

Réalisation d'un PCB sous EAGLE V5.x

Sommaire

Réalisation d'un PCB sous EAGLE V5.x.....	1
1. Introduction.....	2
1.1 Basculement vers le board.....	3
2. Préparation du PCB.....	4
2.1 Positionnement des composants.....	4
2.2 Commande « DISPLAY » - Utilité des couches.....	6
2.3 DRC (default) et NET CLASS – Contraintes électriques.....	7
2.4 Le routage.....	8
2.5 Les options de "Design Rules".....	13
3. L'autorouteur.....	18

1. Introduction

Comme nous l'avons vu dans la réalisation d'un schéma, Eagle a pour philosophie de générer une couche pour chacune des caractéristiques d'un objet. Ainsi pour le symbole d'un composant dans un schéma, il existe au moins 4 couches indispensables : **Pins, Symbols, Names, Values** pour créer un symbole qui reste quelque chose de virtuel. Et pour le schéma, on rajoutera les couches **Net** et éventuellement **Busses**.

Quand il s'agit de créer un circuit imprimé, un PCB, on entre dans un espace réel à trois dimensions avec une multitude de caractéristiques qui n'ont parfois aucun lien avec l'électronique proprement dite. Avant tout, un composant physique aura évidemment un nom et une valeur, les mêmes que son symbole dans le schéma correspondant. Il occupera une **Place** sur le PCB.

Mais un composant aussi doit être soudé à ses points de contact (**Pad**), donc à chaque **Pin** du symbole dans le schéma correspondra au moins un **Pad** dans le board(pcb).

Un composant traversant (through hole) nécessite des trous (**Drill**) avec des pastilles (**Pad**) pour permettre leur soudage. Certains composants nécessitent une fixation par vis, il faudra donc des trous (**HOLE**) souvent sans pastille cuivrée; à l'inverse d'un composant monté en surface (SMD ou CMS) qui aura des pastilles non percées.

Le circuit imprimé aussi a des **dimensions** définies par le cahier des charges, un certain nombre de couches, il peut être simple face (**Top** ou **Bottom**), double-faces (**Top** et **Bottom**) ou multicouche. On passera électriquement d'une couche à l'autre par des **Via** qui sont des trous métallisés donc conducteurs.

Pour répondre à des contraintes électriques, physiques ou de sécurité, certaines zones du PCB seront interdites à tout composant (**Keepout**), parfois ce seront les pistes de cuivre qui seront interdites (**Restrict**).

Dans Eagle, chacune des caractéristiques reprises en gras ci-dessus génère au moins une couche comme par exemple **Top, Bottom, Drill, Hole, Dimension**.

D'autres en nécessitent parfois plusieurs, c'est le cas de

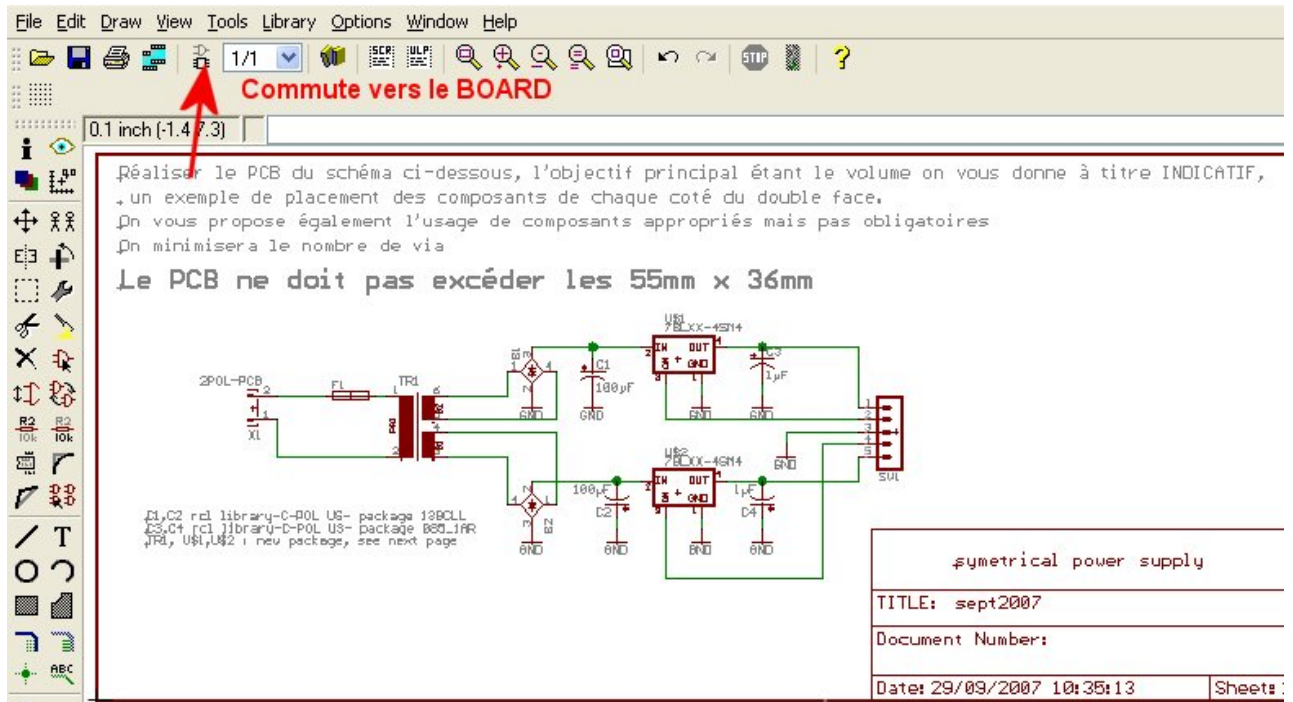
- Place qui donnera les 2 couches **tPlace** (Top Place) et **bPlace** (Bottom Place) tout comme Name, Value, Origine...
- Restrict en donnera 3 : **tRrestrict**, **bRrestrict**, et **vtRrestrict**.

Il existe encore d'autres couches prévues pour des spécifications de réalisation du PCB telles que l'étamage et le vernis; ou bien destinées au montage de la carte (colle et pâte à souder)

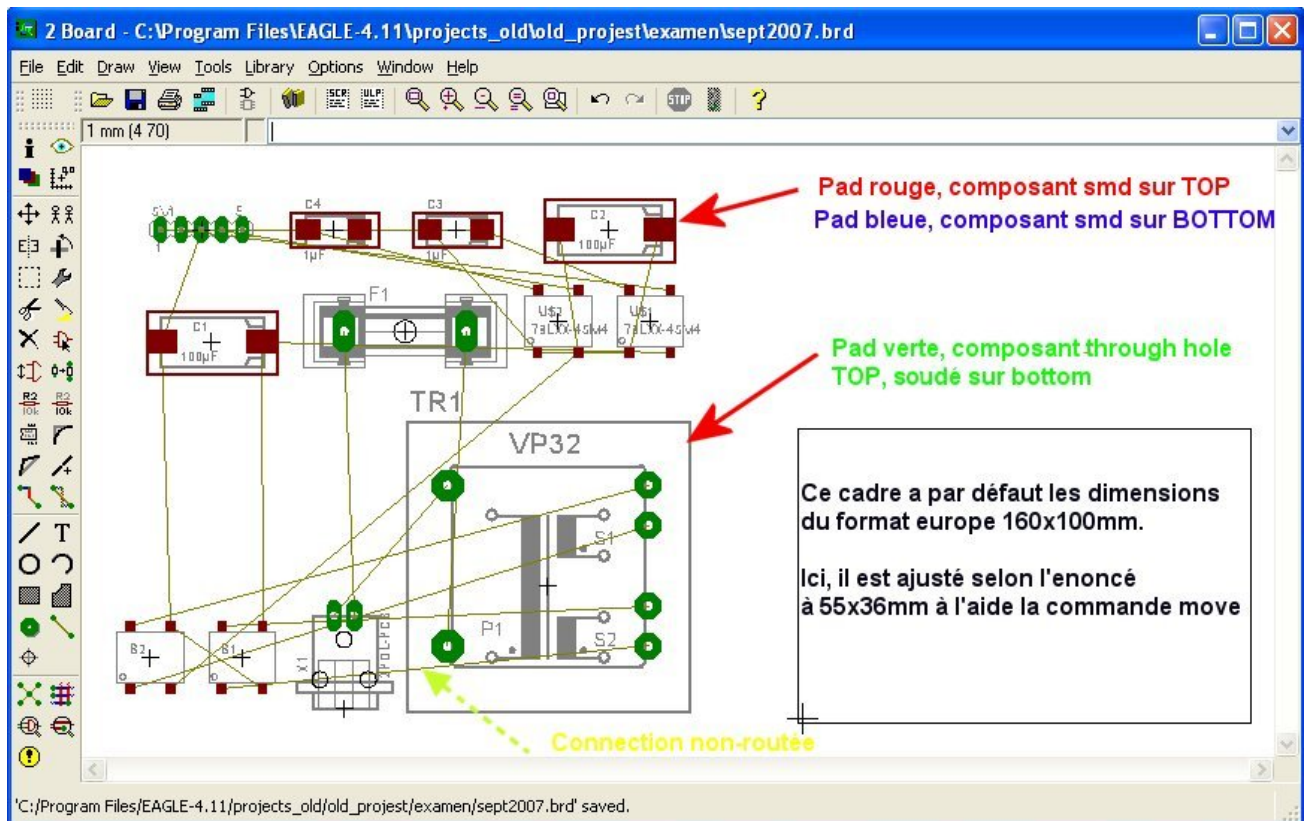
Enfin, nous verrons ultérieurement qu'il sera possible de définir ses propres couches

1.1 Basculement vers le board

Avant de poursuivre le tutorial schématique, nous partirons de l'exercice suivant dont le schéma est donné:



La commutation vers un « board » produit l'écran suivant :



Rem : ici, pour l'impression, l'arrière plan est mis en blanc. (Par défaut, il est noir).

Les traits en jaune représente les connexions, elles correspondent aux **NETS** du schéma.

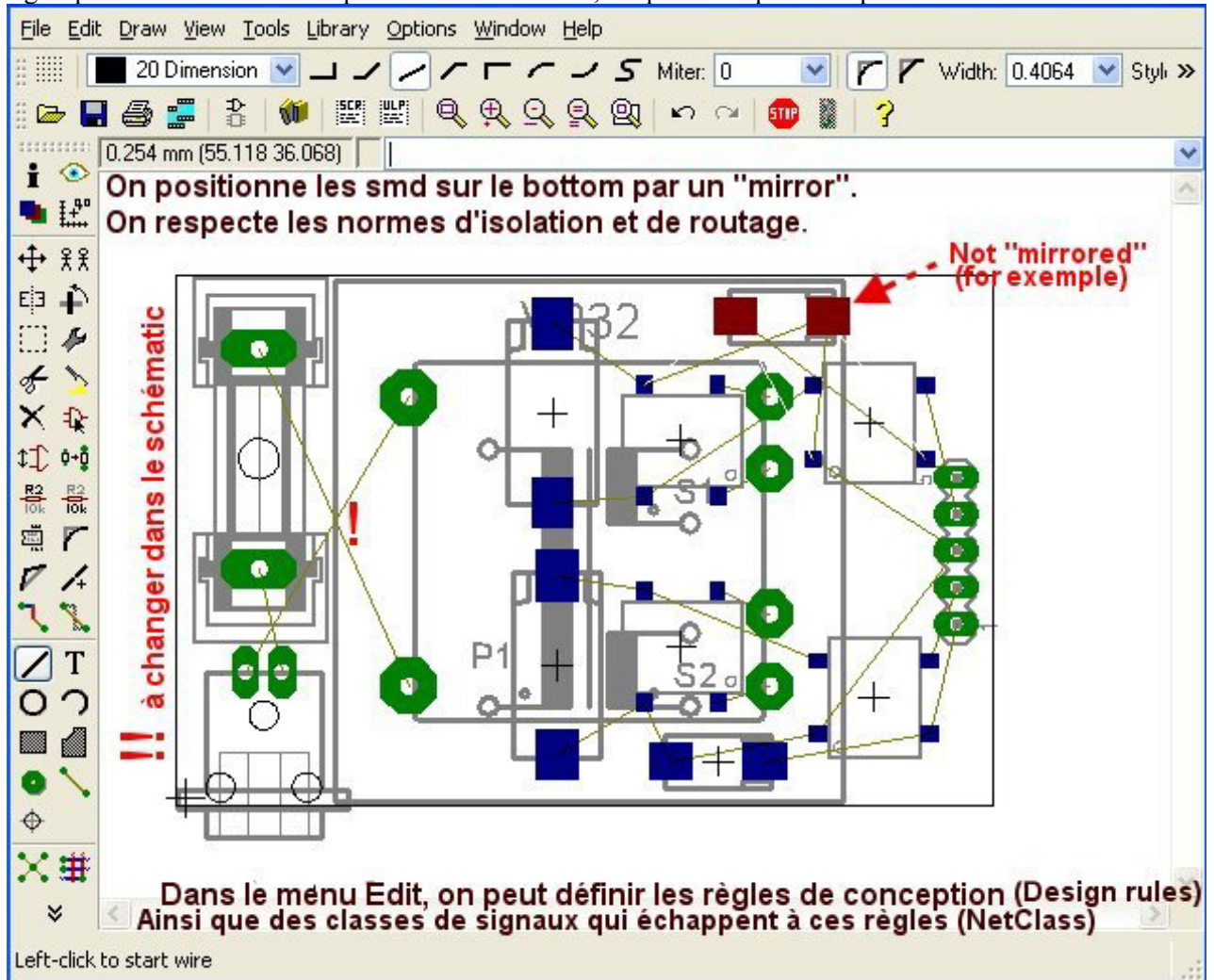
A ce stade, elles ne sont pas routées mais elles vont devenir les pistes du circuit imprimé.

