

Le routage de circuit imprimé, introduction à *KiCad*

Table des matières

1.Introduction.....	1
1.1.Overview rapide de KiCad	2
2.Saisie de schéma sous KiCad : Eeschema.....	5
2.1.Quel symbole choisir ? Cas des transistors.....	5
2.2.Faire un bon placement des symboles	7
2.3.Relier les symboles entre eux.....	8
2.4.La numérotation des composants	9
2.5.La vérification : ERC (Electrical Rules Check).....	9
2.6.L'association d'empreintes.....	9
2.7.Génération de la netlist.....	11
3.Le routage.....	11
3.1.Réglages des paramètres du routeur.....	11
3.2.Délimitation du PCB.....	13
3.3.Importation de la netlist.....	13
3.4.Le placement des composants	14
3.5.Le routage des pistes.....	15
4.Création d'une nouvelle empreinte	18
5.Création d'un nouveau symbole.....	21

Auteur : Thierry Rocacher

1. Introduction

KiCad est un logiciel open-source destiné à la création de **PCB** (**P**rinted **C**ircuit **B**oard). D'apparence monolithique, *KiCad* est en réalité une suite de plusieurs logiciels spécifiques qui coopèrent :

- **KiCad**: Project manager
- **Eeschema**: Schematic editor and component editor
- **Pcbnew**: Circuit board layout editor and footprint editor
- **GerbView**: Gerber viewer

Bien que peu répandu dans le monde professionnel, ce logiciel s'est bien répandu dans la communauté des électroniciens. On trouve sur internet une foule de tutoriels en anglais comme en français. Parmi ses fonctionnalités :

- Taille de circuits non limitée
- multi-couche
- vue 3D
- création des fichiers pour fabrication industrielle professionnelle.

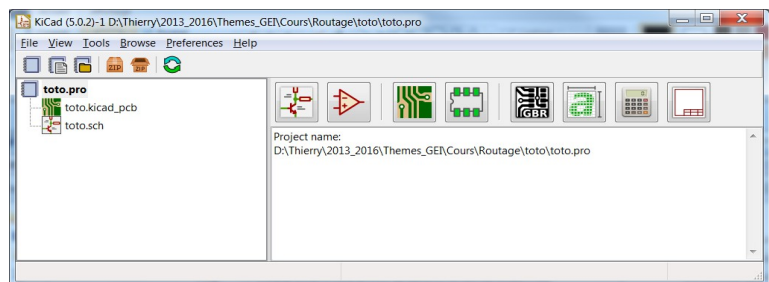
1.1. Overview rapide de *KiCad*

1.1.1. Création d'un projet *KiCad*

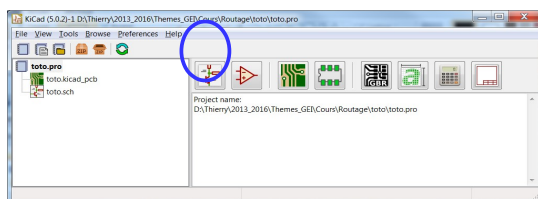
KiCad utilise la notion de projet (comme la majorité des logiciels de CAO et design électronique). Une session *KiCad* correspond à un seul et unique projet sauvegardé avec le suffixe *.pro*.

Lors de la création d'un tel projet, exemple *toto.pro*, *KiCad* crée automatiquement deux fichiers :

- *toto.sch* (schéma électronique)
- *toto.kicad_pcb* (le futur PCB)



1.1.2. Création du schéma électronique

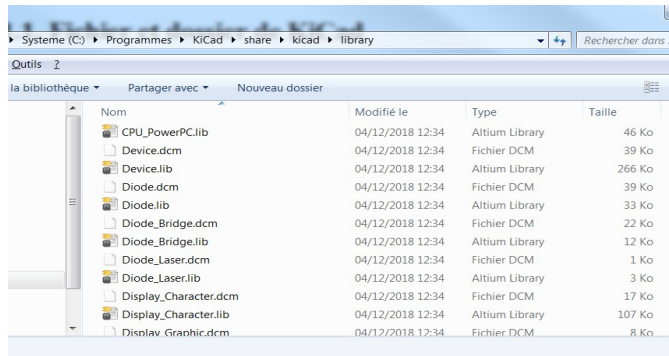


L'appui sur le bouton *Schematic Layout Editor* (ou un double clic sur le fichier *.sch*), lance le logiciel **Eeschema**.

Afin de construire son schéma, l'utilisateur se sert de **librairies de symboles**, *.lib*. Une documentation peut accompagner chaque librairie de symbole, ce sont des fichiers *.dcm*.

De très nombreux symboles sont prédéfinis dans *KiCad*. Ils se trouvent stockés dans le répertoire :

C:\Program Files\KiCad\share\kicad\library



Il est parfois nécessaire de construire son propre symbole, ce sera détaillé dans le chapitre 5.

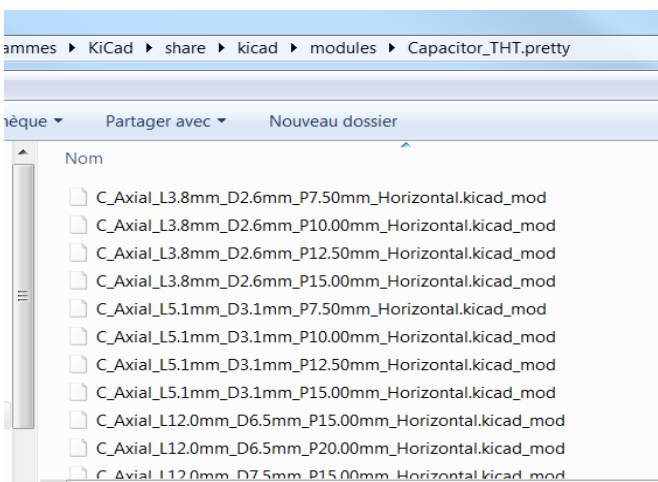
Un fichier *.lib* contient sous forme texte une **multitudes de symboles**.

Lorsque le schéma est terminé, on lui associe des **empreintes** pour pouvoir ensuite démarrer le routage du PCB.

Pour cela, *KiCad* utilise un second type de bibliothèque, il s'agit de **librairie d'empreintes** dont l'extension est *.kicad_mod*. Ces empreintes sont classées par thèmes et sont stockées dans des répertoires qui possèdent une extension *.pretty*. Un répertoire *.pretty* contient donc une multitude d'empreintes chacune décrite par un fichier propre *.kicad_mod*. Un répertoire *.pretty* correspond donc à une librairie d'empreintes.

Par défaut *KiCad* propose une foule d'empreintes dans le répertoire

C:\Program Files\KiCad\share\kicad\modules



A titre d'exemple, examinons les empreintes existantes pour un condensateur : on va ouvrir le répertoire *Capacitor_THT.pretty*.

Il est là encore parfois nécessaire de construire sa propre empreinte si elle n'existe pas par défaut (cf chapitre 4).

Lorsque chaque symbole est associé à une empreinte, l'utilisateur crée la **netlist**. Ce fichier est la conclusion de l'étape de saisie schématique et le tout début de l'étape de routage du PCB. Son extension : *.net*.

