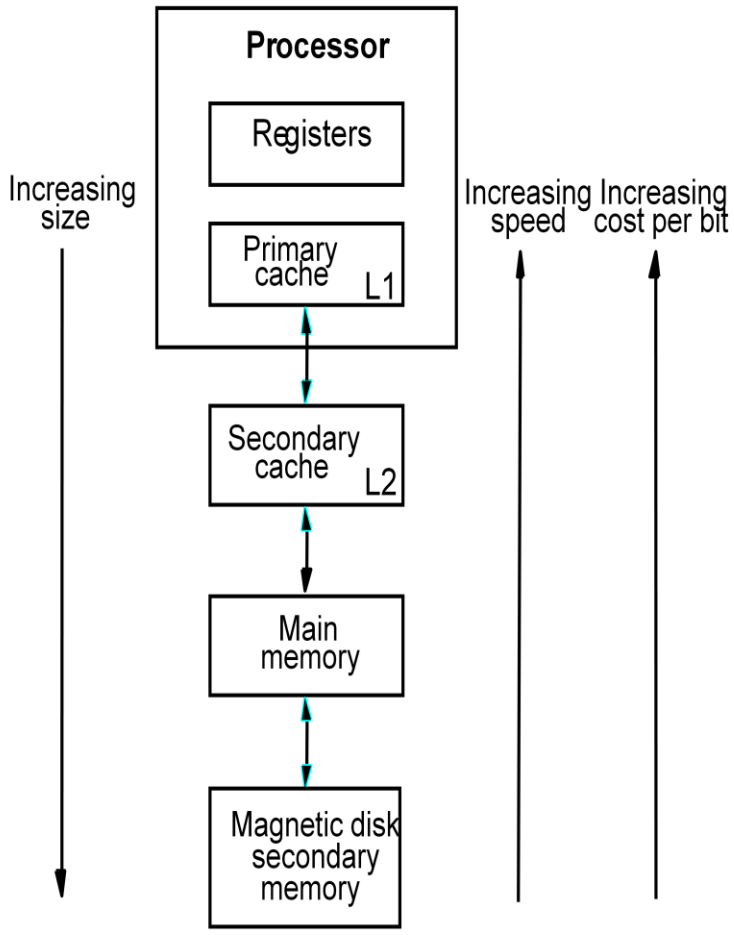
UNIT-V MEMORY ORGANIZATION



- Fastest access is to the data held in processor registers. Registers are at the top of the memory hierarchy.
- Relatively small amount of memory that can be implemented on the processor chip. This is processor cache.
- Two levels of cache. Level 1 (L1) cache is on the processor chip. Level 2 (L2) cache is in between main memory and processor.
- Next level is main memory, implemented as SIMMs. Much larger, but much slower than cache memory.
- Next level is magnetic disks. Huge amount of inexepensive storage.
- Speed of memory access is critical, the idea is to bring instructions and data that will be used in the near future as close to the processor as possible.

- The main memory of a computer is semiconductor memory.The main memory unit is basically consists of two kinds of memory:

RAM (RWM): Random access memory; which is volatile in nature.

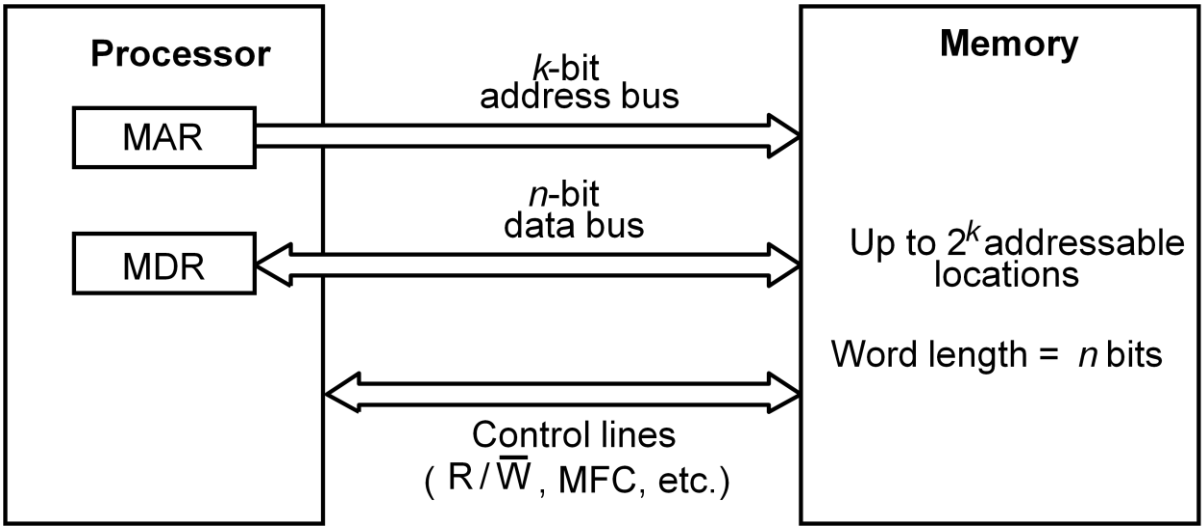
ROM: Read only memory; which is non-volatile.

