

## CHAPITRE 3

### IMPLÉMENTATION DES MODÈLES À VALEUR MOYENNE

#### 3.1 Introduction

Ce chapitre présente la démarche utilisée pour réaliser la simulation en temps réel des modèles à valeur moyenne avec commande externe. Tous les outils matériels et logiciels utilisés ainsi que les étapes nécessaires à la réalisation de cette partie du projet sont présentés.

La section 3.2 présente les outils et les étapes nécessaires à l'implémentation de la commande. Cette section présente la carte de prototypage DS1104 de dSPACE. Les étapes de génération, de téléchargement et de compilation du code C d'une application temps réel sont décrites en détail. La démarche complète de configuration d'un modèle Simulink pour l'implémentation du code C dans dSPACE est exposée. De plus, la méthode pour démarrer une application temps réel à l'aide du logiciel ControlDesk est expliquée.

La section 3.3 présente les outils et les étapes nécessaires à l'implémentation de l'électronique de puissance et de la machine, qui doivent être simulé en temps réel à l'aide d'un simulateur. Le simulateur en temps réel est réalisé grâce à l'outil xPC Target. La description de l'environnement et du fonctionnement de cet outil est faite en détail. L'environnement est composé de deux ordinateurs, soit l'hôte et la cible, ainsi que de deux cartes d'E/S PCI-MIO-16E-4 de National Instruments. La description présente entre autre la façon dont un modèle Simulink est généré en code C et compilé pour constituer une application cible qui est exécutée en temps réel. Les détails de configuration du modèle Simulink pour utiliser les cartes d'E/S et générer une application cible sont décrits.

Les résultats obtenus pour la simulation en temps réel avec commande externe des modèles AC3 et AC6 sont présentés dans les sections 4.4 et 4.5. Ces résultats sont validés par la comparaison avec les résultats de simulation en temps différé présentés dans les sections 4.2 et 4.3.

### **3.2 Implémentation de la commande**

L'implémentation de la commande est réalisée à l'aide de la carte de prototypage DS1104 de la compagnie dSPACE. L'installation et la configuration de la carte DS1104 ont été réalisées à l'aide du document [7]. Les documents [8], [9] et [10] ont été nécessaires pour l'installation et l'utilisation des logiciels de dSPACE.

Cette carte comprend un processeur principal PowerPC de 250 MHz et un DSP esclave. Le DSP est dédié à la génération d'impulsions par modulation de largeur d'impulsion (MLI) et n'est pas utilisé dans le cadre du projet. La carte est munie de quatre entrées analogiques avec des convertisseurs analogiques à numériques (CAN) de 12 bits possédants des temps de conversion de 800 ns, de quatre entrées analogiques multiplexées avec un seul CAN de 16 bits possédant un temps de conversion de 2  $\mu$ s, de huit sorties analogiques avec des convertisseurs numériques à analogiques (CNA) de 16 bits possédant des temps de conversion de 10  $\mu$ s et de vingt E/S numériques. Les entrées et les sorties analogiques ont une plage de tension de -10 V à 10 V. L'accès aux signaux des E/S est réalisé avec le panneau CP1104. L'accès aux signaux analogiques se fait via des prises BNC et l'accès aux signaux numériques se fait via des prises Sub-D. L'annexe 1 présente les spécifications techniques de la carte DS1104 et du panneau CP1104.

La carte DS1104 est installée dans une fente PCI de l'ordinateur hôte. Elle est liée à MATLAB, Simulink et Real-Time Workshop par un outil nommé RTI (Real-Time Interface). Cet outil permet, lors du développement d'un modèle dans Simulink, l'accès aux E/S de la carte et l'implémentation du modèle Simulink dans la carte par la génération

automatique de code. Le schéma du montage matériel et logiciel complet utilisé dans la réalisation de la première partie du projet est présenté par la figure 15.

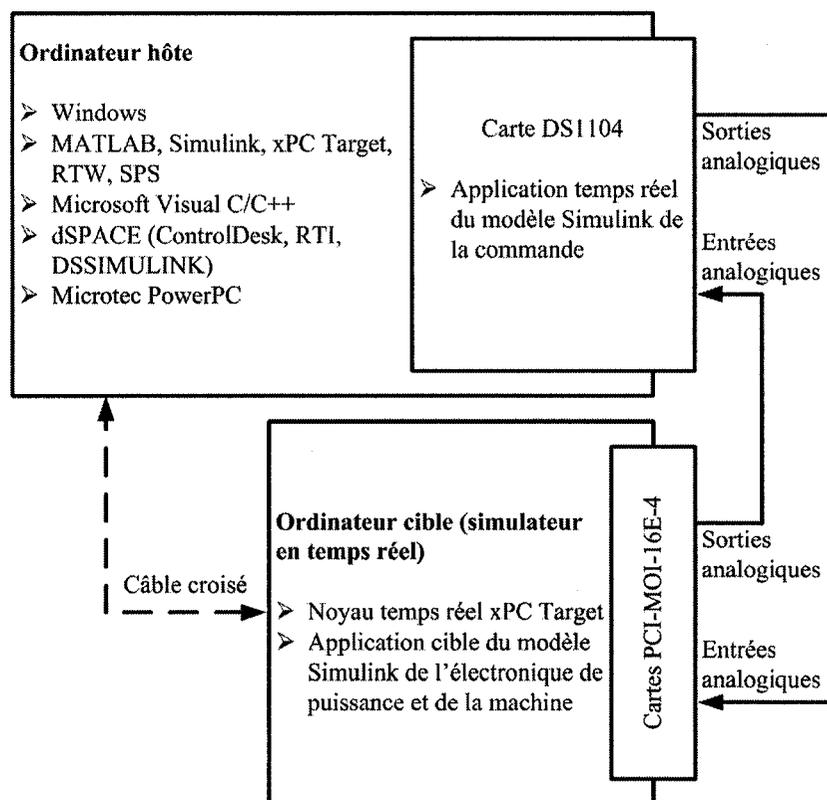


Figure 15 Schéma du montage

Lors de l'installation du logiciel RTI, la librairie dSPACE RTI1104 est incluse dans Simulink via une interface entre MATLAB et dSPACE nommée MLIB. La librairie dSPACE RTI1104 donne accès à toutes les fonctionnalités de la carte. Dans cette partie du projet on s'intéresse particulièrement aux E/S analogiques. Ces E/S sont représentées sous forme de blocs dans Simulink. On insère et connecte les blocs de la librairie au modèle Simulink comme n'importe quel autre bloc standard. Il suffit ensuite de configurer les différents blocs utilisés.

Pour l'implémentation d'un modèle Simulink dans la carte DS1104, RTI doit créer une application temps réel. Real-Time Workshop et RTI génèrent le code C d'une application temps réel à partir du modèle Simulink et des blocs d'E/S de dSPACE en utilisant le format de code temps réel de Real-Time Workshop. Le code est ensuite téléchargé dans la carte DS1104 et compilé par le compilateur Microtec PowerPC pour produire l'application temps réel exécutable. Les étapes du processus de génération d'une application temps réel sont présentées par la figure 16.

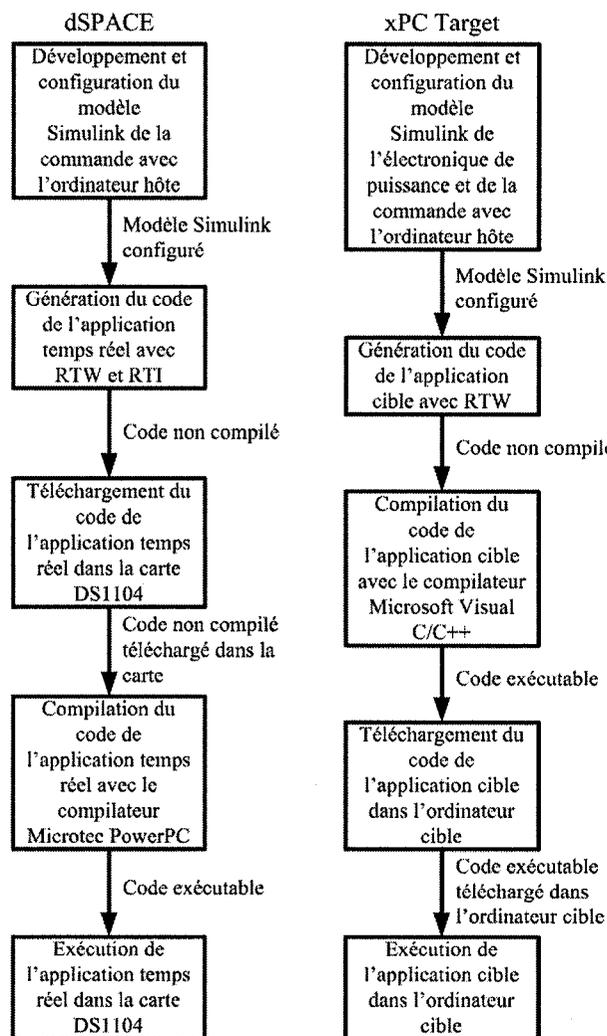


Figure 16 Processus de génération des applications cible et temps réel

Par défaut, l'application temps réel sur la carte démarre automatiquement à la suite de la compilation. Dans la configuration du modèle dans Simulink, on peut spécifier l'état initial de l'application suite à la compilation. Les trois états possibles sont RUN, PAUSE et STOP. Par la suite, on peut modifier l'état de l'application via une variable nommée *SimState*. Cette variable est créée lors de la génération de code et est seulement disponible via le logiciel ControlDesk. Ce logiciel fournit toutes les fonctionnalités de contrôle et d'affichage de la carte DS1104. ControlDesk agit indépendamment de MATLAB et Simulink après la génération du code. Les variables et données nécessaires à ControlDesk sont transmises de Simulink lors de la génération du code via une interface nommée DSSIMULINK. Dans le cadre du projet, ControlDesk est utilisé seulement pour contrôler la variable *SimState*. Lors de la configuration d'un modèle dans Simulink, on spécifie que l'état initial de l'application temps réel soit sur PAUSE. Par la suite, grâce au logiciel ControlDesk, on fait démarrer l'application temps réel au moment désiré via la variable *SimState*.

