

Projet ACOTRIS

Réunion de travail

Présentation de la méthode support

Matlab/Simulink/StateFlow

SOMMAIRE

Introduction

1- Organisation du modèle

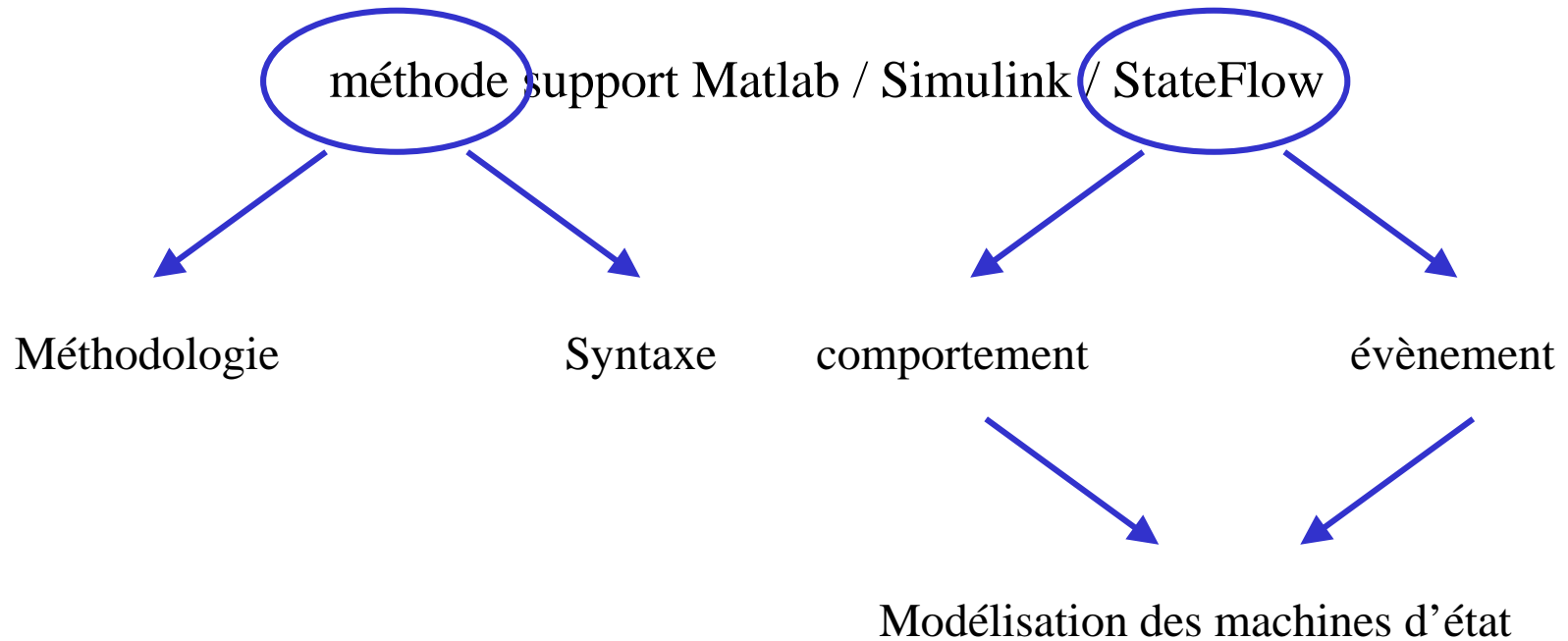
2- Règles de modélisation sous Simulink

3- Règles de modélisation sous StateFlow

4- Exemple du régulateur de vitesse

Débat

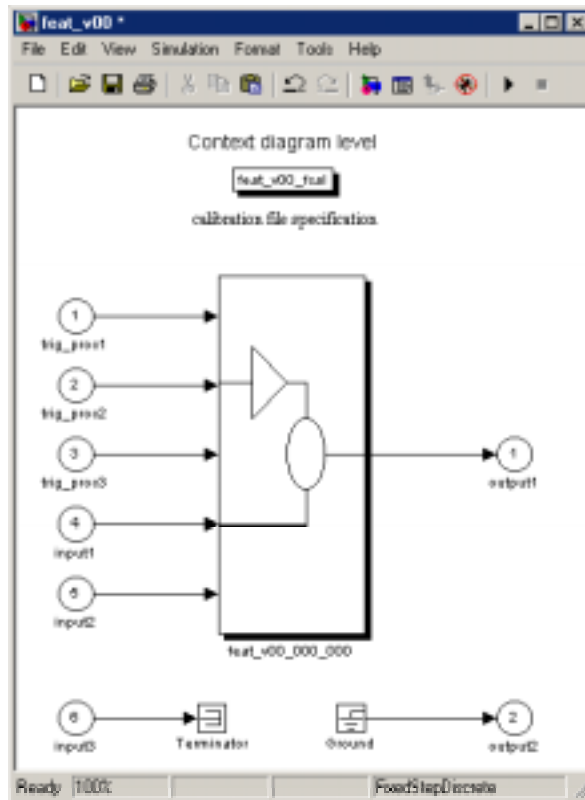
Introduction



➔ Cohésion entre la vue informatique et automatique

