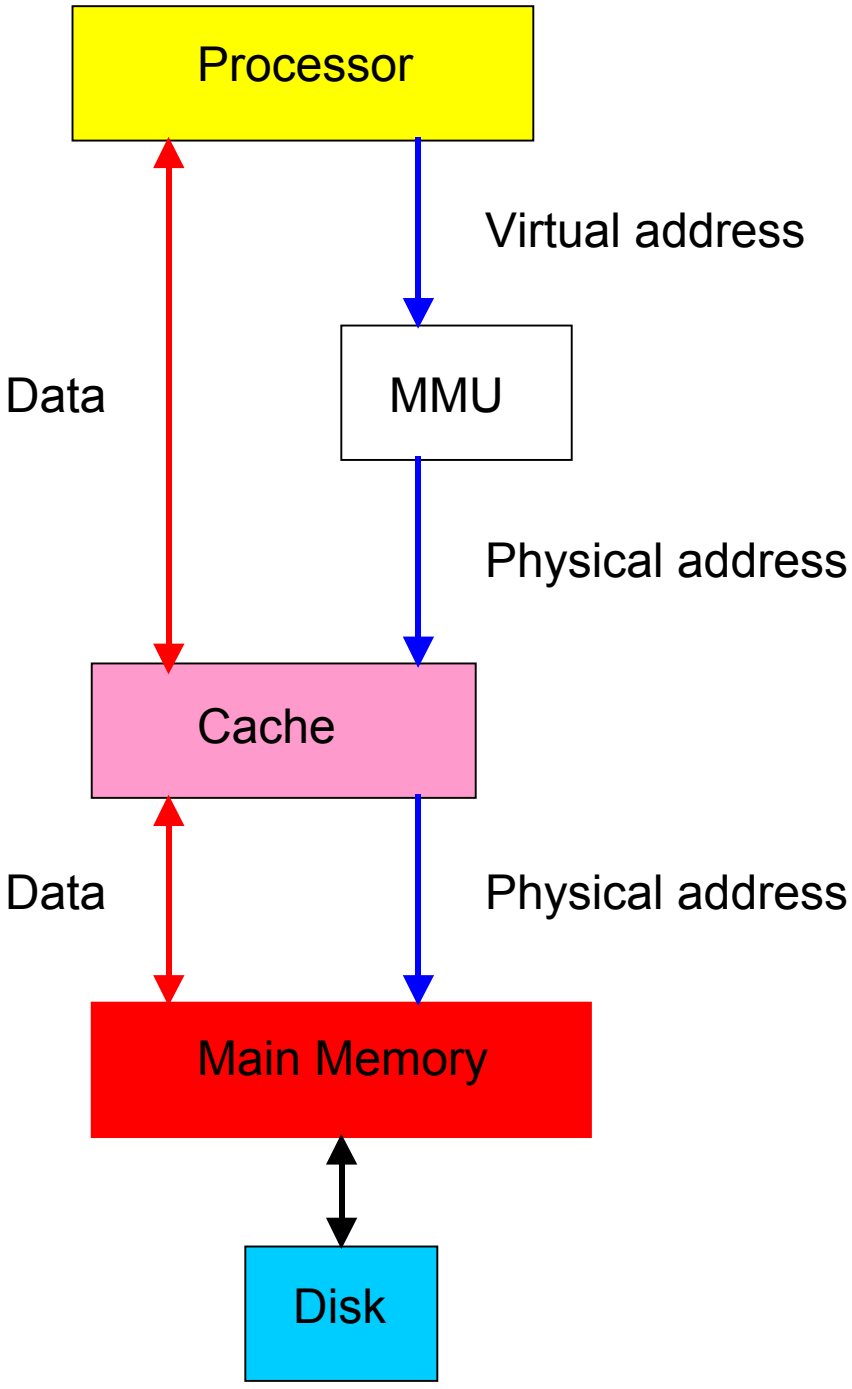
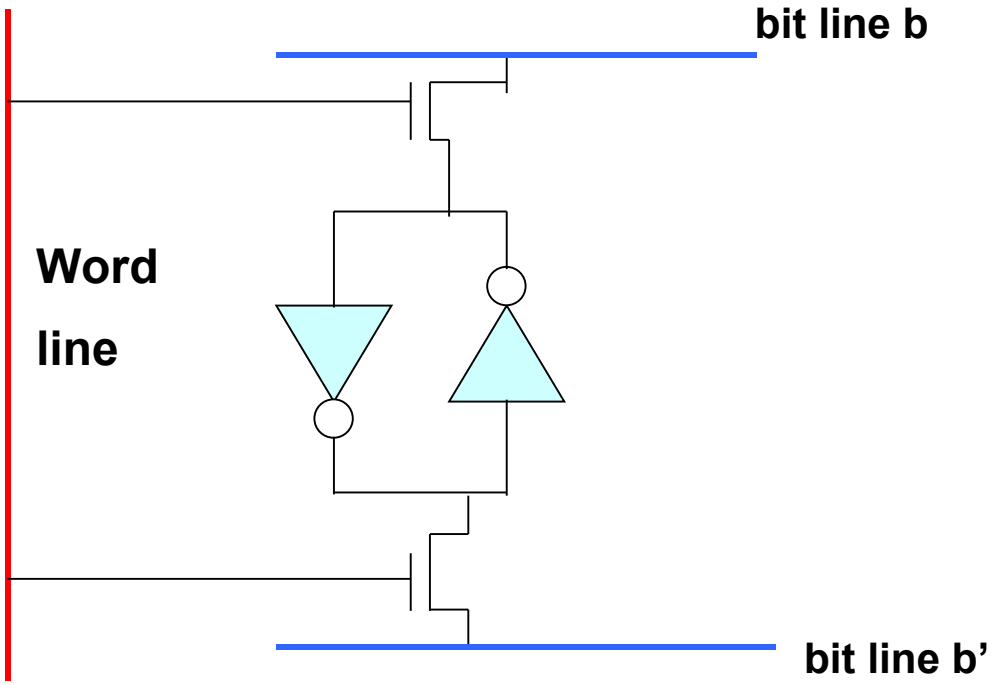
Memory Hierarchy



