

# Architecture de l'ordinateur

Frédérique Barré, Éric Favier & Vincent Lozano

Énise

Année 2007/2008

# Sommaire

## 1 Introduction

# Sommaire

- 1 Introduction
- 2 Historique

# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur

# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur

# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur
- 5 Architecture en couches

# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur
- 5 Architecture en couches

# Introduction

## Définition

« Un ordinateur est une **machine**  
capable de résoudre des **problèmes**  
en appliquant des **instructions** préalablement définies »

Andrew Tanenbaum  
*Architecture de l'ordinateur, InterÉditions, 1988.*

# Introduction

## Objectif de cet enseignement

- Étudier d'un point de vue **logique** (et non électronique)

# Introduction

## Objectif de cet enseignement

- Étudier d'un point de vue **logique** (et non électronique)
  - 1 le fonctionnement des composants internes d'un ordinateur

# Introduction

## Objectif de cet enseignement

- Étudier d'un point de vue **logique** (et non électronique)
  - 1 le fonctionnement des composants internes d'un ordinateur
  - 2 la façon dont ils dialoguent

# Introduction

## Objectif de cet enseignement

- Étudier d'un point de vue **logique** (et non électronique)
  - 1 le fonctionnement des composants internes d'un ordinateur
  - 2 la façon dont ils dialoguent
- Connaître la nature des **informations** manipulées et leur codage

# Introduction

## Objectif de cet enseignement

- Étudier d'un point de vue **logique** (et non électronique)
  - 1 le fonctionnement des composants internes d'un ordinateur
  - 2 la façon dont ils dialoguent
- Connaître la nature des **informations** manipulées et leur codage
- Expérimenter en « **programmant** » en langage machine.

# Sommaire

- 1 Introduction
- 2 Historique**
  - Machines à calculer
  - Métiers à tisser
  - Charles Babbage
  - Les premiers ordinateurs
  - Joseph Von Neuman
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur





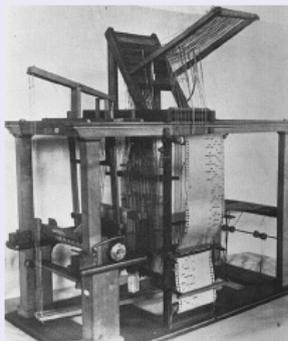




# Métiers à tisser

## 1725 Basile Bouchon (lyonnais)

Utilise une **bande de papier perforé** pour automatiser le tissage des motifs

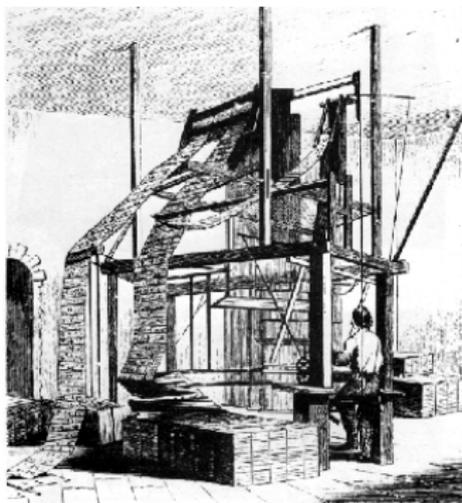




# Métiers à tisser

1800 Joseph-Marie Jacquard

Invente le premier métier entièrement automatique (les cartes perforées contiennent le **programme** fixant le motif ornemental et commandant le mouvement des « lisses »)



# Charles Babbage

1820

Il conçoit la [difference engine](#)

- permettant de graver sur des plateaux de cuivre des tables numériques pour la navigation en mer ;
- basée sur la méthode des différences finies ;
- utilisant des cartes perforées.

# Charles Babbage

## 1820

Il conçoit la **difference engine**

- permettant de graver sur des plateaux de cuivre des tables numériques pour la navigation en mer ;
- basée sur la méthode des différences finies ;
- utilisant des cartes perforées.

## 1823–1843

Il tente de la mettre au point.



# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - ① le magasin (mémoire) ;

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - 1 le magasin (mémoire) ;
  - 2 le moulin (unité de calcul) ;

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - 1 le magasin (mémoire) ;
  - 2 le moulin (unité de calcul) ;
  - 3 l'entrée (lecteur de cartes perforées) ;

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - 1 le magasin (mémoire) ;
  - 2 le moulin (unité de calcul) ;
  - 3 l'entrée (lecteur de cartes perforées) ;
  - 4 la sortie (perforation ou impression)

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - ① le magasin (mémoire) ;
  - ② le moulin (unité de calcul) ;
  - ③ l'entrée (lecteur de cartes perforées) ;
  - ④ la sortie (perforation ou impression)
- pouvant tester si un nombre est positif

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - ① le magasin (mémoire) ;
  - ② le moulin (unité de calcul) ;
  - ③ l'entrée (lecteur de cartes perforées) ;
  - ④ la sortie (perforation ou impression)
- pouvant tester si un nombre est positif
- pouvant faire un branchement conditionnel

# Charles et Ada

1834

Charles rencontre [Ada Lovelace](#) (fille du poète Lord Byron)

Il entreprend la conception de l'[analytical engine](#) :

- composée de 4 parties :
  - ① le magasin (mémoire) ;
  - ② le moulin (unité de calcul) ;
  - ③ l'entrée (lecteur de cartes perforées) ;
  - ④ la sortie (perforation ou impression)
- pouvant tester si un nombre est positif
- pouvant faire un branchement conditionnel
- utilisant un langage de programmation très simple

# Charles et Ada

1842

Ada Lovelace publie la description de l'analytical engine

# Charles et Ada

1842

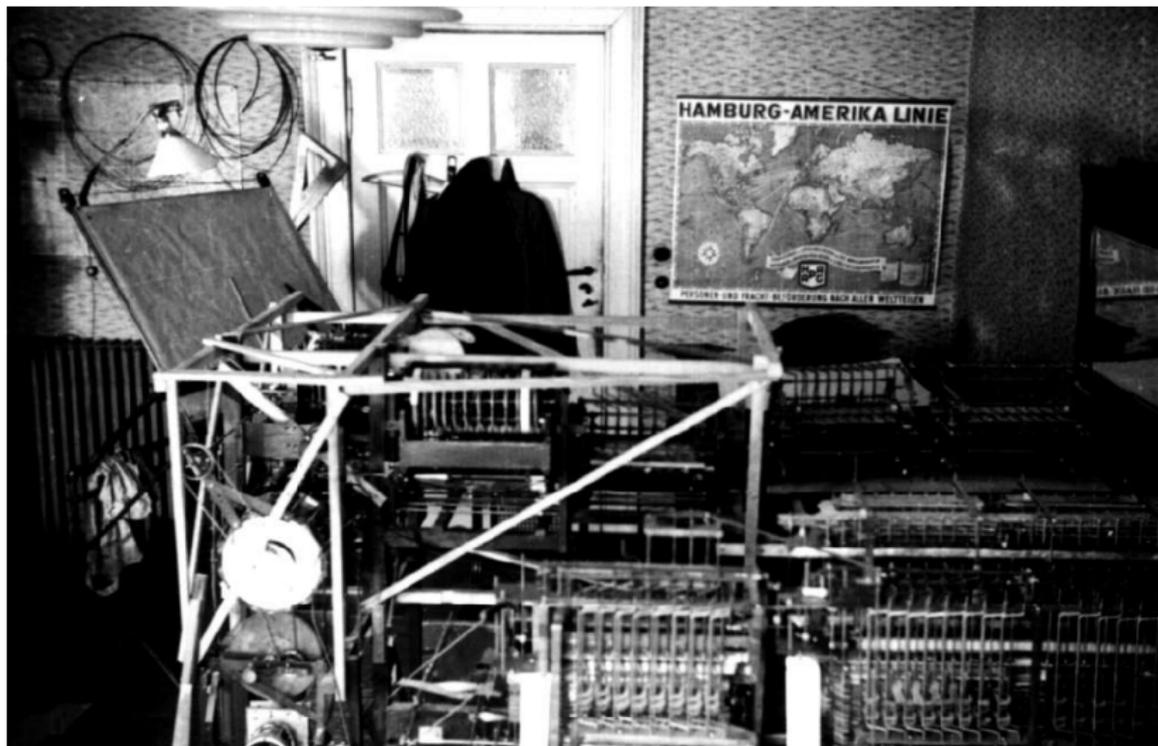
Ada Lovelace publie la description de l'analytical engine

1880

Le fils de Charles, Henry Babbage, réalise une partie de la machine



# 1936 Konrad Zuse



# ENIAC

1945 Electronic Numerical Integrator, Analyzer and Computer

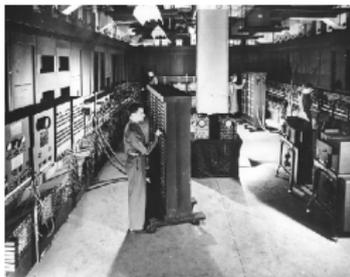
Il s'agit du premier « ordinateur »

# ENIAC

## 1945 Electronic Numerical Integrator, Analyzer and Computer

Il s'agit du premier « ordinateur »

- 17468 tubes électroniques et 1600 relais, 30 tonnes, 150KW, plus deux puissants moteurs, occupant 169 m<sup>2</sup> au sol

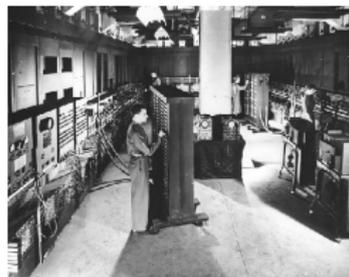


# ENIAC

## 1945 Electronic Numerical Integrator, Analyzer and Computer

Il s'agit du premier « ordinateur »

- 17468 tubes électroniques et 1600 relais, 30 tonnes, 150KW, plus deux puissants moteurs, occupant 169 m<sup>2</sup> au sol
- les données sont sur des cartes perforées

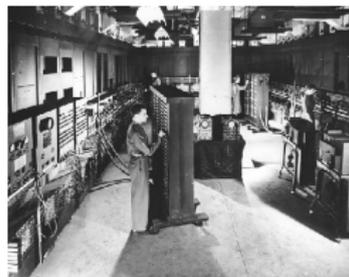


# ENIAC

## 1945 Electronic Numerical Integrator, Analyzer and Computer

Il s'agit du premier « ordinateur »

- 17468 tubes électroniques et 1600 relais, 30 tonnes, 150KW, plus deux puissants moteurs, occupant 169 m<sup>2</sup> au sol
- les données sont sur des cartes perforées
- les programmes sont câblés en mémoire



# Von Neuman

Fin 1945

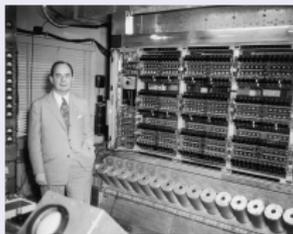
Il propose un modèle d'ordinateur

# Von Neuman

Fin 1945

Il propose un **modèle d'ordinateur**

- machine universelle contrôlée par programme

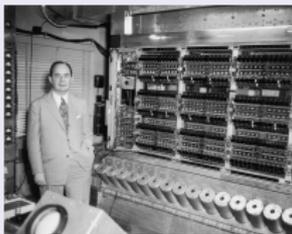


# Von Neuman

Fin 1945

Il propose un **modèle d'ordinateur**

- machine universelle contrôlée par programme
- **instructions** et **données**, sous format binaire, stockées en mémoire

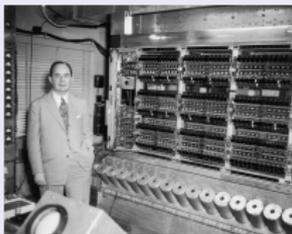


# Von Neuman

Fin 1945

Il propose un **modèle d'ordinateur**

- machine universelle contrôlée par programme
- **instructions** et **données**, sous format binaire, stockées en mémoire
- le programme peut modifier ses propres instructions

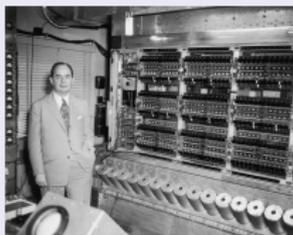


# Von Neuman

Fin 1945

Il propose un **modèle d'ordinateur**

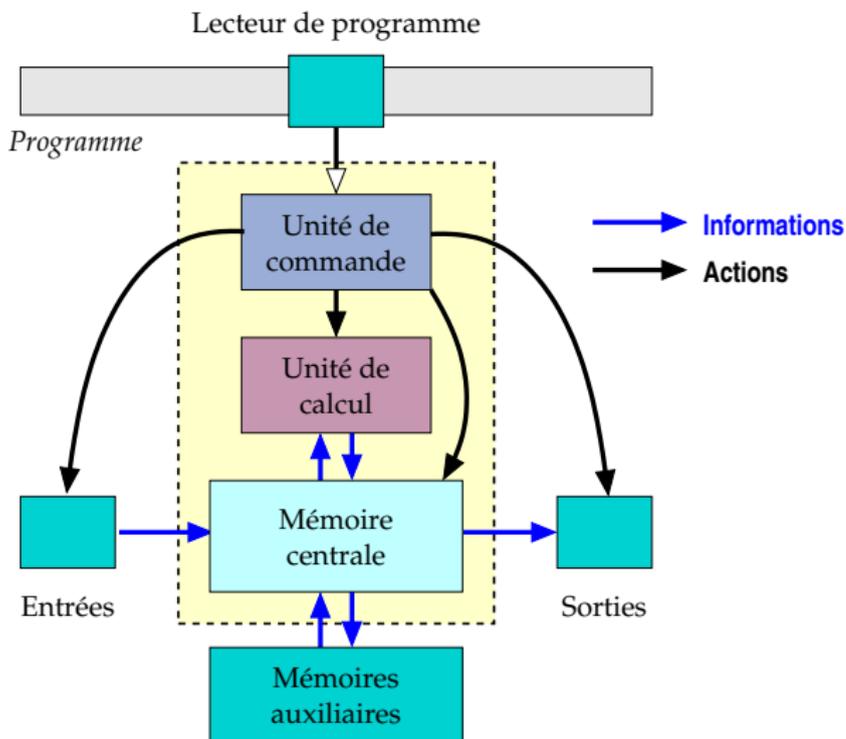
- machine universelle contrôlée par programme
- **instructions** et **données**, sous format binaire, stockées en mémoire
- le programme peut modifier ses propres instructions
- permettant les ruptures de séquence



