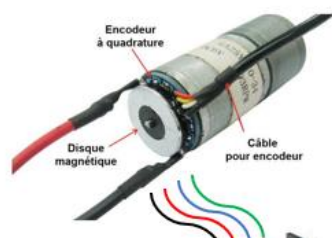
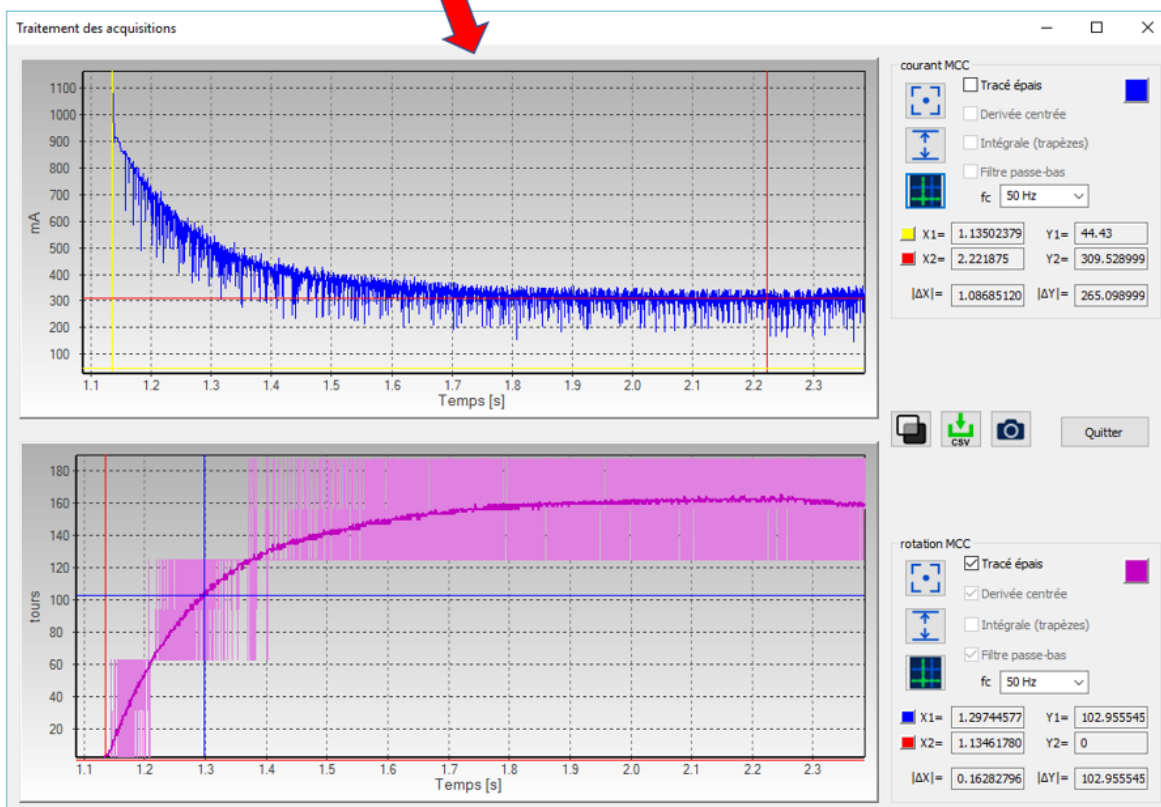


INTERFACE D'ACQUISITION « BBAcq »



SimpleServerTCP.ino



Sommaire


1	Présentation.....	3
2	Fenêtre principale de l'interface BBAcq	4
3	Connexion avec le serveur.....	4
3.1	Liaison série locale	4
3.2	Liaison TCP/IP réseau.....	5
4	Lecture des données	7
4.1	Affichage en temps-réel.....	7
4.2	Formatage des données	9
4.3	Options de lecture.....	11
4.3.1	Zoom/Dézoom	11
5	Traitement des données (mode enregistrement).....	12
5.1	Environnement de travail	13
5.2	Fonctionnalités d'affichage.....	13
5.3	Utilisation des curseurs.....	14
5.4	Fonctions avancées (traitements des données)	15
5.5	Enregistrement des données	16

1 Présentation

BagBox est un système qui, au moyen d'une carte Arduino UNO classique et d'un logiciel dédié, permet de récupérer les données de capteurs connectés à celle-ci, en local ou à distance (à travers un réseau Ethernet local) ; afin de les exploiter en démarche d'ingénierie.

La carte (ou plutôt les cartes) Arduino tiennent aujourd'hui une place importante dans nos systèmes embarqués et instrumentés, de par la simplicité d'interfaçage qu'elles procurent, au travers de la multitude de modules d'extension et de capteurs dédiés, et la simplicité de programmation.

Afin de donner une seconde vie à un système didactisé dont l'interface d'acquisition était devenue inopérante et obsolète, l'idée d'utiliser une carte Arduino comme centrale d'acquisition (exclusivement !) est apparue, non comme une évidence, mais comme une grande possibilité. Sachant que des modules d'extension (comme la célèbre Ethernet Shield) permettent de relier aisément une telle carte à un réseau local, il devenait de plus possible de s'affranchir de la contrainte d'utiliser un poste dédié comme interface de transmission des données acquises.

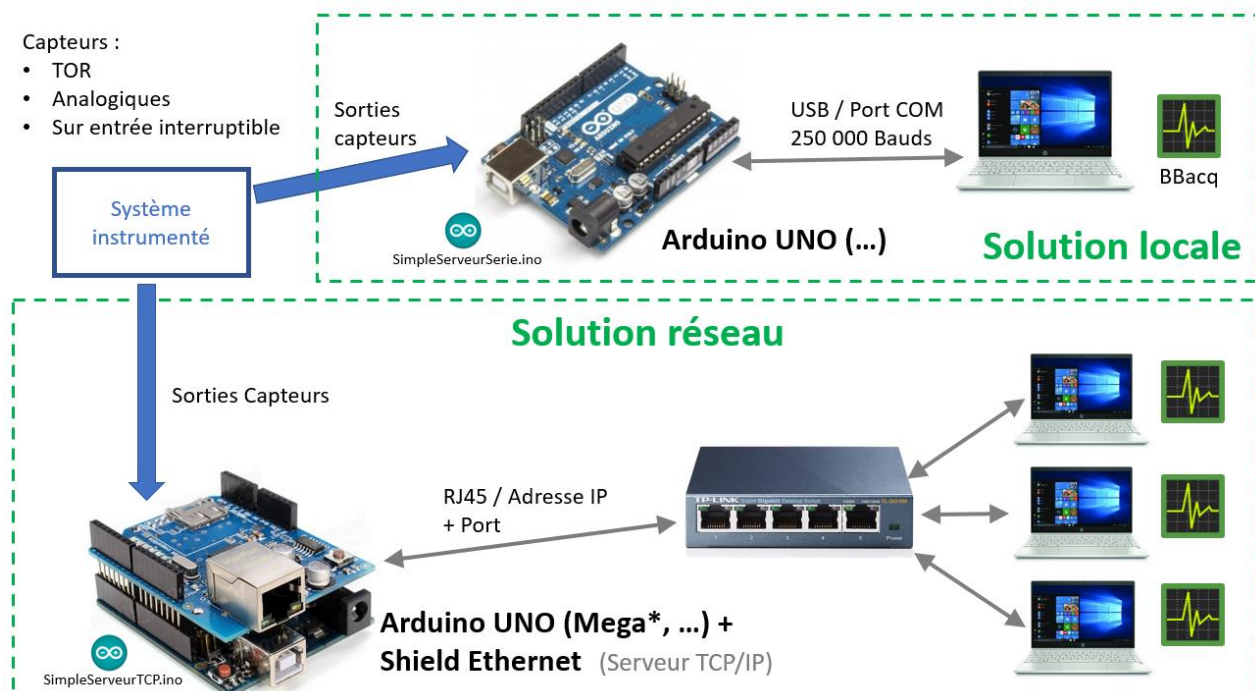
L'interface graphique **BBacq**  développée en lien avec cette centrale d'acquisition offre de nombreuses fonctionnalités, en assurant une **période d'échantillonnage de 500 µs** de chaque capteur. Les performances sont assurées si le capteur en question présente :

