

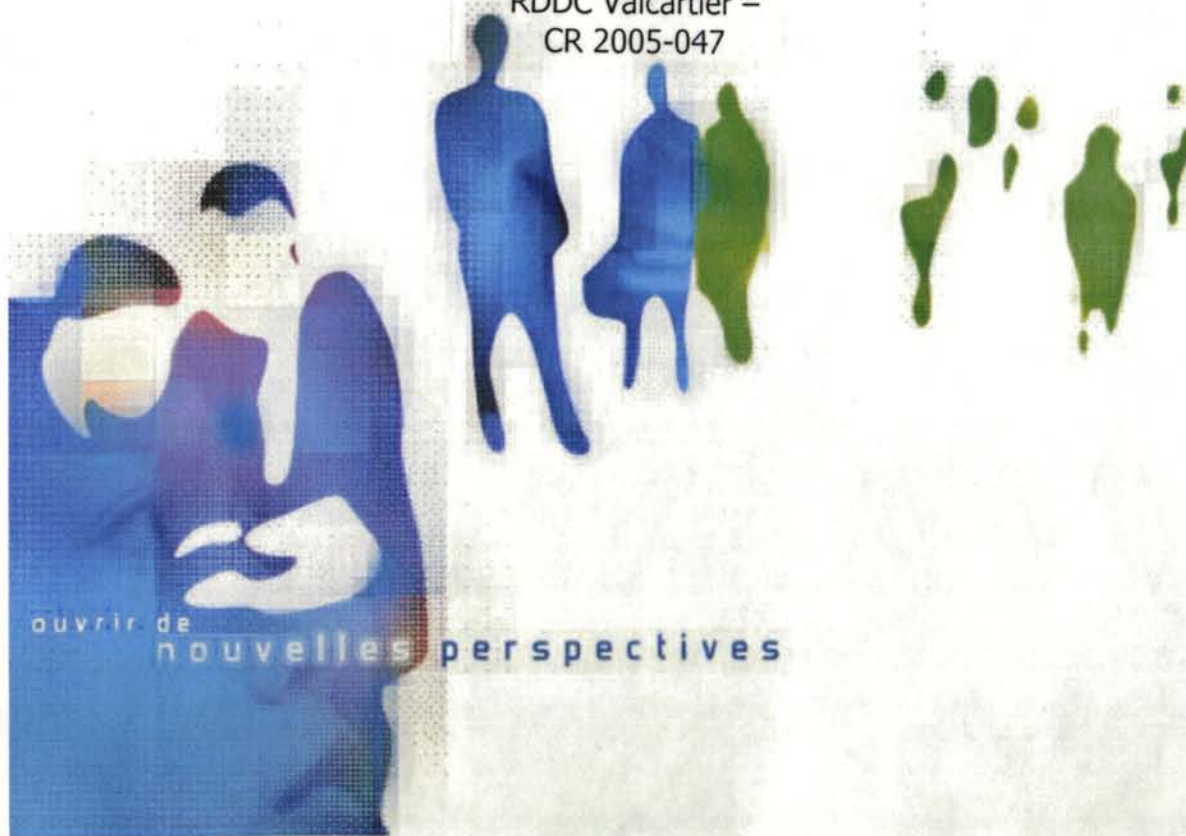


# Expérimentation HWIL du concept HEMi

Pour  
La Section Armes de précision  
R&D pour la défense Canada - Valcartier

par  
*Beaudoin Plante ing*  
*Contrat TPSGC no. W7701-3-1966*  
*Responsable du projet scientifique: Richard Lestage*

RDDC Valcartier –  
CR 2005-047



ouvrir de  
nouvelles perspectives



## TABLE DES MATIÈRES

1.	INTRODUCTION .....	2
2.	VUE D'ENSEMBLE DU HIL .....	3
3.	SIMPLAY .....	4
4.	LABVIEW HOST .....	5
5.	PXI .....	6
6.	SIMULINK.....	7
7.	HARDWARE .....	9
8.	CONCLUSION .....	10



## RÉSUMÉ

Lors de la réalisation du concept HEMi, plusieurs activités ont été réalisées dans différents domaines technologiques. Dans le cadre de ce projet, partie intégrante du TDP HEMi, il a été démontré la possibilité d'intégrer un modèle de missile dans un système « hardware in the loop » (HIL) comprenant les véritables capteurs de guidage. Ce travail a été réalisé dans le but de permettre la vérification de la performance des capteurs de guidage dans un environnement de laboratoire en utilisant un modèle du missile.

