

A large, stylized letter 'A' is formed using the characters 'S' and 'Y'. The 'S' characters are arranged in a grid-like pattern to create the left and right sides of the letter, while 'Y' characters form the central vertical stem and the horizontal crossbar. The overall shape is a bold, blocky 'A' that fills most of the page.

[illegible]

(2)	134	Allocate Device
(3)	279	Deallocate Device
(4)	371	Lock I/O Database and Probe Name Descriptor
(5)	395	Decrement Reference Count, Clear Ownership, and Call Driver


```
0000 1 .TITLE SYSDEVALC - SYSTEM SERVICES FOR DEVICE ALLOCATION
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
0000 28 D. N. CUTLER 9-SEP-76
0000 29
0000 30 MODIFIED BY:
0000 31
0000 32 V03-013 ACG0411 Andrew C. Goldstein, 22-Mar-1984 17:13
0000 33 Unlock device if result buffer probe fails
0000 34
0000 35 V03-012 ACG0399 Andrew C. Goldstein, 24-Feb-1984 18:25
0000 36 Rewrite I/O database search to handle cluster device
0000 37 allocation correctly, move device lock routines to
0000 38 IOSUBPAGD, allow devices with channels assigned to be
0000 39 deallocated, rework 'media_type' to use device name string
0000 40 as the device type name, general code cleanup.
0000 41
0000 42 V03-011 LMP0185 L. Mark Pilant, 1-Feb-1984 14:02
0000 43 Add support for device ACLs.
0000 44
0000 45 V03-010 ROW0240 Ralph O. Weber 11-OCT-1983
0000 46 Correct calling sequence to IOC$CVT_DEVNAM used to create
0000 47 physical device name to supply a R4; name-type, argument.
0000 48 Use R4=-1 which produces the same results at the previous R4
0000 49 PCB address value produced.
0000 50
0000 51 V03-009 TCM0005 Trudy C. Matthews 12-Sep-1983
0000 52 Only take out a lock on the device if the system is
0000 53 participating in a cluster.
0000 54
0000 55 V03-008 TCM0004 Trudy C. Matthews 15-Aug-1983
0000 56 Take advantage of an $ENQ optimization that allows the
0000 57 original device lock and its conversion to be collapsed into
```



```
0000 58 : one call to $ENQ.
0000 59 :
0000 60 : V03-007 TCM0003 Trudy C. Matthews 24-Jun-1983
0000 61 : Move code that constructs allocation class name from
0000 62 : EXE$LOCK_DEV to routine IOC$CVT_DEVNAM, so that it is
0000 63 : available for use by other routines. Move zeroing of
0000 64 : UCB$$_LOCKID field from EXE$DALLOC_DEV to EXE$UNLOCK_DEV.
0000 65 :
0000 66 : V03-006 TCM0002 Trudy C. Matthews 10-Jun-1983
0000 67 : Fix bug in setting up input R4 register in calls to
0000 68 : IOC$CVT_DEVNAM. Add system facility name SYS$ to lock
0000 69 : resource name. Allow caller of EXE$LOCK_DEV to specify
0000 70 : the address of a lock value block.
0000 71 :
0000 72 : V03-005 TCM0001 Trudy C. Matthews 21-Apr-1983
0000 73 : Use the lock manager to enforce cluster-wide device allocation.
0000 74 :
0000 75 : V03-004 KTA3042 Kerbey T. Altmann 20-Mar-1983
0000 76 : Fix scan of device data base in DEALL to include
0000 77 : cluster-wide devices.
0000 78 :
0000 79 : V03-003 ROW0135 Ralph O. Weber 21-OCT-1982
0000 80 : Add test for UCB$$_TEMPLATE in UCB$$_STS of device to be
0000 81 : allocated. If bit is set (indicating a template UCB), return
0000 82 : SSS$_TEMPLATEDEV, "Template device cannot be allocated" error
0000 83 : status. Also move SSS$_ACCVIO error return setup to fix broken
0000 84 : branch destinations.
0000 85 :
0000 86 : V03-002 KTA0101 Kerbey T. Altmann 07-Jun-1982
0000 87 : Add support for optional 4th parameter, MEDIA_TYPE
0000 88 :
0000 89 : V03-001 STJ0258 Steven T. Jeffreys, 12-Apr-1982
0000 90 : If the device is mounted, do not deallocate it, and
