Laser-Scan Ltd.

CONVERT PACKAGE

IFFMOSS Reference

Issue 1.0 - 17-June-1991

CONTENTS

IFFMOSS - Change Record

Version 1.0   S Townrow   17-June-1991

      Module IFFMOSS   - Reorganised package documentation.

PREFACE

Intended audience

      This manual is intended for users of a specific utility of the Laser-Scan CONVERT package running under the VAX/VMS operating system. Each manual contains the documentation for a particular CONVERT utility and a site will only receive new or updated documentation for those utilities which they have purchased.

Structure of this document

      This document is composed of 2 major sections.

      The Introduction is an overview of the CONVERT package and its purpose.

      There then follow the User Guides for the individual modules which comprise CONVERT. Each individual module contains the same basic categories of information. These are:

MODULE   - the name of the CONVERT module.

FORMAT DESCRIPTION   - a description of the data format written or
                        read by the utility programs in this conversion
                        module.

DATA PREPARATION   - guidance on how to digitise or prepare the IFF
                        and other data required by the utility programs
                        in this module.

      For each utility program in the module, there will then be the following categories:

UTILITY   - the name of the utility.

FUNCTION   - a synopsis of what the utility does.

FORMAT   - a summary of the utility command format
                        and command qualifiers. Default qualifier
                        settings are indicated.

PROMPT   - how it prompts the user.

PARAMETERS   - description of expected command parameters.

COMMAND QUALIFIERS   - description of all command qualifiers.
                        Qualifiers are ordered alphabetically and
                        default argument values are indicated.

RESTRICTIONS   - a summary of restrictions on the use of

                        qualifiers

        DESCRIPTION   - the definitive description of the utility
                              action.

        COMMANDS   - for interactive utilities only, a description
                              of all commands. Commands are ordered
                              alphabetically and default argument values are
                              indicated.

        EXAMPLES   - annotated examples of utility useage.

        MESSAGES   - all classes of message are listed and described
                              and suggested user action given. The messages
                              are divided into sections according to message
                              severity within which the messages are ordered
                              alphabetically by message mnemonic.

Conventions used in this document

        Convention   Meaning

        <CR>   The user should press the carriage control key
                            on the terminal

        <CTRL/x>   The phrase <CTRL/x> indicates that the user must
                            press the key labelled CTRL while simultaneously
                            pressing another key, for example, <CTRL/Z>.

        $ **IFF2SIF<CR>**   Command examples show all user entered commands
                            in **bold** type.

        $ **IFF2SIF<CR>**   Vertical series of periods, or ellipsis, mean
                 .   either that not all the data that CONVERT would
                 .   display in response to the particular command is
                 .   shown or that not all the data that the user
                            would enter is shown.

        file-spec...   Horizontal elipsis indicates that additional
                            parameters, values or information can be
                            entered.

        [logical-name]   Square brackets indicate that the enclosed item
                            is optional. (Square brackets are not, however,
                            optional in the syntax of a directory name in
                            a file-specification, or in the syntax of a
                            substring specification in a VMS assisnment
                            statement).

```
          ------------------------------------------------------------------
          Convention            Meaning
          ------------------------------------------------------------------
          'integer'             An integer number is expected in the specified
                                input or output field. (See "Command line data
                                types" below).

          'real'                A real number is expected in the specified input
                                or output field. (See "Command line data types"
                                below).

          'file-spec'           A VMS file specification is expected in the
                                specified input or output field.

          'device-name'         A VMS device specification (for instance, MTA0:)
                                is expected in the specified input or output
                                field.
          ------------------------------------------------------------------
```

CHAPTER 1

MOSS FORMAT DESCRIPTION

FORMAT DESCRIPTION

    Introduction

    I2MOSS provides a translation from Laser-Scan's Internal Feature
    Format (IFF) to a subset of MOSS major option GENIO (**GEN**eral **I**nput and
    Output) format. Only minor option 080 ("General input of
    n-dimensional strings on the standard MOSS input device") of GENIO is
    supported. For a brief comparison of the way IFF and MOSS GENIO
    regard data, see the section on DATA PREPARATION below.

    The format produced is an ASCII text format, as described in the
    document **MOSS Modelling Systems** first published in January 1975 and
    revised and reprinted in 1981.

    Magnetic tape output

    The program writes to magnetic tape in standard MOSS ASCII format.
    The output is at either 800 or 1600 bpi, in any block size that is
    divisible by 80 and which lies in the range 80 to 8000 bytes
    (inclusive). Records have a fixed length of 80 bytes and are padded
    with spaces if necessary.

    There are no header blocks, only a header record at the start of the
    first block of the file. The last block of each file is padded with
    spaces. Each file is separated from the previous by a tapemark, and
    there are two tapemarks after the last file. Files on the tape are
    synonymous with the original IFF files output.

    The program is capable of writing magnetic tapes on either a
    conventional unibus/massbus magnetic tape device or on a serial line
    magnetic tape drive.

    Disk file output

    Output may also be made to disk file. Again, the record length
    defaults to 80 bytes. Records are filled with spaces if necessary.

    Each disk file contains the output from one IFF file.

    These files may then be output to magnetic tape using either of the
    VAX programs BACKUP or EXCHANGE - the former is intended for transfer
    between VAXes, and the latter provides an ANSI standard tape format
    (it replaces the RSX utility FLX on VAXes with VMS version 4.0 and
    up). Consult the relevant Digital manuals for more details of these
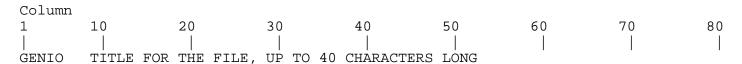   utilities.

