H 5

```
FFFFFFFFFFFFFFF        111             111          XXX         XXX
FFFFFFFFFFFFFFF        111             111          XXX         XXX
FFFFFFFFFFFFFFF        111             111          XXX         XXX
FFF                 111111          111111          XXX         XXX
FFF                 111111          111111          XXX         XXX
FFF                 111111          111111          XXX         XXX
FFF                    111             111             XXX   XXX
FFF                    111             111             XXX   XXX
FFF                    111             111             XXX   XXX
FFFFFFFF.FFF           111             111                XXX
FFFFFFFFFFFF           111             111                XXX
FFFFFFFFFFFF           111             111                XXX
FFF                    111             111             XXX   XXX
FFF                    111             111             XXX   XXX
FFF                    111             111             XXX   XXX
FFF                    111             111          XXX         XXX
FFF                    111             111          XXX         XXX
FFF                    111             111          XXX         XXX
FFF                 111111111       111111111       XXX         XXX
FFF                 111111111       111111111       XXX         XXX
FFF                 111111111       111111111       XXX         XXX
```

_$25

Symb

IOC$
IO_C
IO_C
IO_D
IO_F
IO_S
KILL

KILL
KILL
LB_E
LB_D
LB_F
LB_H
LB_L
LOCA
LOCA
LOCK

LOCK
LOCK
LOCK
LOC_
LOC_
L_CO
L_CO
L_DA
L_DA
MAIN
MAKE
MAKE
MAKE
MAKE
MAKE
MAKE

MAKE
MAKE
MAP_
MAP_

MAP
MARK
MARK
MARK
MARK
MARK

```
RRRRRRRR    EEEEEEEEEE    QQQQQQ    UU      UU  EEEEEEEEEE  UU      UU
RRRRRRRR    EEEEEEEEEE    QQQQQQ    UU      UU  EEEEEEEEEE  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    EEEEEEEE      QQ    QQ  UU      UU  EEEEEEEE    UU      UU
RRRRRRRR    EEEEEEEE      QQ    QQ  UU      UU  EEEEEEEE    UU      UU
RR  RR      EE            QQ  QQQQ  UU      UU  EE          UU      UU
RR  RR      EE            QQ  QQQQ  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  EEEEEEEEEE    QQQQ  QQ  UUUUUUUUUU  EEEEEEEEEE  UUUUUUUUUU      ....
RR      RR  EEEEEEEEEE    QQQQ  QQ  UUUUUUUUUU  EEEEEEEEEE  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  ;       V03-003 ROW0348         Ralph O. Weber          12-APR-1984
0000    50  ;               Change maximum byte count, UCB$L_MAXBCNT, tests to be
0000    51  ;               unsigned.  This should have been done in ROW0218, when
0000    52  ;               SYSACPFDT and IOCIOPOST were fixed, but what can I say,
0000    53  ;               "Nobody's perfect."
0000    54  ;
0000    55  ;       V03-002 ACG0408         Andrew C. Goldstein,    23-Mar-1984  11:01
0000    56  ;               Make all of global storage based
0000    57  ;
```

```
                     0000     58 ;       V03-001 RLRMXBCNT       Robert L. Rappaport     11-Mar-1983
                     0000     59 ;               Allow for segmentation of Logical I/O (and Virtual)
                     0000     60 ;               based on the UCB$L_MAXBCNT field.
                     0000     61 ;
                     0000     62 ;**
                     0000     63
                     0000     64 ;
                     0000     65 ; EQUATED SYMBOLS:
                     0000     66 ;
00000004             0000     67 PACKET  =  4                          ; ADDRESS OF I/O PACKET ARG
00000008             0000     68 LBN     =  8                          ; STARTING LBN OF TRANSFER
0000000C             0000     69 UNMAPPED= 12                          ; COUNT OF UNMAPPED BLOCKS
                     0000     70
                     0000     71         $IRPDEF                       ; DEFINE I/O PACKET OFFSETS
                     0000     72         $IODEF                        ; DEFINE I/O FUNCTION CODES
                     0000     73         $UCBDEF                       ; DEFINE UCB OFFSETS
```

```
                                   0000  ;++
                                   0000  ;
                                   0000  ; FUNCTIONAL DESCRIPTION:
                                   0000  ;
                                   0000  ;       THIS ROUTINE REQUEUES THE INDICATED I/O PACKET TO THE DEVICE
                                   0000  ;       DRIVER FOR WHICH IT WAS ORIGINALLY INTENDED. IT TRANSLATES THE
                                   0000  ;       LBN INTO THE CORRESPONDING PHYSICAL BLOCK NUMBER AND CONVERTS THE
                                   0000  ;       I/O FUNCTION CODE INTO THE APPROPRIATE PHYSICAL FUNCTION.
                                   0000  ;       THE NUMBER OF UNMAPPED BLOCKS IS DEDUCTED FROM THE BYTE COUNT.
                                   0000  ;
                                   0000  ; CALLING SEQUENCE:
                                   0000  ;       CALL REQUEUE_REQ (ARG1, ARG2, ARG3)
                                   0000  ;
                                   0000  ; INPUT PARAMETERS:
                                   0000  ;       ARG1: ADDRESS OF I/O PACKET
                                   0000  ;       ARG2: STARTING LBN OF TRANSFER
                                   0000  ;       ARG3: NUMBER OF BLOCKS UNMAPPED
