



Tom Draper Design - Michèle Tcherevkoff, The Image Bank (télévision)

LE HANDY 21 sera à la fois (de gauche à droite) un téléviseur, un messenger de poche, une radio, un téléphone mobile et un système de navigation connecté au réseau Internet. L'antenne du dispositif transmettra et recevra des signaux de communication qu'un unique microprocesseur généraliste traitera.

à un réseau. Le système se chargera alors de transférer l'appel vers le récepteur de la personne en question.

Ces systèmes peuvent également transmettre des données : reprogrammé, le même matériel deviendra un appareil médical portable, qui enverra et recevra des échographies ou les signaux d'un électrocardiographe. Les médecins munis de ces appareils auront instantanément accès aux dossiers médicaux et aux résultats des analyses de leurs patients, même loin de leur cabinet.

Les caméléons et Oxygen

Les caméléons de la communication utilisant le système *SpectrumWare* seront sans doute les éléments majeurs de l'infrastructure du projet *Oxygen*. Toutefois, dans un réseau sans fil, tel celui qui relierait les *Handy 21*, l'état des canaux varie de

façon imprévisible. De plus, les diverses applications requièrent des largeurs de bande différentes et tolèrent des taux d'erreurs variables, avec, parfois, des niveaux de sécurité élevés. Par exemple, une application de commerce électronique nécessite un cryptage du signal de communication plus performant qu'une application de radio ou de télévision. Or, les interfaces d'un réseau classique sont fixes : elles sont conçues pour fonctionner dans les pires conditions qui puissent être tolérées, plutôt que pour s'adapter aux conditions réellement rencontrées par le système.

Avec le système *SpectrumWare*, nous voulons exploiter mieux les ressources du réseau en changeant les caractéristiques de la communication pour chacune des applications. Les bornes des réseaux devront alors modifier les canaux en fonction du nombre d'unités mobiles dans la zone

couverte et de leurs besoins spécifiques. Le réseau affectera des canaux personnalisés aux *Handy 21* qui nécessiteront une réponse en temps réel ou des débits d'information élevés, alors qu'il fera partager un même canal à d'autres unités moins exigeantes.

Aujourd'hui, nous construisons les programmes de communication qui équiperont les réseaux du futur. Nous conservons trois idées directrices : nos programmes sont tous généralistes ; nous prévoyons une attribution dynamique des ressources, au lieu de prévoir la pire des situations ; nous faisons exécuter par un programme le maximum de tâches, afin de minimiser la part du matériel.

John GUTTAG est directeur du Département de génie électronique et d'informatique de l'Institut de technologie du Massachusetts.

Une puce universelle

ANANT AGARWAL

Constituée de circuits qui se réorganisent automatiquement selon les tâches à accomplir, la puce Raw est au cœur du projet Oxygen.

Pour mettre l'informatique au service des utilisateurs, les informaticiens, les concepteurs de logiciels et les électroniciens repensent l'architecture des systèmes informatiques actuels. Nous étudions notamment un nouveau microprocesseur, la puce *Raw* («brut», en anglais), dont les circuits électriques seront modifiés par les programmes. Grâce à cette conception flexible, la puce *Raw* devrait être plus performante et plus rentable que les microprocesseurs d'aujourd'hui.

La conception de la puce *Raw* a bénéficié de la miniaturisation régulière des microprocesseurs. En 1987, un microprocesseur d'environ 100 000 transistors, capable d'exécuter 20 millions d'instructions par seconde, occupait environ un centimètre carré de silicium. En 1997, un microprocesseur aussi puissant n'occupait que un millimètre carré. Et en 2007, un tel microprocesseur tiendra sur une puce d'un dixième de millimètre carré, soit un dix-millième de la taille du microprocesseur de 1987. Les perspectives sont enthousiasmantes, mais l'informatique n'est-elle qu'une course à la miniaturisation?

