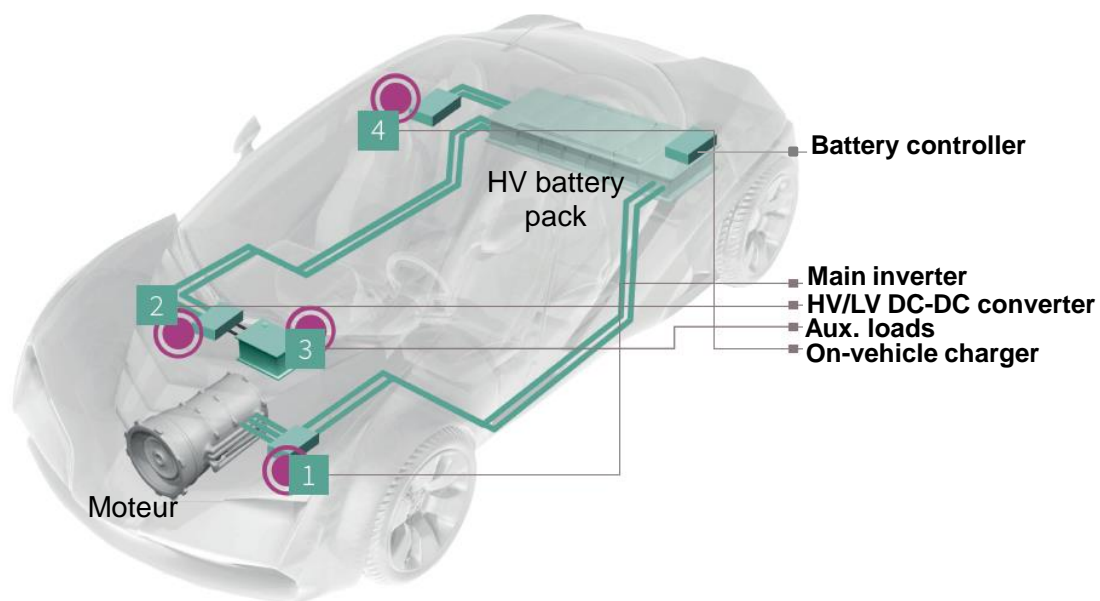


Bureau d'étude Electronique Automobile Module "Main Inverter"



<http://www.alexandre-boyer.fr>

Contenu

I - Contexte du bureau d'étude – commande moteur et électronique pour la traction électrique	3
II - Objectifs du bureau d'étude	5
III - Enoncé du BE.....	5
1. Présentation générale du démonstrateur Main Inverter.....	5
2. Tâches à accomplir.....	7
a. Préalable (séance 1).....	7
b. Partie 1 – Analyse de commande U/f	8
c. Partie 2 – Mise au point de la commande FOC	10
IV - Liste des documents disponibles en ligne	11
V - Annexe 1 – Présentation du matériel.....	13
1. Le banc moteur.....	13
2. Carte microcontrôleur MPC5744P (Panther).....	14
3. Carte gate driver + onduleur MotorGD.....	15
4. Carte interface résolveur PGA411Q1-EVM	17
VI - Annexe 2 - Les logiciels de conception utilisés	19
VII - Annexe 3 – Modèle simplifié (en C) du moteur	20

I - Contexte du bureau d'étude – commande moteur et électronique pour la traction électrique

Le bureau d'étude électronique automobile s'inscrit dans le contexte de l'électrification des véhicules, c'est-à-dire dans l'utilisation directe de l'énergie électrique pour la traction du véhicule, en remplacement des moteurs à combustion interne. Cette tendance nécessite le développement de nouveaux équipements électriques et électroniques, venant s'ajouter aux nombreux autres organes électroniques ajoutés ces dernières années.

La Figure 1 présente une vue d'ensemble typique des équipements embarqués (Electronic Control Unit ECU) indispensables dans un véhicule électrique. L'énergie du système est fournie par plusieurs batteries : une haute tension (HV battery) pour les étages de forte puissance (onduleur de puissance du Main Inverter), une autre faible puissance pour les étages de faible/moyenne puissance. Les ECU principaux sont :

- 1. Main inverter: cet équipement contrôle le moteur électrique responsable de la traction du véhicule. C'est lui qui gère le comportement de conduite du véhicule. Quelle que soit la nature du moteur (synchrone, asynchrone, DC brushless), cet ECU présente la même architecture et fonctionne de la même manière. Il récupère aussi l'énergie produite lors du freinage et la renvoie vers la batterie HV. Dans ce bureau d'étude, la partie récupération d'énergie ne sera pas abordée.
- 2. HV/LV DC-DC converter : cet étage produit la tension 12 V nécessaire à la batterie LV à partir de la batterie HV.
- 3. Auxiliary loads : l'électrification d'un véhicule ne concerne pas simplement le moteur de traction, mais aussi l'ensemble des fonctions du véhicule actionnées autrefois par le moteur à combustion interne (climatisation, pompe à huile, chauffage...).
- 4. On-board charger : la batterie HV doit être régulièrement rechargée sur le réseau électrique secteur, selon les différents standards en cours dans le monde.
- 5. Battery controllers: l'utilisation de batterie, que ce soit plomb-acide ou Lithium, requiert une surveillance constante de la tension, l'état de charge et la température, pour éviter non seulement le vieillissement prématuré lié à des décharges ou des surcharges, mais aussi des conditions de fonctionnement critique.

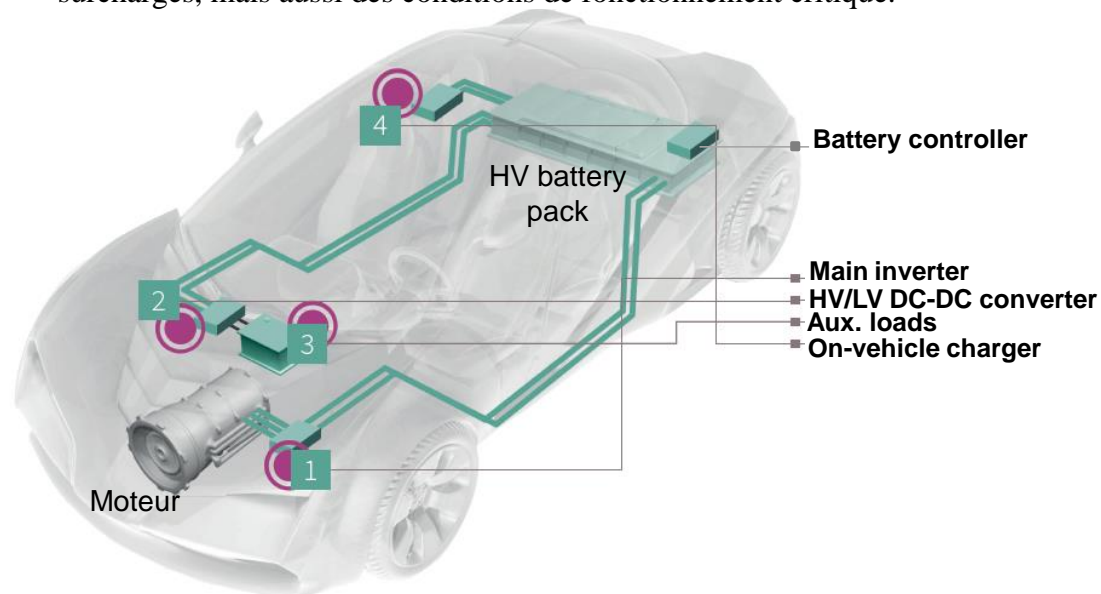


Figure 1 - Vue d'ensemble des équipements électriques et électroniques responsables de la traction d'un véhicule électrique [Infineon]