- une sortie logique en Tout Ou Rien (cas d'un bouton poussoir, un interrupteur, ...) ;
- une sortie analogique (potentiomètre, accéléromètre analogique, résistance de mesure, pont d'extensométrie avec ampli d'instrumentation, ...) ;
- ou si une fonction d'interruption doit être écrite sur front (montant ou descendant) d'une entrée (comme un codeur incrémental).

Les capteurs de type I²C, SPI ou de liaison série spécifique sont à utiliser avec grande précaution, plus de détails quant aux recommandations sont données à la fin, des exemples spécifiques sont fournis à titre d'illustration.

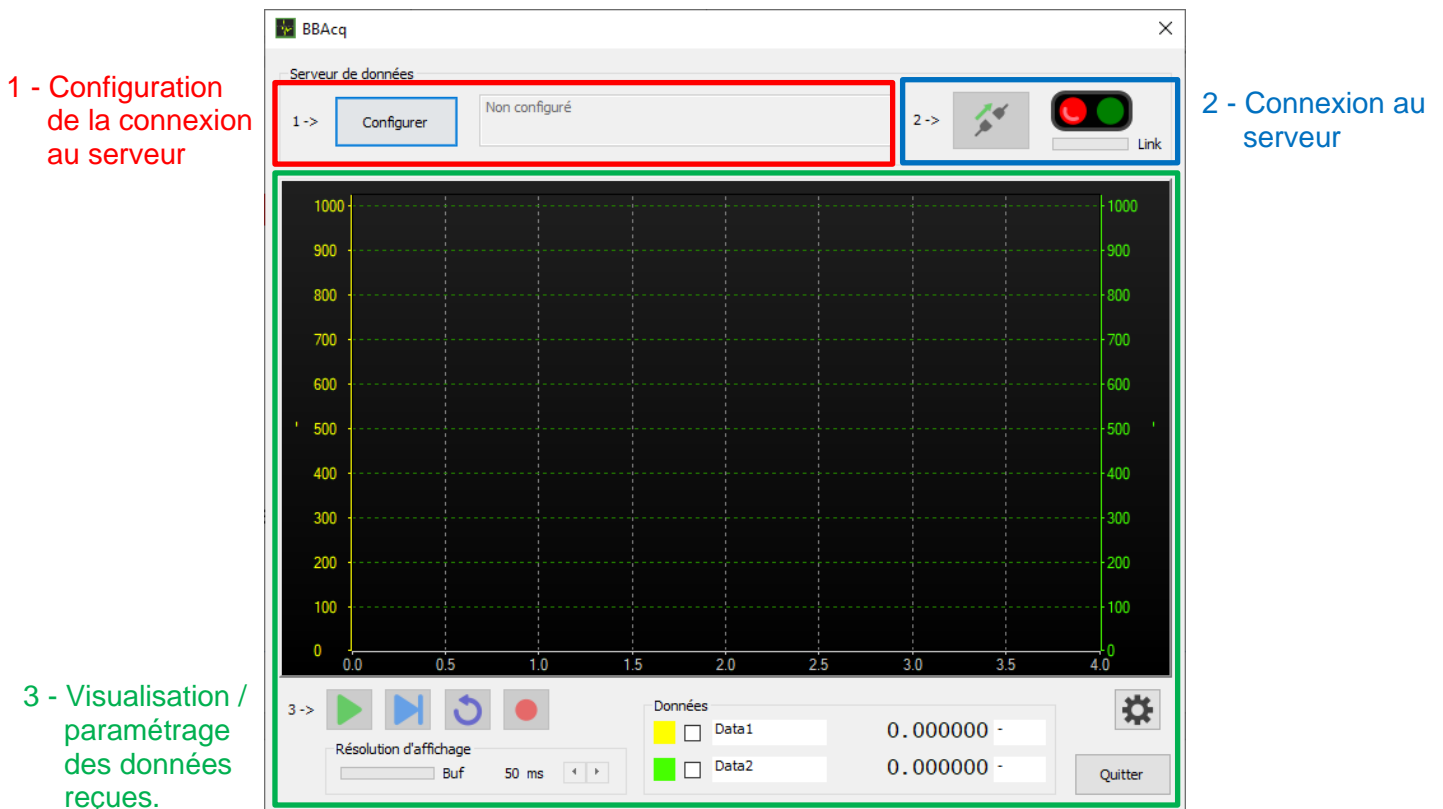
La vue d'ensemble ci-dessous illustre les deux solutions développées :

- une **solution locale** : à travers une liaison série à 250 000 bauds ;
- une **solution réseau** : à travers un réseau local, avec serveur DHCP ou non.



2 Fenêtre principale de l'interface BBAcq

La fenêtre initiale de l'interface est la fenêtre de configuration/connexion/visualisation :



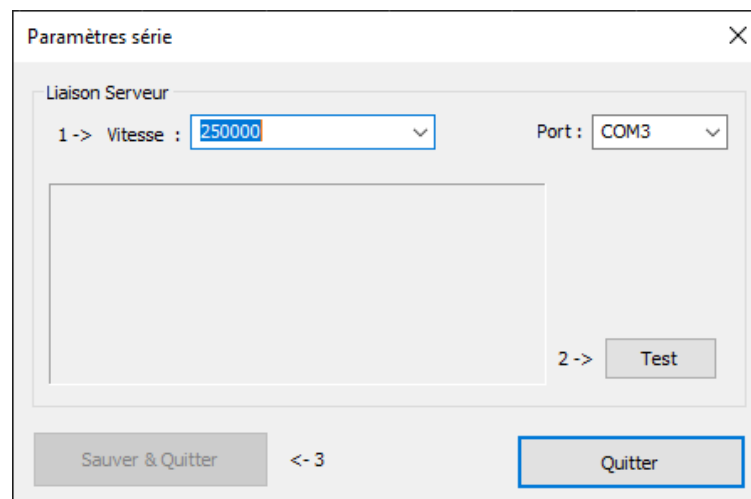
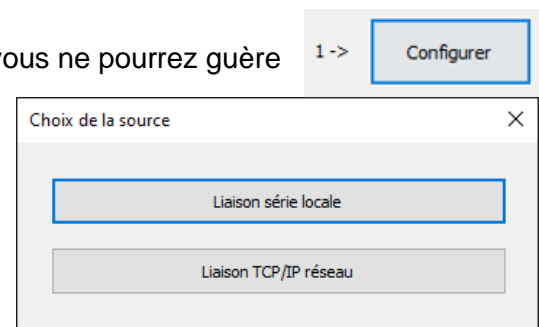
3 Connection avec le serveur

Cliquer en premier lieu sur le bouton "Configurer" (vous ne pourrez guère faire grand chose d'autre...).

