

# RAPPORT BE AUTOMOBILE

## Partie 1 : Analyse de la commande U/f

BOURLOT Xavier, DEVAUD Emma, 5 ESPE, Binôme numéro uno, for the win

15 décembre 2020



## **Table des matières**

<b>1 QUESTIONS D'INTRODUCTION</b>	<b>1</b>
<b>Questions d'introduction</b>	<b>1</b>
<b>2 COMMANDE U SUR F</b>	<b>4</b>
<b>Mise en place de la commande U sur F</b>	<b>4</b>
2.1 Simulation Matlab . . . . .	4
2.2 Simulation sur le moteur virtuel . . . . .	5
2.3 Mise en place de la commande sur le prototype de démonstration . . . . .	6

# 1 QUESTIONS D'INTRODUCTION

(1) Quelles sont les rôles des boucles dites fast-loop et slow-loop du programme ? Quelles sont leurs périodes d'exécution ? Comment sont-elles déclenchées ?

- ☞ Fast loop : acquisition des courants de phase et calcul de la commande pwm, à 120us, triggered par l'ISR de l'adc
- ☞ Slow loop : mise à jour de la consigne de vitesse à 1.20ms. Appelée une fois sur dix dans l'interruption fast loop grâce à une variable cpt\_rSlow\_loop interne.

(2) La mesure des courants de phase, de l'angle et de la vitesse rotor sont-elles indispensables au bon fonctionnement de la commande U/f ? Pourquoi ?

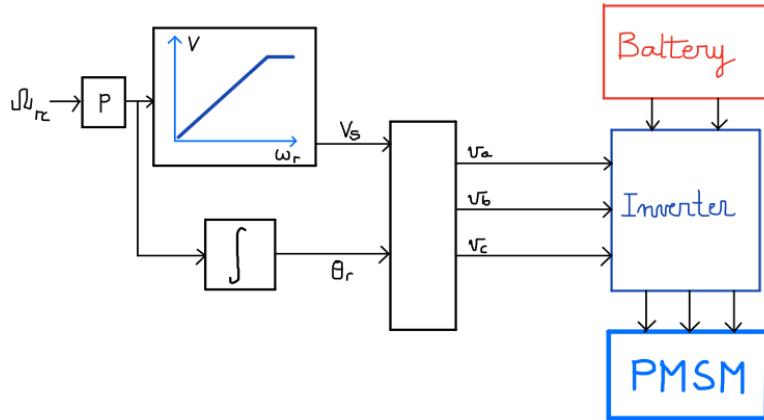


FIGURE 1 – Principe de la commande U/F (schéma G.Garcia)

- ☞ La commande U/F est une commande en boucle ouverte, elle exploite une constante du moteur  $K=U/F$  pour imposer une tension de phase. Donc ces mesures ne sont pas indispensables.
- ☞ Cependant, la mesure de l'angle moteur aide pour la transformation inverse de Park, au lieu d'estimer l'angle de manière logicielle. Elle peut également permettre de mieux gérer la phase d'alignement du rotor avec le champ statorique (différence d'angle au démarrage).

(3) Que représentent les commandes  $U_d$  et  $U_q$  ? Pourquoi cherche-t-on à forcer la commande  $U_d$  à 0 ? Peut-on réellement le garantir ? De quel(s) paramètre(s) dépend  $U_q$  ?

- ☞ Ce sont les amplitudes des tensions dans le plan de Park.
- ☞ On cherche à maximiser la composante tangentielle ( $U_q$ ), et à minimiser la composante radiale ( $U_d$ ) pour obtenir un couple maximal sur le rotor en alignant le vecteur courant du stator (qui produit le champ magnétique) sur le vecteur couple du rotor.
- ☞ Étant en boucle ouverte aucune surveillance n'est effectuée de ce fait on ne peut rien garantir.
- ☞ Les paramètres dont dépend  $U_q$  sont les suivants :
  - résistance interne du bobinage
  - l'inductance cyclique
  - vitesse de rotation du rotor

(4) Les rapports cycliques des commandes PWM appliquées sur les trois phases sont-ils mis à jour simultanément ? Si oui, pourquoi et comment ?

☞ Ils sont mis à jour simultanément, pour éviter des mauvaises configurations entre les bras de l'onduleur, et donc des transitoires importants. Les valeurs de PWM sont préchargées dans des registres (Config\_PWM0\_SignalParams) avant d'être chargés simultanément sur les 3 sous-modules (LoadPWM0\_Params).

