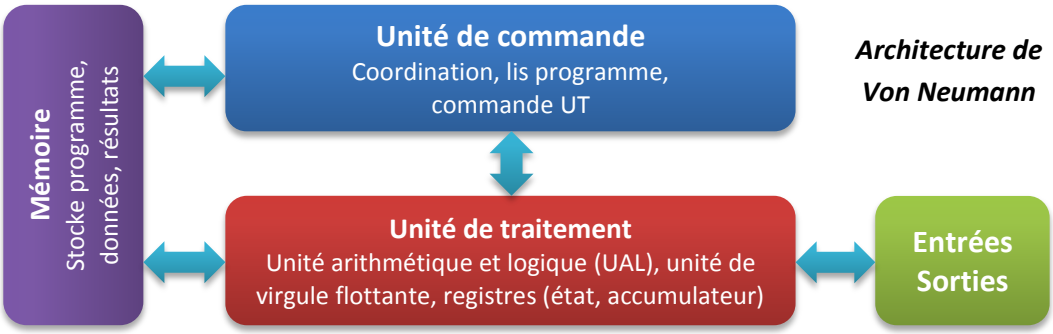
Architecture des systèmes d'exploitation

SE – Chapitre 2

I. Histoire

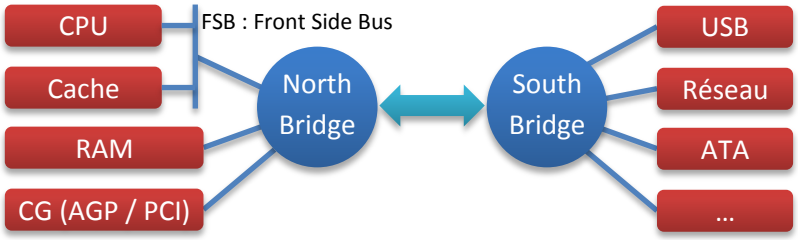
- **1623 :** Schickard – machines à calculer mécaniques
- **1943 :** ENIAC – relais, tubes à vide, imposant
- **1945 :** architecture de Von Neumann – enregistre le programme en mémoire, tiens comptes des résultats (tests, itérations, …), architecture générale, binaire, programme automodifiable.
- **1958 :** révolution du transistor (mini-ordinateurs '65, micro-ordinateurs '74, PC '81)

II. Architecture de Von Neumann et composants



Architecture de Von Neumann

- Processeur (UAL, unité de commande, mémoire cache)
- Bus (permet la communication entre divers composants par multiplexage)



III. Fonctionnement en couches

- Couche physique (0 et 1)
- Couche micro-programmée (macro-instructions)
- Couche « machine » (langage machine)
- Couche du SE (organisation mémoire, exécution des programmes)
- Couche du langage assembleur
- Couche des langages d'application

IV. Représentation de l'information (binaire)

1. Entiers naturels

$$(X)_{10} = x_0 B^0 + x_1 B^1 + \cdots + x_n B^n = (x_1 \dots x_n)_B$$

(x_i restes des divisions successives des quotients par B)

$$(23)_{10} = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^2 + 1 \times 2^0 = (10111)_2$$

2. Réels à virgule fixe

$$(X,Y)_{10} = \underbrace{x_0 B^0 + \cdots + x_n B^n}_{\text{partie entière}} + \underbrace{y_{-1} B^{-1} + y_{-2} B^{-2} + \cdots + y_{-m} B^{-m}}_{\text{partie décimale}} = (x_1 \dots x_n, \quad y_1 \dots y_m)_B$$

$0{,}7 = (?)_2$

$0{,}75 = (0{,}1\underline{0110})_2$

$0{,}7 \times 2 = 1{,}4$	1×2^{-1}	$0{,}6 \times 2 = 1{,}2$	1×2^{-4}
$0{,}4 \times 2 = 0{,}8$	0×2^{-2}	$0{,}2 \times 2 = 0{,}4$	0×2^{-5}
$0{,}8 \times 2 = 1{,}6$	1×2^{-3}	$0{,}4 \times 2 = \dots$	

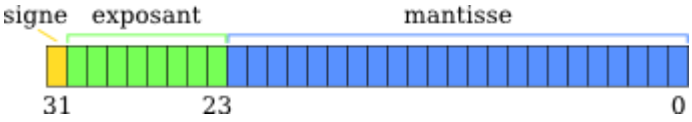
3. Entiers relatifs (avec complément à 2)

- Bit de poids fort : bit de signe (0 positif, 1 négatif).
- Passage de X à $-X$: complémentation (inversion 1 et 0) et ajout de 1.

4. Réels à virgule flottante (IEEE 754)

$$X = (-1)^S \times (1,M)_2 \times 2^{E-127} \quad 0 < E < 255$$

- M = 0, E = 255 $+\infty$
- M = 0, E = 0 0
- M ≠ 0, E = 0 NaN



$$42{,}42 = (2^5 + 2^3 + 2^1 + 0{,}42) = \underbrace{2^5}_{2^{E-127}} \left(1 + \underbrace{2^{-2} + 2^{-4} + \frac{0{,}42}{2^5}}_{0,M} \right)$$

V. Codage des caractères

- ASCII : 128 caractères. 0-31 non imprimables, 32-127 imprimables.
- ASCII étendus : 256 caractères. (latin-1, …)
- UTF-8 : sur 4 octets max (1ers bits du 1er octet indiquent nombre d'octets 1 suivis de 0).
  Ex : A 01000001 / ç **110**00011 **10**100111 / € **1110**010 **10**000010 **10**101100 / …

VI. Portes logiques

ET	OU	NON	NAND	NOR	XOR
$A.B$	$A+B$	$\bar{A}$	$\overline{A.B}$	$\overline{A+B}$	$A \oplus B$
