VisualPlace

Process Production Files

for Manual and Automated Assembly
of Printed Circuit Boards

version 3.3 — May 2024

CompuPhase

Acknowledgements

The ‘Haru PDF Library’ (for generating the PDF reports) is copyright Takeshi Kanno. (http://www.libharu.org/)

The JSON parser in VisualPlace is ‘Parson’ by Krzysztof Gabis.

The IP camera support (for a zoomed real-time preview of a PCB during manual assembly) uses the libjpeg-turbo library. (http://www.libjpeg-turbo.org/)

SVG support is based on the ‘nanosvg’ project by Mikko Mononen.

VisualPlace includes code from the zlib and libpng projects.

Copyright & license

Typeset with TEX in the ‘Adobe Source’ typeface family.

Contents

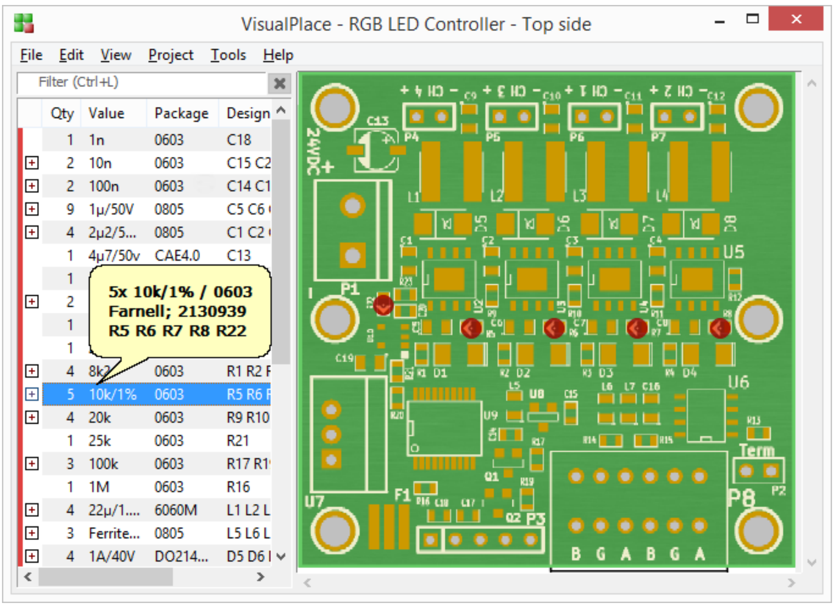
Introduction . . . 1
What does it do? . . . 2
The benefits of VisualPlace . . . 2
Preparing a VisualPlace project . . . 7
Finish the board, set its origin . . . 7
Create the PCB design files . . . 7
Create the fabrication files . . . 9
Create the VisualPlace project . . . 9
Creating a project from scratch . . . 13
Working with VisualPlace . . . 16
Modifying a VisualPlace project . . . 20
Configuration . . . 21
User fields . . . 24
Assembly stages . . . 25
PCB and marker visualisation . . . 26
Adding or correcting a position . . . 27
Adjusting the Bill of Materials . . . 29
Adding fiducials . . . 32
Adding notes . . . 33
Manual population of a PCB . . . 34
Adjusting placement order . . . 35
Grouping, expanding and collapsing tree rows . . . 36
Ticking components off . . . 37
PCB zones . . . 37
Keyboard navigation . . . 39
Live camera view . . . 41
Preparing for automated population . . . 43
Verifying the component placement list . . . 43
Assign parts to stages . . . 48
Convert/export the component placement list . . . 48
Generating reports . . . 50
Coordinates and origins . . . 53
Miscellaneous operations . . . 53
Maintaining VisualPlace tables . . . 55
Package/footprint names: vendor vs normative . . . 55
Creating or modifying a footprint lay-out . . . 57
Adding a component placement correction . . . 59

Appendices 61
A: Application interface 61
B: Normal orientations 64
C: EDA suite notes 68
D: Camera support 81
E: Scanning and recognition 86
F: Using Version Control 89
G: Menu interface 93
H: Keyboard and mouse 100
I: Plug-in interface 105
J: License 113
Index 114

Introduction

VisualPlace is a computer program for creating and managing 'production files' for PCB assembly. Intended use of VisualPlace is in the phase after the PCB design and before (machine) assembly of the board —or *during* assembly in the case of manual assembly. VisualPlace can also be used for inspection and trouble-shooting. VisualPlace has a plug-in architecture for reading the data files, so that it can easily be adapted to various 'electronic design automation' (EDA) suites and pick-&-place machines.

See also appendix C for application notes for various EDA suites.



The design of VisualPlace is based on three main goals:

- ◇ Before sending the production data to an EMS company for assembly, let PCB designers match up the output files to verify that the production data is consistent and complete. Corrections and additions to the production data can be made from within VisualPlace.
- ◇ Enable EMS engineers to verify and 'trouble-shoot' the received placement files and package/footprint information, *before* loading them in the pick-&-place machine —and thereby minimizing the 'off line' time of the pick-&-place machines.
- ◇ Visualize the component placement during manual assembly of prototypes or small series, and optionally visualize the PCB as a live stream from a camera with the placement information overlaid.

For a more detailed explanation of these points, see the section 'The benefits of VisualPlace' later in this chapter.

What does it do?

VisualPlace combines the production files created by an 'electronic design automation' (EDA) suite with silk-screen images (generated from the Gerber files). It then lists the components for the PCB from the 'bill of materials'.

When you select a component, or a group of components, VisualPlace shows the placement and orientation of the component (or group) on the board. This allows you to verify the placement of the components, including their rotation, by comparing the placement with the PCB's silk-screen(s).

VisualPlace has tools to verify whether the packages for components in the bill-of-materials match the footprints on the Gerber files, and tools to scan the designator labels from the silk-screen and map these to nearby footprint outlines.

The format of the production files generated by an EDA suite is different (even if ever so slightly) than what most pick-&-place software accepts by default. VisualPlace allows to export the production files to a machine-specific format or to a common format with a configurable lay-out. A simple 'BoM' editor allows you to modify an existing bill-of-materials, or create one from the component placement list.

The benefits of VisualPlace

One of the goals in PCB production and assembly is an efficient work flow where 'double work' is minimized, mistakes are avoided and machine utilization is maximized. The current process, however, leaves to be desired in this area.

PCBs are created and verified on a computer, using various programs (from an EDA suite). The production files are sent off to a PCB manufacturer (or the PCB manufacturing *department*); these are usually 'Gerber' files. When the PCBs arrive back, they are sent, together with all components and (more) production files to an 'EMS' company (Electronic Manufacturing Services) for assembly and soldering.

Unlike PCB production itself, where Gerber photo-plotter files and Excellon drill files are the *de facto* standard, no standard is predominant for the pick-&-place machines used in the assembly phase. Virtually all EDA software can generate a file with the positions and orientations of the components of a board, and virtually all pick-&-place machines have a computer interface, yet these rarely match. This leads to 'requirements' like converting and fine-tuning the incoming production files (*inefficient*, especially if manual intervention is required, which usually is the case), or even partially or fully (re-)generating the position/orientation data from scratch (*double work*, because the component position data is already in a computer-readable format). It gets worse: all this time, the pick-&-place machine sits idle, and manual intervention/regeneration carries a risk for errors.

When prototypes or small series are made, it is sometimes not cost-effective to program the pick-&-place machine for it —also due to the problems described above. The boards are then assembled by hand. However, the technician doing so basically has to work from an (enlarged) copy of the 'silk-screen' image to locate each component. As a result, a significant amount of time is spent in searching each component on the board. This, too, is inefficient, and it is error prone, because the silk-screen image may be ambiguous.[1]

Benefits for the PCB designer

It is not uncommon for a PCB assembly firm to ask the PCB designers for the silk-screen artwork and the bill-of-materials, and then proceed to build the component placement data from scratch —even though, as stated before, nearly every EDA suite can generate these files. One reason why PCB assembly firms disregard the 'generated' production files from the EDA suite is that these production files are often incomplete and inaccurate. Since it is unknown *where* the production file is accurate and *where* it is *not*, the assembly facility (EMS company) may prefer to ignore it altogether and rebuild it from scratch.

That merits the question: why are production files from an EDA suite so often inaccurate (and incomplete)? In part, the answer is that, while generating the production files is just a 'click on the button', creating good and complete position data can never be completely automated. EDA suites lack a program to let the designer verify the generated data, however, and

[1] When a PCB lay-out is reviewed, engineers typically give lower priority to the silk-screen, since it does not influence the functioning of the board. The 'design rules check' in EDA suites does not verify the accuracy of the silk-screen, either. Therefore, errors in the silk-screen may well slip through the net.

therefore, the PCB designer has no way to know what shape the placement data is in, letting errors in the data go undetected.[2]

With VisualPlace, however, the designer *does* have a tool to verify the production data. If the designer can tell the PCB assembly firm with confidence that the data is accurate, there is one less reason for the assembly firm to waste time in rebuilding it from scratch.

Benefits for automated PCB assembly

Like the PCB designer, an engineer who needs to program the pick-&-place machine can run VisualPlace to verify the quality of the provided production files. If there are errors in the data, or if the data is incomplete, the engineer can take a note of these 'exceptions', or choose to correct them inside VisualPlace. There is no more need to throw the entire file away because there is (or *may be*) an error 'somewhere' in there.

A PCB assembly firm will receive production data in various formats — there is no standard file format, neither *de facto* nor formal, for the component placement. EDA suites use proprietary formats, which sometimes change between releases. Due to a plug-in architecture, VisualPlace detects and supports a variety of file formats for the production data, without needing manual intervention. VisualPlace can also export the data, in either the format specific to the pick-&-place machine or in a configurable CSV format. When all incoming production files are first fed through VisualPlace, the pick-&-place machine needs only to be configured to accept a single format (which may be its native format).

If, apart from the Gerber files for PCB manufacturing, no production data is available, VisualPlace offers assistance to create the required data. A simple, but full-featured, table editor allows you to fill in the values/types and packages for all components. With that list, VisualPlace can then scan the Gerber files for designator labels and map the components to a detected footprint shape that is near a detected label.

Even if the placement instructions must be created or corrected *on the pick-&-place machine itself*, the final program can be read back into VisualPlace —provided that a suitable plug-in is available. This allows for further analysis of the program or the adjustments, and it allows you to let the customer validate the final pick-&-place program before running it.

[2] A 'chicken-&-egg' situation: there is little incentive for the producers of EDA suites to put proper attention to the lay-out, accuracy and completeness of the production files —as many assembly firms ignore them and build their own, but the *reason* that assembly firms are indifferent to these files is *because of* their perceived low quality.

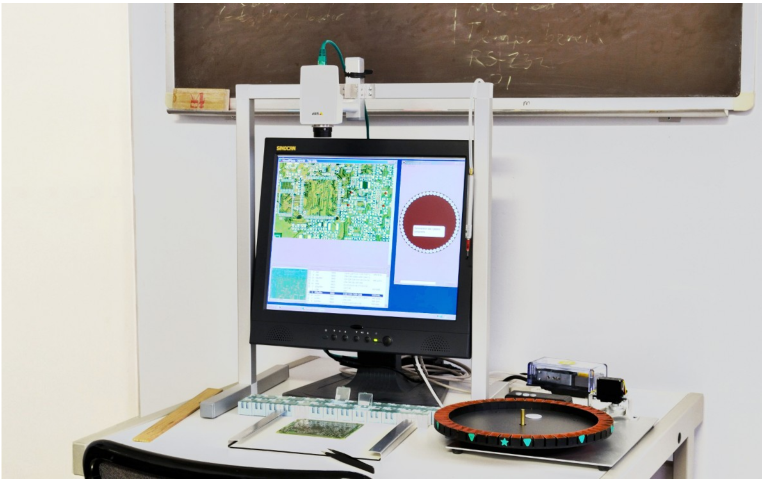


FIGURE 1: *Camera-assisted manual PCB assembly, using VisualPlace*

When preparing a project on a machine, reels and tubes with components must be mounted in the feeders. To do so, positions on the bill-of-materials must be made to match a feeder number. By supporting a bar-code scanner, VisualPlace can avoid reading errors and manual entry errors; scanning a reel on the machine will tell you instantly whether an exact match exists —and if there is no exact match, there might be a problem.

The first benefit for PCB assembly firms is reduction of work, because the component positions do not have to be (re-)generated from scratch. The second benefit is that corrections in component placements that are specific to an EDA suite are automatically applied. Thirdly, the verification, correction and conversion of the production files occurs *off-line*. The pick-&-place machine can continue to run while a (next) project is being prepared, maximizing the machine utilization.

Benefits for manual PCB assembly

The quality of soldered connections depends predominantly on the dosing of the solder paste —position and amount. This is conveniently and accurately realised by using a stencil, but equally easily ruined when a component is not placed correctly on the board (and in the paste) in one go. When correcting the position of a component, you risk to smear the paste off the pads.

Therefore, manual population of a PCB is a job that requires concentration —and a steady hand. Locating the next component to place, by skimming through the bill-of-materials and (enlarged) print-outs of the silk-screen images, then, is a 'distraction' to say the least. VisualPlace helps you during manual assembly of a board, by allowing you to walk through the bill-of-materials, read how many parts of a particular value and footprint are needed, and see in a glance where they are located on the PCB.

A small, but important, detail is that VisualPlace shows you the orientation of the component on a board, which is especially relevant for LEDs and diodes, because the silk-screen may be ambiguous on the required orientation.

VisualPlace is designed to be be highly configurable in the visualization, and to be operated with a minimum of key strokes —possibly even hands-free, with its support for USB foot pedals. Alternatively, a button bar allows you to easily walk through the component list, and zoom in on a particular section of the PCB, on any system equipped with a touch screen.

Switching your focus back & forth between the computer display and the PCB, can become a strain if done for hours at length. When the workstation is fitted with a suitable camera, VisualPlace can show a live stream of the camera, overlaid with the placement information. With such a set-up with the camera floating above the PCB, VisualPlace functions simultaneously as a magnifier and as a component location indicator.

Preparing a VisualPlace project

VisualPlace requires a set of images and 'production files' for its operation. *Which* files you need, depends on the board and on the EDA suite that you use. As an overview: these are the steps in creating the VisualPlace project:

- ◇ Finish and review the board. Set the origin for the plot files and the production files.
- ◇ Plot Gerber files for the silk-screens and the board profile, the solder masks, copper layers and drill file.
- ◇ Generate the component placement list(s) and the bill-of-materials.
- ◇ Launch VisualPlace and create a new project.

The next sections give more detail about each step.

Finish the board, set its origin

VisualPlace needs several files created by the EDA suite, specifically files from the PCB lay-out program. Hence, the PCB design should to be complete before moving on.

VisualPlace aligns the component positions with the silk-screen images using the *extended attributes* of the Gerber X2 format, if available. In absence of extended attributes, VisualPlace assumes that the component positions are relative to either in the upper left corner or in the lower left corner of the board profile. Depending on the EDA suite, it may be convenient to set the PCB origin to either of these corners. Some EDA suites work best when you set the origin in the upper left corner, other work best when the origin is in the lower left corner —see appendix C for specific notes on your EDA suite, or the manual for your EDA suite.

If setting the origin of the PCB is impossible or inconvenient, you can manually align the component positions to the silk-screens after creating the project —see page 12. The above step is therefore not mandatory.

Create the PCB design files

VisualPlace displays the silk-screen(s) and other design files of the PCB. You only need the design files of the sides on which there are components, so one silk-screen image if the board only has components on the top side, and two images if the board has components on both sides.

The preferred file format for the silk-screen (and other) images is the Extended Gerber format (formally the 'RS-274X' format). In particular, if your EDA suite supports it, it is recommended that you create design files in the Gerber X2 format. This latest version of the Gerber file format allows extra features for matching footprints with component packages as specified in the production files. VisualPlace also uses the component information in Gerber X2 files to automatically adjust alignment and component placement.

Next to Gerber, VisualPlace supports bitmap images in the TIFF, PNG, JPEG and BMP file formats for the PCB image.[1] It allows you to project the production data on a scan of the actual PCB. When using a scan, note that VisualPlace needs to know the resolution of the image, which is normally stored in 'dots per inch' (dpi). The best resolution to use depends on the board itself. As a rule, we suggest that the PCB image has a resolution in the range of 300 dpi to 600 dpi.

It is recommended to also create a Gerber file with the PCB profile (this is the drawing of the board outline or contour). Doing so enables VisualPlace to align the component placement lists to the PCB artwork —in particular when the drawings on the silk-screen exceed the PCB edges. Note that VisualPlace can only handle the PCB profile file when the silk-screen images are in Gerber format too.

If your PCB has a cut-out, this must be drawn in the PCB profile file. If the cut-out is in a different layer, VisualPlace will not visualize the cut-out. A routed slot in the PCB can also be in the drill file.

While optional, we recommend that you also add the Gerber files for the solder mask and for the copper layers. These extra Gerber files improve both the visualization of the PCB (which is especially useful when doing manual assembly) and the accuracy of footprint scanning (which is useful for preparing machine assembly).

Even when you plan to use a camera for the PCB assembly, the Gerber files for the silk-screen and the PCB profile are required. VisualPlace uses them to determine the PCB dimensions to properly align the camera image to the component placement data, and also shows the Gerber data in the overview window. Furthermore, corrections and additions to the placement data can only be made on the design files —not on the camera image.

---

[1] Some features, like automatic origin alignment and footprint scanning may not work on bitmap images.

A drill file is useful with camera-assisted assembly of large boards, see the section on enabling the live camera view on page 41 and the section on PCB zones on page 37. VisualPlace supports Excellon and ‘Sieb & Meyer’ drill files, as well as drill files in the Gerber X2 format. The origin of the drill files must be aligned with that of the Gerber design files.

Create the fabrication files

VisualPlace requires a ‘component placement list’ (CPL), which also goes under the names ‘pick&place file’, ‘centroid’ file, ‘XYRS report’ or ‘modules position file’. This file has the positions and orientation of (the centres of) the components. The step-by-step procedure to generate these ‘production files’ depends on the EDA suite, see the manual for your EDA suite and appendix C for details.

The component placement list may already contain all data that VisualPlace needs. In other cases, you can generate a bill-of-materials file (BoM file) to complete the data. Whether a bill-of-materials is required, and how to generate it, depends on your EDA suite, see appendix C.

Create the VisualPlace project

When you have the Gerber files/board images and the fabrication files, you are ready to create the project. In VisualPlace, create a new project by opening the dialog under File / New project. . .

The dialog uses three steps. The first two steps are shown in figure 2, the last in figure 3. The first step is to select the path to the production files, the name of the project, and whether the PCB has components on both sides or only a single side.

The second step show a list of production files that have been found that the configured path. You may select (or deselect) files to include in the project.

All information is then listed in the Project settings dialog —the third step (figure 3). This dialog has dynamic fields. Depending on your choices, not all settings may be visible. On the top of the dialog, you can choose whether the PCB has components on a single side or on both sides. For PCBs with only components on one side, the dialog is simplified.

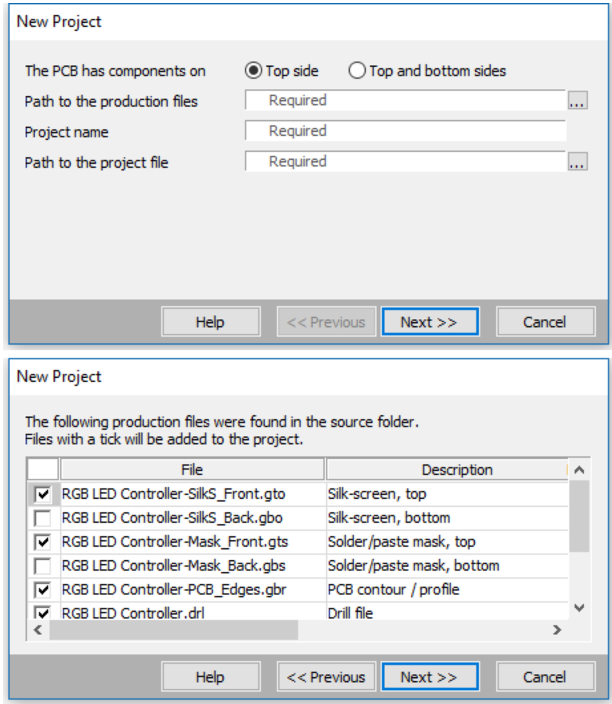


FIGURE 2: *New Project assistant*

The Project settings dialog is mainly split in two sections:

- PCB design files (Gerber files): in the first section, you specify the Gerber files for the PCBs. In the very least, a file for the silk-screen should be selected, but it is recommended to select extra Gerber files to improve the visualization and functionality.
- Component Placement Lists & Bill of Materials: in the second section, you need to specify the files with the placement data. Depending on the comprehensiveness of the CPL file, a bill-of-materials may be optional. When specified, it is used to complement the data of the component placement lists.

For the Silk-screen section, you can specify files for both sides of the PCB. The PCB profile field is for the filename of the Gerber file of the PCB outline (or 'contour') drawing. This file is optional, but recommended.

The PCB profile drawing is used for two purposes by VisualPlace: merging it with the silk-screen images in the case that the silk-screens *exclude* the PCB profile, and aligning the silk-screen images to the board edges (in

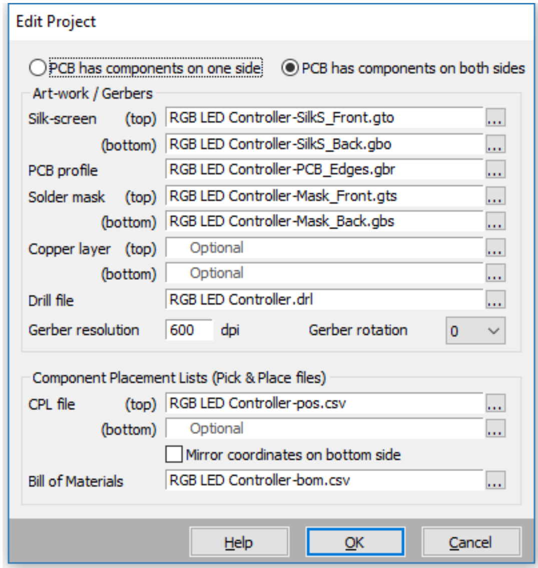


FIGURE 3: *Project settings*

case that drawings on the silk-screen exceed the PCB edges). The pen size, in turn, is used to automatically align the Gerber files with the data in the component placement lists.

The Gerber file for the solder mask (or the solder paste mask) is useful when the silk screen on its own is insufficient to clearly indicate the positions and orientations of all components. In addition, this file improves the accuracy in component placement verification by scanning the Gerber data.

The drill file is especially useful when using a camera for manual assembly.

Note that VisualPlace also supports *bitmap* images for the silk-screen images, as discussed in the section 'Silk-screen image' earlier in this chapter (page 7). When using bitmap images, the field for the PCB profile is ignored.

The section for the Component Placement Lists also has separate fields for the top and bottom sides of the PCB —because many EDA suites store the component placement data in separate files for the top and bottom layers. If the project has only components on the top side, you need to only select the silk-screen and the component placement list for the top side.

When a *single* component placement list contains the component placement for both sides, select the file for the 'top side' and leave the field for the 'bottom side' empty.

The filename for the bill-of-materials is optional, but in most cases it is recommended to fill it in. VisualPlace uses the bill-of-materials to complement the data that it reads from the component placement lists, as well as to get a list of components that lack placement data (i.e. components that are *absent* from the component placement lists).

The resolution of the Gerber file image may be chosen as well. A higher resolution allows to zoom in further, but at the cost of requiring more memory and more processing. In rare cases, the Gerber files may be rotated relative to the component placement lists. This can optionally be adjusted in the Project settings dialog as well.

In most cases, VisualPlace can determine how to align the placement data to the artwork from the Gerber files. However, if your artwork was based on bitmap images instead of Gerber files, or if you did not specify a Gerber for the PCB profile, automatic alignment may fail. In these cases you may need to adjust the settings manually, see Aligning the origin, below.

Aligning the origin

Both the silk-screen image and the component placement lists are based on an origin. When following the directions in appendix C for your EDA suite, the origins of all production files should be aligned. One of the features of the Gerber X2 format is that it adds component properties to the file. These properties can be matched to the data in the component placement list (CPL) file, and the correct alignment can be automatically established from that. Note that you should add a Gerber file for the solder mask and/or the copper layer to the project —the silk-screen artwork does often not contain the required component attributes.

When automatic alignment is not an option, you may visually align the component placement origin relative to the silk screen. For manual alignment, first load the project. Select a component from the component list that is easily identifiable on the silk-screen (see figure 5 on page 16). Then, from the Project menu, select Align placement origin. . . You are now prompted to click at the centre of that component's drawing on the silk-screen. When the origin is set, the project file is adjusted and the project is reloaded.

For best accuracy, pick a small component for this procedure. It is easier to locate the centre of small components than it is for large components.

