CadPack

Import from THEDA

Software tool for import from THEDA Cad format

Technical Info

Version : 2
Code : 81190417.136

SPEA SpA
16, Via Torino
10088 Volpiano - Italy
Tel.: + 39 011 9825 400
Fax: + 39 011 9825 405
E-mail: info@spea.com
Web: www.spea.com

Contents

Introduction **III**

1. THEDA file data **1**

1.1 Part List ... 2

1.2 Net List ... 3

1.3 Coordinates and access list ... 4

1.4 Wiring and Routing list ... 4

2. THEDA file generalities **5**

2.1 THEDA file name ... 5

2.2 THEDA file conversion from Unix to MS-DOS ... 5

3. THEDA file format **6**

3.1 UNIT ... 12

3.2 IDENTIFIER ... 12

3.3 DEVICE_ID ... 13

3.4 PACKAGE_ID ... 13

3.5 PENS ... 13

3.6 REFERENCE_DESIGNATOR ... 14

3.7 LAYER ... 14

3.8 ROTATION ... 15

3.9 X_MIRROR ... 15

3.10 CONTACTED ... 15

3.11 LOCATION ... 16

3.12 PADSTACK_ID ... 17

3.13 LINE_PEN ... 17

3.14 LINE 17
3.15 POLYGON 18
3.16 LIBRARY_IDENTIFIER 18
3.17 UNCONNECTED 18
3.18 PIN_ID 18

4. Import setting **19**

4.1 Pin function assignment 19
4.2 Drawing ref. initials/device type assignment 19

A. Note about the THEDA ASCII text file format **20**

Introduction

CAD files are the base for the automatic generation of test program for InCircuit of any technology.

In order to generate the ICT test program in a short time and without errors, both Bed of Nails and Flying Probe testers require the circuit information available in CAD format.

The Import from THEDA CAD import driver allows to import data present in the THEDA CAD file and convert them in the SPEA Board data format.

Conventions, symbols and abbreviations

In the document, the ⓘ symbol is used to highlight information or notes useful to the reader.

This manual can be updated in accordance with the evolution of the system and associated software.
It may contain preliminary contents or it may not be entirely updated with the latest versions used in the system.

Any remarks on errors and imperfections, or suggestions, can be addressed to:

SPEA SpA
Ufficio Documentazione
16, Via Torino
10088 Volpiano – Italy
Tel.: +39 011 9825400
Fax: +39 011 9825405
Email: info@spea.com
Web: www.spea.com

1. THEDA file data

With the “THEDA CAD files” words we refer to the output information generated by the THEDA CAD-CAE programs for the electrical diagrams design and PCB development, used to develop a test application (test program and adapter design).

Information stored in the “THEDA CAD files” concern an electronic board and can be used by an appropriate program to generate a test program and its test adapter design (Bed of Nails or list of movements for Flying Probes).

Information can be grouped in 4 different categories and typically concern the printed circuit:

Part List
It is the list of all used devices, it must contain: devices drawing reference, part numbers, value, tolerances, device type, etc.
Net List
It is also called wiring list, containing device interconnection data; basically it is presentation of the electrical diagram.
Coordinate and access list
It is the list containing the devices coordinates, concerning their barycentre and pins.
Wiring and Routing list
It is the list containing the path of the Net tracks in the PCB.

For the import of the information above mentioned SPEA has developed the specific program for the translation, stored in a specified format, to its common data bank called “Board Data”.
The name of this type of program is “CAD import driver”.

For the required information, see the list in the following paragraphs.

1.1 Part List

The Part List is an ASCII text file, containing the list of all the parts used to assemble the board; sometimes it can be called **Bill of Material** (BOM).

In the Part List all information concerning the mounted and not mounted parts must be present.
For every part the following information must be defined:

Information	Description
Drawing Reference	Reference designator (e.g. U10, R105, D23, etc.).
Part Number	Device code (e.g. 132549.012, C4QW08, 001-58-AA, etc.).
Value	Device value (e.g. 10KΩ, 10µF, 1mH, etc.).
Tolerance	Positive and negative device tolerances (e.g. 1%, 5%, etc.).
Mounting side	The legal values for this item can be: - **Top** (Component side) - **Bottom** (Soldering side) - **Not mounted Top** - **Not mounted Bottom**
Rotation [1]	Device mounting rotation angle (e.g. 0°, 180°, etc.).
Dimensions [1]	Device dimensions.
Case code [1]	Device package (case) code.

