

# Adaptation des méthodes de l'IOT pour le design des convertisseurs de puissance reprogrammables

Mathilde LONGUET<sup>1</sup>, Jean ALINEI<sup>1</sup>, Luiz F.L. VILLA<sup>1,2</sup>

<sup>1</sup>LAAS-CNRS, <sup>2</sup>Université Paul Sabatier, Toulouse

**RESUME** - Cet article a pour objectif d'analyser la méthodologie de design des plateformes IOT ayant permis son essor et de l'appliquer aux besoins de l'électronique de puissance.

**Mots-clés**—*Méthodologie de design, électronique de puissance, open source, modularité, versatilité*

## 1. INTRODUCTION

Le marché de l'électronique de puissance est fragmenté par industrie, par applicatif et par échelle de puissance. Cette fragmentation rend l'approche classique de design en électronique de puissance *function-oriented*, où chaque solution spécifique à un usage précis [1]. De plus, ces solutions sont présentées sous forme de « boîtes noires » ce qui rend leur compréhension et donc leur modification pour un cas d'usage différent - même légèrement - difficile voire impossible. Cette approche rend le domaine de l'électronique de puissance hermétique et chronophage, ce qui implique un ralentissement des innovations dans ce secteur. Rendre accessible ce domaine technique semble pourtant un enjeu primordial pour l'électrification de nos sociétés et donc pour la transition énergétique [2].

La communauté scientifique d'électronique de puissance a essayé de casser la barrière de complexité inhérente à sa multi-disciplinarité par des techniques telle que le *power electronics building blocks* (PEBB) [3] et le *power converter arrays* (PCA) [4]. A chaque fois, une technique d'un autre domaine scientifique a été appliquée à l'électronique de puissance pour la rendre moins complexe. Pour le PEBB, c'est la modularité des carte mères d'ordinateurs, pour le PCA c'est l'intégration matérielle des transistors en micro-électronique. Cet article présente une nouvelle méthode, basée sur l'intégration ergonomique hardware/software utilisée dans l'IoT.

Le domaine de l'IoT et des micro-contrôleurs a subi un changement radical dans son accessibilité cette dernière décennie. En effet, une simplification de son usage a permis le déploiement de nombreuses nouvelles solutions et une augmentation significative des usagers de la technologie. L'objectif de cette étude est d'étudier des méthodes de design de l'IoT et les transposer dans l'électronique de puissance.

Ce papier s'organise comme suit. La section 2 fait un rappel des besoins en électronique de puissance. La section 3 décrit les techniques mobilisées par l'IoT. La section 4 fait une analyse de l'application de ses techniques en électronique de puissance et la section 5 propose une étude de cas.

## 2. LE BESOIN INTÉGRÉ EN ÉLECTRONIQUE DE PUISSANCE

Un convertisseur de puissance est constitué d'une structure hardware et software. D'un point de vue hardware, tous les convertisseurs de puissance sont basés sur une même suite de 5 sous-modules, expliqués par la Figure 1 et détaillés ci-dessous.

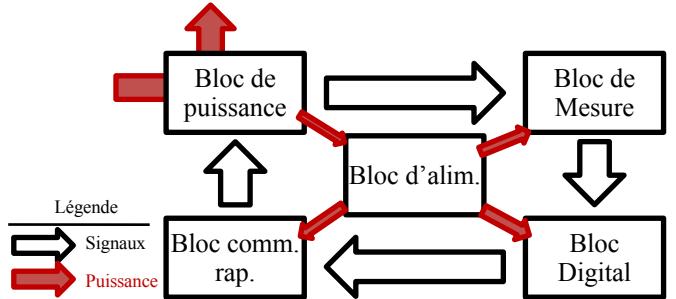


Figure 1 - Les 5 blocs d'un convertisseur de puissance

- **Bloc de puissance** : permet de faire circuler l'énergie entre entrées/sorties par une structure active et un filtre associé ;
- **Bloc de mesure** : embarque différents instruments de mesure permettant d'observer les entrées/sorties ;
- **Bloc de commande rapprochée** : adapte les signaux du microcontrôleur et gère la commutation des éléments actifs qui contrôlent le flux électrique ;
- **Bloc d'alimentation** : récupère une partie de l'énergie et l'adapte pour faire fonctionner l'ensemble du système ;
- **Bloc digital** : contient les périphériques nécessaires au contrôle des flux de puissance en temps réel, la communication et une *stack* logicielle détaillée ci-dessous.