**(5) Comment les courants de phase sont-ils mesurés ? Quels composants sont employés ? Quelle tension est lue par le microcontrôleur lorsqu'aucun courant de phase ne circule ?**

☞ Avec des résistances de shunt placées sur les bras de l'onduleur, on génère une tension proportionnelle au courant de phase, amplifiée par un ampli d'instrumentation et numérisée par les ADCs du microcontrôleur.

☞ Quand  $I=0$  la valeur de tension lue est  $VREF/2$  qui correspond à la tension du MCU (MCU\_VCC) divisée par deux.

**(6) Quel est le rôle de la phase de calibration ? Quelles valeurs de rapports cycliques sont imposés aux commandes PWM des trois phases ? Est-ce une valeur adéquate ?**

☞ Le rôle de la phase de calibration est de supprimer les offsets des mesures de courant (critique pour éviter tout courant DC dans la machine, i.e. pertes).

☞ Pour éviter qu'il y ait un courant constant qui circule dans la machine, on impose 50% de rapport cyclique aux commandes des PWM, soit une tension de phase nulle.

☞ Oui car avec cette commande aucun courant ne circule dans les résistances de shunt.

**(7) Est-ce que les 3 courants de phase sont lus ? Pourquoi ? Quel événement déclenche la lecture des courants ? Comment la synchronisation entre la lecture des courants et cet événement est-elle assurée ?**

☞ Non

☞ Seuls 2 courants de phase sont mesurés, car on peut déterminer mathématiquement l'un des courants à partir des deux autres, car  $I_a+I_b+I_c=0$ .

☞ La commutation de la sortie PWM déclenche le trigger CTU qui permet de fixer un délai avant d'amorcer l'ADC. Ainsi, la mesure en courant est toujours synchronisée avec la commutation, ce qui permet d'éviter de mesurer des phénomènes transitoires, et d'obtenir une valeur de courant moyen réel (pas d'aliasing).

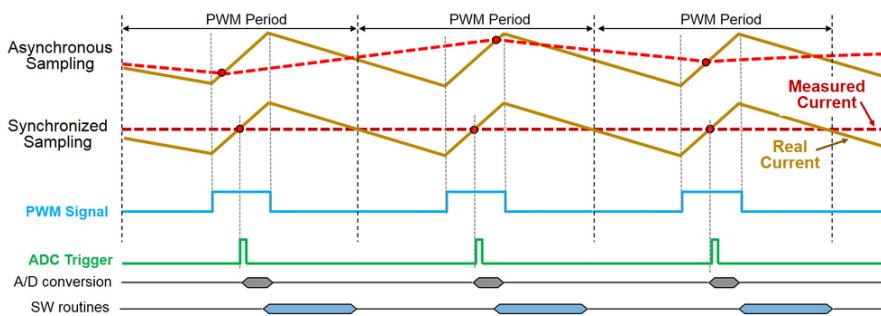


FIGURE 2 – Mesure de courant synchronisée (image NXP)

**(8) Pourquoi deux ADC indépendants sont-ils utilisés pour assurer la lecture des courants de phase ?**

☞ Pour avoir une lecture simultanée des courants : cohérence de phase entre les mesures, et précision temporelle de l'acquisition par rapport au cycle PWM.

**(9) Comment les courants de phase sont-ils mesurés sur la carte gate driver MOTOR GD ? Quelle contrainte le système de mesure des courants de phase impose-t-il sur l'alignement de la PWM ?**

☞ Les courants de phase sont mesurés sur la partie inférieure de chaque bras de l'onduleur, grâce à une résistance de shunt et un amplificateur différentiel (cf Fig.3).

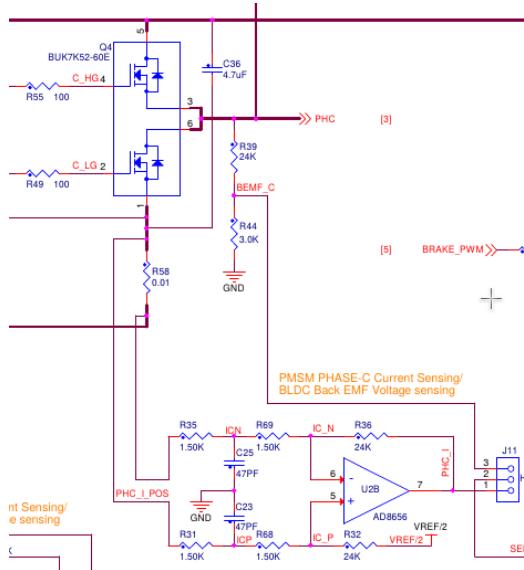


FIGURE 3 – Schéma de mesure d'un courant de phase

☞ Cette configuration implique que la mesure de courant doit être effectuée lorsque la partie inférieure du bras est ON, donc de manière synchrone à la PWM.

