

[illegible]

```
RRRRRRRR      UU      UU      NN      NN      CCCCCCCC      VV      VV      TTTTTTTTTT      UU      UU      IIIIII      CCCCCCCC
RRRRRRRR      UU      UU      NN      NN      CCCCCCCC      VV      VV      TTTTTTTTTT      UU      UU      IIIIII      CCCCCCCC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NNNN      NN      CC      VV      VV      TT      UU      UU      II      CC
RRRRRRRR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RRRRRRRR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
RR      RR      UU      UU      NN      NN      CC      VV      VV      TT      UU      UU      II      CC
UUUUUUUUUU      UU      NN      NN      CCCCCCCC      VV      VV      TT      UUUUUUUUUU      IIIIII      CCCCCCCC
UUUUUUUUUU      UU      NN      NN      CCCCCCCC      VV      VV      TT      UUUUUUUUUU      IIIIII      CCCCCCCC

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
LLLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLLL      IIIIII      SSSSSSSS
```


(1) 46
(1) 275
(1) 354

DECLARATIONS
RUN_PARS - PARSE A STRING
UIC-PARSE ACTION ROUTINES

```
0000 1 .TITLE RUNCVTUIC - CONVERT UIC TO BINARY
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28 ++
0000 29 FACILITY: RUN-DETACHED CLI UTILITY
0000 30
0000 31 ABSTRACT:
0000 32
0000 33 THIS MODULE CONVERTS A UIC TO BINARY GROUP AND MEMBER CODES.
0000 34
0000 35 ENVIRONMENT: USER MODE
0000 36
0000 37 AUTHOR: C. A. MONIA , CREATION DATE: 15-AUG-1977
0000 38
0000 39 MODIFIED BY:
0000 40
0000 41 V03-001 LMP0139 L. Mark Pilant, 18-Aug-1983 12:06
0000 42 Add support for alphanumeric UICs.
0000 43
0000 44 --
```



```
0000 46      .SBTTL  DECLARATIONS
0000 47
0000 48  :
0000 49  : MACROS:
0000 50  :
0000 51  : BUILD DISPATCH TABLE FOR CASE INSTRUCTION
0000 52  :
0000 53
0000 54      .MACRO  CASE, SRC, DISPLIST, TYPE=W, BASE=#0, NMODE=S^#, ?START, ?MAX
0000 55      CASE' TYPE      SRC, BASE, NMODE' <<MAX-START>/2>-1
0000 56  START:
0000 57      .IRP    EP, <DISPLIST>
0000 58      .WORD   EP-START
0000 59      .ENDR
0000 60  MAX:
0000 61      .ENDM
0000 62
0000 63  :
0000 64  : DEFINE STATE TABLE ENTRY
0000 65  :
0000 66
0000 67      .MACRO  STATE  NAME
0000 68      .SAVE
0000 69      .PSECT  TRANTBL, RD, NOWRT
0000 70  $$T=.
0000 71      .PSECT  STATETBL, RD, NOWRT
0000 72      .IF  NB  <NAME>
0000 73  NAME:
0000 74      .ENDC
0000 75  $$$=.
0000 76      .WORD   $$T-TRSIZE
0000 77      .RESTORE
0000 78      .ENDM
0000 79
0000 80  :
0000 81  : DEFINE TRANSITION TABLE ENTRY
0000 82  :
0000 83
0000 84      .MACRO  TRAN, TOKEN, ACTION, NEXTSTATE
0000 85      .SAVE
0000 86      .PSECT  TRANTBL, RD, NOWRT
0000 87      .IF  B   TOKEN
0000 88      .BYTE   -1
0000 89      .RESTORE
0000 90      .MEXIT
0000 91      .IFF
0000 92      .BYTE   TOKEN
0000 93      .ENDC
0000 94      .IF  NB  <ACTION>
0000 95      .WORD   ACTION
0000 96      .IFF
0000 97      .WORD   0
0000 98      .ENDC
0000 99      .IF  NB  <NEXTSTATE>
0000 100     .IF  IDN  $END, <NEXTSTATE>
0000 101     .WORD   0
0000 102     .IFF
```



```
0000 103 .WORD NEXTSTATE
0000 104 .ENDC
0000 105 .IFF
0000 106 .WORD $$$+2
0000 107 .ENDC
0000 108 .RESTORE
0000 109 .ENDM
0000 110
0000 111 :
0000 112 : DEFINE DATA STRUCTURE
