

ASR2-Système : ordinateurs et systèmes d'exploitation

Semestre 2, année 2011-2012

Département d'Informatique
IUT Bordeaux 1

Avril 2012

Introduction aux systèmes d'exploitation

- 1 Qu'est-ce qu'un système d'exploitation ?
 - Définition
 - Intérêt
 - Évolution
- 2 Le premier ordinateur : SSEM (Manchester, 1948)
 - SSEM : éléments
 - Cycle de fonctionnement
 - Jeu d'instructions du SSEM
 - Programmation
 - Et ensuite ?
 - Evolution des technologies (mémoire)
 - Evolution des technologies (processeurs)
- 3 Architecture des ordinateurs
 - Exemple d'architecture : Intel 8080
- 4 Les générations d'ordinateurs
- 5 Évolution des usages
 - Un programme à la fois

Qu'est-ce qu'un système d'exploitation ?

Système d'exploitation

- programme qui permet d'exploiter les ressources matérielles
- fournit **au programmeur d'application** un environnement avec facilité d'emploi et utilisation efficace des ressources.

Facilité d'emploi

Le système d'exploitation cache les détails matériels sous une **couche d'abstraction**.

Il fournit une API (Application Programming Interface) pour utiliser les ressources.

Exemples

- communication réseau par différents moyens
- utilisation de fichiers sur différents supports, à travers le réseau, etc.
- ...

Utilisation des ressources

Le système d'exploitation est un **gestionnaire de ressources** ; il gère l'allocation

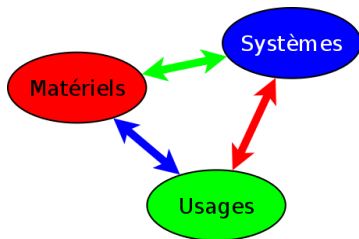
- des processeurs,
- de la mémoire,
- des périphériques
- etc.

Attention, ne pas confondre système d'exploitation et interface utilisateur !

Évolution des systèmes d'exploitation

L'histoire des systèmes d'exploitation est liée à

- l'évolution des matériels
- la manière de les utiliser



Les idées fondamentales ont été inventées et mises en oeuvre entre

- 1948 : le premier ordinateur
- 1964 : les systèmes multitâches en temps partagé (CTSS)
- 1967 : la virtualisation (CP-67, d'IBM)

Le premier ordinateur : SSEM



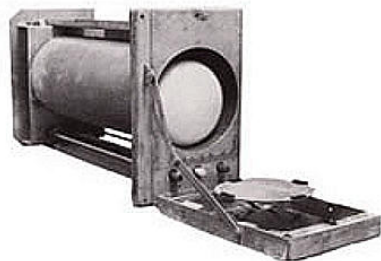
Small Scale Experimental Machine , alias “Baby” (21 juin 1948),

Le premier ordinateur

- calculateur électronique à **programme enregistré**
- Tom **Kilburn**, Geoff **Tootill**, et Fred. **Williams**, université de Manchester
- Tourne la première fois le 21 juin 1948
- Video
<http://news.bbc.co.uk/2/hi/technology/7465115.stm>

Objectifs

- machine **expérimentale**
- test de la mémoire à **tube de Williams-Kilburn**
- **utilisabilité?**



Tube de Williams-Kilburn

Écriture

- **tube cathodique** : un faisceau d'électrons "allume" des points de phosphore sur un écran (
- Pilotage du faisceau en X,Y = accès direct

Lecture

- **point déjà allumé** : émet des électrons secondaires
- **détection par une plaque** devant l'écran

Réalisation

- Machine 32 bits, mémoire de 32 mots (ext. à 2048).
- Deux registres : Accumulateur, Compteur de Programme, Registre d'Instruction

- Jeu de 7 instructions



Contexte (1948)

- depuis la fin du XIXe, machines mécanographiques (cartes perforées) pour le traitement de l'information
- en 1948 les calculateurs électroniques existaient
 - **Z3** de Konrad Zuse en **1942** (relais)
 - **Colossus**, Tommy Flowers, **1943** (1500 tubes à vide)
 - **ENIAC** d'Eckert et Mauchly, **1946** (17500 tubes)

Contexte (1948)

- calculateurs à programmes externes
 - sur bande perforée
 - tableau de prises à enficher
- technologies trop lentes ou trop chères pour le stockage des programmes
 - tubes à vide
 - lignes à retard (mercure)