Choisir selon le type d'interface utilisée, celle correspondant :

3.1 Liaison série locale

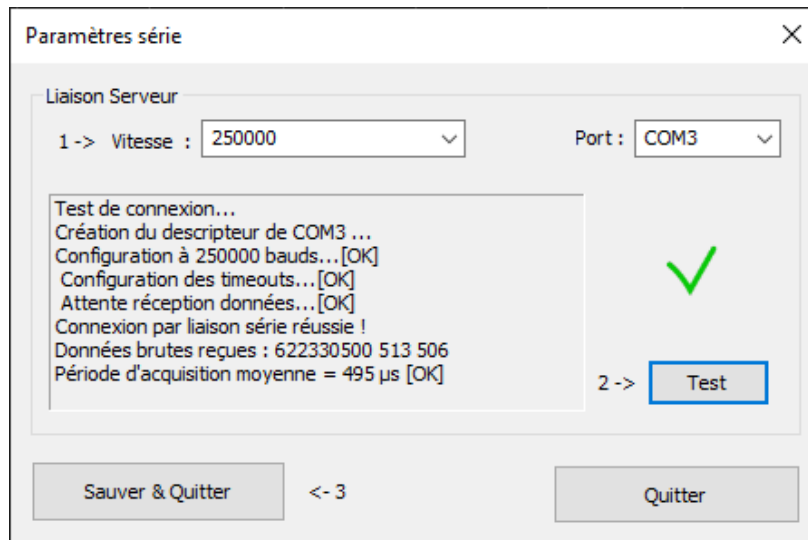
La fenêtre suivante s'ouvre :



Par défaut, la vitesse de transmission est définie à 250 000 bauds (ne pas modifier), et les ports COM disponibles sont proposés.

Remarque : Si aucun port ne s'affiche cela signifie que votre interface n'est pas accessible, donc non reliée au PC par son câble USB.

Sélectionner le bon port COM puis effectuer un test de connexion : la connexion ne sera validée que si le test est passé avec succès, à savoir que le port COM transmet bien des données valides à la vitesse de 250 000 bauds, toutes les 500 µs (à 20% près).



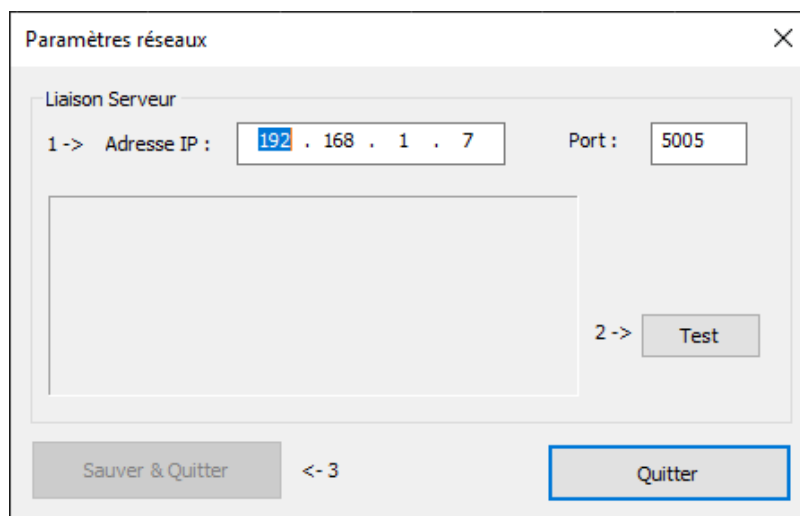
Si le test ne réussit pas c'est soit :

- que le port COM sélectionné n'est pas le bon ;
- qu'il y a un problème en réception série (problème matériel).

Si le test réussit, **valider la configuration en appuyant sur le bouton "Sauver & Quitter"**, puis allez directement à la section "[lecture des données](#)".

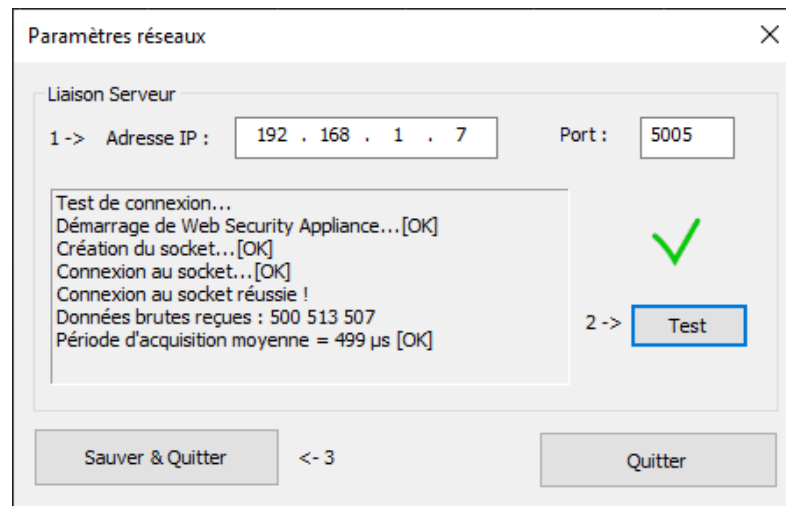
3.2 Liaison TCP/IP réseau

La fenêtre suivante s'ouvre :



Par défaut, une adresse et un port sont déjà renseignés (lors de la toute première connexion, ceux donnés sur l'impression-écran ci-dessus).

Renseignez l'adresse IP de la carte Arduino ainsi que le port de communication (si différents de ceux proposés) **puis effectuer un test de connexion** : la connexion ne sera validée que si le test est passé avec succès, à savoir que des trames TCP/IP sont bien récupérées contenant des données valides, à la période de 500 µs (à 20% près).



Si le test ne réussit pas c'est soit :

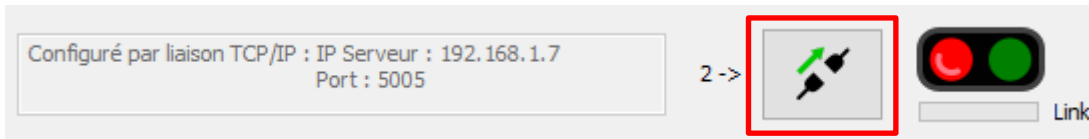
- que l'adresse IP et/ou le port renseignés ne sont pas corrects ;
- qu'il y a déjà un client connecté à l'interface (la communication est mono-client, pour des simplifications de traitement).

Si le test réussit, **valider la configuration en appuyant sur le bouton "Sauver & Quitter"**.

Remarque : à chaque nouvelle connexion client/serveur, le temps d'acquisition est remis à 0 côté serveur, ce qui n'est pas le cas avec la liaison série (le serveur ne sait pas quand quelqu'un se connecte, et envoie ses trames à qui veut bien les lire...).

