

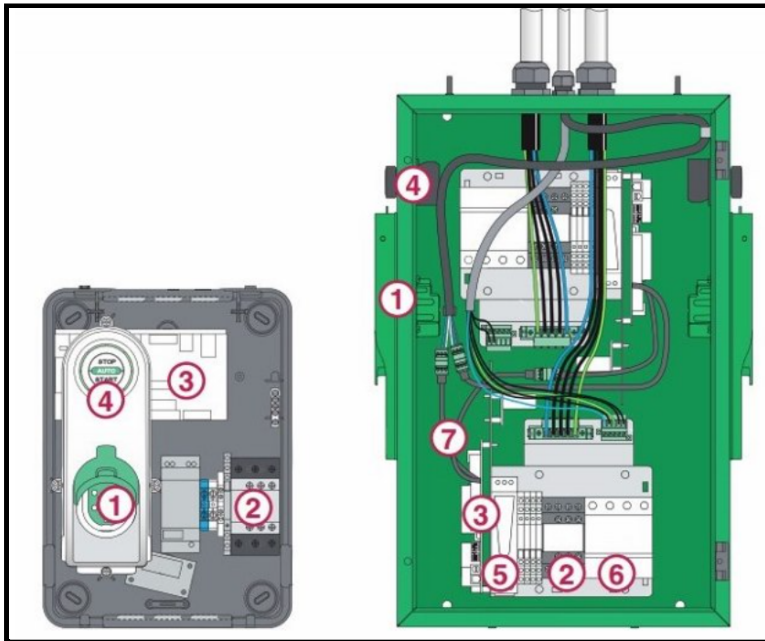
## ACTIVITÉ de OUENZERFI Ayoub

### Introduction :

Ces dernières années, le nombre de véhicules électriques est en augmentation. Ils jouent **un rôle de plus en plus important** dans le fonctionnement du réseau électrique en raison de sa ressource de stockage, où les stratégies de contrôle de chargement et de déchargement et la construction d'installations de chargement sont des **priorités à résoudre dans ce domaine**. C'est pour cela que vous étudierez une **partie précise** du prototypage de cette borne de recharge électrique. Ce projet aborde une **multitude de domaines** tels que: la transformation de signaux, l'utilisation de microcontrôleurs, des parties informatiques...

L'**objectif principal** est d'utiliser toutes vos connaissances de la **formation BUT GEII** que ce soit sur la partie technique, théorique, électronique et informatique. Et de répondre aux différentes questions.

A disposition, un **schéma complet d'une borne de recharge** :



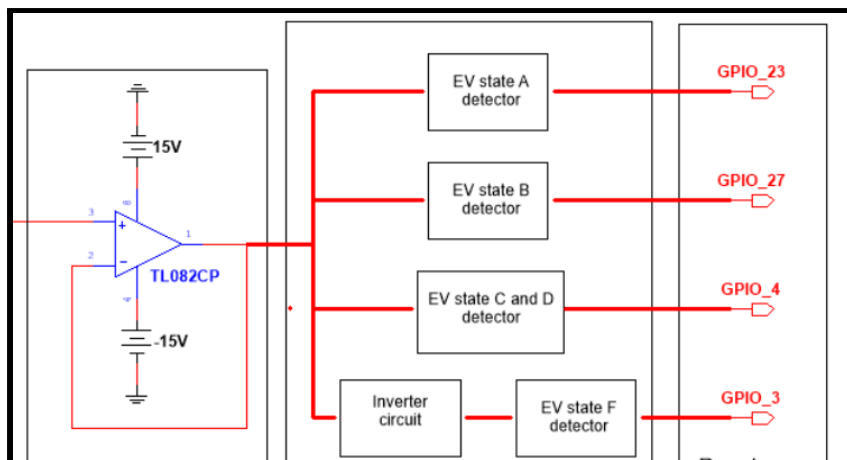
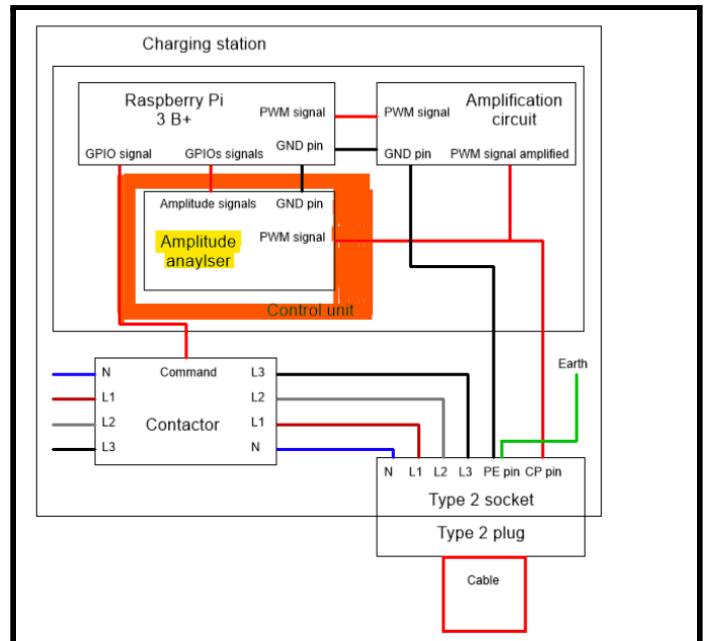
Composition du schéma :

