

## SAE 2 : Concevoir et Réaliser un prototype

Le but de cette SAE est de vous familiariser avec les notions de conception et de réalisation de cartes électroniques en utilisant un logiciel de CAO et le nouveau laboratoire GEII et la conception de logiciel avec le langage Python. Le support utilisé dans cette SAE est un sonomètre.

- 1 Présentation
  - 1.1 Contexte : le sonomètre
- 2 Description de la SAE
- 3 Cahier des charges techniques
  - 3.1 Contraintes de routage :
  - 3.2 Contraintes mécaniques en mils (millièmes de pouce):
  - 3.3 Contraintes de programmation :
    - 3.3.1 Trame issue de la carte d'interface STM32
    - 3.3.2 Convention de nommage :
- 4 Les ressources mises à votre disposition
  - 4.1 Les documents :
  - 4.2 Ressources matérielles :
  - 4.3 Ressources logicielles :
- 5 Planification (Diagramme de GANTT)
- 6 Conditions de travail
- 7 Les livrables
  - 7.1 Livrables CAO
    - 7.1.1 Jalon 1 – 4 décembre 2023 : Schématique
    - 7.1.2 Jalon 2 – 18 décembre 2023 : typon
  - 7.2 Livrables informatiques
    - 7.2.1 Jalon 1 – 4 décembre 2023
    - 7.2.2 Jalon 2 – 21 décembre 2023
  - 7.3 Livrable 3 – 10 janvier 2024 : rapport avec dossier de fabrication
  - 7.4 Livrable 4 : Présentation orale – semaine du 10 au 14 janvier 2024

# 1 Présentation de la SAE

## 1.1 Contexte : le sonomètre

Pour développer vos deux compétences concevoir et réaliser en BUT 1 nous allons travailler à l'élaboration d'un sonomètre.



- Première approche :

On comprend mieux une nouvelle notion si l'on part d'éléments connus. Dans votre vie de tous les jours vous avez certainement déjà vu un vumètre. Il permet de visualiser au moyen de leds (autrefois d'afficheurs à aiguille) la mesure électrique d'un son dans un dispositif audio tel qu'une table de mixage.

- Le sonomètre :

Les points communs :

- Ils travaillent tous les deux dans le domaine des fréquences audibles.
- Ils disposent tous les deux d'un système d'affichage.

Le sonomètre diffère du vumètre par les points suivants : o C'est un appareil industriel.

- o Il dispose de son propre microphone intégré.
- o Il est étalonné pour relever un niveau de pression acoustique (en db) ou d'intensité acoustique (en  $W/m^2$ ).

Voici à quoi peut ressembler l'affichage d'un sonomètre.



## 2 Description de la SAE

Durant cette situation d'apprentissage, vous allez être amené à réaliser :

- Une carte électronique : sonomètre,
- Un programme informatique : interface homme machine (IHM)

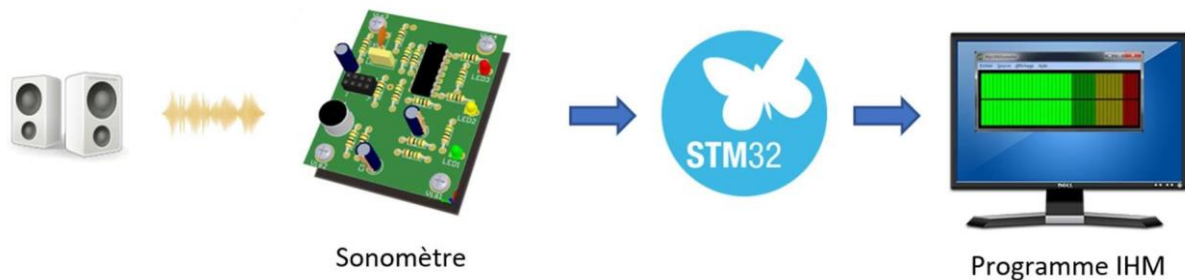


Figure 3: mise en situation de la SAE

A partir d'une source sonore à mesurer, représentée par les deux enceintes sur la figure précédente, la carte électronique « sonomètre » indiquera par l'intermédiaire de 3 LED (Light-Emitting Diode ou DEL : Diode ÉlectroLuminescente) si le niveau sonore ambiant est confortable (vert), trop fort (jaune) ou beaucoup trop fort (rouge).

