

CHAPITRE

1

I PRESENTATION DU SIMULATEUR D'HELICOPTERE

I.1 INTRODUCTION

Les systèmes aéronautiques sont devenus aujourd'hui si difficiles et complexe qu'on ne puisse pas réaliser leurs commandes par des techniques classiques. En effet, des chercheurs automaticiens se sont penchés sur ces problèmes de commande, pour cela Plusieurs prototypes ont été réalisés dans le but de tester de nouvelles techniques de commande. Le simulateur d'hélicoptère (TRMS) est l'un de ces prototypes sur lequel nous allons travailler.

Dans ce chapitre, nous allons d'abord donner un petit aperçu sur le principe du vol des hélicoptères, ensuite nous présentons en détail le simulateur.

I.2 PRINCIPE DU VOL D'UN HELICOPTERE [SITE] [SAMB91][KAFI99]

I.2.1 LES FORCES AGISSANTES SUR L'HELICOPTERE EN VOL

Nous supposons, pour simplifier la présentation du vol de l'hélicoptère, que les pales du rotor principal décrivent un disque dont l'axe passe par le centre de gravité de l'hélicoptère.

En vol, l'hélicoptère est soumis à trois forces [figure I.1] :

- Le poids, appliqué au centre de gravité.
- La force générée par le rotor (portance).
- La traînée générale de l'appareil provoquée en vol de translation, par résistance de l'air sur la structure.



Figure I.1 : forces agissantes sur l'hélicoptère

Pour que l'hélicoptère soit en équilibre, il faut que la résultante du poids et de la traînée soit égale et opposée à la force générée par le rotor. La résultante des forces est alors nulle et le vol est stabilisée : si l'hélicoptère est immobile il le demeure, s'il est soumis à une translation, son mouvement est rectiligne. Le mouvement de l'hélicoptère résulte des variations en intensité et en direction de la portance.

I.2.2 Principe du vol de l'hélicoptère

Essayons de simplifier le principe de vol d'un hélicoptère en nous référant à l'avion.

Dans un avion, la traction est assurée par les hélices qui font prendre de la vitesse à l'avion.

Quant à la portance, ce sont les ailes qui, par leurs formes aérodynamiques, créent une force proportionnelle (au carré généralement) de la vitesse de traction provoquant, ainsi, l'envol de l'avion, dès que sa vitesse est suffisamment importante.

Pour un hélicoptère, c'est le rotor qui joue à la fois le rôle de l'aile et de l'hélice. Il assure, ainsi, traction et portance.

Ainsi, Pour changer de direction l'hélicoptère doit changer l'angle d'attaque de son rotor pour créer une portance dans la direction voulue.

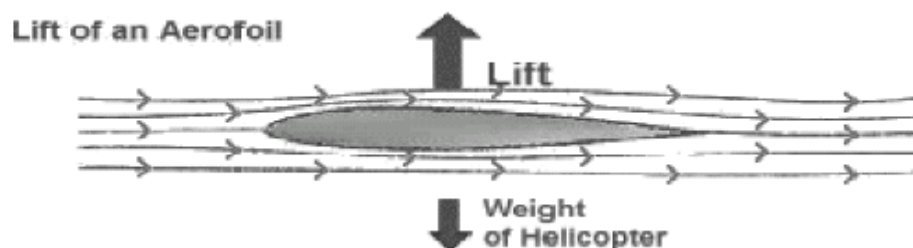


Figure I.2 : principe du vol de l'hélicoptère

Le rotor est entraîné en rotation par un groupe motopropulseur, constitué d'un moteur (à pistons ou turbine à gaz), d'un embrayage et d'un réducteur aussi appelé BTP (boîtier de transmission principal).

Cependant, comme pour l'hélice, le rotor exerce sur l'air un couple dû à son mouvement. L'air en retour (principe d'action-réaction), exerce un autre couple d'intensité égale mais de sens opposé. Imaginez que vous essayez d'introduire une vis encastrée dans un mur extrêmement dur avec une visseuse puissante, vous tourneriez avec la visseuse ! C'est le même phénomène qui est observé avec le rotor principal.

Pour contrer cet effet de couple et permettre un vol stable, on dispose à l'arrière des hélicoptères un "rotor de queue". Son rôle consiste à contrer le couple du rotor principal.

C'est le système le plus couramment utilisé actuellement mais il existe d'autres systèmes qui jouent le même rôle tel que : jet d'air, fenestron, deux rotors principaux (ce qui annule l'anti-couple), mais aussi deux rotors d'anti-couple.

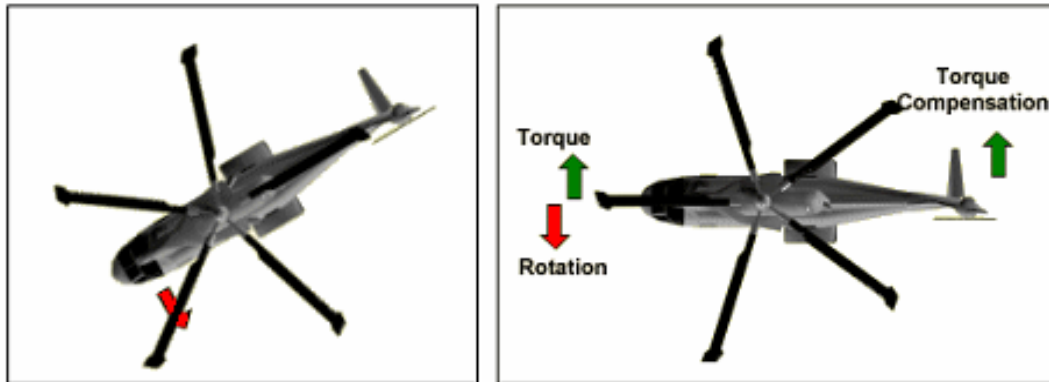


Figure I.3 : Effet de couple et rôle du rotor de queue.