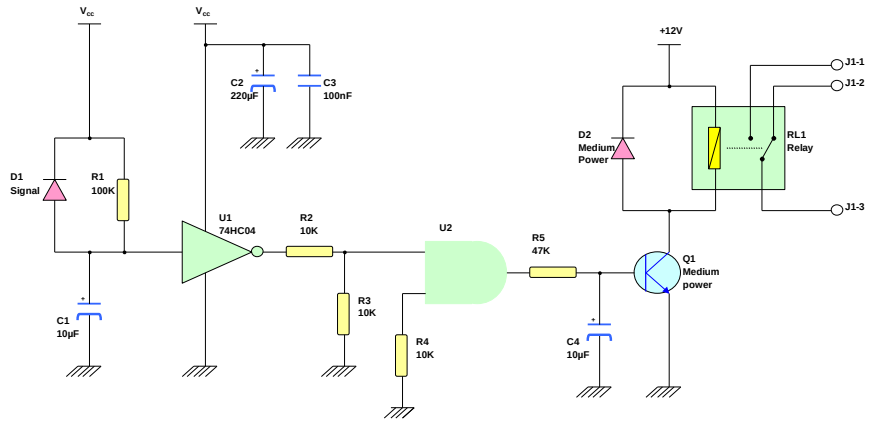
[1] **Optional data (not yet managed)**

1.2 Net List

The Net List is an ASCII text file containing the device interconnection data; it is also called wiring list.
This list must contain the interconnection between devices, including pad and via.
Basically, it is the representation of the electrical diagrams.

For every net the following information must be defined:

Information	Description
Net name	Net identifier (e.g. +5V, RESET, A01, etc.).
Drawing reference	Reference designator of the device connected to the net (e.g. U10, R105, D23, etc.).
Pin name	Name of the device pin connected to the net (e.g. 1, 15, Anode, K, Negative, etc.).
Pin access side	Access side for the device pin, legal values are: - **Top** (Device side access). - **Bottom** (Soldering side access). - **Not accessible** - **All** (both top and bottom side access)



1.3 Coordinates and access list

The Coordinates and access list is an ASCII text file containing the devices coordinates concerning their barycentre and pins. Below, the required information:

Information	Description
Drawing Reference	Reference designator of the device connected to the net (e.g. U10, R105, D23, etc.).
Pin name	Name of the device pin connected to the net (e.g. 1, 15, Anode, K, Negative, etc.).
Pin X position	Pin X-coordinate.
Pin Y position	Pin Y-coordinate.
X barycentre [1]	Device X barycentre.
Y barycentre [1]	Device Y barycentre.

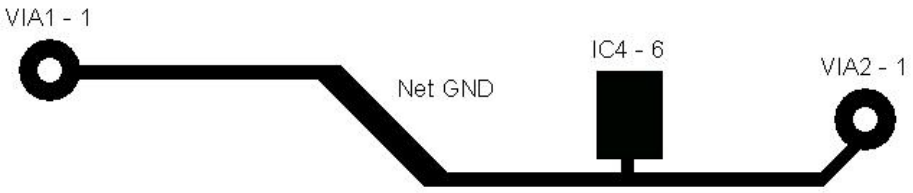
1.4 Wiring and Routing list

The Wiring and Routing list is an ASCII text file that contains all the coordinates of the Net tracks on the PCB and the link with the Net List. So the path of each net on the PCB is described in this file.

For every net the following information must be defined:

Information	Description
Net name	Net identifier (e.g. +5V, RESET, A01, etc.).
X Start	Track segment start X-coordinate.
Y Start	Track segment start Y-coordinate.
X End	Track segment end X-coordinate.
Y End	Track segment end Y-coordinate.
Width	Net segment thickness.
Layer	Layer the segment belongs to.

Example:



[1] **Optional data**

2. THEDA file generalities

2.1 THEDA file name

The THEDA Neutral file name has to have the **.TL** extensions.
It is an ASCII text file and it contains the information related to the board, device and their connections.

2.2 THEDA file conversion from Unix to MS-DOS

When the diagram entry has been entered and checked on the THEDA CAD workstation, the THEDA file **.TL** should be made available for the SPEA system.

The SPEA system is based on a PC platform operating in a Windows® environment, this means that the CAD import driver can manage ASCII Text file in MS-DOS format.

Due to the fact that the THEDA workstation typically uses the Unix operating system, the output ASCII text file has to be converted from Unix to MS-DOS format.

In order to perform the conversion, please refer to appendix A – **Note about the THEDA ASCII text file format**.

