```
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    RRRRRRRR    EEEEEEEEEE    AAAAAA    DDDDDDDD    DDDDDDDD      000000
FFFFFFFFFF    000000    RRRRRRRR    RRRRRRRR    EEEEEEEEEE    AAAAAA    DDDDDDDD    DDDDDDDD      000000
FF          00      00  RR      RR  RR      RR  EE          AA      AA  DD      DD  DD      DD  00      00
FF          00      00  RR      RR  RR      RR  EE          AA      AA  DD      DD  DD      DD  00      00
FF          00      00  RR      RR  RR      RR  EE          AA      AA  DD      DD  DD      DD  00      00
FF          00      00  RR      RR  RR      RR  EE          AA      AA  DD      DD  DD      DD  00      00
FFFFFFFF    00      00  RRRRRRRR    RRRRRRRR    EEEEEEEE    AA      AA  DD      DD  DD      DD  00      00
FFFFFFFF    00      00  RRRRRRRR    RRRRRRRR    EEEEEEEE    AA      AA  DD      DD  DD      DD  00      00
FF          00      00  RR  RR      RR  RR      EE          AAAAAAAAAA  DD      DD  DD      DD  00      00
FF          00      00  RR  RR      RR  RR      EE          AAAAAAAAAA  DD      DD  DD      DD  00      00
FF          00      00  RR    RR    RR    RR    EE          AA      AA  DD      DD  DD      DD  00      00      ....
FF          00      00  RR    RR    RR    RR    EE          AA      AA  DD      DD  DD      DD  00      00      ....
FF            000000    RR      RR  RR      RR  EEEEEEEEEE  AA      AA  DDDDDDDD    DDDDDDDD      000000      ....
FF            000000    RR      RR  RR      RR  EEEEEEEEEE  AA      AA  DDDDDDDD    DDDDDDDD      000000      ....

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  FOR$READ_DO - entry point for FORTRAN READ DIRECT OBJECT-FORMATTED
0000     2          .IDENT  /1-012/         File: FORREADDO.MAR  Edit: JAW1012
0000     3 ;
0000     4 ;*****************************************************************************
0000     5 ;*                                                                           *
0000     6 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     7 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     8 ;*  ALL RIGHTS RESERVED.                                                     *
0000     9 ;*                                                                           *
0000    10 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    11 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    12 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    13 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    14 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    15 ;*  TRANSFERRED.                                                             *
0000    16 ;*                                                                           *
0000    17 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    18 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000    19 ;*  CORPORATION.                                                             *
0000    20 ;*                                                                           *
0000    21 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    22 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    23 ;*                                                                           *
0000    24 ;*                                                                           *
0000    25 ;*****************************************************************************
0000    26 ;
0000    27 ;
0000    28 ;++
0000    29 ; FACILITY: FORTRAN Support Library - user callable
0000    30 ;
0000    31 ; ABSTRACT:
0000    32 ;
0000    33 ;       This module contains the entry point for the FORTRAN
0000    34 ;       READ DIRECT OBJECT-FORMATTED I/O statement.  It is simply
0000    35 ;       a call to FOR$$IO_BEG with bits in R0 which describe the
0000    36 ;       parameter list.  FOR$$IO_BEG interprets the parameters.
0000    37 ;
0000    38 ; MAINTENANCE NOTE:
0000    39 ;       The transfer vector (RTLVECTOR+ALLGBL) must have the following:
0000    40 ;
0000    41 ;       .TRANSFER       FOR$READ_DO
0000    42 ;       .MASK           FOR$$IO_BEG
0000    43 ;       BRW             FOR$READ_DO+2
0000    44 ;
0000    45 ;       This puts the correct mask in entry vector, that is FOR$$IO_BEG entry mask.
0000    46 ;       Furthermore this module must only use R0 and R1
0000    47 ;       since any other register might not be in the entry mask for FOR$$IO_BEG.