0000 113 :
0000 114
0000 115 .MACRO $DSECT
0000 116 .PSECT $AB$$,ABS
0000 117 .ENDM
0000 118
0000 119 :
0000 120 : EQUATED SYMBOLS:
0000 121 :
0000 122 : DEFINE TOKEN VALUES
0000 123 :
0000 124
0000000E 0000 125 CHR$K_EOL=14 : END OF LINE
0000000D 0000 126 CHR$K_PERCE=13 : PERCENT SIGN
0000 127
0000 128 :
0000 129 : DEFINE ARGUMENT LIST OFFSETS
0000 130 :
0000 131
0000 132 $DSECT
0000 133
00000000 0000 134 $$$=.
0000 135
00000004 0000 136 .BLKL 1
00000008 0004 137 STRNG: .BLKL 1 : ADDRESS OF STRING DESCRIPTOR
0000000C 0008 138 VALUE: .BLKL 1 : ADDRESS TO RECEIVE VALUE
000C 139
000C 140 :
000C 141 : DEFINE TRANSITION TABLE OFFSETS
000C 142 :
00000000 000C 143 .=$$$
0000 144
00000001 0000 145 TOKEN: .BLKB 1 : VALUE OF TOKEN
00000003 0001 146 ACTION: .BLKW 1 : ADDRESS OF ACTION ROUTINE
00000005 0003 147 NEXTSTATE: .BLKW 1 : NEXT STATE
0005 148 TRSIZE: : SIZE OF TRANSITION TABLE ENTRY
0005 149
0005 150 :
0005 151 : OWN STORAGE:
0005 152 :
0005 153 : STATE TABLE AND TRANSITION LIST FOR UIC STRING
0005 154 :
0005 155
0005 156 STATE BEGIN
0005 157 TRAN CHR$K_LBRAKT,LBRCKT : LEFT BRACKET
0005 158 TRAN
0005 159 STATE A1 :
```

```
0005 160 TRAN CHR$K_ALPHA,ALPHAN,A1 : ALPHABETIC
0005 161 TRAN CHR$K_NUMERIC,ALPHAN,A1 : NUMERIC
0005 162 TRAN CHR$K_DOT,ALPHAN,A1 : DECIMAL POINT
0005 163 TRAN CHR$K_PERCE,ALPHAN,A1 : PERCENT SIGN
0005 164 TRAN CHR$K_COMMA,COMMA : COMMA
0005 165 TRAN CHR$K_RBRAKT,,A3 :
0005 166 TRAN
0005 167 STATE A2
0005 168 TRAN CHR$K_ALPHA,ALPHAN,A2 : ALPHABETIC
0005 169 TRAN CHR$K_NUMERIC,ALPHAN,A2 : NUMERIC
0005 170 TRAN CHR$K_DOT,ALPHAN,A2 : DECIMAL POINT
0005 171 TRAN CHR$K_PERCE,ALPHAN,A2 : PERCENT SIGN
0005 172 TRAN CHR$K_RBRAKT
0005 173 TRAN
0005 174 STATE A3
0005 175 TRAN CHR$K_EOL,,SEND : TERMINATE ON END OF LINE
0005 176 TRAN
0005 177
00000000 178 .PSECT RUN_CVTUIC_DAT RD,WRT,BYTE
0000 179
00000008 0000 180 GROUP: .BLKL 2 : STRING DESCRIPTOR FOR GROUP
0000000C 0008 181 GRPNUM: .BLKL 1 : GROUP NUMBER
00000014 000C 182 OWNER: .BLKL 2 : STRING DESCRIPTOR FOR OWNER
00000018 0014 183 OWNNUM: .BLKL 1 : OWNER NUMBER
0000001C 0018 184 STRNGAD: .BLKL 1 : ADDRESS OF CURRENT STRING DESCRIPTOR
00000020 001C 185 UIC: .BLKL 1 : UIC ADDRESS
0020 186
0020 187 :+
0020 188 : RUN_CVTUIC - CONVERT UIC TO BINARY
0020 189 :
0020 190 : THIS PROCEDURE IS CALLED TO PARSE A UIC SPECIFICATION AND CONVERT THE
0020 191 : GROUP AND OWNER STRINGS TO BINARY.
0020 192 :
0020 193 : INPUTS:
0020 194 :
0020 195 : AP POINTS TO AN ARGUMENT LIST IN THE FOLLOWING FORMAT
0020 196 :
0020 197 : .LONG 2 : COUNT
0020 198 : .LONG STRNG : ADDRESS OF UIC STRING DESCRIPTOR
0020 199 : .LONG UIC : ADDRESS OF LONGWORD TO RECEIVE UIC
0020 200 :
0020 201 : OUTPUTS:
0020 202 :
0020 203 : RO LBS = UIC SUCCESSFULLY CONVERTED
0020 204 :
0020 205 : RO LBC = SYNTAX ERROR IN UIC SPECIFICATION
0020 206 :
0020 207 :-
0020 208
00000000 209 .PSECT RUN_CVTUIC NOWRT
0000 210
0000 211 RUN_CVTUIC::
0000 212 .WORD ^M<R1,R2,R3,R4,R5,R6,R7> : SAVE R1 - R7
0000 213 CLRQ W^GROUP : RESET STRING DESCRIPTOR FOR GROUP
0000 214 CLRQ W^OWNER : RESET STRING DESCRIPTOR FOR OWNER
0000 215 MOVQ @STRNG(AP),R5 : GET GET STRING DESCRIPTOR IN R5, R6
0000 216 MOVAB W^BEGIN,R7 : POINT TO START OF STATETABLES
```