3. THEDA file format

This is a partial extract of an example of a THEDA output ASCII text file:

```
PC_BOARD
!  IDENTIFIER              := 'SPEA_DEMO';
!  DESCRIPTION             := 'SPEA_DEMO';
!  UNIT                    := 1 MM;
!  WORKING_AREA            := [ ( 0, 0 ), ( 1, 1 ) ];
!  TECHNIQUE               := D;
!  VERSION
!  !  TL_REVISION             := '2.00';
!  !  TL_COMMENT              := 'TL file generated by AUTOBOARD';
!  !  SOFTWARE_REVISION       := 'THEDA 6.0.17';
!  !  SOFTWARE_COMMENT        := 'THEDA Revision 6.0.17 16-Sep-1999';
!  !  LIBRARY_REVISION        := '9.00';
!  !  BOARD_REVISION          := '8.00';
!  !  CREATION_DATE           := '25-FEB-2003';
!  !..FIN_VERSION;
!  DESIGN_DEFAULTS
……
!  DESIGN_OBJECTS
!  !  PENS
!  !  @  LINE_PENS
!  !  @  !  #---------------------------------------#
!  !  @  !  NUMBER   WIDTH   SHAPE_TYPE   LINE_STYLE
!  !  @  !  #---------------------------------------#
!  !  @  !  # 1-60 are used for traces on the board.
!  !  @  !    1       .2       ROUND         SOLID
!  !  @  !    2       .3       ROUND         SOLID
!  !  @  !    3       .4       ROUND         SOLID
!  !  @  !    4       .5       ROUND         SOLID
!  !  @  !    5       .6       ROUND         SOLID
!  !  @  !    6       .76      ROUND         SOLID
!  !  @  !    7        1       ROUND         SOLID
!  !  @  !    8       1.5      ROUND         SOLID
!  !  @  !    9        2       ROUND         SOLID
!  !  @  !    10     2.54      ROUND         SOLID
……
!  !  @  !  #---------------------------------------#
!  !  @  !..FIN_LINE_PENS;
……
!  !  @..FIN_PENS;
……
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  IDENTIFIER             := '9030619';
!  !  @  !  DEVICE_ID              := 'dev_stift_1';
!  !  @  !  PACKAGE_ID             := 'pack_tpoint';
……
!  !  @  !..FIN_COMPONENT[ 1 ];
……
!  !  @  COMPONENT[ 8 ]
!  !  @  !  IDENTIFIER             := '15343421';
!  !  @  !  DEVICE_ID              := 'dev_diode';
!  !  @  !  PACKAGE_ID             := 'pack_chip2012';
!  !  @  !..FIN_COMPONENT[ 8 ];
……
!  !  @  COMPONENT[ 12 ]
!  !  @  !  IDENTIFIER             := '9543052';
!  !  @  !  DEVICE_ID              := 'dev_widerstand';
!  !  @  !  PACKAGE_ID             := 'pack_DO214';
!  !  @  !..FIN_COMPONENT[ 12 ];
……
!  !  @  COMPONENT[ 18 ]
!  !  @  !  IDENTIFIER             := '9547280';
!  !  @  !  DEVICE_ID              := 'dev_elko';
!  !  @  !  PACKAGE_ID             := 'pack_chip7343';
!  !  @  !..FIN_COMPONENT[ 18 ];

……
!  !  @  COMPONENT[ 28 ]
!  !  @  !  IDENTIFIER             := '95587589';
!  !  @  !  DEVICE_ID              := 'dev_opto_ ';
!  !  @  !  PACKAGE_ID             := 'pack_opto';
!  !  @  !..FIN_COMPONENT[ 28 ];
……
!  !  @  COMPONENT[ 33 ]
```

```
!  !  @  !  IDENTIFIER              := '95551354';
!  !  @  !  DEVICE_ID               := 'dev_widerstand';
!  !  @  !  PACKAGE_ID              := 'pack_chip0603.1';
!  !  @  !..FIN_COMPONENT[ 33 ];

!  !  @..FIN_COMPONENTS;
!  !..FIN_DESIGN_OBJECTS;
……
!  NET_LIST
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9562810';
!  !  @  !..FIN_COMPONENT[ 1 ];
!  !  @  COMPONENT[ 2 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'D1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9571794';
!  !  @  !..FIN_COMPONENT[ 2 ];
!  !  @  COMPONENT[ 3 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'DL1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9564728';
!  !  @  !..FIN_COMPONENT[ 3 ];
!  !  @  COMPONENT[ 4 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9528360';
!  !  @  !..FIN_COMPONENT[ 4 ];
!  !  @  COMPONENT[ 5 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9520634';
!  !  @  !..FIN_COMPONENT[ 5 ];
……
!  !  @..FIN_COMPONENTS;
!  !  NETS
!  !  @  NET[ 1 ]
!  !  @  !  IDENTIFIER              := 'NET_1';
!  !  @  !  PINS
!  !  @  !  !  PIN[ 1 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'D1';
!  !  @  !  !  @  PIN_ID                  := '1';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (  105,    1 );
!  !  @  !  !  @..FIN_PIN[ 1 ];
!  !  @  !  !  PIN[ 2 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  !  @  PIN_ID                  := '1';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (  106,    1 );
!  !  @  !  !  @..FIN_PIN[ 2 ];
!  !  @  !  !..FIN_PINS;
!  !  @  !..FIN_NET[ 1 ];
!  !  @  NET[ 2 ]
!  !  @  !  IDENTIFIER              := 'NET_2';
!  !  @  !  PINS
!  !  @  !  !  PIN[ 1 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  !  @  PIN_ID                  := '1';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (   39,    1 );
!  !  @  !  !  @..FIN_PIN[ 1 ];
!  !  @  !  !  PIN[ 2 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  !  @  PIN_ID                  := '2';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (   15,    1 );
!  !  @  !  !  @..FIN_PIN[ 2 ];
!  !  @  !  !..FIN_PINS;
!  !  @  !..FIN_NET[ 4 ];
!  !  @  NET[ 3 ]
!  !  @  !  IDENTIFIER              := 'GND';
!  !  @  !  PINS
!  !  @  !  !  PIN[ 1 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  !  @  PIN_ID                  := '1';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (   38,    1 );
!  !  @  !  !  @..FIN_PIN[ 1 ];
!  !  @  !  !  PIN[ 2 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  !  @  PIN_ID                  := '8';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (   16,    1 );
