

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

ÉCOLE NATIONALE POLYTECHNIQUE

DÉPARTEMENT D'ÉLECTRONIQUE

PROJET DE FIN D'ÉTUDES

EN VUE DE L'OBTENTION DU DIPLOME D'INGENIEUR D'ETAT EN
ELECTRONIQUE

Thème

Etude de la carte dSPACE DS1104

Dirigé par :

Pr D. BERKANI

Etudié par :

KHELIFI Mustapha
SEKOUR Ahmed

Promotion : Juin 2006

Nous remercions le Dieu le tout
puissant

DEDICACES

A mes très chers parents...

Témoignage d'affection et de grande reconnaissance pour tous leurs sacrifices,

Que Dieu les garde pour moi.

A toute ma famille,

Mes sœurs Ouiza, Lydia, et Dyhia,

Mes frères Farid, Sofiane, Moahmed, et Coceila,

A mes deux grands-mères

A mes tantes, oncles et leurs familles,

A mes cousines, cousins et leurs familles,

A mes frères et amis Mehand, Bilal, Malek, Karim et Khelef,

A mon binôme Mustapha avec qui j'ai partagé amitié et travail.

A tous mes amis,

Avec l'expression de tous mes sentiments,

Je dédie ce modeste travail.

Ahmed

DEDICACES

Je dédie ce mémoire à mes très chers parents qui ont cru en moi et qui ne n'ont jamais cessé de m'encourager pendant toute ma scolarisation.

Je le dédie à mes frères et mes soeurs qui m'ont beaucoup aidé.

Sans oublier mon binôme Ahmed avec qui j'ai partagé amitié et travail.

Mes dédicaces et mes respects vont aussi à mes frères et amis : Nabil, Nabil, Khalef, Ouahab, Adel, Rabah, Yahia, Ali, et à tous mes amis de SEG, de l'Ecole et de Bouraoui Amar.

Mustapha

REMERCIEMENTS

Ce travail a été réalisé au sein du laboratoire de communication et signal du département d'électronique de l'École Nationale Polytechnique d'Alger sous la direction de Monsieur D. BERKANI, Professeur à l'ENP.

Nous le remercions en sa qualité d'encadreur, pour sa gentillesse, sa bonne humeur, ses précieux conseils et pour la confiance qu'il nous a témoignée en dirigeant ce travail.

Nous exprimons notre profonde gratitude à Monsieur R. ZERGUI et à Mademoiselle F. MERAZKA qui nous ont fait l'honneur d'examiner ce modeste travail.

Nous tenons à exprimer nos vifs remerciements à Madame G. Voguet qui a revu ce document sur le plan linguistique et sur sa mise en forme.

Nous remercions tous ceux, qui de près ou de loin, nous ont soutenus et aidés dans la réalisation de ce travail.

ملخص

هذا العمل يهتم بدراسة اللوحة الإلكترونية dSPACE DS1104 الموجهة لتطوير أنظمة التحكم و المراقبة. و يحتوي على وصف مختصر للخصائص التقنية لهذه اللوحة, و دراسة لطريقة عمل و استعمال برنامجها ControlDesk, وكذلك البرنامج Real-Time Interface الذي يمثل الرابط بين ControlDesk و Simulink, وهذا الأخير يستعمل لبرمجة اللوحة. و في النهاية, تطبيق لنظام تحكم في صمام مع مراقب PID.

الكلمات المفاتيح : Simulink, Real-Time Interface, ControlDesk, dSPACE DS1104

Résumé

Ce travail s'intéresse à l'étude de la carte dSPACE DS1104 qui dédiée au développement de commandes et de contrôleurs. Il contient une brève description des caractéristiques techniques de la carte, et une étude de fonctionnement et d'utilisation de son logiciel ControlDesk, ainsi que l'interface en temps réel RTI qui représente le lien entre ControlDesk et Simulink, ce dernier est le logiciel de la programmation de la carte. À la fin, une implémentation d'une simulation de commande d'une valve avec un contrôleur PID.

Mots Clés : dSPACE DS1104, ControlDesk, Real-Time Interface et Simulink.

Abstract

This work is interested in the study of the board dSPACE DS1104 which used to the development of commands and controllers. It contains a short description of technical features of the board, and a study of functioning and use of its software ControlDesk, as well as the Real-Time Interface (RTI) which represents the link between ControlDesk and Simulink, this last is the software of programming of the board. At the end, an implementation of a simulation of command of valve with a controller PID.

Key Words: dSPACE DS1104, ControlDesk, Real-Time Interfaces and Simulink.

SOMMAIRE

Introduction générale	1
Chapitre I : dSPACE et ses produits	
Introduction	3
I.1.L'offre dSPACE	3
I.1.1.Cartes et systèmes.....	3
I.1.2.Logiciels	5
Conclusion	7
Chapitre II : Caractéristiques et mise en œuvre de DS1104	
Introduction.....	9
II.1.Composition de la carte DS1104.....	9
II.1.1.Processeur maître PPC (Master)	10
II.1.1.1.Dispositifs d'entrées/sorties du maître PPC	10
II.1.2.Le sous ensemble du processeur DSP	11
II.1.2.1.Dispositifs d'entrée sortie de l'esclave DSP	12
II.1.3.Contrôleur d'interruption	12
II.1.3.1.Interruptions d'utilisateur	12
II.1.3.2.Interruptions d'encodeur	13
II.1.4.Mémoires.....	13
II.1.5.Temporisateurs	14
II.2.Mise en œuvre de la carte DS1104.....	14
II.2.1.Les conditions de base nécessaire avant l'installation du logiciel	14
II.2.2.Installation du logiciel.....	15
Chapitre III : Présentation du logiciel ControlDesk	
Introduction.....	17
III.1.Principes de fonctionnement de ControlDesk.....	18
III.1.1.Les éléments de base de ControlDesk	19
III.1.1.1.La barre des menus (Menu Bar).....	19
III.1.1.2.Barre d'outils (Toolbar)	20
III.1.1.3.Navigateur (Navigator)	20
III.1.1.4.Zone de travail (working area).....	21
III.1.1.5.Fenêtre d'outils (Tool Window)	21
III.1.1.6.Sélecteur d'instrument (instrument selector)	22
III.1.2.Manipulation des fichiers.....	22
III.1.2.1.Utilisation du sélecteur de fichier (File Selector)	23
III.1.3.Messages	24
III.2.La gestion des expérimentations (Managing Experiment)	25
III.2.1.L'utilisation d'Experiment Manager.....	25
III.2.1.1.Exemple de manipulation d'expérimentation	27
III.3.Gestion des plateformes et manipulation des applications	29
III.3.1.Les éléments de bases du gestionnaire de plateforme	29
III.3.2.Enregistrement des plateformes.....	30
III.3.2.1.Enregistrement d'un système à processeur unique (single processor).....	30

III.3.3.Création des fichiers SDF	32
III.3.3.1.Editeur des fichiers SDF	32
III.3.3.2.Création automatique des fichiers SDF	32
III.3.4.Manipulation des applications en temps réel	33
III.3.5.Manipulation des applications sur le processeur esclave DSP	33
III.4.Construction des panneaux d'instruments	34
III.4.1.Outil d'instrumentation	34
III.4.1.1.Navigateur d'instrumentation	35
III.4.1.2.Panneaux d'instrument.....	36
III.4.1.3.Sélecteur d'instruments.....	36
III.4.1.4.Exemple de création d'un nouveau groupe.....	39
III.4.2.Création du panneau d'instrumentation	42
III.4.2.1.Création d'un nouveau panneau d'instrumentation	42
III.4.2.2.Modification des propriétés d'instruments	44
III.4.2.3.Layout de capture.....	46
III.4.3.Manipulation des variables	47
III.4.3.1.Création des fichiers SDF et MAP.....	47
III.4.3.2.Choix de la plateforme de simulation	47
III.4.3.3.Variable browser	48
III.4.4.Connexion de données	49
III.4.4.1.Types de connexion de données	49
III.4.4.2.Connexion de données entre les variables et les instruments	49
III.4.4.3.Connexion de données entre deux instruments.....	50
III.4.4.4.Connexion de l'instrument TableEditor	51
III.4.5.Connexion de données aux instruments d'acquisition de données	52
III.4.5.1.Connexion de données à l'instrument Plotter	52
III.5.Utilisation des panneaux d'instruments	53
III.5.1.Mode Opération (Operation Mode)	53
III.5.1.1.Exemple d'un modèle dans le mode animation	54
III.5.2.Capture des données	56
III.5.2.1.Comment capturer des données	58
III.5.2.2.La page de détails.....	60
III.5.2.3.La page acquisition (Acquisition).....	60
III.5.3.Utilisation de données de Référence	61
III.5.3.1.Génération et sauvegarde des données de référence	62
III.5.3.2.Affichage des données de référence	63
III.6.Expérimentation sur la plateforme de Simulink	64
III.6.1.Conditions de système pour des simulations de Simulink.....	64
III.6.2.Manipulation des simulations de Simulink dans ControlDesk	65
Conclusion	67
 Chapitre IV : L'interface en temps réel RTI	
Introduction.....	69
IV.1.Définition de RTI.....	69
IV.2.Les étapes de développement	69
IV.2.1.Contrôle de la conception et la simulation du modèle de Simulink	69
IV.2.2.Prototypage rapide de contrôle (RCP rapid control prototyping).....	69
IV.2.3.Hardware In the Loop HIL	70
IV.2.4.Génération automatique du code C.....	70

IV.3.Travailler avec RTI et RTI-mp	70
IV.3.1.Activation d'une carte spécifique	70
IV.3.2.L'accès aux blocs de RTI	71
IV.3.3.Les valeurs par défaut des paramètres de simulation	74
IV.4.Les interfaces d'entrée/sortie I/O.....	76
IV.4.1.Les conventions de la nomenclature.....	76
IV.4.2.L'ajout des blocs I/O	77
IV.4.3.Types de données.....	78
IV.5.Création d'un nouveau bloc :	78
IV.5.1.Implémentation de S-Function :	78
IV.5.1.1.Les Bases de S-Functions	78
IV.5.1.2.Composants des S-fonctions.....	78
IV.5.1.3.Le modèle de S-Function.....	79
IV.5.1.4.La construction manuelle de S-Function en code C	79
IV.5.1.5.La création du fichier MEX DLL	80
IV.5.2.Implémentation de User-Code.....	81
IV.5.2.1.Comment utiliser User-Code dans les modèles de Simulink :	81
IV.5.2.2.Pour inclure User-Code dans un model de Simulink.....	82
IV.5.2.3.Pour obtenir un modèle de User-Code.....	82
IV.5.2.4.Construction manuelle de User-Code.....	82
IV.5.2.5.Comparaison entre S-Functions et User-Code	83
IV.6.Construction et téléchargement d'un modèle	84
IV.6.1.Les principes pour la construction et le téléchargement.....	84
IV.6.1.1.Les fichiers.....	84
IV.6.1.2.Le processus de construction.....	84
IV.6.2.La spécification des options du processus de construction :	85
IV.6.3.Construction et téléchargement	86
IV.6.4.Reconstruction du modèle :	87
IV.7.La simulation externe	87
IV.7.1.Exécution d'une simulation externe	87
Conclusion	90

Chapitre V Simulation de commande d'une valve

Introduction :	92
V.1.Description de la valve.....	92
V.2.Description de la commande :	92
V.2.1.Principes de fonctionnement du modèle Simulink :	93
V.2.1.1.Génération de référence :	93
V.2.1.2.La commande PID :	94
V.2.1.3.Simplified throttle:.....	95
V.2.1.4.Termitator :	96
V.2.1.5.RefPos [°]:.....	96
V.2.1.6.Throttle Valve Position :	96
V.3.La simulation :	96
V.3.1.La simulation avec Simulink :	96
V.3.1.1.Les résultats :	96
V.3.2.Simulation avec la carte DS1104.....	97
V.3.2.1.Création d'une nouvelle expérimentation.....	97
V.3.2.2.Les éléments de l'expérimentation	98

V.3.3.Les résultats.....	99
V.4.Comparaison entre Simulink et ControlDesk :	99
Conclusion :	100
Conclusion générale et Perspectives.....	101
Bibliographie	102

LISTE DES FIGURES

Figure I.1	Carte DS1104	4
Figure I.2	Carte DS1005	4
Figure I.3	MicroAutobox	4
Figure I.4	ControlDesk	5
Figure I.5	Real-Time Interface	5
Figure I.6	MotionDesk	6
Figure I.7	CalDesk	6
Figure II.1	L'architecture de la carte DS1104	9
Figure II.2	Signal d'interruption avec un front descendant	13
Figure II.3	Signal d'interruption avec un front montant	13
Figure III.1	Fenêtre principale de ControlDesk	19
Figure III.2	Barre des menus	19
Figure III.3	Navigateur de ControlDesk	20
Figure III.4	Fenêtre d'outils	21
Figure III.5	Sélecteur de fichier	24
Figure III.6	Création d'une nouvelle expérimentation	25
Figure III.7	Fenêtre d'ajout des fichiers	27
Figure III.8	Ajout des fichiers externes	28
Figure III.9	Les fichiers ajoutés à l'expérimentation	28
Figure III.10	Barre d'outil de navigateur de plateforme	29
Figure III.11	Gestionnaire de plateforme	30
Figure III.12	Fenêtre d'enregistrement de la carte	31
Figure III.13	propriétés de la carte	31
Figure III.14	Editeur des fichiers SDF	32
Figure III.15	Chargement d'un fichier dans la carte	33
Figure III.16	Outil d'instrumentation	34
Figure III.17	Navigateur d'instrumentation et son menu contexte	35
Figure III.18	Barre d'outil d'instrumentation	35
Figure III.19	Barre de layouting	36
Figure III.20	Layout et son menu contexte	36
Figure III.21	Groupe d'acquisition de données et ses éléments	40
Figure III.22	Sélecteur d'instrument	40
Figure III.23	Icônes des boutons	41
Figure III.24	Création d'un nouveau groupe d'instrument	41
Figure III.25	Ajout de plotter au nouveau groupe	42
Figure III.26	Propriétés du Layout	43
Figure III.27	Propriétés de l'instrument MultiStateLED	44
Figure III.28	Propriétés du Plotter	46
Figure III.29	Propriétés du Templates	46
Figure III.30	Choix de plateforme	48
Figure III.31	Variable browser	48
Figure III.32	Connexion de données	50
Figure III.33	Menu contexte	50
Figure III.34	Plotter	52

Figure III.35	plotter avec deux variables	53
Figure III.36	Panneau d'instrument avant la simulation.....	55
Figure III.37	Panneau d'instrument après la simulation.....	55
Figure III.38	Modification de la valeur de la variable.....	56
Figure III.39	Fenêtre d'arrangements de la capture.....	57
Figure III.40	Page de capture de variables.....	58
Figure III.41	Page de capture.....	59
Figure III.42	Page de déclenchement (Trigger).....	60
Figure III.43	Page de détails.....	60
Figure III.44	Page d'acquisition	61
Figure III.45	Signal de référence avec la variable correspondante.....	64
Figure III.46	Nouveau modèle Simulink	65
Figure III.47	Menu contexte de la plateforme Simulink.....	66
Figure IV.1	Sélection d'une carte spécifique	71
Figure IV.2	MATLAB Launch Pad.....	72
Figure IV.3	Bibliothèque de RTI 1104.....	73
Figure IV.4	La bibliothèque de dSPACE RTI1104 dans le Simulink	74
Figure IV.5	Simulink Simulation Preferences.....	75
Figure IV.6	Désactivation de l'option 'Block reduction'	76
Figure IV.7	Exemples des noms des blocs I/O.....	77
Figure IV.8	Les blocs Ground et Terminator.....	77
Figure IV.9	Le bouton Build.....	85
Figure IV.10	Les paramètres de simulation de Solutionneur	86
Figure IV.11	Les paramètres de Real-Time Workshop	88
Figure IV.12	Menu Simulation du modèle Simulink.....	88
Figure IV.13	Menu Tools du modèle Simulink.....	89
Figure IV.14	External Mode Control Panel.....	89
Figure IV.15	L'option Normal de menu de Simulation.....	90
Figure V.1	Une valve.....	92
Figure V.2	Modèle de Simulink de commande de la valve.....	93
Figure V.3	Générateur de référence.....	93
Figure V.4	Le sous-système générateur de référence	93
Figure V.5	Le contrôleur PID.....	94
Figure V.6	Le sous-système de contrôleur PID	94
Figure V.7	Le bloc de la valve.....	95
Figure V.8	Le sous-système de la valve	95
Figure V.9	Terminator	96
Figure V.10	Graphe de la référence par Simulink	97
Figure V.11	Graphe de la position de la valve par Simulink.....	97
Figure V.12	Panneau d'instruments de la commande de la valve	98
Figure V.13	Graphe de la référence par ControlDesk	99
Figure V.14	Graphe de la position de la valve par ControlDesk	99

LISTE DES TABLEAUX

Tableau III.1	les types de fichiers et leurs outils	23
Tableau III.2	liste des instruments virtuels	38
Tableau III.3	liste des instruments d'acquisition de données	38
Tableau IV.1	Les valeurs par défaut de RTI.....	75
Tableau IV.2	Les différentes fonctions de User-Code.....	83
Tableau IV.3	Comparaison entre S-function et User-Code	84

Introduction générale

Introduction générale

Plusieurs compagnies des domaines aussi variés que l'automobile, l'aéronautique, la robotique, et l'énergie électrique utilisent les techniques de simulation et de prototypage afin de développer leurs produits plus rapidement et plus efficacement en évitant le processus coûteux du prototypage physique.

Les systèmes de dSPACE constituent une plate-forme matérielle et logicielle très répandue dans les domaines cités ci-dessus. Une caractéristique intéressante de ces systèmes est qu'ils permettent de développer un schéma de simulation ou de commande dans l'environnement Simulink, de compiler le schéma pour l'exécuter sur des cartes qui contiennent des processeurs DSP, et d'exécuter les schémas de commande en temps réel en interagissant avec un système physique.

Pour mener à terme ce travail, le plan que nous avons adopté se résume par les points présentés ci-dessous.

Au premier chapitre, on a présenté les différents produits matériels et logiciels de dSPACE.

Dans le deuxième chapitre on a donné un aperçu général des caractéristiques de la carte DS1104.

Le troisième chapitre est consacré à l'étude de logiciel ControlDesk.

Dans le quatrième chapitre, on a parlé de l'interface en temps réel (RTI) qui permet de faire le lien entre le Simulink et le ControlDesk.

Pour clore le présent travail une démonstration de commande d'une valve est faite afin d'illustrer l'utilisation de Simulink, RTI et ControlDesk.

Chapitre I

dSPACE et ses produits

Introduction

dSPACE est une société créée en 1988 à Paderborn en Allemagne. Elle produit et commercialise des solutions de développements, de tests et de mises au point pour systèmes de contrôle et systèmes embarqués.

Son but est de proposer des solutions combinant des composants matériels et logiciels, simples à utiliser par les ingénieurs.

L'entreprise fonde en 1991 près de Détroit, Michigan, sa filiale dSPACE Inc. qui couvre les Etats-Unis et le Canada. En 1999 et 2001, dSPACE ouvre deux centres d'ingénierie à Stuttgart et Munich afin d'accompagner ses clients dans l'utilisation de ses outils. Depuis novembre 2001, dSPACE a ouvert deux filiales européennes au Royaume- Uni et en France.

Ses solutions sont utilisées au Japon, en Australie, en Chine, en France, en Israël, à Taiwan, en Suède, au Royaume Uni, en République Tchèque et Slovaquie, en Pologne, aux Pays-Bas, aux Indes, aux Etats-Unis, en Corée ... etc.

I.1. L'offre dSPACE [2]

dSPACE propose des systèmes matériels et des logiciels :

I.1.1. Cartes et systèmes

Cette société développe des systèmes mono-cartes et des systèmes modulaires, ainsi qu'un système de développement embarquable sur véhicule.

- Les systèmes mono-cartes possèdent des processeurs rapides, et s'intègrent directement dans un PC via les ports ISA ou PCI.



Figure I.1 Carte DS1104

- Les systèmes modulaires offrent une plate-forme haute performance pour répondre à toutes les exigences. Ces systèmes peuvent s'intégrer dans un PC ou dans des boîtiers externes.



Figure I. 2 Carte DS1005

- Les systèmes de développement embarquables sur véhicule automobile. La MicroAutoBox combine les avantages d'un ordinateur automobile à ceux d'un système de prototypage.



Figure I. 3 MicroAutobox

I.1.2. Logiciels

Ce qui concerne la partie software, dSPACE propose six logiciels principaux :

- Les logiciels d'expérimentation, tel que ControlDesk qui permet la gestion, l'instrumentation et l'automatisation des expériences et applications qui tournent sur son matériel ou le pilotage des modèles Simulink.

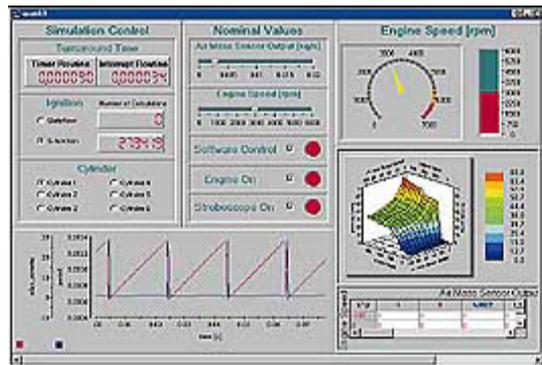


Figure I. 4 ControlDesk

- Les logiciels d'implémentation : Real-Time Interface (RTI) qui est une interface entre un système développé sous Simulink et la carte dSPACE. Il se charge de toutes les tâches d'implémentation.

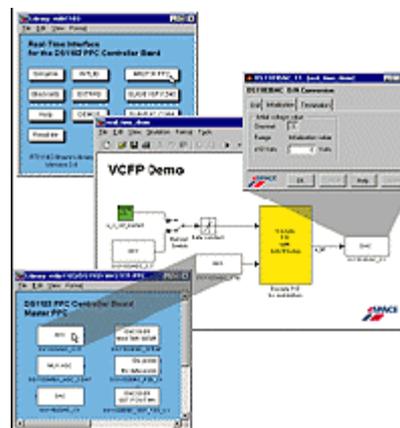


Figure I. 5 Real-Time Interface

- Les logiciels de génération de code C : TargetLink génèrent le code C à partir d'une description graphique faite sous Simulink.

- Le logiciel de visualisation et d'animation en temps réel : MotionDesk

Ce logiciel propose des fonctions de visualisation avancées tout en prenant en compte les contraintes de simulation en temps réel nécessaires.

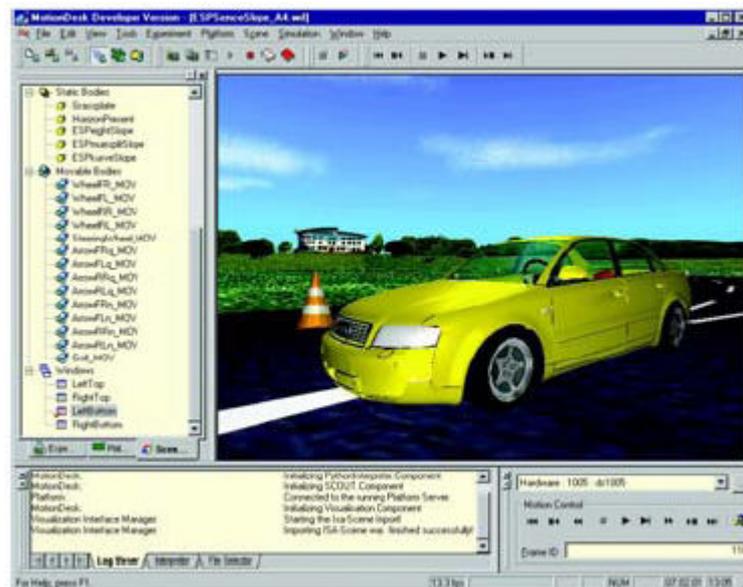


Figure I. 6 MotionDesk

- Le logiciel de test AutomationDesk : Ce nouvel outil permet de faciliter les tests automatisés.

- Le logiciel de calibration CalDesk : Ce logiciel permet de réaliser l'étape finale du développement de calculateur. Il permet d'acquérir des données depuis un calculateur ou des systèmes de mesures externes, et de modifier les paramètres du calculateur pendant les périodes de mise au point.

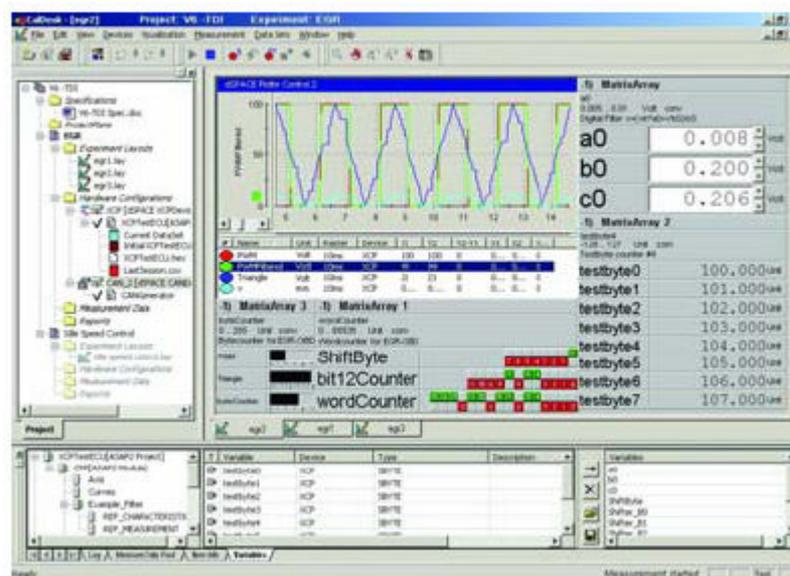


Figure I. 7 CalDesk

Conclusion

dSPACE offre aux développeurs et constructeurs de calculateurs électroniques et de systèmes embarqués la possibilité de réduire fortement leurs temps de développement et d'augmenter notablement la qualité de leurs produits, grâce à un ensemble de logiciels et matériels pouvant être utilisés pour du prototypage rapide et des simulateurs en temps réel.

Chapitre II

Caractéristiques et mise en
œuvre de DS1104

Introduction

La carte de contrôle DS1104 est un matériel qui améliore les performances du PC et qui assure un contrôle rapide et efficace. DS1104 est une preuve évidente que la puissance n'est pas nécessairement chère parce que cette carte est disponible à un prix raisonnable, ce qui en fait un système de développement parfait pour l'industrie et l'université.

II.1. Composition de la carte DS1104 [1] [11]

DS1104 est un système mono-carte contenant deux processeurs (maître, esclave), des contrôleurs d'interruption, des mémoires, des temporisateurs et des interfaces.

L'illustration suivante donne une vue générale de l'architecture du DS1104 :

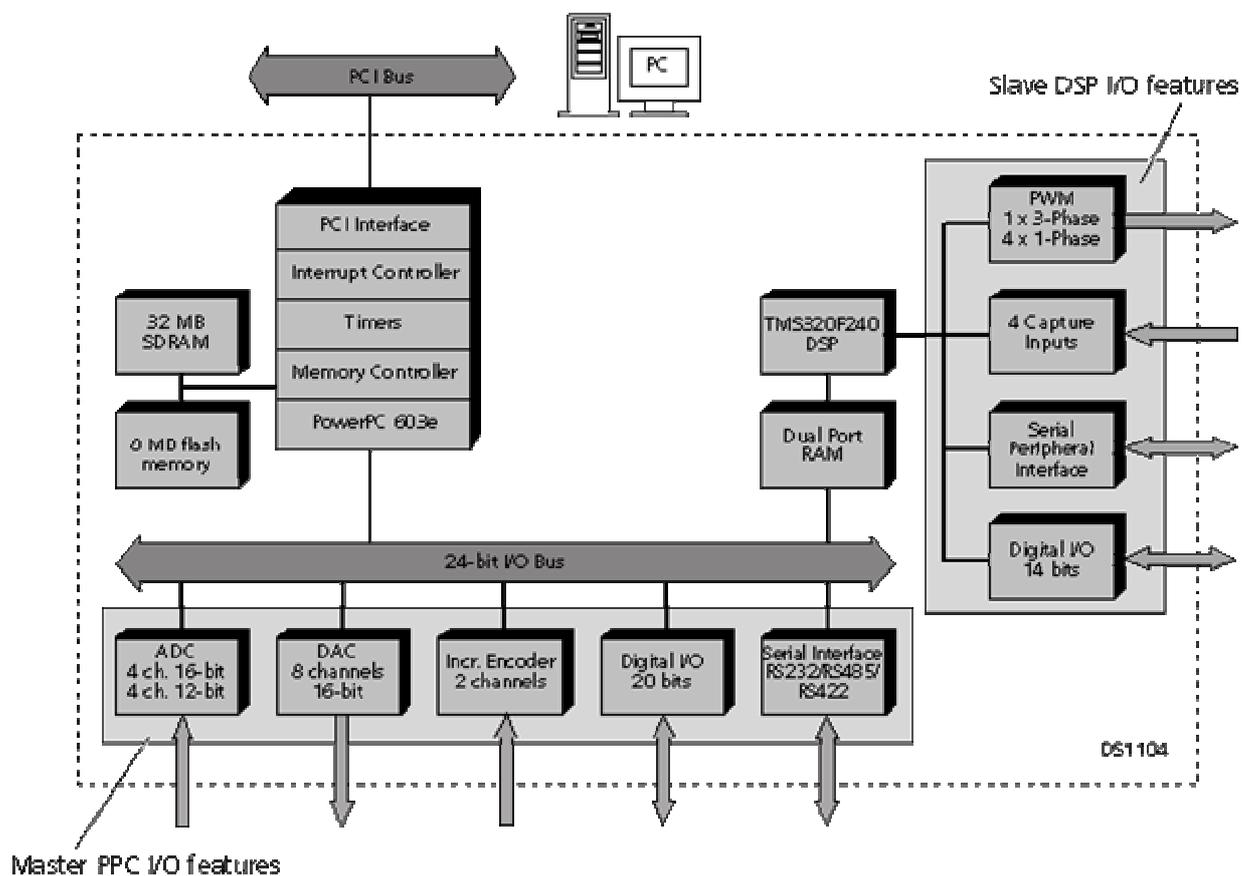


Figure II. 1 L'architecture de la carte DS1104

II.1.1. Processeur maître PPC (Master)

L'unité principale de traitement de DS1104, Motorola MPC8240, se compose de :

- un noyau PowerPC 603e (maître PPC)
 - Fonctionnant à 250 MHz (horloge d'unité centrale de traitement),
 - Contenant une mémoire cache de données L1 de 16-KByte,
 - Contenant une mémoire cache d'instructions L1 de 16-KByte ;
- un contrôleur d'interruption ;
- contrôleur synchrone de la mémoire DRAM ;
- plusieurs temporisateurs ;
- une interface PCI (5 V, 32 bits, 33 MHz) ; [13]

II.1.1.1. Dispositifs d'entrées/sorties du maître PPC

Le maître PPC contrôle les unités entrée/sortie suivantes.