Il faut avant tout positionner correctement les différents composants.

2. Préparation du PCB

2.1 Positionnement des composants

On tentera donc de les placer dans le cadre ajusté aux dimensions souvent imposées, et ce en regroupant ensembles les composants interconnectés, ce qui donne pour une première tentative:



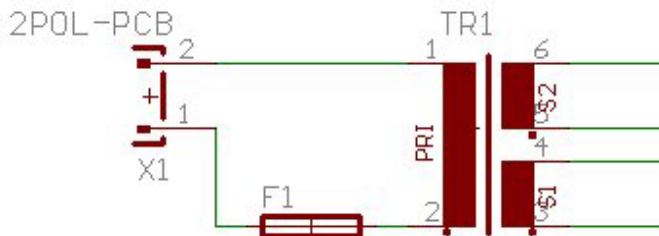
On réalise que le connecteur d'entrée 220V choisis ne répond pas aux normes d'isolation, on le changera en profitant de l'occasion pour éliminer le croisement des signaux (voir le manuel Library)

Comme on travail en simple face, on mettra les smds du coté soudure avec la cmd **Mirror**

On a aussi permuté les connexions du fusible dans le schéma.

La cmd PINSWAP, s'il elle avait été permise sur le fusible et sur le primaire du transfo, aurait donné le même résultat, mais plus rapidement(voir le manuel Library).

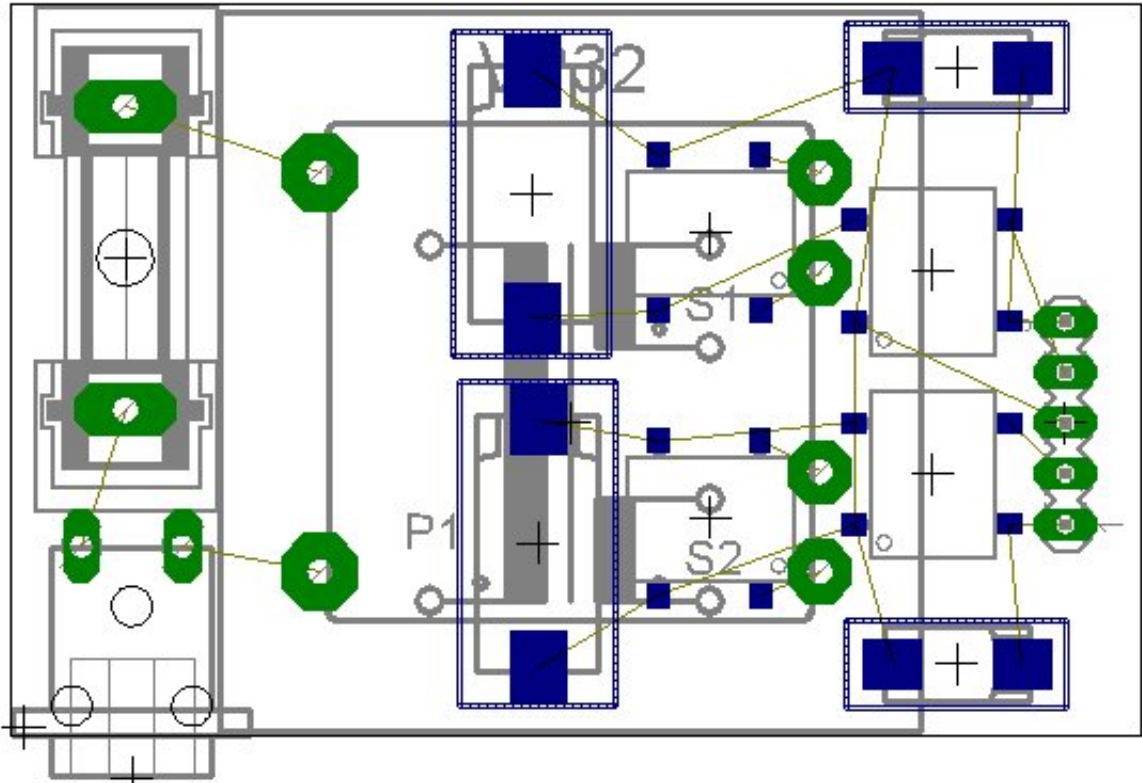
L'espace des pads du connecteur est également modifié dans la librairie, après vérification de l'existence du modèle. DRC et LIB seront étudiées ultérieurement. Après ces modifications, on constate que le board est déjà plus « lisible » et donc



vraisemblablement plus facile à router.

Eagle PCB

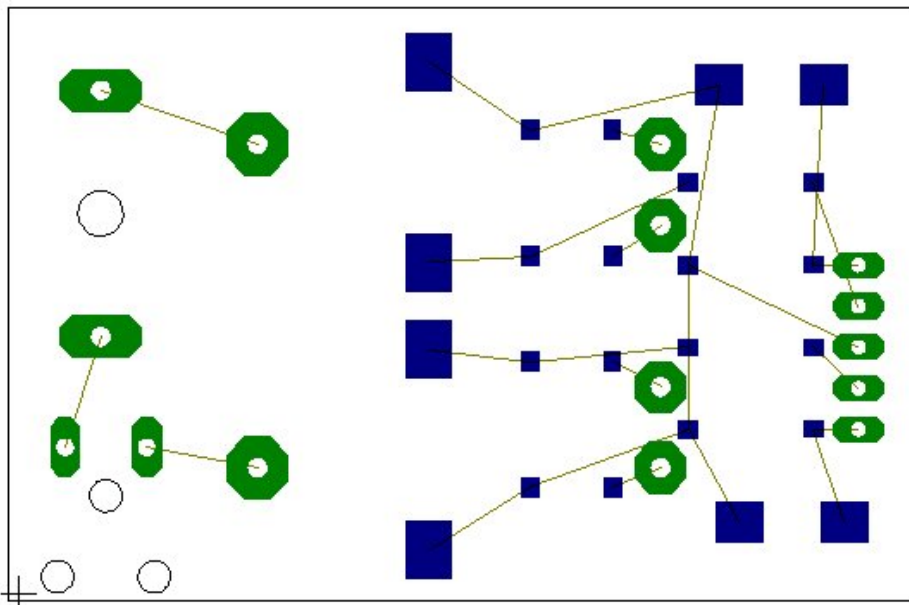
On obtient après ajustement des positions :



Pour procéder au routage de la carte, il y a deux méthodes : manuelle ou automatique.

Elles ne sont pas exclusives, on peut dans un premier temps router manuellement puis passer en mode automatique ou inversement. Un point commun, cependant, est d'avoir un écran lisible, c'est-à-dire ne comportant que les informations utiles au routage.

En désélectionnant les couches superflues (voir commande DISPLAY page suivante), on ne conserve que les couches TOP, BOTTOM, PAD UNROUTED et DIMENSION



Manifestement plus lisible. La commande **Ratsnet**  retrace les connections au plus court. Il est conseillé de l'appliquer avant de commencer le routage, afin de déceler des positionnements à rectifier. Il faut aussi vérifier les règles de routage (Design rules).

2.2 Commande « DISPLAY » - Utilité des couches

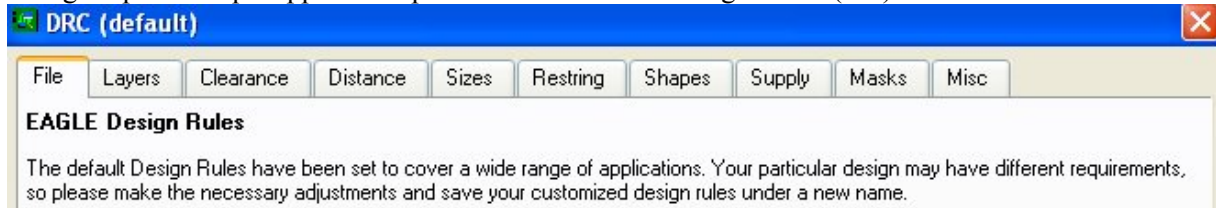
En général, on lira t... pour top, b... pour bottom, v... pour via
Ainsi tPlace=Place couche Top
Le contenu d'une couche désactivée n'est pas affiché, est inaccessible

Nr	Name	Description
1	Top	couche du dessus, côté composants classiques
2	Route2	couches internes réservées au multicouche
15	Route15	
16	Bottom	
17	Pads	pour les points de soudage des composants
18	Vias	pour les trous métallisés entre couches
19	Unrouted	affiche les connexions non-routées
20	Dimension	réservé au dessin de la forme de la carte, délimite l'époxy
21	tPlace	empreinte de la place des composants, face supérieure (Top)
22	bPlace	empreinte de la place des composants, côté soudure (Bottom)
23	tOrigins	couches origine du composant; si actives alors manipulation possible (move, rotate, mirror..) sinon composant inaccessible
24	bOrigins	
25	tNames	Contient le nom du composant, à sa création veiller à le créer avec le texte ">NAME" dans la couche appropriée (Top ou Bot)
26	bNames	
27	tValues	Contient la valeur du composant, à sa création veiller à le créer avec le texte ">VALUE" dans la couche appropriée (Top ou Bot)
28	bValues	
29	tStop	masques pour le soudage, générés automatiquement
30	bStop	
31	tCream	masques pour la pâte à souder, générés automatiquement
32	bCream	
33	tFinish	masques pour la pose de points de colle (souvent pour smd)
34	bFinish	
35	tGlue	réservée aux informations concernant les points de test et éventuellement les procédures de réglage
36	bGlue	
37	tTest	là on dessine les zones interdites aux emplacements des composants, couches testées par le DRC (design rules check)
38	bTest	
39	tKeepout	là on dessine les zones interdites au routage, l'autorouter en tient compte mais pas d'avertissement lors d'un routage manuel. Toutefois le DRC génère une erreur "restrict"
40	bKeepout	
41	tRestrict	
42	bRestrict	visualise les trous de perçage pour les Pad
43	vRestrict	
44	Drills	visualise les trous de perçage pour les vis de montage etc...
45	Holes	

Buttons: New, Change, Del, All, None, OK, Cancel

2.3 DRC (default) et NET CLASS – Contraintes électriques

Avant de procéder au routage (manu ou auto), il est préférable de définir les contraintes spécifiques à chaque signal. Le DRC (**D**esign **R**ules **C**heck) vérifie le respect de plusieurs paramètres. On le configure pour chaque application par le menu EDIT → Design Rules (DR).



Ces paramètres seront contrôlés, sur l'ensemble de la carte et des signaux :

Chaque onglet produit un écran illustré par des aides et des commentaires contextuels.

- **LAYERS** : Les couches utilisées, épaisseur de cuivre, espacement entre couches et leur association pour les vias
- **CLEARANCE** : Les distances d'isolation entre piste/pad/via/et pour les smd
- **DISTANCE** : entre pistes et trous avec les définitions de la couche dimension qui reprend les découpes de la carte
- **SIZES** : les dimensions minimales des pistes, trous et vias
- **RESTRING** : largeur de cuivre qui doit rester autour des trous, après forage
- **SHAPES** : définit la forme des PAD smd
- **SUPPLY** : utilisée pour le multicouche, définition des vias inter-couche
- **MASKS** : délimite les masques de soudure ou de pâte à souder
- **MISC** : précise le test sur la position des composants par rapport à la grille, l'angle des pistes, les fontes des textes, les zones restrictives pour les pistes, le nombre maximum d'erreurs qui arrête le DRC.

