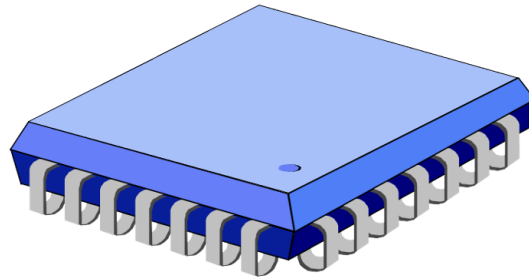


Cours

Architecture des microprocesseurs



doudou.dione@esp.sn

❑ Introduction

- ✓ Microprocesseur

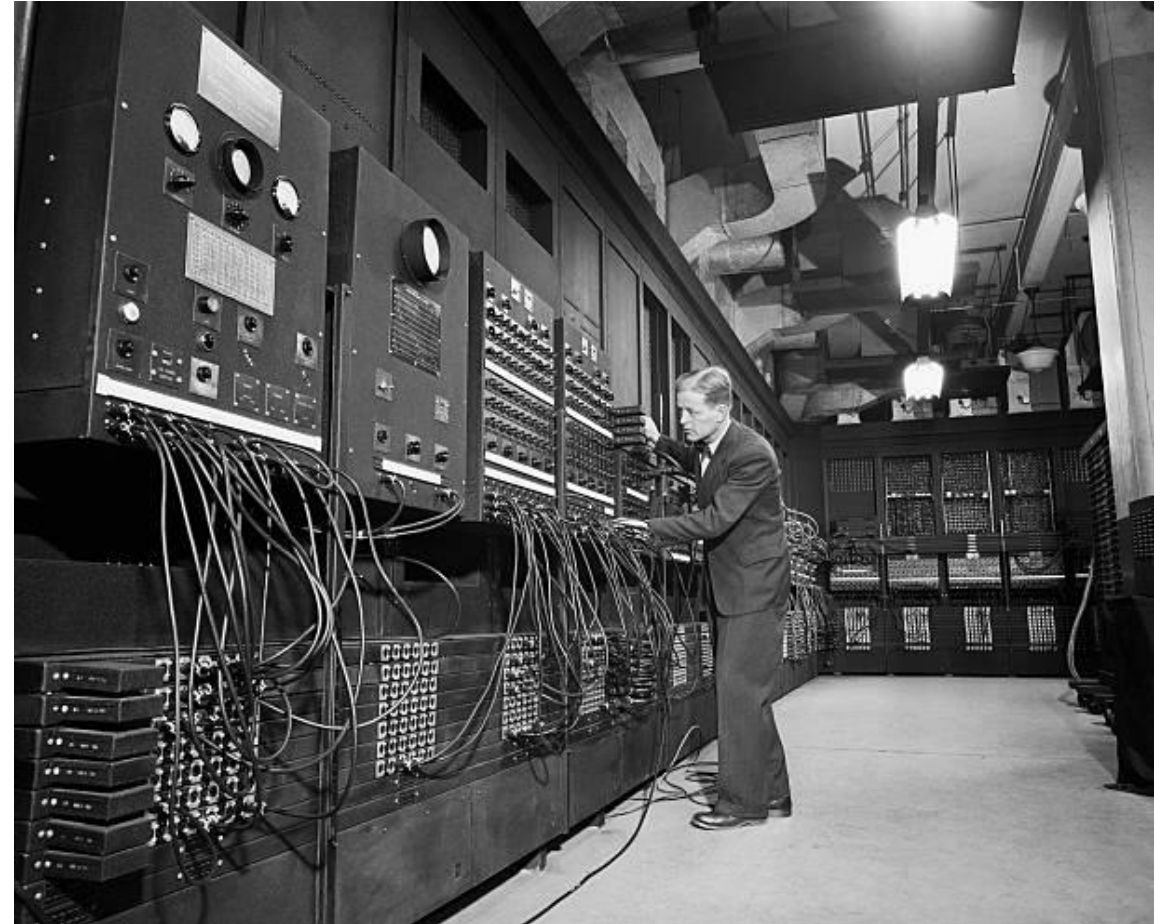
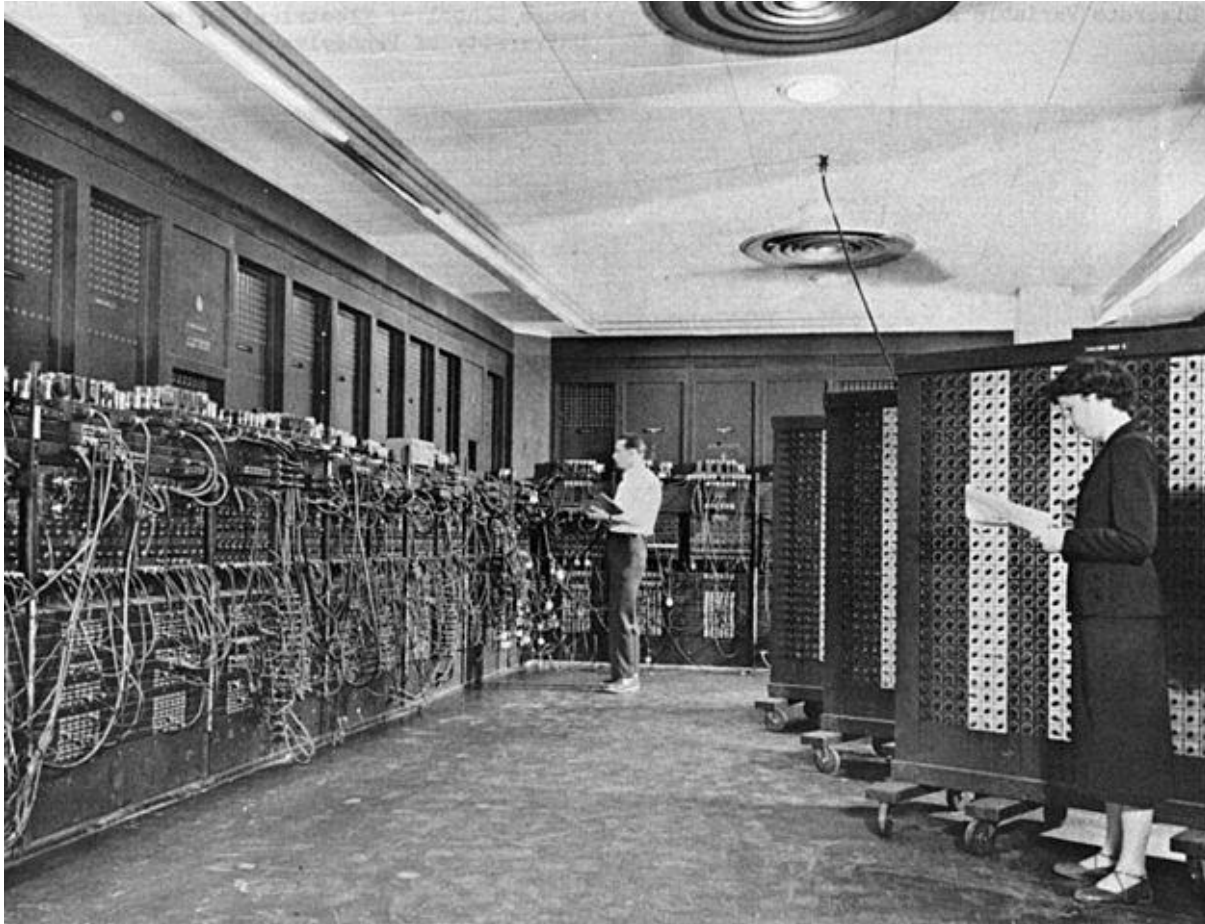
Composant électronique d'un ordinateur qui traite des données à l'aide d'instructions machines



❑ Introduction

✓ Historique

1945



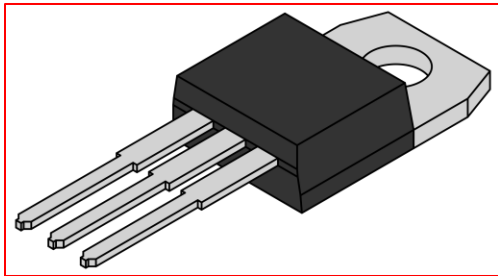
Le premier ordinateur : Electronic Numerical Integrator And Computer (ENIAC), une machine qui pesait 30 tonnes et occupait plus de 160m²

❑ Introduction

✓ Historique

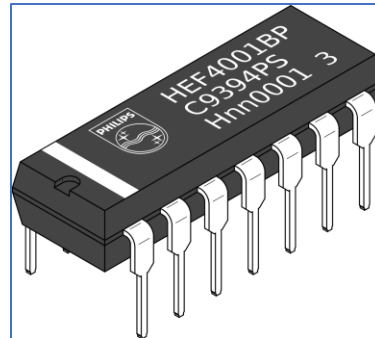
Inventions des premiers circuit intégrés et des transistors

1947



Transistor

1958



Circuit intégré



❑ Introduction

- ✓ Historique



❑ Introduction

✓ Historique

Résumé

Processeur	Année	Caractéristiques principales
Transistors	Années 50-60	Élimination des tubes à vide et des relais électromécaniques .
Intel 4004	1971	Premier processeur à puce unique, 4 bits, capable de 60 000 opérations par seconde .
Intel 8086	1978	Utilisé dans les premiers PC d'IBM, marquant l'ère des microprocesseurs .
Processeurs multi-cœur	Années 2000	Émergence des processeurs multicœur en raison de la dissipation thermique .