1- Unité de ADC (*Analog Digital Converter*)

Le maître PPC contrôle une unité ADC comportant deux types différents de convertisseurs analogique/numérique:

- ❖ un convertisseur analogique/numérique (ADC1) multiplexé à quatre canaux. Les signaux d'entrée du convertisseur sont choisis par un multiplexeur d'entrée 4:1.
 - Résolution : 16 bits,
 - Tension d'entrée : ± 10 V,
 - Erreur : ± 5 mV,
 - Erreur de gain : $\pm 0.25\%$,
 - Rapport signal sur bruit (SNR) : > 80 dB.
- ❖ quatre convertisseurs analogique/numériques parallèles (ADC2... ADC5) avec un canal pour chacun.
 - Résolution : 12 bits,
 - Tension d'entrée : ± 10 V,
 - Erreur : ± 5 mV,
 - Erreur de gain : $\pm 0.5\%$,
 - Rapport signal sur bruit (SNR) : > 70 dB.

2- Unité de DAC (Digital Analog Converter)

Le processeur maître PPC contrôle un convertisseur numérique/analogique.

- 8 Canaux parallèles,
- Résolution : 16 bits,
- Tension d'entrée : ± 10 V,
- Erreur : ± 1 mV,
- Erreur de gain : $\pm 0.1\%$,
- Rapport signal sur bruit (SNR) : > 80 dB.

3- Unité d'entrée/sortie numérique

- 20 bits d'E/S,
- Direction sélectionnable pour chaque canal,
- Courant maximal de sortie : ± 5 mA,
- Erreur de gain : $\pm 0.1\%$,
- Technologie TTL.

4- Interface d'encodeur incrémental

Permet le développement des contrôleurs avancés des robots.

- Canaux d'entrée pour deux encodeurs,
- Support des signaux RS422,
- Compteur du 24-bit,
- Fréquence maximale 1,65 MHz,
- Ligne de terminaison pour les signaux d'entrées différentielles.

5- Interface série

DS1104 contient un récepteur et un émetteur asynchrone universel (Universal Asynchronous Receiver and Transmitter UART) pour accomplir une communication série asynchrone avec les dispositifs externes. UART peut être configuré comme émetteur récepteur RS232, RS422 ou RS485.

II.1.2. Le sous ensemble du processeur DSP

Le sous-ensemble de DSP se compose de :

- un DSP (Digital Signal Processor) : processeur TMS320F240 de Texas Instruments. Ses caractéristiques principales sont :
 - fonctionnant à 25 MHz,

- avec une mémoire de 4K x 16 utilisée pour la communication avec le maître PPC.

II.1.2.1. Dispositifs d'entrée sortie de l'esclave DSP

L'esclave DSP fournit les dispositifs d'entrée/sortie suivants :

- unité d'entrée/sortie numérique de synchronisation qui permet de générer et mesurer des signaux PWM (Pulse Width Modulated) et des signaux carrés.
- unité d'entrée/sortie numérique qui a les caractéristiques.
 - 14 bits d'E/S,
 - Direction sélectionnable pour chaque canal,
 - Courant maximal de sortie : ± 13 mV,
 - Technologie TTL.
- Interface Périphérique Série (SPI : Serial Peripheral Interface)

Le SPI est employé pour une communication synchrone d'une vitesse élevée avec les dispositifs connectés au DS1104, tel qu'un convertisseur analogique/numérique.

II.1.3. Contrôleur d'interruption

Rappelons qu'un processeur ne peut pas traiter plusieurs informations à la fois, mais grâce aux interruptions le processeur peut exécuter ou suspendre un programme donné selon la priorité des interruptions et tout cela est géré par un contrôleur d'interruption. [13]

Les interruptions qui sont disponibles sur la carte sont :

II.1.3.1. Interruptions d'utilisateur

Le MPC8240 du DS1104 fournit quatre interruptions dont l'utilisateur peut se servir comme sources de déclenchement dans une application en temps réel. Ces interruptions doivent être connectées extérieurement au DS1104. Elles sont déclenchées à chaque front descendant d'un signal externe.

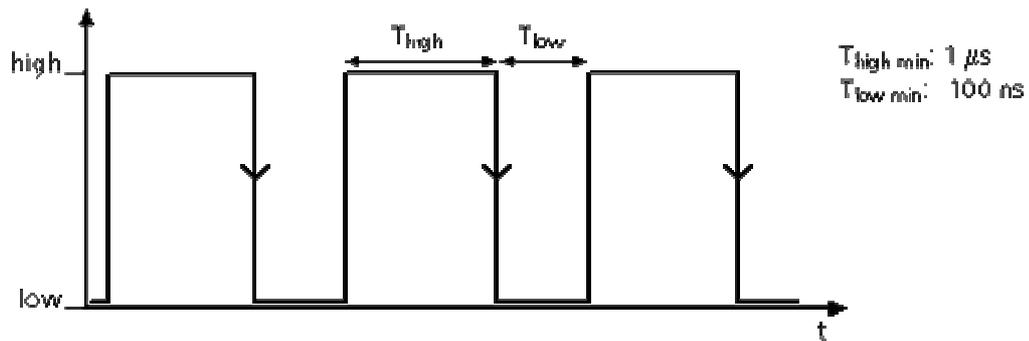


Figure II. 2 Signal d'interruption avec un front descendant

Pour permettre au contrôleur d'interruption d'identifier les interruptions d'utilisateur, l'entrée doit être maintenue au niveau haut pendant au moins $1\ \mu s$ (temps nécessaire pour que le matériel reconnaisse les interruptions). Une restriction causée par le logiciel doit être prise en compte $T_{haut} + T_{bas} > T_{min}$,

Avec T_{min} , est la période d'interruption nécessaire pour l'exécution d'une interruption.

II.1.3.2. Interruptions d'encodeur

Les interruptions déclenchent automatiquement une remise de la position d'encodeur. Le comportement de synchronisation est semblable aux des interruptions d'utilisateur, mais la caractéristique de signal est inversée pendant que l'interruption est déclenchée à chaque front montant du signal.

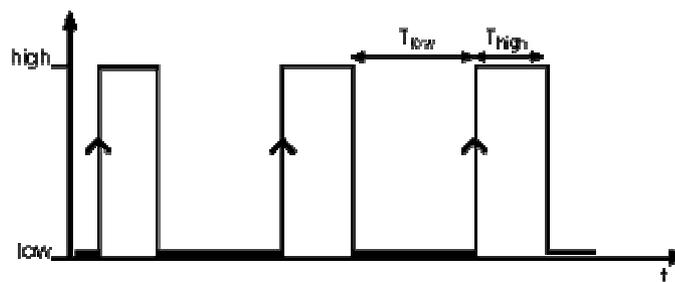


Figure II. 3 Signal d'interruption avec un front montant

II.1.4. Mémoires

La carte DS1104 est équipée de deux mémoires.

- Mémoire Globale :
 - SDRAM synchrone de 32 Mo pour les applications et les données,
 - Mémoire cache (L1),
- Mémoire flash : semi – conducteur, non volatile et réinscriptible
 - 8 Mo divisée en 4 blocs de 2 Mo,

- 6.5 Mo pour des applications spécifiques d'utilisateur,
- 1.5 Mo, réservé pour le programme de boot firmware,

II.1.5. Temporisateurs

Un temporisateur (Timer) est un circuit logique qui permet d'effectuer du comptage de temps, pour la génération de signaux.

La carte DS1104 est équipée de 6 temporisateurs. Ils sont pilotés par l'horloge.

II.2. Mise en œuvre de la carte DS1104 [8] [10]

Avant de commencer l'installation logicielle et matérielle de dSPACE, on doit vérifier si le système répond aux exigences de base.

Si le système est conforme, on peut travailler avec

- RTI et RTI-MP
- ControlDesk Standard et ControlDesk Test Automation

II.2.1. Les conditions de base nécessaire avant l'installation du logiciel

Le processeur Il faut utiliser au moins un Pentium II ou son équivalent (450 MHz) et un adaptateur graphique compatible VGA (Video graphic array).

La mémoire La mémoire RAM exigée doit être au moins 128 MB (256 MB ou plus est fortement recommandée).

L'espace requis sur le disque dur doit être au moins 200 MB.

Système d'exploitation Windows 98, Windows Millenium, Windows NT 4.0 ou Windows 2000, Windows XP n'est pas supporté.

Si l'on travaille avec Windows NT 4.0 ou Windows 2000, il est fortement recommandé d'installer le service pack le plus récent.

Il est déconseillé de travailler sous Windows NT 4.0 avec un service pack 4.

MATLAB la carte exige l'installation de MATLAB R 12.1 (version 6.1) et Simulink 4.0.

Le compilateur C Microtec power PC. L'installation de ce logiciel est nécessaire avant l'installation de dSPACE.

II.2.2. Installation du logiciel

Afin de garantir une meilleure installation on doit suivre les consignes dans l'ordre :

1. insérer le CD-dSPACE dans le lecteur CD,
2. insérer la clé-disque dans le lecteur disquette,
3. ouvrir le CD et exécuter CDsetup,
4. fermer tous les autres programmes avant de continuer l'installation,
5. suivre les instructions données par le programme d'installation,
6. une fois l'installation est terminée, retirer la disquette et le CD-dSPACE,
7. redémarrer le système d'exploitation,
8. sous Windows NT 4.0 ou Windows 2000, il faut entrer en tant qu'administrateur pour compléter l'installation.

Les deux dernières étapes sont obligatoires parce que.

- Certains fichiers DLL installés par dSPACE ne seront enregistrés qu'après le redémarrage du système,
- L'installation et l'enregistrement de dSPACE échoueront si l'on n'accède pas en tant qu'administrateur.

Remarque

Sous Windows NT 4.0 et Windows 2000 le dialogue d'ouverture n'apparaît pas au démarrage (autologon) ce qui n'offre pas la possibilité d'entrer en tant qu'administrateur.

Pour résoudre ce problème on procède comme suit :

- en redémarrant le PC, appuyer sur la touche **SHIFT** et le tenir enfoncée jusqu'à l'apparition du dialogue d'ouverture,
 - supprimer le processus d'autologon,
 - Choisir l'utilisateur qui a tous les droits (Administrateur).
9. Avant chaque démarrage de ControlDesk, brancher la clé matérielle (dongle) dans le port parallèle du PC, afin d'activer la licence.

Conclusion

DS1104 est une carte rassemble deux processeur puissant pour effectuer le traitement en temps réel, ainsi qu'un ensemble des interfaces entrées/sorties pour réaliser des connexions par l'intermédiaire de panneau connexion.

Chapitre III

Présentation du logiciel

ControlDesk

Introduction

Le logiciel ControlDesk est une interface graphique qui permet la gestion et l'administration de la carte DS1104. Il fournit en plus des fonctions d'enregistrement matériel de la gestion des applications, des fonctions de contrôle, de surveillance et d'automatisation des expérimentations.

ControlDesk est divisé en trois parties distinctes, chacune d'elles assure une fonction bien définie.

1. **ControlDesk standard** est la base de chaque expérimentation, il offre une variété d'outils :

- *Les composants de base*

- **Experiment Manager** est un outil qui assure la gestion des données pour commander et contrôler toutes les données appropriées à l'expérience. Le contenu de l'expérience peut être défini par l'utilisateur.
- **Platform Manager** fournit des fonctions pour manipuler les plateformes installées dans le PC (DS1104 et Simulink) et leurs applications. Il fournit également une interface pour manipuler les simulations de Simulink.

Le **platform Navigator** est une interface graphique du **platform Manager**.

- *L'instrumentation*

ControlDesk offre une variété d'instruments virtuels pour construire et configurer les panneaux d'instruments selon les besoins.

Le contrôleur sonore, par exemple génère divers sons (bruits des moteurs, ...etc.) afin de faire des simulations proche du réel.

- *Editeur de paramètre (Parameter Editor)*

ControlDesk fournit l'éditeur de paramètres qui permet la sauvegarde, le chargement et la modification des paramètres et aussi leur manipulation (définir les valeurs et les unités de paramètre, importer et exporter les paramètres, et échanger les paramètres entre Simulink et les applications en temps réel ...etc.).

- *Editeur de code source (Source Code Editor)*

C'est un éditeur optimisé qui permet d'écrire des programmes en langage C ou en langage Python

- *Automatisation (Automation)* [4]

Traitement et exécution automatisés des expérimentations.

- *Python interpreter* est l'outil central d'automatisation, qui permet d'écrire les commandes python et d'exécuter les scripts python
- *Macro Recorder* est un logiciel qui permet d'enregistrer et de relire (replay) les expérimentations et de générer le programme (script) écrit dans le langage python.

2. **ControlDesk Multiprocessor Extension** c'est une extension pour les systèmes multiprocesseurs.
3. **ControlDesk Test Automation** c'est une extension de ControlDesk Standard pour des dispositifs de tests d'automatisation avancés.

III.1 Principes de fonctionnement de ControlDesk [9]

L'installation de dSPACE sur le PC va ajouter automatiquement *ControlDesk* dans le menu « *démarrer* » et elle va créer aussi une icône sur le bureau une fois l'installation terminée avec succès.

Ouverture et fermeture de ControlDesk

On a deux possibilités pour lancer ControlDesk

- dans le menu *démarrer*, *tous les programmes*, *dSPACE Tools*, *ControlDesk*,
- dans le bureau en exécutant directement son icône.

Pour quitter *ControlDesk*, aller dans la barre des menus qui se trouve au dessus de la fenêtre principale de *ControlDesk* (figure 1) et sélectionner *file* puis cliquer sur *exit*.

III.1.1 Les éléments de base de ControlDesk

En lançant ControlDesk, une fenêtre s’ouvre. Les éléments y figurent.

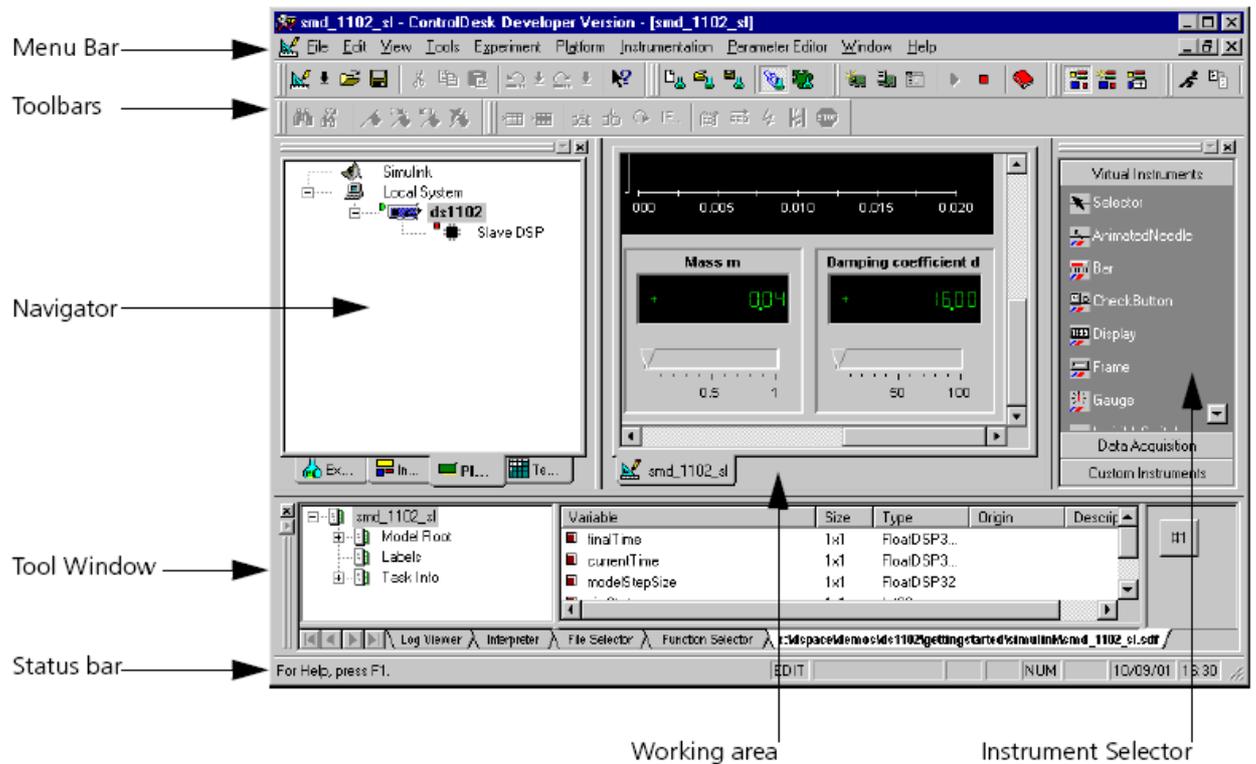


Figure III. 1 Fenêtre principale de ControlDesk

III.1.1.1 La barre des menus (Menu Bar)

Cette barre permet un accès aux différentes fonctions de ControlDesk. On y trouve les menus : *File, Edit, View, Tools, Experiment, Platform, Instrumentation, Window, et Help.* Chacun de ces éléments contient un ensemble de commandes auxiliaires.

La figure 2 montre les différentes fonctions qu’on peut trouver dans le menu « Edit ».

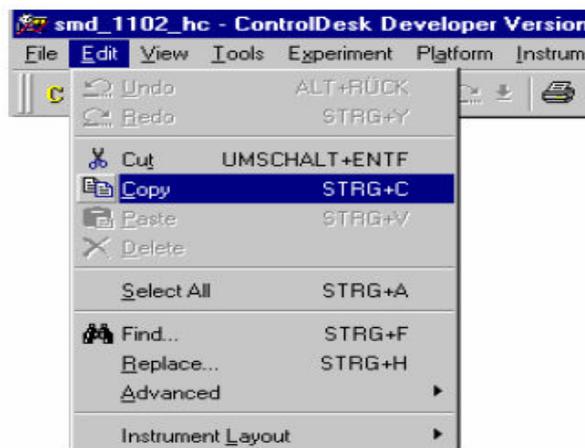


Figure III. 2 Barre des menus

III.1.1.2 Barre d'outils (Toolbar)

La barre d'outils contient des boutons qui assurent un accès rapide aux commandes et aux fonctions les plus utilisées.

Parmi ces boutons citons **Open, Save, Cut, Copy, ...etc**

III.1.1.3 Navigateur (Navigator)

Le navigateur de ControlDesk présente quatre fenêtres de navigation (expérimentation, instrumentation, plateforme, et test d'automatisation). Ce navigateur a pour rôle de structurer les fonctions, gérer les plateformes, manipuler les fichiers, construire les panneaux d'instruments, et créer des tâches automatiques.



Figure III. 3 Navigateur de ControlDesk

- ❖ **Le navigateur d'expérimentation (Experiment Navigator)** affiche tous les fichiers (par exemple, panneau d'instruments, ensemble de paramètres, données de référence) appartenant à l'expérimentation ouverte, et fournit des fonctions pour manipuler les expérimentations.
- ❖ **Le navigateur d'instrumentation (Instrumentation Navigator)** permet de construire les panneaux d'instruments pour commander et contrôler les variables du modèle, et d'afficher les résultats correspondants.
- ❖ **Le navigateur de plateforme (Platform Navigator)** affiche toute les plateformes enregistrées dans le système et fournit des fonctions pour manipuler les différentes applications telles que charger, démarrer, et arrêter.

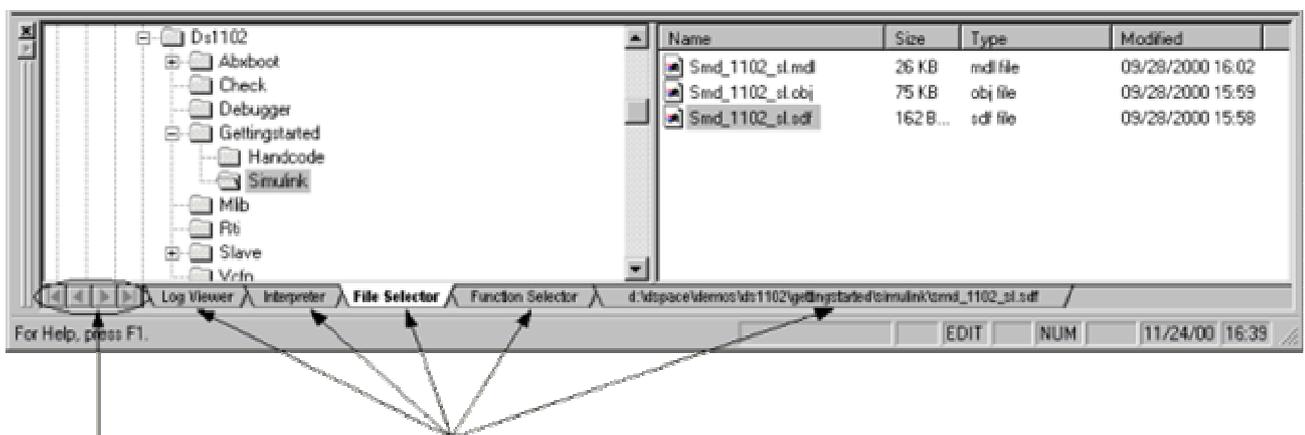
- ❖ Le navigateur de test d'Automatisation (**Test automation Navigator**) fournit toute les fonctions requises pour le test avancé et les tâches d'automatisation.

III.1.1.4 Zone de travail (working area)

Cette zone permet d'afficher et d'éditer les expérimentations ainsi que leurs simulation dans mode animation. Pour un nouveau panneau d'instruments cette zone sert à la création des layouts et à la connexion des données.

III.1.1.5 Fenêtre d'outils (Tool Window)

Cette fenêtre fournit des outils propres à chaque navigateur



Buttons to move
the displayed area

Tabs of the
Tool Window

Figure III. 4 Fenêtre d'outils

Log Viewer permet l'affichage des messages générés par ControlDesk ou par la carte.

Interpreter manipule les commandes et les scripts de python.

File Selector est utilisé pour sélectionner et charger des applications avec l'opération glisser et déposer.

Function selector permet d'afficher et d'appeler les fonctions disponibles dans les modules python.

Reference Data Manager contrôle les données capturées ou les données de référence.

L'utilisation de Glisser et déposer (drag and drop)

ControlDesk utilise la fonction **Glisser et déposer** pour ouvrir, déplacer, ou connecter des éléments.

Pour se faire on procède comme suit :

- cliquer sur un élément avec le bouton gauche de la souris, et le maintenir,

- déplacer l'élément dans son nouvel endroit,
- Relâcher le bouton et déposer l'élément.

III.1.1.6 Sélecteur d'instrument (instrument selector)

Il garantit l'accès à tous les instruments disponibles.

III.1.2 Manipulation des fichiers [7]

ControlDesk utilise des dispositifs spéciaux et des techniques standard pour le traitement des fichiers. Il manipule différents types de fichier afin de gérer les informations et les applications.

Ces fichiers sont enregistrés sous la forme "**nom_du_fichier.ext**". ".ext" représente l'extension, qui est un moyen de reconnaître le type de programme avec lequel ce fichier peut être ouvert.

Voici la liste des fichiers existant :

- **fichiers.CDX (ControlDesk experiment files)** Ils englobent tous les fichiers liés aux expérimentations.
- **fichiers.CON (connection files)** Ils décrivent la connexion entre les instruments et les variables.
- **fichiers.LAY (fichier layout)** Ils contiennent des informations (instrument, taille, position) sur les panneaux d'instruments, mais pas de connexion entre les variables et les instruments.
- **fichiers.MAT (matlab binary file)** Ils sont utilisés pour la sauvegarde des données capturées, ou éditer les tables de données.
- **fichiers.MDL** Ce sont des fichiers ASCII structurés qui contiennent les mots clés et les valeurs des paramètres qui décrivent le modèle Simulink.
- **fichiers.OBJ** Ce sont des fichiers exécutables pour les processeurs en temps réel (sur les cartes DS1003, DS1102) et les DSP esclaves (sur les cartes DS1103, DS1102, et DS2201).
- **fichiers.AXP/PPC** les fichiers AXP sont utilisés par la carte DS1004, et les fichiers PPC sont utilisés par les cartes DS1103, DS1005, et DS1401.
- **fichiers.PAR (parameter files)** Ils contiennent la description, les types, et les valeurs des paramètres.

- **fichiers.SDF (System description files)** Ils décrivent les fichiers à charger individuellement dans la plateforme Simulation. Ils sont générés automatiquement lorsqu'on construit des fichiers TRC avec RTI, ou Simulink simulation.

Remarque : On peut utiliser l'éditeur de fichier SDF pour générer et éditer ces fichiers.

- **fichiers.TRC (Variable description files)** Ils fournissent des informations sur les variables, et leur groupement.

RTI génère tout les variables de Simulink dans le fichier < [sub] model >.trc tandis que pour les variables de simulation Simulink sont disponibles dans le fichier

< [sub] model >_offline.trc.

Lors de la création d'un nouveau fichier, on doit spécifier son type (extension).

Dans le menu *file – new* on n'a accès qu'à certains types de fichiers, qui sont : TXT (texte), PY (python script), C (C Code), H (Header), LAY (Layout), et SE (Stimulus Editor). Pour chaque fichier ControlDesk ouvre une nouvelle fenêtre dans la zone de travail, et invoque l'outil qui correspond au type fichier sélectionné. Le tableau suivant résume les outils associés aux différents fichiers.

Type de fichier	Créé par
CDX	Experiment Manager
CON	Instrumentation Navigator
PAR	Variable Browser
SDF	System Description File Editor and Simulink/Real-Time Interface
SE	Stimulus Editor
MDL	Simulink Interface
TRC, MAP, OBJ	Simulink (TRC) and Real-Time Interface
C, PY, TXT	Source Code Editor

Tableau III. 1 les types de fichiers et leurs outils

III.1.2.1 Utilisation du sélecteur de fichier (File Selector) [9]

Cet outil permet de choisir et de charger les applications (fichiers APL, AXP, DDS, M, MDL, OBJ, PPC, et SDF) avec glisser et déposer.

Pour afficher le sélecteur de fichier, on doit d'abord choisir une plateforme de navigation, ensuite si la fenêtre d'outil est choisie dans la barre de control le sélecteur de fichier s'ouvrira ainsi que le navigateur de plateforme.

Pour charger une application, on procède comme suit :

- choisir la plateforme de navigation,
- sélectionner la description du système, objet, ou modèle ('.mdl', ou '.sdf'), dans le sélecteur de fichier et glisser et déposer le fichier désiré dans la plateforme.

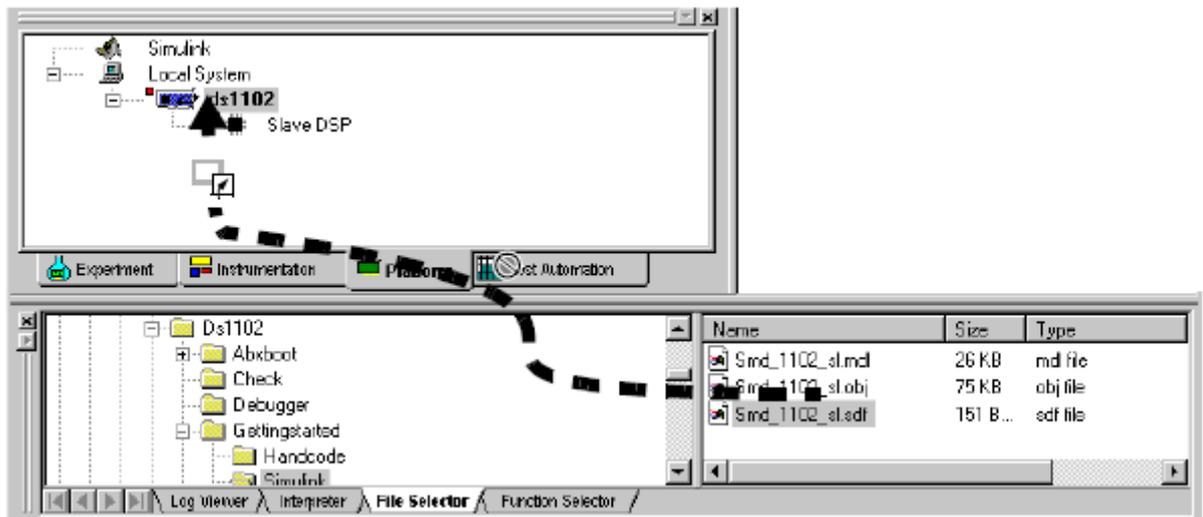


Figure III. 5 Sélecteur de fichier

III.1.3 Messages [7]

Les messages générés par ControlDesk ou la plateforme en temps réel sont :

- ❖ affichés sur l'écran comme boîtes de messages,
- ❖ affichés dans Log Viewer (dans la fenêtre d'outil (*tool window*)),
- ❖ stockés dans le fichier *dspace.log* (pour accéder à ce fichier, aller dans la barre des menus et choisir *View – Log File*).

Remarque :

Si le message est généré par la carte en temps réel, il commence par *Real – Time Processor* et contient les informations suivantes :

Current Number of Message | Board Name | Submodule | Message Text | Error Number

On distingue trois types de messages ERROR, WARNING, INFO.

III.2 La gestion des expérimentations (Managing Experiment) [1]

Experiment Manager est un outil de **ControlDesk** qui gère les fichiers appartenant à l'expérimentation, et fournit les commandes nécessaires à leur manipulation.

