ODB++ SPECIFICATION

Version 7.0

Published December 2010

This document is for information purposes. Mentor Graphics reserves the right to make changes in specifications and other information contained in this document without prior notice, and the user should, in all cases, consult www.odb-sa.com to determine whether any changes have been made.

ODB++ Format and the ODB++ Specification are the sole property of Mentor Graphics or its licensors. Mentor Graphics does not authorize any unapproved derivative versions of the ODB++ Format. In obtaining this document, the user shall not harm Mentor Graphics' interest in the ODB++ Format in any way, including but not limited to: (i) renaming the ODB++ Format; (ii) communicating that the ODB++ Format is owned by anyone other than Mentor Graphics; (iii) associating the ODB++ Format with data that does not conform to the ODB++ Format; or (iv) modifying the graphical logo representing the ODB++ Format.

Download of the ODB++ Specification does not grant a license to develop software interfaces based on the format specification. We openly encourage users to apply for a license to develop ODB++-based software as a Solutions Development Partner within the ODB++ Solutions Alliance at www.odb-sa.com. No representation or other affirmation of fact contained in this publication shall be deemed to be a warranty or give rise to any liability of Mentor Graphics whatsoever.

MENTOR GRAPHICS MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF INTELLECTUAL PROPERTY.

MENTOR GRAPHICS SHALL NOT BE LIABLE FOR ANY DIRECT, INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS PUBLICATION OR THE INFORMATION CONTAINED IN IT, EVEN IF MENTOR GRAPHICS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

U**.S. GOVERNMENT LICENSE RIGHTS:** The documentation was developed entirely at private expense and is commercial computer software documentation within the meaning of the applicable acquisition regulations. Accordingly, pursuant to FAR 48 CFR 12.212 and DFARS 48 CFR 227.7202, use, duplication and disclosure by or for the U.S. Government or a U.S. Government subcontractor is subject solely to the terms and conditions set forth in the license agreement provided with the documentation, except for provisions which are contrary to applicable mandatory federal laws.

TRADEMARKS: The trademarks, logos and service marks ("Marks") used herein are the property of Mentor Graphics Corporation or other parties. No one is permitted to use these Marks without the prior written consent of Mentor Graphics or the owner of the Mark, as applicable. The use herein of a third-party Mark is not an attempt to indicate Mentor Graphics as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A current list of Mentor Graphics' trademarks may be viewed at: www.mentor.com/trademarks.

Mentor Graphics Corporation
8005 S.W. Boeckman Road, Wilsonville, Oregon 97070-7777
Telephone: 503.685.7000
Toll-Free Telephone: 800.592.2210
Website: www.mentor.com

Table of Contents

Chapter 6 Job>steps>layers Entity

Chapter 11 Symbol Definitions

Chapter 1 *Overview*

Introduction

This book contains the full description of the ODB++ CAD/CAM/DFM data exchange format. ODB++ is widely accepted within the electronics industry as an efficient way to move printed circuit bare-board, assembly and test data on the manufacturing-engineering level within design/manufacturing supply chains. It is designed as a simple yet comprehensive description of all entities needed in the manufacturing of a printed circuit board.

Intended Readers

This book is intended for those interested in implementing the ODB++ format, for CAD/CAM applications and for interfacing to logistical supply-chain processes.

Versioning

This specification describes the latest version of ODB++. Subsequent updates to the specification will follow sequentially, independent of software program versions. The following tables describe Valor/Frontline products/releases and the ODB++ versions supported.

Product	Release	Supported ODB++ Versions (Up to and including)
Trilogy/vSure	9.0	7.0
Trilogy/Enterprise	8.2	7.0
Trilogy/Enterprise	8.1	7.0
Trilogy/Enterprise	8.0	7.0
Trilogy/Enterprise	7.6	6.5
Trilogy/Enterprise	7.5	6.5[1]
Trilogy/Enterprise	7.3	6.5[1]
Trilogy/Enterprise	7.2	6.2
Trilogy/Enterprise	7.1	6.2
Trilogy/Enterprise	7.0	6.2
Trilogy/Enterprise	6.3	6.1
Trilogy/Enterprise	6.2	6.1
Trilogy/Enterprise	6.1	6.1
Trilogy/Enterprise	6.0	6.0

Product	Release	Supported ODB++ Versions (Up to and including)
Trilogy/Enterprise	5.3	5.3
Trilogy/Enterprise	5.2	5.2

1 ODB++ version 6.4 and 6.5 files generated from the Frontline GenFlex product may contain a layer type **`MASK`** not presently supported by Enterprise / Trilogy version. A dialog box appears informing the user that the unknown layer will be changed to **`DOCUMENT`** and the context of the layer changed to **`MISC`**. Data contained in **`ADD_TYPE`** and **`COLOR`** is lost.
As of ODB++ version 7.0, the layer type **`MASK`** is fully supported along with user-defined layer subtypes.

Product	Release	Supported ODB++ Versions (Up to and including)
inCAM	2.0	7.0
inCAM	1.2	7.0
GenFlex	2.6	6.4
GenFlex	2.5	6.4
GenFlex	2.3	6.4
GenFlex	2.2	6.4
GenFlex	2.1a	6.4
GenFlex	2.1	6.4
GenFlex	2.0c	6.4
GenFlex	2.0b	6.4
GenFlex	2.0	6.4
GenFlex	1.7	6.4
GenFlex	1.6b	6.4
GenFlex	1.6	6.4
GenFlex	1.5c	6.4
GenFlex	1.5b	6.4
GenFlex	1.5	6.4
GenFlex	1.4	6.4
Genesis	9.6	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.5	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.3	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.2	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.1d	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.1c	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.2b	6.2 (reads 6.3, exports 6.2)[2]

Product	Release	Supported ODB++ Versions (Up to and including)
Genesis	9.1	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.0	6.2 (reads 6.3, exports 6.2)[2]
Genesis	9.0	6.2 (reads 6.3, exports 6.2)
Genesis	8.2c	6.2 (reads 6.3, exports 6.2)
Genesis	8.2b	6.1
Genesis	8.2	6.1
Genesis	8.1b	6.1
Genesis	8.1	6.1
Genesis	8.0b	6.1
Genesis	8.0	6.1
Genesis	7.2c	6.1
Genesis	7.2b	6.1
Genesis	7.2	6.1
Genesis	7.1c	6.1
Genesis	7.1b	6.1
Genesis	7.1	6.1

2 Genesis versions 9.0 and earlier convert the **MASK** layer type to **SIGNAL**. Subsequent versions convert the **MASK** layer type to **DOCUMENT**, as does Trilogy / vSure.

Updates

This section lists the changes made to ODB++ and to the documentation since February 2000.

ODB++ V.7.0 November 2007

(Released as V.7.1 for documentation purposes only.)

- Entity names must not begin with the characters hyphen (-) or plus (+). This is in addition to the previously not allowed character, dot (.).
- System attributes are no longer considered core entities, and may differ between Trilogy / Enterprise and Genesis applications. For a current list of supported attributes, see System Attributes and System Attributes for Genesis, respectively.

ODB++ V.7.0 September 2007

<job_name>/steps/<step_name>/stephdr, **STEP-REPEAT** array

- Steps rotated at any angle are supported. See the **STEP-REPEAT** array, **ANGLE** parameter.
- Flipped steps are supported. See the **STEP-REPEAT** array, **FLIP** parameter.

<job_name>/steps/<step_name>/layers/<layer_name>/features

- Feature transformation functionality allows for any angle rotation for pad and text features, including barcode; and the resizing of features, instead of creating special symbols. See the **orient_def** parameter for pad, text and barcode features.
- Features and coordinates are saved in the units in which they were created to eliminate the loss of precision due to rounding. See the example under features.

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/meas_p

- Checklists can be read in both inches and millimeters. See the example under meas_p.

<job_name>/matrix/matrix

- User-defined layer types are permitted in the layer field **ADD_TYPE**.

Eight new semi-standard symbols added:

- Square/Round Donut
- Rounded Square Donut
- Rectangle Donut
- Rounded Rectangle Donut
- Oval
- Rounded Square Thermal /Rounded Square Thermal (Open Corners)
- Rounded Rectangle Thermal / Rounded Rectangle Thermal (Open Corners)
- Oval Thermal / Oval Thermal (Open Corners)

ODB++ V.6.5 *August-December 2005*

Layer Production Data parameters have been updated for the **LPD** file and an inclusive **LPD_MULTIPLE** file has been added. See "lpd (Layer Production Data)" on page 145.

Supplementary files added to support Orbotech DI machines. *Used by Frontline applications only.* See "DI (Orbotech Direct Imaging Interface)" on page 156.

ODB++ V.6.4 *September 2004*

Layer type **MASK** has been added to the job matrix to accommodate the Frontline GenFlex product along with the two layer fields: **ADD_TYPE** and **COLOR**. See **MASK**, **ADD_TYPE** and **COLOR** in Job Matrix.

ODB++ V.6.3 *July 2004*

CNSA_NET_TYPE_CLEARANCES dependent upon area constraints have been greatly expanded to facilitate those clearances used by Cadence Allegro. In order to maintain backward compatibility and reduce the number of redundant records which could result, **CNSA_KEY_NET_TYPE_CLEARANCES** has been added to assign a name, **main_set_name** = **<set name>** to an existing record of net type

clearances with the same key values (i.e. **constr_area**, **net_type1**, **net_type2**, and **layers**). This **<set name>** is used to create new clearance records with the same specified clearances as those found in **main_set_name**. See "Dependent Upon Constraint Area" on page 93 and **CNSA_KEY_NET_TYPE_CLEARANCES**.

ODB++ V.6.2 *February 2004*

Two record types added to the **net_prp** file:

- **NET_ELECTRICAL_PARAMETERS** contains the electrical parameters of a net designated by **set_name** read from Cadence Allegro.
- **NET_ECSET_ENTRY** links a specific CAD net to an electrical parameter set.

See "Net Type Electrical Parameter Set" on page 96 and "Electrical Set Entry Record" on page 97.

November 2003

Description Aliases, added to the BOM entity to enable 10 descriptive CPN fields and 10 descriptive MPN fields to be replaced with user-defined fields in any of the possible languages of Environment Variable **GENESIS_LANG**. See "# Description Aliases" on page 79.

NET_TYPE_CLEARANCES as well as **NET_TYPE_PHYSICAL_PARAMS** can now be dependent upon "constraint areas". See "Dependent Upon Constraint Area" on page 93.

ODB++V.6.1 (B.06) *June 2003*

The following ODB++ entities have been affected by encryption:

<job_name>/steps/<step_name>/eda/vpl_pkgs

(See "vpl_pkgs" on page 91.)

<job_name>/steps/<step_name>/layers/<layer_name>/components3

(See "components3" on page 121.)

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/def/hdr_p

(See "def/hdr_p" on page 185.)

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/hdr_p

(See "res/hdr_p" on page 186.)

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/report/text_p

(See "report/text_p" on page 187.)

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/meas_p

(See "meas_p" on page 189.)

The **components3** file replaces **components2** representing component data after processing with Assembly Merge (BOM Merge, Library Merge and Board Merge). See "components3" on page 121.

The following ODB++ entities have had the letter 'p' appended to their names.

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/report/tags_p
<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/disp_p

ODB++ V.6.0 *March 2002*

<job_name>/stackups/<stackup_name>/stackup

Addition of a **sub_lam** record.

<job_name>/steps/<step_name>/netlists/cadnet/netlist

New parameter added to net point description:

is_shrink	Y - point size was shrunk to fit solder-mask opening. N - point size is limited only by pad size.

ODB++ V.6.0 (B.04) *September 2001*

<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/cdrhdr2

- These ("<set_name>cdrhdr2 (CDR14 Header - Additional)" on page 136) are CDR parameter values in addition to those in "<set_name>/cdrhdr (CDR Header)" on page 135. All the files under the **cdr_sets** directory are new, as below:

.../cdrhdr
.../cdrhdr2
.../steps/cdr14_stp_main
.../steps/cdr14_stp_pos
.../steps/cdr14_stp_neg
.../steps/clone_<step_name>
.../steps/clone_<step_name>_pos
.../steps/cdr14_stp_on_clones
.../steps/user_def_<step_name>
.../steps/user_def_<step_name>_pos

<job_name>/steps/<step_name>/layers/<layer_name>/cdr14

- The following directories ("user_def_<step_name> (Steps in AOI)" on page 134) have been introduced:

.../steps/user_def_<step_name>
.../steps/user_def_<step_name>_pos

<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/header

NCD header and related NCD entities introduced, as follows:

Header - **.../<layer name>/ncd/<ncd-set name>/header**

Table - **.../<layer name>/ncd/<ncd-set name>/table**

Drill - .../**<layer name>/ncd/<ncd-set name>/drill/<split number>.<stage number>**

<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/header

NCR header - **.../layer name>/ncr/<ncr-set name>/header**

<job_name>/steps/<step_name>/layers/<layer_name>/tools

Drill Tool entity - .../<layer_name>/tools

August 2001

To save the new types of results (Scalar and Text) generated from the two DFM functions **ODB_LAYER_GET_SHAPE_COMP** and **ODB_RES_SET_MEAS_ID_TEXT** with the rest of the results of a checklist, letters that indicate these types have been added to the job file:

<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/meas

Changes/additions are underlined in the Measurement Identification Structure:

<meas_num> <cat_num> <disp_num> <alarm> <ftype1> <fsym1> <ftype2> <fsym2>

<meas_num>	Serial number of measurement (0 and up). A dash (-) prefix signifies a reference measurement.
<cat_num>	Category number (0 and up) which must refer to a valid category in the **res/hdr f**ile.
<disp_num>	The display record number (0 and up) which must refer to a valid category in the **res/sres/<layer_name>/disp** file.
<alarm>	**N** (no alarm) or **Y** (alarm). Action may generate alarm measurements which can be listed together with tags, in the textual report.
<ftype>	Type of feature which contributed to the measurement: **L** Line **P** Pad **S** Surface **A** Arc **T** Text feature **C** Component (top) **c** Component (bottom) **N** Net **D** Diff. pair **X** Free text **V** Scalar value **U** Unit-sensitive scalar value **Q** Square area scalar value
<fsym>	Symbol of feature/component which contributed to the measurement: For **L,P,S,A** and **T** - name of a valid symbol For **C** or **c** - reference designator of the component For **N** - name of the net For **D** - name of the differential pair net For **X** - a text string (without spaces) For **V**, **U** and **Q** - a scalar value

ODB++ V.5.3 September 2000

<job_name>/steps/<step_name>/chk/<checklist_name>/ actions/<action_num>/res/sres/<layer_name>/meas

- New symbols for **<ftype>** and **<fsym>**. See "meas_p" on page 189.

<job_name>/steps/<step_name>/eda/data

- New net Attributes. See "Job>Steps>et (Electrical Test)" on page 190.

<job_name>/steps/<step_name>/eda/net_prp

- New net type clearances. See "<job_name>/steps/<step_name>/eda/net_prp" on page 92.

ODB++ V.5.2 *February 2000*

<job_name>/steps/<step_name>/stephdr

- Two new fields added (**AFFECTING_BOM** & **AFFECTING_BOM_CHANGED**). See "stephdr (Step Header)" on page 71.

<job_name>/steps/<step_name>/eda/data

- Two new fields added to the PIN Record Structure (**<etype> & <mtype>**). See "Job>Steps>et (Electrical Test)" on page 190.

<job_name>/steps/<step_name>/netlists/cadnet/netlist

- New parameters added to the Netlist (x, **e, & by**). See "Netlists" on page 73.

<job_name>/steps/<step_name>/layers/<layer_name>/components2

- New Job Entity. The **components** file describes the original EDA data for a component, while the **components2** file represents component data after processing with Assembly Merge (BOM Merge, Library Merge and Board Merge).

Conventions and Terminology

Entity Definitions

Core

Data entities marked as "core" contain data that form an essential part of modelling the Printed Circuit Assembly (including all aspects of the PCB bare-board). In essence, "core" entities contain all the information necessary for CAM systems to prepare PCB fabrication and assembly operations.

Supplementary

Non-core entities (supplementary) are included in the ODB++ format to support certain CAM and DFM functions specific to certain solution vendors. These supplementary entities are open to all, and are maintained in accordance with the specification, in the same way as the "core" entities.

Hyperlinks

The"Charts" on page 29 are tree-charts that describe the structure of each entity in a job. Blue nodes, usually at the end-nodes of the trees, are hyperlinked. Click to jump to the detailed description of the entity/element.

Chapter 2 *Design Principles*

File System

Hierarchy

ODB++ uses a standard file system structure. A job in ODB++ is represented by a stand-alone directory tree that can be transferred between systems without any loss of data.

The advantages of a directory tree compared to one large file are apparent when a job is being read from disk or saved to disk. The flexible tree structure allows only a small part of the job to be read/saved, avoiding the overhead of reading and writing a large file.

When a job tree has to be transferred to another system, standard 'tar' and compression utilities can be used to convert a directory tree into one flat file.

Mandatory / Optional Files

The following list specifies the files that are mandatory, while those not mentioned are optional:

For the Job:

`job/matrix/matrix`

For each Step defined in the Matrix:

`job/steps/<step_name>/stephdr`

For each Layer defined in the Matrix:

`job/steps/<step_name>/layers/<layer_name>/features or`
`job/steps/<step_name>/layers/<layer_name>/features.Z`

There are also links between files that are implicitly defined in the ODB++ definition which create dependencies between one file and another. For example,. the **`/<step_name>/layers/comp_+_top/components`** file contains links to **`/stepname/eda/data`**.

Legal Entity Names

Job name	Layer name	Attribute name
Step name	Symbol name	Attribute string

ODB++ entity names must follow these rules:

- The length of any name should not exceed 64 characters. However, user attribute strings (not names) are determined by the **MAX_LEN, MIN_LEN** fields in the **<job_name>/misc/userattr** Job File (see <job_name>/misc/userattr)
- Use only the following:
  - lower case letters ('a' through 'z')
  - digits ('0' through '9')
  - punctuation - dash (**-**), underscore (_), dot (**.**) and plus (**+**)
- Names must not start with dot (**.**), hyphen (-), or plus (+) with the exception of attributes which can start with (**.**).

Readable ASCII files

All files in ODB++ are readable ASCII files except those which are Intellectual Property (IP) of Valor Computerized Systems. (This includes Valor VPL packages and checklist data structure.) This concept provides the advanced user with the capability to read database files for understanding. In contrast, binary databases which are still used in older systems prevent the user from reading database files directly and require a special extraction program to retrieve all or part of the database.

In all files, the **#** character specifies a comment. Lines which start with this character are ignored by the system and are only used for readability.

The line separator can be either <LF> or <CR><LF>, depending on operating system and platform.

Units of Measurement

All units are either imperial units (inches, mils) or metric (mm, micron), depending upon the units directive placed at the beginning of the file. If not defined, the default is imperial.

System resolution is 1/10160 mil or 1/400 of a micron. As a result, minimum line width is 1/400 of a micron wide. Minimum measurable distance, or placement tolerance for any feature is also 1/400 of a micron.

Large File Compression

One of the reasons vendors have chosen binary databases in the past was the need to conserve space on hard disks. Modern compression techniques are available today and provide excellent compression ratios, especially for ASCII files with repetitive patterns. Large files in ODB++ are saved in standard UNIX compress format. The compression is optional, and any reader of ODB++ database should expect some files to be in either compressed format (.Z suffix) or without compression. The files which are potentially compressed are clearly identified in the following material.

Sum File

Many of the files in ODB++ have an attached hidden file which provides information about them. The name of the attached file is:

```
.<name>.sum
```

The file contains the following information:

Size - size of the data file

Sum - checksum of the file (can be enabled/disabled as a configuration parameter)

Date - date in which the file was written, where
format is mm/dd/yy before software version 4.3
format is mm/dd/yyyy after version 4.3

Time - of writing

Version - version of the software in which the file was saved

User - user operating the software when file was last saved

Note No verification of the size and sum is done today when the file is read by the system. This was intended to allow advanced users to modify files manually in extreme cases.

Example of a sum file:

```
SIZE=274
SUM=-1
DATE=05/24/97 (after version 4.3 = 05/24/1998)
TIME=20:05:10
VERSION=03.02 (BUILD 00 FOR HP-UX)
USER=MOSHIK
```

Structured Text files

To improve readability, many of the small files in ODB++ contain expressions of the type:

```
<var>=<value>
```

The main advantages of this structure is readability. The user can open a file and understand its contents without having to refer to external sources.

Example (from the **stephdr** file):

```
X_DATUM=0.3
```

A more elaborate structure, which appears in some structured files, describes arrays. Arrays are lists of elements, each one containing several fields. An array element has the following structure:

```
<array_name> {
 <var>=<value>
 <var>=<value>
 ....
}
```

This element will appear a number of times, each time defining an element of the array.

Example (from the `matrix` file):

```
LAYER {
   ROW=1
   CONTEXT=BOARD
   TYPE=COMPONENT
   NAME=COMP_+_TOP
   POLARITY=POSITIVE
   START_NAME=
   END_NAME=
   OLD_NAME=
   ADD_TYPE=MICRO_VIA
   COLOR=606090
}
```

Line Record Text files

Some of the files in the database are relatively large and saving them as structured text files is impractical. These files are saved as line record text files. Each line contains a multitude of fields, typically separated by space characters. Reading or writing such files without proper reference information is more difficult. Typically, the first character or word in each line defines the type of record which the line describes. In many cases, the line order is important. Certain lines require that the following line will exist in a particular sequence.

The maximum characters in one line are, in general, 500 characters, however there are exceptions. Any line over the defined limit will be truncated.

Example (from the feature file):

```
#
#Feature symbol names
#
$0 r50
$1 r70
$2 r80
$3 r93
$4 ths80x60x0x4x15
#
#Feature attribute names
#
@0 .geometry
@1 .pad_usage
#
#Feature attribute text strings
#
&0 systest_board
&1 term_1
&2 via_1
#
#Layer features
#
```

```
L -0.4 -3.6 -0.4 -4 0 P 0;0=0
L -0.4 -4 0 -4 0 P 0;0=0
L -0.4 -1.4 -0.4 -1 0 P 0;0=0
L -0.4 -1 0 -1 0 P 0;0=0
```

Note Hole symbols cannot appear in a feature file.

Angles

Angles are mainly used to position spokes of thermals and to rotate SMDs. The following rules apply:

- Angle values are expressed as integers within the range 0-359, with angle 0 due East with positive values measured counter-clockwise.
- Angles for rectangular thermals can be in 45 degree increments only, whereas they can be other than multiples of 45 degrees in square/round thermals (when not in 45 degrees, the spoke gap will lie along a line extending from the center).
- When the start and end-point of an arc coincide, it is considered a 360 degree arc. There are no single-point arcs in ODB++.

Rotation / Mirroring

- Feature pads are oriented at 90 degree increments, rotated clockwise.
- Mirroring is only on the X axis (left to right, changing X coordinates).
- Diagonal square lines look like rotated rectangles; the endpoints are also rotated (they are not orthogonal).

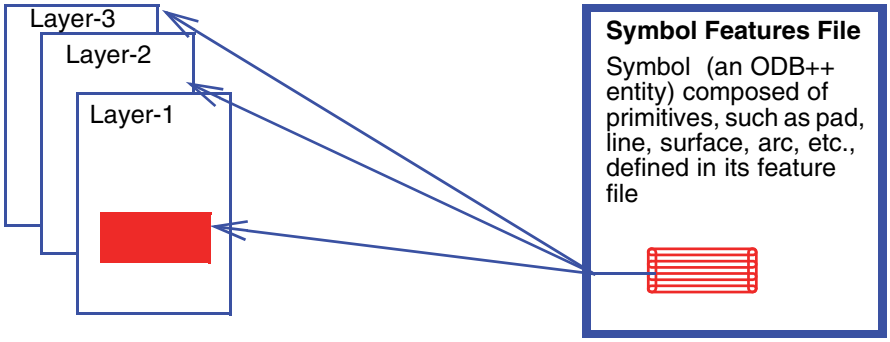
Coordinates

- Coordinate units in feature and symbol files are given in inches with a decimal point. Coordinates are always expressed in inches; but symbol sizes are expressed in microns.
- When you specify an x,y location for a text string the bottom left of the first character is positioned at the coordinates.
- Point coordinates in a netlist file represent the center of pads.

Symbols

Symbols define a wide variety of shapes (see below) that are mostly used to draw pads. A symbol is an ODB++ entity that is defined once and used many times in order not to repeat the definition of a group of features in a layer. A symbol contains a ‘features’ file that has a number of primitive features (such as pad, line, surface, arc, etc.) that compose the symbol in a layer.

A symbol can be referenced from a number of layers in the job at different coordinates. Changing the symbol definition will automatically cause all its representations in the layer/s to change accordingly.



Symbol can be referenced in many layers. When changed, symbols will automatically change in all layers where defined.

ODB++ supports the following types of symbols:

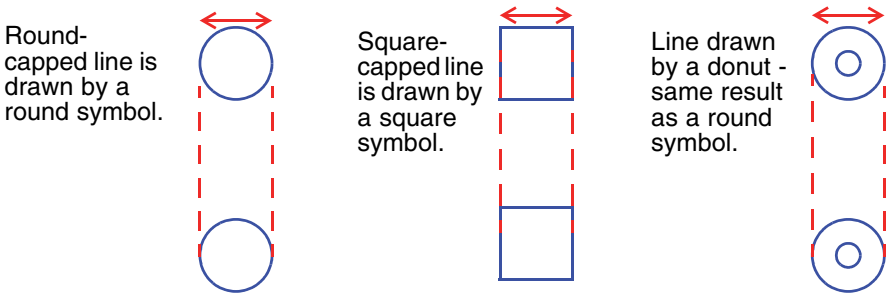
- Standard
- Special (User-Defined)

Standard Symbols

Standard symbols are generated dynamically by the system from their names. They do not require a special graphic symbol entity to be saved in the database. They are round, square or parametric shapes.

For lines, symmetric symbols (where width=height) draw lines with width equal to the width of the symbol, as in figure below:

Width of a line is the width of the symmetric symbol used to draw it.



For example, **`r30`** is automatically generated as a circle feature with a diameter of 30 (mils), **`s200`** is a square with a 200 (mils) diameter (side of square).

Arcs can be drawn with round symbols only.

Pads can be drawn with any symbol. Examples of symbols to draw pads are rectangles = rect width x height (e.g., **`rect100x200`**), ovals = oval width x height (e.g **`oval77x90`**), octagons = hex_s width x height x corner size (e.g. **`hex_s30x50x12`**), and many more.

Units are in imperial units (inches mils).

For example to define a round capped line of width 10 mils, use the symbol '**r10**'. To define the same type line but with a width of 20 mils, specify '**r20**'. To define a square capped line with a width of 10 mils, use '**s10**'.

Drawing a line with an asymmetric symbol generates a one-pixel line ending with the symbol at both ends.

Standard symbols are all positive filled shapes. Holes in symbols are see-through by definition. The internal implementation of complex symbols uses arcs or contour data with cutouts.

Special (User-Defined) Symbols

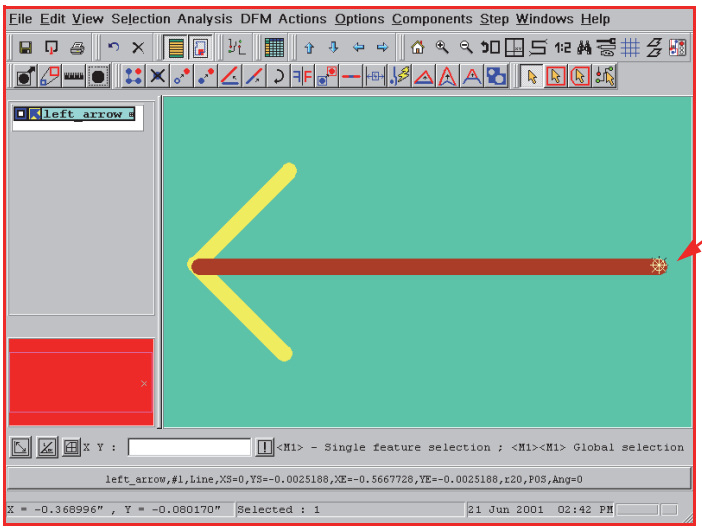
Special symbols are user-defined symbols which have a full graphical description stored in feature files in the job's symbols subdirectory. They can contain any number of features. Special symbols are defined for a job usually for shapes not found among the standard symbols. Special symbol names cannot be identical to those reserved for standard symbols.

User-defined symbols can be saved within the system and used when needed. The system recalls the graphic shape defined by the user.

Note It is always preferable to use a standard symbol, where possible. Special symbols are represented by contours in the shapes list of layer features. Contours require more memory than standard symbols and a great number of them will slow down system processing.

Special symbols are not scalable, such as standard symbols. The reason is that a specific feature file definition is created for each special symbol that defines its shape. Therefore, you need to create a new symbol for each set of parameters. It is preferable to name the symbol to indicate its dimensions.

The figure shows an arrow whose origin (0,0) is at the tip of the butt.



This special symbol is drawn from lines (standard symbols)

When inserting this symbol, the insertion point is at 0,0. This means it is a left arrow

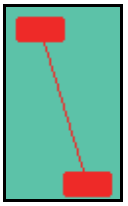
Symbol Characteristics

Asymmetric vs. Symmetric

When asymmetric symbols (such as rectangles) are used to draw diagonal lines, the lines are single-pixel lines whose end-points are the symbols used to draw the lines (see figure below).

When drawn orthogonally (horizontal: y_start= y_end, or vertical: x_start=x_end) the output line has the height/width of the asymmetric shape from start point to end point (the line is created by "dragging" the symbol from one end of the line to the other in the same orientation as it was placed).

For example, if the symbol in the figure is dragged vertically, the width of the line will be the width of the symbol. If dragged horizontally, the width will be the height of the symbol.



Note Single-pixel line width is expressed in the internal software resolution of 10160 pixels = 1 mil, or 400 Pixels = 1 micron, meaning that single-pixel lines are 1/400 of a micron wide.

If the symbol used to draw a line is symmetrical (square or round) the generated line is the side or diameter of the symbol. A round symbol (whose name starts with r) generates round line ends, a square symbol (s) generates square line ends.

For arcs, only round symbols can be used.

Holes in surface features are transparent (empty).

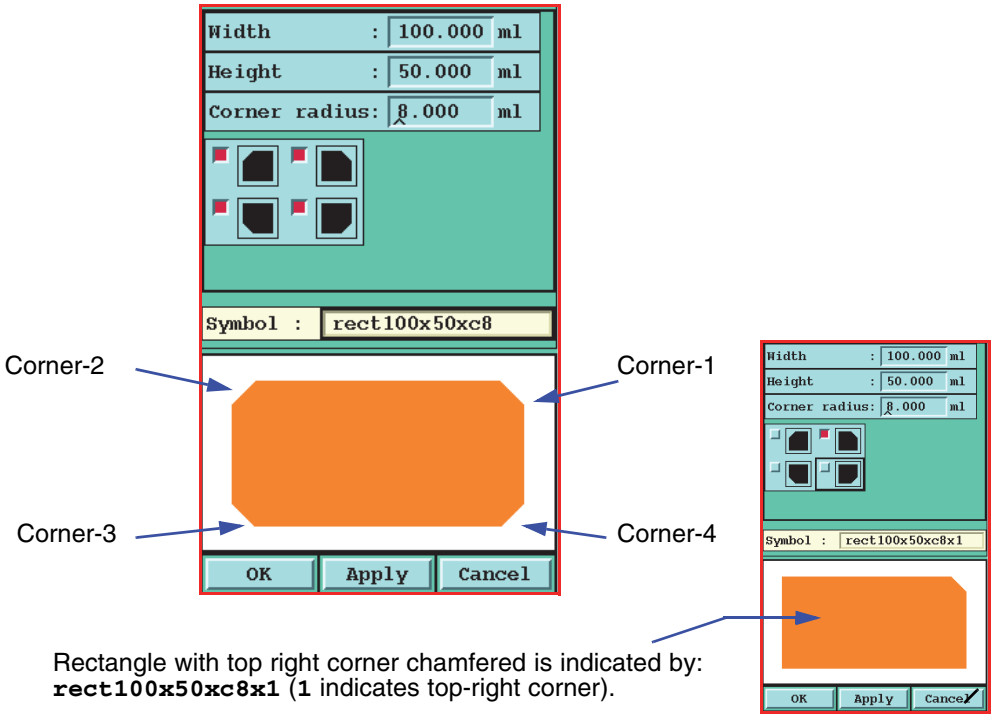
Dimensions of standard symbols can be in non-integers with resolutions up to 1/100 mils

Symbol `r0` is a legal entity represented by a single pixel width.

The standard Octagon symbol corner size is the distance between the bounding box corner and the vertex.

Rounded/Chamfered Rectangles

The corners of rounded/chamfered rectangles can be specified in ascending order counter-clockwise, starting from the top-right corner, as in the following figure:



Shape

An internal geometrical entity (that may consist of a number of features) used by the system during algorithmic operations. Shapes are always positive. They include points, segments, curves, lines, arcs, squares, rectangles and contours(g). For example, a shape can be a contourized(g) shaved pad that consists of a pad and a feature that shaves it.

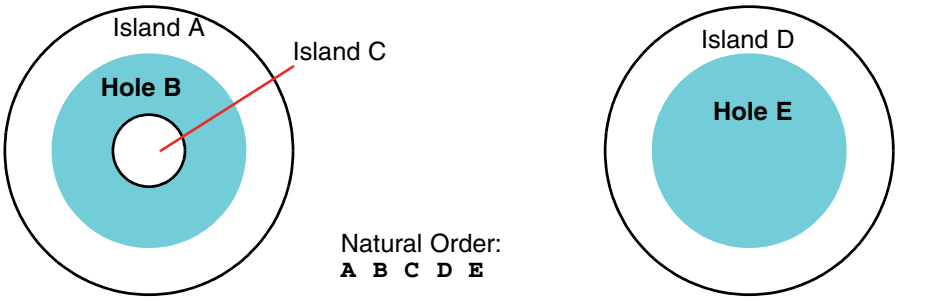
Shapelist

An internal data structure which, during analysis, is created for a layer upon demand. The shapelist simplifies the representation of a layer within the system by dealing with multiple polarities, odd shape symbols, etc. A Shapelist can be deleted to improve memory usage; it will be rebuilt by the system automatically when needed.

Order of Holes/Islands in Surfaces

The order of containment of holes and islands within surfaces determines their natural order. The outermost island comes first. Islands precede holes that are

contained in them. Holes precede islands that are contained in them. Take, for example, the following containment order:



Chapter 3 Job Tree

Job Overview

An ODB++ job can include a number of entities, accessed by a specified editor represented by icons in the Engineering Toolkit. (All entities, including those not supported by the Graphic User Interface (GUI), are discussed in "Charts" on page 29.) The following are currently supported in the GUI:

Steps, which are multi-layer entities (e.g. a single image, a sub panel array, a production panel or a multi layer coupon). Each step contains a collection of layers. Layers are two-dimensional sheets, containing graphics, attributes and annotation. Layers express physical board layers, mask layers, NC drill and rout layers and miscellaneous drawings. All steps in one job have the same list of layers, albeit the contents may be totally different.

A **Matrix**, in which the rows are the job layers and the columns are the job steps. The matrix contains for each row additional information such as the type, polarity and context. The matrix is also crucial in defining the physical order of the layers and the relation of drill layers (through, blind, buried, etc.).

Symbols, single layer graphic entities which can be referenced from within any graphical layer in a step.

Work Forms, user defined collection of fields (textual and graphical) and buttons.

Attributes, user-defined attributes to facilitate automation.

Wheels, aperture tables created in the Wheel Editor Popup.

Input, automatically identifies the format type of the incoming data (Gerber, Excellon drill, etc.) and interprets the Gerber wheel based on predefined wheel templates.

Output, multiple format translators to choose the output device.

User, where user can store his own files.

Extension, used for third party data files.

Log, intended for job specific log files.

Stackups  and Work Flows  may appear if the job contains legacy data.

Each Step entity contains, in addition to general information and the list of layers, several other important subentries:

- Step & repeat information (in the stephdr file), specifying any previous steps which are included in this one and their relative location and orientation.
- Up to three netlists of the step (CAD netlist, reference netlist and current netlist).
- An EDA object, containing data regarding the component packages and pins. It also contains information about the relation of features in the board layers to specific design nets and properties imported from the EDA system.
- An unlimited number of checklists, each one is composed of analysis or DFM actions. An action contains the definitions (parameters to run with) and the results (measurements) of the last successful run.
- A profile which is a schematic border around the step.

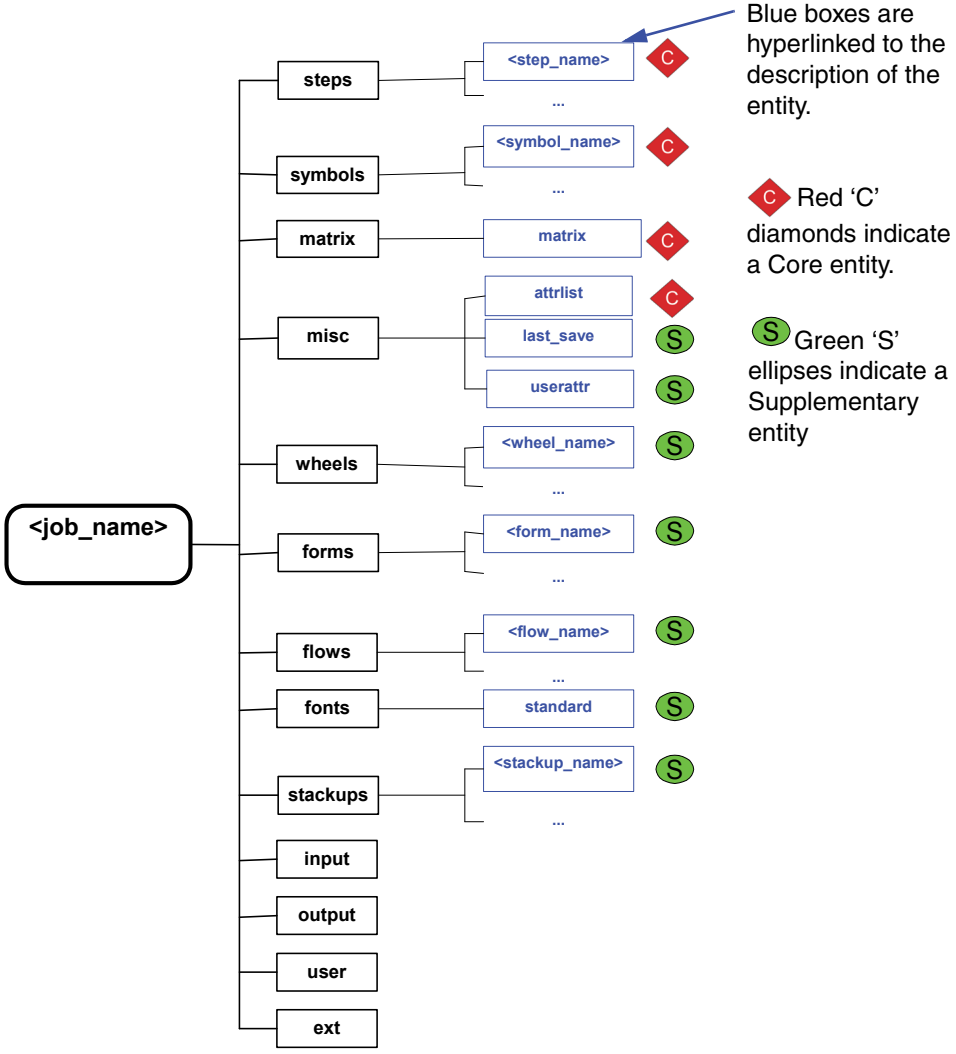
Charts

An ODB++ job is a directory, containing a large number of sub-directories. The following charts describe all main- and sub-directories in chart form. Click on blue nodes to jump to a full description of that entity. Users can create additional directories —for example, **`xml`**, **`whltemps`**, etc.—as long as they do not conflict with ODB++ standard directories. Directories, no longer supported, may appear in ODB++ jobs containing legacy data. This is noted in the specification.

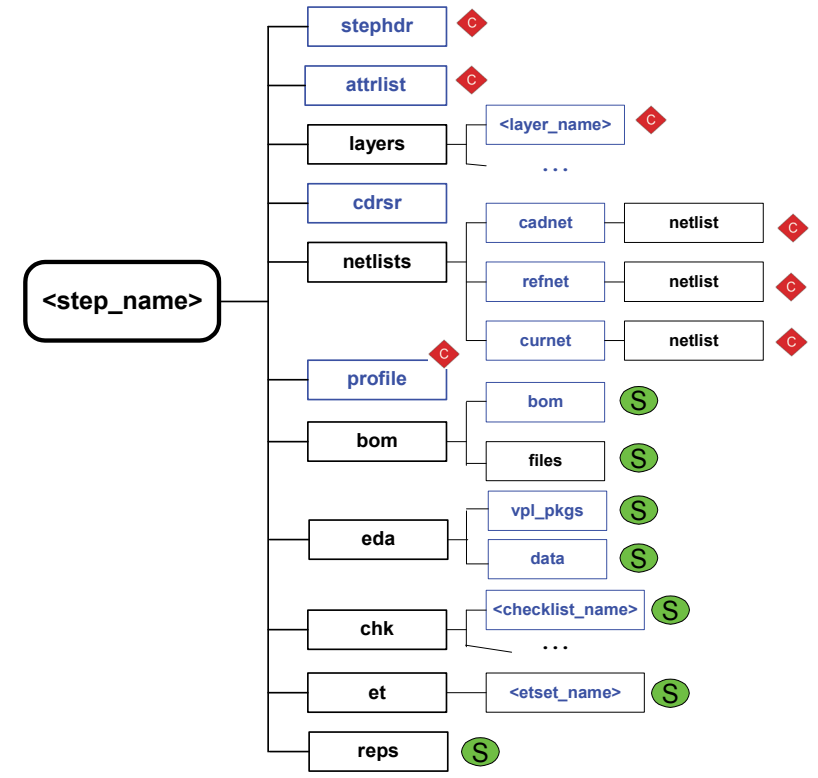
indicates a Core entity that contains data essential to modelling a PCB.

indicates a Supplemental entity with ODB++ format that supports certain CAM and DFM functions.

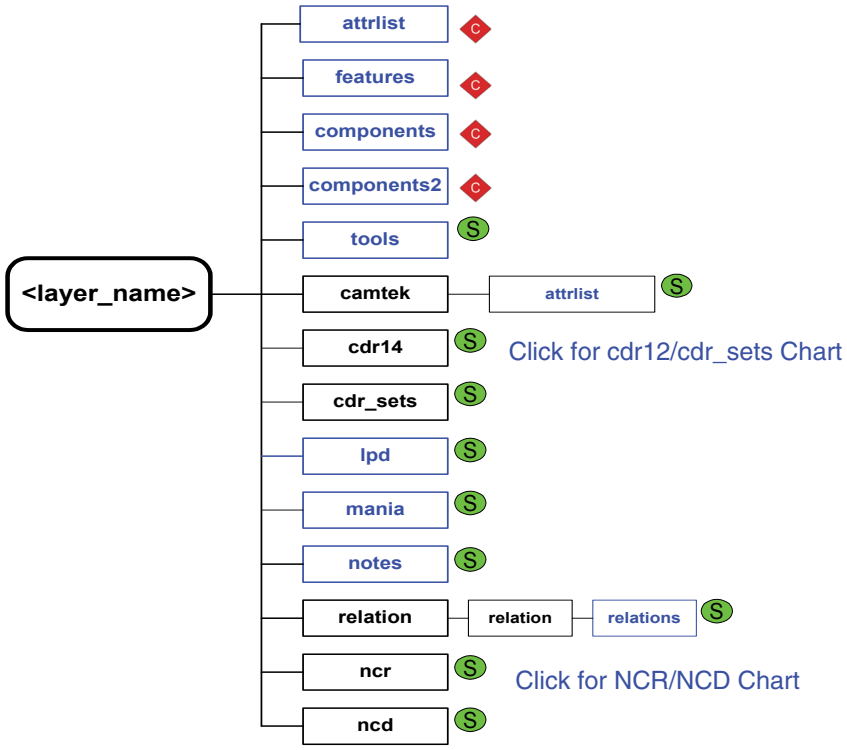
Job Chart


<job_name>
steps
<step_name>
...
symbols
<symbol_name>
...
matrix
matrix
misc
attrlist
last_save
userattr
wheels
<wheel_name>
...
forms
<form_name>
...
flows
<flow_name>
...
fonts
standard
stackups
<stackup_name>
...
input
output
user
ext
C
S
Blue boxes are hyperlinked to the description of the entity.
Red 'C' diamonds indicate a Core entity.
Green 'S' ellipses indicate a Supplementary entity

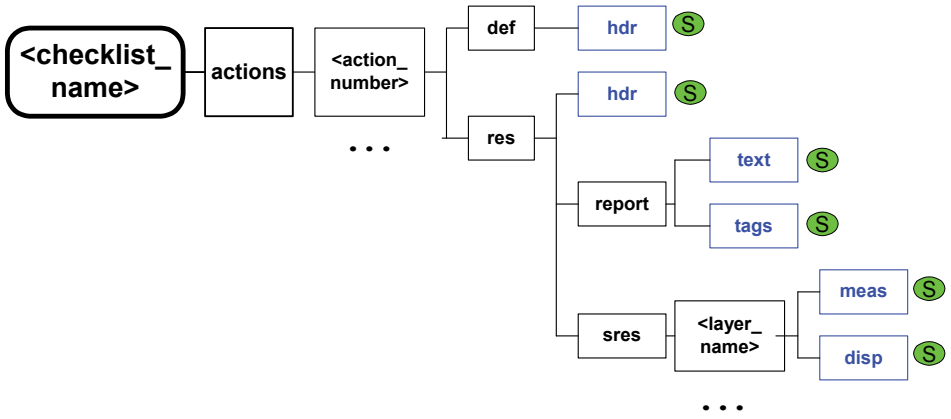
Job>steps



Job>steps>layers

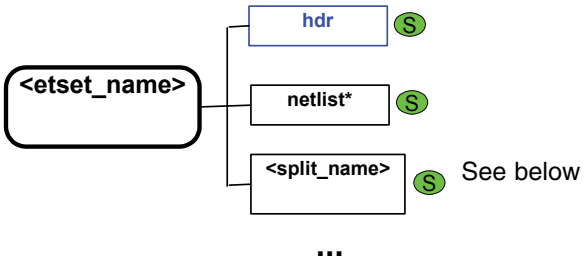


Job>steps>chk (Checklist)

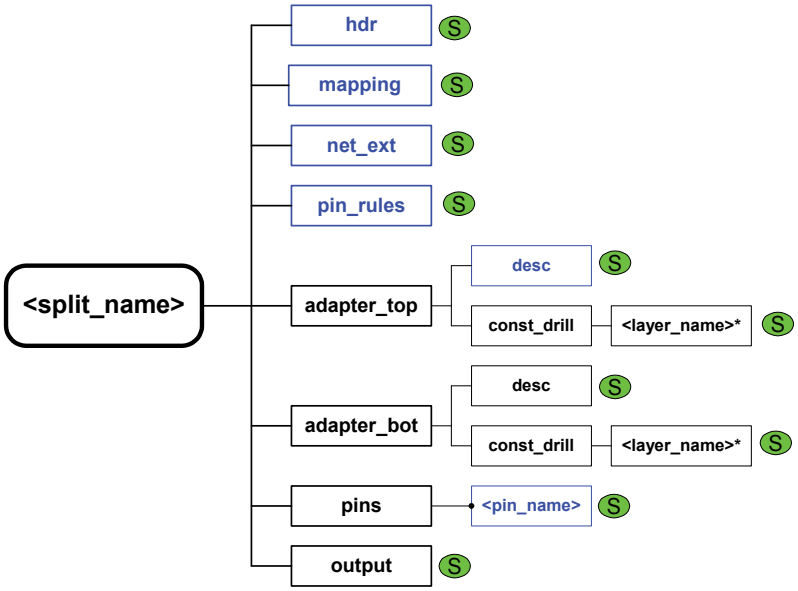


Job>steps>et (Electrical Test)

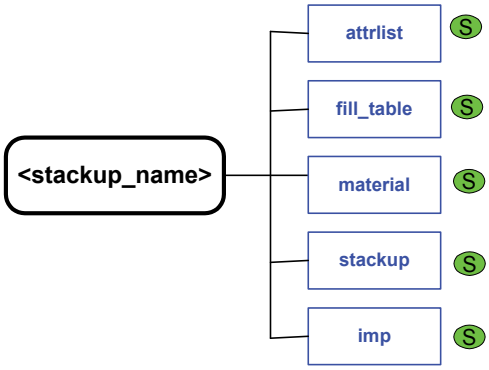
* Netlist is identical in type to the other netlists.



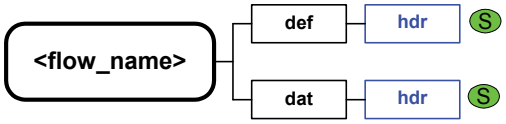
Job>steps>et>split_name



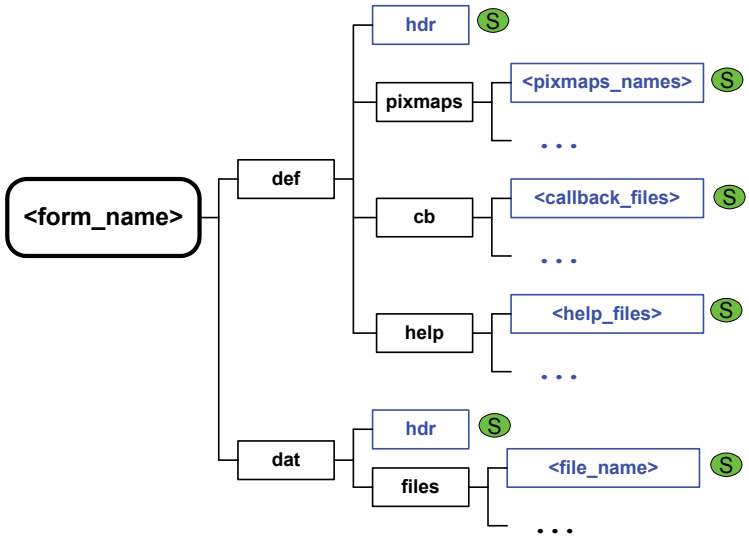
Stackup



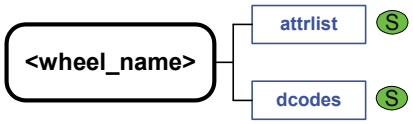
Flows



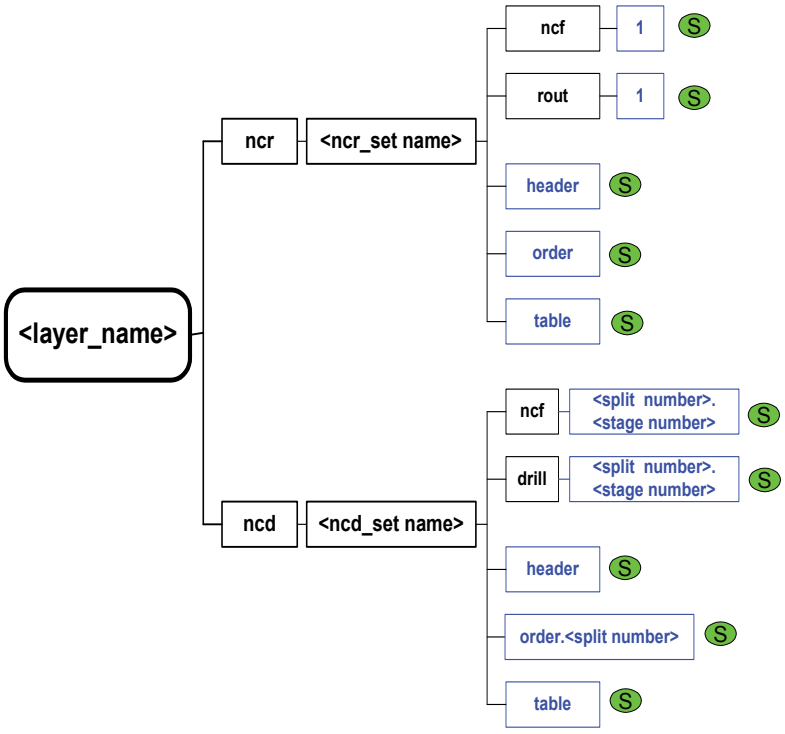
Forms



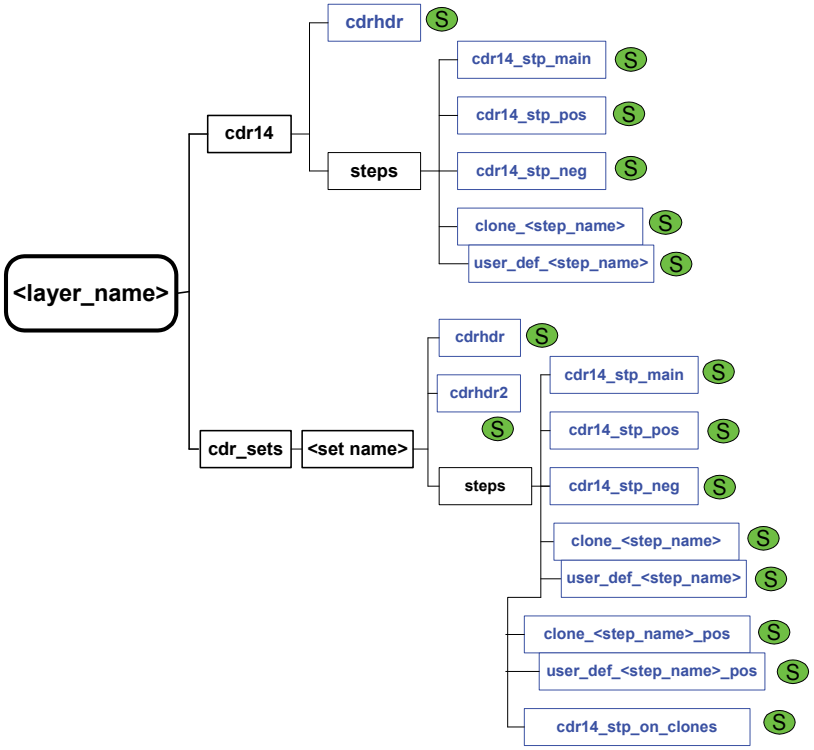
Wheels



ncr - ncd



cdr14 - cdr_sets



Chapter 4 Job Entity Database

This chapter describes in detail each element of the Job Entity database.

steps (See Chapter 5)

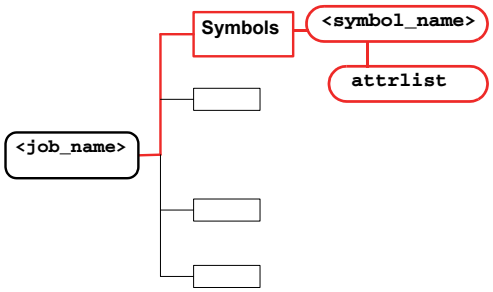
symbols (System and User Symbols)

attrlist (Attribute List)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/symbols/<symbol_name>/attrlist**

This file contains the values for attributes (system and user) of a symbol.

For a list of symbols, see "Symbol Definitions" on page 202.



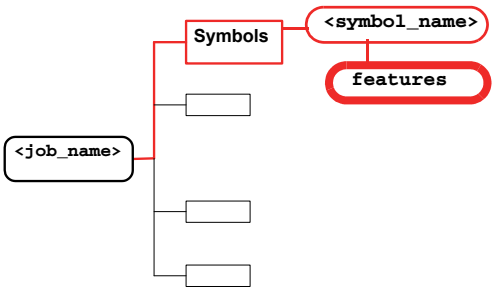
Example

```
.out_break = no
.out_scale = no
.break_away = no
.fill_dx = 0.100000
.fill_dy = 0.100000
.image_dx = -1.000000
.image_dy = -1.000000
connector = no
target = no
component =
comment =
hole_type = plated
serial_number = 15
```

features (Symbol Features)

Type:	Line Record Text
Compression:	Yes
Sum file:	Yes
Path	**<job_name>/symbols/<symbol_name>/features**

The symbol features file describes the graphical shape of the symbol. It is similar in structure to the layer features file.



Example

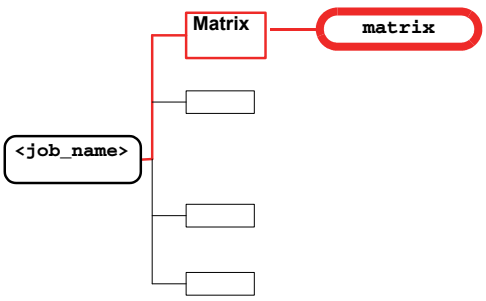
See **<job_name>/steps/<step_name>/layers/<layer_name>/features**

matrix (Job Matrix)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/matrix/matrix**

This file contains all the information which represents the Job Matrix. The Job Matrix is a two-dimensional array, where columns are **steps** - multi-layer entities (such as single images, sub panel arrays, production panels and coupons) and rows are **layers** - sheets on which elements are drawn for plotting, drilling and routing or assembly.

Each job can contain only one matrix file. The library job can contain several matrices.



Example

```
STEP {
   COL=1
   NAME=PCB
}
STEP {
   COL=2
   NAME=PANEL
}
```

```
...
LAYER {
   ROW=1
   CONTEXT=BOARD
   TYPE=COMPONENT
   NAME=COMP_+_TOP
   POLARITY=POSITIVE
   START_NAME=
   END_NAME=
   OLD_NAME=
   ADD_TYPE=
   COLOR=606090
}
LAYER {
   ROW=2
   CONTEXT=BOARD
   TYPE=SILK_SCREEN
   NAME=SST
   OLD_NAME =
   POLARITY=POSITIVE
   START_NAME=
   END_NAME=
   OLD_NAME=
   ADD_TYPE=
   COLOR=606090
}
...
```

The file contains two arrays: **STEP** and **LAYER**

Fields in the STEP array:

COL	The number of the column in the matrix. Columns must be unique positive numbers (1 and above). Gaps are allowed between columns, causing vertical gaps to be created between steps in the displayed matrix.
NAME	The name of the step, according to the legal entity names described earlier. Each named step MUST have a step entity defined under the steps directory of the job, otherwise the job may be unreadable.

Fields in the LAYER array:

ROW	The number of the row in the matrix. Rows must be unique positive numbers (1 and above). Gaps are allowed between rows, causing horizontal gaps to be created between layers in the displayed matrix.
CONTEXT	The layer context must be one of the two values: **BOARD** A layer which participates in the actual board production **MISC** Any other layer which is used for drawings, testing, etc.