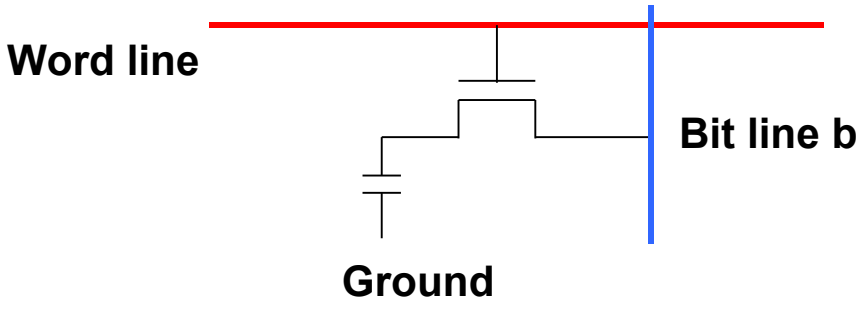
Main Memory

Basic SRAM or Static RAM



The cell switching is fast. No refreshing is necessary.

Dynamic Random Access Memory or DRAM

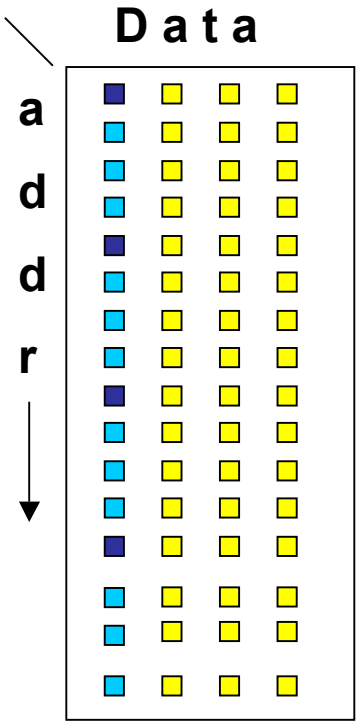


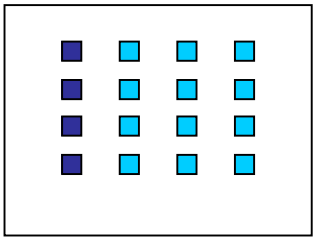
Made of an array of cells.