Dans le menu **Edit**→**Net classes**, on peut définir des classes de signaux

Pour chaque **Net Classes**, l'utilisateur définit, prioritairement aux règles par défaut :

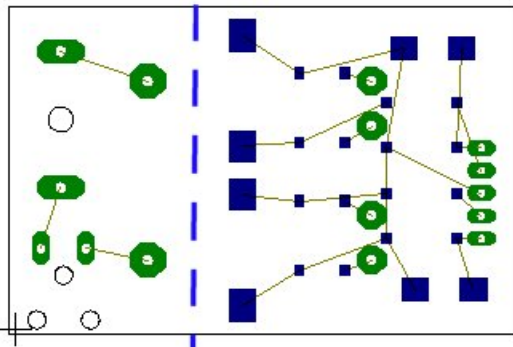
- les distances d'isolation
- la largeur minimum des pistes
- le diamètre minimum de perçage associé à cette classe

En général, pour les signaux de basse tension (<50V) les conditions d'isolation par défaut suffisent, mais pas la largeur de piste qui dépend du courant. L'usage des classes est indiqué pour grouper les connections soumises aux mêmes contraintes. Pour éviter un gaspillage de ressources, on ne fera pas un usage abusif de définition de classe. On peut définir un maximum de 7 classes, avec une matrice permettant de définir l'isolation à respecter entre classes

Dans notre application, nous aurions besoin de deux classes différentes :

la classe défaut (0) et la partie 'secteur 220V' qui requiert au minimum 3mm d'isolation(avec vernis).

Dans l'illustration ci-dessous, les signaux à gauche du pointillé seront assignés à la classe 1 avec 3mm d'isolation.



Et pour l'exemple, on définira une classe 2 avec une largeur de piste de 0,5mm pour les pistes de la sortie la basse tension vers le connecteur.

Nr	Name	Width	Clearance	Drill
<input type="radio"/>	default	0mil	0mil	0mil
<input checked="" type="radio"/>	220V	10mil	120mil	0mil
<input type="radio"/>	BT	20mil	8mil	0mil
<input type="radio"/>		0mil	0mil	0mil

Eagle PCB

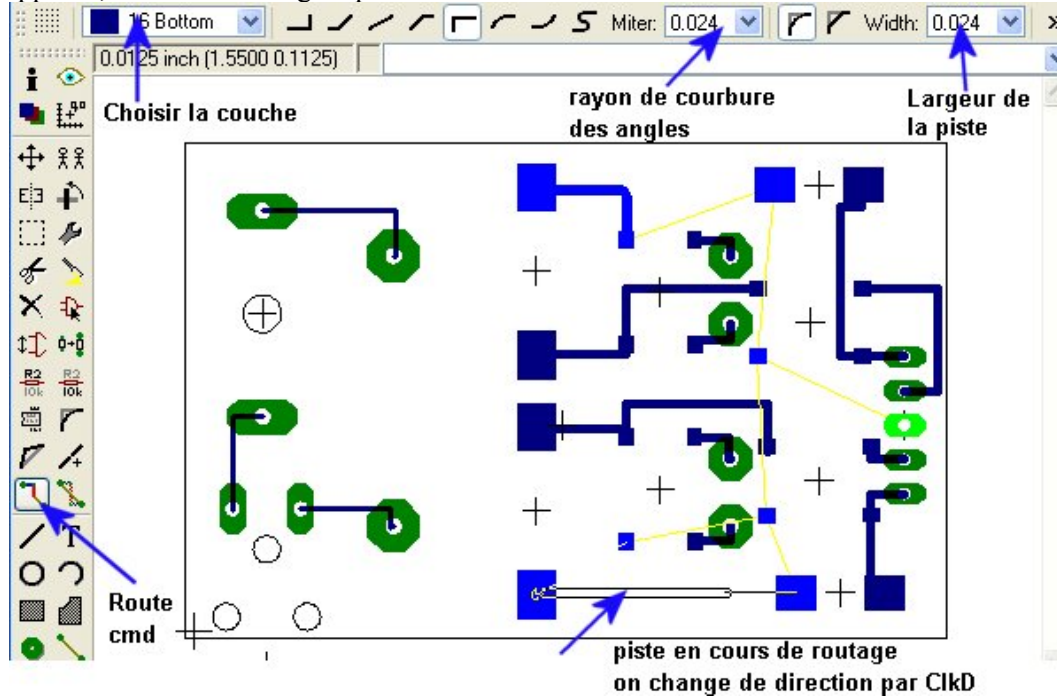
Une fois ces classes définies, on y associe les signaux par la commande CHANGE->CLASS puis en cliquant sur le signal pour l'assigner à la classe présélectionnée.

On peut aussi définir les classes avant de réaliser le schéma, et assigner sa classe au signal au moment de la création de son tracé. Mais ces spécifications n'influencent pas le routage manuel.

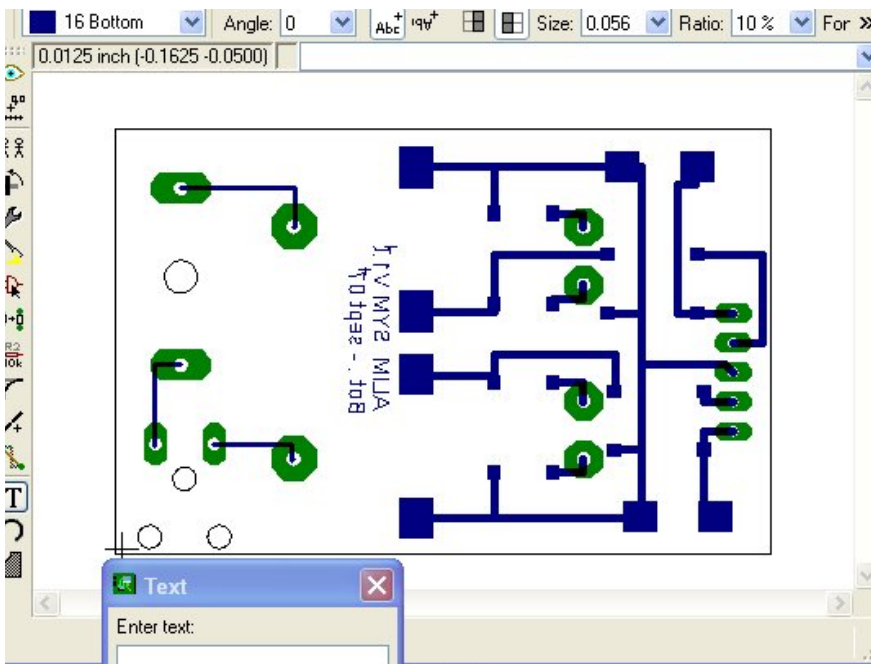
En d'autres termes, les 'design rules' et les 'Net classes' ne sont prises en compte que dans le DRC et par l'AUTOROUTER.

2.4 Le routage

Un routage manuel se réalise avec la cmd ROUTE , à son activation une nouvelle barre d'outils apparaît, et Cl.G sur le signal permet de le router dans la couche active :



On "déroute" un signal, ou un segment de signal, avec la cmd RIPUP 



A la fin, on pourrait avoir ceci, où on a ajouté un texte d'identification avec la cmd **Text** sur la couche bottom:

"ALIM SYM V1.1"

"Bot - sept 07"

La barre d'outils s'adapte. On retrouve des options déjà connues.

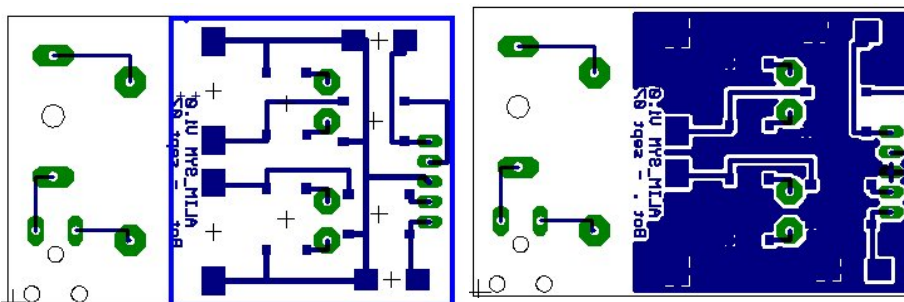
RATIO permet de changer l'épaisseur du trait quand la fonte est vectorielle.

Tout paramètre : taille, largeur, diamètre, isolation... peut être modifié par la cmd **CHANGE**

La commande **CHANGE** permet de changer une caractéristique. On détermine d'abord la nouvelle caractéristique puis on sélectionne l'élément sur lequel elle s'applique. Exemple: Changer une piste vers la couche **BOTTOM** se fera par **CHANGE->LAYER->BOTTOM** puis **CLKG** sur le segment de piste

Pendant le design d'un PCB, on peut changer :	
Layer...	de couche (un composant, une piste, un texte...)
Width	la largeur d'une piste, d'un trait
Style	le style : continu, pointillé...
Cap	la terminaison des arcs, nette ou arrondie
Size	la taille des caractères d'un texte
Font	la fonte du caractère: fixe, proportionnelle ou vectorielle
Ratio...	l'épaisseur du trait d'un caractère
Text	le contenu du champ text déjà présent
Diameter	le diamètre des pastilles de via
Drill	le diamètre de forage d'un trou ou d'un via
Shape	la forme d'une pastille : ronde, carrée, octogonale
Via	les couches concernées par le via
Stop	ON-OFF cream-thermal pour les pad, via, smd
Pour	définit le remplissage des polygone - plein ou hachuré
Rank	de 1(+) à 6(-) priorité dans les polygones sur PCB, 7 pour package
Isolate	fixe l'isolation des polygones, on peut aussi entrer un nombre
Spacing	si polygone fill hatsch -> espacement des hachures
Thermals	autorise ou interdit les ponts thermiques avec les polygones
Orphans	autorise ou interdit les zones orphelines avec les polygones
Class	pour changer la classe d'un signal, voir edit -> netclass
Package	pour changer le boîtier, si variante définie dans le composant
Technology	change TTL, CMOS... si variante définie dans le composant

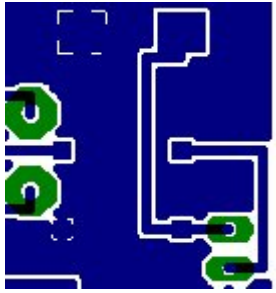
Lorsqu'on a routé le PCB, il est parfois utile de réaliser un ou plusieurs plans de masse. Pour ce faire, on dispose de la cmd « **Polygon** » qui permet de délimiter ce plan. Ci-dessous à gauche, un polygone dont le nom est NS14, car il est considéré comme un signal quelconque. Renommons-le GND, nous obtenons de suite un polygone plein. Les pistes GND sont fondues dans le plan de masse. On revient à la figure de gauche par un **Ripup** sur le polygone. On aperçoit aussi les " THERMAL " : ponts thermiques facilitant la soudure des composants.



La cmd en ligne **SET POLY ON** ou **OFF** permet d'activer ou de désactiver le remplissage du polygone lors d'un Ratsnet


On remarque à droite, près du connecteur, une zone non remplie et une piste près du bord du PCB.

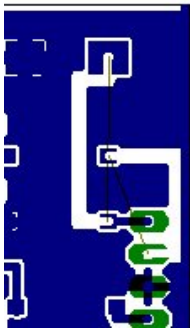
Eagle PCB



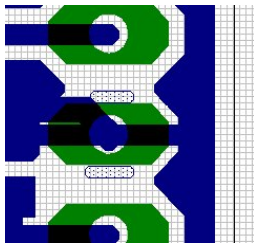
En repositionnant la piste partant de la pin supérieure du connecteur, on permet au plan de masse de couvrir toute la zone, comme le montre la figure ci-contre

Il est possible d'améliorer encore ce design, afin d'éviter qu'une piste passe trop près du bord. Même si les règles d'isolation sont respectées, il est préférable de prendre ce réflexe de s'éloigner du bord tant que s'est possible.

