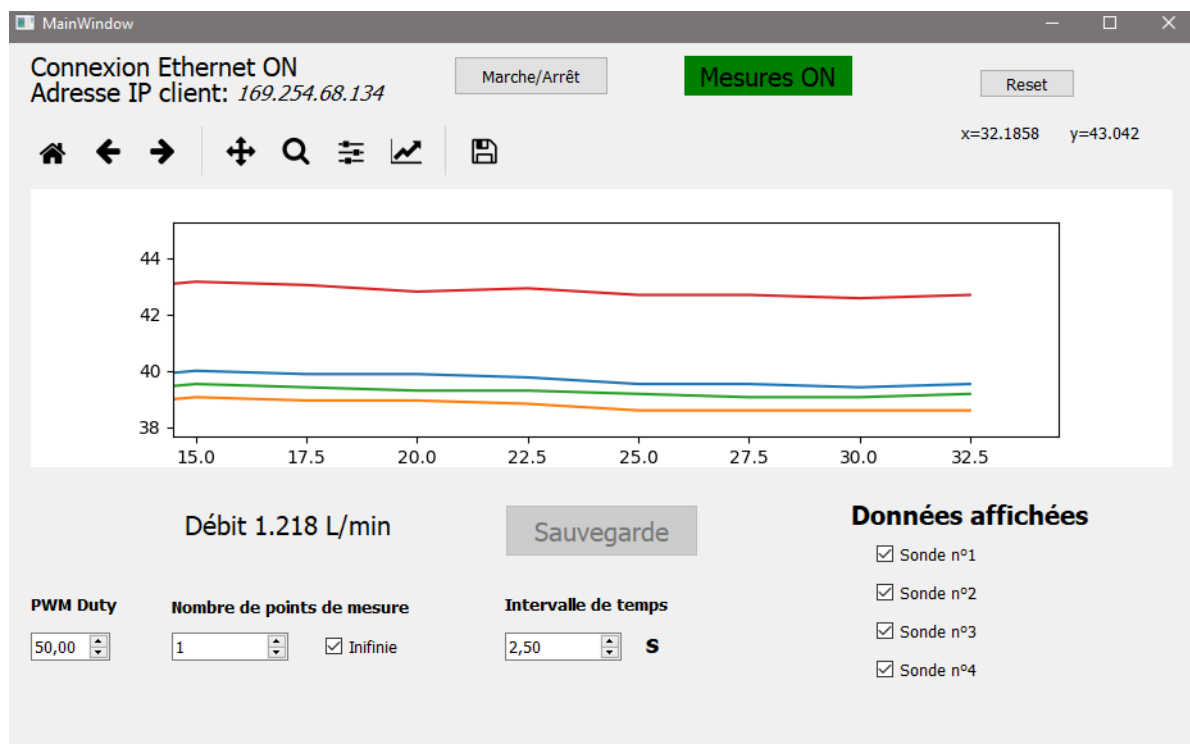


Document utilisateur

Programme de gestion du banc de mesures



Saulquin Clément

IUT GEII Lyon1



Sommaire

I. Présentation générale du logiciel.	3
II. Réglage de l'adresse IP.....	4
III. Réaliser des mesures.....	6
IV. Mesure de débit et PWM	8
V. Annexes.....	9
A. Brochage de la carte arduino.....	9
B. Liens utiles	9

I. Présentation générale du logiciel.

Ce logiciel permet de récupérer les différentes mesures provenant du contrôleur Arduino Uno. Il permet entre autre de récupérer différents relevés de température à différents endroit du banc de mesure pour ensuite les afficher sous forme de graphique puis, de les sauvegarder sous format csv compatible avec le tableur Excel. Il permet aussi d'afficher le débit à l'intérieur du circuit de refroidissement. Enfin le logiciel sert à modifier la fréquence du PWM contrôlant le hacheur servant à modifier la vitesse de la pompe. Toutes ses opérations se font en temps réel et les données sont récupérées grâce à une liaison ethernet entre l'ordinateur hôte et la carte Arduino.

Les codes du logiciel sont disponibles en téléchargement libre sur internet. Vous trouverez le lien de téléchargement dans les annexes partie [Liens utiles](#).

