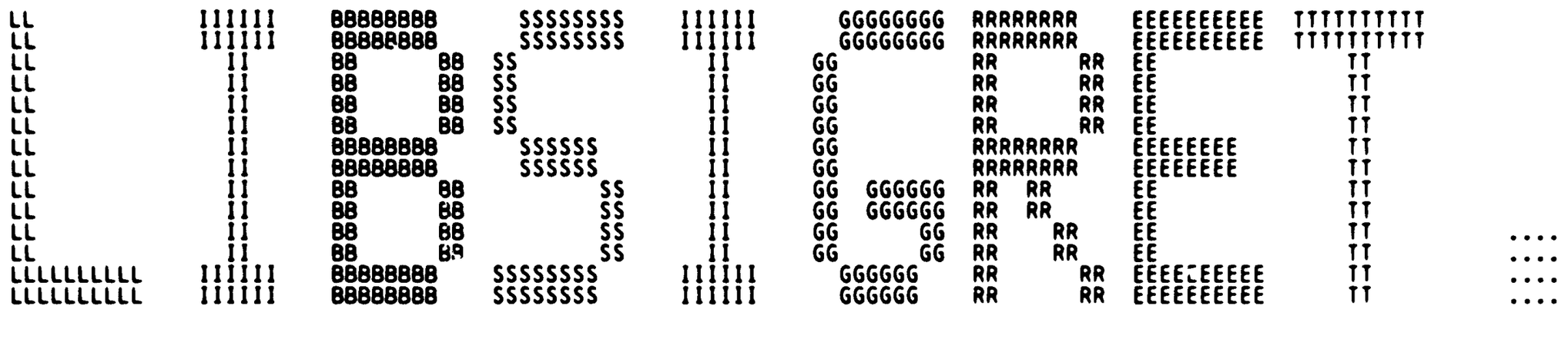
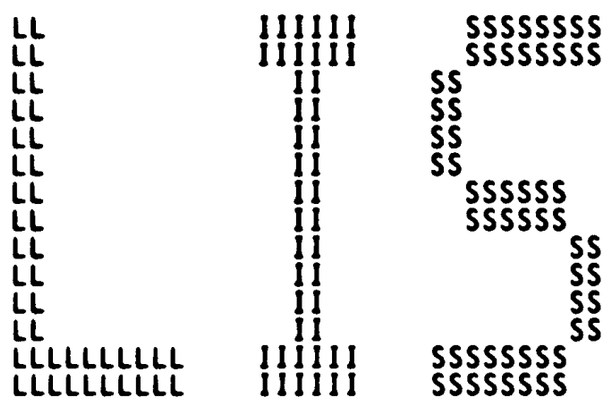
```
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                 IIIIIIIII     BBBBBBBBBBBB     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBB         BBB  RRR         RRR        TTT        LLL
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           TTT        LLL
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           TTT        LLL
LLL                    III        BBBBBBBBBBBB     RRRRRRRRRRRR           TTT        LLL
LLL                    III        BBB         BBB  RRR   RRR              TTT        LLL
LLL                    III        BBB         BBB  RRR   RRR              TTT        LLL
LLL                    III        BBB         BBB  RRR   RRR              TTT        LLL
LLL                    III        BBB         BBB  RRR      RRR           TTT        LLL
LLL                    III        BBB         BBB  RRR      RRR           TTT        LLL
LLL                    III        BBB         BBB  RRR      RRR           TTT        LLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
LLLLLLLLLLLLLLL     IIIIIIIII     BBBBBBBBBBBB     RRR         RRR        TTT        LLLLLLLLLLLLLLL
```


