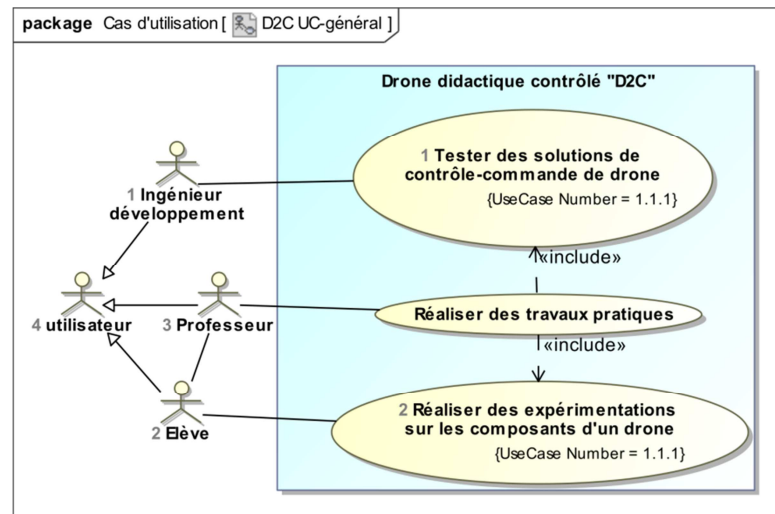
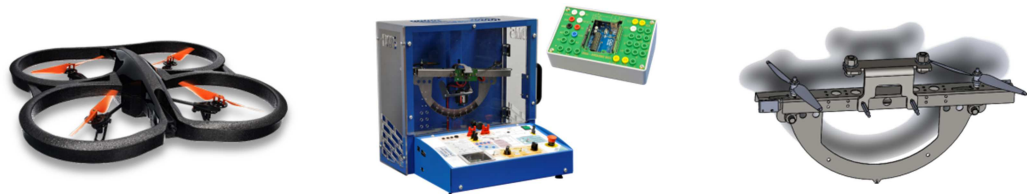


DRONE DIDACTIQUE CONTRÔLE D2C



Les drones sont utilisés pour réaliser des prises de vue aériennes, faire des courses, ou d'autres activités de loisir. Le drone didactique contrôlé permet d'analyser les lois de commandes des moteurs, ou encore de déterminer les efforts de poussée des hélices.

Lors de la mise en énergie du drone, la centrale inertielle doit transmettre au calculateur ses valeurs de référence. Ces valeurs doivent être relevées lorsque la centrale est parfaitement stabilisée en position horizontale : il est donc nécessaire de réaliser une procédure d'initialisation.

Sur un drone réel, l'utilisateur doit poser la machine quelques instants sur une surface horizontale pour réaliser cette phase d'initialisation. **Sur le drone didactique**, c'est le « servo-bloqueur » qui positionne automatiquement le balancier à l'horizontale quelques instants, après la mise en énergie, pour que le microcontrôleur puisse récupérer les valeurs de référence de la centrale inertielle.

Problématique :

On s'intéresse à la procédure permettant d'initialiser le drone :

Comment concevoir et implanter la procédure permettant d'initialiser le système ?

Le but de ce TP est d'essayer de répondre à cette question.

1 DÉCOUVERTE – MANIPULATION – OBSERVATION – DESCRIPTION

Objectif : S'approprier le fonctionnement du D2C et analyser la phase d'initialisation – 20 minutes

Cette première partie nécessite la lecture préalable des fiches : « Mise en œuvre » (6) « Ingénierie systèmes » (7).

Activité 1

Mettre en service le D2C en utilisant la fiche « Mise en œuvre ».

Le diagramme de séquence de la fiche « Ingénierie systèmes » présente l'enchaînement des interactions autour des 6 diodes témoin placées près du microcontrôleur dsPic, ainsi que la diode verte « progr. Prêt » placée au-dessus de la fenêtre du pupitre.

Activité 2

- ❑ Effectuer deux différentes procédures de mise en service du drone didactique D2C, en positionnant le potentiomètre gauche du pupitre « commande moteurs » d'une part à 0 % et d'autre part à 50 % ; expliquer ce qu'a voulu signifier le programmeur du micro-contrôleur à l'utilisateur, en maintenant le clignotement de la diode verte « programme prêt » dans ce deuxième cas.