4 Lecture des données

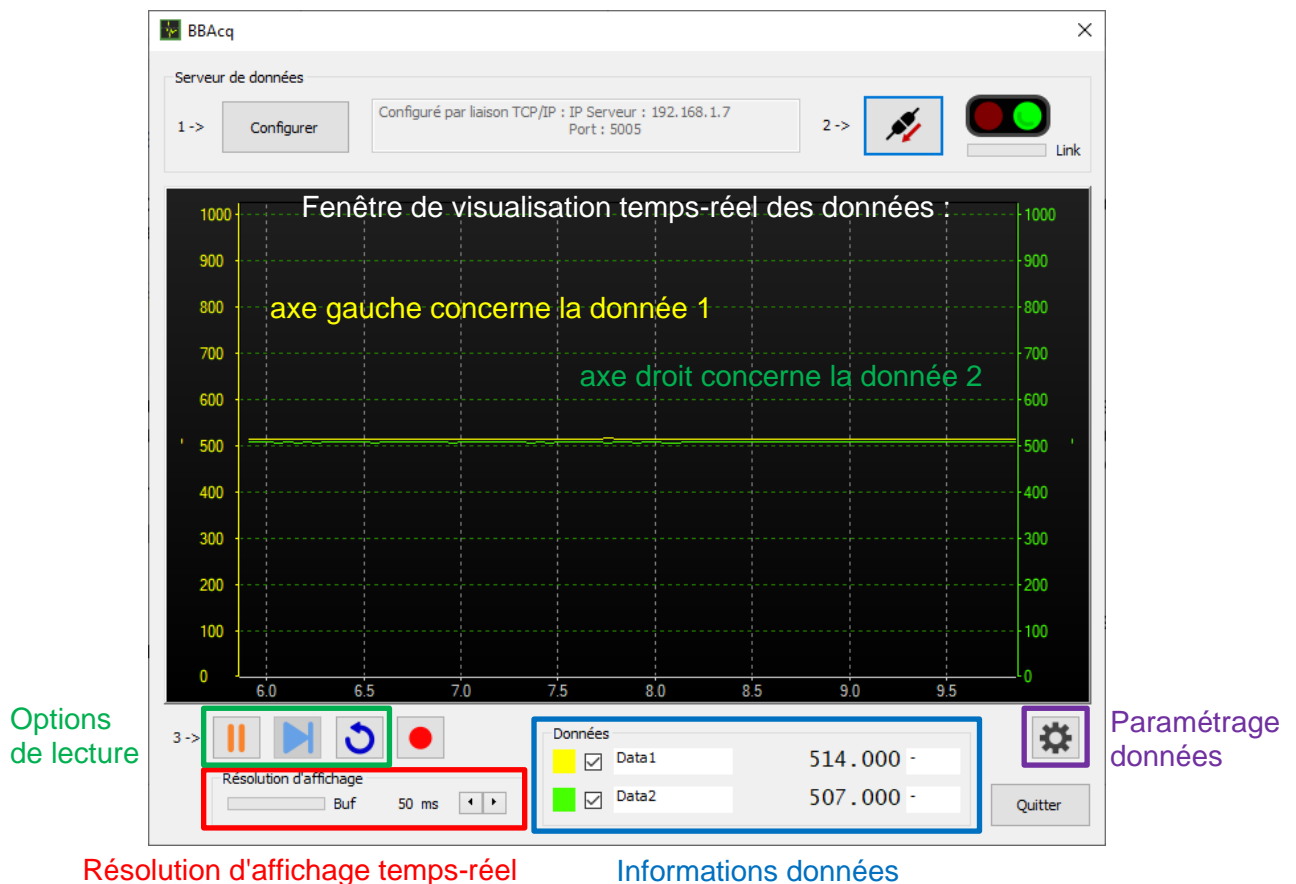
La configuration retenue et validée apparaît dans l'onglet, et il est possible maintenant de démarrer la connexion par l'appui sur le bouton suivant :



Le feu passe au vert, indiquant la circulation des données, et l'activité du flux de donnée est visible par une petite barre de progression indiquant la charge du buffer en réception (il y a plus de fluctuation avec une liaison série qu'avec une liaison réseau, de par la gestion asynchrone des données en réception).

Les données s'affichent alors en temps-réel, sur une **durée de 4 s constante** (choix d'interface, durée estimée suffisante pour observer la plupart des grandeurs physiques d'un système mécatronique, où les constantes de temps vont généralement de la ms à la s, les fréquences de résonance allant du Hz au kHz).

Remarque : lors d'une connexion réseau, celle-ci se déconnecte automatiquement au bout de 2 minutes. Cela libère la ressource automatiquement pour que d'autres clients se connectent.

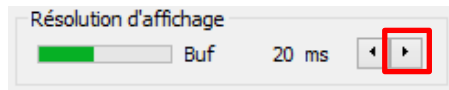


4.1 Affichage en temps-réel

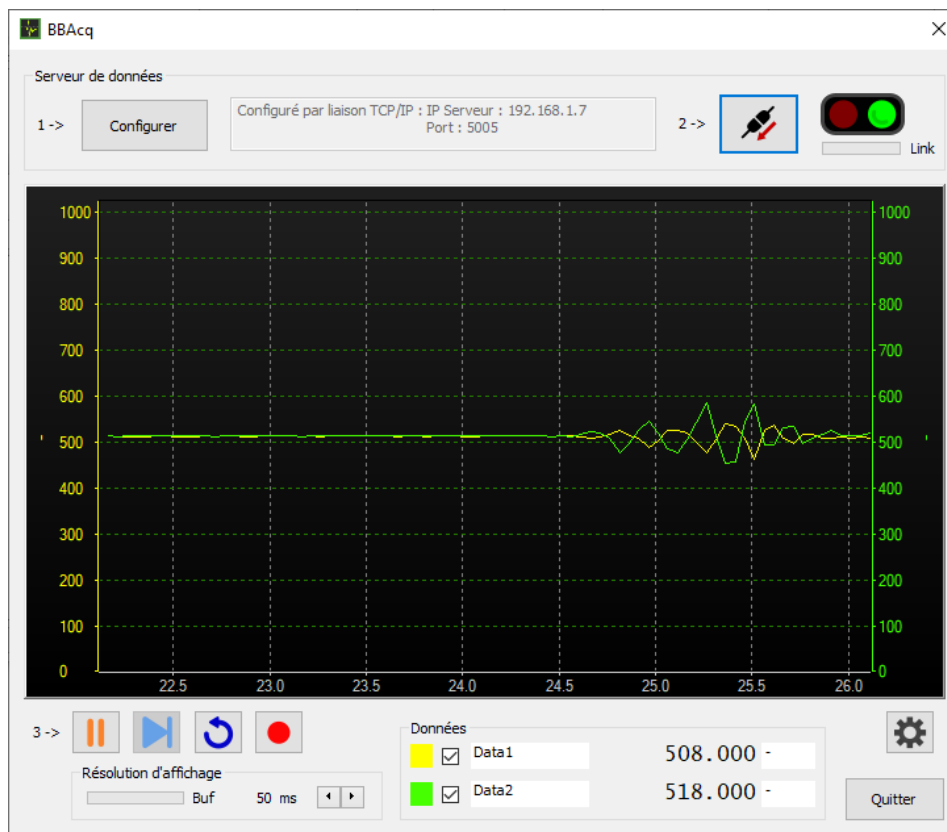
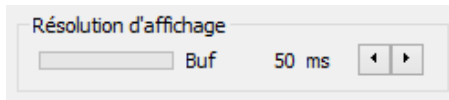
La première des choses à vérifier, est que l'affichage s'effectue bien en temps-réel : l'onglet « Résolution d'affichage » affiche le buffer des données affichées (sous-échantillonnage du buffer de réception), et si jamais celui-ci se remplit régulièrement, cela signifie que le traitement lié à l'affichage graphique dure plus de temps que la période d'affichage sélectionnée (problème d'une FIFO où on écrit plus vite qu'on ne lit). Dans ce cas, il faut augmenter cette période par l'utilisation des flèches gauche et droite prévues à cet effet.