TYPE	The layer type must be one of the following values: **SIGNAL** A layer used for regular signal transfer **POWER_GROUND** A plane layer, used for power or ground signals **MIXED** A combination of a signal and a plane layer **SOLDER_MASK** A layer used for solder mask application **SOLDER_PASTE** A layer used for depositing solder paste for assembly **SILK_SCREEN** A layer used for application of text legend **DRILL** A layer used to produce drill programs **ROUT** A layer used to produce rout program **DOCUMENT** A layer used for drawings, testing, auxiliary processes, etc. **COMPONENT** A layer containing components locations and outlines. **MASK** A layer containing additional information used by the Frontline GenFlex product.
NAME	The name of the layer, according to the legal entity names described earlier. Each named layer MUST have a layer entity defined under the layers directory of each step in the job, otherwise the job may be unreadable.
OLD_NAME	The previous name of the layer. When this field has a value it means that a matrix layer has been renamed and this value is its old name. If the field is blank it means the layer has not been renamed.
POLARITY	This parameter describes the polarity of a whole layer. It is applied to the image when output (to a photoplotter for example). The layer polarity must be one of the two values: **POSITIVE** A copper layer in which features represent copper **NEGATIVE** A copper layer in which features represent laminate
START_NAME, END_NAME	These fields are only active for drill and rout layers. They specify the span of the drill or rout, in case it is partial (e.g. blind or buried via layers). Each field must be a valid board layer name. When the fields are empty, **START_NAME** is assumed to be the first board layer (which is not a drill or rout layer) and **END_NAME** is assumed to be the last board layer (which is not a drill or rout layer).
ADD_TYPE  in 7.0	Contains the layer subtype names—for example, **COVERLAY**, **COVERCOAT**, **PUNCH**, **STIFFENER**, **BEND_AREA**, and **PSA**. The **TYPE** field of these types is one of the existing types (**SM**, **ROUT**, etc.) **Note:** In V6.x, only basic support for these fields was available, and data was stored on **DOC** layers. As of V7.0, this field is user-defined, making other subtypes, such as **MICRO_VIA**, **POWER**, **GROUND**, etc. possible, and the user-define hierarchy is maintained.
COLOR	The RGA representation in percent of the color for display of the layer.

The layers should be ordered according to the stackup of the board, such as:

comp_+_top	comp_+_bot	drill_1 ... drill_5
sigt ... sigb	dril	...

New Layer Types Required for GenFlex 6.4

Several layer types were added to GenFlex to meet the unique needs of the Flex industry:

Mask - Base type for all mask types

Coverlay - (Base type: solder_mask) Clearances of a coverlay layer

Covercoat - (Base type: solder_mask) Clearances of a covercoat layer

Punch - (Base type: rout) The pattern to be punched by a die-cut fixture

Stiffener - (Base type: mask) Shapes and locations where stiffener material will be placed on the PCB

Bend Area - (Base type: mask) For labeling areas on the PCB that will be bent when the PCB is in use

PSA (Pressure Sensitive Adhesive) - (Base type: mask) Shapes and locations where PSA material will be placed on the PCB

Area - (Base type: document) Area definition.

Exposed Area - (Base type: document) Define the exposed area of an inner layer, and the solder mask/ coverlay of the exposed area.

Signal Flex - (Base type: signal) Signal layer for flex board.

Power Ground Flex - (Base type: pg) Power ground layer for flex board.

Mixed Flex - (Base type: mixed) Mixed layer for flex board.

Drawing - (Base type: Doc) Drawing layer definition

Plating_mask (Base type: Mask) Defines which features in the adjacent copper layer should be plated

Immersion_mask (Base type: Mask**)** Defines which features are to be covered during immersion gold process

Osp_mask (Base type: Mask) Defines which features are to be covered with osp finish

Silver_mask (Base type: Mask) Defines the silver mask of the adjacent copper layer.

New types are used in a file **<job_name>/matrix** as an additional **values** to the layer parameter **type** (TYPE = Stiffener).

New Fields in Layer Group Required for GenFlex 6.4

Attached layers – List of layers attached to an area layer.

New field is used in the file **<job_name>/matrix**, in the layer group.

Orientation – Defines the orientation of a layers.

Options are: NOT_DEFINED, UPWARDS, DOWNWARDS

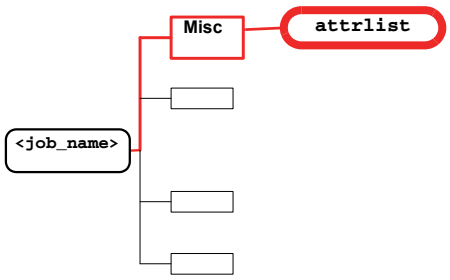
Default value: NOT_DEFINED.

misc (Miscellaneous)

attrlist (Attributes Used in Job)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/misc/attrlist**

This file contains the values for attributes (system and user) of a job. Only attributes (system and user) that have been defined are stored in the job.



Example

```
.customer = abc
connector = no
target = no
```

The file contains lines of the form:

```
<attribute> = <value>
```

System attributes for a job include:

```
.customer
.comment
.primary_side
```

S last_save (Last Time Job Saved)

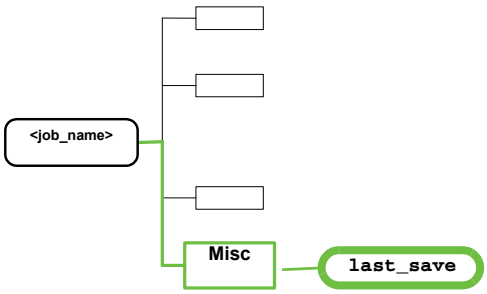
Type:	Line record text
Compression:	None
Sum file:	No
	<job_name>/misc/last_save

This file is written each time a "save" operation is done on a job. It records the time of the save operation.

Example

961224.183210

The file has one line of the format **yyyymmdd.hhmmss**.

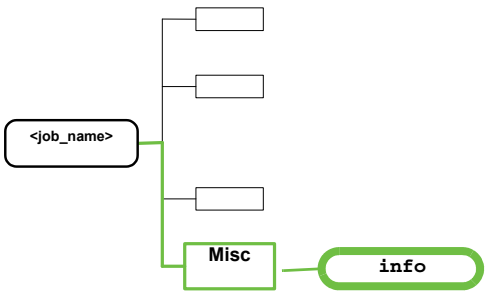


S info (Basic Job Information)

Type:	Line record text
Compression:	None
Sum file:	Yes
	<job_name>/misc/info

This file is written each time a "save" operation is done on a job. It records basic information on the job.

When a job is "opened" within a Valor application, the **MAJOR_VERSION** information is read to ensure that it is compatible with the application. If the **MAJOR_VERSION** number saved in the **info** file is greater than the number of the highest supported **MAJOR VERSION**, the application does not open the job and an error message is displayed.



Example

```
JOB_NAME=k10025_cd2
ODB_VERSION_MAJOR=6
ODB_VERSION_MINOR=2
ODB_SOURCE=Zuken BD
CREATION_DATE=20030727.091213
SAVE_DATE=20030727.091230
SAVE_APP=Trilogy 5000 7.0
SAVE_USER=mikel
```

where:

JOB_NAME is the name of the job.

ODB_VERSION_MAJOR is the major version designation such as '6' in Version 6.2.

ODB_VERSION_MINOR is the minor version designation such as '.2' in Version 6.2.

ODB_SOURCE is the source of data, typically a CAD/EDA system name.

CREATION_DATE and **SAVE_DATE** follow the format **yyyymmdd.hhmmss**.

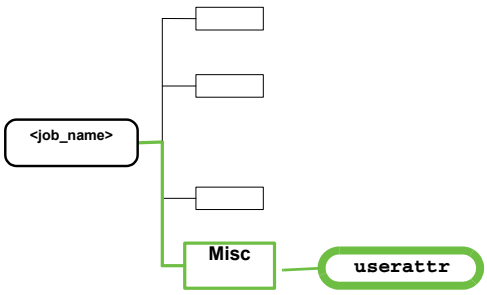
SAVE_APP is the name and number of the application in which the job was saved along with the currently running software version.

SAVE_USER is the login name of the user saving the file.

S *userattr (User Attributes)*

Type:	Structured Text
Compression:	None
Sum file:	Yes
	<job_name>/misc/userattr

This file contains a list of the user attributes which were defined in the library at the time the job was created. It is read each time when the job is opened. All user attributes, for all entities are listed here.



Example

```
BOOLEAN {
   NAME=CONNECTOR
   PROMPT=CONNECTOR :
   ENTITY=ALL
   DEF=NO
}
```

Description

The file contains several arrays. Each array corresponds to one type of attribute:

BOOLEAN
TEXT
OPTION
INTEGER
FLOAT

Fields for a structure of type **BOOLEAN**:

NAME	The name of the attribute
PROMPT	The prompt used on the screen when this attribute is displayed
ENTITY	The entities for which this attribute is applicable. A semi colon separated list of entity types of: job, step, symbol, layer, stackup, wheel, feature, component
DEF	Default value (NO or YES)

Fields for a structure of type **TEXT**:

NAME	The name of the attribute
PROMPT	The prompt used on the screen when this attribute is displayed
MIN_LEN	Minimum length of the text attribute
MAX_LEN	Maximum length of the text attribute
ENTITY	See **ENTITY** for **BOOLEAN**
DEF	Default value

Fields for a structure of type **OPTION**:

NAME	The name of the attribute
PROMPT	The prompt used on the screen when this attribute is displayed
OPTIONS	A semi colon (;) separated list of options
DELETED	A semi colon (;) separated list of the values YES and NO.This corresponds to the list of options, possibly causing an option to be deleted (YES value)
ENTITY	See ENTITY for BOOLEAN
DEF	Default value

Fields for a structure of type **INTEGER**:

NAME	The name of the attribute
PROMPT	The prompt used on the screen when this attribute is displayed
MIN_VAL	Minimum value for the integer attribute
MAX_VAL	Maximum value for the integer attribute
ENTITY	See **ENTITY** for **BOOLEAN**
DEF	Default value

Fields for a structure of type **FLOAT**:

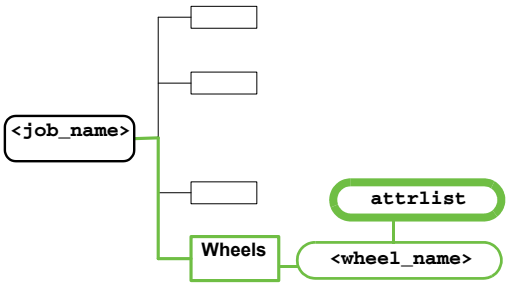
NAME	The name of the attribute
PROMPT	The prompt used on the screen when this attribute is displayed
MIN_VAL	Minimum value for the float attribute
MAX_VAL	Maximum value for the float attribute
ENTITY	See **ENTITY** for **BOOLEAN**
DEF	Default value
UNITS	**NO_UNITS, INCH_MM** or **MIL_MICRONS**. Affects the way the value is displayed (digits after the decimal point).

wheels (Gerber and tool wheels)

attrlist (Attributes Values)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/wheels/<wheel_name>/attrlist`

This file contains the values for attributes (system and user) of a wheel.



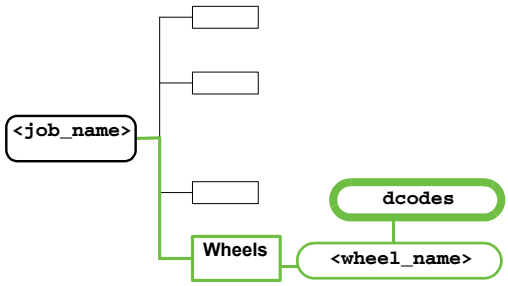
Example

```
comment=<company> wheel
```

dcodes (Wheel Dcodes Definition)

Type:	Line Records Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/wheels/<wheel_name>/dcodes`

This file saves a wheel which is used during Gerber input.



Example

```
dcode10 r12
0 no_mirror
dcode11 r50
0 no_mirror
dcode12 r60
0 no_mirror
dcode13 r10
0 no_mirror
dcode14 r70
dcode15 r80
dcode17 r5
```

```
0 no_mirror
0 no_mirror
0 no_mirror
```

Each line in the file has the format:

```
dcode<n> <symbol_name> <angle> <mirror>
```

Where:

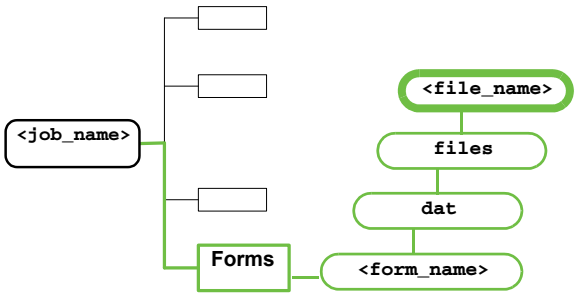
n	Dcode number
sym_name	Symbol name
angle	Always 0 (reserved for future use)
mirror	Always no_mirror (reserved for future use)

forms (Work Forms)

S *dat/files/<file_name> (Image File)*

Type:	XPM or TIFF
Compression:	None
Sum file:	No
Path	**<job_name>/forms/<form_name>/dat/files/<file_name>**

This file corresponds to a form field of the type picture or drawing It contains a graphical image which is stored in this field. The name of the file must be the same as the name of the field.



Example

```
/* XPM */
static char * gns-
genesis186d4.1071 [] =
{
/* width height ncolors cpp [x_hot y_hot] */
"181 172 3 1 0 0",
/* colors */
"  c #EBEBF0F0CFCF",
"! c #FCFC00000000",
"# c #00000000FCFC",
/* pixels */
"                        !!!!!!!!!!!!!!!!!!!!!!!!!",
"                          !!!!!!!!      !!!!!!!!!!!",
"                          !!!!!!!!      !!!!!!!!!!!",
....
"                          !!!!!!!!!!!!!!!!!!!!!!!!!"};
```

Full Description

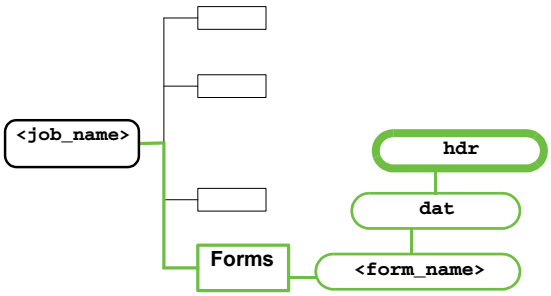
The system currently recognizes 2 standard formats for graphical images:

XPM	X11 pixmap, created by the HP Vueicon program
TIFF	Tagged Image File Format, created by various packages

S dat/hdr (Data Header)

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/forms/<form_name>/dat/hdr**

This file contains the textual contents for various fields in a Work Form. It is updated each time the form contents is changed.



Example

```
STEP=pcb
LAYER_1=L1
1_MIN_P2P=3
1_TYP_P2P=5
1_MIN_P2C=3
1_TYP_P2C=5
1_MIN_C2C=4
1_TYP_C2C=6
```

Each line of the file has the following structure:

<field> = <value>

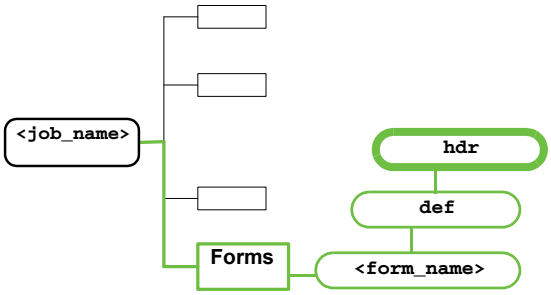
Where:

<field>	The internal (not displayed) name of the form field. This name must exist inside the definition portion of the form.
<value>	The string which represents the contents of the form field.

S def/hdr (Definition Header)

Type:	Structured Text
Compression:	None
Sum file:	No
Path	**<job_name>/forms/<form_name>/def/hdr**

This file contains the definition of Work Form fields, including their types, geometry, action, etc.



Example

```
form {
  VER=0
  LABEL=analysis_results
  UNITS=I
  W=6500
  H=6500
  ACT=
  CLOSE_ACT=
  AUTO_UPDATE=YES
}
textf STEP {
 g {
      X=0
      Y=5736
      W=3501
      H=385
      BW=1
      BG=999980
      READABLE=YES
      EDITABLE=YES
      LTYPE=L
      FONT=tbr18
      LABEL=Step:
      PIXMAP=
      OR=H
 }
 cb {
      HELP=
      ACT=
 }
 te {
      FONT=tbr18
      NUMROWS=1
      NUMCOLS=0
 }
}
```

The file consists of multiple objects, each one representing one field in the form. The first object contains definitions about the form itself. Its structure is:

```
form {
 <field> = <value>
 ....
}
```

<field> can be one of the following types:

VER	Version number (Reserved for future use)
LABEL	Form name, to be displayed in the title bar
UNITS	Should be 0 (Reserved for future use)
W	Width of the form, in mils (0.001")
H	Height of the form, in mils (0.001")
ACT	Name of a call back to be activated each time the form is displayed
CLOSE_ACT	Name of a call back to be activated each time the form is closed.
AUTO_UPDATE	**YES** if the form definition is to be updated from the library each time the form is opened. **NO** if the form definition should not be affected by library changes.

Each following object has the following structure (note that not all substructure appear for each object):

```
<type> <name> {
  g {
     <field> = <value>
     ....
  }
  cb {
     <field> = <value>
     ....
  }
  te {
     <field> = <value>
     ....
  }
  ce {
     <field> = <value>
     ....
  }
  se {
     <field> = <value>
     ....
  }
}
```

<type> can be one of the following values:

sep	A separator object
label	A label object
textf	A text field object
choice	A radio, set or option menu object
picture	A graphical image
drawing	A graphical image
scale	A slider field
button	A push button object

The following table represents the mapping between object types and the substructures which appear in its definition:

	g	cb	te	ce	se
sep	x				
label	x				
textf	x	x	x		
choice	x	x		x	
picture	x	x			
drawing		x	x		
scale	x	x			x
button	x	x			

The fields which are available inside each structure are described below.

The g (geometry) fields:

X	X coordinate of the field lower left (in mils)
Y	Y coordinate of the field lower left (in mils)
W	Width of the field (in mils)
H	Height of the field (in mils)
BW	Border width (in screen pixels)
BG	Background color (A 6 digit number - rrggbb) **rr** - red value between 0 and 99 **gg** - green value between 0 and 99 **bb** - blue value between 0 and 99
READABLE	**YES** - if the field contents are to be displayed| **NO** - if the field contents have to be hidden
EDITABLE	**YES** - if the field can be edited on screen by the operator **NO** - if the field is for display only
LTYPE	**L** - for a textual label **P** - for a pixmap logo
FONT	The font used for the label. A string of the type xyznn where: **x** t(imes), h(elvetica) or c(ourier) **y** b(old) or m(edium) **z** r(egular) or i(talic) **nn** number of points (10,12,14,18 or 24). 1 point = 1/72”.
LABEL	The text to be displayed in the label
JUSTIFY	The justification of the text in th field (**CENTER, LEFT** or **RIGHT**).
PIXMAP	The name of the pixmap used if **LTYPE=P**. The pixmap resides in the def/pixmaps directory of the form.
OR	orientation of the field (relevant for compound field such as text or choice). **H** for horizontal **V** for vertical.

The **cb** (callback) fields:

HELP	The name of the help file for the field (inside the **def/help** directory)
ACT	The name of the activation callback field (inside the **def/cb** directory)

The **te** (text extension) fields:

FONT	Font of the user entered text. See FONT in the **g** (geometry) section above.
NUMROWS	Number of rows in the field (1 for single line, 2 for multi line)
NUMCOLS	Should be 0 (Reserved for future use)
TYPE	Has one of the following values: **TEXT** Free text **INT** Integer values **FLOAT** Floating point (real) values **DATE** Legal date values **TIME** Legal time values
MIN	Minimal value for type INT or FLOAT
MIN_DATE_TIME	Minimal value for type DATE or TIME
MAX	Maximal value for type INT or FLOAT
MAX_DATE_TIME	Maximal value for type DATE or TIME
TEXT_FORMAT	**NONE, UPPER_CASE** or **LOWER_CASE**
DATE_FORMAT	**DD/MM/YY, MM/DD/YY** or **YY/MM/DD**
TIME_FORMAT	**HH:MM:SS**

The **ce** (choice extension) fields:

LTYPE	**L** for a textual options **P** for pixmap (logos) options
MODE	**R** for a radio choice field (one of many) **S** for a set choice field (some or many) **M** for an option menu choice field (one of many)
OR	Orientation of the options in the field. **H** for horizontal **V** for vertical
NUMCOLS	Number of columns for options

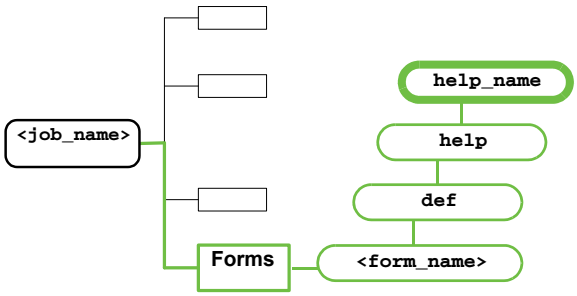
The se (scale extension) fields:

MIN	Minimal value of the slider
MAX	Maximal value of the slider
RADIX	Radix value (Currently must be 0)

S <help_name> (Help Text)

Type:	Free Text
Compression:	None
Sum file:	No
Path	`<job_name>/forms/<form_name>/def/help/<help_name>`

Contains help text to be used by the user who views the form. This file is only a part of the form definition in the library and is not copied into the form copy inside the job.



Example

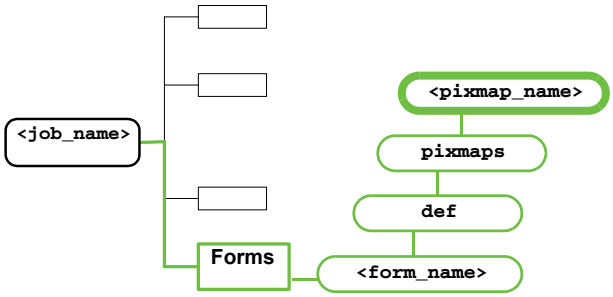
This field must be filled by the letters A,B,C or D

The file has a free text format.

S <pixmap_name> (Pixmap Field File)

Type:	XPM format.
Compression:	None
Sum file:	No
Path	`<job_name>/forms/<form_name>/def/pixmaps/<pixmap_name>`

This file is used for fields which are of LTYPE=P (pixmap). It contains the graphical image to be displayed.



Example:

```
/* XPM */
static char * logo []
= {
/* width height
ncolors cpp [x_hot
y_hot] */
"66 48 6 1 0 0",
/* colors */
"  s iconColor2m whitec white",
". c #000000000000",
"X c #FCFCC6C64D4D",
"o s iconColor5m blackc blue",
"O s iconColor3m blackc red",
"+ s iconColor8m blackc magenta",
/* pixels */
```

```
" .....................
",
" .XXX.XXXXXXXXXXXXX.
",
" .XXX.XXXXXXXXXXXXX.
",
" .XX...XXX...XXX...X.           o
",
" .X.....X.....X......     ooooooo    OOOO O  ++++    ++++
",
...
"
"};
```

XPM - X11 pixmap, created by the HP Vueicon program.

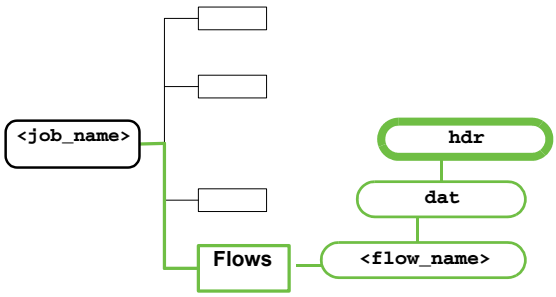
flows (Job Process Charts)

No longer supported. However, this entity may appear in jobs containing legacy data.

S *dat/hdr (Data Header)*

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/flows/<flow_name>/dat/hdr**

This file represents the current state of a Work Flow. It contains a subset of the stages defined in the flow definition section and for each stage it holds information about status, date time, and operator.



Example

```
S0 = START-837427680-
john END-837427684-john
S1 = START-837427689-
john END-837427693-john
S2 = START-837427698-john END-837427703-john
S3 = Y
S9 = START-837427713-james END-837427717-james
S5 =
S6 = START-837427722-james END-837427727-james
S10 = START-837427731-mary END-837427736-mary
S4 = START-837427740-mary
S7 = Minor
S11 = 61
```

```
S12 =
S8 =
```

Each line of the file has the following structure:

```
<stage> = <value>
```

Where:

<stage>	The internal (not displayed) name of the flow stage. This name must exist inside the definition portion of the flow.
<value>	A string which differs according to the type of the stage (see below).

<value> can be:

For a stage of type Condition:

Y for yes
N for no

For a stage of type Switch:

One of the values allowed for the switch, according to the definition part of the flow.

For a stage of type Stage or Subflow:

START-time-operator

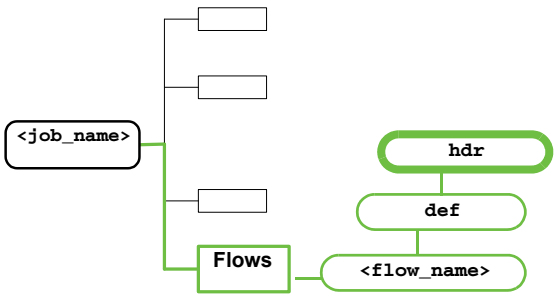
END-time-operator

Note 1) time is UNIX time (seconds starting January 1st, 1970)
2) the END string may not appear for stages in progress
3) both START and END may not appear for stages not started yet

S *def/hdr (Definition Header)*

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/flows/<flow_name>/def/hdr**

This file describes the structure of a Work Flow, including all the stages, the relation between them, and various additional information.



Example

```
flow 0 {
  VER=0
  LABEL=producibility_flow
  AUTO_UPDATE=NO
  MAX_SUBFLOW_LEVEL_ID=1
}
stage 2 {
  LEVEL_ID=1
  NAME=S0
  LABEL=
  TEXT=EDA Input
  FORM=
  GATE=OR
  HELP=
  PRE0=1
  STAGE_ACT=
  OPEN_ACT=
  CLOSE_ACT=
  NEW_LEVEL_ID=0
}
...
switch 11 {
  LEVEL_ID=1
  NAME=S7
  LABEL=
  TEXT=Determine Action
  FORM=
  GATE=OR
  HELP=
  PRE0=8
  PRE1=9
  PRE2=10
  PRE3=
  STAGE_ACT=
  OPEN_ACT=
  CLOSE_ACT=
  NEW_LEVEL_ID=0
}
```

The file consists of the following structures:

```
<type> <serial> {
  fields;
  ...
}
```

Where:

<type> is one of:

flow	Appears once as the first entity of the flow definition
stage	Appears as many times as needed to represent a stage in the process.
cond	Appears as many times as needed to represent a condition.
switch	Appears as many times as needed to represent a switch between multiple stages in the process
subflow	Appears as many times as needed to represent a composite flow which makes this stage.

<serial> is a unique number identifying the stage for the purpose of referencing from other stages.

The fields of a structure of type **flow**:

VER	Version number (Reserved for future use)
LABEL	Flow name to be displayed in the title row
AUTO_UPDATE	Reserved for future use. Should be set to NO
MAX_SUBFLOW_LEVEL_ID	Used internally. Should be set to 1.

The fields of all other structures:

LEVEL_ID	0 for the first structure (which represents the flow) 1 for all other stages.
NAME	Internal name of the stage. Used for references from the data section of the Work Flow.
LABEL	Only for subflow stages. The name of the lower level flow which represents this stage.
TEXT	The actual text which is displayed on the screen for this stage
FORM	An optional Work Form which is attached to the stage
GATE	One of the following values: **OR** - A stage can be started when at least one of its parents was finished. **AND** - A stage can be started when all its parents were finished.
HELP	A name of a file which provides information about the stage and can be displayed when the Work Flow is viewed. The file is a part of the Work Flow definition in the **flows/ <flow_name>/def** help code directory in the library.
PRE<n>	Up to 10 (n = 0 to 9) parents of this stage. The value for this field must be a valid serial number of the parent. If the parent is a condition or a switch, the serial number will be followed by the value for which this is the child.

Example

```
switch 11 {
 ...
```

```
}

stage 12 {
 ...
   PRE0 = 11 OK
   ...
}
```

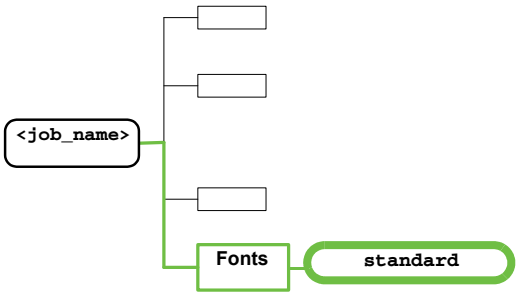
STAGE_ACT	The name of a callback to be executed when a stage changes its status. This field is only used in the main subflow of the flow, which is always the second structure. Callbacks are executed from the **def/cb** directory in the corresponding flow in the library and are not residing inside the job itself.
OPEN_ACT	The name of the callback to be executed when the flow is displayed. Same rules as **STAGE_ACT**.
CLOSE_ACT	The name of the callback to be executed when the flow is closed. Same rules as **STAGE_ACT**.
NEW_LEVEL_ID	For internal use. Must be 1 for the first stage structure and 0 for the rest.

fonts (Fonts used in Job)

standard (Standard Font)

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/fonts/standard**

This file describes the vector representation of all the characters which can be a part of a text feature inside a layer. The system currently supports one font, named **standard**.



Example

```
XSIZE 0.302000
YSIZE 0.302000
OFFSET 0.000000
CHAR!
LINE 0.000000 0.000000 0.000000 0.200000 P R 0.012000
LINE 0.000000 -0.100000 0.000000 -0.100000 P R 0.012000
ECHAR
...
```

```
CHAR ~
LINE -0.100000 0.150000 -0.050000 0.200000 P R 0.012000
LINE -0.050000 0.200000 0.050000 0.100000 P R 0.012000
LINE 0.050000 0.100000 0.100000 0.150000 P R 0.012000
ECHAR
```

The file consists of a header containing global parameters followed by a collection of character blocks.

The header block consists of 3 lines:

XSIZE <size>	Horizontal size of a character, in inches
YSIZE <size>	Vertical size of a character, in inches
OFFSET <size>	Horizontal Distance between the end of one character block and the beginning of the next one.

The character block consists of the following lines:

CHAR <char>	Defines the ASCII character which is defined by this block
LINE <xs> <ys> <xe> <ye> <pol> <shape> <width>	A definition of a line between (xs,ys) and (xe,ye). All coordinates are in inches. **<pol>** is the polarity of the line (P for positive, N for negative) **<shape>** is the shape of the ends of the line (R for rounded, S for square) **<width>** is the line width in inches
ECHAR	Ends the definition of a character

Note The origin of each character is at the lower left corner. For best results, font definition should include all ASCII characters.

SHX Fonts Directory (AutoCAD vector) (GenFlex 6.4)

To implement AutoCAD (.SHX) vector font a new directory **<job_name>/fonts_ex/shx/** is used to keep font files (filename.shx). (Downgrade will delete the directory)

stackups

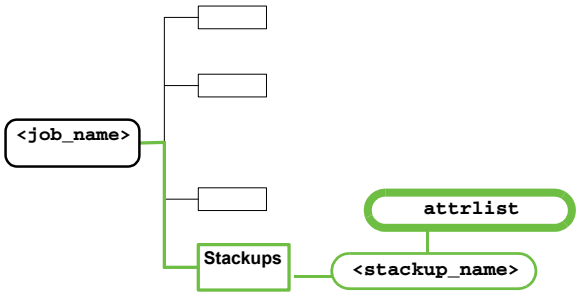
No longer supported. However this entity may appear in jobs containing legacy data.

S *attrlist (Attribute List)*

Type:	Structured Text
Compression:	None

Sum file:	Yes
Path	`<job_name>/stackups/<stackup_name>/attrlist`

This file contains the values for attributes (system and user) of a stackup.



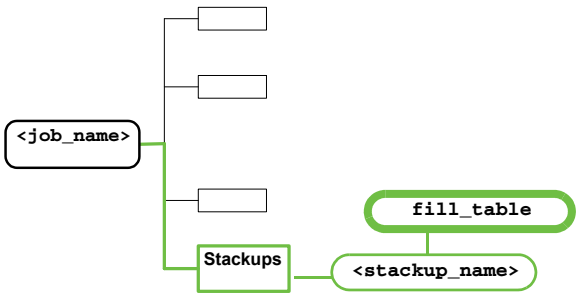
Example

```
comment = Final Stackup
```

S *fill_table (Prepeg Combinations)*

Type:	Line Record Text
Compression:	Yes
Sum file:	No
Path	`<job_name>/stackups/<stackup_name>/fill_table`

Contains the **`fill_table`** active at the time that the stackup was saved. The **`fill_table`** is the list of prepreg combinations which may be placed between foil layers.



Example

```
# A - ounce range for
first facing copper
# B - Copper area for
first facing copper (percentage)
# C - Position of first layer (Inner, Outer, Any)
# D - ounce range for second facing copper
# E - Copper area for second facing copper (percentage)
# F - Position of second layer (Inner, Outer, Any)
# G - Thickness of fill
# H - Positive tolerance on width of fill
# I - Negative tolerance on width of fill
# J - ordered list of names of sheets in fill
#
#      A      B     C       D       E     F       G  H   I
J
begin_fill
fill ((0-10oz 0-100 any)   (0-10oz  0-100 any)   (2  0.5 0.5)
(1080))
fill ((1-1oz  0-100 out)   (0.5-1oz 0-100 inner) (7  1   1)
(113 116))
```

```
fill ((1-1oz  0-100 inner) (1-1oz   0-100 inner) (7  1   1)
(113 113))
fill ((1-1oz  0-100 inner) (1-1oz   0-100 inner) (14 1   1)
(7628 7628))
fill ((1-1oz  0-100 out)   (1-1oz   0-100 inner) (8  1   1)
(116 116))
fill ((1-2oz  0-100 any)   (1-2oz   0-100 any)   (21 2   2)
(7628 7628 7628))
fill ((3-4oz  0-100 any)   (3-4oz   0-100 any)   (20 2   2)
(7628 7628 7628))
fill ((1-2oz  0-100 any)   (1-2oz   0-100 any)   (17 2   2)
(7629 7629 7629))
fill ((1-2oz  0-100 any)   (1-2oz   0-100 any)   (10 2   2)
(113 7628 113))
fill ((0.5-1oz 0-100 out)  (1-2oz   0-100 inner) (10 2   2)
(106 7628 106))
end_fill
```

The first record describes a 2 mil thick fill:

```
fill ((0-10oz 0-100 any)   (0-10oz  0-100 any)   (2 0.5 0.5)
(1080))
```

It consists of single sheet of 1080.

It provides a 2 mil spacer between two copper foils with a positive and negative tolerance of 0.5 mils.

Constructed Fill record

The second record describes a 7 mil thick fill:

```
fill ((1-1oz  0-100 out)   (0.5-1oz 0-100 inner) (7  1   1)
(113 116))
```

- It consists of a sheet of 113 over a sheet of 116.
- It provides a 7 mil spacer between two copper foils with a positive and negative tolerance of 1 mil.
- It may only be used between 1oz copper foil outer layer and 0.5-1oz foil inner layer. It may not be used between other types of foils, or foils located in different layers than those specified (unless the same prepreg combination appears in another record).

The **fill** records are delimited by **begin_fill** at the beginning and **end_fill** at the end.

The fill records are of the form:

```
fill ((A B C)   (D E F)   (G H I) (J))
```

Where the letters contain the following fields:

A	Ounce range for first facing copper
B	Copper area for first facing copper (percentage)
C	Position of first layer (Inner, Outer, Any)
D	Ounce range for second facing copper
E	Copper area for second facing copper (percentage)
F	Position of second layer (Inner, Outer, Any)

G	Thickness of fill (in mils)
H	Positive tolerance on width of fill (in mils)
I	Negative tolerance on width of fill (in mils)
J	Ordered list of names of sheets in fill

General rules

The prepregs which appear in column J must also appear in the 'material' file. There must also be a material of this kind in the bill file of the construct. The foils described in columns A, B, C may be placed either over or under the prepreg sheet combination. The ounce range and the copper area range should be complete as possible.

Actual copper area calculations are not made on the panels (this allows stackups to be designed prior to panelization).

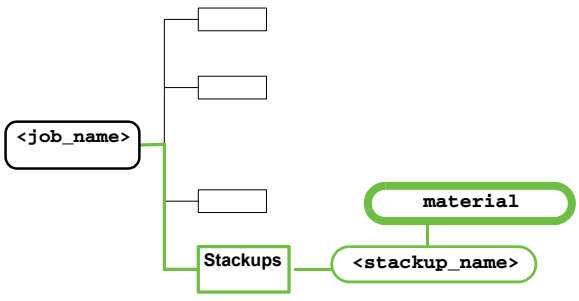
Instead, the following copper area percentage values are automatically assigned to foils, according to the layer type:

Layer type	Assigned copper area
signal	20%
mixed	50%
power and ground	80%

S *material (Material Specifications)*

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/stackups/<stackup_name>/material**

Contains the contents of the generic material file contents at the time that the stackup was saved.



Example

```
pile_begin
#
# Description of foil records
#
# A - Name of foil
# B - Thickness (in ounces) & +/- tolerances
# C - Color in Genesis display
# D - Ductility (HTE or STD)
# E - Resistance (in ohms)
```

```
#
#                    A     B         C      D    E
elem (Simple (Foil (1oz   (1   0 0) 856700 STD   0)))
elem (Simple (Foil (2oz   (2   0 0) 856700 STD   0)))
elem (Simple (Foil (1/2oz (0.5 0 0) 856700 STD   0)))
#
# Description of prepreg & laminate records
#
# A - Prepreg or Laminate
# B - Name
# C - Thickness (in mils) & +/- tolerances
# D - Color in Genesis display
# E - Dry permittivity
# F - Resin percentage (by weight)
# G - Dry weight
# H - Conductivity
#
#             A         B       C             D      E   F  G  H
elem (Simple (Prepreg  (7628   (7   0.5 0.8)   9900 4.2 20 10 0)))
elem (Simple (Prepreg  (108    (1.8 0.2 0.2)   9900 4.2 20 8  0)))
elem (Simple (Prepreg  (106    (1.6 0.2 0.2)   9900 4.3 20 6  0)))
elem (Simple (Laminate (Lam/40 (40  0   0  ) 505050 4.6 0  30 0)))
elem (Simple (Laminate (Lam/32 (32  0   0  ) 505050 4.6 0  26 0)))
#
# Description of core records
#
# A - Name
# B - Total Thickness (in mils) & +/- tolerances
# C - Type of top layer
# D - Name of top layer
# E - Type of middle layer
# F - Name of middle layer
# G - Type of bottom layer
# H - Name of bottom layer
# Note: the (G H) Expression is omitted for single sided clad cores.
#                     A           B          C    D     E   F        G    H
elem (Compound (Core fr4/40_1/1 (42.8 0 0) (Foil 1oz) (Lam Lam/40) (Foil 1oz))
elem (Compound (Core fr4/32_1/1 (34.8 0 0) (Foil 1oz) (Lam Lam/32) (Foil 1oz))
pile_end
```

Foil Record

The first record describes a conductive foil:

```
elem (Simple (Foil (1oz   (1   0 0) 856700 STD)))
```

- Its name is **1oz**
- Its weight is one ounce with positive and negative tolerances of zero
- It will be displayed in a color whose Genesis number is **856700**
- It has standard ductility (valid options: **STD, HTE**)

Prepreg Record

The fourth record describes a prepreg:

```
elem (Simple (Prepreg  (7628   (7   0.5 0.8)   9900 4.2 20 10
0)))
```

- Its name is **7628**

- It is 7 mils thick, with a positive tolerance of 0.5 mils and a negative tolerance of 0.8 mils
- It will be displayed in a color whose Genesis number is **009900**
- It has a dry permittivity value of **4.2**
- It has 20% resin content (by weight)
- It has a dry weight value of **10**
- It has a conductivity value of **0** (zero)

Laminate Record

The seventh record describes a laminate.

```
elem (Simple (Laminate (Lam/40 (40  0   0  ) 505050 4.6 0  30
0)))
```

- Its name is **Lam/40**
- It is 40 mils thick with positive and negative tolerances of zero mils
- It will be displayed in a color whose Genesis number is **505050**
- It has a dry permittivity value of **4.6**
- It has a 0% resin content (by weight)
- It has a dry weight value of **30**
- It has a conductivity value of **0** (zero)

Core Record

The ninth record describes a copper clad laminate core.

```
elem (Compound (Core fr4/40_1/1 (42.8 0 0) (Foil 1oz) (Lam
Lam/40) (Foil 1oz))
```

- Its name is **fr4/40_1/1**
- It is 42.8 mils thick with positive and negative tolerances of zero mils.
- It is constructed of the following layers:
  - A layer of a foil material as described in the first record
  - A layer of a laminate as described in the third record
  - A layer of a foil material as described in the first record.

Note Minimum / maximum core thickness defines the min/max thickness of the dielectric material in the core.

RCC Record

Below is a sample record describing the RCC material:

```
elem (Compound (RCC 1825 (1.4 0 0) (Foil (0.5 (0.5 0 0) 856700
STD 0)) (Resin (RES02 (0.7 0.1 0.1) 99 3.43 100 5 0)) (NULL)))
```

The name of the record is 1825. It is 1.4 mils thick with positive and negative tolerances of zero mils. It is constructed of the two following layers:

- A layer of a foil material
- A layer of a resin material

File Structure

There are three kinds of **elem** records in the file:

1. For foil records
2. For laminate and prepreg records
3. For compound records (Core, RCC).

Foil records are of the following structure:

```
elem (Simple (A (B   (C) D E)))
```

Where the letters refer to the following fields:

A	Name of foil
B	Thickness (in ounces) & +/- tolerances
C	Color in Genesis display
D	Ductility (HTE or STD)
E	Resistance (in ohms)

Laminate and prepreg records are of the following form:

```
elem (Simple (A  (B (C) D E F G H)))
```

A	Prepreg or Laminate
B	Name
C	Thickness (in mils) & +/- tolerances
D	Color in Genesis display
E	Dry permittivity
F	Resin percentage (by weight)
G	Dry weight
H	Conductivity

Compound records are of the following form:

```
elem (Compound (Core A (B) (C D) (E F) (G H))
```

or

```
elem (Compound (RCC A (B) (C D) (E F)())
```

A	Name
B	Total Thickness (in mils) & +/ tolerances
C	Type of top layer
D	Name of top layer
E	Type of middle layer
F	Name of middle layer
G	Type of bottom layer
H	Name of bottom layer

Note The **(G H)** expression is omitted for single sided clad cores and RCC.

General rules:

- The first line of the file and the last line of the file open and close a block.

- Each of the middle lines is a record describing a type of material.
- Any material that appears in a core must be defined as a record by itself as well.
- Material names are limited in length to 16 characters.

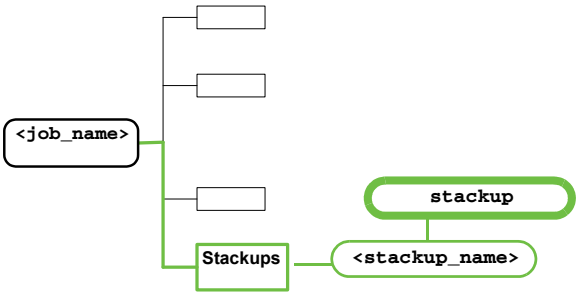
RCC Record

An RCC material is defined as a compound material, similar to Core. It consists of two layers - Foil and Resin.

S *stackup (Build)*

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/stackups/<stackup_name>/stackup**

Describes the build of a stackup.



Example

```
target (112 12 12)
# width height ffu plate_thick mask_thick thick_type vendor layer_match
stk_info 18 24 0 5 2 Laminate Any Yes (drill drill1 drill2 drill3 drill4) (drill drill1)
# The previous line indicates that:
# 18 24: the stackup is made of sheets which are 18 x 24.
# 5: the plating thickness of external layers is 5 mils
# 2: the solder mask thickness is 2 mils
# Laminate: the method for measuring thickness is laminate to laminate
# Any: materials from Any vendor may be used in the stackup
# Yes: one to one correspondence between foil board layers and stackup foil layers.
# (drill drill1 drill2 drill3 drill4): drill layers for sequential lamination.
# (drill drill1): microvia drill layers.
#     min sheets  max sheets
valid 2           3
# The previous line dictates that the minimum number of prepreg sheets used
# to separate layers is two and the maximum is three.
begin_pile
#         thickness       is_mirror construct cost   resin_Er copper_loss
pile_info (105.7 9.9 9.9) Yes       FR-4      11.983 0         0
begin_materials
# A - width
# B - height
# C - cost
```

```
# D - Reserved for future use
# E - material (Foil, Prepreg, Laminate, Core)
# F - weave (Vertical, Horizontal Null)
# G - thickness & +/- tolerances
# H - vendor
# I - generic name (from 'material' file)
# J - catalog number (Less than 16 characters preferred)
# K - construct
# L - Whether material is upsidedown in stackup
#
#    A  B  C     D E        F     G              H I       J      K    L
##############################################################
mat (18 24 0.091 1 Foil    None (1 0 0)          A 1oz     C90126 FR-4 No)
mat (18 24 1.58  1 Prepreg None (1.8 0.26 0.26) B 106     Z1261  FR-4 No)
mat (18 24 1.58  1 Prepreg None (1.8 0.26 0.26) B 106     Z1261  FR-4 No)
mat (18 24 0.139 1 Core    None (8.2 1.5 1.5)   B 8_1/1   H90120 FR-4 No)
mat (18 24 1.2   1 Prepreg None (9.8 0.4 0.4)   B 7628_10 Z0044  FR-4 No)
mat (18 24 0.139 1 Core    None (8.2 1.5 1.5)   B 8_1/1   H90120 FR-4 No)
mat (18 24 1.2   1 Prepreg None (9.8 0.4 0.4)   B 7628_10 Z0044  FR-4 No)
mat (18 24 0.125 1 Core    None (14.1 1.5 1.5)  B 14_1/1  H90119 FR-4 No)
mat (18 24 1.2   1 Prepreg None (9.8 0.4 0.4)   B 7628_10 Z0044  FR-4 Yes)
mat (18 24 0.139 1 Core    None (8.2 1.5 1.5)   B 8_1/1   H90120 FR-4 Yes)
mat (18 24 1.2   1 Prepreg None (9.8 0.4 0.4)   B 7628_10 Z0044  FR-4 Yes)
mat (18 24 0.139 1 Core    None (8.2 1.5 1.5)   B 8_1/1   H90120 FR-4 Yes)
mat (18 24 1.58  1 Prepreg None (1.8 0.26 0.26) B 106     Z1261  FR-4 Yes)
mat (18 24 1.58  1 Prepreg None (1.8 0.26 0.26) B 106     Z1261  FR-4 Yes)
mat (18 24 0.091 1 Foil    None (1 0 0)          A 1oz     C90126 FR-4 Yes)
end_materials
end_pile
sub_lam ((drill Auto Foil) (drill1 Auto Foil) (drill2 Foil Foil) (drill3 Sheet
Foil) (drill4 Mixed Sheet))
```

Explanation

A **sub_lam** record describes the lamination type of each layer from the sequential lamination list.

A layer record of the form (**layer lamination top_lamination**):

Layer - drill or rout layer from the sequential lamination list

Lamination - lamination type of the subassembly. Options are: Auto, Foil, Sheet, Mixed.

Top_lamination - when the lamination is Mixed, top_lamination defines the lamination type of the top layer. Options are: **Foil**, **Sheet**

The **stackup** file contains the following records:

A target record of the form **target <target thickness> <positive tolerance> <negative tolerance>** where all units are in mils.

A **stk_info** record of the form:

stk_info <width> <height> <unused value> <plate_thick> <mask_thick> <thick_type> <vendor> <layer_match>

Where the width and height values are the width and eight of the sheets the stackup is made of.

- The unused value is a numeric value reserved for future use.

- The **thick_type** indicates the method used for measuring stackup thickness.

  vendor is a vendor name if materials in the stackup are from a particular vendor, or **any** if materials may come from any vendor.

  lyr match indicates whether there is a one to one correspondence between foil board layers and stackup foil layers.

Sequential lamination layers are drill and rout layers that generate sequential lamination requirements.

A **pile** record which is delimited by lines of **begin_pile** and **end_pile**.

Containing the subrecords:

pile_info which is a record of the form:

```
pile_info (<thickness>) <is_mirror> <construct> <cost> <resin
Er> <copper loss>
```

Where:

thickness	is the calculated thickness of the stackup with tolerances
is_mirror	indicates whether the stackup is 100% symmetric
construct	indicates the make of the construct in the stackup
cost	the sum of the cost of all materials in the stackup
resin Er	the relative permittivity of the resin used in the resin system of the construct
copper_loss	the thickness of copper lost in internal layers due to processing. This value may also be positive, indicating that internal layers have been plated.

A **materials** sub-record delimited by **begin_materials** and **end_materials** containing **mat** records of the form:

```
mat (A B C D E F (G) H I J K L)
```

Where the letters contain the following fields:

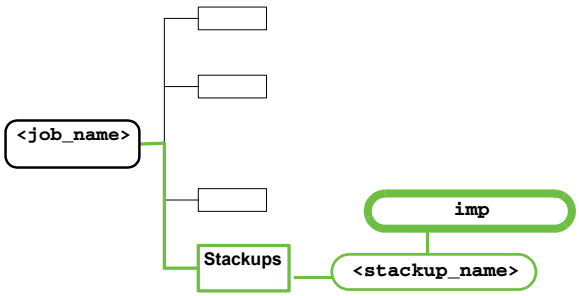
A	width
B	height
C	cost
D	Reserved for future use
E	material (Foil, Prepreg, Laminate, Core, RCC)
F	weave (Vertical, Horizontal Null)
G	thickness & +/- tolerances
H	vendor
I	generic name (from 'material' file)
J	catalog number (Less that 16 characters preferred)
K	construct
L	Whether material is upside-down in stackup
M	Which foils of a core are completely etched off (None, Top, Bottom, Both)

Note Material type (**prepeg, laminate, foil, core, RCC**) may be abbreviated to three letters (**pre, lam, foi, cor, RCC**).

S *imp (Impedance)*

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/stackups/<stackup_name>/imp**

Contains the impedance requirements and results of the stackup.



Example

```
# A - Allowed change to
line width
# B - Units of allowed
change (``Inch''
indicated English. --
i.e. mils)
# C - Width variation
of lines
# D - Units of width variation of lines (``Inch'' indicated
English. -- i.e. mils)
# E - Etch factor (Trapezoidal factor of lines due to etch
process) in mils
# F - Relative Permittivity of soldermask
# G - Impedance frequency (in MHZ)
#         A B      C D      E F G H I
imp_info 1 Inch   1 Inch   1 1 1 3 100
imp_begin
# A - Impedance model
# B - Reference layer
# C - Impedance layer
# D - Second Impedance layer (For broadside differential
models)
# E - Second Reference layer (For Microstrip models)
# F - Original line width (in mils)
# G - Current Line width  (in mils)
# H - Calculated impedance with tolerances (in ohms)
# I - Desired impedance with tolerances    (in ohms)
# J - Original spacing     (in mils) - for differential models
# K - Current spacing      (in mils)
# L - Tolerance of current width (in mils)
#    A                B     C    D  E     F G  H
I       J KL
imp (Surface_Microstrip (12)  (11)  () ()   6 6 (98.3 4.9 4.9)
(90 0 0) 0 02)
imp (Dual_Stripline     (13)  (14)  () (16)  6 6 (138.8 13.8
```

```
13.8) (90 0 0) 0 01)
imp (Dual_Stripline     (13)  (15)  () (16)  6 6 (138.8 13.8
13.8) (0 0 0)  0 01)
imp (Dual_Stripline     (17)  (18)  () (110) 6 6 (138.8 13.8
13.8) (0 0 0)  0 01)
imp (Dual_Stripline     (17)  (19)  () (110) 6 6 (138.8 13.8
13.8) (0 0 0)  0 01)
imp (Surface_Microstrip (111) (112) () ()   6 6 (98.3 4.9 4.9)
(0 0 0)  0 01)
imp_end
```

The impedance records are of the following form:

```
imp (A (B) (C) (D) (E) F G (H) (I) J K L)
```

Where the letters contain the following fields:

A	Impedance model
B	Reference layer
C	Impedance layer
D	Second Impedance layer (For broadside differential models)
E	Second Reference layer (For Microstrip models)
F	Original line width (in mils)
G	Current Line width (in mils)
H	Calculated impedance with tolerances (in ohms)
I	Desired impedance with tolerances (in ohms)
J	Original spacing (in mils) for differential models
K	Current spacing (in mils)
L	Tolerance of current width (in mils)

input

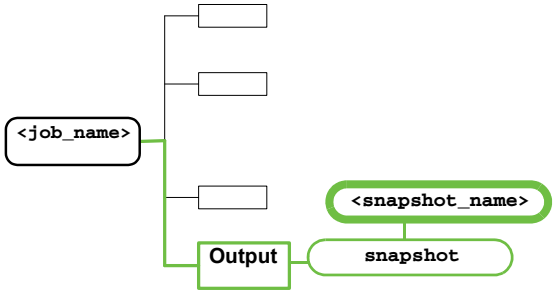
Location of input files when saved to a job.

output

snapshot (Measurement Information))

Path	**<job_name>/output/snapshots/<snapshot_name>**

A snapshot is a screen capture function available in the Graphic Station for the purpose of recording images, notes and measurement details of category violations in analysis.



Each snapshot, consisting of four files each (see list below), is stored in a directory under its own name **<snapshot_name>**. The snapshot name also becomes the name of the four files, each with the appropriate extension. Each snapshot directory consists of the following files:

<snapshot_name>.gif *(image in GIF format)*

<snapshot_name>.nte *(user notes typed in the Snapshot popup)*

<snapshot_name>.txt *(measurement information)*

<snapshot_name>.xpm.gz *(gzipped image in XWindows color bitmap format)*

S *user*

Location to store user files.

S *extension*

Chapter 5 Job>steps Entity

Required for GenFlex 6.4

Layer Profiles File

Profiles data is saved under the step directory in a new file called **lyr_profiles_p**.
Layers profiles file is a standard feature file, which includes surface features only (multi islands and multi holes). Each surface describes one layer profile. Each feature has an attribute **lyr_prf_ref** with reference value = 1,2,3, etc. which is the reference for layers to use a surface as the layer profile.
(Downgrade will delete the file)

Layer Profiles File (Encrypted File)

<job_name>/steps/<step_name>/lyr_profiles_p

Footprint Description File (GenFlex 6.4)

Footprint – an extension data that can replace the profile. Define per step. File is saved under the step directory in a new file called **footprint_p**. The file structure is the same as it is in **profile** file. (Downgrade will delete the file) (See "profile (Outline Shape of Step)" on page 78.)

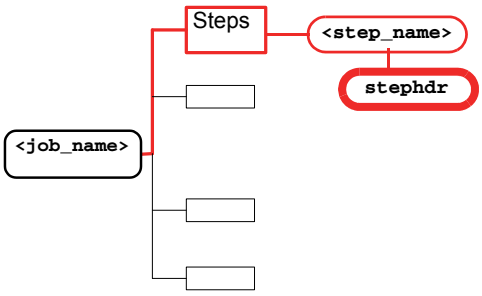
Footprint Description File

<job_name>/steps/<step_name>/footprint_p

stephdr (Step Header)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/stephdr`

This file contains data which is common to the whole step. This includes the step & repeat array for nest steps.



Example

```
X_DATUM=0
Y_DATUM=0

STEP-REPEAT {
   NAME=1UP
   X=1.5
   Y=1.6
   DX=1.2
   DY=1.2
   NX=6
   NY=6
   ANGLE=0
   FLIP=NO
   MIRROR=NO
}

TOP_ACTIVE=1
BOTTOM_ACTIVE=1
RIGHT_ACTIVE=1
LEFT_ACTIVE=1
ONLINE_DRC_NAME=
ONLINE_DRC_MODE=DISABLED
ONLINE_DRC_STAT=RED
ONLINE_DRC_TIME=0
ONLINE_DRC_BEEP_VOL=2
ONLINE_DRC_BEEP_TONE=500
ONLINE_NET_MODE=DISABLED
ONLINE_NET_STAT=RED
ONLINE_NET_TIME=0
ONLINE_NET_BEEP_VOL=2
ONLINE_NET_BEEP_TONE=1000
AFFECTING_BOM=
AFFECTING_BOM_CHANGED=0
```

The file consists of several fields and an array of **STEP-REPEAT** records.The fields are:

X_DATUM	x datum point (used for step & repeat)
Y_DATUM	y datum point (used for step & repeat)
X_ORIGIN	x origin point
Y_ORIGIN	y origin point
TOP_ACTIVE	active area for step & repeat (positive distance from the top edge)
BOTTOM_ACTIVE	active area for step & repeat (positive distance from the bottom edge)
RIGHT_ACTIVE	active area for step & repeat (positive distance from the right edge)
LEFT_ACTIVE	active area for step & repeat (positive distance from the left edge)
ONLINE_DRC_NAME	The name of the checklist (if any) used for on-line DRC
ONLINE_DRC_MODE	One of D**ISABLED, DEFERRED** or **IMMEDIATE**
ONLINE_DRC_STAT	One of **RED, YELLOW** or **GREEN**
ONLINE_DRC_TIME	The last time **check all** was done for on-line DRC
ONLINE_DRC_BEEP_VOL	Beep volume for immediate on-line DRC (0 to 3)
ONLINE_DRC_BEEP_TONE	Beep tone for immediate on-line DRC (200 to 1500)
ONLINE_NET_MODE	One of **DISABLED, DEFERRED** or **IMMEDIATE**
ONLINE_NET_STAT	One of **RED, YELLOW** or **GREEN**
ONLINE_NET_TIME	The last time **check all** was done for on-line netlist
ONLINE_NET_BEEP_VOL	Beep volume for immediate on-line netlist (0 to 3)
ONLINE_NET_BEEP_TONE	Beep tone for immediate on-line netlist (200 to 1500)
AFFECTING_BOM	Name of BOM last used in BOM_MERGE.
AFFECTING_BOM_CHANGED	Indicates whether **AFFECTING_BOM** was changed since last BOM_MERGE. (This requires that BOM_MERGE must be redone before retrieving information relating to BOM, such as Edit>Component>Set Chosen AVL.) If you attempt to retrieve info without performing BOM_MERGE, you will be required to confirm the action.

For the **STEP-REPEAT** array, the fields are:

NAME	Name of the step to be included in the current one (must be a valid step in the same job, without nesting)
X	Start X coordinate for placement of datum point of nested step
Y	Start Y coordinate for placement of datum point of nested step
DX	Horizontal distance between datum points (when angle = 0)
DY	Vertical distance between datum points (when angle = 0)

NX	Number of repetitions horizontally
NY	Number of repetitions vertically
ANGLE	Rotation angle of the steps (0-360 degrees)*
FLIP	Allows two identical steps to be placed on a panel in such a way that on the same side it contains the top of one step and the bottom of the other.
MIRROR	**YES** for mirror (around X axis), **NO** for no mirror

New in 7.0

* Any angle rotation is expressed in the Info Command Interface for **Data type: REPEAT** and **SR** for the Step entity and in the Info output file, **`-t <step> -d REPEAT`** and **`-t <step> -d SR`**, respectively.