Ce bureau d'étude se focalise sur le module Main Inverter, principalement sur les circuits électroniques le composant, sur la commande du moteur et le logiciel embarqué. Son architecture typique est décrite à la Figure 2. Celui-ci est conçu pour minimiser les pertes par commutation de l'onduleur, optimiser la dissipation thermique, améliorer le comportement en conduite, la stabilité du véhicule et garantir la sécurité des passagers. Le cœur de cet équipement est l'onduleur de puissance, chargé de piloter un moteur triphasé (synchrone à aimants permanents, asynchrone ou brushless). Selon la tension de la batterie, il est composé de transistors MOSFET (< 48 V) ou d'IGBT (> 48 V). La commande de ces transistors passe par un "Gate Driver". Une isolation galvanique est requise entre les domaines LV et HV de l'ECU. Celle-ci peut être intégrée dans le gate driver. La commande du moteur est calculée en temps réel à partir d'un microcontrôleur 32 bits multi-cœur, responsables de la gestion de l'équipement. Le pilotage du moteur nécessite aussi la présence de capteurs de position, de vitesse du rotor du moteur électrique (résolveur, encodeur, capteur à effet Hall), et donc de circuits d'interface. D'autres mesures, comme les courants de phase, la tension de batterie, la température sont aussi requises. L'alimentation du microcontrôleur est généralement assurée par un ensemble de régulateurs, intégrés à l'intérieur d'un composant appelé Power System Basis Chip (Power SBC). Celui-ci intègre aussi des fonctions de surveillance des différents domaines de tension, afin de déclencher le passage dans un mode de fonctionnement sûr (safe). Cette fonction s'inscrit dans l'exigence de sûreté de fonctionnement (voir paragraphe suivant). Enfin, il intègre les interfaces de communication vers les autres ECU du véhicule (bus LIN, CAN, Flexray). Les performances requises, les contraintes d'intégration et de coût nécessitent l'utilisation de circuits intégrés dédiés pour l'automobile.

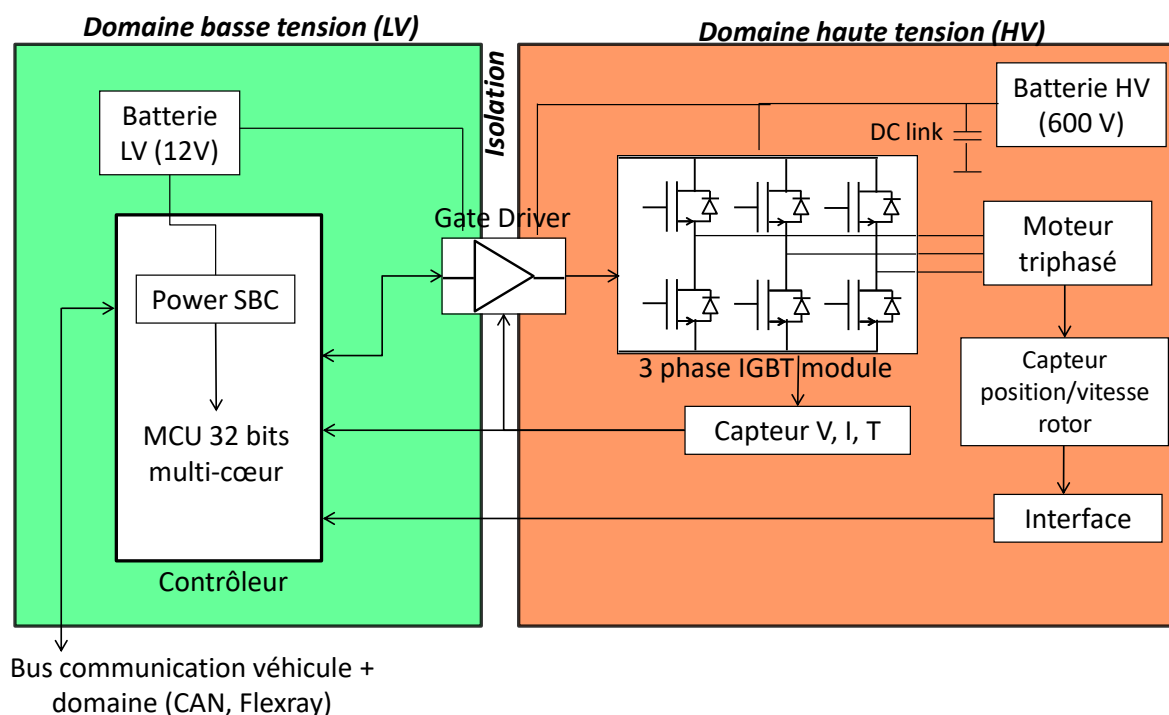


Figure 2 - Architecture typique du module Main Inverter

II - Objectifs du bureau d'étude

Le projet donné dans ce BE est la mise en œuvre d'un démonstrateur matériel reproduisant l'équipement "Main Inverter" dédié à la traction d'un véhicule électrique. Les objectifs principaux sont :

- mettre au point une commande avancée pour le pilotage d'un moteur électrique typique d'une traction automobile, en l'occurrence une commande vectorielle (*Field Oriented Control*), en partant d'une commande U/f
- mettre en œuvre des circuits intégrés dédiés aux applications automobiles (microcontrôleurs, gate driver, capteurs)
- utiliser une librairie de fonctions mathématiques dédiées au contrôle moteur

Les circuits utilisés sont ceux fournis par NXP Semiconductors, partenaire de ce projet. Bien que le moteur utilisé dans ce bureau d'étude présente une puissance électrique moindre que les moteurs employés dans les véhicules électriques pour des raisons de sécurité et de coût, les architectures électroniques, les commandes et le logiciel embarqué seront représentatifs d'un vrai calculateur de commande moteur.

III - Enoncé du BE

1. Présentation générale du démonstrateur Main Inverter

Les différents composants sont intégrés à l'intérieur d'un même ECU (*Electronic Control Unit*). Dans la pratique, ce calculateur est connecté à d'autres équipements du véhicule. Dans cette version, aucune connexion vers d'autres calculateurs ne sera prévue. La Figure 3 décrit l'architecture matérielle de l'application, en mentionnant les composants employés.

