

Proposition d'activité expérimentale utilisant un condensateur pour le programme de spécialité de Terminale Générale.

Extrait du programme concerné :

3. Étudier la dynamique d'un système électrique	
Notions et contenus	Capacités exigibles <i>Activités expérimentales support de la formation</i>
Intensité d'un courant électrique en régime variable. Comportement capacitif. Modèle du condensateur. Relation entre charge et tension ; capacité d'un condensateur. Modèle du circuit RC série : charge d'un condensateur par une source idéale de tension, décharge d'un condensateur, temps caractéristique. Capteurs capacitifs.	Relier l'intensité d'un courant électrique au débit de charges. Identifier des situations variées où il y a accumulation de charges de signes opposés sur des surfaces en regard. Citer des ordres de grandeur de valeurs de capacités usuelles. <i>Identifier et tester le comportement capacitif d'un dipôle. Illustrer qualitativement, par exemple à l'aide d'un microcontrôleur, d'un multimètre ou d'une carte d'acquisition, l'effet de la géométrie d'un condensateur sur la valeur de sa capacité.</i> Établir et résoudre l'équation différentielle vérifiée par la tension aux bornes d'un condensateur dans le cas de sa charge par une source idéale de tension et dans le cas de sa décharge. Expliquer le principe de fonctionnement de quelques capteurs capacitifs. <i>Étudier la réponse d'un dispositif modélisé par un dipôle RC. Déterminer le temps caractéristique d'un dipôle RC à l'aide d'un microcontrôleur, d'une carte d'acquisition ou d'un oscilloscope.</i> Capacité mathématique : Résoudre une équation différentielle linéaire du premier ordre à coefficients constants avec un second membre constant.

Matériel particulier :

2 feuilles d'aluminium (environ A4) plastifiées sur lesquelles on aura découpé le plastique délicatement sur un côté pour fixer un fil sur l'aluminium (sous le scotch blanc sur la photo).

Plaquées l'une contre l'autre, on obtient un condensateur d'une capacité de l'ordre de quelques nanofarads.

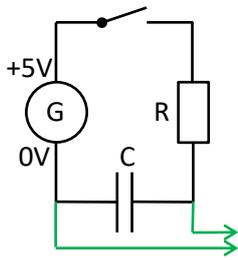
On peut modifier la géométrie du condensateur en l'enroulant sur lui-même et/ou en intercalant différents matériaux.



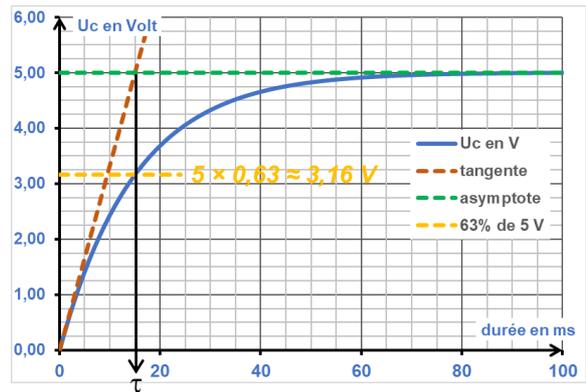
Activité n°1 : Illustrer qualitativement, à l'aide d'un microcontrôleur, l'effet de la géométrie d'un condensateur sur la valeur de sa capacité.

1. Un peu de théorie (vue en cours).

Schéma du circuit de charge d'un condensateur :



Branchement vers une carte d'acquisition, un microcontrôleur ou un oscilloscope



Si l'on suit l'évolution de la tension aux bornes du condensateur en fonction du temps, on obtient une courbe similaire à la courbe bleue ci-dessus.

Dans cet exemple, l'équation de cette courbe est $U_c = 5 \times \left(1 - e^{-\frac{t}{\tau}}\right)$ où τ est appelée temps caractéristique (ici $\tau = 15$ ms).