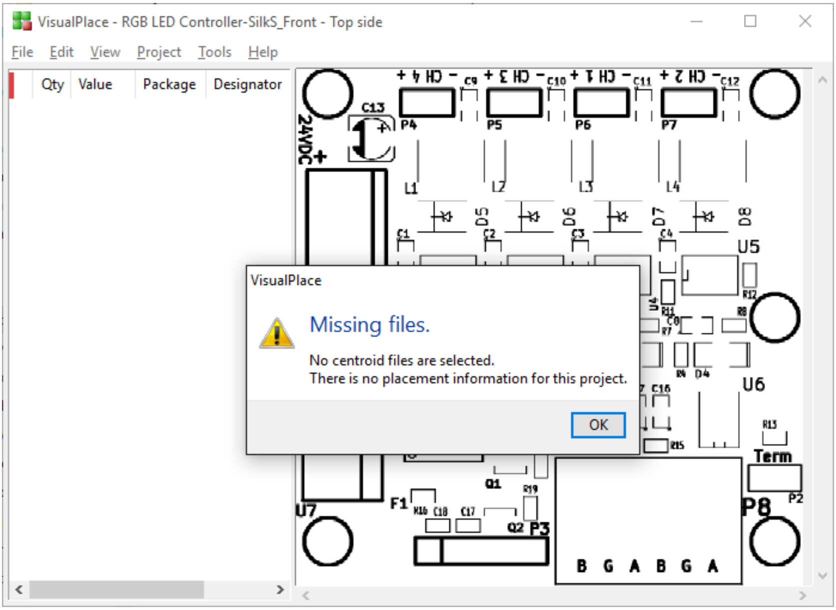


FIGURE 4: *A project created with only a silk-screen Gerber*

Creating a project from scratch

The previous sections assume that all required data is available: the Gerber lay-outs, an optional drill file, component placement lists in a supported format, a bill-of-materials in electronic format, etcetera. VisualPlace can also create a project if one or more of these files are lacking, though with more effort.

When *only* the Gerber file of the silk-screen is set in the project, VisualPlace will show it like in figure 4. The image is in black-&-white because VisualPlace lacks information on the PCB profile (the contour or outline of the PCB). The component list is empty, and VisualPlace also warns that it does not have any placement information (which is obvious with an empty component list).

The first step is to create a component list, by scanning text from the silk-screen image. To start a scan, select Tools / Scan labels from silk-screen from the menu. See appendix E for details about the scanning and text recognition process.

After a scan completes, VisualPlace presents you with a list of new labels. New labels are texts that are recognized on the PCB, but that are not

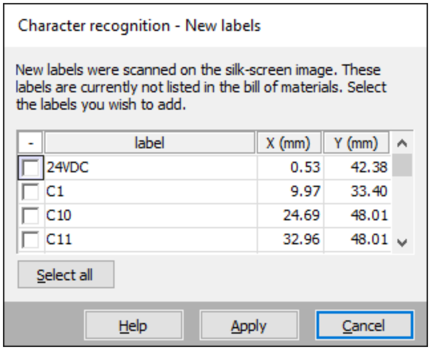


FIGURE 5: *Texts recognized from the silk-screen image*

known as reference designators. In the case that you started a project without having a bill-of-materials, by far most of these new labels will be reference designators, which you will want to include in the bill-of-materials. Therefore, in the New labels dialog, see figure 5, you will typically first click on the button Select all. After that, run through the list to deselect any texts that are *not* designators, before clicking Apply.

After adding the new designators to the bill-of-materials (by clicking Apply in the New labels dialog), the bill-of-materials editor opens. It presents us with an empty table, except for the designators. This table must be filled in, from a bill-of-materials that you have received on paper, or from the schematic, or by some other means. Without the bill-of-materials, you cannot populate the PCB.

It is common to have many components with the same value and package on a PCB. For example, a 100nF 0603 capacitor may well occur up to 50 times on a medium-sized PCB. The editor dialog allows you to fill in the information (possibly including the order number) for one designator, and then to copy those fields to a list of other designators. To use this function, pop-up the context menu on the relevant row (with a right-click of the mouse or with the 'menu'-key) and select Copy fields to other designators... You can then enter a list of designators to copy the fields to. The designators in this list may be separated with space characters or commas. Note that this option is only available when the option One component per row (in the bottom-left corner of the dialog) is active.

The edit dialog for the bill-of-materials is covered in more detail in section Adjusting the Bill of Materials on page 29.

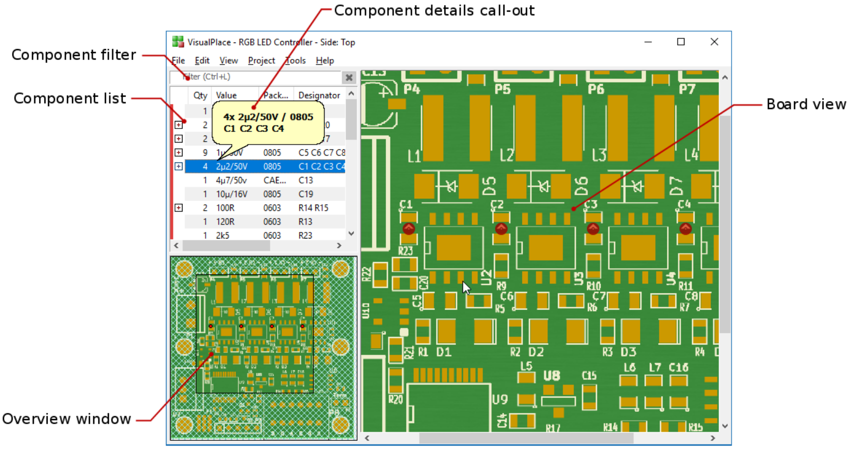
Now that there is a bill-of-materials, the next step is to add a placement position to each component: select Tools / Acquire placement data from scan. VisualPlace will now scan for footprint shapes on the silk-screen, determine the mid-point of these footprints, and match these to a nearby designator.

The results of footprint scanning should be taken as guidance. After the scan completes, you should run through the component list to verify the placements. The footprint scan does currently not detect the component orientation; while verifying the placements, you should assign the proper orientation to each component. For correcting a position (and for setting the orientation), see the section Adding or correcting a position on page 27.

A caveat in setting the orientation is that VisualPlace will only show an oriented centroid marker for *known* footprint shapes. For unknown footprints, VisualPlace shows the position of the component, but not its orientation. You need to add a definition of the footprint, see Mapping a package/footprint to a standard name on page 55.

Working with VisualPlace

After loading a project, the *component list* is filled with the data from the bill-of-materials and the top layer of the PCB is displayed.



Board view

The board view shows centroid markers on the selected parts. A 'centroid' refers to the centre of mass of a component —a pick-&-place machine picks up a component at its centroid, so that it is optimally balanced below its nozzle. For symmetric components the centroid is the centre of the shape, but for asymmetric shapes the centroid may be off the geometric centre.

The default colour for the centroid markers are red for the components on the top side and purple for the components on the bottom side. The colours are configurable, see page 26. The marker has an arrow that indicates the orientation of the component; the marker points towards pin 1 of the component —see also figure 21 (page 47).

Overview window

The overview window shows a scaled-down view of the PCB, and highlights the part that is visible in the board view window. The overview window is only present when the PCB does *not* fit in the board view. If the board view shows the complete PCB, the overview window disappears —see figure 6 on the next page.

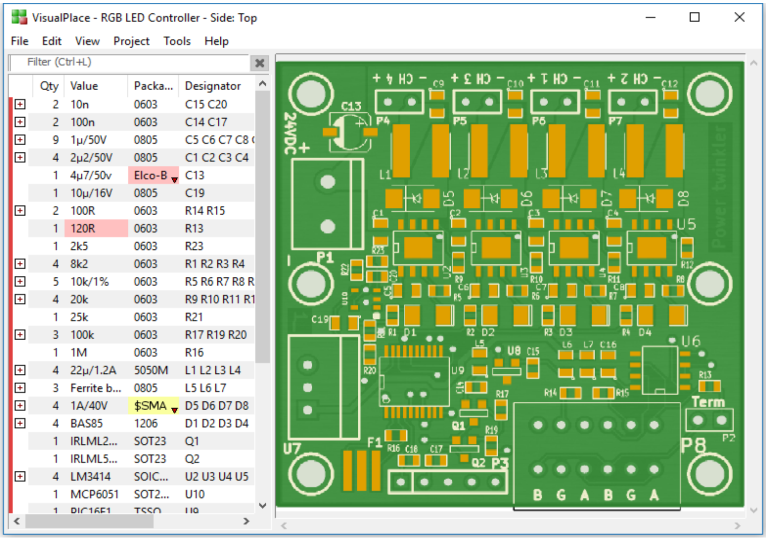


FIGURE 6: *Colours in the component list (left) signal potential problems*

If you click in the overview window, VisualPlace scrolls that position into view in the board view.

Component list

The component list combines information that is collected from the component placement list (CPL file) and the bill-of-materials. A coloured bar at the left of the component list indicates the side that the components are on: top or bottom. Through-hole components are typically sorted at the bottom of the list, and marked as such with a cyan bar in the left margin.

Fields in the component list can be marked with a red or yellow colour to mark special cases or inconsistencies.

• *Value*

A red background in the Value column for a part in the component list indicates a mismatch between the values in the component placement list versus the bill-of-materials. The value in the bill-of-materials takes precedence; the background is coloured to alert you that the production files disagree on the part's value. To correct a mismatch, you can either correct the original production files from the EDA suite, or edit the bill-

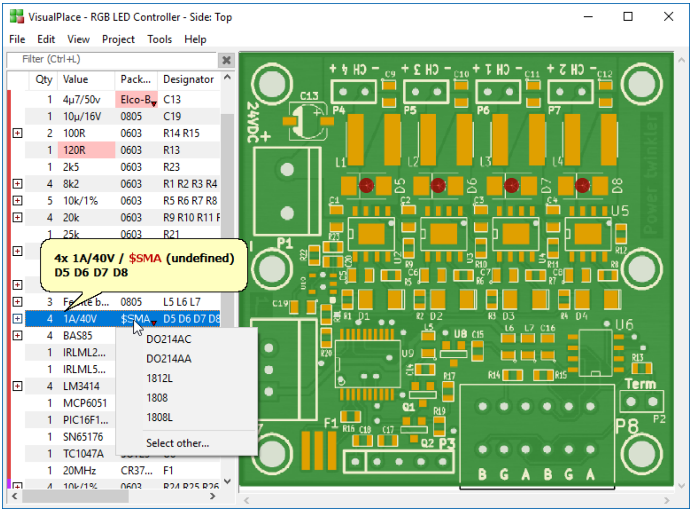


FIGURE 7: *Picking a standard package for an EDA-specific name*

of-materials from within VisualPlace —see section Adjusting the Bill of Materials on page 29.

• *Package*

A red background in the Package column for a part indicates a mismatch between the bill-of-materials versus the component placement lists. That is, the bill-of-materials lists a different package for the component than the placement list. In this case, the package name from the component placement list takes precedence. To correct this situation, correct the production files in the EDA suite or edit the bill-of-materials in VisualPlace (again, see section Adjusting the Bill of Materials).

A yellow background in the Package column for a part means that VisualPlace does not have a footprint lay-out for that part. VisualPlace requires a footprint lay-out both to achieve good accuracy in verifying/adjusting the component positions, and to properly display the orientation of the component. For a part that lacks a footprint definition, the centroid marker is still positioned correctly, but it does not have an arrow.

When the design files are in Gerber X2 format, VisualPlace collects the packages that fit on the pads in the Gerber design files. If it finds any, the field in the Package column has a downward triangle at the right. After clicking on that triangle, you can pick a standard package from a drop-down list. An alternative way to map a package to one of the standard packages, or to create a new footprint lay-out, use the 'package specifications' dialog —see section Mapping a package/footprint to a standard name on page 55.

• *Designators*

In the column for the designators in the component list, each row combines all parts that have the matching value and package. Components that are marked to be 'not mounted' are in the designator list as well, but these components are not included in the 'quantity' column.

A red background in the Designator column indicates that the components on that row are not mounted in the assembly stage that is currently set.

Component filter

Above the component list is an edit field in which you can type keywords. The component list is then filtered to contain only the compponents that match the keywords.

The keywords are matched against the component's value, package, reference, product number and any of the user fields. You can combine multiple keywords with AND and OR operators, or invert the match with the NOT operator. The default operator is OR, so if you type '0603 0805' in the filter (without the quotes), the component list is filtered to all components with 0603 *or* 0805 packages.

Wildcards are supported in the keywords, these are `*`. `?` and `#`. The wildcard `?` stands for *any character*, whereas `#` stands for *any digit*. The `*` stands for zero or more occurences of any character. There is a match if the keywords matches the first characters of the value/package/etc. That is, the keyword 'sot' matches SOT23, SOT189 and SOT353. However, the keyword 'qfp' does *not* match TQFP64 or LPQFP48; to match those, you should instead use the keyword '*qfp'.

In order to filter on a keyword that contains a space, you can enclose the key phrase in double quotes.

A special keyword is 'mount': it matches the mount status of the component. You can use it filter out all components that are marked as 'do not mount'. Alternatively, the term 'NOT mount' shows only the components that are not mounted.

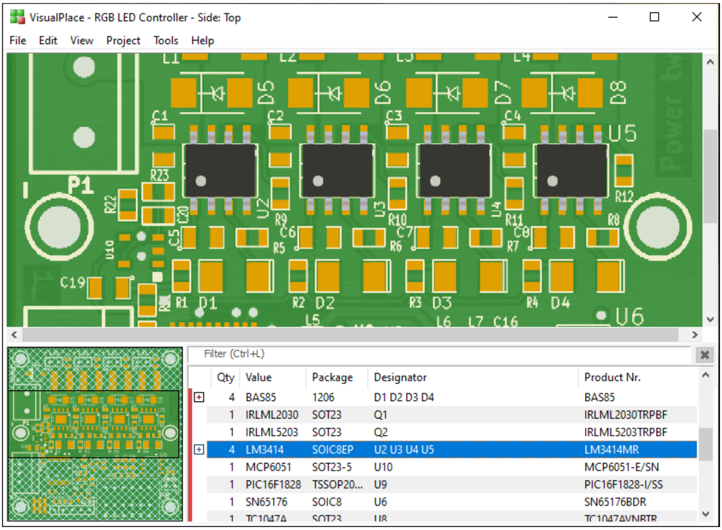


FIGURE 8: *Alternative visualization options*

Component details call-out

A call-out (or 'balloon') optionally shows the information of the selected row in a slightly larger font —this is especially useful for hand-assembly. Specifically, the call-out contains:

- the count of components (on this row) that must be placed,
- the value and package of the component,
- and the list of component references.

Press F3 again to make the call-out disappear. The call-out can also be shown/hidden through the menu. When selecting a part that must *not* be mounted or that has invalid or incomplete position information, the call-out appears automatically to notify this.

Modifying a VisualPlace project

If you need to change settings of a project after its creation, open the project dialog under Project / Project settings... This is the same dialog as the one presented on a new project (see page 9).

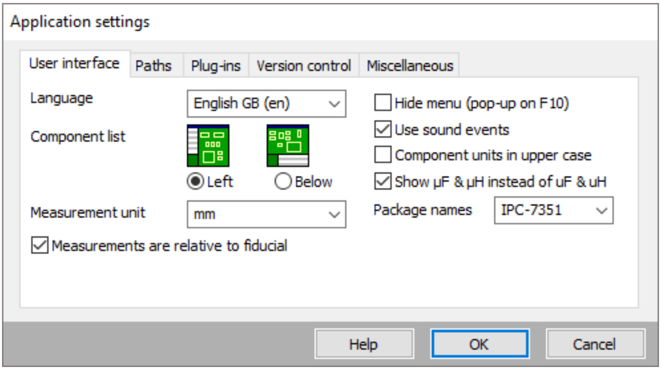


FIGURE 9: *Application settings*

Configuration

Several important settings for the user interface and the file formats and extensions are in the dialog for the application settings, under Tools / Application settings. . .

The options in this dialog are organized in four TAB-pages.

Language (TAB: User interface)
: The language for the user interface. The default is the same as the language that was selected during installation of the application, or English if the application was not installed.

Component list lay-out (TAB: User interface)
: In the common lay-out, the component list is at the left of the board view. An alternative lay-out has the component list below the board view. Compare figure 5 (page 16) versus figure 8 (the previous page).

Measurement unit (TAB: User interface)
: The unit for all measurements and positions, inches or millimetres.

Measurements are relative to fiducial (TAB: User interface)
: Coordinates can be displayed relative to the lower left corner of the PCB design files, or to the primary fiducial (if one exists). This option only changes the coordinates as they are displayed in the user-interface. When exporting a component placement list, the origin can be chosen separately in the export dialog.

Hide menu (TAB: User interface)
: When using VisualPlace for manual assembly, the menu is usually redundant. This option lets you hide the menu. The key F10 shows the

menu again.

Use sound events (TAB: User interface)
: VisualPlace can optionally make a sound when the selection changes. The sound files themselves can be configured in the Control Panel. There are separate sound events for when the selection has changed to a component with the same value and package as the previously selected component, and for when the value/package also changes. For adjusting the sound event assignments, use the Control Panel in Microsoft Windows ('Sounds and Audio Devices').

Component units in upper case (TAB: User interface)
: If set, this option will print units like 'pF' and 'nF' as 'PF' and 'NF'.

Show µF & µH instead of uF & uH (TAB: User interface)
: Uses the 'micro' symbol µ rather than the letter 'u' where micro is meant. This option also shows the Ω symbol rather than the letter 'R' for resistors below 1kΩ.

Package names (TAB: User interface)
: Various EDA suites use different names for the same package. For consistency, VisualPlace translates proprietary package names to a standardized or generic naming convention. This setting allows you to select your preferred naming for packages. See also section Package naming convention on page 56.

Path to data files (TAB: Paths)
: VisualPlace stores package specification tables and footprint correction tables in a set of local data files. When using VisualPlace on multiple workstations, you may want to share these data files between the workstations. See page 60 for a description of the options.

Path to project files (TAB: Paths)
: If this path is set, it is the default location that VisualPlace uses when opening projects or creating a new project. This path may contain the variables `%USERNAME%` and `%SYSTEMNAME%`, which represent the login name of the current user and the network name of the workstation respectively.

Package data file (TAB: Paths)
: The selected file with the design information for standard packages. If left empty, VisualPlace uses the file in the local data directory (of where VisualPlace is installed).

Plug-ins (TAB: Plug-ins)
: The functionality of VisualPlace can be extended with plug-ins. The available appear in this list. You can select the plug-ins you wish to

have active, by putting a tick mark at their left. By double-clicking on an entry, a configuration dialog for the plug-in will appear. See appendix I for developing your own plug-ins.

Version Control System (TAB: Version control)
: The choice of version control system. See appendix F for details.

Path to Version Control tools (TAB: Version control)
: The path to the utilities of the selected version control system. If the utilities are in the system path, VisualPlace fills in this field automatically.

Path to File Compare tool (TAB: Version control)
: The path to a file comparison utility. VisualPlace launches this tool when reviewing changes in the Update & commit dialog (see page 91).

On start-up, open last used project (TAB: Miscellaneous)
: When VisualPlace starts, it can automatically open the project that was active in the last run. If this option is *not* ticked, VisualPlace starts with an empty screen.

Cache converted Gerber files (TAB: Miscellaneous)
: VisualPlace rasterizes Gerber files before presenting them. If this option is set, the rasterized (bitmap) file is stored in a cache folder, so that VisualPlace does not need to repeat the rasterization process each time that the file/project is loaded. This trades space for speed —the cached bitmap file takes space on the drive.

Automatically check for updates (TAB: Miscellaneous)
: If this option is set, VisualPlace will check once per week whether an update is available.

Bar-code scanner (TAB: Miscellaneous)
: The name or serial port for a USB bar-code scanner. The list is filled with USB keyboard devices and USB serial devices. When you do not have a bar-code scanner, select 'none'.

PDF reader (TAB: Miscellaneous)
: The name (plus optionally the path) of the PDF reader to use. This option is only enabled when running under Linux (with Wine); under Microsoft Windows, VisualPlace always uses the PDF reader that is configured as the default.

More visualization settings can be found in the View menu. For example, figure 8 (page 20) also shows component shapes on top of the PCB image —rather than the marker shapes. This option can be set with menu option View / Show package shapes. This, and other options are also in the dialog

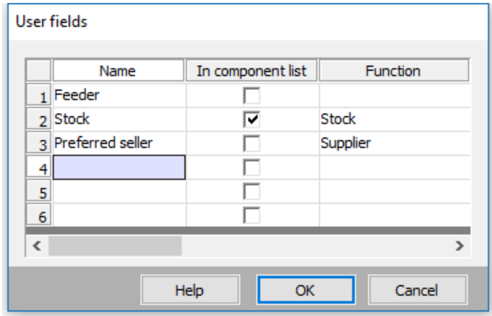


FIGURE 10: *User fields*

below View / Marker, Footprint & PCB visualization. . . See also section PCB and marker visualisation on page 26.

User fields

The component list and the bill-of-materials contains at least the value, the package (footprint) and the reference designator for each component. These lists may contain additional, user-defined fields, such as stock location or supplier information.

The dialog under Tools / User fields allows you to specify these user fields. For each user field, you may select whether or not you wish to have this field appear in the component list. All user fields are written to the bill-of-materials file that VisualPlace generates (if you create or modify the bill-of-materials from VisualPlace). For the reports, in PDF format, you may select which user fields to include.

For each user field you can also set a 'function' for the field. The function may be needed if the relevant component fields are updated from an inventory system, or linked to another application via a plug-in. The currently available functions are:

Stock	Supplier	Status
Storage	Order number	
Manufacturer	Cost	

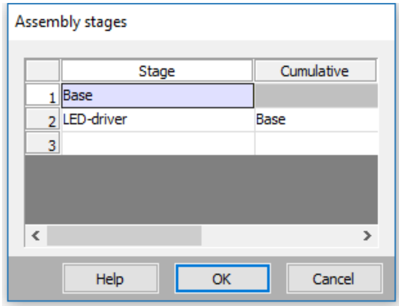


FIGURE 11: *Assembly stages*

Assembly stages

When the assembly is distributed over different pick-&-place machines in an assembly line, or when a project is partially machine-assembled and partially hand-assembled, you can define 'stages' for the assembly and assign parts individually to a particular stage.

Another use for stages is for when a PCB is designed for multiple variants of a product. The PCB will have a collection of components that are always placed, plus one or more sets of components that are only placed for each variant. For this purpose, you assign these components to a separate stage.

For a new project, all parts are assigned to the first stage, and the default name of that default stage is 'Base'. The dialog below Project / Edit assembly stages allows you to change the standard name and define others stages.

A single assembly stage is active at any one time. The parts in the active stage are sorted on top in the component list. The markers for parts in *inactive* stages is the same as components marked as 'do not mount'.

However, a stage can be set as 'cumulative' to another stage. This means that all parts of both these stages are mounted. For variants of a PCB, you would assign the parts that are common to all variants to the base stage, and create additional stages for each variant. These 'variant' stages are then set as cumulative to the base stage.

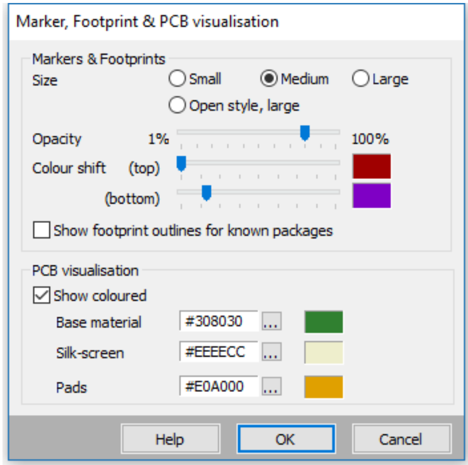


FIGURE 12: *PCB and marker visualisation options*

PCB and marker visualisation

By default, VisualPlace draws a PCB as green with a white silk-screen, markers on the top side as red and markers on the bottom side as purple. These colours, and the size of the markers can be adjusted in the Marker, Footprint & PCB visualisation dialog, below the View menu.

Especially when using a camera for hand assembly, you may want to use the 'open style' shape. With this shape, small chip resistors and capacitors can be viewed inside the marker.

The Opacity control sets the transparency of the markers. When set to a low value, the markers will be barely visible.

If the option Show package shapes for standard packages is ticked, VisualPlace draws the shape and size of the package body and pins, instead of the standard marker. This applies to the standard packages, that VisualPlace knows about. Shapes for non-standard packages (e.g. connectors, switches, ...) can be supplied to VisualPlace as SVG images. The SVG image should have the name of the package, and be copied into the media subdirectory below the installation location of VisualPlace.

See figure 8 on page 20 for an example of the visualization of known packages. Note that if a component has neither a shape that is predefined in VisualPlace nor an SVG image, VisualPlace still shows the standard marker.

Note that the project needs include a Gerber file for the 'PCB profile', in

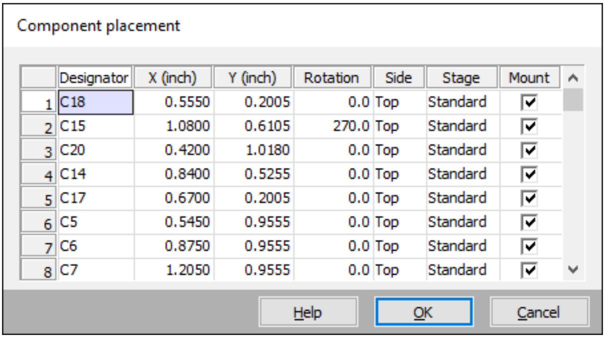


FIGURE 13: *Component placement editor*

order to colour the PCB. When no profile file is present, the PCB is always drawn black on white. The pads are only coloured if a solder mask (or solder paste mask) is set in the project. To edit a project, see Modifying a VisualPlace project (page 20).

Adding or correcting a position

When a component position is lacking from the component placement list generated by the EDA suite, or when its position (or orientation) is incorrect, it is generally advised to correct this in the EDA suite —and then generate the component placement lists again. It may be convenient, though, to add or change the position in VisualPlace itself. For example, for manual population, it may be convenient to also show the locations of through-hole components —components that are frequently lacking from the component placement lists as generated by the EDA suites.