#### (10) Expliquez brièvement le principe et l'intérêt de la Space Vector Modulation (SVM).

☞ Le rôle de la SVM est de produire les PWM qui commandent les trois phases de l'onduleur à partir des commandes en tension  $U_\alpha$  et  $U_\beta$  issues de la transformée de Park inverse. Cette modulation particulière permet d'obtenir des orientations de champ plus fines qu'en commande bipolaire classique, et de maximiser le couple pour une tension d'alimentation donnée (3rd harmonic injection).

☞ L'onduleur est considéré comme une machine à états pouvant atteindre 8 états différents. De ce fait le principal intérêt réside dans le fait que cette méthode permet de calculer directement la commande pour les trois phases. Au contraire des techniques de modulation bipolaire et unipolaire qui considèrent chacune des phases séparément.

#### (11) Quelle est la différence entre l'angle rotor mécanique et électrique ? Lequel intervient dans la transformation de Park ?

☞  $\Omega = \frac{\omega}{p}$   $\Omega$  correspond à l'angle mécanique du rotor. Ici dans notre cas le moteur possède 3 paires de pôles.

☞ L'angle qui intervient dans la transformation de Park est mécanique l'angle entre le rotor et le champ statorique.

#### (12) Quel type de capteur acquiert la position et la vitesse du rotor ? Quel(s) sont les rôles du circuit PGA411 ?

☞ Le capteur utilisé est de type résolveur (cf Fig.4).

☞ Le circuit est capable de générer un sinus pour exciter le résolveur, et de mesurer les retours sin et cos afin de déterminer mathématiquement l'angle et la vitesse du rotor.

#### (13) Pourquoi n'acquiert-on pas directement la vitesse rotor à partir de celle que pourrait renvoyer directement le PGA411 ?

☞ Cette information n'est pas fiable pour de faibles vitesses, à cause de la grande résolution nécessaire dans ce cas. Il est alors préférable d'intégrer manuellement l'angle sur une période plus longue pour obtenir une bonne résolution sur la vitesse.

#### (14) Entrez les paramètres de la LUT reliant la vitesse rotor à atteindre et la tension de commande. Détaillez brièvement votre démarche pour les déterminer ?

☞ cf 2.1

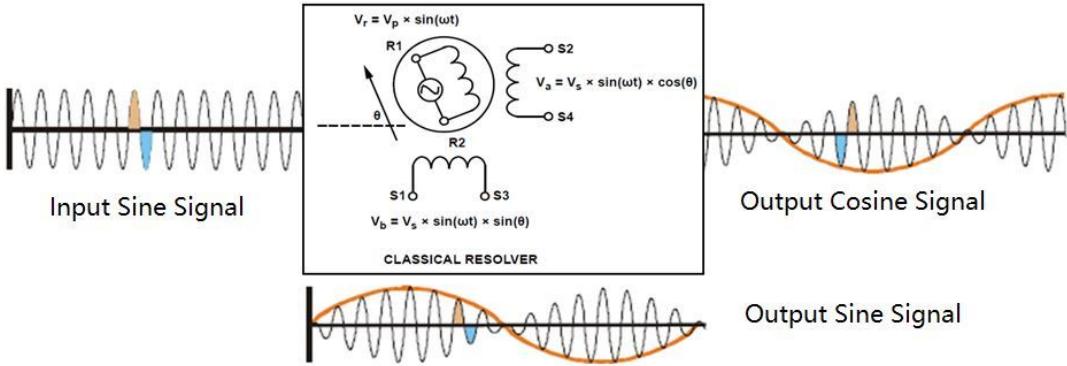


FIGURE 4 – Schéma de principe d'un résolveur

## 2 COMMANDÉ U SUR F

### 2.1 Simulation Matlab

Le modèle fourni sous Matlab permet simuler le comportement du banc d'essai en charge, d'observer les courants et tensions de phases, ainsi que les paramètres mécaniques (couple et vitesse). En particulier, on peut définir graphiquement la courbe de correspondance tension/vitesse de rotation (cf graphe Fig.1). Pour définir la relation, on procède de la manière suivante :