1- Organisation du modèle

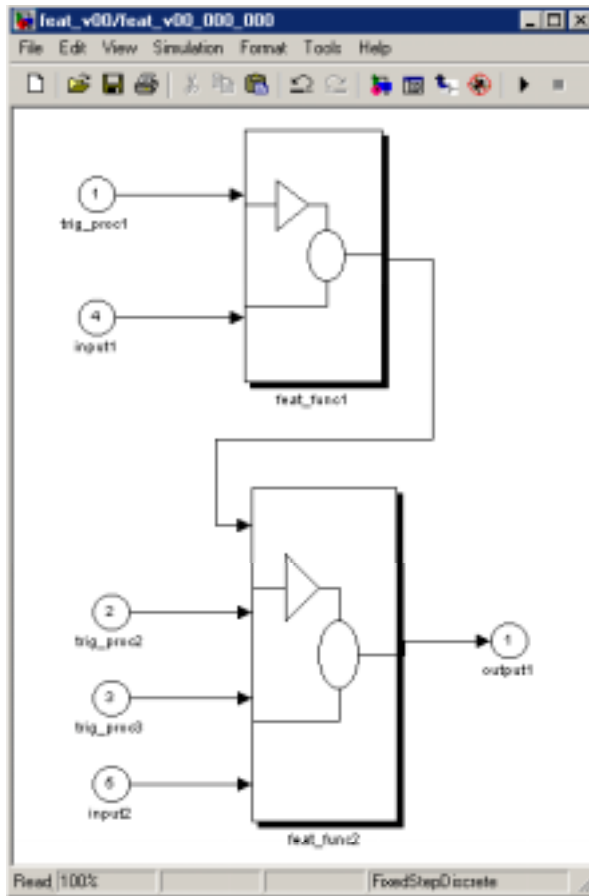
1.1- Context Diagram



➔ niveau le plus haut du système

- fichier de calibration,
- échanges de données via le Workspace de Matlab,
- entrées/sorties du système modélisé.

1.2- Execution Context Diagram



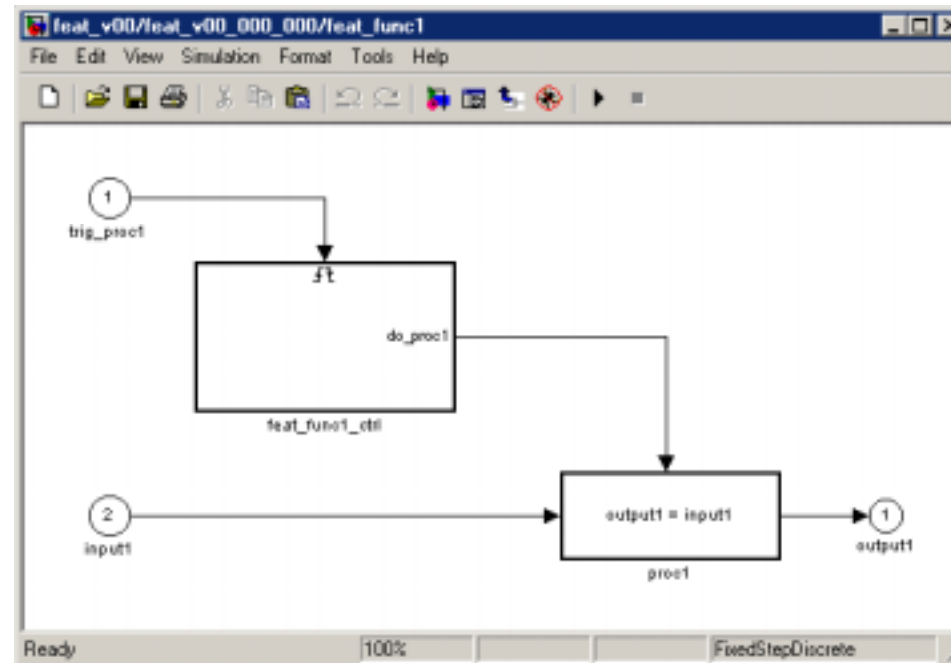
→ décomposition en sous-systèmes

Critères de décomposition :

- fonctionnalités,
- évènements,
- triggers.

