



*Conservatoire National des Arts et Métiers
FOAD Ile De France*

Les Plateformes PC
Exercices et Travaux Pratiques
20 Juillet 2021

Note au lecteur

Ce support d'exercices est un peu particulier :

- 1) Il comporte des compléments au support de cours "Les Plateformes PC".
- 2) Il comporte une étude des principales instructions des microprocesseurs de la famille 80X86 que nous vous conseillons d'assimiler : Tout ce que peut faire un microordinateur et à fortiori tout ce que peut faire son système d'exploitation repose sur les instructions machine.
- 3) Il présente les aspects de dimensionnements et de performances de ces plateformes, votre travail consistera à comparer ces chiffres qui remontent aux PC de l'année 2018 avec les chiffres actuels.
- 4) Il comporte des questions de révision en fin de document.

Tous droits réservés.

*Ce document est un support de cours à l'usage exclusif des auditeurs du Cnam dans le cadre de leur formation.
Tout autre usage est interdit sans l'autorisation écrite du Cnam.*

SOMMAIRE

PREMIERE PARTIE	3
COMPLEMENT : LE CONCEPT D'ORDINATEUR.....	3
PERFORMANCES DES MICROPROCESSEURS	6
ETUDE DU JEU D'INSTRUCTIONS INTEL 80x86	7
DIMENSIONNEMENTS DE LA MEMOIRE CENTRALE.....	11
PERFORMANCES DE LA MEMOIRE CENTRALE.....	11
COMPLEMENTS : MEMOIRE.....	12
DIMENSIONNEMENT MEMOIRE GRAPHIQUE.....	12
DEUXIEME PARTIE	13
COMPLEMENTS : DISQUES DURS	13
PERFORMANCES DISQUES DURS.....	13
COMPLEMENTS : FORMATAGE	14
COMPLEMENTS : DISQUES SSD.....	14
COMPLEMENTS : PARTITIONNEMENT	14
COMPLEMENTS : LES SYSTEMES DE FICHIERS.....	15
COMPLEMENTS : FRAGMENTATION.....	16
COMPLEMENTS : BUS D'ENTREE / SORTIE	17
PERFORMANCES BUS D'ENTREE / SORTIE	17
COMPLEMENTS : INTERRUPTIONS	18
REVISIONS.....	20
ANNEXE A : LE CODAGE DES NOMBRES ENTIERS	21
A.1- ARITHMETIQUE EN COMPLEMENT A 1.....	21
A.2- ARITHMETIQUE EN COMPLEMENT A 2.....	22

PREMIERE PARTIE

Complément : Le Concept d'Ordinateur



Charles Babbage



Ada de Lovelace



Turing

La Machine Analytique de Babbage

Au 19ème siècle après s'être lancé dans la construction d'une "Machine Différentielle" destinée à calculer les fonctions polynomiales, le mathématicien anglais Charles Babbage comprend qu'il est inutile de gaspiller du temps et de l'argent à construire un calculateur ne sachant calculer qu'une seule fonction mathématique alors qu'il ne serait pas beaucoup plus compliqué de construire un calculateur capable de calculer n'importe quelle fonction mathématique.

Pour qu'un calculateur puisse calculer n'importe quelle fonction mathématique, il suffit :

- Qu'il dispose d'une unité de calcul : Les 4 opérations.
- Qu'il soit programmable c'est-à-dire capable de suivre un algorithme¹ : Des séquences de calculs (les 4 opérations) avec d'éventuelles ruptures de ces séquences (débranchements conditionnels).
- Qu'il dispose d'unités de stockage des programmes et des données.
- Qu'il soit capable de communiquer avec l'homme pour échanger ces programmes et ces données.

Le concept de Calculateur Programmable Universel venait de naître.

Voici la "Machine Analytique" de Babbage :

- une unité de calcul appelée "moulin" capable suivre les étapes d'un algorithme (d'exécuter les instructions d'un programme),
- une unité de mémoire "le magasin" constituée de "registres",
- des entrée/sorties : lecteur de cartes perforées², graveur de cartes perforées et imprimante,
- les programmes sont enregistrés sur cartes perforées (comme sur les machines à tisser Jacquard).

- Babbage n'a pas pu développer son projet³ le gouvernement britannique lui ayant coupé les crédits après l'abandon de la machine différentielle, cependant il avait bien compris la portée de son invention et il avait annoncé à ses confrères :

*Je suis en train de concevoir une machine capable à la fois
de calculer, de raisonner et d'écrire.*

¹ Le mot "Algorithme" vient du nom du mathématicien ouzbek Al-Khowarizmi inventeur de l'algèbre..

² Babbage s'est inspiré de la machine à tisser de Jacquard, programmable par carte perforée.

³ Les experts s'accordent à penser aujourd'hui qu'une machine purement mécanique n'aurait pas pu aboutir, qu'il fallait disposer de l'électronique.

