

Thèse de JDN

Retournant à l'EPFL en 1978 après le Collège de l'Elysée, le premier objectif de JD était de faire une thèse, étape indispensable pour une promotion. Le sujet s'est dégagé facilement suite à la grande question qui se posait encore pour les ordis et pour les calculatrices : faut-il calculer en décimal (code BCD ou XS-3) ou en binaire ?

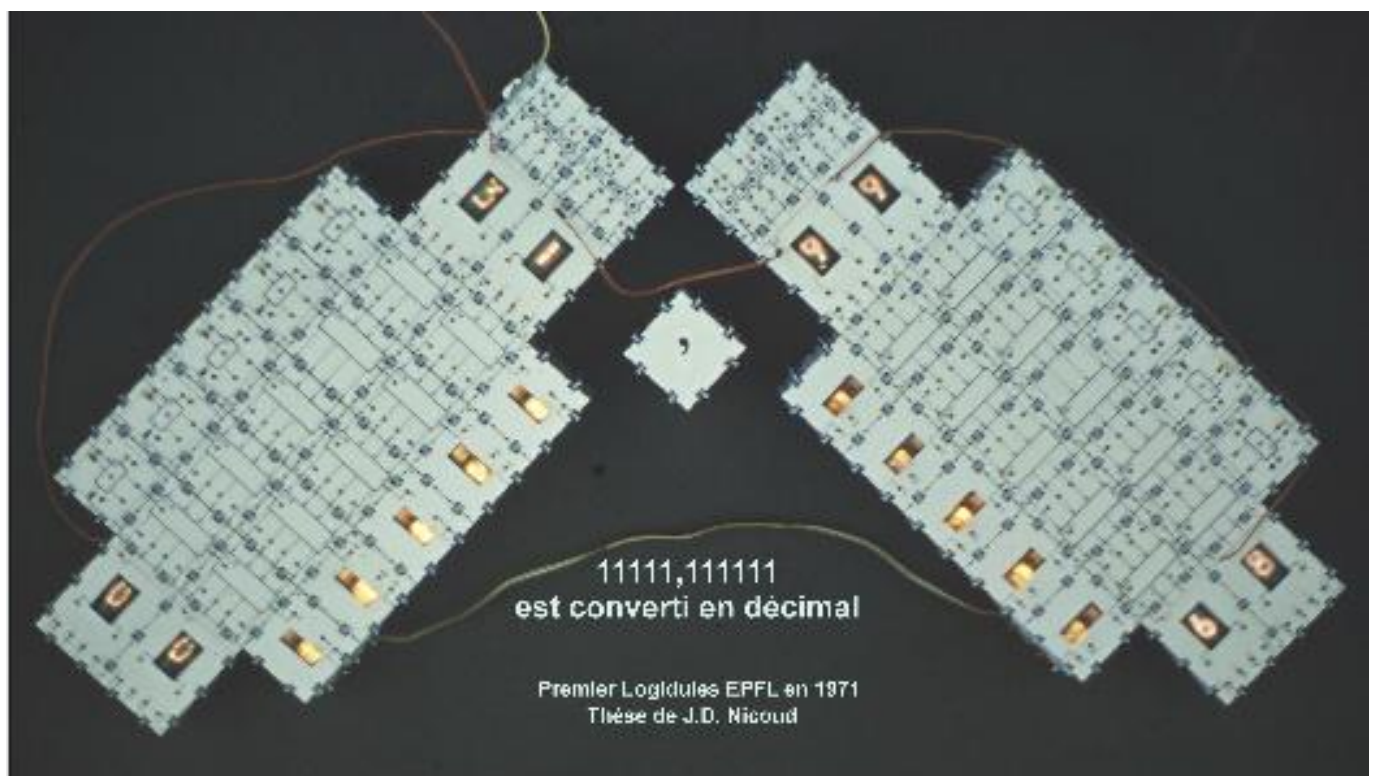
Les projets de calculatrice BIM et DEM étaient la bonne occasion pour optimiser les schémas de calcul et publier des articles courts dans les Electronics Newsletter. Leur acceptation montrait que le travail était original et pour la conversion de binaire en décimal et inverse, pour des nombres entiers ou fractionnaires, l'étude des règles de conversion conduisait à deux algorithmes, et la réalisation pouvait se faire avec des réseaux itératifs utilisant deux types de cellules. La sérialisation conduisait à différents schémas élégants.

Les boîtiers Hopt avec 4 contacts par face ont été utilisés comme démonstrateurs, c'est le début de l'histoire des logidules. Ces boîtiers ont été simulés par des élèves du collège de l'Elysée, ce qui faisait bien apparaître les temps de propagation à travers les cellules, et parfois des corrections d'erreur que l'électronique n'a jamais su bien maîtriser.

Ces résultats ont été publiés dans les Cahiers de la CSL sous forme d'un article de 15 pages fort bien présenté. Le prof Daniel Mange avait obtenu un crédit non négligeable pour publier et diffuser les travaux de recherche de son groupe. Un graphiste faisait les figures (collage de chaque élément de dessin à la cire) et ne disposait que d'une machine à écrire IBM à boule. Sans comité de lecture, les articles étaient publiés en moins de 2 mois.

