

# CvPcb

**19 października 2016**

# Spis treści

**1 Wprowadzenie do CvPcb** **2**

**2 Charakterystyka aplikacji** **2**

2.1 Przypisywanie manualne lub automatyczne . . . 2

**3 Uruchamianie CvPcb** **2**

**4 Polecenia CvPcb** **3**

4.1 Główne okno aplikacji . . . 3

4.2 Główny pasek narzędzi . . . 3

4.3 Polecenia dostępne z klawiatury . . . 4

4.4 Konfigurowanie CvPcb . . . 5

**5 Zarządzanie bibliotekami footprintów** **6**

5.1 Ważna informacja: . . . 6

5.2 Tabele bibliotek footprintów . . . 6

5.2.1 Globalna tabela bibliotek footprintów . . . 7

5.2.2 Lokalna tabela bibliotek footprintów zależna od projektu . . . 7

5.2.3 Konfiguracja początkowa . . . 7

5.2.4 Dodawanie nowych wpisów w tabeli . . . 8

5.2.5 Pobieranie wartości ze zmiennych systemowych . . . 8

5.2.6 Używanie wtyczki GitHub . . . 8

5.2.7 Generalne zalecenia przy używaniu tabeli bibliotek . . . 10

5.3 Użycie Kreatora Tabeli Bibliotek Footprintów . . . 11

**6 Podgląd bieżącego footprintu** **14**

6.1 Polecenie Podgląd footprintu . . . 14

6.1.1 Informacje na pasku statusu . . . 14

6.1.2 Skróty klawiaturowe . . . 14

6.1.3 Polecenia związane z myszą . . . 15

6.1.4 Menu kontekstowe . . . 15

6.1.5 Górny pasek narzędziowy . . . 15

6.1.6 Lewy pasek narzędziowy . . . 16

6.2 Podgląd 3D . . . 17

6.2.1 Polecenia związane z myszą . . . 17

6.2.2 Górny pasek narzędziowy . . . 17

**7 Użycie CvPcb do przypisywania footprintów komponentom** **19**

7.1 Ręczne przypisywanie footprintów do komponentów . . . 19

7.2 Filtrowanie listy footprintów . . . 19

**8 Przypisywanie automatyczne** **24**

8.1 Pliki przypisań . . . . . 24

8.2 Format plików . . . . . 24

8.3 Automatyczne przypisywnie footprintów do komponentów . . . . . 25

*Podręcznik użytkownika*

**Prawa autorskie**

Copyright © 2010-2016. Ten dokument jest chroniony prawem autorskim. Lista autorów znajduje się poniżej. Możesz go rozpowszechniać oraz modyfikować na zasadach określonych w General Public License (http://www.gnu.org/licenses/gpl.html), wersja 3 lub późniejsza, albo określonych w Creative Commons Attribution License (http://creativecommons.org/licenses/by/-3.0/), wersja 3.0 lub późniejsza.

Wszystkie znaki towarowe użyte w tym dokumencie należą do ich właścicieli.

**Współtwórcy**

Jean-Pierre Charras, Fabrizio Tappero, Wayne Stambaugh.

**Tłumaczenie**

Kerusey Karyu <keruseykaryu@o2.pl>, 2014-2015.

**Kontakt**

Wszelkie zauważone błędy, sugestie lub nowe wersje dotyczące tego dokumentu prosimy kierować do:

- W sprawie dokumentacji: https://github.com/KiCad/kicad-doc/issues
- W sprawie oprogramowania: https://bugs.launchpad.net/kicad
- W sprawie tłumaczeń interfejsu użytkownika (i18n): https://github.com/KiCad/kicad-i18n/issues

**Data publikacji i wersja oprogramowania**

Opublikowano 22 Maj 2015.

# 1 Wprowadzenie do CvPcb

CvPcb pozwala na przypisanie każdemu komponentowi jaki występuje na schemacie nazwy footprintu, który będzie go reprezentował na obwodzie drukowanym. To przypisanie będzie dodane do listy sieci utworzonej przez program Eeschema.

Lista sieci utworzona przez Eeschema określa jakie footprinty (fizyczne odzwierciedlenie komponentów) są przypisane do poszczególnych komponentów na schemacie tylko wtedy, gdy pola Footprint tych komponentów zostały wcześniej określone.

Tak jest w przypadku gdy footprinty zostały przypisane podczas rysowania schematu poprzez wstępne wypełnienie pól *Footprint*, lub ustawione wcześniej w bibliotekach i przeniesione na schemat podczas ładowania symbolu.

CvPcb dostarcza o wiele bardziej wygodną w użyciu metodę przypisywania footprintów do komponentów w czasie rysowania schematu. Pozwala on na filtrację list wyboru, podgląd footprintów oraz ich modeli 3D by pomóc wybrać odpowiedni footprint dla każdego z komponentów.

Footprinty mogą być przypisane komponentom manualnie lub automatycznie poprzez skrypty przypisań (pliki .equ). Pliki te stanowią bazę, z której pobierane są dane o domyślnych footprintach dla określonych komponentów.

Ten interaktywny proces jest znacznie prostszy niż bezpośrednie przypisywanie tych informacji z poziomu schematu.

CvPcb pozwala na przeglądanie list dostępnych footprintów oraz podgląd ich wyglądu na ekranie tak, by właściwe footprinty zostały przypisane właściwym komponentom.

**Może być uruchomiony tylko z Eeschema**, za pomocą górnego paska narzędzi, zarówno gdy Eeschema został uruchomiony z Menadżera projektu lub gdy Eeschema został uruchomiony jako samodzielna aplikacja.

Uruchomienie CvPcb z poziomu Eeschema wywołanego z Menedżera projektu jest generalnie lepszym rozwiązaniem, ponieważ:

- CvPcb potrzebuje pliku konfiguracji projektu by wiedzieć jakie biblioteki footprintów należy załadować.
- CvPcb inicjalizuje pola Footprint komponentów na bieżącym projekcie schematu. Jest to możliwe tylko, gdy plik projektu jest w tej samej lokalizacji co otwarty schemat.

Uruchomienie CvPcb z programu Eeschema wywołanego przez Menadżera projektu automatycznie zapewania taki stan rzeczy.



**Ostrzeżenie**
W tej chwili **można** uruchomić CvPcb z uruchomionego samodzielnie Eeschema, lecz należy mieć na uwadze, że dowolny schemat, który nie posiada pliku projektu w tej samej lokalizacji może wykazywać braki w komponentach z powodu braku bibliotek w których są one zapisane. W związku z tym nie będą pokazane w CvPcb. Jeśli nie ma pliku `fp-lib-table` w tej samej lokalizacji co otwarty schemat, także żadne biblioteki skojarzone z projektem nie będą dostępne.

# 2 Charakterystyka aplikacji

## 2.1 Przypisywanie manualne lub automatyczne

CvPcb pozwala na przypisywanie interaktywne (manualne) oraz automatyczne z pomocą plików skryptów przypisań.

# 3 Uruchamianie CvPcb

**CvPcb jest uruchamiany wyłącznie poprzez edytor schematów Eeschema**, za pomocą narzędzia:

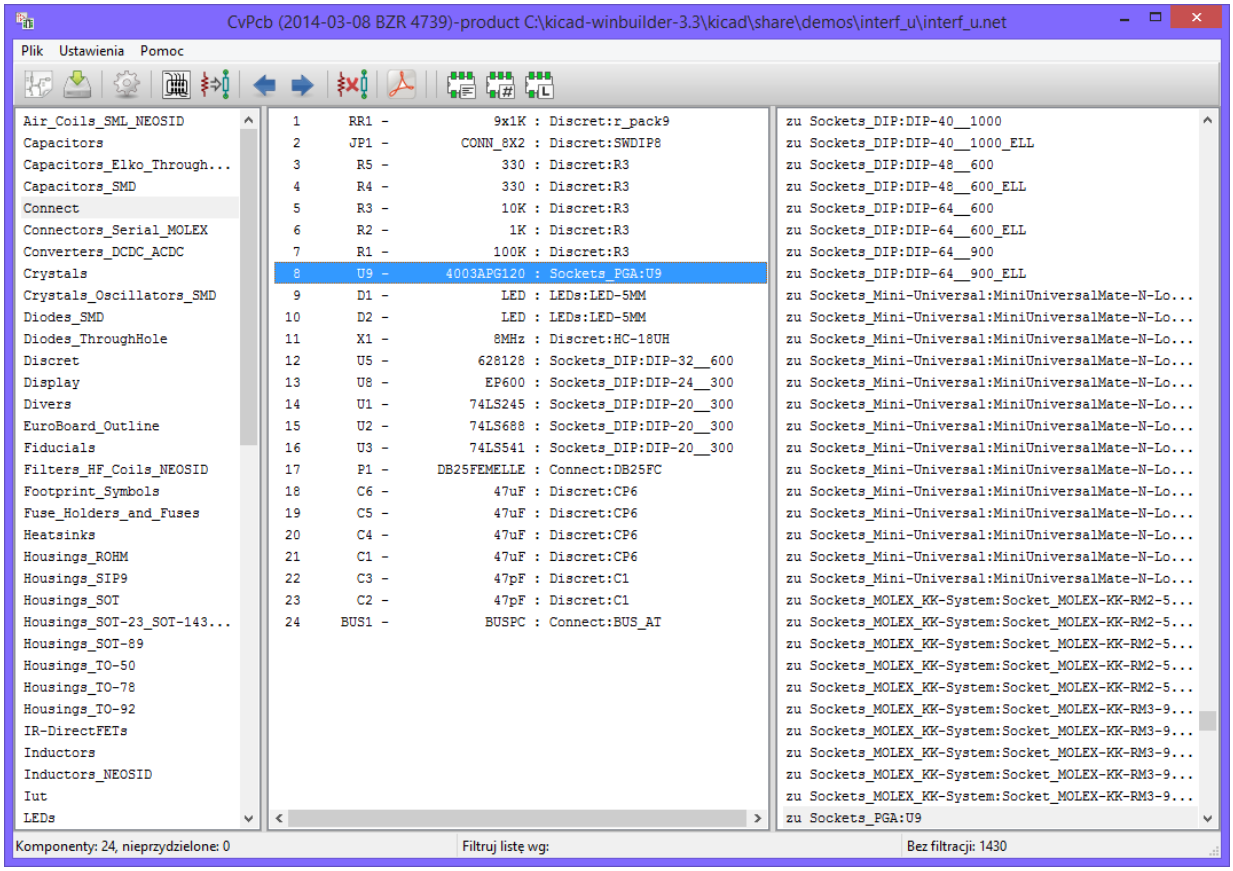


Eeschema automatycznie przekazuje właściwe dane (listę komponentów i footprintów) do CvPcb. Nie ma potrzeby wykonywania aktualizacji (chyba, że niektóre z nowych komponentów nie zostały jeszcze ponumerowane), wystarczy uruchomić CvPcb.