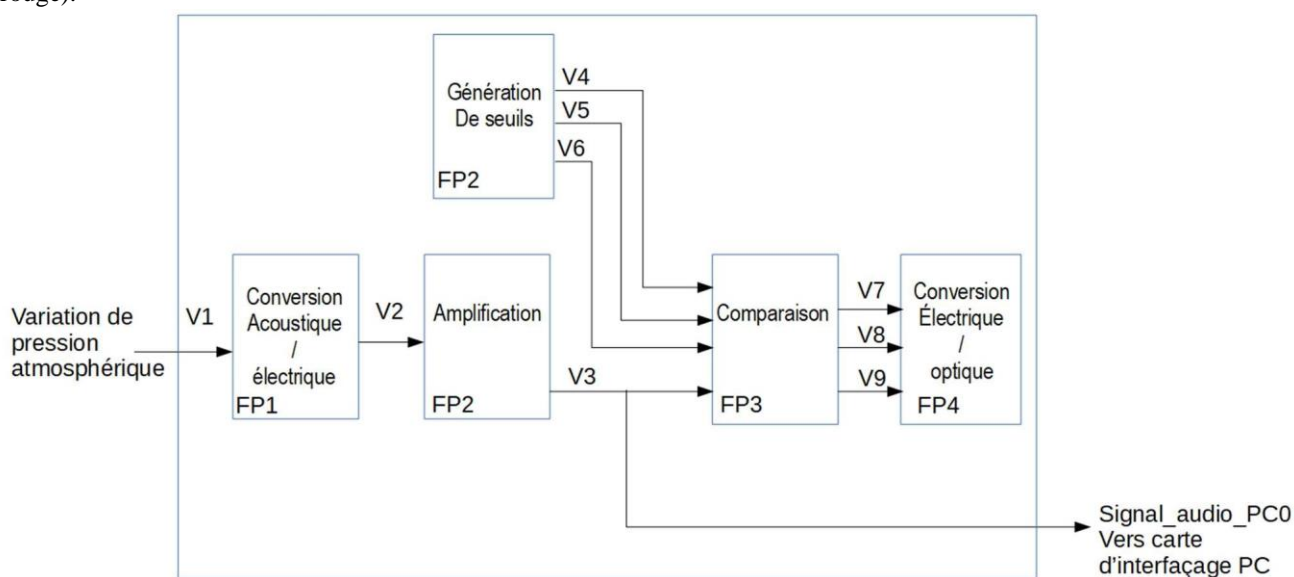
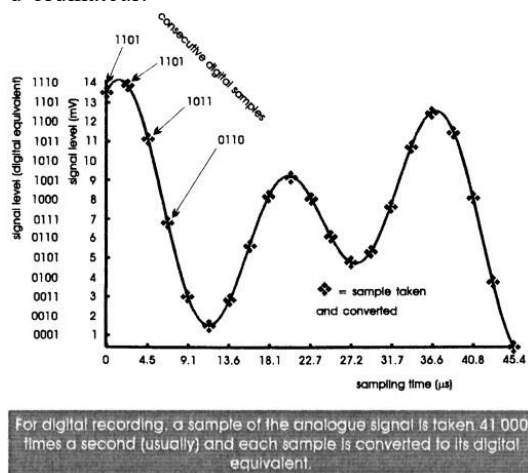


Figure 4: schéma fonctionnel de la carte "sonomètre" à réaliser

Une carte d'interface, fournie et nommée STM32 sur la figure, permettra de convertir les niveaux sonores issus du sonomètre (signal\_audio\_PC0) en données numériques pouvant être traitées et affichées sur un écran d'ordinateur.



