

CvPcb

19. Oktober 2016

Inhaltsverzeichnis

1 Einführung zu CvPcb **2**

2 CvPcb Eigenschaften **2**

2.1 Manuelle oder Automatische Zuordnung . . . 2

3 Aufruf von CvPcb **3**

4 CvPcb Befehle **3**

4.1 Hauptbildschirm . . . 3

4.2 Hauptfenster Werkzeugleiste . . . 4

4.3 Hauptfenster Tastatur-Befehle . . . 4

4.4 CvPcb Konfiguration . . . 5

5 Footprint Bibliotheken-Management **6**

5.1 Wichtiger Hinweis: . . . 6

5.2 Footprint Bibliothek Tabellen . . . 6

5.2.1 Globale Footprint Bibliotheks Tabelle . . . 7

5.2.2 Projektspezifische Footprint Bibliotheks Tabelle . . . 7

5.2.3 Erstkonfiguration . . . 8

5.2.4 Tabelleneinträge hinzufügen . . . 8

5.2.5 Umgebungsvariablen Ersetzung . . . 8

5.2.6 Benutzung des GitHub Plugins . . . 9

5.2.7 Nutzungsmuster . . . 10

5.3 Benutzung des Footprint-Bibliotheks-Tabellen-Assistenten . . . 12

6 Den aktuellen Footprint anzeigen **15**

6.1 Der Zeige Footprint Befehl . . . 15

6.1.1 Status-Balken Information . . . 15

6.1.2 Tastatur-Befehle . . . 15

6.1.3 Maus-Befehle . . . 16

6.1.4 Kontextmenü . . . 16

6.1.5 Horizontale Werkzeugleiste . . . 16

6.1.6 Senkrechte Werkzeugleiste . . . 17

6.2 Anzeige des aktuellen 3D-Modells 18
6.2.1 Maus-Befehle 18
6.2.2 Horizontale Werkzeugleiste 18

7 CvPcb nutzen um Bauteile mit Footprints zu verbinden **20**
7.1 Manuelle Zuweisung von Footprints zu Bauteilen 20
7.2 Footprint-Liste filtern 20

8 Automatische Zuweisung **24**
8.1 Äquivalenz-Dateien 24
8.2 Äquivalenz-Datei Format 24
8.3 Automatische Zuweisung von Footprints zu Bauteilen 25

Referenz-Handbuch

Copyright

This document is Copyright © 2010-2015 by it's contributors as listed below. You may distribute it and/or modify it under the terms of either the GNU General Public License (http://www.gnu.org/licenses/gpl.html), version 3 or later, or the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/), version 3.0 or later.

All trademarks within this guide belong to their legitimate owners.

Mitwirkende

Jean-Pierre Charras, Fabrizio Tappero, Wayne Stambaugh.

Übersetzung

André S.<ansc.de@gmail.com>

Feedback

Please direct any bug reports, suggestions or new versions to here:

- About KiCad document: https://github.com/KiCad/kicad-doc/issues
- About KiCad software: https://bugs.launchpad.net/kicad
- About KiCad software i18n: https://github.com/KiCad/kicad-i18n/issues

Datum der Veröffenlichung und Software-Version

Veröffenlicht am 22. Mai 2015.

1 Einführung zu CvPcb

Cvpcb ist ein Werkzeug, das es Ihnen erlaubt, für das Layout der Leiterplatte Schaltplankomponeten einen Footprint zuzuordnen. Diese Zuordnung wird der Netzliste hinzugefügt, die vom Schaltplan-Tool Eeschema erzeugt wird.

Die Netzlisten-Datei die von Eeschema erzeugt wird, spezifiziert welcher Footprint mit welchem Bauteil verbunden wird, aber nur ,wenn das Footprint-Feld im Bauteil initialisiert wurde.

Das ist der Fall, wenn Footprints während der Schaltplaneingabe zugewiesen wurden, indem das Footprintfeld des Bauteils gesetzt wurde, oder es wird von der Bauteil-Bibliothek gesetzt, wenn das Bauteil geladen wird.

CvPcb stellt eine bequeme Methode zur Footprint-Zuweisung während der Schaltplaneingabe bereit. Es stellt ein Filter für die Footprint-Liste, Footprint-Vorschau und eine 3D-Vorschau bereit, um dabei zu helfen, sicher den richtigen Footprint jedem Bauteil zuzuweisen.

Bauteilen können Footprints manuell oder automatisch zugewiesen werden, indem man Äquivalenz-Dateien (.equ Dateien) erstellt. Äquivalenz-Dateien sind Look-Up-Tabellen, die jedem Bauteil seinen Footprint zuweisen.

Dieser interaktive Ansatz ist einfacher und weniger fehleranfällig als die Footprints direkt im Schaltplaneditor zuzuweisen.

CvPcb zeigt Ihnen die Liste der verfügbaren Footprints auf dem Bildschirm an, um sicher zu stellen, dass Sie den richtigen Footprint zuweisen.

Es kann nur aus Eeschema gestartet werden, aus der obersten Werkzeugleiste, entweder wenn Eeschema aus dem KiCad Projekt-Manager gestartet wird oder wenn Eeschema als eigenständige Anwendung gestartet wird.

CvPcb aus einem über den KiCad-Manager gestarteten Eeschema zu öffnen ist allgemein besser weil:

- CvPcb benötigt die Projekt-Konfigurationsdatei um zu wissen, welche Footprint-Bibliotheken geladen werden müssen.
- CvPcb initialisiert in den Bauteilen das Footprint-Feld für das aktuelle Schaltplanprojekt. Das ist nur möglich, wenn die Projekt-Datei sich im gleichen Pfad wie der geöffnete Schaltplan befindet.

CvPcb aus einem vom KiCad-Manager gestarteten Eeschema zu starten, stellt dies alles automatisch sicher.



Warnung

Sie **können** CvPcb tatsächlich auch aus einem eigenständig gestarteten Eeschema aufrufen, jedoch beachten Sie bitte, dass jeder geöffnete Schaltplan, der keine Projekt-Datei im gleichen Verzeichnis hat, fehlende Bauteile haben kann, da fehlende Bibliotheken nicht in CvPcb angezeigt werden. Wenn es im gleichen Verzeichnis wie der geöffnete Schaltplan keine fp-lib-table Datei gibt, sind auch keine projektspezifischen Footprint-Bibliotheken verfügbar.

---

2 CvPcb Eigenschaften

2.1 Manuelle oder Automatische Zuordnung

CvPcb erlaubt interaktive (manuelle) Zuweisung aber auch automatische Zuweisung über Äquivalenz-Dateien.

3 Aufruf von CvPcb

CvPcb wird nur aus dem Schaltplan-Eingabeprogramm Eeschema aufgerufen, über das Werkzeug:

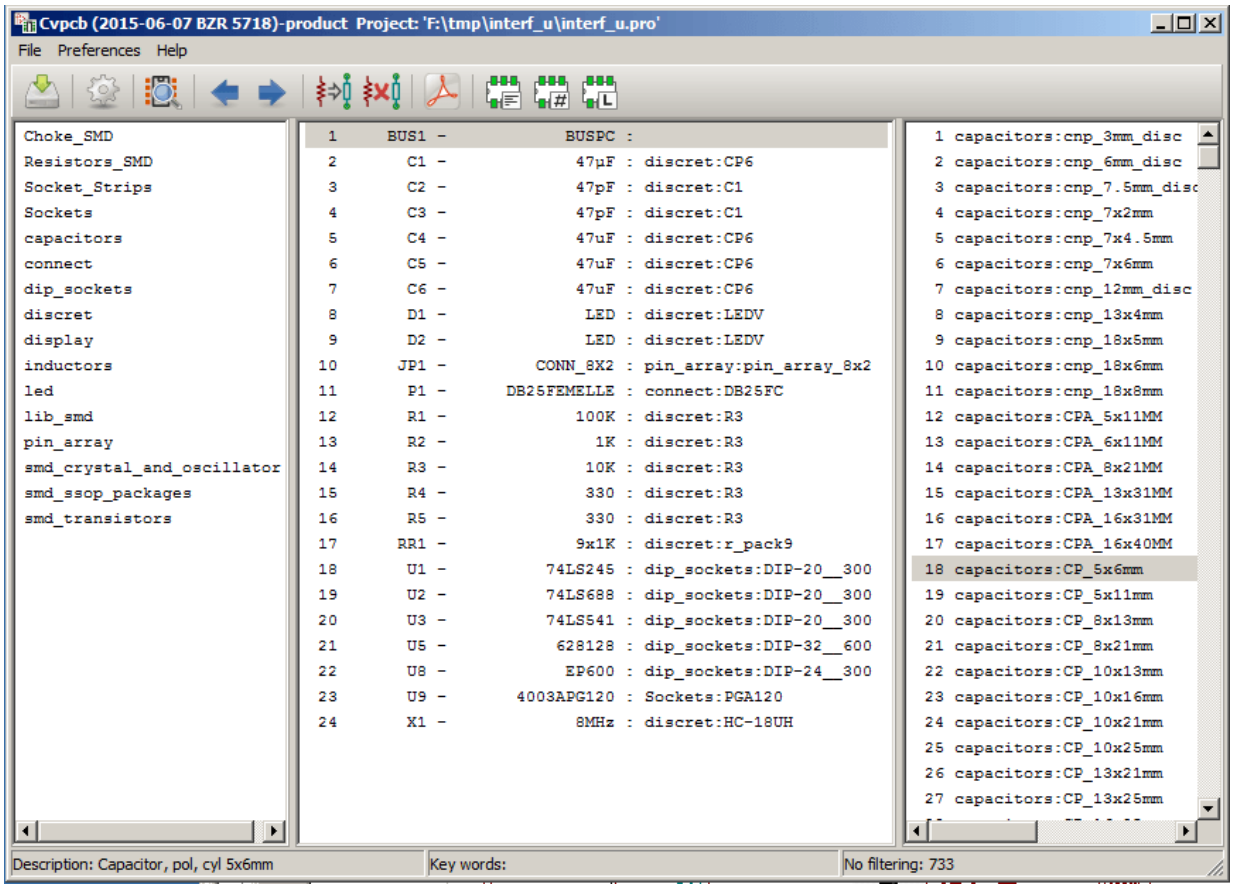


Eeschema übergibt automatisch die korrekten Daten (Bauteilliste und Footprints) an CvPcb. Es muss keine Aktualisierung durchgeführt werden (außer ein paar neue Bauteile haben noch keine Referenz), starten Sie einfach CvPcb.