The MOSS GENIO subset supported

The data is output using only MOSS major option GENIO, minor option
80.   GENIO minor option 80 enables "General input of n-dimensional
strings on the standard MOSS input device".   Since IFF format is
limited to data storage in only three dimensions only 3 dimensional
coordinates may be output to the MOSS file,

This section describes what may be expected in the output.

Summary of MOSS output

MOSS GENIO records are fixed field 80 byte card images.   MOSS GENIO
files always start with a record containing the following:

```
Column
1       10        20        30        40        50        60        70        80
|       |         |         |         |         |         |         |         |
GENIO   TITLE FOR THE FILE, UP TO 40 CHARACTERS LONG
```

This file header record is then followed by the header of the first
coordinate string in the file:

```
Column
1       10        20        30        40        50        60        70        80
|       |         |         |         |         |         |         |         |
080AA10          170.000 0.000     0.000     2.0     0.000     0.000
```

The start of a new string header record contains '080' in columns 1-3.
This signifies to MOSS that the header is for GENIO minor option 080.

The 'AA10' in columns 4-7 is the string label, the equivalent of the
FC (Feature Code) in an IFF feature.

A powerful feature of I2MOSS is the ability to carry into the MOSS
GENIO file the feature coding applied in the IFF file via a user
definable lookup table which associates each IFF feature code with a
MOSS GENIO string label. The lookup table file is explained in the
section decribing data preparation.

Columns 8-15 contain an optional string subreference.   This is not
currently supported by I2MOSS.

Columns 16 to 23 contain the height of a contour. This is taken from
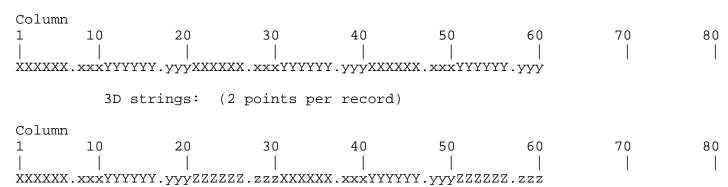the IFF AC of a type specified by the user at I2MOSS run time.

Columns 24-33 and 34-43 contain the optional origin offset (northings
and eastings respectively) for the string. I2MOSS will write into
this field the X,Y offset from the IFF type 2 Map Descriptor origin
offset.   If I2MOSS receives an IFF file having a type 1 MD the string
offset will be 0.0,0.0.

Columns 44-51 contain the dimension of the string. The maximum string
dimensionality supported by I2MOSS is 3 dimensions. I2MOSS can thus
format both IFF 2 dimensional (IFF ST) and 3 dimensional (IFF ZS)
strings.

Columns 52-61 and 62-71 contain the number combination used to signify
the end of the string, (eastings and northings respectively). The
default values are 0.0,0.0. If a string is to be input with the
coordinates 0.0,0.0 then another combination such as 9999.999,9999.999
should be specified on the I2MOSS command line.

The string information then follows according to the standard formt of
F10.3 with the number of points per record as shown:

2D strings: (3 points per record)

```
Column
1        10         20         30         40         50         60         70         80
|         |          |          |          |          |          |          |          |
XXXXXX.xxxYYYYYY.yyyXXXXXX.xxxYYYYYY.yyyXXXXXX.xxxYYYYYY.yyy
```

3D strings: (2 points per record)

```
Column
1        10         20         30         40         50         60         70         80
|         |          |          |          |          |          |          |          |
XXXXXX.xxxYYYYYY.yyyZZZZZZ.zzzXXXXXX.xxxYYYYYY.yyyZZZZZZ.zzz
```

Sufficient records of the above formats are used to define the string.
A whole IFF feature is considered as one MOSS GENIO string. The MOSS
GENIO string is terminated with the values 0.0,0.0 or the number
conbination defined for use as string terminator in the 080 string
header record.

CHAPTER 2

MOSS DATA PREPARATION

DATA PREPARATION

    Comparison of MOSS and IFF

    IFF is a feature oriented data format - data is separated into
    features, and each feature represents one 'thing' on the map. An
    integer feature code is used to say what type of thing it is that the
    feature represents. A feature might thus be a contour, or a house, or
    a river, depending on the type of map being digitised, and the way
    that the data is to be used.

    MOSS GENIO format shares this concept of a 'feature', and the simple
    form of MOSS we produce is essentially a fixed format ASCII
    representation of an IFF file on a feature by feature basis.

    The subset of MOSS GENIO format produced by I2MOSS does not contain
    any facility for the transmission of IFF map level entries, layers,
    text, line thickness or multiple attribute coding of IFF features.

    I2MOSS is thus best suited for use with an IFF containing spot
    heights, contour strings or 3 dimensional strings (held in IFF ZS
    entries). All spot height (IFF one point ST entries) and contour
    features (2 dimensional IFF strings held in ST entries) MUST have a
    height value held in an IFF AC entry. By default this AC will be of
    type 3, which carries the height value as a floating point number.
    Any ACs in IFF features which consist of 3d strings (IFF ZS entries)
    will be ignored. I2MOSS cannot be used to generate MOSS GENIO strings
    of greater than 3 dimensions.

    Pre digitising preparation

    The customer should supply a list of the MOSS GENIO string labels to
    be associated with each feature type on the map, e.g. road casings,
    contours, culverts etc.

    For digitising within IFF, feature codes must be assigned to all of
    these data-types, and an FRT file (together with SRI and TRI files)
    created. Care should be taken in considering what graphical type the
    IFF features should have. I2MOSS cannot, for example, perform the
    interpolation required by graphical type 6 (interpolated curve).