III.2.1 L'utilisation d'Experiment Manager

Travailler avec cet outil exige de respecter les étapes suivantes :

1. créer une nouvelle expérimentation, *File – New Experiment* pour ouvrir la fenêtre (New Experiment), puis la remplir par les données nécessaires,

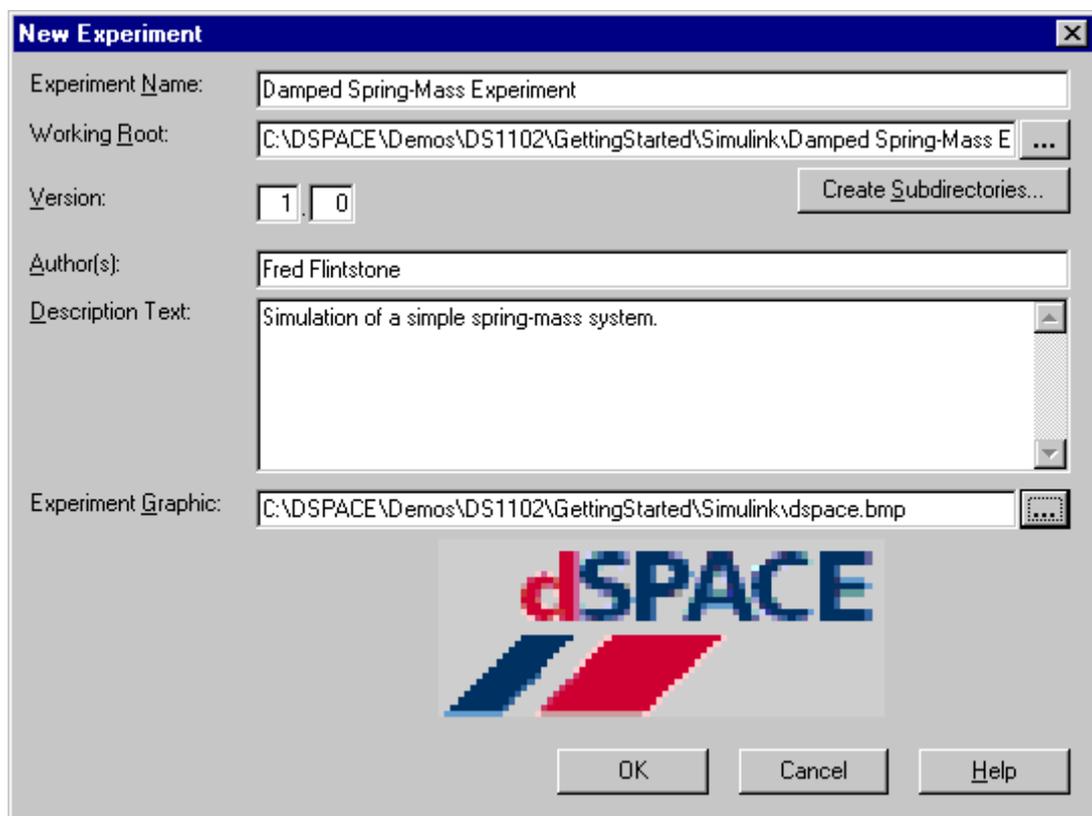


Figure III. 6 Création d'une nouvelle expérimentation

ou modifier les propriétés d'une expérimentation déjà existante. Pour les changer, il faut d'abord ouvrir cette expérimentation *File – Open Experiment*, ensuite on sélectionne dans la barre des menus *Experiment – Configure Setting* pour ouvrir le dialogue (*Configure Experiment Settings*) dans lequel on doit effectuer les modifications nécessaires.

2. ajouter des fichiers aux expérimentations ou supprimer des fichiers des expérimentations.

ControlDesk dispose de plusieurs méthodes pour ajouter un fichier aux expérimentations

- ajouter un fichier d'un composant de ControlDesk
 - choisir *File – open Experiment* dans la barre des menus pour ouvrir l'expérimentation désirée,
 - dans la zone de travail, sélectionner le fichier à ajouter dans l'expérimentation,
 - utiliser la fonction *Add to Experiment* qui se trouve dans le menu contexte pour ajouter le fichier sélectionné.

ControlDesk ajoute les fichiers aux expérimentations et au navigateur d'expérimentation (*Experiment Navigator*).

Remarque : ces fichiers sont ouverts automatiquement lorsque l'expérimentation est chargée.

- ajouter tous les fichiers ouverts
 - choisir *File – open Experiment* dans la barre des menus pour ouvrir l'expérimentation désirée,
 - choisir *File – Add All Opened Files* dans la barre des menus.

L'Experiment Manager ajoute tous les fichiers ouverts de ControlDesk aux expérimentations et au navigateur d'expérimentations.

- importer un fichier de tous types
 - choisir *File – Imports Files* dans la barre des menus et sélectionner le fichier à ajouter.

La suppression d'un fichier de l'expérimentation est faite par la fonction *Remove from Experiment* dans le menu contexte (ou par la touche de clavier *Delete*).

Remarque : pour afficher le menu contexte, on clique par le bouton droit de la souris sur l'élément désiré.

3. Sauvegarder (*file – Save Experiment ou Save Experiment*), fermer (*File – Close Experiment*) et charger (*File – Open Experiment*) l'expérimentation.
4. archiver les expérimentations et leurs fichiers en tant que *ZIP archive* en utilisant la fonction *Experiment – Zip* dans la barre des menus.
5. possibilité d'utiliser l'expérimentation dans une autre plateforme. pour le faire on doit, respecter les instructions suivantes :
 - charger une expérimentation,

- dans le menu contexte de la variable Browser, choisir *Replace SDF-File* et sélectionner le fichier de la description système pour la plateforme désirée dans le dialogue,
- dans le menu contexte de la variable Browser, choisir *Assign Platform*, puis dans le dialogue *Choose Platform* cliquer sur le bouton *Platform*. Cette action charge automatiquement les fichiers objets et les modèles Simulink dans la plateforme sélectionnée,
- sauvegarder la nouvelle expérimentation.

III.2.1.1 Exemple de manipulation d'expérimentation [9]

On veut créer une nouvelle expérimentation nommée *demonstrationexperiment* qui contient le dossier *worddocuments*. On procède comme suit :

1. créer une nouvelle expérimentation (*demonstrationexperiment*),
2. dans le dialogue *New Expériment* entrer les données nécessaires et cliquer sur *Create Subdirectories* pour créer un dossier nommé *Worddocuments*.
3. ajout des fichiers externes et des fichiers SDF

On peut utiliser Microsoft Word pour créer deux fichiers externes (*test.doc* et *description.doc*) dans le dossier *Worddocuments*. Dans la barre des menus on sélectionne *File – import Files* pour ouvrir le dialogue.

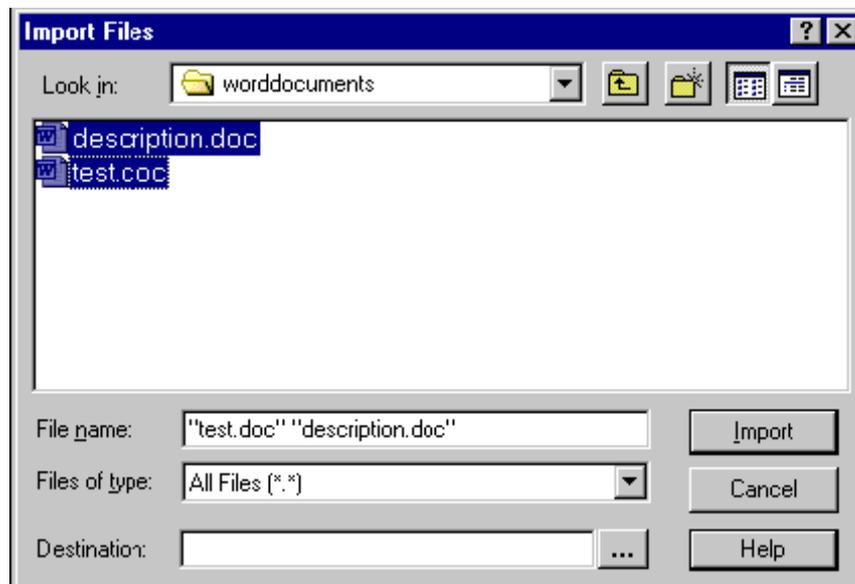


Figure III. 7 Fenêtre d'ajout des fichiers

Sélectionner les deux fichiers puis cliquer sur le bouton *Import* pour les ajouter à l'expérimentation (*demonstrationexperiment*)

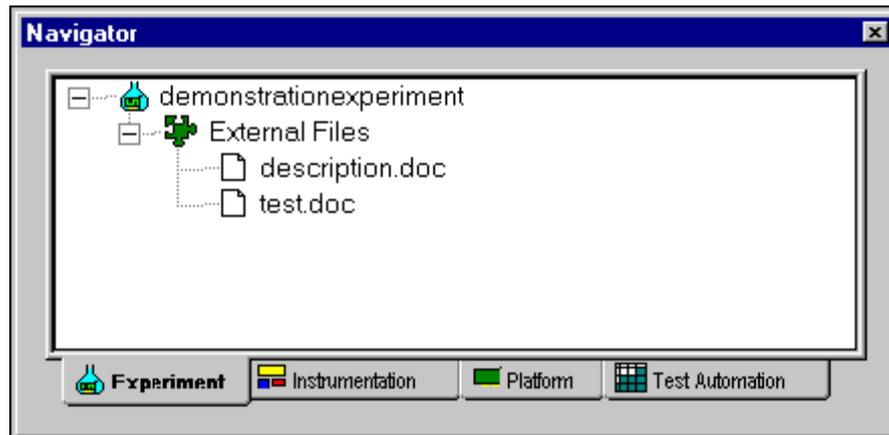


Figure III. 8 Ajout des fichiers externes

Ajouter des fichiers SDF

- dans le menu file, sélectionner *Open Variable File*,
- dans le dialogue *Open*, ouvrir le fichier *smd_1102_sl.sdf* (fichier du type *.sdf),
Ce fichier est chargé dans variable browser,
- Dans le menu contexte de la variable browser, ajouter le fichier à l'expérimentation (*Add Experiment*).

4. vérification des fichiers importés dans l'expérimentation

Cliquer sur *Experiment* dans le Navigateur et sélectionner *Experiment – Directory Hierarchy* dans la barre des menus.

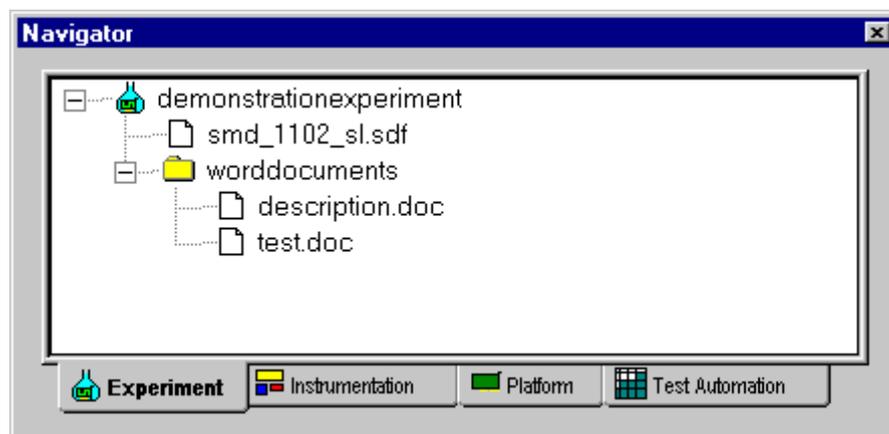


Figure III. 9 Les fichiers ajoutés à l'expérimentation

III.3 Gestion des plateformes et manipulation des applications

(Managing Platforms and Handling Applications) [1] [9]

Le *platform Manager* fournit des fonctions pour manipuler les plateformes installées dans le PC, et les applications assignées à ces plateformes. Il fournit également une interface avec Simulink pour manipuler des simulations Simulink.

III.3.1 Les éléments de bases du gestionnaire de plateforme

Le navigateur de plateforme affiche toutes les plateformes enregistrées par ControlDesk, et fournit des fonctions adéquates à la gestion de ces plateformes. Le sélecteur de fichier permet de charger les applications dans la plateforme en utilisant l'opération glisser et déposer.

Les commandes du navigateur de plateforme

Le navigateur de plateforme et le sélecteur de fichier contiennent leurs propres menus contextes qui fournissent les commandes requises. Certaines de ces commandes sont également disponibles sur le menu de plateforme et sur la barre d'outil du navigateur de plateforme.

Barre d'outil du navigateur de plateforme

La barre d'outil fournit les commandes suivantes (de gauche vers la droite) :
Registre, charger une application ou un modèle, propriétés, recharger une application ou démarrer une simulation avec le Simulink, arrêter le processeur en temps réel ou la simulation avec le Simulink, ouvrir le fichier LOG.



Figure III.10 Barre d'outil de navigateur de plateforme

La barre d'outil est disponible lorsque le navigateur de plateforme est sélectionné dans le menu **View-Toolbars**.

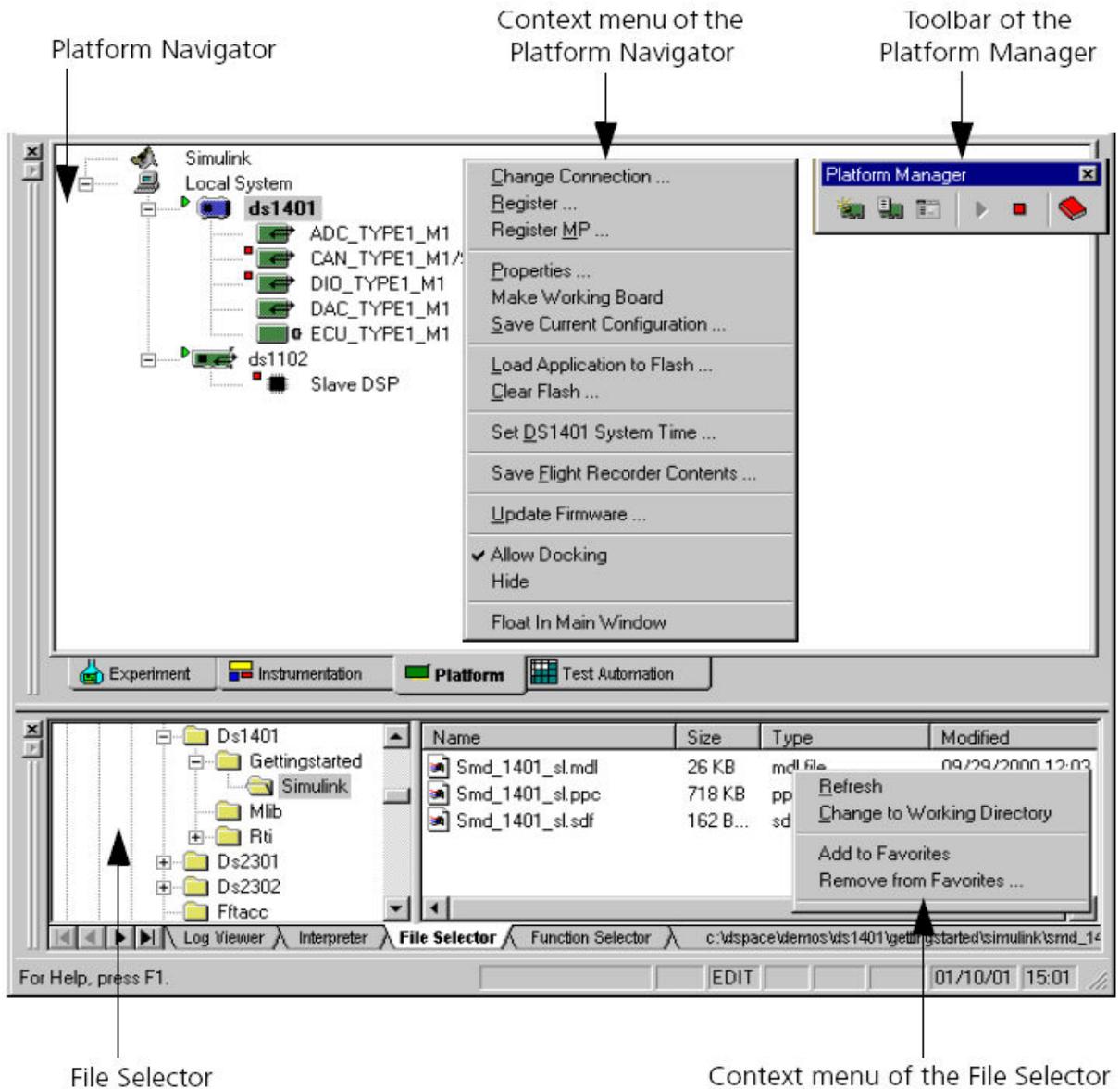


Figure III.11 Gestionnaire de plateforme

III.3.2 Enregistrement des plateformes

L'enregistrement de la carte (en temps réel) de dSPACE pour un processeur simple (single processor) est différent de celui de multiprocesseur (Multiprocessor systems).

III.3.2.1 Enregistrement d'un système à processeur unique (single processor)

Pour enregistrer le processeur dans la plateforme on doit suivre les étapes suivantes :

1. choisir *platform – Inialization – Register*, dans la barre des menus,

2. dans le dialogue *Register Board*, sélectionner le type de la carte installée par exemple DS1103,
3. entrer l'adresse d'E/S choisie dans le champ *Port Adress* en hexadécimales,
4. dans le dialogue *Register Board*, cliquer sur *Register* pour compléter l'enregistrement et retourner dans le navigateur de plateforme.

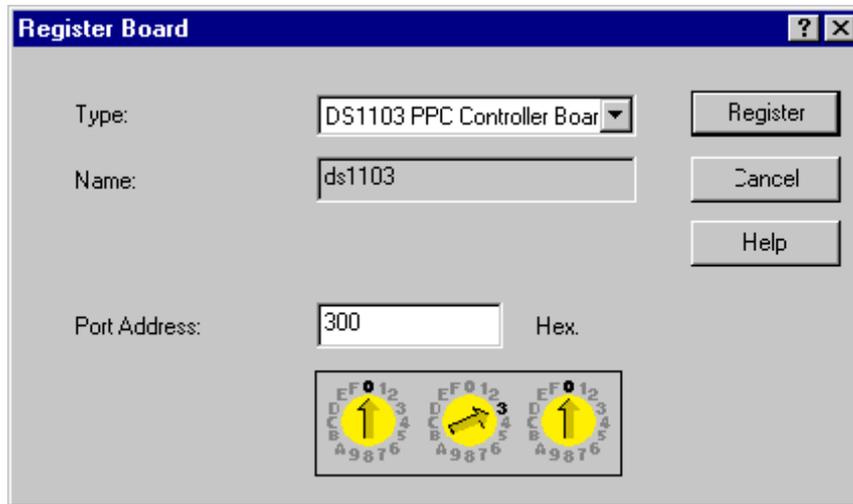


Figure III. 12 Fenêtre d'enregistrement de la carte

En plus des données entrées dans le dialogue *Register Board*, ControlDesk offre plus d'informations sur la carte dans le dialogue *Processor Board Properties* (sélectionner *Platform – Properties* dans la barre des menus), par exemple, la capacité mémoire (RAM), l'horloge de processeur, et les noms des applications chargées sur le processeur ... etc.

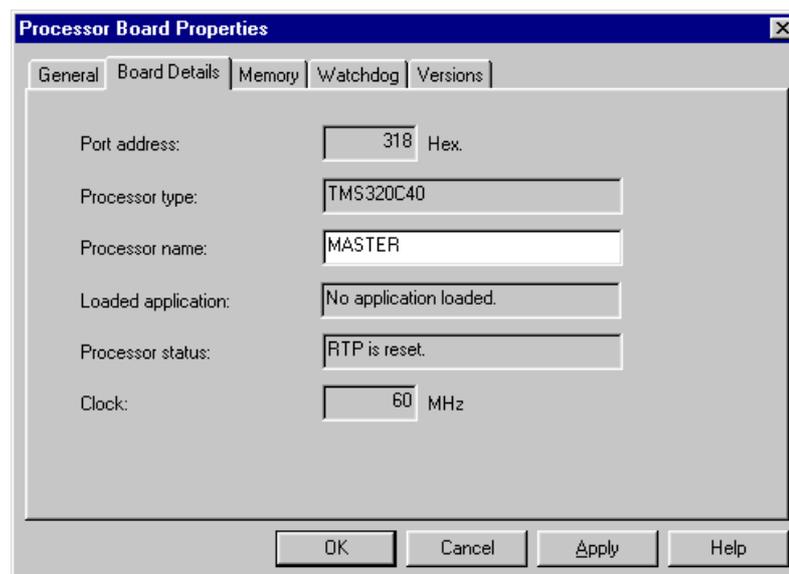


Figure III. 13 propriétés de la carte

III.3.3 Création des fichiers SDF

Les fichiers SDF décrivent les fichiers à charger individuellement dans la plateforme de Simulation. Ils sont soit créés à l'aide de l'éditeur SDF soit générés automatiquement par RTI.

III.3.3.1 Editeur des fichiers SDF

Pour générer un fichier SDF avec l'éditeur de fichier SDF on procède comme suit :

1. choisir *Tools – SDF File Editor*, dans la barre des menus. Un dialogue de *SDF File Editor* s'ouvre ,

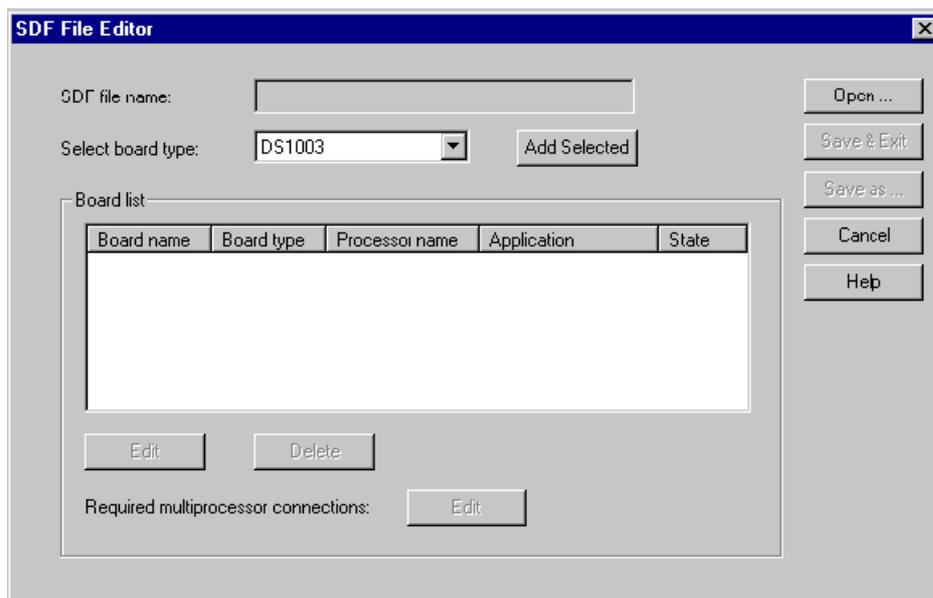


Figure III. 14 Editeur des fichiers SDF

2. dans le champ *Select board type*, sélectionner le type de plateforme et cliquer sur le bouton *Add Selected*. Selon la plateforme choisie, le dialogue *Single Board* ou le dialogue *DSxxxx page* s'ouvre (xxxx correspond au numéro de la carte exemple 1104),
3. dans *Single board (DSxxxx page)*, entrer le nom de la carte, le nom du processeur, l'application qui est utilisée par le processeur, et cliquer *OK*.

III.3.3.2 Création automatique des fichiers SDF

Dans le navigateur de plateforme, charger toutes les applications dans le processeur et dans le menu contexte du navigateur de plateforme, sélectionner *Save Current Configuration* pour que le fichier SDF soit créé.

III.3.4 Manipulation des applications en temps réel

Le navigateur de plateforme facilite la manipulation des applications à charger dans la carte.

Pour charger et démarrer l'application en temps réel, sélectionner le fichier SDF qui correspond à l'application désirée dans le sélecteur de fichier (*File selector*) et glisser et déposer le fichier à la plateforme (DS1104) dans le navigateur de plateforme. Dès que l'application est chargée, le processeur démarre son exécution automatiquement.

Remarque : le chargement d'une application peut se faire autrement, en choisissant *Platform – Application – Load Application* dans la barre des menus.

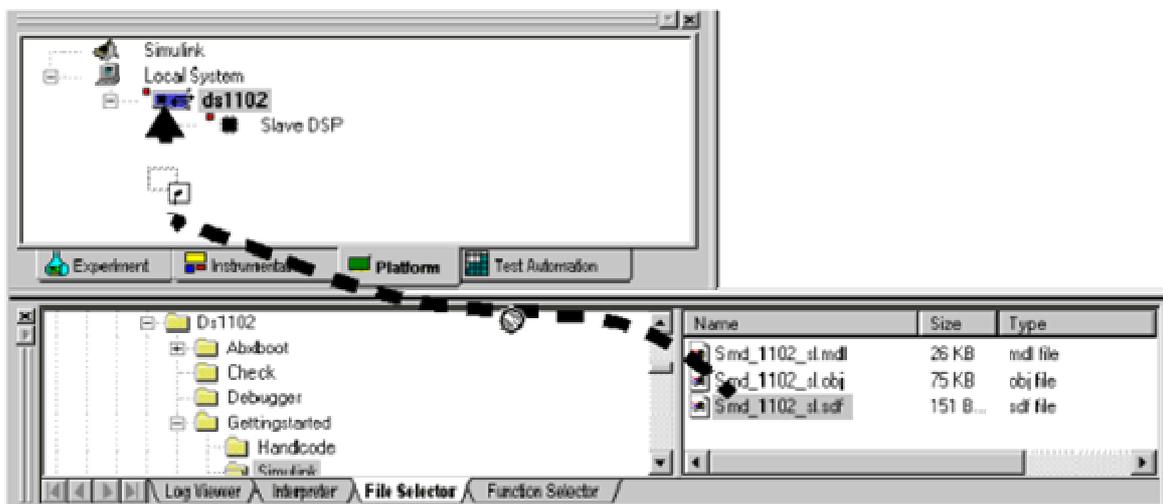


Figure III. 15 Chargement d'un fichier dans la carte

Pour arrêter une application en temps réel, sélectionner le type de la carte dans le navigateur de plateforme, et dans la barre des menus cliquer sur *Platform – Application – stop Real – Time Processor*.

III.3.5 Manipulation des applications sur le processeur esclave DSP

Les cartes DS1102, DS1103, DS1104, DS2201 et DS2210 contiennent un processeur esclave DSP. Habituellement, les applications du processeur DSP sont chargées et lancées en même temps que les applications en temps réel (processeur principal).

Pour assigner une application esclave au processeur DSP (*esclave DSP*)

- dans le navigateur de plateforme, choisir le processeur esclave (Slave DSP),
- dans la barre des menus, cliquer sur *platform – save application– assign save application* ; le dialogue *Load Application to Slave DSP* s'ouvre,
- dans ce dialogue, sélectionner une application esclave (slave application).

III.4 Construction des panneaux d'instruments [1]

Le ControlDesk standard fournit un ensemble d'instruments puissants. Ils sont conçus pour surveiller et commander les variables de simulateur et pour afficher les données capturées. Ces instruments peuvent être arrangés librement dans une ou plusieurs fenêtres, appelées layout, chaque instrument connecté à une variable est appelé panneau d'instruments virtuel. Deux étapes sont nécessaires pour construire un panneau d'instruments virtuel :

- créer un Layout, sélectionner et arranger les instruments,
- connecter les instruments aux variables de simulateur.

III.4.1 Outil d'instrumentation

Il est composé d'un Navigateur d'instrumentation, d'un sélecteur d'instruments, et de la fenêtre Layout.

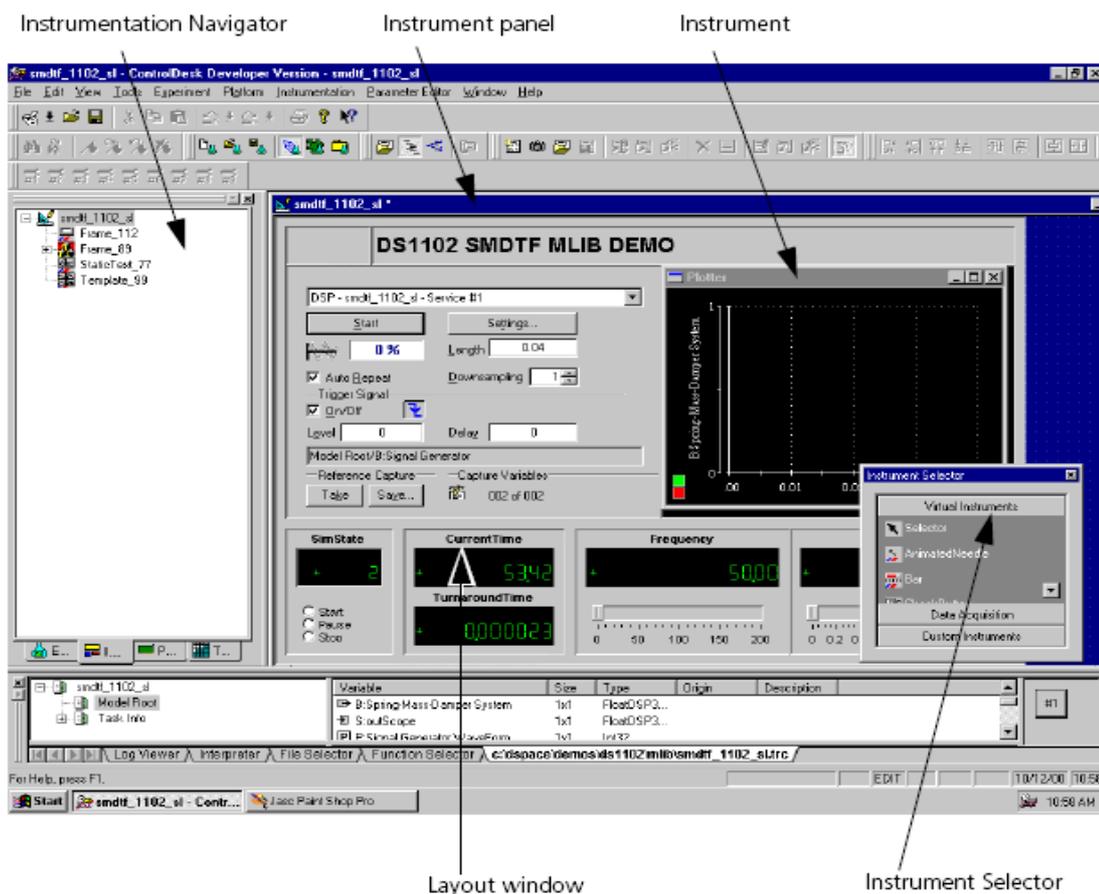


Figure III. 16 Outil d'instrumentation

III.4.1.1 Navigateur d'instrumentation

Le navigateur d'instrumentation affiche la hiérarchie de tous les panneaux d'instruments ouverts et leurs instruments.

