I22 : Architecture ordinateur et Systèmes d'exploitation

Axel Coezard

Janvier 2020

Table des matières

Chapitre 1

Architecture ordinateur

1.1 Introduction

Les ordinateurs viennent du besoin de traiter l'information basée sur les nombres.

1.2 Ecriture positionnelle

def. Si on utilise n chiffres, ils sont codés par n carractères différents.

n. b. La valeur de n définie la base.

e. g. Une base b à exactement b caractères différents : (0, 1, 2, ..., b-1)

1.3 Ordres de grandeur

def. Préfixes d'ordre de grandeur basés sur les puissances de 10 :
- kilo (k) : 1,000 = 10^3
- méga (m) : 1,000,000 = 10^6
- giga (G) : 1,000,000,000 = 10^9
- téra (T) : 10^{12}
- péta (P) : 10^{15}
- exa (E) : 10^{18}
- zetta (Z) : 10^{21}
- yotta (Y) : 10^{24}

def. Préfixes d'ordre de grandeur basés sur les puissances de 2 :
- kibi (ki) : 1,024 = 2^{10}
- mibi (mi) : 1,024 * 1,024 = 2^{20}
- gibi (Gi) : 2^{30}

— Tébi (Gi) : 2^{40}
— Pébi (Pi) : 2^{50}
— Exbi (Ei) : 2^{60}

n. b. Physiquement, la réalisation matérielle de la representation des nombres en base 2 est limitée sur un nombre de bits fixe.

1.4 Arithmétique tronquée

def. Soit le nombre a qui s'écrit sur exactement N bits, si on lui ajoute 1 un bon nombre de fois, on va arriver à la valeur 11......1 avec N chiffres 1.

def. Si on lui ajoute en une fois 1, on appelle addition tronquée à gauche l'opération d'addition qui reste sur N bit.

def. Soit E l'ensemble des nombres représentables sur N bits et on le muni de cette opération d'addition tronquée à gauche. (E, +) définit un groupe commutatif cyclique d'ordre 2^N à :
— + interne : $\forall(a,b) \in E^2, c = a + b$
— Commutativité : $\forall(a,b) \in E^2, a + b = b + a$
— Associativité : $\forall(a,b,c) \in E^3, (a + b) + c = a + (b + c)$
— Element neutre 0 : $\exists e \in E, \forall a \in E, a + e = e + a = a$
— Element symétrique : $\forall a \in E, s(a) = 2^N - a$

def. Soit X la multiplication tronquée à gauche, on peut montrer que X est interne $\forall(a,b) \in E, a \times b \in E$:
— Commutativité : $\forall(a,b) \in E^2, a \times b = b \times a$
— Associativité : $\forall(a,b) \in E^2, a \times (b \times c) = (a \times b) \times c$
— Distributivité : $\forall(a,b,c) \in E^3, a \times (b + c) = a \times b + a \times c$
— Element neutre 1 : $\forall a \in E, a \times 1 = a$
— Element absorbant : $\forall a \in E, a \times 0 = 0$

def. Soit (E,+,$\times$) un anneau d'ordre 2^N. Il n'y a plus de relation d'ordre car il est cyclique.

1.5 Nombres positifs et négatifs

1.5.1 Nombres signés

def. Les nombres signés sont représentés sur N bits pour 2^N valeurs. On sépare l'ensemble en 2, une moitiée pour les positifs, l'autre moitiée pour les négatifs. Le bit de poid fort va faire la distinction entre les deux.

1.5.2 Complément logique à 1

def. On sépare l'ensemble des représentation en 2, les positifs et les négatifs, il y aura un bit de signe. Pour trouver la représentation d'un nombre négatif, on fait son complément logique.

1.5.3 Complément logique à 2

def. On sépare l'ensemble des représentation en 2, les positifs et les négatifs, on calcul l'opposé d'un nombre en prenant son complément logique à 1 et en lui ajoutant 1.

1.5.4 Nombres entiers relatifs

— Interêts : addition direct entre représentation familière.
— Inconvénients : il y a une valeur négative en plus, espace non équilibré.

1.5.5 Représentation biaisée

n. b. On va décaler l'espace de représentation via un biais.

e. g. Avec un biais de 1, on décale d'1 valeur : 0 ⇒ -1, 1 ⇒ 0, ...

n. b. On choisit un biais qui va séparer l'espace le plus équilibré (en 2). La valeur du milieu est $\frac{2^{N-1}}{2}$, le biais est $2^{N-1} - 1$.

1.6 Polynomes disjonctifs et portes logiques

n. b. Lorsqu'on a une foncton $f(x_1, x_2, ..., x_n) \Rightarrow 0, 1$, cela s'appelle une fonction booléenne. On peut donc exprimer toutes les solutions avec un tableau de verité.

n. b. On peut écrire f() sous forme de polynome disjonctif ou conjonctifs.

1.6.1 Polynome disjonctif

def. On exprime f comme un polynome booléen qui sera une somme de "et" $\Rightarrow \times$ et de "ou" $\Rightarrow +$.

n. b. Pour cela, on lit le tableau de valeur et sélectionne toutes les lignes ayant pour valeur 1 par f, et on fait le produit des variables en prenant en compte si elles sont égales à 0 ou à 1.

e. g. On obtient $f = \overline{x}\overline{y}\overline{z} + \overline{x}y\overline{z} + \overline{x}yz$

1.6.2 Circuits en portes logiques

A FAIRE

1.7 Talbes de Karnaugh

A FAIRE

1.8 Système d'exploitation

n. b. C'est un programme contenant des ressources (cpu, mémoire, stockage, périphériques). DIfférentes utilisations sont possibles comme l'écriture de programme ou utiliser un programme. L'OS va servir d'interface entre l'utilisation et les ressources.

1.8.1 Système mono-utilisation

1.8.2 Système multi-utilisation

1.8.3 Gestion de l'utilisation

1.8.4 Gestion de l'information

1.8.5 Gestion des fichiers

1.9 Stratégie de recyclage