Computer Organization and Architecture

Unit III

Dr. Sumita Nainan, MPSTME,NMIMS University

Unit -3 Memory Organization

- **Internal Memory-** Memory characteristics and memory hierarchy.
- **Cache Memory-** Elements of cache design, Address mapping and **Translation-Direct mapping**, Address mapping and translation- **Associative mapping**, Address mapping and translation -**Set associative mapping**
- **Performance characteristics of two level memory**
- **Semiconductor main memory-** Types of RAM, DRAM and SRAM, Chip logic, Memory module organization.
- **High speed memories-** Associative memory, High speed
- Memories- **Interleaved memory.**

Characteristics Of Memory Systems

- The key characteristics of memory systems are:
- 1.Location
- 2. Capacity
- 3. Unit of Transfer
- 4. Access Method
- 5. Performance
- 6.Physical Type

Characteristics Of Memory Systems

1. Location: The computer memory is placed in three different locations:

a. **CPU** :- It is in the form of CPU registers and its internal cache memory (16K bytes in case of Pentium)

b. **Internal** :- It is in the main memory of the system which CPU can access directly

c. **External**: It is in the form of secondary storage devices such as magnetic disk, tapes, etc. The CPU accesses this memory with the help of I/O controllers

2. Capacity:

- It is expressed using two terms: **Word size and number of words.**

a. Word size :- It is expressed in bytes (8-bit). The common word sizes are 8,16 and 32 bits.

b. Number of word :- This term specifies the number of words available in the particular memory device.

e.g. If memory capacity is 4K X 8, then its word size is 8 and number of words are 4K=4096.

Characteristics Of Memory Systems

3. Unit of Transfer :

- It is the **maximum number of bits that can be read or written** into the memory at a time.
- In case of main memory, most of the times it is equal to word size.
- In case of external memory, unit of transfer is not limited to word size, it is often larger than a word and it is referred to as blocks.

4. Access Method :

- There are two different method generally used for memory access.

a. Sequential access:

Here, memory is organized into units of data, called records. If current record is 1, then in order to read record N, it is necessary to read physical records 1 through N -1. A tape drive is an example of sequential access memory.

b. Random access:

Here, each addressable location in memory has an unique address. It is possible to access any memory location at random.

Characteristics Of Memory Systems

5. Performance :-

The performance of the memory system is determined using three parameters:

I. **Access time**: In case of random access memory, it is the time taken by memory to complete read/ write operation from the instant that an address is sent to the memory. On the other hand, for nonrandom access memory, access time is the time it takes to position the read-write mechanism at the desired location.

II. **Memory cycle time**: This term is used only with random access memory and it is defined as *access time plus additional time required before a second access can commence*.

III. **Transfer rate:** It is defined as the rate at which data can be transferred into or out of a memory unit.

6.Physical Type :-

Two most common physical types used today are **semiconductor memory** and **magnetic** surface memory.

- **Physical characteristics :-**

a. **Volatile/Nonvolatile**: If memory can hold data even if power is turned off, it is called as nonvolatile memory; otherwise it is called as volatile memory.

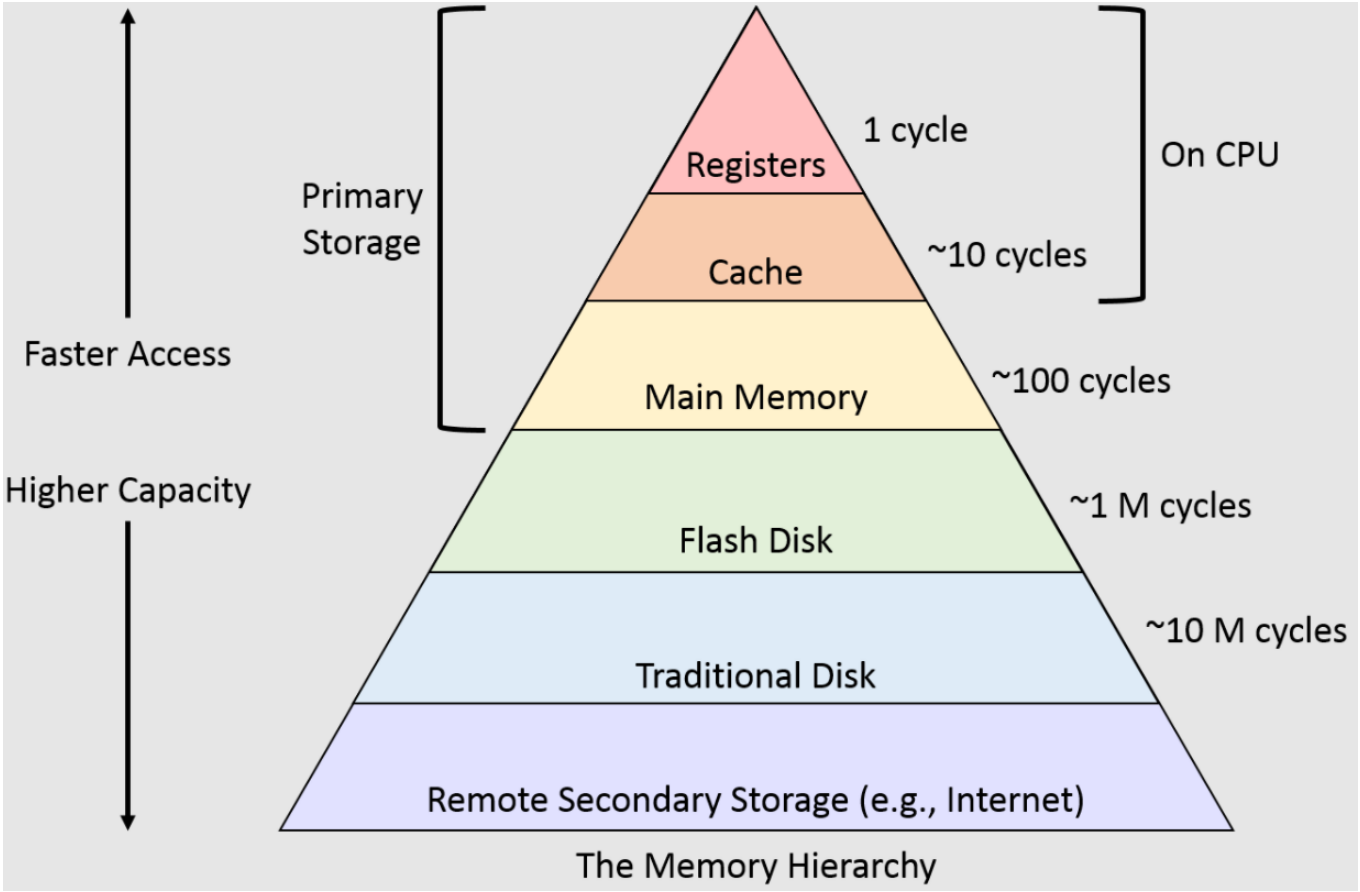
b. **Erasable/Non erasable**: The memories in which data is once programmed cannot be erased are called as nonerasable memories. On the other hand, if data in the memory is erasable then memory is called as erasable memory.

