



Introduction

Présentation du module

Ilias TOUGUI

L'Ecole Supérieure d'Informatique et du Numérique

INF2132 - Systèmes d'Exploitation

PLAN DU COURS

- 1. Présentation du module**
- 2. Introduction**

1. Présentation du module

Bienvenue!

Coordinateur

- Pr. Ilias Tougui, PhD
- Enseignant-Chercheur a l'ESIN, UIR
- Docteur en Sciences de l'ingenieur
- IA et Santé
- Email : Ilias.tougui@uir.ac.ma

Assistant

- Pr. Yasser Aderghal
- Doctorant au TICLAB, UIR
- IA et Réseaux neuronaux graphiques
- Email : yasser-aderghal@uir.ac.ma

Bienvenue en Informatique!

Comprendre et savoir programmer des logiciels



Bienvenue en Informatique!

Comprendre et savoir programmer des logiciels



Rôle d'un système d'exploitation

Simplifier la vie de l'utilisateur

- En servant d'intermédiaire entre le matériel et les applications
- En fournissant une interface intuitive
- En gérant les ressources du système (mémoire, processeur) de manière transparente
- En assurant le multitâche, en organisant les fichiers
- En gérant les entrées/sorties (clavier, écran)
- En offrant des fonctionnalités de sécurité et de maintenance

Rôle d'un système d'exploitation

Simplifier la vie du concepteur de logiciels

- Masquer l'hétérogénéité du matériel
- Masquer les détails de mise en œuvre de bas niveau
- Offrir des abstractions pertinentes et de haut niveau pour développer des logiciels

Pourquoi apprendre le système ?

- Par curiosité : comprendre ce qui se passe sous le capot
- Pour écrire des logiciels qui utilisent efficacement les ressources
- Pour travailler dans les domaines liés aux systèmes
- Gros besoins d'ingénieurs inventifs sur les marchés nationaux et internationaux

Objectifs du module

- Introduire la notion de système d'exploitation
 - Rôle et fonctionnalités
- Introduire les principales abstractions fournies par un système
 - Processus, fichiers, communication inter-processus
- Comprendre les systèmes UNIX par l'exemple
 - À l'aide d'un langage de script (le shell bash)
- Savoir utiliser le langage bash
- Être autonome pour les autres enseignements en informatique

Organisation des Cours

10 séances d'environ 2 heures de cours couvrant les éléments suivants :

- Introduction aux Systèmes d'Exploitation
- Les Commandes UNIX de base
- Les services d'un Système d'Exploitation
- Les appels système
- Gestion des processus
- Sécurité et gestion des utilisateurs
- Systèmes de fichiers
- Gestion de la mémoire
- Programmation Shell en Bash

Organisation des Cours

11 séances de TPs:

- En salles de TP
- Sur vos machines, avec une des options suivantes:
 - Dual-boot Windows + Ubuntu
 - Windows + VM Ubuntu
 - macOS (environnement UNIX native)

Supports:

- <https://www.connect.uir.ac.ma>

Note des Exams et TPs

Note Finale:

- $NF = 25\% \text{ CC} + 25\% \text{ TPs} + 50\% \text{ CF}$
- $NFaR = \text{Max}(NF, 25\% \text{ CC} + 25\% \text{ TPs} + 50\% \text{ CFaR})$

Note des TPs: TPs (1-11)

- Note Générale TP = Moyenne des 11 TPs individuels
- Chaque TP individuel = Assiduité + Performance

Règles d'Absence par TP:

- Absence non justifiée à un TP \rightarrow Note de ce TP = 0/20
- Absence justifiée à un TP \rightarrow 2ème chance (rattrapage possible)

Attention!

N'oubliez pas de programmer, car programmer s'apprend par la pratique

2. Introduction

Introduction

Système d'exploitation

Le système d'exploitation permet d'exploiter les ressources d'un ordinateur (processeur, mémoire, carte graphique, lecteur DVD...etc.).

Les trois phases principales du démarrage d'un ordinateur:

- Boot du système
- Démarrage du système d'exploitation
- Lancement des Programmes

Introduction

Les trois phases principales du démarrage d'un ordinateur

Boot du système

1. **Initialisation du matériel** : Le BIOS vérifie les composants matériels (processeur, mémoire, disques)
2. **POST (Power-On Self Test)** : Tests automatiques pour s'assurer que le matériel fonctionne correctement
3. **Recherche du bootloader** : Le système cherche sur quel périphérique se trouve le système d'exploitation à charger

Introduction

Les trois phases principales du démarrage d'un ordinateur

Démarrage du système d'exploitation

1. **Chargement du noyau** : Le système d'exploitation (Windows, Linux, macOS, etc.) est chargé en mémoire
2. **Initialisation des pilotes** : Les drivers nécessaires pour communiquer avec le matériel sont chargés
3. **Configuration des services système** : Les services essentiels du système démarrent

