

GenCAD Specification

1.4

CIMBridge 97™

mitron

Document Revision: 1.4/A

Confidential and Proprietary

© 1990-1997 by Mitron Corporation. All rights reserved.

Disclaimer

GenCAD specifications, including interface specifications, are proprietary to Mitron. Mitron hereby grants a royalty-free license to others to develop software and hardware products which conform to or interface with such specifications. Mitron, at its own discretion, may change GenCAD specifications from time to time and may modify or cancel this license, and Mitron assumes no liability associated with such changes, including any claims based on reliance based on the previous versions. MITRON MAKES AND LICENSEE RECEIVES NO WARRANTY, EXPRESS OR IMPLIED, AND THEREIN EXPRESSLY EXCLUDED ALL WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. MITRON ASSUMES, AND SHALL HAVE, NO LIABILITY WITH RESPECT TO THE PERFORMANCE OR USE OF THE SPECIFICATIONS OR THE LICENSE GRANTED HEREIN FOR DIRECT, CONSEQUENTIAL, EXEMPLARY, OR INCIDENTAL DAMAGES EVEN IF IT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from Mitron Corporation. Permission is hereby granted to licensed users of Mitron software to print and/or store in electronic format this documentation solely in support of using Mitron software. No liability is assumed for damages resulting from the use of information contained in this document. Mitron reserves the right to change specifications without notice.

Mitron, CIMBridge, CIMBridge 97, CIMTrack, GenCAD, CB/Pro, CB/Doc, CB/Plan, CB/Expert, SmartBOM, CB/Test, PQM, and TRACS III are trademarks of Mitron Corporation. GenCAD is a trademark of Mitron Corporation. All other brand or product names are trademarks or registered trademarks of their respective companies or organizations.

Mitron Corporation is a subsidiary of GenRad, Inc.

Mitron Corporation
10795 SW Cascade Blvd.
Portland, Oregon 97223
(503) 624-1776

CIMBridge 97/A

Confidential and Proprietary

GenCAD Specification 1.4

Contents

Chapter 1. Introduction

Overview of the GenCAD Specification	1
Character Rules	1
Keyword Rules	1
Parameter Rules	2
Coordinate Rules	2
Case Sensitivity	2
General Rules	2

Chapter 2. The GenCAD File

GenCAD File Description	3
The HEADER Section	5
The BOARD Section	7
The PADS Section	11
The PADSTACKS Section	13
The ARTWORKS Section	15
The SHAPES Section	17
The COMPONENTS Section	22
The DEVICES Section	27
The SIGNALS Section	32
The TRACKS Section	35
The LAYERS Section	36
The ROUTES Section	38
The MECH Section	43
The TESTPINS Section	47
The POWERPINS Section	49
The PSEUDOS Section	51
The CHANGES Section	52

Chapter 3. The Panel File

Panel File Contents 59

The HEADER Section 60

The PANEL Section 62

The ARTWORKS Section 65

The MECH Section 67

The BOARDS Section 70

The CHANGES Section 72

Appendix A.Parameter Definitions

Appendix B.Mirror and Rotate Diagram

Appendix C.Version Change Information

Version 1.4 81

Version 1.3 82

Version 1.2 83

Version 1.2 draft 83

Version 1.1 85

Version 1.0 85

Chapter 1

Introduction

Overview of the GenCAD Specification

This specification describes a generic Computer Aided Design (CAD) file, called GenCAD, that contains physical design information for electronic circuit boards or hybrids. The GenCAD file format is the standard input format for Mitron's CIMBridge Manufacturing Framework of software applications used in the assembly and testing of printed circuit boards. Version 1.4 of the GenCAD specification applies to CIMBridge version 3.4 or above. Several CAD systems output the GenCAD format. End-users and other third party software developers are welcome to use this specification for use with CIMBridge or other systems.

In addition to defining the GenCAD file for board design information, this specification defines the Panel file, which describes the panel on which one or more GenCAD boards are placed. These are text based files made up of keywords and parameters. The keywords identify the type of data and the parameters contain the data.

Character Rules

The text used in the file must be ASCII characters in the range *space* (ASCII decimal 32) to *tilde* (ASCII decimal 126). The only control characters allowed are *carriage return* (ASCII decimal 13) and *line feed* (ASCII decimal 10). All other control characters, including *tab* (ASCII decimal 9), and *delete* (ASCII decimal 127) are **not** allowed. Each text line, including the very last line in the file, must be terminated with at least one line feed character. Carriage returns are optional. Extra line feeds and carriage returns can be included when the file is generated but will be ignored when the file is read.

A text end of file marker, such as *Ctrl Z* (ASCII decimal 26), should not be included in the file data.

Keyword Rules

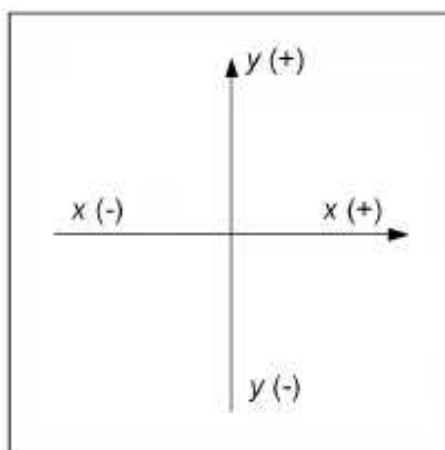
All lines (except blank lines) must start with a keyword and not be preceded by any spaces. All keywords must be in uppercase and separated from the parameters by a space (ASCII decimal 32).

Parameter Rules

Each section of the GenCAD or panel file is described later in this specification. Parameters in brackets $\langle \rangle$ are described in Appendix A. Parameters described as *free fields* allow the user to enter text of any suitable syntax. *Fixed fields* must contain parameters as described in Appendix A. The few optional parameters to keywords are shown inside square brackets like this: [$\langle \text{optional} \rangle$]. Each parameter must be separated by a *space* (ASCII decimal 32).

Coordinate Rules

The GenCAD and panel file uses Cartesian coordinates. When viewing the top of the board or panel, the positive direction for x coordinates is left to right (west to east), and the positive direction for y coordinates is bottom to top (south to north).



Case Sensitivity

The keywords and fixed fields must always be in uppercase letters. All free fields—such as component names—can be upper- or lowercase, but are always case sensitive. For example the component names C1 and c1 are two different components.

General Rules

All data is explicitly defined; there is no implied change of data. For example, placing a component on the bottom of the board does not imply that its shape is mirrored. The shape must be given a mirror parameter if it must be mirrored.

Chapter 2

The GenCAD File

GenCAD File Description

The GenCAD file can have any filename that the user chooses provided that it is acceptable to the operating system(s) in use. The GenCAD file may consist of some or all of the following sections in any order. Each section can appear only once in the file.

Table 1: GenCAD File Information

GenCAD Sections	Meaning
HEADER	General information about the board
BOARD	Outline, cutouts, and other features of assemblies.
PADS	Description of pin copper shapes.
PADSTACKS	Definition of pad collections.
ARTWORKS	Non-electrical feature definitions.
SHAPES	Physical outline and pin information of components.
COMPONENTS	Instance of a device, including location and orientation.
DEVICES	Unique component type information.
SIGNALS	Connectivity information.
TRACKS	Trace width definitions.
LAYERS	Board names and layer set definitions.
ROUTES	Track and copper shape information.
MECH	Mechanical device description and fixture/tooling hole information.
TESTPINS	Test point information for test probes.
POWERPINS	Test point information for power injection probes.
PSEUDOS	List of aliases for old data. This feature is now obsolete.
CHANGES	Modifications to the above selections.

The following items are not explicitly handled by the GenCAD file:

- Silk screen and solder mask information. Silk screen should be represented as artwork attached to part shape or component. Solder mask should be pads in padstack or artwork on Soldermask layer.
- Shaped and pattern filled tracks.
- Track impedances.
- Text that is part of a copper track (route).
- Off-grid data, i.e., items that do not have x,y coordinates.
- Comment fields, i.e., they are not retained in the GenCAD file.
- Hole plating. The SIGNAL section defines the electrical connectivity.
- Offsets for test or power pin placement on pads, vias, or pins.
- Offsets to pads in a pad stack.
- Track and pad separation data.
- Pad-to-pin number conversion. For example, a DIL relay with four physical pins and pad numbers 1, 2, 3, and 4 at pin positions 1, 7, 8, and 14.
- Tracks that are off center from the x,y coordinates.
- Cutouts in the outer layers that reveal the inner layer.
- Boards that use a combination of dimensional units. For example, a board designed in thousandths of an inch with an overlaid section designed in hundredths of a millimeter.
- Text fonts are not supported. A bounding rectangle is defined that specifies the area in which the text fits.
- Dimensional scaling is not allowed.
- Multiple shape definitions for one package style.

The HEADER Section

The HEADER section must be included in the file and must contain the GenCAD keyword. All of the keywords shown below may appear.

```
$HEADER
GENCAD <number>

USER <string>
DRAWING <string>
REVISION <string>
UNITS <dimension>
ORIGIN <x_y_ref>
INTERTRACK <number>
ATTRIBUTE <attrib_ref>
$ENDHEADER
```

\$HEADER and \$ENDHEADER mark the HEADER section of the GenCAD file.

The following paragraphs describe each of the keywords that may appear in the HEADER section.

GenCAD <number>

The GenCAD keyword defines the file type as a GenCAD file. The GenCAD keyword must be followed by a fixed text field that defines the version of GenCAD file specification used to build the file. The version number of this specification is 1.4.

USER <string>

When the GenCAD file is created by using a translation program, the USER keyword usually contains the company name and serial number of the translation program licensee. If the GenCAD file is produced by any other method, this free field can contain any text identifying the originator of the GenCAD file.

DRAWING <string>

The DRAWING keyword is followed by a free text field for the user to specify a number or title of the board.

REVISION <string>

The REVISION keyword is a free text field for the user to define a revision, issue, or modification status of the board.

UNITS <dimension>

The UNITS keyword is a fixed format field used to define the size of the dimensional units used throughout the GenCAD file. See [Appendix A beginning on page 73](#) for the <dimension> parameter definitions.

ORIGIN <x_y_ref>

The ORIGIN keyword defines the CAD coordinates of the origin of the board. The CAD system usually defines the origin at coordinates 0, 0.

If all coordinates are later offset, the ORIGIN parameter will be changed to reflect this offset from the original CAD data. Consider, for example, a square board that has coordinates of -100, -100 in the lower left-hand corner. If the board is offset so that all coordinates are positive, the CAD origin moves to +100, +100, and the ORIGIN keyword will have relative x,y coordinates of +100, +100.

INTERTRACK <number>

The INTERTRACK keyword is used by programs to record implemented change information. The parameter must be set to zero when the GenCAD file is first produced, and will be updated every time changes are incorporated into the main sections (but not when a CHANGE section is added to the \$CHANGES section as described on [page 52](#)).

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the HEADER section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

HEADER Section Example

Below is an example of a HEADER section:

```
$HEADER
GENCAD 1.4
USER "Mitron Europe Ltd. Serial Number 00001"
DRAWING "Modem C100 motherboard 1234-5678"
REVISION "Rev 566g 20th September 1990"
UNITS USER 1200
ORIGIN 0 0
INTERTRACK 0
ATTRIBUTE alpha m_part "BIS 9600"
ATTRIBUTE alpha m_desc "Issue 2"
$ENDHEADER
```

The BOARD Section

The BOARD section defines the outer shape and any internal cutouts of the printed circuit board in terms of absolute x and y coordinates as viewed from the side designated TOP. The BOARD section must be included and can use some or all of the following keywords.

```
$BOARD
THICKNESS <number>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

CUTOUT <string>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

MASK <string> <layer>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

ARTWORK <string> <layer>
TRACK <track_ref>
FILLED <filled_ref>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
TEXT <x_y_ref> <text_ref>
ATTRIBUTE <attrib_ref>
$ENDBOARD
```

\$BOARD and **\$ENDBOARD** mark the BOARD section of the GenCAD file. Each cutout description starts with the CUTOUT keyword; each mask off area description starts with a MASK keyword; and each ARTWORK feature description starts with an ARTWORK keyword. Parameters for keywords are defined in Appendix A.

THICKNESS <number>

The optional THICKNESS keyword defines the thickness of the board in <dimension> units as specified in the HEADER section. This approximate dimension may not be available from the CAD data.

LINE <line_ref>

The LINE keyword defines a straight line that forms part of the outer edge of the board, the internal edge of a cutout, or the outer edge of a masked off area.

ARC <arc_ref>

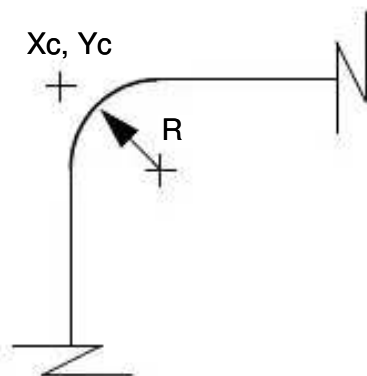
The ARC keyword defines a circular curve that forms part of the outer edge of the board, the internal edge of a cutout, or the outer edge of a masked off area.

Consider a rectangular board with corner radii of radius R units. If the top left hand corner of the rectangular board has the coordinates X_c Y_c , the BOARD section for this corner could be calculated as shown below. Note the counterclockwise arc definition.

```

LINE    ...    ...     $X_c+R$      $Y_c$ 
ARC      $X_c+R$      $Y_c$      $X_c$      $Y_c-R$      $X_c+R$      $Y_c-R$ 
LINE     $X_c$      $Y_c-R$     ...    ...

```

**CIRCLE <circle_ref>**

The CIRCLE keyword defines the outer edge of a circular board, the internal edge of a cutout, or the outer edge of a masked off area.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines the outer edge of a rectangular board, the internal edge of a cutout, or the outer edge of a masked off area.

CUTOUT <string>

The CUTOUT keyword is used to name and define an internal area of the board that has had all layers cut away. If the CAD system does not provide a unique name for each cutout, the GenCAD file must use names such as cutout1, cutout2, etc. The LINE, ARC, CIRCLE and RECTANGLE keywords following a CUTOUT keyword describe the cutout shape. Circular cutouts can alternatively be defined in the MECH section as drilled holes.

MASK <string> <layer>

The MASK keyword is used to name and define an area of the board; anything outside that area is inaccessible to test pins. If a unique name is not provided for the masked off area, the GenCAD file must use names such as mask1, mask2, etc. The LINE, ARC, CIRCLE and RECTANGLE keywords following a MASK keyword describe the masked off shape within which test pins can be placed. Several masked areas can be defined for different fixtures, even if they overlap, as long as they have different names.

An inner area can be masked off by creating a second mask description with the same name. This second mask description must be entirely inside the first. Placing a second mask within the first acts as a toggle, and the area within the second mask becomes inaccessible to test pins. Additional embedded masks also toggle accessibility—a third masked off area within the two others would be accessible to probes.

The <layer> parameter must be the layer for which the mask applies: TOP, BOTTOM, ALL, etc.

ARTWORK <string> <layer>

The ARTWORK keyword is used to name and define an artwork feature of the board. If a unique name is not provided for the artwork feature, the GenCAD file must use names such as artwork1, artwork2, etc. The FILLED, TRACK, LINE, ARC, CIRCLE, RECTANGLE, and TEXT keywords following an ARTWORK keyword describe the artwork feature. Several artwork areas can be defined for different features—even if they overlap—as long as they have different names. The FILLED keyword is used to identify that the following LINE, ARC, CIRCLE, and RECT form an enclosed area or just line artwork.

The <layer> parameter must be the layer for which the mask applies: TOP, BOTTOM, ALL, etc.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the BOARD section. The attribute will apply to the previously defined BOARD, CUTOUT, MASK, or ARTWORK. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

TRACK <track_ref>

The TRACK keyword identifies the track type as defined in the TRACKS section. It can appear at any time in an artwork description where a track changes width within an artwork feature.

FILLED <filled_ref>

The FILLED keyword defines that the following features form an enclosed artwork area. It can appear at any time within an artwork description where a filled or nonfilled area is to be defined. Acceptable values are YES or 0.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to define any text string, size, and location attached to the component. Typically this will be the <component_name>.

The <x_y_ref> parameter must be the *x,y* coordinate of the bottom left hand corner of the text string relative to the origin of the component origin (i.e., the <x_y_ref> of the PLACE keyword). The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

A BOARD Section Example

Below is an example of the BOARD section:

```
$BOARD
LINE 1000 2000 1200 2000
ARC 1200 2000 1200 3000 1180 2500
LINE 1200 3000 1000 3000
LINE 1000 3000 1000 2000
CUTOUT TRANSFORMER_HOLE
CIRCLE 1180 2500 20
ATTRIBUTE board mill "tool 255"
MASK Fixture_1
LINE 1005 2005 1195 2005
ARC 1195 2005 1195 2995 1195 2500
LINE 1195 2995 1005 2995
LINE 1005 2995 1005 2005
ARTWORK ORIGIN_MARKER TOP
TRACK 10
FILLED YES
LINE -100 0 100 0
LINE 0 -100 0 100
$ENDBOARD
```

The PADS Section

The PADS section is used to describe the shape of all the pads used on the printed circuit board. The PADS section must be included, even if only a default pad is described and used for all pads. The keywords used in the PADS section are:

```
$PADS
PAD <pad_name> <pad_type> <drill_size>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>
$ENDPADS
```

\$PADS and \$ENDPADS mark the PADS section of the GenCAD file. Each pad description must start with a PAD keyword.

The layer in which a pad lies is defined in the SHAPE section of the GenCAD specification.

The pad is **always** placed on a shape at the pad origin, or in a pad stack at the pad stack origin.

