```
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFFFFFFFFFFFFFF     000000000     RRRRRRRRRRRR     RRRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFF               000         000 RRR         RRR RRR         RRR       TTT        LLL
FFFFFFFFFFF       000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFF       000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFFFFFFFFFF       000         000 RRRRRRRRRRRR    RRRRRRRRRRRR          TTT        LLL
FFF               000         000 RRR   RRR       RRR   RRR             TTT        LLL
FFF               000         000 RRR   RRR       RRR   RRR             TTT        LLL
FFF               000         000 RRR   RRR       RRR   RRR             TTT        LLL
FFF               000         000 RRR     RRR     RRR     RRR           TTT        LLL
FFF               000         000 RRR     RRR     RRR     RRR           TTT        LLL
FFF               000         000 RRR     RRR     RRR     RRR           TTT        LLL
FFF                 000000000     RRR         RRR RRR         RRR       TTT        LLLLLLLLLLLLLLL
FFF                 000000000     RRR         RRR RRR         RRR       TTT        LLLLLLLLLLLLLLL
FFF                 000000000     RRR         RRR RRR         RRR       TTT        LLLLLLLLLLLLLLL
```

```
FFFFFFFFFF    000000    RRRRRRRR    WW      WW  RRRRRRRR     IIIIII    TTTTTTTTTT    SSSSSSSS  NN      NN
FFFFFFFFFF    000000    RRRRRRRR    WW      WW  RRRRRRRR     IIIIII    TTTTTTTTTT    SSSSSSSS  NN      NN
FF          00      00  RR      RR  WW      WW  RR      RR     II        TT        SS          NN      NN
FF          00      00  RR      RR  WW      WW  RR      RR     II        TT        SS          NN      NN
FF          00      00  RR      RR  WW      WW  RR      RR     II        TT        SS          NNNN    NN
FF          00      00  RR      RR  WW      WW  RR      RR     II        TT        SS          NNNN    NN
FFFFFFFF    00      00  RRRRRRRR    WW      WW  RRRRRRRR     II        TT          SSSSSS    NN  NN  NN
FFFFFFFF    00      00  RRRRRRRR    WW      WW  RRRRRRRR     II        TT          SSSSSS    NN  NN  NN
FF          00      00  RR  RR      WW  WW  WW  RR  RR       II        TT                SS  NN    NNNN
FF          00      00  RR  RR      WW  WW  WW  RR  RR       II        TT                SS  NN    NNNN
FF          00      00  RR    RR    WWWW  WWWW  RR    RR     II        TT                SS  NN      NN      ....
FF          00      00  RR    RR    WWWW  WWWW  RR    RR     II        TT                SS  NN      NN      ....
FF            000000    RR      RR  WW      WW  RR      RR   IIIIII      TT        SSSSSSSS    NN      NN      ....
FF            000000    RR      RR  WW      WW  RR      RR   IIIIII      TT        SSSSSSSS    NN      NN      ....

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
```