    The relationship between the IFF feature codes used during processing
    within the LAMPS system and MOSS GENIO string labels should be defined
    in a I2MOSS cross codes file, (see below).

Three-dimensional and contour data

Three-dimensional MOSS output

IFF2MOSS is normally used to output contour data, but it can also
output MOSS data in three dimensional mode. The output data are three
dimensional coordinates - that is, an X,Y and Z value for each data
point.

Features should be digitised into IFF ZS (3D string) entries if MOSS
3D strings are required for output.

Contour data

If there is only a single height associated with each feature, (e.g.
contours), IFF2MOSS translates any height AC found in the IFF file
into the height field of the 080 string header record.

Contours should be digitised into IFF ST (2D string) entries.

Heights in IFF files used for MOSS formatting

Every IFF feature being output **MUST** contain explicit height data.   If
the feature contains ZS entries, then these will be used directly.   If
the feature contains ST entries, then a height must be supplied in a
type 2 or 3 contour height AC. If MOSS GENIO format is being used to
output data which have no attached height information then it should
be agreed with the customer what value is to be written into the
height field of the '080' string header record.   Even if no height
informtion is required an AC (Ancillary Code) must be present in all
IFF features. This AC by default will cause the transmission of the
height 0.0 to the MOSS GENIO file.

Offsetting data

In some cases, the size of the map being digitised is small compared
to the coordinate values that are required as output - for instance,
if data is output in UTM coordinates (as metres on the ground), then
the numbers being output will be very large. In such cases, the
actual data in the IFF file is generally offset relative to the bottom
left hand corner of the particular sheet, to preserve accuracy. The
offset removed from the data is carried in a type 2 MD (Map
Descriptor) in the IFF file.

I2MOSS enables the user to specify the origin offset of the IFF data
in the origin offset field of a type 2 Map Descriptor.

N.B. MOSS GENIO forbids negative coordinates if the MOSS string
header contains an origin offset.

-- --
The I2MOSS CROSS CODES file

        A powerful feature of I2MOSS is the ability to carry into the MOSS
        GENIO file the feature coding applied in the IFF file via a user
        definable lookup table which associates each IFF feature code with a
        MOSS GENIO string label.   The lookup table file is specified to I2MOSS
        via the I2MOSS/CROSS_CODES qualifier.   The CROSS CODES lookup file is
        an ASCII text file containing IFF feature code/MOSS GENIO string label
        pairs.   The format of this file is best explained using the annotated
        example set out below:

```
!========================================================================
! Example /CROSS__CODES file to cross tabulate IFF feature codes to
! MOSS GENIO string labels. Note free format, two entities per record.
! The first entity is the IFF feature code, the second is the MOSS GENIO
! string label to be output in the MOSS string header for any features
! which had the associated IFF feature code
!
! "!" is a comment delimiter, all text to the right of a "!" will be
! ignored. A "!" must not appear between the IFF feature code and the
! MOSS string label. I2MOSS will report an error if this is found.
!
!
! Either lower or upper case may be used (or a mixture of the two),
! I2MOSS will upper-case all text before output to the MOSS GENIO card
! images
!
!
!
! MOSS string labels are restricted to 4 characters and are usually of
! the form AA00, where AA are 2 alphabetic characters. You are free, of
! course, to use what-ever your local MOSS installation will stand, but
! bear in mind that I2MOSS will truncate your string to 4 characters.
!
!
 121 KI00        ! an in-line comment!
 122 KI00
 124 KI00
 128 KI
 123 KI0
!
! very free format records (all valid)
!
 13                                      RIVS
             123     ROAD
     800                                              case     ! wow
5                   Go


!
! blank lines may be used to separate records if you wish
!


987                     cat
```

0   dog

717   fish

!
! end
!

-- --
Digitising data for transfer to MOSS GENIO format

Features which are to receive different MOSS GENIO string label should
be similarly differentiated by IFF feature code. If desired, data may
be differentiated by feature code in the IFF file but can be given the
same MOSS string label after lookup in the /CROSS_CODES file. The
example /CROSS_CODES file (above) shows that the different IFF feature
codes 121, 122, and 124 all become KI00 on output to MOSS format. It
is thus possible to separate data for check plotting purposes while in
IFF format which are required combined when translated to MOSS format.

IFF layer 0 is reserved for registration mark features and should not
be used to contain any digitising. I2MOSS will ignore the contents of
IFF layer 0.

All contour features should contain an AC (Ancillary Code) to carry
the contour height. By default I2MOSS expects to find a type 3 AC in
each feature. The user may specify alternative AC types used to carry
height information on the I2MOSS command line at run time. Any
features which lack a type 3 AC or an AC of the specified type will
not be transferred to MOSS GENIO format.

No height AC is required in 3D IFF features containing ZS entries.

Care should be taken that the digitised coordinates will fit into the
MOSS GENIO record format of F10.3.   If they do not then output
conversion error will ocurr. To overcome this problem, ITRANS should
be used to subtract an offset from the IFF coordinates. This offset
is placed in the origin offset field of a type 2 MD (Map Descriptor).
The offset to be removed should be chosen carefully so as to ensure
that none of the string coordinates becomes negative.

CHAPTER 3

I2MOSS UTILITY

UTILITY   I2MOSS

FUNCTION

> I2MOSS is an IFF to MOSS GENIO format converter. It produces MOSS
> GENIO card images for 2 and 3 dimensional MOSS strings based on IFF
> spot height, contour and 3 dimensional strings. Output may be
> directed to disk file or to magnetic tape.