- Pour aller jusqu'au bout de son raisonnement, Babbage confie à son assistante Ada comtesse de Lovelace⁴, le soin de développer un programme pour calculer : "Les nombres de Bernoulli". Ada va ainsi écrire le tout premier programme informatique qu'elle baptisera "Algorithme".
- Charles Babbage a été le premier lauréat de la médaille d'or de la Royal Astronomical Society en 1824.

Machines de Turing et de Von Neumann

Les premiers calculateurs programmables et plus ou moins universels, dont les célèbres "Bombes"⁵ de Turing, ont vu le jour durant la seconde guerre mondiale pour répondre à des enjeux militaires : Calcul de trajectoires et Cryptologie.

Peu après la fin de la guerre, les premiers calculateurs programmables universels apparaissent sur le marché, ils sont basés sur l'architecture dite de Von Neumann inspirée de celle des bombes de Turing, architecture toujours d'actualité :

- technologie électronique,
- arithmétique binaire,
- processeur (unité de calcul),
- mémoire (mémoire interne + mémoire périphérique),
- entrées/sorties (écran/clavier, imprimantes ...).

Les Ordinateurs

Il a fallu trouver un terme pour différencier les simples calculateurs des calculateurs universels programmables. Le terme français "Ordinateur" a été proposé en 1955 par l'écrivain Jacques Perret pour répondre à la demande d'IBM France ; devant son succès IBM accepta de mettre ce terme dans le domaine public. Les anglo-saxons quant à eux ont repris l'ancien terme de "Computeur" désignant un homme expert en calculs mathématiques.

** Attention le terme "computer" est plus général que le terme "ordinateur" il désigne un "supercalculateur" mais qui n'est pas forcément un calculateur programmable universel.*

Les Microordinateurs⁶

- En 1971 Intel lance le 1er microprocesseur : Intel-4004 (à registres de 4 bits)
- En 1976 Apple lance le 1er microordinateur personnel
- 1979 1er modem pour microordinateur : modem Hayes 110/300 bauds pour l'Apple
- En 1980 *Seagate Technologies* commercialise le premier disque dur pour microordinateur (format 5 pouces 1/4).
- En 1981 naissance de l'IBM-PC dont le système d'exploitation MS-DOS est réalisé par la jeune société Microsoft.
- En 1983 Apple lance "Lisa" le premier ordinateur personnel à interface graphique
- En 1985 Microsoft lance à son tour l'interface graphique Windows.

⁴ Mathématicienne célèbre, fille du poète Byron

⁵ Calculateurs utilisés par Turing durant la seconde guerre mondiale pour décrypter les messages de l'armée allemande chiffrés avec les machines "Enigma".

⁶ Ordinateur de faible volume dont le processeur central est un unique composant appelé "microprocesseur".

La Loi de Moore



*Dans le film de science-fiction "Terminator",
c'est en 2029 que se déclenche
la guerre entre les humains et les robots.*

D'après la loi de Moore (Ancien président de la société Intel) le nombre de composants intelligents (transistors) des puces électroniques double tous les 18 mois.

Si cela continue en 2029 les puces électroniques contiendront autant de transistors qu'il y a de neurones⁷ dans le cerveau humain soit environ 100 milliards.

*Dans les années 80 on a maîtrisé le graphisme
Avec 1 million de transistors, que sera l'informatique de
demain avec 100 milliards de transistors ?*

⁷ Il s'agit bien sûr d'une comparaison purement quantitative.

Performances des Microprocesseurs

Un microprocesseur exécute une instruction élémentaire par cycle d'horloge processeur. Une instruction élémentaire est une opération logique simple ne faisant pas appel à la mémoire, par exemple un "et logique" entre 2 registres du microprocesseur.

Un microprocesseur cadencé à 1 Ghz, exécute une instruction élémentaire en 1 nano seconde soit un milliardième de seconde.

- L'ordre de grandeur du cycle d'horloge processeur de la génération 2018 est de 3,6 GHZ, soit une instruction toutes les 0,28 ns pour un monocœur et toutes les 68 ps pour un quadricœur.
- Certaines techniques matérielles comme le "parallélisme" et le "pipeline" permettent d'augmenter les performances des microprocesseurs.

Etude du jeu d'Instructions Intel 80x86

Les processeurs de la famille Intel 80x86 possèdent 2 modes de fonctionnement : Le mode réel et le mode protégé :

- Le mode réel est obsolète aujourd'hui ; il gère la mémoire de façon linéaire (max 1Go) et ne possède pas de niveaux d'exécution hiérarchiques.
- En mode protégé, les processeurs de la famille Intel 80x86 gèrent la mémoire paginée et possèdent 4 niveaux d'exécution hiérarchiques décroissants : Ring 0 à 3. Seuls les "ring" 0 (mode noyau) et 3 (mode utilisateur) sont utilisés par les systèmes Windows et Linux.
- Seul le mode noyau (ring 0) permet d'accéder aux instructions de masquage des interruptions et d'accès aux registres des périphériques.