1.1.3. La logique des librairies sur KiCad

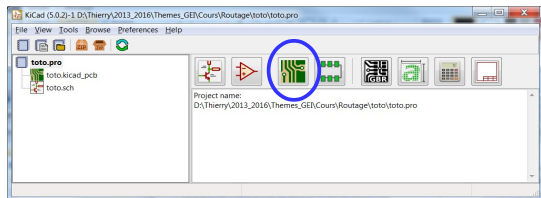
Certains logiciels de routage (*Eagle*, *Altium*) adoptent une logique consistant à créer un composant par un ensemble symbole / empreinte (*symbol/footprint*) indissociable.

Ainsi, autant de composants, autant d'empreintes associées.

KiCad fonctionne différemment. Les symboles et les empreintes sont **dissociées**, comme nous venons de le voir précédemment (*.lib* pour les symboles, *.pretty* et *.kicad_mod* pour les empreintes). L'avantage est alors de recenser une fois pour toute un maximum d'empreintes. Si un nouveau symbole est créé, pas besoin de refaire une empreinte qui existe déjà...

L'inconvénient est qu'à la fin de la saisie de schéma, il est nécessaire de faire une association symbole – empreinte.

1.1.4. Création du PCB



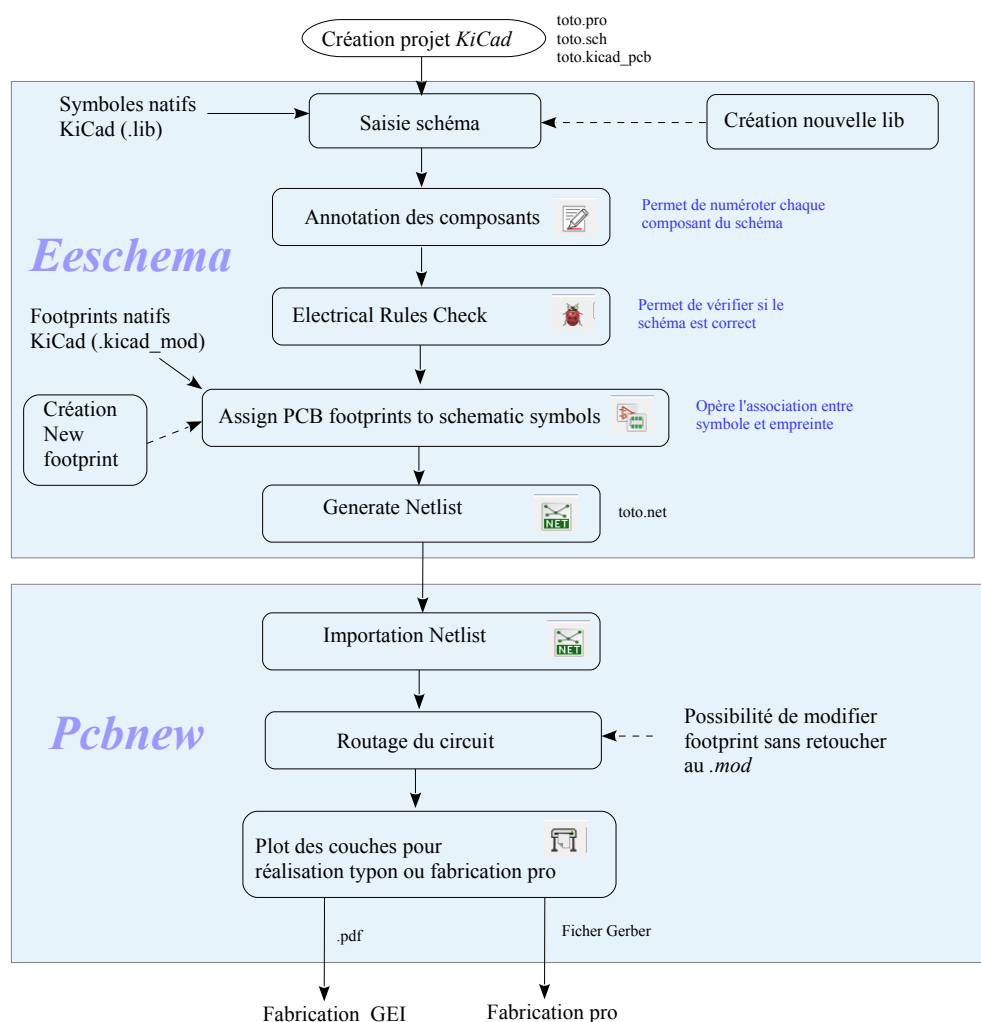
En cliquant sur le bouton *PCB Layout Editor* (ou en double cliquant sur le fichier *.kicad_pcb*), le logiciel *Pcbnew* est lancé.

Il faut alors importer la *Netlist* (produite lors de la saisie de schéma). Les composants (les empreintes ou *footprint* en anglais) sont posées dans un coin du PCB, tous reliés

via des fil jaunes que l'on appelle *airwire* (traduire chevelu en français).

Le but sera dans un premier temps de **placer** les composants avec pertinence, puis de les **relier** (routage) par des **pistes de cuivre**.

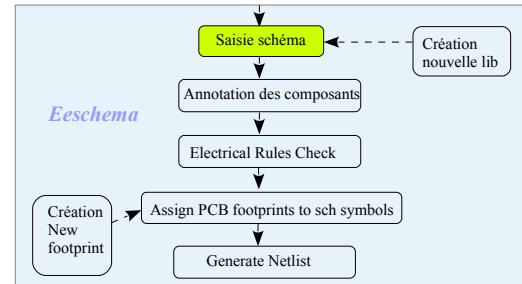
1.1.5. Flowchart pour la réalisation d'un PCB sur KiCad



Toutes ces étapes sont décrites ci-après.

2. Saisie de schéma sous *KiCad : Eeschema*

Pour démarrer ce chapitre, précisons d'entrée une stratégie pour bien zoomer et dézoomer, se déplacer facilement. **Tout se fait avec la molette centrale de la souris.** Survoler le centre à zoomer / dézoomer et jouer avec la molette. De la sorte vous pourrez vous déplacer rapidement où vous voulez sur le schéma.



2.1. Quel symbole choisir ? Cas des transistors...

Pour les composants usuels (résistances, condensateurs, diodes...) on se reportera à la librairie *Device*. Pour des composants spécifiques il faudra chercher en détail dans les autres bibliothèques.

Pour les transistors, les choses se compliquent un tout petit peu. Il en existe des milliers, et sous de nombreux boîtiers.

Il faut donc être très attentif au numéros des pin (1, 2, 3), à leur affectation sur le boîtier, et enfin à la correspondance E/B/C.

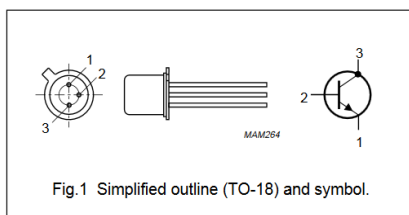
Nous allons prendre 4 exemples pour illustrer le propos.

2.1.1. Exemple 2N2222 et 2N2907

Voici les extraits de documentation :

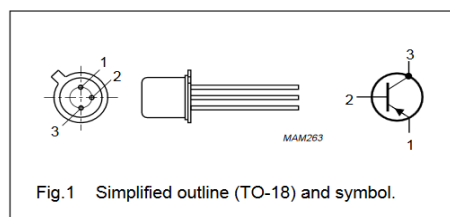
2N2222

PIN	DESCRIPTION
1	emitter
2	base
3	collector, connected to case




2N2907

PIN	DESCRIPTION
1	emitter
2	base
3	collector, connected to case



Le boîtier utilisé est un TO-18 dans les deux cas, avec la même correspondance E/B/C. Nous allons donc vérifier si ce boîtier, sous *KiCad*, possède bien la même numérotation (E/B/C = 1/2/3) :

Dans la barre d'outil horizontale cliquez sur  pour ouvrir l'éditeur d'empreinte (qui va nous permettre d'ouvrir les *.pretty* puis de voir les *.kicad_mod*) .