You can edit the table with component placements by selecting from the menu Edit / Placement table. . . The component data is displayed in a table, see figure 13. The table lists all components in the bill-of-materials. Components that are lacking from the component placement list have empty fields for the position and orientation. To change a field, select it and type the new value. To *modify* the existing contents, without replacing it entirely, select the field and press F2 to enter 'edit mode'.

The positions are either in millimetres or in inches, depending on the selection made in the 'application settings' dialog below Tools / Application settings. . .

The component placement editor (figure 13) only lists existing components, and it does not allow to create new components. If you wish to add a new component, first add it to the bill-of-materials. See Adjusting the Bill of Materials on page 29. Once it is in the bill-of-materials, its position and attributes can be edited.

There is also a visual method for adding or changing the position of a component: first select the component in the component list, then select the menu item Edit followed by Add/change component placement... The component is located to the PCB side that is currently visible. The same functionality is also available from context menus on clicking the right mouse on the component list or on the PCB image (in the board view).

While picking the position for a component, the mouse cursor changes to a cross-hair cursor that allows for precise positioning.

After adding a position for a new component, the orientation of the component is not set: the component will be marked as having an 'unknown orientation'. To set or change the orientation of the component, right-click on the marker for the component and choose Rotate centroid orientation, or edit the placement list through the dialog in figure 13.

Function keys accelerate the adjustment and/or addition of placement information:

- F5 moves the marker of the currently selected component,
- Shift+F5 centres the marker inside the footprint shape,
- F6 rotates the orientation of the marker, by default with $90°$ (this step size can be changed to $45°$ through the menu: Edit / Rotation step).

If multiple components are selected in the component list, the marker that the mouse cursor hovers over, is the one that gets moved or rotated.

VisualPlace can scan the footprints on the silk-screen image and in many cases, it is able determine the centre position of the shape. For a symmetric component, the centre is also the centroid: the position relevant for the pick-&-place machine. You can ask VisualPlace to centre the centroid marker inside the footprint shape by right-clicking on the marker and selecting Centre centroid in footprint from the context menu, or with key combination Shift+F5 (while hovering over the marker).

Footprint scanning requires that the project uses a Gerber file for the silk-screen. When using bitmap images, it may not work or it may be inaccurate.

You can also ask VisualPlace to automatically centre the centroids for new placements. When adding placement information to a component that does not have a know position yet, you will then need to click somewhere

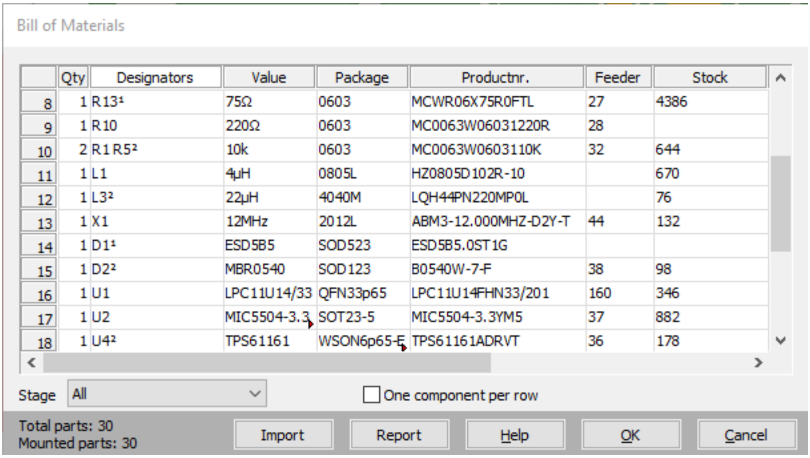


FIGURE 14: *Bill of Materials editor*

inside the shape. VisualPlace automatically centres the marker. When moving a component that already has placement information, VisualPlace *never* automatically centres the marker.

When adding or changing the placement information, either by editing the table or by visually setting its position, VisualPlace switches the component placement list to its intrinsic format. You can export the information into various text formats and machine-specific formats.

Adjusting the Bill of Materials

You can edit the bill-of-materials file inside VisualPlace. Select the item Bill of Materials. . . from the menu Edit to get the dialog for editing the bill-of-materials. When closing the dialog with the button OK, the changes are automatically saved.

VisualPlace uses the production files from your EDA suite —*unless* you edit the bill-of-materials or the component placement list. In that case, VisualPlace saves it in its own format (and adjusts the project accordingly). If updated production files were exported from the EDA suite, and you wish to reload these into a modified project, either edit the Project settings, or revert to the EDA-specific files (see the menu Project) —but you'll loose the changes you made in VisualPlace.

A designator in the 'Designators' column may have a suffix. If that suffix is a $\times$, the component is marked as 'do not mount'. The component is not included in the 'quantity' field. A number in superscript behind the designator, like R5^2, means that the component is assigned to an assembly stage —in this particular example, component R5 belongs to stage 2 (see Assembly stages on page 25 for the definition of the assembly stages). It depends on the selection of the stage, in the bottom left of the dialog (figure 14), whether the components are included in the 'quantity' field.

If no bill-of-materials file was originally assigned to the project, a new file is created and assigned to the project. This way, VisualPlace can create a bill-of-materials from a component placement list —it is however advised to create the bill-of-materials from the EDA suite.

Table editing

To change a field, select it and type the new value. To *modify* the existing contents, without replacing it entirely, select the field and press F2 to enter 'edit mode'. To duplicate the value in the field above, use the Ctrl + D combination.

To change values of individual components, it may be more practical to edit the components individually, with a single component per row. At the bottom row of the dialog (figure 14), the option One component per row toggles between the 'grouped' view (the default) and the view with one component per row. In grouped view, all components with the same value and the same package are collected on a single row.

When the option One component per row is active, you can copy all fields of a row to a list of other rows. For this function, right click on the row to pop up the context menu for it, then select Copy fields to other designators... VisualPlace prompts you for a list of designators, to which the fields must be copied. The designators in this list may be separated with space characters or commas.

To remove dummy components, such as pads for test pins and mounting holes, from the bill-of-materials, erase its reference from the 'Designator' column.

A right-click on a row opens a context menu (typing the 'menu' key does the same). The contents of this menu depends on the active column in the table. Registered plug-ins are also included in the context menu. These plug-ins can, for example, invoke an inventory database system to look the component up. or open the associated data-sheet in a PDF browser. For selecting which plug-ins are available and enabled, see section Configuration on page 21. The context menu may also have a default action

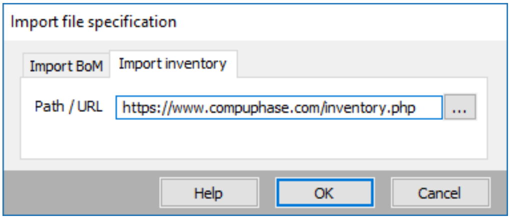


FIGURE 15: *Import component/inventory data*

(set in bold in the menu). If there is a default, this action will run on a double-click on the table cell.

In the 'Package' column, apart from typing a package name directly, you can select one from a list with search and filtering on parameters. For this function, open the context menu and choose 'Select package'. This opens the dialog in figure 25, page 56. See that section for more information on how to select a package.

Bar-code scanner

The bar-code scanner is active in the bill-of-materials table. Scanning a code selects the row with a match in the component value, the product number or in any of the user values.

Importing component data

The Import button on the bill-of-materials editor shows a dialog with two TAB-pages. The first, 'Import BoM' updates the fields in the editor from a bill-of-materials maintained in a separate application. VisualPlace can import from a file or from a web link (HTTP protocol). The browse button can be used to select a file. For a web link, specify the complete URL, including the prefix 'http://' and any 'query string' parameters.

In the case that the imported data includes reference designators for the components, VisualPlace matches the imported data in the reference designators and updates all fields in the table. When the imported data is a general-purpose component table, VisualPlace matches on the value and package and updates only the (manufacturer) product number and the user fields. In both cases, the fields that are modified are coloured light green.

See appendix A for more information on the file/data format.

Importing inventory data

The Import button on the bill-of-materials editor shows a dialog with two TAB-pages. The second, 'Import inventory' matches the fields in the editor against an inventory database (or stock management system) and updates the relevant fields. VisualPlace can import from a file or from a web link (HTTP protocol). The browse button can be used to select a file. For a web link, specify the complete URL, including the prefix 'http://' and any 'query string' parameters.

After loading an inventory file, VisualPlace checks and updates the inventory information. If a product number is set for a component in the bill-of-materials, VisualPlace checks whether this product number is present in the inventory; the field will be coloured red if it is not. If no product number is present, VisualPlace uses the description, value and footprint of the component to find a match in the inventory. If one is found, the relevant product number is filled in. The field is coloured green for a good match, and yellow for a close match.

Furthermore, the fields in the 'Productnr' column will show a drop-down button. When clicking on the drop-down button, a dialog opens with the records of the inventory. This function allows you to manually select a matching component from the inventory, in case an automatic match is unsuccessful.

Next to the product number, VisualPlace can also update the stock from the inventory. For this function, a 'user field' with the name 'stock (or the translation of 'stock' in the language selected for the user-interface) must be configured. See section User fields on page page 24 for more information.

VisualPlace uses plug-ins to provide support for inventory databases —as each uses its own internal format and access method.

Adding fiducials

VisualPlace reads the positions of any fiducial marks from the component placement list. In case that VisualPlace fails to recognize the fiducials —for example, because they are missing from the component placement list, these can be added in VisualPlace. From the Edit menu, choose Fiducials.

The table lists all known fiducials, for both sides of the PCB. A fiducial may be marked as 'primary' or as 'reference'. Many pick-&-place machines use one of the fiducials as the origin. The fiducial that you wish to use as the origin must be marked as the 'primary' fiducial. There should be only a single primary fiducial at each side.

To erase a fiducial position, delete the x and y fields. Fiducials without valid position are automatically deleted.

To manually add a new fiducial, press the button 'Add Fiducial...' below the table. The dialog is temporarily hidden and the cursor changes to a cross-hair. After clicking on the location where the fiducial should be placed, it is added to the list in the dialog. Any particular spot on the PCB can be marked as a fiducial in VisualPlace. In absence of true fiducials, you can use any pad or mechanical hole.

Depending on the board (and the design files that are part of the project), VisualPlace may be able to scan the fiducial marks from the files. Only round fiducials with a diameter betweem 0.5 mm and 3.0 mm are detected. Depending on the size and complexity of the board, this may be a lengthy procedure, as the program needs to perform various check to distinguish test pads from fiducials.

Adding notes

You can add notes to a project to describe anything for which the standard tables are unsuitable. Examples for such notes are the comments about the heat profile that should be used for soldering, or calling attention to particular points that may need review or inspection.

To add a note, either choose the item Add/edit note... from the Edit menu, or right-click on a spot on the PCB image and select Add note... from the context menu. With the latter method, the note is linked to the location that you clicked on —otherwise the icon for the note is automatically positioned in the left margin of the PCB image.

The text of the notes is hidden by default. Instead, an icon for the note is placed on the PCB image. This icon appears as a white 'i' in a blue circle. When hovering with the mouse cursor above the icon, the note pops up.

Notes can be edited, moved and deleted. The first step is to right-click on the icon for the note (a white "i" in a blue circle) and choose the appropriate option from the context menu. To delete a note, choose to edit it; then delete all contents and close the dialog.

The note may optionally be included on the reports for the PCB lay-out or on the bill-of-materials (or both). To change these options, edit the note by right-clicking on its icon.

Manual population of a PCB

The *component list* lists all components by category, value and type (by default). The order in which the categories (resistors, capacitors, transistors, integrated circuits, inductors, ...) are listed, is configurable — choose the item Placement order & flags... from the menu Edit. After selecting a row, the positions of the components in the selected row are marked on the silk-screen image in the *board view*.

In case a selected row is not fully visible in the component list, you can activate a call-out with a summary of the row, using the key F3. Hovering (with the mouse pointer) over the marker of a component also shows you the information on that part in a call-out.

To populate the board, one steps through the component list one row at a time. On occasion, a component may (partially) obscure another one, and the order that these components are placed becomes important. In such situations, you may override the placement order, see the section Adjusting placement order further on in this chapter.

The markers on the silk-screen image have an arrow to indicate the orientation of the components. The arrow points in the direction of pin 1 of the component. For polarized components with two pins, pin 1 is the + pin (capacitors) or the cathode (diodes). Integrated circuits (dual in-line packages, quad packs, etc.) have pin 1 at the upper left when the marker points upwards. See figure 21 (page 47) and appendix B for details.

There are two colours of the marker: the components on the top side are marked as red, the components on the bottom side as purple —these are the defaults, the colours, size and transparency are adjustable, see section PCB and marker visualisation on page 26. In the component list, VisualPlace also groups components to be placed on one side of the board separately from the other side. So, if a board has components on both sides, it may occur that the component list has two separate rows with exactly the same component value and package: one row for all instances of this component on the top side, and one row for all instances on the bottom side.

Through the menu, you can select whether the silk-screen for the top side or that for the bottom side is displayed (alternatively, press function key F4 to toggle between top and bottom view). VisualPlace only displays a single silk-screen at a time. Note that, in contrast to Gerber viewers, the silk-screen for the bottom side is shown as it would appear if you were to look at the PCB. That is, it is mirrored relative to the Gerber file.

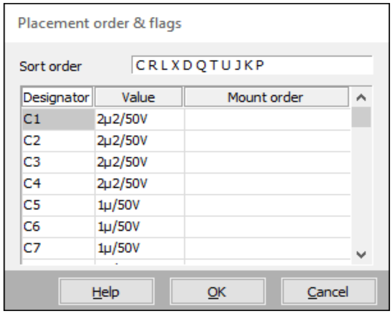


FIGURE 16: *Component sort order*

The components are marked on the silk-screen regardless of which of the PCB sides is currently shown. If you view the top side of the PCB and select a component at the bottom side, its location will be shown. Markers for components on the top and bottom sides have different colours. Note that the PCB in the main view automatically flips as soon as you select a component on the reverse side.

Adjusting placement order

VisualPlace sorts components by their category (resistor, capacitor, ...) and their value. It is practical to place the components in the same order as how they are listed in the component list. However, depending on the PCB lay-out and the components, it may be more efficient, or even required, to place a particular component *before* or *after* other components.

The dialog also allows you to flag components to be *not mounted*. This option is discussed at the end of this section.

To change the sort order, choose the menu item Edit and then Placement order & flags... VisualPlace has a general sort order, based on the prefixes of the categories, and it allows overrides for individual components. The general sort order allows you to place all resistors before all semiconductors, for example.

The table in the dialog (see figure 16) lists all components in the active project. In the third column, you can select an override for the position that a component takes in the component list. The possible options are:

Head — The component is listed topmost in the component list.

Head of category	The component is listed on top of all other components in the same category. For example, if you have a mix of a physically large and small capacitors, it allows you to place the small capacitors *before* the large ones, but still keep all capacitors in one group.
Normal	The default (no override).
Tail of category	The component is listed below all other components in the same category.
Tail	The component is listed at (or near) the bottom of the component list. Note that components that lack placement information, and components that are marked as 'do not mount' are below the components marked as 'tail'.
Do not mount	Although the component has placement information, it is not to be placed on the board. This component is listed at the bottom of the component list —below the components marked as 'tail'.

The 'do not mount' attribute is useful if some components do not need to be placed on a PCB, but you do not want to loose the placement information for these components. For example, a product may be available in an 'entry model' and a 'professional model', both based on the same PCB, but where the professional model has components that are lacking from the entry model.

When components are never needed on a PCB, these can be deleted from the bill-of-materials, see page 29.

Grouping, expanding and collapsing tree rows

In its default mode, the component list groups components by their value and footprint. For example, if a PCB has four 0805 capacitors of 2.2 μF/50V, VisualPlace shows these on a single row in the component list and puts four markers in the board view when the respective row is selected —see the figure on page 16.

When a PCB has *many* components of the same value and footprint, the grouping may become confusing rather than helpful. In these cases, you can expand the row and step through all component positions one-by-one. Every row that groups multiple components has a '+' icon on its left.

VisualPlace can also automatically expand a row —and collapse any previously expanded row. When the menu item View / Auto-expand component list is active, selecting a row in the component list automatically expands it.

Ticking components off

It is usually practical to populate a PCB in the same order as the component list. When a PCB is assembled in multiple phases or in a non-linear fashion, you may want to tick off the components have been placed on the PCB.

VisualPlace allows you to place a tick mark (checkmark) next to every component. If a row in the component list represents a group, the entire group gets a tick mark when all items below it are ticked. You can also *tick* or *untick* an entire group at once.

Before using tick marks, the option needs to be enabled, via the menu Tools and then Tick-marks. This menu also allows to tick or untick all items in the component list.

VisualPlace saves the tick marks in the project file as 'user data'. Alternative uses for the tick marks may therefore be imagined —for example, a project with a PCB on which several components are optional (for 'light' and 'full' versions), can have the required components *ticked* and the optional components *unticked*.

PCB zones

VisualPlace allows you to subdivide the PCB into zones and have the components sorted according to the zone that they are in.

A scenario in which zones are useful is when a particular component must be placed in so many locations on a PCB, that it becomes difficult to keep an oversight. VisualPlace can be set up to show only one one component at a time (see the Auto-expand component list option from the View menu). Zones are a middle ground between these two: when subdividing a PCB into zones, you can focus on the placement of all parts in each zone separately.

When using a camera for manual placement, another scenario for defining zones is when the PCB is too large to completely fit in the camera frame. In this case, zones allow you to populate each zone of the PCB in sequence, and re-align the PCB in between.

Open de dialog below Edit / Placement zones... to define zones. In this dialog, you must first decide in how many columns and how many rows the PCB must be divided. In two grids below the numeric fields with the column/row count, the number of cells changes according to the contents of the numeric fields.

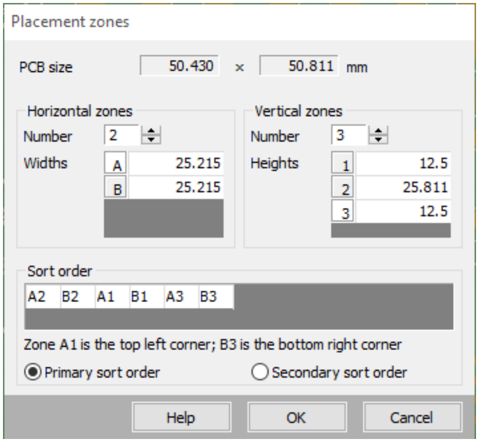


FIGURE 17: *Configuration for assembly zones of a PCB*

By default, a PCB is divided in columns and/or rows of equal size. You can change the width of each column and the height of each row. If you modify a value in the grids for the column or row sizes, the field or the *next* column or row is also adjusted, so that all column widths add up to the PCB width and all row heights add up to the PCB height.

The order in which you walk through the zones is adjustable as well. By default, the order starts with zone A1 (column A, row 1), then moves through all columns (B1, C1, ...) before moving on to the next row (A2, B2, ...). Column A is at the left edge of the PCB and row 1 is at the top of the PCB. Therefore, zone A1 is at the top left corner of the PCB. If you have three columns and two rows, zone C2 would be at the bottom right corner of the PCB.

To change the sort order of the zones, type in the zone codes (A1, B3, ...) in the sequence that they must be ordered. When changing a field in the 'sort order' table, VisualPlace recalculates the sort order so that each zone is in the list —and only once. That is, editing a single field frequently also changes another field in the table.

The sort order of the zones can be a primary criterion to sort the components on, or a secondary criterion. When the zone order is a primary criterion, the components in the component list will first enumerate all components, of all types and all values, that are located in that zone. Below that, the component list enumerates all the components that are in the next zone. As a result, a particular component that must be placed in several zones on the PCB, appears multiple times in the component list.

When the zone sort order is a secondary criterion, the component list groups all parts with the same type, value and footprint (just like in the absence of zones). However, when moving through the component list with the space bar, VisualPlace will at first show all parts (of that type, value and footprint) in the first zone only. At the next press of the space bar, VisualPlace shows the parts in the next zone. For information of the functionality of the space bar, see the section keyboard navigation, below.

Keyboard navigation

For convenience during manual assembly, VisualPlace allows you to navigate the component list with the keyboard. The standard *up arrow* and *down arrow* keys move the selection bar one position up and down, respectively. The *Page Up* and *Page Down* keys also behave as expected.

The keyboard shortcut to expand a row is the → key and the key to collapse a row is ←. See the previous section on expanding and collapsing groups of the same component value/footprint.

Tapping on the *space-bar* moves to the next component to place. It differs from the *down arrow* because it does not necessarily move to the next row in the component list, and it may also skip a row:

- ◇ If auto-expansion of the component list is active, the space-bar skips the rows that holds the groups of components. Instead, it expands the group and jumps to the first *sub*-row of the group (i.e. it moves to the first component of the group).
- ◇ When auto-expansion is *inactive*, the space-bar initially shows as many components of the group as fit in the viewport —but a subsequent press on the space-bar will first scroll to any next group of 'previously unseen' components, before moving on to the next row. If PCB zones are active *and* set up as secondary sort criterion, the space-bar shows the components in each zone individually, in the 'sort order' that has been configured for the zones.

VisualPlace registers two sound events for a change of the selection: one when the selection has changed to a component with the same value and package as the previously selected component, and one for a component with a new value/package. When using the space-bar to move through the component list, you may want to assign sounds to these events, to get an audible cue when changing from one group to another. For adjusting the sound event assignments, use the Control Panel in Microsoft Windows ('Sounds and Audio Devices').

The F7 key also moves down in the component list, like the *down arrow* key, but in addition F7 toggles a tick mark for the row. Tick marks must

FIGURE 18: *USB dual foot switch: for browsing through the component list*

have been enabled first —see the menu Tools and then Tick-marks. You can also right-click in the component list for these menu options.

For a comprehensive list of keyboard shortcuts, see appendix H.

Foot switch or palm switch operation

Especially during manual assembly, and especially when auto-expansion of the component list is active (as covered in the preceding section), it may be practical to move down (and possibly up) through the list with a foot switch, so that the hands remain free for picking up and placing components.

Various USB foot switches are available that are suitable for this task. These foot switches mimic a key press when pressed. In most cases, the *space-bar* key is the most appropriate key to assign to the foot switch — each tap on the foot switch will then move to the next component(s) to place (see also the previous section on keyboard navigation).

If a foot switch is inconvenient, a large-size push-button switch (or 'palm switch') is easily equipped with an USB 'HID' interface to simulate a key press.

Navigation button bar

As an alternative to a foot switch, you can enable a button bar in Visual-Place with a set of navigation buttons: View / Navigation button bar. The button bar appears at the bottom of the board view. It has buttons to step through the component list and to zoom in or out. See figure 19 for an example.

The navigation button bar is intended to be used with a system without keyboard but with touch screen, such as a tablet or a panel PC.

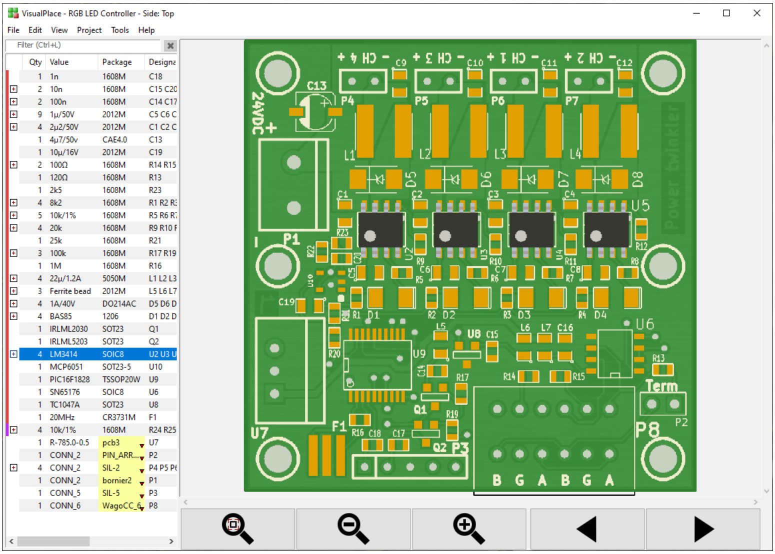


FIGURE 19: *Navigation bar, for use with a touch screen*

If the option for tick marks has been enabled on the component list (see Ticking components off on page 37), clicking on the button for 'next component' also ticks off the component(s) on the current row.

Live camera view

To use VisualPlace with the live stream of a camera, first the camera needs to be set up. This is covered in appendix D. For an impression of camera-assisted manual assembly, see figure 1 (page 5).

Once set up, you can choose item View Camera from the View menu to to toggle between the design files (the silk-screen images) and the video stream from the camera. Alternatively, you can use function key F11.

When in camera-view mode, VisualPlace shows four registration markers —one at each corner of the PCB. You must place the PCB between the registration markers. If the PCB does not fit well between the registration markers, the camera is not calibrated correctly. See appendix D for guidance to calibration.

If the PCB is too large to fit in the camera view, several (and possibly all) registration markers are outside the viewport —and therefore not visible.

An alternative way to align the PCB below the camera uses the drill file. When the camera is active, VisualPlace puts a cross-hair marker in every hole that is larger than 30 mil (approximately 0.8mm). See page 9 to add a drill file to the project. The cross-hair is also drawn on fiducial marks on the PCB. Note that for VisualPlace, fiducials need to be exported into the component placement list.