Characteristics Of Memory Systems

Technology	Storage	Access Method	Alterability	performance	Typical access time t_A
Semiconductor RAM	Electronic	Random	Read/write	Volatile	10 ns
Semiconductor ROM	Electronic	Random	Read only	Non Volatile	10 ns
Magnetic (Hard) disk	Magnetic	Semi-random	Read/write	Non Volatile	50 ns
Optical disk CD-ROM	optical	Semi-random	Read only	Non Volatile	100 ms
Erasable optical disk	optical	Semi-random	Read/write	Non Volatile	100 ms
Magnetic tape	Magnetic	Serial	Read/write	Non Volatile	1 s Depend on access location

Characteristics of some common memory technologies

Memory Hierarchy



The Memory Hierarchy

Characteristics Of Memory Systems

- Ideally, computer memory should be fast, large and inexpensive.
- Unfortunately, it is impossible to meet all the three of these requirements simultaneously. Increased speed and size are achieved at increased cost.
- Very fast memory system can be achieved if SRAM chips are used.
- These chips are expensive and hence it is impracticable to build a large main memory using SRAM chips.
- The only alternative is to use DRAM chips for large main memories.

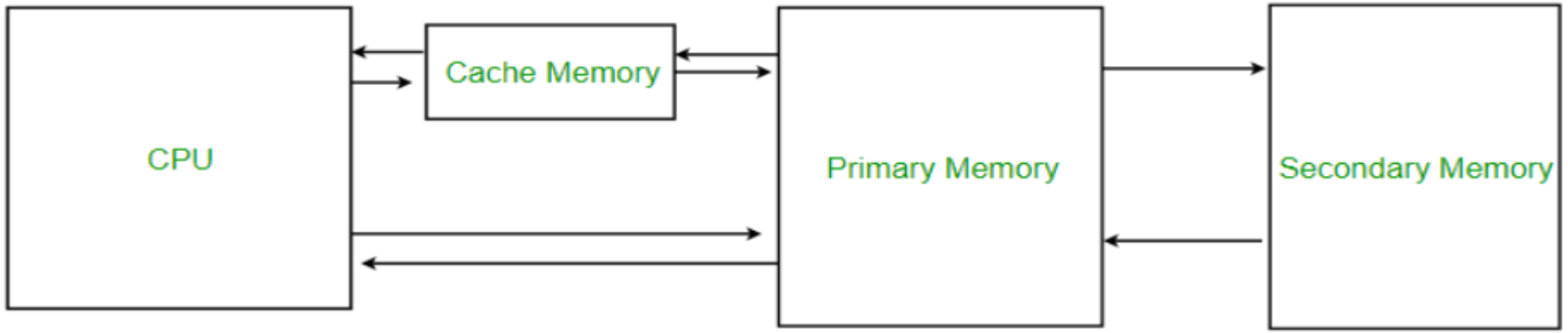
- Processor fetches the code and data from the main memory to execute the program.
- The DRAMs which form the main memory are slower devices.
- So it is necessary to insert wait states in memory read/ write cycles.
- This reduces the speed of execution.
- The solution to this problem is that most of the computer programs work with only small sections of code and data at a particular time.

Characteristics Of Memory Systems

- In the memory system small section of SRAM is added along with main memory, referred to as cache memory.
- The program (code) and data that work at a particular time is usually accessed from the cache memory.
- This is accomplished by loading the active part of code and data from main memory and cache memory.
- The cache controller looks after this swapping between main memory and cache memory with the help of DMA controller.
- The cache memory just discussed is called secondary cache.
- Recent processors have the built-in cache memory called primary cache.
- DRAMs along with cache allow main memories in the range of tens of megabytes to be implemented at a reasonable cost, size and better speed performance.
- But the size of memory is still small compared to the demands of large programs with voluminous data.
- A solution is provided by using secondary storage, mainly magnetic disk and magnetic tapes to implement large memory spaces.

Cache memory

- **Cache Memory** is a special very high-speed memory.
- It is used to speed up and synchronize with high-speed CPU.
- Cache memory is costlier than main memory or disk memory but economical than CPU registers.
- Cache memory is an extremely fast memory type that acts as a buffer between RAM and the CPU.
- It holds frequently requested data and instructions so that they are immediately available to the CPU when needed.
- Cache memory is used to reduce the average time to access data from the Main memory.
- The cache is a smaller and faster memory which stores copies of the data from frequently used main memory locations.
- There are various different independent levels of caches in a CPU, which store instructions and data.



Cache memory

There are a few basic design elements that serve to classify and differentiate cache architectures. They are :

1. Cache Addresses
2. Cache Size
3. Mapping Function
4. Replacement Algorithm
5. Write Policy
6. Line Size
7. Number of caches

Cache memory

1. Cache Addresses

- When virtual addresses are used, the cache can be placed between the processor and the memory management unit (MMU) or between the MMU and main memory.
- A logical cache, also known as a virtual cache, stores data using virtual addresses. The processor accesses the cache directly, without going through the MMU

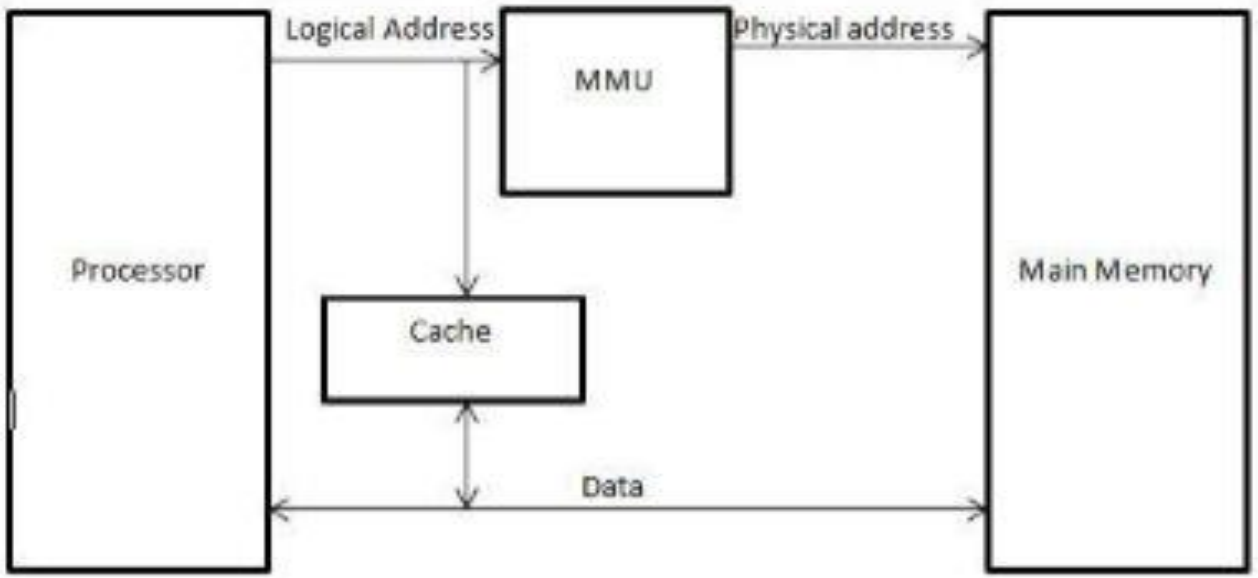


Figure 3: Logical cache

Cache memory

A physical cache stores data using main memory physical addresses. This organization is shown in Figure 4. One advantage of the logical cache is that cache access speed is faster than for a physical cache, because the cache can respond before the MMU performs an address translation.

