

Banc de test pour drones lourds

Soutenance de projet

EI4-SAGI

2017 - 2018

CHOUTEAU Quentin
COURSON Léo
JAUNEAU Vincent

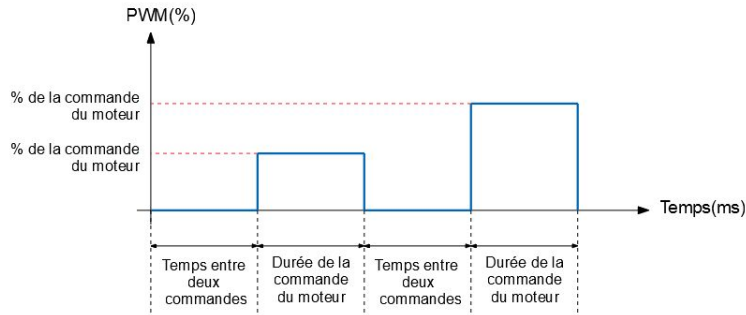


Contexte

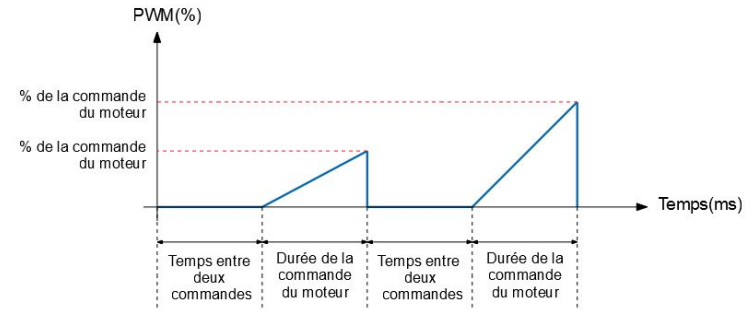
- Objectif → Réaliser un banc de test :
- Contrôler la turbine
 - Mesurer force de poussée, intensité et tension
 - Traiter les données
 - Sauvegarder et Afficher les données sur une interface graphique

Les tests à réaliser

- Test en échelon



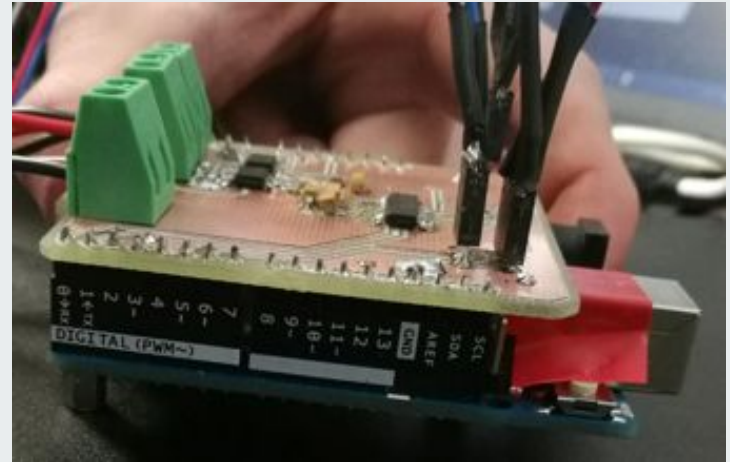
- Test en rampe



Hardware

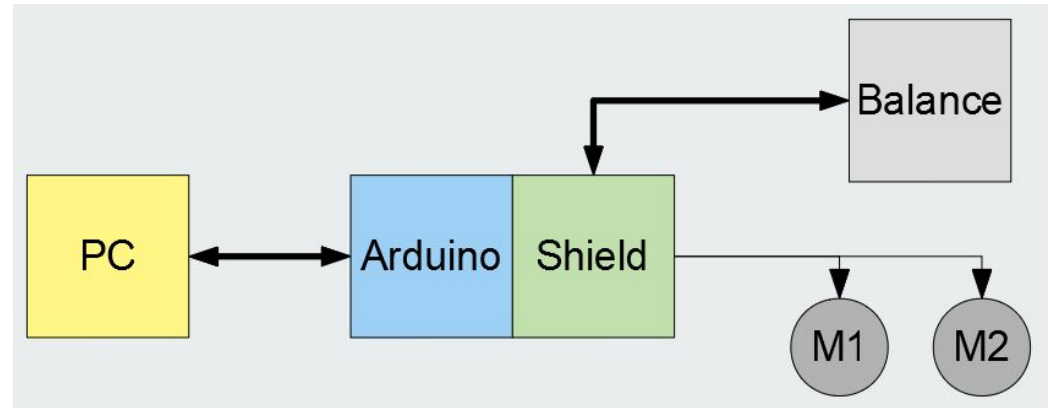
Rôle dans projet

- Etablir la connexion entre l'arduino et les moteurs.
- Récupérer les informations des moteurs.



