



```

IIIIII  SSSSSSSS  DDDDDDDD  SSSSSSSS  000000  RRRRRRRR  TTTTTTTTTT
IIIIII  SSSSSSSS  DDDDDDDD  SSSSSSSS  000000  RRRRRRRR  TTTTTTTTTT
II      SS      DD      SS      00      RR      TT
II      SS      DD      SS      00      RR      TT
II      SS      DD      SS      00      RR      TT
II      SS      DD      SS      00      RR      TT
II      SSSSSS  DD      SSSSSS  00      RRRRRRRR  TT
II      SSSSSS  DD      SSSSSS  00      RRRRRRRR  TT
II      SS      DD      SS      00      RR  RR  TT
II      SS      DD      SS      00      RR  RR  TT
II      SS      DD      SS      00      RR  RR  TT
II      SS      DD      SS      00      RR  RR  TT
IIIIII  SSSSSSSS  DDDDDDDD  SSSSSSSS  000000  RR      TT
IIIIII  SSSSSSSS  DDDDDDDD  SSSSSSSS  000000  RR      TT
IIIIII  SSSSSSSS  DDDDDDDD  SSSSSSSS  000000  RR      TT

```

```

LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      SSSSSS
LL      SSSSSS
LL      SS
LL      SS
LL      SS
LL      SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS

```

```

....
....
....
....

```

.....

(2) 50  
(3) 65

Declarations  
lnk\$sortisects sort image section descriptors

.....

```

0000 1      .title ISDSORT Sort image section descriptors
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 FACILITY
0000 29
0000 30 Linker
0000 31
0000 32 ABSTRACT
0000 33
0000 34 Sort a list of isect descriptor addresses into virtual address order
0000 35
0000 36 ENVIRONMENT
0000 37
0000 38 Native mode, user mode
0000 39
0000 40 AUTHOR
0000 41
0000 42 Benn Schreiber, 17-Jan-1980
0000 43
0000 44 MODIFIED BY
0000 45
0000 46 J02-003 BLS0007 Benn Schreiber 15-Aug-1980
0000 47 Convert to MDL data structures
0000 48 :--

```

0000 50 .sbttl Declarations

0000 51 :

0000 52 : Symbol defintions

0000 53 :

0000 54 :

0000 55 \$isddef ;define isect descriptor

0000 56 \$isldef ;define isect list descriptor

0000 57

00000000 58 .psect \$code\$,exe,nowrt

0000 59

00000028 0000000D 00000004 00000001 0000 60 steps: .long 1, 4, 13, 40, 121, 364, 1093, 3280, 9841, 32767 ;Steps for shellsort

00000C00 00000445 0000016C 00000079 0010

00007FFF 00002671 0020

0000000A 0028

61 numsteps = .-steps/4

0028 62

0028 63 .default displacement,word

```

0028 65 .sbttl lnk$sortisects sort image section descriptors
0028 66
0028 67 :++
0028 68 :
0028 69 : Inputs:
0028 70 :
0028 71 : 4(ap) number of entries in list
0028 72 : 8(ap) Address of list of isect descriptor addresses
0028 73 :
0028 74 : Outputs:
0028 75 :
0028 76 : the list is sort'd
0028 77 :--
0028 78 :
07FC 0028 79 .entry lnk$sortisects,^m<r2, r3, r4, r5, r6, r7, r8, r9, r10>
002A 80 :
002A 81 :
002A 82 : determine highest step to use
002A 83 :
002A 84 :
50 04 AC D0 002C 85 : clrl r10 ;index starts at 0
50 D4 AF4A D1 0030 86 10$: movl 4(ap),r0 ;get number of keys
FFF3 5A 01 07 F1 0035 87 : cmpl steps+8[r10],r0 ;this step high enough?
7E 08 AC 04 C3 0037 88 : bgeq 20$ ;if geq yes
01 5A 07 D0 003D 89 : acbl #<numsteps-3>,#1,r10,10$ ;no--look through all - 3
01 6E D1 0040 90 20$: movl #<numsteps-3>,r10 ;lots of symbols--use all steps
01 49 1B 0045 91 : subl3 #4,8(ap),-(sp) ;set table address-4 on stack
004A 92 : pushl r0 ;set # of entries on stack
004C 93 : cmpl (sp),#1 ;If there are not at least two entries
004C 94 : blequ sort_exit ; then quit now
004C 95 :
004C 96 : now do the shell sort on the list. The shell sort is described in
004C 97 : Knuth Vol. 3 and is also referred to as the Diminishing Increment Sort.
004C 98 :
59 B0 AF4A D0 004C 99 shell_sort:
58 01 A9 9E 0051 100 10$: movl steps[r10],r9 ;get step value for this 't'
56 04 BE48 D0 0055 101 20$: movab 1(r9),r8 ;set up loop for step+1 to index
57 58 59 C3 005A 102 : movl @4(sp)[r8],r6 ;get address of key block for j'th key
54 04 BE47 D0 005E 103 30$: subl3 r9,r8,r7 ;i=j-h
15 00 EF 0063 104 40$: movl @4(sp)[r7],r4 ;get address of key block for i'th key
50 1C A4 0066 105 : extzv #isd$v_vpn,#isd$s_vpn,- ;extract the vpn of the isect
15 00 ED 0069 106 : is($t_hdrisd+isd$l_vpnpc(r4),r0
50 1C A6 006C 107 : #isd$v_vpn,#isd$s_vpn,- ;and compare them
08 1F 006F 108 : is($t_hdrisd+isd$l_vpnpc(r6),r0 ;and compare them
50 59 57 C1 0071 109 50$: blssu 60$
04 BE40 56 D0 0075 110 : addl3 r7,r9,r0 ;compute i+h
10 11 007A 111 : movl r6,@4(sp)[r0] ;ids(i+h) = val
50 59 57 C1 007C 112 60$: brb 70$
04 BE40 54 D0 0080 113 : addl3 r7,r9,r0 ;ids(i+h) = ids(i)
57 59 C2 0085 114 : movl r4,@4(sp)[r0]
D4 14 0088 115 : subl2 r9,r7 ;i=i-h
E5 11 008A 116 : bgtr 30$
FFC3 58 01 6E F1 008C 117 70$: brb 50$ ;go set ids(i+h)=val
B7 5A F4 0092 118 80$: acbl (sp),#1,r8,20$ ;loop for all entries in table
0095 119 : sort_exit: ;loop for all steps
04 0095 120 : ret
0096 121 :