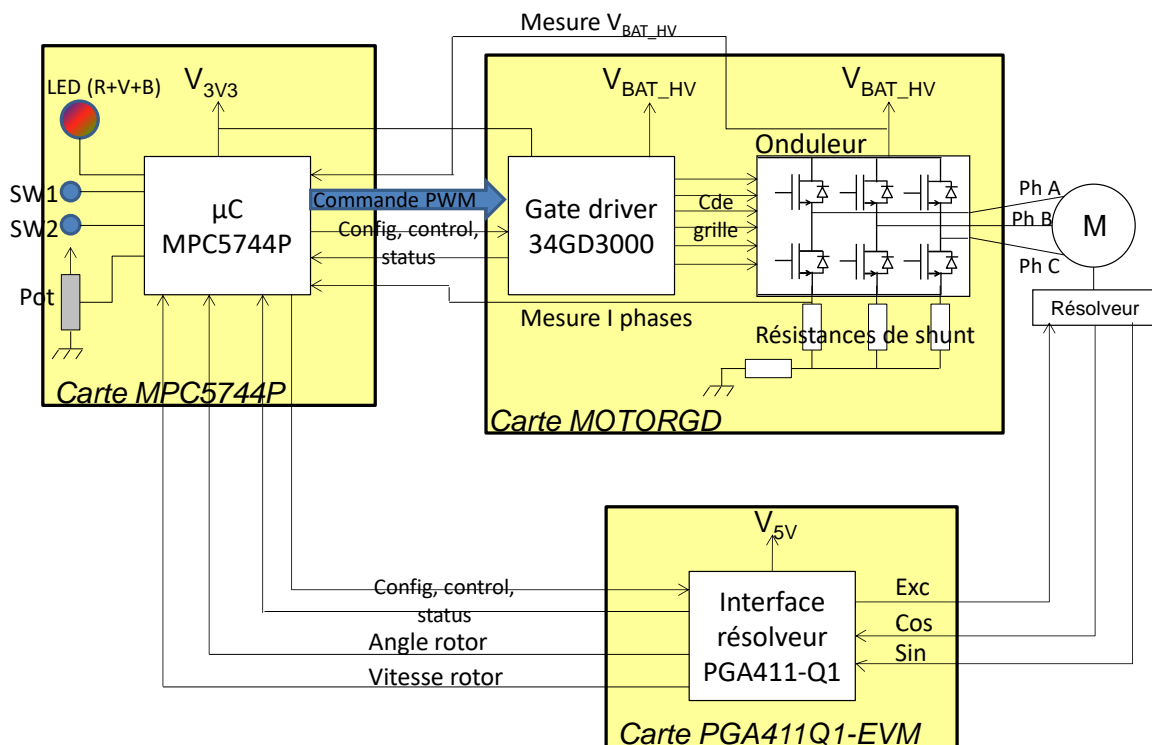


Figure 3 - Architecture matérielle du module Main Inverter à développer dans le bureau d'étude

Les composants sont montés sur des cartes d'évaluation, décrites à l'annexe 1 :

- carte microcontrôleur MPC5744P
- carte gate driver+ onduleur MOTORGD
- carte interface résolveur PGA411Q1-EVM

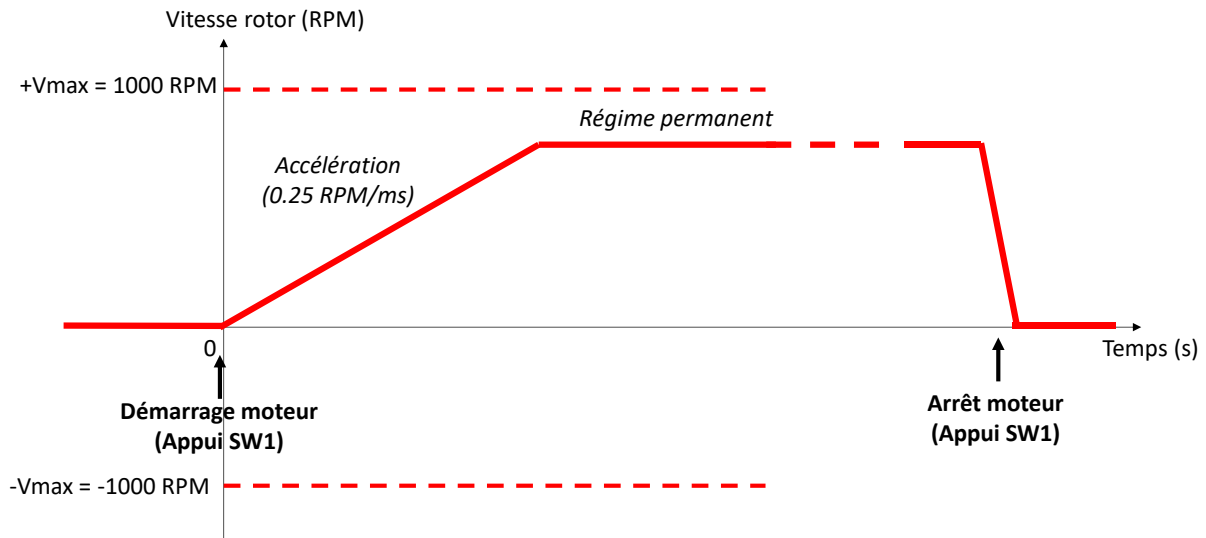
Le moteur de traction employé dans ce projet est un moteur synchrone à aimants permanents. Il s'agit du moteur TGN2-028-30-24. Il intègre un capteur de position basé sur un synchro-résolveur ou résolveur. Il est monté sur le banc présenté à l'annexe 1.

Dans la pratique, le moteur et l'onduleur sont alimentés par la batterie haute tension du véhicule. Dans ce démonstrateur, ces parties seront alimentées par une alimentation de puissance 18 V.

Le tableau ci-dessous résume les caractéristiques et les performances attendues pour ce démonstrateur.

Caractéristiques / performances attendues	Valeurs
Tension d'alimentation du MCU	3.3 V
Tension d'alimentation de l'onduleur	18 V
Fréquence CPU	200 MHz
Période PWM	120 μ s
Période de mise à jour de la commande en couple	120 μ s
Période de mise à jour de la commande en vitesse	1.2 ms
Vitesse maximale	+/- 1000 tours par minute (RPM)
Couple de charge maximale	0.26 N.m
Courant de phase maximal en régime permanent et en phase d'accélération	10 A
Type de correcteur utilisé pour la régulation de vitesse	Proportionnel-intégrateur (PI) : Coefficient proportionnel $K_P = 0.08$ Coefficient intégrateur $K_P = 0.01$
Type de correcteur utilisé pour la régulation de courant	Proportionnel-intégrateur (PI) Coefficient proportionnel $K_P = 0.05$ Coefficient intégrateur $K_P = 5$

Le moteur est mis en marche par un appui sur le bouton SW1 du kit de développement DEVKIT-MPC5744P, selon la consigne de vitesse décrite ci-dessous. Le prochain appui sur SW1 arrête le moteur immédiatement. Au prochain appui, le moteur doit redémarrer suivant la même consigne de vitesse. Une LED verte indique la mise en rotation du moteur et son bon fonctionnement.



2. Tâches à accomplir

Vous travaillerez par binôme. Votre travail se décompose en deux parties principales :

1. A partir d'un programme de commande U/f existant, comprendre l'architecture matérielle et logicielle de la commande (questions associées). Compléter la partie du programme liée à l'acquisition du courant dans les trois phases du moteur. Ajouter un filtre pour éliminer le bruit affectant la mesure du courant de phase. Valider la commande sur le démonstrateur et relever les performances.
2. Adapter le programme de commande U/f pour réaliser une commande vectorielle ou *Field-Oriented Control* (FOC) avec régulation de vitesse, la valider sur le démonstrateur et relever les performances

Les détails de chaque tâche et les livrables attendus sont décrits dans les prochaines parties. A l'issue du BE, vous rendrez un rapport concis, donnant pour les deux parties :

- les réponses aux questions de clarification du programme de commande U/F (première partie)
- les détails principaux de l'implémentation logicielle
- les résultats de mesure

a. Préalable (séance 1)

Téléchargez depuis le site de NXP et installez les logiciels suivants :

- S32 Design Studio for Power Architecture IDE (version 2.1, 2019) : environnement de développement pour les microcontrôleurs de la famille MPC Power Architecture, dont le microcontrôleur utilisé MPC5744P : <https://www.nxp.com/design/software/development-software/s32-design-studio-ide/s32-design-studio-for-power-architecture:S32DS-PA>
- Librairie AMMCLIB (Automotive Math and Motor Control Library) for MPC574xP : contient l'ensemble des fonctions de calcul dédiées au contrôle moteur, sélectionnez

Automotive Math and Motor Control Library Set for MPC574xP dans la liste :
https://www.nxp.com/design/automotive-software-and-tools/automotive-math-and-motor-control-library-ammclib:AMMCLIB?tab=Design_Tools_Tab

- FreeMaster 3.0 : pour la visualisation de l'évolution en temps réel des variables globales du programme embarqué sur le microcontrôleur :
https://www.nxp.com/design/software/development-software/freemaster-run-time-debugging-tool:FREEMASTER?tab=Design_Tools_Tab

b. Partie 1 – Analyse de commande U/f

L'objectif de cette première partie est multiple : se familiariser avec le matériel et les outils logiciels, comprendre la structure logicielle du programme de contrôle moteur qui sera réutilisé dans la partie 2 pour élaborer la commande FOC.