Zooming and scrolling operates differently in camera-view mode. The Zoom menu is disabled; the availability of zooming and the zoom factor depend on the camera. You can use the keys + and – on the numeric keypad to zoom in and out, but only with one level. Note that a PTZ camera is required to support local zooming.

When you click inside the frame of the video stream, a PTZ camera will zoom in to that spot. To zoom out, click-and-hold inside the video frame. The camera will zoom out after approximately 0.5 second. Alternatively, you can use the – key on the numeric keypad to zoom out.

If the PCB is too big to fit completely in camera view, you can define zones for the PCB and populate each zone separately. See the section PCB zones on page 37 for information on zones. To populate a zone completely, before moving to the next zone, make the zone the primary sort criterion.

Preparing for automated population

Pick-&-place machines (also referred to as 'PCB assembly machines') can pick up components from tape, tube or tray carriers and place these on a PCB with high accuracy, and at high speed. But before a pick-&-place machine is able to do so, it has to be programmed. For the task of seamlessly and effortlessly programming a pick-&-place machine, it is vital that the component placement list (input data for the pick-&-place machine) is accurate, complete and in a format that the machine understands.

Verifying the component placement list

A correct component placement list is a great help in automated assembly of a board. Therefore, when you send out a CPL file to an assembly plant, you will want to verify that it is complete and accurate.

For machine assembly, only surface mount components should be in the component placement list. When drawing a new 'library part' in your EDA suite, you can typically select whether the part is 'surface mount' or conventional ('through-hole'). If these attributes are set incorrectly, due to an oversight, the components for that library part may be lacking from the component placement list whilst they should be present, or vice versa: present in the component placement list whereas they should not be.[1]

VisualPlace combines the information from the component placement list with the bill-of-materials. It places any component that is in the bill-of-materials, but absent from the CPL file at the bottom of the list. Therefore, to verify the completeness of the component placement list, follow these steps:

- ◇ verify that the component list is sorted 'on value' (menu View / Sort components / On component value,
- ◇ select the item at the bottom of the component list,
- ◇ if necessary, press F3 to pop up the call-out with the component details,
- ◇ and verify whether the call-out shows a 'UNKNOWN POSITION' warning.

[1] See the notes on your EDA suite in appendix C for why components may be missing from the CPL file.

There may be through-hole components at the bottom of the list, and these will not be placed by a pick-&-place machine. Therefore not all 'UNKNOWN POSITION' warnings may be relevant. By browsing through the components from the bottom up, you will view all components *without* valid placement information first.

Although VisualPlace has a menu option to add placement information to components that are absent from the CPL file (see section Adding or correcting a position on page 27), if the purpose is to create CPL files for automated population, we recommend to make the corrections in the EDA-suite. This avoids having to redo the same correction after every board revision.

Verifying the centroid markers

For automated placement, accuracy is essential. For symmetric packages, the marker that VisualPlace puts on a selected component, should precisely be in the middle of its footprint. The marker indicates the centroid —the centre of mass. For a symmetric package shape, the centroid coincides with the geometric centre.

VisualPlace switches to a cross-hair cursor when you press and hold the Ctrl key. Graticule markers on the cross-hair let you visually determine the centre of a component. The mouse cursor snaps to the (red or purple) marker on a selected component; it may also snap to pad centres if those centres were exported in the Gerber (X2 format) files. To avoid snapping, press and hold the Shift key (in combination with the Ctrl key).

For footprints that are larger than the cross-hair cursor, or for footprints where the centroid marker is *not* in the centre of the footprint, VisualPlace allows you to measure the distance between two points, such as the distance of a marker and an edge of the footprint. To perform a measurement, press and hold the Ctrl key, click on one position on the board and 'drag' (with the left mouse button kept pressed) to another position. VisualPlace shows the distance between the two points in a label attached to the cross-hair cursor. The distance is either in millimetres or in inches, depending on the selection made in the 'Application settings' dialog (Tools menu).

If the centroid position is *incorrect*, the first step is to determine the source of the error:

◇ If *all* positions are incorrect, it is likely that PCB origin is incorrect or that the generated image of the PCB has a profile that does not match the value set in the project —see sections Aligning the origin on page 12 and Create the VisualPlace project on page 9.

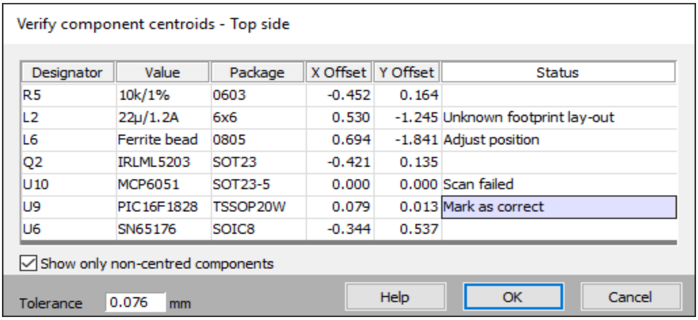


FIGURE 20: *The component placement list compared to the analysis of the Gerber files*

- ◇ If the positions all all components with a particular footprint are incorrect, the footprint offset should be corrected. See section Adding a component placement correction on page 59.
- ◇ To correct the position of a single component, see Adding or correcting a position on page 27.

See also section Verifying positions and displacements on page 53 to determine the offset of the coordinates for the correction.

For a quick overview of all components that may need attention, use the item Verify component placement from the menu Tools. The dialog that opens, shows all components whose position in the component placement list does not match the centre of the drawing on the silk-screen.

The tolerance for comparing the placement positions is configurable, as shown in figure 20. Depending on the application settings, this value is in mm or in mil. When the tolerance is set to 0.1mm, for example, the table only shows parts where the difference in placement position is above 0.1mm. However, if the option Show only non-centred components is off, *all* components will be listed, regardless of the difference.

When you click on the column Designator, VisualPlace selects that component, so that you can visually inspect the position of the centroid marker. A click in the column Status gives you the following options:

Adjust position
: The component is moved to the position that VisualPlace detected, By this, you confirm that VisualPlace scanned the footprint correctly —and that the CPL file was inaccurate. The corrected position is written into the CPL file.

Mark as correct

Consider that the position in the CPL file is correct —and that position obtained from the footprint scan is inaccurate. See below for the reasons why the footprint scan may fail. VisualPlace will ignore the discrepancy between the CPL file and the scan in future scans.

Correct package

Consider that the position in the CPL file is correct *and* that the footprint scan is also correct, but that the package *specification* is incorrect. This happens on packages where the centroid (the centre of mass) does not match the geometric centre —so asymmetric packages. When selecting this option, VisualPlace adds an appropriate correction record to the package.

Note that the 'package specifications' dialog also allows you to add or modify placment corrections; see section Adding a component placement correction on page 59.

Clear status

To not perform any action (this is the default).

Note that the scanning of the footprints from the silk-screen is based on heuristics. These heuristics may fail or may give inaccurate results. In general, *closed rectangular* shapes are handled best. The heuristics are hampered by text or drawings inside the footprint. An 'open' shape or an asymmetric shape may also result in a failure to correctly scan a footprint. In the column Status in the table, it will then say 'Scan failed'.

If the status says 'Footprint mismatch', it indicates that the package that is specified for the component does not fit on the footprint, given the current position and orientation of the component. First point to verify is whether the correct footprint/package is specified for the component. If that is correct, verify the orientation. A failure to scan the silk-screen may also lead to this warning.

The status 'Footprint deviates by. . . ' means that the component fits on the footprint, but it appears not to fit well. For example, a `0603` component fits on a `0805` footprint —though tombstoning may occur during reflow. Some packages allow for greater tolerances in matching a footprint, than what VisualPlace uses by default. When you mark in the table to correct the package specification, VisualPlace adjusts the tolerance for matching package pins to the footprint pads —but only for that footprint.

To improve the accuracy of placement scanning, you are advised to add a mask Gerber file to the project —by preference, the solder paste mask, but the solder mask is suitable as well. To be able to use the mask image, VisualPlace needs to know the orientation of the part and the speci-

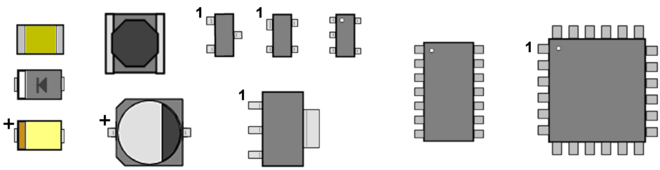


FIGURE 21: *Normal orientations of components, see appendix B*

fication of the footprint (or package) lay-out. See the dialog below Tools / Package specifications... for selecting a specification for each footprint and for creating new footprint lay-out specifications (page 59). When a component lacks a footprint lay-out specification, the status column in the table says 'Unknown footprint lay-out'.

VisualPlace scans the Gerber files at 1000 dpi; the accuracy of detecting footprint centroids is therefore limited 1 mil (0.025mm).

Verifying the component orientations

The component placement list contains the 'rotation' in addition to the position of the board. Rotation is relative to a *normal orientation* of the package shape (and in counter-clockwise direction). The normal orientation of packages is standardized as per IPC-7351 and IEC 61188-7 'Level A'.[2] In summary, when the rotation is zero degrees:

- ◇ Dual in-line packages (SOIC, SSOP, TSSOP, etc.) have pin 1 at the upper left.
- ◇ Quad packs (QFP, TQFP, QFN, etc.) and BGA packages have pin 1 at the upper left. Some PLCC packages have pin 1 in the middle of a side, in this case, pin 1 is at the top edge (when the rotation is zero).
- ◇ Two-pin components (such as resistors and capacitors) have pin 1 at the left. For diodes, pin 1 is the cathode, for polarized capacitors, pin 1 is the + pole.
- ◇ Transistors and packages with three or more pins, have pin 1 at the top left.

As stated above, rotations are counter-clockwise. A rotation of $90°$ is a quarter turn counter-clockwise and a rotation of $270°$ is a quarter turn clockwise. This assumes that you are looking at the side of the board at

[2] There also exists an IEC 61188-7 'Level B' standard, but 'Level A' is predominant, because it is consistent with the IPC norm.

which the component is. In a CAD program, you commonly look at components that are at the bottom side *through the board* from the top side. From a top-side this perspective, rotations of components at the *bottom side* are clockwise.

VisualPlace has an arrow in its marker for a component (or a dot for components that should not be placed). This arrow points to the orientation of the component. It is advised that the silk-screen(s) of the board also have an indication for the polarity of the component. This allows you to verify whether all components are oriented correctly in the component placement list.

Orientations can be corrected similarly as placement positions:

- ◇ To correct the orientation of a footprint, see section Adding a component placement correction on page 59.
- ◇ To correct the orientation of a single component, adjust the centroid marker or edit the placement list as described on page 27.

Assign parts to stages

When the project must be spread over multiple machines, or when the project is machine-assembled in part and hand-assembled in part, you can use 'assembly stages' to organize the work flow.

Before assigning parts to a particular stage, the assembly stages must be defined. See the description of the Assembly stages dialog on page 25.

To assign a component to a stage, you can right-click on a marker of the component or right-click on a row in the component list. In both cases, you can select the assembly stage for that component (or row of components) from the context menu. Alternatively, you can select the stage in the Placement table, see figure 13 (page 27).

Convert/export the component placement list

There exists no standard for the format of CPL files —or rather, each manufacturer of pick-&-place machines and each developer of EDA software uses its own 'standard'. VisualPlace can export the placement data in a machine-specific format, provided that a plug-in for that format is available. Auxiliary data, such as the nozzle or head selection (or constraints), fiducial alignment and other PCB-specific or machine-specific information can be entered directly from VisualPlace.

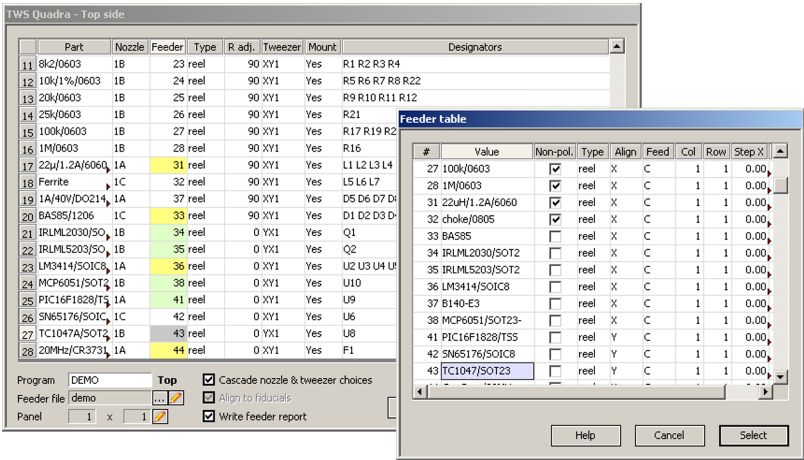


FIGURE 22: *Machine-specific interface to export an optimized component placement list*

To export a component placement list, choose the from menu File / Export / Placement Data... The next step is to give the name and path, ass well as the *format* of the exported file, in a Save as dialog. After confirming these, a dialog will pop up with the settings specific to the machine or format (figure 22).

VisualPlace has internal tables to adjust the 'zero-degree rotation' of individual parts, for the cases that the EDA suite uses a different definition than in figure 21 and appendix B. It also lets you correct inaccuracies of placement positions and remove dummy components.

Many pick-&-place machines require fiducials to attain proper alignment. VisualPlace can recognize fiducials from the CPL files. To do this, either the package name or the type/value of a component must be set to the name 'fiducial' or 'fidu-*xxx*' —where *xxx* is a label for the type of fiducial. Components in the CPL files that are recognized as fiducials are handled as such and do not appear in the bill-of-materials.

When an EDA suite uses a different description than the ones in the previous paragraph, these can be added to the built-in patterns (the built-in patterns can also be removed). This dialog is found in the menu below Tools / Fiducial match patterns... The 'type' flag of a pattern indicates whether a matching name would indicate a standard or a *primary* fiducial. Coordinates in the machine-specific CPL file often use the primary fiducial as the *origin* for the placement data. In the default patterns of

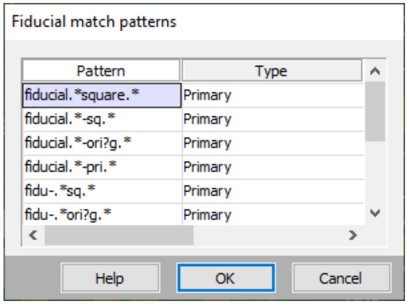


FIGURE 23: *Fiducial match patterns*

VisualPlace, a fiducial is marked as 'primary' if the package or type/value of the fiducial contains one of 'org', 'pri', 'sq' or 'square' (some systems use a square fiducial shape to indicate the primary fiducial).

In the absence of specific support for a particular pick-&-place machine, you can export the placement data in 'comma-separated values' lay-out (CSV). This is a general purpose file format that most machines can import. The exported CSV file contains all corrections done in VisualPlace —whether manual or automatic. See page 59 for details.

Generating reports

In some circumstances, an assembly facility may require the placement information on paper, for example, in the case that they do not have the equipment to transfer computer-generated files directly to a pick-&-place machine.

VisualPlace can generate several reports:

◇ PCB Lay-out: a report with the images of the PCB annotated with either the component designators or the component values. The component designators/values are drawn at the positions of each component. Accordingly, this report shows at a glance whether the placement data is complete and roughly accurate. The report will also help catch the situation that the annotations on the silk-screen are ambiguous or inaccurate.

  The dialog for this report allows you to select the paper lay-out, font size, image magnification and a few other self-explanatory settings. The opacity setting allows you to make the PCB artwork less opaque, which increases contrast with the labels (the 'annotations') and safes toner.

For easier lookup of components, a pair of rulers and/or a grid may be drawn on top of the PCB artwork. The units of the rulers and the grid are the same as those set for the application.

- Bill of Materials: the bill-of-materials, including optional extra (user) fields with ordering or supplier information.

  The Colour printing option in the dialog changes the text background of components, depending on the contents and attributes of the component row: components that lack automated placement information (such as through-hole components) are on an orange background, and components with incomplete placement information are on a magenta background. Using colour printing, these components stand out.

  See section Colour rules on the next page for additional notes on setting up colour rules.

- Placement List: a table with all components and their positions and orientations, relative to the origin of the PCB. This report is similar to the bill-of-materials, but instead of listing the details needed for ordering the components, it lists the positions. This report allows the operator of the pick-&-place machine to type in the data.

- Flipbook for assembly: a combination of the PCB images and the component placement information. This report is the 'paper representation' of the operation of VisualPlace. The report has one page for each group of components with the same value and package (and on the same side of the board). This report is called the 'assembly flipbook', because you 'flip' a page for each new group of components.

  The dialog for this report allows you to select the paper lay-out, font size, image magnification and a few other self-explanatory settings. The opacity setting allows you to make the PCB artwork less opaque, which increases contrast with the labels (the 'annotations') and safes toner.

  The components are visually marked on the PCB image on each page with markers, and their exact locations are noted in a table below the image (the table with component locations can be disabled). The markers have the same colours as set for the application —by default, red for the top side and purple for the bottom side. The marker size, however, can be separately chosen for the report.

  For easier lookup of components, a pair of rulers and/or a grid may be drawn on top of the PCB artwork. The units of the rulers and the grid

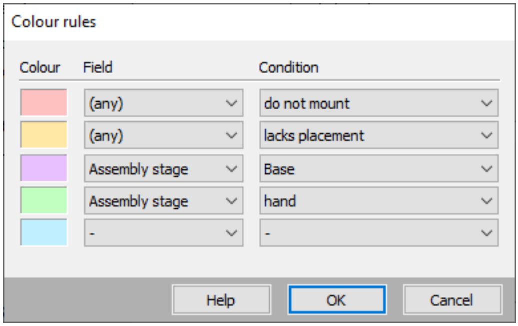


FIGURE 24: *Colour rules for the bill-of-materials.*

are the same as those set for the application.

Components that are marked as 'do not mount' are excluded from the PCB lay-out and the flipbook reports. The PCB lay-out report also excludes the components that lack placement information —the flipbook report, however, lists these at the end.

In other reports, components with the 'do not mount' attribute are identified explicitly as such, and these components are not included in the count of components (the 'quantity' column). That is, if a circuit has twelve resistors of 10k in 0603 package, but three are marked as 'do not mount', the bill-of-materials will have '9' in the 'quantity' column and list the corresponding nine reference designators, plus three more between parentheses and with the label 'do not mount'.

The reports are generated in PDF format and are opened in a PDF viewer such as 'Adobe Reader' or 'Sumatra PDF'. From within the PDF viewer, the report can either be saved (as a copy), sent by e-mail, or be printed.

Colour rules

For the bill-of-materials report, you can set up rules for printing rows in the report on a different background colour. Both the rules and the colours can be selected. The text colour is fixed to black, so the background colour for a rule should be selected such that there is good contrast with black text.

The drop-down lists for the field lets you select a coloumn or column type in the bill-of-materials. The contents of the drop-down lists for the condition depends on the selected field. When this is set to 'Assembly stage', the condition drop-down contains the names of all defined assembly stages.

In other cases, it contains a set of standard conditions. To filter on conditions like '*lacking placement data*' or '*do not mount*' status, the field must be set to '(any)'.

Coordinates and origins

In the data that VisualPlace displays on-screen, in the reports that it generates, and in the files that it exports or saves, VisualPlace assumes a coordinate system:

- ◇ with the origin in the lower left corner of the PCB —in some CPL file formats you also have the option to place the origin at a fiducial,
- ◇ and where the positive x-axis points to the right and the positive y-axis points upwards (on the screen).

VisualPlace uses this coordinate system for both the top side and the bottom side of the PCB. Most pick-&-place machines use this same coordinate system.

Note that a PCB lay-out program shows the bottom layer as viewed from above through the top layer —as if the top layer were transparent. The image of the bottom side that you see in the PCB lay-out program is therefore mirrored from how the bottom side of the PCB will really look. VisualPlace does *not* mirror the bottom side image: it displays the bottom side as it is.

Miscellaneous operations

Verifying positions and distances

The coordinates of any particular spot on the PCB can be viewed, by pressing and holding the Ctrl key. The cursor changes to a cross-hair and it displays the coordinates of the cursor on a small label inside the cursor. These coordinates of the may be in inches or in millimetres, depending on the choice in the made in the 'application settings' (menu Tools, then Application settings. . .).

The coordinates are relative to the bottom left corner of the PCB. If you wish to see a displacement from another position, press and hold the left mouse button (while also keeping the Ctrl key pressed) at the temporary origin and then move the mouse cursor to the spot of interest.

Note that the cross-hair cursor snaps to the markers on selected components. That is, if one or more components are selected in the component list and the (red or purple) markers are visible on those components, the mouse cursor snaps to the centre of those markers if it comes near enough. The mouse only snaps to the centre of markers when the Ctrl key is pressed. To avoid snapping, press and hold the Shift key (in combination with the Ctrl key).

Locating a component by designator

By default, VisualPlace sorts the components by value. When multiple components have the same value, they are grouped on a single row.

To locate a single component by its designator, use the menu item Edit, then Find component, or the key combination Ctrl + F.

When running through all components by their designators, it is convenient to have the component list sorted on the designator. To change the sort order, choose the menu item View /Sort components and then On component designator. Note that components are no longer grouped in this mode: every component is on a separate row in the component list.

Locating a component by value

The find function (menu Edit / Find component) also searches through the component value and the user fields.

When a bar-code scanner is configured, scanning a bar-code also locates the matching component. VisualPlace searches the component value, the product number, and any of the user values for a match with the scanned code. See the dialog below Tools /Application settings for configuring a bar-code scanner. The bar-code scanner must have a USB interface and be configured to emulate either a keyboard, or a serial device.

Query component details by position

When double-clicking on a location in the board view, VisualPlace selects the component nearest to where you clicked, and shows a call-out with details of the component. See figure 27 for an example of the information shown.

Only components with known positions (i.e. with placement information) are taken into account. You must double-click within one inch of the centroid marker of a component for it it to be found.

You can also double-click on a marker for a component. This will select only that single part, and expand the relevant group in the component list.

Maintaining VisualPlace tables

To accurately display the component orientations, as well as to apply automatic corrections and adjustments to particular components, VisualPlace needs to have detailed data about the packages for the components. VisualPlace comes with a few tables for the more common EDA suites, but these suites evolve and you may also have obtained component libraries from elsewhere —or have drawn your own. If components that need a placement correction are lacking from the VisualPlace tables, you have to add them yourself.

VisualPlace allows to map a footprint name or package name to a 'normative' name. Since a footprint or package often exists under several EDA-specific or manufacturer-specific names, mapping those names to a single normative name reduces the number of required footprint lay-outs. VisualPlace comes with a library of common footprints, further reducing the number of definitions that need to be made.

Package/footprint names: vendor vs normative

To be able to show accurate information on the positions and orientations of components, VisualPlace requires a lay-out of the respective footprints. In the component list, unknown footprints or packages are shown on a red background —see figure 6 (page 17).

That said, for package that is marked as 'unknown', VisualPlace may actually have a specification in its library under a different name. Some EDA suites refer to the common 0805 package as SM0805, or C0805, or 2012[0805], or other names. VisualPlace allows you to link an EDA-specific name to a 'normative' name from its library. In its reports and in the component list, VisualPlace uses the normative names, if available —thereby achieving consistency in the package names.

The Package specifications dialog (below Tools / Package specifications. . . , see figure 27, page 59) shows a list of package names from the active project, plus the relevant information on each package.

If the field in the column 'Lay-out' is 'new. . . ', VisualPlace does not yet have a lay-out for that footprint. There are two options to amend this:

- A click in the 'Normative name' column, allows you to choose one of the known lay-outs; see figure 25.

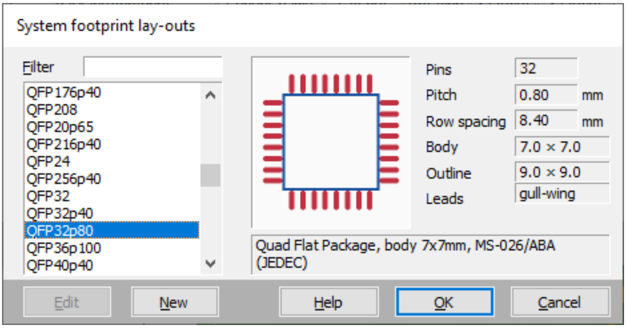


FIGURE 25: *Dialog for selecting a normative package name.*

When the fabrication files are still in the original (EDA-specific) format, VisualPlace adds a rule in its internal data files to map the vendor-specific name to the selected normative name. In addition, you have the option to add this rule to the project itself (again, see figure 27). This is convenient when you share the project, but keep the data files local.

Note that when you have edited the the bill-of-materials or the component placement list, VisualPlace has converted these files to its 'intrinsic' format. For the intrinsic format, does not create a rule (as described above). Instead, VisualPlace stores the translated 'normative' names in the fabrication files (BoM and CPL).

◇ From the dialog in figure 25, you can create a new footprint lay-out by selecting the New button. This is covered in the section Creating or modifying a footprint lay-out (on the next page).

In this 'Filter' field of the dialog in figure 25, you can filter on prefixes (such as typing 'tssop' to limit the list to TSSOP packages), you can in addition filter on pin count, pitch or span. For example, entering 'pins=32' limits the list to components with 32 pins. Multiple keywords can be combined; for example, 'pins=32 gullwing' filters all packages that have gull-wing pin style and a pin count of 32. Hovering over the 'Filter' field shows a tooltip with the full list of supported keywords.

Package naming convention

While the preceding section refers to a 'normative' name for a package, there are actually multiple standards. VisualPlace allows you to select a naming convention that suits your requirements and preferences. This setting is in the 'User interface' tab of the Application settings dialog, see section Configuration on page 21.