D'un point de vue logiciel, un *stack* classique d'informatique embarquée utilisé en électronique de puissance, avec les temps correspondant aux différentes tâches, est représenté en Figure 2 et détaillé ci-dessous.

- **Tâche de communication** : Gère les échanges de données asynchrones avec l'extérieur. Elle opère typiquement à l'échelle de la milliseconde ;
- **Tâche applicative** : Déploie l'applicatif auquel est lié le convertisseur comme la charge de batterie ou MPPT. Cela se traduit en pratique typiquement par une machine à état. Cette couche opère à la centaine de microsecondes ;
- **Tâche de commande** : Déploie la boucle de régulation rapide qui assure le suivi de consigne par exemple. C'est la tâche qui récupère des données issue des ADC. Cette couche opère à la dizaine de microsecondes ;
- **Périphériques** : Couche ultra rapide où se trouvent les ADC, la génération des PWM entre autres. Elle opère à la centaine de nanosecondes ;

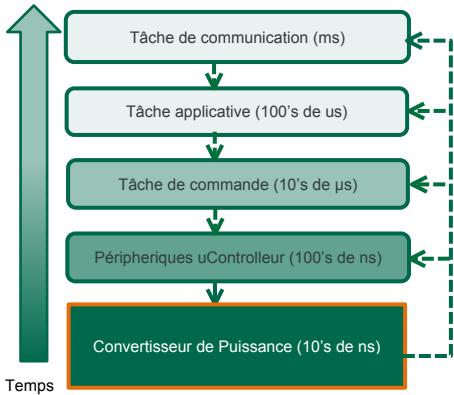


Figure 2 - Stack logiciel en électronique de puissance

Chacun de ces blocs nécessite un développement spécifique, dans l'électronique, l'informatique industrielle et l'automatique. Cet embriement thématique est à l'origine de la *barrière de complexité* de l'électronique de puissance.

### 3. LA MÉTHODOLOGIE IOT

Le domaine de l'IoT a été transformé par des acteurs ayant cassé leur barrières de complexité respectives. Cela a été fait par la mise en place des designs spécifiques, basé sur une forte intégration hardware/software, afin de démocratiser et simplifier l'accès à la micro-électronique. Le Tableau 1 donne une analyse des trois designs mis en place par ces acteurs.

Tableau 1. Techniques de design de l'IoT

Cible	Technique d'intégration	
	Hardware	Software
Arduino	Petite carte avec un accès simplifié aux broches du microcontrôleur par connecteur dédié	Suite logicielle intégrée sur un éditeur qui gère aussi les bibliothèques des tiers (capteurs)
Feather	Carte avec un facteur de forme encore plus réduit	Bibliothèques intégrés à la suite Arduino
Raspberry pi	Connecteurs standards en bord de carte	OS embarqué avec couche d'abstraction

A partir de cette analyse, les fonctions nécessaires à la simplification en IoT sont :

- Création des cartes électroniques avec une connectique qui permet l'ajout facile des modules supplémentaires (*shields*)
- Couches d'abstraction logicielles permettant de réduire le nombre de paramètres à saisir par l'utilisateur ;
- La gestion de l'intégration des produits tiers à travers une interface de développement logiciel intégrée ;
- Mise à disposition de la totalité de la technologie à travers des licences libres, ce qui permet aux utilisateurs de l'intégrer dans leur designs.