0000 91 : return SSS$_DEV_MOUNT as the status code.
0000 92 :
0000 93 : **
0000 94 :
0000 95 : System services for device allocation
0000 96 :
0000 97 : Allocate device
0000 98 : Deallocate device
0000 99 :
0000 100 : Macro library calls
0000 101 :
0000 102 :
0000 103 : $CANDEF ; define cancel reason codes
0000 104 : $DDTDEF ; define DDT offsets
0000 105 : $DEVDEF ; define device characteristics
0000 106 : $IOCDEF ; define search routine flags
0000 107 : $PCBDEF ; define PCB offsets
0000 108 : $SSDEF ; define system status values
0000 109 : $UCBDEF ; define UCB offsets
0000 110 :
0000 111 :
0000 112 : Local symbols
0000 113 :
0000 114 : Argument list offset definitions for allocate device
```

```
0000 115 ;
0000 116 ;
00000004 0000 117 DEVNAM=4 ; address of device name string descriptor
00000008 0000 118 PHYLEN=8 ; address to store length of physical name
0000000C 0000 119 PHYBUF=12 ; address of physical name buffer descriptor
00000010 0000 120 ALACMODE=16 ; access mode
00000014 0000 121 FLAGS=20 ; flags longword, containing
00000000 0000 122 DEVICE_TYPE=0 ; device name is a device type string
0000 123 ;
0000 124 ;
0000 125 ; Argument list offset definitions for deallocate device
0000 126 ;
0000 127 ;
00000004 0000 128 DEVNAM=4 ; address of device name string descriptor
00000008 0000 129 DLACMODE=8 ; access mode
0000 130 ;
0000 131 ;
00000000 132 .PSECT Y$EXEPAGED
```



```
0000 134 .SBTTL Allocate Device
0000 135
0000 136 :+
0000 137 :
0000 138 : EXESALLOC - allocate device
0000 139 :
0000 140 : This service provides the capability to reserve a device for exclusive
0000 141 : use.
0000 142 :
0000 143 : INPUTS:
0000 144 :
0000 145 :     DEVNAM(AP) = address of device name string descriptor.
0000 146 :     PHYLEN(AP) = address to store length of physical name.
0000 147 :     PHYBUF(AP) = address of physical name buffer descriptor.
0000 148 :     ALACMODE(AP) = access mode device is to be allocated to.
0000 149 :     FLAGS(AP) = flags longword (optional)
0000 150 :
0000 151 :     R4 = current process PCB address.
0000 152 :
0000 153 : OUTPUTS:
0000 154 :
0000 155 :     R0 low bit clear indicates failure to allocate device.
0000 156 :
0000 157 :     R0 = SS$ _ACCVIO - device name string, device name string
0000 158 :         descriptor, or physical name buffer descriptor
0000 159 :         cannot be read by calling access mode, or physical
0000 160 :         name buffer cannot be written by calling access
0000 161 :         mode.
0000 162 :
0000 163 :     R0 = SS$ _IVSTSFLG - invalid flag bits set
0000 164 :
0000 165 :     R0 = SS$ _IVLOGNAM - zero or greater than maximum length
0000 166 :         device name string specified.
0000 167 :
0000 168 :     R0 = SS$ _IVDEVNAM - device name string contains invalid
0000 169 :         characters or no device name string descriptor
0000 170 :         specified.
0000 171 :
0000 172 :     R0 = SS$ _NONLOCAL - specified device exists on a remote
0000 173 :         system.
0000 174 :
0000 175 :     R0 = SS$ _NOSUCHDEV - specified device does not exist on
0000 176 :         host system.
0000 177 :
0000 178 :     R0 = SS$ _NODEVAVL - no allocatable unit can be found.
0000 179 :
0000 180 :     R0 = SS$ _DEVALLOC - device already allocated to another
0000 181 :         process.
0000 182 :
0000 183 :     R0 = SS$ _DEVMOUNT - device currently mounted and cannot
0000 184 :         be allocated.
0000 185 :
0000 186 :     R0 = SS$ _DEVOFFLINE - device is marked offline or
0000 187 :         unavailable.