- 1. Socle de prise
- 2. Contacteur électrique
- 3. Cartes électroniques
- 4. Boutons de commande
- 5. Alimentation 24 V
- 6. Parafoudre
- 7. Communication

Vous avez la composition d'une borne de recharge de véhicule. On décide de se focaliser sur la composition (3) =>

### Cartes électroniques.

Retrouvez ci-contre l'intégralité des composants de contrôle pour la borne de recharge. Et, maintenant focalisons-nous sur la partie "amplitude analyser". Donc, la détection d'état de chargement d'un véhicule.



Cette partie précise de l'ensemble de la borne de recharge permet d'analyser en fonction de l'amplitude du signal, un état.

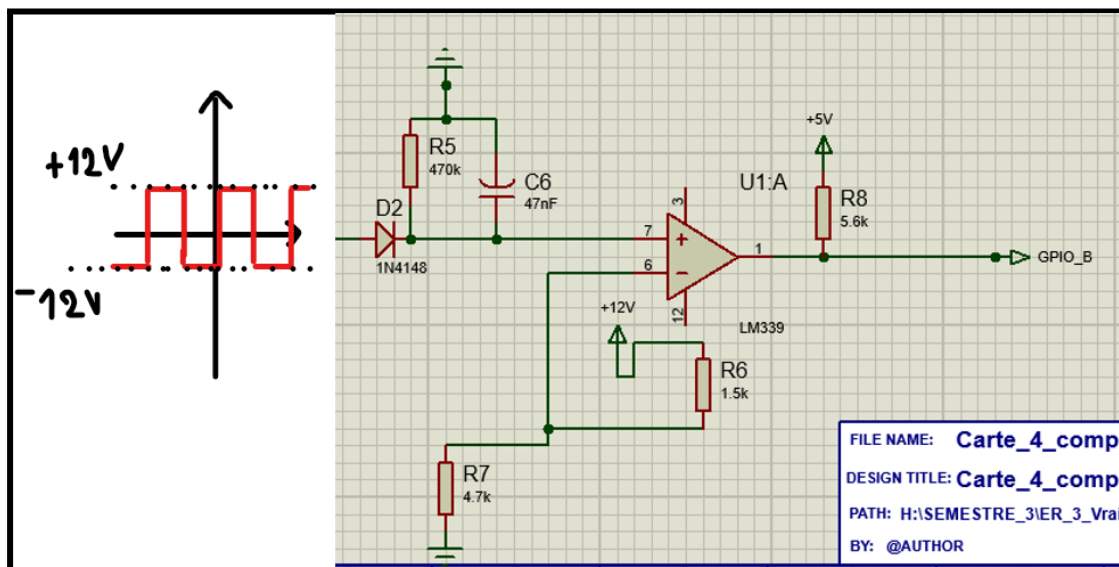
## 1. Les différentes questions :