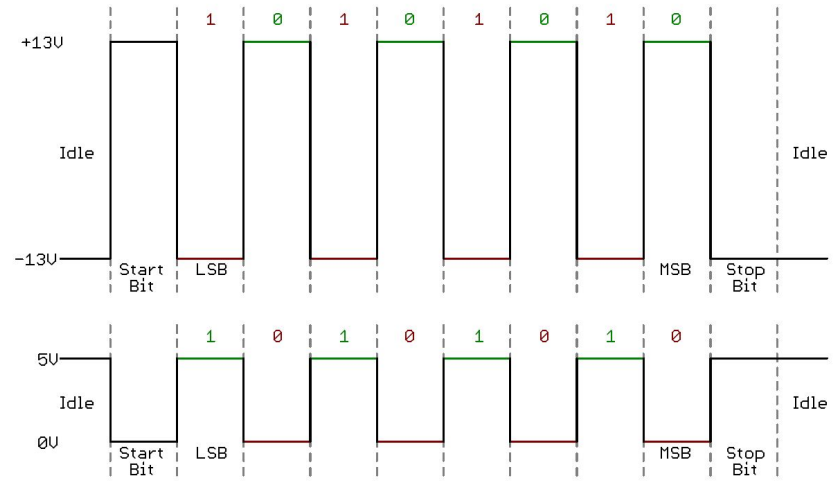
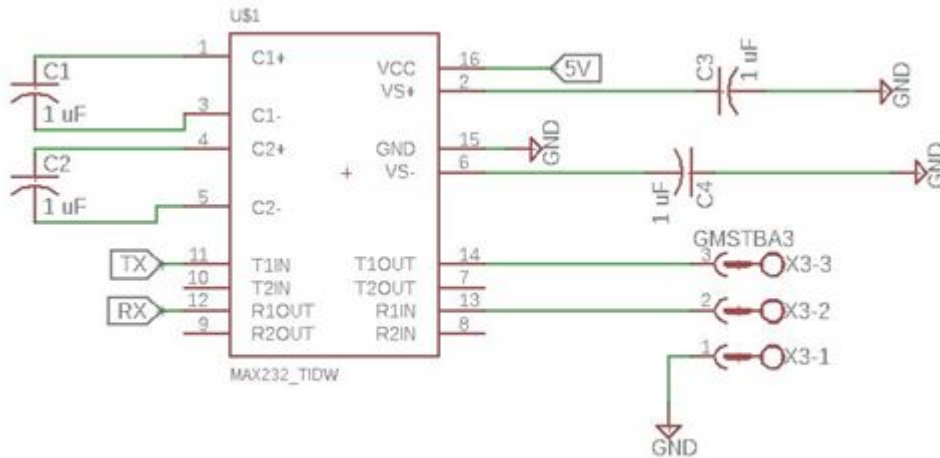
Shield Arduino

- Conversion RS232 → TTL
- Valeur du courant
- Valeur de la tension
- Commande des moteurs

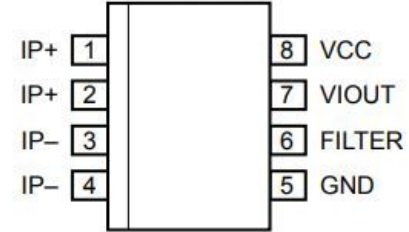


Conversion RS232 -> TTL

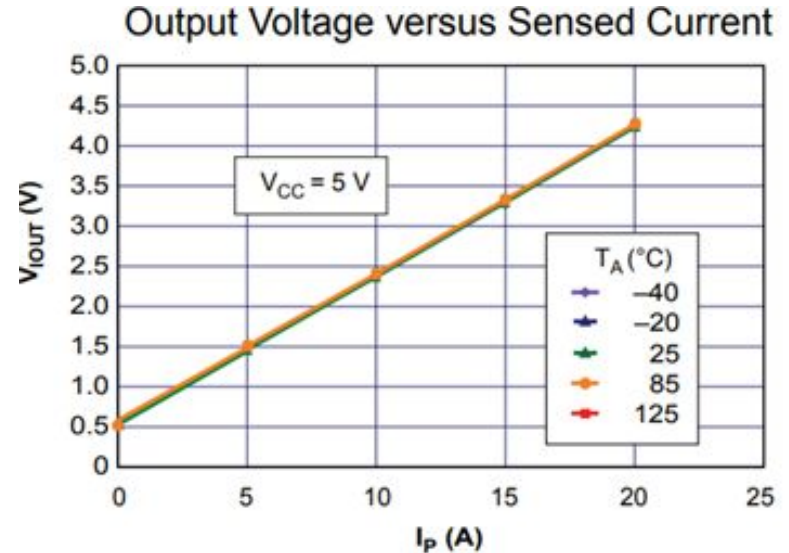
- Alimentation : +5 V
- $V_{s+} = +10\text{ V}$ et $V_{s-} = -10\text{ V}$



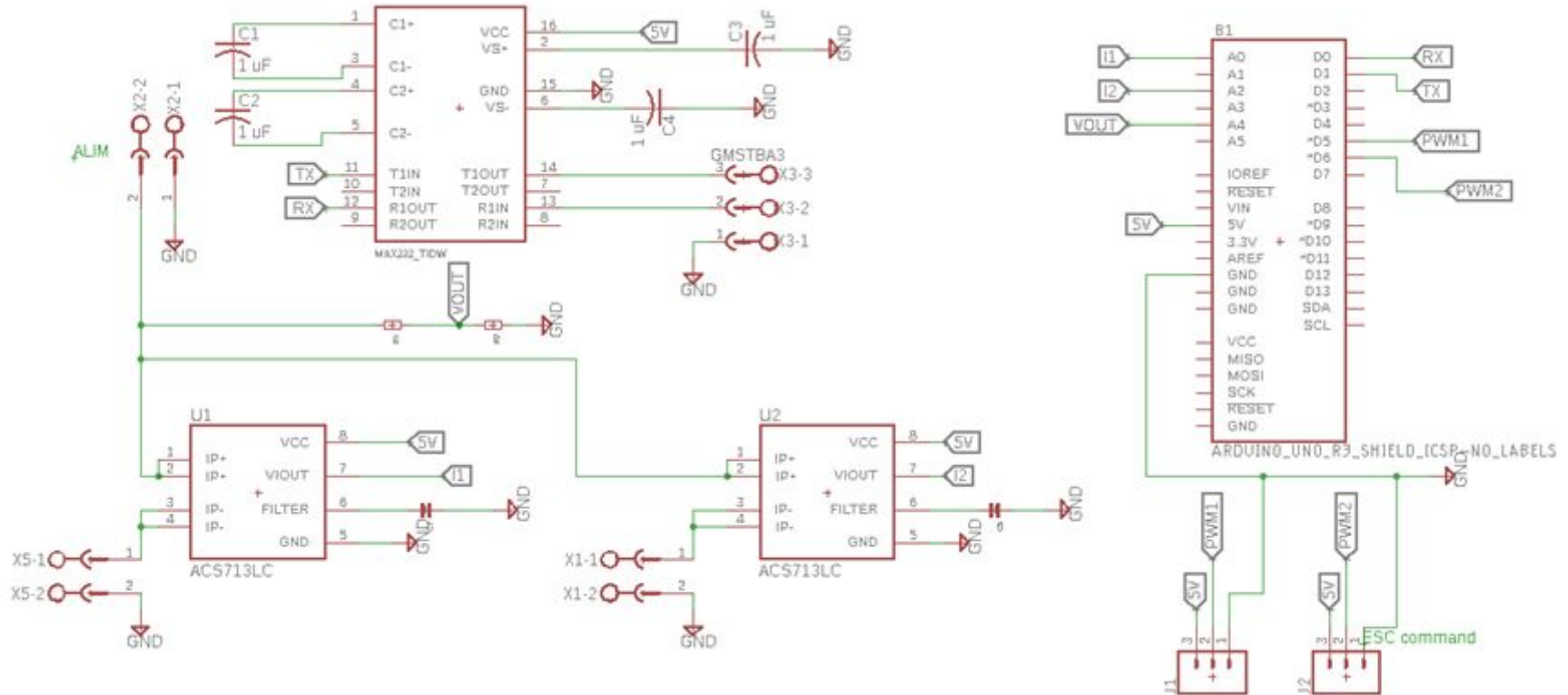
Courant (ACS713)