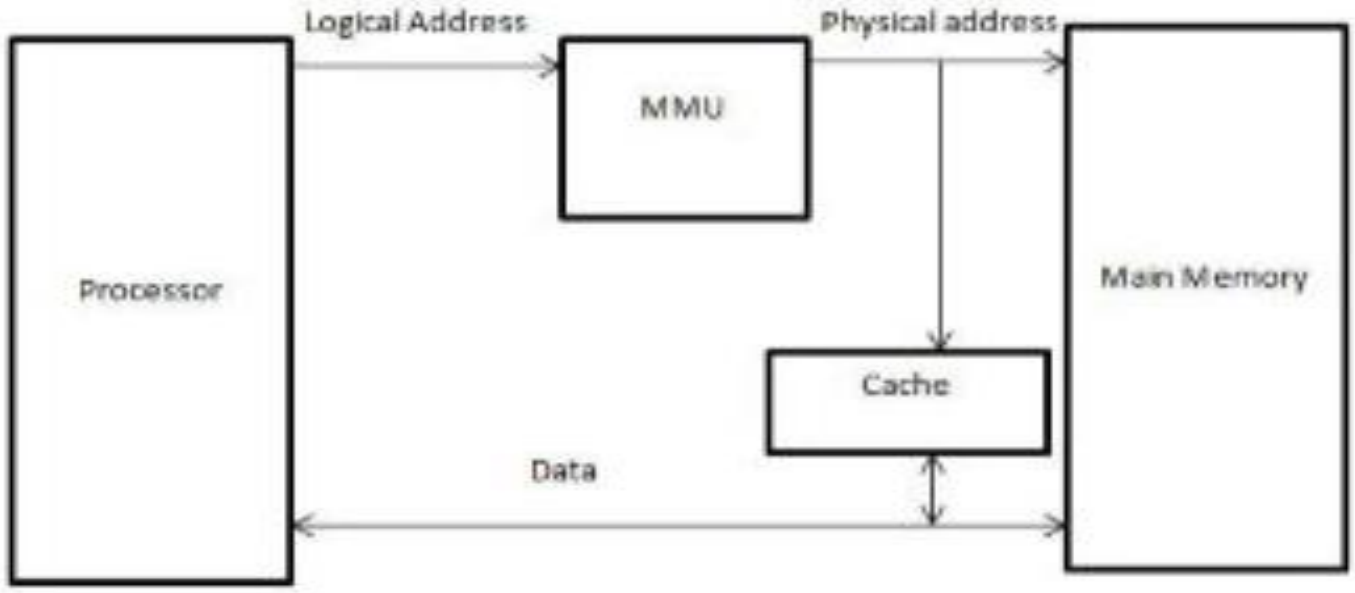


Figure 4: Physical cache

Cache memory

2. Cache Size:

The size of the cache should be small enough so that the overall average cost per bit is close to that of main memory alone and large enough so that the overall average access time is close to that of the cache alone.

3. Mapping Function:

- As there are fewer cache lines than main memory blocks, an algorithm is needed for mapping main memory blocks into cache lines.
- Further, a means is needed for determining which main memory block currently occupies a cache line.
- The choice of the mapping function dictates how the cache is organized.
- Three techniques can be used: direct, associative, and set associative.
- **DIRECT MAPPING:** The simplest technique, known as direct mapping, maps each block of main memory into only one possible cache line.
- The direct mapping technique is simple and inexpensive to implement.
- **ASSOCIATIVE MAPPING:** Associative mapping overcomes the disadvantage of direct mapping by permitting each main memory block to be loaded into any line of the cache
- **SET-ASSOCIATIVE MAPPING:** Set-associative mapping is a compromise that exhibits the strengths of both the direct and associative approaches. With set-associative mapping, block can be mapped into any of the lines of set j.

Cache memory

4. Replacement Algorithms:

- Once the cache has been filled, when a new block is brought into the cache, one of the existing blocks must be replaced.
- For direct mapping, there is only one possible line for any particular block, and no choice is possible.
- For the associative and set associative techniques, a replacement algorithm is needed.
- To achieve high speed, such an algorithm must be implemented in hardware.
- Least Recently Used (LRU), Least Frequently Used(LFU), First In First Out (FIFO) are some replacement algorithms.

5. Write Policy

- When a block that is resident in the cache is to be replaced, there are two cases to consider.
- If the old block in the cache has not been altered, then it may be overwritten with a new block without first writing out the old block.

Cache memory

5. Write Policy

- If at least one write operation has been performed on a word in that line of the cache, then main memory must be updated by writing the line of cache out to the block of memory before bringing in the new block.
- The simplest policy is called write through. Using this technique, all write operations are made to main memory as well as to the cache, ensuring that main memory is always valid.
- An alternative technique, known as write back, minimizes memory writes.
- With write back, updates are made only in the cache.
- When an update occurs, a dirty bit, or use bit, associated with the line is set.
- Then, when a block is replaced, it is written back to main memory if and only if the dirty bit is set.

Cache memory

6. Line Size

- Another design element is the line size.
- When a block of data is retrieved and placed in the cache, not only the desired word but also some number of adjacent words is retrieved.
- Basically, as the block size increases, more useful data are brought into the cache.
- The hit ratio will begin to Increase
- However, as the block becomes even bigger and the probability of using the newly fetched information becomes less than the probability of reusing the information that has to be replaced, the hit ratio will decrease.
- The relationship between block size and hit ratio is complex, depending on the locality characteristics of a particular program, and no definitive optimum value is found as of yet.

Cache memory

7.Number of Caches

- When caches were originally introduced, the typical system had a single cache.
- More recently, the use of multiple caches has become an important aspect.
- There are two design issues surrounding number of caches.
- MULTILEVEL CACHES: Most contemporary designs include both on-chip and external caches. The simplest such organization is known as a two-level cache, with the internal cache designated as level 1 (L1) and the external cache designated as level 2 (L2). There can also be 3 or more levels of cache. This helps in reducing main memory accesses.
- UNIFIED VERSUS SPLIT CACHES: Earlier on-chip cache designs consisted of a single cache used to store references to both data and instructions. This is the unified approach. More recently, it has become common to split the cache into two: one dedicated to instructions and one dedicated to data. These two caches both exist at the same level. This is the split cache. Using a unified cache or a split cache is another design issue.

Cache memory

- A cache memory system includes a small amount of fast memory (SRAM) and a large amount of slow memory (DRAM).
- This system is configured to simulate a large amount of fast memory.
- Figure7.2 shows a cache memory system.
- It consists of following sections:
- Cache: This block consists of static RAM (SRAM).
- Main Memory: This block consists of Dynamic RAM (DRAM).
- Cache Controller: This block implements the cache logic.