### 3.2.1 Développement et configuration des modèles

Dans le projet de recherche, les applications temps réel sont créées à partir des commandes des modèles AC3 et AC6 à valeur moyenne. Dans le cas d'AC3 la commande est constituée d'une commande de vitesse et d'une commande à flux orienté. Pour AC6 la commande est constituée d'une commande de vitesse et d'une commande vectorielle. Le chapitre 2 présente la description détaillée des commandes de chaque modèle. Les figures 22, 23, 25 et 26 présentent les schémas Simulink pour les commandes des modèles AC3 et AC6. Ces figures présentent les blocs d'E/S analogiques de la carte DS1104. Ces blocs sont disponibles dans la librairie dSPACE RTI1104 de Simulink. La figure 17 présente les volets sous lesquels sont disponibles les E/S analogiques de la carte DS1104.

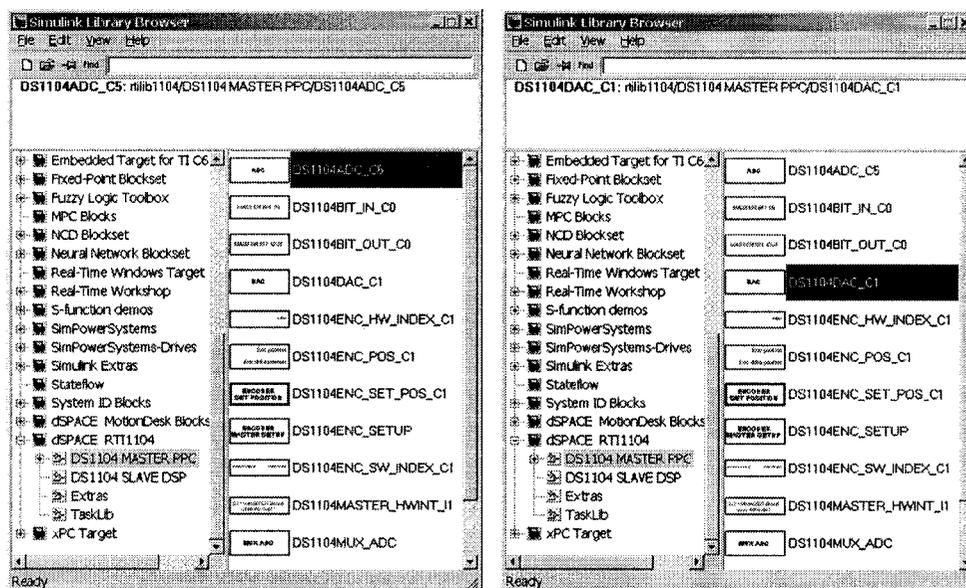


Figure 17 Blocs d'E/S analogiques de la carte DS1104

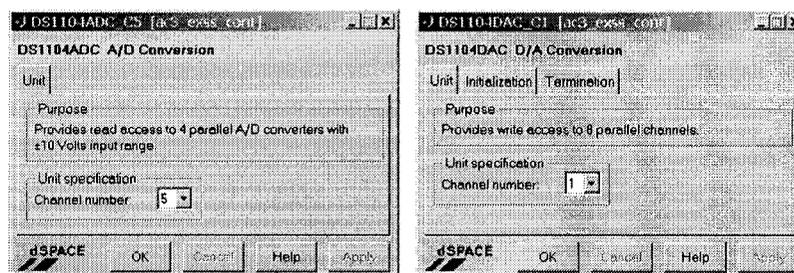


Figure 18 Fenêtres de configuration des blocs d'E/S analogiques

Les blocs d'entrées analogiques ne nécessitent pas de configuration particulière. La seule information requise est le numéro de l'entrée désirée. Les plages de tension des entrées sont de -10 V à 10 V et la période d'échantillonnage est celle du modèle, soit de 20  $\mu$ s dans tous les cas. Les blocs de sorties analogiques, quant à eux, nécessitent un peu plus d'information. Ces blocs nécessitent le numéro de la sortie désirée, l'état initial et l'état final. Ces états sont fixés à 0 V pour toutes les sorties dans le cadre du projet. La plage de

tension des sorties est de -10 V à 10 V et la période d'échantillonnage est de 20  $\mu$ s dans tous les cas. La figure 18 présente les fenêtres de configuration des blocs d'entrée et de sortie analogiques de dSPACE.

Le modèle de Simulink doit être configuré afin d'être généré en code C pour créer une application temps réel pour dSPACE. Pour faire cette configuration, il faut choisir *Configuration parameters* à partir du menu *Simulation* dans l'interface de Simulink. Ceci ouvre la fenêtre de configuration des paramètres qui est présentée à la figure 19. Dans l'onglet *Solver*, il faut inscrire 0 pour le temps de départ dans la case *Start time* et déterminer le temps d'arrêt dans la case *Stop time*. Ensuite, on choisit l'algorithme de calcul à pas fixe de type discret en sélectionnant *Fixed-step* dans la case *Type* et *Discrete* dans la case *Solver*. Par la suite, il faut inscrire la période d'échantillonnage de 20  $\mu$ s dans la case *Fixed-step size*. Finalement, dans la case *Mode*, il faut spécifier que le modèle fonctionne en tâche unique en sélectionnant *Single Tasking*.

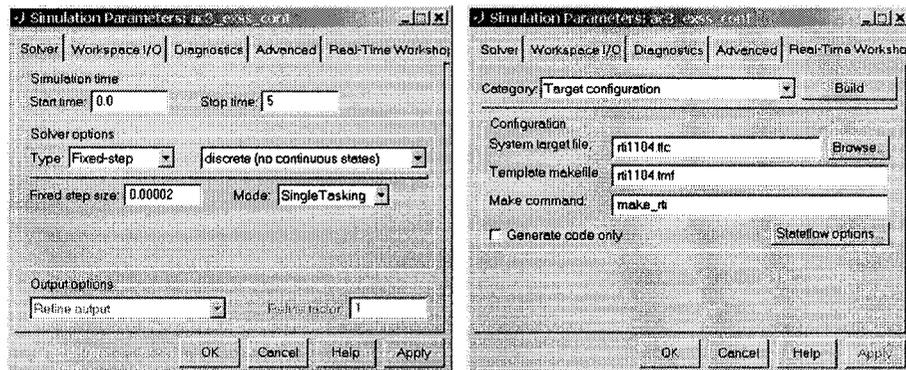


Figure 19 Fenêtres de configuration des paramètres de dSPACE

Dans l'onglet *Real-Time Workshop*, en appuyant sur *Browse*, on sélectionne *rti1104.tlc*, qui spécifie la génération du code C du modèle pour la carte DS1104. Par la suite, on sélectionne *RTI simulation options* dans la case *Category*. On spécifie que l'exécution est réalisée en temps réel en sélectionnant *real-time* dans la case *Execution mode*. Il faut aussi

spécifier que l'état initial de l'application temps réel soit pause en sélectionnant *pause* dans la case *Initial simulation state*.

Afin de lancer la génération, le téléchargement et la compilation du code, il suffit d'appuyer sur le bouton *build* de la fenêtre de configuration des paramètres sous l'onglet *Real-Time Workshop*, présenté à la figure 19. Ces étapes sont exécutées automatiquement avec Real-Time Interface (RTI), Real-Time Workshop (RTW) et le compilateur C Microtec PowerPC.

Lorsque les étapes de génération, de téléchargement et de compilation du code sont terminées, l'application temps réel sur la carte DS1104 est prête à être démarrée. Cette application est démarrée via la variable *SimState* qui est disponible dans ControlDesk. La figure 20 présente l'interface de ControlDesk.

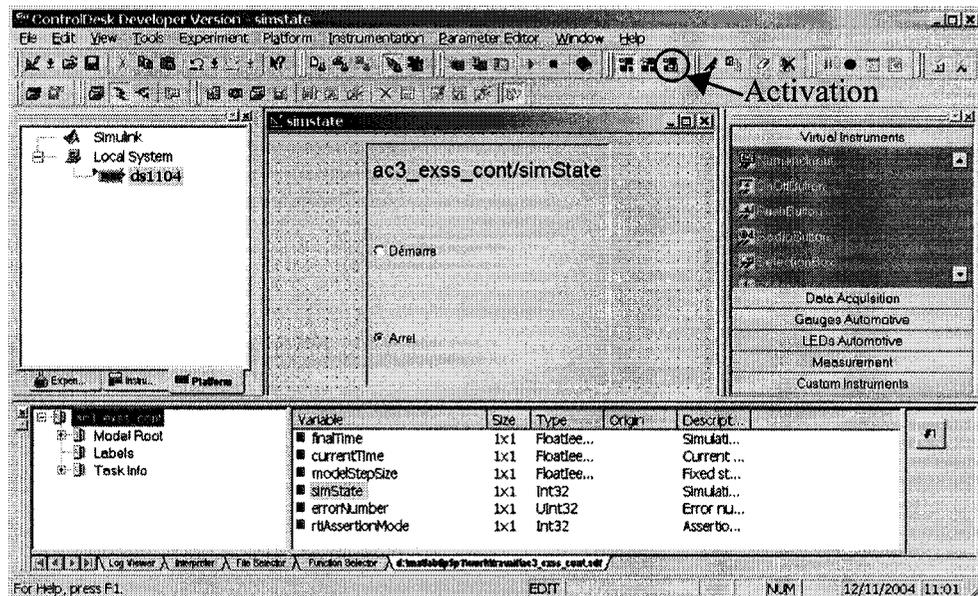


Figure 20 Interface de ControlDesk

Lors de la génération du code, DSSIMULINK crée un fichier dans ControlDesk contenant tous les signaux et variables du modèle Simulink. Ce fichier est présenté au bas de la figure 20. La variable *SimState* est contenue dans ce fichier. Pour utiliser cette variable, une interface de contrôle est développée dans la fenêtre de développement de ControlDesk. L'interface de contrôle comprend un seul bouton de type *RadioButton*. Ce bouton est sélectionné dans le sélecteur d'instruments et inséré dans la fenêtre de développement de ControlDesk. Sur la figure 20, le sélecteur d'instrument est situé à droite et la fenêtre de développement est située au centre. Le bouton est associé à la variable *SimState* en faisant glisser la variable sur le bouton. Il faut ensuite double cliquer sur le bouton pour accéder à sa configuration. La figure 21 présente la fenêtre de configuration du *RadioButton*.