LL IIIIII BBBBBBBB SSSSSSSS IIIIII GGGGGGGG RRRRRRRR EEEEEEEEEE TTTTTTTTTT
LL IIIIII BBBBBBBB SSSSSSSS IIIIII GGGGGGGG RRRRRRRR EEEEEEEEEE TTTTTTTTTT
LL II BB BB SS II GG RR RR EE TT
LL II BB BB SS II GG RR RR EE TT
LL II BB BB SS II GG RR RR EE TT
LL II BB BB SS II GG RR RR EE TT
LL II BBBBBBBB SSSSSS II GG RRRRRRRR EEEEEEEE TT
LL II BBBBBBBB SSSSSS II GG RRRRRRRR EEEEEEEE TT
LL II BB BB SS II GG GGGGGG RR RR EE TT
LL II BB BB SS II GG GGGGGG RR RR EE TT
LL II BB BB SS II GG GG RR RR EE TT
LL II BB BB SS II GG GG RR RR EE TT
LLLLLLLLLL IIIIII BBBBBBBB SSSSSSSS IIIIII GGGGGG RR RR EEEEEEEEEE TT
LLLLLLLLLL IIIIII BBBBBBBB SSSSSSSS IIIIII GGGGGG RR RR EEEEEEEEEE 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
LLLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLLL IIIIII SSSSSSSS

```
;     1          0001  0 MODULE LIB$SIG_TO_RET (%TITLE'Library routine to convert a signal to a return'
;     2          0002  0         IDENT = 'T-011'          ! File: LIBSIGRET.B32  Edit: SBL1011
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1
;     6          0006  1 !
;     7          0007  1 !*****************************************************************************
;     8          0008  1 !*                                                                           *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                                      *
;    12          0012  1 !*                                                                           *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    18          0018  1 !*  TRANSFERRED.                                                              *
;    19          0019  1 !*                                                                           *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    22          0022  1 !*  CORPORATION.                                                              *
;    23          0023  1 !*                                                                           *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*****************************************************************************
;    29          0029  1 !
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:  Utility Library
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       LIB$SIG_TO_RET converts any signaled condition value into
;    37          0037  1 !       an ordinary procedure return to the caller of the procedure
;    38          0038  1 !       which established the handler which called LIB$SIG_TO_RET.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:  User mode, re-entrant, AST level or not or mixed.
;    41          0041  1 !
;    42          0042  1 ! AUTHOR:       Thomas N. Hastings CREATION DATE:  09-June-1977
;    43          0043  1 !
;    44          0044  1 ! MODIFIED BY:
;    45          0045  1 !
;    46          0046  1 !       Thomas N. Hastings, 09-June-1977: VERSION 01
;    47          0047  1 ! 01    -  original
;    48          0048  1 ! 01-06 - Change to STARLET library.  DGP 20-Apr-78
;    49          0049  1 ! 01-07 - Change REQUIRE files for VAX system build.  DGP 28-Apr-78
;    50          0050  1 ! 01-08 - Change STARLET to RTLSTARLE to avoid conflicts.  DGP 1-May-78
;    51          0051  1 ! 01-09 - Change file name to LIBSIGRET.B32, and change the name of
;    52          0052  1 !           the REQUIRE file similarly.
;    53          0053  1 ! 1-010 - Update the copyright notice.  JBS 16-NOV-78
;    54          0054  1 ! 1-011 - Use prologue file.  SBL 24-June-1983
;    55          0055  1 !--
```

```
;    57          0056  1 !
;    58          0057  1 ! PROLOGUE FILE:
;    59          0058  1 !
;    60          0059  1
;    61          0060  1 REQUIRE 'RTLIN:LIBPROLOG';              ! LIB$ definitions
;    62          0131  1
;    63          0132  1 !
;    64          0133  1 ! TABLE OF CONTENTS:
;    65          0134  1 !
;    66          0135  1
;    67          0136  1 FORWARD ROUTINE
;    68          0137  1         LIB$SIG_TO_RET;         ! Convert a signal to a return
;    69          0138  1
;    70          0139  1 !
;    71          0140  1 ! MACROS:
;    72          0141  1 !
;    73          0142  1 !       NONE
;    74          0143  1 !
;    75          0144  1 ! EQUATED SYMBOLS:
;    76          0145  1 !
;    77          0146  1 !       NONE
;    78          0147  1 !
;    79          0148  1 ! OWN STORAGE:
;    80          0149  1 !
;    81          0150  1
;    82          0151  1 !
;    83          0152  1 ! EXTERNAL REFERENCES:
;    84          0153  1 !
;    85          0154  1
```

```
;    87          0155  1 GLOBAL ROUTINE LIB$SIG_TO_RET  (        ! Convert a signal to a return
;    88          0156  1         SIG_ARGS_ADR,    ! Adr. of signal args vector
;    89          0157  1         MCH_ARGS_ADR)    ! Adr. of mechanism args vector
;    90          0158  1                 =        ! Value is success, unless failure from SYS$UNWIND
;    91          0159  1
;    92          0160  1 !++
;    93          0161  1 ! FUNCTIONAL DESCRIPTION:
;    94          0162  1 !
;    95          0163  1 !       LIB$SIG_TO_RET is called with the argument list passed to a condition
;    96          0164  1 !       handler by the condition handling facility.  It converts the
;    97          0165  1 !       signaled condition into a return to the procedure which called the
;    98          0166  1 !       procedure which established the handler handling the signal.  The
;    99          0167  1 !       stack is unwound to the caller of the establisher and the condition code
;   100          0168  1 !       is returned as the value in R0.
;   101          0169  1 !
;   102          0170  1 !       In BLISS the argument list can be passed in toto by using the
;   103          0171  1 !       BUILTIN function CALLG and AP register, thus:
;   104          0172  1 !
;   105          0173  1 !               status = CALLG (.AP, LIB$SIG_TO_RET);
;   106          0174  1 !
;   107          0175  1 !       Or more simply by: ENABLE LIB$SIG_TO_RET ();
;   108          0176  1 !       If there is no need for the handler to do any more processing
;   109          0177  1 !
;   110          0178  1 ! FORMAL PARAMETERS:
;   111          0179  1 !
;   112          0180  1 !       SIG_ARGS_ADR.rl.ra      Adr. of signal args vector
;   113          0181  1 !       MCH_ARGS_ADR.ml.ra      Adr. of mechanism args vector
;   114          0182  1 !       any other args to handler
;   115          0183  1 !
;   116          0184  1 ! IMPLICIT INPUTS:
;   117          0185  1 !
;   118          0186  1 !       NONE
;   119          0187  1 !
;   120          0188  1 ! IMPLICIT OUTPUTS:
;   121          0189  1 !
;   122          0190  1 !       NONE
;   123          0191  1 !
;   124          0192  1 ! ROUTINE VALUE:
;   125          0193  1 ! COMPLETION CODES:
;   126          0194  1 !
;   127          0195  1 !       SS$_NORMAL if SYS$UNWIND ok, else error codes form SYS$UNWIND.
;   128          0196  1 !
;   129          0197  1 ! SIDE EFFECTS:
;   130          0198  1 !
;   131          0199  1 !       Causes the stack to marked to be unwound to the caller of the
;   132          0200  1 !       establishing procedure of the handler which was called on this signal.
;   133          0201  1 !
;   134          0202  1 !--
;   135          0203  1
;   136          0204  2     BEGIN
;   137          0205  2     MAP
;   138          0206  2         SIG_ARGS_ADR: REF BLOCK[8, BYTE],       ! Signal vector
;   139          0207  2         MCH_ARGS_ADR: REF BLOCK[20, BYTE];      ! mechanism vector
;   140          0208  2
;   141          0209  2     !+
;   142          0210  2     ! If this is unwind condition, just let unwinding continue since
;   143          0211  2     ! propably it was this handler which invoked the unwind.
```

```
;    2      !-
;    2
;    2      IF .SIG_ARGS_ADR[CHF$L_SIG_NAME] EQL SS$_UNWIND THEN RETURN SS$_NORMAL;
;    2
;    2      !+
;    2      ! Copy condition value to saved image of R0
;    2      !-
;    2
;    2      MCH_ARGS_ADR[CHF$L_MCH_SAVR0] = .SIG_ARGS_ADR[CHF$L_SIG_NAME];
;    2
;    2      !+
;    2      ! Set to unwind stack using default depth and default new PC,
;    2      ! namely return to caller of the procedure which established the handler
;    2      !-
;    2
;    2      RETURN $UNWIND ();
;    1      END;                          ! End of LIB$SIG_TO_RET routine


                                                        .TITLE  LIB$SIG_TO_RET Library routine to convert a sig
;                                                                              nal to a return
                                                        .IDENT  \1-011\

                                                        .EXTRN  SYS$UNWIND

                                                        .PSECT  _LIB$CODE,NOWRT,  SHR,  PIC,2

                                        0000 00000      .ENTRY  LIB$SIG_TO_RET, Save nothing            ; 0155
                 51        04  AC  D0 00002             MOVL    SIG_ARGS_ADR, R1                         ; 0214
      00000920   8F        04  A1  D1 00006             CMPL    4(R1), #2336                             ;
                               04  12 0000E             BNEQ    1$                                       ;
                 50            01  D0 00010             MOVL    #1, R0                                   ;
                                   04 00013             RET                                              ;
                 50        08  AC  D0 00014 1$:         MOVL    MCH_ARGS_ADR, R0                         ; 0220
            0C   A0        04  A1  D0 00018             MOVL    4(R1), 12(R0)                            ;
                               7E  7C 0001D             CLRQ    -(SP)                                    ; 0227
      00000000G  00            02  FB 0001F             CALLS   #2, SYS$UNWIND                           ;
                                   04 00026             RET                                              ; 0228

; Routine Size:  39 bytes,    Routine Base:  _LIB$CODE + 0000
```

```
;  162          0229  1 END                                     ! End of module LIB$SIG_TO_RET
;  163          0230  0 ELUDOM
```

```
;                                      PSECT SUMMARY
;
;       Name                        Bytes                          Attributes
;
;  _LIB$CODE                              39  NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)
```

```
;                               Library Statistics
;
;                                               -------- Symbols --------       Pages       Processing
;       File                                    Total    Loaded   Percent       Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1             9776         6         0          581       00:00.7
;  _$255$DUA28:[LIBRTL.OBJ]RTLLIB.L32;1            36         0         0            8       00:00.0
```

```
;                                      COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:LIBSIGRET/OBJ=OBJ$:LIBSIGRET MSRC$:LIBSIGRET/UPDATE=(ENH$:LIBSIGRET
;       )

; Size:          39 code + 0 data bytes
; Run Time:         00:02.2
; Elapsed Time:     00:14.6
; Lines/CPU Min:     6160
; Lexemes/CPU-Min: 10982
; Memory Used:  34 pages
; Compilation Complete
```

0209 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LIBPOLYG
LIS

LIBREMQHI
LIS

LIBRENAME
LIS

LIBSCANC
LIS

LIBSIGSTO
LIS

LIBROOBJ
LIS

LIBRUNPRO
LIS

LIBSIGNAL
LIS

LIBPUTOUT
LIS

LIBREMQTI
LIS

LIBSIGRET
LIS

LIBSIMTRA
LIS

LIBSCOPY
LIS

LIBPOLYH
LIS

LIBREVERT
LIS