

EAGLE V5.x

Manuel d'utilisation

1. Introduction

Un schéma est l'interface entre une conception électronique et sa réalisation.

C'est lui qui montre comment, d'un cahier des charges, on a obtenu électroniquement les fonctionnalités désirées. Il doit le faire le plus clairement possible, avec le cas échéant les références nécessaires à mieux cerner la réalisation pratique, à pouvoir y localiser aisément les composants. Tout comme Boileau qui a écrit : "Ce qui se conçoit bien, s'énonce clairement et les mots pour le dire arrivent aisément", on pourrait dire : " Une solution électronique qui se conçoit bien se dessine clairement, et les symboles pour le faire se dispose aisément."

Un schéma doit pouvoir se lire comme un livre.

On y trouvera donc l'équivalent d'un titre, d'un auteur, d'une édition, d'une table des matières, voire d'une bibliographie, d'une préface et une structure dans sa présentation.

On doit aussi pouvoir y apporter des modifications et pas uniquement sur le poste où il a été créé. Il est en effet inconcevable qu'un schéma ne soit modifiable que sur un seul poste.

Les quelques pages suivantes ont pour but de donner une méthode de création de schéma sous EAGLE 5.x, qui, sans se prétendre ni universelle ni idéale, aboutira à un dossier répondant aux mieux à ces objectifs. Ce petit manuel ne concerne que l'approche schéma, d'autres reprendront le PCB, la librairie, le CAM processor et les ULP.

Le dossier complet ainsi comportera tous les éléments nécessaires à la modification du schéma ou à l'édition de n'importe quel composant, quelque soit le poste sur lequel on travail.

Une connaissance correcte du logiciel est évidemment nécessaire et, à défaut, devra être acquise dans un contexte hors de toute contrainte temporelle urgente. Il importe de pouvoir maîtriser les subtilités de l'outil pour en tirer un maximum d'efficacité. Consacrer un temps certain à cet apprentissage s'apparente donc d'avantage à un investissement qu'à une nécessité temporaire.

Le circuit imprimé (PCB pour "printed circuit board") est la matérialisation des connexions présentes dans le schéma. Le PCB est un tracé de pistes de cuivre établissant des contacts entre différents points d'une carte électronique. Ces tracés qui peuvent être réalisés sur plusieurs couches et sur des supports très différents (époxy, plastic, verre, céramique...) seront l'objet de la seconde partie du guide. Une troisième partie traitera de la création des composants et des librairies tandis que la dernière donnera la procédure à suivre pour générer les fichiers utiles à la production et au montage automatisés.

Dans le guide d'utilisation du logiciel, nous conserverons la dénomination anglo-saxonne des outils et objets définis et utilisés par le logiciel(par exemple WIRE pour fil, ou encore NET pour signal..).

Nous utiliserons des abréviations dont voici la première partie de la légende, concernant la réalisation schématique :

Cl. : 1 Click souris ou pad et donc **Cl.D** pour Click droit et **Cl.G** : Click gauche

cmd : commande contextuel dans la zone de commande

lcmd: ligne de commande, zone de saisie de texte

CP. : Control Panel : fenêtre principale d' EAGLE

Lbr. : Librairie

P. : Projects

N. : New

Nav. : navigateur

Sch. : schéma

Brd. : board, le dessin du circuit imprimé(PCB)

Top : le coté composant

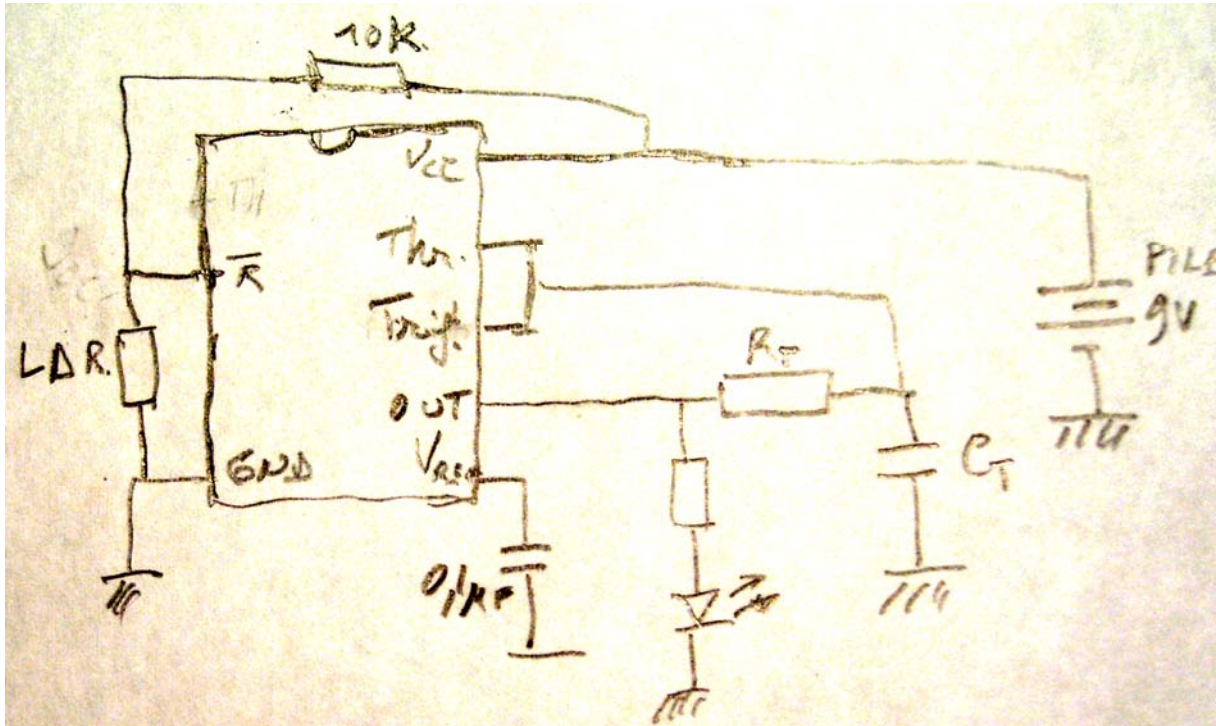
Bot(tom): le coté soudure des composants traversant, soit le coté piste d' un PCB simple face.

Comme objectif, support de progression dans le logiciel, nous partirons d'une application didactique simple.

Il s'agira de réaliser un flash infrarouge alimenté par une pile de 9V, très bon marché et occupant un faible volume, donc utilisant un minimum de composant.

Le système se déclenchera automatiquement de nuit.

Une solution qui vient rapidement à l'esprit est la suivante, croquée en vitesse au crayon, sur le premier morceau de papier disponible...



La valeur finale des composants sera déterminée ultérieurement.

Dans une première approche, nous allons réaliser ce schéma dans Eagle avec des composants standards.

Notre schéma comportera donc 1xNE555, 1xLDR, 1xLED IR, 4xR, 2xC et un connecteur pour pile 9V.

Le logiciel étant assez complet, on peut raisonnablement espérer que tous les composants nécessaires seront déjà prédéfinis.

