



INTRODUCTION A STATEFLOW®

Pour les curieux et ceux qui s'y intéressent...

**Modélisation & Simulation des
Systèmes à Evènements
Discrets (et Continus).**

Ph. Hautcoeur

« Les professeurs sont faits pour les gens qui n'apprendraient rien tout seuls. Le savoir qui compte est celui qu'on se donne soi-même par curiosité, passion de savoir. »



Paul Léautaud

Philippe Hautcoeur

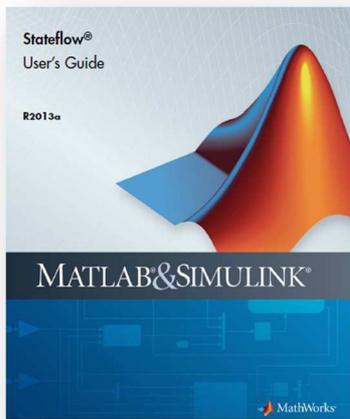
Professeur de Sciences Industrielles de l'Ingénieur
Classes Préparatoires aux Grandes Ecoles
Lycée Clemenceau à Nantes

Ce document évolue grâce à votre concours. >>>> [contact : philippe.hautcoeur@ac-nantes.fr](mailto:philippe.hautcoeur@ac-nantes.fr)



L'utilisation à caractère commerciale ou l'hébergement sur un site de ce document sans autorisation de l'auteur est strictement interdite.

Avant-propos...



Stateflow[®] est un module développé par la société américaine **MathWorks**[®] qui permet la simulation de machines d'état.

Une machine d'état comporte un nombre fini d'états. Elle modélise le comportement de systèmes qui passent d'un état à un autre en réponse à des événements. On parle alors de systèmes à événements discrets.

Comme son nom l'indique, ce module permet de tracer des diagrammes d'état (« State Chart ») et des diagrammes de flux (« Flow Chart »).

Stateflow[®] est intégré à **Matlab**[®] et **Simulink**[®]. Les modèles construits pourront par conséquent comporter des blocs des différentes « toolboxes » de Simulink et/ou appeler des **fonctions Matlab** et/ou des **fonctions Simulink** comme nous le verrons. Ainsi le modèle global d'un système complexe pourra comporter des **modèles linéaires continus** construits avec Simulink sous la forme de schéma-blocs, **des machines à état** construites avec Stateflow[®] ou encore des **modèles acausaux** réalisés en utilisant **Simscape**[®].

Aussi Stateflow[®] permet de simuler le comportement de **systèmes hybrides** c'est-à-dire à événements **discrets et continus**. C'est par exemple le cas d'une balle qui rebondit sur le sol. En effet son déplacement dans l'air est continu alors qu'à chaque rebond, considéré comme un événement, sa trajectoire est modifiée. C'est encore le cas si un robot doit éviter un obstacle présent sur sa trajectoire.

Une connaissance approfondie de Matlab[®] et Simulink[®] n'est pas indispensable pour commencer à travailler avec Stateflow[®]. Dans la plupart des exemples traités avec la version 2013b, les chemins menant aux composants des bibliothèques Simulink[®] utilisées sont précisés.

Après avoir présenté les différents outils et quelques applications, nous verrons comment implémenter un programme réalisé avec Stateflow[®] vers une cible telle que la carte **Arduino Méga 2560**.

Le guide de l'utilisateur complet de Stateflow[®] est téléchargeable sur le site de MathWorks[®] :

http://www.mathworks.com/help/pdf_doc/stateflow/sf_ug.pdf

Avant propos...	3
Chapitre 1 - Pour bien commencer	7
1.1- Préparation et présentation de l'interface	7
Chapitre 2 - Le « Chart »	11
A propos du fonctionnement de la machine d'état...	13
2.1- State	14
2.1.1 - « Label » et mots clé d'un état	14
A propos des transitions et des évènements	15
2.1.2 - « Label » d'une transition	15
2.1.3 - Cas particuliers de la transition réflexive et de la transition interne	17
2.1.4 - Prise en compte de l'activité d'un état dans une transition	18
2.1.5 - Les évènements extérieurs	18
A propos des opérateurs temporels	26
L'opérateur « after »	27
L'opérateur « before »	27
L'opérateur « at »	28
L'opérateur « every »	28
L'opérateur « temporalCount »	28
A propos des super-états ou états composites	29
Décomposition exclusive (OR) ou parallèle (AND) des états composites	29
Utilisation d'un « Subchart »	30
2.2- Default Transition	31
2.3- Junction	32
A propos des « Flow Charts »	35
2.4- History junction	41
2.5- Box	46
2.6- Simulink Function	47
2.7- Graphical function	53
2.8 MATLAB Function	64
2.9 Truth table	73
Chapitre 3 - D'autres exemples d'application du « Chart »	82
3.1- Initialisation de l'axe linéaire Mazet®	82
3.2- Simulation du fonctionnement d'un chronomètre	86
3.3- Pilotage d'une plateforme omnidirectionnelle	97

3.4- Traitement des informations délivrées par un Codeur SinCos _____	103
Chapitre 4 - Le « Chart (MATLAB) » _____	112
4.1- Exemple de la régulation TOR d'un four _____	112
Chapitre 5 - La « State Transition Table » _____	117
5.1- Exemple du codeur incrémental _____	121
Chapitre 6 - La « Truth Table » _____	128
6.1- Exemple de la commande d'un pont roulant _____	129
Chapitre7 - Prototypage _____	140
7.1 - Arduino et Simulink® _____	140
7.2 - La carte Arduino Mega 2560 _____	141
7.3 - Application : le Gyrodriver™ _____	142
Chapitre 8 - Proposition de méthodologie _____	155
Chapitre 9 - SysML State Machine vs Stateflow® _____	158