

Prototypage rapide de commande à base de DSP : Application à un soft starter

Grace GANDANEGARA¹, Lionel DORBAIS¹, Delcho PENKOV²

¹Schneider Electric, Strategy&Technology, 37 Quai Paul Louis Merlin, 38050 Grenoble Cedex 9

²Schneider Electric, Energy Business, 20 Avenue des Jeux Olympiques, 38050 Grenoble Cedex 9

RESUME—Dans les projets d'innovation en génie électrique, la modélisation prend une place très importante, notamment dans la conception de la loi de commande. Par ailleurs, grâce à l'outil de génération de code automatique, la validation de cette commande pourra être faite sur une plateforme de prototypage rapide de commande. Pour réduire le temps à partir de la conception de la commande au produit final, une plateforme de prototypage rapide à base de DSP a été développée. Elle permet de prendre en compte les contraintes liés aux périphériques de microprocesseur dans le modèle de simulation sous Matlab/Simulink et, grâce aux outils de Matlab/Simulink, de générer le code de commande automatiquement vers le microprocesseur. Cette plateforme contient également la partie hardware pour faciliter la conception de la carte de commande. Un cas d'étude et une application sur un projet de soft starter sont présentés pour montrer l'intérêt de cette plateforme.

Mots-clés— Plateforme de prototypage rapide de commande, DSP, Matlab/Simulink, modèle, simulation, génération de code automatique, soft starter.

1. INTRODUCTION

Le principe de Model-Based Design [1][2] est de plus en plus appliqué dans les projets d'innovation ou de renouvellement de produits. La modélisation prend la place importante dans la conception de la commande ainsi la validation par simulation ou émulation.

Plusieurs logiciels de modélisation et simulation sont disponibles. Pour les automaticiens, Matlab/Simulink est très répandu car il permet d'intégrer la commande en fonction de transfert, en S function ou en script Matlab dans le modèle de simulation. Le couplage avec des boîtes à outils tels que Sim Power System, Simscape ou PLECS Blockset nous permet de modéliser les systèmes en génie électrique.

Une fois que la commande est validée par la simulation, il est nécessaire de l'implémenter ensuite rapidement sur une maquette à l'aide d'une plateforme de prototypage rapide. Plusieurs plateformes existent et sont commercialisées, avec leurs logiciels et interfaces de programmation.

Pour le besoin des projets d'innovation en électronique de puissance et en commande de moteur, une plateforme de prototypage rapide de commande à base d'un DSP a été développée dans notre département.

Cet article sera consacré à cette plateforme de prototypage rapide. Dans un premier temps, la plateforme de prototypage rapide ainsi le choix de DSP sont exposés. Ensuite, un cas d'étude est présenté comme exemple. L'application dans un projet d'innovation d'un soft starter est décrite. Enfin, les conclusions et les perspectives de ces travaux sont présentées.

2. PROTOTYPAGE RAPIDE ET DSP

2.1. Rôle du prototypage rapide dans l'approche Model-Based Design

Dans des projets d'innovation, l'approche Model-Based Design est souvent adoptée. Les avantages de cette approche ont déjà été démontrés dans [3][4], tels que l'efficacité (les itérations de validation bien définies), la qualité (la vérification de manière continue par rapport à la spécification) et la réduction de coût et de temps de développement.

Selon le principe de Model-Based Design, le développement de la commande pour le système peut se présenter comme le cycle V présenté à la Fig.1. La spécification est consultée dans toutes les étapes pour la validation. Dans cet article, nous nous sommes intéressés aux étapes Conception, Prototypage et Génération de Code.

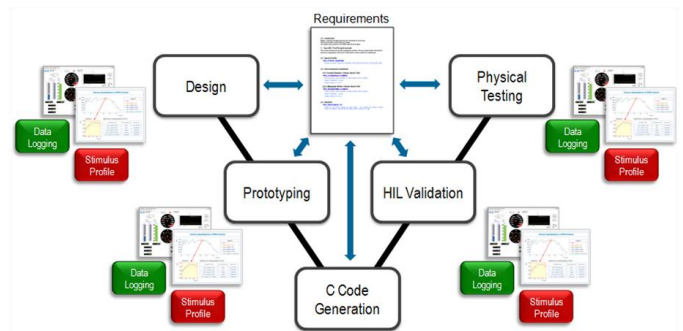


Fig.1 Cycle V de développement de commande.

