CSE
2105

Computer Architecture
Memory

Moniruzzaman

Adjunct Lecturer
North Western University, Khulna-9000

Confession

- Most of the materials have been collected from Internet.
- Images are taken from Internet.
- Various books are used to make these slides.
- Various slides are also used.
- References & credit:
  - Atanu Shome, Assistant Professor, CSE, KU.
  - Computer Organization and Design: the Hardware/Software Interface - Textbook by David A Patterson and John L. Hennessy.
  - Computer Organization and Architecture - Book by William Stallings

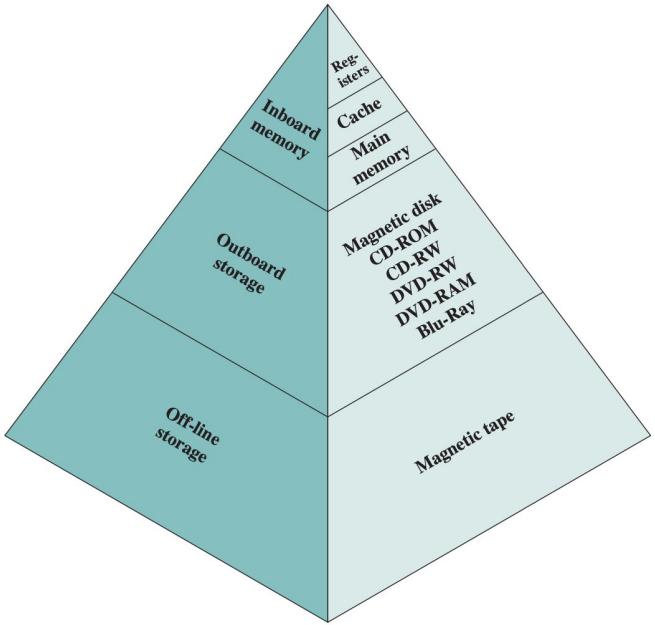
moniruzzaman

Memory

Memory is the electronic holding place for the instructions and data a computer needs to reach quickly.

- How much? How fast? How expensive?
- Faster access time, greater cost per bit
- Greater capacity, smaller cost per bit
- Greater capacity, slower access time

Memory Hierarchy



- Decreasing cost per bit
- Increasing capacity
- Increasing access time
- Decreasing frequency of access of the memory by the processor

Locality of Reference

- Also known as the principle of locality
- The tendency of a processor to access the same set of memory locations repetitively over a short period of time.

Locality of Reference - **Types**

Temporal locality refers to the reuse of specific data, and/or resources, within a relatively small time duration.

Spatial locality (also termed data locality) refers to the use of data elements within relatively close storage locations.