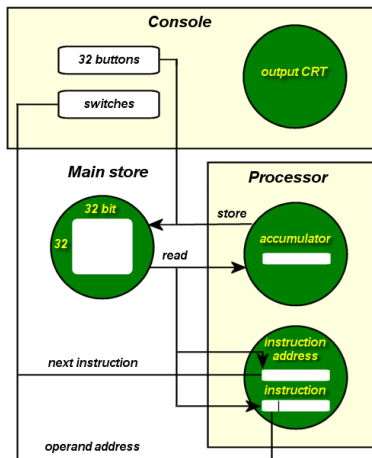
Du tube de Williams-Kilburn au SSEM

- En 1947, W et K arrivent à stocker 2048 bits sur un écran d'oscilloscope
- technologie performante et “bon marché”
- tests manuels ne permettent pas de s'assurer de la fiabilité
- donc **décision de réaliser un calculateur expérimental**

- calcul du plus grand diviseur propre de 2^{18}
- programme de 17 instructions
- divisions par soustraction successives
- 52 minutes de calcul,
- 3,5 millions d'instructions, environ 1100 instr/seconde
- résultat = ?

SSEM : éléments

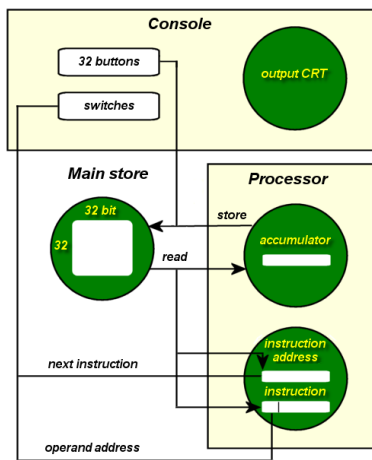
- mémoire 32 mots de 32 bits (M), sert d'affichage
- 3 registres
 - **current instruction** (CI) : 11 bits, numéro de l'instruction en cours
 - **program instruction** (PI) : 32 bits, instruction en cours
 - **accumulateur** (A) : 32 bits.
- bit AUTO (marche/arrêt)
- additionneur/soustracteur



SSEM : Cycle de fonctionnement

si $AUTO = 1$, le processeur exécute des cycles :

- 1 $PI = M[CI]$ recherche de l'instruction (**FETCH**)
- 2 exécution de l'instruction dans PI (**EXECUTE**)
- 3 $PI = PI + 1$



Les instructions du SSEM

Les instructions sur 32 bits contiennent

- Un **code opération** F sur 3 bits
- Une **adresse** S sur 11 bits

F (3)	(2)	S (11)	(16)
f f f	- -	s s s s s s s s s s s	- - - - - - - - - - - - - -

Les autres bits sont ignorés.

Jeu d'instructions

Seulement 7 instructions :

F	nom	effet	description
000	JMP	CI = M[S]	saut indirect
001	SUB	A -= M[S]	soustraction
010	LDN	A = -M[S]	chargement de l'opposé
011	CMP	si A négatif, CI++	saute instruction si A négatif
100	JPR	CI += M[S]	saut relatif
101	-	-	-
110	STO	M[S] = A	rangement
111	HLT	AUTO = 0	arrêt

Pas d'addition ?

Exemple de programme : addition de deux nombres

Calcule $Z = X + Y$, avec X, Y et Z situées en 20, 21, 22

num	instruction	effet
0	LDN 20	A contient -X
1	SUB 21	A contient $-X-Y = -(X+Y)$
2	STO 22	Z contient $-(X+Y)$
3	LDN 23	A contient X+Y
4	STN 23	Z contient X+Y
5	HLT	arrêt

en langage d'assemblage (anachronique)

Les suites du SSEM : technologies

- Les tubes de Williams ont été utilisés dans quelques machines (IBM 701, 1953)
- technologie abandonnée peu après au profit des **tores de ferrite**, utilisés entre 1955 et 1975.
- L'université de Manchester a développé le **premier ordinateur à transistors** (1953).

Machines à transistors

Invention du transistor

- à effet de champ en 1925, et oublié...
- bipolaire à points de contact, en 1947 (Shockley, Bardeen et Brattain)

IBM 608 (1957) première machine entièrement transistorisée :
seconde génération.

