

Une « table ronde virtuelle » sur les logiciels pour les conceptions IoT



Par Cliff Ortmeyer, Global Head of Solutions Development, chez Farnell element14

Dans l'environnement de conception actuel, le contenu logiciel d'un projet l'emporte généralement sur l'aspect matériel, souvent de plusieurs longueurs d'avance. Dans l'espace de conception de systèmes intégrés, la distinction entre « contenu » et la « taille de l'équipe » régulièrement invoquée, est importante. En particulier dans les petites entreprises, ces différents rôles ne sont pas toujours pris en compte par les différents ingénieurs. Flexibilité et adaptabilité sont des qualités essentielles de l'équipe de conception, ainsi que des caractéristiques fondamentales du support proposé par les fournisseurs du silicium sur lequel les conceptions intégrées seront exécutées.

La réutilisation est une notion clé dans la conception de systèmes intégrés. Côté matériel, les mêmes blocs de circuit apparaissent sur une famille de périphériques, ajoutant ou omettant une gamme de fonctionnalités. Ils peuvent être adaptés de manière substantielle ou légèrement, voire pas du tout, lorsqu'ils sont utilisés dans la génération de produits suivante. La réutilisation dans la dimension logicielle peut aller du simple encodage d'une fonction de la même manière qu'auparavant, car elle fonctionnait, à l'inclusion de systèmes d'exploitation prêts à l'emploi et l'adaptation de solutions génériques d'exemples d'applications.

Nous avons demandé à plusieurs fabricants de microcontrôleurs de relever quelques points clés de l'assistance qu'ils offrent à cet égard dans le but de faciliter le processus de conception de leurs dispositifs à base de silicium (souvent au niveau du pilote de périphérique et du fonctionnement utile) mais également de développer l'implication des équipes d'ingénierie envers le fournisseur qui donnera des conseils pour le choix des dispositifs et concernant les stratégies logicielles.

Chez **STMicroelectronics**, Laurent Vera, EMEA Marketing Director décrit l'effort que l'entreprise a fait pour offrir un outil logiciel comme le STM32Cube, qui « [...] est constitué de deux composants principaux : les bibliothèques de logiciels intégrées STM32CubeMX et STM32Cube. » Ce dernier regroupe des éléments tels que la génération de code de configuration C pour le multiplexage de broches, l'arbre d'horloge, la configuration d'intergiciels et de périphériques avec des assistants graphiques, la création de projets compatibles avec des chaînes d'outils pour environnement de développement intégré, le calcul de la consommation énergétique pour une séquence d'applications définie par l'utilisateur, l'importation directe de bibliothèques de logiciels intégrées du STM32 Cube depuis st.com, des fonctions de mise à jour intégrées.

Les bibliothèques de logiciels intégrées du STM32Cube incluent, selon M. Vera : « La couche d'abstraction matérielle HAL, permettant la portabilité entre différents périphériques STM32 via des appels API standardisés ; les API Low-Layer (LL), un ensemble d'API léger, optimisé et pensé pour des experts, conçu à la fois pour de meilleures performances et une exécution plus efficace ; et un ensemble de composants intergiciels, tels que des RTOS, une bibliothèque USB, un système

de fichiers, une pile TCP/IP, une bibliothèque à détection tactile ou graphique (varie selon la série du microcontrôleur). »

Renesas, fournisseur de semi-conducteurs ayant investi massivement dans une approche d'intégration logiciel + matériel, avec son offre Synergy, a pour objectif de permettre aux concepteurs de commencer le développement logiciel au niveau de l'API, avec un code puissant de niveau inférieur entièrement en place. « Le développement logiciel », déclare Andy Harding, Director Broad-based Solutions chez Renesas, « c'est la raison d'être de la plate-forme Synergy. Nous sommes conscients que les clients disposent aujourd'hui de plus d'ingénieurs logiciels que matériels. Et si nous pouvons leur proposer une programmation à un niveau plus haut (API), sans passer de temps sur un code qui ne fait pas la différence (pilotes, intergiciels, etc.) alors ils peuvent gagner du temps et économiser de l'argent et se concentrer sur la différenciation de leurs produits sur le marché. » Il ajoute : « Si vous employez des personnes ayant une expertise [en conception de circuits à base de silicium] avec de bons architectes de logiciels [...] il y a des avantages supplémentaires à ce que les logiciels soient programmés par des ingénieurs logiciel chez le fabricant de dispositifs [...] Et c'est en cela que la conception de logiciels en gardant la partie silicium en tête vous permettra d'obtenir de meilleures performances. »



Chez **NXP**, Geoff Lees, SVP and General Manager Microcontrollers, commente l'importance de l'assistance que l'entreprise a acquise au sein de ses équipes d'ingénierie : « Nous avons toujours été à la pointe des performances [...] des applications telles que la conversion d'énergie, le contrôle des moteurs et la connectivité sans fil.