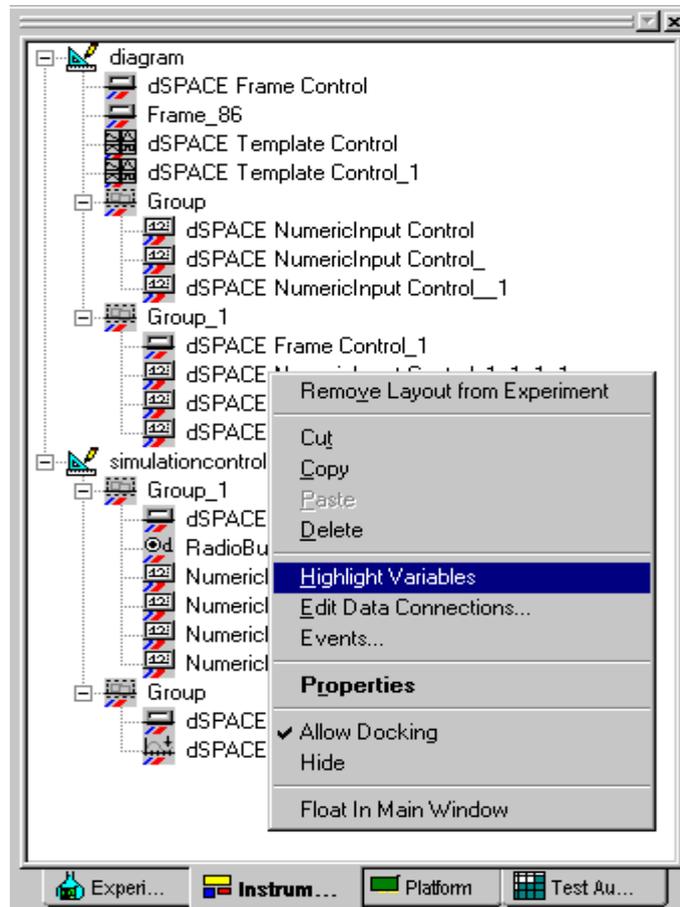


Figure III. 17 Navigateur d'instrumentation et son menu contexte

Pour faciliter la manipulation des instruments, le navigateur dispose de plusieurs commandes via la barre d'outils d'instrumentation et de Layout.

La barre d'outils d'instrumentation permet la commutation entre les différents modes.

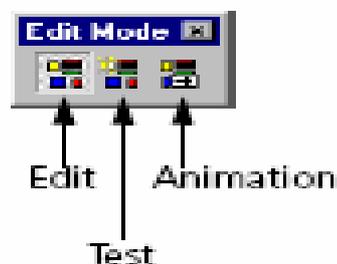


Figure III. 18 Barre d'outil d'instrumentation

Mode édition (Edit mode) : il permet de créer, et de modifier les panneaux d'instruments.

Mode test (Test mode) : il permet de tester l'interface des panneaux d'instruments.

Dans ce mode, il est possible d'utiliser des instruments d'entrée (par exemple un glisseur) et de modifier les connexions de données.

Mode animation (Animation mode) : il rend les panneaux d'instruments opérationnels.

Il permet aussi de changer les paramètres à l'aide des instruments d'entrée, et de visualiser les signaux de sortie.

Les commandes du Layout

La barre d'outils Layouting est activée lorsqu'on sélectionne Layouting dans le menu *View* – *toolbars*.



Figure III. 19 Barre de layouting

III.4.1.2 Panneaux d'instruments

La fenêtre Layout est utilisée pour construire les panneaux d'instruments ; selon les besoins de l'application, et pour connecter les variables de simulateur de l'application en temps réel aux instruments.

La figure suivante illustre une fenêtre Layout et son menu contexte :

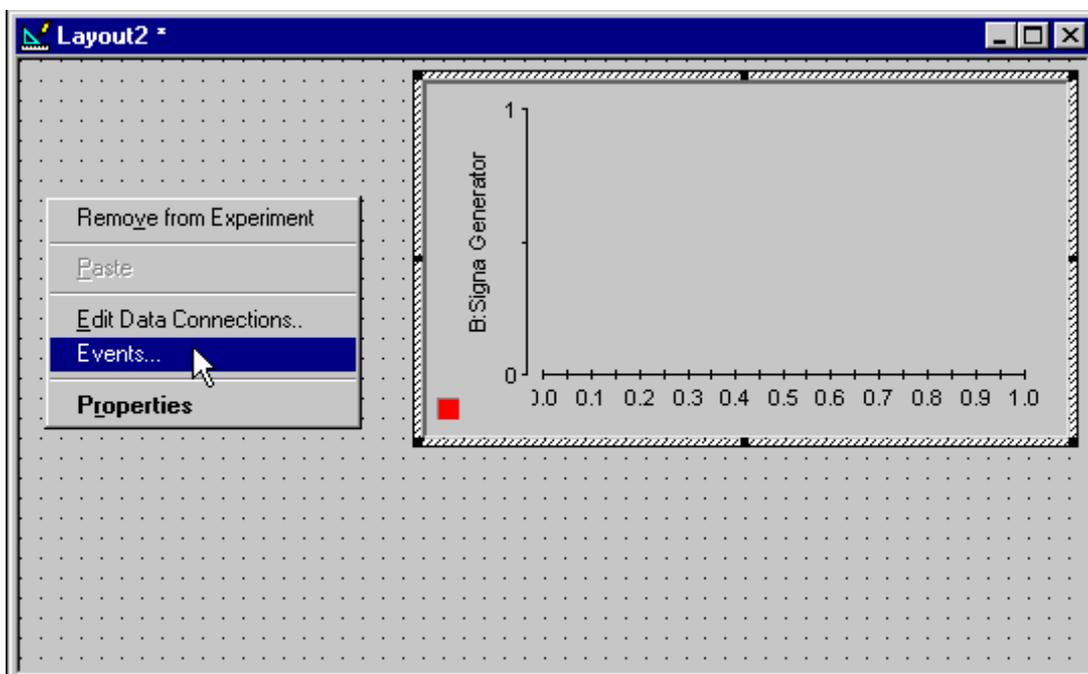


Figure III. 20 Layout et son menu contexte

III.4.1.3 Sélecteur d'instruments

Le sélecteur d'instruments assure un accès facile et rapide à tous les instruments.

ControlDesk standard offre un ensemble d'instruments divisés en 3 groupes :

a) **Les instruments virtuels**

Ils sont utilisés pour afficher les valeurs des variables connectées, ou pour entrer des valeurs dans la plateforme de simulation.

Voici la liste des instruments virtuels disponibles.

Icône	Instrument	But
	AnimatedNeedle	To display the value of a connected numeric variable by a needle deflection.
	Bar	To display the value of a variable by a bar deflection.
	CheckBox	To display whether the value of a connected variable or instrument matches predefined values or to write a predefined value to a connected variable or instrument.
	Display	To display numerical values with a predefined format.
	Frame	To arrange instruments on the layout.
	Gauge	To display the value of a connected numeric variable by a needle deflection on a circular scale.
	InvisibleSwitch	To simulate the different buttons (On/Off button, Push button, Check button) with a transparent background.
	Knob	To set the value of the connected numeric variable or display the value of the connected variable on a circular scale.
	Message	To display messages that are related to values of the connected numeric variable.
	MultiStateLED	To display an LED or image that is related to values of the connected numeric variable.
	NumericInput	To enter or display a value that is transferred to a connected variable or instrument.
	OnOffButton	To set the value of the connected numeric variable or activate/close an event-based layout.
	PushButton	To write a predefined value to a connected variable or instrument and/or start an action, for example to call up a layout.
	RadioButton	To set an option and transfer a specified value to the simulation platform or display the state of a variable.
	Slider	To enter values of a connected variable to the simulation platform or display the values of the variable.
	StaticText	To display explanations or inscriptions on the layout.

	TableEditor	To display and change values of a connected table variable in a chart and a grid view.
---	-------------	--

Tableau III. 2 liste des instruments virtuels

b) Instruments d'acquisition de données (*Data acquisition instruments*)

Ils sont utilisés pour capturer les données de simulation (le tracé dans le temps).

Les instruments disponibles sont

Icon	Instrument	Purpose
	Capture Settings	To control/observe the properties of data captures.
	LogicAnalyzer	To display bit patterns of captured integer variables.
	Plotter	To display the values of several variables related to different y-axes.
	Template	To combine the functions of the LogicAnalyzer, Plotter and XYPlot instruments.
	XYPlot	To display variables as functions of other variables.

Tableau III. 3 liste des instruments d'acquisition de données

Création d'instruments dans le panneau d'instruments

- dans le sélecteur d'instrument, cliquer sur un groupe (Virtual instruments, Data acquisition... etc),
- cliquer sur l'instrument désiré,
- dans le panneau d'instruments, dessiner un rectangle avec la souris pour ouvrir le l'instrument.

Ajout d'un nouvel instrument dans la bibliothèque d'instruments (Custom Instruments Library)

- dans le panneau d'instruments, sélectionner l'instrument désiré,
- dans le sélecteur d'instrument, ouvrir le groupe désiré,
- glisser et déposer l'instrument sélectionné vers le groupe ouvert dans le sélecteur d'instrument. Un dialogue « *Custom Instrument Properties* » est appelé,
- dans ce dialogue, sélectionner une icône pour représenter le nouvel instrument dans le sélecteur d'instrument,
- entrer le nom du nouvel instrument, puis cliquer sur OK.

Le nouvel instrument est ajouté dans la bibliothèque d'instruments (*Custom Instrument Library*). L'icône sera affichée dans le groupe.

Groupe d'instruments

Pour un accès facile et rapide aux instruments, le sélecteur d'instruments permet de construire des groupes d'instruments.

La création d'un nouveau groupe d'instruments dans le sélecteur d'instrument se fait comme suit :

- dans le menu contexte du sélecteur d'instrument, choisir *Customise* pour ouvrir le dialogue *Customise Instrument selector*,
- dans la page *Groups in Instrument Selector*, cliquer sur *New*. Un nouveau bouton avec le nom *NewGroup* est créé dans le sélecteur d'instruments,
- sélectionner *NewGroup* dans la liste *Groups*, et entrer son nom dans le champ *Group Name*,
- cliquer sur *OK* pour appliquer les changements,
- ajouter des instruments au nouveau groupe (new group).

Ajout des instruments au nouveau groupe

- dans le sélecteur d'instruments, cliquer sur le groupe désiré (à personnaliser),
- dans le menu contexte du sélecteur d'instruments, choisir *Customise* pour ouvrir le dialogue *Customise Instrument selector*,
- dans la page *Instrument Libraries*, sélectionner la bibliothèque et les instruments désirés,
- glisser et déposer l'instrument dans le sélecteur d'instrument.

L'icône et le nom de l'instrument sont affichés dans le groupe sélectionné du sélecteur d'instrument.

Pour supprimer un groupe d'instrument, taper la touche *Delete* du clavier.

III.4.1.4 Exemple de création d'un nouveau groupe

Cet exemple consiste à créer un nouveau groupe nommé *MyGroup* dans le sélecteur d'instrument qui va contenir les instruments *MyLED_1*, *MyKnob_1* (instruments virtuels), *Plotter* et *Template* (acquisition de données).

1. création d'un groupe d'instrument, *MyGroup*
 - dans la barre des menus, choisir *Instrumentation – Edit Mode*,
 - si le sélecteur d'instrumentation n'est pas affiché, choisir *View – Controlbars – Instrument Selector* dans la barre des menus pour l'afficher,

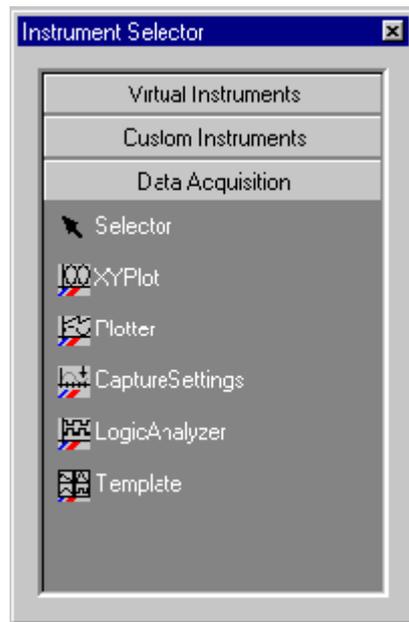


Figure III. 21 Groupe d'acquisition de données et ses éléments

- dans le menu contexte de sélecteur d'instrument, choisir *Customise* pour appeler le dialogue *Custom Instrument Selector*,
- dans ce dialogue, cliquer sur *New* puis sélectionner *NewGroup* dans la liste *Groups*
- entrer le nom du nouveau groupe (MyGroup)
- cliquer sur OK,

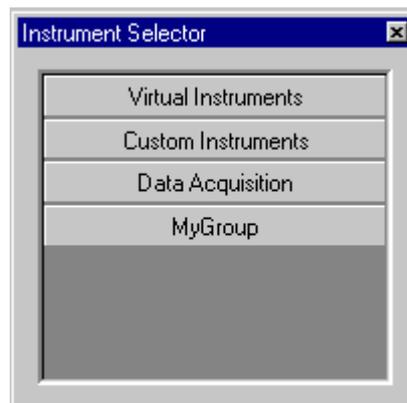


Figure III. 22 Sélecteur d'instrument

2. ajout d'instruments au nouveau groupe MyGroup

- créer les instruments MultiStateLED et Knob (Virtual instrument) dans le Layout,
- cliquer sur le groupe MyGroup dans le sélecteur d'instrument pour l'afficher,

- glisser et déposer l'instrument MultiStateLED dans le sélecteur d'instrument. Le dialogue *Custom Instruments Properties (Button Appearance)* s'ouvre,

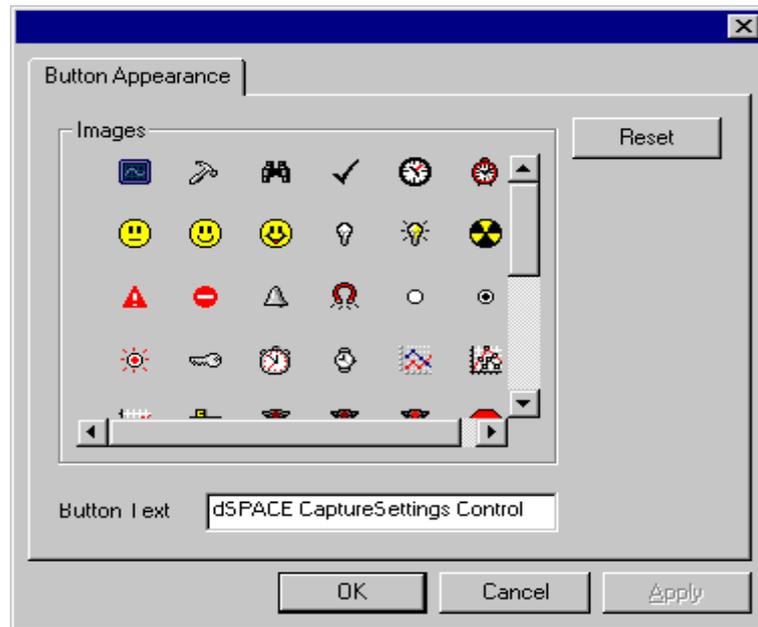


Figure III.23 Icônes des boutons

- dans ce dialogue, sélectionner une icône pour représenter l'instrument,
- dans le champ *Button Text*, entrer le nom de l'instrument (exemple MyLED_1), puis cliquer sur OK,
- refaire la même la chose pour l'instrument Knob.

Les deux instruments sont ajoutés au groupe (MyGroup).

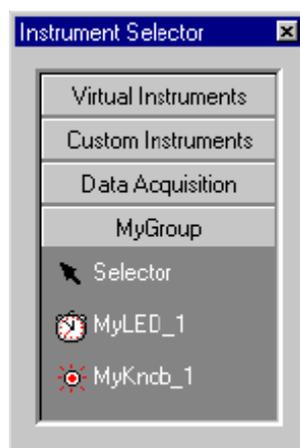


Figure III. 24 Création d'un nouveau groupe d'instrument

3. ajouter *Plotter et Template* au groupe

- cliquer sur le bouton MyGroup dans le sélecteur d'instrument,
- dans le menu contexte de sélecteur d'instrument, choisir *Customise* pour ouvrir le dialogue *Custom Instrument Selector*,

- dans la page *Instrument Libraries*, choisir *Data acquisition*,
- sélectionner l'instrument *Plotter* et le glisser et déposer dans le sélecteur d'instrument,
- refaire les mêmes étapes pour l'instrument *Template*,
- cliquer sur OK pour accepter les changements.

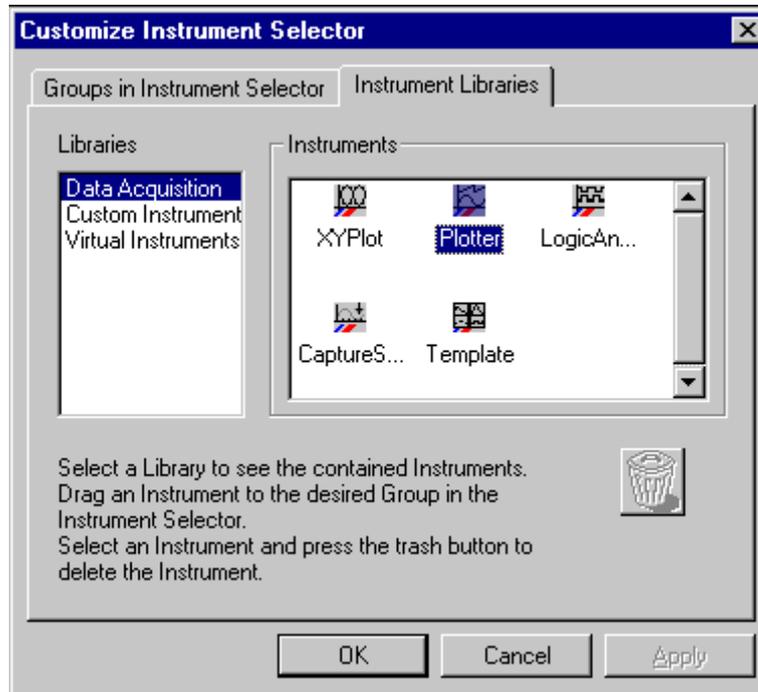


Figure III. 25 Ajout de plotter au nouveau groupe

III.4.2 Création du panneau d'instruments

ControlDesk fournit plusieurs fonctions pour construire des panneaux d'instruments.

III.4.2.1 Création d'un nouveau panneau d'instruments

Créer un nouveau panneau instruments exige les étapes suivantes :

- créer un nouveau Layout

On distingue deux modes de Layout :

- **Layout standard** qui peut contenir les instruments virtuels et les instruments d'acquisition de données,
- **Layout de capture** qui contient seulement les instruments d'acquisition de données.

Dans la barre des menus, choisir *file – New Layout* pour créer un nouveau Layout.

Les propriétés de chaque Layout (dans le menu contexte, sélectionner *Properties*) peuvent être modifiées individuellement. La fenêtre *Layout Properties* contient deux pages, *Layout* et *Data Kernel*.

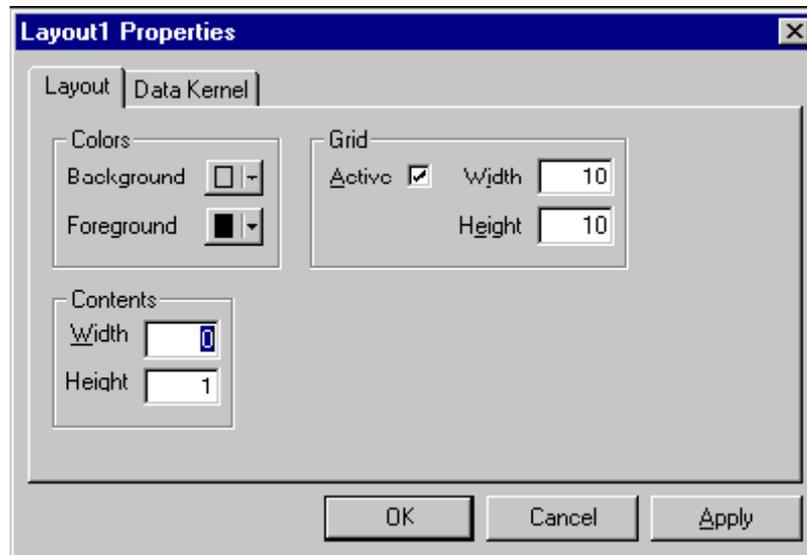


Figure III. 26 Propriétés du Layout

- insérer un instrument

On peut ne ajouter des instruments aux Layouts qu'au mode Edit (Edit Mode). Pour ouvrir *Edit Mode*, sélectionner *Instrumentation – Edit Mode* dans la barre des menus.

Pour ajouter des instruments au panneau d'instruments,

- dans le sélecteur d'instrument, cliquer sur un groupe d'instruments,
- cliquer sur l'icône de l'instrument,
- dans le panneau d'instruments, utiliser la souris pour dessiner un rectangle.

Le nouvel instrument est ouvert dans le panneau d'instruments et son icône est ajouté au navigateur d'instrumentation. Le cadre de couleur rouge (le rectangle) indique qu'il n'y a pas de connexion de données.

Pour supprimer un instrument dans le panneau, sélectionner l'instrument désiré puis dans la barre des menus, choisir *Edit – Delete* ou *suppr* du clavier.

Il suffit d'un double clic sur l'instrument pour modifier ses propriétés.

III.4.2.2 Modification des propriétés d'instruments

Chaque instrument possède des propriétés qui définissent son comportement. Il suffit de cliquer sur *Properties* dans le menu contexte de l'instrument ou un double clic sur l'instrument pour afficher le dialogue *Properties*.

Utilisation des légendes (Using Captions)

Les légendes sont utilisées pour marquer les instruments, afficher le contenu des variables, et le nom de la variable.

Afin de bien comprendre on va illustrer les étapes avec un exemple.

- Dans le dialogue *Properties*, sélectionner la page *Captions*,



Figure III. 27 Propriétés de l'instrument MultiStateLED

- Cliquer sur *Color* pour choisir la couleur des textes de légende,
- Cliquer sur *Font* pour choisir la police. Enter *output* comme texte à afficher dans la légende (caption),
- Cliquer sur *Macro* et choisir le nom de la variable. Le macro $\{\%VARIABLE\%$ est ajouté au texte dans le domaine des textes. Il affichera le nom de la variable dans la variable connectée.

Conversion des valeurs

Les nombres à virgule flottante utilisés sur le PC peuvent être convertis en utilisant des fonctions bien appropriées (*forward scaling function*).

Les entiers utilisés sur l'ECU (engine control units) peuvent être convertis en utilisant des fonctions correspondantes (backward scaling function).

Plusieurs instruments emploient des fonctions appropriées pour représenter les valeurs selon nos besoins.

Ces fonctions peuvent être simples et linéaires telles que $f(x)=ax+b$, ou plus complexes, $f(x) = (ax^2 +bx+c)/(dx^2 +ex+f)$.

Fonction d'échelle (Scaling function)

Le scaling function forward convertit les valeurs de variable du simulateur en valeurs appropriées pour être affichées par l'instrument.

Le scaling function backward convertit les valeurs pour être écrites par l'instrument dans le variable du simulateur.

Si la fonction *forward* est une fonction mathématique linéaire, alors la fonction *backward* est générée automatiquement ; par contre si la fonction mathématique n'est pas linéaire, il faut spécifier la fonction *backward*.

Si on définit la fonction d'échelle pour une variable, et si on connecte cette variable à l'instrument, les variables de la fonction sont automatiquement transférées du fichier TRC à l'instrument.

Mode de conversion de valeur (Value conversion mode)

Cette caractéristique présente deux modes dans un instrument virtuel.

- le mode de conversion source (*value conversion mode source*) affiche le nombre sans échelle,
- le mode converti (*converted*) génère le nombre à afficher en utilisant la fonction d'échelle.

Les fonctions d'échelles sont intégrées sous différentes formes, selon l'instrument.

Dans le groupe d'instruments virtuels, seuls les instruments suivants contiennent la page de conversion de valeurs (*value conversion*), *Bar*, *Display*, *Gauge*, *Knob*, *Numerilnput*, et *Slider*, et dans le groupe d'instrument d'acquisition de données, seul l'instrument *plotter* contient le cadre conversion de valeurs (*value conversion*) dans la page *signals*.

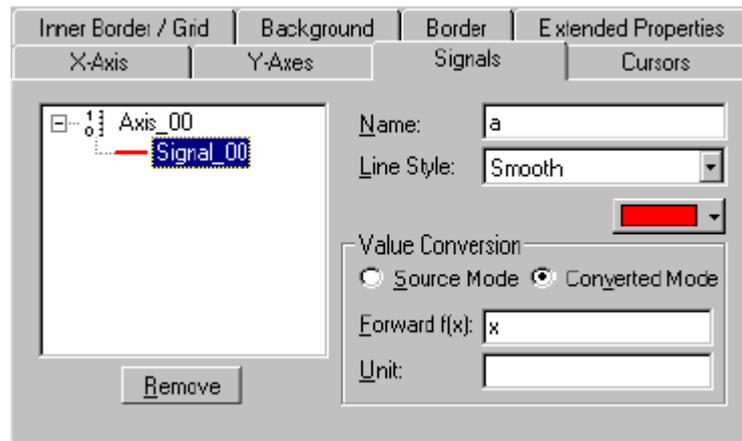


Figure III. 28 Propriétés du Plotter

III.4.2.3 Layout de capture

Ce Layout n'est utilisable que pour certains instruments d'acquisition de données Plotter, XYPlot et LogicAnalyzer.

Dans la barre des menus, choisir *file – new – capture layout* pour créer un nouveau layout.

Ajout d'instrument au layout (capture layout)

- dans le menu contexte du layout de capture, cliquer sur *properties* pour ouvrir le dialogue *Template properties*,

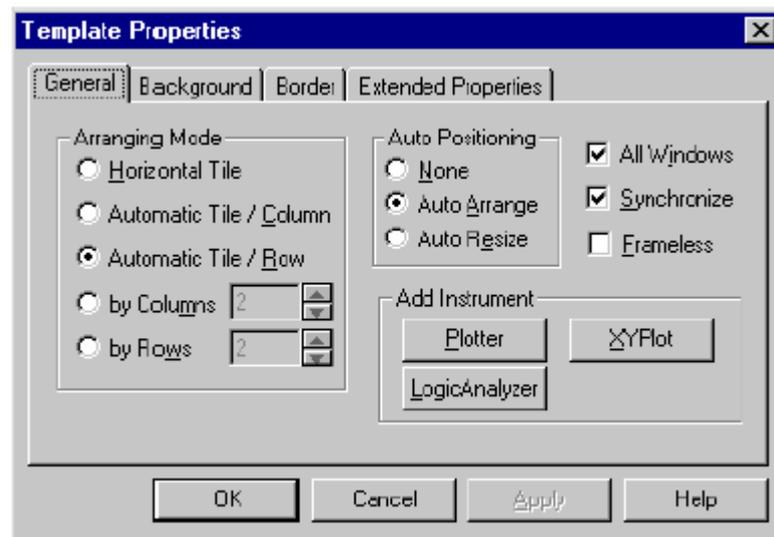


Figure III. 29 Propriétés du Templates

- cliquer sur l'instrument à ajouter au layout dans le cadre *Add Instrument* (voir la figure), puis cliquer sur OK.

Ajout d'un plotter par connexion de variables au layout de capture

- dans la fenêtre Variable Browser, sélectionner une variable,
- glisser et déposer la variable sélectionnée dans le layout de capture, pour ajouter l'instrument *plotter*.

Pour supprimer un instrument de layout de capture

- dans la barre des menus, choisir *Instrumentation – Animation Mode*.
- dans la disposition de capture, cliquer sur l'instrument à supprimer. La barre d'outil de layout de capture sera automatiquement affichée.
- dans la barre d'outils de layout de capture, désactiver *Frameless View* en cliquant sur le bouton correspondant. Les instruments ne peuvent être supprimés que dans ce mode.

III.4.3 Manipulation des variables

Les panneaux d'instruments sont utilisés pour visualiser les signaux, contrôler les paramètres des applications en temps réel et les simulations Simulink.

III.4.3.1 Création des fichiers SDF et MAP

La création des fichiers objets génère automatiquement les fichiers SDF et MAP, lorsqu'on travaille avec RTI.

Pour les simulations Simulink, on peut invoquer *Build Process* via le menu contexte de la plateforme Simulink.

Création des fichiers SDF pour des simulations Simulink

Ouvrir le menu contexte de la plateforme Simulink et cliquer *Build Variable Description File* pour générer les fichiers SDF.

Les fichiers générés sont nommés <model>_offline.trc. En plus, le fichier SDF <model>_offline.sdf est créé et chargé automatiquement dans la fenêtre Variable Browser.

III.4.3.2 Choix de la plateforme de simulation

La plateforme de simulation peut être la carte DS1104 ou Simulink. S'il n'y a qu'une seule plateforme enregistrée par ControlDesk, elle est employée automatiquement, et s'il y en a plusieurs, ControlDesk sélectionne la plateforme spécifiée par le fichier de description système (*systeme description file*).

Pour choisir la plateforme de simulation, on doit d'abord charger le fichier de description système, puis dans le menu contexte de la variable browser, on clique sur *Assign Platform* pour ouvrir le dialogue *Choose Platform*.

