Architecture et Système

Stefan Schwoon

Cours L3, 2023/2024, ENS Paris-Saclay

Organisation

Cours: Lundi 13h45–15h45, 1Z61 (Stefan Schwoon)
TP: Mercredi 13h45–16h45, 1S53 (Nicolas Margulies et Théo Vignon)

Contrôles de connaissances : 2 Projets + Examen (un tiers chacun)

Seconde session : rattrapage d'examen, projets non rattrapables

Transparents etc : `http://www.lsv.fr/~schwoon/`

E-mail: `schwoon@lsv.fr`

Bureau: 2U56

Contenu du cours

Contenu

Comment fonctionne un ordinateur ? Que se passe-t-il à l'intérieur ?

Aspects matériel (architecture) et logiciel (système)

Exemples :

matériel : circuits, assembleur, représentation de données, mémoire, interruptions, . . .

logiciel : shell, programmation système, système fichiers, réseau, . . .

Contenu du cours

Objectifs:

Connaissances pratiques pour vos tâches quotidiennes, p.ex. en expérimentation, programmation, manipulation de données etc

Système d'exploitation : Linux/POSIX

Langages utilisés : assembleur (un petit peu) et surtout C

Littérature

Architecture :

John P. Hayes, *Computer Architecture and Organization*, McGraw Hill (3rd edition)

Système d'exploitation :

Andrew S. Tanenbaum, *Operating systems*, Prentice Hall

Contenu d'aujourd'hui

Historique (très) abregée des ordinateurs

Développement de l'architecture

Début sur les circuits logiques

Les premières machines de calcul mécaniques

Au 17e siècle : Wilhelm Schickard / Blaise Pascal / Gottfried Leibniz

Machines méchaniques, basées sur le système décimal

Pascaline : addition, subtraction (par complément)

Machine de Leibniz : multiplication en plus (grandes problèmes méchaniques)

motivation : astronomique (S.), calcul financier, impôts (P.), philosophique (L.)

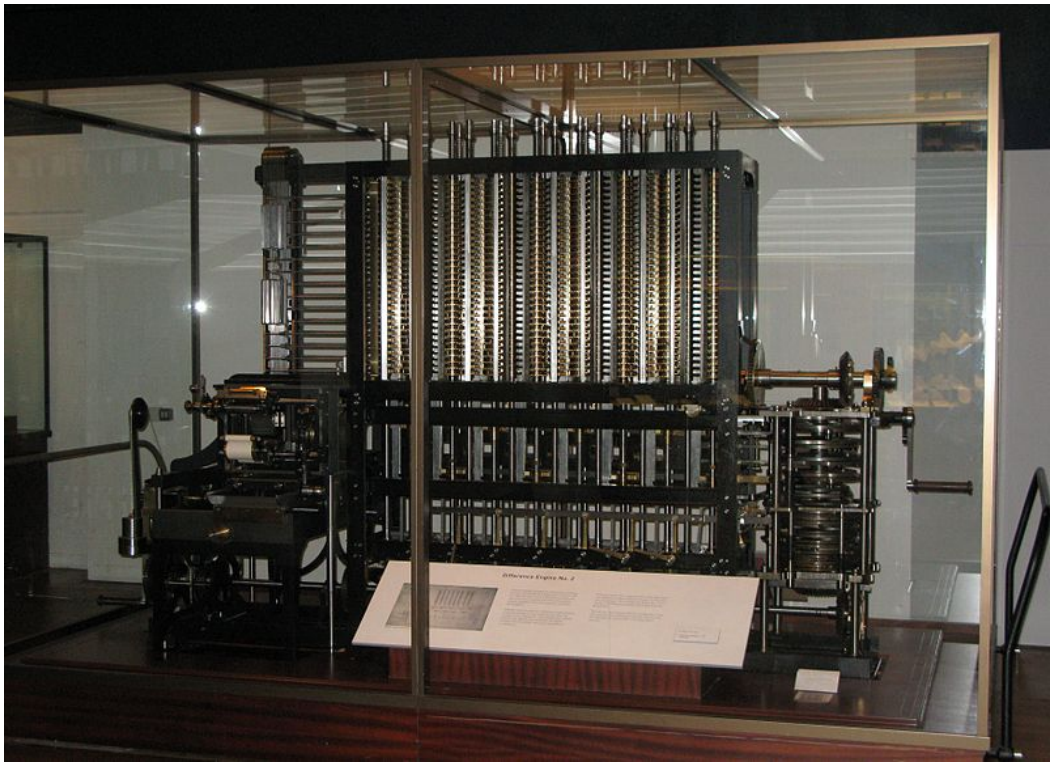
production en masse : à partir du 19e siècle (Thomas de Colmar)