# 4 Polecenia CvPcb

## 4.1 Główne okno aplikacji

Poniższa ilustracja pokazuje widok głównego okna programu CvPcb.




Panel listy bibliotek (z lewej strony) zawiera listę dostępnych w projekcie bibliotek. Panel ten współpracuje z opcją filtrowania według bibliotek. Panel komponentów (w środku) zawiera listę komponentów odczytanych z listy sieci. Panel footprintów (z prawej strony) zawiera listę footprintów odczytanych z dostępnych bibliotek. Zawartość tej listy może być filtrowana. Panel komponentów może być pusty jeśli nie została odczytana lista sieci, tak samo jak panel footprintów jeśli nie znaleziono żadnych bibliotek footprintów lub filtr nie pasuje do żadnego z nich.

## 4.2 Główny pasek narzędzi



Znaczenie poszczególnych przycisków jest następujące:

| | |
|---|---|
|  | Transferuje bieżące przypisania do Eeschema (są to wartości pola Footprint). |
|  | Uruchamia menu konfiguracji CvPcb. |
|  | Wyświetla bieżący footprint (czyli ten który obecnie jest wskazany na liście dostępnych footprintów). |

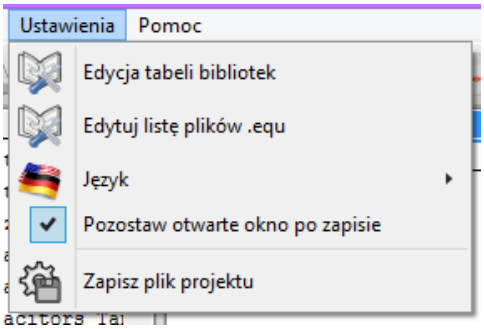
| | |
|---|---|
|  | Automatycznie przeskakuje do poprzedniego elementu, któremu jeszcze nie został przypisany żaden footprint. |
|  | Automatycznie przeskakuje do następnego elementu, któremu jeszcze nie został przypisany żaden footprint. |
|  | Automatycznie przypisuje nazwy footprintów korzystając z plików przypisań automatycznych `.equ`. Użycie tego narzędzia domyślnie przyjmuje, że te pliki są dostępne. |
|  | Kasuje wszystkie przypisania. |
|  | Wyświetla dokumentację footprintu, jeśli istnieje. |
|  | Włącza lub wyłącza filtrowanie za pomocą filtrów footprintów zapisanych we właściwościach wybraneego komponentu. |
|  | Włącza lub wyłącza filtrowanie za pomocą filtra używającego klucza w postaci ilości wyprowadzeń wybraneego komponentu. |
|  | Włącza lub wyłącza filtrowanie za pomocą filtra używającego klucza w postaci wybranej biblioteki. |

## 4.3 Polecenia dostępne z klawiatury

Poniższa tabela zawiera listę klawiszy i powiązanych z nimi akcji dla okna głównego:

| | |
|---|---|
| Strzałka w prawo / Tab | Aktywuje kolejny panel znajdujący się na prawo od aktualnie aktywnego panelu. Wraca do pierwszego panelu gdy aktywnym panelem jest ostatni panel. |
| Strzałka w lewo | Aktywuje poprzedni panel znajdujący się na lewo od aktualnie aktywnego panelu. Wraca do ostatniego panelu gdy aktywnym panelem jest pierwszy panel. |
| Strzałka w górę | Wybiera poprzedni element na obecnie wybranej liście. |
| Strzałka w dół | Wybiera następny element na obecnie wybranej liście. |
| Page Up | Wybiera pierwszy element na górze widocznej części listy. |
| Page Down | Wybiera ostatni element na dole widocznej części listy. |
| Home | Wybiera pierwszy element z bieżąco wybranej listy. |
| End | Wybiera ostatni element z bieżąco wybranej listy. |

## 4.4 Konfigurowanie CvPcb




CvPcb może być automatycznie zamknięty po zapisaniu pliku z przypsaniami lub nie.

Uruchomienie menu konfiguracji bibliotek powoduje otwarcie następującego okna dialogowego.

W zależności od wersji programu CvPcb, są dwa różne modele zarządzania bibliotekami:

- Zarządzanie bibliotekami starszego typu, używa plików `.mod` oraz list plików bibliotek.
- Nowy format “Pretty” używa jednego pliku dla każdego footprintu. Używana jest zatem lista folderów. Każdy folder (o nazwie zakończonej przez `.pretty`) to osobna biblioteka. Gdy używany jest nowy model zarządzania, można użyć również natywnych plików bibliotek z programu gEDA/gPCB oraz EAGLE XML (wersje 6.x lub późniejsze).

# 5 Zarządzanie bibliotekami footprintów

## 5.1 Ważna informacja:

***Sekcja ta dotyczy tylko tych wersji programu KiCad, które zostały wydane przed grudniem 2013***

## 5.2 Tabele bibliotek footprintów

Od grudnia 2013 r., Pcbnew oraz CvPcb używa nowego modelu zarządzania bibliotekami, bazującego na ***tabelach bibliotek footprintów***, które pozwalają na **bezpośrednie użycie bibliotek** typu

- Bibliotek starszego typu - Legacy (pliki .mod)
- Biblioteki KiCad Pretty z plików lokalnych (z dysku stałego komputera) (foldery z rozszerzeniem .pretty, zawierające pliki .kicad_mod)
- Biblioteki KiCad Pretty z serwerów zdalnych (z repozytorium GitHub programu KiCad, lub z innych repozytoriów GitHub)
- Biblioteki gEDA (foldery zawierające pliki .fp)
- Biblioteki programu EAGLE

---

**Notatka**

- Można zapisywać tylko biblioteki KiCad *.pretty* jako pliki lokalne na dysku komputera (oraz pliki .kicad_mod wewnątrz tych folderów).
- Wszystkie pozostałe formaty są tylko do doczytu.

---

Poniższy rysunek pokazuje okno dialogowe służące do edycji tabeli bibliotek, które można wywołać poleceniem “Tabele bibliotek” z menu “Ustawienia”.

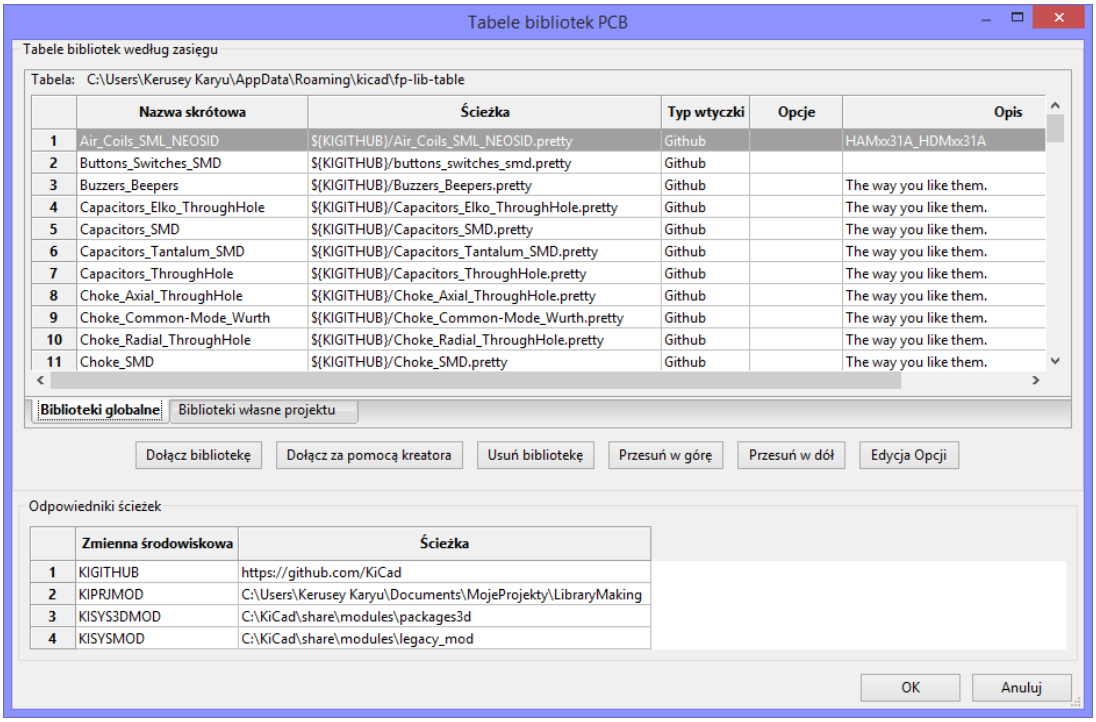



Tabela bibliotek footprintów jest używana do mapowania plików bibliotek obsługiwanych przez program do ich **nazw skrótowych**. Nazwa skrótowa jest używana do wyszukiwania footprintów zamiast poprzedniej metody z wyszukiwaniem plików zgodnie z ustalonym układem ścieżek dostępu.