### 4. LA MÉTHODOLOGIE DE DIMENSIONNEMENT PROPOSÉE

Une transposition de la méthodologie mise en place dans le domaine de l'IoT à l'électronique de puissance pourrait être décrite par les actions suivantes :

- Simplifier la présentation de l'électronique de puissance en une topologie par blocs (entre mesures, contrôle, convertisseur, alimentation et entrées/sorties de puissance) ;

- Donner accès aux broches du microcontrôleur utilisé par le convertisseur;
- Définir l'accès des entrées/sorties de puissance et de communication à travers des connecteurs standards ;
- Intégrer et abstraire le logiciel permettant l'utilisation de la carte. Cela passe par une pré-configuration, ainsi qu'une mise en place d'API. L'objectif est de simplifier l'écriture des paramétrages de puissance;
- Mettre à disposition la totalité du design du convertisseur à travers des licences open-source (ouverture de la carte, diffusion des Bill of Materials, ...)

Le Tableau 2 décline l'application de ces propositions sur les blocs de puissance définis en partie 2.

Tableau 2. Blocs d'un convertisseur et leurs fonctions associées

Sous Bloc	Méthode proposée
Puissance	Associer le nombre de fonctionnalités de puissance à la définition de la plage de puissance, tension, courant lors du choix d'une topologie
Mesures	Définir une plage de mesure appropriée pour toutes les fonctionnalités couvertes par la puissance définie, définir la précision nécessaire pour le pire cas, définir la bande passante des capteurs
Micro-contrôleur	Choisir la cible qui intègre tous les cas de commandes associés aux fonctionnalités de puissance choisies, choisir une cible avec un périphérique de PWM à haute précision, choisir une cible avec un ADC intégré avec la PWM, choisir une cible avec des fonctionnalités de communication
Driver	Définir les besoins d'isolation, définir les besoins d'interconnexion, définir la technique de driver associée, définir la fréquence maximale d'opération
Alimentation	Définir les plages de tension, définir les besoins de filtrage, définir les plages de puissance à plein calcul

Ce modèle permet de définir les paramètres à mettre en avant pour les utilisations de puissance. Les autres informations nécessaires à la configuration du matériel devront être abstraites, afin que l'utilisateur aient seulement accès aux informations nécessaires aux usages de la puissance.

Le Tableau 3 décline l'application des tâches définies dans la partie 2 sur la couche logiciel du convertisseur.

Tableau 3. Blocs d'un convertisseur et leurs fonctions associées

Tâche	Méthode proposée
Communication	Définir quelle type de modularité est attendue pour les modules de puissance et choisir un SoC qui supporte plusieurs chaînes de communication
Applicative	Définir les types de cas applicatifs de la cible de puissance et estimer les contraintes de calcul
Commande	Définir la flexibilité nécessaire à la commande bas niveau et estimer le besoin en terme de périphériques
Périphérique micro-contrôleur	Choisir une cible qui intègre l'ensemble des contraintes et qui est compatible avec un <i>Real Time Operating System</i> adapté

Pour l'intégration logiciel, la solution adoptée est d'utiliser *real-time operating system (RTOS)*. Cette solution permet d'ordonner les tâches et de créer une abstraction des périphériques, de l'architecture des tâches et de la communication.

### 5. ÉTUDES DE CAS

Le LAAS-CNRS développe un convertisseur de puissance à partir de cette analyse d'intégration logicielle et matérielle de l'IoT. Ce convertisseur constituera l'étude de cas de cet article.

L'étude de cas sera divisée en une analyse de l'intégration *Hardware* et *Software* du convertisseur de puissance développé. Les choix faits seront analysés à la lumière de la méthodologie expliquée ci-dessus et les résultats analysés dans une session de discussion.

### 5.1. Technique d'intégration Hardware

Le processus de définition du cahier de charge matériel se base sur les 5 blocs typiques de l'électronique de puissance illustrés dans la Figure 1. Le cahier des charges du convertisseur est donné au Tableau 4.