Exemple :

À 20 ms, le buffer se remplit : les données affichées sont retardées, des données seront perdues quand le buffer sera plein. Il faut augmenter la période.



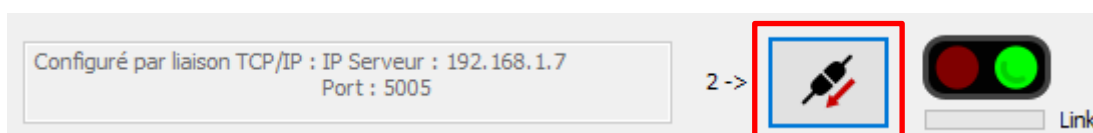
À 50 ms, le buffer se vide puis reste constamment vide : on affiche suffisamment vite pour qu'il n'y ait pas de bufferisation (on lit plus vite qu'on n'écrit), synchronisation et affichage temps-réel assuré.



Affichage temps-réel : les variations générées apparaissent instantanément par la droite.

Remarque : cette période ne concerne que l'affichage, qui est gourmand en terme de ressources graphiques, indépendamment de la bufferisation de toutes les données, qui elle s'effectue toujours à 500 μ s sur une durée de 4 s (soit 8000 échantillons max). Cette fonctionnalité permet de s'affranchir des performances graphiques variables d'un ordinateur à l'autre. Les valeurs courantes seront de 50 ms pour les moins performants à 10 ms pour les plus performants, et bien souvent 20 ms suffira.

Pour se déconnecter, même temporairement, il suffit d'appuyer à nouveau sur le bouton de connexion, qui a changé d'apparence (cela libère le serveur en version réseau pour permettre à un autre client de se connecter) :

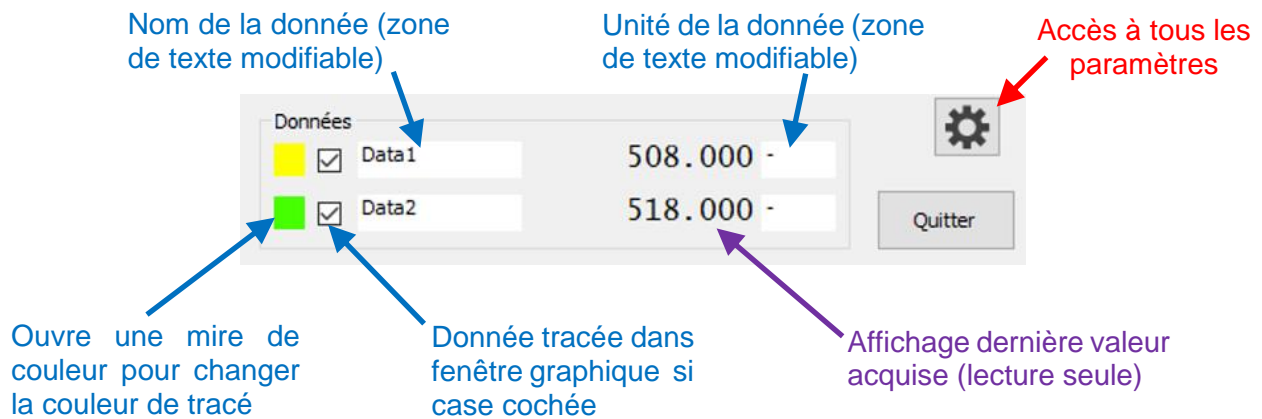



la configuration reste toujours valide, et au prochain appui la connexion s'effectuera de nouveau (sauf si, dans le cas d'une liaison réseau, un autre client s'est connecté entre temps, ce qui aura pour effet de bloquer durant quelques secondes l'interface, avant de rendre la main sans réaliser la connexion).

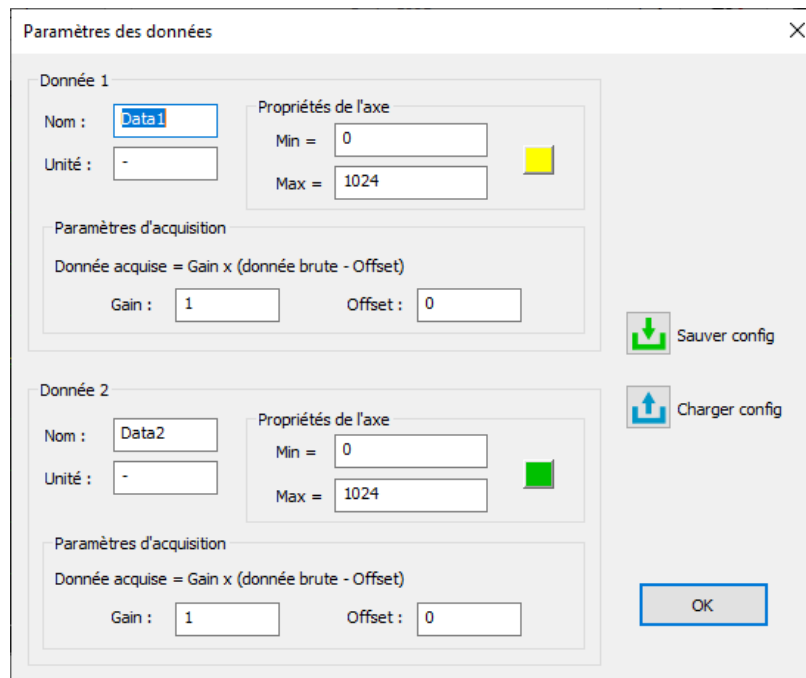
Remarque : si pendant une connexion établie il y a rupture physique de liaison, le logiciel se mettra en attente bloquante, et il faudra très probablement forcer la fermeture du logiciel avant de le relancer.

4.2 Formatage des données

Le nom, l'unité et la couleur du tracé des données sont directement modifiables dans l'onglet prévu à cet effet, où la dernière valeur acquise est affichée en permanence :



Pour avoir accès à tous les paramètres (en particulier le gain, l'offset, l'axe d'affichage) il faut cliquer sur le bouton de paramétrage , qui ouvre la fenêtre suivante :



En plus de pouvoir définir le gain, l'offset, les valeurs min et max des axes (et les caractéristiques précédentes), il est également possible de sauvegarder le paramétrage actuel ou de charger un paramétrage précédemment enregistré (fichiers d'extension ".ini", comme le fichier "config.ini" situé dans le répertoire du logiciel, et qui enregistre automatiquement la configuration en quittant le programme).

Ainsi, pour deux données qui sont des accélérations et après calibration (mesures au repos et soumis à la pesanteur), la configuration suivante est obtenue, où maintenant les valeurs affichées sont des accélérations exprimées en m/s^2 :

Paramètres des données

Donnée 1

Nom : Propriétés de l'axe
 Unité : Min = Max =
 Paramètres d'acquisition
 Donnée acquise = Gain x (donnée brute - Offset)
 Gain : Offset :

Donnée 2

Nom : Propriétés de l'axe
 Unité : Min = Max =
 Paramètres d'acquisition
 Donnée acquise = Gain x (donnée brute - Offset)
 Gain : Offset :

et dont la visualisation en temps-réel est la suivante :



4.3 Options de lecture

Les 3 boutons en bas à gauche de la fenêtre permettent de modifier la "lecture" des données.



Mode lecture



Mode Pause

Selon que l'on est en mode lecture ou pause, ils offrent les fonctionnalités suivantes :

	Bouton Lecture/Pause : bouton unique permettant de basculer d'un mode à l'autre.
	Bouton RAZ : permet de remettre à 0 l'axe des temps. Disponible uniquement en mode lecture.
	Bouton Single Shot : permet d'effectuer une seule acquisition de 4 s, en remettant l'axe des temps à 0. Disponible uniquement en mode pause.

