

# Projet GE S7

Asservissement d'une machine à courant continu  
Électrotechnique - Module 3

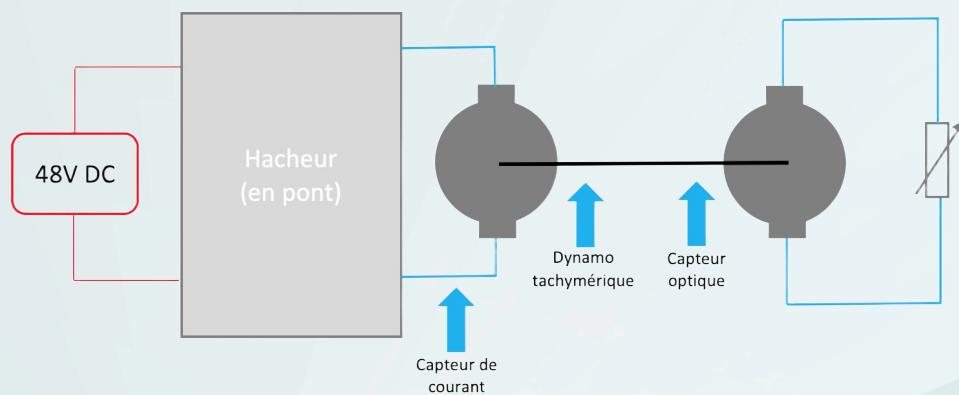
**TRENCHANT Evan**

**TROULLIER Laël**

**VIRQUIN Rudy**

*GE4, groupe 2*

## RAPPORT FINAL



# Table des matières

<b>1 Cahier des charges</b>	<b>3</b>
1. Notations des variables pour le projet . . . . .	3
2. Spécifications du cahier des charges . . . . .	3
<b>2 Planification du projet</b>	<b>4</b>
<b>3 Étude de la MCC et de son hacheur</b>	<b>5</b>
1. Modélisation de la MCC à vide . . . . .	5
2. Modélisation de la MCC avec charge . . . . .	9
3. Modélisation de la MCC avec charge et hacheur . . . . .	12
4. Simplification du schéma bloc . . . . .	15
<b>4 Étude de l'asservissement de la MCC</b>	<b>16</b>
1. Asservissement en courant . . . . .	16
2. Asservissement en vitesse avec dynamo tachymétrique . . . . .	19
3. Asservissement en vitesse avec codeur incrémental . . . . .	22
<b>5 Dimensionnement des asservissements</b>	<b>26</b>
1. Réalisation des soustracteurs . . . . .	26
2. Dimensionnement des correcteurs réels . . . . .	26
3. Dimensionnement des autres composants de la chaîne d'asservissement . . . . .	29
<b>6 Conception des PCB</b>	<b>34</b>
<b>7 Assemblage du système</b>	<b>36</b>
1. Commande des composants . . . . .	36
2. Soudure du PCB . . . . .	36
3. Réalisation du boitier . . . . .	37
<b>Conclusion</b>	<b>40</b>
<b>Liste des figures et chemins d'accès</b>	<b>41</b>

# 1 - Cahier des charges

## 1. Notations des variables pour le projet

Le tableau suivant récapitule l'ensemble des notations utilisées dans ce projet, ainsi que les valeurs et unités correspondantes pour notre moteur à courant continu.

Notations	Nom	Valeurs	Unités
<b>Paramètres nominaux</b>			
$M_0$	couple en rotation lente	0,54	N.m
$I_0$	courant permanent rotation lente	4,5	A
$U$	tension d'alimentation de définition	49	V
$N$	vitesse de définition	3 000	tr/min
<b>Paramètres maximaux</b>			
$U_{max}$	tension maximale	65	V
$N_{max}$	vitesse maximale	4 800	tr/min
$I_{max}$	courant impulsif	13	A
<b>Constantes électromécaniques</b>			
$K_e$	constante de fem (à 25°C)	13,3	V/(1000 tr/min)
$K_\Phi$	constante de couple électromagnétique	0,127	N.m/A
<b>Paramètres de frottement</b>			
$T_f$	couple de frottement sec	2,4	N.cm
$K_d$	coefficient de viscosité	0,53	N.cm/(1000 tr/min)
<b>Paramètres électriques</b>			
$R$	résistance du bobinage (à 25°C)	1,52	Ω
$L$	inductance du bobinage	2,2	mH
$R_{ch}$	résistance de charge	10	Ω
<b>Paramètres mécaniques</b>			
$J$	inertie du rotor	$8,3 \cdot 10^{-5}$	kg.m <sup>2</sup>
$M$	masse du moteur	1,34	kg
$T_{th}$	constante de temps thermique	7	min
<b>Variables dynamiques</b>			
$\Omega$	vitesse de rotation		rad/s
$C_{em}$	couple électromagnétique		N.m
$C_r$	couple résistant total		N.m
$i$	courant d'induit		A
$u$	tension d'alimentation		V
$e$	force contre-électromotrice		V

## 2. Spécifications du cahier des charges

Le cahier des charges du projet est défini comme suit :

### Asservissement en courant

- Temps de réponse maximal de 10 fois la période de la MLI soit 0,45 ms
- Dépassemement maximal de 20%

### Asservissement en vitesse

- Temps de réponse à 5% divisé par 3 par rapport à la boucle ouverte
- Dépassemement maximal de 20%

## 2 - Planification du projet

La réalisation de ce projet d'asservissement d'une machine à courant continu nécessite une planification rigoureuse pour garantir l'atteinte des objectifs dans les délais impartis. Cette section présente l'organisation temporelle du projet et le diagramme de Gantt détaillant les différentes phases de développement.

### Diagramme de Gantt

Le diagramme de Gantt ci-dessous illustre la planification détaillée du projet s'étendant de septembre à janvier, avec les dépendances entre les tâches et les jalons importants.

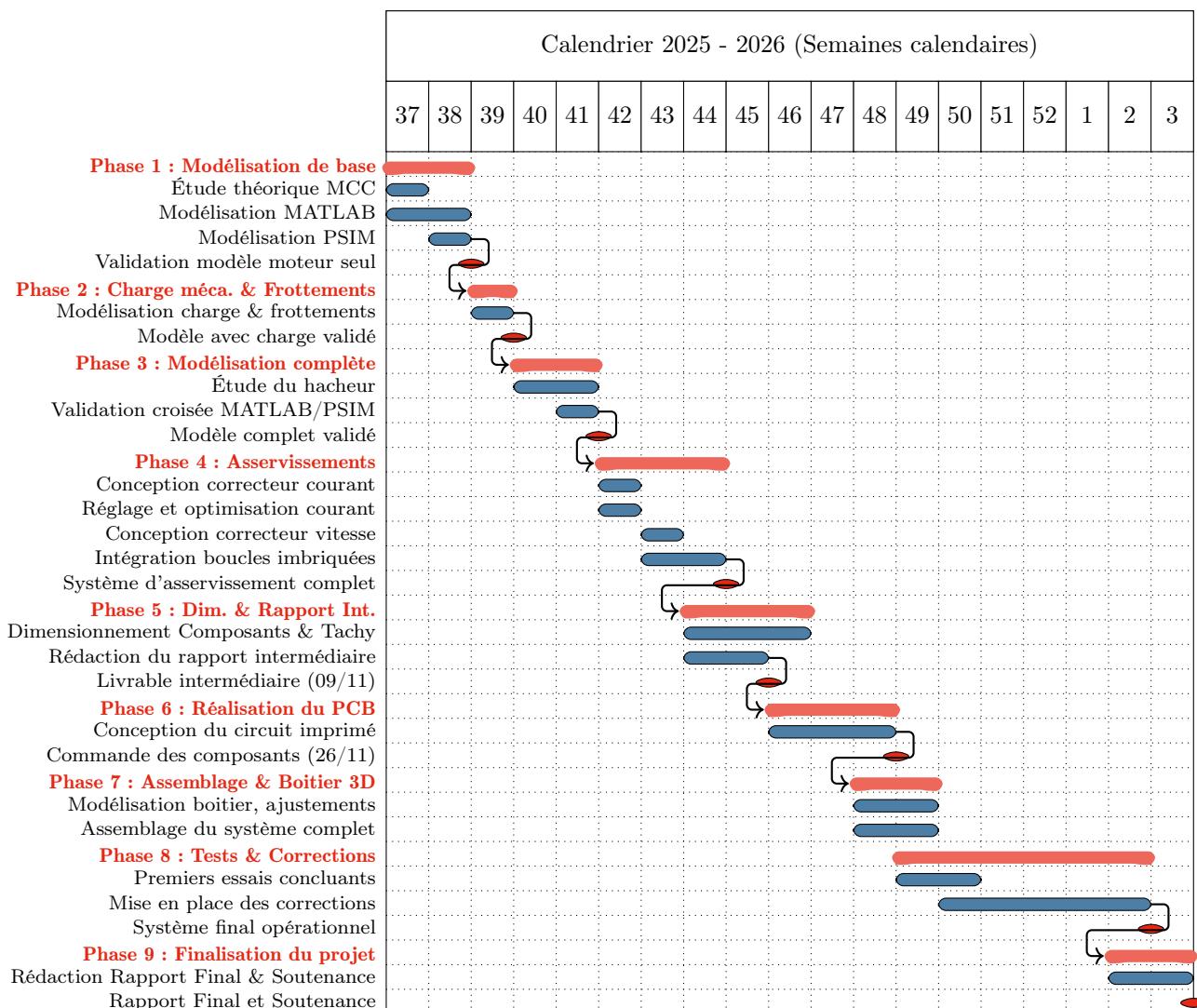


FIGURE 2.1 – Diagramme de Gantt étendu du projet (Septembre - Janvier)

## 3 - Étude de la MCC et de son hacheur

### 1. Modélisation de la MCC à vide

#### 1.1. Équations de la MCC

La machine à courant continu (MCC) est un dispositif électromécanique capable de convertir l'énergie électrique en énergie mécanique, et inversement. Cette conversion bidirectionnelle s'effectue grâce au passage d'un courant continu dans son induit, générant ainsi un couple moteur qui entraîne l'arbre en rotation. En mode générateur, le principe est inversé : la rotation de l'arbre produit une force électromotrice aux bornes de la machine. La MCC constitue ainsi un élément clé dans de nombreux systèmes d'entraînement industriels et de conversion d'énergie.

#### Formules d'une MCC sans pertes et sans charge

Les équations régissant le comportement d'une machine à courant continu sans pertes et sans charge sont les suivantes :

##### Équations temporelles de la MCC

$$u(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t)$$

$$C_{em}(t) = K_c \cdot i(t)$$

$$e(t) = K_e \cdot \Omega(t)$$

$$J \frac{d\Omega}{dt} = C_{em}(t)$$

On remplacera ici  $K_c$  et  $K_e$  par  $K_\Phi$ .

#### Transformée de Laplace

Les équations précédentes peuvent être exprimées dans le domaine de Laplace, ce qui facilite l'analyse et la modélisation du système :

##### Équations dans le domaine de Laplace

$$I(s) = \frac{U(s) - K_\Phi \Omega}{R + L \cdot s}$$

$$C_{em}(s) = K_\Phi \cdot I(s)$$

$$E(s) = K_\Phi \cdot \Omega(s)$$

$$\Omega(s) = \frac{K_\Phi \cdot I(s)}{J \cdot s}$$

## Conversion vitesse-tension

Le moteur possède une force contre-électromotrice  $E$  proportionnelle à sa vitesse de rotation. Les données constructeur indiquent :  $E = 13,3$  V pour 1000 tr/min.

En notant  $N$  la vitesse en tr/min et  $\Omega$  la vitesse en rad/s, on obtient les relations suivantes :

$$E = \frac{13,3}{1000} \cdot N$$

$$\Omega = \frac{\pi N}{30} \Rightarrow E = \underbrace{\frac{13,3}{1000} \cdot \frac{30}{\pi}}_{0,127} \cdot \Omega$$

La constante de fcem est donc :  $K_\Phi = 0,127$  V/(rad/s).

Une fois le modèle mathématique de la MCC défini à partir des équations présentées ci-dessus, nous pouvons procéder à sa simulation numérique. Cette simulation sera réalisée successivement sur deux environnements logiciels complémentaires : Simulink, qui permet une approche par schéma-blocs, et PSIM, orienté vers la simulation de circuits de puissance. La confrontation des résultats obtenus avec les deux outils permettra de valider notre modélisation théorique.

### 1.2. Simulation avec Simulink

Dans l'environnement Simulink, notre démarche consiste à traduire le système d'équations différentielles sous forme d'un schéma-blocs interconnectés, où chaque bloc représente une opération mathématique élémentaire.

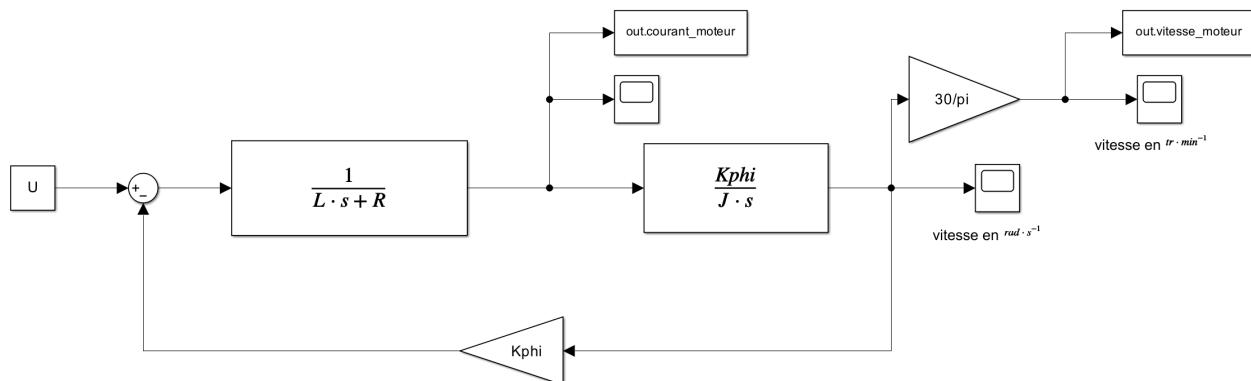


FIGURE 3.1 – Schéma Simulink du moteur à courant continu à vide

On remarque que le schéma Simulink présenté en Figure 3.1 intègre les équations de la MCC sans charge. Les simulations effectuées avec ce modèle permettent d'obtenir les réponses temporelles du courant et de la vitesse du moteur à vide, illustrées dans la Figure 3.2.

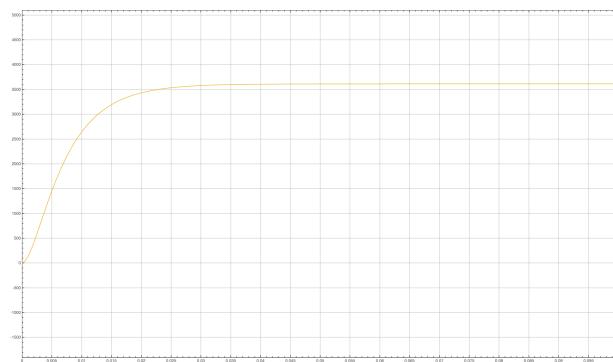
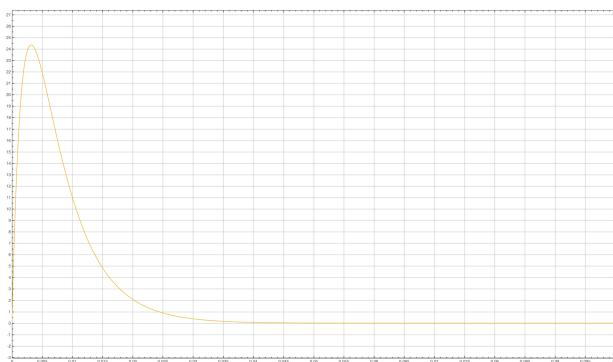


FIGURE 3.2 – Résultats de simulation des réponses en courant (gauche) et en vitesse (droite) du moteur à vide

### 1.3. Simulation sur PSIM

Pour la simulation sur PSIM, nous adoptons une approche méthodologique complémentaire. L'utilisation de deux méthodes distinctes (Simulink et PSIM) constitue une validation croisée : si les résultats concordent, nous pourrons confirmer la validité de nos modélisations. Contrairement à Simulink qui nécessite la construction du modèle à partir des équations différentielles, PSIM propose une approche plus directe basée sur l'utilisation de composants prédéfinis. Le logiciel met à disposition une bibliothèque de machines à courant continu standards qu'il suffit de paramétrier avec les caractéristiques de notre moteur réel. Cette méthodologie se rapproche davantage du fonctionnement physique du système et réduit significativement le nombre de calculs intermédiaires.

Le principe de modélisation sous PSIM repose sur l'assemblage de composants dont on définit ensuite les propriétés. Par exemple, pour simuler notre moteur, nous plaçons simplement un bloc « moteur DC » dans l'environnement de travail, puis nous renseignons ses paramètres électriques et mécaniques.

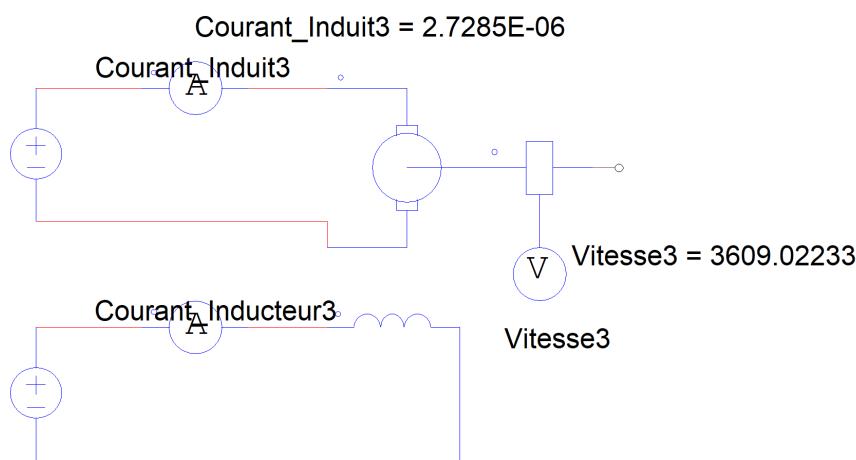


FIGURE 3.3 – Modélisation du moteur à courant continu sur PSIM

### 1.4. Comparaison des résultats

Nous comparons les résultats avec un programme *MATLAB*. Pour cela, nous avons besoin de récupérer les valeurs de notre modélisation sous forme de tableau :  $X = [t \ i(t) \ \Omega(t)]$ .

