Dix stratégies pour drastiquement réduire la complexité de la conception multi-plate-forme dans le monde de l'IoT

Auteur: Stefan Ingenhaag, ingénieur marketing des solutions MCU/MPU, ICBG, Renesas Electronics Europe

Chaque projet d'ingénierie exige des compromis délicats en termes de performance, de consommation d'énergie et de prix. Il est donc capital de trouver l'équilibre entre ces trois facteurs qui dépendent également du produit, du marché, et du timing.



Les prévisions de croissance des applications liées à l'Internet des Objets (IoT) fournissent de nouvelles opportunités pour les vendeurs et leurs équipes de conception, mais rajoutent des défis techniques matériels et logiciels à leur liste déjà longue. Le matériel et le logiciel sont intimement liés, et créent ensemble une plate-forme qui nécessitera des stratégies pour réduire la complexité de la conception multi-plate-forme. Certaines de ces stratégies comprennent :

1. LIMITER LES E/S DES CAPTEURS ET DES TRANSDUCTEURS

Établir dès le début si les E/S devront être fixes ou limitées en nombre comme en type, ou si leur quantité et leur type doivent pouvoir facilement varier. Cette décision a un impact direct sur

la sélection du MCU et des périphériques. C'est particulièrement critique lorsque les entrées/sorties ne sont pas juste de simples points numériques de basse tension, mais peuvent comporter des capteurs de température, des moteurs, et même des bus de communication, aussi bien dans des formats série que parallèle.



2. UTILISATION D'UN MODULE RF EXTERNE CERTIFIÉ

Un module qui est distinct du processeur central de l'application est la solution la plus logique dans de nombreux cas. Une solution hautement intégrée sur une puce unique peut sembler attractive en termes d'espace occupé sur la carte, de puissance, et de coût. Mais cela signifie également que des modifications ou des extensions dans les protocoles sans fil, la plage requise, voire d'autres exigences réglementaires, imposeront une refonte majeure de la conception, ou peut-être un nouveau firmware pour la liaison MCU et RF. Même si la partie codage semble simple (ce qui est peu probable), le MCU peut être inadéquat face aux nouvelles exigences et a besoin d'être lui-aussi mis à jour, ce qui augmente le temps et les risques de développement.

3. NÉGOCIER ENTRE PUISSANCE ET PERFORMANCE

Comprenez clairement où le MCU que vous sélectionnez se situe dans le schéma indiquant la puissance par rapport à la performance. Lorsque vous montez le long de la courbe de performance requise, il y aura des points qui seront des seuils au-dessus desquels vous devrez

choisir des MCU plus puissants qui consommeront plus d'énergie. Un constat qui induit une question subsidiaire : si vous descendez vers le bas de la courbe et avez besoin de moins de ressources, vous pourriez opter pour un MCU plus petit et moins cher qui consommera également moins d'énergie. Vérifiez juste que le MCU spécifique que vous choisissez possède une gestionnaire avancée des horloges suivant les fonctionnalités et les modes d'énergie. Ainsi vous pourrez optimiser les séquences opérationnelles pour une plus faible consommation d'énergie globale, tout en étant capable de gérer les périodes d'activité intensive nécessitant plus de puissance.



4. SIMPLIFIER VOTRE SÉCURITÉ

Certains processeurs ont des caractéristiques matérielles embarquées dédiées qui gèrent automatiquement certains aspects de la sécurité, indépendamment du logiciel d'application ou même du RTOS choisi. Ils peuvent donc accélérer et simplifier la résolution de vos problèmes de sécurité. Encore mieux, si cette même fonction de sécurité intégrée est disponible sur tous les MCU de votre sélection, cette partie importante du défi IoT peut être retirée de la liste «à faire», quelque soit le processeur spécifique sélectionné.

5. STANDARDISEZ VOTRE SYSTÈME

Optez pour un MCU standard 8/16bits de faible puissance, puis implémentez différentes tailles de mémoire (embarquée ou externe) suivant les exigences se déplaçant à la fois vers le haut et vers le bas de l'échelle taille/performances ; Ou optez pour un MCU 32bits unique plus grand et

acceptez qu'il y ait des capacités inutilisées dans des applications d'entrée de gamme, mais avec l'avantage d'une cohérence du code et des pilotes, ce qui simplifie la BOM et les procédures de test.

