

M2 EEA SME

Janvier 2020

EIEAS3FM

Architecture de l'électronique et conception conjointe

Auteur :

Nicolas OTAL

Antoine ROUTIER

Encadrants :

M. JAMMES

Introduction

Sigles et acronymes

VHDL	<i>VHSIC1 Hardware Description Language</i>
VHSIC	<i>Very High Speed Integrated Circuit</i>
AMS	<i>Analog and Mixed-Signal</i>

Table des matières

Introduction	2
1 Simulation d'un système de freinage sans ABS	5
1.1 Analyse instanciation véhicule	6
1.2 Analyse instanciation roue	7
1.3 Analyse instanciation frein	9
1.4 Analyse instanciation maître cylindre	11
1.5 Modélisation du régulateur de pression	11
2 Modélisation du système de freinage équipé d'un ABS	12
3 Intégration simplifiée du moteur thermique	13
4 Conclusion	14

1 Simulation d'un système de freinage sans ABS

Durant la pratique des travaux sur VHDL-AMS il est demandé aux étudiants de réaliser et d'instancier étapes par étapes les différentes parties d'un système de freinage d'une voiture. Pour la réalisation du travail nous avons travaillé avec le logiciel ModelSim et un éditeur de texte pour venir rédiger, en VHDL, les différentes instances du véhicule jusqu'à l'intégration de l'ABS.

