


```

CCCCCCCC      JJ FFFFFFFFFF      CCCCCCCC      LL      UU      UU      SSSSSSSS      TTTTTTTTTT      RRRRRRRR
CCCCCCCC      JJ FFFFFFFFFF      CCCCCCCC      LL      UU      UU      SSSSSSSS      TTTTTTTTTT      RRRRRRRR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FFFFFFFF      CC            LL      UU      UU      SSSSSS      TT          RRRRRRRR
CC            JJ FFFFFFFF      CC            LL      UU      UU      SSSSSS      TT          RRRRRRRR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CC            JJ FF            CC            LL      UU      UU      SS          TT          RR          RR
CCCCCCCC      JJJJJJ      CCCCCCCC      LLLLLLLLLL      UUUUUUUUUU      SSSSSSSS      TT          RR          RR
CCCCCCCC      JJJJJJ      CCCCCCCC      LLLLLLLLLL      UUUUUUUUUU      SSSSSSSS      TT          RR          RR

```

```

.L           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
LLLLLLLLLLL IIIIII      SSSSSSSS
LLLLLLLLLLL IIIIII      SSSSSSSS

```

```

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

```

(2)	46
(3)	83
(4)	125
(5)	161

DECLARATIONS
CJF\$MIN_JOURNAL - Minimum Journal Message Processor
CJF\$MIN_BUILD - Minimum CJF Message Build Routine
CJF\$DISPATCH - First Level CJF Dispatch Routine

```

0000 1 .TITLE CJFCLUSTER - Minimal Cluster CJF Support
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: EXECUTIVE, CLUSTER MANAGEMENT
0000 31 :
0000 32 : ABSTRACT:
0000 33 : This module contains the minimal Common Journaling Facility support
0000 34 : for a cluster environment.
0000 35 :
0000 36 : ENVIRONMENT: VAX/VMS
0000 37 :
0000 38 : AUTHOR: Ralph Weber, CREATION DATE: 31-March-1983
0000 39 :
0000 40 : MODIFIED BY:
0000 41 :
0000 42 :
0000 43 :--
0000 44 :

```

CL
V0

63
74

6C
69
6D

53
6D

66
20
72

58
62
65
6D

41
6D
75
73

72

```

0000 46      .SBTTL  DECLARATIONS
0000 47      :
0000 48      : INCLUDE FILES:
0000 49      :
0000 50      $CDRPDEF      ; CDRP offsets
0000 51      $CLSMGDEF     ; Cluster message offsets
0000 52      $CLUBDEF      ; Cluster block
0000 53      $CSBDEF       ; Cluster system block
0000 54      $IPLDEF       ; Processor IPL definitions
0000 55      $PDTDEF       ; Port descriptor table
0000 56      $$$DEF        ; System status codes
0000 57      :
0000 58      :
0000 59      : MACROS:
0000 60      :
0000 61      :
0000 62      :
0000 63      : EQUATED SYMBOLS:
0000 64      :
0000 65      :
0000 66      :
0000 67      : OWN STORAGE:
0000 68      :
0000 69      :
0000 70      :
0000 71      : *****
0000 72      :
0000 73      : NOTE: The following assumptions are in effect for this entire module.
0000 74      :
0000 75      : *****
0000 76      :
0000 77      ASSUME  IPL$_SYNCH  EQ  IPL$_SCS
0000 78      :
0000 79      .DEFAULT      DISPLACEMENT,WORD
0000 80      :
00000000 81      .PSECT  $$$100, LONG

```

CL
V0
69
41

6D
20
72
68

61
73

41
61
6F

20
73
74

6F
72

```

0000 83      .SBTTL  CJF$MIN_JOURNAL - Minimum Journal Message Processor
0000 84
0000 85      :++
0000 86      :
0000 87      : Functional Description:
0000 88      :
0000 89      :   When no real journal message dispatcher exists, this routine processes
0000 90      :   all incoming journaling messages.  If no response is required, the
0000 91      :   message is ignored.  If a response is required, a response with a
0000 92      :   status of $$$_NOSUCHNODE is returned.
0000 93      :
0000 94      : Inputs:
0000 95      :
0000 96      :   R2      Address of message
0000 97      :   R3      Senders CSB
0000 98      :   R4      Address of PDT
0000 99      :   R5      Address of a CDRP (if response required)
0000 100     :
0000 101     : Outputs:
0000 102     :
0000 103     :   CJMSG$Q_STATUS(R2) = $$$_NOSUCHNODE in a response message
0000 104     :--
0000 105     :
0000 106     CJF$MIN_JOURNAL::
0000 107
0000 108     TSTL  CLMSG$L_RSPID(R2)      ; Is a response required?
0003 109     BEQL  70$                ; Branch if no response.
0005 110
0005 111     BSBW  CNX$INIT_CDRP        ; Initialize supplied CDRP.
2C A5 09 A2 90 0008 112     MOVB  CLMSG$B_FUNC(R2), - ; Save incoming function code
                                CDRP$L_VAL1(R5) ; for the response.
4C A5 25 AF 9E 000D 113     MOVAB B^CJF$MIN_BUILD, - ; Set address of minimum build
                                CDRP$L_MSGBLD(R5) ; message routine in CDRP.
53 4C A3 D0 0012 114     MOVL  CSB$L_CSID(R3), R3 ; Get CSID from CSB.
                                CNX$SEND_MSG_RESP ; Send the response message.
50 55 D0 0016 115     MOVL  R5, R0 ; Setup CDRP for deallocation.
00000000 GF 17 001C 116     JMP   G^EXE$DEANONPAGED ; Deallocate CDRP and return.
                                ;
                                ; No response required.
                                ; Deallocate message buffer
                                ; and return to caller.
0022 120
0022 121
0022 122     70$: BRW  CNX$DEALL_MSG_BUF_CSB
0025 123