Ouvrez la bibliothèque *Package_TO_SOT_THT*, puis le footprint *TO-18-3*:

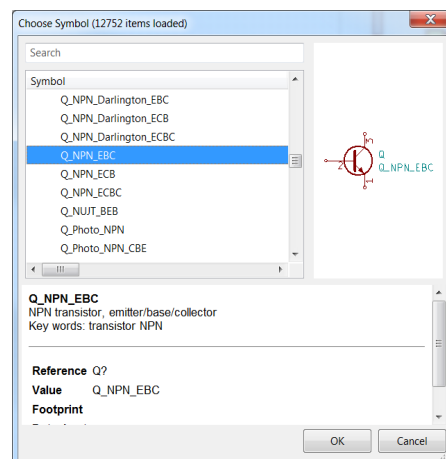
Nous voilà rassurés, la correspondance numéro/fonction est la bonne !



Il faudra donc utiliser, pour le 2N2222 le symbole NPN ayant la bonne numérotation :

On trouve toutes les combinaisons voulues dans la librairie *Device*, choisir le bon symbole et ne pas se tromper !!

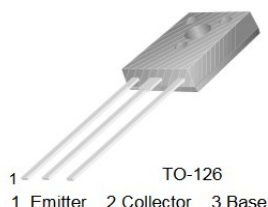
Place → *Symbol*. Librairie *Device*, symbole *Q_NPN_EBC* (on suit l'ordre naturel 1,2,3). Ensuite, modifier le nom dans le schéma en remplaçant *Q_NPN_EBC* par *2N2222*. L'association au boîtier TO-18-3 se fera lors de l'association *footprint* – *symbol*.



2.1.2. Exemple BD139 et BD138

Voici les extraits de documentation :

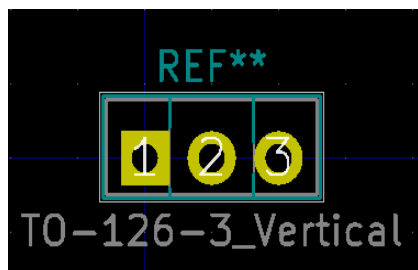
BD135/137/139



BD136/138/140



Ils sont là aussi encapsulés dans un même boîtier, le TO-126, avec la même affectation des broches. Nous vérifions que le package TO-126 de *KiCad* est bon :



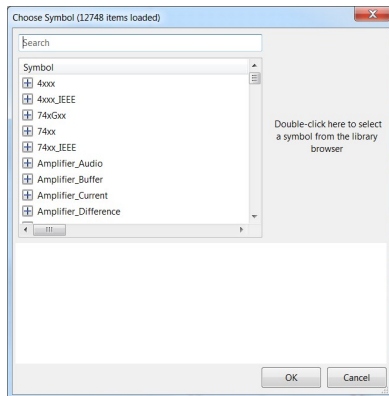
L'empreinte est bien conforme. Nous ferons donc attention, dans le schéma, à choisir le bon symbole (différent des 2N2222 et 2N2907) :

Par exemple, pour le PNP BD138 on choisira l'empreinte *Q_PNP_ECB*.

2.2. Faire un bon placement des symboles sur le schéma

2.2.1. Comment placer un symbole ?

L'essentiel du travail dans l'élaboration d'un schéma est le placement des symboles. Pour placer un symbole, aller dans le menu *Place* → *Symbol*, puis cliquer à l'endroit où vous souhaitez placer le composant :



Choisir la bibliothèque, puis le symbole dans la bibliothèque, ou taper directement le nom du composant dans le champ *search*

Une fois placé, survoler le composant avec la souris puis :

- **<M>** pour déplacer le composant,
- **<R>** pour le faire tourner,
- **<X>** pour une symétrie horizontale,
- **<Y>** pour une symétrie verticale,
- **<C>** pour copier,
- **<Suppr>** pour le supprimer.

Notez qu'un clic droit permet d'avoir accès à tous les menus contextuels.

NB : La bibliothèque *Device* est une sorte de « fourre-tout » où l'on trouve toute sorte de composants usuels.

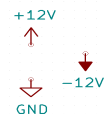
2.2.2. Quelle logique de placement ?

Il existe des règles d'usage à respecter pour une lisibilité claire et à destination de la plus large communauté de concepteurs en électronique :

- Placer les potentiels les plus haut en haut, les plus bas en bas,
- Séparer les arrivées d'alimentations du reste du circuit,
- Utiliser des symboles d'alimentation pour ne pas surcharger le schéma :

Ces symboles sont accessibles dans la librairie **power**.

- placer les entrées à gauche, les sorties à droite,



2.3. Relier les symboles entre eux

Maintenant que les composants sont placés, il faut les relier. Notez qu'on peut très bien placer qu'une partie des composants puis les relier, et continuer partie par partie. Rien n'impose de tout placer puis de tout relier. A vous de voir.

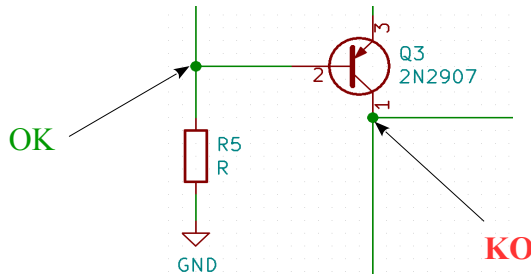
2.3.1. Comment relier ?

La commande se trouve dans le menu vertical de droite :  , ou simplement le raccourcis **<W>**.

2.3.2. Quelle logique pour relier ?

Quelques règles à respecter quand on relie les symboles :

- on utilise uniquement des angles droits, pas d'angle à 45°,
- une jonction en T ne se fait jamais sur une borne de composant :




- de la même manière, un croisement ne se fait JAMAIS au dessus d'un composant.

2.3.3. Le nommage des nets

Il est impératif de donner un nom à TOUS les fils (nets) de votre schéma. Lors du routage, ce nommage sera parfois extrêmement précieux. Lors des étapes de vérifications (*ERC* et *DRC*), le nommage aidera grandement pour trouver les points qui posent problème. Etape **indispensable** donc.

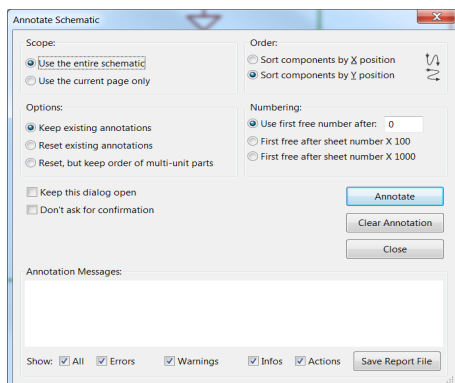
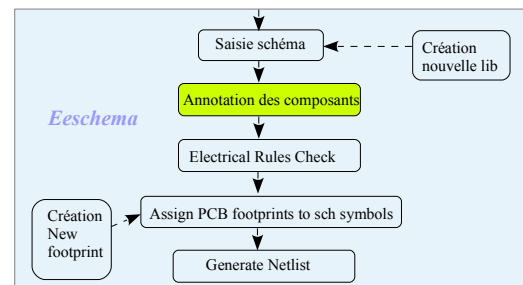
2.4. La numérotation des composants

Elle se fait par la commande .