Activité 3

Préparer une synthèse qui devra contenir :

- ❑ une présentation du système, de son fonctionnement et de ses constituants ;
- ❑ une synthèse des activités précédentes.

2 APPROPRIATION DE LA PROBLÉMATIQUE

Objectif 3 : s'approprier la problématique – Durée : 15 min.

Rappel : l'objectif de ce TP est de concevoir la commande du bloqueur.

Activité 4

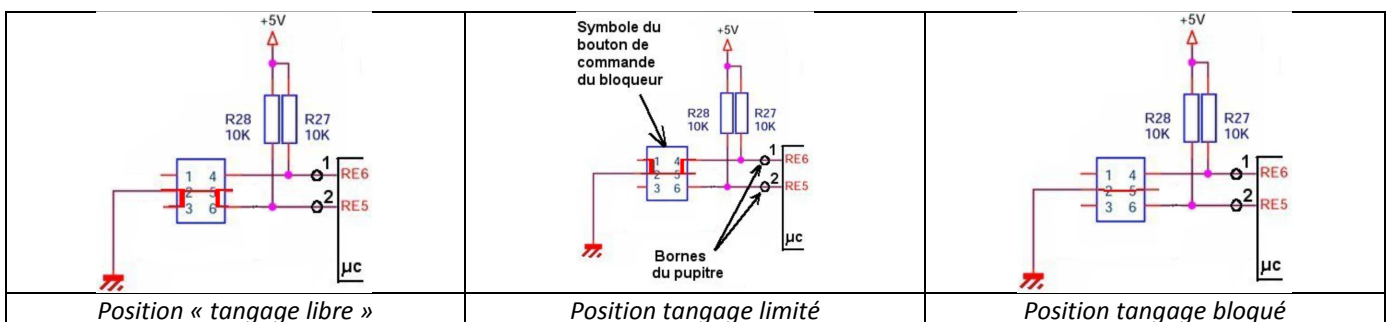
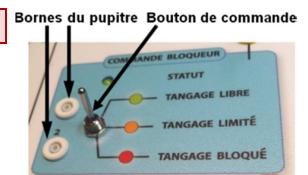
- ❑ Proposer une méthode permettant de répondre à la problématique.

3 PROGRAMMATION DE LA COMMANDE DU SERVOMOTEUR BLOQUEUR

Objectif 4 : programmer la commande du servomoteur bloqueur – Durée : 45 min.

3.1 Exploitation du fonctionnement du bouton

Le bouton de commande manuelle du servomoteur bloqueur est un bouton à trois positions. Il n'est prévu que deux entrées numériques (RE5, RE6) du microcontrôleur pour prendre en compte l'état du bouton. Les figures ci-dessous présentent le circuit créé par le bouton, dans chacune des trois positions. Les deux bornes 1 et 2 sont disponibles pour une connexion externe qui sera exploitée ultérieurement.



Activité 5

- ❑ Les résistances « R27 » et « R28 » sont des « résistances de tirage » qui positionnent les entrées au 5 V lorsqu'elles ne sont pas reliées à la masse. Compléter le tableau suivant :

Position	Borne 1 – RES6	Borne 2 – RES5
Tangage libre	5 V – état binaire 1	0 V – état binaire 0
Tangage limité		
Tangage bloqué		

- ❑ Le diagramme d'états « commande_bloqueur_1 » de la fiche « Diagramme d'états » visualise les états du « servo moteur bloqueur » en fonction de la position du bouton. Donner les 5 conditions (de transition ou « de garde ») manquantes.

3.2 Simulation de la commande

Le diagramme d'états du fichier « Simulation_servo_tangage_1_v2_sujet.slx », correspond au diagramme SysML « commande_bloqueur_1 ».

Activité 6

- ☐ Ouvrir le fichier Matlab et compléter ses conditions (transitions).
- ☐ Simuler le fonctionnement.
- ☐ Valider la programmation et faire évoluer le graphe, en double-cliquant sur les « switch ». Selon la position de ces switches, vérifier que le graphe simulé évolue en accord avec le servo bloqueur du système didactique D2C.

3.3 Exécution de la commande sur le drone didactique

Il faudra ici se référer au paragraphe 8.1.