Tableau 4. Blocs génériques associés à la carte TWIST

Sous Bloc	Cahier de charges	Descriptif
Puissance	Buck synchrone, 300W, 200kHz, 12-100V, 8A, non-isolé, reversible en courant	La topologie choisie permet Buck, Boost, Buck-Boost et Onduleur mono-phasé.
Mesures	Isolées, tension et courant sur tout les voies de puissance, bande passante du kHz, bande passante de 1MHz pour des courants sur les inductances	La flexibilité applicative choisie nécessite une visibilité totale des grandeurs électriques. Le pilotage en peak current control nécessite des capteurs sur le chemin du courant d'inductance avec une forte bande passante.
Digital - Micro-contrôleur	STM32G474RE, mono-coeur, 11 sorties PWM avec une résolution de 168ps, intégration ADC-PWM, 24 canaux ADC 12 bits, 3 canaux DAC 12 bits, accélération matérielle de calcul, FMAC, CORDIC, communication multiples	La cible choisie est adaptée pour la commande machine et propose une forte intégration entre ses périphériques.
Digital - Communication	Multi-bus, CAN-Bus, RS485, Analogique, Synchronisation	La communication multi-bus choisie permet la mise en parallèle de plusieurs cartes pour exploiter leur modularité.
Driver	Bootstrap isolé avec gestion de temps mort intégrée	Le driver intègre l'isolation galvanique et permet une redondance du temps mort
Alimentation	Flyback, 700kHz, 10-120V, 500mA, connecté au côté haut du buck synchrone	Une solution flexible qui permet le blackstart du convertisseur en buck ou en boost

La Figure 3 montre la carte étudiée avec les différents blocs.

La topologie de puissance choisie est un double Buck synchrone illustré en Figure 4. Les deux bras permettent à la fois un fonctionnement en buck et boost, entrelacé ou indépendant. Ils peuvent aussi être combinés pour faire un onduleur mono-phasé.

De plus, l'intégration matérielle finale a pris en compte des aspects de facilité d'accès aux bords des cartes aux voies de puissance et de communication. Cette facilité d'accès est aussi mises en place dans les méthodologies IoT.