❑ Introduction

- ✓ Différence entre processeur et microprocesseur

Aspect	Processeur	Microprocesseur
Définition	Composant électronique qui exécute des instructions et effectue des calculs dans un système informatique.	Type de processeur conçu sur une seule puce de circuit intégré.
Applications	Utilisé dans divers systèmes informatiques, tels que les serveurs, les ordinateurs centraux et les superordinateurs.	Couramment utilisé dans l'électronique grand public et les systèmes embarqués.
Complexité	Peut être plus grand et plus puissant, adapté aux tâches complexes.	Plus petit et plus spécialisé, adapté aux systèmes embarqués et autres appareils.
Architecture	Peut avoir plusieurs cœurs et une vitesse d'horloge variable.	Comprend des millions, voire des milliards, de transistors sur une seule puce de silicium.
Fonctions	Gère les opérations, les entrées/sorties et coordonne les activités d'autres composants matériels et logiciels.	Effectue des opérations arithmétiques, logiques et de contrôle, manipulant des données.

❑ Introduction

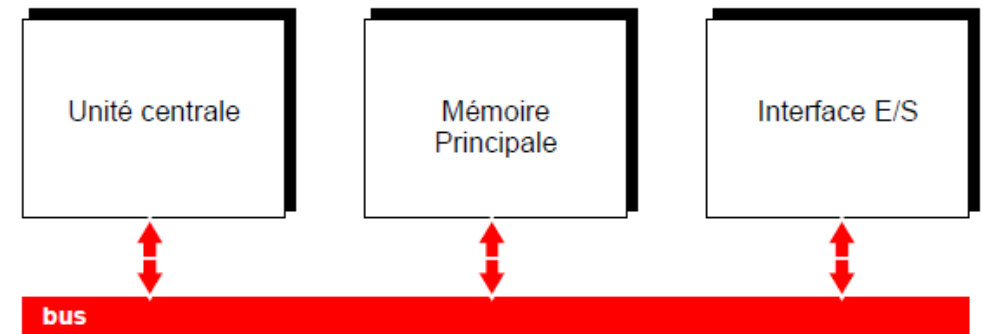
Concepts de base

✓ Modèle de Von Neuman (1945)

- Conçue par le mathématicien et physicien John Von Neumann en 1945.
- Les données du programme et les instructions sont stockées dans la même mémoire.
- Utilise un bus commun pour transférer les données et les instructions.
- Nécessite deux cycles d'horloge pour exécuter une instruction.
- Coûte moins cher et est utilisée dans les ordinateurs personnels et les petits systèmes

Décompose l'ordinateur en 4 parties :

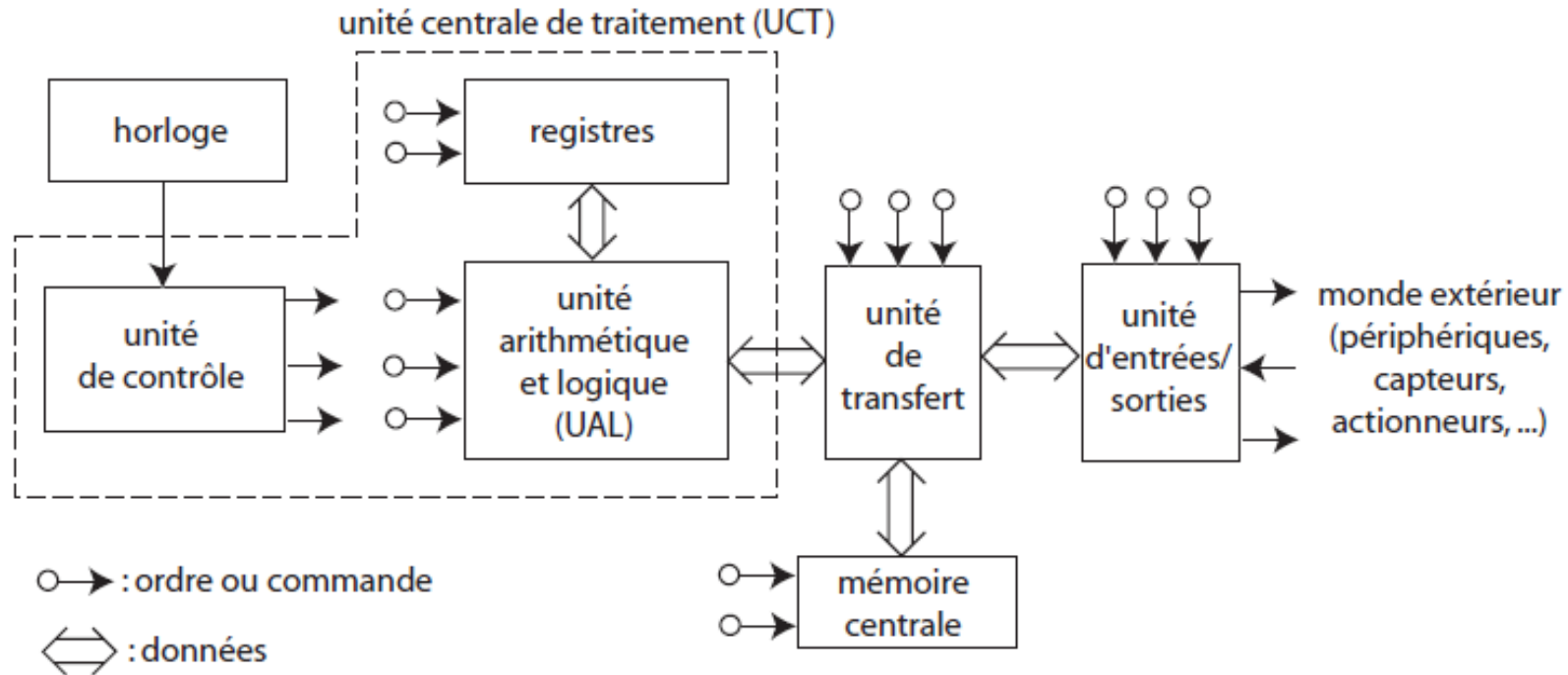
- Unité Arithmétique et Logique (UAL)
- Unité de contrôle
- Mémoire
- Entrées/Sorties



❑ Introduction

Concepts de base

Architecture d'un microprocesseur



Un microprocesseur consiste en une unité centrale de traitement (UAL + registres + unité de contrôle) entièrement contenue dans un seul circuit intégré

❑ Introduction

Concepts de base

✓ Les registres

Les registres sont des emplacements de stockage temporaires situés à l'intérieur du processeur. Ils sont utilisés pour stocker des données et des instructions pendant l'exécution d'un programme.

Les registres sont plus rapides que la mémoire principale (RAM) car ils sont intégrés directement dans le processeur.

Il existe plusieurs types de registres :

- registre d'instruction (contenant l'instruction en cours d'exécution),
- registre d'adresse (stockant l'adresse mémoire),
- registre d'accumulateur (utilisé pour les opérations arithmétiques).

❑ Introduction

Concepts de base

- ✓ Les registres

- registre d'instruction

Nom du Registre	Description
PC (Program Counter)	Contient l'adresse mémoire de la prochaine instruction à exécuter.
IR (Instruction Register)	Stocke l'instruction actuellement en cours d'exécution.
MAR (Memory Address Register)	Contient l'adresse mémoire d'une donnée à lire ou à écrire.
MDR (Memory Data Register)	Stocke temporairement les données lues depuis ou écrites vers la mémoire.
R0, R1, R2, ...	Registres généraux pour stocker des données temporaires.
SP (Stack Pointer)	Pointe vers le sommet de la pile (utilisé pour les appels de fonctions).
FP (Frame Pointer)	Pointe vers le cadre de la fonction en cours d'exécution.

❑ Introduction

Concepts de base

✓ Les instructions

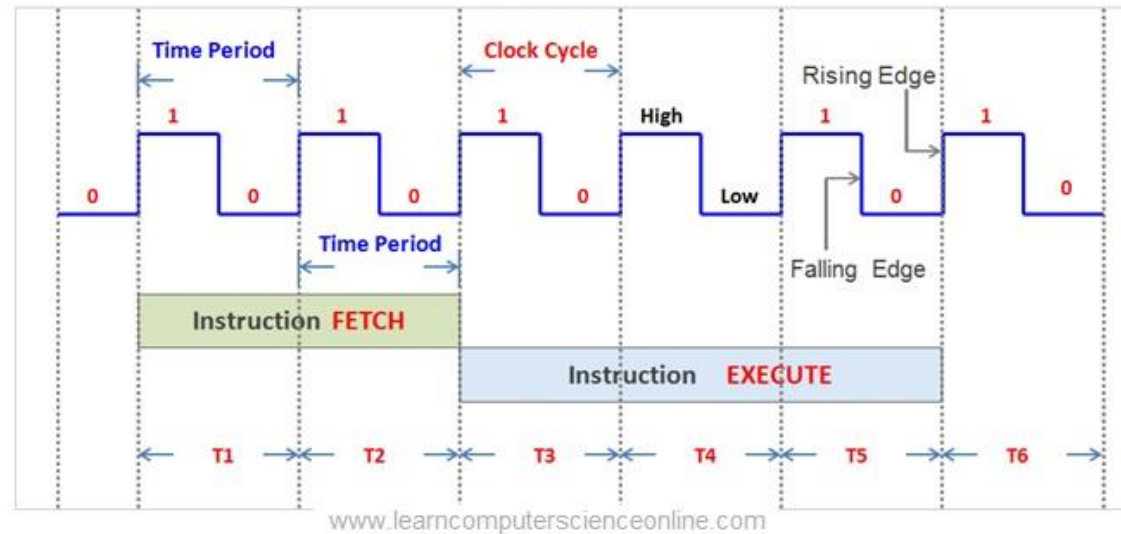
- Les instructions sont les commandes élémentaires comprises par le processeur. Elles décrivent les opérations à effectuer (comme l'addition, la soustraction, le chargement de données, etc.).
- Les instructions sont codées en langage machine (binaire) et sont exécutées séquentiellement par le processeur.