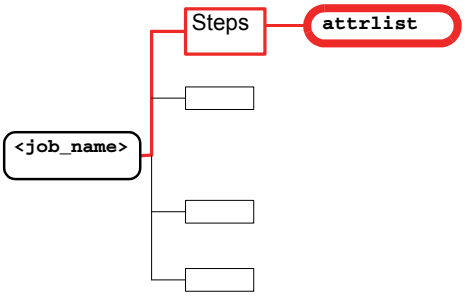
attrlist (Attribute List)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**`<job_name>/steps/<step_name>/attrlist`**

This file contains the values for attributes (system and user) of a step.

Example

```
.out_drill_full = no
.out_drill_optional = no
.out_rout_optional = no
.fs_direction_top =
left2right
.fs_direction_bottom =
right2left
comment = Production Step
```



layers (See Chapter 6)

Netlists

Required for GenFlex 6.4

In ODB++, a testable inner net point has the line prefix @**t**. Also, the ODB++ code describing the net point contains the name of the layer in which it is located. A highlighted sample ODB++ file is given below.

```
H optimize n staggered n
$0 NET00000
$1 NET00001
#
#Netlist points
#
) 0.01 0.0689076 0.3563026 B e e arsize_top=0 arsize_bot=0 is_shrink=n
) 0.015 0.3983194 0.3680672 T e e arsize_top=0 arsize_bot=0 is_shrink=n
@t 0 0 0.1294118 0.1899159 T l1 0.02 0.04 m e staggered 0 0 0  is_shrink=n
@t 0 0 0.1764707 0.1445378 T l1 0.02 0.04 m e staggered 0 0 0  is_shrink=n
@t 0 0.01 0.1668989 0.2306203 B buried m e  arsize_top=0 arsize_bot=0 is_shrink=n
@t 0 0 0.2285715 0.2319328 D l2 0.02 0.04 m e staggered 0 0 0  is_shrink=n
 0.01 0.0739496 0.0302522 B e e arsize_top=0 arsize_bot=0 is_shrink=n
 0.015 0.4184874 0.0252101 T e e arsize_top=0 arsize_bot=0 is_shrink=n
@t 1 0 0.2319328 0.184874 D l2 0.04 0.02 m e staggered 0 0 0  is_shrink=n
 0 0.2319328 0.0336135 D 0.03 0.03 e e staggered 0 0 0 is_shrink=n
```

In the netlist description

- The line prefix @t (highlighted in yellow)
- The words "l1", "l2", and "buried" are the layer names in which these net points are located. These layer names are highlighted in grey, and do not appear in lines that do not describe testable inner net points.
- Side Flags labels for testable inner net points (highlighted in red)

  T - Top Test Points

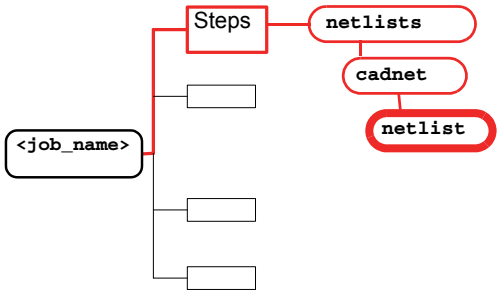
  D - Down Test Points

  B - Both Test Point
- For testable inner net points, the layer name is listed immediately after the side flag (T,D,B). This is visible above, where the grey-shaded boxes with line names appear immediately after the red-shaded side flags.

cadnet / netlist (CADnet)

Type:	Line Record Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/netlists/cadnet/netlist**

This file represents a netlist as it was read from an external CAD system. A netlist is a collection of nets, each one referencing a group of points.



Definitions

Basic Netlist - contains only drill holes stored for drilled SMD pads.

Extended Netlist - contains both holes and drilled SMD.

Extended Netpoint - drill hole which has associated SMD pads.

Complex Netpoint - consists of both the hole and drilled SMD's, as stored in the netlist.

Netpoint Extension - drilled SMD pad stored in netlist together with the drill hole.

Test Side - of net-point is stored in netlist and determined by the Netlist Optimizer (cannot be changed except with the Electrical Testing Manager (ETM).

Example

```
H optimize n
$0 &1N1096
$1 &1N1526
$2 &1N289
$3 &1N312
$4 &1N338
$5 &1N340
$6 $NONE$
...
#
#Netlist points
#
0 0.002 4.96 -2.64 T e e staggered 0 0 0
0 0.002 4.94 -2.4775 T e e staggered 0 0 0
0 0.002 4.945 -2.575 B e e staggered 0 0 0
1 0.002 4.31 -4.045 T e e staggered 0 0 0
1 0.002 4.27 -3.893 T e e staggered 0 0 0
```

Note When a feature does not have a net defined it is assigned to **`NET $NONE$`**. All non-assigned features are defined as **`NET $NONE`** (see "NET - Electrical Net Record" on page 86).

The first line of the file has the form:

`H optimize <y|n>`

`y` - if netlist was optimized by the netlist optimizer

`n` - if netlist was not optimized

For CAD netlist, the net will always be non-optimized.

The next section of the file contains the nets, in the following format:

`$<serial_num> <net_name>`

Where:

<serial_num>	**is the net serial number, starting with 0**
<net_name>	**is the original net name as read from CAD**

Net names are for reference only. Every **<serial_num>** net is considered a different electrical net, and should be electrically isolated from all different **<serial_num>** nets. Ideally, each and every **<serial_num>** net should have a different **<net_name>**.

The last section contains the net points. Each one has the following format:

```
<net_num> <radius> <x> <y> <side> [ <w> <h> ] <epoint> <exp> [ <c> ]
[staggerred <sx> <sy> <sr>] [v]  [f] [t] [m][<x>] [<e>] [<by>]
```

Where:

net_num	The number of the net (start from -1), corresponding to the previously defined netlist section (when a feature does not belong to a net it is defined as **$NONE$**). Net numbers start from -1 (-1 represents a tooling hole).
radius	Drill radius (inches) or 0.002 for SMD pads
x,y	point coordinates (inches)
side	**T** for top **D** for bottom **B** for both
w,h	Width and height of non-drilled pads (only when radius = 0)
epoint	**e** for net end point **m** for net mid point
exp	**e** for solder mask exposed point **c** for solder mask covered point **p** for solder mask covered primary point on top layer **s** for solder mask covered secondary point on bottom layer
c	Comment point
sx,sy	Coordinates of staggered point
sr	Radius of staggered point
v	**v** for a via point
f	Fiducial point
t	Test point
m	Appears when a netlist point is designated as a test point by assigning it the **.critical_tp** attribute. Normally this is applied to mid-points that need to be tested. The Netlist Optimizer determines mid-points to be not testable unless assigned this attribute. If both **.non_tp** and **.critical_tp** are assigned to the same point, **.critical_tp** takes precedence and the mid point is tested. In case of a drilled pad, the attribute must be added to the drill hole.
x	‘eXtended' appears if net point is extended
e	‘<Extension>' appears if net point is an extension

by	**{ c s b n }** c - test from component side s - test from solder side b - test from both sides a - test from any one side. n - side not defined (if **<by>** value not defined, **n** is assumed)
arsize_top	'Annular Ring size for Top' represents the minimum width of exposed copper (from solder mask) around a drill hole on the top outer layer.
arsize_bot	Same as for **arsize_top** but for bottom part of the hole. If hole does not go through top / bottom layer, the corresponding parameter (**arsize_top** / **arsize_bot**) should not be defined or set to 0. Parameters are keyword parameters and may be placed at any place after the positional ones.
is_shrink	Y - point size was shrunk to fit solder-mask opening. N - point size is limited only by pad size.

***Example -* arsize_top / arsize_bot**

```
4 0.023622 0.726 0.3351969 B m e  arsize_top=0.016378 arsize_bot=0.0161873
4 0.015748 0.7460787 0.5300787 B e e by=a arsize_top=0.011752 arsize_bot=0.0116406
4 0.011811 0.0358425 0.1450394 B m c  arsize_top=0 arsize_bot=0
```

Example -

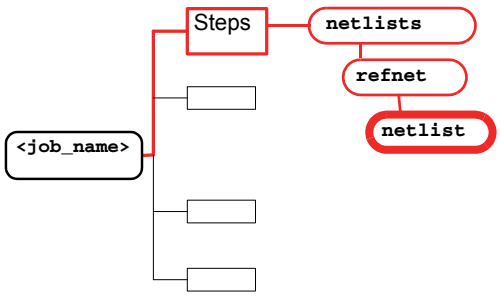
```
0 0.00675 0.8 3.3 B m e v x by=b
0 0 2.5 3.214393 T 0.04242 0.04242 e s staggered 0 0.01325 0 e by=c
```

A net point description for an extended point does not have to be grouped together in the netlist file.

refnet / netlist (Reference)

Type:	Line Record Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/netlists/refnet/netlist**

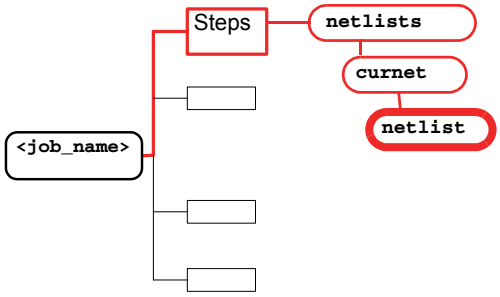
This file contains the reference netlist for the step. A reference netlist can be copied from the CAD netlist, the current netlist or the current-based-cad netlist.



curnet /netlist (Current)

Type:	Line Record Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/netlists/curnet/netlist**

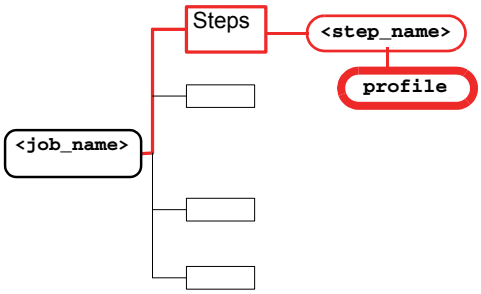
This file contains the Current netlist for the step. This is a temporary netlist that exists in the system memory only and is never saved with the job. It is extracted from the board layer in its current edited state, and always reflects any edits or modifications.



profile (Outline Shape of Step)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/profile**

The profile provides the outline shape of the step. It is required by many operations. A profile can be one closed polygon shape.



Example

```
#
#Layer features
#
S P 0
OB 0 0 I
OS 0 10
OS 10 10
OS 10 0
OS 0 0
OE
SE
```

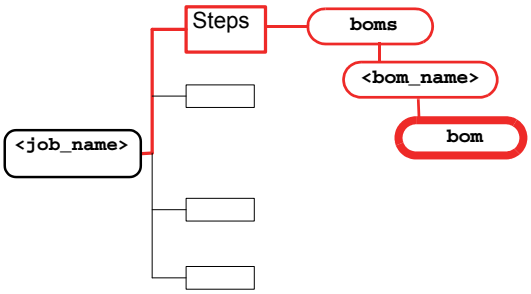
The profile consists of one positive surface feature. Refer to the description of surface features inside the description of **<job_name>/steps/<step_name>/layers/<layer_name>/features** ("features" on page 108).

bom (Bill of Materials)

bom

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job_name>/steps/<step_name>/boms/<bom_name>/bom**

Example



 # Header Parameters

```
HEADER
BRD
REV
HEADER_END
```

 # Description Aliases

```
DESC_ALIASES
LANG fr
INDEX 1
CPN Cout
CPN Cost
```

maximum of 10 entries

```
INDEX 1
MPN Benefice
MPN Profit Margin
```

maximum of 10 entries

```
DESC_ALIASES_END
```

 # Reference Descriptors and matching Customer Parts

```
RD_CPN
REF XTAL1
LNFILE 5 Rev14.v1
```

(where **5** *is the source line number and* **Rev14.v1** *is the source BOM)*

```
CPN 004-020-101
LNFILE 5 Rev14.v1

REF Y8
LNFILE 7 Rev14.v1
CPN 004-040-101
LNFILE 7 Rev14.v1


RD_CPN_END
```

 # Customer Parts and matching Manufacturer Parts

```
CPN_MPN
CPN 004-020-101
LNFILE 5 Rev14.v1
VPL_MPN
VPL_VND TOYOCOM
```

```
MPN TQC-216C-6R
LNFILE 5 Rev14.v1
VND TOYOCOM
LNFILE 5 Rev14.v1
QLF 0
CHS 1

CPN 004-020-101
LNFILE 5 Rev14.v1
VPL_MPN
VPL_VND VF
MPN VM6S-20.0000-16PF
LNFILE 6 Rev14.v1
VND VALPEY-FISHER
LNFILE 6 Rev14.v1
QLF 0
CHS 0


CPN_MPN_END
```

 # Customer Parts and description

```
CP
CPN 004-020-101
LNFILE 5 Rev14.v1
IPN
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
PKG
LNFILE 0 Rev14.v1
QNT 1
ITEM 0

CPN 004-040-101
LNFILE 7 Rev14.v1
IPN
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
DSC
```

```
LNFILE 0 Rev14.v1
DSC
LNFILE 0 Rev14.v1
PKG
LNFILE 0 Rev14.v1
QNT 1
ITEM 0

CP_END
FILE_END
```

Description

The file is divided into four sections. Each section starts with a header (equivalent to section name) and ends with *name*_**END**.

HEADER section

Contains two parameters:

Parameter	Description
BRD	Board number
REV	Revision

Source information for those parameters indicated in each section are saved in the corresponding **files** sub-directory (for example, **LNFILE 5 Rev14.v1** (*where* **Rev14.v1** *is the source file and* **5** *is the source line number*).

DESC_ALIASES section

Contains seven parameters:

Parameter	Description
LANG	One of the possible languages allowed by environment variable **GENESIS_LANG**.
INDEX	A numeral, 1-10 corresponding to the 10 descriptions in the BOM to replace **DESC<index>**.
CPN	The alias for the CPN field in the BOM in the selected language.
CPN	The alias for the CPN field in the BOM in English.
INDEX	A numeral, 1-10 corresponding to the 10 descriptions in the BOM to replace **PART_DESC<index>**.
MPN	The alias for the MPN field in the BOM in the selected language.
MPN	The alias for the MPN field in the BOM in English.

RD_CPN section

Contains the Reference Descriptors and their matching Customer Parts:

Parameter	Description
REF	Reference designator name
CPN	Customer part number

LNFILE is saved for all parameters.

CPN_MPN section

Contains Customer Parts and their matching Manufacturer Parts:

Parameter	Description
CPN	Customer Part Number
VPL_MPN	MPN from the VPL database corresponding to original MPN (as determined in BOM Validation)
VPL_VND	Manufacturer from the VPL corresponding to original Vendor (as determined in BOM Validation)
MPN	Manufacturer Part Number
VND	Manufacturer (Vendor) name
QLF	Qualify - whether the part (**MPN+VENDOR**) is qualified for production: **-1** - Not qualified **0** - Unknown **1** - Qualified
CHS	Chosen - if this part is chosen from among the alternate parts for the CPN. Only one can be Chosen.

LNFILE is saved for **CPN MPN VND**

CP section

Contains Customer Parts and their description:

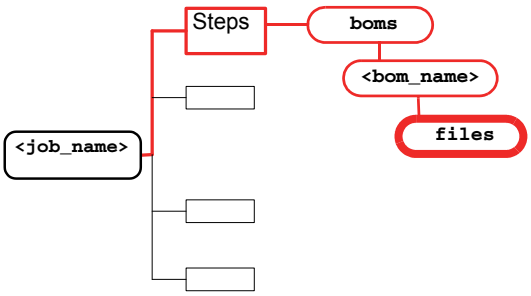
Parameter	Description
CPN	Customer Part Number
IPN	Internal Part Number
DSC	Up to 5 descriptions
PKG	Package name
QNT	Reference Designator quantity
ITEM	Item number

LNFILE is saved for **CPN IPN DSC PKG**

files (Source Files)

Type:	Directory
Compression:	None
Sum file:	No
Path	**<job_name>/steps/<step_name>/boms/<bom_name>/files**

The **files** sub-directory contains the source files that generated the BOM entity (BOM and AVL).

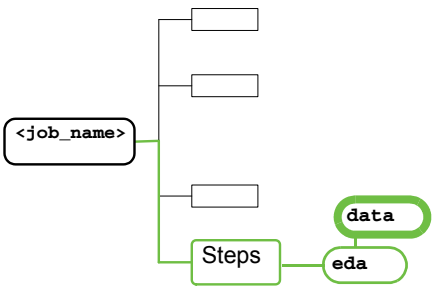


eda (Electronic Design Automation)

data

Type:	Line Record Text
Compression:	Yes
Sum file:	No
Path	**<job_name>/steps/<step_name>/eda/data**

This file contains information which is read from the EDA system database directly. It covers the library of CAD and user-defined VPL packages, their outlines and properties, net connectivity information and more.



Example

```
#
HDR Mentor Boardstation database
LYR sst sst+1 sigt sig2 sig3
sig4 sig7 sig8 sig9 sigb smt smb
drill spt ssb spb ssb+1 pg6 pg5
#
PRP MILLING_ORIGIN 'MILLING 0 0.0 0' 0 0
PRP DRILL_ORIGIN '' 0 0
PRP FIXED_COMPONENT_LOCATION 'M3,m1' 1
....
#
#Net attribute names
```

```
#
#@0 .critical_net
#@1 .diff_pair
#@2 .net_type
#@3 .electrical_class
#@4 .dpair_gap
#@5 .eclass_rise_time
#@6 .eclass_max_stub_length
#
#
#Net attribute text strings
#
#&0 DEFAULT
#&1 clocks
#&2 EC_PUA
#&3 local
...

# NET 0
NET /D_CL_TX_CLK ;0,2=1,3=2,5=1.500000
SNT TOP T 16 0
FID C 2 33
FID C 10 33
FID C 13 30
...
# PKG 1
PKG *PDXC-L10/HX-L127W51T97 0.1 -0.25 -0.145 0.25 0.145
RC -0.25 -0.1 0.5 0.2
PIN 1 T -0.2 -0.1105 0 E S
RC -0.2125 -0.145 0.025 0.069
PIN 3 T -0.1 -0.1105 0 E S
RC -0.1125 -0.145 0.025 0.069
PIN 5 T 0 -0.1105 0 E S
RC -0.0125 -0.145 0.025 0.069
PIN 7 T 0.1 -0.1105 0 E S
RC 0.0875 -0.145 0.025 0.069
PIN 9 T 0.2 -0.1105 0 E S
RC 0.1875 -0.145 0.025 0.069
PIN 2 T -0.2 0.1105 0 E S
RC -0.2125 0.076 0.025 0.069
PIN 4 T -0.1 0.1105 0 E S
RC -0.1125 0.076 0.025 0.069
PIN 6 T 0 0.1105 0 E S
RC -0.0125 0.076 0.025 0.069
PIN 8 T 0.1 0.1105 0 E S
RC 0.0875 0.076 0.025 0.069
PIN 10 T 0.2 0.1105 0 E S
RC 0.1875 0.076 0.025 0.069
#
# PKG 2
PKG *MBCY-T2/XC-L80W80T115 0.1377953 -0.1574803 -0.1574803
0.1574803
0.1574803
```

```
CR 0 0 0.1574803
PIN P T 0.0688976 0 0 E T
CR 0.0688976 0 0.011811
PIN N T -0.0688976 0 0 E T
CR -0.0688976 0 0.011811

# FGR 1907
FGR TEXT
PRP string '030'
FID C 14 11018
FID C 14 11018
```

The file consists of records of the following types:

Main Records:

HDR	File Header
LYR	Layer Names
NET	Electrical Net Record
SNT	Subnet Record
PKG	Package Record
PIN	Pin Record
FGR	Feature Group Record
FID	Feature ID record
PRP	Property record

Net Attributes Header:

This header contains a table of net attribute names as well as a table of all net attribute values that are strings. The structure of the net attributes header is the same as that for features and components, except that each line begins with #.

```
#Net attribute names
#@<num1> <attribute_name>
#@<num2> <attribute_name>
#Net attribute text strings
#&<num3> <string>
```

Usage: <num1>,<num2>=<num3>

Outline Records:

CR	Circle record
SQ	Square Record
RC	Rectangle record
CT, OB, OS, OC, OE, CE	Contour record

Following is the format and description of each record.

HDR - File Header

This record contains the EDA system which was the source of the data.

Structure:

HDR <source>

Where:

<source> can be:

"Mentor Boardstation neutral file"

"Mentor Boardstation database"

"Cadence Allegro extract file"

"Zuken Redac CADIF file"

"PADS PowerPCB"

LYR - Layer Names

This record contains the names of the layers which are referenced in FID records later.

LYR <name1> .... <namen>

Where:

<namex>	A legal name of a layer listed in the job matrix

NET - Electrical Net Record

This record contains a start record of an electrical net. Each net consists of one NET line and 0 or more SNT records

NET <name>

Where:

<name>	The name of the net as defined in the EDA system*
<attributes>	This data is the same as for feature attributes (in the features file). It consists of comma separated list of values. Each can be: **n** indicating that (boolean) attribute **n** is set **n=m** indicating that option attribute **n** has value **m** **n=i** indicating that integer attribute **n** has value **i** **n=f** indicating that floating attribute **n** has value **f** **n=s** indicating that text attribute **n** has header value **s**
Note: n must match a @ record in the attribute header **s** must match a & record in the attribute header.	

Net names are for reference only. Every NET record is considered a different electrical net, and should be electrically isolated from all NET records. Ideally, the **<name>** should be unique across all NET records.

When a feature does not have a net defined it is assigned to **NET $NONE$**. All un-assigned outer layer pads are defined as **NET $NONE$**. With more than one **$NONE$** net, each is disconnected from the other. Any two points of a **$NONE$** net can be connected or disconnected, depending on the design.

In the Compare function of the Netlist Analyzer, disconnected **$NONE$** nets are not reported as opens. Shorts between **$NONE$** nets and other nets are reported. If no special treatment is done on the Xpert then if a **$NONE$** net points form more than one net typically they will all be reported as a large **$NONE$** net broken into subnets.

SNT - Subnet Record

This record contains a portion of a net. This portion can be:

toeprint	A connection of a component pin to the board
via	A connectivity padstack between layers
trace	A collection of lines/arcs leading from point to point
plane	A surface used for connectivity purposes

Each subnet record is followed by zero or more FID records mapped to the board features which are part of this subnet.

Structure for toeprint:

```
SNT TOP <side>  <comp_num>  <pin_num>
```

Where:

<side>	**T** for TOP, **B** for bottom
<comp_num>	Number of component in the components file (**comp_+_top/** components or **comp_+_bot/**components)
<pin_num>	Number of pin in the component

Structure for via

```
SNT VIA
```

Structure for trace

```
SNT TRC
```

Structure for plane

```
SNT PLN <fill_type> <cutout_type> <fill_size>
```

Where:

<fill_type>	**S** for solid **H** for hatched **O** for outline
<cutout_type>	**C** for circle **R** for rect **O** for octagon **E** for exact
<fill_size>	Size in inched of fill brush

Note The values for **SNT PLN** must appear with legal values, but the software does not consider them internally.

PKG - Package Record

This record contains a definition of a package, which is the generic shape of a component (e.g. each component refers to a package).

Each **PKG** line **must** be followed immediately by an outline record/s, 0 or more property (PRP) records and 0 or more PIN records.

Structure:

```
PKG <name> <pitch> <xmin> <ymin> <xmax> <ymax>
```

Where:

<name>	The name of the package as defined in the EDA system (geometry in Mentor terms, SYM_NAME in Cadence terms)
<pitch>	Distance between center of closest pins, in inches
<xmin>,<ymin>,<xmax>,<ymax>	Bounding box of package, relating to package datum

Note ODB++ requires closed geometries (polygons must be closed).

PIN - Pin Record

This record contains a definition of a pin, which belongs to a package. Each pin is followed by (an) outline record(s).

Structure:

PIN <name> <type> <xc> <yc> <fhs> <etype> <mtype>

Where:

<name>	The name of the pin as defined in the EDA system
<type>	**T** for thru-hole (top--->bottom) **B** for blind (<top--->inner or inner--->bottom) **S** for surface (<top--->top or bottom-->bottom)
<xc> <yc>	Center of pin, relating to package datum
<fhs>	Finished hole size (Unused at the moment - should be 0)
<etype>	PIN Electrical Type: E - Electrical; M - Non-Electrical (Mechanical); U - Undefined
<mtype>	PIN Mount Type. S - SMT. D - Recommended SMT pad (where the pin size is the recommended pad size and not the pin size). T - Thru-hole. R - Thru-hole where the pin size is the recommended hole size and not the pin size. P - Pressfit. N - Non board, pins without contact area with the board. Used in components with lead forms of types: Solder Lug, High Cable, or Quick Connect. H - Hole, for physical holes that appear without the physical pin. U - Undefined.

By default, the last two parameters (**<pin_type>** and **<pin_mount_type>**) are defined as 'U' (Unknown) Only for packages that are imported from the VPL database are they defined otherwise.

FGR - Feature Group Record

This record contains the definition of a group of related features (e.g. the strokes of a text record).

Each **FGR** line is followed by zero or more **FID** records mapped to the board features which are part of this subnet.

```
FGR <type>
```

Where:

<type>	Only allowed value is TEXT

FID - Feature ID Record

This record contains a link to a feature in the board. The record is used to connect subnets and feature groups to the board features which are part of them.

```
FID <type> <lyr_num> <f_num>
```

Where:

<type>	**C** - copper **L** - laminate **H** - hole
<lyr_num>	A layer number (0 ... n-1) corresponding to the names of layers in the LYR record described earlier
<f_num>	A feature number (0 ... n-1) corresponding to the feature record sequence in the features file*.

* See below for example of **f_num** sequence in a feature file:

```
#
#Layer features
#
P 4.057087 4.5 6 P 0 0;1=6,3=0          #f_num = 0
P 4.057087 4.57874 6 P 0 0;1=6,3=0      #f_num = 1
P 4.057087 3.633858 6 P 0 0;1=6,3=0     #f_num = 2
P 4.057087 3.712598 6 P 0 0;1=6,3=0     #f_num = 3
```

PRP - Property Record

This record represents a property of the board, a net, a package or a feature group. A property consists of a name, a string value and 0 or more floating numbers.

```
PRP <name> '<value>' n1 n2 ...
```

Where:

<name>	The name of the property
<value>	The string of the property (between quotes)
n1,n2,...	The floating numbers to be kept in the property

Outline Records

Outline records **must** follow a **PKG** or **PIN** record. They describe the shape of the package/pin.

Note A PKG record must have an outline record as the immediate next entry (an outline record can be more than one line). A PIN record does require an outline record but not immediately after.

A shape can consist of a simple shape (circle, square, rectangle) or a complex contour.

CR - Circle record

```
CR <xc> <yc> <radius>
```

SQ - Square Record

```
SQ <xc> <yc> <half side>
```

RC - Rectangle record

```
RC <lower_left_x> <lower_left_y> <width> <height>
```

CT ... CE - Contour record

The structure of a contour record is the same as a surface feature in the features file and is restricted by the same limitations.

A contour consists of one or more polygons.

- Intersection is not allowed between edges of the same polygon

- Intersection is not allowed between edges of different polygons

- The polygons must form a closed shape

- Holes must be graphically contained inside island polygons

- The curves must be consistent (the start, end, and center point must construct a legal curve).

A polygon starts with **OB** command, contains **OS** (segment) or **OC** (curve) commands and ends with an **OE** command.

```
OB <start_x> <start_y>  I/H
```
(I=island, H=hole)

```
OS <end_x> <end_y>
```

```
OC <end_x> <end_y>  <center_x> <center_y> <cw>
```
(cw = Y or N)

```
OE
```

Net Attributes

The net attributes are found in the file “data”, under the EDA directory of the job.

Each net can have attributes in the same way it is done for features and components. That is, each net name may be followed by a semi-colon followed by net attribute values, i.e., ‘**NET <net_name> ; <net attributes>**’

Also, in the EDA ‘data’ file, the net attributes header is found. This header contains a table of net attribute names as well as a table of all net attribute values that are strings. The structure of the net attributes header is exactly like the one used for features and components. The only difference is that for net attributes the header is commented (with #), in order to be read by Enterprise versions prior to v5.3. The header is located before the first net record.

For example:

NET /D_CL_TX_CLK;0,2=1,3=2,5=1.500000

This should be interpreted as follows:

Net named '**/D_CL_TX_CLK**' has the following attributes:

- attribute #0,
- attribute #2, value: 1
- attribute #3, value: 2
- attribute #5, value: 1.5

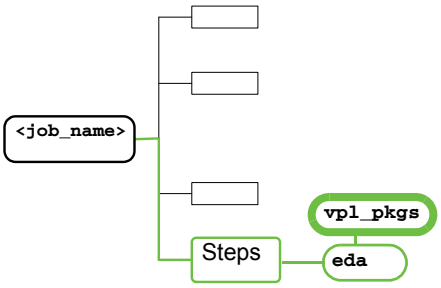
A look at the attribute header reveals the following:

- attribute #0 is **.critical_net**, which is boolean, thus its appearance means: TRUE.
- attribute #2 is **.net_typ**e, which is of type string; its value is index 1, i.e., "clocks".
- attribute #3 is **.electrical_class**, which is of type string; its value is index 2, i.e., "EC_PUA".
- attribute #5 is **.eclass_rise_time**, which is float and its value is 1.5.

vpl_pkgs

Type:	Encrypted
Compression:	No
Sum file:	No
Path	**<job_name>/steps/<step_name>/eda/vpl_pkgs**

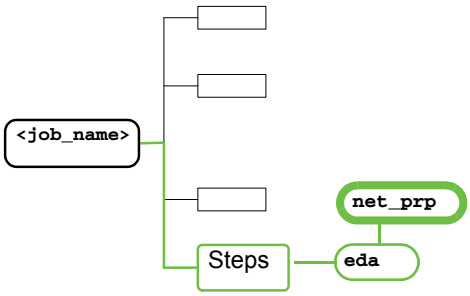
This file contains encrypted information relating to the library of VPL packages taken from the Valor Parts Library.



S net_prp (Net Type) Clearances Records

Type:	
Compression:	
Sum file:	
Path	**<job_name>/steps/<step_name>/eda/net_prp**

The EDA directory of the job, contains a file named **net_prp**. This file contains net type clearances. Versions prior to V5.3 are not aware of its existence, and thus cannot take advantage of it. There are two types of net type clearance records:



- Independent of constraint area **NET_TYPE_CLEARANCES**.
- Dependent upon constraint area. (Ignored by software versions prior to V7.1.)

The structure of this file is a list of net type clearance, net type physical and/or electrical parameter records.

Independent of Constraint Area

```
NET_TYPE_CLEARANCES {
    net_type1 = <net type 1>
    net_type2 = <net type 2>
    layers = <layer names>
    via2via = <clearance>
    trace2trace = <clearance>
    via2trace = <clearance>
    pin2pin = <clearance>
    via2pin = <clearance>
    trace2pin = <clearance>
    plane2plane = <clearance>
    via2plane = <clearance>
    trace2plane = <clearance>
    pin2plane = <clearance>
}
```

<net type 1> and **<net type 2>** are either net types that are supposed to be defined in the ‘data’ file, or an asterisk ‘*’.

<layer names> is either a list of the job layer names (separated by semi-colons ‘;’) or an asterisk ‘*’.

The 10 clearances that come after are optional. If exist, they are given in inches.

Examples:

```
NET_TYPE_CLEARANCES {
    net_type1 = local
    net_type2 = clock
    layers = sigt;sig4
    via2via = 0.005000
    trace2trace = 0.005000
```

```
    ...
}

NET_TYPE_CLEARANCES {
    net_type1 = local
    net_type2 = *
    layers = *
    via2via = 0.005000
    trace2trace = 0.005000
    ...
}
```

Dependent Upon Constraint Area

```
CNSA_NET_TYPE_CLEARANCES {
   constr_area = <area name>
   net_type1 = <net type 1>
   net_type2 = <net type 2>
   layers = <layer names>
   via2via = <clearance>
   trace2trace = <clearance>
   via2trace = <clearance>
   pin2pin = <clearance>
   via2pin = <clearance>
   trace2pin = <clearance>
   plane2plane = <clearance>
   via2plane = <clearance>
   trace2plane = <clearance>
   pin2plane = <clearance>
   bb_via2bb_via = <clearance>
   bb_via2line = <clearance>
   bb_via2smd_pin = <clearance>
   bb_via2shape = <clearance>
   bb_via2tst_pin = <clearance>
   bb_via2tst_via = <clearance>
   bb_via2thru_pin = <clearance>
   bb_via2thru_via = <clearance>
   line2line = <clearance>
   line2smd_pin = <clearance>
   line2shape = <clearance>
   line2tst_pin = <clearance>
   line2tst_via = <clearance>
   line2thru_pin = <clearance>
   line2thru_via = <clearance>
   shape2smd_pin = <clearance>
   shape2shape = <clearance>
   shape2tst_pin = <clearance>
   shape2tst_via = <clearance>
   shape2thru_pin = <clearance>
   shape2thru_via = <clearance>
   smd_pin2smd_pin = <clearance>
   smd_pin2tst_pin = <clearance>
   smd_pin2tst_via = <clearance>
   smd_pin2thru_pin = <clearance>
```

Clearances for Mentor jobs. (Also applies to Cadence jobs created in V7.2 or earlier.

Clearances from **bb_via2bb_via** to **tst_via2bond_pad** are for Cadences jobs from V7.3.

```
    smd_pin2thru_via = <clearance>
    tst_pin2tst_pin = <clearance>
    tst_pin2tst_via = <clearance>
    tst_pin2thru_pin = <clearance>
    tst_pin2thru_via = <clearance>
    tst_via2tst_via = <clearance>
    tst_via2thru_pin = <clearance>
    tst_via2thru_via = <clearance>
    thru_pin2thru_pin = <clearance>
    thru_pin2thru_via = <clearance>
    thru_via2thru_via = <clearance>
    hole2hole = <clearance>
    hole2owire = <clearance>
    owire2owire = <clearance>
    dwire2hole = <clearance>
    dwire2owire = <clearance>
    thru_pin2bond_pad = <clearance>
    smd_pin2bond_pad = <clearance>
    thru_via2bond_pad = <clearance>
    bond_pad2bond_pad = <clearance>
    bond_pad2line = <clearance>
    bond_pad2shape = <clearance>
    bb_via2bond_pad = <clearance>
    tst_pin2bond_pad = <clearance>
    tst_via2bond_pad = <clearance>
    dpair_sep_prim = <clearance>
    dpair_spe_scnd = <clearance>
}
```

<area_name> is the name of the constraint area (an arbitrary string of less than 65 characters or an asterisk '*').

<dpair_sep_prim> and **<dpair_sep_scnd>** are determining clearances for differential pair nets, regardless of feature type. These fields are optional. When they exist, they are expressed in inches.

Other fields are the same as in **NET_TYPE_CLEARANCES**.

Examples:

```
CNSA_NET_TYPE_CLEARANCES {
    constr_area = AGP-EDGE-AREA
    net_type1 = NO-TYPE
    net_type2 = NO-TYPE
    layers = top
    tst_via2tst_via = 0.010000
    line2line = 0.010000
    tst_via2line = 0.008000
    smd_pin2smd_pin = 0.025000
    smd_pin2tst_via = 0.025000
    line2smd_pin = 0.009700
    shape2shape = 0.010000
    shape2tst_via = 0.006000
    line2shape = 0.010000
    shape2tst_pin = 0.010000
    dpair_sep_prim = 0.005000
```

```
    dpair_sep_scnd = 0.050000
}

CNSA_NET_TYPE_CLEARANCES {
    constr_area = *
    net_type1 = *
    net_type2 = *
    layers = int4
    tst_via2tst_via = 0.005000
    line2line = 0.010000
    bb_via2line = 0.007000
    thru_pin2thru_via = 0.005000
    smd_pin2thru_via = 0.005000
    line2smd_pin = 0.007000
    shape2shape = 0.010000
    shape2tst_via = 0.008000
    line2shape = 0.010000
    shape2smd_pin = 0.010000
    dpair_sep_prim = 0.005000
    dpair_sep_scnd = 0.050000
}
```

```
CNSA_KEY_NET_TYPE_CLEARANCES {
    constr_area = <area name>
    net_type1 = <net type1>
    net_type2 = <net type2>
    layers = <layer names>
    main_set_name = <set name>
}
```

<main_set_name> is the name of the set assigned to the **CNSA_NET_TYPE_CLEARANCES** set with the same values of **constr_area**, **net_type1**, **net_type2** and **layers**.

```
CNSA_KEY_NET_TYPE_CLEARANCES {
    constr_area = <area name>
    net_type1 = <net type1>
    net_type2 = <net type2>
    layers = <layer names>
    set_name = <set name>
}
```

<set_name> is the name of the new set with the same clearances as the one named **main_set_name**.

```
CNSA_NET_TYPE_PHYSICAL_PARAMS {
    constr_area = <area name>
    net_type = <net type>
    layers = <layer names>
    min_line_width = <parameter value>
    min_neck_width = <parameter value>
    max_line_length = <parameter value>
}
```

<area_name> is the name of the constraint area (an arbitrary string of less than 65 characters or an asterisk '*').

<net type> is either the net type that is supposed to be defined in the 'data' file, or an asterisk '*'.

<layer names> is either a list of the job layer names (separated by semi-colons ';') or an asterisk '*'.

The three parameter values are optional. If they exist, they are given in inches.

Note The net type physical parameters record is ignored by software versions prior to V7.1.

Examples:

```
CNSA_NET_TYPE_PHYSICAL_PARAMS {
    constr_area = BGA-080-AREA
    net_type = 36MIL-TRACE
    layers = bottom
    min_line_width = 0.005000
    min_neck_width = 0.004000
    max_line_length = 0.100000
}

CNSA_NET_TYPE_PHYSICAL_PARAMS {
    constr_area = *
    net_type = *
    layers = top
    min_line_width = 0.005000
    min_neck_width = 0.004000
    max_line_length = 0.100000
}
```

Net Type Electrical Parameter Set

```
NET_ELECTRICAL_PARAMS {
    ecset_name = <set name >
    dpair_prim_gap = <parameter value>
    dpair_line_width = <parameter value>
    dpair_neck_gap = <parameter value>
    dpair_neck_width = <parameter value>
    dpair_coupled_tol_min = <parameter value>
    dpair_coupled_tol_pl = <parameter value>
    dpair_minimum_spacing = <parameter value>
    dpair_gather_control = <parameter value>
    dpair_max_uncoupled_len = <parameter value>
    dpair_phase_control = <parameter value>
    dpair_phase_tolerance_min = <parameter value>
    dpair_phase_tolerance_pl = <parameter value>
}
```

<set name> is the name of the electrical parameters set as read from Cadence Allegro. All parameter values are optional. If they exist, they are expressed in inches.

Example:

```
NET_ELECTRICAL_PARAMS {
    ecset_name = DP-IBM-6GAP
    dpair_prim_gap = 0.006000
    dpair_line_width = 0.004000
    dpair_neck_gap = 0.000000
    dpair_neck_width = 0.004000
    dpair_coupled_tol_min = 0.000100
    dpair_coupled_tol_pl = 0.000100
    dpair_minimum_spacing = 0.005800
    dpair_gather_control = 0.000000
    dpair_max_uncoupled_len = 0.100000
    dpair_phase_control = 0.000000
    dpair_phase_tolerance_min = 0.000000
    dpair_phase_tolerance_pl = 0.000000
}
```

Electrical Set Entry Record

Electrical set entry record is intended to link CAD net and electrical parameter set. The record has the following structure:

```
NET_ECSET_ENTRY {
    net_name = <net name>
    ecset_name = <set name >
}
```

<net name> is the name of the CAD net (not a type as for other records in the file)

<set name> is the name of the electrical parameters set (the same name as mentioned in the **ecset_name** field of **NET_ELECTRICAL_PARAMS**)

Example:

```
NET_ECSET_ENTRY {
    net_name = SC_D_SCSI_SEL_N
    ecset_name = DP-IBM-6GAP
}
```

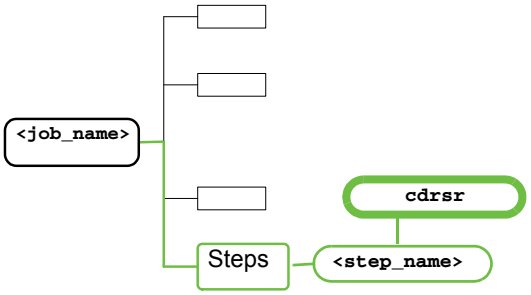
chk (Checklists) (See Chapter 9)

et (See Chapter 10)

S cdrsr (AOI Panelization)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/cdrsr`

This file describes the AOI panelization - a panelization which may be defined for AOI purposes, for the duplication of inspection areas and exclusion zones during the process of preparing CDR setup for Orbotech AOI machines.



Example:

```
PANELIZATION=USER_DEFINE
D

PCB {
    X_DATUM=3.219249311023622
    Y_DATUM=3.702147440944882
    X_MIN=1.797249606299212
    Y_MIN=2.280147834645669
    X_MAX=4.568325787401575
    Y_MAX=5.561685433070866
    SNAP=0
    INSPECT=0
    NAME=1
    STEP=

    STEP-REPEAT {
        NAME=
        X=3.219249311023622
        Y=3.702147440944882
        DX=4.995229330708661
        DY=-0.0364615157480315
        NX=2
        NY=1
        ANGLE=0
        MIRROR=NO
    }
}

USE_STEPS=
GENESIS_VERSION=
```

Description

Parameter	Description
PANELIZATION USER_DEFINED	Implies that the AOI panelization is used for CDR purposes.
PCB	Array defining the PCBs which consist of the AOI panelization.
USE_STEPS	Field not in use.
GENESIS_VERSION	Field not in use.

PCB Array Structure

Parameter	Description
X_DATUM	x datum point (used for step & repeat)
Y_DATUM	y datum point (used for step & repeat)
X_MIN	minimal x coordinate of PCB (for defining PCB dimensions)
Y_MIN	minimal y coordinate of PCB (for defining PCB dimensions)
X_MAX	maximal x coordinate of PCB (for defining PCB dimensions)
Y_MAX	maximal y coordinate of PCB (for defining PCB dimensions)
SNAP	0. Field not in use.
INSPECT	0. Field not in use.
NAME	PCB name as given during definition of AOI panelization. Names are restricted to integer numbers (>= 1).
STEP	Field not in use.
STEP-REPEAT	Array defining the step&repeat of the PCB in the panel. Similar to the STEP-REPEAT array in <job_name>/steps/<step_name>/stephdr.

STEP-REPEAT Array Structure

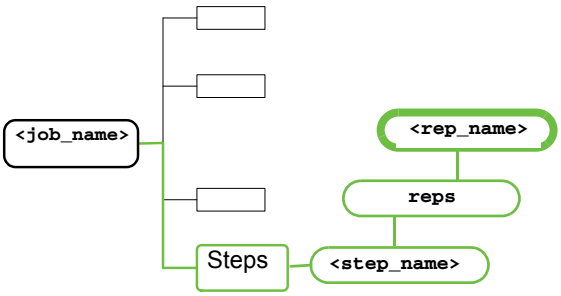
Parameter	Description
NAME	Field not in use. The NAME field as it appears in the PCB array is used instead.
X	Start X coordinate for placement of datum point of step.
Y	Start Y coordinate for placement of datum point of step.
DX	Horizontal distance between datum points (when angle = 0).
DY	Vertical distance between datum points (when angle = 0).
NX	Number of repetitions horizontally.
NY	Number of repetitions vertically.
ANGLE	Rotation angle of the steps (0, 90, 180 or 270 degrees).
MIRROR	YES for mirror (around X axis), NO for no mirror.

S reps (Reports)

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	`<job_name>/steps/<step_name>/reps/<rep_name>`

Example

```
TTL Library Merge Report
MSV 0
CAT No package in
library
CAT No part in library
CAT Ambiguous package
CAT No pin 1
CAT No BOM data
CAT No vendor name
CAT No vendor code
CAT Placement mismatch
(one)
CAT Placement mismatch (all)
CAT Inconsistent package rotation
CAT CAD/VPL pin count mismatch
CAT Package found
CAT Placement successful
_END_CAT
ITM 2 1
TXT VCODE: DALE, MPN: HAZ470MBABRAK
VAL S C388
LYR comp_+_top
AUX art.3
LIM 25222200 6146800 25831800 6553200
SHP S RC 2.4825 0.605 0.06 0.04
ITM 2 1
TXT VCODE: DALE, MPN: HAZ470MBABRAK
VAL S C389
LYR comp_+_top
AUX art.3
LIM 24968200 12623800 25577800 13030200
SHP S RC 2.4575 1.2425 0.06 0.04
```



Description

TTL - Report Title

```
TTL <title>
```

`<title>`	String serving as the report title (for display and for output).

MSV - Maximum Severity

MSV <sev>

<sev>	An integer (0, 1, or 2) indicating the highest severity level of any item in the report: 0 = error (highest possible severity) 1 = warning 2 = informational.

CAT - Category name

CAT <name> ...

_END_CAT - End of categories list

<name>	String serving as the printed/displayed name for this category. The list of categories ends with **_END_CAT**, and categories are later referenced by their index in this list, starting with 1.

ITM - Item entry

ITM <cat> <sev>

<cat>	Index of the category to which this item belongs, in the category listing.
<sev>	Severity of this item: 0 - error 1 - warning 2- informational An ITM record is followed by data pertaining to this item up to the next ITM record or the end of the file.

TXT - Item text description

TXT <string>

<string>	A text string describing this item. If omitted, the item's description will be the name of the category to which this item belongs.

VAL - Item value record

VAL S <str>
VAL I <intval>
VAL D <floatval>

<str>	String value
<intval>	Integer value
<floatval>	Floating-point value

All, some, or none of these records can be present for any item. How the values are interpreted, depends on the viewing method of the report in the code and cannot be modified.

LYR - Item layer

```
LYR <lyrname>
```

<lyrname>	Name of a layer in the job's matrix. This layer is the primary layer associated with the item.

AUX - Auxiliary layers

```
AUX <lyrname1> <lyrname2> ...
```

<lyrname1>, <lyrname2> etc.	Names of layers in the job matrix. These layers serve as 'auxiliary layers' for this item, usually meaning that they will also be displayed when the item is displayed graphically.

LIM - Item limits record

```
LIM <xmin> <ymin> <xmax> <ymax>
```

<xmin>, <xmax>	Lower and upper limits for the X-axis of the graphical display area for this item.
<ymin>, <ymax>	Lower and upper limits for the Y-axis.

Note The limits define a “window” on the board where the location and size of the window are dependent on the type of report item. The report viewing functions of the Engineering Toolkit will zoom to an area twice the size of this window when the report item is highlighted.
When the report items are components, the shape and the limits are based on the component body outline.
In cases where the limits are not set (that is, **xmin==xmax**), the window is based on the “shape” borders.

Note See “Units of Measurement” on page 19.

SHP - Item shape record

```
SHP S <shaperec>
```

<shaperec>	One of the following: **CR** - Circle record **SQ** - Square Record **RC** - Rectangle record **CT...CE** - Contour record

Chapter 6 Job>steps>layers Entity

Requiring Implementation for GenFlex 6.4

Partial S&R Data File

Partial S&R data is saved under the layer directory in a new file called **layerhdr_p**. The file structure is the same as a STEP-REPEAT section in **stphdr** file. The data is used instead of S&R data of the step for display and/or output the layer. File uses only lowest level of steps and their transformations to the panel. In addition the file includes a parameter **PRF_REF_NUM = xxx** where: xxx = 1,2,3, etc. is a reference number to define a certain layer profile (see **lyr_profiles_p** file).
(Downgrade will delete the file) (See "stephdr (Step Header)" on page 71.)

Partial S&R Data File & Layer Profile Reference Number (Encrypted Files)

<job_name>/steps/<step_name>/layers/<lyr_name>/layerhdr_p

Scaling per Step Data File (Implemented in Genesis v9.3b also)

Scaling data is saved under the layer directory in a new file called **subsrdata_p**.
The file structure includes a **SUBSR** sections:

```
SUBSR {
    STEP=PCB
    XA=0.0218488188976378
    YA=0.01848740157480315
    XSCALING=1.001
    YSCALING=1.001
}
```

(Downgrade will delete the file)

Scaling per Step Data File (Encrypted File)

<job_name>/steps/<step_name>/layers/<layer_name>/subsrdata _p

Dimension File (Encrypted File)

<job_name>/steps/<step_name>/layers/<layer_name>/dimension_p

Description of the Dimension File

File name: “dimension”

File location: Inside a layer directory that is inside a step.

Header

Version	Version number of file. This value is currently unused. It may be used in the future if the data format of the dimension entity changes
Scale Percent	Indicates the scaling of the drawing as a percentage
PS_SIZE	The name of the postscript page size of the drawing
PS_ORIENTATION	Landscape or Portrait
Paper Width	Paper width in inches
Paper Height	Paper height in inches
Paper Pos X & Paper Pos Y	The coordinates of the lower left corner of the page
Paper_*_Margin	The top
bottom	left and right margins of the page
Active_*	The coordinates of the active area of the drawing. (This is the region where the actual drawing is supposed to be drawn)
Line width	The width (in points) of the dimension drawing lines
Post_Decimal_Dist	The number of places after the decimal point to be used for distances
Post_Decimal_Pos	The number of places after the decimal point to be used for locations
Post_Decimal_Angle	The number of places after the decimal point to be used for angles
FONT	The name of the font to use for the drawing
FONT_WIDTH	The width of the font in points

FONT_HEIGHT	The height of the font in points
EXT_OVERLEN	How far the extension lines extend past the dimension line
CENTER_MARKER_LEN	The length of the center marker in file units
BASELINE_SPACING	The spacing to be used between measurements in Baseline dimensions
DIMENS_COLOR_FEATURE	The output color of layer features
DIMENS_COLOR_DIMENS	The output color of dimensions
DIMENS_COLOR_DIMENS_TEXT	The output color of dimension text
DIMENS_COLOR_PROFILE	The output color of the profile
ORIGIN_X,ORIGIN_Y	The location of the origin of the drawing

Sample Header:

```
VERSION=1
SCALE_PERCENT=300
PS_SIZE=A4
PS_ORIENTATION=LANDSCAPE
PAPER_WIDTH=12
PAPER_HEIGHT=8.3
PAPER_POS_X=-1.0
PAPER_POS_Y=-0.8
PAPER_TOP_MARGIN=0.1
PAPER_BOTTOM_MARGIN=0.1
PAPER_LEFT_MARGIN=0.1
PAPER_RIGHT_MARGIN=0.1
ACTIVE_X00=0.3
ACTIVE_Y00=0.3
ACTIVE_X11=1.9
ACTIVE_Y11=1.9
LINE_WIDTH=0.22
POST_DECIMAL_DIST=3
POST_DECIMAL_POS=3
POST_DECIMAL_ANGLE=3
FONT=SIMPLEX.SHX
FONT_WIDTH=4.6
FONT_HEIGHT=4.6
EXT_OVERLEN=0.04
CENTER_MARKER_LEN=0.05
BASELINE_SPACING=0.012
DIMENS_COLOR_FEATURE=757575
DIMENS_COLOR_DIMENS=9900
DIMENS_COLOR_DIMENS_TEXT=99
DIMENS_COLOR_PROFILE=990000
ORIGIN_X=0.0
ORIGIN_Y=0.2
```

Dimension entity

The dimension entity can include many dimension records of the following format:

Type	One of horiz
vert	parallel
radial	diam
angle	center
Ref[1-3]	Coordinates of reference points of the dimension.
Line_Pt_x & Line_Pt_y	A point on the line that the text is written on.
Offset	Offset from measuring location to start drawing line
Arrow_Pos	Arrows drawn INSIDE or OUTSIDE

The dimension record contains a text record that describes the text label of the dimension. It contains the following fields.

Prefix	text to be written before the value
Value	text to be drawn for dimension value
View Units	Whether the units should be written
Units	One of inch,mm,mil,ym,deg,r,g (ym is microns, r is radians, g is gradians) Radians gradians and degrees must be used for and may only be used for angular dimensiosn.
Outside	Whether the text is to be drawn outside the dimension range (Applies only to horizontal, vertical, angular and parallel dimensions)
Tol Up & Tol Down	Text to be written for the upper and lower tolerances of the dimension
Suffix	Text to be written at the end of the measurement text
Note	Text to be written under the measurement text
X & Y	Coordinates of the dimension text
Angle	Agle of the dimension text (in degrees)

Sample dimension record:

```
DIMENSION {
   TYPE=HORIZ
   REF1X=0.56
   REF1Y=0.69
   REF2X=1.30
   REF2Y=0.68
```

```
    REF3X=0
    REF3Y=0
    LINE_PT_X=1.10
    LINE_PT_Y=0.35
    OFFSET=0
    ARROW_POS=INSIDE

    TEXT {
      PREFIX=
      VALUE=0.741
      UNITS=INCH
      VIEW_UNITS=NO
      OUTSIDE=NO
      TOL_UP=
      TOL_DOWN=
      SUFFIX=
      NOTE=
      X=0.56
      Y=0.70
      ANGLE=0
    }
  }
```

The DIMENSION record may also contain the coordinates of up to 2 extension lines. This is supported only for angular dimensions. Below is a sample record.

```
    EXT_LINE {
      PS_X=-0.43
      PS_Y=0.55
      PE_X=-0.47
      PE_Y=0.58
    }
```

Predefined text symbols support

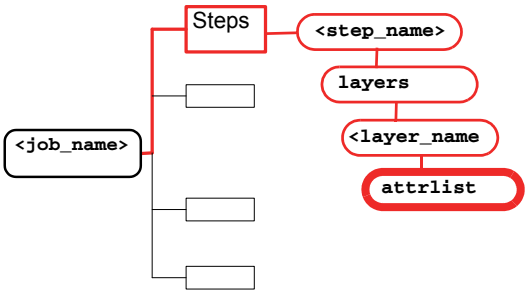
As part of the dimension project the support of adding 4 new text symbols is added to dimensions text. These symbols may be used in text string by typing the pre-defined sequence (like in AutoCAD):

± - %%p
∅ - %%c
° - %%d
µ - %%m

attrlist (Attribute List)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/layers/<layer_name>/attrlist`

This file contains the values for attributes (system and user) of a layer.



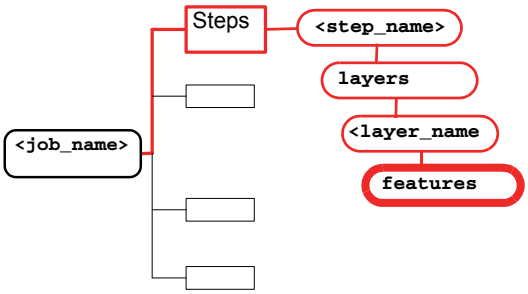
Example

```
.out_mirror = no
.inp_file =
.eda_layers =
"signal_2","signal","VIA"
.out_angle = 0.0
.out_polarity = positive
.out_x_scale = 1.000000
.out_y_scale = 1.000000
.out_comp = 0.000000
```

features

Type:	Line Record Text
Compression:	Yes
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/layers/<layer_name>/features`

The features file contains most of the graphical information of a layer (except for component layers which have the components file). Special symbols also have a feature file to describe their shape.



The feature files have 4 sections:

- Symbols table

  Contains the names of all the symbols used by the features in the file and corresponding serial numbers for reference by the feature records.

- Attribute table

  Contains the names of attributes used by the features in the file, and the corresponding serial numbers for reference by the feature records

- Attribute texts

  Contains a list of text strings which are values for textual attributes.

- Features list

  Contains the actual features data

in 7.0

As of ODB++ V.7.0, features and coordinates are saved in the units in which they were created to avoid loss of precision due to rounding. In every features file saved for each layer and special symbol, there can be a line with the units definition to be applied to the features in the file—**`U <INCH|MM>`**. If this line does not exist, **`INCH`** is assumed. All coordinate values will be interpreted as inches or millimeters. Resize factors for special symbols are interpreted as mils or microns. (Exceptions are discussed where relevant.)

This also applies to semi-standard symbols described at the beginning of the feature file. The format is:

`$<serial_number> <symbol_name> [I|M]`

`I` imperial units

`M` metric units

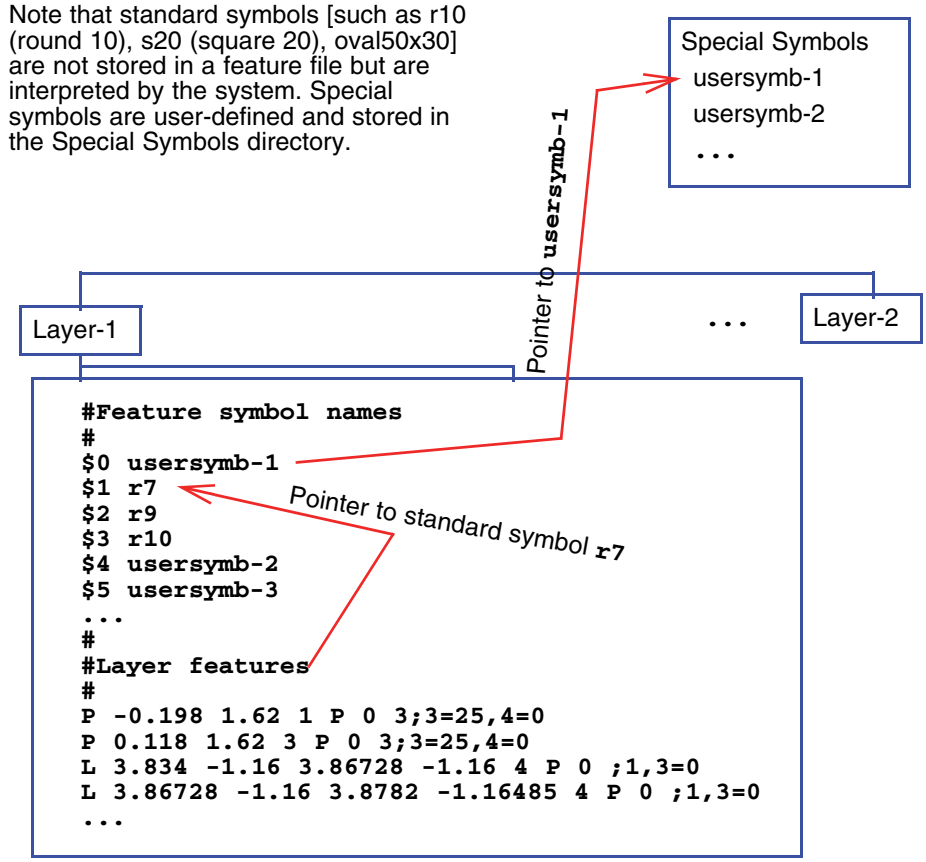
If no unit type is indicated, **`I`** is assumed.

