

Architectures matérielles — Découvrir Von Neumann

1 Architecture Von Neumann

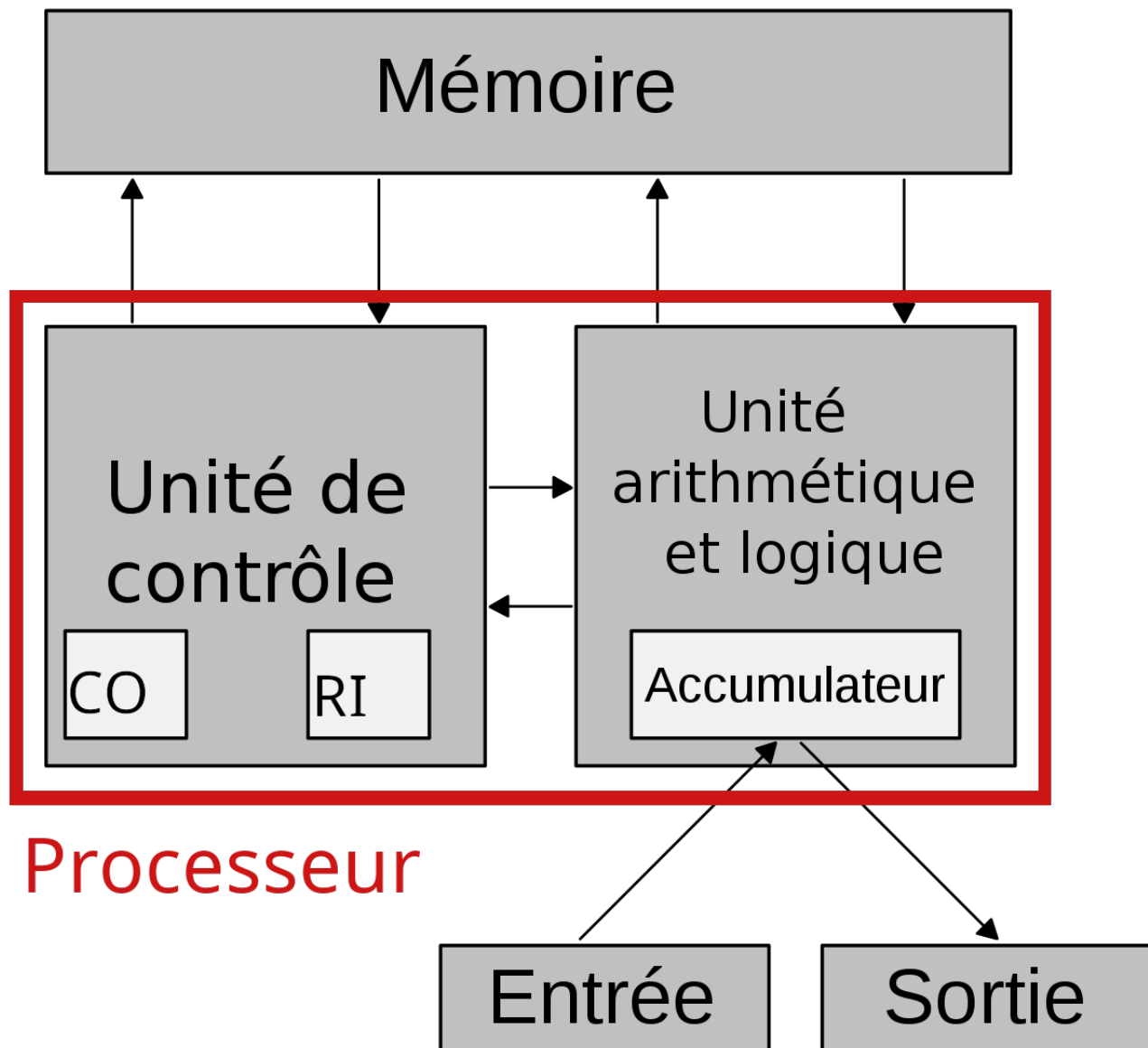


FIGURE 1 – Architecture Von Neumann. Les registres sont sur fond clair

2 La mémoire vive (ou RAM)

Il s'agit d'un tableau de bits (64 dans les ordinateurs récents).

Adresse (ici sur 4 bits)	valeur (ici sur 8 bits)
0000	0110 1010
0001	1111 1111
0010	1111 0000
...	...