Les systèmes sont modélisés afin de pouvoir faire la synthèse de lois de commande appropriées selon le cahier des charges ou la spécification exigée. Puis, les modèles et la synthèse de commande sont validés par simulations. Après la validation par la simulation, le test sur une maquette pourrait alors être envisagé. La plateforme de prototypage rapide de commande peut être utile car il permet :

- d'implémenter rapidement la commande,
- de valider les modèles selon la spécification ou le cahier des charges,
- de tester et d'améliorer les lois de commande, et
- de mettre en place rapidement les systèmes de commande pour tester des composants.

2.2. Plateformes existantes de prototypage rapide de commande

Il existe plusieurs plateformes de prototypage rapide de commande telles que RT Lab (Opal RT)[6], Compact RIO (National Instruments)[7], BoomBox (Imperix)[8], dSPACE[9], xPC-Target (Mathworks)[10] ou Speedgoat[11]. Certaines plateformes ont leur propre logiciel de développement. Dans ce cas là, il faudrait que la commande soit recodée ou redésignée. Cette étape pourrait être une source d'erreurs et de différences par rapport au résultat de simulation. Les plateformes comme RT Lab, Compact RIO, xPC-Target et Speedgoat peuvent s'interfacer directement ou facilement avec Matlab/Simulink. Par contre, ces plateformes sont encore loin du produit final car souvent la commande sera embarquée dans un ou plusieurs microprocesseurs ou FPGAs.

Pour avoir les avantages d'un prototypage rapide de commande comme cités précédemment et être proche du produit final en utilisant Matlab/Simulink comme l'environnement unique de logiciel depuis la conception jusqu'à la génération de code, il est décidé d'utiliser directement un microprocesseur de type DSP pour la plateforme de prototypage rapide.

2.3. Choix du microprocesseur DSP

L'utilisation très répandue de microprocesseurs TMS320F28335 est bien remarquée sur les produits avec un ou des convertisseurs statiques, comme dans un variateur de vitesse, onduleur UPS, onduleur solaire, etc. En effet, ce microprocesseur a une fréquence élevée de 150 MHz, des périphériques intégrés (ADC, ePWM, eCAP) et des modes de communication (CAN, SPI, I2C, RS232, ...)[5]. Sachant que ce microprocesseur de type DSP est inclus parmi les cibles que Matlab/Simulink avec l'outil Embedded Coder pourra supporter pour générer le code automatiquement. Par ailleurs, ce microprocesseur peut aussi traiter les calculs en point flottant de type single de manière plus optimisée par rapport à d'autres microprocesseurs. Ceci facilite la mise au point de contrôle implémenté.

2.4. Plateforme de prototypage rapide à base de DSP

L'outil Embedded Coder permet d'avoir le modèle « coder » qui sera utilisé dans la génération de code, mais il ne peut pas être simulé. Par conséquence, ce modèle ne pourra pas être validé par la simulation, ce qui pourrait être un inconvénient. Effectivement, l'architecture et la configuration de périphériques ainsi la gestion des interruptions ont le rôle très important dans le fonctionnement de la commande. Par exemple, dans un onduleur triphasé, l'acquisition de la valeur de courant devrait être faite aux bons moments (au milieu de l'état de conduction). Ceci pourrait être réalisé grâce à l'interruption générée aux moments précis par le module PWM associé. Si cette partie n'est pas simulée, les variables mesurées ne seront pas correctes en simulation.

Pour ces raisons, une plateforme de prototypage rapide à base de DSP TMS320F28335 a été développée. Cette plateforme contient la partie software sous Matlab/Simulink en créant les bibliothèques DSP Simu, DSP Coder, les scripts et les templates, et la partie hardware sous le logiciel Cadence pour faciliter la conception de la carte de commande. Grâce aux fichiers templates, les paramètres du modèle sont déjà configurés (soit en simulation, en « RAM » ou en FLASH »).

