KiCad ことはじめ

October 9, 2018

Contents

1 KiCad のご紹介 **1**

1.1 KiCad のダウンロードとインストール 2

1.1.1 GNU/Linux の場合 2

1.1.2 Apple macOS の場合 2

1.1.3 Windows の場合 3

1.2 サポート 3

2 KiCad 作業の流れ **4**

2.1 概要 4

2.2 フォワード／バック・アノテーション 6

3 KiCad の使用 **7**

3.1 ショートカット・キー 7

3.1.1 アクセラレーター・キー 7

3.1.2 ホットキー 7

3.1.3 例 8

4 電子回路図の描画 **9**

4.1 Eeschema の使用 9

4.2 KiCad でのバス接続 23

5 プリント基板のレイアウト **25**

5.1 Pcbnew の使用 25

5.2 ガーバーファイルの生成 33

5.3 GerbView の使用 34

5.4 Freerouter による自動配線 34

6 KiCad のフォワード・アノテーション 36

7 Make schematic symbols in KiCad 38

7.1 コンポーネント・ライブラリ・エディタの使用 38

7.2 コンポーネントのエクスポート、インポート、変更 41

7.3 quicklib による回路図コンポーネントの作成 41

7.4 大量ピンの回路図コンポーネントの作成 42

8 フットプリントの作成 45

8.1 フットプリント・エディタの使用 45

9 KiCad プロジェクトファイルの可搬性について 47

10 KiCad ドキュメントの詳細 49

10.1 Web 上の KiCad ドキュメント 49

洗練された電子プリント回路基板の開発を成功に導く、*KiCad* をマスターするための必須で簡潔なガイド

著作権

This document is Copyright © 2010-2018 by its 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.

このガイドの中のすべての商標は、正当な所有者に帰属します。

* 貢献者 *

David Jahshan, Phil Hutchinson, Fabrizio Tappero, Christina Jarron, Melroy van den Berg.

翻訳

je6bmq <je6bmq AT gmail.com>, 2018. starfort <starfort AT nifty.com>, 2017. kinichiro <kinichiro.inoguchi AT gmail.com>, 2015. silvermoon <silvermoon AT kicad.jp>, 2011-2015. yoneken <yoneken AT kicad.jp>, 2011-2015.

フィードバック

バグ報告や提案はこちらへお知らせください:

- KiCad のドキュメントについて: https://github.com/KiCad/kicad-doc/issues
- KiCad ソフトウェアについて: https://bugs.launchpad.net/kicad
- KiCad ソフトウェアの国際化 (i18n) について: https://github.com/KiCad/kicad-i18n/issues

発行日

2015 年 5 月 16 日

Chapter 1

KiCad のご紹介

KiCad は、電子回路図と PCB レイアウトを作成するためのオープンソースのソフトウェア・ツールです。その一つの外観の下に、KiCad は、以下の独立したソフトウェア・ツール群のすばらしい組み合わせを実現しています:

プログラム名	説明	拡張子
KiCad	プロジェクト・マネージャ	`*.pro`
Eeschema	回路図エディタ (回路図とコンポーネント)	`*.sch, *.lib, *.net`
Pcbnew	PCB レイアウト	`*.kicad_pcb`
GerbView	ガーバー・ビューア	全ての通常のガーバー
Bitmap2Component	ビットマップをコンポーネントやフットプリントに変換	`*.lib, *.kicad_mod, *.kicad_wks`
PCB Calculator	コンポーネント、線幅、電気的安全間隔、カラーコード等のための計算機	なし
Pl Editor	図枠エディタ	`*.kicad_wks`

注意
拡張子の一覧は完全ではなく、KiCad が連携するファイルの一部のみを含んでおり、どのファイルがどの KiCad アプリケーションで使われるのかの基本的な理解に役立ちます。

KiCad は、複雑な電子基板の開発やメンテナンスに十分使えるほどに成熟していると考えることができます。

KiCad には基板サイズの制限がなく、最大 32 の導体レイヤ、最大 14 のテクニカルレイヤそして最大 4 の補助レイヤを容易に扱えます。KiCad は、ガーバー・ファイル、ドリル・ファイル、コンポーネント・ロケーション・ファイル等のプリント基板作成に必要な全てのファイルを作ることができます。

オープンソース (GPL ライセンスに基づく) であるため、KiCad は、オープンソース志向の電子機器作成プロジェクトに理想的なツールです。

インターネット上の KiCad のホームページ:

http://www.kicad-pcb.org/

1.1 KiCad のダウンロードとインストール

KiCad は GNU/Linux、Apple macOS 、Windows で動作します。以下の URL で最新のインストール手順とダウンロードのリンクが見つかります:

http://www.kicad-pcb.org/download/

重要項目

KiCad 安定版は KiCad Stable Release Policy に基づいて定期的にリリースされます。新機能は継続的に開発ブランチに追加されています。もしあなたが新機能を活用したりテストをするなら、あなたのプラットホームの最新の夜間ビルドをダウンロードしてください。夜間ビルドにはファイルの破損や不正なガーバーの生成のようなバグがあるかもしれませんが、KiCad 開発チームは新機能の開発中も開発ブランチを可能な限り使用可能にしておくことを目標にしています。

1.1.1 GNU/Linux の場合

KiCad の安定版はほとんどのディストリビューション・パッケージ・マネージャで kicad と kicad-doc として見つけることができます。もしあなたがお使いのディストリビューションが最新の安定版を提供してなければ、不安定版の指示に従って、最新の安定版を選択してインストールしてください。

Ubuntu の場合、KiCad の不安定版の夜間ビルドをインストールする最も簡単な方法は *PPA* と *Aptitude* によるものです。端末で次のようにタイプします:

```
sudo add-apt-repository ppa:js-reynaud/ppa-kicad
sudo aptitude update && sudo aptitude safe-upgrade
sudo aptitude install kicad kicad-doc-en
```

Fedora の場合、KiCad の不安定版の夜間ビルドをインストールする最も簡単な方法は *copr* によるものです。KiCad を copr でインストールするには端末で次のようにタイプします:

```
sudo dnf copr enable mangelajo/kicad
sudo dnf install kicad
```

あるいは、コンパイル済みバージョンの KiCad をダウンロードしてインストールしたり、直接 KiCad のソースコードをダウンロードしてコンパイルしてインストールすることもできます。

1.1.2 Apple macOS の場合

macOS 用の安定版の KiCad は以下の URL で見つけられます: http://downloads.kicad-pcb.org/osx/stable/

不安定版の夜間ビルドは以下の URL で見つけられます:
http://downloads.kicad-pcb.org/osx/

1.1.3 Windows の場合

Windows 用の安定版の KiCad は以下の URL で見つけられます:
http://downloads.kicad-pcb.org/windows/stable/

Windows 用の夜間ビルドは以下の URL で見つけられます:
http://downloads.kicad-pcb.org/windows/

1.2 サポート

何か思いついたり、発言したいことがあったり、質問があったり、ヘルプが必要だったり…いずれの場合でも:

- Visit the forum
- 参加してみましょう。#kicad IRC channel on Freenode
- Watch tutorials

Chapter 2

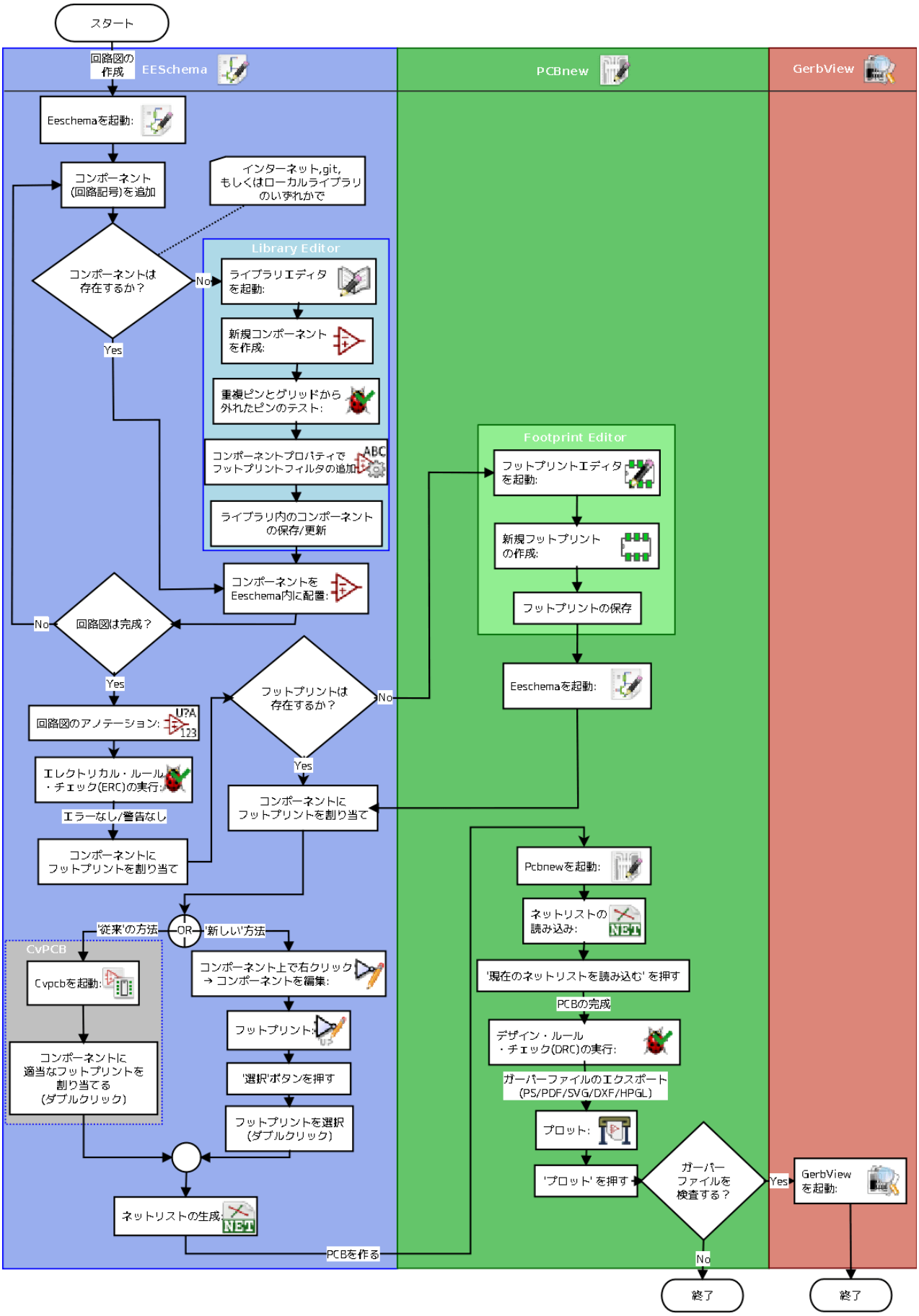
KiCad 作業の流れ

他の PCB 設計ツールとの類似性がある一方で、KiCad は、回路図のコンポーネントとフットプリントを別に持つ独自のワーク・フローを特徴としています。回路図を作成した後の時点でのみ、コンポーネントには実際のフットプリントが割り当てられます。

2.1 概要

KiCad での作業の流れは２つの主要なタスクで構成されています: 回路図の描画と基板のレイアウトです。コンポーネント・ライブラリとフットプリント・ライブラリの両方がこれら 2 つのタスクのために必要です。KiCad には多くをコンポーネントとフットプリントが含まれており、新しく作るためのツールも持っています。

下図は KiCad の作業の流れを示すフローチャートです。このフローチャートはあなたが取る必要のあるステップを、どんな順序で行うのかを説明しています。適用できる場合には便宜上アイコンを付け加えています。


スタート
回路図の作成
EESchema
PCBnew
GerbView
Eeschemaを起動:
コンポーネント(回路記号)を追加
インターネット,git,もしくはローカルライブラリのいずれかで
コンポーネントは存在するか？
No
Yes
Library Editor
ライブラリエディタを起動:
新規コンポーネントを作成:
重複ピンとグリッドから外れたピンのテスト:
コンポーネントプロパティでフットプリントフィルタの追加
ライブラリ内のコンポーネントの保存/更新
コンポーネントをEeschema内に配置:
回路図は完成？
No
Yes
回路図のアノテーション:
エレクトリカル・ルール・チェック(ERC)の実行:
エラーなし/警告なし
コンポーネントにフットプリントを割り当て
フットプリントは存在するか？
No
Yes
コンポーネントにフットプリントを割り当て
'従来'の方法
OR
'新しい'方法
CvPCB
Cvpcbを起動:
コンポーネントに適当なフットプリントを割り当てる(ダブルクリック)
コンポーネント上で右クリック → コンポーネントを編集:
フットプリント:
'選択'ボタンを押す
フットプリントを選択(ダブルクリック)
ネットリストの生成:
PCBを作る
Footprint Editor
フットプリントエディタを起動:
新規フットプリントの作成:
フットプリントの保存
Eeschemaを起動:
Pcbnewを起動:
ネットリストの読み込み:
'現在のネットリストを読み込む' を押す
PCBの完成
デザイン・ルール・チェック(DRC)の実行:
ガーバーファイルのエクスポート(PS/PDF/SVG/DXF/HPGL)
プロット:
'プロット' を押す
ガーバーファイルを検査する？
Yes
No
GerbView を起動:
終了
終了

For more information about creating a component, read Making schematic symbols. And for information about how to create a new footprint, see Making component footprints.

Quicklib は、素早く KiCad のコンポーネントを作成することのできる Web ベースのインターフェイスを持ったツールです。Quicklib についての更なる情報は、「Quicklib による回路図コンポーネントの作成」を参照して下さい。

2.2 フォワード／バック・アノテーション

電子回路図を完全に描き終わったら、次のステップはそれを PCB に移すことです。しばしば、追加のコンポーネントを付け足す必要があったり、部品が違うサイズへと変更されたり、ネット名が変更されたりといったことが起こります。これは 2 つの方法で行うことができます: フォワード・アノテーションとバック・アノテーションです。

フォワード・アノテーションとは回路図の情報を対応する PCB レイアウトに送る機能です。少なくとも最初に一度は回路図から PCB へインポートしなければならないので、これは基本的な機能です。その後、フォワード・アノテーションは PCB へ回路変更を順次伝えることを可能にします。フォワード・アノテーションについての詳細は、「KiCad のフォワード・アノテーション」の章で議論されています。

バック・アノテーションは、PCB レイアウトの変更をその対応する回路図に送り戻す処理です。バック・アノテーションを招く２つの似たような原因はゲートの入替とピンの入替です。このような場合、ゲートやピンは機能的には同一ですが、レイアウトの都合によってのみ制限を受けるのでピンやゲートは実際にはレイアウト中に決定されます。選択が PCB でなされたなら、この変更は回路図へと差し戻されます。

Chapter 3

KiCad の使用

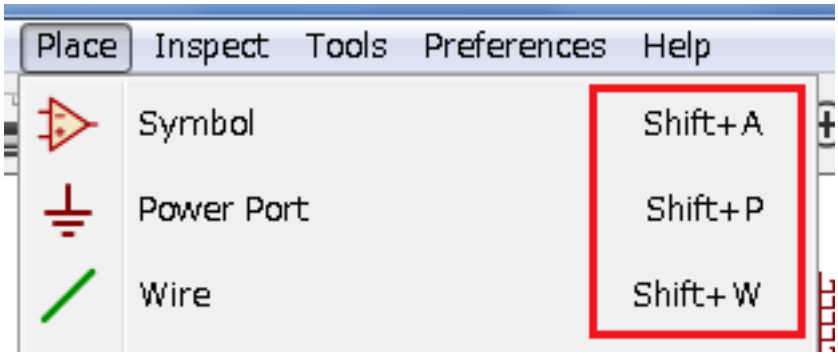
3.1 ショートカット・キー

KiCad は関連はあっても異なる２種類のショートカット・キーを持っています: アクセラレーター・キーとホットキーです。マウスを使う代わりにキーボードをコマンド変更に使うことで、どちらも KiCad での作業迅速化に使われます。

3.1.1 アクセラレーター・キー

アクセラレーター・キーは、メニューやツールバー・アイコンのクリックと同様の効果を持ちます: コマンドは入力されますが、左マウス・ボタンがクリックされるまでは何も起きません。コマンド・モードで入力はしたいけど直ちに実行はしたくない時にアクセラレーター・キーを使用します。

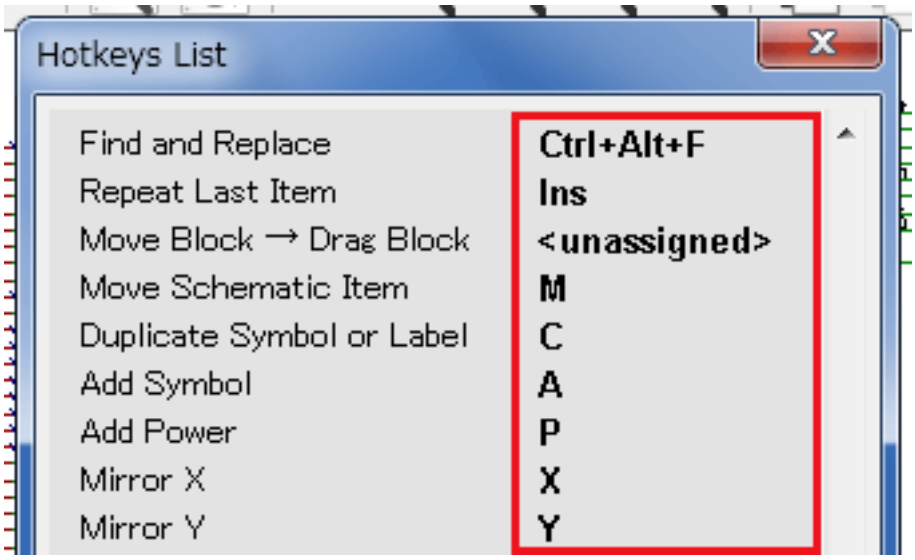
アクセラレーター・キーは



3.1.2 ホットキー

ホットキーはアクセラレーター・キーに左マウス・クリックを足したものと同じです。ホットキーを使用すると現在のカーソル位置で直ちにコマンドを開始します。ワークフローを中断することなく直ちにコマンドを変更するためにはホットキーを使用します。

To view hotkeys within any KiCad tool go to ***Help*** → ***List Hotkeys*** or press Ctrl+F1:



You can edit the assignment of hotkeys, and import or export them, from the ***Preferences → Hotkeys Options*** menu.