Non, l'architecture courante des microprocesseurs devra changer! Entre le matériel et les logiciels, la plupart des ordinateurs personnels utilisent comme interface un jeu d'instructions, nommé *ISA*, qui régit le transfert des données, des mémoires aux unités fonctionnelles où elles sont traitées (additionnées, multipliées, etc.) Par exemple, une instruction du type «ADD, registre 7, case mémoire 1024, registre 8» commande au microprocesseur de stocker dans le registre 7 la somme des informations du registre 8 et de la case mémoire 1024. Cependant,

les instructions n'informent pas les programmes de l'emplacement des cases de mémoire ou des unités fonctionnelles : aujourd'hui, dans un microprocesseur, chaque case de mémoire est donc connectée à chaque unité de traitement. Cette architecture limite les performances des puces.

La diminution de taille des transistors augmentera encore le nombre de sites de stockage et d'unités fonctionnelles sur chaque puce, et réduira le cycle

Une nouvelle architecture

Nous heurtons-nous à un mur de complexité, de vitesse et d'efficacité énergétique? Tous les ordinateurs personnels contiennent un microprocesseur que l'on complète par des cartes électroniques, telles une carte modem, une carte graphique, une carte son... Ces cartes sont encombrantes. Des ingénieurs en micro-électronique doivent optimiser l'assemblage de puces spécifiques pour chaque type de carte, en minimisant la taille des connexions électriques.

Nous proposons de simplifier la production micro-électronique en utilisant des portes logiques, c'est-à-dire des assemblages de transistors qui commandent la circulation du courant électrique – et donc le traitement de l'information – sur les puces. Dans une dizaine d'années, les puces seront composées de milliards de portes logiques. Une partie seulement sera utilisée pour les applications ; le reste sera à la disposition d'autres programmes, nommés compilateurs, qui géreront la circulation de l'information.

C'est là le projet que nous avons en cours : nous construirons un processeur et un compilateur qui configureront automatiquement les connexions. Ainsi, le compilateur adaptera la puce à l'exécution des diverses applications, lesquelles seront écrites en langages de programmation de haut niveau, tel le *C* ou *Java*.

Nous avons nommé notre système *Raw*, car il met directement en contact le matériel, notamment les connexions, avec le compilateur. Dans les architectures existantes, le logiciel commande les opérations logiques de la puce (les fonctions de base telles



Jean-Michel Thibaut

de l'horloge interne de la puce, c'est-à-dire le temps d'exécution d'une opération de base, telle une addition. Néanmoins, la longueur totale des connexions restera proportionnelle à la surface de la puce, sans l'être au cycle d'horloge : les temps de transfert des données le long des connexions seront ainsi de plus en plus longs, jusqu'à limiter les performances de la puce. De surcroît, les microprocesseurs d'architecture classique consommeront beaucoup d'énergie, car l'énergie nécessaire à l'aiguillage des signaux dépend de la longueur des connexions.

«AJOUTER» et «RETRANCHER»), mais pas les connexions. En revanche, en utilisant les portes logiques libres pour diriger et stocker les signaux qui circulent sur les connecteurs de la puce, le compilateur du système *Raw* programmera les connexions.

La puce *Raw* est composée d'un réseau de nombreux pavés tous identiques (voir la figure 1), qui contiennent chacun des mémoires et des unités fonctionnelles. En outre, chaque pavé comporte un commutateur qui commande les connexions avec les pavés adjacents.

