



INFORMATIQUE INDUSTRIELLE

Les GPU enfin prêts pour une utilisation industrielle

Le calcul sur GPU, qui consiste à utiliser des processeurs graphiques en tant que coprocesseurs de calcul, connaît un franc succès dans le secteur de l'informatique grand public et des supercalculateurs. Les industriels, pour leur part, attendaient que des produits adaptés à leurs besoins (disponibilité des composants sur le long terme, cartes et systèmes durcis) arrivent sur le marché. Aujourd'hui, les premiers GPU taillés pour l'industrie sont disponibles. Les déploiements devraient démarrer d'autant plus vite que de grands éditeurs comme Mathworks ou National Instruments s'y intéressent de près.

Recherche fondamentale, conception de circuits électroniques, prospection pétrolière et analyses sismographiques, imagerie médicale : les secteurs qui utilisent les GPU sont déjà nombreux. Pour chacun de ces projets, les GPU ont apporté des gains non négligeables, avec des temps de calcul réduits d'un facteur 10, voire 100 dans certains cas.

dans la recherche principalement. Ce sont les débuts du GPGPU (General Purpose computing on a Graphics Processing Unit), qui signifie littéralement "faire des calculs génériques avec un processeur graphique".

Pérennité et mémoire ECC

Pour leurs calculs GPGPU, les chercheurs emploient des PC du commerce avec des cartes graphiques de dernière génération. Mais une telle architecture ne pourrait pas être utilisée telle quelle

dans l'industrie. Pour que la technologie GPU quitte les laboratoires, il faut qu'elle soit implantée sur des cartes réellement industrielles, offrant une disponibilité supérieure à celle proposée pour le matériel grand public.

Avec le lancement des cartes Firestream en 2009, AMD fournit un premier élément de réponse. « Ces cartes, →

En 2004, le fabricant de cartes graphiques Nvidia lançait la NV30, première puce graphique programmable 32 bits. Les chercheurs s'y intéressent immédiatement, et pour cause : ces puces, prévues à l'origine pour calculer des polygones, se révèlent extrêmement performantes pour réaliser des calculs matriciels. Mais à l'époque, ces composants ne pouvaient être programmés qu'avec des langages très orientés "graphique" (OpenGL, Direct3D, etc.). Deux ans plus tard, Nvidia propose une solution avec la technologie Cuda (Compute Unified Device Architecture). Cette architecture matérielle permet aux processeurs graphiques d'exécuter d'autres

fonctions que l'affichage. Cuda est aussi un environnement de développement. En effet, plutôt que de créer un nouveau langage de programmation, les ingénieurs de Nvidia préfèrent ajouter des fonctions aux langages courants (C, C++, Fortran, Java et .Net). Avec des outils tels que "C for Cuda", il devient facile d'exécuter certaines parties d'un programme sur un processeur graphique (ou GPU, pour Graphics Processing Unit). Par la suite, les sociétés Apple, AMD (qui a racheté ATI en 2006), Intel et Nvidia établissent les spécifications d'OpenCL, un modèle de programmation standard et Open Source pour tous types de systèmes hétérogènes (associant CPU et GPU). Les projets se multiplient,

L'essentiel

- L'utilisation de processeurs graphiques (GPU) pour réaliser des calculs est désignée par l'acronyme GPGPU.
- Des gammes de cartes industrielles à base de GPU sont désormais disponibles sur le marché.
- Des éditeurs de logiciels industriels adaptent leurs produits aux GPU.
- Les applications industrielles devraient se multiplier.



Progressivement, les constructeurs ont proposé des versions industrielles de leurs cartes à base de GPU (ici une carte à refroidissement passif).

À l'origine, les GPU étaient utilisés uniquement pour l'affichage. Les industriels de l'animation et du jeu vidéo en font un usage de plus en plus fréquent.

Ces cartes et modules industriels intéressent immédiatement les fabricants de serveurs. Aujourd'hui, tous les grands acteurs de ce secteur proposent des systèmes mêlant CPU et GPU. Il faut dire que ces nouveaux serveurs permettent des gains de place importants.