Dans ce dialogue, choisir une plateforme et cliquer sur OK.



Figure III. 30 Choix de plateforme

Remarque :

Le champ 'NoBoard' permet de construire les panneaux d'instruments sans connexion aux plateformes ou même sans le matériel dSPACE.

Dans ce cas il est impossible d'exécuter le mode animation (animation mode).

III.4.3.3 Variable browser

Le variable browser affiche la hiérarchie et les variables des applications en temps réel ou les variables de simulation Simulink après le chargement des fichiers de description système.

On trouve deux zones de variables browser :

- les variables en arborescence (Variable Tree)
- la liste des variables (Variable list) affiche les variables correspondantes.

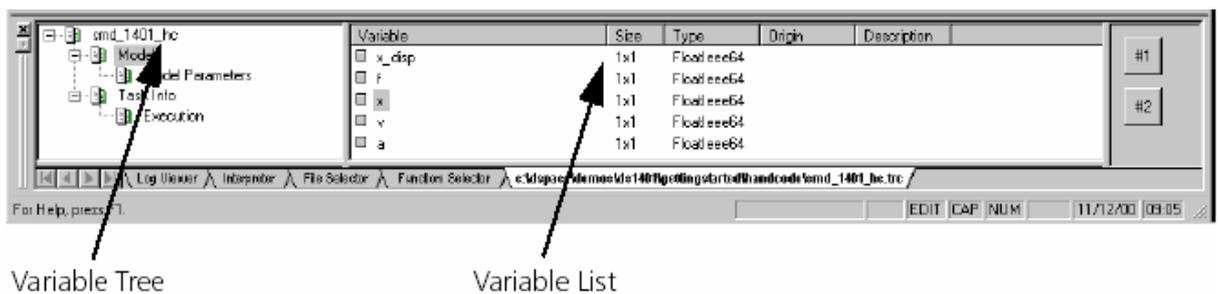


Figure III. 31 Variable browser

III.4.4 Connexion de données

L'étape suivant la construction des panneaux d'instruments est l'établissement des connexions de données entre les différents éléments. Les raccordements de données sont utilisés pour envoyer et recevoir des données ou pour commander des layouts.

Pour déclencher le transfert de données ou l'exécution d'une action, on utilise des événements. Ces événements peuvent être standard au système ou spécifiques à un instrument tel que WriteData.

Types d'événements

Deux types d'événements peuvent être employés, pour déclencher le transfert de données entre deux éléments d'un panneau d'instruments ou d'une variable.

1. *ystème polling* utilisé pour déclencher une opération périodiquement. Un système de temporisation produit l'événement qui déclenche le transfert de données de toutes les variables connectées ou des instruments.

On distingue deux types :

- système poll ; déclenche le transfert de données d'instruments virtuels,
- système poll2 ; déclenche le transfert de données aux instruments d'acquisition de données.

2. *événements spécifiques aux instruments*

La plupart des instruments fournissent plusieurs événements.

On peut prendre n'importe quel instrument dans un layout en tant que source d'événement. L'instrument doit contenir au moins un événement, ceci nous permet de contrôler le transfert de données dans une expérimentation.

Par défaut, l'événement *WriteData* est utilisé pour déclencher une opération d'écriture dans les variables connectées ou dans les instruments.

III.4.4.1 Types de connexion de données

On peut construire différents types de connexions entre les éléments du panneau d'instruments, les connexions peuvent être choisies dans le menu contexte.

III.4.4.2 Connexion de données entre les variables et les instruments

On peut créer une connexion de données standard en utilisant l'opération glisser et déposer ou bien avec la menu contexte.

Une fois que ControlDesk réalise la connexion de données le cadre rouge de l'instrument disparaît.

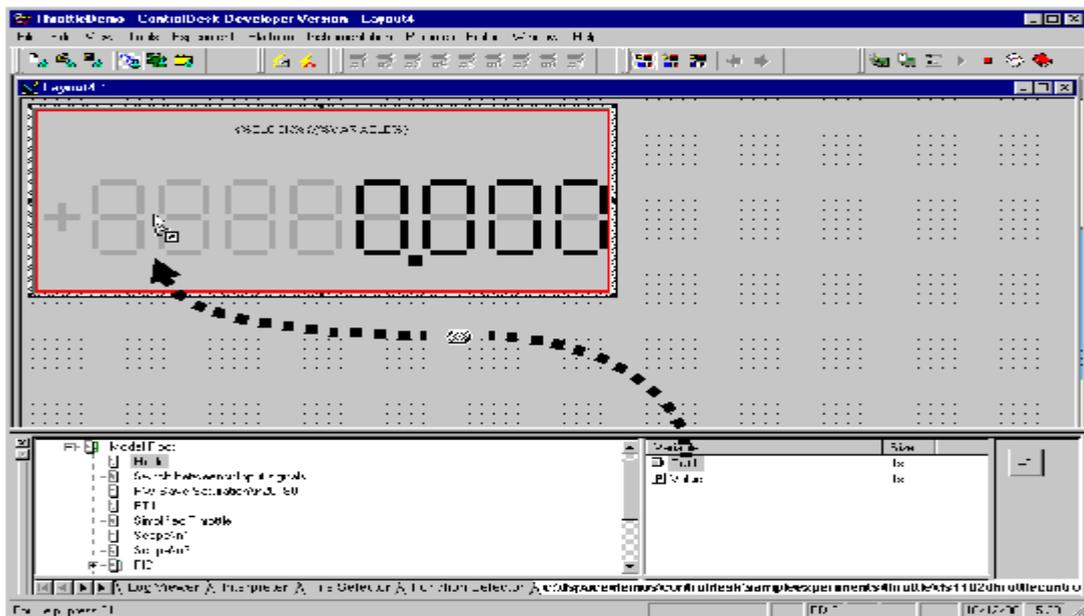


Figure III. 32 Connexion de données

III.4.4.3 Connexion de données entre deux instruments

On peut également transférer des données d'un instrument à un autre. L'instrument peut ne pas être dans le même layout.

Pour choisir le raccordement de données entre deux instruments.

- Choisir l'instrument dans le layout ou dans le navigateur d'instruments,
- glisser et déposer l'instrument sélectionné avec le bouton droit de la souris à l'instrument de cible.

Le menu contexte est affiché.

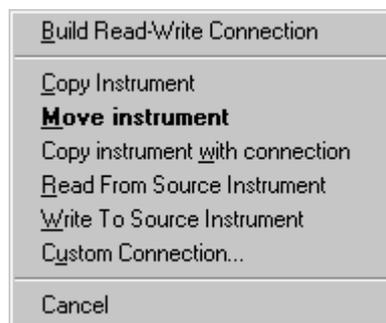


Figure III. 33 Menu contexte

Les raccordements de données qui ne sont pas disponibles ne sont pas réalisés.

- Choisir le type de connexion de données à réaliser.

III.4.4.4 Connexion de l'instrument TableEditor

L'instrument de *TableEditor* est utilisé pour afficher les valeurs d'une table *look up*, d'une matrice ou d'un vecteur connectés. Les valeurs sont affichées numériquement dans une grille et graphiquement dans un diagramme.

Les tables *look up* sont des ensembles de paramètres qui peuvent être affichés dans l'instrument *TableEditor*.

Pour relier une table de look-up

- choisir une table de *look-up* à partir de variable browser /Editeur de paramètres,
- glisser et déposer (bouton gauche de la souris) la table de *look-up* sélectionnée précédemment dans l'instrument de *TableEditor*.

Les valeurs sont affichées quand le mode animation (*animation mode*) est lancé.

Connexion de variables, matrices et vecteurs à l'instrument TableEditor

On peut relier la matrice (ou le vecteur) entière à la table. On peut également connecter des vecteurs séparément à la rangée de la grille (axe des abscisses) ou à la colonne de la grille (axe des ordonnées). Ainsi on peut reproduire une table *look-up* qui dépend de plusieurs variables.

Pour relier une matrice/vecteur entière à l'instrument de TableEditor.

- Dans la liste des variables, sélectionner la matrice/vecteur en cliquant sur l'icône correspondante,
- glisser et déposer (bouton gauche de la souris) la matrice dans la grille ou dans le diagramme du *TableEditor*.

Pour relier un vecteur aux axes des abscisses ou des ordonnées de la grille

- dans la liste des variables, sélectionner le vecteur en cliquant sur l'icône correspondante,
- glisser et déposer (bouton gauche de la souris) le vecteur dans la colonne gauche de la grille. le vecteur est assigné à l'axe des abscisses du diagramme,
- dans la liste des variables, choisir le deuxième vecteur.
- glisser et déposer (bouton gauche de la souris) le vecteur dans la rangée (la ligne) en haut de la grille. Le vecteur est assigné à l'axe des ordonnées du diagramme.

III.4.5 Connexion de données aux instruments d'acquisition de données

Les étapes suivantes expliquent comment connecter des données à l'instrument *Plotter*. Ces démarches sont aussi applicables pour les instruments *Template* et *LogicAnalyzer*.

Utilisation d'un Plotter pour une seule variable

- dans le sélecteur d'instrument, choisir le groupe d'acquisition de données,
- créer un *Plotter*,
- sélectionner une variable dans le variable browser ou dans l'éditeur de paramètres,
- glisser et déposer la variable dans l'axe des y.

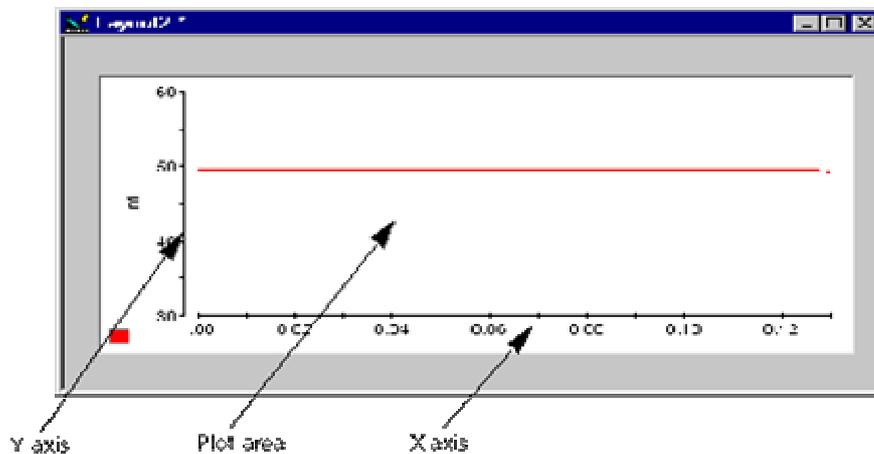


Figure III. 34 Plotter

Utilisation d'un Plotter pour plusieurs variables

- dans le sélecteur d'instruments, choisir le groupe d'acquisition de données,
- créer un *Plotter*,
- glisser et déposer la première variable dans l'axe des ordonnées y,
- pour relier plusieurs variables à l'instrument *Plotter*, sélectionner une autre variable, ControlDesk assigne une nouvelle couleur pour chaque nouveau signal et affiche le nom de la variable sur l'axe des ordonnées (ou des abscisses).

III.4.5.1 Connexion de données à l'instrument *Plotter*

On veut connecter les deux variables à l'instrument en utilisant des axes d'ordonnées différents:

Signal de sortie: x

Signal sortie: f

- dans la barre des menus, choisir *File- Open Variable File* et ouvrir le fichier de description système *smd_xxxx_hc.sdf*,
- réer un nouveau panneau d'instruments,
- créer un *Plotter*,
- dans la fenêtre variable browser, cliquer sur le fichier *smd_xxxx_hc.sdf*.
- dans la liste variable browser, choisir la variable x pour l'ajouter (glisser et déposer) dans l'axe des ordonnées de l'instrument *Plotter*,
- répéter la même chose pour la variable f.

La figure suivante illustre deux signaux connectés à l'instrument Plotter.

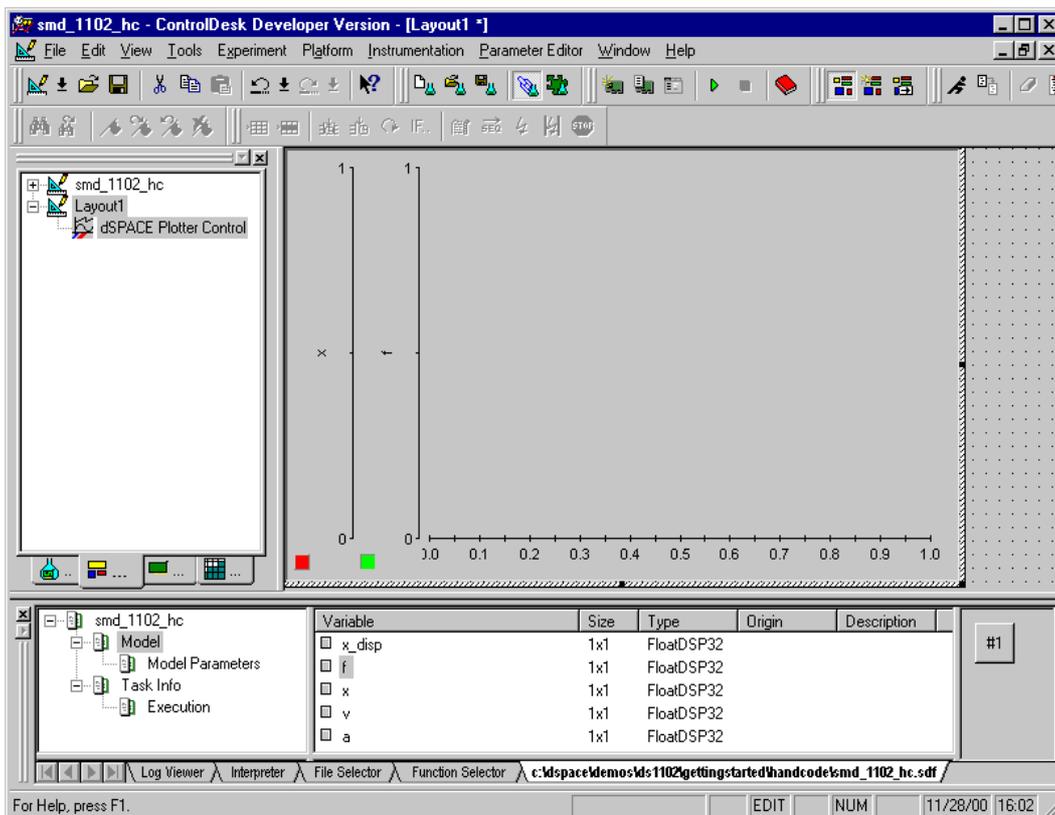


Figure III. 35 plotter avec deux variables

III.5 Utilisation des panneaux d'instruments [1]

Un bon fonctionnement des panneaux d'instruments exige certains arrangements et procédures à respecter.

III.5.1 Mode Opération (Operation Mode)

Dès que les raccordements de données entre les variables de simulateur et les instruments sont générés, le panneau d'instruments peut être opérationnel.

On a deux modes, pour rendre un panneau d'instruments opérationnel.

- Le mode test (Test mode) vérifie l'interface utilisateur du panneau d'instrument. Dans ce mode il est possible d'utiliser des instruments d'entrée (par exemple, un glisseur) et de modifier des raccordements de données, mais il n'y a aucun transfert de données à partir ou vers la plateforme de simulation.
- Le mode animation (Animation mode) rend le panneau d'instruments opérationnel. Dans ce mode les raccordements de données sont activés et les données sont transférés à partir ou vers la plateforme de simulation. Ce mode permet
 - de changer les valeurs de paramètre interactivement avec des instruments d'entrée (tels que des glisseurs),
 - de changer les raccordements de données,
 - de visualiser les signaux avec des instruments d'acquisition de données (tels que Plotters),
 - de capturer les données et sauvegarder les résultats.

Avant de lancer le mode animation, on doit vérifier d'abord si les conditions suivantes sont satisfaites :

- les panneaux d'instruments sont bien chargés,
- les fichiers de description système sont bien chargés.
- les raccordements de données sont définis,
- les fichiers objets sont changés.

III.5.1.1 Exemple d'un modèle dans le mode animation

Cet exemple montre comment lancer le mode animation et comment actionner un instrument pour ajuster une valeur de paramètre.

On charge d'abord le fichier de description système `smd_xxxx_hc.sdf` (`%DSPACE_ROOT%\Demos\DSXXXX\GettingStarted\Handcode\smd_xxxx_hc.sdf`).

Pour charger l'application et créer les instruments

- dans le navigateur de plateforme, charger le fichier de description système `smd_xxxx_hc.sdf`,

ControlDesk charge le fichier de description système et le fichier de description variable (TRC) associé.

- dans le navigateur, sélectionner la page d'instrumentation,
- créer un nouveau panneau d'instrument,
- créer un Plotter et connecter la variable `x` à l'instrument,

- créer un glisseur (gamme 0...1) et connecter la variable m à l'instrument,
- créer un afficheur et connecter la variable m à l'instrument,

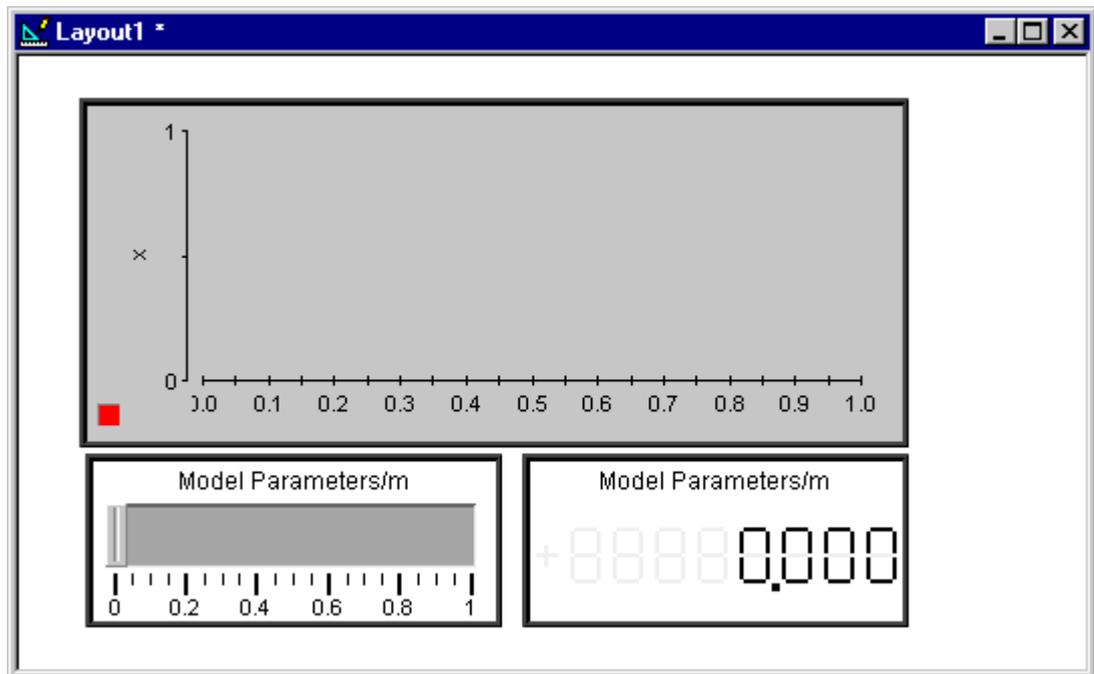


Figure III. 36 Panneau d'instruments avant la simulation

- dans la barre des menus, choisir *Instrumentation – Animation Mode*.

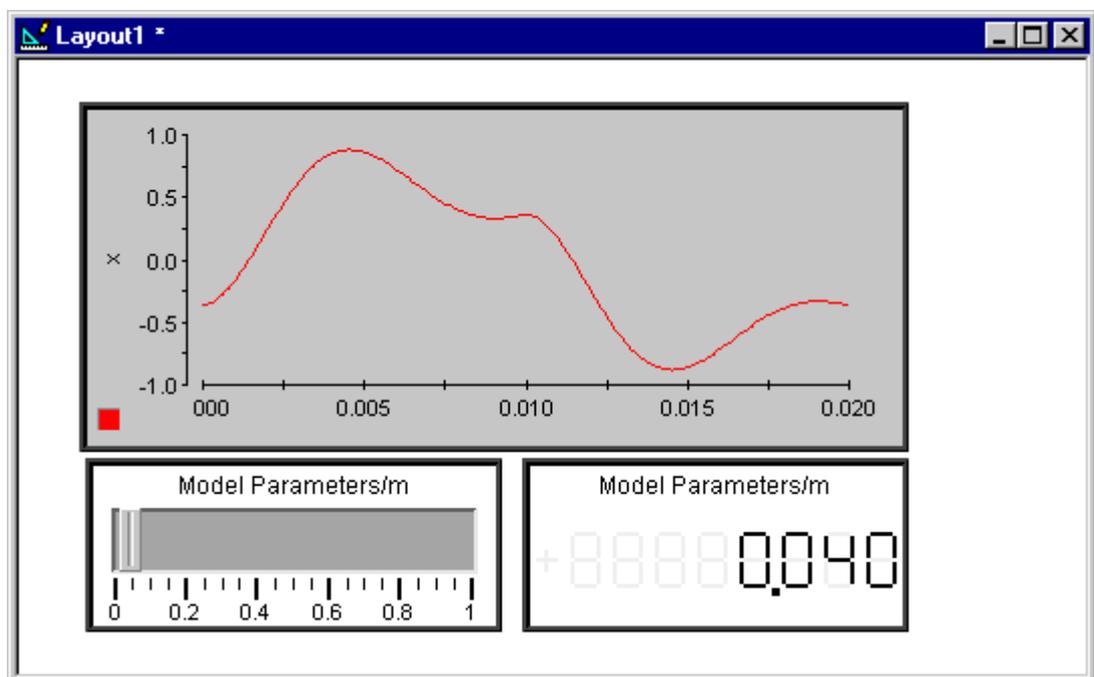


Figure III. 37 Panneau d'instruments après la simulation

- sélectionner le glisseur (*slider*), et modifier la valeur de la variable (0,10)

La forme du signal devient

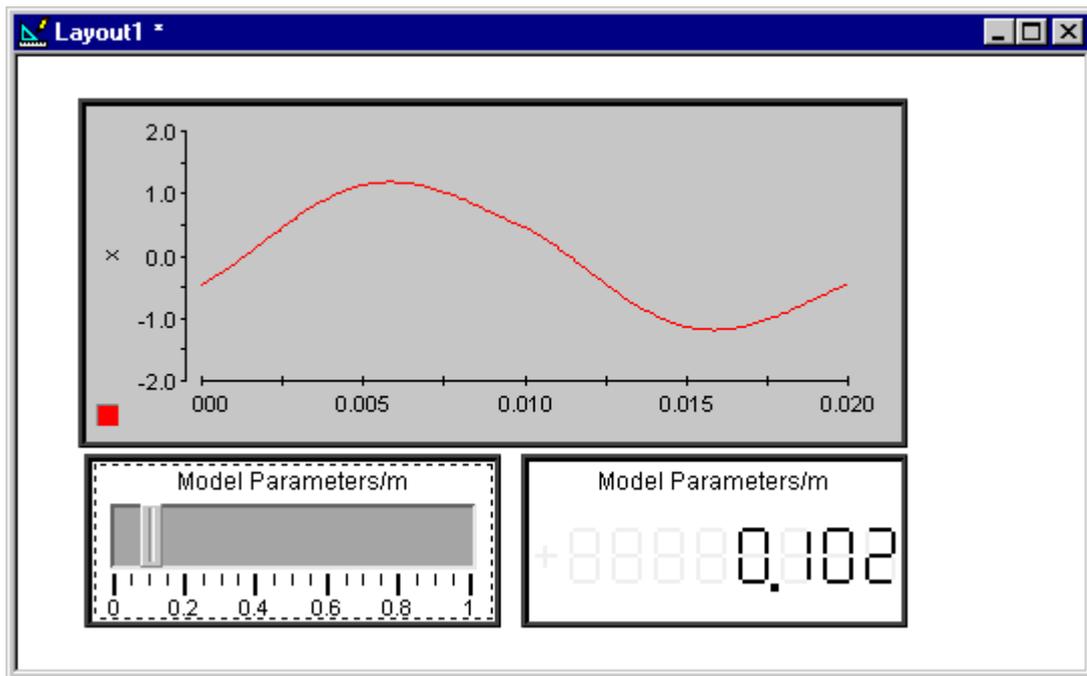


Figure III. 38 Modification de la valeur de la variable

III.5.2 Capture des données

ControlDesk permet de capturer les données des applications en temps réel ou des simulations Simulink. Ces données sont sauvegardées et contrôlées par les instruments d'acquisition de données.

Avec *time stamping* chaque valeur capturée est assignée à un temps. Ainsi, on peut afficher la valeur en fonction du temps de capture.

La fenêtre d'arrangements de capture (*Capture Setting Window*) est utilisée pour commander la saisie de données (data capture) dans ControlDesk. Il contient une ou plusieurs pages: le CaptureSettings instruments.

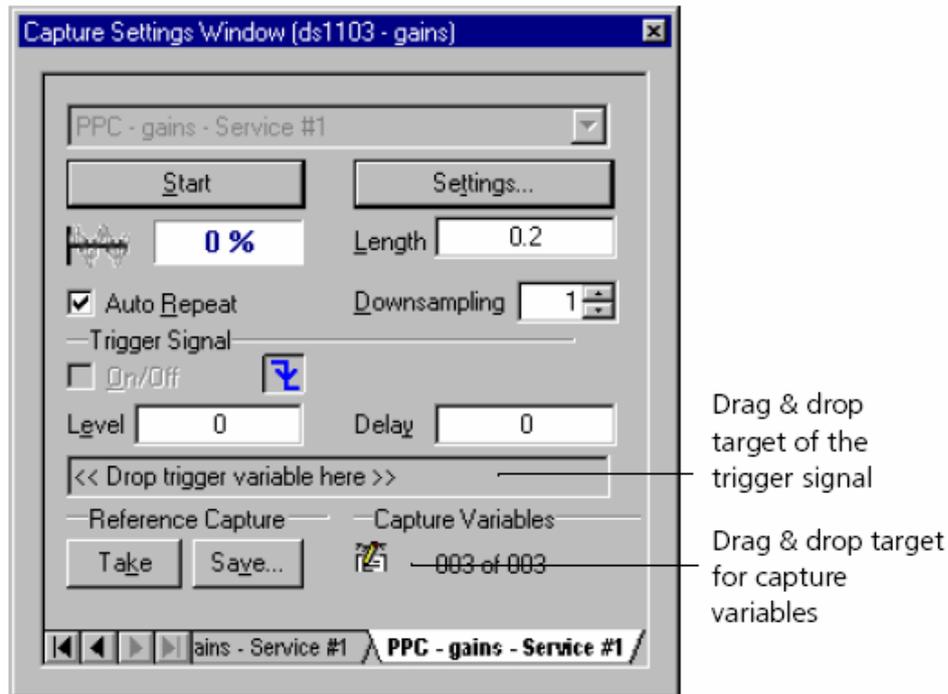


Figure III. 39 Fenêtre d'arrangements de la capture

La fenêtre *Capture Setting Window* peut être affichée en utilisant le menu contexte de la barre de control de ControlDesk.

Le *CaptureSettings instrument* permet de placer des paramètres de saisie de données pour les différents services indiqués dans le modèle. Chaque étiquette dans la fenêtre *Capture Setting Window* correspond à un service.

Les arrangements de chaque *CaptureSettings instrument* peuvent être appliqués à n'importe quel instrument d'acquisition de données.

Choix de capture pour l'acquisition des données

- dans la fenêtre variable browser, cliquer sur le bouton de la capture voulue.

Le dialogue *Capture Setting Window* est affiché avec le service correspondant. Chaque bouton correspond à un service qui est spécifié dans l'application ou la simulation Simulink.

Choix d'une variable pour la saisie de données (data capture)

- dans la fenêtre variable browser, choisir une variable pour la capture de données,
- glisser et déposer la variable sur l'icône au-dessous de *Capture Variables* dans l'instrument de *CaptureSettings*.

Vérification des variables connectées et utilisées

- dans *CaptureSettings instrument*, cliquer sur *Setting* pour ouvrir le dialogue *CaptureSettings Control Properties*.

- dans le dialogue *CaptureSettings Control Properties*, sélectionner la page de *CaptureVariables*.

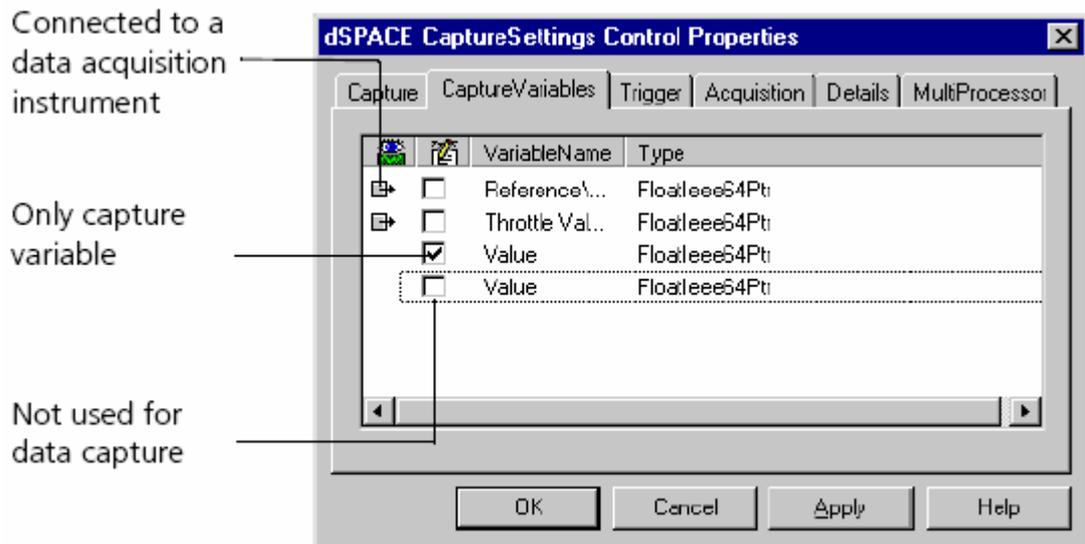


Figure III. 40 Page de capture de variables

Les variables de simulateur connectées aux instruments d'acquisition de données peuvent être identifiées par l'icône dans la première colonne (connectés). Les raccordements de données invalides (la source ou la cible de raccordement de données n'existe pas) sont affichés en rouge.