On retrouve bien les valeurs attendues, notamment en vitesse où on a la relation suivante à

l'état stable :

$$\Omega = \frac{U}{K_\Phi} = \frac{48}{0,127} = 377,95 \text{ rad/s} \quad \text{ou} \quad N = \frac{U}{K_e} = \frac{48}{13,3} = 3609 \text{ tr/min}$$

Les courbes de la Figure 3.4 montrent une excellente concordance entre les résultats obtenus avec Simulink et PSIM, validant ainsi notre modélisation du moteur à courant continu à vide. Nous pouvons donc passer aux simulations suivantes : le moteur avec des pertes et sa charge.

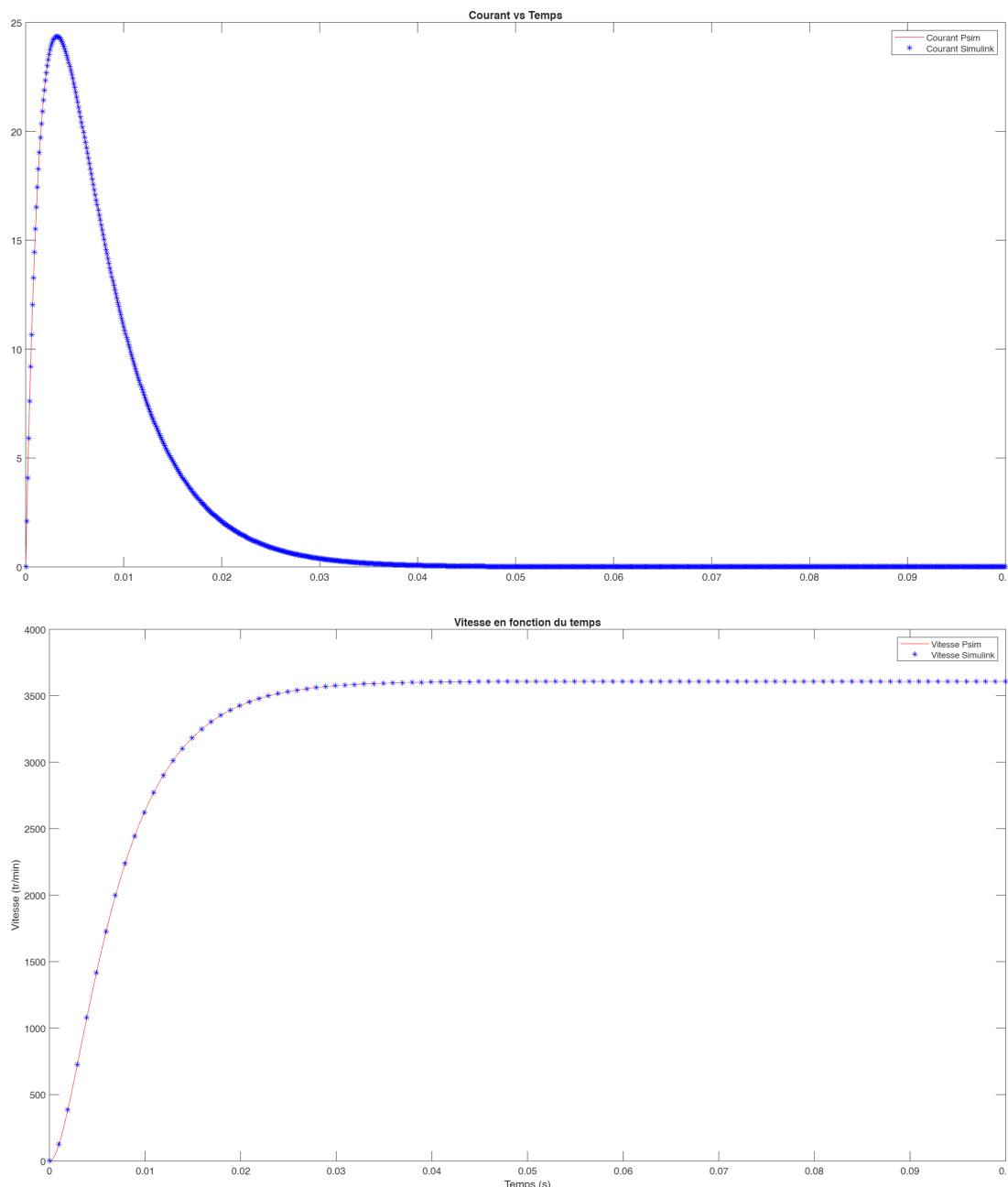


FIGURE 3.4 – Comparaison des résultats Simulink et PSIM pour le moteur à vide

## 2. Modélisation de la MCC avec charge

Dans cette partie, nous allons effectuer la même logique qu'avec le moteur à vide. Cette fois-ci le moteur a des pertes, appelées frottements, et le couple résistant du moteur est pris en compte. On note :

$$f = \frac{30}{\pi} \cdot \frac{1}{1000} \cdot 0,53 \cdot 10^{-2} = 5,06 \cdot 10^{-5} \text{ N.m/rad.s}$$

$$C_r = 2 \cdot C_0 = 0,024 \text{ N.m}$$

### 2.1. Équations de la MCC avec pertes

Les équations de notre système vont donc changer. En effet en ajoutant les frottements, le système devient :

Formules du moteur avec pertes

$$u(t) = R \cdot i(t) + L \frac{di(t)}{dt} + e(t)$$

$$C_{em}(t) = K_c \cdot i(t)$$

$$e(t) = K_e \cdot \Omega(t)$$

$$J \frac{d\Omega}{dt} = C_{em}(t) - 2f \cdot \Omega(t) - C_r(t)$$

Ce qui donne dans le domaine de Laplace :

Équations dans le domaine de Laplace

$$I(s) = \frac{U(s) - K_\Phi \Omega}{R + L \cdot s}$$

$$C_{em}(s) = K_\Phi \cdot I(s)$$

$$E(s) = K_\Phi \cdot \Omega(s)$$

$$\Omega(s) = \frac{C_{em} - C_r}{J \cdot s + f}$$

Une fois les équations trouvées, nous allons pouvoir déterminer le schéma bloc et effectuer les simulations.

## 2.2. Simulation Simulink

Le schéma bloc n'est plus le même. En effet, celui-ci devient plus complexe, puisque qu'on y ajoute le couple résistant  $C_r$  et les frottements  $f$ .

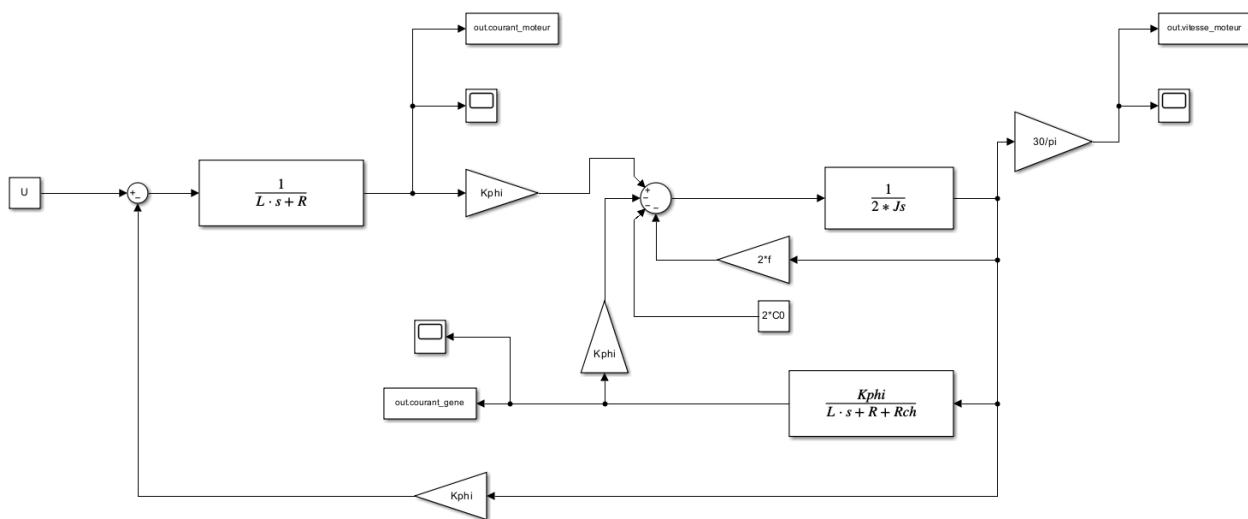


FIGURE 3.5 – Schéma Simulink du moteur à courant continu avec charge et frottements

## 2.3. Simulation sur PSIM

De manière analogue à la simulation du moteur à vide, nous utilisons PSIM pour valider le modèle Simulink incluant cette fois-ci la charge et les frottements. La démarche reste similaire : nous utilisons le bloc moteur à courant continu de PSIM, mais en y ajoutant les paramètres de couple résistant ( $C_r$ ) et de frottement visqueux ( $f$ ). Cette approche directe, basée sur des composants physiques, permet de vérifier la cohérence des résultats obtenus par la modélisation par équations dans Simulink.

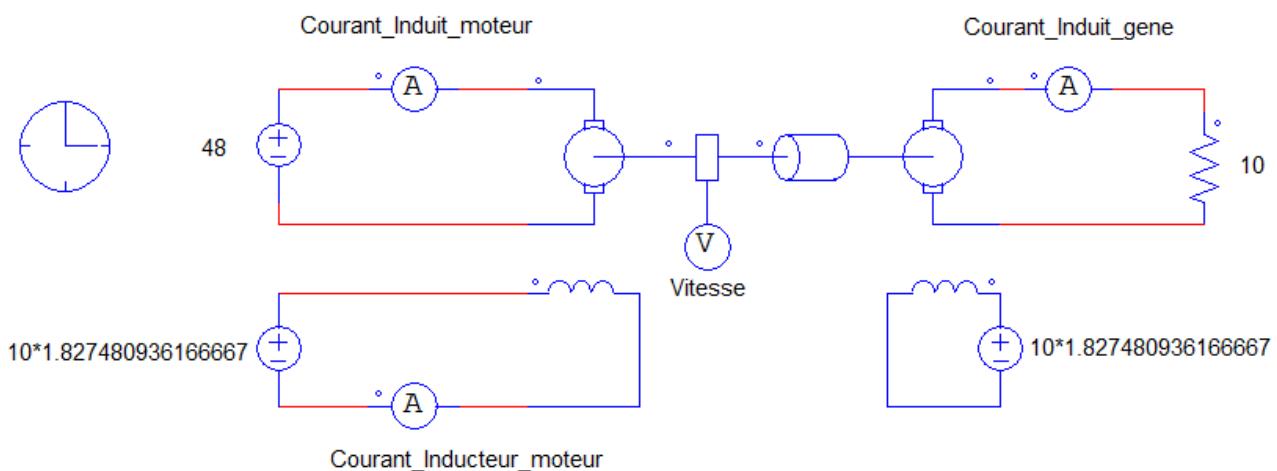


FIGURE 3.6 – Modélisation du moteur avec charge et frottements sur PSIM

## 2.4. Comparaison des résultats

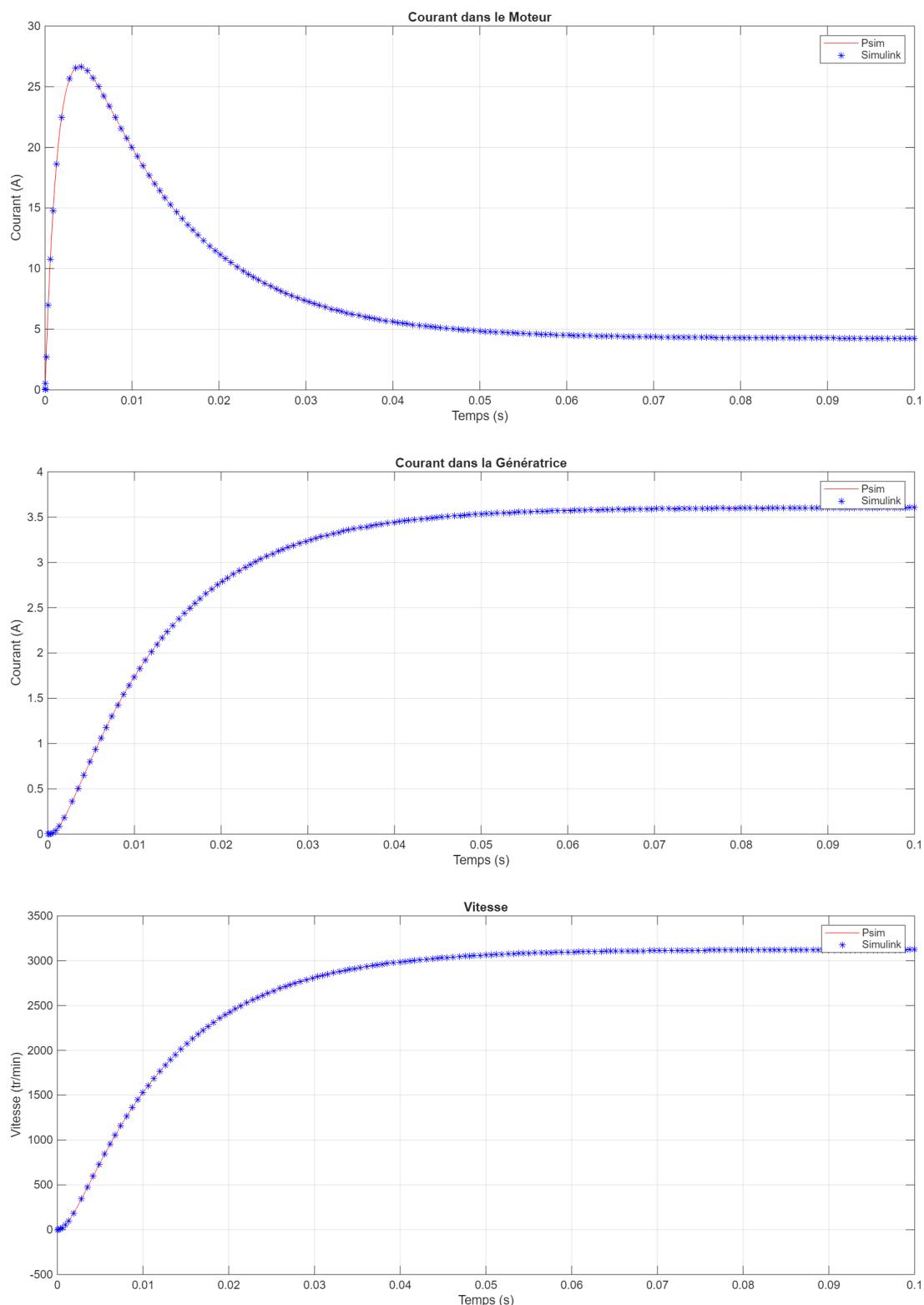


FIGURE 3.7 – Comparaison des résultats Simulink et PSIM pour le moteur avec charge

Les courbes de la Figure 3.7 confirment la validité de notre modélisation avec charge et frottements. On observe une superposition quasi-parfaite des résultats Simulink (marqueurs) et PSIM (ligne continue) sur l'ensemble des grandeurs mesurées. La vitesse atteint un régime permanent à environ 3100 tr/min, valeur inférieure à celle du moteur à vide (3609 tr/min) en raison de la présence de la charge de  $10 \Omega$ , du couple résistant  $C_r$  et des frottements visqueux. Le courant

d'induit du moteur présente un pic initial caractéristique du démarrage, puis se stabilise à une valeur permanente correspondant à l'équilibre entre le couple moteur et les couples résistifs. Cette concordance valide l'implémentation des pertes mécaniques dans les deux environnements de simulation.

### 3. Modélisation de la MCC avec charge et hacheur

#### 3.1. Modélisation du hacheur

Le hacheur, ou convertisseur continu–continu, est un dispositif d'électronique de puissance qui permet de modifier la valeur moyenne d'une tension continue à l'aide d'interrupteurs commandés. Il est couramment utilisé pour piloter des charges électriques (ici une machine à courant continu) avec un bon rendement.

Sur PSIM, le hacheur est modélisé par un pont en H composé de quatre transistors MOSFET pilotés en paires. Le schéma fonctionnel comporte le pont en H alimenté par une source continue (ici 48 V), un générateur de signal triangulaire et un comparateur qui produit le signal PWM de commande.