Les suites du SSEM (machines)

- Le SSEM a posé les bases de l'architecture des machines modernes
- l'université de Manchester a développé (avec Ferranti) des ordinateurs, commercialisés à partir de 1951 :
- Mark 1, sur lequel ont tourné tourné
 - le premier programme de musique
 - le premier programme de jeu (résolution de problèmes d'échec, mat en deux coups)
- les premiers super-calculateurs : ATLAS / ATLAS 2 / TITAN

On retrouve les mêmes principes dans tous les ordinateurs.

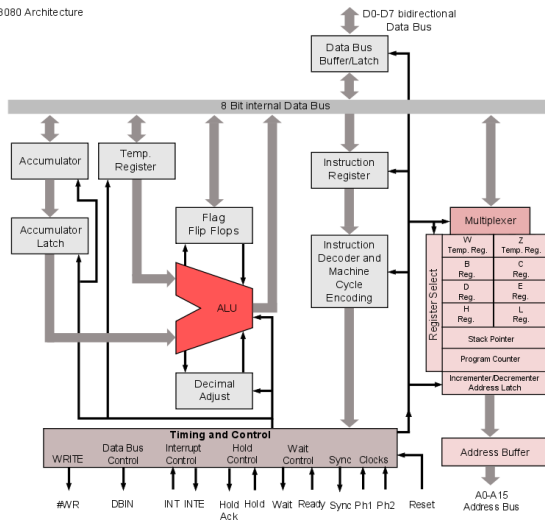
- processeur
- mémoire
- périphériques

Ce qui peut changer, dans le processeur :

- la taille des mots
- les registres
- les instructions
- les modes d'adressages
- ...

Architecture d'un processeur : 8080 Intel (1974)

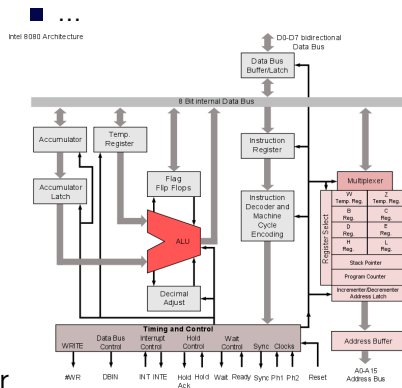
Intel 8080 Architecture



Architecture du 8080

On retrouve

- accumulateur (8 bits)
- compteur de programme (16 bits)
- unité arithmétique et logique
- registre d'instruction
- bus de données et d'adresses
- registres généraux



r

Généralités d'ordinateurs

Génération	Période	structure Technologies et machines
Première	1945-1955	tubes à vide
Seconde	1955-1965	machines à transistors mini-ordinateurs : PDP-1 de DEC (1960) 4 kilo-mots de 18 bits, 120.000 \$
Troisième	1965-1971	circuits intégrés Gamme System/360 d'IBM (1966)
Quatrième	1971-...	circuits VLSI, micro-processeurs.

Évolution des usages et des systèmes

Type d'usage	Type de système
Un programme à la fois	-
Enchaînement automatique	moniteur résident
Plusieurs programmes en mémoire	système multi-tâches
Conversationnel	système temps partagé

Évolutions matérielles : protection mémoire, mode superviseur, instructions privilégiées, interruptions, mémoire virtuelle...

Mode d'utilisation : un programme à la fois

Description

Chaque utilisateur prend possession de la machine à son tour, y transfère son programme et le fait exécuter.

Limitations

- pertes de temps entre deux travaux
- nécessité d'une planification
- matériel mal rentabilisé

Mode d'utilisation : Enchaînement de travaux

Description : le "traitement par lots"

L'utilisateur "client" amène son travail à un opérateur qui constitue des "lots" (batch) de travaux.
La machine enchaîne automatique le passage d'un travail au suivant.

Système

Le **moniteur d'enchaînement des travaux** (programme résident) est chargé en mémoire en début d'exploitation.
Il assure le chargement automatique et l'exécution des travaux.
Il fournit des "appels systèmes" pour accéder aux périphériques.

Attention : il faut ajouter des **protections**

Système résident : protections nécessaires

Protections du système envers les programmes utilisateurs

- protéger l'espace mémoire du système
- empêcher l'accès direct aux périphériques

Protections : solutions techniques

- Processeur à 2 modes de fonctionnement
 - **normal**
 - **privilegié** (réservé au système).
- **Espaces mémoires séparés** pour le système et les programmes
- **Instructions privilégiées** :
 - changement de mode,
 - opérations d'entrées-sorties
 - ...