1.3- Décomposition hiérarchique

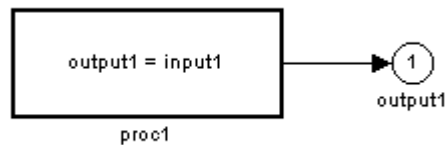
1 Sous-système = 1 C-Spec + n P-Spec



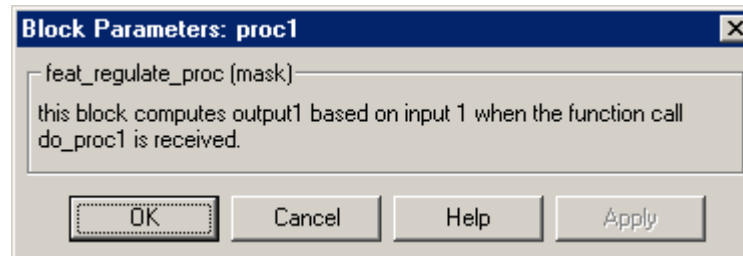
2- Règles de modélisation sous Simulink

2.1- Masquage des P-Spec

→ utilisation de pseudo-code



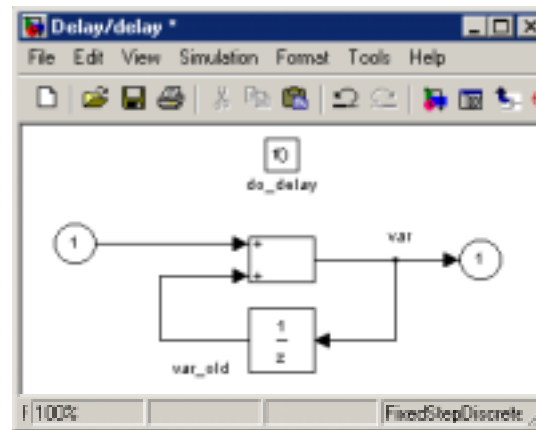
→ utilisation de description



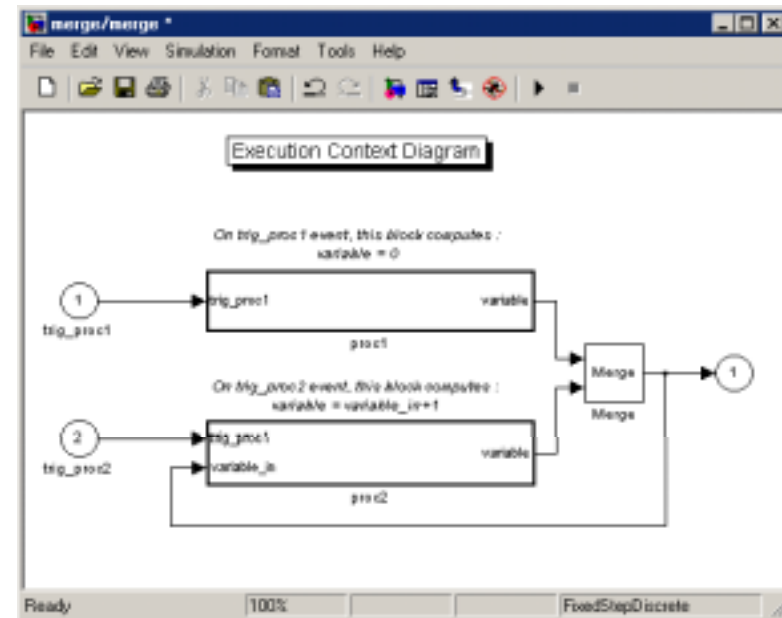
2.2- Mémorisation des données

→ l'utilisation de bloc Data Storage Memory est interdit

→ bloc Delay



→ bloc Merge



2.3- Signaux

- existence de règles et d'une syntaxe concernant respectivement les signaux et leur nom,
- existence de règles pour l'utilisation des blocs Goto/From Tags,
- existence de règles pour l'utilisation des vecteurs (tableaux).