Une instruction machine est l'opération élémentaire que le processeur peut accomplir, une instruction est composée de deux champs :

- le code opération, représentant l'action que le processeur doit accomplir,
- le code opérande, définissant les paramètres de l'action : Il peut s'agir d'une donnée ou d'une adresse mémoire.

*Les instructions d'un ordinateur sont fondamentales
car elles illustrent tout ce que l'on peut "faire" avec cet ordinateur.*

Nous allons vous présenter ci-dessous les instructions de base incontournables de la famille Intel 80x86. Nous vous présenterons ces instructions sous leur forme "mnémonique" et non pas leur représentation binaire.

Nous nous sommes basés sur la version 32 bits du jeu d'instructions :

- Les registres de travail **EAX, EBX, EDI** etc. sont sur 32 bits, d'où le préfixe **E** (Extended) de ces registres 32 bits (les anciens registres 16 bits étaient **AX, BX, DX** ...).
- Le mot de base **DW** (Double Word) est un mot de 32 bits (Word ou **W** correspond aux anciens mots de 16 bits).

** Le nombre d'octets d'une instruction machine est variable (de 1 à 4 octets).*

Algorithmique

• Les 4 Opérations

ADD EAX, EBX	; EBX + EAX : résultat dans EAX
SUB EAX, EBX	; EBX - EAX : résultat dans EAX
IMUL EBX	; EBX * EAX : résultat dans registre double EAX-EDI
IDIV EBX	; EBX / EAX : quotient dans EAX et reste dans EDI

** Les nombres sont gérés en arithmétique en complément à 2 (cf. annexes).*

• Les Débranchements (Tests)

JZ label	; test conditionnel, débranchement ; vers le label si AX est à zéro
Label:	; label identifiant une instruction
JMP label	; débranchement inconditionnel

- Lecture / Ecriture mémoire

```
MOV EAX, [adresse] ; lecture du DW à l'adresse mémoire "adresse" dans EAX
MOV EBX, [adresse] ; idem dans le registre EBX
MOV [adresse], EAX ; écriture de EAX dans le DW à l'adresse mémoire "adresse"
MOV [adresse], EBX ; idem dans le registre EBX
```

* Le microprocesseur accède directement à sa mémoire interne (il ne s'agit pas de périphériques) .

- Affectation de valeurs constantes

```
MOV EAX, 25 ; donner au registre EAX la valeur 25
```

- Les Boucles Itératives

```
MOV ECX, 10 ; compteur de boucle
DEBUT: ; label début de boucle
...
LOOPZ DEBUT ;décrémentation du registre de boucle ECX
;et retour à DEBUT si ECX différent de 0
```

- Les Tableaux

```
MOV EAX, [TABLE+EDI] ; EDI = registre de décalage
; va correspondre à un index dans le tableau
...
TABLE: ; label début du tableau
DW <mot de 32 bits>
DW <mot de 32 bits>
DW <mot de 32 bits>
...
```

Les Sous-Programmes

Les contextes d'exécution des sous-programmes sont stockés dans la pile d'exécution, il s'agit là du mode de fonctionnement de la majorité des langages de programmation.

- Les Sous-Programmes

```
CALL TOTO ; IP+1 sauvegardé dans la pile d'exécution (adresse retour)
; IP positionné à l'adresse du label TOTO
RET ; dépile (adresse retour) et repositionne IP
```

- La Pile d'Exécution

```
PUSH EAX ; ESP = ESP - 4
; MOV [ESP], EAX
POP EAX ; MOV EAX, [ESP]
; ESP = ESP + 4
```

* **ESP** (*Extended Stack Pointer*) est le register qui contient l'adresse du sommet de la pile, comme la pile est inversée en mémoire quand elle grandit **ESP** est diminué de 4 octets et quand elle rétrécit **ESP** est augmenté de 4 octets.

Le Pilotage des Périphériques

Un équipement périphérique possède au moins 3 registres de pilotage :

- un registre de commande
- un registre d'état
- un registre de données

Il existe 3 techniques de pilotage des périphériques :

1- Par "scrutation" : On écrit une commande de transfert dans le registre de commande et on teste cycliquement le registre de statut pour savoir si le transfert a été effectué. La donnée à transférer se trouve dans un registre de données.

2- Par "interruption" : On écrit une commande de transfert dans le registre de commande et le périphérique lance une interruption (cf. ci-dessous "La Gestion des Interruptions") en fin de transfert. La donnée à transférer se trouve dans un registre de données.

3- Par Accès direct à la mémoire, il s'agit d'un composant matériel "DMA"⁸, associé au contrôleur de périphérique que l'on programme en lui spécifiant une zone mémoire centrale (et non pas des registres de périphériques). Le composant effectue le transfert dans (ou à partir de) cette zone mémoire, sans intervention du processeur et lance une interruption en fin de transfert.