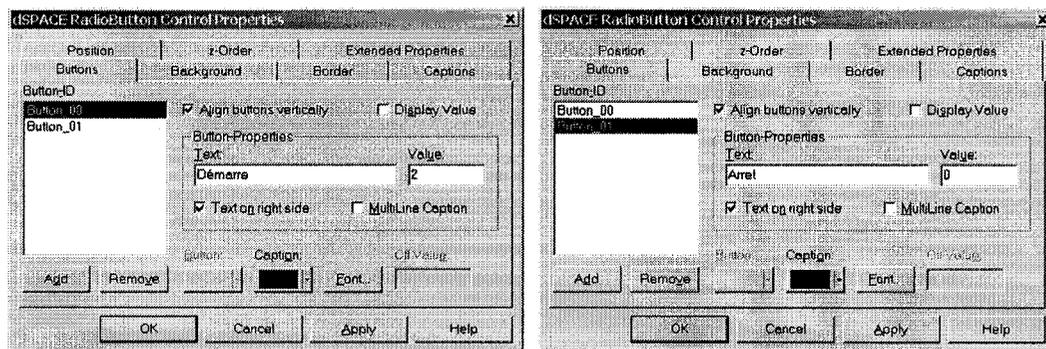


Figure 21 Fenêtres de configuration du *RadioButton*

Des valeurs sont associées aux états de la variable. La valeur 2 est associée à RUN, la valeur 1 à PAUSE et la valeur 0 à STOP. Le bouton utilisé est configuré pour sélectionner les états RUN et STOP. La figure 21 présente la configuration du bouton pour démarrer et arrêter la simulation. Pour démarrer l'application temps réel, le bouton d'activation doit tout d'abord être appuyé afin d'activer l'interface de contrôle dans ControlDesk. Ce bouton est identifié sur la figure 20. Lorsque l'interface de contrôle est

activée, l'application temps réel est démarrée en appuyant sur la case *Démarre* du *RadioButton*.

### 3.2.1.1 Modèle AC3

Comme présenté dans la section 2.2.1, le modèle AC3 comprend une commande composée d'une commande de vitesse et d'une commande à flux orienté (FOC). Cette commande nécessite quatre entrées analogiques de la carte DS1104 pour recevoir la vitesse mécanique du moteur ( $\omega_m$ ) et les courants statoriques triphasés ( $i_{abc}$ ) provenant du simulateur. Deux sorties analogiques sont nécessaires afin de transmettre les références des courants statoriques  $i_{as}^*$  et  $i_{bs}^*$  à l'onduleur triphasé du simulateur. La figure 22 présente le schéma Simulink de la commande du modèle AC3 à valeur moyenne.

Les quatre entrées analogiques non multiplexées de la carte DS1104, dont les numéros sont 5, 6, 7 et 8, sont utilisées. Les deux sorties analogiques utilisées sont les sorties 1 et 2. Il faut inscrire les numéros des entrées et des sorties dans les fenêtres de configuration des blocs comme présenté à la figure 18. Des gains sont ajoutés aux entrées et aux sorties pour faire correspondre les plages de valeurs. À partir des résultats de simulation en temps différé du modèle AC3 à valeur moyenne présentés dans la section 4.2 par la figure 36, la plage maximale de fonctionnement observée pour les courants est de -500 A à 500 A et la plage maximale pour la vitesse est de 0 RPM à 500 RPM. Dans le cas des E/S analogiques de dSPACE, elles fonctionnent sous une plage de valeur de -1 à 1 qui correspond à une plage de tension de -10 V à 10 V. Pour faire correspondre la plage des sorties, les références de courants sont divisées par un gain de 500 A, puis pour faire correspondre la plage des entrées, les courants et la vitesse mécanique sont multipliés par des gains de 500 A et 500 RPM.

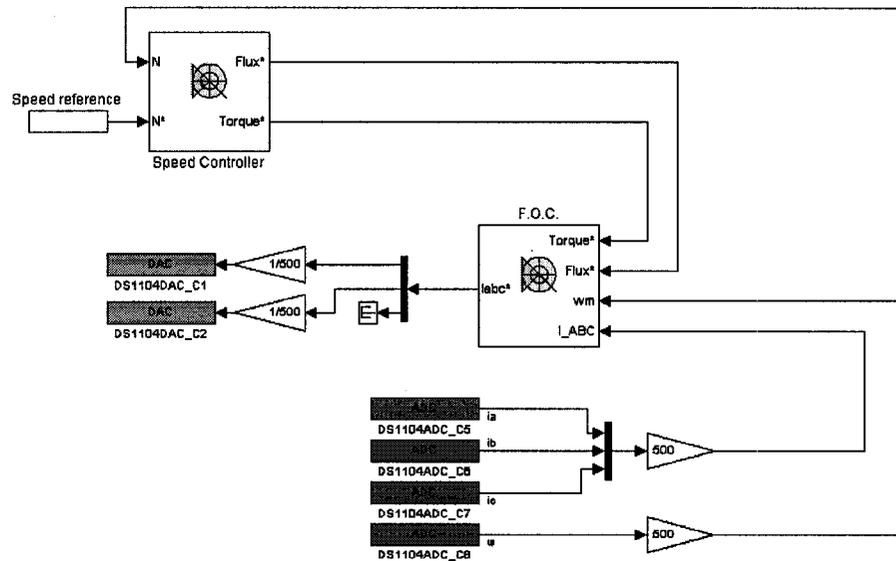


Figure 22 Schéma Simulink de la commande

Les résultats obtenus pour la simulation en temps réel du modèle AC3 à valeur moyenne avec commande externe sont présentés dans la section 4.4. Les premiers résultats n'étant pas satisfaisants, un modèle amélioré de la commande a été réalisé. Le schéma Simulink du modèle amélioré est présenté par la figure 23. Ce modèle comporte deux modifications par rapport au premier modèle.

La première modification est le passage de la vitesse mécanique de l'entrée analogique numéro 8 avec un CAN de 12 bits vers l'entrée analogique numéro 1 avec un CAN de 16 bits. Il faut changer le bloc d'entrée analogique pour le bloc d'entrées analogiques multiplexées. Pour configurer ce bloc il faut seulement sélectionner l'entrée numéro 1. Le passage de toutes les entrées vers les quatre entrées analogiques multiplexées avec le CAN de 16 bits n'est pas possible, puisque qu'après expérimentation, il a été observé que le temps de conversion était beaucoup trop élevé pour la simulation en temps réel.

La deuxième modification est le doublage de la résolution des signaux de sortie. Le principe utilisé est de séparer un signal en deux et de transmettre chaque partie sur une sortie différente. Puisque le signal est transmis sur deux sorties avec des CNA de 12 bits, la résolution du signal augmente à 24 bits. Le schéma de principe est présenté par la figure 24.

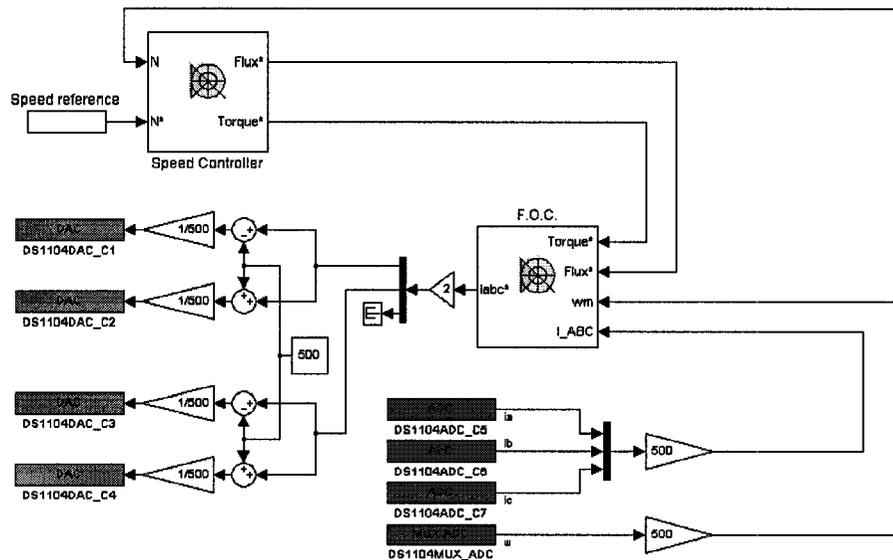


Figure 23 Schéma Simulink amélioré de la commande

Dans ce cas-ci, les signaux de sortie sont les références des courants statoriques. Les signaux sont premièrement multipliés par un gain de 2 afin de doubler leurs amplitudes et ainsi obtenir une plage maximale de courant de -1 000 A à 1 000 A. Un biais de 500 A est ensuite soustrait aux signaux pour obtenir une plage de -1 500 A à 500 A. Afin de faire correspondre les plages de sorties, les signaux sont divisés par un gain de 500 A. La plage résultante est de -3 à 1. Les sorties analogiques sont saturées pour les valeurs inférieures à -1, ce qui a pour effet de tronquer les alternances négatives et ainsi transmettre seulement les alternances positives des signaux. Pour obtenir les alternances négatives, la même démarche est utilisée en faisant l'addition du biais de 500 A. Les alternances positives des signaux sont transmises par les sorties 1 et 3, puis les

alternances négatives par les sorties 2 et 4. Les signaux sont par la suite reconstitués dans le simulateur et les détails sont présentés dans la section 3.3.1.1.

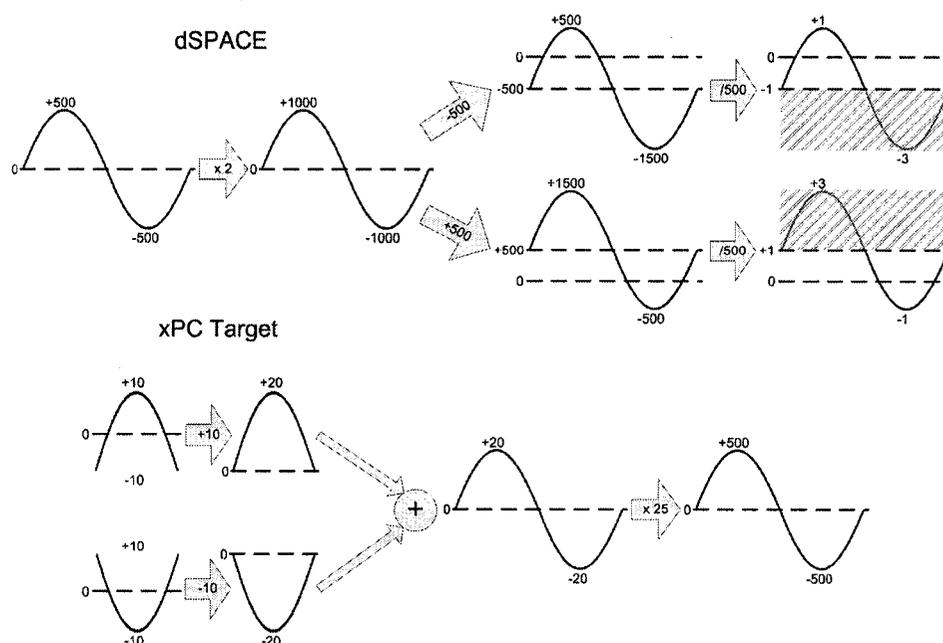


Figure 24 Schéma de principe du doublage de la précision

### 3.2.1.2 Modèle AC6

Comme présenté dans la section 2.2.2, le modèle AC6 comprend une commande composée d'une commande de vitesse et d'une commande vectorielle. Cette commande nécessite deux entrées analogiques de la carte DS1104 pour recevoir la vitesse mécanique du moteur ( $\omega_m$ ) et la position angulaire mécanique du rotor ( $\theta_r$ ) provenant du simulateur. Deux sorties analogiques sont nécessaires afin de transmettre les références des courants statoriques  $i_a^*$  et  $i_b^*$  à l'onduleur triphasé du simulateur. La figure 25 présente le schéma Simulink de la commande du modèle AC6 à valeur moyenne.