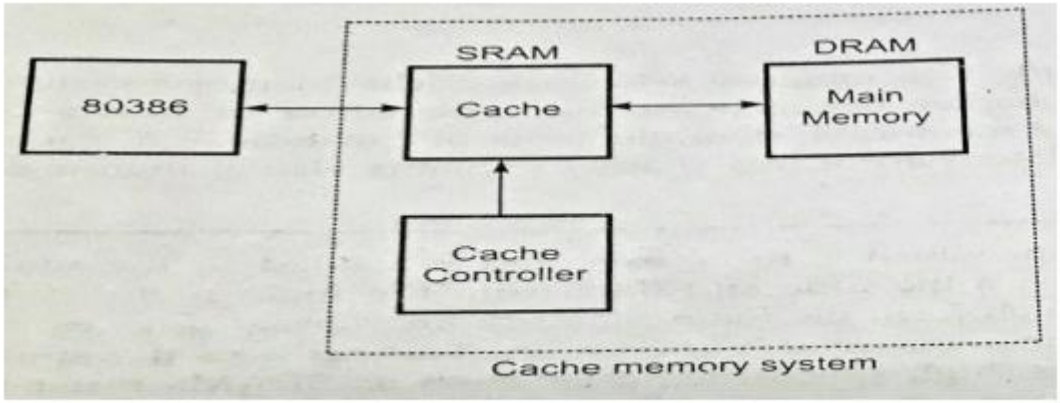


Figure 7.2 Cache memory system

Cache memory

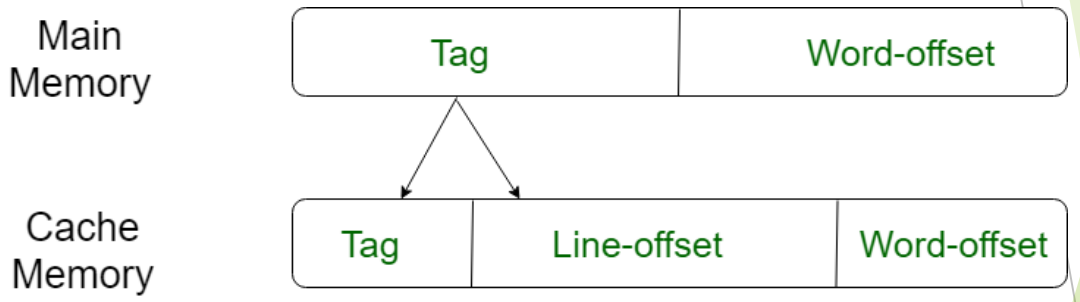
- **Cache Performance:**
- When the processor needs to read or write a location in main memory, it first checks for a corresponding entry in the cache.
- If the processor finds that the memory location is in the cache, a **cache hit** has occurred and data is read from cache
- If the processor **does not** find the memory location in the cache, a **cache miss** has occurred. For a cache miss, the cache allocates a new entry and copies in data from main memory, then the request is fulfilled from the contents of the cache.
- The performance of cache memory is frequently measured in terms of a quantity called **Hit ratio.**
-
- Hit ratio = hit / (hit + miss) = no. of hits/total accesses

Cache memory

- **Cache Mapping:**
- There are three different types of mapping used for the purpose of cache memory which are as follows: ***Direct mapping, Associative mapping, and Set-Associative mapping. These are explained below.***
- **Direct Mapping –**
  The simplest technique, known as direct mapping, maps each block of main memory into only one possible cache line.
- Or
- In Direct mapping, assign each memory block to a specific line in the cache.
- If a line is previously taken up by a memory block when a new block needs to be loaded, the old block is trashed.
- An address space is split into two parts ***index field and a tag field***.
- The cache is used to store the tag field whereas the rest is stored in the main memory. Direct mapping`s performance is directly proportional to the Hit ratio.

Cache memory

- i = j modulo m
- where
- i=cache line number
- j= main memory block number
- m=number of lines in the cache
- For purposes of cache access, each main memory address can be viewed as consisting of three fields.
- The least significant w bits identify a unique word or byte within a block of main memory.
- In most contemporary machines, the address is at the byte level.
- The remaining s bits specify one of the 2^s blocks of main memory.
- The cache logic interprets these s bits as a tag of s-r bits (most significant portion) and a line field of r bits.
- This latter field identifies one of the $m=2^r$ lines of the cache.



Cache organization

- Usually, the cache memory can store a reasonable number of blocks at any given time, but this number is small compared to the total number of blocks in the main memory.
- The correspondence between the main memory blocks and those in the cache is specified by a mapping function.
- There are two main mapping techniques which decide the cache organization:

1. Direct-mapping technique

2. Associative-mapping technique

The associative mapping technique is further classified as fully associative and set associative techniques.

To discuss these techniques of cache mapping we consider a cache consisting of 128 block of 32 words each, for a total of 4096(4K) words, and assume that the main memory has 64K words. This 64K words of main memory is addressable by a 16-bit address and it can be viewed as 2 K blocks of 16 words each. The group of 128 blocks of 32 words each in main memory from a page.

Cache organization

1. Direct-mapping technique

- In this technique, each block from the main memory has only one possible location in the cache organization.
- In this example, the block I of the main memory maps on to block I module 128 of the cache, as shown in figure
- Therefore, whenever one of the main memory blocks 0, 128,256,…… is loaded in the cache, it is stored in cache block 0. Blocks 1, 129, 257,…… are stored in cache block 1, and so on.

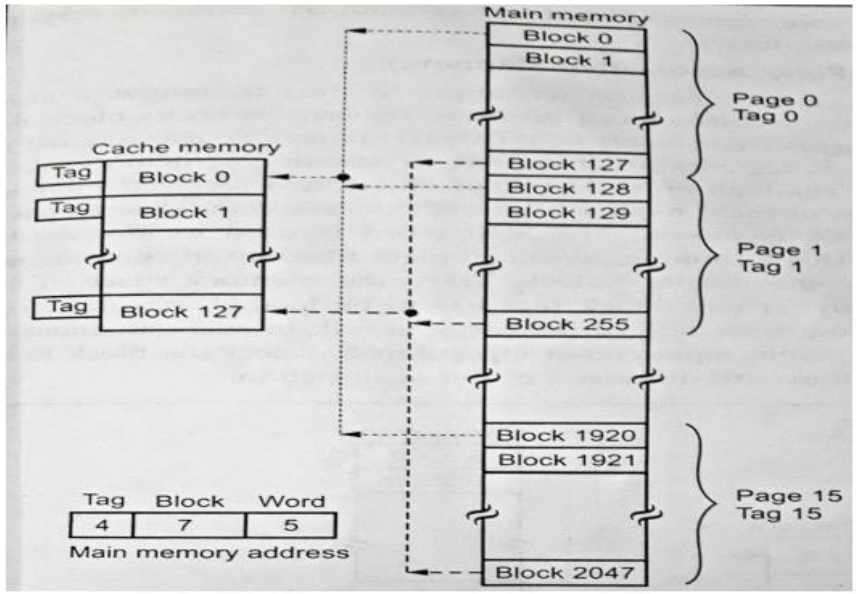


Figure 7.3 Direct mapped cache

Cache organization

1. Direct-mapping technique

- The lower order 5-bits select one of the 32 words in a block.
- This field is known as word field.
- The second field known as block field is used to distinguish a block from other blocks.
- When a new block enters the cache, the 7- bit cache block field determiners the cache position in which this block must be stored. The third field is a tag field. It is used to store the high-order 4-bits of memory address of the block.
- These 4-bit (tag bits) are used to identify which of the 16 blocks (pages) that are mapped into the cache.
- When memory is accessed, the 7-bit cache block field of each address generated by CPU points to a particular block location in the cache.

