

# Objectif : asservissement numérique d'un oscillateur à quartz sur GPS

Introduction

Schéma

Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python

Contexte :

- un oscillateur à quartz est stable sur le court terme mais dérive sur le long terme ( $>1$  s)
- GPS est un système de datation précis fournissant un signal (1 PPS) avec une exactitude de l'ordre de 100 ns toutes les secondes
- **Objectif** : contrôle du quartz cadencant un microcontrôleur sur le 1 PPS de GPS

⇒

- ① architecture générale du système
- ② mise en œuvre du microcontrôleur et composants passifs périphériques (quartz, reset/programmation. communication)
- ③ simulation du filtre (PWM  $\rightarrow$  DAC)
- ④ routage en respectant des contraintes mécaniques
- ⑤ insertion du circuit dans FreeCAD

# Objectif : architecture du système

Introduction

Schéma

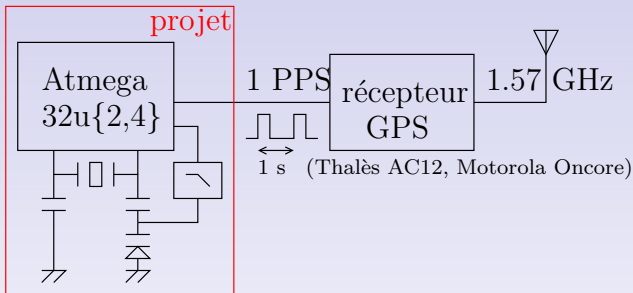
Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python



- 1 microcontrôleur + périphériques
- 2 varicap en parallèle d'un condensateur de pieds pour tirer la fréquence du quartz
- 3 mesure de l'intervalle entre deux impulsions 1 PPS par Input Capture
- 4 pas de DAC  $\Rightarrow$  PWM (rapport cyclique) et filtre passe bas

# Principe de KiCAD <sup>1</sup> <sup>2</sup>

Introduction

Schéma

Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python

Debian/GNU Linux :

```
apt-get install kicad kicad-packages3d kicad-footprints kicad-symbols kicad-templates  
kicad-libraries
```

version 5.1.4

(354 MB de packages 3D !)

aussi fonctionnel sous MS-Windows & MacOS X

Créé en 1992 by J.-P. Charras (IUT de Grenoble), pris en main par le CERN en 2013, soutenu par Digikey ... un outil spécifique par étape

- schéma (Schematic Layout Editor) – *Electrical Rules Check*
- assignation des boîtiers à chaque composant
- exporter netlist qui fait le lien avec le routeur
- Pcbnew (printed circuit board) – *Design Rules Check*
- export STEP/VRML (CAD) & Gerber

---

<sup>1</sup>W. Stambaugh, *KiCad Version 5 New Feature Demo*, FOSDEM 2018, à [https://archive.fosdem.org/2018/schedule/event/cad\\_kicad\\_v5/](https://archive.fosdem.org/2018/schedule/event/cad_kicad_v5/)

<sup>2</sup>W. Stambaugh, *KiCad Project Status*, FOSDEM 2019, à <https://archive.fosdem.org/2019/schedule/event/kicad/>

# Schéma

Introduction

Schéma

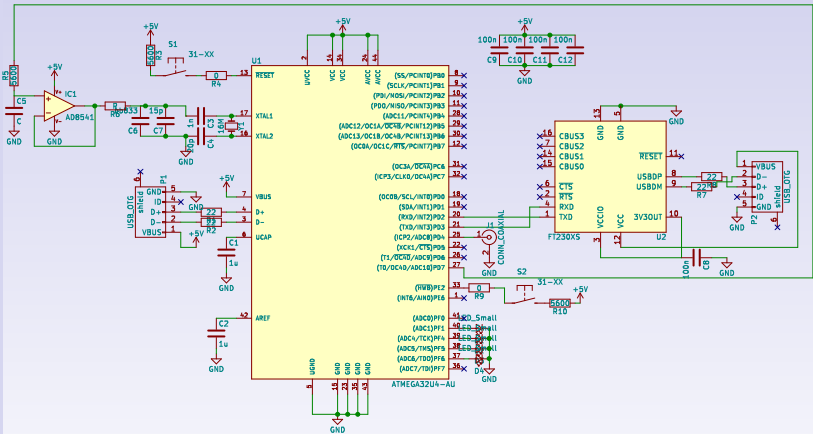
Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python



