

# Conception

J'ai généré une disposition plus ou moins idéale pour la disposition des touches en fonction de la forme de ma main sur ergopad : <https://pashutk.ru/ergopad/>

On vas tenter de générer un pcb depuis un logiciel prévu pour ça : ergogen le logiciel demande nodeJS, un interprète javascript que je ne connais pas et qui fut plutôt laborieux à installer sur un linux, on installe ensuite ergopad. (J'ai par la suite trouvé une version web beaucoup plus pratique et qui ne nécessite pas de mettre les mains dans le cambouis : <https://ergogen.cache.works/#>) un clavier normal a en general 26 touches de lettres, 22 pour les chiffres et caractères spéciaux puis les touches de control (2 alt, 2 ctrl, 2 maj, vermaj, retours windows, tabb, dell, espace), la croix directionnelle, les 12 touches de fonction, le pavé numérique et éventuellement des touches de macro. Tout ça pour un total de 88 touches, 56 réellement indispensable.

config des pins ici : <https://github.com/benvallack/ergogen>

On ne possède que 5 doigts qui ont entre 3 et 4 positions confortables pour une touche chacun (8 pour l'index et 3 pour le pouce), un total de 44 touches possibles au total. On va tenter de réduire au maximum le nombre de touches nécessaires par exemple en mettant différentes organisations de clavier différentes interchangeables avec une touche. ref :

<https://www.youtube.com/watch?v=8wZ8FRwOzhU&t=132s>

Switch brown (retours tactiles mais silencieux) <https://fr.aliexpress.com/item/32899833333.html>  
[https://fr.aliexpress.com/item/1005001864766812.html?spm=a2g0o.productlist.0.0.6b584a82GRTs8m&algo\\_pvid=afcbc6e4-3e33-4a80-9bb9-defdb27c54b4&algo\\_exp\\_id=afcbc6e4-3e33-4a80-9bb9-defdb27c54b4-1&pdp\\_ext\\_f=%7B%22sku\\_id%22%3A%2212000017909409069%22%7D&pdp\\_npi=2%40dis%21EUR%21%2115.42%21%21%21%21%21%402100bb5116531312875265836e5f85%2112000017909409069%21sea](https://fr.aliexpress.com/item/1005001864766812.html?spm=a2g0o.productlist.0.0.6b584a82GRTs8m&algo_pvid=afcbc6e4-3e33-4a80-9bb9-defdb27c54b4&algo_exp_id=afcbc6e4-3e33-4a80-9bb9-defdb27c54b4-1&pdp_ext_f=%7B%22sku_id%22%3A%2212000017909409069%22%7D&pdp_npi=2%40dis%21EUR%21%2115.42%21%21%21%21%21%402100bb5116531312875265836e5f85%2112000017909409069%21sea)

carte : <https://fr.aliexpress.com/item/4000991785942.html> (je vais finalement utiliser une carte promicro car déjà implémentée dans le logiciel qui génère le pcb)