On obtient ainsi après **Ripup** des 2 signaux, puis nouveau routage suivi d'un **Ratsnest**  :

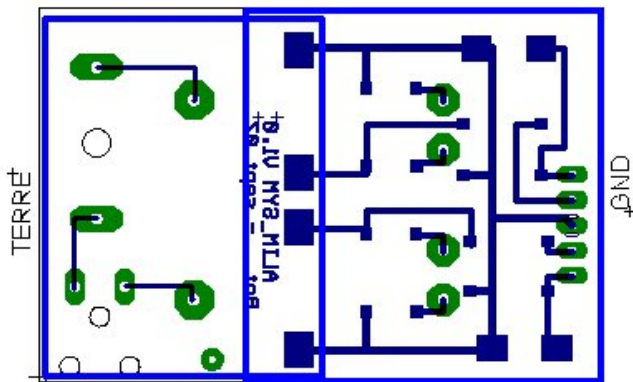


On remarque des traces étranges, pas très élégantes au niveau du pad central du connecteur. Il s'agit des ponts thermiques autorisés dans le DRC et du routage du signal GND qui se superpose au plan de masse. Ils sont arrêtés pour respecter les conditions d'isolation.



On peut aisément remédier à ce problème en utilisant la couche bRestrict pour interdire des pistes de part et d'autre du pad. On sera amené à réduire la grille pour s'approcher au plus près du pad. On trace deux traits (« WIRE ») dans cette couche et on 'ratsnest'. Nous avons alors le pcb ci-contre.

Pour finaliser la carte, ajoutons encore un plan de masse du côté du 220V pour une éventuelle mise à terre. Nous plaçons un **composant** pad que nous nommons TERRE, puis traçons un polygone du même nom. Nous avons donc deux plans de « masse » GND et TERRE.



En cas de recouvrement, comme sur la figure ci-contre, l'attribut RANK permet de régler les conflits d'isolation.

Par défaut le rank est mis à 1.

Si nous changeons le RANK de TERRE à 2, il sera moins prioritaire au remplissage que GND qui sera alors tracé en premier lieu, avec ses conditions..

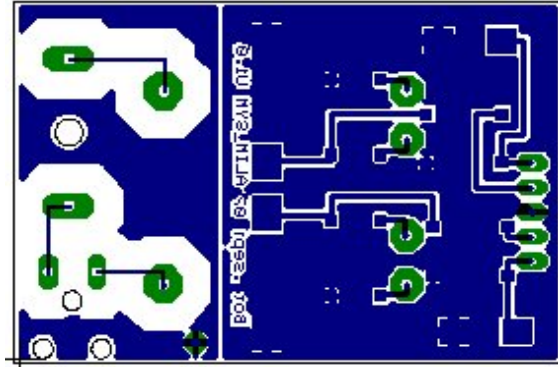
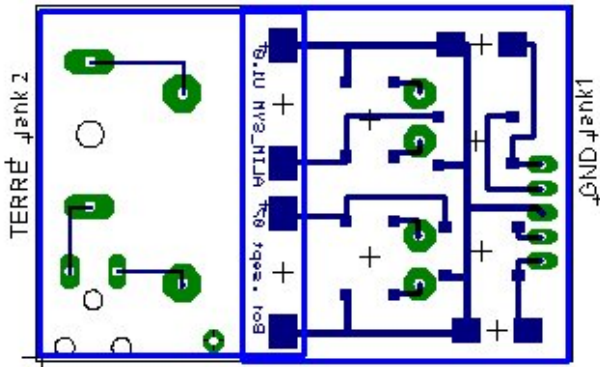
Remarque: Les plans peuvent se définir avant de commencer un routage

Eagle PCB

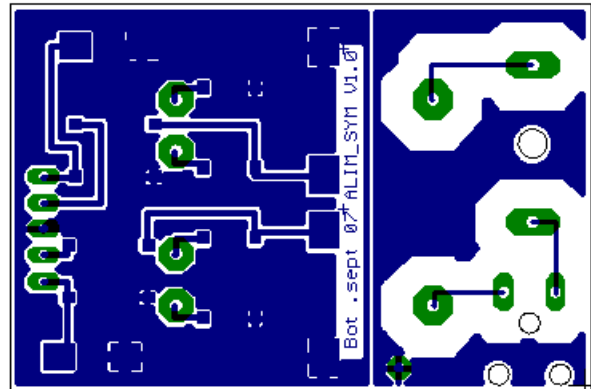
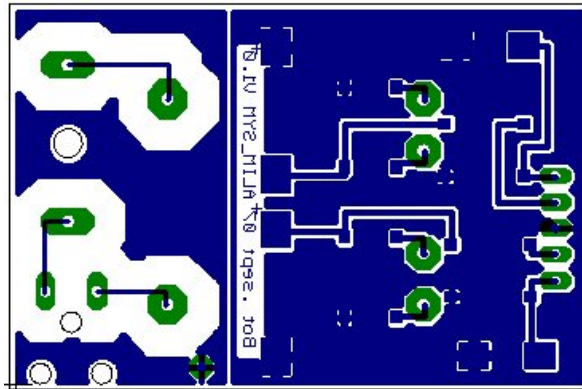
Voici après modification

puis

ratsnest

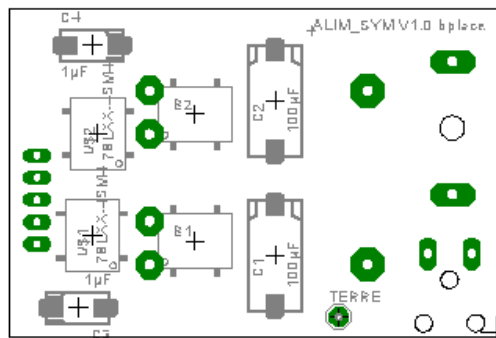
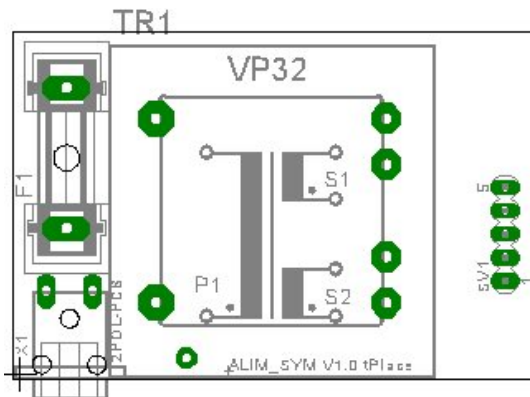


On améliore l'aspect du board en encadrant le texte d'un rectangle dans la couche bRestrict. On a finalement vue du bottom par transparence et sa vue miroir :



Avec les vues correspondantes tPlace et

bPlace (vue miroir):

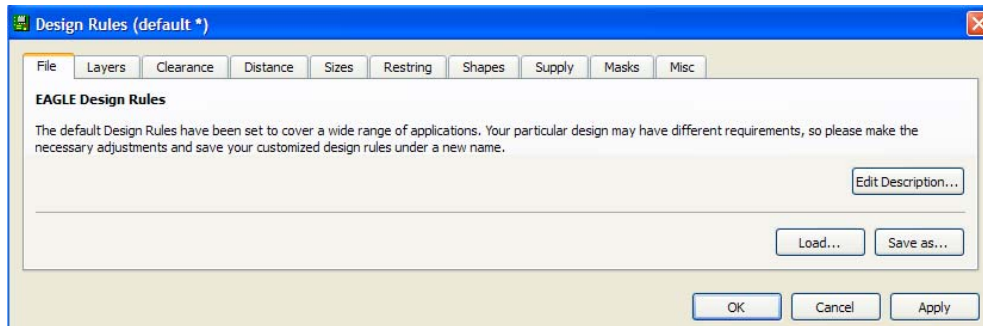


Il est temps de vérifier la conformité de notre routage. Lançons un DRC (Design Rules Check).

Un premier écran Design Rules (DR) apparaît qui permet de définir certains paramètres :

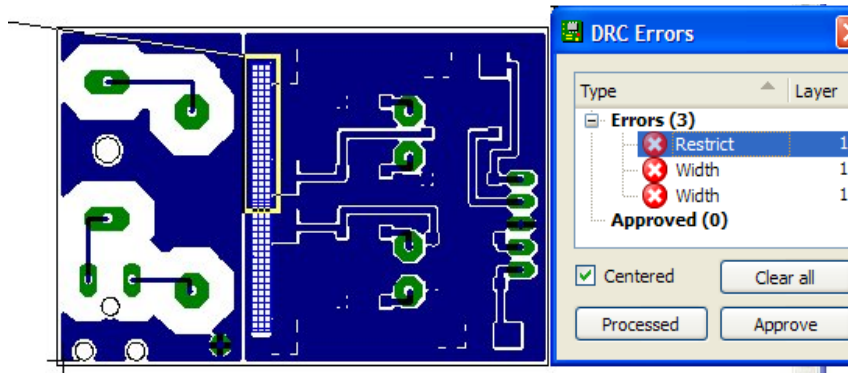
Acceptons les règles par défaut, celles que nous avons définies dans le NetClasses sont incluses dans le projet et sont prioritaires sur les spécifications par défaut.

Nous examinerons ces différents onglets ultérieurement. Faisons ClkD sur OK



Eagle PCB

Une fenêtre montre les erreurs, un pointeur cible sur le PCB, l'erreur sélectionnée dans cette fenêtre.



Et si la case « Centered » est cochée, le défaut est centré dans la fenêtre.

Les 2 erreurs 'Width' désignent le texte dont le trait a une épaisseur inférieure à celle spécifiée par défaut ; sans impact sur le circuit électronique. Pour éliminer cette erreur, on peut changer le **Ratio** du texte.

On pourrait aussi créer une couche réservée à ces informations (voir tdocu ou bdocu)

L'erreur 'Restrict' n'est pas clairement visible, elle est dissimulée derrière l'erreur 'Width'.

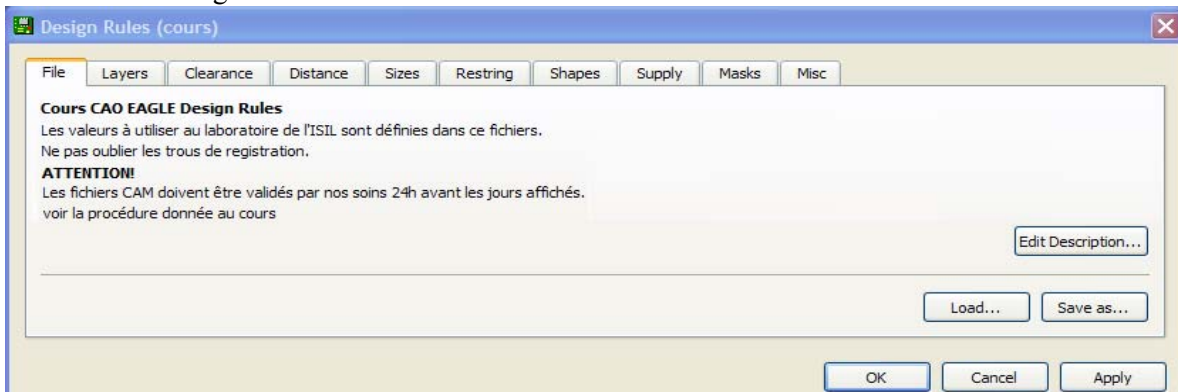
Corrigeons les erreurs 'Width', puis affichons la couche bRestrict par la commande Display, nous constatons que la jambe de la lettre p du texte 'sept 07' touche le cadre bRestrict, ce qui viole la règle et provoque un message d'erreur (sans incidence dans ce cas). Pour éliminer cette erreur, il faut déplacer le texte.

Les messages que peut délivrer l'exécution d'un DRC sont bien sûr fonction des choix réalisés dans l'éditeur DR et dans les "Net classes" (onglet Edit), mais ils résultent aussi d'autres inepties comme router hors carte, croiser deux pistes dans une même couche, traverser un trou avec une piste...

Voyons donc en détail l'éditeur DR; puis ensuite ses messages d'erreur.

Dans l'onglet **File**, un bouton "**Edit Description...**", permet de décrire le profil du DR, ce commentaire apparaîtra dans la fenêtre droite du CP, on peut aussi sauvegarder le DR ou en charger un autre ou encore appliquer les modifications au profil en cours.

Les fichiers de règles ont l'extension .DRU.