→ à refroidissement passif et à la disponibilité prolongée (trois ans minimum), apportent la "double précision" pour les calculs à virgule flottante (chaque valeur est codée sur 64 bits au lieu de 32, ce qui permet de meilleures approximations et donc une réduction du risque d'erreurs), lance Bruno Stefanizzi, spécialiste GPU chez AMD. Surtout, elles offrent un rendement performances/consommation encore inégalé à ce jour sur le marché. » En 2010, Nvidia va plus loin avec son architecture Fermi. Les puces Fermi bénéficient de la double précision et d'une disponibilité

prolongée, comme celles d'AMD, mais elles se distinguent par l'intégration d'une mémoire à correction d'erreur (ECC, pour Error Correction Coding). « Cette technologie était très attendue par les industriels : certains d'entre eux, notamment dans le secteur de l'embarqué, n'auraient même pas envisagé de porter la moindre application sur du GPU s'il n'y avait pas cette mémoire ECC », explique Jean-Christophe Baratault, directeur des ventes GPU computing chez Nvidia. Autre preuve qu'on a bien quitté le monde du graphique pour celui du calcul scien-

tifique : on ne trouve même plus de sortie graphique sur les dernières cartes industrielles. Ces GPU sont disponibles sous forme de cartes et de modules à refroidissement passif, ainsi qu'en racks 19 pouces 1U. Et ces produits se trouvent déjà tous intégrés dans des machines de nos partenaires OEM (HP, Dell, Bull, Sun, entre autres). »

Une réelle avancée en termes de performances

Installées dans des serveurs, ces nouvelles cartes apportent des gains importants en

Le portage de programmes sur GPU en 4 étapes

Le portage de programmes sur GPU est totalement différent de la répartition de plusieurs ordinateurs (on parle de "cluster d'ordinateurs"). En effet, pour qu'un programme soit exécuté sur un cluster, il faut le scinder en plusieurs morceaux qui seront exécutés séparément sur chacun des ordinateurs du cluster (aussi appelés "nœuds"). Avec le calcul sur GPU, l'approche est différente : les GPU n'intègrent pas autant d'instructions que les processeurs classiques, mais ils disposent de beaucoup de mémoire et leurs fonctions de répartition des tâches entre les cœurs (parallélisation) sont très avancées. Le meilleur moyen de tirer parti de la puissance d'un GPU consiste donc à lui faire exécuter une même opération algébrique simple sur une grande quantité de données.

Le plus souvent, pour porter un programme sur une architecture CPU/GPU, on suivra un processus en quatre étapes :

1 Déterminer les zones du programme à porter sur le GPU.

Il s'agit de mettre en évidence les parties de l'application qui sont les plus gourmandes en temps d'exécution (on les appelle des "hotspots"). Une fois que ces zones ont été repérées, le développeur édite son code dans un environnement compatible Cuda, OpenCL ou DirectCompute (selon le type de GPU utilisé), et y ajoute des annotations. Ces annotations seront interprétées par le compilateur pour que les lignes de code correspondantes soient exécutées sur le GPU et non plus sur le CPU.

2 Optimisation de la gestion mémoire. Le CPU et le GPU disposent tous deux de leur espace mémoire réservé. Le CPU a un accès rapide à sa mémoire (30 Go/s pour les chipsets les plus récents), mais le GPU va encore plus vite : il communique avec sa mémoire à des vitesses de l'ordre de 150 Go/s. Le goulot d'étranglement est donc la liaison PCI Express entre le CPU et le GPU beaucoup moins performante (de l'ordre de 8 Go/s). On comprend donc l'intérêt d'étudier avec précision chaque transfert de données. Concrètement, on fera en sorte d'envoyer les données au GPU suffisamment tôt dans le programme, de telle sorte que tous les transferts soient terminés une fois qu'on demande au GPU de commencer à travailler. Des outils

tels que HMPP, de la société CAPS Entreprise, peuvent assister l'utilisateur pour l'optimisation de ce processus.

3 Mesurer les gains. Une fois qu'une partie du code original est portée sur le GPU et que les transferts mémoire sont pris en compte, on effectue des tests pour déterminer les gains obtenus en termes de temps de calcul.

4 Identifier de nouvelles zones à optimiser. Une fois que les gains sont mesurés, on reprend à l'étape ① pour identifier un nouveau "hotspot" et réduire encore les temps de calcul.

Une fois le programme parfaitement optimisé pour une utilisation sur architecture hétérogène, le code pourra être porté sans modification sur n'importe quel modèle de GPU d'un même constructeur. C'est l'un des intérêts de la technologie GPGPU : on peut faire les tests sur un GPU grand public avant d'investir dans un GPU industriel. Bien entendu, certains GPU ne pourront pas exécuter des commandes propres aux dernières versions de Cuda ou d'OpenCL, mais le code pourra tourner quand même. Il n'y a que dans le cas d'une architecture avec plusieurs GPU que le code devra être adapté en fonction du nombre de GPU présents.