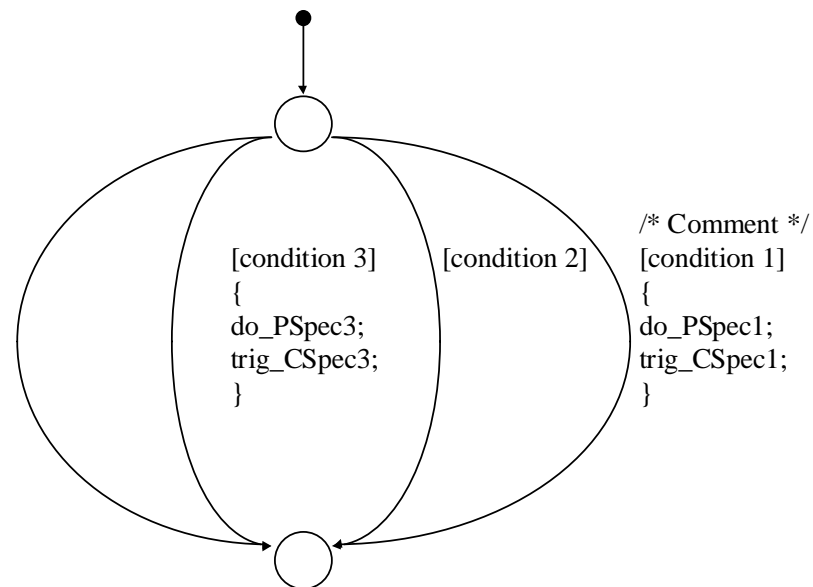
2.4- S-Function

- éviter d'utiliser ce type de bloc.

3- Règles de modélisation sous StateFlow

3.1- FlowChart

- Concepts fondamentaux



• FlowCharts autorisés

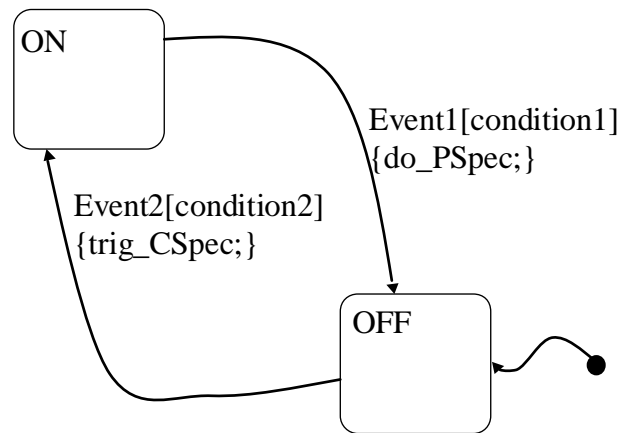
If-Then	
If-Then-Elseif	
OR gate	
AND gate	

While loop	
For loop	
Switch case	

➔ pas d'imbrication qui se traduirait par une instruction goto.