# Sommaire

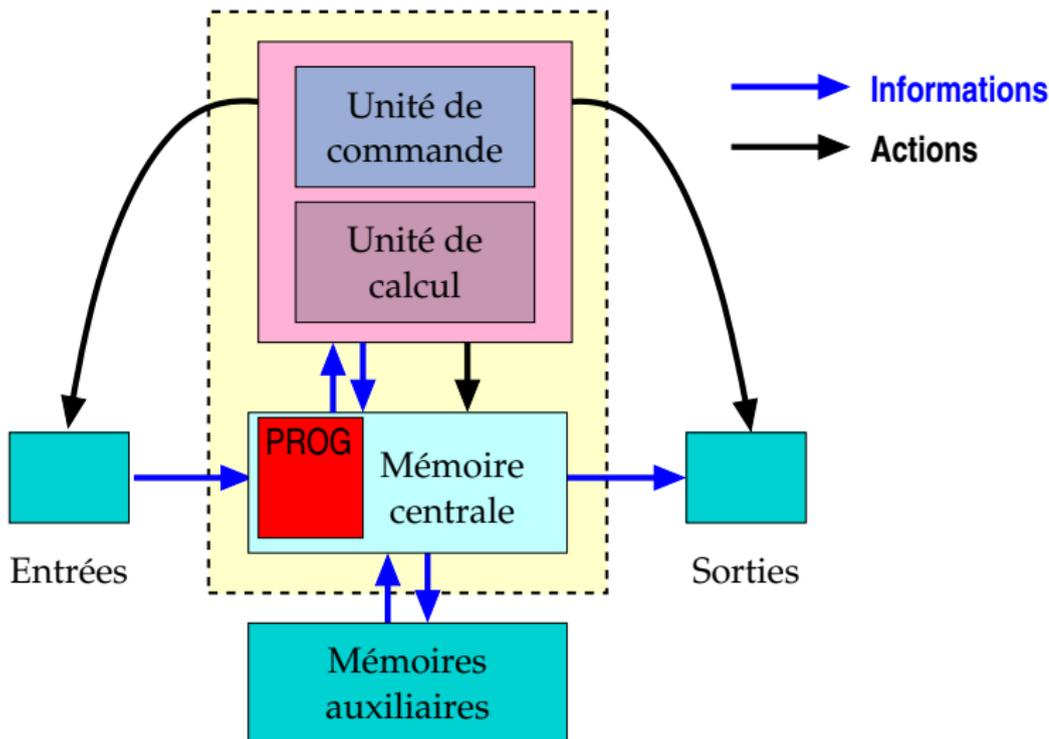
- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur**
  - Schéma général
  - Unité centrale
  - Mémoire centrale
  - Les bus
  - Périphériques
- 4 Fonctionnement de l'ordinateur

# Machine de Babbage

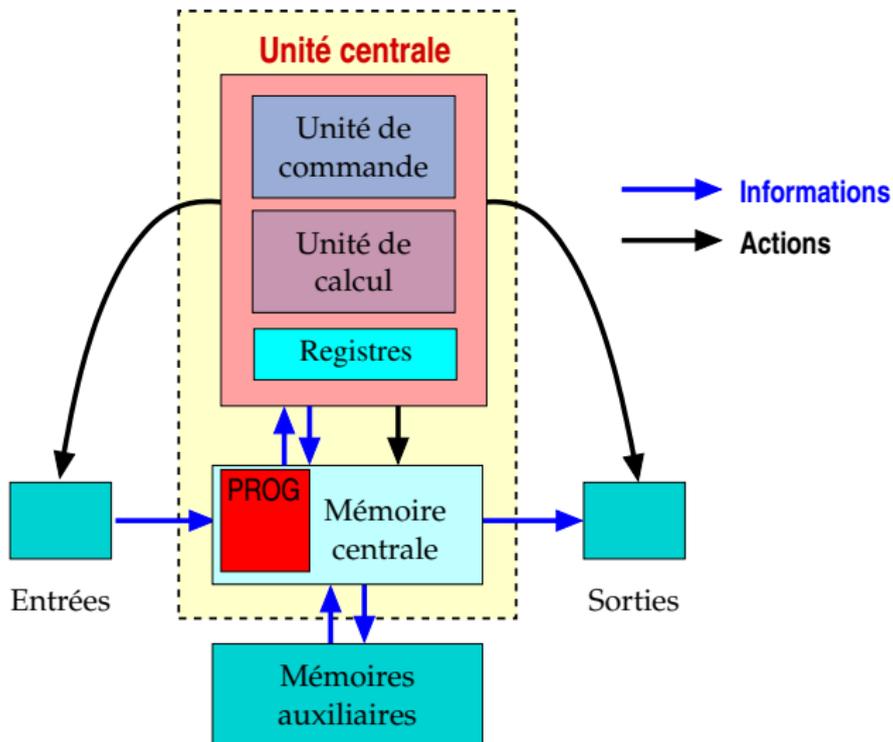




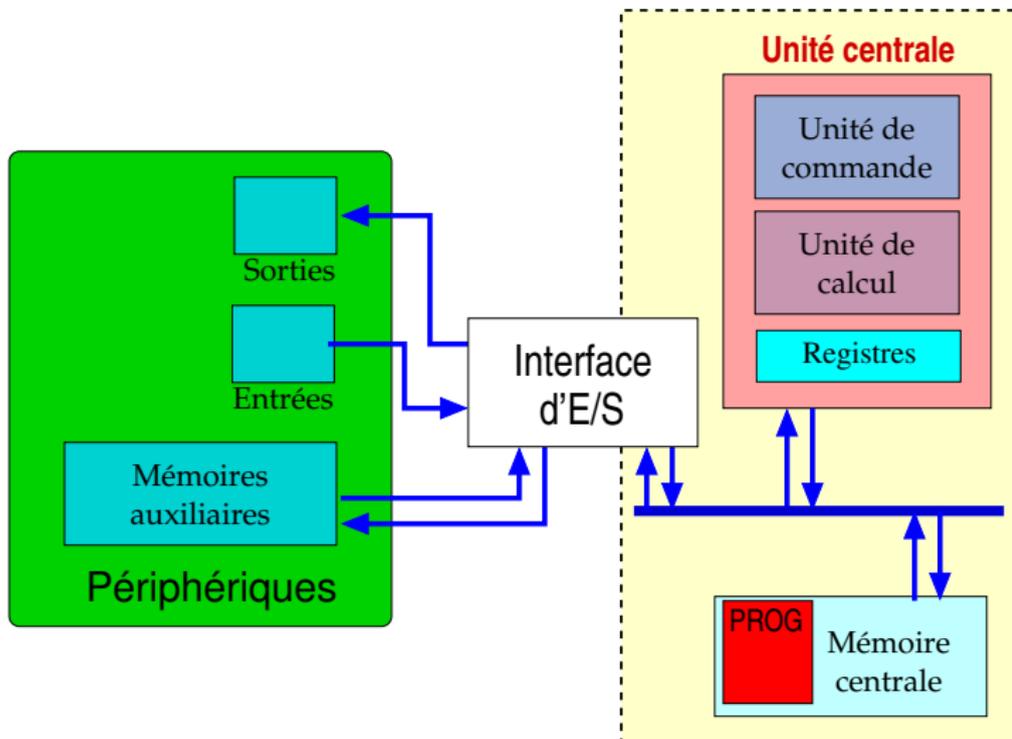
# Schéma général



# Schéma général



# Schéma général



# Unité centrale

En anglais

central processing unit (CPU)

# Unité centrale

## En anglais

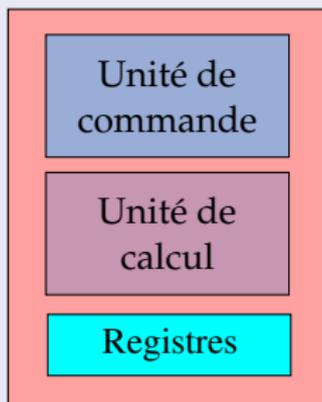
control processing unit (CPU)

## Une définition

C'est le « cerveau » de l'ordinateur,  
son rôle est d'exécuter  
les programmes  
chargés en mémoire.

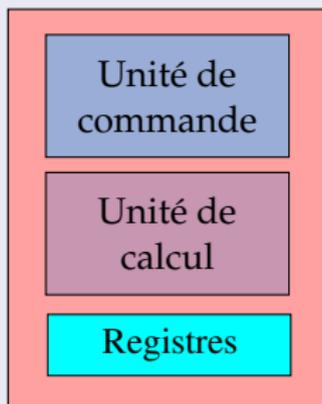
# Unité centrale (ou processeur)

Elle comporte trois éléments reliés par un bus interne



# Unité centrale (ou processeur)

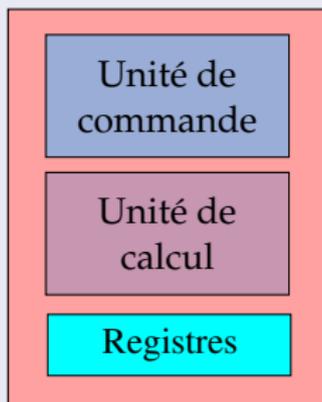
Elle comporte trois éléments reliés par un bus interne



Unité de commande (UC) cadencée  
par une horloge pilote  
l'exécution des  
instructions

# Unité centrale (ou processeur)

Elle comporte trois éléments reliés par un bus interne

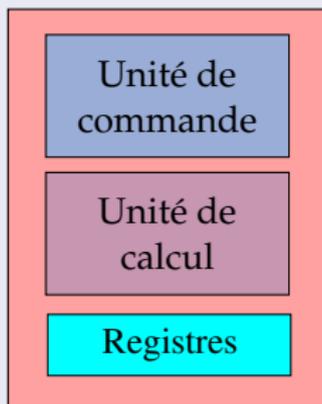


Unité de commande (UC) cadencée par une horloge pilote l'exécution des instructions

Unité arithmétique et logique (UAL) exécute des opérations arithmétiques et booléennes

# Unité centrale (ou processeur)

Elle comporte trois éléments reliés par un bus interne



**Unité de commande (UC)** cadencée par une horloge pilote l'exécution des instructions

**Unité arithmétique et logique (UAL)** exécute des opérations arithmétiques et booléennes

**Registres** mémoire de taille réduite mais d'accès très rapide.

# Unité centrale (ou processeur)

## Définition

**Microprocesseur** : UC intégrée dans un circuit électronique

## Exemples

- INTEL 80286 : 16 bits
- INTEL 80386 et 80486 : 32 bits
- INTEL PENTIUM : 32 bits
- INTEL ITANIUM : 64 bits
- MOTOROLA 68000 : 16/32 bits
- POWERPC
- SPARC

# Mémoire centrale

C'est la mémoire interne de l'ordinateur

Elle est à distinguer de la mémoire « auxiliaire »  
(périphériques de stockage)

# Mémoire centrale

C'est la mémoire interne de l'ordinateur

Elle est à distinguer de la mémoire « auxiliaire »  
(périphériques de stockage)

À tout instant on ne trouve en mémoire centrale que :

- 1 les programmes en cours d'exécution
- 2 les données manipulées par ces programmes

# Mémoire centrale

C'est la mémoire interne de l'ordinateur

Elle est à distinguer de la mémoire « auxiliaire »  
(périphériques de stockage)

À tout instant on ne trouve en mémoire centrale que :

- 1 les programmes en cours d'exécution
- 2 les données manipulées par ces programmes

## Sa structure

Bascules pouvant prendre deux états :

- désignés par 0 et 1
- appelées BIT(s) pour Binary digit

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

### Octets et multiples

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

### Octets et multiples

- octet (*byte*) : 8 bits

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

### Octets et multiples

- octet (*byte*) : 8 bits
- kilo-octet (Ko) : 1024 octets

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

### Octets et multiples

- octet (*byte*) : 8 bits
- kilo-octet (Ko) : 1024 octets
- mega-octet (Mo) : 1024 Ko = 1048576 octets

# Mémoire centrale

## Quelques définitions

### Adresse

c'est un **numéro** permettant de désigner une cellule

### Mot / cellule

- plus petite quantité de mémoire adressable : **8 bits**
- une cellule de 8 bits peut contenir  $2^8$  valeurs différentes

### Octets et multiples

- octet (*byte*) : 8 bits
- kilo-octet (Ko) : 1024 octets
- mega-octet (Mo) : 1024 Ko = 1048576 octets
- giga-octet (Go) : 1024 Mo = 1073741824 octets

# Les bus

## Bus de données

Il achemine les données entre les dispositifs d'entrées/sorties, la mémoire et les registres.