Les deux entrées analogiques 5 et 6 de la carte DS1104 sont utilisées. Les deux sorties analogiques utilisées sont les sorties 1 et 2. Des gains sont ajoutés aux entrées et aux

sorties pour faire correspondre les plages de valeurs. Les résultats de simulation en temps différé du modèle AC6 à valeur moyenne sont présentés dans la section 4.3 par les figures 37 et 38. La plage maximale de fonctionnement observée pour les courants est de -15 A à 15 A selon la figure 37. D'après la figure 38, la plage maximale pour la vitesse mécanique est de -1 rad/s à 32 rad/s et la plage maximale pour la position angulaire est de 0 rad à 32 rad. Pour faire correspondre la plage des sorties, les références de courants sont divisées par un gain de 15 A, puis pour faire correspondre la plage des entrées, la vitesse mécanique et la position angulaire sont multipliées par des gains de 32 rad/s et de 32 rad.

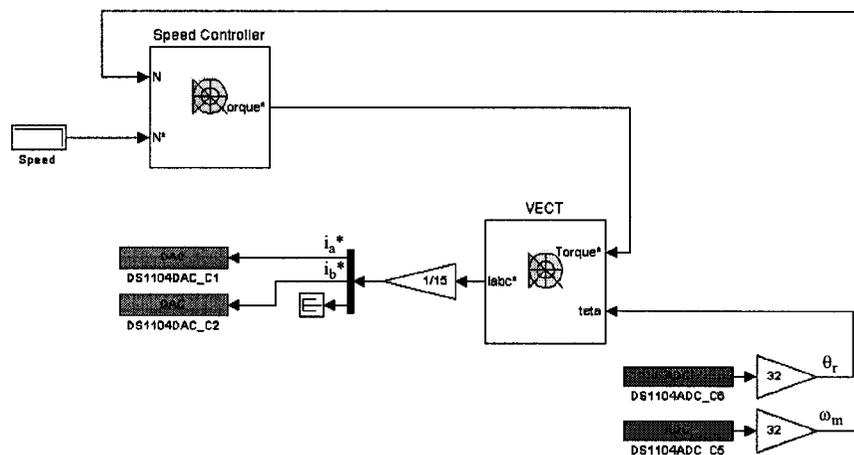


Figure 25 Schéma Simulink de la commande

Les résultats obtenus pour la simulation en temps réel du modèle AC6 à valeur moyenne avec commande externe sont présentés dans la section 4.5. Les premiers résultats n'étant pas satisfaisants, un modèle amélioré de la commande a été réalisé. Le schéma Simulink du modèle amélioré est présenté par la figure 26. Ce modèle comporte deux modifications par rapport au premier modèle.

La première modification est le doublage de la résolution des signaux de sortie. Le principe utilisé est le même que dans la section 3.2.1.1 comme présenté par la figure 24.

Les signaux de sortie sont les références des courants statoriques. Les signaux sont premièrement multipliés par un gain de 2 afin de doubler leurs amplitudes et ainsi obtenir une plage maximale de courant de -30 A à 30 A. Un biais de 15 A est ensuite soustrait aux signaux pour obtenir une plage de -45 A à 15 A. Afin de faire correspondre les plages de sorties, les signaux sont divisés par un gain de 15 A. La plage résultante est de -3 à 1. Puisque les sorties analogiques sont saturées pour les valeurs inférieures à -1, les alternances négatives sont tronquées et ainsi seulement les alternances positives des signaux sont transmises. Pour obtenir les alternances négatives, la même démarche est utilisée en faisant l'addition du biais de 15 A. Les alternances positives des signaux sont transmises par les sorties 1 et 3, puis les alternances négatives par les sorties 2 et 4. Les signaux sont par la suite reconstitués dans le simulateur et les détails sont présentés dans la section 3.3.1.2.

La deuxième modification est le doublage de la résolution des signaux d'entrée. Cette modification n'est pas possible dans le cas du modèle AC3 car les quatre sorties analogiques des cartes PCI-MIO-16E-4 sont utilisées. Dans le cas du modèle AC6, les deux signaux de sortie du simulateur peuvent être doublés. La commande reçoit donc la vitesse mécanique et la position angulaire sur quatre entrées et doit reconstituer ces signaux. Les deux parties de la vitesse mécanique sont reçues par les entrées 5 et 6, puis les deux parties de la position angulaire par les entrées 7 et 8.

Dans le cas de la vitesse mécanique, le signal à reconstituer possède une plage de -1 rad/s à 32 rad/s. Les signaux reçus par les entrées 5 et 6 possèdent une plage de -1 à 1. L'entrée 5 reçoit le signal qui correspond à la plage de 16,5 rad/s à 32 rad/s et l'entrée 6 reçoit le signal qui correspond à la plage de -1 rad/s à 16,5 rad/s. Un biais de 1 est additionné au signal de l'entrée 5 et soustrait au signal de l'entrée 6 pour obtenir des plages respectives de 0 à 2 et de -2 à 0. Les deux signaux sont ensuite additionnés pour obtenir un seul signal possédant une plage de -2 à 2. Le signal est multiplié par un gain de 16,5/2 rad/s ce qui produit une plage de -16,5 rad/s à 16,5 rad/s. Finalement, la

reconstitution du signal est complète en additionnant un biais de 15,5 rad/s pour obtenir la plage de -1 rad/s à 32 rad/s. Cette démarche suit similairement le principe présenté par la figure 24 pour reconstituer un signal provenant de la commande dans le simulateur. La même démarche est utilisée pour reconstituer le signal de la position angulaire qui possède une plage de 0 rad à 32 rad.

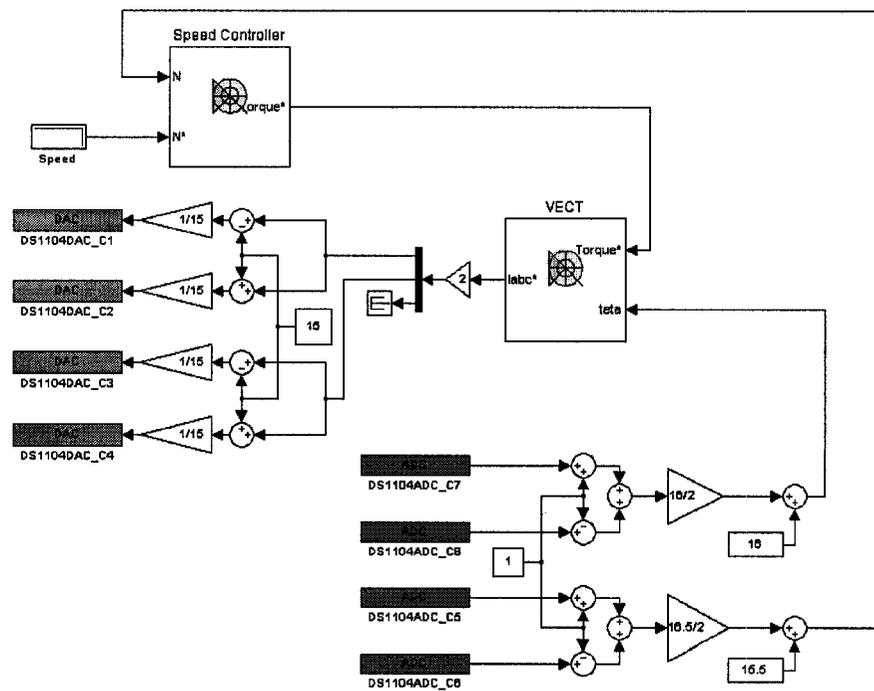


Figure 26 Schéma Simulink amélioré de la commande

### 3.3 Implémentation de l'électronique de puissance et de la machine

La simulation en temps réel de l'électronique de puissance et de la machine est réalisée avec l'ordinateur cible. Cet ordinateur constitue le simulateur en temps réel. Le schéma du montage matériel et logiciel complet utilisé dans la réalisation de la première partie du projet est présenté par la figure 15. Le simulateur communique avec la commande externe via deux cartes d'E/S PCI-MIO-16E-4 de National Instruments installées dans

les fentes PCI de l'ordinateur cible. Chaque carte possède seize entrées analogiques avec des CAN de 12 bits possédant un temps de conversion de 2  $\mu$ s, deux sorties analogiques avec des CNA de 12 bits possédant un temps de conversion de 1  $\mu$ s, huit E/S numériques et deux compteurs de 24 bits. Les entrées et les sorties analogiques ont une plage maximale de tension de -10 V à 10 V. L'accès aux signaux des E/S est réalisé avec des panneaux BNC-2110. L'accès aux signaux analogiques se fait via des prises BNC et l'accès aux signaux numériques se fait via un terminal de 30 broches. L'annexe 2 présente les spécifications techniques de la carte PCI-MIO-16E-4 sous la référence NI 6040E.

xPC Target est l'outil de simulation utilisé pour réaliser la simulation en temps réel des modèles à valeur moyenne. Les documents [3] et [4] ont été nécessaires pour l'utilisation d'xPC Target. Cet outil utilise des ordinateurs personnels standard. Son environnement est composé de l'ordinateur hôte et de l'ordinateur cible. L'ordinateur hôte sert d'outil de développement avec MATLAB et Simulink pour créer des modèles et l'ordinateur cible permet de réaliser les simulations en temps réel.

xPC Target permet l'addition des blocs d'E/S des cartes PCI-MIO-16E-4 dans les modèles. Il utilise Real-Time Workshop et le compilateur Microsoft Visual C/C++ pour créer du code exécutable. Le code est téléchargé de l'ordinateur hôte vers l'ordinateur cible qui est opéré par le noyau temps réel d'xPC Target. Après le téléchargement du code exécutable, l'application cible peut être exécutée en temps réel. Les étapes du processus de génération d'une application cible sont présentées par la figure 16.