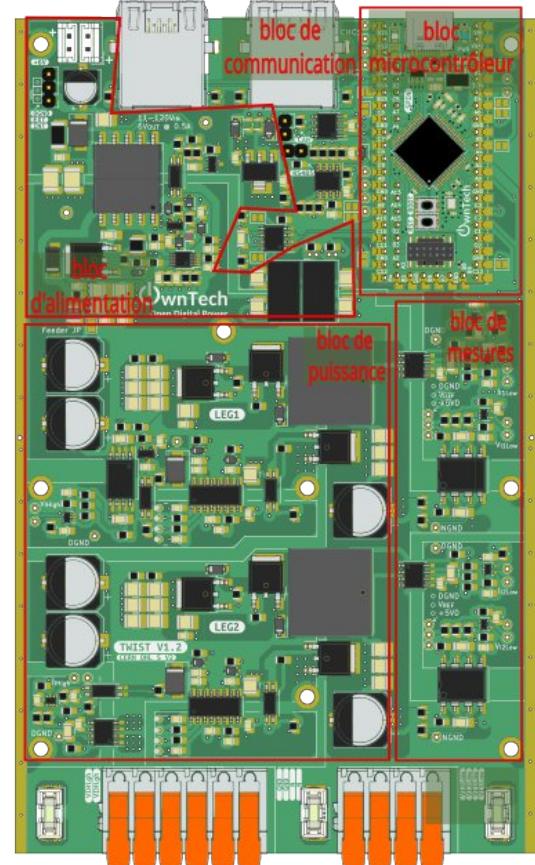


Figure 3 – L'architecture en bloc du convertisseur étudié

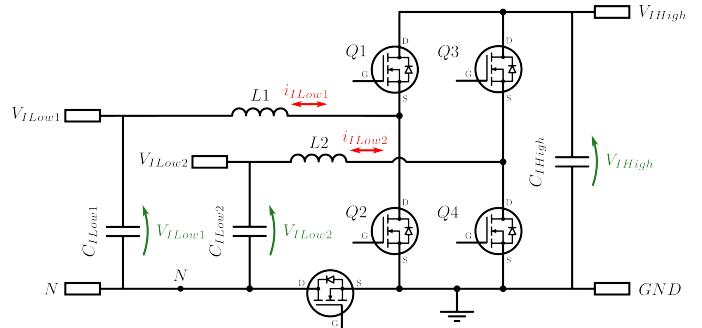


Figure 4 - La topologie choisie : Un double buck synchrone

L'intégration matérielle de la partie digital a été découpé en deux parties. Le micro-contrôleur a été intégré dans une carte dédiée, illustrée dans la Figure 6. Cela répond au besoin de création des connectiques standards qui permet sa ré-utilisation avec d'autres modules de puissance.

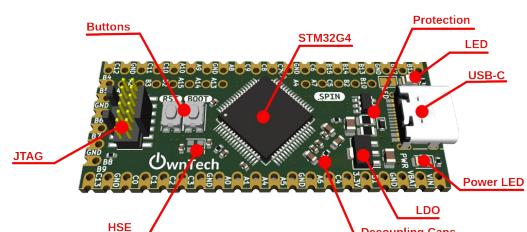


Figure 6 – Les périphériques d'accès au micro-contrôleur

La communication a été intégrée sur le bord supérieur de la carte de puissance. Cela permet de séparer le bord pour la connexion de la puissance de celui de la connexion de la communication.

Toutes les voies de puissance et de communication sont accessibles à travers des connecteurs doubles, ce qui permet la connexion de plusieurs cartes en *daisy chain*. La communication utilise des câbles RJ45 qui ont été choisis par leur ubiquité et blindage. La Figure 5 montre l'interconnexion entre 3 cartes.

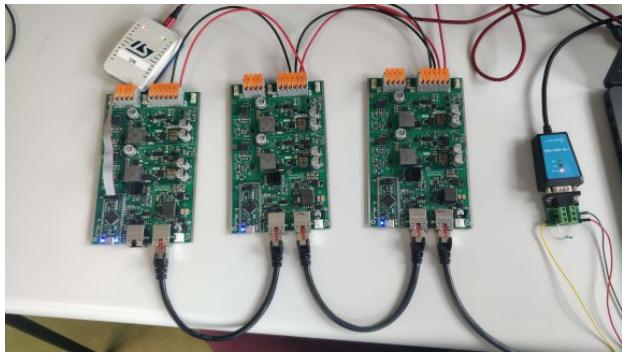


Figure 5 - L'intégration matérielle de la carte TWIST

La connexion en *daisy chain* des cartes répond au besoin de modularité et standardisation de l'IoT. Cela permet de les synchroniser et répondre à plusieurs cas applicatifs typiques en électronique de puissance.

### 5.2. Technique d'intégration Software

Le processus de définition du cahier de charge logiciel se base sur les 4 tâches types de la stack digital illustrée en Figure 2. Le cahier des charges du logiciel est donné au Tableau 5.

Tableau 5. Blocs génériques associés à la carte TWIST

Tâche	Intégration logicielle
Communication	Une tâche asynchrone qui puisse être appelée en différée pour le CAN. Un module logiciel qui puisse être appelé dans une tâche temps-réel pour le RS485 et l'analogique.
Applicative	Une tâche asynchrone qui puisse être appelée en différée.
Commande	Une tâche temps réel qui puisse être à la fois programmée pour faire de la commande en <i>voltage mode</i> (calcul de PWM) ou en <i>current mode</i> (calcul de seuil de comparaison). Cette tâche doit être compatible avec une synchronisation en tant que maître ou esclave.
Périphérique micro-contrôleur	Timer haute résolution (HRTim) intégré avec le déclenchement de l'ADC. Forte intégration avec un DMA pour décharger la CPU.
RTOS	Le RTOS choisi a été le ZephyrOS, qui supporte l'utilisation de la puce STM32 choisie. Il permet la création et automatise gestion des tâches asynchrones. Il intègre aussi un moyen automatisé de déclarer le matériel et d'activer des modules logiciels à travers des <i>device trees</i> .