Protections : fonctionnement

- En mode normal, le processeur
 - n'a pas accès à l'espace système
 - ne peut pas exécuter les instructions privilégiées
- les **violations d'accès** provoquent le **retour au système** (interruption, trap, exception), qui met fin au programme utilisateur.
- Un programme peut faire des demandes d'E/S au système par une instruction "**appel système**".

Déroulement d'un "appel système"

Quand un programme en mode normal fait un appel système :

- 1 **déroutement à une adresse fixée** du système, en mode privilégié
- 2 le système exécute l'opération d'E/S
- 3 à la fin, **retour** au programme demandeur, en mode normal.

Incidences sur le matériel

- **Modes** : un (**bit superviseur**)
- **Protection mémoire** : deux registres contiennent les adresses de début et de fin de l'espace utilisateur
- **Interruptions** :
 - un registre indique l'**adresse de la routine de traitement de l'interruption**
 - sauvegarde du compteur de programme dans un autre registre quand une interruption se produit.
 - restauration en fin d'interruption

Modifications matérielles

- Modifications simples du matériel
- proposées par John McCarthy au MIT (fin des années 50)
- implémentées au début des années 60 dans les machines commercialisées
- également nécessaires pour le multi-traitement

Usages : le multi-traitement

On remarque que

- les entrées-sorties sont **très lentes**
(bande perforée : 300 octets/s)
- programmes bloqués par les opérations d'E/S **synchrones**
- du temps de calcul potentiel est donc gaspillé.

Pour mieux rentabiliser les machines :

- faire tourner un programme pendant que l'autre attend

Usages : le multi-traitement

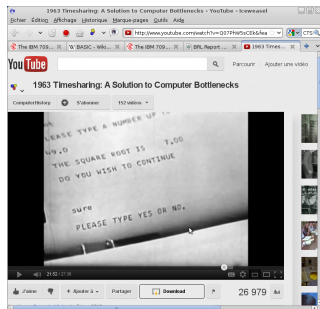
Idée :

- entrées-sorties déléguées à un **contrôleur de périphérique**
- opérations **asynchrones**
- le contrôleur envoie un signal (**interruption**) en fin d'opération.

Système multi-tâche : gère l'avancement de plusieurs programmes présents en mémoire.

Usages : temps partagé

- utilisation interactive depuis des terminaux
- chacun veut un temps de réponse raisonnable
- partage du temps entre les utilisateurs



« *Hundreds of people will one day be able to use tomorrow's computers simultaneously* » (1963)

<http://www.youtube.com/watch?v=Q07PhW5sCEk>

Nécessite

- système de **temps partagé préemptif**
pour empêcher un processus qui boucle de monopoliser la machine
- **mémoire virtuelle**
pour disposer d'une mémoire plus grande que la mémoire réelle

En résumé

- 1 un **système d'exploitation** est un programme qui gère des ressources : mémoire, périphériques, et ... temps.
- 2 intermédiaire entre les programmes et le matériel
- 3 grands principes mis en oeuvre dans les ordinateurs commercialisés des années 50 et 60.
- 4 évolution conjointe avec le matériel et les usages

En résumé

- 1 un **système d'exploitation** est un programme qui gère des ressources : mémoire, périphériques, et ... temps.
- 2 intermédiaire entre les programmes et le matériel
- 3 grands principes mis en oeuvre dans les ordinateurs commercialisés des années 50 et 60.
- 4 évolution conjointe avec le matériel et les usages

Quoi de neuf ?

L'évolution n'est pas terminée !

Exemple

- **miniaturisation** : un ordinateur dans quelques centimètres cubes.
- produit grand public : ordinateurs portables, smartphones
- mémoire et processeur suffisants pour faire tourner des applications sur un **système multitâche**

Mais...

L'évolution n'est pas terminée !

Exemple

- **miniaturisation** : un ordinateur dans quelques centimètres cubes.
- produit grand public : ordinateurs portables, smartphones
- mémoire et processeur suffisants pour faire tourner des applications sur un **système multitâche**

Mais...

- autonomie électrique ?

Impact sur...

L'évolution n'est pas terminée !

Exemple

- **miniaturisation** : un ordinateur dans quelques centimètres cubes.
- produit grand public : ordinateurs portables, smartphones
- mémoire et processeur suffisants pour faire tourner des applications sur un **système multitâche**

Mais...

- autonomie électrique ?

Impact sur...

- les matériels
- les systèmes ?

Dans un système multitâche :

- Gestion des processus
- Gestion de la mémoire
- Périphériques et systèmes de fichiers
- ...