```
DDDDDDDDDDD     EEEEEEEEEEEEEEE  LLL             TTTTTTTTTTTTTTT     AAAAAAAAA
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  LLL             TTTTTTTTTTTTTTT     AAAAAAAAA
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  LLL             TTTTTTTTTTTTTTT     AAAAAAAAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEEEEEEEEEE      LLL                   TTT          AAA       AAA
DDD         DDD EEEEEEEEEEE      LLL                   TTT          AAA       AAA
DDD         DDD EEEEEEEEEEE      LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAAAAAAAAAAAAAA
DDD         DDD EEE              LLL                   TTT          AAAAAAAAAAAAAAA
DDD         DDD EEE              LLL                   TTT          AAAAAAAAAAAAAAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDD         DDD EEE              LLL                   TTT          AAA       AAA
DDDDDDDDDDD     EEEEEEEEEEEEEEE  LLLLLLLLLLLLLLLL      TTT          AAA       AAA
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  LLLLLLLLLLLLLLLL      TTT          AAA       AAA
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  LLLLLLLLLLLLLLLL      TTT          AAA       AAA
```

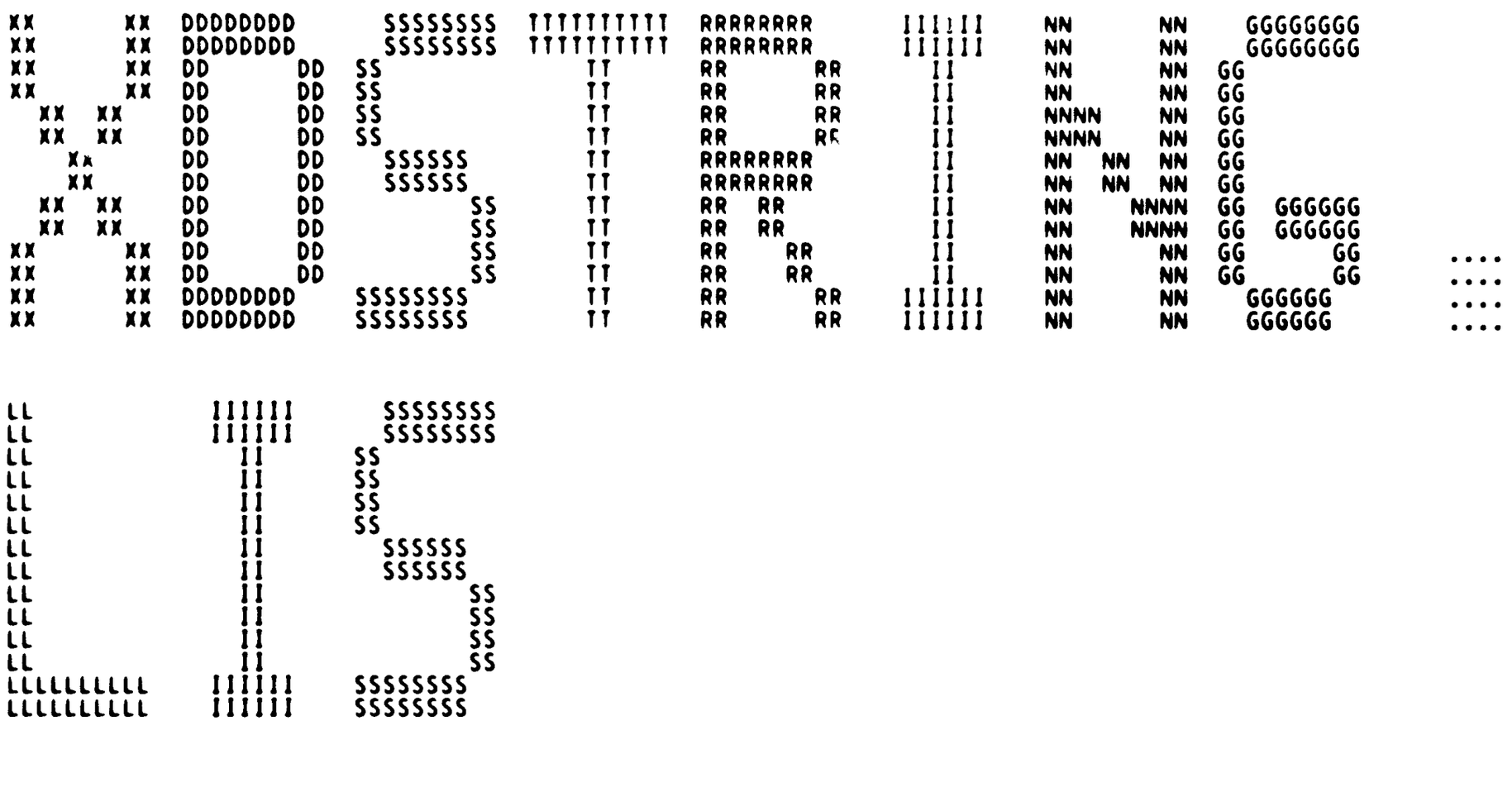
XX XX DDDDDDDD SSSSSSSS TTTTTTTTTT RRRRRRRR IIIIII NN NN GGGGGGGG
XX XX DDDDDDDD SSSSSSSS TTTTTTTTTT RRRRRRRR IIIIII NN NN GGGGGGGG
XX XX DD DD SS TT RR RR II NN NN GG
XX XX DD DD SS TT RR RR II NN NN GG
XX XX DD DD SS TT RR RR II NNNN NN GG
XX XX DD DD SS TT RR RR II NNNN NN GG
XX DD DD SSSSSS TT RRRRRRRR II NN NN NN GG
XX DD DD SSSSSS TT RRRRRRRR II NN NN NN GG
XX XX DD DD SS TT RR RR II NN NNNN GG GGGGG
XX XX DD DD SS TT RR RR II NN NNNN GG GGGGG
XX XX DD DD SS TT RR RR II NN NN GG GG
XX XX DD DD SS TT RR RR II NN NN GG GG
XX XX DDDDDDDD SSSSSSSS TT RR RR IIIIII NN NN GGGGGG