Avec cette plateforme, les contraintes liées aux périphériques et des interruptions sont intégrées dans le modèle de simulation de Matlab/Simulink, grâce à la bibliothèque DSP Simu (Fig.2). Dans ce modèle de simulation, nous séparons la partie physique de la partie software. La partie physique contient le modèle fonctionnel des périphériques de DSP, toutes les connectiques de DSP, les circuits de conditionnement de signaux ainsi que le modèle du système à commander. Fig.3 présente un exemple du modèle hardware des modules ePWMs de DSP. La structure de synchronisation entre ces modules est représentée et prise en compte dans la modélisation. Pour un module d'ADC, le modèle hardware respecte également le temps d'acquisition, la saturation de tension et la mise à échelle. L'exemple d'un signal sinusoïdal de 1000 Hz connecté à 8 entrées ADC est présenté à la Fig.4.

La partie software contient les paramétrages des périphériques, la gestion des interruptions, et la commande discrétisée qui seront programmés dans le DSP. Le système à contrôler est modélisé avec PLECS Blockset dans la partie physique.

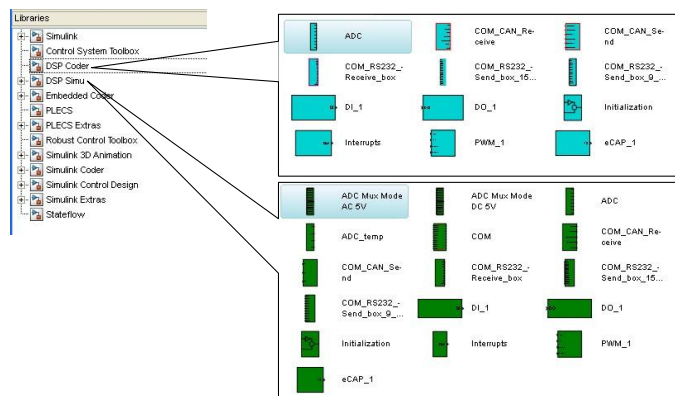


Fig.2 Bibliothèque DSP Simu et DSP Coder.

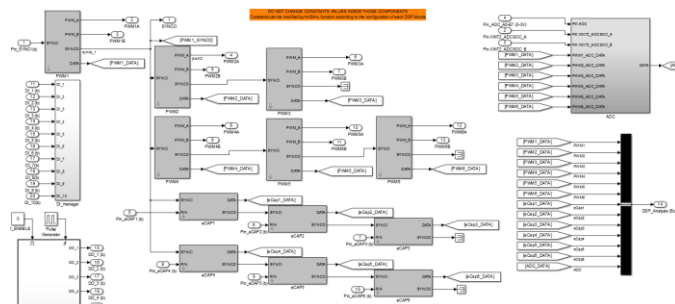


Fig.3 Exemple du modèle hardware des ePWMs de DSP.

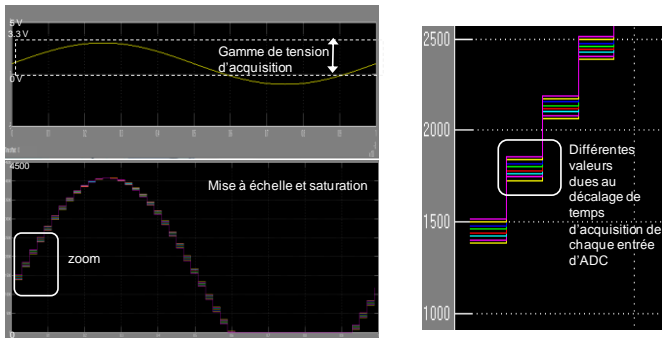


Fig.4 Fonctionnement en simulation d'ADC pour une sinusoïde de 1000 Hz.

Avec ce modèle, la commande peut être validée en simulation. Une fois validée, avec une seule ligne de commande, le modèle de simulation est transformé en modèle coder (et *vice versa*). Ceci est un avantage car il n'est pas nécessaire de refaire le modèle de la commande. Les paramètres qui ont été saisis dans les composants DSP Simu (ADC, PWM, DI, DO, ...) sont recopiés directement dans les composants DSP Coder. Il y a deux types de modèle coder : soit en RAM soit en FLASH. La différence se pose sur la volatilité du code compilé dans le DSP cible. Sachant que la génération de code en FLASH nécessite plus de temps de traitement par rapport à RAM.