- “M” ove pour déconnecter les fils, “G” rab pour conserver les fils connectés lors du déplacement
- “R”otate déconnecte les fils
- Not connected (“x”) pour valider Electrical Rule Check

# Ajout d'une empreinte manquante

Introduction

Schéma

**Dessiner sa  
propre empreinte**

Simulation SPICE

Routage

Mécanique –  
insertion dans  
FreeCAD

Script Python

# Simulation SPICE

Introduction

Schéma

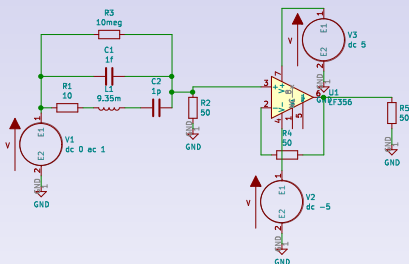
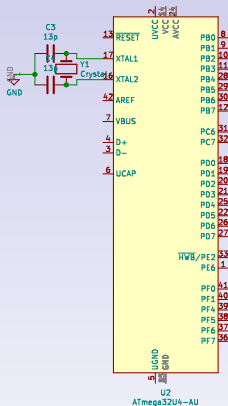
Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python



- Composant spécifique SPICE (Add → pspice → VSOURCE)
- Définition des propriétés de la source dans sa valeur
- Désactiver les composants qui ne font pas partie de la simulation (Edit → Edit Spice Model → Disable symbol for simulation)
- Tools → Simulator → Simulation → Settings
- Run/Stop Simulation → Probe pour sélectionner le point de mesure
- Simulation → Show SPICE Netlist pour récupérer le fichier ngspice

# Simulation SPICE

- Tune pour balayer la valeur d'un composant et voir son impact sur la fonction de transfert



Introduction

Schéma

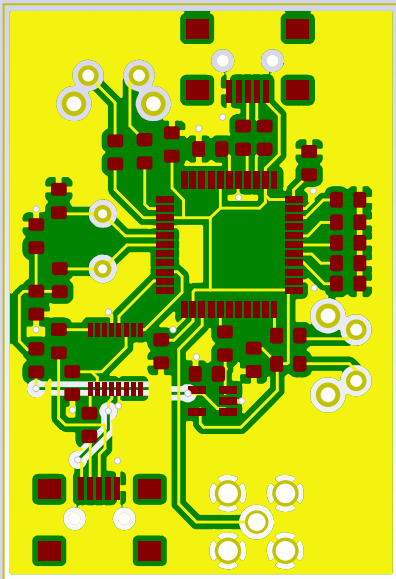
Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python



## Routage

- 2-couches (bottom = plan de masse)
- se rappeler de la fonction de chaque composant dans le schéma lors du placement
- quartz au plus près du microcontrôleur pour éviter l'inductance des pistes
- condensateurs de pieds au plus près du quartz
- condensateurs de découplage au plus près des alimentations (coins)
- surface = prix de fabrication / v.s. nombre de couches
- matrice de vias : click droit sur via et "Create Array"



# Conception mécanique-électronique

Introduction

Schéma

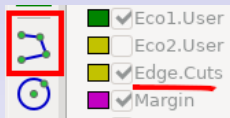
Dessiner sa  
propre empreinte

Simulation SPICE

Routage

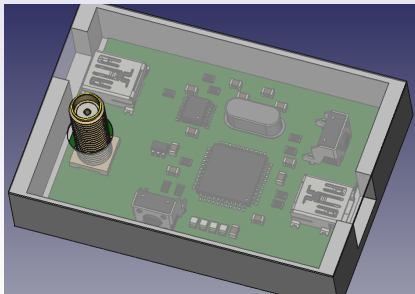
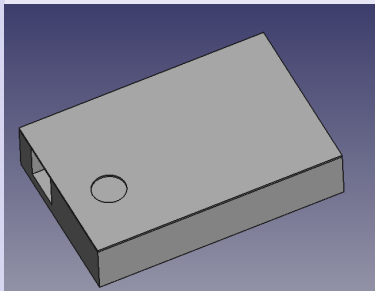
Mécanique –  
insertion dans  
FreeCAD

Script Python



Est-ce que le circuit rentrera dans son boîtier ?