6. SÉLECTION D'UN SYSTÈME D'EXPLOITATION

Dans certains cas, un simple OS « single-thread » économique sera suffisant, mais pour de nombreux projets, vous aurez besoin d'un système d'exploitation temps réel (RTOS). Quelque soit votre choix, vous devez évaluer l'évolutivité du système d'exploitation et la disponibilité de versions de petite, moyenne et grande empreinte. Soyez réaliste quant à la taille de la version la plus petite, et aux capacités correspondantes – vous ne voulez pas «tomber dans une impasse» concernant la capacité du système d'exploitation alors que votre projet est à 80% achevé.

7. LES MISES À JOUR DU HARDWARE PAR RAPPORT AU SOFTWARE

A des points clés le long de la courbe des ressources logicielles requises pour effectuer des tâches supplémentaires (temps de développement, ressources du processeur), vous aurez à choisir entre l'ajout d'un circuit périphérique pour décharger le MCU pleinement exploité, ou opter pour un MCU plus rapide. Soyez préparés à cette éventualité, en analysant quand un MCU plus puissant peut vous permettre de remplacer une tâche matérielle par une exécution logicielle, économisant ainsi sur le coût en composants, l'empreinte, et la puissance, mais avec la probable pénalité d'un temps de développement et de débogage plus long.



8. CHOISIR VOTRE PROTOCOLE DE CONNECTIVITÉ ATTENTIVEMENT
Regardez les protocoles «plus légers» optimisés pour l'IoT plutôt que basés sur le modèle client/serveur HTTP des navigateurs Internet. Cela permettra de réduire la pile de protocoles et les exigences de traitement par un facteur de deux ou plus encore, tout en vous permettant de gérer une variété de dispositifs IoT et leurs périphériques. Dans le même temps, considérez ce qui se passera lorsque les exigences de connectivité (protocole, vitesse, intégrité) augmenteront, face à un marché plus exigeant.

9. ÉTABLIR LE PLAN DE TEST DÈS LE DÉBUT DE LA CONCEPTION
C'est à la fois essentiel et complexe, en particulier lorsque le sans-fil fait partie de la conception. De quelle manière la vérification, informelle puis formelle, que le produit final répond aux normes techniques industrielles du marché et aux exigences réglementaires affecte votre cycle « fixé» et donc votre temps de mise sur le marché. Si vous devez changer que ce soient les procédures de test des prototypes ou la mise en place des tests de production alors que vous redimensionnez votre produit et ajoutez des fonctionnalités pour gérer différentes applications, cela représente beaucoup plus de travail, d'incertitude et de risque. L'utilisation de modules matériels et logiciels pré-certifiés, obtenus sous licence, assurera la conformité et la compatibilité pour la plupart des aspects de la conception finale, mais pas tous. Soyez très vigilants sur toutes les spécifications réglementaires de haut niveau concernant la conception et la vérification (comme pour la fiabilité des produits médicaux) qui peuvent impacter vos logiciels – et auxquels de vos produits elles s'appliquent, si ce n'est à tous.

10. SÉCURITÉ, SÉCURITÉ, SÉCURITÉ
Adoptez des techniques logicielles et des tactiques capables d'évoluer au travers des produits, correspondant aux exigences applicatives et aux interfaces utilisateurs pour l'IoT (le cas échéant), tels que les pare-feu, l'authentification, et même les mots de passe. Identifiez les ressources de sécurité dont vous avez besoin dans une liste hiérarchique – démarrage sécurisé, authentification, communication sécurisée, pare-feu, détection du sabotage, rapports d'événements, audit des commandes à distance et gestion des stratégies de sécurité – et veillez à ce que l'implémentation effective de chacune d'elles soit appropriée et faisable, compte tenu des ressources logicielles dont vous disposez. Évaluez si le «fardeau» d'ajouter de la sécurité à divers produits vous poussera à peut-être opter pour un MCU plus grand ou plus rapide, et ayez un plan pour vérifier la robustesse des étapes de sécurité que vous implémentez.

Conclusion
Lorsque de nouveaux produits ou des produits supplémentaires sont développés, le «sweet spot» va sans aucun doute être déplacé pour répondre à l'évolution des exigences tout en évitant des compromis excessifs. Les concepteurs ont les moyens de garantir que ces changements n'impacteront pas les coûts, le calendrier ou la charge de travail plus que ce qui est absolument nécessaire en étudiant leur gamme de produits actuels et futurs, puis en s'appuyant sur cette étude pour choisir une plate-forme qui puisse être adaptable à divers types de produits avec un minimum de conception et un maximum de réutilisation.