A partir de ce modèle coder, la génération de code automatique pourra être faite vers le DSP cible sur la carte de commande. Enfin, le test sur le banc de test peut être effectué. Ce processus est décrit dans la Fig.5.

La bibliothèque hardware contient les schémas électroniques des fonctions qui sont utilisables avec ou pour le DSP. Par exemple le schéma électrique de filtre, de communication, ou de conditionnement de signal d'un capteur afin de l'adapter avec l'entrée analogique de l'ADC peut être trouvé dans la bibliothèque. Tous les schémas sont faits dans le logiciel PSpice Cadence et documentés. Grâce à cette bibliothèque *scheme-reusing*, le temps de conception hardware sera réduit. Fig.6 montre l'exemple pour un filtre du deuxième ordre dont la documentation décrit la fonction de transfert et l'outil de calcul. Ce composant de bibliothèque peut être exporté directement dans le projet hardware sous Cadence.

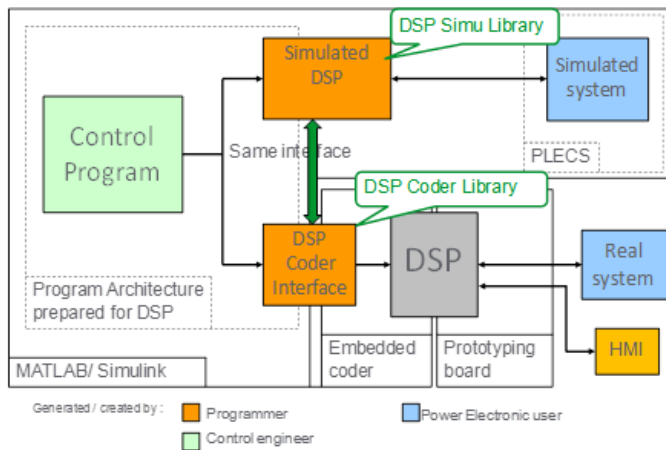
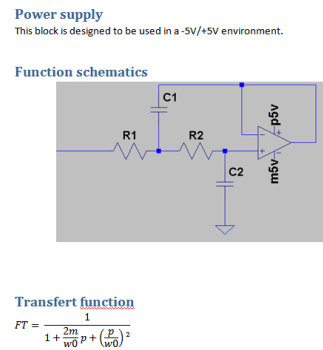
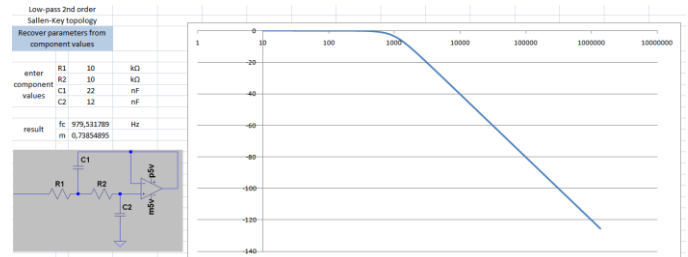


Fig.5 Plateforme de prototypage rapide à base de DSP.



(a) schéma et fonction de transfert



(b) outil de calcul

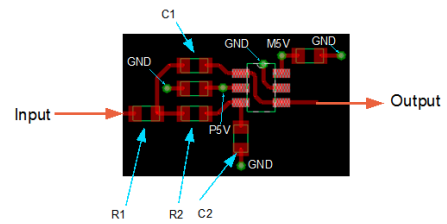


Figure 4 : Layout example

(c) Empreint schématique

Fig.6 Exemple de la bibliothèque hardware pour un filtre du deuxième ordre.

L'outil profilage (profiler) de Embedded Coder permet d'analyser le temps d'exécution de chaque bloc. Il permet également de détecter un dépassement de temps de calcul et de vérifier la synchronisation entre blocs.

Grace à S-fonction de Matlab/Simulink, il est possible d'intégrer un bout de programme écrit en langage C dans le modèle de commande. Il sera donc greffé dans le code global généré par Matlab/Simulink. Cette fonctionnalité permet d'intégrer les fonctions spécifiques déjà développées auparavant (par exemple la protection, le filtre numérique, ...) ou la fonction spéciale pour gérer les périphériques qui ne peut pas être faite directement avec Embedded Coder (par exemple le changement d'affectation d'une E/S numérique).