Cache organization

1. Direct-mapping technique

- The high-order 4-bits of the address are compared with the tag bits associated with that cache location.
- If they match, then the desired word is in that block of the cache.
- If there is no match, then the block containing the required word must first be read from the main memory and loaded into the cache.
- This means that to determine whether requested word is in the cache, only tag field is necessary to be compared. This needs only one comparison.
- The main drawback of direct mapped cache is that if processor needs to access same memory locations from two different pages of the main memory frequently, the controller has to access main memory frequently.
- Since only one of these locations can be in the cache at a time.
- For example , if processor want to access memory location 100H from page 0 and then from page 2, the cache controller has to access page 2 of the main memory. Therefore, we can say that direct-mapped cache is easy to implement, however, it is not very flexible.

Cache organization

1. **Associative-mapping technique**

- The figure 7.4 shows the associative mapping technique.
- In this technique, a main memory block can be placed into any cache block position.
- As there is no fix block, the memory address has only two fields: word and tag.
- This technique is also referred to as fully-associative cache

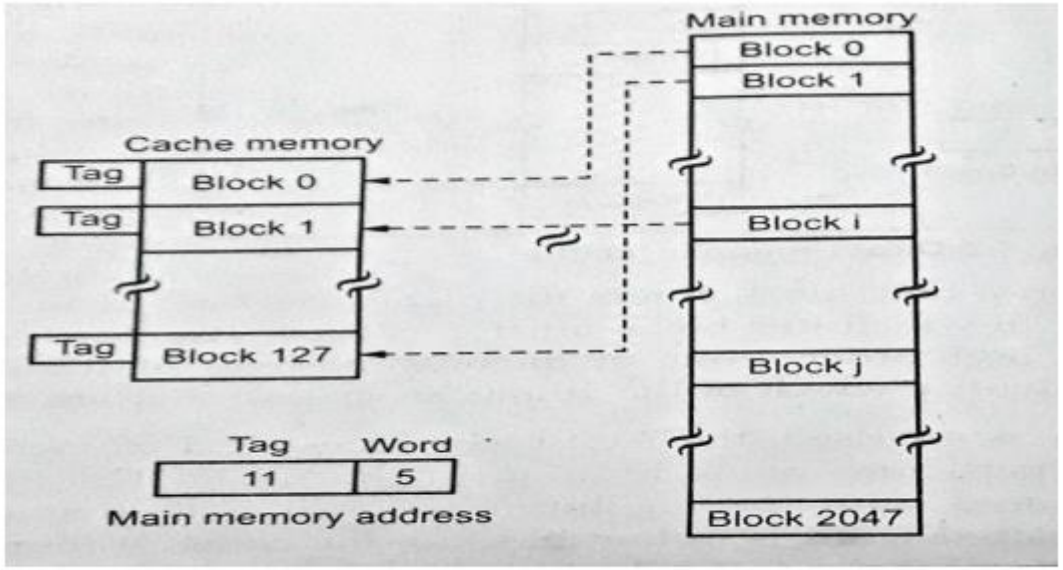


Figure 7.4 Associative-mapped cache

Cache organization

1. **Associative-mapping technique**

- The 11-tag bits are required to identify a memory block when it is resident in the tag bits of each block of the cache to see if the desired block is present.
- Once the desired block is present, the 5-bit word is used to identify the necessary word from the cache.
- This technique gives complete freedom in choosing the cache location in which to place the memory block.
- Thus, the memory space in the cache can be used more efficiently.
- A new block that has to be loaded into the cache has to replaced (remove) an existing block only if the cache is full.
- In such situations, it is necessary to use one of the possible replacement algorithm to select the block to be replaced.
- In associative-mapped cache, it is necessary to compare the higher-order bits of address of the main memory with all 128 tag corresponding to each block to determine whether a given is in the cache.
- This is the main disadvantage of associative-mapped cache.

Cache organization

1. **Set-Associative mapping technique**

- The set-associative mapping is a combination of both direct and associative mapping.
- It contains several groups of direct mapped blocks that operate as several direct mapped caches in parallel.
- A block of data from any page in the main memory can go into a particular block location of any direct-mapped cache.
- Hence the contention problem of the direct-mapped technique is eased by having a few choices for block placement.
- The required address comparisons depend on the number of direct mapped caches in the cache system.
- These comparisons are always less than the comparisons required in the fully-associative mapping.

Cache organization

1. **Set-Associative mapping technique**

- Figure 7.5.shows two way set associative cache.
- Each page in the main memory is organized in such a way that the size of each page is same as the size of one directly mapped cache.
- It is called two-way set associative cache because each block from main memory has two choices for block placement.
- In this technique, block 0, 64,128,…….,1984 of main memory can map into any of the two (block 0) blocks of set 0, block1, 65,129,……., 1985 of main memory can map into any of the two (block 1) blocks of set 1 and so on.

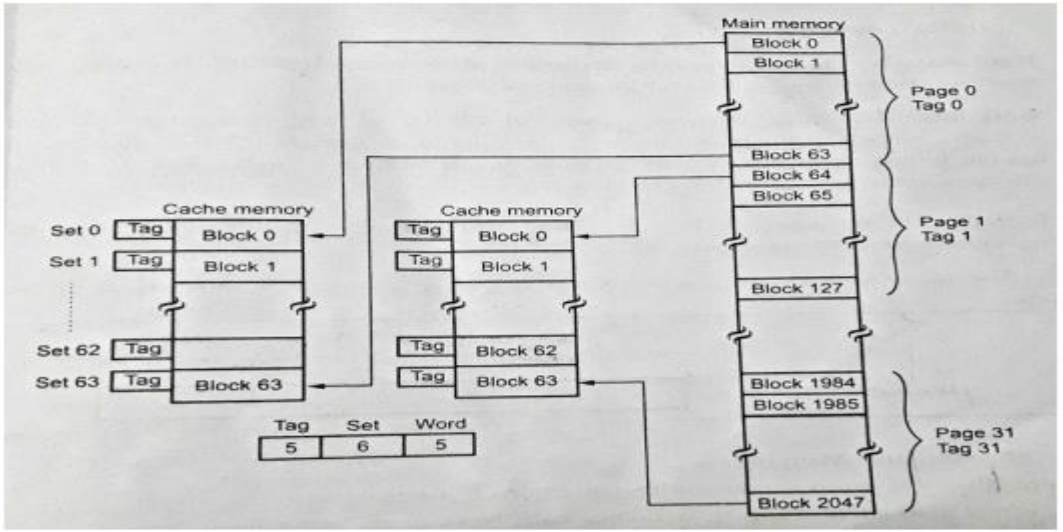


Figure 7.5 Two-way set associative cache

Cache organization

1. **Set-Associative mapping technique**

- As there are two choices, it is necessary to compare address of memory with the tag bits of corresponding two block locations of particular set.
- Thus for two-way set associative cache we required two comparisons to determine whether a given block is in the cache.
- Since from there are two direct mapped caches, any two bytes having same offset from different pages can be in the cache at a time.
- This improves the hit rate of the cache system

Virtual Memory

- In most modern computers, the physical main memory is not as large as the address space spanned by an address issued by the processor.
- Here, the virtual memory technique is used to extend the apparent size of the physical memory.
- It uses secondary storage such as disks, to extend the apparent size of the physical memory.
- When a program does not completely fit into the main memory, it is divided into segments.
- The segments which are currently being executed are kept in the main memory and remaining segments are stored in the secondary storage devices, such as a magnetic disk.
- If an executing program needs a segment which is not currently in the main memory, the required segment is copied from the secondary storage device.
- When new segment of a program is to be copied into a main memory, it must replace another segment already in the memory.

