
Description des ordinateurs

Ordinateur?

- > Machine qui traite les nombres, mais aussi les lettres, caractères, tableaux, sons, images,...
- > Traitement des informations: processeur
- > Data processing (de 1 à une infinité de dimensions) et Image processing (image car on les voit -> 2 ou 3 dimensions uniquement)
- > Processeur et mémoire, périphériques entrée et sortie, stockage

Algorithme: traitement que la machine fait sur les données (~= pseudo code)

Programme: algorithme traduit dans un langage de programmation

Machine Turing:

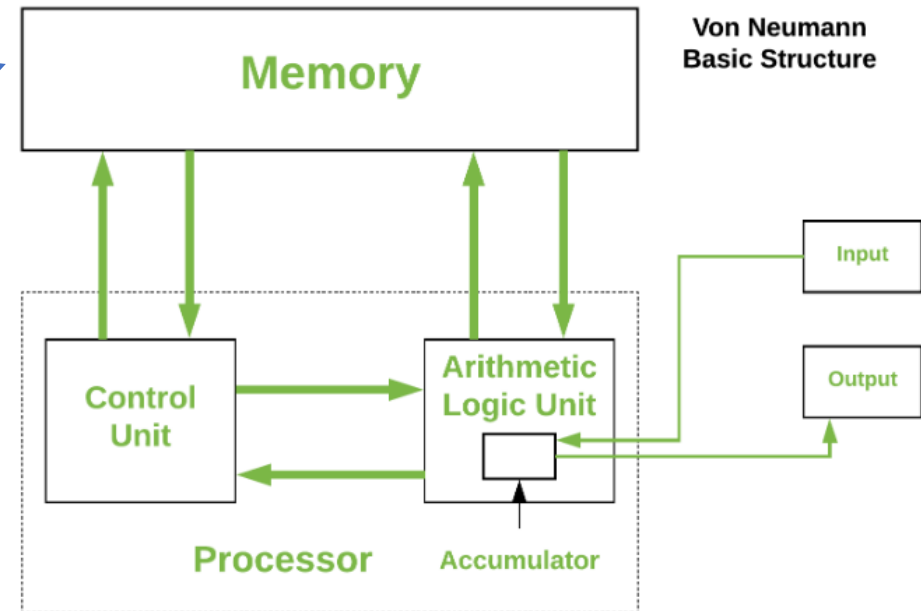
- comporte un ruban divisé en cases
- binaire
- lecture d'une case à la fois
- écriture dans une case à la fois
- elle décale le ruban d'une seule case à gauche ou droite
- comporte table d'action et table d'état

Description grossière: le ruban se déplace, l'œil reste fixe, le système lit tout le nombre de gauche à droite et, une fois la fin du nombre repéré (vide), se met à additionner le nombre de droite à gauche. La transformation de 1 en 0 fait continuer l'addition (report au chiffre de gauche) tandis que le changement de 0 en 1 en annonce la fin.

Nb: l'écriture est indispensable après lecture donc, il réécrira les chiffres tels quels après lecture.

ENIAC - Electronic Numerical Integrator, Analyser and Computer

- 1944-1946
- Entièrement électronique
- Financé par l'armée américaine
- 30 tonnes, 18000 tubes électroniques
- 1500 m²
- A servi pour les calculs de la bombe H
- Mémoire centrale permanente = organe de **rangement** des informations utilisées par le processeur;
On **exécute** un programme en le **chargeant** (=copiant) dans la mémoire centrale.
- Accumulator = registre A



Informatique :

- Désigne une discipline née avec l'**ordinateur**;
- Néologisme français datant de 1962 condensant les mots **information** et **automatique**;
- **Science de l'information**;
- Les Anglo-Saxons parlent de **Computer Science** et **data processing**

Théorie de Shannon :