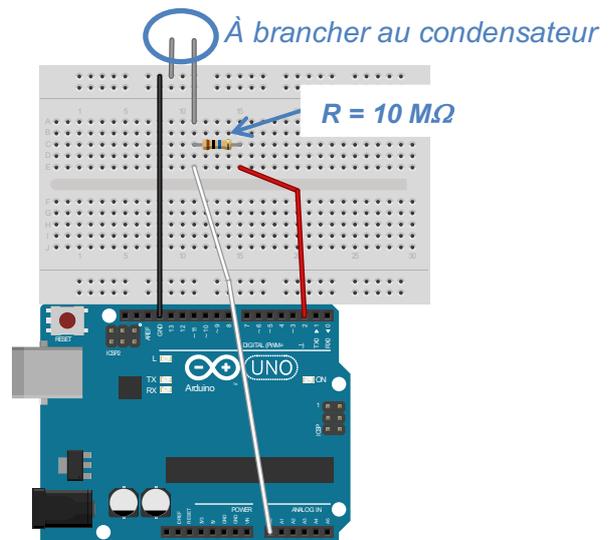
On montre que : $\tau = R \times C$.

2. Circuit à réaliser :

Ce circuit correspond à celui schématisé ci-dessus sans interrupteur.

La borne 2 peut être portée à +5V pour charger le condensateur, ou à 0V pour le décharger.

La borne A0 permet de mesurer la tension aux bornes du condensateur.



3. Programme arduino™ :

Ouvrir le 1^{er} programme nommé « *programme_1_mesure_tau* ».

Le choix a été fait de mettre le programme dans le setup, ainsi, une mesure est fait au lancement du programme et non en permanence.

Pour faire 1 autre acquisition, il suffit de relancer le programme en appuyant sur le bouton reset de la carte.

Ce programme affiche la valeur du temps caractéristique du circuit, puis celle de la capacité du condensateur.

a. Prise en main du programme :

Les lignes 1 à 11 définissent les variables qui seront utilisées par le programme.

1. En ligne 10, la variable R prend la valeur de la résistance, quelle est son unité ?

Les lignes 15 et 16 préparent l'écriture des résultats dans le moniteur série.

La ligne 17 lance une boucle « for » qui va réaliser 10 fois les instructions comprises entre les lignes 18 et 30.

2. Expliquer pourquoi les lignes 19 à 21 permettent de décharger le condensateur.

La fonction millis() est une fonction "chronomètre" : elle prend la valeur de la durée écoulée depuis le lancement du programme (ou depuis l'appui sur reset) en millisecondes.

3. Le circuit n'a pas d'interrupteur, mais la ligne 24 peut être comparée à une action sur un interrupteur : quelle serait cette action ?

On rappelle que la borne A0 donne une valeur numérique de la tension. Elle peut prendre $2^{10} = 1024$ valeurs différentes comprises entre 0 (qui correspond à 0 V) et 1023 (qui correspond à 5 V).

4. En ligne 26, on attend que la valeur lue sur la borne A0 dépasse 647 (c'est-à-dire $0,63 \times 1024$), pourquoi cette valeur ?
5. Expliquer pourquoi en ligne 29, la variable "tau" prendra la valeur du temps caractéristique du circuit en millisecondes.
6. Quand on arrivera à la 10^e répétition de la boucle, que contiendra la variable nommée "tau10" ?
7. Quel est l'intérêt de faire cette boucle de 10 mesures ?
8. Justifier que les lignes 32 et 33 permettent d'obtenir la valeur de la capacité en nanofarad dans la variable C.

b. Manipulation 1 :

Réaliser le circuit. Le condensateur sera constitué des deux feuilles d'aluminium superposées. On pourra s'assurer qu'elles restent plaquées l'une à l'autre en posant un livre dessus.

Téléverser le programme dans la carte.