2. Eagle schematic capture

2.1 Ecran de contrôle ou CONTROL PANEL

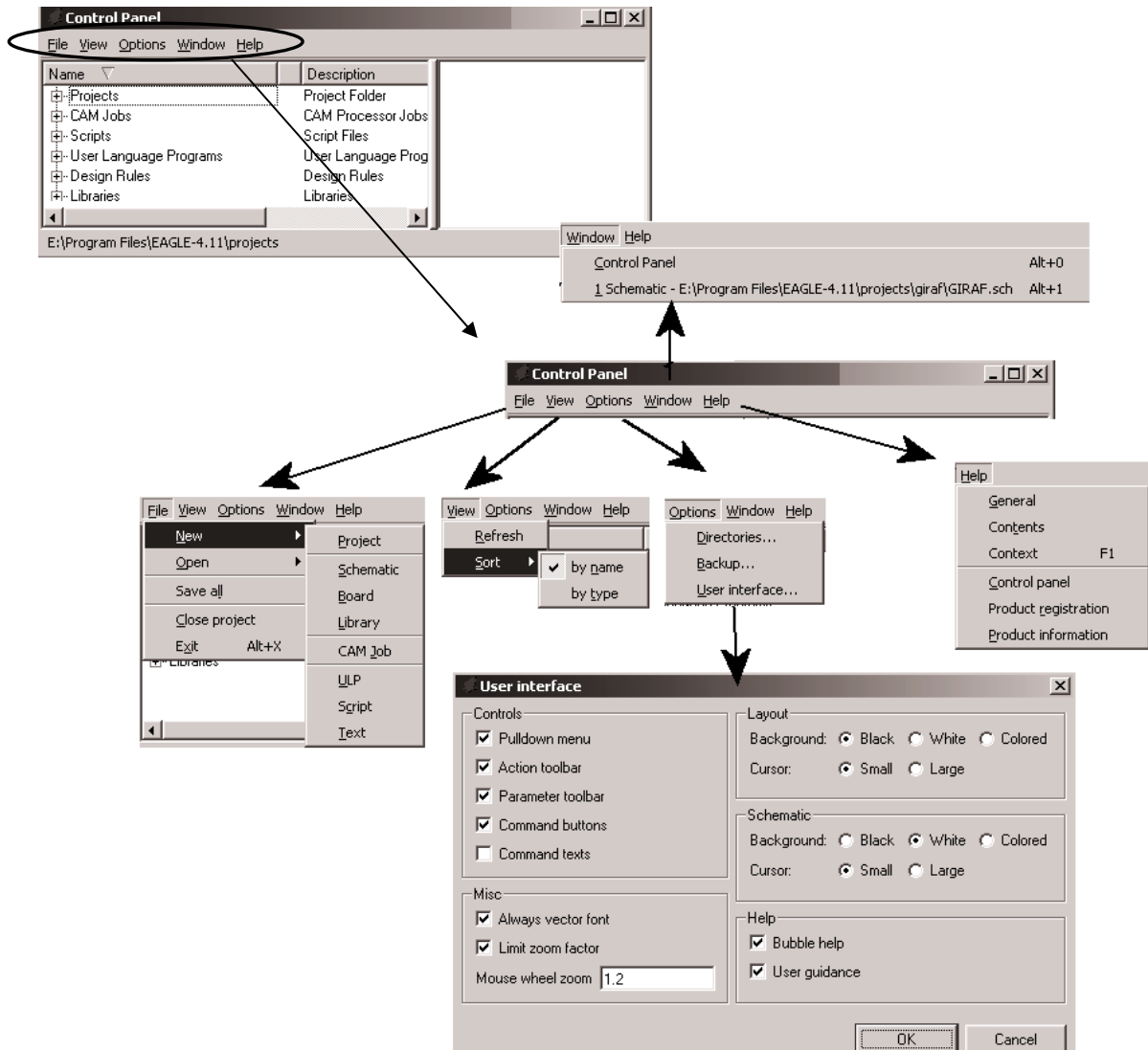
EAGLE (Easy Applicable Graphical Layout Editor)

Le lancement du logiciel aboutit à la fenêtre suivante, appelée "Control Panel" car elle centralise en fait toutes les informations, fonctionnalités et tous les projets.

Elle constitue aussi le point de départ et de référence de toute fonctionnalité disponible sous Eagle.

On prendra donc le réflexe d'y revenir si l'on s'est quelque peu égaré dans les dédales du logiciel.

La fenêtre principale comprend, outre la barre supérieure de menu et inférieure d'information, trois zones d'affichage texte. Examinons la barre de menu.



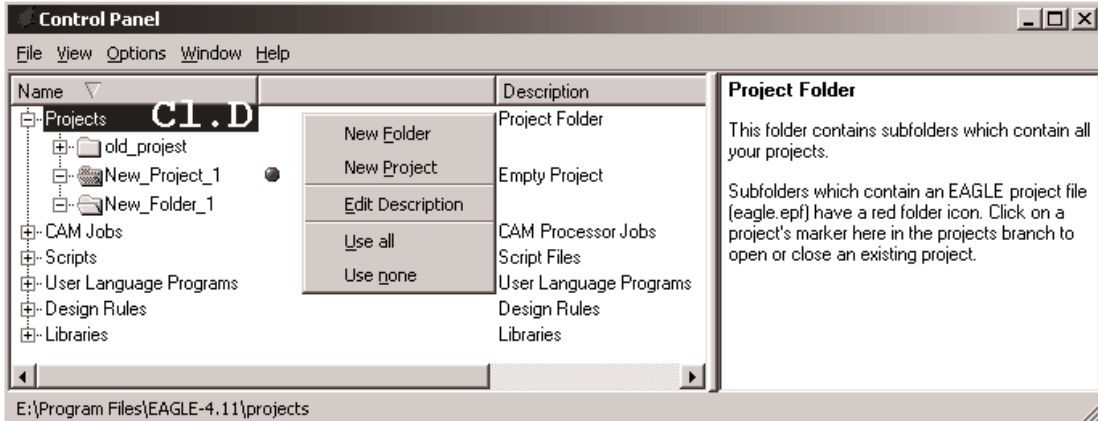
Les trois zones textuelles :

La zone de gauche reprend l'arborescence globale du logiciel. On utilisera dans un premier temps le dossier "**Projects**" et le dossier "**Librairies**". Le premier reprendra dans des sous-dossiers distincts, toutes les informations propres à chaque projet c'est à dire les schémas du projet, la librairie du projet et tout autre fichier utile au projet.

Le dossier "**Librairies**" renferme la définition de tous les composants fournis par défaut avec Eagle, ceux téléchargés sur "<http://www.cadsoftusa.com>" ou "<http://www.cadsoft.de>" ainsi que tous ceux qui ont dus être créés à l'occasion des différents projets.

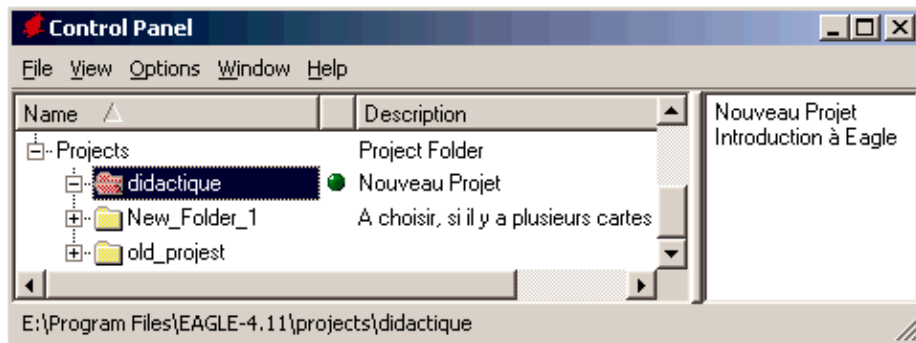
Les zones de droite donne une description de la sélection de gauche; crée par la commande "Description". La colonne "Description" reprend la première ligne de cette description, la suite étant dans la fenêtre d'extrême droite.

Nous avons besoin de ce dossier pour créer notre premier projet.



Par exemple :

Par un Cl.D sur "Projects", on peut créer un dossier "New Folder" pour une application nécessitant plusieurs cartes électroniques distinctes ou un projet "New project" pour créer une carte indépendante.



Remarquez que l'icône des dossiers est jaune contre rouge pour le dossier du projet.

