

Fonctions et circuits booléens en architectures des ordinateurs

I. Contexte [NSIP 23]

Définition 1 L'Architecture de Von Neumann modélise un ordinateur comme une mémoire et une unité centrale de traitement, le programme et les données étant stockées dans la mémoire.

Définition 2 L'unité centrale de traitement consiste en une unité de contrôle, chargée d'exécuter les instructions machines, et une unité de traitement, qui applique les effets des instructions.

Définition 3 Un jeu d'instructions ou architecture, est l'ensemble des opérations qu'un processeur peut exécuter.

Définition 4 Un langage d'assemblage ou assembleur est un ensemble d'instruction ou pseudo-instruction .

Exemple 5 Un jeu d'instructions simple utilisant une mémoire pourrait être composé des 3 instructions écrites en langages d'assemblage : add, store et load.

Définition 6 Une microarchitecture l'implémentation matérielle d'une unité centrale de traitement d'un jeu d'instructions.

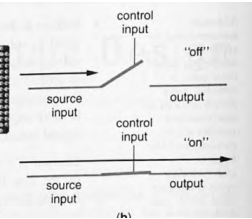
II. Circuits Combinatoires [VAH 2]

A. Circuits matériels

Définition 7 L'interrupteur laisse passer le courant si et seulement si l'entrée de contrôle est à « on ».

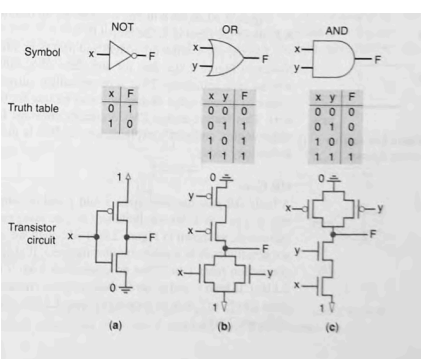
Histoire 8 Miniaturisation interrupt- **Schémas 9** [VAH Fig 2.3] teurs

- 1930s Relays
- 1940s Tubes à vide
- 1950s Transistors indépendants
- 1960s Circuits intégrés (CMOS)

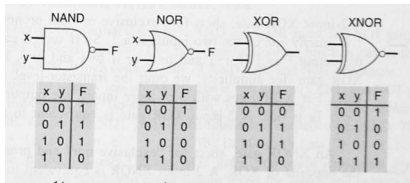


Idée 10 Pour « calculer », on va pouvoir implémenter des fonctions, avec des fils d'entrée et des fils de sortie avec des interrupteurs connectés entre eux.

Shéma 11 Portes NOT, OR, AND [VAH fig 2.8] On peut par exemple implémenter les fonctions NOT, OR et AND :

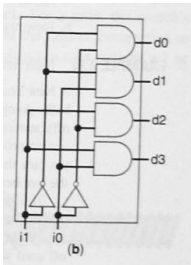


Remarque 12 Autres portes On peut de la même façon implémenter d'autres portes logiques : NAND, OR, XOR, XNOR.



[VAH fig 2.45]

Définition 13 Décodeur Un décodeur prend un nombre binaire b à n bit en entrée et met uniquement sa b -ième sortie sur 2^n à 1, décodant ainsi le nombre b .



Définition 14 Multiplexeur Un multiplexeur prend M entrées, et 1 sortie, et ne laisse passer qu'une seule des entrées vers la sortie. Un ensemble d'entrées additionnels, les «lecteurs», déterminent quelle entrée faire passer.

Remarque 15 Un multiplexeur avec n sélecteur peut traiter jusqu'à 2^n entrées.

Définition 16 Démultiplexeurs et encodeurs On peut également construire les circuits inverse des multiplexeurs et des décodeurs.

Implémentation 17 Construction d'un additionneur à n bits [PAT A.6] À l'aide de portes logiques, on peut construire un circuit qui calcule l'addition de deux nombres écrits en binaire sur n bits en $\mathcal{O}(\log n)$ traversées de portes.

B. Simplification de fonctions booléennes [NSIP Chap 22.2]

Définition 18 **Fonction booléenne** Une fonction booléenne est une fonction qui prend en entrée un ou plusieurs *bits* et qui produit en résultat un unique *bit*.

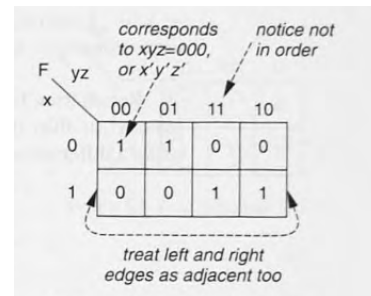
Définition 19 La **table de vérité** d'une fonction à n *bits* en entrée est une table de 2^n lignes correspondant aux résultats de la fonction sur les 2^n combinaisons possibles des entrées.

Exemple 20 Le schéma 9 et la remarque 10 montrent les tables de vérités des fonctions booléennes NOT, AND, ...