Par défaut, lors d'une connexion, l'affichage se met automatiquement en mode lecture.

4.3.1 Zoom/Dézoom

Il est possible de zoomer/translater, à l'aide de la souris :

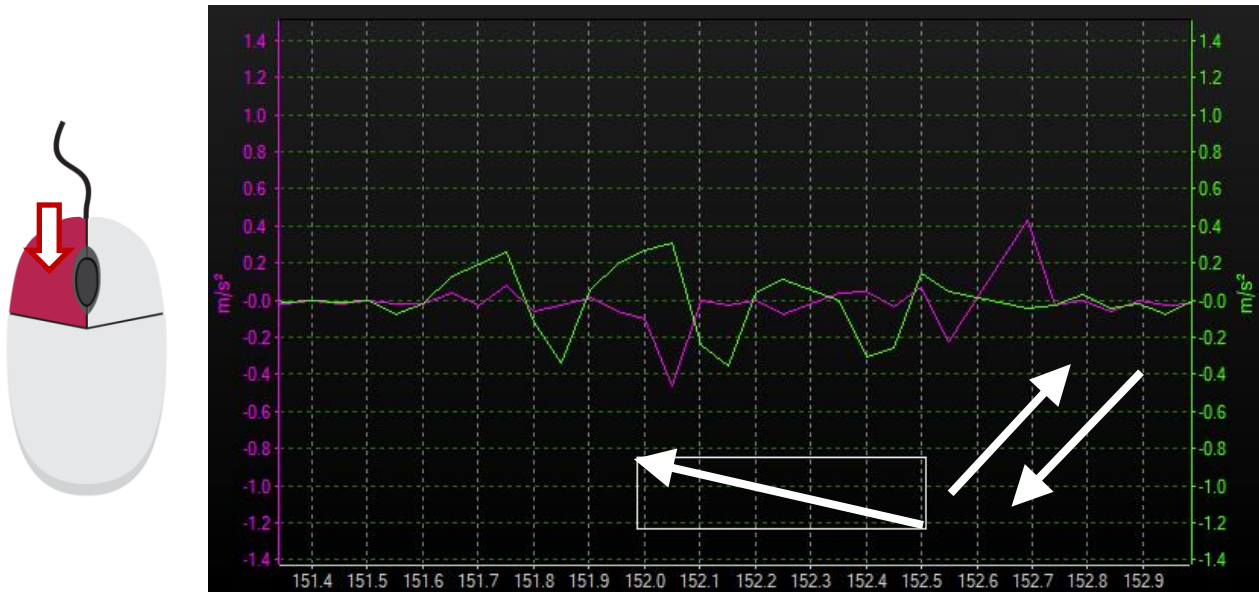
- **Zoom** : s'effectue en maintenant le **clic gauche** souris et en **dessinant une fenêtre du haut gauche vers le bas droit** :



Zoom

Remarque : le zoom temporel ne fonctionnera qu'en mode pause, non en mode lecture, la fenêtre se redimensionnant automatiquement à 4 s. En revanche celui en amplitude fonctionnera en permanence, même en mode lecture.

- **Dézoom** : s'effectue également en maintenant le **clic gauche** souris, mais en **dessinant toute autre type de fenêtre** (de haut en bas, du haut droit vers le bas gauche). Remet les axes d'origine, peu importe la fenêtre dessinée :



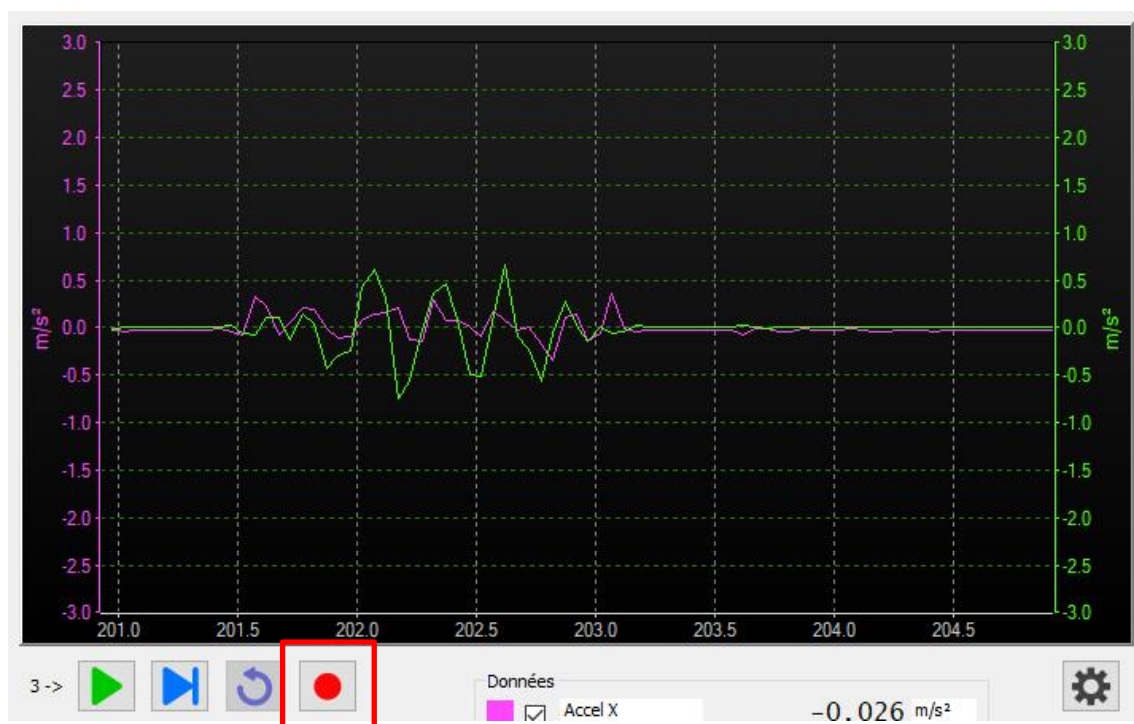
Dézoom (Fenêtre d'origine)

- **Translation** : s'effectue en maintenant le **clic droit** souris, les mouvements de la souris déplaçant maintenant les axes (mais préférer modifier les axes dans les paramètres).