Si on choisit une seule variable de capture pour la saisie de données (data capture), aucune icône n'est affichée dans la première colonne. Pour utiliser les variables, sélectionner le *checkbox* dans la deuxième colonne (mesure) (voir la figure ci-dessus).

III.5.2.1 Comment capturer des données

La saisie de données (data capturing) commencera juste après que l'animation est lancée, si le mode automatique (Auto start) est activé à la page *capture property* dans le dialogue *CaptureSettings Control Properties*. Autrement, on peut utiliser le bouton *Start/Stop* pour démarrer et arrêter la capture manuellement.

Choix du signal de déclenchement (trigger signal)

- dans la fenêtre variable browser, glisser et déposer le signal désiré sur l'icône *Trigger Signal* dans la fenêtre *CaptureSettings l'instrument*.
- spécifier les arrangements du signal de déclenchement (trigger signal), c'est-à-dire, le retard (*delay*), front montant ou descendant, et niveau de déclenchement (*level*).

Start/stop data capturing

- dans la fenêtre *Capture Setting Window*, cliquer sur le bouton *Settings* pour ouvrir le dialogue *CaptureSettings Control Properties* et pour choisir la page *Capture*,
- dans la page *Capture*, sélectionner *Auto start with animation* pour commencer la saisie de données dès que l'animation est lancée,

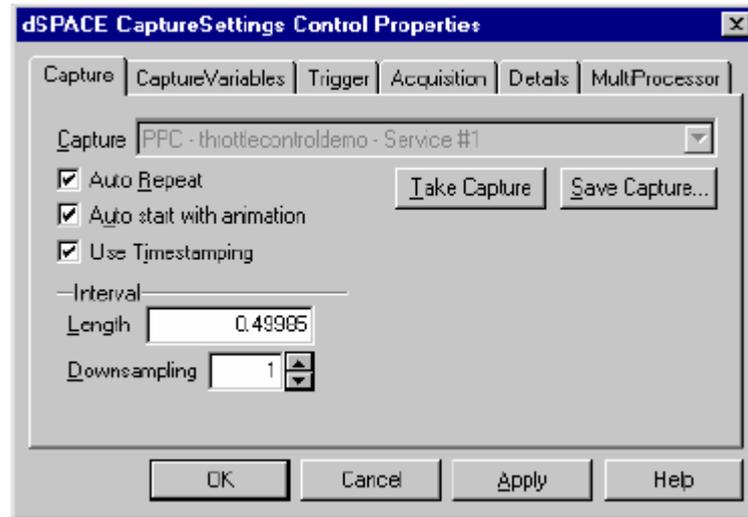


Figure III. 41 Page de capture

- dans la fenêtre *Capture Setting Window*, cliquer sur *Start/Stop* pour démarrer ou arrêter la saisie de données (*data capturing*) manuellement.
Auto Repeat permet de répéter la saisie de données (*data capturing*) automatiquement.

Signal de déclenchement (trigger signal)

La page de déclenchement (*Trigger*) dans le dialogue *CaptureSettings Control Setting* nous permet de spécifier les arrangements du signal de déclenchement (retard, front montant ou descendant, niveau), elle permet aussi d'activer ou de désactiver le signal de déclenchement.

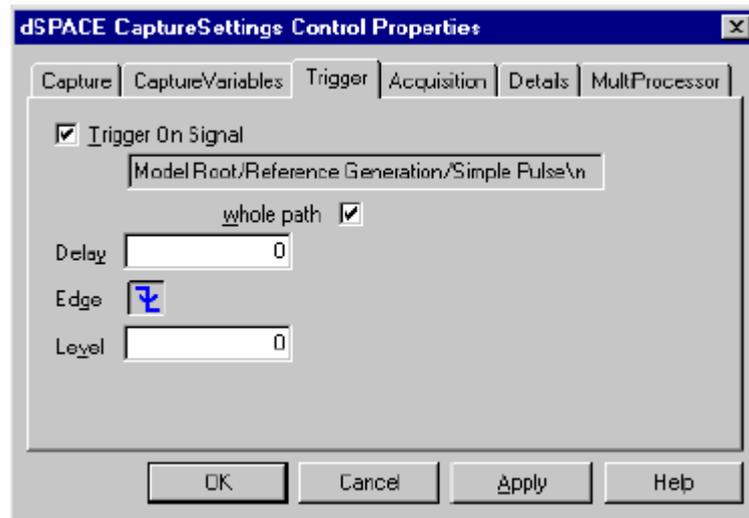


Figure III. 42 Page de déclenchement (Trigger)

III.5.2.2 La page de détails

Dans la page détails, on trouvera des informations sur la carte, les fichiers de description de système utilisés.

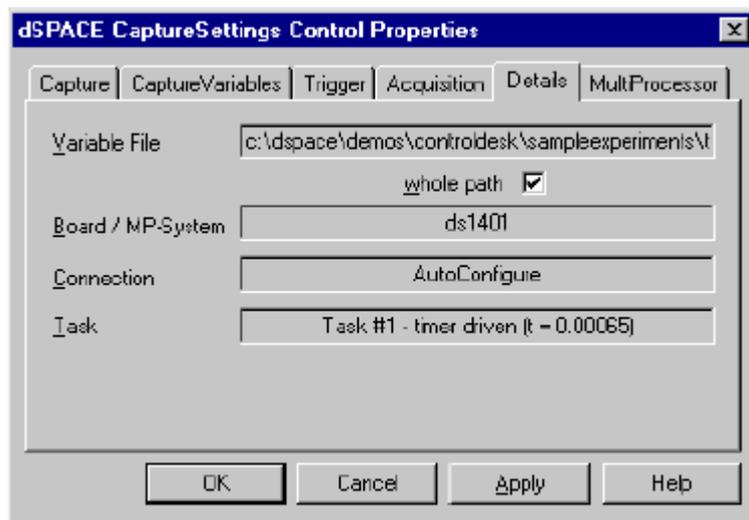


Figure III. 43 Page de détails

III.5.2.3 La page acquisition (Acquisition)

La page d'acquisition nous permet de spécifier le mode de stockage.

Mode simple exécute une saisie de données simple. Ce mode capture seulement les petites périodes des signaux. Même si on choisit la répétition automatique (Auto repeat), deux captures consécutives sont indépendantes. Par exemple, il peut y avoir un retard entre elles.

Mode autosave exécute une saisie de données simple et sauvegarde automatiquement les données dans un fichier spécifique (toujours le même fichier).

Mode Autaname exécute une saisie de données simple et sauvegarde les données automatiquement. Chaque saisie de données est sauvegardée dans un dossier différent. Les noms de fichier sont générés automatiquement.

Mode continu (continuous) capture des données continuellement. Le mode continu exécute une saisie de données sans fin en utilisant un buffer cyclique. L'utilisation du champ *length* dans le dialogue *CaptureSettings instrument* permet de spécifier la durée de la capture et la taille de buffer cyclique sur la plateforme de simulation (la valeur doit être supérieure à 2 secondes).

Mode Stream To Disk utilisé pour écrire les données capturées sur le disque dur.

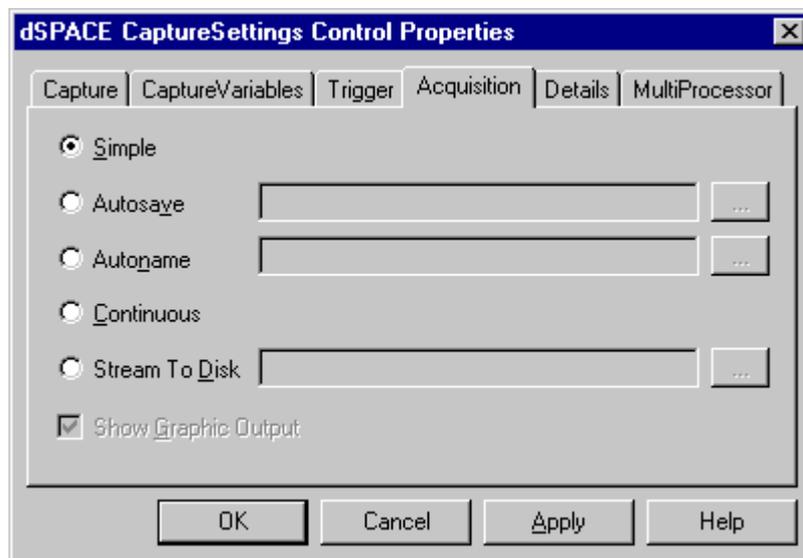


Figure III. 44 Page d'acquisition

III.5.3 Utilisation de données de Référence

Les données de référence sont utilisées pour comparer les signaux de données d'une expérience avec des données des expériences anciennes ou avec des signaux calculés dans MATLAB.

La fenêtre *Reference Data Manager* peut lire les fichiers de données générés par le mode *Stream To disk*. Cette fenêtre est accessible dans l'outil Window de ControlDesk, elle est affichée après le chargement de données de référence.

Les fichiers de données générés par le Reference Data Manager

Pour la sauvegarde des données capturées, ControlDesk dispose de deux formats différents de fichier de données :

Format de fichier CSV

Le *format comma – separated values* (CSV) est un format ASCII qui peut être employé pour stocker des données de référence.

Format de fichier MAT

Ce format binaire peut être utilisé pour stocker des données de référence pour un traitement avec MATLAB. ControlDesk génère des fichiers de données de référence compatibles avec le format binaire des versions MATLAB 5,2, 5,3 et 6,0.

III.5.3.1 Génération et sauvegarde des données de référence

Une fois que la capture des variables en temps réel est complète, les données peuvent être utilisées comme signal de référence. On distingue deux modes de sauvegarde des données.

- Mode de saisie simple (*single capture mode*)

Dans ce mode les données de référence peuvent être sauvegardées. La durée de la saisie de données (*data capture*) est déterminée par la valeur de la longueur (*length*) dans le dialogue *Capture Setting Window*.

Pour générer et sauvegarder les données de référence d'une saisie simple

- charger une application et un panneau d'instruments contenant des instruments d'acquisition de données,
- connecter les variables désirées de la fenêtre variable browser à l'instrument d'acquisition de données,
- dans le *CaptureSettings* de l'instrument, indiquer les arrangements de la saisie de données selon les conditions voulues. Sélectionner le mode simple dans la page d'acquisition,
- dans la barre des menus, choisir *Instrumentation – Animation Mode*,
- compléter la saisie de données en temps réel,
- dans la fenêtre *Capture Settings Window*, cliquer sur le bouton *Take* pour générer un nouvel ensemble de données de référence.

Cette action ajoute un nouvel arbre (*tree*) de données capturées à la page de *Rference Data Manager*,

- dans le menu de contexte, sélectionner *Save* pour sauvegarder les données de référence comme fichier CSV ou MAT.

- *Mode continu (continuous mode)*

Dans ce mode, les données de référence couvrant une plus grande période peuvent être sauvegardées en utilisant *Stream To Disk*.

Pour générer et sauvegarder les données de référence (continues) via le dispositif disque Stream To Disk

- charger une application et un panneau d'instruments contenant des instruments d'acquisition de données,
- cliquer sur le bouton *Settings* pour ouvrir le dialogue *CaptureSettings Control Properties*, et sélectionner la page d'acquisition,
- dans la page d'acquisition, choisir le *Stream To Disk*, et spécifier le nom de fichier,
- dans la barre des menus, sélectionner *Instrumentation – Animation Mode* et activer la fenêtre *Capture Settings Window*,
- dans la fenêtre *Capture Settings Window*, cliquer sur le bouton *start* pour démarrer la saisie de données en temps réel.

Remarque :

L'exécution de *Stream To Disk* doit être arrêté manuellement pas le bouton *stop* dans la fenêtre *Capture Settings Window*.

III.5.3.2 Affichage des données de référence

Les données des signaux de référence (montrées dans la fenêtre *Reference Data Manager*) peuvent être affichées de la même manière que celles des signaux en temps réel.

Pour afficher des données de référence on procède comme suit

- dans la barre des menus, sélectionner *file – Load Reference Group*,
- dans ce dialogue, choisir *file* et le cliquer sur *Open*,

Un nouvel arbre (*tree*) de référence sera ajouté au *Reference Data Manager*,

- sélectionner le signal de référence désiré dans la liste des variables de *Reference Data Manager* et le déplacer avec glisser et déposer dans l'instrument d'affichage (exemple *Plotter*).

L'illustration suivante montre le signal de référence avec la variable correspondante.

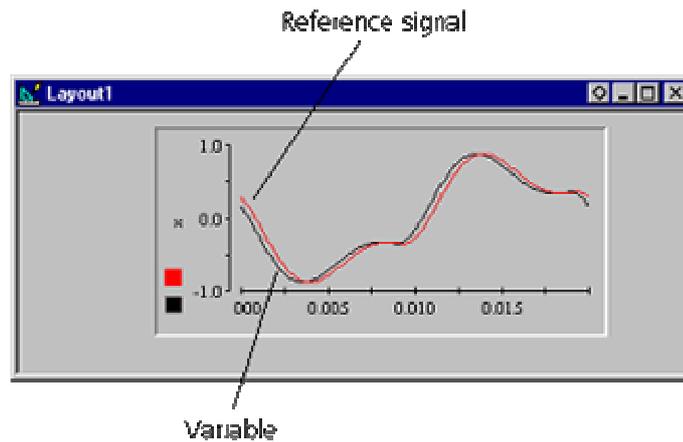


Figure III. 45 Signal de référence avec la variable correspondante

III.6 Expérimentation sur la plateforme de Simulink [7] [9]

Avec le contrôle intégré des simulations Simulink, nous pouvons acquérir des données et valider nos modèles sur le PC.

III.6.1 Conditions de système pour des simulations de Simulink

La simulation des modèles Simulink sur le PC exige une puissance de calcul importante pour le calcul dans MATLAB et la visualisation dans ControlDesk. Ainsi, les simulations Simulink ne sont pas effectuées en temps réel.

Recommandation matérielle

Le matériel suivant est recommandé:

- processeur, au moins 450 MHz,
- mémoire 256 MB de RAM.

Recommandation logicielle

Pour un usage correct des simulations Simulink, ControlDesk exige que les logiciels suivants soient installés:

- l'un des systèmes d'exploitation suivants
 - Windows NT 4.0 (service pack 6a ou plus),
 - Windows 2000.
- MATLAB 5,3 (R11) et Simulink 3,0 (ou plus).

III.6.2 Manipulation des simulations de Simulink dans ControlDesk

La manipulation des simulations Simulink dans ControlDesk est identique aux manipulations des modèles dans Simulink.

Création d'un modèle scratch

Si aucun modèle n'est disponible (chargé) dans la carte, on peut appeler Simulink et créer un nouveau modèle dans ControlDesk via le navigateur de plateforme. Ce modèle est appelé automatiquement et contient quelques blocs simples.

Pour créer un nouveau modèle

- dans le navigateur de plateforme, ouvrir le menu contexte de la plateforme de Simulink et sélectionner *New Model*,
- sauvegarder le modèle,

La fenêtre de MATLAB/Simulink s'ouvre,

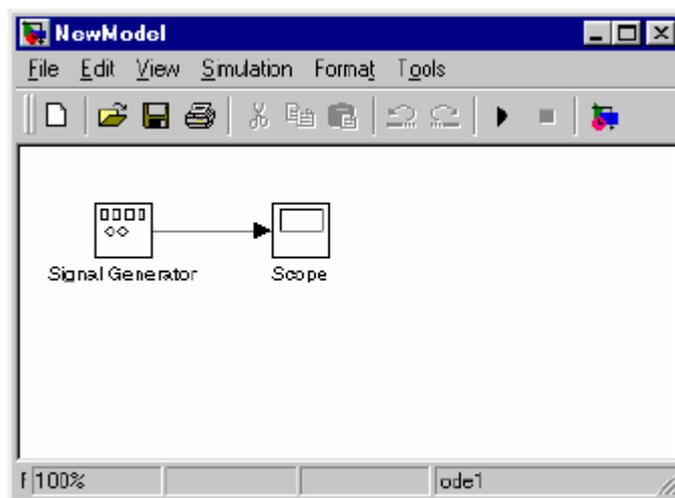


Figure III. 46 Nouveau modèle Simulink

- dans le menu du modèle Simulink, sélectionner *Simulation - parameter* et modifier les paramètres de la simulation selon les besoins de l'application.

Chargement d'un modèle existant

ControlDesk dispose de différentes méthodes pour charger un modèle. On peut employer les méthodes suivantes pour charger un modèle:

- glisser et déposer le fichier de description de système (SDF) ou le fichier modèle (MDL) de sélecteur de fichier (file selector) à la plateforme de Simulink dans le navigateur de plateforme,
- cliquer sur le bouton *load Application/Model* dans la barre d'outils (toolbar) de la *plateforme manager* et sélectionner le fichier à charger,

- ouvrir le menu contexte d'une variable (variable tree) et cliquer sur Assign Platform pour lancer MATLAB et charger le modèle associé.

Comment préparer le modèle et changer l'état de simulation

Bien qu'on puisse commencer une simulation Simulink après que le modèle soit chargé, on ne peut pas surveiller les signaux et contrôler les paramètres dans ControlDesk jusqu'à ce que le fichier de description système associé à ce modèle soit chargé. Pour surveiller les signaux, on doit réaliser d'abord les étapes préparatoires suivantes.

- Charger un fichier existant (fichier description système) ou construire un nouveau en choisissant *Build Variable Description File* à partir du menu contexte de la plateforme Simulink,

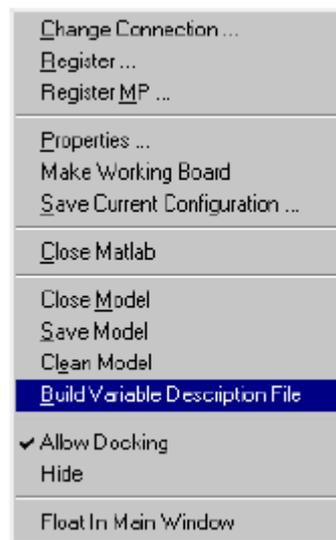


Figure III. 47 Menu contexte de la plateforme Simulink

- Ouvrir un panneau d'instruments existant ou créer un nouveau et connecter les variables de la simulation Simulink aux instruments.
- basculer en mode animation (*animation mode*) pour surveiller la simulation dans ControlDesk.

Pour commencer la simulation, cliquer sur le bouton *Reload Application/Start Simulink Simulation* dans le directeur de matériel (*Hardware Manager*) ou sélectionner *platform - Application – Start Simulink Simulation* dans la barre des menus.

Les panneaux d'instruments sont devenus opérationnels et les connexions de données sont activées.

Pour arrêter la simulation, cliquer sur le bouton *StopRTP/Simulink Simulation* dans la *plateforme Manger* ou sélectionner *platform - Application – Stop Simulink Simulation* dans la barre des menus.

Conclusion

Dans ce chapitre nous avons présenté les éléments de base du logiciel de la carte dSPACE ControlDesk qui nous permet l'implémentation des modèles Simulink et l'affichage des résultats de la simulation.

Chapitre IV

L'interface en temps réel RTI

Introduction

Dans le chapitre précédent on a décrit le logiciel Controldesk de la carte DS1104 et on a vu que sa programmation est faite par Simulink, le logiciel qui accompagne MATLAB. Cependant, ces deux logiciels ne suffisent pas et ils nécessitent un autre logiciel qui réalise le lien entre eux, qui est le RTI.

IV.1. Définition de RTI [3]

L'interface en temps réel (RTI) pour les systèmes de dSPACE est un lien entre Simulink et le matériel de dSPACE.

RTI permet d'ajouter au modèle de Simulink des interfaces I/O, des interruptions...etc et d'établir le code en temps réel, de télécharger et d'exécuter le sur le matériel de dSPACE.

RTI-mp (RTI multiprocessor) permet de partager le modèle d'un système multiprocesseur et d'affecter les parties du modèle aux différentes CPUs.

IV.2. Les étapes de développement

Pour développer en industrie un système de commande ou de contrôle, en utilisant le matériel de dSPACE, il y a quatre étapes de développement

IV.2.1. Contrôle de la conception et la simulation du modèle de Simulink

Dans cette étape on utilise seulement le Simulink. La caractéristique principale de ce type de simulation est que l'ordinateur a autant de temps que nécessaire pour exécuter le modèle de Simulink.

Ainsi, si le modèle est simple, les résultats peuvent être calculés rapidement. Si le modèle est complexe, beaucoup plus de temps est nécessaire pour effectuer les calculs.

IV.2.2. Prototypage rapide de contrôle (RCP rapid control prototyping)

La situation est différente une fois que le modèle de contrôle simulé répond aux espérances et on veut l'examiner sur le matériel réel. Puisque le modèle pourrait avoir besoin d'autres modifications, on ne peut pas le réaliser à cette étape du développement.

Par conséquent, on doit connecter le matériel réel au modèle de contrôleur qui est simulé en temps réel en utilisant un panneau de connexion qui relie le matériel avec la carte dSPACE à l'aide de son port de sortie. Cette technique s'appelle rapid control prototyping (RCP) ou Prototypage rapide de contrôle.

La caractéristique principale de cette simulation en temps réel est qu'elle doit être effectuée aussi rapidement que le vrai système fonctionnerait réellement, ce qui permet de combiner la simulation et le matériel réel.

Pour le RCP, la simulation en temps réel est assez importante. La puissance de calcul exigée par la simulation en temps réel dépend fortement des caractéristiques du modèle simulé: s'il contient des calculs très exigeants, on doit fournir beaucoup de puissance de calcul car la synchronisation entre le modèle simulé et le matériel réel ne peut pas être satisfaite sans cela. Les systèmes de dSPACE accomplissent cette demande de puissance de calcul.

IV.2.3. Hardware In the Loop HIL

Quand le contrôleur simulé peut contrôler le matériel, on produit typiquement le contrôleur réel. Pour les tests finaux on connecte habituellement le vrai contrôleur à un modèle du matériel, qui doit être simulé en temps réel. De cette façon on peut assurer que le contrôleur ne contient aucune erreur qui pourrait endommager le matériel. Cette technique s'appelle Hardware in The Loop (HIL).

IV.2.4. Génération automatique du code C

Pour les systèmes de dSPACE, l'interface en temps réel (RTI) et Real-Time Workshop du MathWorks effectuent cette fonction. Le code en temps réel des modèles de Simulink est produit automatiquement et implémenté sur le matériel de dSPACE.

RTI effectue cette tâche ; on doit seulement ajouter les blocs de dSPACE (par exemple, des interfaces I/O) au modèle de Simulink.

IV.3. Travailler avec RTI et RTI-mp [5]

IV.3.1. Activation d'une carte spécifique

Si on a plusieurs cartes de dSPACE, par exemple DS1104 DS1102 DS1005, il faut en activer une et une seule, car on ne peut pas travailler avec plus d'une carte à la fois, et chacune a sa propre bibliothèque.

Pour lancer RTI avec une carte particulière,

- quand MATLAB est démarré, RTI affiche un dialogue qui donne le choix de la carte désirée,



Figure IV.1 Sélection d'une carte spécifique

- si on choisit 'Do not show this dialog again', MATLAB commencera toujours par la dernière carte activée,

Pour activer une carte différente,

- dans la fenêtre de MATLAB on tape `rti<XXXX>` pour lancer la carte désirée (on remplace `rti< XXXX >` par sa référence par exemple, `rti1103`). La bibliothèque correspondante de RTI s'ouvre automatiquement.

IV.3.2. L'accès aux blocs de RTI

Les divers blocs de RTI permettent d'accéder au matériel de dSPACE.

Il y a deux voies pour obtenir un bloc de RTI: on peut utiliser le browser de la bibliothèque de Simulink, ou la bibliothèques des blocs de RTI.

MATLAB Launch Pad fournit des raccourcis aux diverses bibliothèques des blocs, aux bibliothèques des démonstrations, l'aide en ligne et des informations sur l'Internet.

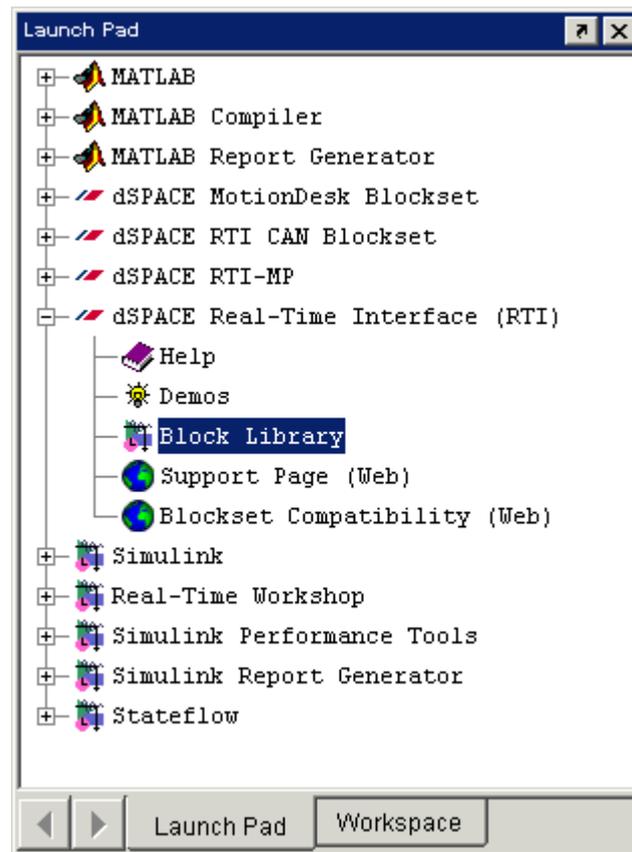


Figure IV.2 MATLAB Launch Pad

- Pour accéder à un bloc de RTI par la bibliothèque de RTI :
 - 1- on ouvre la bibliothèque de RTI en tapant 'rti' dans la fenêtre de MATLAB, et puis on ouvre la sous bibliothèque correspondante, et on choisit le bloc,
 - 2- on ajoute le bloc au modèle de Simulink.

L'illustration suivante montre la bibliothèque de dSPACE RTI1104:

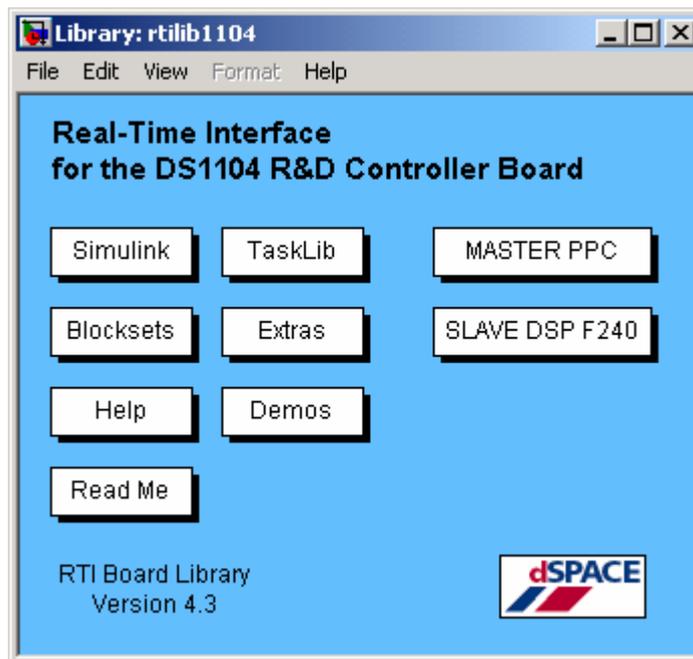


Figure IV.3 Bibliothèque de RTI 1104

- Pour accéder à un bloc de RTI par le browser de la bibliothèque de Simulink :
 - 1- on ouvre le browser par le bouton de Simulink qui se trouve dans la barre d'outils de la fenêtre MATLAB,
 - 2- on ouvre la bibliothèque correspondante de dSPACE, et on choisit le bloc,
 - 3- on ajoute le bloc au modèle de Simulink.

L'illustration suivante montre la bibliothèque de dSPACE RTI1104 dans le browser de bibliothèque de Simulink:

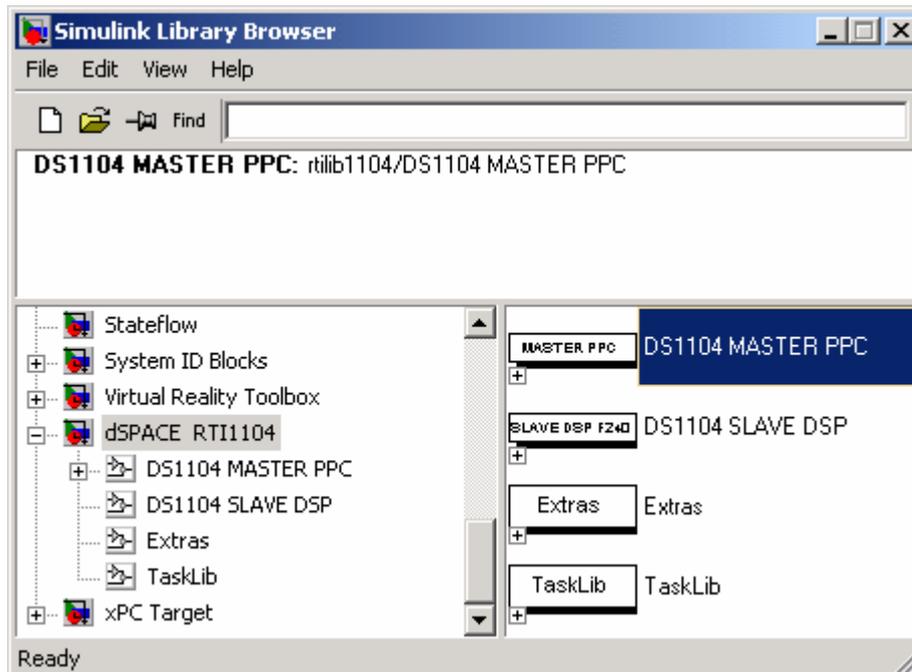


Figure IV.4 La bibliothèque de dSPACE RTI1104 dans le Simulink

IV.3.3. Les valeurs par défaut des paramètres de simulation

Chaque nouveau modèle de RTI a des paramètres de construction et de simulation qui sont placés en valeurs par défaut. Pour certains paramètres et options (tels que les configurations de Solutionneur ou Solver) on peut définir les valeurs par défaut dans le dialogue des préférences de Simulink. Les valeurs par défaut initiales, qui sont placées à l'installation de MATLAB, ne sont pas appropriées aux applications en temps réel.