Introduction

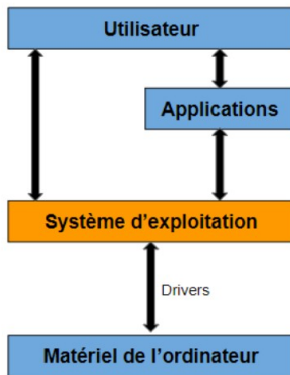
Les trois phases principales du démarrage d'un ordinateur

Lancement des Programmes

1. **Interface utilisateur** : Affichage de l'environnement graphique ou de la ligne de commande
2. **Services utilisateur** : Démarrage des applications configurées pour se lancer automatiquement
3. **Prêt à l'utilisation** : L'utilisateur peut maintenant interagir avec le système et lancer ses programmes

Système d'exploitation

Le système d'exploitation fait l'interface entre les utilisateurs et le matériel:



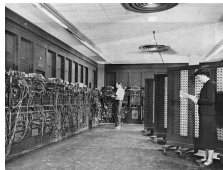
Historique

1ère génération (1945–1957) : les tubes à vide et les cartes enfichables

1946: Ordinateur ENIAC

Architecture à base de lampes et tubes à vide :

- 30 tonnes, 72 m² au sol
- 5000 additions par seconde (0,005 MIPS)
- Les tubes agissaient comme des interrupteurs électroniques où les états marche/arrêt représentaient des chiffres binaires (0 ou 1)



ENIAC: Electronic Numerical Integrator And Computer

Evolution Architecturales

1ère génération (1945–1957)

Caractéristiques:

- Pas de système d'exploitation
- ENIAC: premier ordinateur électronique numérique généraliste

Inconvénients:

- Programmé manuellement
- Grande lenteur/fragilité

Evolutions Architecturales

1ère génération (1945–1957)

Plan Matériel:

- Tubes à vide
- Mémoire limitée
- Très cher
- 1 seul espace mémoire \Rightarrow ne peut exécuter qu'un seul programme

Plan Logiciel:

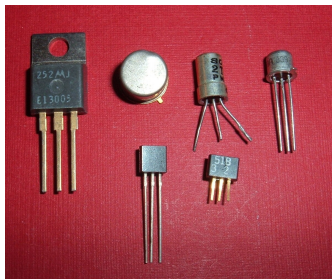
- Exploitation séquentielle:
 - Chargement
 - Exécution
 - Impression
- Langage: Machine ou assembleur

Historique

2ème génération (1958–1964) : les transistors et le traitement par lots

1947 : Invention du transistor

- Inventé par les physiciens américains John Bardeen, Walter Brattain et William Shockley aux Bell Labs.
- Remplacé les volumineux et fragiles tubes à vide électroniques
- Récompensée par le prix Nobel de physique en 1956 pour ses inventeurs



Evolution Architecturales

2ème génération (1958–1964)

Caractéristiques:

- Transistor: plus petit, moins cher, plus fiable que les tubes à vide
- Apparition des langages FORTRAN et C
- Apparition des systèmes d'exploitation (SE) : FMS (Fortran Monitor System) et IBSYS (IBM 7094)

Evolution Architecturales

2ème génération (1958–1964)

Plan Matériel:

- Apparition des transistors \Rightarrow ordinateurs plus petits
- Mémoire à tore magnétique \Rightarrow plus de mémoire disponible
- Microprogrammation \rightarrow programmation plus facile

Plan Logiciel:

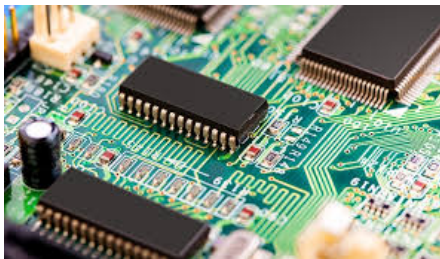
- Traitement par lots: plusieurs programmes peuvent être regroupés dans un lot et exécutés l'un après l'autre
- Langage évolué (Fortran / C)
- Contrôle de processus \Rightarrow ancêtre des systèmes d'exploitation
- Les interruptions \Rightarrow gestion du matériel plus efficace

Historique

3ème génération (1965–1971) : les circuits intégrés et la multiprogrammation

1958: Invention du circuit intégré sur silicium

- **1958:** Jack Kilby chez Texas Instruments invente le premier circuit intégré en connectant plusieurs transistors sur une puce de germanium.
- **1959:** Robert Noyce (Fondateur d'Intel) chez Fairchild Semiconductor développe le procédé "planar" permettant la fabrication de circuits intégrés monolithiques en silicium.