5 Traitement des données (mode enregistrement)



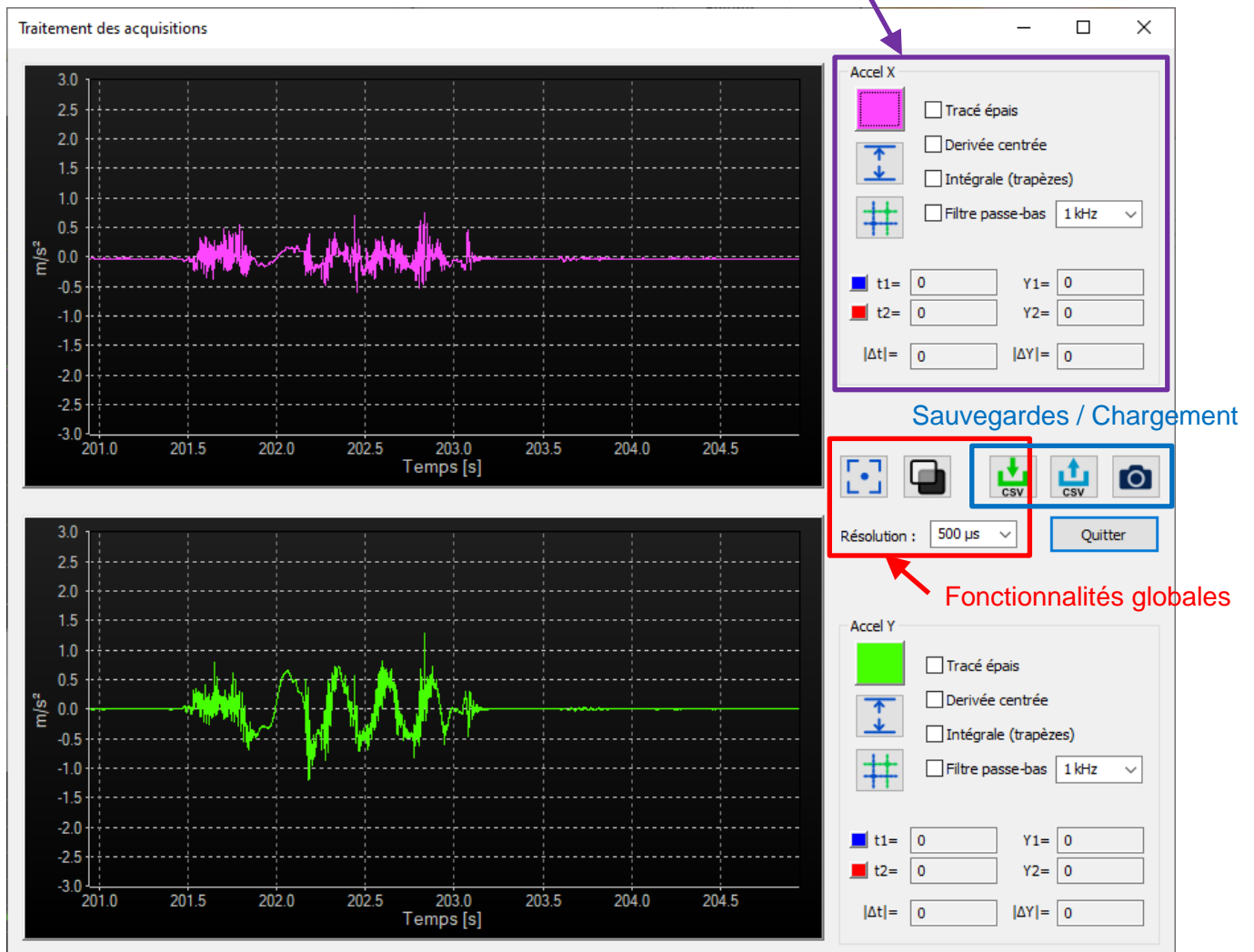
Le dernier bouton à droite des options de lecture est le bouton d'enregistrement des données, qui permet de récupérer toutes les données acquises durant les 4 dernières secondes afin de les traiter et les exporter pour une exploitation plus poussée (comme importer les données dans un tableur, dans un logiciel de simulation de type Matlab/Scilab, ...).



5.1 Environnement de travail

La fenêtre suivante s'ouvre alors, **redimensionnable si besoin**, où les deux données sont séparées afin d'être traitées indépendamment :

Fonctionnalités propres à chaque donnée





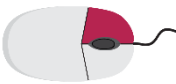
5.2 Fonctionnalités d'affichage

Les fonctionnalités globales s'appliquent sur les deux graphiques simultanément, à savoir :

	Bouton Reset : remet chaque fenêtre à l'échelle d'origine, sur tous les axes.
	Bouton Noir & Blanc : permet de basculer d'un environnement graphique sur fond noir à un environnement sur fond blanc.
	Bouton Résolution d'affichage : permet de choisir une période d'échantillonnage plus grande, et de réaliser ainsi un sous-échantillonnage des données.

Remarque : par défaut, à l'ouverture l'affichage est en full HD (si l'on peut dire...), soit pour un échantillonnage à 500 µs.

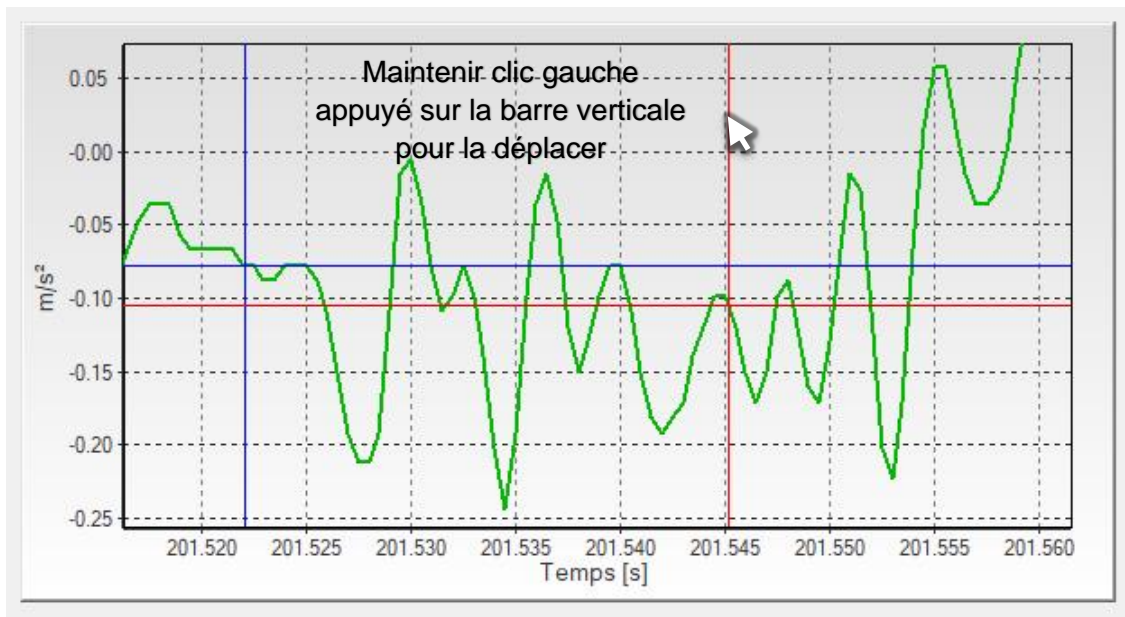
D'autres fonctionnalités sont disponibles, ne s'appliquant elles qu'à la donnée concernée :

	Bouton Couleur : ouvre une mire de couleurs permettant de changer la couleur du tracé.
	Bouton Mise à l'échelle : permet d'adapter automatiquement l'échelle verticale des axes aux données visualisées. L'axe temporel demeure inchangé.
	Zoom/Dézoom : pas de bouton spécifique, le principe étant le même que celui décrit précédemment .
<input type="checkbox"/> Tracé épais	Tracé épais : si la case est cochée, permet d'augmenter l'épaisseur du tracé ou non (utile en cas d'utilisation des fonctions avancées).