3.2- Machine d'état

→ Concepts fondamentaux

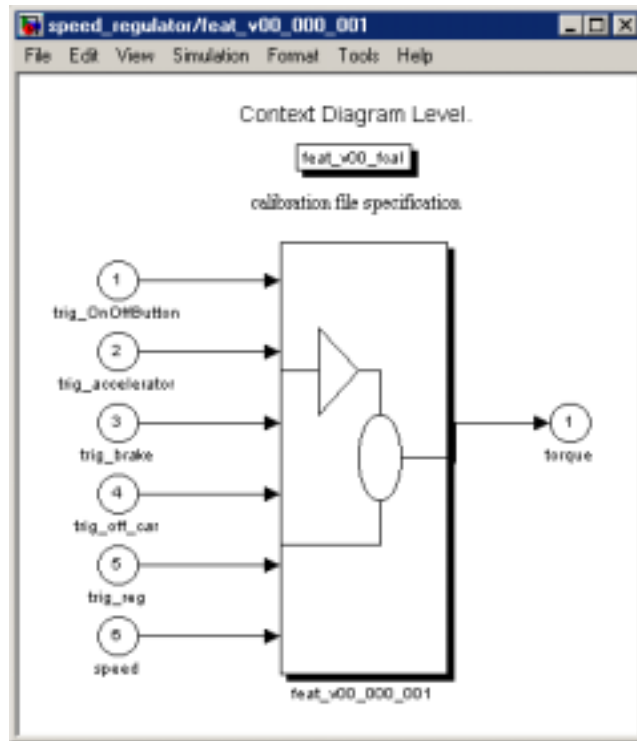


3.3- Diagrammes d'action de transition mixant les états/flux

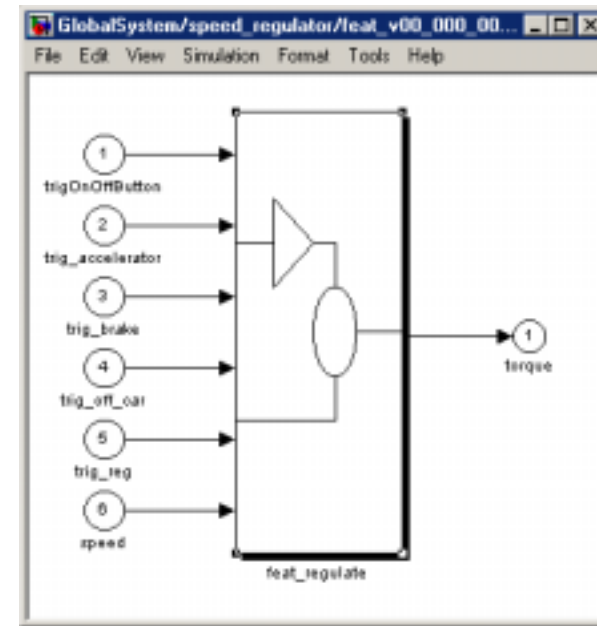
<p>Transition Etat/Etats</p>	
<p>Opération périodique</p>	
<p>Concurrence</p>	
<p>Sous-programme</p>	