- Permanent information is kept in ROM and the user space is basically in RAM.
- The smallest unit of information is known as bit (binary digit)
- in one memory cell one bit of information can be stored.

8 bit together is termed as a byte.

- The maximum size of main memory that can be used in any computer is determined by the addressing scheme.
- A computer that generates 16-bit address is capable of addressing upto 2^{16} which is equal to 64K memory location.
- Similarly, for 32 bit addresses, the total capacity will be 2^{32} which is equal to 4G memory location.
- In some computer, the smallest addressable unit of information is a memory word and the machine is called word-addressable .

- Maximum size of the Main Memory
- byte-addressable
- CPU-Main Memory Connection



Memory Operation

CPU initiates a memory operation by loading the appropriate data i.e., address to MAR.

- Memory read operation,

CPU sets the read memory control line to 1.

Then the contents of the memory location is brought to MDR.

The memory control circuitry indicates this to the CPU by setting MFC to 1.

- Memory write operation

CPU places the data into MDR .

Sets the write memory control line to 1.

Once the contents of MDR are stored in specified memory location, then the memory control circuitry indicates the end of operation by setting MFC to 1.

Measures for the speed of a memory

- Memory Access Time

A useful measure of the speed of memory unit is the time that elapses between the initiation of an operation and the completion of the operation .

(Ex. The time between Read and MFC)

- Memory cycle time

This is the minimum time delay between the initiation two independent memory operations

(Ex. two successive memory read operation).

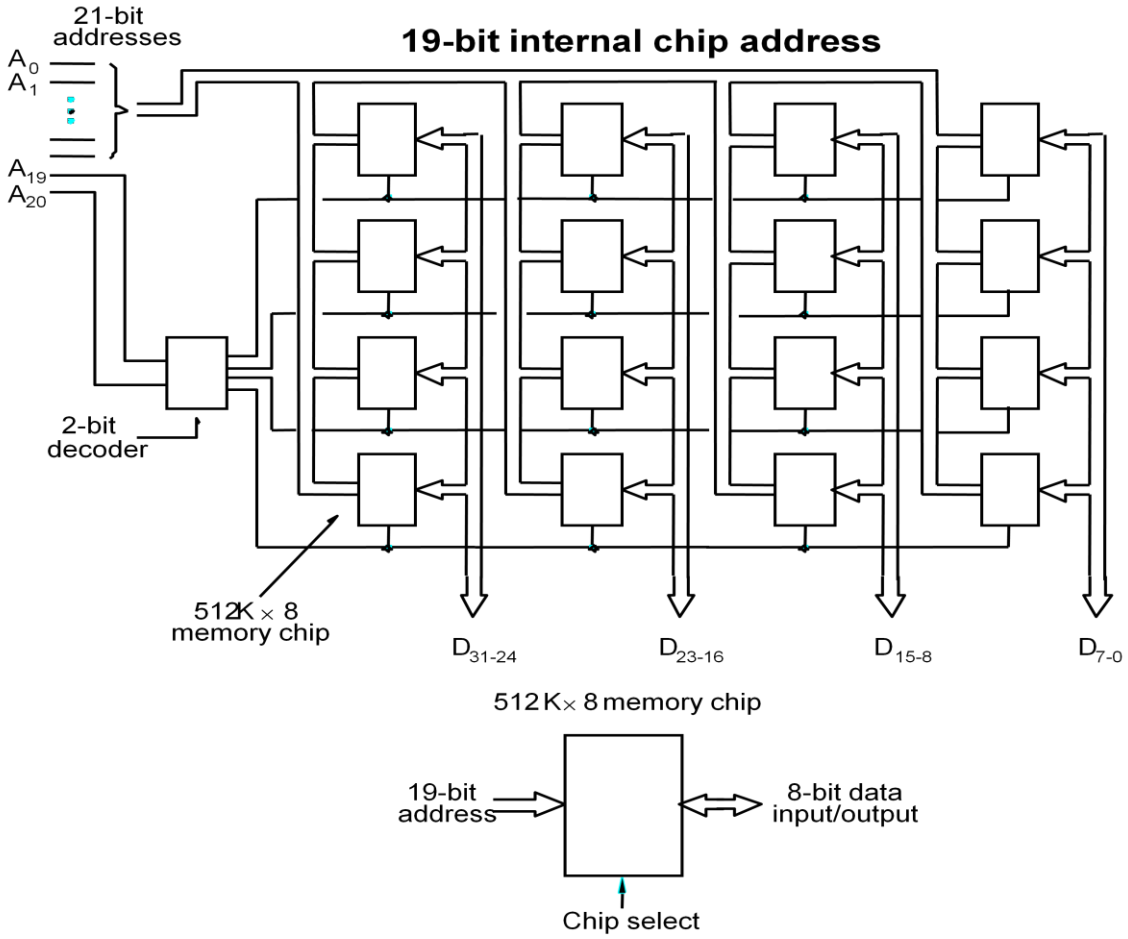
- Memory cycle time is slightly larger than memory access time.