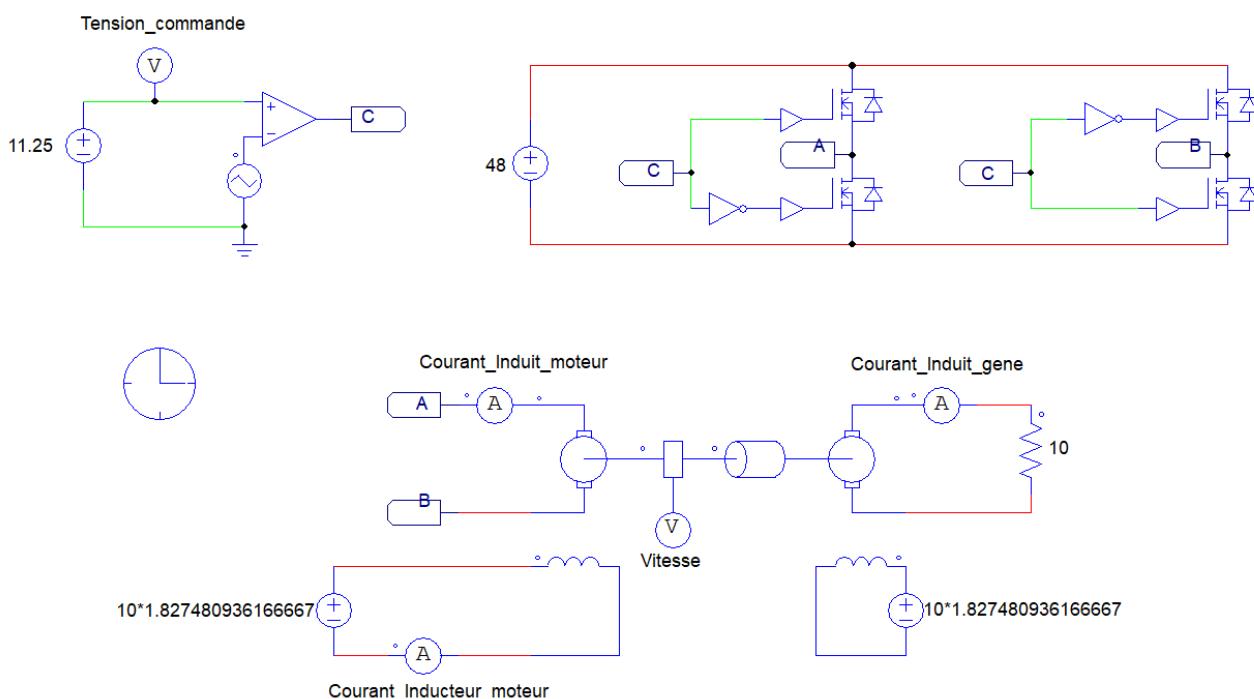


FIGURE 3.8 – Modélisation du hacheur sur PSIM

Le hacheur fournit une tension de sortie moyenne qui peut être positive ou négative (mode bipolaire). La valeur moyenne de la tension de sortie  $V_s$  dépend de trois grandeurs : la tension d'alimentation (ici 48 V), la tension de commande continue (notée  $V_{commande}$ ) et le rapport cyclique  $\alpha$  du signal de commande (duty cycle) appliqué aux interrupteurs.

La commutation des transistors s'effectue par paires (par exemple 1 et 4 d'une part, 2 et 3 d'autre part) de façon complémentaire afin de piloter le pont en H. Le rapport cyclique  $\alpha$  correspond à la proportion de la période pendant laquelle l'interrupteur est fermé. Pour  $\alpha = 0$  la tension de sortie vaut  $V_s = -48$  V, pour  $\alpha = 1$  on obtient  $V_s = +48$  V, et pour des valeurs intermédiaires de  $\alpha$  la tension de sortie varie de manière (approximativement) linéaire entre ces extrêmes. Ainsi, une augmentation de  $\alpha$  entraîne une augmentation de  $V_s$ .

La commande PWM est générée en comparant, au moyen d'un comparateur (ou d'un AOP configuré), une tension continue de référence  $V_e = V_{commande}$  avec un signal triangulaire  $V_{triangle}$

qui varie typiquement de 0 à 15 V. Le comparateur produit alors le signal  $V_{\text{pwm}}$  dont le rapport cyclique est proportionnel à  $V_{\text{commande}}$ . En sortie du comparateur, des buffers (bootstrappers) sont placés pour assurer une adaptation d'impédance et piloter correctement les étages de commande des MOSFET.

### 3.2. Caractéristique du hacheur

Le hacheur utilisé possède une caractéristique de transfert linéaire. La tension de sortie moyenne  $\langle u_{AB} \rangle$  dépend de la tension de commande  $u_{\text{commande}}$  selon la relation suivante :

**Relation tension de sortie :**

$$\begin{aligned}\langle u_{AB} \rangle &= \frac{96}{15} (u_{\text{commande}} - 7,5) \\ &= \frac{96}{15} u_{\text{commande}} - 48 \\ &= 6,4 u_{\text{commande}} - 48\end{aligned}$$

**Caractéristiques principales :**

- Pente : 6,4 V/V
- Tension maximale : 48 V à 15 V de commande
- Point de fonctionnement intermédiaire (en rouge) : (11,25 V; 24 V)

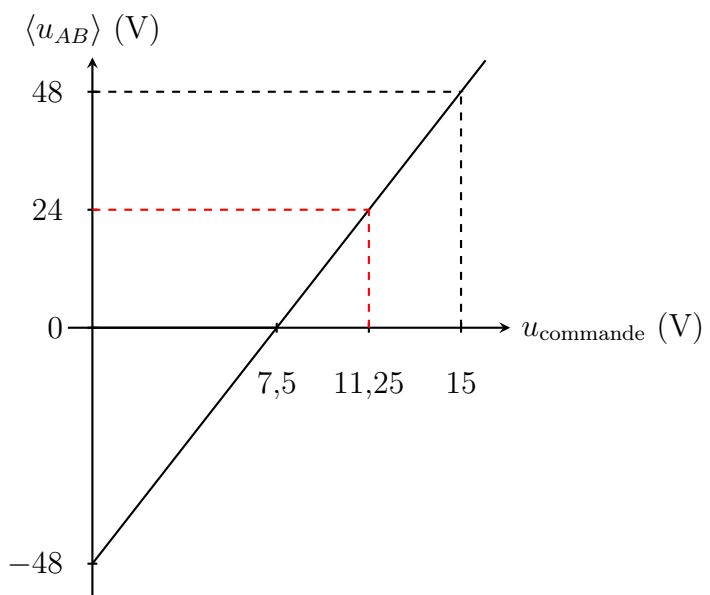


FIGURE 3.9 – Caractéristique de transfert du hacheur

Cette caractéristique montre que le hacheur présente un comportement linéaire avec un gain de 6,4. La tension de sortie est limitée à  $\pm 48$  V.

### 3.3. Simulation sur Simulink

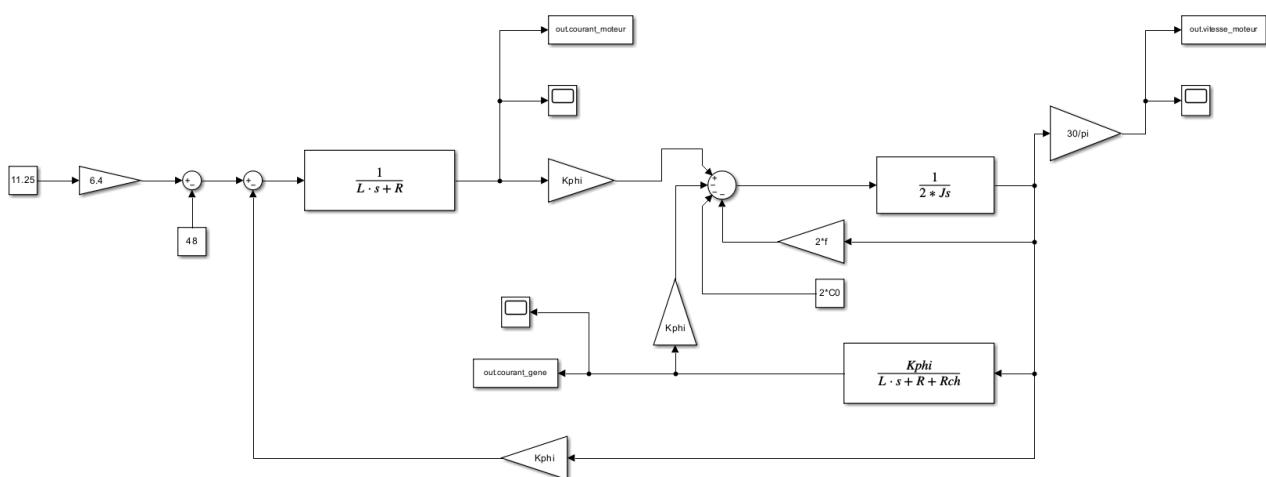


FIGURE 3.10 – Modélisation du moteur avec hacheur sur Simulink

### 3.4. Comparaison des résultats

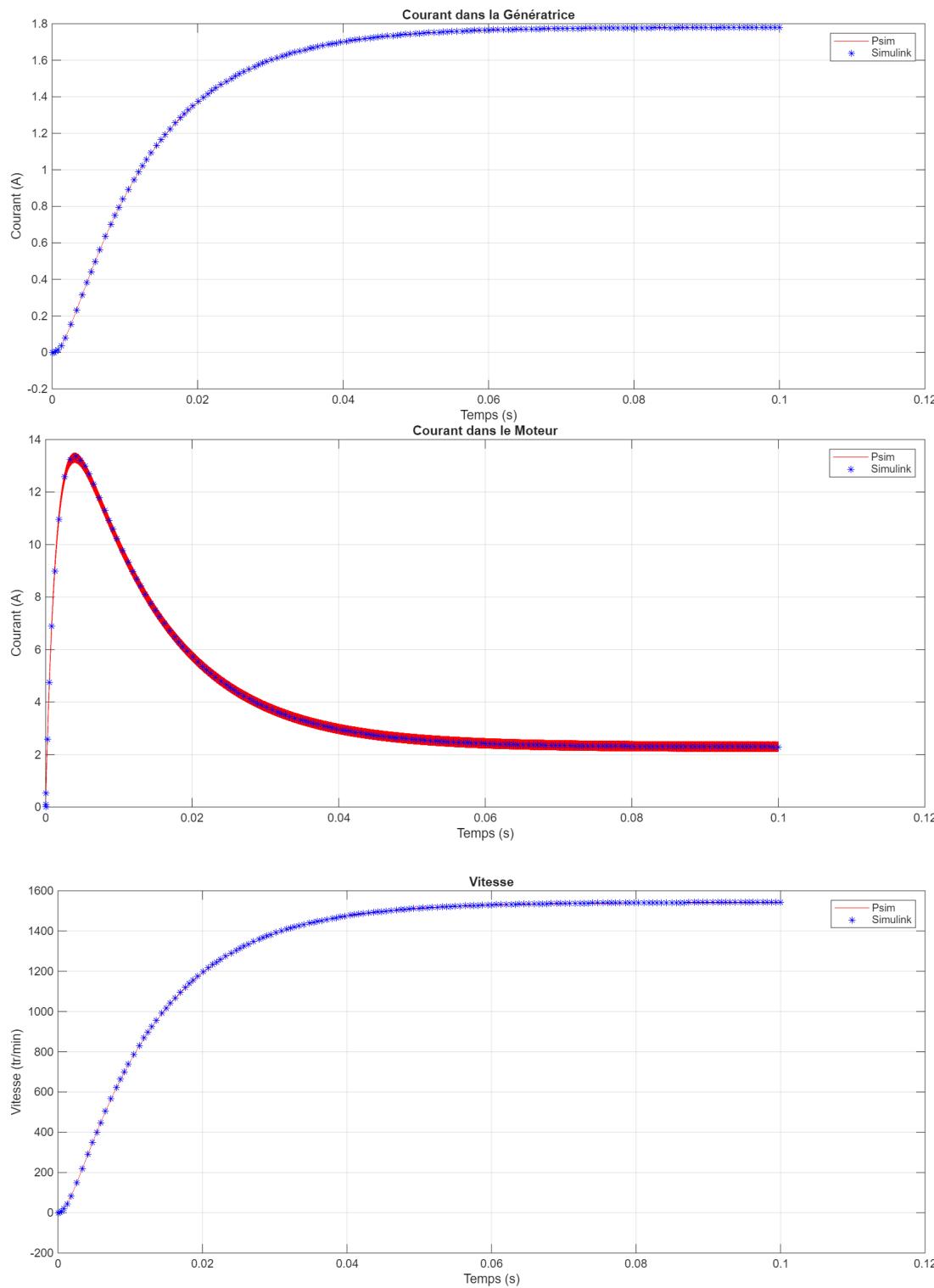


FIGURE 3.11 – Comparaison des résultats Simulink et PSIM pour le moteur avec hacheur

Les résultats présentés dans la Figure 3.11 démontrent la cohérence entre les modélisations Simulink et PSIM du système complet moteur-hacheur. Les trois grandeurs observées (courant génératrice, courant moteur et vitesse) montrent une excellente superposition des courbes issues des deux simulateurs. On constate notamment que l'introduction du hacheur impose une dynamique légèrement différente par rapport à l'alimentation directe, avec des ondulations

caractéristiques de la modulation PWM. La vitesse converge vers une valeur de régime permanent proche de 3100 tr/min, cohérente avec le point de fonctionnement imposé par la tension de commande du hacheur. Cette validation croisée confirme la justesse de notre modélisation du convertisseur statique et de son interaction avec la charge mécanique.

#### 4. Simplification du schéma bloc

A ce stade de l'analyse, nous disposons du schéma bloc complet de notre système, avant asservissement en vitesse. Comme illustré dans la Figure 3.12, ce schéma est assez complexe en raison de la présence de plusieurs boucles de rétroaction imbriquées. Pour faciliter l'analyse et la conception des contrôleurs, il est nécessaire de simplifier ce schéma bloc en combinant les différents éléments.

Cette simplification permet d'obtenir une représentation plus concise du système, facilitant ainsi l'analyse et la conception des contrôleurs.

$$\begin{cases} \varepsilon = e - \textcolor{blue}{B} \cdot \Omega - \textcolor{green}{C} \cdot \Omega \\ s = \textcolor{red}{A} \cdot \varepsilon = \textcolor{red}{A} \cdot (e - (\textcolor{blue}{B} + \textcolor{green}{C}) \cdot s) \end{cases}$$

$$\Rightarrow s = \frac{\textcolor{red}{A} \cdot e}{1 + \textcolor{red}{A} \cdot (\textcolor{blue}{B} + \textcolor{green}{C})} = H \cdot e$$

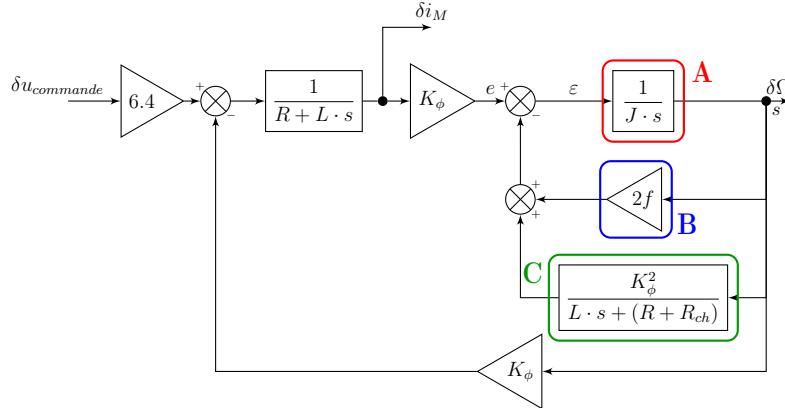


FIGURE 3.12 – Schéma bloc de l'asservissement en courant avec boucle de courant imbriquée

Le diagramme bloc simplifié de l'asservissement en vitesse est présenté dans la Figure 3.13.

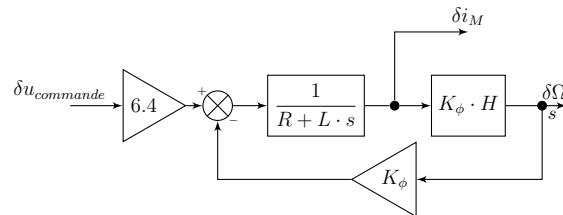


FIGURE 3.13 – Schéma bloc de l'asservissement en courant simplifié

On peut alors écrire la fonction de transfert en boucle ouverte du système simplifié :

$$\delta i_M(s) = \frac{1}{1 + \frac{(K_\phi)^2 \cdot H}{R + L \cdot s}} \cdot 6,4 \cdot \delta u_{commande}(s) = FTBO(s) \cdot 6,4 \cdot \delta u_{commande}(s)$$

Pour s'assurer que notre fonction de transfert est correcte, il faudrait la valider par des simulations numériques en comparant les résultats obtenus. Il serait alors possible de poursuivre l'analyse en utilisant cette fonction de transfert simplifiée pour concevoir et optimiser le contrôleur de l'asservissement en vitesse du moteur.

## 4 - Étude de l'asservissement de la MCC

### 1. Asservissement en courant

L'asservissement en courant du moteur constitue la boucle interne de notre système de régulation en cascade. Son rôle principal est d'assurer que le courant absorbé par le moteur suive fidèlement la consigne fournie par l'asservissement en vitesse. Cette régulation permet de contrôler précisément le couple moteur et d'améliorer la dynamique globale du système.

#### 1.1. Modélisation sur Simulink

L'asservissement en courant doit appliquer la consigne déterminée par l'asservissement en vitesse. Pour cela, nous mettons en œuvre un correcteur Proportionnel-Intégral (PI). Le cahier des charges impose un temps de réponse égal à 10 fois la période de la modulation de largeur d'impulsion (MLI), soit 0.45 ms, avec un dépassement maximal de 20 %.

À l'aide de l'outil PID Tuner de Simulink, nous obtenons un jeu de paramètres qui répondent au cahier des charges pour le correcteur PI :

- Gain proportionnel :  $K_p = 5,053$
- Gain intégral :  $K_i = 36\,349,6$

Il est important de souligner que la notation du correcteur PI diffère entre Simulink et PSIM. Sur PSIM, la valeur du paramètre  $\tau$  représente le rapport  $\frac{K_p}{K_i}$ , tandis que sur Simulink, les deux gains sont exprimés directement.