Each cell contains a transistor and a capacitor. Data (stored as electric charge) is lost unless the cells are periodically refreshed.

Much slower than SRAM, but high packing density. Widely used main memory.

Two-dimensional addressing

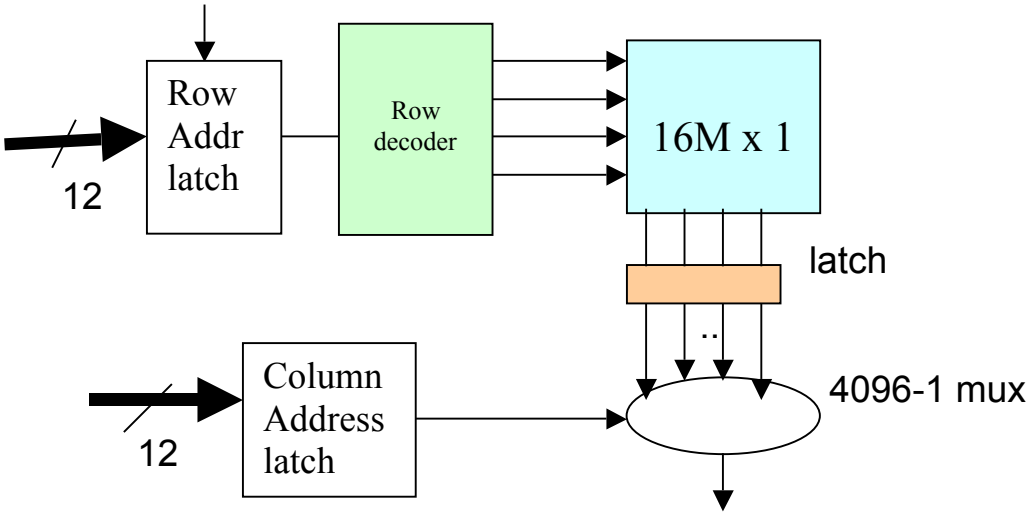




Arrange each column into a square matrix, and use separate row and column decoders.

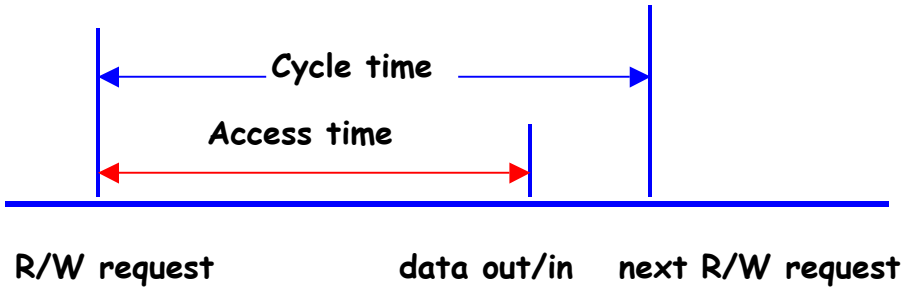
Needs ONE 4-16 address decoder to select an address

RAS (Row Address Select)



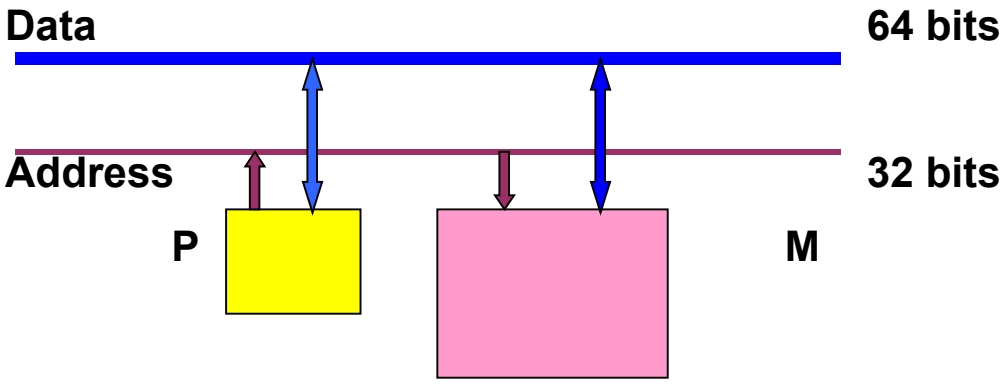
Fewer output lines of decoders simplify wiring.