For digital recording, a sample of the analogue signal is taken 41 000 times a second (usually) and each sample is converted to its digital equivalent.

Figure 5: principe de la Conversion Analogique Numérique

Cette carte est basée sur un Convertisseur Analogique Numérique (CAN ou ADC : Analog to Digital Converter) qui à intervalle de temps régulier vient prélever la valeur (prendre un « échantillon ») du signal « signal\_audio\_PC0 » et la converti en une valeur numérique sur 8 bits dans notre cas .

Comme on peut le voir sur la figure ci-contre (exemple sur 4 bits), à chaque « échantillonnage » correspond une valeur numérique (□) image de la valeur analogique (courbe en trait continu).

A cette information, la carte d'interface ajoutera également une information sur l'état des LED (allumée ou éteinte).

Afin de savoir où se situe chaque information dans le

signal envoyé au PC, la carte STM32 émettra périodiquement une trame (succession d'information) de 4 octets dont le format est le suivant :

Entête ADC	Valeur sur 8 bits ADC							Entête Leds	Leds On/Off					
(AB)	LSB						MSB	(CD)	0	0	0	0	0	

- Entête ADC est un octet qui vaudra toujours AB en hexadécimal. Ainsi dès que vous recevez AB vous savez que l'octet suivant correspond à la valeur issue d'un Convertisseur Analogique Numérique (CAN ou ADC : Analog to Digital Converter).
- Entête Leds vaudra toujours CD en hexadécimal. L'octet qui suivra sera un « miroir » de l'état des 3 leds rouge, jaune et verte de votre sonomètre.

Vous aurez donc également à concevoir la partie logicielle de visualisation du niveau sonore sur l'écran de cet ordinateur. Pour cela vous devrez :

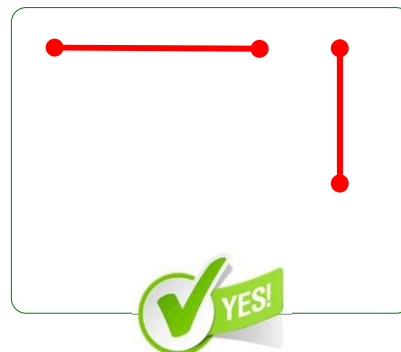
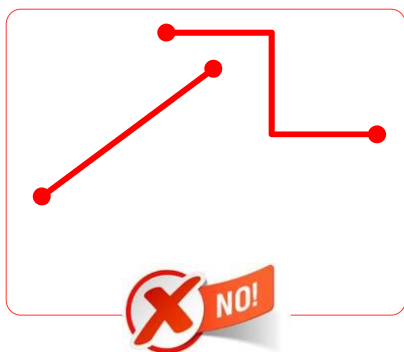
- Acquérir la trame émise par la carte d'interface STM32,
- Exploiter la trame reçue, c'est-à-dire extraire le niveau sonore et l'état des 3 leds
- Mettre à jour l'affichage.

