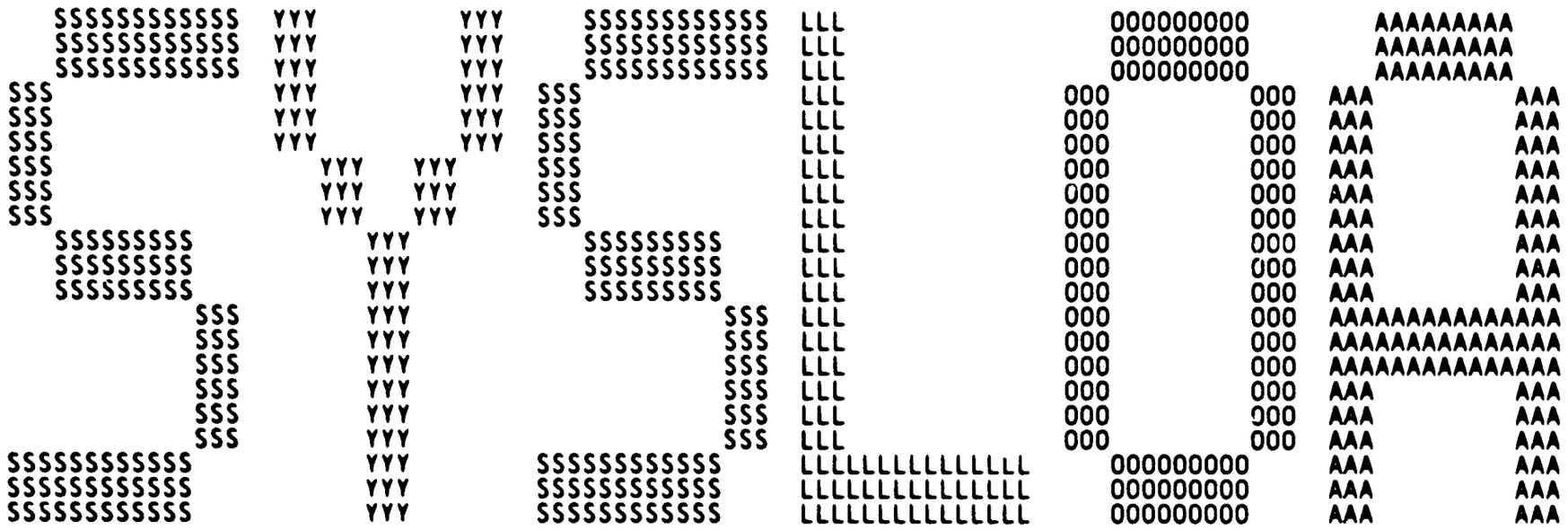
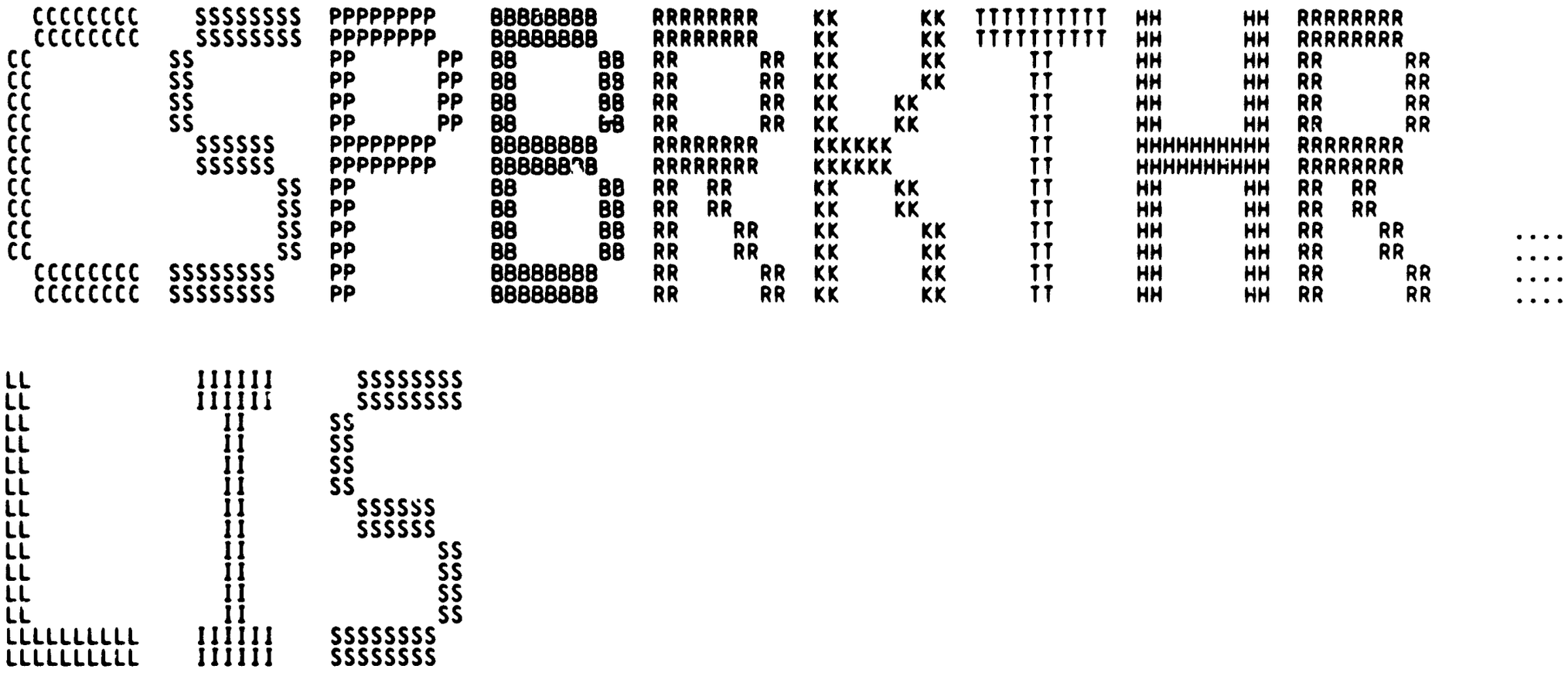