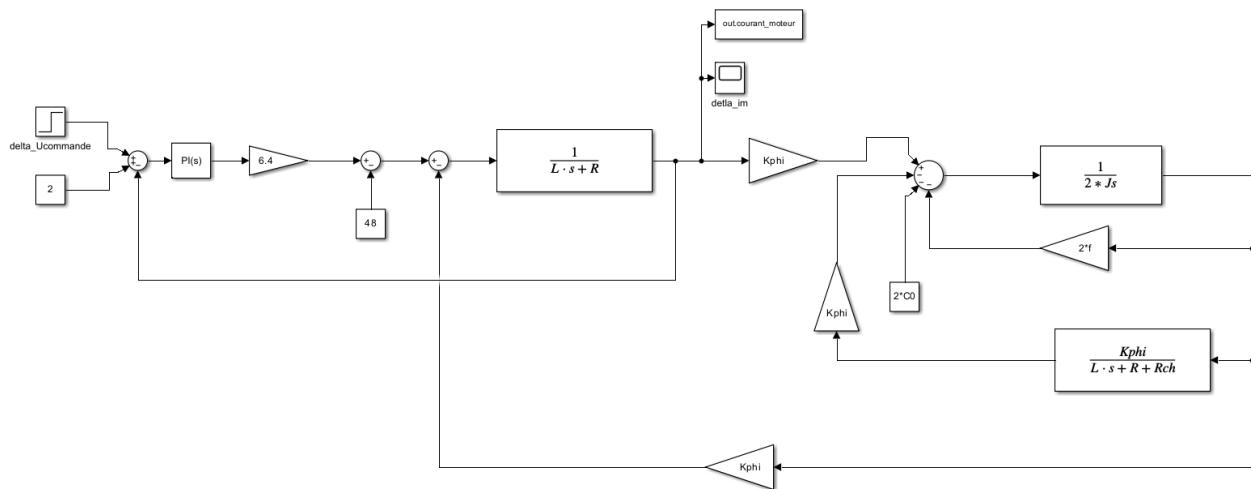


FIGURE 4.1 – Schéma complet de l'asservissement en courant du moteur dans Simulink

#### 1.2. Modélisation sur PSIM

Pour adapter le correcteur PI de Simulink à PSIM, il faut convertir les paramètres en utilisant les relations suivantes :

Dans Simulink :

$$G(s) = K_p + \frac{K_i}{s} = K_p \frac{s + \frac{K_i}{K_p}}{s} = K_p \frac{1 + \frac{K_p}{K_i} \cdot s}{\frac{K_p}{K_i} \cdot s}$$

Dans PSIM :

$$G(s) = k \frac{1 + T \cdot s}{T \cdot s}$$

D'où la correspondance :

$$k = K_p \quad \text{et} \quad T = \frac{K_p}{K_i}$$

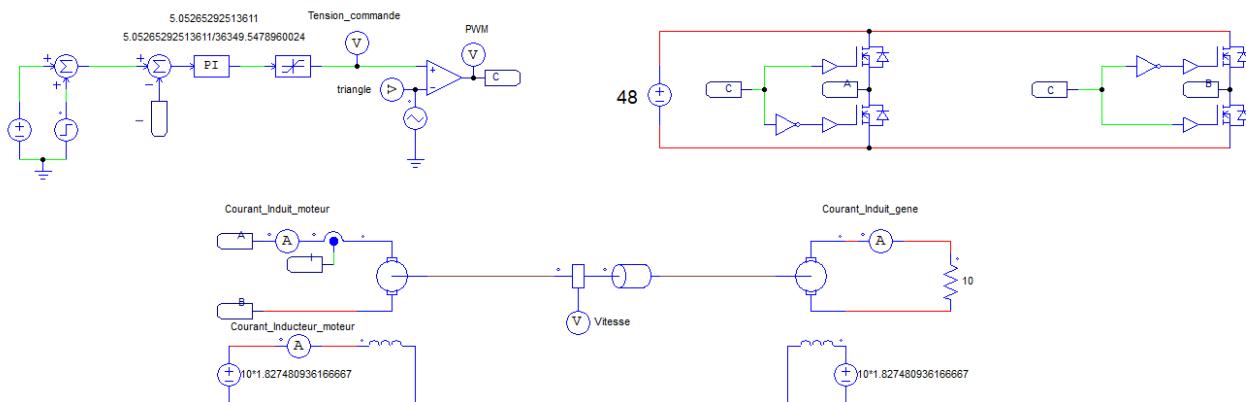


FIGURE 4.2 – Schéma complet de l'asservissement en courant du moteur dans PSIM

### 1.3. Comparaison des résultats

Dans cette section, nous allons comparer les résultats obtenus avec les deux outils de simulation, Simulink et PSIM.

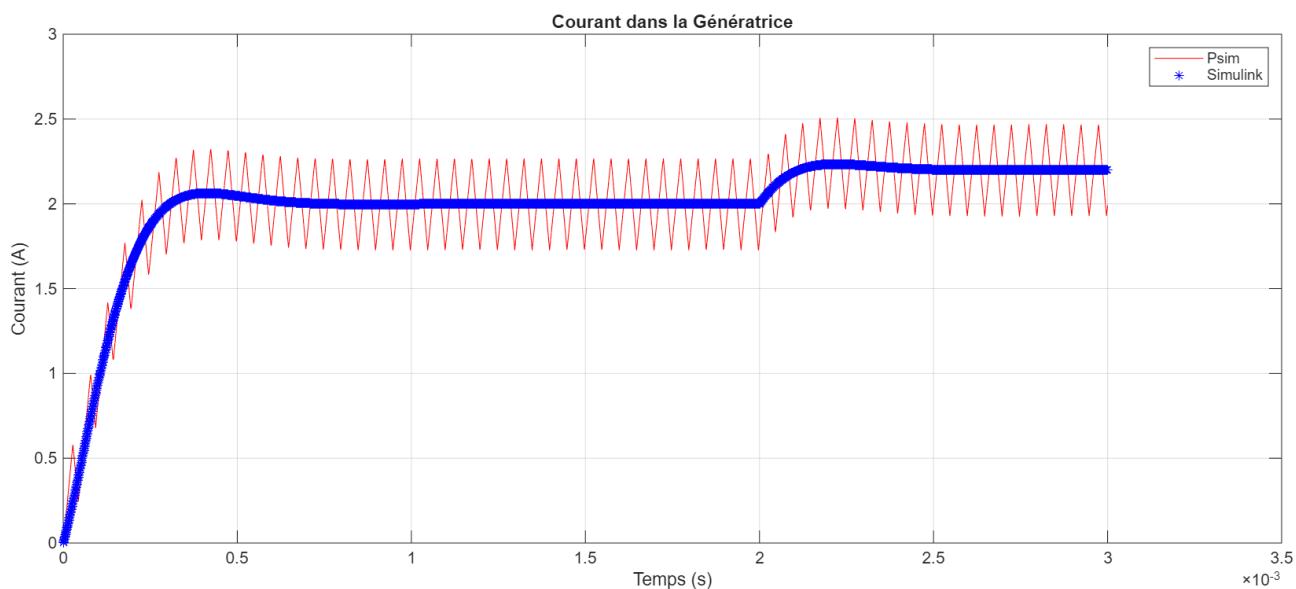


FIGURE 4.3 – Comparaison des réponses indicielles de l'asservissement en courant du moteur entre Simulink et PSIM

La figure 4.3 illustre la comparaison des réponses indicielles de l'asservissement en courant du moteur entre Simulink et PSIM. Les deux courbes sont quasiment superposées, ce qui confirme la cohérence des modélisations réalisées dans les deux environnements.

On constate que la réponse indicelle de l'asservissement en courant du moteur présente un dépassement d'environ 19% et un temps de réponse de 0,35 ms. Ce qui respecte les spécifications

du cahier des charges qui exige un dépassement maximal de 20% et un temps de réponse maximal de 10 fois la période de la MLI soit 0,45 ms.



## 2. Asservissement en vitesse avec dynamo tachymétrique

L'asservissement en vitesse constitue la boucle externe et la régulation principale de notre système de contrôle en cascade. Contrairement à l'asservissement en courant qui opère sur une grandeur électrique interne, l'asservissement en vitesse agit directement sur la grandeur mécanique d'intérêt pour l'application finale. C'est donc cet asservissement qui détermine les performances réelles du système vis-à-vis des exigences du cahier des charges.

La régulation en vitesse permet de garantir que le moteur atteint et maintient la vitesse de rotation souhaitée, indépendamment des perturbations externes telles que les variations de charge ou les frottements. Elle assure également la rapidité de la réponse du système face aux changements de consigne, tout en limitant les dépassements pour éviter les sollicitations mécaniques excessives.

Dans cette section, nous utilisons une dynamo tachymétrique comme capteur de vitesse. Ce capteur analogique fournit une tension proportionnelle à la vitesse de rotation, ce qui permet une mise en œuvre simple et une mesure continue de la vitesse.

### 2.1. Principe de fonctionnement de la dynamo tachymétrique

La dynamo tachymétrique est une petite génératrice à courant continu couplée mécaniquement au moteur. Elle génère une tension proportionnelle à sa vitesse de rotation selon la relation :

$$V_{\text{tachy}} = K_{\text{tachy}} \cdot \Omega$$

où  $K_{\text{tachy}}$  est la constante de la dynamo tachymétrique (en V/(rad/s)) et  $\Omega$  la vitesse angulaire du moteur.

Cette caractéristique linéaire en fait un capteur particulièrement adapté pour les asservissements analogiques, car la tension de sortie peut être directement utilisée comme signal de rétroaction sans nécessiter de conversion numérique-analogique.

### 2.2. Caractérisation en boucle ouverte

Nous commençons par la simulation de la vitesse du moteur en boucle ouverte sur Simulink afin d'établir une référence de performance. Nous réalisons le schéma suivant :

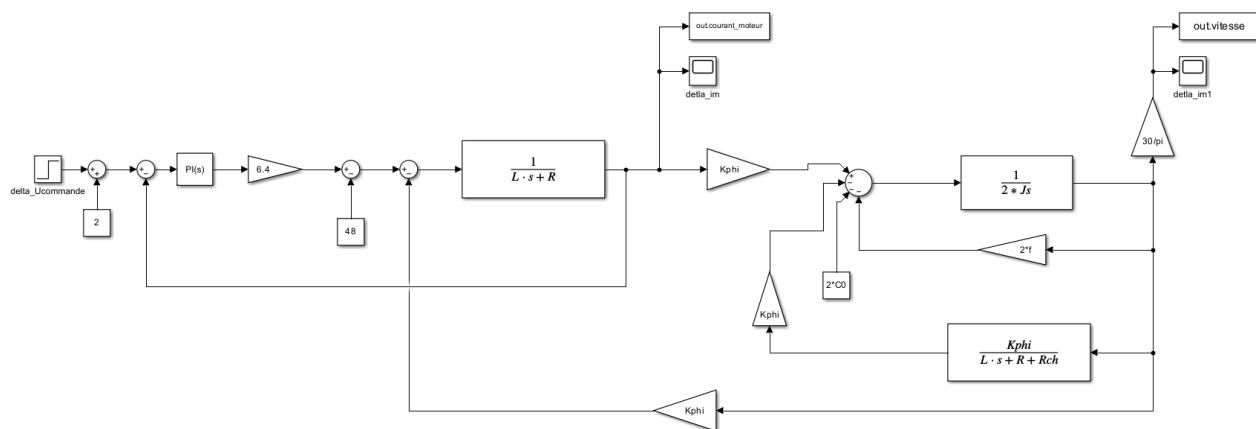


FIGURE 4.4 – Schéma de la simulation en boucle ouverte de la vitesse du moteur dans Simulink

La simulation en boucle ouverte nous permet d'observer la réponse naturelle du système sans rétroaction. Cette étape est essentielle pour comprendre les limites intrinsèques du système et

établir les objectifs d'amélioration pour l'asservissement. Pour effectuer cette caractérisation, nous attendons d'abord que le système atteigne son régime permanent, puis nous appliquons une consigne en échelon pour observer la dynamique de réponse.

L'analyse de la réponse indicielle révèle un temps de réponse à 95 % en boucle ouverte de 322.1 ms. Cette valeur relativement élevée indique une dynamique lente qu'il faudra améliorer.

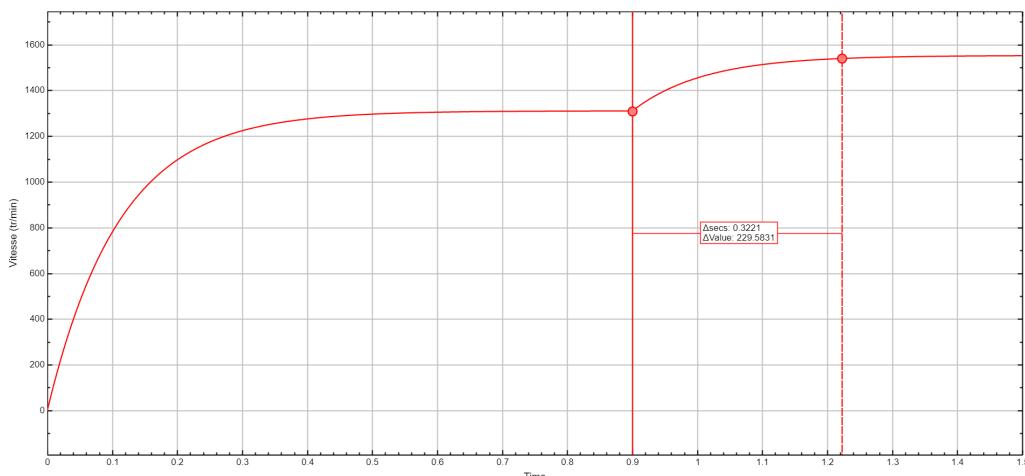


FIGURE 4.5 – Réponse de la vitesse du moteur en boucle ouverte dans Simulink

### 2.3. Conception du correcteur PI

Face aux limitations observées en boucle ouverte, nous mettons en œuvre un asservissement en boucle fermée avec un correcteur PI. L'objectif de cet asservissement est d'améliorer drastiquement les performances dynamiques du système. Nous visons un temps de réponse à 5 % trois fois plus rapide qu'en boucle ouverte, soit 107.37 ms, tout en maintenant un dépassement maximal de 20 % pour limiter les contraintes mécaniques.

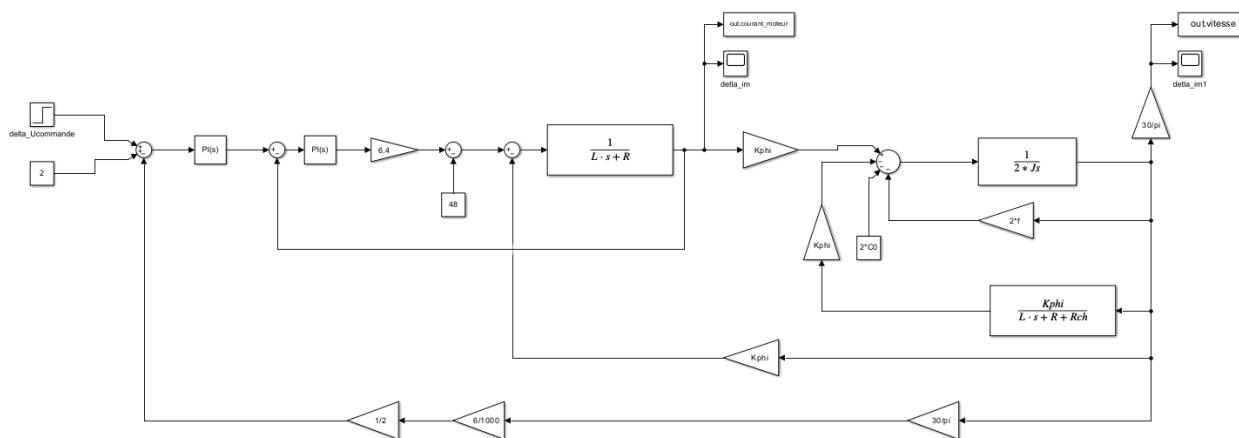


FIGURE 4.6 – Schéma de l'asservissement en vitesse du moteur dans Simulink

Le schéma présente l'architecture complète du système de régulation en cascade : la boucle externe d'asservissement en vitesse génère une consigne de courant qui est ensuite régulée par la boucle interne d'asservissement en courant. Cette structure en cascade permet de bénéficier d'une meilleure robustesse et d'une dynamique optimisée.

À l'aide de l'outil PID Tuner de Simulink, nous déterminons les paramètres optimaux du correcteur PI permettant de satisfaire les spécifications temporelles imposées :

- ▶ Figure 4.5: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/BO\_Vitesse/BO\_Vitesse\_VALIDE\_06\_11\_2025.slx
- ▶ Figure 4.6: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/BF\_Vitesse/BF\_Courant\_Vitesse\_VALIDE\_06\_11\_2025.slx

- Gain proportionnel :  $K_p = 2,00$
- Gain intégral :  $K_i = 78,86$

Ces valeurs résultent d'un compromis entre rapidité et stabilité, garantissant une réponse dynamique tout en évitant les oscillations excessives qui pourraient endommager le système mécanique.