!  !  @  !  !  @..FIN_PIN[ 2 ];
!  !  @  !  !..FIN_PINS;
!  !  @  !..FIN_NET[ 3 ];
!  !  @  NET[ 4 ]
```

```
!  !  @  !  UNCONNECTED;
!  !  @  !  PINS
!  !  @  !  !  PIN[ 1 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  !  @  PIN_ID                  := '2';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (    5,    7 );
!  !  @  !  !  @..FIN_PIN[ 1 ];
!  !  @  !  !  PIN[ 2 ]
!  !  @  !  !  @  REFERENCE_DESIGNATOR    := 'D1';
!  !  @  !  !  @  PIN_ID                  := '1';
!  !  @  !  !  @  SCHEMATIC_SYM_PIN       := (   50,    1 );
!  !  @  !  !  @..FIN_PIN[ 2 ];
!  !  @  !  !..FIN_PINS;
!  !  @  !..FIN_NET[ 4 ];
!  !  @..FIN_NETS;
!  !..FIN_NET_LIST;
!  PHYSICAL_LAYOUT
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  ROTATION                :=    270;
!  !  @  !  LOCATION                := ( 24.12992, 39.89904 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  NUMBER                  := 1;
!  !  @  !  !  @  !  IDENTIFIER              := '1';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p2443';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  NUMBER                  := 2;
!  !  @  !  !  @  !  IDENTIFIER              := '2';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  ROTATION                :=    180;
!  !  @  !  !  @  !  LOCATION                := ( 6.700001, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p2443';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 1 ];
!  !  @  COMPONENT[ 2 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'D1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  LOCATION                := ( 12.80579, 35.03072 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  NUMBER                  := 1;
!  !  @  !  !  @  !  IDENTIFIER              := '1';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p1624';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  NUMBER                  := 2;
!  !  @  !  !  @  !  IDENTIFIER              := '2';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 4.000001, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p1624';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 2 ];
!  !  @  COMPONENT[ 3 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'DL1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  LOCATION                := ( 27.19908, 4.23332 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  NUMBER                  := 1;
!  !  @  !  !  @  !  IDENTIFIER              := '1';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
```

```
!  !  @  !  !  @  !  LOCATION                := ( 0, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p1015';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  NUMBER                  := 2;
!  !  @  !  !  @  !  IDENTIFIER              := '2';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 2, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p1015';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 3 ];
!  !  @  COMPONENT[ 4 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  X_MIRROR;
!  !  @  !  ROTATION                :=    320;
!  !  @  !  LOCATION                := ( 15.488, 9.144 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  NUMBER                  := 1;
!  !  @  !  !  @  !  IDENTIFIER              := '1';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, 0 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  NUMBER                  := 2;
!  !  @  !  !  @  !  IDENTIFIER              := '2';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, .5 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p02504';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @  PIN[ 3 ]
!  !  @  !  !  @  !  NUMBER                  := 3;
!  !  @  !  !  @  !  IDENTIFIER              := '3';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, .999999 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 3 ];
!  !  @  !  !  @  PIN[ 4 ]
!  !  @  !  !  @  !  NUMBER                  := 4;
!  !  @  !  !  @  !  IDENTIFIER              := '4';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [  ];
!  !  @  !  !  @  !  LOCATION                := ( -4.768372e-07, 1.4 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 4 ];
!  !  @  !  !  @  PIN[ 5 ]
!  !  @  !  !  @  !  NUMBER                  := 5;
!  !  @  !  !  @  !  IDENTIFIER              := '5';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( 0, 1.799999 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 5 ];
!  !  @  !  !  @  PIN[ 6 ]
!  !  @  !  !  @  !  NUMBER                  := 6;
!  !  @  !  !  @  !  IDENTIFIER              := '6';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( -4.768372e-07, 2.199999 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 6 ];
!  !  @  !  !  @  PIN[ 7 ]
!  !  @  !  !  @  !  NUMBER                  := 7;
!  !  @  !  !  @  !  IDENTIFIER              := '7';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  !  LOCATION                := ( -4.768372e-07, 2.599999 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 7 ];
!  !  @  !  !  @  PIN[ 8 ]
!  !  @  !  !  @  !  NUMBER                  := 8;
!  !  @  !  !  @  !  IDENTIFIER              := '8';
!  !  @  !  !  @  !  NEEDED                  := [  ];
!  !  @  !  !  @  !  CONTACTED               := [ SOLDER_SIDE ];