2010 marque l'arrivée sur le marché des premiers produits réellement durcis, prévus pour une utilisation en environnements difficiles.

GE Intelligent Platforms joue un rôle moteur pour l'adoption des GPU dans l'aéronautique et la défense. Il propose les premiers systèmes VPX mêlant CPU et GPU.

performances. Pour s'en rendre compte, comparons les GPU actuels par rapport aux processeurs x86. Les derniers nés d'Intel, les Core i7, comportent de deux à six cœurs x86 et peuvent exécuter environ 50 milliards d'opérations à virgule flottante par seconde (on parle de 50 Gigaflops). Quant aux puces GPU les plus récentes, elles comptent jusqu'à 480 cœurs et leur capacité de calcul en virgule flottante dépasse les 500 Gigaflops en double précision (2 700 Gigaflops théoriques en simple précision pour la Firestream). Pour donner un ordre d'idée, un seul serveur 1U équipé d'une carte CPU et de trois cartes GPU délivre des performances équivalentes à une armoire de serveurs x86. Et ce n'est qu'un début : AMD et Nvidia se sont engagés dans une véritable course à l'armement. C'est ainsi qu'AMD, qui dispose des compétences à la fois en GPU et en CPU, va lancer en 2011 sa nouvelle architecture Fusion : des puces hybrides, qui associent CPU et GPU sur un seul composant. Elles marqueront une étape supplémentaire dans l'amélioration du ratio performance/consommation.

Les utilisateurs déjà organisés en communautés

Bien entendu, pour que cette technologie soit adoptée, il faut que les développeurs puissent se l'approprier. Peu d'inquiétudes sur ce point. Les langages OpenCL et Cuda disposent déjà d'une vaste communauté d'utilisateurs. « On compte aujourd'hui plus de 100 000 développeurs Cuda actifs, et près de 1 000 clusters tirent déjà parti de la technologie GPU, indique Jean-Christophe Baratault (Nvidia). Surtout, le langage est enseigné dans 350 universités de par le monde, ce qui fait qu'aujourd'hui la plupart des ingénieurs qui sortent des écoles d'informatique savent programmer sur GPU. » Un fait qui explique le succès des GPU, car ces jeunes diplômés en font la promotion au sein de leurs entreprises. Il faut dire que, pour une fois, une technologie qui s'appuie sur du matériel grand public présente de réels

avantages industriels. Explications de Michael Stern, responsable produits multiprocessing chez GE Intelligent Platforms : « Le point fort du calcul sur GPU, c'est la compatibilité totale entre les produits industriels et grand public, que ce soit pour les logiciels de développement ou pour les GPU eux-mêmes. L'intérêt est de pouvoir évaluer la technologie à moindre frais, car le portage sur la cible matérielle finale ne nécessite aucune adaptation du code. Pour la plupart des industriels, il reste encore inconcevable de valider du code sans disposer de la plate-forme matérielle finale, et c'est en cela que les GPU sont révolutionnaires. Aujourd'hui, dans les grands groupes, il se trouve toujours quelqu'un parmi les jeunes ingénieurs pour être convaincu que tel ou tel algorithme gagnerait à être exécuté sur un GPU. Il lui suffit de faire le travail d'adaptation lui-même, et n'importe quel PC du commerce peut suffire à faire la démonstration. Si la direction est convaincue, elle pourra alors investir dans des cartes industrielles à base de GPU et l'application sera portée telle quelle sur la cible finale ».

Pour savoir si le GPGPU est adapté à une application, il faut comprendre le fonctionne-

ment d'un processeur graphique. La puissance de calcul ne fait pas tout, et deux aspects sont importants à retenir.

Pour quelles applications ?

Premièrement, ils s'utilisent uniquement en tant que coprocesseurs : un GPU ne sait pas tout faire. Il a besoin d'être associé à un processeur classique pour l'envoi des instructions et la gestion des entrées/sorties.