Asynchronous DRAMs vs SRAMs

- Static RAMs (SRAMs):
  - Consist of circuits that are capable of retaining their state as long as the power is applied.
  - Volatile memories, because their contents are lost when power is interrupted.
  - Access times of static RAMs are in the range of few nanoseconds. (fast)
  - However, the cost is usually high.
  - Cache memory
- Dynamic RAMs (DRAMs):
  - Do not retain their state indefinitely.
  - Contents must be periodically refreshed.
  - Contents may be refreshed while accessing them for reading.
  - Main memory

- Both static and dynamic RAMs are volatile, that is, it will retain the information as long as power supply is applied.
- A dynamic memory cell is simpler and smaller than a static memory cell.
  Thus a DRAM is more dense, i.e., packing density is high (more cell per unit area).
  DRAM is less expensive than corresponding SRAM.
- DRAM requires the supporting refresh circuitry. For larger memories, the fixed cost of the refresh circuitry is more than compensated for by the less cost of DRAM cells.
- SRAM cells are generally faster than the DRAM cells. Therefore, to construct faster memory modules (like cache memory) SRAM is used.

Structure of large memories: Static memories



Implement a memory unit of 2M words of 32 bits each.Use 512x8 static memory chips.Each column consists of 4 chips.Each chip implements one byte position. A chip is selected by setting its chip select control line to 1. Selected chip places its data on the data output line, outputs of other chips are in high impedance state.21 bits to address a 32-bit word.High order 2 bits are needed to select the row, by activating the four Chip Select signals. 19 bits are used to access specific byte locations inside the selectedchip.

Memory controller

- To reduce the number of pins, the dynamic memory chips use multiplexed address inputs.
- Address is divided into two parts:
  - High-order address bits select a row in the array.
  - They are provided first, and latched using RAS signal.
  - Low-order address bits select a column in the row.
  - They are provided later, and latched using CAS signal.
- However, a processor issues all address bits at the same time.

- In order to achieve the multiplexing, memory controller circuit is inserted between the processor and memory.

Read-Only Memories (ROMs)

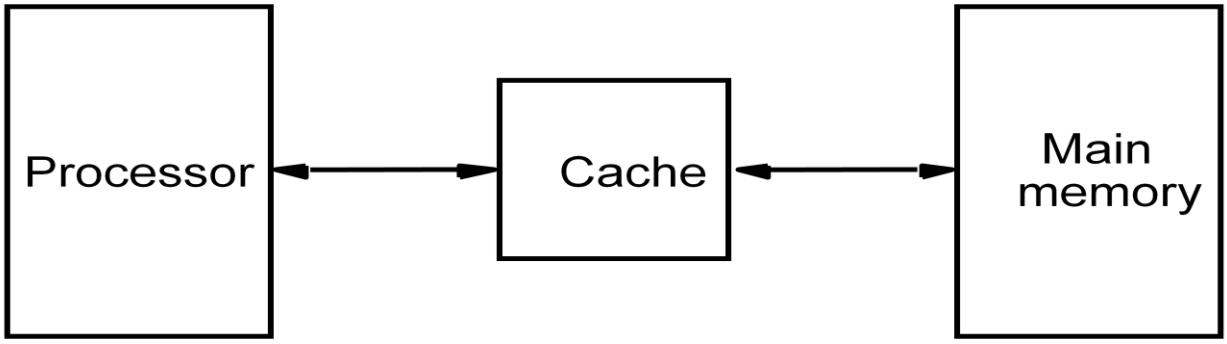
- SRAM and SDRAM chips are volatile:
  - Lose the contents when the power is turned off.
- Many applications need memory devices to retain contents after the power is turned off.
  - For example, computer is turned on, the operating system must be loaded from the disk into the memory.
  - Store instructions which would load the OS from the disk.
  - Need to store these instructions so that they will not be lost after the power is turned off.
  - We need to store the instructions into a non-volatile memory.
- Non-volatile memory is read in the same manner as volatile memory.
  - Separate writing process is needed to place information in this memory.
  - Normal operation involves only reading of data, this type of memory is called Read-Only memory (ROM).
- Read-Only Memory:
  - Data are written into a ROM when it is manufactured.
- Programmable Read-Only Memory (PROM):
  - Allow the data to be loaded by a user.
  - Process of inserting the data is irreversible.
  - Storing information specific to a user in a ROM is expensive.
  - Providing programming capability to a user may be better.
- Erasable Programmable Read-Only Memory (EPROM):
  - Stored data to be erased and new data to be loaded.
  - Flexibility, useful during the development phase of digital systems.
  - Erasable, reprogrammable ROM.
  - Erasure requires exposing the ROM to UV light.
- Electrically Erasable Programmable Read-Only Memory (EEPROM):
  - To erase the contents of EPROMs, they have to be exposed to ultraviolet light.