Example

```
#
#Units
#U
U MM
.....
.....
#
#Feature symbol names
#
$0 r120                 #not semi-standard--units are mm as are the layer units
$1 rect20x60 M          # rect 20 by 60 microns
$2 rect3x5 I            #rect 3 by 5 mils
$3 r10
.....
#
#Feature attribute names
#
@0 .smd
@1 .nomenclature
@2 .test_point
@3 .geometry
@4 .pad_usage
.....
#
#Feature attribute text strings
#
&0 9796334
&1 fid_0_0_0
&2 moire
&3 p115_115_115_095
...
```

```
#
#Layer features
#
P -0.198 1.62 16 P 0 3;3=2,4=0
P 0.118 1.62 16 P 0 3;3=25,4=0
L 3.834 -1.16 3.86728 -1.16 2 P 0 ;1,3=0
L 3.86728 -1.16 3.8782 -1.16485 2 P 0 ;1,3=0
....
S P 0
OB -0.013 2.427 I
OS -0.013 2.218
OS -0.263 2.218
OS -0.263 2.427
OS -0.219 2.427
OS -0.219 2.262
OS -0.057 2.262
OS -0.057 2.427
OS -0.013 2.427
OE
SE
....
```

The following figure describes how symbols are treated. The sample feature file has been stripped of all but the relevant sections for this example:



The feature symbol names section:

This section contains the symbols used by features in the file. The format of each line is:

```
$<serial_num> <symbol_name> [<I|M]
```

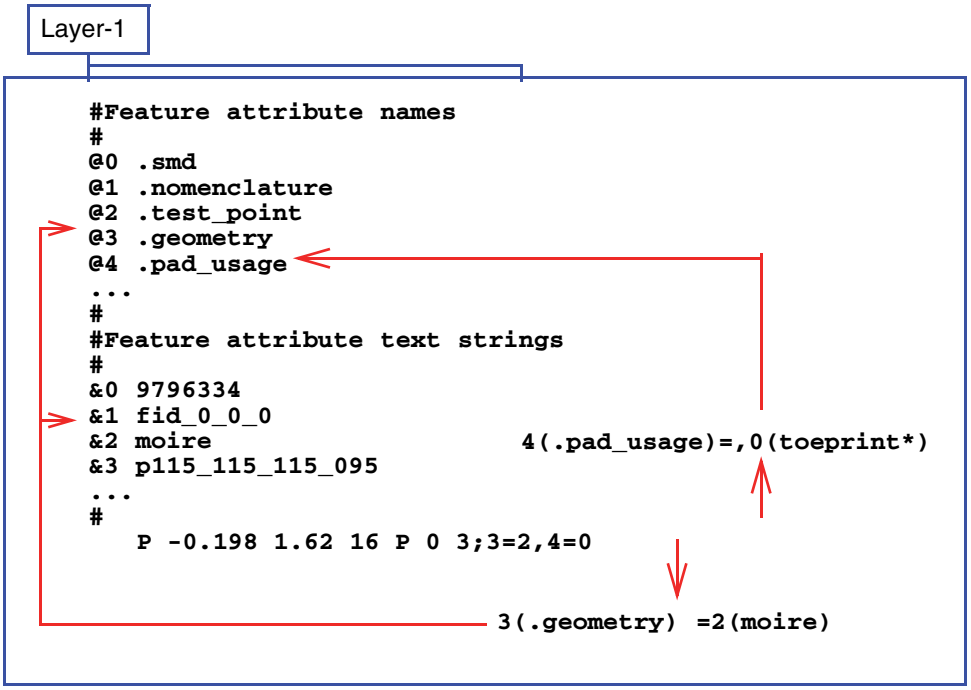
See "Symbol Definitions" on page 202 for symbol naming conventions.

The feature attribute names section:

This section defines the names of attributes used by features in the file. The format of each line is:

```
@<serial_num> <attribute_name>
```

Attribute starting with the dot (.) character are system attributes (some system attributes must be defined for certain processes, such as analysis). Other attributes are user defined attributes.



* When the attribute is of type Option, the reference number points to the attribute options. In the case of **.pad_usage** these are:

```
0 - toeprint;
1 - via;
2 - g_fiducial;
3 - l_fiducial;
4 - tooling_hole.
```

Therefore, 4=0 in the feature record above, means **.pad_usage=toeprint**

The feature attribute text strings section:

This section contains texts which are values of textual feature attributes. Like its predecessors, the reason for this section is to save the repetition of long texts for each feature which uses it. The format of each line is:

```
&<serial_num> <text>
```

The features section:

This is the main section of the features file. It contains all the features in the file. Most features are represented by a single line in the file. Surface features may require multiple lines. The general format of a feature line is:

```
<type> <params> ; <atr>[=<value>],...
```

Where:

<type>	feature type which can be: **L** Line **P** Pad **A** Arc **T** Text **B** Barcode **S** Surface
<params>	A different set for each type. See below
<atr>	An attribute number, referencing an attribute from the feature attribute names section.
<value>	An attribute value which: - Is omitted for boolean attributes - Is a number for integer and float attributes - Is an option number for an option attribute - Is a number referencing the feature attribute text strings section for a textual attribute

The **<params>** field:

For line (L) records:

```
<xs> <ys> <xe> <ye> <sym_num> <polarity> <dcode>
```

xs, ys	start point
ye, ye	end point
sym_num	A serial number of the symbol in the feature symbol names section
polarity	P for positive, N for negative
dcode	gerber dcode number (0 if not defined)

For pad (P) records:

<x> <y> <apt_def> <polarity> <dcode> <orient_def>

x, y	center point
<apt_def>	This value can be expressed in one of two ways: **-1 <sym_num> <resize_factor> | <sym_num>** If the symbol is resized **apt_def** begins with -1 and contains three numbers. Otherwise, it consists of a single number as in V.6.x—a serial number of the symbol in the feature symbol names section. The resize factor is expressed in thousandths of the units being used (mils or microns).
polarity	P for positive, N for negative
dcode	gerber dcode number (0 if not defined)
orient_def New in 7.0	pad orientation. This value is expressed as: **0|1|2|3|4|5|6|7|8<rotation>|9<rotation>** **0 : 0** degrees, no mirror **1 : 90** degrees, no mirror **2 : 180** degrees, no mirror **3 : 270** degrees, no mirror **4 : 0** degrees, mirror in X axis **5 : 90** degrees, mirror in X axis **6 : 180** degrees, mirror in X axis **7 : 270** degrees, mirror in X axis **8 : any angle** rotation, no mirror **9 : any angle** rotation, mirror in X axis If the first number of orientation definition is an integer from **0** through **7**, it is legacy date from before ODB++ V.7.0 and will be handled as in V.6.x. If the first number is **8** or **9**, it is a two number definition, with the following number representing rotation. **Note:** To maintain backward compatibility, values **0**-**7** are read from legacy data, but saved in the new format.

Example of Pad Records in Feature File

Special pad **const_1** at location **x**=1.0, **y**=2.0 positive, with **dcode** 4, is used as an example for different transformations:

```
#
#Feature symbol names
#
$0 const_1

P 1.0 2.0 0 P 4 1                 # rotated by 90
P 1.0 2.0 0 P 4 8 30.0            # rotated by 30
P 1.0 2.0 -1 0 0.02 P 4 1         # rotated by 90 resized by 0.02 mil
P 1.0 2.0 -1 0 0.02 P 4 8 30.0    # rotated by 30 resized by 0.02 mil
```

The same information is displayed in the Info Command Interface for **Data type: FEATURES** and the Info output file, **-t <leayr> -d FEATURES**.

For arc (A) records:

<xs> <ys> <xe> <ye> <xc> <yc> <sym_num> <polarity> <dcode> <cw>

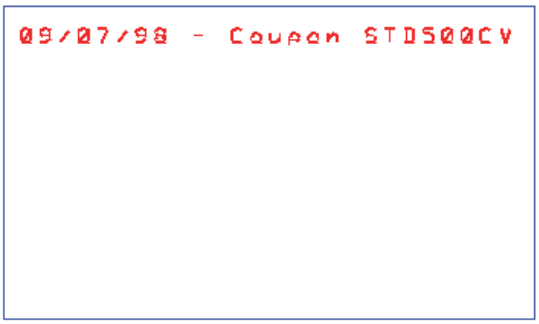
xs, ys	start point
ye, ye	end point
yc, yc	center point
sym_num	A serial number of the symbol in the feature symbol names section
polarity	**P** for positive, **N** for negative
dcode	gerber dcode number (0 if not defined)
cw	**Y** for clockwise, **N** for counter clockwise

For text (T) records:

<x> <y> <font> <polarity> <orient_def> <xsize> <ysize> <width factor> <text> <version>

x, y	text location (bottom left of first character for 0 orientation)
font	font name (Currently must be 'standard')
polarity	**P** for positive, **N** for negative
orient_def New in 7.0	text orientation. This value is expressed as: **0|1|2|3|4|5|6|7|8 <rotation>|9<rotation>** **0 : 0** degrees, no mirror **1 : 90** degrees, no mirror **2 : 180** degrees, no mirror **3 : 270** degrees, no mirror **4 : 0** degrees, mirror in X axis **5 : 90** degrees, mirror in X axis **6 : 180** degrees, mirror in X axis **7 : 270** degrees, mirror in X axis **8 : any angle** rotation, no mirror **9 : any angle** rotation, mirror in X axis If the first number of orientation definition is an integer from **0** through **7**, it is legacy date from before ODB++ V.7.0 and will be handled as in V.6.x. If the first number is **8** or **9**, it is a two number definition, with the following number representing rotation. **Note:** To maintain backward compatibility, values **0**-**7** are read from legacy data, but saved in the new format.
xsize,ysize	Character size
width factor	width of character segment (in units of 12 mils) i.e. 1 = 12 mils, 0.5 = 6 mils
text	text string.
version	text field version values: **0** previous version **1** current version

Example of Text Records in Feature File



The frame on the left shows the text entered in a blank layer in the Graphic Station.
Below is the Feature file created for this layer.

The actual text string contained within single quotes (text records can have embedded space characters).
Note that the Date format can be:
DD/MM/YYYY as of version 4.3

1
current version*

```
#
#Feature attribute names
#
@0 .nomenclature
#
#Layer features
#
T 4.033375 6.377506 standard P 0 0.2 0.2 2.00000 '$$DATE-DDMMYY - Coupon STD500CV' 1;0
```

Polarity=Positive

font=standard

Orientation=0

T
Indicates the record is text.

The X, Y coordinates locating the bottom left of the first character of the text string.

X size, Y size of text string (in this case, 200 mils each).

Width factor=2.00000

* This field is for future use. Currently, when version=0 it indicates the previous version. Old versions of feature files may not have this field at all, in this case it is assumed version=0. The version field does not affect the interpretation of the text data.

For barcode (B) records:

```
<x> <y> <barcode> <font> <polarity> <orient_def> E <w> <h>
<fasc> <cs> <bg> <astr> <astr_pos> <text>
```

See parameters of T (text) records for dynamic values.

in 7.0

x,y	test location (bottom left of first character for 0 orientation
barcode	barcode name (currently must be UPC39)
font	font name (currently must be 'standard')
polarity	**P** for positive, **N** for negative
orient_def	text orientation: same as for T (text) records
E	a constant value (reserved for future use)
w	element width
h	barcode height
fasc	**Y** for full ASCII, **N** for partial ASCII
cs	**Y** for checksum, **N** for no checksum

bg	**Y** for inverted background, **N** for no background
astr	**Y** for an addition of a text string
astr_pos	**T** for adding the string on top, **B** for bottom
Text	text string

For surface (S) records:

A surface is different from other features; it consists of multiple records:

```
S <params> ; <atr>=<value>...
<polygon 1>
<polygon n>
SE
```

The <params> section contains: **<polarity> <dcode>**

polarity - P for positive, N for negative

dcode - gerber dcode number (0 if not defined)

The first line is followed by a list of polygons. Each polygon is a collection of segments (lines without width) and curves (arcs without a width). Polygons must meet the following restrictions:

- Intersection is not allowed between edges of the same polygon.
- Intersection is not allowed between edges of different polygons.
- The polygons must form a closed shape (e.g, a polygon that contains only 2 segments is not valid).
- Holes must be graphically contained inside island polygons. The direction of island must be clockwise and of holes must be counter clockwise.
- The curves must be consistent (the start, end, and center point must construct a legal curve).

If any of the above mentioned violations occurs, the system will not be able to read the file, and will return an error.

The syntax of the polygons description for a surface feature is as follows:

```
OB  <xbs> <ybs> <poly_type>
OS  <x> <y>
OC  <xe> <ye> <xc> <yc> <cw>
OE
```

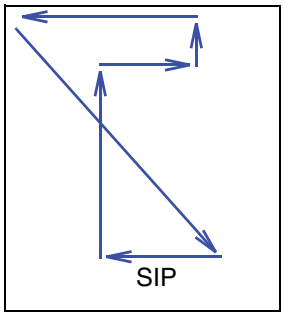
Where:

xbs,ybs	polygon start point
poly_type	**I** for island, **H** for hole
x, y	segment end point (previous polygon point is the start point)
xe, ye	curve end point (previous polygon point is the start point)
xc, yc	curve center point
cw	**Y** for clockwise, **N** for counter clockwise

It is recommended that polygons be represented each as a single island, since a multi-island polygon is electrically disconnected. As a single feature, it is supposed to be connected to a single net.

Self intersecting Polygons

A self intersecting polygon (SIP) is a polygon with two non-consecutive edges (segments or curves) which touch each other. Legal polygons are those whose edges intersect only at endpoints of consecutive edges (see figure on the right).



SIPs are not a good base for mathematical representation. Problematic operations are:

- Resize (enlarge, shrink, change shape)
- Calculation of copper areas (where unambiguous definition of the copper location is essential)

Changes Required for GenFlex 6.4

The changes in the **features** file are caused by implementation of the following capabilities:

- Support for any angle rotation of texts
- Support of non-Latin text features
- Support for additional fonts

Feature Symbol Names section

This section format has not changed. It should be noted that there are a new reserved name to the symbol that is:

- Rotated or SHX or non-Latin text (text parameters are stored in attributes see section 6.2.2)
- Rotated standard symbol (rotation angle (CW) is a part of the symbol name)
- Barcodes CODE-128b, CODE-128c and ECC-200 (barcode parameters are stored in attributes see section 6.2.3)

For example:

```
#
#Feature symbol names
#
$0 text
$1 text+1
$2 text+2
$3 s100_45
$4 rect20x10_30
$5 barcode
$6 barcode+1
```

Font name in text description record

In addition to ODB++ fonts, described in document 0202, there supported AutoCad fonts (.shx). Font files are stored in the **<job_name>/fonts_ex/shx/** directory.

Existing format:

<x> <y> <font_name> <polarity> <orient> <xsize> <ysize> <width factor> <text>

Where:

<font_name> - Name of one of font files existing in the directory **fonts** & **fonts_ex/shx**

Text specification

Since **<text>** field can contain non-Latin text, and there exist several types of supported fonts, the additional considerations are done for text storage.

The text is stored as a multi-byte sequence in encoding that matches the font file. This makes easier to recreate a graphics representation of a text string.

For example, for **standard** file (or other existing ODB++ font file format) the encoding is ASCII, which is actually no change in comparison to the past.

AutoCAD fonts (.shx) have Windows encoding which is country dependent. For example, Japanese encoding is SJIS, and Chinese encoding is Big5. Text in the feature record is stored in the corresponding encoding. If this text should be further presented in (non-graphic) GUI with other encoding, the information of the original font encoding can be found separately, with the font file description.

Text symbol to represent rotated & SHX & non-Latin text

To represent a text a special symbol with predefined name **text+nn** is used.

A symbol includes a text built by simple lines to back compatibility. In addition it should include next attributes:

.text	= 12345	// Text string
.nomenclature_type	= standard	// Font name
.text_x_size	= 0.200000	
.text_y_size	= 0.200000	
.text_line_width	= 23.622047	
.text_rotation	= 30.000000	

The attributes saves text parameters used if the text should be modified.

Text symbol to represent barcode

To represent a barcode a special symbol with predefined name **barcode+nn** is used.

A symbol includes a barcode built by simple lines to back compatibility. In addition it should include next attributes:

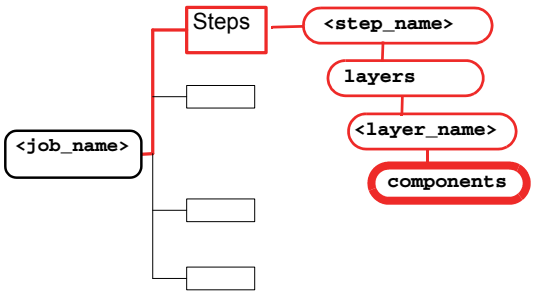
.text	= 12345	// Text string
.barcode_type	= ecc-200	// Barcode type (code128b/code128c/ecc-200)
.barcode_matrix	= 10x10	// Barcode matrix (for type=ecc200)
.barcode_bg	= 1	// Barcode background(1-yes;0-no)

```
.barcode_astr          = 1          // Barcode with text(1-yes;0-no)
.barcode_astr_pos      = 0          // Barcode text position(0-top;1-bottom)
.text_x_size           = 0.200000   // Barcode height
```

components

Type:	Line record Text
Compression:	Yes
Sum file:	No
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/components**

Each layer of type component has a unique name: **comp_+_top** or **comp_+_bot**. There are a maximum of two such layers in each job. Each layer has a components file which contains the information about actual components placed on the layer. The components file contains references to the **<step_name>/eda/data** file described earlier.



Example

```
#
#Component attribute names
#
@0 .comp_ign_spacing
@1 .no_tp_under
@2 .no_text_under
@3 .thvpad_required
@4 .comp_type
@5 .comp_height
@6 .comp_weight
# CMP 0
CMP 13 -0.04 1.22 270.0 N B70 2248827-0001
;1,2,4=11,5=0.100000,6=0.035273
PRP REFLOC 'IN,0.2,-0.225,270,CC,0.035,0.035,0.009,std,1'
TOP 0 -0.198 1.62 270.0 N 223 0 B70-1
TOP 1 0.118 1.62 270.0 N 223 1 B70-2
TOP 2 -0.04 1.22 270.0 N 466 0 B70-3
#
....
```

The components file may have a header, listing names of attributes used by the components in the file and possible textual values. This header is similar to the header of the features file (records starting with the @ and & character) and is described there.

Following the attribute header, components are listed in order, using 3 types of records:

CMP	Starts a component
PRP	Property of a component
TOP	Toeprint of a component

Following is the format and description of each record.

CMP - Component Record

This record contains a definition of a component.

Each **CMP** line is followed by 0 or more property (PRP) records and 0 or more TOP records.

```
CMP <pkg_ref> <x> <y> <rot> <mirror> <comp_name> <part_name> ;
<attributes>
```

Where

<pkg_ref>	The number of the package in the eda/data file
<x>,<y>	The board location of the component in inches
<rot>	The rotation of the component, in degrees, clockwise.
<mirror>	N for not mirrored, M for mirrored
<comp_name>	component name (reference designator)
<part_name>	part identification
<attributes>	This data is the same as for feature attributes (in the features file). It consists of comma separated list of values. Each can be: **n** indicating that (boolean) attribute **n** is set **n=m** indicating that option attribute **n** has value **m** **n=i** indicating that integer attribute **n** has value **i** **n=f** indicating that floating attribute **n** has value **f** **n=s** indicating that text attribute **n** has header value **s**
Note: **n** must match a @ record in the attribute header **s** must match a & record in the attribute header.	

TOP - Toeprint Record

This record contains a definition of a toeprint of a component.

```
TOP <pin_num> <x> <y> <rot> <mirror> <net_num> <subnet_num>
<toeprint_name>
```

Where:

<pin_num>	The pin number inside the package of the component
<x>,<y>	The board location of the pin in inches
<rot>	The rotation of the component, in degrees, clockwise.
<mirror>	N for not mirrored, M for mirrored
<net_num>	Number of net in the eda/data file*
<subnet_num>	Number of subnet within referenced net
<toeprint_name>	Name of the toeprint

* The **net_num** used in the TOP record corresponds to the sequence of the Net records in the **eda/data** file. The first Net record is **net_num 0**, the second is **net_num 1** and so on.

PRP - Property Record

This record represents a property of the component.

A property consists of a name, a string value and 0 or more floating numbers.

```
PRP <name> '<value>' n1 n2 ...
```

Where:

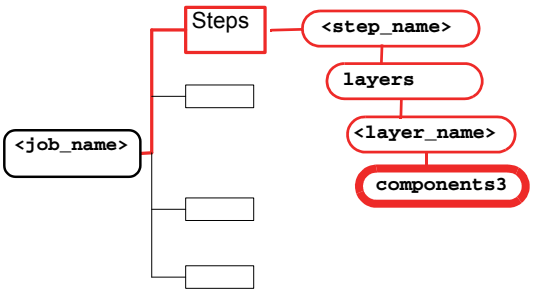
<name>	The name of the property
<value>	The string of the property (between quotes)
n1,n2,...	The floating numbers to be kept in the property

components3

Type:	Line Record Text
Compression:	yes
Sum file:	no
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/components3**

The **components** file describes the original EDA data for a component, while the **components3** file presents the data after processing with Assembly Merge (Bom Merge, Library Merge and Board Merge).

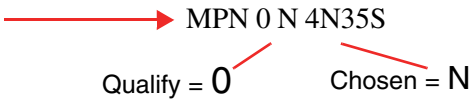
Note See preceding section for further information on the example below.



Example

```
#
#Component attribute names
#
@0 .comp_polarity
@1 .comp_height

# CMP 0
CMP 0 27.235992 6.19674 270.0 N W3_30B *
#
# BOM DATA
CPN 070-000-016
PKG
IPN
VPL_VND
```

```
VPL_MPN
VND CAL GREG
MPN 0 Y 4N35S
VPL_VND MOTOROLA
VPL_MPN
VND MOTOROLA
MPN 0 N 4N35S
VPL_VND SMNS-MTST
VPL_MPN
VND SIEMENS
MPN 0 N 4N35-X017
VPL_VND TI
VPL_MPN 4N35DCJ
VND TI
MPN 0 Y 4N35DCJ
# CMP 1
```



The component BOM DATA section contains BOM information on component

Parameter	Description
CPN	Customer part number
PKG	Package name
IPN	Internal part number
DSC	Up to 5 descriptions
VPL_VND	Manufacturer from the VPL corresponding to original vendor (as determined in BOM Validation)
VPL_MPN	MPN from the VPL database corresponding to original MPN (as determined in BOM Validation)
VND	Manufacturer (vendor) name
MPN	Manufacturer part number
Qualify	Whether the part (vendor+mpn) is qualified for production: **-1** - not qualified **0** - unknown **1** - qualified
Chosen	If this part is chosen from among the alternate parts for the CPN, only one can be chosen. **y** - yes, **n** - no

The MPN line contains the following parameters separated by spaces:

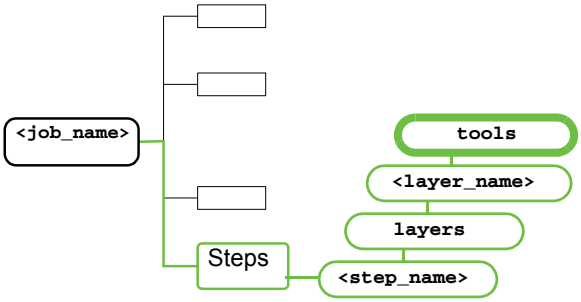
qualify chosen MPN

The section: **VPL_VND + VPL_MPN + VND + MPN** repeats for all the alternate parts of that CPN.

S tools (Drill Tools)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/tools**

This file contains the tools table of a drill layer, initially created during input and further enhanced by the Drill Tools Manager.



Example

```
THICKNESS=0.0625
USER_PARAMS=method25

TOOLS {
   NUM=1
   TYPE=VIA
   TYPE2=STANDARD
   MIN_TOL=0
   MAX_TOL=0
   BIT=
   FINISH_SIZE=11.5
   DRILL_SIZE=13.5
}
TOOLS {
   NUM=2
   TYPE=PLATED
   TYPE2=STANDARD
   MIN_TOL=0
   MAX_TOL=0
   BIT=
   FINISH_SIZE=15
   DRILL_SIZE=19
}
```

Note See "Units of Measurement" on page 19.

The file contains 2 global parameters and a **TOOLS** array.

The global parameters are:

THICKNESS	board thickness (mils)
USER_PARAMS	free text that is used by the hook **drill_size** when converting finished hole sizes to drilled hole sizes

The fields of the **TOOLS** array structure are:

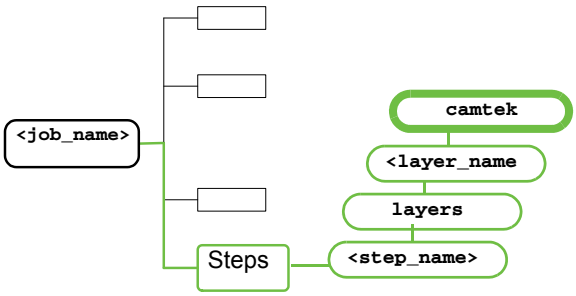
NUM	tool number
TYPE	one of **PLATED, NON_PLATED, VIA**

TYPE2	one of **STANDARD**, **PHOTO**, **LASER**, **PRESS_FIT** (default: **STANDARD**) used in the "**TOOLS**" section of the "**tools**" file.
MIN_TOL, MAX_TOL	allowed tolerances (mils)
BIT	drill bit string
FINISH_SIZE	finished drill size (mils)
DRILL_SIZE	calculated drill size (mils)

camtek

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job name>/steps/<step name>/layers/<layer name>/camtek**

The file contains parameters of a Camtek set, describing parameters to be used when testing the layer for this entity.



ANGLE	Alignment of panel: rotation value
MIRROR	Alignment of panel: mirroring value
X_SCALE	Alignment of panel: X axis scale value
Y_SCALEL	Alignment of panel: Y axis scale value
POLARITY	Polarity of panel
DRILLS	Whether drill holes must be considered?
ETCH	Value of etch factor
RESOLUTION	Mania pixel size
MIN_LINE	Minimal line value
MIN_SPACE	Minimal space value
REG_DEFINED	True if registration pins were defined
REG_X1	X coordinate of first registration pin
REG_Y1	Y coordinate of first registration pin
REG_X2	X coord of second registration pin
REG_Y2	Y coord of second registration pin
CALIB	Calibration value
THICKNESS	Board thickness

TOLERANCE	Tolerance
LAMINATION	Lamination type
MACHINE	Machine type
REG_METHOD	Type of registration
SCAN_AREA	Dimension of area-to-test
EXCLUSION	Dimension of area not to test (rectangle, circle or polygon)

Example

```
ANGLE=0
MIRROR=NO
X_SCALE=1
Y_SCALE=1
POLARITY=POSITIVE
DRILLS=NO
ETCH=0
RESOLUTION=1.25
MIN_LINE=0
MIN_SPACE=0
REG_DEFINED=YES
REG_X1=1.552809350393701
REG_Y1=0.2045889763779528
REG_X2=1.475844488188976
REG_Y2=-0.1904967519685039
CALIB=C1
THICKNESS=0
TOLERANCE=0
LAMINATION=SHEET
MACHINE=PANEL
REG_METHOD=1

SCAN_AREA {
     X1=-0.9562419291338583
     Y1=-0.9396204724409449
     X2=1.753758070866142
     Y2=1.065379527559055
 }
EXCLUSION {
     X1=-0.4072266732283464
     Y1=0.2558988188976378
     X2=-0.114760531496063
     Y2=0.5381029527559055
 }
EXCLUSION {
     X1=0.5471363188976378
     Y1=0.3636495078740157
     X2=0.7626376968503937
     Y2=0.4970551181102362
 }

 EXCLUSION_C {
     X=0.8190785433070866
```

```
        Y=0.3482564960629921
        R=0.3574053149606299
    }

    EXCLUSION_P {

        POINT {
            X=0.7010658464566929
            Y=-0.2725925196850394
        }
        POINT {
            X=0.7677687007874016
            Y=-0.3700812007874016
        }
        POINT {
            X=0.3624209645669291
            Y=-0.3752122047244095
        }
        POINT {
            X=0.3418969488188977
            Y=-0.2366755905511811
        }
        POINT {
            X=0.3675518700787402
            Y=-0.1084009842519685
        }
    }
```

Translation of AOI-SET Fields into Camtek Output

```
ANGLE=90              -> to calculate transformation from panel coords to AOI table coords
MIRROR=YES            -> to calculate transformation from panel coords to AOI table coords
X_SCALE=1.01          -> inf.dat, Layer-info, Xstretch
Y_SCALE=1.02          -> inf.dat, Layer-info, Ystretch
POLARITY=NEGATIVE     -> to allow AOI machine to identify copper
DRILLS=NO             -> whether to create drill data (drill01.dat)
ETCH=1.1              -> inf.dat, Layer-info, Etch
RESOLUTION=0.5        -> inf.dat, Layer-info, PixSize
MIN_LINE=0            -> none
MIN_SPACE=0           -> none
REG_DEFINED=YES       -> inf.dat, Layer-info, Ref_pins
REG_X1=1              -> inf.dat, Layer-info, Ref_pins
REG_Y1=9              -> inf.dat, Layer-info, Ref_pins
REG_X2=9              -> inf.dat, Layer-info, Ref_pins
REG_Y2=9              -> inf.dat, Layer-info, Ref_pins
CALIB=A0              -> inf.dat, Layer-info, Calib
THICKNESS=2.2         -> inf.dat, Layer-info, Thick
TOLERANCE=0           -> none
LAMINATION=FOIL       -> out of lamination, layer number and number of layers:
                         inf.dat, Layer-info, Layer_view
                         inf.dat, Layer-info, Layer_pair
MACHINE=ARTWORK       -> none
REG_METHOD=1          -> inf.dat, Layer-info, Align_method

SCAN_AREA {           -> to calculate number of frames and overlap values
    X1=0.4914508858267717
```

```
    Y1=0.5613854330708662
    X2=9.581646948818898
    Y2=8.516830905511812
}

EXCLUSION {              -> don't-inspect areas, as listed in mask01.dat
    X1=0
    Y1=0
    X2=10
    Y2=0.5648308070866142
}

EXCLUSION {              -> yes, the hdr file may contain multiple EXCLUSION arrays
    X1=0
    Y1=0.5648308070866142
    X2=0.4946468503937008
    Y2=1.498834744094488
}
```

attrlist (Attribute List)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/camtek/attrlist**

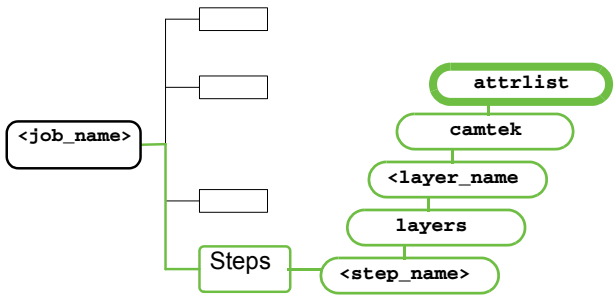
The user attributes are defined by Camtek, and a user attributes ASCII file is normally supplied with the Camtek AOI system. In the inf.dat output file:

<output path>/<job name>/<layer number>/<AOI set name>/inf.dat

the user attribute values appear (as set) in the [Learn Type Definitions] section.

Example

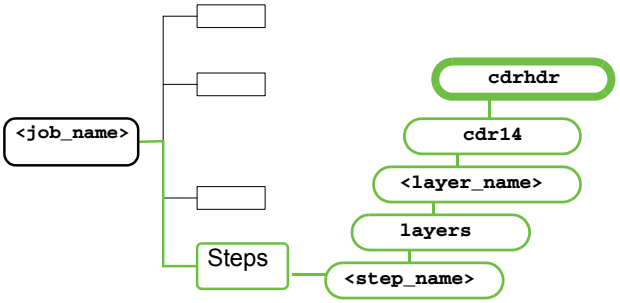
```
.drc_min_space = 5
.drc_min_width = 7
.drc_add_rad = 2
```



S cdrhdr (CDR14 Header)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/cdrhdr`

This file contains the values for CDR14 parameters of the layer.



Example

```
NOM_SPACE=0.02
MIN_SPACE=0.015
NOM_LINE=0.02
MIN_LINE=0.015
MULTI_LINE=YES
MULTI_LINES {
   MIN_WIDTH=0.15
   MAX_WIDTH=0.17
}
MULTI_LINES {
   MIN_WIDTH=0.18
   MAX_WIDTH=0.20
}
...

WORKING_TABLE=24X24
PADS {
    SYMBOL_NAME=r100
}
PADS {
    SYMBOL_NAME=rect150x120
}
...
STAGES {
    STAGE_NAME=PHOTO
    ETCH_SET=YES
    ETCH=0.005
CLASSES {
        CLASS_NAME=photo1
    }
    ...

    TOOLSET_NUMBER=-1
MANUAL_ALIGNMENT {
        IS_USED=YES
OFFSET {
            X=20
            Y=10
        }
        MIRROR=X
```

```
        ANGLE=270
        POLARITY=POSITIVE
    }
}
...

DRILLED_STAGE=-1
SCALE {
    X=1
    Y=1
}
SCALE_ORIGIN {
    X=0
    Y=0
}
```

Description

NOM_SPACE	Nominal Spacing. Valid range is 0.0005..0.128 (inch). Null/default value is 0.0.
MIN_SPACE	Minimal Spacing where **MIN_SPACE <= NOM_SPACE**. Valid range is 0.0005..0.128 (inch) Null/default value is 0.0.
NOM_LINE	Nominal Line Width. Valid range is 0.0005..0.128 (inch) Null/default value is 0.0.
MIN_LINE	Minimal Line Width, where **MIN_LINE <= NOM_LINE**. Valid range is 0.0005..0.128 (inch) Null/default value is 0.0.
MULTI_LINE	Yes/No. Yes - use **MULTI_LINES** array instead of **MIN_LINE**
MULTI_LINES	Array of a maximum of 4 elements defining multiple line width ranges.
WORKING_TABLE	Name of the working table matching the template **<w>X<h>** where **w** is table width **h** is table height. Should be defined in the **cdr14.ini** file.
PADS	Array of a maximum 8 elements defining pads symbol names
HOLES	Array of a maximum of 8 elements defining holes symbol names.
CLEARANCES	Array of a maximum of 8 elements defining clearance symbol names.
STAGES	Array of a maximum of 10 elements defining working stages parameters.
DRILLED_STAGE	Index of the stage in STAGES array which is defined as a drill stage. Null/default value is **-1**

SCALE	X and Y scale factors applied on output. Valid range is 0.001..9.99 (0.1..999%). Null/default value is 1.0.
SCALE_ORIGIN	X and Y scale origin coordinates. Valid range is unlimited. Null/default value is 0.0.
PANELIZATION	**PANEL_DEFINED**: use the genesis automatic panelization. **USER_DEFINED**: Use the panelization supplied by user.
GENESIS_VERSION	Version of the Genesis software which created the **cdr14** set of the form <**major**>.<**minor**><**patch**>.
MARGINS_SET	0. Field not in use.
X_MARGIN	0. Field not in use.
Y-MARGIN	0. Field not in use.

MULTI_LINES Array Structure:

MIN_LINE	Minimal Line Width
MAX_LINE	Maximal Line Width where **MIN_LINE <= MAX_LINE** and **MIN_LINE <= NOM_LINE**. Valid range is 0.0005..0.128 (inch). Null/default value is 0.0

* **MULTI_LINES** pairs should be defined in the order of **MIN_WIDTH** increasing.

PADS/HOLES/CLEARANCES Arrays Structure

SYMBOL_NAME	Feature symbol name.

STAGES Array Structure

STAGE_NAME	Working stage name. Should be defined in the **cdr14.ini** file.
ETCH_SET	Yes/No. Yes - use the **ETCH** value. No - use the default **ETCH** value defined for the stage in the **cdr14.ini** file instead.
ETCH	Etch value. Valid range is -255.0..255.0 (inch). Null/default value is 0.0.
CLASSES	Class names for the stage. Array for a maximum of 5 elements.
DRILL_LAYER	Name of the drill layer. Relevant only for stage defined as drill stage.
TOOLSET_NUMBER	Toolset number used for the stage alignment, should be defined in the **cdr14.ini** file. Null/default value is -1.

MANUAL_ALIGNMENT	Stage alignment used if no toolset defined.
DRILL_LAYERS	Names of drill layers. In case of multiple drill layers, names are separated by semi-colons (**;**)

* Either **TOOLSET_NUMBER** or **MANUAL_ALIGNMENT** should be defined for each stage.

MANUAL_ALIGNMENT Structure

IS_USED	Yes/No. Yes - use manual alignment instead of toolset.
OFFSET	X and Y alignment offsets.
MIRROR	None/X/Y/Both.
ANGLE	0/90/180/270 measured in degrees CW (clockwise).
POLARITY	Positive/Negative.

Translation of CDR-SET Fields into AOIProg Commands

```
CDR-SET Field                          AOIProg Translation
-------------                          -------------------
NOM_SPACE=0.008                      | SPACE = 8.000
MIN_SPACE=0.007                      | MSPACE = 0.875 (= MIN_SPACE/NOM_SPACE.
                                     |                  if MIN_SPACE not set,
                                     |                  cdr14_min_spacing_factor cfg value
                                     |                  is taken)

NOM_LINE=0.008                       | LINE = 8.000
MIN_LINE=0.007                       | MLINE = 0.875 (= MIN_LINE/NOM_LINE.
                                     |                  if MIN_LINE not set,
                                     |                  cdr14_min_line_factor cfg value
                                     |                  is taken)

MULTI_LINE=NO                        | No direct translation to AOIProg command.

WORKING_TABLE=24x24                  | Appears as a comment in AOIProg file.

PADS {                               | PAD = 1:100.000, 1:200.000
    SYMBOL_NAME=r100
}
PADS {
    SYMBOL_NAME=r200
}


HOLES {                                | HOLE = 1:50.000, 1:75.000
    SYMBOL_NAME=r50
}
HOLES {
    SYMBOL_NAME=r75
}



STAGES {
    STAGE_NAME=COPPER
    ETCH_SET=YES
    ETCH=0.0005                        | ETCH\COPPER = 0.500
```

```
    CLASSES {
        CLASS_NAME=c_sig_cop          | CLASS \COPPER = c_sig_cop:drl
    }
    DRILL_LAYER=drl

    TOOLSET_NUMBER=99             | If toolset alignment, translated to
                                  | TOOL \COPPER = 99, 10000.00,10000.0, H, RCCW270
                                  | (where 10000.00,10000.0, H, RCCW270 are the
                                  | Toolset parameters)


    MANUAL_ALIGNMENT {
        IS_USED=YES               | If toolset alignment, IS_USED set to NO

        OFFSET {
            X=10
            Y=10
        }
        MIRROR=Y                  | Mirror around Y axis - Horizontal
        ANGLE=90                  | Rotation 90 deg CW (= 270 deg CCW)
        POLARITY=POSITIVE         | If Negative, NEG is added to the TT command.
    }                             | The MANUAL_ALIGNMENT structure,
                                  | together with the layer's alignment targets,
                                  | determine the AOIProg's CT and TT commands:
                                  | CT \COPPER = 9000.000:9000.000:1:133.000,
1000.000:9000.000:1:133.000
                                  | TT \COPPER = 1000.000:1000.000:1,
1000.000:9000.000:1, H, RCCW270

    DRILL_LAYERS=drl              | drill layer name is added to the CLASS command:
                                  | CLASS \COPPER = c_sig_cop:drl
                                  | In case of more than one drill layer, all defined
                                  | drill layers are merged into a single temporary  layer
                                  | named mdxxxx
}

DRILLED_STAGE=0                   | No direct translation to AOIProg command

SCALE {                           | No direct translation to AOIProg command
    X=1
    Y=1
}


SCALE_ORIGIN {                    | No direct translation to AOIProg command
    X=0
    Y=0
}

PANELIZATION=PANEL_DEFINED
GENESIS_VERSION=08.01DV
MARGINS_SET=0
X_MARGIN=0
Y_MARGIN=0
INSPECTED_STEPS=pcb               | The steps that are translated into
                                  | PCB/RPCB AOIProg commands.
```

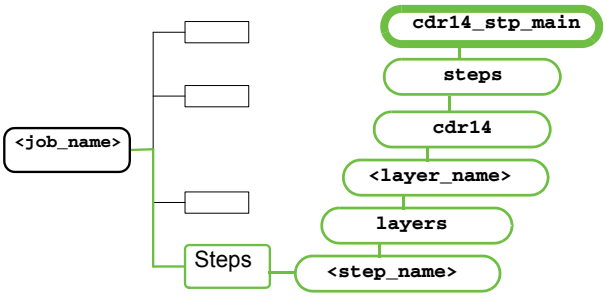
S cdr14_stp_main (CDR14 Main Step)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/cdr14_stp_main
```

This step holds the cdr14 graphic data. Inspection areas and exclusion zones are placed in the **lyr_area** layer and alignment targets are placed in the **lyr_targ** layer.

See "stephdr (Step Header)" on page 71 for step structure and "Job>steps>layers Entity" on page 103 for layer structure.

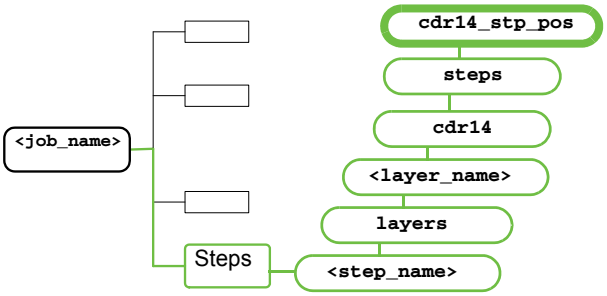
This step contains the steps described below, optionally step&repeated.



S crd14_stp_pos (CDR14 Positive Step)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/cdr14_stp_pos
```

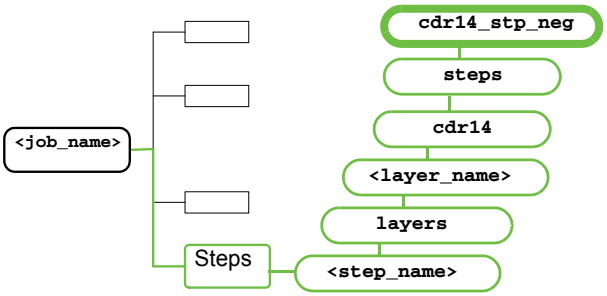
This step contains the inspection areas represented as positive features (a rectangular surface) in its **lyr_area** layer. This step can be placed by a maximum of one step&repeat command.



S crd14_stp_neg (CDR14 Negative Step)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/cdr14_stp_neg
```

This step contains the non-step&repeated exclusion zones as a negative feature (a rectangular / polygonal surface or round pad) in its **lyr_area** layer.

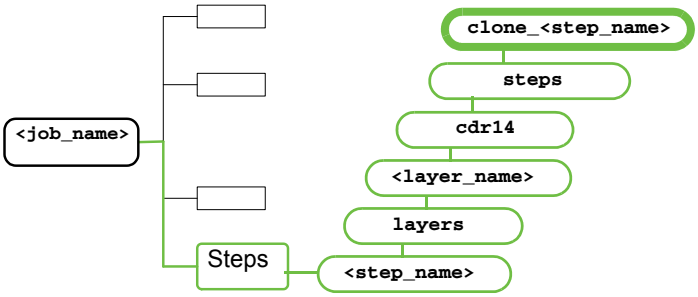


clone_<step_name> (S&R Exclusion Zones)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/clone_<step_name>
```

These steps contain the step&repeated exclusion zones as negative features (rectangular/polygonal surface or round pad) in its **lyr_area** layer.

These steps are step&repeated with respect to the corresponding step **<step_name>**.



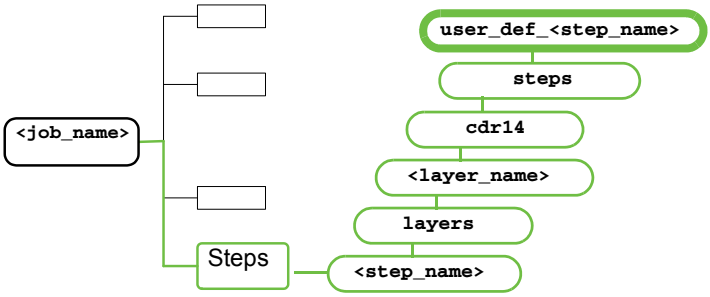
user_def_<step_name> (Steps in AOI)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/user_def_<step_name>
```

These steps are created only in case AOI panelization was defined using the CDR interface (see "cdrsr (AOI Panelization)" on page 98).

These steps replace the **clone_<step_name>** steps.

<step_name> is the step identification as supplied by the AOI panelization.



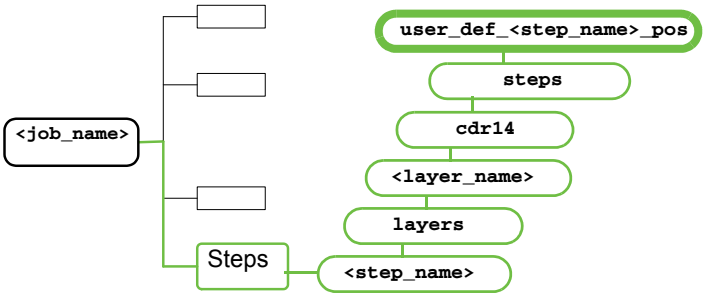
user_def_<step_name>_pos (Steps - AOI)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr14/steps/user_def_<step_name>_pos
```

These steps are created only in case AOI panelization was defined using the CDR interface (see "cdrsr (AOI Panelization)" on page 98).

These steps replace the **clone_<step_name>_pos** steps.

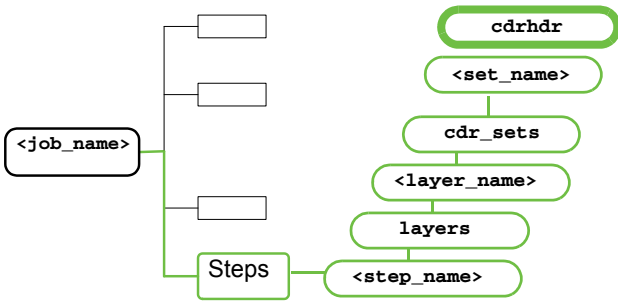
<step_name> is the step identification as supplied in the AOI panelization.



S <set_name>/cdrhdr (CDR Header)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/cdrhdr**

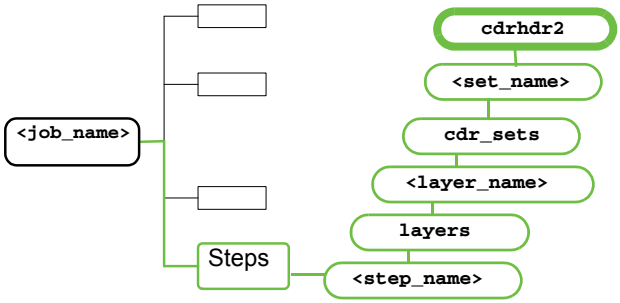
This entity is as described in "cdrhdr (CDR14 Header)" on page 128.



S <set_name>cdrhdr2 (CDR14 Header - Additional)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	`<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/cdrhdr2`

This file contains values for CDR parameters of a layer, in addition to those in "<set_name>/cdrhdr (CDR Header)" on page 135.



Example:

```
LAYER_TYPE=IL
PATTERN_TYPE=POWER_GR
OUND
POLARITY=NEGATIVE

FEATURES {

    FTRS_YN {
        VALUE_STATUS=AUTO_SETUP
        PADS=NO
        LINES=NO
        CLEARANCES=YES
        NFPS=NO
        THERMALS=YES
        SMDS=NO
        THROUGH_HOLES=NO
        MICRO_VIAS=NO
        BLIND_VIAS=NO
        CROSS_HATCH=NO
    }
    NOM_NFP_SPACING=0
    MIN_NFP_SPACING=0
    NOM_AR=0
    MIN_AR=0
    NOM_LASER_DRL=0
    MIN_LASER_DRL=0
}


HISTOGRAMS {

    HIST {
        H_TYPE=THERMAL
        H_SHAPE_TYPE=MERGED
        SORT_TYPE=WIDTH
        RESOLUTION=0.00025
        DRILL_LYR_HIST=
```

```
        H_CALCULATED=YES
        DATA {
            SIZE=0.09
            COUNT=206
        }
        DATA {
            SIZE=0.095
            COUNT=10
        }
        DATA {
            SIZE=0.1
            COUNT=120
        }
    }
    HIST {
        H_TYPE=CLEARANCE
        H_SHAPE_TYPE=MERGED
        SORT_TYPE=WIDTH
        RESOLUTION=0.00025
        DRILL_LYR_HIST=
        H_CALCULATED=YES

        DATA {
            SIZE=0.04
            COUNT=11942
        }
        DATA {
            SIZE=0.06
            COUNT=1002
        }
    }
}

N_MACHINES=1

MACHINE {
    MACHINE=Vision
    RESOLUTION=0
    RULE_FILE=PGIL.rul
    DATABASE=ALPHA
}


STAGES {
    STAGE_NAME=BARE_COPPER
    PRIORITY=5
    COPPER_WT=0
    MATERIAL=
    PANEL_THICK=0
    N_MACHINES=1

    MACHINE {
        MACHINE=Vision
```

```
            TABLE=standard
            PIN_SET=
            REGISTRATION=PINS
            PIN1=FFC
            PIN2=F4
        }
    }
```

Description

Parameter	Description
LAYER_TYPE	Layer type: OL (outer layer), IL (inner layer), or ILWH (inner layer with holes). Null/default value is IL.
PATTERN_TYPE	Pattern type: SIGNAL, POWER_GROUND, MIXED, LASER_DRILL or PHOTO_VIA. Null/default value is SIGNAL.
POLARITY	Polarity of layer: POSITIVE or NEGATIVE. Null/default value is POSITIVE.
FEATURES	Array for defining the layer's characteristics: features types and nominal values.
HISTOGRAMS	Array for defining feature histograms.
N_MACHINES	The CDR set may contain setup for more than one AOI machine type.
MACHINE	Array for defining machine specific parameters.
STAGES	Array of a maximum of 10 elements defining parameters for working stages, in addition to parameters defined in STAGES array in **cdrhdr** file.

FEATURES Array Structure

Parameter	Description
FTRS_YN	Array for defining which feature types exist in the layer.
NOM_NFP_SPACING	Nominal NFP spacing. Valid range is 0.0005**...**0.128 (inch). Null/default value is 0.0.
MIN_NFP_SPACING	Minimal NFP Spacing where MIN_NFP_SPACE <= NOM_NFP_SPACE. Valid range is 0.0005**...**0.128 (inch). Null/default value is 0.0.
NOM_AR	Nominal Annular Ring. Valid range is 0.0005**...**0.128 (inch). Null/default value is 0.0.
MIN_AR	Minimal Annular Ring where MIN_AR <= NOM_AR. Valid range is 0.0005..0.128 (inch). Null/default value is 0.0.
NOM_LASER_DRL	Nominal Laser drill. Valid range is 0.0005**...**0.128 (inch). Null/default value is 0.0.
MIN_LASER_DRL	Minimal Laser drill where: MIN_LASER_DRL <=NOM_LASER_DRL. Valid range is 0.0005**...**0.128 (inch). Null/default value is 0.0.

FTRS_YN Array Structure

Parameter	Description
VALUE_STATUS	Source of feature existence list. Options: NOT_SET, MANUAL_SET, USER_CONFIG or AUTO_SETUP. Null/default value is NOT_SET.
PADS	Does the layer contain pads? Options: YES, NO or UNKOWN. Null/default value is NO.
LINES	Does the layer contain lines? Options: YES, NO or UNKOWN. Null/default value is NO.
CLEARANCES	Does the layer contain clearances? Options: YES, NO or UNKOWN. Null/default value is NO.
NFPS	Does the layer contain NFPs? Options: YES, NO or UNKOWN. Null/default value is NO.
THERMALS	Does the layer contain thermal pads? Options: YES, NO or UNKOWN. Null/default value is NO.
SMDS	Does the layer contain SMD pads? Options: YES, NO or UNKOWN. Null/default value is NO.
THROUGH_HOLES	Does the layer contain thru-holes? Options: YES, NO or UNKOWN. Null/default value is NO.
MICRO_VIAS	Does the layer contain micro vias (laser or photo)? Options: YES, NO or UNKOWN. Null/default value is NO.
BLIND_VIAS	Does the layer contain blind vias? Options: YES, NO or UNKOWN. Null/default value is NO.
CROSS_HATCH	Does the layer contain cross hatches? Options: YES, NO or UNKOWN. Null/default value is NO.

Histograms Array Structure

Parameter	Description
HIST	An array of elements defining feature histograms. Elements may be repeated for every feature type.

HIST Array Structure

Parameter	Description
H_TYPE	Histogram type. Options: PAD, LINE, SMD, NFP, THERMAL, CLEARANCE or DRILL_LAYER.
H_SHAPE	Histogram shape. Separate histograms may be created for features of different shapes. Options: MERGED, ROUND, SQUARE or RECT. Default value is MERGED, meaning that all features of the same type are counted in a single histogram regardless of their geometrical shape.

Parameter	Description
SORT_TYPE	By which dimension are features sorted into histogram rows? Options: WIDTH, HEIGHT or WIDTH_AND_HEIGHT. Default value = WIDTH.
RESOLUTION	Width of histogram row (in inches), i.e the delta between minimum and maximum features sizes which are inserted to the same histogram row. Resolution must be > 0.0 and <= 1.0.
DRILL_LYR_HIST	If histogram of type DRILL_LAYER, this entry specifies the drill layer's name. A separate histogram is created for each drill layer. Histogram rows are for drill hole sizes.
H_CALCULATED	Yes or NO. Histograms may not be calculated if the layer is too heavy (feature-wise, according to user-defined criteria).
DATA	An array of elements defining histogram rows.

DATA Array Structure

Parameter	Description
SIZE	Size of features counted in a histogram row (in inches).
COUNT	Number of features counted in a histogram row.

MACHINE Array Structure

Parameter	Description
MACHINE	Name of AOI machine. Options: Vision.
RESOLUTION	Inspection resolution. Null/default value is 0.0.
RULE_FILE	Name of rule file to be used on AOI Manager to complete pre-setup generated by CDR.
DATABASE	Name of machine's database, to which learn results should be saved.

STAGES Array structure

Parameter	Description
STAGE_NAME	Working stage name. Should be one of the stage names supported by AOI Manager: BARE_COPPER, BRWN_BLCK_OXIDE, TIN_LEAD_BEFORE, TIN_LEAD_AFTER, DBL_TRET_COPPER, DIAZO, SILVER_HALIDE, PHOTORESIST, PHOTOVIA or LASER_DRILL.
PRIORITY	Processing priority by AOI Manager. Integer number between 1 (highest priority) to 10 (lowest).
COPPER_WT	Copper weight (in units as entered).
MATERIAL	Panel material.
PANEL_THICK	Panel thickness (in inches).

Parameter	Description
N_MACHINE	The stage entry may contain parameters for more than one AOI machine type.
MACHINE	Array for defining machine specific parameters for current work stage.

MACHINE Array Structure

Parameter	Description
MACHINE	Name of AOI machine. Options: Vision.
TABLE	Name of AOI machine table. Options are according to machine type. Options for Vision machine are standard, wide, large or extra large.
PIN_SET	Name of additional pin set.
REGISTRATION	Type or machine registration. Options: PINS or PINLESS.
PIN1	Name of pin in the machine's table.
PIN2	Name of pin in the machine's table.

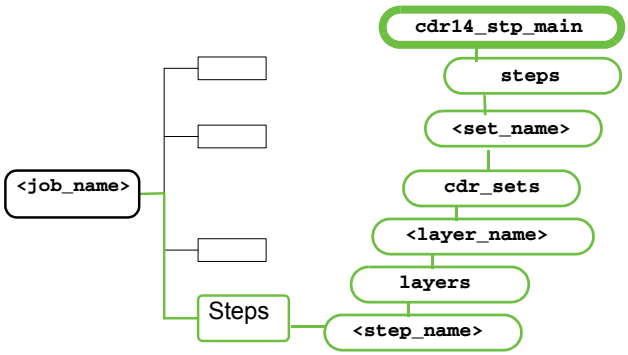
S steps/cdr14_stp_main (cdr Graphic Data)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/cdr14_stp_main**

This step holds the **cdr** graphic data. Inspection areas and exclusion zones are placed in the **lyr_area** layer and alignment targets are placed in the **lyr_targ** layer.

See "Job>steps Entity" on page 70 for step structure and "Job>steps>layers Entity" on page 103 for layer structure.

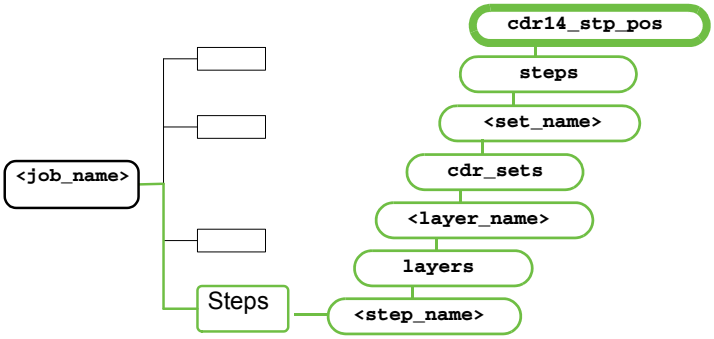
This step contains the steps described below, optionally step&repeated:



cdr14_stp_pos (Empty)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/cdr14_stp_pos
```

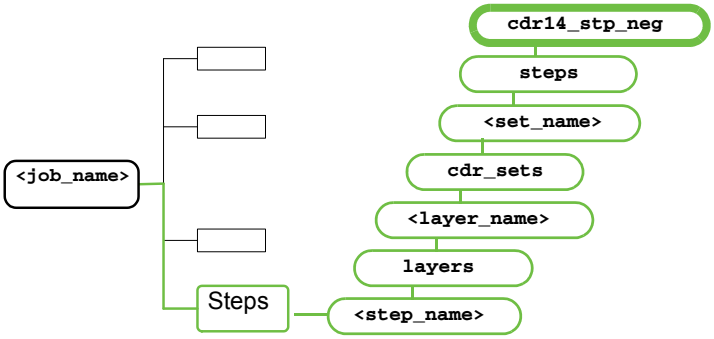
This step is left empty.



cdr14_stp_neg (Empty)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/cdr14_stp_neg
```

This step is left empty.

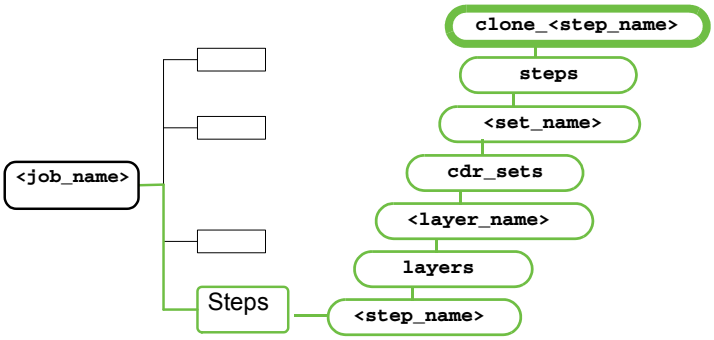


clone_<step_name> (Inspection Areas)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/clone_<step_name>
```

These steps contain step&repeated inspection areas as positive features and step&repeated exclusion zones as negative features (rectangular/polygonal surface or round pad) in its **lyr_area** layer.

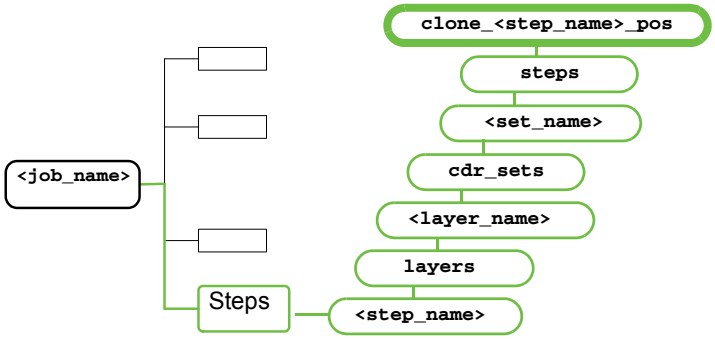
These steps are step&repeated with respect to the corresponding step **<step_name>**.



clone_<step_name>_pos (Automatic Inspection Area)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/clone_<step_name>_pos
```

These steps contain the 'automatic' inspection area that covers the whole step area (as defined by the step's profile or the step features' bounding box) as a positive rectangular/polygonal surface in its **lyr_area** layer.