```

ISDSORT  
V04-000

Sort image section descriptors L 15  
lnk\$sortisects sort image section descri 15-SEP-1984 23:55:41 VAX/VMS Macro V04-00  
5-SEP-1984 01:42:10 [LINKER.SRC]ISDSORT.MAR;1  
0096 122 .END

ISDSORT  
Symbol table

Sort image section descriptors

M 15

15-SEP-1984 23:55:41  
5-SEP-1984 01:42:10

VAX/VMS Macro V04-00  
[LINKER.SRC]ISDSORT.MAR;1

Page 5  
(3)

LN  
VO

```

ISD$L_VPNPFC      = 00000004
ISD$S_VPN         = 00000015
ISD$V_VPN         = 00000000
ISL$B_NEWPRT      00000016
ISL$C_SIZE        00000018
ISL$K_SIZE        00000018
ISL$L_BUFADR      00000008
ISL$L_BUFDSC      00000008
ISL$L_BUFEND      0000000C
ISL$L_CLUDSC      00000010
ISL$L_NXTISD      00000000
ISL$L_PREVISD     00000004
ISL$T_HDRISD      00000018
ISL$W_FLAGS       00000014
LNK$SORTISECTS    = 00000028  RG   02
NUMSTEPS          = 0000000A
SHELL_SORT        0000004C  R    02
SORT_EXIT         00000095  R    02
STEPS             00000000  R    02
  
```

-----  
! 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	00000018 ( 24.)	01 ( 1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
\$CODES	00000096 ( 150.)	02 ( 2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.86
Command processing	129	00:00:00.74	00:00:04.51
Pass 1	146	00:00:01.67	00:00:06.43
Symbol table sort	0	00:00:00.06	00:00:00.06
Pass 2	45	00:00:00.49	00:00:02.23
Symbol table output	4	00:00:00.01	00:00:00.07
Psect synopsis output	1	00:00:00.01	00:00:00.05
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	356	00:00:03.07	00:00:14.21

The working set limit was 1200 pages.  
6331 bytes (13 pages) of virtual memory were used to buffer the intermediate code.  
There were 10 pages of symbol table space allocated to hold 84 non-local and 10 local symbols.  
122 source lines were read in Pass 1, producing 15 object records in Pass 2.  
9 pages of virtual memory were used to define 8 macros.



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

Macro library name	Macros defined
-----	-----
_\$255SDUA28:[SYS.OBJ]LIB.MLB;1	1
-\$255SDUA28:[LINKER.OBJ]LNK.MLB;1	1
-\$255SDUA28:[SYSLIB]STARLET.MLB;2	3
TOTALS (all libraries)	5

148 GETS were required to define 5 macros.

There were no errors, warnings or information messages.

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

B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z  
[  
] \_  
,  
-  
.  
/



**0215 AH-BT13A-SE**  
**VAX/VMS V4.0**

**DIGITAL EQUIPMENT CORPORATION**  
**CONFIDENTIAL AND PROPRIETARY**