Deuxièmement, certains types de calculs gagnent plus que d'autres à être portés sur GPU. Dans un processeur classique, les instructions sont traitées en série, les unes à la suite des autres, tandis que le GPU exécute un traitement en parallèle des données. Dans le monde x86, des outils existent pour répartir des programmes entre les cœurs d'un CPU, mais ils demandent des compétences avancées. A l'inverse, la force des GPU réside dans leur capacité à gérer automatiquement un très grand nombre de cœurs, sans aucune intervention de l'utilisateur. Malgré tout, « les GPU ne remplaceront jamais les CPU, assure →

Basé en Chine, le Nebulae figure en 2^e place du Top500 (classement mondial des supercalculateurs). Ce système compte pas moins de 4640 GPU Tesla de Nvidia. Plus que ses performances (1,27 PetaFlops), c'est sa consommation électrique qui impressionne : Nebulae consomme 2,55 Megawatts, à comparer aux 7 Megawatts nécessaires à l'alimentation du Jaguar (1^{er} au Top500 avec 1,77 PetaFlops).



→ Jean-Christophe Baratault (Nvidia). Un CPU apporte de l'intelligence au système, notamment par son aptitude à anticiper les prochaines instructions. Un GPU est moins intelligent, mais il dispose d'une grande réserve de puissance "brute": on lui envoie des opérations à réaliser et il se débrouille pour exécuter ces opérations le plus vite possible, en les répartissant sur un grand nombre de cœurs.»

Le portage d'une application consiste donc à déterminer, pour chaque programme, les parties qui doivent être prises en charge par le CPU et celles qui doivent être traitées par le GPU. Toutes les opérations à base de matrices, typiquement, gagnent beaucoup à être exécutées sur GPU. D'une manière générale, du moment qu'un calcul fait appel à de grandes quantités de données ou dès lors qu'on effectue la même opération plusieurs fois sur un ensemble de données, le portage sur GPU fournit de très bons résultats.

Au sein des plus grands organismes de recherche français (CNES, CEA ou encore à l'ONERA), on travaille sur des GPU depuis plusieurs années, avec selon les projets des temps de calculs divisés par 10, 20, 50 voire 80. Et aujourd'hui, tous les grands groupes qui traitent des modèles complexes disposent d'équipes travaillant sur GPU. Tout le monde s'y intéresse, que ce soit dans les secteurs de l'énergie (prospection pétrolière,

Les militaires demandent toujours plus de performances pour leurs algorithmes de traitement radar. Pour eux, l'utilisation des GPU est tout indiquée, que ce soit pour effectuer le traitement des données ou l'affichage des résultats.

notamment), de la défense, du médical, de la chimie, du traitement vidéo, de la modélisation ou de la simulation. D'autres secteurs en dehors de l'industrie, comme la finance ou la météorologie, en tirent également parti puisqu'ils exploitent des modèles et des algorithmes très complexes.

Matlab et LabView à l'heure des GPU

Nous l'avons vu, les constructeurs de processeurs graphiques ont fait évoluer leurs produits pour s'adapter aux contraintes de l'industrie. Pour sa part, Nvidia a établi plusieurs accords de partenariat avec quelques grands noms du logiciel industriel. C'est le cas de National Instruments qui a apporté des

modifications à LabView pour le support des GPU. L'environnement de programmation graphique était déjà capable de générer du code pour des processeurs multicœur et des composants FPGA. Il donne désormais accès aux bibliothèques Cuda pour déployer des applications sur des plates-formes hétérogènes (CPU et GPU).

Matlab offre également la possibilité d'effectuer des calculs sur GPU. «Nous avons ajouté des fonctions à la Parallel Computing Toolbox, notre outil pour la mise en parallèle de calculs, explique Gilles Guillemain, directeur commercial chez Mathworks France. Jusqu'à présent, on pouvait répartir un algorithme sur plusieurs cœurs d'un processeur ou sur plusieurs ordinateurs reliés en cluster. Désormais, on pourra aussi générer du code Cuda.» Il existe trois méthodes pour exploiter un GPU sous Matlab: soit on déclare une matrice dans la mémoire du GPU (en utilisant la nouvelle fonction "gpuArray") et tous les calculs seront exécutés directement sur le GPU, soit on utilise la fonction "arrayFun" pour porter une partie d'un calcul sur le GPU, soit enfin on importe directement des algorithmes Cuda sous Matlab ou Simulink (réservé aux experts en langage Cuda). «Les utilisateurs de Matlab sont des spécialistes du calcul matriciel. Ils savent depuis longtemps que les GPU répondent à leurs besoins. C'est pourquoi ces fonctions étaient très attendues, poursuit Ascension Vizinho-Coutry, responsable des ingénieurs d'application chez Mathworks. Jusqu'à présent, tous les retours d'expériences sont très positifs et les gains sont substantiels, avec des temps de calcul pouvant être réduits d'un facteur 50 pour les applications de spectrogrammes (diagrammes temps/fréquence), pour les algorithmes itératifs de type Monte-Carlo, pour les algorithmes financiers, pour toutes les applications impliquant des transformées de Fourier, et enfin pour le traitement d'images (une image étant considérée comme une matrice).»