Activité 7

- ☐ Sous Matlab, ouvrir le programme « Execution_servo_tangage_1_v2_sujet.slx ». Lorsque la connexion du servo bloqueur n'est pas réalisée, vérifier que les états du graphe Matlab évoluent correctement en fonction de la position du bouton de commande bloqueur.
- ☐ Lorsque la connexion du fil de commande du servo bloqueur à la broche 3 de l'Arduino est réalisée, observer les mouvements du servo-bloqueur et valider en conséquence la programmation réalisée.
- ☐ En cas d'écart entre le cahier des charges et le fonctionnement du système, revoir la programmation.

3.4 Procédure d'accostage progressif

Pour éviter l'accostage brutal du bloqueur avec le balancier, le programmeur a mis en place une procédure d'accostage progressif. Ce comportement est implémenté dans le fichier « Execution_servo_tangage_4_tempo_v2.slx ».

Activité 8

- ☐ L'Arduino-box étant toujours connectée au système D2C, débrancher la connexion de la broche 3, puis lancer l'exécution de ce programme. Rebrancher la connexion de la broche 3 et effectuer des commandes du bouton du servo bloqueur, de la position « tangage limité » à la position « tangage bloqué ».
- ☐ Observer le comportement du servo bloqueur, puis du graphe à l'intérieur de l'état « mvt_controle » lors de ces commandes.
- ☐ Tracer le graphe de la position angulaire du servo en fonction du temps, pour décrire le comportement du servo bloqueur lors de l'activation de cet état « mvt_controle ».
- ☐ En supposant que le servo bloqueur soit très rapide, donner le temps théorique de passage de la position 60° à la position « tangage bloqué ». Est-ce équivalent au déplacement en temps réel ?

Réaliser une synthèse de vos activités.

4 PROGRAMMATION DE LA PROCÉDURE DE MISE EN SERVICE – PROGRAMMATION PARTIELLE CONCERNANT LE CAPTEUR GYROMÈTRE

Objectif : L'objectif du travail est de programmer sous Matlab-Stateflow, la procédure concernant l'initialisation du « gyromètre » :

- ☐ attente d'une valeur stable (écart < 5 points) par test sur 10 couples de mesures ;
- ☐ réalisation d'une série de 50 mesures et obtention de la moyenne de ces mesures.

On s'intéresse au diagramme d'états « Initialisation » et au programme « Execution_gyro_2_sujet.slx ». On s'appuiera sur la fiche du paragraphe 8.2.

Activité 8

- ☐ Réaliser le câblage et mettre en service le D2C.
- ☐ Lancer l'exécution sous Matlab-Stateflow, de ce programme « Execution_gyro_2_sujet.slx » tel quel (sans modification, il se bloque sur l'état « calcule_ecart »).
- ☐ Relever les évolutions de l'afficheur « Display_entrée ».
- ☐ Expliquer ce qu'est le « bruit de quantification » de la conversion analogique numérique du signal du gyromètre (numérisé sur 10 bits), et justifier que pour ce travail, on considère que le balancier du drone didactique est stabilisé si l'écart entre deux mesures reste inférieur à 5 points.

Activité 9

- ☐ Compléter le diagramme d'état en utilisant la fiche 8.2.
- ☐ Valider expérimentalement la programmation permettant de réaliser la « stabilisé GYRO » en agitant ou non (manuellement) le balancier du drone didactique lors de l'exécution du programme.
- ☐ Sur la partie de programme « Calcul et MEMO valeur moyenne GYRO », (la mesure étant supposée stabilisée), le

calcul de moyenne sur 50 mesures permet d'éliminer le bruit de quantification dans la prise de référence du Gyromètre.

- Cahier des charges : il faut mettre en place l'itération et compléter le graphe pour le calcul de la moyenne sur 50 mesures et son affichage.
- Valider la programmation par l'intermédiaire du câblage avec le drone didactique D2C, et l'afficheur « Display_sortie ».

5 SYNTHÈSE

Objectif : exposer le travail effectué – 10 minutes

Activité 10

Proposer un poster présentant une synthèse de votre travail. Sur ce poster devront apparaître les éléments clé des différents temps forts abordés précédemment ainsi que la démarche scientifique mise en œuvre pour répondre à la problématique. Les outils de communication nécessaires à sa rédaction sont laissés à votre initiative.