Pozwala to programowi CvPcb na dostęp do footprintów za pomocą tej samej nazwy w różnych bibliotekach gwarantując tym samym, że właściwy footprint zostanie załadowany z odpowiedniej biblioteki. Pozwala to również na obsługę bibliotek pochodzących z innych programów (z pomocą wtyczek) EDA, takich jak np. Eagle czy gEDA.

### 5.2.1 Globalna tabela bibliotek footprintów

Globalna tabela bibliotek footprintów zawiera listę biblioteki, które są dostępne zawsze, niezależnie od obecnie wczytanego projektu. Tabela ta jest zapisana w pliku `fp-lib-table` w katalogu domowym użytkownika. Jego rzeczywista lokacja zależy użytego systemu operacyjnego.

### 5.2.2 Lokalna tabela bibliotek footprintów zależna od projektu

Lokalna tabela bibliotek footprintów zależna od projektu zawiera listę bibliotek, które są dostępne wyłącznie w obecnie wczytanym projekcie. Lokalna tabela może być modyfikowana tylko wtedy, gdy zostanie ona załadowana razem z listą sieci tego projektu. Gdy projekt nie został załadowany lub gdy taka lokalna tabela nie istnieje, tworzona jest pusta tabela, którą będzie można wypełnić i później zapisać razem z plikiem przypisań footprintów (z rozszerzeniem `.cmp`).

### 5.2.3 Konfiguracja początkowa

Gdy CvPcb lub Pcbnew zostanie uruchomiony i globalna tabela bibliotek `fp-lib-table` nie zostanie znaleziona w katalogu domowym użytkownika, CvPcb lub Pcbnew będzie próbował skopiować domyślną tabelę bibliotek `fp-lib-table` zapisaną w folderze `template` do pliku `fp-lib-table` w katalogu domowym użytkownika.

Jeśli plik `fp-lib-table` nie może zostać odnaleziony, to zamiast operacji kopiowania zostanie utworzona pusta tabela. Gdyby taka sytuacja miała miejsce użytkownik ma też możliwość skopiowania pliku `fp-lib-table` samodzielnie lub “ręczne” skonfigurowania tabeli.

Domyślna tabela bibliotek zawiera wszystkie standardowe biblioteki jakie zostały zainstalowane razem z programem KiCad EDA Suite.

Oczywiście, użytkownicy pragnący dostosować konfigurację bibliotek do własnych potrzeb powinni to zrobić tuż po zainstalowaniu programu KiCad.

(Zbyt duża ilość aktywnych bibliotek będzie skutkować dłuższym czasem wyszukiwania footpritntów)

#### 5.2.4 Dodawanie nowych wpisów w tabeli

By móc używać biblioteki najpierw należy dodać globalną lub lokalną tabelę. Lokalna tabela ma zastosowanie tylko gdy istnieje otwarta lista sieci projektu.

*Każda pozycja tabeli musi posiadać unikalną nazwę skrótową*.

Nie musi ona mieć jakiegokolwiek związku z bieżącą nazwą pliku lub ścieżki do niego. Znak dwukropka `:` nie może być używany w nazwach skrótowych. Każda pozycja musi również odnosić się do prawidłowej ścieżki/nazwy pliku w zależności od typu biblioteki. Ścieżki do plików mogą być bezpośrednie, względne lub pochodzić ze specjalnych zmiennych systemowych - opisanych dalej.

Aby biblioteka została wczytana przez CvPcb musi być także wybrana właściwa wtyczka obsługująca dany format pliku. CvPcb obecnie wspiera następujące formaty plików bibliotek: KiCad Legacy, KiCad Pretty, Eagle oraz gEDA.

Istnieje również pole przeznaczone do wpisania opisu dla danego wpisu w tabeli. Pole z opcjami nie jest w tej chwili używane, zatem umieszczanie jakichkolwiek opcji nie ma znaczenia przy ładowaniu bibliotek.

Proszę zauważyć, że nie można umieścić dwóch takich samych nazw skrótowych w jednej tabeli. Jednakże, można wpisać tą samą nazwę skrótową w globalnej i lokalnej tabeli bibliotek. Tabela lokalna ma większy priorytet niż tabela globalna w takim przypadku. Gdy wpisy zostaną zdefiniowane w lokalnej tabeli bibliotek, to plik `fp-lib-table` zawierający te wpisy zostanie umieszczony w folderze skąd pochodzi lista sieci.

#### 5.2.5 Pobieranie wartości ze zmiennych systemowych

Jednym z największych zalet tabeli bibliotek footprintów jest możliwość używania odnośników do zmiennych systemowych. Pozwala to na zdefiniowanie własnych ścieżek do bibliotek w zmiennych systemowych i używanie ich w projektach. Odnośniki do zmiennych systemowych można wplatać w treść pól zawierających ścieżkę do pliku używając powszechnie znanego formatu `*${nazwa_zmiennej}`.

Domyślnie KiCad podczas pracy definiuje **dwie zmienne systemowe**:

- zmienna **`KIPRJMOD`**. Wskazuje zawsze na katalog główny obecnego projektu i nie może być modyfikowana.
- zmienna **`KISYSMOD`**. Wskazuje na miejsce gdzie zainstalowano domyślne biblioteki programu KiCad.

Można re-definiować samodzielnie zmienną **`KISYSMOD`** za pomocą okna dialogowego wywoływanego przez **Ustawienia** → **Konfiguracja ścieżek dostępu**, co pozwala na zastąpienie standardowych bibliotek ich własnymi odpowiednikami.

Podczas wczytywania listy sieci projektu, CvPcb definiuje zmienną **`KIPRJMOD`** używając do tego ścieżki dostępu do tego pliku (zwykle jest to ścieżka dostępu do projektu).

Pcbnew także definiuje tą zmienną podczas ładowania pliku z obwodem drukowanym.

To pozwala na zapisanie niektórych bibliotek w folderze projektu bez potrzeby definiowania pełnej ścieżki do niego (która nie zawsze jest znana) i umieszczenie ich w tabeli bibliotek znajdującej się również w folderze projektu.

#### 5.2.6 Używanie wtyczki GitHub

GitHub to specjalna wtyczka pozwalająca na łączenie się ze zdalnym repozytorium GitHub zawierającym footprinty w formacie Pretty (Pretty to nazwa nowego formatu zapisu footprintów w programie KiCad). Repozytorium to jest tylko do odczytu, ale wtyczka umożliwia również dostęp do technologii "Copy On Write" (COW) wspierającej możliwość edycji footpritnów odczytanych z repozytorium GitHub https://github.com/KiCad i zapisanie ich nowych wersji na dysku lokalnym, które później można

wysłać do opiekunów repozytorium w celu ich aktualizacji. By dodać wpis GitHub do tabeli bibliotek, pole "Ścieżka" musi zostać wypełniona ważnym adresem URL do repozytorium GitHub.

Przykładowo

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

lub

https://github.com/KiCad

Zwykle poprawna ścieżka URL jest tworzona wg następującego schematu:

https://github.com/nazwa_użytkownika/nazwa_repozytorium

Pole "Typ Wtyczki" musi być ustawione jako "Github". Aby włączyć funkcję "Copy On Write" należy w polu "Opcje" dodać parametr **allow_pretty_writing_to_this_dir** który zawierał będzie ścieżkę na dysku lokalnym gdzie zapisywane będą pliki z modyfikacjami. Jeśli ta opcja zostanie pominięta to biblioteka GitHub jest tylko do odczytu. Footprinty tam zapisane są połączeniem części tylko do odczytu repozytorium GitHub i treści lokalnych zmian by utworzyć zmodyfikowaną bibliotekę footprintów. Każda modyfikacja biblioteki GitHub będzie trafiać do tej lokalnej biblioteki hybrydowej COW umieszczonej w odpowiednim folderze `*.pretty`. Należy w tym miejscu nadmienić, iż część rezydentna COW pochodząca z repozytorium GitHub jest zawsze tylko do odczytu, co oznacza, że nie można niczego samodzielnie usunąć lub zmodyfikować bezpośrednio w samym repozytorium GitHub. Niezależnie czy biblioteka będzie hybrydowa, czyli połączona z lokalnej części tylko do odczytu i zapisu, czy tylko część zdalną przeznaczoną tylko do odczytu, będzie ona dalej zwana biblioteką "Github" w dalszych rozważaniach.

Poniższa tabela pokazuje wpis z tabeli bibliotek, której nie została przypisana opcja **allow_pretty_writing_to_this_dir**:

| **Nazwa skrótowa** | **Ścieżka** | **Typ wtyczki** | **Opcje** | **Opis** |
|---|---|---|---|---|
| github | https://github.com/liftoff-sr/-pretty_footprints | Github | | Liftoff's GH footprints |

Następna tabela pokazuje wpis z tabeli bibliotek z opcją dotyczącą COW. Zmienna `${HOME}` jest tylko przykładowa. Folder `github.pretty` jest umieszczony w folderze do którego prowadzi ścieżka `${HOME}/pretty/`. W każdym przypadku użycia opcji **allow_pretty_writing_to_this_dir**, wymagane jest samodzielne utworzenie tego folderu i musi on posiadać rozszerzenie `*.pretty`.

| **Nazwa skrótowa** | **Ścieżka** | **Typ wtyczki** | **Opcje** | **Opis** |
|---|---|---|---|---|
| github | https://github.com/liftoff-sr/-pretty_footprints | Github | allow_pretty_writing_to_this_dir= ${HOME}/pretty/github.pretty | Liftoff's GH footprints |

Footprinty pobierane z repozytorium mają zawsze pierwszeństwo przed tymi umieszczonymi w folderze na który wskazuje opcja **allow_pretty_writing_to_this_dir**. Po zapisaniu footprintu do lokalnego folderu przechowującego hybrydowe pliki COW, np. poprzez zapisanie zmian w edytorze footprintów, żadne aktualizacje GitHub nie będą widoczne podczas ładowania footprintów o tej samej nazwie, niż te, które zostały zapisane lokalnie.