### • 1.A Question 1 / SOLUTION :

- I. Pour l'analyse de l'état "B", avec B à l'état haut pour une amplitude de signal supérieure à +9V, réfléchir à un montage pouvant réaliser cette détection, ainsi que déterminer la valeur des résistances pour cette valeur de tension. Pour cela, prendre en entrée un **signal carré en  $\pm 12V$**  et mettre à 1 (5 V) une sortie qui pourra être détectée par un **GPIO** d'une carte électronique tel qu'un Arduino.

Indice : Aidez-vous des connaissances assimilées lors du module système électronique (SE) pour comprendre le principe de filtre, diode afin de récupérer seulement la partie positive du signal d'entrée. De plus, avoir connaissance des montages "comparateur" avec transistor est un avantage.

### **SOLUTION :**

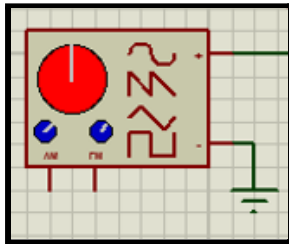


Pour cette solution vous avez l'ensemble du montage que les futurs étudiants devront déterminer. Sans connaître non plus les valeurs précises (résistance de pull-up etc.).

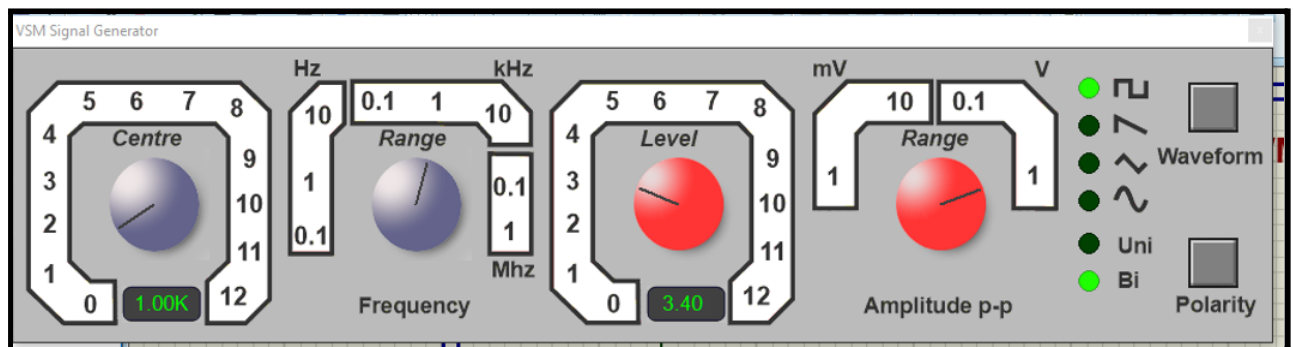
• 1.B Question 2 / SOLUTION :

- Vérifier votre montage **en simulation** sur le logiciel de CAO PROTEUS, en utilisant la fonctionnalité (**Digital Oscilloscope**) et en plaçant un signal d'entrée avec la fonction (**VSM Signal Generator**).