55 0000'CF 7C 0002 213
000C'CF 7C 0006 214
57 04 BC 7D 000A 215
0000'CF 9E 000E 216


```

      00A1 30 0013 217      BSBW  RUN_PARS      : PARSE STRING
      1C 50 E9 0016 218      BLBC  R0,T0$      : IF LBC SYNTAX ERROR
51 000C'CF 9E 0019 219      MOVAB W^OWNER,R1    : GET OWNER STRING ADDRESS
      61 B5 001E 220      TSTW  (R1)          : ANY OWNER STRING?
      38 13 0020 221      BEQL  40$          : NO, ASSUME ALPHA UIC
      55 D4 0022 222      CLRL  R5           : YES, CLEAR WORD INDEX
      10 10 0024 223      BSBB  20$          : CONVERT GROUP TO NUMERIC
      31 50 E9 0026 224      BLBC  R0,40$      : TRY FOR ALPHA UIC CONVERSION
51 0000'CF 9E 0029 225      MOVAB W^GROUP,R1    : GET ADDRESS OF GROUP STRING
      55 D6 002E 226      INCL  R5           : INCREMENT WORD INDEX
      04 10 0030 227      BSBB  20$          : CONVERT OWNER TO NUMERIC
      25 50 E9 0032 228      BLBC  R0,40$      : TRY FOR ALPHA UIC CONVERSION
      0035 229 10$:      :
      04 0035 230      RET                    :
      0036 231      :
      0036 232      : CONVERT GROUP OR OWNER TO BINARY
      0036 233      :
      0036 234      :
      0036 235      :
      0036 236 20$:      :
7E 001C'CF 9E 0036 237      MOVAB W^UIC,-(SP)    : PUSH ADDRESS TO RECEIVE VALUE
      52 6E D0 003B 238      MOVL  (SP),R2      : COPY ADDRESS
      7E 51 D0 003E 239      MOVL  R1,-(SP)     : PUSH ADDRESS OF STRING DESCRIPTOR
00000000'EF 02 FB 0041 240      CALLS #2,LIB$CVT_OCTBIN : CONVERT TO BINARY
      0E 50 E9 0048 241      BLBC  R0,30$      : IF LBC ERROR
      50 D4 004B 242      CLRL  R0           : ASSUME VALUE TOO LARGE
      02 A2 B5 004D 243      TSTW  2(R2)       : VALUE OK?
      07 12 0050 244      BNEQ  30$          : IF NEQ NO
      08 BC45 62 B0 0052 245      MOVW  (R2),@VALUE(AP)[R5] : RETURN VALUE
      50 D6 0057 246      INCL  R0           : SET SUCCESS
      0059 247 30$:      :
      05 0059 248      RSB                    :
      005A 249      :
      005A 250      : SINCE THE OCTAL CONVERSION FAILED, TRY TO CONVERT AS AN ALPHANUMERIC UIC
      005A 251      :
      005A 252      :
      005A 253      :
      005A 254 40$:      CLRL  -(SP)          : TRY FOR GROUP TRANSLATION
      00000008'EF 7E D4 005A 254      CLRL  -(SP)          : TRY FOR GROUP TRANSLATION
      00000000'EF 9F 005C 255      PUSHAB GRPNUM      :
      00000000'EF 9F 0062 256      PUSHAB GROUP      :
      00000000'EF 03 FB 0068 257      CALLS #3,SYSS$ASCTOID :
