

Fonctionnement des ordinateurs

chapitre VI : mémoires

Prof. Xavier Gandibleux

Université de Nantes
Département Informatique – UFR Sciences et Techniques

Année académique 2018-2019

Fonctionnement des ordinateurs



Xavier.Gandibleux@Univ-Nantes.fr 0

Mémoires

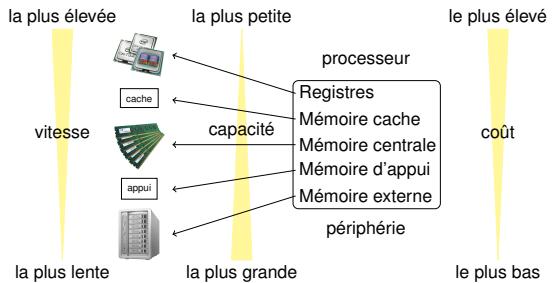
vue d'ensemble des mémoires
examen microscopique de la mémoire centrale
nature du contenu en mémoire centrale
opérations de lecture et écriture en mémoire centrale
caractéristiques d'une mémoire centrale
occupations de l'espace en mémoire centrale
regard sur les hiérarchies de mémoires

Mémoires

Vue d'ensemble

Mémoires

Vue d'ensemble



Fonctionnement des ordinateurs



Xavier.Gandibleux@Univ-Nantes.fr 2

Fonctionnement des ordinateurs



Xavier.Gandibleux@Univ-Nantes.fr 3

Mémoires

Vitesses caractéristiques :

- ▶ bascules, registres dans le processeur (< 1 ns)
- ▶ mémoire statique (\approx 10 ns)
- ▶ mémoire dynamique (\approx 100 ns)
- ▶ mémoire flash (\approx 10 μ s/50 ns)
- ▶ disque SSD (\approx 100 μ s)
- ▶ disque magnétique (\approx 10 ms)
- ▶ disque optique (\approx 150 ms)

Mémoires

Unités de capacité mémoire :

Système international des unités (SI) :

1 bit (b)

8 bits = 2^3 bits = 1 octet (o) = 1 byte (B)

1ko = 1 000 octets = 1kB = 10^3 bytes

1Mo = 1 000 000 octets = 1MB = 10^6 bytes

1Go = 1 000 000 000 octets = 1GB = 10^9 bytes

1998, des préfixes pour les multiples en base 2 (IEC) :

ki (kibi) : 1ki = $(2^{10})^1$ bits = 1024 bits

Mi (mebi) : 1Mi = $(2^{10})^2$ bits

Gi (gibi) : 1Gi = $(2^{10})^3$ bits

Exemples :

1kibit = 2^{10} bits = 1024 bits

1kbit = 10^3 bits = 1000 bits

1MIB = 2^{20} B = 1 048 576 B = 8 388 608 bits

1MB = 10^6 B = 1 000 000 B = 8 000 000 bits

Fonctionnement des ordinateurs



Xavier.Gandibleux@Univ-Nantes.fr 4

Fonctionnement des ordinateurs



Xavier.Gandibleux@Univ-Nantes.fr 5

Mémoires

Usages, types, capacités :

Mémoire centrale

- mémoire située près du processeur
- RAM, mémoire flash
- de quelques Mo à quelques Go

Mémoire de masse

- stockage de longue durée
- disque durs magnétiques, SSD
- plusieurs centaines de Go à quelques To

Archivage

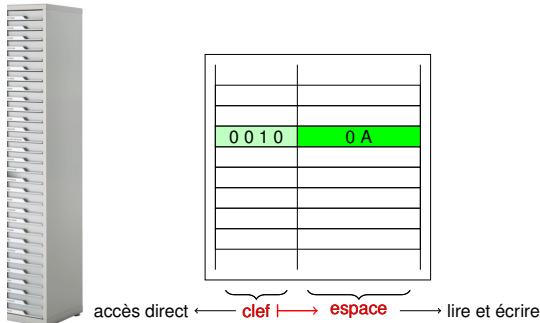
- stockage de très longue durée
- bandes magnétiques
- plusieurs centaines de Go à quelques To
- lent, accès séquentiel, faible coût