Dans le projet de recherche, l'ordinateur hôte utilisé est muni d'un microprocesseur Intel Pentium 4 de 1,7 GHz et de 512 MB de mémoire RAM. L'ordinateur cible est muni d'un microprocesseur Intel Pentium 4 de 2,0 GHz et de 512 MB de mémoire RAM. D'un point de vue logiciel, la version 6.5.1 de MATLAB est utilisée. Cette version de MATLAB comprend la version 2.0.1 d'xPC Target, la version 5.1 de Simulink, la

version 5.1 de RTW et la version 3.0 de la librairie de SimPowerSystems (SPS). La version 6.0 du compilateur Microsoft Visual C/C++ est utilisée.

xPC Target ne requiert pas de système d'exploitation comme DOS, Windows ou Linux sur l'ordinateur cible. L'ordinateur cible est démarré avec une disquette de démarrage qui comprend le noyau temps réel d'xPC Target. Le noyau gère de manière optimale l'ordinateur cible pour la simulation en temps réel et n'accède pas le disque dur.

La disquette de démarrage est créée avec l'ordinateur hôte à l'aide d'un outil de configuration nommé xPC Target Setup. La figure 27 présente la fenêtre de l'outil de configuration, qui est obtenue en entrant la commande *xpcsetup* à l'invite de MATLAB.

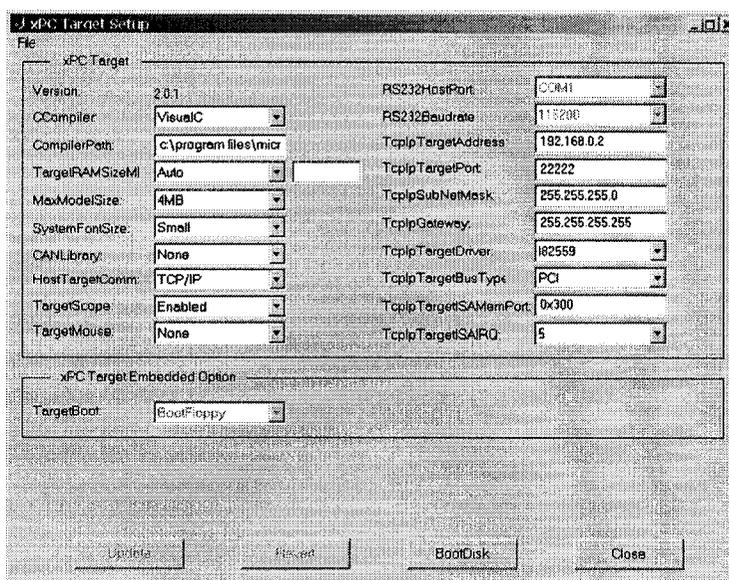


Figure 27 Fenêtre de configuration d'xPC Target

La configuration fournie à xPC Target l'information nécessaire sur les logiciels et le matériel utilisés. Les informations essentielles sont le compilateur C utilisé et son chemin d'accès, la taille de la mémoire RAM de l'ordinateur cible, la taille maximale de

la mémoire réservée pour l'application cible, le type de communication entre les ordinateurs, les adressages TCP/IP des ordinateurs, le pilote de la carte Ethernet de l'ordinateur cible et le type du bus de l'ordinateur cible. Les informations contenues dans la figure 27 sont les informations relatives au projet de recherche. La création de la disquette de démarrage est faite en appuyant sur le bouton *BootDisk* de la fenêtre xPC Target Setup.

Après son démarrage, le noyau affiche un message de bienvenue avec l'information sur la connexion entre l'ordinateur hôte et l'ordinateur cible. Le noyau active ensuite l'application de chargement et se met en attente pour le téléchargement de l'application cible à partir de l'ordinateur hôte. Lors du téléchargement, l'application de chargement reçoit le code de l'application cible, copie les différentes parties de code aux adresses désignées et prépare le démarrage de l'application cible.

Le code d'initialisation de l'application cible réserve le restant de la RAM inutilisée pour le noyau et la pile. La mémoire disponible est la RAM totale installée dans l'ordinateur cible moins les 4 MB réservés pour l'application cible. Dans le projet, la mémoire disponible pour le noyau et la pile est donc de 508 MB. Normalement, la plus grande partie de la pile est utilisée pour faire la sauvegarde des signaux durant la simulation.

L'application cible est capable d'exécuter des tâches en temps réel à très haute vitesse. Le temps d'exécution de tâche varie selon la grosseur et la complexité du modèle, ainsi que des composantes de l'ordinateur cible.

Le noyau temps réel d'xPC Target sauvegarde les données des signaux de l'application cible dans la RAM de l'ordinateur cible. L'acquisition des données est faite durant la simulation et la visualisation des signaux est faite lorsque la simulation est terminée. Les données sont recueillies pendant l'exécution des tâches et sont associées à un temps

précis. Lorsque la simulation est terminée, le téléchargement des données de l'ordinateur cible vers l'ordinateur hôte est fait pour ensuite visualiser les signaux.

La communication entre l'ordinateur hôte et l'ordinateur cible est faite à partir d'une connexion directe de type réseau via un câble Ethernet croisé. Les deux ordinateurs sont connectés au réseau avec des cartes Ethernet utilisant un protocole de communication TCP/IP. Puisque seulement deux cartes Ethernet sont supportées par xPC Target, la carte utilisée dans l'ordinateur cible du projet est la carte Pro/100S d'Intel, qui comporte un seul connecteur RJ45. Il n'y a pas de limitation pour la carte Ethernet de l'ordinateur hôte et la carte 3Com EtherLink XL 10/100 est utilisée. La connexion entre les deux ordinateurs permet de télécharger l'application cible de l'hôte vers la cible et de contrôler l'application cible. La connexion permet aussi de télécharger les données de l'ordinateur cible vers l'ordinateur hôte afin de visualiser les signaux.

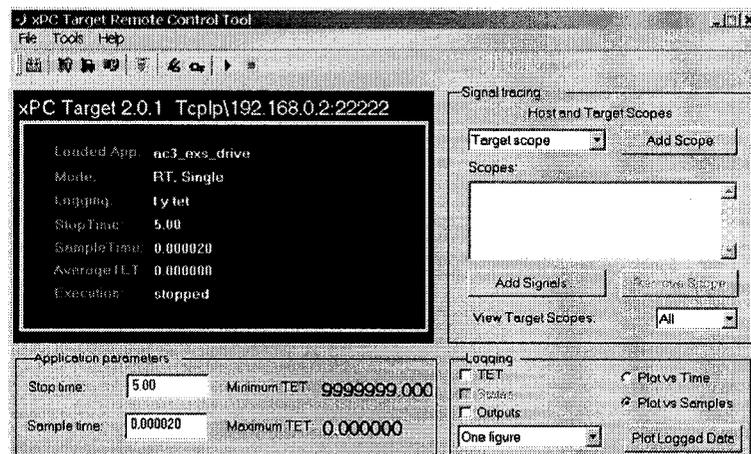


Figure 28 Fenêtre de contrôle d'xPC Target

Le téléchargement et le contrôle de l'application cible se fait de l'ordinateur hôte avec un outil nommé xPC Target Remote Control Tool. La fenêtre de contrôle, qui est présentée par la figure 28, est obtenue en entrant la commande *xpcrctool* à l'invite de MATLAB.

Cet outil permet de lancer la génération, la compilation et le téléchargement dans l'ordinateur cible du code de l'application cible à partir d'un fichier Simulink (.mdl). L'outil permet aussi le démarrage et l'arrêt de l'application cible, la modification de la période d'échantillonnage et du temps d'arrêt, ainsi que la réception de l'information sur les performances de l'application cible et du microprocesseur de l'ordinateur cible.

Le noyau d'xPC Target à deux modes pour l'exécution de la simulation en temps réel, soit le mode par interruption, qui est le mode par défaut, et le mode par polling. Le mode par interruption est le mode qui offre la plus grande flexibilité et qui devrait être utilisé pour exécuter des applications dont la période d'échantillonnage est relativement lente. Le désavantage de ce mode est qu'il présente une latence provenant du traitement des interruptions. Cette latence est d'environ 7  $\mu$ s pour le microprocesseur utilisé. Par contre, le mode par polling qui est considéré comme un mode plus brute et primitif, ne possède pas cette latence car il ne permet aucune interruption. Ce mode est donc privilégié pour exécuter une application cible dont la période d'échantillonnage est proche des limites du microprocesseur, ce qui est le cas dans le projet de recherche.

xPC Target supporte une vaste gamme de cartes d'E/S [5]. Les cartes utilisées dans le projet ont été mentionnées précédemment et sont deux cartes PCI-MIO-16E-4 de National Instruments. Ces cartes ont été choisies pour leurs multiples fonctionnalités et leurs performances. Les pilotes de ces cartes sont accessibles via les blocs de la librairie d'xPC Target dans Simulink. On insère et connecte les blocs de cette librairie dans le modèle Simulink comme n'importe quel autre bloc standard. Plusieurs types de blocs sont disponibles, soit des E/S analogiques, des E/S numériques, des supports séries RS-232/422/485, des compteurs, des encodeurs incrémentaux, etc. Dans cette partie du projet, les blocs utilisés sont les E/S analogiques.

### 3.3.1 Développement et configuration des modèles

Comme mentionné précédemment, une application cible est créée à partir d'un modèle Simulink. Dans le projet, les modèles consistent en l'électronique de puissance et les machines des modèles AC3 et AC6 à valeur moyenne. L'électronique de puissance est constituée dans les deux cas d'une source de tension CC et d'un onduleur triphasé. Les figures 32 à 35 présentent les schémas Simulink de l'électronique de puissance et des machines des modèles AC3 et AC6 à valeur moyenne.