Access time and cycle time



Memory Bandwidth

Number of bytes read from or written into the memory per second (SRAM ~ 0.5 ns – 5 ns, DRAM ~ 50 ns)



Q. Cycle time = 10 ns. Bandwidth = ?

Increasing Memory Bandwidth

1. Use expensive technology like SRAM.

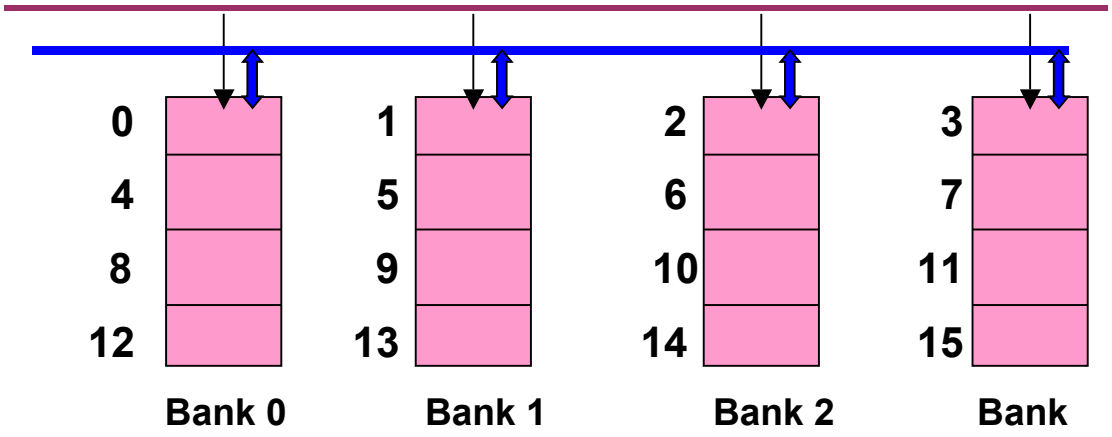
2. Use wide main memory.

64-bit transfer has been the *de facto* standard for PCs. Trend towards increasing it to 128 bits

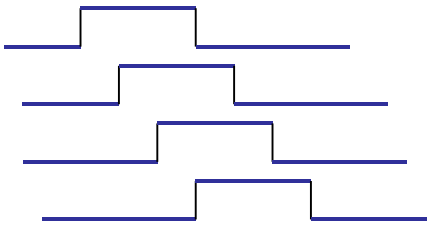
DEC Alpha 21064 main memory is 256 bit wide.

3. Interleaved Memory: Low order interleaving

Example configuration with interleaving factor=4



3

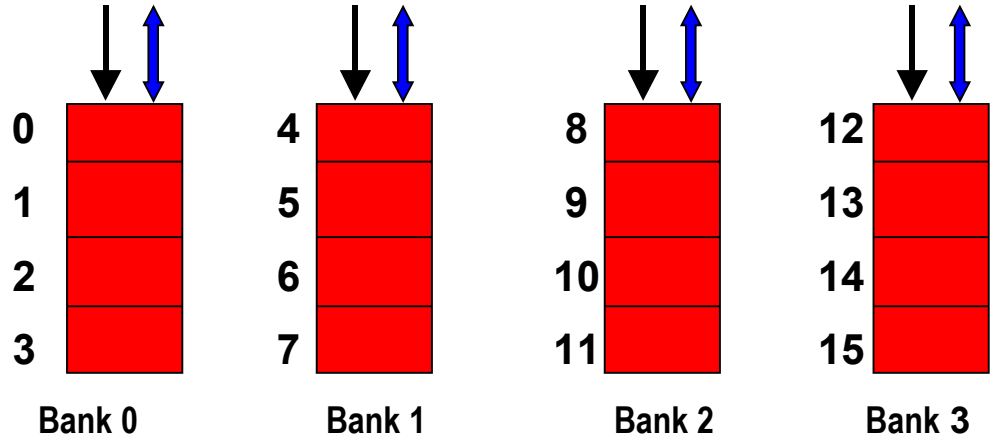


Reads from sequential locations can overlap
if the bus is fast. Improves memory bandwidth.
Non-sequential reads may lead to *bank conflict*.