9. Déterminer la capacité C du condensateur. Cette valeur servira de référence par la suite.
10. Intercaler différents matériaux entre les 2 plaques et indiquer comment la capacité est modifiée.
11. Proposer une expérience permettant de déterminer l'influence de l'épaisseur du matériau inséré entre les plaques.
12. Après validation, réaliser l'expérience, noter vos résultats expérimentaux et proposer une modélisation mathématique de l'évolution de la capacité en fonction de l'épaisseur.

c. Manipulation 2 :

Enrouler le condensateur sur lui-même.

13. La capacité du condensateur est-elle modifiée ?

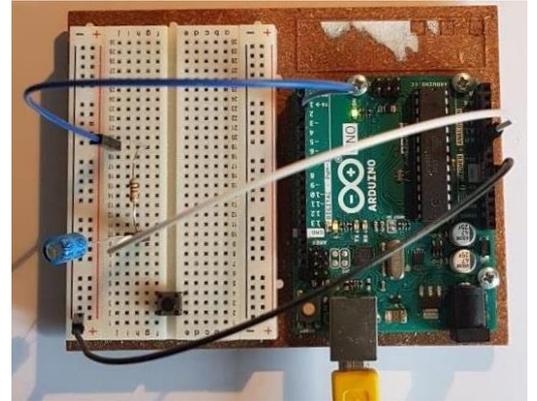
En plaçant ce condensateur dans une éprouvette, on peut y ajouter de l'eau.

14. Réaliser une expérience permettant de déterminer l'influence du niveau d'eau dans l'éprouvette. Noter vos résultats expérimentaux et proposer une modélisation mathématique de l'évolution de la capacité en fonction de la hauteur d'eau.
15. À quoi peut servir un tel dispositif.

Activité n°2 : Étudier la réponse d'un dispositif modélisé par un dipôle RC.

1. Circuit à réaliser :

Ici le condensateur est un condensateur dont la capacité affichée est de $100 \mu\text{F} \pm 20\%$ et le conducteur ohmique a une résistance affichée de $1 \text{ k}\Omega \pm 5\%$.



2. Programme arduino™ :

Ouvrir le 2^e programme nommé « *charge_decharge_C_arduino_pour_tableur* ».

Le choix a été fait de mettre le programme dans le setup, ainsi, une mesure est faite au lancement du programme et non en permanence.

Pour faire 1 autre acquisition, il suffit de relancer le programme en appuyant sur le bouton reset de la carte.

Ce programme affiche les couples (temps ; tension aux bornes du condensateur) toutes les 10 ms dans le moniteur série.

Un copier coller permet de les récupérer dans un tableur.

Si besoin, dans un tableur, "ctrl +H" permet d'ouvrir une fenêtre pour, par exemple, remplacer les points pas de virgules.

a. Prise en main du programme :

Le programme est similaire au premier.

Les lignes 1 à 9 définissent les variables qui seront utilisées par le programme.

La ligne 11 prépare l'écriture des résultats dans le moniteur série.

Les lignes 13 à 15 organisent la décharge du condensateur.

1. Que se passe-t-il lors de l'exécution de la ligne 18 ?
2. Quelle est la différence entre la variable Temps et la variable Duree ?
3. Quel est le rôle de la ligne 23 ?
4. À quoi peut-on voir que la charge du condensateur va durer 1s ?
5. Quelle ligne déclenche la décharge du condensateur ?

b. Prise et mise en forme des mesures :

Une série de mesures est prise au lancement du programme. On peut recommencer, en effaçant le moniteur série et en appuyant sur le bouton "reset" sur la carte.

Ensuite, par copier-coller dans un tableur, on peut créer un fichier csv qui sera exploitable par un programme python™.

Sinon, après avoir créé deux variables (tms et U) dans le tableur latis-pro®, on peut importer les valeurs par copier-coller.

c. Modélisation :

Dans un premier temps, créer une variable t et faire calculer $t = tms/1000$ qui permet de convertir la durée en seconde.

