Les contraintes de sûreté de fonctionnement dans le développement de systèmes multicœurs pour applications avioniques

Alex Wilson, directeur du développement commercial, Aérospatial et Défense, Wind River

L'usage de composants System-On-Chip qui associent de nombreux cœurs de processeurs au sein d'un seul et même composant se banalise de manière inexorable dans un certain nombre de secteurs applicatifs, et notamment dans les dernières générations de smartphones et dans les équipements réseau les plus puissants. Les secteurs industriels traditionnellement conservateurs comme l'aéronautique expriment également un fort intérêt pour les processeurs multicœurs, obligeant par là-même les concepteurs de systèmes à en évaluer les contraintes pour garantir une sûreté de fonctionnement. Problèmes qui ne concernent pas forcément les marchés de l'électronique grand public et de l'informatique généraliste.

Le multicœur prend la voie des airs

Les architectures de processeurs actuellement déployées dans les systèmes avioniques ont tendance à afficher des caractéristiques très traditionnelles, les équipementiers préférant s'appuyer sur des processeurs monocœur, conformes à des standards industriels bien connus et bien maîtrisés. Néanmoins, les contraintes strictes de dimensions, de poids et de consommation électrique (SWaP) auxquelles doivent se conformer les systèmes de prochaine génération, ainsi que le coût de maintenance des équipements traditionnels, poussent aujourd'hui l'industrie avionique à envisager l'usage d'architecture multicœurs particulièrement intégrées. Historiquement, l'industrie aéronautique, tout comme le marché de la Défense, ont exercé une influence notable sur le développement des technologies et des architectures de composants. Mais cette influence diminue au fur et à mesure que le cycle de l'innovation est tiré par le silicium commercial à hautes fonctionnalités, gros volume, bas coût et basse consommation. C'est la raison pour laquelle les fabricants d'équipements avioniques préfèrent désormais utiliser des technologies dites « sur étagère » ou COTS (Commercial Off-The-Shelf) pour doper les fonctions de leurs systèmes tout en réduisant les coûts de développement.

Ces nouveaux développements génèrent de nouveaux défis. Selon la firme d'analyses de marché VDC, les architectures multicœurs sont promises à une forte croissance avec une progression annuelle moyenne de 21,4% d'ici à 2013 (composants, outils et services compris). Environ 7,1% des conceptions actuelles utilisent des architectures multicœurs et ce pourcentage devrait progresser à hauteur de 11,4% d'ici à 2013. La même enquête VDC a néanmoins montré que 75% des utilisateurs interrogés disposaient de moins de deux ans d'expérience dans le domaine des technologies multicœurs.

Les contraintes de certification posent une autre sorte de défi à l'adoption des technologies multicœurs par les concepteurs de systèmes critiques, tout particulièrement dans le domaine de l'avionique où l'obtention des certifications RTCA DO-178C et EUROCAE ED-12C, extrêmement strictes, sont nécessaires.

Une complexité système croissante

La sûreté de fonctionnement est clairement l'un des critères de prime importance dans le monde de l'avionique, mais ce secteur voit aussi poindre une forte tendance à l'augmentation des fonctionnalités des équipements. Pour gérer au mieux ces deux objectifs, les équipementiers OEM du secteur doivent réussir à intégrer de nombreuses applications sur des plates-formes de calcul partagées, tout en garantissant une stricte séparation entre ces mêmes applications afin de limiter l'impact des erreurs et de simplifier leur gestion. Dans le cas des plus grands avions de ligne, l'habitude jusqu'ici a consisté à utiliser des dizaines, voire des centaines, d'ordinateurs ou d'unités remplaçables (LRU – Line Replaceable Units) différents pour fournir des fonctions opérationnelles distinctes au sein de l'appareil, comme le système de commande du train d'atterrissage, le radar météorologique, la gestion du carburant, etc. Dans cette architecture de systèmes avioniques « fédérative », chaque LRU dispose de sa propre structure de coûts de développement et de déploiement et ajoute sa propre contribution au budget SWaP global des équipements électroniques de l'avion. Par ailleurs, il convient de ne pas oublier les coûts additionnels des pièces de rechange, de maintenance et de formation associés à cette multitude de systèmes distincts, ainsi que les questions particulièrement complexes de gestion de l'obsolescence.