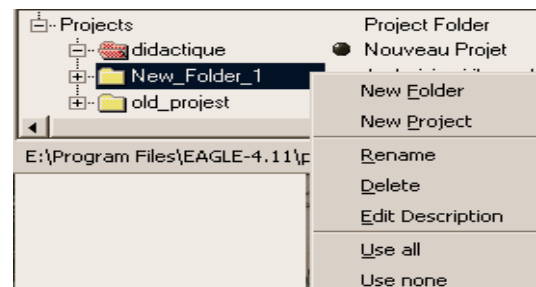
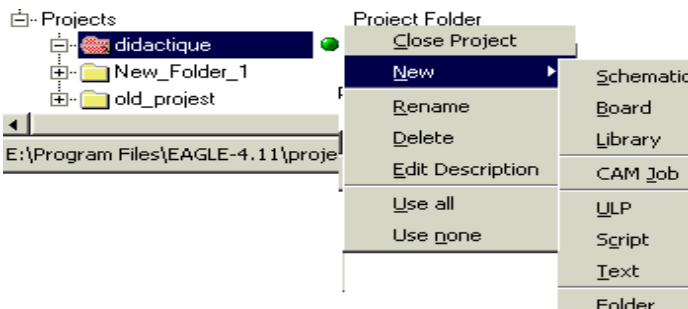
De plus, un point vert à droite du projet indique qu'il est ouvert, alors il ne peut être ni renommé ni supprimé. La commande "Edit Description" permet de décrire le projet ou le dossier.

La description apparaît dans la fenêtre de droite, tandis que seule la première ligne apparaît dans la colonne de droite dans la fenêtre de gauche. Créons notre projet "didactique".

Les commandes "Use all" et "Use none" définissent si, pour le projet ouvert, on va utiliser toutes les bibliothèques, ou si, au contraire, on créera sa propre bibliothèque, ce qui est vivement conseillé et facile à faire. Néanmoins, nous n'activerons que les bibliothèques utilisées.

Un Clid sur "didactique" ne donne pas le même menu contextuel que pour un dossier

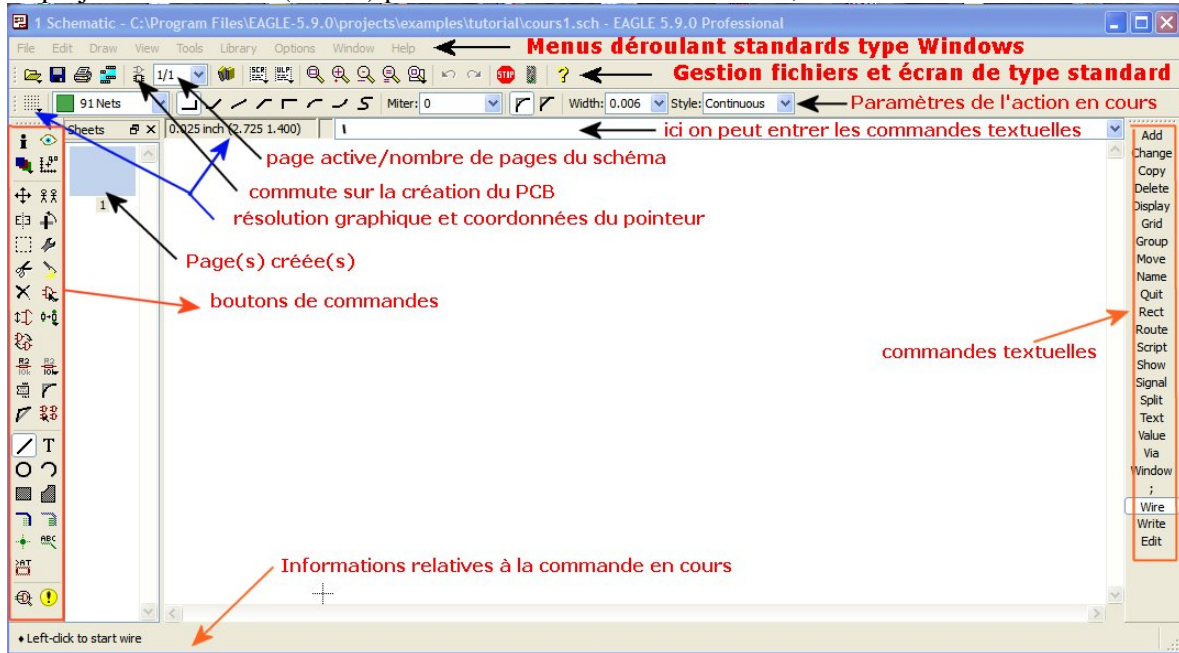
A l'exception des options CAM Job, ULP qui seront vues ultérieurement, ces menus parlent d'eux-mêmes.



2.2 L'éditeur de schéma ou Schematic editor

2.2.1 L'écran d'accueil

Le projet doit être actif (OPEN) pour créer un New → Schematic, nous obtenons.



On nommera ce futur schéma lors de la première sauvegarde.

Un Cl.D sur une des barres d'icônes permet de sélectionner celle que l'on veut conserver à l'affichage. (Voir aussi la commande "User Interface" du menu option du C.P.)

Les fonctionnalités des "commands buttons" et des "commands text" sont identiques et peuvent toutes être introduites textuellement dans la ligne de commande.

Les barres des actions et des paramètres sont contextuelles, elles se complètent d'icônes relatifs aux options disponibles avec la commande active.

Les icônes d'action présents sur la figure correspondent à la cmd NET.

On n'y retrouve les actions élémentaires de Windows :



à droite : ouverture, sauvegarde et impression de fichier (voir menu File)



à gauche : la gestion du graphisme (voir menu Window),



undo et redo (inactives si grisées) et l'aide mais aussi

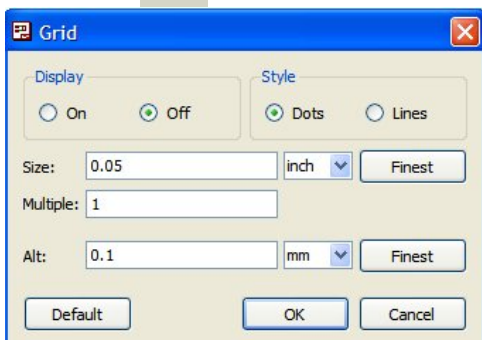


deux commandes "Stop" et "Go" pour interrompre ou lancer une action



commandes que nous verrons plus loin.

L'icône des paramètres écrans permet de régler l'apparence, le pas et les unités de l'affichage.



Display permet d'afficher ou non le style points/lignes pour la grille de positionnement (et de connexions !).

Size fixe la résolution graphique (et physique).

Multiple divise la grille affichée par le nombre spécifié.

Alt fixe une grille alternative activée par la touche "alt" du clavier, très pratique pour passer des mesures décimétriques aux mesures anglo-saxonnes par exemple lors du dessins de boîtiers.

Il est suivi des icones relatifs à la cmd en cours, ici des paramètres pour la commande "wire"



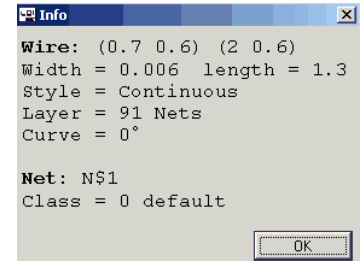
Avant de concevoir notre premier schéma, examinons les commandes disponibles.

Les commandes sont reprises sous formes littérales, avec en gras le minimum à écrire en ligne de commande pour activer la commande textuellement

2.2.2 Les commandes graphiques



Information : fournit toutes les informations relatives à l'objet. On sélectionne l'icône, puis l'objet, ici le signal N\$1

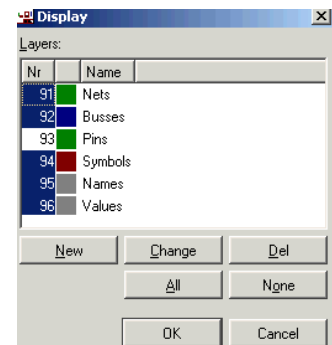


Display: cette commande fait apparaître un des principes de l'éditeur graphique : à chacun sa couche (**layer**), autrement dit, il faut voir le schéma comme une superposition de plusieurs transparents.

Comme on le sait, un schéma est composé de symboles interconnectés.

Un symbole (**Symbol**) représente un composant électronique qui possède donc des points de connexion (**pin** sur le schéma **pad** sur le PCB), un nom (**Name**) et une valeur (**Value**). Les liaisons peuvent se représenter par de simples "fils" (**Nets**) ou par des bus (**Busses**). On remarquera que les couches sont numérotées, en vidéo inverse signifie que la couche est affichée. On arrange donc l'affichage comme on l'entend pour une meilleure lisibilité, ou une impression ciblée.