La fenêtre suivante apparaît :


Une fois renseignée, cliquer que *Annotate*.

Les composants sont alors tous repérés.



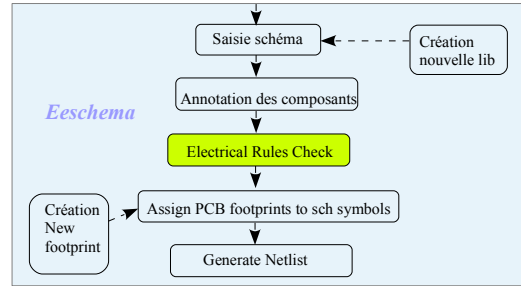
2.5. La vérification : *ERC* (*Electrical Rules Check*)

C'est une étape à ne pas oublier ! Elle vous permet de voir s'il y a des erreurs grossières dans votre saisie de schéma (patte de composant non reliée, court-circuit entre deux sorties, etc...).

Cliquer sur le bouton : 

Les messages donnés doivent tous être analysés. Il se peut que certaines erreurs n'en soient pas, et que vous assumiez ces messages.

Dès lors pensez à cliquer sur *Delete Markers* après avoir relancé un *ERC*.

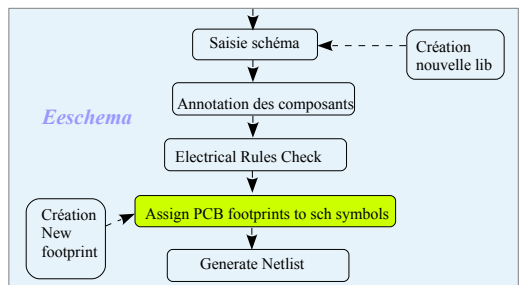


2.6. L'association d'empreintes

2.6.1. Mise en garde

C'est probablement l'étape la plus délicate de la saisie de schéma : bien choisir l'empreinte des composants. Il est primordial à ce stade d'être très clair sur la forme du composant :

- Sa taille : l'espace occupé sur le *PCB* mais aussi sa hauteur (si le circuit est contenu dans un boîtier, la hauteur est forcément limitée),
- l'espace entre pattes de connexion,
- le diamètre des pattes de connexion (une erreur à ce niveau est **catastrophique** si le circuit est réalisé de manière professionnelle avec des trous métallisés, le *PCB* d'un centaine d'euros est à jeter...),



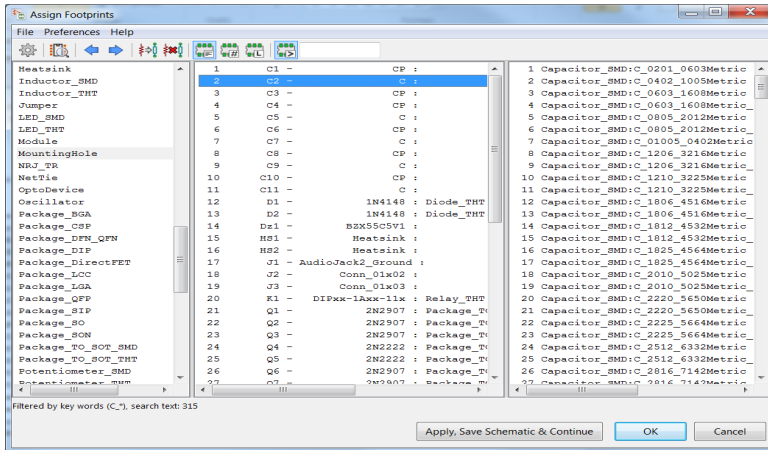
Mais aussi et surtout on portera une attention toute particulière à :

- l'affectation numéro de patte de l'empreinte ↔ pin sur la schématique. Une incohérence est là encore catastrophique.

2.6.2. Comment associer une empreinte à un symbole ?

Cliquez sur le bouton . Après le chargement des bibliothèques d'empreintes, la fenêtre suivante apparaît :

La colonne centrale donne la liste des symboles présents dans le schéma, associés ou non (après le ' :') à une empreinte.




La colonne de gauche liste les bibliothèques d'empreintes (les *.pretty*) tandis que la colonne de droite indique les diverses empreintes (*.kicad_mod*) disponibles dans la bibliothèque d'empreintes surlignée.

L'objectif de cette étape est d'effectuer toutes les associations.




Pour cela, vous pouvez vous servir des multiples filtres possibles :



Nous conseillons d'utiliser le filtre *filter footprints list from library* qui permet de faire apparaître les bibliothèques pour ensuite choisir les empreintes contenues.


Le bouton  est pratique puisqu'il permet de visualiser la forme de l'empreinte.

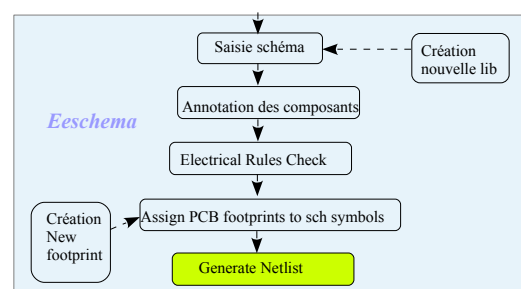
Pour autant, cette visualisation ne permet pas de mesurer le *footprint* précisément, ce qui empêche de valider complètement le choix.

Afin de pouvoir mesurer le composant, nous conseillons de revenir sur le schéma, et de venir cliquer sur l'icône  (*Footprint editor*). Là vous pourrez ouvrir l'empreinte souhaitée et vous pourrez faire des mesures précises :  , .

Lorsque tous les composants sont associés, procéder à une sauvegarde.

2.7. Génération de la netlist

Enfin, vous pouvez créer la *netlist* en appuyant sur  dans la barre de menu du schéma.



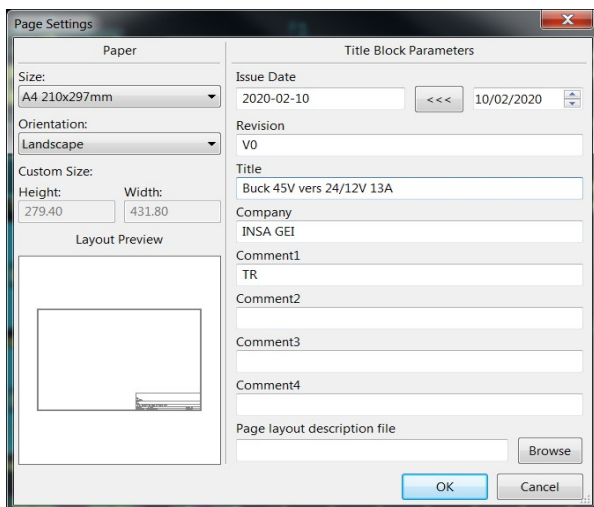
3. Le routage

Maintenant que la *netlist* est faite, vous allez pouvoir passer aux choses sérieuses, le routage. Avant de commencer l'opération, il est nécessaire de procéder à un certain nombre de réglages.