La possibilité de gérer toutes ces applications sur un seul composant de silicium ou, tout du moins, sur un nombre réduit de systèmes est donc particulièrement attractive, dès lors qu'il est possible de prouver qu'un niveau équivalent de sécurité ou de sûreté de fonctionnement peut être atteint. Quoi qu'il en soit, une réduction du nombre de pièces de rechange et des problèmes de maintenance ne peut qu'améliorer la fiabilité globale et diminuer les coûts opérationnels.

Le maintien des standards

L'industrie avionique a déjà adopté le principe de consolidation en utilisant la virtualisation dans des appareils comme le Boeing 787, et ce via un concept d'Avionique modulaire intégrée (ou IMA – Integrated Modular Avionics) qui repose sur le standard ARINC 653 et qui autorise l'exécution de multiples applications sur un seul et unique cœur. La norme ARINC 653 est la spécification universellement acceptée pour le partitionnement spatial et temporel dans les systèmes d'exploitation temps réel (RTOS) critiques vis-à-vis de la sûreté de fonctionnement. Ce standard peut également accommoder plusieurs niveaux de sûreté de fonctionnement au sein d'un même système. Aussi un système IMA est-il capable d'assurer simultanément, sur le même cœur, l'exécution d'applications de Niveau A (le niveau de criticité le plus élevé dans la norme DO-178C, *Software Considerations in Airborne Systems and Equipment Certification*) et de Niveau B, en s'appuyant sur du logiciel et la structure du processeur pour assurer une isolation parfaite entre applications.

La figure 1 montre une architecture monocœur typique sur laquelle s'exécute un système ARINC 653 et où chaque application, telle que le contrôleur de vol ou le radar, tourne avec un niveau différent de sûreté de fonctionnement. L'OS s'appuie sur le processeur pour configurer ces différentes zones et procurer une excellente isolation entre applications. Afin d'assurer le support de multiples niveaux de sûreté de fonctionnement sur un seul et unique processeur, l'intégrateur système doit procéder à une analyse de sûreté et prouver la robustesse du système via des mesures de performances du processeur et de la réactivité du noyau, ce dernier gérant les communications avec toutes les ressources système et contrôlant les entrées-sorties.

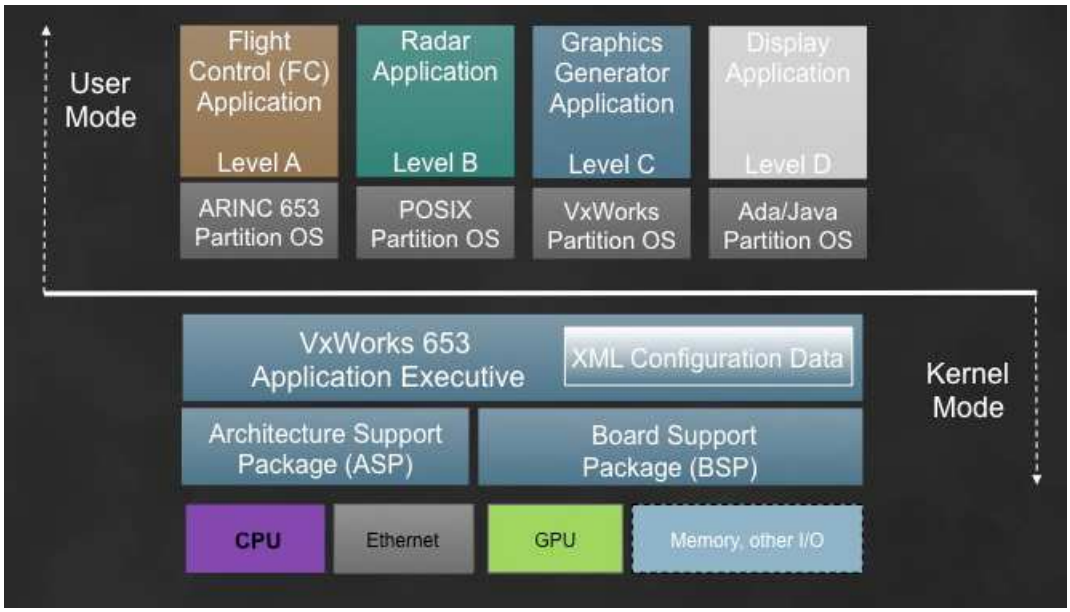


Figure 1 : Architecture monocœur typique avec exécution d'une partition ARINC 653

Choisir une architecture multicœur

Les processeurs multicœurs augmentent la complexité des plates-formes avioniques et peuvent fortement impacter les analyses de sûreté sur ces mêmes plates-formes. Au premier abord, l'architecte système peut opter pour un OS SMP (Symmetric Multi-Processing), qui exécute une seule et unique instance d'OS sur l'ensemble des différents cœurs. Mais il peut également choisir d'isoler chaque OS via une approche AMP (Asymetric Multi-Processing) supervisée, chaque OS étant couplé à un cœur individuel (voir figure 2).