Cache Memory

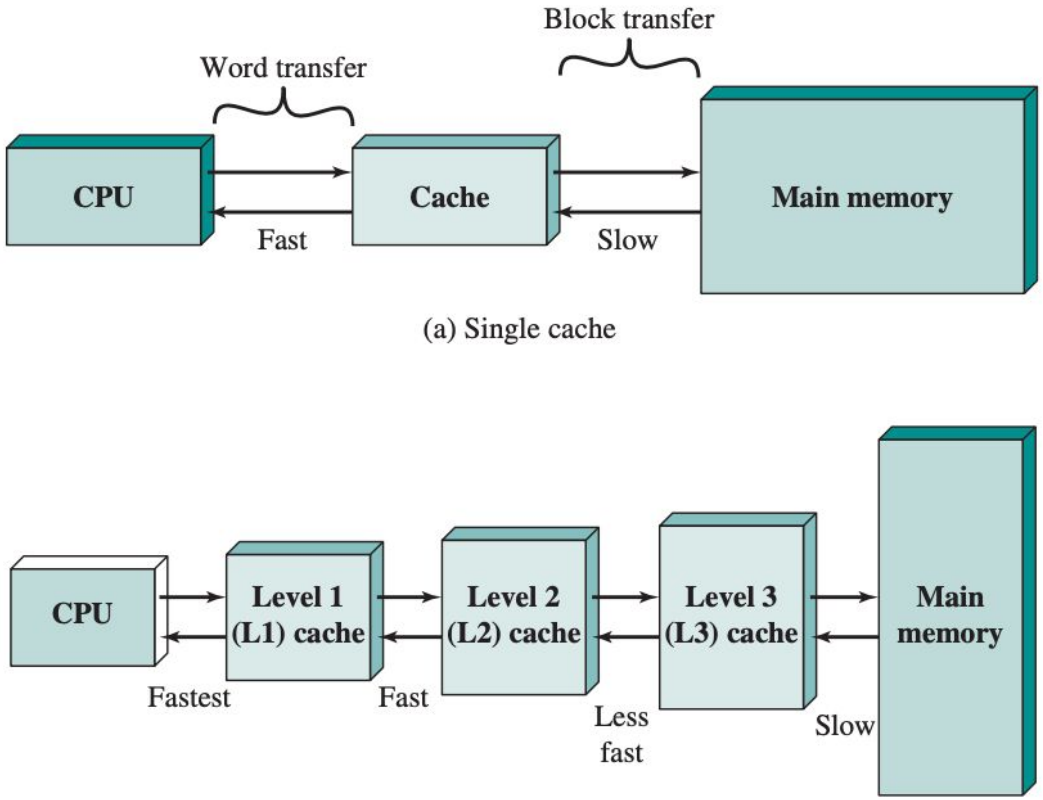
Cache memory

Cache memory is a small-sized type of volatile computer memory that provides

- high-speed data access to a processor and
- stores frequently used computer programs, applications and data.

It stores and retains data only until a computer is powered up.

