





```
0000 1 .TITLE MPWIND - MAP BLOCKS THROUGH FILE WINDOW
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 **
0000 29
0000 30 FACILITY: F11ACP STRUCTURE LEVEL 1
0000 31
0000 32 ABSTRACT:
0000 33
0000 34 THIS ROUTINE MAPS THE GIVEN VIRTUAL BLOCK(S) INTO THE
0000 35 CORRESPONDING LOGICAL BLOCKS.
0000 36
0000 37 ENVIRONMENT:
0000 38
0000 39 STARLET OPERATING SYSTEM, INCLUDING PRIVILEGED SYSTEM SERVICES
0000 40 AND INTERNAL EXEC ROUTINES. THIS ROUTINE MUST BE EXECUTED IN
0000 41 KERNEL MODE.
0000 42
0000 43 --
0000 44
0000 45 AUTHOR: ANDREW C. GOLDSTEIN, CREATION DATE: 3-MAR-1977 11:15
0000 46
0000 47 MODIFIED BY:
0000 48
0000 49 **
0000 50
0000 51
0000 52 INCLUDE FILES:
0000 53
0000 54 .INCLUDE FCPDEF.MAR
0000 55
0000 56
0000 57 EQUATED SYMBOLS:
```

```
0000 58 :  
0000 59 : AP OFFSETS  
0000 60 :  
00000004 0000 61 VBN = 4 ; DESIRED VBN  
00000008 0000 62 WINDOW = 8 ; WINDOW ADDRESS  
0000000C 0000 63 COUNT = 12 ; BLOCK COUNT TO MAP  
00000010 0000 64 UNMAPPED= 16 ; ADDRESS OF LONGWORD TO STORE  
0000 65 ; COUNT OF UNMAPPED BLOCKS  
0000 66  
0000 67  
0000 68 $WCBDEF ; DEFINE WINDOW STRUCTURE
```

```

0000 70 :++
0000 71 :
0000 72 : FUNCTIONAL DESCRIPTION:
0000 73 :
0000 74 :     THIS ROUTINE MAPS THE GIVEN VIRTUAL BLOCK(S) INTO THE
0000 75 :     CORRESPONDING LOGICAL BLOCKS.
0000 76 :
0000 77 : CALLING SEQUENCE:
0000 78 :     CALL     MAP_WINDOW (ARG1, ARG2, ARG3, ARG4)
0000 79 :
0000 80 : INPUT PARAMETERS:
0000 81 :     ARG1: DESIRED VBN
0000 82 :     ARG2: ADDRESS OF WINDOW TO USE
0000 83 :     ARG3: NUMBER OF BLOCKS TO MAP
0000 84 :
0000 85 : IMPLICIT INPUTS:
0000 86 :     CURRENT_UCB CONTAINS UCB ADDRESS OF UNIT IN PROCESS
0000 87 :
0000 88 : OUTPUT PARAMETERS:
0000 89 :     ARG4: ADDRESS OF LONGWORD TO RECEIVE UNMAPPED COUNT
0000 90 :
0000 91 : IMPLICIT OUTPUTS:
0000 92 :     NONE
0000 93 :
0000 94 : ROUTINE VALUE:
0000 95 :     LBN IF ANY BLOCKS MAPPED
0000 96 :     -1 IF NONE MAPPED
0000 97 :
0000 98 : SIDE EFFECTS:
0000 99 :     NONE
0000 100 :
0000 101 :--
0000 102 :
00000000 103 : .PSECT $CODE$,NOWRT, LONG
0000 104 :
0000 105 MAP_WINDOW::
0000 106 : .WORD ^M<R2,R3,R4,R5> ; SAVE REGISTERS
52 08 AC DO 0002 107 : MOVL WINDOW(AP),R2 ; GET WINDOW ADDRESS
55 0000'CF DO 0006 108 : MOVL W^CURRENT_UCB,R5 ; GET UCB ADDRESS FOR MAPPER
000B 109 :
50 04 AC DO 000B 110 10$: MOVL VBN(AP),R0 ; GET VBN
51 0C AC 09 78 000F 111 : ASHL #9,COUNT(AP),R1 ; GET EXPLICIT COUNT
00000000'9F 16 0014 112 20$: JSB @#IOC$MAPVBLK ; CALL SYSTEM MAPPING ROUTINE
0A 50 E9 001A 113 : BLBC R0,40$ ; BRANCH IF NO MAP
001D 114 :
001D 115 : SUCCESSFUL MAP - RETURN LBN AND COUNT OF UNMAPPED BLOCKS IF WANTED
001D :16 :
001D 117 : MOVL R1,R0 ; LBN TO ROUTINE VALUE
10 BC 50 51 DO 001D 117 : MOVL R1,R0 ; LBN TO ROUTINE VALUE
52 F7 8F 78 0020 118 : ASHL #-9,R2,@UNMAPPED(AP) ; STORE RESULT
04 0026 119 30$: RET ; AND RETURN
0027 120 :
0027 121 : WE GET HERE IF THE MAP FAILS COMPLETELY. RETURN -1 AS VALUE.
0027 122 :
0027 123 40$: MOVL COUNT(AP),@UNMAPPED(AP) ; RETURN ENTIRE COUNT AS UNMAPPED
10 BC 0C AC DO 0027 123 40$: MOVL COUNT(AP),@UNMAPPED(AP) ; RETURN ENTIRE COUNT AS UNMAPPED
50 01 CE 002C 124 : MNEGL #1,R0 ; VALUE = -1
04 002F 125 : RET
0030 126 :