4 CvPcb Befehle

4.1 Hauptbildschirm

Das Bild unten zeigt das Hauptfenster von CvPcb.



Der linke Bereich enthält die Liste der verfügbaren Footprint-Bibliotheken, die dem Projekt zugewiesen sind. Der mittlere Bereich enthält die Liste der Bauteile, die aus der Netzliste geladen wurden. Der rechte Bereich enthält die Liste der verfügbaren Footprints, die aus den Projekt-Footprint-Bibliotheken geladen wurden. Der Bauteil-Bereich wird leer sein, wenn keine Neztliste geladen wurde und der Footprint-Bereich kann ebenfalls leer sein, wenn keine Footprint-Bibliotheken gefunden wurden.

4.2 Hauptfenster Werkzeugleiste



Die oberste Werkzeugleiste erlaubt den leichten Zugriff auf folgende Befehle:

	Übertrage die aktuelle Footprint-Zuweisung an Eeschema (das ist der Inhalt des Footprint-Felds).
	Das CvPcb Konfigurations-Menü aufrufen.
	Zeige den Footprint des ausgewählten Bauteils im Footprint- Fenster.
	Wähle automatisch das vorherige Bauteil in der Liste ohne Footprint Zuweisung.
	Wähle automatisch das nächste Bauteil in der Liste ohne Footprint Zuweisung.
	Weise Footprints automatisch Bauteilen zu, unter Benutzung einer Äquivalenz-Datei.
	Lösche alle Footprint-Zuweisungen.
	öffne die gewählte Footprint-Dokumentations-Datei unter Benutzung des Standard-PDF-Betrachters.
	Ein- oder Ausschalten der Filterung, um die Liste der Footprints auf die für das gewählte Bauteil gefilterten Footprints zu begrenzen.
	Ein- oder Ausschalten der Filterung, um die Liste auf die Pin-Anzahl des gewählten Bauteils zu begrenzen.
	Ein- oder Ausschalten der Filterung, um die Liste der Footprints auf die gewählte Bibliothek zu begrenzen.

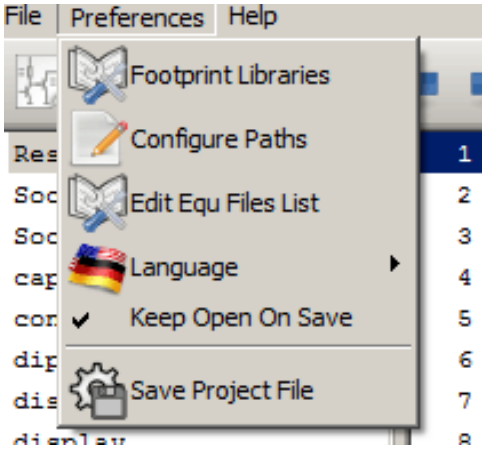
4.3 Hauptfenster Tastatur-Befehle

Die folgende Tabelle listet die Tastatur-Befehle für das Hauptfenster auf:

Rechter Pfeil / Tab	Aktiviere den nächsten Bereich rechts vom aktuell aktiven Breich. Springe zum ersten Bereich, wenn aktuell der letzte Bereich aktiv ist.
Linker Pfeil	Aktiviere den nächsten Breich links vom akutell aktivierten Bereich. Springe zum letzten Bereich, wenn der erste Bereich akutell aktiv ist.
Pfeil nach oben	Wähle den vorherigen Eintrag der akutell gewählten Liste.

Pfeil nach unten	Wähle den nächsten Eintrag der akutell gewählten Liste.
Seite nach oben	Wähle den Eintrag eine ganze Seite weiter oben in der aktuell gewählten Liste.
Seite nach unten	Wähle den Eintrag eine ganze Seite nach unten in der aktuell gewählten Liste.
Pos 1	Wähle den ersten Eintrag in der akutell gewählten Liste.
Ende	Wähle den letzten Eintrag in der akutell gewählten Liste.

4.4 CvPcb Konfiguration



CvPcb kann automatisch nach dem Speichern der Footprint-Zuordnungsdatei geschlossen werden, oder nicht.

Aufrufen des „Bibliotheken“ Eintrags im „Einstellungen“ Menü zeigt den Bibliotheks-Konfigurations-Dialog an.

Abhängig von der CvPcb Version gibt es 2 unterschiedliche Methoden des Bibliotheks-Management:

- Das historische Management, das *.mod Dateien und eine Bibliotheks-Liste von Dateien verwendet.
- Das neue „Pretty“ Format, des eine Datei pro Footprint verwendet. Es verwendet eine Verzeichnis-Liste. Jedes Verzeichnis (*.pretty Verzeichnis-Name) ist eine Bibliothek. Wenn diese neue Methode des Bibliotheks-Management verwendet wird, können sie ebenfalls native Bibliotheken aus GEDA/GPCB oder sogar Eagle XML-Format Dateien verwenden.

5 Footprint Bibliotheken-Management

5.1 Wichtiger Hinweis:

Dieser Abschnitt ist nur für KiCad-Versionen ab Dezember 2013 relevant

5.2 Footprint Bibliothek Tabellen

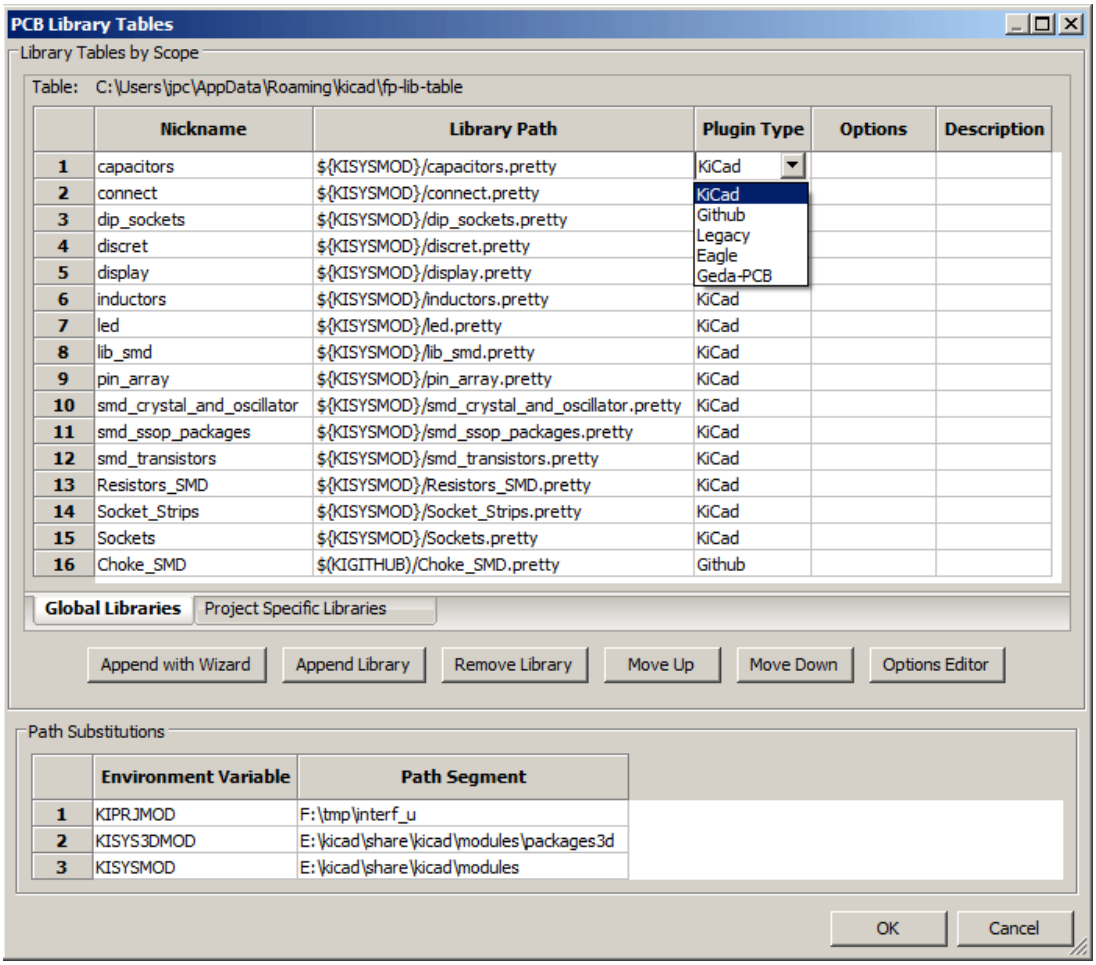
Seit Dezember 2013 verwenden Pcbnew und CvPcb ein neues Bibliotheks-Verwaltungs-Werkzeug das auf ***footprint library tables*** basiert, was es erlaubt **Footprint Bibliotheken direkt zu verwenden** von:

- KiCad alte Footprint Bibliotheken (.mod Dateien)
- KiCad neue *.pretty* Footprint Bibliotheken (auf der lokalen Festplatte) (Verzeichnisse mit .pretty Erweiterung, die .kicad_mod Dateien enthalten)
- KiCad neue *.pretty* Footprint Bibliotheken (auf unserem Github-Server, oder anderem Github-Server)
- GEDA Bibliotheken (Verzeichnisse die .fp Dateien enthalten)
- Eagle Footprint Bibliotheken

Anmerkung

- Sie können nur KiCad *.pretty* Footprint Bibliotheken auf ihrer lokalen Festplatte schreiben (und die .kicad_mod Dateien in diesen Verzeichnissen)
- Alle anderen Formate können nur gelesen werden.