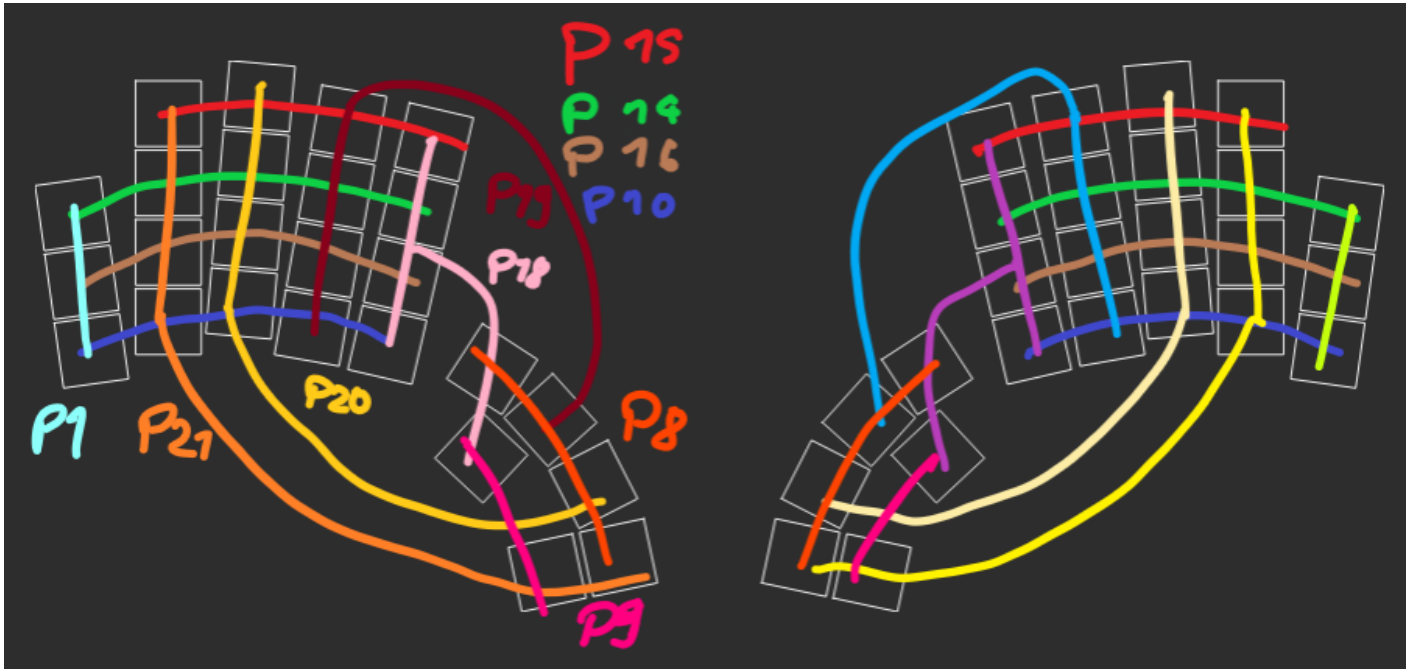
diode : <https://fr.aliexpress.com/item/32947233235.html>

Bépo la config de touche

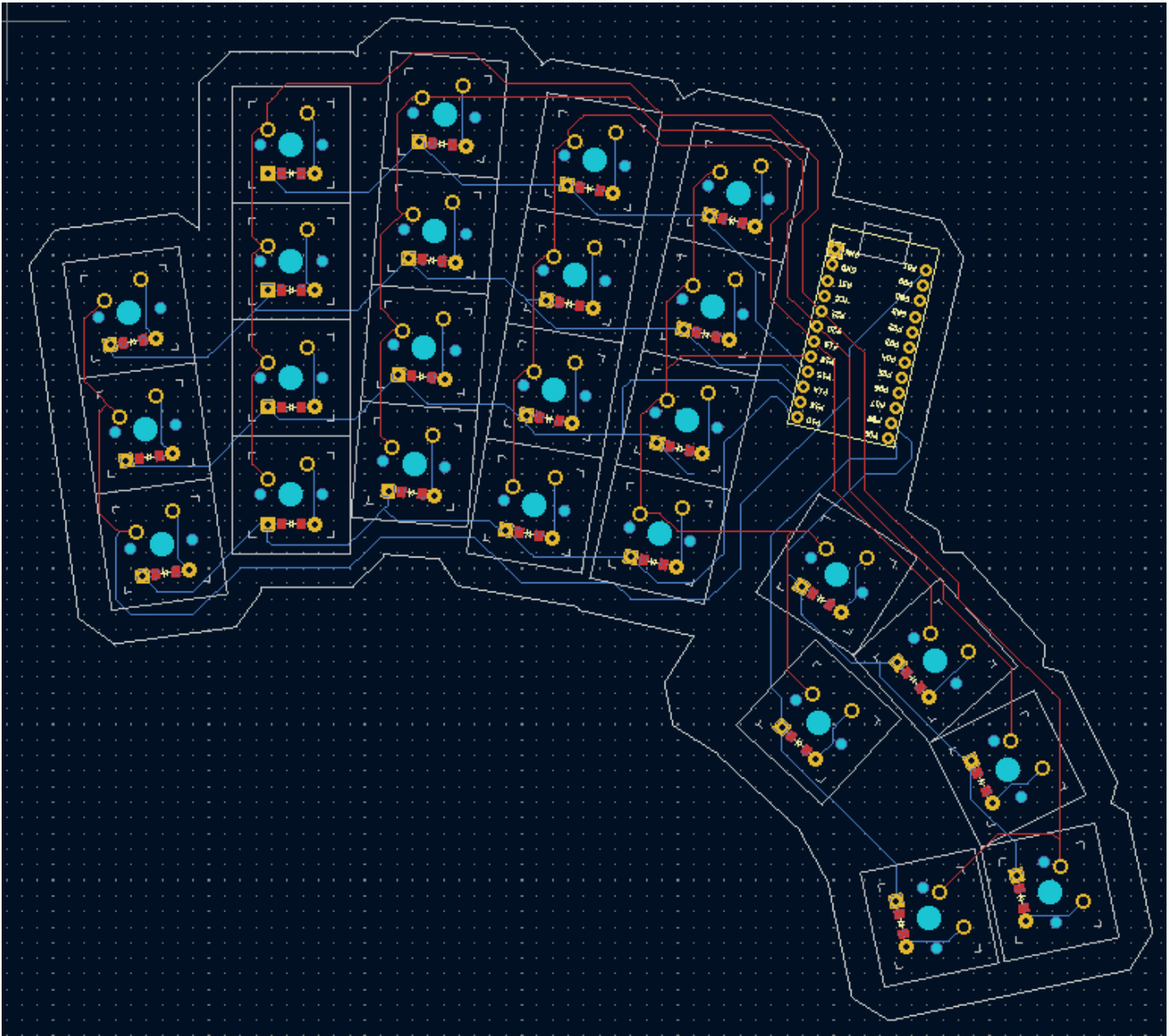
autres exemples : <https://github.com/benvallack/ergogen/blob/master/config.yaml>