Zawsze należy korzystać z odrębnego folderu `*.pretty` dla poszczególnych bibliotek GitHub i nigdy nie powinno się łączyć folderów przez przypisywanie tego samego folderu do innych bibliotek GitHub.

Także, nie powiino się używać tego samego folderu COW w całej tabeli. Mogłoby to doprowadzić do bałaganu nad którym nie byłoby można zapanować.

Wartości symboliczne w zmiennych systemowych zapisane w notacji ${nazwa_zmiennej} przypisane do opcji **allow_pretty_writing_to** będą rozwijane automatycznie by utworzyć właściwą ścieżkę, tak samo jak to ma miejsce w polu *Ścieżka*.

Co robić z plikami w COW? System COW to element przyśpieszający współużytkowanie footprintów.

Jeśli zawartość COW będzie regularnie przesyłana do zarządcy repozytorium GitHub, będzie można pomóc w uaktualnianiu kopii znajdujących się w repozytorium zdalnym. Całość jest bardzo prosta. Za pomocą poczty elektronicznej należy wysłać

pliki `*.kicad_mod` znajdujące się w folderach systemu COW do osoby zarządzającej repozytorium. Po otrzymaniu potwierdzenia, że zmiany zostały zaakceptowane i wprowadzone, można skasować wysłane pliki z COW. Nowe wersje plików zostaną pobrane z repozytorium GitHub. Głównym celem jest utrzymywanie jak najmniejszego zestawu plików systemu COW jak tylko jest to możliwe poprzez regularne przesyłanie zawartych w niej plików do repozytorium znajdującego się pod adresem https://github.com.

#### 5.2.7 Generalne zalecenia przy używaniu tabeli bibliotek

Biblioteki footprintów mogą być zdefiniowane globalne lub lokalnie dla obecnie wczytanego projektu. Biblioteki umieszczone w globalnej tabeli bibliotek użytkownika są zawsze dostępne i są zapisane w pliku `fp-lib-table` w katalogu domowym użytkownika.

Globalne biblioteki będą dostępne nawet jeśli nie została otwarta lista sieci danego projektu.

Inaczej sprawa się ma w przypadku lokalnych bibliotek, które są aktywne wyłącznie dla bieżącej listy sieci.

Lokalna tabela bibliotek jest zapisywana w pliku `fp-lib-table` umieszczonym w tej samej ścieżce co lista sieci. Nie ma przeszkód co do definiowania odnośników do bibliotek w obu tabelach.

Są jednak zalety i wady każdego z rozwiązań, które należy rozważyć. Można zdefiniować wszystkie biblioteki w globalnej tabeli bibliotek, co oznacza, że będą one zawsze dostępne gdy będą potrzebne. Wadą takiego rozwiązania będzie szybkość wyszukiwania w nich odpowiedniego footprintu. Można zdefiniować wszystkie biblioteki w lokalnej tabeli bibliotek.

Zaletą takiego rozwiązania będzie możliwość zdefiniowania tylko tych bibliotek, które będą w danej chwili potrzebne oraz skrócenie czasu ich przeszukiwania.

Wadą tego rozwiązania będzie zaś to, że będzie trzeba zawsze pamiętać, by dodać odpowiednie biblioteki dla każdego nowego projektu. Można zdefiniować biblioteki w obu tabelach jednocześnie.

Sensowne staje się wtedy wpisanie bibliotek, które są wykorzystywane prawie we wszystkich projektach do tabeli globalnej, a w lokalnych tabelach umieszczać tylko te, które są przydatne tylko w tym konkretnym projekcie. Będzie to rozwiązanie, które będzie posiadało największą elastyczność kosztem zmniejszenia szybkości wyszukiwania.

## 5.3 Użycie Kreatora Tabeli Bibliotek Footprintów

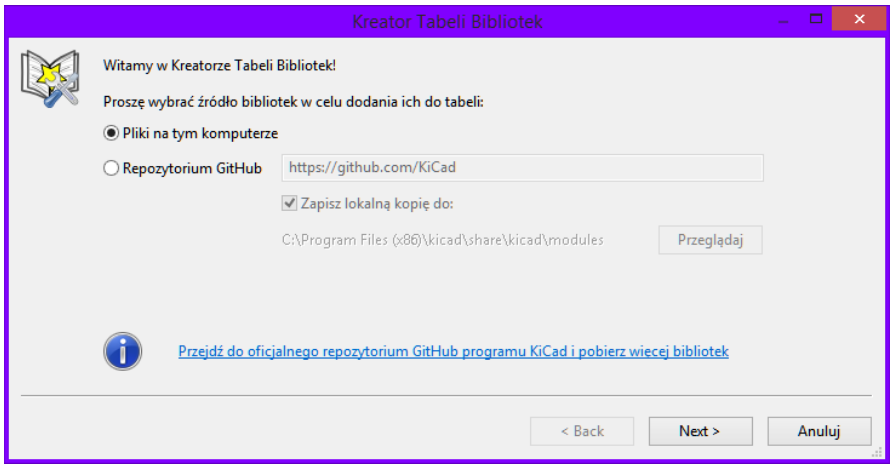
Kreator dodawania bibliotek footprintów do tabeli bibliotek jest dostępny z *okna dialogowego służącego do edycji tabel bibliotek*.

Biblioteki mogą być dowolnego typu, który jest wspierany przez program KiCad.

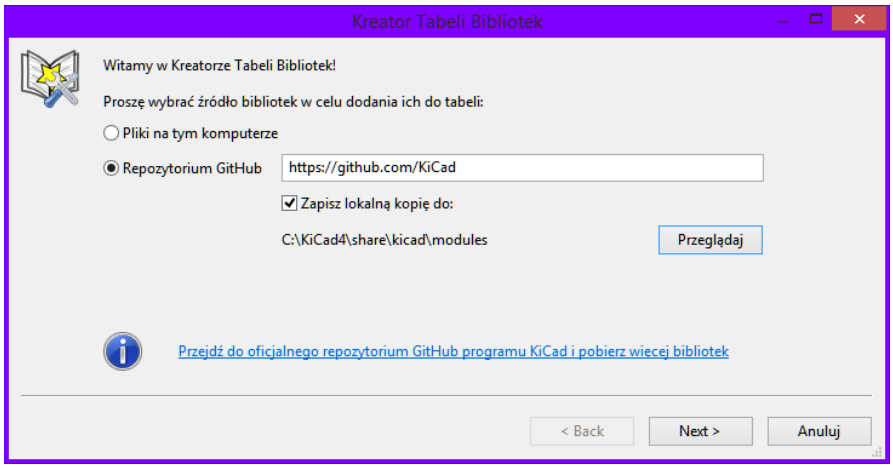
Może on dodawać “lokalne” biblioteki lub biblioteki zdalne z repozytoriów GitHub.

Gdy biblioteki znajdują się w repozytorium GitHub, mogą one zostać dodane jako zdalne biblioteki, bądź **pobrane oraz dodane już jako *lokalne bibliotelki***.

Tutaj zostały wybrane biblioteki lokalne.

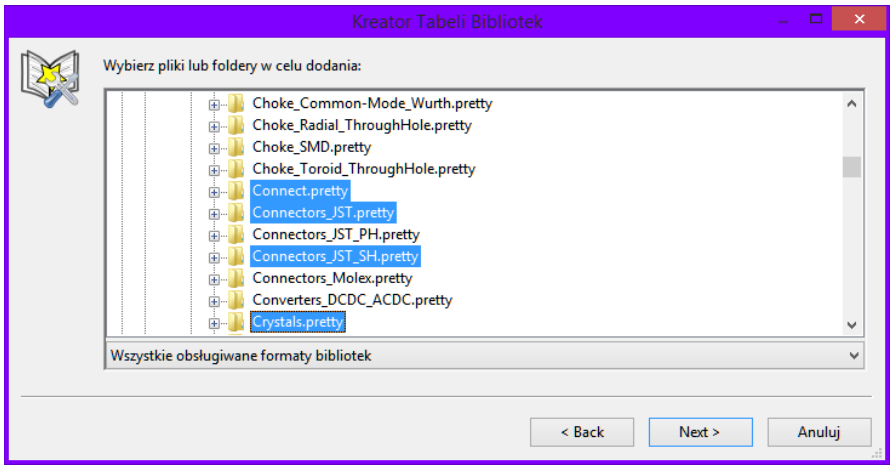



Tutaj zostały wybrane biblioteki zdalne.