3. CAS D'ETUDE

Pour valider cette plateforme, nous avons pris un cas d'étude d'un correcteur de type PI. Le modèle de circuit électronique est présenté à la Fig.7. Un prototype a été réalisé pour pouvoir se connecter directement avec la carte d'essai (*Experimentercard*) qui contient une carte de control de DSP (Fig.8). Les résultats obtenus par le modèle de simulation et par le test réel sont présentés à la Fig.9. Nous pouvons voir la cohérence entre ces résultats.

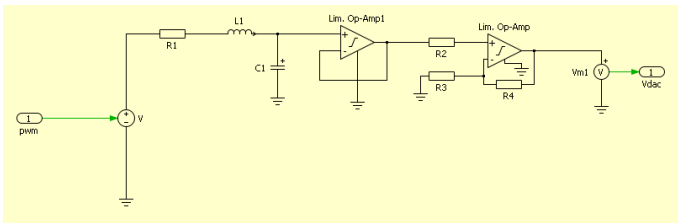


Fig.7Modèle d'un correcteur PI.

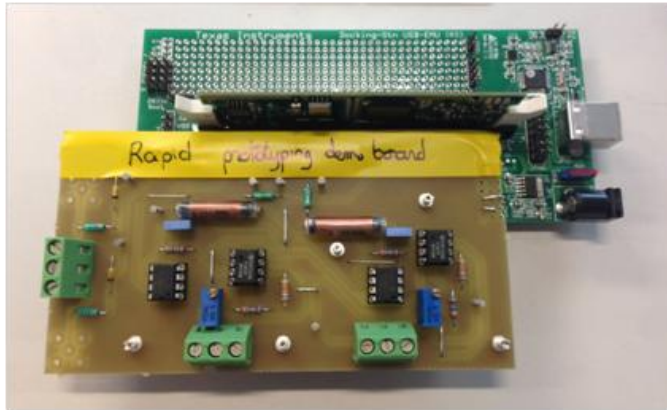


Fig.8Prototype du correcteur PI.

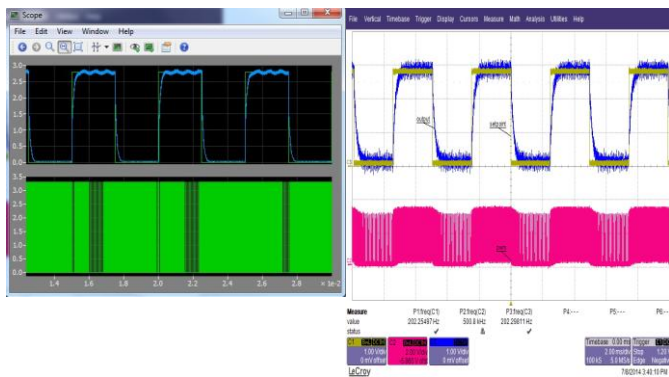


Fig.9Résultats pour un signal carré de 200 Hz : simulation et test réel.

4. APPLICATION SUR UN SOFT STARTER

Le projet a pour but d'améliorer un soft starter. Le convertisseur est constitué par deux thyristors en tête bêche (stack) et plusieurs stacks pour chaque phase. Une étude a été menée pour identifier la cause des casses sur des soft starter en moyenne tension (MT) et trouver la solution. En effet, la forte montée de courant au début de la fermeture des thyristors est la cause principale de ces casses. La solution optimale proposée est de donner l'ordre de fermeture aux bons moments quand la tension de circuit *snubber* soit dans le creux ou le minimum local. Il n'est pas facile de détecter les minimums locaux à partir de cette tension. En revanche, il est faisable à partir de la dérivée de courant qui traverse le circuit *snubber* pendant l'ouverture de thyristor. Les capteurs Rogowski sont donc utilisés pour mesurer les dérivées de courants. La validation a été faite d'abord pour le module de soft starter basse tension pour valider le principe.