Le rotor anti-coupl , son interaction sonore avec le rotor principal le rend bruyant, dangereux (extrêmement coupant), sensible au vent de travers consomme environ 10 % de la puissances, il est également coûteux et fragile, toutes ces raisons motivent les constructeurs à l'éviter ou à l'optimiser.

I.3 DESCRIPTION DU SIMULATEUR [MAN1]

Le TRMS (Twin Rotor Mimo System) est un système physique aérodynamique conçu pour le développement et l'implémentation de nouvelles lois de commandes. Ce-ci inclut, la modélisation de la dynamique du système, l'identification, l'analyse et la conception de divers contrôleurs par des méthodes classiques et modernes. Son comportement ressemble à celui d'un hélicoptère, de point de vue de commande ; il présente un système non linéaire d'ordre supérieur avec des couplages significatifs.

Le TRMS est formé de [figure I.4]:

- une poutre qui peut pivoter sur sa base de telle manière qu'elle puisse tourner librement dans le plan vertical et horizontal.
- Deux propulseurs (principal et secondaire) fixées aux deux extrémités de la poutre, ils sont formés d'une hélice, un moteur à courant continu ainsi qu'un bouclier pour des raisons de sécurité
- Un contrepoids fixé sur la tige à son pivot, son rôle est de diminuer les vibrations (oscillations) de la poutre
- Une tour pour maintenir la poutre

- Une base comprenant des circuits électroniques pour l'adaptation, synchronisation et filtrage des signaux entrants et sortants
- Un boîtier de marche/arrêt des moteurs

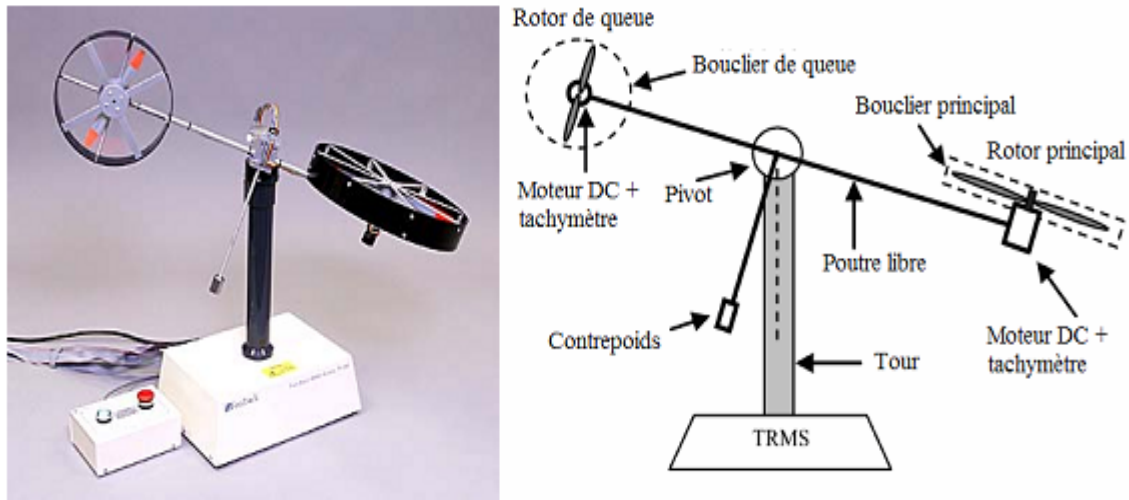


Figure I.4 : le simulateur d'hélicoptère TRMS (Twin Rotor MIMO System)

Le modèle du TRMS est multi-variable à deux entrées et deux sorties (MIMO 2*2). Les sorties étant l'angle d'élévation et l'angle d'azimut, les entrées étant les tensions appliquées aux deux moteurs. L'articulation sphérique permet à la poutre de pivoter simultanément dans le plan horizontal et vertical, c'est un système à deux degrés de liberté, mais on peut restreindre le mouvement à un degré de liberté par de 2 vis. [figure I.5]