❑ Introduction

Concepts de base

- ✓ Les cycles d'horloge

- Le processeur fonctionne en suivant une horloge interne. Chaque cycle d'horloge représente une unité de temps.
- Pendant chaque cycle d'horloge, le processeur effectue une opération (lecture d'instruction, exécution d'une instruction, etc.).
- La vitesse d'horloge (mesurée en hertz) détermine la rapidité d'exécution des instructions.

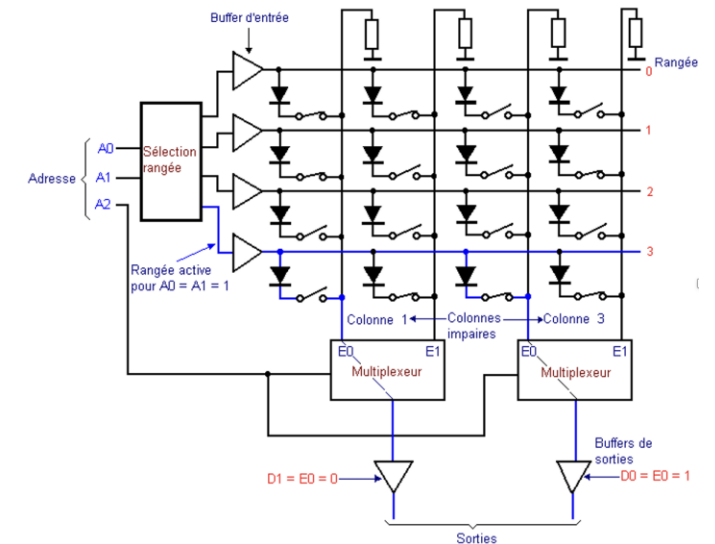
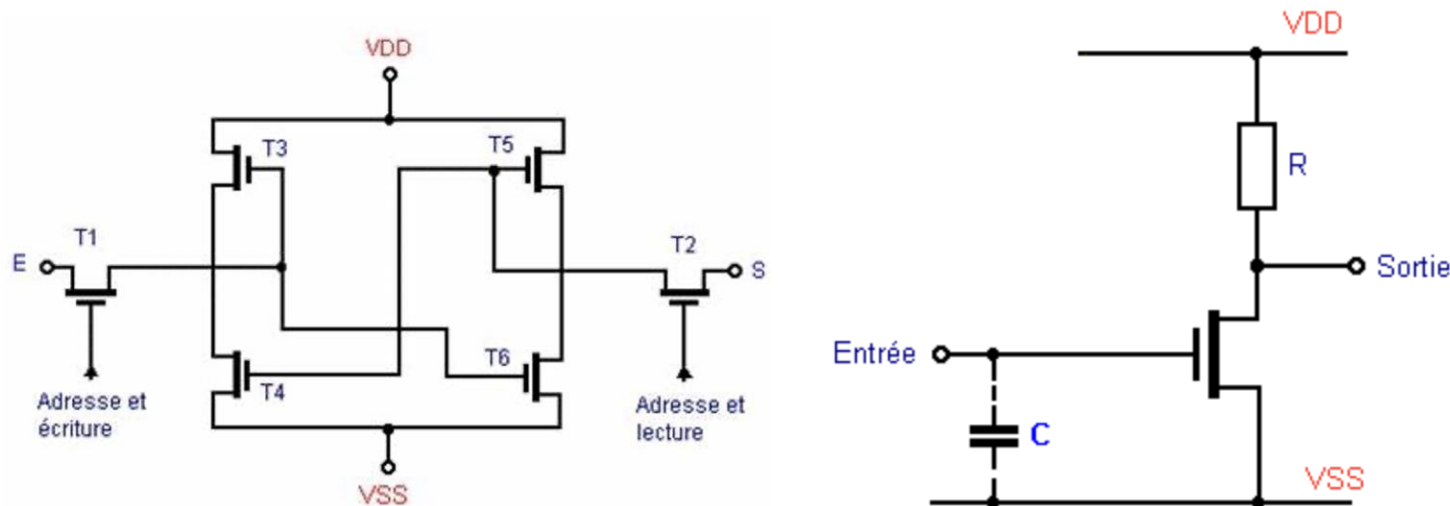


□ Introduction

Concepts de base

✓ Les niveaux de mémoire

- Les niveaux de mémoire désignent les différentes couches de stockage utilisées par le processeur.
- Registres: Les plus rapides, mais aussi les plus petits.
- Cache: Mémoire intermédiaire entre les registres et la RAM.
- RAM (mémoire principale): Plus grande, mais plus lente que le cache.
- Disque dur/SSD: Mémoire de stockage permanente, utilisée pour stocker des programmes et des données.



❑ Introduction

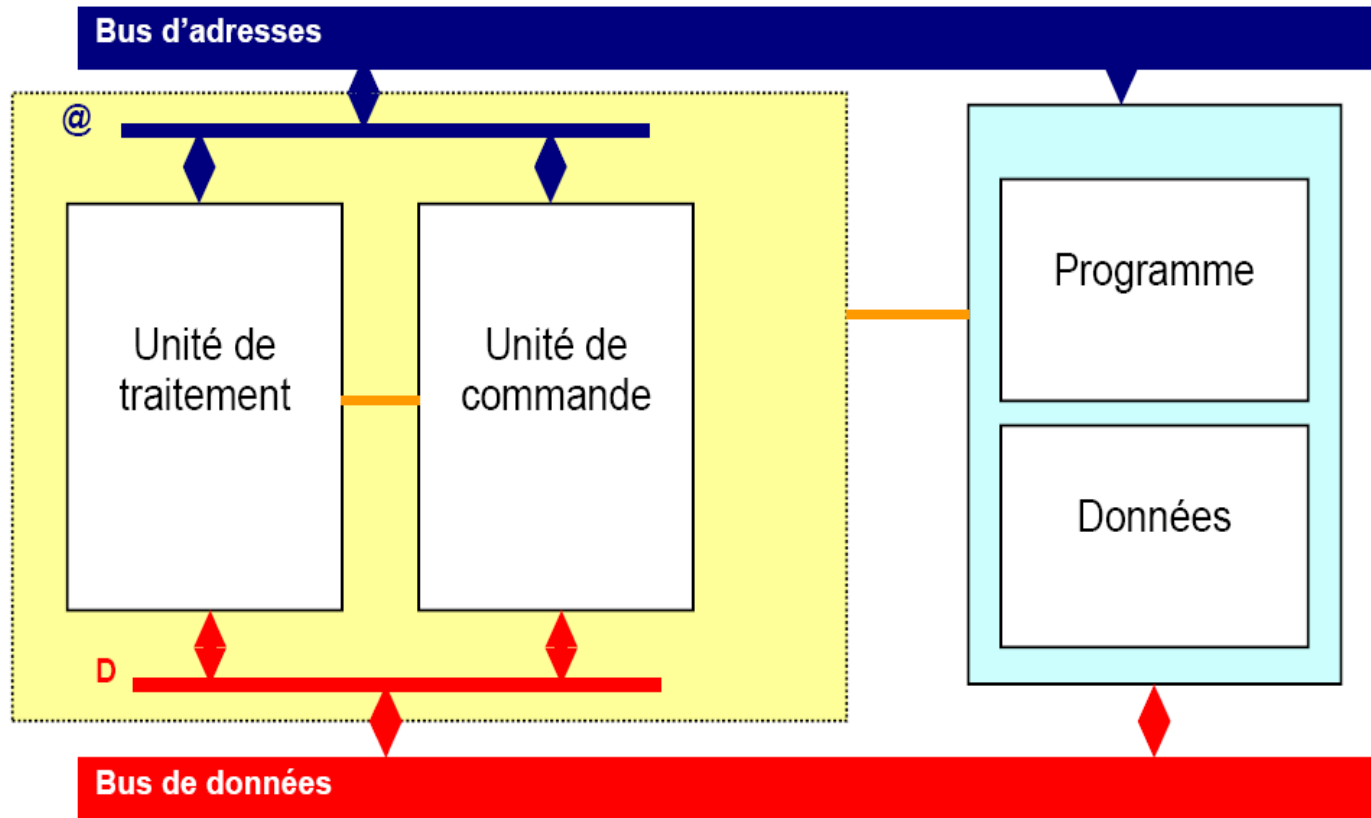
Concepts de base

Evolution des processeurs

Processeur	Cœurs	Transistors	Gravure
Intel 4004	1	2300	10 000
Intel 8080	1	4500	6 000
Core 2 Duo	2	+400 Millions	45
.....
Core iX	8	Plusieurs milliards	10