<https://github.com/benvallack/ergogen/blob/master/config-flipper.yaml>

après un passage (long et fastidieux) sur ergogen on obtient ca (le code est en dessous) :



J'ai rajouté par dessus en couleur comment je voulais relier électroniquement les touches, il faut également rajouter une diode à chaque touche pour éviter une contre réaction et pouvoir appuyer sur plusieurs touches en même temps, c'est également géré par le logiciel. on obtient à la fin un fichier Kicad que nous n'avons plus qu'à dessiner les pistes à la main en reliant les composants déjà placés et reliés par des traits (définis dans ergogen également grâce à la condition "net"). Et voilà le résultat sur Kicad :



Il ne reste plus qu'à le graver au fablab sur la technodrill ou via des services de gravures proposées en ligne.

Je l'ai fait graver au fablab avec l'aide de Manuel Bouyer. On est d'abord passé par un logiciel permettant de faire le tracé puis on l'a envoyé directement sur la machine après avoir fait les réglages physiques nécessaires comme la mise à niveau du plateau en fonction de l'épaisseur de la plaque. (voir le guide d'utilisation sur le wiki). Pour ce circuit on a besoin d'une plaque double couche, on va donc avoir besoin de retourner la plaque pour faire la gravure des deux cotées. On a utilisé 3 fraises différentes, une pour découper les contours et les trous pour les touches, une pour les trous des pins et une dernière pour la gravure. A cause de problème de calibration, certaines pistes ne sont pas isolées de la masse, j'ai donc terminé de découper la piste au scalpel à l'aide de la loupe binoculaire.

Je me suis rendu compte un fois terminé que la carte scencée se positionner sur la carte ne correspondait pas avec celle du PCB, ce n'est pas un soucis, j'ai juste pris des pins coudé pour la souder à la carte.

Par ailleurs le schéma exploite les deux couches qui doivent parfois être reliées, malheureusement la technodrill ne permet pas de faire ceci, je vais donc devoir souder les touches des deux côtés de la plaque. C'est un peu périlleux car on risque d'abîmer les touches et on doit systématiquement vérifier la conductivité électrique à chaque soudure mais totalement faisable. On soude aussi les diodes et on coupe les pattes qui dépassent.