** Les registres des périphériques se trouvent dans les contrôleurs des périphériques. Vus du processeur ce sont des mots mémoire identifiées par une adresse (sur les microprocesseurs 80x86 l'adressage mémoire centrale et l'adressage registres des périphériques sont disjoints).*

*** Le microprocesseur peut continuer d'accéder à la mémoire centrale durant un transfert DMA car le BUS système gère les concurrences d'accès.*

- Lecture / Ecriture registres d'entrée/sortie

❖ Ces instructions sont réservées au mode noyau.

IN EAX, adresse ; lecture registre périphérique d'adresse "adresse" dans **EAX**
OUT adresse, EAX ; écriture de **EAX** dans le registre périph. d'adresse "adresse"

⁸ Direct Memory Access

La Gestion des Interruptions

Pour pouvoir fonctionner en multitâche (interrompre la tâche courante pour passer à une autre tâche), ou pour pouvoir piloter les périphériques en mode "interruption" il faut disposer d'un mécanisme d'interruptions (cf. chapitre correspondant).

❖ *Ces instructions sont réservées au mode noyau à l'exception de **INT** qui est autorisée en mode utilisateur pour les interruptions correspondant à des appels système.*

INT num	<ul style="list-style-type: none">; lancer l'interruption logicielle numéro num; (num est sur un octet donc interruptions de 0 à 255); CS sauvegardé dans la pile système (segment retour); IP+1 sauvegardé dans la pile système (offset retour); Registre Flags sauvegardé dans la pile système; CS et IP positionnés à l'adresse de la routine d'interruption
IRET	<ul style="list-style-type: none">; retour de la routine d'interruption; restaurer les registres IP, CS et Flags
CLI	; masquer les interruptions
STI	; démasquer les interruptions

Dimensionnements de la Mémoire Centrale

On utilise de la D-RAM⁹ ou RAM dynamique, laquelle est moins coûteuse que la RAM statique (S-RAM). Sur les plateformes PC, il s'agit plus précisément de SDRAM¹⁰, car les accès du processeur à cette mémoire sont synchrones.

Sur les PC actuellement les barrettes de mémoire vive les plus répandues sont les DDR¹¹ (en version DDR5 aujourd'hui).

Les barrettes mémoire existent en 2 formats : DIMM¹² utilisée généralement pour les ordinateurs de bureau et SO-DIMM¹³ de plus petite taille utilisée généralement pour les ordinateurs portables. La variante DDRxL (L comme Light), peut fonctionner soit en tension standard DDR (1,5 v) soit en plus basse tension (1,35 v).

- L'ordre de grandeur de la taille de la mémoire vive sur les microordinateurs de la génération 2018 est de 8 GO.

Performances de la Mémoire Centrale

Le cycle d'horloge mémoire correspond à celui de l'horloge système.

Temps d'accès

Le temps d'accès est dépendant :

- de la fréquence de l'horloge système,
- de l'organisation interne de la mémoire (matrices colonnes/lignes ...),
- du protocole d'échange (commandes, adresses ...).

A titre indicatif ce temps devrait se situer entre 6 et 10 cycles d'horloge mémoire, et la moitié seulement de ce temps pour la mémoire DDR.

Par exemple avec une mémoire DDR de fréquence 400 MHz :

- cycle d'horloge mémoire : $1/400.000.000 = 2,5$ nano seconde
- temps d'accès vraisemblable du mot mémoire : entre 7,5 ns et 12,5 ns

- L'ordre de grandeur du temps d'accès mémoire pour les microordinateurs de la génération 2018 se situe entre 1ns et 5 ns.

Débit Théorique

Une autre caractéristique de la mémoire vive est son débit théorique.

*Par exemple une mémoire DDR de fréquence 400 MHz avec des mots de 8 octets on aura :
Débit théorique = $8 * 400 \text{ MHz} = 3,2 \text{ GO par seconde}$*

⁹ Random Access Memory : Sur les premiers ordinateurs on avait des bandes magnétiques comme mémoire périphérique, la lecture des bandes s'effectuait de façon séquentielle et pour atteindre une information il fallait faire défiler plus ou moins de bande ; c'est pourquoi par opposition la mémoire vive était appelée RAM (ou accès aléatoire -direct- à la mémoire).

¹⁰ Synchronous Dynamic RAM

¹¹ Les mémoires Double Data Rate sont capables d'effectuer 2 transferts d'un mot d'information par cycle d'horloge

¹² Dual in-line memory module

¹³ Small outline DIMM

Compléments : Mémoire

Extension Mémoire

Sur les machines Windows

- L'extension mémoire s'appelle "mémoire virtuelle", il s'agit d'un fichier système "**Pagefile.sys**".
- C'est l'utilitaire **System** qui permet de reconfigurer la mémoire virtuelle.