❑ Le microprocesseur

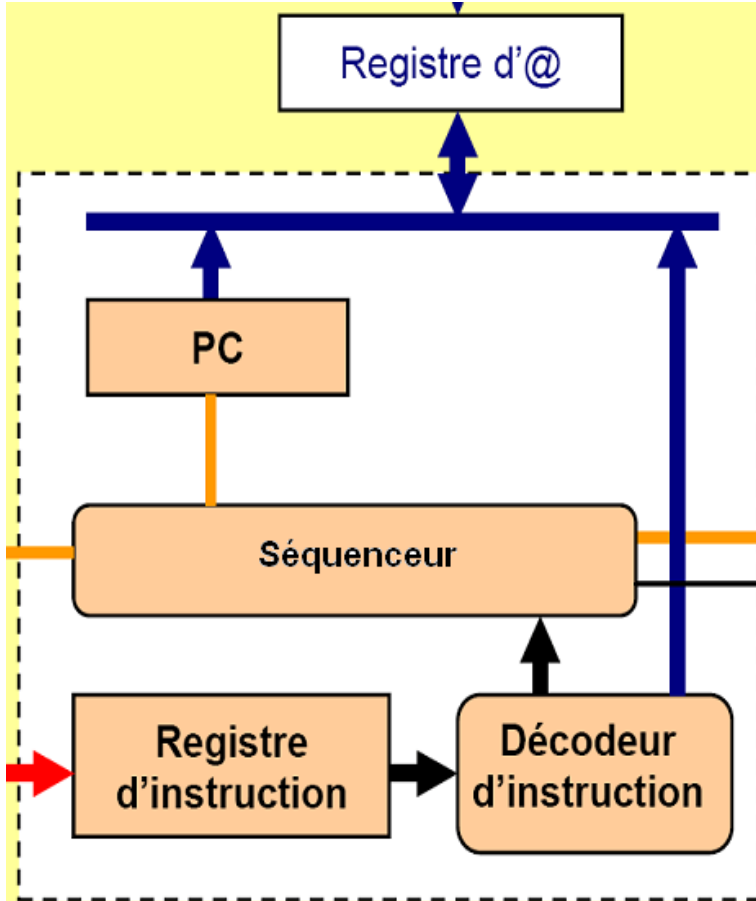
- ✓ Architecture interne



❑ Le microprocesseur

- ✓ Architecture interne

L'unité de commande

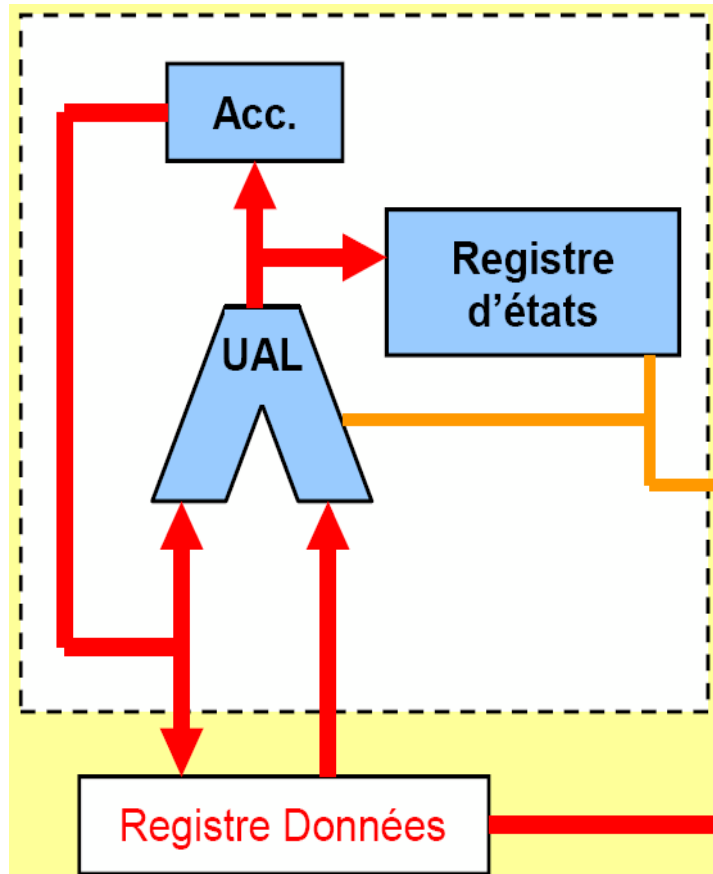


1. **Le compteur de programme (PC : Programme Counter)**
appelé aussi Compteur Ordinal (CO) est constitué par un registre dont le contenu est initialisé avec l'adresse de la première instruction du programme. Il contient toujours l'adresse de la prochaine instruction à exécuter.
2. **Le registre d'instruction et le décodeur d'instruction** : Chacune des instructions à exécuter est transféré depuis la mémoire dans le registre instruction puis est décodée par le décodeur d'instruction.
3. **Bloc logique de commande (ou séquenceur)** : Il organise l'exécution des instructions au rythme d'une horloge. Il élabore tous les signaux de synchronisation internes ou externes (bus de commande) du microprocesseur en fonction de l'instruction qu'il a à exécuter. Il s'agit d'un automate réalisé de façon micro-programmée.

❑ Le microprocesseur

✓ Architecture interne

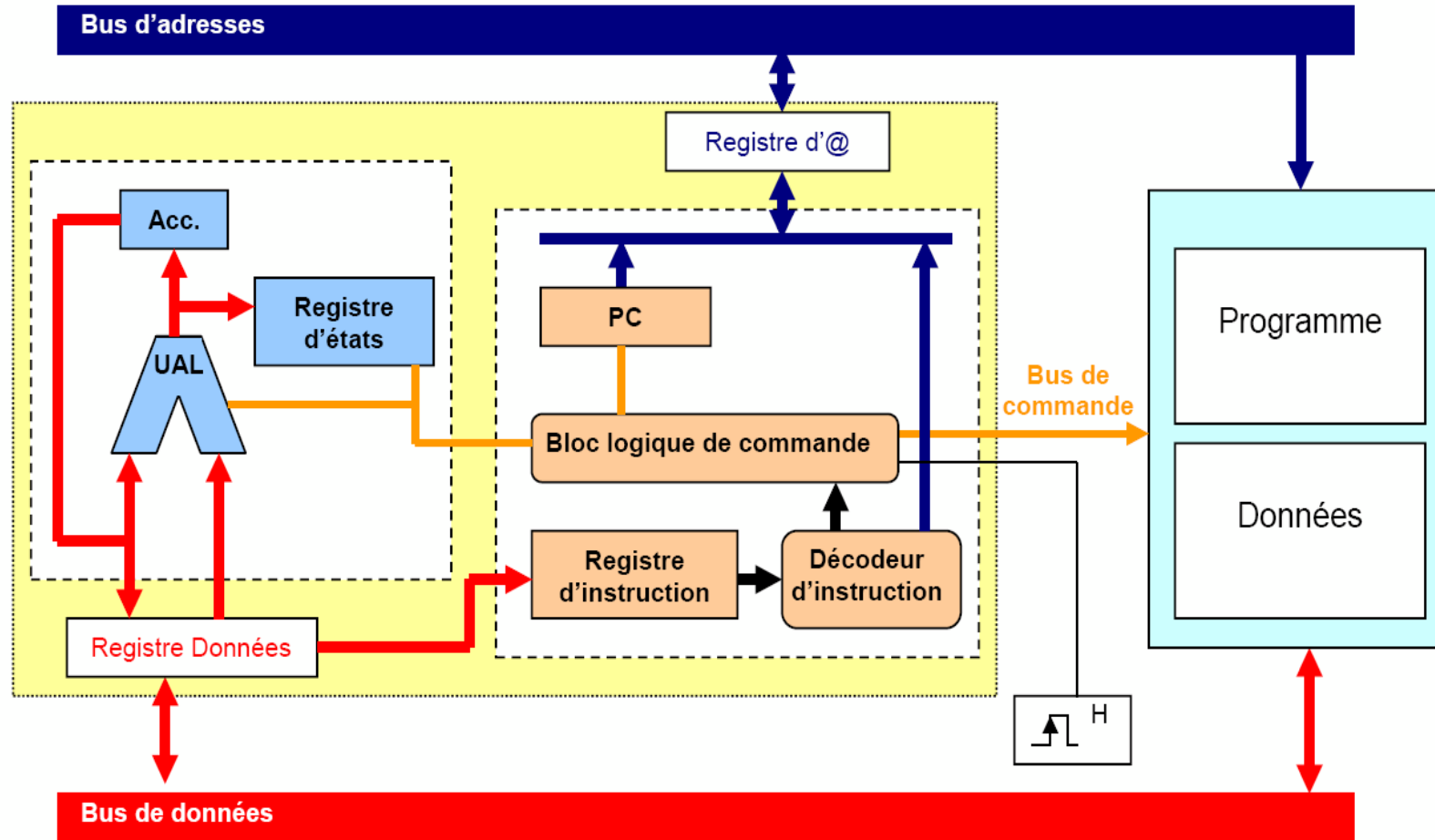
L'unité de traitement



1. **Les accumulateurs** sont des registres de travail qui servent à stocker une opérande au début d'une opération arithmétique et le résultat à la fin de l'opération.
2. **L'Unité Arithmétique et Logique (UAL)** est un circuit complexe qui assure les fonctions logiques (ET, OU, Comparaison, Décalage, etc...) ou arithmétique (Addition, soustraction...).
3. **Le registre d'état** est généralement composé de 8 ou 16 bits à considérer individuellement. Chacun de ces bits est un indicateur dont l'état dépend du résultat de la dernière opération effectuée par l'UAL. On les appelle *indicateur d'état* ou *flag* ou *drapeaux*. Dans un programme le résultat du test de leur état conditionne souvent le déroulement de la suite du programme.