Première machine dite programmable

Difference Engine (1822-1832) de Charles Babbage

Une opération (l'addition) appliquée simultanément à plusieurs registres

"Programmation" consiste en déterminant les valeurs initiales



Source: Wikimedia Commons, utilisateur geni; reconstruction exposée au Musée de la Science, Londres

Fonctionnement de la Difference Engine

Fonctionnement (permet de calculer des polynômes) :

pour n registres, fixer les valeurs initiales $x_1^{(0)}, \ldots, x_n^{(0)}$;

appliquer la régle $x_{i+1}^{(k+1)} := x_i^{(k)} + x_{i+1}^{(k)}$, pour $1 \leq i < n$ et $k \geq 0$

La machine réellement construit avait 3 régistres à 6 chiffres.

Motivation : calcul automatique des tableaux mathématiques

Problèmes méchaniques empêchent une construction efficace

Innovations : mémoire, itération

Machine théorique conçue par Babbage

Analytical Engine (travail théorique, jamais mis en production)

Opérations multiples (addition/multiplication/. . .)

Séquence d'opérations programmable avec branchement conditionnel

Fin 19e/début 20e siècle

Améliorations méchaniques et en utilisabilité

Des machines construites pour un but précis :

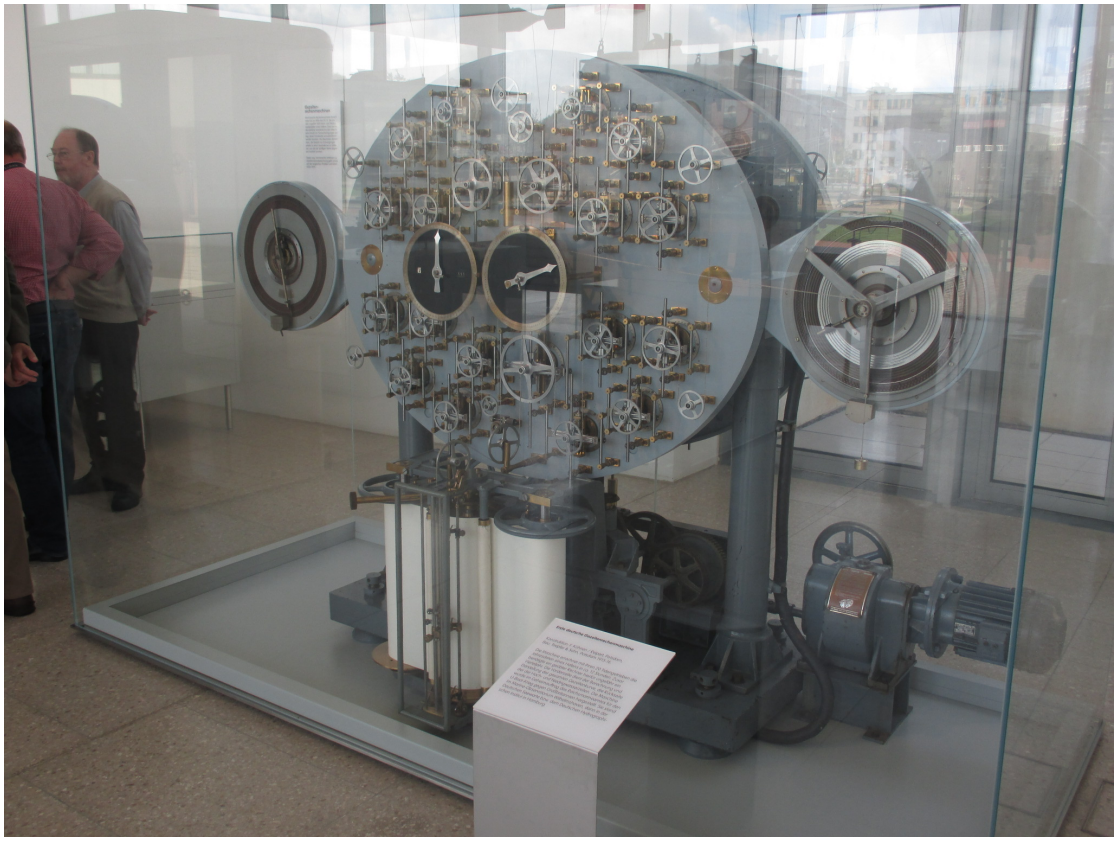
calculatrices utilisés par les ingénieurs, navigateurs, militaires, . . .

automation dans la production des tableaux résultants

Encore des machines méchaniques basées sur le système décimal.