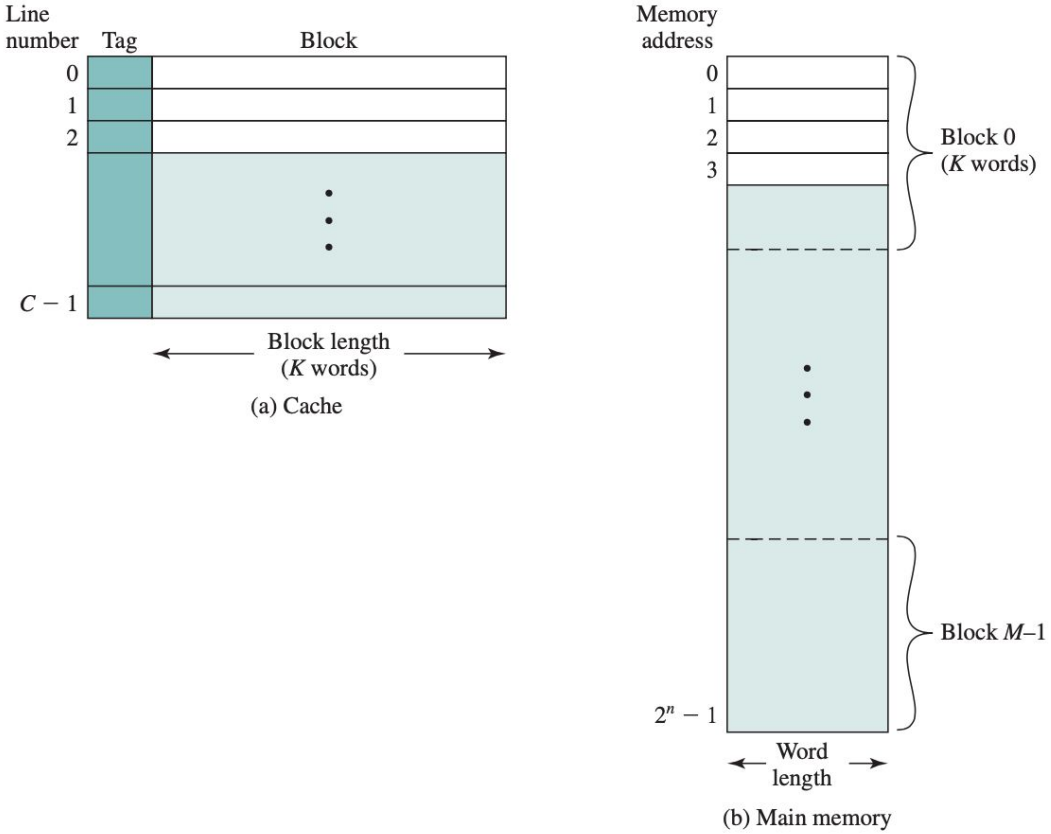
Cache and Main Memory Connection



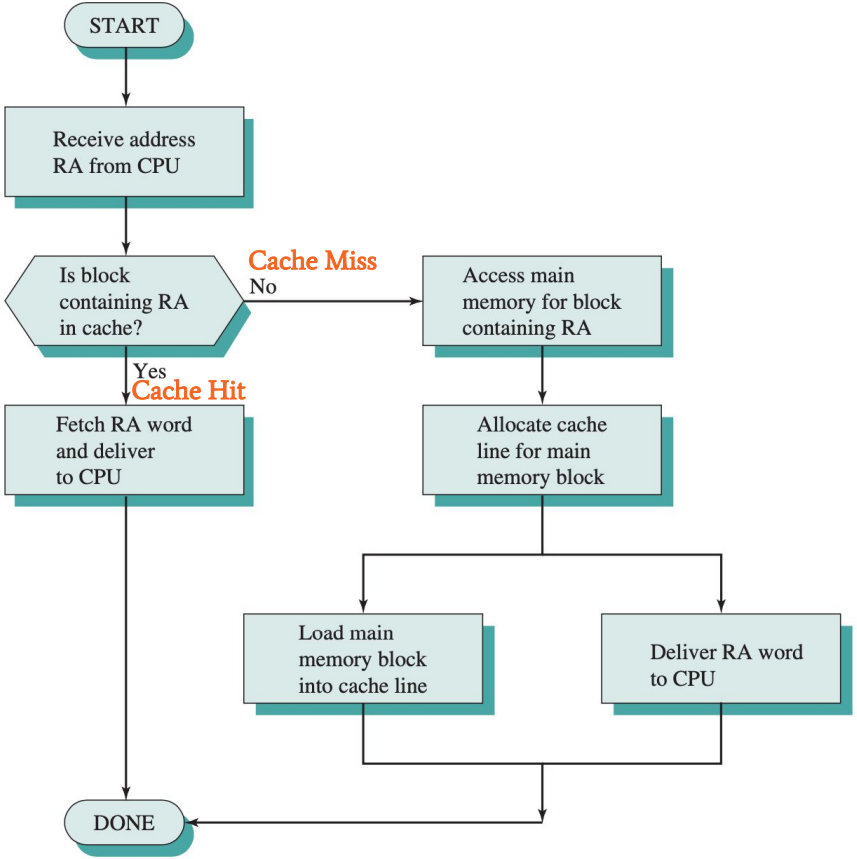
L1, L2, and L3 Cache

- **L1 cache**, or *primary cache*, is extremely fast but relatively small, and is usually embedded in the processor chip as CPU cache.
- **L2 cache**, or *secondary cache*, is often more capacious than L1. L2 cache may be embedded on the CPU, or it can be on a separate chip or coprocessor and have a high-speed alternative system bus connecting the cache and CPU.
- **Level 3 (L3) cache** is specialized memory developed to improve the performance of L1 and L2.
    - L1 or L2 can be significantly faster than L3, though L3 is usually double the speed of DRAM.
    - With multicore processors, each core can have dedicated L1 and L2 cache, but they can share an L3 cache.

Cache and Main Memory Structure



Cache Read Operation



Cache Mapping

- Direct Cache Mapping
- Fully Associative
- Set Associative

Direct Cache Mapping

Direct Cache Mapping

Address size ???

$2^6 = 64$ (6 bit)

Cache line = 4

Cache		
Line no.	Tag	Line
0		
1		
2		
3		

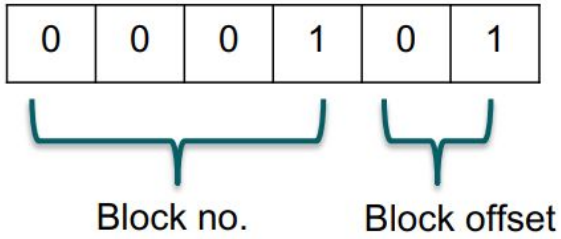
Block size = 4

RAM size = 64 words

	RAM
0	0, 1, 2, 3, ← Block
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping

CPU Genrated Address



Cache		
Line no.	Tag	Line
0		0 / 4 / 8 / 12
1		1 / 5 / 9 / 13
2		2 / 6 / 10 / 14
3		3 / 7 / 11 / 15

	RAM
0	0, 1, 2, 3,
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping

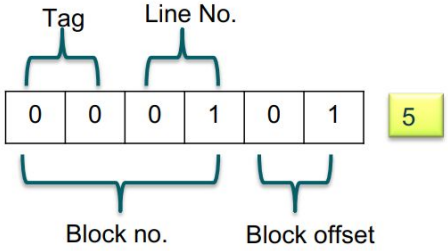
A	B	C	D	Decimal
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

Cache		
Line no.	Tag	Line
0		0 / 4 / 8 / 12
1		1 / 5 / 9 / 13
2		2 / 6 / 10 / 14
3		3 / 7 / 11 / 15

	RAM
0	0, 1, 2, 3,
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping

CPU Genrated Address



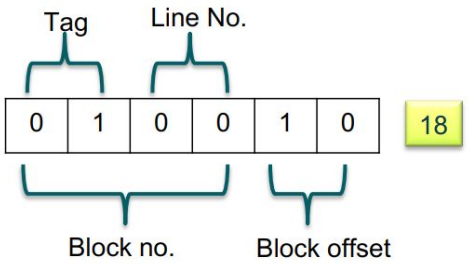
Cache Example		
Line no.	Tag	Line
0	01	16, 17, 18, 19
1	00	4, 5, 6, 7
2		
3		