[VAH Chap 6.2]

Méthode 21 **Tableaux de Karnaugh** **Exemple 22**

Un tableau de Karnaugh est une méthode visuelle pour minimiser les fonctions booléennes à 2 jusqu'à 4 variables en regroupant les « 1 » et les « 0 » d'un tableau à deux dimensions.



Remarque 23 L'algorithme de Quine-McCluskey, lorsque l'on a plus que 4 variables, généralise le principe de regroupements des cas, ce qui est impossible avec un tableau de Karnaugh à deux dimensions.

Définition 24 Un **circuit combinatoire** n'exécute qu'un seul calcul, celui d'une fonction booléenne.

III. Circuits Séquentiels [VAH 3]

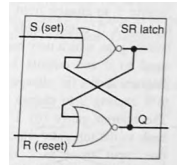
A. Comment stocker 1 bit

Définition 25 Un **circuit Séquentiel** est un circuit dont les sorties ne dépendent pas seulement des entrées, mais aussi d'un **état courant**, qui correspond à toutes les informations enregistrées en mémoire.

L'état dépend lui de la séquence des valeurs qu'ont prises les entrées du circuit.

Définition 26 **Bascule SR** Une bascule **Schéma 27** [VAH fig 3.4]

RS est un circuit asynchrone à une sortie et deux entrées, Reset et Set, définissant la sortie à 0 et 1 respectivement.



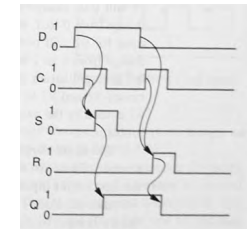
Définition 28 **Horloge** Une horloge est un circuit produisant un signal alternant entre les valeurs 0 et 1.

Définition 29 **Circuits Synchrones/Asynchrones** Un circuit séquentiel dont la mémoire ne peut changer que lorsque le signal d'horloge est à 1 est dit **synchrone**. Un circuit séquentiel qui n'utilise pas d'horloge est en revanche appelé **asynchrone**.

Définition 30 Une **Bascule D** est un circuit à une entrée et une sortie connectée à une horloge, tel que la valeur de sortie devient la valeur d'entrée à chaque période de l'horloge.

Définition 31 Un **chronogramme d'un circuit** est **Exemple 32** **Chronogramme d'une bascule D**

une courbe représentant l'évolution de la valeur booléenne sur certains fils du circuit en fonction du temps.



Remarque 33 On a donc réussi à construire un circuit séquentiel, la sortie d'une bascule D dépend de son état précédent.

On peut se servir de ces bascules pour stocker de l'information, et donc construire des mémoires.

B. Mémoire et Registres

Définition 34 Un **registre** est un composant séquentiel pouvant stocker plusieurs bits. On peut construire un registre basique en utilisant plusieurs bascules D.

Définition 35 **RAM** [NSIP 23.1] La **mémoire vive**, ou **RAM** (*Random Access Memory*) est un circuit stockant des données

tant qu'il est alimenté. Il prend en entrée une adresse et un signal (lecture/écriture), et en sortie la valeur à cette adresse.

Définition 36 [PAT A-9] La SRAM stocke les données statiquement dans des bascules D. Les données sont conservés tant qu'un courant est appliqué.

Définition 37 La DRAM stocke les données dynamiquement à l'aide de condensateur et d'un transistor par bit. Elle est donc plus dense et moins chère que la SRAM, mais doit être régulièrement *rafraîchie* à cause de la décharge des condensateurs.

IV. Processeur

A. Machines de Moore et Mealy [VAH 6.3 index Mealy]

Définition 38 Une machine de Moore (resp. de Mealy) est un transducteur fini (automate fini avec sortie) déterministe tel que, durant la lecture d'un mot d'entrée, chaque état (resp. transition) visité produit une lettre en sortie. Pour tout mot d'entrée, on obtient un mot de sortie de la même taille.

Implémentation 39 On peut implémenter une machine de Moore ou de Mealy à l'aide d'un circuit séquentiel, en utilisant un registre pour stocker l'état courant de l'automate, et des portes logiques pour réaliser la fonction de transition et la sortie de la machine.

B. Unités de contrôle et de traitement [VAH 8]

Définition 40 L'unité de traitement sert à réaliser un calcul, celui des différentes instructions. Ce dernier se fait en trois étapes :

- Charger des données
- Transformer ces données
- Enregistrer ces données

c'est souvent une unité arithmétique et logique (ALU).

