

sont FAUX, la sortie devient FAUX. Dans les autres cas, la sortie ne change pas. Les éléments C de Muller servent notamment à contrôler le flux des données. Par un agencement astucieux de plusieurs d'entre eux, ils permettent de gérer l'avancée de données le long d'une chaîne de passeurs électronique sans le besoin d'une horloge.

Notre groupe de recherche a récemment introduit un nouveau type de circuit *Rendez-vous*, nommé GasP (voir l'encadré page 41). Le GasP est dérivé d'une famille plus ancienne de circuits mise au point par Charles Molnar, qui travailla aux Laboratoires *Sun Microsystems*. Ch. Molnar avait baptisé sa création asP* (pour protocole de pulsation asynchrone et symétrique). Nous y avons ajouté le G, pour obtenir l'onomatopée (*gasp*) qui, en anglais, traduit la surprise, surprise créée ici par la rapidité des nouveaux circuits. Nous avons établi que les modules GasP ont une rapidité et une efficacité énergétique aussi bonnes que les éléments C de Muller, qu'ils s'assemblent mieux avec les circuits ordinaires et qu'ils sont beaucoup plus faciles à utiliser dans des circuits complexes.

Les circuits *Arbitre* remplissent, quant à eux, une autre fonction essentielle pour les ordinateurs asynchrones. Un *Arbitre*, tel un agent de la circulation posté à une intersection qui décide quel véhicule passera le premier, orchestre le traitement des données. S'il n'y a qu'une requête, l'*Arbitre* donne immédiatement le feu vert à l'action correspondante, et met en attente une éventuelle seconde requête

jusqu'à ce que la première soit terminée. Quand l'*Arbitre* reçoit deux requêtes quasi simultanément, il doit décider à laquelle accorder la priorité. Par exemple, quand deux processeurs réclament l'accès à une mémoire au même moment, le rôle de l'*Arbitre* est de n'accorder l'accès qu'à un processeur à la fois. L'*Arbitre* garantit que deux actions ne se déroulent jamais en même temps.

L'âne de Buridan

Bien que les circuits *Arbitre* n'accèdent jamais à plus d'une requête à la fois, il n'existe aucun *Arbitre* qui tranche entre les diverses requêtes en un temps limité. Les *Arbitre* actuels prennent leurs décisions très rapidement en moyenne, généralement en quelques picosecondes (un millième de milliardième de seconde). Cependant, quand les demandes sont très rapprochées, les circuits mettent parfois deux fois plus longtemps à trancher, voire, dans quelques rares cas, dix fois plus.

Les difficultés auxquelles est confronté l'*Arbitre* pour prendre ses décisions rappellent celles de l'âne de Buridan. Selon cette parabole attribuée à Buridan, philosophe français du XIV^e siècle, un âne affamé et assoiffé, se trouvant à égale distance d'un picotin d'avoine et d'un seau d'eau, mourrait de faim et de soif parce qu'il serait incapable de choisir entre l'un ou l'autre. Nous sommes confrontés tous les jours à ce type de dilemmes, par exemple lorsque deux

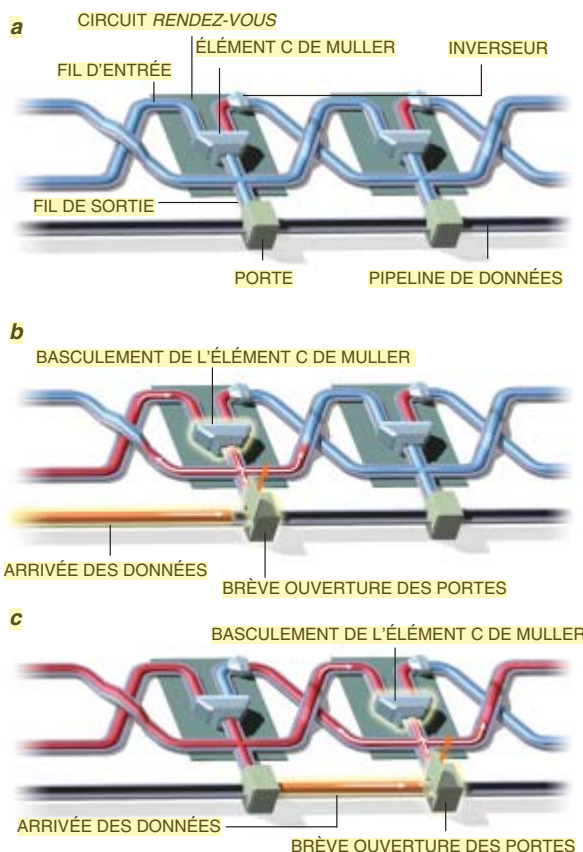
Fonctionnement d'un circuit *Rendez-vous*

