


```

RRRRRRRR      EEEEEEEEEEE      QQQQQQ      UU      UU      EEEEEEEEEEE      UU      UU
RRRRRRRR      EEEEEEEEEEE      QQQQQQ      UU      UU      EEEEEEEEEEE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RRRRRRRR      EEEEEEEEEEE      QQ      QQ      UU      UU      EEEEEEEEEEE      UU      UU
RRRRRRRR      EEEEEEEEEEE      QQ      QQ      UU      UU      EEEEEEEEEEE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EE      QQ      QQ      UU      UU      EE      UU      UU
RR      RR      EEEEEEEEEEE      QQQQ      QQ      UUUUUUUUUU      EEEEEEEEEEE      UUUUUUUUUU
RR      RR      EEEEEEEEEEE      QQQQ      QQ      UUUUUUUUUU      EEEEEEEEEEE      UUUUUUUUUU

```

```

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

```

```

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
LLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLL      IIIIII      SSSSSSSS

```

```
0000 1 .TITLE REQUEU - REQUEUE REQUEST TO DRIVER
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 REQUEUES THE INDICATED I/O PACKET TO THE DEVICE
0000 35 : DRIVER FOR WHICH IT WAS ORIGINALLY INTENDED.
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 CALLED IN
0000 41 : KERNEL MODE.
0000 42
0000 43 :--
0000 44
0000 45 : AUTHOR: ANDREW C. GOLDSTEIN 14-MAR-78 10:43
0000 46
0000 47 : MODIFIED BY:
0000 48
0000 49 : V03-001 RLRMXBCNT Robert L. Rappaport 11-Mar-1983
0000 50 : Allow for segmentation of Logical I/O (and Virtual)
0000 51 : based on the UCBSL_MAXBCNT field.
0000 52
0000 53 :**
0000 54
0000 55
0000 56 : EQUATED SYMBOLS:
0000 57 :
```

REQUEU
V04-000

- REQUEUE REQUEST TO DRIVER

D 11

16-SEP-1984 00:44:41 VAX/VMS Macro V04-00
5-SEP-1984 01:08:57 [F11A.SRC]REQUEU.MAR;1

Page 2
(1)

00000074	0000	58	PACKET = 4
00000008	0000	59	_BN = 8
0000000c	0000	60	UNMAPPED= 12
	0000	61	
	0000	62	\$IRPDEF
	0000	63	\$IODEF
	0000	64	\$UCBDEF

: ADDRESS OF I/O PACKET ARG
: STARTING LBN OF TRANSFER
: COUNT OF UNMAPPED BLOCKS

: DEFINE I/O PACKET OFFSETS
: DEFINE I/O FUNCTION CODES
: DEFINE UCB OFFSETS

RI
V(

```

0000 66 :++
0000 67 :
0000 68 : FUNCTIONAL DESCRIPTION:
0000 69 :
0000 70 :     THIS ROUTINE REQUEUES THE INDICATED I/O PACKET TO THE DEVICE
0000 71 :     DRIVER FOR WHICH IT WAS ORIGINALLY INTENDED. IT TRANSLATES THE
0000 72 :     LBN INTO THE CORRESPONDING PHYSICAL BLOCK NUMBER AND CONVERTS THE
0000 73 :     I/O FUNCTION CODE INTO THE APPROPRIATE PHYSICAL FUNCTION.
0000 74 :     THE NUMBER OF UNMAPPED BLOCKS IS DEDUCTED FROM THE BYTE COUNT.
0000 75 :
0000 76 : CALLING SEQUENCE:
0000 77 :     CALL REQUEUE_REQ (ARG1, ARG2, ARG3)
0000 78 :
0000 79 : INPUT PARAMETERS:
0000 80 :     ARG1: ADDRESS OF I/O PACKET
0000 81 :     ARG2: STARTING LBN OF TRANSFER
0000 82 :     ARG3: NUMBER OF BLOCKS UNMAPPED
0000 83 :
0000 84 : IMPLICIT INPUTS:
0000 85 :     CURRENT_UCB: ADDRESS OF REQUEST UCB
0000 86 :
0000 87 : OUTPUT PARAMETERS:
0000 88 :     NONE
0000 89 :
0000 90 : IMPLICIT OUTPUTS:
0000 91 :     NONE
0000 92 :
0000 93 : ROUTINE VALUE:
0000 94 :     NONE
0000 95 :
0000 96 : SIDE EFFECTS:
0000 97 :     REQUEST QUEUED TO UCB
0000 98 :
0000 99 :--
0000 100 :
0000 101 : .PSECT $CODE$,NOWRT, LONG
0000 102 :
0000 103 REQUEUE_REQ:
0000 104     .WORD    ^M<R2,R3,R4,R5>           ; SAVE REGISTERS
50 53 04 AC 003C 0002 105     MOVL    PACKET(AP),R3           ; GET PACKET ADDRESS
50 55 0000 CF 0006 106     MOVL    W^CURRENT_UCB,R5         ; GET UCB ADDRESS
32 A3 0000 01FF 8F C0 0016 107     ASHL    #9,UNMAPPED(AP),R0       ; GET BYTE COUNT OF UNMAPPED BLOCKS
32 A3 0000 01FF 8F AA 001E 108     BEQL    10$                       ; BRANCH IF ALL MAPPED - NO FIXUP
32 A3 0000 01FF 8F AA 001E 109     SUBL    R0,IRPSL_BCNT(R3)         ; AND SUBTRACT FROM TRANSFER COUNT
50 00B4 C5 D0 0024 110     ADDL    #511,IRPSL_BCNT(R3)      ; ROUND BYTE COUNT TO NEXT BLOCK BOUNDARY
50 FE00 8F 3C 002B 111     BICW    #511,IRPSL_BCNT(R3)      ; IN CASE FULL BYTE COUNT CONTAINS A PARTIAL
32 A3 50 D1 0030 112 10$:     MOVL    UCBSL_MAXBCNT(R5),R0     ; R0 = 0 or Max. permissible BCNT.
32 A3 50 D0 0036 113     BNEQ    20$                       ; NEQ implies Max. permissible BCNT in R0.
50 08 AC D0 003A 114     MOVZWL #512*127,R0              ; If 0, use default Max. permissible.
32 A3 50 D1 0030 115 20$:     CMPL    R0,IRPSL_BCNT(R3)         ; See if BCNT too large.
32 A3 50 D0 0036 116     BGEQ    30$                       ; GEQ implies we are OK.
50 08 AC D0 003A 117     MOVL    R0,IRPSL_BCNT(R3)         ; Else scale down to maximum allowed.
50 00000000 9F 16 003E 118 30$:     MOVL    LBN(AP),R0               ; GET STARTING LBN
50 00000000 9F 16 0044 119     JSB     @#IOC$CVTLOGPHY          ; CONVERT TO PHYSICAL BLOCK
50 00000000 9F 16 0044 120     JSB     @#EXE$INSIOQ            ; AND QUEUE TO DRIVER
50 00000000 9F 16 0044 121
50 00000000 9F 16 0044 122