注意
このドキュメントでは、ホットキーをこのような括弧で表現しています: [a] もし [a] を見たなら、キーボードの"a" キーを押して下さい。

3.1.3 例

回路図で線を引く簡単な例を考えてみましょう。

アクセラレータ・キーを使う場合、" ワイヤを配置" コマンドを呼び出すよう"Shift + W" を押します (カーソルの形が変わります)。続いて、配線を開始するためには、線を引き始めたい場所でマウスを左クリックします。

ホットキーでは、単に [w] を押すと現在のカーソル位置から直ちに配線が始まります。

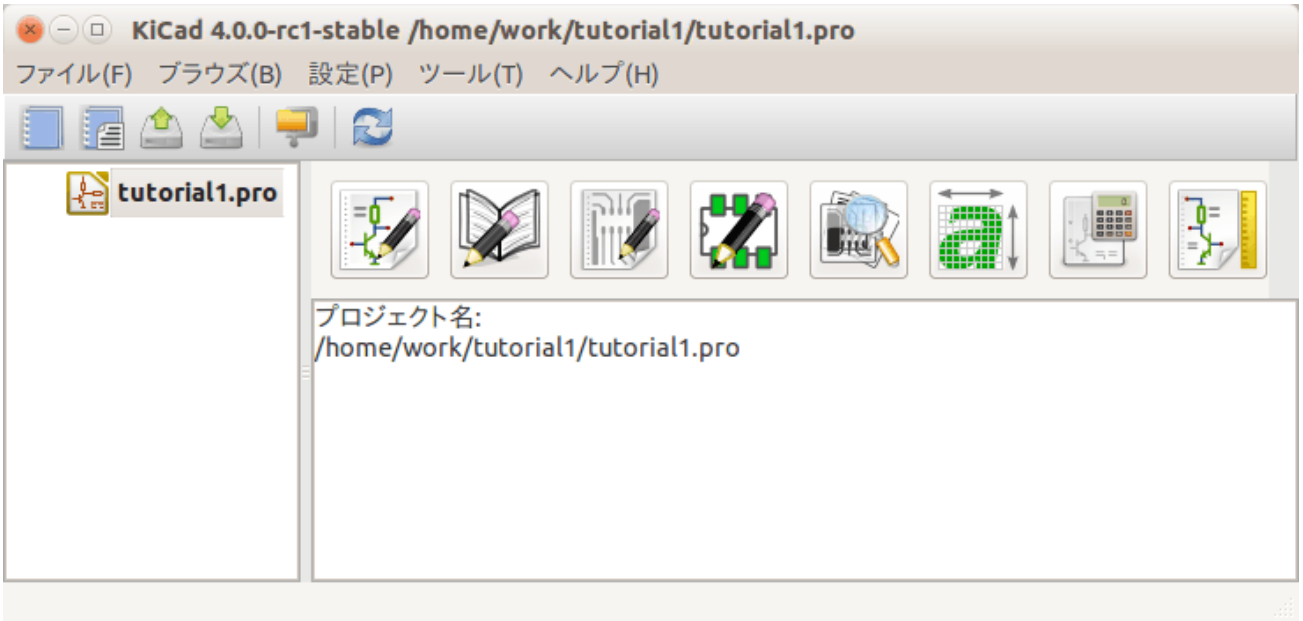
Chapter 4

電子回路図の描画

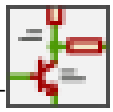
この章では KiCad を使ってどのように電子回路図を描画するかを学びます。

4.1 Eeschema の使用

1. Windows の場合、kicad.exe を実行して下さい。Linux の場合、端末で *kicad* とタイプして下さい。KiCad プロジェクト・マネージャのメイン画面が現れるでしょう。ここからあなたは次の 8 つの独立したソフトウェア・ツールを使うことができます: *Eeschema*, コンポーネント・ライブラリ・エディタ, *Pcbnew*, フットプリント・エディタ, *GerbView*, *Bitmap2Component*, *PCB Calulator* そして *Pl Editor* です。主要なツール群がどのように使われるのか理解するために KiCad 作業の流れの図を参照して下さい。



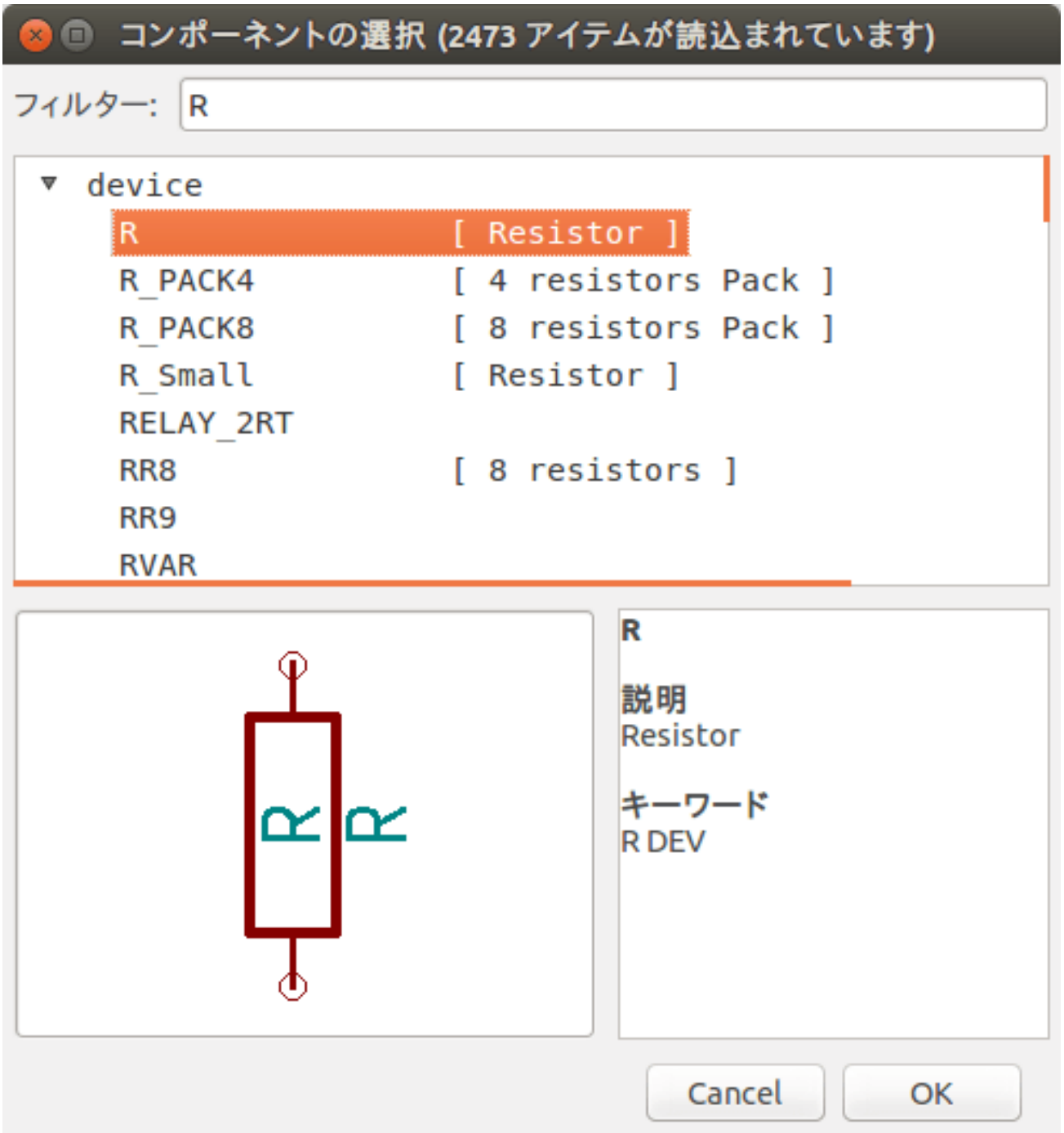
2. Create a new project: **File** → **New** → **Project**. Name the project file *tutorial1*. The project file will automatically take the extension ".pro". The exact appearance of the dialog depends on the used platform, but there should be a checkbox for creating a new directory. Let it stay checked unless you already have a dedicated directory. All your project files will be saved there.

3. 回路図の作成から始めてみましょう。回路図エディタ *Eeschema* を起動します  。左から 1 番目のボタンです。

4. Click on the *Page Settings* icon on the top toolbar. Set the appropriate *paper size* (*A4,8.5x11* etc.) and enter the Title as *Tutorial1*. You will see that more information can be entered here if necessary. Click OK. This information will populate the schematic sheet at the bottom right corner. Use the mouse wheel to zoom in. Save the whole schematic: **File** → **Save**

5. We will now place our first component. Click on the *Place symbol* icon in the right toolbar. You may also press the *Add Symbol* hotkey [a].

6. Click on the middle of your schematic sheet. A *Choose Symbol* window will appear on the screen. We′ re going to place a resistor. Search / filter on the *R* of **R**esistor. You may notice the *Device* heading above the Resistor. This *Device* heading is the name of the library where the component is located, which is quite a generic and useful library.



7. Double click on it. This will close the *Choose Symbol* window. Place the component in the schematic sheet by clicking where you want it to be.

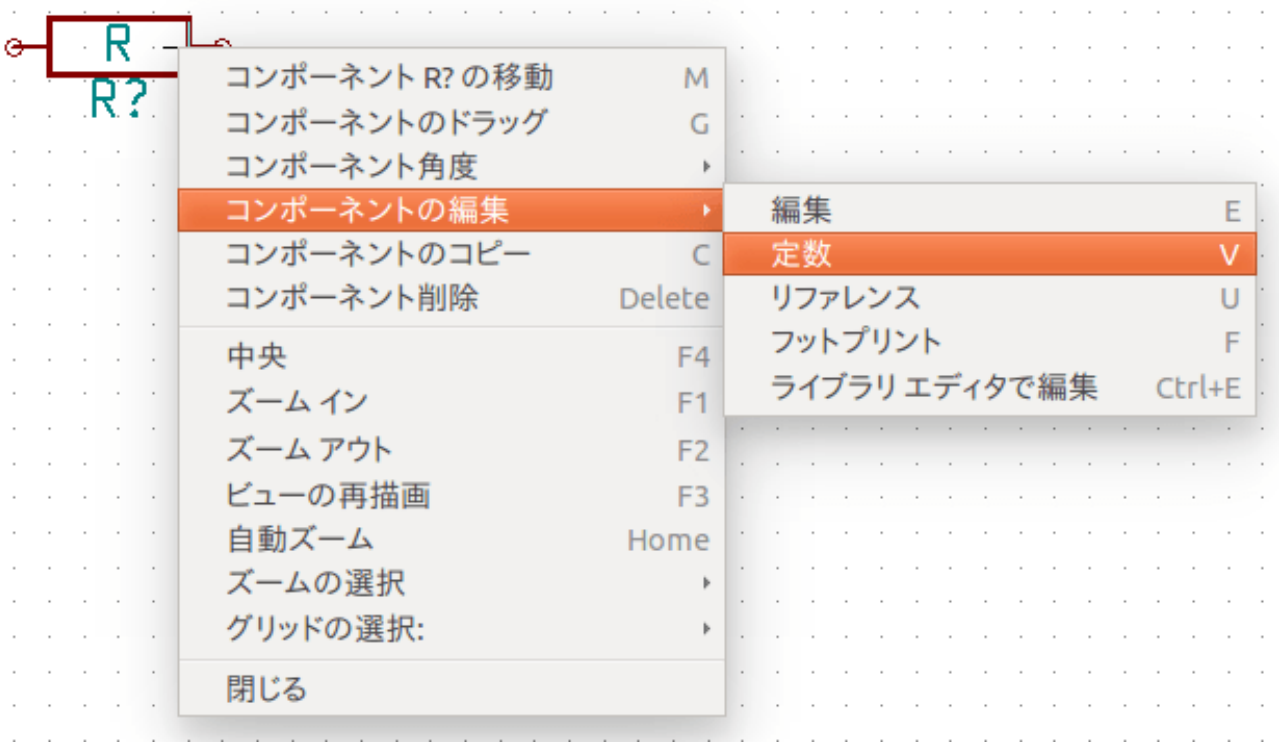
8. ズームインするためにコンポーネント上で拡大鏡をクリックします。あるいは、ズームインとズームアウトをするためにマウスのホイールを使います。水平方向と垂直方向にパンするにはマウスの（中央）ホイールを押します。

9. コンポーネント R の上にマウス・カーソルを合わせて、[r] を押します。コンポーネントが回転します。回転させるためにコンポーネント上でクリックする必要はありません。

注意

If your mouse was also over the *Field Reference* (*R*) or the *Field Value* (*R?*), a menu will appear. You will see the *Clarify Selection* menu often in KiCad; it allows working on objects that are on top of each other. In this case, tell KiCad you want to perform the action on the *Symbol* ⋯*R*⋯.

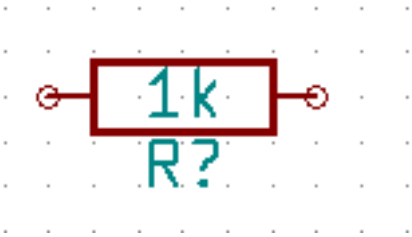
10. Right click in the middle of the component and select **Properties** → **Edit Value**. You can achieve the same result by hovering over the component and pressing [v]. Alternatively, [e] will take you to the more general Properties window. Notice how the right-click menu below shows the hotkeys for all available actions.



11. The Edit Value Field window will appear. Replace the current value *R* with *1 k*. Click OK.

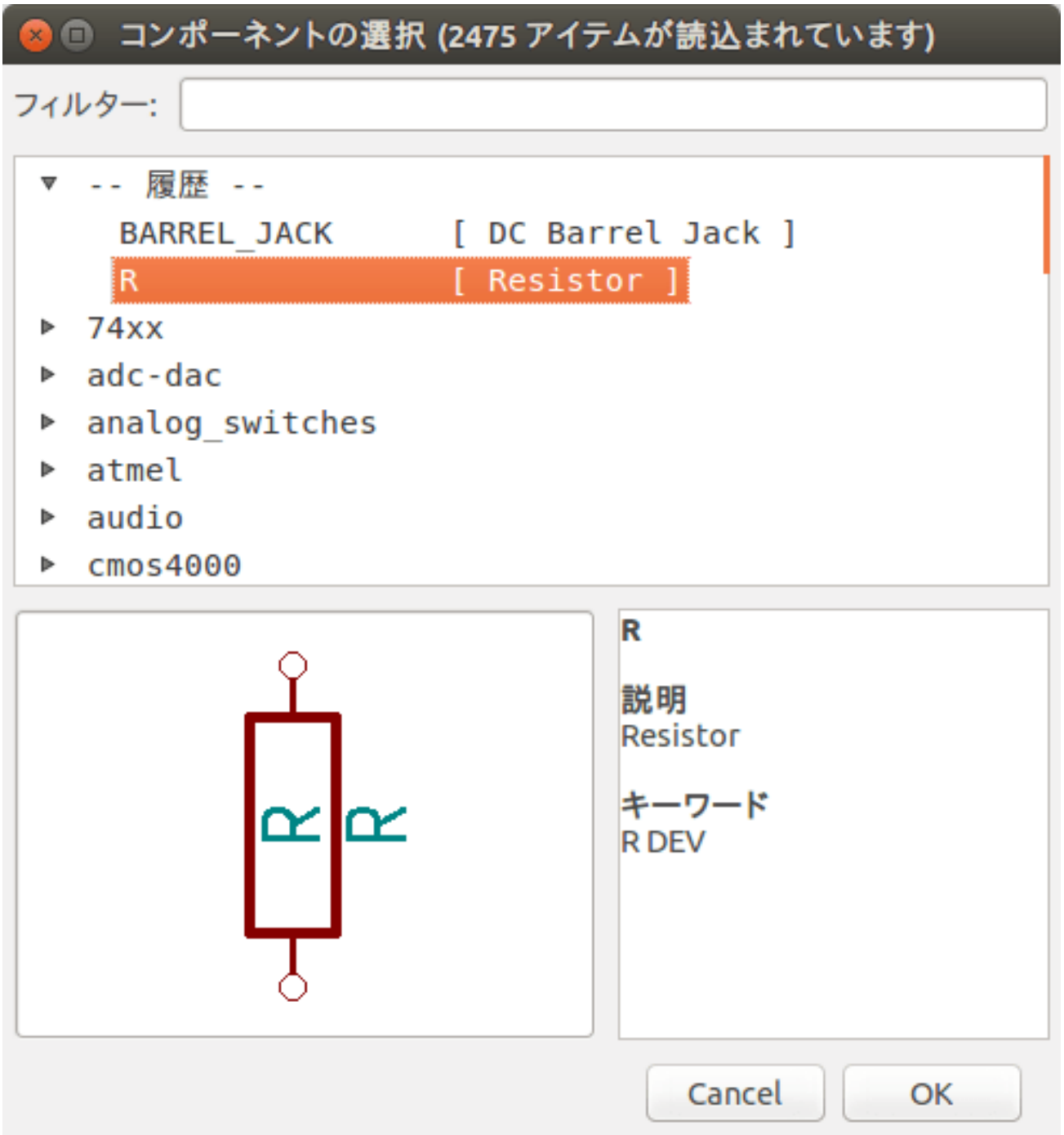
注意

リファレンス・フィールド (R?) を変更しないで下さい。これは後ほど自動的に行われます。これで抵抗器の中の定数は *1k* となりました。



12. To place another resistor, simply click where you want the resistor to appear. The symbol selection window will appear again.