Les circuits *Rendez-vous* gèrent le flux des données à travers des circuits sans horloge centrale. Ici un pipeline électronique est contrôlé par une chaîne de circuits *Rendez-vous*, nommés éléments C de Muller. Chacun de ces éléments autorise le passage des données (*en orange*) seulement quand l'étage précédent est «plein» (les données sont prêtes à poursuivre leur chemin) et que l'étage suivant est «vide».

Chaque élément C de Muller a deux entrées et une sortie. La sortie devient FAUX quand les deux entrées sont FAUX, et redevient VRAI quand les deux entrées sont VRAI (dans le diagramme, les signaux VRAI sont en bleu, et les signaux FAUX en rouge). L'inverseur fait en sorte que les entrées initiales diffèrent de sorte qu'au départ, tous les étages sont vides.

Supposons que l'entrée de gauche soit VRAI et celle de droite FAUX (a)

Un changement de l'entrée de gauche de VRAI à FAUX (b) indique que l'étage de gauche est plein, c'est-à-dire que des données sont arrivées. Comme les deux entrées de l'élément sont maintenant toutes deux FAUX, l'élément bascule sa sortie sur FAUX. Cette modification a trois conséquences : des données sont transmises dans le pipeline pour poursuivre leur trajet (les portes s'ouvrent), un signal FAUX est renvoyé vers l'élément C de Muller précédent pour que l'étage soit vidé et un signal FAUX est expédié vers l'élément C de Muller suivant, de sorte que l'étage suivant est plein (c).



Bryan Christie

personnes arrivant ensemble devant une porte marquent une pause avant de décider qui passera en premier. Dans ces situations, l'ordre de passage n'a pas d'importance, la seule difficulté consiste à prendre la décision.

Le rôle de l'*Arbitre* est précisément de prendre cette décision. Un *Arbitre* a deux états stables correspondant aux deux choix. On peut se représenter ces deux états, l'un comme la France, l'autre comme l'Espagne. Chaque requête expri-

mée auprès d'un *Arbitre* pousse le circuit vers l'un ou l'autre de ces états stables, comme un grêlon, tombé au sommet des Pyrénées, pourrait rouler aussi bien vers la France que vers l'Espagne. Entre les deux états stables, il doit néanmoins y avoir une ligne métastable, équivalente à la ligne de crête des Pyrénées. Si le grêlon tombe précisément sur cette ligne de partage, il peut rester en équilibre quelques instants sur la crête avant de basculer vers l'Espagne ou

Processeurs asynchrones et relativité

Les questions soulevées par I. Sutherland et J. Ebergen, *a priori* de nature technologique, sont aussi le reflet de questions fondamentales sur le temps. Pour connaître le temps, il ne suffit malheureusement pas de le mesurer. La difficulté rencontrée pour synchroniser tous les calculs au sein d'une puce électronique résulte aussi de ce fait : le temps n'existe pas sous la forme d'une grandeur physique unique qu'il suffirait de mesurer à l'endroit où l'on se trouve. C'est cette constatation qui a conduit à l'introduction de la théorie de la relativité. Certes, le temps est obtenu à l'aide des indications des horloges, mais ce n'est pas suffisant. Comme le dit Einstein en 1905 : « Il nous faut garder à l'esprit que tous les jugements dans lesquels le temps joue un rôle sont toujours des événements simultanés. Lorsque, par exemple, je dis : « Tel train arrive ici à 7 heures », cela signifie à peu près : « Le passage de la petite aiguille de ma montre sur le 7 et l'arrivée du train sont des événements simultanés. » Einstein ajoute plus loin : « Cette définition ne suffit cependant plus dès lors qu'il s'agit de relier temporellement des séries d'événements qui se produisent à des endroits différents. » En fait, pour disposer d'un temps coordonné en des lieux différents, il faut synchroniser les horloges qui s'y trouvent.