1. Choix d'une valeur de  $K_v > 2.2V/kRPM$  valeur min selon la doc moteur (tenir compte des frottements). On a donc la pente de la partie linéaire de la courbe.
2. Convertir les volts en rapport cyclique ( $100\% \Leftrightarrow V_{alim}$ )
3. Calculer la vitesse pour laquelle la tension d'alimentation est saturée :  $Vitesse_{max} = \frac{V_{alim} * 1000RPM}{K_v}$ , et saturer la sortie à  $100\%$  de rapport cyclique au delà.
4. Déterminer la tension min à appliquer au moteur pour vaincre les frottements. Calculer la vitesse de rotation à l'intersection du seuil bas et de la droite de  $K_v$ , et saturer à la tension min la commande pour les vitesses inférieures ou égales.

Graphiquement, le résultat est observable sur la figure suivante :

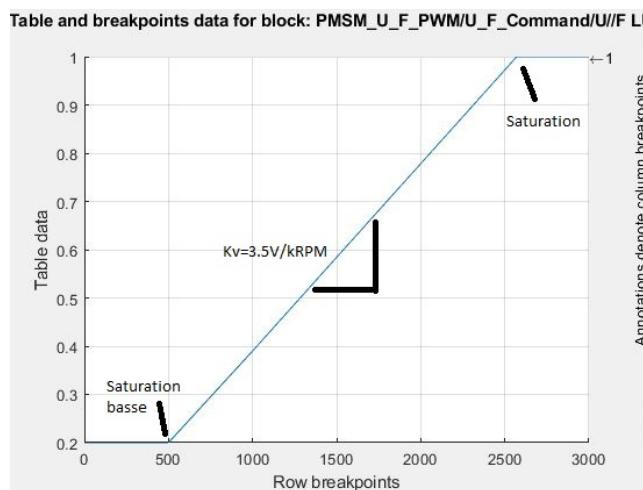


FIGURE 5 – Visualisation de la LUT sous matlab

Nous avons réalisé des simulations à partir du modèle simulink les résultats sont disponibles ci-dessous :

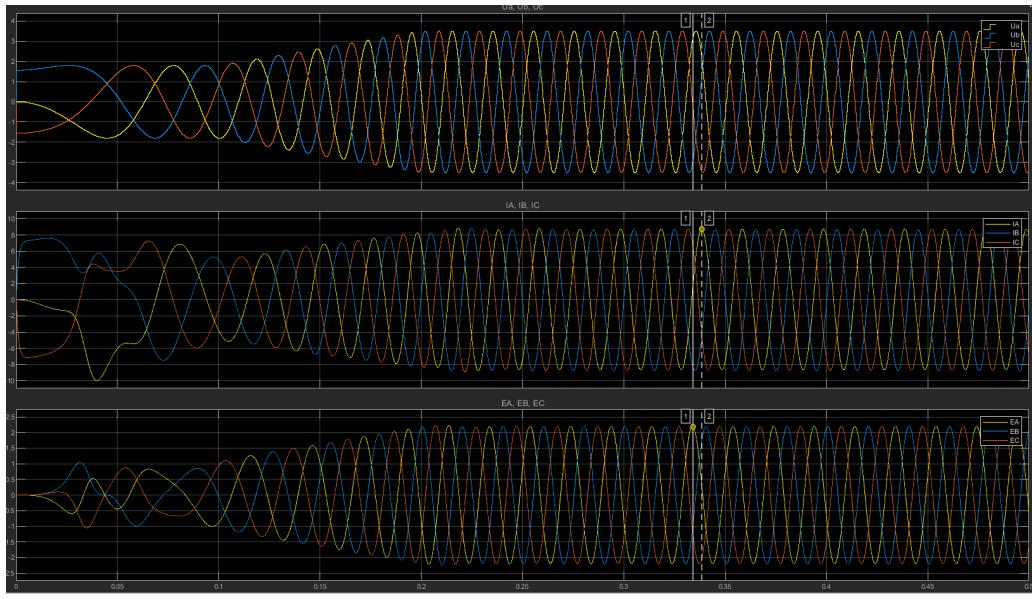


FIGURE 6 – Visualisation des courants, des tensions et de la fem pour une consigne à 1000 RPM