XX XX DDDDDDDD SSSSSSSS TT RR RR IIIIII NN NN GGGGGG
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

Table of contents

```
0000     1          .TITLE  XDSTRING       XDELTA Stored Command Strings
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  ;  Modified by:
0000    28  ;
0000    29  ;       V02-003 LJK0030         Lawrence J. Kenah       10-Jun-1981
0000    30  ;               Add second set of stored text strings to accomodate the
0000    31  ;               longword PFN arrays that result when more than 32 Mbytes
0000    32  ;               of physical memory is present on the system.
0000    33  ;
```

```
                     0000     35          .SUBTITLE       DECLARATIONS
                     0000     36
                     0000     37 ;
                     0000     38 ;  Equated Symbols:
                     0000     39 ;
0000000D             0000     40          CR=^O15
0000000A             0000     41          LF=^O12
                     0000     42
                     0000     43 ;
                     0000     44 ;  Own Storage:
                     0000     45 ;
                00000000     46          .PSECT  Z$DEBUGXDSTR,BYTE
                     0000     47
```

```
                                                0000     49          .SUBTITLE       Stored commands for word PFN arrays
                                                0000     50 ;
                                                0000     51 ;  This first set of stored commands assumes that the PFN data base link
                                                0000     52 ;  arrays contain word elements. Ths initial contents of both XE and XF
                                                0000     53 ;  cause this initialization sequence to take place.