Pour ce faire, il suffit d'échanger des signaux. Malheureusement, il existe une vitesse maximale à laquelle peuvent se propager toutes formes de signaux, fixée par la vitesse de la lumière dans le vide, de l'ordre de 300 000 kilomètres par seconde. Il en résulte que la simultanéité de deux événements se produisant en des lieux différents ne peut plus être définie de façon absolue, mais prend un caractère conventionnel. De ce point de vue, les concepteurs des microprocesseurs actuels sont confrontés au même problème que les ingénieurs de la fin du XIX^e siècle : ils voulaient accorder les horaires des trains, mais se heurtaient à la difficulté de synchroniser les horloges des gares sur un territoire large de plusieurs centaines de kilomètres. Einstein, alors employé du bureau des brevets à Berne, voyait passer des demandes de brevets pour des systèmes de synchronisation électromagnétique des horloges. La théorie de la relativité qui en a résulté permet aujourd'hui de disposer, sur toute la Terre, d'une référence de temps, et ainsi de systèmes de repérage, tel le GPS.

Le problème de synchronisation se pose également dans les processeurs, où les signaux électriques n'ont pas le temps de se propager d'un bout à l'autre de la puce entre deux bips de l'horloge. L'horloge d'un ordinateur à 2 gigahertz a une période de 0,5 nanoseconde (un demi milliardième de seconde), c'est-à-dire qu'en une période, les signaux au sein des puces ne parcourent que quelques millimètres, alors que la largeur des puces est couramment de l'ordre de deux centimètres. Tout comme la simultanéité, l'ordre dans le temps de deux événements n'est clairement défini que dans le cas où les événements se produisent au même endroit. Pour des événements *A* et *B* distants dans l'espace, l'ordre temporel « *A* avant *B* » n'est définissable que si un signal peut

voyager de *A* vers *B*, car dans ce cas la réception du signal est toujours postérieure à son émission. On dit encore que *A* et *B* sont en relation causale. Comme les signaux se propagent à vitesse finie, tous les couples d'événements ne satisfont pas nécessairement cette condition. Les couples d'événements qui ne peuvent pas être reliés de cette façon ne sont pas ordonnés dans le temps, et ne sont donc pas liés causalement.

Ainsi, au sein d'une puce, il n'existe pas d'ordre temporel total pour tous les événements qui s'y produisent. Dans les circuits synchrones, l'horloge maîtresse distribue un temps périodique à tous les endroits de la puce, selon un arbre dont toutes les branches sont parcourues en des temps approximativement égaux. Les événements du calcul sont ainsi sélectionnés de façon telle que leur ordre temporel et, par conséquent, leur relation causale sont définis sans ambiguïté au niveau global.

En revanche, dans les circuits asynchrones, l'activité d'un opérateur logique est déclenchée par l'arrivée de données sur ses entrées. Cela ne permet de définir un ordre temporel que pour une partie des événements du calcul seulement. Un calcul dépend en général de l'ordre global dans lequel les opérations s'effectuent. Or, dans certains cas, l'ordre relatif où arrivent les données sur différents opérateurs peut dépendre des durées de propagation. Afin de maîtriser le calcul effectué, une méthode consiste à restreindre les calculs possibles et à concevoir la puce de façon telle que ne subsistent que des calculs indépendants des retards dus aux propagations (on qualifie alors le calcul de « DI », c'est-à-dire insensible au délai). Certes, les performances en vitesse dépendent de ces retards, mais le résultat du calcul, lui, n'en dépend pas. La condition DI conduit à un ensemble de contraintes logiques, dites règles de Udding, qui restreignent le comportement des opérateurs logiques élémentaires (comme le *Rendez-vous* ou l'*Arbitre*). En fait, ces règles traduisent de façon logique l'existence ou l'absence d'une relation causale physique entre les événements du calcul.

La théorie de la relativité et les circuits asynchrones semblent deux domaines bien éloignés. Pourtant, ils se rejoignent sur une même question centrale : celle de l'existence d'une référence de temps et d'un ordre temporel. Cette convergence permet de mieux cerner les problèmes liés au calcul asynchrone, et d'envisager de concevoir des puces encore plus performantes. En retour, la physique doit aussi y gagner : la forte miniaturisation des circuits logiques (l'épaisseur d'une grille de transistor se rapproche des distances interatomiques) ne rend-elle pas indispensable de maîtriser l'espace-temps jusqu'au domaine microscopique ?

Philippe MATHERAT, Laboratoire traitement et communication de l'information, CNRS, École nationale supérieure des télécommunications et Marc-Thierry JAEKEL, Laboratoire de physique théorique, CNRS, École normale supérieure