Mémoires

Mémoire centrale : examen microscopique

Mémoires

Mémoire centrale

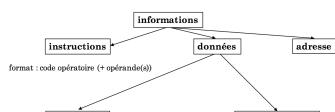


Mémoires

Mémoire centrale : nature du contenu en mémoire

Mémoires

Nature du contenu en mémoire



Nombre représentés

- en direct
entiers non-signés
- en complément à 2
entiers signés
- en virgule fixe
en virgule flottante
fractionnaires
- standard ASCII
American Standard Code for Information Interchange
- standard EBCDIC
Extended Binary Coded Decimal Interchange Code
- standard UNICODE
constitué d'un répertoire de 128 172 caractères
couvrant une centaine d'écritures.
Sous sa forme UTF-8 (UTF, Universal Transformation Format), l'Unicode offre une certaine interopérabilité
avec le code ASCII

Données codées par tables

Table ASCII

Decimal Hex Char	Decimal Hex Char	Decimal Hex Char	Decimal Hex Char
00	00	00	00
01	01	01	01
02	02	02	02
03	03	03	03
04	04	04	04
05	05	05	05
06	06	06	06
07	07	07	07
08	08	08	08
09	09	09	09
0A	0A	0A	0A
0B	0B	0B	0B
0C	0C	0C	0C
0D	0D	0D	0D
0E	0E	0E	0E
0F	0F	0F	0F
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
1A	1A	1A	1A
1B	1B	1B	1B
1C	1C	1C	1C
1D	1D	1D	1D
1E	1E	1E	1E
1F	1F	1F	1F
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
2A	2A	2A	2A
2B	2B	2B	2B
2C	2C	2C	2C
2D	2D	2D	2D
2E	2E	2E	2E
2F	2F	2F	2F
30	30	30	30
31	31	31	31
32	32	32	32
33	33	33	33
34	34	34	34
35	35	35	35
36	36	36	36
37	37	37	37
38	38	38	38
39	39	39	39
3A	3A	3A	3A
3B	3B	3B	3B
3C	3C	3C	3C
3D	3D	3D	3D
3E	3E	3E	3E
3F	3F	3F	3F
40	40	40	40
41	41	41	41
42	42	42	42
43	43	43	43
44	44	44	44
45	45	45	45
46	46	46	46
47	47	47	47
48	48	48	48
49	49	49	49
50	50	50	50
51	51	51	51
52	52	52	52
53	53	53	53
54	54	54	54
55	55	55	55
56	56	56	56
57	57	57	57
58	58	58	58
59	59	59	59
5A	5A	5A	5A
5B	5B	5B	5B
5C	5C	5C	5C
5D	5D	5D	5D
5E	5E	5E	5E
5F	5F	5F	5F
60	60	60	60
61	61	61	61
62	62	62	62
63	63	63	63
64	64	64	64
65	65	65	65
66	66	66	66
67	67	67	67
68	68	68	68
69	69	69	69
6A	6A	6A	6A
6B	6B	6B	6B
6C	6C	6C	6C
6D	6D	6D	6D
6E	6E	6E	6E
6F	6F	6F	6F
70	70	70	70
71	71	71	71
72	72	72	72
73	73	73	73
74	74	74	74
75	75	75	75
76	76	76	76
77	77	77	77
78	78	78	78
79	79	79	79
7A	7A	7A	7A
7B	7B	7B	7B
7C	7C	7C	7C
7D	7D	7D	7D
7E	7E	7E	7E
7F	7F	7F	7F

Table ASCII

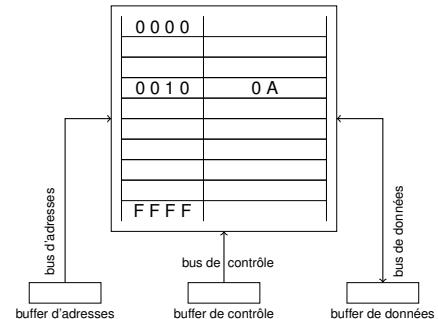
Mémoires

Mémoire centrale : Opérations de lecture et écriture

Mémoires

Mémoire centrale

Opérations de lecture et écriture :



Mémoires

Types de bus

- ▶ **bus d'adresses** : transporte les adresses mémoire auxquelles le processeur souhaite accéder (unidirectionnel).
- ▶ **bus de données** : véhicule les instructions en provenance ou à destination du processeur (bidirectionnel).
- ▶ **bus de contrôle** : transporte les ordres et les signaux de synchronisation de l'unité de commande (bidirectionnel).
- ▶ **bus d'extension** (bus d'entrée/sortie) : permet aux divers composants de la carte-mère (USB, cartes PCI, disques durs, lecteurs, ...) de communiquer entre eux.

