



```

UU      UU  PPPPPPP  CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AAAAAA  TTTTTTTTTT
UU      UU  PPPPPPP  CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AAAAAA  TTTTTTTTTT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UU      UU  PPPPPPP  CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AAAAAA  TTTTTTTTTT
UU      UU  PPPPPPP  CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AAAAAA  TTTTTTTTTT
UU      UU  PP        PP        CC        AAAAAAAAAA  SS        EE        DD        DD  AAAAAAAAAA  TT
UU      UU  PP        PP        CC        AAAAAAAAAA  SS        EE        DD        DD  AAAAAAAAAA  TT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UU      UU  PP        PP        CC        AA        AA  SS        EE        DD        DD  AA        AA  TT
UUUUUUUU  PP        CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AA        AA  TT
UUUUUUUU  PP        CCCCCCCC  AAAAAA  SSSSSSSS  EEEEEEEEE  DDDDDDDD  AA        AA  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
LLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLL  IIIIII  SSSSSSSS

```

```

0000 1 .TITLE UPCASEDAT ; MISCELLANEOUS EXECUTIVE TABLES
0000 2 .IDENT 'V04-000'
0000 3
0000 4 *****
0000 5
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *****
0000 26
0000 27 ABSTRACT: MISCELLANEOUS EXECUTIVE TABLES.
0000 28
0000 29 Ron Schaefer 3-Jan-1984
0000 30
0000 31 MODIFIED BY:
0000 32
0000 33 V03-C01 TMK0001 Todd M. Katz 27-Mar-1984
0000 34 Add EXE$LNМ SYNTAX DAT, a table utilized whenever it is
0000 35 discovered that a logical name block is to be contained within
0000 36 a logical name table directory in order to:
0000 37
0000 38 1. Verify that the syntax of each of the name stings (logical
0000 39 name string or logical name table name string) contained
0000 40 within the logical name block is acceptable.
0000 41
0000 42 2. To move each of the name strings into its appropriate
0000 43 position within the logical name block.
0000 44 :--