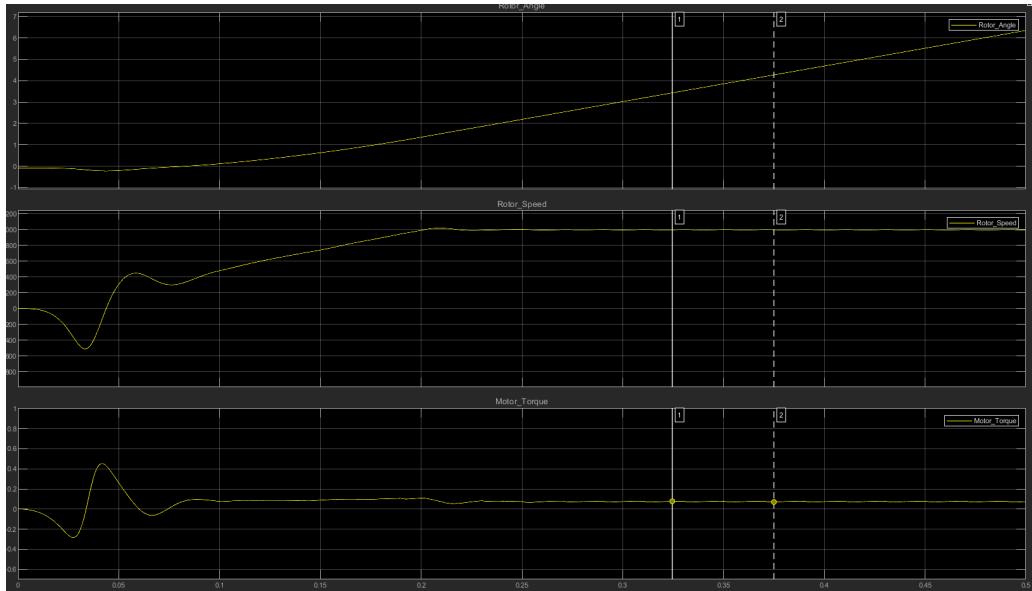


FIGURE 7 – Visualisation des grandeurs mécaniques : l'angle, vitesse et couple du rotor

## 2.2 Simulation sur le moteur virtuel

Une fois nos simulations validées sous Simulink, et avant de tester notre commande sur le moteur réel, nous avons implantée notre commande sur un "moteur virtuel". Nous avons utilisé S32 Design Studio ainsi que kit de développement DEVKIT-MPC5744P pour simuler le comportement du moteur. Pour observer les résultats, l'application FREEMASTER nous a permis de récupérer les différents signaux importants et de les afficher comme sur un oscilloscope.

Nous avons effectué plusieurs tests avec des valeurs de consigne différentes :

- +/- 30 RPM
- +/- 1000 RPM
- +/- 3000 RPM

Cela nous a permis de vérifier que les phases et la sortie évoluent correctement avant de tester sur le matériel. La simulation fonctionnant correctement pour les vitesses plus élevées ainsi que pour les vitesses négatives, nous avons mis en place la commande sur le banc moteur réel.

Ci dessous les résultats sous FREEMASTER pour une commande de 30 RPM.

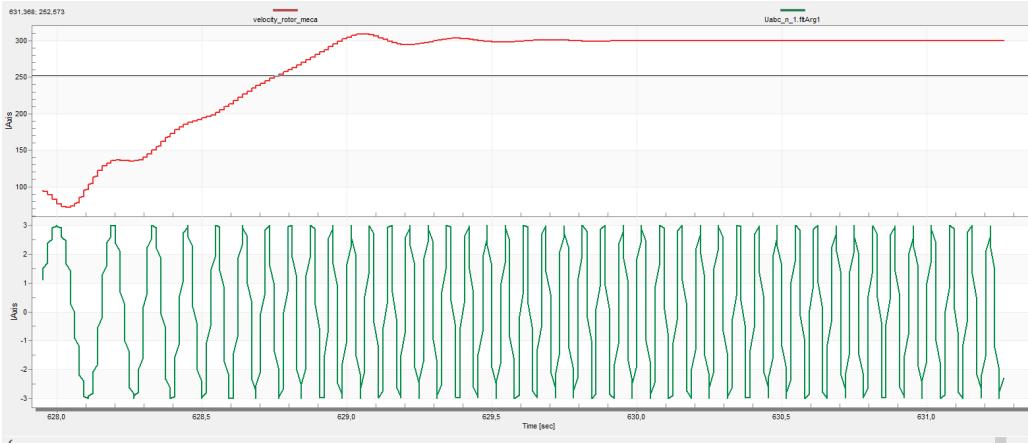


FIGURE 8 – Vitesse du rotor et phase UA du moteur virtuel.