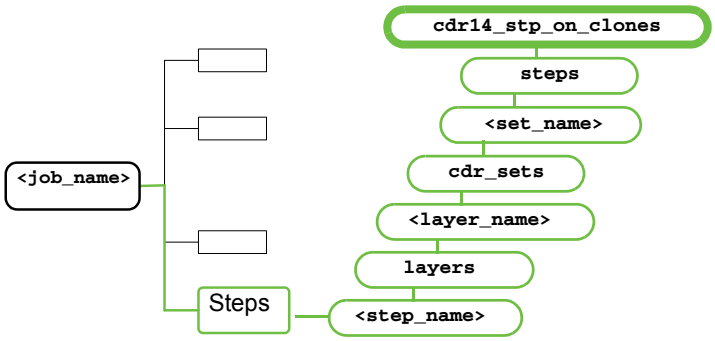
The **clone_<step_name>_pos** step appears in a single step&repeat command in the step&repeat table of step **clone_<step_name>**.

cdr14_stp_on_clones (Non-stp&rpt Zones)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/cdr14_stp_on_clones
```

This step contains non-step&repeated inspection areas as positive features and non-step&repeated exclusion zones as negative features (rectangular/polygonal surface or round pad) in its **lyr_area** layer.

This step appears as the last step&repeat command in the step&repeat table of step **cdr14_stp_main**.



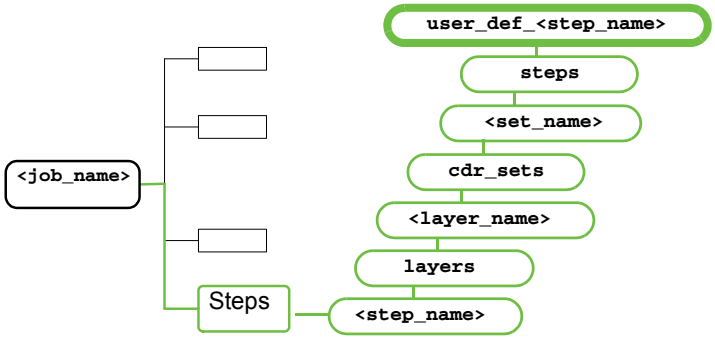
S user_def_<step_name> (AOI Panelization)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/user_def_<step_name>
```

These steps are created only in case AOI panelization was defined using the CDR interface ("cdrsr (AOI Panelization)" on page 98).

These steps replace the **clone_<step_name>** steps.

<step_name> is the step identification as supplied in the AOI panelization.



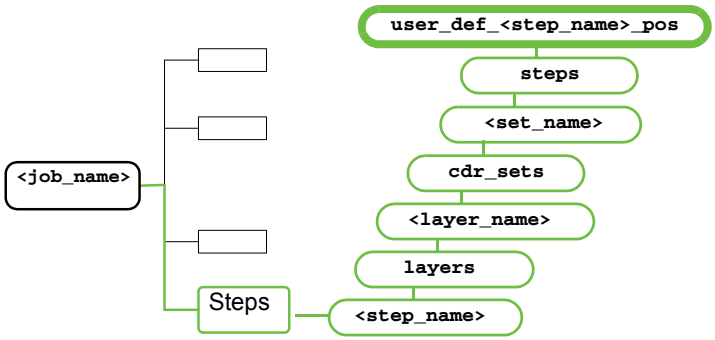
S user_def_<step_name>_pos (AOI Panelization)

```
<job_name>/steps/<step_name>/layers/<layer_name>/cdr_sets/<set_name>/steps/user_def_<step_name>_pos
```

These steps are created only in case AOI panelization was defined using the CDR interface ("cdrsr (AOI Panelization)" on page 98).

These steps replace the **clone_<step_name>_pos** steps.

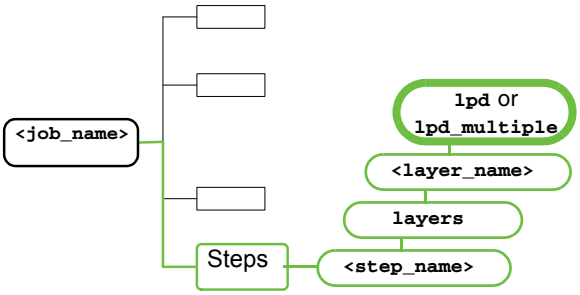
<step_name> is the step identification as supplied in the AOI panelization.



S lpd (Layer Production Data)

Type:	Line Record Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/lpd** or **lpd_multiple**

Layer Production data contains the plotting parameters for Orbotech plotters stored in one of two possible files—**lpd** or **lpd_multiple**. The **lpd_multiple** file contains **lpd** settings for all possible plotters; the **lpd** file contains settings for only the **EITHER TYPE** plotter. If both **lpd** and **lpd_multiple** files exist, the **lpd** file is ignored.



LPD

Image Production Parameters produced in Genesis are used by Orbotech plotters when processing IMG and OPFX files. When used for IMG output, the file is the **/<panel>/<layer>/layerhdr** file. The following is a list of elements that appear in the Genesis popup for setting parameters:

DEVICE_TYPE	Defines to which plotter the parameters are set. In an **lpd** file it is always **EITHER TYPE**. In **lpd_multiple** it is one of: **EITHER TYPE**, **LP7008**, **XPRESS**, **LP5008**, **DP100** or **LP9008**.
WAS_INPUT	Indicates where production parameters originated. Values: **YES**, **NO**
IS_DEFINED	For internal use. Values: **YES**, **NO**
POLARITY	Determines how to plot the image. Values: **POSITIVE**–the plot is the same as the entity image on the screen. **NEGATIVE**—entity image is reversed.
SPEED	Defines plotter speed. Values: minimum = **0** (automatic), maximum = **255**
XSTRETCH, YSTRETCH	To stretch or shrink (in percent) the X or Y dimension of the plotting entity.
XSHIFT, YSHIFT	Distance to shift the plotting entity along the X or Y axis in inches.
XMIRROR, YMIRROR	Mirror the plotting entity along the X or Y axis. If 0, no mirroring occurs.

COPPER_AREA	The copper area calculated during plot pre-processing in square inches.
XCENTER, YCENTER	Reference line in the X or Y axis around which point stretch begins.
PLOT_KIND1, PLOT_KIND2	Specifies the type of **lpd** set. Values: 0—5000 plotter 56—5008 plotter and newer **PLOT_KIND2** must be the same as **PLOT_KIND1**.
MINVEC	Widen all lines narrower than this width (expressed in mils) by the value defined in **ADVEC**.
ADVEC	The extent by which to enlarge lines defined in **MINVEC** expressed in mils.
MINFLASH	Widen all pads smaller than this size by the value defined in **ADFLASH**.
ADFLASH	The extent by which to enlarge pads defined in **MINFLASH**.
CONDUCTOR1	Specific line width to be incremented by 0.5mils.
CONDUCTOR2	Specific line width to be incremented by 0.5mils.
CONDUCTOR3	Specific line width to be incremented by 0.5mils.
CONDUCTOR4	Specific line width to be incremented by 0.5mils.
CONDUCTOR5	Specific line width to be incremented by 0.5mils.
MEDIA	Defines the plot media (film type). Values: **FIRST**, **SECOND**, **THIRD**
RESOLUTION	Resolution for plotting. Values: **HALF_MIL**, **QUARTER_MIL**, **EIGHTH_MIL**, **SIXTEENTH_MIL**.
SMOOTHING	To leave a smooth or rough edge on diagonal lines. Values: **SMOOTH**, **ROUGH**
SWAP_AXES	To switch the X and Y coordinates of the image before plotting. Values: **NO_SWAP**, **SWAP**
ELPD_IS_DEFINED	Specifies whether extended production data is defined. Values: **YES**/**NO**.
RESOLUTION_VALUE	Numeric resolution value.
RESOLUTION_UNITS	Units for resolution. Values: **MIL**/**MICRON**.
ENLARGE_POLARITY	To use extended production data only for features of the specified polarity. Values: **NONE**, **POSITIVE**, **NEGATIVE**, **BOTH**. Does not apply if polarity specified is **NONE**.
OTHER_POLARITY	Relevant only if **Apply enlarge to** is set to **positive** or **negative**. Values: **SIZE_INVERSELY**, **LEAVE_AS_IS**.
ENLARGE_PANEL_ELEMENT	Enlarge panel elements as PCB elements. Values: **YES**/**NO**.
ALLOW_PCB_PANEL_OVERLAP	Allow the panel data to overlap the PCB. Values: **YES**/**NO**.

ELEMENT_IMAGE_SYMBOLS	Enlarge components of the image symbol as regular features. Values: **YES**/**NO**.
ENLARGE_ZERO_LENGTH_VEC	Consider zero length lines as pads for the purpose of enlarging. Values: **YES**/**NO**
ENLARGE_SYMBOLS	Enlarge **ALL** or **SELECTED** symbols. If **ALL** symbols, you are asked to specify by how much to enlarge. Values: **NONE**, **ALL**, **SELECTED**.
ENLARGE_SYMBOLS_BY	Specify the degree of enlargement in mils for symbols.
SELECTED_SYMBOL[1-10]	Specify the symbol names to be enlarged.
SYMBOL_ADD[1-10]	Specify the extent by which to enlarge the symbol of the same index number in mils.
QUALITY	Defines the LP9008 plotter working mode. Values: **AUTO**, **FAST**, **FINE**.
ENLARGE_CONTOURS_BY	Specify the contour compensation in mils.

Example of LPD

```
DEVICE_TYPE=EITHER TYPE
WAS_INPUT=NO
IS_DEFINED=YES
POLARITY=NEGATIVE
SPEED=0
XSTRETCH=100.12
YSTRETCH=100.23
XSHIFT=0
YSHIFT=0
XMIRROR=2.5
YMIRROR=0
COPPER_AREA=0
XCENTER=0
YCENTER=0
PLOT_KIND1=56
PLOT_KIND2=56
MINVEC=4
ADVEC=1
MINFLASH=7
ADFLASH=1
CONDUCTOR1=5
CONDUCTOR2=6
CONDUCTOR3=7
CONDUCTOR4=0
CONDUCTOR5=0
MEDIA=FIRST
RESOLUTION=QUARTER_MIL
SMOOTHING=SMOOTH
SWAP_AXES=NO_SWAP
ELPD_IS_DEFINED=YES
RESOLUTION_VALUE=0.25
RESOLUTION_UNITS=MIL
QUALITY=AUTO
```

```
ENLARGE_POLARITY=POSITIVE
OTHER_POLARITY=SIZE_INVERSELY
ENLARGE_PANEL_ELEMENTS=YES
ALLOW_PCB_PANEL_OVERLAP=YES
ENLARGE_IMAGE_SYMBOLS=YES
ENLARGE_ZERO_LENGTH_VEC=YES
ENLARGE_SYMBOLS=SELECTED
ENLARGE_SYMBOLS_BY=0
SELECTED_SYMBOL1=DPF.11_11
SYMBOL_ADD1=1
SELECTED_SYMBOL2=E-VGP1
SYMBOL_ADD2=1
SELECTED_SYMBOL3=OCA_IPC_M_SYM
SYMBOL_ADD3=1
SELECTED_SYMBOL4=PET-MARK-X_21
SYMBOL_ADD4=1
SELECTED_SYMBOL5=
SYMBOL_ADD5=0
SELECTED_SYMBOL6=
SYMBOL_ADD6=0
SELECTED_SYMBOL7=
SYMBOL_ADD7=0
SELECTED_SYMBOL8=
SYMBOL_ADD8=0
SELECTED_SYMBOL9=
SYMBOL_ADD9=0
SELECTED_SYMBOL10=
SYMBOL_ADD10=0
ENLARGE_CONTOURS_BY=0
```

LPD_MULTIPLE

The **lpd_multiple** file is a structured text file containing parameters for one or more plotter types. The file is structured as follows:

```
LPD {
```

The same parameters as in an lpd file.

```
}
LPD {
...
}
```

Each device type appears only once in the file.

Example

```
LPD {
    DEVICE_TYPE=EITHER TYPE
    WAS_INPUT=NO
    IS_DEFINED=YES
    POLARITY=NEGATIVE
    SPEED=0
    XSTRETCH=100.12
    YSTRETCH=100.23
    XSHIFT=0
```

```
YSHIFT=0
XMIRROR=2.5
YMIRROR=0
COPPER_AREA=0
XCENTER=0
YCENTER=0
PLOT_KIND1=56
PLOT_KIND2=56
MINVEC=4
ADVEC=1
MINFLASH=7
ADFLASH=1
CONDUCTOR1=5
CONDUCTOR2=6
CONDUCTOR3=7
CONDUCTOR4=0
CONDUCTOR5=0
MEDIA=FIRST
RESOLUTION=QUARTER_MIL
SMOOTHING=SMOOTH
SWAP_AXES=NO_SWAP
ELPD_IS_DEFINED=YES
RESOLUTION_VALUE=0.25
RESOLUTION_UNITS=MIL
QUALITY=AUTO
ENLARGE_POLARITY=POSITIVE
OTHER_POLARITY=SIZE_INVERSELY
ENLARGE_PANEL_ELEMENTS=YES
ALLOW_PCB_PANEL_OVERLAP=YES
ENLARGE_IMAGE_SYMBOLS=YES
ENLARGE_ZERO_LENGTH_VEC=YES
ENLARGE_SYMBOLS=SELECTED
ENLARGE_SYMBOLS_BY=0
SELECTED_SYMBOL1=DPF.11_11
SYMBOL_ADD1=1
SELECTED_SYMBOL2=E-VGP1
SYMBOL_ADD2=1
SELECTED_SYMBOL3=OCA_IPC_M_SYM
SYMBOL_ADD3=1
SELECTED_SYMBOL4=PET-MARK-X_21
SYMBOL_ADD4=1
SELECTED_SYMBOL5=
SYMBOL_ADD5=0
SELECTED_SYMBOL6=
SYMBOL_ADD6=0
SELECTED_SYMBOL7=
SYMBOL_ADD7=0
SELECTED_SYMBOL8=
SYMBOL_ADD8=0
SELECTED_SYMBOL9=
SYMBOL_ADD9=0
SELECTED_SYMBOL10=
SYMBOL_ADD10=0
```

```
    ENLARGE_CONTOURS_BY=0
}

LPD {
    DEVICE_TYPE=LP5008
    WAS_INPUT=NO
    IS_DEFINED=YES
    POLARITY=NEGATIVE
    SPEED=0
    XSTRETCH=100.12
    YSTRETCH=100.23
    XSHIFT=0
    YSHIFT=0
    XMIRROR=2.5
    YMIRROR=0
    COPPER_AREA=0
    XCENTER=0
    YCENTER=0
    PLOT_KIND1=56
    PLOT_KIND2=56
    MINVEC=4
    ADVEC=1
    MINFLASH=7
    ADFLASH=1
    CONDUCTOR1=5
    CONDUCTOR2=6
    CONDUCTOR3=7
    CONDUCTOR4=0
    CONDUCTOR5=0
    MEDIA=FIRST
    RESOLUTION=QUARTER_MIL
    SMOOTHING=SMOOTH
    SWAP_AXES=NO_SWAP
    ELPD_IS_DEFINED=YES
    RESOLUTION_VALUE=0.25
    RESOLUTION_UNITS=MIL
    QUALITY=AUTO
    ENLARGE_POLARITY=POSITIVE
    OTHER_POLARITY=SIZE_INVERSELY
    ENLARGE_PANEL_ELEMENTS=YES
    ALLOW_PCB_PANEL_OVERLAP=YES
    ENLARGE_IMAGE_SYMBOLS=YES
    ENLARGE_ZERO_LENGTH_VEC=YES
    ENLARGE_SYMBOLS=SELECTED
    ENLARGE_SYMBOLS_BY=0
    SELECTED_SYMBOL1=DPF.11_11
    SYMBOL_ADD1=1
    SELECTED_SYMBOL2=E-VGP1
    SYMBOL_ADD2=1
    SELECTED_SYMBOL3=OCA_IPC_M_SYM
    SYMBOL_ADD3=1
    SELECTED_SYMBOL4=PET-MARK-X_21
    SYMBOL_ADD4=1
```

```
    SELECTED_SYMBOL5=
    SYMBOL_ADD5=0
    SELECTED_SYMBOL6=
    SYMBOL_ADD6=0
    SELECTED_SYMBOL7=
    SYMBOL_ADD7=0
    SELECTED_SYMBOL8=
    SYMBOL_ADD8=0
    SELECTED_SYMBOL9=
    SYMBOL_ADD9=0
    SELECTED_SYMBOL10=
    SYMBOL_ADD10=0
    ENLARGE_CONTOURS_BY=0
}

LPD {
    DEVICE_TYPE=LP7008
    WAS_INPUT=NO
    IS_DEFINED=YES
    POLARITY=NEGATIVE
    SPEED=0
    XSTRETCH=100.12
    YSTRETCH=100.23
    XSHIFT=0
    YSHIFT=0
    XMIRROR=2.5
    YMIRROR=0
    COPPER_AREA=0
    XCENTER=0
    YCENTER=0
    PLOT_KIND1=56
    PLOT_KIND2=56
    MINVEC=4
    ADVEC=1
    MINFLASH=7
    ADFLASH=1
    CONDUCTOR1=5
    CONDUCTOR2=6
    CONDUCTOR3=7
    CONDUCTOR4=0
    CONDUCTOR5=0
    MEDIA=FIRST
    RESOLUTION=QUARTER_MIL
    SMOOTHING=SMOOTH
    SWAP_AXES=NO_SWAP
    ELPD_IS_DEFINED=YES
    RESOLUTION_VALUE=0.25
    RESOLUTION_UNITS=MIL
    QUALITY=AUTO
    ENLARGE_POLARITY=POSITIVE
    OTHER_POLARITY=SIZE_INVERSELY
    ENLARGE_PANEL_ELEMENTS=YES
    ALLOW_PCB_PANEL_OVERLAP=YES
```

```
    ENLARGE_IMAGE_SYMBOLS=YES
    ENLARGE_ZERO_LENGTH_VEC=YES
    ENLARGE_SYMBOLS=SELECTED
    ENLARGE_SYMBOLS_BY=0
    SELECTED_SYMBOL1=DPF.11_11
    SYMBOL_ADD1=1
    SELECTED_SYMBOL2=E-VGP1
    SYMBOL_ADD2=1
    SELECTED_SYMBOL3=OCA_IPC_M_SYM
    SYMBOL_ADD3=1
    SELECTED_SYMBOL4=PET-MARK-X_21
    SYMBOL_ADD4=1
    SELECTED_SYMBOL5=
    SYMBOL_ADD5=0
    SELECTED_SYMBOL6=
    SYMBOL_ADD6=0
    SELECTED_SYMBOL7=
    SYMBOL_ADD7=0
    SELECTED_SYMBOL8=
    SYMBOL_ADD8=0
    SELECTED_SYMBOL9=
    SYMBOL_ADD9=0
    SELECTED_SYMBOL10=
    SYMBOL_ADD10=0
    ENLARGE_CONTOURS_BY=0
}

LPD {
    DEVICE_TYPE=LP9008
    WAS_INPUT=NO
    IS_DEFINED=YES
    POLARITY=NEGATIVE
    SPEED=0
    XSTRETCH=100.33
    YSTRETCH=99.78400000000001
    XSHIFT=0
    YSHIFT=0
    XMIRROR=2.5
    YMIRROR=0
    COPPER_AREA=0
    XCENTER=0
    YCENTER=0
    PLOT_KIND1=56
    PLOT_KIND2=56
    MINVEC=4
    ADVEC=1
    MINFLASH=7
    ADFLASH=1
    CONDUCTOR1=5
    CONDUCTOR2=6
    CONDUCTOR3=7
    CONDUCTOR4=0
    CONDUCTOR5=0
```

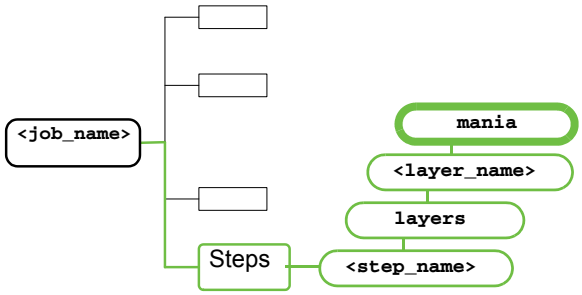
```
    MEDIA=FIRST
    RESOLUTION=EIGHTH_MIL
    SMOOTHING=SMOOTH
    SWAP_AXES=NO_SWAP
    ELPD_IS_DEFINED=YES
    RESOLUTION_VALUE=1
    RESOLUTION_UNITS=MICRON
    QUALITY=FINE
    ENLARGE_POLARITY=POSITIVE
    OTHER_POLARITY=SIZE_INVERSELY
    ENLARGE_PANEL_ELEMENTS=YES
    ALLOW_PCB_PANEL_OVERLAP=YES
    ENLARGE_IMAGE_SYMBOLS=YES
    ENLARGE_ZERO_LENGTH_VEC=YES
    ENLARGE_SYMBOLS=SELECTED
    ENLARGE_SYMBOLS_BY=0
    SELECTED_SYMBOL1=DPF.11_11
    SYMBOL_ADD1=1
    SELECTED_SYMBOL2=E-VGP1
    SYMBOL_ADD2=1
    SELECTED_SYMBOL3=OCA_IPC_M_SYM
    SYMBOL_ADD3=1
    SELECTED_SYMBOL4=PET-MARK-X_21
    SYMBOL_ADD4=1
    SELECTED_SYMBOL5=
    SYMBOL_ADD5=0
    SELECTED_SYMBOL6=
    SYMBOL_ADD6=0
    SELECTED_SYMBOL7=
    SYMBOL_ADD7=0
    SELECTED_SYMBOL8=
    SYMBOL_ADD8=0
    SELECTED_SYMBOL9=
    SYMBOL_ADD9=0
    SELECTED_SYMBOL10=
    SYMBOL_ADD10=0
    ENLARGE_CONTOURS_BY=0
}

LPD {
    DEVICE_TYPE=DP100
    WAS_INPUT=NO
    IS_DEFINED=YES
    POLARITY=NEGATIVE
    SPEED=0
    XSTRETCH=100.001
    YSTRETCH=100.003
    XSHIFT=0
    YSHIFT=0
    XMIRROR=0
    YMIRROR=2.500075
    COPPER_AREA=0
    XCENTER=2500
```

```
    YCENTER=2500
    PLOT_KIND1=56
    PLOT_KIND2=56
    MINVEC=4
    ADVEC=1
    MINFLASH=7
    ADFLASH=1
    CONDUCTOR1=5
    CONDUCTOR2=6
    CONDUCTOR3=7
    CONDUCTOR4=0
    CONDUCTOR5=0
    MEDIA=FIRST
    RESOLUTION=QUARTER_MIL
    SMOOTHING=SMOOTH
    SWAP_AXES=NO_SWAP
    ELPD_IS_DEFINED=YES
    RESOLUTION_VALUE=0.25
    RESOLUTION_UNITS=MIL
    QUALITY=AUTO
    ENLARGE_POLARITY=POSITIVE
    OTHER_POLARITY=SIZE_INVERSELY
    ENLARGE_PANEL_ELEMENTS=YES
    ALLOW_PCB_PANEL_OVERLAP=YES
    ENLARGE_IMAGE_SYMBOLS=YES
    ENLARGE_ZERO_LENGTH_VEC=YES
    ENLARGE_SYMBOLS=SELECTED
    ENLARGE_SYMBOLS_BY=0
    SELECTED_SYMBOL1=DPF.11_11
    SYMBOL_ADD1=0.5
    SELECTED_SYMBOL2=E-VGP1
    SYMBOL_ADD2=0.5
    SELECTED_SYMBOL3=OCA_IPC_M_SYM
    SYMBOL_ADD3=0.5
    SELECTED_SYMBOL4=PET-MARK-X_21
    SYMBOL_ADD4=0.5
    SELECTED_SYMBOL5=
    SYMBOL_ADD5=0
    SELECTED_SYMBOL6=
    SYMBOL_ADD6=0
    SELECTED_SYMBOL7=
    SYMBOL_ADD7=0
    SELECTED_SYMBOL8=
    SYMBOL_ADD8=0
    SELECTED_SYMBOL9=
    SYMBOL_ADD9=0
    SELECTED_SYMBOL10=
    SYMBOL_ADD10=0
    ENLARGE_CONTOURS_BY=0
}
```

S mania (MANIA Automatic Optical Inspection)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/mania**

The file contains parameters of a Mania set describing parameters to be used when testing the layer for this entity.



angle	Alignment of panel: Rotation value.
mirror	Alignment of panel: Mirroring value
offset	Alignment of panel: Offset value
<mirror>	N for not mirrored, M for mirrored
resolution	Mania pixel size
scan_area	Size of scan area to be used for testing
outdir	Location to place output files for the Mania Sapphire AOI machine
size_table	Set of legal space and track values

Example

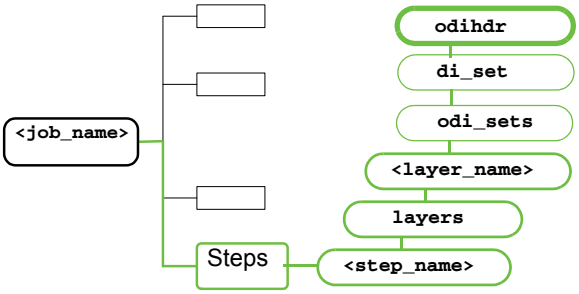
```
ANGLE=0
MIRROR=NO
OFFSET {
      X=0
      Y=0
 }
RESOLUTION=0.8267716535433072
SCAN_AREA {
      X1=-1.613008070866142
      Y1=-1.437398326771653
      X2=2.081300787401575
      Y2=1.52845531496063
 }
OUTDIR=/tmp
SIZE_TABLE {
      SPACE=5
      TRACK=20
 }
SIZE_TABLE {
```

```
        SPACE=4
        TRACK=34
}
```

DI (Orbotech Direct Imaging Interface)

Type:	Structured Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/odi_sets/di_set/odihdr**

This file contains the values for DI parameters of the layer. (For Frontline Users Only)



Description

SIDE	Top/Bottom
PARTNER	Name of the pair layer
SCALE_MODE	Name of the process for Image Production Parameters (according to configuration)
SCALE_NAME	No Scale/Fixed Scale/Auto Scale/One Pass Scale/ Fixed Measure Scale/Auto Segment Scale
BOARD_NAME	Name of the PCB Material (according to configuration)
THICKNESS	PCB Material thickness (inch)
RESIST_TYPE	Name of the Resist type (according to configuration)
TREATMENT	Name of the Treatment (used by the DI registration system for selecting targets)
CLIP_LIMITS	Define clip area
IMAGE_ROTATE	Panel on the PCB Material 0/90/180/270 measured in degrees CW (clockwise)
ALIGN	Define alignment (PCB Material on the DI Table)
REGIST_TOOL	CCD/UV Marker/BEAM
REGIST_FILE	Name of the registration file for CCD/Beam registration (Not applicable for UV Marker)

CLIP_LIMITS Array Structure

X_MIN	Minimal X
Y_MIN	Minimal Y
X_MAX	Maximal X
Y_MIN	Maximal Y

ALIGN Array Structure

USED	No—Alignment not set Yes—Alignment is set
ALIGN_TRANS	Define alignment transformation of the PCB Material on the table (see below)
ALIGN_TOOL	Pins/Ruler/DIM_Late_Select (DIM_Late_Select—alignment transformation is set on the DI Manager application.)
ALIGN_TYPE	Left/Right/Center/TopLeft_BottomRight (applicable in Ruler only)
RULER_NAME	Name of the ruler (applicable in Ruler only)
PINSET_NAME	Name of the pin set (applicable in Pins only)

ALIGN_TRANS Array Structure

XOFFSET	The X position on the DI Table of the step's lower left corner after rotation and mirror.
YOFFSET	The Y position on the DI Table of the step's lower left corner after rotation and mirror.
ANGLE	0/90/180/270 measured in degrees CW (clockwise)
MIRROR	No/Yes

Example

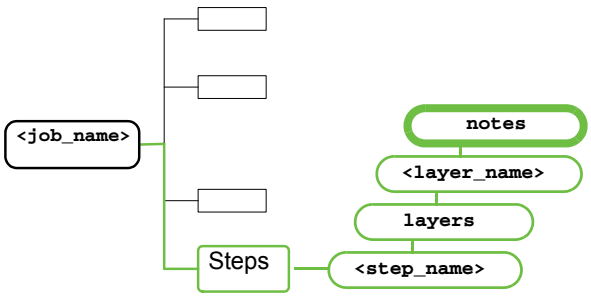
```
SIDE=TOP
PARTNER=IN03
PPOCESS=
SCALE_MODE=FIXED MEASURE SCALE
BOARD_NAME=20X16
THICKNESS=0.0787
RESIST_TYPE=ES102
TREATMENT=BRUSHING_LIGHT

CLIP_LIMITS {
    X_MIN=0
    Y_MIN=0
    X_MAX=12
    Y_MAX=16
}
```

```
IMAGE_ROTATE = 0

ALIGN {
    USED=YES

    ALIGN_TRANS {
        XOFFET=6
        YOFFSET=3
        ANGLE=0
        MIRROR=NO
    }
    ALIGN_TOOL=RULER
    ALIGN_TYPE=CENTER
    RULER_NAME=RULER1
    PINSET_NAME=
}

REGIST_TOOL=CCD
REGIST_FILE=
```

S notes (Electronic Job Notes)

Type:	Line Record Text
Compression:	None
Sum file:	Yes
Path	**<job_name>/steps/<step_name>/layers/<layer_name>/notes**

This file contains all the notes added by the user to the graphical layer.



Example

```
866467418,moshik,2.03807,
-1.22818,,,,,,First
line\nSecond line
```

Each line in the notes file has the following format:

```
<time>,<user>,<x>,<y>,
,,,,,<note>
```

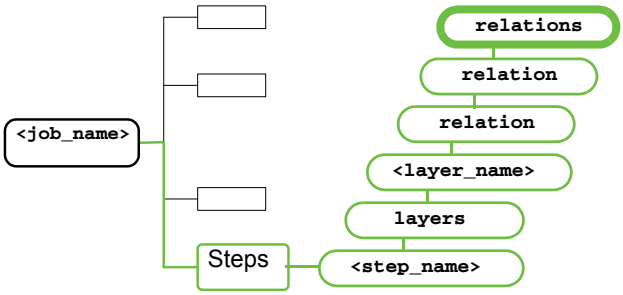
Where:

<time>	Last update date.UNIX time (seconds starting January 1st, 1970)
<user>	Last user updating the note
<x>,<y>	Graphic location, in inches
<note>	Up to 4 lines of text when the \n character describes the line break

S relations (Connections between Features)

Type:	Structured Text
Compression:	None
Sum file:	Yes
`<job_name>/steps/<step_name>/layers/<layer_name>/relation/relation/relations`	

This optional file contains all dimension and connections between features in a layer.



Examples

```
"relations" {
    "version"=1
    "rel_type"=DIM
    "dimension" {
        "dmode"=DXDY
        "dx"=0
        "dy"=0.74
        "angle"=0
        "linetype"=HORZ
        "is_special"=0
        "source_f" {
            "type"=FEAT
            "feature" {
                "index"=0
                "mode"=ALL
            }
        }
        "dest_f" {
            "type"=FEAT
            "feature" {
                "index"=1
                "mode"=ALL
            }
        }
    }
    "graphic" {
        "grp-params" {
            "ang_arrmode"=EDGE
            "dim_arrmode"=EDGE
            "ang_boxmode"=SQR
            "dim_boxmode"=SQR
            "inline_mode"=ALL
            "outline_mode"=ALL
            "font_spec"=TMR10
            "text_sufx"=
        }
        "dim_x"=0
        "dim_y"=0.374
        "ang_x"=0
```

```
        "ang_y"=0
    }
}

"relations" {
    "version"=1
    "rel_type"=CON
    "connection" {
        "feature-1" {
            "type"=FEAT
            "feature" {
                "index"=3
                "mode"=PS
            }
        }
        "feature-2" {
            "type"=FEAT
            "feature" {
                "index"=4
                "mode"=PE
            }
        }
        "feature-c" {
            "type"=FEAT
            "feature" {
                "index"=-1
                "mode"=ALL
            }
        }
        "mode"=CORNER
        "size1"=0
        "size2"=0
        "type_x"=DIST
        "type_y"=DIST
        "point_rel_2_f1"=ALL
        "func"=LINE2ARC
        "intersect"=0
        "radius"=0
    }
    "graphic" {
        "grp-params" {
            "ang_arrmode"=EDGE
            "dim_arrmode"=EDGE
            "ang_boxmode"=SQR
            "dim_boxmode"=SQR
            "inline_mode"=ALL
            "outline_mode"=ALL
            "font_spec"=
            "text_sufx"=
           }
        "dim_x"=0
        "dim_y"=0
        "ang_x"=0
```

```
        "ang_y"=0
    }
}
```

The file contains relations in the following structure:

version = always one (for future use)

rel_type = **CON** for connection or **DIM** for dimension.

connection or dimension structure = according to type

graphic structure:

grp-params structure:

ang_arrmode	Graphic description of dimension arrow in angle dimension	**EDGE** - arrow head outline **FULL** - not supported **NONE** - no arrow
dim_arrmode	Graphic description of dimension arrow in regular dimension	**EDGE/FULL/NONE**
ang_boxmode	Graphic description of box displayed around dimension value in angle dimension:	**SQR** -square box **RND** -round box **NONE** -values only **EMPTY**-does not display any thing
dim_boxmode	Graphic description of box displayed around dimension value in regular dimension	(options the same as above)
inline_mode	To display or not display the inner line of the dimension	**ALL** - display **NONE** - don't display
outline_mode	Graphic description of the extension to the inner line of dimension:	**ALL** - show extended line as an extension to the inner line **HORZ** - shows extended line horizontally **VERT**- shows extended line vertically
font_spec	Type and size of font used to display dimension values **<font type> <medium(M) or bold(B)> <regular(R) or italic(I)> <font size>**	where font types can be Times(**T**)/Helvetica(**H**)/ Courier(**C**)
text_sufx	Suffix to attach to dimension text	
dim_x	Place of dimension box in x (inches)	
dim_y	Place of dimension box in y (inches)	
ang_x	Place of angle box in x (inches)	
ang_y	Place of angle box in y (inches)	

Dimension type structure:

dmode	Only DXDY available	
dx	Delta x in inches	
dy	Delta y in inches	
angle	Angle if exist	
linetype	For line destinations	**HORZ** - horizontal **VERT** - vertical **DIAG** - diagonal
is_special	Dimension belongs to a symbol predefined **source_f** or **dest_f**	
source_f		**type** is always **FEAT** **feature** **index** - feature index in database **mode** - dimension mode **ALL** - all features **PS** - start of feature **PE** - end of feature
dest_f		**type** is always **FEAT** **feature** **index** - feature index in database **mode** - dimension mode **ALL** - all features **PS** - start of feature **PE** - end of feature

Connection type structure:

feature-1, feature-2	Features that are connected in the above feature type	
feature-c	Connecting feature in the above feature type(-1 if none)	
mode	Connection mode	**ROUND/CORNER/CHAMFER**
size1	Chamfer connections	if type_x is **DIST**: distance in x to cut from feature in inches if type_x is **ANGLE**: angle between chamfer line and feature
size2	Same as above for y	
type_x	Chamfer connection if the above size in **DIST** or **ANGLE**	
type_y	the same as above for y	

point_rel_2_f1	In case of more than one intersection point between features ALL/PS/PE	
func	Connection function:	**LINE2ARC** - intersection **LINE2CIRCLE** - line tangent to 2 circles **CIRCLE2LINE** - circle tangent 2 two lines **CIRCLE2CIRCLE** - arc tangent 2 two circles
intersect	For tangent features indication if arcs should be fixed.	
radius	Radius of round intersection and of circle	

Note Some of the relations, such as Dimension Types and Connection Types, relate mostly for use in a rout layer, even though they can be used in any other layer. A rout layer should be created exclusively for the definition of a rout.

NCD Entity see “NCD Entity” on page 164

NCR Entity see “NCR Entity” on page 177

Chapter 7 NCD Entity

The **ncd set** entity contains parameters and data for the Auto Drill Manager.

Required for GenFlex 6.4

New fields in NCD files *(Implemented in Genesis v9.3b also)*

File: **<job_name>/steps/<step_name>/layers/<layer_name>/ncd/<ncd_set_name>/header**

A new field **sr_r_command = no/yes** (use R command in excellon S&R repletion) is added to the format section.

```
format {
        . . .
      sr_r_command=YES
        . . .
}
```

File: **<job_name>/steps/<step_name>/layers/<layer_name>/ncd/<ncd_set_name>/table**

New fields are added:

- **stages** = no/yes - YES if the table was built using **.drill_stage** attribute.
- **by_length** = no/yes - YES if the table was built consider slot length.
- **nibble_type** = machine/software/start_end - START_END is a new option for slot pilot tools creation.
- **ind_sort** = 0…- table entry index.
- **stage_att** = [-1,0,1,2] - value of **.drill_stage** attribute
- **fix_tool_order_neg** = -xxx - used instead of

negative **fix_tool_order**

For example:

```
stages = YES
by_length = YES
entry {
        . . .
        nibble_type             = START_END
        . . .
        ind_sort                = 1
        stage_att               = 2
        fix_tool_order_neg      = -1
}
```

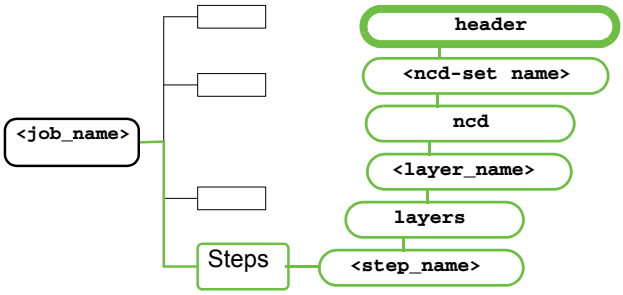
S *header*

Type:	Structured Text
Compression:	None
Sum file:	Yes
<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/header	

The **ncd** set entity contains data and parameters for the Auto Drill Manager.

This file describes the parameters used for creation of output NC files.

Note NCD = Numerical Control Drill.



Example:

```
machine=EXCELLON
thickness=0
rout_layer=NCD-R
no_touch_cpr=NO
tent_ar=0.012
num_stages=3
num_splits=1

reg {
    xsize1=24.5
    xsize2=24.5
    ysize=30
    xover=2
    angle=0
    mirror=NO
```

```
    xoff=0
    yoff=0
    xorigin=0
    yorigin=0
    version=1
    dx1=0
    dy1=0
    dx2=0
    dy2=0
    xscale=1.02
    yscale=0.99
    xscale_o=1.3
    yscale_o=-3
}
format {
    format=EXCELLON2
    zeroes=TRAILING
    units=INCH
    tool_units=INCH
    nf1=2
    nf2=4
    decimal=NO
    sr_start_code=25
    modal_ords=YES
    single_sr=NO
    sr_zero_set=NO
    repetitions=SR
    incremental=NO
}
split {
    axis=NONE
    coord=0
    sign=NEGATIVE
}
optimize {
    break_sr=YES
    optimize=YES
    iterations=10
    reduction_percent=1
    xspeed=400
    yspeed=400
    diag_mode=45ORT
}
z_axis {
    z_head=0
}
time {
    bit_change=0
    tool_change=0
}
tools_assign {
    fixed_tools=NO
    mode=INCREASING_SIZE
```

```
}
start_end {
    with_pilots=NO

    split {
        start=YES
        end=YES
        x=3.5
        y=1.2
        angle=0
        num_cols=0
        dist_type=SPACING
        min_dist=20
        min_hits=0
        min_size=0
        max_size=0
        end2=NO
        x2=0
        y2=0
        angle2=0
        num_cols2=0
    }
    split {
        start=NO
        end=NO
        x=0
        y=0
        angle=0
        num_cols=0
        dist_type=SPACING
        min_dist=0
        min_hits=0
        min_size=0
        max_size=0
        end2=NO
        x2=0
        y2=0
        angle2=0
        num_cols2=0
    }
}
```

Description

machine	Machine file name
thickness	Board thickness
rout_layer	Name of rout layer, where features redirected for rout will be located
no_touch_cpr	**Yes** – allows touching of copper by holes (Yes/No)
tent_ar	Tenting annular ring (inch/mm). If the distance of a hole from copper is less than **tent_ar**, it is considered touching copper

num_stages	Number of stages
num_splits	Number of splits

REG Structure

xsize1	Size of spindle table 1 in X
xsize2	Size of spindle table 2 in X
ysize	Size of spindle table in Y
xover	Overlap in table 1 and table 2 in X (used when panel is larger than table)
angle	Panel angle origin on table
mirror	Data is to be mirrored when placed on table (**Yes/No**)
xoff	Shift of panel in X relative to machine corner
yoff	Shift of panel in Y relative to machine corner
xorigin	Shift of zero in X to locate the machine 0
yorigin	Shift of zero in Y to locate the machine 0
version	Version of X/Y output that should be created
dx1	Shift in X to be applied in split situation for table 1
dy1	Shift in Y to be applied in split situation for table 1
dx2	Shift in X to be applied in split situation for table 2
dy2	Shift in Y to be applied in split situation for table 2
xscale	Scale factor by X
yscale	Scale factor by X
xscale_o	X scale anchor
yscale_o	X scale anchor

Format Structure

Format structure describes the format of output NC file.

format	Format type of Output NC file
zeroes	Zeros omitted (**Trailing/Leading/None**)
units	Units the drill data is created in (**Inch/mm**)
tool_units	Units the drill size is created in (**Inch/mm**)
nf1	Numbering format most significant number (**M.L**) e.g. 2.4
nf2	Numbering format least significant number (**M.L**) e.g. 2.4
decimal	To place a decimal point in the output coordinates **Yes** - places decimal point, for example, X4.345Y2.3 **No** – doesn't place decimal point X210Y340)

sr_start_code	Starting code for step & repeat block (Excellon format). Some machines use the M25 code for step & repeat, and others use M31. This value should be **25**, or **31**.
modal_coords	To remove identical X,Y coordinates. (**Yes/No**). For example, we have two sets of coordinates: x1=5.4, y1=2.5 x2=6.3, y2=2.5 When modal_coords = Yes, the following results in the file: X5.4Y2.5 X6.3
single_sr	Applies to a step & repeat block that has only one 'repeat'. **Yes** - that repeat will be placed within a step & repeat block. In Excellon format it means that it will be inside a M25 block. (**Yes/No**).
sr_zero_set	Sets step & repeat block location relative to datum. **Yes** - sets all the coordinates of a step & repeat block relative to a specified datum. (**Yes/No**).
repetitions	Repetitions type: **SR** – Step & repeat blocks **Subroutine** – subroutines (only in Hitachi format)
incremental	**Yes** - each point's coordinates are given as an increment to a previous point.

Split Structure

Split structure describes the panel split when a panel size is larger than a machine table.

axis	Axis to split a panel at: **None** – no split **X** – split by X axis **Y** – split by Y axis
coord	The coordinate at which to split along the axis
sign	Direction from a split line: **Negative** – all points to the left (or below) a split line belong to split **Positive** – all points to the right (or above) a split line belong to split

Optimize Structure

Optimize structure contains parameters that affect drill optimization algorithm.

break_sr	Break step & repeat (**Yes/No**)
optimize	Run drill optimization (**Yes/No**)
iterations	Number of iterations for optimization
reduction_percent	Stop optimization iterations when improvement on any given iteration falls to less than the percent specified.
xspeed	Relative spindle speed in X

yspeed	Relative spindle speed in Y
diag_mode	Specifies how a drill approaches a point relative to the last point: **45ORT** (where **ort** = orthogonal) - starts at 45 degrees then continues along the X/Y axis. **ORT45 -** starts along the X/Y axis and branches out at 45 degrees. **DIRECT -** goes from point to point directly (in this case time and distance optimization are the same).

Z_AXIS Structure

z_head	Default clearance of tool from board

Time Structure

Time structure contains values that are used for estimating the drill time display in the Report.

bit_change	Time in minutes for bit change
tool_change	Time in minutes for tool change

Tools_assign Section

fixed_tools	Using fixed tools described in machine file (Yes/No)
mode	Tool assignment mode: **INCREASING_SIZE/ DECREASING_SIZE/ INCREASING_COUNT/DECREASING_COUNT**

Start_end Section

Start_end section describes parameters for creation of start/end coupon.

with_pilots	**Yes** – drill verification holes with their pilot holes. (**Yes/No**).

Array of **split** structures always contain two items. The first item describes start/end coupon for first split, and the second item for the second split.
This array is contained inside **Start_end** structure.

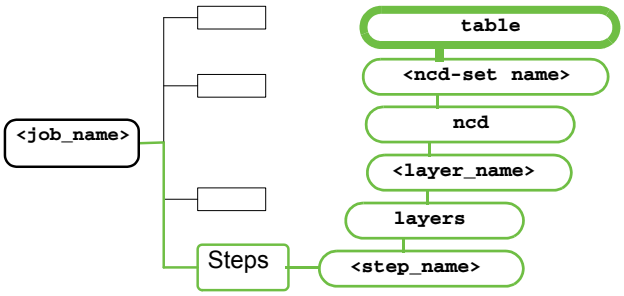
start	Starting hole coupon （**Yes/No**）
end	Ending hole coupon （**Yes/No**）
x	X coordinate of a first hole in the coupon
y	Y coordinate of a first hole in the coupon
angle	Direction of row (column) of holes (in angles)
num_cols	Number of hole columns
min_dis	Minimum distance between holes in coupon
dist_type	**Spacing** - defines minimum space between holes/slots edges **Center** - defines minimum distance between holes/slots centers
min_hits	Minimum number of drills. Don't place a drill on the coupon for tools that have less hits than this number.

min_size	Tool size to place in coupon
max_size	Maximum tool size to place in coupon
end2	**Yes** – use separate end coupon (**Yes/No**).
x2	X coordinate of a first hole in the separate end coupon
y2	Y coordinate of a first hole in the separate end coupon
angle2	Direction of row (column) of holes (in angles) in the separate end coupon
num_cols2	Number of hole columns in the separate end coupon

table

Type	Structured Text
Compression	None
Sum file	Yes
<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/table	

This file contains the NC table. It consists of an array of records, each one describing a table entry.



Example

```
entry {
    shape=HOLE
    type=PLATED
    size=30
    touch_copper=NO
    flag=0
    count=1
    finish_size=-9.84251968503937e-005
    min_tol=0
    max_tol=0
    designator=
    slot_len=0
    pilot=NO
    parent=-1
    mode=REGULAR
    spindle_speed=300
    feed_rate=85
    nibble_type=MACHINE
    tool_size=30
    max_hits=500
    stage=2
    rout_mode=SEPARATE
    hits=1
    text_count=3
}
```

Entry Structure

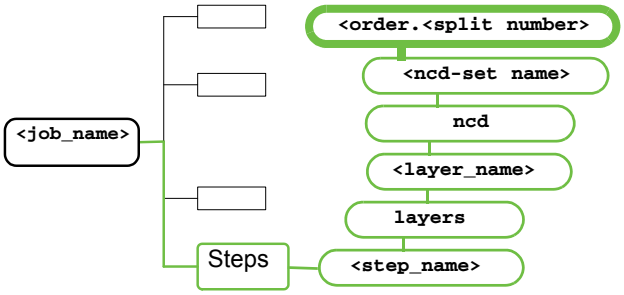
shape	**Hole / Slot**
type	**Plated** / **Non_plated** / **Via**
size	Drill size
touch_copper	**Yes** – there are drills of this size that touch copper
flag	Drill flag (passed by attributes)
count	Number of drills/slots
finish_size	Finish size
min_tol	Minimum size tolerance
max_tol	Maximum size tolerance
designator	Drill designator (comment)
slot_len	Total slot length
pilot	**Yes** - the drill is a pilot drill
parent	Parent index. Index table row that describes the parent for the pilot drill
mode	**Regular** – produce regular drill **Nibble** – produce nibble drill **Rout** – send the drills to a rout layer
spindle_speed	Spindle speed
feed_rate	Feed rate
nibble_type	Nibble type for nibble drills /slots: **Machine** – produce a machine command for the nibble **Software** – produce a sequence of smaller drills by using a nibble algorithm
rout_mode	**Separate** (always this value)
tool_size	Tool size
max_hits	Maximum hits parameter
stage	Stage number
hits	Number of hits
text_counts	Number of canned text features. If this number >0 the table entry describes a drilled (canned) text rather than a hole.

S order

Type	Structured Text
Compression	None
Sum file	Yes
<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/order.<split number>	

Split numbers can have values of 1 or 2.

The order file contains records defining the step drilling order in the Auto Drill Manager.



Description

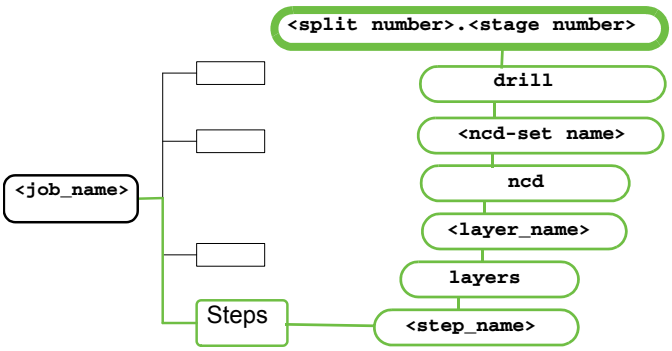
sr_line	Row Number of step & repeat table
sr_nx	Number of the step by X
sr_ny	Number of the step by Y
serial	Serial number of the step
full	Full step processing (Yes/No)
optional	Optional step (Yes/No)

S drill file

Type	Line record Text
Compression	None
Sum file	no
<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/drill/<split number>.<stage number>	

Split numbers can be 1 or 2; stage numbers can be from 1 to 3. Thus, drill file names look like “1.1”, “1.2”, “2.3”

Drill file is an intermediate output file produced by the Auto Drill Manager. This file is translated to an NC file.



Example

```
BS pcb
R 0.3496454
0.574853 90 N Y N N
```

```
R 0.5470589 0.0137815 0 N Y Y N
R 0.0470589 0.0137815 0 N N Y N
H 4 0.144246 0.032847 N N N Ns
H 2 0.209031 0.202162 N N N N
H 5 0.033487 0.051513 N N N N
BE pcb
S 0 0.383224 0.1250633 0.408224 0.1683646 N N N N
H 4 0.0824924 0.430607 N N N N
H 2 0.2518074 0.3658219 N N N N
H 1 0.492556 0.195215 N N N N
H 4 0.6808214 0.35187 N N N N
H 5 0.1011584 0.5413659 N N N N
H 6 0.8271084 0.504596 N N N N
T 0 1.235303 1.8140007 '' 0 N Y N N 0 N
T 0 1.187303 1.8860007 '' 0 N Y N N 0 N
T 0 1.211303 1.8860007 '' 0 N Y N N 0 N
T 1 2.9772455 1.8220594 'ABC' 0 N N N N 0 N
```

The file consists of lines. The first letter or two letters of a line define the type:

BS	block start record
BE	block end
R	repetition record
H	hole record
S	slot record
T	canned (drilled) text record.

Block start record - has the format: **`BS <step_name>`**; where **`step_name`** is the name of step being repeated by this block

Block end record - has the format: **`BE <step_name>`**; where **`step_name`** is the name of step being repeated by this block

Repetition record - describes a single repetition of a block. It has the format:

`R <dx> <dy> <angle> <mirror> <order> <full> <optional>`

`dx`	Offset of this repetition block by X
`dy`	Offset of this repetition block by Y
`angle`	Rotation angle of the step repetition (**0, 90, 180, 270**)
`mirror`	Mirror (**Y/N**)
`order`	**Y** – an order is set for this repetition (Y/N)
`full`	Full step processing (**Y/N**)
`optional`	Optional step processing (**Y/N**)

Hole record - has the format:

`H <row_number> <X> <Y> <optional> <basic> <noopt> <noscale>`

`row_number`	Row number in the NC table
`x, y`	Coordinates of the hole

optional	Optional drill (**Y/N**)
basic	Basic drill (**Y/N**)
noopt	Don't optimize this drill (**Y/N**)
noscale	Don't scale this drill (**Y/N**)

Slot record - has the format:

S <row_number> <X1> <Y1> <X2> <Y2> <optional> <basic> <noopt> <noscale>

row_number	Row number in the NC table
x1, y1	Coordinates of the start of the slot
x2, y2	Coordinate of the end of the slot
optional	Optional drill (**Y/N**)
basic	Basic drill (**Y/N**)
noopt	Don't optimize this drill (**Y/N**)
noscale	Don't scale this drill (**Y/N**)

Text record - has the format:

T <row_number> <X> <Y> <text> <font> <optional> <basic> <noopt> <noscale>

row_number	Row number in the NC table
x, y	Coordinates of the hole
text	Text string
font	Text dot matrix (5 – 5x7, 6 – 6x7)
optional	Optional drill (**Y/N**)
basic	Basic drill (**Y/N**)
noopt	Don't optimize this drill (**Y/N**)
noscale	Don't scale this drill (**Y/N**)

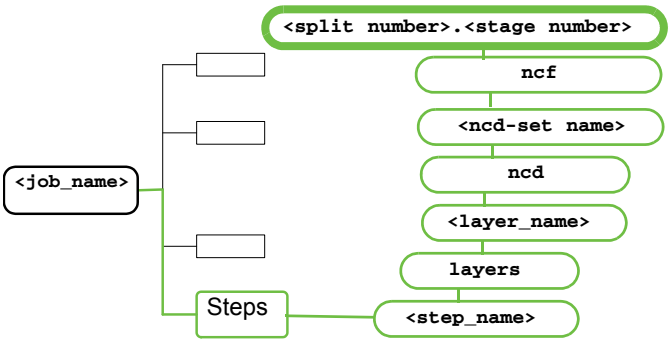
Note If the text string is empty, that line describes a single line of broken text. Broken text is a text that is output as a sequence of drills rather than a special NC command for text output.

S *NC File*

Type	Line record Text
Compression	None
Sum file	No
<job name>/steps/<step name>/layers/<layer name>/ncd/<ncd-set name>/ncf/<split number>.<stage number>	

Split numbers can be 1 or 2. Stage numbers can be from 1 to 3. Thus, NC file names look like “1.1”, “1.2”, “2.3”

The NC file is a real output file produced by the Auto Drill Manager. It has a format as requested by a user (e.g., Excellon, Posalux etc.).



Chapter 8 NCR Entity

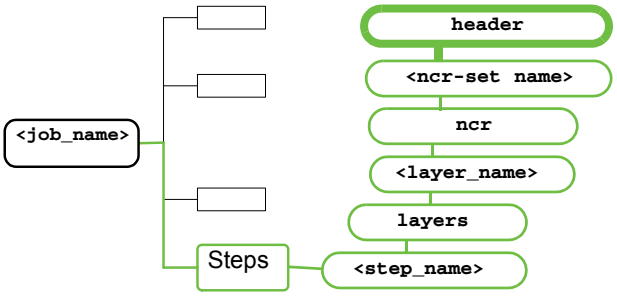
The **ncr set** entity contains parameters and data for the Auto Rout Manager.

S NCR header

Type	Structured Text
Compression	None
Sum file	Yes
<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/header	

This file describes the parameters used for creation of output NC files.

Note NCR = Numerical Control Rout.



Example

```
machine=EXCELLON
thickness=0
drill_layer=NCR-DRILL
sr_zero_drill_layer=
break_sr=NO
ccw=NO
angle_lines=NO
short_lines=NONE
press_down=YES
last_z_up=16
max_arc_ang=180
sep_lyrs=NO

reg {
    xsize=30
    ysize=50
    xover=0
    angle=0
    mirror=NO
    xoff=0
    yoff=0
    xorigin=0
    yorigin=0
    version=1
    xscale=1
    yscale=1
    xscale_o=0
    yscale_o=0
```

```
}
format {
    format=EXCELLON1
    zeroes=TRAILING
    units=MM
    tool_units=MM
    nf1=3
    nf2=2
    decimal=YES
    sr_start_code=25
    modal_coords=YES
    repetitions=SR
    single_sr=YES
    sr_zero_set=NO
}
```

Description

machine	Machine file name
thickness	Board thickness
drill_layer	Name of drill layer, where features redirected for drill will be placed
sr_zero_drill_layer	Name of a drill layer where a zero pad is located
break_sr	Break step & repeat (**Yes/No**)
ccw	Machine is in counter-clockwise mode (**Yes/No**)
angle_lines	Replace short lines by angled lines (**Yes/No**)
short_lines	Short lines mode: **None** – processing is defined by the angle_lines parameter **Box** – not supported **Angle** – replace short lines by angled lines **Toolout** – not supported
press_down	Pressure foot down on all tool ups (**Yes/No**)
last_z_up	Last tool up definition (**16** for M16 or **17** for M17)
max_arc_ang	Maximum angle of arc, any arc that has a sweep more than this value will be subdivided. (**0-360**)
sep_lyrs	Use separate layers when outputting steps with different orientations (**Yes/No**)

REG

Structure

xsize	Width of machine table
ysize	Height of machine table
xover	Not used
angle	Panel angle origin on table
mirror	Data are to be mirrored when placed on table (Yes/No)
xoff	Shift of panel in X relative to machine corner

yoff	Shift of panel in Y relative to machine corner
xorigin	Shift of zero in X to locate the machine 0
yorigin	Shift of zero in Y to locate the machine 0
version	Version of X/Y output that should be created (**1-8**)
xscale	Scale factor by X
yscale	Scale factor by Y
xscale_o	X scale anchor
yscale_o	Y scale anchor

Format Structure

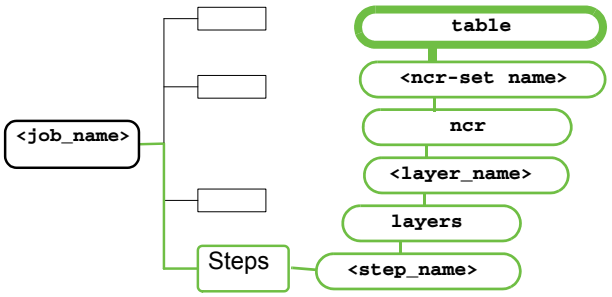
Format Structure describes the format of output NC file.

format	Format type of Output NC file
zeroes	Zeros omitted **(Trailing/Leading/None)**
units	Units in which the output data is created (**Inch/mm**)
tool_units	Units for tool definition data (**Inch/mm**)
nf1	Numbering format's most significant number (**M.L**) e.g. 2.4
nf2	Numbering format's least significant number (**M.L**) e.g. 2.4
decimal	To place a decimal point in the output coordinates. **Yes** - places decimal point, for example, X4.345Y2.3 **No** – does not place decimal point X210Y340
sr_start_code	Starting code for step & repeat block (Excellon format). Some machines use the M25 code for step & repeat, and others use M31. Value should be **25**, or **31**.
modal_coords	Modal coordinates (**Yes/No**). Used to remove identical X,Y coordinates. For example, we have two sets of coordinates: x1=5.4, y1=2.5 x2=6.3, y2=2.5 When modal_coords = Yes, then the following results in the file: X5.4Y2.5 X6.3
single_sr	Applies to a step & repeat block that has only one 'repeat'. **Yes** - that repeat will be placed within a step & repeat block. In Excellon format it means that it will be inside a M25 block.
sr_zero_set	Sets step & repeat block location relative to a pad on a special drill layer. **Yes** - sets all the coordinates of a step & repeat block relative to a specified pad or datum.
repetitions	Repetitions type: **SR** – step & repeat blocks **Subroutine** – subroutines (only in Hitachi format)

S table

Type	Structured Text
Compression	None
Sum file	Yes
`<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/table`	

This file contains the NC table. It consists of an array of records, each one describing a table entry.