❑ Le microprocesseur

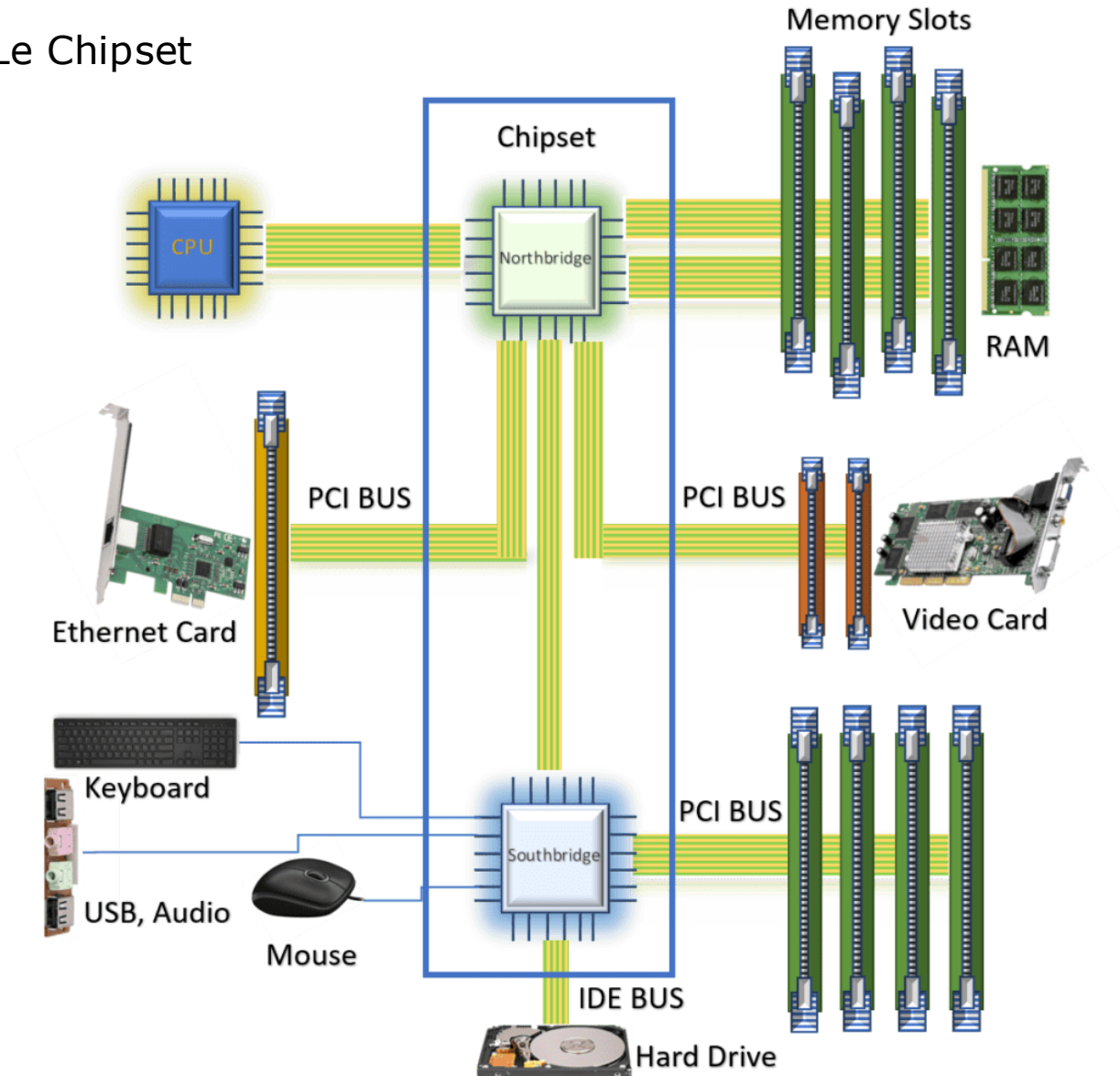
✓ Architecture interne



❑ Introduction

Concepts de base

Le Chipset



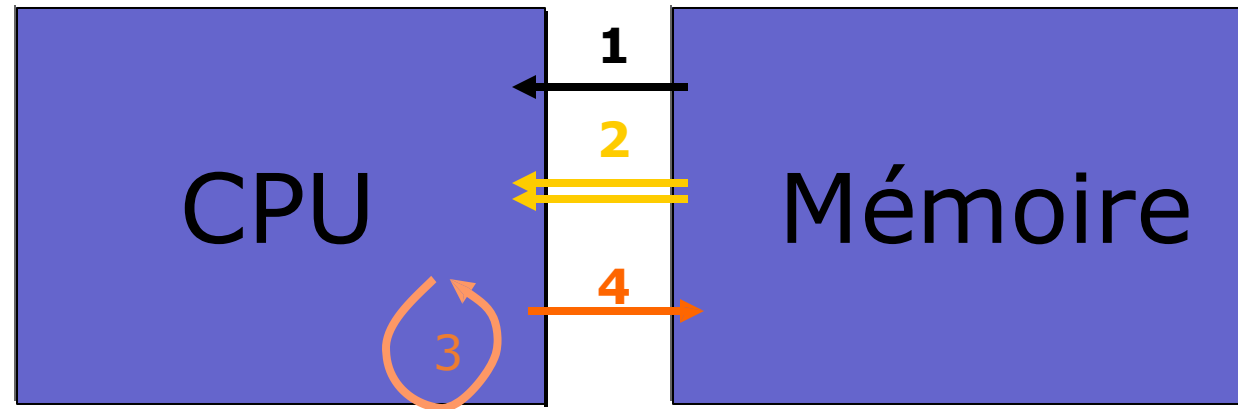
Le chipset est un composant essentiel intégré à la carte mère d'un micro-ordinateur. Il joue un rôle crucial dans la fluidité des communications au sein du système.

- ✓ Le pont nord gère la communication entre le processeur, la mémoire et la carte graphique.
- ✓ Le pont sud gère la communication avec d'autres périphériques tels que les ports USB, les ports SATA et les ports Ethernet



❑ Le microprocesseur

Le fonctionnement basique d'une opération



(1)Charger une instruction depuis la mémoire

(2)Charger les opérandes depuis la mémoire

(3)Effectuer les calculs

(4)Stocker le résultat en mémoire

❑ Le microprocesseur

Le traitement des instructions

Organisation d'une instruction

Le microprocesseur ne comprend qu'un certain nombre d'instructions qui sont codées en binaire.

Une instruction est composée de deux éléments :

- **Le code opération** : C'est un code binaire qui correspond à l'action à effectuer par le processeur
- **Le champ opérande** : Donnée ou bien adresse de la donnée.

La taille d'une instruction peut varier, elle est généralement de quelques octets (1 à 8), elle également de l'architecture du processeur.

❑ Le microprocesseur

Le traitement des instructions

Exemple d'instruction

- Instruction d'addition :

Accumulateur = Accumulateur + Opérande

Correspondant à l'instruction ADD A,#2

Instruction (16 bits)	
Code opératoire (5 bits)	Champ opérande (11 bits)
ADD A	#2
11001	000 0000 0010

Cette instruction est comprise par le processeur par le mot binaire :

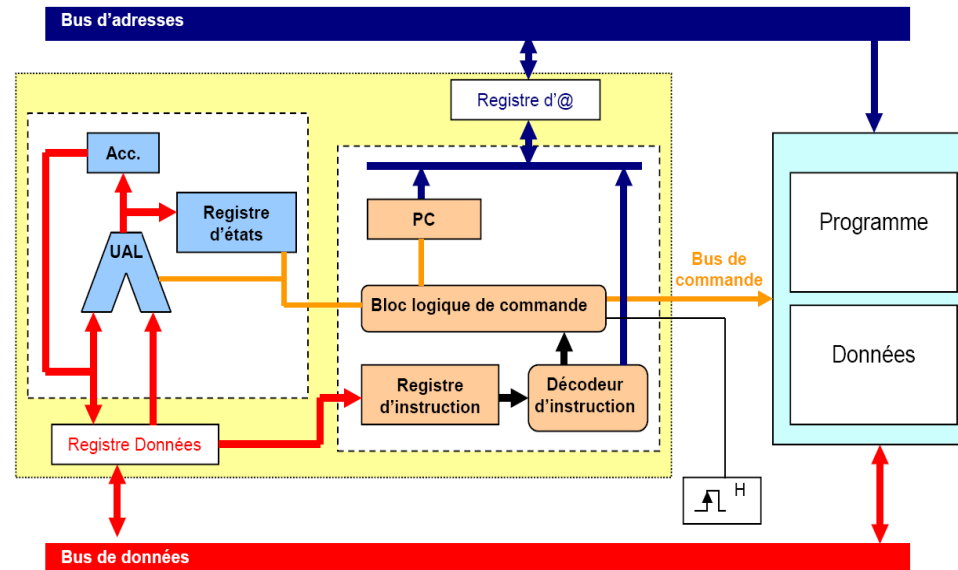
11001 000 0000 0010 = code machine

❑ Le microprocesseur

Le traitement des instructions

Phase 1 : Recherche de l'instruction en mémoire

- La valeur du PC est placée sur le bus d'adresse par l'unité de commande qui émet un ordre de lecture.
- Après le temps d'accès à la mémoire, le contenu de la case mémoire sélectionnée est disponible sur le bus des données.
- L'instruction est stockée dans le registre d'instruction du processeur