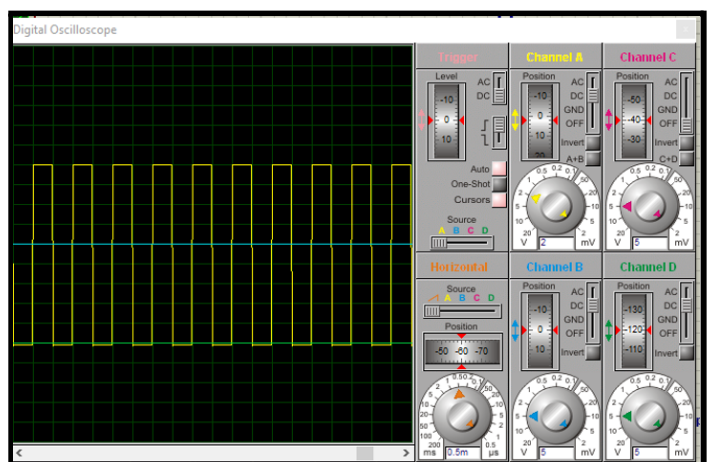
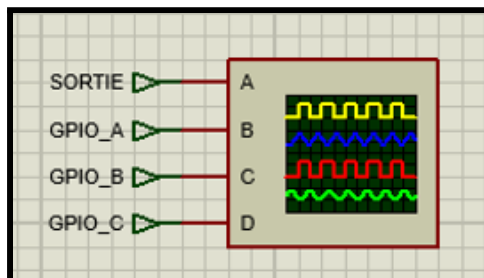
**SOLUTION :**



Voici le générateur de signal sur PROTEUS (VSM Signal Generator).



Ici, vous pouvez retrouver la partie configuration du signal d'entrée. Ici on peut mettre **un signal carré** de 1 kHz avec une amplitude de 12V.



Ici vous pouvez retrouver l'oscilloscope avec un label qui entre dans celui-ci. Dans le cas de cet exercice on ne fait entrée **que GPIO B**. Et voici la fenêtre qu'il faut bien régler pour la simulation.

Astuce pour bien régler l'oscilloscope pour la simulation : "Il faut mettre les autres Channel à 0 et souvent prendre le Channel correspondant, l'allumer puis, l'éteindre ainsi de suite jusqu'à voir la signal apparaître."

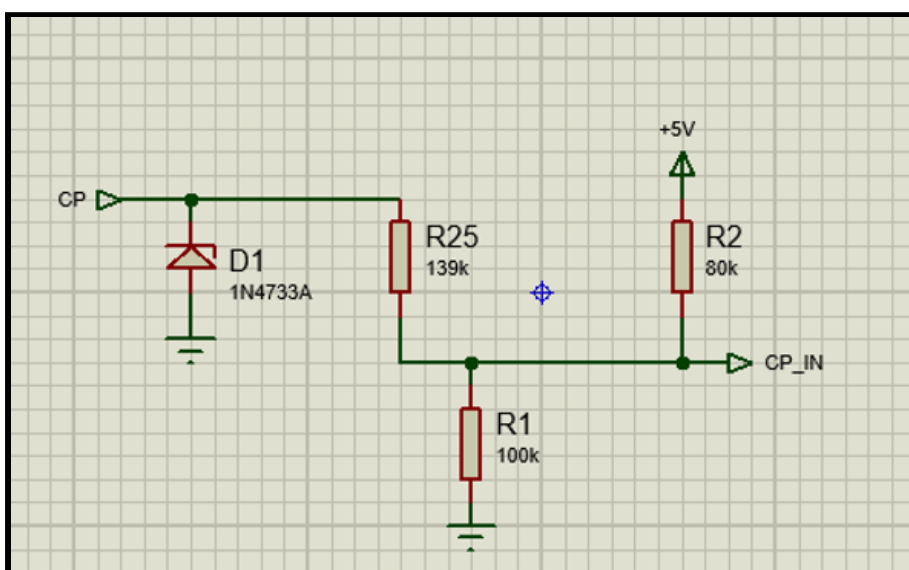
• **1.C Question 3 / SOLUTION :**

3. Sachant que les deux précédentes questions se sont basées sur seulement un état. Vous vous rendez compte que cette **carte électronique de contrôle** serait longue à faire pour tous les états.

Réfléchissez à **un autre montage** (une autre solution) qui pourrait **convertir** un signal carré en entrée de [-12 à +12V] en un signal carré de 0 à 5V. Prendre dans ce montage seulement des **résistances R1, R2 et R3**.

Indice : Aidez-vous des connaissances assimilées lors du module système électronique (SE) par rapport aux théorèmes de Millman principalement.