En ligne de commande "**Di -pi na -96**", cache les couches **Pins** et **Value** (96) et affiche la couche "**Name**".



Show : montre en surbrillance les éléments apparentés du schéma, par exemple l'ensemble des liaisons de même nom (par exemple : **Sh +Vcc**) ou encore toutes les portes d'un même circuit intégré. Les informations apparaissent dans le bandeau inférieur



Mark : permet positionner une origine d'axes auxiliaires, ce nouveau repère fournit des coordonnées rectangulaires et polaires entre parenthèses, utile pour les boîtiers des composants ou dans le PCB (positionnement, perçage ...). On enlève ce repère avec la commande "**GO**" juste après "**Mark**"



Move : déplace les objets, Cl.G pour sélection ou déposer, Cl.D pour rotation. Remarque, un Cl.D avant Cl.G sélectionne le dernier groupe défini.



Copy : réalise une copie de l'objet pointé par Cl.G (pas de groupe possible)



Mirror : réalise une symétrie orthogonale dont l'axe vertical passe par le curseur



Rotate : Réalise une rotation trigonométrique centrée sur le curseur. Cl.G pour sur un objet, Cl.D pour le groupe prédéfini



Group : Permet de grouper des éléments pour les manipuler ensembles. Cl.G pour démarrer le polygone, Cl.D pour le terminer. Le groupe défini passe en surbrillance.



Change : est un menu qui permet de modifier une caractéristique d'un objet, d'un composant, d'un signal, d'un groupe ou du logiciel. Par exemple l'épaisseur des traits, la couche de travail...



Cut : Copie le groupe dans un tampon, l'origine du groupe est pointée par un Cl.G ou centrée si on lance la cmd GO



Paste : copie l'origine du tampon à l'endroit du curseur.



Delete : efface l'objet sous le curseur par Cl.G, ou le dernier groupe défini par CL.D
Remarque: vous disposez des cmds Undo(F9) et Redo(F10)

2.2.3 Commandes schématiques

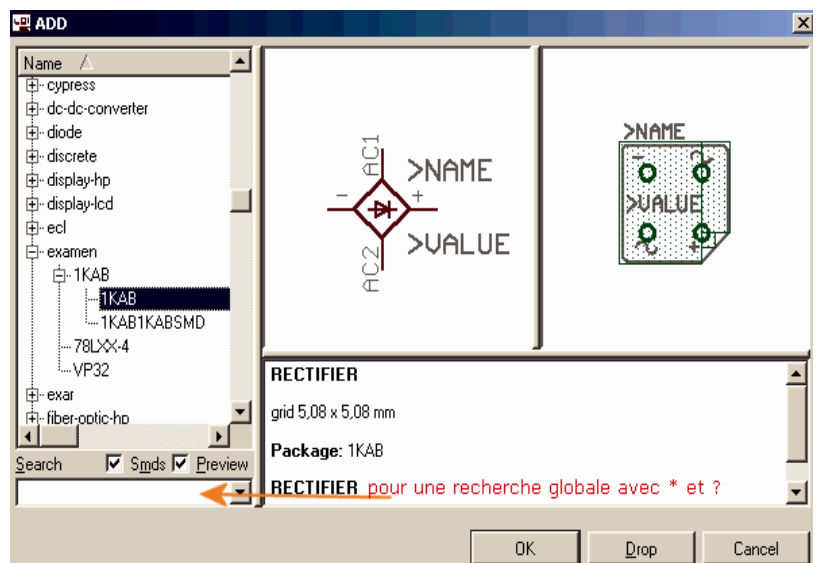


Add : La commande indispensable pour construire le schéma, c'est elle qui permet d'importer les composants au départ des bibliothèques. Celle qui nécessite aussi une bonne préparation pour bonne convivialité.

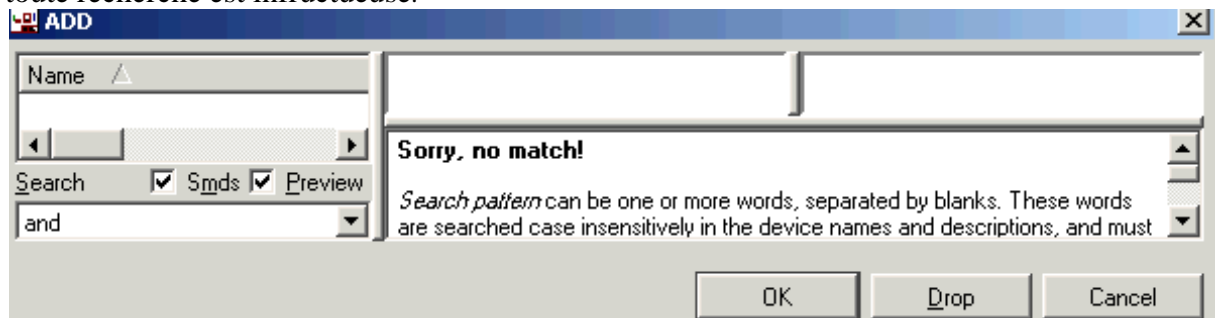
Si, dans le C.P., propriété de bibliothèque par Cl.D, on a choisi "Use all", la cmd "Add" donne la fenêtre suivante :

On y distingue :

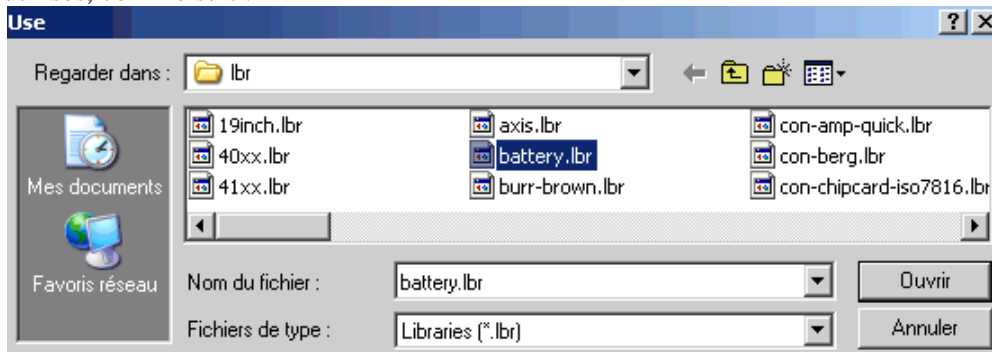
- > L'arborescence des bibliothèques et de leurs composants. Par défaut, on dispose d'environ 200 bibliothèques dont certaines comptent plusieurs centaines de composants. Une recherche imprécise peut donc s'avérer très laborieuse.
- > Le symbole et le boîtier du composant sélectionné
- > Une description du composant
- > Une ligne de saisie pour une recherche par mot clé, (* et ? acceptés).
- > **Drop** : retire la sélection de l'affichage (par opposition à la commande "Use").



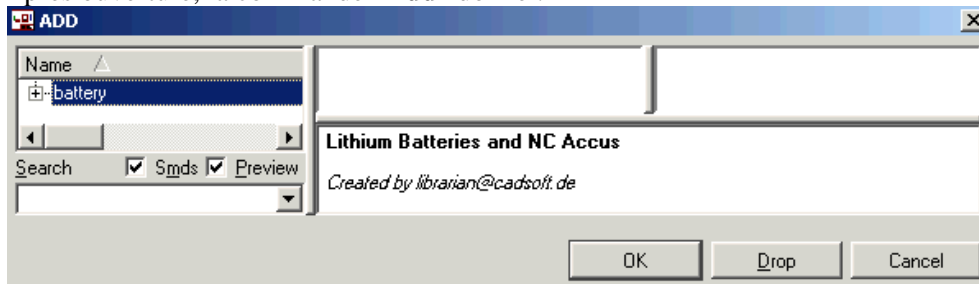
Par contre, si on a choisi la commande "Use none", la cmd "Add" donne des zones vides et toute recherche est infructueuse.