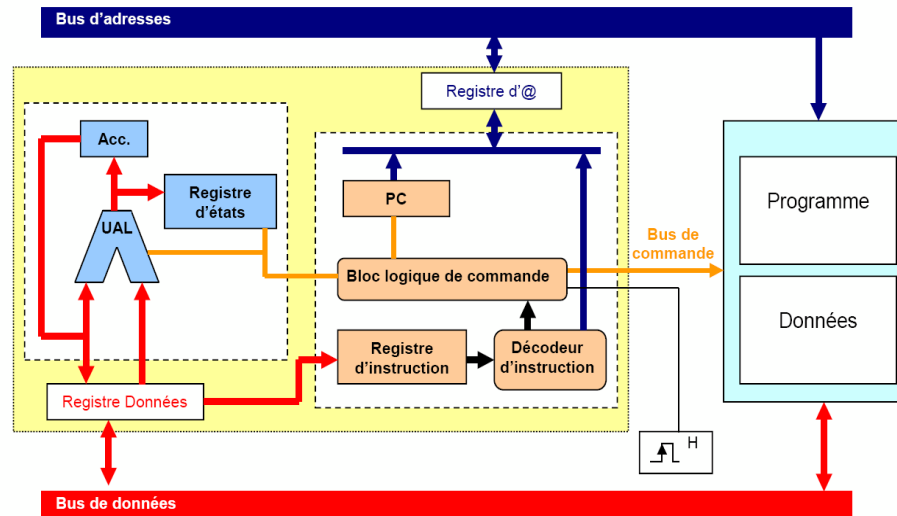


❑ Le microprocesseur

Le traitement des instructions

Phase 2 : Décodage et recherche de l'opérande

- L'unité de commande transforme l'instruction en une suite de commandes élémentaires nécessaires au traitement de l'instruction.
- Si l'instruction nécessite une donnée en provenance de la mémoire, l'unité de commande récupère sa valeur sur le bus de données.
- L'opérande est stocké dans le registre de données

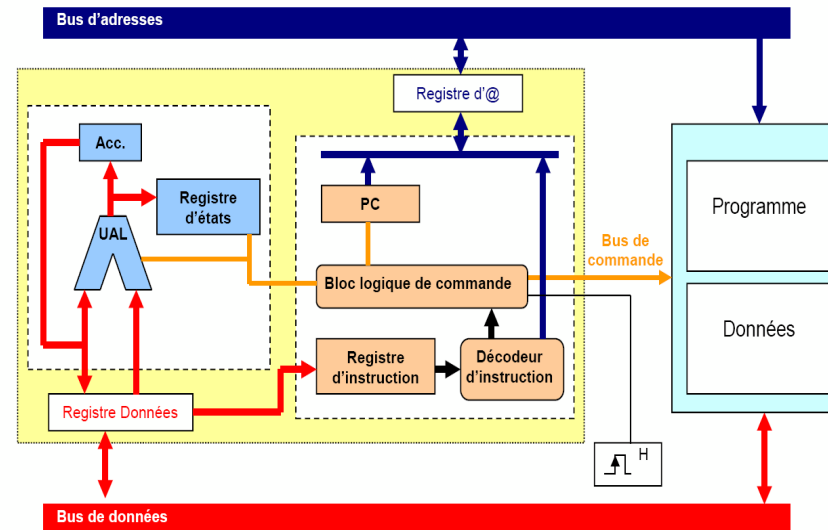


❑ Le microprocesseur

Le traitement des instructions

Phase 3 : Exécution de l'instruction

- Le séquenceur réalise l'instruction.
- Les drapeaux sont positionnés (registre d'état).
- L'unité de commande positionne le PC sur l'instruction suivante.



❑ Le microprocesseur

Le traitement des instructions

Les architectures RISC et CISC

Actuellement l'architecture des microprocesseurs composent de deux grandes familles :

- L'architecture CISC (Complex Instruction Set Computer)
- L'architecture RISC (Reduced Instruction Set Computer)

Architecture RISC

- Instructions simples ne prenant qu'un seul cycle
- Instructions au format fixe
- Décodeur simple (câblé)
- Beaucoup de registres
- Peu de modes d'adressage
- Compilateur complexe

Architecture CISC

- Instructions complexes prenant plusieurs cycles
- Instructions au format variable
- Décodeur complexe (microcode)
- Peu de registres
- Beaucoup de modes d'adressage
- Compilateur simple

❑ Le microprocesseur

Les modes d'adressages

Ce sont les diverses manières de définir la localisation d'une opérande. Les trois modes d'adressage les plus courants sont :

- Adressage immédiat
- Adressage direct
- Adressage indirect

❑ Le microprocesseur

Les modes d'adressages

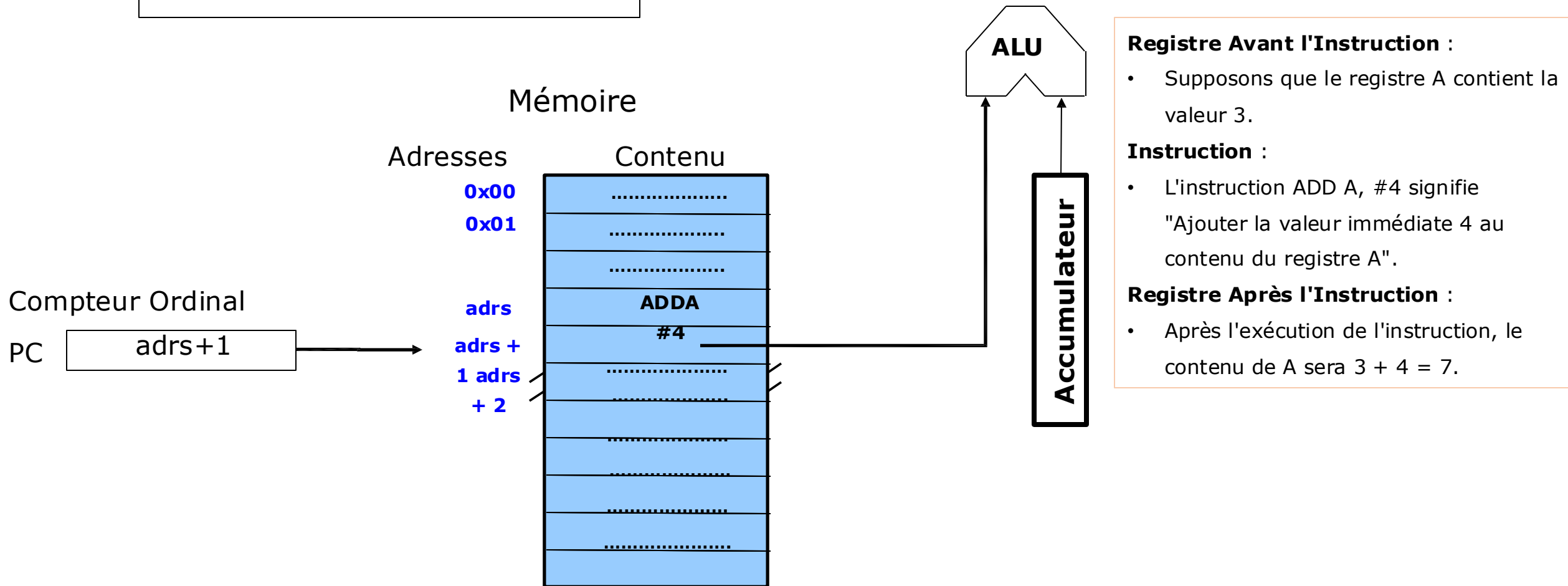
Immédiat

Exemple : ADD A #4

ADD : Opcode de l'instruction qui indique une addition.

A : Le registre où le résultat de l'addition sera stocké.