0000    48 ;
0000    49 ; ENVIRONMENT: User access mode; mixture of AST level or not
0000    50 ;
0000    51 ; AUTHOR:       Richard B. Grove, CREATION DATE: 28-May-78
0000    52 ;
0000    53 ; MODIFIED BY:
0000    54 ;       T. Hastings, 29-July-78
```

```
0000     56          .SBTTL  HISTORY         ; Detailed Current Edit History
0000     57
0000     58
0000     59 ; Edit History for Version 1
0000     60 ;
0000     61 ; 0-10  - Add comment about vectors.  TNH 23-June-78
0000     62 ; 0-12  - Pass arg in R0, not R0R, add comments.  TNH 29-July-78
0000     63 ; 1-001 - Update version number and copyright notice.  JBS 16-NOV-78
0000     64 ; 1-002 - Change statement type symbols to be LUB$K...  JBS 07-DEC-78
0000     65 ; 1-003 - Change statement type symbols to be ISB$K...  JBS 11-DEC-78
0000     66 ; 1-004 - Add "-" to the PSECT directive.  JBS 22-DEC-78
0000     67 ; 1-005 - Add FOR$READ_KF, FOR$READ_KO, FOR$REWRITE_SF, FOR$REWRITE_SO,
0000     68 ;         FOR$READ_IF, FOR$READ_IO, FOR$WRITE_IF, FOR$WRITE_IO,
0000     69 ;         FOR$READ_KU, FOR$REWRITE_SU,
0000     70 ;         SBL 2-May-1979
0000     71 ; 1-006 - Remove all entry points that need object time formatting,
0000     72 ;         putting them in FOR$ENTRY_OBJ so that we can arrange to
0000     73 ;         load the format compiler only when it is needed.
0000     74 ;         JBS 26-JUN-1979
0000     75 ; 1-007 - Remove entry point FOR$ENCODE_MF; we will code a new module
0000     76 ;         for it and FOR$$IO_BEG, to see how much I/O initiation time
0000     77 ;         improves.  JBS 02-JUL-1979
0000     78 ; 1-008 - Do likewuse for FOR$READ_DU and FOR$WRITE_DU.  JBS 03-JUL-1979
0000     79 ; 1-009 - Remove all entry points and add FOR$READ_DO;  each entry
0000     80 ;         point gets its own module do we can selectively load
0000     81 ;         the necessary UDF and REC modules.  JBS 09-JUL-1979
0000     82 ; 1-010 - Correct some typos in the references to REC and UDF levels.
0000     83 ;         JBS 12-JUL-1979
0000     84 ; 1-011 - New parameter format for FOR$$IO_BEG.  SBL 5-Dec-1979
0000     85 ; 1-012 - Change BRW FOR$$IO_BEG+2 to JMP G^FOR$$IO_BEG+2.  JAW 21-Feb-1981
```

```
                        .SBTTL  DECLARATIONS

;
; INCLUDE FILES:
;

                        $FORPAR                         ; Define inter-module FORTRAN symbols
                        $ISBDEF                         ; Define statement type symbols

;
; EXTERNAL SYMBOLS:
;

                        .DSABL  GBL                     ; Declare all external symbols
                        .EXTRN  FOR$$IO_BEG             ; common I/O statement processing
;+
; The following references are to make sure the necessary UDF and REC
; modules are loaded.  These are the routines which are called through
; the dispatch tables in FOR$$DISPAT.
;-
                        .EXTRN  FOR$$UDF_RF0, FOR$$UDF_RF1, FOR$$UDF_RF9
                        .EXTRN  FOR$$REC_RD0, FOR$$REC_RD1, FOR$$REC_RD9
;+
; The following reference makes sure the format compiler is loaded.
;-
                        .EXTRN  FOR$$FMT_COMPIL

;
; MACROS:
;
;       NONE
;
; PSECT DECLARATIONS:
;

00000000                .PSECT  _FOR$CODE PIC,USR,CON,REL,LCL,SHR,EXE,RD,NOWRT,LONG

;
; EQUATED SYMBOLS:
;


;
; OWN STORAGE:
;
;       NONE
;
```

```
                        0000   135          .SBTTL  FOR$READ_DO - READ DIRECT OBJECT-FORMATTED
                        0000   136
                        0000   137 ;++
                        0000   138 ; FUNCTIONAL DESCRIPTION:
                        0000   139 ;
                        0000   140 ;       Initialize the FORTRAN I/O system to perform
                        0000   141 ;       a READ DIRECT OBJECT-FORMATTED I/O statement.