FORMAT

> $ I2MOSS input-file-spec output-file-spec

Command Qualifiers	**Defaults**
/CROSS_CODES[=file-spec]	MOSS strings not labelled.
/HEIGHT_AC=integer	Heights expected in type 3 ACs.
/[NO]LOG	/NOLOG
/NAME="ascii-string"	File name is used as model name.
/NOFC=[integer[,...]]	All IFF feature codes formatted.
/NOLAYER=[integer[,...]]	All IFF layers formatted.
/TERMINATOR=real	MOSS default 0.0, 0.0 used.

> **Qualifiers for magnetic tape output:**

/BLOCK_SIZE=integer	/BLOCK_SIZE=800
/DENSITY=800	/DENSITY=1600
/REWIND	/NOREWIND
/SERIAL_MAGTAPE	Output to unibus/massbus magtape

PROMPTS

> _IFF-file: input-file-spec
> _MOSS-file: output-file-spec

PARAMETERS

input-file-spec

> - specifies the IFF file which is to be formatted into MOSS GENIO card
>   images. Any part of the file name which is not specified will be
>   taken from the default specification 'LSL$IF:IFF.IFF'.

output-file-spec

> - specifies the MOSS GENIO file which is to be created. Any part of the
>   file name which is not explicitly given will be taken from the parsed
>   input specification, but the default extension '.MOSS'.   I2MOSS can
>   output the MOSS GENIO format data to magnetic tape. The default
>   action is to create a disk file. Should output to magnetic tape be
>   required it is necessary to supply the desired tape device name as the
>   file-spec (see examples below).

QUALIFIERS

/CROSS_CODES=file-spec

> - enables the user to specify a file containing a table of IFF feature
>   codes and MOSS GENIO string labels. This table is used to convert the
>   IFF feature codes into MOSS GENIO string labels to enable strings with
>   different IFF feature codes to be differentiated in the MOSS format
>   file. The default action of I2MOSS is not to output any MOSS GENIO
>   string labels at all.   If /CROSS_CODES is specified without a file
>   argument then the default /CROSS_CODES file; LSL$MOSS:MOSSCODES.DAT is
>   searched for and opened.   If /CROSS_CODES is specified with a file
>   argument then the file-spec supplied is parsed against the default
>   file-spec 'LSL$MOSS:MOSSCODES.DAT'. A description of the layout of a
>   /CROSS_CODES file together with an annotated example is given in the
>   DESCRIPTION section below.

/HEIGHT_AC=integer

> - by default I2MOSS expects all IFF 2d string features (and spot height
>   features held in IFF ST entries) to have a type 3 AC containing the
>   feature height as a floating point number.   /HEIGHT_AC enables the
>   user to specify that the feature height is to be extracted from the
>   longword field of the specified AC type as an integer value. Only one
>   type of AC may be specified for one run of I2MOSS. Features which do
>   not contain an AC of the specified type will not be output into MOSS
>   GENIO format.

/LOG
/NOLOG (default)

> - this will result in supplementary messages being sent to SYS$OUTPUT.
>   Supplementary messages are generated when a file is successfully
>   opened and a reassuring message output each time a new IFF layer is
>   found in the IFF file.

/NAME='ascii-string'

>   - enables the user to specify the model name to be written into the
>   first record of the MOSS GENIO file immediately after the GENIO option
>   keyword. The model name may have a maximum of 40 characters.   By
>   default I2MOSS will extract and use the filename from the IFF input
>   file specification as the MOSS model name. Thus if the input IFF file
>   was DUA3:[BUREAU.MOSS]CULVERTSANDGUTTERS.IFF;9 the default model name
>   would be "CULVERTSANDGUTTERS".

/NOFC=integer[,...]

>   - enables deselection for MOSS output of all IFF features which have
>   feature codes of the specified values. By default I2MOSS will process
>   all features in the input IFF file (except those in layer 0).   The
>   maximum number of feature codes which may be deselected is 1024.

/NOLAYER=integer[,...]

>   - enables deselection for MOSS output of all IFF features which lie in
>   the layers with the specified values. The maximum number of IFF
>   layers which may be deselected is 1024. By default I2MOSS will not
>   convert to MOSS GENIO any features which lie within IFF layer 0, as
>   this layer is reserved for digitising control and registration
>   features.

/TERMINATOR=integer

>   - by default I2MOSS uses the default MOSS GENIO string termination value
>   of 0.0 0.0.   /TERMINATOR enables the user to specify a different
>   coordinate to be used to signify the end of a MOSS string.
>   /TERMINATOR expects only one argument as the x and y ordinate of the
>   terminating string coordinate will be made the same.   If I2MOSS is
>   used on an IFF file containing a type 2 IFF MD (Map Descriptor) which
>   is set up, the /TERMINATOR argument must not be negative. MOSS GENIO
>   forbids negative coordinates if the MOSS string header contains an
>   origin offset, such an origin offset will be extracted from a type 2
>   IFF MD entry.

Qualifiers for use with magnetic tape output.

/BLOCK_SIZE=integer

>   - by default I2MOSS writes magnetic tape with a block size of 800 bytes.
>   /BLOCK_SIZE enables the user to specify a different block size. MOSS
>   GENIO files consist of card images each represented by an 80 byte
>   record.   The /BLOCK_SIZE argument must be divisible by 80, and lie in
>   the range 80 to 8000. I2MOSS checks that these criteria are met.

/DENSITY=integer

>   - I2MOSS can write magnetic tapes at 800 and 1600 bpi. By default 1600
>   bpi is used. /DENSITY enables the user to specify the tape density to
>   be used, only /DENSITY=800 (bpi and /DENSITY=1600 (bpi) are valid.