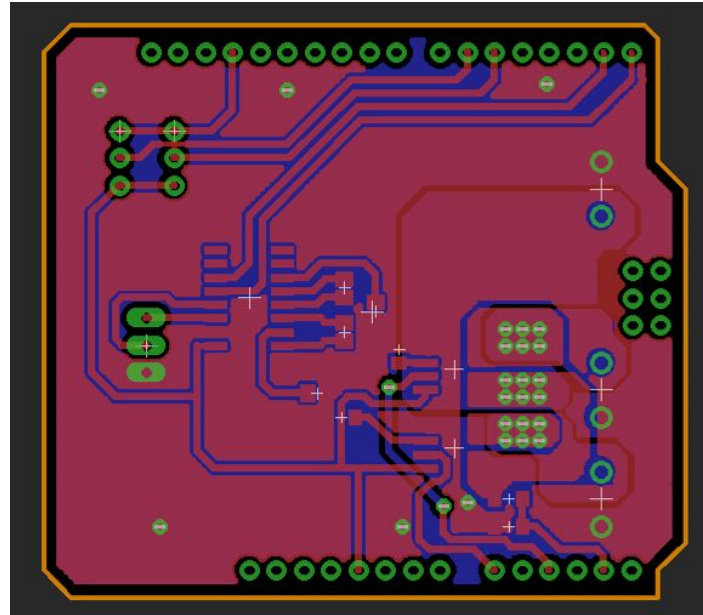
- Courant circule entre IP+ et IP-
- Image du courant sur Viout
- Alimentation : 5V
- $I_P = (V_{iout} - 0.5) / 0.1875$