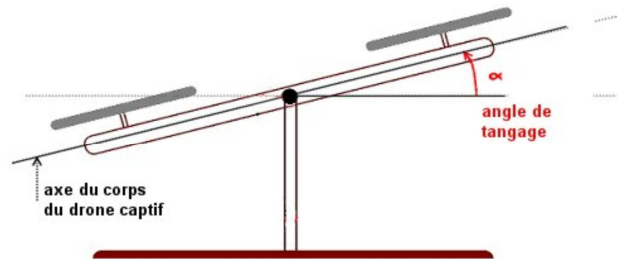
6 FICHE MISE EN ŒUVRE DU D2C

6.1 Description Générale

Obtenir le vol stable d'un quadri rotor n'est pas chose facile ; la mise en place de boucles d'asservissement autour de la centrale inertielle, et de traitements numériques dans le microcontrôleur permettent d'atteindre cet objectif.

L'étude de cette stabilisation est l'objet de ces travaux autour de l'asservissement de tangage.

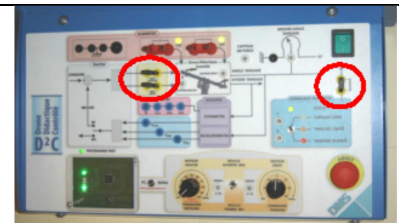
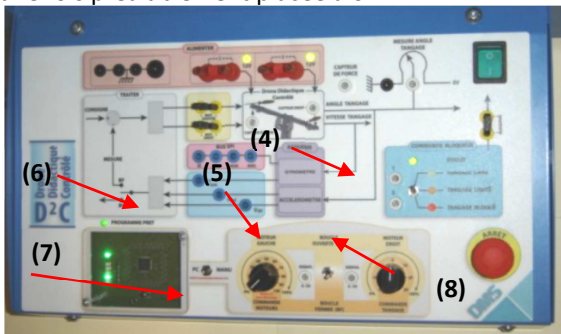
Pour pouvoir réaliser un vol en translation, le drone doit pouvoir être placé dans une position inclinée



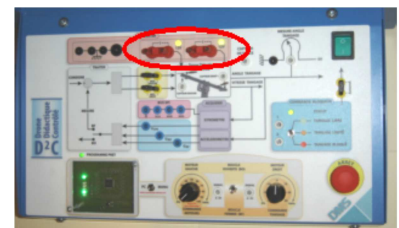
6.2 Liste des contrôles à réaliser avant démarrage

1. Connexion USB avec le PC : pas nécessaire.
2. 3 cavaliers noirs commande moteurs et bloqueur **en place**.
3. 2 cavaliers rouges alimentation moteurs en place.
4. Bouton « commande bloqueur » sur « tangage libre ».
5. Bouton BO/BF sur « Boucle fermée ».
6. Bouton « PC/MANU » sur MANU
7. Potentiomètre « COMMANDE MOTEURS » en position gauche (0).
8. Potentiomètre « COMMANDE TANGAGE » en position centrale.

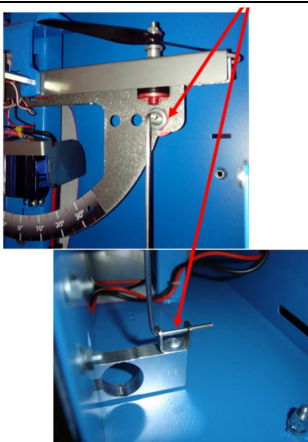
Nota : par sécurité, la « commande tangage » n'est active que si la « commande moteur » a été une fois préalablement placée à 0.