De la même manière, LS-Dyna et Ansys, deux

Les initiatives libres ne manquent pas

Chercheurs, universitaires, industriels, professionnels de l'animation et du jeu vidéo...

le calcul sur GPU intéresse beaucoup de monde, et comme souvent aujourd'hui dans le monde de l'informatique, différentes communautés se créent autour de la technologie.

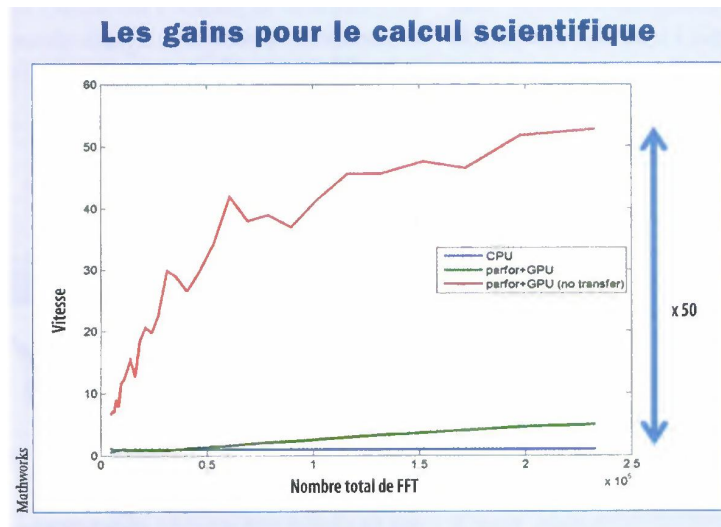
- Les développeurs Cuda sont très nombreux (l'environnement de développement de Nvidia a été téléchargé plus de 100 000 fois l'an dernier). Pour l'instant, il est le plus avancé en matière de fonctionnalités pour le calcul scientifique, c'est pourquoi il intéresse aujourd'hui le secteur de l'industrie. Toutefois, Cuda reste une technologie propriétaire, et les industriels ne sont pas forcément prêts à dépendre d'un seul constructeur, a fortiori quand celui-ci est issu du monde du jeu vidéo. C'est la raison pour laquelle le consortium Khronos Group a fondé en 2008 le langage OpenCL. Ce langage, poussé conjointement par Apple, AMD, Intel et Nvidia, devrait à terme intégrer toutes les fonctionnalités mises au point par les différentes communautés d'utilisateurs OpenCL, Cuda et Direct Compute (l'environnement de développement créé par Microsoft).
- On peut noter également le lancement en mars 2010 du projet OpenGPU. Ce dernier a plusieurs objectifs: développer des outils pour faciliter le portage d'applications OpenCL vers des architectures hétérogènes (puis hybrides avec la sortie des puces Fusion en 2011), proposer des plates-formes matérielles de référence permettant une optimisation de la consommation énergétique des supercalculateurs, et enfin aider à la validation de ces architectures sur le terrain. Le projet, supporté par les pôles de compétitivité System@tic Paris Région et Cap Digital, regroupe à la fois des industriels (Bull, Thalès et Total), des sociétés de services (Wallix, HPC Project, Numtech, Alliance Services Plus, Ateji et ESI, entre autres), des laboratoires de recherche et organismes d'état (INRIA, CEA, INRA ou encore GENCI) ainsi que des universitaires (comme Centrale Paris).
- Pour finir, si nous avons évoqué le partenariat entre Nvidia et Mathworks concernant l'intégration de fonctions Cuda sous Matlab, il nous faut également signaler l'implication de SciLab dans la technologie GPGPU. Le consortium SciLab (qui propose un outil de calcul scientifique, équivalent Open Source de Matlab) travaille également sur la compatibilité entre le logiciel SciLab et OpenCL.