Schématique

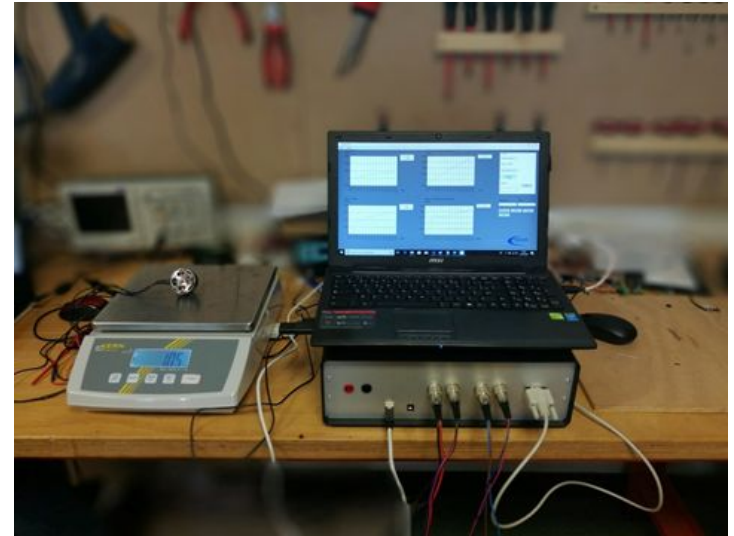


CAO et résultat

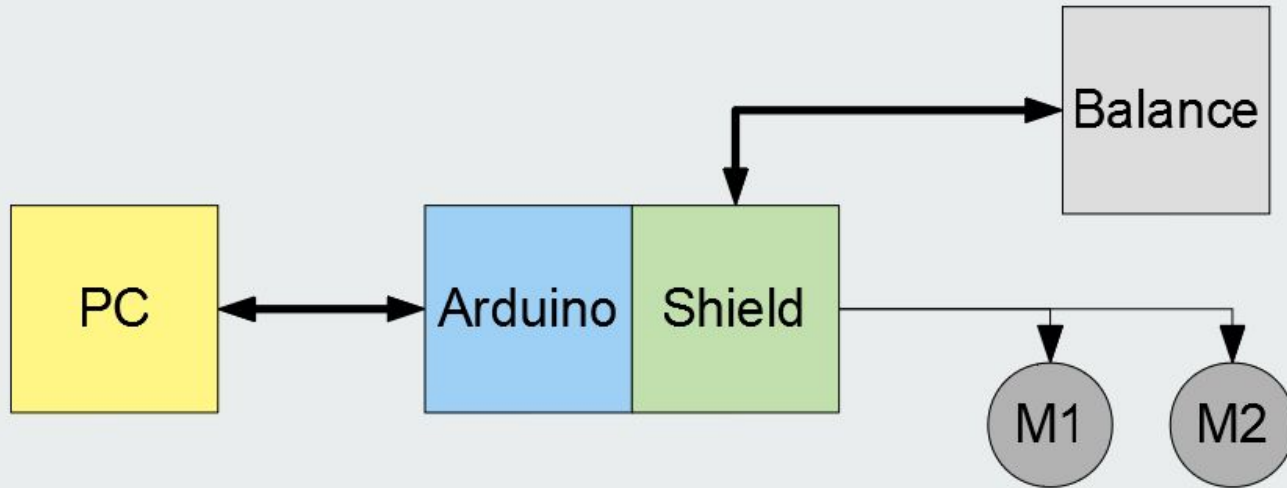


Connectique et boîtier

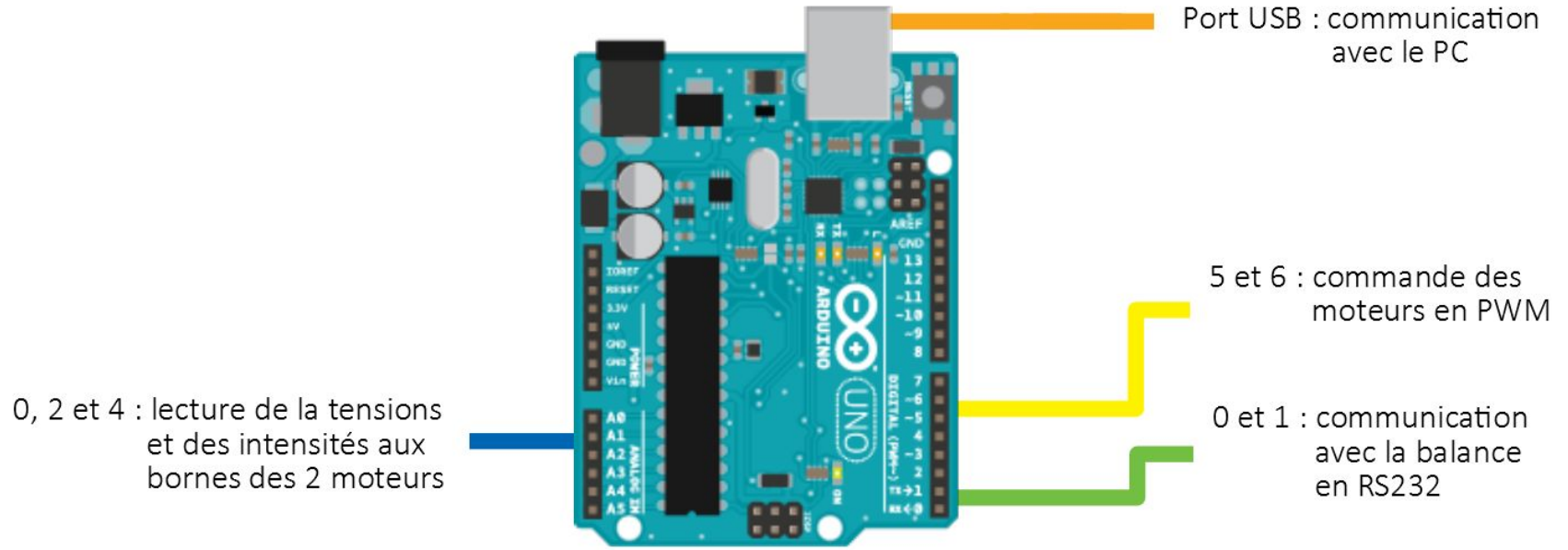
Rôle	Type de connecteur	Quantité	Image
Commande moteur	Connecteur DIN 3 pins	2 (1/moteur)	
Puissance moteur	Connecteur DIN 4 pins	2 (1/moteur)	
Connexion balance	Connecteur DB9	1	
Alimentation	Fiche Banane	2	



Arduino



Configuration

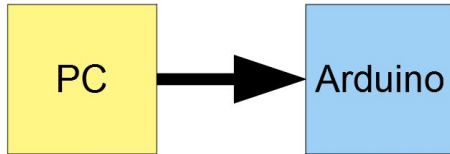


Communication avec le PC

Deux types de communication

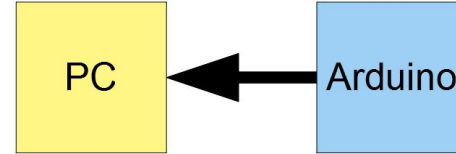
PC → Arduino

- Envoi de la procédure de test



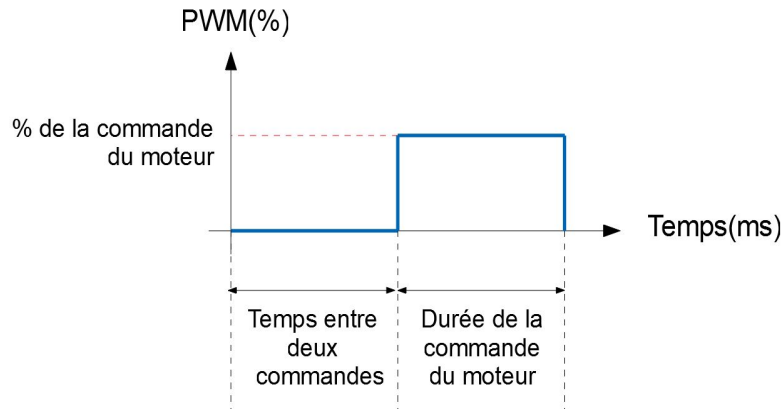
Arduino → PC

- Envoi à interval régulier des données récupérées lors du test



Communication avec le PC

Réception de la procédure de test (PC → Arduino)



Commande des moteurs

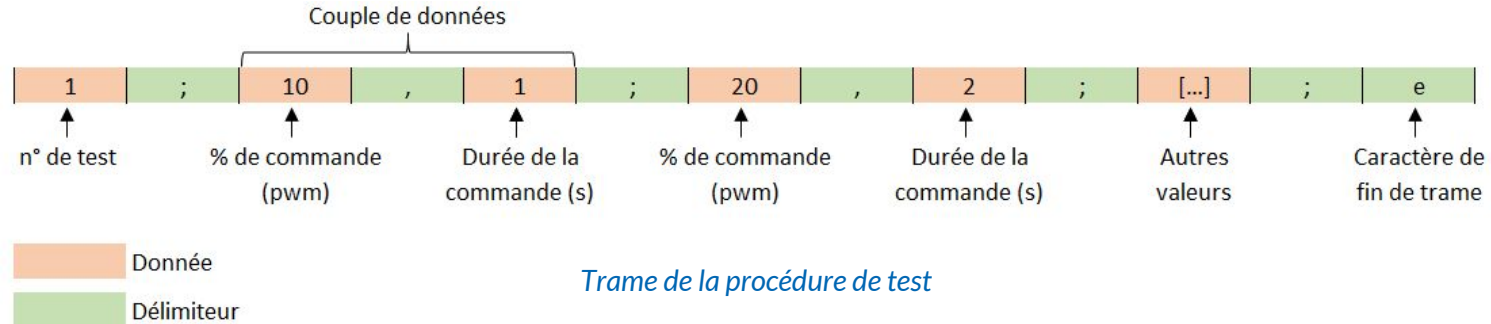
Informations nécessaires :