**SOLUTION :**



Montage assez simple qui prend en compte 3 résistance et on peut enlever la diode si dans nos équations on prend en compte le fait que le signal d'entrée est en +12V,-12V. Sinon le laisser.

- **1.D Question 4 / SOLUTION :**

4. Ayant déterminé ce montage très simple, **déterminer les relations** qui se cachent derrière. Déterminer **Vcc = 5V** en fonction de R1,R2,R3 et la tension positive du signal d'entrée.

Puis, déterminer **GND = 0 V** en fonction de R1,R2,R3 et la partie négative du signal d'entrée.

**SOLUTION :**

$$0 = \frac{-12}{(R1)} + \frac{(V_{cc})}{(R3)} \text{ avec } V_{cc} = 5V$$

ET

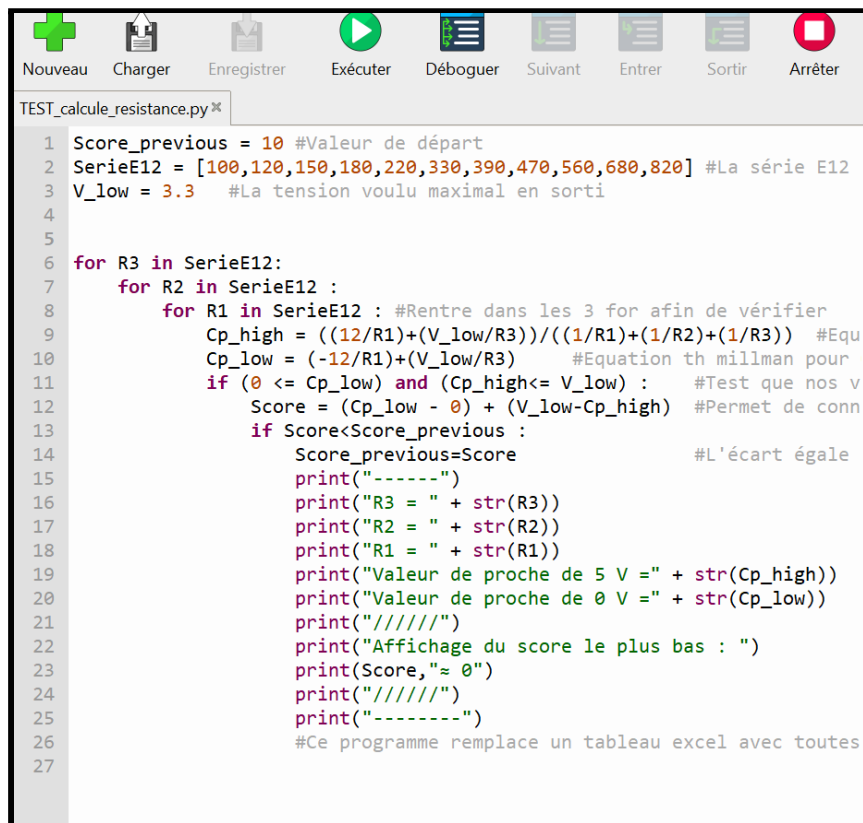
$$\text{donc } \rightarrow 5V = \frac{\left(\frac{+12}{(R1)} + \frac{(5)}{(R3)}\right)}{\left(\frac{1}{(R1)} + \frac{1}{(R2)} + \frac{1}{(R3)}\right)} \rightarrow 1^{\text{ère}} \text{ équation}$$

Les équations sont assez simples à trouver lorsque la partie réalisation du montage est bien faite.

- **1.E Question 5 / SOLUTION :**

5. A l'aide de ces relations. Définir une **composition parfaite** de valeur de résistance pour avoir les valeurs déjà énoncées. Pour, cela sur l'**IDE Thonny** qui intègre Python faire un **programme** pour la détermination de la meilleure composition. Dans cette question sera attendu l'utilisation de **condition "IF"**, d'un tableau qui comprend la valeur des résistances de **séries E12**, de **boucle "while"** et des équations définies précédemment. Ce code vous permet de tester **10 648 possibilités** rapidement.