### 3 Cahier des charges techniques

#### 3.1 Contraintes de routage :

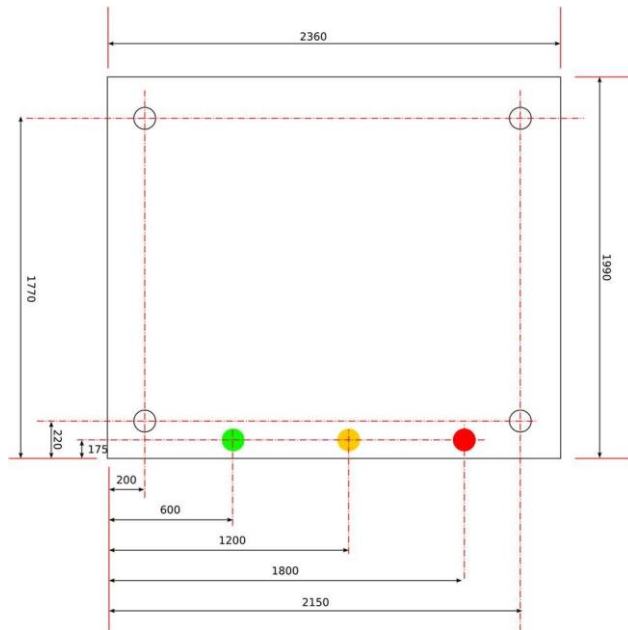
- Le routage sera réalisé en simple face (Bottom Layer).
- Le nombre de straps (horizontal et/ou vertical) autorisés est de 4
- Nom d'un membre du groupe sur la face cuivre
- Utilisation des règles de routage du fichier : Regles\_Routage\_IUT.RUL (rappelées en annexes)

Exemple de straps non valides et valides :



#### 3.2 Contraintes mécaniques en mils (millièmes de pouce):

Nous considérons ici que votre carte devra s'intégrer dans un boîtier déjà existant de l'entreprise. La position des 4 vis, des 3 leds ainsi que les dimensions de la carte sont donc figées tel que :



### 3.3 Contraintes de programmation :

#### 3.3.1 Trame issue de la carte d'interface STM32

Entête ADC	Valeur sur 8 bits ADC						Entête Leds	Leds On/Off					
(AB)	LSB					MSB	(CD)	0	0	0	0	0	

- Entête ADC est un octet qui vaudra toujours AB en hexadécimal. Ainsi dès que vous recevez AB vous savez que l'octet suivant correspond à la valeur issue d'un Convertisseur Analogique Numérique (CAN ou ADC : Analog to Digital Converter).
- Entête Leds vaudra toujours CD en hexadécimal. L'octet qui suivra indique l'état (on / off de chacune des 3 leds) suivant le code :
  - o Si aucune led n'est allumée la valeur de l'octet sera (0000 0000) = (00)
  - o Si la led verte est allumée la valeur de l'octet sera (0000 0001) = (01)
  - o Si les leds orange et verte sont allumées la valeur de l'octet sera (0000 0011) = (03)
  - o Si les leds rouge, orange et verte sont allumées la valeur de l'octet sera (0000 0111) = (07)

#### 3.3.2 Convention de nommage :

- les noms de variables devront avoir un préfixe (voir ci-dessous) suivi d'un nom qui permet de décrire le rôle de cette variable. Chaque nouveau mot du nom commencera par une majuscule.  
La convention des préfixes sera la suivante :
  - int pour les variables entières,
  - float pour les variables réelles,
  - str pour les chaînes de caractères,
  - bool pour les booléens,
  - lst pour les listes à une dimension,
  - tab pour les listes à deux dimensions,
  - label pour les widgets Label tkinter ,
  - button pour les widgets Button tkinter ,
  - entry pour les widgets Entry tkinter ,
  - frame pour les widgets Frame tkinter,
  - labelframe pour les widgets LabelFrame tkinter ,

- radiobutton pour les widgets Radiobutton tkinter,
  - o checkbutton pour les widgets Checkbutton tkinter,
  - o listbox pour les widgets Listbox tkinter,
  - o ...
- Par exemple : `intNombreDeJoursAvantNoel` est un nommage de nom de variable valable.
- les noms des fonctions et procédures devront être écrits en minuscule et les mots composant le nom seront séparés par un underscore « \_ »
- Par exemple : `def ramasser_les_cadeaux_sous_le_sapin()` : est un nommage de nom de procédure ou de fonction valable.
- les noms de classes n'auront pas de préfixe mais suivront la même règle que pour les noms de variables
- Par exemple : `class traineauDuPereNoel` : est un nommage de nom de classe valable