4- Exemple du régulateur de vitesse

4.1- Context Diagram

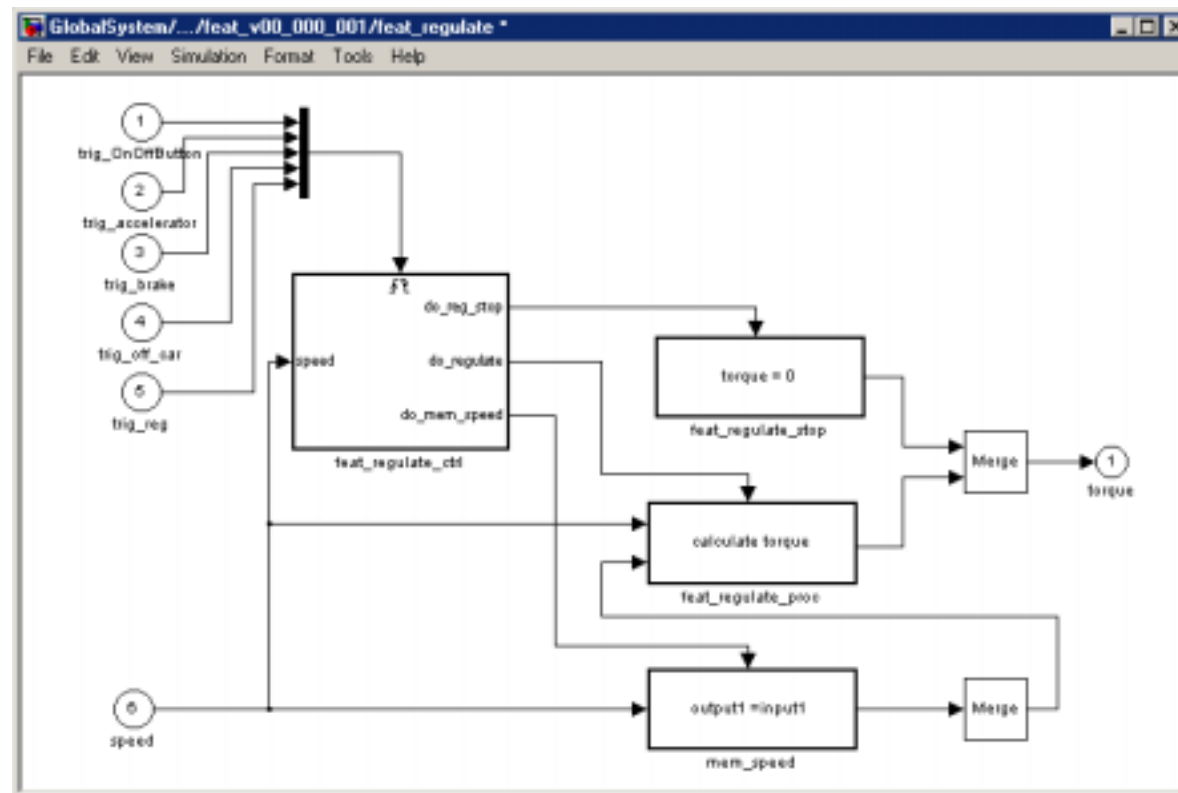


4.2- Execution Context Diagram

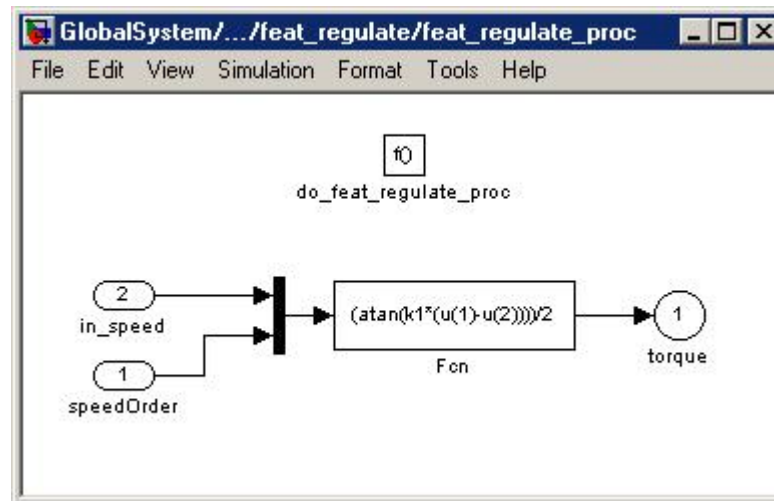