3.1. Réglages des paramètres du routeur

Paramétrons la page en premier :

File → *Page Settings*

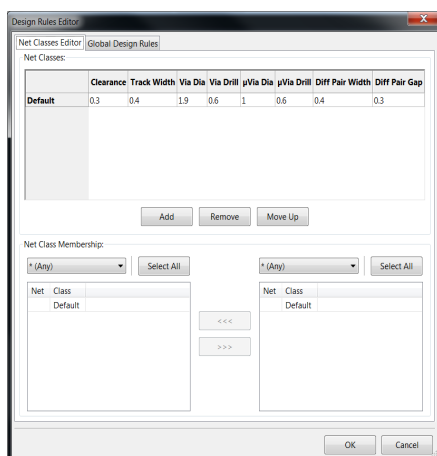


Les informations entrées dans ce cadre seront affichées dans le cartouche du document.

Remarque : A faire aussi côté schématique !

Il faut maintenant paramétrer le *DRC* (*Design Rules check*). Il doit être réglé pour correspondre au process de réalisation souhaité. Par exemple, si la réalisation se fait au GEI avec un process de type gravure au perchlore de fer, alors on se reportera au document [1] pour obtenir les paramètres.

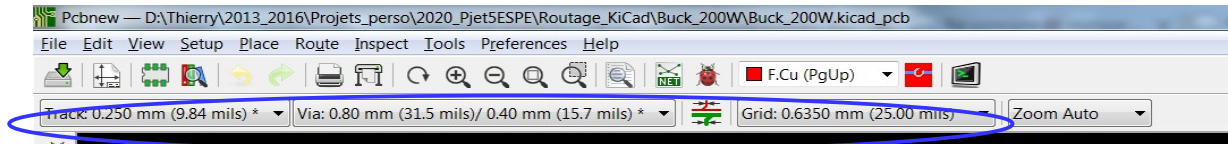
Setup → *Design Rules...* , onglet *Net Classes Editor*



La fenêtre a été remplie en fonction du process de fabrication (ici perchlore de fer). Le *DRC* suivra donc ces règles lorsqu'il procédera à la vérification. Il est donc **très important de bien remplir ce cadre**.

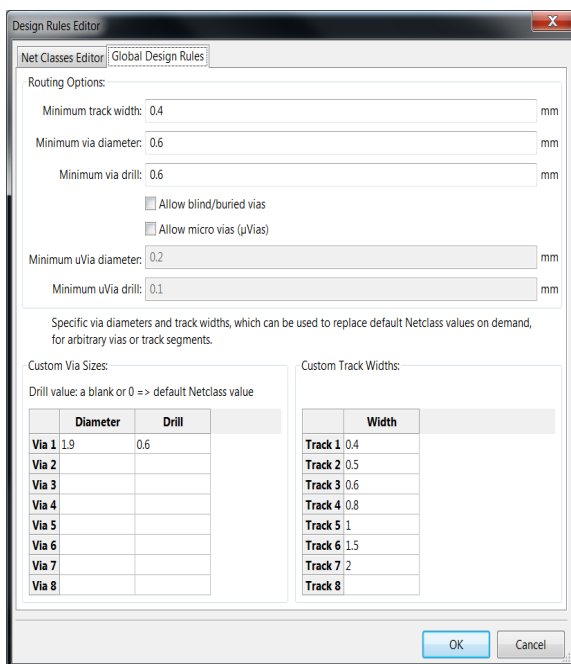
Lorsqu'on fait du routage, il est classique de modifier les tailles des pistes, le diamètre des via. La grille nécessite aussi d'être modifiée en cours de travail.

Ces 3 éléments sont facilement modifiables grâce aux onglets situés en haut, sous la barre des menus :



Par défaut, il n'y a qu'un choix possible. Afin d'élargir le choix des pistes et le diamètre des via, rendez-vous dans le menu :

Setup → *Design Rules...* , onglet *Global Design Rules*



Dans ce cadre vous pouvez rentrer des diamètres de vias différents mais aussi et surtout des largeurs de pistes différentes.

Vous disposerez de ces nouvelles géométries dans les menus déroulants présentés ci-dessus.


Maintenant que les réglages sont faits, on va pouvoir démarrer le routage en commençant par l'importation de la *netlist*.

Remarque :

Dans l'onglet vu précédemment *Net Classes Editor*, il est possible de construire plusieurs classes, au delà de la classe *Default*. Cela permet de classer les connectiques en plusieurs catégories ayant des propriétés différentes.

Par exemple on peut imaginer des pistes de puissance nécessairement plus larges que des pistes signaux.

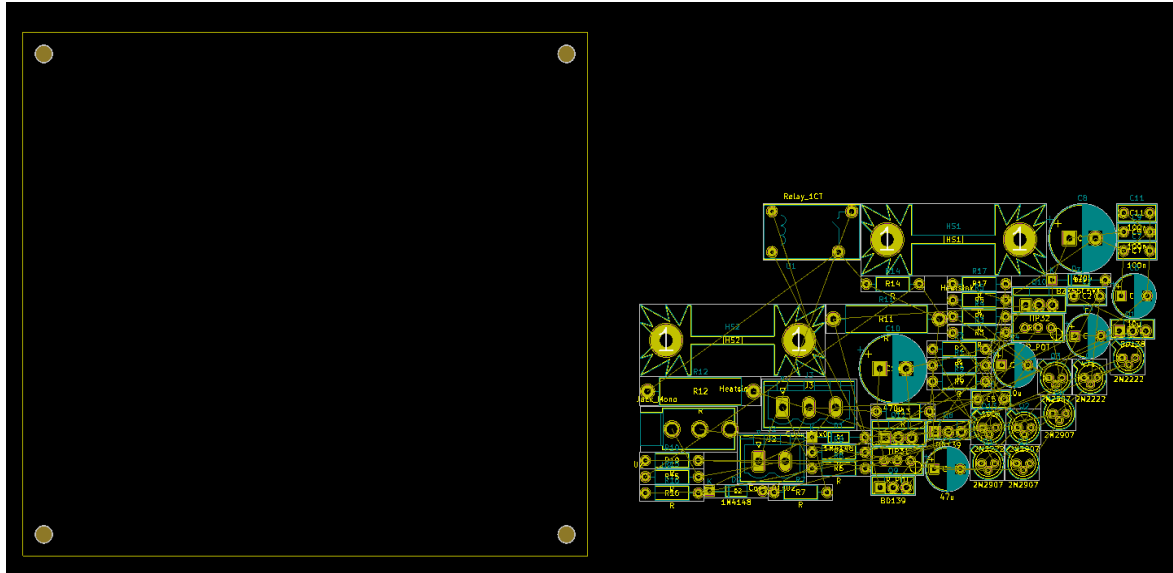
3.2. Délimitation du *PCB*

Vous pouvez créer des formes très variées de *PCB*. Utiliser la couche *Edge.Cuts*. Pour définir le contour du *PCB* puis les commandes de tracé : .

Usuellement, la forme est rectangulaire (mais peut être originale suivant le boîtier dans lequel le *PCB* sera embarqué), avec 4 points de fixation : 4 trous de typiquement de diamètre 3mm.

3.3. Importation de la netlist

Dans le menu horizontal *Tools* → *Load Netlist ...* Puis cliquer sur le bouton *Read Current Netlist*.



Les composants sont « jetés » dans un coin de l'écran, tous reliés les uns aux autres par des *air-wires* (chevelu) en jaune ici.