SSSSSSSSSSS YYY YYY SSSSSSSSSSS LLL 000000000 AAAAAAAAA
SSSSSSSSSSS YYY YYY SSSSSSSSSSS LLL 000000000 AAAAAAAAA
SSSSSSSSSSS YYY YYY SSSSSSSSSSS LLL 000000000 AAAAAAAAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSS YYY YYY SSS LLL 000 000 AAA AAA
SSSSSSSSS YYY SSSSSSSSS LLL 000 000 AAA AAA
SSSSSSSSS YYY SSSSSSSSS LLL 000 000 AAA AAA
SSSSSSSSS YYY SSSSSSSSS LLL 000 000 AAA AAA
SSS YYY SSS LLL 000 000 AAAAAAAAAAAAAAA
SSS YYY SSS LLL 000 000 AAAAAAAAAAAAAAA
SSS YYY SSS LLL 000 000 AAAAAAAAAAAAAAA
SSS YYY SSS LLL 000 000 AAA AAA
SSS YYY SSS LLL 000 000 AAA AAA
SSS YYY SSS LLL 000 000 AAA AAA
SSSSSSSSSSS YYY SSSSSSSSSSS LLLLLLLLLLLLLLL 000000000 AAA AAA
SSSSSSSSSSS YYY SSSSSSSSSSS LLLLLLLLLLLLLLL 000000000 AAA AAA
SSSSSSSSSSS YYY SSSSSSSSSSS LLLLLLLLLLLLLLL 000000000 AAA AAA