                                                0000     54 ;
                                                0000     55 XDS$GT_WORD_PFN::
                                     0D 4C 5B 0D 0000     56          .ASCII  <CR>^[L^<CR>
                        0D 58 3B 36 2C 51 2F 36 58 0004     57          .ASCII  ^X6/Q,6;X^<CR>          ;SET X REGISTERS 6 THROUGH D
                        0D 58 3B 37 2C 51 2F 37 58 000D     58          .ASCII  ^X7/Q,7;X^<CR>
                        0D 58 3B 38 2C 51 2F 38 58 0016     59          .ASCII  ^X8/Q,8;X^<CR>
                        0D 58 3B 39 2C 51 2F 39 58 001F     60          .ASCII  ^X9/Q,9;X^<CR>
                        0D 58 3B 41 2C 51 2F 41 58 0028     61          .ASCII  ^XA/Q,A;X^<CR>
                  0D 4C 5B 22 57 5B 30 37 2B 45 58 0031     62          .ASCII  ^XE+70[W''[L^<CR>       ;CR, LF
                        0D 58 3B 42 2C 51 2F 42 58 003C     63          .ASCII  ^XB/Q,B;X^<CR>
                        0D 58 3B 43 2C 51 2F 43 58 0045     64          .ASCII  ^XC/Q,C;X^<CR>
                        0D 58 3B 44 2C 51 2F 44 58 004E     65          .ASCII  ^XD/Q,D;X^<CR>
                                                0057     66 ;
                                                0057     67 ;  See ASSUME macros at bottom of page for meaning of hard coded constants
                                                0057     68 ;
0D 58 3B 45 2C 46 38 2B 51 3D 45 58 0057     69          .ASCII  ^XE=Q+8F,E;X^<CR>       ;XE PRINTS PFN DATA BASE FOR PFN IN X0
0D 58 3B 46 2C 41 37 2B 51 3D 46 58 0063     70          .ASCII  ^XF=Q+7A,F;X^<CR>       ;XF PRINTS PFN DATA BASE FOR PFN IN R0
                                 00 006F     71          .BYTE   0                       ;TERMINATOR FOR INITIAL LOAD OF XE AND XF
                                                0070     72
                                                0070     73 XDS$CRLFMSG:
                                    0A 0D       0070     74          .BYTE   CR,LF
                                                0072     75 XDS$PFNMSG:
                     41 54 41 44 4E 46 50 20    0072     76          .ASCII  ^ PFNDATA^
                                                007A     77
                                                007A     78 XDS$PFNR0:
                                       0D       007A     79          .BYTE   CR
                  0D 4C 5B 22 57 5B 41 2D 46 58 007B     80          .ASCII  ^XF-A[W''[L^<CR>        ;CRLF
                  58 3B 30 2C 51 2F 57 5B 30 52 0085     81          .ASCII  ^R0[W/Q,0^ <^X3B> ^X^   ;DISPLAY PFN, SET X0
                                                008F     82 XDS$PFNX0:
                                       0D       008F     83          .BYTE   CR
4C 5B 32 2B 2E 22 57 5B 41 2D 46 58 0090     84          .ASCII  ^XF-A[W".+2[L".+4''^ ;CRLF "PFN DATA"
                        22 34 2B 2E 22 009C
                                       0D       00A1     85          .BYTE   CR
                        3D 30 58 57 5B 00A2     86          .ASCII  ^[WX0=^                 ;DISPLAY PFN
                     2F 4? 5B 43 58 2B 30 58 00A7     87          .ASCII  ^X0+XC[B/^              ;DISPLAY STATE
                     2F 4? 5B 44 58 2B 30 58 00AF     88          .ASCII  ^X0+XD[B/^              ;DISPLAY TYPE
                  2F 57 5B 39 58 2B 32 2A 30 58 00B7     89          .ASCII  ^X0+2+X9[W/^            ;DISPLAY REFCOUNT
                     2F 42 58 2B 32 2A 30 58 00C1     90          .ASCII  ^X0+2+XB/^              ;DISPLAY BLINK/WSLX
                     2F 41 58 2B 32 2A 30 58 00C9     91          .ASCII  ^X0+2+XA/^              ;DISPLAY FLINK/SHRCNT
                  2F 4C 5B 37 58 2B 34 2A 30 58 00D1     92          .ASCII  ^X0+4+X7[L/^            ;DISPLAY PTE POINTER
                     2F 38 58 2B 34 2A 30 58 00DB     93          .ASCII  ^X0+4+X8/^              ;DISPLAY BACKUP ADDRESS
                  2F 57 5B 36 58 2B 32 2A 30 58 00E3     94          .ASCII  ^X0+2+X6[W/^            ;DISPLAY SWAP VBN
                  0D 4C 5B 22 57 5B 41 2D 46 58 00ED     95          .ASCII  ^XF-A[W''[L^<CR> ;FINAL CRLF, SET LONGWORD DISPLAY
                                 00 00F7     96          .BYTE   0
                                                00F8     97
                                                00F8     98 ;
                                                00F8     99 ;  The difference between the initial and working contents of both XE and
                                                00F8    100 ;  XF must be hard coded into ASCII strings. The ASSUME macros here make
                                                00F8    101 ;  sure that these strings are prpoerly initialized in the event that text
                                                00F8    102 ;  is added to cause these differences to change. A third assumption, the
                                                00F8    103 ;  location of the CRLF string, is also checked with the ASSUME macro.