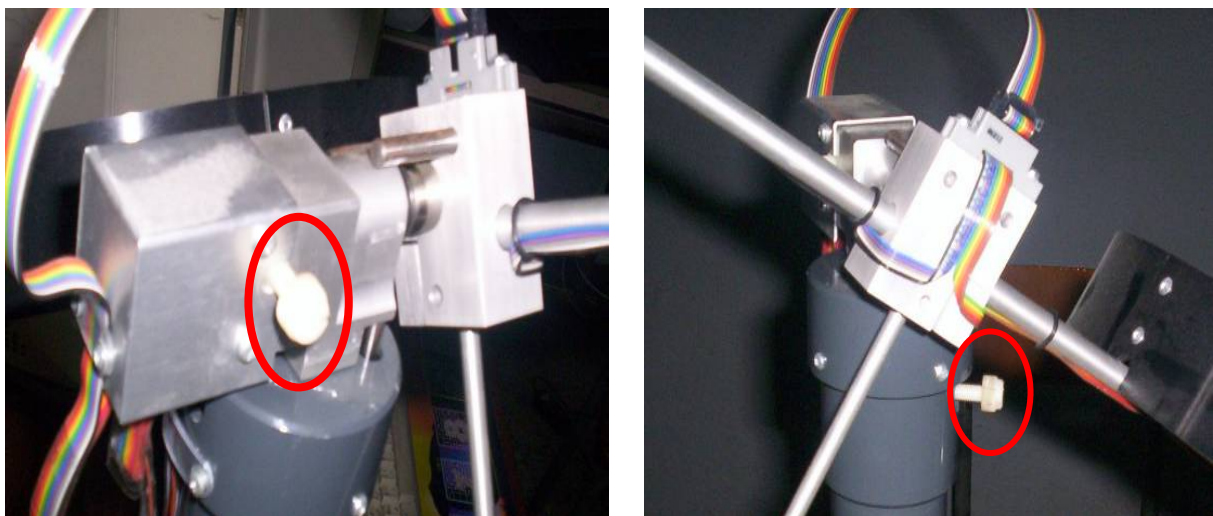


Figure I.5 : les deux vis de fixation

L'objectif de commande est de faire pivoter la poutre autour des deux axes rapidement et avec précision, et cela en variant les forces aérodynamiques des deux propulseurs. Dans un hélicoptère normal la force aérodynamique est commandée en changeant l'angle d'attaque. Néanmoins dans le TRMS cet angle est fixé, et les forces aérodynamiques sont commandées en changeant la vitesse des propulseurs. On utilise donc les tensions appliquées aux moteurs comme variable de commande.

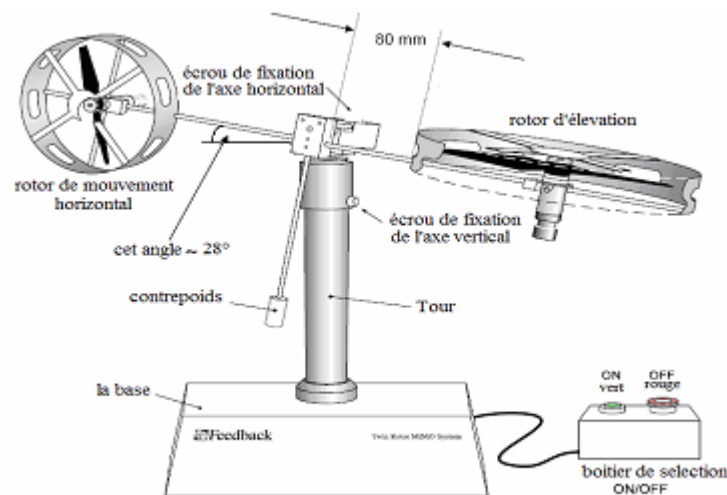


Figure I.6 : Description du TRMS.

I.3.1 Propulseurs

Le TRMS possède deux propulseurs, un principal qui agit dans le plan vertical, et l'autre secondaire qui agit dans le plan horizontal. Chaque propulseur est composé d'une hélice. Ces dernières sont entraînées par des moteurs à courant continu (marque MAXON), couplés avec des tachymètres. La tension de commande varie entre $\pm 2.5\text{ V}$, le signe de la tension indique le sens de rotation.

I.3.2 Commande des moteurs

Le principe des moteurs électriques à courant continu utilisés est que leurs vitesses de rotation est proportionnelle à la tension qui leur est appliquée. Le mécanisme utilisé est un hacheur, qui est un convertisseur statique alimenté par une source de tension continue qui produit aux bornes du moteur une tension de valeur moyenne réglable. Ce hacheur est commandé par des trains d'impulsions (MLI) qui sont générés à partir du signal de commande provenant du PC à travers la carte d'acquisition.

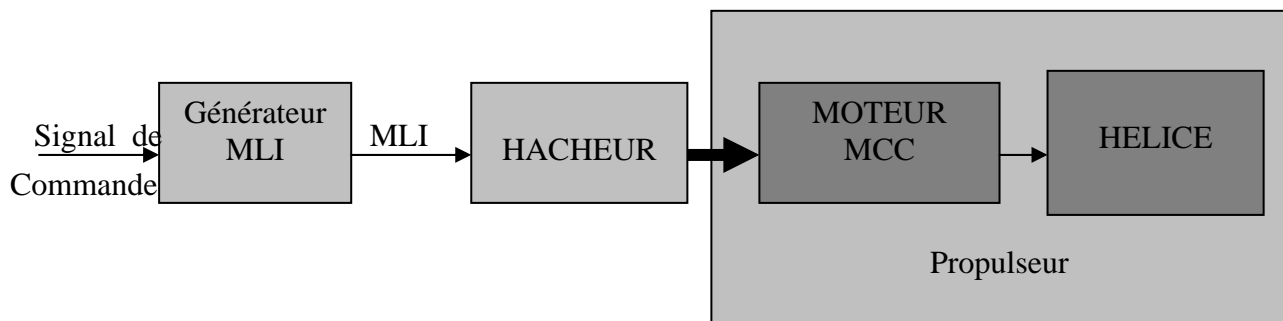


Figure I.7 : schéma synoptique du fonctionnement du propulseur

I.3.3 Les encodeurs optiques

Le TRMS a deux capteurs pour mesurer l'orientation de la poutre dans l'espace, ils se situent à l'intérieur du pivot sphérique de la poutre. Il s'agit de deux encodeurs optiques incrémentaux : le principe de fonctionnement est le suivant [figure I.8]:

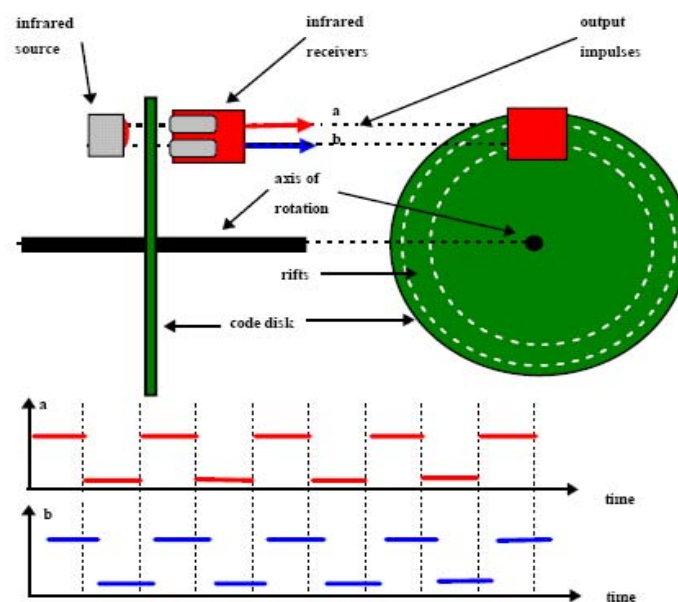


Figure I.8 : encodeurs optiques

Les faisceaux lumineux émis par les deux photo diodes (A et B) passent par deux anneaux de fentes sur le disque. Les fentes ont une différence de phase, de sorte que la

tension des récepteurs (A et B) soient des ondes rectangulaires avec une différence de phase. Le signe de la différence de phase permet de déterminer la direction de la rotation

I.3.4 La base du TRMS

À l'arrière de la base on trouve les différentes connectiques [figure I.9]:

- **Connecteur ON / Off** : fiche DIN pour connecter la boîte de l'intercepteur ON/OFF des moteurs
- **Sélecteur 110/220v**
- **Fiche d'alimentation 110/220 V**
- **Connecteur CN1** : il se branche avec le PL1 de la boîte d'adaptation, avec une large nappe 40 pistes. Les signaux sont ceux des deux capteurs de positions, c'est des signaux digitaux sur 16 bits,
- **Connecteur CN2** : il se branche avec PL3, avec une nappe 20 pistes. Les signaux sont les tensions de commandes des deux moteurs
- **Connecteur CN3** ; il se branche avec PL2, avec une nappe 20 pistes. Les signaux sont ceux des tachymètres

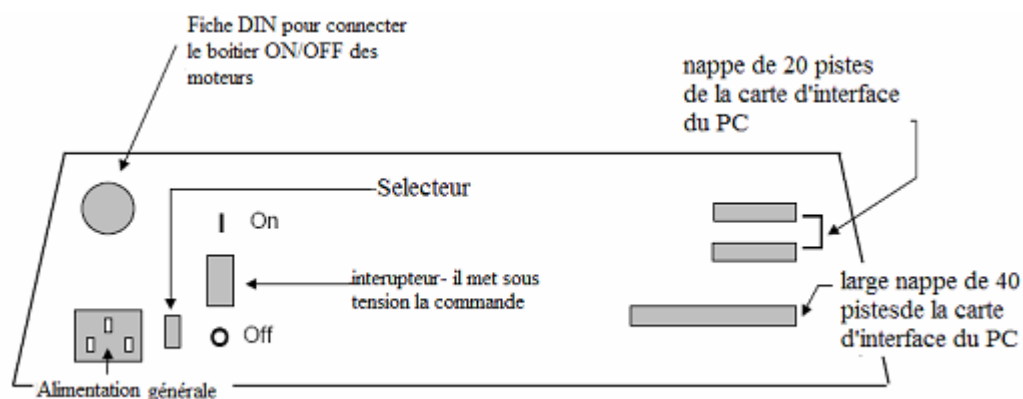
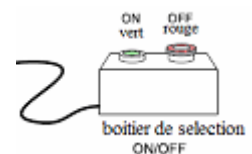


Figure I.9 : la base du TRMS

I.3.5 Boîtier marche/arrêt

C'est un petit boîtier avec deux boutons poussoirs pour, l'un vert pour fermer le circuit d'amplification des moteur, et l'autre rouge pour l'ouvrir et couper l'alimentation, son rôle principal est de protéger les moteurs en cas de mauvaise manœuvre par l'utilisateur. Il est connecté à l'arrière de la base du TRMS



I.3.6 Le (SCSI BOX)

La (SCSI BOX) figure I.10.b a comme rôle d'adapter les signaux entre la carte d'acquisition et les différents connecteurs de la base du TRMS, et aussi séparer ces différents signaux en fonction de leurs types. La boîte dispose de quatre connecteurs :

- un connecteur 68 pins pour connecter le câble global 68 pin SCSI de la carte d'acquisition
- un grand connecteur PL1 pour les entrées digitales des capteurs de positions (deux encodeurs optiques)
- un petit connecteur PL2 pour les sorties analogiques (deux tensions de commandes des moteurs)
- un autre petit connecteur PL3 pour les entrées analogiques (retour tachymétrique des deux moteurs)