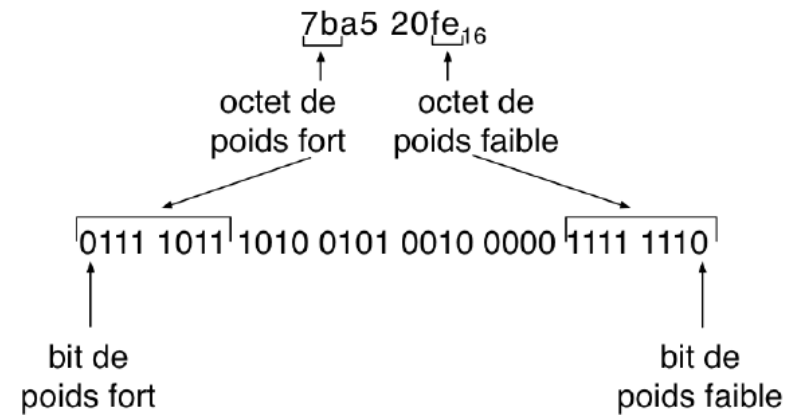
Définition: La quantité d'information d'un message représente le nombre de symboles binaires nécessaires pour représenter ce message. Ces symboles, appelés bits, correspondent aux chiffres en base 2, 0 et 1.

$\log_2 a(x)$ et a^x sont réciproques

$\log_2(a^x) = x$

On en vient donc au BIT : **B**inary digi**T**

Nombres

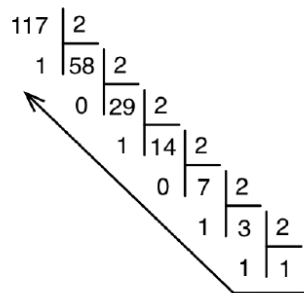
Figure 1.1

Correspondance base 2 – base 8 ou 16

Commencer le regroupement dans l'octet de poids faible
(S'il reste trop peu de bits à la fin-> ajouter 0)

101101000110 en base **8** : regrouper par **3** octets → 101**101**000**110** → 5506
 101101000110 en base **16** : regrouper par **4** octets → 101**101**000**110** → B46

Correspondance base 2 – base 10

Figure 1.2

Conversion décimal vers binaire.

Addition

$$\begin{array}{r}
 \begin{array}{ccccccc}
 1 & & 1 & & 1 & & 1 \\
 & & & 1 & & 1 & & 1 \\
 & & & & 0 & & 1 & & 0 & & 1 & & 1
 \end{array} \\
 + \begin{array}{ccccccc}
 1 & 1 & 0 & 1 & 1 & 1 & 0
 \end{array} \\
 \hline
 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1
 \end{array}$$

0 + 0 donne 0 de résultat et 0 de retenue.

0 + 1 donne 1 de résultat et 0 de retenue.

1 + 0 donne 1 de résultat et 0 de retenue.

Soustraction

$$\begin{array}{r}
 \begin{array}{ccccccc}
 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
 & & & 1 & & 1 & & 0 & & 1 & & 0
 \end{array} \\
 - \begin{array}{ccccccc}
 & 1 & 1 & 1 & 1 & 0 & 1 & 0
 \end{array} \\
 \hline
 1 & 0 & 1 & 1 & 0 & 1 & 1
 \end{array}$$

0 – 0 donne 0 de résultat et 0 de retenue.

0 – 1 donne 1 de résultat et 1 de retenue.

1 – 0 donne 1 de résultat et 0 de retenue.

1 – 1 donne 0 de résultat et 0 de retenue.

Multiplication

$$\begin{array}{r}
 \\
 \\
 \times \\
 \hline
 1 \\
 \\
 + \\
 + \\
 \hline
 1
 \end{array}$$

On reporte la première valeur autant de fois qu'on croise un 1 sur la seconde.

Les espaces remplis par un point noir sont considérés comme des 0.

On applique ensuite les règles de l'addition

Signé complément à 2:

Tout ce qui dépasse la moitié du range non signé (arrondi à l'inférieur) se déplace vers le négatif tel que:

N=3 [0,7] --> [-4,3]

Formule générale $[-2^{n-1}, 2^{n-1}-1]$

0 devant un nombre positif

1 devant un nombre négatif

0110100 +52 en CA2

1001011 en CA1

+ **1**

1001100 -52 en CA2

Calculs des nombres signés

La soustraction de deux nombres négatifs se faire **comme une addition** en nombres signés.

