DCL

```
IIIIII     NN      NN  TTTTTTTTTT  DDDDDDD   EEEEEEEEEE  FFFFFFFFFF
IIIIII     NN      NN  TTTTTTTTTT  DDDDDDD   EEEEEEEEEE  FFFFFFFFFF
  II       NN      NN      TT      DD    DD  EE          FF
  II       NN      NN      TT      DD    DD  EE          FF
  II       NNNN    NN      TT      DD    DD  EE          FF
  II       NNNN    NN      TT      DD    DD  EE          FF
  II       NN NN   NN      TT      DD    DD  EEEEEEE     FFFFFFF
  II       NN  NN  NN      TT      DD    DD  EEEEEEE     FFFFFFF
  II       NN   NNNN       TT      DD    DD  EE          FF
  II       NN   NNNN       TT      DD    DD  EE          FF
  II       NN      NN      TT      DD    DD  EE          FF
  II       NN      NN      TT      DD    DD  EE          FF
IIIIII     NN      NN      TT      DDDDDDD   EEEEEEEEEE  FF
IIIIII     NN      NN      TT      DDDDDDD   EEEEEEEEEE  FF
                                                                    ....
                                                                    ....
```

```
SSSSSSSS   DDDDDDD   LL
SSSSSSSS   DDDDDDD   LL
SS         DD    DD  LL
SS         DD    DD  LL
SS         DD    DD  LL
SS         DD    DD  LL
 SSSSSS    DD    DD  LL
 SSSSSS    DD    DD  LL
     SS    DD    DD  LL
     SS    DD    DD  LL
     SS    DD    DD  LL
     SS    DD    DD  LL
SSSSSSSS   DDDDDDD   LLLLLLLLLL
SSSSSSSS   DDDDDDD   LLLLLLLLLL
```

```
{
{       Command language interpreter interface definitions
{
{ IDENT 'V04-000'
{

{
{*****************************************************************************
{*                                                                          *
{*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                *
{*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
{*   ALL RIGHTS RESERVED.                                                   *
{*                                                                          *
{*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
{*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH LICENSE  AND WITH THE   *
{*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
{*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
{*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
{*   TRANSFERRED.                                                           *
{*                                                                          *
{*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
{*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
{*   CORPORATION.                                                           *
{*                                                                          *
{*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
{*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
{*                                                                          *
{*                                                                          *
{*****************************************************************************
{

{++
{ FACILITY:  DCL & MCR Command language interpreters
{
{ ABSTRACT:
{
{       These are the interface data structure definitions
{       which describe the way the CLI$INTERFACE routines use
{       the request descriptor blocks.
{
{ ENVIRONMENT:
{
{       VAX/VMS operating system. supervisor mode.
{
{ AUTHOR:  Peter George, Feb 1983
{
{ Modified by:
{
{--
```

```
module $INT;

aggregate INTDEF structure fill prefix INT_;
    TYPE byte unsigned;                             /* TYPE OF REQUEST
    FILL_1 byte dimension 3 fill prefix INTDEF tag $$;  /* SPARE
    TABLES longword unsigned;                       /* ADDRESS OF COMMAND TABLES
    ENTLEN word unsigned;                           /* LENGTH OF ENTITY NAME
    FILL_2 word fill prefix INTDEF tag $$;          /* SPARE
    ENTADDR longword unsigned;                      /* ADDRESS OF ENTITY NAME
end INTDEF;

aggregate INTDEF1 structure fill prefix INT_;
    FILL_5 byte dimension 8 fill prefix INTDEF tag $$;
    RETLEN word unsigned;                           /* LENGTH OF RETURN VALUE
    FILL_3 word fill prefix INTDEF tag $$;          /* SPARE
    RETADDR longword unsigned;                      /* ADDRESS OF RETURN VALUE
    GETVM longword unsigned;                         /* ADDRESS OF LIB$GET_VM ROUTINE
    FREEVM longword unsigned;                        /* ADDRESS OF LIB$FREE_VM ROUTINE
    LIST longword unsigned;                          /* ADDRESS OF AUXILIARY ARGUMENT LIST
    constant "LENGTH" equals . prefix INT_ tag K;
    constant "LENGTH" equals . prefix INT_ tag C;
end INTDEF1;

aggregate INTDEF2 structure fill prefix INT_;
    LISTLEN longword unsigned;                       /* LENGTH OF AUXILIARY ARGUMENT LIST
    PROMPTRTN longword unsigned;                      /* ADDRESS OF PROMPT ROUTINE
    CONTINRTN longword unsigned;                      /* ADDRESS OF CONTINUATION ROUTINE
    PMPTLEN word unsigned;                            /* LENGTH OF PROMPT STRING
    FILL_4 word fill prefix INTDEF tag $$;           /* SPARE
    PMPTADDR longword unsigned;                       /* ADDRESS OF PROMPT STRING
end INTDEF2;

end_module $INT;
```

INTDEF
SDL

COMMAND
LIS

DESCRVAL
LIS

CONVERT
LIS

CLIMAC
MAR

CLIMSG
LIS

CONNECT
LIS

DCLPARSE
LIS

CHARMANIP
LIS

EXAMDEP
LIS

EXIT
LIS

INTIMAGES
MAR

DCXSTART
LIS

CLIGBL
LIS

DISALLOW
LIS

CLINT
LIS

CANCEL
LIS