Example

```
entry {
    type=CHAIN
    step_name=PANEL
    chain=1
    chain2=0
    size=0.02
    comp=LEFT
    path=0.7816070866141732
    count=0
    flag=0
    cw=0
    tool_size=0.02
    dup=NO
    parent=-1
    comp_factor=0.02
    spindle_speed=0
    feed_rate=0
    spiral=NONE
    mode=ROUT
    group=NEW
    order=1
    optional=NO
}
...
```

Entry Structure

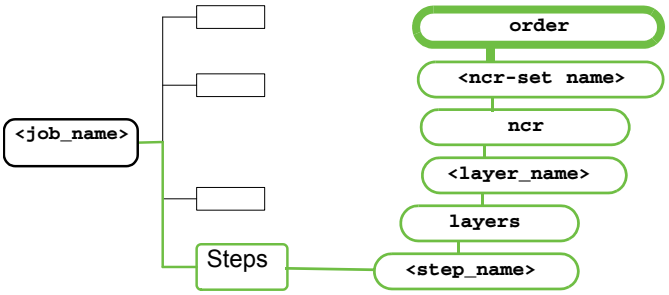
type	**Chain/Hole**
step_name	Name of step
chain	Chain number
chain2	Tag for secondary tool for the same chain
Size	Rout size
comp	Compensation (**Left/Right/None**)
path	Total path length (only for chains)
count	Number of holes
Flag	Rout flag (passed by attributes)

cw	Clockwise (**Yes/No**)
tool_size	Tool size
dup	Duplicate chain (**Yes/No**)
parent	Parent index. Index of table row that describes the parent for the duplicated chain
comp_factor	Compensation factor
spindle_speed	Spindle speed
feed_rate	Feed rate
spiral	Not used
mode	**Rout** – process in Auto Rout Manager **Drill** – place features on drill layer
group	**New** – start a new step & repeat block **Same** – continue the previous step & repeat block
order	Table row order for sorting purposes
optional	Optional output (**Yes/No**)

S *order*

Type	Structured Text
Compression	None
Sum file	Yes
<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/order	

The order file contains records defining the step processing order in the Auto Rout Manager.



Example

```
entry {
    order_sr {
        line=0
        nx=0
        ny=0
    }
    serial=1
    optional=NO
}
...
```

Description

serial	Serial number of the step
optional	Step is optional (**Yes/No**)

Order_sr Structure

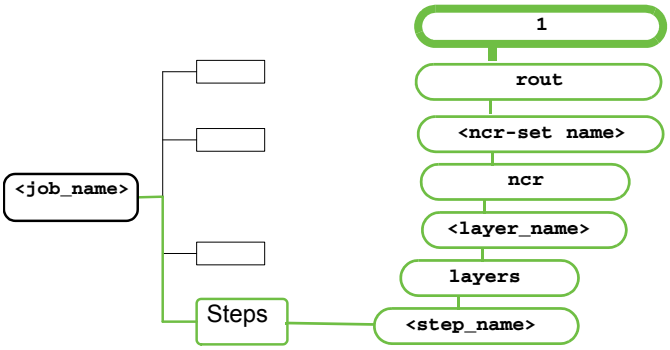
line	Row Number of step & repeat table
nx	Number of the step by X
ny	Number of the step by Y

S *rout file*

Type	Line Record Text
Compression	None
Sum file	no
<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/rout/1	

(File name is always 1.

Rout file is an intermediate output file generated by the Auto Rout Manager. It is translated into an NC file.



Example

```
BS pcb
R 0.0470589
0.0137815 0 N N N
R 0.5470589
0.0137815 0 N N N
R 0.0496454
0.574853 90 N N N
R 0.3496454 0.574853 90 N N N
R 0.6496454 0.574853 90 N N N
L 2 0 0 0 0 0.25 N N N
L 2 0 0 0.25 0.25 0.25 N N N
L 2 0 0.25 0.25 0.25 0.1 N N N
A 2 0 0.25 0.1 0.15 0 0.15 0.1 N N Y N
L 2 0 0.15 0 0 0 N N N
BE pcb
BS panel
R 0 0 0 N N N
L 0 0 0.064084 0.678131 0.845691 0.678131 N N N
BE panel
BS panel
R 0 0 0 N N N
L 1 0 0.674931 0.954491 0.931745 0.803752 N N N
BE panel
```

The file consists of lines. The first letter or two letters of a line define the type:

BS	block start record
BE	block end
R	repetition record
H	hole record

L	line record
A	arc record.

Block start record - has the format: **`BS <step_name>`**, where **`step_name`** is the name of the step being repeat20
ed by this block

Block end record - has the format: **`BE <step_name>, where step_name`** is the name of step being repeated by this block

Repetition record - describes a single repetition of a block. It has the format:
`R <dx> <dy> <angle> <mirror> <order> <full> <optional>`

`dx`	Offset of this repetition block by X
`dy`	Offset of this repetition block by Y
`angle`	Rotation angle of the step repetition (**0, 90, 180, 270**)
`mirror`	Mirror (**Y/N**)
`order`	**Y** – an order is set for this repetition (Y/N)
`optional`	Optional step processing (**Y/N**)

Hole record - has the format:
`H <row_number> <feed> <X> <Y> <optional> <basic> <noscale>`

`row_number`	Row number in the NC table
`feed`	Feed rate
`X, Y`	Coordinates of the hole
`optional`	Optional hole (**Y/N**)
`basic`	Basic hole (**Y/N**)
`noscale`	Don't scale this hole (**Y/N**)

Line record - has the format:
`L <row_number> <feed> <XS> <YS> <XE> <YE> <optional> <basic> <noscale>`

`row_number`	Row number in the NC table
`feed`	Feed rate
`xs, ys`	Coordinates of the line start
`xe, ye`	Coordinates of the line end
`optional`	Optional rout (**Y/N**)
`basic`	Basic rout (**Y/N**)
`noscale`	Don't scale this feature (**Y/N**)

Arc record - has the format:
`A <row_number> <feed> <XS> <YS> <XE> <YE> <XC> <YC> <optional> <basic> <cw> <noscale>`

`row_number`	Row number in the NC table
`feed`	Feed rate
`xs, ys`	Coordinates of the arc start
`xe, ye`	Coordinates of the arc end

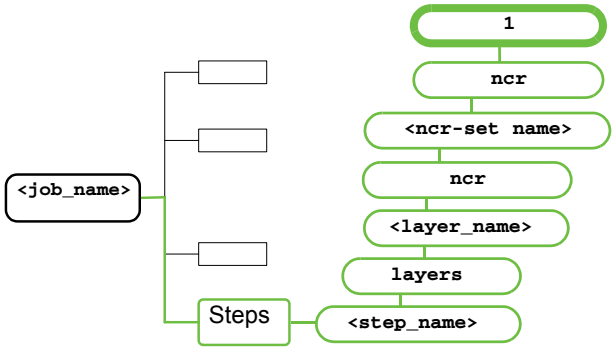
xc, yc	Coordinates of the arc center
optional	Optional rout (Y/N)
basic	Basic rout (Y/N)
cw	Clockwise (Y/N)
noscale	Don't scale this feature (Y/N)

S NC File

Type	Line Record Text
Compression	None
Sum file	no
<job name>/steps/<step name>/layers/<layer name>/ncr/<ncr-set name>/ncr/1	

File name is always 1.

NC file is a real output file generated by the Auto Rout Manager. It has a format as requested by a user (e.g., Excellon, Posalux etc.).



Chapter 9 Job>steps>chk (Checklists)

Required for GenFlex 6.4

Header File for each checklist

Each checklist has header file in a new file called **hdr** in the directory **<job_name>/steps/<step_name>/chk/<checklist_name>.**
The file should have parameter **SAVE_APP** = <Application_name>
Application names for example Genesis, GenFlex, InCAM, etc.
Each application should recognize the parameter and decide if the encrypted checklist may be modified and saved.

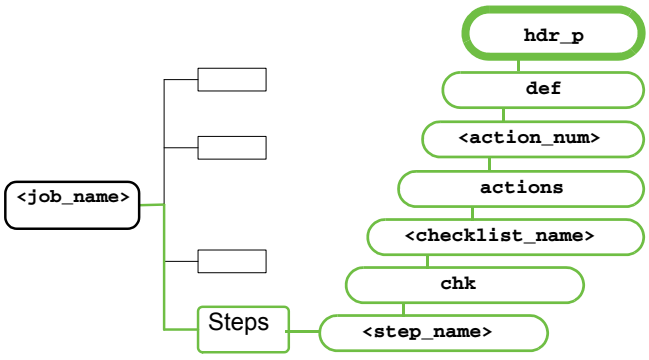
Encrypted Checklists

All created or updated checklists (include VALOR checklists) will be encrypted and saved with names ended by "_p" (example: hdr_p) exactly as VALOR does.

S def/hdr_p

Type:	Encrypted
Compression:	None
Sum file:	Yes
<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/def/hdr_p	

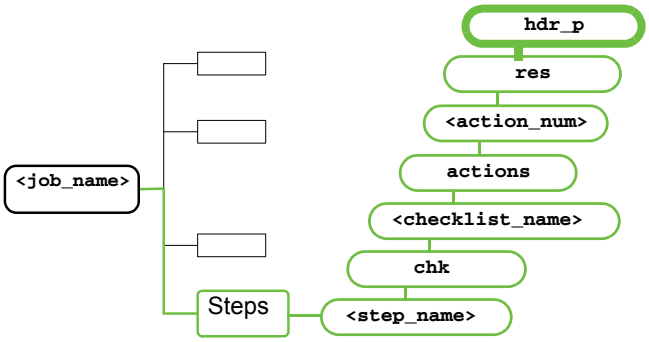
This file contains encrypted information relating to the default values of the action parameters. It also determines the location of the action in the checklist.



S res/hdr_p

Type:	Encrypted
Compression:	None
Sum file:	Yes
`<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/hdr_p`	

This file contains encrypted information relating to the results of a successful run of an action. It includes the status, date, time and duration of run, a list of generated categories with their characteristics and a list of result attributes generated by the action run.



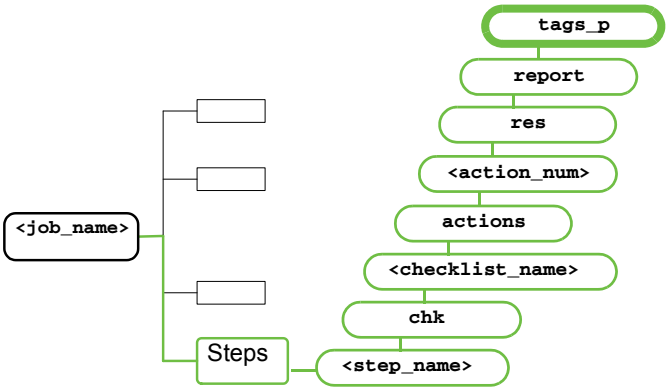
S report/tags_p

Type:	Line Record Text
Compression:	None
Sum file:	Yes
`<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/report/tags_p`	

The tags file is an optional file which links lines in the textual report with graphical locations in the associated step.

Example

```
2807 2851 0 -4 0
1237
2851 2900 -0.4 -
3.6 0 1238
2900 2949 -0.4 -
1.4 0 1239
2949 2997 4.1 -1.4
0 1240
2997 3043 3.7 -1 0 1241
3043 3091 4.1 -3.6 0 1242
...
```



Each line in the tags file has the following format:

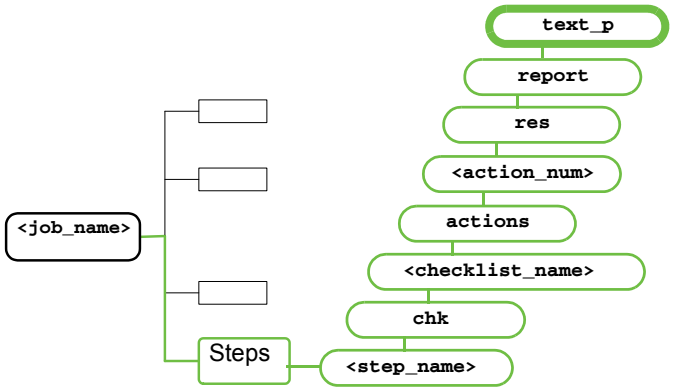
```
<start_pos> <end_pos> <x> <y> <sres_num> <meas_num>
```

start_pos, end_pos	Start and end character position of the tag in the report
x,y	Graphic location, in inches
sres_num	The number of the sub-result within the action result (Each result directory contains one or more sub-result directories)
meas_num	The number of the measurement within the measurement of the sub result.

S *report/text_p*

Type:	Encrypted
Compression:	No
Sum file:	Yes
<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/report/text_p	

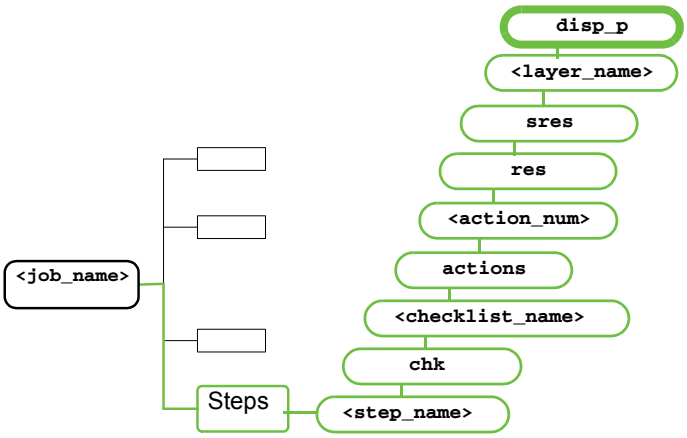
This file contains encrypted information relating to the free text report created by the analysis or DFM action.



S disp_p

Type:	Structured Text
Compression:	None
Sum file:	Yes
`<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/disp_p`	

This file contains display records. Display records are groups of layers which will be displayed when a certain measurement is selected. Each measurement (see below) may refer to a display record from this file.



Example

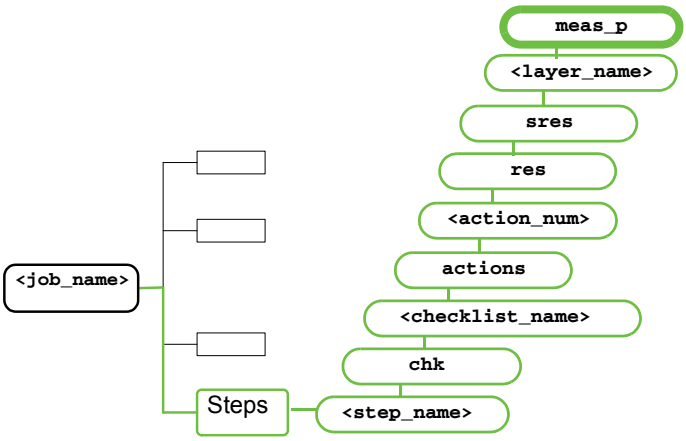
```
DISPLAY {
   L0 = sigt
}
DISPLAY {
   L0 = sigt
   L1 = drill
}
DISPLAY {
   L0 = sigt
   L1 = rout
}
```

The file includes an array of **DISPLAY** records. Each record contains a variable (up to 4) number of layers, tagged L0 to L3. Each layer should reference an existing layer in the job matrix. Nonexisting layers are ignored.

meas_p

Type:	Encrypted
Compression:	None
Sum file:	Yes
`<job_name>/steps/<step_name>/chk/<checklist_name>/actions/<action_num>/res/sres/<layer_name>/meas_p`	

This file contains encrypted information relating to all the measurements which were created for an Action in one sub-result (one layer), for all categories.



New in 7.0

As of ODB++ version 7.0, checklist measurements generated in millimeters (microns) can be read due to the introduction of a 'units' header at the beginning of the results file in the format:

U<INCH|MM>

These units affect the measurement ID of features and shape coordinates. It does not affect a measurement value which is scalar.

Example:

```
#
#Units
#
U MM

0 10 0 N P rect120x250                  #feature measurement in millimeters
S RC 2.95 1.65 0.12 0.25                #shape coordinates in millimeters
1 10 0 N P rect4.724x9.843_30
S RC 2.55 2.45 0.22 0.27
2 10 0 N P rect4.724x9.843_60
S RC 3.74 2.37 0.27 0.22
```

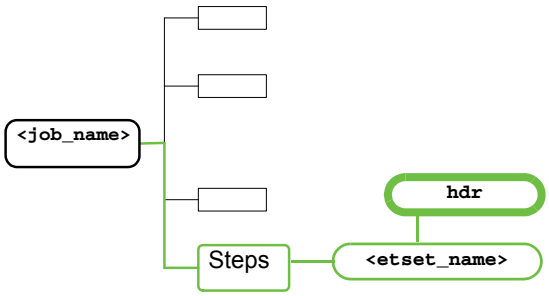
Chapter 10 Job>Steps>et (Electrical Test)

S <etset_name>/hdr

Type:	Structured Text
Compression:	None
Sum file:	No
Path	**<job Name>/steps/<step_name>/et/<etset_name>/hdr**

All the coordinates in this section are taken to be board coordinates. Within ETM (Electrical Test Manager) we use two coordinate systems: board and adapter coordinates. Board coordinates are the coordinate system found throughout Enterprise / Genesis, whereas Adapter coordinates refer to the coordinate system as they should appear within the adapter.

This file provides general information at the **etset** level.



Example

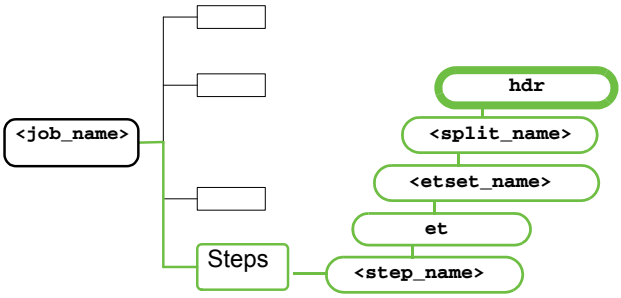
```
X_DATUM=16.33848897637795
Y_DATUM=12.99838996062992
WIDTH=7.322835039370079
HEIGHT=5.960630019685039
STATUS=new
NET_STATUS=undefined
```

X_DATUM	X-coordinate of the datum (inches)
Y_DATUM	Y- coordinate of the datum (inches)
WIDTH	Total width of the data (inches)
HEIGHT	Total height of the data (inches)
STATUS	New/ready/not_ready
NET_STATUS	Undefined/ok/disrupted

S <split_name> / hdr

Type:	Structured Text
Compression:	None
Sum file:	No
Path	**<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/hdr**

This file provides general information at the split level.



Example

```
ACCESS=grid_grid
ADAPTER=mct_tws
TESTER=mania
STYLE=mania
TRANS_NET2A {
    XC=0
    YC=0
    XOFF=-4.905
    YOFF=-9.633749999999999
    ROTATE=0
    MIRROR=no
}

TRANS_A2NET {
    XC=0
    YC=0
    XOFF=-4.905
    YOFF=-9.633749999999999
    ROTATE=0
    MIRROR=no
}

READY_STATE {
    SPLIT_READY=yes
    NETS_NOT_CHANGED=yes
    ALL_PINS_ASSIGNED=no
    P2G_NOT_CHANGED=yes
    PLATES_READY=no
    FA_DONE=no
    MAP_DONE=no
    DRILL_FILES_READY=no
    OUTPUT_DONE=yes
}

READY=no
COMPONENT_SIDE_UP=no
CHANGED=no

NC_SET_PARAMETERS {
    TYPE=1
```

```
    NAME=format
    VALUE=excellon
}

NC_SET_PARAMETERS {
    TYPE=3
    NAME=test_ar_above
    VALUE=135.0
}

NC_SET_PARAMETERS {
    TYPE=4
    NAME=stagger
    VALUE=yes
}

NC_SET_PARAMETERS {
    TYPE=4
    NAME=finished_drills
    VALUE=yes
}

OUT_FORMAT=mania_b640
CAR_FORMAT=tws2000

ADAPTER_POS {
    REFFERS=profile
    ALIGN=5
}
```

This file contains nine global parameters and four arrays.
The global parameters are:

ACCESS	At present only flying probe and universal (grid) testers are supported. Hence the options are: NO_TEST/FP/ TOP_GRID/BOT_GRID/GRID_GRID
ADAPTER	User name for adapter
TESTER	mania/ everett charles/ circuitline/ luther maelzer/ probot/ bsl/ integritest/ microcraft/ atg
STYLE	regular/ mania (meaning that the pin can bend)
READY	No/Yes - for internal use
COMPONENT_SIDE_UP	No/Yes
CHANGED	No/Yes - for internal use
OUT_FORMAT	mania_b640/evc/circuit_line/lm-udl/probot/bsl/ integritest/ microcraft/atf/tif/tti/anf/ ipc356/ipcd-356a
CAR_FORMAT	None/epc/tws2000

The arrays **TRANS_NET2A** and **TRANS_A2NET** describe the transformation when converting from Board coordinate system to Adapter coordinate system and vice-versa. Their fields are the standard transformation fields, which are:

XC	X-coordinate of the centre of the transformation
YC	Y-coordinate of the centre of the transformation
XOFF	X-offset of the transformation
YOFF	Y-offset of the transformation
ROTATE	(0/1/2/3) x 90°
MIRROR	No/Yes

The fields of the array **READY_STATE** are (for internal use only):

SPLIT_READY	No/Yes
NETS_NOT_CHANGED	No/Yes
ALL_PINS_ASSIGNED	No/Yes
P2G_NOT_CHANGED	No/Yes
PLATES_READY	No/Yes
FA_DONE	No/Yes
MAP_DONE	No/Yes
DRILL_FILES_READY	No/Yes
OUTPUT_DONE	No/Yes

The fields of the array **OUTPUT_PARAMETRS** and **NC_SET_PARAMETRS** are:

TYPE	0-empty/ 1-text/ 2-integer/ 3-double/ 4-boolean
NAME	Parameter name
VALUE	Parameter Value

The fields of the array **ADAPTER_POS** are:

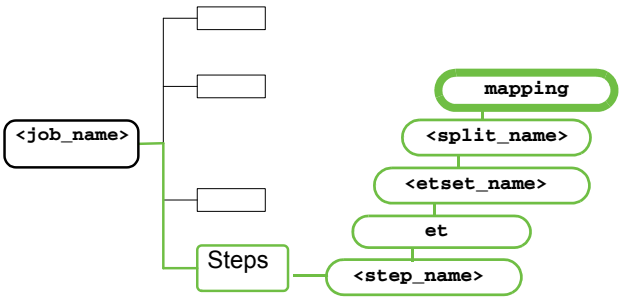
REFFERS	Profile/ net_limit
ALIGN	0-empty, 1-upper-left, 2-upper-mid, 3-upper-right, 4-mid-left, 5-mid-mid, 6-mid-right, 7-lower-left, 8-lower-mid, 9-lower-right

S *<split_name> / mapping*

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/mapping**

This file gives general information at the split level.

Example




```
   #
# Mapping File
#
#                                  GNET GRID      GRID      ASSIGN TEST    TEST
#  ID     X         Y       PIN    NUM IND X     IND Y     TO     ALLOWED FROM
#
    0  6.100000  5.449000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 0
    1  6.100000  5.349000 39003d    0  6.095000  5.366250 TOP    TOP       TOP       ON_AR N NET# 0
    2  6.000000  5.324000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 0
    3  6.050000  5.349000 39003d    0  5.995000  5.366250 TOP    TOP       TOP       ON_AR N NET# 0
    4  7.050000  5.449000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 1
    5  7.050000  5.349000 39003d    0  7.095000  5.266250 TOP    TOP       TOP       ON_AR N NET# 1
    6  6.938000  5.424000 39003d    0  6.995000  5.466250 BOT    BOT       BOT       ON_AR N NET# 1
    7  6.450000  6.324000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 2
    8  6.350000  6.324000 39003d    0  6.295000  6.366250 TOP    TOP       TOP       ON_AR N NET# 2
    9  7.375000  6.324000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 3
   10  7.275000  6.324000 39003d    0  7.195000  6.366250 TOP    TOP       TOP       ON_AR N NET# 3
   11  7.050000  6.874000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 4
   12  7.050000  6.949000 39003d    0  7.095000  6.866250 TOP    TOP       TOP       ON_AR N NET# 4
   13  6.600000  6.899000 39003     0  6.595000  6.866250 BOT    BOT       BOT       ON_AR N NET# 4
   14  6.100000  6.874000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   15  6.100000  6.949000 39003d    0  6.095000  6.966250 TOP    TOP       TOP       ON_AR N NET# 5
   16  8.950000  5.824000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   17  9.050000  5.424000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   18  8.750000  5.824000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   19  8.450000  5.924000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   20  8.300000  5.674000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   21  9.150000  5.124000 NO_PIN    0  0.000000  0.000000 NONE   BOTH      NO        ON_AR N NET# 5
   22  9.675000  5.674000 39003     0  9.695000  5.666250 TOP    TOP       TOP       ON_AR N NET# 5
```

Each row has the following structure

```
<id> <board x> <board y> <pin> <grid num> <grid x> <grid y> <assign> <testable
   side> <side tested> <on annular ring> <net number>
```

Id	ID number of the point
Board x,board y	Location of the test point on the board
Pin	Pin name assigned to this point
Grid num	The number of the grid structure used
Grid x,grid y	The grid position assigned to this point
Assign	The grid side to whom the point is assigned
testable side	The side from where the point can be tested
side tested	The side from where the point should be tested

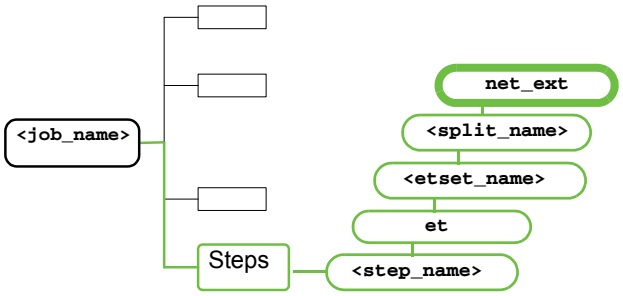
on annular ring	Whether the test point is on the annular ring
net number	Corresponds to the number in the netlist

All the sides here refer to the board side (i.e. Top is the component side etc.)

S <split_name> / net_ext

Type:	Line Record Text
Compression:	None
Sum file:	No
Path	**<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/net_ext**

This file gives general information regarding the test status of each net in this specific split.



Example

```
ET_NET {
    NETNUM=0
    TYPE=all
}

ET_NET {
    NETNUM=1
    TYPE=all
}

ET_NET {
    NETNUM=2
    TYPE=shorts
}
...
```

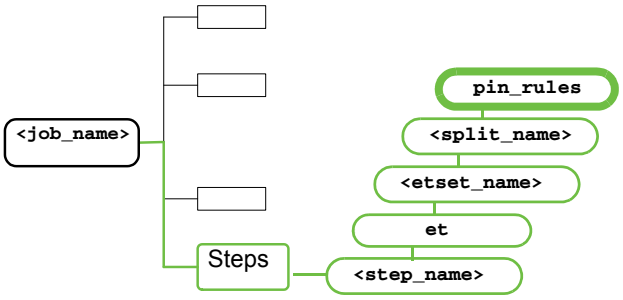
Each array **ET_NET** has the following structure:

<NETNUM>	Number of the net as in the netlist
<TYPE>	NO_TEST/SHORTS/CONNECT/ALL

S <split_name> / pin_rules

Type:	Structured Text
Compression:	None
Sum file:	No
Path	**<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/pin_rules**

This file lists the rules to be used in assigning pins. The pins themselves are defined in the **<pin_name>** section.



Example

```
RULE {
    NAME=39003
    TYPE=def_smd
    SIDE=both
    PITCH_MIN=0
    PITCH_MAX=0
    SIZE_MIN=0
    SIZE_MAX=0
    DIST_X=0
    DIST_Y=0
    NUM_NEEDED=0
    EQUALS_TO=1
    CONTACT_MIN=0
    ALIGN_MIN=0
    ALIGN_MAX=0
```

This file is made up of repetitions of the **RULE** array;

NAME	Name used to refer to the pin
TYPE	def_smd/ def_pin/ def_over/ smd/ hole/ npth/ comp/ air/ tool/ fp_info/ alignment
SIDE	Side as determined by the placement in the adapter (both/top/ bot)
PITCH_MIN	Minimum pitch required for this pin
PITCH_MAX	Maximum pitch required for this pin
SIZE_MIN	Minimum contact size for the pin head
SIZE_MAX	Maximum contact size for the pin head
DIST_X ,DIST_Y	Distance between air-holes
NUM_NEEDED	Minimum number required
EQUALS_TO	Number of pins equivalent to a compensation post
CONTACT_MIN	Contact size for flying probe
ALIGN_MIN	Minimum number of alignment points for flying probe output
ALIGN_MAX	Maximum number of alignment points for flying probe output

This table shows which field is relevant to each type

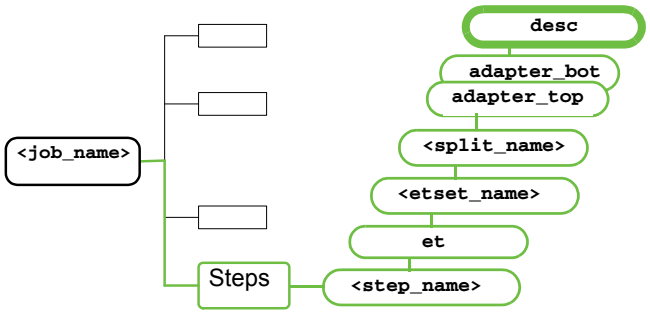
TYPE	NAME	SIDE	PITCH	SIZE	DIST	NUM_NEEDED	EQUALS_TO	CONTACT	ALIGN
def_smd	x	x							
def_pin	x	x							
def_over	x	x							
smd	x	x	x	x					

hole	x		x		x									
npth	x		x		x		x							
comp	x		x						x					
air	x		x			x								
tool	x		x				x							
fp_info				x		x							x	
alignment														x

adapter_top(bot) / desc

Type:	Structured Text
Compression:	None
Sum file:	No
<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/adapter_top(bot)/desc	

This file provides general information describing the build-up of the adapter.



Example

```
X_MIN=-2.605
Y_MIN=-0.93375
X_MAX=18.295
Y_MAX=13.56625

GRID {
   STEP_X=0.1
   STEP_Y=0.1
   X_MIN=-1.705
   Y_MIN=-0.03375
   X_MAX=17.395
   Y_MAX=12.66625
}
HEIGHT=3.75

PLATES {
   ELEVATION=0.1125
   THICKNESS=0.12
   COUNTER_SINK_TOP_H=0
   COUNTER_SINK_TOP_R=0
   COUNTER_SINK_BOT_H=0
   COUNTER_SINK_BOT_R=0
   PLATE_NAME=tus
   CONST_DRILL=tus.3
   SPEC_PROCESS=none
   MASK_SIZE=0
   GUIDING_MAX_PITCH=0
   GUIDING_DEFL_LIMITS=0
   GUIDING_SMALL_SIZE=0
   GUIDING_BIG_SIZE=0
```

```
    GUIDING_MARGIN=0

    TRANS_PLATE {
        XC=0
        YC=0
        XOFF=0
        YOFF=0
        ROTATE=0
        MIRROR=no
    }
}

TOOLS {
    NAME=man_tool2
    X=16.9298279527559
    Y=13.91177598425197
}
```

Note See "Units of Measurement" on page 19.

This file contains five global parameters and three arrays. The global parameters are:

X_MIN, Y_MIN	Bottom left corner of the adapter
X_MAX, Y_MAX	Top right corner of the adapter
HEIGHT	Vertical distance from the grid to the board

Numerous grids can be defined. For example double density grids require two definitions of grids each with a step of 100 mil with a 50 mil step between them.

The fields of the **GRID** array are:

STEP_x, STEP_Y	Grid step in inches
X_MIN, Y_MIN	Bottom left corner of the grid
X_MAX, Y_MAX	Top right corner of the grid

The **PLATES** array has a sub array, **TRANS_PLATE**. The fields of the **PLATES** array are:

ELEVATION	Distance from the board to the plate
THICKNESS	Thickness of the plate
COUNTER_SINK_TOP_H	Counter sink depth for the upper side
COUNTER_SINK_TOP_R	Counter sink drill holes on top layer with a radius less than this value
COUNTER_SINK_BOT_H	Counter sink depth for the lower side
COUNTER_SINK_BOT_R	Counter sink drill holes on bottom layer with a radius less than this value
PLATE_NAME	Name of the plate
CONST_DRILL	Name of the layer from which the constant drills are taken

SPEC_PROCESS	NONE/ SHOULDER/ MASK/ GUIDING
MASK_SIZE	Standard drill size used for the mask layer
GUIDING_MAX_PITCH	Maximum pitch required for detecting rows for guiding plate
GUIDING_DEFL_LIMITS	Below this value, **GUIDING_SMALL_SIZE**. is used
GUIDING_SMALL_SIZE	Standard drill size for pins with a small deflection
GUIDING_BIG_SIZE	Standard drill size for pins with a large deflection
GUIDING_LINE_SIZE	Size of the routing line
GUIDING_MARGIN	Additional border width around the component area

The fields of the sub array **TRANS_PLATE** (the transformation parameters for each plate to be used in the output of the drill file) are:

XC,YC
XOFF,YOFF
ROTATE
MIRROR

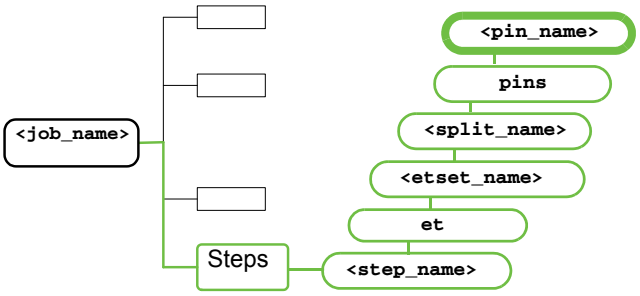
The fields of the **TOOLS** array are:

NAME	Name of the tooling pin
X,Y	Its position on the board

S *pins / <pin_name>*

Type:	Structured Text
Compression:	None
Sum file:	No
<job Name>/steps/<step_name>/et/<etset_name>/<split_name>/pins/<pin_name>	

For each pin there should be the following structure to describe its setup.



Example

```
NAME=39003d
TYPE=probe
SYMBOL=oval24.0x24
.0
HEIGHT=3.75
DEFLECTION=0.3
MAN_DEFLECTION=0.3
BOARD_SNAP=pad
GRID_SNAP=grid
CRIMP=0
```

```
MEASURES {
   OFFSET=0
   DIAMETER=0.024
   SPACING=0
   DRILL_SIZE=0.029
}

MEASURES {
   OFFSET=1
   DIAMETER=0.024
   SPACING=0
   DRILL_SIZE=0.029
}

MEASURES {
   OFFSET=1
   DIAMETER=0.024
   SPACING=0
   DRILL_SIZE=0.045
}

MEASURES {
   OFFSET=3.75
   DIAMETER=0.024
   SPACING=0
   DRILL_SIZE=0.045
}
```

This file contains nine global parameters and one array. The global parameters are:

NAME	Name of the pin
TYPE	probe/tooling/align_pt
SYMBOL	Name of the symbol used to represent the pin in the display
HEIGHT	Total pin height
DEFLECTION	Normal maximum deflection allowed
MAN_DEFLECTION	Maximum deflection possible
BOARD_SNAP	pad/ npth/ empty/ none
GRID_SNAP	grid/ gnone
CRIMP	Not in use

The fields of the **MEASURES** array are:

OFFSET	Distance from the pin head
DIAMETER	Cross-sectional diameter of the pin at this offset
SPACING	Required spacing for the pin at this offset
DRILL_SIZE	Recommended drill size (if zero then the drill size is calculated)

There is also another tree, which has some of the above files.

Under **/genesis/sys/hooks/** there is the "**et**" directory. Here are stored a library of pins and adapters that are generally available.

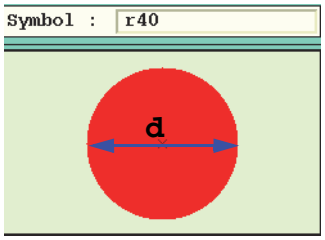
et *(scripts specifically used in etm)*
pins *(parent pin library)*
<pin_name> *(explained above)*
adapters
<adapter_name>
hdr *(explained above)*
pin_rules *(explained above)*
adapter_top
desc *(explained above)*
const_drill
<layer_name> **(***according to standard definitions)*
adapter_bot
desc *(explained above***)**
const_drill
<layer_name>*(according to standard definitions)*
pins *(pins defined with rules for each adapter)*
<pin_name> *(explained above)*

Chapter 11 Symbol Definitions

Standard Symbols

The system supports the following standard (system) symbols:

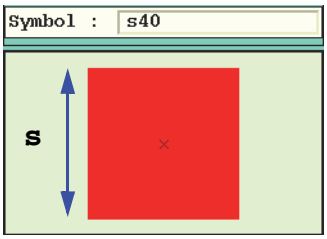
Round



r<d>

d - circle diameter

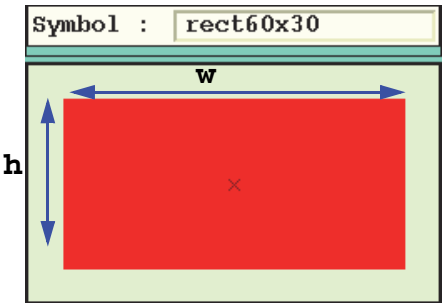
Square



s<s>

s - square side

Rectangle

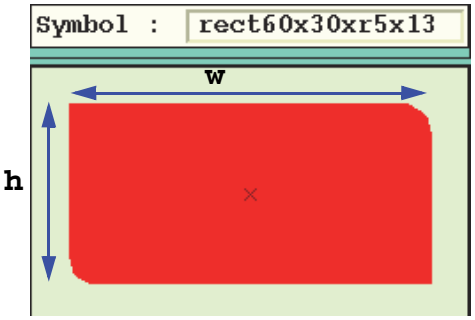


rect<w>x<h>

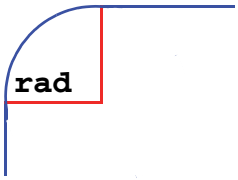
w - rectangle width
h - rectangle height

Rounded Rectangle

rect<w>x<h>xr<rad>x<corners>

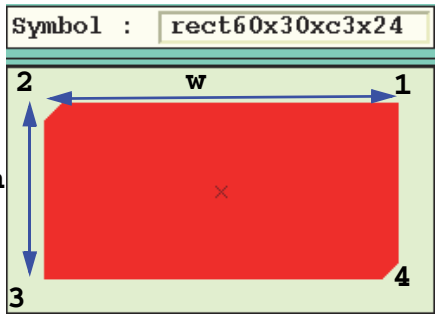


w - rectangle width
h - rectangle height
rad - corner radius
corners - a combination of which corners are rounded.
x<corners> is omitted if all corners are rounded.

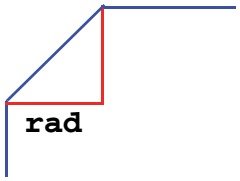


Chamfered Rectangle

rect<w>x<h>xc<rad>x<corners>

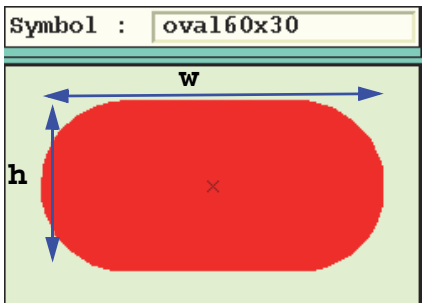


w - rectangle width
h - rectangle height
rad - corner radius
corners - a combination of which corners are rounded.
x<corners> is omitted if all corners are rounded.



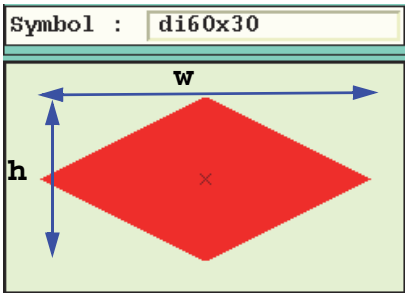
Oval

oval<w>x<h>



w - oval width
h - oval height

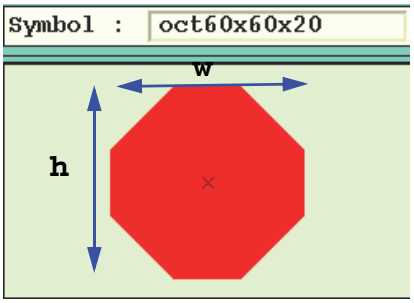
Diamond



di<w>x<h>

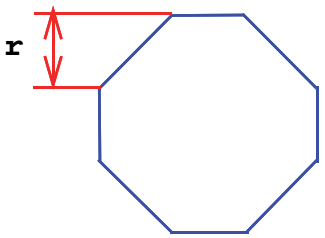
w - diamond width
h - diamond height

Octagon

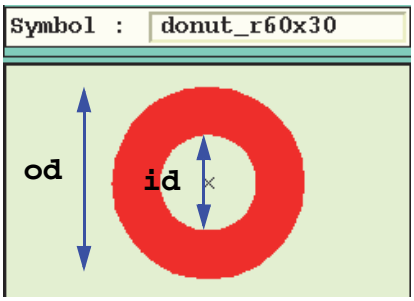


oct<w>x<h>x<r>

w - octagon width
h - octagon height
r - corner size



Round Donut

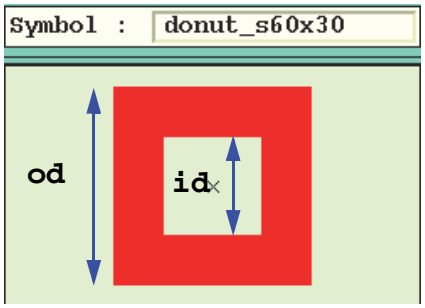


donut_r<od>x<id>

od - outer diameter
id - inner diameter

Square Donut

donut_s<od>x<id>

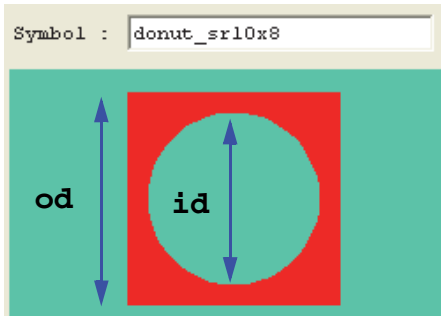


od - outer diameter
id - inner diameter

New in 7.0

Square/Round Donut

donut_sr<od>x<id>

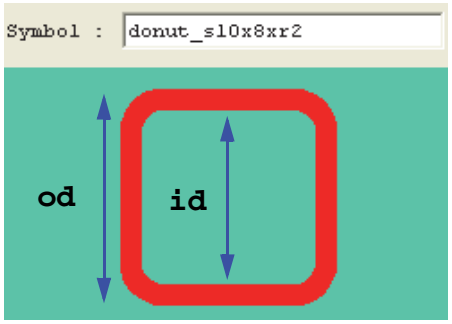


od - outer diameter
id - inner diameter

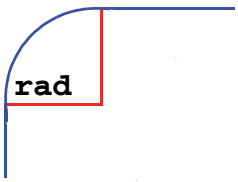
New in 7.0

Rounded Square Donut

donut_s<od>x<id>x<rad>x<corners>

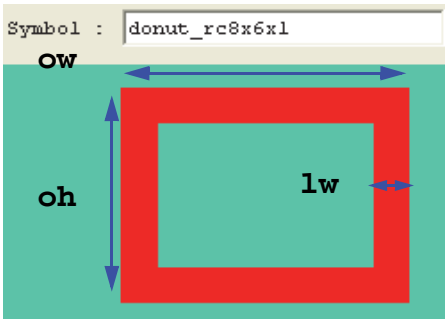


od - outer diameter
id - inner diameter
rad - corner radius
corners - a combination of which corners are rounded.
x<corners> is omitted if all corners are rounded.



New in 7.0

Rectangle Donut

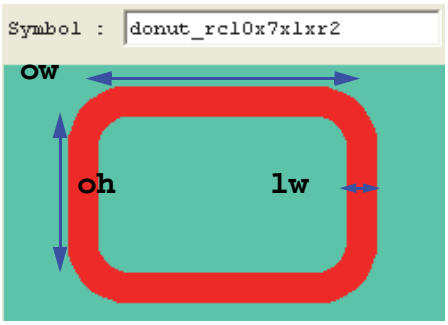


donut_rc<ow>x<oh>x<lw>

ow - outer width
oh - outer height
lw - line width

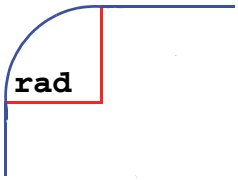
New in 7.0

Rounded Rectangle Donut



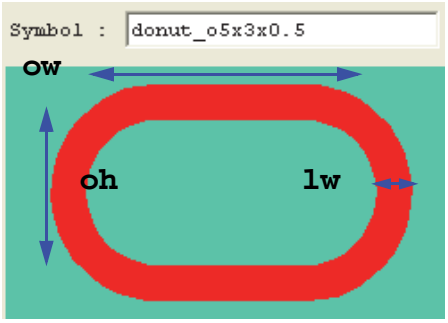
donut_rc<ow>x<oh>x<lw>x<rad>x<corners>

ow - outer width
oh - outer height
lw - line width
rad - corner radius
corners - a combination of which corners are rounded.
x<corners> is omitted if all corners are rounded.



New in 7.0

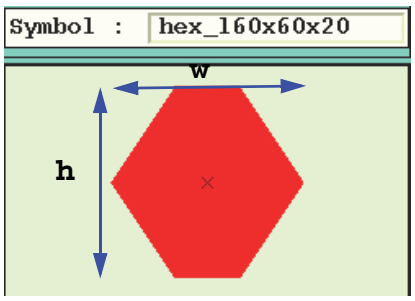
Oval Donut



donut_o<ow>x<oh>x<lw>

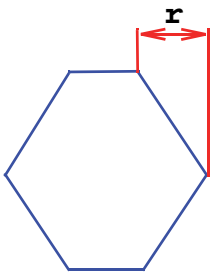
ow - outer width
oh - outer height
lw - line width

Horizontal Hexagon

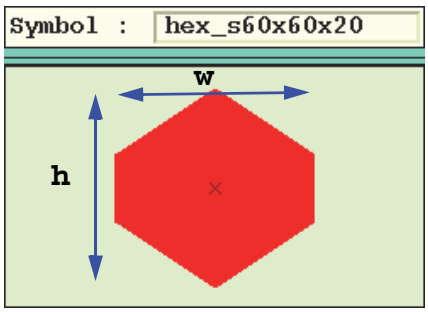


hex_l<w>x<h>x<r>

w - hexagon width
h - hexagon height
r - corner size

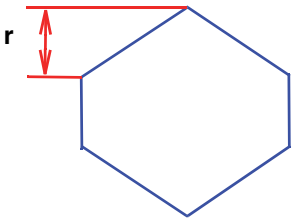


Vertical Hexagon

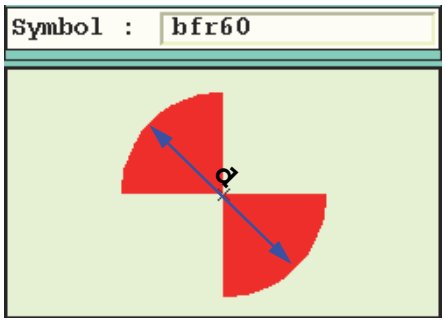


hex_s<w>x<h>x<r>

w - hexagon width
h - hexagon height
r - corner size



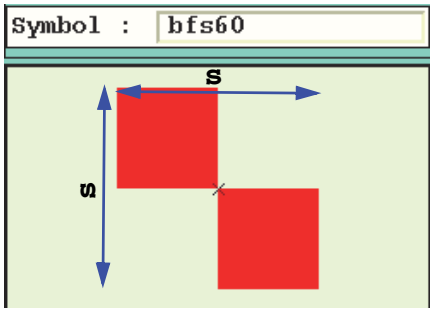
Butterfly



bfr<d>

d - diameter

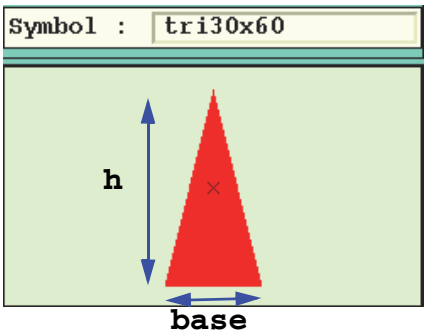
Square Butterfly



bfs<s>

s - size

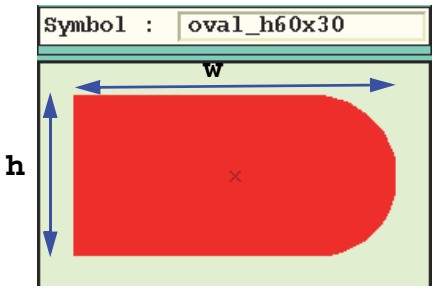
Triangle



tri<base>x<h>

base - triangle base
h - triangle height

Half Oval

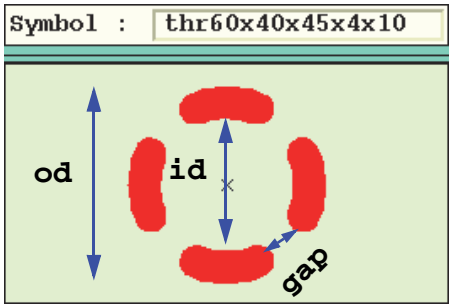


oval_h<w>x<h>

w - width
h - height

Round Thermal (Rounded)

thr<od>x<id>x<angle>x<num_spokes>x<gap>

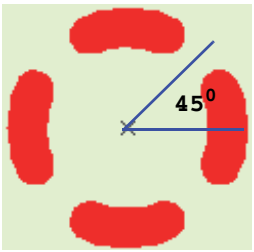


num_spokes = 4

od - outer diameter
id - inner diameter
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap

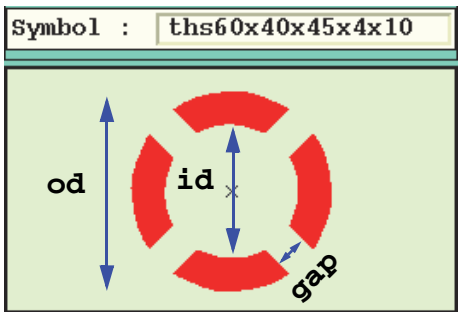
Specification of **od** and **id** determine the air gap (size of laminate separation)

angle



Round Thermal (Squared)

ths<od>x<id>x<angle>x<num_spokes>x<gap>

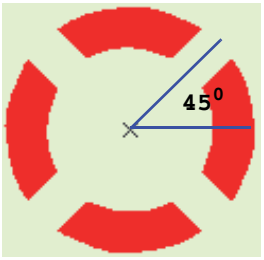


num_spokes = 4

od - outer diameter
id - inner diameter
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap

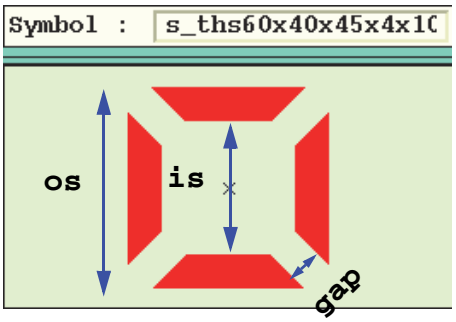
Specification of **od** and **id** determine the air gap (size of laminate separation)

angle



Square Thermal

s_ths<os>x<is>x<angle>x<num_spokes>x<gap>

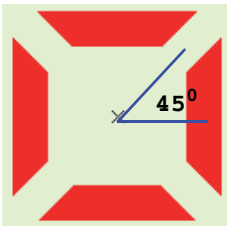


num_spokes = 4

os - outer size
is - inner size
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap

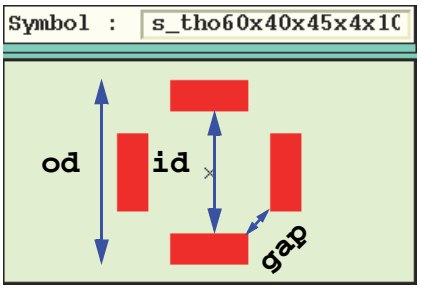
Specification of **os** and **is** determine the air gap (size of laminate separation)

angle



Square Thermal (Open Corners)

s_tho<od>x<id>x<angle>x<num_spokes>x<gap>

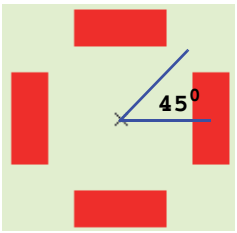


num_spokes = 4

od - outer diameter
id - inner diameter
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap

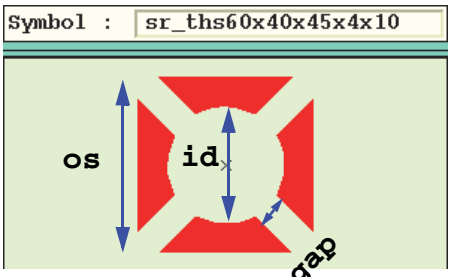
Specification of **od** and **id** determine the air gap (size of laminate separation)

angle



Square-Round Thermal

sr_ths<os>x<id>x<angle>x<num_spokes>x<gap>

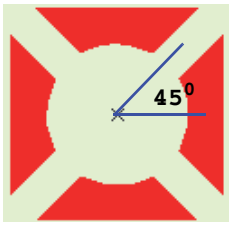


num_spokes = 4

os - outer size
id - inner diameter
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap

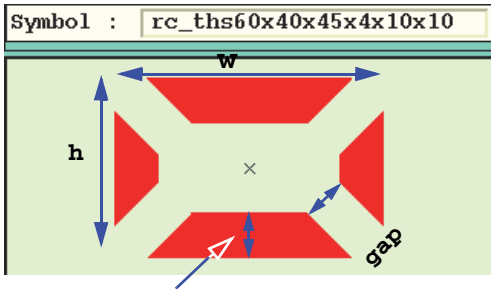
Specification of **os** and **id** determine the air gap (size of laminate separation)

angle



Rectangular Thermal

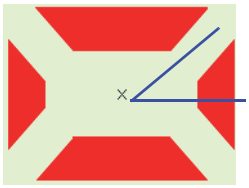
rc_ths<w>x<h>x<angle>x<num_spokes>x<gap>x<air_gap>



air gap num_spokes = 4

w - outer width
h - outer height
angle - gap angle from 0^{0*}
num_spokes - number of spokes
gap - size of spoke gap
air_gap - size of laminate separation

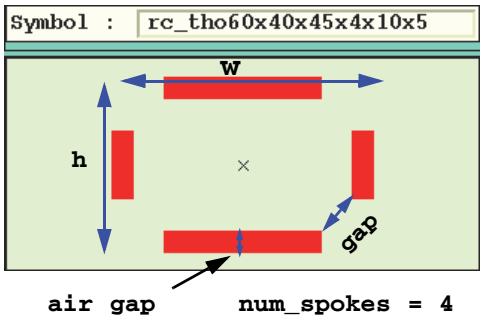
angle



* **angle** is limited to multiples of 45 degrees.

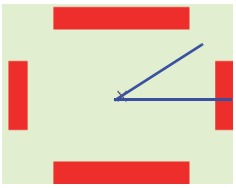
Rectangular Thermal (Open Corners)

rc_tho<w>x<h>x<angle>x<num_spokes>x<gap>x<air_gap>



w - outer width
h - outer height
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap
air gap - size of laminate separation

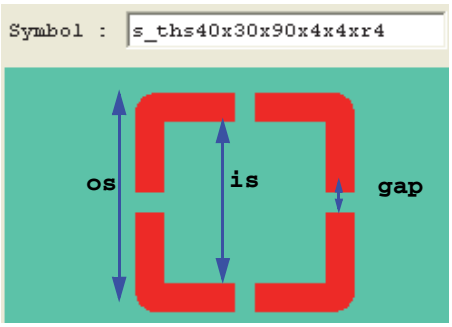
angle



New in 7.0

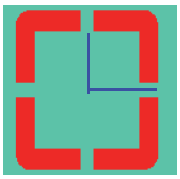
Rounded Square Thermal

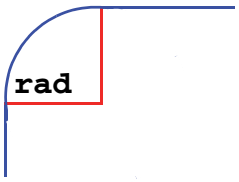
s_ths<os>x<is>x<angle>x<num_spokes>x<gap>x<r<rad>x<corners>



os - outer size
is - inner size
angle - gap angle angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap
rad - corner radius
corners - a combination of which corners are rounded. **x<corners>** is omitted if all corners are rounded.

angle

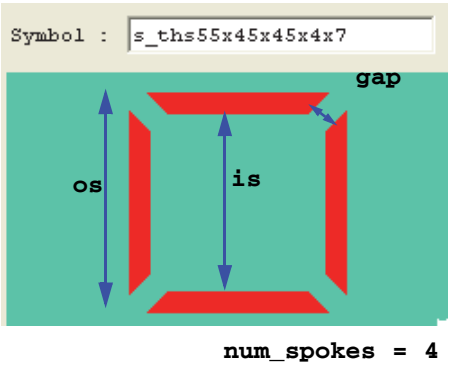




New in 7.0

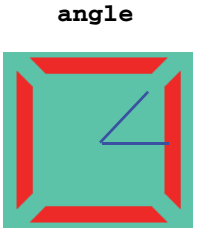
Rounded Square Thermal (Open Corners)

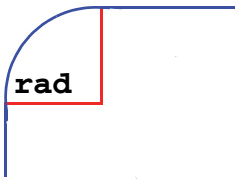
s_ths<os>x<is>x<angle>x<num_spokes>x<gap>x r<rad>x<corners>



num_spokes = 4

os - outer size
is - inner size
angle - gap angle from 45^0
num_spokes - number of spokes
gap - size of spoke gap
rad - corner radius
corners - a combination of which corners are rounded. **x<corners>** is omitted if all corners are rounded.