The options for the naming convention are:

Trade name — The name that is generally used in the industry. This is also the default setting in VisualPlace.

EIA imperial — Names according to the EIA PDP-100 standard based on measures in inches. The PDP-100 standard only covers chip and molded packages. For packages where PDP-100 does not apply, the trade name is used.

EIA metric — Names according to the EIA PDP-100 standard based on measures in millimetres. For packages where PDP-100 does not apply, the trade name is used.

IPC-7351 — IPC-7351 is a comprehensive standard on surface mount components, and it includes a naming convention. The IPC-7351 covers a broad range of surface-mount components. VisualPlace follows the upcoming revision C of the standard.

ED-7303 — The JEITA ED-7303 standard only contains packages for semiconductors —in particular, packages for integrated circuits. For packages where ED-7303 does not apply, EIA metric is used (and if that standard does not apply either, the trade name is used).

The package names listed in the 'System footprint lay-outs dialog (see figure 25) is also affected by the choice of the naming convention.

Creating or modifying a footprint lay-out

If the column 'Lay-out' in the Package specifications dialog says 'new...' or 'edit...', a click on that field opens a dialog to edit the lay-out; see figure 26.

In the Footprint lay-out dialog, you can enter the number of pins or pads for the footprint. Below the field for the number of pins/pads is a table with the X and Y positions for each pin. This table dimensions itself automatically when you change the pin count. The values to enter for the X and Y fields should be the centre positions of each pin/pad, relative to the package centroid.

For regularly shaped footprints (such as SOIC, TSSOP or QFN), you can generate the pad positions in the table. As the first step, fill in the number

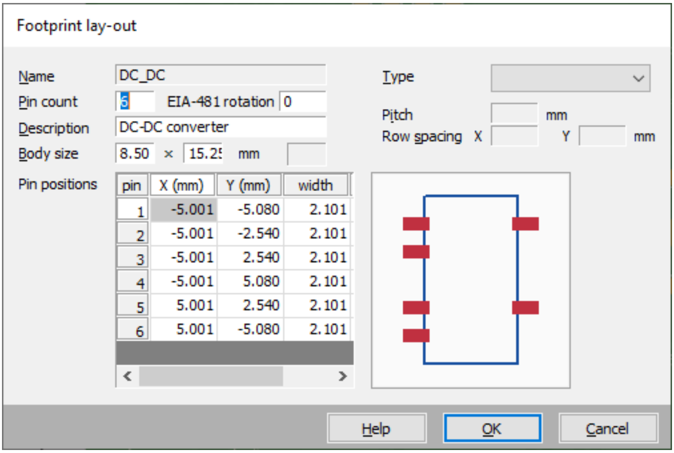


FIGURE 26: *Creating the lay-out of the pins/pads of a footprint.*

of pins. Then choose the template ('Type' in the dialog) for the general footprint shape and fill in the pin pitch and possibly the horizontal and vertical spans.

- ◇ For a 2-pin footprint, only the pitch is needed.
- ◇ For a dual-row footprint (SOIC, TSSOP, ...), the pitch and the 'X' span are needed.
- ◇ For a quad-row footprint, the pitch and both the 'X' and 'Y' spans are needed.

When changing the fields for the footprint or for the template, the pin/pad table and the footprint preview are automatically updated.

The preview indicates the pads with a dot or a rectangle. VisualPlace only relies on the centre of a pad; the size and shape of the pads are only for a better visual appearance only.

The body size for a component does not need to be precise; VisualPlace uses it to only improve its heuristics in footprint scanning. VisualPlace furthermore assumes that the centroid of the component (the centre of mass), is the mid-point of the body.

CompuPhase also distributes the program Packages. Packages is a tool to describe a component package and its footprints in detail. The collection of predefined footprints that VisualPlace comes with, is created with this program. The alternative to describing a new footprint inside VisualPlace, is to create it in Packages and export it to the `packages.json` file that is part of the VisualPlace distribution.

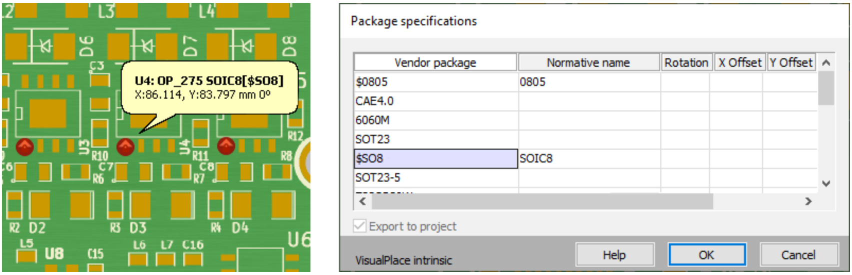


FIGURE 27: *The call-out shows the original footprint name '`$SO8`', linked to the normative name '`SOIC8`'. Both the position and the orientation of package `$SO8` should be corrected.*

Adding a component placement correction

An EDA suite may use a different *normal* orientation for some footprints that those in appendix B. Likewise, the 'origin' of a footprint in the EDA suite's part libraries may not map to the correct centroid position for a pick-&-place machine.

When loading a project, VisualPlace automatically adjusts rotations and positions based on tables with 'known corrections'. These tables are specific to each EDA suite. VisualPlace allows you to extend (or modify) these tables.

To add a correction, open the dialog below Tools / Package specifications. . . (figure 27). The dialog presents a list of packages/footprints that are specific to the current EDA suite. For each row, you can add a mapping to a 'normative name', plus adjustments for the *normal* orientation and the centroid position.

To change an item, select it in the list and enter new values (use F2 to modify a field). Insert a new package by scrolling to the bottom of the list and enter the values in the empty row at the bottom. To delete a package/footprint, erase the package name in the left-most column. When pressing OK to close the dialog, rows lacking a name will not be stored (and thereby, the respective definitions will effectively be erased).

The value for the adjustment in orientation (the 'rotation') is in degrees, counter-clockwise. The values that you enter in the dialog should be positive; if you need a rotation of $-90°$ (i.e. a quarter-turn clockwise), specify the rotation as '270' (degrees). The offset positions for the centroid marker (as it is defined on the footprint) is in mil[1] or in millimetres, de-

[1] 1 mil is 0.001 inch.

pending on the choice in the application settings. The offset is relative to the *normal* orientation of the footprint. A positive value for the 'x' offset will move the centroid marker to the right if the rotation is zero degrees, up if the rotation is $90°$, to the left if the rotation is $180°$, etc.

If a normative name is provided for a package, both names appear in the call-out that appears when hovering with the mouse over a marker —see figure 27. The original package name appears between square brackets behind the normative name. When no normative name is linked to the package, there is only a single package name in the call-out.

The placement corrections are stored in a local data file. When you move the project to another workstation, the corrections should be propagated to this other workstation too —in one of three possible ways:

◇ You can synchronize the files with the placement corrections over all workstations, for example by putting it under version control. The files to synchronize are '`user.ini`' and '`user.glyph`', which are found in either the '`data`' directory below the directory where VisualPlace is installed, or (if VisualPlace does not have *write access* into its installation path) in the *Application Data* directory of the user profile. When corrections are made on several workstations, this may actually be your best option.

  VisualPlace integrates support for Subversion and RCS. One of these systems is therefore the preferred option, if you do not have a version control system already. See appendix F for details.

◇ Export the component placement list after having added and verified the placement corrections —see also page 48. When VisualPlace exports a component placement list, it applies all corrections in the exported file. This is your best option if you need to import the centroid file on a machine or workstation that does not run VisualPlace.

◇ Export the corrections to the *project file*. This is an option in the Package specifications dialog; when selected, the applicable corrections are not only written to the local data file, but also to the project file. When a project is read in on another workstation, the corrections in the project take precedence over the local data files.

  When the placement data is in VisualPlace's *intrinsic* format, corrections are always stored in the project file, as these cannot be stored as format-specific corrections. You can still export the component placement list (a second time) to *apply* the corrections (see the previous bullet point).

APPENDIX A

Interfacing with other applications

Special applications exists for maintaining a bill-of-materials —though most people do it in a spreadsheet program. VisualPlace also offers functionality to edit a bill-of-materials and it offers specific features to make the process less error prone. For example, edits in the bill-of-materials are automatically synchronized with the component placement list. The quantities of parts of a particular value and a particular package are automatically maintained, and components that are marked as 'do not mount' are excluded from the quantities.

However, VisualPlace may also lack features that you need —no inventory database is integrated, for example. For some inventory systems, an interface to VisualPlace can be made with a plug-in (see appendix I), but for others, the only way to share data is by exchanging the 'CSV' file for the bill-of-materials.

Editing a VisualPlace BoM in an application

When editing the bill-of-materials, VisualPlace writes the data in comma-separated values format. In handling these CSV files, VisualPlace preserves unrecognized columns. Therefore, an inventory system is free to append columns with additional information, such as stock or pricing details, and these will be preserved when editing the file with VisualPlace. Note that VisualPlace also allows you to *export* the bill-of-materials in a few formats, one of which is CSV. When exporting a file, only the columns that you specify will be written. In other words, unhandled columns are preserved in the CSV file that VisualPlace maintains as part of the project, but they are not copied to a file that you *export* from VisualPlace.

Spreadsheet applications do not typically use the CSV file format as their native format. To properly import and export CSV files, the spreadsheet application should be configured to use the comma as a separator. Depending on the locale, you may need to change the 'Regional and Language settings' when using Excel, in order to change the 'list separator' from a semicolon to a comma. VisualPlace writes the designators, the packages and the values of the components as *text* type (i.e. enclosed in double quotes), but it supports fields *without* double quotes as well.

VisualPlace furthermore requires that the first row of the CSV file contains the column names —a convention that most spreadsheet programs also support. For columns are required, and these must be named `Designators`, `Package`, `Value` and `Quantity` (or `Qty`). These fields should not be erased or renamed (VisualPlace will re-insert them if they are missing).

When importing CSV files in Excel, it is advised to use the Import External Data or Get External Data function instead of opening the file. Importing the file via the 'Text Import Wizard' presents a dialog that allows you to mark selected columns as 'text'—otherwise Excel may interpret packages like 0805 as numbers rather than text, and drop the leading zero. LibreOffice Calc always displays a dialog for setting CSV options and column configuration.

When using equations in cells, the spreadsheet application must furthermore be configured to export the *equations* of the respective cells, instead of the displayed values.

- ◇ Microsoft Excel writes into the CSV file, what it displays in the sheet. Hence, the solution is to display the formulas in the worksheet cells before saving the file. For recent versions of Excel, click the 'Microsoft Office button', then Excel Options. In the dialog, choose Advanced / Display options for this worksheet. Place a tick mark in the option Show formulas in cells instead of their calculated results.
- ◇ For LibreOffice Calc choose File / Save as... and in the file dialog, place a tick mark in the option Edit filter settings. After clicking OK (and possibly after clicking away the warning message that the CSV format has its limitations), you will get a dialog for the filter settings; in this dialog, tick the option Save cell formulas instead of calculated values.

Importing a CSV file into a BoM

When using an 'inventory system' or other application that can generate CSV files, an alternative is to import these CSV files into the bill-of-materials that VisualPlace maintains. To import a file, open the bill-of-materials editor below Edit / Bill of Materials... Then click on the Import button. Before importing a file, you may need to adjust the 'user field' settings in VisualPlace and the CSV export settings in the other application.

VisualPlace can import from a file or from a URL. When importing from a URL, the server must return the data in a plain text format that is compatible with the CSV format.

VisualPlace requires the CSV file to have column descriptions on the first row. The columns '`Value`' and '`Package`' must be present in the CSV file, because VisualPlace uses these fields to match the records in the CSV files to the bill-of-materials. If a column for the reference designator list is present, VisualPlace uses it as well, for matching rows of the imported CSV file to the rows in the bill-of-materials.

VisualPlace only imports fields from the CSV file that are defined as 'user fields'. For example, if you wish to import the column 'Stock' from a CSV file, one of the user fields must be set to '`stock`'. See page 24 for setting the user fields.

Several plug-ins for generating machine-specific CPL files or assembly 'programs' can generate a feeder report. These reports are in a CSV format. Machine-specific data, such as the feeder numbers, can therefore be imported into the bill-of-materials.

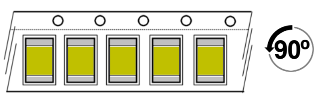
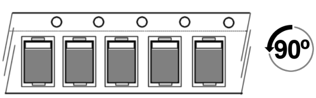
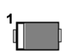
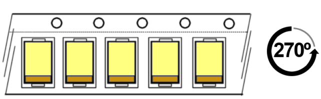
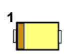
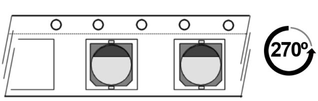
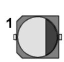
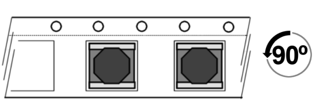
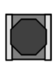
APPENDIX B

Normal orientations of components

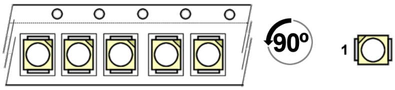
The normal orientation is the orientation that the component has when its rotation is zero degrees. There are two predominant standards for the *normal orientation* (or ‘zero-orientation’) of components on a PCB. Fortunately, these two standards, IPC-7351 and IEC 61188-7 ‘Level A’, are compatible with each other.[1]

Most surface mount components come packaged in tape, on reel. A separate standard exists for how the components are packaged in carrier tape, EIA-481. Manufacturers generally adhere to the EIA-481 standard, but many also specify *exceptions* for specific component packages; for example, many manufactures have continued to package QFP and QFN packages according to the rules in EIA-481**C** instead of adopting the changes of EIA-481**D**.

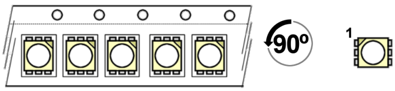
The table below shows the normal orientations according to IPC-7351 and the packaging orientation according to the latest EIA-481 standard (EIA-481D).

EIA-481D	**IPC-7351**	**Notes**
		Non-polarized chip or moulded packages, e.g. resistors, capacitors, inductors, crystals.
		Polarized chip or moulded packages (except capacitors, see below). For diodes, pin 1 is the cathode.
		Tantalum capacitors or other capacitors in moulded packages. Pin 1 (with a bar) is the positive pole.
		Electrolytic capacitors. Pin 1 is the positive pole. (The black bar marks the negative pole.)
		Power inductors (non-polarized).

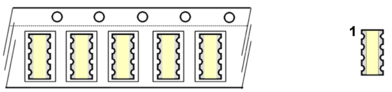
[1] There also exists an IEC 61188-7 ‘Level B’ standard, but ‘Level A’ is predominant.



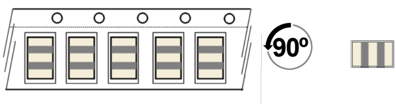
PLCC2 package (for LEDs). See also the notes below.



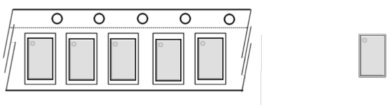
PLCC4 and PLCC6 packages (for multi-colour LEDs). See also the notes below.



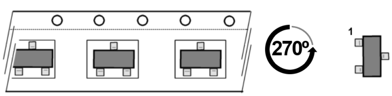
Resistor arrays and capacitor arrays in a chip package.



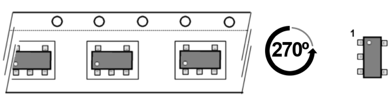
Ceramic resonators with 3 pins. See the notes below for crystals and oscillators with two or four pins.



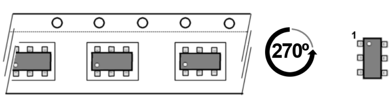
Rectangual 4-pin crystals. See the notes below for crystals with two pins, and oscillators with a square shape.



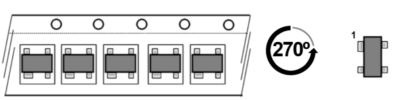
3-pin SOT style transistor, e.g. SOT23, SOT323, SOT523, SC70, TO-236.



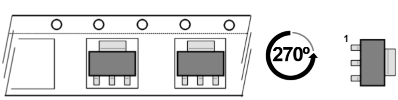
5-pin SOT style packages, e.g. SOT23-5, SC70-5, SSOP-5.



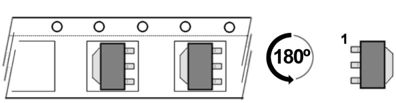
6-pin & 8-pin SOT style packages, e.g. SOT23-6, SC70-6, SSOP-6.



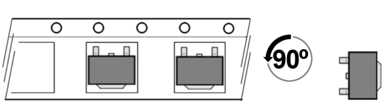
SOT143 and SOT343.



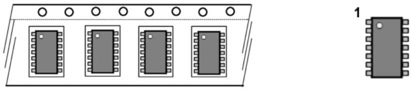
SOT223, SC73, TO-89, TO-261. (See below for similar packages.)



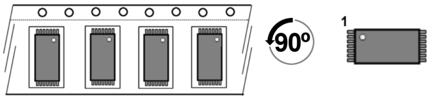
SOT89, TO-293. (See above and below for similar packages.)



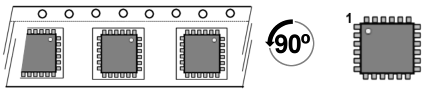
DPAK D2PAK, TO-220, TO-252, TO-263. (See above for similar packages.)



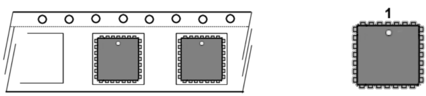
Rectangular IC packages: SOIC, SSOP, TSSOP, TSOP type 2 (but see below for TSOP type 1).



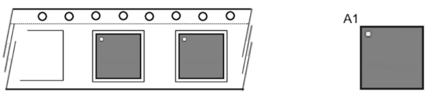
TSOP type 1.



Square IC packages: QFP, QFN, SON, DFN (but not BGA and most PLCC, see below).



Square IC packages that have pin 1 in the middle of the row, which is common for PLCC packages.



Square Grid array IC packages (BGA, PGA); see above for QFN, QFP, and other square IC packages.

Notes

• PLCC2, PLCC4 and PLCC6

The low pin count PLCC packages for LEDs have a chamfered corner to mark pin 1. For PLCC2, pin one is usually the cathode (conforming to IPC-7351); for RGB LEDs in PLCC4 and PLCC6 packages, pin 1 is typically the common anode.

The PLCC4 package uses a clockwise pin numbering scheme, starting from pin 1 in the upper left corner. The zero-orientation is not explicitly defined in the IPC-7351 standard (IPC-7351 assumes a counter-clockwise pin numbering).

The PLCC6 package uses a counter-clockwise pin numbering scheme, similar to dual-line IC packages.

• Crystals, ceramic resonators, oscillators

Crystals in a 2-pin 'chip' package follow the orientation of other 2-pin non-polarized chip packages (like MLCC capacitors). Ceramic resonators with three aligned pins are oriented as shown in the table.

For crystals and crystal oscillators with four pins (or more), the orientation in tape is inconclusive. Rectangular shaped crystals generally follow the alignment as shown in the table. Square packages (which are typically oscillators, rather than crystals) regularly use the orientation of PLCC4.

4-pin Crystals and oscillators may use a clockwise pin numbering, like PLCC4. For crystals, the pin numbering direction is mostly irrelevant, because the crystal is typically connected between pins 1 and 3 (with pins 2 and 4 either unconnected, or both connected to ground). Regardless of the direction of the pin numbering, pins 1 & 3 are at the same positions.

• Tube and tray orientations

Components are oriented sideways in tube (sticks), such that the pins of consecutive components do not touch. The end of the tube that 'pin 1' of the component points to is often marked with a green stop or pin.

JEDEC Trays have a chamfered corner. If the tray is oriented such that the chamfered corner is on the upper left, the components in the tray are oriented according to IPC-7351.

APPENDIX C

EDA suite application notes

VisualPlace comes with parsers for various EDA suites. The specific format for component placement lists and bill-of-materials files differs per EDA suite and sometimes per version of an EDA suite. Regardless of the differences between EDA suites, VisualPlace comes with three 'generic' file parsers for the component placement list and the bill-of-materials.

- A general 'comma-separated values' (CSV) parser can read the bill-of-materials and the CPL files created by several EDA suites. This CSV parser supports files where the fields are separated with commas or semicolons, and that have a descriptive header on the first row.
- The 'XLS' format is a Microsoft Excel file format, that is now also produced by several other (spreadsheet) programs. The VisualPlace parser for the XLS files requires a descriptive header on the first row. The 'Open XML' file format (extension '`.xlsx`') is *not* supported.
- The new Gerber 'X3' format for component placement data aims to be a vendor-neutral standard for component placement lists. The specification is elaborate enough that a separate bill-of-materials is usually not needed.
- Alternatively, some EDA suites are able to generate files in the GenCAD format. Although the GenCAD was originally intended for inspection and testing, it contains all the information that is needed to extract component placement. Typically, the GenCAD file contains sufficient information for the bill-of-materials, too.

Allegro Design

Allegro Design is a product of Cadence Design Systems, Inc. for Microsoft Windows and Linux, aimed at the high-end market. A lower-cost version (with limited features) of the Allegro software is sold as OrCAD.

To create a component placement list, select the menu item Tools / Reports. In the dialog that appears, double-click on Component Report in the top list, to add it to the 'selected reports'. Then, enter a filename for the report, ensure that the Write Report option is ticked, and click on the Report button.

The VisualPlace plug-in works with the default configuration of the component report. A bill-of-materials is not needed when using one of the default report files for the component placement list.

Altium Designer (Protel)

With its first version released in 1985, Protel is one of the oldest EDA suites. At the turn of the century, the company started to focus more on FPGA design and renamed its product (and itself) to Altium, to distinguish it from the earlier PCB lay-out suite.

• Set the board origin

For Altium, it is recommended to set the 'relative origin' to the lower left corner of the PCB and to generate all output files according to this relative origin.

Protel 99SE always generates the Gerber files according to the absolute origin (while the drill file and the component placement list use the user-defined origin). Setting a relative origin is therefore *not* recommended.

• Create Gerber files

In Altium, choose File / Fabrication Outputs / Gerber files. Altium often puts the board profile (contour) in the 'KeepOut' layer or the 'Mechanical 1' layer (standard Gerber extensions '`.gko`' and '`.gm1`' respectively). In the Gerber set-up dialog, choose the coordinates to be relative to the 'relative origin' under the Advanced tab.

For Protel 99SE, generate the Gerber files from the CAM Manager: choose File / CAM Manager from the PCB editor menu.

• Create the component placement list

For Altium, select File / Assembly Outputs / Generate Pick&Place File in the PCB Editor. For Protel 99SE, use the CAM Manager to write the component placement list. Select either 'text' or 'CSV' format; you can choose either metric or imperial units.

Protel and older versions of Altium Designer write *three* coordinate pairs for the component location: a calculated middle point, a reference/origin point and the coordinate of 'pin 1' of the component. VisualPlace uses the coordinates for the reference point.

For Protel and Altium Designer up to version 15.1, we suggest to set the 'comment' property to '=VALUE' for all components. With this setting, the generated component placement list contains all information that is required by VisualPlace, and no extra bill-of-materials needs to be generated.

Starting with version 16, however, Altium Designer uses a different file format for the component placement list, which lacks a 'comment' column. Adding a separate bill-of-materials file is therefore recommended.

CadStar

CADSTAR is a schematic capture and PCB lay-out program distributed by Zuken.

All manufacturing output is created as a batch process, of which various templates are available. To create the Gerber files and the other production files, choose Manufacturing Export from the File menu, and then choose Batch Process. . .

In the dialog that opens, open a 'PPF' template that is suitable for the project, or create one. The templates are in turn based on 'Report Generator Files'. VisualPlace has been tested with the default report files for the component placement list: 'placement.rgf' and 'placement2.rgf'. CadStar creates a single file for component positions on both sides of the PCB. The through-hole components are also included in the file.

A bill-of-materials is not needed when using one of the default report files for the component placement list. If one is desired, VisualPlace supports the default '`<Parts List>`' format for the BoM.

CircuitStudio

CircuitStudio is essentially a feature-stripped version of Altium Designer, distributed by Altium Limited.

• Create the component placement list

From the main menu, select Project / Project Actions to open the Generate output files dialog. In the section Assembly, set a tick mark on 'Generates pick and place file'. In the configuration options for these files, it is recommended to generate the CPL file in 'CSV' format —see figure 28.

• File selection in the project

CircuitStudio creates a file with the extension '`.Outline`' for the PCB profile, but VisualPlace fails to automatically detect this file on the creation of a new project. Instead, VisualPlace typically selects the file with the '`.GKO`' extension for the profile. When creating a new project, you will therefore usually need to override the default selection of VisualPlace by the appropriate profile Gerber.

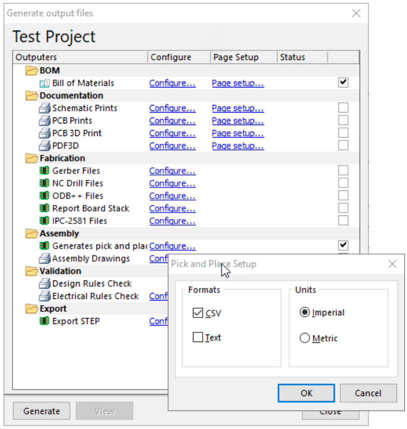


FIGURE 28: *CircuitStudio options for component placement lists*

DipTrace

DipTrace is a schematic capture and PCB lay-out program distributed by Novarm.