```

```

0000 0000 46
0000 0000 47 .PSECT Y$$PAGED_DATA,PAGE
0000 0000 48
0000 0000 49
0000 0000 50 :
0000 0000 51 : STRING UPCASE TABLE FOR THE DEC MULTINATIONAL CHARACTER SET
0000 0000 52 :
0000 0000 53 : This data table contains the upcase characters for the DEC
0000 0000 54 : multinational character set. This table is indexed by
0000 0000 55 : character code value.
0000 0000 56 :
0000 0000 57 EXE$UPCASE_DAT::
0000 0000 58
0000 0000 59 :
0000 0000 60 : 7-bit ASCII
0000 0000 61 :
0000 0000 62 :
07 06 05 04 03 02 01 00 0000 63 .BYTE 0, 1, 2, 3, 4, 5, 6, 7 : NUL - BEL
0F 0E 0D 0C 0B 0A 09 08 0008 64 .BYTE 8, 9, 10, 11, 12, 13, 14, 15 : BS - SI
17 16 15 14 13 12 11 10 0010 65 .BYTE 16, 17, 18, 19, 20, 21, 22, 23 : DLE - ETB
1F 1E 1D 1C 1B 1A 19 18 0018 66 .BYTE 24, 25, 26, 27, 28, 29, 30, 31 : CAN - US
27 26 25 24 23 22 21 20 0020 67 .BYTE 32, 33, 34, 35, 36, 37, 38, 39 : SP ! " # $ % & '
2F 2E 2D 2C 2B 2A 29 28 0028 68 .BYTE 40, 41, 42, 43, 44, 45, 46, 47 : ( ) * + , - . /
37 36 35 34 33 32 31 30 0030 69 .BYTE 48, 49, 50, 51, 52, 53, 54, 55 : 0 1 2 3 4 5 6 7
3F 3E 3D 3C 3B 3A 39 38 0038 70 .BYTE 56, 57, 58, 59, 60, 61, 62, 63 : 8 9 : ; < = > ?
47 46 45 44 43 42 41 40 0040 71 .BYTE 64, 65, 66, 67, 68, 69, 70, 71 : @ A B C D E F G
4F 4E 4D 4C 4B 4A 49 48 0048 72 .BYTE 72, 73, 74, 75, 76, 77, 78, 79 : H I J K L M N O
57 56 55 54 53 52 51 50 0050 73 .BYTE 80, 81, 82, 83, 84, 85, 86, 87 : P Q R S T U V W
5F 5E 5D 5C 5B 5A 59 58 0058 74 .BYTE 88, 89, 90, 91, 92, 93, 94, 95 : X Y Z [ \ ] ^ _
47 46 45 44 43 42 41 60 0060 75 .BYTE 96, 65, 66, 67, 68, 69, 70, 71 : ` a b c d e f g
4F 4E 4D 4C 4B 4A 49 48 0068 76 .BYTE 72, 73, 74, 75, 76, 77, 78, 79 : h i j k l m n o
57 56 55 54 53 52 51 50 0070 77 .BYTE 80, 81, 82, 83, 84, 85, 86, 87 : p q r s t u v
7F 7E 7D 7C 7B 7A 79 78 0078 78 .BYTE 88, 89, 90, 123, 124, 125, 126, 127 : x y z { | } ~ DEL
0080 79 :
0080 80 :
0080 81 : 8-bit DEC Multinational
0080 82 :
0080 83 :
87 86 85 84 83 82 81 80 0080 84 .BYTE 128, 129, 130, 131, 132, 133, 134, 135 : 8-bit controls
8F 8E 8D 8C 8B 8A 89 88 0088 85 .BYTE 136, 137, 138, 139, 140, 141, 142, 143 : 8-bit controls
97 96 95 94 93 92 91 90 0090 86 .BYTE 144, 145, 146, 147, 148, 149, 150, 151 : 8-bit controls
9F 9E 9D 9C 9B 9A 99 98 0098 87 .BYTE 152, 153, 154, 155, 156, 157, 158, 159 : 8-bit controls
A7 A6 A5 A4 A3 A2 A1 A0 00A0 88 .BYTE 160, 161, 162, 163, 164, 165, 166, 167 : 8-bit graphics
AF AE AD AC AB AA A9 A8 00A8 89 .BYTE 168, 169, 170, 171, 172, 173, 174, 175 : 8-bit graphics
B7 B6 B5 B4 B3 B2 B1 B0 00B0 90 .BYTE 176, 177, 178, 179, 180, 181, 182, 183 : 8-bit graphics
BF BE BD BC BB BA B9 B8 00B8 91 .BYTE 184, 185, 186, 187, 188, 189, 190, 191 : 8-bit graphics
C7 C6 C5 C4 C3 C2 C1 C0 00C0 92 .BYTE 192, 193, 194, 195, 196, 197, 198, 199 : 8-bit upcase alphas
CF CE CD CC CB CA C9 C8 00C8 93 .BYTE 200, 201, 202, 203, 204, 205, 206, 207 : 8-bit upcase alphas
D7 D6 D5 D4 D3 D2 D1 D0 00D0 94 .BYTE 208, 209, 210, 211, 212, 213, 214, 215 : 8-bit upcase alphas
DF DE DD DC DB DA D9 D8 00D8 95 .BYTE 216, 217, 218, 219, 220, 221, 222, 223 : 8-bit upcase alphas
C7 C6 C5 C4 C3 C2 C1 C0 00E0 96 .BYTE 192, 193, 194, 195, 196, 197, 198, 199 : 8-bit lowercase alphas
CF CE CD CC CB CA C9 C8 00E8 97 .BYTE 200, 201, 202, 203, 204, 205, 206, 207 : 8-bit lowercase alphas
D7 D6 D5 D4 D3 D2 D1 F0 00F0 98 .BYTE 240, 209, 210, 211, 212, 213, 214, 215 : 8-bit lowercase alphas
FF FE DD DC DB DA D9 D8 00F8 99 .BYTE 216, 217, 218, 219, 220, 221, 254, 255 : 8-bit lowercase alphas