# Les bus

## Bus de données

Il achemine les données entre les dispositifs d'entrées/sorties, la mémoire et les registres.

## Bus d'adresses

Il transporte les adresses des instructions et des données (adresses en mémoire centrale ou adresses des dispositifs d'entrées/sorties)

# Les bus

## Bus de données

Il achemine les données entre les dispositifs d'entrées/sorties, la mémoire et les registres.

## Bus d'adresses

Il transporte les adresses des instructions et des données (adresses en mémoire centrale ou adresses des dispositifs d'entrées/sorties)

## Bus de commande

Il est constitué d'un ensemble de lignes acheminant divers signaux de contrôle (comme par exemple les interruptions)

# Périphériques

## Définition

C'est ce qu'il y a « autour » de l'unité centrale

# Périphériques

## Définition

C'est ce qu'il y a « autour » de l'unité centrale

## Caractéristiques

On distingue les périphériques :

# Périphériques

## Définition

C'est ce qu'il y a « autour » de l'unité centrale

## Caractéristiques

On distingue les périphériques :

D'entrée/sortie clavier, écran, souris, imprimante, table traçante, ...

# Périphériques

## Définition

C'est ce qu'il y a « autour » de l'unité centrale

## Caractéristiques

On distingue les périphériques :

**D'entrée/sortie** clavier, écran, souris, imprimante, table traçante, ...

**De stockage** mémoires « auxiliaires » ou secondaires (lecteurs de disques, bandes magnétiques, ...)

# Périphériques

## Définition

C'est ce qu'il y a « autour » de l'unité centrale

## Caractéristiques

On distingue les périphériques :

**D'entrée/sortie** clavier, écran, souris, imprimante, table traçante, ...

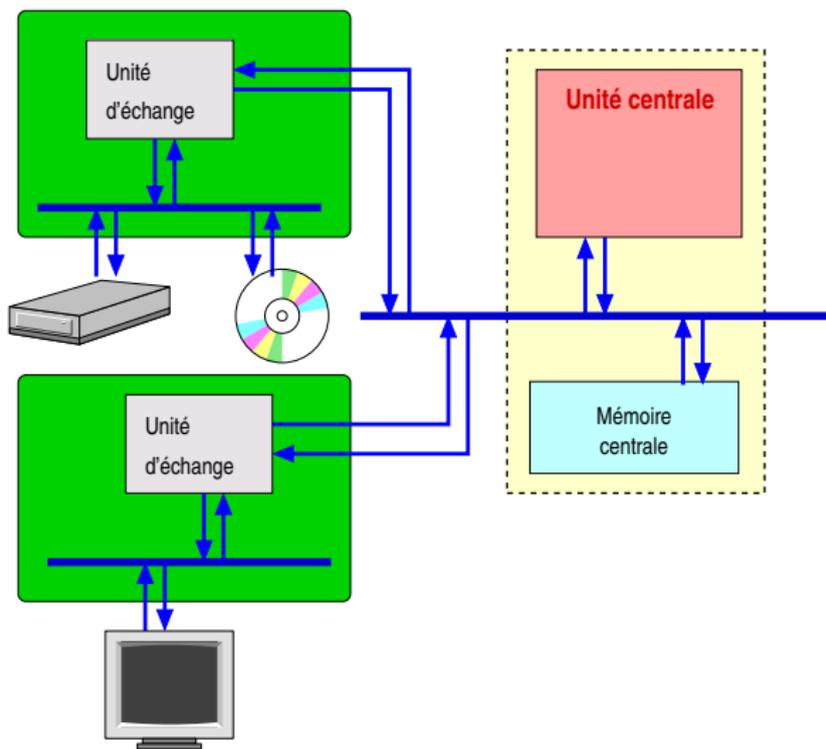
**De stockage** mémoires « auxiliaires » ou secondaires (lecteurs de disques, bandes magnétiques, ...)

## Coût

Inversement proportionnel au temps d'accès

# Périphériques

## Schéma général



# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur**
  - Utilisation d'un langage d'assemblage
  - Simulation du fonctionnement (1)
  - Simulation du fonctionnement (2)
- 5 Architecture en couches

# Langage d'assemblage

# Langage d'assemblage

## Transferts mémoire

MOV <destination> , <source>

Exemples :

# Langage d'assemblage

## Transferts mémoire

MOV <destination> , <source>

Exemples :

- MOV RD1,17 : transfert de la valeur 17 vers RD1

# Langage d'assemblage

## Transferts mémoire

MOV <destination> , <source>

Exemples :

- MOV RD1, 17 : transfert de la valeur 17 vers RD1
- MOV RD2, [adr] : transfert de la mémoire vers RD2

# Langage d'assemblage

## Transferts mémoire

MOV <destination> , <source>

Exemples :

- MOV RD1, 17 : transfert de la valeur 17 vers RD1
- MOV RD2, [adr] : transfert de la mémoire vers RD2
- MOV [adr], RD1 : transfert de RD1 vers la mémoire

# Langage d'assemblage

## Transferts mémoire

MOV <destination> , <source>

Exemples :

- MOV RD1 , 17 : transfert de la valeur 17 vers RD1
- MOV RD2 , [adr] : transfert de la mémoire vers RD2
- MOV [adr] , RD1 : transfert de RD1 vers la mémoire

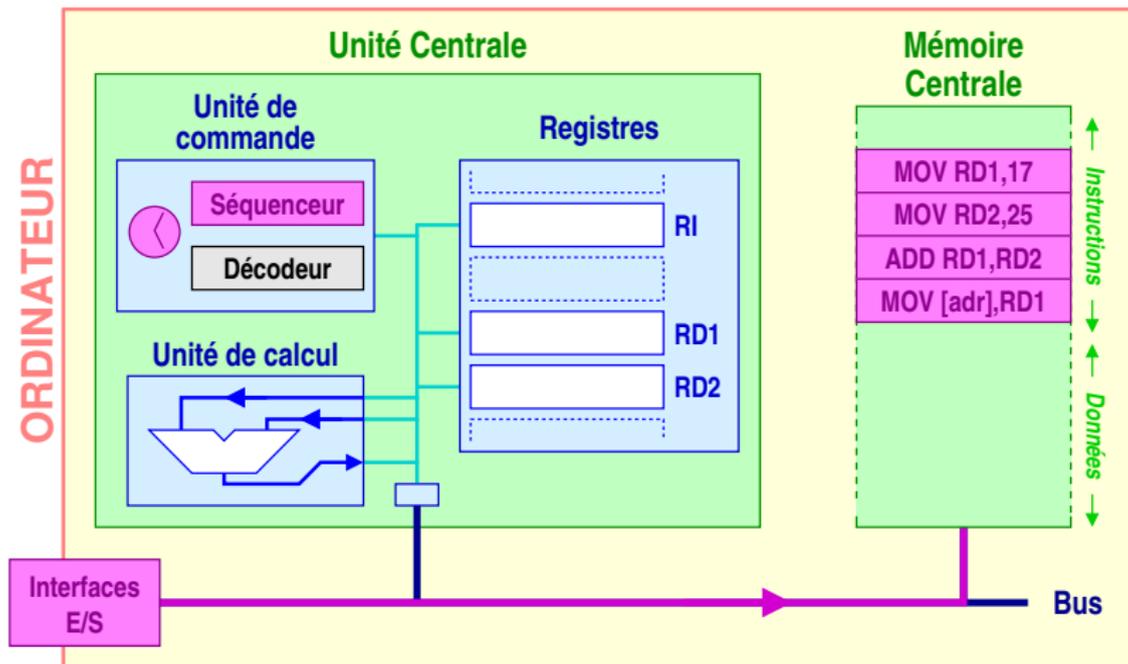
## Opérations arithmétiques

ADD <RDi> , <RDj>

Les valeurs stockées dans les registres de données RDi et RDj sont additionnées et le résultat est placé dans RDi.

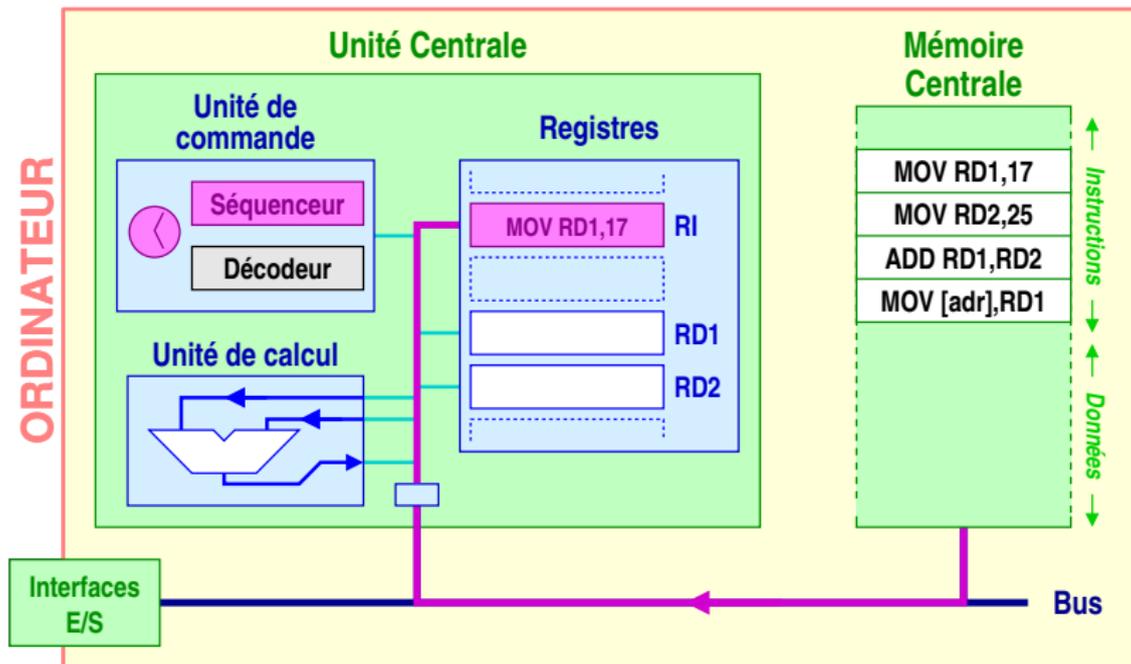
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



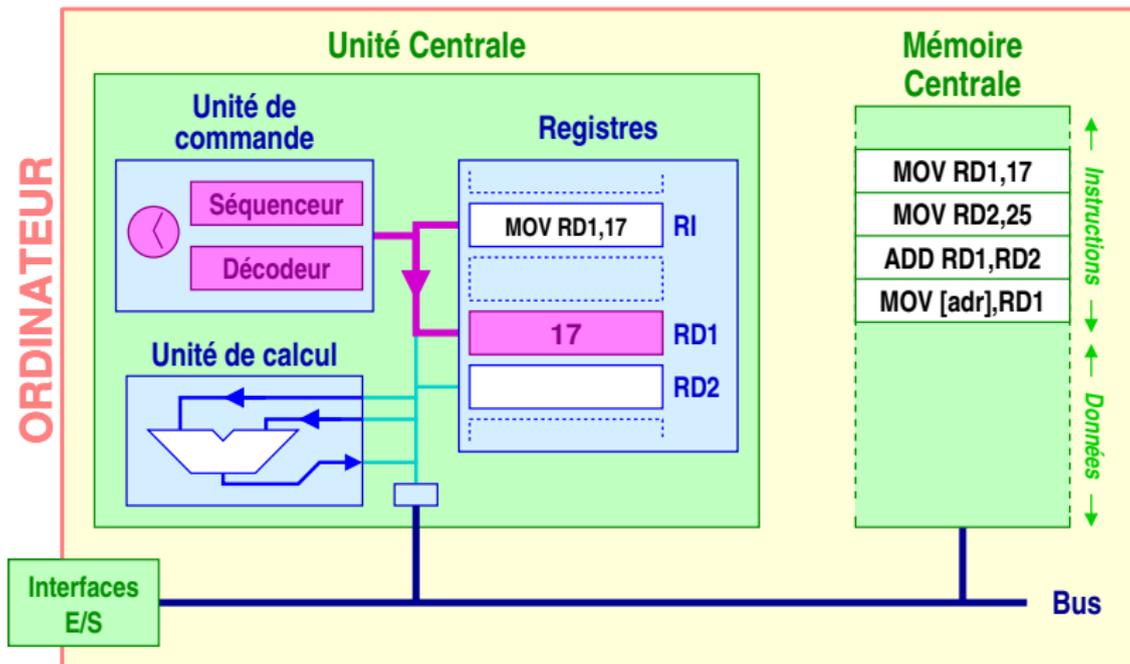
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