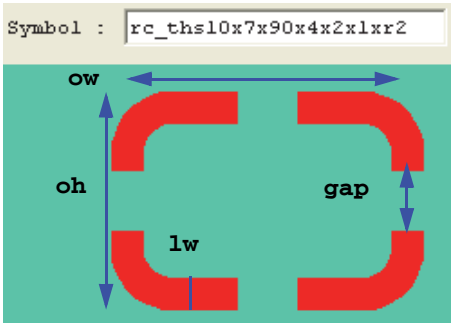




New in 7.0

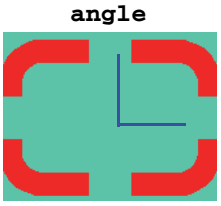
Rounded Rectangle Thermal

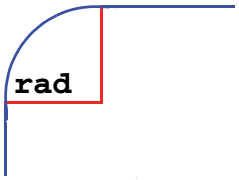
rc_ths<ow>x<oh>x<angle>x<num_spokes>x<gap>x <lw>xr<rad>x<corners>



num_spokes = 4

ow - outer width
oh - outer height
lw - line width
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap
rad - corner radius
corners - a combination of which corners are rounded.
x<corners> is omitted if all corners are rounded.

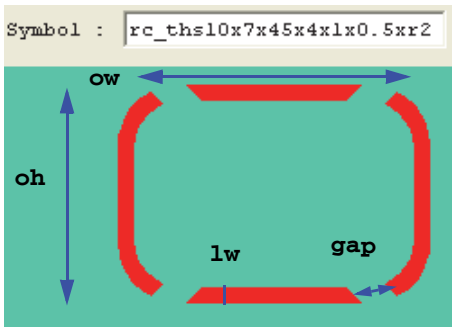




New in 7.0

Rounded Rectangle Thermal (Open Corners)

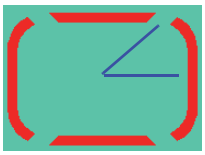
rc_ths<ow>x<oh>x<angle>x<num_spokes>x<gap>x<lw>xr<rad>x<corners>

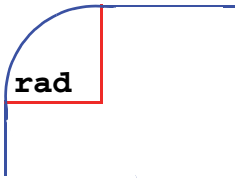


num_spokes = 4

ow - outer width
oh - outer height
lw - line width
angle - gap angle of 45^0
num_spokes - number of spokes
gap - size of spoke gap
rad - corner radius
corners - a combination of which corners are rounded. **x<corners>** is omitted if all corners are rounded.

angle

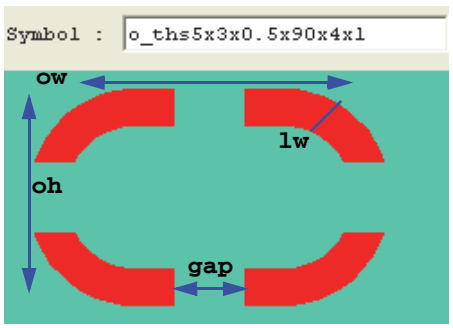




New in 7.0

Oval Thermal

o_ths<ow>x<oh>x<angle>x<num_spokes>x<gap>x<lw>



num_spokes = 4

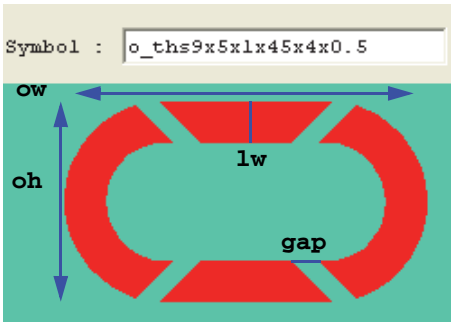
ow - outer width
oh - outer height
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap
lw - line width

angle

New in 7.0

...al Thermal (Open Corners)

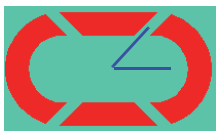
o_ths<ow>x<oh>x<angle>x<num_spokes>x<gap>x<lw>



num_spokes = 4

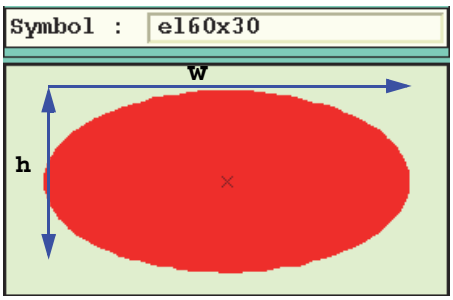
ow - outer width
oh - outer height
angle - gap angle from 0^0
num_spokes - number of spokes
gap - size of spoke gap
lw - line width

angle



Ellipse

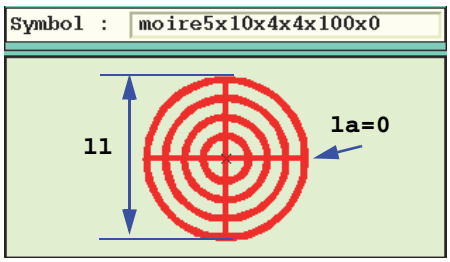
el<w>x<h>



w - width
h - height

Moire

moire<rw>x<rg>x<nr>x<lw>x<ll>x<la>

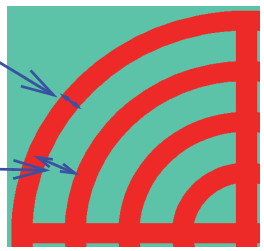


number of rings (nr)=4

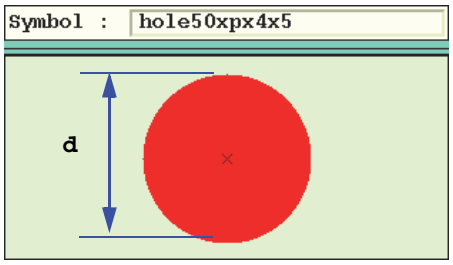
rw - ring width
rg - ring gap
nr - number of rings
lw - line width
ll - line length
la - line angle

ring width (rw)

ring gap (rg) from center of ring to edge of adjacent ring



Hole



hole<d>x<p>x<tp>x<tm>

d - hole diameter
p - plating status (p(lated), n(on-plated) or v(ia))
tp - + tolerance
tm - - tolerance
This symbol is specifically intended for wheels created by the Wheel Template Editor for drill files.

Null

null<ext>

ext - extension number
This symbol is empty and used as a place holder for non-graphic features.

Rotated Standard Symbols

Prior to V8.0, pads and text could only be rotated in increments of 90 degrees. Angles other than 0, 90, 180 or 270 were considered special symbols, which increase feature data stored in the job, and slow down the data loading process. As of V8.0, the rotation of pads and text at any angle is allowed.

However, for versions prior to V8.0, to create symbols that are rotated at angles that are not in the standard 90 degree increments, a symbol is required to be created for each angle. For example, **rect25x50_315** is a standard rectangle rotated around its center to 315 degrees. These symbols are created automatically when specified by name when adding a pad, for example. They can also be resized as needed.

A corresponding feature file containing a single contour representing the symbol must be placed in:

<job_name>/symbols/<symbol_name>/features

Note Rotation is clockwise.

Appendix A System Attributes

The following table is a combined list of the system attributes currently used by the Enterprise and Trilogy programs. As of V8.0, system attributes are not considered core entities. (See "Entity Definitions" on page 17.) Therefore, for Genesis products, see "System Attributes for Genesis" on page 261.

The columns are:

Attributes - the internal name of the attribute

Type - Float, Integer, Boolean, Text, Option

Entity - the ODB++ entity (such as Job, Step, Feature, Component, Symbol, Wheel, ...) to which the attribute can be assigned.

Attribute List

Attribute	Type	Entity	Description	Display Name
`.all_eda_layers`	Text	Step	(0 to 1000 characters) This attribute consists of a list of all the layers in the current EDA design (not ODB++). It used in the graphic synchronization with the EDA system.	
`.aoi_cpbm`	Integer	Feature	(0 to 255) Obsolete	
`.aoi_cpcu`	Integer	Feature	(0 to 255) Obsolete	
`.aoi_drbm`	Integer	Feature	(0 to 255) Obsolete	
`.aoi_drcu`	Integer	Feature	(0 to 255) Obsolete	
`.aoi_value`	Integer	Feature	(0 to 255) Obsolete	
`.ar_pad_drill_bottom_max`	Float	Feature	(-10 to 1000) Assigned to a drill to define the maximum annular ring size in mils or microns between the drill and the copper of the bottom layer of the drill span.	Max Pad AR Drill Bottom

Attribute	Type	Entity	Description	Display Name
.ar_pad_drill_bottom_min	Float	Feature	(-10 to 1000) Assigned to a drill to define the minimum annular ring size in mils or microns between the drill and the copper of the bottom layer of the drill span.	Min Pad AR Drill Bottom
.ar_pad_drill_inner_max	Float	Feature	(-10 to 1000) Assigned to a drill to define the maximum annular ring size in mils or microns between the drill and the copper of an inner layer in the drill span.	Max Pad AR Drill Inner
.ar_pad_drill_inner_min	Float	Feature	(-10 to 1000) Assigned to a drill to define the minimum annular ring size in mils or microns between the drill and the copper of an inner layer in the drill span.	Min Pad AR Drill Inner
.ar_pad_drill_top_max	Float	Feature	(-10 to 1000) Assigned to a drill to define the maximum annular ring size in mils or microns between the drill and the copper of the top layer of the drill span.	Max Pad AR Drill Top
.ar_pad_drill_top_min	Float	Feature	(-10 to 1000) Assigned to a drill to define the minimum annular ring size in mils or microns between the drill and the copper of the top layer of the drill span.	Min Pad AR Drill Top
.ar_sm_drill_bottom_max	Float	Feature	(-10 to 1000) Assigned to a drill piercing the bottom layer, to define the maximum annular ring size in mils or microns between the drill and the soldermask on the bottom layer.	Max SM AR Drill Bottom
.ar_sm_drill_bottom_min	Float	Feature	(-10 to 1000) Assigned to a drill piercing the bottom layer, to define the minimum annular ring size in mils or microns between the drill and the soldermask on the bottom layer.	Min SM AR Drill Bottom

.ar_sm_drill_top_max	Float	Feature	(-10 to 1000) Assigned to a drill piercing the top layer, to define the maximum annular ring size in mils or microns between the drill and the soldermask on the top layer.	Max SM AR Drill Top
.ar_sm_drill_top_min	Float	Feature	(-10 to 1000) Assigned to a drill piercing the top layer, to define the minimum annular ring size in mils or microns between the drill and the soldermask on the top layer.	Min SM AR Drill Top
.ar_sm_pad_bottom_max	Float	Feature	(-10 to 1000) Assigned to a drill piercing the bottom layer, to define the maximum annular ring size in mils or microns between the drilled pad of the bottom layer and the soldermask above.	Max SM AR Bottom
.ar_sm_pad_bottom_min	Float	Feature	(-10 to 1000) Assigned to a drill piercing the	Min SM AR Bottom

Attribute	Type	Entity	Description	Display Name
			or microns between the drilled pad of the bottom layer and the soldermask above.	
.ar_sm_pad_top_max	Float	Feature	(-10 to 1000) Assigned to a drill piercing the top layer, to define the maximum annular ring size in mils or microns between the drilled pad of the top layer and the soldermask above.	Max SM AR Top
.ar_sm_pad_top_min	Float	Feature	(-10 to 1000) Assigned to a drill piercing the top layer, to define the minimum annular ring size in mils or microns between the drilled pad of the top layer and the solder mask above.	Min SM AR Top
.area_name	Text	Feature	[0 to 64) Assigned to surface features which are drawn in a process map layer. A process map layer is used in assembly analysis for determining the process type used in the location a measurement is found.	DFx Area Name

.array_with_rotation	Boolean	Step	(Default=No) If Yes, this step is a multi-panel array, with the same panel possibly appearing in 180-degree rotation to itself	
.artwork	Text	Feature	(0-1000) Indicating to which entity the feature belongs (component, package, net, board)	Feature Source
.assembly_proc_bottom	Text	Step	(0 to 20) Default assembly process for the bottom side, to be used when there is no specific area defined in the process map layer (or no process map layer at all)	
.assembly_proc_top	Text	Step	(0 to 20) Default assembly process for the top side, to be used when there is no specific area defined in the process map layer (or no process map layer at all).	

Attribute	Type	Entity	Description	Display Name
			Defines the angle at which a board is inserted into a 5DX machine. Values are translated as 0, 90, 180, 270 degrees.	
.bit	Text	Feature	(0 - 64) Contains the drill designator which is set to each tool in the Drill Tools Manager.	
.board_mark	Option	Feature	(bbm, gpm) In Vi-Technology output, this controls whether a step or a panel needs to be inspected: **bbm** - This feature is a bad board mark. Skip inspection of the step. **gpm** - This feature is a good panel mark. The panel can be accepted for printing without scanning its steps for bad board marks. **Note:** From V7.6 this replaces the obsolete attribute **.skip_indicator.**	
.board_thickness	Float	Job	(0.0 to 10.0; default = 0.0) Total thickness of the board expressed in inch or mm.	

.bond_name	Text	Feature	Name of the wire bond.	Name of wire bond
.bonding_profile	Text	Feature	Name of the bonding profile.	Bonding profile
.break_away	Boolean	Symbol	(Default = No) Assigned to a symbol representing a break-away to be inserted into any line or arc of the rout path. When adding a **break_away** symbol through dimensions, it automatically adjusts to the line or arc angle, breaks that feature (in the breaking points defined in that symbol with the **.brk_pnt** attribute), and adds all the necessary connections and dimensions.	
.brk_point	Boolean	Feature	Assigned to a pad or a dpoint in a break-away symbol (that was given the attribute **.break_away**). When adding the	

Attribute	Type	Entity	Description	Display Name
			line/arc is broken at the connection point with the dpoint that has the **.brk_point** attribute. In each break-away symbol there should be two points with this attribute.	
.cad_local_footprint_change	Boolean	Comp.	(Default = No) Indicates whether there has been a local change to a pad code in the local design.	CADStar Pad Change
.cad_package_name	Text	Comp.	(0-10000) Contains the full CAD package name of a Cadstar component. This name can be longer than the Valor package name which is limited to 64 characters.	CADStar Package Name
.cad_part_override	Text	Comp.	(0-64) Assigns component properties in accordance with data received from the **ASSY_PN_OVERRIDE** property.	CAD Part Varient Support
.center_fiducial	Boolean	Comp.	(Default = No) Specifies component is expected to have a fiducial at its center.	Center Fiducial Required

.color	Text	Feature; Comp.	(0 to 1000 for each color) Can be attached to any feature or component to define the color to be used in plotting a layer in HPGL-1 or 2. The format is **rrggbb** (where **r**=red, **g**=green, **b**=blue). **White - .color = "999999"** **Black - .color = "000000"** **Red - .color = "990000"** **Green - .color = "009900"** **Yellow - .color = "009999"** **Blue - .color = "000099"** **Magenta - .color = "990099"** **Cyan - .color = "999900"**	HPGL Output Color
.comment	Text	Job; Step; Layer; Wheel; Symbol; Stackup	(0 to 500) Used for general textual comments.	
.comp	Option	Feature	(none; right; left) For a chained feature, this attribute sets the offset of the cutting tool from the rout path. Three options: - None - in center of the rout path - Left - to the left of the rout path in the direction of cutting - Right - to the right of the path	
.comp_height	Float	Comp.	(0.0 to 10.0) Stores the height of the component above the board surface expressed in inch or mm.	Height

.comp_height_area	Integer	Feature; Comp.	(0.0 to 1000000.0; default=0) The Cadence Allegro interface uses this attribute to assign the same ID to a component under which there is an area with space for a shorter component, and to the feature that defines the area. This is to avoid reporting this component as violating the height restrictions of the area, and only report violations for other components that are to be placed in the area under this component.	
.comp_htol_minus	Float	Comp.	(0.0 - 10.0) Contains the minus tolerance for component height expressed in inch or mm, used for calculation of plug-in boards.	Height Tolerance Minus
.comp_htol_plus	Float	Comp.	(0.0 - 10.0) Contains the plus tolerance for component height expressed in inch or mm, used for calculation of plug-in boards.	Height Tolerance Plus
.comp_ign_spacing	Boolean	Comp.	(Default = No) This attribute, when set, disables spacing checks on a component during assembly analysis. It is used for printed components which have no actual body	Ignore during Spacing Analysis
.comp_ignore	Boolean	Comp.	(Default=No) Determines whether the component is to be ignored when calculating statistics, or during certain operations, such as Analysis.	Ignore
.comp_mount_type	Option	Comp.	(Other; SMT; THMT; PressFit) Indicates whether the component is a surface mount, through-hole mount, press-fit or other. (SMT;THMT;PRESSFIT)	Mount Type
.comp_name	Text	Comp.	Name of the die component on the HDI technology layer.	Name of die component

Attribute	Type	Entity	Description	Display Name
.comp_polarity	Option	Comp.	Assigned to components when packages are imported from the VPL (Valor Parts Library) with the value of: - POLARIZED, has a specific pin designated as pin #1. - NON_POLARIZED has no specific pin #1. A component without this attribute means that its package was not imported from the VPL.	Polarity
.comp_type	Option	Comp.	This attribute is very important for determining dynamic categories during assembly analysis. It represents the type of the component. - axial - bga - cbga - cob - dip - discrete - discrete402 - discrete603 - label - pga - pihconn - pihmisc - plcc - pqfp - printed - qfp - radial - sip - smtconn - smtmisc - socket - soic - soj - sop - sot - tab - tqfp - tsoic - tsop This attribute is only used if both **_comp_type** and **.comp_type2** (see below) are not present. **Note:** Do not use the underscore "_" character in the Type values of this attribute.	Type I

Attribute	Type	Entity	Description	Display Name
.comp_type2	Option	Comp.	Options - axial - axial-large - bga - cbga - cob - csp - dip - dip300 - dip600 - discrete - discrete201 - discrete402 - discrete603 - electro-mech - flipchip - label - lcc - lqfp - pfconn - pga - pihconn-inline - pihconn-rt-angle - pihmisc - pih-polar - plcc - pqfp - printed - qfp - radial - radial-tall - sip - smtconn - smtelect-mech - smtmisc - smtmixedconn - smtpolar - socket - soic - soj - solderable-mech - sop-ssop - sot - tab - tqfp - tsoic - tsop - tsop-tssop This attribute represents the type of the component used in dynamic categories during assembly analysis when user attribute, **_comp_type**, is not defined and at least one component in the step has this attribute. (**.comp_type**, if also present, is ignored.) **Important**: Do not use the underscore "_" character in the Type values of this attribute.	Type II
.comp_variant_list	Text	Comp.	(0 - 1000) Consists of a list of variants where a component is used.	Variant List
.comp_weight	Float	Comp.	(0.0 - 1000.0) Stores the weight of the component (in ounces) for the purpose of the total weight calculation.	Weight

Attribute	Type	Entity	Description	Display Name
.copper_weight	Float	Layer	(0.0 to 1000.0; default = 1.0) The weight in ounces of one square foot of copper.	
.critical_net	Boolean	Feature; Net	Specifies critical nets.	SQA Critical Net
.critical_tp	Boolean	Feature	Assigned to the mid-point of a netlist to force it to become a testpoint (it will not be removed by the Netlist Optimizer). If both **.non_tp** and **.critical_tp** are assigned to the same point, **.critical_tp** takes precedence and the mid point is tested. In case of a drilled feature the attribute must be added to the drill hole.	Netlist Critical Midpoint Output
.cu_base	Boolean	Layer	(Default = No) This attribute indicates to an analysis action (Signal Layer Checks or Power & Ground Checks) that the specific via layer is built in such a way that it necessitates a copper pad on each layer of the stackup, since the vias are drilled and filled (rather than plated), and the pads are an essential element in ensuring connectivity.	
.current_variant	Text	Step	(0 - 100) Consists of the name of the current variant for a step.	
.customer	Text	Job	(0 - 100) This attribute is used for information purposes. It is used specifically in the input process when processing the **lyr_rule** file.	
.cut_line	Integer	Feature	(0 - 100000; default = 0) Assigned to lines added in the creation of film layers by the film optimization algorithm. The attribute is given to three kinds of lines: - frame of the film - cutting lines inside the film - frame of each layer inside the film.	Film Optimization Cut Line
.data_source	Text	Job; Step	(0 - 100) The source of the data. For example, Cadence, Mentor.	

Attribute	Type	Entity	Description	Display Name
.desc1....10	Text	Comp.	(0 - 1000) The line mode command **comp_attr_from_desc_param** can be used to store the values in the ten BOM description fields into the corresponding one of these ten description attributes.	General Description 1....10
.design_center	Text	Step	(0 - 100) The design center from which the job originated.	
.design_origin_x	Integer	Job	(-254000000 to 254000000) Defines the design origin X coordinate. Currently, it is automatically set in the CADIF input process.	
.design_origin_y	Integer	Job	(-254000000 to 254000000) Defines the design origin Y coordinate. Currently, it is automatically set in the CADIF input process.	
.diff_pair	Text	Net	(0 to 64) Differential pair name associating two nets that must be routed together.	
.dpair_gap	Float	Net	(0.0 to 10.0) Spacing gap value expressed in inch or mm specifying the spacing between differential pair nets.	
.drc_add_rad	Integer	Mania_ AOI	(0 to 100; default = 2) For AOI - add lines with this radius when adding shapes.	
.drc_assembly_lyrs	Option	Feature	(Top; Bottom; Both) In Component Analysis, specifies whether the keepout/keepin area applies to Top, Bottom, or Both component layers. In Testpoint Analysis, as above to outer layers.	Assigned Area to Component Side
.drc_bend_keepout	Boolean	Feature	NOT USED	
.drc_board	Boolean	Feature	Assigned to a DRC area defined for the whole board.	
.drc_comp_height	Boolean	Feature	Assigns component height restriction to a keepin/keepout area.	Component Height for Area

Attribute	Type	Entity	Description	Display Name
.drc_comp_height_lyr	Text	Job	(0 to 64) Stores name of document layer in which all component height restriction keepin/keepout areas are stored.	
.drc_comp_keepin	Boolean	Feature	Defines an area as the board's component placement keepin boundary.	Component Keep In
.drc_comp_keepin_lyr	Text	Job	(0 to 64) Stores name of document layer in which all component keepin areas are stored.	
.drc_comp_keepout	Boolean	Feature	Defines an area as the board's component placement keepout boundary.	Component Keep Out
.drc_comp_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all component keepout areas are stored.	
.drc_etch_lyrs	Text	Feature	(0 to 1000) Value = layer names separated by semi-colons (;). User-defined attribute for user to specify name of layers in which to activate keepin/keepout areas. For example, when **.drc_etch_lyrs = pg1;pg2**, this enables you to select/highlight (on the relevant document layer) keepin/keepout areas that are active in layers **pg1**, **pg2**. The attributes: **.drc_etch_lyrs_bit** and **.drc_etch_lyrs** must both specify the same layers. If there is a discrepancy between the two, then **.drc_etch_lyrs_bit** is the determining attribute.	DFx Area Layers by Name
.drc_etch_lyrs_all	Boolean	Feature	Defines a keepin/keepout area as effective on all layers.	DFx Area All Layers

Attribute	Type	Entity	Description	Display Name
.drc_etch_lyrs_bit	Text	Feature	(0 to 64) Values = string consisting of '0' and '1' characters. Allows the keepin/keepout area to apply only to specified board layers. The attribute's length is equal to the number of board layers. 0 = ignore layer, 1 = activate areas in that layer	DFx Area Selected Layers
.drc_max_height	Float	Feature	(0.0 to 10.0) Stores the maximum height of components to be allowed in a height restriction area (area with **.drc_comp_height** attribute) expressed in inch or mm.	Maximum Height for Component
.drc_mech	Boolean	Feature	Obsolete	
.drc_min_height	Float	Feature	(0.0 to 10.0) Stores the minimum height of components to be allowed in a height restriction area (area with **.drc_comp_height** attribute) expressed in inch or mm.	Minimum Height for Component
.drc_min_space	Integer	Mania_ AOI	(1 to 100; default = 5) Minimum spacing. (Obsolete)	
.drc_min_width	Integer	Mania_ AOI	(1 to 100; default = 7) Minimum track width. (Obsolete)	
.drc_pad_keepout	Boolean	Feature	Specifies area to be used as pads keepout boundary.	Pad Keep Out
.drc_pad_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all pad keepout areas are stored. Default as defined in the **drc_pad_keepout** configuration parameter.	
.drc_plane_keepout	Boolean	Feature	Specifies area to be used as planes keepout boundary	Plane Keep Out
.drc_plane_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all plane keepout areas are stored.	
.drc_ref_des	Text	Feature	(0 to 100) Assigned to DRC areas defined for components.	

Attribute	Type	Entity	Description	Display Name
.drc_route_keepin	Boolean	Feature	Specifies areas to be used as the rout keepin boundary (rout=lines, arcs, vias, pads and surfaces on signal and/or power&ground layers).	Route Keep In
.drc_route_keepin_lyr	Text	Job	(0 to 64) Stores name of document layer in which all rout keepin areas are stored.	
.drc_route_keepout	Boolean	Feature	Specifies areas to be used as the rout keepout boundary (rout=lines, arcs, vias, pads and surfaces on signal and/or power&ground layers).	Route Keep Out
.drc_route_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all rout keepout areas are stored.	
.drc_tp_keepin	Boolean	Feature	Defines areas to be used as testpoint keepin area boundaries.	Testpoint Keep In
.drc_tp_keepin_lyr	Text	Job	(0 to 64) Stores name of document layer in which all testpoint keepin areas are stored.	
.drc_tp_keepout	Boolean	Feature	Specifies areas to be used as the testpoint keepout boundary.	Testpoint Keep Out
.drc_tp_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all testpoint keepout areas are stored.	
.drc_trace_keepout	Boolean	Feature	Defines areas to be used as trace keepout boundaries (traces=lines and arcs on signal and/or power&ground layers).	Trace Keep Out
.drc_trace_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all traces keepout areas are stored.	
.drc_via_keepout	Boolean	Feature	Defines areas to be used as vias keepout boundaries.	Via Keep Out
.drc_via_keepout_lyr	Text	Job	(0 to 64) Stores name of document layer in which all vias keepout areas are stored.	

Attribute	Type	Entity	Description	Display Name
.drill	Option	Feature	(plated; non_plated; via) Assigned to hole features in drill layers. It defines the type of the drill and is used extensively during fabrication analysis.	Drill Type
.drill_flag	Integer	Feature	(0 to 100000; default = 0) Used by the Auto Drill Manager. It is an integer feature attribute that should be used on the drill layer. When the Auto Drill Manager package creates the NC Drills table it separates the different drills based on several values: size, drill type and also the value of this attribute. This is useful in cases where specific drills need to be treated in a specific way.	
.drill_layer_direction	Option	Layer	(top2bottom, bottom2top) Whether a pad is reported as on the top or bottom of the drill layer is determined by this attribute. If set to bottom2top, pads on the bottom are reported at ‘top’.	
.drill_noopt	Boolean	Feature	Used by the 'Auto Drill Manager'. Feature attribute that is used on the drill layers. Setting a group of drills with this value will force the drill optimizer to keep the order within that group. This is important for preventing the drill path to pass through mechanical pins.	
.drill_sr_zero	Option	Feature	(1; 2; 3) Used in the Auto Drill Manager to be assigned to a single drill feature in the PCB step. If a single feature in a step is assigned, it is used for setting the 'step & repeat zero offset' of that step. That is, that feature will receive the coordinates - (0,0) in the step & repeat block, and all other coordinates will be relative to it. In order for this attribute to be used, other configuration parameters of the package should be set.	

Attribute	Type	Entity	Description	Display Name
.drill_stage	Option	Feature	(1;2;3) Used in the Auto Drill Manager on the drill layer. This attribute receives three values - '1', '2', and '3', specifying the drill stage of that specific drill hole/slot.	
.dxf_dimension	Boolean	Feature	Assigned during DXF file input to mark its features as part of a DXF dimension entity.	DXF Dimension
.eclass_accumulative_ parallel_dist_list	Text	Net	(0 - 255) List of electrical class rules (blank-separated) defining the maximum distance between the two traces of nets considered parallel in the accumulative categories of ‘Parallel Nets’.	
.eclass_accumulative_ parallel_max_length_list	Text	Net	(0 - 255) List of electrical class rules (blank-separated) defining the maximum distance between the two traces of nets considered parallel in the accumulative categories of ‘Parallel Nets’.	
.eclass_impedance	Float	Net	(0.0 to 1000.0) Electrical class rule	
.eclass_individual_ parallel_dist_list	Text	Net	(0 - 255) List of blank-separated electrical class rules- defines the separation distance within which two traces are considered parallel. Each parallel section in a net, if more than one, is checked separately against the value of this attribute.	
.eclass_individual_ parallel_max_length _list	Text	Net	(0 - 255) List of electrical class rules (blank-separated).- defines the maximum length that two nets can run parallel to each other. Each parallel section in a net, if more than one, is checked separately against the value of this attribute.	

Attribute	Type	Entity	Description	Display Name
.eclass_individual_ parallel_min_jog_list	Text	Net	(0 - 255) List of electrical class rules (blank-separated) - defines the distance parallel traces that deviate must maintain the deviation before it is considered a break in parallelism. Each parallel section in a net, if more than one, is checked separately against the value of this attribute.	
.eclass_max_stub_length	Float	Net	(0.0 to 100.0) Electrical class rule - high limit of the stub length expressed in inch or mm.	
.eclass_max_via_count	Integer	Net	(0 to 1000) Maximal number of vias on the nets.	
.eclass_min_stub_length	Float	Net	(0.0 to 100.0) Electrical class rule - low limit of the stub length expressed in inch or mm.	
.eclass_rise_time	Float	Net	(0.0 to 100.0) Electrical class rule specifying the interval of a rising signal transition (low to high)	
.eclass_voltage_swing	Float	Net	(0.0 to 100.0) Electrical class rule	
.ecmp_layer_tech	Option	Layer	(none, additive, subtractive) Assigns a technology type attribute to an embedded components layer used in the Embedded Passives check.	
.ecmp_max_value	Float	Feature	(0.0 to 1000000.0) Maximum nominal value received at input (its value plus a tolerance).	
.ecmp_min_value	Float	Feature	(0.0 to 1000000.0) Minimum value received at input (its value minus a tolerance).	
.ecmp_name	Text	Feature	(0 to 64 characters) Name assigned to an embedded passive feature.	
.ecmp_type	Option	Feature	(resistor; capacitor) Assigns a component type to an embedded component.	

Attribute	Type	Entity	Description	Display Name
.ecmp_value	Float	Feature	(0.0 to 1000000.0) Embedded passive nominal value. For resistors it is the resistance in ohms.	
.eda_dimension_id	Integer	Feature	(0 - 100000; default = 0) Assigns system-generated ID to dimensions	
.eda_layers	Text	Layer	(0 to 1000) Contains the EDA system layer names which compose a physical layer. It is loaded during the direct EDA translation and is used for graphic synchronization with the EDA system.	
.electrical_class	Text	Net	(0 to 64) Electrical class name associating a net with a set of electrical call rules. Electrical class rules include physical and electrical limitations required to assure and analyze the signal integrity of a high speed net.	
.et_adjacency	Float	Layer	(1.0 to 1000.0; default = 20.0) A distance value (per layer) to use in netlist adjacency calculation for moving probe testers (currently BSL and PROBOT).	
.et_align	Boolean	Feature	Determines that a feature will be used as an alignment target for PROBOT output	
.extended	Integer	Feature	(-1 to 100000; default = 0) Assigned to construction features (lines and pads) added to assist in the generation of a rout path. These features have zero width and are not output to the rout machine as regular features. They are used, for example, as source elements from which to create actual features by dimensions. If the attribute value is not zero then the feature is an extended feature and the decimal value is its serial value in the layer (to be referenced in dimension creation).	

Attribute	Type	Entity	Description	Display Name
.fab_drc	Text	Step	(0 to 20) Stores the default DRC area name. This name is applied when no specific area is defined in the DRC map layer, or no such map layer at all. If the value of this attribute is not defined, then the default DRC name is applied from the configuration variable **default_fab_drc.**	
.feature_ignore	Boolean	Feature	Copper features with this attribute are ignored in analysis actions. (Currently implemented for rout tests only.)	
.feed	Integer	Feature	(0 to 100000; default = 0) For a chained feature, this attribute sets the table feed rate when routing.	
.fiducial_name	Text	Feature	(0 to 64) This attribute is used for **etec** output format. A pad that was given a fiducial name is used for registration between layers.	
.fiducial_rdlist	Text	Feature	(0 - 1000) This attribute is assigned **local** fiducial features. It can consist of a list of REFDES (separated by semicolons '**;**'); a list of the component/s using this local fiducial.	
.fill_dx	Float	Symbol	(0.000001 to 50.0; default = 0.1) This attribute is used as the default horizontal distance between symbols when the symbol is used for pattern filling.	
.fill_dy	Float	Symbol	(0.000001 to 50.0; default = 0.1) This attribute is used as the default vertical distance between symbols when the symbol is used for pattern filling	
.foot_down	Boolean	Feature	(Default=No) Attached to feature it causes a **foot_down_cmd** to be generated by the Auto Rout Manager in the rout file just before the feature. Used only for Excellon files (ignored for other formats).	

Attribute	Type	Entity	Description	Display Name
.fs_direction_bottom	Option	Step	(Right2Left; Left2Right; Top2Bottom; Bottom2Top). This attribute is used for the thieving pad check in assembly analysis. It determines the flow direction for the bottom layer. Thieving pad check is required for some components during the flow solder process	
.fs_direction_top	Option	Step	(Left2Right; Right2Left; Top2Bottom; Bottom2Top) This attribute is used for the thieving pad check in assembly analysis. It determines the flow direction for the top layer.Thieving pad check is required for some components during the flow solder process.	
.full_plane	Boolean	Feature	NOT USED	
.gencad_device_ntol	Float	Comp.	0 - 1000000; default = 0.0) This is a real value expressing the percent of the value to use as a tolerance (negative tolerance). This is used for all devices: Range of characters: all floating point numbers.	
.gencad_device_ptol	Float	Comp.	(0 - 1000000; default = 0.0) This is a real value expressing the percent of the value to use as a tolerance (positive tolerance). This is used for all devices:	
.gencad_device_style	Text	Comp.	(0 - 64) This attribute is an enhancement of **.gencad_device_type** and is used to store the style of the component as defined in GenCAD (such as, NPN, PNP, NFET, PFET, NJFET, PJFET, TTL, CMOS and ECL)	
.gencad_device_type	Text	Comp.	(0 - 64) Stores the type of the component as defined in GenCAD (such as, RES, VRES, DIODE, ZENER, LOGIC, SWITCH, CONN, etc.).	
.gencad_device_value	Text	Comp.	(0 - 64) Stores the electrical value of a component.	

Attribute	Type	Entity	Description	Display Name
.geometry	Text	Feature	(0 to 500) Contains the name of the padstack which created this feature. It is loaded during direct EDA translation. For layers which are created from component layers during the 'Draw to Layer' operation, the attribute will contain (for centroid pads) useful information on the component, package and part name.	
.global_camtek_aoiset	Text	Job	(0-80) Contains the name of the AOIset to be assigned to each layer upon layer selection in the CAMTEK AOI Interface. Once a name is defined, the AOIset field in the CAMTEK popup will be filled with this name and a new AOIset created in the layer (if already exists, the AOIset will become the current set). The value in this attribute overrides the value defined in the configuration parameter **camtek_def_aoiset**, but if no value is specified in this attribute, the **camtek_def_aoiset** value will apply.	
.gold_plating	Boolean	Feature	This attribute should be attached (manually) to features which are a part of a gold plated connector. It is used during auto-panelization to orient the gold plated area toward the extreme side of the panel.	
.guard_comp	Boolean	Comp.	(Default = No) Assigned to a component that “guards” other components. If TRUE, this component is considered a "guard component" (that is, not likely to be knocked off the board accidentally. To be used in future actions.)	

Attribute	Type	Entity	Description	Display Name
.hatch	Boolean	Feature	Assigned to hatched planes [filled with lines (hatches) or cross lines (cross-hatch) instead of solid copper]. The lines which make up the border and fill the surface are hatches.	
.hatch_border	Boolean	Feature	The lines making up the border of a surface.	
.hatch_serrated_border	Boolean	Feature	Assigned to features that are added for partial hatch. The difference between regular hatch and partial hatch is that in partial hatch the cells along the border that intersect the border line are filled; the feature(s) that fill these cells are assigned this attribute.	
.hdi_assembly_tech	Option	Layer	(None, WireBond, FlipChip, Hybrid) The attribute defines the type of HDI assembly technology identified in the job. If None, the special layer is not created.	HDI assembly technology
.hdi_drc	Text	Step	(0 to 20) Default area name applied to all HDI measurements.	
.hp3070_comment	Text	Comp.	(0 - 64) Allows the contents of the attribute field to be appended to a component record (preceded by a "**!**"). For example: **C1 PN"11_215705" "11_215705 POLCAP_10UF,20%,10V TOP"; !** *comment*	
.hp3070_common_pin	Text	Comp.	(0 to16). For the device SWITCH this is used to designate the COMMON pin.	
.hp3070_contact_pin	Text	Comp.	(0 to16). For the device SWITCH this is used to designate the CONTACT pin.	

Attribute	Type	Entity	Description	Display Name
.hp3070_device	Text	Comp.	(0 to 16) The device of the component, one of the following: - CAPACITOR - CONNECTOR - DIODE - FET - FUSE - INDUCTOR - JUMPER PIN - LIBRARY - POTENTIOMETER - RESISTOR - SWITCH - TRANSISTOR - ZENER All other components will be categorized as Undefined.	
.hp3070_fail_msg	Text	Comp.	(0-64) Specifies the failure message associated with the component. This applies to all device types. In output of HP3070 formats, the text field (within quotes) consisting of the Part number and this error message will be truncated to 40 characters.	
.hp3070_hi_value	Float	Comp.	(0-100000.0; default = 0.0) Specifies the upper test limit of the device. Its specific meaning is dependent on the device type. - For DIODE: Upper test limit, in volts for the diode's forward bias voltage. - For FET: The high resistance limit in ohms. - For TRANSISTOR: The high limit for the transistor beta.	
.hp3070_lo_value	Float	Comp.	(0-100000.0; default = 0.0) Specifies the lower test limit of the device. Its specific meaning is dependent on the device type. - For DIODE: Lower test limit, in volts, for the diode's forward bias voltage. - For FET: The low resistance limit in ohms. - For TRANSISTOR: The low limit for the transistor beta.	

Attribute	Type	Entity	Description	Display Name
.hp3070_probe_access	Text	Feature; Comp.	(0-64) Specifies the probe access for the component and toeprint. This value will be applied to ALL the pins of the component. Known values are: PREFERRED, NO_PROBE, TOP, TOP NO_ACCESS, BOTH MANDATORY, and MANDATORY NO_MANUAL though other values are possible. If toeprints are assigned this attribute, their settings override the component setting.	
.hp3070_seriesr	Float	Comp.	(0-100000.0; default = 0.0) For INDUCTOR devices this is used to specify the series resistance (in Ohms).	
.hp3070_test	Boolean	Comp.	(Default=No) Determines that a component be tested. This attribute applies to all device types. Devices of type CONNECTOR must be NT (Not Tested).	
.hp3070_tol_neg	Float	Comp.	(0-100; default = 0.0) This is a real value expressing the percent of the value to use as a tolerance (negative tolerance). This is used for devices: - CAPACITOR - INDUCTOR - POTENTIOMETER - RESISTOR - ZENER	
.hp3070_tol_pos	Float	Comp.	(0-100; default = 0.0) This is a real value expressing the percent of the value to use as a tolerance (positive tolerance). This is used for the devices: - CAPACITOR - INDUCTOR - POTENTIOMETER - RESISTOR - ZENER	

Attribute	Type	Entity	Description	Display Name
.hp3070_type	Text	Comp.	(0 - 8) The type of device:. For CAPACITOR: - F = Capacitor Value is Fixed. - V = Capacitor Value is Variable. For FET: - N = N-Channel Field Effect Transistor - P = P-Channel Field Effect Transistor For INDUCTOR: - F = Inductor value is Fixed - V = Inductor value is Variable For JUMPER: - O or OPEN = Jumper is Open - C or CLOSED = Jumper is Closed For RESISTOR: - F = Resistor value is Fixed - V = Resistor value is Variable For TRANSISTOR: - N = Transistor is an NPN - P = Transistor is a PNP Range of characters: 0-8	
.hp3070_value	Text	Comp.	(0-16) The value of the component. The meaning varies depending on the component device. For CAPACITOR it is used for capacitance (in Farads). For INDUCTOR it is the inductance (in Henries). For PIN LIBRARY it is used for the PN (Part Name). For the devices POTENTIOMETER and RESISTOR, it is used for the device's resistance. For the ZENER device it specifies the breakdown voltage (in Volts).	
.ignore_net	Boolean	Net	When this attribute is assigned to a net, it is ignored during Testpoint Allocation Analysis. No potential testpoints are assigned, they are not reported in the 'Nets without Potential TPs' category, the Testpoints Allocation Report, or in "Total Number of Nets.'	

Attribute	Type	Entity	Description	Display Name
.image_dx **.image_dy**	Float	Symbol	(-1.0 to 50.0; default = -1.0) These values are set when inputting Image files into the system. They contain the datum point of an Image special symbol entity used to set the datum when performing output back into Image format. These values should not be changed by the user as this can cause data corruption.	
.imp_line	Boolean	Feature	Assigned to lines which are impedance-controlled. When set, it prevents the lines from being rerouted or thinned during signal layer optimization.	
.ind_orient_req	Boolean	Comp.	(Default = No) Indicates that the component requires silkscreen orientation indication. (To be used in future actions.)	
.inp_file	Text	Layer	(0 to 500) Contains the name of the file (Gerber, Drill) from which the data was input into the layer.	
.is_burried	Boolean	Comp.	(Default = No) Assigned to buried components specifically input from CADIF files in order to mark them as buried. This attribute, although specifically designed for CADIF files, can be used in any other function or script. Note that the attribute name is misspelled, but that is its name.	
.is_capped	Boolean	Feature	Used on via pads on top & bottom signal layers to indicate that the via is capped on this side.	
.is_shadowed	Boolean	Comp.	(Default = No) Components with this attribute are considered for the Shadowing categories, as the shadowed component.	
.is_wirebonded	Boolean	Comp.	(Default = No) Defines a component to be wire-bonded. Currently, it is set in the CADIF input process.	

Attribute	Type	Entity	Description	Display Name
.label_clearance	Boolean	Comp.	(Default = No) Assigned to components which are not allowed to be too close to a glued label (e.g. fine pitch SOIC components). During the component analysis, these components are checked vs. the label components.	
.layer_dielectric	Float	Layer	(0.0 to 0.5 inch; default = 0.0001) Specifies the dielectric thickness below a layer expressed in inch or mm.	
.layer_hdi_type	Option	Layer	(Buildup; Core) Distinguishes buildup layers from core layers in HDI jobs. Some HDI categories are relevant to buildup or core layers but not to both. Therefore, it is important to set this value appropriately.	
.layer_class	Text	Layer	(0 to 1000) Used to differentiate between layers. It enables you to set different ERF ranges for inner layers than for outer layers.	
.local_fiducial_dist	Float	Comp.	(0.0 to 100.0) Defines the allowed distance of fiducials from the outline of the components which require local fiducials (See **.num_local_fiducial**). If set to 0, the fiducials must be included INSIDE the outline. Distance expressed in inch or mm.	
.lpol_done	Boolean	Layer	(Default = No) Indicates to the output that polarity sort according to a format has already been done during film optimization.	
.lpol_surf	Boolean	Feature	(Default = No) Indicates surface modified by layer polarity reduction algorithm.	
.machine_pkg	Text	Comp.	(0 - 100) Assigned to a component to indicate the name of a corresponding package in the assembly machine libraries.	

Attribute	Type	Entity	Description	Display Name
.mechanical	Boolean	Comp	Components with this attribute are placed in the **MECHNICAL** section of the GenCAD output file when the GenCAD output parameter **Mechnical Components** is set to **Attribute**.	
.merge_processes	Text	Step	(0 - 64) A list of the last three merge actions in the order in which they were run. The list is updated each time a merge (BOM, Library, Board) is run. It is for informational purposes and does not have to be changed by the user.	
.min_line_width	Float	Net.	(0.0 - 100.0) Assigned to nets that should have a minimum trace width, i.e. each line of the net should have a width of at least this value expressed in inch or mm. If such nets have a split (the net traces split and then meet again) the sum of the split traces should be at least this value.	
.mount_hole	Boolean	Feature	Used on drill features to indicate that they are mounting holes.	
.mount_stage	Integer	Comp.	(0 - 255; default = 0) User-defined integer used to assign machine number in the assembly line where component is to be placed.	
.n_electric	Boolean	Feature	Assigned to a feature, defines it as non-electric (it is not considered for the current netlist for the step).	
.needs_guarding	Boolean	Comp.	(Default = No) Yes - this component needs to be protected by guard components (see **.guard_comp**) else it is likely to be knocked off the board accidentally.	
.net_length_max	Float	Net	(0.0 to 100.0) High limit of net length expressed in inch or mm.	

Attribute	Type	Entity	Description	Display Name
.net_length_min	Float	Net	(0.0 to 100.0) Low limit of net length expressed in inch or mm.	
.net_name	Text	Feature	(0 to 64) Set by the netlist layer. Contains the net name.	
.net_physical_type	Text	Feature	(0 to 64) Physical type of constraint area used for search in table that contains physical parameters of nets.	
.net_point	Boolean	Feature	When assigned to a pad in an inner layer, defines the pad as an internal test point.	
.net_spacing_type	Text	Feature	(0 to 64) SQA area name of an SQA area map.	
.net_type	Text	Net	(0 to 64) A name for the type of net. The **.net_type** attribute can reference the set of routing rules for a net.	
.neutralization_angle	Float	Comp.	(0.0-360.0) An attribute attached to each **Rotation Neutralization** processed component stating the angle of rotation counter-clockwise from Valor standard orientation.	
.neutralization_info	Text	Step	(0 - 200) Attached to the step where **Rotation Neutralization** has been performed. This attribute contains the information **<CPL|CAD>;<DataCenter>;Site**. Site is read from configuration parameter **organization** of the computer where **Rotation Neutralization** was performed.	
.neutralization_reviewed	Boolean	Comp.	Attached to each component in a package reviewed in **Rotation Neutralization**, i.e. a package not accepted automatically as being **Known** or **Safe** or by clicking **Accept Category**.	

Attribute	Type	Entity	Description	Display Name
.neutralization_ss_ layers	Text	Step	(0 to 200) Attached to the step where **Rotation Neutralization** has been performed. This attribute designates which layers are to be considered the silkscreen layers.	
.no_copper_shape_under	Boolean	Comp.	(Default = No) This attribute indicates that the component should not have copper pads or surfaces underneath it. See also **.no_trace_under**.	
.no_fiducial_check	Boolean	Comp.	(Default = No) Components with this attribute are not checked for the “Component Covers Fiducial” category, or for any of the categories under the Coverage test.	
.no_hole_under	Boolean	Comp.	(Default = No) If Yes, no drill holes are allowed under this component.	
.no_pop	Boolean	Comp.	(Default = No) A RefDes with the attribute **.no_pop** (non populated) declares a component as being not populated for the current version of the BOM. When attributed as **.no_pop** (Yes), even though the component is defined in the CAD data it will not be placed during the assembly process.	
.no_protrude_board	Boolean	Comp	Indicates that component pin length (as defined by attribute **.pin_length**) should be less than the board thickness so that the pins do not protrude from the other side of the board.	
.no_text_under	Boolean	Comp.	(Default = No) Assigned to a component, does not allow silk screen text to be placed under the component outline. Printed components (e.g. edge connectors) may not have this attribute.	

Attribute	Type	Entity	Description	Display Name
.no_tp_under	Boolean	Comp.	(Default = No) Assigned to a component, does not allow testpoints to be placed under the component outline. Printed components (e.g. edge connectors) may not have this attribute.	
.no_trace_under	Boolean	Comp.	(Default = No) Yes - traces are NOT allowed under the component except for those that touch the component's toeprint pads and exit the component on that toeprint's side.	
.no_uncap_via_under	Boolean	Comp.	(Default = No) Yes - uncapped vias are NOT allowed under this component.	
.nomenclature	Boolean	Feature	Defines a feature as a nomenclature (legend) feature. This attribute affects the fabrication analysis by directing spacing checks between such features into a new category (Text to text).	
.non_tp	Boolean	Feature	Assigned to a feature causes it NOT to be considered as a net testpoint. It is used for connectivity calculation but is not used as a test point (bare board testing).	
.num_local_fiducials	Integer	Comp.	(0 to 20; default = 0) Defines how many local fiducials are expected to be inside or near a component. This is checked during Fiducial Analysis.	
.orbotech_plot_stamp	Boolean	Features	NOT USED	
.orig_surf	Integer	Feature	(0-2147483647;default = 0) Identifies original surface which will be rebuilt.	
.otherside_keepout	Option	Comp.	(full_area; pins_only; pads only) Defines for components whether the other side of the board may also contain components in the same area.	

Attribute	Type	Entity	Description	Display Name
.out_angle	Option	Layer	(0.0; 90.0; 180.0; 270.0; default = 0.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.	
.out_break	Boolean	Feature; Symbol	(Default = No) Feature and symbol attribute. When assigned to a specific feature using a special symbol, the feature will be broken into it's primitives in the output translation stage, regardless of the settings of other output parameters. If the attribute is set for a special symbol (entity attribute) then all features that use these symbols will always be broken into primitive features in the output translation stage, regardless of the settings of any other output parameters	
.out_comp	Float	Layer	(-100.0 to 100.0; default = 0.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.	
.out_drill_full	Boolean	Step	(Default = No) The STEP entity attribute used by the Auto Drill Manager. This attribute can be used for drilling coupon STEPs that need to be fully drilled before continuing to the next step & repeat entity.	
.out_drill_optional	Boolean	Feature; Step	(Default = No) Used by the 'Auto Drill Manager'. Both a STEP entity and feature attribute. If the drill feature is set with this attribute it will have the '**/**' command prefix in the final output file, indicating that the drill is optional. If a step entity attribute is set, then all the commands that are part of that step will have the '**/**' command prefixed. Thus, the whole step is optional.	

Attribute	Type	Entity	Description	Display Name
.out_drill_order	Integer	Step	(-10000 to 10000; default = 0) The STEP entity attribute used by the Auto Drill Manager. The attribute controls the order in which the steps will be drilled. Thus, who is first, second,,,etc. The attribute has the following valid values: 0 - no special order for that step 1 - first 2 - second 3 - and above - order from the beginning -1 - last -2 - one before last -3 - and on (drill order from the end)	
.out_flag	Integer	Feature	(-1 to 1000000; default = -1) Used for Excellon translation.	
.out_mirror	Boolean	Layer	(Default = No) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.	
.out_name	Text	Step	(0 to 64) Entity attribute that is used by the Image output translator. If this attribute is not an empty string it will serve as the entity name on the Image system. If it is an empty string the original system entity name will be used. This attribute is important in ases where the Genesis name does not form a legal Image name. If this attribute is not set, the Genesis output translator decides about the new name with its own internal algorithm.	
.out_orig	Boolean	Feature	Sets an origin point for the layer data that is transmitted to the NC routing machine.	
.out_polarity	Option	Layer	(Positive; Negative) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.	

Attribute	Type	Entity	Description	Display Name
.out_rout_optional	Boolean	Feature; Step	(Default = No) Used by the Auto Drill Manager. Both a STEP entity and feature attribute. If drill feature is set with this attribute it will have the '/' command in front of it in the final output file. This means that the drill is optional. If a step entity attribute is set then all the commands that are part of that step will have the '/' command at the beginning. Thus, the whole step is optional.	
.out_rout_order	Integer	Step	(-10000 to 10000; default = 0) STEP entity attribute used by the Auto Drill Manager. The attribute controls the order in which the steps will be drilled. Thus, who is first, second,,,etc. The attribute has the following valid values: 0 — no special order for that step 1 — first 2 — second 3 and above — order from the beginning -1 — last -3 and on — drill order from the end	
.out_scale	Boolean	Feature; Symbol	(Default = No) Feature and symbol attribute. In the output translation package there is a special parameter that controls the way features will be scaled. In two of the options the user can specify whether certain features can be scaled or not. This is important in cases where special registration targets would not be scaled together with all the other features. This special output option applies only to features that have this attribute set. In case of a special symbol, the customer can set the attribute, and by this control the scaling of all features that use this symbol.	

Attribute	Type	Entity	Description	Display Name
.out_x_scale	Float	Layer	(0.000001 to 5.0; default = 1.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.	
.out_y_scale	Float	Layer	(0.000001 to 5.0; default = 1.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.	
.output_dcode	Integer	Feature	(0-1000000) Assigned to features to provide action codes for an assembly machine, such as the GSI Lumonics laser cutter.	
.package_version	Text	Comp.	(0 to 50) Used for Zuken Board Designer translation.	
.pad_usage	Option	Feature	(toeprint; via; g_fiducial; l_fiducial; tooling_hole) This attribute defines the specific usage of a pad. It is loaded during the direct EDA translation and by the attribute derivation script.	
.part_desc1...10	Text	Comp.	(0 to 1000) The line mode command **comp_attr_from_desc_param** can be used to store the values in the ten BOM description fields into the corresponding one of these ten part description attributes.	
.patch	Boolean	Feature	Assigned to patches added by the pinhole elimination DFM action.	Copper Patch
.pattern_fill	Boolean	Feature	Assigned to features which are added during a pattern fill operation, either manually or through the Copper Balance DFM action.	Film Optimization Cut Line
.pf_optimized	Boolean	Feature	(Yes, No) This attribute is attached to a modified padstack on the pad in the matching drill layer when padstack optimization is implemented.	

Attribute	Type	Entity	Description	Display Name
.physical_type	Text	Net	(0 to 64) Physical type of net.	
.pilot_hole	Integer	Feature	(0 to 100000; default = 0) Attribute assigned to pads that are pilot holes in a chain (holes that are drilled in each tool down in the chain rout path before routing the chain). Pilot holes are set from the chaining popup. The value of the attribute is the serial chain number to which the pilot hole belongs. When merging or inserting chains, the pilot holes are updated automatically.	
.pin_length	Float	Comp.	(0.0-10.0) The length of the component pins expressed in inch or mm. (Relevant to TH pins.)	
.pin_name	Text	Comp.	Name of the die bump.	Name of die bump
.pitch	Float	Feature	NOT USED	
.plated_type	Option	Feature	(Standard, Press_fit) Defines plated hole type in drill layers using the Attributes popup or the Drill Tool Manager.	
.polarity_marker	Integer	Comp.	(1-10000; default = 1) An attribute indicating which pin of the component is Pin 1.	
.primary_side	Option	Job	(Top; Bottom) Indicates the primary side for this job.	
.rot_correction	Integer	Comp.	(0-359) Component machine rotation correction to apply.	
.rout_chain	Integer	Feature	(0 to 100000; default = 0) Contains the serial number of the chain to which the feature belongs. Features belonging to that chain are rearranged in the features database according to their order inside the chain. Additional attributes that are added to a chained feature: **.feed, .speed, .rout_flag, .comp**	

Attribute	Type	Entity	Description	Display Name
.rout_flag	Integer	Feature	(0 to 100000; default = 0). For each chained feature this attribute represents a numeric value supplied to a chain to provide data for the automatic process of the Auto Rout Manager.	
.shave	Boolean	Feature	(Default = No) Assigned to all the shaves (negative merges) that the silk screen optimization adds in merge mode.	
.sip	Option	Feature	(Detected / Repaired). Indicates whether the SIP (self-intersecting polygon) has been detected or repaired.	
.sliver_fill	Boolean	Feature	Assigned to all the fills added by the sliver fill DFM actions.	
.smd	Boolean	Feature	Assigned to outer layer pads designated as toeprints which are lands for SMD components. It is set by the 'Set SMD Attribute' Cleanup Action.	
.smt_direction_bottom	Option	Step	(Left2Right; Top2Bottom; Right2Left; Bottom2Top) Defines the direction of the SMT process flow on the bottom side.	
.smt_direction_top	Option	Step	(Left2Right; Top2Bottom; Right2Left; Bottom2Top) Defines the direction of the SMT process flow on the top side.	
.source_llayer	Text	Feature	(0-64) This attribute is used by the Enterprise Mentor EDA translator to identify the "Source Logical Layer" of features (traces) appearing on signal or mixed layers. The translator uses this attribute in a filtering stage that addresses pad/signal mapping.	
.spacing_req	Float	Feature	(0.0 to 100.0) (units = mils or microns)	
.speed	Integer	Feature	(0 to 100000; default = 0) For a chained feature this attribute sets the spindle speed (in revolutions per minute) when routing.	

Attribute	Type	Entity	Description	Display Name
.spo_h_fact	Float	Feature; Comp.	(0.3 <-> 2.0; default = 0.8) When **.spo_h_mode** = Factor, **.spo_h_fact** specifies the factor by which paste pad heights are sized relative to their SMD pads. For example, 0.9 means height is 90% of SMD pad.	
.spo_h_mode	Option	Feature; Comp.	(Distance, Factor, Value) Defines how heights of paste pads are sized: by distance, factor or value.	
.spo_h_val	Float	Feature; Comp.	(-500 to +500; default = 5) When **.spo_h_mode** = Distance, **.spo_h_val** is the reduction/ expansion in mils or microns of the paste pad width relative to the SMD pad width. For example, **.sp_h_val** = 5.0 mils shrinks paste pad by 5.0 mils (2.5 mils on each side) relative to SMD pad width. Positive number results in smaller paste pad, negative number in larger paste pad. When **.spo_h_mode** = Value, **.spo_h_val** becomes the absolute width of the paste pad (for example, 5.0 mils becomes the actual width of the paste pad).	
.spo_move_center	Float	Feature; Comp.	(range: -500 to 500, default=0) To move the paste pad from the SMD pad center. A positive value will move the paste from the component center out. A negative value will move the paste towards the component center. Values expressed in mils or microns.	
.spo_p_fact	Float	Feature; Comp.	(0.3 <-> 2.0; default=0.8) When **.spo_p_mode** = Factor, **.spo_p_fact** specifies the factor by which paste pad heights are sized relative to their SMD pads. For example, 0.9 means area is 90% of SMD pad.	
.spo_p_mode	Option	Feature; Comp.	(Distance, Area) Defines how paste pads for non-standard symbol SMD pads are sized: by distance, or area.	