Virtual Memory

- In modern computers, the operating system moves program and data automatically between the main memory and secondary storage.
- Techniques that automatically swaps program and data blocks between main memory and secondary storage device are called virtual memory management.
- The address that processor issues to access either instruction or data are called virtual or logical address.
- These addresses are translated into physical addresses by a combination of hardware and software components.
- If a virtual address refers to a part of the program or data space that is currently in the main memory, then the contents of the appropriate location in the main memory are accessed immediately.
- On the other hand, if the reference address is not in the main memory, its contents must be brought into a suitable location in the main memory before they can be used.

Virtual Memory

- Figure 7.6 Shows a typical memory organization that implements virtual memory.
- *The memory management unit controls this virtual memory system.*
- MMU translates virtual address into physical address assuming that all programs and data are composed of fixed length unit called pages, as shown in the figure.
- Pages constitute the basic unit of information that is moved between the main memory and the disk whenever the page translation mechanism determines that a swapping is required.

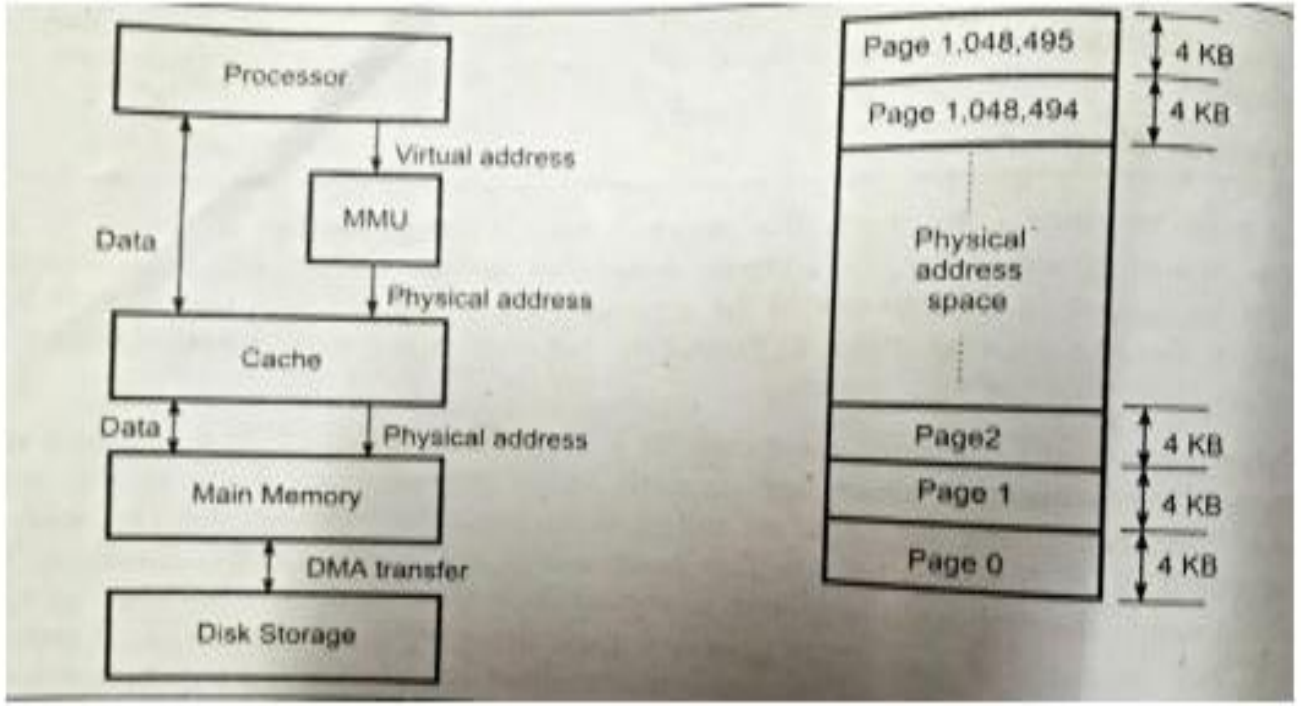


Figure 7.6 Virtual memory organization

Figure 7.7 Paged organization of the physical address space

Address Translation

- Figure 7.8 shows the translation of the virtual page number to a physical page number.
- In virtual memory, the address is divided into two parts, a virtual page number and a page offset.
- The physical page number constitutes the MSBs of the physical address while the page offset, which is not changed, constitutes the LSBs.
- The number of bits in the page offset field decides the page size.