```

```
!  !  @  !  !  @  !  LOCATION                := ( 0, 2.999999 );
!  !  @  !  !  @  !  PADSTACK_ID             := 'p025022';
!  !  @  !  !  @  !..FIN_PIN[ 8 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 4 ];
!  !  @  COMPONENT[ 5 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  ROTATION                :=    180;
!  !  @  !  LOCATION                := ( 18.30911, 4.974151 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  NUMBER                 := 1;
!  !  @  !  !  @  !  IDENTIFIER             := '1';
!  !  @  !  !  @  !  NEEDED                 := [  ];
!  !  @  !  !  @  !  CONTACTED              := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  LOCATION               := ( 0, 0 );
!  !  @  !  !  @  !  PADSTACK_ID            := 'p0610a';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  NUMBER                 := 2;
!  !  @  !  !  @  !  IDENTIFIER             := '2';
!  !  @  !  !  @  !  NEEDED                 := [  ];
!  !  @  !  !  @  !  CONTACTED              := [ COMPONENT_SIDE ];
!  !  @  !  !  @  !  ROTATION               :=    180;
!  !  @  !  !  @  !  LOCATION               := ( 1.6, 0 );
!  !  @  !  !  @  !  PADSTACK_ID            := 'p0610a';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 5 ];
.....
!  !  @..FIN_COMPONENTS;
!  !  SIGNALS
!  !  @  SIGNAL[ 1 ]
!  !  @  !  IDENTIFIER             := 'NET1';
!  !  @  !  TRACES
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE_PEN                := 7;
!  !  @  !  !  LINE                    := [ (  3.704155,  37.93016 ),
!  !  @  !  !                              (  3.704155,  38.84071 ) ];
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE                    := [ (  3.704155,  36.40655 ),
!  !  @  !  !                              (  3.704155,  37.93016 ) ];
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE                    := [ (  4.762485,  35.34822 ),
!  !  @  !  !                              (  3.704155,  36.40655 ) ];
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE                    := [ (  8.358555,  38.52321 ),
!  !  @  !  !                              (  7.619976,  38.52321 ) ];
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE                    := [ (  9.207471,  38.52321 ),
!  !  @  !  !                              (  8.358555,  38.52321 ) ];
!  !  @  !  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  LINE                    := [ (  3.704155,  37.93016 ),
!  !  @  !  !                              (  3.704155,  39.05238 ) ];
!  !  @  !  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  LINE                    := [ (  3.704155,  36.40655 ),
!  !  @  !  !                              (  3.704155,  37.93016 ) ];
!  !  @  !  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  LINE                    := [ (  8.358556,  38.52321 ),
!  !  @  !  !                              (  7.619976,  38.52321 ) ];
!  !  @  !  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  LINE                    := [ (  9.207471,  38.52321 ),
!  !  @  !  !                              (  8.358556,  38.52321 ) ];
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  POLYGON                 := [ (  9.207471,  38.52321 ),
!  !  @  !  !                              (  14.18162,  38.52321 ),
!  !  @  !  !                              (  14.60495,  38.09988 ) ];
!  !  @  !  !..FIN_TRACES;
!  !  @  !  TEST_POINTS
!  !  @  !  !  TEST_POINT[ 1 ]
!  !  @  !  !  @  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  @  LOCATION                := ( 3.598322, 37.46488 );
!  !  @  !  !  @  NEEDED                  := [  ];
!  !  @  !  !  @  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  PADSTACK_ID             := 'tptdef';
!  !  @  !  !  @..FIN_TEST_POINT[ 1 ];
!  !  @  !  !  TEST_POINT[ 2 ]
!  !  @  !  !  @  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  @  LOCATION                := ( 8.043308, 38.62905 );
```

```
!  !  @  !  !  @  NEEDED                  := [  ];
!  !  @  !  !  @  CONTACTED               := [ SOLDER_SIDE ];
!  !  @  !  !  @  PADSTACK_ID             := 'tptdef';
!  !  @  !  !  @..FIN_TEST_POINT[ 2 ];
!  !  @  !  !..FIN_TEST_POINTS;
!  !  @  !  VIAS
!  !  @  !  !  VIA[ 1 ]
!  !  @  !  !  @  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  @  LOCATION                := ( 3.704155, 36.40655 );
!  !  @  !  !  @  NEEDED                  := [ COMPONENT_SIDE,
!  !  @  !  !  @                             SOLDER_SIDE ];
!  !  @  !  !  @  CONTACTED               := [ COMPONENT_SIDE,
!  !  @  !  !  @                             SOLDER_SIDE ];
!  !  @  !  !  @  PADSTACK_ID             := 'viadef';
!  !  @  !  !  @..FIN_VIA[ 1 ];
!  !  @  !  !  VIA[ 2 ]
!  !  @  !  !  @  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  @  LOCATION                := ( 9.207471, 38.52321 );
!  !  @  !  !  @  NEEDED                  := [ COMPONENT_SIDE,
!  !  @  !  !  @                             SOLDER_SIDE ];
!  !  @  !  !  @  CONTACTED               := [ COMPONENT_SIDE,
!  !  @  !  !  @                             SOLDER_SIDE ];
!  !  @  !  !  @  PADSTACK_ID             := 'viadef';
!  !  @  !  !  @..FIN_VIA[ 2 ];
!  !  @  !  !..FIN_VIAS;
!  !  @  !..FIN_SIGNAL[ 1 ];
!  !  @..FIN_SIGNALS;
!  !..FIN_PHYSICAL_LAYOUT;
……
!..FIN_PC_BOARD;
```