Evolution Architecturales

3ème génération (1965–1971)

Caractéristiques:

- Amélioration des coûts et des performances (circuits intégrés)
- Une famille d'ordinateurs compatibles entre eux
- Une architecture et un même jeu d'instructions
- Apparition de la multiprogrammation (partitionnement de la mémoire pour des tâches différentes)

Inconvénients:

- Système d'exploitation énorme et très complexe
- Structure en bus complexe

Evolutions Architecturales

3ème génération (1965–1971)

Plan Matériel:

- Circuit intégré
- Mémoire à semi-conducteur
- Mini-informatique
- Machine universelle

Plan Logiciel:

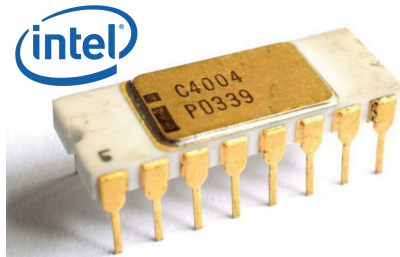
- Multiprogrammation \Rightarrow plusieurs programmes peuvent s'exécuter en même temps
- Partitionnement de la mémoire \Rightarrow isolation des programmes
- Chargement continu des programmes depuis le disque dur
- Suivi de l'exécution des programmes

Historique

4ème génération (1972–1977) : les ordinateurs personnels

1971: Processeur Intel 4004

- 2300 transistors dans un unique circuit intégré
- Fréquence de 740 kHz, 0.092 MIPS



Evolutions Architecturales

4ème génération (1972–1977)

Caractéristiques:

- Développement des circuits VLSI (Very Large Scale Integration) : plus de 1000 composants sur une seule puce de circuits intégrés
- Ils ont la même architecture que les mini-ordinateurs, mais leur prix est beaucoup moins élevé
- Il existe deux systèmes d'exploitation principaux : MS-DOS (Microsoft Inc.) et UNIX



Evolutions Architecturales

4ème génération (1972–1977)

Plan Matériel:

- Circuit intégré VLSI
- Micro-informatique
- Mémoire à très grande capacité

Plan Logiciel:

- Machine virtuelle
- Apparition des Réseaux Informatique
- Architecture distribuée
- Téléinformatique

Historique

5ème génération (1978–????) : les ordinateurs personnels portables et de poche

2011: Processeur Intel Core i7 2600K

- Plus de 1,4 milliards de transistors
- Fréquence de 3,4 GHz
- 8 cœurs, 8 threads
- 128300 MIPS

Entre le 4004 et le Core i7 2600K :

- La fréquence a été multipliée par 4600
- La puissance en MIPS a été multipliée par 1,4 million

Cœurs et Threads

Cœur (Core)

Composant physique du processeur capable d'exécuter des instructions - c'est le "hardware" qui fait le travail.

Thread

Séquence logique d'instructions envoyées au processeur - c'est la "tâche" que doit accomplir le cœur.

Un processeur avec 8 cœurs et 8 threads signifie que le processeur possède 8 unités physiques de calcul (les cœurs) et que chaque cœur ne peut traiter qu'une seule séquence d'instructions à la fois (un thread par cœur).

Evolution Architecturales

5ème génération (1978–????)

Caractéristiques:

- Apparition des clusters de calculs: plusieurs ordinateurs qui travaillent ensemble
- Apparition des PIC (Personal Intelligent Communicator de chez Sony) et des PDA (Personal Digital Assistant, comme le Newton de chez Apple) ⇒ ancêtres des smartphones
- Miniaturisation des ordinateurs
- Démocratisation de l'informatique et d'Internet
- Apparition des datacenters et du Cloud

Evolutions Architecturales

5ème génération (1978–????)

Plan Matériel:

- Machine parallèle (e.g., dual core, etc...)

Plan Logiciel:

- Architectures spécialisées (e.g., smartphones)
- Traitement parallèle

Récapitulatif

Génération	Dates approximatives	Technologie	Performance (ops/sec)
1	1946-1957	Tube à vide	40 000
2	1958-1964	Transistor	200 000
3	1965-1971	Intégration à petite et moyenne échelle	1 000 000
4	1972-1977	Intégration à grande échelle	10 000 000
5	1978-????	Intégration à très grande échelle	100 000 000

Architecture de base d'un ordinateur

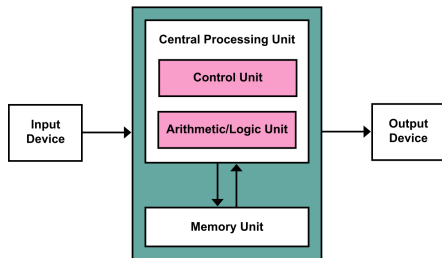
L'ordinateur doit donc posséder:

- Une ou plusieurs unités de stockage, pour mémoriser le programme en cours d'exécution ainsi que les données qu'il manipule
- Une unité de traitement permettant l'exécution des instructions du programme et des calculs sur les données qu'elles spécifient
- Différents dispositifs « périphériques » servant à interagir avec l'extérieur :
 - clavier, écran, souris, carte graphique, carte réseau, etc.