Si la retenue du **débordement** est **identique** à la retenue **précédente**, il n'y a **pas** d'overflow. (s'il n'y en a pas, c'est 0)

On va représenter le nombre en termes d'exposants de 2 de telle manière que l'on arrive à 1,M

Pour ce faire, diviser (ou multiplier) le nombre par 2 autant de fois que nécessaire

$$-3,5 = -1,75 \times 2^1$$

Signe négatif donc on commence par 1

Exposant binaire, on ajoute 127 à l'exposant de 2

$$1 + 127 = 128$$

Exposant réel = 1

Exposant binaire = 128

$$0,75 \times 2 = 1,5 \quad \rightarrow 2^{-1} \quad \text{On récupère ce qui suit la virgule pour le calcul suivant. } 5$$

$$0,5 \times 2 = 1,0 \quad \rightarrow 2^{-2} \quad \text{On récupère ce qui suit la virgule pour le calcul suivant. } 0$$

$$0,0 \times 2 = 0 \quad \text{On arrête}$$

1 10000000 110000000000000000000000

Signe exposant pseudo mantisse

- 128 0,5 + 0,75

$$-1, M \times 2^e = (-1 + 0, M) \times 2^e = 6(1 + 2^{-1} + 2^{-2}) \times 2^1$$

--

$$0.040625 \rightarrow 1,625 \times 2^{-2}$$

$$0,625 \times 2 \rightarrow 1,250 \quad 2^{-1}$$

$$0,25 \times 2 \rightarrow 0,50 \quad 0 \text{ devant la virgule donc rien}$$

$$0,5 \times 2 \rightarrow 1,00 \quad 2^{-3}$$

Valeur des exposants négatifs de 2:

2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
0,5	0.25	0,125	0,0625	0,03125	0.015625	0.0078125	0.00390625
1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256

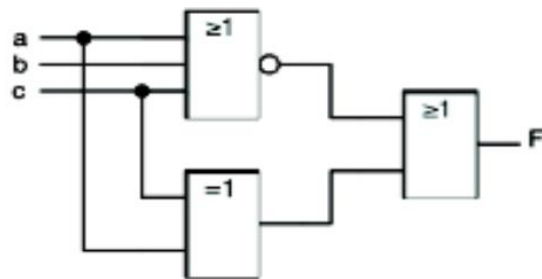
Minterme

a	b	c	F(a,b,c)
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Minterme des lignes valant 1 :

$$\bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + ab\bar{c}$$

$$\begin{aligned}
 F(a,b,c) &= \bar{a}bc + \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + a\bar{b}\bar{c} + ab\bar{c} \\
 &= \bar{a}\bar{b}\bar{c} + (\bar{a}c + a\bar{c})\bar{b} + (\bar{a}c + a\bar{c})b \\
 &= \bar{a}\bar{b}\bar{c} + (a \oplus c)(\bar{b} + b) = \bar{a}\bar{b}\bar{c} + (a \oplus c) \\
 &= \overline{a+b+c} + (a \oplus c)
 \end{aligned}$$



Circuit combinatoire équivalent à F.

Tableau de Karnaugh

a	b	c	d	$F_1(a,b,c,d)$
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

dc \ ba	00	01	11	10
00	1	1	0	0
01	0	1	1	1
11	0	1	1	0
10	0	0	0	1

On peut reprendre des éléments déjà pris dans d'autres ensembles

1e schéma

1101 peut être regroupé avec 1001 pour faire un paquet de 2

2e schéma

Les bords sont continuité des bords opposés, comme s'il s'agissait d'un tableau pouvant se déplacer horizontalement ou verticalement en se décalant, rejetant les éléments sortant du tableau vers la zone entrante. On pourra donc regrouper, ici, les 4.

Puisqu'on doit sortir un 1, on mettra toujours une variable à 0 en négatif (barre) pour assurer qu'il vaut 1.

Lorsqu'on note un paquet, on élimine d'office l'élément qui s'est déplacé à chaque annotation (car elles s'éliminent l'une l'autre)

Lorsqu'on a un paquet de 2, on élimine toujours une variable, 4 éliminera 2 variables, 8 éliminera 3 variables et 16, puisque tout sera à 1, ne laissera aucune variable car il vaudra 1

Une fois tous les tableaux effectués et les formules établies, on additionnera toutes ces formules.