Das Bild unten zeigt den Footprint-Bibliotheks-Tabelle Bearbeiten Dialog, welcher über das Aufrufen des „Footprint Bibliotheken“ Eintrags im „Einstellungen“ Menü geöffnet werden kann.



Die Footprint Bibliotheks Tabelle wird benutzt, um eine Footprint-Bibliothek jedes unterstützten Bibliothek-Typs mit einem Bezeichner zu verbinden. **Dieser Bezeichner wird benutzt, um nach Footprints zu suchen** anstelle der bisherigen Methode, welche auf die Anordnung des Bibliotheks-Such-Pfads angewiesen war

Das erlaubt es CvPcb auf Footprints mit dem gleichen Namen in unterschiedlichen Bibliotheken zuzugreifen und dabei sicherzustellen, dass der korrekte Footprint aus der passenden Bibliothek geladen wird. Es ermöglicht CvPcb ebenfalls Bibliotheken unterschiedlicher PCB-Editoren wie Eagle und GEDA zu laden.

5.2.1 Globale Footprint Bibliotheks Tabelle

Die globale Footprint Bibliotheks Tabelle enthält eine Liste von Bibliotheken die immer verfügbar sind, unabhängig von der aktuell geladenen Projektdatei. Die Tabelle ist in der Datei fp-lib-table im Home-Verzeichnis des Benutzers gespeichert. Die Position des Verzeichnisses hängt vom verwendeten Betriebssystem ab.

5.2.2 Projektspezifische Footprint Bibliotheks Tabelle

Die projektspezifische Footprint Bibliotheks Tabelle enthält die Liste von Bibliotheken die spezifisch für die aktuell geladene Projektdatei verfügbar sind. Die projektspezifische Footprint Bibliotheks Tabelle kann nur bearbeitet werden, wenn sie gleichzeitig

mit der Projekt-Netzliste geladen wird. Wenn keine Projektdatei geladen ist oder es keine Footprint Bibliotheks Tabllen-Datei im Projektpfad gibt, wird eine leere Tabelle erzeugt, welche bearbeitet werden kann und später mit der Footprint-Zuweisung-Datei gespeichert wird.

5.2.3 Erstkonfiguration

Wenn Pcbnew oder CvPcb das erste mal ausgeführt werden und die globale Footprint-Tabellen-Datei **fp-lib-table** wurde nicht im Home-Verzeichnis des Benutzers gefunden, werden Pcbnew oder CvPcb versuchen, die Standard-Footprint-Tabelle fp-lib-table, die im System-KiCad Vorlagen-Verzeichnis gespeichert ist, in das Home-Verzeichnis des Benutzers zu kopieren.

Wenn fp-lib-table nicht gefunden werden kann, wird eine leere Footprint-Bibliotheks-Tabelle im Home-Verzeichnis des Benutzers erzeugt. Wenn das passiert, kann der Benutzer entweder eine fp-lib-table von Hand kopieren oder die Tabelle von Hand einrichten.

Die Standard-Footprint-Bibliotheks-Tabelle enthält viele der Standard-Footprint-Bibliotheken, die als Teil von KiCad installiert werden.

Offensichtlich ist **als Erstes** zu tun diese Tabelle zu bearbeiten (hinzufügen/entfernen von Einträgen) ensprechend Ihrer Tätigkeit und den Bibliotheken, die Sie für alle ihre Projekte benötigen.

(Zu viele Bibliotheken zu laden dauert sehr lange)

5.2.4 Tabelleneinträge hinzufügen

Um eine Footprint-Bibliothek zu benutzen, muss sie zuerst entweder zur globalen oder zur projektspezifischen Tabelle hinzugefügt werden. Die projektspezifische Tabelle ist nur verwendbar, wenn Sie eine Netzlisten-Datei geöffnet haben.

Jeder Bibliothekseintrag muss einen eindeutigen Bezeichner haben.

Dieser Bezeichner braucht in keiner Weise mit dem tatsächlichen Bibliotheksnamen oder -pfad in Verbindung zu stehen. Der Doppelpunkt : darf nirgends im Namen verwendet werden. Jeder Bibliothekseintrag muss einen gültigen Pfad und/oder Dateinamen aufweisen, abhängig von der Art der Bibliothek. Pfade können absolut, relativ oder durch Umgebungsvariablen-Ersetzung definiert werden. (Siehe nächsten Abschnitt)

Der korrekte Plugin-Typ muss ausgewählt worden sein, damit die Bibliothek richtig gelesen werden kann. KiCad unterstützt derzeit das Lesen von KiCad alt, KiCad Pretty, Eagle und GEDA Footprint-Bibliotheken.

Es gibt ebenfalls ein Beschreibungsfeld um eine Beschreibung des Bibliotheksentrags zu ergänzen. Das Optionsfeld wird derzeit nicht verwendet, das hinzufügen von Optionen hat keinen Einfluss, wenn Bibliotheken geladen werden.

Bitte beachten Sie, dass Sie keine doppelten Namen in der gleichen Tabelle verwenden können. Jedoch können Sie doppelte Bezeichner in der globalen und projektspezifischen Footprint-Bibliotheks-Tabelle verwenden. Der projektspezifische Tabelleneintrag wird den globalen Tabelleneintrag überschreiben, wenn doppelte Bezeichner auftreten. Wenn Einträge in der projektspezifischen Tabelle definiert sind, wird eine fp-lib-table Datei in das Verzeichnis der aktuell geöffneten Netzliste geschrieben.

5.2.5 Umgebungsvariablen Ersetzung

Eine der mächtigsten funktionen der Footprint-Bibliotheks-Tabelle ist die Ersetzung der Umgebungsvariablen. Das erlaubt Ihnen, eigene Pfade in Umgebungsvariablen zu definieren, wo Ihre Bibliotheken gespeichert sind. Die Ersetzung von Umgebungsvariablen ist durch die Syntax `${ENV_VAR_NAME}` im Footprint-Bibliotheks-Pfad unterstützt.

Standardmäßig definiert KiCad zur Laufzeit **zwei Umgebungsvariablen**:

- die **KIPRJMOD** Umgebungsvariable. Diese zeigt immer auf das aktuelle Projektverzeichnis und kann nicht geändert werden.
- die **KISYSMOD** Umgebungsvariable. Diese zeigt auf den Ort der Standard-Footprint-Bibliotheken, die mit KiCad installiert wurden.

Sie können `KISYSMOD` überschreiben indem Sie es in den Einstellungen/Pfad konfigurieren festlegen, was es Ihnen erlaubt, eigene Bibliotheken anstelle der KiCad Standard-Footprint-Bibliotheken zu verwenden.

Wenn eine Projekt-Netzlisten-Datei geladen ist, definiert CvPcb `KIPRJMOD` indem es den Dateipfad verwendet (den Projektpfad).

Pcbnew definiert ebenfalls diese Umgebungsvariable, wenn es eine Board-Datei lädt.

Das erlaubt es Ihnen, Bibliotheken im Projektpfad zu speichern, ohne den absoluten Pfad (welcher nicht immer bekannt ist) zur Bibliothek in der projektspezifischen Footprint-Bibliotheks-Datei angeben zu müssen.

5.2.6 Benutzung des GitHub Plugins

The GitHub is a special plugin that provides an interface for read only access to a remote Git Hub repository consisting of pretty (Pretty is name of the KiCad footprint file format) footprints and optionally provides „Copy On Write“ (COW) support for editing footprints read from the GitHub repo and saving them locally. Therefore the „Git Hub“ plugin is for **read only accessing remote pretty footprint libraries at** https://github.com. To add a GitHub entry to the footprint library table the „Library Path“ in the footprint library table row a must be set to a valid GitHub URL.

Zum Beispiel:

https://github.com/liftoff-sr/pretty_footprints

oder

https://github.com/KiCad

Typischerweise haben GitHub-URLs diese Form:

https://github.com/user_name/repo_name

Der „Plugin Typ“ muss auf „Github“ gesetzt sein. Um die „Copy On Write“-Funktion (”COW”) einzuschalten, muss die Option **allow_pretty_writing_to_this_dir** in den „Optionen“-Einstellungen des Footprint-Bibliotheks-Tabellen-Eintrags ergänzt werden. Diese Option ist der „Bibliotheks-Pfad“ für das lokale Speichern von geänderten Kopien von Footprints, die vom GitHub-Repository gelesen wurden. Die Footprints, die in dieses Verzeichnis gespeichert werden, werden mit dem Nur-Lese-Teil des GitHub-Repository kombiniert und die Footprint-Bibliothek zu erzeugen. Wenn diese Option fehlt, dann ist die GitHub-Bibliothek schreibgeschützt. Wenn die Option für eine GitHub-Bibliothek gesetzt ist, dann werden alle Schreibvorgänge auf diese Hybrid-Bibliothek auf das lokale *.pretty Verzeichnis ausgeführt. Beachten Sie, dass der auf github.com angesiedelte teil dieser hybriden ”COW” Bibliothek immer schreibgeschützt ist, was bedeuted, dass Sie nichts löschen können oder irgendeinen Footprint direkt im angegebenen GitHub Repository ändern können. Der kombinierte Bibliotheks-Typ wird weiter ”Github” genannt werden, in allen weiteren Ausführungen, aber er besteht aus sowohl dem lokalen Schreib-/Lese-Teil als auch dem schreibgeschützten Teil (auf github.com).

Die Tabelle unten zeigt einen Footprint-Bibliotheks-Tabellen-Eintrag ohne die Option **allow_pretty_writing_to_this_dir**:

Nickname	Library Path	Plugin Type	Options	Descript.
github	https://github.com/liftoff-sr/pretty_footprints	Github		Liftoff's GH footprints

Die Tabelle unten zeigt einen Footprint-Bibliotheks-Tabellen-Eintrag mit der gesetzen "COW" Option. Betrachten Sie die Benutzung der Umgebungsvariablen ${HOME} nur als ein Beispiel. Das github.pretty Verzeichnis befindet sich im &{HOME}/pretty/ Verzeichnis. Immer wenn Sie die Option **allow_pretty_writing_to_this_dir** verwenden, werden Sie dieses Verzeichnis manuell vorher anlegen müssen und es muss mit der Erweiterung **.pretty** enden.