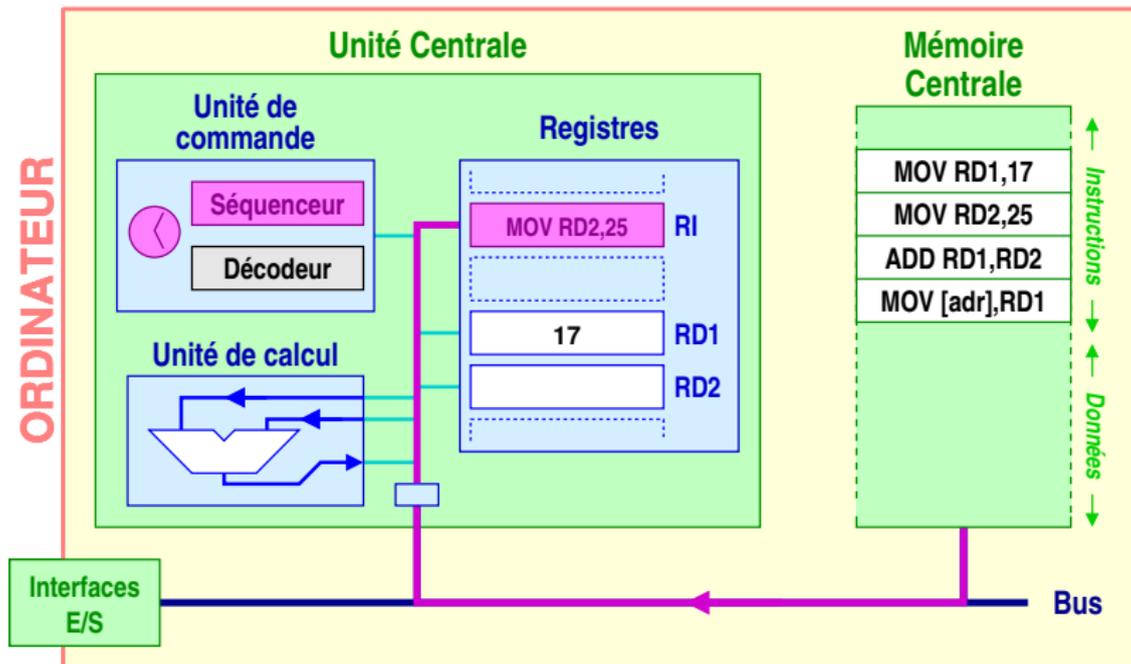
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



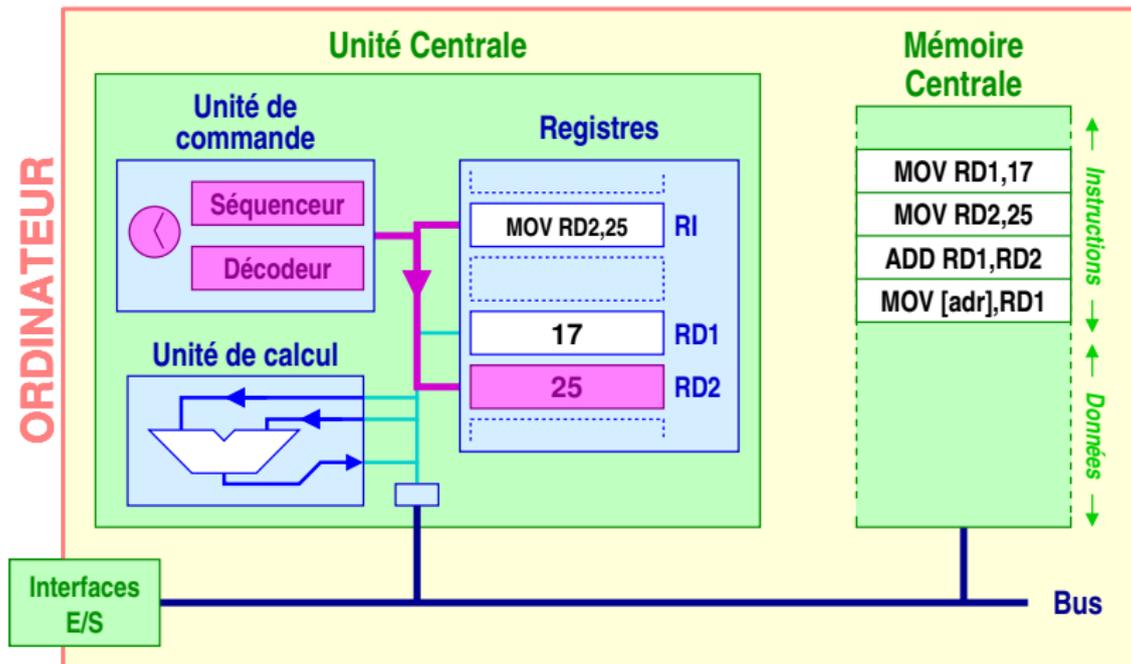
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



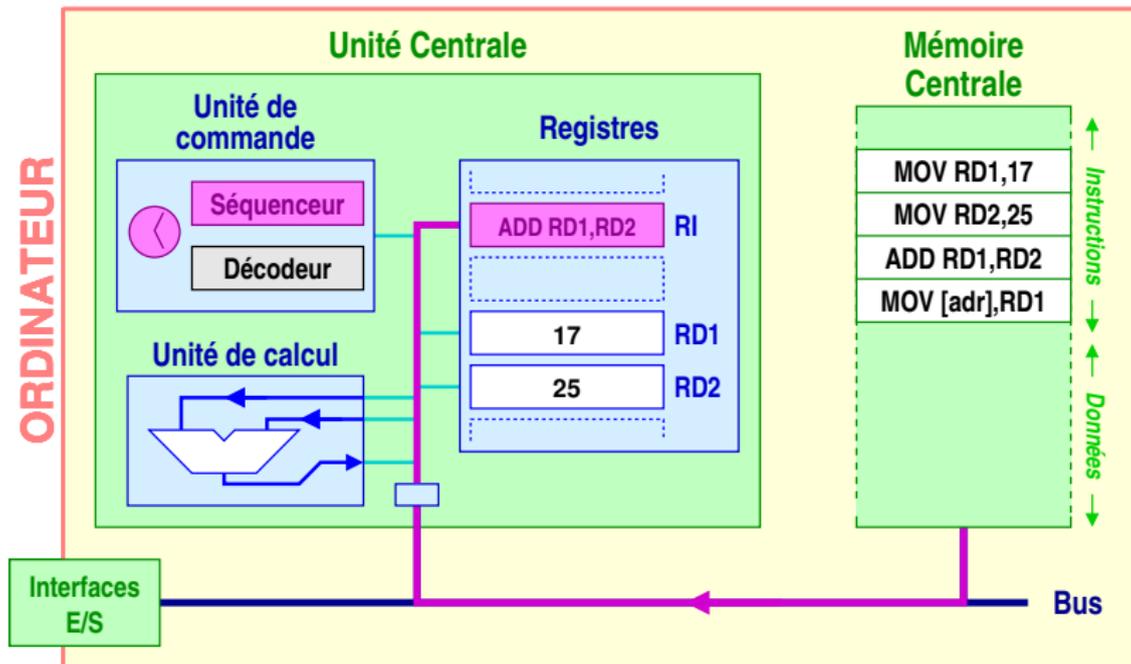
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



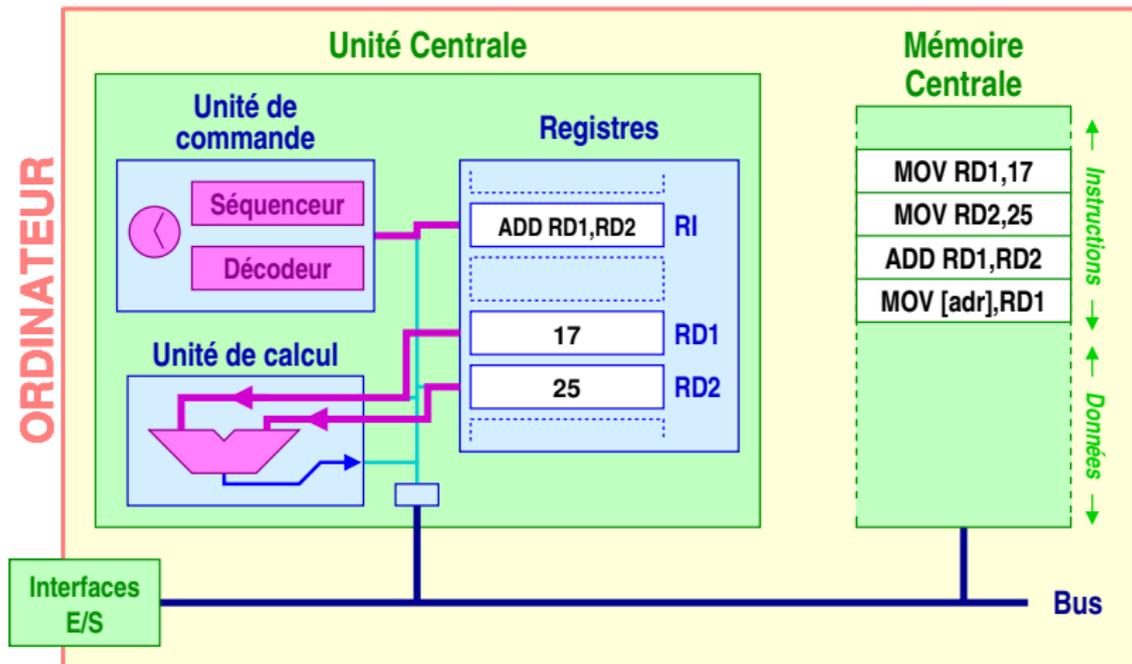
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



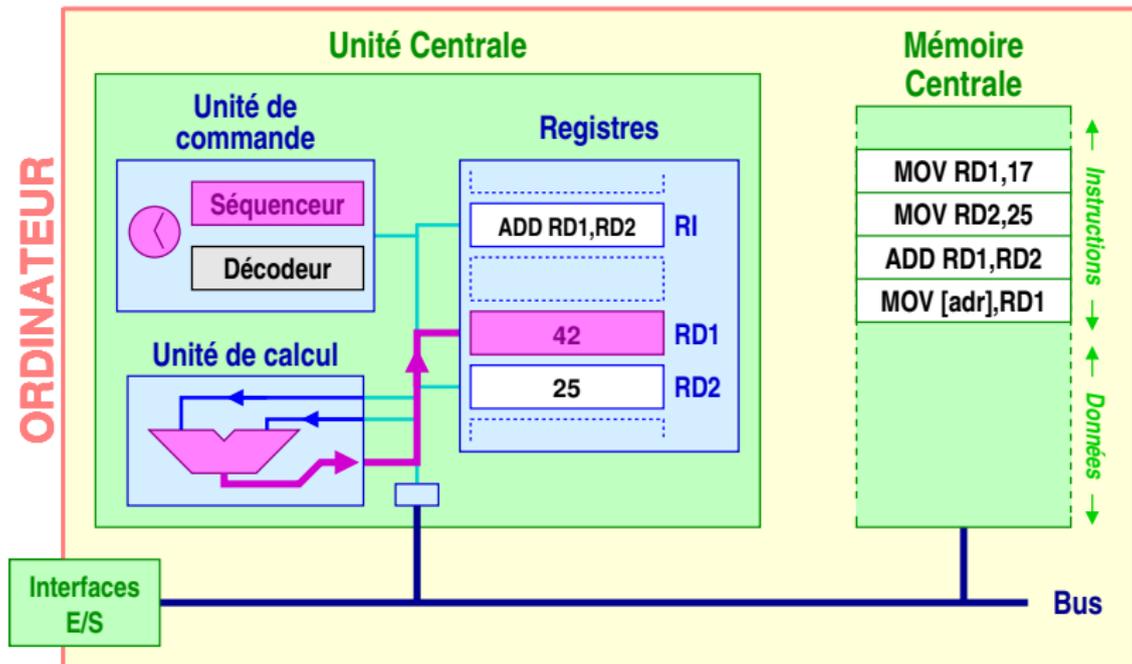
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



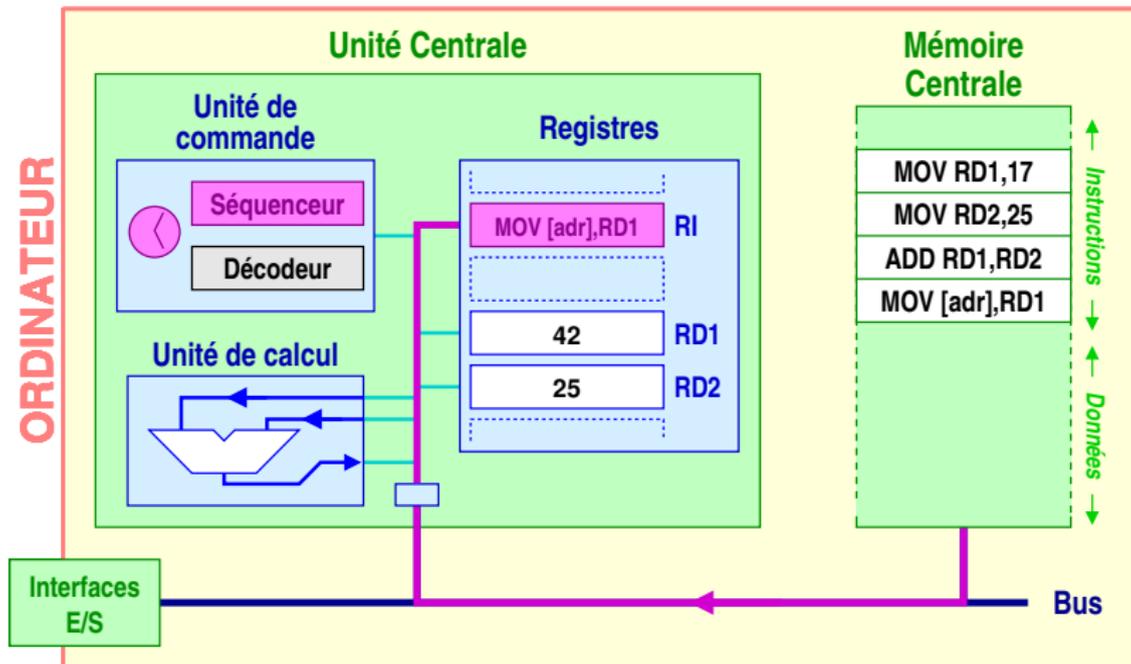
Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)