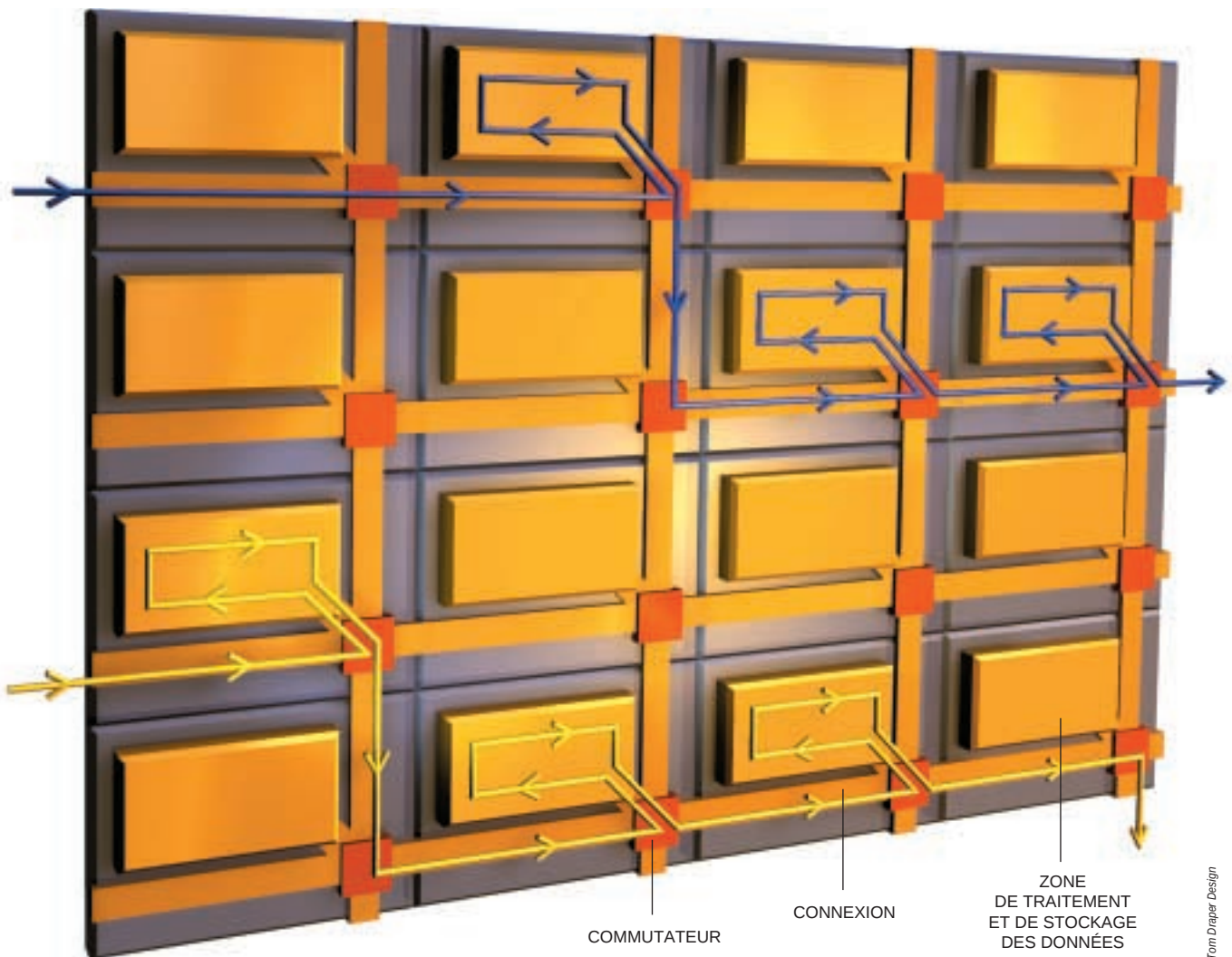
Le compilateur programmera les commutateurs de tous les pavés afin de créer un circuit spécifique à chaque cycle du fonctionnement de la puce.

Le trajet du signal est modifié pour chaque application.

Dans une première étape, le compilateur fragmente les connexions de la puce afin d'éviter les connexions longues (d'un pavé à un autre éloigné), qui requièrent de longs délais : chaque commutateur comporte des registres qui stockent les données de passage, de sorte que les connexions les plus longues sont divisées en de nombreux segments ; les signaux parcourent ainsi toute la longueur du connecteur, en passant d'un registre au suivant à chaque cycle d'horloge. Ainsi on réduit le cycle d'horloge (la fréquence d'horloge, c'est-à-dire le nombre de cycles par seconde, est alors augmentée). En 2010, les puces *Raw*

auront des fréquences d'horloge de 10 à 15 gigahertz (milliards de cycles par seconde), alors qu'aujourd'hui le microprocesseur le plus performant a une fréquence d'horloge de un gigahertz seulement. Les signaux parcourront les connexions fragmentées en plusieurs cycles d'horloge, mais de nombreuses données (une pour chaque segment) se déplaceront simultanément. Lorsque la première aura atteint sa destination, les composantes suivantes d'un signal arriveront successivement, cycle après cycle, augmentant ainsi le débit d'information dans la connexion.