• Set the board origin, create Gerber files

For DipTrace, the board origin should be set to the *lower left* corner of the board. To set the board origin, select the menu item View / Define Origin or use the toolbar button. When generating the Gerber files, also put a tick mark in the Use Design Origin option.

See the left dialog in figure 29 for the settings.

• Create the component placement list

In the 'PCB Layout' program, select Export from the File menu and then Pick and Place... VisualPlace requires at least the the columns RefDes, Mid X, Mid Y, Side and Rotation columns —these are present in the default settings for the 'Pick and Place' report.

For the Component Coordinates, select the option By Component Center. The component origins are frequently not well defined in DipTrace libraries. Make sure that the options Add Header and Use Design Origin are both selected. The report should be exported to file, in 'Excel CSV' format.

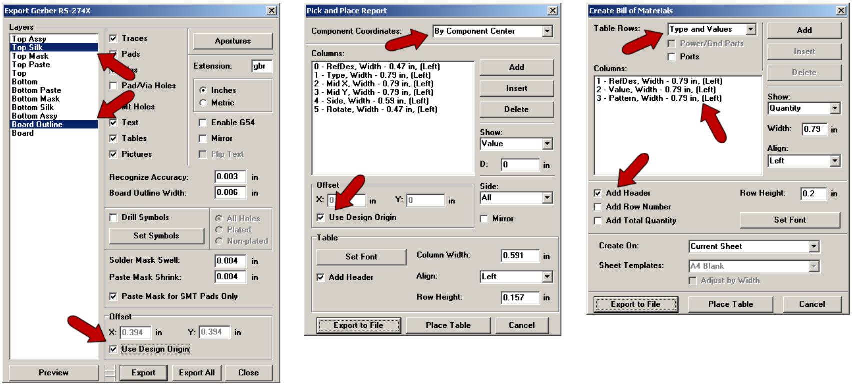


FIGURE 29: *DipTrace export: Gerber (left), component placement (middle), bill-of-materials (right)*

Beware of commas in any of the fields, notably in the Value field. DipTrace uses a comma as the field separator and it does not 'quote' the fields; therefore, if the value field contains a comma (e.g. as a decimal separator, common in Europe), the meaning of the comma becomes ambiguous: 'part of the field' or 'field separator'. One option is to not export the Value column in the 'Pick and Place' file (VisualPlace does not require it).

See the middle dialog in figure 29 for the settings.

• Create the bill-of-materials

In the 'Schematics' program, select Bill of Materials... from the Objects menu. The bill-of-materials report does not define any columns by default, so these have to be added. VisualPlace requires that you define (at least) the columns RefDes, Value and Pattern.

Make sure that the Add Header is selected. For convenience, set the Table Rows selection to Type and Values. The bill-of-materials should be exported to file, in 'Excel CSV' format.

See the right dialog in figure 29 for the settings.

EAGLE

The Autodesk EDA suite 'EAGLE' (formerly from CadSoft Computer) used to be a popular entry-level product, because of its low price and its extensibility using scripts. It is now part of the 'Fusion 360' bundle.

• Create Gerber files

The silk-screen layers for the top and bottom of the PCB are called the 'tPlace' or 'bPlace' layers in EAGLE. It is recommended to plot the PCB profile as a separate Gerber; the profile is the 'Dimension' layer. For solder mask Gerbers, plot the 'tStop/bStop' layers or the 'tCream/bCream' layers. You will find the Gerber export below File / CAM processor. For the format, select `GERBERS_RS274X`.

EAGLE sets the PCB origin to the lower left corner, but VisualPlace can extract the origin from the generated Gerber files. There is no need to adjust the board origin.

• Create component placement list

EAGLE does not produce CPL and BoM files by itself, but requires a 'user language program' (ULP) to do so. VisualPlace comes with a suitable ULP to generate a CPL file that contains all information that it needs. The use of this ULP is preferred, as several other ULPs for 'position data' have various limitations —such as a difficult to parse lay-out, the omission of a descriptive header, or an unspecified unit of measurements...

In the EAGLE *Layout editor*, select the ULP icon on the upper tool bar (or the menu item File / Run ULP...). Then select the ULP '`visualplace-smd`' for the generation of the component placement list. The ULP for VisualPlace is installed in the '`data`' sub-directory below the directory where VisualPlace is installed. You may find it convenient to copy the VisualPlace ULP to to the standard EAGLE ULP folder.

• Create the bill-of-materials

When using the VisualPlace ULP for the component placement list, no BoM file is needed. The component placement list already contains the information that VisualPlace requires.

If you wish to export a bill-of-materials regardless, select File / Export / BOM from the *Schematic editor*, and select values for the 'list type' and CSV for the 'output format'.

Easy-PC / DesignSpark PCB

Easy-PC is a schematic capture and PCB lay-out program by Number One Systems (part of WestDev Ltd.). DesignSpark PCB is a derivative of Easy-PC, distributed as free software by RS-Components.

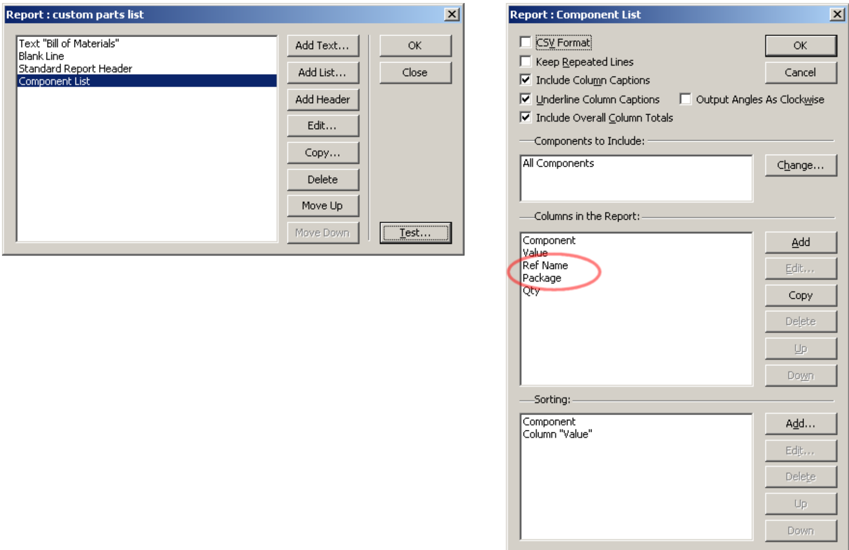


FIGURE 30: *Easy-PC / DesignSpark: adjusting the bill-of-materials report*

• Setting the board origin

For Easy-PC (and DesignSpark PCB), the board origin should be set to the *lower left* corner of the board. The origin of a board can be set through a right-click menu: click with the right mouse button on the PCB and select Set System Origin At Cursor. The program also allows you to set a relative origin, but for the fabrication files, the system origin is the one that is relevant.

• Create the component placement list

From the Output menu, select Reports..., and then the report 'components position' (section 'user reports'). Note that there is also a 'components position csv' report, but that report lacks a header that VisualPlace needs to identify the file.

• Create a bill-of-materials

From the Output menu, select Reports..., and then the report 'custom parts list' (section 'user reports'). Note that there is also a 'Bill Of Materials' in the 'Built-in Reports' section, but that report lacks the package (footprint) column. Similarly, the 'custom parts list csv' report lacks a header that VisualPlace needs to identify the file.

The standard 'custom parts list' report does not have a column with the package information. VisualPlace needs this column. So, before generating the bill-of-materials for the first time, select the report and click Edit. . . , then select the line 'Component list' and click Edit. . . again —See the left dialog in figure 30. In the columns of the report, add the field 'Package' and move it above 'Qty' (because the 'quantity' column is right-aligned, and in the Easy-PC/DesignSpark format, right-aligned columns must come behind all left-aligned columns).

Europlacer

The Europlacer '`.dp`' files contain the instructions for placement for their iineo, XPii and Flexys series.

To use a '`.dp`' file for the placement data in VisualPlace, it must be accompanied with a bill-of-materials that includes the value and the package of each component. Also required are Gerber files or pictures of the PCB. These files must be generated by other software (such as an EDA tool), the Europlacer software does not create a bill-of-materials or picture files.

An Europlacer program may define 'patterns' with a sequence of placement instructions that are repeated at different sections on the PCB. A typical use of patterns is for panelizing a PCB. The VisualPlace plug-in will only show each pattern once; it ignores the 'repeat pattern' command. The effect is that, when an Europlacer program is a program for a panel, VisualPlace will only show the placement data for one board in the panel (the other boards in the panel have identical placement information).

KiCad

KiCad is an open-source EDA suite, providing schematic capture and PCB lay-out design.

• Set the board origin, create Gerber files

The default origin of a board is the upper left corner of the drawing sheet; use the 'Offset adjust' tool of the PCBnew program —see figure 31, to set the origin to either the upper left corner or the lower left corner of the *board profile*. The layer for the profile is called 'Edges' in KiCad.

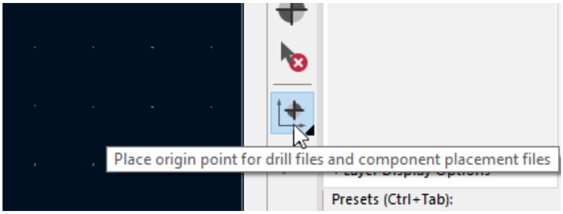


FIGURE 31: *KiCad: the 'Origin adjust' tool of the* PCBnew *program*

When plotting the board, set the Plot Origin to Auxiliary axis (which is not the default). If you omit this step, the component placement list will be relative to the new origin, but the Gerber files will still be relative to the upper left corner of the page.

• Create the component placement list

KiCad refers to the component placement list as 'Footprint Position' files (or 'Modules Position' in older versions of KiCad). To create the file, start the PCBnew program of the KiCad suite and choose the item Fabrication Outputs of the File menu). On recent versions of KiCad, you can select to create a separate file for each side of the board, or a single file for both sides.

On recent versions of KiCad, you can select the format for the CPL file as 'ASCII', 'CSV' or 'Gerber X3'. All three formats are supported, but KiCad's CSV format lacks information on the units for position and angles (e.g. millimetres versus inches). Therefore, either of the other two formats (ASCII or Gerber X3)is recommended.

The Footprint Position file lists only the footprints that have the 'Placement Type' attribute set to 'Surface mount' (in older versions of KiCad, this attribute was called 'normal+insert'). You therefore have to make sure that the components that you wish to appear in the component placement list have this attribute. Recent versions of KiCad also have an option in the 'Generate Footprint Position Files' dialog to set this attribute for all SMD components on the PCB.

• Create a bill-of-materials

The bill-of-materials is also produced by the PCBnew program of the KiCad suite: see the item Fabrication Outputs of the File menu. This file contains the package information (i.e. the footprints), which is lacking from the Modules Position file.

KiCad can also export a bill-of-materials from the schematic capture program (EESchema), but these files do not necessarily contain the package information. We therefore advise to export the bill-of-materials from the PCB editor (PCBnew).

Mentor Graphics

Mentor Graphics provides multiple programs for PCB lay-out. This section covers the PADS suite (PowerPCB).

• Create the component placement list

The PADS lay-out program installs the script 'XY Positions' in the Tools/BasicScripts menu. Running this script generates a component placement list with the name POSITIONSXXXX.TXT in the active directory.

Later versions of PADS replaced the 'XY Positions' script with 'Excel Part List Report', which can be read by the 'generic XLS' parser of VisualPlace.[1] However, the 'XY Positions' script is still available, in the 'Unsupported' folder of PADS. Also available in this folder is the script 'Part List Report' written Kevin Adams (Xerox Engineering Systems), that can also be used with VisualPlace.

• Create a bill-of-materials

The bill-of-materials also has to be written by a script. However, none of the scripts provided with the PADS program supplies all the required information.

The alternatives are to either create the bill-of-materials in VisualPlace (see page 29), or to create it in a spreadsheet program and export it in CSV or XLS formats, so that it can be parsed with the generic plug-ins.

OrCAD

As of version 16, OrCAD PCB Designer is a cut-down version of Allegro Design. It uses the same file formats for the bill-of-materials and the component placement list. VisualPlace detects OrCAD files as Allegro format. See the section on Allegro (page 68) for export options.

---

1 Mentor Graphics PADS uses an OLE connection with Microsoft Excel to generate the XLS file, and hence it requires Microsoft Excel.

Pulsonix

The standard footprint libraries of Pulsonix are quite inconsistent in the zero-orientation of the component. The correction tables for Pulsonix that are provided with VisualPlace are by far not comprehensive (because of the large amount of corrections that would be needed). It is recommended to generate the corrections project for project.

• Create Gerber files

Gerber files are generated through the Output / CAM/Plot menu. There is no option to export a Gerber file for the PCB profile (contour). As a result, the origin of the component placement list must usually be manually aligned to the origin of the Gerber files (see page 12).

• Create the component placement list

To create the component placement list, choose Output / Reports. In the reports dialog, select Pick And Place Csv. The generated file has the components for both sides of the PCB and it includes through-hole components.

• Create a bill of materials

For the bill-of-materials, also choose Output / Reports. In the reports dialog, select Parts List Csv.

Note: generating the reports for the component placement list and the bill of materials must be done in two steps. When selecting both reports in the dialog, Pulsonix combines these in a single file (which is not supported by VisualPlace).

TARGET 3001!

TARGET 3001! is the successor of TARGET 2001!. It is developed and distributed by Ing.-Büro Friedrich. This section applies to the standard versions, the free 'unlimited' version provided by PCB-Pool is *limited* to generating files suitable *only* for the PCB-Pool service, and it cannot be used with VisualPlace.

• Create Gerber files

Generate Gerber files for layer 21 ('placement print top') and, if there are components at the bottom side, layer 7 ('placement print bottom'). To create the files, choose the menu File, then Input / Output Formats, then Production and finally (X)Gerber and drill output.

The default file extensions that Target 3001! gives to the silk-screen plots are '`.PosiTop`' and '`.PosiBot`'. VisualPlace does not recognize these extensions as 'Gerber files'. When browsing for the silk-screen plots (in the project dialog), set the file type to 'All files (*.*)'.

• Create the component placement list

To create the component placement list, choose the menu File, then Input / Output Formats, then Production and finally Pick-And-Place Automat. The dialog allows you to choose whether to include all components or only the SMD components; for pick-&-place machines, and whether to create a single file for both the top and bottom layers or to create separate files for each side.

• Create a bill-of-materials

No bill-of-materials is needed for VisualPlace. The component placement list contains all the required data.

Ultiboard

ULTIboard was originally a DOS design package, by Ultimate Technology. It later moved to Microsoft Windows and was renamed to 'Electronic Workbench'. After being aquired by National Instruments, it was renamed back to Ultiboard.

At some point, the product switched to a new file format for the component placement list. The old format is referenced in this appendix a the 'ULTIboard 5' format.

• Create Gerber files

From the menu File choose Postprocessing. At the choice of plotter/printer, choose photoplotter and subsequently select RS274 or higher.

Then, below Select settings, select to process the 'silkscreen'. The default extension for the generated file is '`.g6`', but you may choose a different extension. With a similar procedure, generate the PCB profile (default extension is '`.g7`') and optionally the solder masks (default extensions '`.g2`' and '`.g3`' for the top and bottom layers respectively).

• Create the component placement list

To create the component placement list, from the menu File choose Export and then Placement. Depending on the version of ULTIboard, the generated files have the extension '`.cmp`' or '`.ctr`'.

When generating component placement lists for VisualPlace, keep the following in mind:

- ◇ For ULTIboard 5, it is important to use the default unit of measurement for ULTIboard (which is *nanometres*). Since ULTIboard 5 does *not* store the unit in the file itself, the plug-in for VisualPlace must *assume* a fixed unit. This does not apply to modern versions of ULTIboard.
- ◇ Export at least the columns: REFDES, VALUE, SHAPE, X_CENTRE, Y_CENTRE and ROT, and in that order. You may add more columns to this list, but put them at the end.

ULTIboard 5 (for DOS and Windows) generates separate files for the components on the top layer and those for the bottom layer, but it does not store the 'side' information in the CPL files itself. ULTIboard 5 also stores through-hole components in the CPL file (top side). Since no flag for the mounting style of component is stored, VisualPlace assumes all components to be SMD components.

Modern versions of ULTIboard do not have the limitations mentioned above: through-hole components are still stored in the CPL file, but are marked as such and 'side' information is stored in the CPL file as well.

• Create a bill-of-materials

The BoM files generated by ULTIboard (DOS or Windows) do not add any relevant data for VisualPlace, so the field for the BoM may be left empty in the VisualPlace project.

APPENDIX D

Camera interface and set-up

VisualPlace can use a live stream from a camera as the background canvas to put the markers on. This enables you to use VisualPlace both as a 'magnifier' during manual PCB assembly as well as a positioning system that tells you where to place the next component.

In other words, instead of constantly switching between looking at the computer screen (for the component's location) and to the PCB itself to place the component, you instead keep looking at the display. The display shows the live action of the component in the tweezers in your hand as well as the marker that indicates where to place it. This way of working is more relaxed, and typically quicker.

Before you can use VisualPlace with a camera, you must first acquire a suitable camera plus a camera stand, and configure both the camera and VisualPlace.

Selecting a camera

The camera should be a 'network camera' (or IP camera). VisualPlace communicates with the camera over the Ethernet interface, using the TCP/IP protocol. Cameras that connect to a USB port are *not* supported.[1]

VisualPlace uses the MJPEG codec. Almost all network cameras support this codec. There is typically a trade-off between image size and image quality on one side and frame rate and latency on the other. A camera with (digital) PTZ has the advantage that it can show a detailed image of a small section of the PCB and still send relatively small images —thereby achieving a high frame rate. PTZ stands for 'Pan, Tilt, Zoom'.

We advice to have a cabled connection between the camera and the workstation. In our experience, WiFi adds to the latency of the video stream.

The camera is best positioned perpendicular above the PCB at a height that it does not block the eyesight. That is, the camera floats roughly 50cm above the table top. This means that the camera should have a long-focus lens, to get enough details at that distance —the advised angle of view is $20°$ to $30°$.

---

[1] The rationale being that USB cameras have a higher latency and lower throughput (frame rate) at high resolutions than network cameras.

The camera stand should be a table-top stand that keeps the camera at 40cm to 60cm above the table. There exist special stands for document scanning, and these are often suitable. The *de facto* standard thread size for camera mounts is 1/4-20 UNC, but manufacturers of network cameras sometimes follow their own standards instead.

Setting up the camera

Setting up a camera for VisualPlace is a three-step process —or a four-step one if you have not installed the cabling between the camera and the workstation already.

To start setting up the camera, open the dialog below View / Camera set-up.

• Step 1: connecting the camera

The IP address (or domain name), and the user name and password for the camera must be given, and then the camera must be configured. Most network cameras do not use domain names. Most cameras are configured to use DHCP out of the box, but a fixed address is advisable.

VisualPlace only supports 'basic authentication', which is also what most cameras support. You may, of course, configure the camera to work without authentication. Security is typically not important for the connection between VisualPlace and the inspection camera.

Once the IP address and authentication parameters are filled in, you can click on the button Adjust camera to configure the camera itself. VisualPlace launches a web browser with the camera address to let you configure the camera from the camera's own integrated web server. Properly configuring the camera is the most important step.

One particular setting that may need to be adjusted is whether the image of the camera must be rotated or flipped. This is easy to verify in the preview of the camera that the web browser will show.

When using a PTZ camera, you should disable the default functionality of the camera to zoom back to overview more after a configured time. The reason is that the VisualPlace application is not informed when the camera decides to zoom back, and it may then assume the wrong zoom level and/or focus position.

• Step 2: connecting the video and control streams

To show a video stream, or control it, two paths must be filled in: the path for the MJPEG stream and the path to control the camera. Some cameras will tell you the paths to use on their integrated web server, for others, you need to look it up in the camera manual. The default values that VisualPlace puts in these fields are valid for AXIS cameras.

If a camera does not support PTZ, the field for the Control path should be left empty.

The button (Re)connect button allows you to test the new settings.

• Step 3: calibrating the scale

After completing the first two steps, images from the camera should now appear in the dialog window. VisualPlace draws a graticule, a grid of red lines, on top of the camera image.

Just above the area with the camera image are a slide bar to calibrate the scale of the video, and a check button for switching between zoomed mode and overview mode. The zoomed mode is only available for a PTZ camera. The Control path for camera commands must also be filled in before the check button is functional.

To calibrate the camera, put a sheet of graph paper below the camera. Set the graticule to the appropriate interval. VisualPlace comes with a PDF file of a 10mm grid, as shown in figure 32; you can find it in the '`camera`' sub-directory below the path where VisualPlace is installed. When printing the image, verify that the grid is exactly 160mm wide and high; common (laser) printers have limited accuracy and a 600 dpi printer may turn out to print 597 dots in an inch —a 0.4% deviation. Most PDF readers allow you to fine-tune the scale of the print-out.

The Scale track bar allows you to move the scale up and down. This action changes the dimensions of the graticule (the video image stays the same). For a PTZ camera, you will have to adjust the camera in normal ('overview') mode and in zoomed mode.

The Barrel track bar allows you to correct for barrel & pincushion distortion in the camera.[2] When you put a sheet of graph paper below the camera and the lines come out as curved in the camera image, this is caused

[2] Barrel distortion occurs when magnification at the edge of the lens is weaker than the middle. It is a common distortion on wide angle lenses.

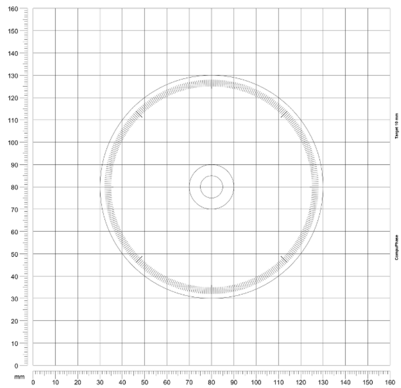


FIGURE 32: *A calibration target for adjusting the camera scale, barrel & keystone distortion.*

by barrel/pincushion distortion. The Keystone track bar corrects for keystone distortion. When the lines are skewed, this is due to keystone distortion.

Note that when adjusting the barrel & keystone correction, VisualPlace deforms the graticule, instead of rectifying the image. The motivation is that rectifying the image is a processor-intensive operation that may aversely affect the camera frame rate and latency. Low latency is of more importance than a faithful image.

The adjustments for barrel and keystone correction should be done in normal ('overview') mode —*not* in zoomed mode).

After adjusting the scale, barrel and keystone settings, you can verify the accuracy by loading the 'Camera Test' project that is also in the '`camera`' sub-directory. This test project has dummy components placed at a 10mm grid —matching the calibration grid.

Other settings and options

It is advised to use the large 'open style' for the markers that VisualPlace uses to indicate the position of the components. You may want to lower the opacity of the markers as well.

Especially when using the large 'open style' markers, the markers may overlap for closely spaced components. It may be more practical during assembly to activate the option to automatically expand the component list, so that only a single component is marked at a time —see menu View / Auto-expand component list.

To move through the component list, we advise to acquire and set up a foot switch. A single foot switch, to move down, is sufficient to keep your hands free from operating VisualPlace for most tasks. A dual foot switch, to move both up and down, may prove practical at times.

Supporting other cameras

We lack the resources to test VisualPlace with all existing cameras. For plain network cameras without PTZ-capabilities, the path to the MJPEG stream is most likely sufficient for VisualPlace, as network cameras generally conform to the multi-part HTTP streaming protocol.

Cameras with PTZ capabilities use brand-specific command sets. Currently, the VAPIX command set from the AXIS cameras is assumed. To support other cameras, you must configure the command strings in the INI file of VisualPlace. In particular, you should add (or adjust) the following section to the file `VisualPlace.ini`:

LISTING:

```
[PTZ-commands]
zoom-at=center=%X%2C%Y&imagewidth=%W&imageheight=%H&zoom=9999
zoom-out=zoom=0
```

The strings behind the keywords `zoom-at` and `zoom-out` must be adjusted to your camera. The strings, `%X` and `%Y` will be replaced by the pixel coordinate that should be in the centre of the zoomed picture, and `%W` and `%H` are replaced by the frame width and height. Furthermore, for most cameras, URL-encoding applies, so special characters must also be encoded appropriately.

APPENDIX E

Scanning PCBs and text recognition

VisualPlace can scan text from a silk-screen image. This is helpful for cases where the Gerber files for a PCB are available, but the centroid files are lacking or incomplete.

After completing a scan, VisualPlace may present you with a list of designator labels that it has detected on the silk-screen, but that are not present in the bill of materials. Possible causes are:

- The bill of materials is indeed incomplete. In this case, you can place a check-mark in front of the designator label. The designator will then be added to the bill of materials when you click Apply.
- The OCR engine has misdetected a glyph —for example, it has read `R1B` instead of `R18`. The designator label should then just be corrected. You can select the field in the table and correct the name. There is no need to place a check-mark in front of the corrected name. However, you must click Apply to validate all corrections.
  Note that when glyphs are regularly misdetected, you may need to re-train the OCR engine (the next page).
- The OCR engine has misdetected a footprint outline or other graphics on the silk-screen as text. These items must simply be skipped (do not place a check-mark in front of these 'labels'). Again, if this happens regularly, you can reduce such misdetection by training the OCR engine.

If you click Apply and at least one new designator was added, a new dialog will open with the bill of materials editor.