Ce projet a permis de démontrer la possibilité de réaliser une interopérabilité entre différents outils tels que Simulink et Labview dans un contexte temps réel (real time). Cette approche donne des résultats très intéressants sur le PXI. Des tests de performance ont été réalisés avec un modèle de missile et la vitesse de boucle (loop) a été vérifiée jusqu'à 1000 Hz. Les données obtenues avec Labview en comparaison de ceux du même modèle dans Simulink sont identiques à une précision de  $10^{-6}$ .

## ABSTRACT

During the realization of the HEMi concept, several activities have been realized in different technological areas. As part of this project related to the HEMi TDP, a hardware-in-the-loop system composed of real guidance sensors and a model of the missile has been demonstrated. This work has been realized in order to allow the validation of the performance of the guidance sensors in a laboratory experiment using a model of the missile.

This project has demonstrated the interoperability between different modelling tools such as Simulink and Labview in the context of real-time operations. This approach gives interesting results for a PXI computer platform. Performance tests have been realised with a missile model up to a rate of 1000 Hz. Results obtained directly in Simulink compared with the one from the same model in Labview are identical to an accuracy of  $10^{-6}$ .



## 1. INTRODUCTION

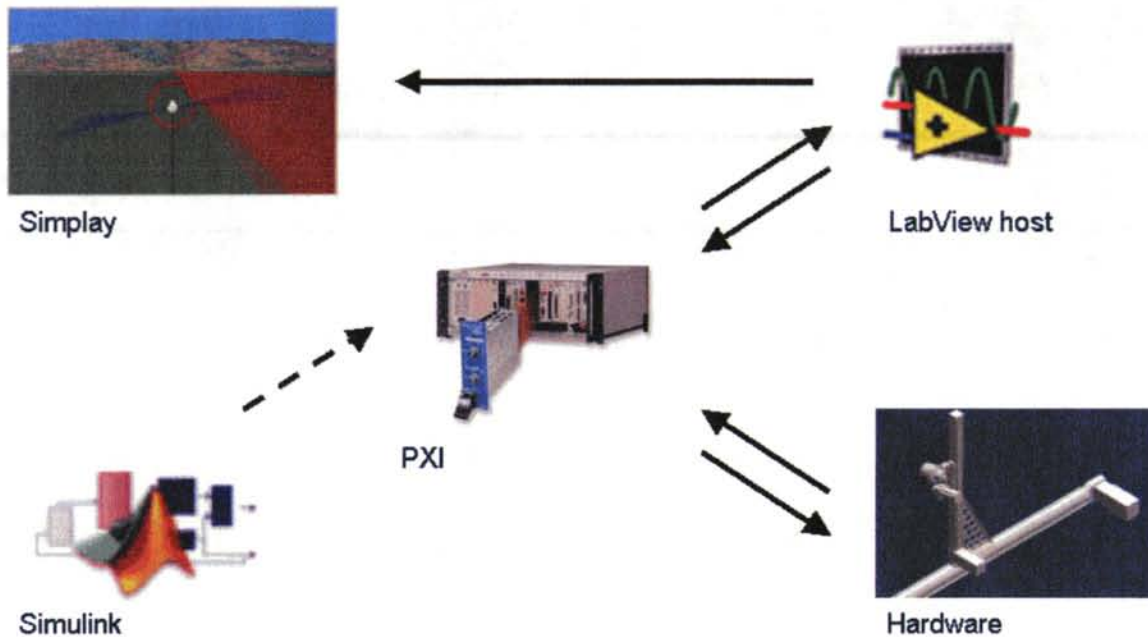
Lors de la réalisation du concept HEMi, plusieurs activités ont été réalisées dans différents domaines technologiques. Dans le cadre de ce projet, partie intégrante du TDP HEMi, il a été démontré la possibilité d'intégrer un modèle de missile dans un système « hardware in the loop » (HIL) comprenant les véritables capteurs de guidage. Ce travail a été réalisé dans le but de permettre la vérification de la performance des capteurs de guidage dans un environnement de laboratoire en utilisant un modèle du missile.