- Physically removed from the circuit.
- EEPROMs the contents can be stored and erased electrically.

- Flash memory:
  - Has similar approach to EEPROM.
  - Read the contents of a single cell, but write the contents of an entire block of cells.
  - Flash devices have greater density.
    - Higher capacity and low storage cost per bit.
  - Power consumption of flash memory is very low, making it attractive for use in equipment that is battery-driven.
  - Single flash chips are not sufficiently large, so larger memory modules are implemented using flash cards and flash drives.

Cache Memories

- Processor is much faster than the main memory.
  - As a result, the processor has to spend much of its time waiting while instructions and data are being fetched from the main memory.
  - Major obstacle towards achieving good performance.
- Speed of the main memory cannot be increased beyond a certain point.
- Cache memory is an architectural arrangement which makes the main memory appear faster to the processor than it really is.
- Cache memory is based on the property of computer programs known as “locality of reference”.
- Analysis of programs indicates that many instructions in localized areas of a program are executed repeatedly during some period of time, while the others are accessed relatively less frequently.
  - These instructions may be the ones in a loop, nested loop or few procedures calling each other repeatedly.
  - This is called “locality of reference”.
- Temporal locality of reference:
  - Recently executed instruction is likely to be executed again very soon.
- Spatial locality of reference:
  - Instructions with addresses close to a recently instruction are likely to be executed soon.



- Processor issues a Read request, a block of words is transferred from the main memory to the cache, one word at a time.
- Subsequent references to the data in this block of words are found in the cache.
- At any given time, only some blocks in the main memory are held in the cache. Which blocks in the main memory are in the cache is determined by a "mapping function".
- When the cache is full, and a block of words needs to be transferred

  from the main memory, some block of words in the cache must be

  replaced. This is determined by a "replacement algorithm".
- Existence of a cache is transparent to the processor. The processor issues Read and Write requests in the same manner.
- If the data is in the cache it is called a Read or Write hit.
- Read hit:
  - The data is obtained from the cache.
- Write hit:
  - Cache has a replica of the contents of the main memory.
  - Contents of the cache and the main memory may be updated simultaneously. This is the write-through protocol.
  - Update the contents of the cache, and mark it as updated by setting a bit known as the dirty bit or modified bit. The contents of the main memory are updated when this block is replaced. This is write-back or copy-back protocol.
- If the data is not present in the cache, then a Read miss or Write miss occurs.
- Read miss:
  - Block of words containing this requested word is transferred from the memory.
  - After the block is transferred, the desired word is forwarded to the processor.

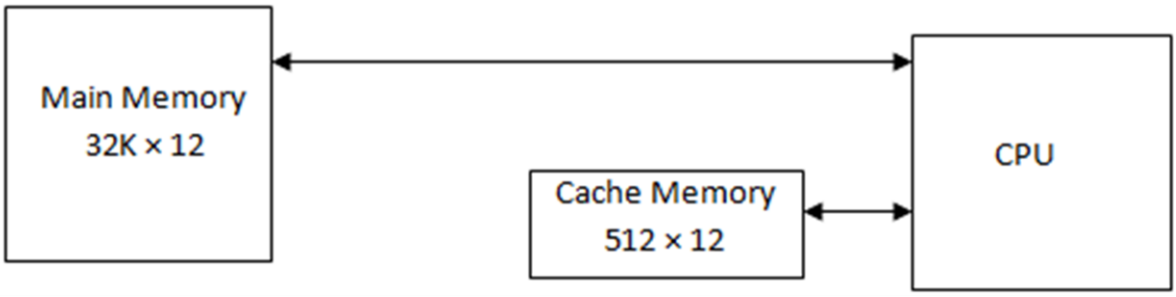
- The desired word may also be forwarded to the processor as soon as it is transferred without waiting for the entire block to be transferred. This is called load-through or early-restart.

- Write-miss:
  - Write-through protocol is used, then the contents of the main memory are updated directly.
  - If write-back protocol is used, the block containing the addressed word is first brought into the cache. The desired word is overwritten with new information.

Mapping functions

- Mapping functions determine how memory blocks are placed in the cache.
- Three mapping functions:
  - Direct mapping
  - Associative mapping
  - Set-associative mapping.

Example:



Direct Mapping

❑ Simplest mapping technique - each block of main memory maps to only one cache line

i.e. if a block is in cache, it must be in one specific place

Main memory locations can only be copied into one location in the cache, accomplished by dividing main memory into blocks that correspond in size with the cache.