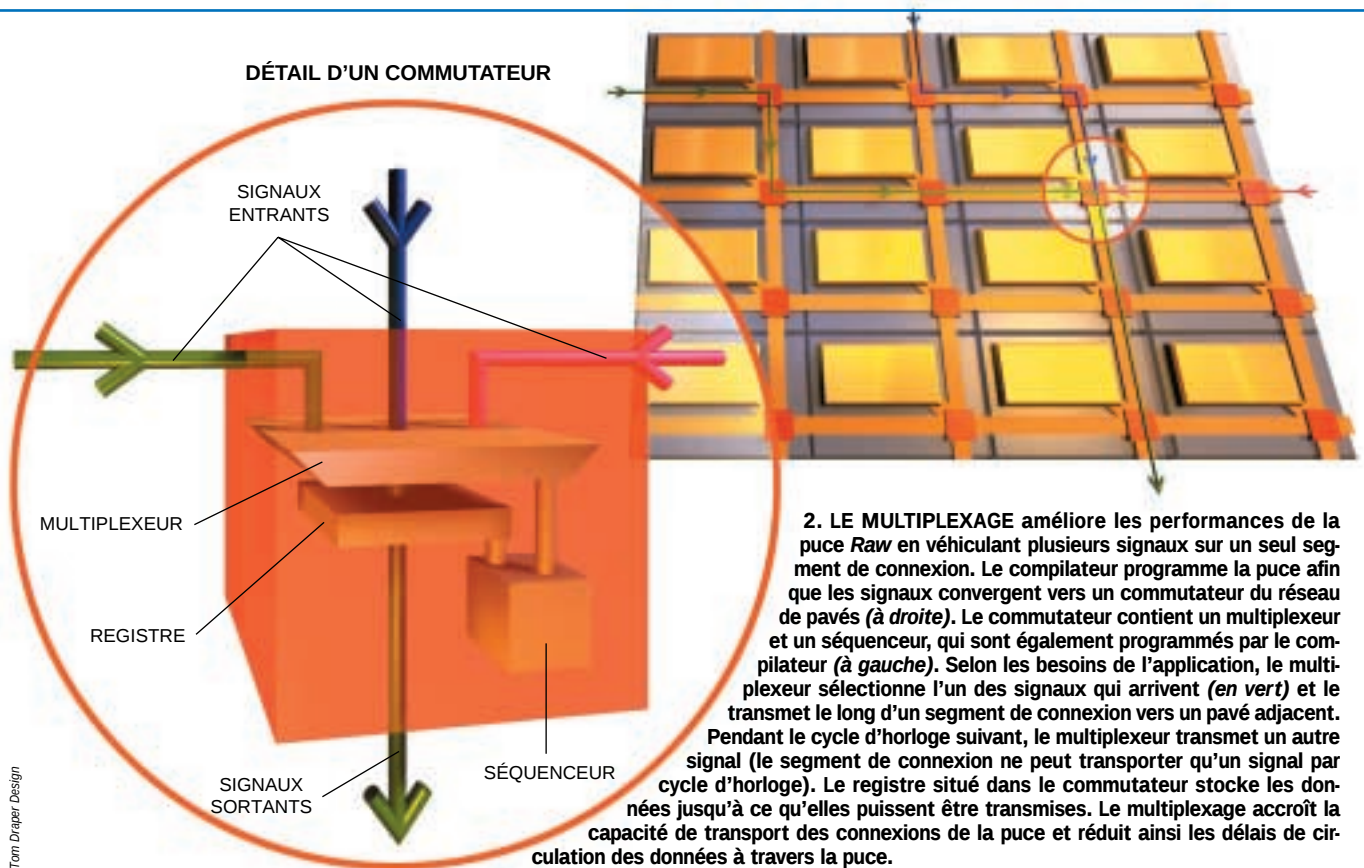
Le compilateur placera les données dans les emplacements de mémoire les plus proches possible des unités