À l'aide de l'outil de modélisation de latis-pro[®], modéliser la charge du condensateur. On peut utiliser un modèle utilisateur à partir de l'équation théorique : $U_C = 5 \times \left(1 - e^{-t/\tau}\right)$

Dans l'outil de modélisation, rentrer la formule : $A \cdot \text{Exp}(-t/\text{tau})$ et faire calculer le modèle.

6. Quelle valeur le logiciel propose-t-il pour A ? Est-ce cohérent avec la théorie ?
7. Quelle valeur le logiciel propose-t-il pour le temps caractéristique ?
En tenant compte de la précision donnée par le fabricant pour C et R, vérifier que cette valeur est cohérente.

d. Pour aller plus loin :

Réaliser une expérience pour déterminer l'influence sur le temps caractéristique si on remplace le condensateur de 100 μF par 2 condensateurs de 100 μF en série, puis en dérivation.

Quelques résultats

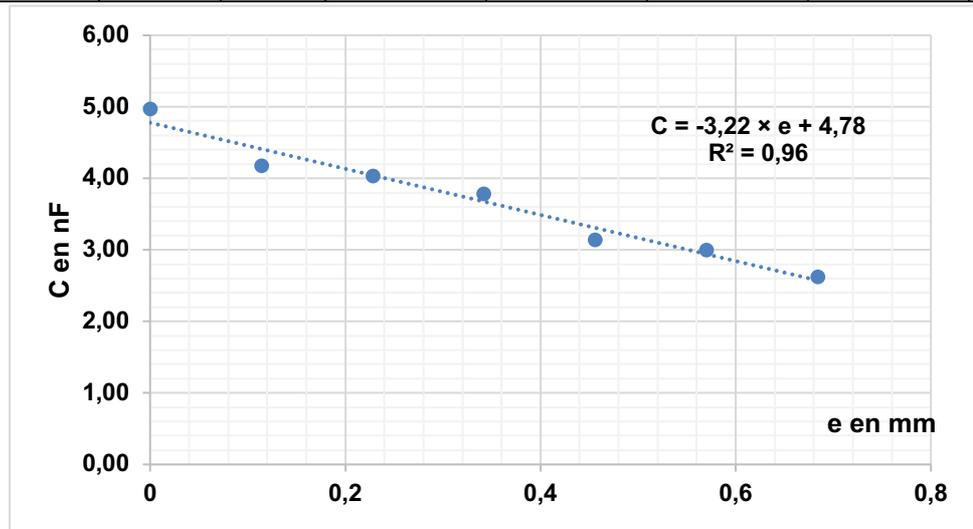
1. L'influence de l'épaisseur du matériau :

Obtenu en intercalant de plus en plus de feuilles de papier entre les plaques :

nb feuilles	épaisseur papier mm	C en nF	mesures					
0	0	4,97		4,9	4,93	5,04	5	4,99
1	0,114	4,18		4,25	4,13	4,14	4,2	4,17
2	0,228	4,04		3,99	4,11	4,06	3,92	4,1
3	0,342	3,78		3,8	3,74	3,79	3,81	3,75
4	0,456	3,14		3,12	3,1	3,22	3,13	3,12
5	0,57	3,00		3,02	3,04	3,02	2,92	3
6	0,684	2,63		2,6	2,63	2,58	2,58	2,74