L'intérêt principal de cette approche est d'étudier le comportement d'un vrai système de guidage, sans avoir les contraintes associées à des tirs réels de missile. De cette façon, les faiblesses et les problèmes opérationnels sont détectés très tôt dans le processus de conception de missile. Les ajustements et corrections sont donc rapidement réalisés dans un environnement sécuritaire de laboratoire.

Il serait illusoire de penser que tous les problèmes de tir de missiles seront réglés avec ce seul banc de test. Cependant cette approche offre quelques avantages et non les moindres : Diminuer le temps de conception d'un missile, réaliser des tirs en laboratoire, dans différentes conditions environnementales, difficilement reproductibles en réalité et il ne faudrait pas oublier l'aspect financier entre un tir de missile en laboratoire versus un tir réel.

Maintenant que les motivations de la réalisation de ce projet sont connues, le présent document fait le tour des différentes technologies exploitées dans ce projet de même que leurs modes d'exploitation.

## 2. VUE D'ENSEMBLE DU HIL



La principale activité au cours de ce projet est de réaliser une application Labview distribuée qui intègre différentes technologies qui seront détaillées dans les prochains chapitres. Voici les principales :

**Simplay** : C'est une application de visualisation dans un environnement synthétique du déroulement du tir de missile.

**Labview host** : Permet de contrôler le déroulement du tir

**PXI**: Le cœur de l'installation, il supporte l'application Labview Target qui contient le modèle Simulink du missile. Contrôlé par le LabviewHost, il acquiert les données des capteurs du Hardware et envoie les commandes à ce dernier.

**Simulink**: C'est le modèle de missile proprement dit. Ce modèle est généré avec un « RTW system » particulier pour être utilisé dans Labview sur la plate-forme PXI.

**Hardware**: C'est l'ensemble du montage qui contient la tête de missile avec ses senseurs et ses contrôleurs de positions opérés par le PXI.



### 3. SIMPLAY

La principale tâche de ce logiciel est de réaliser une visualisation synthétique des opérations réalisées avec les installations.

Le groupe « Simulation d'engagements, systèmes d'armes » a développé ce produit. Pour ce projet, le produit a été modifié pour permettre une nouvelle fonctionnalité, soit le mode « control activeX ».

Pour l'utiliser en mode activeX, il faut préalablement installer Simplay et s'assurer de son bon fonctionnement avec le fichier de démonstration « DemoSimplay.xls » qui nécessite les objets suivants :

Hemibooster.plb  
HEMIdart.plb  
grid.plb  
circle.plb  
tube.plb  
t-80.plb  
lake.plb  
golfhole.plb

Si la démonstration ne fonctionne pas bien, l'installation ne s'est peut être pas bien faite. Il faudra fermer la démonstration et procéder à l'enregistrement du composant OCX par l'opération suivante :