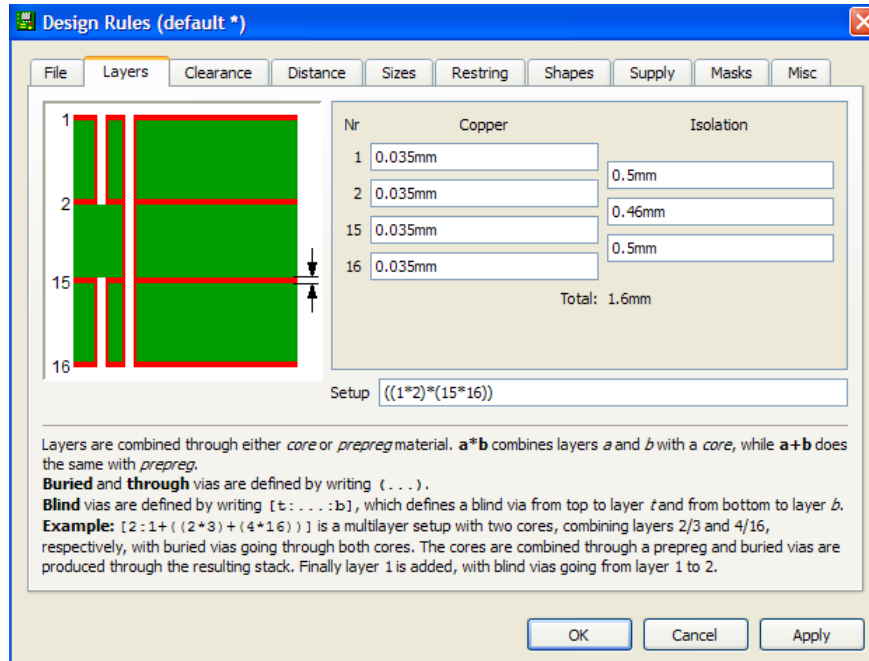
Le fichier COURTS.DRU placé sur l'école virtuelle reprend les règles à respecter pour pouvoir faire tirer un PCB dans nos laboratoires. Il tient compte des possibilités de nos équipements, il est donc inutile de vouloir les dépasser sans nous consulter. Pour les étudiants inscrits, le fichier est accessible à l'adresse:

http://ecolevirtuelle.provincedeliege.be/ctrl/ctrl_gestion.pageBrowser?p_jsCtrlName=#45659.

2.5 Les options de "Design Rules"

L'onglet **Layers** reprend les caractéristiques des couches et des vias :

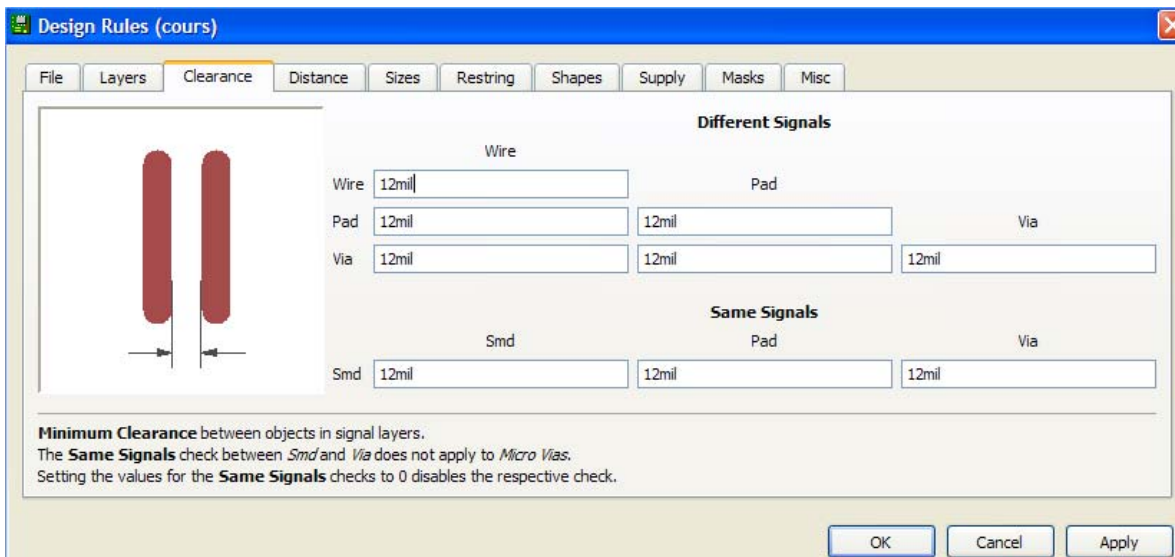
On y spécifie l'épaisseur du cuivre (35, 70, 105, 140µm) et l'épaisseur d'isolant entre chaque couche.



Dans une configuration multicouche on distingue les couches internes (**inner layers**) des couches externes (top et bottom). Les vias exclusivement internes (**buried**) ne sont pas visibles ni par le top ni par le bottom. Les vias **through** relient le top au bottom, ils traversent la carte. Les vias aveugles (**blind**) relient une couche externe (top ou bottom) à une couche interne. Dans le setup, ‘*’ et () définissent les groupements internes (buried via) qui associent

deux couches comme top et bottom en double face.

Les [] définissent les liens des groupes internes seulement avec les 2 couches externes.



L'onglet **Clearance** définit les distances d'isolation entre signaux différents ainsi que pour des signaux identiques.

On remarquera que l'illustration de gauche change avec la donnée sélectionnée, ainsi pour l'isolation entre **wire** et **pad** on obtient l'illustration ci-contre.

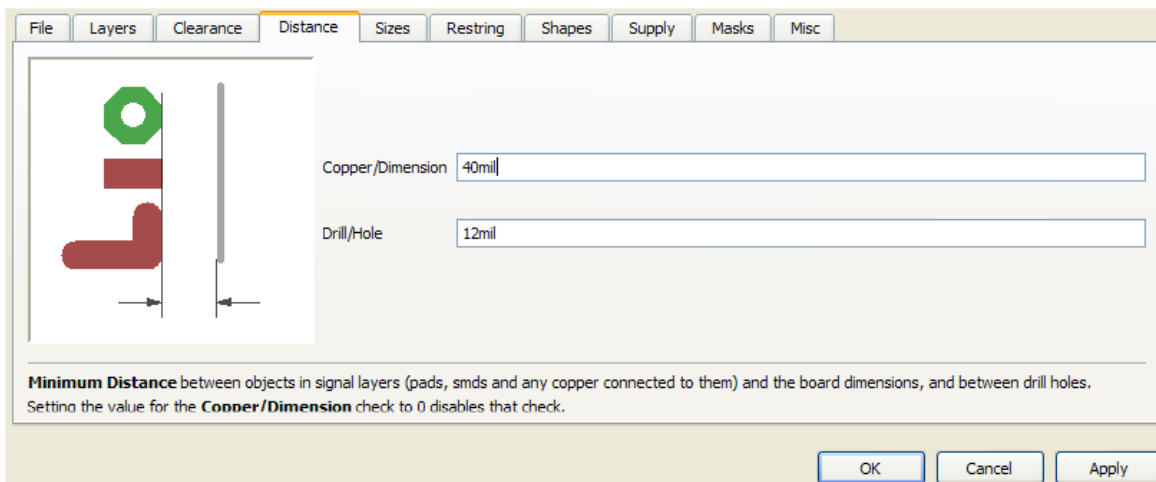
Pour chaque piste on définit l'isolation par défaut avec les autres pistes, avec les pad et les vias étrangers. On définit également les distances minimales des pads entre eux et avec les vias.



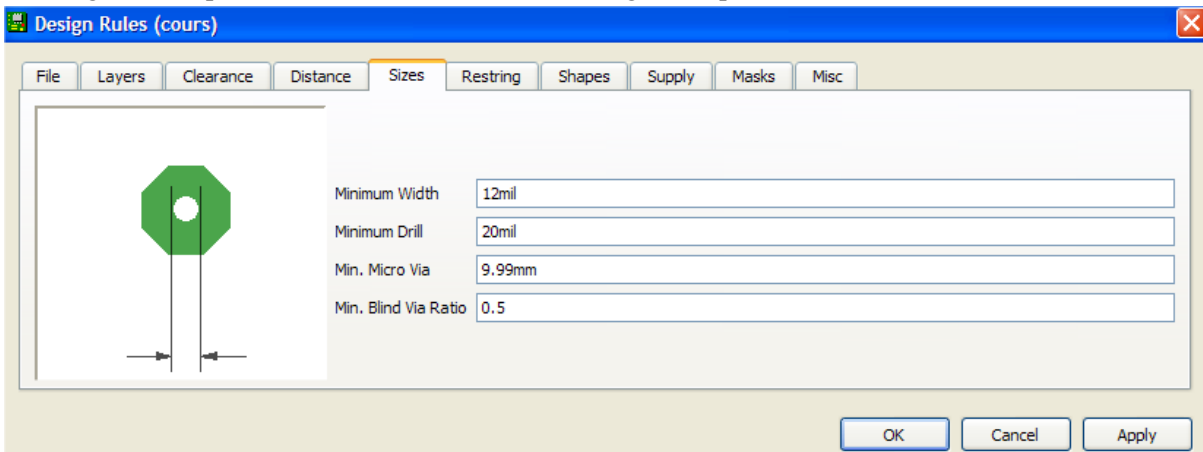
Eagle PCB

L'onglet **Distance** détermine les distances minimales à respecter, à celles définies dans la couche dimension. Cela concerne les bords de la carte, mais aussi les trous et autres découpes.

ATTENTION ! Mettre 0 désactive le test, le DRC ne teste plus les pistes passant sur des trous !

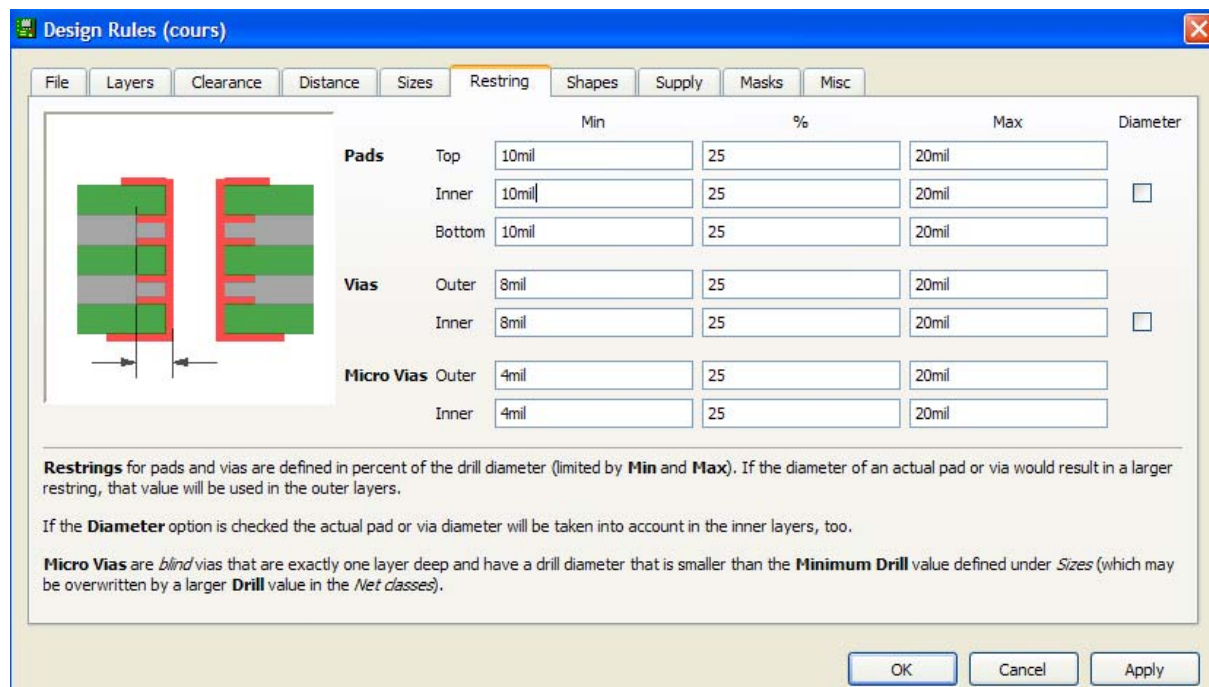


L'onglet **Sizes** pour les dimensions minimales : largeur de piste, diamètres, microvia...



L'onglet **Restraining** indique en % du diamètre, l'anneau de cuivre restant après forage.

Exemple : un anneau de 0,2mm pour un trou de 0,8mm



Eagle PCB

Dans chaque couche, pour les pads et les vias. Il est possible de spécifier des valeurs différentes entre top, bottom et couches internes.