Le projet **PMSM_TGN002_cde_Uf_etudiant.zip** vous est fourni. Il permet une commande de type U/f du moteur, ainsi qu'une lecture en temps réel de la vitesse du moteur et des courants de phase (attention : la fonction assurant une partie de la mesure du courant de phase est à compléter). Une description haut niveau du programme est fourni dans le fichier **Software_Details_Uf_Command_2024.pdf**.

Pour prendre en main les différents environnements logiciels et outils, il vous est demandé de programmer le microcontrôleur, de réaliser un test d'abord en **mode simulation** (voir annexe 3) puis en **mode réel** sur la maquette et de relever les résultats suivants :

- Temps d'exécution des boucles dites *Fast Loop* et *Slow Loop* (pour cela, vous pourrez utiliser la pin appelée *User_Test*, située sur la broche PA14 du microcontrôleur et accessible sur la broche 6 du connecteur J6 de la carte MPC5744P)
- Erreur vitesse et stabilité aux vitesses suivantes : +/- 30 RPM, +/- 300 RPM
- Courant de phase maximal aux vitesses précédentes

Vous pourrez utiliser l'outil **Freemaster** pour visualiser l'évolution des paramètres de sortie du modèle moteur (courants de phase, angle et vitesse).

i. Analyse du programme

Pour vous aider à mieux comprendre le principe de la commande et la structure logicielle du programme, plusieurs questions vous sont posées. Vous indiquerez vos réponses dans un rapport dans lequel vous inclurez aussi les résultats de mesure.

Questions :

- Quelle est la fréquence de fonctionnement du CPU ? de l'ADC ? du module FlexPWM ? Quelle est la résolution temporelle de la PWM ?
- Quelles sont les rôles des boucles dites *fast-loop* et *slow-loop* du programme ? Quelles sont leurs périodes d'exécution ? Comment sont-elles déclenchées ?
- 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 ?
- 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 ?

- Les rapports cycliques des commandes PWM appliquées sur les trois phases sont-ils mis à jour simultanément ? Si oui, pourquoi et comment ?
- Où et comment les courants de phase sont-ils mesurés sur la carte gate driver MOTORGD ? Quelles sont les avantages et inconvénients de cette mesure ? Quels composants sont employés ? Quelle tension est lue par le microcontrôleur lorsqu'aucun courant de phase ne circule ? Complétez la fonction `Compensate_Iphase_MotorGD_FLT` (définie dans `Current_Sensor.c`) ainsi que les constantes associées à la mesure de courant de phase.
- Quel événement déclenche la lecture des courants ? Comment la synchronisation entre la lecture des courants et cet événement est-elle assurée ?
- Pourquoi deux ADC indépendants sont-ils utilisés pour assurer la lecture des courants de phase ?
- 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 ?
- Est-ce que les 3 courants de phase sont lus ? Pourquoi ?
- Quels sont les rôles des variables `DeadTime_PWM`, `Align_PWM` et `Delay_Trig_ADC_us` ?
- Quelle est la différence entre l'angle rotor mécanique et électrique ? Lequel intervient dans la transformation de Park ?

ii. Lecture du courant de phase

L'extraction du courant de phase à partir de la mesure de la tension aux bornes des résistances de shunt dans les bras de l'onduleur est assurée par la fonction `Compensate_Iphase_MotorGD_FLT` (définie dans `Current_Sensor.c`). Dans le programme `PMSM_TGN002_cde_Uf_etudiant` qui vous est fournie, cette fonction est vide et renvoie des courants de phase nuls.

Il vous est demandé de compléter cette fonction (les constantes requises pour l'opération de calcul du courant sont déjà définies dans le fichier `Current_Sensor.h`). Vous testerez cette fonction en mode réel.

iii. Mise en place d'un filtre sur la mesure de courant.

La mesure de courant de phase est bruitée. Ce bruit peut gêner voire déstabiliser la commande du moteur, dès que celle-ci se base sur le courant de phase (ce qui n'est pas le cas avec la mesure U/f, mais ce sera peut-être le cas avec la commande FOC).

Afin de réduire l'effet du bruit sur la mesure de courant, on propose d'ajouter un filtre numérique passe-bas à réponse impulsionnelle infinie (IIR). Sa fréquence de coupure sera fixée à 30 Hz. Proposez une solution et testez-la.

Remarque 1 : les prototypes de plusieurs fonctions ont déjà été prévu à cet effet, dans `Current_Sensor.c` :

- `void Perform_Current_Meas_Calibration(void);`
- `void Config_Filter_Current(void);`
- `void raz_Filter_Current(void);`

La variable `Iphase_filter` a déjà été déclarée pour stocker le résultat du filtrage du courant. Vous réutiliserez ces fonctions et cette variable dans la mise au point du filtre.

Remarque 2 : la validation du filtre pourra se faire dans le mode moteur virtuel. Dans `modele_moteur_virtuel.c`, un modèle de bruit sinusoïdal se superposant au courant de phase est prévu. Sa fréquence est fixée par la variable `FreqNoise`, son amplitude par la variable `AmplNoise`. Vous pourrez ainsi tester le bon fonctionnement du filtre dans la version virtuelle du banc moteur, avant de lancer un essai sur le moteur.

c. Partie 2 – Mise au point de la commande FOC

Cette deuxième partie est dédiée à la mise au point d'une commande FOC pour piloter le moteur. Cette mise au point se fera en repartant du programme de commande U/f. Une double régulation sera utilisée : une régulation en couple (ou courant) et une régulation en vitesse.

Le travail se fera en trois étapes vous aidant dans l'élaboration de la commande :

1. Proposez un schéma-bloc de la commande FOC et de la régulation en courant et en vitesse du moteur. Indiquez les fonctions de la librairie `AMMCLIB` qui seront utilisées et leurs éventuels paramètres.

2. Reprendre le programme de commande U/f pour l'adapter à une commande FOC. Pour valider le bon fonctionnement logiciel, vous commencerez à travailler avec un modèle virtuel du moteur (voir annexe 3). En effet, en travaillant directement sur le moteur et le gate driver lors du debug du programme de commande, toute erreur (mauvaise configuration des correcteurs, mauvais paramétrage de la PWM, lecture erronée de l'angle rotor ou des courants de phase, ...) peut conduire à une dégradation du gate driver en raison des tensions et courants excessifs qu'il risque de devoir supporter. Il est donc préférable de pouvoir debugger le programme sur microcontrôleur, mais sans le gate driver et les différents capteurs en utilisant un modèle électromécanique représentatif. Vous pourrez utiliser `Freemaster` pour visualiser l'évolution des paramètres de sortie du modèle moteur (courants de phase, angle et vitesse).