Définition 41 L'unité de contrôle sert à diriger les calculs faits par l'unité de traitement, c'est une machine de Mealy ou de Moore. Cela se fait généralement en trois étapes :

- Récupérer

- Décoder l'instruction
- Exécuter l'instruction dans l'unité de traitement

DEV **Implémentation 42** [VAH 8.3] Unité de traitement pour un processeur à trois instructions

Implémentation 43 Unité de contrôle pour un processeur à trois instructions

C. Performances [PAT 1.6]

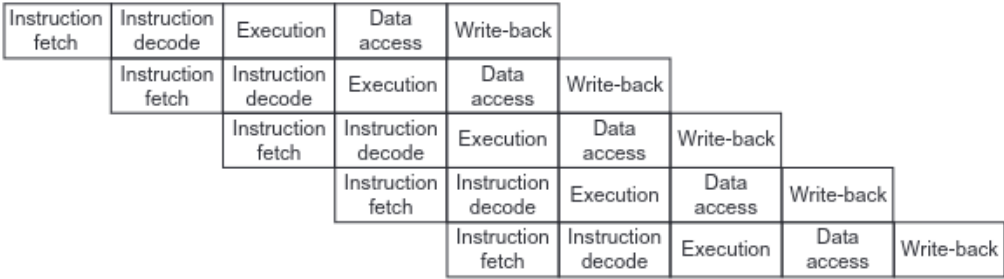
Définition 44 Le chemin critique dans un circuit est un chemin entre une entrée et une sortie passant par le plus de portes. Il détermine la période minimale de l'horloge.

Définition 45 Cycles Par Instruction (CPI) est le nombre moyen de tours d'horloge nécessaire à la réalisation d'une instruction.

Proposition 46 La performance d'un processeur est mesurée en nombre d'instructions par secondes et donc calculée par $f_{\max} \times \text{CPI}$ avec f_{\max} la fréquence maximale de son horloge.

Définition 47 Un pipeline [PAT 1.8] est une technique permettant d'augmenter le CPI d'un processeur sans en diminuer la fréquence en superposant les étapes de contrôle et de traitement des instructions.

Exemple 48 [PAT fig 4.42]



Remarque 49 Aléas [PAT Chap 4.8] Les dépendances de données entre les calculs successifs sont des obstacles de l'exécution pipeline.

Fonctions et circuits booléens en architectures des ordinateurs	
I. Contexte [NSIP 23]	
1 Def L'Architecture de Von Neumann	11 Shéma Portes NOT, OR, AND [VAH fig 2.8]
2 Def L'unité centrale de traitement	12 Rem Autres portes
3 Def Un jeu d'instructions	13 Def Décodeur
4 Def Un langage d'assemblage	
5 Ex	
6 Def Une microarchitecture	
II. Circuits Combinatoires [VAH 2]	
A. Circuits matériels	
7 Def L'interrupteur	14 Def Multiplexeur
8 Histoire Miniaturisation interrupteurs	15 Rem
9 Schémas [VAH Fig 2.3]	16 Def Démultiplexeurs et encodeurs
10 Idée	17 Implem Construction d'un additionneur à n bits [PAT A.6]
B. Simplification de fonctions booléennes [NSIP Chap 22.2]	
18 Def Fonction booléenne	26 Def Bascule SR
19 Def La table de vérité	27 Schéma [VAH fig 3.4]
20 Ex	28 Def Horloge
21 Métho Tableaux de Karnaugh	29 Def Circuits Synchrones/Asynchrones
22 Ex	30 Def Une Bascule D
23 Rem L'algorithme de Quine-McCluskey,	31 Def Un chronogramme
24 Def Un circuit combinatoire	32 Ex Chronogramme d'une bascule D
III. Circuits Séquentiels [VAH 3]	
A. Comment stocker 1 bit	
25 Def Un circuit Séquentiel	33 Rem
	B. Mémoire et Registres
	34 Def Un registre
	35 Def RAM [NSIP 23.1]
36 Def [PAT A-9] La SRAM	42 Implem [VAH 8.3] Unité de traitement pour un processeur à trois instructions
37 Def La DRAM	43 Implem Unité de contrôle pour un processeur à trois instructions
IV. Processeur [NAN]	
A. Machines de Moore et Mealy [VAH 6.3 index Mealy]	
38 Def Une machine de Moore (resp. de Mealy)	C. Performances [PAT 1.6]
39 Implem	44 Def Le chemin critique
	45 Def Cycles Par Instruction (CPI)
	46 Prop La performance d'un processeur
	47 Def Un pipeline [PAT 1.8]
	48 Ex [PAT fig 4.42]
B. Unités de contrôle et de traitement [VAH 8]	
40 Def L'unité de traitement	49 Rem Aléas [PAT Chap 4.8]
41 Def L'unité de contrôle	

Remarque

La dernière sous-partie sur les performances peut être étendue en ajoutant des notions sur [PAT 1.7 (the power wall)] ou [PAT 1.8 (L'évolution des performances depuis les années 80)] pour combler facilement la dernière page / meubler à l'oral.

Bibliographie

[NSIP] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen, *Numériques et Sciences Informatiques 1er*.

[VAH] F. Vahid, *Digital Design*.

[PAT] D. A. Patterson & J. L. Henessi, *Computer Organisation and Design, RISC-V Edition*.