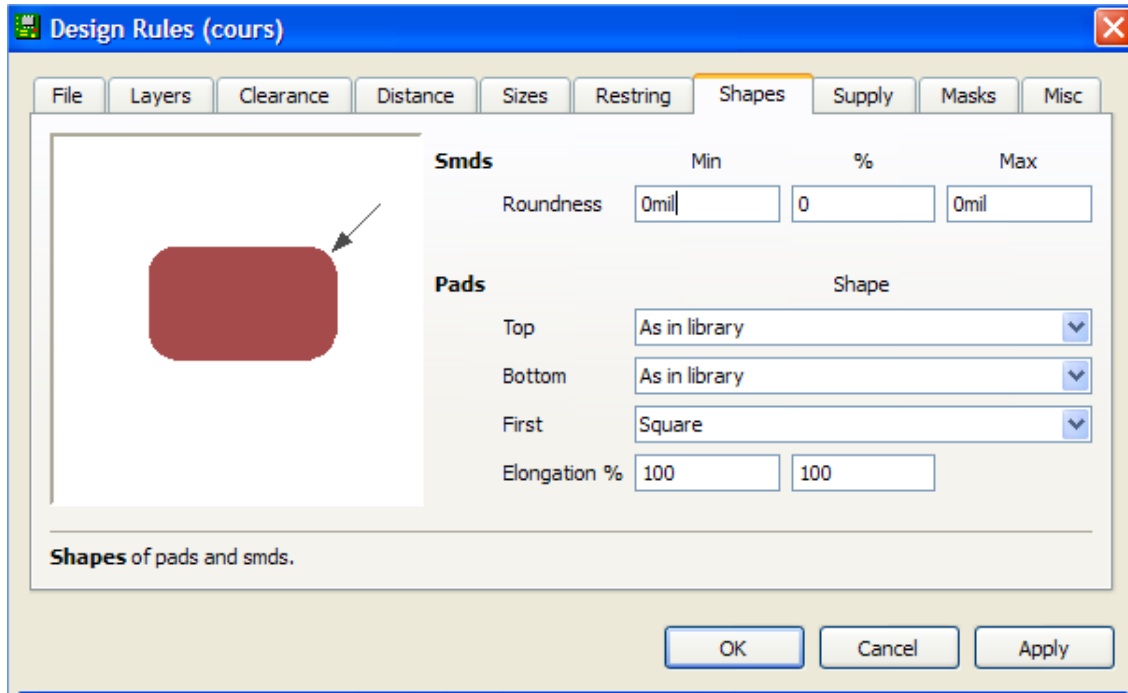
Les pads du top peuvent donc être différents de ceux du bottom.

Si diamètre est coché, pad et via devront avoir le même diamètre dans les couches internes.

L'onglet **Shapes** permet de modifier les formes des pads et pastilles smd.

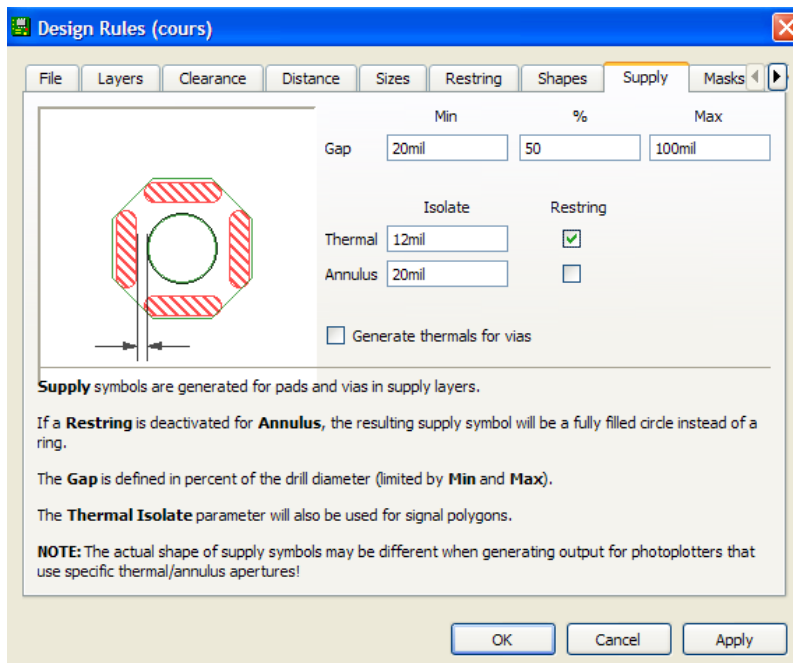
SMDS : Spécifie l'arrondi des pads smds, 100% transforme une pastille carrée en un cercle.

PADS : On peut imposer la forme des pads différente sur le top et le bottom, ne changer que la première pin d'un CI...



Apply : réalise immédiatement la modification dans le board courant.

L'onglet **Supply** concerne les couches d'alimentations (pont thermique dans les plans de masse)



Gap : c'est la largeur du pont thermique (en % du diamètre) entre l'anneau et le plan.

Annulus : largeur de l'anneau

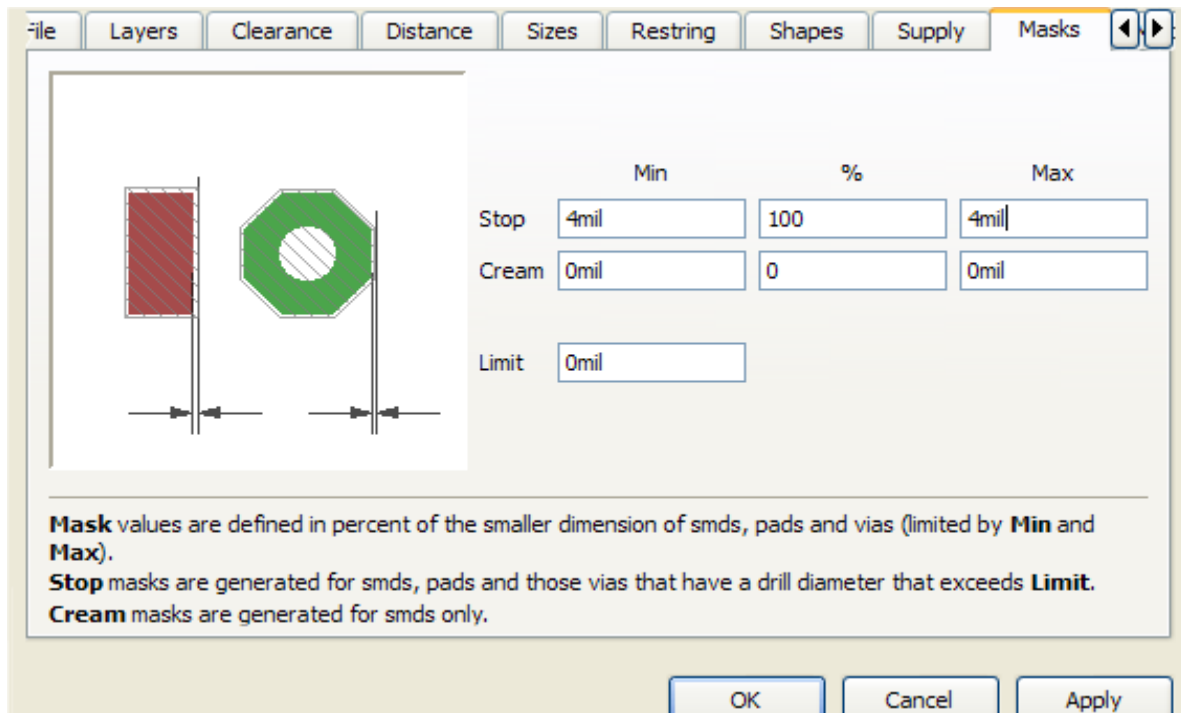
Thermal : largeur de l'isolation thermique

Restricting Thermal : distance au trou définie dans l'onglet Restricting

Restricting Annulus : si cochée alors cercle plein au lieu d'un anneau

Eagle PCB

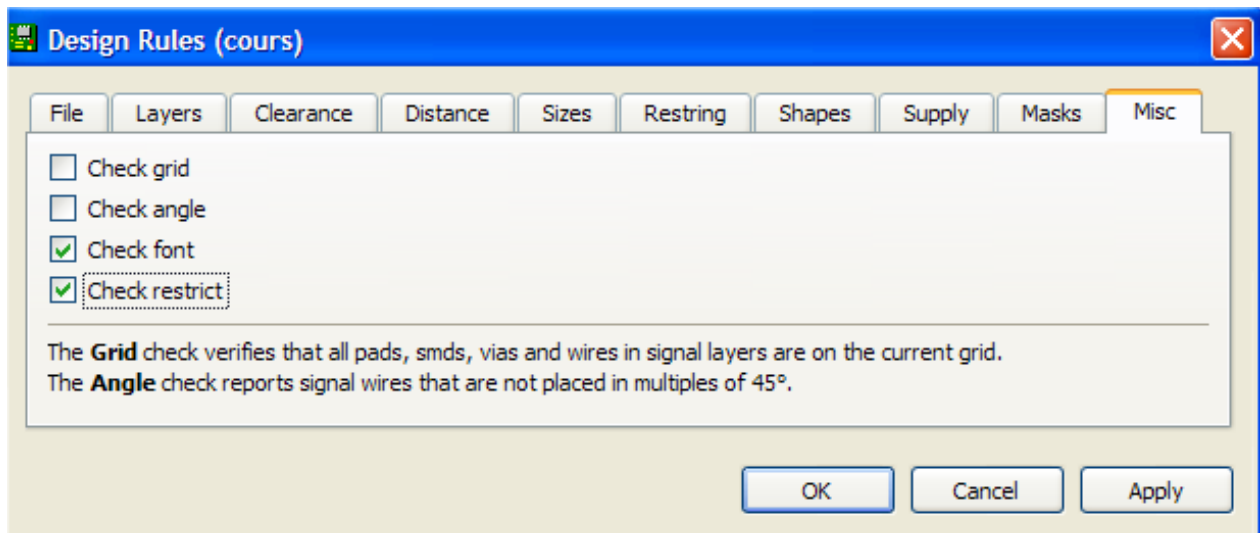
L'onglet **Mask** reprend les paramètres des masques de soudure pads et smds
Stop mask est défini pour les pads et les smds; masque pour les zones sans vernis.



Cream mask : masque pour la pâte à souder, n'est défini que pour les smds
Les valeurs sont données en % de la plus petite dimension.

Limit : indique le diamètre minimum des via qui généreront un stop mask
(Voir aussi les flags STOP et CREAM dans la définition du package)

L'onglet **Misc** définit divers paramètres à contrôler :



Grid : pour vérifier que les éléments sont correctement positionnés sur la grille courante.

Pas souvent coché car on utilise aussi souvent une grille métrique et une grille anglo-saxonne.

Angle : vérifie que les tracés de pistes respectent des pentes multiples de 45°.

Font : Vérifie que le texte est écrit en fonte vectorielle, la seule fonte respectée par le CAM processor. Ne pas cocher cette case et utiliser le CAM processeur pour générer des fichiers Gerber pour la fabrication risque d'avoir des effets désastreux à la sortie.










La taille des caractères n'étant pas respectée peut engendrer des courts-circuits.

Restrict : Vérifie que du cuivre (piste, pad, polygone) ne se trouve pas dans une zone définie dans une des couches Restrict

Remarque :

Après avoir réglé tous les paramètres, on peut sauver le DRC avec le nom du schéma.

Ensuite on click OK et le logiciel cherche les erreurs dont les plus courantes sont les suivantes:

	Angle	La pente n'est pas un multiple de 45°
	Clearance	Distance d'isolation non respectée
	Dimension	Trop près du bord de la carte, ou d'une découpe
	Drill Size	Diamètre de forage hors limite
	Keepout	Composant placé à un endroit interdit
	Off Grid	L'origine du composant n'est pas sur la grille
	Overlap	Une piste passe sur une autre ou sur un pad
	Restrict	Une piste passe dans une zone interdite
	Width	La largeur de la piste est hors des limites de la classe

Cette liste n'est pas exhaustive, par exemple des signaux routés en pointillé généreront le message d'erreur **Wirestyle**, d'autre actions comme l'usage de fonction interdite dans certaines couches peuvent générer des "**Layer Abuse**"

On peut énumérer d'autres messages d'erreurs :

Blind via ratio, Drill Distance (distance entre deux trous non respectée),

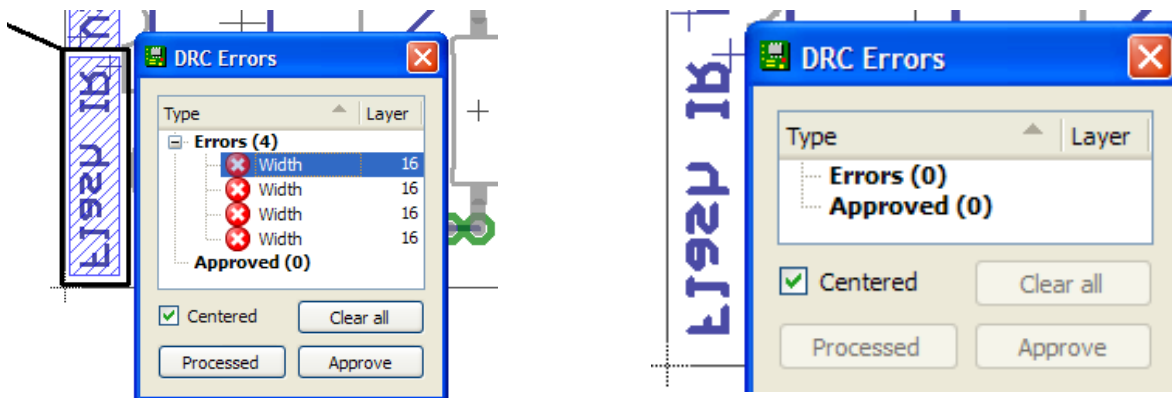
Invalid Polygon (généré si un polygone n'est pas connexe),

Layer Setup (conflit entre la définition des couches et l'usage de certains objets comme des smds au top en simple face où seule la couche bottom est active),

Micro Via Size, No vector font, No real Vector Font, pour lesquels on ira voir l'aide en ligne.