éditeurs renommés dans les domaines de la modélisation mécanique et du calcul de structures, devraient lancer des versions "compatibles Cuda" de leurs logiciels d'ici à 2011. Tous ces éditeurs ont choisi Cuda plutôt que OpenCL, uniquement parce que Nvidia a beaucoup investi dans le développement de bibliothèques propres au calcul scientifique. Gageons que lorsqu'OpenCL sera au niveau de Cuda dans ce domaine, ces éditeurs proposeront le même type de passerelles pour OpenCL (pour les industriels, le fait qu'OpenCL soit un standard est plus rassurant).

Un écosystème de prestataires

On assiste également à l'émergence d'un certain nombre de sociétés spécialisées dans le portage d'applications sur plates-formes CPU+GPU. C'est le cas du français CAPS Entreprise. Cette émanation de l'INRIA de Rennes, qui réalise de nombreux projets de portage sur processeurs graphiques Nvidia ou AMD, est avant tout éditeur de logiciel. Sa suite d'outils HMPP facilite la répartition d'algorithmes sur différents cœurs, qu'il



Mathworks a réalisé un comparatif de performances pour les applications de spectrométrie (qui utilisent beaucoup de transformées de Fourier). La courbe ci-contre montre l'importance de l'optimisation des échanges entre CPU et GPU : les gains les plus importants (facteur 50) sont réalisés lorsque l'intégralité des données est chargée dans le GPU. Si les paquets de données sont envoyés au fur et à mesure dans le GPU, les gains ne dépassent pas un facteur 5 (ce qui reste non négligeable).

s'agisse de CPU multicœurs ou de GPU. Stéphane Bihan, responsable des ventes EMEA et grands comptes chez CAPS Entreprise, observe que «les GPU intéressent des spécialistes en physique du globe, en prospection pétrolière, en finance, en traitement du signal, etc., mais aucun d'entre eux n'est développeur Cuda ou OpenCL.

C'est pourquoi nous proposons de les assister dans leurs travaux. De plus, même si la programmation est relativement simple (Cuda et OpenCL s'appuient sur des langages courants), il faudra toujours utiliser des logiciels pointus si l'on veut obtenir un logiciel parfaitement optimisé.» A noter que l'outil HMPP effectue pour l'instant de la gén- ➔



→ ration et de la compilation de code, mais qu'il va continuer à s'enrichir de nouvelles fonctions. «L'objectif est de proposer bientôt un environnement de développement complet, poursuit Stéphane Bihan (CAPS Entreprise). A partir du code, le logiciel pourra fournir un ensemble d'indicateurs d'efficacité et il orientera l'utilisateur sur ses choix en matière d'optimisation».

La société HPC Project accompagne elle aussi ses clients dans leurs travaux d'optimisation de code. Pour le portage de calculs sur GPU, l'entreprise montpelliéraine s'est associée à Nvidia pour proposer des services avancés sur plates-formes Cuda.

Autre prestataire français bien implanté dans le monde du GPU computing: Vega Technologies. Cette société toulousaine réalise du portage d'applications spatiales, pour le CNES principalement. Comme il s'agit essentiellement de travaux sur des images de grande taille et de grande résolution, l'utilisation de GPU est tout indiquée, et les gains en performances vont d'un facteur 2 à 500 en comparaison avec une architecture x86. Au final, Jean-Christophe Baratault (Nvidia) se félicite qu'«un véritable écosystème industriel se soit construit autour de la technologie GPU computing. Nous considérons d'ailleurs 2010 comme étant "l'année zéro" pour le déploiement des GPU dans l'industrie. En effet, c'est seulement depuis cette année que tous les composants sont réunis: les cartes se sont industrialisées, les fournisseurs d'équipements

informatiques les ont intégrées à leurs gammes, des partenariats ont été établis avec de grands éditeurs, et des sociétés proposent des services et tous les outils d'optimisation associés»

Des systèmes GPU durcis pour l'embarqué

Même si des cartes industrielles à base de GPU sont disponibles sur le marché, il restait tout de même à satisfaire les besoins des marchés de l'aéronautique et de la défense. Ces derniers s'intéressent également au calcul sur GPU car il répond parfaitement à leurs problématiques dites de SWaP (Size, Weight and Power, taille, poids et puissance). Que ce soit pour des avions, des bateaux, des chars, des sous-marins, et à plus forte raison des drones, les constructeurs ont besoin de systèmes toujours plus compacts, légers et à basse consommation. C'est la raison pour laquelle le constructeur américain GE Intelligent Platforms a établi un partenariat de haut niveau avec Nvidia, pour l'utilisation de la technologie Cuda dans des cartes durcies. GE Intelligent Platforms propose donc depuis cette année une gamme de cartes et de systèmes GPU durcis au format VPX. «Le format VPX était le plus adapté, car il est le seul à proposer à la fois un bon niveau de durcissement et un fond de panier avec des liaisons à très large bande