3. Réaliser des essais sur le moteur et relever les performances.

Résultats à relever :

- Temps d'exécution des boucles dites *Fast Loop* et *Slow Loop* (pour cela, vous pourrez utiliser la pin appelée *User_Test*, située sur la broche PA14 du microcontrôleur et accessible sur la broche 6 du connecteur J6 de la carte MPC5744P)
- Erreur vitesse et stabilité aux vitesses suivantes : +/- 30 RPM, +/- 300 RPM
- Erreur sur I_d et I_q pour les vitesses précédentes
- Courant de phase maximal aux vitesses précédentes et comparaison avec ceux obtenus avec la commande U/f

IV - Liste des documents disponibles en ligne

Sur Moodle et sur la page web <http://www.alexandre-boyer.fr/enseignements.htm>, un ensemble de documents sont mis à votre disposition. Ceux-ci sont disponibles dans le fichier BE_electronique_automobile_documents.zip, organisé de la manière suivante :

-  Enoncé du bureau d'étude
-  Présentation du bureau d'étude
-  Documents de prise en main du microcontrôleur + logiciels
-  Datasheets
-  Projet Commande U/f
-  Modèles Simulink (moteur et commandes)
-  Informations moteurs du banc

- enonce BE automobile 5SE_2023-2024.pdf : l'énoncé du bureau d'étude
- Documents de prise en main :
 - Presentation_of_MPC5744P_2023-24.pdf : un document d'aide à la programmation du microcontrôleur MPC5744P
 - Introduction_softwares_Automotive_Electronics_2023-24.pdf : un document de prise en main des différents outils logiciels utilisés durant le BE
- Datasheets :
 - Microcontrôleur:
 - MPC5744PRM.pdf : Reference manual du MPC5744P
 - DEVKIT-MPC5744P-QSG.pdf : guide utilisateur de la carte DEVKIT-MPC5744P
 - DEVKIT-MPC5744P Schematic_RevB (SCH-29333).pdf : schématique de la carte DEVKIT-MPC5744P
 - Gate driver :
 - MC34GD3000.pdf : datasheet du gate driver MC34GD3000
 - DEVKIT-MOTORGD_QSG.pdf : guide utilisateur de la carte DEVKIT-MOTORGD
 - SPF-29379_B2.pdf : schématique de la carte DEVKIT-MOTORGD
 - Interface résolveur :
 - pga411-q1.pdf : datasheet de l'interface résolveur PGA411-Q1
 - slau658-EVM user guide.pdf : guide utilisateur de la carte PGA411Q1-EVM
- Projet S32 de la commande U/f du moteur TGN2-0028-30-24 : PMSM_TGN002_cde_Uf_etudiant.zip, et description associée : Software_Details_Uf_Command_2023.pdf
- Informations sur les moteurs :
 - TGN2-0028-30-24.xlsx : caractéristiques du moteur synchrone TG Drives TGN2-0028-30-24
 - EN_FT_89890011.pdf : caractéristiques du moteur à courant continu Crouzet 89 890 011

Reportez-vous au cours de Moteurs et commandes pour le véhicule électrique, pour clarifier la compréhension du fonctionnement du moteurs synchrone à aimants permanents, sa modélisation et sa commande.

Pour une compréhension de la mise en œuvre pratique d'une commande vectorielle pour moteur synchrone à aimants permanents, nous vous conseillons le tutoriel "3-Phase PMSM Control Workshop with NXP's Model-Based Design" proposé par NXP. Il est disponible à

l'adresse suivante : <https://community.nxp.com/thread/464336>. Même si le microcontrôleur utilisé est différent du MPC5744P, ce tutoriel vous aidera à réfléchir à l'architecture générale du module Main Inverter et à identifier les "astuces" matérielles permettant de mettre en œuvre la commande vectorielle.

V - Annexe 1 – Présentation du matériel

Dans ce BE, nous disposons de plusieurs maquettes, de composants dédiés automobile fournis par la société NXP Semiconductor, et de cartes d'interface. Les notes d'application des différents composants vous sont fournies.

L'ensemble des documents sont disponibles sur Moodle ou sur le site <http://www.alexandre-boyer.fr>.

1. Le banc moteur

Le banc moteur utilisé dans ce bureau d'étude est présenté sur la Figure 5. Il intègre deux moteurs, dont les arbres sont accouplés :

- le moteur synchrone à aimants permanents TG Drives TGN2-0028-30-24
- le moteur à courant continu Crouzet 89 890 011

Ces deux moteurs sont conçus pour être alimentés sous 24 Vdc nominal et pour des vitesses nominales de 3000 RPM. Dans ce BE, nous nous limiterons à 18 Vdc et 1000 RPM.

Dans ce bureau d'étude, la machine synchrone est utilisée en motrice. La machine à courant continu est utilisée comme charge réglable (couple résistant) en fonction de la résistance qui est connectée à ses bornes. Pour des raisons de test, il est tout à fait possible de modifier le sens de fonctionnement de ces moteurs : utiliser la machine à courant continu comme moteur et la machine synchrone comme génératrice.

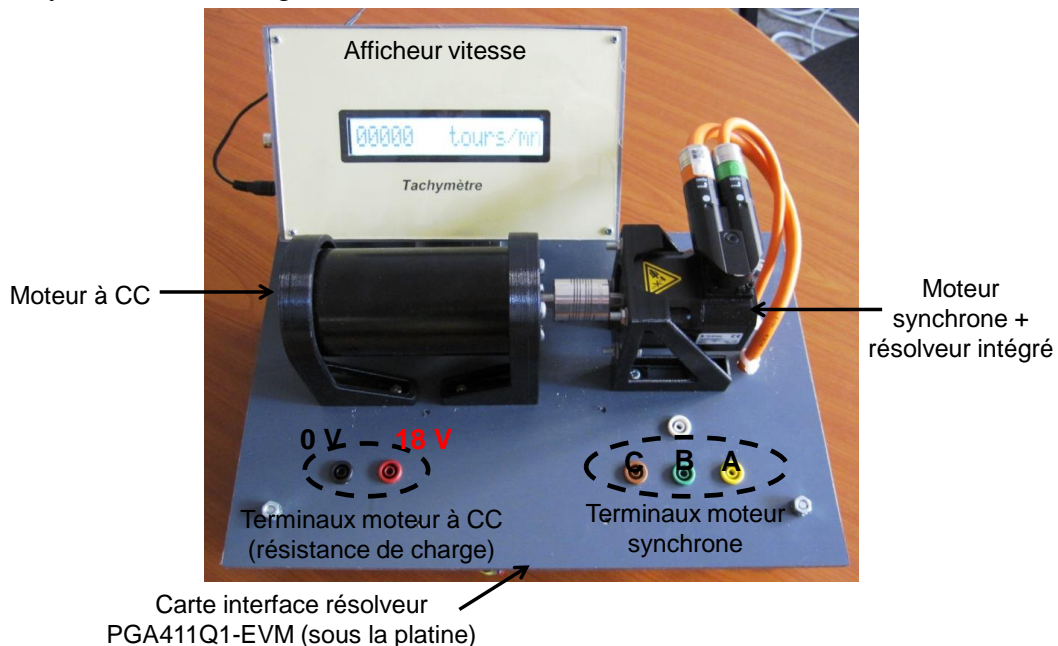


Figure 5 - Banc moteur utilisé dans le bureau d'étude

Un capteur de position et de vitesse du rotor est intégré dans le moteur TGN2-0028-30-24. Il s'agit d'un capteur de type synchro-résolveur, nécessitant un circuit d'interface pour l'exciter et mesurer les signaux induits sur les bobines cos et sin. Le circuit employé est le PGA411-Q1 développé par Texas Instruments (voir Annexe 1 - partie 4). Le kit d'évaluation PGA411-Q1 EVM est utilisé. Les terminaux du résolveur sont connectés sur cette carte. Par défaut, elle est montée sous le banc moteur. Le circuit d'interface communique avec un module afficheur LCD pour afficher la vitesse de rotation du moteur. Celui-ci est alimenté sous une tension continue de 5 V.