```

MPWIND  
V04-000

- MAP BLOCKS THROUGH FILE WINDOW K 5

16-SEP-1984 00:44:22 VAX/VMS Macro V04-00  
5-SEP-1984 01:08:38 [F11A.SRC]MPWIND.MAR;1

Page 4  
(2)

PM  
VO

0030 127  
0030 128  
0030 129 .END

.....

MPWIND  
Symbol table

- MAP BLOCKS THROUGH FILE WINDOW L 5

16-SEP-1984 00:44:22 VAX/VMS Macro V04-00  
5-SEP-1984 01:08:38 [F11A.SRC]MPWIND.MAR;1

Page 5  
(2)

PM  
VO

```

AOB_TYPE           = 00000005
BITMAP_TYPE        = 00000001
COUNT             = 0000000C
CURRENT_UCB        ***** X 02
DIRECTORY_TYPE     = 00000002
FCB_TYPE           = 00000000
HEADER_TYPE        = 00000000
INDEX_TYPE         = 00000003
IOCSMAPVBLK       ***** X 02
MAP_WINDOW         = 00000000 RG 02
MVL_TYPE           = 00000004
RVT_TYPE           = 00000003
UNMAPPED           = 00000010
VBN                = 00000004
VCB_TYPE           = 00000002
WCB_TYPE           = 00000001
WINDOW             = 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 |
| \$AB\$\$   | 00000000 ( 0.)  | 01 ( 1.)  | NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE       |
| \$CODE\$   | 00000030 ( 48.) | 02 ( 2.)  | NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC LONG     |

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

| Phase                  | Page faults | CPU Time    | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization         | 36          | 00:00:00.08 | 00:00:01.23  |
| Command processing     | 139         | 00:00:00.77 | 00:00:04.45  |
| Pass 1                 | 142         | 00:00:01.64 | 00:00:06.04  |
| Symbol table sort      | 0           | 00:00:00.07 | 00:00:00.09  |
| Pass 2                 | 39          | 00:00:00.60 | 00:00:02.48  |
| Symbol table output    | 4           | 00:00:00.02 | 00:00:00.06  |
| Psect synopsis output  | 1           | 00:00:00.02 | 00:00:00.06  |
| Cross-reference output | 0           | 00:00:00.00 | 00:00:00.00  |
| Assembler run totals   | 364         | 00:00:03.21 | 00:00:14.50  |

The working set limit was 1050 pages.  
6379 bytes (13 pages) of virtual memory were used to buffer the intermediate code.  
There were 10 pages of symbol table space allocated to hold 97 non-local and 4 local symbols.  
232 source lines were read in Pass 1, producing 13 object records in Pass 2.  
12 pages of virtual memory were used to define 11 macros.

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

| Macro library name                  | Macros defined |
|-------------------------------------|----------------|
| -----                               | -----          |
| _\$255\$DUA28:[SYS.OBJ]LIB.MLB;1    | 1              |
| _\$255\$DUA28:[SYSLIB]STARLET.MLB;2 | 3              |
| TOTALS (all libraries)              | 4              |

139 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:MPWIND/OBJ=OBJ\$:MPWIND MSRC\$:FCPPRE/UPDATE=(ENH\$:FCPPRE)+MSRC\$:MPWIND/UPDATE=(ENH\$:MPWIND)+EXECMI.\$/LIB



The image displays a grid of 100 terminal windows, arranged in 10 rows and 10 columns. Each window contains a program name followed by the letters 'LIS'. The programs are: Row 1: REQUEL LIS, RWATTR LIS; Row 2: MOOTFY LIS; Row 3: SCHFCB LIS; Row 4: MAKACC LIS; Row 5: MPWIND LIS; Row 6: MAPUBN LIS, PMS LIS, RDHEDR LIS, RWJB LIS; Row 7: RETDIR LIS; Row 8: ROBLOK LIS; Row 9: SMALOC LIS; Row 10: MAKMBE LIS, MAKSTR LIS, MXTHOR LIS. The background of each window is dark with light-colored text and some graphical elements like bar charts.