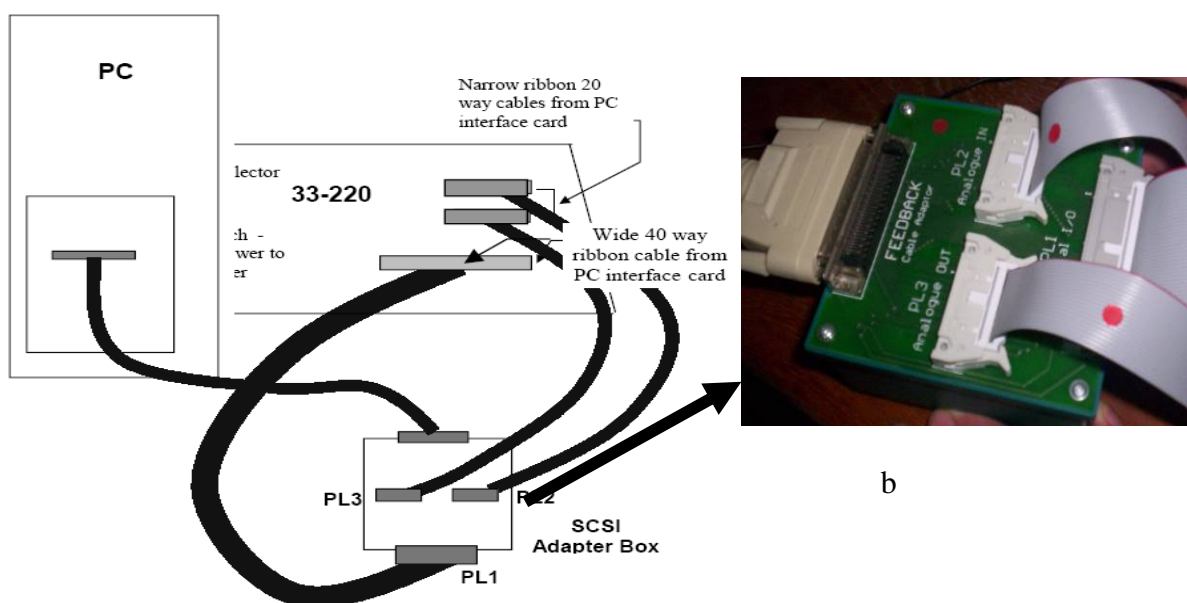


Figure I.10 : branchement des différents connecteurs

I.3.7 Carte d'acquisition : ADVANTECH PCI 1711

C'est une carte d'acquisition universelle qui s'installe sur le port PCI du PC de commande et dispose de connecteurs extérieurs pour des entrées/sorties analogiques et digitaux. Ces principales caractéristiques sont :

- La fonction Plug & Play
- 16 entrées analogiques configurables 16 simples (single-ended) ou source flottante
- Convertisseurs A/D industriels normalisés à approximations successives
- 12 bits utilisés pour la conversion des entrées analogiques
- La fréquence maximale d'échantillonnage est de 100KHz.
- Gamme des entées analogiques est programmables et contrôlable par software *
- Chaque canal à sa gamme individuelle stockée dans la RAM de la carte
- 2 sorties analogiques (convertisseur D/A)
- 16 canaux d'entées digitales
- 16 canaux de sorties digitales
- Un compteur/timer programmable
- Scanne automatique des gains/canaux.

Son principal rôle dans notre application, c'est qu'elle permet la commande digitale d'un système continu à travers un PC, en convertissant les signaux analogiques en numériques et vice-versa . Dans notre cas, on utilise :

- Deux sorties analogiques (convertisseur D/A) pour délivrer les tensions de commandes de références
- Les entrées digitales des deux encodeurs optiques pour mesurer l'angle d'élévation et d'azimut de la poutre
- Deux entrées analogiques (convertisseur A/D) pour récupérer le retour tachymétrique des deux moteurs

I.4 ENVIRONNEMENT DE DEVELOPPEMENT [RTW][RTWINTGT][MAN1]

Ce banc d'essai utilise une approche novatrice d'implémentation, qui jouit d'une flexibilité accrue et d'une architecture totalement ouverte, cette approche est appelée HIL pour « **Hardware-In-the-Loop** ». Ou bien « **Rapid Prototyping** »

Cette approche élimine l'étape entre la synthèse du régulateur en simulation, et son implémentation réelle sur un système embarqué spécialisé. Cette dernière cause des pertes de temps considérables, en testant et en calibrant à nouveau le régulateur implémenté, voici ci-dessous, un petit comparatif entre cette approche et l'approche classique :