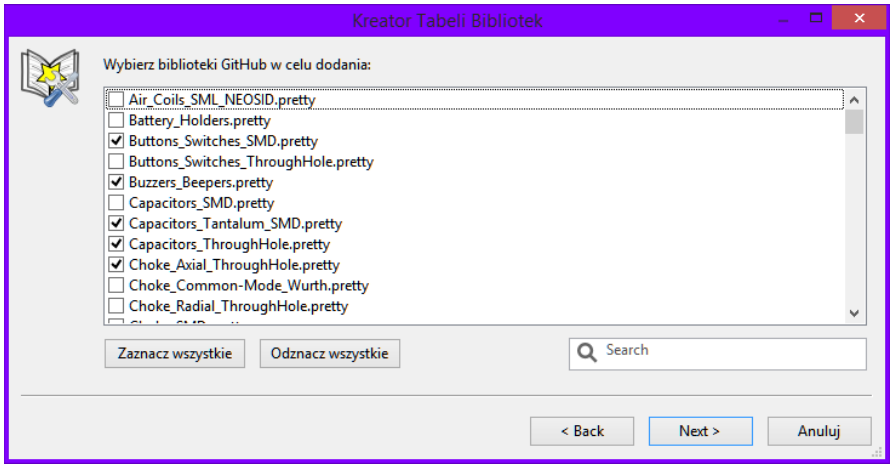



W zależności od wybranej opcji, jedna z tych stron zostanie wyświetlona, gdzie należy wybrać listę bibliotek do dodania:

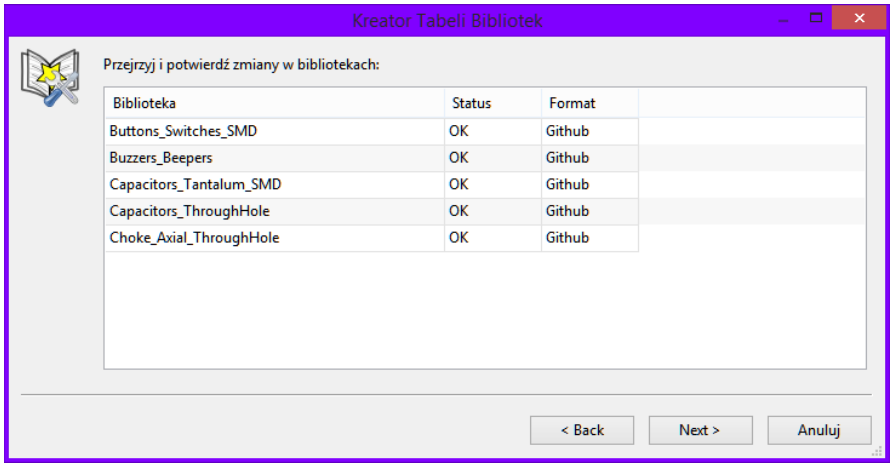
Tutaj zostały wybrane biblioteki lokalne.




Tutaj zostały wybrane biblioteki zdalne.




Po wybraniu zestawu bibliotek, następna strona dokona ich walidacji:

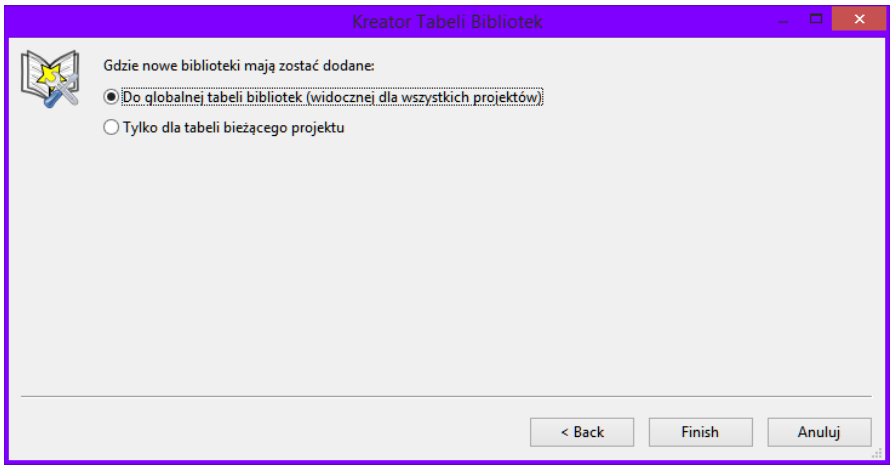



Jeśli któreś z bibliotek byłyby niepoprawne (nie obsługiwane, nie będące bibliotekami footprintów ...) będą one oznaczone statusem “BŁĘDNA”.

Ostatnią opcją do wyboru jest wybór tabeli do jakiej miałyby trafić wybrane biblioteki:

- Tabela globalna

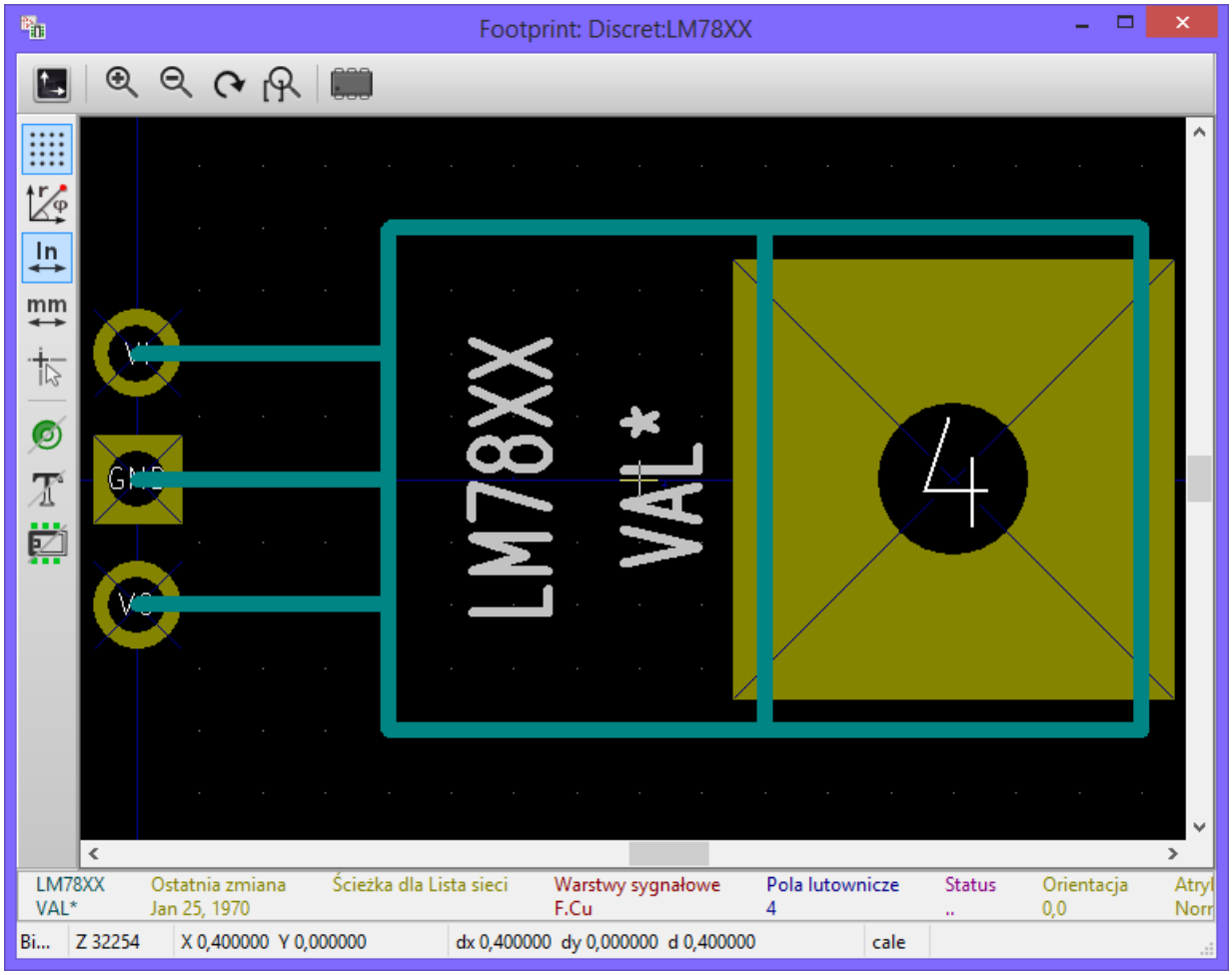
- Tabela lokalna (zależna od projektu)



# 6 Podgląd bieżącego footprintu

## 6.1 Polecenie Podgląd footprintu

Polecenie “Podgląd footprintu” pozwala na wyświetlenie bieżącego footprintu, czyli tego który aktualnie jest wskazany na liście w panelu footprintów. Można przeglądać w ten sposób listę footprintów klikając na ich nazwy przy pozostawionym oknie podglądu footprintów. Można również podglądać widok 3D (jeśli moduły maja przypisane kształty 3D).




### 6.1.1 Informacje na pasku statusu

Pasek statusu jest umieszczony na dole głónego okna CvPcb i dostarcza użytecznych informacji dla użytkownika. Poniższa tabela opisuje zawartość każdego panelu na pasku statusu.

| | |
|---|---|
| Lewy | Liczba komponentów: w sumie, nieprzypisanych |
| Środkowy | Wartość przypisanego filtra wybranego komponentu |
| Prawy | Tryb filtracji oraz ilość dostępnych footprintów |

### 6.1.2 Skróty klawiaturowe

| | |
|---|---|
| F1 | Przybliżanie |
| F2 | Oddalanie |
| F3 | Przerysowanie widoku |

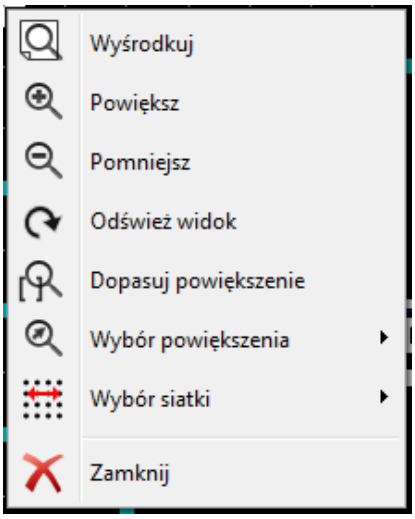
| | |
|---|---|
| F4 | Przesunięcie kursora do środka okna |
| Home | Dopasowanie powiększenia by pokazać cały footprint w oknie |
| Spacja | Ustawienie punktu bazowego dla współrzędnych relatywnych w miejscu kursora |
| Strzałka w prawo | Przesunięcie kursora o jedną pozycję siatki w prawo |
| Strzałka w lewo | Przesunięcie kursora o jedną pozycję siatki w lewo |
| Strzałka w górę | Przesunięcie kursora o jedną pozycję siatki w górę |
| Strzałka w dół | Przesunięcie kursora o jedną pozycję siatki w dół |