13. 前に選んだ抵抗器が履歴リスト上に *R* として表示されています。OK をクリックしてコンポーネントを配置します。



14. In case you make a mistake and want to delete a component, right click on the component and click *Delete*. This will remove the component from the schematic. Alternatively, you can hover over the component you want to delete and press [Delete].

15. 既に回路図シート上にあるコンポーネントにマウス・カーソルを合わせて [c] を押すことにより、複製することができます。新しく複製したコンポーネントを置きたい場所をクリックして下さい。

16. Right click on the second resistor. Select *Drag*. Reposition the component and left click to drop. The same

functionality can be achieved by hovering over the component and by pressing [g]. [r] will rotate the component while [x] and [y] will flip it about its x- or y-axis.

注意

Right-Click → **Move** or [m] is also a valuable option for moving anything around, but it is better to use this only for component labels and components yet to be connected. We will see later on why this is the case.

---

17. 2 つ目の抵抗器にマウス・カーソルを置いて [v] を押して編集します。*R* を *100* で置き換えます。"Ctrl+Z" によってどんな編集操作も「元に戻す」ができます。

18. Change the grid size. You have probably noticed that on the schematic sheet all components are snapped onto a large pitch grid. You can easily change the size of the grid by **Right-Click** → **Grid**. *In general, it is recommended to use a grid of 50.0 mils for the schematic sheet.*

19. We are going to add a component from a library that isn′ t configured in the default project. In the menu, choose **Preferences** → **Manage Symbol Libraries**. In the Symbol Libraries window you can see two tabs: Global Libraries and Project Specific Libraries. Each one has one sym-lib-table file. For a library (.lib file) to be available it must be in one of those sym-lib-table files. If you have a library file in your file system and it′ s not yet available, you can add it to either one of the sym-lib-table files with **Browse Libraries**. For practise we will now add a library which already is available.

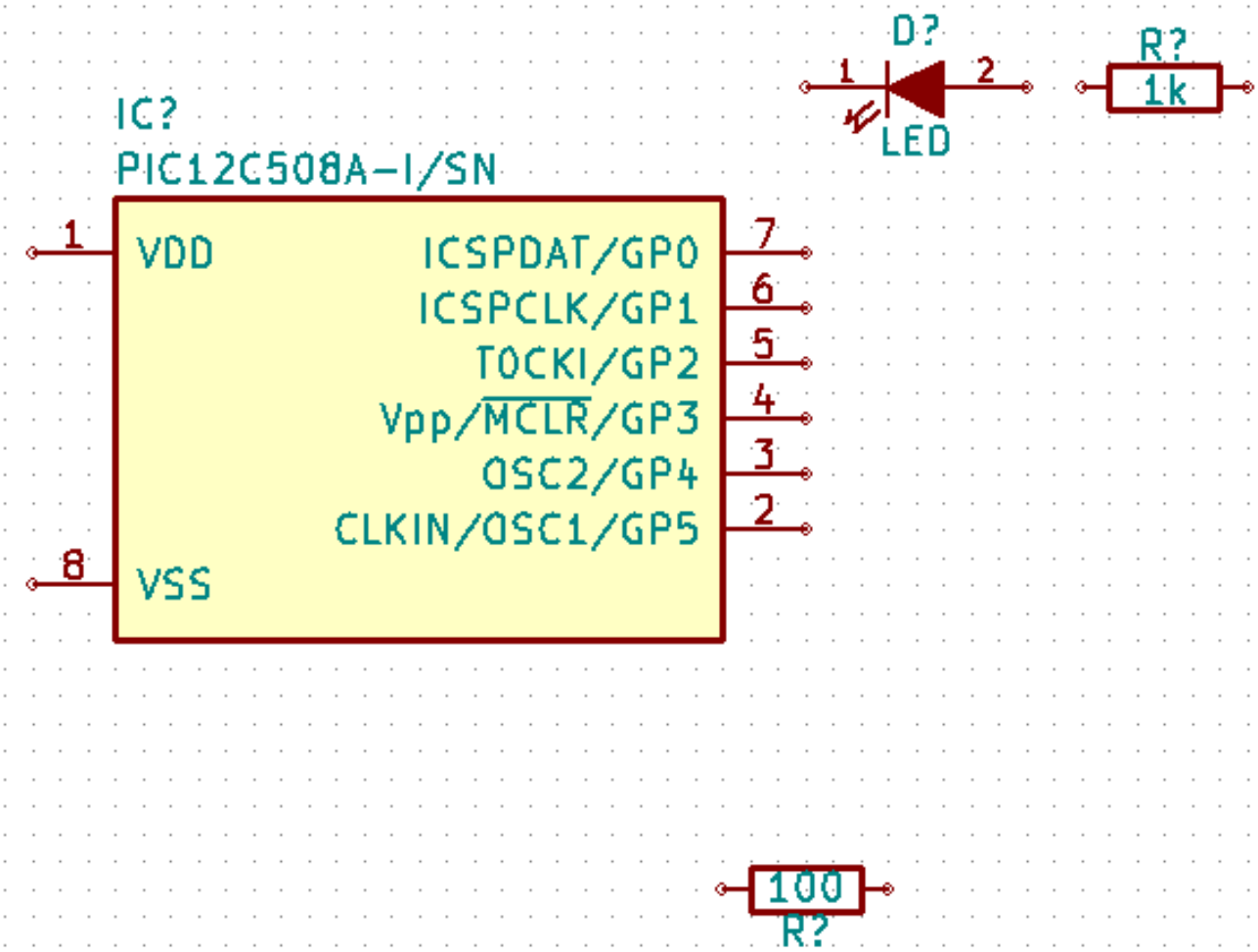
20. You need to find where the official KiCad libraries are installed on your computer. Look for a `library` directory containing a hundred of `.dcm` and `.lib` files. Try in `C:\Program Files (x86)\KiCad\share\` (Windows) and `/usr/share/kicad/library/` (Linux). When you have found the directory, choose and add the *MCU_Microchip_PIC12.lib* library and close the window. You will get a warning that the name already exists in the list; add it anyways. It will be added to the end of of the list. Now click its nickname and change it to *microchip_pic12mcu*. Close the Symbol Libraries window with OK.

21. Repeat the add-component steps, however this time select the *microchip_pic12mcu* library instead of the *Device* library and pick the *PIC12C508A-ISN* component.

22. マイクロコントローラのコンポーネントにマウス・カーソルを置きます。再び [x] と [y] がコンポーネントを反転させることに注目しましょう。元の方向へコンポーネントを戻します。

23. Repeat the add-component steps, this time choosing the *Device* library and picking the *LED* component from it.

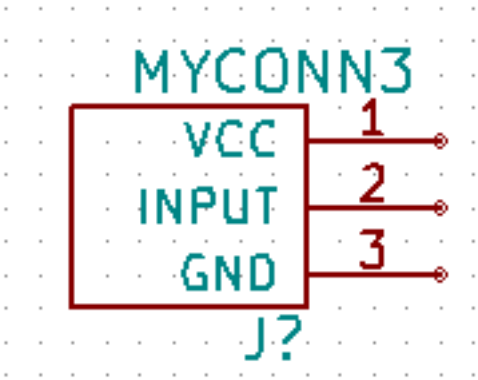
24. 回路図シート上の全てのコンポーネントを以下のように整理します。



25. 私達は 3 ピンのコネクタのために *MYCONN3* という回路図コンポーネントを作る必要があります。「KiCad 回路図コンポーネントの作成」 の章にジャンプし、一からコンポーネントを作る方法を学び、この章に戻ってきて下さい。

26. さあ、新たに作成されたコンポーネントを配置することができるようになりました。[a] を押して *myLib* ライブラリにある *MYCONN3* コンポーネントを選びます。

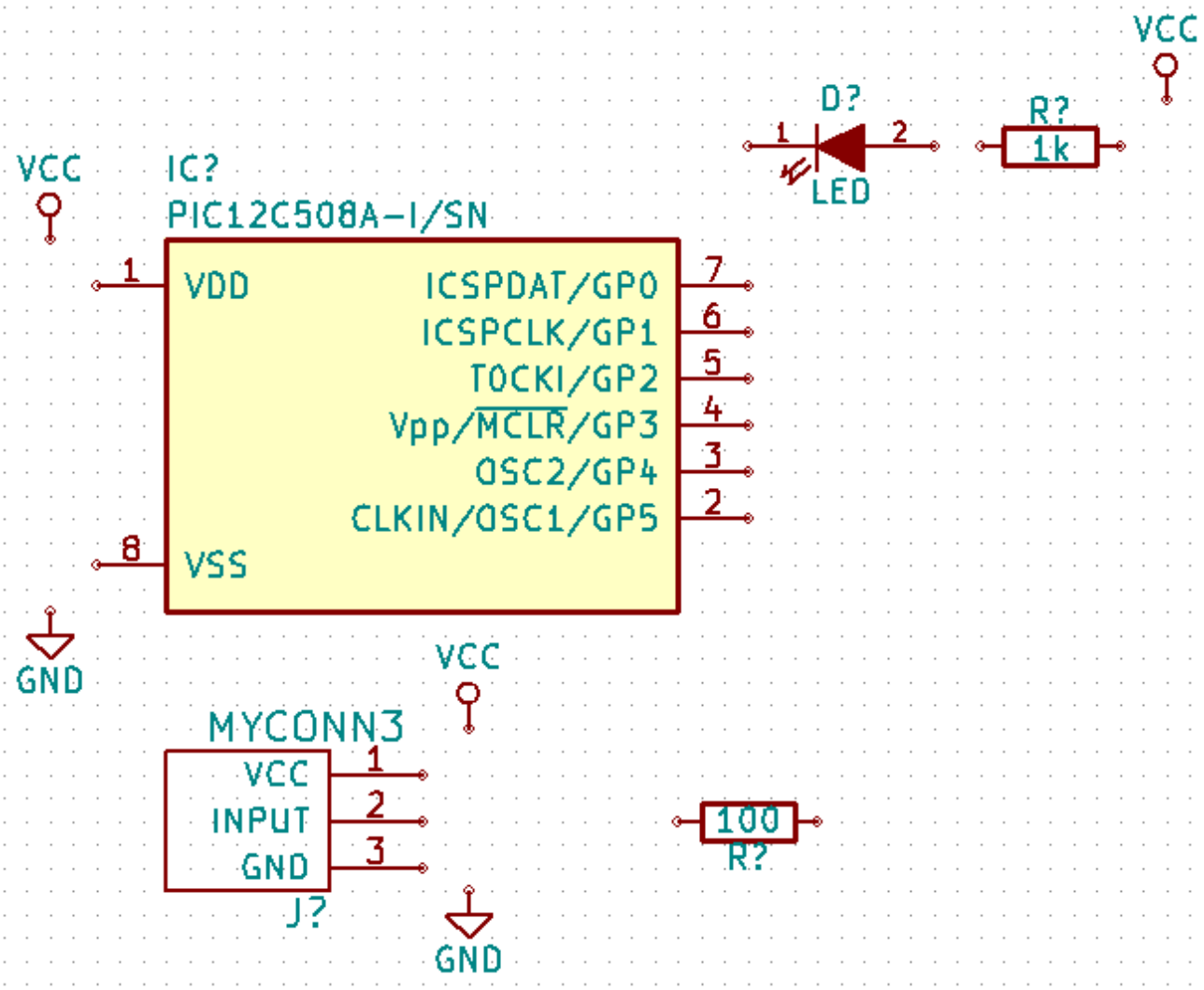
27. コンポーネントの識別子 *J?* が *MYCONN3* ラベルの下に見えます。位置を変えたいなら、*J?* の上で右クリックして リファレンスの移動 ([m] と同等) をクリックします。これを行う前か最中にズーム・インすると便利かもしれません。コンポーネントの *J?* を以下に示すように再配置します。ラベルは何回でも好きなだけ動かすことができます。



28. It is time to place the power and ground symbols. Click on the *Place power port* button on the right toolbar. Alternatively, press [p]. In the component selection window, scroll down and select *VCC* from the *power* library. Click OK.

29. VCC 部品を配置するために、1k の抵抗器のピンの上方をクリックします。マイクロコントローラの *VDD* の上方をクリックします。コンポーネント選択履歴の中から *VCC* を選び、VDD ピンの隣に配置します。追加作業を繰り返して、VCC 部品を *MYCONN3* の VCC ピンの上方に配置します。

30. ピン追加のステップを続けますが、今度は GND 部品を選択します。GND 部品を *MYCONN3* の GND ピンの下に配置します。別の GND シンボルを、マイクロコントローラの VSS ピンの左に配置します。回路図はこのようになっているはずです。



31. 次に、全てのコンポーネントの配線をします。右ツールバーの ワイヤの配置アイコン をクリックします。

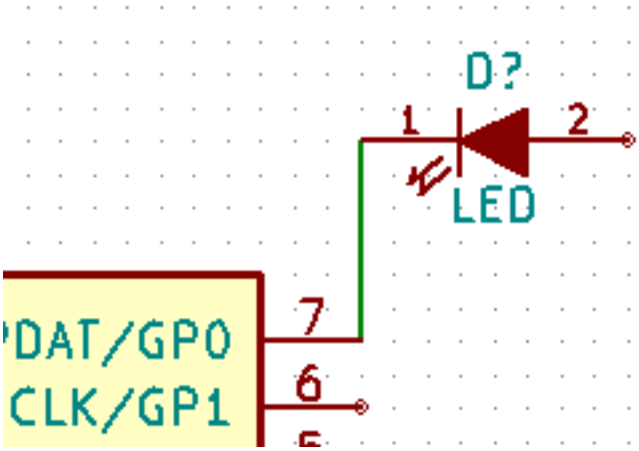
注意

Be careful not to pick *Place bus*, which appears directly beneath this button but has thicker lines. The section Bus Connections in KiCad will explain how to use a bus section.

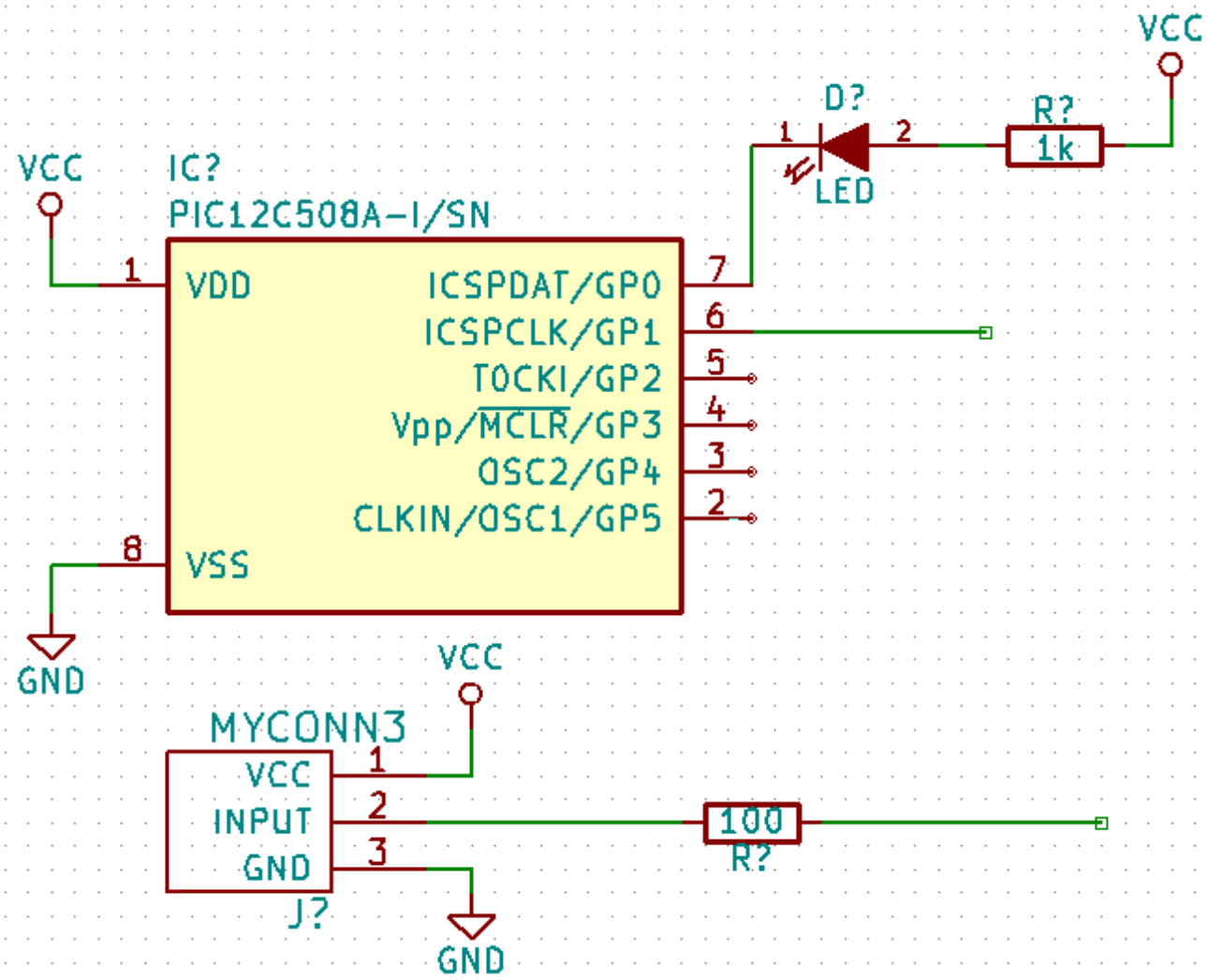
---

32. マイクロコントローラ 7 ピンの端にある小さな円をクリックし、それから LED の 1 ピンの小さな円をクリックします。接続を配置する時にズーム・インすることができます。

注意
配線されたコンポーネントを再配置したい場合、[g] (掴む) を使うことが重要で、[m] (移動) は使うべきではありません。(掴む) を使うことでワイヤの接続を維持できます。コンポーネントの移動方法を忘れたならステップ 24 を見直しましょう。



33. この処理を繰り返して以下の全てのコンポーネントの配線を済ませます。ワイヤを終端するにはダブルクリックをします。VCC と GND のシンボルへの配線は、VCC シンボルなら下部に、GND シンボルなら上部中央にタッチします。下のスクリーンショットを見てください。



34. We will now consider an alternative way of making a connection using labels. Pick a net labelling tool by clicking on the *Place net label* icon A on the right toolbar. You can also use [l].