Ces modèles comportent les blocs de pilotes pour les E/S analogiques des cartes PCI-MIO-16E-4. Ces blocs sont disponibles dans la librairie d'E/S de xPC Target de Simulink. La figure 29 présente les volets sous lesquels sont disponibles les E/S analogiques des cartes PCI-MIO-16E-4.

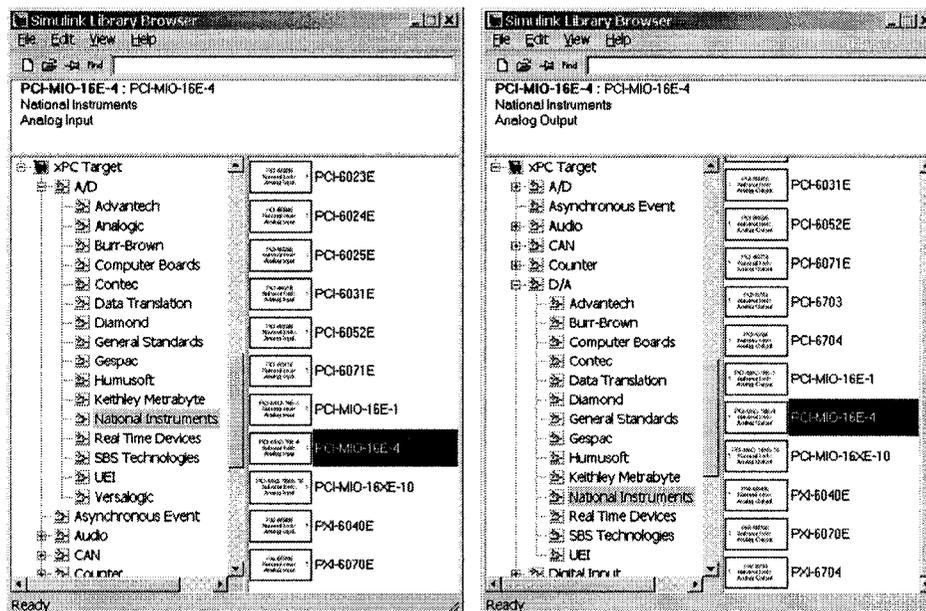


Figure 29 Blocs d'E/S analogiques de la carte PCI-MIO-16E-4

Le bloc d'entrée analogique doit être configuré avec les informations concernant le nombre d'entrées utilisées, les plages de tension désirées, le type de couplage des entrées, la période d'échantillonnage et l'emplacement de la carte sur le bus PCI. La figure 30 présente le bloc de configuration d'entrée analogique. Dans ce cas, il y a quatre entrées utilisées et leurs plages de tension sont de -10 V à 10 V. Le vecteur de couplage des entrées permet de déterminer si les entrées sont simples ou différentielles. Dans le projet, toutes les entrées analogiques sont simples. La période d'échantillonnage est de 20  $\mu$ s et l'emplacement de la carte est dans la neuvième fente du bus PCI de l'ordinateur cible. Les deux cartes PCI-MIO-16E-4 sont situées sur le bus PCI de l'ordinateur cible, la première carte est dans la neuvième fente et la deuxième carte est dans la dixième fente. L'aide de MATLAB présente en détail toutes les informations sur la configuration du bloc d'entrée analogique de la carte PCI-MIO-16E-4.

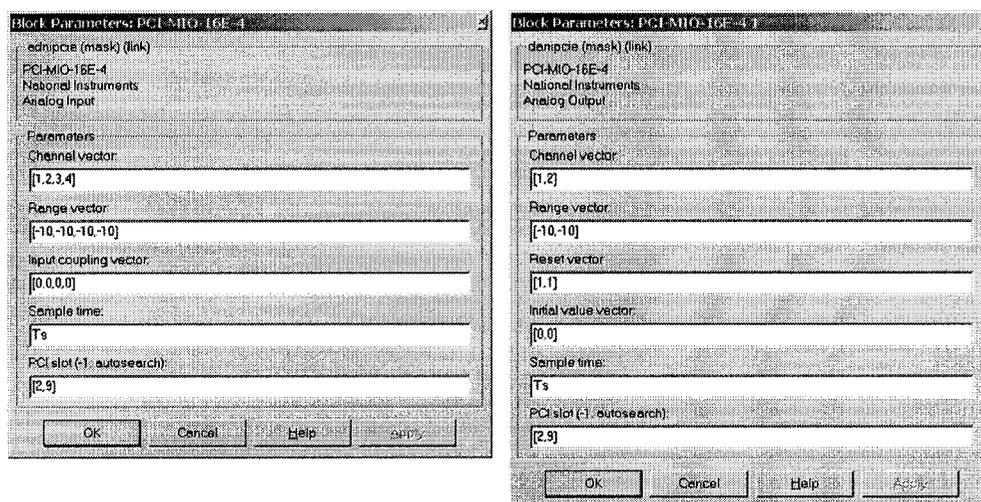


Figure 30 Fenêtres de configuration des blocs d'E/S analogiques

Pour les blocs de sortie analogique, les informations nécessaires concernent les numéros des sorties utilisés, les plages de tensions désirées, les valeurs initiales, le comportement des valeurs finales, la période d'échantillonnage et l'emplacements de la carte. La figure

30 présente le bloc de configuration de sortie analogique. Il y a deux sorties utilisées, les plages de tension sont de -10 V à 10 V, la période d'échantillonnage est de 20  $\mu$ s et la carte est située dans la neuvième fente du bus PCI de l'ordinateur cible. Les valeurs initiales des sorties sont de 0 V et les valeurs finales sont mises à 0 V à la terminaison du modèle. L'aide de MATLAB présente en détail toutes les informations sur la configuration du bloc de sortie analogique de la carte PCI-MIO-16E-4.

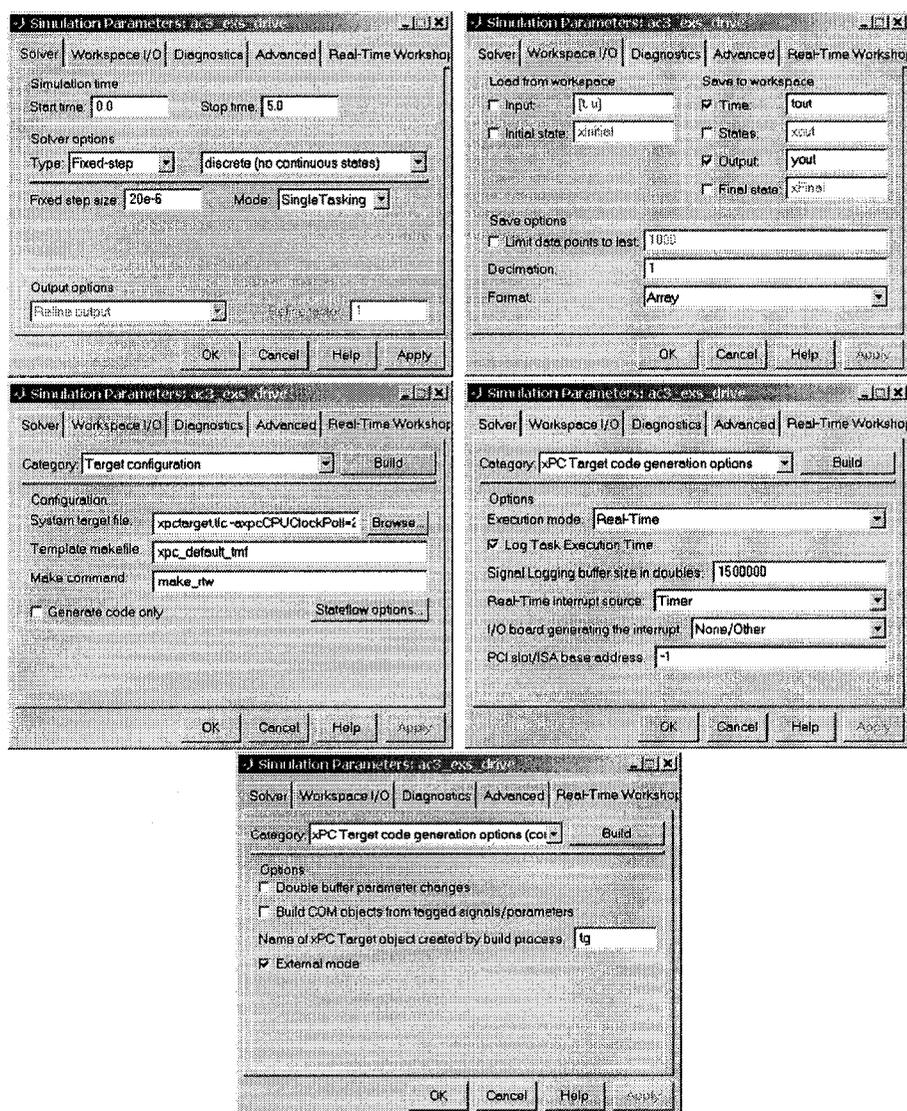


Figure 31 Fenêtres de configuration des paramètres d'xPC Target

Un modèle Simulink doit être configuré pour fonctionner avec xPC Target. La fenêtre de configuration des paramètres présentée à la figure 31 est obtenue en sélectionnant *Simulation parameters* dans le menu *Simulation* de la fenêtre Simulink du modèle. Dans l'onglet *Solver*, il faut inscrire les temps de départ et d'arrêt dans les cases *Start time* et *Stop time*. Dans ce cas la simulation débute à 0 s et se termine à 5 s. L'algorithme de calcul à pas fixe de type discret est choisi en sélectionnant *Fixed-step* dans la case *Type* et *Discrete* dans la case *Solver*. Par la suite, il faut inscrire la période d'échantillonnage de 20  $\mu$ s dans la case *Fixed-step size*. Finalement, il faut spécifier que le modèle exécute seulement une tâche en sélectionnant *Single Tasking* dans la case *Mode*.

Dans l'onglet *Workspace I/O*, il faut sélectionner *Time* et *Output* dans la section *Save to workspace* pour sauvegarder les signaux de sortie et le temps lors de la simulation du modèle. De plus, dans la section *Save options*, il faut s'assurer que la case *Limit data points to last* ne soit pas sélectionnée afin de ne pas limiter le nombre de points acquisitionnés lors de la simulation. Il faut aussi sélectionner *Array* dans la case *Format* afin d'obtenir des matrices de données comprenant les signaux de sorties et le temps.