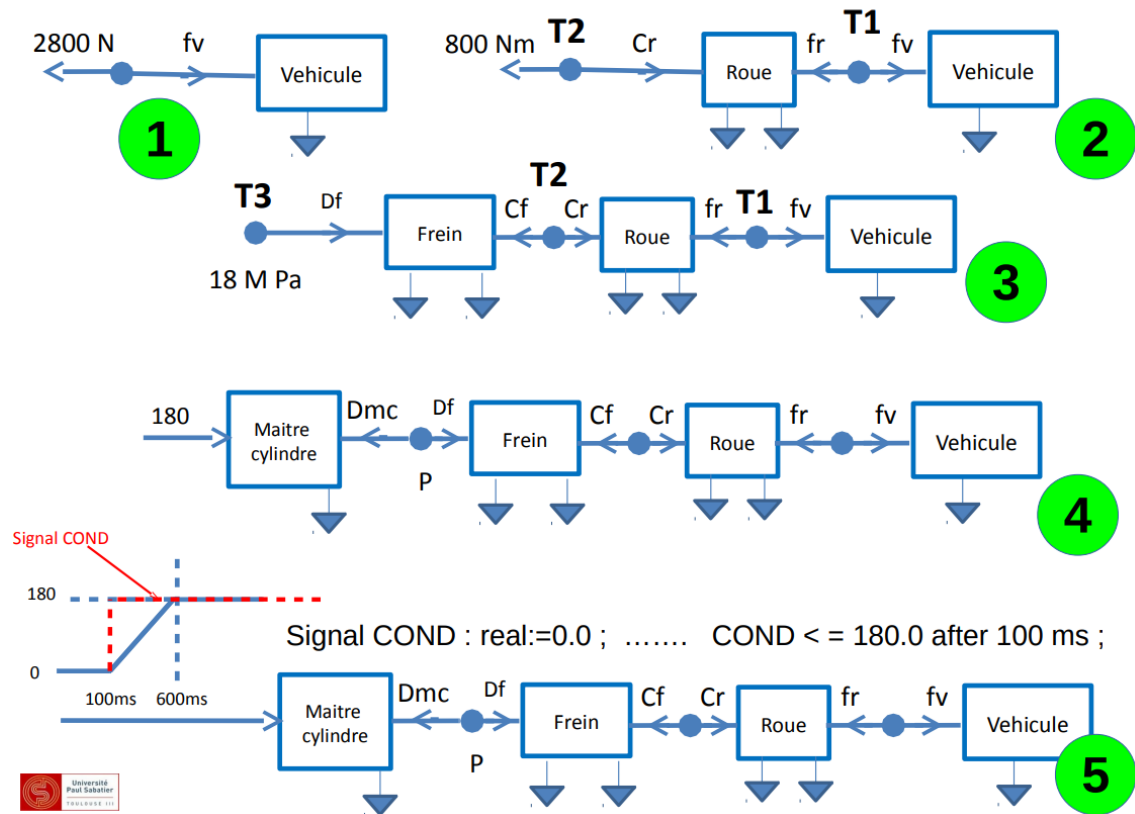


FIGURE 1 – Implémentation des différentes parties du système de freinage d'une voiture

Il est demandé d'instancier étapes par étapes différents blocs avec leurs terminaux coresspondants, comme indiqué dans le powerpoint de présentation du TP.

1.1 Analyse instanciation véhicule

Dans cette première partie nous devons instancier le véhicule fournit dans les fichiers du projet. Sur la Figure 2 on peut voir le schéma bloc qu'il faut instancier et décrire par la suite son entité en VHDL-AMS.

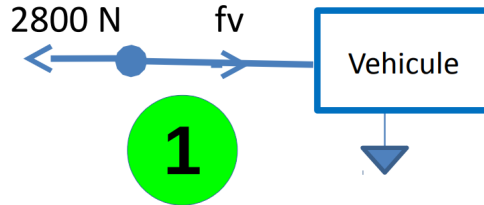


FIGURE 2 – Instanciation du véhicule avec son terminal

Dans l'entité du véhicule nous avons donc à définir certaines quantités qui va nous permettra de résoudre certaines équations :

- $m \rightarrow$ Masse véhicule.
- $C_x \rightarrow$ Coefficient aérodynamique.
- $S \rightarrow$ Surface Frontale.
- $V_{init} \rightarrow$ Vitesse initiale du vehicule.

```
15  -----
16  -- Test véhicule seul
17  -----
18  architecture A of test is
19
20  terminal t1 : translational_velocity;
21  quantity force through t1;
22
23  begin
24
25  force==2800.0;
26  Auto:  entity vehicule(one)
27  generic map
28  (
29    m=>m_veh, cx=>0.3,
30    S=>1.8, v_init=>28.0
31  )
32  port map(roue=>t1);
33  end A;
34
```

FIGURE 3 – Code VHDL Architecture A - Véhicule

Pour la première partie, véhicule il faudra mettre en place un terminal T1 ayant pour type "Translational_velocity". Celui-ci sera plus tard relié au Terminal "Roue", du même type, pour nous permettre la visualisation de la vitesse du véhicule.

Avec l'aide du logiciel ModelSim nous pouvons simuler le temps de freinage nécessaire à l'arrêt du véhicule lorsqu'on lui applique une force de 2800 N au terminal T1.