Nickname	Library Path	Plugin Type	Options	Descript.
github	https://github.com/liftoff-sr/pretty_footprints	Github	allow_pretty_writing_to_this_dir= ${HOME}/pretty/github.pretty	Liftoff's GH footprints

Das Laden von Footprints wird immer vorrangig mit den lokalen Footprints erfolgen, die im Verzeichnis liegen, das durch die Option **allow_pretty_writing_to_this_dir** vorgegeben ist. Sobald sie einmal einen Footprint in das lokale Verzeichnis der "COW" Bibliothek gespeichert haben, indem sie einen Footprint im Footprint-Editor abspeichern, werden keine Updates von GitHub mehr durchgeführt, wenn ein Footprint geladen wird, von dem es eine lokale Speicherung gibt.

Führen Sie immer ein separates lokales *.pretty Verzeichnis für jede GitHub Bibliothek, kombinieren Sie sie nie indem Sie auf das gleiche Verzeichnis mehr als einmal verweisen.

Verwenden Sie ebenfalls nicht das gleiche "COW" (*.pretty)-Verzeichnis in einem Footprint-Bibliotheks-Tabellen-Eintrag. Das wird höchstwahrscheinlich zu einem Durcheinander führen.

Der Wert der Option **allow_pretty_writing_to_this_dir** wird jede Umgebungsvariable erweitern unter Benutzung der ${} Notation um den Pfad in der gleichen Weise zu erzeugen wie die „Bibliotheks-Pfad" Einstellung.

Was ist der Sinn von "COW"? Es ist der Turbo-Lader für das Teilen von Footprints.

Wenn Sie regelmäßig ihre "COW" pretty Footprint Änderungen an den GitHub Repository Betreiber schicken (per Email), können Sie dabei helfen die GitHub-Kopie aktuell zu halten. Mailen Sie einfach die individuellen *.kicad_mod Dateien, die Sie in Ihrem "COW" Verzeichnis finden an den Betreiber des GitHub Repositorys. Nachdem Sie die Bestätigung erhalten haben, dass ihre Änderungen übernommen wurden, können Sie sicher Ihre "COW" Dateien löschen und der auf GitHub aktualisierte schreibgeschützte Teil der Bibliothek wird übernommen. Ihr Ziel sollte es sein, den Teil der "COW" Dateien so klein wie möglich zu halten indem Sie häufig Kopien an die geteilten Hauptbibliothek auf https://github.com schicken.

5.2.7 Nutzungsmuster

Footprint-Bibliotheken können entweder global oder spezifisch für das aktuell geladene Projekt definiert werden. Footprint-Bibliotheken, die in der globalen Tabelle des Benutzers definiert sind, sind immer verfügbar und werden in der fp-lib-table Datei im Home-Verzeichnis des Benutzers gespeichert.

Auf globale Footprint-Bibliotheken kann immer zugegriffen werden, selbst wenn keine Projekt-Netzlisten-Datei geöffnet ist.

Die projektspezifische Footprint-Tabelle ist nur für die aktuell geöffnete Netzlisten-Datei aktiv.

Die projektspezifische Footprint-Bibliotheks-Tabelle wird in die fp-lib-table Datei im Pfad der aktuell geöffneten Netzliste gespeichert. Sie können Bibliotheken frei in jeder der beiden Tabellen definieren.

Es gibt Vor- und Nachteile bei jeder Methode. Sie können alle Ihre Bibliotheken in der globalen Tabelle definieren, was bedeutet, dass sie immer verfügbar sind, wenn Sie sie brauchen. Der Nachteil dabei ist, dass Sie immer eine (große) Menge von Bibliotheken durchsuchen müssen, um den Footprint zu finden, den Sie suchen. Sie können alle Ihre Bibliotheken auf einer projektspezifischen Basis definieren.

Der Vorteil davon ist, dass Sie nur die Bibliotheken definieren müssen, die Sie tatsächlich für das Projekt brauchen, was die Suche verkleinert.

Der Nachteil ist, dass Sie immer daran denken müssen, jede benötigte Footprint-Bibliothek für jedes Projekt hinzuzufügen. Sie können ebenfalls Footprint-Bibliotheken sowohl global und projektspezifisch definieren.

Ein Nutzungsmuster könnte sein, dass Sie ihre meistgenutzten Bibliotheken global definieren und die nur für das Projekt benötigten in der projektspezifischen Bibliotheks-Tabelle. Es gibt keine Einschränkung, wie Sie ihre Bibliotheken definieren.

5.3 Benutzung des Footprint-Bibliotheks-Tabellen-Assistenten

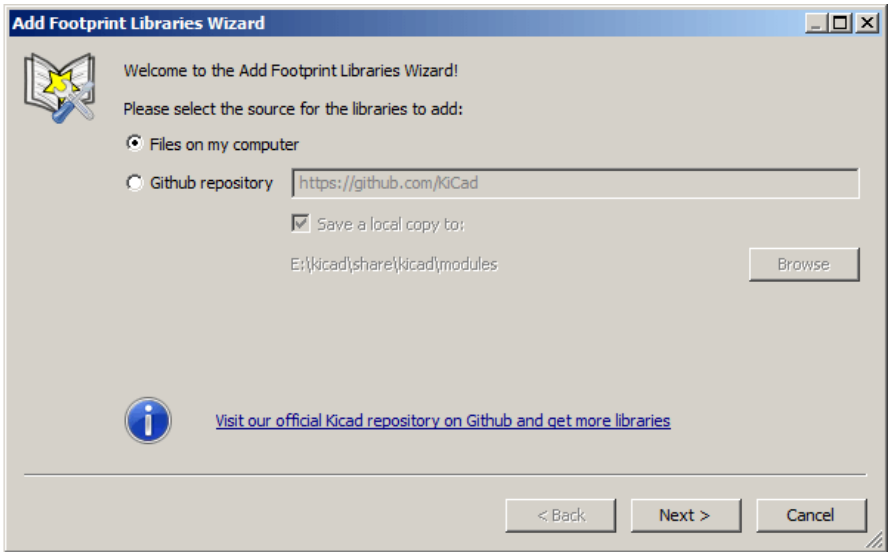
Ein Assistent um Footprint-Bibliotheken zu den Footprint-Bibliotheks-Tabellen hinzuzufügen, ist über den *Footprint-Bibliotheks-Tabelle bearbeiten Dialog* verfügbar.

Beachten Sie, dass Bibliotheken jede Art von Footprint-Bibliothek sein können, die von KiCad unterstützt wird.

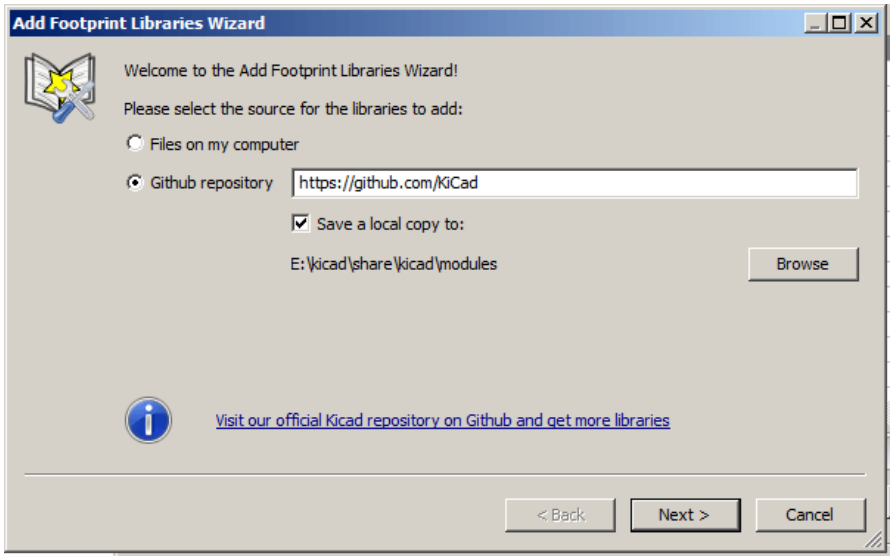
Er kann „lokale“ Bibliotheken oder Bibliotheken von einem GitHub Repository hinzufügen.

Wenn Bibliotheken in einem GitHub Repository liegen, können Sie als ”entfernte” Bibliotheken eingefügt werden oder **heruntergeladen und als *lokale Bibliotheken* hinzugefügt werden**.

Hier ist die lokale Bibliotheks-Option gewählt.

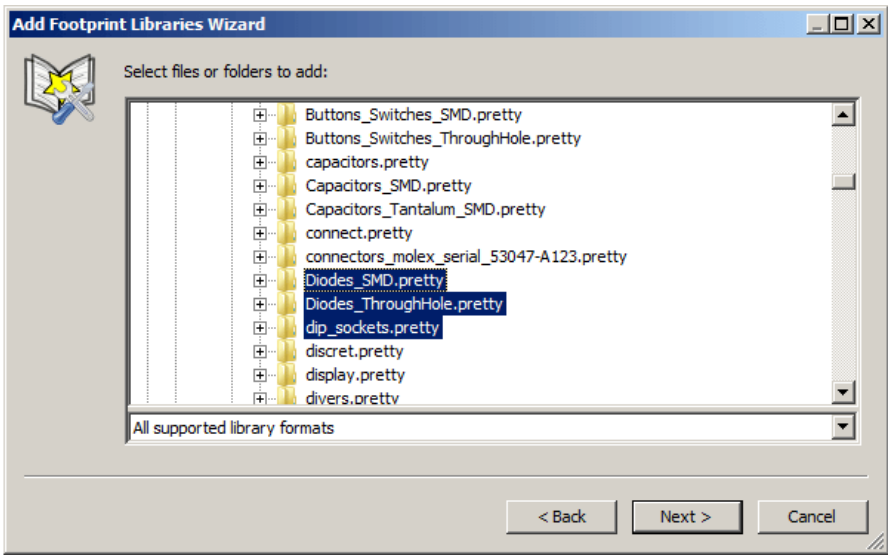


Hier ist die ”entfernte” Bibliotheks-Option gewählt.