La thèse présentée était un classeur avec quelques textes de liaison entre les articles publiés et Daniel Mange, qui savait éplucher les règlements, a obtenu pour moi le droit d'imprimer un extrait de la thèse, formé donc de deux articles publiés dans les cahiers de la CSL. Frais et délais minimisés à l'extrême, mais cela a quand même coûté 2100 francs – le salaire d'un mois !

En résumé, cette thèse s'est étalée sur 18 mois, avec quantités d'activités en parallèles, pour s'occuper des développements des calculatrices BIM et DEM, de l'ordinateur série des nombreux projets d'étudiant. Un article de 11 pages dans la revue la plus sérieuse IEEE Computer a été accepté et publié en avril 1974, après le délai de relecture habituel de 10 mois.



L'algorithme pour convertir de binaire en décimal se fait par addition des puissances de 2 correspondants à la position.

L'algorithme itératif est une suite de multiplications et d'additions, en commençant par les poids forts. Il se laisse plus facilement réaliser, car il suffit d'itérer la même opération.

$$N = 101100_{\text{base } 2} = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2 + 0 = 32 + 8 + 4 = 44_{\text{base } 10}$$

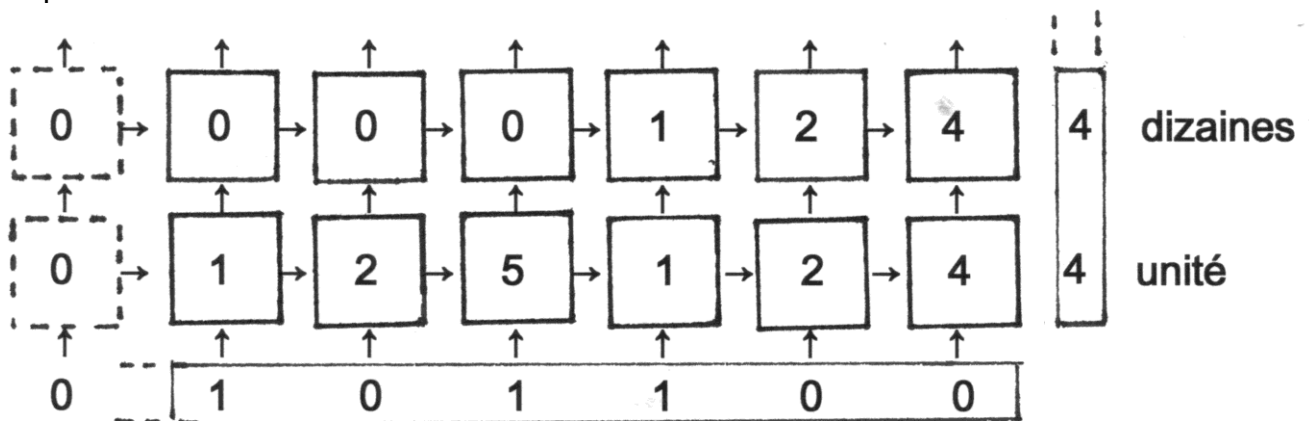
$$101100_{\text{base } 2} = (((((1 \cdot 2 + 0) \cdot 2 + 1) \cdot 2 + 1) \cdot 2 + 0) \cdot 2 + 0) = 44_{\text{base } 10}$$

$\begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array}$
 $\begin{array}{c} = 2 \\ = 5 \\ = 11 \\ = 22 \\ = 44 \end{array}$

On voit que chaque accolade multiplie par 2 le résultat précédent et ajoute le chiffre correspondant, ce qui peut s'écrire :

$$\begin{array}{ccccccc} 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ (((((0 \times 2 + 0) \times 2 + 1) \times 2 + 0) \times 2 + 1) \times 2 + 1) \times 2 + 0) \times 2 + 0) \\ 0 & 1 & 2 & 5 & 11 & 22 & 44 \end{array}$$

La même structure s'applique pour les dizaines, ce qui se traduit par des blocs fonctionnels identiques :



Ce qui est vraiment intéressant, c'est que chaque bloc effectue une opération arithmétique que l'on peut confier à un système logique. Il y a un digit (4 bits) en entrée et en sortie, l'effet des chiffres binaires (bits) se propage verticalement. Cela a conduit à la réalisation en logidules de la page précédente.

Mais la partie droite du montage est erronée ! Convertir en décimal codé binaire est très compliqué (BCD, Excess-3, Gray). Il suffit de voir que 0,001 binaire (1/8) s'écrit 0.125, donc 0.0001 0010 0101 en BCD. Convertir en hexadécimal serait trivial.

Pour les nombres positifs, cette [vidéo](#) a été faite en demandant à des élèves d'appliquer la règle des cellules de conversion. Par rapport aux logidules, la disposition des élèves est inversée; les poids forts sont en haut. Avec un registre à décalage, la 2^e [vidéo](#) montre que l'on peut pousser les bits dans moins de cellules de conversions.