PAD <pad_name> <pad_type> <drill_size>

The PAD keyword defines each pad type. All the parameters must be included.

The <pad_name> parameter is a free field text string that must contain a unique pad name. This name must be used throughout the GenCAD file for all references to this pad type. If the CAD system does not define pad names or just uses a sequenced order, then names such as pad1, pad2, etc., must be used.

The <pad_type> parameter is a fixed field text string that must be the pad type as define in Appendix A. If the pad type is not explicitly defined then this parameter should default to POLYGON.

The <drill_size> parameter must always define the drilled hole size (diameter) of the pad in <dimension> units. A zero drill size means that there is no hole through the pad. If the drill size is not defined a <drill_size> of -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The PAD keyword can be used to describe just a drilled hole.

LINE <line_ref>

The LINE keyword defines a straight line that makes up the outer edge of the pad shape. The coordinates given must always be relative to the origin of the pad. If the pad has a hole, then the hole's center is **always** at the origin of the pad.

ARC <arc_ref>

The ARC keyword defines a circular (or elliptical) arc that makes up the outer edge of the pad shape. The coordinates given must be relative to the origin of the pad. If the pad has a hole, then the hole center is **always** at the origin of the pad.

CIRCLE <circle_ref>

The CIRCLE keyword defines a circular pad. The coordinates given must be relative to the origin of the pad. The drill location for a PADSTACK is always at the PADSTACK origin (i.e., there is no way to offset a drill.)

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines a rectangular or square pad. The coordinates given must be relative to the origin of the pad. The drill location for a PADSTACK is always at the PADSTACK origin (i.e., there is no way to offset a drill.)

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the PADS section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A PADS Section Example

```
$PADS
PAD p0101 FINGER 32
LINE 100 50 -100 50

ARC -100 50 -100 -50 -100 0
LINE -100 -50 100 -50
ARC 100 -50 100 50 100 0
PAD p1053 ROUND 20
CIRCLE 0 0 30
PAD p2034 BULLET 32
ARC 0 -50 0 50 0 0
LINE 0 50 -100 50
LINE -100 50 -100 -50
LINE -100 -50 0 -50
PAD d_hole_50 ROUND 50
CIRCLE 0 0 25
PAD 3 RECTANGULAR 0
RECTANGLE -5.2 -5.2 10.4 10.4
$ENDPADS
```

The PADSTACKS Section

The PADSTACKS section is optional, and is used to describe how a group of pads are arranged. The keywords used in the PADSTACKS section are:

```
$PADSTACKS
PADSTACK <pad_name> <drill_size>
PAD <pad_name> <layer> <rot> <mirror>
ATTRIBUTE <attrib_ref>
$ENDPADSTACKS
```

\$PADSTACKS and \$ENDPADSTACKS mark the PADSTACKS section of the GenCAD file. Each pad stack description must start with a PADSTACK keyword.

PADSTACK <pad_name> <drill_size>

The PADSTACK keyword defines a name for each pad stack.

The <pad_name> parameter is a free field text string that must contain a unique pad stack name. This name must be used throughout the GenCAD file for all references to this pad stack. The pad stack name cannot be one of the pad names defined in the PADS section. If the CAD system does not define pad stack names, or just uses a sequenced order, then names such as

padstack1, padstack2, etc., must be used instead.

The <drill_size> parameter must always define the drilled hole size of the pad stack in <dimension> units. This size will override any drill sizes used for the individual pad definitions. A drill size of 0 (ASCII decimal 48) means that there is no hole through the pad stack. If the drill size is not defined, a <drill_size> of -1 (ASCII decimal 45 and ASCII decimal 49) must be used. A <drill_size> of -2 can be used if the pad stack drill size is to be the same size as the first pad defined in the pad stack.

PAD <pad_name> <layer> <rot> <mirror>

The pad keyword is used to list the pads in the pad stack. All pads in the stack are placed on top of each other so that all pad origins are at the same x,y coordinate on the board.

The <pad_name> parameter is the pad name as defined in the PADS section.

The <layer> parameter is the board layer on which the pad is placed. Layer parameters of TOP, INNER, BOTTOM, and ALL are allowed. Specific inner layers, as defined in the LAYERS section (e.g., INNER3, INNER4), can be used instead of INNER. LAYERSETs may also be used when the same pad is used on several layers. The actual board layer for the pad will change to the opposite side (i.e., the PADSTACK layers will be inverted) if the SHAPE is flipped in the COMPONENTS section.

The <rot> parameter must be the pad rotation. The rotation is the angle between the pad position as defined in the PAD section and the pad position in the pad stack, measured counter-clockwise. Any pad mirroring must be done before the pad is rotated.

The <mirror> parameter must be the pad mirror definition. The pad can be placed normally or can be mirrored in either the x or y plane before becoming part of the pad stack. (The whole pad stack can be mirrored when it is associated with a SHAPE description).

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the PADSTACKS section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A PADSTACKS Section Example

```
$PADSTACKS
PADSTACK p_stack1 -1
PAD p102_4 TOP 180 0
PAD s102_4 BOTTOM 0 0
PADSTACK p_stack2 -1
PAD r_r3 TOP 180 MIRRORX
PAD r_r0 INNER1 180 MIRRORX
PAD r_r0 INNER2 180 MIRRORX
PAD r_r3 BOTTOM 180 MIRRORX
$ENDPADSTACKS
```

The ARTWORKS Section

The ARTWORKS section in the GenCAD file is used to describe a library of artwork features: text, lines, arcs, circles, rectangles, or filled areas. One artwork description can be used for many shapes and components. The ARTWORKS section must be included, and can have the following keywords.

```
$ARTWORKS
ARTWORK <artwork_name>
LAYER <layer_name>
TRACK <track_name>
FILLED <filled_ref>
TEXT <x_y_ref> <text_par>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>
$ENDARTWORKS
```

\$ARTWORKS and \$ENDARTWORKS mark the ARTWORKS section of the GenCAD file. Each artwork description must start with the ARTWORK keyword.

ARTWORK <artwork_name>

The ARTWORK keyword defines a name for the set of artwork features. Each <artwork_name> defined in the ARTWORK section must be unique. If the CAD system does not define artwork names, or just uses a sequenced order, then names such as artwork1, artwork2, etc., must be used.

LAYER <layer_name>

The LAYER keyword must be included to indicate the side of the board to which the artwork features are attached. This layer parameter does not imply any changes to the artwork. For example, if the layer is BOTTOM it does not imply that the artwork is mirrored, nor does it imply that artwork features that were on the top are now on the bottom.

FILLED <filled_ref>

The FILLED keyword defines that the following features form an enclosed artwork area. It can appear at any time within an artwork description where a filled or nonfilled area is to be defined. Acceptable values are YES or 0.

TRACK <track_name>

The TRACK keyword identifies the track type as defined in the TRACKS section. It can appear at any time in an artwork description where a track changes width within an artwork feature.

LINE <line_ref>

The LINE keyword defines a straight line that makes up the artwork feature. The coordinates given must always be relative to the origin of the artwork feature. (The origin of the artwork feature is **always** placed on the board at the <x_y_ref> coordinates of the referencing item.)

ARC <arc_ref>

The ARC keyword defines circular (or elliptical) arcs, including semicircles. The coordinates given must be relative to the origin of the artwork feature.

CIRCLE <circle_ref>

The CIRCLE keyword defines a circular artwork. The coordinates given must be relative to the origin of the artwork.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines a rectangular artwork feature. The coordinates given must be relative to the origin of the artwork feature.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to define any text string, size, and location attached to the component. Typically this will be the <component_name>.

The <x_y_ref> parameter must be the x,y coordinate of the bottom left hand corner of the text string relative to the origin of the component origin (i.e., the <x_y_ref> of the PLACE keyword). The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information relevant to the ARTWORKS section. Examples of artwork attributes are pin1, marker, and bar_code. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

An ARTWORKS Section Example

Here is an example of the ARTWORKS section for a pin 1 marker on the silkscreen layer of the board:

```
$ARTWORKS
ARTWORK PIN1_MARKER
LAYER SILKSCREEN_TOP
FILLED 0
TRACK 10
LINE 0 -100 0 100
LINE -100 0 100 0
CIRCLE 0 0 100
$ENDARTWORKS
```

The SHAPES Section

The SHAPES section of the GenCAD file is used to describe a library of component outlines. The true physical dimensions of the component are what is required here, but any description will do. One shape can be used for many components. The SHAPES section must be included and can have the following keywords.

```
$SHAPES
SHAPE <shape_name>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
FIDUCIAL <x_y_ref>
INSERT <string>
HEIGHT <height>
ATTRIBUTE <attrib_ref>

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror>
ATTRIBUTE <attrib_ref>

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror>
ATTRIBUTE <attrib_ref>

PIN <pin_name> <pad_name> <x_y_ref> <layer> <rot> <mirror>
ATTRIBUTE <attrib_ref>
$ENDSHAPES
```

\$SHAPES and **\$ENDSHAPES** mark the SHAPES section of the GenCAD file. Each shape description must start with the **SHAPE** keyword. The **PIN** keyword(s) must be added after the shape description.

The shape described does not have to be enclosed.

If the CAD system does not specifically yield the shapes of devices, a shape definition can be produced for every device in the GenCAD file using the *x* and *y* coordinates of the device pins.

Any text-positioning information given with the CAD shape description belongs only in the **COMPONENTS** section. Therefore, if shape-related text is required, it must be copied into the **COMPONENTS** section for every occurrence of the shape.

SHAPE <shape_name>

The **SHAPE** keyword defines a name for the component outline on the board. Each **<shape_name>** defined in the **SHAPE** section must be unique. If the CAD system does not define shape names, or just uses a sequenced order, then names such as shape1, shape2, etc., must be used.

LINE <line_ref>

The **LINE** keyword defines a straight line that makes up the device shape. The coordinates given must always be relative to the origin of the shape. (The origin of the shape is **always** placed on the board at the device **<x_y_ref>** coordinates).

ARC <arc_ref>

The ARC keyword defines circular (or elliptical) arcs, including semicircles. The coordinates given must be relative to the origin of the shape. See Appendix B for further information about ARC definitions.

CIRCLE <circle_ref>

The CIRCLE keyword defines a circular shape. The coordinates given must be relative to the origin of the shape.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines a rectangular shape. The coordinates given must be relative to the origin of the shape.

FIDUCIAL <x_y_ref>

The optional FIDUCIAL keyword is used to hold the *x,y* coordinates of a fiducial point attached to a shape. The <x_y_ref> parameter must be the *x,y* coordinates of the center of the fiducial point relative to the origin of the shape. Fiducials that are not attached to a shape may be defined in the MECH section using the keyword FHOLE. This feature is not descriptive enough for today's CAD/CAM needs. We suggest using the FID feature for all new development. All parsers should be able to read and write this feature for backwards compatibility.

INSERT <string>

The INSERT keyword is an optional free field used to specify the package style for component insertion. They may be a general package style or specific packages. The true physical dimensions of the package could be described by the ATTRIBUTE keywords. Here are a few examples of the INSERT keyword parameters.

TH	Through hole, other than those defined below.
AXIAL	Used for axial packages.
RADIAL	Used for radial packages.
DIP	Used for dual in-line packages.
SIP	Used for single in-line packages.
ZIP	Used for zigzag packages.
CONN	Used for through hole connectors.
SMD	Used for surface mounted packages.
OTHER	Used for technologies other than TH or SMT.

HEIGHT <height>

The HEIGHT keyword is used to identify the component's maximum height from the board's finished surface. The height is given as type <number> in <dimension> units.

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror>

The ARTWORK keyword used to give the component any artwork feature defined in the ARTWORKS section of the GenCAD file. Components may use different artwork features at different locations on the board.

The <artwork_name> parameter must be the artwork name as defined in the ARTWORKS section in the GenCAD file.

The <x_y_ref> parameter must be the relative position of the artwork item origin with respect to the origin of the shape.

The <rot> parameter must be the rotation of the artwork feature around the artwork origin. The rotation is the angle between the artwork feature as defined in the ARTWORK section and the position of the shape, measured counterclockwise. Any mirroring must be done before the ARTWORK feature is rotated.

The <mirror> parameter must be the artwork feature mirror definition. The artwork feature can be placed normally or mirrored in either the *x* or *y* plane before becoming part of the SHAPE definition. When an artwork feature is mirrored, all the artwork feature items are mirrored, but remain on the same layer. (The whole shape can be mirrored and flipped when it is associated with a COMPONENT description.)

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror>

The FID keyword defines fiducials, using the pad as described in the PADS section or the PADSTACKS section, for each of the shapes fiducials.

The <fid_name> parameter must be the device FID name. The name is only for reference but must be unique within the shape definition.

The <pad_name> parameter must be the pad name as defined in the PADS section or the pad stack name defined in the PADSTACKS section of the GenCAD file.

The <x_y_ref> parameter must be the relative position of the center of the fiducial with respect to the origin of the shape.

The <layer> parameter is the layer on which the pad or pad stack is placed relative to the layer for which the shape is defined. The actual layer on which the pad is situated will depend upon the <flip> parameter of the COMPONENT SHAPE keyword. For example, consider a shape that has been described for a surface mount component on the TOP side of the board with a pad placed on the shape's TOP. If a component using this shape is then placed on the BOTTOM of the board, the pad (and shape) will still be situated on the TOP of the board. The pad can only have its layer changed explicitly by the <flip> parameter in the COMPONENT SHAPE keyword. The layer parameter may also be a layer set if the same pad type is used on all layers within that layer set.

For pad stacks the layer parameter TOP means place the pad stack so that the pads lie on the layers defined in the pad stack. If the layer parameter is BOTTOM then all the layers are swapped TOP to BOTTOM so that, for example, a pad defined as TOP in the pad stack will be placed on the BOTTOM of the shape, pad defined as INNER1 will be on INNER4, INNER2 on INNER3, INNER3 on INNER2, INNER4 on INNER1 and BOTTOM on TOP. The whole shape, with the padstacks, can be flipped when it is associated with a COMPONENT description. To avoid confusion when using named inner layers it is best to define the pad stacks and shapes from the TOP.

The <rot> parameter must be the pad or pad stack rotation. The rotation is the angle between the pad or pad stack position as defined in the PAD or PADSTACK section and the pad or pad stack position on the shape, measured counterclockwise. Any pad or pad stack mirroring must be done before the pad or pad stack is rotated.

The <mirror> parameter must be the pad or pad stack mirror definition. The pad or pad stack can be placed normally or can be mirrored in either the x or y plane before becoming part of the SHAPE definition. When a pad stack is mirrored all the pads making up the stack are mirrored but stay on the same layer. (The whole shape can be mirrored when it is associated with a COMPONENT description).

PIN <pin_name> <pad_name> <x_y_ref> <layer> <rot> <mirror>

The PIN keyword defines pads, as described in the PADS section or the PADSTACKS section, for each of the device pins.

The <pin_name> parameter must be the device pin name. If the CAD system uses pad numbers for the shape descriptions, the pad numbers must correlate to pin names. Device pin names must always be used in the GenCAD file. A single pin can have several pads defined in the GenCAD file for any layers. Pads on inner layers can be excluded in the GenCAD file. The same pin name can also be used to define just a drill hole without a pad by using a pad the same size as the drill hole.

The <pad_name> parameter must be the pad name as defined in the PADS section or the pad stack name defined in the PADSTACKS section of the GenCAD file.

The <x_y_ref> parameter must be the relative position of the center of the pin with respect to the origin of the shape.

The <layer> parameter is the layer on which the pad or pad stack is placed relative to the layer for which the shape is defined. The actual layer on which the pad is situated will depend upon the <flip> parameter of the COMPONENT SHAPE keyword. For example, consider a shape that has been described for a surface mount component on the TOP side of the board with a pad placed on the shape's TOP. If a component using this shape is then placed on the BOTTOM of the board, the pad (and shape) will still be situated on the TOP of the board. The pad can only have its layer changed explicitly by the <flip> parameter in the COMPONENT SHAPE keyword. The layer parameter may also be a layer set if the same pad type is used on all layers within that layer set.

For pad stacks the layer parameter TOP means place the pad stack so that the pads lie on the layers defined in the pad stack. If the layer parameter is BOTTOM then all the layers are swapped TOP to BOTTOM so that, for example, a pad defined as TOP in the pad stack will be placed on the BOTTOM of the shape, a pad defined as INNER1 will be on INNER4, INNER2 on INNER3, INNER3 on INNER2, INNER4 on INNER1, and BOTTOM on TOP. The whole shape, with the padstacks, can be flipped when it is associated with a COMPONENT description. To avoid confusion when using named inner layers it is best to define the pad stacks and shapes from the TOP.

The <rot> parameter must be the pad or pad stack rotation. The rotation is the angle between the pad or pad stack position as defined in the PAD or PADSTACK section and the pad or pad stack position on the shape, measured counterclockwise. Any pad or pad stack mirroring must be done before the pad or pad stack is rotated.

The <mirror> parameter must be the pad or pad stack mirror definition. The pad or pad stack can be placed normally or can be mirrored in either the x or y plane before becoming part of the SHAPE definition. When a pad stack is mirrored, all the pads making up the stack are mirrored but stay on the same layer. (The whole shape can be mirrored when it is associated with a COMPONENT description.)