Sur les machines Linux

- L'extension mémoire correspond à une partition indépendante "**swap**".
- Pour reconfigurer cette partition il faut utiliser l'utilitaire de partitionnement "**fdisk**".

Mémoire cache disque

Sur les machines Windows

Si on "éjecte" le périphérique, il sera synchronisé avant d'être éjecté.

Sur les machines Linux

- Si on "démonte" le périphérique il sera synchronisé avant d'être démonté.
- La commande "**sync**" permet de synchroniser globalement tous les périphériques.

Dimensionnement Mémoire Graphique

- L'ordre de grandeur de la taille de la mémoire graphique sur les microordinateurs de la génération 2018 varie entre 512 MO et 2 GO.

DEUXIEME PARTIE

Compléments : Disques Durs

L'unité physique de lecture/écriture sur le disque dur est le secteur

Standards

Il y a 2 grands standards en matière de disques durs sur les microordinateurs, le standard IDE¹⁴ (contrôleur natif sur les plateformes PC) et le standard SCSI¹⁵ (issu du monde Unix, nécessite une carte contrôleur spécifique sur les plateformes PC).

Les disques durs IDE possèdent 2 standards en matière d'interface :

- ATA¹⁶ : ancien standard à transfert parallèle
- SATA¹⁷ : nouveau standard plus performant à transfert série.

Performances Disques Durs

Le temps d'accès moyen au secteur du disque est la somme de 3 temps :

- Temps de positionnement moyen des têtes sur la piste concernée.
- Temps de latence ou temps moyen écoulé entre le moment où les têtes sont positionnées sur la piste concernée et le moment où elles passent au dessus du secteur concerné (ce temps dépend de la vitesse de rotation du disque).
- Temps de lecture/écriture du secteur.

Le temps d'accès le plus important est le temps de positionnement.

- L'ordre de grandeur des temps d'accès des disques durs SATA II est de :
 - Positionnement : 7ms
 - Latence : 3ms
 - lecture/écriture secteur : 1 micro-seconde

Débit maximal

Une autre caractéristique d'un disque dur est son débit maximal, c'est-à-dire son débit théorique pour la lecture en continu, sans déplacement des têtes et sans temps de latence.

Attention : le débit maximal d'un disque dur peut être limité par le contrôleur surtout si celui-ci pilote plusieurs disques durs en parallèle.

- Le débit maximal des disques durs SATA II est de 3GB/s.

¹⁴ Integrated Drives Electronics

¹⁵ Small Computer System Interface

¹⁶ Advanced Technology Attachment

¹⁷ Serial Advanced Technology Attachment

Compléments : Formatage

Une numérotation logique des secteurs LBA¹⁸ de 1 à n sera mise en place par le "firmware" du disque dur.

Outils Linux

- Linux identifie les disques durs par un type (**sd** pour le type sata) et une lettre minuscule dans le type **a**, **b**, **c** ... Soit "**sda**" pour le premier disque dur de type sata.
- L'utilitaire Linux de formatage de bas niveau est la commande "**format**".

Outils Windows

- Windows identifie les disques durs directement par un numéro **0**, **1**, **2** ... Soit "**0**" pour le premier disque dur.
- Les utilitaires de formatage de bas niveau des disques durs IDE ne sont pas nativement disponibles sur les plateformes Windows.

Compléments : Disques SSD

Théoriquement les systèmes d'exploitation vont tenir compte des limitations des disques SSD (nombre d'écritures max) :

- pas de défragmentation automatique (cela est inutile et use le disque),
- une stratégie d'occupation des secteurs limitant les réécritures multiples des mêmes secteurs.

- L'ordre de grandeur du nombre d'écritures maximales pour un disque SSD SATA II de la génération 2018 est de :

- 100.000 pour les SSD professionnels SLC (Single Layer Cell)
- 10.000 pour les SSD MLC (Multiple Layer Cell, en fait Double Layer Cell)
- 1.000 pour les SSD TLC (Triple Layer Cell)
- 1.000 pour les SSD QLC (Quadruple Layer Cell)

- Les disques SSD-SLC sont un peu plus rapides que les disques de type MLC lesquels sont un peu plus rapides que les disques de type TLC et QLC.

Compléments : Partitionnement

Windows

- Windows peut se contenter d'une seule partition principale. Les partitions (principales et secondaires) sont considérées comme des lecteurs logiques. Les lecteurs logiques sont identifiés par une lettre (**C:**, **D:**, **E:** etc.).
- La partition principale bootable du premier disque dur, sera identifiée par la lettre "**C**" et lancera le système.
- C'est l'utilitaire "**Diskpart**" qui permet de partitionner les disques durs.