Il nous suffit alors d'utiliser la commande "Use" du menu déroulant "Library", ou encore de taper USE en ligne de commande pour sélectionner uniquement les bibliothèques comprenant les composants utilisés, comme suit :



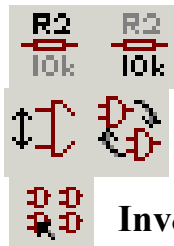
Après ouverture, la commande "Add" donne :



On procède de même pour chaque bibliothèque requise dans la réalisation du schéma.

Une méthode portable sera examinée après l'étude détaillée des bibliothèques et des composants.




Il nous reste quelques commandes à voir avant de réaliser un schéma.



Name, Value : qui permettent de spécifier le nom ou la valeur d'un composant

Pinswap, Gateswap : permute les accès d'une porte ou les portes d'un circuit intégré. Commandes valable pour des objets ou accès de même fonctionnalité.

Invoke : vérifie si toutes parties d'un composant sont utilisées (QUAD NAND)

Enfin pour les connexions, nous avons la commande **Net**  pour des connexions simples, à ne pas confondre avec le **Wire**  qui ne fait pas de liaison électrique automatique, mais réalise simplement un trait. On verra au cours que la commande **Bus**  permet de clarifier les schémas à grande densité de connexions.

Junction : permet de confirmer et de visualiser la connexion entre deux signaux qui se croisent et qui sont effectivement en liaison. Par exemple, ci contre N\$5 croise N\$1 sans liaison, Il est vivement conseillé de pratiquer de la sorte et de placer une jonction là où elle existe. Ce n'est pas toujours automatique.



Label : CIG sur le net ou le bus

ATtribute permet de spécifier des noms et valeurs différentes de celles affichées

Erc : Electrical Rule Check- crée un fichier d'extension .erc reprenant les erreurs présentes dans le schéma

Affiche le rapport ERC

Les autres icônes ne sont pas utiles pour l'instant.

3. Création d'un schéma

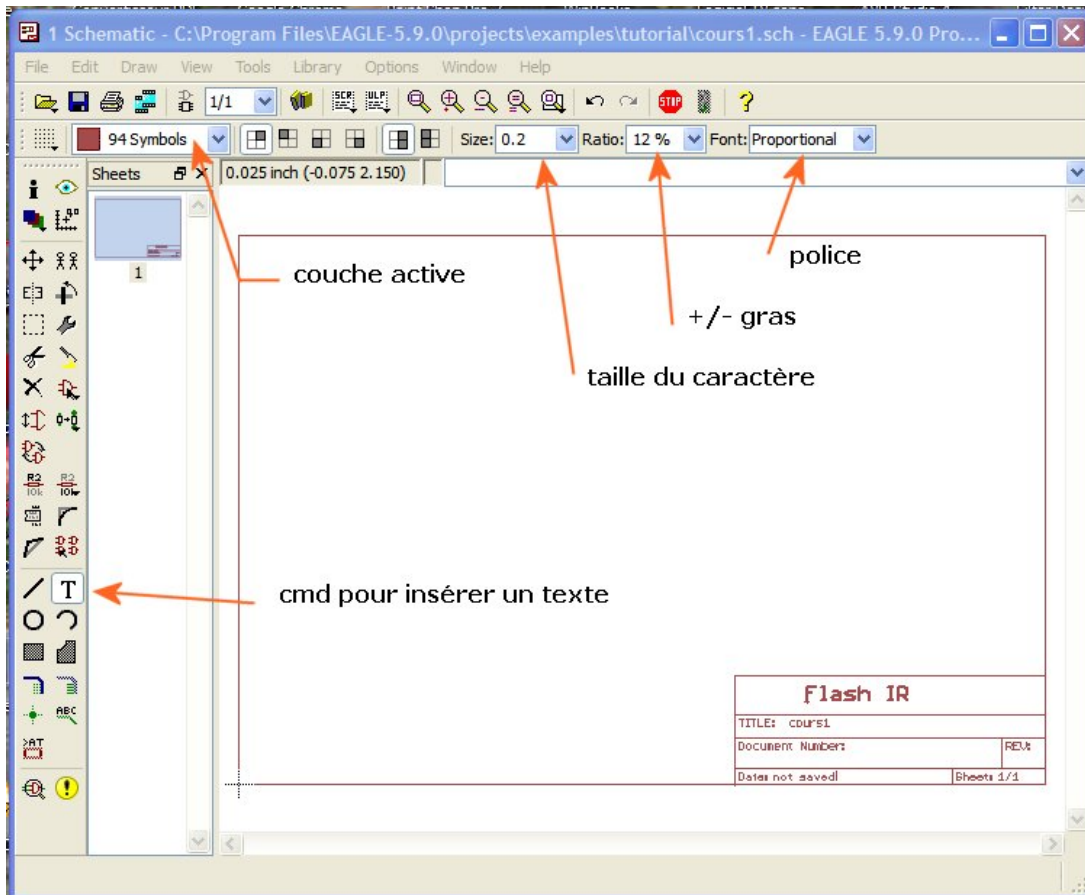
3.1 Mise en page

La première chose à faire, c'est préparer son espace de travail, c'est-à-dire délimiter la zone graphique par un cadre et une cartouche. La **Lbr "FRAME"** nous y aidera :

Dans le **CP**, → **CLD** sur le dossier **Lbr** → **USE NONE**; ensuite **CID** sur la **Lbr "FRAME"** → **USE**, idem pour les **Lbr LED, LINEAR** et **RCL**.

Revenons à notre schéma, avec la cmd **ADD**, ajoutons le composant "DINA4_L" de la **Lbr FRAME**, nous veillerons à placer le coin inférieur gauche à l'origine.

Avec la cmd **TEXT**, on ajoutera le nom du montage dans la cartouche. On obtient :



Les variables ">**DRAWING_NAME**, >**SHEET**, >**LAST_DATE_TIME**, >**PLOT_DATE_TIME**" permettent d'insérer le nom du fichier, le numéro de page, les dates de dernière modification ou impression,

Remarque: Puisque nous savons que le composant DINA4_L est dans la **Lbr** on peut aussi écrire en **cmd** : "**AD DINA4_L@FRAMES.lbr**" même si la **Lbr** n'est pas en active.

On peut aussi écrire pour une recherche : **AD DINA4??@FRAMES.lbr**, mais la recherche n'aura lieu que dans les **Lbr** actives (in use).

Nous pouvons maintenant utiliser la cmd **ADD** pour importer les composants connus :

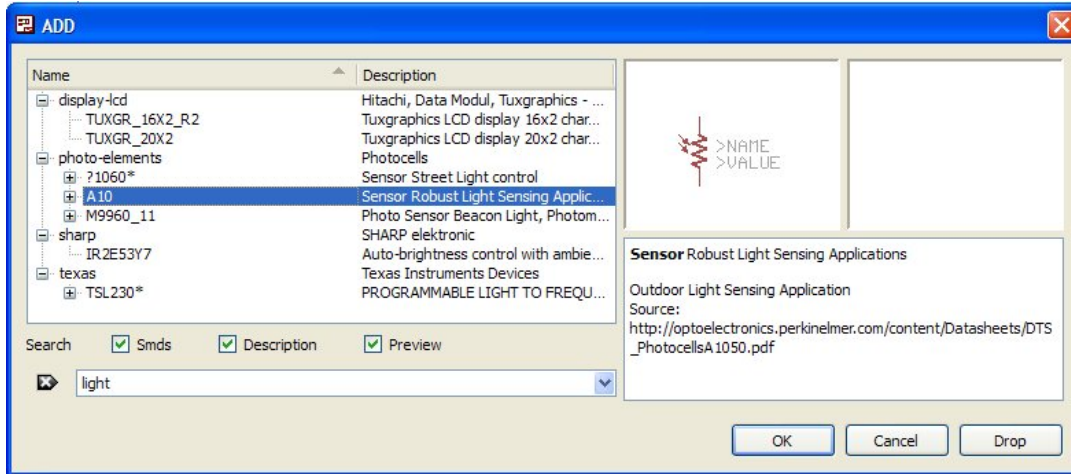
- de la **Lbr LED** : une LED 5mm
- de la **Lbr RCL** : 3 résistances R-EU 0207/10 et un C-EU025 -024x044
- de la **Lbr LINEAR** : un LM555N

- Enfin, il nous reste une LDR et un connecteur pour une pile 9V à trouver, et nous ignorons s'ils sont définis et si oui, dans quelle **Lbr**.