#4 : La valeur immédiate qui sera ajoutée au contenu de A



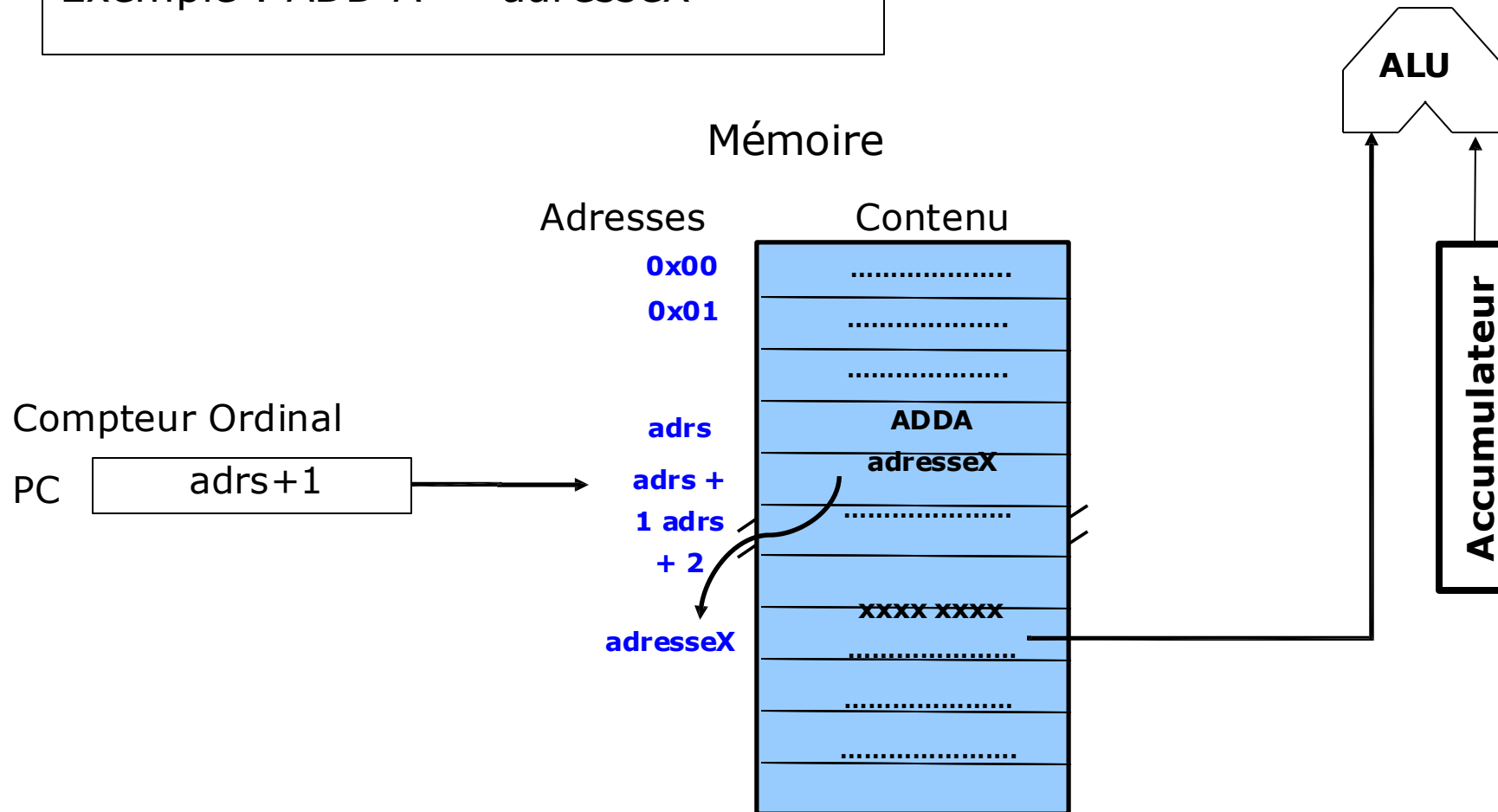
❑ Le microprocesseur

Les modes d'adressages

Direct

Exemple : ADD A adresseX

Ici, adresseX est une adresse mémoire. L'instruction ajoute la valeur stockée à l'adresse mémoire adresseX au contenu de A.



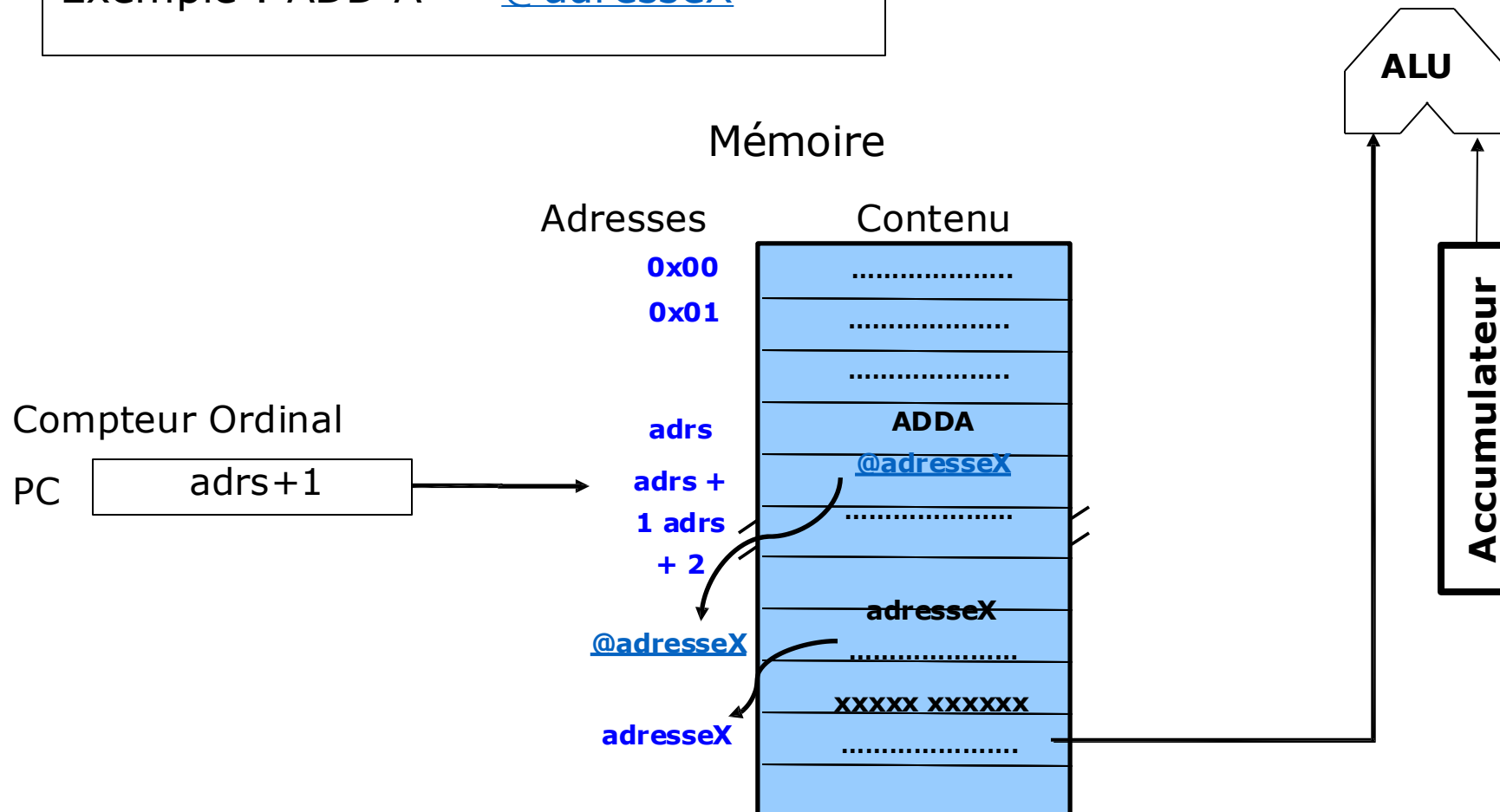
❑ Le microprocesseur

Les modes d'adressages

Indirect

Exemple : ADD A @adresseX

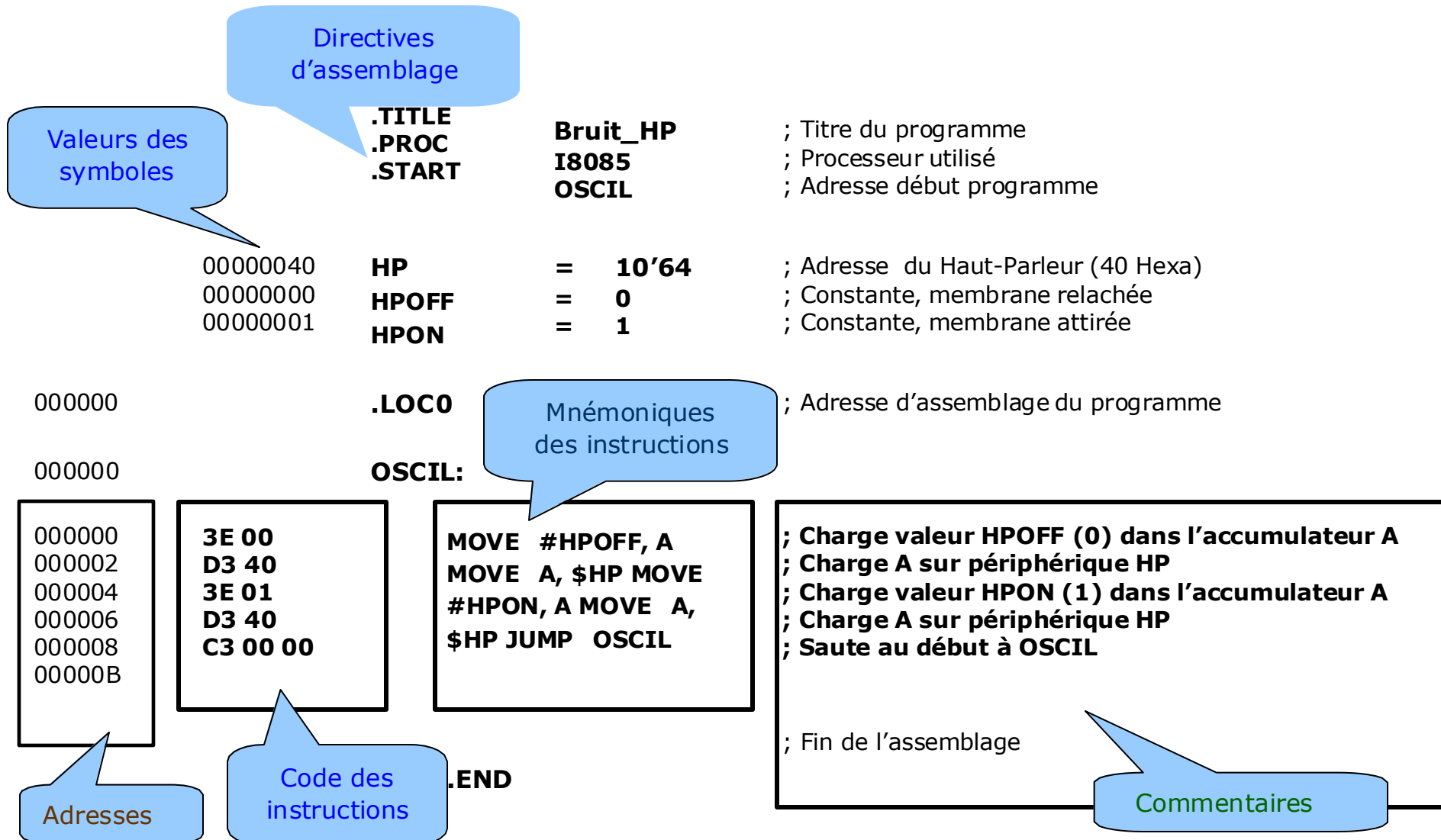
Ici, l'adresse mémoire où se trouve l'opérande est contenue dans le registre B. L'instruction ajoute la valeur située à l'adresse pointée par B au contenu de A.