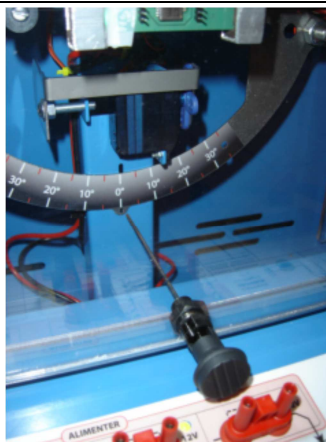
(2) 3 cavaliers noirs



(3) Cavaliers rouges



(2). Tige moteur droit à enlever



(3). Tige élastique tirée



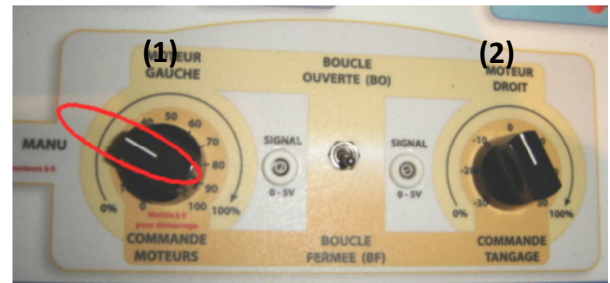
(4). Porte fermée

1. Si le système était en fonctionnement, l'éteindre pour retrouver les réglages par défaut.
2. Tige de mesure d'effort (sous le moteur droit) **enlevée**.
3. **Tige élastique** de blocage (au centre) placée **en position tirée** (et bloquée dans cette position).
4. **Porte d'accès** aux moteurs **fermée** (côté droit du châssis).
5. Système branché ; « Arrêt Général » déverrouillé ; interrupteur « 0/1 » sur 1.
6. Si les points suivants sont déjà en place, le système s'initialise de la façon suivante :
 - ☐ bips des moteurs : mélodie, puis 3 bips courts, puis un bip long ;
 - ☐ clignotement de la diode « programme prêt » ;
 - ☐ système bloqueur en déplacement rapide vers le blocage, puis finalisation du blocage en déplacement lent ;
 - ☐ extinction successive de trois diodes sur la carte électronique près du microcontrôleur ;
 - ☐ retour du bloqueur dans la position désignée par le bouton « commande bloqueur »



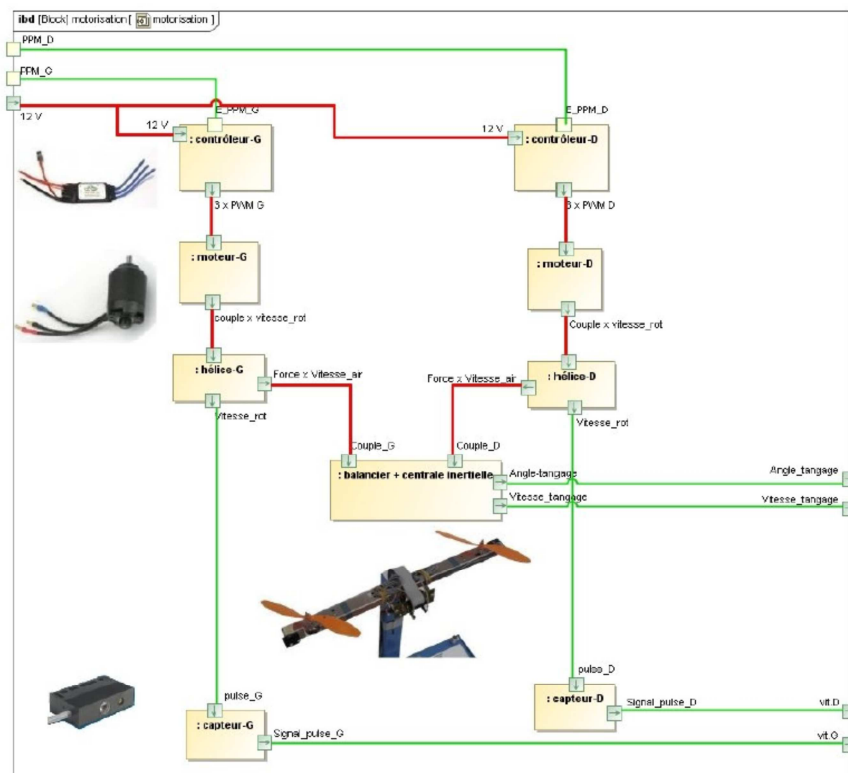
6.3 Pilotage de la position tangage

1. Agir progressivement sur le potentiomètre « COMMANDE MOTEURS » pour le placer à une position d'environ 25 % (le démarrage s'effectue à 18 %). Il s'agira de **choisir une vitesse suffisamment faible pour limiter les nuisances acoustiques**, ainsi que les vibrations dues à la motorisation.
2. Agir ensuite sur le potentiomètre « COMMANDE TANGAGE », pour positionner le balancier du drone didactique à l'inclinaison voulue.