                                                00F8    104 ;
```

```
00F8   105          ASSUME <XDS$PFNXO - XDS$GT_WORD_PFN> EQ <^X8F> ; Stored string for XE
00F8   106          ASSUME <XDS$PFNRO - XDS$GT_WORD_PFN> EQ <^X7A> ; Stored string for XF
00F8   107
00F8   108          ASSUME <XDS$CRLFMSG - XDS$GT_WORD_PFN> EQ <^X70>
00F8   109          ASSUME <XDS$PFNRO - XDS$CRLFMSG> EQ <^X0A>
```

```
                                00F8   111          .SUBTITLE       Stored commands for longword PFN arrays
                                00F8   112 ;
                                00F8   113 ;  This second set of stored commands is identical to the first except for
                                00F8   114 ;  those displays that reflect longword (instead of word) PFN sizes. All
                                00F8   115 ;  characters that are different are reflected in the comments. INIT detects
                                00F8   116 ;  the longword PFN arrays and alters the initial contents of XE and XF to
                                00F8   117 ;  this alternate set of stored commands.
                                00F8   118 ;
                                00F8   119 XDS$GT_LONG_PFN::
                          0D 4C 5B 0D   00F8   120          .ASCII  <CR>^[L^<CR>
                0D 58 3B 36 2C 51 2F 36 58   00FC   121          .ASCII  ^X6/Q,6;X^<CR>          ;SET X REGISTERS 6 THROUGH D
                0D 58 3B 37 2C 51 2F 37 58   0105   122          .ASCII  ^X7/Q,7;X^<CR>
                0D 58 3B 38 2C 51 2F 38 58   010E   123          .ASCII  ^X8/Q,8;X^<CR>
                0D 58 3B 39 2C 51 2F 39 58   0117   124          .ASCII  ^X9/Q,9;X^<CR>
                0D 58 3B 41 2C 51 2F 41 58   0120   125          .ASCII  ^XA/Q,A;X^<CR>
          0D 4C 5B 22 57 5B 30 37 2B 45 58   0129   126          .ASCII  ^XE+70[W"[L^<CR>        ;CR, LF
                0D 58 3B 42 2C 51 2F 42 58   0134   127          .ASCII  ^XB/Q,B;X^<CR>
                0D 58 3B 43 2C 51 2F 43 58   013D   128          .ASCII  ^XC/Q,C;X^<CR>
                0D 58 3B 44 2C 51 2F 44 58   0146   129          .ASCII  ^XD/Q,D;X^<CR>
0D 58 3B 45 2C 46 38 2B 51 3D 45 58   014F   130          .ASCII  ^XE=Q+8F,E;X^<CR>       ;XE PRINTS PFN DATA BASE FOR PFN IN X0
0D 58 3B 46 2C 41 37 2B 51 3D 46 58   015B   131          .ASCII  ^XF=Q+7A,F;X^<CR>       ;XF PRINTS PFN DATA BASE FOR PFN IN R0
                                00 0167   132          .BYTE   0                       ;TERMINATOR FOR INITIAL LOAD OF XE AND XF
                                0168   133
                                0168   134 XDS$CRLFMSG_LONG:
                             0A 0D   0168   135          .BYTE   CR,LF
                                016A   136 XDS$PFNMSG_LONG:
                41 54 41 44 4E 46 50 20   016A   137          .ASCII  " PFNDATA"
                                0172   138
                                0172   139 XDS$PFNR0_LONG:
                                0D 0172   140          .BYTE   CR
          0D 4C 5B 22 57 5B 41 2D 46 58   0173   141          .ASCII  ^XF-A[W"[L^<CR>         ;CRLF
          5B 3B 30 2C 51 2F 4C 5B 30 52   017D   142          .ASCII  ^R0[L/Q,0^ <^X3B> ^X^   ;DISPLAY PFN, SET X0 (* [L *)
                                0187   143 XDS$PFNX0_LONG:
                                0D 0187   144          .BYTE   CR
4C 5B 32 2B 2E 22 57 5B 41 2D 46 58   0188   145          .ASCII  ^XF-A[W".+2[L".+4"^ ;CRLF "PFN DATA"
                   22 34 2B 2E 22   0194
                                0D 0199   146          .BYTE   CR
                   3D 30 58 4C 5B   019A   147          .ASCII  ^[LX0=^                 ;DISPLAY PFN (* [L *)
             2F 42 5B 43 58 2B 30 58   019F   148          .ASCII  ^X0+XC[B/^              ;DISPLAY STATE