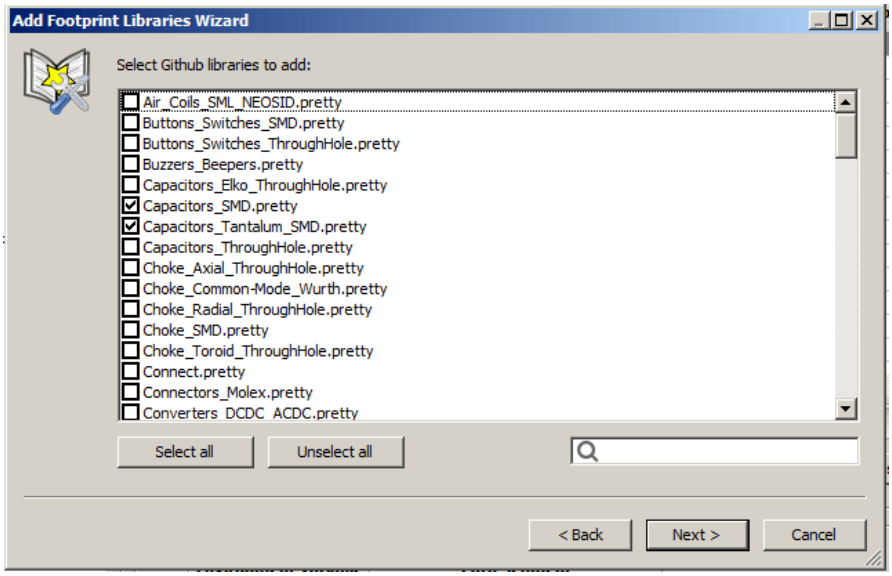


Abhängig von der gewählten Option wird eine dieser Seiten angezeigt, um eine Liste von Bibliotheken auszuwählen:

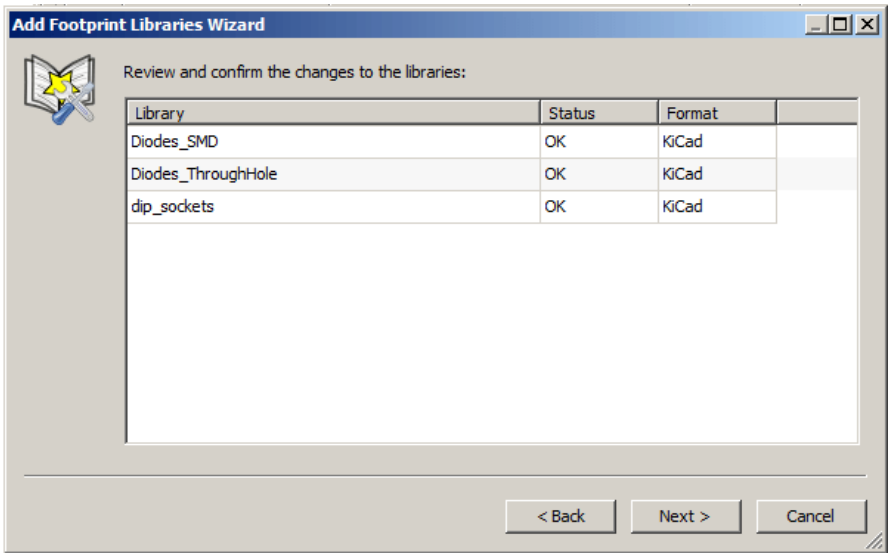
Hier ist die lokale Bibliotheks-Option gewählt.



Hier ist die ”entfernte” Bibliotheks-Option gewählt.



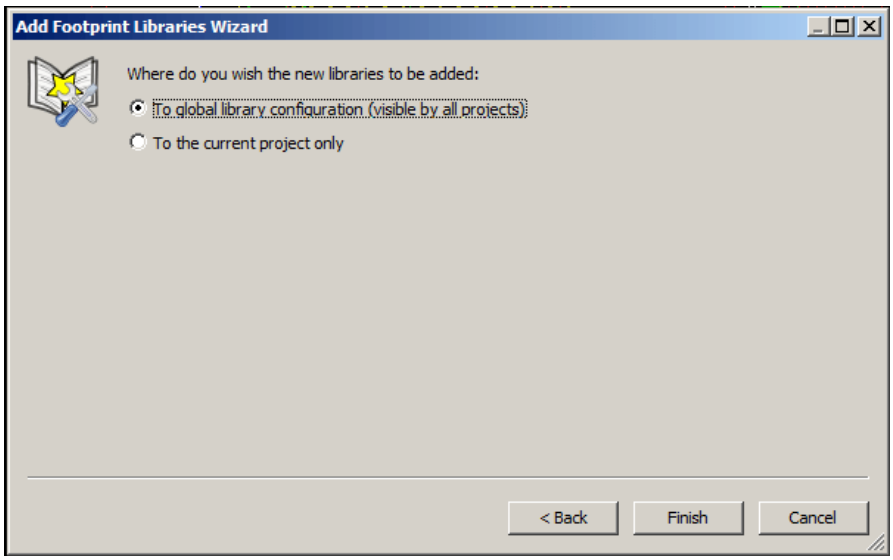
Nachdem ein Satz von Bibliotheken ausgewählt wurde, überprüft die nächste Seite die Auswahl:



Wenn einige ausgewählte Bibliotheken nicht korrekt sind (nicht unterstützt, keine Footprint-Bibliothek…) werden sie als „UNGÜLTIG“ gekennzeichnet.

Die letzte Auswahl ist die zu füllende Footprint-Bibliotheks-Tabelle:

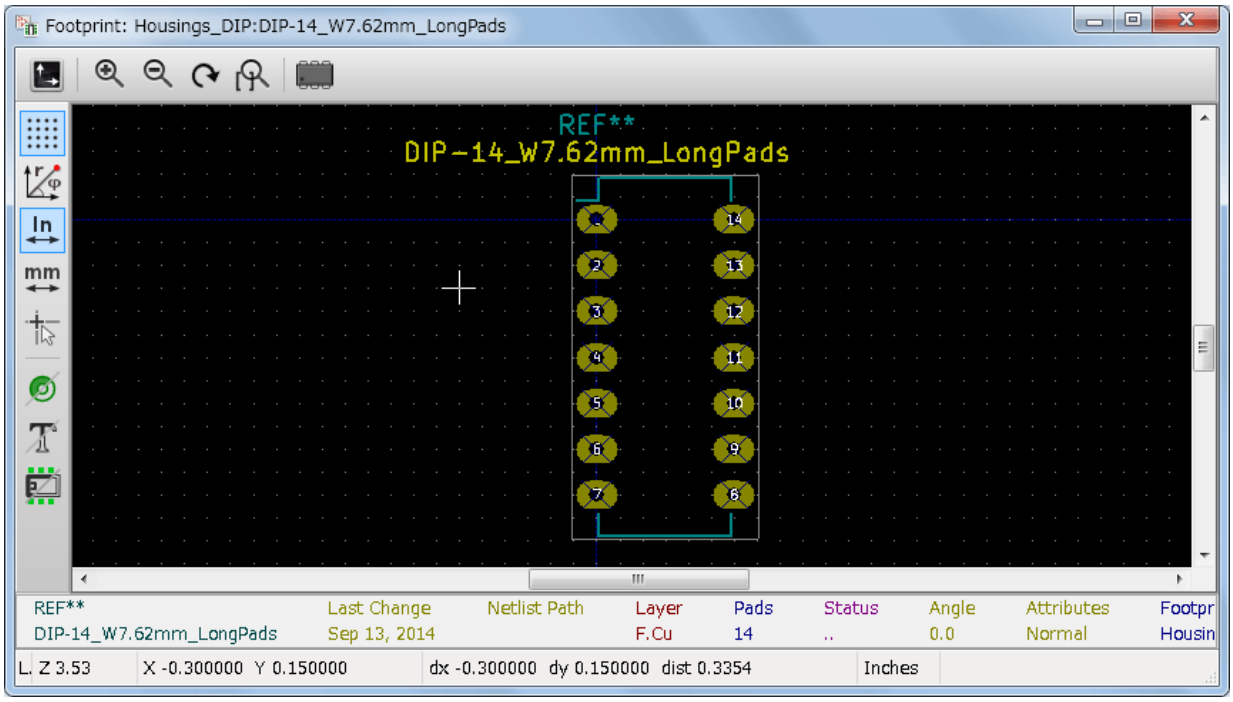
- Die globale Tabelle
- Die lokale Tabelle (die projektspezifische Tabelle)



6 Den aktuellen Footprint anzeigen

6.1 Der Zeige Footprint Befehl

Der Zeige Footprint Befehl zeigt den aktuell ausgewählten Footprint im *Footprint* Fenster. Ein 3D Modell des Bauteils kann angezeigt werden, wenn einer erstellt und dem Footprint zugewiesen wurde. Unten ist das Fooprint-Anzeige-Fenster..



6.1.1 Status-Balken Information

Die Statuszeile befindet sich am unteren Ende des neuen CvPcb Hauptfensters und stellt nützliche Informationen für den Benutzer bereit. Die folgende Tabelle definiert die Inhalte jedes Bereichs in der Statuszeile.