0000 188 :
0000 189 :     R0 = SS$ _TEMPLATEDEV - device is a template and cannot
0000 190 :         be allocated
```

```
0000 191 :
0000 192 :
0000 193 :
0000 194 :
0000 195 :
0000 196 :
0000 197 :
0000 198 :
0000 199 :
0000 200 :
0000 201 :
0000 202 :
0000 203 :
0000 204 :
0000 205 :
0000 206 :
0000 207 :
0000 208 :
0000 209 :
0000 210 :
0000 211 :
0000 212 :
0000 213 :-
0000 214 :
0000 215 :
0000 216 :
0000 217 10$: BSBW IOC$UNLOCK_DEV ; unlock the device
0003 218 15$: BRW ACCVIO ; common ACCVIO handling
0006 219 :
0006 220 :
0008 221 :
000D 222 :
0010 223 :
0012 224 :
0014 225 :
0017 226 :
0019 227 :
0020 228 :
0025 229 :
0029 230 :
002B 231 :
002D 232 :
0030 233 20$: BSBW IOC$SEARCH ; search for generic allocatable device
0033 234 : ; note all checks made by search routine
0033 235 30$: MOVL R0,R6 ; save search status
0036 236 40$: BLBC R6,90$ ; if lbc search failure
0039 237 : MOVL R1,R5 ; save device UCB address
003C 238 :
003C 239 :
003C 240 :
003C 241 :
0040 242 :
0042 243 :
0048 244 :
004B 245 :
004D 246 :
0052 247 :

56 0144 8F 3C 0006 220 .ENTRY EXES$ALLOC,^M<R2,R3,R4,R5,R6>
00FD 30 0008 221 MOVZWL #SS$_IVDEVNAM,R6 ; assume invalid device name
24 13 000D 222 BSBW LOCK ; lock I/O data base and get device name
52 7C 0010 223 BEQL 40$ ; if eql no device name specified
05 6C 91 0012 224 CLRQ R2 ; zero flags and value block
17 1F 0014 225 CMPB (AP),#FLAGS/4 ; see if flags argument is present
0017 226 BLSSU 20$ ; branch if not
0019 227 IFNORD #4,FLAGS(AP),15$ ; probe it, since it's optional
56 017C 8F 3C 0020 228 MOVZWL #SS$_IVSTSFLG,R6 ; assume invalid flags
01 14 AC D1 0025 229 CMPL FLAGS(AP),#1 ; test for valid bits
05 1F 0029 230 BLSSU 20$ ; zero - OK
79 1A 002B 231 BGTRU 90$ ; invalid bits set
52 02 88 002D 232 BISB #IOC$M_TYPE,R2 ; set device type mode
FFCD 30 0030 233 BSBW IOC$SEARCH ; search for generic allocatable device
56 50 D0 0033 234 : ; note all checks made by search routine
6D 56 E9 0036 235 30$: MOVL R0,R6 ; save search status
55 51 D0 0039 236 40$: BLBC R6,90$ ; if lbc search failure
003C 237 : MOVL R1,R5 ; save device UCB address
003C 238 :
003C 239 :
003C 240 :
51 0C AC D0 003C 241 MOVL PHYBUF(AP),R1 ; get address of name buffer descriptor
43 13 0040 242 BEQL 70$ ; if eql none
50 61 3C 0042 243 IFNORD #8,(R1),10$ ; can name buffer descriptor be read?
38 13 0048 244 MOVZWL (R1),R0 ; get length of name buffer
00FF 8F 50 B1 004B 245 BEQL 70$ ; if eql zero length buffer
04 1B 004D 246 CMPW R0,#255 ; was buffer size reasonable?
0052 247 BLEQU 50$ ; branch if ok
```



```
50 FF 8F 9A 0054 248
51 04 A1 D0 0058 249 50$: MOVZBL #255,R0 ; use smaller, but still absurd size
                                MOVL 4(R1),R1 ; get address of name buffer
                                IFNOWRT R0,(R1),10$ ; can name buffer be written?
                                MOVL PHYLEN(AP),R3 ; get address to store name length
                                BEQL 60$ ; if eql none
                                IFNOWRT #2,(R3),10$ ; can name length be written?
53 08 AC D0 0062 251 60$: MNEGL #1,R4 ; use display format device name
                                MOVL IOCS$CVT_DEVNAM ; convert device name and unit
                                MOVL CIL$GL_PCB,R4 ; restore PCB address in R4
54 00000000 EF D0 0074 256 MOVL R0,R6 ; save completion status
                                D5 007E 258 TSTL R3 ; address to store length specified?
                                03 13 0080 259 BEQL 70$ ; if eql none specified