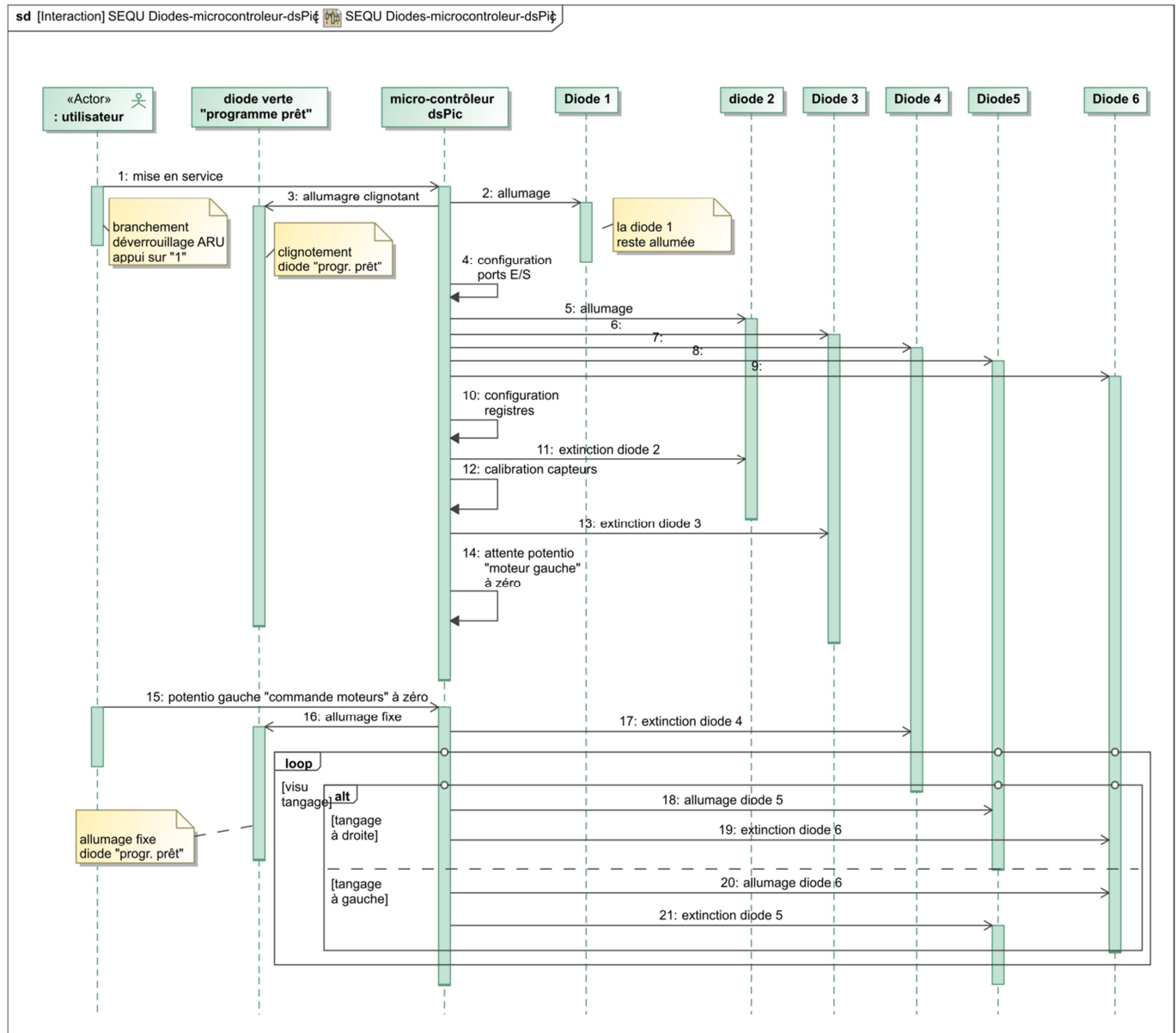


7 FICHE « INGÉNIERIE SYSTÈMES »

7.1 IBD

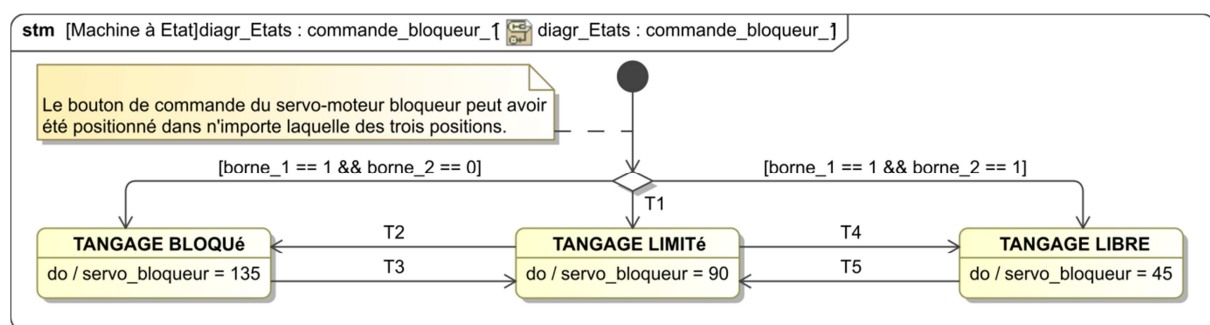


7.2 Diagramme de séquence de la procédure d'initialisation



7.3 Diagramme d'état – Commande bloqueur

Variables d'entrée	Variables de sortie	Angle du servomoteur
<input type="checkbox"/> borne_1, borne_2 : booléens.	<input type="checkbox"/> servo_bloqueur : entier de 45° à 135°.	<input type="checkbox"/> Tangage libre : 135°. <input type="checkbox"/> Tangage limité : 90°. <input type="checkbox"/> Tangage bloqué : 45°.