## 2.4. Validation sur PSIM

Après avoir conçu et validé le correcteur PI sur Simulink, nous procédons à l'implémentation du système complet sur PSIM. Cette étape est cruciale car PSIM offre une modélisation plus proche de la réalité physique, notamment en ce qui concerne les composants électroniques de puissance et leurs caractéristiques non-idiées. Comme pour l'asservissement en courant, il est nécessaire d'adapter les paramètres du correcteur PI de Simulink vers la notation utilisée par PSIM, en utilisant les relations de conversion présentées précédemment :

$$k = K_p \quad \text{et} \quad T = \frac{K_p}{K_i}$$

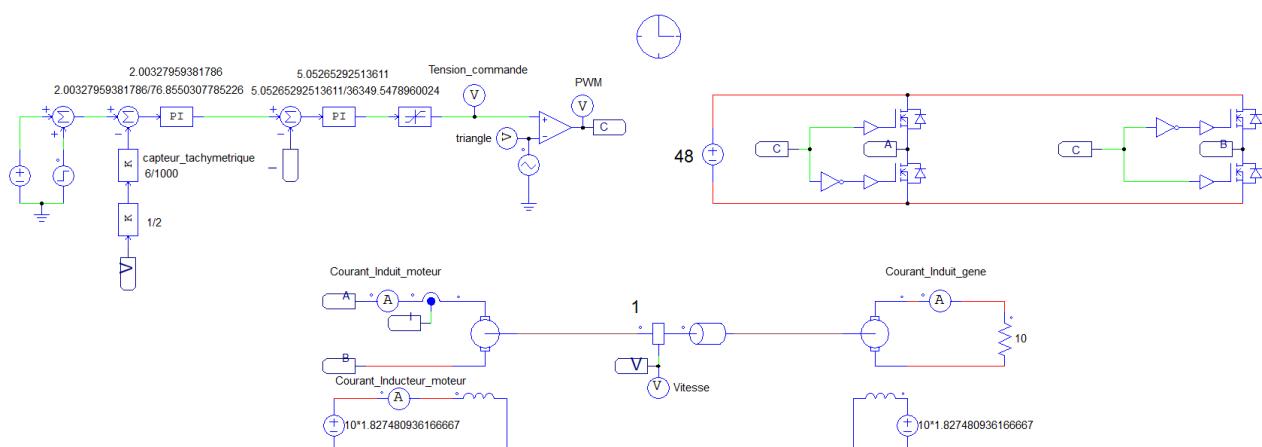


FIGURE 4.7 – Schéma de l'asservissement en vitesse du moteur dans PSIM

Le schéma PSIM intègre l'ensemble de la chaîne de régulation en cascade : l'asservissement en vitesse commande l'asservissement en courant, qui lui-même pilote le hacheur quatre quadrants alimentant le moteur à courant continu. Cette modélisation complète permet de simuler le comportement réel du système dans des conditions proches de l'implémentation finale.

## 2.5. Comparaison et validation des performances

La validation de notre approche de modélisation nécessite une comparaison rigoureuse entre les résultats obtenus sur Simulink et PSIM. Cette confrontation permet non seulement de vérifier la cohérence de nos modèles, mais également de confirmer que les performances visées sont effectivement atteintes dans les deux environnements de simulation.

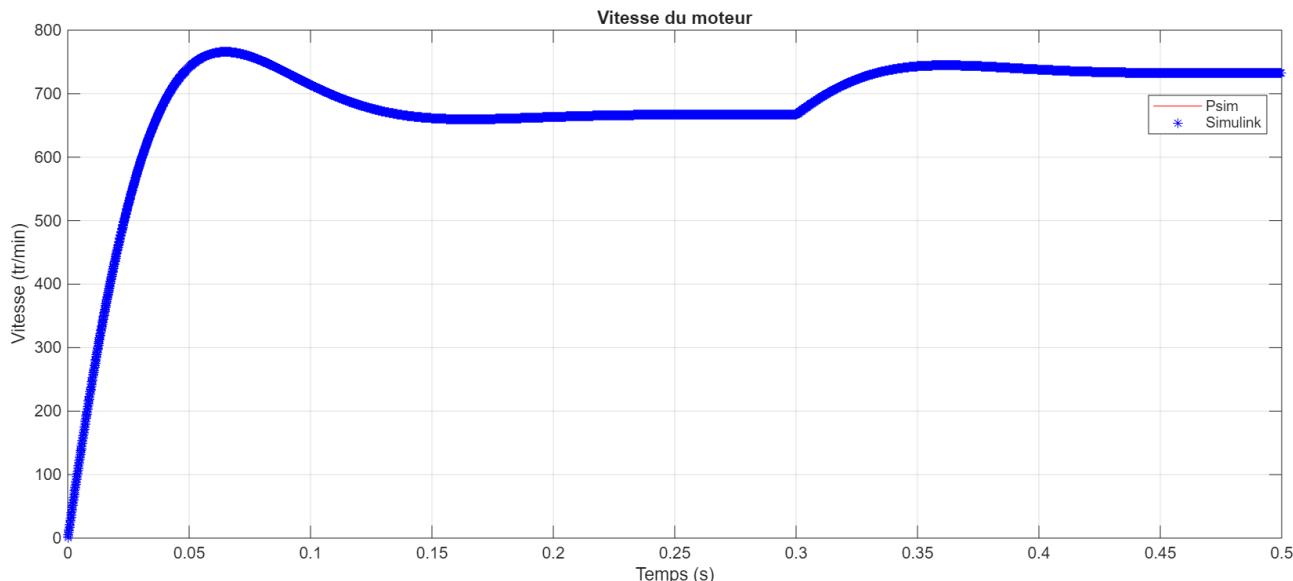


FIGURE 4.8 – Comparaison des réponses en vitesse entre Simulink et PSIM

La figure 4.8 illustre une superposition quasi-parfaite des réponses temporelles obtenues dans les deux environnements. Cette concordance remarquable valide la cohérence de nos modélisations et confirme que les paramètres du correcteur PI ont été correctement adaptés entre Simulink et PSIM. Pour faciliter la visualisation des deux courbes, un facteur de 0.99 a été appliqué aux valeurs de vitesse issues de Simulink.

L'analyse des performances révèle que l'asservissement en vitesse respecte pleinement les spécifications du cahier des charges :

- Le temps de réponse à 5 % est d'environ 106 ms, soit plus de trois fois plus rapide que la réponse en boucle ouverte (107.37 ms)
- Le dépassement observé reste inférieur à 20 % ( 17.8 %), ce qui limite les contraintes mécaniques sur le système d'entraînement
- Le régime permanent est atteint sans erreur statique grâce à l'action intégrale du correcteur PI

### 3. Asservissement en vitesse avec codeur incrémental

Le cahier des charges impose également la possibilité d'asservir la vitesse du moteur à l'aide d'un codeur incrémental. Contrairement à la dynamo tachymétrique qui fournit un signal analogique continu, le codeur incrémental est un capteur numérique qui génère des impulsions proportionnelles à la vitesse de rotation. Ce type de capteur offre une précision élevée et une meilleure immunité au bruit électromagnétique que les capteurs analogiques.

Dans cette section, nous présentons d'abord le principe de fonctionnement du codeur incrémental, puis nous validons son utilisation dans l'asservissement en vitesse à travers des simulations avec des correcteurs PI idéaux sur Simulink et PSIM.

#### 3.1. Principe de fonctionnement du codeur incrémental

Le fonctionnement du codeur incrémental repose sur un disque rotatif équipé de fentes. Celui-ci est interposé entre une diode électroluminescente et un capteur photodiode. Lorsque le disque tourne, les fentes permettent à la lumière de passer à travers, générant ainsi des impulsions électriques dans le capteur. Le nombre d'impulsions générées par tour dépend de la résolution du codeur.

Le codeur incrémental fourni est composé de deux pistes de fentes décalées d'un quart de période, ce qui permet de déterminer le sens de rotation du moteur en analysant la séquence des impulsions générées par les deux capteurs.

### 3.2. Calcul de la vitesse et du sens de rotation

Pour mesurer la vitesse de rotation du moteur, il suffit de compter le nombre d'impulsions générées par le codeur sur une période de temps donnée. La vitesse angulaire peut être calculée en utilisant la formule suivante :

$$\Omega = \frac{N \cdot 2\pi}{N_T \cdot T}$$

où :

- $\Omega$  est la vitesse angulaire en radians par seconde (rad/s)
- $N$  est le nombre d'impulsions comptées
- $N_T$  est le nombre d'impulsions par tour du codeur
- $T$  est la période de temps pendant laquelle les impulsions sont comptées en secondes (s)
- $2\pi$  est une constante pour convertir les tours en radians

Pour connaître le sens de rotation, on analyse la séquence des impulsions des deux pistes. Si la piste A précède la piste B, le moteur tourne dans un sens. Si la piste B précède la piste A, le moteur tourne dans le sens inverse.

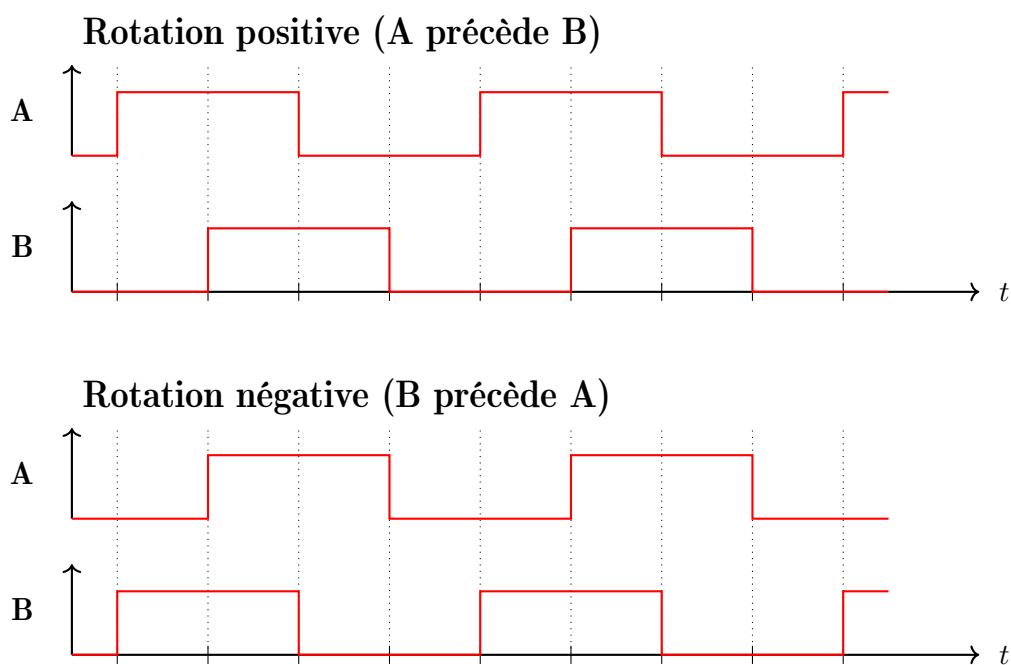


FIGURE 4.9 – Chronogrammes selon le sens de rotation avec le codeur incrémental

### 3.3. Simulation avec correcteurs PI idéaux sur PSIM

A présent, nous implémentons l'asservissement en vitesse avec le codeur incrémental en utilisant des correcteurs PI idéaux (blocs PI de PSIM). Cette étape permet de valider le principe de fonctionnement du système avant de passer à l'implémentation avec des composants réels. Nous obtenons le schéma suivant pour la partie codeur incrémental :

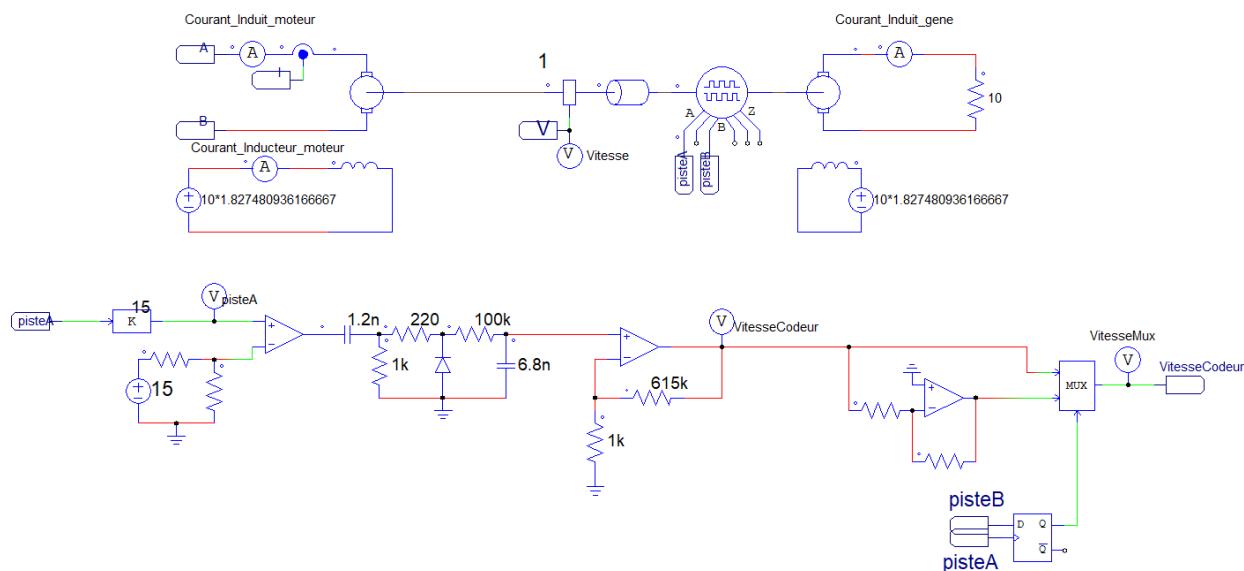


FIGURE 4.10 – Schéma de la partie codeur incrémental dans PSIM

Le schéma complet de l'asservissement en vitesse avec codeur incrémental devient ainsi :

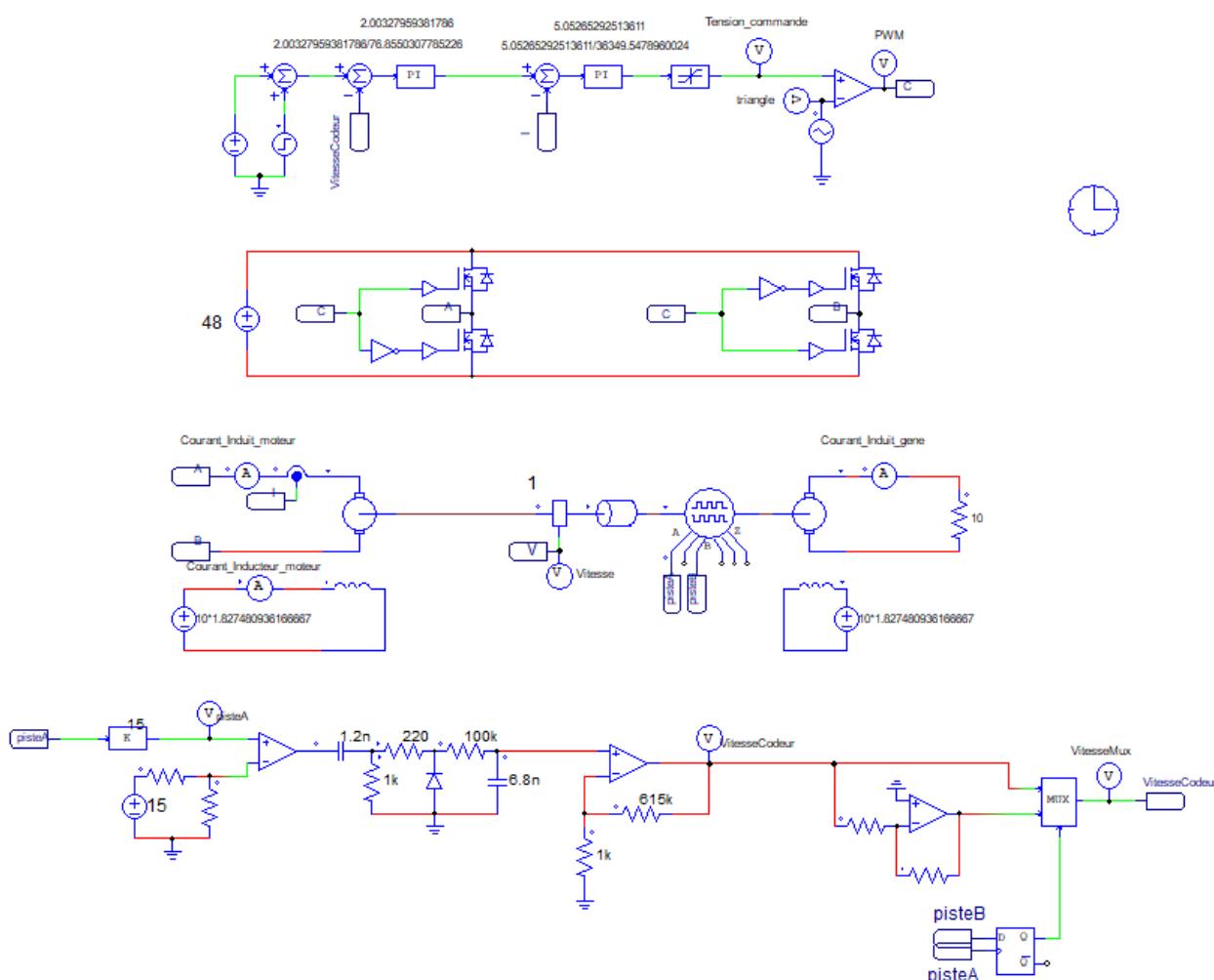


FIGURE 4.11 – Schéma de l'asservissement en vitesse avec codeur incrémental

### 3.4. Comparaison des résultats avec codeur incrémental

Nous comparons ici les résultats obtenus avec Simulink sans le codeur incrémental et PSIM avec l'asservissement en vitesse utilisant le codeur incrémental. Cette comparaison permet de valider la cohérence de nos modélisations et de vérifier que les performances sont conformes au cahier des charges.