Les 1 sont inscrits là où la fonction renvoie un 1 selon les données du tableau.

Pour les données du tableau, on ne change d'un bit à la fois dans la note suivante.

On ne passera donc jamais de 00 à 11!!

On fait des paquets de **2, 4, 8, 16**

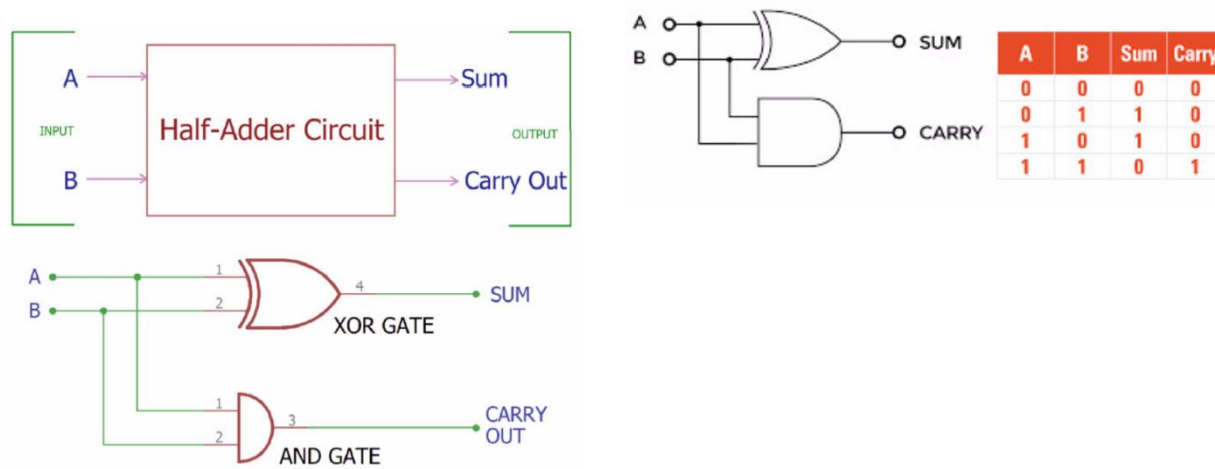
Expression algébrique du premier tableau:

$$F_1(a,b,c,d) = \bar{b} \bar{c} \bar{d} + ac + \bar{a} bc \bar{d} + \bar{a} b \bar{c} d$$

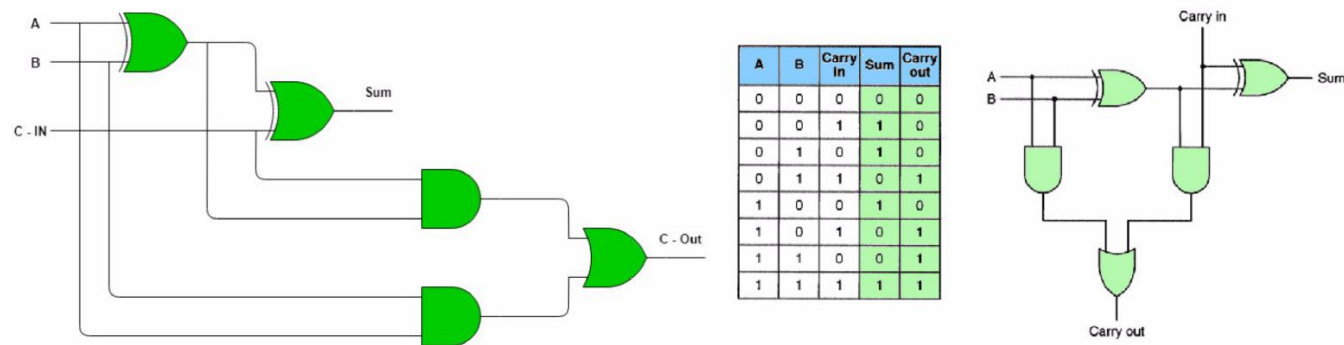
Expression algébrique du second tableau:

$$F_1(a,b,c,d) = \bar{b} c \bar{d} + bcd + \bar{a} . \bar{c}$$

Demi-additionneur



Additionneur





AND gate



OR gate



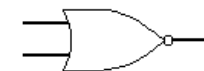
XOR gate



Inverter or NOT gate



NAND gate



NOR gate

Input 1	Input 2	Output
	1	
1		
1	1	1

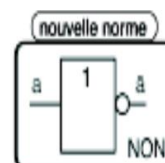
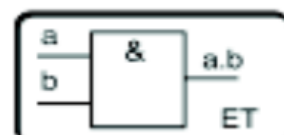
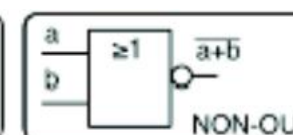
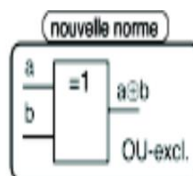
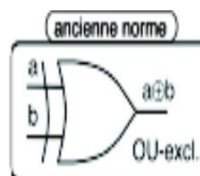
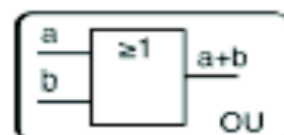
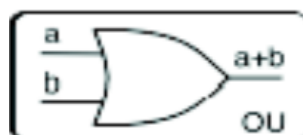
Input 1	Input 2	Output
	1	1
1		1
1	1	1