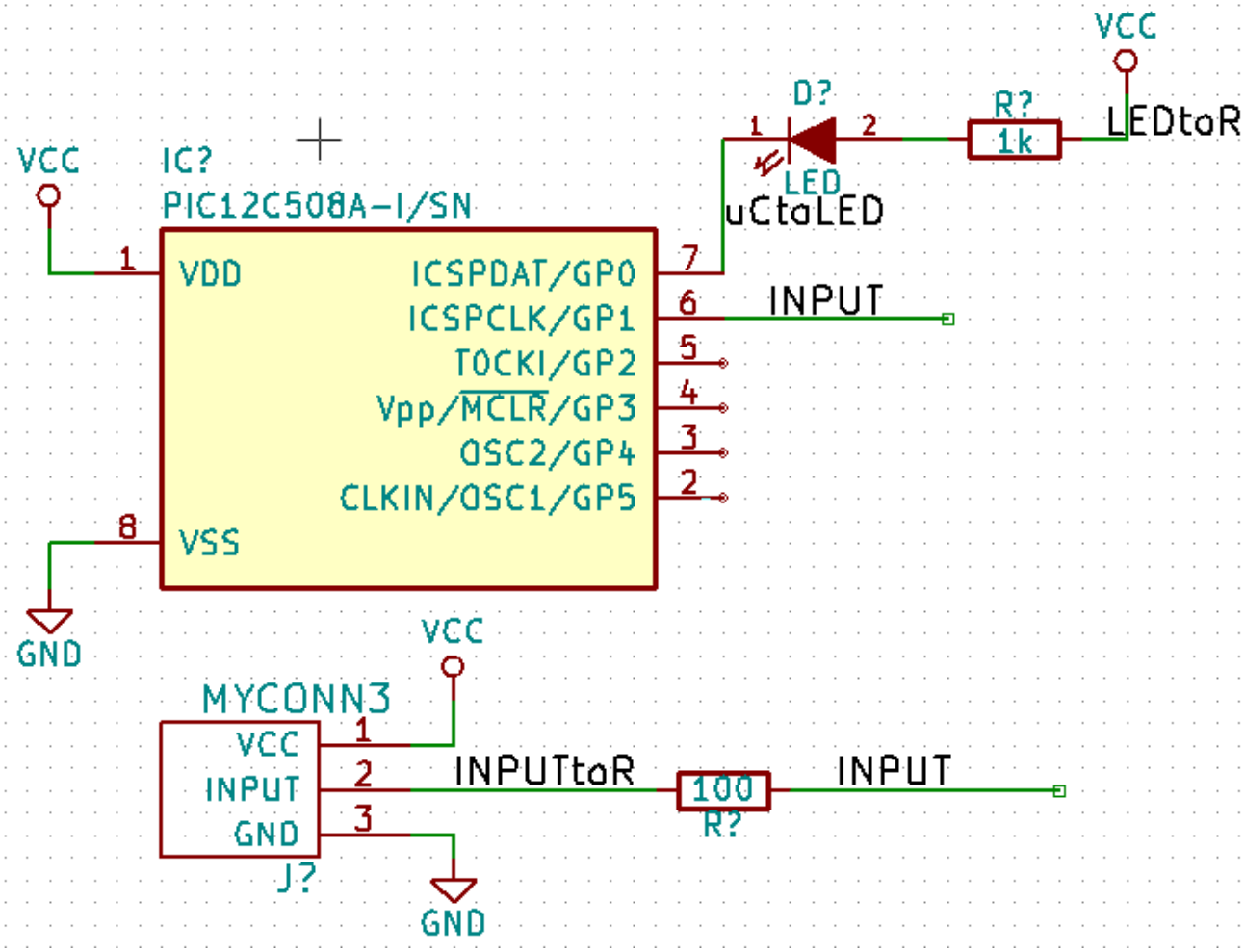
35. マイクロコントローラのピン 6 に接続されているワイヤの中程をクリックします。このラベルに *INPUT* と名前をつけます。

36. 同じ手順に従い 100Ω の抵抗器の右にある配線に別のラベルを配置します。同様に *INPUT* と名前を付けます。2 つの同じ名前を持つラベルは、見えない接続を PIC のピン 6 と 100Ω 抵抗器の間に作ります。これは線が混雑した複雑なデザインの場合に便利な配線テクニックです。ラベルを配置するのに必ずしもワイヤは必要ではなく、単純にピンにラベルを付けることができます。

37. ラベルはまた、単に配線に有用な名前をつける目的で使うこともできます。PIC のピン 7 にラベルを付けて *uCtoLED* と名前をつけたり、抵抗器と LED の間のワイヤに *LEDtoR* と名前をつけたり、*MYCONN3* と抵抗器の間のワイヤに *INPUTtoR* つけたり、です。

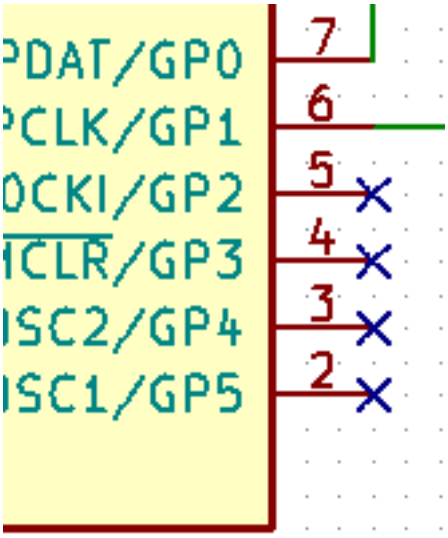
38. VCC と GND の線にラベルをつける必要はありません。そのラベルは、接続されている電源オブジェクトから暗黙的に定義されています。

39. 下図に最終的な結果がどのように見えるかを示します。



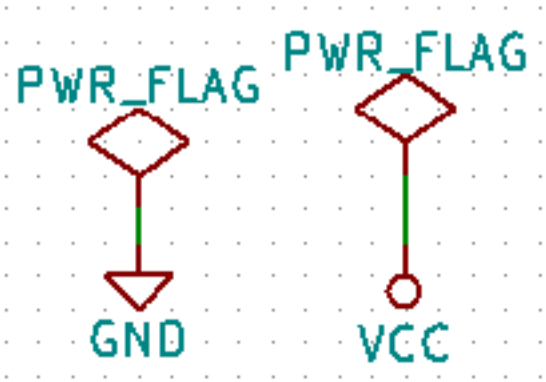
40. それでは接続されていないワイヤに対処しましょう。接続されていないピンやワイヤは KiCad にチェックされた時に警告されます。これらの警告を避けるため、ワイヤが接続されていないのが意図的であることをプログラムに指示することができます。

41. Click on the *Place no connection flag* icon on the right toolbar. Click on pins 2, 3, 4 and 5. An X will appear to signify that the lack of a wire connection is intentional.



42. 見えない電源ピンを持つコンポーネントがあります。左ツールバーの 非表示ピンを表示のアイコン をクリックすることでそれらを見えるようにできます。VCC と GND の名前付けが尊重される場合、隠れた電源ピンは自動的に接続されます。一般的には、隠れた電源ピンを作らないように努力すべきです。

43. It is now necessary to add a *Power Flag* to indicate to KiCad that power comes in from somewhere. Press [a] and search for *PWR_FLAG* which is in *power* library. Place two of them. Connect them to a GND pin and to VCC as shown below.



注意
これにより典型的な回路図チェックの警告を避けることができます: 警告 power_in ピンは駆動されていません (Net xx)

---

44. Sometimes it is good to write comments here and there. To add comments on the schematic use the *Place text* icon T on the right toolbar.

45. All components now need to have unique identifiers. In fact, many of our components are still named *R?* or *J?*. Identifier assignation can be done automatically by clicking on the *Annotate schematic symbols* icon on the top toolbar.

46. 回路図のアノテーションウィンドウでは、全ての回路図、階層を使用を選択し、アノテーションボタンをクリックします。確認のメッセージに OK をクリックし、閉じるをクリックします。コンポーネントの全ての *?* がどのように数字に置き換えられたか注目してください。各識別子はユニークです。私達の例では、*R1* 、*R2* 、*U1* 、*D1* そして *J1* と名付けられました。

47. ここで回路図のエラーをチェックします。トップツールバーの エレクトリカルルールチェックの実行アイコン をクリックします。実行ボタンをクリックします。接続されていないワイヤ等のエラーや警告のレポートが通知されます。エラー 0、警告 0 にしましょう。エラーや警告がある場合には、小さな緑色の矢印が回路図上のエラーや警告のある場所に表示されます。よりエラーの情報を得るため *ERC* レポートファイルの生成をチェックし、実行ボタンを再度押します。

注意
If you have a warning with "No default editor found, you must choose it", try setting the path to `c:\windows\notepad.exe` (windows) or `/usr/bin/gedit` (Linux).

---

48. The schematic is now finished. We can now create a Netlist file to which we will add the footprint of each component. Click on the *Generate netlist* icon on the top toolbar. Click on the *Generate Netlist* button and save under the default file name.

49. ネットリストを生成したら、トップツールバーにある *CvPcb(*コンポーネントとフットプリントの関連付け*)* を実行のアイコン をクリックします。ファイルが存在しませんのエラーが表示されたら OK をクリックします。

50. *Cvpcb* allows you to link all the components in your schematic with footprints in the KiCad library. The pane on the center shows all the components used in your schematic. Here select *D1*. In the pane on the right you have all the available footprints, here scroll down to *LED_THT:LED-D5.0mm* and double click on it.

51. 右側のペインに、利用可能なフットプリントの内、選択されたサブグループだけを表示することもできます。これは KiCad が適切なフットプリントだけをあなたに提示しようとするためです。このフィルタを有効/無効するにはこれらのアイコン をクリックします。

52. For *U1* select the *Package_DIP:DIP-8_W7.62mm* footprint. For *J1* select the *Connector:Banana_Jack_3Pin* footprint. For *R1* and *R2* select the *Resistor_THT:R_Axial_DIN0207_L6.3mm_D2.5mm_P2.54mm_Vertical* footprint.

53. If you are interested in knowing what the footprint you are choosing looks like, you can click on the *View selected footprint* icon for a preview of the current footprint.

54. You are done. You can save the schematic now by clicking **File** → **Save Schematic** or with the button *Apply, Save Schematic & Continue.*

55. You can close *Cvpcb* and go back to the *Eeschema* schematic editor. If you didn' t save it in Cvpcb save it now by clicking on **File** → **Save**. Create the netlist again. Your netlist file has now been updated with all the footprints. Note that if you are missing the footprint of any device, you will need to make your own footprints. This will be explained in a later section of this document.

56. KiCad プロジェクト・マネージャに切り替えます。

57. ネットリストには、全てのコンポーネントとそのそれぞれのピン接続が記述されています。ネットリストはテキストファイルなので、容易に検査したり編集したり書いたりすることができます。

注意
ライブラリファイル (**.lib*) もテキストファイルなので、同様に容易に編集したり記述することができます。

---

58. To create a Bill Of Materials (BOM), go to the *Eeschema* schematic editor and click on the *Generate bill of materials* icon on the top toolbar. By default there is no plugin active. You add one, by clicking on **Add Plugin** button. Select the *.xsl file you want to use, in this case, we select *bom2csv.xsl.*

注意

Linux:

もし xsltproc が見つからない場合、ダウンロードして次のようにインストールすることができます:

```
sudo apt-get install xsltproc
```

Ubuntu のような Debian 由来のディストリビューション用。または

```
sudo yum install xsltproc
```

RedHat 由来のディストリビューション用。もし 2 種類のディストリビューション以外のものをお使いの場合は、お使いのディストリビューションのパッケージ・マネージャー・コマンドを使って xsltproc パッケージをインストールして下さい。

xsl ファイルが置かれる場所: */usr/lib/kicad/plugins/*.

Apple OS X:

もし xsltproc が見つからない場合、これが含まれるよう Apple のサイトから Apple Xcode ツールをインストールするか、ダウンロードして次のようにインストールすることができます:

```
brew install libxslt
```

xsl ファイルが置かれる場所: */Library/Application Support/kicad/plugins/*.

Windows:

xsltproc.exe と含まれる xsl ファイルは、< *KiCad install directory* >*bin* と < *KiCad install directory* > *bin**scripting**plugins* にそれぞれに置かれるでしょう。

全てのプラットホーム:

最新の bom2csv.xsl はここから入手できます:

https://raw.githubusercontent.com/KiCad/kicad-source-mirror/master/eeschema/plugins/xsl_scripts/-bom2csv.xsl

KiCad は自動的にコマンドを生成します、例えばこのように:

```
xsltproc -o "%O" "/home/<user>/kicad/eeschema/plugins/bom2csv.xsl" "%I"
```

拡張子を与えたい場合、このコマンドラインを変更します:

```
xsltproc -o "%O.csv" "/home/<user>/kicad/eeschema/plugins/bom2csv.xsl" "%I"
```

ヘルプボタンを押すことで更に情報が得られます。

59. さあ 生成を押してみましょう。ファイル (プロジェクトと同じ名前です) はプロジェクトフォルダの中にあります。***.csv** ファイルを LibreOffice Calc か Excel で開いてみましょう。インポートのウィンドウが表示されたら OK を押します。

あなたはこれで次の章の PCB レイアウトのパートに進む準備ができました。ですが先に進む前に、コンポーネントのピンをバスで接続する方法をさっと見ておきましょう。

4.2 KiCad でのバス接続

コンポーネント A の連続したピンと、別のコンポーネント B の連続したピンを接続しなければならないことがあります。この場合 2 つの選択肢があります: 私達が既に見てきたラベルによる方法とバス接続です。どうするのか見てみましょう。

1. 4 ピンのコネクタが 3 つあり、ピン同士を接続したいのだと仮定しましょう。ラベルを使って ([l] を押す) コネクタ P4 のピン 4 に *a1* という名前のラベルを付けます。[Insert] を押して、ピン 4 の下のピン 3 に自動的に同様の処理を行います。ラベルは自動的に *a2* とリネームされます。

2. [Insert] を更に 2 回押してください。このキーは 最後のアイテムをリピート機能に対応しており、あなたの仕事を楽にしてくれるとても便利なコマンドです。

3. ラベル付け作業を他の 2 つのコネクタ CONN_2 と CONN_3 にも繰り返して終えました。このまま進めて PCB を作成すると、3 つのコネクタがお互い接続されていることでしょう。図 2 に説明したことが表されています。美しく見せるために、図 3 に示すよう、アイコン を使った一連の ワイヤ-バスエントリを配置と、アイコン を使ったバスラインを付け加えることも可能です。PCB にはなんの影響もありませんが。

4. 図 2 の、ピンに接触している短いワイヤは厳密には不要であることを指摘しておきます。実際、ラベルはピンに直接付けることができるのでした。

5. もう一歩進んで、4 つ目のコネクタ CONN_4 を仮定します。何らかの理由により、そのラベル付けは少し異なっています (b1, b2, b3, b4)。私達は *Bus a* と *Bus b* をピンとピンの方法で接続したいのです。ピンにラベル付けする方法 (も可能ですが) は使わずに、代わりに、バス毎に 1 ラベルで、バスラインにラベル付けする方法で実現したいわけです。

6. 前に説明したラベル付けの方法を使って、CONN_4 に接続してラベルを付けます。ピンには b1, b2, b3 そして b4 と名前を付けます。アイコン を使った一連の ワイヤ-バスエントリを配置と、アイコン を使ったバスラインに、ピンを接続します。図 4 を見て下さい。

7. CONN_4 のバスにラベルを付けて ([l] キーを押して) *b[1..4]* と名付けます。

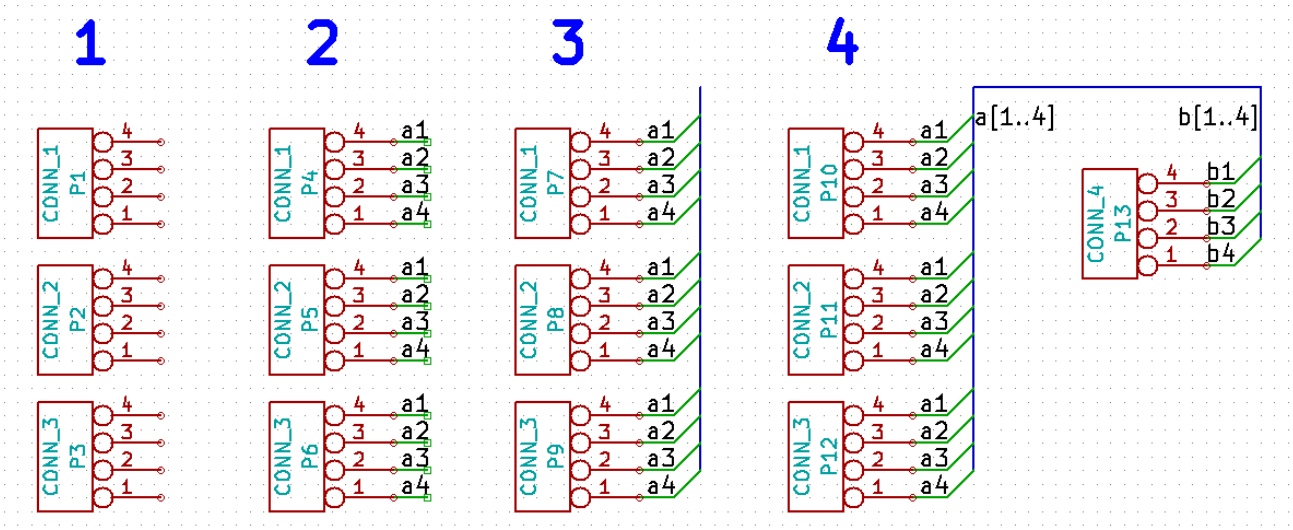
8. Put a label (press [l]) on the previous bus and name it *a[1..4]*.