¹⁸ Logical block adresssing

Linux

- Un système Linux a besoin au minimum de 2 partitions : 1 partition système et 1 partition de swap `"/swap"` . Il est préconisé d'utiliser une partition différente pour les répertoires utilisateurs `"/home"` .
- Sur un disque dur IDE les partitions principales sont identifiées par les numéros 1 à 4 et les partitions secondaires sont numérotées à partir de 5. Soit `"sda1"` pour la première partition principale du premier disque de type sata.
- Sur un disque dur SCSI les partitions sont identifiées par les numéros 1 à 16.
- L'utilitaire Linux de partitionnement est la commande `"fdisk"`.

Compléments : Les Systèmes de Fichiers

L'unité de stockage d'un système de fichiers est le granule ou disk-block.

FAT

Le système de fichiers FAT (File Allocation Table) est caractérisé par l'utilisation d'une table d'allocation des fichiers / granules.

Dans cette table sont enregistrés de façon chaînée les granules utilisés par chaque fichier.

NTFS

Le système de fichiers NTFS utilise un système basé sur une structure **MFT** (Table de Fichiers Maître), chaque fichier est représenté par un enregistrement dans la **MFT**. La **MFT** qui contient la liste des granules du fichier, est utilisée en coordination avec le fichier **\$Bitmap** qui contient la liste des granules disponibles.

Les fichiers de petite taille (700 à 800 octets, selon la longueur du nom du fichier), sont directement stockés dans la **MFT**, en utilisant l'espace réservé pour la liste des granules. Ceci permet de limiter la perte d'espace disque par les petits fichiers.

NTFS permet aussi l'utilisation de noms longs et de faire la différence entre les noms de fichiers en majuscules et en minuscules

Pour ce qui est des performances, l'accès aux fichiers sur une partition NTFS est plus rapide que sur une partition de type FAT32.

Mais c'est au niveau de la sécurité que NTFS prend toute son importance, car il permet de gérer les attributs de protection des fichiers.

EXT

Le système de fichiers EXT est le système natif de fichiers de Linux. Il est basé sur l'**inode** qui est un bloc d'information associé à un fichier.

Chaque fichier possède son **inode** laquelle contient ses attributs ainsi que 15 pointeurs sur ses parties.

- Les 10 premiers pointeurs sont directs et peuvent pointer chacun sur un granule

- Le pointeur 13 est dit indirect de niveau 1 et peut pointer sur un bloc de 256 pointeurs directs
- Le pointeur 14 est dit indirect de niveau 2 et peut pointer sur un bloc de 256 pointeurs indirects de niveau 1
- Le pointeur 15 est dit indirect au niveau 3 et peut pointer sur un bloc de 256 pointeurs indirects de niveau 2

Ainsi la taille maximale d'un fichier géré par un système EXT avec des granules de 1024 octets serait :

$1024 * (10 + 256 + 256 ** 2 + 256 ** 3)$ soit un peu plus de 16 GO.

Outils Linux

- L'utilitaire Linux de création d'un système de fichiers à l'intérieur d'une partition est la commande "**mkfs**" (make file system).

Outils Windows

- L'utilitaire MS-DOS de création d'un système de fichiers à l'intérieur d'une partition est la commande "**Format**".

** Le formatage complet remet à 0 tous les octets de la partition, alors que le formatage dit rapide ne le fait pas.*

Compléments : Fragmentation

Les systèmes de fichiers adoptent des stratégies différentes pour limiter la fragmentation des disques.

Dos FAT

Aucune stratégie : Les fichiers sont enregistrés les uns à la suite des autres et lorsqu'il n'y plus assez de place le système fragmente.

Windows NTFS

Une stratégie que nous allons résumer en : Les fichiers sont enregistrés les uns à la suite des autres mais en ménageant des zones d'extension vides entre eux. Cela limite la fragmentation.

L'utilitaire "**defrag**" permet de mesurer le taux de fragmentation du disque et éventuellement de le défragmenter.

Linux EXT

Une stratégie que nous allons résumer en :

- Les fichiers sont enregistrés très loin les uns des autres.
- Un mécanisme "allocate-on-flush" : On conserve longtemps le fichier en mémoire cache ce qui fait que lors de l'écriture on dispose de plus d'information quant à sa taille pour mieux le positionner.

Cette stratégie évite généralement la fragmentation tant que l'on ne dépasse pas 80% d'occupation du disque.

C'est pourquoi Linux ne fournit pas d'outils natifs pour défragmenter les disques, la préconisation étant de passer à un disque plus gros lorsque l'on commence à dépasser les 80% d'occupation.

Pour mesurer le taux de fragmentation d'un fichier utiliser la commande **"filefrag"**.

Compléments : Bus d'Entrée / Sortie

Pour pouvoir fonctionner correctement alors qu'il est partagé entre plusieurs équipements chaque Bus possède un protocole de communication. En particulier le Bus système possède un arbitre qui permet de gérer les conflits d'accès entre plusieurs processeurs ou un processeur et le mécanisme DMA.