Input 1	Input 2	Output
	1	1
1		1
1	1	

Input	Output
1	
	1

Input 1	Input 2	Output
		1
	1	1
1		1
1	1	

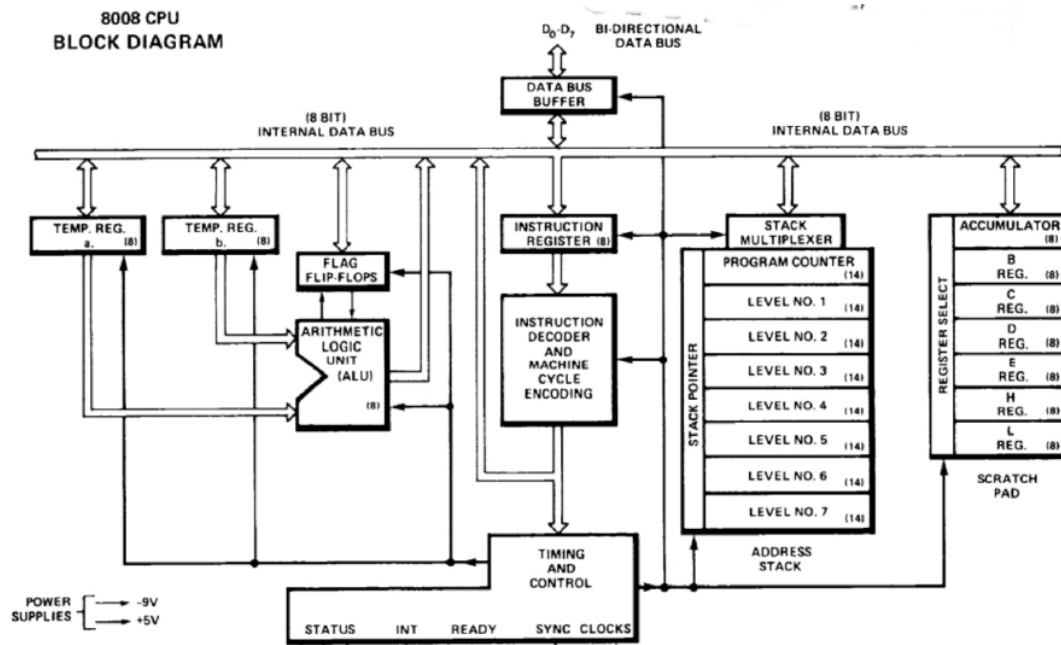
Input 1	Input 2	Output
		1
	1	
1		
1	1	



Processeurs

NOM	4004	8008	8086	8088	80186	80286	80386	80486
ANNÉE	1971	1972	1978	1979	1982		1985	
FRÉQUENCE	100 KHz	500 KHz	5-10 MHz	5 MHz	6-25 MHz	6-25 MHz	33 MHz	25-50 MHz
DATA BUS	4 bits	8 bits	16 bits	8 bits	16 bits	16 bits	32 bits	32 bits
INSTRUCTIONS BUS	4 bits	14 bits	16 bits					32 bits
ADRESS BUS			20 bits		20 bits	24 bits	32 bits	32 bits
REGISTRES	A C	ABCDEHL						
SEGMENTATION			16 seg 1MB				+ FS, GS	
MULTIPLEX			V	V	V	//- FIN -\\		
PREFETCH			6 Bytes	V	V	V	V	V
MICROCODE			6 MHz					
MÉMOIRE MAX			1 MB		1 MB	16 MB		4 GB
MODE						+ Protected	+ Virtual	
PIPELINE							3 stages	5 stages
CACHE								L1 (SRAM)

NOM	PENTIUM	PENTIUM P6	PENTIUM 4	INTEL CORE	INTEL CORE I
ANNÉE	1993	1995	2000	2006	2010
FRÉQUENCE	66 MHz	1 GHz	1.3-3.8 GHz	1-3 GHz	
DATA BUS	32 bits		32 bits	64 bits	
INSTRUCTIONS BUS	32 bits		40 bits	64 bits	
ADRESS BUSS	32 bits	36 bits			
REGISTRES		Renommés			
SEGMENTATION					
MULTIPLEX					
PREFETCH	V	V	V		V
MICROCODE					
MÉMOIRE MAX		64 GB	1 TB		
MODE					
PIPELINE		5-14 stages	20 stages		20-24 stages
CACHE	L1 : 16 KB	L2 : 256 KB		64 KB/1 MB	64/256 KB Per core
HYPER-THREADING			Simple/Dual	Simple/Dual	Quad – Octa
PCI EXPRESS					Intégré



- Flèches blanches -> BUS interne (data)
- Flèches noires: signaux envoyés (bus d'instructions)
- D₀ à D₇: les pattes de sorties du processeur
- ALU: Arithmetic logic unit
- Pile(Stack): espace mémoire réservé à la RAM où est inscrit d'où on fait un JUMP avec possibilité de retour, par exemple
- Reg. Temp: tampon registre
- Data Buffer: espace consacré au stockage les données qui arrivent plus rapidement que le BUS de sortie ne peut les extraire.