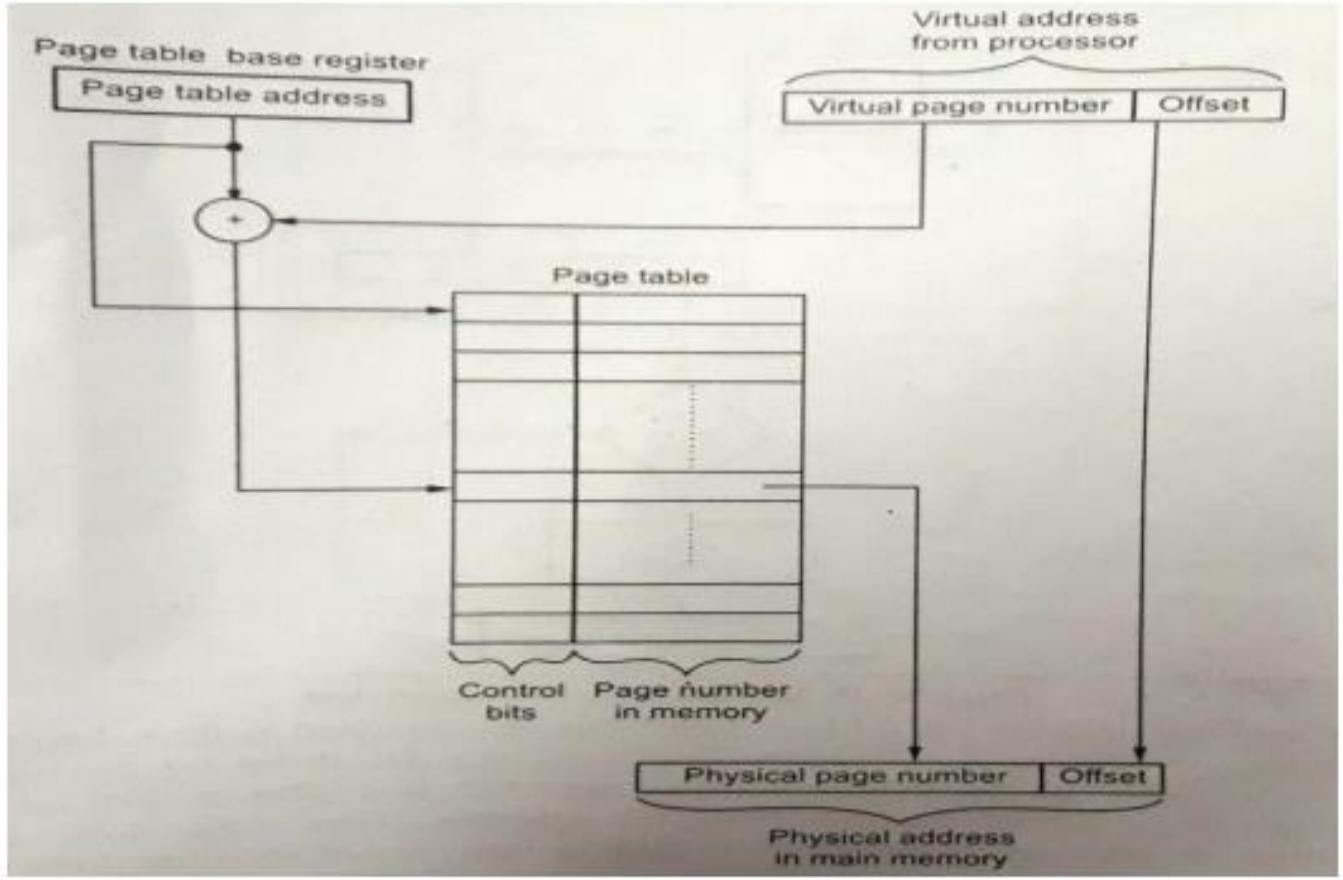


Figure 7.8 Virtual to physical address translation

Address Translation

- The **page table** is used to keep the information about the **location of each page in the main memory.**
- This information includes the **a)** main memory address where the page is stored and **b)** the current status of the page.
- To obtain the address of the corresponding entry in the page table, the virtual page number is added with the contents of page table base register (which stores the starting address of the page table).

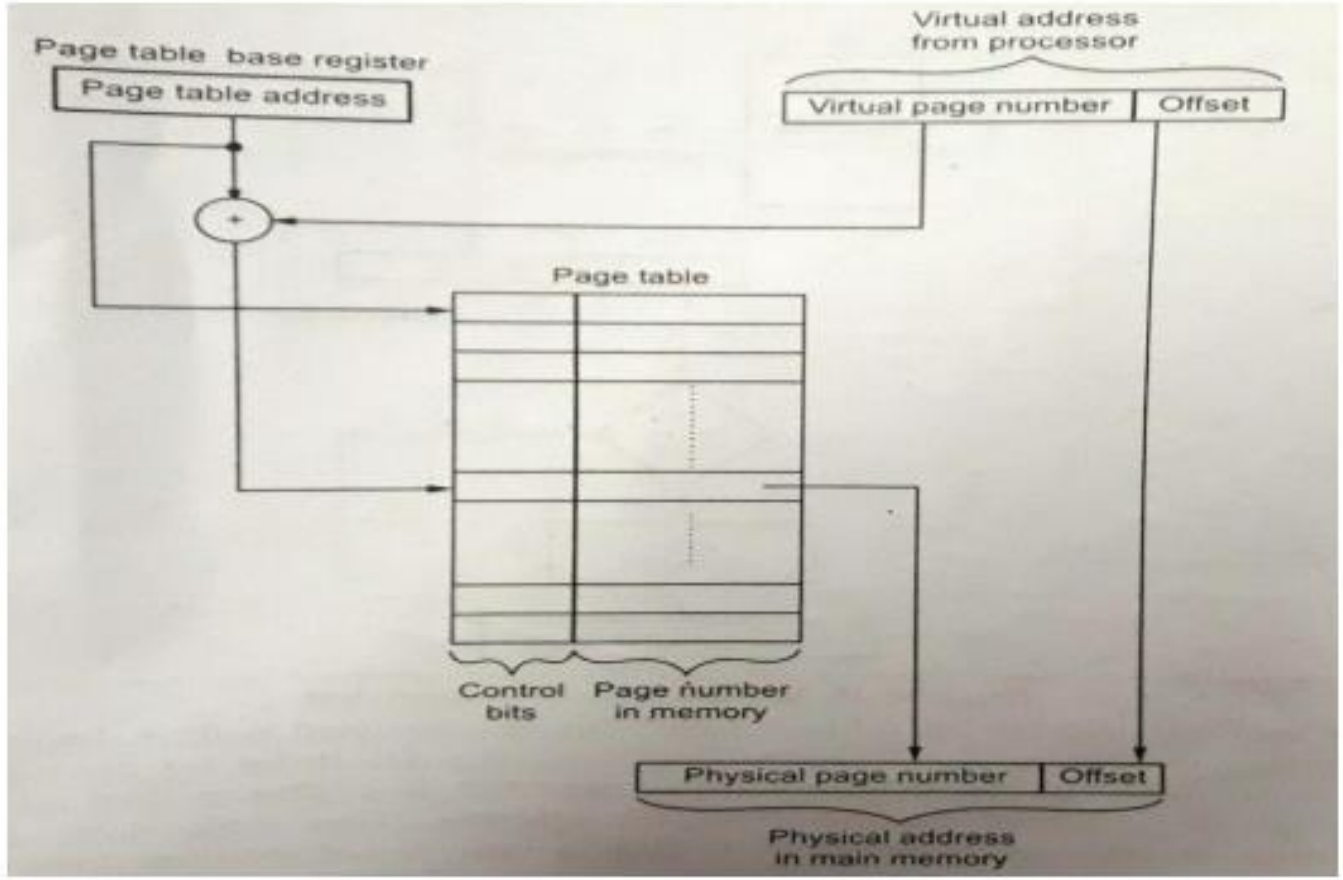


Figure 7.8 Virtual to physical address translation

Address Translation

- The entry in the page table gives the physical page number, in which offset is added to get the physical address of the main memory.
- ***If the page required by the processor is not in the main memory, the page fault occurs and the required page is loaded into the main memory from the secondary storage memory by special routine called page fault routine.***
- ***This technique of getting the desired page in the main memory is called demand paging.***