Pour gagner du temps, nous allons intuitivement utiliser l'outil de recherche de la cmd ADD, après avoir pris soin d'activer "USE ALL" pour les **Lbr** dans le CP.

Pour la LDR:

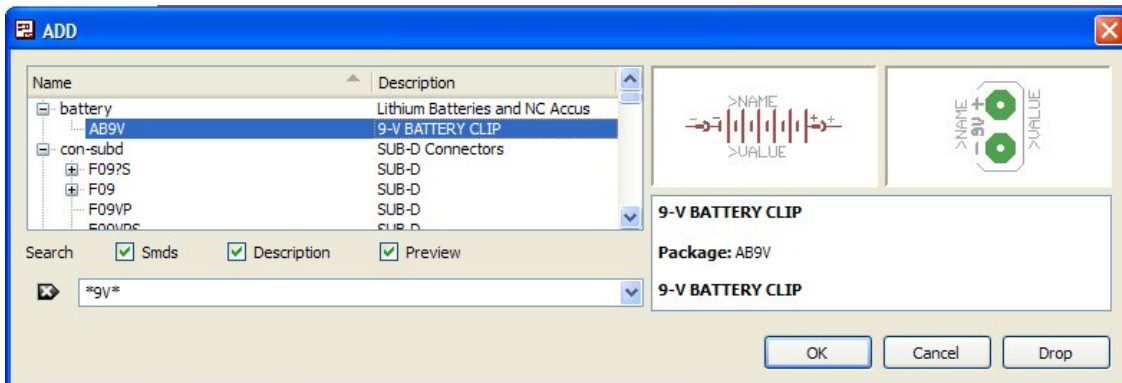
une recherche avec "**LDR**" produit "**Sorry, no match!**", par contre "**light**" produit :



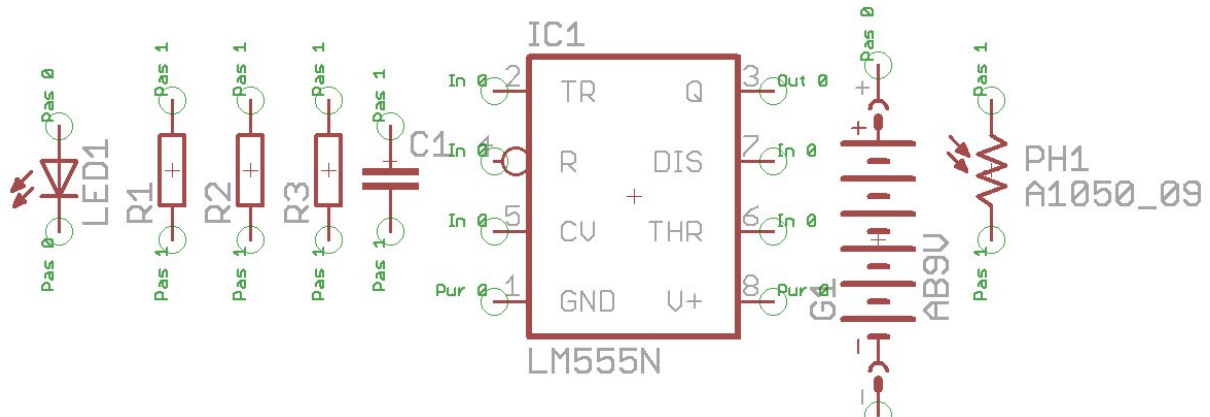
A10-09 nous conviendra

- Pareillement, pour le connecteur 9V, une recherche avec "9V" nous sommes dirigés vers la **Lbr** "TRAFO", qui n'apporte rien.

Par contre en tapant "***9V***", nous obtenons...




ce que nous cherchons. En résumé nous obtenons dans nous schéma

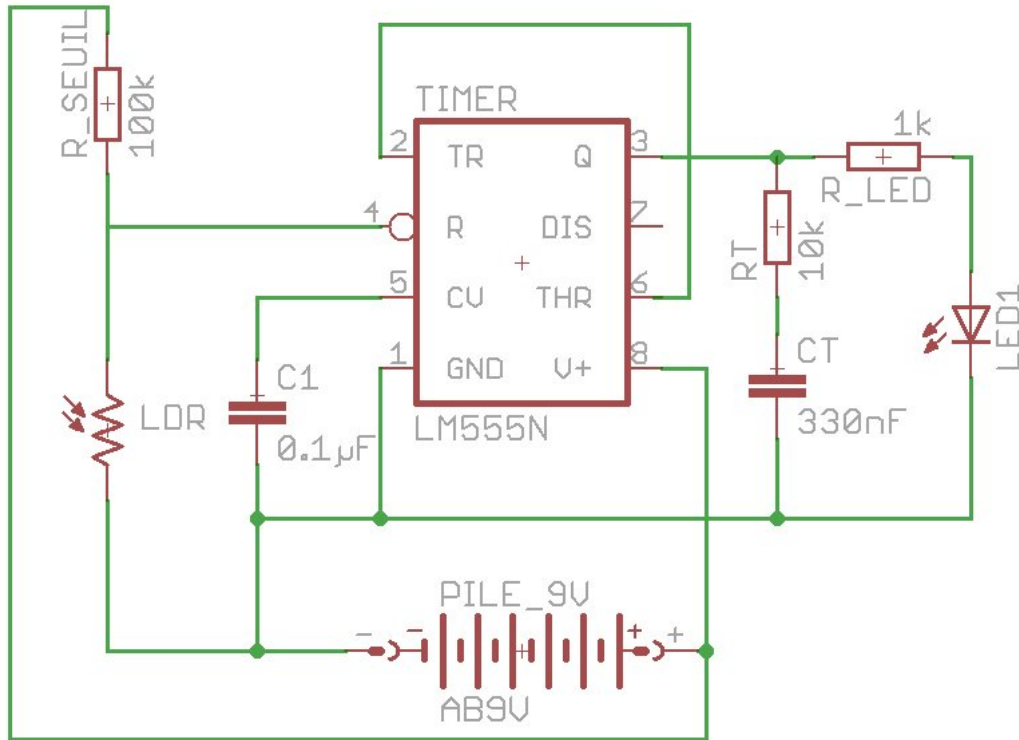


- Ici, la couche PIN est activée, on y voit la zone de contact ainsi que les caractéristiques des pins: nature, direction, swapelevel,.. qui seront étudiées avec les librairies.

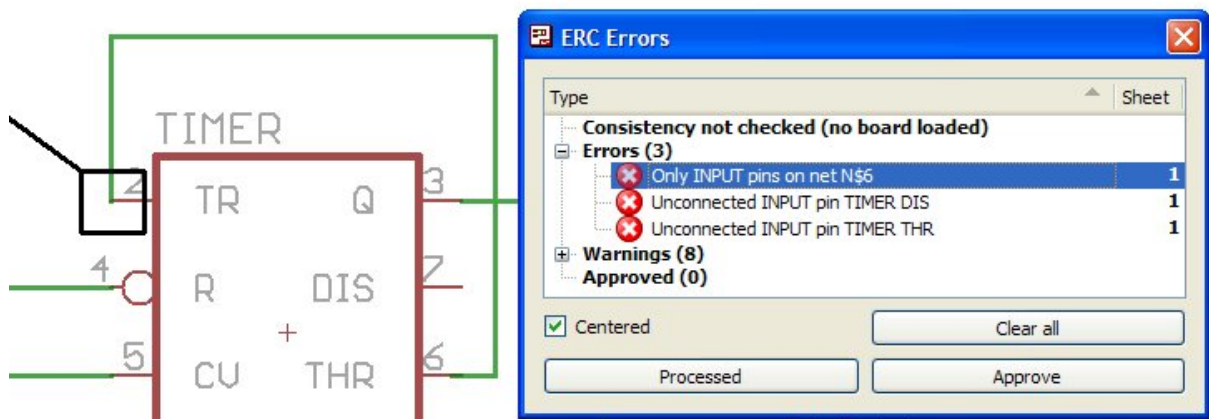
Il fallait une R et un C en plus, oublié dans l'effervescence du projet

La cmd "COPY", soit le bouton  appliqué une fois sur R1 et une fois sur C2 permet d'ajouter les composants manquants. Nous avons maintenant 9 composants, le compte y est. Renommons-les RT, CT, LDR, LED, R_LED, PILE_9V, R_SEUIL_TIMER. Cela rendra le schéma plus lisible. Fixons les valeurs.