KiCAD exporte un fichier STEP (penser à définir le contour du circuit en Edge.Cuts et Add Graphics Line), FreeCAD importe un fichier STEP :



# Conception mécanique-électronique

Introduction

Schéma

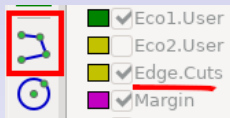
Dessiner sa  
propre empreinte

Simulation SPICE

Routage

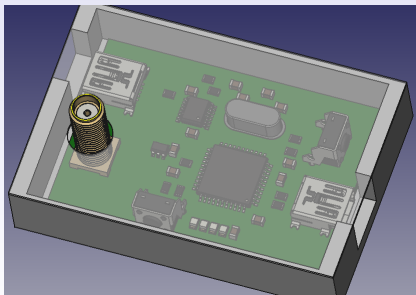
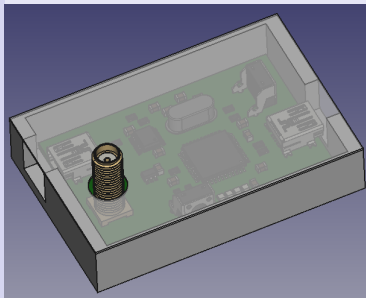
Mécanique –  
insertion dans  
FreeCAD

Script Python



Est-ce que le circuit rentrera dans son boîtier ?

KiCAD exporte un fichier STEP (penser à définir le contour du circuit en Edge.Cuts et Add Graphics Line), FreeCAD importe un fichier STEP :



# Interagir avec KiCAD en Python <sup>3</sup>

Introduction

Schéma

Dessiner sa propre empreinte

Simulation SPICE

Routage

Mécanique – insertion dans FreeCAD

Script Python

Les fichiers stockés par KiCAD sont en ASCII et limpides à lire

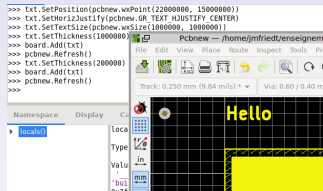
```
(pad 29 smd rect (at 5.7 -0.8) (size 1.5 0.55) (layers F.Cu F.Paste F.Mask))
...
(fp_line (start 1 0.6) (end -1 0.6) (layer F.Fab) (width 0.1))
...
(via (at 37.1 32.2) (size 0.6) (drill 0.4) (layers F.Cu B.Cu) (net 2))
(segment (start 37.1 32.2) (end 37.2 32.2) (width 0.25) (layer B.Cu) (net 2) (tstamp 59845870))
...
```

et donc à éditer avec son outil favori (sed)  
apt-get install python3-wxgtk4.0 pour  
PCBNew → Tools → Scripting Console

Exemple :

```
import pcbnew
board=pcbnew.GetBoard()
newvia=pcbnew.VIA(board)
newvia.SetLayerPair(pcbnew.PCBNEW_LAYER_ID_START, →
↳pcbnew.PCBNEW_LAYER_ID_START+31)
newvia.SetPosition(pcbnew.wxPoint(15000000,15000000))
newvia.SetViaType(pcbnew.VIA_THROUGH)
newvia.SetWidth(1000000)
board.Add(newvia)
pcbnew.Refresh()

txt = pcbnew.TEXTE_PCB(board)
txt.SetText("Hello")
txt.SetPosition(pcbnew.wxPoint(22000000, 15000000))
txt.SetHorizJustify(pcbnew.GR_TEXT_HJUSTIFY_CENTER)
txt.SetTextSize(pcbnew.wxSize(1000000, 1000000))
txt.SetThickness(200000)
board.Add(txt)
pcbnew.Refresh()
```



Unité :  $10^{-6}$  mm

[http://docs.kicad-pcb.org/doxygen/layers\\_\\_id\\_\\_colors\\_\\_and\\_\\_visibility\\_8h.html](http://docs.kicad-pcb.org/doxygen/layers__id__colors__and__visibility_8h.html)  
F\_Cu = PCBNEW\_LAYER\_ID\_START,  
B\_Cu = PCBNEW\_LAYER\_ID\_START+31

<sup>3</sup><https://kicad.mmccoo.com/kicad-scripting-table-of-contents/> (pour KiCAD 4) 11/11