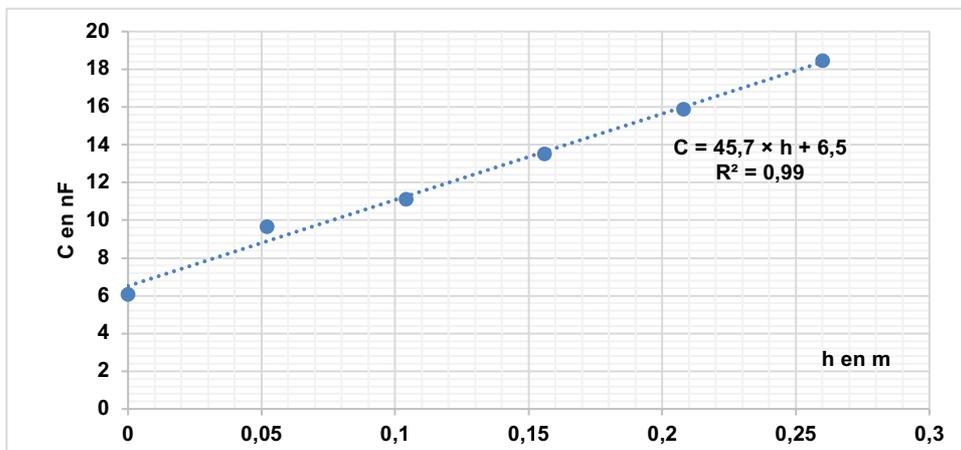
500

57



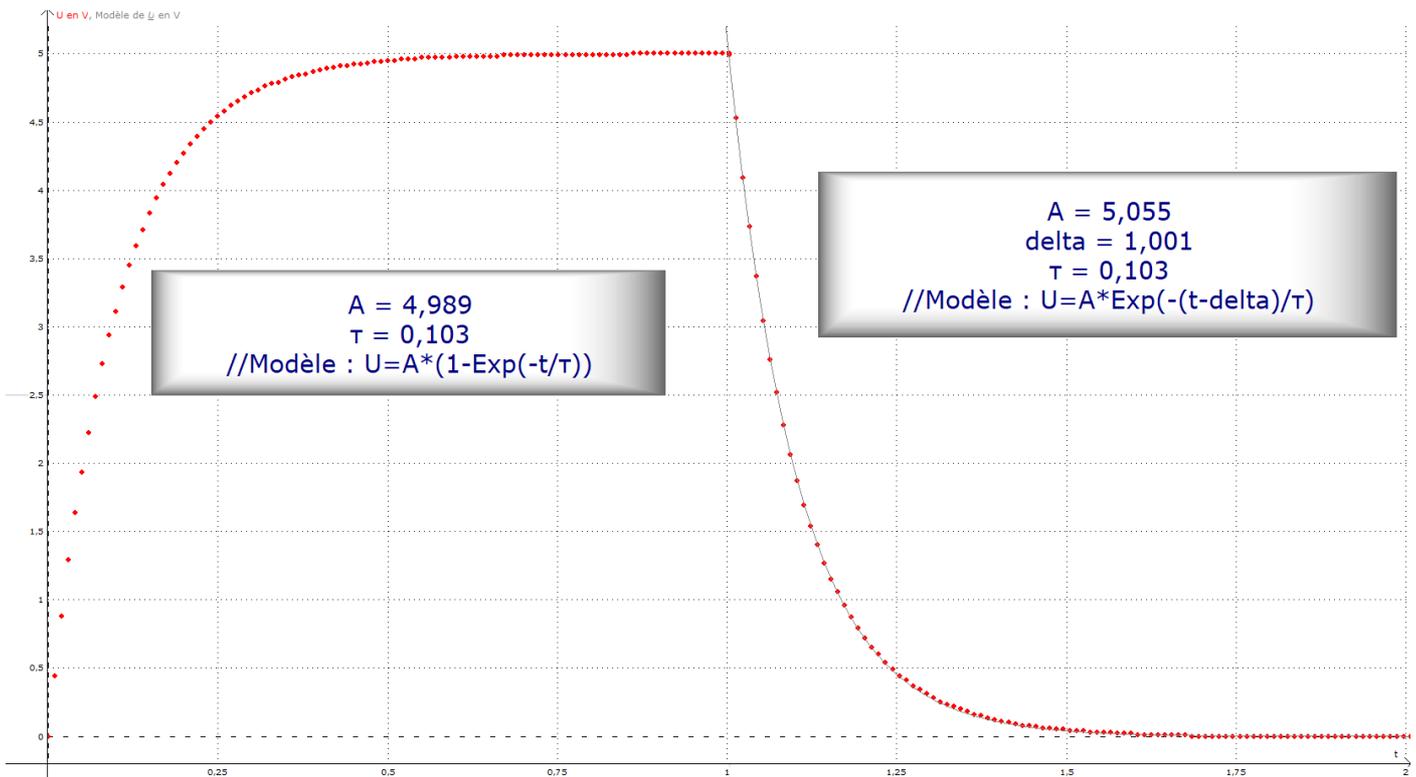
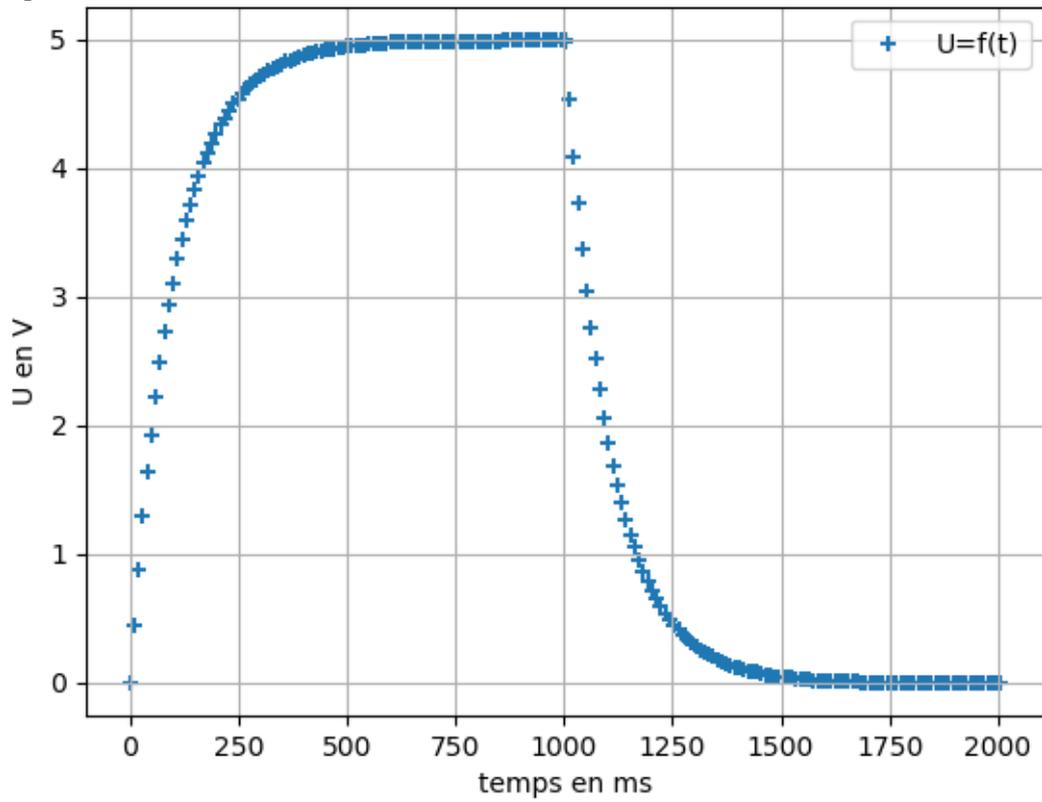
2. L'influence de la hauteur d'eau :

volume eau	hauteur en m	C en nF	mesures :				
0	0	6,062	6	6,08	6,09	6,09	6,05
50	0,052	9,656	9,62	9,63	9,61	9,71	9,71
100	0,104	11,116	10,82	10,99	11,07	11,3	11,4
150	0,156	13,522	12,81	13,18	13,71	14,01	13,9
200	0,208	15,876	15,31	15,85	15,93	16,11	16,18
250	0,26	18,466	17,95	18,3	18,53	18,71	18,84



3. L'influence de l'épaisseur de la hauteur d'eau :

Charge et décharge du condensateur :



Programmes :