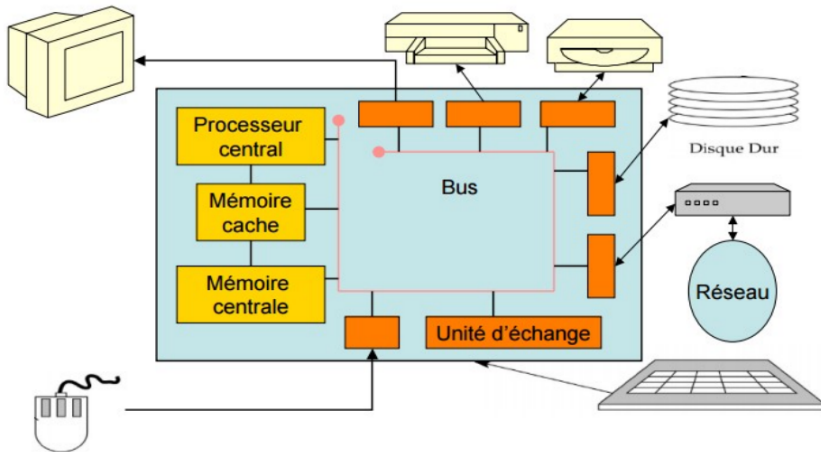
Architecture de Von Neumann

La machine de Von Neumann (EDVAC) est composée de:

1. Une mémoire principale (MP) qui contient données et instructions
2. Une unité arithmétique et logique (UAL) capable de fonctionner sur des données binaires
3. Une unité de contrôle (UC) qui interprète les instructions en mémoire et en entraîne l'exécution
4. Un dispositif d'entrée et de sortie (E, S) pris en charge par l'unité de contrôle



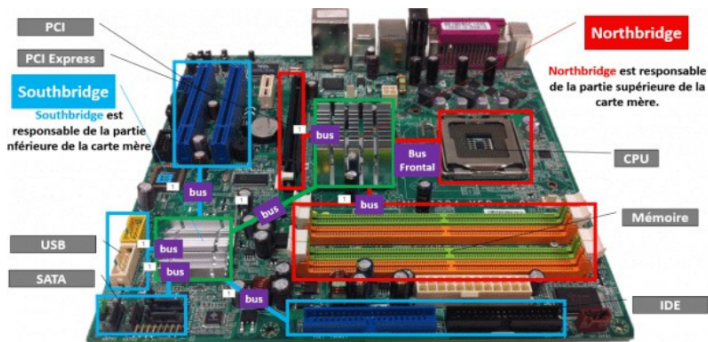
Communication entre les composants



Les bus

Les composants de l'ordinateur sont reliés par un ou plusieurs bus, ensembles de fils parallèles servant à la transmission des adresses, des données, et des signaux de contrôle

(e.g., bus PCI-express, bus USB)



Types de bus

Bus d'Adressage

Transporte les adresses mémoire (unidirectionnel) pour indiquer où lire ou écrire les données.

Bus de Données

Transporte les données elles-mêmes (bidirectionnel) entre le processeur, la mémoire et les périphériques.

Bus de Commandes

Transporte les signaux de contrôle (bidirectionnel) pour coordonner les opérations comme lecture/écriture.

Les Interruptions

Définition

Événements asynchrones provenant de périphériques externes qui interrompent temporairement l'exécution du processeur pour traiter une requête urgente. Le processeur sauvegarde son état actuel, traite l'interruption via une routine spéciale, puis reprend l'exécution normale.

Exemple

Quand vous appuyez sur une touche du clavier, le contrôleur du clavier envoie une interruption au processeur pour lui signaler qu'une nouvelle donnée est disponible

Les Exceptions

Définition

Événements synchrones générés par le processeur lui-même lors de l'exécution d'une instruction problématique, souvent liés à des erreurs de programmation. Elles empêchent l'exécution correcte de l'instruction en cours et déclenchent un traitement d'erreur.

Exemple

Division par zéro dans un programme - le processeur détecte l'erreur mathématique et génère une exception qui permet au système d'exploitation d'afficher un message d'erreur au lieu de planter

Systèmes d'exploitation



Catégories des systèmes d'exploitation

UNIX-based

- Linux (Ubuntu, Fedora, Kali, ...)
- Solaris
- Mac OS

Windows

- Vista
- Seven
- Windows 8/10/11

Mobile

- Android
- iOS
- Windows Phone

Real Time

- QNX
- VxWorks