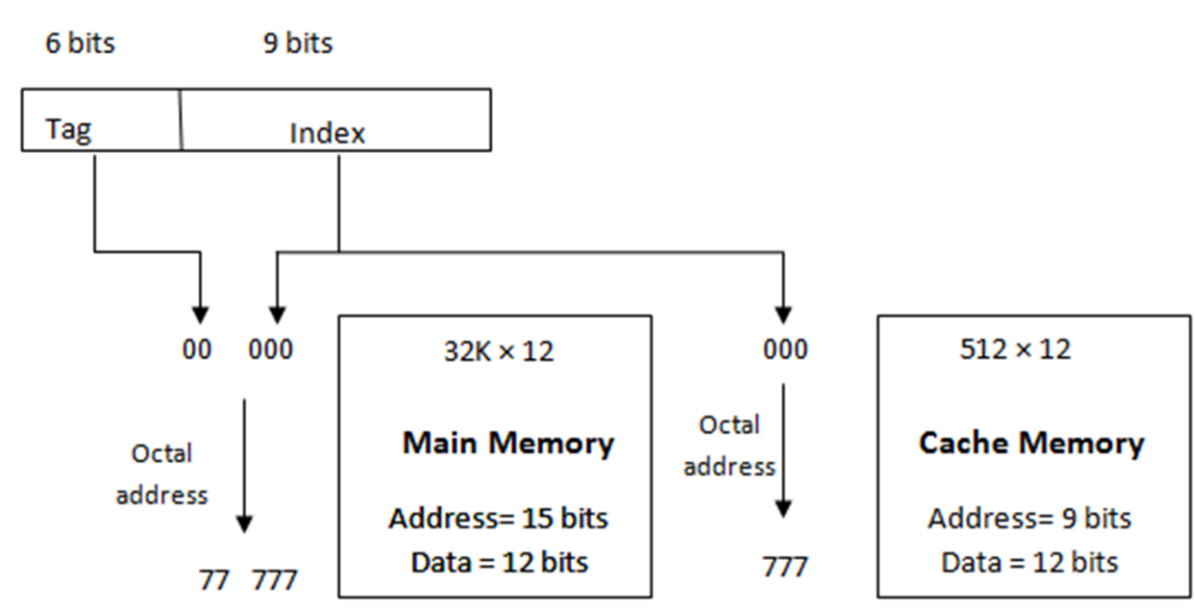
❑ Formula to map a memory block to a cache line:

◆ i = j mod c

- i=Cache Line Number (block address of cache)

- j=Main Memory Block Number(block address of main memory)
- c=Number of Lines in Cache(no.of cache blocks)

◆ i.e. we divide the memory block by the number of cache lines and the remainder is the cache line address



Main Memory

Address	Data
00 000	5670
00 777	7523
01 000	1256
01 777	5321
67 125	7432
77 777	5432

Cache Memory

Index	Tag	Data
000	00	5670
777	00	7523
000	01	1256
125	51	1560
777	77	5432

Direct Mapping pros & cons

- Simple
- Inexpensive
- Fixed location for given block
    - If a program accesses 2 blocks that map to the same line repeatedly, cache misses are very high

Associative Mapping

An associative mapping uses an associative memory.

This memory is being accessed using its contents.

Each line of cache memory will accommodate the address (main memory) and the contents of that address from the main memory.

That is why this memory is also called content addressable Memory (CAM). It allows each block of main memory to be stored in the cache.