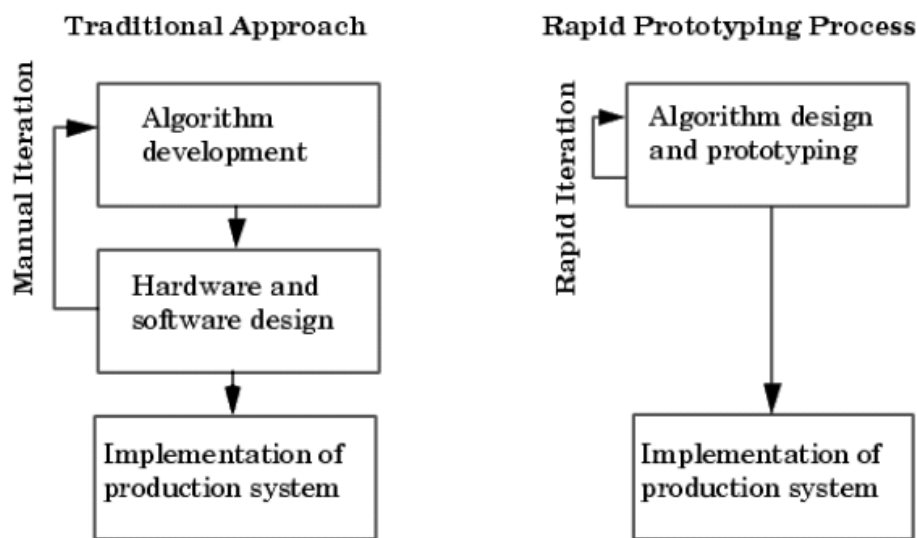


Figure I.11 : rapid prototyping VS processus de développement traditionnel

Avec le « hardware-in-the-loop », on teste directement le contrôleur qu'on a synthétisé en simulation sur le système physique réel, avec un simple PC de développement. En effet tous ce qu'on a besoin est :

- un PC standard (pour le développement et pour la commande)
- une carte d'acquisition pour la conversion A/D et D/A
- le système physique à commander
- les logiciels nécessaires

Les logiciels utilisés sont :

- **MATLAB** : il joue le rôle d'une plateforme, où tous les autres composants s'exécutent, c'est l'environnement le plus utilisé dans le monde par les ingénieurs, il fournit plusieurs routines sophistiquées de calculs numériques .
- **SIMULINK** : C'est un langage de haut niveau graphique, avec lequel on programme avec des objets ou blocks, ces objets peuvent être soit des block standards de bien des blocks spécialisées écrites par l'utilisateur comme des S-fonctions
- **REAL TIME WORKSHOP** : c'est le programme le plus important, il génère automatiquement du code source C++ optimisé, à partir du modèle simulink. De plus, ce code est paramétrable pour plusieurs targets (cible) d'implémentation. Dans notre cas le target est un processeur Pentium sous WINDOWS comme système d'exploitation
- **Compilateur C++** : il compile le code généré par real time workshop et fait l'édition des liens pour générer un exécutable qui communique avec la cible d'implémentation (target), dans notre cas, le compilateur est VISUAL C++ 6.0 PRO
- **REAL TIME WINDOWS TARGET** : Dans le cas d'un environnement WINDOWS, ce petit kernel (noyau), est essentiel, son rôle est primordial, il assure le temps réel pour notre régulateur. Car WINDOWS est un OS événementiel donc il n'est pas temps réel. Il s'exécute comme un service résident dans le niveau zéro (mode noyau) et intercepte les interruptions matérielles avant WINDOWS, ensuite, Il communique avec l'exécutable du régulateur et interface avec le système physique à travers la carte d'acquisition. Il contrôle donc le flux de données et de signaux du modèle (maintenant exécutable) vers le système physique et vice versa
- On a besoin aussi, des bibliothèques, ou drivers de la carte d'acquisition, pour les intégrer dans notre programme

Voici ci-dessus un schéma [figure I.12] qui illustre la relation qui existe entre les différents modules :

