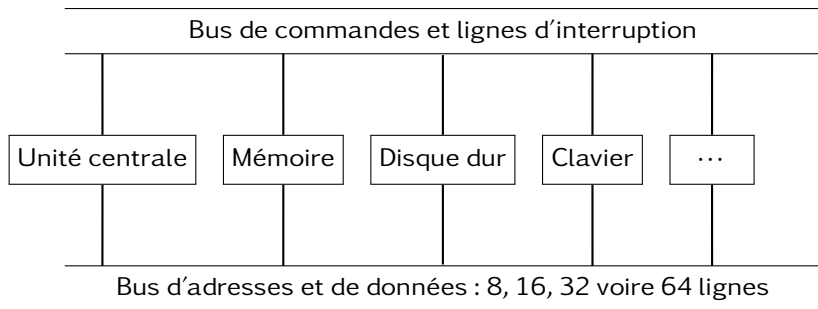


L'architecture matérielle d'un micro-ordinateur



Les huit principes de base d'un ordinateur - I

- ▶ Les données sont codées en fonction de leur type : le naturel 10 est codé comme le nombre 10 en binaire ; le relatif -2 est codé en binaire « complément à 2 » ; le nombre réel 2,1 est codé en deux composantes, la mantisse et l'exposant ; le caractère A est codé, en utilisant la convention ASCII, comme le nombre 65, etc.
- ▶ L'octet (une suite de 8 bits) est la plus petite quantité d'information adressable et déplaçable ; la taille des données est exprimée en octets ; par exemple, un caractère est codé sur un octet, un nombre entier sur 4 ou 8 octets, etc.
- ▶ Une opération et un opérande définissent une instruction ; le dernier opérande est le registre de l'unité centrale ; par exemple, ADDI 193 veut dire « additionner, d'une part, le nombre entier situé à l'adresse 193 dans la mémoire et, d'autre part, le nombre entier contenu dans le registre de l'unité centrale ».

Les huit principes de base d'un ordinateur - II

- ▶ Le jeu d'instructions varie d'un modèle de micro-processeur à l'autre.
- ▶ Les instructions à exécuter sont des données ; elles doivent être présentes dans la mémoire avant d'être exécutées.
- ▶ Les instructions sont exécutées *a priori* en séquence mais certaines instructions sont des instructions de branchement à des adresses particulières dans le code ; l'exécution d'un programme peut être interrompue par un évènement interne ou externe.
- ▶ Le nombre de lignes du bus fixe la taille des adresses et la taille maximale de la mémoire ; un bus de largeur 8 (on parle d'un micro-ordinateur 8 bits) peut adresser $2^8 = 256$ octets différents ; un bus de largeur 64 peut adresser 2^{64} octets soit $\approx 18\,446\,744\,074$ megaoctets.
- ▶ Les données ne sont pas persistantes dans la mémoire ; elles doivent être enregistrées sur un support pour être conservées.

Quand j'appuie sur une touche du clavier...

1. le clavier va interrompre l'unité centrale, grâce à la ligne d'interruption, pour la prévenir de cet évènement ;
2. l'unité centrale va demander à la mémoire, grâce au bus de commandes, de se préparer
 - 2.1 à lire sur le bus l'adresse à laquelle le code de la touche doit être enregistré ;
 - 2.2 puis à lire sur le bus le code de la touche ;
3. l'unité centrale va écrire sur le bus l'adresse en mémoire du code de la touche ;
4. la mémoire va lire sur le bus cette adresse ;
5. l'unité centrale va demander au clavier d'écrire sur le bus le code de la touche ;
6. le clavier va écrire sur le bus le code de la touche ;
7. la mémoire va lire sur le bus le code de la touche et l'enregistrer à la bonne adresse ;
8. la mémoire va interrompre l'unité centrale pour lui dire que l'opération d'entrée/sortie est achevée.

Les quatre segments de la mémoire attribuée à un programme

text	data	heap	stack
les instructions	les données « statiques »	les données « à façon »	la pile

Ma première compilation d'un programme en C++

```
1 #include <iostream>
2
3 int x, y ;    // Variables entières statiques.
4
5 int main() { // Point d'entrée dans le programme.
6     x = 1 ;
7     y = x + 10 ;
8     return 0 ; // Code de retour égal à 0.
9 }
```

Traduction dans un code exécutable fictif

Segment	Adresse	Code	Explication
text	0	10 100	Registre \leftarrow 1
	2	11 101	Mémoire(101) \leftarrow Registre
	4	10 101	Registre \leftarrow Mémoire(101)
	6	60 102	Registre \leftarrow Registre + Mémoire(102)
	8	11 103	Mémoire(103) \leftarrow Registre
	10	00 104	Retour avec le code 0
data	100	01	Constante 1
	101	—	Emplacement réservé pour la variable x
	102	10	Constante 10
	103	—	Emplacement réservé pour la variable y
	104	00	Constante 0

... en supposant que...

1. l'instruction « charger le registre » est codée 10 ;
2. l'instruction « enregistrer le registre » est codée 11 ;
3. l'instruction « additionner des entiers » est codée 60 ;
4. l'instruction « retourner au système d'exploitation » est codée 0 ;
5. le segment data commence à l'adresse 100.