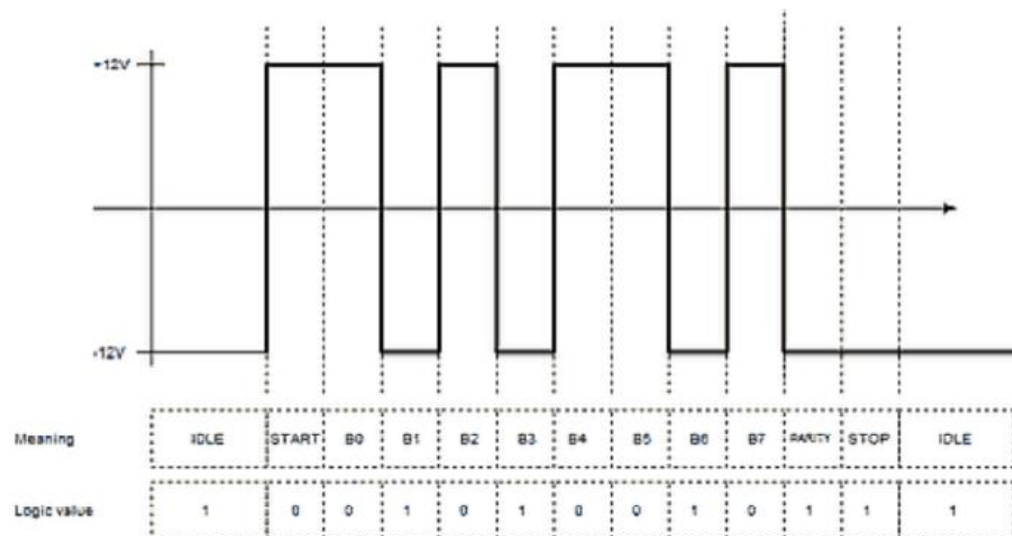
Vocabulaire

- Accumulateur (registre A) : celui dans lequel on doit mettre une des données (d'une addition par exemple).
- IP : instruction pointer. Un pointeur qui va se placer sur les lignes d'instruction à utiliser.
- SP : stack pointer. Le pointeur de pile.
- Microcode : instructions concises (set the ALU's carry input to zero; store the result value in the register 8; modification des flags,...).
- Pipeline: Le pipeline permet de lancer les instructions en décalage plutôt que les unes à la suite des autres. Il permettra donc d'effectuer plusieurs opérations **différentes** en même temps et augmentera donc le nombre d'instruction sur un temps donné.
- Segmentation : technique divisant la mémoire physique ou virtuelle en zones dotées d'une adresse et d'une taille, permettant la séparation des données et du programme dans des espaces logiquement indépendants.
 - o Rapidité d'accès ;
 - o Offre une plus grande protection des données et du code grâce aux niveau de privilège de chaque segment;
 - o Permet de placer le programme dans une partie différente de la mémoire à chaque utilisation
 ➔ CS(R): code segment DS(R): data segment SS(R): stack segment
- Cache : la mémoire cache est une mémoire intégrée au CPU. Elle est plus rapide d'accès car intégrée, contrairement à la RAM. Il y a 2 niveaux de cache :
 - o L1 qui est très rapide car à proximité extrême;
 - o L2 qui est un peu plus lent car un peu plus distant (était externe au CPU avant).
-
- BUS : structure d'interconnexion (un ensemble de lignes de communications) raccordant plusieurs circuits ou unités de fonctionnement d'un ordinateur. Les différents types de BUS sont adressage, de données et de commandes. (ex : ISA, PCI, PCIe, USB)
- MOSFET (**M**etal **O**xyde **S**emiconductor **F**ield **E**ffect **T**ransistor): transistor à effet de champs dont la grille est isolée par un semi-conducteur. Lorsque le du courant arrive par la grille, un effet de champs se forme sous la couche du semi-conducteur, attirant les électrons et créant un « canal » permettant le transfert des données entre les entre la source et le drain. Moins énergivore, conservation des données (stockage de la charge, et donc d'un bit), moins cher à la fabrication et plus petit (-> il ne faut plus qu'un transistor par bit). Aussi plus lent car les condensateurs prennent du temps à charger.
- ALU (**A**rithmetic **L**ogic **U**nit): fait les opérations -> additions, soustractions, opérations logiques, modification de registre, décalage, remise à zéro,...

Port série

- Habituellement DB9, même forme que VGA mais 9 broches ;
- Chez Cisco : fiche RJ45 coté routeur / Autres : souvent USB ;
- Souvent en version asynchrone (ne s'accorde pas avec les autres périphériques que ce soit en fréquence, quantité de data, timing) ;
- Très lent : jusqu'à 100 Kb/s, habituellement 9600 b/s ;
- Date des années '60 ;
- Certaines pattes sont devenues inutiles (car utiles pour la connexion téléphonique), on utilise encore :
 - o RX : **R**eception **D**ata ;
 - o TX : **T**ransfer **D**ata ;
 - o GND : le sol.

RS232 Transmission of the letter 'J' - 0x4A



-8N1 : 8 bits, **N**one (parity bit), 1bit de stop (toujours 1 de start);

-On envoie le byte **à l'envers**, **0** start – **1** stop
01001010 -> **0010100101** ;

-**0** en haut et **1** en bas ;

-NRZ : **N**o **R**eturn to **Z**ero : voltage toujours positif ou négatif sinon problème.

USB

- Jusqu'à 127 périphériques ;
- Plug and play ;
- Transfert des données et alimentation (max 100 mA) ;
- Compatible avec la version précédente directe uniquement ;
- 4 fils :
 - o 5 volt : alimentation
 - o Data +
 - o Data -
 - o Sol
- USB fiche C : pas de sens d'insertion.
- HID : **H**uman **I**nterface **D**evice - classes de périphériques pour identification -> table comprenant la majorité des périphériques utilisables par un utilisateur et prévoyant une majorité des opérations possibles par ces périphériques (un périphérique n'aura par exemple pas accès à toutes ses options mais il fonctionnera de manière générique)

VERSION	ANNÉE	VITESSE
1.1	1997	12 Mb/s
2.0	2000	480 Mb/s
3.0	2008	5 Gb/s
3.1	2013	10 Gb/s

Random Access Memory

Durant l'évolution de la RAM :

- La consommation n'a eu de cesse de diminuer;
- La conservation des données hors alimentation est passée d'inexistante à un temps très limité;
- La vitesse de transfert a été très fortement améliorée grâce à la technologie DDR notamment.