                                   0000  ;
                                   0000  ; IMPLICIT INPUTS:
                                   0000  ;       CURRENT_UCB: ADDRESS OF REQUEST UCB
                                   0000  ;
                                   0000  ; OUTPUT PARAMETERS:
                                   0000  ;       NONE
                                   0000  ;
                                   0000  ; IMPLICIT OUTPUTS:
                                   0000  ;       NONE
                                   0000  ;
                                   0000  ; ROUTINE VALUE:
                                   0000  ;       NONE
                                   0000  ;
                                   0000  ; SIDE EFFECTS:
                                   0000  ;       REQUEST QUEUED TO UCB
                                   0000  ;
                                   0000  ;--
                                   0000
                               00000000          .PSECT  $CODE$,NOWRT,LONG
                                   0000
                                   0000  REQUEUE_REQ::
                            003C   0000          .WORD   ^M<R2,R3,R4,R5>         ; SAVE REGISTERS
            53    04 AC       D0   0002          MOVL    PACKET(AP),R3           ; GET PACKET ADDRESS
          55    0000'CA       D0   0006          MOVL    W^CURRENT_UCB(R10),R5   ; GET UCB ADDRESS
            1C A3    55       D0   000B          MOVL    R5,IRP$L_UCB(R3)        ; STORE POSSIBLY CHANGED UCB ADDRESS
        50    0C AC    09     78   000F          ASHL    #9,UNMAPPED(AP),R0      ; GET BYTE COUNT OF UNMAPPED BLOCKS
                       12     13   0014          BEQL    10$                     ; BRANCH IF ALL MAPPED - NO FIXUP
            32 A3    50       C2   0016          SUBL    R0,IRP$L_BCNT(R3)       ; AND SUBTRACT FROM TRANSFER COUNT
   32 A3   000001FF 8F        C0   001A          ADDL    #511,IRP$L_BCNT(R3)     ; ROUND BYTE COUNT TO NEXT BLOCK BOUNDARY
       32 A3   01FF 8F        AA   0022          BICW    #511,IRP$L_BCNT(R3)     ; IN CASE FULL BYTE COUNT CONTAINS A PARTIAL
          50   00B4 C5        D0   0028  10$:    MOVL    UCB$L_MAXBCNT(R5),R0    ; R0 = 0 or Max. permissible BCNT.
                       05     12   002D          BNEQ    20$                     ; NEQ implies Max. permissible BCNT in R0.
          50   FE00 8F        3C   002F          MOVZWL  #512*127,R0             ; If 0, use default Max. permissible.
                                   0034  20$:
            32 A3    50       D1   0034          CMPL    R0,IRP$L_BCNT(R3)       ; See if BCNT too large.
                       04     1E   0038          BGEQU   30$                     ; GEQU impli^s we are OK.
            32 A3    50       D0   003A          MOVL    R0,IRP$L_BCNT(R3)       ; Else scale down to maximum allowed.
                                   003E  30$:
            50    08 AC       D0   003E          MOVL    LBN(AP),R0              ; GET STARTING LBN
          00000000'9F         16   0042          JSB     @#IOC$CVTLOGPHY         ; CONVERT TO PHYSICAL BLOCK
```

```
                00000000'9F  16  0048   132          JSB     @#EXE$INSIOQ            ; AND QUEUE TO DRIVER
                             04  004E   133          RET
                                 004F   134
                                 004F   135
                                 004F   136
                                 004F   137          .END
```

```
ACL_TYPE                = 00000007
AQB_TYPE                = 00000005
BITMAP_TYPE             = 00000001
CACHE_TYPE              = 00000006
CHIP_TYPE               = 00000008
CURRENT_UCB               ********    X   02
DATA_TYPE               = 00000004
DIRECTORY_TYPE          = 00000002
EXE$INSIOQ                ********    X   02
FCB_TYPE                = 00000000
HEADER_TYPE             = 00000000
INDEX_TYPE              = 00000003
IOC$CVTLOGPHY             ********    X   02
IRP$L_BCNT              = 00000032
IRP$L_UCB               = 0000001C
LBN                     = 00000008
MVL_TYPE                = 00000004
PACKET                  = 00000004
QUOTA_TYPE              = 00000005
REQUEUE_REQ               00000000 RG   02
RVT_TYPE                = 00000003
UCB$L_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
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
$CODE$	0000004F (79.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.52
Command processing	106	00:00:00.72	00:00:02.83
Pass 1	268	00:00:07.26	00:00:15.66
Symbol table sort	0	00:00:01.42	00:00:02.86
Pass 2	44	00:00:01.42	00:00:03.71
Symbol table output	4	00:00:00.03	00:00:00.17
Psect synopsis output	2	00:00:00.03	00:00:00.19
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	456	00:00:10.97	00:00:25.94

```
The working set limit was 1200 pages.
41385 bytes (81 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 875 non-local and 3 local symbols.
238 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                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$:FCPPRE/UPDATE=(ENH$:FCPPRE)+MSRC$:REQUEU/UPDATE=(ENH$:REQUEU)+EXECML$/LIB
```

RWVB
LIS

RDBLOK
LIS

REQUEU
LIS

RWATTR
LIS

REMOVE
LIS

RDHEDR
LIS

RETDIR
LIS