/REWIND

>   - by default I2MOSS does not rewind the magnetic tape before writing a
>   MOSS file to it.   I2MOSS looks for end of volume, signified by two
>   tape marks, and positions the write head between the two tape marks
>   before writing the file. The second tape mark is over-written and a
>   new end of volume substituted at the end of the new file.   Thus more
>   than one MOSS file may be written to the same tape. /REWIND enables
>   the user to specify that the tape is to be rewound and the new file is
>   to be written starting at the beginning of the tape. Any existing
>   file(s) on the tape will be over-written.

/SERIAL_MAGTAPE

>   - if a device name is specified on the output file-spec I2MOSS checks to
>   determine whether the device is a magnetic tape drive. If however the
>   magnetic tape drive is a non-standard DEC machine on a serial line
>   I2MOSS determines that it is trying to output to a terminal (on the
>   serial line) and does not do the tape output required.
>   /SERIAL_MAGTAPE, when used in conjunction with a magnetic tape logical
>   name on the output file specification will result in the magnetic tape
>   drive (perceived as a terminal) to be used correctly as a magnetic
>   tape drive.

DESCRIPTION

>   I2MOSS is Laser-Scan's IFF to MOSS GENIO formatter.   It offers the
>   following features:

>   o   converts data from Laser-Scan IFF format disk file to MOSS GENIO
>   card images on magnetic disk;

>   o   optionally converts data from Laser-Scan IFF format disk file to
>   MOSS GENIO card images on magnetic tape;

>   o   if magnetic tape output is selected the user may choose between
>   800 bpi and 1600 bpi, and also to select the output block size;

>   o   can format IFF 2 dimensional (IFF ST) and 3 dimensional (IFF ZS)
>   strings;

>   o   enables the user to specify the IFF AC type used to carry the
>   string height for 2 dimensional strings;

>   o   enables the user to specify that strings identified by particular
>   IFF feature codes may given specific MOSS GENIO string labels to
>   enable the strings to be differentiated in the MOSS file;

     o   enables the user to specify the MOSS GENIO string terminator
        combination;

     o   enables the user to specify the MOSS GENIO model name string;

     o   enables the user to specify the MOSS GENIO string origin offset
        for all strings in the MOSS file based on the IFF type 2 Map
        Descriptor origin offset;

     o   enables the user to specify whether particular IFF features are to
        be excluded from the MOSS GENIO file on the basis of IFF feature
        code or position within IFF layer.

I2MOSS must be supplied with an IFF containing spot heights, contour
strings or 3 dimensional strings (held in IFF ZS entries). All spot
height (IFF one point ST entries) and contour features (2 dimensional
IFF strings held in ST entries) MUST have a height value held in an
IFF AC entry. By default this AC will be of type 3, which carries the
height value as a floating point number. Any ACs in IFF features
which consist of 3d strings (IFF ZS entries) will be ignored. For
further details of I2MOSS digitising requirements see the data
preparation section above.

A powerful feature of I2MOSS is the ability to carry into the MOSS
GENIO file the feature coding applied in the IFF file via a user
definable lookup table which associates each IFF feature code with a
MOSS GENIO string label. For details of the content and layout of
I2MOSS CROSS CODES files see the data preparation section above.

I2MOSS and logical names

I2MOSS requires that logical name LSL$IF: (or IF:) should point to
the directory containing the IFF file which is to be converted into
MOSS GENIO format. It also requires that logical name LSL$MOSS: be
set up to point to a directory containing the default lookup table for
the /CROSS_CODES command qualifier. The default lookup table must be
called LSL$MOSS:MOSSCODES.DAT. An annotated example of such a lookup
table is given in the DESCRIPTION section. It is recommended that
LSL$MOSS: point to a directory which is quite separate from the
normal working directories to reduce the chance of the lookup file(s)
being accidentally deleted.

RESTRICTIONS

I2MOSS operates with the following restrictions:

     o   I2MOSS cannot be used to generate MOSS GENIO strings of greater
        than 3 dimensions.

o   Features IFF layer 0 will not be formatted into MOSS GENIO card
    images, as these features are reserved for use as digitising
    control and registration marks.

o   No IFF text entries will be processed.

o   The /NOLAYER and /NOFC qualifiers can each take a maximum of 128
    arguments (including, of course, burst numeric ranges) only.

o   The /NAME qualifier takes an ASCII string of maximum length 40
    bytes.

o   MOSS string origin offsets are only available from the type 2 MD
    entry (Map Descriptor). If a type 2 MD origin offset is used in
    conjunction with /TERMINATOR the /TERMINATOR argument must not be
    negative. MOSS GENIO forbids negative coordinates if the MOSS
    string header contains an origin offset.

o   If using a magnetic tape drive on a serial line (perceived as a
    terminal) the /SERIAL_MAGTAPE qualifier must be specified.

o   If writing to magnetic tape the block size must be divisible by
    80, and lie in the range 80 to 8000.

EXAMPLES

$ **I2MOSS CONTOURS AMOS**
 ELAPSED: 00:01:32.06 CPU: 0:00:53.65 BUFIO: 29 DIRIO: 126 FAULTS: 108
$

        This example demonstrates the most common use of I2MOSS.   IFF file
        LSL$IF:CONTOURS.IFF has been successfully processed and a MOSS GENIO
        format disk file LSL$IF:AMOS.MOSS;0 has been produced.   The GENIO
        strings will not have string labels.