8 PILOTAGE DU SYSTÈME PAR MATLAB – STATEFLOW

Problème lié à la programmation de la carte Arduino par Matlab – Stateflow

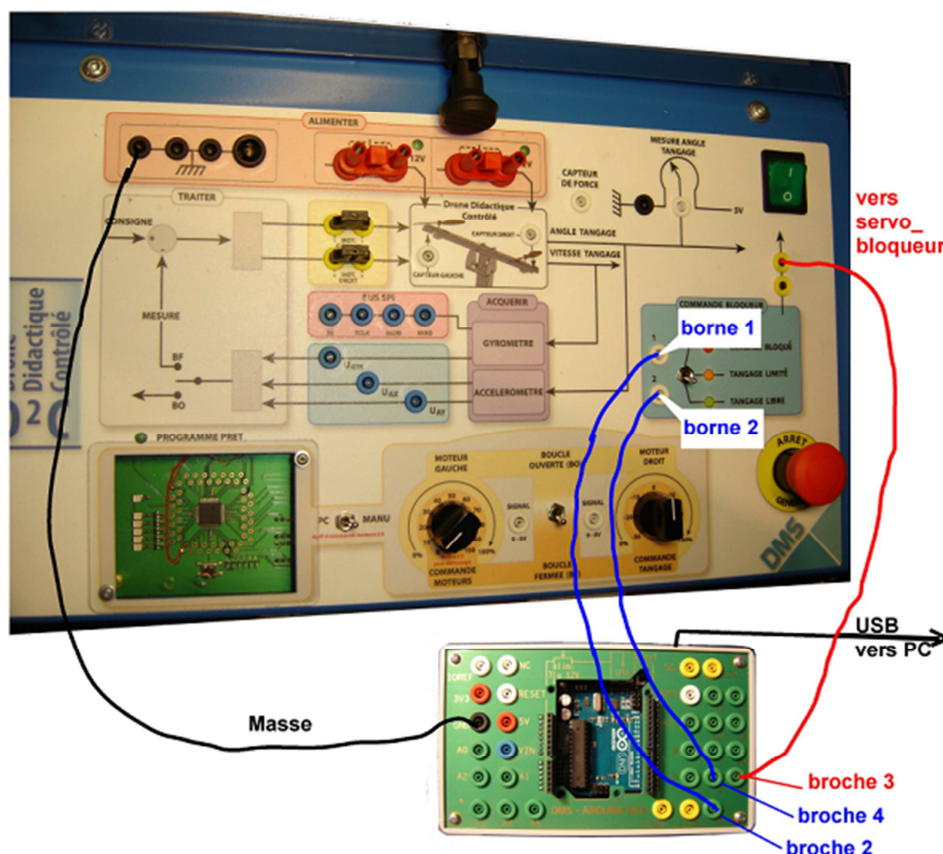
Lorsqu'un programme est préparé sous Stateflow et qu'il n'est pas en activité, les signaux générés par l'Arduino-box vers le drone didactique D2C ne sont pas maîtrisés en particulier lors des phases de lancement de l'exécution du programme ou des phases d'arrêt de l'exécution du programme.