             2F 42 5B 44 58 2B 30 58   01A7   149          .ASCII  ^X0+XD[B/^              ;DISPLAY TYPE
          2F 57 5B 39 58 2B 32 2A 30 58   01AF   150          .ASCII  ^X0*2+X9[W/^            ;DISPLAY REFCOUNT
          2F 4C 5B 42 58 2B 34 2A 30 58   01B9   151          .ASCII  ^X0*4+XB[L/^            ;DISPLAY BLINK/WSLX (* times 4 , [L added *)
             2F 41 58 2B 34 2A 30 58   01C3   152          .ASCII  ^X0*4+XA/^              ;DISPLAY FLINK/SHRCNT (* times 4 *)
             2F 37 58 2B 34 2A 30 58   01CB   153          .ASCII  ^X0*4+X7/^              ;DISPLAY PTE POINTER (* [L eliminated *)
             2F 38 58 2B 34 2A 30 58   01D3   154          .ASCII  ^X0*4+X8/^              ;DISPLAY BACKUP ADDRESS
          2F 57 5B 36 58 2B 32 2A 30 58   01DB   155          .ASCII  ^X0*2+X6[W/^            ;DISPLAY SWAP VBN
          0D 4C 5B 22 57 5B 41 2D 46 58   01E5   156          .ASCII  ^XF-A[W"[L^<CR> ;FINAL CRLF, SET LONGWORD DISPLAY
                                00 01EF   157          .BYTE   0
                                01F0   158
                                01F0   159 ;
                                01F0   160 ;  The difference between the initial and working contents of both XE and
                                01F0   161 ;  XF must be hard coded into ASCII strings. The ASSUME macros here make
                                01F0   162 ;  sure that these strings are prpoerly initialized in the event that text
                                01F0   163 ;  is added to cause these differences to change. A third assumption, the
                                01F0   164 ;  location of the CRLF string, is also checked with the assume macro.
                                01F0   165 ;
                                01F0   166          ASSUME <XDS$PFNX0_LONG - XDS$GT_LONG_PFN> EQ <^X8F> ; Stored string for XE
```

```
01F0   167          ASSUME <XDS$PFNRO_LONG - XDS$GT_LONG_PFN> EQ <^X7A> ; Stored string for XF
01F0   168
01F0   169          ASSUME <XDS$CRLFMSG_LONG - XDS$GT_LONG_PFN> EQ <^X70>
01F0   170          ASSUME <XDS$PFNRO_LONG - XDS$CRLFMSG_LONG> EQ <^X0A>
01F0   171
01F0   172          .END
```

```
CR                              = 0000000D
LF                              = 0000000A
XDS$CRLFMSG                       00000070 R      01
XDS$CRLFMSG_LONG                  00000168 R      01
XDS$GT_LONG_PFN                   000000F8 RG     01
XDS$GT_WORD_PFN                   00000000 RG     01
XDS$PFNMSG                        00000072 R      01
XDS$PFNMSG_LONG                   0000016A R      01
XDS$PFNRO                         0000007A R      01
XDS$PFNRO_LONG                    00000172 R      01
XDS$PFNXO                         0000008F R      01
XDS$PFNXO_LONG                    00000187 R      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
Z$DEBUGXDSTR	000001F0 (496.)	01 (1.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.03	00:00:00.47
Command processing	130	00:00:00.34	00:00:01.89
Pass 1	110	00:00:00.53	00:00:04.52
Symbol table sort	0	00:00:00.00	00:00:00.01
Pass 2	48	00:00:00.28	00:00:01.12
Symbol table output	3	00:00:00.01	00:00:00.01
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	330	00:00:01.21	00:00:08.04

```
The working set limit was 1050 pages.
3379 bytes (7 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 12 non-local and 0 local symbols.
172 source lines were read in Pass 1, producing 12 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.
```

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYS.OBJ]LIB.MLB;1	0
_255DUA28:[SYSLIB]STARLET.MLB;2	1
TOTALS (all libraries)	1

```
6 GETS were required to define 1 macros.

There were no errors, warnings or information messages.
```

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

DIFDEF
MDL

DIF

DIFF
MAP

XDSTRING
LIS

DIFPRE
REQ

DIFGETCMD
LIS

DIFHEXOCT
LIS

DATA
LIS

XDELTA
LIS