                                63 51 B0 0082 260 MOVW R1,(R3) ; insert length of converted string
                                0085 261 ;
                                0085 262 ; Finish allocating the device. The lock, if any, was taken out by IOCS$SEARCH.
                                0085 263 ; All we have to do is mark up the UCB.
                                0085 264 ;
07 38 A5 17 E3 0085 265 70$: BB$C$ #DEV$V_ALL,UCB$L_DEVCHAR(R5),80$ ; if clr, device not allocated
56 0641 8F 3C 008A 266 MOVZWL #SS$ _DEVALRALLOC,R6 ; set device already allocated status
                                15 11 008F 267 BRB 90$
                                0091 268
50 10 AC 02 00 EF 0091 269 80$: EXTZV #0,#2,ALACMODE(AP),R0 ; get specified access mode
                                FF66' 30 0097 270 BSBW EXE$MAXACMODE ; maximize access mode
                                5F A5 50 90 009A 271 MOVW R0,UCB$B_AMOD(R5) ; set allocating access mode
                                5C A5 B6 009E 272 INCW UCB$W_REFC(R5) ; count a reference
                                2C A5 60 A4 D0 00A1 273 MOVL PCB$L_PID(R4),UCB$L_PID(R5) ; store owner PID
                                50 56 D0 00A6 274 90$: MOVL R6,R0 ; set completion status
                                FF54' 31 00A9 275 BRW IOCS$UNLOCK
                                00AC 276
                                00AC 277
                                .DISABLE LSB
```

```
00AC 279 .SBTTL Deallocate Device
00AC 280
00AC 281 :+
00AC 282 :
00AC 283 : EX$DALLOC - deallocate device
00AC 284 :
00AC 285 : This service provides the capability to relinquish exclusive use of a
00AC 286 : device.
00AC 287 :
00AC 288 : INPUTS:
00AC 289 :
00AC 290 :     DEVNAM(AP) = address of device name string descriptor. Zero
00AC 291 :     implies all.
00AC 292 :     DLACMODE(AP) = access mode for deallocation.
00AC 293 :
00AC 294 :     R4 = current process PCB address.
00AC 295 :
00AC 296 : OUTPUTS:
00AC 297 :
00AC 298 :     R0 low bit clear indicates failure to deallocate device.
00AC 299 :
00AC 300 :         R0 = SS$ ACCVIO - device name string or device name string
00AC 301 :         descriptor cannot be read by calling access mode.
00AC 302 :
00AC 303 :         R0 = SS$ DEVASSIGN - device cannot be deallocated because
00AC 304 :         process still has channels assigned.
00AC 305 :
00AC 306 :         R0 = SS$ DEVMOUNT - device cannot be deallocated because it
00AC 307 :         is still mounted.
00AC 308 :
00AC 309 :         R0 = SS$ DEVNOTALLOC - device not allocated or not allocated
00AC 310 :         to process.
00AC 311 :
00AC 312 :         R0 = SS$ IVDEVNAM - device name string contains invalid
00AC 313 :         characters.
00AC 314 :
00AC 315 :         R0 = SS$ IVLOGNAM - zero or greater than maximum length
00AC 316 :         device name string specified.
00AC 317 :
00AC 318 :         R0 = SS$ NOPRIV - calling access mode does not have privilege
00AC 319 :         to deallocate device.
00AC 320 :
00AC 321 :         R0 = SS$ NOSUCHDEV - specified device does not exist on
00AC 322 :         host system.
00AC 323 :
00AC 324 :     R0 low bit set indicates successful completion.
00AC 325 :
00AC 326 :         R0 = SS$ _NORMAL - normal completion.
00AC 327 :
00AC 328 :
00AC 329 .ENTRY EX$DALLOC, ^M<R2,R3,R4,R5,R8>
50 08 AC 02 00 013C 00AE 330 EXTZV #0,#2,DLACMODE(AP),R0 ; get deallocation access mode
FF49' 30 00B4 331 BSBW EX$MAXACMODE ; maximize access mode
55 50 D0 00B7 332 MOVL R0,R5 ; save maximized access mode
51 10 00BA 333 BSBB LOCK ; lock I/O data base and search for device