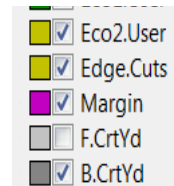
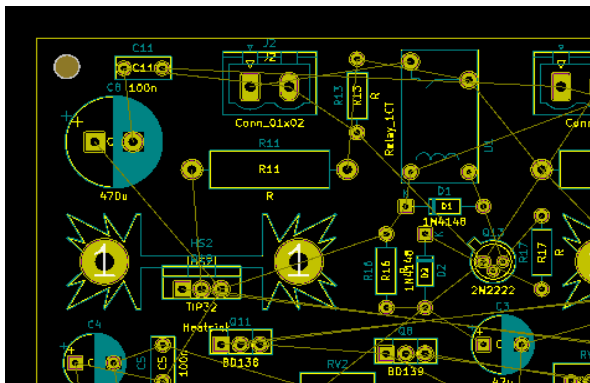
3.4. Le placement des composants

Cette étape est délicate et doit être mûrement réfléchie. D'une manière générale, il faut se donner comme méthode :

- placer en premier les connecteurs sur le *PCB* de manière réfléchie (dépend de la disposition du *PCB* dans le boîtier à l'intérieur duquel il sera logé). De la même manière, s'il existe des potentiomètres de commande (à ne pas confondre avec des résistances ajustables destinées à un réglage au tournevis en usine) , des LED ou autre équipement de façade, ces éléments doivent être placés en premier,
- respecter une logique de placement qui améliore la CEM [1],
- grouper les composants par fonction. Pour cela, travailler à partir du schéma avec le *PCB* actif en même temps. **Un clic sur un symbole du schéma provoque l'éclairage du *footprint* correspondant sur le *PCB*.** Ainsi on peut très facilement repérer les composants appartenant à une même fonction,
- tenter de minimiser les croisements de chevelu.

Conseils :

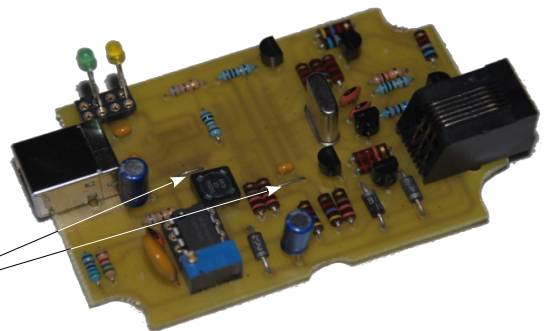
- Si le design se fait avec des composants traversants, une grille au pas 1.27mm (50mils) est conseillée. Cela correspond à un ½ pas normalisé (2.54mm – 100mils),
- Faire disparaître la couche *CrtYd* (décocher la case) de manière à ne pas surcharger le *PCB*.

**Exemple de placement****3.5. Le routage des pistes****3.5.1. Stratégie générale**

Maintenant que les composants sont mis en place, on va pouvoir les relier ensemble. Pour du routage simple face ou double face, la stratégie est la suivante :

- essayer de tout faire avec la couche bottom uniquement. Laisser les pistes impossibles (qui se croisent),
- si le nombre de pistes restées impossibles est raisonnable, (3 ou 4 par exemple), on peut insérer 3 ou 4 pistes en *top layer*, droite de longueur 10 mm maximum. Le circuit sera fait en **simple** face. Les rares régions en *top layer* seront matérialisés par des *straps* (des fils nus soudés à la place de la piste de cuivre),

straps



- si par contre, le nombre de *straps* dépasse 4, alors il convient de se résoudre à router en double face. Mieux vaut un double face routé avec une certaine cohérence, qu'un simple face dont certaines pistes font des « km » en zigzag ... Le choix simple ou double face n'est donc pas évident et se fait en cours d'opération,
- Si le choix du double face se fait, alors il faut prendre garde au vias (liaison ponctuelle entre des pistes *top* et *bottom*) : si la réalisation est industrielle, en trou-métallisé, on ne se pose pas trop de questions. Par contre si on fait du prototypage, souder des vias trop nombreux peut vite être fastidieux. Il est alors plus pertinent de profiter de composants traversants pour faire un via : on soudera la patte dessus et dessous,
- **ATTENTION : pas de routage en top vers des composants plaqués sur le PCB : la soudure est impossible !**

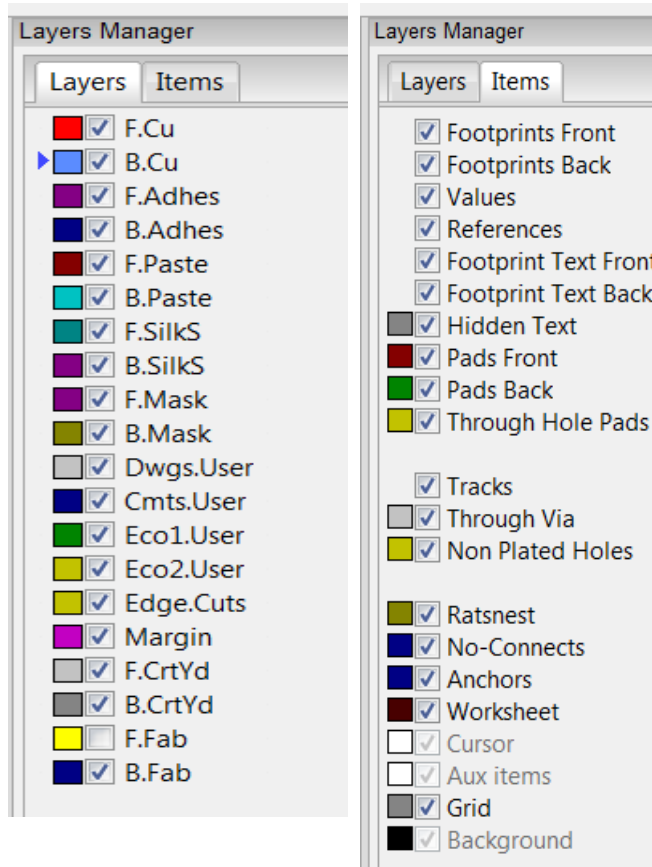
3.5.2. Les couches de PCB avec KiCad

Sur la droite, le *Layers Manager* contient deux volets : *Layers* et *Items*. L'utilisateur peut cocher (visible) / décocher (non visible) ce qu'il souhaite.

Les *Items* sont des objets. Choisir de cacher par exemple *Footprints Front*, masquera les composants en *top*, donc les couches qui composent les *Footprints*.

Voici un petit glossaire des couches sous KiCad :

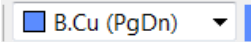
- **Adhesive** : couche qui permet de construire un masque (ou pochoir) qui permettra de déposer une colle pour fixer les composants *CMS*, pour soudage à la vague.
- **Solder Paste** : masque permettant de déposer la pâte à braser. Nécessaire pour les *CMS*.

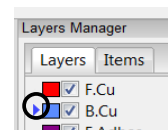


- **Silk Screen** : sérigraphie, ce qui sera imprimé sur le *PCB*,
- **Solder Mask** : masque permettant le dépôt du vernis de protection (laisse que les *pads* apparents pour pouvoir braser),
- **Courtyard** : utilisé pour montrer les limites du composant placé sur le *PCB*,
- **Fabrication** : ??,
- **Edge.cuts** : limites de découpe (frontières externes + trous ou autre fentes sur le board),
- **Comments, E.C.O.1, E.C.O.2, Drawings** : 4 couches pour le user permettant de donner des indications diverses. Non utilisées pour la fabrication.