9. これでボタン を使って、バス a[1..4] とバス b[1..4] を接続することでできるようになりました。

10. 2 つのバスを接続することで、ピン a1 は自動的にピン b1 に接続され、a2 は b2 に接続され、以下同様となります。図 4 は最終結果がどのように見えるか示しています。

    注意
    [Insert] による 最後のアイテムをリピート機能は、繰り返される周期的な作業にうまく使うことができます。例えば、短いワイヤを全てのピンに接続する図 2、図 3、図 4 での作業はこの機能でできました。

11. [Insert] による 最後のアイテムをリピート機能は、アイコン を使った、たくさんの連続した ワイヤ-バスエントリを配置にも適用できます。

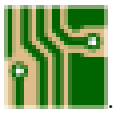

1
2
3
4
CONN_1 P1
CONN_2 P2
CONN_3 P3
CONN_1 P4
CONN_2 P5
CONN_3 P6
CONN_1 P7
CONN_2 P8
CONN_3 P9
CONN_1 P10
CONN_2 P11
CONN_3 P12
CONN_4 P13
a[1..4]
b[1..4]
a1
a2
a3
a4
b1
b2
b3
b4

Chapter 5

プリント基板のレイアウト

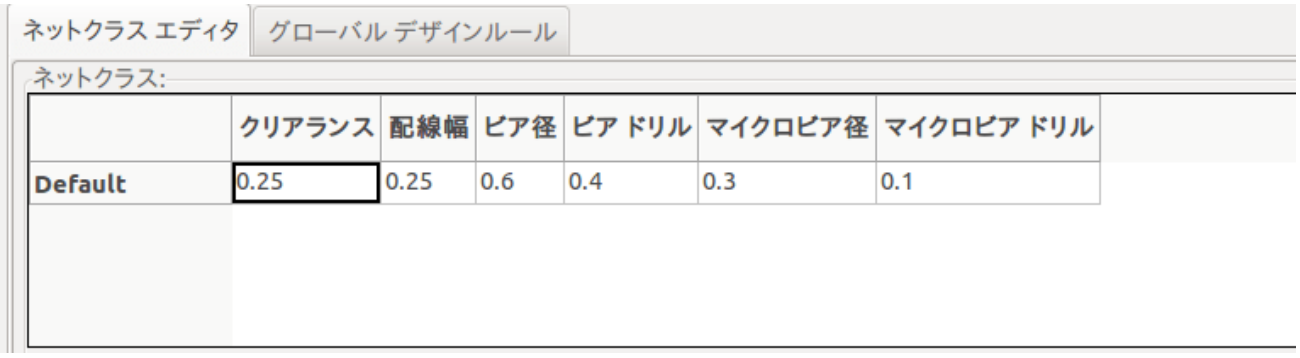
生成したネットリストのファイルを使って PCB のレイアウトを行う時が来ました。これは *Pcbnew* ツールで行います。

5.1 Pcbnew の使用

1. From the KiCad project manager, click on the *Pcb layout editor* icon  . The *Pcbnew* window will open. If you get an error message saying that a **.kicad_pcb* file does not exist and asks if you want to create it, just click Yes.

2. いくつか回路図情報を入力することから始めます。トップツールバーの ページ設定のアイコン をクリックします。適切な ページサイズ (*A4*'、'*8.5x11* など) に設定して、タイトルを *Tutorial 1* と設定します。

3. It is a good idea to start by setting the **clearance** and the **minimum track width** to those required by your PCB manufacturer. In general you can set the clearance to *0.25* and the minimum track width to *0.25*. Click on the **Setup** → **Design Rules** menu. If it does not show already, click on the *Net Classes Editor* tab. Change the *Clearance* field at the top of the window to *0.25* and the *Track Width* field to *0.25* as shown below. Measurements here are in mm.



4. Click on the *Global Design Rules* tab and set *Minimum track width* to *0.25*. Click the OK button to commit your changes and close the Design Rules Editor window.

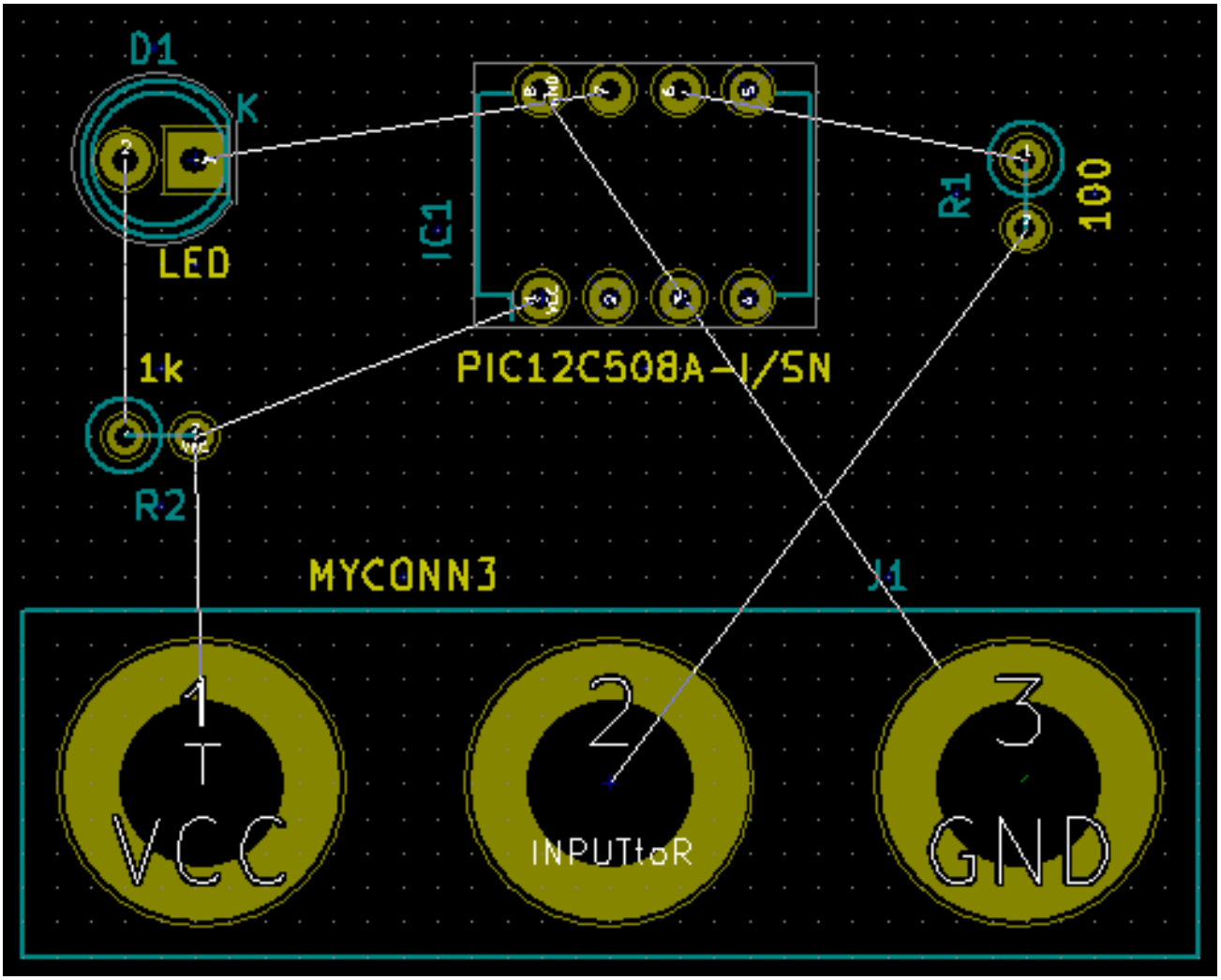
5. Now we will import the netlist file. Click on the *Read netlist* icon on the top toolbar. The netlist file *tutorial1.net* should be selected in the *Netlist file* field if it was created from Eeschema. Click on *Read Current Netlist*. Then click the *Close* button.

6. All components should now be visible. They are selected and follow the mouse cursor.

7. Move the components to the middle of the board. If necessary you can zoom in and out while you move the components. Click the left mouse button.

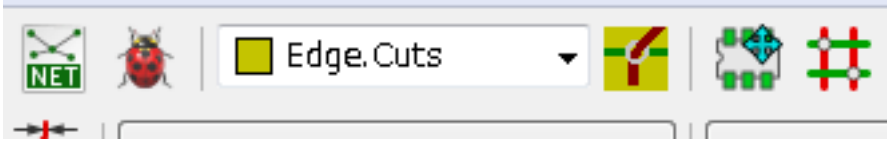
8. 全てのコンポーネントは ラッツネストと呼ばれる細いワイヤの組で接続されています。ボードのラッツネストを非表示のボタン が押されていることを確認します。これでラッツネストが全てのコンポーネントを接続している様子が見えます。

9. You can move each component by hovering over it and pressing [m]. Click where you want to place them. Alternatively you can select a component by clicking on it and then drag it. Press [r] to rotate a component. Move all components around until you minimise the number of wire crossovers.



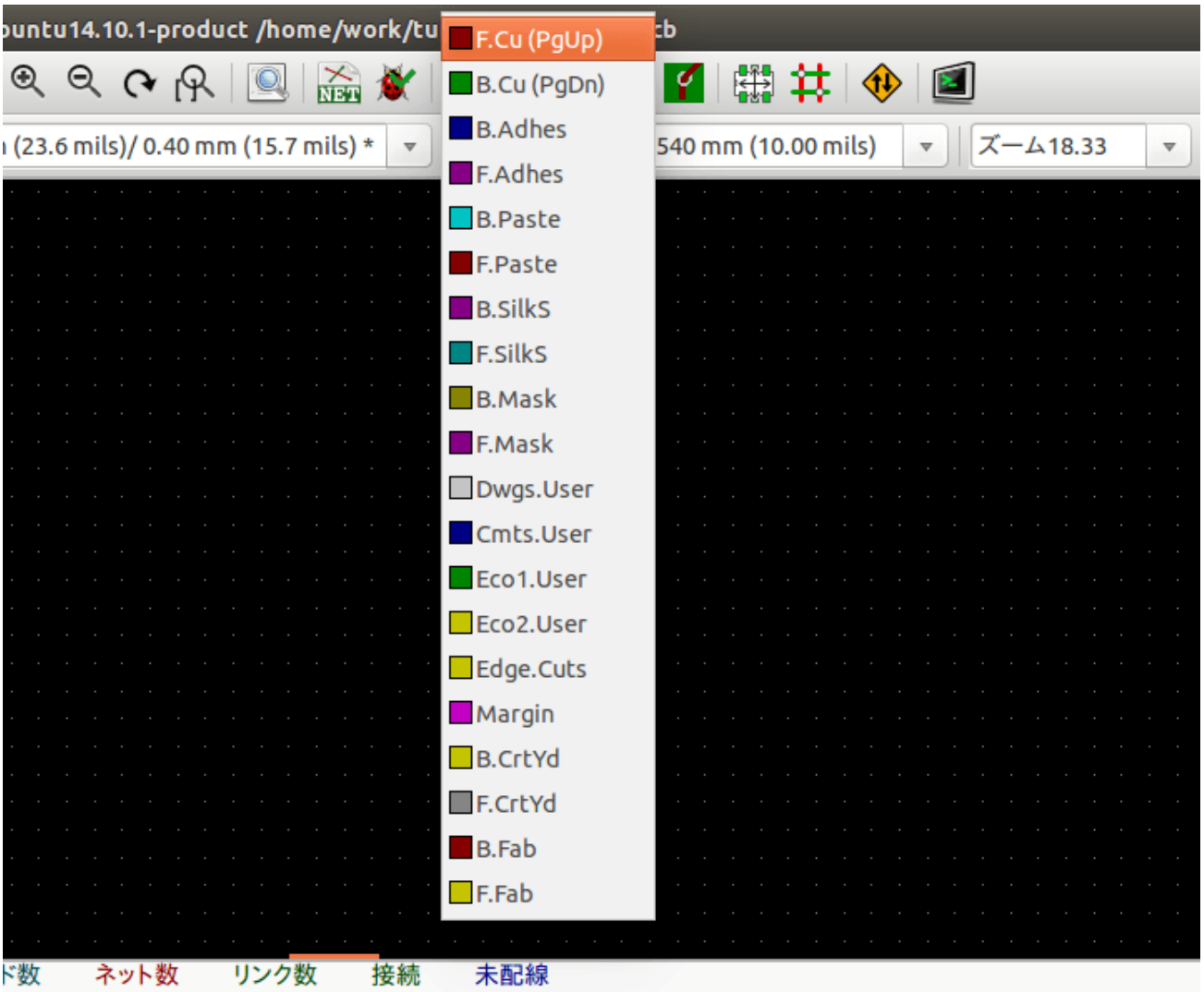
10. Note how one pin of the 100 ohm resistor is connected to pin 6 of the PIC component. This is the result of the labelling method used to connect pins. Labels are often preferred to actual wires because they make the schematic much less messy.

11. Now we will define the edge of the PCB. Select the *Edge.Cuts* layer from the drop-down menu in the top toolbar. Click on the *Add graphic lines* icon on the right toolbar. Trace around the edge of the board, clicking at each corner, and remember to leave a small gap between the edge of the green and the edge of the PCB.



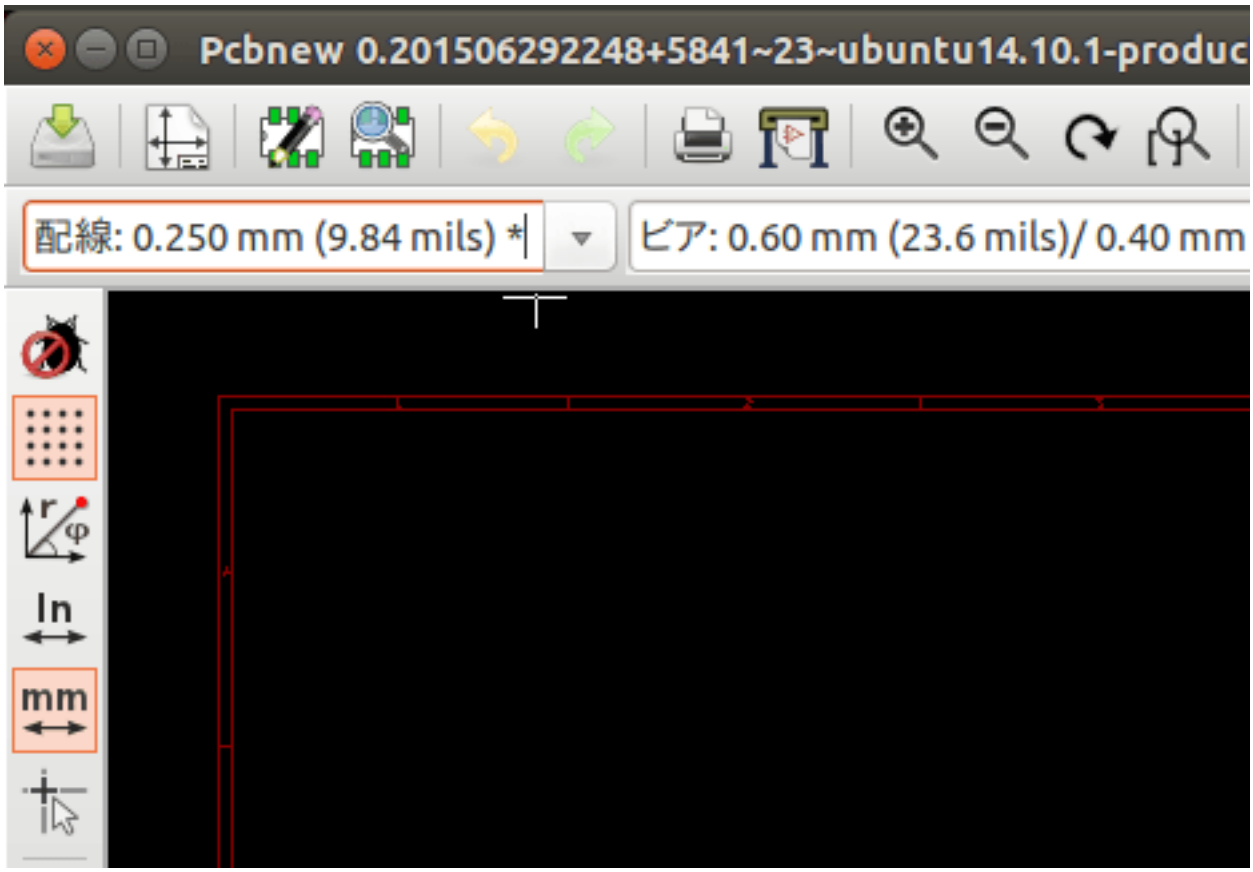
12. 次に GND を除く全てのワイヤを接続してしまいましょう。実際の所、全ての GND の接続は、基板の底部の銅箔面 (*B.Cu* と呼ばれます) に配置されたグランドプレーンを使って一気に行います。

13. 私達はどの銅のレイヤで作業するかを選ばねばなりません。トップツールバーにあるドロップダウンメニューの *F.Cu ("PgUp"* キー*)* を選択します。



14. If you decide, for instance, to do a 4 layer PCB instead, go to **Setup** → **Layers Setup** and change *Copper Layers* to 4. In the *Layers* table you can name layers and decide what they can be used for. Notice that there are very useful presets that can be selected via the *Preset Layer Groupings* menu.