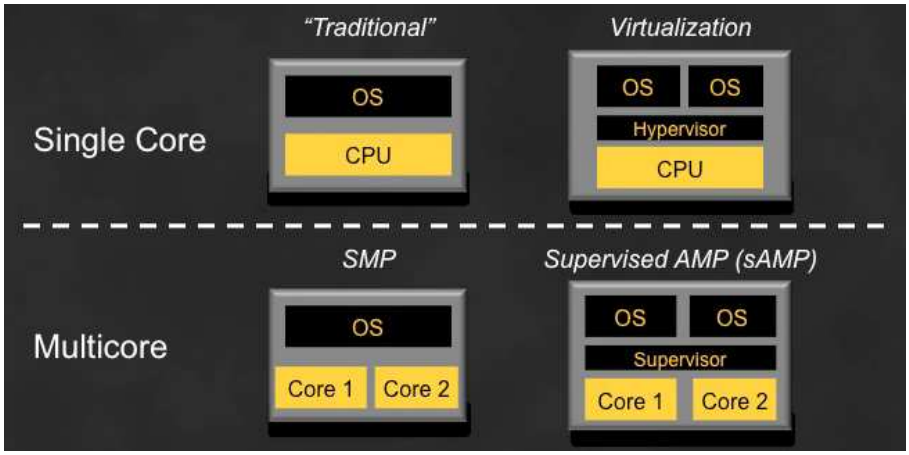


Figure 2 : Configurations logicielles primaires ; il est possible d'utiliser des combinaisons arbitraires de ces configurations primaires pour générer des configurations plus évoluées

En théorie, il est plus facile de fournir le niveau d'isolation requis lorsque chaque cœur exécute un OS différent, tandis que le partage d'un même OS par de nombreux cœurs signifie que n'importe quelle application pourrait tourner sur n'importe quel cœur, de la même manière que Windows tourne sur un PC par exemple. Il est par ailleurs potentiellement plus simple de faire migrer une application sur un système SMP, car l'application ignore le fait qu'il existe de multiples cœurs, l'OS se chargeant de prendre en charge et de gérer toutes les ressources de calcul. Mais ce modèle présente un sérieux inconvénient en termes de sûreté de fonctionnement car la norme DO-178C exige une preuve de déterminisme. L'OS peut en effet lancer, à un moment donné, l'exécution d'une application particulière sur l'un des cœurs et, à un autre moment, la lancer sur un cœur différent. Si l'application s'exécute sur un autre processeur, un rechargement de mémoire cache pourra s'avérer nécessaire, ce qui modifie les performances système. Qui plus est, l'analyse du pire temps d'exécution (WCET – Worst-Case Execution Time), déjà complexe en soi et requise pour la sûreté de fonctionnement, pourrait s'avérer encore plus longue et coûteuse à produire.

Les architectures basées sur le principe AMP, quant à elles, peuvent être supervisées par un logiciel du nom d'hyperviseur pour contrôler l'accès aux ressources matérielles partagées et supporter de multiples applications dotées de niveaux de criticité différents. Par ailleurs, afin d'accroître le déterminisme, il est possible de forcer une application spécifique à s'exécuter sur un cœur donné, et ce quelle que soit la configuration du système. La figure 3 schématise un sac électronique de vol (EFB) sur une architecture multicœur, où la complexité est accrue de manière significative par la présence de cœurs de processeur supplémentaires. Différents types d'applications peuvent s'exécuter avec différents niveaux de sûreté de fonctionnement, ce qui permet notamment l'exécution d'un environnement Linux ou Android sur l'un ou plusieurs de ces cœurs, tandis que d'autres cœurs sont réservés à l'exécution de VxWorks® adressant les contraintes de sûreté de fonctionnement critique de Niveau A, par exemple. Le système est partitionné et le nombre adéquat de cœurs de processeur est alloué afin de fournir le niveau de performances recherché.