#### 6.1.3 Polecenia związane z myszą

| | |
|---|---|
| Kółko myszy | Przybliżanie lub oddalanie widoku wokół bieżącej pozycji kursora |
| Ctrl + Kółko myszy | Przesuwanie widoku w prawo lub w lewo |
| Shift + Kółko myszy | Przesuwanie widoku w dół lub w górę |
| Kliknięcie prawym przyciskiem | Otwarcie menu kontekstowego |

#### 6.1.4 Menu kontekstowe

Wyświetlane jest poprzez kliknięcie prawym klawiszem myszy:




| | |
|---|---|
| Wybór powiększenia | Bezpośredni wybór powiększenia obrazu. |
| Wybór siatki | Bezpośredni wybór wyświetlanej siatki. |

#### 6.1.5 Górny pasek narzędziowy

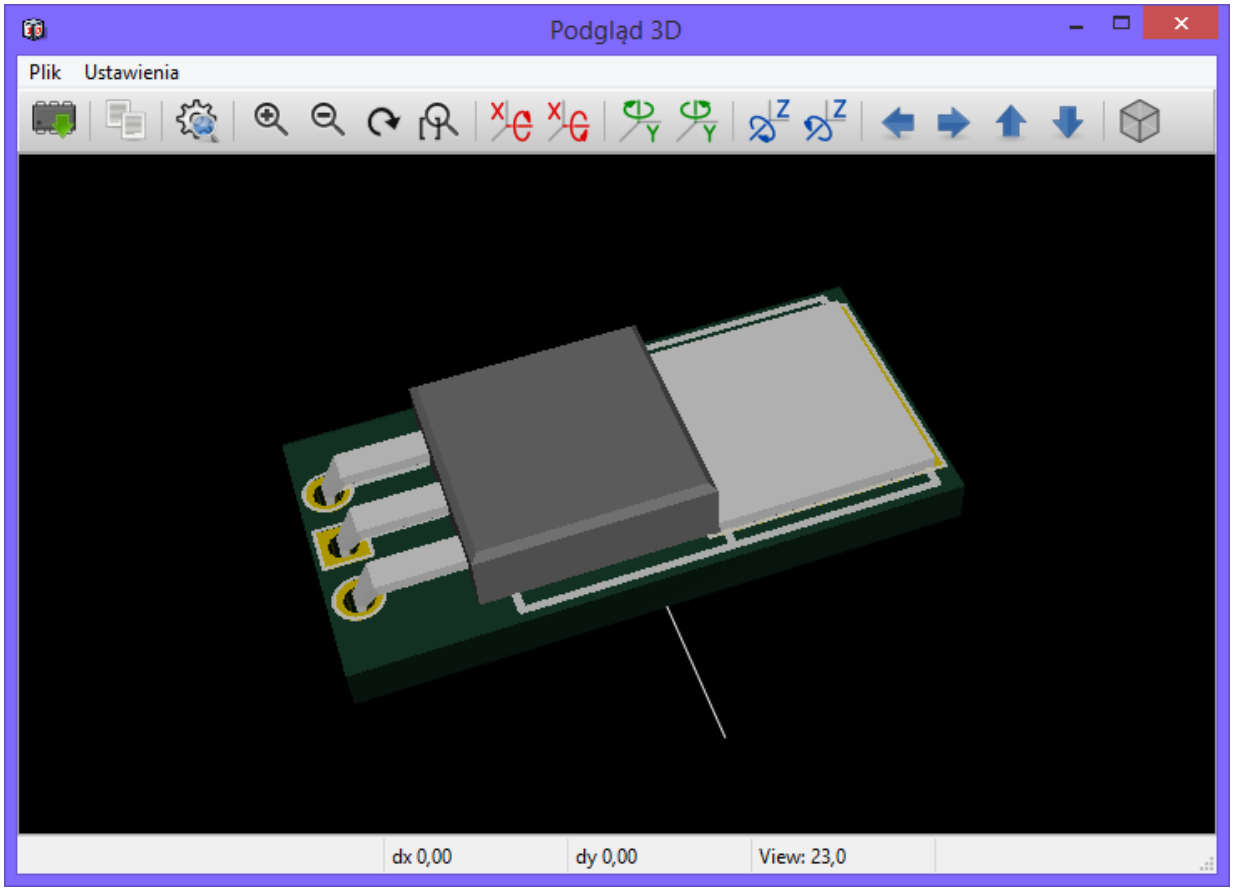
| | |
|---|---|
|  | Pokazuje okno z ustawieniami wyświetlania |
|  | Przybliżanie |
|  | Oddalanie |

| | |
|---|---|
|  | Przerysowanie widoku |
|  | Ustawienie powiększenia by całość zmieściła się na ekranie |
|  | Otwiera przeglądarkę modeli 3D |

### 6.1.6 Lewy pasek narzędziowy

| | |
|---|---|
|  | Pokazuje lub ukrywa siatkę |
|   | Pokazuje koordynaty jako polarne lub kartezjańskie |
|   | Pokazuje koordynaty w calach |
|   | Pokazuje koordynaty w milimetrach |
|  | Przełącza styl kursora |
|  | Przełącza pomiędzy wyświetlaniem pól lutowniczych w trybie pełnym lub tylko zarys |
|  | Przełącza pomiędzy wyświetlaniem tekstu w trybie pełnym lub tylko zarys |
|  | Przełącza pomiędzy wyświetlaniem krawędzi w trybie pełnym lub tylko zarys |

## 6.2 Podgląd 3D




### 6.2.1 Polecenia związane z myszą

| | |
|---|---|
| Kółko myszy | Przybliżanie i oddalanie widoku w bieżącej lokalizacji kursora |
| Ctrl + Kółko myszy | Przesuwanie widoku w prawo lub lewo |
| Shift + Kółko myszy | Przesuwanie widoku w górę lub prawo |

### 6.2.2 Górny pasek narzędziowy

| | |
|---|---|
|  | Przeładowywuje model 3D |
|  | Kopiuje obraz 3D do schowka |
|  | Ustawia opcje przeglądarki 3D |
|  | Przybliżenie |
|  | Oddalanie |
|  | Przerysowanie widoku |

| | |
|---|---|
|  | Ustawienie powiększenia by całość zmieściła się na ekranie |
|  | Obraca w lewo wokół osi X |
|  | Obraca w prawo wokół osi X |
|  | Obraca w lewo wokół osi Y |
|  | Obraca w prawo wokół osi Y |
|  | Obraca w lewo wokół osi Z |
|  | Obraca w prawo wokół osi Z |
|  | Przesuwa widok w lewo |
|  | Przesuwa widok w prawo |
|  | Przesuwa widok w górę |
|  | Przesuwa widok w dół |
|  | Włącza lub wyłącza tryb projekcji ortagonalnej. |

# 7 Użycie CvPcb do przypisywania footprintów komponentom

## 7.1 Ręczne przypisywanie footprintów do komponentów

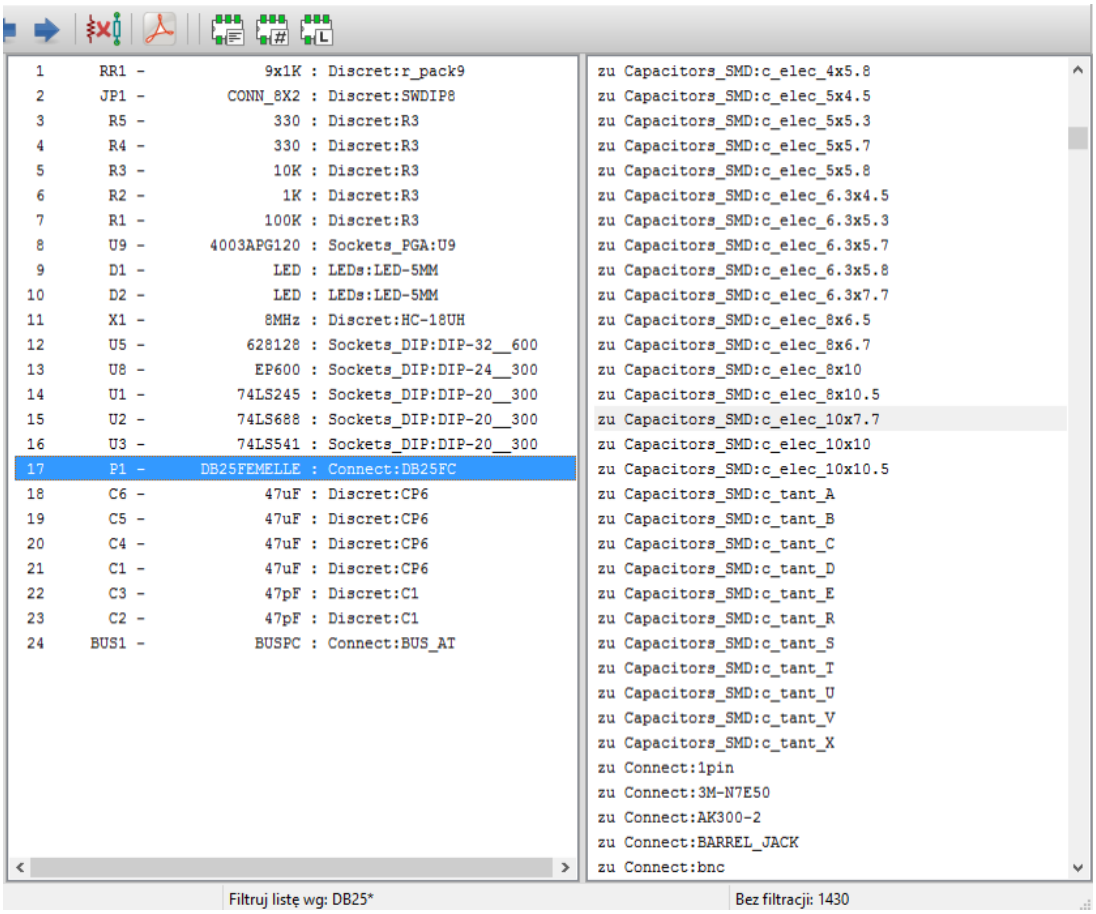
By manualnie przypisać footprint do komponentu należy najpierw wybrać komponent z panelu komponentów. Następnie wybrać footprint w panelu footprintów i kliknąć dwukrotnie na nazwie wybranego footprintu. Następny komponent nieposiadający przypsania zostanie wybrany automatycznie. Zmiana przypisania jest wykonywana w ten sam sposób.