Mémoires

Remarque 1 :

si (format des données en mémoire = 8 bits)
et (donnée à mémoriser > 8 bits) (Ex : adresse sur 16 bits : 2AF9)
alors
données sont décomposées en paquets (ici de 8 bits) comme suit :
- information de poids plus fort (pour l'exemple : 2A)
:
- information de poids plus faible (pour l'exemple : F9)
selon

convention "big endian"
(information de poids plus fort d'abord)

000F
0010
0011
0012

convention "small endian"
(information de poids plus faible d'abord)

000F
0010
0011
0012

Mémoires

Remarque 2 :

si (adresse mémoire = 16 bits) (Ex : 2AF9)
et (bus d'adresse = 8 bits)

alors

- 1) adresse mémoire décomposée en
 - adresse haute(@H) ; exemple : 2A
 - adresse basse(@B) ; exemple : F9

- 2) @H et @B circulent sur le bus d'adresse
selon la convention big ou small endian

nb : l'opération nécessite 2 fois plus de temps (2 cycles)
fsi

Mémoires

Mémoire centrale : caractéristiques d'une mémoire

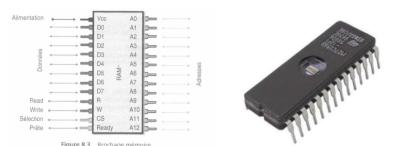


Figure 8.3 Brochage mémoire Brochage EPROM

Hiérarchies de mémoires

Le problème

- ▶ rapide = cher + faible capacité (SRAM)
- ▶ grande capacité = lent (DRAM, disque)

Solution

- ▶ combiner une petite mémoire rapide
- ▶ ... avec une grande mémoire lente
- ▶ et faire en sorte que les données voulues soient souvent dans la mémoire rapide.

Hiérarchies de mémoires

Principes

- ▶ **localité temporelle**
quand on accède à une donnée,
il y a des chances qu'on y accède de nouveau bientôt
 - ▶ boucles
 - ▶ usage répété de variables
- ▶ **localité spatiale**
quand on accède à une donnée,
il y a des chances qu'on accède ensuite à des données voisines
 - ▶ séquentialité du code, boucles
 - ▶ tableaux, codage contigu de l'information

Hiérarchies de mémoires

Application

- ▶ **localité temporelle**
 - ▶ on conserve dans une mémoire rapide les données que l'on consulte
 - ▶ quand la mémoire rapide est pleine, on élimine les données les plus anciennes
- ▶ **localité spatiale**
 - ▶ on charge dans la mémoire rapide les données voisines de celle que l'on consulte

Hiérarchies de mémoires

Cache

- ▶ petite mémoire statique, grande mémoire dynamique
-
- ```
graph LR; CPU[CPU] --> CacheL1[Cache L1]; CacheL1 --> CacheL2[Cache L2]; CacheL2 --> Mem[Memory]
```
- ▶ caches L1 séparés pour instructions et données

### Appui (cache de disque)

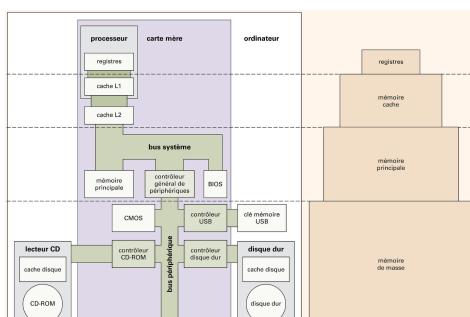
mémoire dynamique, disque dur : but = accélérer les accès au disque

### Mémoire virtuelle

mémoire dynamique, disque dur : but = augmenter la capacité mémoire

## Mémoires

### Mémoires et bus en résumé



## Suite...

processeurs