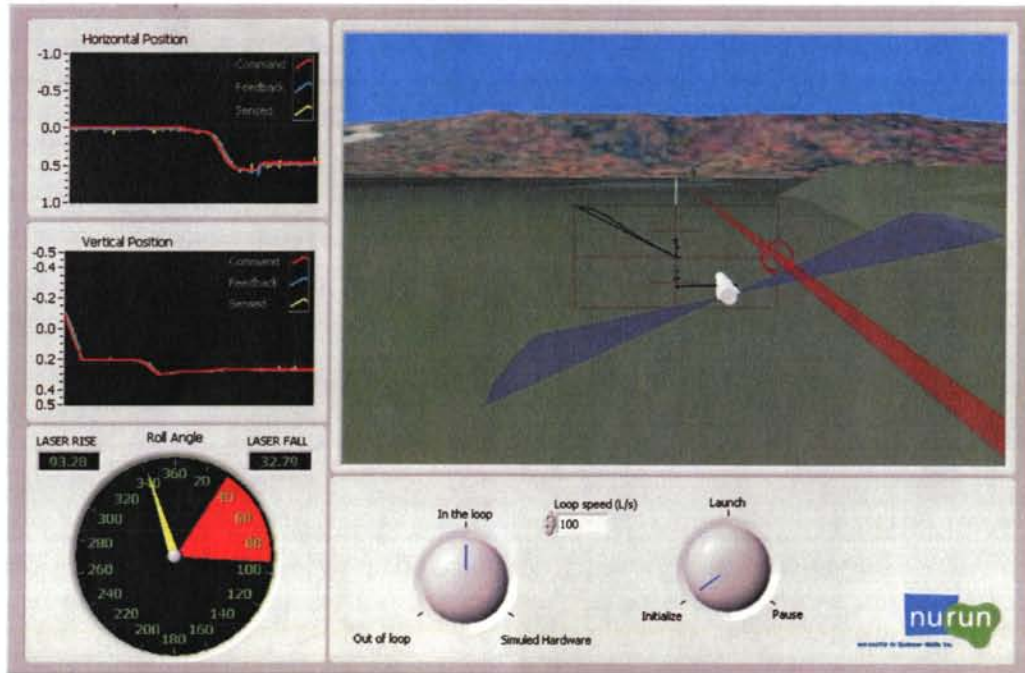
Dans la dialog box « run », taper regsvr32.exe "<votrérépertoire>\Simplay\SimplayOCX.ocx" et réessayer la démonstration.

Pour en connaître plus sur le mode « Control activeX », voir la littérature de Microsoft sous la rubrique « control activeX ». Pour connaître les API de Simplay, consultez l'aide de ce dernier.

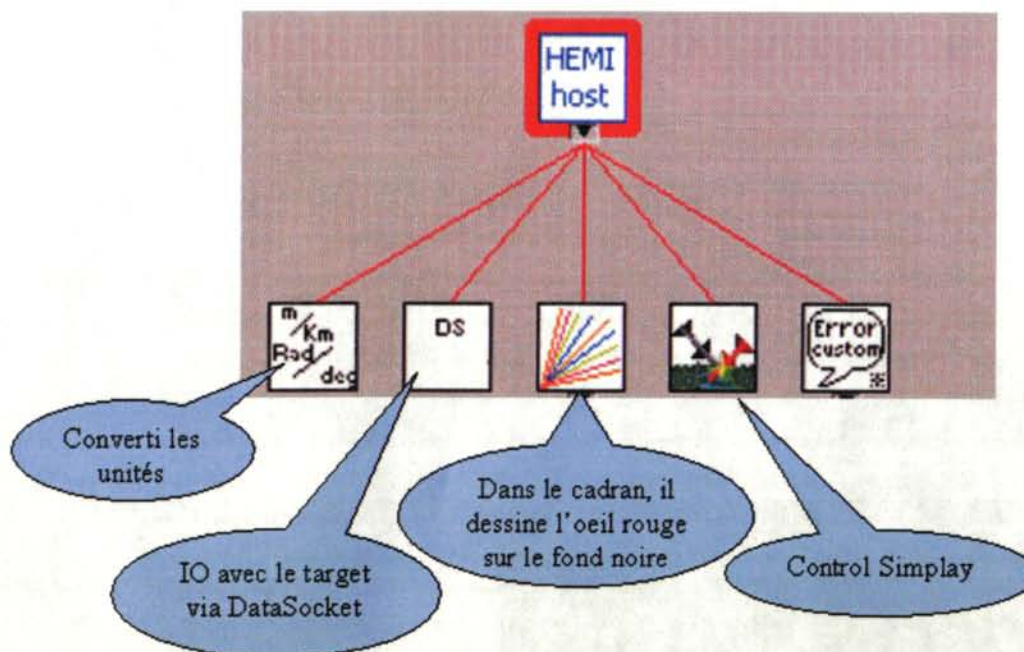


#### 4. LABVIEW HOST

La principale utilité de cet outil est de permettre à l'utilisateur d'interagir avec l'application. Elle contient le control activeX Simplay et communique avec le Target par le protocole de communication « dataSocket ». Cette application se trouve sous le répertoire ...\\Host\\Host\_Main.vi



Cet outil développé en Labview, nécessite une compilation en .exe pour pouvoir être utilisé en simultané avec l'interface usager de l'application du PXI. Voici sommairement les fonctions des différents sous-vi.