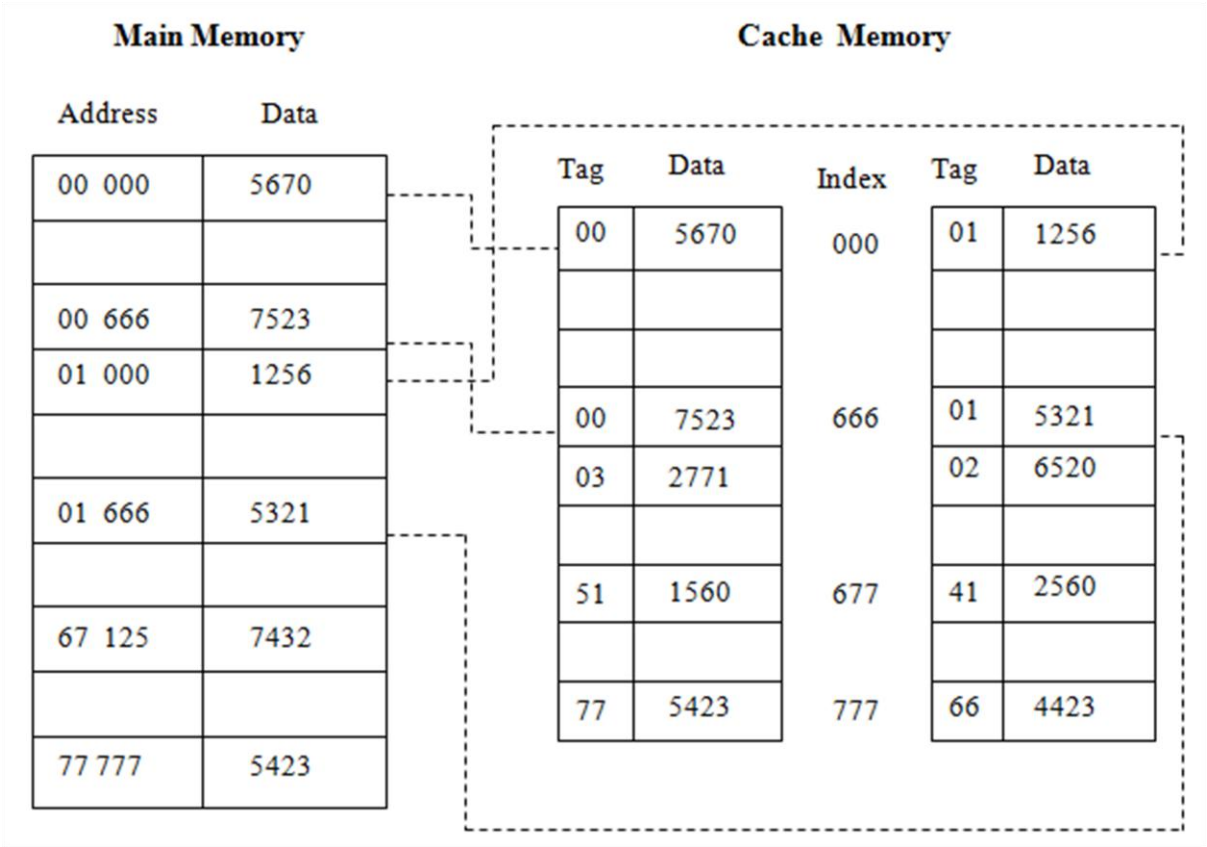
- A fully associative mapping scheme can overcome the problems of the direct mapping scheme
    - A main memory block can load into any line of cache
    - Memory address is interpreted as tag and word
    - Tag uniquely identifies block of memory
    - Every line's tag is examined for a match
    - Also need a Dirty and Valid bit
- But Cache searching gets expensive!
    - Ideally need circuitry that can simultaneously examine all tags for a match
    - Lots of circuitry needed, high cost
- Need replacement policies now that anything can get thrown out of the cache (will look at this shortly)

Set Associative Mapping

That is the easy control of the direct mapping cache and the more flexible mapping of the fully associative cache.

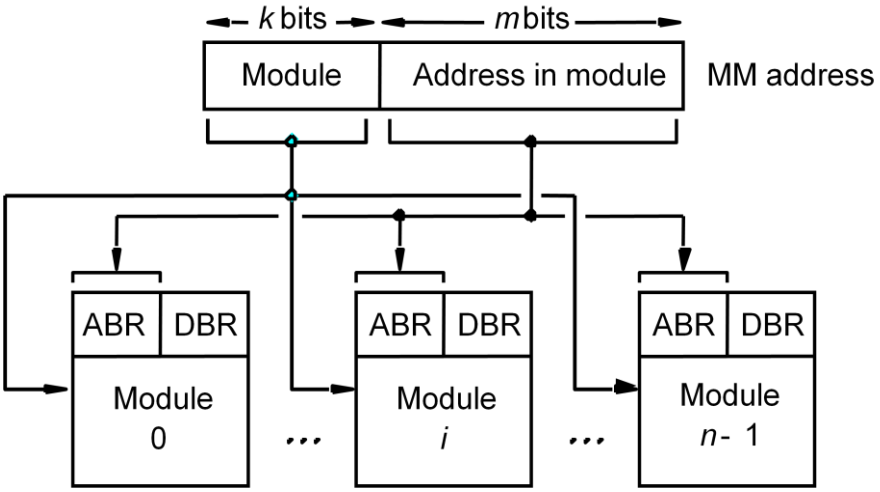
In set associative mapping, each cache location can have more than one pair of tag + data items.

That is more than one pair of tag and data are residing the at same location of cache memory. If one cache location is holding two pair of tag + data items, that is called *2-way associative mapping*.