En conséquence, dans la suite du travail sous Matlab, il faudra toujours gérer la connexion des fils de commande des actionneurs (servomoteurs ou moteurs) de la manière suivante :

- ❑ d'une part vérifier qu'ils ne sont pas connectés avant de lancer l'exécution du programme sous Matlab-Stateflow ; ne les connecter que lorsque le programme est pleinement opérationnel ;
- ❑ d'autre part les déconnecter avant d'arrêter l'exécution du programme sous Matlab-Stateflow.

8.1 Câblage de l'Arduino Box

- ❑ Retirer le cavalier de la commande du servo bloqueur.
- ❑ Connecter les fils reliant le système D2C à l'arduino-box (sauf la broche 3) selon la figure ci-dessous.
- ❑ Sous Matlab, lancer l'exécution du programme complété, puis seulement ensuite, connecter la broche 3.

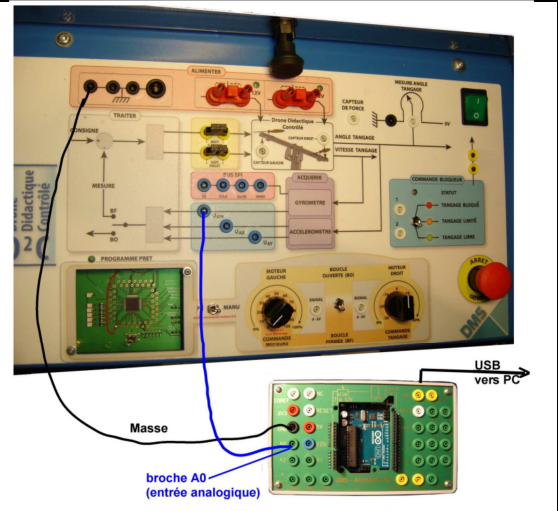


- ❑ Lorsque la connexion du servo bloqueur n'est pas réalisée : Vérifier que les états du graphe Matlab évoluent correctement en fonction de la position du bouton de commande bloqueur.
- ❑ Lorsque la connexion du fil de commande du servo bloqueur à la broche 3 de l'Arduino est réalisée, observer les mouvements du servo bloqueur et valider en conséquence la programmation réalisée.
- ❑ En cas d'écart entre le cahier des charges et le fonctionnement du système, revoir la programmation.

8.2 Capteur gyromètre

Le programme « Execution_gyro_2_sujet.slx » a été préparé pour mettre en place le détail des deux états : « vérification stabilité GYRO » et « Calcul et MEMO valeur moyenne GYRO ».

Réaliser le câblage pour la prise d'information de la « mesure GYRO » sur l'entrée analogique A0 de l'arduino-Box, ainsi que le câblage de la masse.



Cahier des charges de la programmation :

Sur la partie de programme « vérification stabilité GYRO » : il faut

- ❑ **Utiliser** le calcul de l'écart entre deux mesures successives et vérifier que cet écart est inférieur à 5 unités ;
- ❑ si ce n'est pas le cas, **remettre** le compteur n à zéro, reprendre le test pour **réitérer** à nouveau 10 fois le processus de test ; à chaque nouveau test, si l'écart est supérieur à 5 point, on reprend le processus au début avec n=0 .
- ❑ si c'est le cas **itérer** 10 fois le processus et **faire évoluer** les états actifs vers la procédure de calcul de valeur moyenne GYRO, située en dessous.

Sur la partie de programme « Calcul et MEMO valeur moyenne GYRO », (la mesure étant supposée stabilisée), le calcul de moyenne sur 50 mesures permet d'éliminer le bruit de quantification dans la prise de référence du Gyromètre.

Cahier des charges : il faut mettre en place l'itération et compléter le graphe pour le calcul de la moyenne sur 50 mesures et son affichage.