1E 13 00BC 334 BEQL 20$ ; if eql no device name specified
FF3F' 30 00BE 335 BSBW IOC$SEARCHDEV ; search for physical device
```



```
60 A4 47 50 E9 00C1 336 BLBC R0,60$ ; if lbc search failure
      2C A1 D1 00C4 337 CMPL UCBSL_PID(R1),PCBSL_PID(R4) ; device allocated to current process?
      3B 12 00C9 338 BNEQ 50$ ; if neq no
5F A1 55 91 00CB 339 CMPB R5,UCBSB_AMOD(R1) ; can specified access mode deallocate device
      02 15 00CF 340 BLEQ 10$ ; if !eq yes
      52 11 00D1 341 BRB NOPRIV ; if gtr no
      00D3 342
2E 38 A1 17 E1 00D3 343 10$: BBC #DEV$V_ALL,UCBSL_DEVCHAR(R1),50$ ; if clr, device not allocated
      56 10 00D8 344 BSBB DECF ; decrement reference count and call driver
      51 11 00DA 345 BRB UNLOCK ;
      00DC 346
      00DC 347 ;
      00DC 348 ; Null device name - deallocate all devices
      00DC 349
      7E 5A 7D 00DC 350 20$: MOVQ R10,-(SP) ; save some registers
      5A 7C 00DF 351 CLRQ R10 ; initial conditions
      FF1C 30 00E1 352 30$: BSBW IOC$SCAN_IODB ; scan the IO database
      06 50 E8 00E4 353 BLBS R0,40$ ; got a good UCB
      5A 8E 7D 00E7 354 MOVQ (SP)+,R10 ; end of the line, restore registers
      003D 31 00EA 355 BRW NORMAL ; and get out
      00ED 356
      2C AA D1 00ED 357 40$: CMPL UCBSL_PID(R10),-
      60 A4 12 00F0 358 PCBSL_PID(R4) ; device allocated to current process?
      ED 91 00F2 359 BNEQ 30$ ; if neq no
5F AA 55 91 00F4 360 CMPB R5,UCBSB_AMOD(R10) ; can access mode deallocate device?
      E7 14 00F8 361 BGTR 30$ ; if gtr no
      17 E1 00FA 362 BBC #DEV$V_ALL,-
      E2 38 AA 00FC 363 UCBSL_DEVCHAR(R10),30$ ; if clr, device not allocated
      51 5A D0 00FF 364 MOVL R10,RT ; transfer UCB pointer
      2C 10 0102 365 BSBB DECF ; decrement reference count and call driver
      DB 11 0104 366 BRB 30$ ;
      0106 367
50 0858 8F 3C 0106 368 50$: MOVZWL #SS$ DEVNOTALLOC,R0 ; set device not allocated to process
      20 11 010B 369 60$: BRB UNLOCK ;
```

```
010D 371 .SBTTL Lock I/O Database and Probe Name Descriptor
010D 372
010D 373
010D 374 : Subroutine to lock I/O data base, search for device, and check for mailbox.
010D 375 :
010D 376
010D 377 LOCK: : lock I/O data base and search for device
010D 378 JSB SCH$IOLOCKW : lock I/O database for write access
51 04 AC D0 0113 379 MOVL DEVNAM(AP),R1 : get address of device name string descript
06 13 0117 380 BEQL 10$ : if eql none
05 0119 381 IFNORD #8,(R1),ACCVIO : can device name descriptor be read?
011F 382 10$: RSB :
0120 383 :
0120 384 : Miscellaneous exits
0120 385 :
50 0C 3C 0120 386 ACCVIO: MOVZWL #SS$ ACCVIO,R0 : set access violation status
08 11 0123 387 BRB UNLOCK :
0125 388 :
50 24 3C 0125 389 NOPRIV: MOVZWL #SS$ NOPRIV,R0 : set no privilege
03 11 0128 390 BRB UNLOCK :
012A 391 :
50 01 3C 012A 392 NORMAL: MOVZWL #SS$ NORMAL,R0 : set normal completion
FED0' 31 012D 393 UNLOCK: BRW IOC$UNLOCK : unlock I/O data base and return
```



```
0130 395 .SBTTL Decrement Reference Count, Clear Ownership, and Call Driver
0130 396
0130 397 ;
0130 398 ; Subroutine to decrement reference count, clear ownership, and call driver
0130 399 ;
0130 400
0130 401 DECREF:
0130 402 MOVZWL #SS$ DEVMOUNT,R0 ; assume device mounted
0135 403 BBS #DEV$V_MNT,- ; branch if device mounted
0137 404 UCB$$_DEVCHAR(R1),10$
013A 405 PUSHL R5 ; save R5
013C 406 MOVL R1,R5 ; set address of device UCB
013F 407 BSBW IOC$DALLOC_DEV ; clear device allocation
0142 408 MOVL (SP)+,R5 ; restore R5
0145 409 10$: RSB
0146 410
0146 411
0146 412
0146 413 .END
```