Durant ce bureau d'étude, la carte PGA411-Q1 EVM sera déconnectée du module LCD car le PGA411 fera partie intégrante du système de commande du moteur synchrone et communiquera avec le microcontrôleur MPC5744P.

Les caractéristiques détaillées des moteurs sont disponibles dans les documents suivants :

- TGN2-0028-30-24.xlsx pour le moteur synchrone
- EN_FT_89890011.pdf pour le moteur à courant continu

Afin de simuler le comportement du moteur pour un type de commande donné, un modèle Simulink du moteur vous est fourni à l'annexe 3.

2. Carte microcontrôleur MPC5744P (Panther)

La kit DEVKIT-MPC5744P est une carte d'évaluation du microcontrôleur 32 bits MPC5744P (Panther), conçu par NXP Semiconductor. Ce microcontrôleur double cœur *lockstep* est dédié aux applications de contrôle moteur pour l'automobile. Ce microcontrôleur a été conçu pour répondre aux exigences *safety* posées par le standard ISO26262 avec un niveau ASIL D. Une description de la structure du microcontrôleur et de ses périphériques est donnée dans *Presentation_of_MPC5744P_2019-20.pdf*. La carte est présentée sous la forme d'un shield Arduino. Elle est conçue pour s'interfacer directement avec la carte d'évaluation MotorGD, présentée dans la prochaine partie.

La Figure 6 présente la carte et les principaux connecteurs et jumpers.

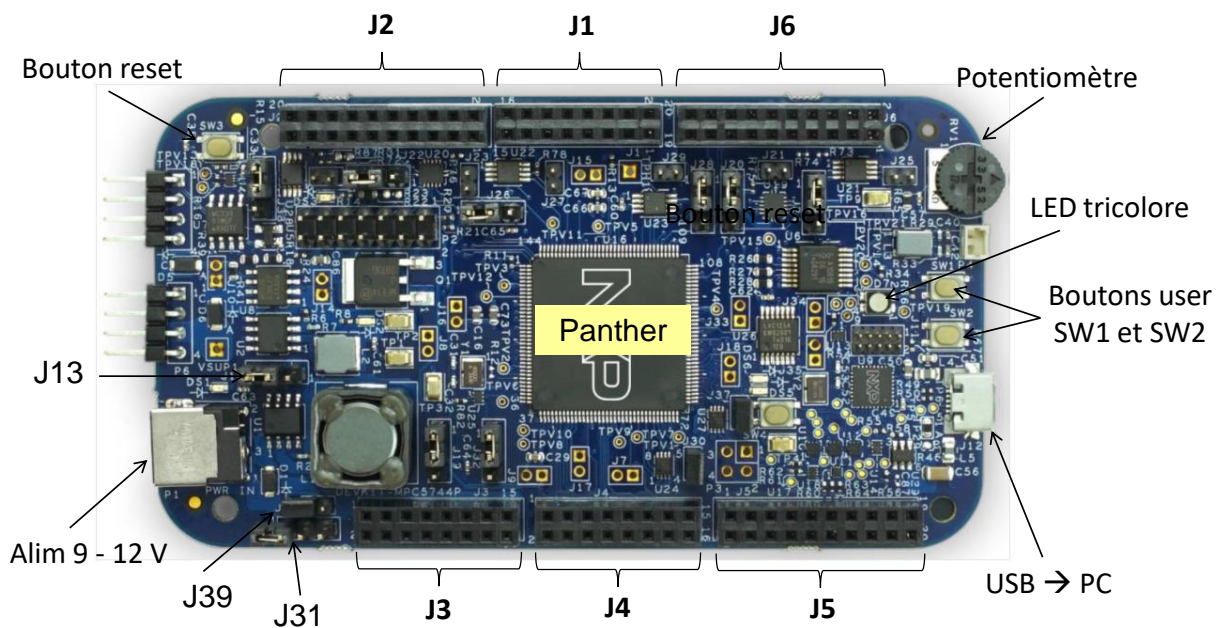


Figure 6 - Carte microcontrôleur MPC5744P

Le microcontrôleur fonctionne sous 3.3 V. La carte peut être alimentée de deux manières différentes, selon la position du jumper J13 :

- à travers le connecteur USB, sous 5 V (position 2-3)
- à travers le jack sous 9-12 V (position 1-2)

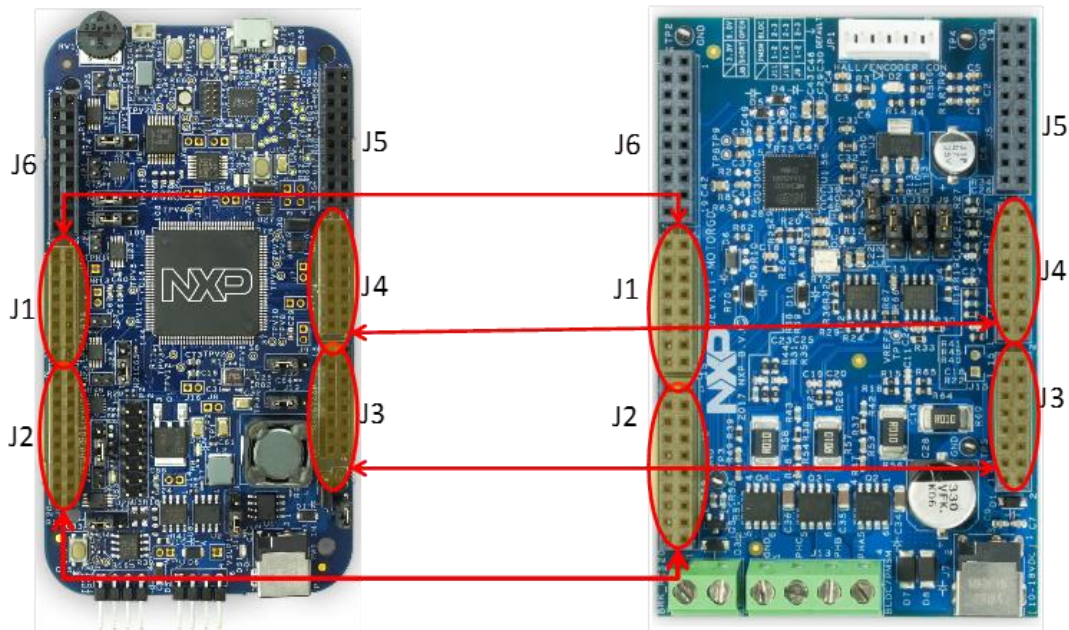
La carte MPC5744P peut fournir une alimentation basse tension (EXT_LV_PWR) à la carte MotorGD à travers la broche J3-3. Selon la position du jumper J31, cette tension est de 3.3 V

La carte présente aussi les connexions vers les trois phases d'un moteur triphasé (notées PhA, PhB et PhC) à travers le bornier à vis J13.