<i>SRAM</i>	<i>DRAM</i>	<i>SDRAM</i>
Static Access Memory	Dynamic Access Memory	Synchronous Dynamic Access Memory
RS flip-flop	Transistor et condensateur → ½ transistor = moins cher → Tps chargement + => condensateur	Synchronisée avec l'horloge CPU
Bipolaire	Conserve la charge un court instant	66 à 33 MHz // 3.3 v Voltage et consommation diminuent
1 bit conservé pour 2 transistors Pas d'énergie = pas de données (immédiat)	Besoin d'être rafraîchie (charge/ms) Adresse sous forme de matrice	2000 - DDR Technologie augmentant la quantité de data transmise par tour d'horloge

DDR SDRAM Standard	Internal rate (MHz)	Bus clock (MHz)	<u>Prefetch</u>	Data rate (MT/s)	Transfer rate (GB/s)	Voltage (V)
SDRAM	100-166	100-166	1n	100-166	0.8-1.3	3.3
DDR	133-200	133-200	2n	266-400	2.1-3.2	2.5/2.6
DDR2	133-200	266-400	4n	533-800	4.2-6.4	1.8
DDR3	133-200	533-800	8n	1066-1600	8.5-14.9	1.35/1.5
DDR4	133-200	1066-1600	8n	2133-3200	17-21.3	1.2

Le voltage minimum possible d'utilisation avec les matériaux actuels étant de 0.7v, on n'évoluera plus beaucoup en ce sens.

Mémoire SD

NOR

Lecture/écriture par bits/mots

Vitesse d'écriture faible (au moins 10x moins rapide)

Se retrouve dans le BIOS et les microcontrôleurs

Cher à la fabrication*NAND*

Lecture/écriture par blocs, pages

VNAND : stockage de plusieurs bits sur 1 cellule (FGMOS)

Se retrouve dans les cartes mémoires, SSD, SDcards

FGMOS(**peu cher**)

Cycles d'utilisation limités : système intelligent tendant à utiliser chaque cellule à tour de rôle plutôt que d'utiliser les mêmes zones

Élément de conservation de charge non-conducteur par nature

SSD : **S**olid **S**tate **D**rive

Vitesse largement supérieure à celle des HDD

➔ Jusqu'à **2000 MB/s** en lecture et **1700 MB/s** en écriture.

Interruptions

Vocabulaire :

- PIC : **P**rogrammable **I**nterrupt **C**ontroller : puce programmable et connectable en cascade (intégrée au Southbridge avec le temps).
- IDT : **I**nterrupt **D**escription **T**able : table d'adresses de 256 valeurs d'interruptions.
- ISR : **I**nterrupt **S**ervice **R**outine : routine de réaction préétablie (intégrée au Kernel de l'OS).
- IRQ : **I**nterrupt **R**equest : liste des interruptions.

PIC doit être configuré pour recevoir **IRQ** et les envoyer au CPU qui doit être configuré pour les recevoir et réagir en conséquence via l'**IDT**. Ces configurations sont préétablies dans le kernel de l'OS via l'**ISR**.

- NMI : **N**on **M**askable **I**nterrupt : Interruption non masquable (erreur d'un composant, corruption mémoire, problème d'alimentation,...)
- INTR : Masquable **I**nterrupt : Interruption masquable (logicielle)

3 types d'interruptions :

- Hardware : Clavier, carte réseau (interruption à chaque pression sur une touche)
- Software : Selon les instructions du logiciel
- CPU (sous classe de software) : division par zéro

Il y a 256 vecteurs d'erreur, les 32 premières sont réservées aux exceptions du processeur.

- FGMOS : Floating Gate MOSFET