Programme n°1 :

```
1 // broches
2 const int(Alim) = 2;
3 const int(mesure) = A0;
4
5 // grandeurs
6 unsigned long debutCharge;
7 unsigned long tau;
8 unsigned long tau10 = 0;
9 float C ;
10 float R = 1000000;
11 int i;
12
13 void setup(){
14 // initialisation moniteur série
15 Serial.begin(9600);
16 Serial.println(" mesure... ");
17 for(i=0 ; i<10 ; i++){
18 // décharge préalable de C (mise à 0V pendant 1s)
19 pinMode(Alim, OUTPUT);
20 digitalWrite(Alim, LOW);
21 delay(1000); //attente de 1s
22
23 // charge de C
24 digitalWrite(Alim, HIGH);
25 debutCharge=millis();
26 while(analogRead(mesure) < 647){
27 //647 = 63% de 1023 on attend...
28 }
29 tau = millis() - debutCharge;
30 tau10=tau10 + tau ;
31 }
32 float (tau) = float(tau10)/10;
33 float C = float(tau)*1000000/R; // C en nF
34 //affichage :
35 Serial.print("constante de temps : ");
36 Serial.print(tau);
37 Serial.println(" ms ");
38 Serial.print("capacité : ");
39 Serial.print(C);
40 Serial.println(" nF ");
41 }
42
43 void loop(){
44 }
```

Programme n°2 :

```
1 // broches
2 const int(Alim) = 2;
3 const int(mesure) = A0;
4 // grandeurs
5 unsigned long Temps =0;
6 unsigned long Duree = 0;
7 float Uc;
8 int IntervalMesure = 10; //mesures toutes les 10 ms
9
10 void setup(){
11   Serial.begin(9600); // initialisation port série
12 // décharge préalable de C
13   pinMode(Alim, OUTPUT);
14   digitalWrite(Alim, LOW);
15   delay(2000); //attente de 2s
16 // charge de C
17   int i = 0;
18   digitalWrite(Alim, HIGH);
19   while(i<=100){
20     if (millis()-Temps >= IntervalMesure){
21       Temps=millis();
22       Duree = millis()-1999;
23       Uc = (float(analogRead(mesure)) * 5 /1023);
24       Serial.print(Duree);
25       Serial.print("\t");
26       Serial.print(Uc);
27       Serial.print("\n");
28       i++;
29     }
30   }
31 // décharge de C
32   i = 0;
33   Temps = millis();
34   Temps = Temps - Temps; // mise à 0 du chrono
35   digitalWrite(Alim, LOW);
36   while(i<=100){
37     if (millis()-Temps >= IntervalMesure){
38       Temps = millis();
39       Duree = millis()-1999;
40       Uc = (float(analogRead(mesure)) * 5 /1023);
41       Serial.print(Duree);
42       Serial.print("\t");
43       Serial.print(Uc);
44       Serial.print("\n");
45       i++;
46     }
47   }
48 }
49 void loop(){
50 }
```

Programme python™ :

Nécessite que les données soient dans un fichier csv appelé 'valeurs.csv' dans le même dossier que ce programme python™.

```
1 import matplotlib.pyplot as plt
2
3 t = []
4 U=[]
5 #ouverture du fichier csv en lecture
6 fichier= open('valeurs.csv','r')
7 fichier.readline()
8
9 #lecture du tableau ligne par ligne pour remplir progressivement
10 # les variables en remplaçant les , par des .
11 for ligne in fichier:
12     ligne_lue=ligne.split(';')
13     t.append(float(ligne_lue[0].replace(',','.')))
14     U.append(float(ligne_lue[1].replace(',','.')))
15
16 fichier.close()
17
18 plt.scatter(t,U, marker="+", label='U=f(t)')
19 plt.xlabel('temps en ms')
20 plt.ylabel('U en V')
21
22
23 plt.grid()
24 plt.legend()
25 plt.savefig("charge_decharge.png")
26 plt.show()
```