```

```

0025 125      .SBTTL  CJF$MIN_BUILD - Minimum CJF Message Build Routine
0025 126
0025 127      :++
0025 128      :
0025 129      : Functional Description:
0025 130      :
0025 131      : This routine is called by the acknowledged message services to build a
0025 132      : minimum CJF response message. It is called as many times as needed to
0025 133      : successfully transfer the message.
0025 134      :
0025 135      : Inputs:
0025 136      :
0025 137      : R2      Address of message
0025 138      : R3      Address of CSB
0025 139      : R4      Address of PDT
0025 140      : R5      Address of CDRP
0025 141      :
0025 142      : Outputs:
0025 143      :
0025 144      : Minimal CJF response message built in message buffer pointed to by R2.
0025 145      :
0025 146      :--
0025 147
0025 148 CJF$MIN_BUILD:
0025 149
0025 150      MOVB  #<CLSMMSG$M_RESPMSG -      ; Build a response CJF facility
002A 151      ! CLSMMSG$K_FAC CJF>, -      ; code.
002A 152      CLSMMSG$B FACILITY(R2)
002A 153      MOVB  CDRP$L_VAL1(R5), -      ; Return function code from
002F 154      CLSMMSG$B FUNC(R2)              ; incoming message.
0C A2 028C 8F 3C 002F 155      MOVZWL #SS$ NOSOCHNODE, -      ; Set error status which
0035 156      CJMSG$Q_STATUS(R2)          ; signals no journal driver
0035 157      CLRL  CJMSG$Q_STATUS+4(R2)    ; here.
0038 158
0038 159      RSB                          ; Return.

```

CL
V0

64
6E
20

65
75

69
69

68
65

69
2C

```
0039 161 .SBTTL CJF$DISPATCH - First Level CJF Dispatch Routine
0039 162
0039 163 :++
0039 164 :
0039 165 : Functional Description:
0039 166 :
0039 167 : This routine is the target of the first level received-new-message
0039 168 : dispatcher. In turn, this routine dispatches through the CLUB to the
0039 169 : second level dispatcher in the journal driver or to CJF$MIN_JOURNAL if
0039 170 : the journal driver is not yet loaded.
0039 171 :
0039 172 : Inputs:
0039 173 :
0039 174 : R2 Address of message
0039 175 : R3 Senders CSB
0039 176 : R4 Address of PDT
0039 177 :
0039 178 : Outputs:
0039 179 :
0039 180 : R0 Destroyed
0039 181 : R2 Address of message (unchanged)
0039 182 : R3 Senders CSB (unchanged)
0039 183 : R4 Address of PDT (unchanged)
0039 184 :
0039 185 :--
0039 186
0039 187 CJF$DISPATCH::
0039 188
50 00000000'GF 00 0039 189 MOVL G^CLUSGL CLUB, R0 ; Get CLUB address.
14 B0 17 0040 190 JMP @CLUB$JNL_DISPT(R0) ; Forward first level dispatch
0043 191 ; to journal driver or
0043 192 ; CJF$MIN_JOURNAL.
0043 193
0043 194 .END
```


CJFCLUSTER
Symbol table

- Minimal Cluster CJF Support

L 5

16-SEP-1984 00:23:05 VAX/VMS Macro V04-00
5-SEP-1984 04:06:52 [SYSLOA.SRC]CJFCLUSTER.MAR;1

Page 6
(5)

CL
VO

```

CDRPSL_MSGBLD      = 0000004C
CDRPSL_VAL1       = 0000002C
CJF$DISPATCH     = 00000039  RG   02
CJF$MIN_BUILD     = 00000025  R    02
CJF$MIN_JOURNAL   = 00000000  RG   02
CJMSG$Q_STATUS    = 0000000C
CLMSG$B_FACILITY  = 00000008
CLMSG$B_FUNC      = 00000009
CLMSG$K_FAC_CJF   = 00000003
CLMSG$L_RSPID     = 00000004
CLMSG$M_RESPMSG   = 00000080
CLUSGL_CCUB       = *****  X   02
CLUBSL_JNL_DISPT  = 00000014
CNX$DEALL_MSG_BUF_CSB = *****  X   02
CNX$INIT_CDRP     = *****  X   02
CNX$SEND_MSG_RESP = *****  X   02
CSB$L_CSTD        = 0000004C
EXE$DEANONPAGED  = *****  X   02
IPL$SCS          = 00000008
IPL$SYNCH        = 00000008
SS$NOSUCHNODE    = 0000028C
  
```

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS	0000000U (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
\$\$\$100	00000043 (67.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	36	00:00:00.06	00:00:00.41
Command processing	146	00:00:00.46	00:00:02.62
Pass 1	341	00:00:07.18	00:00:24.24
Symbol table sort	2	00:00:01.31	00:00:05.72
Pass 2	51	00:00:01.21	00:00:04.67
Symbol table output	4	00:00:00.03	00:00:00.03
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	584	00:00:10.27	00:00:37.72

The working set limit was 1500 pages.
59540 bytes (117 pages) of virtual memory were used to buffer the intermediate code.
There were 70 pages of symbol table space allocated to hold 1249 non-local and 1 local symbols.
194 source lines were read in Pass 1, producing 13 object records in Pass 2.
15 pages of virtual memory were used to define 14 macros.

04

! Macro library statistics !

Macro library name	Macros defined
-----	-----
-\$255\$DUA28:[SYSLOA.OBJ]CLUSTER.MLB;1	1
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	5
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	5
TOTALS (all libraries)	11

1335 GETS were required to define 11 macros.

There were no errors, warnings or information messages.

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