Scanning centroid positions

When the reference designators on a PCB have been collected, VisualPlace can subsequently scan for footprint outlines in the vicinity of the text label. This operation is in the menu Tool / Acquire placement data from scan. This operation *only* adds centroid positions for those items for which no placement data is known. The newly added centroid positions must still be visited to specify their orientation. Also note that the legends on a silk-screen may not be unambiguous.

There are several factors that affect the success rate of the position scanning. An unambiguous, silk-screen with the legends clearly laid out is, of course, essential. Making sure that all components have well-defined

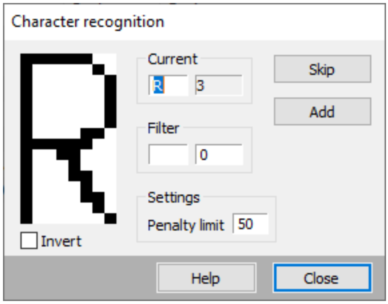


FIGURE 33: *Training the character recognition*

footprints also improves the accuracy of the scan —see also the topic Package/footprint names: vendor vs normative, page 55

Training recognition

Many EDA suites use a vector font that is a variation of the Hershey Simplex font. The optical character recognizer (OCR engine) in VisualPlace has been 'trained' with several variations of the Hershey Simplex font. When the engine frequently mismatches symbols, this is usually due to the use of a font that has different metrics than those in the OCR tables. To improve recognition, it is then necessary to 'train' the new font. The same is also needed for recognition of characters that are not yet in the tables. The VisualPlace tables only include the digits and the upper case letters.

After activating training mode, VisualPlace will prompt you for each glyph that it extracts. It gives you the processed bitmap extracted from the silkscreen, the character that it has detected and the 'penalty' for that detection. The lower the penalty, the stronger the match is. You then have the option to:

- ◇ Skip the glyph, by which you tell the OCR engine that you agree with the match.
- ◇ Add the glyph to the tables, after correcting the character if needed.
- ◇ Cancel further recognition.

There are two reasons for adding a glyph to the OCR tables. First and foremost, if VisualPlace has misdetected a character, you should correct it and add it. In addition, if VisualPlace has detected a character correctly, but with a high penalty, it may also be useful to add it, to improve the detection accuracy of other instances of the same character.

However, care should be taken not to overdo the training. A penalty below 10 means that VisualPlace considers it a strong match. If it is also correct, then, as a general rule, you can skip it. Adding shapes/characters pairs whose match penalty is low, grows the tables and decreases recognition speed without improving the accuracy of recognition.

Text may be placed on a PCB horizontally and vertically, and even upside down. VisualPlace scans the glyphs in four directions. It may on occasion present an upside-down character as the closest match. The same reasoning applies: if VisualPlace is correct and the penalty for the detection is low, you can skip the glyph; on the other hand, if VisualPlace is *not* correct or if the penalty is high, you should correct the detected character *and* click on the Invert check-mark to correct the orientation before adding the glyph.

For quicker training, the dialog lets you set a filter on what glyphs you wish to train. You can filter on:

- ◇ A character: in which case you will only be presented with shapes that are detected as that specific character.
- ◇ A minimum penalty: in which case all glyphs that are detected with a lower penalty are automatically skipped.

On a frequent specific mismatch —for example, when the digit '5' is regularly detected as letter 'S', you may want to set the filter to the letter 'S' and then train only the (misdetected) character '5'. On training a new font in general, you may want to set the minimum penalty to 10 (for example) and review/train only the characters that have a higher penalty on detection.

VisualPlace attempts to reject graphics on the silk-screen image that are clearly not characters, but small-shaped footprints may slip through this filter. There are two options to reduce the *false positives*:

- ◇ Most misdetected graphics are 'recognized' with a high penalty. You can adjust the Penalty limit in order to automatically reject any shape with a penalty exceeding this limit.
- ◇ For specific markings, you can train the OCR engine to detect these shapes as 'not a character'. You do so by clearing the character before pressing the Add button.

APPENDIX F

Using Version Control

Two main purposes of a version control system are to maintain a history of all changes in the design files of a project, and to manage the changes to those files when multiple people may be working on the same project. Version control helps keep all users stay up to date, and offers users a local environment in which they can freely experiment and work on the files, without breaking the work of the other members of the team.

As is clear, version control is indispensable in a collaborative work environment. But there are also advantages using it when you are the only one to work on a particular project. For ensuring the quality of a project, it is often required to have a log of the changes, which is what a version control system provides. An additional advantage is that version control systems inherently keep a back-up of the data.

VisualPlace has built-in support for two version control systems: *Subversion* and *Revision Control System* (RCS). Subversion uses a client-server architecture with a central repository. It is a good choice for teams. RCS is a local version control solution, which does not require a server or repository, and which is straightforward to set up. It is an adequate choice when working alone on a project.

Note that you can use other version control systems with VisualPlace as well. Subversion and RCS have the advantage that you get notifications of updates and local changes from within VisualPlace, and that you can synchronize these updates and changes with VisualPlace as well. If you already have a version control system, we recommend that you keep using that; however, if you are not yet using version control, we recommend either Subversion (when working in a team) or RCS (when working alone).

Setting up the software

Subversion

Subversion is a client-server system. You need both a client and a server. The server holds the 'repository', which is the collection of all data files, plus the complete history of the data files. The Subversion server can be on some central computer that the other workstations have access to. Alternatively, you can use a hosting service for Subversion (both free and commercial).

It is common to install the Subversion server as a module of the Apache web server, but this is not required. Subversion comes with the `svnserve` program —a stand-alone server. On Linux and related systems, svnserve can be started from a boot script (in '`init.d`'); on Microsoft Windows, it can be added as a service.

On each workstation on which you run VisualPlace, you should also install the Subversion client. The version that VisualPlace needs is the 'command line client'. Most GUI clients for Subversion also include the command line tools, but you may need to explicitly install them. For example, TortoiseSVN, a popular client for Microsoft Windows, has by default the option for the command line tools *deselected*.

After setting up the software, you have to create a repository on the Subversion server and create folders in that repository. For this, you are referred to the Subversion manual. One of these folders should be for the 'user data' files that VisualPlace maintains. These files contain both package definitions that you created or modified, package name mappings (from EDA-specific to generic names) and glyph training data for OCR.

RCS

The RCS tools only need to be extracted to the harddisk. There is nothing to configure or set up.

RCS is a bare-bones system, and that shows in the software tools for managing the 'version files'. Most are plain and simple programs, but perfectly adequate for a single developer working on a project. A GUI client for RCS must be separately installed —the distribution for RCS does not include one.

File comparison utility

It is quite helpful to review the changes made to a file in each step that you commit to version control. In order to do that, install a 'file comparison' utility (or 'file differencing' utility). These kinds or programs show you the differences between two files, or (in our case) the differences between two revisions of a file. If you use TortoiseSVN, you can use its file comparison tool 'TortoiseMerge'. Another popular utility is WinMerge. Both these are freeware.

VisualPlace

In VisualPlace, version control must be configured in the menu Tools / Application settings. This dialog has a TAB page for version control settings, see section Configuration on page 21.

After you have selected either Subversion or RCS, VisualPlace tries to locate the location of the software in the Windows registry keys or in the path. If the path is not automatically set, you can browse for the system yourself.

In the path for the file comparison utility, you may add command line options behind the path. In the options the sequences '%b' and '%w' will be replaced by the filenames of the 'base' version of the file and the 'working copy' of the file (the working copy is the current version of the file, on the local drive). Adding options or the '%b' and '%w' sequences is optional with most file comparison utilities.

Putting files under version control

You use the client of your version control system to place files under version control. The particular procedure depends on the version control system: in Subversion it is an 'add' command; in RCS you perform an initial 'check in'.

Apart from the files that are part of a VisualPlace project (notably the bill-of-materials, component-placement list, and the project file itself), you may also want to place the configuration data files under version control. Specifically, it concerns the files '`user.ini`' and '`user.glyph`'. These are found in either:

- the path set in the 'Path to data files' option in the Application settings dialog (page 21);
- the '`data`' directory below where VisualPlace is installed;
- The 'VisualPlace' directory below the 'Application settings' folder ('Roaming') for the current user.

If the files are there, you must add them to version control. Before you add them, you must place them in the correct local folder (if they are not already there). To add the files to version control, you use the Subversion client.

Synchronizing modifications

A tool to keep the user data files and the production files of the active project synchronized between the workstation and the repository, is in the menu under Tools / Update & commit...

When opened, the dialog shows a list of files that are used in the project. For each of the files, the dialog gives its status in version control. Files that you changed, can be committed to version control; files that were

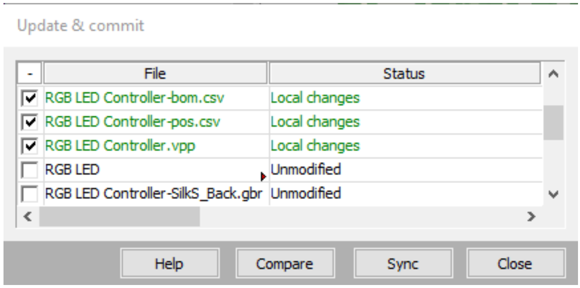


FIGURE 34: *Synchronizing with version control*

updated on the server (by a team member) can be updated. Both these actions are performed with the Sync button. Before synchronizing, you can select which files to apply this to, by placing or removing a tick mark (checkmark) in front of the filename.

For local modifications that are committed to to repository, you are presented a text box in which you can add a description of the modifications. For example, if the production files changed because you marked a few components as 'do not mount', you can note that in the text box. A 'commit comment' is optional, but it is strongly recommended to briefly and specifically describe what changed.

Viewing local modifications

After selecting a file in the list, a click on the button Compare (or a double-click on the filename), opens the file comparison utility with the current version and the base version of the file (the base version is the file as it was before any local modifications).

Watching notifications

While running, VisualPlace checks the status of the production files in the active project plus the user data files. If it detects that there are local changes or that the repository holds updated files, it shows a notification in the bottom-right area of the main view.

APPENDIX G

Menu interface

File menu

New project. . .
: Creates a new project and shows a dialog with the project settings. See page 9 for details.

Open project. . .
: Shows the 'file open' dialog, for opening a VisualPlace project that was created earlier. Archived projects can also be opened (no changes can be made to archived projects, though).

Recent projects
: A submenu with recently used projects, for quick selection.

Archive project. . .
: Stores all files that are referenced in the current project in a 'ZIP' file. This includes the PCB artwork (the Gerber files), the bill-of-materials, the component placement list and the project information file itself.

Export / Bill of Materials. . .
: Writes a new file for the bill-of-materials. The file format for the output can be selected in the Save as. . . dialog.

Export / Placement Data. . .
: Writes a new component placement list that includes any corrections/adjustments made in VisualPlace.

The file format of the exported file can be selected in the Save as. . . dialog. This may either be a format specific to a pick-&-place machine, or a general purpose text file.

When the PCB has components on a single side, only one file is written, with the name that you specify. When the PCB has components on both sides, you can choose to create either a combined file or separate files for both sides. When selecting separate files, the words '-top' and '-bottom' are added to the filename that you choose.

Reports / PCB lay-out. . .
: Starts a PDF viewer with a report of the images of the PCB, with optional rulers and an optional grid. The component designators or their values can optionally be printed on top of the images. For a PCB with all components on a single side, the report has only one page; for a PCB with components on both sides, the report has two pages.

Reports / Bill of materials
: Starts a PDF viewer with a report of the bill of materials. There is no placement information in this report.

Reports / Placement List
: Starts a PDF viewer with a report of the placement information of all components, enriched with information from a bill-of-materials.

Reports / Flipbook for assembly
: Starts a PDF viewer with a report of the images of the PCB combined with the component information and the component placement. The report has a separate page for each group of components with the same value and package (and that are mounted on the same side of the PCB).

Exit application
: Quits the program.

Edit menu

Undo. . .
: Changes that you make to component positions and orientations can be undone. Operations like origin alignment (of the PCB and placement data) cannot be undone.

Bill of Materials. . .
: Opens a dialog to edit the bill of materials of the active project. See page 29 for more information. When VisualPlace saves the edited bill-of-materials back, it will be in its internal format. To store the bill-of-materials in a different format, export it via the File menu.

Placement List. . .
: Opens a dialog to directly edit the component positions for the active project. VisualPlace saves the edited component placement list in its internal format. To store the component placement list in a different format, export it via the File menu.

Fiducials...

Opens a dialog with a table for editing, creating, or deleting fiducial mark positions. It can also extract fiducial positions from the Gerber files and drill file. See Adding fiducials, page 32.

Placement order & flags...

The component list (see figure 5 and figure 8) is sorted on component *category* (semiconductor, resistor, capacitor, etc.) and value. The category is implied by the first few letters of the reference designator. The order of the categories can be adjusted in this dialog. It allows you to list the semiconductors before the passive components, or vice versa. See page 35

Placement zones...

Define zones in the PCB to organize the order in which components are placed on the PCB. See page 37 for more information.

Add/change component placement

Allows you to manually add placement information for a component that is not listed in the component placement list, or to modify the position of an individual component.

Rotation step

The orientation of a component is indicated by a marker, and it can be rotated through a right-click menu on that marker. The rotation step is $90°$ by default, but it may be set to $45°$ via this menu option.

Add/edit note...

Opens a dialog to enter a general purpose note. The same dialog also lets you browse through existing notes, or to delete a note.

Find component...

Allows you to search for a single component by its designator.

View menu

Flip top/bottom

Toggles between the top view and the bottom view of a board with components on both sides.

Call-out with component details
: Toggles a call-out with the full information on the component that is currently selected (in the component list) on or off. See figure 5.

View Camera
: Toggles between viewing the design files (the silk-screen images) and a video stream from a camera. A camera must first have been set up.

Camera set-up...
: Opens a dialog to configure and calibrate a camera for use with VisualPlace; see appendix D.

Sort components
: The component list (see figure 5 and figure 8) can be sorted on value & category, on designator, or on package. Sorting on value/category is the default: it groups all components with the same value and package. Note that when the component list is sorted on designator, the 'number of components per row' is always 1 (as designators are unique) and therefore the *count* column is hidden.

Zoom
: The silk-screen image can be zoomed between 25% and 100%, and it can be made to fit inside the window. The + and – on the numeric keypad (of the keyboard) also allow you to zoom up and down, and (depending on the keyboard) the standard + and – may work as well.

Rotate PCB view
: The visual representation of the board can be rotated in increments of $90°$. Note that functionality for aligning the placement origin to the silk-screen, and for moving or adding component positions, are only available when the image is in its normal (zero-degrees) orientation.

: To rotate the Gerber files *only*, so *without* also rotating the component placement data, see the Project settings dialog on page 12.

Smooth scrolling
: When the silk-screen image does not fit in the window, VisualPlace scrolls components into view after the selection changes in the component list, or after a click in the overview window (see figure 5). VisualPlace can either move to the target location in one quick step, or smoothly in an animated scroll. The

latter option makes it easier to follow the board as it scrolls to the new location, at the cost of slowing the scrolling down.

Auto-expand component list
: The 'auto-expand' mode is only relevant when multiple components are grouped on a row in the component list. That is, when the component list is sorted on designator, this option is ignored. See also page 36.

Navigation button bar
: Shows or hides a button bar at the bottom of the 'board view', with buttons to step through the component list and to change the zoom level.

Show package shapes
: Shows or hides the images of the packages instead of the markers, for standard packages. When VisualPlace does not have a graphic representation is for a package, it shows the marker instead.

Marker, Footprint & PCB visualization...
: Opens a dialog where you can adjust the size, colour and opacity of the markers, as well as the colours for the PCB and pads.

Project menu

New project...
: Opens an assistant to create a new project, based on existing PCB design files and production files.

Project settings...
: Opens the dialog with the settings of the active project.

Align placement origin...
: In case the data in the component placement list is misaligned with the Gerber images, manual alignment is needed. The procedure is described on page 12.

Format/Convert to: VisualPlace
: Enables you to convert the component placement list to the intrinsic format used by VisualPlace (if the format is not already in the intrinsic format).

Format/Revert to: *EDA-specific*
: Shows the name of the EDA suite that generated the (original) component placement data. If the current component placement list is in the intrinsic format, it enables you to revert back to the original files.

Active assembly stage
: Opens a sub-menu to select one of the listed assembly stages as the 'active stage'.

Edit assembly stages. . .
: Opens a dialog for changing and adding assembly stages.

Tools menu

Update and commit. . .
: When the project is under version control, this dialog allows you to synchronize the local copy ('workspace' copy) with the one in the repository —either updating your local copy from the repository, or storing your changes in the repository.

Package specifications. . .
: Opens a dialog for adding a correction for the orientation and/ or the position of the centroid marker for a particular footprint, as well as for package/footprint names to other (normative) names. See page 59 for details.

Verify component placements. . .
: Analyzes the Gerber files to determine the positions and footprint shapes of the components, and compares these with the placement information from the fabrication files. The dialog lists the components for which the scan data mismatches the fabrication data. This dialog may give you an indication of which components may need to be adjusted. See page 45.

Scan labels from silk-screen
: The submenu enables you to scan and recognize texts from the silk-screen image, for the PCB side that is currently displayed. It offers the option for training the character recognition. See appendix E.

Acquire placement data from scan
: Once the reference designators have been scanned and recognized (see the previous menu option), this option starts a scan to find a placement position matching the designators. See appendix E.

Tick marks (component list)
: Enables/disables tick marks (checkmarks) in the component list, and activates operations to 'tick' or 'untick' the items in the component list. See section Ticking components off on page 37 for more information.

User fields (for BoM)...
: Allows you to enter up to four additional fields to include in the bill-of-materials (and to read back from a BoM). The user fields may optionally also be present in the component list.

Fiducial match patterns...
: Opens a dialog for setting match patterns for detection of fiducial definitions from a CPL file.

Application settings...
: Opens a dialog for application-wide settings and configurations, see page 21

Help menu

User Manual...
: Shows the manual in the chosen language, if available.

Keyboard shortcuts...
: Shows the documentation on all shortcut keys in VisualPlace.

Mouse shortcuts...
: Shows the documentation on the available mouse operations, in each particular context.

Release notes...
: Shows the release notes for the current version, including a list of recent changes.

Getting started...
: Starts a presentation in web browser that introduces the application and demonstrates its basic usage.

Information...
: Shows a dialog with version and copyright information.

Check online for updates...
: Tests whether you are currently running the latest version.

APPENDIX H

Keyboard and mouse interface

Keyboard interface

The following keyboard shortcuts are defined for VisualPlace.

F1 Help. When a dialog box is active, context-sensitive help is activated (if available); otherwise, the manual is opened.

F2 Edit table field. In a dialog that contains a table, F2 starts editing a table cell while keeping the contents that the cell already contains.

F3 Toggle call-out with component details. In the case that the information shown in the component list is truncated (or too small), the pop-up call-out can display the complete information in a larger font. Key F3 toggles this call-out on or off.

F4 Flip the PCB. This function key is only active for PCBs with components on both sides —both top and bottom sides also need to have been set up in the project.

F5 Move the position of the centroid marker of a selected component. After selecting this function, the cursor changes to a cross-hair for improved accuracy in setting the component position. If the selected component has no placement information, placement information is added to the component, but the rotation is set to *unknown*. See section Adding or correcting a position (page 27) for details.

Shift + F5 Move the marker of a selected component to the centre of the footprint. This function works best for regular (rectangular) shapes that have no text or drawings inside the shape. See section Adding or correcting a position (page 27) for details.

F6 Rotate the centroid marker —thereby adjusting the orientation of the component in the placement data. The marker is rotated in counter-clockwise direction. The default rotation step is $90°$, but it can be changed to $45°$. See section Adding or correcting a position (page 27) for details.

F7	Toggle a tick mark in the component list and move the selection in the component list to the next row. Tick marks must have been enabled first. See the section Ticking components off (page 37) for details.
F10	If the application menu is set to be automatically hidden, function key F10 makes the menu (temporarily) visible.
F11	Toggle the board view between the live camera stream and the silk-screen images for a PCB. See appendix D for details.
F12	Edit the bill-of-materials.
Shift + F12	Edit the component placement list.
Space bar	Move to the next component(s) that were not yet visualized. If all components on the selected row fit in the viewport, a press on the space bar moves the selection to the next row. Otherwise, the space bar scrolls the viewport to the 'not-yet-viewed' components before moving to the next group. See the section Keyboard navigation (page 39) for details.
+ / -	Zoom in or zoom out. Use the + and – keys on the numeric keypad (depending on the keyboard, the standard + and – may also work). See also Ctrl + 0.
Ctrl + 0	Zooms the PCB to fit the board viewport. See also + and – . Note: this is the 'control' key plus the 'zero' digit, not the letter 'O'.
Ctrl + D	Duplicate table field. In a dialog that contains a table, Ctrl + D copies the value in the cell at the preceding row into the current cell, then moves the current position one cell down.
Ctrl + F	Find component or text. In the board view, this key combination pops up a dialog that allows you to locate any component by its designator. In any dialog that presents a table, this key combination allows you to search through the table for a text. Note that the search is restricted to the active column —so to search for a text in a particular column, you must first select a cell in that column, and then start the search.
Ctrl + L	Set the focus to the filter edit box, above the component list. See the section Component filter (page 19) for details.
Ctrl + P	Open a dialog for all reports.

Ctrl + Mouse

Pressing the Ctrl key and keeping it pressed, changes the mouse cursor to a cross-hair, for precise positioning and coordinate verification.

Mouse interface

VisualPlace implements the common mouse operations like selecting an item from a list, and picking menu items. Beyond simple selection and button activation, VisualPlace has a set of mouse operations, whose functionality depends on the context. A list of these operations is below.

• Hovering

◇ Hovering with the mouse over a marker shows the placement information of the component (position plus rotation), plus its value and package.

◇ Hovering with the mouse over an icon for a note, shows the text of the note in a call-out window (or 'balloon' window).

• Swiping

◇ In design mode, a 'click & drag' movement (or 'swipe') in the board view scrolls the PCB image.

Alternative ways to move a particular area of the PCB into view are to click inside the overview window, or to use the scroll bars. In camera mode, you can click (without dragging) on a position in the board view to move the camera to that location.

• Left click

◇ A click on a position in the *overview window* scrolls the viewport to that position.

◇ In camera mode, a click on a location on the camera image, zooms in on that location (provided that PTZ camera is available).

◇ In camera mode, a 'click & hold' anywhere on the camera image zooms out (this requires a PTZ camera). You will need to keep the left mouse buttons pressed for 0.5 seconds.

• Left double click

◇ A double click on a location on the board shows the information on the component that is nearest to that location. VisualPlace also selects that component in the component list.

◇ When a group of components is selected, a double click on a marker for one of these, selects only that single component.

• Right click

◇ A right-click opens a context menu. You can right-click on a row in the component list, on any location in the board view, on a marker for a component, and on the icon for a note. Each context menu is specific to the item that you click on.

• Mouse wheel

◇ The mouse wheel scrolls lists and tables up and down. When the mouse cursor is in the board view, the mouse wheel zooms the image in and out. This is active for both design and camera modes (a PTZ camera is required).

• Mouse status with Ctrl & Shift keys

◇ When the Ctrl key is kept pressed, the mouse cursor changes to a cross-hair cursor in the board view and the numeric field in the cursor shows the coordinates (in mm or inch, depending on the unit chosen in the application settings).

◇ With the Ctrl key held down, the mouse cursor snaps to the centre of any marker. If the Shift key is also down, this snapping does not occur.

◇ When the left mouse button is clicked and dragged while keeping the Ctrl key down, the numeric field in the cross-hair cursor shows the distance from the position where the button was clicked to the new position. This allows you to measure the distance between two parts, for example.

APPENDIX

Plug-in interface

Plug-ins are used to read CPL and BoM files, for diverse EDA suites. Plug-ins can also be used to interface with other software or equipment. A plug-in is a DLL that must contain a few predefined functions.

This appendix is split into sections to cover 'input' plug-ins, 'output' plug-ins and 'action' plug-ins separately. A plug-in may have multiple interfaces (e.g. *both* for input and for action).

'Input' plug-in: for EDA suite support

At the very least, a plug-in that adds support for reading the production files of an EDA suite must include the `vp_LoadCentroid` function, to load the component placement data. A function to also load a bill-of-materials is optional, but recommended.

LISTING: Plug-in interface, main functions

```
int vp_LoadCentroid(LPCSTR filename, PARTINFO *info, int size,
                    int side, double x_origin, double y_origin,
                    double pcbwidth, double pcbheight);

int vp_LoadBOM(LPCSTR filename, PARTINFO *info, int size);
```

The first three parameters of both functions are: the full filename to the CPL/BoM file, a pointer to an array with records and the size of that array. The `size` parameter is the number of records in the array `info`. The functions must return the number of records that they read from the file, but must not store any records beyond `size`.

To determine the amount of memory to allocate for the placement data and the bill-of-materials, VisualPlace calls each function twice. The first call has parameter `info` set to `NULL` and `size` to zero. When called with these parameters, the functions should return the number of records they can read from the file, but not store any data. VisualPlace uses the return value to allocate the required memory, and then calls the function again, with valid values for `size` and `info`.

Function `vp_LoadCentroid` has five more parameters:

- ◇ The 'side' of the PCB currently being processed —this is 1 for 'top' and 2 for 'bottom'. A plug-in must only collect the components on that side from the CPL file (if a board has components on both sides, VisualPlace calls the plug-in separately for both sides.
  When a CPL file lacks side information, the plug-in must collect all components in the file, and store the `side` parameter in the relevant field of the `info` structure (for CPL file formats that lack side information, the components for top and bottom sides are in separate files, and both should be set in the project).
- ◇ The offset, *x* & *y*, of the origin of the placement data relative to the bottom left corner of the PCB (see below for details). These value are in inch. If no profile of the PCB is detected, VisualPlace instead passes the bottom left corner of the bounding box around the silk-screen data.
- ◇ The size, *width* & *height*, of the PCB (in inch).