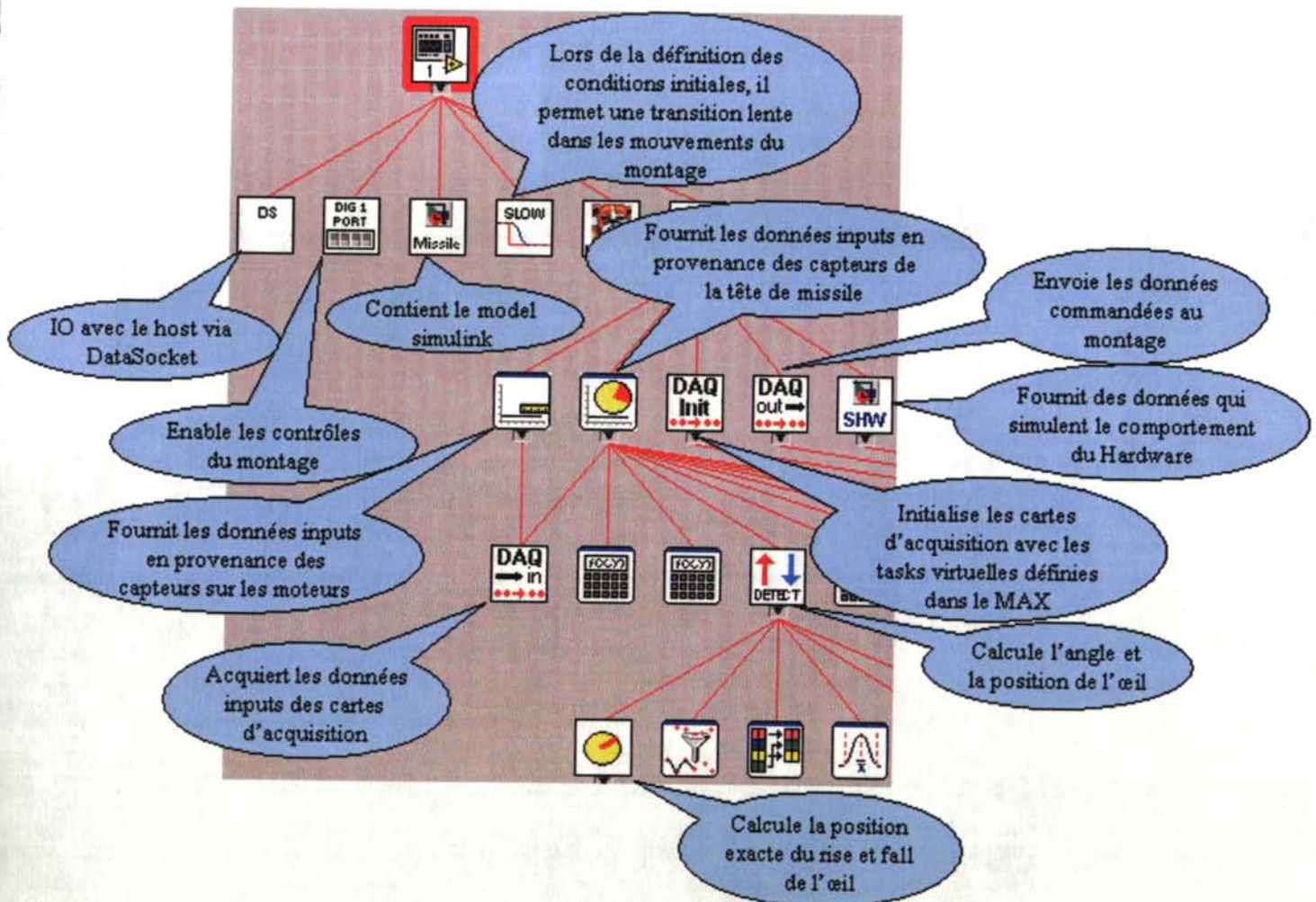


## 5. PXI

Le PXI contient l'application Labview qui inclut le modèle du missile. Cette application se trouve sous ...\\Target\\HIL.vi. Pour s'assurer d'un bon fonctionnement de l'ensemble du montage, l'ordre de démarrage devrait être fait comme suit : « dataSocket server», le PXI, le montage Hardware et enfin l'application Host.