Par la suite, il nous faut coder la carte, j'ai opté pour une configuration Bépo avec deux autres layers pour les caractères spéciaux mais je ne parviens toujours pas à programmer la carte correctement. Je compte ajouter un bouton pour changer de mode plus facilement mais je vais sûrement avoir besoin de rajouter une deuxième carte pour plus de touches.

Je vais ensuite imprimer des touches en 3D, avec une précision maximale pour être plus agréable sous les doigts. J'ai imprimé un premier prototype qui me semble concluant. Reste le problème de comment inscrire les lettres sur les touches. Soit renforcé directement dans la touche à l'impression, soit peinte. Et ensuite couverte par de la résine. Sinon je pourrais créer un moule à partir d'une impression dans du silicone puis couler de la résine époxy, même si elle sera sûrement un peu lourde.

Enfin on peut tenter de faire une couverture pour la carte électronique, par exemple en bois grâce à la découpeuse laser, faire un revêtement pour protéger la carte de l'eau ou de l'humidité et les composants (résine époxy ? vernis ?), des pieds en caoutchouc antidérapant pour la stabilité et l'orientation du clavier, modifier un câble pour en avoir un plus joli et raccorder avec l'esthétique du clavier, etc....

forme sur ergo : ça nous permet de créer un PCB sur Kicad et un contour à partir de la définition des positions des touches.

## Points

```
zones:
  ptit:
    anchor:
      rotate: 8
    rows:
      1:

      row_net: P10
      mirror.row_net: P10
      2:
```

row\_net: P16  
mirror.row\_net: P16

3:

row\_net: P14  
mirror.row\_net: P14

columns:

o:

key:

column\_net: P1

mirror.column\_net: P11

3autres:

anchor:

rotate : -8

ref: ptit\_o\_1

shift : [22, 5]

rows:

1:

row\_net: P10  
mirror.row\_net: P10

2:

row\_net: P16  
mirror.row\_net: P16

3:

row\_net: P14  
mirror.row\_net: P14

4:

row\_net: P15  
mirror.row\_net: P15

columns:

annulaire:

rotate: 0

key:

column\_net: P21

mirror.column\_net: P11

majeur:

rotate: -5

stagger: 5

spread: 20

key:

column\_net: P20

mirror.column\_net: P9

index2:

rotate: -5

stagger: -5

spread: 20

key:

column\_net: P19

mirror.column\_net: P8

index1:

rotate: -3

stagger: -1

spread: 21

key:

column\_net: P18

mirror.column\_net: P7

poucehaut:

anchor:

ref: 3autres\_index1\_1

rotate: -20

shift: [29.5,0]

rows:

1:

row\_net: P8

mirror.row\_net: P8

columns:

1:

key:

column\_net: P18

mirror.column\_net: P7

2:

rotate: -15

stagger : -3

spread: 21

key:

column\_net: P19

mirror.column\_net: P8

3:

rotate: -15

stagger : -3

spread: 21

key:

column\_net: P20

mirror.column\_net: P9

4:

rotate: -15

stagger : -3

spread: 21

key:

column\_net: P21

mirror.column\_net: P10

pouce2:

rows:

1:

row\_net: P9

mirror.row\_net: P9

columns:

1:

key:

column\_net: P21

mirror.column\_net: P10

anchor:

ref: poucehaut\_4\_1

rotate: 0

shift: [0,-20]

espace:

rows:

1:

row\_net: P9

mirror.row\_net: P9

columns:

1:

```
key:
  column_net: P18
  mirror.column_net: P7
anchor:
  ref: 3autres_index1_1
  rotate: -30
  shift: [32,-24]
mirror:
  ref: poucehaut_4_1
  distance: 50
key:
  footprints:
    - type: mx
      nets:
        from: mx
        to: =column_net
      params.keycaps: true
    - type: diode
      anchor:
        shift: [0, -4.7]
      nets:
        from: mx
        to: =row_net
```