- Disposons ensuite les composants en tenant compte des connexions à établir.
On pourrait par exemple obtenir le schéma suivant:



La cmd ERC nous indique que des connexions ne sont pas réalisées correctement.



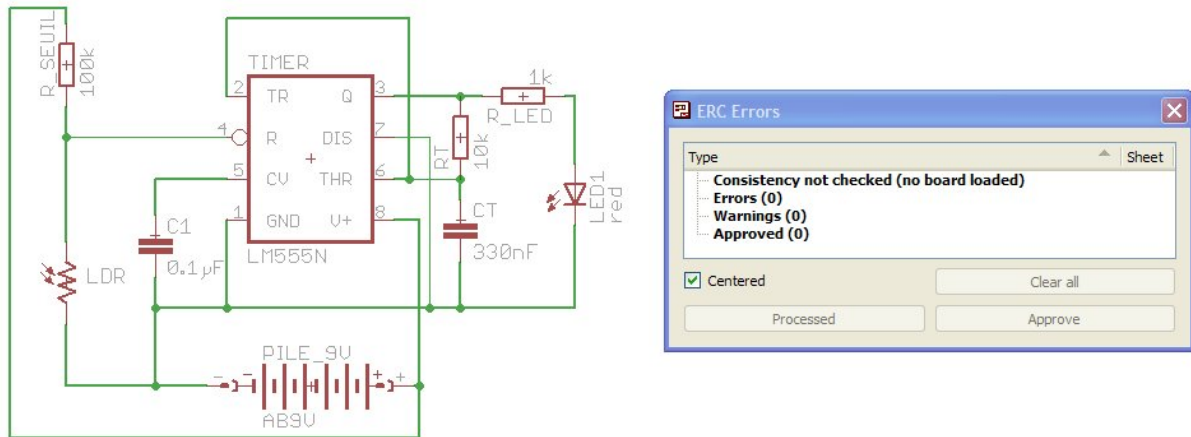
Dans le cas N\$6, il manque une liaison avec Rt_Ct.

La cmd "SHOW" sur ce fil ne mettra pas la pin 6 (THR) du 555 en surbrillance, indiquant ainsi l'absence de connexion sur cette pin.

La cohérence avec le PCB n'est testée car il n'existe pas.

Enfin il reste 8 avertissements (warning) à contrôler et idéalement à éliminer comme c'est impérativement le cas pour les erreurs.

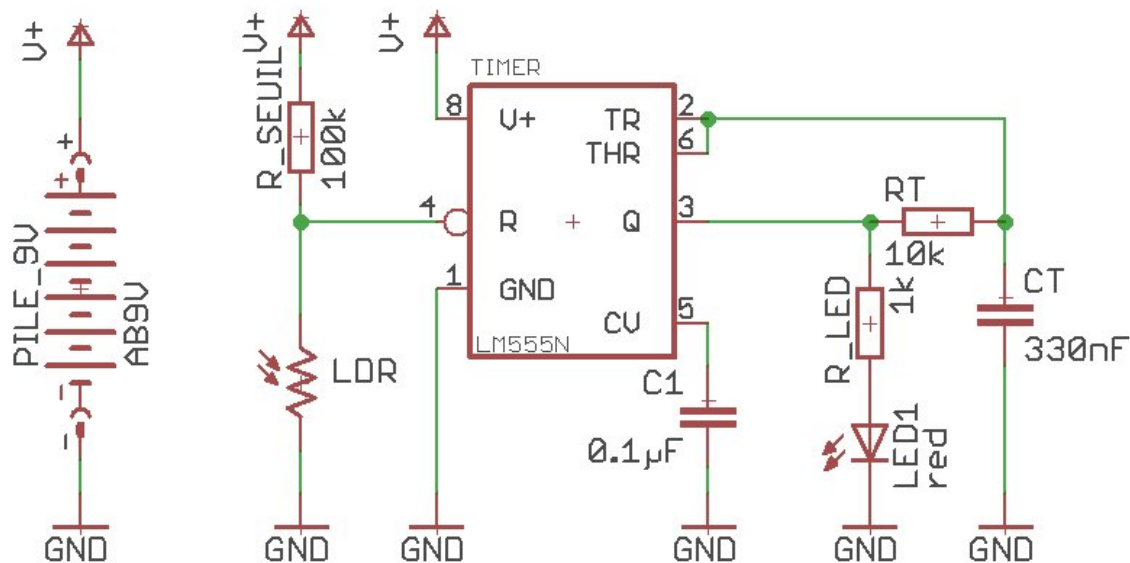
Après ces corrections, le schéma ne présente plus ni d'erreur ni d'avertissement, mais reste néanmoins peu lisible, voyez plutôt les croisements qu'il présente:



Une des causes est l'alimentation 9V qui est centralisée sur le connecteur de la pile, une alternative consisterait à utiliser des symboles "SUPPLY" disponible en librairies et connecté automatiquement entre eux (ceux de même nom) mais de façon invisible

En outre, dans le chapitre consacré à l'étude des librairies, nous verrons comment modifier le symbole pour l'adapter à la lisibilité de notre schéma.

On peut ainsi arriver à ce schéma, nettement plus lisible:



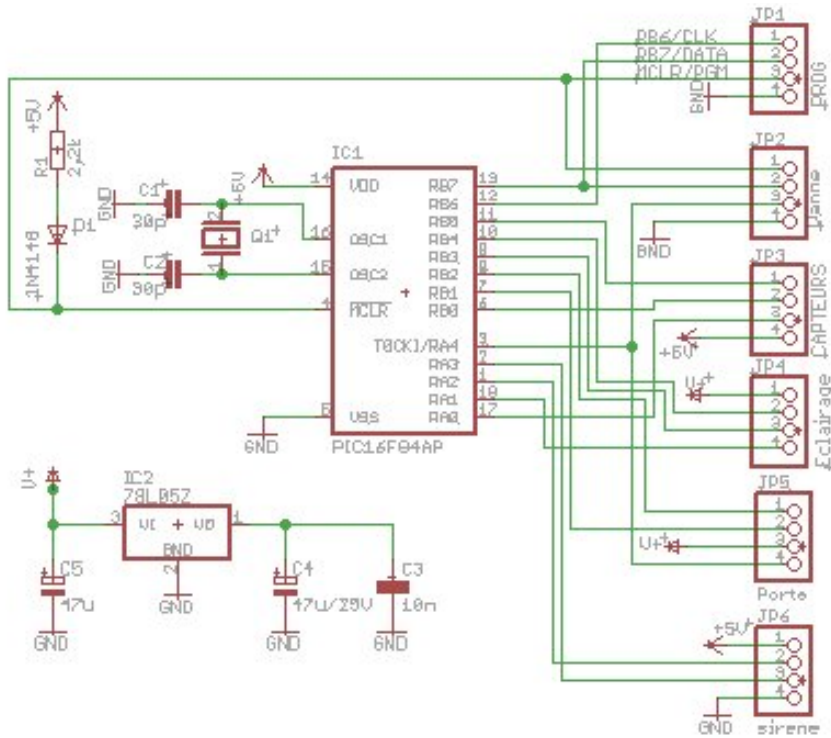
ERC: No errors/warnings. +

Dans l'application présente, on a pu modifier le symbole pour atteindre une meilleure lisibilité du schéma.