The Import from THEDA CAD driver is able to correctly identify and use the following labels (identifiers):

- **UNIT**
- **IDENTIFIER**
- **DEVICE_ID**
- **PACKAGE_ID**
- **PENS**
- **REFERENCE_DESIGNATOR**
- **LAYER**
- **ROTATION**
- **LOCATION**
- **X_MIRROR**
- **CONTACTED**
- **PADSTACK_ID**
- **LINE_PEN**
- **LINE**
- **POLYGON**
- **LIBRARY_IDENTIFIER**
- **UNCONNECTED**
- **PIN_ID**

In the next paragraphs, a short description for each label (identifier) is provided.

3.1 UNIT

The **UNIT** identifier is used to specify the used unit of measurement for coordinates and dimensions.

Valid values are:

- **MM** to specify that all the coordinates and dimensions are expressed in millimeters
- **INCHES** to specify that all the coordinates and dimensions are expressed in inches

The following example shows the syntax used for the **UNIT** identifier:

```
.....
!  UNIT                    := 1 MM;
.....
```

3.2 IDENTIFIER

The **IDENTIFIER** keyword is used to identify an object inside of the section.

The meaning of the keyword changes depending on the section where it is used:

- In the **COMPONENT** section it identifies the Part Number
- In the **PIN** section it identifies the Pin Name
- In the **SIGNAL** and **NET** section it identifies the Net Name

Example of **IDENTIFIER** keyword:

```
.....
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  IDENTIFIER              := '9400318';
.....
!  !  @  !..FIN_COMPONENT[ 1 ];
.....
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  IDENTIFIER              := '1';
.....
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
.....
!  !  SIGNAL[ 1 ]
!  !  @  !  IDENTIFIER              := 'NET1';
.....
!  !  @  !..FIN_SIGNAL[ 1 ];
.....
!  @  NET[ 1 ]
!  !  @  !  IDENTIFIER              := 'NET_1';
.....
!  !  @  !..FIN_NET[ 1 ];
.....
```

3.3 DEVICE_ID

The **DEVICE_ID** keyword contains the Device Name of the component.

The following example shows the used syntax for the **DEVICE_ID** identifier:

```
.....
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  IDENTIFIER              := '9200719';
!  !  @  !  DEVICE_ID              := 'dev_stift';
!  !  @  !  PACKAGE_ID             := 'pack_tpoint';
.....
!  !  @  !..FIN_COMPONENT[ 1 ];
.....
```

3.4 PACKAGE_ID

The **PACKAGE_ID** keyword contains the Package Name of the component.

The following example shows the used syntax for the **PACKAGE_ID** identifier:

```
.....
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  IDENTIFIER              := '9009679';
!  !  @  !  DEVICE_ID              := 'dev_stift';
!  !  @  !  PACKAGE_ID             := 'pack_ tpoint';
.....
!  !  @  !..FIN_COMPONENT[ 1 ];
.....
```

3.5 PENS

The **PENS** subsection describes the lines width.
This information is used in the **TRACES** section for tracks width.

Example of **PENS** subsection:

```
.....
!  !  PENS
!  !  @  LINE_PENS
!  !  @  !  #---------------------------------------#
!  !  @  !  NUMBER   WIDTH   SHAPE_TYPE   LINE_STYLE
!  !  @  !  #---------------------------------------#
!  !  @  !  # 1-60 are used for traces on the board.
!  !  @  !    1       .2       ROUND         SOLID
!  !  @  !    2       .3       ROUND         SOLID
!  !  @  !    3       .4       ROUND         SOLID
!  !  @  !    4       .5       ROUND         SOLID
!  !  @  !    5       .6       ROUND         SOLID
!  !  @  !    6       .76      ROUND         SOLID
!  !  @  !    7        1       ROUND         SOLID
!  !  @  !    8       1.5      ROUND         SOLID
!  !  @  !    9        2       ROUND         SOLID
!  !  @  !    10     2.54      ROUND         SOLID
.....
!  !  @  !  #---------------------------------------#
!  !  @  !..FIN_LINE_PENS;
.....
!  !  @..FIN_PENS;
.....
```

3.6 REFERENCE_DESIGNATOR

The **REFERENCE_DESIGNATOR** keyword is used to identify an object inside the section.
It is used in the **NET_LIST** and **PHISYCAL_LAYOUT** sections; it identifies the Drawing Reference of the component.