      44 50 E9 006F 258      BLBC  R0,70$      : XFER IF IN ERROR, ALL ATTEMPTS FAILED
      0000000C'EF B5 0072 259      TSTW  OWNER      : ANY OWNER FIELD TO TRANSLATE?
      0A 12 0078 260      BNEQ  50$          : XFER IF NOT
08 BC 00000008'EF D0 007A 261      MOVL  GRPNUM,@VALUE(AP) : ELSE GROUP IS REALLY FULL UIC
      2F 11 0082 262      BRB  60$          : GO FINISH UP
      7E D4 0084 263 50$:      CLRL  -(SP)          : TRY FOR OWNER TRANSLATION
      00000014'EF 9F 0086 264      PUSHAB OWNNUM      :
      0000000C'EF 9F 008C 265      PUSHAB OWNER      :
      00000000'EF 03 FB 0092 266      CALLS #3,SYSS$ASCTOID :
      1A 50 E9 0099 267      BLBC  R0,70$      : XFER IF IN ERROR, ALL ATTEMPTS FAILED
      50 D4 009C 268      CLRL  R0           : ASSUME GROUPS DON'T MATCH
00000016'EF 0000000A'EF B1 009E 269      CMPW  GRPNUM+2,OWNNUM+2 : DO THEY REALLY?
      0B 12 00A9 270      BNEQ  70$          : XFER IF NOT, TIS IN ERROR
08 BC 00000014'EF D0 00AB 271      MOVL  OWNNUM,@VALUE(AP) : ELSE RETURN THE CONVERTED UIC
      50 01 D0 00B3 272 60$:      MOVL  #1,R0      : NOTE SUCCESS
      04 00B6 273 70$:      RET                    : RETURN TO CALLER WITH STATUS
```



```
00B7 275 .SBTTL RUN_PARS - PARSE A STRING
00B7 276
00B7 277 :+
00B7 278 : RUN_PARS - PARSE A STRING
00B7 279 :
00B7 280 : THIS ROUTINE IS ENTERED VIA A BSB OR JSB TO PERFORM TABLE DRIVEN PARSING.
00B7 281 : IT REQUIRES A SET OF STATE AND CORRESPONDING TRANSITION TABLES THAT DEFINE
00B7 282 : THE SYNTAX OF THE STRING TO BE PARSED.
00B7 283 :
00B7 284 : THE SET OF TOKENS RECOGNIZED IS LIMITED TO THE FOLLOWING:
00B7 285 :
00B7 286 :     ALPHABETIC
00B7 287 :     NUMERIC
00B7 288 :     DOT (.)
00B7 289 :     SLASH (/)
00B7 290 :     SEMICOLON (;)
00B7 291 :     LEFT BRACKET ([)
00B7 292 :     RIGHT BRACKET (])
00B7 293 :     COLON (:)
00B7 294 :     BLANK
00B7 295 :     DOLLAR SIGN ($)
00B7 296 :     UNDERSCORE (_)
00B7 297 :     PERCENT (%)
00B7 298 :     END-OF-STRING
00B7 299 :
00B7 300 :
00B7 301 : INPUTS:
00B7 302 :
00B7 303 :     R5, R6 = STRING DESCRIPTOR
00B7 304 :     R7 = ADDRESS OF STATE TABLES
00B7 305 :
00B7 306 : OUTPUTS:
00B7 307 :
00B7 308 :     R0 LBS = STRING IS SYNTACTICALLY CORRECT
00B7 309 :
00B7 310 :     R0 LBC = SYNTAX ERROR
00B7 311 :
00B7 312 :     R5 = COUNT OF REMAINING UNPROCESSED CHARACTERS
00B7 313 :     R6 = ADDRESS OF CHARACTER IN ERROR
00B7 314 :
00B7 315 : -
00B7 316 :
00B7 317 RUN_PARS::