## 4 Les ressources mises à votre disposition

### 4.1 Les documents :

- La schématique du sonomètre (Sonometre.pdf)
- La liste des footprints des composants (BOM.pdf)
- Le tutoriel d'KiCad (logiciel de CAO (Conception Assistée par Ordinateur) dédié à la réalisation de carte électronique)
- Les tutoriels sur l'utilisation des bibliothèques Python
- Les cours du semestre 1

### 4.2 Ressources matérielles :

- Plaque de test de type Labdec.
- Composants électroniques disponibles dans le département GEII □ Matériel de laboratoire.
- Un banc de test disponible par salle (107, 108, 202, 210, 211) □ Ordinateur de l'IUT.

### 4.3 Ressources logicielles :

- KiCad + Bibliothèque de composants de l'IUT « LIB\_IUT »
- EduPython

## 5 Les livrables

### 5.1 Livrables CAO

#### 5.1.1 Jalon 1: Schématique

Vous rendrez une schématique KiCad par étudiant.

Le document attendu doit contenir :

- La schématique du sonomètre qui vous a été fournie refaite sous KiCad.
- L'assignation des footprints.

#### 5.1.2 Jalon 2 : typon

Les fichiers nécessaires au tirage de la carte seront validés lors de la dernière séance de SAÉ encadrée positionnée dans votre emploi du temps.

Ceux-ci devront être conforme au cahier des charges et comporter le nom du réalisateur.

### 5.2 Livrables informatiques

#### 5.2.1 Jalon 1

Vous proposerez une esquisse de l'aspect visuel du programme fenêtré (widgets tkinter utilisées / méthode et emplacement des widgets à utiliser) ainsi qu'un algorithme du programme console de récupération/traitement des données (avec un découpage en sous fonctions et/ou sous procédures si nécessaire).

### 5.2.2 Jalon 2

Vous restituerez deux programmes :

- Le premier sous forme « fenêtrée » qui permet de faire la simulation de l’affichage en mode VU-mètre ou sous forme d’une LED virtuelle en plaçant des boutons qui devront être enlevés sur le projet final et qui permettront, pour le premier d’avoir l’affichage du premier niveau (1 led allumée sur la carte), pour le second, l’affichage du deuxième niveau (2 leds allumées sur la carte) et pour le troisième, l’affichage du troisième niveau (3 leds allumées sur la carte). On pourra aussi faire un bouton pour effacer l’affichage.
- Le second qui sera un programme de type « console » qui permettra de se connecter à la carte de liaison via le port USB, de lire 400 bits puis de transformer les bits reçus sous forme d’une chaîne de caractère. L’objectif sera de récupérer toutes les valeurs après le bit ‘CD’ et d’en garder la plus grande valeur parmi toutes ces valeurs puis d’afficher cette valeur maximale dans la console sous la forme de leds virtuelles ou d’un VU-mètre

### 5.3 Livrable 3: rapport avec dossier de fabrication

#### **CAO :**

Vous réaliserez le dossier de avec les informations suivantes :

- o Schématique
- o Nomenclature (liste des composants, Bill Of Material)
- o Implantation des composants (sérigraphie)
- o Top Layer sans plan de masse
- o Bottom Layer sans plan de masse (avec nom en effet miroir)
- o Bottom Layer avec plan de masse
- o Vue 3D de la carte

#### **Informatique :**

Vous restituerez un programme « fenêtré » complet et correctement commenté permettant de se connecter à la carte de liaison, puis d’afficher en temps réel le niveau sonore relevé (1, 2 ou 3 leds allumées) ou de le représenter sous la forme d’un VU-mètre.

### 5.4 Livrable 4 : Présentation orale – semaine du 15 au 19 janvier 2023

On attend de vous une présentation orale avec démonstration de votre carte opérationnelle et du logiciel que vous aurez développé.

Vous disposerez de 10 minutes pour l’exposé et la démonstration suivis de 5 minutes de questions.