Notre équipe d'ingénierie système et d'ingénierie des applications [...] comprend à la fois nos produits et les applications de nos clients, afin de pouvoir être d'une aide précieuse pour nos clients. Ils testent souvent en élaborant des designs de référence, des démonstrations ou des produits presque finis, en utilisant des logiciels prêts à l'emploi, de l'électronique, etc. Parfois, les

clients peuvent utiliser les systèmes que nous fabriquons et nous considérons notre équipe de systèmes comme le « premier client ». Au cours des cinq dernières années, nous nous sommes beaucoup concentrés sur les logiciels. Aujourd'hui, il ne s'agit pas de fournir une compatibilité large, comme Android, mais de fournir des solutions de couche d'application de produit [...] La moitié de notre équipe d'ingénieurs est constituée d'ingénieurs logiciels. »

Geoff Lees relève également l'importance du logiciel pilote pour les fonctions périphériques. « Vous pouvez disposer du meilleur matériel, mais si le logiciel ne peut pas l'exploiter, le client ne peut pas y accéder [...] Les pilotes permettent aux périphériques d'être gérés par un logiciel système de niveau supérieur. Nos équipes d'ingénierie optimisent les pilotes. Parfois, nous pouvons accéder à des fonctionnalités de hautes performances et mettre en place des logiciels open source, en créant des « pilotes enrichis » pour Android, Linux et des RTOS plus standardisés tels que FreeRTOS, ARM Embed OS, UCOS. »

Silicon Labs se concentre particulièrement sur les microcontrôleurs avec une connectivité sans fil intégrée et met à disposition une suite de logiciels qui s'appuie sur cette fonctionnalité. Tom Pannell, Senior Director of Marketing chez Silicon Labs, explique de quelle manière l'entreprise propose des « piles logicielles pour les communications sans fil, ainsi que du matériel. En fait, nous sommes passés d'une entreprise tournée vers les puces à une entreprise orientée verticalement et fournissant des circuits intégrés, des logiciels et des outils. Cela nous permet de mettre plus en valeur les circuits intégrés. L'intérêt est fondamentalement présent dans l'interconnexion entre logiciel et matériel. Lorsque nous développons des produits, nos équipes logicielles collaborent avec l'équipe hardware. Ainsi, les ingénieurs logiciels et matériels peuvent influencer la définition des produits. Les outils sont également très importants : le monde n'est pas parfait. Si un client souhaite déployer 200 nœuds maillés, il doit disposer d'outils qui lui montrent comment le maillage fonctionne/se forme. Ils ont besoin d'outils qui leur donnent un aperçu au niveau des paquets. »

L'ensemble de compétences associées à la transférabilité et à la réutilisation de logiciels entre les gammes de produits et les familles de périphériques est appelé « nimbleness » (« adaptabilité ») par **Texas Instruments**. À ce propos, Henry Wiechman, Embedded Processor Software Marketing and Strategy, nous dit : « L'assistance pour la plate-forme SimpleLink [basée sur l'ARM de TI] est fournie par une suite d'outils et un SDK [kit de développement partagé]. Le SDK fournit aux composants logiciels une portabilité du code à travers toute la gamme. La possibilité de passer d'un membre à l'autre est importante pour l'adaptabilité... L'adaptabilité est importante [bien que] curieusement, les clients ne semblent pas être conscients qu'ils en ont besoin. Ils savent simplement qu'ils doivent développer le produit A. Cependant, lorsqu'ils commercialisent le produit A, parfois ils voient le marché changer en cours de route. Si les clients ne pensent pas à l'adaptabilité, ils sont désavantagés lorsque le marché évolue. L'adaptabilité leur permet de pivoter rapidement, de se développer et de s'adapter au marché.

Le collègue de Wiechman, Adrian Fernandez, Microcontroller Development Experience Manager, fait néanmoins remarquer que les fonctions couramment utilisées peuvent tirer parti de la migration vers une forme matérielle, [rééquilibrant] « matériel et logiciel : pour disposer de davantage de matériel pour des tâches particulières. Nous avons déjà des appareils multicœurs qui utilisent une combinaison d'accélérateurs matériels », conclut M. Fernandez.

Les équipes commerciales terrain des principaux distributeurs [tels que Farnell element14] peuvent également constituer une ressource précieuse. Elles peuvent non seulement apporter leur expérience dans le cadre d'une grande variété de projets, tout en respectant les clauses de confidentialité mais elles savent également où trouver les supports nécessaires pour chaque fabricant.