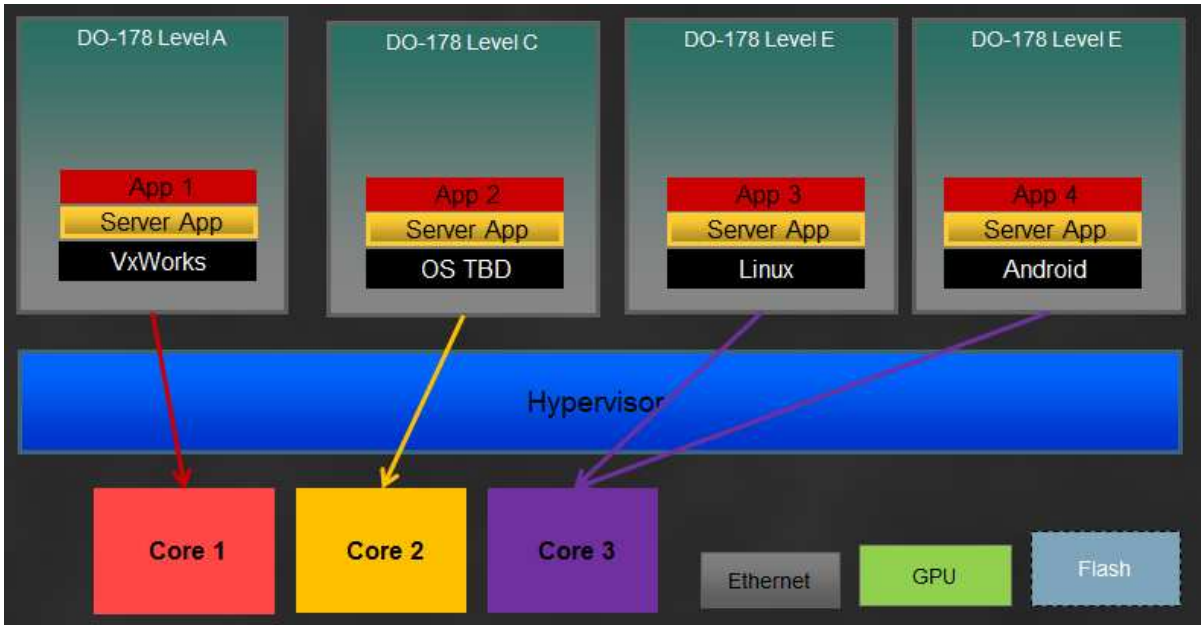


Figure 3 : Exemple d'une sacoche électronique de vol déployée dans un environnement multicœur

Les ressources partagées

Toutefois, dans les deux modes de traitement - SMP et AMP supervisé -, le problème des ressources partagées entre les cœurs et les applications reste le même. Et c'est de loin le plus grand défi à relever pour les processeurs multicœurs en termes de sûreté de fonctionnement.

Pour résoudre les problèmes de sûreté de fonctionnement sur un processeur multicœur typique, il faut répondre à de multiples questions. Existe-t-il un cache partagé de niveau 1 ou de niveau 2 et, si c'est le cas, comment la perte de déterminisme est-elle gérée au niveau des caches lorsque s'exécute une application de Niveau A ? L'existence d'un point unique de défaillance au niveau de l'accès mémoire est-elle acceptable ? Le système intègre-t-il un fond de panier à architecture de commutation ou un fond de panier à architecture de bus standard, et, dans ce dernier cas, comment le processeur gère-t-il les priorités d'accès dans ses différents modes de fonctionnement ? Que se passe-t-il au moment de la réception d'une interruption au niveau des entrées/sorties de l'équipement, et quel est le processeur qui prend en charge la gestion de cette interruption? Les interruptions déclenchées par l'horloge sont-elles synchronisées entre tous les cœurs, ou y a-t-il une entrée d'horloge unique pour chacun des cœurs ? Et le processeur de Niveau A, doit-il gérer le système de contrôle de vol ou le système de divertissement à bord ?

Il n'existe pas de règles ou de directives industrielles portant sur l'usage de processeurs multicœurs dans les systèmes avioniques. Qui plus est, tous les composants multicœurs sont différents ; les fabricants de semiconducteurs ne les ont pas conçus de manière similaire et chaque architecture doit être examinée indépendamment des autres pour procéder à une analyse de sûreté de fonctionnement. Les développeurs doivent simplement effectuer des essais, examiner le fonctionnement et le comportement des composants et régler les problèmes en conséquence.