Pour les modifier :

- 1- ouvrir le menu fichier de MATLAB et choisir 'préférences',
- 2- ouvrir le noeud de Simulink dans l'arbre de préférences, et choisir 'simulation',
- 3- indiquer les configurations désirées. Pour la conception des applications en temps réel avec RTI, on devrait indiquer les valeurs par défaut suivantes dans la page 'solver' :

Setting	Value
Simulation time frame	
Start time	0.0
Stop time	inf
Solver options frame	
Type	"ode1"
Mode	"SingleTasking"
Fixed step size	0.001 ... 0.01

Tableau IV.1 Les valeurs par défaut de RTI

- Fixed step size: Pour indiquer le temps d'échantillonnage du modèle.
- Solver type: c'est le type de solveur des équations du modèle, dans ce cas 'ode1' c'est la méthode d'EULER qui est appropriée pour le plus part des applications.
- Solver mode: Pour spécifier le mode d'exécution du modèle.

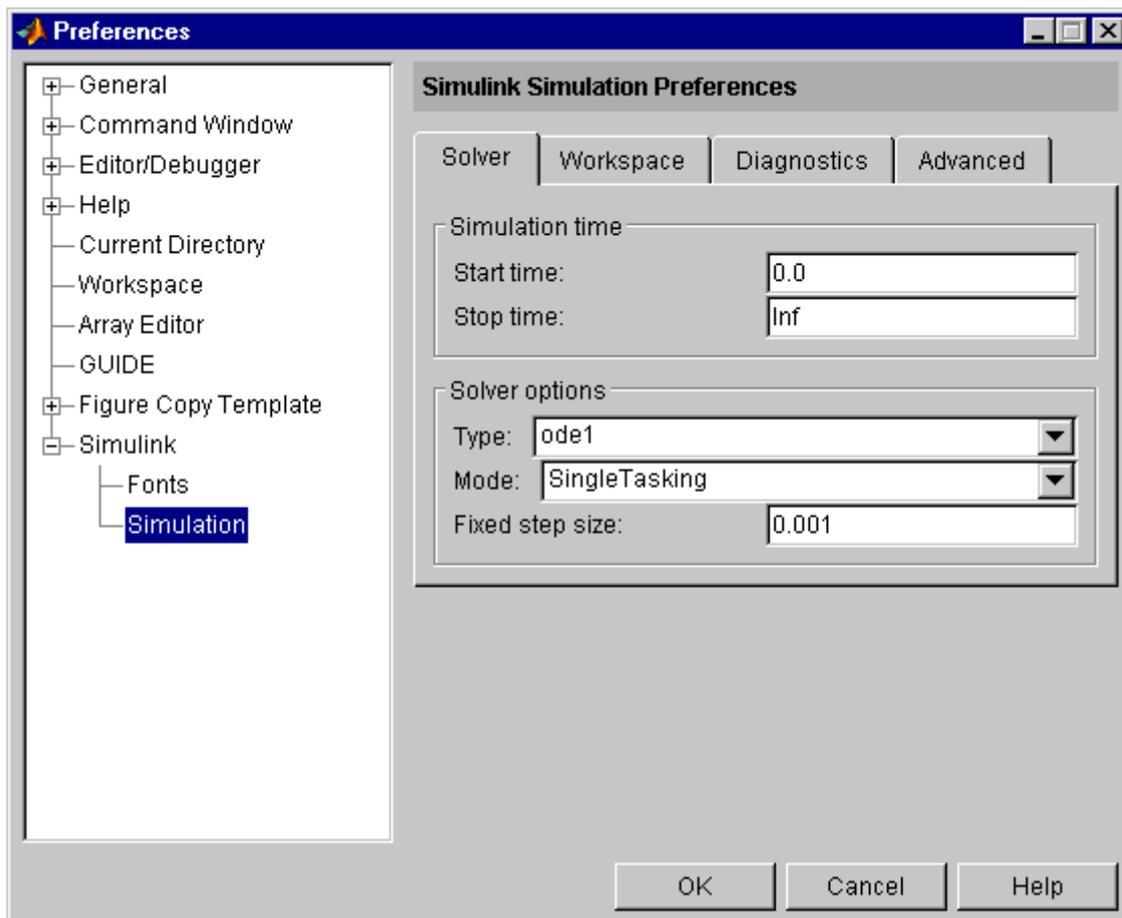


Figure IV.5 Simulink Simulation Preferences

4- Dans la page 'Advanced' décocher l'option 'Block reduction', car cette option n'est pas supportée par RTI. Autrement, la marche à suivre de construction échouera.

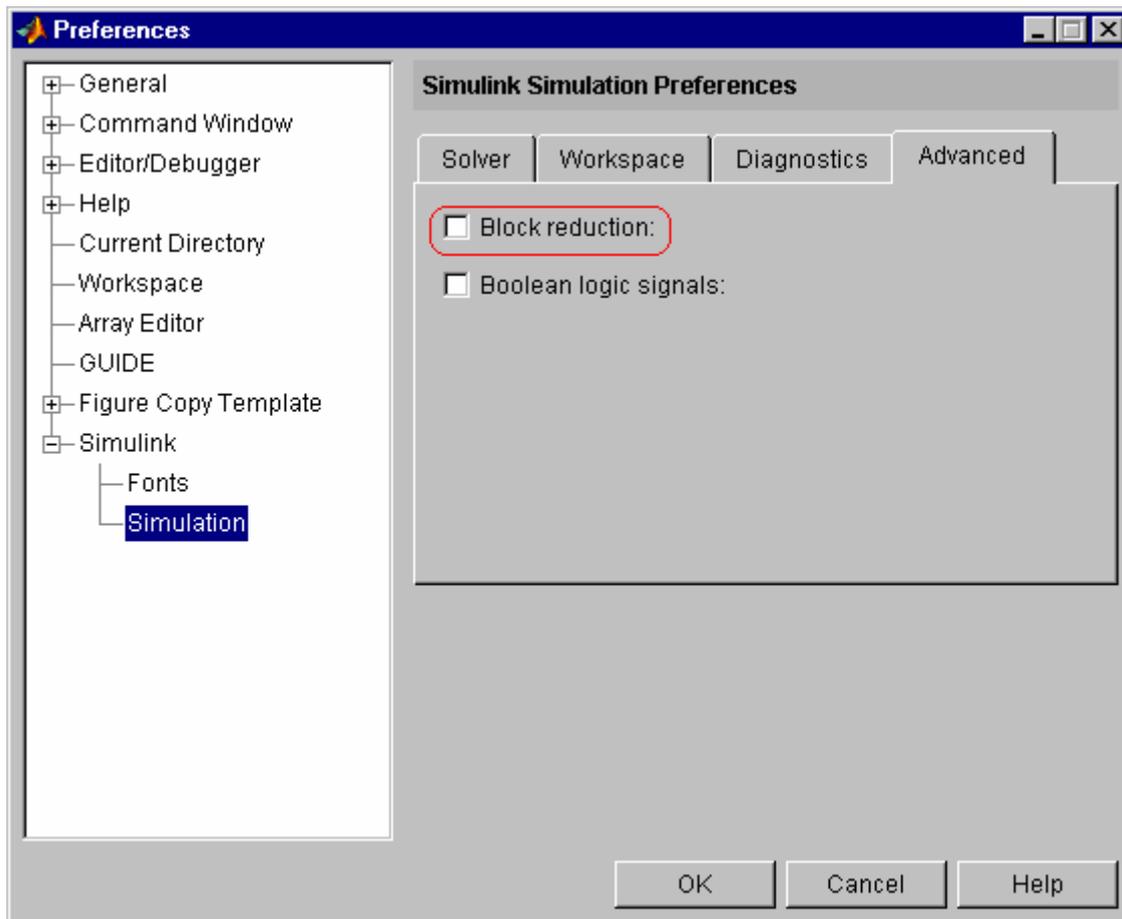


Figure IV.6 Désactivation de l'option 'Block reduction'

IV.4. Les interfaces d'entrée/sortie I/O [3]

Pour connecter la simulation en temps réel à l'extérieur, on doit introduire des interfaces I/O dans le modèle. Ils permettent de remplacer des parties du modèle simulé par le matériel réel.

IV.4.1. Les conventions de la nomenclature

- Les noms de bloc commencent toujours par le nom de la carte.
- On ajoute la description de leur fonctionnalité.
- Dans les systèmes modulaires, le suffixe B et le numéro de la carte sont ajoutés.
- Pour MicroAutoBox, le suffixe M et le numéro de module sont ajoutés.
- Il y a d'autres suffixes: G indique un numéro de groupe, CON un numéro de convertisseur, P un numéro de port.

Si le nom du bloc ne correspond pas aux conventions de dSPACE, il est affiché en italiques. L'illustration suivante montre quelques exemples des noms des blocs I/O:

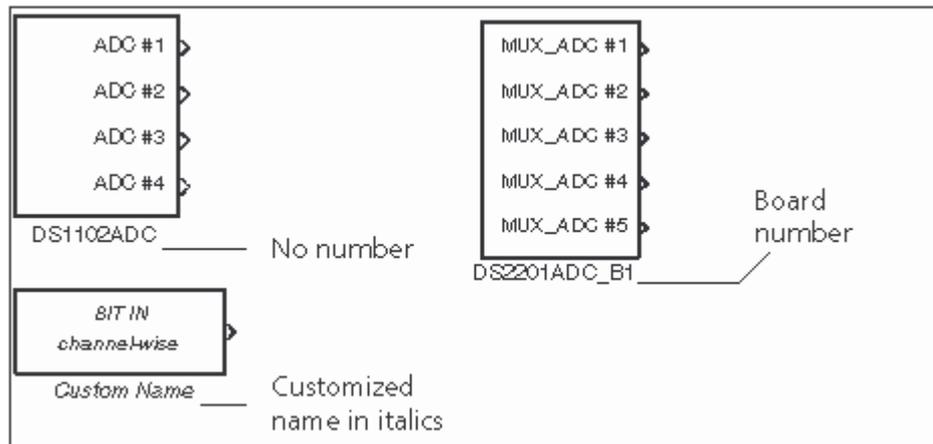


Figure IV.7 Exemples des noms des blocs I/O

IV.4.2. L'ajout des blocs I/O

Si on veut remplacer des blocs simulés par les blocs matériels correspondants on doit ajouter des blocs I/O au modèle.

Pour ajouter des blocs I/O au modèle, on procède comme suit

- 1- Dans le browser de la bibliothèque de Simulink, on montre la bibliothèque de dSPACE des I/O qui correspond à l'unité I/O qu'on veut utiliser.
- 2- On ajoute le bloc approprié I/O au modèle de Simulink.
- 3- On connecte le bloc au modèle avec n'importe quel bloc standard de Simulink.
- 4- On édite les configurations du bloc pour répondre aux exigences du modèle.

On peut ouvrir le dialogue des paramètres du bloc par un double click sur le bloc.

Si on laisse des ports inutilisés ouverts, Simulink produit un message d'avertissement par défaut.

Pour éviter des avertissements dus aux ports non liés, on les connecte tous aux blocs Ground ou Terminator, qui peuvent être trouvés dans la bibliothèque des systèmes et des signaux de Simulink.

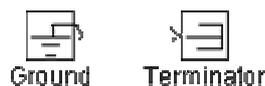


Figure IV.8 Les blocs Ground et Terminator

IV.4.3. Types de données

RTI utilise les types de données suivants:

- Les blocs I/O à un seul bit utilisent le type de données booléennes. Par conséquent, on peut les connecter directement à tous les opérateurs logiques.
- Les blocs I/O à un octet utilisent le type de données uint8 (Unsigned 8-bit integer).
- Les encodeurs incrémentaux utilisent le type de données int8 (Signed 8-bit integer).
- Les blocs DAC et ADC utilisent les données de type double.

IV.5. Création d'un nouveau bloc [5]

RTI est conçu pour fournir une manipulation des blocs I/O des cartes dSPACE à partir d'un modèle de Simulink. Cependant, quelques blocs particuliers ne sont pas disponibles dans RTI. Dans ce cas, on peut construire ce bloc par une S-fonction (Simulink function) ou par User-Code de RTI.

La S-fonction est un dispositif de Simulink, User-Code est un dispositif semblable, mais fourni par RTI.

IV.5.1. Implémentation de S-Function

IV.5.1.1. Les Bases de S-Functions

Les S-Functions sont écrites dans différents langages tels que M, C ou Fortran. Si on veut utiliser une S-Function avec un système de dSPACE, elle doit être écrite en C seulement car Real-Time Workshop et RTI ne supportent pas de S-fonctions en M, C++ ou Fortran.

IV.5.1.2. Composants des S-functions

Pour utiliser une S-Function dans un modèle, on doit suivre ce qui suit:

- Le fichier source de S-Functions doit être écrit en utilisant un ensemble de fonctions normalisées de C, par exemple, mdlStart().
 - Le fichier de MEX DLL (également appelé MATLAB exécutable) est compilé à partir du fichier source de S-fonction. On doit toujours compiler un fichier à jour de MEX DLL après avoir changé le fichier source de S-fonction.
 - Real-Time Workshop et Real-Time Interface utilisent les fichiers source pendant le processus de compilation du modèle.

- Un fichier à jour de MEX DLL est exigé.
 - Simulink utilise le fichier DLL pour la simulation.
 - Real-Time Workshop et Simulink emploient le fichier DLL pour connaître certaines propriétés de la S-functions, par exemple, les dimensions des signaux et le temps d'échantillonnage.
- Un bloc de S-Function doit être inséré dans le modèle de Simulink et ses paramètres de bloc doivent indiquer son nom et ses paramètres.

IV.5.1.3. Le modèle de S-Function

Puisque toutes les S-functions doivent avoir le même ensemble de sous-programmes, on devrait toujours commencer la programmation avec un modèle de S-Function.

- Si on veut créer une nouvelle S-function, on prend le fichier `sfuntmpl_basic.c` trouvé dans le dossier de MATLAB `< MATLAB_root >\simulink\src\`.
- Si on veut modifier une S-function spécifique de RTI, on prend le fichier désiré de la S-function, trouvé dans le dossier de RTI
`%dspace_root%\MATLAB\RTI<XXXX>\SFcn\`.
XXXX représente le nom de la carte, dans notre cas c'est 1104.

Les S-functions initiales sont restaurés et mis à jour pendant n'importe quelle nouvelle installation. Ainsi avant qu'on modifie n'importe quelle S-function, on devrait la copier au dossier du travail qui retient le modèle et on la renomme. Elle est alors disponible pour tous les modèles situés dans ce dossier.

Pour obtenir un modèle de S-function

- 1- On copie le modèle désiré de S-function au dossier de travail qui retient le modèle de Simulink.
- 2- On choisit un nom pour la S-function et on renomme le fichier. On écrit le nom de la S-function (sans extension `.c`). Quand on configure le bloc de S-function pour le modèle de Simulink, on écrit le nom de la S-function (sans extension `de c`) dans le dialogue de paramètres de bloc.

IV.5.1.4. La construction manuelle de S-Function en code C

Si on a besoin d'une S-function spécifique qui n'est pas basée sur une S-function prédéfinie par RTI, on doit la créer à partir de zéro en utilisant le fichier `sfuntmpl_basic.c` de Simulink.

On suit les instructions données ici pour éviter des difficultés communes en implémentant des S-functions.

IV.5.1.4.1 Rôle des fonctions C

Une S-function se compose d'un ensemble fixe de fonctions de `mdl<...>()` qui s'appellent par les fonctions du modèles `Mdl<...>()`, avec lesquelles, Real Time Workshop produit le modèle global.

mdlStart() contient le code d'initialisation de la S-function, qui est exécutée une fois quand la simulation est commencée.

mdlInitializeConditions() place les valeurs initiales des variables d'état de la S-function.

On l'exécute une fois quand la simulation est commencée.

mdlInitializeSizes() définit les caractéristiques de la taille de S-function, telles que le nombre d'entrées, de sorties. Simulink lit les caractéristiques pour configurer le bloc de S-function.

mdlInitializeSampleTimes() définit les temps d'échantillonnage pour la S-function.

mdlOutputs() contient le code pour calculer les sorties du bloc de la S-function.

mdlUpdate() contient le code algébrique pour calculer les variables d'états discrètes de la S-function.

mdlDerivatives() contient le code pour calculer les dérivés des variables d'état continues.

mdlTerminate() contient le code d'achèvement de la S-function, qui est exécutée une fois que la simulation est arrêtée. On peut l'utiliser pour écrire les valeurs d'achèvement aux dispositifs de sortie.

IV.5.1.4.2 L'information de la taille

L'information de la taille est une information très essentielle pour une S-function, puisqu'elle définit le nombre d'entrées, de sorties, et le temps d'échantillonnage. Simulink et Real Time Workshop exigent cette information pour vérifier si le bloc de S-function a le nombre correct de ports. Par conséquent, on doit indiquer cette information en `mdlInitializeSizes()` et `mdlInitializeSampleTimes()`.

IV.5.1.5. La création du fichier MEX DLL

En plus du fichier source C d'une S-function, Simulink et Real Time Workshop exigent une version compilée de la S-function, c'est le fichier MEX DLL. Si on exécute une simulation de Simulink, Simulink emploie le fichier de MEX DLL pour calculer la S-function. Si on

produit un code en temps réel, Real Time Workshop a besoin du fichier de MEX DLL pour lire l'information de la taille de S-function.

On peut produire un fichier MEX DLL pour une S-function par avec la commande **mex** de MATLAB. Cependant, MATLAB doit être installé correctement avant qu'on puisse utiliser la commande **mex**.

Pour installer la commande MEX de MATLAB on tape la commande **mex -setup** et on suit les instructions sur l'écran. Quand MATLAB est prêt à compiler des fichiers MEX DLL, on peut produire le fichier MEX DLL pour une S-function comme suit:

➤ On tape `mex < source>.c -v` dans la fenêtre de MATLAB. On remplace la <source> par le nom de fichier source C.

L'option '-v' fait le compilateur de MEX affiche les options utilisées, et émettre tous les avertissements.

IV.5.2. Implémentation de User-Code

User-Code est un outil de RTI de dSPACE qui permet de créer un nouveau bloc de Simulink. Il peut être introduit dans le modèle de Simulink au moyen des blocs Data Store Memory.

IV.5.2.1. Comment utiliser User-Code dans les modèles de Simulink :

L'interface entre Simulink et User-Code est implémentée par l'intermédiaire des blocs Data Store de Simulink.

- Chaque bloc Data Store Memory définit et initialise une simple variable de communication.



- Le bloc Data Store Read représente la transition de User-Code au modèle de Simulink.



- Le bloc Data Store Write représente la transition du modèle de Simulink à User-Code.



S'il y a des blocs Data Store Memory dans le modèle de Simulink, on obtient un groupe séparé de Data Stores dans Variable browser de ControlDesk. Ce groupe permet d'accéder aux variables de communication de User-Code

IV.5.2.2. Pour inclure User-Code dans un model de Simulink

- 1- Pour chaque variable de communication, on met un bloc séparé de Data Store Memory dans le modèle. On peut trouver ce bloc dans la bibliothèque de connexions de Simulink.
- 2- Pour chaque variable de communication, on ouvre le dialogue de paramètres du bloc Data Store Memory. Dans la zone Data Store Name, on indique le nom de la variable de communication: par exemple, limite.
En outre on indique sa valeur initiale et - par l'intermédiaire de la largeur du vecteur - sa taille.
- 3- Pour transférer des données à partir de l'User-Code au modèle de Simulink, on met un bloc Data Store Read dans le modèle de Simulink. Pour transférer des données du modèle de Simulink à User-Code, on utilise un bloc Data store Write. Les deux blocs sont trouvées dans la bibliothèque de connexions de Simulink.
- 4- Pour chaque bloc, on ouvre le dialogue de paramètres et on indique Data Store Name de la variable de communication à laquelle on veut accéder : par exemple, limite. Ce nom doit apparaître le même nom qu'on a donné au bloc correspondant de Data Store Memory. On peut également indiquer un temps d'échantillonnage pour la variable de communication.

IV.5.2.3. Pour obtenir un modèle de User-Code

- 1- On ouvre le dossier qui retient le modèle de Simulink.
- 2- On lance le processus de construction pour le modèle de Simulink.

Une fois qu'on a construit le modèle, on trouvera le fichier <model>_usr.c dans le dossier contenant le modèle. Ce modèle contient un ensemble de fonctions vides de C dans lesquelles on peut introduire le code C.

IV.5.2.4. Construction manuelle de User-Code

Le fichier de User-Code <[sub]model>_usr.c contient un ensemble de fonctions vides de C. Chaque fonction de C a un but particulier. On doit écrire le code C dans ces fonctions C selon leurs buts.

Pour transférer des données à partir de User-Code au modèle de Simulink on doit employer la fonction `usr_input()`. Si on veut transférer les données du Simulink à User-Code on doit utiliser `usr_output()`. La fonction `usr_sample_input()` supplémentaire peut être employée pour commencer le processus de conversion pour un ADC (Analog Digital Converter). Ce processus est alors terminé avant que la fonction `usr_input()` commence.

IV.5.2.4.1 Comment construire manuellement User-Code

On remplit les fonctions vides du fichier C `<[sub]model>_usr.c`. On met le code C dans les fonctions appropriées selon leurs buts comme indiqué ci-dessous:

But	Fonction
L'initialisation, par exemple, écriture des valeurs d'initialisation des dispositifs de sortie.	<code>usr_initialize()</code>
Pour commencer la conversion, par exemple, pour les dispositifs analogiques numériques.	<code>usr_sample_input()</code>
Accès de lecture, par exemple, pour des dispositifs d'entrée	<code>usr_input()</code>
Accès d'écriture, par exemple, pour des dispositifs de sortie	<code>usr_output()</code>
Écriture des valeurs d'achèvement des dispositifs de sortie	<code>usr_terminate()</code>
Tâche de fond	<code>usr_background()</code>

Tableau IV.2 Les différentes fonctions de User-Code

IV.5.2.5. Comparaison entre S-Functions et User-Code

S-functions et User-Code ont des avantages et des inconvénients, qui sont énumérés ci-dessous.

	Avantages	Inconvénients
S-function	Peut être exécuté dans n'importe quelle tâche du programme en temps réel. Facile à utiliser par l'intermédiaire du bloc de la S-function de Simulink.	Code source n'est pas aussi facile que celui de User-Code.
User-Code	Contrôle précis de la période d'initialisation/exécution dans l'exécution du modèle. Code source plus simple que celui des S-fonctions.	Toujours exécuté avec la tâche de temporisateur la plus rapide.

Tableau IV.3 Comparaison entre S-function et User-Code

IV.6. Construction et téléchargement d'un modèle [3]

Quand un modèle de Simulink est complet, le code en temps réel peut être produit et téléchargé au système de dSPACE.

IV.6.1. Les principes pour la construction et le téléchargement

IV.6.1.1. Les fichiers

RTI produit toujours un fichier de description de système (SDF), qui contient toutes les informations appropriées sur les divers fichiers d'exécution et des fichiers TRC.

Avec ControlDesk on emploie ce fichier pour télécharger l'application au système de dSPACE ou pour accéder aux signaux et aux paramètres de la simulation.

Plusieurs fichiers intermédiaires sont produits pendant le processus de construction.

IV.6.1.2. Le processus de construction

Quand on exécute le processus de construction (par exemple, par **build** dans la page de Real Time Workshop), un certain nombre d'utilitaires sont appelés séquentiellement pour établir le programme en temps réel et pour le télécharger au matériel. Le processus de construction peut être divisé en deux phases séparées: la phase de génération de code et la phase de compilation.

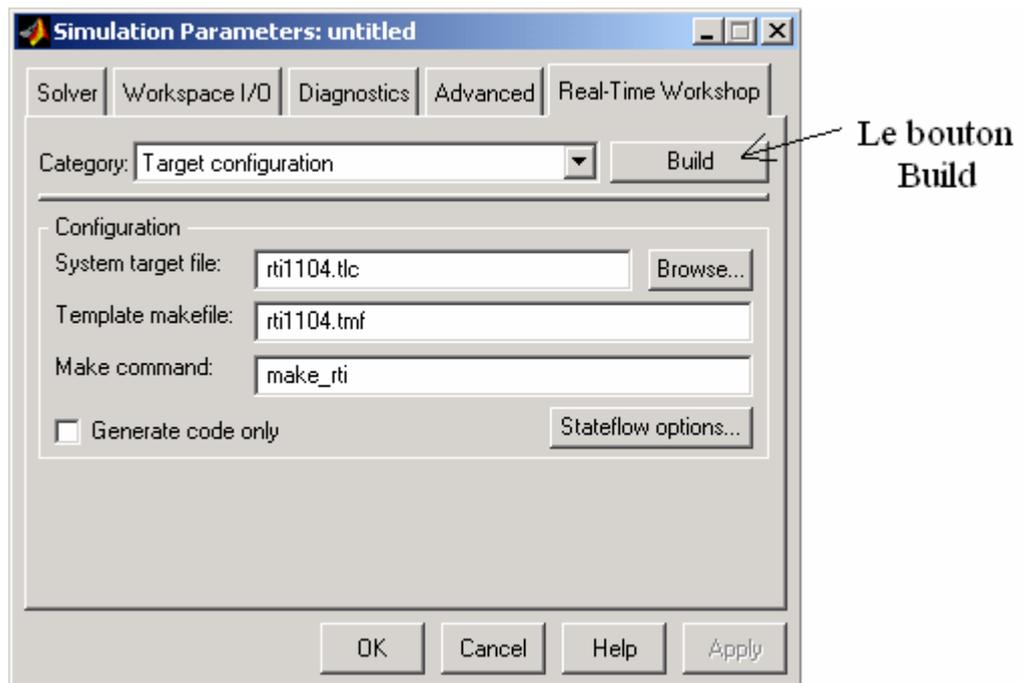


Figure IV.9 Le bouton Build

IV.6.1.2.1 Phase de génération de code

Dans cette phase tout le code de C requis pour l'application en temps réel est produit.

IV.6.1.2.2 Phase de compilation

Dans cette phase les sources C sont traduites en application finale, qui peut être exécutée sur le matériel de dSPACE.

IV.6.2. La spécification des options du processus de construction :

On doit indiquer la majorité de configurations appropriées dans le dialogue de paramètres de simulation. Pour cela, il faut ouvrir le modèle de Simulink et cliquer sur le menu Simulation et puis choisir Simulation parameters.

L'illustration suivante montre les configurations correspondantes pour le dialogue de paramètres de simulation

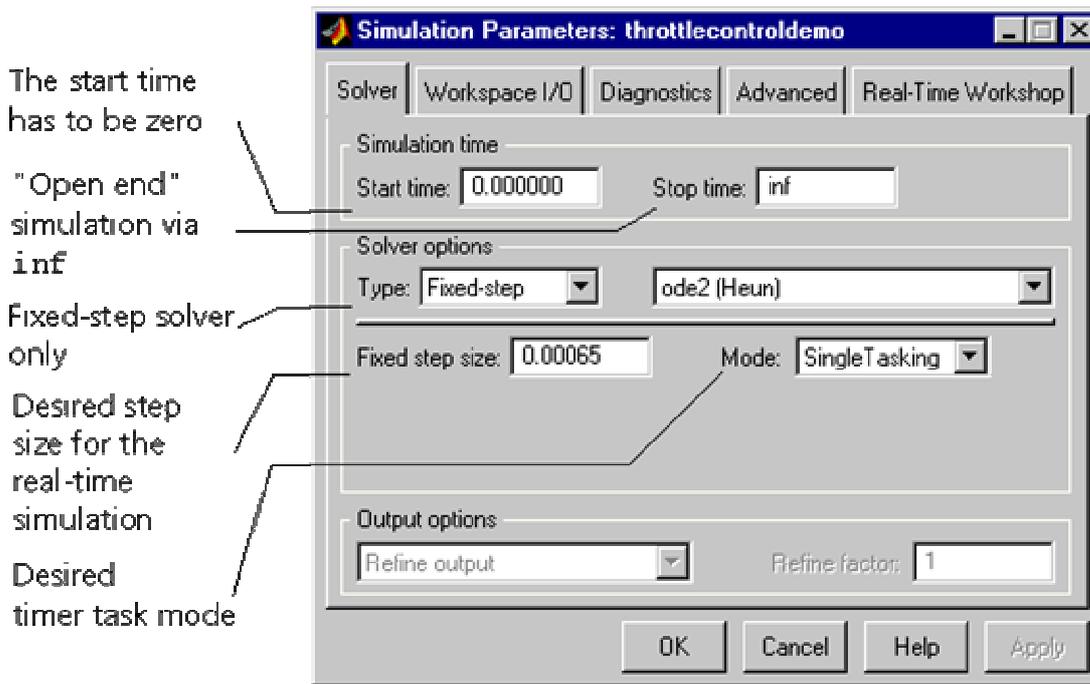


Figure IV.10 Les paramètres de simulation de Solutionneur

Les configurations sont :

- Start Time (le temps de départ) : doit être à zéro,
- Stop Time (le temps d'arrêt) : doit être 'inf' c-a-d infini,
- Fixed-Step: signifie que le temps d'échantillonnage est fixe.
- Fixed Step size: la valeur du temps d'échantillonnage.
- Mode : le mode du modèle de Simulink est 'Simple tâche'.

IV.6.3. Construction et téléchargement

Une fois qu'on a indiqué les paramètres de simulation, on peut commencer la procédure de construction et de téléchargement.

Avec la commande **cd** de MATLAB, on change le dossier source avec le dossier qui contient le modèle de Simulink qu'on veut le charger sur la carte DS1104.

On ouvre le modèle et on commence le processus de construction:

On clique sur la commande Tools – Real-Time Workshop – Build Model à partir de la barre de menu du modèle.