Selon les positions des jumpers J9, J10 et J11, soit les mesures de courant de phase (position 1-2), soit les tensions de phase (position 2-3) sont disponibles sur les sorties de la carte pour une acquisition analogique-numérique par le microcontrôleur de la carte MPC5744P.

Les informations sur la carte, notamment sa schématique et les valeurs des composants passifs montés, sont disponibles dans le document SPF-29379_B2.pdf.

Une assignation des broches reliant la carte MPC5744P et la carte MotorGD a été établie. Elle est décrite ci-dessous.



MPC5744P		MOTOR GD	
Pin connecteur	I/O - fonction	Pin connecteur	I/O - fonction
J1-2	PA3	J1-1	GD_EN
J1-4	PA4	J1-3	GD_RST
J1-5	PA11 (PWM0_0_chA)	J1-5	PWMA_HS
J1-7	PA10 (PWM0_0_chB)	J1-7	PWMA_LS
J1-9	PF0 (PWM0_1_chA)	J1-9	PWMB_HS
J1-11	PD14(PWM0_1_chB)	J1-11	PWMB_LS
J1-13	PA12 (PWM0_2_chA)	J1-13	PWMC_HS
J1-15	PA13(PWM0_2_chB)	J1-15	PWMC_LS
J1-8	PA6 (eTimer2-ch2)	J9-3	BEMF_A
J1-10	PA7 (eTimer2-ch3)	J10-3	BEMF_B
J1-12	PA8 (eTimer2-ch4)	J11-3	BEMF_C

MPC5744P		MOTOR GD	
Pin connecteur	I/O - fonction	Pin connecteur	I/O - fonction
J2-5	PC4 - CS0	J2-5	SPI_CS
J2-7	PC6 – Sout	J2-7	SPI_MOSI
J2-9	PC7 – Sin	J2-9	SPI_MISO
J2-11	PC5 – SCLK	J2-11	SPI_SCLK
J2-13	GND	J2-13	GND
J2-18	PA14	J2-17	OC_OUT
J2-19	PA15	J2-19	GD_INT

MPC5744P		MOTOR GD	
Pin connecteur	I/O - fonction	Pin connecteur	I/O - fonction
J4-1	PB8 – ADC0-AN1	J4-1	Courant Vss
J4-3	PB12 – ADC0/1-AN14	J4-3	Tension Vbat
J4-5	PB9 – ADC0/1-AN11	J4-5	Courant Ph A
J4-7	PB10 – ADC0/1-AN12	J4-7	Courant Ph B
J4-9	PB11 – ADC0/1-AN13	J4-9	Courant Ph C

MPC5744P		MOTOR GD	
Pin connecteur	I/O - fonction	Pin connecteur	I/O - fonction
J3-3	Ext_Lv_Pwr (3.3 V)	J3-3	MCU-VCC (3.3 V)
J3-11	GND	J3-11	GND
J3-13	GND	J3-13	GND

Figure 8 - Assignation des broches d'interconnexion entre les cartes MPC5744P et MotorGD

4. Carte interface résolveur PGA411Q1-EVM

L'interface résolveur est le PGA411-Q1 développé par Texas Instruments. Il s'agit d'une interface résolveur complètement intégrée, dédiée aux applications automobiles exigeant un niveau de sûreté de fonctionnement ASIL D.

Le PGA411-Q1 est une interface de résolveur intégrant un amplificateur d'excitation sinusoïdale du résolveur de fréquence et d'amplitude réglables, ainsi que l'électronique d'acquisition de l'angle et de la vitesse du rotor à partir des signaux cos et sin délivrés par le résolveur. La résolution (sur 10 bits ou 12 bits) de la mesure d'angle est de 0.35° ou 0.09°, celle sur la vitesse de 572 RPM ou 36 RPM. Le composant est alimenté sous 5 V. Un convertisseur DC-DC boost interne génère l'alimentation nécessaire au fonctionnement de l'amplificateur d'excitation. L'ensemble des blocs internes du PGA411-Q1 intègre un diagnostic interne afin de détecter des conditions de fautes. Les fautes sont indiquées par une broche dédiée (Fault) et l'origine du problème peut être diagnostiquée par bus SPI.

Le composant est monté sur la carte d'évaluation PGA411Q1-EVM. La Figure 9 décrit la carte avec les principaux connecteurs et jumpers.

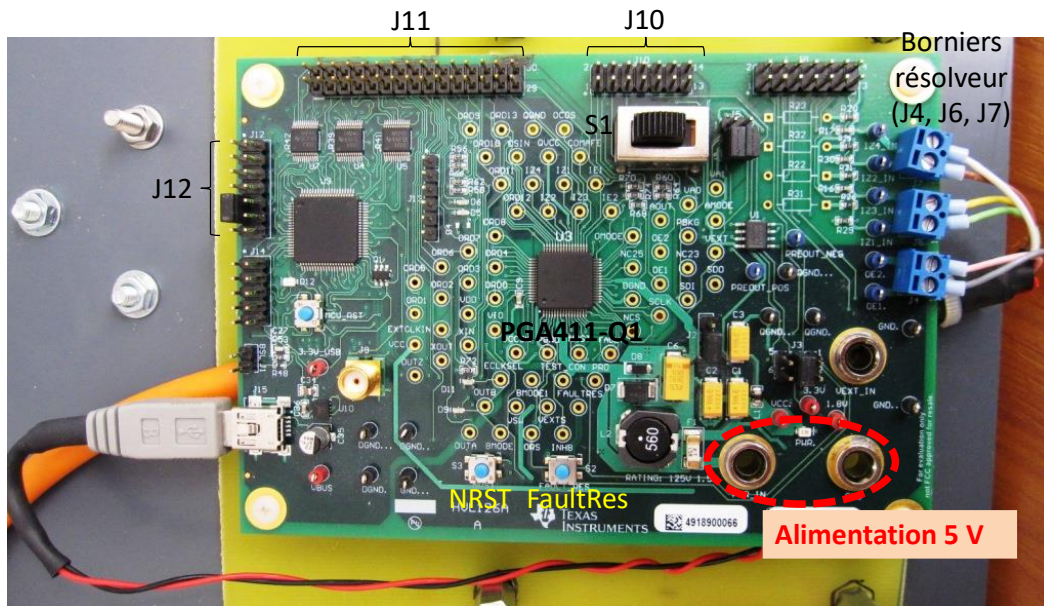


Figure 9 - Carte PGA411Q1-EVM

Cette carte est alimentée sous 5 V à travers les fiches bananes PWR_IN et GND. Les borniers à vis J4, J6 et J7 sont dédiés respectivement à la connexion des terminaux d'excitation, cos et sin du résolveur. Lors de la première prise en main de ce composant, vérifiez à l'oscilloscope que les signaux sur ces différents terminaux respectent les limites en tension recommandées par Texas Instruments.