Question. How can we avoid bank conflicts?

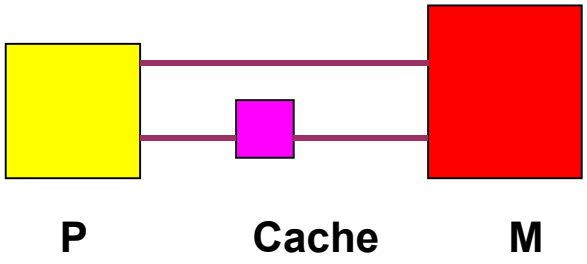
4. Interleaved Memory: High order interleaving

Wide main memory with independent memory banks (called superbanks) make better sense in multiprocessors. Better parallelism is possible, and provides better resilience to bank failure. Each bank has separate data & address lines.



Each superbank may (internally) use low order interleaving.

A First Look at Cache



Cache

It is a small high-speed memory. Stores data from some frequently used addresses (of main memory).

Cache hit

Data found in cache. Results in data transfer at maximum speed.

Cache miss

Data not found in cache. Processor loads data from M and copies into cache (miss penalty).

What are Hit and Miss ratios?

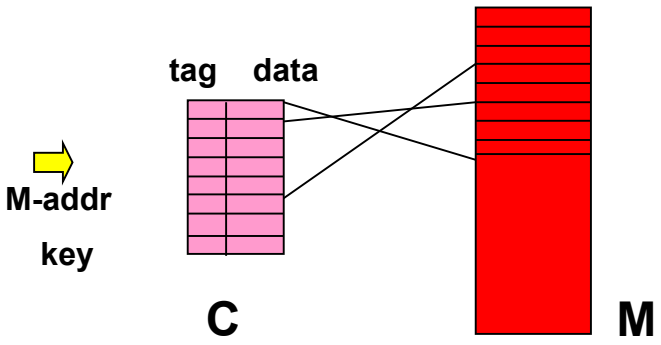
What happens during a write operation?

A Second Look at Cache

Cache Line

Cache is partitioned into lines (also called blocks). Each line has 4-64 bytes in it. During data transfer, a whole line is read or written.

Each line has a tag that indicates the address in M from which the line has been copied.



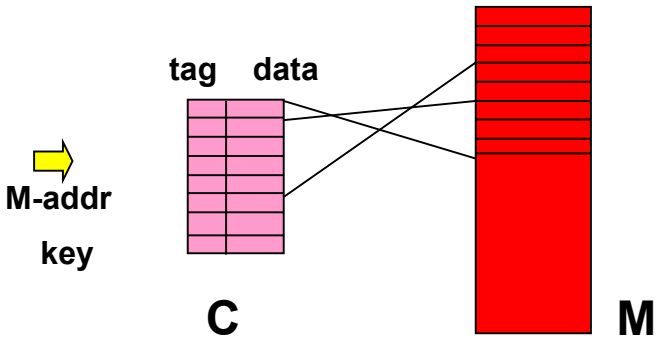
Cache hit is detected through an associative search of all the tags. Associative search provides a fast response to the query: ***Does this key match with any of the tags?***

Data is read only if a match is found.

Types of Cache

1. Fully Associative
2. Direct Mapped
3. Set Associative

Fully Associative Cache



No restriction on mapping from M to C.

Associative search of tags is expensive.

Feasible for very small size caches only.

The secret of success

Program locality.