- L'horloge système et les contrôleurs et interfaces des Bus côté microprocesseur se trouvent généralement réunis dans un composant unique sur la carte mère que l'on appelle "Chipset".

Exemples :

- Bus Sata pour raccorder les disques durs
 - Bus PCI-Express¹⁹ pour les cartes d'extension PC
 - Bus USB²⁰ pour les périphériques externes
- Etc.

Performances Bus d'Entrée / Sortie

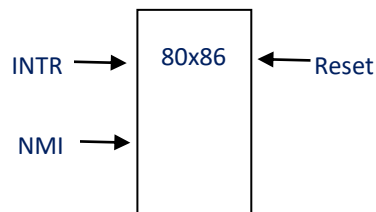
- L'ordre de grandeur du débit du Bus PCI-Express x8 (8 ports) de la génération PCIe 2.0 est de 4 GO/s.
- L'ordre de grandeur du débit d'un port USB v3 est de 4,8 GB/s.

¹⁹ Peripheral Component Interconnect, Le Bus PCI-Express est un Bus série qui se peut comporter de 1 à 32 ports

²⁰ Universal Serial *Bus*

Compléments : Interruptions

Interruptions microprocesseur 80x86



Contrôleurs d'interruption

Les interruptions périphériques sont gérées par 2 contrôleurs PIC²¹ en cascade* :

Interruption	Fonction	Priorité (1 = la plus forte)
IRQ0	Programmable Interval Timer	1
IRQ1	keyboard controller	2
IRQ2	<i>Réservée au montage en cascade</i>	*
IRQ3	8250 UART serial port COM2 and COM4	11
IRQ4	8250 UART serial port COM1 and COM3	12
IRQ5	Intel 8255 parallel port LPT2	13
IRQ6	Intel 82072A floppy disk controller	14
IRQ7	Intel 8255 parallel port LPT1	15
IRQ8	real-time clock (RTC)	3
IRQ9	IRQ9 –	4
IRQ10	IRQ10 –	5
IRQ11	IRQ11 –	6
IRQ12	Intel 8042 PS/2 mouse controller	7
IRQ13	math coprocessor	8
IRQ14	hard disk controller 1	9
IRQ15	hard disk controller 2	10

* L'ordre des priorités est historique, il provient du montage en cascade des 2 "PIC".

- L'interruption lancée par les contrôleurs PIC est dirigée vers l'entrée **INTR** du microprocesseur.
- Le microprocesseur doit acquitter une interruption pour pouvoir recevoir la suivante (instruction **EOI**²²), en cas d'interruptions en attente, les contrôleurs remontent les interruptions vers le microprocesseur dans l'ordre de leurs priorités.

²¹ Peripheral Interface Controller

²² End Of Interrupt

Table des vecteurs d'Interruption

La table des vecteurs d'interruptions contient un descripteur (ou vecteur) par interruption soit 256 descripteurs en tout (interruptions de 0 à 255).

La table des vecteurs d'interruptions est chargée en mémoire vive au démarrage du système.

Le descripteur d'interruption contient :

- l'adresse en mémoire du gestionnaire (ou routine) d'interruption.
- un indicateur masquable / non-masquable.
- le niveau d'exécution minimal autorisé pour lancer cette interruption par programmation

- Si une interruption est lancée :
 - Elle ne sera prise en compte qu'à la fin de l'instruction en cours (une interruption n'interrompt pas l'instruction courante).
 - A la fin de l'instruction en cours, si les interruptions ne sont pas masquées, le gestionnaire d'interruption correspondant est lancé :
 - 1) Stocker le contexte d'exécution dans la pile système : registres d'état et compteur ordinal.
 - 2) Mettre dans le compteur ordinal l'adresse du gestionnaire d'interruption.
 - 3) Exécuter le gestionnaire jusqu'à l'instruction retour d'interruption (**IRET**)
 - 4) Restaurer le contexte précédent à partir de la pile du système.
 - Sinon l'interruption est mémorisée dans une file d'attente.
- Si on manque d'interruptions on peut installer plusieurs **ISR** (Interrupt Service Routine) à l'intérieur d'un gestionnaire d'interruption, ils seront tous activés et chaque **ISR** vérifiera s'il est concerné avant de traiter l'interruption.

Révisions

- Nous avons vu à propos du Calculateur Programmable Universel de Babbage que pour qu'une machine soit capable de calculer n'importe quelle fonction mathématique, il faut qu'elle dispose :

- Des 4 opérations mathématiques : + - x /
- Du débranchement conditionnel (test)
- Des lectures /écritures de données en mémoire

1) Quelles instructions assembleurs 80x86 sont nécessaires pour répondre à ces fonctionnalités ?

2) Quelles instructions complémentaires pour permettre un dialogue homme machine ?

3) Quelles instructions supplémentaires pour gérer le multitâche ?