$ **I2MOSS TST/NOLAY=(1230:1270)/LOG/NAME="EXAMPLE MODEL ONE" JIM<CR>**
%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.RO]TST.IFF;0 opened for read
%I2MOSS-I-MOSOPNOUT, MOSS file LSL$IF:JIM.MOSS;0 opened for output
%I2MOSS-I-SKIPLYR, skipping IFF layer 1230
%I2MOSS-I-LAYER, IFF layer 1 found
%I2MOSS-I-LAYER, IFF layer 88 found
%I2MOSS-W-MISSHT, IFF feature FSN 270 (36) has no height AC
-I2MOSS-I-SKIPFEAT, skipping IFF feature FSN 270 (36)
%I2MOSS-I-LAYER, IFF layer 66 found
%I2MOSS-I-LAYER, IFF layer 55 found
%I2MOSS-I-LAYER, IFF layer 44 found
%I2MOSS-W-MISSHT, IFF feature FSN 330 (44) has no height AC
-I2MOSS-I-SKIPFEAT, skipping IFF feature FSN 330 (44)
%I2MOSS-I-LAYER, IFF layer 11 found
%I2MOSS-I-LAYER, IFF layer 151 found
%I2MOSS-I-LAYER, IFF layer 9999 found
 ELAPSED: 00:00:04.97 CPU: 0:00:01.94 BUFIO: 42 DIRIO: 21 FAULTS: 171
$

>   This example demonstrates the use of I2MOSS with the /LOG qualifier
>   set.   The number of messages output has increased. Notice that there
>   are some warning messages caused by the specified features lacking a
>   height AC.   By default the height AC is taken to be type 3 (floating
>   point height). The /NOLAYER qualifier has been used to exclude any
>   IFF layer found in the IFF file which lies in the range 1230 to 1270.
>   Although the input file LSL$IF:TST.IFF has been successfully processed
>   those features which lacked a height AC will not have been transferred
>   to the MOSS GENIO file. The IFF file should be modified using LITES2
>   to add the missing ACs and I2MOSS re-run. Despite the omitted
>   features the run complete successfully and $STATUS is set to
>   SS$_NORMAL.

$ **I2MOSS SEWERS/HEIGHT_AC=2/LOG/CROSS_CODES=PROJECT1.DAT SEWERS<CR>**
%LSLLIB-I-IFFOPENED, DUA3:[BUREAU.RO]SEWERS.IFF;0 opened for read
%I2MOSS-I-MOSOPNOUT, MOSS file LSL$IF:SEWERS.MOSS;0 opened for output
%I2MOSS-I-CRSSOPNIN, file LSL$MOSS:PROJECT1.DAT opened for /CROSS_CODES
%I2MOSS-I-LAYER, IFF layer 12 found
%I2MOSS-I-LAYER, IFF layer 11 found
%I2MOSS-I-LAYER, IFF layer 16 found
%I2MOSS-I-LAYER, IFF layer 9 found
 ELAPSED: 00:00:04.99 CPU: 0:00:01.96 BUFIO: 110 DIRIO: 22 FAULTS: 180
$

>   This example illustrates the use of the /CROSS_CODES qualifier to
>   specify a look-up file for translation of IFF feature codes into MOSS
>   GENIO string labels. Notice that the /CROSS_CODES file-spec was
>   supplied without a device and directory specification and so the
>   default device and directory specification have been applied using
>   logical name LSL$MOSS:. The run complete successfully and $STATUS is
>   set to SS$_NORMAL.

$ **I2MOSS TST/HEIGHT_AC=2/LOG/REWIND/BLOCK_SIZE=2048/DENSITY=800 MSA0:<CR>**

%I2MOSS-E-INVBLK, invalid block size 2048
-I2MOSS-I-BLKDIV, block size must be divisible by 80
 ELAPSED: 00:00:00.51 CPU: 0:00:00.17 BUFIO: 2 DIRIO: 0 FAULTS: 76

    Although intended to create an output file on the magnetic tape unit
MSA0: this run of I2MOSS has failed.   By default I2MOSS writes
magnetic tape with a block size of 800 bytes.   In this example the
user has specified the /BLOCK_SIZE qualifier to change the block size
written to tape.   MOSS GENIO files consist of card images each
represented by am 80 byte record. The /BLOCK_SIZE argument must be
divisible by 80, and lie in the range 80 to 8000. I2MOSS checks that
these criteria are met. One of them was not. Notice that the /REWIND
qualifier was specified to force the output file to be written at the
beginning of the tape, thus over-writing any data already on the tape.
The default tape density of 1600 bpi has been explicitly replaced by
the /DENSITY=800 (bpi) qualifier. $STATUS is set to SS$_ABORT.

$ **I2MOSS/SERIAL_MAGTAPE/REWIND/BLOCK_SIZE=4000/DENSITY=800<CR>**
_IFF-file: **TST<CR>**
_MOSS-file: **LSL$TERTAPE:<CR>**
 ELAPSED: 00:00:00.64 CPU: 0:00:00.27 BUFIO: 3 DIRIO: 0 FAULTS: 111
$

    In this example output is to be to the serial line magnetic tape unit
pointed to by the logical name LSL$TERTAPE:. As the tape unit is on a
serial line (which I2MOSS will perceive to be a terminal device),
I2MOSS must be told that this is a non-standard (ie non
massbus/unibus) tape drive with the /SERIAL_MAGTAPE qualifier.   The
file is to be written at 800 bpi with a block size of 4000. The tape
will be rewound before any output begins.   The run has completed
successfully, $STATUS is set to SS$_NORMAL.

MESSAGES (informational)

These messages give information only, and require no immediate action by the
user (except   perhaps   to   verify that the correct options have been selected).
Most will only appear if /LOG has been specified on the   command   line, unless
they are used to provide supplementary information after an error has occurred.

BLKDIV, block size must be divisible by 80

   Explanation: You have used /BLOCK_SIZE to specify a different block size
   from the default of 800 bytes.   MOSS GENIO files consist of card images each
   represented by am 80 byte record.   The   /BLOCK_SIZE   argument must be
   divisible by 80, and lie in the range 80 to 8000.