Dans l'onglet *Real-Time Workshop*, en appuyant sur *Browse*, on sélectionne *xpctarget.tlc* pour générer le code C du modèle pour xPC Target. Par défaut, le mode d'exécution de la simulation en temps réel est le mode par interruption. Pour sélectionner le mode par polling, il faut inscrire *-axpcCLOCKPoll=2000* à la suite de *xpctarget.tlc* dans la case *System target file*, où le nombre 2 000 correspond à la vitesse du microprocesseur de l'ordinateur cible en mégahertz (MHz).

La configuration du modèle se poursuit en sélectionnant *xPC Target code generation options* dans le menu déroulant *Category*. Il faut spécifier que l'exécution est réalisée en temps réel en sélectionnant *Real-Time* dans la case *Execution mode*. Il faut aussi spécifier le nombre maximum de points pouvant être acquisitionnés dans la case *Signal Logging buffer size in doubles*. Le nombre choisi est de 1 500 000 pour le cas demandant

le plus de points, soit dans le cas du modèle AC3 où il y a six signaux échantillonnés pendant 5 s avec une période d'échantillonnage de 20  $\mu$ s.

### 3.3.1.1 Modèle AC3

Comme présenté dans la section 2.2.1, le modèle AC3 comprend une source CC idéale, un onduleur triphasé et une machine asynchrone. Le simulateur nécessite deux entrées analogiques d'une carte PCI-MIO-16E-4 pour recevoir les références des courants statoriques  $i_{as}^*$  et  $i_{bs}^*$  de la commande. Les deux entrées analogiques utilisées sont les entrées 1 et 2 de la première carte située dans la neuvième fente du bus PCI de l'ordinateur cible. Le simulateur nécessite aussi les quatre sorties analogiques des cartes pour transmettre la vitesse mécanique du moteur ( $\omega_m$ ) et les courants statoriques triphasés ( $i_{abcs}$ ) à la commande. La figure 32 présente le schéma Simulink de l'électronique de puissance et de la machine du modèle AC3 à valeur moyenne.

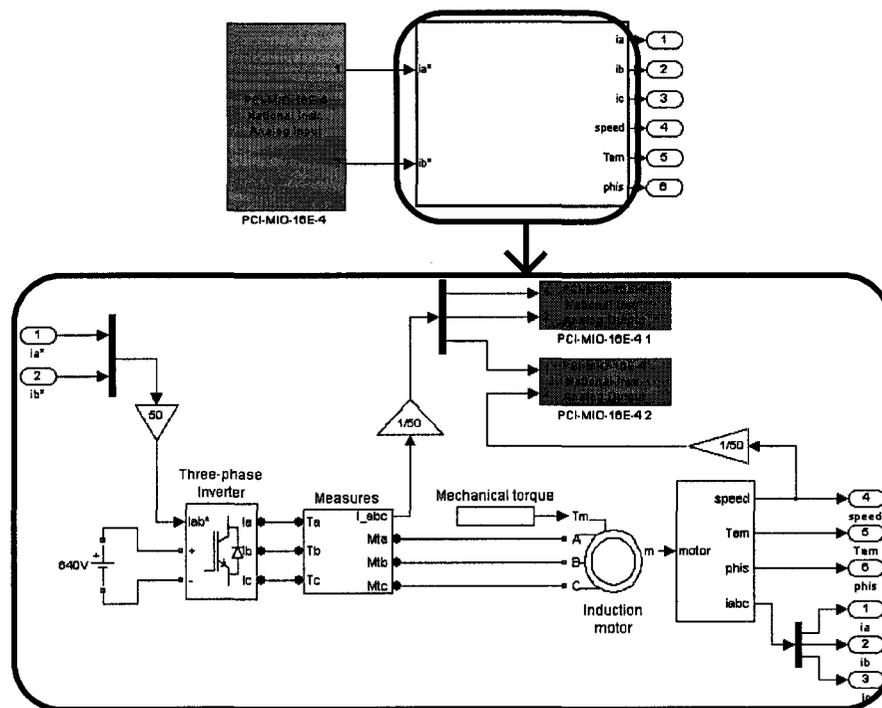


Figure 32 Schéma Simulink de l'électronique de puissance et de la machine

La configuration des blocs d'E/S analogiques du modèle est la même que celle présentée à la figure 30, à l'exception du nombre d'entrées analogiques qui est de deux. Des gains sont ajoutés aux entrées et aux sorties pour faire correspondre les plages de valeur tout en étant cohérent avec l'ajustement des plages de la commande. Comme mentionné dans la section 3.2.1.1, la plage maximale de fonctionnement observée pour les courants est de -500 A à 500 A et la plage maximale pour la vitesse est de 0 RPM à 500 RPM. Les plages de fonctionnement des E/S analogiques des cartes PCI-MIO-16E-4 sont de -10 V à 10 V. Pour faire correspondre la plage des entrées, les références de courants sont multipliées par un gain de 50 A/V, puis pour faire correspondre la plage des sorties, les courants et la vitesse mécanique sont divisés par des gains de 50 A/V et de 50 RPM/V.

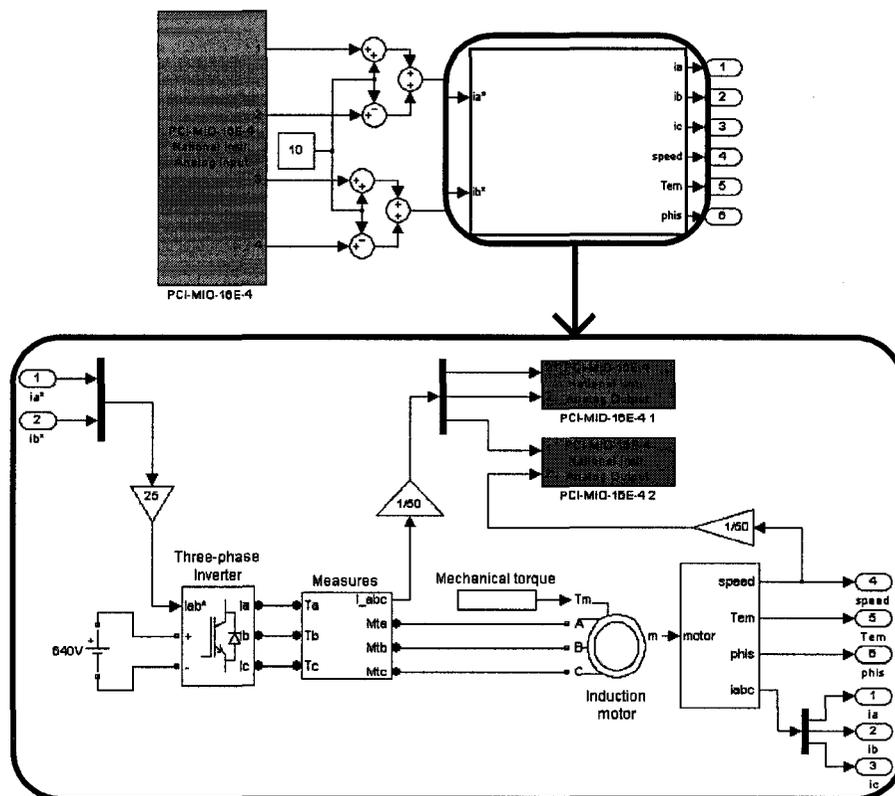


Figure 33 Schéma Simulink amélioré de l'électronique de puissance et de la machine

Les résultats obtenus pour la simulation en temps réel du modèle AC3 à valeur moyenne avec commande externe sont présentés dans la section 4.4. Comme spécifié dans la section 3.2.1.1, un modèle amélioré de la commande a été réalisé car les premiers résultats n'étaient pas satisfaisants. Un modèle amélioré de l'électronique de puissance et de la machine a aussi été réalisé pour être compatible avec la commande. Le schéma Simulink du modèle amélioré est présenté par la figure 33. Ce modèle comporte une modification par rapport au premier modèle.

La modification est le doublage de la résolution des signaux d'entrées. Le schéma de principe de la reconstitution des signaux provenant de la commande est présenté par la figure 24. Le modèle reçoit les références des courants statoriques  $i_{as}^*$  et  $i_{bs}^*$  sur quatre entrées et doit reconstituer ces signaux. Les deux parties de la référence  $i_{as}^*$  sont reçues par les entrées 1 et 2, tandis que les deux parties de la référence  $i_{bs}^*$  par les entrées 3 et 4.

Les deux signaux à reconstituer possèdent une plage de -500 A à 500 A. Les signaux reçus par les entrées possèdent une plage de -10 V à 10 V. Les entrées impaires reçoivent les signaux qui correspondent aux plages de 0 A à 500 A et les entrées paires reçoivent les signaux qui correspondent aux plages de -500 A à 0 A. Un biais de 10 V est additionné aux signaux des entrées impaires et soustrait aux signaux des entrées paires pour obtenir des plages respectives de 0 V à 20 V et de -20 V à 0 V. Les signaux 1 et 2 sont ensuite additionnés pour obtenir un seul signal possédant une plage de -20 V à 20 V. La même opération est réalisée avec les signaux 3 et 4. Finalement, les deux signaux résultants sont multipliés par un gain de 25 A/V pour obtenir des plages de -500 A à 500 A. Les deux références de courants sont alors reconstituées. La configuration de blocs d'E/S analogiques est la même que celle présentée par la figure 30. Cependant, il faut ajouter un second bloc pour les sorties de la deuxième carte située dans la dixième fente du bus PCI de l'ordinateur cible. La configuration de ce bloc est la même que celle de la figure 30 à l'exception de l'emplacement de la carte. L'emplacement est modifié en inscrivant [2,10] dans la case *PCI Slot*.

### 3.3.1.2 Modèle AC6

Comme présenté dans la section 2.2.2, le modèle AC6 comprend une source CC idéale, un onduleur triphasé et une machine synchrone à aimants permanents. Le simulateur nécessite deux entrées analogiques pour recevoir les références des courants statoriques  $i_a^*$  et  $i_b^*$  de la commande. Les deux entrées analogiques 1 et 2 de la première carte PCI-MIO-16E-4 située dans la neuvième fente du bus PCI de l'ordinateur cible sont utilisées. Le simulateur nécessite aussi deux sorties analogiques pour transmettre la vitesse mécanique du moteur ( $\omega_m$ ) et la position angulaire du rotor ( $\theta_r$ ) à la commande. Les sorties 1 et 2 de la première carte sont utilisées. La figure 34 présente le schéma Simulink de l'électronique de puissance et de la machine du modèle AC6 à valeur moyenne.