Names given to device pins (such as O/P, CE, RD) are not supported in the SHAPE descriptions, but may be included in the DEVICE section.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the SHAPES section. The attribute will apply to the previously defined SHAPE, FID, FIDUCIAL, PIN, or ARTWORK item. Examples of shape attributes are center of gravity coordinates, height, weight, body length, body width, body diameter, wire diameters, and component center coordinates for insertion. An example of a pin attribute is mechanical (i.e., no electrical function). See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A SHAPES Section Example

Here is an example of the SHAPE section for a bullet shaped component with pins on a diagonal, using different pads top and bottom of the board and a pin 1 marker on the layer specified in the artwork feature:

```
$SHAPES
SHAPE CAP_SUPPRESS_TYPE_____24
LINE -1000 200 -1000 -200
LINE -1000 -200 1000 -200
ARC 1000 -200 1000 200 1000 0
LINE 1000 200 -1000 200
PIN 1 p102_4 -100 100 TOP 315 0
PIN 1 s106_6 -100 100 BOTTOM 315 MIRRORX
PIN 2 p102_4 100 -100 TOP 135 0
PIN 2 s106_6 100 -100 BOTTOM 135 MIRRORX
ARTWORK PIN1_MARKER 0 400 0 0
FID PRIMARY OPTICAL1 0 0 TOP 0 0
$ENDSHAPES
```

The COMPONENTS Section

The COMPONENTS section holds all the information pertinent to each component on the board. This information includes component positions on the board and a reference to the device description. The COMPONENTS section must be included and must have the keywords DEVICE, PLACE, LAYER, ROTATION, and SHAPE completed.

```
$COMPONENTS
COMPONENT <component_name>
DEVICE <part_name>
PLACE <x_y_ref>
LAYER <layer>
ROTATION <rot>
SHAPE <shape_name> <mirror> <flip>

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>
ATTRIBUTE <attrib_ref>

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>
ATTRIBUTE <attrib_ref>

TEXT <x_y_ref> <text_par>
SHEET <string>
ATTRIBUTE <attrib_ref>
$ENDCOMPONENTS
```

\$COMPONENTS and \$ENDCOMPONENTS mark the COMPONENTS section of the GenCAD file. Each component description must start with the COMPONENT keyword, followed by the other keywords in any sequence. The keywords DEVICE, PLACE, LAYER, ROTATION, SHAPE and SHEET must only be used once per component. The keyword ARTWORK may be used as many times as necessary to define all the instances of artwork features.

COMPONENT <component_name>

The COMPONENT keyword must be included and is a free field for the user to define a unique name for each component, usually the CAD component reference designator. If the CAD system does not define component names, or just uses a sequenced order, then names such as u1, u2, u3, etc., must be used instead. Examples of component name are U32, C202, R51, and PLA132_a.

DEVICE <part_name>

The DEVICE keyword must be included, and is a free field for the user to cross reference the component designator name to a description of the device in the DEVICES section. The DEVICE <part_name> must be unique for each device (i.e., two resistors of the same size, value, etc., but different tolerance, must be given different <part_names>). The <part_name> can be a manufacturer's part number, a library reference part name, a stock number, or anything else that uniquely defines the device description. If a unique <part_name> cannot be extracted from the CAD data, names such as part1, part2, etc., must be used.

To guarantee the integrity of the references, we recommend that you prefix the <part_name> with the reference designator and an underscore. This provides a one-to-one relationship between the DEVICES and COMPONENTS sections, ensuring conformance to the specification.

PLACE <x_y_ref>

The PLACE keyword must be included to give the absolute *x* and *y* coordinates of the origin of the component on the board. The origin of the component is used in the SHAPE section to describe the relative shape and pin locations.

LAYER <layer>

The LAYER keyword must be included to indicate on which side of the board the component is attached. The layer parameter INNER or ALL should not be used. This layer parameter does NOT imply any change of shape. For example, if the layer is BOTTOM it does NOT imply that the shape is mirrored, NOR does it imply that pads that were on the TOP are now on the BOTTOM.

ROTATION <rot>

The ROTATION keyword must be included to give the rotation of the component on the board relative to what is defined in the SHAPE section. The angle is measured counterclockwise from the defined position in the SHAPE section to the position on the board when viewed looking down onto the TOP of the board. Shape mirroring takes affect before the component is rotated. Any PINs defined with the SHAPE will be rotated with the shape.

SHAPE <shape_name> <mirror> <flip>

The SHAPE keyword must be included to give the component a shape that has been defined in the SHAPES section of the GenCAD file. All components must have a shape definition but identical components can use different shapes at different locations on the board.

Component pins are always attached to the shape definition in the GenCAD file. If a CAD system attaches pins to directly to components or if shape defined pins are offset on a component basis then each component can have its own unique shape definition within the GenCAD file. Each shape pin coordinate would then incorporate the pin offset.

The <mirror> parameter must be the mirror definition. Mirroring the shape is usually done when the CAD system provides shapes for the TOP of the board and assumes automatic mirroring and layer change if the component is placed on the BOTTOM of the board. In this GenCAD specification you must explicitly state the mirror. The mirror parameter does not interact with the component <layer> parameter. For example, take a component that calls up a shape with mirror. If the shape has been defined for the TOP of the board in the SHAPE section, and the component is located on the TOP layer, then the component and its mirrored shape will be on the TOP of the board.

The <flip> parameter must be the flip definition. Flipping the shape is usually done when the CAD system provides shapes for the TOP of the board and assumes automatic layer changes for components placed on the BOTTOM of the board. Flipping will change the layers for shape, pins, pads, and padstacks. For example, when flipped the pads defined in the shape section as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2 etc., Flipping is independent of the component <layer> parameter, so if the CAD system does automatic flipping for components on the bottom then for every component placed on the bottom the <flip> parameter must be set.

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>

The ARTWORK keyword used to give the component any artwork feature that has been defined in the ARTWORKS section of the GenCAD file. Components may use different artworks at different locations on the board.

The <artwork_name> parameter must be the artwork name as defined in the ARTWORKS section in the GenCAD file.

The <x_y_ref> parameter must be the relative position of the artwork item origin with respect to the origin of the component.

The <rot> parameter must be the rotation of the artwork feature around the artwork origin. The rotation is the angle between the artwork feature as defined in the ARTWORK section and the position on the component, measured counterclockwise. Any mirroring must be done before the ARTWORK feature is rotated.

The <mirror> parameter must be the artwork feature mirror definition. The artwork feature can be placed normally or can be mirrored in either the x or y plane before becoming part of the SHAPE definition. When an artwork feature is mirrored all the artwork feature's item are mirrored but stay on the same layer.

The <flip> parameter must be the flip definition. Flipping the artwork feature is usually done when the CAD system provides artwork for the TOP of the board and assumes automatic layer changes for components placed on the BOTTOM of the board. Flipping will change the layers for all the artwork features. For example, when flipping the artwork the features defined as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2, etc. Flipping is independent of the component <layer> parameter, so if the CAD system does automatic flipping for components on the bottom then for every component placed on the bottom the <flip> parameter must be set.

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>

The FID keyword defines fiducials, using the pad as described in the PADS section or the PADSTACKS section, for each of the component fiducials. Fiducials that are exactly the same for all the same components can and should be defined in the SHAPE used by the component.

The <fid_name> parameter must be the component fid name. The name is only for reference but must be unique within the component definition.

The <pad_name> parameter must be the pad name as defined in the PADS section or the pad stack name defined in the PADSTACKS section of the GenCAD file.

The <x_y_ref> parameter must be the relative position of the center of the fiducial with respect to the origin of the component.

The <layer> parameter is the layer on which the pad or pad stack is placed relative to the layer for which the shape is defined. The actual layer on which the pad is situated will depend upon the <flip> parameter of the COMPONENT SHAPE keyword. For example, take a shape that has been described for a surface mount component on the TOP side of the board with a pad placed on the shape's TOP. If a component using this shape is then placed on the BOTTOM of the board, the pad (and shape) will still be situated on the TOP of the board. The pad can only have its layer changed explicitly by the <flip> parameter in the COMPONENT SHAPE keyword. The layer parameter may also be a layer set if the same pad type is used on all layers within that layer set.

For pad stacks the layer parameter TOP means place the pad stack so that the pads lie on the layers defined in the pad stack. If the layer parameter is BOTTOM then all the layers are swapped TOP to BOTTOM so that, for example, a pad defined as TOP in the pad stack will be placed on the BOTTOM of the shape, pad defined as INNER1 will be on INNER4, INNER2 on INNER3, INNER3 on INNER2, INNER4 on INNER1 and BOTTOM on TOP. The whole shape, with the padstacks, can be flipped when it is associated with a COMPONENT description. To avoid confusion when using named inner layers it is best to define the pad stacks and shapes from the TOP.

The <rot> parameter must be the pad or pad stack rotation. The rotation is the angle between the pad or pad stack position as defined in the PAD or PADSTACK section and the pad or pad stack position on the shape, measured counterclockwise. Any pad or pad stack mirroring must be done before the pad or pad stack is rotated.

The <mirror> parameter must be the pad or pad stack mirror definition. The pad or pad stack can be placed normally or can be mirrored in either the x or y plane before becoming part of the SHAPE definition. When a pad stack is mirrored all the pads making up the stack are mirrored but stay on the same layer. (The whole shape can be mirrored when it is associated with a COMPONENT description).

The <flip> parameter must be the flip definition. Flipping the fiducial feature is usually done when the CAD system provides artwork for the TOP of the board and assumes automatic layer changes for components placed on the BOTTOM of the board. Flipping will change the layers for all the PAD PADSTACK features. For example, when flipping the PAD PADSTACK the PAD defined as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2, etc. Flipping is independent of the component <layer> parameter, so if the CAD system does automatic flipping for components on the bottom then for every component placed on the bottom the <flip> parameter must be set.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to define any text string, size and location that is attached to the component. Typically this will be the <component_name>.

The <x_y_ref> parameter must be the x,y coordinate of the bottom left hand corner of the text string relative to the origin of the component origin (i.e., the <x_y_ref> of the PLACE keyword).

The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

SHEET <string>

The optional SHEET keyword can be used to define the schematic sheet number, a zone or anything else that is a location property of the component.

The one and only parameter is a free text field.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the COMPONENTS section. The attribute will apply to the previously defined COMPONENT, SHAPE, FID, or ARTWORK item. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A COMPONENTS Section Example

Here is an example of a COMPONENTS section for a diode named D102 and a logic device called U7.

```
$COMPONENTS
COMPONENT D102
DEVICE 1N4148
PLACE 1200 1800
LAYER TOP

ROTATION 90
SHAPE DO35_a MIRRORX 0
ARTWORK ORIGIN_MARKER 0 0 0 MIRRORX 0
TEXT 50 -50 100 90 0 TOP D102 42 -50 500 200
SHEET 12_B3
COMPONENT U7
DEVICE 74LS04
PLACE 0.003 9.52527
LAYER BOTTOM
ROTATION 12.25
SHAPE DIL14 0 FLIP
ARTWORK PIN1_MARKER 6500 2400 0 MIRRORX FLIP
$ENDCOMPONENTS
```

The DEVICES Section

The DEVICES section holds all the physical device descriptions for all the components used on the board. These keywords apply to the physical device itself and do not depend upon any characteristic of the printed circuit board (i.e., are independent of SHAPE, PADS, <mirror>, etc.). The DEVICES section must be included and can use some or all of the keywords listed below, in any sequence. Each keyword can only be used once for each device.

```
$DEVICES
DEVICE <part_name>
PART <part_name>
TYPE <string>
STYLE <string>
PACKAGE <string>
PINDESC <pin_name> <string>
PINFUNCT <pin_name> <string>
PINCOUNT <p_integer>
VALUE <string>
TOL <string>
NTOL <string>
PTOL <string>
VOLTS <string>

DESC <string>
ATTRIBUTE <attrib_ref>
$ENDDEVICES
```

\$DEVICES and \$ENDDEVICES mark the DEVICES section of the GenCAD file. Each device description must start with the DEVICE keyword.

DEVICE <part_name>

The DEVICE keyword must be included at the beginning of each device description. The <part_name> is used for cross referencing the device description to the component name as defined by the DEVICE keyword in the COMPONENTS section. There can be one common DEVICE description for several components of the same value, etc. There can also be a DEVICE description for every component (in which case the DEVICE <part_name> can be the <component_name>). There can NOT be more than one DEVICE description for any one component. The DEVICE <part_name>s must be unique.

PART <part_name>

The PART keyword is an optional free field for the user to define a corporate part number or a unique CAD part name. It has no cross reference to any other GenCAD section. This PART part name does NOT have to be unique to any one part. For example all 1% 0.5W resistors might have part name MRS25. The ATTRIBUTE keywords can also be used for additional part names.

TYPE <string>

The TYPE keyword is an optional free field for the user to define the component type. Typical parameters are listed below. These should be used where possible, but any description can be inserted here. If the device types are not explicitly defined by the TYPE keyword, the device type may be extracted from the component designator name (e.g., RES for component name R32). This extraction can be done by either the translator or the host system. Especially useful are type parameters to distinguish between polarized and non polarized capacitors.

RES	Used for two terminal resistors
VRES	Used for variable resistors
RPACK	Used for resistor packs
CAP	Used for two terminal non polarized capacitors
VCAP	Used for variable capacitors
PCAP	Used for two terminal polarized capacitors
TCAP	Used for Tantalum capacitors
CPACK	Used for Capacitor packs
IND	Used for inductors
VIND	Used for variable inductors
TFR	Used for transformers
DIODE	Used for diodes, including Schottky
DIAC	Used for diacs
ZENER	Used for zener diodes
BRIDGE	Used for silicon bridge rectifiers
PNP,NPN	Used for transistors, unijunctions and darlington
NFET,PFET	Used for Field Effect Transistors and IGBTs
TRIAC	Used for triacs
SCR	Used for Thyristors
FVR	Used for voltage regulators
OPTO	Used for opto-isolators
OPAMP	Used for operational amplifier ics
LOGIC	Used for all logic families
XTAL	Used for crystals
RELAYMB	Used for relays with make before break contacts
RELAYCH	Used for relays with change-over contacts
SWITCH	Used for switches
CONN	Used for connectors
TLAND	Used for Test Land components

STYLE <string>

The STYLE keyword is an optional free field for the user to define an enhancement to the TYPE keyword. Listed below are some examples:

NPN	Used for transistors
PNP	Used for transistors
NFET	Used for N type MOSFETS
PFET	Used for P type MOSFETS
NJFET	Used for N type JFETS
PJFET	Used for P type JFETS
TTL	Used for logic
CMOS	Used for logic
ECL	Used for logic

PACKAGE <string>

The PACKAGE keyword is an optional free field for the user to define the device package used by the CAD system. This keyword should not be used if the CAD system uses different packages for the same device (i.e., uses DEVICE 74HCT04 for 14 pin DIL, flatpak and smd). If different packages are used for the same device, the package information should be added to the COMPONENTS section using the ATTRIBUTE keyword on a component by component basis. Typical package names are DIL_8, TO99, SOT23.

PINDESC <pin_name> <string>

This is an optional keyword that can be included if a device pin has been given a name or a description by the CAD system. It is a free field to hold any CAD data that relates to the device pin. Examples of pin descriptions are O/P, CE, READ and clk. Particularly useful is the naming of the anode and cathode on diodes and zeners; collector, base and emitters on transistors; and gate, source, and drain on fets. If the description is always an accurate description of the pin's functionality (such as ANODE), the PINFUNCT keyword can be used.

The <pin_name> parameter must have the device pin name used in the same context as the \$SHAPES/PIN keyword and \$SIGNALS/NODE keyword. The PINDESC <string> allows the user to add attribute name data to an existing pin.

PINFUNCT <pin_name> <string>

The PINFUNCT keyword is an optional keyword that can be used to describe the functionality of each pin of the device. This keyword should be used to take CAD data that is required by tester output data. Any CAD data that might be relevant to tester outputs should be held in the PINDESC keyword.

The <pin_name> parameter must have the device pin name. Every component defined in the COMPONENT section that uses these DEVICE pin functions must use the same pin names as defined in the component shapes. Only one PIN keyword is allowed for each pin for any one device.

The <string> parameter is a free field text string which must be an attribute of the device pin number. The user can define his own special function parameters but typical ones are listed below.

DRIVER	Used for outputs which drive a net
RECEIVER	Used for inputs
BIDIRECTIONAL	Used for bidirectional logic
POWER	Used for any power supply pin
GROUND	Used for ground pins
NC	Used for a pin that is internally disconnected
ANALOG	Used for analog pins
DIGITAL	Used for any family of logic pins
INACTIVE	Used for resistor, capacitor, etc. pins
ANODE	Used for diodes, zeners, unijunctions, thyristors, etc.
CATHODE	Used for diodes, zeners, unijunctions, thyristors, etc.
BASE	Used for transistors
COLLECTOR	Used for transistors (including IGBTs)
EMITTER	Used for transistors (including IGBTs)
GATE	Used for FETs, unijunctions, IGBTs, thyristors, etc.
SOURCE	Used for FETs
DRAIN	Used for FETs
CASE	Used for connection to device screen or can

PINCOUNT <p_integer>

This is an optional keyword that can be included to define the number of physical pins on the device. The PINCOUNT data should only be used to cross check the shape definition and possibly the signal nodes.

VALUE <string>

The VALUE keyword is an optional free field for the user to give the device a value. The type of value entered must always be the same for all the same type of devices. For example if the value used for resistors is resistance, then all resistors must have values that are resistance. The ATTRIBUTE entries can be used if values can not be explicitly extracted.

TOL <string>

The TOL keyword is an optional free field for the user to specify a \pm tolerance for the device. If the positive and negative tolerance values are different, the keywords NTOL and PTOL must be used instead of TOL. Any string can be entered as a positive floating point percentage value. The ATTRIBUTE entries can be used if the tolerance can not be explicitly extracted.