## 7.2 Filtrowanie listy footprintów

Jeśli wybrany komponent lub biblioteka jest podświetlona gdy jedna lub więcej opcji filtrowania jest włączona, lista footprintów w programie CvPcb jest odpowiednio filtrowana.

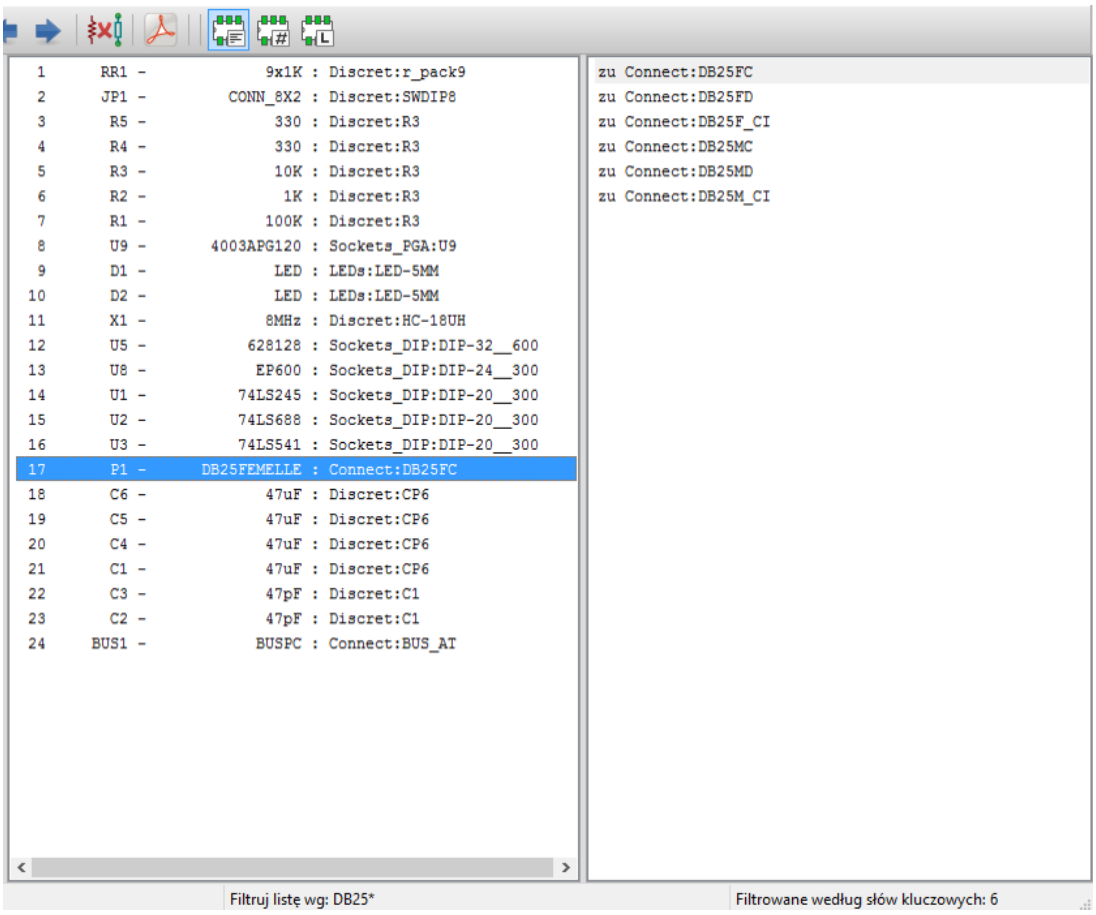
Ikony  włączają lub wyłączaja poszczególne filtry. Gdy filtracja nie jest włączona, wyświetlana jest pełna lista footprintów.

Bez filtrowania:

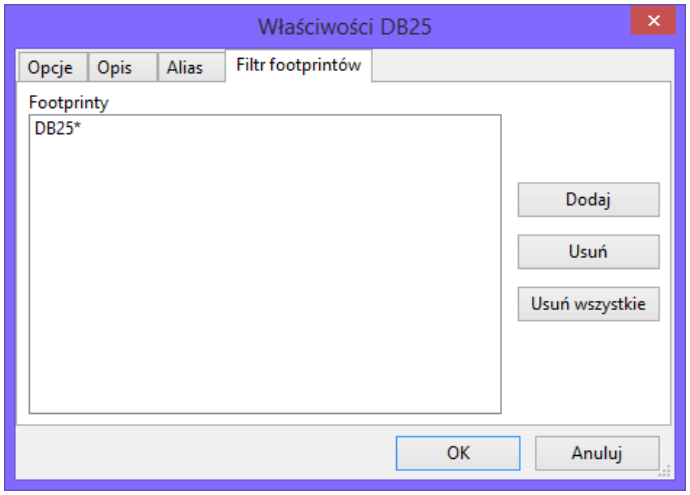



Filtrowanie za pomocą listy filtrów przypisanej do wybranego komponentu. Filtry te są wyświetlane na centralnym panelu pasku statusu na dole głównego okna programu.

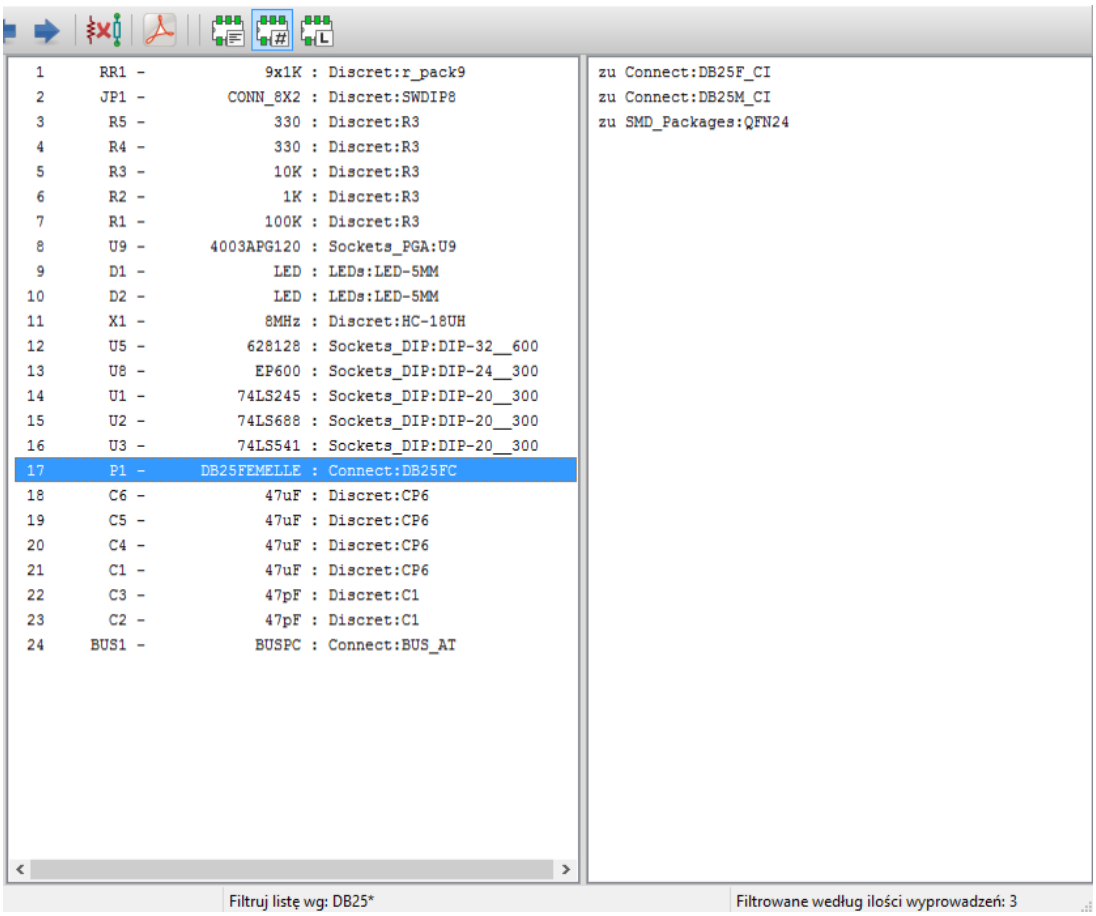
Odfiltrowane poprzez listę dozwolonych footprintów jaką posiada wybrany komponent:



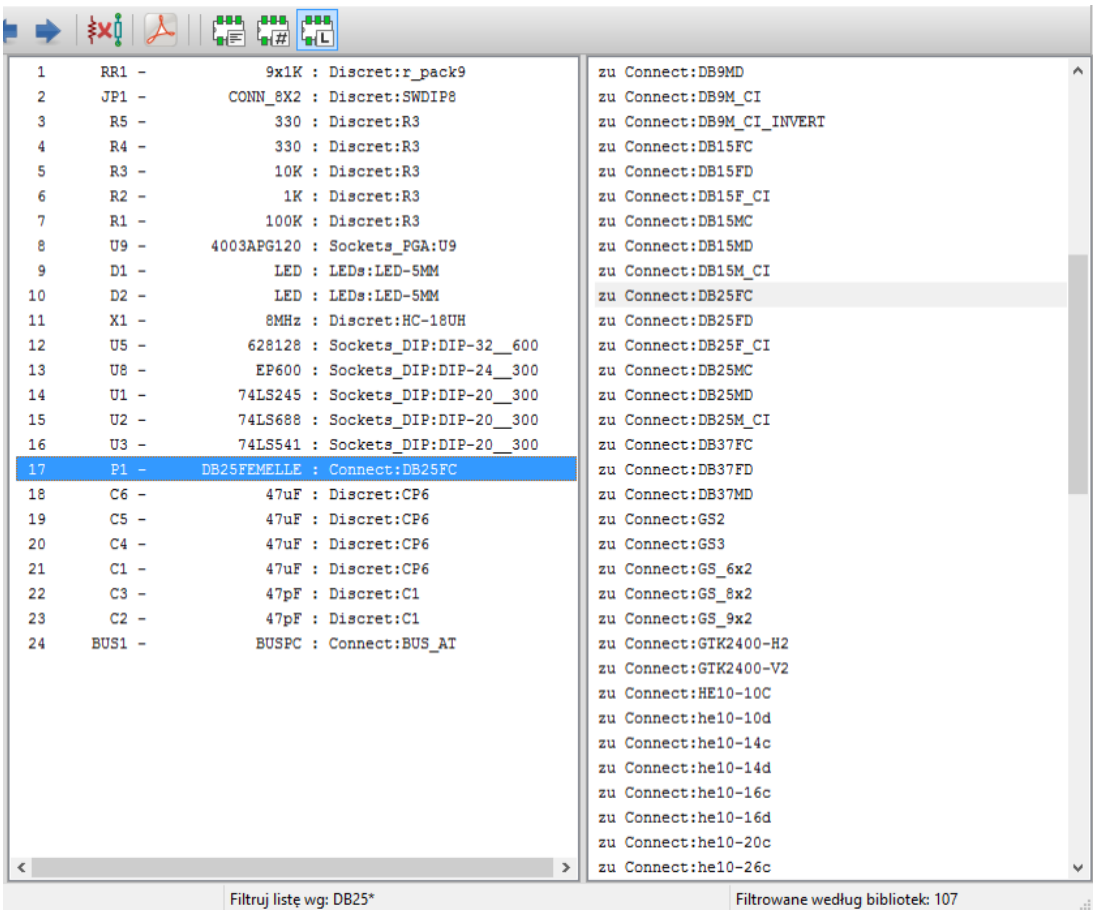

W edytorze bibliotek komponentó w programie Eeschema, lista footprintów została ustawiona za pomocą wpisów w zakładce filtrowania footprintów w oknie dialogowym właściwości komponentów, jak pokazano niżej.




Odfiltrowane poprzez liczbę wyprowadzeń jaką posiada wybrany komponent:

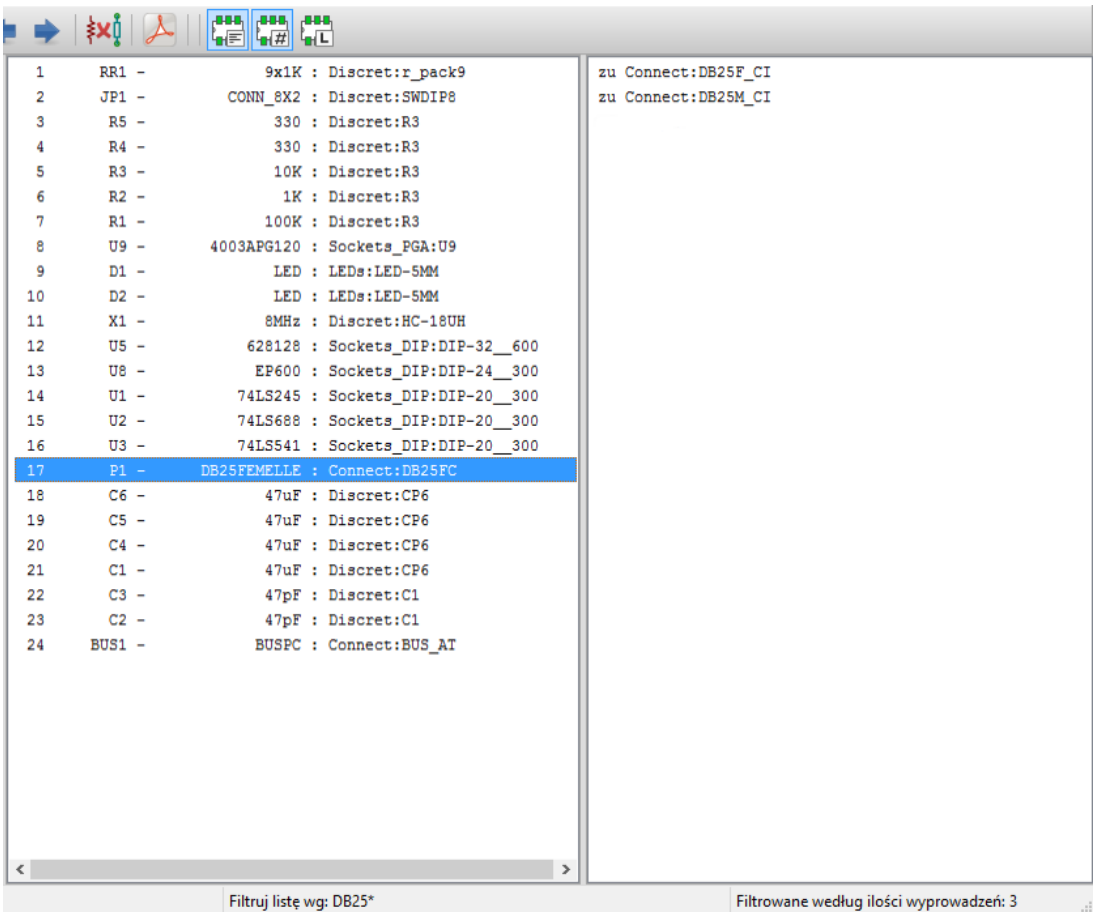



Odfiltrowane poprzez aktualnie wybraną bibliotekę.




Kolejne klucze filtrowania mogą być włączane niezależnie by tworzyć zaawansowane filtry pozwalające zredukować ilość footprintów do wyboru w panelu footprintów.

Odfiltrowane poprzez liczbę wyprowadzeń oraz dozwolony komponent:


1 RR1 - 9x1K : Discret:r_pack9
2 JP1 - CONN_8X2 : Discret:SWDIP8
3 R5 - 330 : Discret:R3
4 R4 - 330 : Discret:R3
5 R3 - 10K : Discret:R3
6 R2 - 1K : Discret:R3
7 R1 - 100K : Discret:R3
8 U9 - 4003APG120 : Sockets_PGA:U9
9 D1 - LED : LEDs:LED-5MM
10 D2 - LED : LEDs:LED-5MM
11 X1 - 8MHz : Discret:HC-18UH
12 U5 - 628128 : Sockets_DIP:DIP-32__600
13 U8 - EP600 : Sockets_DIP:DIP-24__300
14 U1 - 74LS245 : Sockets_DIP:DIP-20__300
15 U2 - 74LS688 : Sockets_DIP:DIP-20__300
16 U3 - 74LS541 : Sockets_DIP:DIP-20__300
17 P1 - DB25FEMELLE : Connect:DB25FC
18 C6 - 47uF : Discret:CP6
19 C5 - 47uF : Discret:CP6
20 C4 - 47uF : Discret:CP6
21 C1 - 47uF : Discret:CP6
22 C3 - 47pF : Discret:C1
23 C2 - 47pF : Discret:C1
24 BUS1 - BUSPC : Connect:BUS_AT
zu Connect:DB25F_CI
zu Connect:DB25M_CI
Filtruj listę wg: DB25*
Filtrowane według ilości wyprowadzeń: 3

# 8 Przypisywanie automatyczne

## 8.1 Pliki przypisań

Pliki skryptów przypisań pozwalają na automatyczne przypisanie footprintów komponentom.

Pobierane są z nich nazwy footprintów na podstawie wartości komponentów (z pól *Wartość*) poszczególnych komponentów. Pliki te mają standardowe rozszerzenie **.equ**.

Są one zapisane jako zywkły tekst, zatem jest możliwa ich edycja w każdym zwykłym edytorze tekstu.

## 8.2 Format plików

Pliki **.equ** zawierają zwykły tekst, gdzie każda linia odpowiada jednemu komponentowi. Każda z linii posiada strukturę:

**'wartość' 'nazwa footprintu'**

Każda nazwa musi być zamknięta w apostrofach, obie nazwy muszą być rozdzielone przynajmniej jednym znakiem spacji.

*Przykład:*

Jeśli komponent U3 to układ scalony 14011 i jego footprint to 14DIP300, linia powinna wyglądać następująco:

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

Dowolna linia rozpoczynająca się od **#** jest traktowana jako komentarz.

Poniżej znajduje się przykład pliku przypisań:

```
#integrated circuits (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'

#regulators
'LP2985LV' 'SOT23-5'
```

## 8.3 Automatyczne przypisywnie footprintów do komponentów

Proces automatycznego przypisywania jest uruchamiany przez wybranie ikony automatycznego przypisywania na górnym pasku narzędzi.

*Wszystkie komponenty jakie zostały znalezione (na podstawie ich wartości) w pliku* `.equ` *będą miały przypisany automatycznie wskazany tam footprint.*