3 Les Registres

Le processeur ne traite pas directement les données en RAM, il utilise pour cela des registres (de la mémoire directement incluse dans le processeur).

Parmi ces registres, nous allons utiliser les trois suivants :

RI le registre d'instruction; il contient l'instruction en cours;

CO le compteur ordinal; il contient l'adresse de la prochaine instruction à exécuter;

AC l'accumulateur; registre de l'UAL chargé de stocker une des opérands avant l'exécution puis le résultat de l'opération.

4 Les instructions

Un processeur est capable d'exécuter des instructions.

DÉFINITION 1

Le **langage machine** est une succession de bits qui est interprétable par le processeur d'un ordinateur.

Un **langage assembleur** est le langage machine machine ou des combinaisons de bits sont représentées par des "symboles" qu'un être humain peut mémoriser.

Un langage assembleur est souvent spécifique un type de processeur.

Instruction machine	Instruction Assembleur	Effet
0000 xxxx	LD X	Lit le contenu de la cellule X ($ACC \leftarrow \#X$)
0001 xxxx	STO X	Enregistre dans la cellule X ($\#X \leftarrow ACC$)
0010 xxxx	ADD X	Ajoute le contenu de la cellule X ($ACC \leftarrow ACC + \#X$)
0011 xxxx	SUB X	Soustrait le contenu de la cellule X ($ACC \leftarrow ACC - \#X$)
0100 aaaa	JMP ADR	Saute à l'adresse ADR ($CO \leftarrow A$)
0101 aaaa	JMPZ ADR	Saute à l'adresse ADR si nul
0110 aaaa	JMPP ADR	Saute à l'adresse ADR si positif
0111 aaaa	JMPN ADR	Saute à l'adresse ADR si négatif

FIGURE 2 – Un exemple de langage machine, dont une opération est numérotée sur 4 bits et un paramètre sur 4 bits également.

Voici un exemple d'état de la mémoire et des registres. Que va faire le processeur ainsi démarré?

Adresse	Contenu	Étape	RI	CO	Acc	RAM1	RAM2
0	25	Initiale	Ø	8	Ø	25	14
1	14	1					
...	...	2					
8	LD 0	3					
9	SUB 1	4					
10	STO 2						
11	END						

4.1 Exercice

Voici un exemple d'état de la mémoire et des registres. Complétez le tableau de droite pour simuler les actions du processeur et essayez de décrire ce que fait ce programme.

Adresse	Contenu	Étape	RI	CO	Acc	RAM1	RAM2
0	3	Initiale		8			
1	20	1					
...	...	2					
8	LD 0	3					
9	ADD 10	4					
10	STO 2						
11	END						

4.2 Exercice

Voici un exemple d'état de la mémoire et des registres. Complétez le tableau de droite pour simuler les actions du processeur et essayez de décrire ce que fait ce programme.

Adresse	Contenu	Étape	RI	CO	Acc	RAM1	RAM2
0	3	Initiale		8			
1	20	1					
...	...	2					
8	LD 0	3					
9	ADD 10	4					
10	JMP 8	5					
11	END	6					
		7					
		8					

4.3 Exercice

Bonus. Voici un exemple d'état de la mémoire et des registres. Complétez le tableau de droite pour simuler les actions du processeur et essayez de décrire ce que fait ce programme.

Adresse	Contenu	Étape	RI	CO	Acc	RAM1	RAM2
0	3	Initiale		8			
1	20	1					
...	...	2					
8	LD 0	3					
9	SUB 1	4					
10	JUMPZ 13	5					
11	JUMP 9	6					
12	JMP 8	7					
13	STO 0	8					
14	END						

4.4 Exercice

Bonus. Écrire un programme en langage assembleur qui récupère la valeur en RAM à l'adresse 0, à ajoute 1 et stock le résultat à l'adresse 0.

4.5 Exercice

Bonus. Écrire un programme en langage assembleur qui récupère la valeur en RAM à l'adresse 0, à ajoute 1 si elle est négative (et retranche 1 si elle est positive) et stock le résultat à l'adresse 0.

4.6 Exercice

Bonus. Écrire un programme en langage assembleur qui récupère la valeur en RAM à l'adresse 0, à ajoute 1 si elle est négative (et retranche 1 si elle est positive) et stock le résultat à l'adresse 0, puis recommence jusqu'à ce que la valeur soit nulle.