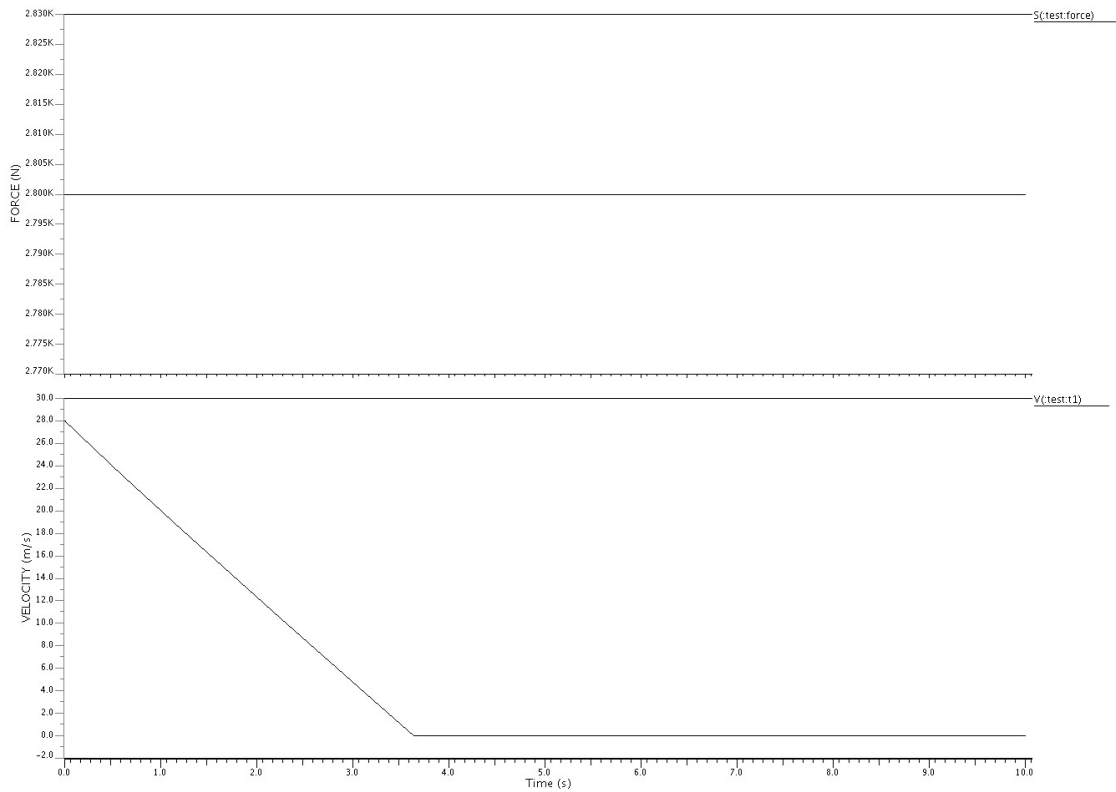


FIGURE 4 – Simulation Terminal T1 Force et Temps pour le freinage

1.2 Analyse instanciation roue

Une fois le véhicule instancié on nous demande d'ajouter le système "Roue". Dans cette partie on déclare en code la partie générique suivante :

- $route$ → Etat de la route (mouillée, sèche).
- m → Masse vehicule.
- rR → Rayon de la roue.
- IR → Inertie de la roue.
- μ_{0D} → Coefficient adhérence sans glissement sur sol sec.
- As → Facteur de décroissance de l'adhérence.
- μ_{0W} → Coefficient adhérence sans glissement sur sol mouillé.
- Vc → Caractéristique de la surface.

Dans cette partie du code il y aura deux terminaux à instancier au niveau de la roue :

- *veh* → De type "translational_velocity".
- *frein* → De type "rotational_velocity".

```
56 -----
57 -- Test véhicule + roue
58 -----
59 architecture C of test is
60
61     terminal t1 : translational_velocity;
62     terminal t2 : rotational_velocity;
63     quantity couple through t2;
64
65     begin
66
67         couple==800.0;
68
69         Auto: entity vehicule(one)
70         generic map
71         (
72             m=>m_veh,cx=>0.3,
73             S=>1.8,v_init=>28.0
74         )
75         port map(roue=>t1);
76         URoue: entity roue(A)
77         generic map
78         (
79             route=>seche, m=>m_veh,
80             rR=>0.275, IR=>0.4, mu0_D=>1.0,
81             As=>0.01, mu0_W=>0.5, Vc=>27.8
82         )
83     port map(veh=>t1,frein=>t2);
84 end C;
```

FIGURE 5 – Code VHDL Architecture C - Roue

Une fois la "Roue" implémentée on se retrouve sur ModelSim avec deux Terminaux, T2 étant le dernier terminal instancié qui représente le terminal d'entrée de "Roue" et T1 son Terminal de sortie. On observe grâce à ModelSim la vitesse rotationnelle de "Roue" ainsi que la vitesse du véhicule. On observe que la vitesse de "Roue" chute brusquement avant de réduire linéairement comme la vitesse du véhicule.