II. Réglage de l'adresse IP.

Le démarrage du logiciel demande de paramétrer l'adresse IP de la carte Arduino. En effet l'adresse IP de la carte doit se situer dans le même sous-réseau que la machine hôte du programme. Je vais donc vous expliquer les différentes étapes permettant de faire ce réglage.

1°) Ouvrez une console de commande Windows, pour cela aller dans le menu démarrer puis dans la case 'recherche' saisissez *cmd* puis taper sur la touche Entrée.

2°) Taper ensuite la commande *ipconfig* dans le terminal et relever l'adresse IP a la ligne « Adresse IPv4 » et le masque de sous réseau a la ligne « Masque de sous-réseau ».

```
C:\Users\Utilisateur>ipconfig

Configuration IP de Windows

Carte Ethernet Ethernet :

    Suffixe DNS propre à la connexion. . . :
    Adresse IPv6 de liaison locale. . . . : fe80::347c:caf0:bf53:7ddd%4
    Adresse IPv4. . . . . : 192.168.1.45
    Masque de sous-réseau. . . . . : 255.255.255.0
    Passerelle par défaut. . . . . : 192.168.1.1
```

3°) Déterminer ensuite la futur adresse de la carte Aruidno. Pour cela reprenez l'adresse IP de votre ordinateur et modifier le dernier chiffre de cette adresse en faisant attention à ce que ce chiffre reste dans la plage de valeurs [1 ; 255]. Dans cet exemple, une adresse IP possible pour la carte est 192.168.1.46.

4°) Une fois l'adresse IP de l'Arduino choisis ouvrez la programme Arduino et modifier la ligne sur l'image si dessous en séparant les chiffre de l'adresse IP par des virgule ' , ' et non par des points.

```
///Initialisation des variables///
IPAddress ip(192, 168, 1, 46); // /\ l'adresse IP doit être dans le même sous réseau que l'ordinateur
```

Ensuite brancher la carte sur le PC via la liaison USB et télé verser le programme dans l'Arduino.

5°) Reliez ensuite votre PC à l'Arduino avec un câble RJ45. Reprenez ensuite votre console de commande et envoyer un ping vers votre carte Arduino afin de vérifier que les deux machines peuvent communiquer sans soucis. Pour cela taper la commande *ping @IP* en remplaçant @IP par l'adresse de l'Arduino. Si toutes les requêtes ont étaient reçut alors la communication fonctionne parfaitement. Si

aucunes trames n'a étaient reçut revérifier l'adresse IP de l'Arduino.

```
C:\Users\Utilisateur>ping 192.168.1.46

Envoi d'une requête 'Ping' 192.168.1.46 avec 32 octets de données :
Réponse de 192.168.1.46 : octets=32 temps<1ms TTL=128
Réponse de 192.168.1.46 : octets=32 temps<1ms TTL=128
Réponse de 192.168.1.46 : octets=32 temps<1ms TTL=128
Réponse de 192.168.1.46 : octets=32 temps<1ms TTL=128

Statistiques Ping pour 192.168.1.46:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms
```

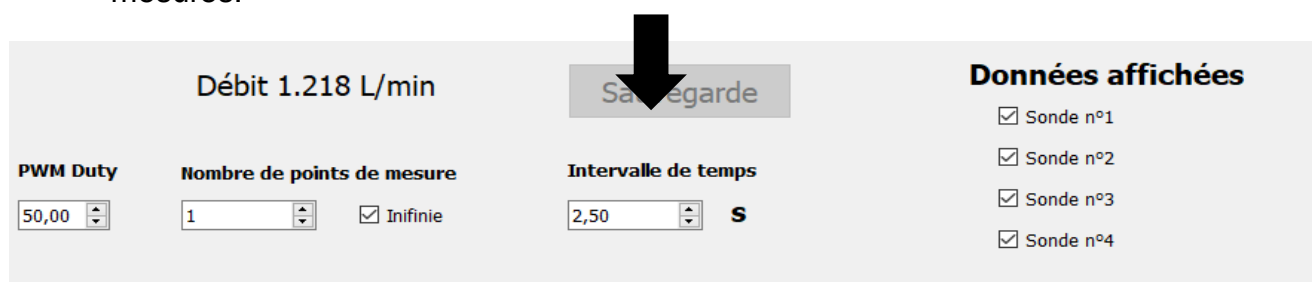
6°) Pour finir dans le fichier **donnees** modifier l'adresse IP de la carte Arduino en utilisant la notation en décimal pointées.