```

```

0100 101
0100 102 :
0100 103 : LOGICAL NAME STRING SYNTAX TABLE.
0100 104 :
0100 105 : This table is utilized whenever it is discovered that a logical name block is
0100 106 : to be contained within a logical name table directory to verify that the
0100 107 : syntax of each of the name strings contained within the logical name block is
0100 108 : acceptable, and to move each of the name strings into its appropriate position
0100 109 : within the logical name block. This table is indexed by character code value.
0100 110 :
0100 111 :
0100 112 EXESLNM_SYNTAX_DAT::
0100 113 :
0100 114 :
0100 115 : 7-bit ASCII
0100 116 :
0100 117 :
00 00 00 00 00 00 00 00 0100 118 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : NUL - BEL
00 00 00 00 00 00 00 00 0108 119 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : BS - SI
00 00 00 00 00 00 00 00 0110 120 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : DLE - ETB
00 00 00 24 00 00 00 00 0118 121 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : CAN - US
00 00 00 00 00 00 00 00 0120 122 .BYTE 0, 0, 0, 0, 36, 0, 0, 0 : SP ! " # $ % & '
00 00 00 00 00 00 00 00 0128 123 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : ( ) * + - . /
37 36 35 34 33 32 31 30 0130 124 .BYTE 48, 49, 50, 51, 52, 53, 54, 55 : 0 1 2 3 4 5 6 7
00 00 00 00 00 00 39 38 0138 125 .BYTE 56, 57, 0, 0, 0, 0, 0, 0 : 8 9 : ; < = > ?
47 46 45 44 43 42 41 00 0140 126 .BYTE 0, 65, 66, 67, 68, 69, 70, 71 : @ A B C D E F G
4F 4E 4D 4C 4B 4A 49 48 0148 127 .BYTE 72, 73, 74, 75, 76, 77, 78, 79 : H I J K L M N O
57 56 55 54 53 52 51 50 0150 128 .BYTE 80, 81, 82, 83, 84, 85, 86, 87 : P Q R S T U V W
5F 00 00 00 00 5A 59 58 0158 129 .BYTE 88, 89, 90, 0, 0, 0, 0, 95 : X Y Z [ \ ] ^ _
47 46 45 44 43 42 41 00 0160 130 .BYTE 0, 65, 66, 67, 68, 69, 70, 71 : ` a b c d e f g
4F 4E 4D 4C 4B 4A 49 48 0168 131 .BYTE 72, 73, 74, 75, 76, 77, 78, 79 : h i j k l m n o
57 56 55 54 53 52 51 50 0170 132 .BYTE 80, 81, 82, 83, 84, 85, 86, 87 : p q r s t u v w
00 00 00 00 00 5A 59 58 0178 133 .BYTE 88, 89, 90, 0, 0, 0, 0, 0 : x y z { | } ~ DEL
0180 134 :
0180 135 :
0180 136 : 8-bit DEC Multinational
0180 137 :
0180 138 :
00 00 00 00 00 00 00 00 0180 139 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit controls
00 00 00 00 00 00 00 00 0188 140 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit controls
00 00 00 00 00 00 00 00 0190 141 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit controls
00 00 00 00 00 00 00 00 0198 142 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit controls
00 00 00 00 00 00 00 00 01A0 143 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit graphics
00 00 00 00 00 00 00 00 01A8 144 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit graphics
00 00 00 00 00 00 00 00 01B0 145 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit graphics
00 00 00 00 00 00 00 00 01B8 146 .BYTE 0, 0, 0, 0, 0, 0, 0, 0 : 8-bit graphics
C7 C6 C5 C4 C3 C2 C1 C0 01C0 147 .BYTE 192, 193, 194, 195, 196, 197, 198, 199 : 8-bit upcase alphas
CF CE CD CC CB CA C9 C8 01C8 148 .BYTE 200, 201, 202, 203, 204, 205, 206, 207 : 8-bit upcase alphas
D7 D6 D5 D4 D3 D2 D1 00 01D0 149 .BYTE 0, 209, 210, 211, 212, 213, 214, 215 : 8-bit upcase alphas
DF 00 DD DC DB DA D9 D8 01D8 150 .BYTE 216, 217, 218, 219, 220, 221, 0, 223 : 8-bit upcase alphas
C7 C6 C5 C4 C3 C2 C1 C0 01E0 151 .BYTE 192, 193, 194, 195, 196, 197, 198, 199 : 8-bit lowercase alphas
CF CE CD CC CB CA C9 C8 01E8 152 .BYTE 200, 201, 202, 203, 204, 205, 206, 207 : 8-bit lowercase alphas
D7 D6 D5 D4 D3 D2 D1 00 01F0 153 .BYTE 0, 209, 210, 211, 212, 213, 214, 215 : 8-bit lowercase alphas
00 00 DD DC DB DA D9 D8 01F8 154 .BYTE 216, 217, 218, 219, 220, 221, 0, 0 : 8-bit lowercase alphas
0200 155
0200 156 .END

```

EXE\$LNLM\_SYNTAX\_DAT 00000100 RG 01  
 EXE\$UPCASE\_DAT 00000000 RG 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			
Y\$\$PAGED_DATA	00000200 ( 512.)	01 ( 1.)	NOPIC USR	CON	REL	LCL	NOSHR	EXE	RD	WRT	NOVEC	PAGE			

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.04	00:00:01.05
Command processing	112	00:00:00.55	00:00:09.07
Pass 1	68	00:00:00.84	00:00:08.23
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	47	00:00:00.36	00:00:03.28
Symbol table output	2	00:00:00.01	00:00:00.00
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	263	00:00:01.82	00:00:21.76

The working set limit was 900 pages.  
 4222 bytes (9 pages) of virtual memory were used to buffer the intermediate code.  
 There were 10 pages of symbol table space allocated to hold 2 non-local and 0 local symbols.  
 156 source lines were read in Pass 1, producing 12 object records in Pass 2.  
 0 pages of virtual memory were used to define 0 macros.

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

Macro library name	Macros defined
\$_255\$DUA28:[SYS.OBJ]LIB.MLB;1	0
\$_255\$DUA28:[SYSLIB]STARLET.MLB;2	0
TOTALS (all libraries)	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:UPCASEDAT/OBJ=OBJ\$:UPCASEDAT MSRC\$:UPCASEDAT/UPDATE=(ENH\$:UPCASEDAT)+EXECML\$/LIB