Simulation du fonctionnement (1)

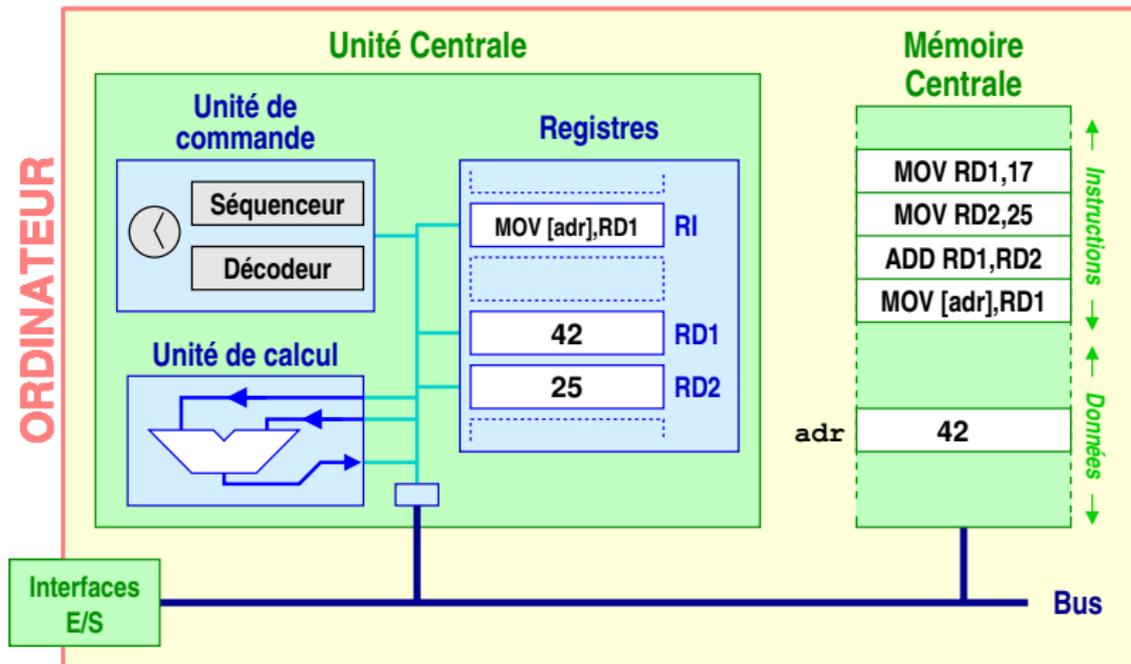
# Simulation du fonctionnement (1)



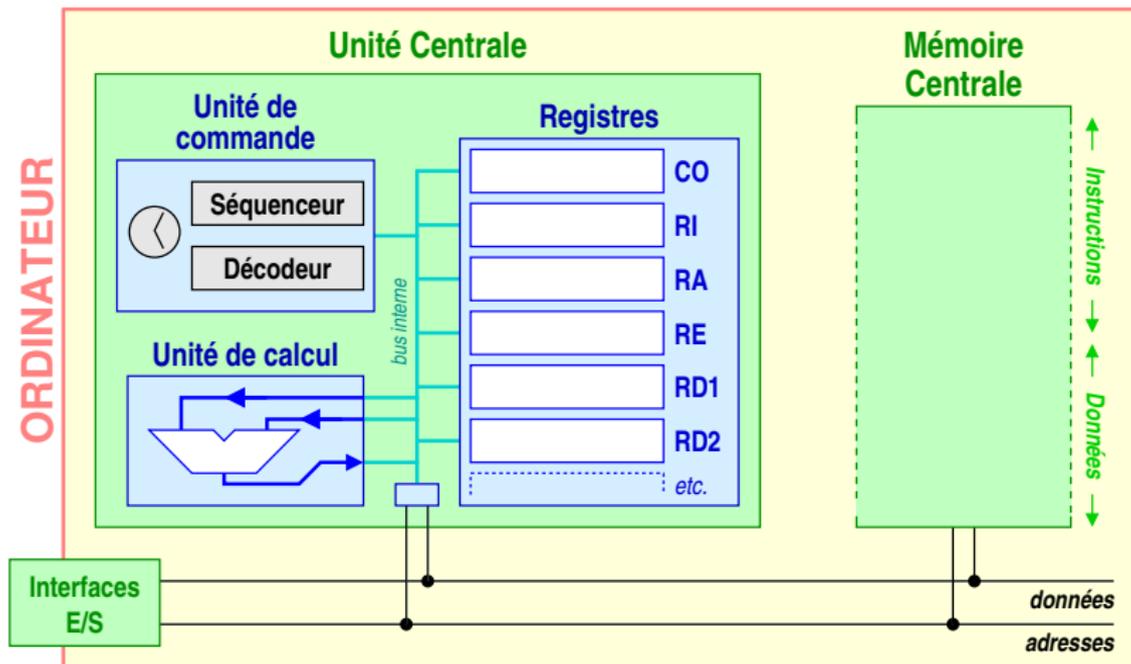


Simulation du fonctionnement (1)

# Simulation du fonctionnement (1)

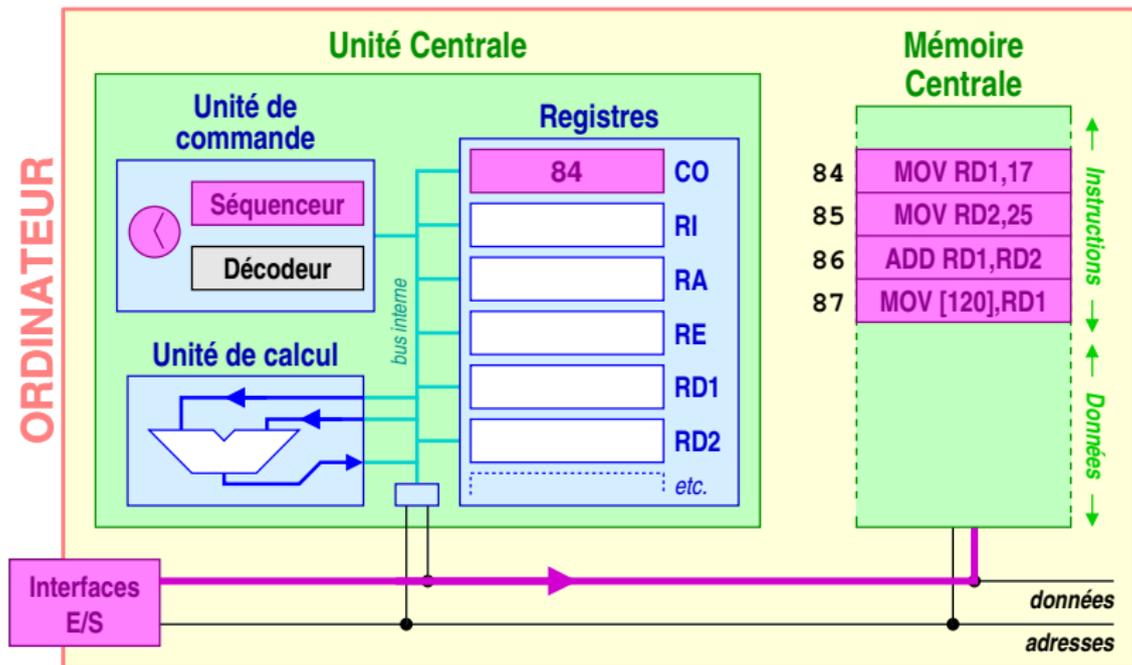


# Simulation du fonctionnement (2)



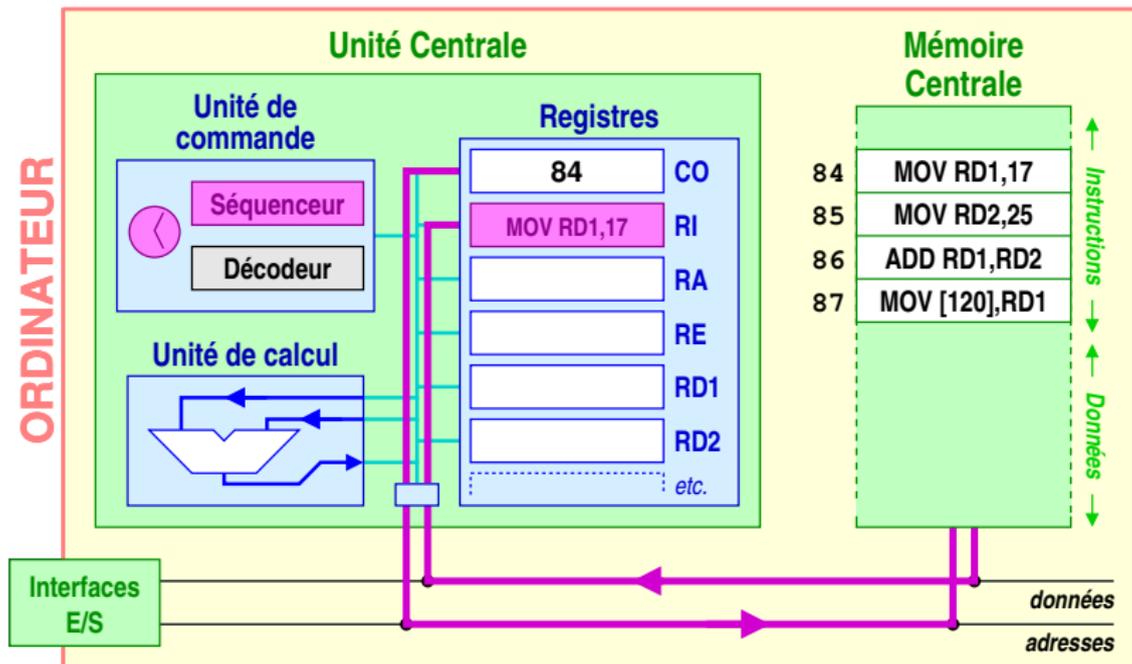
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



## Simulation du fonctionnement (2)

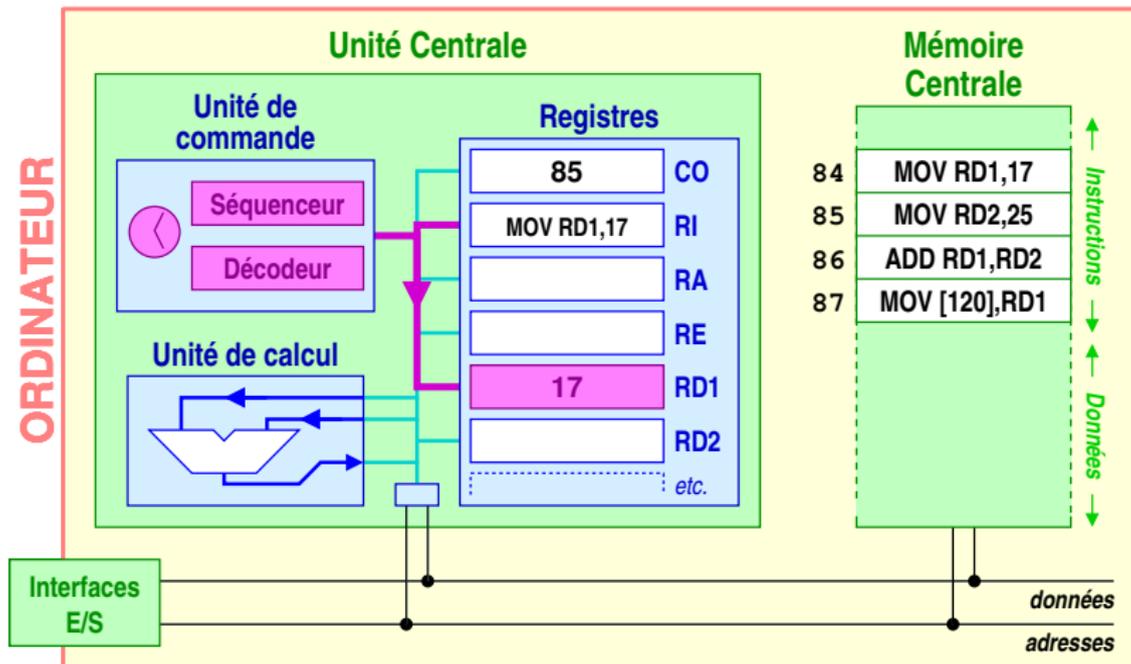
## Simulation du fonctionnement (2)