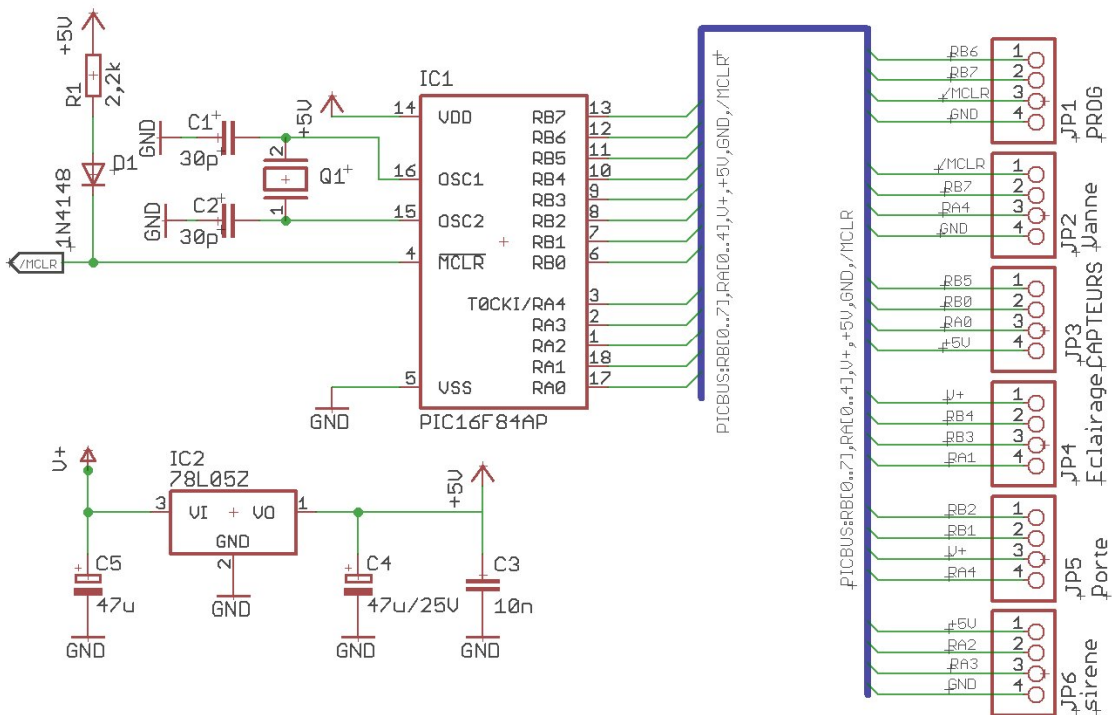
Ce n'est pas toujours possible, la plupart des application à μC ne permettent pas ce type d'arrangement, simplement parce les lignes d'I/O ont souvent plusieurs fonctions et sont donc connectées à plusieurs **pin**.

En voici un exemple.



Le schéma ci-dessous




peut devenir

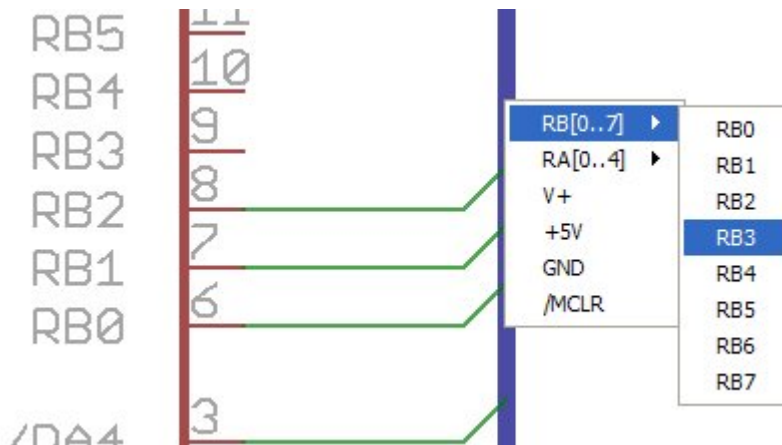


On a utilisé la cmd "SMASH" pour positionner le nom et la valeur (rôle) des connecteurs. Une croix sous la première lettre indique alors l'origine du texte. Sft+Smash annule la commande

Le bus est dessiné avec la cmd  puis nommé avec la cmd NAME  puis " PICBUS:RB[0..7],RA[0..4],V+,+5V,GND,/MCLR" en ligne de cmd. [x..y] permet de spécifier un groupe de signaux (données, adresses...) de même nature indexées de x à y. Le nom PICBUS est symbolique et donc optionnel

Pour router un signal du bus, sélectionner la cmd NET  puis CIG sur le bus


Tous les signaux constituant le bus s'affichent et il suffit de sélectionner avec la souris le signal que l'on veut tracer.



l'on veut tracer.

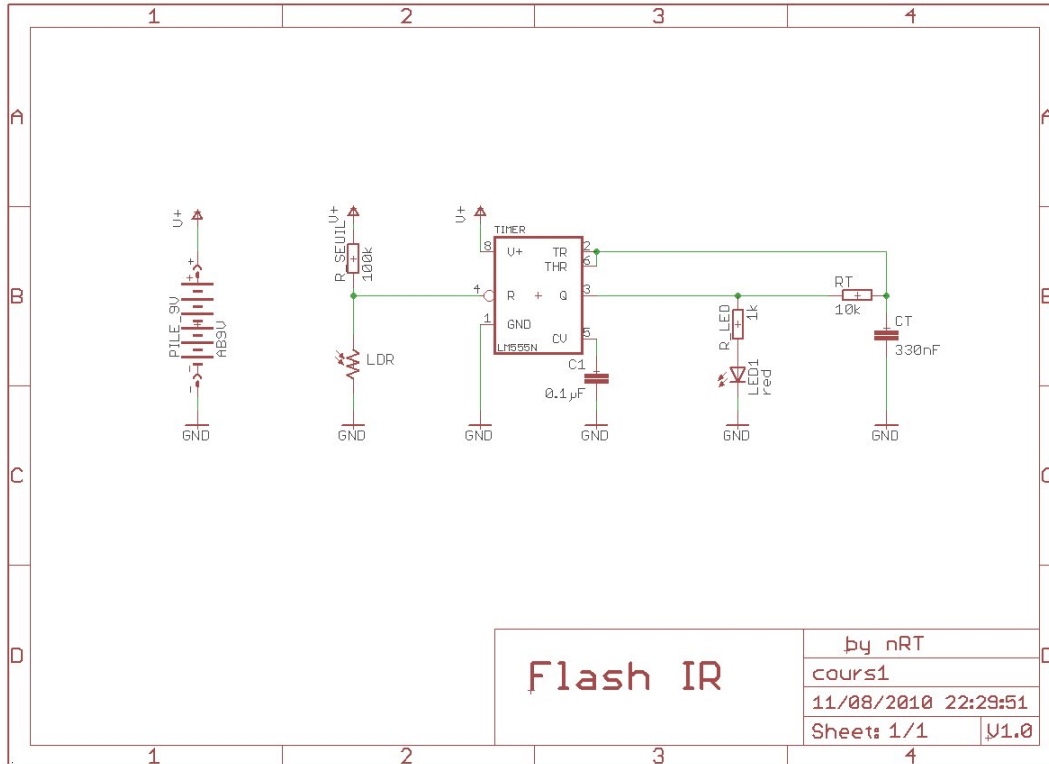
Remarque:

On peut partir du bus ou arriver au bus dans le tracé, la sélection du signal se fait soit au départ soit à l'arrivée.

L'étape suivante consiste à labéliser les signaux avec la cmd "LABEL" , on peut ensuite afficher le nom du net qui part du bus, CLG sur la trace puis déposer le nom en utilisant éventuellement la grille alternative en pressant la touche "Alt" du clavier pendant le déplacement du nom.

Cette méthode demande peut-être plus de temps mais elle impose de la rigueur et produit un résultat nettement plus lisible.

Mais revenons-en à notre projet FLASH IR, comme le schéma est simple, on peut changer de frame, et passer au format A5, avec une localisation possible des composants (voir les ULP). En ayant complété tous les champs, après enregistrement, nous obtenons:



Passons maintenant au circuit imprimé avec le bouton 

Pour la suite, consulter le tutorial PCB.