   User action: Check that the /BLOCK_SIZE   argument   size   criteria   are   met
   before retyping the command line.

CRSSOPNIN, file 'file-spec' opened for /CROSS_CODES

   Explanation: The specified lookup table file has been   successfully   opened
   for   input.   IFF feature code to MOSS GENIO string label correlation will be
   based on its contents.

   User action: None.

LAYER, IFF layer 'integer' found

   Explanation: I2MOSS has reached the specified IFF layer while processing
   the IFF input file.

   User action: None.

MINMAXBLK, minimum block size is 80, maximum block size is 8000

   Explanation: By default I2MOSS writes magnetic tape with a   block   size   of
   800   bytes.   /BLOCK_SIZE enables the user to specify a different block size.
   MOSS GENIO files consist of card images   each   represented   by   an   80   byte
   record.   The   /BLOCK_SIZE   argument must be divisible by 80, and lie in the
   range 80 to 8000.   I2MOSS checks that these criteria are met.

   User action: Check that the /BLOCK_SIZE   argument   size   criteria   are   met
   before retyping the command line.

MOSOPNOUT, MOSS file 'file-spec' opened for output

Explanation: The specified MOSS file has been successfully created and opened for output.

User action: None.

POSSDEN, possible densities are 800 and 1600 bpi

Explanation: by default I2MOSS writes magnetic tape with a density of 1600 bpi. An alternative density of 800 bpi is provided via the /DENSITY qualifier. You have specified an argument to /DENSITY which is neither 800 bpi or 1600 bpi.

User action: Check that the /DENSITY argument is either 800 or 1600 when retyping the command line.

SKIPFEAT, skipping IFF feature FSN integer (integer)

Explanation: As a result of specifying feature codes with the /NOFC qualifier IFF features with the specified feature codes will not be output to the MOSS GENIO file. This message tells the user that a particular feature is being ignored.

User action: None.

SKIPLAY, skipping IFF layer 'integer'

Explanation: As a result of specifying IFF layers with the /NOFC qualifier IFF features within the specified IFF layers will not be output to the MOSS GENIO file. This message tells the user that a particular layer is being ignored.

User action: None.

MESSAGES (warning)

These messages indicate that I2MOSS has been unable to guarantee that a
specified feature has been correctly processed.

LABTOOLNG, MOSS string label is too long 'ascii-string'

    Explanation: MOSS GENIO string labels can only be 4 characters long.
    I2MOSS has read from the /CROSS_CODES file a label which has more than 4
    characters. The string label has been truncated and written to the MOSS
    GENIO file.

    User action: Check that a truncated string label is acceptable. Why was it
    too long in the first place?

MISSHT, IFF feature FSN 'integer' ('integer') has no height AC
    Explanation: The specified IFF feature has no AC entry of the required type
    (default type 3 or some other identified by /HEIGHT_AC). The IFF feature
    will not be transferred to the MOSS file.

    User action: Check why the feature does not have a height AC. If necessary
    modify the IFF feature using LITES2 then rerun I2MOSS.

NOTEOF, tape was not at end of file. Rewinding and looking for end of volume.

    Explanation: The /NOREWIND qualifier (default) was specified and the tape
    drive write head was not at the last end of file position. The tape will be
    rewound and the end of volume (signified by two successive tape marks) will
    be found. The write head is then positioned between the two tape marks and
    output can then begin.

    User action: None.

UNSETMD, IFF type 2 map descriptor is unset

    Explanation: The input IFF file has a type 2 MD (Map Descriptor) which is
    unset.

    User action: Ensure that the IFF MD is meant to be unset.

MESSAGES (error)

These messages indicate an error in processing which has caused the program to
terminate. The most likely causes are a corrupt or otherwise invalid input IFF
file, or an error related to command line processing and file manipulation. It
is most unlikely that any output file produced will be correctly processed.
Generally, in the event of a processing error I2MOSS will delete the MOSS output
file before calling exit, (unless writing to magnetic tape).

BADDEV, error enquiring about device 'device name'

   Explanation: If the output file specification contains a device name I2MOSS
   uses a VMS system service (LIB$GETDVI) to determine whether the output
   device is a magnetic tape unit. The system service considers the output
   device name to be invalid. Bear in mind that the output file specification
   may have defaulted from the input file-specification which may contain a
   device name.

   User action: Ensure that the output device name was correctly specified.
   If the problem persists consult your system manager.

COMPOSINV, comment delimiter position invalid in /CROSS_CODES file in line 'int'
   'string'

   Explanation: A line in the /CROSS_CODES file contains a comment delimiter
   (a "!") in an invalid position. The message contains a string 'string' and
   the line number 'integer' which should help identify the offending line.

   User action: Correct the offending /CROSS_CODES file line. Re-run I2MOSS.

CROSSEOF, unexpected end of /CROSS_CODES file detected

   Explanation: The last line in the /CROSS_CODES file contains only one of
   the expected pair of values or the /CROSS_CODES file is empty.

   User action: Correct the offending /CROSS_CODES file line. Re-run I2MOSS.

ERRDCRSS, error reading line 'integer' of /CROSS_CODES file

   Explanation: The specified line in the /CROSS_CODES file contains only one
   of the expected pair of values, or an alphabetic character in the IFF
   feature code field.

   User action: Correct the offending /CROSS_CODES file line. Re-run I2MOSS.

ERROPMOSS, error opening MOSS file 'file-spec' for output

Explanation: The specified file cannot be created and opened for writing.
This message will usually be followed by another giving the reason for the
failure.

User action: The accompanying message should indicate the cause of the
error.   Check the file specification for errors, check that that directory
is not write-protected.   If the problem continues, notify the system
manager.