La Figure 7 montre la structure de l'intégration logicielle développée.

La première couche basse représente les modules déjà présents dans Zephyr (en bleu) et ceux développés au LAAS-CNRS pour intégrer la reconfigurabilité des périphériques

spécialisées pour la puissance disponibles dans le micro-contrôleur choisi.

Une API bas niveau a ensuite été développée afin d'offrir une couche d'abstraction des périphériques du micro-contrôleur. Cela permet de répondre au besoin de simplification des paramètres à saisir par l'utilisateur final.

Des librairies peuvent être interfacées avec cette API bas niveau, spécialement pour déployer des algorithmes de contrôle ou autres. Le code utilisateur vient s'appuyer sur cet ensemble d'éléments pour déclarer des tâches, gérer l'acquisition des données et déployer un code applicatif. Des exemples clé en main sont mis à disposition des utilisateurs dans un souci pédagogique et d'accessibilité comme a été fait dans l'IoT.

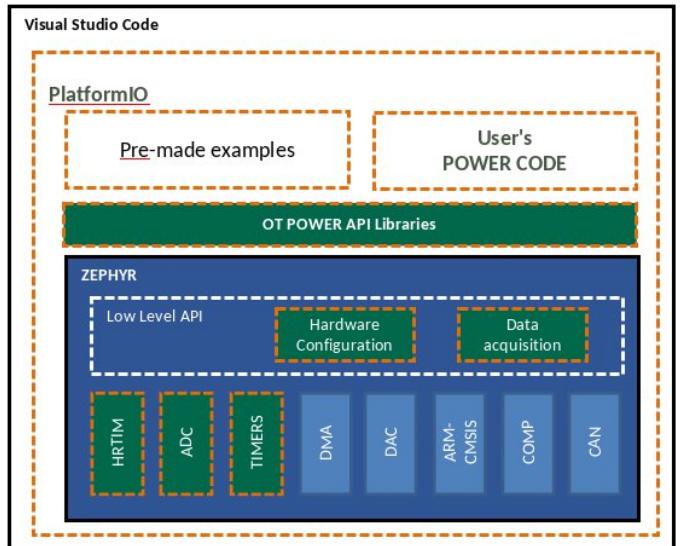


Figure 7 – Les modules de paramétrabilité

La figure 8 représente l'architecture du code utilisateur. Il reprend la simplification proposée dans l'IoT où l'utilisateur vient paramétriser son logiciel et matériel pour ensuite saisir le code lié aux tâches asynchrones et la tâche temps-réel. Les modules mobilisés dans le code sont affichés à droite.

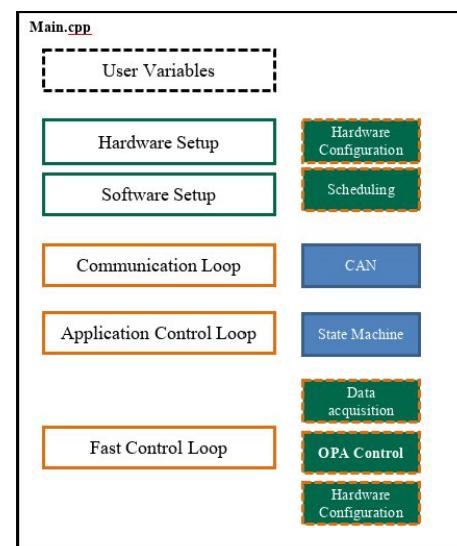


Figure 8 – L'architecture du code utilisateur

## 6. DISCUSSION

La méthode de dimensionnement de convertisseurs de puissance proposée implique 3 phases.