outlines.exports:

```
plate:
  - type: keys
    side: left
    size: 30
    bevel: 5
  - type: polygon
    points:
      - ref: pouce2_1_1
        shift: [0,-12]
      - ref: espace_1_1
        shift: [0,-12]
      - ref: poucehaut_2_1
        shift: [0,-12]
      - ref: poucehaut_4_1
```



```
    shift: [0,-12]
operation: add
- type: rectangle
size: [30, 30]
anchor:
  ref: poucehaut_1_1
  shift: [-14,-35]
bevel: 5
operation: add
- type: rectangle
size: [24, 43]
anchor:
  ref: 3autres_index1_4
  shift: [14, -40]
bevel: 5
operation: add
- type: rectangle
size: [24, 43]
anchor:
  ref: 3autres_index1_4
  shift: [9, -40]
operation: add
- type: rectangle
size: [24, 43]
anchor:
  ref: 3autres_index1_4
  shift: [14, -60]
operation: add
```

```
# - type: rectangle # size: [35, 18] # anchor: # rotate: -90 # ref: 3autres_index1_4 # shift: [16, -2]
# operation: stack # - type: circle # radius: 2 # anchor: # ref: 3autres_index1_4 # shift: [25, -19.5]
# operation: stack
```

```
platemirror:
- type: keys
side: right
size: 30
bevel: 5
- type: polygon
points:
```

- ref: mirror\_pouce2\_1\_1  
shift: [0,-12]
- ref: mirror\_espace\_1\_1  
shift: [0,-12]
- ref: mirror\_poucehaut\_2\_1  
shift: [0,-12]
- ref: mirror\_poucehaut\_4\_1  
shift: [0,-12]

operation: add

- type: rectangle

size: [30, 30]

anchor:

- ref: mirror\_poucehaut\_1\_1  
shift: [14,-35]

bevel: 5

operation: add

- type: rectangle

size: [24, 60]

anchor:

- ref: mirror\_3autres\_index1\_1  
shift: [38, -16]

bevel: 5

operation: add

- type: rectangle

size: [24, 60]

anchor:

- ref: mirror\_3autres\_index1\_1  
shift: [30, -16]

bevel: 5

operation: add

arduino:

- type: rectangle

size: [24, 43]

anchor:

- ref: 3autres\_index1\_4  
shift: [14, -40]

bevel: 5

operation: add

- type: rectangle

size: [24, 43]

```
anchor:
  ref: 3autres_index1_4
  shift: [9, -40]
operation: add
- type: rectangle
size: [24, 43]
anchor:
  ref: 3autres_index1_4
  shift: [14, -60]
operation: add
finalkey:
- plate
- platemirror
- type: keys
side: both
size: 18
operation: subtract
final:
- plate
- platemirror
- arduino
```

## PCBS :

```
left:
  outlines:
    contours:
      outline: plate
      layer: Edge.Cuts
  footprints:
    - type: promicro
      anchor:
        rotate: -90
        ref: 3autres_index1_4
        shift: [25, -19.5]
right:
  outlines:
    contours:
      outline: platemirror
      layer: Edge.Cuts
```

footprints:

- type: promicro

anchor:

rotate: -90

ref: 3autres\_index1\_4

shift: [25, -19.5]

- type: jumper

anchor:

ref: mirror\_3autres\_index1\_2

shift: [-10, 0]

nets:

from: P18

to: P20

both:

outlines:

contours:

outline: final

layer: Edge.Cuts

footprints:

- type: promicro

anchor:

rotate: -90

ref: 3autres\_index1\_4

shift: [25, -19.5]

#ligne L1 = P10 L2 = P16 L3 = P14 L4 = P15 L5 = P8 L6 = P9

#colonnes main gauche C1 = P1 C2 = P21 C3 = P20 C4 = P19 C5 = P18 #colonnes main droites C6 = P7 C7 = P8 C8 = P9 C9 = P10 C10 = P11

Si vous essayez de reproduire ceci et avez des problèmes avec Ergogen, n'hésitez pas a me contacter. J'ai suffisamment galéré avec ce truc pour que maintenant je puisse aider d'autres personnes. [romain.tessier@etu.sorbonne-universite.fr](mailto:romain.tessier@etu.sorbonne-universite.fr)

---

Revision #3

Created 16 August 2022 14:11:05 by Clara

Updated 3 October 2022 13:02:28 by Tessier Romain