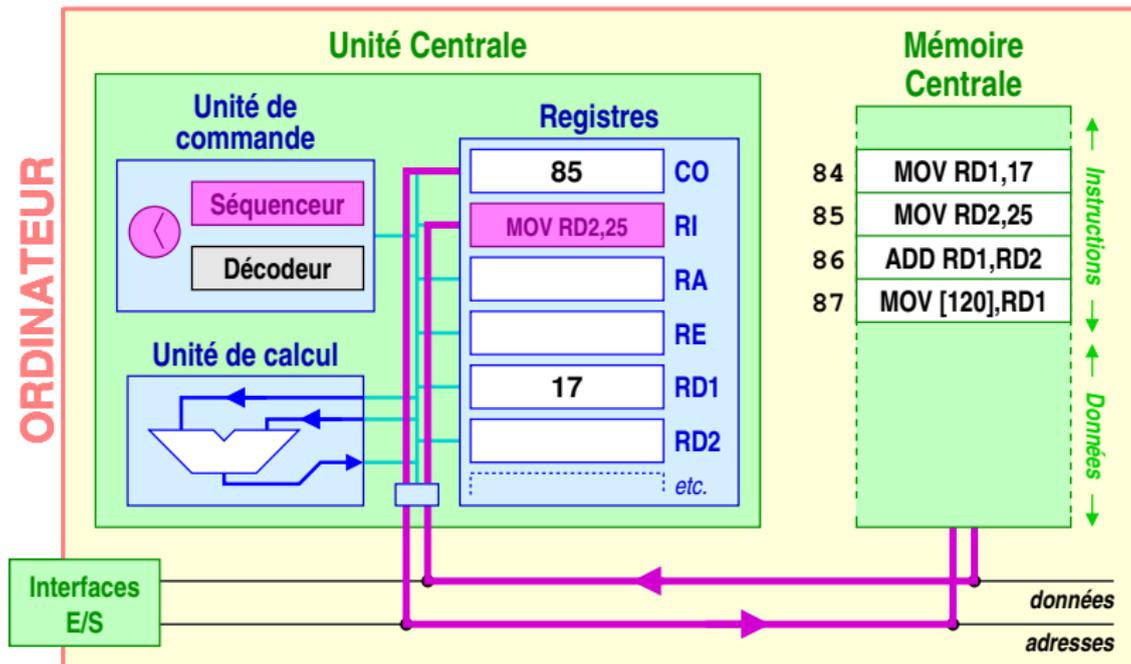
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



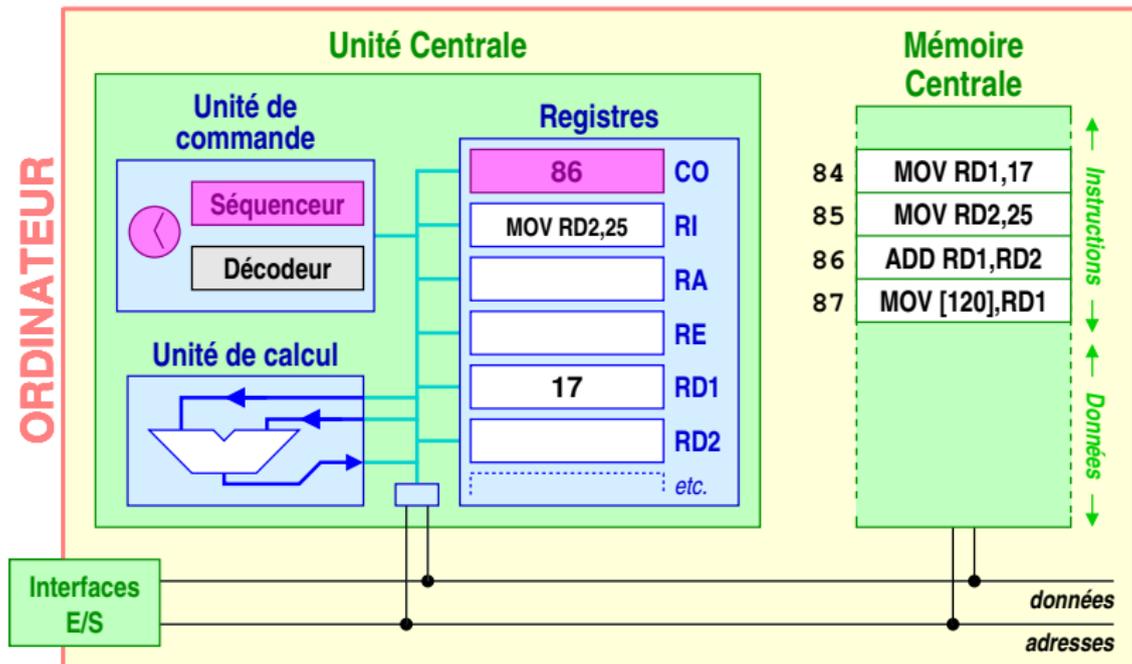
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



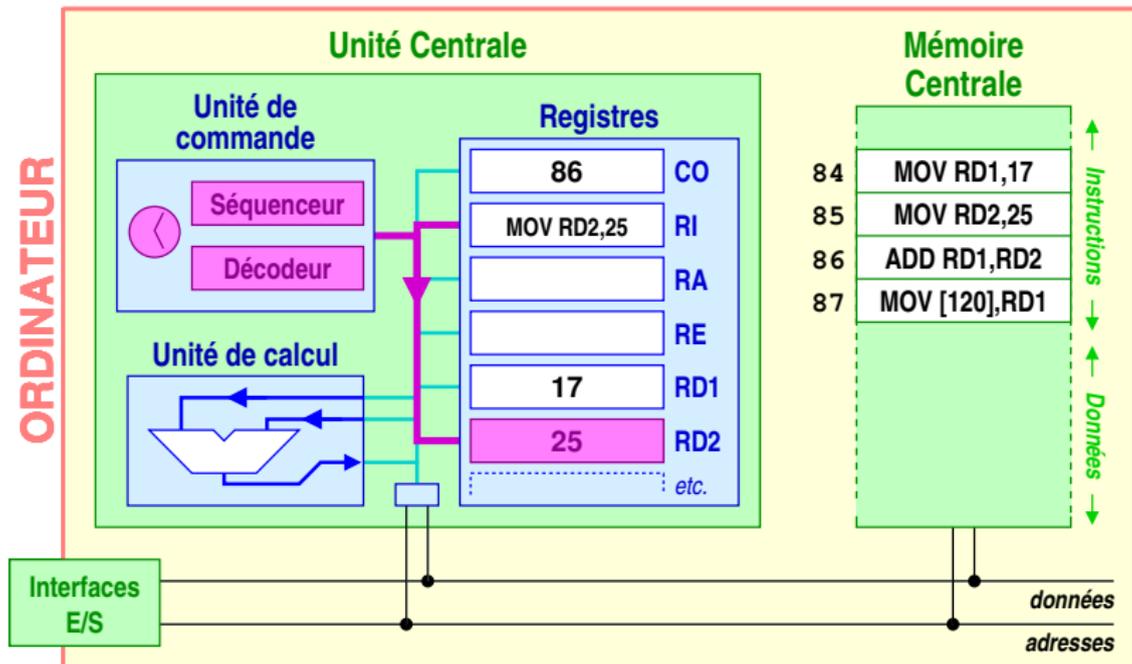
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



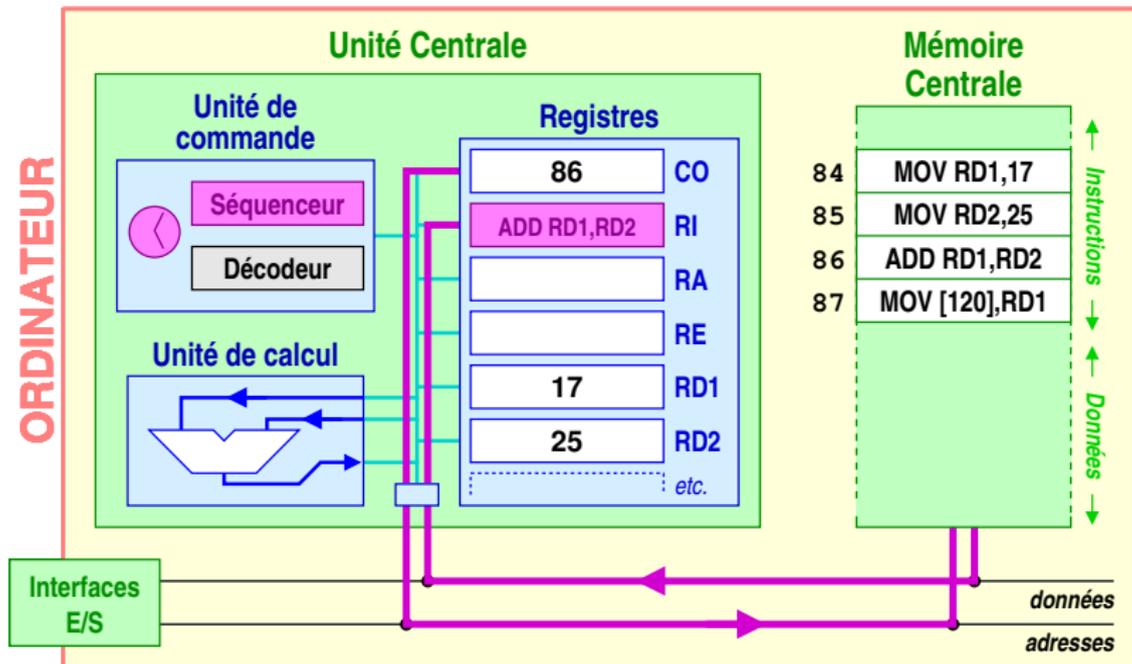
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



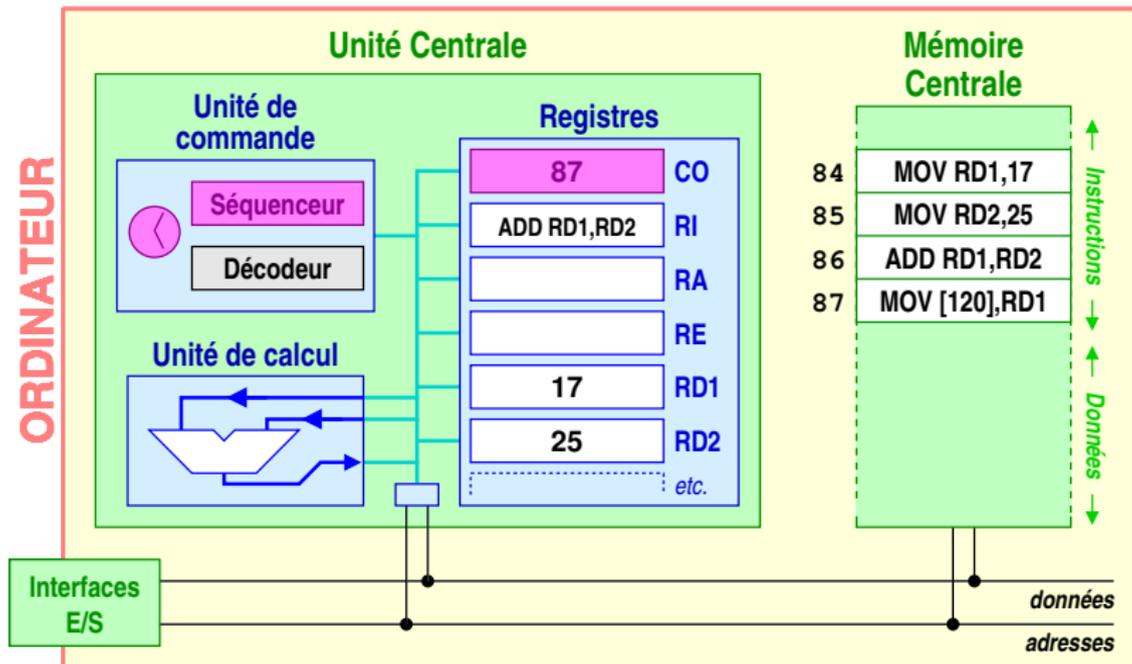
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



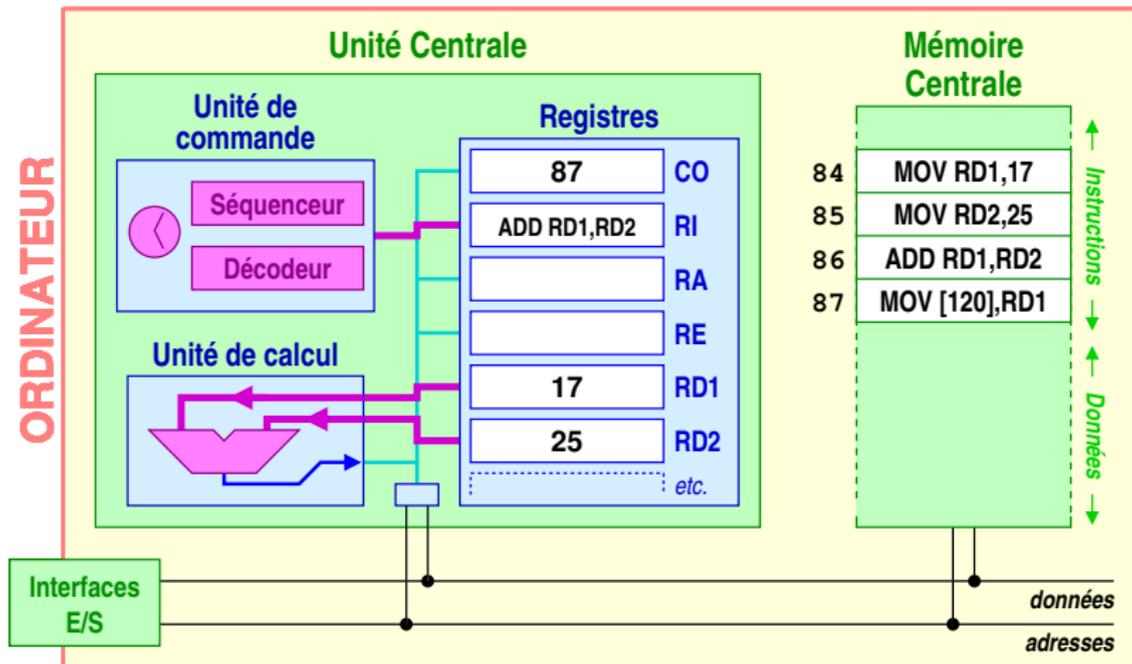
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)