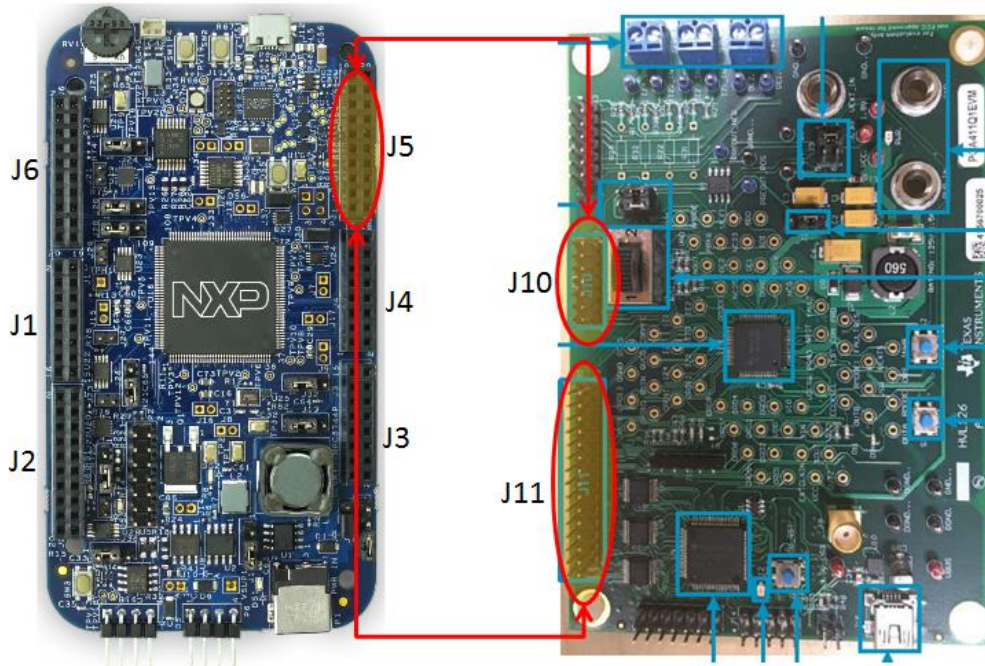
Bien que le PGA411-Q1 soit alimenté sous 5 V, il peut être interfacé avec des composants alimentés sous 3.3 V. Pour garantir une conversion 5 V - 3.3 V, vérifiez que le jumper J3 soit placé dans la position 3-4. Vérifiez que le jumper J5 soit dans les positions 3-5 et 4-6 pour assurer la lecture des sorties de l'amplificateur d'excitation par l'étage de réception du PGA411-Q1. Vérifiez aussi que le switch S1 est placé dans la position 1, pour éviter une atténuation importante des signaux délivrés par les terminaux cos et sin du résolveur.

Les boutons NRST et FaultRes permettent respectivement de mettre en reset le PGA411-Q1 et de remettre à zéro les conditions de fautes.

La carte intègre un microcontrôleur MSP430 de Texas Instruments, permettant à la carte d'être évaluée à l'aide d'une application logicielle propriétaire. Cette partie de la carte est alimentée via le connecteur USB J15. Dans ce BE, cet outil ne sera pas utilisé. Pour éviter tout conflit lors de la connexion avec la carte MPC5744P, vérifiez qu'une alimentation de 5 V est fournie par le connecteur USB J15 et que la broche MCU_RST (broche J12-11) est tirée à la masse pour inhiber le MSP430.

Les informations sur la carte, notamment sa schématique, sont disponibles dans le document slau658-EVM user guide.pdf

La carte PGA411Q1-EVM est normalement intégrée au banc moteur et connectée à l'écran LCD du banc pour afficher la vitesse de rotation de l'arbre moteur. Pour les besoins du BE, la carte PGA411Q1-EVM sera sortie du banc moteur et déconnectée de l'écran LCD (celui-ci ne pourra donc plus afficher la vitesse). Elle sera connectée à la carte MPC5744P, selon l'assignation décrite à la Figure 10.



MPC5744P		PGA411Q1-EVM	
Pin connecteur	I/O - fonction	Pin connecteur	I/O - fonction
J5-3	PA1 - DSPI2_SOUT	J11-12	SDI
J5-5	PA2 - DSPI2_SIN	J11-14	SDO
J5-19	PG10 – DSPI2_CS3	J11-8	NCS
J5-1	PA0 – DSPI2_SCK	J11-10	SCLK
J5-12	GND	J10-14	GND
J5-7	PG8	J11-22	Fault
J5-8	PD8	J11-20	FaultRes
J5-13	PG9	J11-30	BMODE0
J5-16	PE4	J10-10	NRST
J5-15	PE2	J11-26	AMODE
J5-18	PC2	J11-28	OMODE
J5-17	PE5	J11-27	VA0
J5-20	PB14	J11-29	VA1

Figure 10 - Assignment des broches d'interconnexion entre les cartes MPC5744P et PGA411Q1-EVM

VI - Annexe 2 - Les logiciels de conception utilisés

Durant ce bureau d'étude, plusieurs logiciels seront utilisés pour ajuster les paramètres de la commande moteur, développer le code embarqué, programmer le microcontrôleur, le debugger et le tester sur cible matérielle. Les outils logiciels suivants seront utilisés :

- S32 Design Studio (S32DS) for Power Architecture : environnement intégré de développement et de debug in-situ pour MPC5744P
- Freemaster : monitoring en temps réel de l'application embarquée dans le MPC5744P

- Librairie AMMCLIB : elle intègre l'ensemble des fonctions mathématiques basiques et avancées pour le contrôle moteur. AMMCLIB fonctionne à la fois sur S32DS et Simulink.

Reportez-vous au document [Introduction_softwares_Automotive_Electronics_2020-21.pdf](#) pour plus d'informations sur l'utilisation de ces outils

VII - Annexe 3 – Modèle simplifié (en C) du moteur

Ce modèle, écrit en C et pour des essais sur microcontrôleur, est dédié à la validation du logiciel embarqué associé à votre commande. Le modèle proposé est très simplifié et n'a pas vocation à simuler exactement la réponse du PMSM utilisée. Il permet de faire tourner votre code sur le microcontrôleur, sans avoir besoin de connecter la carte gate driver et le moteur. Vous pourrez ainsi debugger votre commande sans risquer de dégradation de la carte gate driver.

Ce modèle a vocation à fournir au programme gérant la commande du moteur une information sur l'angle et la vitesse du rotor, ainsi que sur les courants de phase, qu'aurait un modèle de moteur auquel on appliquerait la commande générée par le programme. En raison de la non linéarité du modèle électromécanique du moteur PMSM et de sa charge, un modèle électromécanique simplifié est proposé pour limiter le temps de calcul requis par ce modèle. Cependant, malgré sa précision limitée, ce modèle reproduira un comportement représentatif du moteur. Il vous aidera à valider le code C associé à votre commande.

Ce modèle a été conçu pour que son utilisation se fasse « sans efforts », c'est-à-dire sans modification de votre programme. L'ensemble des constantes et des fonctions de ce modèle sont présentes dans les fichiers `modele_moteur_virtuel.h/c`, que vous pourrez inclure à votre projet. Pour utiliser le modèle moteur dans votre programme, il suffira de réaliser les opérations suivantes :

Dans la fonction `main`, lors de l'initialisation et l'appel de la fonction `init_Cmd_Moteur_Variables` :

- Si l'argument d'entrée = `MOTEUR_REEL`, le programme doit être utilisé sur le banc moteur réel. Avec un commande FOC, les mesures de courants, d'angle et de vitesse rotor sont utilisées.
- Si l'argument d'entrée = `MOTEUR_VIRTUEL`, le programme doit être utilisé sans le banc moteur (microcontrôleur seul). Le programme se base sur un modèle de simulation simplifié du moteur pour déterminer les courants de phase, l'angle et la vitesse du rotor.