Table of contents

```
0000     1          .TITLE  FOR$WRITE_SN - FORTRAN WRITE Sequential NAMELIST
0000     2          .IDENT  /1-002/                 ; File: FORWRITSN.MAR, Edit: SBL1002
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: FORTRAN Language Support
0000    31  ;
0000    32  ; ABSTRACT:
0000    33  ;
0000    34  ;       This module contains FOR$WRITE_SN, the user interface to the FORTRAN
0000    35  ;       I/O Run-Time Library which begins a WRITE NAMELIST statement.
0000    36  ;
0000    37  ; ENVIRONMENT: Runs at any access mode, AST Reentrant
0000    38  ;
0000    39  ; MAINTENANCE NOTE:
0000    40  ;       The transfer vector (RTLVECTOR+ALLGBL) must have the following:
0000    41  ;
0000    42  ;       .TRANSFER       FOR$WRITE_SN
0000    43  ;       .MASK           FOR$$IO_BEG
0000    44  ;       JMP             FOR$WRITE_SN+2
0000    45  ;
0000    46  ;       This puts the correct mask in entry vector, that is FOR$$IO_BEG entry mask.
0000    47  ;       Furthermore this module must only use R0 and R1
0000    48  ;       since any other register might not be in the entry mask for FOR$$IO_BEG.
0000    49  ;
0000    50  ; AUTHOR: Steven B. Lionel, CREATION DATE: 21-July-1980
0000    51  ;
0000    52  ; MODIFIED BY:
0000    53  ;
0000    54  ; 1-001 - Original. SBL 21-July-1980
0000    55  ; 1-002 - Second colon after entry point name accidentally left off; replace.
0000    56  ;         SBL 11-Nov-1980
0000    57  ;--
```

```
                                J 10
0000     59          .SBTTL  DECLARATIONS
0000     60 ;
0000     61 ; LIBRARY MACRO CALLS:
0000     62 ;
0000     63          $ISBDEF                 ; Intra-statement block definitions
0000     64 ;
0000     65 ; EXTERNAL DECLARATIONS:
0000     66 ;
0000     67          .DSABL  GBL             ; Force all external symbols to be declared
0000     68          .EXTRN  FOR$IO_BEG      ; Common initialization routine
0000     69          .EXTRN  FOR$UDF_WN0, FOR$UDF_WN9      ; To pull in weak references
0000     70          .EXTRN  FOR$REC_WSN0, FOR$REC_WSN1    ; To pull in weak references
0000     71
0000     72 ;
0000     73 ; MACROS:
0000     74 ;
0000     75 ;        NONE
0000     76 ;
0000     77 ; EQUATED SYMBOLS:
0000     78 ;
0000     79 ;        NONE
0000     80 ;
0000     81 ; OWN STORAGE:
0000     82 ;
0000     83 ;        NONE
0000     84 ;
0000     85 ; PSECT DECLARATIONS:
0000     86 ;
00000000 87          .PSECT _FOR$CODE PIC, USR, CON, REL, LCL, SHR, -
0000     88                      EXE, RD, NOWRT, LONG
0000     89
```

```
                     0000     91          .SBTTL  FOR$WRITE_SN - Write sequential NAMELIST
                     0000     92 ;++
                     0000     93 ; FUNCTIONAL DESCRIPTION:
                     0000     94 ;
                     0000     95 ;       This routine begins a FORTRAN WRITE NAMELIST statement.
                     0000     96 ;
                     0000     97 ; CALLING SEQUENCE:
                     0000     98 ;
                     0000     99 ;       CALL FOR$WRITE_SN (unit.rl.v, namelist.rlu.ra [, err_eql.j.r])
                     0000    100 ;
                     0000    101 ; FORMAL PARAMETERS:
                     0000    102 ;
                     0000    103 ;       unit            - logical unit number
                     0000    104 ;       namelist        - address of namelist descriptor block (see below)
                     0000    105 ;       err_eql         - address of instruction to branch to if an error occurs
                     0000    106 ;
                     0000    107 ; IMPLICIT INPUTS:
                     0000    108 ;
                     0000    109 ;       NONE
                     0000    110 ;
                     0000    111 ; IMPLICIT OUTPUTS:
                     0000    112 ;
                     0000    113 ;       NONE
                     0000    114 ;
                     0000    115 ; COMPLETION STATUS:
                     0000    116 ;
                     0000    117 ;       NONE
                     0000    118 ;
                     0000    119 ; SIDE EFFECTS:
                     0000    120 ;
                     0000    121 ;       One WRITE NAMELIST statement will be executed.
                     0000    122 ;
                     0000    123 ;--
                     0000    124
                     0000    125 FOR$WRITE_SN::
             0000'   0000    126          .MASK   FOR$$IO_BEG
          50   13     9A   0002    127          MOVZBL  #ISB$K_ST_TY_WSN, R0    ; Move statement type
00000002'GF    17     17   0005    128          JMP     G^FOR$$IO_BEG+2         ; Jump to FOR$$IO_BEG
                            000B    129
                            000B    130          .END                            ; End of module FOR$WRITE_SN
```

```
FOR$$IO_BEG                     ********    X   00
FOR$$REC_WSNO                   ********    X   00
FOR$$REC_WSN1                   ********    X   00
FOR$$UDF_WNO                    ********    X   00
FOR$$UDF_WN9                    ********    X   00
FOR$WRITE_SN                    00000000 RG     01
ISB$K_ST_TY_WSN                 = 00000013
```

Psect synopsis

```
PSECT name                      Allocation          PSECT No.  Attributes
----------                      ----------          ---------  ----------
.  ABS  .                       00000000  (    0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD  NOWRT NOVEC BYTE
_FOR$CODE                       0000000B  (   11.)  01 (  1.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD  NOWRT NOVEC LONG
```

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	37	00:00:00.09	00:00:00.91
Command processing	121	00:00:00.50	00:00:04.09
Pass 1	122	00:00:01.13	00:00:04.89
Symbol table sort	0	00:00:00.17	00:00:00.47
Pass 2	38	00:00:00.39	00:00:01.81
Symbol table output	2	00:00:00.02	00:00:00.02
Psect synopsis output	2	00:00:00.02	00:00:00.08
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	324	00:00:02.32	00:00:12.27

The working set limit was 1050 pages.
5742 bytes (12 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 168 non-local and 0 local symbols.
130 source lines were read in Pass 1, producing 8 object records in Pass 2.
8 pages of virtual memory were used to define 1 macro.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[FORRTL.OBJ]FORRTL.MLB;1	1
_255DUA28:[SYSLIB]STARLET.MLB;2	0
TOTALS (all libraries)	1

164 GETS were required to define 1 macros.

There were no errors, warnings or information messages.

```
MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:FORWRITSN/OBJ=OBJ$:FORWRITSN MSRC$:FORWRITSN/UPDATE=(ENH$:FORWRITSN)+LI
```

HELP

FORWRITDF
LIS

VMSHELP
MAP

HLD
MDL

FORWRITDU
LIS

FORVECTOR
LIS

FORWRITIL
LIS

FORWRITSF
LIS

HLD

HLDMACROS
MAR

FORVM
LIS

FORWRITSN
LIS

HLD
MAP

FORWRITSU
LIS

RSXDEFS
MAR

FORUNLOCK
LIS

FORWRITDO
LIS

HELP
LIS

FORWRITIF
LIS

FORWRITIO
LIS

HLDCSTA
LIS

FORUDFWU
LIS

FORUNDERF
LIS

FORWRITSL
LIS

FORWRITSO
LIS