```

REQUEU
V04-000

- REQUEUE REQUEST TO DRIVER

F 11

16-SEP-1984 00:44:41 VAX/VMS Macro V04-00
5-SEP-1984 01:08:57 [F11A.SRC]REQUEU.MAR;1

Page 4
(2)

04	004A	123	RET
	004B	124	
	004B	125	
	004B	126	
	004B	127	.END

REQUEU
Symbol table

- REQUEUE REQUEST TO DRIVER

G 11

16-SEP-1984 00:44:41 VAX/VMS Macro V04-00
5-SEP-1984 01:08:57 [F11A.SRC]REQUEU.MAR;1

Page 5
(2)

AQB_TYP	=	00000005		
BITMAP_TYP	=	00000001		
CURRENT_UCB	=	*****	X	02
DIRECTORY_TYPE	=	00000002		
EXESINSIOG	=	*****	X	02
FCB_TYPE	=	00000000		
HEADER_TYPE	=	00000000		
INDEX_TYPE	=	00000003		
IOC\$CVTLOGPHY	=	*****	X	02
IRPSL_BCNT	=	00000032		
LBN	=	00000008		
MVL_TYPE	=	00000004		
PACKET	=	00000004		
REQUEUE_REQ	=	00000000	RG	02
RVT_TYPE	=	00000003		
UCBSL_MAXBCNT	=	000000B4		
UNMAPPED	=	0000000C		
VCB_TYPE	=	00000002		
WCB_TYPE	=	00000001		

! 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	00000000 (0.)	01 (1.)	NOPIC USR	CON	ABS	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE	
\$CODES	0000004B (75.)	02 (2.)	NOPIC USR	CON	REL	LCL	NOSHR	EXE	RD	NOWRT	NOVEC	LONG	

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.13	00:00:00.96
Command processing	130	00:00:00.75	00:00:04.64
Pass 1	270	00:00:07.61	00:00:20.96
Symbol table sort	0	00:00:01.39	00:00:02.50
Pass 2	41	00:00:01.37	00:00:04.88
Symbol table output	4	00:00:00.05	00:00:00.05
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	479	00:00:11.33	00:00:34.02

The working set limit was 1350 pages.
41203 bytes (81 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 870 non-local and 3 local symbols.
230 source lines were read in Pass 1, producing 13 object records in Pass 2.
14 pages of virtual memory were used to define 13 macros.

! Macro library statistics !

Macro library name	Macros defined
-----	-----
\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	2
\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	6

920 GETS were required to define 6 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:REQUEU/OBJ=OBJ\$:REQUEU MSRC\$:FCPPRE/UPDATE=(ENHS:FCPPRE)+MSRC\$:REQUEU/UPDATE=(ENHS:REQUEU)+EXECMLS/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 '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 text is faint and the background is dark.