NTOL <string>

The NTOL keyword is an optional free field for the user to specify a negative (minimum) tolerance for the device. Any string can be entered as a positive floating point percentage value. (A negative sign is implicit in the definition of NTOL and is ignored on reading the file and is never inserted on writing a GenCAD file). The ATTRIBUTE entries can be used if the tolerance can not be explicitly extracted.

PTOL <string>

The PTOL keyword is an optional free field for the user to specify a positive (maximum) tolerance for the device. Any string can be entered as a positive floating point percentage value. The ATTRIBUTE entries can be used if the tolerance can not be explicitly extracted.

VOLTS <string>

The VOLTS keyword is an optional free field for the user to specify a voltage value for the device.

DESC <string>

The DESC keyword is an optional free text field that can be used to describe the device. If device VALUE and TOL are not supplied, the host system will try to extract value and tolerance from this description.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the DEVICES section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A DEVICES Section Example

```
$DEVICES
DEVICE 89-1N4148
PART 1N4148
TYPE DIODE
PINDESC 1 anode
PINDESC 2 cathode
DESC "Diode 1N4148 bandoleer reverse voltage 100V"
$ENDDEVICES
```

The SIGNALS Section

The SIGNALS section defines all the connectivity information of the printed circuit board. The SIGNALS section must be included in the GenCAD file and should have the form:

```
$SIGNALS
SIGNAL <sig_name>
NODE <component_name> <pin_name>
NAILLOC <component_name> <pin_name> <tp_name> <X_Y_ref> <tan> <tin> <probe>
<layer>
ATTRIBUTE <attrib_ref>
$ENDSIGNALS
```

\$SIGNALS and \$ENDSIGNALS mark the SIGNALS section of the GenCAD file. Each connectivity description must start with a SIGNAL keyword.

Vias, pad and test points can only be included in this section if they have been represented as components in the COMPONENTS section, with matching shape in the SHAPES section.

The SIGNALS section on its own will give a pin to pin "rat's nest" wire list.

SIGNAL <sig_name>

The SIGNAL keyword must be included and should contain the signal or net name for one track or route on the board. Signal names may be purely numeric when the cad system uses numbers instead of names. The SIGNAL keyword must be the first keyword for each connection description. The connection description continues on every following line until another SIGNAL keyword or \$ENDSIGNALS appears.

Two signals can NOT be electrically connected on the bare board. This means that each node has to be unique, and that the ROUTES section must not place copper between two signals. However, two signals can be electrically connected through a device such as a wire link.

The <sig_name> parameter is the unique signal or net name as used in the ROUTES section. Nets that have only one node (such as unconnected component pins) may be included if the signal name used is unique.

NODE <component_name> <pin_name>

The NODE keyword defines all connections to components on the net by component and pin.

The <component_name> parameter is the component name as defined by the COMPONENT keyword in the COMPONENT section.

The <pin_name> parameter is the pin name as defined by the PIN keyword in the SHAPES section. Pin descriptions as defined in the DEVICES section must not be used.

The same node (component name and pin name) may occur in different signals. The interpretation of whether the signals are joined is left to the user.

**NAILLOC <component_name> <pin_name> <tp_name> <X_Y_ref> <tan> <tin>
<probe> <layer>**

The NAILLOC (NAIL LOCation) keyword specifies a particular NODE (previously defined in CAD) as the preferred target for a bed-of-nails testpin to physically access the SIGNAL. NAILLOC can be considered to be an attribute of an existing pin that defines the location as the preferred point of electrical access to the SIGNAL. Each SIGNAL may have many NAILLOC keywords. NAILLOC may also be used within the \$ROUTES section to define an attribute of a VIA as nail location. The parameter list is a little different to that defined above when used in this context.

The <component_name> parameter is the component name as defined by the COMPONENT keyword in the COMPONENT section. The component name **must** also be defined in the \$COMPONENTS section.

The <pin_name> parameter is the pin name as defined by the PIN keyword in the SHAPES section. Pin descriptions as defined in the DEVICES section must not be used. The <pin_name> **must** have been already defined in the component \$SHAPE section.

The <tp_name> is the test pin name. If the CAD does not assign a unique name for the nail location (distinct from the node name), then this field should be written as -1.

The <X_Y_ref> is the absolute coordinate of the nail position. Usually this will be the same as the component pin position, but it is permissible to use this coordinate to offset the probe contact point from the component-pin pad origin. However, the X-Y location should still be within the boundary of the pad extents. If the testpin position is coincident with the pad origin, then it is also permitted to write -32767 -32767 in this context to indicate the nail location inherits the position of the NODE.

The <tan> parameter must be the Tester Assigned number. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <tin> parameter must be the Tester Interface name. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <probe> parameter must be the probe type. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used. The probe type is typically its size in thousands followed by a type code. Listed below are some examples:

100T (T = tulip tip)

75C (C = castle tip)

ATE

Burn_In

The <layer> parameter must be the layer on which the probe is to be applied. Only TOP or BOTTOM layers are currently supported.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the SIGNALS section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

Typically the following attributes may be used:

Noprobe	Used to suppress allocation of test probes on a net.
Power	Used to define a net as a power track.
Ground	Used for ground.
Analog	Used for purely analog signals.
Clock	Used for clock lines.
Fast	Used for high speed digital signals.
Noaccess	Used if net is not accessible.
Critical	Used to highlight a net.

Another attribute called Number can be used for using CAD numbering of signals (nets). Conversely, if the <sig_name> used throughout the GenCAD file is the CAD signal (net) number, the ATTRIBUTE Name can be used for the CAD signal (net) name.

Attributes for individual nodes are not held in the GenCAD file.

A SIGNALS Section Example

```
$SIGNALS
SIGNAL data_bus_7
NODE IC3 2
NODE R2 2
NAILLOC R2 2 -1 500 2500 -1 -1 100T BOTTOM
NODE IC4 2
NODE 6Ic2 p34A
NAILLOC 6Ic2 p34A -1 800 3000 -1 -1 75T BOTTOM
SIGNAL ADDRESS_BUS_4
NODE U1 2
NODE PL12 132
NAILLOC PL12 132 -1 200 200 -1 -1 100T BOTTOM
$ENDSIGNALS
```

The TRACKS Section

The TRACKS section of the GenCAD file defines a unique name for each type of track and its width used on the printed circuit board. The GenCAD file only holds track width information. The TRACKS section must be included and can have the following keywords.

```
$TRACKS  
TRACK <track_name> <track_width>  
$ENDTRACKS
```

\$TRACKS and \$ENDTRACKS mark the TRACKS section of the GenCAD file.

TRACK <track_name> <track_width>

The TRACK keyword defines a width name for each track width. If composite tracks are used then the overall width should be used. Tracks of the same width can have different names. The <track_name> parameter must be a unique name. The <track_width> parameter must be the track width in terms of dimensional units. If track widths are not known, the <track_width> should be set to -1 (ASCII decimal 45 and ASCII decimal 49).

A TRACKS Section Example

Here is an example of the TRACKS section:

```
$TRACKS  
TRACK 1 10  
TRACK 2 15  
TRACK 3 17.5  
$ENDTRACKS
```

The LAYERS Section

The optional LAYERS section of the GenCAD file defines a unique name for each CAD layer used in the printed circuit board. The LAYERS section is optional and can have the following keywords.

```
$LAYERS
DEFINE <layer> <string>
LAYERSET <layer>
LAYER <layer>
LAYER <layer>
$ENDLAYERS
```

\$LAYERS and \$ENDLAYERS mark the LAYERS section of the GenCAD file. The <layer> parameter includes a set of predefined names:

TOP	BOTTOM
INNER	INNER _x
ALL	SILKSCREEN_TOP
SILKSCREEN_BOTTOM	SOLDERMASK_TOP
SOLDERMASK_BOTTOM	SOLDERPASTE_TOP
SOLDERPASTE_BOTTOM	POWER _x
GROUND _x .	

See Appendix A for parameter definitions.

DEFINE <layer> <string>

The DEFINE keyword is used to cross reference the GenCAD layer names to the original CAD layer names.

The <layer> parameter must be the GenCAD layer parameter. If the layers section is included it must always have either a TOP or a BOTTOM reference for single sided boards. It must have both a TOP and a BOTTOM reference for multilayer boards. The TOP and BOTTOM layers must be copper layers accessible to test probes. (The layer parameter ALL is not allowed here).

The all inner copper layers must be defined from the TOP downwards using the INNER_x layer parameters in ascending consecutive numerical order starting with INNER1. The INNER_x layer may also be used for non-copper layers although the LAYER_x parameter is preferred. (The TOP layer is NOT INNER1). The order of the GenCAD DEFINE keywords is not significant.

For non-copper layers such as silk screens, solder masks, conformal coating, etc., use SILKSCREEN_TOP, SILKSCREEN_BOTTOM, SOLDERMASK_TOP, SOLDERMASK_BOTTOM, SOLDERPASTE_TOP, SOLDERPASTE_BOTTOM to define the layer parameter. See Appendix A for further information on layer definitions.

The LAYER_x parameters are generally used for optional layers but may also be used if the CAD extraction tool does not provide absolute layers such as TOP, BOTTOM, etc.

The <string> parameter is a free field for the user to enter the CAD layer name or number.

LAYERSET <layer>

The LAYERSET keyword is used to define sets of layers used for pad stacks and vias. The LAYERSET keyword is followed by the <layer> fixed field for layer sets (i.e., LAYERSET1, LAYERSET2, etc.). A layer set for all copper layers need not be defined as the <layer> fixed field ALL is also the layer set named ALL. The layer set name for all layers may be explicitly defined using the LAYERSET and LAYER keywords.

LAYER <layer>

The LAYER keyword is used to list the layers that make up the layer set. It is followed by the layer fixed field.

A LAYERS Section Example

Here is an example of the LAYERS section:

```
$LAYERS
DEFINE TOP 1:copper
DEFINE SOLDERPASTE_TOP 9:solderpaste_top
DEFINE SILKSCREEN_TOP 2:silk_screen
DEFINE SOLDERMASK_TOP 3:solder_mask_top
DEFINE INNER1 4:copper
DEFINE INNER2 5:copper
DEFINE POWER1 6:power
DEFINE GROUND1 7:gnd
DEFINE SOLDERMASK_BOTTOM 8:solder_mask_bottom
DEFINE SILKSCREEN_BOTTOM 9:silkscreen_bottom
DEFINE SOLDERPASTE_BOTTOM 10:solderpaste_bottom
LAYERSET LAYERSET1
LAYER INNER1
LAYER INNER2
LAYERSET LAYERSET2
LAYER TOP
LAYER INNER1
$ENDLAYERS
```

The ROUTES Section

The ROUTES section is optional and used to describe the track routing and all copper shapes on the board. The ROUTES section may have any of the following keywords.

```
$ROUTES
ROUTE <sig_name>
TRACK <track_name>
LAYER <layer>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

VIA <pad_name> <x_y_ref> <layer> <drill_size> <via_name>
NAILLOC <via_name> <tp_name> <X_Y_ref> <tan> <tin> <probe> <layer>
ATTRIBUTE <attrib_ref>

TESTPAD <pad_name> <x_y_ref> <rot> <mirror> <testpad_name>
ATTRIBUTE <attrib_ref>

PLANE <string>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

TEXT <x_y_ref> <text_par>
$ENDROUTES
```

\$ROUTES and \$ENDROUTES mark the ROUTES section of the GenCAD file. Each route description must start with a ROUTE keyword.

The coordinates given for the track route must always be in the center of the track.

ROUTE <sig_name>

The ROUTE keyword identifies the signal name as defined in the SIGNALS section. Only one ROUTE keyword is allowed per <sig_name>.

TRACK <track_name>

The TRACK keyword identifies the track type as defined in the TRACKS section. The TRACK keyword can appear at any time within a route description where a track changes width within a net or route.

LAYER <layer>

The LAYER keyword defines the layer where the track is running and can appear at any time within a route description where a track changes layer. The layer parameter can not be ALL, INNER, or a layer set.

LINE <line_ref>

The LINE keyword describes two coordinates between which a track runs in a straight line. The layer and width must have been previously defined. After the PLANE keyword the LINE keyword describes the outer edge of the plane.

ARC <arc_ref>

The ARC keyword describes two coordinates between which a track runs in a circular curve centered on a third coordinate. The layer and width must have been previously defined. After the PLANE keyword the ARC keyword describes the outer edge of the plane.

CIRCLE <circle_ref>

The CIRCLE keyword describes a circular track such as a guard track around a pin. The layer and width must be previously defined. After the PLANE keyword the CIRCLE keyword describes the outer edge of a circular plane.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword describes a rectangular or square track such as a guard track around a pin. The layer and width must be previously defined. After the PLANE keyword the

RECTANGLE keyword describes the outer edge of a rectangular plane.

VIA <pad_name> <x_y_ref> <layer> <drill_size> <via_name>

The optional VIA keyword defines the location of vias on the board. The via type (plated, solid or hole) is not held in the GenCAD file. The via pad rotation and mirror are also not held in the GenCAD file as the pads are usually round.

The via is associated with the signal name most recently defined in the ROUTES section with the ROUTE keyword.

The <pad_name> parameter must be the pad name for the pad or pad stack used for the via. If a pad or pad stack has not been defined for the via then names such as viapad1, viapad2, etc. must be used of each type of pad.

The <x_y_ref> parameter must be the absolute coordinates of the center of the via.

The <layer> parameter must be the layer. The layer can be TOP or BOTTOM if the via only appears on the top or bottom of the board (blind via). The layer parameter can be ALL if the via goes right through the board, or it can be INNER if the via does not appear on the top or the bottom of the board (buried via). A layer set as defined by the LAYERSET keyword in the LAYERS section may also be used here. If a pad stack is used for the via then this layer parameter is ignored.

The <drill_size> parameter must be the via hole size. This hole size will override any size defined by the pad or padstack. A hole size of 0 (ASCII decimal 48) means that there is no hole. A hole size of -1 (ASCII decimal 45 and ASCII decimal 49) can be used when the size is unknown, and -2 (ASCII decimal 45 and ASCII decimal 50) can be used when the hole size is that defined by the pad or pad stack.

The <via_name> parameter is a unique name or number given to each via. (On multilayer boards two vias can have the same x,y coordinates but must have different names). If the vias are not named by the CAD system then names such as via1, via2, etc., must be used. Each via can have only one VIA keyword definition.

NAILLOC <via_name> <tp_name> <X_Y_ref> <tan> <tin> <probe> <layer>

The NAILLOC (NAIL LOCation) keyword specifies the named Via (previously defined in CAD) as the preferred target for a bed-of-nails testpin to physically access the route/signal. NAILLOC can be considered to be an attribute of the Via pin that defines the location as the preferred point of electrical access to the route/signal. Each ROUTE <sig_name> may have many NAILLOC keywords. NAILLOC may also be used within the \$SIGNALS section to define an attribute of a NODE as the nail location. The parameter list is a little different to that defined above when used in this context.

The <via_name> parameter is the unique name or number given to the via location when the pad geometry/position is defined using the VIA keyword. The <via_name> **must** cross-reference a VIA position already defined by the VIA keyword on the current ROUTE <sig_name>.

The <tp_name> is the test pin name. If the CAD does not assign a unique name for the nail location (distinct from the via name), then this field should be written as -1

The <X_Y_ref> is the absolute coordinate of the nail position. Usually this will be the same as the component pin position, but it is permissible to use this co-ordinate to offset the probe contact point from the via pad origin. However, the X-Y location should still be within the boundary of the pad extents. If the testpin position is coincident with the via origin, then it is also permitted to write -32767 -32767 in this context to indicate the nail location inherits the position of the VIA.

The <tan> parameter must be the Tester Assigned number. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <tin> parameter must be the Tester Interface name. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <probe> parameter must be the probe type. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used. The probe type is typically its size in

thousands followed by a type code. Listed below are some examples:

100T (T = tulip tip)

75C (C = castle tip)

ATE

Burn_In

The <layer> parameter must be the layer on which the probe is to be applied. Only TOP or BOTTOM layers are currently supported.

TESTPAD <pad_name> <x_y_ref> <rot> <mirror> <testpad_name>

The TESTPAD keyword defines the location of a pad, its type and orientation. Such pads are usually test lands or edge connector pads. If pads are not accessible to test probes, they should not be included in this section. (To be accessible the pads must be on the TOP or BOTTOM and should NOT be covered by the solder mask). The layer must have been previously defined in the route description.

The <pad_name> parameter must be the pad name as defined in the PADS section.

The <x_y_ref> parameter must be the absolute coordinates of the pad.

The <rot> parameter must be the rotation of the pad measured counterclockwise from the pad definition.

The <mirror> parameter must be the mirror definition.

The <testpad_name> parameter must be a unique name for the testpad. If the CAD system does not name the testpads then names such as test1, test2, etc., must be used.

PLANE <string>

The PLANE keyword defines an enclosed area of copper using the LINE, ARC, CIRCLE and RECTANGLE keywords. A ROUTE keyword, TRACK keyword and LAYER keyword must appear before a PLANE keyword to define a signal name, a component pin pad to plane clearance and a layer for the plane. If component pins and pads are found within the area defined (on the same layer), the copper area is assumed to be clear of the pins by a non copper track of type <track_name>. Component pins are only connected to the copper plane if the SIGNALS section defines the component pins as being on the same signal as the plane.

The <string> parameter after the keyword is a free text field used to give the plane a unique name. If the CAD system does not provide unique names then names such as plane1, plane2, etc., must be used.

