


```

LL          000000  CCCCCCCC  KK      KK  DDDDDDDD  BBBB88888
LL          000000  CCCCCCCC  KK      KK  DDDDDDDD  BBBB88888
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KKKKKK  DD          DD  BBBB88888
LL          00      00  CC          KKKKKK  DD          DD  BBBB88888
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LL          00      00  CC          KK      KK  DD          DD  BB      BB
LLLLLLLLLLL 000000  CCCCCCCC  KK      KK  DDDDDDDD  BBBB88888
LLLLLLLLLLL 000000  CCCCCCCC  KK      KK  DDDDDJDD  BBBB88888
                                     ....
                                     ....
                                     ....
                                     ....

```

```

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
LLLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLLL IIIIII  SSSSSSSS

```



```

0000 54
0000 55 :++
0000 56 : LOCK_IODB -- Routine to lock the I/O data base Mutex.
0000 57 :
0000 58 : Calling Sequence:
0000 59 :
0000 60 :     CALL LOCK_IODB ()
0000 61 :
0000 62 : Input Parameters:
0000 63 :
0000 64 :     none
0000 65 :
0000 66 : Implicit Inputs:
0000 67 :
0000 68 :     none
0000 69 :
0000 70 : Output Parameters:
0000 71 :
0000 72 :     none
0000 73 :
0000 74 : Implicit Outputs:
0000 75 :
0000 76 :     none
0000 77 :
0000 78 : Routine Value:
0000 79 :
0000 80 :     none
0000 81 :
0000 82 : Side Effects:
0000 83 :
0000 84 :     !/O data base mutex locked.
0000 85 :
0000 86 : --
0000 87 :
00000000 88 : .PSECT $LOCKEDC1$,NOWRT
0000 89 :
0000 90 LOCK_IODB::
50 00000000'9F 003C 0000 91 : .WORD ^M<R2,R3,R4,R5> ; save registers
54 00000000'9F DE 0002 92 : MOVAL @#IOC$GL_MUTEX,R0 ; get i/o data base mutex
00000000'9F D0 0009 93 : MOVL @#SCH$GL_CURPCB,R4 ; get own pcb address
00000000'9F 16 0010 94 : JSB @#SCH$LOCKW ; and lock it
0000 95 : RET

```

```

0017 97
0017 98 :++
0017 99 : UNLOCK_IODB -- routine unlocks the i/o data base mutex.
0017 100 :
0017 101 : Calling sequence:
0017 102 :
0017 103 :     CALL UNLOCK_IODB ()
0017 104 :
0017 105 : Input Parameters:
0017 106 :
0017 107 :     none
0017 108 :
0017 109 : Implicit Inputs:
0017 110 :
0017 111 :     none
0017 112 :
0017 113 : Output Parameters:
0017 114 :
0017 115 :     none
0017 116 :
0017 117 : Implicit Outputs:
0017 118 :
0017 119 :     none
0017 120 :
0017 121 : Routine Value:
0017 122 :
0017 123 :     none
0017 124 :
0017 125 : Side Effects:
0017 126 :
0017 127 :     I/o data base mutex unlocked.
0017 128 :     IPL lowered to 0
0017 129 :
0017 130 :--
0017 131 :
00000017 132 : .PSECT $LOCKEDC1$,NOWRT
0017 133 :
0017 134 UNLOCK_IODB::
003C 0017 135 : .WORD ^M<R2,R3,R4,R5> : save registers
50 00000000'9F DE 0019 136 : MOVAL @#IOC$GL_MUTEX,R0 : get i/o data base mutex
54 00000000'9F D0 0020 137 : MOVL @#SCH$GL_CURPCB,R4 : and own pcb address
00000000'9F 16 0027 138 : JSB @#SCH$UNLOCK : and unlock it
002D 002D 139 : SETIPL #0 : also lower ipl
04 0030 140 : RET
0031 141 :
0031 142 :
0031 143 :
0031 144 : .END

```

LOCKDB
Symbol table

- LOCK AND UNLOCK I/O DATA BASE

F 4

16-SEP-1984 02:05:16
5-SEP-1984 02:11:58

VAX/VMS Macro V04-00
[MTAACP.SRC]LOCKDB.MAR;1

Page 4
(5)

```

AQB_TYPE      = 00000005
FCB_TYPE      = 00000000
IOCSGL_MUTEX  ***** X 02
LOCK_IODB     00000000 RG 02
MVL_TYPE      = 00000004
PRS_IPL       = 00000012
RVT_TYPE      = 00000003
SCH$GL_CURPCB ***** X 02
SCH$LOCKW     ***** X 02
SCH$UNLOCK    ***** X 02
UNLOCK_IODB   00000017 RG 02
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
\$LOCKEDC1\$	00000031 (49.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

```

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

```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	39	00:00:00.09	00:00:01.66
Command processing	131	00:00:00.66	00:00:04.85
Pass 1	144	00:00:01.59	00:00:07.61
Symbol table sort	0	00:00:00.08	00:00:00.44
Pass 2	40	00:00:00.58	00:00:03.71
Symbol table output	3	00:00:00.02	00:00:00.02
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	361	00:00:03.05	00:00:18.32

The working set limit was 1050 pages.
6752 bytes (14 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 92 non-local and 0 local symbols.
327 source lines were read in Pass 1, producing 13 object records in Pass 2.
16 pages of virtual memory were used to define 14 macros.

```

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

```

Macro library name	Macros defined
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	1
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	5

146 GETS were required to define 5 macros.

LOG
V04

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:LOCKDB/OBJ=OBJ\$:LOCKDB MSRC\$:MTADEF1/UPDATE=(ENH\$:MTADEF1)+MSRC\$:LOCKDB/UPDATE=(ENH\$:LOCKDB)+EXECMLS/LIB