Le lancement automatique de l'application Labview sur le PXI, peut se faire avec les opérations suivantes :

1. Open a new VI in Labview.
2. Select **Operate»Switch Execution Target** to target the Real-Time PXI controller on the network.
3. Then run the Application Builder by selecting **Tools»Build Application or Shared Library (DLL)**.
4. Click the **Source Files** tab and add your top level VI or VIs. Browse through the other settings.
5. When you are ready, click the **Build** button. Labview builds an executable named startup.exe and places it in the c:\ni-rt\startup directory on the controller.







Lors de la réalisation de cette application, certaines stratégies opérationnelles ont été mises de l'avant. Habituellement, un montage HIL requiert un « sample par loop » et souvent la régularité de la vitesse d'échantillonnage est plus ou moins importante. Ce montage, au contraire, requiert une régularité très précise dans la vitesse d'échantillonnage et nécessite plus d'un échantillon par boucle pour accroître la précision des angles de l'œil. Pour permettre une régularité dans la vitesse d'échantillonnage, le contrôle est donc laissé aux cartes d'acquisition qui sont plus précises qu'une application software. La variation du nombre d'échantillons par boucle est modifiable par le champ Résolution du front Panel du HIL.vi.

Une autre solution a été mise de l'avant pour palier les imperfections du hardware telles que le parallélisme des fenêtres de la tête de missile, la position [0,0] sensed vs commander, etc. À l'initialisation du mode « sensed », l'application compare les données commandées avec les données recueillies et il définit automatiquement le diamètre du laser beam. Par la suite, cette valeur est utilisée comme base de calcul dans les différentes opérations subséquentes.

## 6. SIMULINK

L'outil Simulink contient le modèle du missile. Le choix de la technologie Simulink pour la modélisation du missile permet des modifications majeures et des améliorations très facilement et rapidement. Sa fonction RTW permet de générer un code compilé performant pour le Real time de la cible. À cet effet, un RTW a été développé pour créer un code compilé propre à Labview du PXI.

Afin de pouvoir utiliser cette capacité d'interopérabilité voici quelques informations complémentaires.

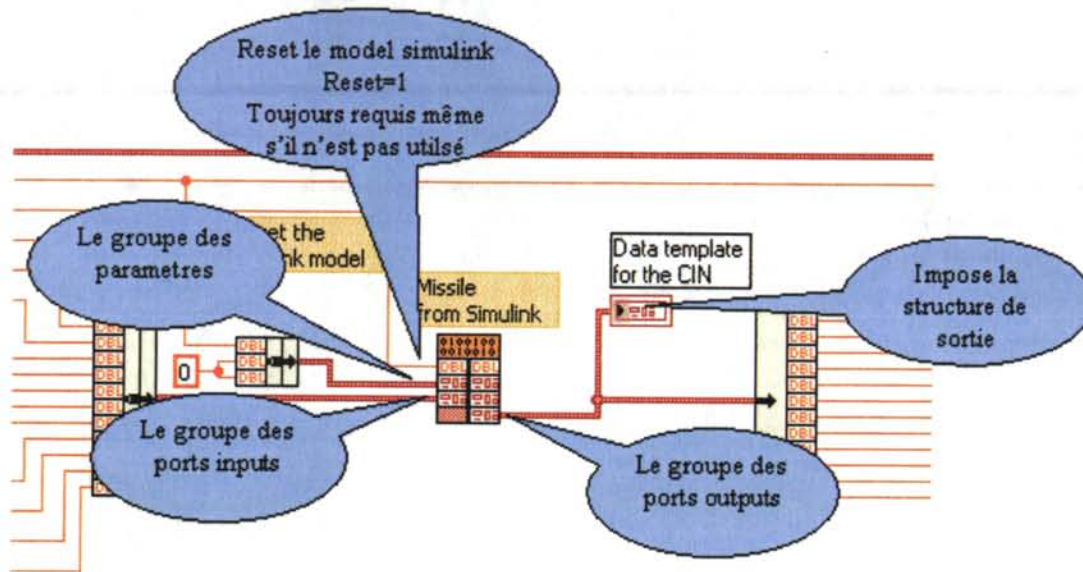
Dans un premier temps, pour générer un code pour le PXI, il est important d'avoir le répertoire TLCLabview dans les setPaths de Matlab. IMPORTANT, ce répertoire doit être en mode « write » car lors de sa première utilisation, le code initialise des paths dans les différents fichiers de ce répertoire. De plus, Labview doit être installé sur la station, plus précisément le répertoire cintools de Labview. L'utilisation de ce TLC est similaire aux autres TLC standards.