If a second plane is described with x,y coordinates inside another plane on the same layer, then it is assumed that this inner shape is not copper.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to specify characters written on the layer. The <x_y_ref> parameter is the x,y coordinates for the origin of the text. The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the ROUTE, VIA, TESTPAD or PLANE keywords. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A ROUTES Section Example

```
$ROUTES
ROUTE CPU_PAGE_12_U23_3
TRACK 20
LAYER TOP
LINE 1000 2000 1200 2000
VIA DEFAULT_VIA 1200 2000 ALL -1 VIA15
NAILLOC VIA15 -1 1200 2000 -1 -1 100T BOTTOM
LAYER INNER
LINE 1200 2000 1200 2200
ROUTE D24
TRACK 10
LAYER BOTTOM
LINE 1300 1400 1200 2200
TESTPAD p106_2 1200 2200 0 MIRRORX TL102
ROUTE GND
TRACK 20
LAYER BOTTOM
PLANE shield
RECTANGLE 1000 2000 50 50
$ENDROUTES
```

The MECH Section

The MECH section is optional and can be used to describe tooling fixture holes and mechanical components attached to the board. The MECH section can have any of the following keywords.

```
$MECH
HOLE <x_y_ref> <drill_size>

FHOLE <x_y_ref> <drill_size>

MECHANICAL <part_name> [<string>]
PLACE <x_y_ref>
LAYER <layer>
ROTATION <rot>
SHAPE <shape_name> <mirror> <flip>

ATTRIBUTE <attrib_ref>
ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>
ATTRIBUTE <attrib_ref>

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>
ATTRIBUTE <attrib_ref>

$ENDMECH
```

HOLE <x_y_ref> <drill_size>

The optional HOLE keyword defines holes that have not been defined elsewhere in the GenCAD file. Holes such as connector fixings, heat sink fixings and panel mountings can be defined here. Holes that are used for fixture location should be defined using the FHOLE keyword. The first parameter must be the absolute coordinates of the center of the hole. The second parameter must be the hole diameter in <dimension> units. The holes defined in this section go completely through the board.

FHOLE <x_y_ref> <drill_size>

The optional FHOLE keyword defines holes in the board that are used for fixture location. The <x_y_ref> parameter must be the absolute coordinates of the center of the hole. The <drill_size> parameter must be the hole diameter in <dimension> units. The parameter -1 (ASCII decimal 45 and ASCII decimal 49) can be used if the hole diameter is not specified. The parameter may also be 0 (ASCII decimal 48) if the location is just for reference and there is no hole (e.g., a fiducial point). Non-circular reference/fixture holes should be described as fiducials in the MECH section or as cutouts in the BOARD section.

MECHANICAL <part_name> [<string>]

The optional MECHANICAL keyword defines mechanical components that have not been defined elsewhere in the GenCAD file (i.e., items that are not in the COMPONENTS section and are not pads, vias or test pins). The most important of these mechanical components are IC sockets that could be automatically placed on the board by insertion machines.

The <part_name> parameter must be the device name. This free text field can contain a manufacturer's part number, or a library part number, or a stock number, or anything else which uniquely defines the device.

The <string> parameter can be included to indicate what electrical component the mechanical device is attached to. (For an IC holder this could be the IC <component_name>).

The mechanical device can be described in the COMPONENTS and SHAPE sections but must not be included in the SIGNALS section (net list). The MECHANICAL keyword must be followed by PLACE and LAYER. The shape may be referenced by the ROTATION and SHAPE keywords.

FIDUCIAL <fid_name>

The optional FIDUCIAL keyword must be a unique name given to the board fiducial point. This feature is not descriptive enough for today's CAD/CAM needs. We suggest using the FID feature for all new development. All parsers should be able to read and write this feature for backwards compatibility.

PLACE <x_y_ref>

The PLACE keyword must be included with the FIDUCIAL or MECHANICAL keywords to give the absolute x and y coordinates of the origin of the item on the board. The origin of the item is used in the SHAPE section to describe the relative shape.

LAYER <layer>

The LAYER keyword must be included with the FIDUCIAL or MECHANICAL keywords to specify which side of the board the item is on. The layer parameter INNER or ALL should not be used. This layer parameter does NOT imply any change of shape. For example, if the layer is BOTTOM it does NOT imply that the shape is mirrored.

ROTATION <rot>

The ROTATION keyword may be included with the MECHANICAL keyword to give the rotation of the item on the board relative to that defined in the SHAPE section of the GenCAD file. The angle is measured counterclockwise from the defined position in the SHAPE section to the position on the board when viewed looking down onto the TOP of the board. Shape mirroring takes affect before the item is rotated.

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>

The FID keyword defines the fiducial marking, using the pad definition as described in the PADS section or the PADSTACKS section, for each of the fiducials.

The <fid_name> parameter must be the fiducial fid name. The name is only for reference but must be unique within the MECH section.

The <pad_name> parameter must be the pad name as defined in the PADS section or the pad stack name defined in the PADSTACKS section of the GenCAD file.

The <x_y_ref> parameter must be the relative position of the center of the fiducial with respect to the mechanical component origin.

The <layer> parameter is the layer on which the pad or pad stack is placed. The actual layer on which the pad is situated will depend upon the <flip> parameter.

For pad stacks the layer parameter TOP means place the pad stack so that the pads lie on the layers defined in the pad stack. If the layer parameter is BOTTOM then all the layers are swapped TOP to BOTTOM so that, for example, a pad defined as TOP in the pad stack will be placed on the BOTTOM, pad defined as INNER1 will be on INNER4, INNER2 on INNER3, INNER3 on INNER2, INNER4 on INNER1 and BOTTOM on TOP. The whole padstacks, can be flipped by setting the flip parameter. To avoid confusion when using named inner layers it is best to define the pad stacks and shapes from the TOP.

The <rot> parameter must be the pad or pad stack rotation. The rotation is the angle between the pad or pad stack position as defined in the PAD or PADSTACK section and the pad or pad stack position, measured counterclockwise. Any pad or pad stack mirroring must be done before the pad or pad stack is rotated.

The <mirror> parameter must be the pad or pad stack mirror definition. The pad or pad stack can be placed normally or can be mirrored in either the x or y plane. When a pad stack is mirrored all the pads making up the stack are mirrored but stay on the same layer.

The <flip> parameter must be the flip definition. Flipping the fiducial is used to flip the pad or pad stack so that all the pad padstack layers re swapped TOP to BOTTOM. Inner layers are not

flipped unless they are explicitly named INNER1, INNER2, etc.

SHAPE <shape_name> <mirror> <flip>

The SHAPE keyword may be included with the MECHANICAL keyword to give the item a shape that has been defined in the SHAPES section of the GenCAD file.

The <mirror> parameter must be the mirror definition. Mirroring the shape is usually done when the CAD system provides shapes for the TOP of the board and assumes automatic mirroring and layer change if the item is placed on the BOTTOM of the board. In this GenCAD specification you must explicitly state the mirror. The mirror parameter DOES NOT interact with the mechanical component <layer> parameter. For example, take an item that calls up a shape with mirror. If the shape has been defined for the TOP of the board in the SHAPE section, and the item is located on the TOP layer, the item and its mirrored shape will be on the TOP of the board.

The <flip> parameter must be the flip definition. Flipping the shape is usually done when the CAD system provides shapes for the TOP of the board and assumes automatic layer changes for mechanical components placed on the BOTTOM of the board. Flipping will change the layers for shape, pins, pads and padstacks. For example, when flipped the pads defined in the shape section as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2, etc. Flipping is independent of the component <layer> parameter, so if the CAD system does automatic flipping for mechanical components on the bottom then for every component placed on the bottom the <flip> parameter must be set.

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>

The ARTWORK keyword used to give the mechanical component any artwork feature that has been defined in the ARTWORKS section of the GenCAD file. Mechanical components may use different artworks at different locations on the board.

The <artwork_name> parameter must be the artwork name as defined in the ARTWORKS section in the GenCAD file.

The <x_y_ref> parameter must be the relative position of the artwork item origin with respect to the origin of the mechanical component.

The <rot> parameter must be the rotation of the artwork feature around the artwork origin. The rotation is the angle between the artwork feature as defined in the ARTWORK section and the position on the mechanical component, measured counterclockwise. Any mirroring must be done before the ARTWORK feature is rotated.

The <mirror> parameter must be the artwork feature mirror definition. The artwork feature can be placed normally or can be mirrored in either the x or y plane before becoming part of the mechanical component definition. When an artwork feature is mirrored all the artwork feature's item are mirrored but stay on the same layer.

The <flip> parameter must be the flip definition. Flipping the artwork feature is usually done when the CAD system provides artwork for the TOP of the board and assumes automatic layer changes for mechanical components placed on the BOTTOM of the board. Flipping will change the layers for all the artwork features. For example, when flipping the artwork the features defined as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2, etc. Flipping is independent of the mechanical component <layer> parameter, so if the CAD system does automatic flipping for mechanical components on the bottom then for every component placed on the bottom the <flip> parameter must be set

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to an object in the MECH section. The attribute will apply to the previously defined MECHANICAL, FID, FIDUCIAL, or ARTWORK item. See [Appendix A beginning on page](#)

[73](#) for the <attrib_ref> parameter definitions.

A MECH Section Example

```
$MECH
HOLE 1000 1000 25
HOLE 1000 3125 -1
FHOLE -4000 -6800 100
FHOLE 4000 6800 100
FID FID1 FID80 4000 1000 TOP 0 0 0
ARTWORK FORMATC -5000 -2000 0 0 0
$ENDMECH
```

The TESTPINS Section

The TESTPINS section is optional and can be used to define the position of test nails within an in-circuit bed-of-nails fixture. The intent is that these nails will contact the printed circuit board (PCB) on an existing target object (pad), such as a pin, via or dedicated test-land. The geometry of the target object (pad) that is the home for the TESTPIN **must** be defined elsewhere in the GenCAD data—either as a \$COMPONENT instance or \$ROUTE/TESTPAD keyword. This section should only be used to describe the targets for test nails that have been defined elsewhere in the GenCAD data. The TESTPINS section may have any of the following keywords:

```
$TESTPINS
TESTPIN <tp_name> <x_y_ref> <sig_name> <tan> <tin> <probe> <layer>
TEXT <x_y_ref> <text_par>
ATTRIBUTE <attrib_ref>
$ENDTESTPINS
```

\$TESTPINS and \$ENDTESTPINS are used to mark the TESTPINS section in the GenCAD file. Up to 65535 test pins can be defined in this section.

TESTPIN <tp_name> <x_y_ref> <sig_name> <tan> <tin> <probe> <layer>

The TESTPIN keyword defines the location of test pins on the board.

The first <tp_name> must be the test pin name. Each TESTPIN keyword must have a unique test pin name. (The test pin name may be the Tester Assigned number or the Tester Interface number).

The <x_y_ref> parameter must be the test pin coordinates on the board. For unallocated test pins the coordinates must be -32767 -32767 to indicate that the test pin has not been placed on the board.

The <sig_name> parameter must be the signal name of the net on which the test pin has been placed. A test pin can not be placed on anything that is not connected to a signal name via copper on the board. For unused test pins the signal name must be -1 (ASCII decimal 45 and ASCII decimal 49) to indicate that the test pin has not been assigned to a signal. (The test pin coordinates should also be -32767 -32767).

The <tan> parameter must be the Tester Assigned number. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <tin> parameter must be the Tester Interface number. If this parameter has not been defined then -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <probe> parameter must be the probe type. If this parameter has not been defined then a -1 (ASCII decimal 45 and ASCII decimal 49) must be used. The probe type is typically its size in thou followed by a type code. Listed below are some examples:

100T	(T = tulip tip)
75C	(C = castle tip)
ATE	
Burn_In	

The <layer> parameter must be the layer on which the probe is to be applied (TOP or BOTTOM).

TEXT <x_y_ref> <text_par>

The TEXT keyword is used to attach a text string to a previously defined test pin definition in the GenCAD file. The <x_y_ref> parameter is the relative x,y coordinates of the bottom left hand corner of the first character of the text string relative to the test pin. The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

Unallocated testpins must NOT have TEXT keywords.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the TESTPINS section. The attribute will apply to the most recently defined TESTPIN item, and not to the section as a whole. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A TESTPINS Section Example

```
$TESTPINS
TESTPIN 1 100 -100 data_bus_7 F234 3_023 100 BOTTOM
TESTPIN TP_1 345 567 ground -1 1_001 500 BOTTOM
TEXT 10 10 50 0 MIRRORX BOTTOM TP_1 10 10 210 60
$ENDTESTPINS
```

The POWERPINS Section

The POWERPINS section is optional and can be used to define the position of nails within an in-circuit bed-of-nails fixture that are connected to a tester power supply. Nails that are connected to tester driver/sensors should use the \$TESTPINS section. The intent is that power injection nails will contact the printed circuit board (PCB) on an existing target object (pad), such as a pin, via or dedicated test-land. The geometry of the target object (pad) that is the home for the POWERPIN **must** be defined elsewhere in the GenCAD data—either as a \$COMPONENT instance or \$ROUTE/TESTPAD keyword. This section should only be used to describe the position of targets for test nails that have been defined elsewhere in the GenCAD data. The POWERPINS section may have any of the following keywords.

```
$POWERPINS
POWERPIN <tp_name> <x_y_ref> <sig_name> <tan> <tin> <probe> <layer>
TEXT <x_y_ref> <text_par>
ATTRIBUTE <attrib_ref>
$ENDPOWERPINS
```

\$POWERPINS and \$ENDPOWERPINS are used to mark the POWERPINS section in the GenCAD file.

POWERPIN <tp_name> <x_y_ref> <sig_name> <tan> <tin> <probe> <layer>

The POWERPIN keyword defines the location of power injection pins on the board.

The <tp_name> parameter must be the power pin name. Each POWERPIN keyword must have a unique pin name. (The pin name may be the Tester Assigned number or the Tester Interface number).

The <x_y_ref> parameter must be the pin coordinates on the board. For unallocated power pins the coordinates must be -32767 -32767 to indicate that the power pin has not been placed on the board.

The <sig_name> parameter must be the signal name of the net on which the power pin has been placed. A power pin can not be placed on anything that is not connected to a signal name via copper on the board. For unused power pins the signal name must be -1 (ASCII decimal 45 and ASCII decimal 49) to indicate that the power pin has not been assigned to a signal. (The power pin coordinates should also be -32767 -32767).

The <tan> parameter must be the Tester Assigned number. If this parameter has not been defined then a -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <tin> parameter must be the Tester Interface number. If this parameter has not been defined then a -1 (ASCII decimal 45 and ASCII decimal 49) must be used.

The <probe> parameter must be the probe type. If this parameter has not been defined then a -1 (ASCII decimal 45 and ASCII decimal 49) must be used. The probe type is typically its size in thou followed by a type code.

The <layer> parameter must be the layer on which the probe is to be applied (TOP or BOTTOM).

TEXT <x_y_ref> <text_par>

The TEXT keyword is used to attach a text string to a previously defined power pin definition in the GenCAD file. The <x_y_ref> parameter must be the power pin name. The second parameter is the relative x,y coordinates of the bottom left hand corner of the first character of the text string relative to the position of the power pin named. The <text_par> parameter defines the size of text, rotation, mirror, layer, and dimensions and location of the bounding

rectangle.

Unallocated power pin must NOT have TEXT keywords.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the POWERPINS section. See [Appendix A, Parameter Definitions, on page 71](#) for the <attrib_ref> parameter definitions.

A POWERPINS Section Example

```
$POWERPINS
POWERPIN 1 100 -100 data_bus_7 F234 3_023 100 TOP
POWERPIN %G_1 345 567 ground -1 1_001 500 BOTTOM
$ENDPOWERPINS
```

The PSEUDOS Section

This section is now obsolete because all the string limitations have been eliminated.

The PSEUDOS section keeps a record of the original CAD names that were too long to use in the GenCAD file.

```
$PSEUDOS
SIGNAL <sig_name> <string>
DEVICE <part_name> <string>
PART <part_name> <string>
SHAPE <shape_name> <string>
COMPONENT <component_name> <string>
TESTPAD <pad_name> <string>
PADSTACK <pad_name> <string>
PIN <pin_name> <string>
TESTPIN <tp_name> <string>
POWERPIN <tp_name> <string>
VIA <via_name> <string>
TESTPAD <testpad_name> <string>
$ENDPSEUDOS
```

\$PSEUDOS and \$ENDPSEUDOS mark the PSEUDOS section of the GenCAD file. Up to 65535 signal and 65535 device name pseudonyms can be defined in this section.

Each keyword is used to identify the pseudonyms for the original CAD names. The first parameter is the new unique name used throughout the GenCAD file. The second parameter is the original name given by the CAD system.

A PSEUDOS Section Example

```
$PSEUDOS
SIGNAL MOTHERBOARD_PAGE_34001 MOTHERBOARD_PAGE_34_U235_32_$a
DEVICE $74LS245 LIBRARY&PART*LOGIC#74LSXXX.\74FAMILY\74ls245\part.lst
$ENDPSEUDOS
```

The CHANGES Section

The CHANGES section can be used to define a sequential list of changes or additions to the previously defined GenCAD sections. The changes listed must NOT have been implemented on the main sections of the GenCAD file. The CHANGES section can reside in a separate file but must still have a HEADER section. The CHANGES section is optional and can use some or all of the keywords listed below.

```
$CHANGES
CHANGE <string>
SIGNAL <sig_name> <sig_name>
DEVICE <part_name> <part_name>
{any GenCAD type section}
$ENDCHANGES
```

The SIGNAL and DEVICE keywords are used for global changes to signal and part names. They do not correspond to any syntax used in other sections of the GenCAD file, so they must be defined outside any \$section/\$END section (see [“A CHANGES Section Example” on page 57](#)).

\$CHANGES and \$ENDCHANGES mark the CHANGES section of the GenCAD file. Each change description must start with the CHANGE keyword.