4.3- Le sous-système feat_regulate

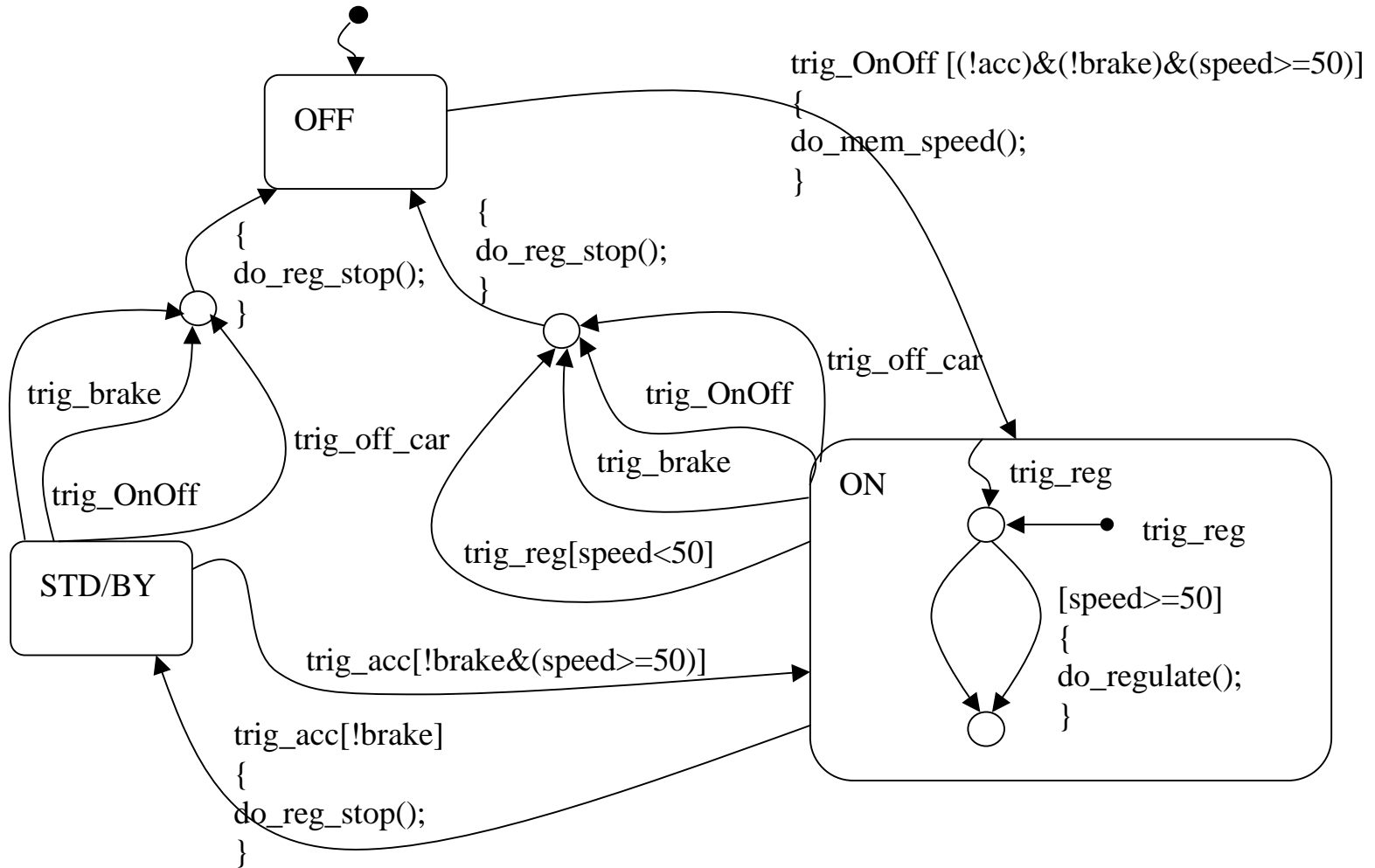
→ vue globale



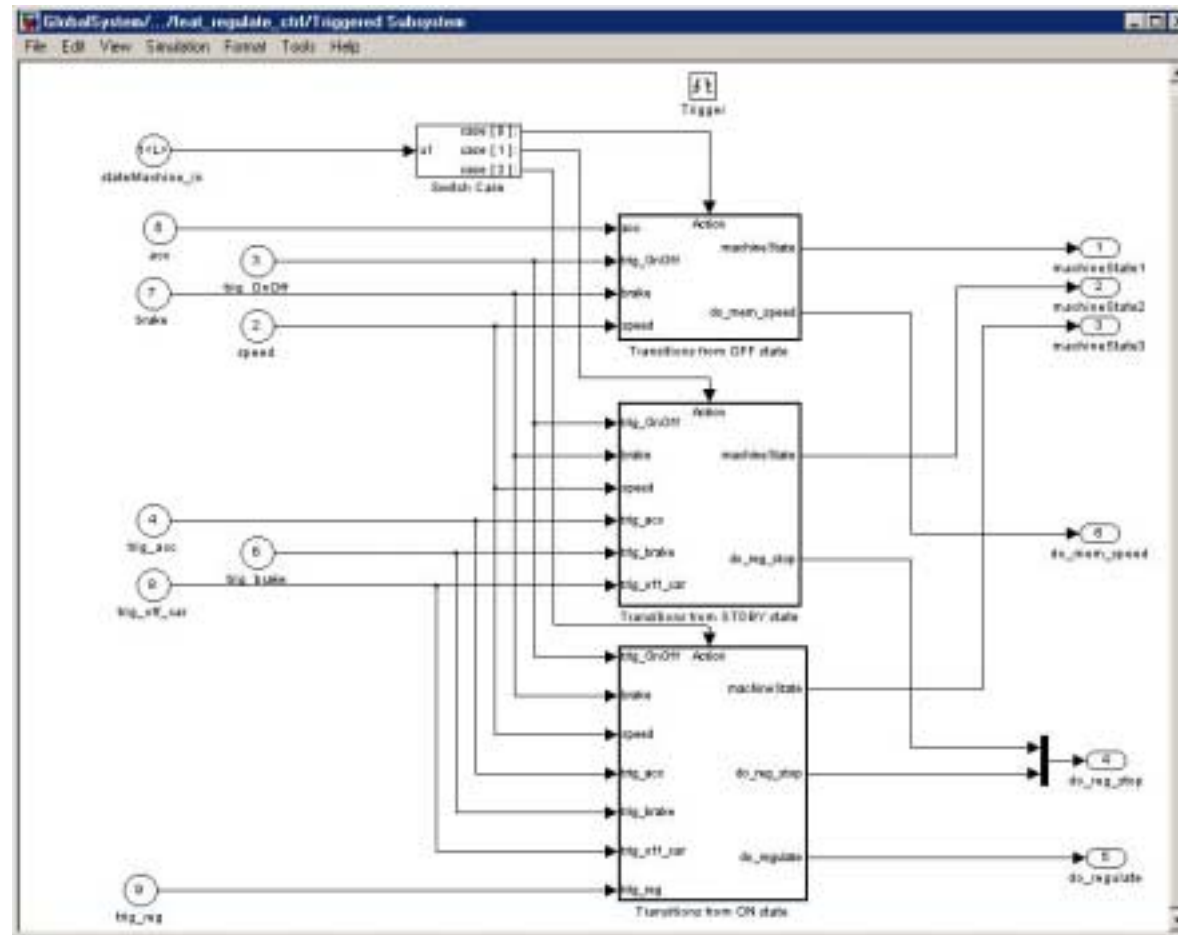
→ Détail du P-Spec feat_reg_proc sous Simulink