Left	Component count: total, unassigned
Middle	Filter list of the selected component
Right	Filtering mode and count of available footprints

6.1.2 Tastatur-Befehle

F1	Hineinzoomen
F2	Herauszoomen
F3	Anzeige erneuern
F4	Curser in die Mitte des Anzeigefensters bewegen
Pos 1	Footprint ins Anzeigefenster einpassen
Leerzeichen	Setze relative Koordinaten zur aktuellen Cursor-Position
Rechter Pfeil	Cursor eine Rasterposition nach rechts bewegen

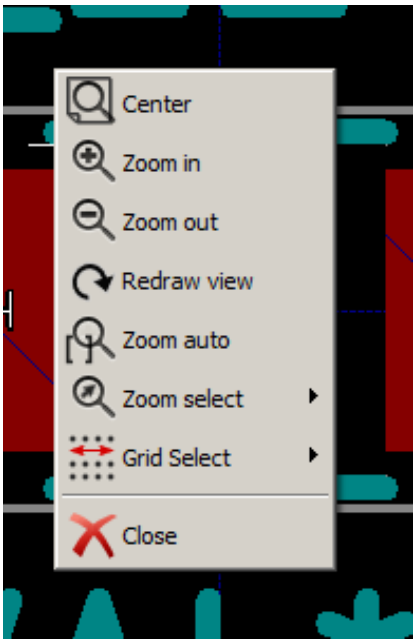
Linker Pfeil	Cursor eine Rasterposition nach links bewegen
Pfeil hoch	Cursor eine Rasterpositiono nach oben bewegen
Pfeil runter	Cursor eine Rasterposition nach unten bewegen

6.1.3 Maus-Befehle

Scroll-Rad	Hinein- und Herauszoomen an der aktuellen Cursor-Position
Strg + Scroll-Rad	Bildinhalt nach rechts und links verschieben
Umschalt + Scroll-Rad	Bildinhalt nach oben und unten verschieben
Rechte Maustaste	Kontextmenü öffnen

6.1.4 Kontextmenü

Wird bei einem Rechtsklick mit der Maus angezeigt:



Zoomauswahl (Zoom auswählen)	Direkte Auswahl des Anzeige-Zooms.
Rasterauswahl (Raster auswählen)	Direkte Auswahl des Rasters.

6.1.5 Horizontale Werkzeugleiste

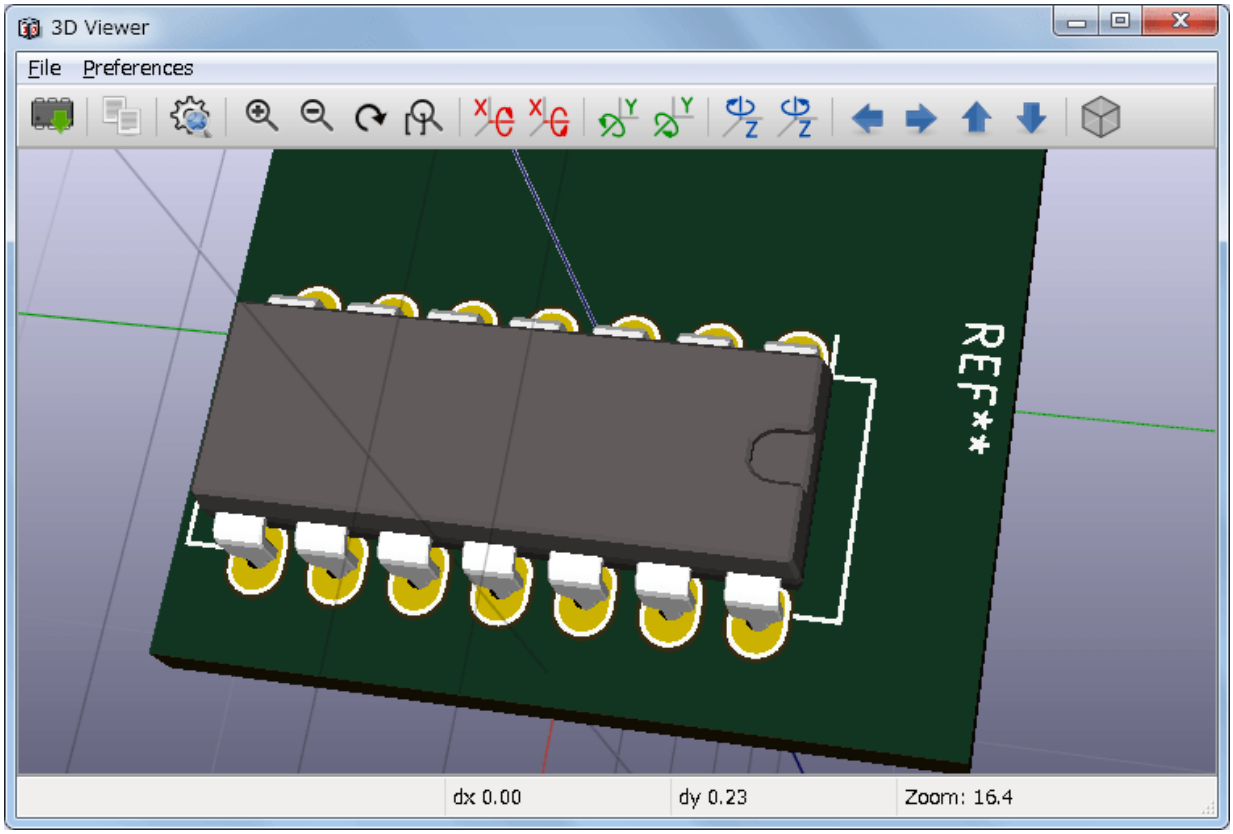
	Zeige Dialog für Anzeigeeinstellungen
	Hineinzoomen
	Herauszoomen

	Anzeie Aktualisieren
	Zeichnung in Anzeigefläche einpassen
	Öffne 3D-Modell-Anzeige

6.1.6 Senkrechte Werkzeugleiste

	Raster anzeigen oder verbergen
	Koordinaten in Polar- oder Rechteck-Notation anzeigen
	Koordinaten in Zoll anzeigen
	Koordinaten in Millimeter anzeigen
	Zeiger-Stil umschalten
	Umschalten zwischen Umriss- oder Normaldarstellung für Lötflächen.
	Umschalten zwischen Umriss- oder Normalmodus für Text
	Umschalten zwischen Umriss- und Normalmodus für Bauteilränder

6.2 Anzeige des aktuellen 3D-Modells



6.2.1 Maus-Befehle

Scroll-Rad	Hinein- und Herauszoomen an der aktuellen Cursor-Position
Strg + Scroll-Rad	Bildinhalt nach rechts und links verschieben
Umschalt + Scroll-Rad	Bildinhalt nach oben und unten verschieben

6.2.2 Horizontale Werkzeugleiste

	3D-Modell neu laden
	3D-Bild in Zwischenablage kopieren
	3D-Anzeigeeinstellungen setzen
	Hineinzoomen
	Herauszoomen
	Anzeie Aktualisieren

	Zeichnung in Anzeigefläche einpassen
	Rückwärts entlang der X-Achse rotieren
	Vorwärts entlang der X-Achse rotieren
	Rückwärts entlang der Y-Achse rotieren
	Vorwärts entlang der Y-Achse rotieren
	Rückwärts entlang der Z-Achse rotieren
	Vorwärts entlang der Z-Achse rotieren
	Bildinhalt nach links verschieben
	Bildinhalt nach rechts verschieben
	Bildinhalt nach oben verschieben
	Bildinhalt nach unten verschieben
	Ein- und Ausschalten des orthogonalen Anzeigemodus

7 CvPcb nutzen um Bauteile mit Footprints zu verbinden

7.1 Manuelle Zuweisung von Footprints zu Bauteilen

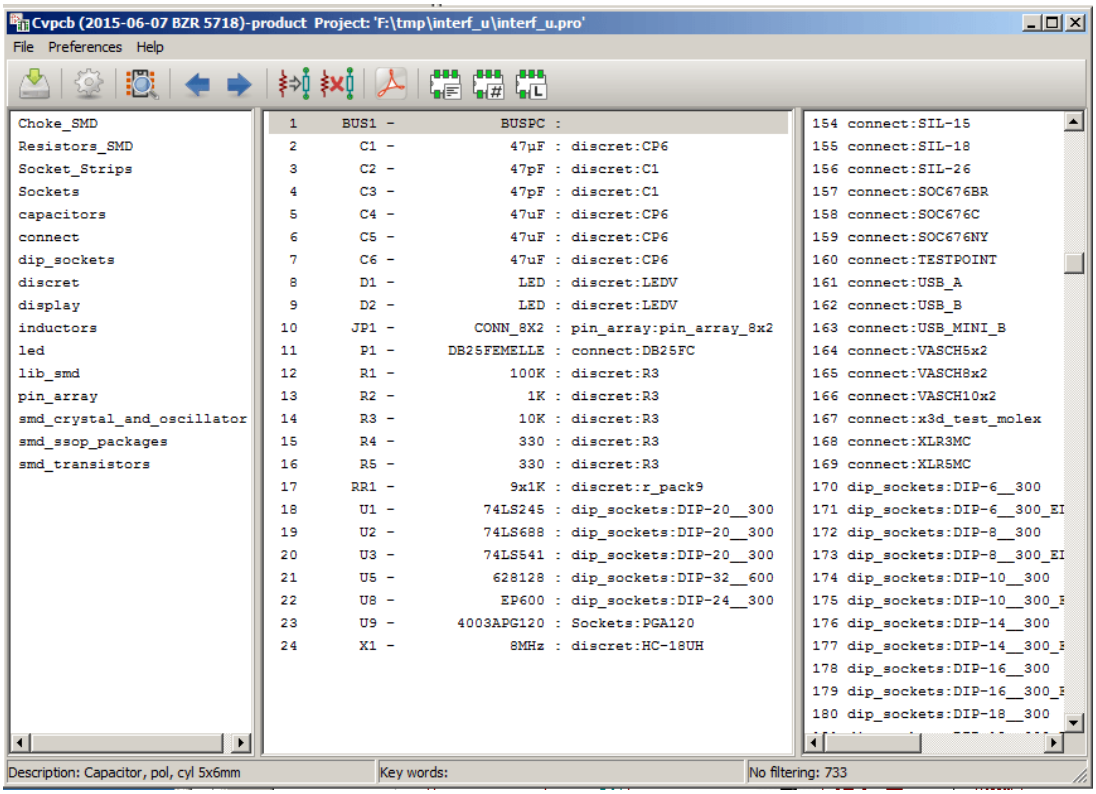
Um manuell einen Footprint mit einem Bauteil zu verbinden, wählen Sie zuerst ein Bauteil im Bauteil-Bereich. Dann wählen Sie einen Footprint im Footprint-Bereich indem Sie mit der linken Maustaste doppelt auf den Namen des gewünschten Footprints klicken. Das nächste Bauteil ohne Zuweisung in der Liste wird automatisch ausgewählt. Den Footprint zu ändern, wird in der gleichen Weise durchgeführt.