Attribute	Type	Entity	Description	Display Name
.spo_p_val	Float	Feature; Comp.	(-500 to 500; default=5) When **.spo_p_mode** = Distance, **.spo_p_val** is the reduction/ expansion of the paste pad width relative to the SMD pad width expressed in mils or microns. For example, **.sp_p_val** = 5.0 mils shrinks paste pad by 5.0 mils (2.5 mils on each side) relative to SMD pad width. Positive number results in smaller paste pad, negative number in larger paste pad.	
.spo_s_fact	Float	Feature; Comp.	(0.3 <-> 2.0; default=0.8) When **.spo_s_mode** = Factor, **.spo_s_fact** specifies the factor by which paste pad heights are sized relative to their non-standard SMD pads. For example, 0.9 means height is 90% of SMD pad.	
.spo_s_mode	Option	Feature; Comp.	(Distance, Factor, Value, Area) Defines how heights of paste pads for symmetric SMD pads are sized: by distance, factor, value, area.	
.spo_s_val	Float	Feature; Comp.	(-500 to 500; default=5) When **.spo_s_mode** = Distance, **.spo_s_val** is the reduction/ expansion expressed in mils or microns of the paste pad width relative to their non-standard SMD pad width. When **.spo_h_mode** = Value, **.spo_h_val** becomes the absolute size of the paste pad.	
.spo_shape	Text	Feature; Comp.	Specifies the symbol to be used as the solder paste pad applied to a feature (smd pad) or to the toeprints of a component. The initial orientation of the symbol is also affected by the **.spo_shape_rotate** attribute when defined.	

Attribute	Type	Entity	Description	Display Name
.spo_shape_rotate	Float	Feature; Comp.	(0 to 360; default = 0) Specifies the initial rotation of the symbol defined in the **.spo_shape** attribute. Both **.spo_shape_rotate** and **.spo_shape** should appear as a pair at the level at which they are activated (component or feature). This means, for example, that an **.spo_shape_rotate** defined without a corresponding **.spo_shape** in the feature level is ignored.	
.spo_shape_stretch	Boolean	Feature; Comp.	(Default = No) Specifies that the symbol defined in **.spo_shape** is to be stretched to fit the copper pad dimensions. The stretch limits are determined by applying the SPO width and height parameters (**pp_w_***, **pp_h_***) or attributes (**.spo_w_***, **.spo_h_***) on the copper pad bounding box. Both **.spo_shape_stretch** and **.spo_shape** should appear as a pair at the level at which they are activated (component or feature). This means, for example, that an **.spo_shape_stretch** defined without a corresponding **.spo_shape** in the feature level is ignored.	
.spo_w_fact	Float	Feature, Comp.	(0.3 <-> 2.0; default=0.8) When **.spo_w_mode** = Factor, **.spo_w_fact** specifies the factor by which paste pad widths are sized relative to their SMD pads. For example, **.spo_w_fact** = 0.9 width of paste pad is 90% of width of SMD pad.	
.spo_w_mode	Option	Feature, Comp.	(Distance; Factor; Value; Area) Defines how widths of paste pads are sized: by distance, factor or value.	

Attribute	Type	Entity	Description	Display Name
.spo_w_val	Float	Feature, Comp.	(-500 to 500; default=5) When **.spo_w_mode** = Distance, **.spo_w_val** is the reduction/ expansion expressed in mils or microns of the paste pad width relative to the SMD pad width. For example, **.sp_w_val** = 5.0 mils shrinks paste pad by 5.0 mils (2.5 mils on each side) relative to SMD pad width. Positive number results in smaller paste pad, negative number in larger paste pad. When **.spo_w_mode** = Value, **.spo_w_val** becomes the absolute width of the paste pad (for example, 5.0 mils becomes the actual width of the paste pad).	
.src_orientation	Integer	Comp.	(-1 to 3; default = -1) Defines the zero orientation of this component relative to its orientation in the packages database. That is, the orientation of the component on the automated assembly tape, or (for manually inserted components) the orientation in which pin #1 is "in the same position" for all similar components.	
.station	Text	Comp.	NOT USED (Min_len= 0, Max_len = 255)	
.string	Text	Feature	(0 to 1000) For nomenclature features, the value of this attribute is the original text string which the feature is part of. During EDA input, all occurrences of the asterisk character '*' are replaced by the hyphen character '-' (the reason: when filtering, the asterisk character '*' is used to denote any substring match).	

Attribute	Type	Entity	Description	Display Name
.string_angle	Float	Feature	(0 to 360) For nomenclature features, the value of this attribute is the original (in the input file) text rotation angle which the feature is part of. This attribute is assigned to translated jobs of formats: Cadence Allegro BRD/APD, Mentor BoardStation, PADS PowerPCB.	
.string_justification	Option	Feature	String justification: tl, tc, tr, cl, cc, cr, bl, bc, br **tl, tc, tr**—top-left, top-center, top-right **cl, cc, cr**—center-left, center-center, center-right **bl, bc, br**—bottom-left, bottom-center, bottom-right (Default = **bl**)	
.string_mirrored	Boolean	Feature	(No, Yes) Assigned to mirrored strings.	
.tear_drop	Boolean	Feature	Assigned to features which are added during a tear drop operation, either manually or through the Teardrop Creation DFM action.	
.technology	Text	Job	(0 - 100) Defines the technology used in creating the job. Currently it is set automatically in the CADIF process.	
.test_point	Boolean	Feature	Assigned to features which are used for In-Circuit Testing operations. It is loaded during the direct EDA translation and is used during the Testpoint Analysis action. Also supported in Zuken BD.	

Attribute	Type	Entity	Description	Display Name
.test_potential	Option	Feature	An attribute attached to features being considered as testpoints (potential testpoints) for In-Circuit Testing operations. It is assigned either by the Testpoint Allocation Action or manually by the user. **potential_tp_by_analysis** - a feature meeting all criteria of the Testpoint Allocation Action. **potential_tp_manually** - a feature to be used as a testpoint though it does not meet all criteria. **not_potential_tp_manually** - a feature not to be used as a testpoint even though it meets all criteria.	
.testpoint_count	Integer	Net	(-1 to 10000) Specifies the number of testpoints expected on this net. If this variable is not defined, or its value is **-1**, the number of expected testpoints in unlimited. A value of **-1** is given to a net that does not require a specific number of test points. When the Num-verify test is run, it ignores such nets (even when the ERF variable **v_testpoint_count_default** is defined).	
.testpoint_name	Text	Feature	(0 to 64) Name of the testpoint.	
.thvpad_required	Boolean	Comp.	(Default = No) Assigned to components which require a thieving pad check during the Padstack Analysis action (e.g. fine pitch SOIC).	

Attribute	Type	Entity	Description	Display Name
.toep_nochk_o_side	Boolean	Comp.	(Default = No) Assigned to components so that their toeprints on the opposite side will be excluded from the Toeprint to Toeprint category measurement results in Padstack Analysis, and from the Component to Toeprint category in Component Analysis. **Important Note**: This attribute is applied only when the following ERF variables are set: **c2toep_by_comp=1** (in **component.erf**) **toep2toep_by_comp=1** (in **padstack.erf**)	
.toep_spacing_req	Float	Comp.	(1 to 100; default = 5) Assigned to components for reporting in the Toeprint to Toeprint category in the Signal Layers Check in Analysis. It defines the maximum spacing expressed in inch or mm within which to report pad to pad spacing measurements. Units expressed in inch or mm.	
.tooling_hole	Boolean	Feature	Used on drill features to indicate that they are tooling holes.	
.user_bom_rev	Text	Comp.	(0 - 1000) Used to describe user modified component extensions. Although a user can set a component to a different CPN by using the Set CPN function, this attribute disappears when BOM Merge is performed.	
.variant_list	Text	Job	(0 - 1000) Consists of a list of all possible variants of a job.	
.vcut	Boolean	Feature	Assigned to rout features that are cut in a V-shape (such as in the figure). Another machine performs the V-shape cutting. 	

Attribute	Type	Entity	Description	Display Name
.via_type	Option	Feature	(Drilled; Laser; Photo) Assigned to via drills for the classification of various via pad and via drill categories in the HDI analysis.	
.viacap_layer	Option	Step	(Top; Bottom; Both; None; default = None) Defines on which layer via capping can occur, if any.	
.wheel_type	Option	Wheel	(Gerber; Tools) Specifies for a wheel whether it is used for Gerber files translation or for drill file translation.	

Appendix B System Attributes for Genesis

The following table is a list of the system attributes currently used by Genesis programs.
The columns are:

Attributes - the internal name of the attribute

Type - Float, Integer, Boolean, Text, Option

Entity - the ODB++ entity (such as Job, Step, Feature, Component, Symbol, Wheel, ...) to which the attribute can be assigned.

Attribute List

Attribute	Type	Entity	Description
.action_mask_layer	Integer	Layer	Name of the mask layer.
.action_mask_layer_type	Text	Layer	Sets the type of mask: inclusion or exclusion.
.array_with_rotation	Boolean	Step	If TRUE, this step is a multi-panel array, with the same panel possibly appearing in 180-degree rotation to itself
.assembly_proc_bottom	Text	Step	(0 to 20) Default assembly process for the bottom side, to be used when there is no specific area defined in the process map layer (or no process map layer at all)
.assembly_proc_top	Text	Step	(0 to 20) Default assembly process for the top side, to be used when there is no specific area defined in the process map layer (or no process map layer at all).
.avoid_shave	Boolean	Feature	(Yes, No) If set, tells a DFM action not to shave a pad with this attribute.
.bit	Text	Feature	(0 - 64) Contains the drill designator which is set to each tool in the Drill Tools Manager.
.board_thickness	Float	Job	(0.0 to 10.0) Total thickness of the board.

<table>
<tr><th>Attribute</th><th>Type</th><th>Entity</th><th>Description</th></tr>
<tr><td>.break_away</td><td>Boolean</td><td>Symbol</td><td>Assigned to a symbol representing a break-away to be inserted into any line or arc of the rout path. When adding a break_away symbol thru dimensions, it automatically adjusts to the line or arc angle, breaks that feature (in the breaking points defined in that symbol with the .brk_pnt attribute), and adds all the necessary connections and dimensions.</td></tr>
<tr><td>.brk_point</td><td>Boolean</td><td>Feature</td><td>Assigned to a pad or a dpoint in a break-away symbol (that was given the attribute .break_away). When adding the break-away to the line/arc in the layer, thru dimensions, the line/arc is broken at the connection point with the dpoint that has the .brk_point attribute. In each break-away symbol there should be two points with this attribute.</td></tr>
<tr><td>.canned_text</td><td>Boolean</td><td>Feature</td><td>Indicates that a text is drilled (applies to features).</td></tr>
<tr><td>.cdr_mirror</td><td>Text</td><td>Layer</td><td>(No, Yes)
The mirroring of a layer for AOI inspection is set in the .cdr_mirror layer attribute.
If Yes, the layer is mirrored for AOI inspection.
If No, the layer is not mirrored.
If unset, the mirroring of the layer is assumed to be the opposite of mirroring for plottint. The mirroring used for poltting is deduced fromt he combination of two factors: the value of the layer attribute .out_mirror, and the existence of mirroring in the Image Production layer parameters. The table below summarizes the possibilities.

<table><tr><th>Mirroring in Image Production Parameters</th><th>Value of .out_mirror</th><th>AOI Mirror (Result)</th></tr><tr><td>No</td><td>No</td><td>Yes</td></tr><tr><td>No</td><td>Yes</td><td>No</td></tr><tr><td>Yes</td><td>No</td><td>No</td></tr><tr><td>Yes</td><td>Yes</td><td>No</td></tr></table></td></tr>
<tr><td>.cdr_val</td><td>Integer</td><td>Feature</td><td>(-1 to 100000)</td></tr>
<tr><td>.cdr14_stages</td><td>Text</td><td>Feature</td><td>(0 - 400)
Assigned to alignment target features, and describes the work stage(s) for which the target was set.</td></tr>
<tr><td>.cdr14_zone_type</td><td>Text</td><td>Feature</td><td>(0 - 30)
Assigned to features representing exclusion zones, describing zone type as set by the operator.</td></tr>
<tr><td>.center_fiducial</td><td>Boolean</td><td>Comp.</td><td>(Yes, No)
Specifies component is expected to have a fiducial at its center.</td></tr>
</table>

Attribute	Type	Entity	Description
.color	Text	Feature	(00 to 99 for each color) Can be attached to any feature or component to define the color to be used in plotting a layer in HPGL-1 or 2. The format is **rrggbb** (where **r**=red, **g**=green, **b**=blue). **White - .color = "999999"** **Black - .color = "000000"** **Red - .color = "990000"** **Green - .color = "009900"** **Yellow - .color = "009999"** **Blue - .color = "000099"** **Magenta - .color = "990099"** **Cyan - .color = "999900"**
.combined_size	Float	Feature	(0 to 100000.0)j Keep the original size for combined tools. If the tool is the combined drill size, the attribute equals the combined drill size. For non-combined tools, the attribute is undefined.
.comment	Text	Job	(0 to 500) Used for general textual comments.
.comp	Option	Feature	(none; right; left) For a chained feature, this attribute sets the offset of the cutting tool from the rout path. Three options: - None - in center of the rout path - Left - to the left of the rout path in the direction of cutting - Right - to the right of the path
.comp_height	Float	Comp.	(0.0 to 10.0) Stores the height of the component above the board surface .
.comp_htol_minus	Float	Comp.	(0.0 - 10.0) Contains the minus tolerance for component height, used for calculation of plug-in boards.
.comp_htol_plus	Float	Comp.	(0.0 - 10.0) Contains the plus tolerance for component height, used for calculation of plug-in boards.
.comp_ign_spacing	Boolean	Comp.	This attribute, when set, disables spacing checks on a component during assembly analysis. It is used for printed components which have no actual body
.comp_ignore		Comp.	Determines whether the component is to be ignored when calculating statistics, or during certain operations, such as Analysis.
.comp_mount_type	Option	Comp.	(Other; SMT; THMT) Indicates whether the component is a surface mount, through-hole mount, press-fit or other. (SMT;THMT;PRESSFIT)

Attribute	Type	Entity	Description
.comp_type	Option	Comp.	This attribute is very important for determining dynamic categories during assembly analysis. It represents the type of the component. This attribute is only used if both **_comp_type** and **.comp_type2** (see below) are not present. **Note:** Do not use the underscore "_" character in the Type values of this attribute. - axial - bga - cbga - cob - dip - discrete - discrete402 - abel - pga - pihconn - pihmisc - plcc - pqfp - printed - qfp - radial - sip - smtconn - smtmisc - socket - soic - soj - sop - sot - tab - tqfp - tsoic - tsop
.comp_weight	Float	Comp.	(0.0 - 1000.0) Stores the weight of the component (in ounces) for the purpose of the total weight calculation.
.connection_id	Integer	Feature	In JTAG operations, all traces and pads electrically connected to a cut polyline are asigned the attribute **.connection_id**. This attribute is used to aid in reconnecting the traces. Its value is the value of the attribute **.jtag_component_id** * 100, plus a value that relates to the internal index of the originating JTAG pad.
.copper_weight	Float	Layer	(0.0 to 1000.0) The weight in ounces of one square inch of copper.
.critical_net	Boolean	Feature/ Net	Specifies critical nets.
.critical_tp	Boolean	Feature	Assigned to the mid-point of a netlist to force it to become a testpoint (it will not be removed by the Netlist Optimizer). If both **.non_tp** and **.critical_tp** are assigned to the same point, **.critical_tp** takes precedence and the mid point is tested. In case of a drilled feature the attribute must be added to the drill hole.
.cu_base	Boolean	Layer	This attribute indicates to an analysis action (Signal Layer Checks or Power & Ground Checks) that the specific via layer is built in such a way that it necessitates a copper pad on each layer of the stackup, since the vias are drilled and filled (rather than plated), and the pads are an essential element in ensuring connectivity.
.customer	Text	Job	(0 to 100) This attribute is used for information purposes. It is used specifically in the input process when processing the **lyr_rule** file.

Attribute	Type	Entity	Description
.cut_line	Integer	Feature	Assigned to lines added in the creation of film layers by the film optimization algorithm. The attribute is given to three kinds of lines: - frame of the film - cutting lines inside the film - frame of each layer inside the film.
.deferred	Boolean	Feature	Indicates a plot stamp feature is flagged as deferred while being output to LP7008 and DP100.
.depth	Float	Layer	(1.0 - 1000.0) Depth of drill layer in mils (applies to layers)
.design_center	Text	Step	(0-100) The design center from which the job originated.
.design_origin_x	Integer	Job	(minus 254000000 to plus 254000000) Defines the design origin X coordinate. Currently, it is automatically set in the CADIF input process.
.design_origin_y	Integer	Job	(minus 254000000 to plus 254000000) Defines the design origin Y coordinate. Currently, it is automatically set in the CADIF input process.
.drc_add_rad	Integer	Mania_ AOI	(0 to 100) For AOI - add lines with this radius when adding shapes.
.drc_min_space	Integer	Mania_ AOI	(1 to 100) Minimum spacing. (Obsolete)
.drc_min_width	Integer	Mania_ AOI	(1 to 100) Minimum track width. (Obsolete)
.drill	Option	Feature	(plated; non_plated; via) Assigned to hole features in drill layers. It defines the type of the drill and is used extensively during fabrication analysis.
.drill_flag	Integer	Feature	(0 to 100000) Used by the Auto Drill Manager. It is an integer feature attribute that should be used on the drill layer. When the Auto Drill Manager package creates the NC Drills table it separates the different drills based on several values: size, drill type and also the value of this attribute. This is useful in cases where specific drills need to be treated in a specific way.
.drill_noopt	Boolean	Feature	Used by the 'Auto Drill Manager'. Feature attribute that is used on the drill layers. Setting a group of drills with this value will force the drill optimizer to keep the order within that group. This is important for preventing the drill path to pass through mechanical pins.

Attribute	Type	Entity	Description
.drill_sr_zero	Option	Feature	(1; 2; 3) Used in the Auto Drill Manager to be assigned to a single drill feature in the PCB step. If a single feature in a step is assigned, it is used for setting the 'step & repeat zero offset' of that step. That is, that feature will receive the coordinates - (0,0) in the step & repeat block, and all other coordinates will be relative to it. In order for this attribute to be used, other configuration parameters of the package should be set.
.drill_stage	Option	Feature	(1;2;3) Used in the Auto Drill Manager on the drill layer. This attribute receives three values - '1', '2', and '3', specifying the drill stage of that specific drill hole/slot.
.dxf_dimension	Boolean	Feature	Assigned during DXF file input to mark its features as part of a DXF dimension entity.
.eda_layers	Text	Job	(0 to 1000) Contains the EDA system layer names which compose a physical layer. It is loaded during the direct EDA translation and is used for graphic synchronization with the EDA system.
.entity_version	Integer	Step, Symbol	(0 to 2147418112) Counts the number of changes made in an entity (applies to steps and symbols). **Note** - Do not modify!
.et_adjacency	Float	Layer	(1.0 to 1000.0) A distance value (per layer) to use in netlist adjacency calculation for moving probe testers (currently BSL and PROBOT).
.et_align	Boolean	Feature	Determines that a feature will be used as an alignment target for PROBOT output
.et_stamp	Boolean	Feature	(No, Yes) A feature tagged with this attribute is used as a stamp point in Hioki output.
.etch_comp_addition	Boolean	Feature	
.etm_adapter_h	Integer	Step	(0.000001-5000) Adapter Height in Mils.
.etm_constant_drill_usage	Option	Feature	(plate; cs_board; cs_grid; test)
.etm_height	Float	Layer	For the Job to Adapter option. Defines the height of the plate in the adapter represented by the given layer.
.etm_mirror	Boolean	Layer	For the Job to Adapter option. Updates the mirror of the drill output transformation for the required plate.
.etm_pin_name	Text	Feature	(0-64) ETM pin name.
.etm_pin_style	Option	Step	(Regular, Mania) ETM Pin Guiding Style.

Attribute	Type	Entity	Description
.etm_prim_sink_h	Float	Layer	(0.0 to 1000.0) For the Job to Adapter option. Defines the depth of the countersink from the board side of the plate for the required plate.
.etm_prim_sink_r	Float	Layer	(0.0 to 1000.0) Countersink Threshold radius on the primary side. (ET)
.etm_prim_sink_s	Float	Layer	(0.0 to 1000.0) Countersink drill size on primary side. (ET)
.etm_repair_fmt	Option	Step	ETM Repair file format. Options: None, EPC (note that it is 'repear' in the attribute name)
.etm_rotate	Option	Layer	(0; 90; 180; 270) For the Job to Adapter option. Defines the rotation of the drill output transformation for the given plate definition.
.etm_sec_sink_h	Float	Layer	(0.0 to 1000.0) For the Job to Adapter option in the ETM. Defines the depth of the countersink for the grid side of the required plate.
.etm_sec_sink_r	Float	Layer	(0.0 to 1000.0) (ETM). Countersink Threshold radius on secondary side.
.etm_sec_sink_s	Float	Layer	For the Job to Adapter option. Not used.
.etm_shift_x	Float	Layer	(-100000.0 to 100000.0) For the Job to Adapter option in the ETM. Defines the **x** offset of the drill output trnasformation for the given plate (represented by the layer to which it is assigned).
.etm_shift_y	Float	Layer	(-100000.0 to 100000.0) For the Job to Adapter option in the ETM. Defines the **y** offset of the drill output trnasformation for the given plate (represented by the layer to which it is assigned).
.etm_step_x	Float	Layer	(0.0 to 1000.0) For the Job to Adapter option in the ETM. Defines the step of the grid being defined on the x axis.
.etm_step_y	Float	Layer	(0.0 to 1000.0) For the Job to Adapter option in the ETM. Defines the step of the grid being defined on the y axis.
.etm_tester	Text	Step	(0-64). Options Mania; Everett Charles; Circuitline; Luther; Maelzer; Probot; BSL; IntegriTest; MicroCraft; ATG. ETM tester name.
.etm_thickness	Float	Layer	(0.0 to 1000.0) For the Job to Adapter option in the ETM. Specifies the thickness of the plate being defined.

Attribute	Type	Entity	Description
.extended	Integer	Feature	(-1 to 100000) Assigned to construction features (lines and pads) added to assist in the generation of a rout path. These features have zero width and are not output to the rout machine as regular features. They are used, for example, as source elements from which to create actual features by dimensions. If the attribute value is not zero then the feature is an extended feature and the decimal value is its serial value in the layer (to be referenced in dimension creation).
.feed	Integer	Feature	(0 to 100000) For a chained feature, this attribute sets the table feed rate when routing.
.fiducial_name	Text	Feature	(0 to 64) This attribute is used for **etec** output format. A pad that was given a fiducial name is used for registration between layers.
.fill_dx	Float	Symbol	(0.000001 to 50.0) This attribute is used as the default horizontal distance between symbols when the symbol is used for pattern filling.
.fill_dy	Float	Symbol	(0.000001 to 50.0) This attribute is used as the default vertical distance between symbols when the symbol is used for pattern filling
.flipped_of	Text	Step; Layer	This attribute defines a STEP as a flipped step. When attached to a LAYER, it indicates that the layer was created as a result of (layer) flipping. The attribute value is the name of the original (unflipped) layer. This is done in order to keep the elements of the original layer.
.flipped_out_of_date	Boolean	Step	**No** (default) = indicates that the flipped step is an accurate copy of the original step. **Yes** = indicates that the flipped step is no longer an accurate copy of the original step. One or the other has changed since the first flipping operation that created the step.
.foot_down	Text	Feature	Attached to feature it causes a **foot_down_cmd** to be generated by the Auto Rout Manager in the rout file just before the feature. Used only for Excellon files (ignored for other formats).
.fs_direction_bottom	Option	Step	(Left2Right; Right2Left; Top2Bottom; Bottom2Top). This attribute is used for the thieving pad check in assembly analysis. It determines the flow direction for the bottom layer. Thieving pad check is required for some components during the flow solder process
.fs_direction_top	Option	Step	(Left2Right; Right2Left; Top2Bottom; Bottom2Top) This attribute is used for the thieving pad check in assembly analysis. It determines the flow direction for the top layer.Thieving pad check is required for some components during the flow solder process.
.full_plane	Boolean	Feature	NOT USED

Attribute	Type	Entity	Description
.generated_net_point	Boolean	Feature	Openings in the solder mask covering the outer layer which expose locations that could be used as test points are inserted into the layer as rectangular, square or round pads, and marked with the attribute **.generated_net_point**.
.geometry	Text	Feature	(0 to 100) Contains the name of the padstack which created this feature. It is loaded during direct EDA translation. For layers which are created from component layers during the 'Draw to Layer' operation, the attribute will contain (for centroid pads) useful information on the component, package and part name.
.global_camtek_aoiset	Text	Job	(0-80) Contains the name of the AOIset to be assigned to each layer upon layer selection in the CAMTEK AOI Interface. Once a name is defined, the AOIset field in the CAMTEK popup will be filled with this name and a new AOIset created in the layer (if already exists, the AOIset will become the current set). The value in this attribute overrides the value defined in the configuration parameter **camtek_def_aoiset**, but if no value is specified in this attribute, the **camtek_def_aoiset** value will apply.
.gold_plating	Boolean	Feature	This attribute should be attached (manually) to features which are a part of a gold plated connector. It is used during auto-panelization to orient the gold plated area toward the extreme side of the panel.
.guard_comp	Boolean	Comp.	Assigned to a component that “guards” other components. If TRUE, this component is considered a "guard component" (that is, not likely to be knocked off the board accidentally. To be used in future actions.)
.hatch	Boolean	Feature	Assigned to hatched planes [filled with lines (hatches) or cross lines (cross-hatch) instead of solid copper]. The lines which make up the border and fill the surface are hatches.
.hatch_border	Boolean	Feature	The lines making up the border of a surface.
.hatch_serrated_border	Boolean	Feature	Assigned to features that are added for partial hatch. The difference between regular hatch and partial hatch is that in partial hatch the cells along the border that intersect the border line are filled; the feature(s) that fill these cells are assigned this attribute.
.hp3070_common_pin	Text	Comp.	(0-16). For the device SWITCH this is used to designate the COMMON pin.
.hp3070_contact_pin	Text	Comp.	(0-16). For the device SWITCH this is used to designate the CONTACT pin.

Attribute	Type	Entity	Description
.hp3070_device	Text	Comp.	(0 -16) The device of the component, one of the following: All other components will be categorized as Undefined. - CAPACITOR - CONNECTOR - DIODE - FET - FUSE - INDUCTOR - JUMPER PIN - LIBRARY - POTENTIOMETER - RESISTOR - SWITCH - TRANSISTOR - ZENER
.hp3070_fail_msg	Text	Comp.	(0-64) Specifies the failure message associated with the component. This applies to all device types. In output of HP3070 formats, the text field (within quotes) consisting of the Part number and this error message will be truncated to 40 characters.
.hp3070_hi_value	Float	Comp.	(0-100000.0) Specifies the upper test limit of the device. Its specific meaning is dependent on the device type. - For DIODE: Upper test limit, in volts for the diode's forward bias voltage. - For FET: The high resistance limit in ohms. - For TRANSISTOR: The high limit for the transistor beta.
.hp3070_lo_value	Float	Comp.	(0-100000.0) Specifies the lower test limit of the device. Its specific meaning is dependent on the device type. - For DIODE: Lower test limit, in volts, for the diode's forward bias voltage. - For FET: The low resistance limit in ohms. - For TRANSISTOR: The low limit for the transistor beta.
.hp3070_probe_access	Text	Comp.	(0-64) Specifies the probe access for the component and toeprint. This value will be applied to ALL the pins of the component. Known values are: PREFERRED, NO_PROBE, TOP, TOP NO_ACCESS, BOTH MANDATORY, and MANDATORY NO_MANUAL though other values are possible. If toeprints are assigned this attribute, their settings override the component setting.
.hp3070_seriesr	Float	Comp.	(0-100000.06) For INDUCTOR devices this is used to specify the series resistance (in Ohms).
.hp3070_test	Text	Comp.	Determines that a component be tested. This attribute applies to all device types. Devices of type CONNECTOR must be NT (Not Tested).

Attribute	Type	Entity	Description
.hp3070_tol_neg	Float	Comp.	(0-100) This is a real value expressing the percent of the value to use as a tolerance (negative tolerance). This is used for devices: - CAPACITOR - RESISTOR - INDUCTOR - ZENER - POTENTIOMETER
.hp3070_tol_pos	Float	Comp.	(0-100) This is a real value expressing the percent of the value to use as a tolerance (positive tolerance). This is used for the devices: - CAPACITOR - RESISTOR - INDUCTOR - ZENER - POTENTIOMETER
.hp3070_type	Text	Comp.	The type of device:. For CAPACITOR: - F = Capacitor Value is Fixed. - V = Capacitor Value is Variable. For FET: - N = N-Channel Field Effect Transistor - P = P-Channel Field Effect Transistor For INDUCTOR: - F = Inductor value is Fixed - V = Inductor value is Variable For JUMPER: - O or OPEN = Jumper is Open - C or CLOSED = Jumper is Closed For RESISTOR: - F = Resistor value is Fixed - V = Resistor value is Variable For TRANSISTOR: - N = Transistor is an NPN - P = Transistor is a PNP Range of characters: 0-8
.hp3070_value	Text	Comp.	(0-16) The value of the component. The meaning varies depending on the component device. For CAPACITOR it is used for capacitance (in Farads). For INDUCTOR it is the inductance (in Henries). For PIN LIBRARY it is used for the PN (Part Name). For the devices POTENTIOMETER and RESISTOR, it is used for the device's resistance. For the ZENER device it specifies the breakdown voltage (in Volts).

Attribute	Type	Entity	Description
.ignore_action	Boolean	Feature	This attribute can be assigned to individual features. Any feature possessing this attribute is ignored by the action. This attribute is useful if a specific feature has none of the other attributes defined in the ERF variable **v_ignore_attrs**. The **.ignore_action** attribute must be specified in the list of attributes defined in **v_ignore_attrs** to enable it.
.image_dx **.image_dy**	Float	Symbol	(-1.0 to 50.0) These values are set when inputting Image files into the system. They contain the datum point of an Image special symbol entity used to set the datum when performing output back into Image format. These values should not be changed by the user as this can cause data corruption.
.imp_line	Boolean	Feature	Assigned to lines which are impedance-controlled. When set, it prevents the lines from being rerouted or thinned during signal layer optimization.
.ind_orient_req	Boolean	Comp.	Indicates that the component requires silkscreen orientation indication. (To be used in future actions.)
.infeed_speed	Integer	Feature	(0 to 100000)
.inp_file	Text	Layer	(0 to 480) Contains the name of the file (Gerber, Drill) from which the data was input into the layer.
.inp_net_name	Text	Feature	(0 to 100) This attribute contains netlist information sent by the DPF input translator.
.inp_x_scale, .inp_y_scale	Float	Layer	(-9.99999 to 9.99999) These attributes are used in input and output for NEC format. During NEC input, the values of the GSCL NEC command are stored in them. The NEC output writes the GSCL command to the output file is the values are other than 1.
.is_burried	Boolean	Comp.	Assigned to buried components specifically input from CADIF files in order to mark them as buried. This attribute, although specifically designed for CADIF files, can be used in any other function or script. Note that the attribute name is misspelled, but that is its name.
.is_capped	Boolean	Feature	Used on via pads on top & bottom signal layers to indicate that the via is capped on this side.
.is_shadowed	Boolean	Comp.	Components with this attribute are considered for the Shadowing categories, as the shadowed component.
.jtag_component_id	Integer	Feature	(1 to 100) Component ID numbers are assigned to each JTAG feature using this attribute. All pads belonging to the same JTAG feature share the same ID number.

Attribute	Type	Entity	Description
.label_clearance	Boolean	Comp.	Assigned to components which are not allowed to be too close to a glued label (e.g. fine pitch SOIC components). During the component analysis, these components are checked vs. the label components.
.layer_class	Text	Layer	(0 to 1000) Specifies layer classification.
.layer_dielectric	Float	Layer	(0.0001 to 0.5 inch) Specifies the dielectric thickness below a layer.
.layer_hdi_type	Option	Layer	(Buildup; Core) Distinguishes buildup layers from core layers in HDI jobs. Some HDI categories are relevant to buildup or core layers but not to both. Therefore, it is important to set this value appropriately.
.local_fiducial_dist	Float	Comp.	(0.0 to 100.0) Defines the allowed distance of fiducials from the outline of the components which require local fiducials (See **.num_local_fiducial**). If set to 0, the fiducials must be included INSIDE the outline.
.lpol_done	Boolean	Layer	Indicates to the output that polarity sort according to a format has already been done during film optimization.
.lpol_surf	Boolean	Feature	Indicates surface modified by layer polarity reduction algorithm.
.merge_processes	Text	Step	A list of the last three merge actions in the order in which they were run. The list is updated each time a merge (BOM, Library, Board) is run. It is for informational purposes and does not have to be changed by the user.
.mount_hole	Boolean	Feature	Used on drill features to indicate that they are mounting holes.
.n_electric	Boolean	Feature	Assigned to a feature, defines it as non-electric (it is not considered for the current netlist for the step).
.naming_convention	Option	CAMTEK-AOISET	(Numeric; Layer name) When set to **Numeric** (default), the output directory for each layer is a number. When set to **Layer name**, output directory for each layer is the layer name.
.nec_cbnk_blank_name	Text	Layer	Contains blank records derived from CBNK records during NEC input translation.
.nec_n1_draw_num	Text	Layer	(0 to 20) Contains drawing number and version number derived from N1 records during NEC input translation.
.nec_n1_rev	Text	Layer	(0 to 2) Contains revision number derived from N1 records during NEC input translation.
.nec_n2_draw_num	Text	Layer	(0 to 20) Contains drawing number and version number derived from N2 records during NEC input translation.

Attribute	Type	Entity	Description
.nec_n2_rev	Text	Layer	(0 to 2) Contains revision number derived from N2 records during NEC input translation.
.nec_n3_edit_level	Text	Layer	(0 tof 1) min_length=0; max length=1 Contains editing level information derived from N3 records during NEC input translation.
.nec_n3_lyr_type	Text	Layer	(0 to 3) min_length=0; max_length=3 Contains layer type information derived from N3 records during NEC input translation.
.nec_n3_pol	Text	Layer	(0 to 1) min_length=0; max length=1 Contains polarity information derived from N3 records during NEC input translation.
.nec_n3_prod_rev	Test	Layer	(0 to 2) min_length=0; max length=2 Contains production revision information derived from N3 records during NEC input translation.
.nec_n3_target_layer	Text	Layer	(0 to 2) min_length=0; max length=2 Contains target layer information derived from N3 records during NEC input translation.
.needs_guarding	Boolean	Comp.	**True** - this component needs to be protected by guard components (see **.guard_comp**) else it is likely to be knocked off the board accidentally.
.net_point	Boolean	Feature	When assigned to a pad in an inner layer, defines the pad as an internal test point.
.net_type	Text	Net	(0 to 64) A name for the type of net. The **.net_type** attribute can reference the set of routing rules for a net.
.neutralization_angle	Float	Comp.	(0.0-360.0) An attribute attached to each **Rotation Neutralization** processed component stating the angle of rotation counter-clockwise from Valor standard orientation.
.neutralization_info	Text	Step	Attached to the step where **Rotation Neutralization** has been performed. This attribute contains the information **<CPL|CAD>;<DataCenter>;Site**. Site is read from configuration parameter **organization** of the computer where **Rotation Neutralization** was performed.
.neutralization_reviewed	Boolean	Comp.	Attached to each component in a package reviewed in **Rotation Neutralization**, i.e. a package not accepted automatically as being **Known** or **Safe** or by clicking **Accept Category**.
.neutralization_ss_layers	Text	Step	(0 to 200) Attached to the step where **Rotation Neutralization** has been performed. This attribute designates which layers are to be considered the silkscreen layers.
.nfp	Boolean	Feature	Indicates that a pad is not functional (applies to features).

Attribute	Type	Entity	Description
.no_fiducial_check	Boolean	Comp.	Components with this attribute are not checked for the “Component Covers Fiducial” category, or for any of the categories under the Coverage test.
.no_hole_under	Boolean	Comp.	If TRUE, no drill holes are allowed under this component.
.no_text_under	Boolean	Comp.	Assigned to a component, does not allow silk screen text to be placed under the component outline. Printed components (e.g. edge connectors) may not have this attribute.
.no_tp_under	Boolean	Comp.	Assigned to a component, does not allow testpoints to be placed under the component outline. Printed components (e.g. edge connectors) may not have this attribute.
.no_uncap_via_under	Boolean	Comp.	TRUE - uncapped vias are NOT allowed under this component.
.non_tp	Boolean	Feature	Assigned to a feature causes it NOT to be considered as a net testpoint. It is used for connectivity calculation but is not used as a test point (bare board testing).
.notest_req	Boolean	Feature	Any pad assigned with this attribute will not be tested. If it is tested by other means, drop back will be performed.
.num_local_fiducials	Integer	Comp.	(0 to 20) Defines how many local fiducials are expected to be inside or near a component. This is checked during Fiducial Analysis.
.numbered_layer	Text	Layer	(0 to 500) This attribute marks a layer as a numbered layer in PCB Numbering.
.orbotech_plot_stamp	Boolean	Features	NOT USED
.orig_surf	Integer	Feature	(0 -1000000) Identifies original surface which will be rebuilt.
.otherside_keepout	Option	Comp.	(full_area; pins_only) Defines for components whether the other side of the board may also contain components in the same area.
.out_angle	Option	Layer	(0.0; 90.0; 180.0; 270.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.
.out_break	Boolean	Symbol	Feature and symbol attribute. When assigned to a specific feature using a special symbol, the feature will be broken into it's primitives in the output translation stage, regardless of the settings of other output parameters. If the attribute is set for a special symbol (entity attribute) then all features that use these symbols will always be broken into primitive features in the output translation stage, regardless of the settings of any other output parameters
.out_comp	Float	Layer	(-100.0 to 100.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.

Attribute	Type	Entity	Description
.out_drill_full	Boolean	Step	The STEP entity attribute used by the Auto Drill Manager. This attribute can be used for drilling coupon STEPs that need to be fully drilled before continuing to the next step & repeat entity.
.out_drill_optional	Boolean	Step	Used by the 'Auto Drill Manager'. Both a STEP entity and feature attribute. If the drill feature is set with this attribute it will have the '/' command prefix in the final output file, indicating that the drill is optional. If a step entity attribute is set, then all the commands that are part of that step will have the '/' command prefixed. Thus, the whole step is optional.
.out_drill_order	Integer	Step	(-10000 to 10000) The STEP entity attribute used by the Auto Drill Manager. The attribute controls the order in which the steps will be drilled. Thus, who is first, second,,,etc. The attribute has the following valid values: 0 - no special order for that step 1 - first 2 - second 3 - and above - order from the beginning -1 - last -2 - one before last -3 - and on (drill order from the end)
.out_flag	Integer	Feature	(-1 to 1000000) Used in Excellon translation. If set is will dictate the dcode number.
.out_mirror	Boolean	Layer	Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the 'step' to be translated.
.out_name	Text	Step	(0 to 64) Entity attribute that is used by the Image output translator. If this attribute is not an empty string it will serve as the entity name on the Image system. If it is an empty string the original system entity name will be used. This attribute is important in cases where the Genesis name does not form a legal Image name. If this attribute is not set, the Genesis output translator decides about the new name with its own internal algorithm.
.out_nc_ignore	Boolean	Feature	Indicates a feature is not output during drill or rout process.
.out_nc_verify	Boolean	Feature	Prevents the output of drill/rout coupons. Features bearing this attribute are updated during drill/rout output procedures.
.out_polarity	Option	Layer	(Positive; Negative) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.

Attribute	Type	Entity	Description
.out_rout_optional	Boolean	Step	Used by the Auto Drill Manager. Both a STEP entity and feature attribute. If drill feature is set with this attribute it will have the '/' command in front of it in the final output file. This means that the drill is optional. If a step entity attribute is set then all the commands that are part of that step will have the '/' command at the beginning. Thus, the whole step is optional.
.out_rout_order	Integer	Step	(-10000 to 10000) STEP entity attribute used by the Auto Drill Manager. The attribute controls the order in which the steps will be drilled. Thus, who is first, second,,,etc. The attribute has the following valid values: 0 — no special order for that step 1 — first 2 — second 3 and above — order from the beginning -1 — last -3 and on — drill order from the end
.out_scale	Boolean	Symbol	Feature and symbol attribute. In the output translation package there is a special parameter that controls the way features will be scaled. In two of the options the user can specify whether certain features can be scaled or not. This is important in cases where special registration targets would not be scaled together with all the other features. This special output option applies only to features that have this attribute set. In case of a special symbol, the customer can set the attribute, and by this control the scaling of all features that use this symbol.
.out_x_scale	Float	Layer	(0.000001 to 5.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.
.out_y_scale	Float	Layer	(0.000001 to 5.0) Layer entity attributes with default values that are used by the output translator. These values populate the output screen when selecting the step to be translated.
.pad_usage	Option	Feature	(toeprint;via;g_fiducial;l_fiducial;tooling_hole) This attribute defines the specific usage of a pad. It is loaded during the direct EDA translation and by the attribute derivation script.
.patch	Boolean	Feature	Assigned to patches added by the pinhole elimination DFM action.
.pattern_fill	Boolean	Feature	Assigned to features which are added during a pattern fill operation, either manually or through the Copper Balance DFM action.

Attribute	Type	Entity	Description
.pilot_hole	Integer	Feature	(0 to 100000) Attribute assigned to pads that are pilot holes in a chain (holes that are drilled in each tool down in the chain rout path before routing the chain). Pilot holes are set from the chaining popup. The value of the attribute is the serial chain number to which the pilot hole belongs. When merging or inserting chains, the pilot holes are updated automatically.
.plated_type	Option	Feature	Defines plated hole type in drill layers using the Attributes popup or the Drill Tool Manager.
.pnl_class	Text	Step	(0-64) The value of the attribute is the name of the panel class whose parameters were used by the Automatic Panelization algorithm. Used only when the step is created by the Automatic Panelization Package.
.pnl_pcb	Text	Step	(0-64) The value of the attribute is the name of the panelized PCB whose parameters were used by the Automatic Panelization algorithm. Used only when the step is created by the Automatic Panelization Package.
.pnl_place	Text	Step; Feature	(0-64) Applies to STEP and FEATURE. The value of the attribute is the name of the placement rule used when an element was added to the panel overlay. Used only when an element is added to the panel overlay by the Automatic Panelization Package.
.pnl_scheme	Text	Step	(0-64) The value of the attribute is the name of the panelization scheme whose rules were used in creating the panel overlay. Used only when the panel step was created by the Auto Panelization Package.
.polarity_marker	Integer	Comp.	(1-10000) An attribute indicating which pin of the component is Pin 1. (Default=1)
.primary_side	Option	Job	(Top; Bottom) Indicates the primary side for this job.
.rotated_of	Text	Step	(0-64) Source step of a rotated step
.rotation_angle	Float	Step	(-360.0 to 360.0) Angle of rotation (in degrees) that this step was rotated (applies to steps)
.rout_chain	Integer	Feature	(0 to 100000) Contains the serial number of the chain to which the feature belongs. Features belonging to that chain are rearranged in the features database according to their order inside the chain. Additional attributes that are added to a chained feature: **.feed, .speed, .rout_flag, .comp**

Attribute	Type	Entity	Description
.rout_cutoff_feed	Integer	Feature	(0 - 100000) For a chained surface feature, defines the feed of the chain cutoff.
.rout_flag	Integer	Feature	(0 to 100000). For each chained feature this attribute represents a numeric value supplied to a chain to provide data for the automatic process of the Auto Rout Manager.
.rout_plated	Boolean	Feature	Indicates a plated feature on a rout layer. **Note:** The **.drill** attribute can still be used in rout layers, but the **.rout_plated** attribute takes precedence if both exist.
.rout_plunge_feed	Integer	Feature	(0 - 100000) For a chained surface feature, defines the feed of the chain plunge.
.rout_plunge_mode	Option	Feature	(none;straight;overlap;arc;diag;diag_ang) For a chained surface feature, defines the mode of the chain plunge.
.rout_plunge_val_a	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_b	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_c	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_d	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_e	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_f	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_v1	Integer	Feature	(0 - 90) (grad) For a chained surface feature, defines one the chain plunge parameters.
.rout_plunge_val_v2	Integer	Feature	(0 - 90) (grad) For a chained surface feature, defines one the chain plunge parameters.
.rout_pocket_direction	Option	Feature	(standard; opposite) For a chained surface feature, this attribute defines the rout direction of the chain pocket.

Attribute	Type	Entity	Description
.rout_pocket_feed	Integer	Feature	(0 - 100000) For a chained surface feature, defines the feed of the chain pocketing.
.rout_pocket_mode	Option	Feature	(none; concentric) For a chained surface feature, defines the mode of the chain pocket.
.rout_pocket_overlap	Float	Feature	(minus100.0 to plus 100.00) For a chained surface feature, defines one of the chain plunge parameters.
.rout_tool	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines the tool size for the outline and plunge rout path.
.rout_tool2	Float	Feature	(0.0 - 100.0) For a chained surface feature, defines the tool size for the pocket (rout to dust) rout path.
.rout_type	Option	Feature	(regular; pocket) For a chained feature, defines the type of chain.
.se_coupon	Option	Step	(none; drill; rout) Defines a step as a start/end coupon of a certain type.
.se_coupon_direct	Option	Layer	(0;90;180;270) Define the direction from the start point to the next point in start/end coupon.
.se_coupon_dist	Float	Layer	(0 - 10000) Distance is measured between drill/slot edges or centers in start/end coupon.
.se_coupon_dist_type	Option	Layer	(Spacing; Center) Distance is measured between drill/slot edges or centers in start/end coupon.
.se_coupon_max_size	Float	Layer	(0.0 - 10000.0) No verification holes will be created. for all tool sizes greater than this parameter (mils/microns).
.se_coupon_method	Option	Layer	(None; Auto; From Point) Drill/slot location calculation method in start/end coupon.
.se_coupon_min_hits	Integer	Layer	(0 - 10000) Start/end drill coupon. If number of drills of certain tool is less than required quantity, the verfication holes of this size will not be created.
.se_coupon_min_size	Float	Layer	(0.0 - 10000.0) No verification holes will be created. for all tool sizes less than this parameter (mils/microns).
.se_coupon_mode	Option	Step	(Start_End; Start; End) Defines the start/end coupon mode.
.se_coupon_order	Integer	Step	(1 - 100) Sequential order of the start/end coupon steps of the same type and mode.

Attribute	Type	Entity	Description
.se_coupon_slot_angle	Float	Layer	(0.0 - 360.0) Slot angle in start/end coupon (degrees).
.se_coupon_slot_length	Float	Layer	(0.0 - 10000.0) Slot length for start/end coupon (mils/microns).
.se_coupon_split_num	Option	Step	(1;2) A split number the start/end coupon belongs to.
.sequential_lamination	Boolean	Layer	(No, Yes)
.shave	Boolean	Feature	Assigned to all the shaves (negative merges) that the silk screen optimization adds in merge mode.
.sliver_fill	Boolean	Feature	Assigned to all the fills added by the sliver fill DFM actions.
.smd	Boolean	Feature	Assigned to outer layer pads designated as toeprints which are lands for SMD components. It is set by the 'Set SMD Attribute' Cleanup Action.
.smt_direction_bottom	Option	Step	(Left2Right; Top2Bottom; Right2Left; Bottom2Top) Defines the direction of the SMT process flow on the bottom side.
.smt_direction_top	Option	Step	(Left2Right; Top2Bottom; Right2Left; Bottom2Top) Defines the direction of the SMT process flow on the top side.
.source_llayer	Text	Feature	(0-64) This attribute is used by the Enterprise Mentor EDA translator to identify the "Source Logical Layer" of features (traces) appearing on signal or mixed layers. The translator uses this attribute in a filtering stage that addresses pad/ signal mapping.
.source_name	Text	Step; Symbol	(0-64) The name of the source step (or symbol) of a flipped step (or symbol).
.spacing_req	Float	Feature	(0.0 > 100.0) Specifies required spacing from a feature.
.speed	Integer	Feature	(0 to 100000) For a chained feature this attribute sets the spindle speed (in revolutions per minute) when routing.
.spo_h_fact	Integer	Feature, Comp.	(0.3 <-> 2.0) When **.spo_h_mode** = Factor, **.spo_h_fact** specifies the factor by which paste pad heights are sized relative to their SMD pads. For example, 0.9 means height is 90% of SMD pad.
.spo_h_mode	Integer	Feature, Comp.	(values = Distance, Factor, Value) Defines how heights of paste pads are sized: by distance, factor or value.

Attribute	Type	Entity	Description
.spo_h_val	Integer	Feature, Comp.	(-500 to +500) When **.spo_h_mode** = Distance, **.spo_h_val** is the reduction/expansion of the paste pad width relative to the SMD pad width. For example, **.sp_h_val** = 5.0 mils shrinks paste pad by 5.0 mils (2.5 mils on each side) relative to SMD pad width. Positive number results in smaller paste pad, negative number in larger paste pad. When **.spo_h_mode** = Value, **.spo_h_val** becomes the absolute width of the paste pad (for example, 5.0 mils becomes the actual width of the paste pad).
.spo_move_center	Integer	Feature, Comp.	(range: -500 to 500) To move the paste pad from the SMD pad center. A positive value will move the paste from the component center out. A negative value will move the paste towards the component center.
.sr_pcb	Boolean	Step	(0.0 to 1000.0) Indicates the name of the pcb step placed in the panel by automatic panelization.
.src_orientation	Integer	Comp.	(-1 to 3) Defines the zero orientation of this component relative to its orientation in the packages database. That is, the orientation of the component on the automated assembly tape, or (for manually inserted components) the orientation in which pin #1 is "in the same position" for all similar components.
.step_numbering	Text	Feature	(0 to 500) Text features used for PcB numbering are assigned this attribute.
.string	Text	Feature	(0 to 1000) For nomenclature features, the value of this attribute is the original text string which the feature is part of. During EDA input, all occurrences of the asterisk character '*' are replaced by the hyphen character '-' (the reason: when filtering, the asterisk character '*' is used to denote any substring match).
.string_angle	Float	Feature	(0 to 360) For nomenclature features, the value of this attribute is the original (in the input file) text rotation angle which the feature is part of. This attribute is assigned to translated jobs of formats: Cadence Allegro BRD/APD, Mentor BoardStation, PADS PowerPCB.
.surface_outline_widths	Float	Feature	(000.1 > 100.0) Assigned to area shapes created from closed polylines. Value=width of the dource polyline.
.tampering_feature	Boolean	Feature	(No, Yes If set, indicates a tapered feature.

Attribute	Type	Entity	Description
.tear_drop	Boolean	Feature	Assigned to features which are added during a tear drop operation, either manually or through the Teardrop Creation DFM action.
.test_point	Boolean	Feature	Assigned to features which are used for In-Circuit Testing operations. It is loaded during the direct EDA translation and is used during the Testpoint Analysis action. Also supported in Zuken BD.
.test_potential	Option	Feature	An attribute attached to features being considered as testpoints (potential testpoints) for In-Circuit Testing operations. It is assigned either by the Testpoint Allocation Action or manually by the user. **potential_tp_by_analysis** - a feature meeting all criteria of the Testpoint Allocation Action. **potential_tp_manually** - a feature to be used as a testpoint though it does not meet all criteria. **not_potential_tp_manually** - a feature not to be used as a testpoint even though it meets all criteria.
.test_req	Boolean	Feature	(No, Yes) Must test any pad marked with this attribute. If the test fails, drop back is performed.
.text	Text	Symbol	(0 to 1000) Size of text content.
.text_line_width	Float	Symbol	(0.0 to 100.0) Text line width.
.text_rotation	Float	Symbol	(0.0 degrees to 360.0 degrees) Angle of text rotation.
.text_x_size	Float	Symbol	(0 < size < 0.2 inches) Text character size in the X dimension. Relevant if text type = ‘string’.
.text_y_size	Float	Symbol	(0 < size < 0.2 inches) Text character size in the Y dimension. Relevant if text type = ‘string’.
.thvpad_required	Boolean	Comp.	Assigned to components which require a thieving pad check during the Padstack Analysis action (e.g. fine pitch SOIC).
.tie	Boolean	Feature	(No, Yes)
.tiedown	Boolean	Feature	(No, Yes)
.toep_spacing_req	Float	Comp.	(1 to 500) Assigned to components for reporting in the Toeprint to Toeprint category in the Signal Layers Check in Analysis. It defines the maximum spacing within which to report pad to pad spacing measurements. Units: inch/mm
.tooling_hole	Boolean	Feature	Used on drill features to indicate that they are tooling holes.

Attribute	Type	Entity	Description
.transform_data	Text	Step	This attribute is necessary for rebuilding dependent steps. If the attribute exists in the step, Genesis saves the data necessary for rebuilding dependent steps, and enables the automatic update of dependent steps. If the attribute does not exist (old jobs), automatic update is canceled. The data necessary for rebuilding dependent steps is not saved.
.via_type	Option	Feature	(Drilled; Laser; Photo) Assigned to via drills for the classification of various via pad and via drill categories in the HDI analysis.
.viacap_layer	Option	Step	(Top; Bottom; Both; None) Defines on which layer via capping can occur, if any.
.wheel_type	Option	Wheel	(Gerber; Tools) Specifies for a wheel whether it is used for Gerber files translation or for drill file translation.

Appendix C *Frequently Asked Questions*

C.1. Why is the database in ASCII?

An ASCII database provides the user with numerous advantages:

- It is easy to read and understand
- Translators to and from the database formats are easier to write
- The data is portable between different architectures, independent of byte order, floating point formats, etc.

By compressing the ASCII files using standard compress commands, the size of the data is even smaller than the binary equivalent! This is due to the fact that the compression algorithm is adaptive and work very well when certain strings are repeated.

C.2. When I wish to rotate a feature pad by 90 degrees is the aperture rotated left or right?

Clockwise.

C.3. Regarding donuts, butterflies, thermals, and moires, do any of these symbols have negative components?

Standard symbols are all positive. All holes in symbols are see-thru by definition.

C.4. When the start and end-points of a feature coincide, is this considered a 360-degree arc or a single point? Can I draw an arc with a square symbol?

A 360-degree arc; there are no single point arcs in the ODB++ database. Arcs can be drawn only with a round symbol.

C.5. When I specify an x,y location for text where will the text string be located?

The x,y coordinates will determine the bottom left corner position of the first character of the text string.

C.6. What is the meaning of the optimize field in a netlist file?

It indicates that the net has been optimized by the Netlist Optimizer function and the end-point markers have been removed from mid-points.

C.7. In a netlist file, how is the radius field supposed to be set for drills of 0.002 inches thru non-SMDs?

The radius field will be 0.001 mils in this case.

C.8. In a netlist file, what does the term staggered points mean?

These are points that have been staggered by the staggering algorithm to make them accessible to test probes.

C.9. *For rectangular thermals can I define spoke angles at other than multiples of 45 degrees?*

Rectangular thermals cannot have spoke angles of 45 degrees, only square/round thermals can have angles that are non-multiples of 45 degrees.

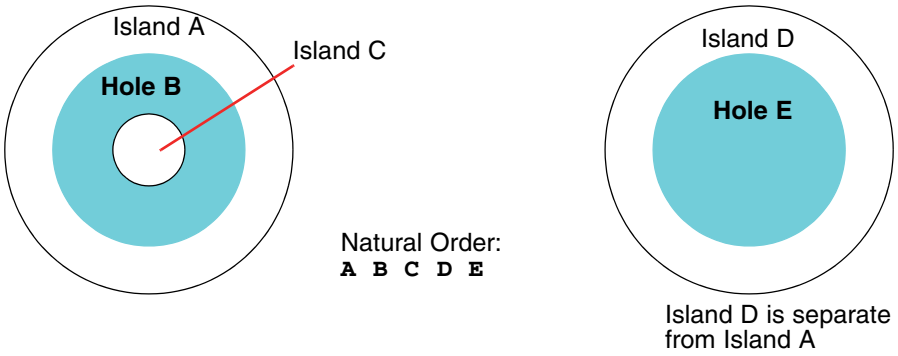
C.10. *Can feature files of user-defined symbols contain references to other user-defined symbols?*

Yes, they can. But recursion, direct or indirect is not allowed.

C.11. *Regarding surfaces, is there a particular order in specifying holes and islands?*

The order of containment must be preserved. Islands precede holes that are contained in them. Holes precede islands that are contained in them.

Take, for example, the following containment order:



C.12. *Regarding surfaces, does the outermost island come first?*

Yes.

C.13. *Regarding rounded or chamfered rectangles, how do I specify corners?*

Corners must be specified in ascending order, starting from #1 the top-right corner going counter-clockwise (that is, top-left corner=2, bottom-left=3, etc.).

C.14. *If I want to offset e a rectangular pad in X or Y, should I a symbol and use the standard valor definitions to create the offset. As an example, suppose we have a rect pad 70x50 with an X offset of 5. The feature file for the symbol I create would contain:*

```
#
# Symbol name
#
$0 rect70x50
#
# Pad definition
#
P .005 .0 0 P 0 0
```

Is this correct?

Instead of defining your own symbols with offsets, you should use the standard rect symbol and offset the coordinate that references it (in the layer features file).

C.15. Why are user-defined symbols not scalable? This means that for every pad size which does not fit in the standard I will have to create a new symbol.

Yes, that is right. User-defined special symbols cannot be scaled as standard symbols can. You need to create a new symbol for each set of parameters (make the name signify the dimensions of the symbol, such as: rect70x50, rect50x30, etc.). See "Symbols" on page 22 for further details.

C.16. Regarding properties (PRP) on components, is there a list of properties that are recognized by the system (such as with system attributes). Where can I find it? The same goes for PRP in the `eda/data` file.

There is no list of predefined properties in the ODB++ database. These are EDA-specific.When we input Mentor data we read all the properties of the components in the data. These properties are shown when displaying a component in the Graphic Station. They can also be used to automatically set an attribute by calling a function that maps properties to attributes.

C.17. When defining a PKG record (using `/steps/step_name/eda/data`), it seems that ODB++ expects closed geometries.Is it critical to have only closed elements?

Yes, you should close all polygons.

C.18. In what order should the matrix layers be in?

The layers should be ordered according to the stackup of the board: i.e. comp_+_top, sigt .. sigb, comp_+_bot, drill, drill_1to5, etc.

C.19. I have problems with the `/steps/step_name/eda/data` in the PKG section?

A PKG record must be followed (the next line) by an outline record. check for some PKG records that have PRP's before the outline and make sure you have an outline defined.

C.20. How are the net_num records numbered?

The **`net_num`** used in the TOP record corresponds to the sequence of the Net records in the **`eda/data`** file. The first Net record is **`net_num 0`**, the second is **`net_num 1`** and so on.

C.21. We have an elaborate tool set to define routing slots and milling contours. How is this data written to ODB++?

Milling (referred to as routing in Fabrication, and not to be confused with routing of traces in design) is handled by defining a 'rout' layer (similar to a drill layer). The features in this layer correspond to the outline of the shapes that need to be cut out. A rout layer is like any other layer, but in order for it to be used during fabrication, should contain only lines, arcs and circular pads (rout machines can also drill).

C.22. What net do I assign to points that have no net defined?

All features which do not have a net defined should be assigned to net **`$NONE$`**which should be defined in the eda/data file. You must add the net and all it's points to the CADnet.

C.23. Is there any restriction on the maximum line length in an ODB++ file? Can comment lines be more than 500 characters?

The restrictions are different for different files. In general 500 characters are the limit, but there are exceptions. Any line over the defined limit will be read with the remainder of the line ignored, so comments can be longer than the limit.