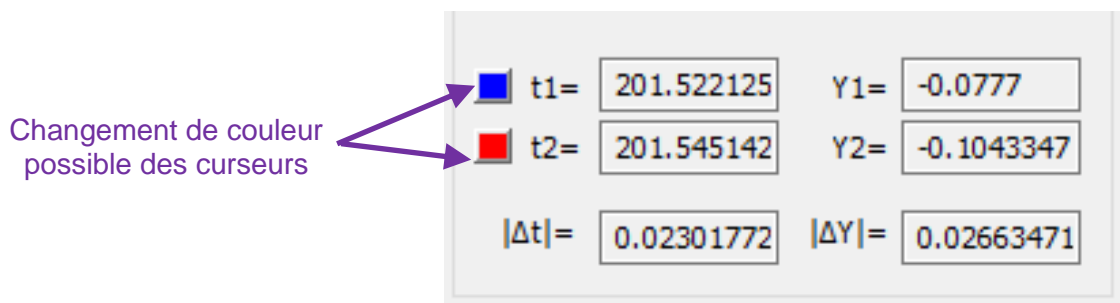
5.3 Utilisation des curseurs



Pour chaque donnée visualisée, l'appui sur ce bouton affiche deux curseurs qui suivent automatiquement le tracé, et qui peuvent être déplacés en cliquant sur la barre verticale de celui-ci.



Durant ce mode de fonctionnement, seules quelques fonctionnalités graphiques restent disponibles, l'intérêt étant de ne rester dans ce mode que le temps d'effectuer les mesures nécessaires (absolues ou différentielles), qui s'affichent en temps-réel dans la fenêtre de donnée :

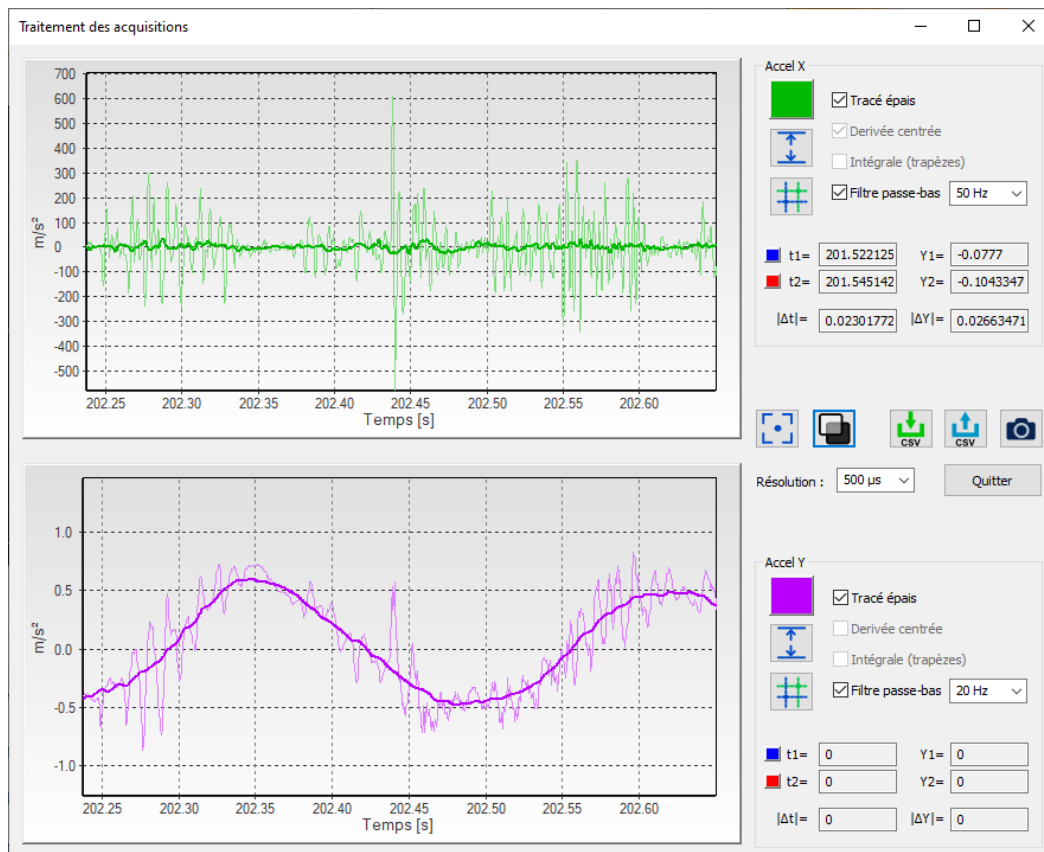


Une fois les mesures finies, on sort de ce mode en appuyant à nouveau sur le bouton de curseur.

5.4 Fonctions avancées (traitements des données)

Trois fonctionnalités permettent de réaliser des traitements sur chaque donnée, à savoir :

<input type="checkbox"/> Dérivée centrée	<p>Fonction Dérivée : si case cochée, réalise une dérivation approchée par la formule symétrique, telle que :</p> $\left. \frac{ds(t)}{dt} \right _{t=t_n} \cong \frac{s(t_{n+1}) - s(t_{n-1}))}{t_{n+1} - t_{n-1}}$ <p>Cette fonction est exclusive à la fonction intégrale.</p>
<input type="checkbox"/> Intégrale (trapèzes)	<p>Fonction Intégrale : si case cochée, réalise une intégration par la méthode des trapèzes, telle que :</p> $\int_{t_0}^{t_n} s(t) dt \cong I(t_n) = I(t_{n-1}) + \frac{(s(t_n) + s(t_{n-1})) \cdot (t_n - t_{n-1}))}{2}$ <p>avec $I(t_0) = 0$.</p> <p>Cette fonction est exclusive à la fonction dérivée.</p>
<input type="checkbox"/> Filtre passe-bas 1 kHz	<p>Fonction Filtre passe-bas : si case cochée, réalise un filtre RIF moyenneur centré, tel que :</p> $f(t_n) = \frac{1}{N} \sum_{i=-N/2+1}^{N/2} s(t_{n+i})$ <p>Qui réalise une approximation d'un filtre passe-bas de fréquence de coupure (inférieure) à F_e/N à déphasage minimal (fonction de transfert réelle en sinus cardinal...).</p> <p>Le tracé se superpose au tracé précédent, qui devient légèrement transparent.</p>






Exemple de combinaison des traitements possibles. Le tracé épais permet un meilleur rendu.


Remarque : lors d'un filtrage, les fonctions dérivée et intégrale sont indisponibles. Pour changer de fonction il faut donc désactiver le filtrage, effectuer les changements, puis le réactiver. De même, la fréquence de coupure peut être modifiée lors d'un filtrage, mais les modifications ne seront appliquées que si l'on désactive puis réactive le filtrage.

5.5 Enregistrement des données

Pour finir, cette interface ne serait pas complète sans la possibilité d'exporter les données acquises pour une utilisation spécifique (traitements sur tableur, comparaison avec résultats issus d'une simulation, ...).

Le format retenu est le format CSV, la séparation des différents champs étant effectuée par des points-virgules. Il est possible de plus de charger des données préalablement enregistrées pour les traiter à nouveau dans l'interface, et d'exporter les graphiques obtenus en image.

	Bouton Save CSV : enregistre les données de la fenêtre d'affichage, données brutes + données traitées (un message d'avertissement le rappelle avant charge enregistrement).
	Bouton Load CSV : charge des données brutes (uniquement, même si des données traitées sont présentes dans le fichier) préalablement enregistrées avec le bouton "Save CSV".
	Bouton Save Image : enregistre chaque fenêtre graphique dans deux images séparées. Les formats disponibles sont : BMP, JPEG, PNG ou GIF.

Remarque sur le chargement des données : si l'on veut uniquement retraiter d'anciennes données, il est possible d'ouvrir directement l'interface de traitement par le bouton  au démarrage du logiciel, même si aucune connexion ni donc aucune donnée n'a été acquise.