3.5.3. Routage : How to ?

Router une piste : → commencer en **bottom (back)** puis **top (forward)** si nécessaire

- choisir la couche active  ou placer la flèche dans le *Layer Manager* devant la couche souhaitée,
- cliquer le bouton de routage de piste 



Dérouter une piste : survoler la piste et la supprimer <Suppr>. Le chevelu n'est pas effacé seul le cuivre l'est.

insérer un via : démarrer une piste, appuyer sur <v>. Le routage se prolonge avec la couche opposée.

Modifier une largeur de piste en cours de routage : cliquer sur <W> pour passer d'une largeur de piste à la suivante (parmi celles définies cf 3.1). Cliquer sur <Ctrl><W> pour diminuer la largeur.

Mettre tout une piste en surbrillance : cliquer sur le bouton



Mettre en surbrillance toute la couche active : cliquer sur le bouton



Déplacer une piste sans la casser (Drag) : sélectionner la piste puis <G>.

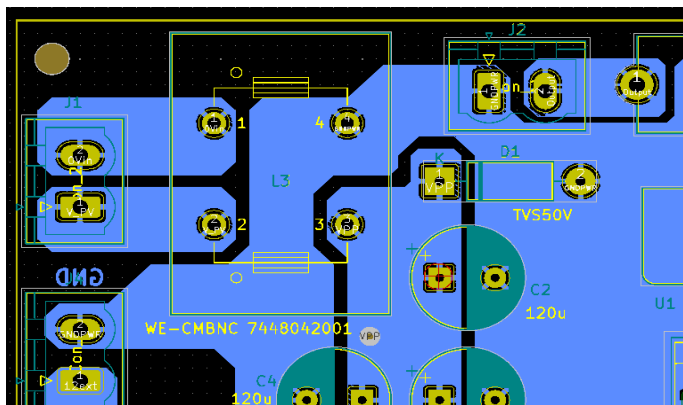
Déplacer une piste sans la casser et en conservant les angles (Drag) : sélectionner la piste puis <D>.

Modifier les pads d'un footprint : Soit on le fait dans la librairie (modification du *.kicad_mod*), soit on le fait directement sur le PCB (mais ce n'est pas rétro-annoté sur la librairie). Cette dernière solution est envisageable uniquement si seuls quelques composants sont concernés.

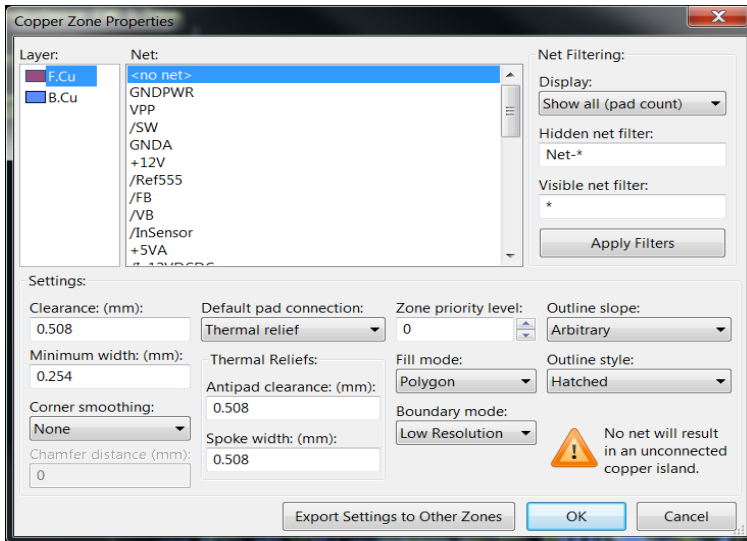
3.5.4. Les plans de cuivre

Il s'avère parfois utile de créer une surface (un polygone) de cuivre plutôt qu'une simple piste. Deux cas en particulier peuvent se présenter :

- besoin d'une grande largeur pour passer un fort courant,
- besoin de créer un plan de masse (réduit les boucles de masse et réduit la sensibilité aux champs électromagnétiques extérieurs).



Faire une surface de cuivre : Place → Zone



La fenêtre ci-contre s'affiche alors.

→ sélectionnez le net que vous voulez transformer en une zone de cuivre,

→ sélectionnez la bonne couche de cuivre,

→ sélectionnez l'isolation que voulez entre la surface et le reste des pistes différentes,

→ tracez le polygone.

Une fois tracé, la surface se remplira de telle sorte à ne pas court-circuiter de pistes différentes.

Pour plus de clarté par la suite on peut choisir de remplir ou non les zones de cuivre,

Afficher les polygones remplis :

Ne pas remplir les polygones :




NB : si des composants sont déplacés impliquant des pads dans une zone de cuivre, cette dernière est modifiée. Pour acter la modification, lancer un *DRC* (*DRC* puis bouton *Start DRC*).

NB : On peut dessiner un polygone qui en englobe un autre. Pour que *KiCad* ne fasse pas de court-circuit, il faut alors préciser la priorité de polygone (*Zone Priority Level*, cf fenêtre ci dessus) . Celui qui est intérieur devra avoir une priorité plus grande.

4. Création d'une nouvelle empreinte

Il est parfois impératif de créer une nouvelle empreinte. Pour cela, il est nécessaire de créer une nouvelle librairie contenant le *footprint* souhaité (il ne faut pas ajouter des empreintes à une librairie originale de *KiCad* sinon on perd la cohérence d'une installation à l'autre).

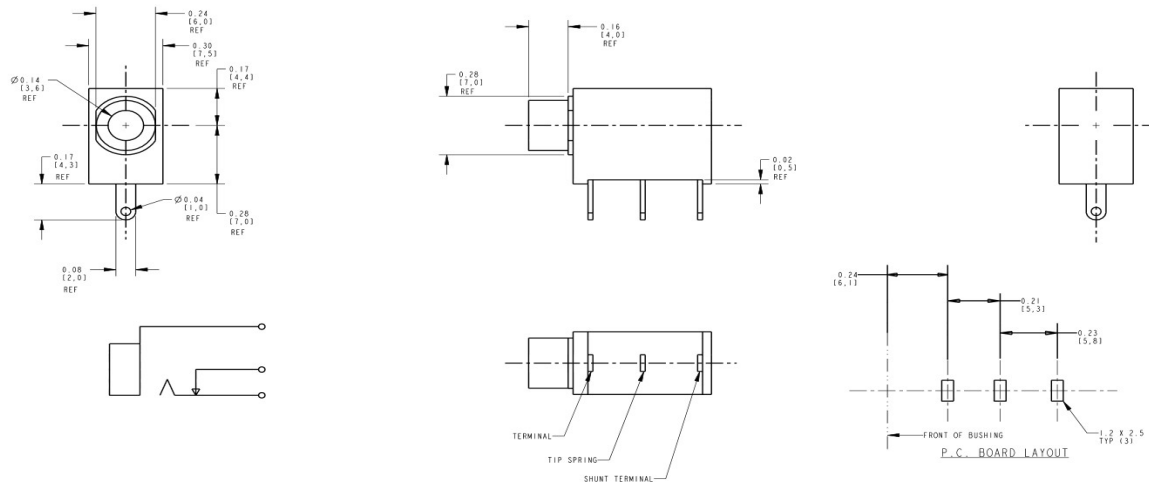
Lancer l'éditeur d'empreinte depuis le projet *KiCad* ou dans *Pcbnew*  .

A titre d'exemple, nous allons construire l'empreinte d'un socle Jack. On part donc de la spécification technique. Il s'agit de la référence *35RAPC2AV*.