❑ Le microprocesseur

Les modes d'adressages

Exemple d'exécution



❑ Le microprocesseur

Les modes d'adressages

Exemple d'exécution

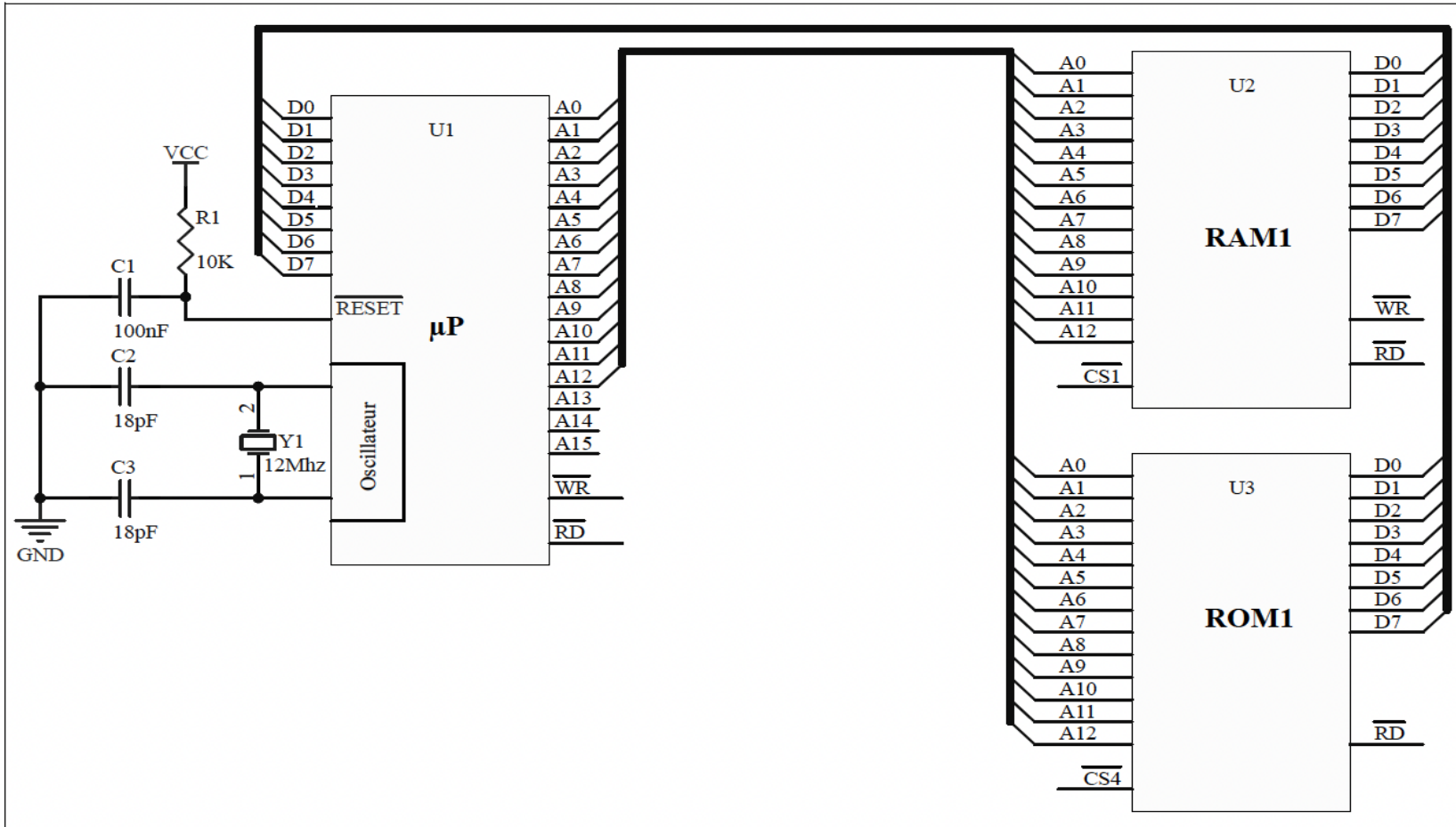
Programme:
instructions

Vue symbolique

Adresses

.00	3E	MOVE	,A	MOVE #HPOFF,A
.01	00		#HPOFF	
.02	D3	MOVE	A,	MOVE A,\$HP
.03	40		\$HP	
.04	3E	MOVE	,A	MOVE #HPON,A
.05	01		#HPON	
.06	D3	MOVE	A,	MOVE A,\$HP
.07	40		\$HP	
.08	C3	JUMP		JUMP OSCIL
.09	00			
0A	00	OSCIL		

Exercice d'application



1. Quelle est l'architecture de ce microprocesseur ?
2. Donner la capacité de chaque mémoire en kbits et en koctets.
3. On veut adresser la mémoire RAM1 à partir de l'adresse 0000H et la ROM1 à partir de l'adresse E000H.
 - a. Compléter le tableau suivant, en indiquant dans la troisième colonne l'adresse la plus basse et l'adresse la plus haute de chaque mémoire en hexadécimal.

		Adr. en Hexa	A15 A14 A13 A12	A11 A10 A9 A8	A7 A6 A5 A4	A3 A2 A1 A0
RAM1	Adr. basse					
	Adr. haute					
ROM1	Adr. basse					
	Adr. haute					

Exercice d'application

b. Donner les équations logiques des lignes $\overline{CS1}$ et $\overline{CS4}$ en fonction des lignes d'adresses A15, A14 et A13.

c. Compéter alors le schéma de décodage d'adresse en utilisant des opérateurs logiques.

On veut étendre la capacité mémoire de cette carte en ajoutant deux mémoires RAM2 et ROM2 identiques aux précédentes.

a. Donner l'adresse de base des nouvelles mémoires, si l'on veut que deux mémoires de même type soient adjacentes.

b. Donner les équations logiques des lignes de sélection $\overline{CS2}$ et $\overline{CS3}$ (correspondant respectivement aux mémoires RAM2 et ROM2) en fonction des lignes d'adresses A15, A14 et A13.

c. Donner le schéma de décodage d'adresses de toutes les mémoires en utilisant le circuit intégrées 74138.

❑ Le microprocesseur

Exercice d'application

A	74138	$\overline{Y0}$	$\overline{G2A}$	$\overline{G2B}$	G1	C	B	A	$\overline{Y0}$	$\overline{Y1}$	$\overline{Y2}$	$\overline{Y3}$	$\overline{Y4}$	$\overline{Y5}$	$\overline{Y6}$	$\overline{Y7}$
B		$\overline{Y1}$	1	X	X	X	X	X	1	1	1	1	1	1	1	1
C		$\overline{Y2}$	X	1	X	X	X	X	1	1	1	1	1	1	1	1
		$\overline{Y3}$	X	X	0	X	X	X	1	1	1	1	1	1	1	1
		$\overline{Y4}$	0	0	1	0	0	0	0	1	1	1	1	1	1	1
		$\overline{Y5}$	0	0	1	0	0	1	1	0	1	1	1	1	1	1
G1		$\overline{Y6}$	0	0	1	0	1	0	1	1	0	1	1	1	1	1
$\overline{G2A}$		$\overline{Y7}$	0	0	1	0	1	1	1	1	1	0	1	1	1	1
$\overline{G2B}$			0	0	1	1	0	0	1	1	1	1	0	1	1	1
				0	0	1	1	1	0	1	1	1	1	1	0	1
			0	0	1	1	1	1	1	1	1	1	1	1	0	

5. On suppose que le programme utilisateur commence au début de la mémoire programme (ROM), et que le vecteur RESET est situé à l'adresse FFFE_H.

Donner le contenu des cases mémoires d'adresses FFFE_H et FFFF_H (ce microprocesseur a une architecture little-endian, c.-à-d. que l'octet le plus faible est placé à l'adresse basse et l'octet le plus fort est placé à l'adresse haute).