La virtualisation

La virtualisation peut simplifier considérablement le processus de développement. Un système virtualisé fait appel à une couche logicielle dite « hyperviseur » qui supporte plusieurs systèmes d'exploitation et offre des communications rapides entre processeurs.

L'hyperviseur offre un certain niveau d'isolation entre les environnements d'exécution et se charge de la gestion des entrées/sorties, tout en permettant aux développeurs de systèmes avioniques de mettre en œuvre un partitionnement ARINC 653. Ce même hyperviseur configure la façon dont tel cœur gère telle zone mémoire, précise quel cœur prend en charge l'interface réseau, etc. La virtualisation permet la gestion de multiples composants du point du vue logiciel. Du point de vue matériel néanmoins, il se peut que l'isolation - lors de l'usage de caches partagés par exemple - reste encore à prouver.

Une couche de virtualisation telle que Wind River Hypervisor contribue à alléger les contraintes SWaP dans les systèmes avioniques (voir figure 4).

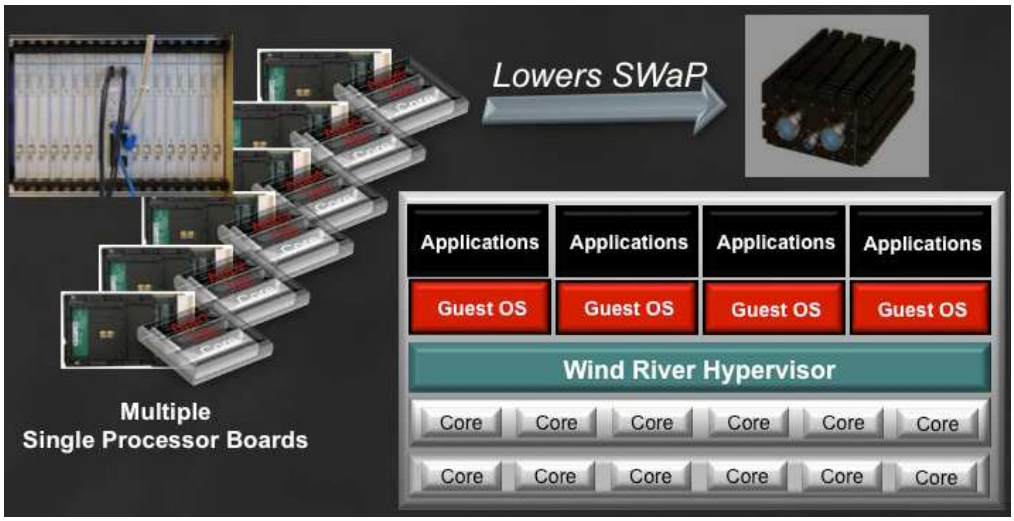


Figure 4 : La virtualisation permet de diminuer les dimensions, le poids et la consommation électrique (SWaP) des systèmes

L'hyperviseur fournit une couche d'abstraction vis-à-vis des ressources matérielles sous-jacentes et permet de configurer des partitions logicielles qui peuvent être gérées facilement et qui peuvent migrer sur les différents cœurs. Si l'application et les systèmes d'exploitation sous contrôle (« guests » OS) sont correctement spécifiés, il est alors parfaitement envisageable de s'appuyer sur la virtualisation logicielle pour faire tourner des systèmes critiques sur des plates-formes multicœurs, tout en assurant une sûreté de fonctionnement. En résumé, l'hyperviseur contrôle les ressources matérielles, garantit une isolation robuste entre environnements, et permet l'exécution simultanée d'un assortiment de systèmes d'exploitation, pour une intégration plus rapide et des mises à niveau technologiques plus aisées.

--- FIN ---

Alex Wilson obtained a BSc(Eng) in Electrical Engineering from Imperial College, London in 1986. Prior to Wind River, Alex worked at British Aerospace on Automated Test Equipment for various Inertial Navigation Systems using VME and RTOS

technology. He then worked as a Field Applications Engineer (FAE) for Motorola Computer Group working with 68k and PowerPC VME boards and 3rd party Real Time Operating Systems. He joined Wind River in the UK as an FAE in 1996 supporting VxWorks and Tornado. In 2002 he became European Business Development Manager for Wind River focusing on the Aerospace and Defence market. As Director of Business Development, Aerospace and Defence for Wind River, he is responsible for A&D programs, and is part of the Wind River Strategic Marketing Team.