- Type du test (rampe / échelon)
- Temps entre 2 commandes
- Pourcentage **ET** Durée de la commande des moteurs

Communication avec le PC

Réception de la procédure de test (PC → Arduino)

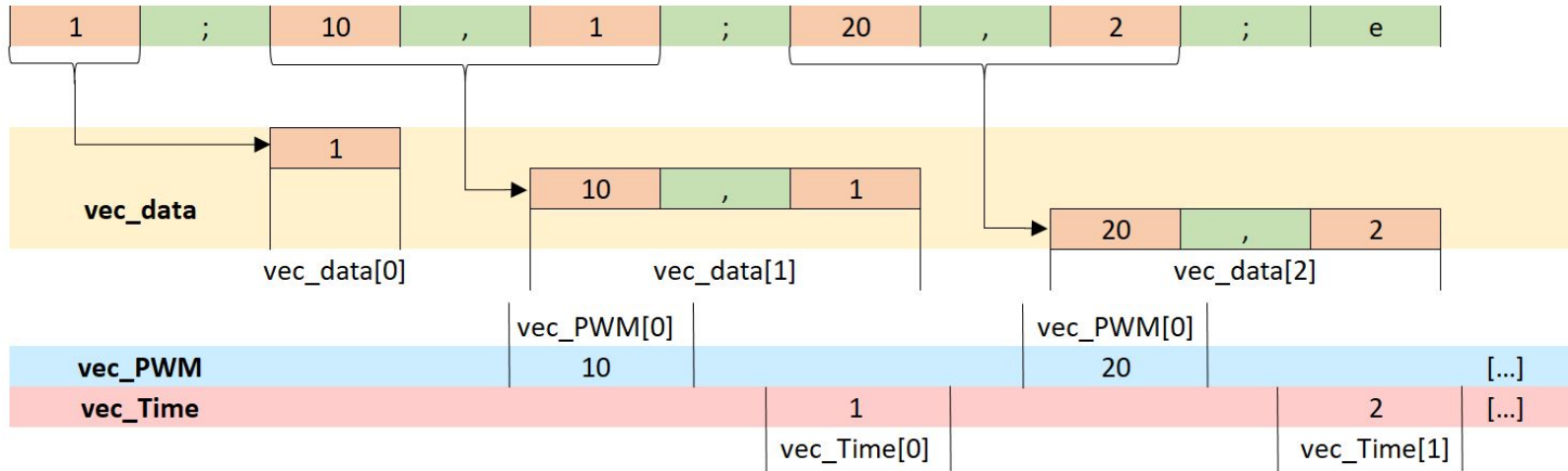
- N° de test : Type de test **ET** Temps entre 2 commandes
- Couples de données : Pourcentage **ET** Durée d'une commande



Communication avec le PC

Réception de la procédure de test (PC → Arduino)

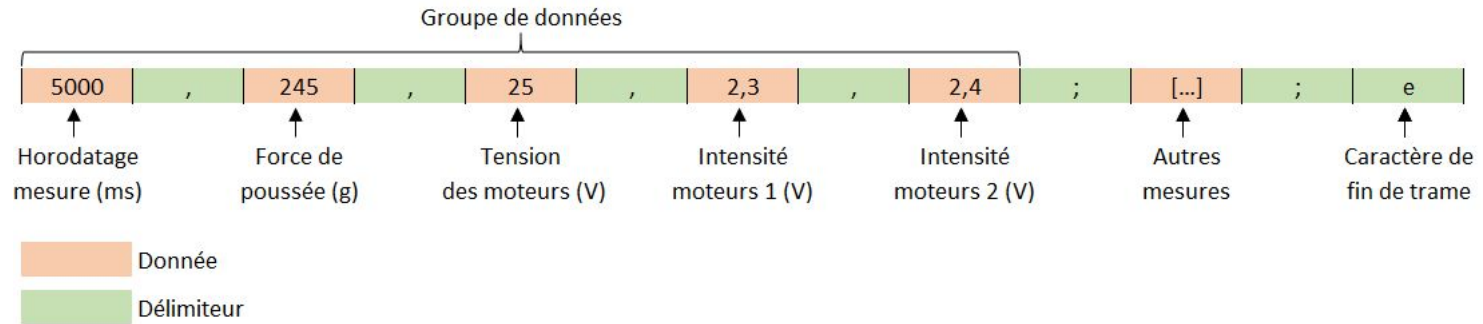
- strtok() : Fractionne une chaîne de caractère suivant des délimiteurs



Sauvergarde de la procédure

Communication avec le PC

Envoi des données récupérées (Arduino → PC)



Trame d'envoi des données

Communication avec le PC



Envoi des données récupérées (Arduino → PC)

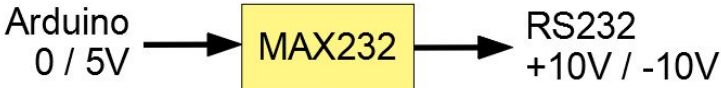
- Envoi périodique
 - Fonction `millis()` → renvoie le temps écoulé depuis le démarrage de l'arduino

```
if((millis() - previous_time_send) >= time_interval_send && in_command)
{
    sendToPC();
}
```

Appel de la fonction d'envoi des données

Communication avec la balance

Communication en RS232 → via broches 0 (Tx) et 1 (Rx)