15. Click on the *Route tracks* icon on the right toolbar. Click on pin 1 of *J1* and run a track to pad *R2*. Double-click to set the point where the track will end. The width of this track will be the default 0.250 mm. You can change the track width from the drop-down menu in the top toolbar. Mind that by default you have only one track width available.

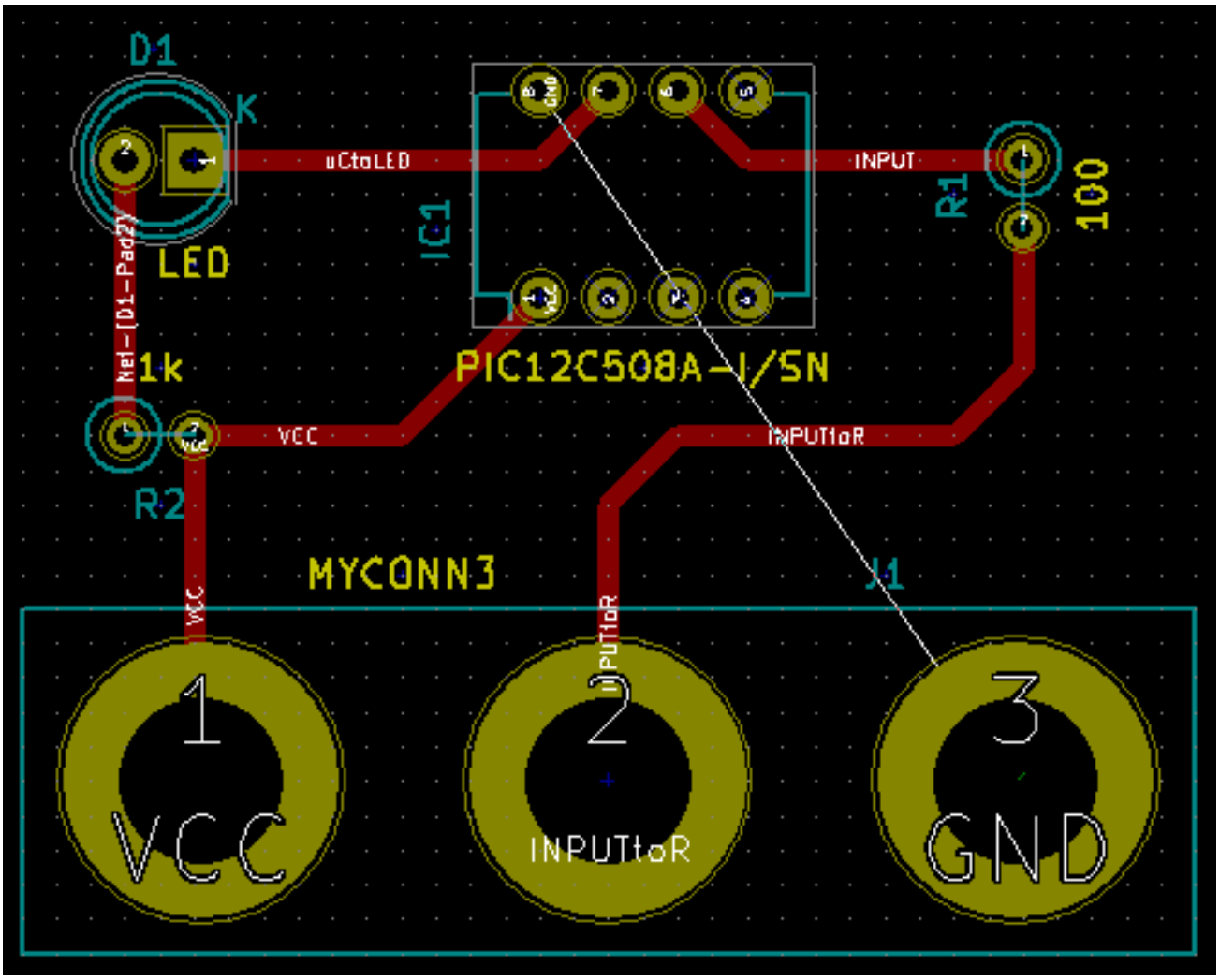


16. If you would like to add more track widths go to: **Setup** → **Design Rules** → **Global Design Rules** tab and at the bottom right of this window add any other width you would like to have available. You can then choose the widths of the track from the drop-down menu while you lay out your board. See the example below (inches).

17. Alternatively, you can add a Net Class in which you specify a set of options. Go to **Setup** → **Design Rules** → **Net Classes Editor** and add a new class called *power*. Change the track width from 8 mil (indicated as 0.0080) to 24 mil (indicated as 0.0240). Next, add everything but ground to the *power* class (select *default* at left and *power* at right and use the arrows).

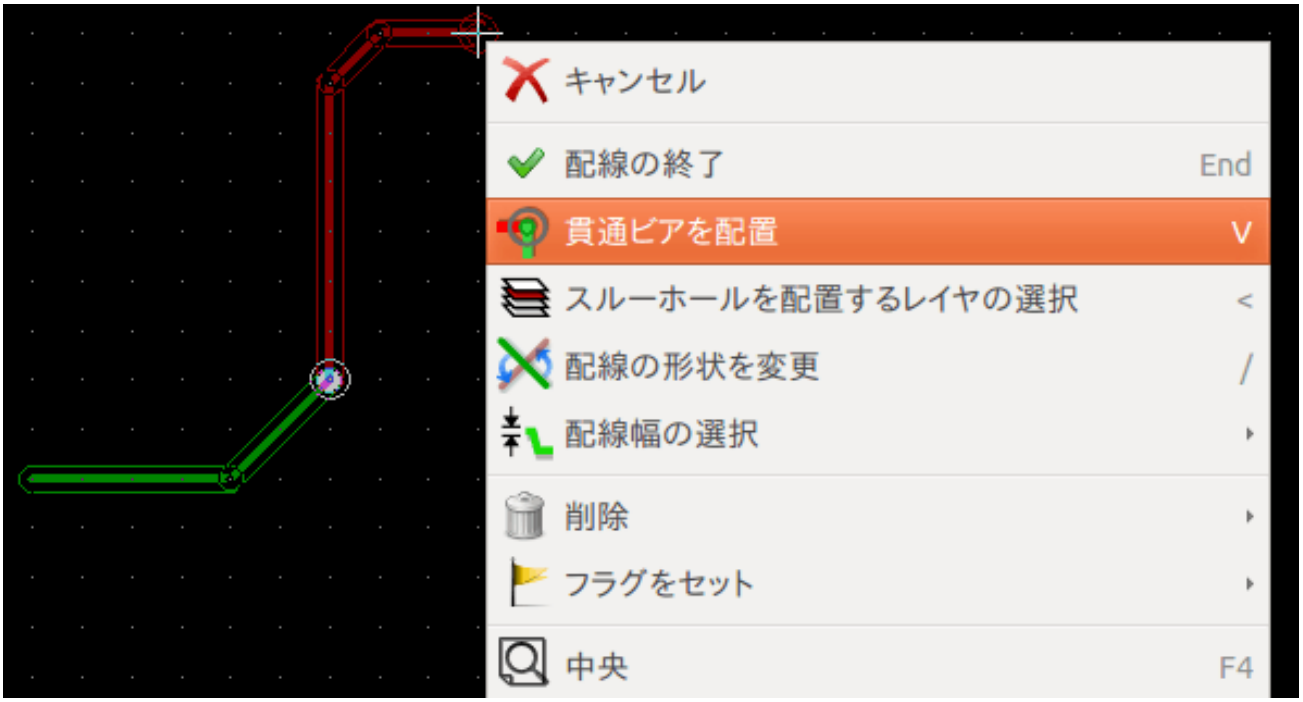
18. If you want to change the grid size, **Right click** → **Grid**. Be sure to select the appropriate grid size before or after laying down the components and connecting them together with tracks.

19. J1 のピン 3 を除く全ての配線が接続されるまで、この作業を繰り返します。あなたの基板は以下の例のようになるでしょう。



20. Let' s now run a track on the other copper side of the PCB. Select *B.Cu* in the drop-down menu on the top toolbar. Click on the *Route tracks* icon . Draw a track between pin 3 of J1 and pin 8 of U1. This is actually not necessary since we could do this with the ground plane. Notice how the colour of the track has changed.

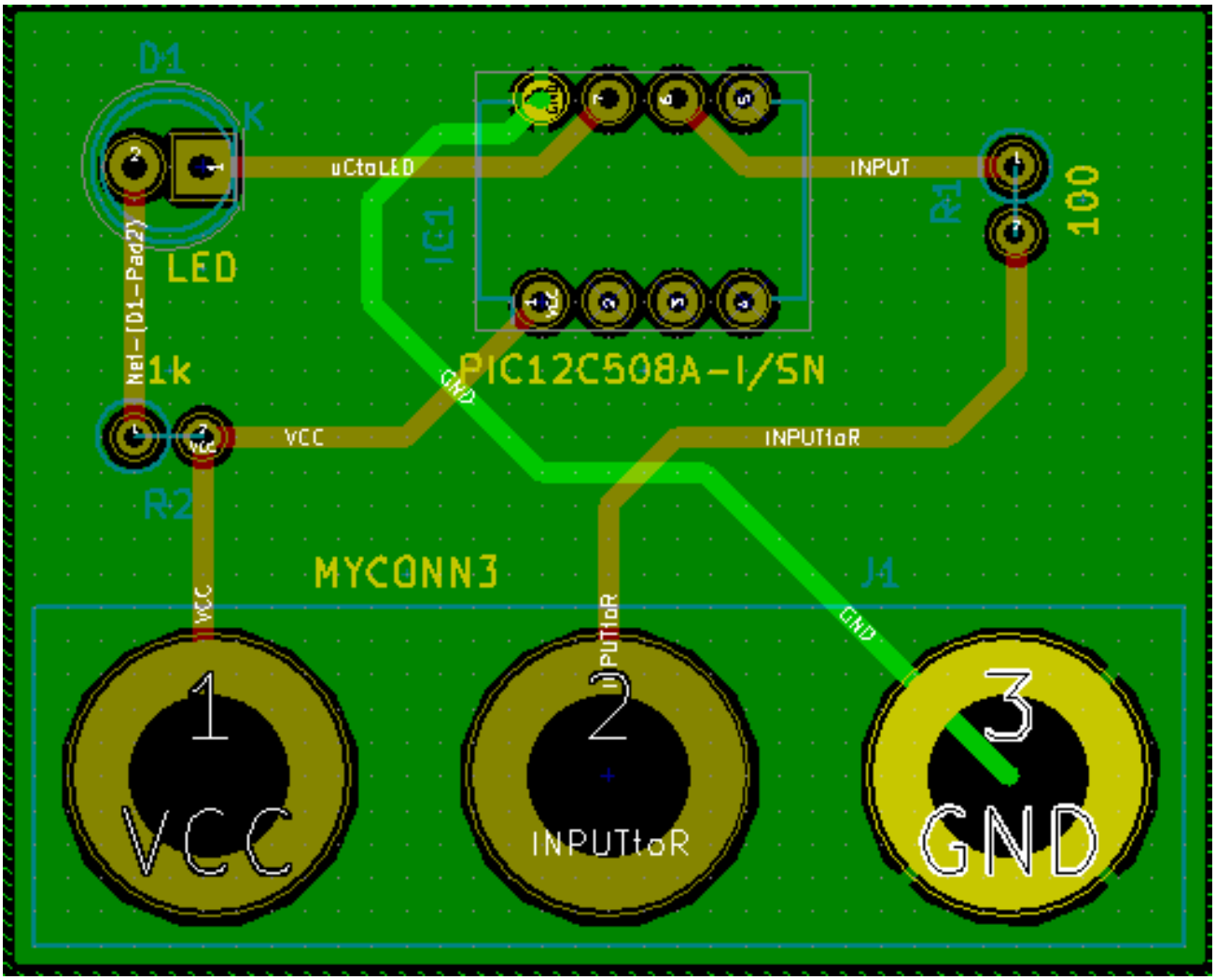
21. レイヤの変更によってピン **A** からピン **B** に行きましょう。ビアを配置することで、配線をしている銅プレーンを変更することが可能です。上面の銅プレーンに配線している時に、右クリックして 貫通ビアの配置を選択するか単に [v] を押します。これにより配線を終えた所で底面のレイヤに行くことができます。



22. When you want to inspect a particular connection you can click on the *Highlight net* icon on the right toolbar. Click on pin 3 of J1. The track itself and all pads connected to it should become highlighted.

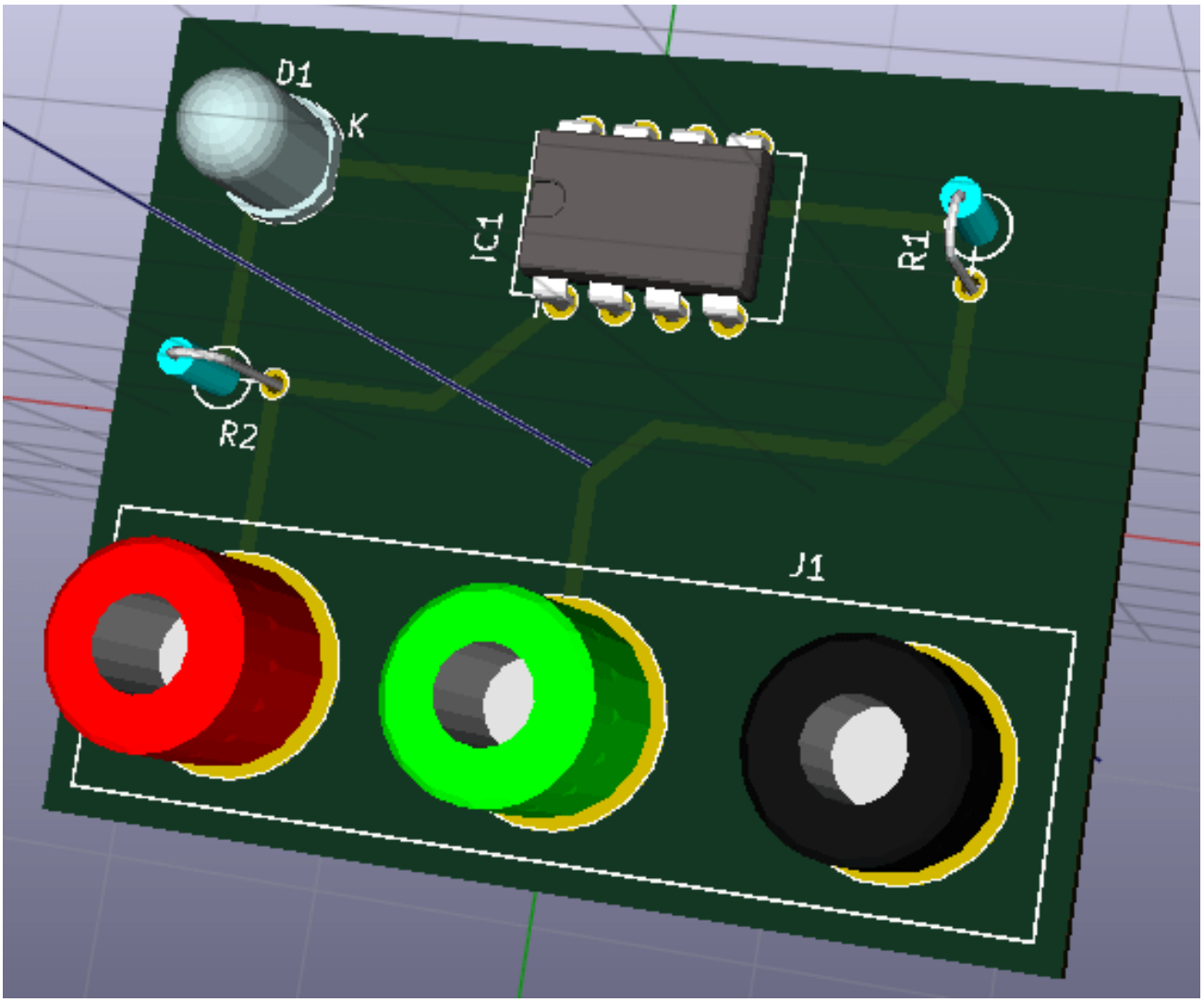
23. Now we will make a ground plane that will be connected to all GND pins. Click on the *Add filled zones* icon on the right toolbar. We are going to trace a rectangle around the board, so click where you want one of the corners to be. In the dialogue that appears, set *Default pad connection* to *Thermal relief* and *Outline slope* to *H,V and 45 deg only* and click OK.

24. Trace around the outline of the board by clicking each corner in rotation. Finish your rectangle by clicking the first corner second time. Right click inside the area you have just traced. Click on *Zones*→'Fill or Refill All Zones'. The board should fill in with green and look something like this:



25. Run the design rules checker by clicking on the *Perform design rules check* icon on the top toolbar. Click on *Start DRC*. There should be no errors. Click on *List Unconnected*. There should be no unconnected items. Click OK to close the DRC Control dialogue.