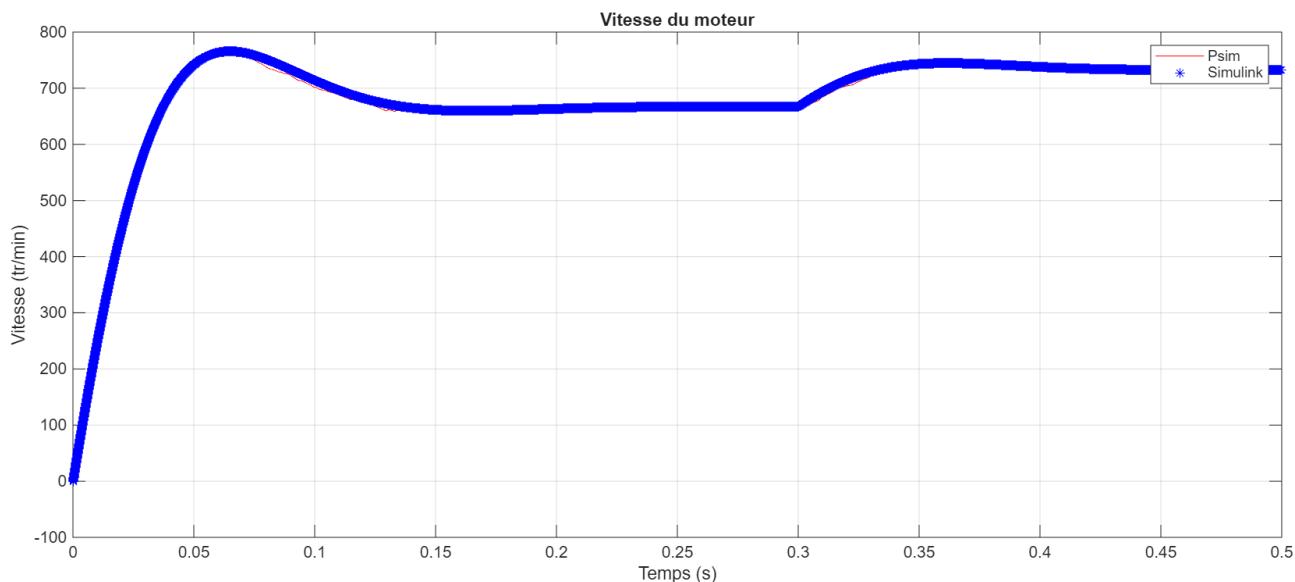


FIGURE 4.12 – Comparaison des résultats de simulation entre Simulink et PSIM avec codeur incrémental

La figure 4.12 présente une superposition des réponses en vitesse obtenues avec l'asservissement utilisant la dynamo tachymétrique sur Simulink et celui utilisant le codeur incrémental sur PSIM. Les deux courbes présentent une concordance satisfaisante, validant ainsi la cohérence entre les deux approches de mesure de vitesse.

On observe toutefois une différence notable dans l'allure des signaux : la réponse avec la dynamo tachymétrique présente une courbe lisse, tandis que celle avec le codeur incrémental affiche des oscillations caractéristiques. Cette différence s'explique par la nature fondamentalement différente des deux capteurs :

- **Dynamo tachymétrique** : capteur analogique générant une tension continue proportionnelle à la vitesse instantanée, produisant un signal de rétroaction lisse.
- **Codeur incrémental** : capteur numérique générant des impulsions discrètes. La vitesse est calculée en comptant les impulsions sur une fenêtre temporelle, introduisant une quantification qui crée un effet d'escalier entre deux impulsions successives.

Ces oscillations sont intrinsèques à la mesure numérique et leur amplitude dépend de la résolution du codeur (1800 impulsions/tour, soit  $0.36^\circ$ ) et de la période d'échantillonnage.

Malgré ces oscillations, les performances dynamiques moyennes restent conformes au cahier des charges : temps de réponse à 5 % d'environ 106 ms, dépassement inférieur à 20 %, et absence d'erreur statique. Le codeur incrémental constitue donc une alternative viable à la dynamo tachymétrique, offrant une meilleure immunité au bruit et une précision accrue. Les oscillations peuvent être atténuées par filtrage numérique si nécessaire.

## 5 - Dimensionnement des asservissements

Après avoir validé la structure de contrôle avec des régulateurs PI idéaux en simulation, il est nécessaire de passer à l'étape de réalisation pratique. Cette phase consiste à traduire les fonctions de transfert théoriques en circuits électroniques réels à base d'amplificateurs opérationnels (AOP).

Le dimensionnement des asservissements constitue une étape cruciale du projet, car elle établit le lien entre la théorie et la mise en œuvre physique. Chaque bloc de la chaîne de commande doit être soigneusement dimensionné afin de garantir que les performances obtenues en simulation soient effectivement reproduites sur le système réel. Dans ce chapitre, nous détaillons le dimensionnement de chacun des éléments constitutifs de la boucle d'asservissement, en commençant par les circuits soustracteurs, puis en abordant les correcteurs PI pour les boucles de courant et de vitesse.

### 1. Réalisation des soustracteurs

Pour commencer nous réalisons un montage soustracteur permettant de comparer la sortie du montage à la commande.

On utilise alors ce montage à AOP qui permet, en choisissant les mêmes valeurs de résistances, d'obtenir un soustracteur simple :

$$V_s = V_2 - V_1$$

On choisit des résistances de  $10k\Omega$  pour la réalisation pratique.

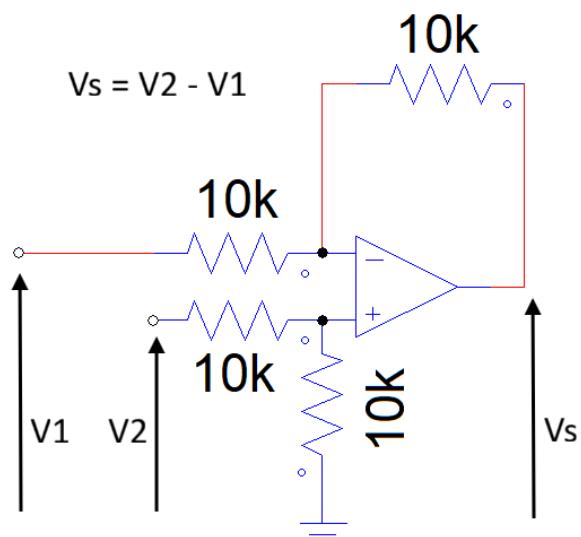


FIGURE 5.1 – Montage soustracteur à AOP

### 2. Dimensionnement des correcteurs réels

#### 2.1. Structure du correcteur PI

La structure du correcteur PI à implémenter est représentée à la figure 5.2.

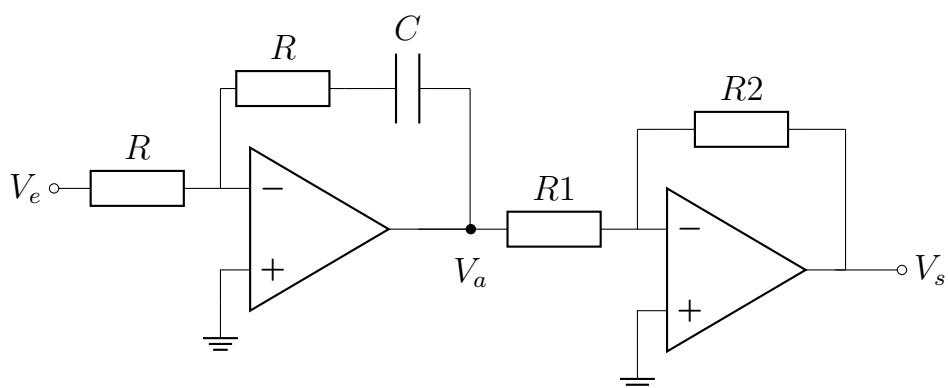


FIGURE 5.2 – Schéma d'un correcteur PI avec des amplificateurs opérationnels

## 2.2. Calcul de la fonction de transfert globale

La fonction de transfert globale du système s'exprime comme suit :

$$H(s) = \frac{V_s(s)}{V_e(s)} = \frac{V_a(s)}{V_e(s)} \cdot \frac{V_s(s)}{V_a(s)} = H_1(s) \cdot H_2(s)$$

Dans cette configuration, on suppose que les amplificateurs opérationnels sont idéaux, ce qui implique  $\varepsilon = 0 \implies V^+ = V^-$ . Le dimensionnement des correcteurs s'effectue en séparant les deux fonctions de transfert.

### 2.3. Dimensionnement du correcteur $H_1(s)$

Pour le premier étage du correcteur, on introduit une impédance équivalente :

$$Z_{eq}(s) = R + \frac{1}{C \cdot s} = \frac{RC \cdot s + 1}{sC}$$

En appliquant le théorème de Millman à l'entrée négative du premier AOP, on obtient :

$$V^- = \frac{\frac{V_e}{R} + \frac{V_a}{Z_{eq}(s)}}{\frac{1}{R} + \frac{1}{Z_{eq}(s)}}$$

Puisque  $V^- = 0V$ , le numérateur doit être nul. On a donc :

$$\frac{V_e}{R} + \frac{V_a}{Z_{eq}(s)} = 0$$

En réarrangeant, on obtient :

$$\begin{aligned} \frac{V_a}{Z_{eq}(s)} &= -\frac{V_e}{R} \\ V_a &= -\frac{Z_{eq}(s)}{R} V_e \end{aligned}$$

Et donc :

$$H_1(s) = \frac{V_a(s)}{V_e(s)} = -\frac{Z_{eq}(s)}{R} = -\frac{\frac{RC \cdot s + 1}{sC}}{R} = -\frac{RC \cdot s + 1}{RC \cdot s}$$

### 2.4. Dimensionnement du correcteur $H_2(s)$

On applique le théorème de Millman au nœud  $V^-$  du second AOP :

$$V^- = \frac{\frac{V_a(s)}{R_1} + \frac{V_s(s)}{R_2}}{\frac{1}{R_1} + \frac{1}{R_2}}$$

Puisque  $V^- = 0V$  (masse virtuelle), le numérateur doit être nul :

$$\frac{V_a(s)}{R_1} + \frac{V_s(s)}{R_2} = 0$$

On isole  $V_s(s)$  pour trouver la fonction de transfert :

$$\frac{V_s(s)}{R_2} = -\frac{V_a(s)}{R_1} \implies V_s(s) = -V_a(s) \cdot \frac{R_2}{R_1}$$

La fonction de transfert du second étage s'exprime alors par :

$$H_2(s) = \frac{V_s(s)}{V_a(s)} = -\frac{R_2}{R_1}$$



## 2.5. Fonction de transfert globale et identification

On calcule la fonction de transfert globale  $H(s)$  en multipliant  $H_1(s)$  et  $H_2(s)$  :

$$H(s) = H_1(s) \cdot H_2(s) = \left( -\frac{RC \cdot s + 1}{RC \cdot s} \right) \cdot \left( -\frac{R_2}{R_1} \right)$$

Les deux signes négatifs s'annulent. On réarrange les termes :

$$H(s) = \frac{R_2}{R_1} \cdot \left( \frac{RC \cdot s + 1}{RC \cdot s} \right)$$

On sépare la fraction pour faire apparaître la forme canonique :

$$H(s) = \frac{R_2}{R_1} \cdot \left( \frac{RC \cdot s}{RC \cdot s} + \frac{1}{RC \cdot s} \right) = \frac{R_2}{R_1} \left( 1 + \frac{1}{RC \cdot s} \right)$$

La forme de Laplace standard (ou canonique) pour un correcteur PI est :

$$H_{PI}(s) = K_p \left( 1 + \frac{1}{T_i s} \right)$$

## 2.6. Identification des paramètres

Par identification directe avec la structure PI souhaitée, on obtient les relations suivantes :

$$\begin{aligned} K_p &= \frac{R_2}{R_1} \\ T_i &= RC \end{aligned}$$

## 2.7. Choix des composants

À partir des paramètres théoriques des correcteurs PI déterminés précédemment, nous procérons à l'identification des composants passifs normalisés (résistances et condensateurs) dont les valeurs commerciales permettent d'approcher au mieux les caractéristiques désirées. Les résultats de cette identification sont présentés ci-après :

### Correcteur PI pour l'asservissement en courant :

Les paramètres du correcteur sont :  $K_p = 5,053$  et  $T_i = K_p/K_i = 1,39 \times 10^{-4}$  s.

Par identification avec les relations  $K_p = R_2/R_1$  et  $T_i = RC$ , nous obtenons :

$$R \cdot C = 10 \times 10 \times 10^{-6} = 1 \times 10^{-4} \approx T_i \quad \text{et} \quad \frac{R_2}{R_1} = \frac{8,2 \times 10^3}{1,5 \times 10^3} = 5,467 \approx K_p$$

Composant	Valeur
$R$	$10 \Omega$
$C$	$10 \mu\text{F}$
$R_1$	$1,5 \text{k}\Omega$
$R_2$	$8,2 \text{k}\Omega$



### Correcteur PI pour l'asservissement en vitesse :

Les paramètres du correcteur sont :  $K_p = 6,8$  et  $T_i = K_p/K_i = 13,45$  ms.

Par identification avec les relations  $K_p = R_2/R_1$  et  $T_i = RC$ , nous obtenons :

$$R \cdot C = 3,3 \times 10^3 \times 3,9 \times 10^{-6} = 0,0129 \approx T_i \quad \text{et} \quad \frac{R_2}{R_1} = \frac{6,8 \times 10^3}{1 \times 10^3} = 6,8 \approx K_p$$

Composant	Valeur
$R$	$3,3 \text{ k}\Omega$
$C$	$3,9 \mu\text{F}$
$R_1$	$1 \text{ k}\Omega$
$R_2$	$6,8 \text{ k}\Omega$

## 3. Dimensionnement des autres composants de la chaîne d'asservissement

Après avoir dimensionné les correcteurs PI et les soustracteurs, il est nécessaire de compléter la chaîne d'asservissement avec les autres éléments essentiels : le limiteur de courant pour protéger le système, et les capteurs de vitesse (dynamo tachymétrique et codeur incrémental) qui nécessitent des circuits de conditionnement pour adapter leurs signaux.

### 3.1. Limiteur de courant

Afin de garantir le respect des contraintes fixées par le cahier des charges, il est impératif de limiter le courant circulant dans le système à une valeur maximale de  $5 \text{ A}$ . Cette protection est assurée par un circuit écrêteur de tension, réalisé à l'aide de résistances et diodes.

#### Principe de fonctionnement

Le montage proposé repose sur une architecture de redressement simple alternance permettant d'effectuer un écrêtage asymétrique du signal.

Durant l'alternance positive du signal d'entrée, la diode D se trouve polarisée en direct et devient conductrice. Le courant traverse alors le diviseur résistif constitué des résistances en série ( $20 \text{ k}\Omega + 10 \text{ k}\Omega$ ), générant une chute de tension aux bornes de cet ensemble. Cette tension présente une forme identique à celle de l'alternance positive du signal d'entrée.

Lors de l'alternance négative, la diode D se trouve polarisée en inverse, ce qui se traduit par une impédance interne très élevée. Dans cette configuration, le courant traversant le diviseur résistif devient quasi nul, entraînant une chute de tension négligeable à ses bornes. Le signal de sortie est ainsi écrété pour les valeurs négatives.

L'amplificateur opérationnel monté en configuration suiveur, placé en sortie du circuit, assure une fonction d'adaptation d'impédance. Il permet de découpler les étages tout en préservant l'intégrité du signal écrêté sans charge supplémentaire sur le montage à diode.

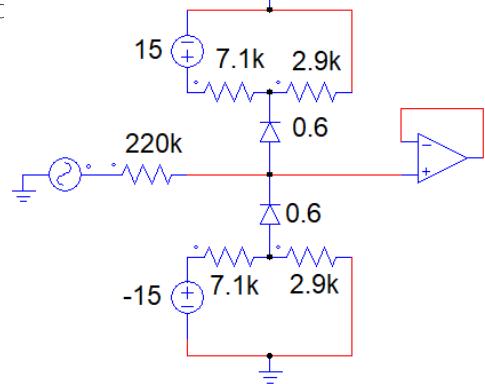


FIGURE 5.3 – Montage limiteur de tension à diodes

### 3.2. Capteur dynamo tachymétrique

La dynamo tachymétrique est un capteur analogique qui convertit directement la vitesse de rotation en une tension proportionnelle. Ce composant nécessite un conditionnement de signal pour adapter sa sortie aux niveaux requis par notre système d'asservissement.

#### Caractéristiques du capteur

Le capteur dynamo tachymétrique utilisé possède une constante de conversion  $K_{\text{tachy}} = 6 \times 10^{-3}$  V/(rad/s). Cette valeur indique la tension générée par le capteur pour une vitesse de rotation donnée.

#### Conditionnement du signal

Sur PSIM, nous modélisons le capteur en appliquant le facteur 6/1000 à la vitesse mesurée pour représenter la dynamo tachymétrique réelle. Pour adapter la tension de sortie aux niveaux requis par le système, nous ajoutons un pont diviseur de tension qui divise par deux la tension de sortie.