Exemple d'une machine analogue

Calculatrice de marées de 1915



machine exposée au Musée allemand maritime, Bremerhaven

Calcule les marées d'un port donnée (dans la mer du nord) pour un an dans 12h

La “machine” de Turing

Alan Turing (1936)

Conception théorique d’une machine universelle :

> ruban infini pour stocker des données + un état interne
>
> la machine lit un symbole à la fois, se déplace par une position et modifie son état selon des règles fixes

Machine de Turing universelle: peut simuler d’autres machines de Turing
$\Longrightarrow$ un programme est une donnée

Modèle reconnu comme étant équivalent en pouvoir à un ordinateur quelconque; terminaison indécidable

Les années 1930

Les machines de Konrad Zuse : Z1 (1938) and Z3 (1941)

- fonctionnement électro-méchanique
- calcul binaire, virgule flottante
- programmation avec des boucles (mais pas de branchement conditionnel)

Source: Wikimedia Commons, utilisateur Venusianer; replique exposée au Musée allemand, Munich

Pendant la guerre

Colossus (britannique, 1943, pour briser des codes)

- existence connue publiquement depuis les années 1970
- construit pour un objectif précis, pas universelle, pas vraiment programmable
- calcul binaire

Mark I (américaine, 1944, calcul ballistique et bombe atomique)

- calcul décimal
- programmable par bande perforée
- Turing-complet

Les premiers ordinateurs électriques

ENIAC (1946)

construit à l'Université de la Pennsylvanie

Poids : 30 tonnes; 18.000 tubes électroniques

encore décimal (20 registres à 10 chiffres)

programmable en (dé)branchant des câbles

temps requis pour une multiplication: 3 ms

utilisé pour des calculs ballistiques

Les ordinateurs construits par von Neumann

EDVAC (1951)

Calcul binaire

programme stocké en mémoire

instructions de la forme (a_1, a_2, a_3, a_4, op):
appliquer op sur les données aux adresses a_1 et a_2, stocker le résultat à a_3, prochaine instruction à a_4.

branchement conditionnel: comparer les données à a_1, a_2, continuer soit à a_3 ou à a_4.

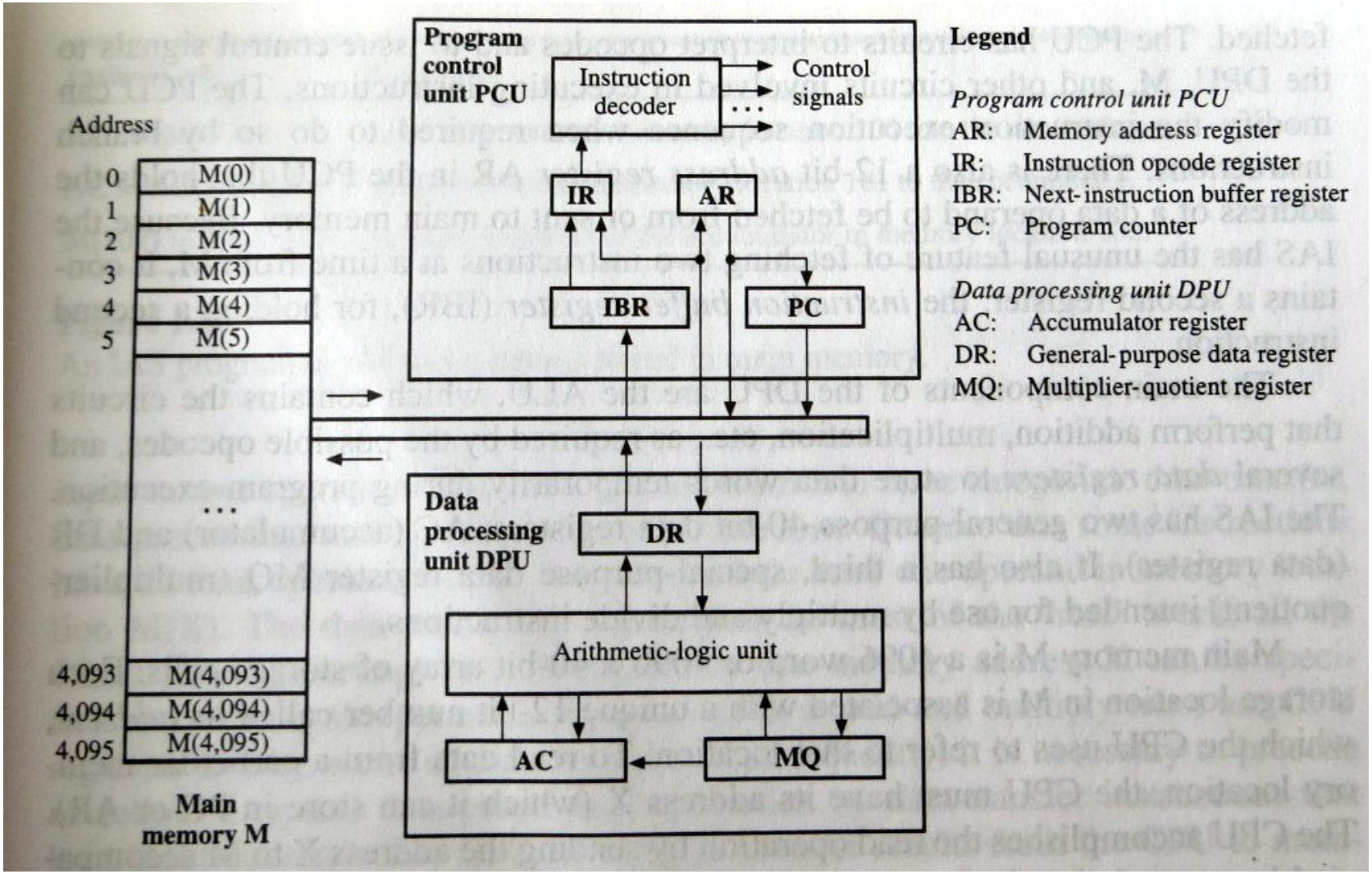
La machine de l'IAS (construit à Princeton)