Tous ces messages trouvent leur utilité dans le respect des différentes spécifications faites dans l'éditeur DR, des règles implicites ou d'usage chez les sous-traitants.

Enfin, le résultat d'un DRC, nous donne la possibilité d'approuver des erreurs ou des avertissements, Cela pourrait-être le cas pour un texte dont le trait est trop fin, mais augmenter le ratio peut éliminer le message d'erreur comme le montre les figures ci-dessous.




Ceci pour énoncer une règle importante :

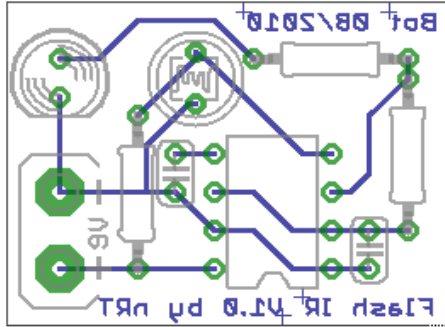
" Le DRC d'un PCB idéal ne produira aucune erreur ni avertissement, même approuvé" C'est l'objectif à atteindre sans tromperie. **Ne pas changer les paramètres du DRC pour éliminer les messages d'erreurs, au détriment des règles imposées est faire preuve de sagesse.** Seuls quelques cas exceptionnels (masse digitale connectée à une masse analogique par exemple) peuvent être tolérés.

L'essentiel reste donc de bien maîtriser les contraintes liées à son PCB et de les traduire correctement en paramètres dans les DRC.

3. L'autorouteur

L'autorouteur  est un outil de routage automatique très performant, s'il est correctement configuré. Sa configuration est cependant délicate à effectuer tant le nombre de paramètres est important et leur impact sur le logiciel routeur pas toujours facile à déterminer voire à appréhender.

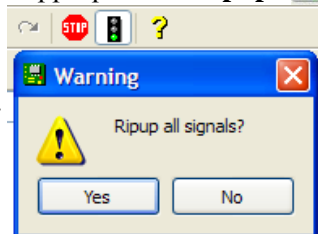
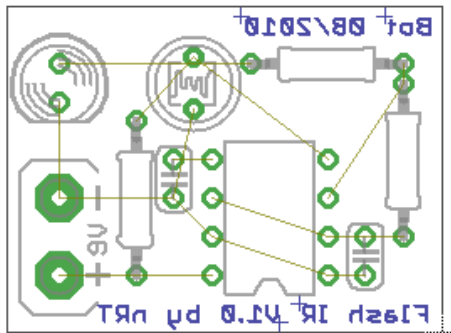
Reprenons l'exercice FLASH IR, de la section schématique, un routage manuel sur le bottom pourrait donner ceci.



Ce routage est assez aisé car la carte est simple et comporte peu de connexions.


En d'autres termes, l'autorouteur n'est vraiment requis, mais son usage permettra de mettre en lumière sa complexité.

Nous partirons donc de cette carte, en lui appliquant un **Ripup**  → puis **Go** .



Une fenêtre apparaît et Cl.Yes et nous retrouvons la carte non-routée.

L'**AUTOROUTER** permet un routage automatique dans le respect du DR et des spécifications du Net Classes.

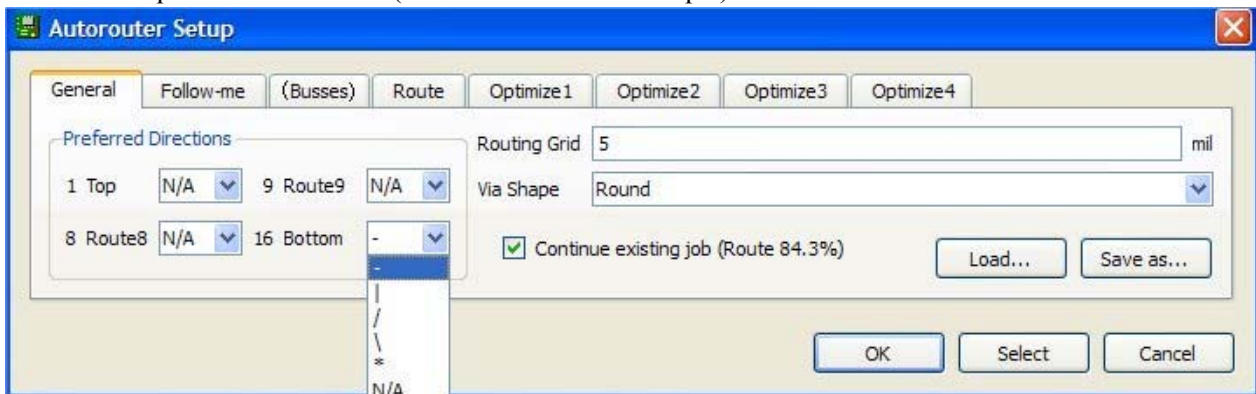
On le lance par l'icône  et



une fenêtre apparaît, dont l'aspect dépend de la licence et de la configuration du DR. Ici, seulement une couche est active et cela nous donne la fenêtre ci-dessous (des couches sont

supprimées pour réduire la taille graphique).

N/A signifie couche non-active, les traits - / \ | fixent les directions préférentielles, * pas de préférence. Je conseille de prendre au plus 25mil comme grille de routage, 10 est un bon compromis selon la mémoire disponible dans le PC (de moins en moins critique).



Load permet de charger un Setup déjà sauvegardé (*.ctl files).

Select pour sélectionner les signaux à router avec la commande **GO**, mais il est préférable d'avoir préalablement défini toutes les options de l'autorouteur. En ligne de cmd, "**auto V+**" produit le même genre de sélection, par contre "**auto !V+**" routera tout sauf V+

Les onglets entre () sont désactivés comme ici (Busses) par exemple qui est généralement utilisé en premier lieu si des bus sont définis dans le schéma. Contrairement aux autres, les 2 onglets **General** et **Follow-me** ne peuvent être désactivés.

"**Continue existing job**" apparaît lorsqu'on relance un auto-routage qui avait été interrompu. On décoche la case pour redémarrer un nouveau routage.

Eagle PCB

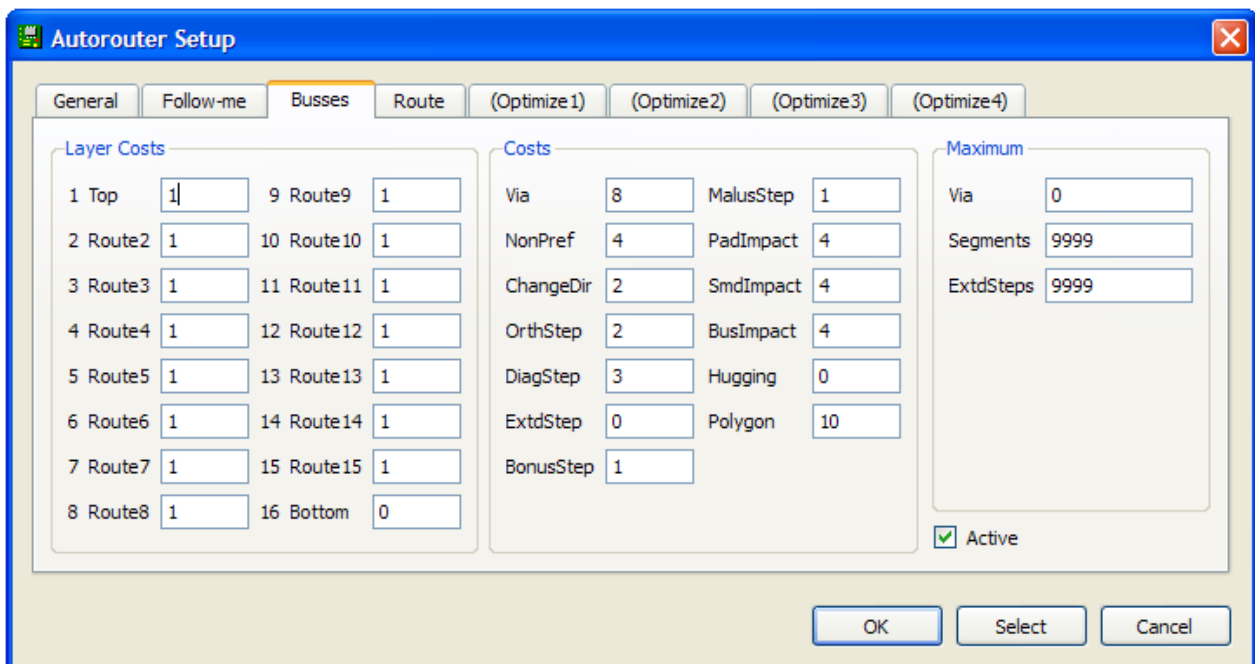
L'onglet "**Follow-me**" est un routeur spécifique réservé aux signaux pour lesquels on a spécifié 8 ou 9 comme "**wire bend style**" par la cmd en ligne "**set wire_bend 8**" pour router sur une seule face, mais au plus court, tandis que la cmd en ligne "**set wire_bend 9**" routera au plus court sur les deux faces
Voici l'essentiel de la cmd **set bend_wire**(voir le manuel) :

Bend angle for wires	SET WIRE_BEND bend_nr;
	<i>bend_nr</i> can be one of:
	0: Starting point - horizontal - vertical - end
	1: Starting point - horizontal - 45° - end
	2: Starting point - end (straight connection)
	3: Starting point - 45° - horizontal - end
	4: Starting point - vertical - horizontal - end
	5: Starting point - arc - horizontal - end
	6: Starting point - horizontal - arc - end
	7: "Freehand" (arc that fits to wire at start, straight otherwise)
	8: Route short end of airwire in Follow-me router
	9: Route both ends of airwire in Follow-me router
	Note that 0, 1, 3 and 4 may contain additional miter wires (see <u>__miter</u>).
	SET WIRE_BEND @ bend_nr ...;
	Defines the bend angles that shall be actually used when switching with the right mouse button.
	SET WIRE_BEND @;
	Switches back to using all bend angles.

Nous laisserons à qui le veut, le soin d'explorer et d'exploiter cette voie.

L'onglet "**Busses**" est prévu pour les systèmes avec bus de données/adresses ou de même nature et devant présenter des caractéristiques similaires de routage (longueur,...). Indiqué pour les liaisons entre µP et mémoire par exemple.