Cache line replacement

To fetch a new line after a miss, an existing block has to be replaced. Two common policies for identifying the victim block are

- LRU (Least Recently Used)
- Random

Estimating Average Memory Access Time

Average memory access time =

Hit time + Miss rate x Miss penalty

Assume that

Hit time = 2 ns

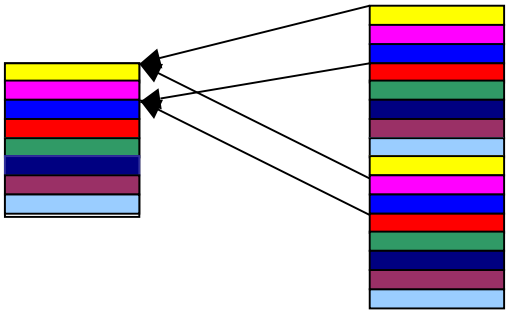
Miss rate = 5%

Miss penalty = 100 ns.

The average memory access time = 7 ns.

Better performance at a cheaper price!

Direct-Mapped Cache



A given memory block can be mapped into one and only cache block.

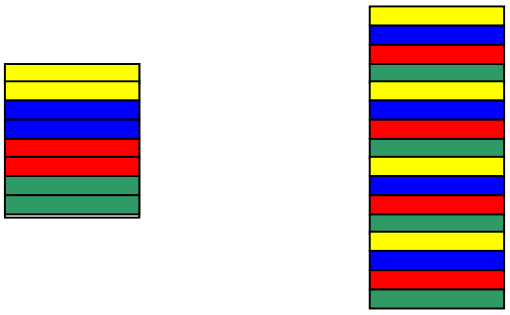
Advantage

No need of expensive associative search!

Disadvantage

Miss rate may go up due to possible increase of mapping conflicts.

Set-Associative Cache



Two-way Set-associative cache

N-way set-associative cache

Each M-block can now be mapped into any one of a **set of N C-blocks**. The sets are predefined. If total number of blocks in the cache = K then

N = 1	Direct-mapped cache
N = K	Fully associative cache

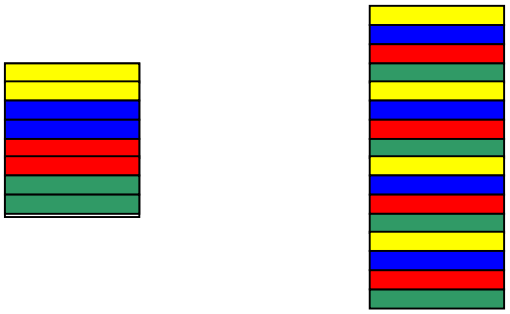
Most commercial cache have N = 2, 4, or 8.

Cheaper than a fully associative cache.

Lower miss ratio than a direct mapped cache.

But direct-mapped cache is the fastest.

Address translation



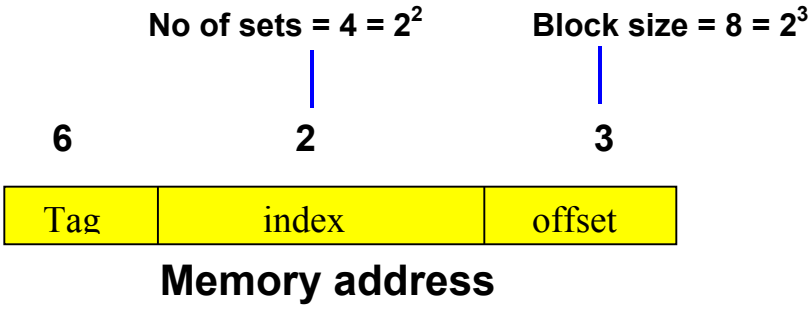
Main memory size = 2 KB

Block size = 8 bytes

Cache size = 64 bytes

Set size = 2

Number of sets in cache = 4



To locate an M-block in cache, check the tags in the set S = (M-block) mod (number of sets) i.e. the index field.

Specification of a cache memory

Block size	4-64 byte
Hit time	1-2 cycle
Miss penalty Access Transfer	8-32 cycles 6-10 cycles 2-22 cycles
Miss rate	1-20%
Cache size L1 L2	 8KB-32KB 128KB-2 MB
Cache speed L1 L2*	 0.5 ns 1.0 ns * on-chip cache