50 006C 8F 3C 0130 402
OB 38 A1 E0 0135 403
55 51 DD 0137 404
FE BE 30 013A 405
55 8E D0 013C 406
D0 013F 407
05 0142 408
0145 409
0146 410
0146 411
0146 412
0146 413

SYSDEVALC
Symbol table

- SYSTEM SERVICES FOR DEVICE ALLOCATION

D 12

16-SEP-1984 02:00:06
5-SEP-1984 03:52:38

VAX/VMS Macro V04-00
[SYS.SRC]SYSDEVALC.MAR;1

Page 11
(5)

ACCVIO	00000120	R	02
ALACMODE	= 00000010		
CTL\$GL_PCB	*****	X	02
DECREP	00000130	R	02
DEVS\$V_ALL	= 00000017		
DEVS\$V_MNT	= 00000013		
DEVICE_TYPE	= 00000000		
DEVNAM	= 00000004		
DLACMODE	= 00000008		
EXES\$ALLOC	00000006	RG	02
EXES\$DALLOC	000000AC	RG	02
EXES\$MAXACMODE	*****	X	02
FLAGS	= 00000014		
IOCS\$CVT_DEVNAM	*****	X	02
IOCS\$DALLOC_DEV	*****	X	02
IOCS\$M_TYPE	= 00000002		
IOCS\$SCAN_IODB	*****	X	02
IOCS\$SEARCH	*****	X	02
IOCS\$SEARCHDEV	*****	X	02
IOCS\$UNLOCK	*****	X	02
IOCS\$UNLOCK_DEV	*****	X	02
LOCK	0000010D	R	02
NOPRIV	00000125	R	02
NORMAL	0000012A	R	02
PCBS\$L_PID	= 00000060		
PHYBUF	= 0000000C		
PHYLEN	= 00000008		
SCH\$IOLOCKW	*****	X	02
SS\$ACCVIO	= 0000000C		
SS\$DEVALRALLOC	= 00000641		
SS\$DEVMOUNT	= 0000006C		
SS\$DEVNOTALLOC	= 00000858		
SS\$IVDEVNAM	= 00000144		
SS\$IVSTSFLG	= 0000017C		
SS\$NOPRIV	= 00000024		
SS\$NORMAL	= 00000001		
UCBS\$B_AMOD	= 0000005F		
UCBS\$L_DEVCHAR	= 00000038		
UCBS\$L_PID	= 0000002C		
UCBS\$W_REFC	= 0000005C		
UNLOCK	0000012D	R	02

+-----+
! Psect synopsis !
+-----+

PSECT name

. ABS .
\$ABSS
Y\$EXEPAGED

Allocation

00000000 (0.)
00000000 (0.)
00000146 (326.)

PSECT No.

00 (0.)
01 (1.)
02 (2.)

Attributes

NOPIC USR
NOPIC USR
NOPIC USR

CON
CON
CON

ABS
ABS
REL

LCL
LCL
LCL

NOSHR
NOSHR
NOSHR

NOEXE
EXE
EXE

NORD
RD
RD

NOWRT
WRT
WRT

NOVEC
NOVEC
NOVEC

BYTE
BYTE
BYTE

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

Phase	Page faults	CPU Time	Elapsed Time
-----	-----	-----	-----
Initialization	37	00:00:00.08	00:00:00.79
Command processing	135	00:00:00.53	00:00:02.76
Pass 1	307	00:00:09.35	00:00:29.17
Symbol table sort	0	00:00:01.52	00:00:03.80
Pass 2	87	00:00:01.84	00:00:04.19
Symbol table output	7	00:00:00.08	00:00:00.42
Psect synopsis output	1	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	576	00:00:13.44	00:00:41.16

The working set limit was 1500 pages.

53116 bytes (104 pages) of virtual memory were used to buffer the intermediate code.

There were 60 pages of symbol table space allocated to hold 1028 non-local and 18 local symbols.

413 source lines were read in Pass 1, producing 19 object records in Pass 2.

16 pages of virtual memory were used to define 15 macros.

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

Macro library name	Macros defined
-----	-----
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	7
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	5
TOTALS (all libraries)	12

1097 GETS were required to define 12 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SYSDEVALC/OBJ=OBJ\$:SYSDEVALC MSRC\$:SYSDEVALC/UPDATE=(ENH\$:SYSDEVALC)+EXECMLS/LIB

0383 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SYSRMPSC
LIS

SYSDCLEXH
LIS

SYSDEVALC
LIS

SYSCURTIM
LIS

SYSDBGLSC
LIS

SYSENQDEQ
LIS

SYSDCLCMH
LIS

SYSDERLMB
LIS

SYSDBSSGN
LIS

SYSDELPRC
LIS