1. LA PUCE RAW est un réseau rectangulaire constitué de nombreux pavés identiques (seule une petite partie de la puce est représentée ici). Chaque pavé contient une unité de calcul et des mémoires (rectangles orange). Les signaux se propagent le long de connexions, reliant chaque pavé à ses voisins, et sont aiguillés par des commutateurs placés à l'intersection des

connexions. Les trajets des signaux sont déterminés par un compilateur, qui programme les commutateurs selon les besoins de l'application en cours. Avec la puce *Raw*, plusieurs applications fonctionneront simultanément. Par exemple, des signaux vidéo (en bleu) et radio (en jaune) suivront chacun le chemin le mieux adapté à leur traitement.

Tom Draper Design



Tom Draper Design

fonctionnelles qui les traiteront. Le nombre de cycles d'horloge nécessaires aux valeurs d'un signal pour se propager d'un site à un autre sera ainsi minimisé.

Dans une deuxième étape, nous avons tenté d'éviter le gaspillage des connexions en utilisant chacune d'entre elles pour relier plusieurs emplacements de mémoire et plusieurs unités fonctionnelles. Ce procédé, nommé multiplexage, est comparable au regroupement de plusieurs routes issues de différentes villes en une grosse autoroute. Lorsque plusieurs signaux parviennent à une extrémité d'un segment de connexion, un multiplexeur, composé de portes logiques, les sépare : pendant un cycle d'horloge, un seul signal est transmis le long du segment donné (voir la figure 2). Ainsi, les connexions dont les multiplexeurs ont été optimisés par le compilateur achemineront beaucoup plus de signaux que les connexions classiques.

Enfin, le compilateur dirigera les signaux le long de chemins optimaux, s'assurant qu'ils arrivent quand il le faut, là où ils sont nécessaires.

Grâce à cette architecture, de grandes quantités de données, par exemple

de l'information vidéo, seront véhiculées directement dans les parties de la puce où les calculs sont effectués : les performances et l'efficacité énergétique seront bien supérieures aux possibilités actuelles. La puce Raw que nous construisons aura plus de 1 000 broches d'entrée ou de sortie de données, soit dix fois plus que les microprocesseurs d'aujourd'hui.

La puce à tout faire

La puce Raw pourra effectuer des tâches diverses, tels le cryptage, la reconnaissance de la parole, les jeux ou la communication, dans un appareil unique. Quand l'utilisateur demandera : «Transforme-toi en téléphone», son appareil trouvera le logiciel approprié, l'installera et le compilateur adaptera les connecteurs de la puce Raw.

La même puce pourra ainsi commander plusieurs interfaces de communication, tels une caméra, un écran vidéo, des haut-parleurs, un microphone, une antenne pour les télécommunications et un convertisseur de signaux analogiques en signaux numériques. Ce dernier sera intégré à la puce Raw : toutes les fonctions pour lesquelles nous sommes aujourd'hui

obligés d'acheter du matériel spécialisé seront ainsi assurées grâce aux applications configurables directement dans les connexions de la puce.

Nous avons déjà construit un compilateur capable de programmer les applications sur un simulateur de la puce Raw, par exemple une radio avec une puce Raw de 128 pavés (voir *Les caméléons de la communication*, page 40). Nos résultats indiquent que l'application fonctionnera dix fois plus vite sur la puce Raw que sur n'importe quel microprocesseur classique. Nous avons multiplié par dix encore cette vitesse en optimisant l'application sans l'aide d'un compilateur. Cependant, nous souhaitons mettre au point un compilateur 100 fois plus performant. La puce Raw deviendra alors une puce logique universelle, qui remplacera les microprocesseurs généralistes et les micropuces spécialisées.

Anant AGARWAL est codirecteur du projet Raw du Laboratoire d'informatique de l'Institut de technologie du Massachusetts.

Michael DERTOUZOS, *What Will Be*, Harper Collins, 1997.