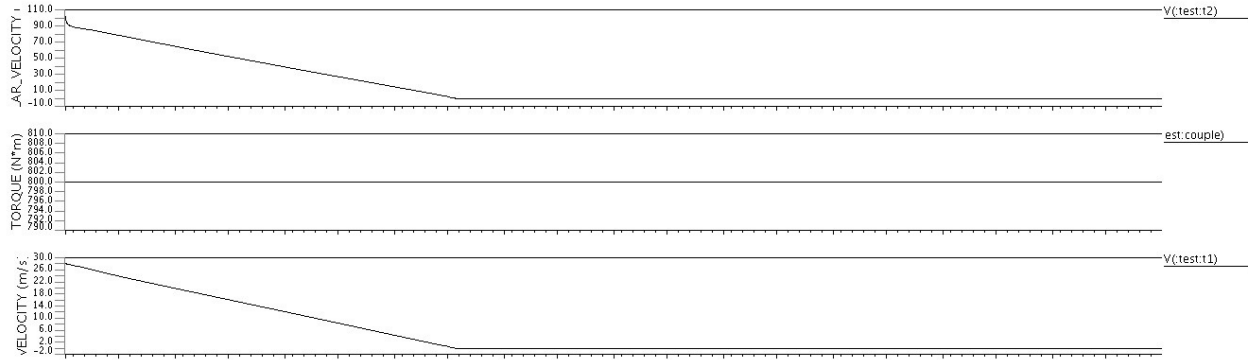


FIGURE 6 – Simulation Terminal T2 et T1

1.3 Analyse instantiation frein

Une fois le véhicule et la Roue instanciés on nous demande d'ajouter le système "Frein". Dans cette partie on déclare en code la partie générique suivante :

- $coef_{fric}$ → Coefficient de friction des plaquettes.
- S → Surface piston = 10 cm².
- R → Rayon moyen du disque.

Dans cette partie du code il y aura deux terminaux à instancier au niveau de la du Frein :

- $Roue$ → Type "rotational_velocity".
- MC → Type "fluidic" à l'entrée du Frein.

Dans cette nouvelle instance nous venons d'implémenter le "Frein" permettant l'obtention de la vitesse de rotation du disque dans le terminal 2 en fonction de la pression appliquée à son entrée. On définit une quantité de Pression (across) et débit (Through) pour le nouvel Terminal. On vient ensuite fixer la valeur de pression à 18 MPa pour pouvoir résoudre l'équation.

```

85 -----
86 -- Test véhicule + roue + frein
87 -----
88 architecture D of test is
89
90     terminal t1 : translational_velocity;
91     terminal t2 : rotational_velocity;
92     terminal t3 : fluidic;
93     quantity pression across debit through t3;
94     begin
95
96     pression == 18000000.0;
97
98     Auto: entity vehicule(one)
99     generic map (m=>m_veh,cx=>0.3,S=>1.8,v_init=>28.0)
100     | port map(roue=>t1);
101     URoue: entity roue(A)
102     generic map
103     (
104         route=>seche, m=>m_veh,
105         rR=>0.275, IR=>0.4, mu0_D=>1.0,
106         As=>0.01, mu0_W=>0.5, Vc=>27.8
107     )
108     port map(veh=>t1,frein=>t2);
109     Ufrein: entity frein(one)
110     generic map
111     (
112         coef_fric=>0.36, S=>1.0e-3, R=>0.12
113     )
114     | port map(roue=>t2, MC=>t3);
115 end D;

```

FIGURE 7 – Code VHDL Architecture D - Roue

1.4 Analyse instanciation maître cylindre

a

1.5 Modélisation du régulateur de pression

a

2 Modélisation du système de freinage équipé d'un ABS

3 Intégration simplifiée du moteur thermique

4 Conclusion