- Quel est l'ordre de grandeur du nombre d'instructions élémentaires (sans accès mémoire) par seconde pour les microprocesseurs octocœurs actuels ?
- Quelle est la différence entre la mémoire dynamique et la mémoire statique ?
- Quelle est la différence entre la mémoire vive et la mémoire morte ?
- Que représente l'antémémoire du microordinateur ?
- Que représente la mémoire virtuelle du microordinateur ?
- Que représente la mémoire cache du microordinateur ?
- Que représente un secteur du disque dur ?
- Que représente le LBA d'un disque dur ?
- Que représente un granule ou disk-block du système de fichiers ?
- En combien de partitions principales au maximum peut-on partitionner un disque DOS/IDE ?

Annexe A : Le Codage des Nombres Entiers

Il existe 2 standards de codification des nombres entiers en informatique : L'arithmétique en complément à 1 et l'arithmétique en complément à 2.

Dans les 2 cas les nombres ont une taille limitée, généralement par la taille du mot machine :

- Avec une codification sur 16 bits on peut exprimer 65.536 (2^{16}) nombres différents.
- Avec une codification sur 32 bits on peut exprimer plus de 4 milliards (2^{32}) de nombres différents.
- Avec une codification sur 64 bits on peut exprimer plus de 16 trillions²³ (2^{64}) nombres différents.

Les langages de programmation et les applications peuvent construire des surcouches de codification des nombres :

- Java gère des nombres de 64 bits même sur une plate-forme 32 bits
- Python gère de nombres entiers d'une taille potentiellement illimitée
- Une application bien écrite peut calculer le nombre π avec des millions de décimales.

A.1- Arithmétique en complément à 1

- Pour faciliter la compréhension nous utiliserons des nombres binaires de taille 4 bits -

Codification des nombres positifs :

Le premier bit est à 0. Les autres bits représentent la valeur absolue du nombre en binaire :

```
0000 = +0
0001 = +1
0010 = +2
0011 = +3
```

Codification des nombres négatifs :

Le premier bit est à 1. Les autres bits représentent le complément à 1 de la valeur absolue du nombre en binaire :

```
1111 = -0
1110 = -1
1101 = -2
1100 = -3
```

- Ce sont les 2 zéros positif et négatif qui rendent ce système peu convivial (nécessité de tester une égalité à zéro par 2 égalités : $(= 0)$ ou $(= -0)$.
- Avec des mots de 16 bits on peut exprimer les valeurs de -32767 à -0 et de 0 à 32767 .

²³ Million (Mega) = 10^6 , billion (Tera) = 10^{12} et trillion = 10^{18}

Addition en complément à 1

Pour additionner 2 nombres on effectue une addition binaire et s'il y a une retenue à gauche on ajoute 1 au résultat, par exemple :

$$\begin{array}{r} 0001 \text{ (1)} \\ 0010 \text{ (2)} \\ \hline 0011 \text{ (3)} \end{array} \quad \text{et} \quad \begin{array}{r} 1110 \text{ (-1)} \\ 1101 \text{ (-2)} \\ \hline 1011 \\ \text{Retenue à gauche on rajoute 1} \\ 1 \\ \hline 1100 \text{ (-3)} \end{array}$$

Pour effectuer une soustraction $A - B$ on additionne A et $(-B)$.

A.2- Arithmétique en complément à 2

Codification des nombres positifs :

Comme pour l'arithmétique en complément à 1, le premier bit est à 0 et les autres bits représentent la valeur absolue du nombre :

$$\begin{array}{l} 0000 = +0 \\ 0001 = +1 \\ 0010 = +2 \\ 0011 = +3 \end{array}$$

Codification des nombres négatifs :

Le premier bit est à 1. Les autres bits représentent le complément à 1 de la valeur absolue du nombre en binaire auquel on rajoute 1 et s'il y a une retenue à gauche on l'ignore.

$$\begin{array}{l} 0000 = -0 \quad (1111 +1) \text{ retenue à gauche ignorée} \\ 1111 = -1 \quad (1110 +1) \\ 1110 = -2 \quad (1101 +1) \\ 1101 = -3 \quad (1100 +1) \end{array}$$

- Les 2 zéros (+0 et -0) sont codifiés de la même manière : 0000
- Il y a un nombre positif en moins (par rapport aux négatifs). Avec des nombres de 16 bits on peut exprimer les valeurs de -32768 à 32767.

Addition en complément à 2

Pour additionner 2 nombres on effectue une addition binaire et s'il y a une retenue à gauche on l'ignore, par exemple :

$$\begin{array}{r} 0001 \text{ (1)} \\ 0010 \text{ (2)} \\ \hline 0011 \text{ (3)} \end{array} \quad \text{et} \quad \begin{array}{r} 1111 \text{ (-1)} \\ 1110 \text{ (-2)} \\ \hline 1101 \text{ (-3)} \\ \text{retenue à gauche ignorée} \end{array}$$

Pour effectuer une soustraction $A - B$ on additionne A et $(-B)$.