FILEID**CSPBRKTHR



```
0000     1          .TITLE  CSPBRKTHRU
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  ;++
0000    28  ;
0000    29  ; FACILITY:     VMS Cluster Server Process
0000    30  ;
0000    31  ; ABSTRACT:     Client support in CSP process for $BRKTHRU.
0000    32  ;
0000    33  ; AUTHOR:       Jake VanNoy
0000    34  ;
0000    35  ; DATE:         22-APR-1983
0000    36  ;
0000    37  ; REVISION HISTORY:
0000    38  ;
0000    39  ;       V03-003 ADE0001         Alan D. Eldridge        18-Jan-1984
0000    40  ;               Make call to $BRKTRHU asynchronous
0000    41  ;
0000    42  ;       V03-002 JLV0298         Jake VanNoy             28-JUL-1983
0000    43  ;               Add $BRKDEF.
0000    44  ;
0000    45  ;       V03-001 JLV0285         Jake VanNoy             28-JUL-1983
0000    46  ;               Make word values longwords before passing them to $BRKTHRU.
0000    47  ;
0000    48  ;--
```

```
0000     50          $BRKDEF                 ; Breakthru symbols
0000     51          $CSDDEF                 ; CSP protocol offsets
0000     52
```

```
                                0000     54
                                0000     55 ;
                                0000     56 ; R2 -> CSD from source
                                0000     57 ;
                                0000     58
                                0000     59 CSP$BRKTHRU::                        ; Breakthru write routine
                                0000     60
                    01FC 8F     BB   0000     61          PUSHR   #^M<R2,R3,R4,R5,R6,R7,R8>
                                0004     62
                       57 16 A2     D0   0004     63          MOVL    CSD$L_SENDOFF(R2),R7    ; send data address
                                0008     64
                       58 5E        D0   0008     65          MOVL    SP,R8                   ; Save SP
                                000B     66
                00000800 8F     CA   000B     67          BICL    #BRK$M_CLUSTER,-
                      04 A7          0011     68                  CSD$L_BRK_FLAGS(R7)     ; Set no more cluster sends
                                0013     69
                       53 7E    7E   0013     70          MOVAQ   -(SP),R3                        ; allocate descriptor
                   63 1E A7     3C   0016     71          MOVZWL  CSD$W_BRK_MSGLEN(R7),(R3)       ; message length
                04 A3 20 A7     9E   001A     72          MOVAB   CSD$T_BRK_MSGBUF(R7),4(R3)      ; and address
                                001F     73
                       54 7E    7E   001F     74          MOVAQ   -(SP),R4                        ; allocate descriptor
                   64 08 A7     9A   0022     75          MOVZBL  CSD$T_BRK_SENDTO(R7),(R4)       ; message length
                04 A4 09 A7     9E   0026     76          MOVAB   CSD$T_BRK_SENDTO+1(R7),4(R4)    ; and address
                                002B     77
                   55 18 A7     3C   002B     78          MOVZWL  CSD$W_BRK_SNDTYP(R7),R5         ; Send type
                   56 1A A7     3C   002F     79          MOVZWL  CSD$W_BRK_REQID(R7),R6          ; Requestor ID
                   52 1C A7     3C   0033     80          MOVZWL  CSD$W_BRK_TIMOUT(R7),R2         ; time out value
                                0037     81
                                0037     82          $BRKTHRU_S -
                                0037     83                  MSGBUF = (R3),-
                                0037     84                  FLAGS  = CSD$L_BRK_FLAGS(R7),-
                                0037     85                  CARCON = CSD$L_BRK_CARCON(R7),-
                                0037     86                  TIMOUT = R2,-
                                0037     87                  REQID  = R6,-
                                0037     88                  SENDTO = (R4),-
                                0037     89                  SNDTYP = R5,-
                                0037     90                  ASTADR = CSP$$RESUME,-
                                0037     91                  ASTPRM = CSP$GL_CURCTX
                                005D     92
                       5E 58    D0   005D     93          MOVL    R8,SP                           ; Restore stack
                    01FC 8F     BA   0060     94          POPR    #^M<R2,R3,R4,R5,R6,R7,R8>       ; Restore regs
                      07 50     E9   0064     95          BLBC    R0,100$                         ; If LBC, didn't queue request
00000000'EF           00        FB   0067     96          CALLS   #0,CSP$$WAIT                    ; Else, wait for AST
                                006E     97                                                      ;
50      00000000'8F             D0   006E     98 100$:    MOVL    #SS$_NORMAL,R0                  ; Return success
                                05   0075     99          RSB                                     ; Done
                                0076    100
                                0076    101 .END
```

```
$$T1                            = 00000000
BRK$M_CLUSTER                   = 00000800
CSD$L_BRK_CARCON                = 00000000
CSD$L_BRK_FLAGS                 = 00000004
CSD$L_SENDOFF                   = 00000016
CSD$T_BRK_MSGBUF                = 00000020
CSD$T_BRK_SENDTO                = 00000008
CSD$W_BRK_MSGLEN                = 0000001E
CSD$W_BRK_REQID                 = 0000001A
CSD$W_BRK_SNDTYP                = 00000018
CSD$W_BRK_TIMOUT                = 0000001C
CSP$$RESUME                       ********    X   01
CSP$$WAIT                         ********    X   01
CSP$BRKTHRU                       00000000 RG     01
CSP$GL_CURCTX                     ********    X   01
SS$_NORMAL                        ********    X   01
SYS$BRKTHRU                       ********   GX   01
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
. BLANK .	00000076 (118.)	01 (1.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
ABS	00000000 (0.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.05	00:00:00.61
Command processing	112	00:00:00.42	00:00:04.84
Pass 1	151	00:00:01.43	00:00:06.94
Symbol table sort	0	00:00:00.09	00:00:00.09
Pass 2	35	00:00:00.32	00:00:01.22
Symbol table output	4	00:00:00.02	00:00:00.02
Psect synopsis output	0	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	334	00:00:02.35	00:00:13.74

The working set limit was 1200 pages.
8791 bytes (18 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 135 non-local and 1 local symbols.
101 source lines were read in Pass 1, producing 12 object records in Pass 2.
14 pages of virtual memory were used to define 13 macros.

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

Macro library name                     Macros defined
------------------                     --------------
_$255$DUA28:[SYSLOA.OBJ]CLUSTER.MLB;1         1
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                0
_$255$DUA28:[SYSLIB]STARLET.MLB;2             9
TOTALS (all libraries)                       10

251 GETS were required to define 10 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:CSPBRKTHR/OBJ=OBJ$:CSPBRKTHR MSRC$:CSPBRKTHR/UPDATE=(ENH$:CSPBRKTHR)+EXECML$/LIB+LIB$:CLUSTER/LIB
```

0393 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

CSPCALL
LIS

CSPCJFMAS
LIS

CSP
LIS

CSPBRKTHR
LIS

CONUTIL
LIS

CSPCALLACT
LIS

CONSUBS
LIS