Après toutes ces étapes, le logiciel une fois lancé l'indicateur en haut à droite de l'interface graphique devrait indiquer que la communication est établie. Si ce n'est pas le cas cliquez sur le bouton « Réessayer », vérifier que le câble RJ45 est bien branché et que la carte arduino est bien alimenter.

III. Réaliser des mesures

Pour commencer une mesure vérifier d'abord que la communication ethernet entre le logiciel et le contrôleur. Si ce n'est pas le cas appuyé sur le bouton « Réessayer ». Si ça ne marche toujours pas je vous réfère au paragraphe précédent.

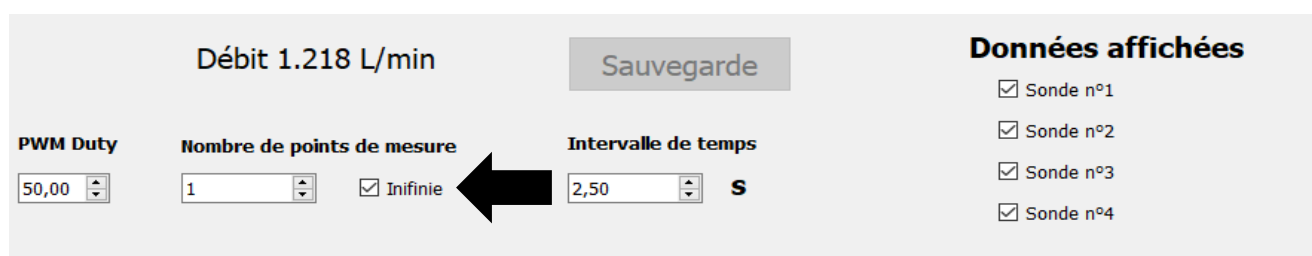
Avant de lancer une mesure il faut configurer la mesure. Deux paramètres sont important, le nombre de points de mesures et le temps entre chaque point de mesures.



The screenshot shows a software interface for configuring measurements. At the top, it displays 'Débit 1.218 L/min' and a 'Sauvegarde' button. Below this, there are three main configuration sections: 'PWM Duty' with a value of 50,00; 'Nombre de points de mesure' with a value of 1 and an 'Infinie' checkbox; and 'Intervalle de temps' with a value of 2,50 and a unit 's'. To the right, under 'Données affichées', there are four checkboxes for 'Sonde n°1', 'Sonde n°2', 'Sonde n°3', and 'Sonde n°4', all of which are checked. A large black arrow points down to the 'Sauvegarde' button.

Pour modifier le temps entre chaque point de mesure, il vous suffit de le modifier dans la zone prévue à cet effet. Le temps est à rentrer en seconde et il est possible de rentrer des temps à virgules.

Pour le nombre de point de mesure deux choix sont possibles. Soit vous rentrer un nombre de point définis dans la zone de saisie prévue à cet effet, soit vous cochez la case « infinie » et les mesures se feront tant que vous n'avez pas arrêter vous-même le programme.