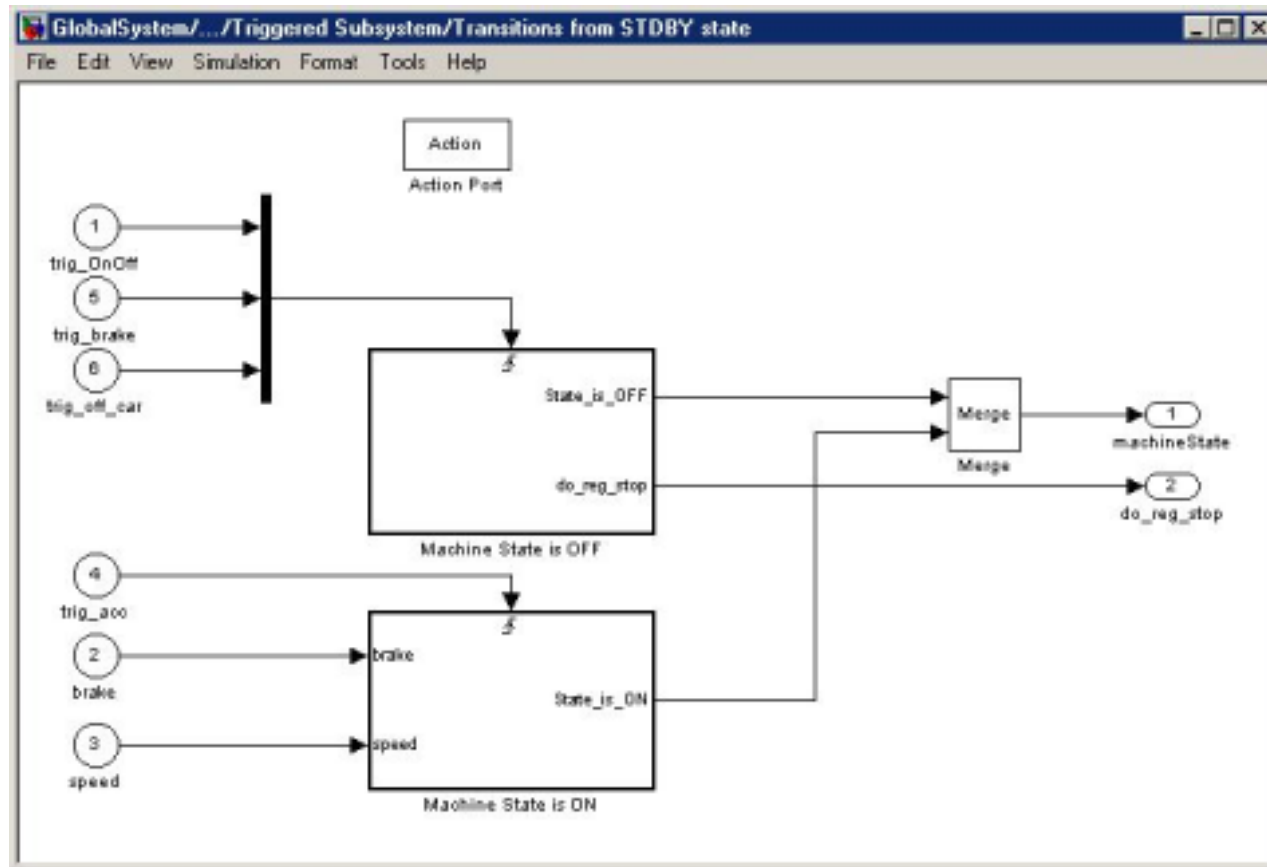
→ C-Spec en StateFlow



→ C-Spec en Simulink



→ Détail du C-Spec From STDBY state



Débat

Le système à modéliser doit être un système discret. Les systèmes modélisés sont des modèles continus d'où la problématique suivante :

Comment discrétiser le système en vue d'appliquer la méthode présentée précédemment?