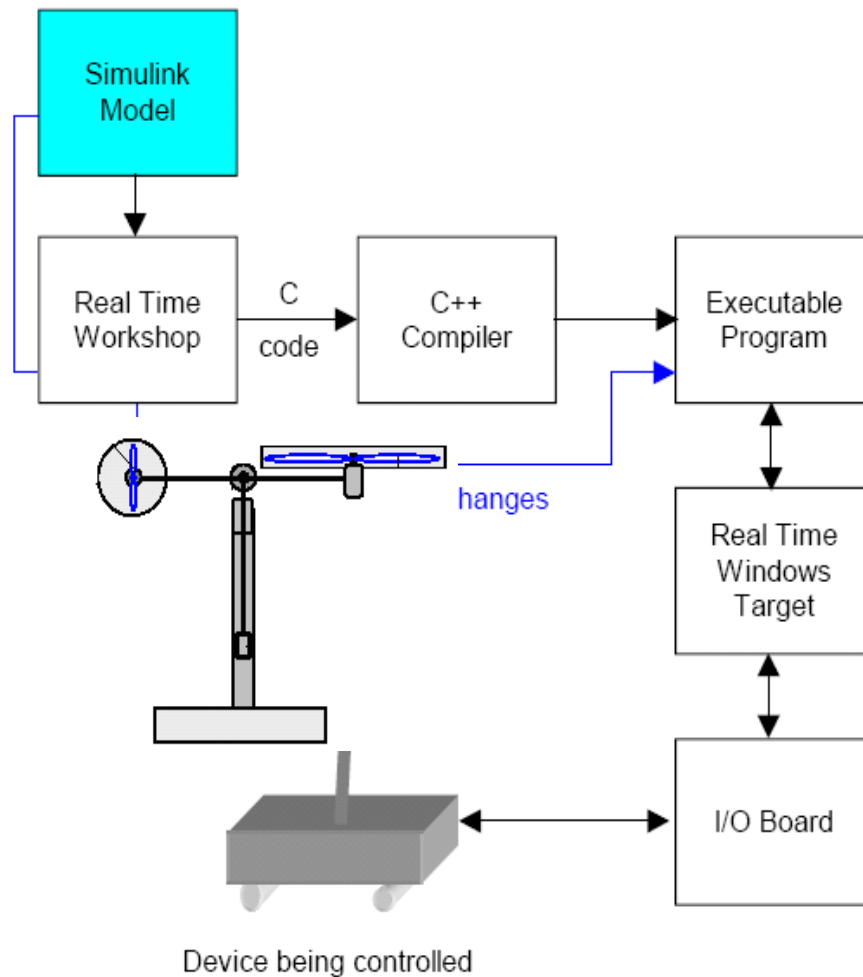


Figure I.12 : processus de génération du code exécutable

Et voici un autre schéma [figure I.13] qui illustre les différentes cibles (targets) et les différentes manières dont on peut implémenter notre régulateur

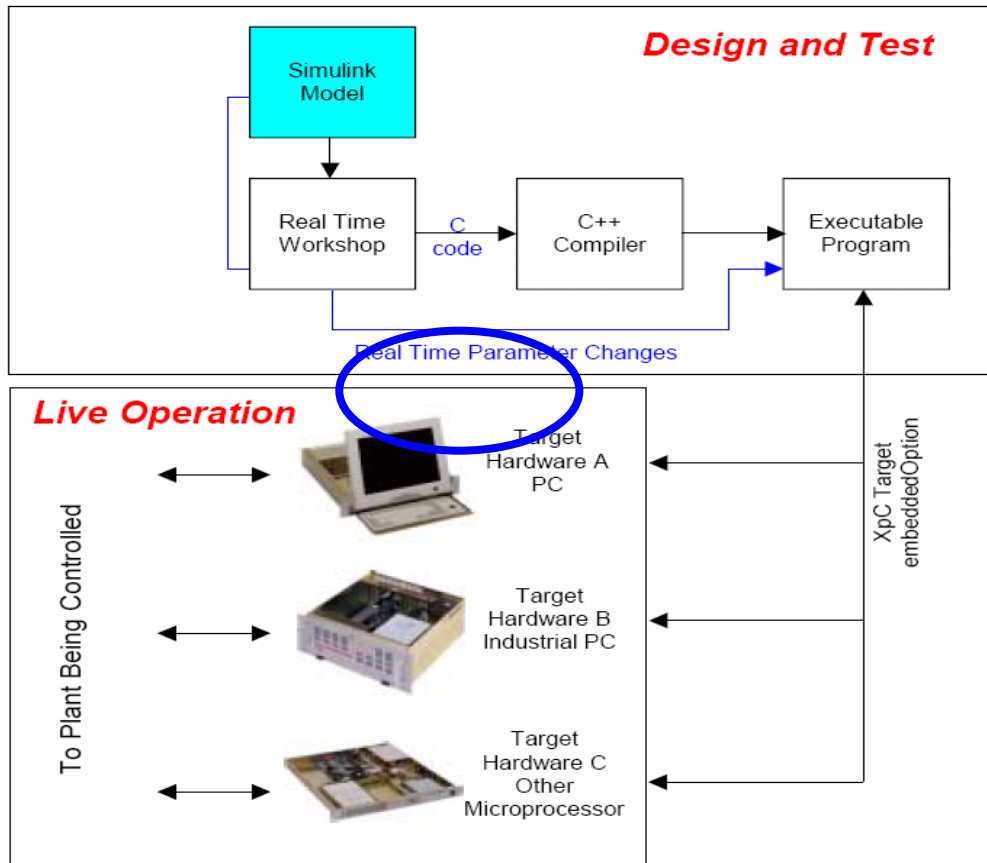


Figure I.13 : différentes cibles d'implémentation

Pour passer d'un régulateur simulé (non temps réel) vers un programme temps réel, il suffit juste d'ajouter dans le modèle de simulation sous SIMULINK, les différents blocs des entrées sorties de la carte d'acquisition (ANALOG INPUT, ANALOG OUTPUT, ENCODER CHANNEL ...etc)

Pour notre carte d'acquisition voici la librairie utilisée :

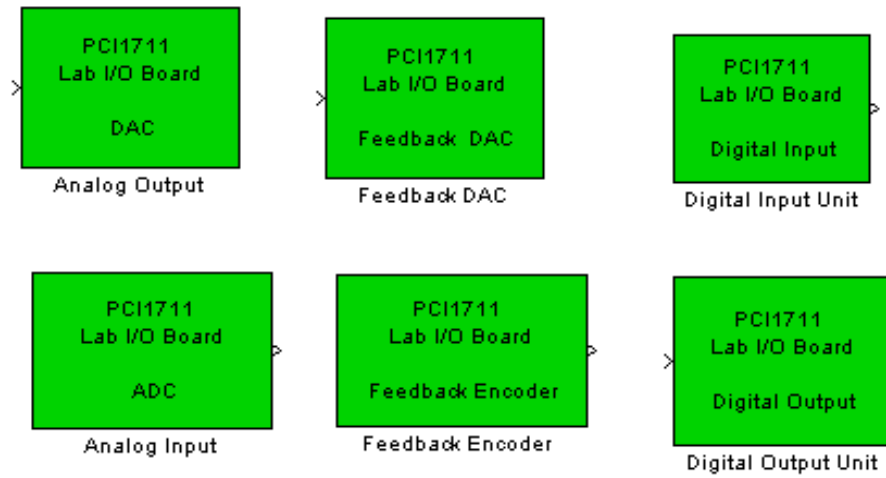


Figure I.14 : librairie de la carte d'acquisition

Et voici ci-dessous un exemple complet de commande en temps réel avec MATLAB :