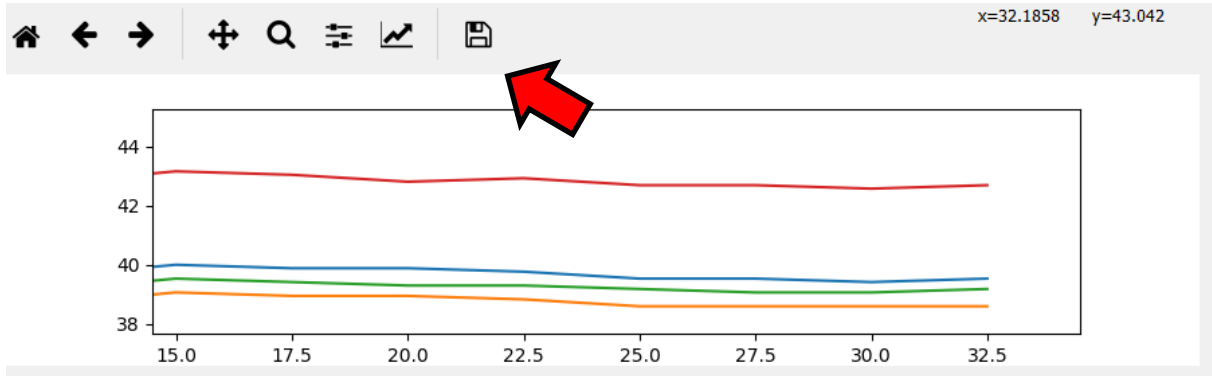
This screenshot is similar to the previous one, showing the same configuration interface. However, a large black arrow points to the 'Infinie' checkbox in the 'Nombre de points de mesure' section, highlighting it as a configuration option.

Une fois les paramètres effectués, vous pouvez alors lancer les mesures en appuyant sur le bouton « Marche/Arrêt ». Les mesures de température et de débit sont alors lancées en temps réel. Vous pouvez modifier les courbes affichées sur le graphique à tout moment à l'aide des cases à cocher dans la partie « Données affichées ». Ces cases n'ont pas d'influence sur la récupération et la sauvegarde des données mais uniquement sur l'affichage des courbes sur le graphique.

Si la connexion ethernet est perdue en cours de mesures, le programme s'arrête automatiquement. Vérifier alors la connexion du câble ethernet sur l'ordinateur ainsi

que sur la carte Arduino. Vous pourrez alors reprendre les mesures sans perdre de données.

Pour arrêter la mesure en cours cliquez sur le bouton « Marche/Arrêt ». Vous pourrez alors sauvegarder toutes les données récupérées dans un fichier en format csv compatible avec le tableur excel en appuyant sur le bouton « Sauvegarde ». Vous pouvez également sauvegarder les courbes en format PNG en appuyant sur l'icône de la disquette dans la barre des tâches du graphique.



Si les mesures ne correspondent pas à vos attentes vous pouvez toujours réinitialiser les mesures effectuées en cliquant sur le bouton « Reset ».

IV. Mesure de débit et PWM

Le logiciel permet de connaître le débit circulant dans les tuyaux en temps réel. La mesure est effectuée toutes les secondes puis le résultat est converti en litre par minutes (L/min). Cette mesure est à caractère indicatif et n'est pas d'une extrême précision mais permet de donner un bon ordre de grandeur du débit régnant dans le circuit de refroidissement. La mesure du débit est lancée au démarrage du programme et non au lancement d'une mesure.

Pour finir vous pouvez modifier à tout moment le rapport cyclique du PWM envoyé sur le hacheur contrôlant la vitesse de la pompe. Pour cela modifiez la valeur dans la zone prévue à cet effet. La valeur est à donner en pourcentage et peut donc aller de 0% (arrêt) à 100% (plein régime). La valeur par défaut est de 50%. Lors d'une modification de la valeur du rapport cyclique du PWM, une trame ethernet est directement envoyée vers la carte arduino qui modifie de suite le rapport cyclique. La fréquence de sortie du PWM est de 25KHz et est directement envoyée au hacheur de contrôle.

V. Annexes

A. Brochage de la carte arduino

Broches analogiques A0, A1, A2, A3 : Entrée analogique des capteurs de température.

Broche digitale 2 : Entré du capteur de débit délivrant des impulsions.

Broche digitale 9 : Sortie du PWM 25KHz

B. Liens utiles

Site officiel Arduino : <https://www.arduino.cc/>

Site officiel de Python : <https://www.python.org/>

Site officiel de Pyinstaller (permet de convertir un .py en .exe) :
<http://www.pyinstaller.org/>

Téléchargement de l'archives avec l'exécutable et le programme Python :
<http://www.mediafire.com/file/vp5hndzw8gdxznc/programme.rar/file>