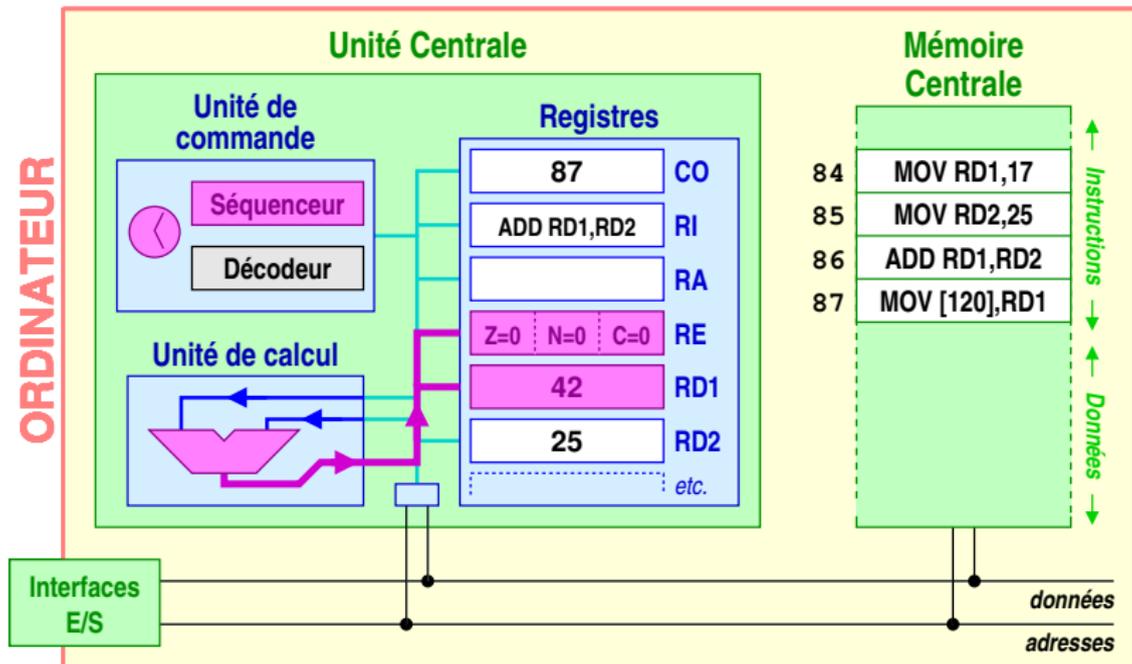


## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)

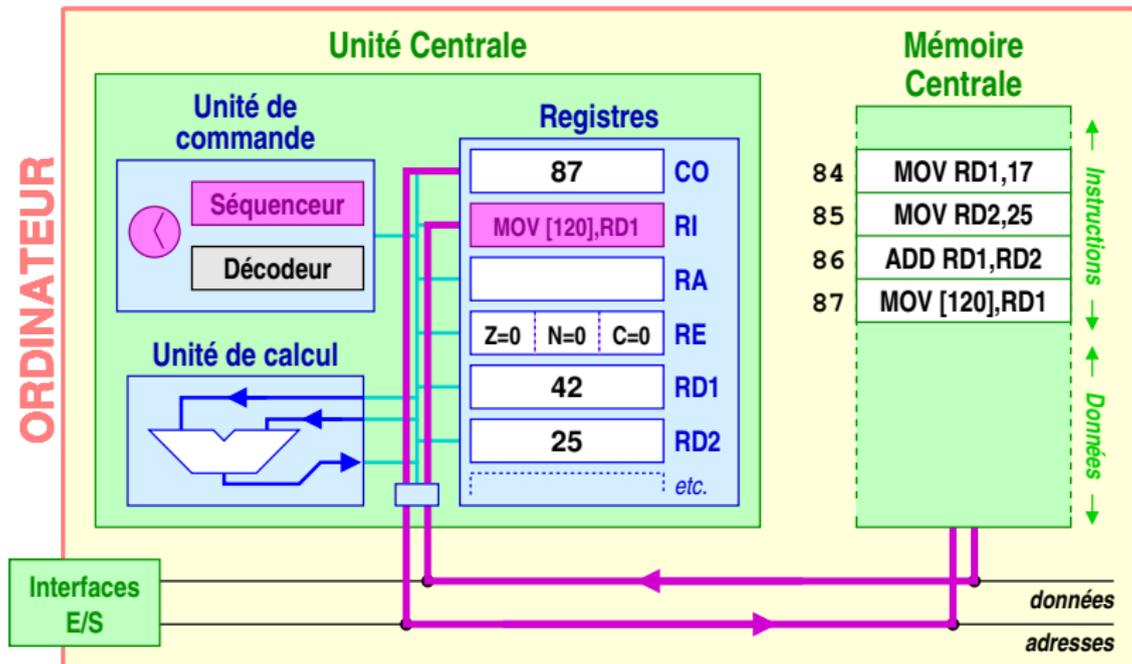


# Simulation du fonctionnement (2)

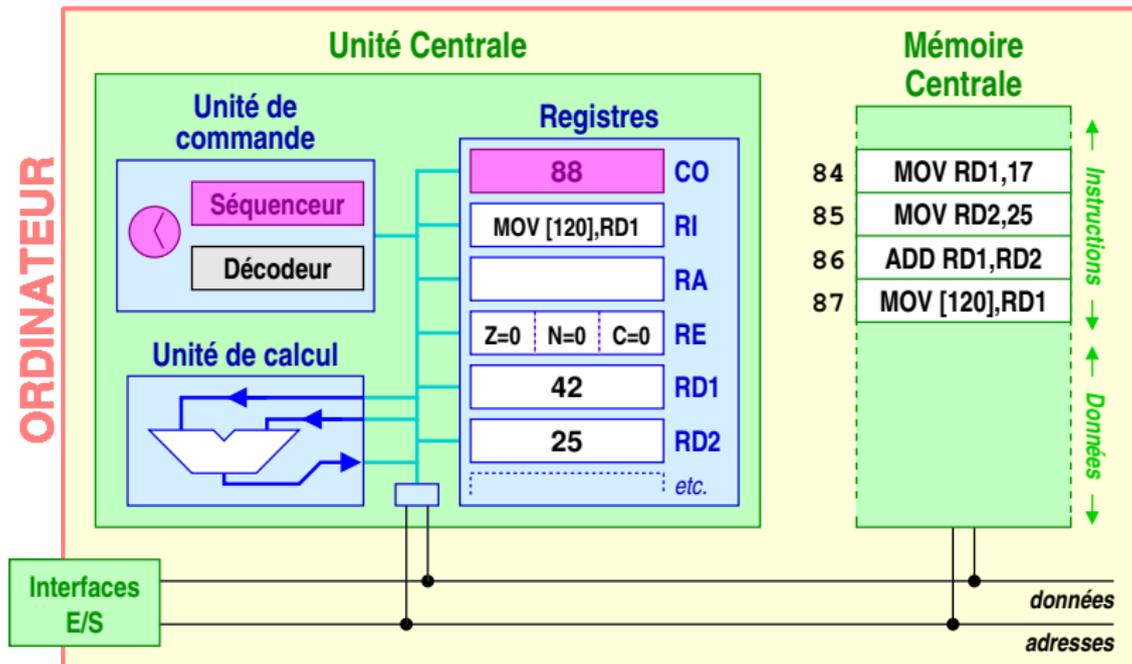


## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)

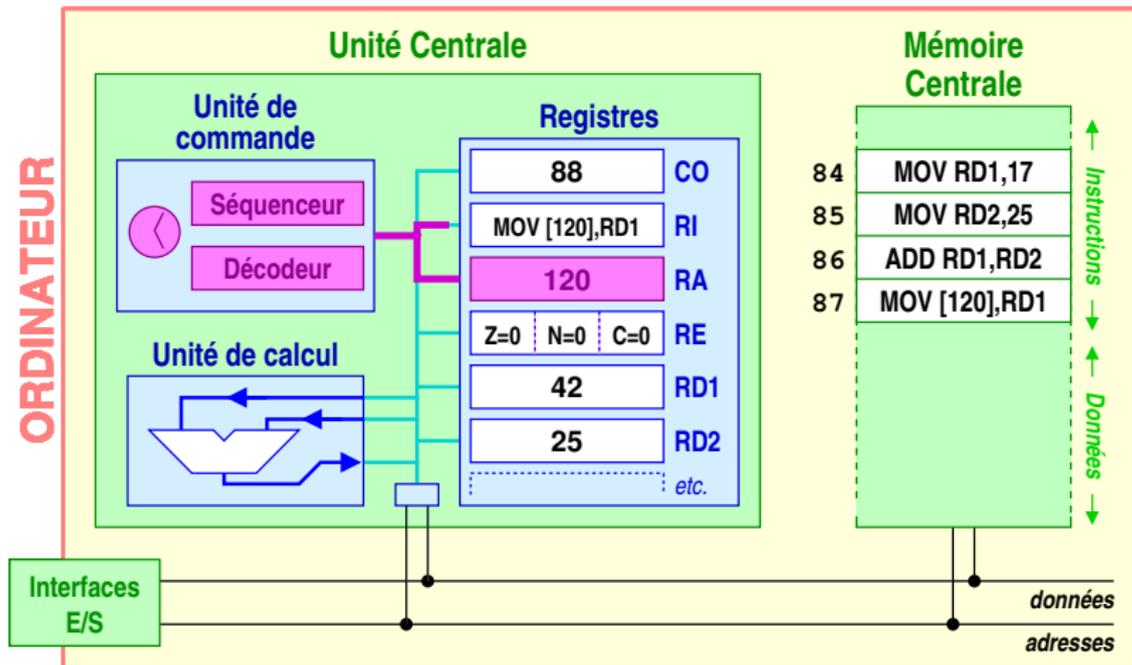


# Simulation du fonctionnement (2)



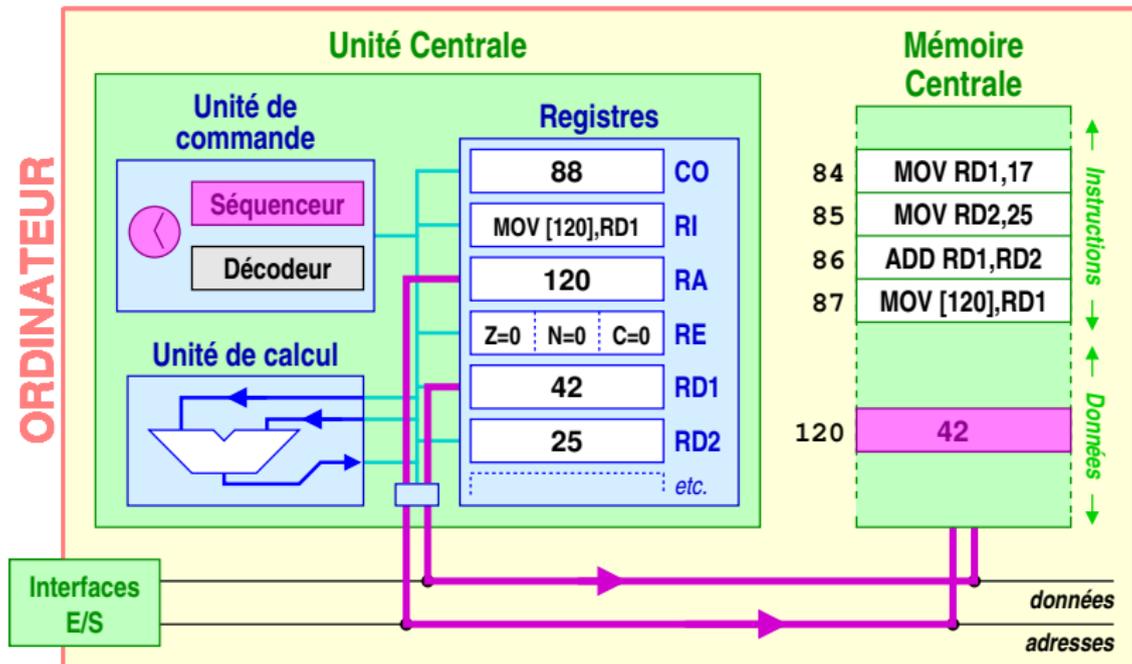
## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)