VisualPlace uses the bottom left corner of the PCB envelope as the reference point for placement data, as well as for locating any fiducials. With 'envelope' is meant, the bounding rectangle around the PCB profile. So if a PCB has chamfered or rounded corners, the 'bottom left corner' that VisualPlace uses lies outside the PCB. Furthermore, 'bottom left' is relative to how VisualPlace presents the PCB. When viewing the bottom side, VisualPlace flips the Gerber artwork horizontally. As a result, the *bottom left* corner of the bottom view is the *bottom right* corner of the top view.

The plug-in must relocate all component coordinates to be relative to the bottom left corner of the board. Some EDA suites generate information in the CPL file itself to do this conversion; otherwise a plug-in can use the origin offset that is passed to this function. VisualPlace uses dimensions in inch internally, a plug-in must convert coordinates to inch if they are not already in inch.

While scanning for a plug-in for a new set of CPL/BoM files, VisualPlace runs through all plug-ins and calls these functions. If a plug-in function does not detect a data format that it supports, it should return zero (regardless of the parameters passed to the function).

The structure format of the `PARTINFO` structure that the functions in the plug-in functions fill in, is documented in the interface file '`plugin.h`' that is installed in the '`source`' directory below the VisualPlace main directory. The plug-in should clear any field in this structure that it does not fill in (i.e. set strings to empty strings and values to zero).

As is apparent, both the placement-loading function and the BoM-loading function fill in the same structure. Which fields are read from the CPL file and which from the bill-of-materials, depends entirely on the EDA suite and the plug-in. Some EDA suites store all required data in the CPL file:

placement information, package, value and designators. No BoM file is required in these cases.

The position and rotation values in the PARTINFO structure must be stored so that it conforms to the coordinate system used by VisualPlace —see page 53. A few EDA suites use a y-axis that goes top-down; a plug-in for such a suite should invert the y-axis values. Most EDA-suites store the x-axis values for components on the bottom layer inverted, because PCB lay-out programs display the bottom-layer as a mirrored image. For VisualPlace, the x-axis values must *not* be inverted —VisualPlace displays the bottom layer *not* mirrored. The zero-orientations must be conforming IPC-7351, see appendix B.

VisualPlace supports 'user fields' in the bill of materials (and the CPL file). A plug-in can optionally implement the function vp_UserFields to receive the names of the user fields. The first parameter holds an array of points to strings (with the user field names), the second parameter has the count of elements in the array. Elements in the pointer array may be NULL if the corresponding user field was *not* defined.

If a 'function' was set for the user field (see User fields on page 24), the name of the function is passed to the plug-in. The function name can be one of the following (regardless of the language selected for the user interface of VisualPlace):

Stock	the amount in stock or the status of the stock,
Storage	the location where the component is stocked,
Manufacturer	the name of the manufacturer,
Supplier	the name of the preferred supplier or distributor,
OrderNumber	the supplier's order number or SKU for the component,
Cost	the price or cost of the component,
Status	the production status of the component (active, end-of-life, obsolete).

If no function is set for the user field, the name of the field is instead passed to the plug-in.

LISTING: Plug-in interface, user-field information

```
void vp_UserFields(LPCSTR userfields[], int number);
```

For some formats, the position and orientation of the components can only be determined after *both* the placement files and the bill-of-materials have been read. The plug-in needs to define a function vp_PostLoad for this case. This function is called with all relevant information, combined from all production files.

LISTING: Plug-in interface, post-processing input data

```
int vp_PostLoad(PARTINFO *info, int size);
```

The plug-in returns general information through the function `vp_Info`. The information passed backed through the `info` parameter contains a version number for the plug-in interface specification and a ‘loading priority’ number. The ‘interface version’ number allows VisualPlace to check whether it can support the plug-in. The purpose of the load-order priority is to make VisualPlace try precise and explicit plug-ins before running more generic plug-ins. As explained in appendix C, VisualPlace comes with a generic CPL/BoM format plug-in, that can collect data from files created by several EDA suites. However, if a specific plug-in for an EDA suite is available, VisualPlace should try that filter first: a specific filter can usually get more information from the file (with less ‘assumptions’) than a generic plug-in.

LISTING: Plug-in interface, information function

```
void vp_Info(PLUGININFO *info);
```

The structure format of the `PLUGININFO` structure is documented in the interface file ‘`plugin.h`’ —see also the next section for a summary.

‘Output’ plug-in: to write CPL or BoM files

For operation with a pick-&-place machine, it is best to create the data files in the native format of the machine. VisualPlace has one placement data format built-in (which is its native format) and supports output in other formats through plug-ins.

An output plug-in must provide the function `vp_Info`.

LISTING: Plug-in interface, information function

```
void vp_Info(PLUGININFO *info);
```

The `PLUGININFO` structure has five fields. The plug-in should copy a name that identifies the pick-&-place machine in the field `description`, and add the extension of the file format. The machine name and the file format must be separated with a vertical bar (‘`|`’). For example, if a machine from ‘BrandX’ has model name ‘Unit24’ and uses files in CSV format, the description string could look like:

```
"BrandX Unit24.csv"
```

The version field in the structure should be set to the current version for the plug-in interface. This version number is documented in the interface file 'plugin.h'. The priority field is ignored for output plug-ins. The field errorcode should be set to the most recent error that the plug-in encountered (or 0 if none); the errorline field is irrelevant for output plug-ins.

Next to the vp_Info function, the plug-in must have vp_WriteCentroid or vp_WriteBOM (or both).

LISTING: Plug-in interface, main functions

```
BOOL vp_WriteCentroid(LPCSTR filename,
                      const PARTINFO *info, int size);
BOOL vp_WriteBOM(LPCSTR filename,
                 const PARTINFO *info, int size);
```

Both functions receive the same data, but each is called to write a different data file.

The size parameter holds the number of components that are in the info list. If a PCB has components on two sides, the function is called twice once for each side.

The structure format of the PARTINFO structure that the functions in the plug-in functions fill in, is documented in the interface file 'plugin.h' that is installed in the 'source' directory below the VisualPlace main directory.

The functions return TRUE on success and FALSE on failure. In the case of failure, a plug-in can return the reason of the failure in the errorcode field of the PLUGININFO structure on the next call to vp_Info. After a failure of either of these functions, VisualPlace may call vp_Info to get the errorcode, in order to give a more specific error message.

An output plug-in can implement the function vp_Configure to get some additional information from VisualPlace for the purpose of showing a configuration dialog or for storing/retrieving settings. The syntax and functionality of the vp_Configure is described in the section of 'action' plug-ins (see below). In the case of an output plug-in, though, the last parameter (Silent) is always true and the function is called immediately before the call to vp_WriteCentroid.

'Action' plug-in: to interface with applications

VisualPlace can interface with other applications or equipment if a suitable plug-in is available. Typical uses for these plug-ins are:

◇ Print a label for a component from the bill-of-materials or the component list. The label can optionally include a reference to the project name and the row in the bill-of-materials. Currently, plug-ins are available for the Dymo LabelWriter and the Zebra Technology label printers (plus generally any label printer that supports the ZPL printer language).
◇ Look up a component in an inventory system.
◇ Drive a component carousel to move the bin with the selected component to the pick-up position, e.g. during manual assembly.
◇ Control a laser-pointer system that points out the locations on the PCB where the currently selected component(s) must be placed.

In summary, these plug-ins are invoked when performing an action on a group of components, which is why they are called 'action' plug-ins.

The 'action' plug-ins follow the same structure as the 'input' plug-ins for EDA suite support, but use different functions.

LISTING: Plug-in interface, part selection notification function

```
void vp_NotifySelect(const PARTINFO *info, int count);
```

At a minimum, an *action* plug-in must have the `vp_NotifySelect` function. VisualPlace calls this function whenever the selection changes in the component list. The parameters to the function are an array with the components that are currently selected, and the count of items in the array. For example, if the selected row in VisualPlace is the a capacitor of 100nF, and there are eight instances of this component on the PCB, the parameter `info` will hold eight elements with the full information of these components (including their positions and any user fields).

An exception to the above is when the plug-in is invoked from the bill-of-materials dialog (see section Adjusting the Bill of Materials on page 29). In this case, only a single `PARTINFO` entry is passed (parameter `count` is always 1) for the selected row, and no position data is passed in.

LISTING: Plug-in interface, plug-in configuration function

```
BOOL vp_Configure(HWND hwndParent, LPCSTR Profile, LPCSTR Project, BOOL
Silent);
```

For configuration, a plug-in declares the function `vp_Configure`. The handle to the main application window of VisualPlace is passed in, as well as the full path and filename of the INI file that VisualPlace uses to store its settings in (parameter `Profile`). A plug-in can use this same INI file for its own configuration. The parameter `Project` holds the active project name, which is technically also an INI file. If no project is open, this parameter may be `NULL` or an empty string. The last parameter determines how the plug-in should respond. If `Silent` is *true*, the plug-in should initialize and configure itself *silently* —i.e. without giving feedback to the user. When the `Silent` parameter is *false*, the plug-in should always show a dialog box for the user to modify any configuration settings.

Action plug-ins may also have the `vp_Info` function, with the same syntax as the function for 'input' plug-ins. The 'loading priority' value of the plug-in also determines the order of execution of the 'action' plug-ins.

'Inventory' plug-in: for inventory look-up

The bill-of-materials exported from an EDA suite often lists only the value and package of each component, but not its manufacturer product number, preferred supplier, or stock location. These fields can be added manually, in the BoM editor (see Adjusting the Bill of Materials on page 29), but VisualPlace can also import them from an inventory database. See section Importing inventory data on page 32 for more information on inventory look-up.

LISTING: Plug-in interface, information function

```
    int vp_Inventory(LPCSTR filename, PARTINFO *info, int size);
```

The purpose of `vp_Inventory` is to format and parse the data of an inventory database (or the exported data of a stock management system) into a data structure defined by VisualPlace. Like the `vp_LoadBOM` function, the parameters are the full filename to the inventory file, a pointer to an array with records and the size of that array. The `size` parameter is the number of records in the array `info`. The function must return the number of records that it reads from the file, but must not store any records beyond `size`. Note that the file passed to this function may be a temporary file, for example a file downloaded from a remote system via a URL.

To determine the amount of memory to allocate for the component data, VisualPlace calls `vp_Inventory` twice. The first call has parameter `info` set to `NULL` and `size` to zero. The function should return the number of records they can read from the database, but not store any data. VisualPlace uses the return value to allocate the required memory, and then calls the function again, with valid values for `size` and `info`.

The `vp_Inventory` assumes fixed roles for the user fields in the `info` structure. The plug-in should store the following information in each user field:

field 0	Stock: the number of components in stock
field 1	Storage: the stock location of the component
field 2	Manufacturer
field 3	Supplier: the preferred supplier or distributor

field 4 Cost: the price or cost of the component
field 5 Status: for a status like 'obsolete' or 'not for new designs'

APPENDIX J

License

This manual and the software programs consisting of all the files included in the package labelled 'VisualPlace' and hereafter collectively referred to as 'the product', is copyright © 2010–2024 CompuPhase. The product under this license is provided free of charge. You are granted a non-exclusive license to use the product for under the following conditions:

YOU MAY:

◇ Use the product as many times as you like, for as long as you like.

◇ Copy the product and distribute copies of the product, provided that you distribute the complete and unmodified product (including this manual with the copyright, license and disclaimer of warranty).

YOU MAY NOT:

◇ Use the product for illegal purposes.

◇ Distribute altered versions of the product, or distribute only parts of the product, or repackage (parts of) the product as a component of another product or product suite.

◇ Remove or conceal the copyright of the product, or claim copyright on (parts of) the product.

◇ Charge money or fees for the product, except to cover distribution costs.

LIMITED WARRANTY:

CompuPhase cannot be held liable for *any* damage or loss of profits that results from the use of the product (or part thereof), or from the inability to use it, to the extent that the law permits.

Index

!

× symbol, 29
µ symbol, 22
Ω symbol, 22

A

Adding components, 27
Aligning the origin, 12
Allegro Design, **68**, 77
Altium Designer, **69**, 70
Apache Subversion, *see* Subversion
Archive project, 93
Assembly
- ~ flipbook, 51
- ~ machine, *see* Pick-&-place machine
- ~ program, *see* Component Placement List (machine)
- ~ stages, 19, **25**, 30, 48

Attributes (module), 76
Authentication (camera), 82
Auto-expand group, **36**, 37, 39, 40, 97
Autodesk, 72

B

Background image, *see* Silk-screen image
Balloon pop-up, *see* Call-out
Bar-code scanner, 4, **23**, 31, 54
Barrel distortion, 83
BGA package, 66
Bill of Materials, 2, 9, 61, 62
- colour printing, 51
- editing the ~, 14, **29**
- import, 31
- mismatch with CPL, 17, 18

Bitmap formats, *see* Image formats
Board population, *see* PCB assembly
Board view, **16**, 34, 36, 40
BoM, *see* Bill of Materials
Bottom left corner, 106
Bounding box, 106
Button bar, *see* Navigation ~

C

Cache, 23
Cadence Design Systems, 68
CadSoft, 72
CadStar, **70**
Call-out, **20**, 34, 60
Camera, 6, 8, 11, 26, 37, **41**
- authentication, 82
- ~ calibration, 83
- ~ configuration, 82
- ~ distortion, 83
- ~ lens, 81
- PTZ ~, 42, 81, 83, 85
- shortcut key, 101

Carousel (component ~), 110
Category (components), 34, 35
Centre
- geometric ~, 16, 44, 46
- ~ of mass, *see* Centroid

Centring a centroid marker, **28**, 100
- automatic, 28

Centroid, **16**, 97
- centring a ~ marker, **28**, 28, 100
- ~ correction, **46**, 59
- ~ marker, **16**, 44, 46, 59, 97, *see also* Marker

Centroid file, *see* Component Placement List
Checkmarks, 37, 40
CircuitStudio, **70**
Clockwise, *see* Rotation
Collapsing groups, 36
Colours
- component list, 17, 18
- ~rules (bill of materials), 52

Comma-separated values, 4, 50, 61, 62, 68
Commit, *see* Version Control
Component
- ~ carousel, 110
- find ~, **54**, 54, 101
- ~ groups, 36, 39
- ~ packaging, 64
- through-hole ~, 17, 27, 43
- ~ visualization, 23, 26

Component list, 16, **17**
- colours, 17, 18
- ~ filter, 19, 101
- sort ~, 54

Component Placement List, 9
- export ~, **48**, 60, 93
- intrinsic format, 29, 56, 60, 97
- machine-specific ~, 48, 63
- mismatch with BoM, 17, 18
- ~ origin, *see* Origin (fabrication files)

Context menu, 14, 30, 33
Contour (PCB), *see* PCB profile
Coordinates (pick-&-place), 53, 107

Coordinates (view), 53
Correction
  placement ~, **46**, 59
CPL, *see* Component Placement List
Cross-hair cursor, 28, 44, 53, 100, 102, 104
Cumulative (assembly stage), 25
Cut-out (PCB), 8

D

Datum position, *see* Origin (PCB)
Decal, *see* Footprint
DesignSpark PCB, **73**
DFN package, 66
DipTrace, **71**
Distortion (camera), 83
DLL, 105
Do not mount (attribute), 19, 25, **36**, 36, 52
DPAK package, 65
dpi, 7
Drill file, 8, 11, 41
  ~ alignment, 41
  ~ format, 8
Dummy components, 30, 49
Dymo LabelWriter, 110

E

EAGLE, **72**
Easy-PC, **73**
ED-7303 (standard), 57
EDA, 1, 2, 105, *see also* appendix C
Edges (PCB), *see* PCB contour
EIA-481 (standard), 64
Electronic Workbench, *see* Ultiboard
EMS, 1, 2
Envelope, 106
Europlacer, 75
Excel, *see* Microsoft Excel
Excellon drill file, *see* Drill file
Expanding groups, **36**, 39
Export, 2
  ~ BoM, 93
  ~ component placement list, **48**, 60, 93
Extended attributes, *see* Gerber X2 format
Eye strain, 6

F

Fabrication outputs, 9
Fiducial, **32**, 42, 48, 49, 99
  ~ match patterns, 49
  primary ~, 21, 32, 49
Field colours, 17, 18
File comparison, 23, 90–92
Filter
  component list, **19**, 101
  packages, 31
Find components, *see* Locating ~
Flip PCB, 95, 100
Flipbook, *see* Assembly flipbook
Foot switch, 6, 40, 85, *see also* Palm ~
Footprint, 44, 47, 59, 64
  delete a ~, 59
  ~ deviates by. . . (warning), 46
  ~ lay-out, 18, 55, **57**
  link ~, 55, 60
  ~ scanning, *see* Scan footprints
Footprint visualization, *see* Package ~

G

GenCAD (file format), 68
Generic footprint name, 55
Geometric centre, 16, 44, 46
Gerber format, 2, 7, 46
  X2 format, 7–9, 12, 18, 44
  X3 format, 68, 76
Glyph, 86, 87
Graph paper, 83
Graticule, *see* Cross-hair cursor
Grid, 51
Groups, 36, 39

H

Hands-free operation, *see* Foot switch
Hershey Simplex font, 87
Heuristic, 46
Hidden menu, 21, 101
Hovering (mouse), 103

I

IEC 61188-7 (standard), 47, **64**
Image, *see also* Silk-screen
  ~ formats, 7
  ~ resolution, 7
Import, 31, 32, 62
  ~ from URL, 31, 32, 62
  ~ inventory, 32
Inconsistent
  ~ package, 18
  ~ value, 17
Inter-Process Communication, 109
Internationalization, *see* Language settings
Intrinsic format (CPL file), 29, 56, 60, 97
Inventory system, 24, 32, 61, 111

IP camera, *see* Camera
IPC-7351 (standard), 47, 57, **64**, 66

J

JEDEC tray, 67

K

Keyboard shortcuts, **39**, 99, 100
  ←/→ (expand/collapse), 39
  +/- (zoom in/out), 42, 96, 101
  Ctrl+D (duplicate field), 30, 101
  F2 (edit field), 27, 30, 100
  F3 (toggle call-out), 20, 34, 43, 100
  F4 (flip top/bottom), 34, 100
  F5 (move component), 28, 100
  F6 (rotate centroid), 28
  F6 (rotate component), 100
  F7 (toggle tick mark), 39, 101
  F10 (show menu), 101
  F11 (toggle camera), 41, 101
  F12 (edit BoM), 101
  Shift+F5 (centre centroid), 28, 100
  Shift+F12 (edit placements), 101
  space-bar (move down), 39, 40, 101
Keystone distortion, 83
KiCad, **75**

L

Label printing, 110
Land pattern, *see* Footprint
Language settings, 99, *see also* Locale
Latency (camera), 81
Lay-out (user interface), 21
Lens (camera), 81
LibreOffice, 62
License, 113
Link (footprint), 55, 60
Linux, 23
List, *see* Component list
Locating (components), 54

M

Machine-specific placement file, *see* Component Placement List (machine)
Marker, **16**, 25, 44, 46, 59, 81, 97, 103
  centring a ~, **28**, 28, 100
  ~ colour, 26, 34
  ~ correction, 46, 59
  open style ~, 84
  ~ orientation, 34, 48
  Rotating a ~, 100
  ~ size, 26, 34
  verifying centroid ~, 44
Mask (Gerber), *see* Solder mask
Measurements
  distance, 44, **53**, 104
  position, 53
Mentor Graphics, 77
Menu
  hidden ~, 21, 101
  ~ interface, 93
micro symbol, 22
Microsoft Excel, 61, 62, 68, 77, *see also* XLS file format
Mil (unit), 47, **59**
MJPEG, 81, 85
Module attributes, 76
Modules position file, *see* Component Placement List
Moulded package, 64
  capacitor, 64
Mount order, *see* Placement order
Mounting holes, 30, 49
Mouse
  ~ hovering, 103
  ~ shortcuts, 99, 103
  ~ swiping, 103
  ~ wheel, 104
  zoom-out, 42

N

Naming convention (packages), 22, **56**
Navigation button bar, **40**, 97
Network camera, *see* Camera
Non-polarized chip, 64
Normal orientation, 47, **64**, 66
Notes, 33
Novarm, 71
Number One Systems, 73
Numeric keypad, 42, 96, 101

O

OCR, *see* Optical Character Recognition
Ohm symbol, 22
Opacity, 26, 34, 50
Optical Character Recognition, 2, 13, **86**, 86, 87, 90
  penalty, 87
OrCAD, 68, **77**
OrCad, *see also* Allegro Design
Orientation
  component ~, 48, 100
  ~ correction, 59
  normal ~, 47, **64**
Origin
  aligning the ~, 12
  component ~, 44
  ~ of fabrication files, 7, 12, 73–75, 106
Outline, *see* PCB profile
Override mount order, *see* Placement order
Overview window, **16**, 96, 103

P

Package, *see also* Footprint
  draw ~ shapes, 23, 26, 97
  ~ naming convention, 22, **56**
  ~ visualization, 23, 26, 97
Package selection, 31
Packages (application), 58
Packaging (components), **64**
PADS, **77**
Palm switch, 40, *see also* Foot switch
PCB
  ~ assembly, 1
  ~ origin, *see* Origin of fabrication files
  ~ outline drawing, *see* PCB Contour
  ~ profile, 8, **10**, 12
  ~ scan, 8
  variants, *see* Assembly stages
  ~ zones, *see* Zones
PCB profile, 13, 106
PDF
  ~ reader, 23
  ~ reports, 52, *see also* Reports
PDP-100 (standard), 57
Penalty (OCR), 87
PGA package, 66
Pick-&-place machine, 43
  support for ~, 50
Pincushion distortion, 83
Placement
  adding a ~, 28
  changing ~ orientation, 28, 100
  ~ correction, *see* Centroid correction
  ~ order, 34, 35
PLCC package, 66
PLCC2, 65
PLCC6, 65
Plug-ins, 1, 22, 30, **105**
  action ~, 109
  input ~, 105
Polarized chip, 64
Pop-up menu, *see* Context menu
Population, *see* PCB assembly
Printing, 52, *see also* Reports
  colour ~, 51
Profile, *see* PCB profile
Project file, 9
Protel, **69**, *see also* Altium Designer
PTZ (camera), 42, 81, 83, 85
Pulsonix, **78**
Push-button switch, *see* Palm switch

Q

QFN package, 66
QFP package, 66

R

RCS, 60, **89**, 90, 91, *see also* Version Control
Recipe, *see* Component Placement List (machine)
Red field (component list), 17, 18, *see also* Colours
Registration markers, 41
Reports, **50**, 94, 101
  excluded components, 52
Resolution (image), 7
Revision control, *see* Version Control
Right-click menu, *see* Context menu
Rotation
  direction of ~, 47
  ~ of component, 28, 100
  ~ of Gerber data, 12
  ~ of view, 96
  step size ($90°/45°$), 28, 95
RS-274X, *see* Gerber
RS-Components, 73
Rulers, 51

S

Scan (PCB image), 8, 13, **86**
Scan footprints, 28, 86
  resolution, 47
Search
  ~ component, 101, *see also* Locating ~
  ~ table, 101
Shortcut key, *see* Keyboard shortcut
Sieb & Meyer, *see* Drill file
Silk-screen, 7
  ~ colour, 26
SKU, 107
Slot (PCB), 8
SMD, *see* Surface Mount Device
Snapping (mouse cursor), 44, 53, 104
SOIC package, 66
Solder mask, 46
Solder paste stencil, 5
SON package, 66
Sorting (component list), 34, 54
SOT package, 65
Sound events, 22, 39
Spreadsheet, 61, 68
SSOP package, 66
Stage, *see* Assembly stages
Stencil, *see* Solder paste stencil
Stock management, *see* Inventory system
Stratch
  starting from ~, 13
Subversion, 60, **89**, 89, 91, *see also* Version Control
Sumatra PDF, 52
Surface Mount Devices, 43, 79, 80
SVG images, 26

Swiping, 103
Synchronizing files, *see* Version Control

T

Table editing, 100, 101
Tape on reel, 64
TARGET 3001, **78**
Text Import Wizard (Excel), 62
Thou (unit), *see* Mil
Through-Hole components, 17, 27, 43
Tick off (components), **37**, 40, 101
TortoiseSVN, 90
Touch screen, 40
Translation (footprint), *see* Link ~
Transparency, *see* Opacity
Tray (JEDEC), 67
TSOP-I package, 66
TSSOP package, 66

U

ULP, 73
Ultiboard, **79**
Undo, 94
Unit, **21**, 27, 44, 51, 59, 99
  upper case, 22
URL-encoding, 85
User interface, 16
User-defined field, 19, **24**, 32, 51, 63, 99, 107

V

Variants, *see* Assembly stages
Version Control, 23, 60, **89**, 89–91, 98
  commit, 92

W

WestDev Ltd., 73
Wildcards (filter), 19
Wine, 23
WinMerge, 90

X

X2 format, *see* Gerber X2 format
XLS (file format), 68, *see also* Microsoft Excel

Y

Yellow field (component list), 18, *see also* Colours

Z

Zebra label printers, 110
Zero orientation, *see* Normal orientation
ZIP file, 93
Zones, **37**, 39
  ~ codes, 38
  ~ sort order, 38, 42
Zoom, 42, 96, 101
  camera ~, 42, 81, 83, 85
ZPL (printer language), 110
Zuken, 70