55      B6 00B7 318 INCW    R5
56      D7 00B9 319 DECL    R6
20$:    00BB 320
57      67 3C 00BB 321 MOVZWL (R7),R7
50      0E D0 00BE 322 MOVL    #CHRSK_EOL,R0
55      55 B7 00C1 323 DECB    R5
0D      13 00C3 324 BEQL    30$
56      D6 00C5 325 INCL    R6
50      0D D0 00C7 326 MOVL    #CHRSK_PERCE,R0
25      66 91 00CA 327 CMPB    (R6),#A/Z/
03      13 00CD 328 BEQL    30$
FF2E'   30 00CF 329 BSBW    CHRSTSTCHR
00D2    330 30$:
57      05 A0 00D2 331 ADDW    #TRSIZE,R7
: POINT TO NEXT TABLE ENTRY
```

```

67  18  50  91  00D9  334  TSTB  TOKEN(R7)      : AT END OF TRANSITION TABLE?
67  50  91  00D9  334  BLSS  ILLCHR      : IF LSS YES, ILLEGAL
50  01  A7  3C  00DE  336  CMPB  R0, TOKEN(R7) : TEST INPUT
50  01  A7  3C  00DE  336  BNEQ  30$      : IF NEQ, GO TO NEXT TRANSITION
50  01  A7  3C  00DE  336  MOVZWL ACTION(R7),R0 : GET ADDRESS OF ACTION ROUTINE
50  01  A7  3C  00DE  336  BEQL  40$      : IF EQL NONE
50  01  A7  3C  00DE  336  JSB   (R0)      : CALL ACTION ROUTINE
57  03  A7  3C  00E6  340  40$: MOVZWL NEXTSTATE(R7),R7 : GET OFFSET TO NEXT STATE
50  01  CF  12  00EA  341  BNEQ  20$      : IF NEQ HAVE NEXT STATE
50  01  D0  00EC  342  MOVL  #1,R0      : SET SUCCESS
50  01  02  11  00EF  343  BRB   EXIT      : EXIT
50  01  02  11  00EF  343  OOF1  344  :
50  01  02  11  00EF  343  OOF1  345  : ILLEGAL CHARACTER
50  01  02  11  00EF  343  OOF1  346  :
50  01  02  11  00EF  343  OOF1  347  :
50  01  02  11  00EF  343  OOF1  348  :
50  01  02  11  00EF  343  OOF1  349  ILLCHR: CLRL  R0      : SET FAILURE
50  01  02  11  00EF  343  OOF1  350  EXIT:  RSB
50  01  02  11  00EF  343  OOF3  351
50  01  02  11  00EF  343  OOF3  352
```



```
.SBTTL UIC PARSE ACTION ROUTINES

00F4 354
00F4 355
00F4 356
00F4 357 : ACCUMULATE AN ALPHANUMERIC STRING
00F4 358 :
00F4 359
00F4 360 ALPHAN:
00F4 361 DOT:
0018'DF D6 00F4 362 INCL @W^STRNGAD : INCREMENT STRING COUNT
05 00F8 363 RSB
00F9 364
00F9 365 :
00F9 366 : COMMA
00F9 367 :
00F9 368
00F9 369 COMMA:
0018'CF 000C'CF 9E 00F9 370 MOVAB W^OWNER,W^STRNGAD : BEGIN OWNER STRING DESCRIPTOR
0010'CF 01 A6 9E 0100 371 MOVAB 1(R6),W^OWNER+4 : POINT TO START OF STRING
05 0106 372 RSB : RETURN
0107 373
0107 374 :
0107 375 : LEFT BRACKET
0107 376 :
0107 377
0107 378 LBRCKT:
0018'CF 0000'CF 9E 0107 379 MOVAB W^GROUP,W^STRNGAD : POINT TO GROUP STRING DESCRIPTOR
0004'CF 01 A6 9E 010E 380 MOVAB 1(R6),W^GROUP+4 : POINT TO START OF STRING
05 0114 381 RSB
0115 382
0115 383 .END
```