Les drones sont réellement une des applications privilégiées pour l'utilisation des GPU dans l'embarqué. Ces appareils ont des impératifs très stricts en termes de poids et de consommation électrique. Surtout, ils doivent réaliser de l'acquisition et du traitement de données en temps réel, puis renvoyer ces données traitées vers le centre de commandement. Des opérations qui nécessitent une grande puissance de calcul en virgule flottante...

passante», explique Philippe Constanty, directeur des ventes pour le marché des systèmes embarqués chez GE Intelligent Platforms. La gamme se compose de deux cartes openVPX: une carte GPU (à installer dans un système déjà équipé d'une carte CPU), et une carte CPU+GPU à base d'Intel Core 2 Duo et de Nvidia GT240 à 96 cœurs, qui pourra quant à elle être utilisée seule. On trouve également des systèmes complets, sous la forme de boîtiers durcis (connectique militaire, refroidissement par convection ou par conduction) qui intègrent la carte CPU+GPU évoquée précédemment. Enfin, grâce à un kit de démarrage lancé tout récemment, les temps de mise sur le marché peuvent être écourtés.

Il ne s'agit pas là d'un acte isolé: d'autres constructeurs de systèmes aéronautiques et militaires, comme Mercury Computer Systems ou encore Extreme Engineering Solutions (représenté en France par Neomore) annoncent également la sortie prochaine de produits durcis à base de GPU.

D'après Elie Gasnier, directeur des ventes et responsable marketing chez Ecrin Systems (partenaire exclusif de GE Intelligent Platforms en France pour les applications militaires et aéronautiques), «tous les grands groupes du secteur de la défense s'intéressent au calcul sur GPU pour leurs applications radar, sonar, traitement de flux vidéo et autres systèmes d'acquisition de cibles. Ils évaluent déjà les GPU en remplacement des technologies DSP et FPGA, qui imposent des cycles de développement beaucoup plus longs. En effet, pouvoir valider des concepts et des algorithmes sans disposer de la cible matérielle est pour eux un avantage non négligeable. Pour cela, ils n'attendaient plus que des produits durcis disponibles sur étagère, et ces derniers deviennent une réalité. Les premiers déploiements ne vont donc plus tarder.»

Frédéric Parisot

Un châssis qui accélère les temps de développement

Le constructeur américain GE Intelligent Platforms complète sa gamme de cartes et de systèmes VPX à base de GPU avec un kit de démarrage. Ce produit relativement original s'adresse aux industriels qui souhaitent évaluer la technologie GPU pour des applications sur systèmes VPX 3U ou 6U, en leur permettant de gagner en temps de développement. Explications de Philippe Constanty, directeur des ventes pour le marché des systèmes embarqués chez GE Intelligent Platforms:

«concevoir une carte fond de panier pour un système VPX coûte cher. Surtout, une fois que ce fond de panier a été développé, on ne peut plus rien changer. Le problème avec les nouvelles technologies comme le GPU, c'est que les industriels ont envie de voir leur application tourner, et cela pour plusieurs raisons: ils veulent vérifier que leurs algorithmes peuvent bien être portés sous Cuda, et ils veulent aussi vérifier que les gains en temps de calcul sont aussi importants que sur le PC de démonstration. Enfin, ils souhaitent pouvoir commencer à optimiser leur application sans attendre nécessairement que le fond de panier soit produit en série. Pour toutes ces raisons, nous proposons ce châssis sur lequel

on travaille sans fond de panier, les cartes étant reliées uniquement avec des câbles. Le client peut ajouter facilement des cartes pour tester toutes les combinaisons qu'il souhaite (par exemple: 1 carte CPU et 4 cartes GPU, ou 2 cartes CPU et 2 cartes GPU). Surtout, il peut faire différents essais pour dimensionner de manière optimale chaque liaison entre cartes (PCI Express x8 ou x16, par exemple).»

Avec ce châssis sans fond de panier, les industriels peuvent commencer à programmer sitôt qu'ils ont reçu leur kit de démarrage.