mémoire: $4096 = 2^{12}$ cellule ('mots') de 40 bit

notion d'un *processeur central* et *unité arithmétique-logique* (CPU et ALU)

considerée comme le prototype de l'architecture moderne

Architecture de l'IAS



Instructions de l'IAS

Un mot dans la mémoire peut être interpreté soit comme une instruction, soit comme une valeur numérique.

Interpretation numérique : entier ou virgule fixe entre -1/+1

Instruction : un mot = 2 instructions de 20 bits

format (op, a) pour manipuler mémoire et registres

– op: code d'opération (8 bit) / a = adresse de 12 bit

transfert de données entre mémoire et registres/entre registres

addition/multiplication

opérations de contrôle (branchement conditionnel, code automodifiant)

Ordinateurs de “deuxième géneration” (50s-60s)

Couche physique :

- remplacement de tubes par des transitors
- plus petit, moins cher, plus rapide, plus fiable

Architecture:

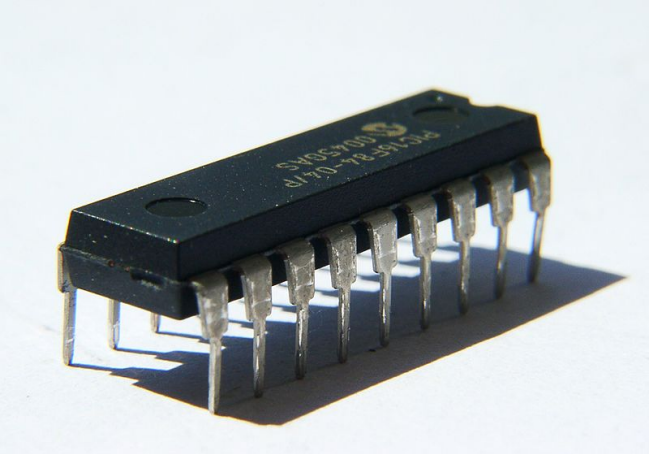
- instructions plus puissantes : adressage indirect, registres d’index
- registres et opérations pour les virgules flottantes
- récursion (opération sur pile)

Ordinateurs de “troisième géneration” (1960s-70s)

Arrivée des circuits integrés (IC)

beaucoup de transitors sur un espace énormement reduit

$\rightarrow$ opérations plus rapides



Mémoires à vitesses différentes (cache); integrée soit sur IC, soit ailleurs

Optimisations (À partir des années 80)

Découpage du processeur en plusieurs sous-unités travaillant en parallèle sur des tâches bien précises : obtention d'une instruction, décodage, exécution, ...

Mise en place de plusieurs unités d'exécution *concurrentes*

Développement d'une couche de micro-architecture :

une instruction assembleur est découpée en plusieurs micro-instructions

ordonnancement de ces micro-instructions sur plusieurs unités en parallèle

exécution "hors-ordre", prédiction des branches, exécution spéculative, ...