### 2.3 Mise en place de la commande sur le prototype de démonstration

Le comportement du moteur en simulation étant vérifié (notamment la présence des phases sinusoïdales), il est alors possible de tester le code sur machine réelle.

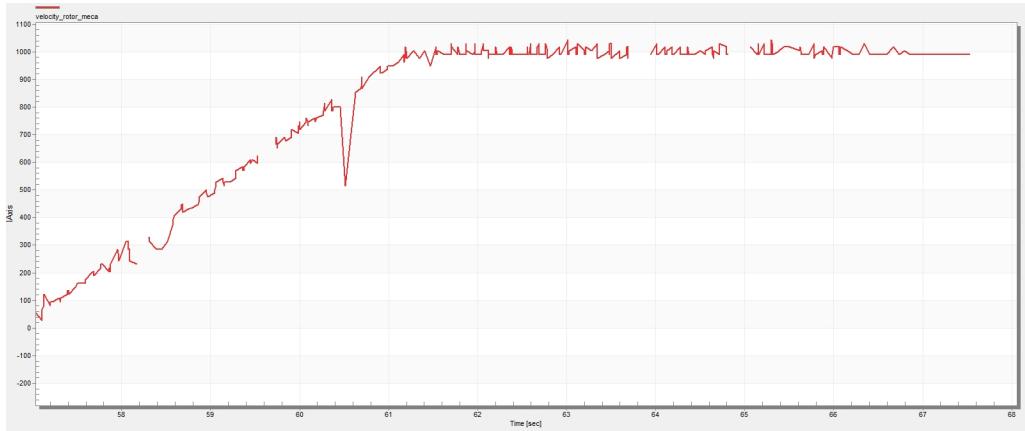


FIGURE 9 – Réponse en vitesse à une rampe de 1000RPM,  $K_v = 3.5V/kRPM$

Le réponse en vitesse correspond bien à la consigne et à la simulation. Le courant de phase affiché ici n'est pas représentatif à cause d'un faible taux d'échantillonage.

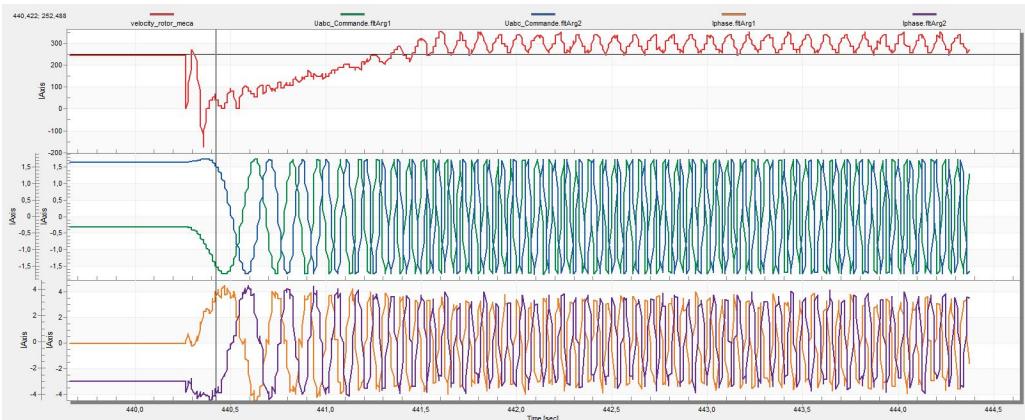


FIGURE 10 – Réponse en vitesse à une rampe de 300RPM,  $K_v = 3.5V/kRPM$

On observe sur les relevés à vitesse faibles une anomalie périodique de la vitesse, correspondant probablement à des

défauts de régularité de l'entrefer, causant une asymétrie du champ et donc du couple produit.

Vitesse (RPM)	30	300	1000
Erreur de vitesse & stabilité	oscillations periodiques (+-100%)	faibles oscillations periodiques autour de 300	léger bruit mais consigne suivie
Courant phase (A)	4.1	3.8	8.2