Pour réaliser cette solution, une nouvelle commande et la carte associée devraient être faites. Ce projet a été utilisé comme une opportunité pour montrer la faisabilité de la plateforme de prototypage rapide. Le DSP TMS320F28335 a été choisi comme microprocesseur pour sa performance. Une nouvelle carte de commande a été conçue pour intégrer les

capteurs Rogowski et pour pouvoir entrer dans le module soft starter. La conception de la carte de commande a été faite en utilisant la bibliothèque hardware selon la spécification.

La première étape consiste à valider cette plateforme et la modélisation du système.

Pour cela, un mode manuel est programmé. Dans le mode manuel, l'angle d'amorçage des thyristors (alpha) est donné par utilisateur. Cet angle est mesuré à partir du passage de zéro de la tension. Pour cela, les ePWMs sont synchronisées par rapport au passage de zéro de la tension. Par ailleurs, pour effectuer le traitement du signal qui nécessite 48 échantillons par période, des interruptions sont générées à cette fréquence pour faire l'échantillonnage et le calcul rapide. Sachant que la fréquence du signal peut légèrement varier d'où l'intérêt de bien calculer la fréquence grâce au traitement du signal. Fig.10 présente le modèle de soft-starter avec la commande.

Ces conditions sont prises en compte dans le modèle de simulation grâce à la bibliothèque DSP Simu.

A partir du modèle coder associé, le code de la commande est généré vers le microprocesseur de la carte de commande.

La même charge passive (R et L) a été prise sur le banc de test et dans le modèle de simulation. La Fig.11 présente les résultats obtenus pour un alpha de 110°.

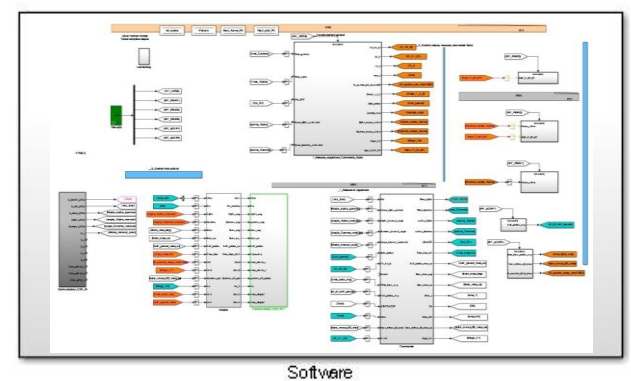
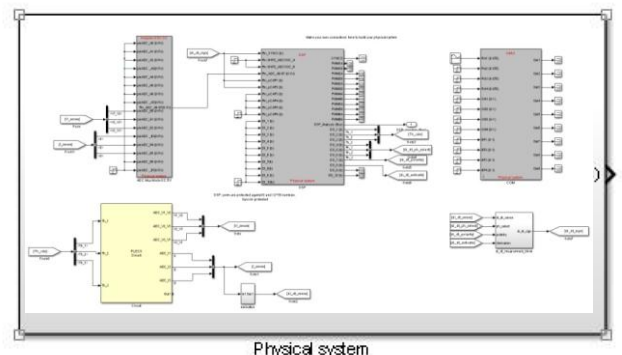


Fig.10Modèle de soft starter avec la commande.

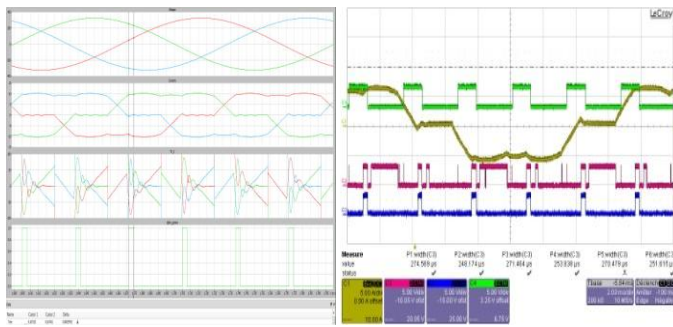


Fig.11 Résultats sur un soft starter pour un angle d'amorçage de thyristor de 110°: simulation et test réel.