Example of **REFERENCE_DESIGNATOR** keyword:

```
.....
!  NET_LIST
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9562270';
!  !  @  !..FIN_COMPONENT[ 1 ];
.....
!  !  @..FIN_COMPONENTS;
.....
!  PHYSICAL_LAYOUT
!  !  COMPONENTS
!  !  @  COMPONENT[ 1 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  ROTATION                :=    270;
!  !  @  !  LOCATION                := ( 24.12992, 39.89904 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
.....
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 1 ];
.....
```

3.7 LAYER

The **LAYER** keyword is used to identify a layer of the board.
The meaning of the keyword changes depending on the section where it is used:

- In the **COMPONENT** and **TEST_POINT** section it identifies the Mount side
- In the **TRACES** section it identifies the layer of the track

Example of **LAYER** keyword:

```
.....
!  !  @  COMPONENT[ 5 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  ROTATION                :=    180;
!  !  @  !  LOCATION                := ( 18.30911, 4.974151 );
.....
!  !  @  !..FIN_COMPONENT[ 5 ];
.....
!  !  @  !  TRACES
!  !  @  !  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  !  LINE_PEN                := 7;
!  !  @  !  !  LINE                    := [ (  3.704155,  37.93016 ),
!  !  @  !  !                              (  3.704155,  38.84071 ) ];
!  !  @  !  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  !  LINE                    := [ (  3.704155,  37.93016 ),
!  !  @  !  !                              (  3.704155,  39.05238 ) ];
!  !  @  !  !..FIN_TRACES;
!  !  @  !  TEST_POINTS
!  !  @  !  !  TEST_POINT[ 1 ]
!  !  @  !  !  @  LAYER                   := SOLDER_SIDE;
.....
!  !  @  !  !  @..FIN_TEST_POINT[ 1 ];
.....
```

3.8 ROTATION

The **ROTATION** keyword describes the rotation of the component.

Example of **ROTATION** keyword:

```
.....
!  !  @  COMPONENT[ 5 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'R1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := COMPONENT_SIDE;
!  !  @  !  ROTATION                :=    180;
!  !  @  !  LOCATION                := ( 18.30911, 4.974151 );
.....
!  !  @  !..FIN_COMPONENT[ 5 ];
.....
```

3.9 X_MIRROR

The **X_MIRROR** keyword indicates if the component is mirrored respect to X axes.

Example of **X_MIRROR** keyword:

```
.....
!  !  @  COMPONENT[ 4 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'IC1';
!  !  @  !  PLACED;
!  !  @  !  LAYER                   := SOLDER_SIDE;
!  !  @  !  X_MIRROR;
.....
!  !  @  !..FIN_COMPONENT[ 4 ];
.....
```

3.10 CONTACTED

The **CONTACTED** keyword indicates the access side for each pins of the component.

Example of **CONTACTED** keyword:

```
.....
!  !  @  !  !  @  PIN[ 1 ]
.....
!  !  @  !  !  @  !  CONTACTED              := [ COMPONENT_SIDE ];
.....
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
.....
!  !  @  !  !  @  !  CONTACTED              := [ COMPONENT_SIDE ];
.....
!  !  @  !  !  @..FIN_PINS;
.....
```

3.11 LOCATION

The **LOCATION** keyword is used in most sections and it identifies the coordinates of the object.
The meaning of the keyword changes depending on the section where it is used:

- In the **COMPONENT** section it identifies the barycentre of the component
- In the **PIN** section it identifies the coordinates of the pin as to the barycentre of the component
- In the **TEST_POINT** section it identifies the position of the test point
- In the **VIA** section it identifies the position of the via

Example of **LOCATION** keyword:

```
.....
!  PHYSICAL_LAYOUT
!  !  COMPONENTS
!  !  @  COMPONENT[ 5 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'R1';
.....
!  !  @  !  LOCATION               := ( 18.30911, 4.974151 );
!  !  @  !  PACKAGE
!  !  @  !  !  PINS
!  !  @  !  !  @  PIN[ 1 ]
.....
!  !  @  !  !  @  !  LOCATION               := ( 0, 0 );
.....
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
.....
!  !  @  !  !  @  !  LOCATION               := ( 1.6, 0 );
.....
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  @..FIN_PINS;
!  !  @  !  !..FIN_PACKAGE;
!  !  @  !..FIN_COMPONENT[ 5 ];
.....
!  !  @..FIN_COMPONENTS;
!  !  SIGNALS
!  !  @  SIGNAL[ 1 ]
.....
!  !  @  !  TEST_POINTS
!  !  @  !  !  TEST_POINT[ 1 ]
.....
!  !  @  !  !  @  LOCATION               := ( 3.598322, 37.46488 );
.....
!  !  @  !  !  @..FIN_TEST_POINT[ 1 ];
!  !  @  !  !..FIN_TEST_POINTS;
!  !  @  !  VIAS
!  !  @  !  !  VIA[ 1 ]
.....
!  !  @  !  !  @  LOCATION               := ( 3.704155, 36.40655 );
.....
!  !  @  !  !  @..FIN_VIA[ 1 ];
!  !  @  !  !..FIN_VIAS;
!  !  @  !..FIN_SIGNAL[ 1 ];
!  !  @..FIN_SIGNALS;
!  !..FIN_PHYSICAL_LAYOUT;
.....
```

3.12 PADSTACK_ID

The **PADSTACK_ID** keyword indicates the type of pad each pins of the component.