INVBLK, invalid block size 'integer'

Explanation: By default I2MOSS writes magnetic tape with a block size of
800 bytes. /BLOCK_SIZE enables the user to specify a different block size.
MOSS GENIO files consist of card images each represented by an 80 byte
record.   The /BLOCK_SIZE argument must be divisible by 80, and lie in the
range 80 to 8000. I2MOSS checks that these criteria are met.

User action: Check that the /BLOCK_SIZE argument size criteria are met
before retyping the command line.

INVDENS, invalid tape density selected

Explanation: The /DENSITY qualifier has been specified with an invalid tape
density. I2MOSS supports only 800 bpi and 1600 bpi tape output.

User action: Re-enter the command line and specify either /DENSITY=800 or
/DENSITY=1600.

INVALFC, 'integer' is an invalid feature code argument - value must lie in range
    0 - 32767

Explanation: The specified feature code lies outside the indicated range
for valid IFF feature codes.

User action: Respecify the I2MOSS command line ensuring that any feature
code specifications lie within the range 0 - 32767.

INVALLAY, 'integer' is an invalid layer argument - value must lie in range
    0 - 32767

Explanation: The specified feature code lies outside the indicated range
for valid IFF layer numbers.

User action: Respecify the I2MOSS command line ensuring that any layer
number specifications lie within the range 0 - 32767. (Note that by default
layer 0 is reserved for registration marks and grid features).

MTBLK, error writing magnetic tape block

Explanation: I2MOSS has received a message from the I2MOSS magnetic tape
i/o routines that an error occurred writing a block to the tape.

User action: The accompanying message should indicate the cause of the
error.   Check that the tape is still loaded, check that the tape unit is
online, check that that tape is not write-protected.   If the problem
continues, notify the system manager.

MTOPEN, error initialising tape unit 'device name'

Explanation: I2MOSS has determined that the output device name is that of a
magnetic tape unit and has attempted to initialise the tape unit. It has
failed.

User action: The accompanying message should indicate the cause of the
error. Check that the tape is mounted, check that the tape is loaded, check
that the tape unit is online, check that that tape is not write-protected.
If the problem continues, notify the system manager.

NEGCOORD, IFF feature FSN 'integer' ('integer') has negative coordinate.
     Invalid with MOSS origin offset.

Explanation: I2MOSS has detected an IFF feature containing a negative
coordinate in spite of the fact that the IFF file contains a set type 2 MD
(Map Descriptor) entry. The map descriptor contains an origin offset. MOSS
GENIO forbids negative coordinates with a string origin offset.

User action: Correct the offending coordinate(s). Re-run I2MOSS.

OPNCROSS, error opening file 'file-spec' for /CROSS_CODES

Explanation: The specified file cannot be opened for reading. This message
will usually be followed by another giving the reason for the failure.

User action: The accompanying message should indicate the cause of the
error. Check the file specification for errors, check that logical name
LSL$MOSS: has been assigned to the correct device and directory, check that
the file and directory are not protected against read access. If the
problem continues, notify the system manager.

TOOMNYLAB, too many MOSS labels defined

Explanation: The file specified by the /CROSS_CODES qualifier (default
LSL$MOSS:MOSSCODES.DAT) contains more than 500 IFF feature code to MOSS
GENIO string label assignments.

User action: Reduce the number of IFF feature code to MOSS GENIO string
label assignments. Re-run I2MOSS.

UNDEFCLAB, MOSS string label for IFF feature code undefined

Explanation: The specified line in the /CROSS_CODES file contains only one
of the expected pair of values, or an alphabetic character in the IFF
feature code field.

User action: Correct the offending /CROSS_CODES file line. Re-run I2MOSS.

UNEXPEOF, unexpected end of IFF file

Explanation: I2MOSS has detected the end of the IFF file, but has not
detected an IFF 'EJ' entry.

User action: Use IMEND on the file, which will correctly position the EOF
marker and insert an EJ entry at the end of the file. Re-run I2MOSS on the
corrected file.

WREOT, error writing tape mark on 'device name'

Explanation: I2MOSS has received a message from the mag tape i/o routines
that an error occurred while writing a tapemark to the tape.

User action: The accompanying message should indicate the cause of the
error. Check that the tape is still loaded, check that the tape unit is
online, check that that tape is not write-protected. If the problem
continues, notify the system manager.

MESSAGES (fatal)

These messages indicate a fatal error in processing which has resulted in
immediate program termination and may indicate an error in the program itself.
It is most unlikely that the output file will be correctly processed.
Generally, in the event of a processing error I2MOSS will delete the MOSS output
file before calling exit, (unless writing to magnetic tape).

INSUFBUF, insufficient coordinate buffer space for IFF feature FSN 'integer'
    ('integer')

    Explanation: I2MOSS buffers up all the coordinates from all IFF ST entries
    for a single IFF feature before outputing to MOSS file. The specified
    feature has more than 50000 points. There is insufficient internal buffer
    space for I2MOSS to hold any more points.

    User action: Check that this is a valid IFF feature. If it is then it will
    have to be split into smaller features none of which exceed 50000 points.

MESSAGES (other)

In addition to the above messages which are generated by I2MOSS itself, other
messages may be produced by the command line interpreter (CLI) and by Laser-Scan
libraries. In particular, messages may be generated by the IFF library.   These
are introduced by '%IFF' and are documented in the IFF library users' guide. In
most cases IFF errors will be due to a corrupt input file, and this should be
the first area of investigation. If the cause of the error cannot be traced by
the user and Laser-Scan are consulted, then the input file should be preserved
to facilitate diagnosis.