En mode manuel, il est également possible de donner l'ordre de fermeture des thyristors à partir de passage par zéro de la dérivée de courant. Ce mode est appelée delta-gamma. Un circuit de conditionnement de signal a été conçu pour amplifier et saturer les signaux de dérivée de courants sortant des capteurs. Ces signaux amplifiés de dérivée de courant permettent de synchroniser un module de capture (eCAP) qui mesurera avec précision la durée d'une demi-période de la dérivée de courant transitoire pendant l'ouverture de thyristor. Fig.12 présente le résultat de test sur un moteur asynchrone de 90 KW pour une commande manuelle delta-gamma pour 10 demi-périodes, avec la durée d'une demi-période de 200 μ s. La précision de la détection de passage par zéro de la dérivée de courant et de la fermeture de thyristor est constatée (2,047 ms au lieu de 2 ms).

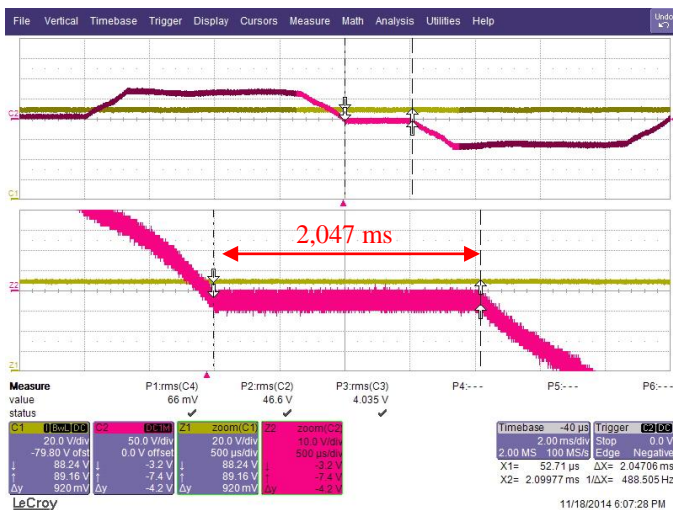


Fig.12 Résultats sur un soft starter pour un ordre de fermeture de thyristor de 10 demi-périodes sur un moteur asynchrone de 90 KW.

Une fois que les deux modes manuels sont validés, le mode automatique peut être intégré.

5. CONCLUSIONS

La plateforme de prototypage rapide à base de DSP a été présentée. Elle permet de réduire le temps de conception de commande jusqu'à l'implémentation sur un microprocesseur cible. Le fait d'utiliser un seul logiciel (Matlab/Simulink) depuis la modélisation jusqu'à la génération de code facilite le processus et évite la reprogrammation de commande dans un autre logiciel, qui est souvent une source d'erreurs. Par ailleurs, cette plateforme est plus réaliste car on prend en compte l'architecture de DSP, les contraintes des périphériques et la gestion des interruptions dans le modèle de simulation.

L'application sur un cas d'étude et un projet de soft starter a montré l'intérêt et la faisabilité de cette plateforme.

Cette plateforme pourra être élargie afin de pouvoir supporter d'autres microprocesseurs cibles.

6. REFERENCES

- [1] M. Ahmadian, Z.J. Nazari, N. Nakhaee, Z. Kostic, « Model based design and SDR », 2nd IEE/EURASIP Conference on DSP Enabled Radio, ISBN 0-86341-560-1, 2005.
- [2] A. Frederiksen, « Model-Based Design of Advanced Motor Control Systems », Analog Devices Technical Article, MS-2577, 2013.
- [3] J. Godbersen, « Experiences of Introducing Model-Based Design at Danfoss Solar Inverters », Nordic Matlab Expo 2014, Stockholm, Sweden, May 2014.
- [4] L. Michaels *et al.*, « Model-Based Systems Engineering and Control System Development via Hardware-in-the-Loop Simulation », SAE International, 2010-01-2325.
- [5] Texas Instruments, « TMS320F28335, TMS320F28334, TMS320F28332, TMS320F28235, TMS320F28234, TMS320F28232 Digital Signal Controllers (DSCs) – Data Manual », SPRS439M, August 2012.
- [6] RT Lab : www.opal-rt.com
- [7] Compact RIO : www.ni.com/compactrio/
- [8] BoomBox : imperix.ch/products/control/boombox/intro
- [9] dSPACE : www.dspace.com
- [10] xPC Target : fr.mathworks.com/videos/introduction-to-xpc-target-turnkey-68908.html
- [11] Speedgoat : www.speedgoat.ch