**Au niveau du modèle Simulink, certaines règles doivent être suivies. En voici quelques-unes :**

- ① Le modèle doit comporter au moins un « inline parameters » ;
- ① Avoir au moins un port input et output ;
- ① Dans la dialog box « Configuration Parameters », il est important de mettre le paramètre Stop time = 0 pour permettre une synchronisation parfaite du modèle Simulink avec Labview ;
- ① Dans la dialog box « Configuration Parameters », le choix du « RTW system target file : » doit être labrt.tlc pour générer le bon fichier .lsb ;
- ① On peut optionnellement ajouter un port pour indiquer la fin de la simulation. Ce dernier pourra alors être utilisé dans Labview ;
- ① Il existe un exemple dans le répertoire TLCLabview portant le nom de testTLCLabview.mdl



Dans Labview aussi, certaines règles doivent-êtré suivies. En voici quelques-unes :



- ⓪ Pour importer le fichier .lsb, utilisez le symbole « Code interface node » dans la palette « Advanced » ;
- ⓪ Le nombre de pins in et out du symbole doit toujours être de la forme illustrée ci-haut quel que soit le nombre de paramètres, inputs et outputs ;
- ⓪ La première pin input est le reset du modèle Simulink. Elle n'est pas visible dans le modèle Simulink ;
- ⓪ La deuxième pin input est toujours le groupe de paramètres. La correspondance entre la structure entrant sur la pin de Labview et la liste des paramètres dans Simulink est séquentielle et indépendante du nom des paramètres ;
- ⓪ La troisième pin input est toujours le groupe de ports inputs. La correspondance entre la structure entrant sur la pin de Labview et la liste des ports inputs dans Simulink est séquentielle et indépendante du nom des ports ;
- ⓪ La quatrième pin input n'est jamais utilisée ;
- ⓪ Les 3 premières pins output ne sont jamais utilisées ;
- ⓪ La quatrième pin output est toujours le groupe de ports output et doit toujours exister. La correspondance entre la structure entrant sur la pin de Labview et la liste des ports outputs dans Simulink est séquentielle et indépendante du nom des ports. Dans le cas ci-haut présentée une astuce a été utilisée pour permettre d'identifier les signaux sortants de la pin en la raccordant sur une structure de données qui fournit des « labels » aux signaux.



## 7. **HARDWARE**

La principale fonction du Hardware, est de réaliser le comportement mécanique du missile à partir des données commandées et de recueillir les données mesurées. Lors de la mise en route des cartes d'acquisition, une solution a été trouvée pour s'assurer du bon fonctionnement du hardware avec le software. Il s'agit d'une routine d'initialisation du Hardware pour synchroniser la position du hardware avec les données commandées.



## 8. CONCLUSION

Ce projet à permis de démontrer la possibilité de réaliser une interopérabilité entre différents outils tel que Simulink et Labview dans un contexte du real time. Cette approche donne des résultats très intéressants sur le PXI. Des tests de performance ont été réalisés avec un modèle de missile et la vitesse de loop a été vérifiée jusqu'à 1000 Hz. Les données obtenues avec Labview en comparaison de ceux du même modèle dans Simulink sont identiques à une précision de  $10^{-6}$ .

Certains travaux pourraient être réalisés pour augmenter la robustesse de l'ensemble du processus et rendre le système plus intuitif pour l'utilisateur. D'autres fonctionnalités pourraient aussi être ajoutées telle que la possibilité de faire varier le timestep du modèle Simulink avec Labview. Ces travaux pourront être considérés plus tard.



# 523290  
CA025611