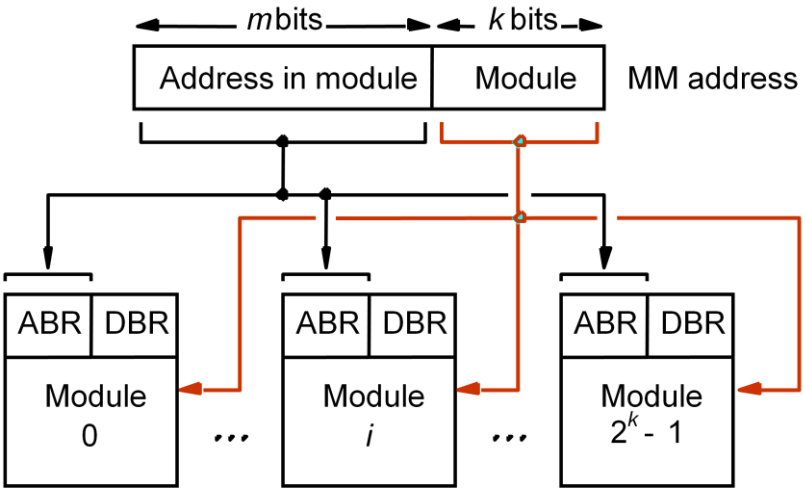


Interleaving

- Divides the memory system into a number of memory modules. Each module has its own address buffer register (ABR) and data buffer register (DBR).
- Arranges addressing so that successive words in the address space are placed in different modules.
- When requests for memory access involve consecutive addresses, the access will be to different modules.
- Since parallel access to these modules is possible, the average rate of fetching words from the Main Memory can be increased.



- Consecutive words are placed in a module.
- High-order k bits of a memory address determine the module.
- Low-order m bits of a memory address determine the word within a module.
- When a block of words is transferred from main memory to cache, only one module is busy at a time.



- Consecutive words are located in consecutive modules.
- Consecutive addresses can be located in consecutive modules.
- While transferring a block of data, several memory modules can be kept busy at the same time.

Writing into the cache

When memory write operations are performed, CPU first writes into the cache memory. These modifications made by CPU during a write operations, on the data saved in cache, need to be written back to main memory or to auxiliary memory.

The two popular cache write policies are:

Write –through and Write back

- **Write-through:**

- Each write operation involves writing to the main memory.
- If the processor has to wait for the write operation to be complete, it slows down the processor.
- Processor does not depend on the results of the write operation.
- Write buffer can be included for temporary storage of write requests.
- Processor places each write request into the buffer and continues execution.
- If a subsequent Read request references data which is still in the write buffer, then this data is referenced in the write buffer.

- **Write-back:**

- Block is written back to the main memory when it is replaced due to a miss.
- If the processor waits for this write to complete, before reading the new block, it is slowed down.
- Fast write buffer can hold the block to be written, and the new

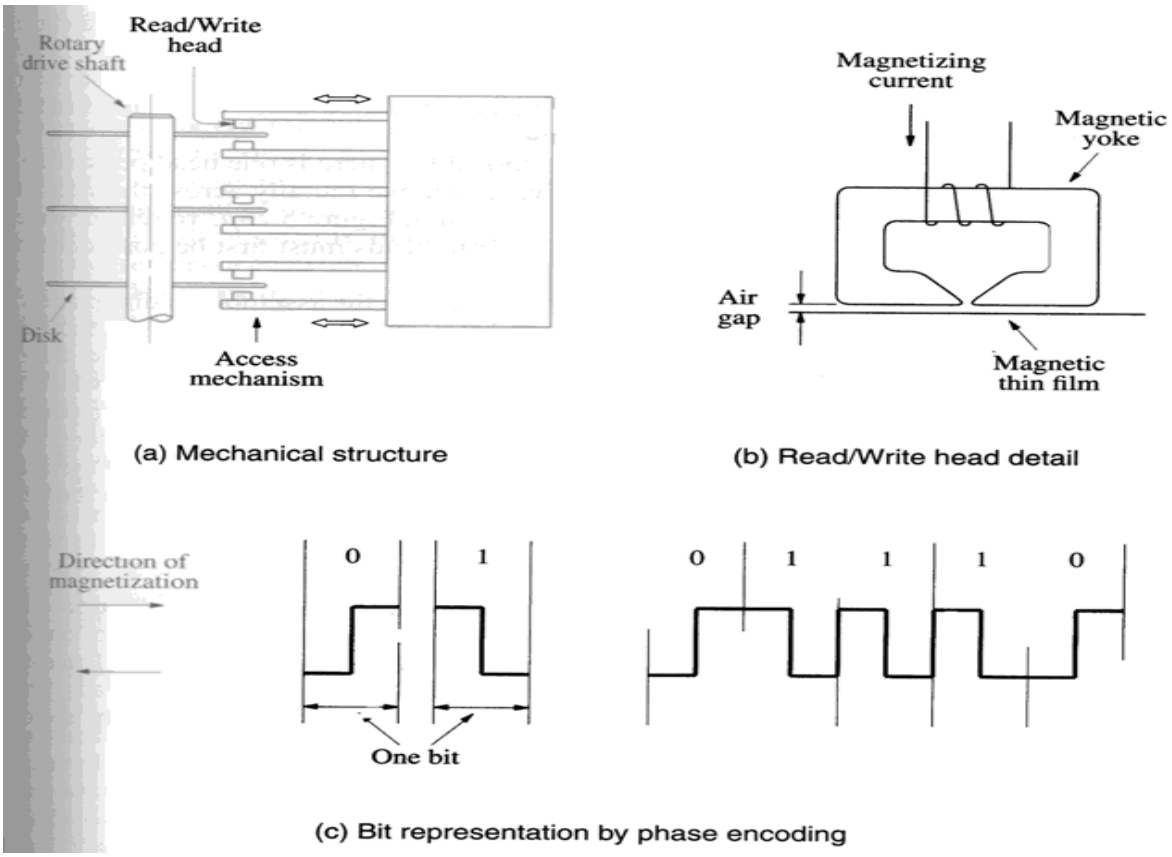
  block can be read first.
- Dirty bit is set when we write to the cache, this indicates the cache is now inconsistent with main memory.
- Dirty bit for cache slot is cleared when update occurs.

