VMS operating system, user mode.
41 0041 1
42 0042 1 AUTHOR: Sharon Reynolds, CREATION DATE: Feb-1984
43 0043 1
44 0044 1
45 0045 1 --
46 0046 1 REQUIRE 'SRC$:ERFDEF.REQ' ;
47 0332 1
48 0333 1 EXTERNAL ROUTINE
49 0334 1 Write_msg ;
50 0335 1
```

```

52 0336 1 GLOBAL Routine OUTPUT_LINES : NOVALUE =
53 0337 Begin
54 0338
55 0339 !++
56 0340
57 0341 Calling Sequence:
58 0342
59 0343 OUTPUT_LINES (control_string,param_list)
60 0344 NOTE: This routine will calculate the actual number of parameters
61 0345 that were passed to it. This means that there can be multiple
62 0346 sets (control_string/param_list) of call parameters specified.
63 0347
64 0348 Input Parameters:
65 0349
66 0350 Control_string = address of descriptor containing fao control string.
67 0351 Param_list = address of parameter list.
68 0352
69 0353 Output Parameters:
70 0354
71 0355 None
72 0356
73 0357 !--
74 0358
75 0359 BUILTIN
76 0360 Actualcount,
77 0361 Actualparameter ;
78 0362
79 0363 LOCAL
80 0364 Msg_buffer: Vector[1024,byte],
81 0365 Control_string,
82 0366 Msg_desc: Vector [2],
83 0367 Out_flag: Initial (0),
84 0368 Param_list,
85 0369 Status ;
86 0370
87 0371
88 0372 ! Get the fao control string and the parameter list.
89 0373
90 0374 ! Incr I from 1 to ACTUALCOUNT()/2 do
91 0375 Begin
92 0376 Control_string = ACTUALPARAMETER(2*(.I - 1) + 1) ;
93 0377 Param_list = ACTUALPARAMETER(2*(.I - 1) + 2) ;
94 0378
95 0379 Msg_desc[0] = %ALLOCATION(msg_buffer) ;
96 0380 Msg_desc[1] = msg_buffer ;
97 0381
98 0382 P P If NOT (status = $FAOL (CTRSTR = .control_string,
99 0383 OUTBUF = msg_desc,
100 0384 OUTLEN = msg_desc,
101 0385 PRMLST = .param_list))
102 0386
103 0387 Then Signal_stop (.status) ;
104 0388
105 0389
106 0390
107 0391 ! Output the message to the user specified output device.
108 0392

```



```

: 109      0393 3      WRITE_MSG (msg_desc,out_flag) ;
: 110      0394      End ;
: 111      0395      End ;
: 112      0396      Return ;
: 113      0397 2      End ;
: 114      0398 1      End ; ! routine
    
```

```

.TITLE ERFOUTPUT ERF output routines
.IDENT \V04-000\

.EXTRN WRITE_MSG, SYSSFAOL

.PSECT $CODE,NOWRT, PIC,2

.ENTRY OUTPUT_LINES, Save R2,R3,R4,R5,R6
MOVAB -1032(SP), SP
CLRL OUT_FLAG
MOVZBL (AP), R3
DIVL2 #2, R3
CLRL I
BRB 3$
ASHL #1, I, R0
MOVL -4(AP)[R0], CONTROL_STRING
ASHL #1, I, R0
MOVL (AP)[R0], PARAM_LIST
MOVZWL #1024, MSG_DESC
MOVAB MSG_BUFFER, MSG_DESC+4
PUSHL PARAM_LIST
PUSHAB MSG_DESC
PUSHAB MSG_DESC
PUSHL CONTROL_STRING
CALLS #4, SYSSFAOL
MOVL R0, STATUS
BLBS STATUS, 2$
PUSHL STATUS
CALLS #1, LIB$STOP
PUSHL SP
PUSHAB MSG_DESC
CALLS #2, WRITE_MSG
AOBLEQ R3, I, 1$
RET
    
```

```

: 0336
: 0337
: 0374
:
: 0376
: 0377
: 0379
: 0380
: 0385
:
: 0387
: 0393
:
: 0374
: 0398
    
```

: Routine Size: 96 bytes, Routine Base: \$CODE + 0000

```

: 115      0399 1
: 116      0400 1 End      ! module
: 117      0401 0 ELUDOM
    
```

.EXTRN LIB\$STOP

PSECT SUMMARY

```

:      Name                Bytes                Attributes
: $CODE                    96 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON, PIC,ALIGN(2)

```

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	23	0	1000	00:02.0

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:ERFOUTPUT/OBJ=OBJ\$:ERFOUTPUT MSRC\$:ERFOUTPUT/UPDATE=(ENH\$:ERFOUTPUT)

```

: Size:          96 code + 0 data bytes
: Run Time:      00:06.7
: Elapsed Time: 00:22.4
: Lines/CPU Min: 3607
: Lexemes/CPU-Min: 24512
: Memory Used: 79 pages
: Compilation Complete

```



The image displays a grid of 144 small document thumbnails, arranged in 12 rows and 12 columns. Each thumbnail represents a different software component or manual page. The thumbnails are arranged in a grid, with some larger and more prominent than others. The following table lists the titles of the thumbnails that are clearly legible:

Row	Column	Thumbnail Title
1	10	ERFCOM LIS
1	11	ERFMSG LIS
2	10	ERF5VECT LIS
3	10	ERF2VECT LIS
3	11	ERFOUTPUT LIS
4	10	ERFCOMVEC LIS
5	10	ERFBRIVEC LIS
6	10	ERF3VECT LIS
6	11	ERFDISKVE LIS
7	11	ERFPARSER LIS
8	10	ERFBUSVEC LIS
8	11	ERFINIVEC LIS
9	10	ERF4VECT LIS
9	11	ERFINICOM LIS
10	10	ERF1VECT LIS