NOTE: Use of the CHANGES section by any new application writing GenCAD files is **not** recommended.

CHANGE <string>

The CHANGE keyword defines the start of change information. The CHANGE parameter can be used to identify the change.

SIGNAL <sig_name> <sig_name>

The SIGNAL keyword allows the GenCAD signal names to be renamed. The <sig_name> parameter must be the existing signal name and the second <sig_name> parameter must be the new signal name. When the changes are incorporated into the main sections of the GenCAD file, all occurrences of the old signal name are changed for the new name. (The sections in which the signal name is used are \$SIGNALS keyword SIGNAL, \$ROUTES keyword ROUTE, \$TESTPINS keyword TESTPIN and \$PSEUDOS keyword SIGNAL). The original CAD signal name held in the PSEUDOS section can **never** be changed.

DEVICE <part_name> <part_name>

The DEVICE keyword allows the GenCAD device names to be renamed. The <part_name> parameter must be the existing device name and the second <part_name> parameter must be the new device name. When the changes are incorporated into the main sections of the GenCAD file, all occurrences of the old device name are changed for the new name. (The sections in which the part name is used are \$COMPONENTS keyword DEVICE, \$DEVICES keyword DEVICE.)

Changes to the HEADER Section

The some of the HEADER keyword parameters can be changed by including a HEADER section in the CHANGES section. The syntax must be the same as that described in the main HEADER section.

GENCAD	Should not be changed unless the GenCAD specification changes are reflected in the file sections.
USER	The new user description replaces the old one.
DRAWING	Should not be changed.
REVISION	The new revision description replaces the old one.
UNITS	The units can be changed but there may be rounding errors on dimensions.
ORIGIN	The new origin coordinate replaces the old one.
INTERTRACK	The parameter can NOT be changed in the CHANGES section. Only the host system can update this parameter.

Changes to the BOARD Section

The BOARD shape keywords and parameters can be changed by redefining the whole BOARD section in the CHANGES section. The board mask area can be added, with the same syntax as described in the main BOARD section, by using the MASK keyword in a BOARD section within a CHANGE section.

Changes to the PADS Section

New pads can be defined using a unique <pad_name> and keyword descriptions in a PADS section within the CHANGES section. Existing pad parameters can only be changed by redefining the whole pad description using the same <pad_name>. Pad definitions can not be deleted from the change section.

Changes to the PADSTACKS Section

New padstacks can be defined using a unique <pad_name> and keyword descriptions in a PADSTACKS section within the CHANGES section. Existing pad parameters can only be changed by redefining the whole pad stack description using the same <pad_name>. Pad stack definitions can not be deleted from the change section.

Changes to the ARTWORKS Section

New artworks can be defined using a unique <artwork_name> and keyword descriptions in an ARTWORKS section within the CHANGES section. Existing artwork parameters, can only be changed by redefining the whole artwork definition using the same <artwork_name>. Artwork definitions can not be deleted from the change section.

Changes to the SHAPES Section

New shapes can be defined using a unique <shape_name> and keyword descriptions in a SHAPES section within the CHANGES section. Existing shape parameters, including PIN, can only be changed by redefining the whole shape definition using the same <shape_name>. Shape definitions can not be deleted from the change section.

Changes to the COMPONENTS Section

New components can be added using a unique <component_name> and keyword descriptions in a COMPONENTS section within the CHANGES section. Existing component parameters can only be changed by redefining the whole component definition using the same <component_name>. The <part_name> must be globally changed (see [“DEVICE <part_name> <part_name>” on page 52](#)).

Changes to the DEVICES Section

New devices can be added using a unique <part_name> and keyword description in a DEVICES section within the CHANGES section. Existing device keywords can be added or changed, as detailed below, by using an existing device <part_name> in a DEVICES section within a CHANGES section. Keyword parameters can be deleted by not specifying a parameter after the keyword.

DEVICE	The DEVICE keyword parameter can only be changed globally (see “DEVICE <part_name> <part_name>” on page 52).
TYPE	Can be changed or added after the device has been specified.
PART	Can be changed or added after the device has been specified.
STYLE	Can be changed or added after the device has been specified.
PACKAGE	Can be changed or added after the device has been specified.
INSERT	Can be changed or added after the device has been specified.
PINDESC	Can be changed or added after the device has been specified.
PINFUNCT	Can be changed for each pin by redefining all parameters. An additional attribute can only be added by specifying all the existing ones as well. If only one attribute is described then all previously defined attributes will be lost for that pin. (The <pin_name> can NOT be changed). Each PIN number must be uniquely defined.
PINCOUNT	Can be changed or added after the device has been specified.
VALUE	Can be changed or added after the device has been specified.
TOL	Can be changed or added after the device has been specified.
NTOL	Can be changed or added after the device has been specified.
PTOL	Can be changed or added after the device has been specified.

VOLTS	Can be changed or added after the device has been specified.
DESC	Can be changed or added after the device has been specified.
ATTRIBUTE	All can be changed or added after the device has been specified.

Changes to the SIGNALS Section

New signals can be added using a unique <sig_name> and keyword descriptions in a SIGNALS section within the CHANGES section. Existing signals can only have the ATTRIBUTE keyword added or changed. The SIGNAL keyword parameter can only be changed globally (see [“SIGNAL <sig_name> <sig_name>” on page 52](#)).

NODE	Cannot be changed or added without specifying the whole signal description.
ATTRIBUTE	Can be added or changed for a signal after a SIGNAL <sig_name> keyword in a SIGNALS section within a CHANGES section. An attribute can be deleted by not specifying a parameter.

Changes to the TRACKS Section

New tracks can be defined using a unique <track_name> in a TRACKS section within the CHANGES section. Existing track parameters can only be changed by redefining the whole track type using the same <track_name>. Track definitions can not be deleted from the change section.

Changes to the LAYERS Section

Addition or removal of layers in NOT allowed.

DEFINE	Can be used to redefine the GenCAD to CAD layer relationship within a LAYERS section of the CHANGE section.
LAYERSET	Can not be redefined.
LAYER	Can not be used.

Changes to the ROUTES Section

The ROUTE keyword parameter can only be changed globally (see [“SIGNAL <sig_name> <sig_name>” on page 52](#)). Each complete signal track route can be changed by using an existing <sig_name> followed by a route description. Similarly a new route can be added using a new <sig_name> provided that a new SIGNALS description is added as well. The route description (LINE, LAYER, TRACK, etc.) can not be changed alone.

Changes to the MECH Section

The holes and fixture holes in the board can be added to by defining new coordinates and drill size in a MECH section within a CHANGES section. Existing holes can be deleted by using the existing coordinates and a new drill size of 0.

Mechanical components can be added using the DEVICE keyword. Mechanical components can be deleted by using the default coordinates of -32767 -32767 for an existing <part_name>.

Changes to the TESTPINS Section

New test pins can not be created. CBTest, or other programs, will provide a list of unused and unallocated test pins with unique test pin names. The coordinates, signal name and probe type can be changed for any defined test pin by using the TESTPIN keyword in a TESTPINS section within a CHANGES section.

Text associated with an existing test pin can be changed following a TESTPIN definition in the CHANGE section.

Changes to the POWERPINS Section

New power injection pins can not be created. CBTest or other programs, will provide a list of unused and unallocated power pins with unique power pin names. The coordinates, signal name and probe type can be changed for any defined power pin by using the POWERPIN keyword in a POWERPINS section within a CHANGES section.

Text associated with an existing power injection pin can be changed following a POWERPIN definition in the CHANGE section.

Changes to the PSEUDOS Section

This section is now obsolete because PSEUDOS is obsolete. The SIGNAL keyword parameter <sig_name> can only be changed globally (see [“SIGNAL <sig_name> <sig_name>” on page 52](#)). The DEVICE keyword parameter <part_name> can only be changed globally (see [“DEVICE <part_name> <part_name>” on page 52](#)). The original CAD signal and device names can NEVER be changed.

Changes to the CHANGES Section

The change sections themselves can NEVER be changed. CBTest, or other programs, will incorporate the changes in CHANGES section sequentially. This means that the last CHANGES section should contain change information as if all the previous CHANGES sections have been incorporated into the main sections of the GenCAD file.

A CHANGES Section Example

```
$CHANGES
CHANGE ADJUST_SIGNALS
SIGNAL pullup_23 POW_23          <- global signal name change
DEVICE RES_234_bin16a RES_234_bin16b <- global device name change
$HEADER
REVISION REV B 20th Feb 1992

$ENDHEADER
$DEVICES
DEVICE RES_234_bin16b
TYPE RES                        <- adds or changes TYPE
ATTRIBUTE 1 24 T4              <- adds or changes ATTRIBUTE 1 24
$ENDDEVICES
CHANGE Add_attributes          <- 2nd change information
$DEVICES
DEVICE RES_543
VALUE 2k2                      <- adds or changes VALUE
PINFUNCT 1 BIDIRECTIONAL      <- adds or changes PIN 1 attribute
$ENDDEVICES
$SIGNALS
SIGNAL Data_bus_4              <- add attribute to existing signal
ATTRIBUTE NOPROBE              <- add or changes existing attribute
SIGNAL AD1

ATTRIBUTE                      <- deletes the attribute for signal AD1
SIGNAL CE2                     <- deletes signal CE2
SIGNAL ADDITIONAL_TRACK        <- adds signal
NODE IC1 1
NODE IC2 2
$ENDSIGNALS
$ENDCHANGES
```


Chapter 3

The Panel File

Panel File Contents

The panel file can have any filename that the user chooses provided that it is acceptable to the operating system(s) that is being used.

The panel file will consist of the following sections in any order. Each section must only appear once in the file.

HEADER	General information about the panel.
PANEL	Panel dimensions.
ARTWORKS	Artwork feature definitions.
MECH	Mechanical description and fixing hole information.
BOARDS	Identification and position of the board(s).
CHANGES	Change section.

The following items are not explicitly handled by the panel file:

- Board (and component) heights
- Off-grid data
- Comment fields
- Panels that use a combination of dimensional units.

The HEADER Section

The HEADER section must be included and contain the GENPANEL keyword. All of the keywords shown below may appear. Each keyword must be followed by one parameter.

```
$HEADER
GENPANEL <number>
USER <string>
DRAWING <string>
REVISION <string>
UNITS <dimension>
ORIGIN <x_y_ref>
INTERTRACK <number>
ATTRIBUTE <attrib_ref>
$ENDHEADER
```

The \$HEADER and \$ENDHEADER mark the HEADER section of the panel file.

GENPANEL <number>

The GENPANEL keyword defines the file as a panel file. The GENPANEL keyword must be followed by a fixed text field used to define the version of the GenCAD Specification that was used to build the panel. The version number of this specification can be found on the cover page of this manual.

USER <string>

When the panel file is created by a translation program, the USER keyword usually contains the company name and serial number to whom the software is licensed to. If the panel file is produced by any other method, the USER free field can contain any text identifying the originator of the panel file.

DRAWING <string>

The DRAWING keyword is followed by a free text field for the user to define a number or title for the panel.

REVISION <string>

The REVISION keyword is a free text field for the user to define a revision, issue or a modification status of the panel.

UNITS <dimension>

The UNITS keyword is a fixed format field used to define the size of the dimensional units used throughout the panel file.

ORIGIN <x_y_ref>

The ORIGIN keyword defines the coordinates of the origin of the panel. The origin is initially defined at coordinates 0, 0. If all coordinates are later offset, the ORIGIN parameter is changed to reflect this offset from the original origin. For example take a square panel which has bottom left hand corner coordinates of -100, -100 and a board placed with its origin at -50, -50. If all coordinates are offset so that they are all positive, the CAD origin moves to +100, +100. In this case the ORIGIN keyword will have relative x,y coordinates of +100, +100. The board will now be at +50, +50.

INTERTRACK <number>

The INTERTRACK keyword is used by programs to record implemented change information. The parameter must be set to zero when the panel file is first produced. This parameter will be updated every time changes are incorporated into the main sections (but not when a CHANGE section is added to the \$CHANGES section).

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the HEADER section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A HEADER Section Example

Here is an example of a HEADER section:

```
$HEADER
GENPANEL 1.4
USER "Mitron Europe Ltd. Serial Number 000"
DRAWING "Panel 6 up - Modem C100 motherboard 1234-5678"
REVISION "Rev 566g 20th September 1990"
UNITS USER 1200
ORIGIN 0 0
INTERTRACK 0
$ENDHEADER
```

The PANEL Section

The PANEL section defines the outer shape and any internal cutouts of the panel in terms of absolute x and y coordinates as viewed from the top. The panel outline shape, the cutout shapes, and the mask off shapes described must be fully enclosed. The PANEL section must be

included and can use some or all of the following keywords.

```
$PANEL
THICKNESS <number>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

CUTOUT <string>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

MASK <string>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>

ARTWORK <string> <layer>
FILLED <filled_ref>
TRACK <track_ref>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
TEXT <x_y_ref> <text_ref>
ATTRIBUTE <attrib_ref>

$ENDPANEL
```

The \$PANEL and \$ENDPANEL mark the PANEL section of the panel file. Each cutout description must start with the CUTOUT keyword and each mask off area description must start with a MASK keyword and each ARTWORK feature description must start with an artwork keyword.

The panel outline, cutout and mask shape can be given in a clockwise or counterclockwise direction. (Arcs must be defined in an counterclockwise direction.)

THICKNESS <number>

The optional THICKNESS keyword defines the thickness of the panel in <dimension> units as specified in the HEADER section.

LINE <line_ref>

The LINE keyword defines a straight line that forms part of the outer edge of the panel or the internal edge of a cutout or the outer edge of a masked off area.

ARC <arc_ref>

The ARC keyword defines an arc that forms part of the outer edge of the panel or the internal edge of a cutout or the outer edge of a masked off area.

CIRCLE <circle_ref>

The CIRCLE keyword defines the outer edge of a circular panel or the internal edge of a cutout or the outer edge of a masked off area.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines the outer edge of a rectangular or square panel or the internal edge of a cutout or the outer edge of a masked off area.

CUTOUT <string>

The CUTOUT keyword is used to name and define an internal area of the panel which has had parts cut away. If the CAD system does not provide a unique name for each cutout, the panel file must use cutout1, cutout2 etc. The LINE, ARC and CIRCLE keywords following a CUTOUT keyword describe the cutout shape. Circular cutouts can alternatively be defined in the MECH section as drilled holes.

MASK <string>

The MASK keyword is used to name and define areas of the panel outside which is not accessible to test pins or insertion machines. If a name is not provided for the masked off area, the panel file must use mask1, mask2 etc. The LINE, ARC and CIRCLE keywords following a MASK keyword describe the masked off shape inside which components can be placed. Several masked areas can be defined for different fixtures, even if they overlap, as long as they have different names.

An inner area can be masked off by creating a second mask description with the same name. This second mask description must be wholly inside the first. This inner most area is NOT accessible, effectively toggling the accessibility. A third masked off area within two others would be accessible to insertion machines, probes etc.

ARTWORK <string> <layer>

The ARTWORK keyword is used to name and define an artwork feature of the panel. If a unique name is not provided for the artwork feature, the GenCAD file must use names such as artwork1, artwork2 etc. The FILLED, TRACK, LINE, ARC, CIRCLE, RECTANGLE and TEXT keywords following an ARTWORK keyword describe the artwork feature. Several artwork areas can be defined for different features, even if they overlap, as long as they have different names. The FILLED keyword is used to identify that the following LINE, ARC, CIRCLE and RECT form an enclosed area or just line artwork. The use of LAYER or FILLED NO in a FILLED set will also terminate an enclosed area.

The <layer> parameter must be the layer for which the mask applies, either TOP, BOTTOM, ALL, etc.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to define any text string, size, and location attached to the component. Typically this will be the <component_name>.

The <x_y_ref> parameter must be the x,y coordinate of the bottom left hand corner of the text string relative to the origin of the component origin (i.e., the <x_y_ref> of the PLACE keyword). The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the PANEL section. The attribute will apply to the previously defined PANEL, CUTOUT, MASK, or ARTWORK itempanel, cutout or mask. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A PANEL Section Example

Here is an example of the PANEL section:

```
$PANEL
LINE 1000 2000 1200 2000
ARC 1200 2000 1200 3000 1180 2500
LINE 1200 3000 1000 3000
LINE 1000 3000 1000 2000
CUTOUT "Protruding nut"
CIRCLE 1180 2500 20
MASK Fixture_1
LINE 1005 2005 1195 2005
ARC 1195 2005 1195 2995 1195 2500
LINE 1195 2995 1005 2995
LINE 1005 2995 1005 2005
$ENDPANEL
```

The ARTWORKS Section

The ARTWORKS section to the PANEL file is used to describe a library of artwork text, lines, arcs circles, rectangles or filled areas. One artwork can be used for many mechanical components. The ARTWORKS section must be included and can have the following keywords.

```
$ARTWORKS
ARTWORK <artwork_name>
LAYER <layer>
TRACK <track_name>
FILLED <filled_ref>
TEXT <x_y_ref> <text_par>
LINE <line_ref>
ARC <arc_ref>
CIRCLE <circle_ref>
RECTANGLE <rectangle_ref>
ATTRIBUTE <attrib_ref>
$ENDARTWORKS
```

The \$ARTWORKS and \$ENDARTWORKS mark the ARTWORKS section of the PANEL file. Each artwork description must start with the ARTWORK keyword.

ARTWORK <artwork_name>

The ARTWORK keyword defines a name for the artwork feature for reference Each <artwork_name> defined in the ARTWORK section must be unique. If the CAD system does not define artwork names, or just uses a sequenced order, then names such as artwork1, artwork2 etc., must be used.