Cache algorithms (Page replacement policies)

- Replacement algorithms are used when there are no available space in a cache in which to place a data. Four of the most common cache replacement algorithms are described
- First-in, First-out(FIFO): Evict the page that has been in the cache the longest time.
- Least recently used (LRU): Evict the page whose last request occurred furthest in the past.(least recently used page)
- Least Frequently Used (LRU): The LRU algorithm selects for replacement the item that has been least frequently used by the CPU.
- Random: Choose a page at random to evict from the cache.

Secondary Storage

1. Magnetic Hard Disks



Access Data on a Disk

- ✓ Sector header
- ✓ Following the data, there is an error-correction code (ECC).
- ✓ Formatting process
- ✓ Difference between inner tracks and outer tracks
- ✓ Access time – seek time / rotational delay (latency time)
- ✓ Data buffer/cache

Disk Controller

- ✓ Seek
- ✓ Read
- ✓ Write
- ✓ Error checking

2. Optical Disks

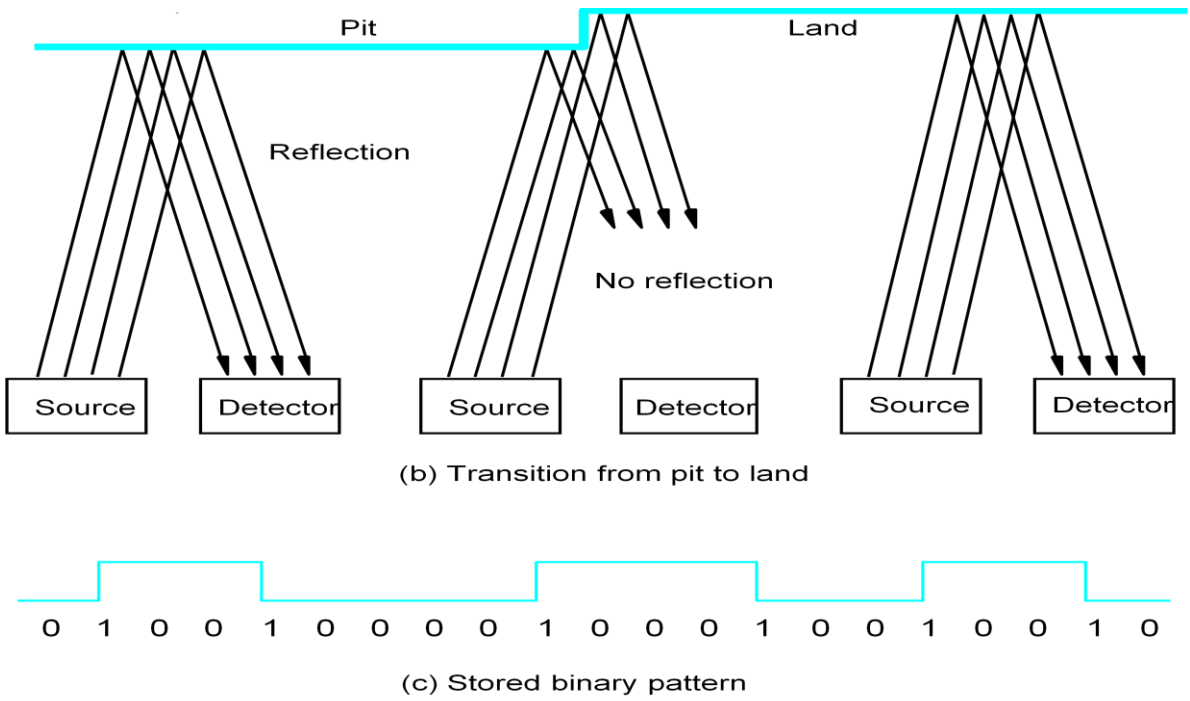


Fig. Optical disk.

- ✓ CD-ROM
- ✓ CD-Recordable (CD-R)
- ✓ CD-ReWritable (CD-RW)
- ✓ DVD
- ✓ DVD-RAM