26. ファイル → 保存をクリックしてファイルを保存します。3D で基板を閲覧するためには 表示 → **3D** ビューアをクリックします。



27. PCB の周囲でマウスをドラッグして PCB を回転させることができます。
28. 基板はこれで完成です。メーカーに送るためにはガーバーファイルを生成する必要があります。

5.2 ガーバーファイルの生成

PCB が完成したら、全てのレイヤのガーバーファイルを生成して、基板を製造してくれるあなたのお好みの PCB メーカーに送ることができます。

1. From KiCad, open the *Pcbnew* board editor.
2. Click on **File** → **Plot**. Select *Gerber* as the *Plot format* and select the folder in which to put all Gerber files. Proceed by clicking on the *Plot* button.
3. ドリルファイルを生成するには、*Pcbnew* から ファイル → プロットを再び行います。デフォルトの設定でよいでしょう。
4. これらは典型的な 2 層 PCB を製造するために選択する必要があるレイヤです:

Layer	KiCad Layer Name	Default Gerber Extension	”Use Protel filename extensions” is enabled
Bottom Layer	B.Cu	.GBR	.GBL
Top Layer	F.Cu	.GBR	.GTL
Top Overlay	F.SilkS	.GBR	.GTO
Bottom Solder Resist	B.Mask	.GBR	.GBS
Top Solder Resist	F.Mask	.GBR	.GTS
Edges	Edge.Cuts	.GBR	.GM1

5.3 GerbView の使用

1. To view all your Gerber files go to the KiCad project manager and click on the *GerbView* icon. On the drop-down menu or in the Layers manager select *Graphic layer 1*. Click on **File** → **Open Gerber file(s)** or click on the icon . Select and open all generated Gerber files. Note how they all get displayed one on top of the other.
2. Open the drill files with **File** → **Open Excellon Drill File(s)**.
3. Use the Layers manager on the right to select/deselect which layer to show. Carefully inspect each layer before sending them for production.
4. The view works similarly to Pcbnew. Right click inside the view and click *Grid* to change the grid.

5.4 Freerouter による自動配線

Routing a board by hand is quick and fun, however, for a board with lots of components you might want to use an autorouter. Remember that you should first route critical traces by hand and then set the autorouter to do the boring bits. Its work will only account for the unrouted traces. The autorouter we will use here is FreeRouting.

注意

FreeRouting is an open source java application. Currently FreeRouting exists in several more or less identical copies which you can find by doing an internet search for "freerouting". It may be found in source only form or as a precompiled java package.

---

1. From *Pcbnew* click on **File** → **Export** → **Specctra DSN** and save the file locally. Launch FreeRouter and click on the *Open Your Own Design* button, browse for the *dsn* file and load it.
2. Freerouter は、手配線と自動配線の両方で、KiCad が現時点で持っていない特徴を持っています。Freerouter は 2 つの主要なステップで操作します: 最初に基板に配線を行い、次にそれを最適化します。完全最適化には時間がかかるかもしれませんが、いつでも必要なら停止することができます。
3. トップバーにある *Autorouter* ボタンをクリックすることで自動配線を開始できます。下部のバーは実行中の配線作業の情報を知らせてくれます。もしも *Pass* カウントが 30 以上になったら、あなたの基板はおそらくこのルーターでは自動配線できないでしょう。コンポーネント間隔をもっと広げたり良い向きに回転させて

から再挑戦しましょう。部品の回転と位置決めのゴールは、ラッツネストの空中交差の数を少なくすることです。

4. マウスの左クリックにより、自動配線を停止して最適化プロセスを自動的に開始させることができます。もう一度左クリックすると、最適化プロセスが停止します。停止する必要がないのであれば、Freerouter が仕事を終えるのに任せるのが良いでしょう。

5. **File** → **Export Specctra Session File** メニューをクリックして基板ファイルを *.ses* 拡張子で保存します。Freerouter ルールファイルは保存する必要がありません。

6. Back to *Pcbnew*. You can import your freshly routed board by clicking on **File** → **Import** → **Spectra Session** and selecting your *.ses* file.

If there is any routed trace that you do not like, you can delete it and re-route it again, using [Delete] and the routing tool, which is the *Route tracks* icon on the right toolbar.

Chapter 6

KiCad のフォワード・アノテーション

電子回路図、フットプリントの割り当て、基板レイアウト、ガーバーファイルの生成を完了したら、基板を現実のものとするために、PCB メーカーに全てを送る準備ができました。

しばしば、この線形の作業の流れは一方向ではなくなります。例えば、既にこの作業の流れを完了していたはずの基板を変更/拡張しなければならない時ですが、あなたは、部品を動かしたり、他のものと入れ替えたり、部品を変更したり、等々する必要が起こりえます。この変更の作業の中で、あなたがやりたくないことは基板全体を一から配線し直すことでしょう。代わりにこのようにすべきです:

1. 仮想的なコネクタの CON1 を CON2 と入れ替えたいと想定しましょう。
2. あなたは既に完成した回路図と配線を終えた PCB を持っています。
3. KiCad から *Eeschema* を起動して、CON1 を削除し CON2 を追加して変更を行います。アイコン で回路図プロジェクトを保存して、トップツールバーにある ネットリストの生成のアイコン NET をクリックします。
4. ネットリストをクリックし 保存します。デフォルトのファイル名で保存しましょう。古いファイルを上書きします。
5. CON2 にフットプリントを割り当てます。トップツールバーにある *CvPcb(*コンポーネントとフットプリントの関連付け*)* を実行のアイコン をクリックします。新しいデバイスの CON2 にフットプリントを割り当てます。残りのコンポーネントは以前に割り当てたフットプリントのままです。*CvPcb* を閉じます。
6. Eeschema に戻り、ファイル → 回路図プロジェクトの保存をクリックしてプロジェクトを保存します。Eeschema を閉じます。
7. KiCad プロジェクト・マネージャから *Pcbnew* のアイコンをクリックします。*Pcbnew* のウィンドウが開きます。
8. 古い、配線済みの基板が自動的に開きます。新しいネットリストファイルをインポートしましょう。トップツールバーにある ネットリストの読込みのアイコン NET をクリックします。

9. ネットリストファイルを参照するボタンをクリックして、ネットリストファイルを選択し、現在のネットリストを読み込むをクリックします。そして 閉じるボタンをクリックします。
10. この時点で、以前の全ての部品が配線済みのレイアウトが見られます。左上の隅に全ての未配線の部品、今回の例では CON2 が見えるでしょう。マウスで CON2 を選択します。基板の中央まで移動しましょう。
11. CON2 を配置して配線します。完了したらいつものように保存してガーバーファイルの生成をしましょう。

ここに書いた手順は必要に応じて何度でも容易に繰り返すことが可能です。上述したフォワード・アノテーションとは別に、バック・アノテーションとして知られる別の方法があります。この方法は Pcbnew で配線済みにした基盤の変更を、回路図とネットリストに反映することを可能にします。が、バック・アノテーションはそんなに便利ではないのでここでは書きません。

Chapter 7

Make schematic symbols in KiCad

Sometimes a symbol that you want to place on your schematic is not in a KiCad library. This is quite normal and there is no reason to worry. In this section we will see how a new schematic symbol can be quickly created with KiCad. Nevertheless, remember that you can always find KiCad components on the Internet.

In KiCad, a symbol is a piece of text that starts with *DEF* and ends with *ENDDEF*. One or more symbols are normally placed in a library file with the extension *.lib*. If you want to add symbols to a library file you can just use the cut and paste commands of a text editor.

7.1 コンポーネント・ライブラリ・エディタの使用

1. 新しいコンポーネントの作成には コンポーネント・ライブラリ・エディタ (*Eeschema* の一部) を使います。プロジェクトフォルダ *tutorial1* 内に *library* というフォルダを作りましょう。新コンポーネントを作ったら、そこに新しいライブラリファイル *myLib.lib* を置きます。

2. さあ新しいコンポーネントの作成を始めましょう。KiCad から *Eeschema* を起動して、コンポーネント・ライブラリ・エディタのアイコン をクリックし、新規コンポーネント作成のアイコン をクリックします。コンポーネント・プロパティのウィンドウが現れます。新しいコンポーネントを *MYCONN3* と名付け、デフォルトのリファレンス記号を *J* に、パッケージ内のユニット数を *1* に設定します。OK をクリックします。警告が出たら yes をクリックしておきます。この時点ではコンポーネントは、そのラベルだけで構成されています。ピンをいくつか足してみましょう。右ツールバーにある コンポーネントにピンを追加' のアイコン をクリックします。ピンを配置するには、シートの'*MYCONN3* ラベルの下あたりを左クリックします。

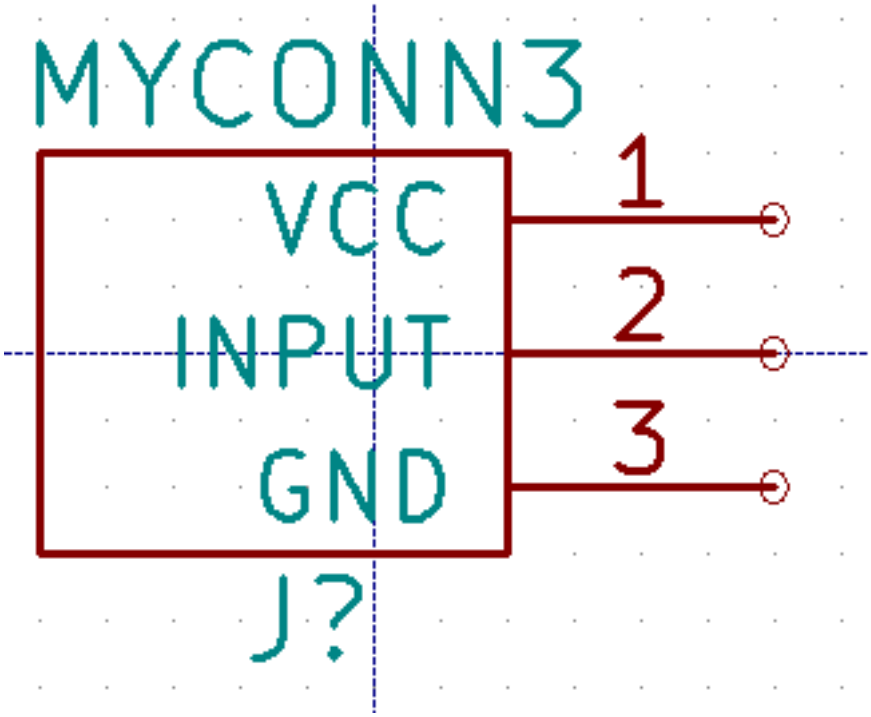
3. 現れた ピンのプロパティウィンドウで、ピン名を *VCC* 、ピン番号を *1* 、エレクトリック・タイプを 電源入力に設定して OK をクリックします。

4. 適当な場所、*MYCONN3* ラベルの右下あたり、をクリックしてピンを配置します。

5. ピン配置のステップを繰り返します。今度は ピン名は *INPUT* で、ピン番号は *2* で、エレクトリックタイプは パッシブとします。

6. ピン配置のステップを繰り返します。今度は ピン名は *GND* で、ピン番号は *3* で、エレクトリックタイプはパッシブとします。ピンを順に重ねて配置します。コンポーネントのラベルの *MYCONN3* をページの中心 (青いラインが交差する所) にします。

7. 次にコンポーネントの輪郭を描きましょう。コンポーネントのボディに矩形を入力のアイコン をクリックします。以下に示すようにピンに隣接して長方形を描きます。まず、長方形の左上の角にしたい所をクリックします（マウスボタンを押したままにしません)。そして、長方形の右下の角にしたい所をクリックします。



8. 矩形を黄色で塗りつぶしたい場合、まず 設定 → 色の設定で" ボディ背景色" を" 黄色 4" に設定します。そして、編集画面上で塗りつぶしたい矩形の枠上にマウスカーソルを置いて [e] を押して 図形のプロパティウィンドウを表示し、背景色で塗りつぶしを選択します。

9. コンポーネントをあなたのライブラリ *myLib.lib* に保存しましょう。新しいライブラリへ現在のコンポーネントを保存のアイコン をクリックして、フォルダ *tutorial1/library/* を選び、新しいライブラリファイルを *myLib.lib* という名前で保存します。

10. 設定 → コンポーネントライブラリで、*tutorial1/library/* を ユーザ定義の検索パスに追加し、*myLib.lib* を コンポーネントライブラリファイルに追加します。

11. 作業ライブラリの選択のアイコン をクリックします。ライブラリの選択ウィンドウ内で *myLib* を選択して OK をクリックします。ウィンドウの上部が現在使用中のライブラリを示しており、それが *myLib* であることに注意しましょう。

12. トップツールバーにある 現在のライブラリ内の現在のコンポーネントを更新のアイコン をクリックします。トップツールバーにある ディスクに現在のライブラリを保存のアイコン をクリックして変更を全て保存します。どの確認メッセージにも はいをクリックして下さい。新しい回路図コンポーネントは完成して、ウィンドウのタイトルバーに示されているライブラリから使えます。

13. コンポーネント・ライブラリ・エディタのウィンドウを閉じます。Eeschema のウィンドウに戻ります。あなたの新しいコンポーネントはライブラリ *myLib* から利用できます。

14. ライブラリパスに追加することによって、ライブラリの *file.lib* ファイルを利用できるようになります。*Eeschema* から 設定 → コンポーネントライブラリとして、それへのパスを ユーザ定義の検索パスに追加し、*file.lib* を コンポーネントライブラリファイルに追加します。

7.2 コンポーネントのエクスポート、インポート、変更

一からコンポーネントを作るよりも、既に作られたものから始めてそれを変更する方が簡単なこともあります。この章では、KiCad の標準ライブラリ *device* から、あなた独自のライブラリ *myOwnLib.lib* にコンポーネントをエクスポートして、それを変更する方法を見ていきましょう。

1. KiCad から *Eeschema* を起動して、コンポーネント・ライブラリ・エディタのアイコン をクリックして、作業ライブラリの選択のアイコン をクリックし、*device* ライブラリを選びます。現在のライブラリからエディタへコンポーネントを読み込むのアイコン をクリックして *RELAY_2RT* をインポートします。
2. コンポーネントのエクスポートのアイコン をクリックして、フォルダ *library/* へ行き、新しいライブラリファイルを *myOwnLib.lib* という名前で保存します。
3. ライブラリパスに追加することで、このコンポーネントとライブラリ全体 *myOwnLib.lib* が利用できるようになります。*Eeschema* から 設定 → コンポーネントライブラリとして、*library/* を ユーザ定義の検索パスに追加し、*myOwnLib.lib* を コンポーネントライブラリファイルに追加します。追加したら"OK" ボタンを押してウィンドウを閉じます。
4. 作業ライブラリの選択のアイコン をクリックします。ライブラリの選択ウィンドウ内で *myOwnLib* を選択して OK をクリックします。ウィンドウの上部が現在使用中のライブラリを示しており、それが *myOwnLib* であることに注意しましょう。
5. 現在のライブラリからエディタへコンポーネントを読み込むのアイコン をクリックして *RELAY_2RT* をインポートします。
6. これでコンポーネントを好きなように変更できます。マウスカーソルをラベル *RELAY_2RT* に重ねて [e] を押して *MY_RELAY_2RT* にリネームします。
7. トップツールバーにある 現在のライブラリ内の現在のコンポーネントを更新のアイコン をクリックします。トップツールバーにある ディスクに現在のライブラリを保存のアイコン をクリックして変更を全て保存します。

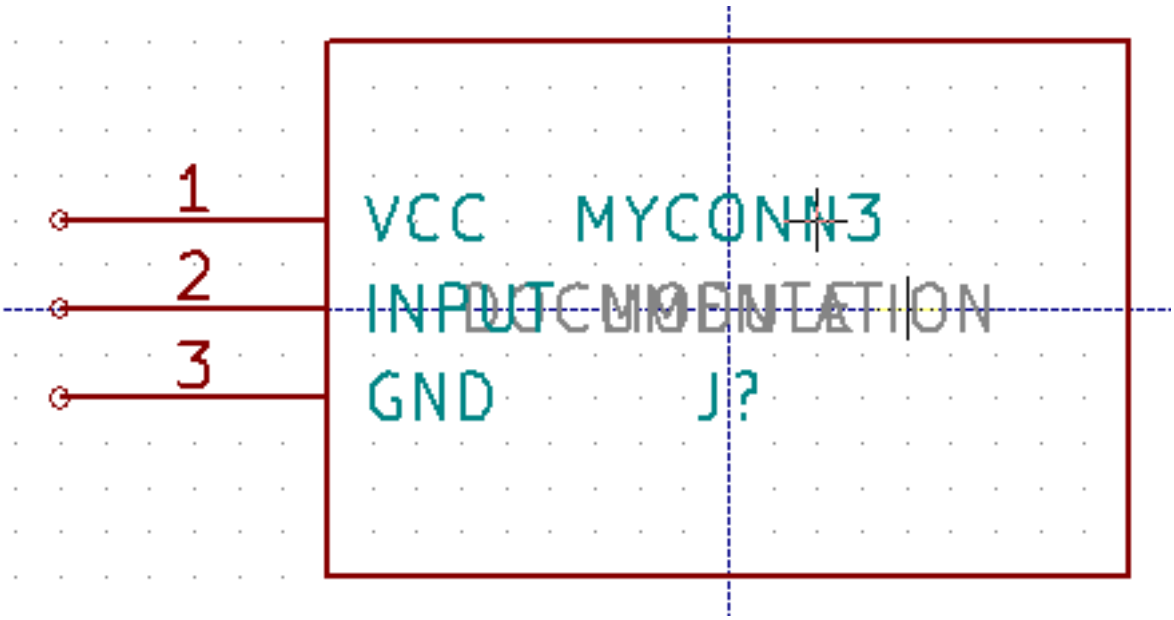
7.3 quicklib による回路図コンポーネントの作成

この章では、インターネットのツール *quicklib* を使った、回路図コンポーネント MYCONN3 (前の「MYCONN3」参照) の別の作成方法を紹介します。

1. *quicklib* のウェブページに行きましょう: http://kicad.rohrbacher.net/quicklib.php
2. ページに次の情報を入力しましょう: Component name: MYCONN3、Reference Prefix: J、Pin Layout Style: SIL、Pin Count, N: 3
3. *Assign Pins* をクリックします。ページに次の情報を入力しましょう: Pin 1: VCC、Pin 2: input、Pin 3: GND 。Type は 3 つのピンとも Passive を選択します。

4. *Preview* をクリックします。満足なら *Build Library Component* をクリックします。ファイルをダウンロードして *tutorial1/library/myQuickLib.lib* と名付けます。できました！