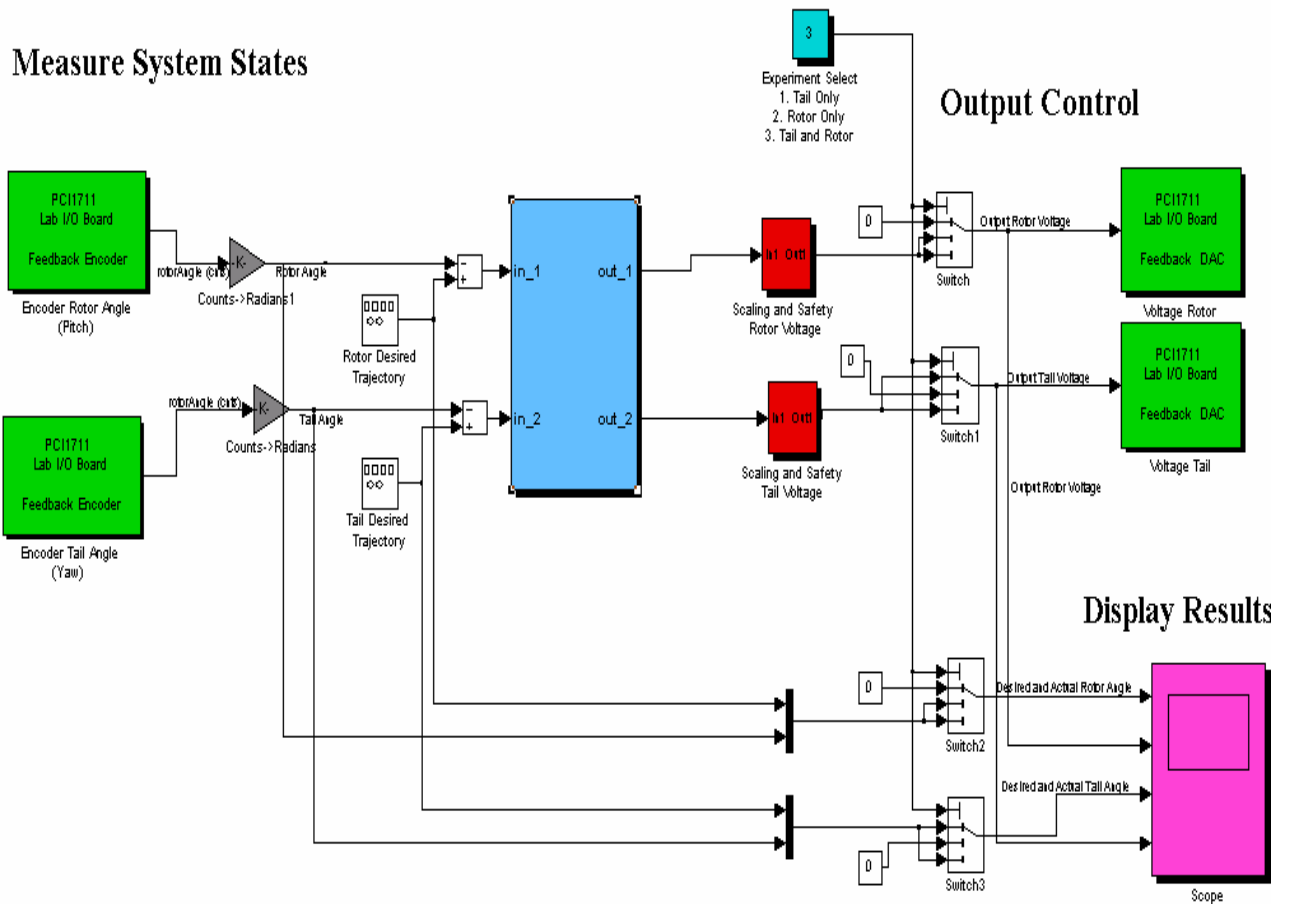


Figure I.15 : exemple d'implémentation d'un contrôleur

En vert : blocs de communications avec la carte d'acquisition, à gauche les encodeurs et à droite des sorties analogiques (tension de commande)

En rouge : blocs de protection des moteurs, car la tension max est de $\pm 2.5\text{ V}$

En gris : blocs de conversion, car les encodeurs incrémentaux donnent le nombre d'impulsions qu'il faut convertir en radian

En bleu : est la partie essentielle qui est le bloc de notre régulateur. Ce bloc peut être écrit par n'importe quel langage, on peut même importer un régulateur déjà écrit et l'implémenter, grâce à l'architecture ouverte de cette approche.

Après compilation du programme, il sera indépendant de MATLAB et SIMULINK, néanmoins, on peut utiliser juste SIMULINK comme interface homme machine, pour visualiser et récupérer les signaux.

I.5 CONCLUSION

Dans ce chapitre, nous avons d'abord donné le principe du vol d'un hélicoptère réel. Ensuite nous avons présenté une brève description du simulateur (TRMS). Ainsi que son principe de fonctionnement. Enfin, nous avons expliqué l'environnement de développement utilisé.