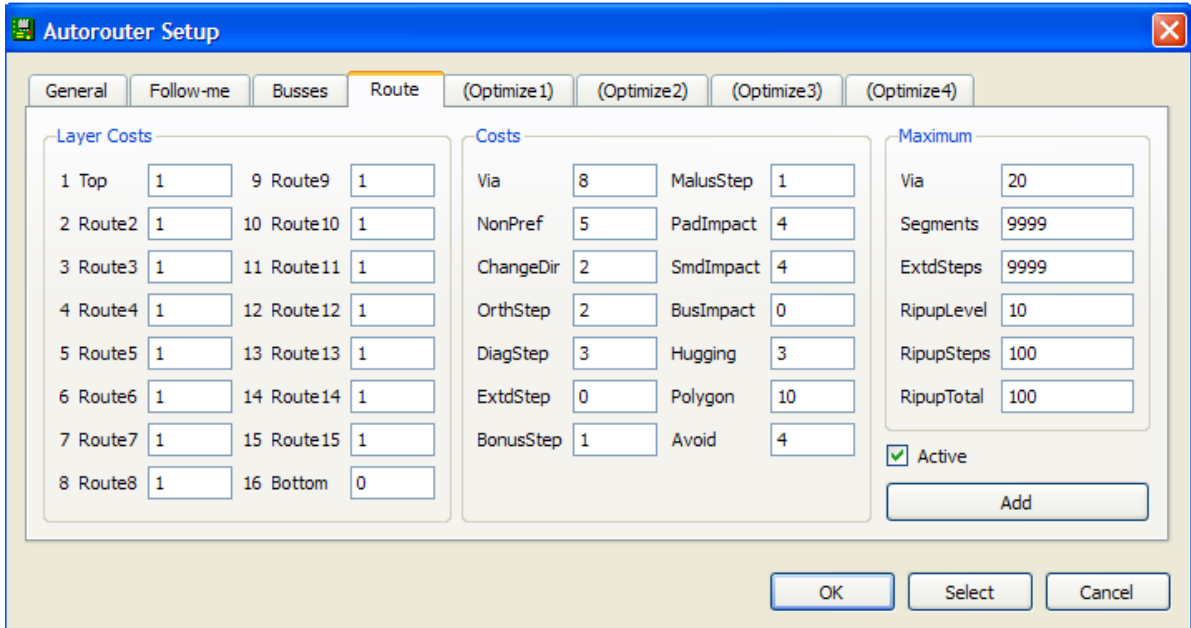
Si on veut utiliser cette option, il y a lieu de ne pas mixer les bus, il est préférable de déclarer 2ou + bus homogènes plutôt qu'un seul bus hétérogène.



Eagle PCB

Mais comme l'onglet suivant "**Route**" présente au moins les mêmes options, en plus nous y verrons les **Ripups** et **Avoid**. Nous allons donc détailler les options de cet écran.

Cette fenêtre est divisée en 3 sections : 2 pour des "coûts" et à droite pour les maximums (0..9999). L'autorouteur travaille avec un facteur de coût virtuel (de 0 à max 99) attaché à chaque paramètre.



Certains paramètres ont des plages de valeurs nettement plus réduites.

Ainsi, la section "**Layer Costs**" (0..20), pour des PCB multicouche permet de favoriser l'usage d'une couche en lui assignant un coût faible, ou d'en défavoriser l'usage avec un coût élevé.

En général, les couches " top et bottom" reçoivent 0 quand elles sont utilisées.

La section centrale "**Costs**" fonctionne semblablement sur les paramètres du routage.

Les valeurs affichées sont celles par défaut. Reprenons chacune d'elles avec leur plage de valeurs.

Via (0..99): Contrôle l'utilisation des vias, une faible valeur en favorise l'usage, permettant ainsi le respect de la direction préférentielle (PrefDir), à l'opposé, une valeur élevée en favorisera le non respect. En général, on route avec une faible valeur, puis on optimise avec des valeurs élevées. 99 n'empêche pas l'usage de via.

NonPref (0..10): Pour le tracé les pistes dans la direction préférée (voir onglet General). Une faible valeur autorise davantage de changements de direction.
99 oblige le routage exclusivement dans la direction de préférence.

ChangeDir (0..25): Pour réduire le nombre de changement de direction avec une valeur élevée.

OrthStep (2), DiagStep(3): à ne pas modifier, traduction qu'une hypoténuse est toujours plus courte que la somme des deux autres cotés.

ExtdStep (0..30): traduit le coût d'un segment de piste (un pas de la grille), à 45° de la PrefDir, au-delà du maximum autorisé. Exemple, si dans Maximum, **maxExtSteps=0**, chaque pas de la piste coûtera la valeur définie entre 0 et 30.

Mais **maxExtSteps=10**, alors la piste pourra traverser 10 pas sans surcoût.

Mettre **ExtSep=99** et **maxExtSteps=0** évite le tracé à 45° de PrefDir

On routera avec une faible valeur, qui augmentera pour l'optimisation.

BonusStep, MalusStep (1..3): Coefficient de pondération des l'influence des zones favorables et défavorables pour le routage du signal. (voir PadImpact)

PadImpact, SmdImpact (0..10): Les pastilles pour souder les composants (smds et traversants)

Eagle PCB

BusImpact : génèrent avec le logiciel des zones favorables (dans la PrefDir) ou défavorable (perpendiculaire à PrefDir). Un signal routé dans la PrefDir, le sera d'autant plus loin des pastilles que ce coefficient est élevé, mais le plus longtemps possible dans la PrefDir. Pour de grande densité de composant, il est conseillé d'augmenter la valeur.

Hugging (0..5) : Une valeur élevée force le rapprochement des segments parallèles.
On recommande une valeur plus faible pour l'optimisation.

Avoid (0..10): Durant les ripups, indique comment les anciens tracés doivent être considérés comme zones à éviter. L'effet est d'autant plus prononcé que la valeur est grande.
Ce paramètre n'est actif que pour l'onglet "**route**"

Polygon (0..30): Chaque pas de routage dans le polygone est associé à ce coût. Une forte valeur diminue le risque de voir le polygone se fractionner.

La section "**Maximum**" contrôle les paramètres suivants :

Via (0..30): Le nombre maximum de via pour une connexion

Segments(0..9999): le nombre maximum de segments pour une connexion.

ExtdSteps (0..9999): Le nombre de pas de grille à 45° de la PrefDir autorisés gratuitement

Les Ripup/Retry

Ces paramètres ne sont actifs que pour l'onglet "**route**", et si ces valeurs sont élevées, le programme risque de prendre davantage de temps à respecter les contraintes imposées.

L'autorouteur travaille par tentative itérative. Il route les signaux les uns après les autres, jusqu'à être bloqué, à ne plus trouver de passage pour le signal. Alors il mémorise la situation, puis "déroute" (ripup) des signaux déjà routés jusqu'à concurrence du nombre spécifié par **RipupLevel**.

Ensuite, il recommence une nouvelle tentative qui peut conduire à un nouveau ripup/retry imbriqués dans le premier.

Le nombre maximum de ripup imbriqués est fixé par **RipupSteps**.

RipupTotal fixe le maximum des signaux dérivés simultanément, tous comptabilisés.

Si une de ces valeurs est dépassée, l'autorouteur arrête ses tentatives et déclare le signal non-routable. Il remet ses compteurs à zéros, reprend à son dernier routage valide et poursuit alors de la même manière avec les signaux restants.

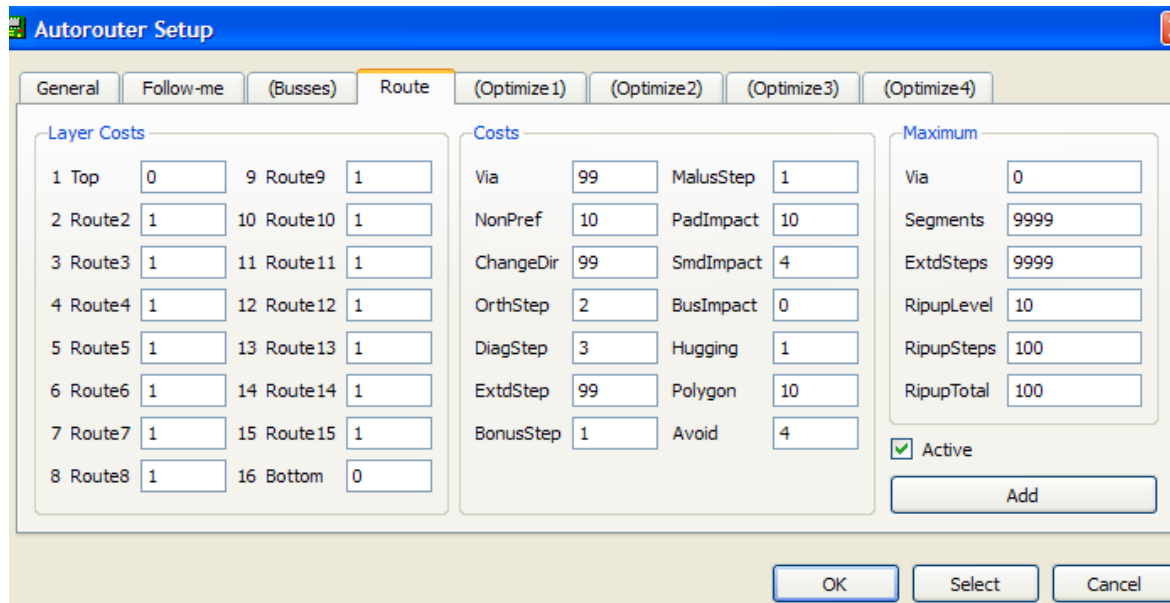
A l'issue de son travail, le logiciel crée un fichier récapitulatif *.pro, où l'on peut lire diverses informations dont le nombre de connexions à router et le résultat de son travail.

Pour configurer efficacement ces paramètres, il y a lieu de tenir compte du nombre de connexions à réaliser, des ressources du PC et du temps que l'on octroie au logiciel pour satisfaire nos exigences. Mais si il y a 20 connexions à router, il est inutile de mettre **RipupLevel**=100, mais bien indiqué de le mettre à 20.

Eagle PCB

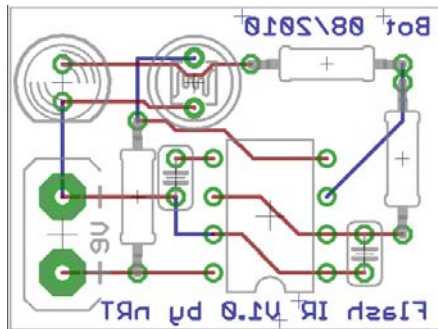
Par contre **RipupStep** peut très bien être fixé à 100, et **RipupTotal** à 1000, ce qui risque évidemment de prendre énormément de temps. Il vaut mieux analyser le résultat d'une première approche et voir ce qui pose problème, avant d'affiner la valeur de ces paramètres; car déplacer un composant de quelques mm suffit souvent à débloquer la situation.

L'optimisation

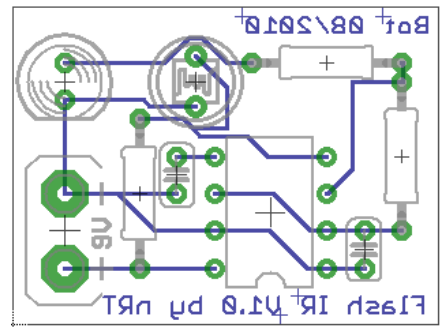


Le nombre de phases d'optimisation est laissé libre car on peut les activer ou désactiver, on peut aussi en supprimer ou en créer de nouvelles. Comme on le voit ci-dessous, les paramètres restent identiques (sauf avoid et ripup)

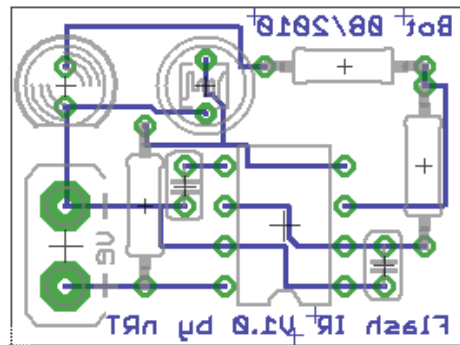
La meilleure façon d'apprendre à utiliser ces phases est de faire des essais sur des PCB de complexités différentes, afin de réaliser les variations qu'apportent ces paramètres dans le routage du circuit. Ainsi notre application Flash IR routée avec la configuration ci-dessus donne la figure de gauche.



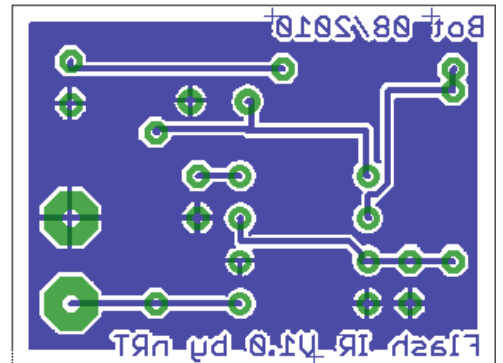
A droite, avec Top Layer cost mis à 99, BonusStep et MalusStep mis à 3



Si on active optimise1 avec comme changement Segments=5 et ExtdSteps=1, on obtient



Cela montre comment peuvent influencer les variations de quelques paramètres. A droite, ci-contre, on a ajouté un plan de GND, et on a tourné la LDR de 90°.



La richesse de ce routeur est loin d'être épuisée dans ce fascicule...