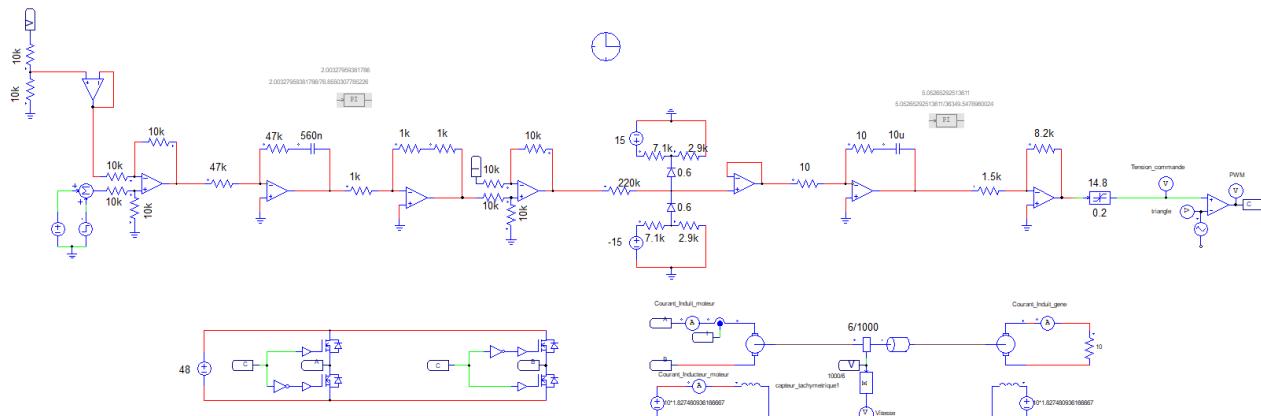


FIGURE 5.4 – Schéma de modélisation du capteur tachymétrique sur PSIM

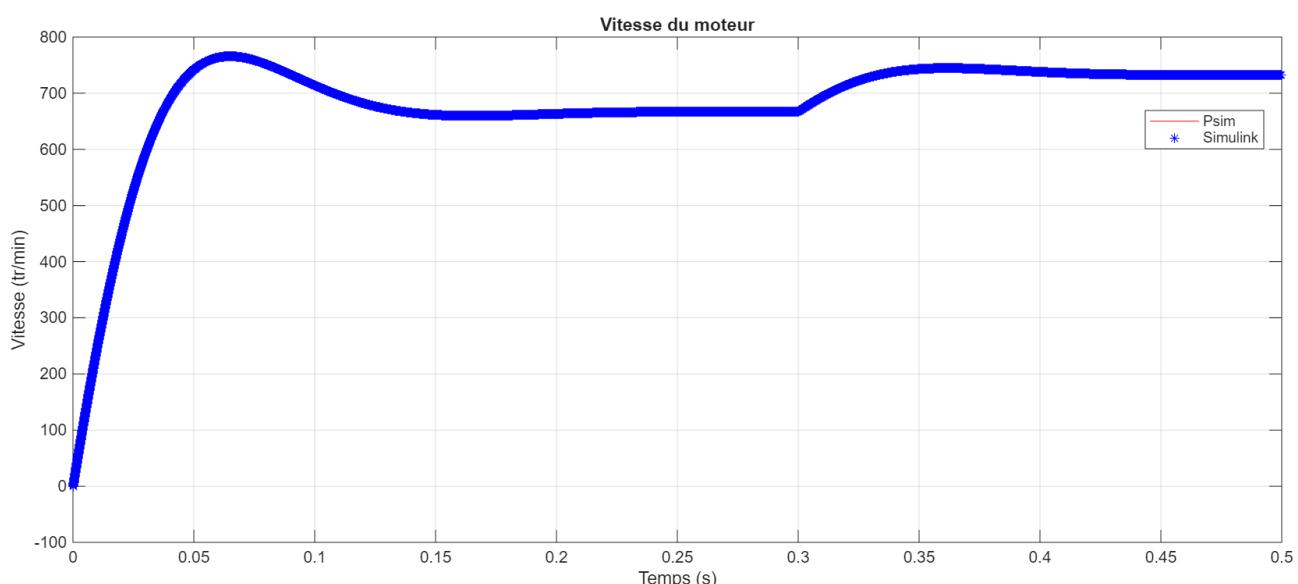


FIGURE 5.5 – Comparaison entre Simulink et PSIM pour le capteur tachymétrique et les choix de composants

La figure 5.5 illustre la comparaison entre les modélisations Simulink avec correcteurs PI idéaux et PSIM avec composants réels (résistances et condensateurs). On observe une excellente concordance entre les deux courbes, sachant que nous avons à nouveau appliqué un facteur 0,99 aux données Simulink. Le graphe confirme que les choix de composants permettent de reproduire les performances théoriques. Les performances dynamiques restent conformes aux spécifications, démontrant que le passage aux composants réels n'altère pas significativement le comportement de l'asservissement.

### 3.3. Codeur incrémental

Le codeur simulé sous PSIM délivre une tension comprise entre 0 et 1 V, alors que le codeur réel fournit un signal de 0 à 15 V. Pour corriger cette différence d'échelle et obtenir une simulation représentative, un gain de 15 a été ajouté à l'entrée du bloc de calcul de la vitesse de rotation.

#### Circuit de conversion impulsions/tension

Pour utiliser le codeur incrémental dans notre système d'asservissement analogique, il est nécessaire de convertir les impulsions numériques en une tension analogique proportionnelle à la vitesse. Cette conversion peut être réalisée par un circuit fréquence-tension (convertisseur F/V). Le schéma complet de l'asservissement en vitesse avec codeur incrémental devient ainsi :

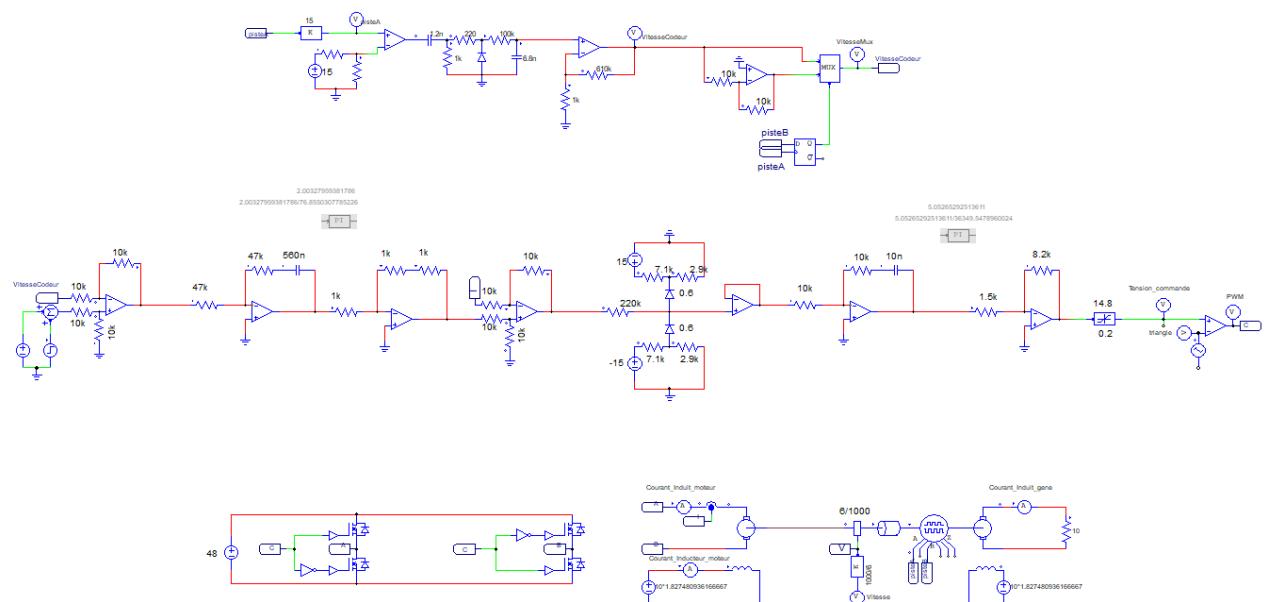


FIGURE 5.6 – Schéma complet de l'asservissement en vitesse avec codeur incrémental

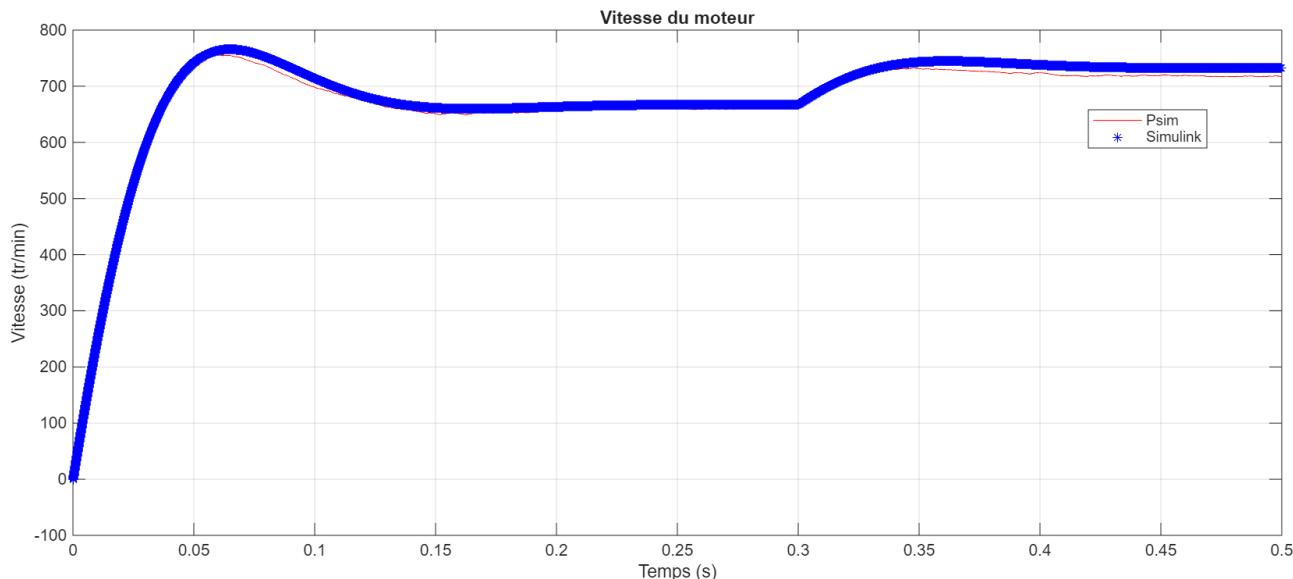


FIGURE 5.7 – Comparaison entre Simulink et PSIM pour le codeur incrémental et les choix de composants

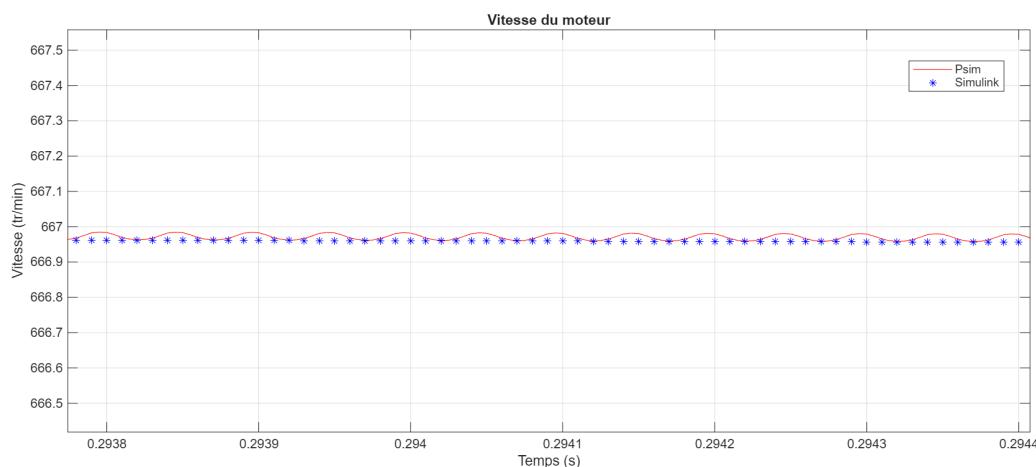


FIGURE 5.8 – Zoom sur la comparaison entre Simulink et PSIM pour le capteur tachymétrique

La figure 5.7 illustre la comparaison entre les modélisations Simulink avec correcteurs PI idéaux et PSIM avec composants réels (résistances et condensateurs). On observe une bonne concordance entre les deux courbes, confirmant que les choix de composants permettent de reproduire les performances théoriques. Les oscillations caractéristiques du codeur incrémental sont présentes sur la simulation PSIM, validant la cohérence de la modélisation du capteur numérique. Les performances dynamiques restent conformes aux spécifications, démontrant que le passage aux composants réels n'altère pas significativement le comportement de l'asservissement.

## 6 - Conception des PCB

Afin de réaliser le circuit imprimé de notre projet, nous avons utilisé le logiciel de conception de PCB *Proteus*, dans lequel nous avons reproduit le schéma électrique complet vu précédemment. Cependant, pour faciliter la réalisation du PCB, nous avons dû apporter quelques modifications au schéma initial pour bien préparer la carte à la fabrication et à l'assemblage. Pour cela, nous avons créé les différents empreintes des composants utilisés, en veillant à respecter les dimensions réelles de chaque composant, données par les fiches techniques des fabricants. Nous avons également pensé à définir les zones de cuivre pour l'alimentation et la masse (plan de masse), afin de faciliter le routage et d'améliorer les performances électriques de la carte. Enfin, nous avons ajouté des perçages pour les fixations mécaniques de la carte dans son boîtier.

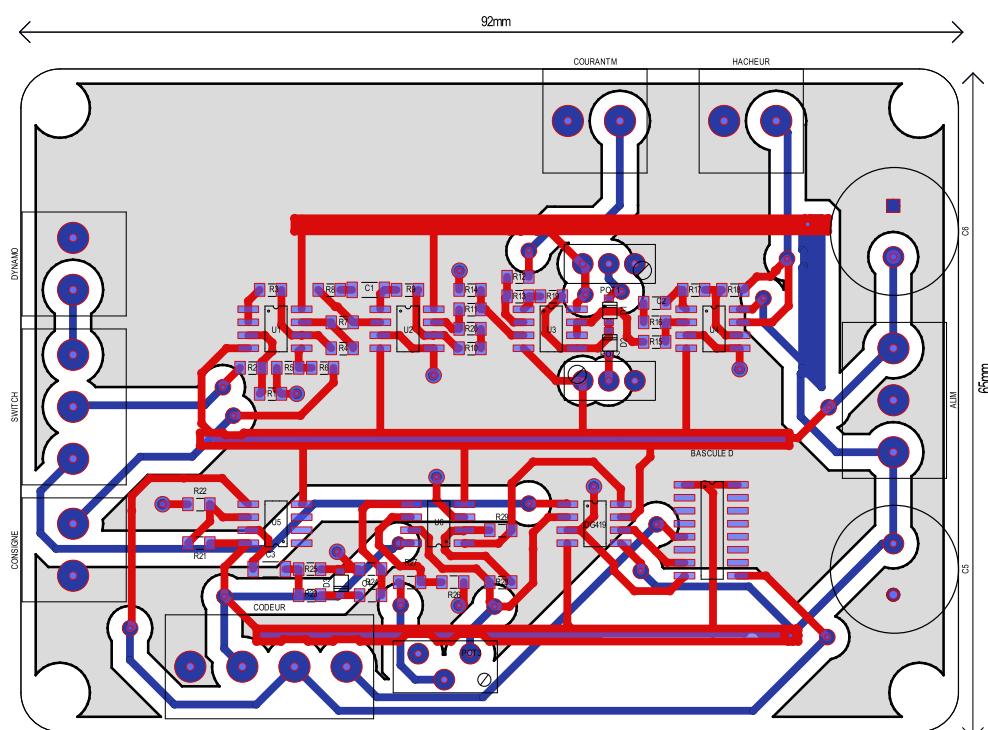
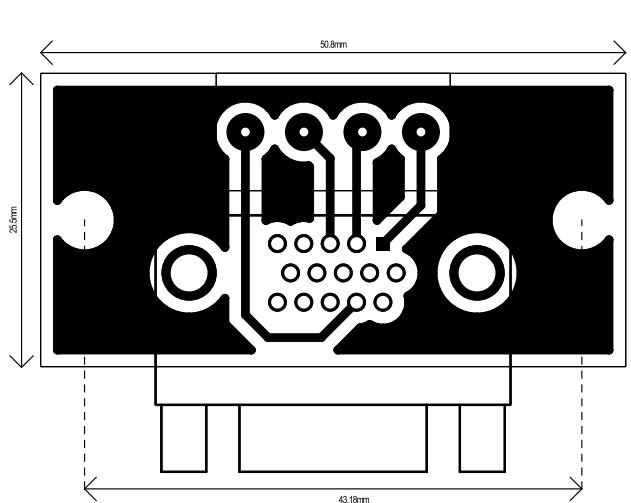


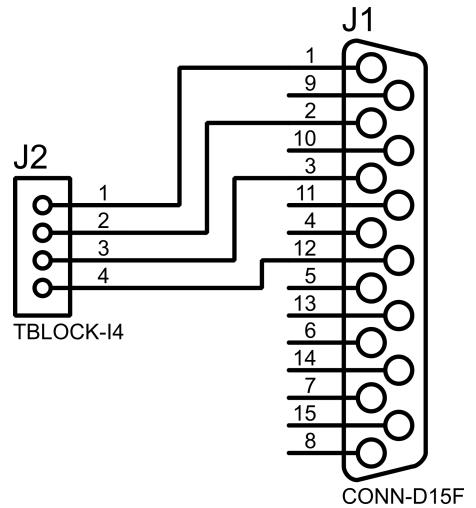
FIGURE 6.1 – Schéma de la conception du PCB

Concernant le placement des composants sur le PCB, nous avons répartis les composants sur deux "étages". Le premier étage (celui du haut sur l'image 6.1) comporte toute la chaîne d'asservissement du moteur en courant et en vitesse. Tandis que le second étage (celui du bas sur l'image 6.1) contient les composants liés au traitement du signal de sortie du codeur incrémental.

En ce qui concerne le connecteur VGA, nous avons également conçu son PCB et son schéma électrique dans *Proteus*, en respectant le standard de câblage VGA prescrit sur Moodle pour assurer une bonne compatibilité avec l'équipement que nous utilisons.



(a) PCB du connecteur VGA



(b) Schéma du connecteur VGA

FIGURE 6.2 – Conception du connecteur VGA

# 7 - Assemblage du système

## 1. Commande des composants

En parallèle de la réalisation du circuit imprimé, nous avons réalisé la commande de composants chez le fournisseur Farnell.

Le budget étant restreint à 25€, nous avons fait en sorte de récupérer un maximum de composants déjà présents en PFGE pour limiter le coût de notre commande. De plus, nous avons mutualisé les commandes des différents groupes pour éviter les frais de livraison et diminuer le coût de certains composants achetés en plus gros lot.