- Carte électronique :  Arduino 0 / 5V → MAX232 → RS232 +10V / -10V

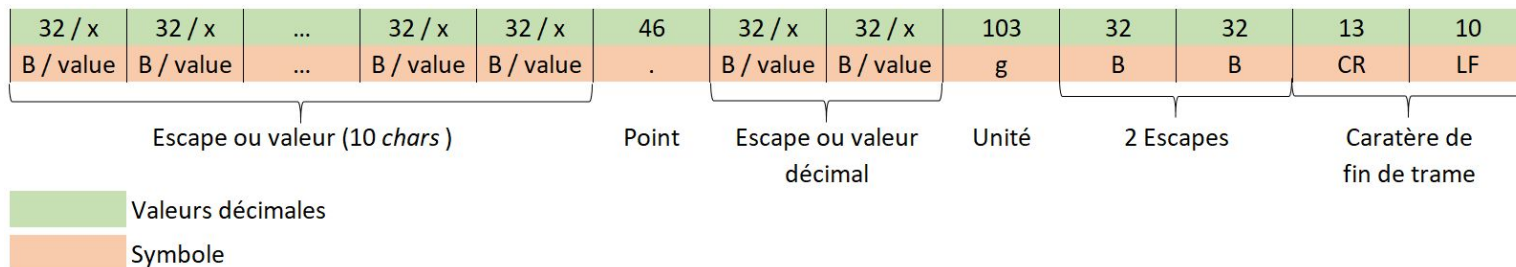
Configuration de la balance

- 19200 bauds (Maximum)
- Ordre télécommande (Envoie du poids sur envoi d'un 'W')
- Plus de tare automatique

Communication avec la balance

Trames renvoyées par la balance

Trame renvoyant le poids



Trame d'erreur

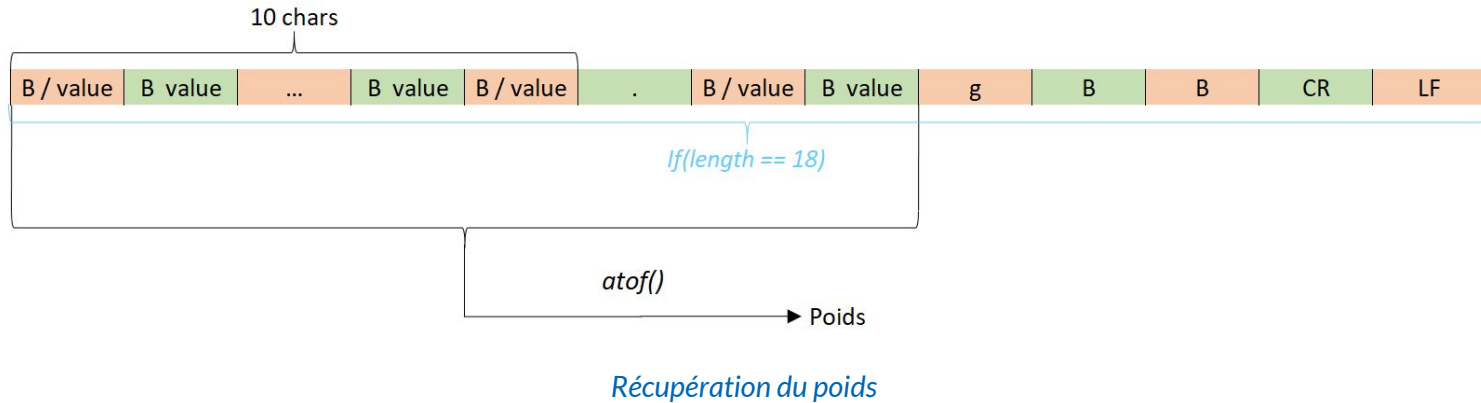
x	13	10
Code d'erreur	CR	LF

- Indique une mauvaise configuration de la balance

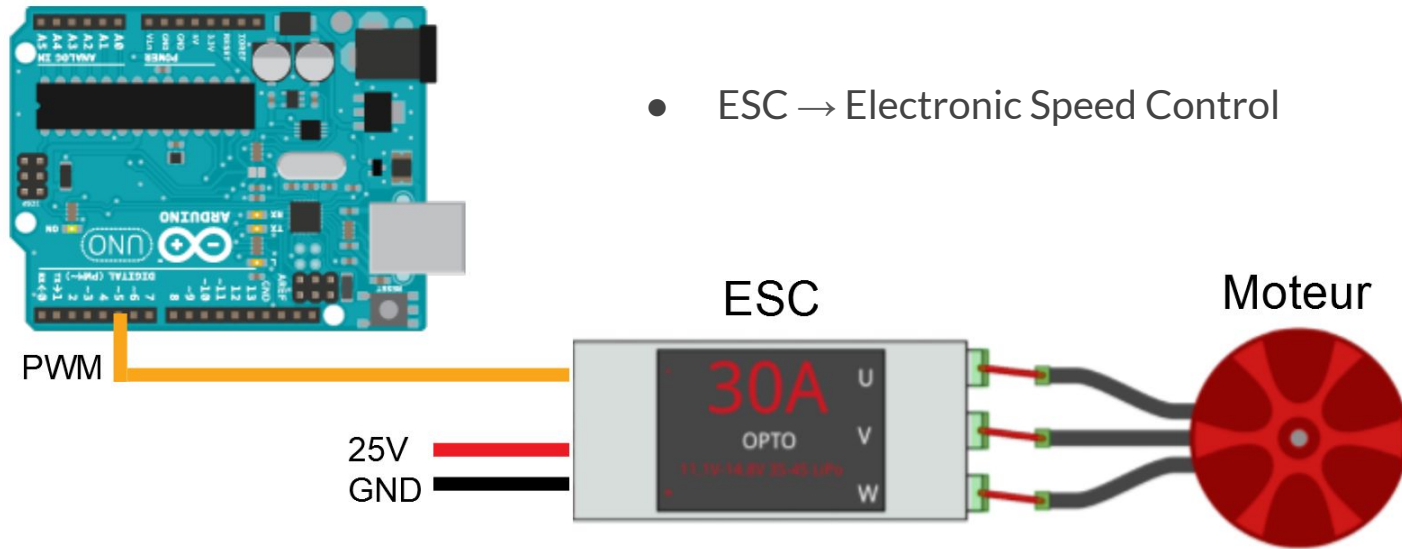
Communication avec la balance

Poids récupéré juste avant l'envoi des données (= requête : 'W')