                        0000   142 ;
                        0000   143 ; CALLING SEQUENCE:
                        0000   144 ;
                        0000   145 ;       CALL FOR$READ_DO (unit.rl.v, record_no.rl.v, format_adr.rt.r
                        0000   146 ;               [, err_adr.j.r [, end_adr.j.r]])
                        0000   147 ;
                        0000   148 ; INPUT PARAMETERS:
                        0000   149 ;
                        0000   150 ;       unit.rl.v               logical unit number
                        0000   151 ;       record_no.rl.v          record number to read
                        0000   152 ;       format_adr.rt.r         format string (needs compilation)
                        0000   153 ;       [err_adr.j.r]           optional ERR= address
                        0000   154 ;       [end_adr.j.r]           optional END= address
                        0000   155 ;
                        0000   156 ; IMPLICIT INPUTS:
                        0000   157 ;
                        0000   158 ;       NONE except those used by FOR$$IO_BEG.
                        0000   159 ;
                        0000   160 ; OUTPUT PARAMETERS:
                        0000   161 ;
                        0000   162 ;       NONE
                        0000   163 ;
                        0000   164 ; IMPLICIT OUTPUTS:
                        0000   165 ;
                        0000   166 ;       NONE except those left by FOR$$IO_BEG.
                        0000   167 ;
                        0000   168 ; COMPLETION CODES:
                        0000   169 ;
                        0000   170 ;       NONE
                        0000   171 ;
                        0000   172 ; SIDE EFFECTS:
                        0000   173 ;
                        0000   174 ;       NONE except those of FOR$$IO_BEG.
                        0000   175 ;
                        0000   176 ;--
                        0000   177
                  0000' 0000   178 FOR$READ_DO::   .MASK   FOR$$IO_BEG
50    0106 8F     3C    0002   179          MOVZWL  #ISB$K_ST_TY_RDF+       -
                        0007   180                  <1@FOR$V_OBJ_FMT>, R0   ; Statement type
 00000002'GF      17    0007   181          JMP     G^FOR$$IO_BEG+2         ; branch past call mask
                        000D   182
                        000D   183
                        000D   184          .END
```

```
FOR$$FMT_COMPIL                 ********    X   00
FOR$$IO_BEG                     ********    X   00
FOR$$REC_RD0                    ********    X   00
FOR$$REC_RD1                    ********    X   00
FOR$$REC_RD9                    ********    X   00
FOR$$UDF_RF0                    ********    X   00
FOR$$UDF_RF1                    ********    X   00
FOR$$UDF_RF9                    ********    X   00
FOR$READ_DO                     00000000 RG     01
FOR$V_OBJ_FMT                 = 00000006
ISB$K_ST_TY_RDF               = 00000006
```

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                       0000000D  (   13.)  01 (  1.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD  NOWRT NOVEC LONG
```

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.07	00:00:01.25
Command processing	117	00:00:00.57	00:00:03.63
Pass 1	123	00:00:01.33	00:00:03.87
Symbol table sort	0	00:00:00.20	00:00:00.22
Pass 2	46	00:00:00.53	00:00:02.43
Symbol table output	3	00:00:00.04	00:00:00.14
Psect synopsis output	2	00:00:00.02	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	323	00:00:02.77	00:00:11.57

```
The working set limit was 1050 pages.
6727 bytes (14 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 188 non-local and 0 local symbols.
184 source lines were read in Pass 1, producing 8 object records in Pass 2.
9 pages of virtual memory were used to define 2 macros.
```

Macro library statistics

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

```
183 GETS were required to define 2 macros.

There were no errors, warnings or information messages.
```

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

FORRAB
LIS

FORREADDF
LIS

FOROPNKEY
LIS

FORPAUSE
LIS

FORRANDOM
LIS

FOROPEN
LIS

FOROPENDE
LIS

FORREADDO
LIS

FORREADIF LIS
FORREADIO LIS
FORRECPRO LIS
FORREWRSU LIS
FORSIGNAL LIS
FORREADKO LIS
FORREWIND LIS
FORREADSF LIS
FORREWRSO LIS
FORREADSN LIS
FORREADDU LIS
FORREADSU LIS
FORSECNDS LIS
FORREADIL LIS
FORREADKF LIS
FORREADKU LIS
FORREWRSF LIS
FORREADSL LIS
FORREADSO LIS