LAYER <layer>

The LAYER keyword must be included to indicate on which side of the panel the artwork features are attached. This layer parameter does NOT imply any change of the artwork. For example, if the layer is BOTTOM it does NOT imply that the artwork is mirrored, NOR does it imply that artwork features that were on the TOP are now on the BOTTOM.

FILLED <filled_ref>

The FILLED keyword defines that the following features form an enclosed artwork area.

The FILLED keyword can appear at any time within an artwork description where a filled or nonfilled area is to be defined. Acceptable values are YES or NO,

TRACK <track_name>

The TRACK keyword identifies the track type as defined in the TRACKS section. The TRACK keyword can appear at any time within an artwork description where a track changes width within an artwork feature.

LINE <line_ref>

The LINE keyword defines a straight line that makes up the artwork feature. The coordinates given must always be relative to the origin of the artwork. (The origin of the artwork is always placed on the board at the <x_y_ref> coordinates of the referencing item.)

ARC <arc_ref>

The ARC keyword defines circular (or elliptical) arcs, including semicircles. The coordinates given must be relative to the origin of the artwork.

CIRCLE <circle_ref>

The CIRCLE keyword defines a circular artwork. The coordinates given must be relative to the origin of the artwork.

RECTANGLE <rectangle_ref>

The RECTANGLE keyword defines a rectangular artwork. The coordinates given must be relative to the origin of the artwork.

TEXT <x_y_ref> <text_par>

The optional TEXT keyword can be used to define any text string, size, and location attached to the component. Typically this will be the <component_name>.

The <x_y_ref> parameter must be the x,y coordinate of the bottom left hand corner of the text string relative to the origin of the component origin (i.e., the <x_y_ref> of the PLACE keyword). The <text_par> parameter defines the size of text, rotation, mirror, layer, text string, and dimensions and location of the bounding rectangle.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the ARTWORKS section. Examples of artwork attributes can be pin1 marker and bar_code. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

AN ARTWORKS Section Example

Here is an example of the ARTWORK section for a pin 1 marker on the silkscreen layer of the board:

```
$ARTWORKS
ARTWORK PIN1_MARKER

LAYER SILKSCREEN1
FILLED YES
TRACK 10
LINE 0 -100 0 100
LINE -100 0 100 0
CIRCLE 0 0 100
$ENDARTWORKS
```

The MECH Section

The MECH section is optional and can be used to describe fixing holes and mechanical components attached to the panel. The MECH section can have any of the following keywords.

\$MECH

```
FHOLE <x_y_ref> <drill_size>  
FIDUCIAL <fid_name>  
ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>  
ATTRIBUTE <attrib_ref>  
PLACE <x_y_ref>  
FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>  
ATTRIBUTE <attrib_ref>  
LAYER <layer>  
$ENDMECH
```

FHOLE <x_y_ref> <drill_size>

The optional FHOLE keyword defines holes in the panel that are used for fixture location. The <x_y_ref> parameter must be the absolute coordinates of the center of the hole. The <drill_size> parameter must be the hole diameter in <dimension> units. The parameter -1 (ASCII decimal 45 and ASCII decimal 49) can be used if the hole diameter is not specified. The parameter may also be 0 (ASCII decimal 48) if the location is just for reference and there is no hole (e.g., a fiducial point).

FIDUCIAL <fid_name>

The optional FIDUCIAL keyword must be a unique name given to the panel fiducial point. If one is not provided by the CAD system then names such as pan_fid1, pan_fid2 etc. must be used. The FIDUCIAL keyword must be followed by PLACE and LAYER keywords. This feature is not descriptive enough for today's CAD/CAM needs. We suggest using the FID feature for all new development. All parsers should be able to read and write this feature for backwards compatibility.

PLACE <x_y_ref>

The PLACE keyword must be included with the FIDUCIAL keyword to give the absolute *x* and *y* coordinates of the origin of the item on the panel.

LAYER <layer>

The LAYER keyword must be included with the FIDUCIAL keyword to specify which side of the panel the item is on. The layer parameter INNER or ALL should not be used.

FID <fid_name> <pad_name> <x_y_ref> <layer> <rot> <mirror> <flip>

The FID keyword defines the fiducial marking, using the pad definition as described in the PADS section or the PADSTACKS section, for each of the fiducials.

The <fid_name> parameter must be the fiducial fid name. The name is only for reference but must be unique within the MECH section.

The <pad_name> parameter must be the pad name as defined in the PADS section or the pad stack name defined in the PADSTACKS section of the GenCAD file.

The <x_y_ref> parameter must be the relative position of the center of the fiducial with respect to the board.

The <layer> parameter is the layer on which the pad or pad stack is placed. The actual layer on which the pad is situated will depend upon the <flip> parameter.

For pad stacks the layer parameter TOP means place the pad stack so that the pads lie on the layers defined in the pad stack. If the layer parameter is BOTTOM then all the layers are swapped TOP to BOTTOM so that, for example, a pad defined as TOP in the pad stack will be placed on the BOTTOM, pad defined as INNER1 will be on INNER4, INNER2 on INNER3, INNER3 on INNER2, INNER4 on INNER1 and BOTTOM on TOP. The whole padstacks, can be flipped by setting the flip parameter. To avoid confusion when using named inner layers it is best to define the pad stacks and shapes from the TOP.

The <rot> parameter must be the pad or pad stack rotation. The rotation is the angle between the pad or pad stack position as defined in the PAD or PADSTACK section and the pad or pad stack position, measured counterclockwise. Any pad or pad stack mirroring must be done before the pad or pad stack is rotated.

The <mirror> parameter must be the pad or pad stack mirror definition. The pad or pad stack can be placed normally or can be mirrored in either the x or y plane. When a pad stack is mirrored all the pads making up the stack are mirrored but stay on the same layer.

The <flip> parameter must be the flip definition. Flipping the fiducial is used to flip the pad or pad stack so that all the pad padstack layers re swapped TOP to BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2 etc.

ARTWORK <artwork_name> <x_y_ref> <rot> <mirror> <flip>

The ARTWORK keyword used to give the mechanical component any artwork feature that has been defined in the ARTWORKS section of the GenCAD file. Mechanical components may use different artworks at different locations on the board.

The <artwork_name> parameter must be the artwork name as defined in the ARTWORKS section in the GenCAD file.

The <x_y_ref> parameter must be the relative position of the artwork item origin with respect to the origin of the mechanical component.

The <rot> parameter must be the rotation of the artwork feature around the artwork origin. The rotation is the angle between the artwork feature as defined in the ARTWORK section and the position on the mechanical component, measured counterclockwise. Any mirroring must be done before the ARTWORK feature is rotated.

The <mirror> parameter must be the artwork feature mirror definition. The artwork feature can be placed normally or can be mirrored in either the x or y plane before becoming part of the mechanical component definition. When an artwork feature is mirrored all the artwork feature's item are mirrored but stay on the same layer.

The <flip> parameter must be the flip definition. Flipping the artwork feature is usually done when the CAD system provides artwork for the TOP of the board and assumes automatic layer

changes for mechanical components placed on the BOTTOM of the board. Flipping will change the layers for all the artwork features. For example, when flipping the artwork the features defined as being TOP will become BOTTOM. Inner layers are not flipped unless they are explicitly named INNER1, INNER2 etc. Flipping is independent of the mechanical component <layer> parameter, so if the CAD system does automatic flipping for mechanical components on the bottom then for every component placed on the bottom the <flip> parameter must be set

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the MECHANICAL, FID, FIDUCIAL, ARTWORK or SHAPE in the MECH section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A MECH Section Example

```
$MECH
FHOLE -4000 -6800 100
FHOLE 4000 6800 100
ARTWORK ORIGIN_MARKER 0 0 0 0
$ENDMECH
```

The BOARDS Section

The BOARDS section specifies which boards are fixed to the panel and their location on the panel. The BOARDS section must be included in the panel file.

```
$BOARDS

BOARD <string>
FILE <filename>
FIDUCIAL <x_y_ref>
ROTATION <rot>
SIGMOD <string>
PLACE <x_y_ref>
ATTRIBUTE <attrib_ref>
$ENDBOARDS
```

The \$BOARDS and \$ENDBOARDS mark the BOARDS section of the panel file.

BOARD <string>

The BOARD keyword is used to name or number the board. The parameter is a free field for the user to identify each board on the panel.

FILE <filename>

The FILE keyword is used to cross reference the board to the GenCAD file name.

FIDUCIAL <x_y_ref>

The FIDUCIAL keyword defines the board reference point. The parameter is the board reference point specified as an x,y coordinate relative to the board origin. The board reference point might be the board origin, a fixing hole, a mechanical part, a device pin or anything else related to the board. The reference point does not have to be within the board shape.

The board reference coordinates must be given in the dimensional units used in the GenCAD file for that board.

ROTATION <rot>

The ROTATION keyword defines the rotation of the board before it is placed on the panel. The rotation is measured counterclockwise from the GenCAD file orientation to the position on the panel, about the board origin as viewed from the top. Rotating the board about the reference point has the same affect as the board is always placed using the reference point. Board mirroring must be done before any rotation.

SIGMOD <string>

The SIGMOD is an optional keyword that can be used to modify identical names used on all boards on the panel. The parameter is used to change the board signal names, component names, pin names etc. so that all signals, component names, pin names etc. of all the boards on the panel are unique. The first character of the string must be a + (ASCII decimal 43) for prefixes or a - (ASCII decimal 45) for postfixes. The remaining text is either prefixed or postfixes to the signals, component names, pin names etc for that board. There must be no spaces in the name modifier.

For example, take the signal name DATA_2 and the name modifier + #1_ then the resulting signal would be #1_DATA_2. For the node name IC3-1 and the name modifier -@A the resulting node name would be IC3@A-1.

PLACE <x_y_ref>

The PLACE keyword defines the *x,y* coordinates of the board reference point relative to the origin (datum) of the panel. The dimensional units used here must be the unit defined in the HEADER section of the panel file.

ATTRIBUTE <attrib_ref>

The optional ATTRIBUTE keyword can be used to add other information that is considered relevant to the BOARDS section. See [Appendix A beginning on page 73](#) for the <attrib_ref> parameter definitions.

A BOARDS Section Example

```
$BOARDS
BOARD 1
FILE /usr/alf/pl/moth.cad
FIDUCIAL 0.0 0.0
ROTATION 0
SIGMOD +1
PLACE 16.00 16.00
BOARD 2
FILE /usr/alf/pl/moth.cad
FIDUCIAL 0 0
ROTATION 90
SIGMOD +2
PLACE -16 16
$ENDBOARDS
```

The CHANGES Section

The CHANGES section can be used to define a sequential list of changes or additions to the previously defined panel file sections. The changes listed must NOT have been implemented on the main sections of the panel file. The CHANGES section can reside in a separate file. The

CHANGES section is optional and can have some or all of the keywords listed below.

```
$CHANGES
CHANGE <string>
    <any panel type section>
$ENDCHANGES
```

CHANGE <string>

The CHANGE keyword defines the start of change information. The CHANGE parameter can be used to identify the change.

Changes to the HEADER section

The some of the HEADER keyword parameters can be changed by including a HEADER section in the CHANGES section. The syntax must be the same as that described in the main

HEADER section.

GENPANEL	Should not be changed unless the GenCAD specification changes are reflected in the file sections.
USER	The new user description replaces the old one.
DRAWING	Should not be changed.
REVISION	The new revision description replaces the old one.
UNITS	The units can be changed but there may be rounding errors on dimensions.
ORIGIN	The new origin coordinate replaces the old one.
INTERTRACK	The parameter can NOT be changed in the CHANGES section. Only CBTest, or other programs, can update this parameter.

Changes to the PANEL Section

The PANEL shape keywords and parameters can be changed by redefining the whole PANEL section in the CHANGES section. The panel mask area can be added, with the same syntax as described in the main PANEL section, by using the MASK keyword in a PANEL section within a CHANGE section.

Changes to the MECH Section

The fixture holes in the panel can be added to by defining new coordinates and drill size in a MECH section within a CHANGES section. Existing holes can be deleted by using the existing coordinates and a new drill size of 0.

Changes to the BOARDS Section

The BOARDS section can be changed by referring to the board name or number. A board can be removed from the panel by specifying coordinates that are off the panel.

Appendix A

Parameter Definitions

<arc_ref>	<p>The arcs defined by the GenCAD specification are either circular or elliptical. Parabolic or hyperbolic curves are not supported. Rectangular coordinates are always used.</p> <p>For circular arcs, the <arc_ref> is three coordinates of the type <x_y_ref>. Circular arcs are the most commonly used arcs and can be defined by:</p> <p>ARC <x_y_ref> <x_y_ref> <x_y_ref></p> <p>The first parameter is the start coordinate of the arc, the second parameter is the end coordinate of the arc, and the last parameter is the center of the circular arc. For example:</p> <p>ARC 1000 -200 1000 200 1000 0</p> <p>The GenCAD arc must always be drawn from the start coordinate counter-clockwise about the center to the end coordinate. This allows arcs to be defined for 0 to 360 degrees of rotation.</p> <p>Elliptical arcs may also be defined but they must have one axis parallel to the x axis and travel through an angle of 90 degrees or less. If the arc is elliptical then two more parameters are required of the type <number> in <dimension> units. The ARC definition becomes:</p> <p>ARC <x_y_ref> <x_y_ref> <x_y_ref> <number> <number></p> <p>The first parameter is the start coordinate of the arc, the second parameter is the end coordinate of the arc, and the third parameter is the center of the circular arc. The fourth and fifth parameters are the major radius and minor radius of the ellipse. Note that the axis of the ellipse must be in the horizontal and vertical planes.</p>
<artwork_name>	The name given to artworks must be of the type <string>.
<attrib_ref>	<p>The ATTRIBUTE keyword has the following parameters:</p> <p><string> <string> <string></p> <p>where the first parameter is the attribute category, the second is the attribute name and the third is the attribute data. The attribute category can be used to group a set of attributes.</p>
<circle_ref>	<p>One <x_y_ref> type parameter to define the center of the circle followed by one <number> type parameter to define the radius of the circle in <dimension> units.</p>

<component_name>	Component designator name on the PCB. These must be unique for each component on any one board and should be of the type <string>.
<dimension>	<p>This fixed field defines the dimension of a unit used throughout the GenCAD file.</p> <p>INCH—Inches.</p> <p>THOU—for thousandths of an inch.</p> <p>MM—Millimeters.</p> <p>MM100—for hundredths of a millimeter.</p> <p>USER <p_integer>—number of units to the inch.</p> <p>USERM <p_integer>—number of units to a centimeter.</p> <p>USERMM <p_integer>—number of units to a millimeter.</p>
<drill_size>	Drill size (hole diameter) in the type <number>. The drill size must always be in the same <dimension> units as defined in the HEADER section. Drill size codes are NOT supported.
<fid_name>	A name given to a unique fiducial using the type <string>.
<filename>	Any system-compatible file name using the type <string>. Do not include directory information with the file name. In general, lower-case letters are recommended for file names unless the operating system is able to distinguish between and maintain upper- and lowercase letters.
<filled_ref>	<p>Can have one of the following fixed fields.</p> <p>0—The following ARC, CIRCLE, LINE and RECTANGLE features are just drawn the width of the TRACK that is defined and referenced.</p> <p>YES —The following ARC, CIRCLE, LINE and RECTANGLE features form an enclosed and filled area. This stays active until another FILLED or LAYER change occurs. It is also terminated at the end of the ARTWORK section.</p>
<flip>	<p>Can have one of the following fixed fields.</p> <p>0—The shape is not flipped. Use the shape as defined in the shape section. (ASCII decimal 48).</p> <p>FLIP—Flip all layers for artwork, shape, pins, pad and pad stacks.</p>
<height>	Height from the finished board surface to the maximum height of the component shape by type <number> in <dimension> units.
<layer>	<p>Can have one of the following fixed fields.</p> <p>TOP—top of the board. This is sometimes called the component side.</p> <p>BOTTOM—bottom of the board. This must be the opposite side of the board to TOP.</p> <p>SOLDERMASK_TOP —soldermask on the top of the board. This must be the top side soldermask. Any feature using layer will swap with the soldermask_bottom when the <flip> parameter is set.</p>

SOLDERMASK_BOTTOM—solder mask on the bottom side of the board. This must be the opposite side of the board to soldermask_top. Any feature using layer will swap with the soldermask_top when the <flip> parameter is set.

SILKSCREEN_TOP—silkscreen on the top side of the board. This must be the top side silkscreen. Any feature using layer will swap

with the silkscreen_bottom when the <flip> parameter is set.

SILKSCREEN_BOTTOM—silkscreen on the bottom side of the board. This must be the opposite side of the board the silkscreen_top. Any feature using layer will swap with the silkscreen_top when the <flip> parameter is set.

SOLDERPASTE_TOP—solderpaste on the top side of the board. This must be the top side solderpaste. Any feature using layer will swap with the solderpaste_bottom when the <flip> parameter is set.

SOLDERPASTE_BOTTOM—solderpaste on the bottom side of the board. This must be the opposite side of the board to solderpaste_top. Any feature using layer will swap with the solderpaste_top when the <flip> parameter is set.

POWERx—specific power layer/plane.

GROUNDx—specific ground layer/plane.

INNER—all inner layers combined (i.e., not TOP, or BOTTOM, SILKSCREEN, SOLDERPASTE, SOLDERMASK).

INNERx—a specific inner layer where x can be 1 to 999 inclusive (e.g., INNER2, INNER8).

ALL—all copper layers of the board.

LAYERx—specific layers that can not be defined by the above parameters. x can be 0 to 1000 inclusive (e.g., LAYER0, LAYER1).

LAYERSETx—used for sets of layers as defined in the LAYERS section. x can have any value from 1 to 1000 inclusive.