5. それを KiCad を使って見てみましょう。KiCad プロジェクト・マネージャから *Eeschema* を起動して、コンポーネント・ライブラリ・エディタのアイコン をクリックし、コンポーネントのインポートのアイコン をクリックし、*tutorial1/library/* へ行き *myQuickLib.lib* を選択します。



6. ライブラリパスに追加することで、このコンポーネントとライブラリ全体 *myQuickLib.lib* が利用できるようになります。*Eeschema* から 設定 → コンポーネントライブラリとして、*library/* を ユーザ定義の検索パスに追加し、*myQuickLib.lib* を コンポーネントライブラリファイルに追加します。

このコンポーネント作成方法は、たくさんのピンを持つコンポーネントを作りたい時に非常に効果的であると想像できるでしょう。

7.4 大量ピンの回路図コンポーネントの作成

quicklib による回路図コンポーネントの作成の章で、ウェブベースのツール *quicklib* を使った回路図コンポーネントの作成方法を見ました。しかし、時に、大量ピン (数百のピン) を持つ回路図コンポーネントを作る必要があることに気付くでしょう。KiCad では、これはそんなにややこしい仕事ではありません。

1. 50 ピンのデバイスのための回路図コンポーネントを作成したいとします。複数の少量ピンの図形、例えば 25 ピンの図形を 2 つ、を使ってそれを描くのが一般的な方法です。このコンポーネント表現はピン接続が容易でしょう。

2. このコンポーネントを作成する最良の方法は、*quicklib* を使って別々に 25 ピンのコンポーネントを生成して、Python スクリプトでそれらのピン番号を振り直し、最終的に 2 つを統合して一組の DEF と ENDDEF の中にコピー＆ペーストすることです。

3. 以下に *in.txt* ファイルと *out.txt* ファイルと連動して使えるシンプルな Python スクリプトの例を示します。これは *in.txt* ファイルの中の全ての行に対して、`X PIN1 1 -750 600 300 R 50 50 1 1 I` を `X PIN26 26 -750 600 300 R 50 50 1 1 I` のように数字の振り替えをします。

シンプルなスクリプト

```
#!/usr/bin/env python
''' simple script to manipulate KiCad component pins numbering'''
import sys, re
try:
    fin=open(sys.argv[1],'r')
    fout=open(sys.argv[2],'w')
except:
    print "oh, wrong use of this app, try:", sys.argv[0], "in.txt out.txt"
    sys.exit()
for ln in fin.readlines():
    obj=re.search("(X PIN)(\d*)(\s)(\d*)(\s.*)",ln)
if obj:
    num = int(obj.group(2))+25
    ln=obj.group(1) + str(num) + obj.group(3) + str(num) + obj.group(5) +'\n'
    fout.write(ln)
fin.close(); fout.close()
#
# for more info about regular expression syntax and KiCad component generation:
# http://gskinner.com/RegExr/
# http://kicad.rohrbacher.net/quicklib.php
```

1. 2 つのコンポーネントを一つに統合するために、Eeschema からコンポーネント・ライブラリ・エディタを使って、1 番目のコンポーネントを移動し、2 番目に重ならないようにしてやる必要があるでしょう。以下に最終的な.lib ファイルとその *Eeschema* での表現を示します。

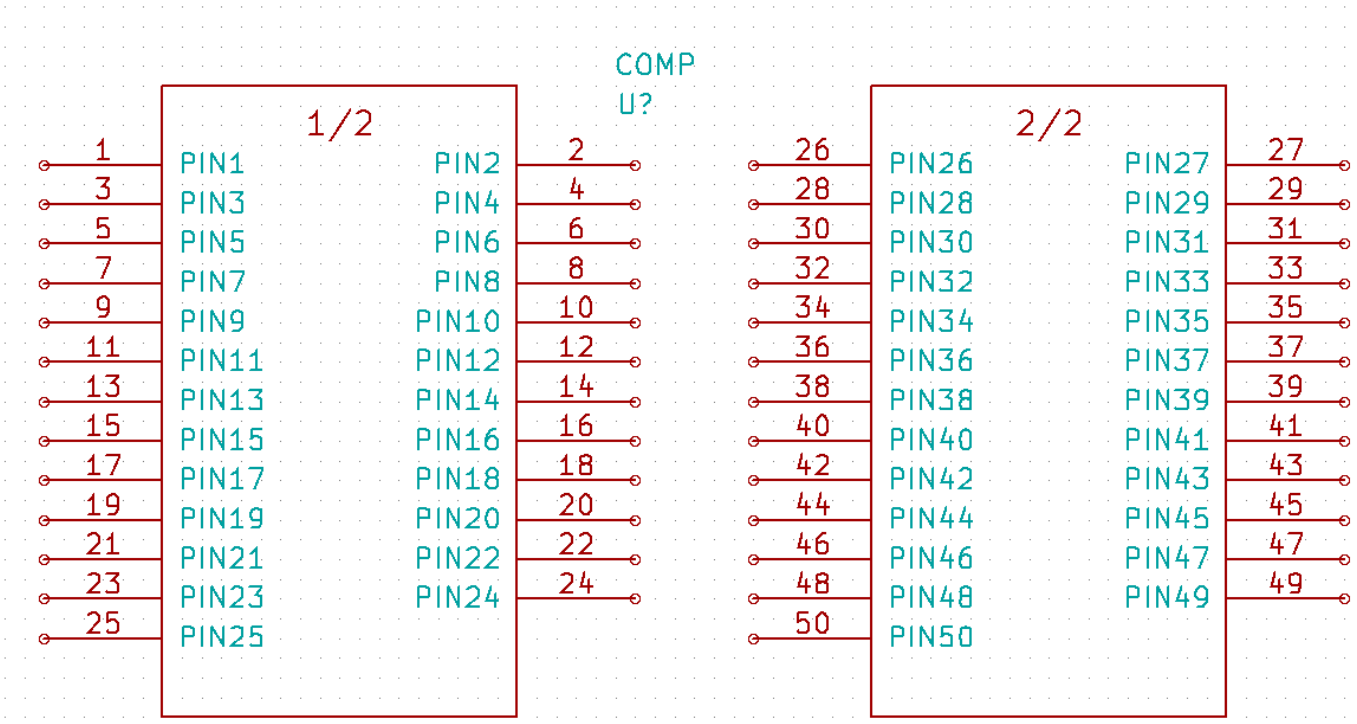
***.lib ファイルの内容**

```
Eeschema-LIBRARY Version 2.3
#encoding utf-8
# COMP
DEF COMP U 0 40 Y Y 1 F N
F0 "U" -1800 -100 50 H V C CNN
F1 "COMP" -1800 100 50 H V C CNN
DRAW
S -2250 -800 -1350 800 0 0 0 N
S -450 -800 450 800 0 0 0 N
X PIN1 1 -2550 600 300 R 50 50 1 1 I

...

X PIN49 49 750 -500 300 L 50 50 1 1 I
ENDDRAW
```

```
ENDDEF
#End Library
```



1. ここで紹介している Python スクリプトは、ピン番号とピンラベルを操作するとても強力なツールです。その威力は、難解ではありますが驚くほどに便利な正規表現構文による、ということを覚えておいて下さい: http://gskinner.com/RegExr/

Chapter 8

フットプリントの作成

他の、一種類のライブラリに回路図記号とフットプリントの両方を含む EDA ソフトウェア・ツールと異なり、KiCad の *.lib* ファイルは回路図記号を、*.kicad_mod* ファイルはフットプリントを含んでいます。*CvPcb* は回路図記号にフットプリントを割り当てるのに使われます。

.lib と同様に、*.kicad_mod* ファイルはテキストファイルで、一つのフットプリントを含むことができます。

KiCad には豊富なフットプリントのライブラリがありますが、あなたが必要とするフットプリントが KiCad のライブラリ内にないこともあるでしょう。ここでは KiCad で新しい PCB フットプリントを作成する手順を示します:

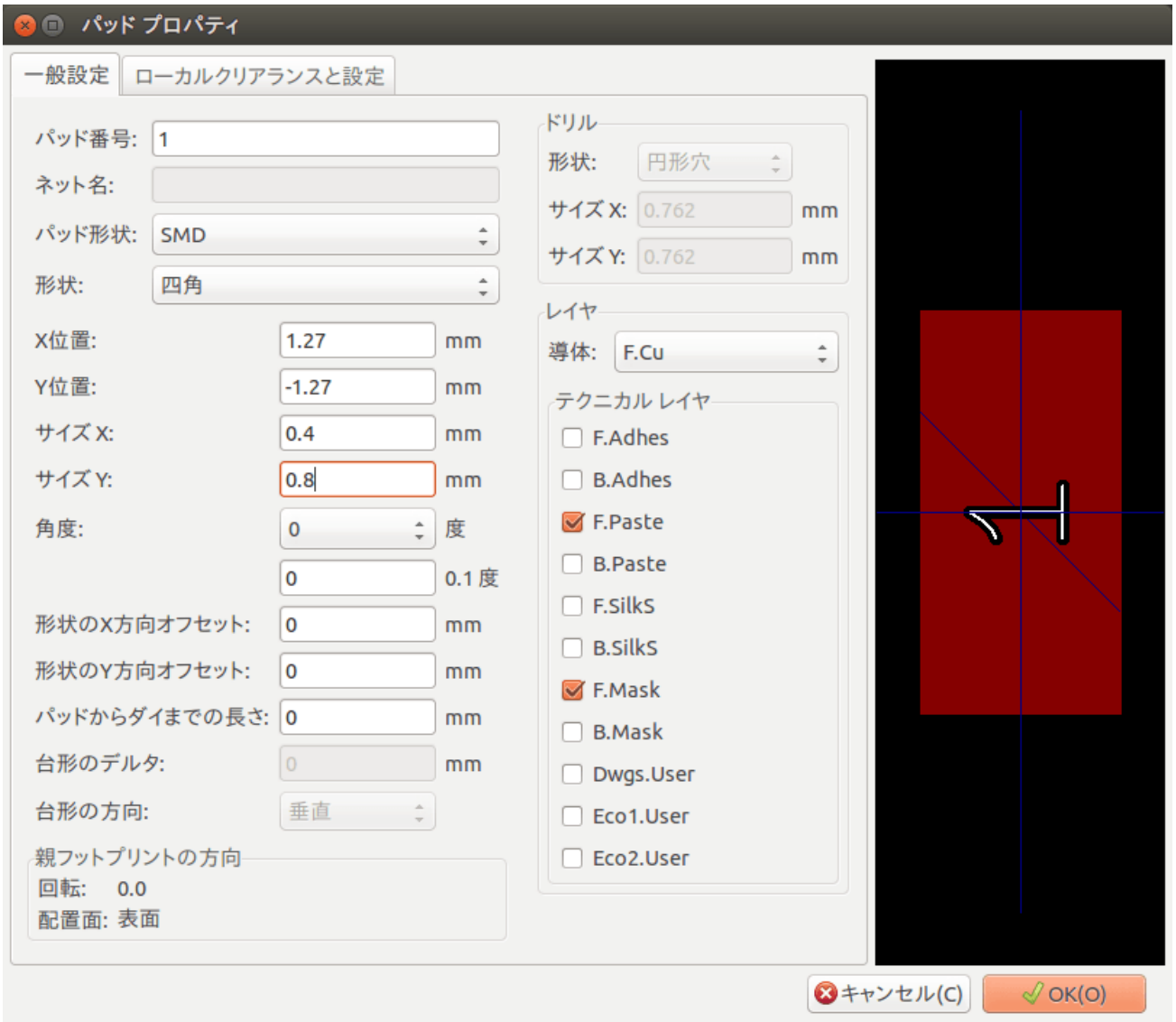
8.1 フットプリント・エディタの使用

1. KiCad プロジェクト・マネージャから *Pcbnew* ツールを起動します。トップツールバーにある フットプリントエディタを開くのアイコン をクリックします。フットプリント・エディタが開きます。

2. We are going to save the new footprint *MYCONN3* in the new footprint library *myfootprint*. Create a new folder *myfootprint.pretty* in the *tutorial1/* project folder. Click on the **Preferences** → **Footprint Libraries Manager** and press *Append Library* button. In the table, enter "myfootprint" as Nickname, enter "${KIPRJMOD}/myfootprint.pretty" as Library Path and enter "KiCad" as Plugin Type. Press OK to close the PCB Library Tables window. Click on the *Select active library* icon on the top toolbar. Select the *myfootprint* library.

3. トップツールバーにある 新規フットプリントのアイコン をクリックします。フットプリント名に *MYCONN3* を入力します。画面中央に *MYCONN3* ラベルが表示されます。そのラベルの下に *REF** があります。MYCONN3 の上で右クリックし、REF の上方に移動します。REF* 上で右クリックして、テキストの編集を選び、*SMD* にリネームします。テキストプロパティの 表示の項目で 非表示を選択します。

4. 右ツールバーにある パッドを追加' のアイコン を選択します。パッドを配置するため画面をクリックします。新しいパッド上で右クリックし' パッドを編集をクリックします。代わりに [e] も使えます。



5. パッド番号を *1* に、形状を 四角に、パッド形状を *SMD* に、サイズ *X* を *0.4* に、サイズ *Y* を *0.8* に設定し、OK をクリックします。再度 パッド入力をクリックし、もう 2 つパッドを配置します。

6. グリッドサイズを変更したいなら 右クリック → グリッドの選択とします。配置する前に適切なグリッドサイズを選択しましょう。

7. 上の図のように *MYCONN3* と *SMD* のラベルを外側に移動します。

8. パッドを配置する時には、相対的な距離を測る必要があります。相対座標軸の *(0,0)* としたい所にカーソルを置き、スペースキーを押します。カーソルを動かすと、カーソル位置の相対的な指標がウィンドウの下方に見えるでしょう。新しい原点を設定するにはスペースキーを押しましょう。

9. フットプリントの輪郭を加えましょう。右ツールバーにある 図形ライン（またはポリゴン）を入力のボタン をクリックします。パーツの周囲にコネクタの外形を描きます。

10. トップツールバーにある アクティブなライブラリへフットプリントを保存のアイコン をクリックし、デフォルト名の MYCONN3 で保存します。

Chapter 9

KiCad プロジェクトファイルの可搬性について

あなたの KiCad プロジェクトを、他の人が完全に取り込んで使えるために、どのファイルを送る必要があるでしょうか？

誰かと KiCad のプロジェクトを共有する時、回路図ファイル *.sch* 、基板ファイル *.kicad_pcb* 、プロジェクトファイル *.pro* 、ネットリストファイル *.net* を、コンポーネントライブラリファイル *.lib* 及びフットプリントライブラリファイル *.kicad_mod* と一緒に送ることが重要です。この方法でのみ完全に自由に回路図と基板を変更することができます。

KiCad の回路図では、回路図記号を含む *.lib* ファイルが必要です。それらのライブラリファイルは *Eeschema* の設定で読み込まれる必要があります。一方、基板 (*.kicad_pcb* ファイル) では、フットプリントは *.kicad_pcb* ファイルの中に保存させることができます。誰かに *.kicad_pcb* ファイルだけを送っても、彼はその基板を見たり編集したりできるわけです。しかし、もし彼がネットリストからコンポーネントを読み込みたいのであれば、フットプリントライブラリ (*.kicad_mod* ファイル) が存在し、回路図の場合と同様に、*Pcbnew* の設定で読み込まれている必要があります。同様に、*CvPcb* でモジュールが表示されるようにするためには、*Pcbnew* の設定で *.kicad_mod* ファイルが読み込まれていることが必要です。

もし誰かがフットプリント付きの *.kicad_pcb* ファイルを送ってくれたら、あなたはそれを他の基板でも利用したいでしょう。フットプリント・エディタを開き、現在の基板からフットプリントを読み込み、保存するか別のフットプリント・ライブラリ・ファイルにエクスポートします。*.kicad_pcb* ファイルから全てのフットプリントを一度にエクスポートすることもできます。**Pcbnew** → ファイル → フットプリントのアーカイブ → フットプリントアーカイブの作成で、基板上の全てのフットプリントの *.kicad_mod* ファイルが生成されます。

結論として、PCB だけを配布したいのなら、基板ファイル *.kicad_pcb* だけで十分です。しかし、あなたが他の人に、回路図と部品と基板の、利用と変更の完全な自由を与えたいのならば、以下のプロジェクトディレクトリを zip して送ることをお勧めいたします。

```
tutorial1/
|-- tutorial1.pro
|-- tutorial1.sch
|-- tutorial1.kicad_pcb
|-- tutorial1.net
```

```
|-- library/
|   |-- myLib.lib
|   |-- myOwnLib.lib
|   \-- myQuickLib.lib
|
|-- myfootprint.pretty/
|   \-- MYCONN3.kicad_mod
|
\-- gerber/
    |-- ...
    \-- ...
```

Chapter 10

KiCad ドキュメントの詳細

この文書は KiCad の大部分の機能のクイックガイドでした。より詳細な手順については、それぞれの KiCad ツールから参照できるヘルプファイルを参照して下さい。ヘルプ → マニュアルをクリックです。

KiCad には、その 4 つのソフトウェアコンポーネント全てに、とても良い多言語のマニュアルが付いてきます。

全 KiCad マニュアルの英語版は KiCad と一緒に配布されます。

マニュアルに加えて、KiCad はこの、他の言語にも翻訳されたチュートリアルと一緒に配布されます。このチュートリアルは無料で最新版の KiCad と共に配布されます。このチュートリアルとマニュアルはご利用の KiCad と共にパッケージされています。

Linux の場合、ディストリビューションにもよりますが、一般的に次のディレクトリでしょう:

```
/usr/share/doc/kicad/help/en/
/usr/local/share/doc/kicad/help/en
```

Windows の場合:

```
<installation directory>/share/doc/kicad/help/en
```

OS X の場合:

```
/Library/Application Support/kicad/help/en
```

10.1 Web 上の KiCad ドキュメント

最新の KiCad のドキュメンテーションは、複数の言語に翻訳されて Web 上にあります。

http://kicad-pcb.org/help/documentation/