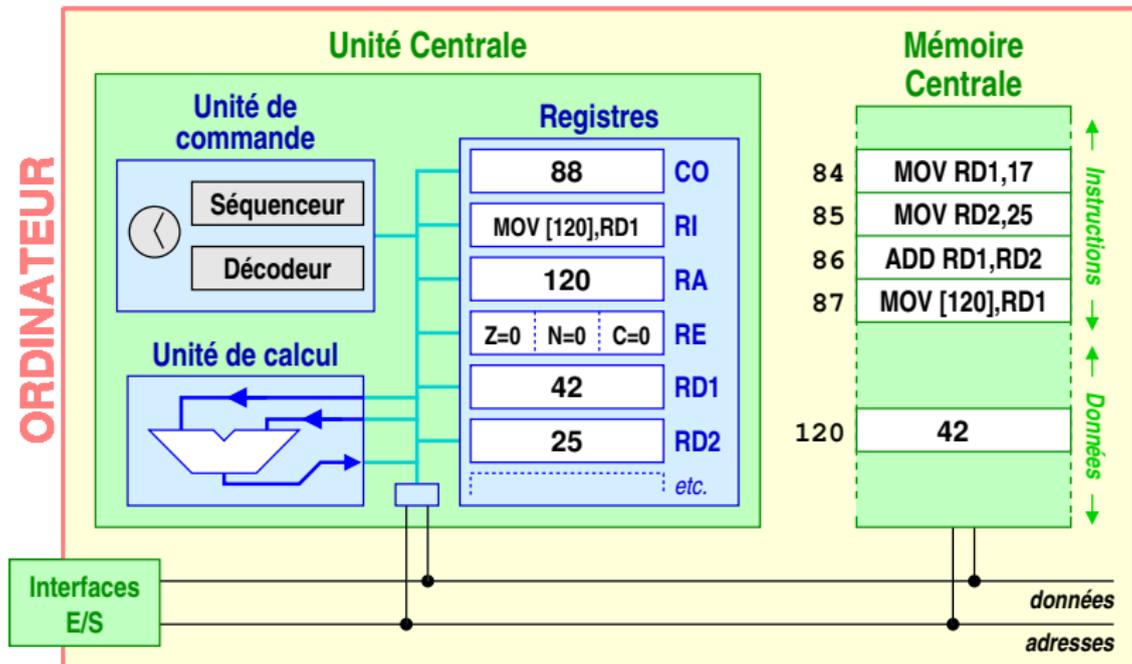


## Simulation du fonctionnement (2)

## Simulation du fonctionnement (2)



# Simulation du fonctionnement (2)



# Sommaire

- 1 Introduction
- 2 Historique
- 3 Structure de l'ordinateur
- 4 Fonctionnement de l'ordinateur
- 5 Architecture en couches**

# Architecture en couches

# Architecture en couches

## Ordinateur des années 50

# Architecture en couches

## Ordinateur des années 50

Deux couches logicielles viennent se superposer à la machine physique :

Couche	Langage(s)
Langage d'assemblage	Assembleur
Machine traditionnelle	Langage machine
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)
Machine traditionnelle	Langage machine
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)
Système d'exploitation	Langage de commande
Machine traditionnelle	Langage machine
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

Couche	Langage(s)
Langage d'assemblage	Assembleur
Système d'exploitation	Langage de commande
Machine traditionnelle	Langage machine
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

<b>Couche</b>	<b>Langage(s)</b>
Langages de programmation	Pascal, C, ADA, Prolog, etc.
Langage d'assemblage	Assembleur
Système d'exploitation	Langage de commande
Machine traditionnelle	Langage machine
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

## Découpage de référence

Le découpage de référence introduit par Andrew TANENBAUM compte 7 couches :

<b>Couche</b>	<b>Langage(s)</b>
Programmes d'applications	
Langages de programmation	Pascal, C, ADA, Prolog, etc.
Langage d'assemblage	Assembleur
Système d'exploitation	Langage de commande
Machine traditionnelle	Langage machine
Micro-programmée	Micro-programme (ROM)
Physique	Combinaison de portes logiques

# Architecture en couches

# Architecture en couches

## Modèle simplifié

# Architecture en couches

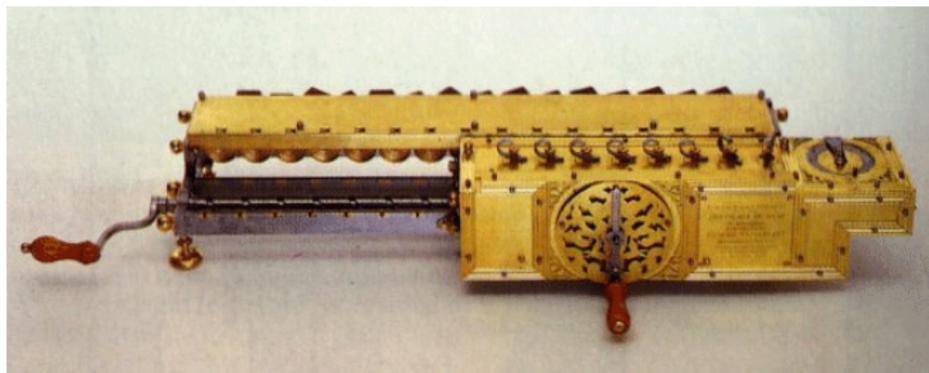
## Modèle simplifié

Couche	Langage(s)
Langages de programmation	Pascal, C, ADA, Prolog, etc.
Système d'exploitation	Langage de commande
Machine traditionnelle	Langage machine
Physique	Combinaison de portes logiques

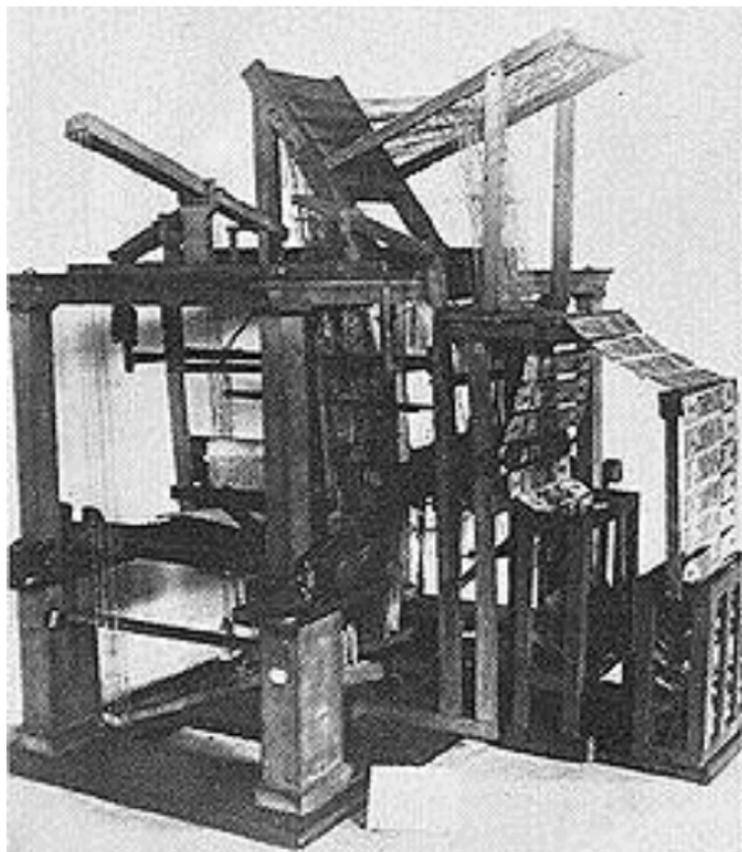
# La Pascaline



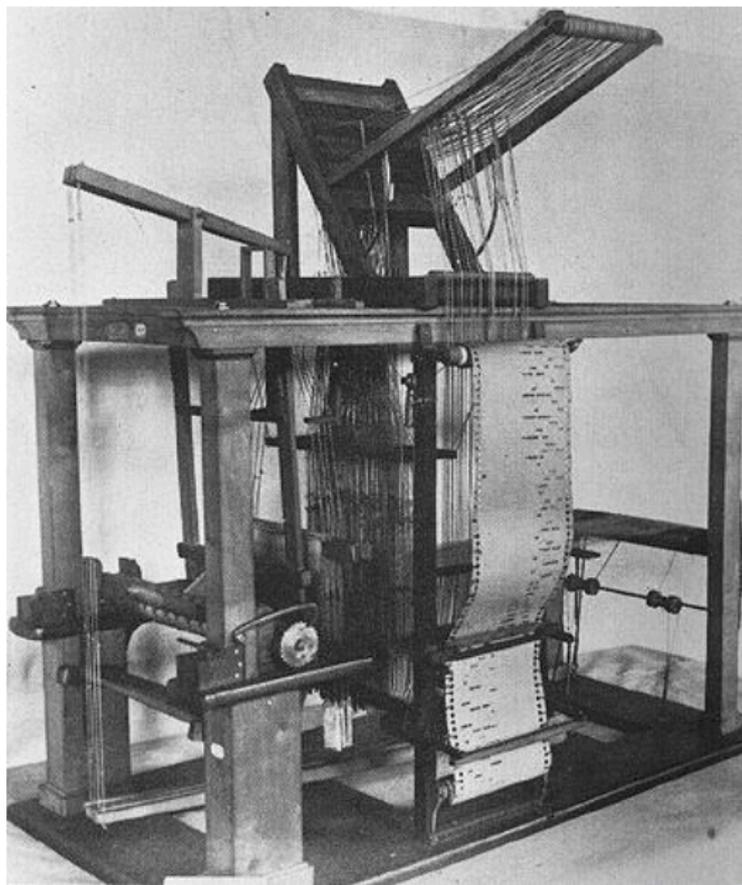
# La machine de Leibnitz



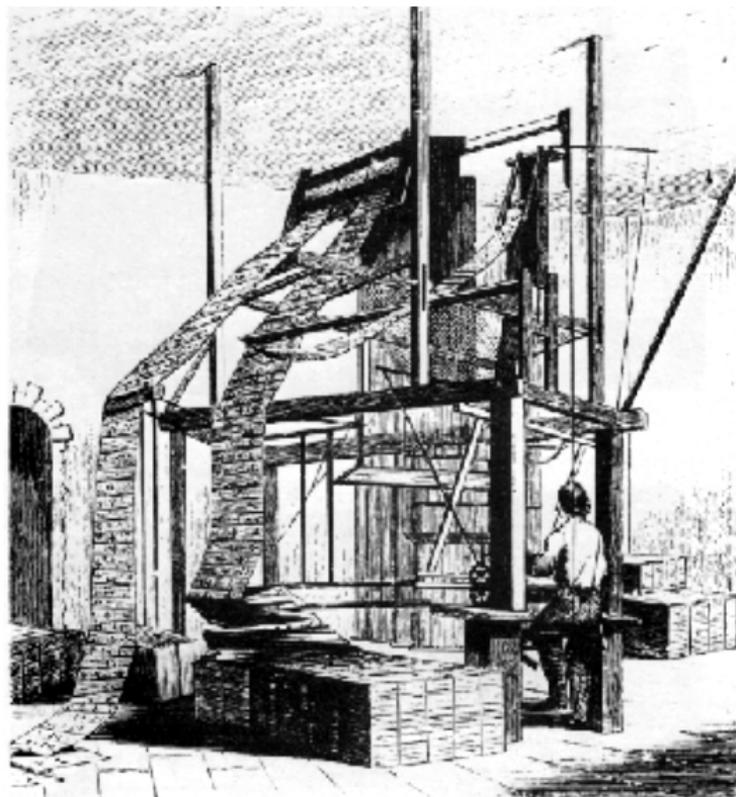
# La machine de Falcon



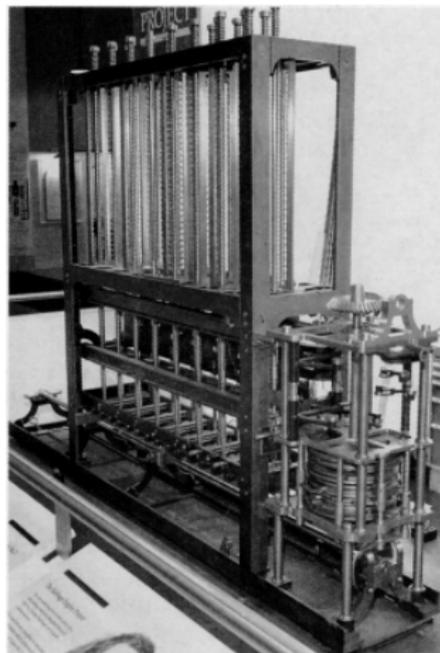
# La machine de Bouchon



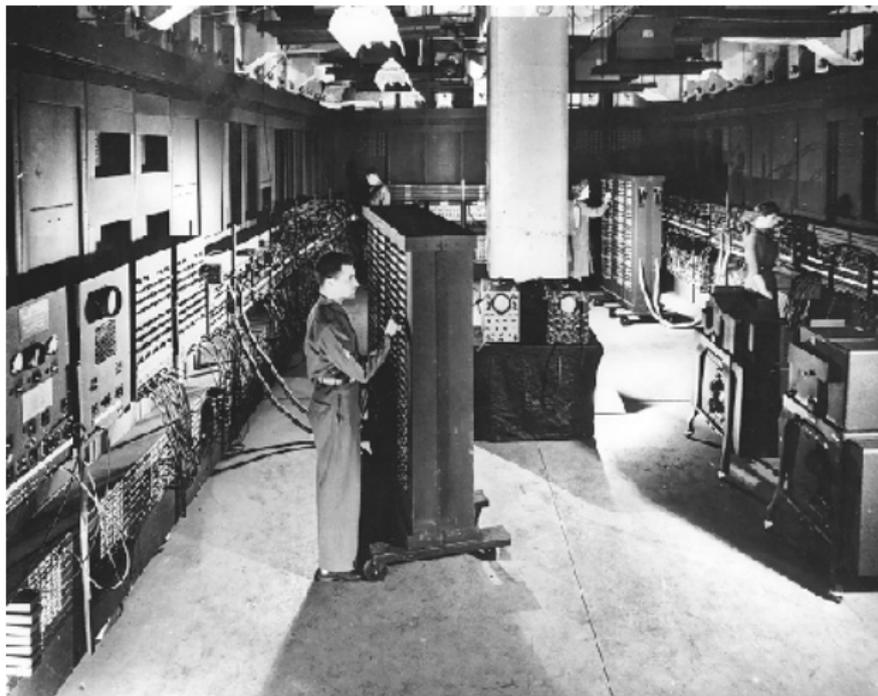
# La machine de Jacquard



# La machine de Babbage



# L'Eniac



# Von Neumann