7.2 Footprint-Liste filtern

Wenn das ausgewählte Bauteil und/oder Bibliothek hervorgehoben wird, wenn ein oder mehr Filter-Optionen eingeschaltet werden, wird die angezeigte Footprint-Liste in CvPcb entsprechend gefiltert.

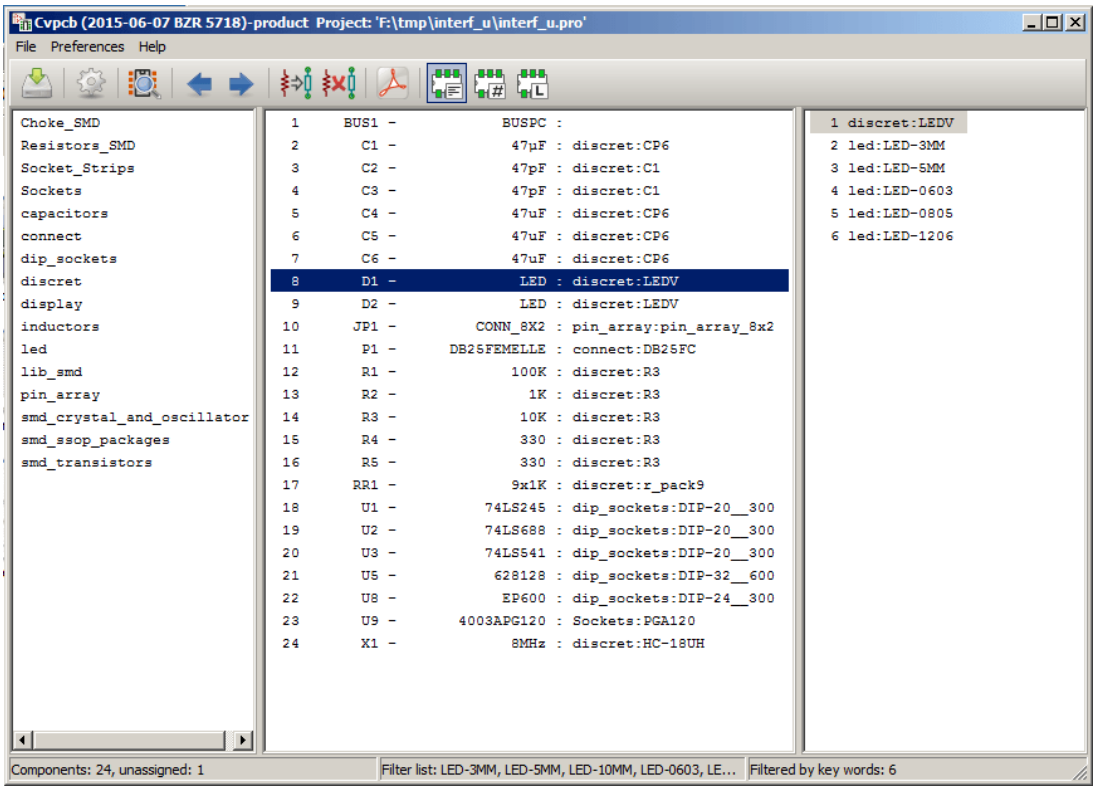
Die Icons  schalten die Filterfunktion ein und aus. Wenn die Filterung ausgeschaltet ist, wird die gesamte Footprint-Liste angezeigt.

Ohne Filterung:

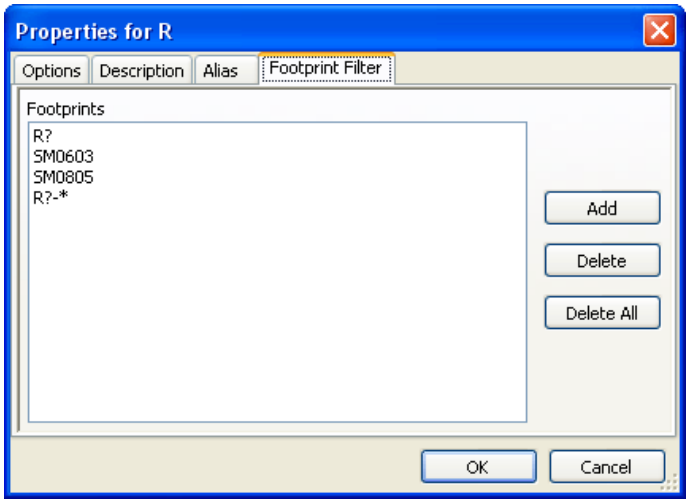


Gefiltert anhand der Liste von Footprint-Filtern die dem ausgewählten Bauteil zugewiesen sind. Die Bateilfilter sind im mittleren Bereich der Statusleiste am unteren Rand des Hauptfensters angezeigt.

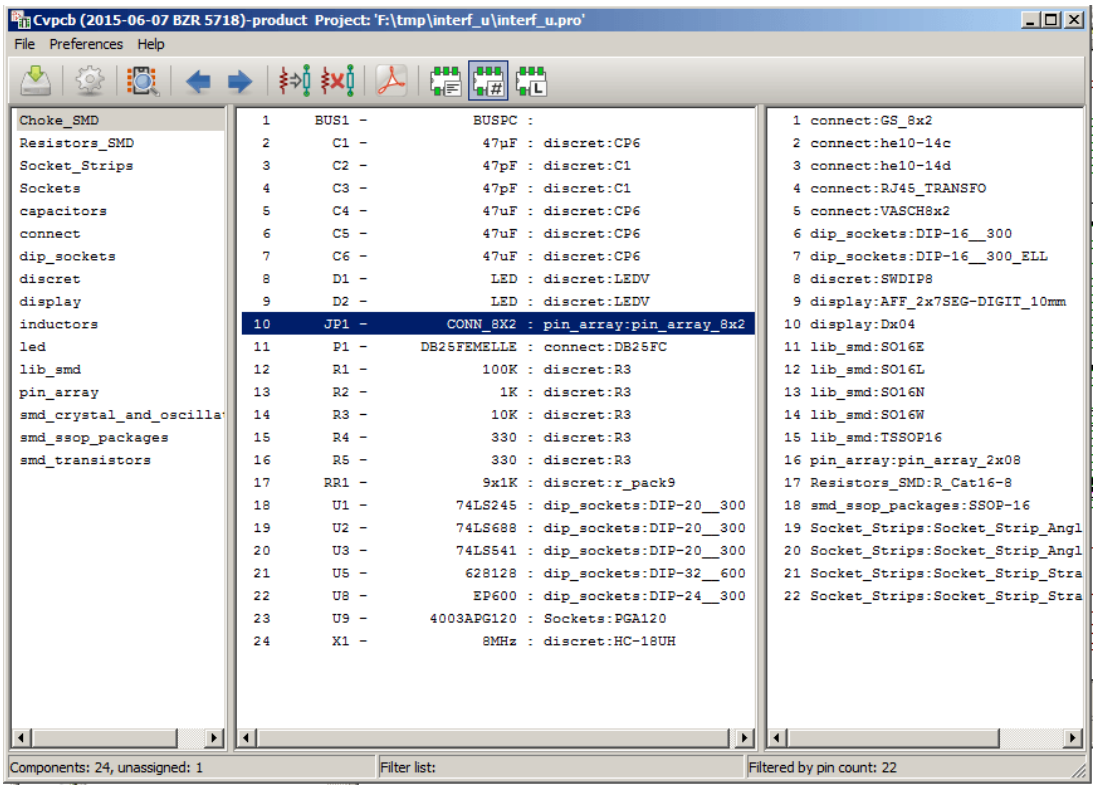
Gefiltert anhand des Footprint-Filters des ausgewählten Bauteils:



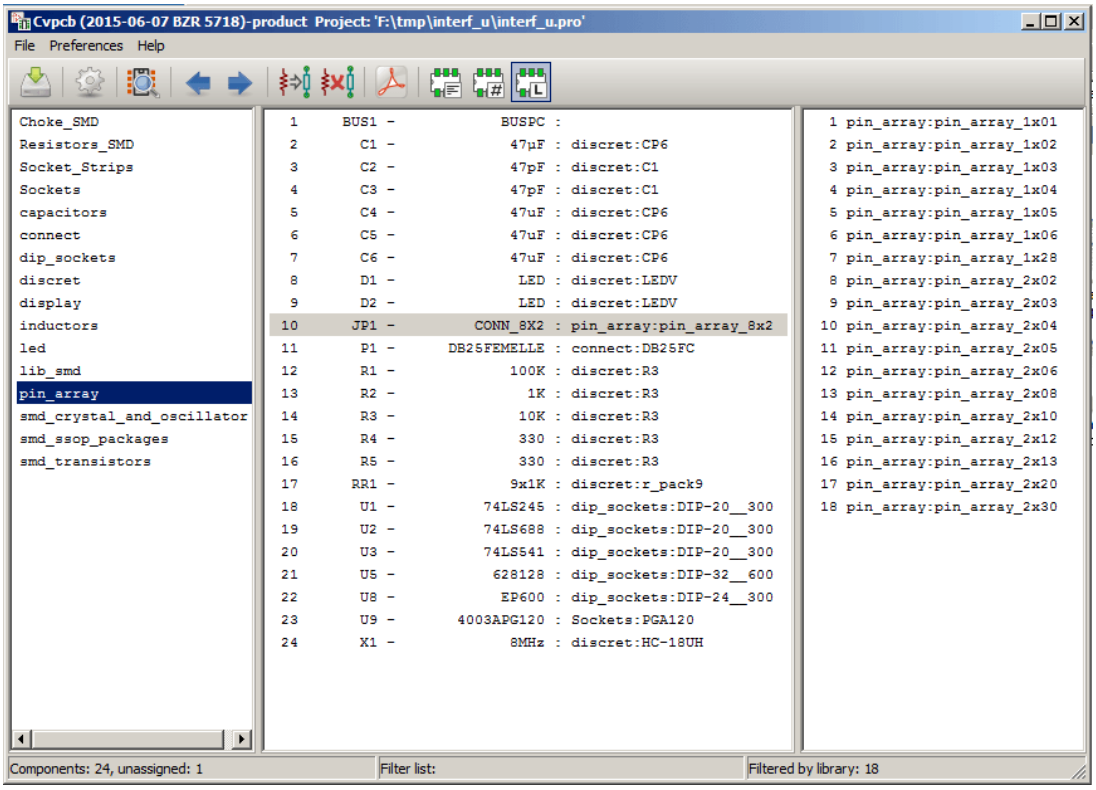
Im Bauteil-Bibliotheks-Editor in Eeschema wurde die Footprint-Liste gesetzt, unter Nutzung der Einträge im Footprint-Filter-Tab des Bauteil-Eigenschaften-Dialogs wie unten gezeigt:



Gefiltert nach Pin-Anzahl der gewählten Bauteils:

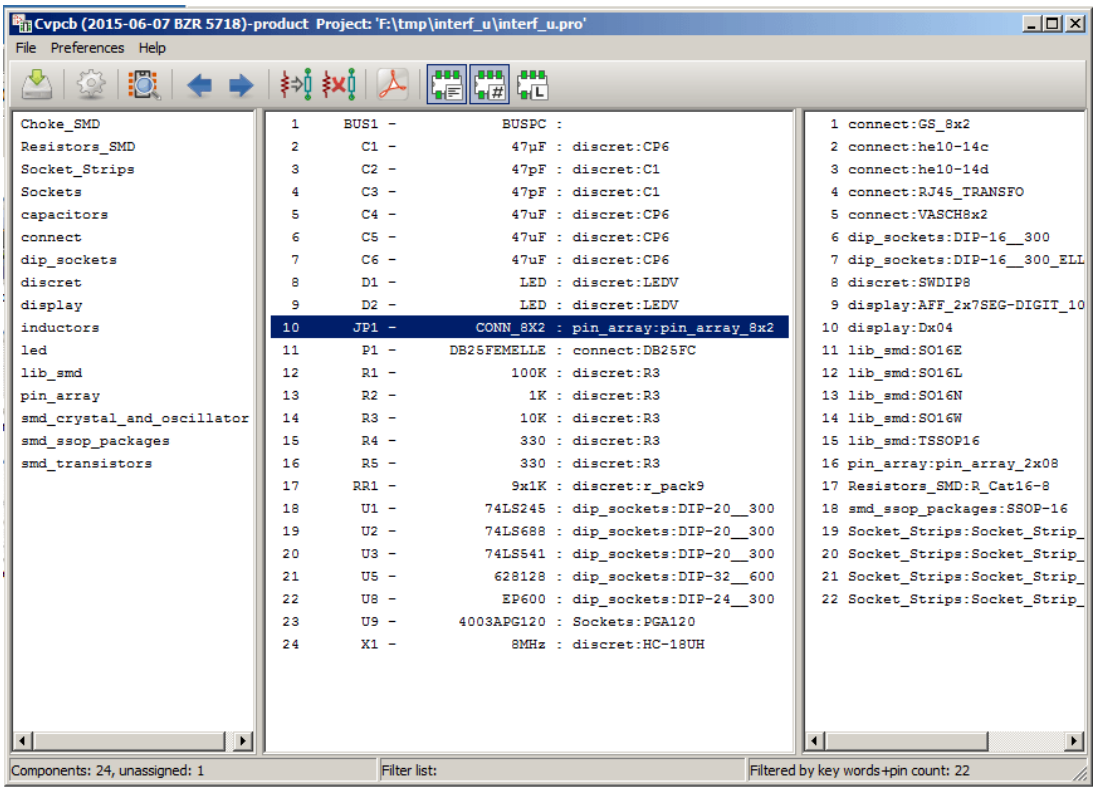


Gefiltert nach der ausgewählten Bibliothek:



Die Filterung kann kombiniert werden, um komplexere Filter zu bilden, um die Anzahl der im Footprint-Bereich angezeigten Footprints zu reduzieren.

Gefiltert nach ausgewähltem Bauteil Pin-Anzahl und Bauteil-Filter:


Cvpcb (2015-06-07 BZR 5718)-product Project: 'F:\tmp\interf_u\interf_u.pro'
File Preferences Help
Choke_SMD
Resistors_SMD
Socket_Strips
Sockets
capacitors
connect
dip_sockets
discret
display
inductors
led
lib_smd
pin_array
smd_crystal_and_oscillator
smd_ssop_packages
smd_transistors
1 BUS1 - BUSPC :
2 C1 - 47µF : discret:CP6
3 C2 - 47pF : discret:C1
4 C3 - 47pF : discret:C1
5 C4 - 47uF : discret:CP6
6 C5 - 47uF : discret:CP6
7 C6 - 47uF : discret:CP6
8 D1 - LED : discret:LEDV
9 D2 - LED : discret:LEDV
10 JP1 - CONN_8X2 : pin_array:pin_array_8x2
11 P1 - DB25FEMELLE : connect:DB25FC
12 R1 - 100K : discret:R3
13 R2 - 1K : discret:R3
14 R3 - 10K : discret:R3
15 R4 - 330 : discret:R3
16 R5 - 330 : discret:R3
17 RR1 - 9x1K : discret:r_pack9
18 U1 - 74LS245 : dip_sockets:DIP-20__300
19 U2 - 74LS688 : dip_sockets:DIP-20__300
20 U3 - 74LS541 : dip_sockets:DIP-20__300
21 U5 - 628128 : dip_sockets:DIP-32__600
22 U8 - EP600 : dip_sockets:DIP-24__300
23 U9 - 4003APG120 : Sockets:PGA120
24 X1 - 8MHz : discret:HC-18UH
1 connect:GS_8x2
2 connect:he10-14c
3 connect:he10-14d
4 connect:RJ45_TRANSFO
5 connect:VASCH8x2
6 dip_sockets:DIP-16__300
7 dip_sockets:DIP-16__300_ELL
8 discret:SWDIP8
9 display:AFF_2x7SEG-DIGIT_10
10 display:Dx04
11 lib_smd:SO16E
12 lib_smd:SO16L
13 lib_smd:SO16N
14 lib_smd:SO16W
15 lib_smd:TSSOP16
16 pin_array:pin_array_2x08
17 Resistors_SMD:R_Cat16-8
18 smd_ssop_packages:SSOP-16
19 Socket_Strips:Socket_Strip_
20 Socket_Strips:Socket_Strip_
21 Socket_Strips:Socket_Strip_
22 Socket_Strips:Socket_Strip_
Components: 24, unassigned: 1
Filter list:
Filtered by key words+pin count: 22

8 Automatische Zuweisung

8.1 Äquivalenz-Dateien

Äquivalenz-Dateien erlauben eine automatische Zuweisung von Footprints zu Bauteilen.

Sie listen den Namen des zugehörigen Footprints gemäß des Namens (*value field*) des Bauteils. Diese Dateien haben typischerweise die **.equ** Datei-Erweiterung.

Sie sind reine Textdateien und können mit einem normalen Texteditor bearbeitet werden und müssen vom Nutzer erstellt werden.

8.2 Äquivalenz-Datei Format

Äquivalenz-Datei besteht aus einer Zeile für jedes Bauteil. Jede Zeile hat folgende Struktur:

‚Bauteilwert' ‚Footprint-Name'

Jeder Name muss mit einem einfachen Anführungszeigen ' eingeschlossen werden und der Bauteil- und Footprint-Name müssem mit einem oder mehreren Leerzeichen getrennt sein.

Beispiel:

Wenn das U3 Bauteil der Schaltkreis 14011 ist und sein Footprint 14DIP300 ist, lautet die Zeile:

```
,14011' ,14DIP300'
```

Jede Zeile die mit einem **#** beginnt ist ein Kommentar.

Hier ist ein Beispiel einer Äquivalenz-Datei:

```
#Integrierte Schaltkreise (SMD):
'74LV14' 'SO14E'
'74HCT541M' 'SO20L'
'EL7242C' 'SO8E'
'DS1302N' 'SO8E'
'XRC3064' 'VQFP44'
'LM324N' 'S014E'
'LT3430' 'SSOP17'
'LM358' 'SO8E'
'LTC1878' 'MSOP8'
'24LC512I/SM' 'SO8E'
'LM2903M' 'SO8E'
'LT1129_SO8' 'SO8E'
'LT1129CS8-3.3' 'SO8E'
'LT1129CS8' 'SO8E'
'LM358M' 'SO8E'
'TL7702BID' 'SO8E'
'TL7702BCD' 'SO8E'
'U2270B' 'SO16E'
#Xilinx
'XC3S400PQ208' 'PQFP208'
```

```
'XCR3128-VQ100' 'VQFP100'
'XCF08P' 'BGA48'

#upro
'MCF5213-LQFP100' 'VQFP100'

#Spannungsregler
'LP2985LV' 'SOT23-5'
```

8.3 Automatische Zuweisung von Footprints zu Bauteilen

Klicken Sie auf den "Automatische Footprint Zuordung"-Button in der obersten Werkzeugleiste um eine Äquivalenz-Datei abzuarbeiten.

Alle Bauteile die über ihren Wert in der ausgewählten Äquivalenz-Datei (.equ) gefunden wurden, bekommen ihren Footprint automatisch zugewiesen.*