- Rien n'est renvoyé au PC → nouvel essai de la récupération du poids à la prochaine itération



Contrôle des moteurs



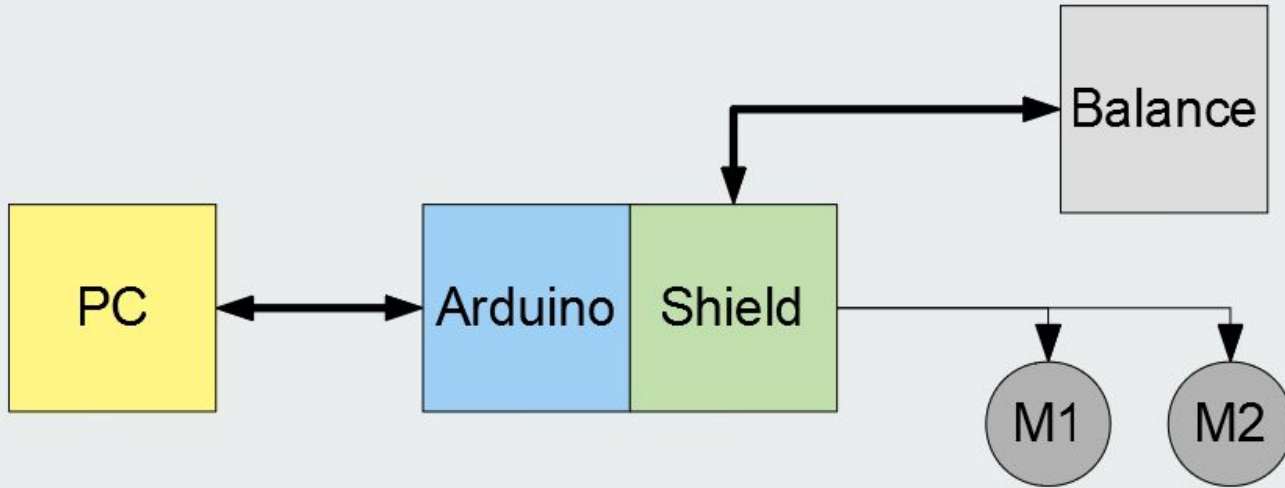
Contrôle des moteurs

- Commande en échelon
 - Commande → variable *time_until_next_control*
 - Arrêt → *vec_Time* (temps de commande)

```
//CONTROL MOTOR
if((millis() - init_time) >= time_until_next_control && !in_command && nb_test==0)
//STOP MOTORS
if((millis() - previous_time_control) >= vec_Time[id_command]) && in_command)
```

Arrêt et contrôle des moteurs

Interface



IHM



IHM

DAO :

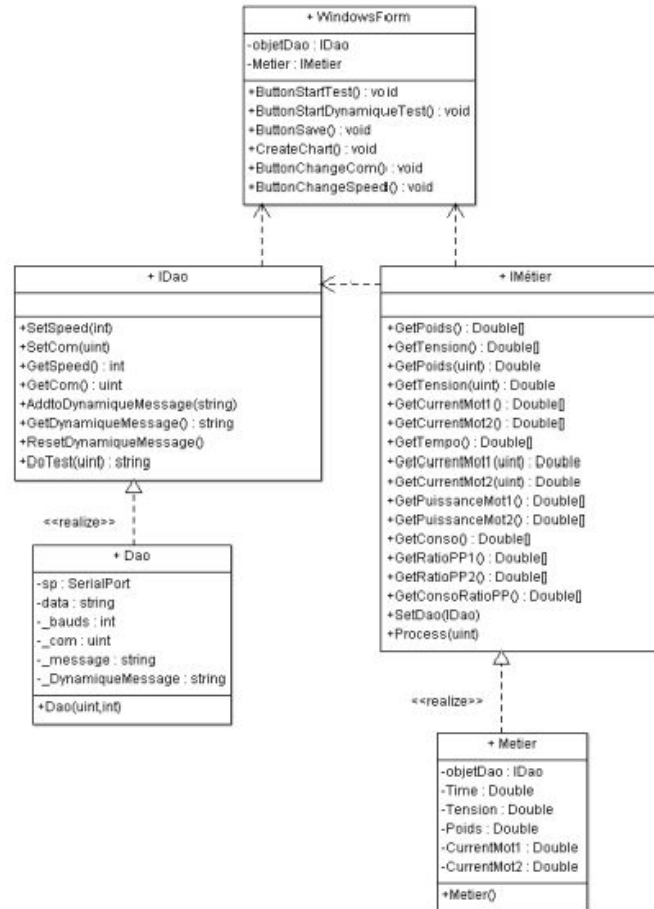
- Fonctions de configuration
- Réalisation des tests

Métier:

- Récupération des données
- Calcul sur ces données

UI:

- Affichage des données
- Sauvegarde



IHM

Controls :

- Affichage de la configuration
- Choix / lancement du test
- Enregistrement des données

Test dynamique:

- Préparation du test
- Lancement du test

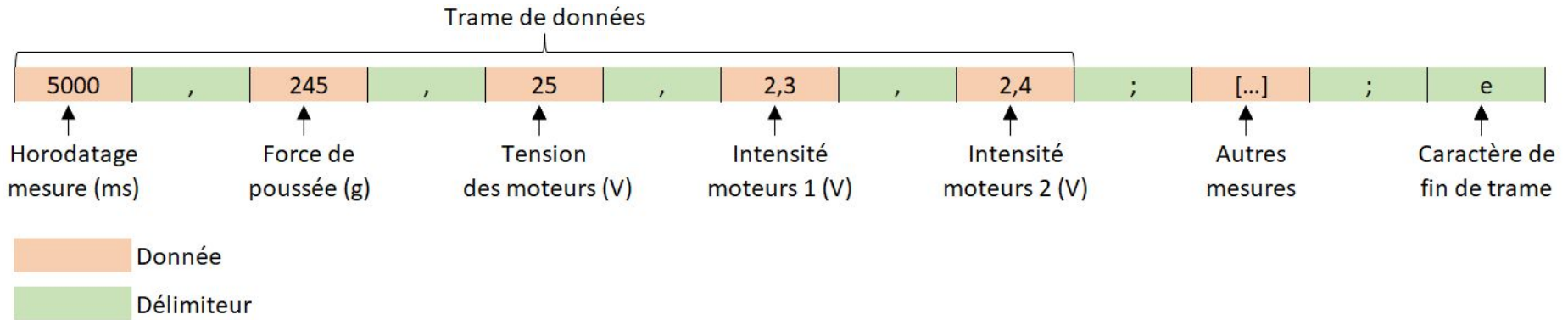
The 'Controls' window displays the following configuration options:

- Communication port : 3
- Speed : 115200
- Test EGR (dropdown menu)
- Start test (button)
- Filename : (text input field)
- Save (button)
- csv (dropdown menu)

The dynamic test configuration panel includes the following elements:

- %max power (text input field)
- Tempo (text input field)
- Add to test (button)
- Show (button)
- Start (button)
- RAZ (button)

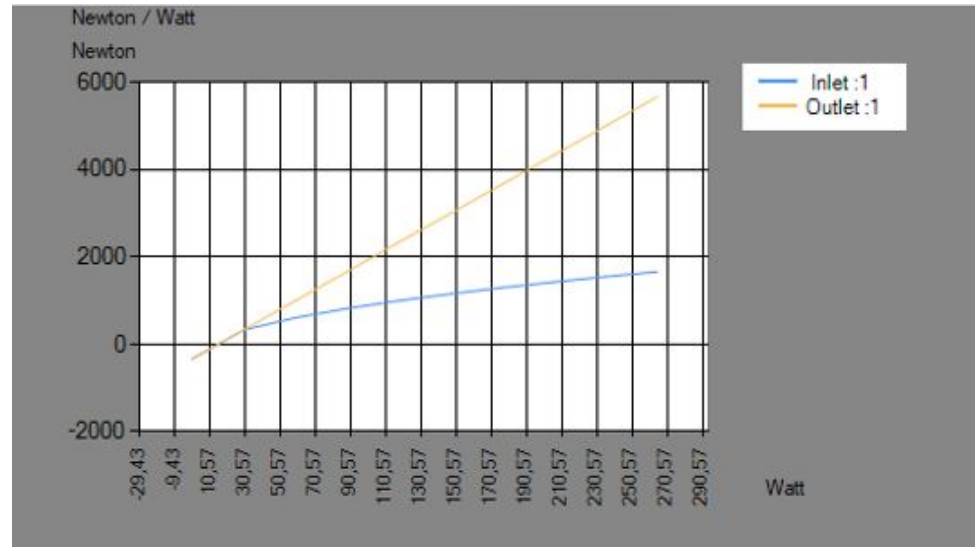
IHM



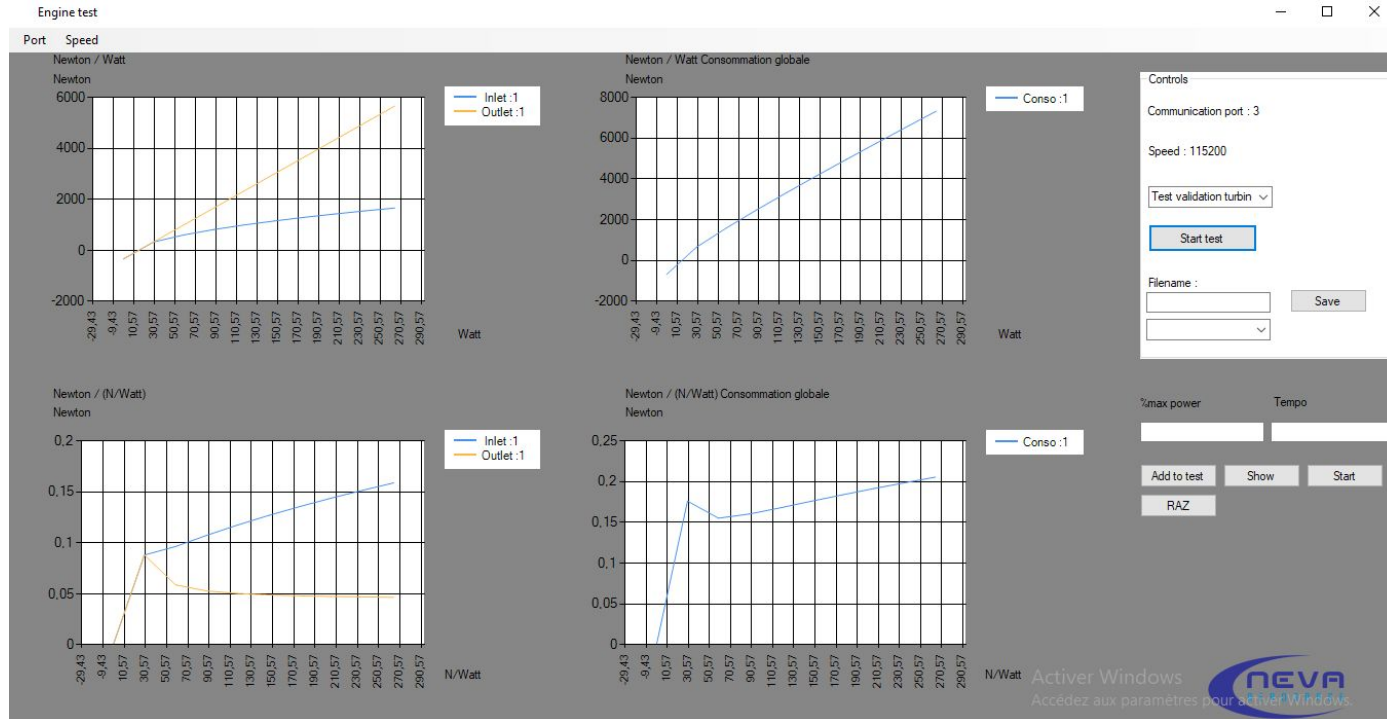
IHM

Affichage :

- Utilisation de chart, item de winform
- 4 affichage :
 - Newton / Watt
 - Newton / Watt global
 - Newton / ratio PP
 - Newton / ratio PP global



Résultat final





Conclusion