Voici la liste des composants que nous avons dû commander :

Composant	Référence	Lien Farnell	Quantité	Prix HT
TL082	TL082CDT	Lien	6	3,52 €
Diodes	1N4148WS	Lien	3	0,50 €
Potentiomètre 10kΩ	T93YA103KT20	Lien	2	1,90 €
Bascule D	HEF40175BT,653	Lien	1	1,06 €
MUX	DG419DY-E3	Lien	1	1,57 €
Capacité 470 $\mu$ F	25ZLK470M8X20	Lien	2	0,83 €
Résistance 47kΩ	CRCW080547K0FKEA	Lien	1	0,22 €
Résistance 8,2kΩ	MCWF08P8201FTL	Lien	1	0,11 €
Résistance 560kΩ	ESR10EZPF5603	Lien	1	0,75 €
Capacité 560nF	MC1206B564K250CT	Lien	1	0,79 €
Capacité 6.8nF	MC1206B682K201CT	Lien	1	0,41 €
Capacité 1.2nF	CC1206JKNPOZBN122	Lien	1	0,63 €
<b>TOTAL TTC</b>				<b>14,8 €</b>

Tableau 1 : Liste des composants commandés

## 2. Soudure du PCB

Une fois que le PCB a été produit par les techniciens et que nous avons reçu la commande, nous sommes passés à la soudure de la carte.

Pour ce faire, nous avions décidé de réaliser une carte entièrement avec des composants montés en surface pour obtenir la carte la plus petite possible. Nous avons donc réalisé en premier la soudure des composants CMS au four à souder, nous avons donc placé les AOP, résistances, capacités et diodes ainsi que la bascule D et l'interrupteur commandé.

Une fois ces composants soudés, nous nous sommes occupés des composants traversant : les différents borniers, potentiomètres et capacités de découplage. Nous avons soudé ces éléments au fer à souder.

Cela nous donne le PCB visible à côté.

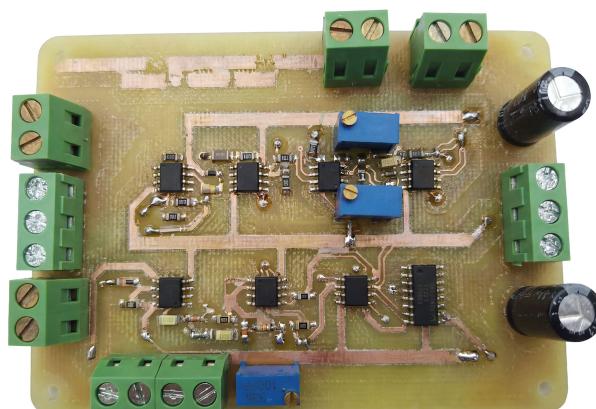


FIGURE 7.1 – PCB après soudure des composants

Pour ce qui est de la connexion de notre carte au système réel, nous devons utiliser des câbles coaxiaux pour récupérer les différentes tensions utiles au fonctionnement du système. De plus, notre système doit être alimenté en +15V et -15V à l'aide d'une alimentation de labo.

Enfin, il doit être possible de choisir avec quel capteur nous réalisons l'asservissement, donc nous utilisons un interrupteur pour basculer entre la dynamo tachymétrique et le codeur incrémental qui est relié à un port VGA.

Ce qui fait que nous devons réaliser la soudure de fils reliant la carte à ces composants de connectique :

- 4 fiches BNC
- 3 fiches bananes
- 1 interrupteur
- 1 connecteur VGA

### 3. Réalisation du boitier

Pour maintenir le PCB et les connectiques en place, nous avons réalisé une boîte imprimée en 3D pour y fixer tous les composants.

Cette boîte a été pensée pour être la plus pratique possible si des modifications doivent être réalisées. C'est pourquoi nous avons décidé de fixer la totalité des composants sous le couvercle de cette boîte pour que, si nous voulons l'ouvrir, nous ayons facilement accès à tous les composants. Le but était vraiment d'éviter que des fils relient le couvercle et la boîte, ce qui est embêtant quand on veut l'ouvrir.

Nous avons aussi fait en sorte d'indiquer sur le couvercle l'utilité de chaque connectique en imprimant le texte d'une autre couleur.



FIGURE 7.2 – Vue 3D du boitier

Nous avons mis en place plusieurs éléments visant à rendre notre boîte plus fonctionnelle :

- Des empreintes à double meplat pour fixer les fiche BNC, évitant ainsi qu'elles ne tournent lors des connexions/déconnexions. Cela permet également d'alligner parfaitement les fiches BNC.
- Un epaulement sur le pourtour du couvercle pour qu'il s'emboîte parfaitement dans la boîte même sans être vissé et éviter ainsi toute entrée de poussière.
- Des pieds en caoutchouc pour éviter que la boîte ne glisse sur le poste de travail

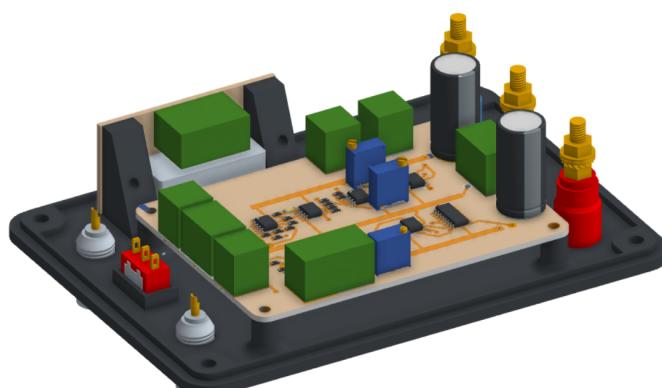


FIGURE 7.3 – Vue 3D du boitier ouvert

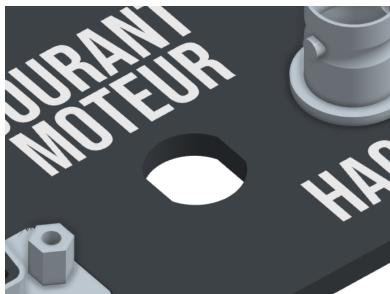


FIGURE 7.4 – Empreintes à double meplat



FIGURE 7.5 – Épaulement du couvercle

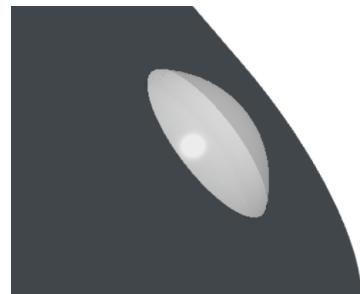


FIGURE 7.6 – Pieds en caoutchouc

Cette modélisation 3D complète de notre système nous a permis de vérifier que tous les composants rentraient correctement dans la boîte avant de lancer l'impression 3D.

Voici le résultat du projet entièrement assemblé :



FIGURE 7.7 – Couvercle du boitier



FIGURE 7.8 – Vue d'ensemble du boitier

Pour réaliser les connexions entre le PCB et les différentes connectiques, nous avons utilisé des câbles que nous avons soudé aux fiches BNC, puis vissé aux borniers du PCB. Les fiches bananes ont été connecté au PCB via des câbles avec des cosses pour effectuer les connexions.

Pour faire cela proprement, nous avons coupé les fils pour qu'ils soient le plus courts possible pour éviter leur encombrement. Nous avons également rajouté des gaines thermo rétractables à chaque soudure pour éviter tout faux contact ou court-circuit.

Voici le PCB relié à tous les organes de connexions :

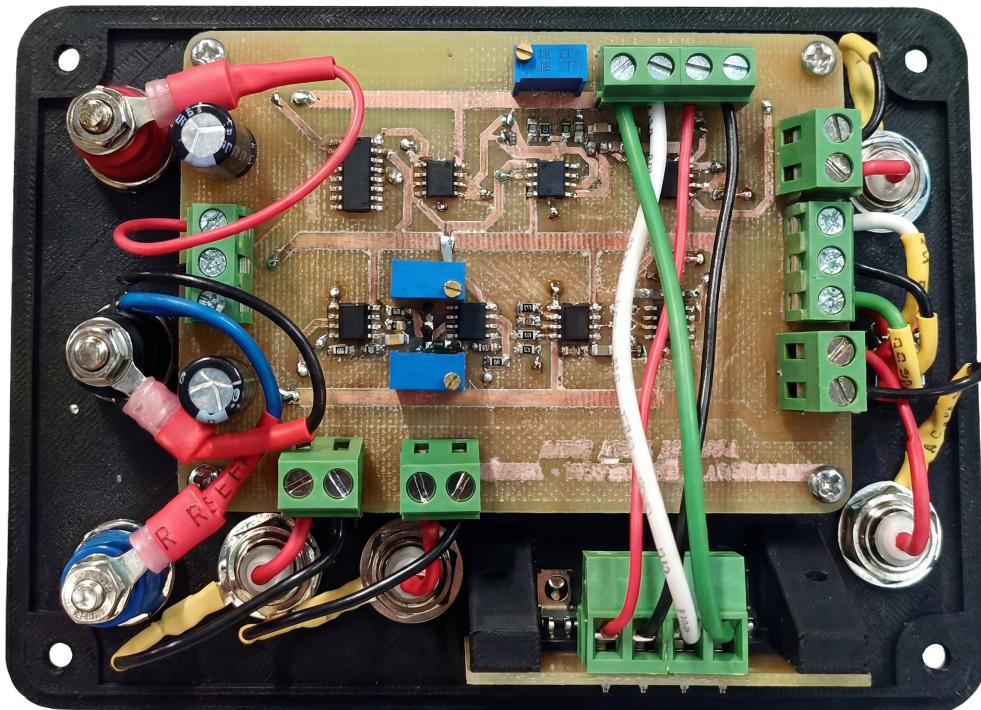


FIGURE 7.9 – PCB relié à toutes les connectiques

# Conclusion

Ce projet d'asservissement d'une machine à courant continu a permis d'acquérir une vision complète de la chaîne de régulation, depuis la modélisation théorique jusqu'au dimensionnement des composants électroniques réels. À ce stade du projet, il convient de dresser un bilan des travaux réalisés et d'identifier les tâches qui restent à accomplir pour finaliser le système.

## Travaux restant à réaliser

Malgré les avancées significatives effectuées en simulation et en dimensionnement, plusieurs étapes essentielles restent à compléter avant la finalisation du projet :

1. **Réalisation du circuit imprimé (PCB)** : La conception et la fabrication de la carte électronique permettant d'implémenter physiquement l'ensemble des circuits dimensionnés (correcteurs PI, soustracteurs, limiteur de tension, conditionnement des capteurs).
2. **Assemblage et câblage du système** : Le montage des composants sur le PCB, le raccordement du moteur, des capteurs (dynamo tachymétrique et codeur incrémental) et de l'alimentation de puissance.
3. **Tests et validation expérimentale** : La mise en service progressive du système réel, comprenant :
  - La vérification du bon fonctionnement des correcteurs PI analogiques
  - Les tests de l'asservissement en courant puis en vitesse
  - La comparaison des performances mesurées avec les résultats de simulation
4. **Réglage et optimisation** : L'ajustement fin des paramètres si nécessaire pour compenser les écarts entre le modèle théorique et la réalité (tolérances des composants, pertes non modélisées, perturbations électromagnétiques).
5. **Caractérisation complète** : L'évaluation des performances dans différentes conditions de fonctionnement (variations de charge, changements de consigne, robustesse face aux perturbations).

## Bilan et perspectives

Les travaux menés ont établi des fondations solides pour la réalisation pratique du système. La modélisation mathématique de la MCC a été développée et validée sur MATLAB/Simulink et PSIM, incluant la machine à vide, avec charge mécanique et hacheur quatre quadrants. Deux boucles de régulation en cascade ont été conçues et optimisées, respectant les spécifications du cahier des charges en termes de temps de réponse et de dépassement. L'étude comparative de deux capteurs (dynamo tachymétrique et codeur incrémental) a validé la flexibilité du système, et le dimensionnement complet des circuits analogiques réels a été réalisé avec succès.

Ce projet a permis de développer une méthodologie rigoureuse allant de la modélisation théorique à la réalisation pratique, tout en maîtrisant les outils de simulation et en validant systématiquement chaque étape par comparaison croisée. La réalisation expérimentale du système constituera l'aboutissement de ces travaux, permettant de confronter les prédictions théoriques à la réalité. Les fondations méthodologiques acquises constitueront des atouts précieux pour les développements futurs et l'adaptation du système à d'autres applications industrielles.

# Liste des figures et chemins d'accès

Cette section recense l'ensemble des figures présentes dans ce rapport ainsi que le chemin d'accès aux fichiers de simulation correspondants.

Figure 3.1: Modélisation\_Finale\_VALIDE\_13\_01\_2026/00\_Moteur\_a\_Vide/Moteur\_Seul\_VALIDE\_12\_09\_2025.slx

Figure 3.2: Modélisation\_Finale\_VALIDE\_13\_01\_2026/00\_Moteur\_a\_Vide/Moteur\_Seul\_VALIDE\_12\_09\_2025.slx

Figure 3.3: Modélisation\_Finale\_VALIDE\_13\_01\_2026/00\_Moteur\_a\_Vide/Moteur\_Seul\_VALIDE\_12\_09\_2025.psimsch

Figure 3.4: Modélisation\_Finale\_VALIDE\_13\_01\_2026/00\_Moteur\_a\_Vide/Comparaison\_Moteur\_Seul\_VALIDE\_12\_09\_2025.m

Figure 3.5: Modélisation\_Finale\_VALIDE\_13\_01\_2026/01\_Moteur\_Charge\_Frottement/Moteur\_Charge\_Frottement\_VALIDE\_03\_11\_2025.slx

Figure 3.6: Modélisation\_Finale\_VALIDE\_13\_01\_2026/01\_Moteur\_Charge\_Frottement/Moteur\_Charge\_Frottement\_VALIDE\_03\_11\_2025.psimsch

Figure 3.7: Modélisation\_Finale\_VALIDE\_13\_01\_2026/01\_Moteur\_Charge\_Frottement/Comparaison\_Courant\_Moteur\_Charge\_Frottement\_VALIDE\_03\_11\_2025.m

Figure 3.8: Modélisation\_Finale\_VALIDE\_13\_01\_2026/02\_Moteur\_Hacheur/Moteur\_Hacheur\_VALIDE\_03\_11\_2025.psimsch

Figure 3.10: Modélisation\_Finale\_VALIDE\_13\_01\_2026/02\_Moteur\_Hacheur/Moteur\_Hacheur\_VALIDE\_03\_11\_2025.slx

Figure 3.11: Modélisation\_Finale\_VALIDE\_13\_01\_2026/02\_Moteur\_Hacheur/Comparaison\_Courant\_Moteur\_Hacheur\_VALIDE\_03\_11\_2025.m

Figure 4.1: Modélisation\_Finale\_VALIDE\_13\_01\_2026/03\_BF\_Courant/BF\_Courant\_VALIDE\_06\_11\_2025.slx

Figure 4.2: Modélisation\_Finale\_VALIDE\_13\_01\_2026/03\_BF\_Courant/BF\_Courant\_VALIDE\_06\_11\_2025.psimsch

Figure 4.3: Modélisation\_Finale\_VALIDE\_13\_01\_2026/03\_BF\_Courant/Comparaison\_Asservissement\_Courant\_VALIDE\_06\_11\_2025.m

Figure 4.4: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/B0\_Vitesse/B0\_Vitesse\_VALIDE\_06\_11\_2025.slx

Figure 4.5: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/B0\_Vitesse/B0\_Vitesse\_VALIDE\_06\_11\_2025.slx

Figure 4.6: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/BF\_Vitesse/BF\_Courant\_Vitesse\_VALIDE\_06\_11\_2025.slx

Figure 4.7: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/BF\_Vitesse/BF\_Asservissement\_Courant\_Vitesse\_VALIDE\_06\_11\_2025.psimsch

Figure 4.8: Modélisation\_Finale\_VALIDE\_13\_01\_2026/04\_BF\_Courant\_Vitesse/Comparaison\_BF\_Vitesse/Comparaison\_Asservissement\_Courant\_Vitesse\_VALIDE\_06\_11\_2025.m

Figure 4.10: Modélisation\_Finale\_VALIDE\_13\_01\_2026/05\_Codeur\_Incrémentale/BF\_Vitesse/Codeur\_Incrementale\_VALIDE\_08\_11\_25.psimsch

Figure 4.11: Modélisation\_Finale\_VALIDE\_13\_01\_2026/05\_Codeur\_Incrémentale/BF\_Vitesse/Codeur\_Incrementale\_VALIDE\_08\_11\_25.psimsch

Figure 4.12: Modélisation\_Finale\_VALIDE\_13\_01\_2026/05\_Codeur\_Incrémentale/Comparaison\_Vitesse/Codeur\_Incremental\_VALIDE\_08\_11\_2025.m

Figure 5.4: Modélisation\_Finale\_VALIDE\_13\_01\_2026/06\_Choix\_Des\_Composants/Tachy/Choix\_composants\_Tachy\_VALIDE\_08\_11\_25.psimsch

Figure 5.5: Modélisation\_Finale\_VALIDE\_13\_01\_2026/06\_Choix\_Des\_Composants/Tachy/Comparaison\_Composants\_Tachy\_VALIDE\_08\_11\_2025.m

Figure 5.6: Modélisation\_Finale\_VALIDE\_13\_01\_2026/06\_Choix\_Des\_Composants/Codeur/Choix\_Composants\_Codeur\_Incremental\_VALIDE\_08\_11\_25.psimsch

Figure 5.7: Modélisation\_Finale\_VALIDE\_13\_01\_2026/06\_Choix\_Des\_Composants/Codeur/Comparaison\_Composants\_Codeur\_Incremental\_VALIDE\_08\_11\_25.m