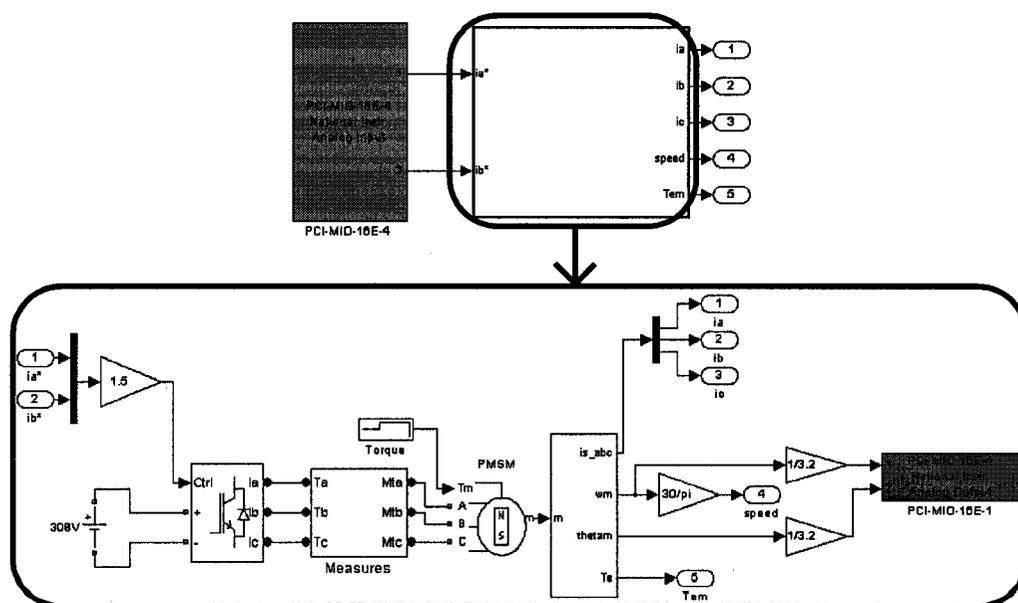


Figure 34 Schéma Simulink de l'électronique de puissance et de la machine

La configuration des blocs d'E/S analogiques du modèle est la même que celle présentée à la figure 30, mais pour seulement deux entrées analogiques. Des gains sont ajoutés aux

E/S du modèle pour faire correspondre les plages de valeur de façon cohérente avec l'ajustement des plages de la commande. Comme mentionné dans la section 3.2.1.1, la plage maximale de fonctionnement observée pour les courants est de -15 A à 15 A, la plage maximale pour la vitesse mécanique est de -1 rad/s à 32 rad/s et la plage maximale pour la position angulaire est de 0 rad à 32 rad. Les plages de fonctionnement des E/S analogiques des cartes PCI-MIO-16E-4 sont de -10 V à 10 V. Pour faire correspondre la plage des entrées, les références des courants sont multipliées par un gain de 1,5 A/V, tandis que pour faire correspondre la plage des sorties, la vitesse mécanique et la position angulaire sont divisés par des gains de 3,2 (rad/s)/V et de 3,2 rad/V.

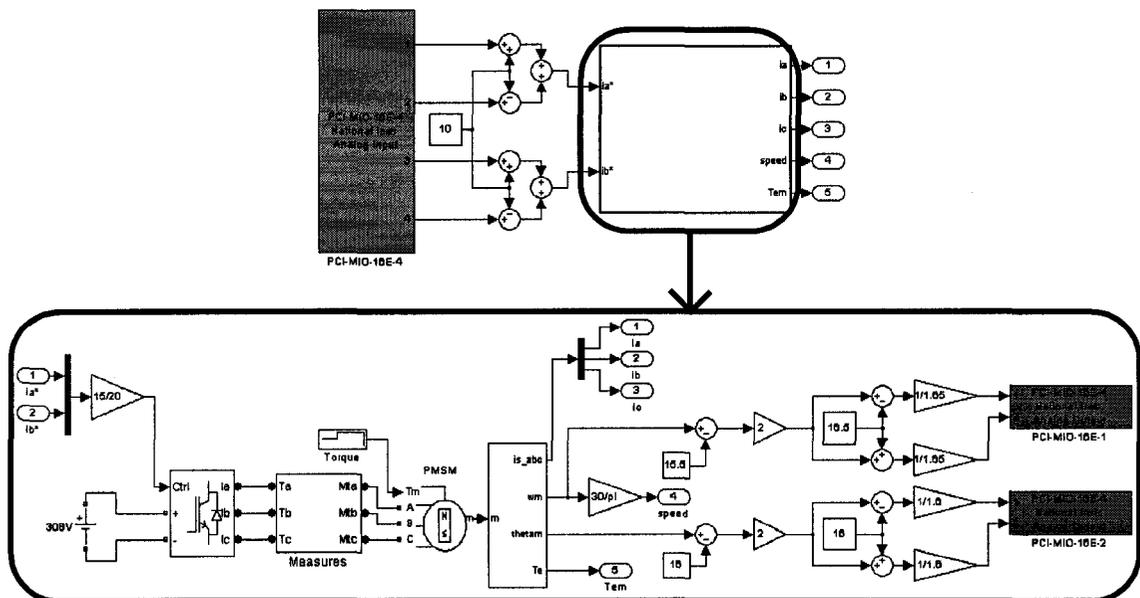


Figure 35 Schéma Simulink amélioré de l'électronique de puissance et de la machine

Les résultats obtenus pour la simulation en temps réel du modèle AC6 à valeur moyenne avec commande externe sont présentés dans la section 4.5. Comme spécifié dans la section 3.3.1.2, un modèle amélioré de la commande a été réalisé car les premiers résultats n'étaient pas satisfaisants. Un modèle amélioré de l'électronique de puissance et de la machine a aussi été réalisé pour être compatible avec la commande. Le schéma

Simulink du modèle amélioré est présenté par la figure 35. Ce modèle comporte deux modifications par rapport au premier modèle.

La première modification est le doublage de la résolution des signaux d'entrées. Le schéma de principe de la reconstitution des signaux provenant de la commande est présenté par la figure 24. Le modèle doit reconstituer les deux références des courants statoriques reçues sur quatre entrées. Les deux parties de la référence  $i_a^*$  sont reçues par les entrées 1 et 2, tandis que les deux parties de la référence  $i_b^*$  sont reçues par les entrées 3 et 4.

Les deux signaux à reconstituer possèdent une plage de -15 A à 15 A. Les signaux reçus par les entrées possèdent une plage de -10 V à 10 V. Les entrées impaires reçoivent les signaux qui correspondent aux plages de 0 A à 15 A et les entrées paires reçoivent les signaux qui correspondent aux plages de -15 A à 0 A. Un biais de 10 V est additionné aux signaux des entrées impaires et soustrait aux signaux des entrées paires pour obtenir des plages respectives de 0 V à 20 V et de -20 V à 0 V. Les signaux 1 et 2 sont ensuite additionnés pour obtenir un seul signal possédant une plage de -20 V à 20 V. La même opération est réalisée avec les signaux 3 et 4. Finalement, les deux signaux résultants sont multipliés par un gain de 15/20 A/V pour obtenir des plages de -15 A à 15 A. Les deux références de courants sont alors reconstituées.

La deuxième modification est le doublage de la résolution des signaux de sortie. Le principe utilisé est le même que dans la section 3.2.1.1 comme présenté par la figure 24. Les signaux de sortie sont la vitesse mécanique et la position angulaire du rotor. Tout d'abord, un biais est soustrait aux signaux puisqu'ils ne sont pas symétriques par rapport à l'abscisse. Le biais est de 15,5 rad/s pour la vitesse et de 16 rad pour la position angulaire. Les plages de valeurs résultantes sont de -16,5 rad/s à 16,5 rad/s et de -16 rad à 16 rad. Les signaux sont ensuite multipliés par un gain de 2 pour doubler leurs amplitudes et ainsi obtenir des plages maximales de -33 rad/s à 33 rad/s et de -32 rad à 32

rad. Un biais de 16,5 rad/s est soustrait et additionner au signal de la vitesse pour obtenir deux signaux avec des plages respectives de -49,5 rad/s à 16,5 rad/s et de -16,5 rad/s à 49,5 rad/s. Dans le cas de la position angulaire, la même opération est effectuée avec un biais de 16 rad pour obtenir deux signaux avec des plages de -48 rad à 16 rad et de -16 rad à 48 rad. Les signaux de vitesse sont divisés par un gain de 1,65 (rad/s)/V, ce qui produit des plages de -30 V à 10 V et de -10 V à 30 V. Les mêmes plages sont obtenues pour les signaux de la position angulaire avec la division par un gain de 1,6 rad/V. Puisque les sorties analogiques sont saturées entre -10 V et 10 V, les plages de -30 V à -10V et les plages 10 V à 30 V sont tronquées. La partie supérieure de la vitesse est alors transmise par un signal et la partie inférieure par un autre signal. Le même résultat est obtenu pour la position angulaire. Les deux signaux de la vitesse sont transmis à la commande par les deux sorties analogiques de la première carte PCI-MIO-16E-4. Pour ce qui est des deux signaux de la position angulaire, ils sont transmis par les deux sorties analogiques de la deuxième carte PCI-MIO-16E-4. Les signaux sont par la suite reconstitués dans la commande et les détails sont présentés dans la section 3.2.1.2.

La configuration de blocs d'E/S analogiques est la même que celle présentée par la figure 30. Cependant, il faut ajouter un second bloc pour les sorties de la deuxième carte située dans la dixième fente du bus PCI de l'ordinateur cible. La configuration de ce bloc est la même que celle de la figure 30 à l'exception de l'emplacement de la carte. L'emplacement est modifié en inscrivant [2,10] dans la case *PCI Slot*.

### 3.4 Conclusion

Ce chapitre a présenté la démarche utilisée pour réaliser la simulation en temps réel avec commande externe des modèles d'entraînement électrique à valeur moyenne présentés dans le chapitre 2. Tous les outils matériels et logiciels utilisés ainsi que les étapes nécessaires à la réalisation de cette partie du projet ont été présentés.

Le prochain chapitre présente et valide les résultats obtenus pour la simulation en temps réel avec commande externe des modèles d'entraînement électrique à valeur moyenne. La validation de la simulation est réalisée par la comparaison des résultats obtenus avec des résultats de simulation en temps différé.

[MCours.com](https://www.m-cours.com)