IV.6.4. Reconstruction du modèle :

On doit reconstruire le modèle entier si on change n'importe quel élément de ce qui suit:

- La dimension de n'importe quelle variable de S-function (nombre des entrées, des sorties, etc.),
- La période d'échantillonnage de n'importe quelle S-function,
- Les dimensions de 'Data Store Memories' utilisés pour le transfert de données entre le modèle de Simulink et User-Code.

Si on modifie une S-function existante on a besoin de recompiler le fichier MEX DLL. Cependant, on peut réutiliser l'ancien fichier MEX DLL si on ne change pas l'information de la taille de la S-function.

IV.7. La simulation externe [5]

Le mode externe est un mode de simulation fourni par Simulink. Ce mode concerne deux environnements de calcul séparés, un hôte et une cible.

- L'hôte est l'ordinateur qui exécute MATLAB et Simulink.
- La cible est le matériel de dSPACE qui exécute l'application qu'on a créée avec Workshop et RTI.

En utilisant le mode externe, on peut modifier les paramètres de bloc de l'application en temps réel sans régénération et recompilation du code, et sans ControlDesk.

En mode externe, les changements de paramètres du modèle de Simulink sont automatiquement téléchargés à l'application en temps réel qui fonctionne actuellement sur le matériel de dSPACE.

IV.7.1. Exécution d'une simulation externe

Les instructions suivantes décrivent comment on commence la simulation externe après la construction et le téléchargement de l'application en temps réel et comment on change les paramètres de la simulation par l'intermédiaire du modèle de Simulink.

Pour commencer la simulation externe

1- On ouvre les paramètres de simulation et dans la page Real-Time Workshop on choisit la catégorie 'RTI code generation options' et on coche l'option External simulation.

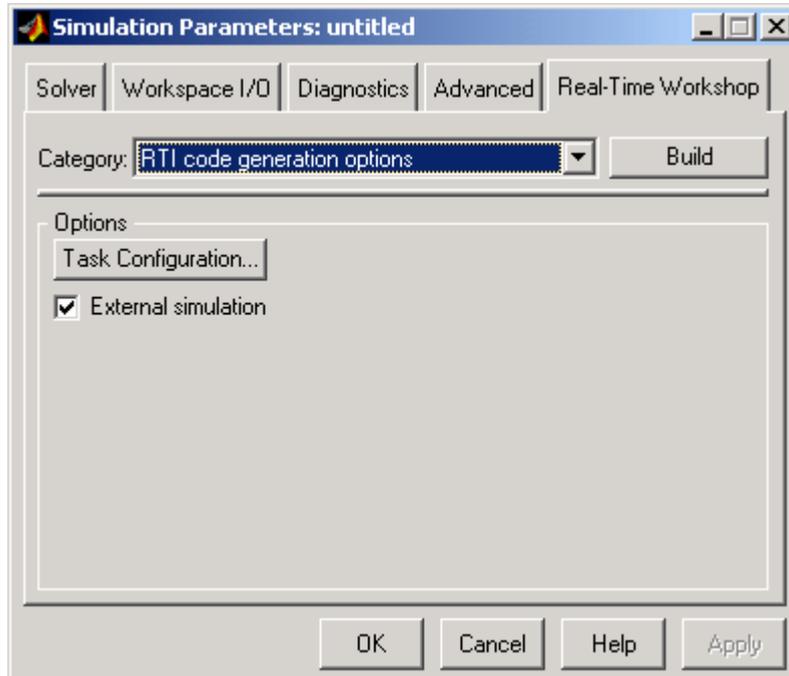


Figure IV.11 Les paramètres de Real-Time Workshop

2- On construit et on télécharge le modèle comme d'habitude, de sorte qu'il fonctionne sur le matériel de dSPACE.

3- On choisit la commande Simulation- External.

4- On choisit la commande Simulation-Connect to target.

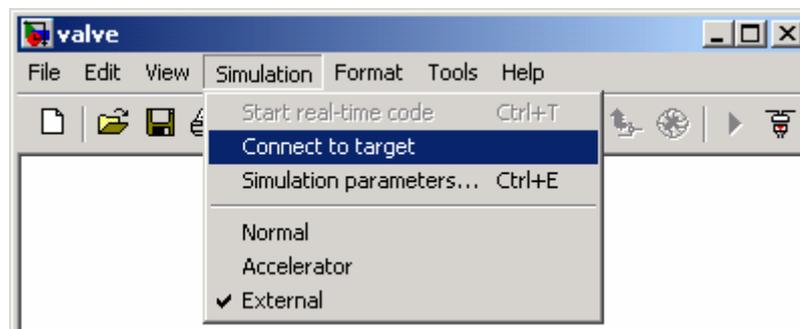


Figure IV.12 Menu Simulation du modèle Simulink

Pour télécharger des paramètres par l'intermédiaire de la simulation externe :

1- on change les paramètres dans le modèle de Simulink selon les conditions, par défaut, chaque paramètre est téléchargé immédiatement,

2- si on veut traiter en groupe le téléchargement de plusieurs paramètres, on choisit la commande **Tools – External Mode Control Panel** et puis **batch download**.

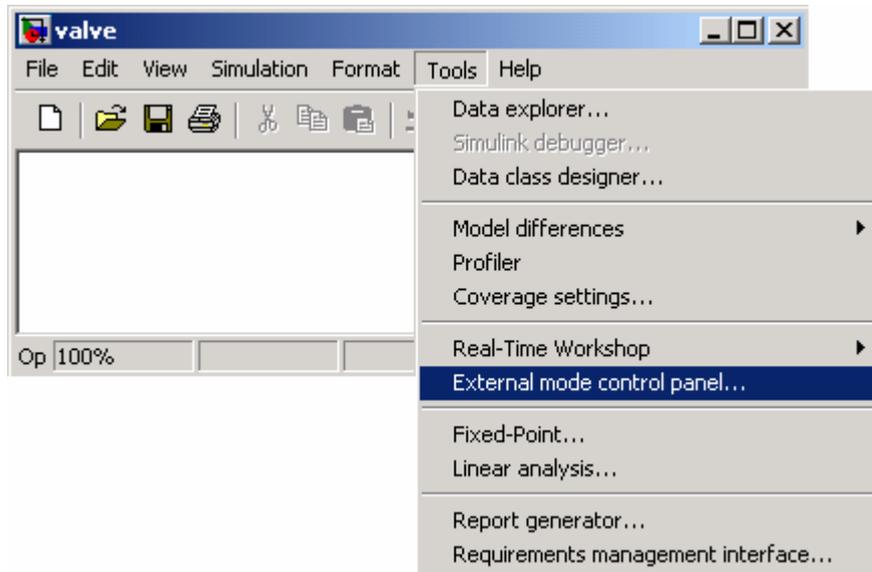


Figure IV.13 Menu Tools du modèle Simulink

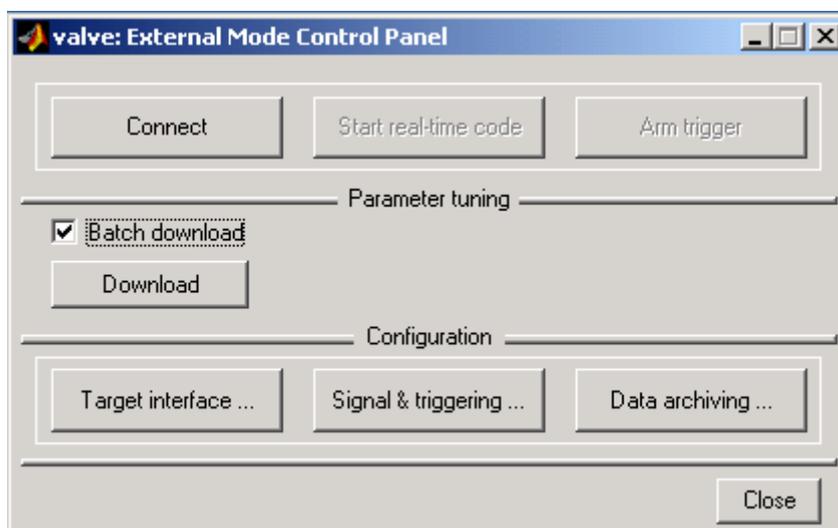


Figure IV.14 External Mode Control Panel

Alors on doit cliquer sur le bouton **Download** chaque fois qu'on veut télécharger tous les paramètres changés dès le dernier téléchargement.

Pour arrêter la simulation externe

- 1- on choisit la commande **Simulation – Disconnect from target**,
- 2- avant qu'on puisse effectuer une simulation standard de Simulink pour le modèle après avoir travaillé en mode externe de simulation, on choisit la commande **Simulation – Normal**.

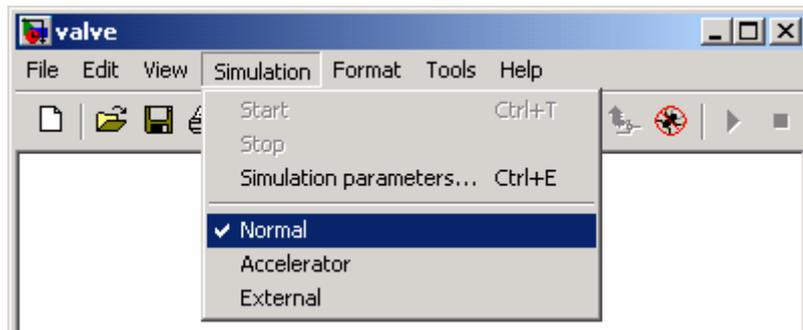


Figure IV.15 L'option Normal de menu de Simulation

Conclusion

L'interface en temps réel RTI est la liaison entre Simulink et la carte de dSPACE DS1104, et elle permet la construction et le téléchargement du modèle de Simulink, ainsi que l'exploitation des interfaces d'entrée/sortie comme les DAC et ADC. Ce qui il faut retenir est la configuration des paramètres appropriés de simulation, la méthode de construction et le téléchargement du modèle au matériel de dSPACE.

Chapitre V

Simulation de commande
d'une valve

Introduction :

Ce qu'on a vu précédemment est une étude et description matérielle et logicielle de la carte dSPACE DS1104. Dans ce chapitre on va implémenter, sur cette carte, un modèle de Simulink qui représente une commande d'une valve.

V.1. Description de la valve

On suppose qu'on veut développer une commande électronique pour une valve comme celle montrée ci-dessous.



Figure V. 1 Une valve

Une **valve** est un dispositif mécanique qui règle le flux de fluides (que ce soient des gaz, ou des liquides) par l'ouverture, la fermeture partielle des diverses voies de passage. [12]
Le système de commande doit contrôler et modifier l'ouverture de la valve suivant une référence donnée.

V.2. Description de la commande [1]

La première étape de développant est de modéliser la valve et la représenter dans le modèle de Simulink. Un modèle approprié de Simulink est montré dans l'illustration ci-dessous:

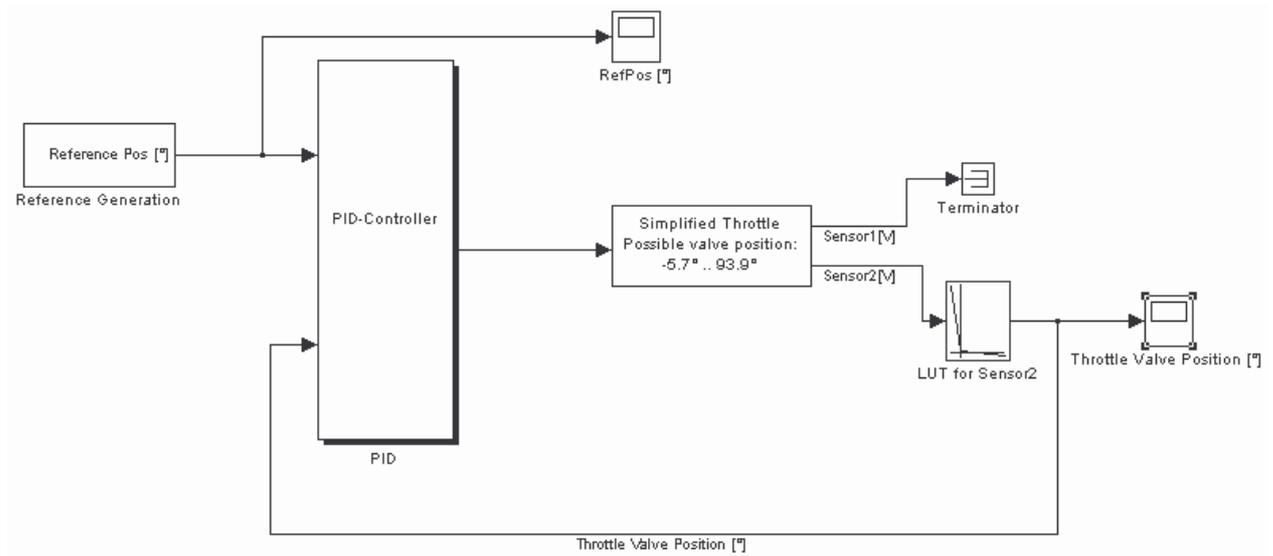


Figure V. 2 Modèle de Simulink de commande de la valve

V.2.1.Principes de fonctionnement du modèle Simulink : [14]

Le modèle de Simulink se compose de plusieurs blocs et des sous- systèmes :



Figure V. 3 Générateur de référence

V.2.1.1 Génération de référence :

C'est un sous-système qui permet de générer la référence de la position de la valve :

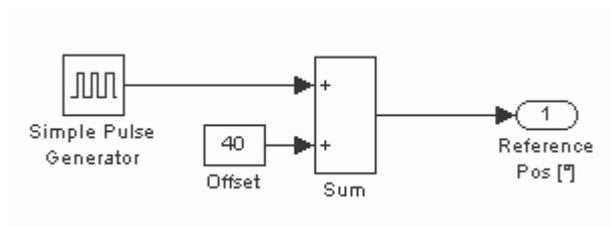


Figure V. 4 Le sous-système générateur de référence

- Simple Pulse Generator : c'est générateur d'impulsions.
- Offset : qui génère une tension constante '40'.
- Sum : c'est un sommateur qui additionne la valeur constante d'offset aux impulsions
- Reference Pos[°] : est un port de sortie du sous-système qui fournit la référence de position de la valve.

V.2.1.2 La commande PID :

C'est un sous-système composé de plusieurs blocs



Figure V. 5 Le contrôleur PID

À l'intérieur de ce bloc on trouve :

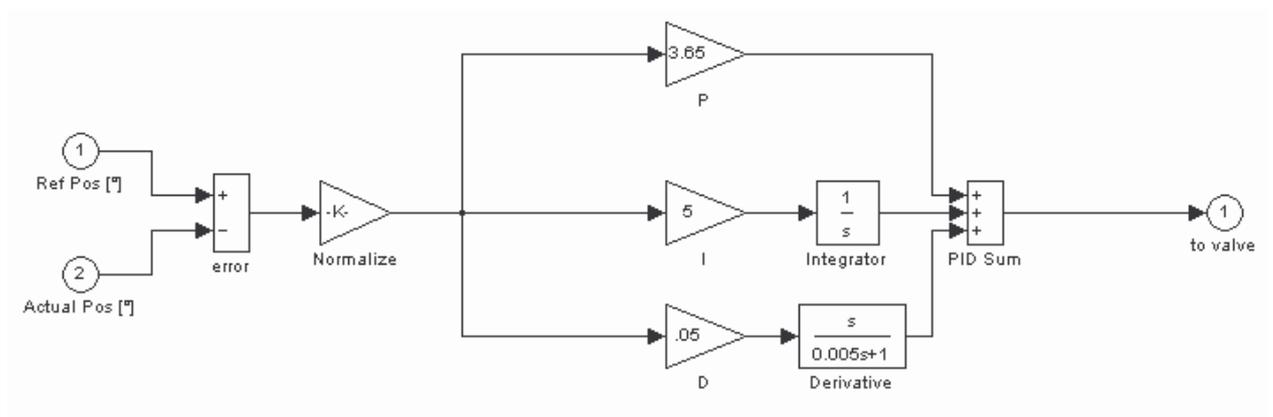


Figure V. 6 Le sous-système de contrôleur PID

- Ref Pos [°] : est un port d'entrée qui est relié avec le bloc de sortie précédent du sous-système Reference generation.
- Actual Pos [°] : port d'entrée représente la position actuel de la valave.
- error : est un soustracteur qui calcule la différence entre les deux positions : référence et actuel.
- Normalize : est un gain
- P : est un gain de 3.65
- I : est un gain de 5
- Intégrator : c'est un intégrateur représenté par sa fonction de transfert $1/s$

- D : un gain de 0.05
- Derivative : un dérivateur et sa fonction de transfert $s/(0.005+1)$
- PID Sum : un sommateur à trois entrées.
- To valve : un port de sortie qui fourni la commande la valve.

V.2.1.3 Simplified throttle:

Le sous-système simplifié d'une valve électrique, qui contient principalement un moteur à courant continu, une valve, deux ressorts, et deux capteurs. L'angle d'ouverture de la valve est commandé par PID. Si le signal est nul, les ressorts poussent la valve dans une position zéro définie comme 0° .

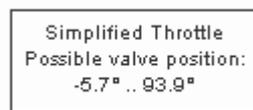


Figure V. 7 Le bloc de la valve

Dans ce sous-système on trouve le modèle de Simulink suivant :

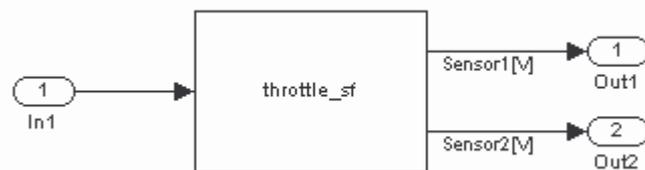


Figure V. 8 Le sous-système de la valve

- In 1 : un port d'entrée qui est relié avec le port de sortie du bloc PID.
- throttle_sf : c'est un bloc crée avec une S-function qui modélise la valve.
- Sensor 1 et 2 [V] : sont deux port de sortie représentant les deux capteurs qui fournissent la position de la valve.

V.2.1.4 Terminator :

Dans ce modèle on utilise un seul port de sortie de la valve pour afficher sa position. Le deuxième port est relié à un terminator pour éviter les messages d'avertissement de Simulink qui correspond aux ports ouverts inutilisés.



Figure V. 9 Terminator

V.2.1.5 RefPos [°]:

C'est un oscilloscope qui permet la visualisation de la position de référence de la valve.

V.2.1.6 Throttle Valve Position :

Un autre oscilloscope pour la visualisation de la position de la valve.

V.3. La simulation :

V.3.1. La simulation avec Simulink :

Dans cette partie en utilisant Simulink seulement sans la carte DS1104

V.3.1.1 Les résultats :

Après avoir changé les paramètres de P,I et D, on a pu avoir une bonne commande de la valve en suivant la référence.

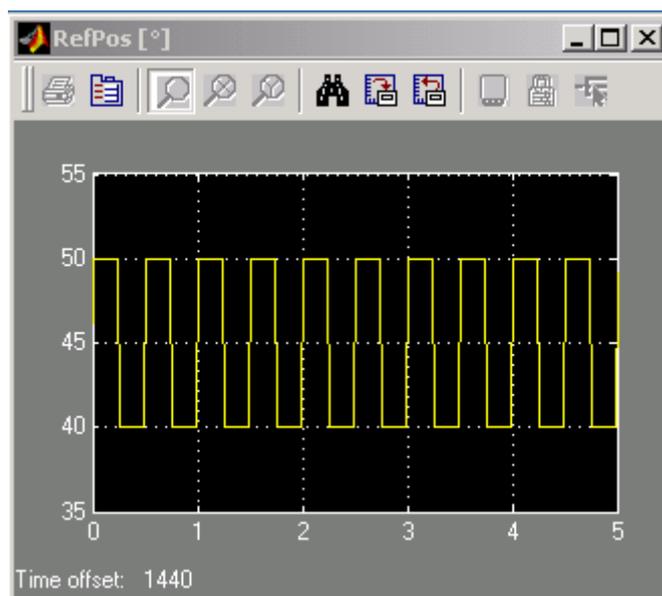


Figure V. 10 Graphe de la référence par Simulink

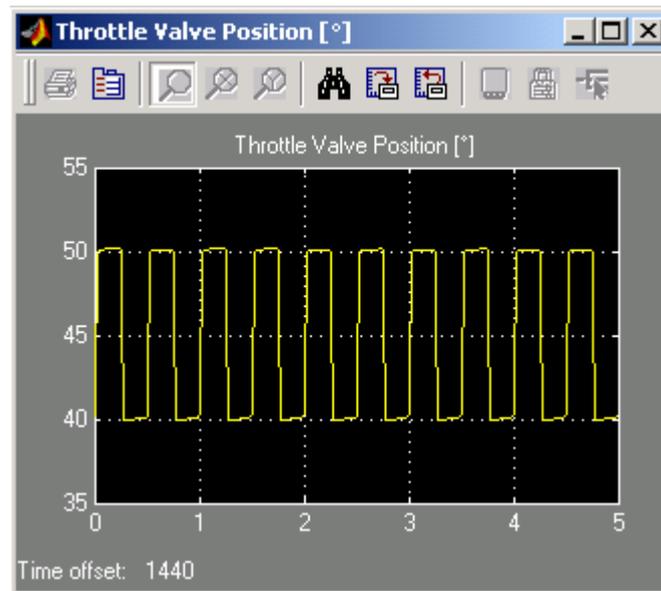


Figure V. 11 Graphe de la position de la valve par Simulink

On voit bien que la valve suit la référence avec une certaine erreur.

V.3.2.Simulation avec la carte DS1104

Après la construction et le téléchargement de ce modèle sur le matériel de dSPACE en utilisant le bouton 'Build' de la page Real-Time Workshop, on peut démarrer la simulation avec ControlDesk.

V.3.2.1 Création d'une nouvelle expérimentation

Pour une bonne utilisation de ControlDesk et pour chaque nouvelle simulation, il faut créer une expérimentation en cliquant sur File-New Experiment.

L'expérience permet de grouper tous les éléments de simulation comme les Layouts, et quand on l'ouvre, le fichier de description de système (SDF) se charge automatiquement.

V.3.2.2 Les éléments de l'expérimentation

Après la création de l'expérimentation, on construit un panneau d'instruments qui est une interface permettant de visualiser les variables et modifier les paramètres de simulation.

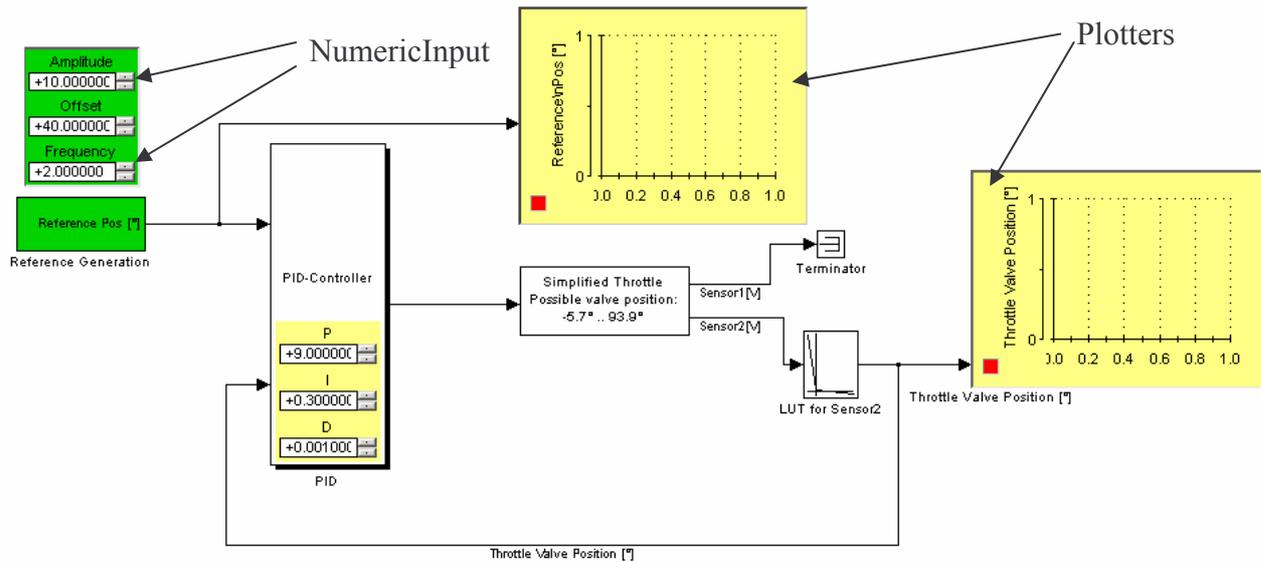


Figure V. 12 Panneau d'instruments de la commande de la valve

Ce panneau contient deux plotters, un pour la position de référence et l'autre pour la position de la valve, six NumericInput, trois pour les paramètres de PID et trois pour les paramètres de générateur de référence (l'amplitude des impulsions, la fréquence et l'offset).

Pour pouvoir utiliser ce panneau il faut connecter les données en les glissant à partir de 'variables list' et en les déposant sur l'instrument qui convient (Plotter ou NumericInput).

Après ces opérations, on charge le fichier SDF du modèle à partir de 'File Selector' en le glissant et déposant (drag and drop) sur l'image de la carte DS1104 dans 'Platform Navigator'.

V.3.3. Les résultats

On procède avec la même manière, on change les paramètres D, I et P avec les mêmes valeurs de Simulink on trouve les graphes suivants :

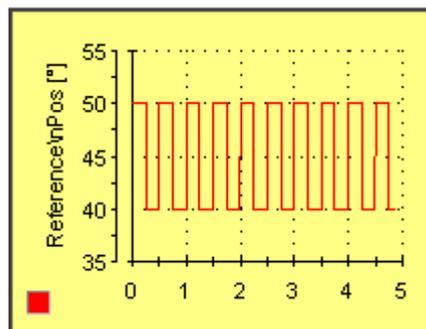


Figure V. 13 Graphe de la référence par ControlDesk

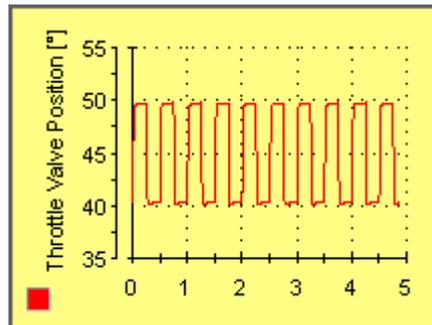


Figure V. 14 Graphe de la position de la valve par ControlDesk

On remarque qu'on a obtenu les mêmes graphes que Simulink, et c'est logique.

V.4. Comparaison entre Simulink et ControlDesk :

Les résultats des deux simulations sont les mêmes mais la simulation avec ControlDesk permet de construire un panneau d'instrument qui est une interface dans laquelle on visualise toutes les variables qui nous intéressent (grâce aux Plotters par exemple) et on peut aussi changer tous les paramètres nécessaires en cours de simulation (par exemple NumericInput).

La simulation avec Simulink c'est la première étape de développement de la commande, et si on a un panneau de connexion et une valve, on peut passer à la deuxième étape 'RCP', dans cette étape on change le bloc qui représente la valve dans le modèle Simulink avec un bloc d'entrée/sortie de RTI qui convient par exemple un DAC, et on construit et télécharge ce modèle de Simulink sur la carte DS1104 et on la connecte avec la valve par l'intermédiaire du panneau de connexion. À ce stade, on peut commander cette valve en temps réel et voir l'effet et l'efficacité de la commande en changeant ses paramètres.

Conclusion :

Simulink, la carte dSPACE DS1104 et son logiciel ControlDesk sont des outils puissants pour le développement des commandes et des contrôleurs en parcourant les quatre étapes de développement : simulation en Simulink jusqu'à la génération de code.

Conclusion générale et Perspectives

Conclusion générale et Perspectives

Ce PFE nous a permis d'atteindre certains objectifs :

- La mise en marche de la carte DS1104.
- Comprendre ses principes de fonctionnement.
- La programmation de la carte avec Simulink.
- La familiarisation avec ControlDesk.

Les DSPs sont généralement programmables en C, dSPACE a réussi de réunir la puissance de calcul des DSPs avec Simulink qui un outil puissant de simulation et développement des commandes et contrôleurs, facile à utiliser et ne nécessite pas des connaissances approfondies en programmation. Ainsi, ses cartes contiennent des interfaces d'entrée/sortie permettant de connecter des dispositifs externes pour les commander en temps réel.

Il est clair que ce travail n'est qu'à son début. En effet, notre ambition est de réaliser un modèle Simulink permettant la commande et le contrôle d'un bras manipulateur, par exemple.

L'idéal serait d'avoir un panneau de connexion pour qu'on puisse connecter la carte avec l'élément à commander et voir l'efficacité du modèle de Simulink et faire les testes en temps réel.

Pour bien exploiter la puissance de la carte DS1104, il est préférable d'avoir la version 5 de ControlDesk qui est compatible avec Windows XP, et le plus intéressant est la compatibilité de cette version avec Simulink de MATLAB 7 qui fournit un large choix et contient une grande bibliothèque des blocs.

Bibliographie

- [1] HelpDesk : l'aide de la carte dSPACE DS1104
- [2] www.edf.fr/fichiers/pdf/dp_idea.pdf
- [3] 'Real-Time Interface Implementation guide For Release 3.4' dSPACE, May 2002
- [4] 'ControlDesk Automation Guide For Release 3.4' dSPACE, May 2002
- [5] 'ImplementRTI.pdf', CD Solution for Control Release 3.4 For MATLAB R12.1 dSPACE
- [6] 'CDAutoGuide.pdf', CD Solution for Control Release 3.4 For MATLAB R12.1 dSPACE
- [7] 'ControlDesk Experiment guide For Release 3.4' dSPACE, May 2002
- [8] 'DS1104 R&D Controller Board Installation and Configuration guide For Release 3.4' dSPACE, May 2002
- [9] 'CDExpGuide.pdf', CD Solution for Control Release 3.4 For MATLAB R12.1 dSPACE.
- [10] 'DS1104InstConfig.pdf', CD Solution for Control Release 3.4 For MATLAB R12.1 dSPACE
- [11] 'DS1104Features.pdf', CD Solution for Control Release 3.4 For MATLAB R12.1 dSPACE
- [12] www.wikipedia.org
- [13] www.commentcamarche.net
- [14] Help Simulink 4.1