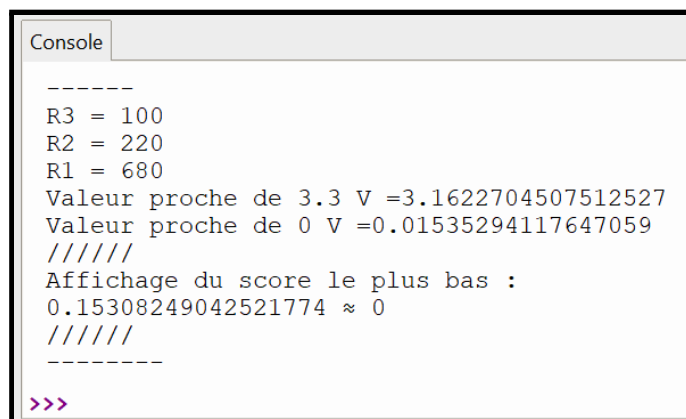
**Série E12 : [100,120,150,180,220,330,390,470,560,680,820]** ; et les multiples de ces valeurs.

**SOLUTION :**


```

TEST_calcul_resistance.py
1 Score_previous = 10 #Valeur de départ
2 SerieE12 = [100,120,150,180,220,330,390,470,560,680,820] #La série E12
3 V_low = 3.3 #La tension voulu maximal en sorti
4
5
6 for R3 in SerieE12:
7     for R2 in SerieE12 :
8         for R1 in SerieE12 : #Rentre dans les 3 for afin de vérifier
9             Cp_high = ((12/R1)+(V_low/R3))/((1/R1)+(1/R2)+(1/R3)) #Equ
10            Cp_low = (-12/R1)+(V_low/R3) #Equation th millman pour
11            if (0 <= Cp_low) and (Cp_high<= V_low) : #Test que nos v
12                Score = (Cp_low - 0) + (V_low-Cp_high) #Permet de conn
13                if Score<Score_previous :
14                    Score_previous=Score #L'écart égale
15                    print("-----")
16                    print("R3 = " + str(R3))
17                    print("R2 = " + str(R2))
18                    print("R1 = " + str(R1))
19                    print("Valeur de proche de 5 V =" + str(Cp_high))
20                    print("Valeur de proche de 0 V =" + str(Cp_low))
21                    print("////////")
22                    print("Affichage du score le plus bas : ")
23                    print(Score,"≈ 0")
24                    print("////////")
25                    print("-----")
26                    #Ce programme remplace un tableau excel avec toutes
27

```

**Exécution du code :**


```

Console
-----
R3 = 100
R2 = 220
R1 = 680
Valeur proche de 3.3 V =3.1622704507512527
Valeur proche de 0 V =0.01535294117647059
////////
Affichage du score le plus bas :
0.15308249042521774 ≈ 0
////////
-----
>>>

```

Ici, j'ai pris la tension d'entrée à 3.3V et non les 5V lors de la détermination des équations. Donc voici l'explication de mon programme ci-dessous :

“ Ayant déterminé l'**ensemble des équations** qui permettent de connaître la composition parfaite de résistance je vous met une **capture de programme** réalisé qui permet de faire ceci. Avec ici, on le remarque le tableau qui contient les valeurs des résistances de la **série E12**. Ce code est **assez simple** et permet d'aller beaucoup plus vite que de faire un fichier excel et tester les **10 648 possibilités**.

Donc, le **programme** avec le PC et sa puissance de calcul permet d'aller **beaucoup plus vite**. Donc ce code utilise des notions déjà vu en **Python** donc une **condition if** dans une autre. Et à la fin, un test pour se rapprocher le plus de la valeur que l'on souhaite. On pose un **variable "Score"** qui est l'écart entre la valeur instantanée et celle voulu et plus le score est petit, on enregistre celui-ci et on continue jusqu'à trouver ce plus petit score et on l'enregistre et affiche **la combinaison de résistance** qui permet ce petit score."