1. **Dimensionnement fonctionnel :** Choisir quel type de fonction de puissance doit être assurée par le convertisseur, quel cas applicatifs à être couverts, la modularité attendue et les exigences en terme de communication entre modules
2. **Co-dimensionnement matériel et logiciel :** Choisir une topologie de puissance adaptée aux fonctions ciblées, choisir des capteurs adaptés aux grandeurs et aux fonctions choisies, choisir le micro-contrôleur adapté aux modes de contrôle bas niveau, choisir le RTOS adapté au micro-contrôleur
3. **Co-vérification matériel et logiciel :** Vérification du rendement et des *ripple* de tension et courant pour les fonctions ciblées, vérification du temps de réponse à un échelon de rapport cyclique, vérification des temps de synchronisation entre cartes, vérification de l'instant de déclenchement des ADCs, vérification des temps d'exécution des différentes fonctions du code

L'originalité de la méthode proposée est d'introduire des contraintes de fonctionnalité liées à l'IOT et de co-dimensionner le logiciel et le matériel de puissance. Cette approche hybride cherche à anticiper l'ensemble des cas d'usage du convertisseur et donne la possibilité de faire des compromis de design entre hardware et software. Cela peut inclure le niveau de flexibilité fonctionnel versus le rendement pour les différentes cas d'usage ou la bande passante d'un module versus la bande passante de plusieurs modules ensemble.

La méthode de co-vérification doit inclure des figures de mérite liées à la fois à la performance logicielle et matérielle du convertisseur. Les détails des méthodologies de vérification ne sont pas couverts par cet article.

## 7. CONCLUSION

Le domaine de l'IOT a réussi à casser la « barrière de complexité » des technologies liées à l'informatique classique et industrielle. Cela a été fait par une intégration matérielle et logicielle qui permet de simplifier l'utilisation des cibles. Dans cet article, cette méthode a été analysée et transposée dans l'électronique de puissance.

Une analyse générale des besoins en électronique de puissance a été proposée pour établir une vue d'ensemble sa « barrière de complexité ». En parallèle cet article a analysé les techniques proposées par l'IOT. Une analyse croisée entre ses résultats a permis d'établir une méthodologie générique de dimensionnement IOT appliquée à l'électronique de puissance.

L'article ensuite propose une étude de cas par l'analyse d'un convertisseur de puissance par le biais de son intégration matérielle et logicielle. L'analyse de l'intégration matérielle se structure au tour des 5 blocs fonctionnels du convertisseur. L'analyse de l'intégration logiciel est faite au tour de la stack numérique d'un convertisseur de puissance.

Une discussion finalise l'article où l'originalité de la méthodologie est mise en avant : coupler les besoins

fonctionnels au co-dimensionnement matériel et logiciel du convertisseur de puissance.

Des travaux futurs à cette méthodologie est d'inclure les contraintes en automatique et de dimensionner une deuxième cible matérielle de puissance. Un descriptif des méthodes de vérification de l'intégration logicielle-matérielle est aussi une piste de travail futur.

## 8. REFERENCES

- [1] S. Peyghami, P. Palensky and F. Blaabjerg, "An Overview on the Reliability of Modern Power Electronic Based Power Systems," in IEEE Open Journal of Power Electronics, vol. 1, pp. 34-50, 2020
- [2] S. K. Mazumder et al., "A Review of Current Research Trends in Power-Electronic Innovations in Cyber-Physical Systems," in IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 9, no. 5, pp. 5146-5163, Oct. 2021
- [3] H. L. Ginn, N. Hingorani, J. R. Sullivan and R. Wachal, "Control Architecture for High Power Electronics Converters," in Proceedings of the IEEE, vol. 103, no. 12, pp. 2312-2319, Dec. 2015
- [4] A. Andreta, L. F. Lavado Villa, Y. Lembeye, and J. C. Crebier, "A Novel Automated Design Methodology for Power Electronics Converters," Electronics, vol. 10, no. 3, p. 271, Jan. 2021