<line_ref>

Two pairs of coordinates between which a straight line is drawn. The two parameters should be of the type <x_y_ref>.

<mirror>

This parameter can have one of the following fixed fields:

0—The item is NOT mirrored. (ASCII decimal 48).

MIRRORX—for an item mirrored about the x axis of the board or panel, as if a mirror is placed on the x axis causing all y coordinates to change while the x coordinates remain unchanged. All the mirrored item's y coordinates relative to the item's origin change sign, but not magnitude.

MIRROR—for an item mirrored about the y axis of the board or panel, as if a mirror is placed on the y axis causing all x coordinates to change while the y coordinates remain unchanged. All the mirrored item's x coordinates relative to the item's origin change sign, but not magnitude.

For the GenCAD file the x axis and y axis refer to the board axis and

NOT the component axis. For the panel file the x and y axis refer to the panel axis. Note that in some CAD systems the mirroring is about the x axis of the component no matter what its rotation is. Any mirroring of the item must be done before it is rotated. A mirrored item is still placed using the item's origin.

<number>

Any number within the range 3.4×10^{-38} to 3.4×10^{38} with 7 digit precision. The number can be positive or negative and can be a floating decimal point or can be an integer. The format is defined below.

Optional parameters are in { } brackets; s represents the sign of the number, either + (ASCII decimal 43) or - (ASCII decimal 45); d represents one or more digits (i.e., 0 1 2 3 4 5 6 7 8 9) and e represents e (ASCII decimal 101) or E (ASCII decimal 69) for the optional exponential part. Spaces are NOT allowed between ANY of the characters

making up the number.
{s} {d} { . } {d} { e {s} d }

<p_integer>

A positive integer number within the range 1 to 65535. A preceding + is allowed.

<pad_name>

The name given to pad shapes must be of the type <string>.

<pad_type>

FINGER—a solid rectangular pad with semicircular ends (obround).

ROUND—a solid circle.

ANNULAR—any shape ring of copper (width unspecified).

BULLET—a solid rectangular pad with one semicircular end.

RECTANGULAR —a solid rectangle or square.

HEXAGON—a solid hexagonal pad with equal length sides.

OCTAGON—a solid octagonal pad with equal length sides.

POLYGON—a solid polygon defined with LINES and ARCS.

UNKNOWN—unknown shape – not defined with LINES and ARCS.

<part_name>

A name given to a unique part and must be of the type <string>.

<pin_name>

Is the CAD pin name or number of the device and must be of the type <string>. For example 1, 12, 32a, A12.

<probe>

Is the test pin probe name and must have the type <string>.

<rectangle_ref>	A rectangle (or square) is defined as an <i>x,y</i> coordinate (relative to the origin of the shape, pad etc) of the type <x_y_ref> followed by an <i>x</i> dimension of type <number> and a <i>y</i> dimension of type <number>. The <i>x</i> and <i>y</i> dimensions can be positive and negative. For example, if the <i>x</i> dimension is positive and the <i>y</i> dimension is negative, the <i>x,y</i> coordinate is the top left hand corner of the rectangle. The rectangle sides must be parallel to the <i>x</i> and <i>y</i> axis of the shape definition. (The shape may then be rotated to any angle).
<rot>	A rotational angle of the type <number> degrees within the range 0 to 360, measured counterclockwise as viewed from the top, about the item's origin. Any rotation must be done after mirroring.
<shape_name>	The name given to shapes must be of the type <string>.
<sig_name>	The name given to signals must be of the type <string>.
<string>	A collection of ASCII characters. The characters must be within the range ! (ASCII decimal 32) to ~ (ASCII decimal 126) inclusive. Spaces are allowed if the string is enclosed in double quotes (ASCII decimal 34). If double quotes are required within the string, use the escape character \ (ASCII decimal 92) first.
<tan>	Tester Assigned Number must be of the type <string>.
<testpad_name>	The test land name must be of the type <string>.
<text_par>	<p>This parameter is made up of several parameters that define a text string.</p> <p>The first parameter must be the text size in the type <number> in <dimension> units.</p> <p>The second parameter must be the rotation of the text in the type <rot>. The text rotation is always counterclockwise about the text origin which is the bottom left corner of the first character of the string regardless of whether the string is mirrored or not.</p> <p>The third parameter must be the <mirror> definition.</p> <p>The fourth parameter must be the <layer> definition.</p> <p>The fifth parameter must be a string, and if spaces are to be included, the string should be enclosed in double quotes.</p> <p>The sixth parameter must be <rectangle_ref> defining the minimum, maximum rectangular area the text string must fit within.</p>
<tin>	Tester Interface Number must be of the type <string>.
<tp_name>	Test pin can be a driver, sensor, or power pin, and the name must be of the type <string>.
<track_name>	The track type name must be of the type <string>.
<track_width>	This is the track width in <dimension> units and must be in the type <number>.

GenCAD Specification 1.4

<via_name>

The via name must be of the type <string>.

<x_y_ref>

A pair of numbers defining the x and y coordinates of a point on or off board and should be of the type <number> <number>. For example - 1200 +3000 or 1.2005 0.0035

Appendix B

Mirror and Rotate Diagram

The GenCAD mirror and rotation results diagram shown on the next page shows an 8 pin DIL device placed on the top and bottom of a board at 90 degree rotations. The following points should be noted:

1. The original shape ($\langle \text{rot} \rangle = 0$ and $\langle \text{mirror} \rangle = 0$) was defined for the TOP of the board. It could have been defined for the bottom of the board. Whether the shape was defined for TOP or BOTTOM of the board the original shape will always have $\langle \text{rot} \rangle = 0$ and $\langle \text{mirror} \rangle = 0$.
2. All shapes are viewed from the TOP of the board looking down and through the board.
3. Mirroring **must** be applied BEFORE rotating.
4. MIRRORX mirrors the component about the x axis of the board. Mirroring is NOT done about the x or y axes of the component. The y coordinates of the shape change, the x coordinates remain the same.
5. MIRRORY mirrors the component about the y axis of the board. Mirroring is NOT done about the x or y axes of the component. The x coordinates of the shape change, the y coordinates remain the same.
6. Rotation is always counterclockwise as viewed from the top of the board, even if the component is on the BOTTOM.
7. Pads attached to the shape go through the same shape mirroring and rotation.
8. Rotation and mirroring do not affect the components position on the board. The component is still placed using the shape origin.
9. The same rules apply to pads on shapes and boards on panels.

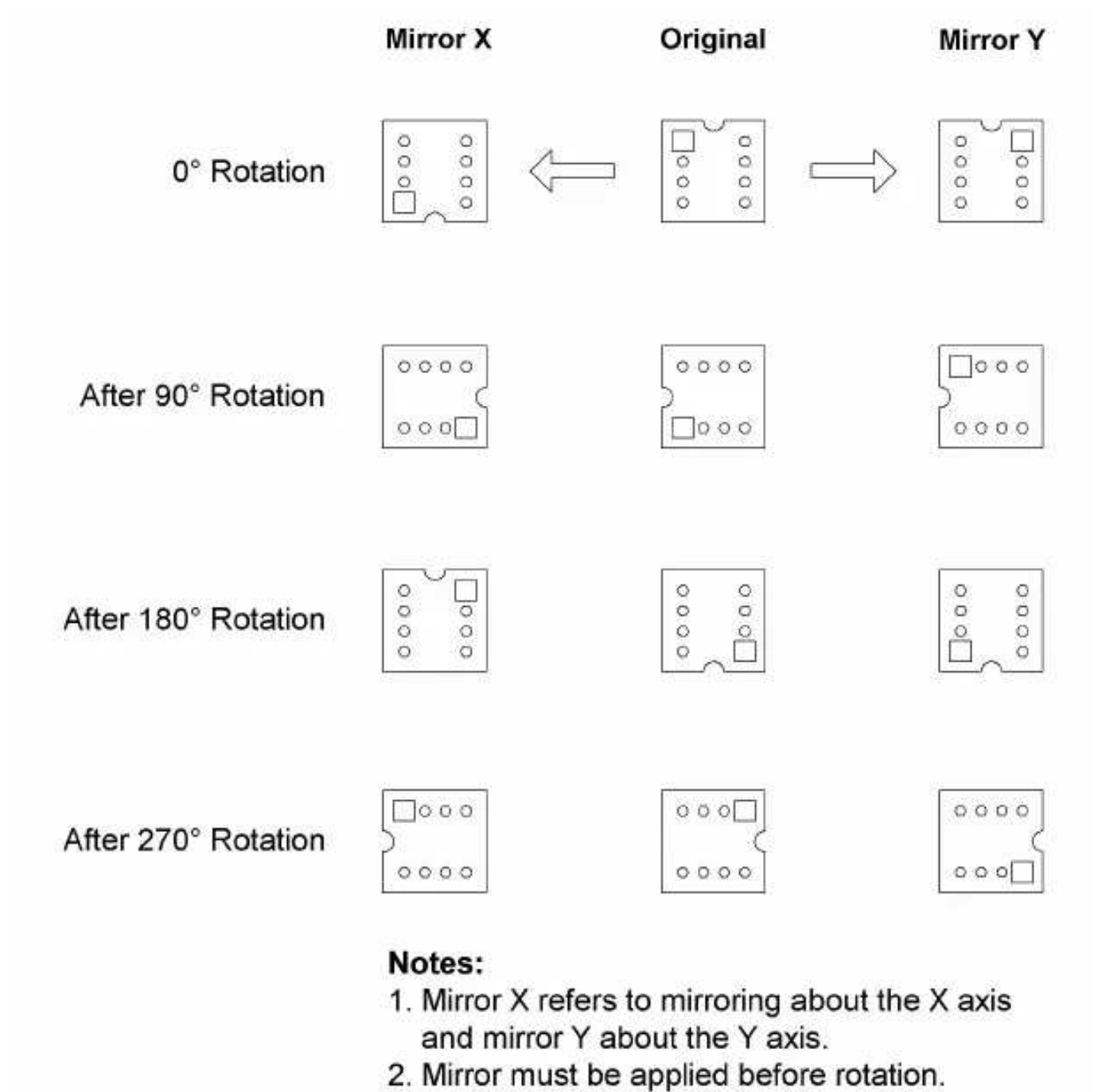


Figure 1. Mirror and Rotation Results

Appendix C

Version Change Information

Version 1.4

1. ARTWORK keywords have been added to BOARD, SHAPES, COMPONENTS, MECHANICAL and PANEL sections. Artwork is graphical information that is not associated with a specific network nor part of a component outline. Artwork can be used for auto plating, text on any layer (such as barcode designation), or other copper, silkscreen, soldermask geometry features.
2. All strings are now only limited to system resources. Spaces are allowed in the string if the string is enclosed in double quotes. (See the explanation for <string> in Appendix A.)
3. The only quantity limit in this specification is in the LAYERS section. Quantity limits have been removed from string size.
4. TEXT parameter—Added <rectangle_ref> to the <text_ref> to specify the minimum, maximum rectangle the text will fit within. This rectangle identifies the area the text will fill and can be used to feature clearances and eliminates any questions related to fonts sizes and shape issues between CAD systems.
5. BOARD section—Created ARTWORK to allow ARTWORK features attached to the board.
6. ARTWORKS section—Created ARTWORKS section to allow a library of artwork features to reference.
7. SHAPES section—ARTWORK keyword added. FID keyword added to allow a more complete definition of a fiducial. The FIDUCIAL keyword is now obsolete.
8. COMPONENTS section—ARTWORK feature added to create an instance of an artwork feature relative to the component.
9. MECH section—ARTWORK feature added to allow the instance of an ARTWORK feature relative to the MECHANICAL component. The FIDUCIAL keyword is now obsolete.
10. LAYER section—Added SILKSCREEN_TOP, SILKSCREEN_BOTTOM, SOLDERMASK_TOP, SOLDERMASK_BOTTOM, SOLDERPASTE_TOP, and SOLDERPASTE_BOTTOM, POWERx, GROUNDx.
11. PSEUDO section—Pseudo feature is now obsolete because of the removal of all the string limitations.
12. PANEL section—ARTWORK feature added to allow ARTWORK features attached to the panel.
13. Formatting changes.

Version 1.3

1. ATTRIBUTE keywords have been added to nearly every section and keyword as a method of adding extra data in a list form.
2. USERMM has been added to the <dimension> definition.
3. Additional pad shape keywords HEXAGON, OCTAGON and UNKNOWN have been added.
4. SHAPES section-INSERT keyword added.
5. DEVICES section-PROP keywords removed. The ATTRIBUTE keywords should be used instead.
6. INSERT keyword removed. The SHAPE INSERT keyword should now be used or DEVICE ATTRIBUTE used.
7. ROUTES section-PAD renamed to TESTPAD to avoid confusion with other types of pads.
8. PSEUDOS section-Parameters have been swapped.
9. PART, SHAPE, COMPONENT, PAD, PADSTACK, PIN, TRACK, VIA and TESTPAD keywords added.
10. MECH section-FIDUCIAL keyword added with associated PLACE, and LAYER keywords.
11. DEVICE keyword renamed to MECH, PLACE, LAYER, ROTATION and SHAPE keywords added.
12. PANEL file-THICKNESS added to PANEL section.
13. FIDUCIAL added to PANEL MECH section.

The initial version 1.3 was published on 30 March, 1994. Formatting was changed and descriptions of several sections was improved in the version published 5 December, 1994. There are no changes in format or interpretation.

Version 1.2

There has been some clarification to the text for GenCAD 1.2 copies marked DRAFT. There is NO difference in format or interpretation between GenCAD 1.2 DRAFT dated 23rd April 1993 and GenCAD 1.2 dated 12th August 1993. The items that have been clarified are:

1. Drill size is the diameter.
2. Layersets can be used with PADSTACKS.
3. Shapes (and their pads) are not automatically transferred on to the bottom of a board when a component is placed on the bottom as was the case in GenCAD 1.0 and GenCAD 1.1. The <flip> parameter must be used to explicitly move the shape as described in GenCAD 1.2
4. The DEVICE <part_name> may also be <component_name> when there is a one to one cross reference between the COMPONENTS and the DEVICE sections.
5. Alternate pin names or numbers can not be used in the NODE <pin_name> parameter.
6. The same node may appear in different signals (connected nets). The interpretation is let to the end user.
7. INNERx layers can be used for any type of inner layer. If TOP and BOTTOM layers are used they must be copper.
8. Each via can have only ONE VIA definition.
9. Rectangle definition <rectangle_ref> example given.

Version 1.2 draft

1. Minor text changes and section renumbering.
2. BOARD section-An optional board thickness is now included.
3. The MASK keyword now has a layer parameter.
4. An optional RECTANGLE keyword has been added.
5. PADS section-An optional RECTANGLE keyword has been added.
6. PADSTACKS section-A completely new optional section has been added for CAD systems that use pad stacks.
7. SHAPES section-An optional RECTANGLE keyword has been added.
8. An optional FIDUCIAL keyword has been added.
9. COMPONENTS section-A flip parameter has been added to the SHAPE keyword.
10. The <designator> parameter name has been changed to <component_name> parameter. This is just a change in name within this specification and does not imply any change of CAD data.
11. An optional SHEET keyword has been added.
12. DEVICES section-An optional PART keyword has been added.
13. An optional PACKAGE keyword has been added.

GenCAD Specification 1.4

14. The PIN keyword in the DEVICES section has been renamed to PINFUNCT to avoid confusion with the PIN keyword in the SHAPES section. Only one PINFUNCT parameter is now allowed per pin.
15. SIGNALS section-The NODE parameter <pin_name> has been changed to <component_name> <pin_name>. There is no hyphen between the two parameters. See <pin_name> below.
16. Only one ATTRIBUTE keyword per signal is now allowed.
17. TRACKS section-The TRACK parameter <track_name> has been changed from <p_integer> to <string8>.
18. LAYERS section-A completely new optional section has been added to define layers and layer sets.
19. ROUTES section-The PAD keyword has an additional parameter to give each pad or test land a unique name.
20. The VIA keyword has an additional pad name parameter.
21. The PLANE keyword has an additional plane name parameter.
22. An optional RECTANGULAR keyword has been added.
23. Only one ROUTE keyword per signal is allowed.
24. An optional TEXT keyword has been included to accept data for text strings in copper (or silk screen).
25. The names given to vias must now be unique.
26. MECH section-FHOLE can now be used for board fiducial points that do not have holes.
27. PANEL section-An optional RECTANGULAR keyword has been added.
28. The MASK keyword now has a layer parameter.
29. BOARDS section-The section has been changed to include keywords.
30. <pin_name>-Alphanumeric pin names are now allowed. All references to pin_number have been changed to pin_name.
31. <rot>-Rotation parameter has been changed from <p_integer> to <number> so that fractions of a degree can be specified.
32. A mirror and rotation diagram has been added—see [Appendix C beginning on page 81](#).

Version 1.1

1. Minor changes to text and section renumbering.
2. BOARD section - MASK - addition for internally masked off areas.
3. ROUTES section - VIA - <via_name> added.
4. TESTPINS section - TESTPIN - <layer> added.
5. TESTPINS section - TEXT - <tp_name> deleted. The text must follow its TESTPIN definition.
6. <text_par> parameter - text is limited to <string24>.
7. PSEUDOS section -PART keyword changed to DEVICE keyword. The PART keyword is reserved for the actual device <part_name> used in manufacturing the board, as opposed to the CAD <part_name>.
8. CHANGES section - PART keyword changed to DEVICE keyword.
9. PANEL file definition added.
10. SHAPES section - PIN - additional drill hole sized pads can be used.
11. Clarification of the rotate and mirror parameters.

Version 1.0

Original specification.