RUNCVTUIC
Symbol table

- CONVERT UIC TO BINARY

I 12

15-SEP-1984 23:41:49 VAX/VMS Macro V04-00 Page 9
4-SEP-1984 23:17:36 [CLIUTL.SRC]RUNCVTUIC.MAR;1 (1)

\$\$\$	= 00000000		
\$\$\$	= 00000006	R	03
\$\$T	= 0000003F	R	02
A1	00000002	R	03
A2	00000004	R	03
A3	00000006	R	03
ACTION	00000001		
ALPHAN	000000F4	R	05
BEGIN	00000000	R	03
CHRSK_ALPHA	*****	X	02
CHRSK_COMMA	*****	X	02
CHRSK_DOT	*****	X	02
CHRSK_EOL	= 0000000E		
CHRSK_LBRAKT	*****	X	02
CHRSK_NUMERIC	*****	X	02
CHRSK_PERCE	= 0000000D		
CHRSK_RBRAKT	*****	X	02
CHRSTCHR	*****	X	05
COMMA	000000F9	R	05
DOT	000000F4	R	05
EXIT	000000F3	R	05
GROUP	00000000	R	04
GRPNUM	00000008	R	04
ILLCHR	000000F1	R	05
LBRCKT	00000107	R	05
LIB\$CVT OCTBIN	*****	X	05
NEXTSTATE	00000003		
OWNER	0000000C	R	04
OWNNUM	00000014	R	04
RUN_CVTUIC	00000000	RG	05
RUN_PARS	000000B7	RG	05
STRNG	00000004		
STRNGAD	00000018	R	04
SYSSASCTOID	*****	X	05
TOKEN	00000000		
TRSIZE	00000005		
UIC	0000001C	R	04
VALUE	00000008		

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	0000000C (12.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
TRANTBL	00000045 (69.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE
STATETBL	00000008 (8.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE
RUN_CVTUIC_DAT	00000020 (32.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
RUN_CVTUIC	00000115 (277.)	05 (5.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

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

Phase	Page faults	CPU Time	Elapsed Time
-----	-----	-----	-----
Initialization	10	00:00:00.07	00:00:00.71
Command processing	83	00:00:00.91	00:00:04.77
Pass 1	97	00:00:01.56	00:00:07.06
Symbol table sort	0	00:00:00.02	00:00:00.02
Pass 2	77	00:00:00.85	00:00:03.42
Symbol table output	4	00:00:00.05	00:00:00.47
Psect synopsis output	2	00:00:00.03	00:00:00.05
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	275	00:00:03.49	00:00:16.50

The working set limit was 900 pages.

8279 bytes (17 pages) of virtual memory were used to buffer the intermediate code.

There were 10 pages of symbol table space allocated to hold 38 non-local and 10 local symbols.

383 source lines were read in Pass 1, producing 18 object records in Pass 2.

4 pages of virtual memory were used to define 4 macros.

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

Macro library name	Macros defined
-----	-----
_\$255\$DUA28:[CLIUTL.OBJ]CLIUTL.MLB;1	0
-\$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\$:RUNCVTUIC/OBJ=OBJ\$:RUNCVTUIC MSRC\$:RUNCVTUIC/UPDATE=(ENH\$:RUNCVTUIC)+EXECMLS/LIB+LIB\$:CLIUTL/LIB

0051 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