Cache		
Line no.	Tag	Line
0		0 / 4 / 8 / 12
1		1 / 5 / 9 / 13
2		2 / 6 / 10 / 14
3		3 / 7 / 11 / 15

	RAM
0	0, 1, 2, 3, ←
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping

CPU Genrated Address



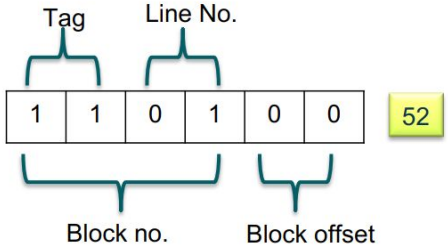
Cache Example		
Line no.	Tag	Line
0	01	16, 17, 18, 19
1	00	4, 5, 6, 7
2		
3		

Cache		
Line no.	Tag	Line
0		0 / 4 / 8 / 12
1		1 / 5 / 9 / 13
2		2 / 6 / 10 / 14
3		3 / 7 / 11 / 15

	RAM
0	0, 1, 2, 3,
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping

CPU Genrated Address

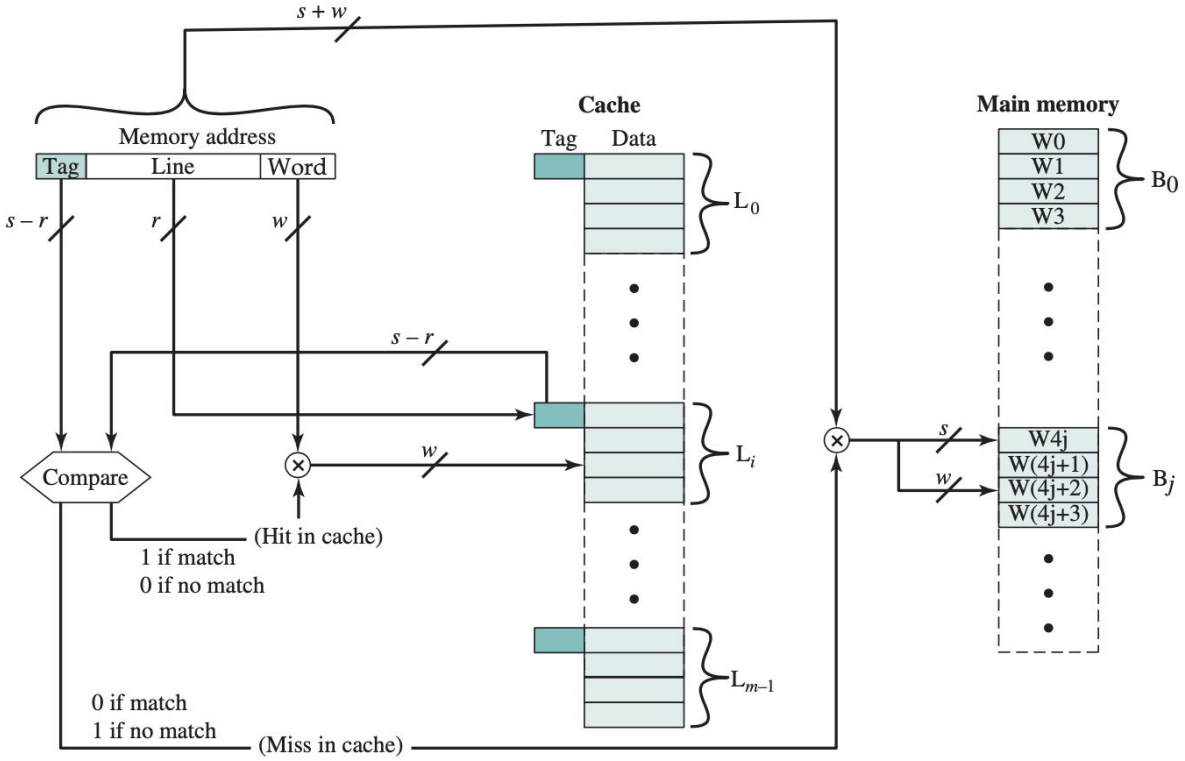


Cache Example		
Line no.	Tag	Line
0	01	16, 17, 18, 19
1	00	4, 5, 6, 7
2		
3		

Cache		
Line no.	Tag	Line
0		0 / 4 / 8 / 12
1		1 / 5 / 9 / 13
2		2 / 6 / 10 / 14
3		3 / 7 / 11 / 15

	RAM
0	0, 1, 2, 3,
1	4, 5, 6, 7,
2	8, 9, 10, 11,
3	12, 13, 14, 15,
4	16, 17, 18, 19,
5	20, 21, 22, 23,
6	24, 25, 26, 27,
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	60, 61, 62, 63

Direct Cache Mapping



Thank You