Example of **PADSTACK_ID** keyword:

```
!  !  @  !  !  @  PIN[ 1 ]
!  !  @  !  !  @  !  PADSTACK_ID              := 'p0610a';
!  !  @  !  !  @  !..FIN_PIN[ 1 ];
!  !  @  !  !  @  PIN[ 2 ]
!  !  @  !  !  @  !  PADSTACK_ID              := 'p0610a';
!  !  @  !  !  @  !..FIN_PIN[ 2 ];
!  !  @  !  !  TEST_POINT[ 1 ]
!  !  @  !  !  @  PADSTACK_ID              := 'tptdef';
!  !  @  !  !  @..FIN_TEST_POINT[ 1 ];
!  !  @  !  !  VIA[ 1 ]
!  !  @  !  !  @  PADSTACK_ID              := 'viadef';
!  !  @  !  !  @..FIN_VIA[ 1 ];
```

3.13 LINE_PEN

The **LINE_PEN** keyword describes the pen that it is used for drawing the current line.

Example of **LINE_PEN** keyword:

```
!  !  @  !  TRACES
!  !  @  !  !  LINE_PEN                 := 7;
!  !  @  !  !..FIN_TRACES;
```

3.14 LINE

The **LINE** keyword describes the start coordinates and the end coordinates of the line.

Example of **LINE** keyword:

```
!  !  @  !  TRACES
!  !  @  !  !  LINE                     := [ (  3.704155,  37.93016 ),
!  !  @  !  !                               (  3.704155,  38.84071 ) ];
!  !  @  !  !..FIN_TRACES;
```

3.15 POLYGON

The **POLYGON** keyword describes the start coordinates and the end coordinates of a series of lines.

Example of **POLYGON** keyword:

```
!  !  @  !  TRACES
!  !  @  !  !  POLYGON                 := [ (  9.207471,  38.52321 ),
!  !  @  !  !                             (  14.18162,  38.52321 ),
!  !  @  !  !                             (  14.60495,  38.09988 ) ];
!  !  @  !  !..FIN_TRACES;
```

3.16 LIBRARY_IDENTIFIER

The **LIBRARY_IDENTIFIER** keyword describes the identifier of the Part Number, it is linked with the label **IDENTIFIER** in the **COMPONENTS > COMPONENT [n]** section.

Example of **LIBRARY_IDENTIFIER** keyword:

```
!  !  @  COMPONENT[ 1 ]
!  !  @  !  REFERENCE_DESIGNATOR    := 'C1';
!  !  @  !  TYPE                    := NORMAL_COMPONENT;
!  !  @  !  LIBRARY_IDENTIFIER      := '9572813';
!  !  @  !..FIN_COMPONENT[ 1 ];
```

3.17 UNCONNECTED

The **UNCONNECTED** keyword indicates which pins are unconnected.

Example of **UNCONNECTED** keyword:

```
!  !  @  NET[ 4 ]
!  !  @  !  UNCONNECTED;
!  !  @  !..FIN_NET[ 4 ];
```

3.18 PIN_ID

The **PIN_ID** keyword indicates the Pin Name.

Example of **PIN_ID** keyword:

```
!  !  @  !  PINS
!  !  @  !  !  PIN[ 1 ]
!  !  @  !  !  @  PIN_ID                 := '1';
!  !  @  !  !  PIN[ 2 ]
!  !  @  !  !  @  PIN_ID                 := '1';
!  !  @  !  !  @..FIN_PIN[ 2 ];
!  !  @  !  !..FIN_PINS;
```

4. Import setting

4.1 Pin function assignment

This assignment table must be filled, in order to correctly execute the CAD file import.

In order to correctly test some polarized devices such as diodes, bipolar transistors, etc., it is basic to identify correctly the pin function (i.e. anode, base, etc.) of each pin.

The fields contained in the table, are described below:

Field	Description
Device Type	Identifies the type of device (example: Resistors, Capacitors, Digital Devices, Diodes etc.).
Pin Function	Function concerning the Pin.
Pin Name	Pin reference.
Cad Pin	Pin reference in Cad file.

4.2 Drawing ref. initials/device type assignment

The THEDA file typically contains all information about the devices, such as value, tolerances and type; which are fundamental from the point of view of the test program generation.

The fields contained in the table are described below:

Field	Description
Drawing Reference	Initial letter identifying the **Device Type**.
Device Type	Identifies the type of device (example: Resistors, Capacitors, Digital Devices, Diodes etc.).
Default Tol+, Tol-	Value and tolerance of the device only if required (as for resistors).

I It could happen that in the CAD file they are missing. For each drawing reference initial, the displayed table enables to define the following data default values:

- Device type
- Default positive tolerance
- Default negative tolerance

This means that if, for any reason, the CAD file does not contain the information mentioned above, the default values will be used.

A. Note about the THEDA ASCII text file format

The THEDA CAD-CAE typically runs under Unix operating system and generates its neutral ASCII output file in Unix format.

The Unix ASCII text files use the "$0a_{hex}$" ASCII character as end of line identifier.
The Windows® (MS-DOS) operating system uses the ASCII "$0d_{hex}$" and "$0a_{hex}$" characters as end of line identifier for ASCII text files.
This means that output ASCII text files may require an ASCII format conversion (from Unix to Windows® format).

This operation can be performed using "WordPad", a standard text file editor.
Open the THEDA ASCII file with this editor and save it, this operation will automatically perform the conversion from ASCII Unix format to ASCII Windows® format.