Le composant ressemble à ceci :



Les dimensions sont les suivantes , *inches (mm)*:

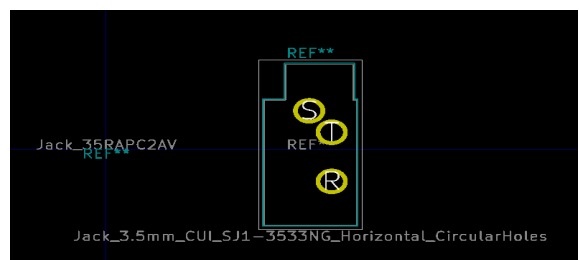


On peut démarrer le travail « from scratch » ou bien s'inspirer de l'existant. Nous proposons ici de partir depuis la librairie connecteur audio.

File → *Open Footprint*

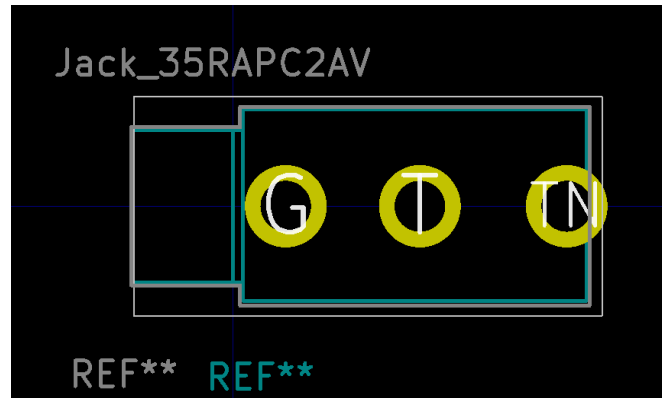
puis *Select Browser*. Sélectionner *Connector_Audio*, puis la seconde empreinte qui est assez ressemblante...

Le composant s'affiche. Sélectionnez-le et copiez le.



Ensuite, créez une nouvelle empreinte : *File* → *New footprint...*, puis collez. Vous devriez obtenir quelque chose comme ça :

Il faut effacer l'ancien texte, et modifier le schéma jusqu'à obtenir le symbole ci-contre.



On notera que les couches utilisées pour délimiter le composant sont :

- *F.Fab*
- *F.CrtYd*
- *F.Silks*

Ce sont des couches qui contribuent au dessin du composant. *F.CrtYd* (*Courtyard*) est la plus large. Elle délimite largement l'espace pris par le composant. Les deux autres couches collent plus au contour réel du composant.

Remarque : Les pastilles ont bien un perçage de 2mm. Les diamètres sont de 3.2mm. Enfin, les noms des contacts seront ceux à utiliser pour faire la correspondance entre symbole et footprint.

Sauvegarde empreinte et librairie

Pour sauvegarder l'empreinte, il faut d'abord créer une librairie, c'est à dire, créer un répertoire avec l'extension *.pretty*.

Cela se fait sous *KiCad*, pour cela cliquez sur . Choisissez un emplacement local. Il s'agit d'une bibliothèque personnelle. Une fois la librairie créée, l'empreinte y est sauvee automatiquement. Par exemple :

/MaLib_Ampli/Cmp_Ampli.pretty

Gestion des librairies personnelles (spécifiques)

Pour pouvoir utiliser la bibliothèque ou même retravailler le *footprint*, il faut indiquer à *KiCad* le chemin de cette nouvelle bibliothèque locale :

Preferences → *Manage Footprint Libraries*, puis onglet *Project Specific Libraries*. Enfin cliquez sur *Browse Libraries* et aller chercher le répertoire nouvellement créé *.pretty*.

Votre bibliothèque est maintenant visible comme toutes celles natives de *KiCad*.

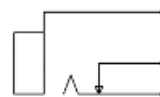
Effacer une empreinte dans une bibliothèque

Il se peut que vous souhaitiez effacer une empreinte dans votre bibliothèque. Cela ne se fait pas depuis *KiCad*. Il faut aller dans le répertoire *.pretty* correspondant à la bibliothèque puis effacer l'empreinte *.kicad_mod*.

5. Création d'un nouveau symbole

Tout comme pour les empreintes il est parfois nécessaire de produire un symbole qui n'existe pas. Pour illustrer le propos, nous allons poursuivre avec la création du symbole relatif à la prise jack vue dans le chapitre 4.


La forme du symbole visé est celui donné dans la datasheet :



Etant donné qu'il existe des formes similaires dans *KiCad* nous ne partirons pas de zéro.

Création d'une bibliothèque de symbole .lib

Lancez l'éditeur de symbole depuis le projet *KiCad* ou depuis le schéma : cliquez sur .

Cliquer alors sur  pour créer votre fichier .lib. On le placera dans le répertoire contenant le répertoire .pretty fait dans la partie 4. Ceci n'est pas une obligation, mais c'est logique !

Par exemple :

/MaLib_Ampli/Cmp_Ampli.lib

Lors de la création *KiCad* propose *Project* ou *Global*. Préférer *Project* vu que c'est une bibliothèque « locale ».

Récupération d'un symbole existant

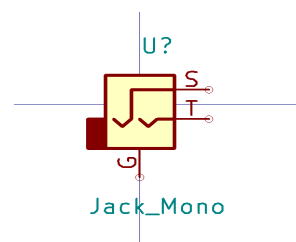
Dans la librairie *Connecteur*, le symbole *AudioJack2_Ground* ressemble bien à notre symbole. Ouvrez ce symbole dans l'éditeur.

Sélectionnez-le et copiez-le.

Sélectionnez votre librairie dans la colonne de gauche, clic droit puis créez un nouveau symbole qui vous appellerez par exemple *Jack_Mono*.

Collez le symbole.

Normalement vous devez obtenir quelque chose qui ressemble à ceci en guise de nouveau symbole dans votre nouvelle librairie locale :



Modification du symbole

Par défaut la grille est réglée à 50mils (1 mil = 1milli-inch= $2,54 \cdot 10^{-3}$ cm = 0,0254 mm), ce qui correspond donc à un pas de 1,27mm.

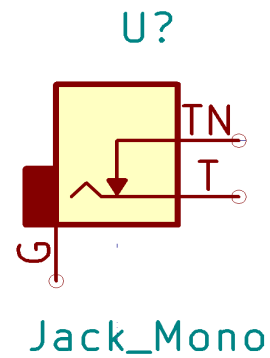
Il est fortement conseillé de conserver cette grille pour la disposition des pins (ici S, T et G). De cette manière, vous pourrez utiliser ce composant dans un schéma sans avoir besoin de réduire la grille.

Par contre, vous pouvez sans problème réduire la grille pour l'aspect dessin du symbole. C'est ici ce que nous allons faire pour obtenir la résolution des petits triangles du symbole (25mils donc).

Changer la grille : Preferences → General Options ...

La barre de dessin se trouve à droite, verticale.

Voici le symbole que l'on cherche à obtenir :



A Faire : les plots, prints , les graphiques

Références et liens intéressants

Aide KiCad accessible depuis le logiciel,

[1] TR : « La réalisation de circuits imprimés, Règles de routages pour débiter », PCB_FabricationGEI.pdf

[2] aide pour la création d'empreintes http://wikilab.myhumankit.org/index.php?title=Projets:Cr%C3%A9ation_de_Symboles_d%27Empreintes_et_Formes_3D_pour_Kicad#Cr.C3.A9ation_du_symbole_dans_Kicad

[3] Calcul de largeurs de pistes en fonction du courant : <https://www.4pcb.com/trace-width-calculator.html>