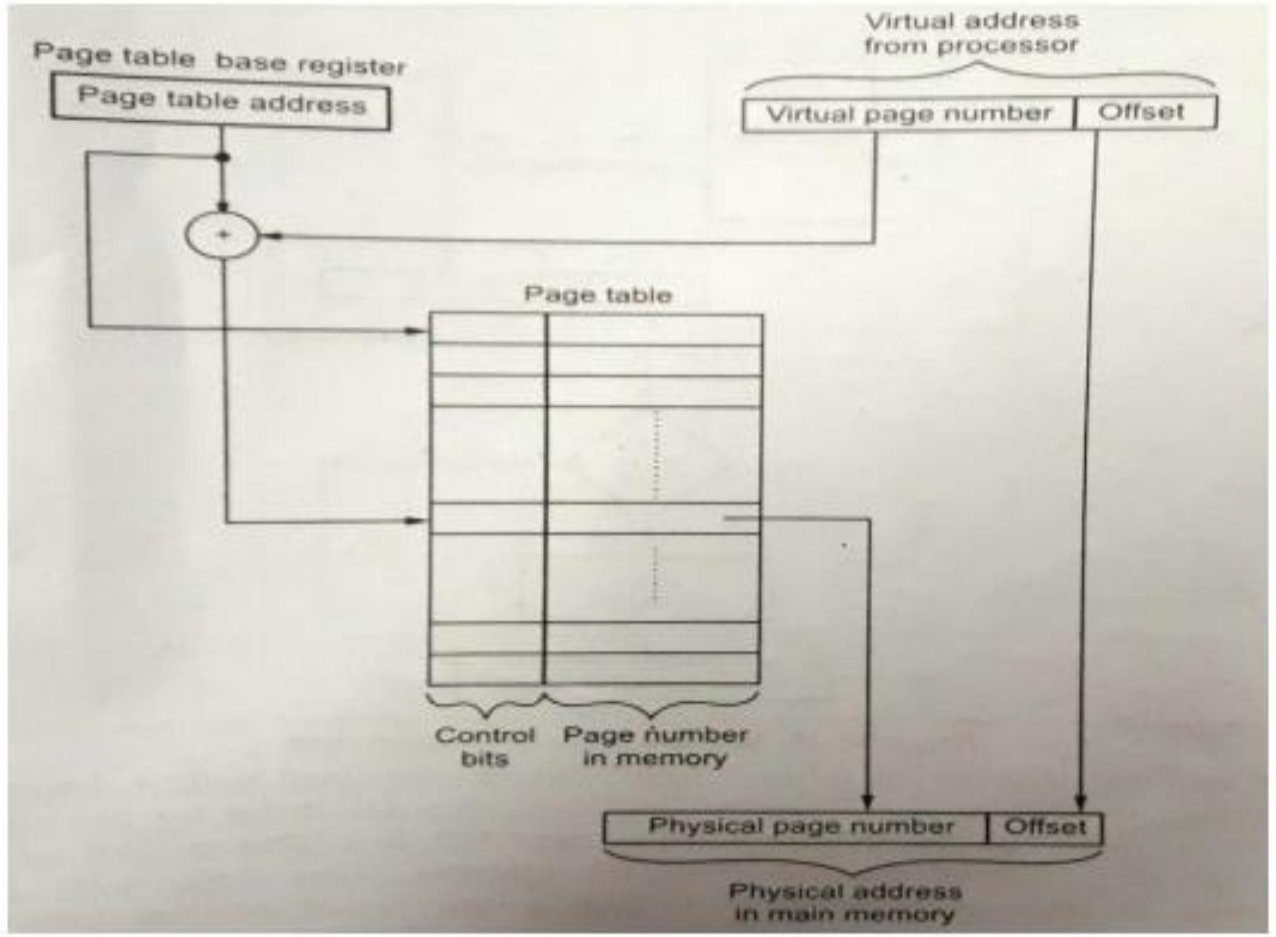
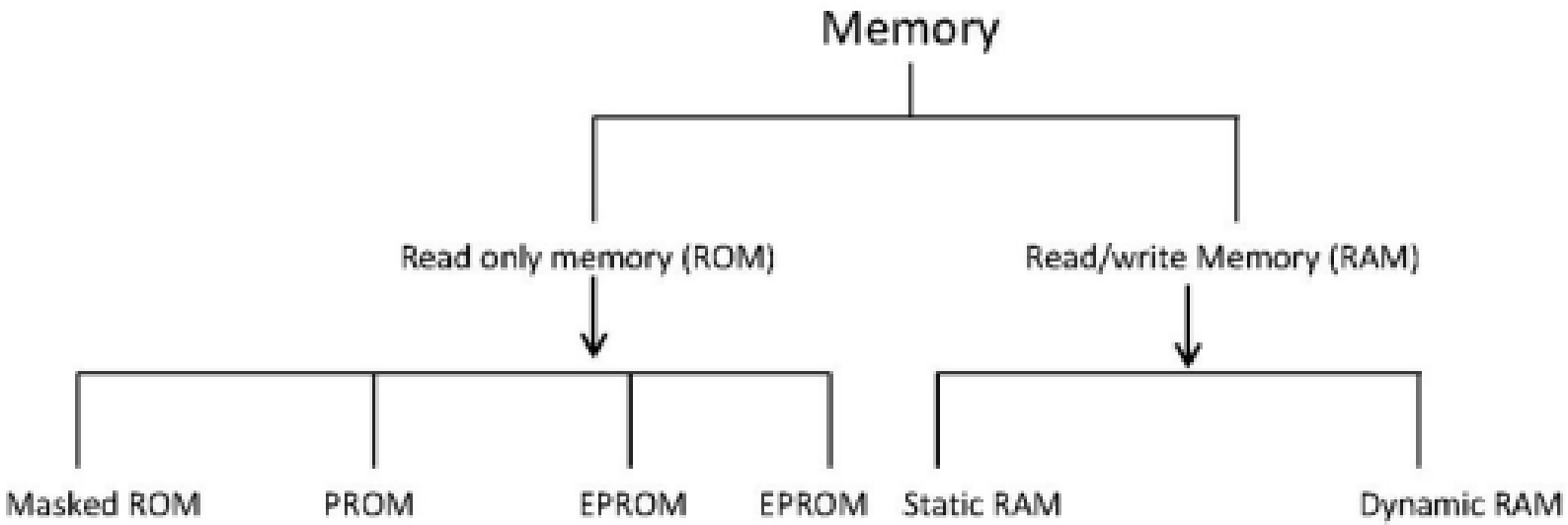


Figure 7.8 Virtual to physical address translation

Semiconductor Main Memory Organization

- Main memory consists of DRAMs supported with SRAM cache.
- These are semiconductor memories.
- The semiconductor memories are classified as shown in fig



DRAM

Dynamic RAM is a form of random access memory. DRAM uses a capacitor to store each bit of data, and the level of charge on each capacitor determines whether that bit is a logical 1 or 0. However these capacitors do not hold their charge indefinitely, and therefore the data needs to be refreshed periodically. As a result of this dynamic refreshing it gains its name of being a dynamic RAM.

DRAM is the form of semiconductor memory that is often used in equipment including personal computers and workstations where it forms the main RAM for the computer. The semiconductor devices are normally available as integrated circuits for use in PCB assembly in the form of surface mount devices or less frequently now as leaded components.

Disadvantages of DRAM

1. Complex manufacturing process
2. Data requires refreshing
3. More complex external circuitry required (read and refresh periodically)
4. Volatile memory
5. Relatively slow operational speed
6. Need to refresh the capacitor charge every once in two milliseconds.

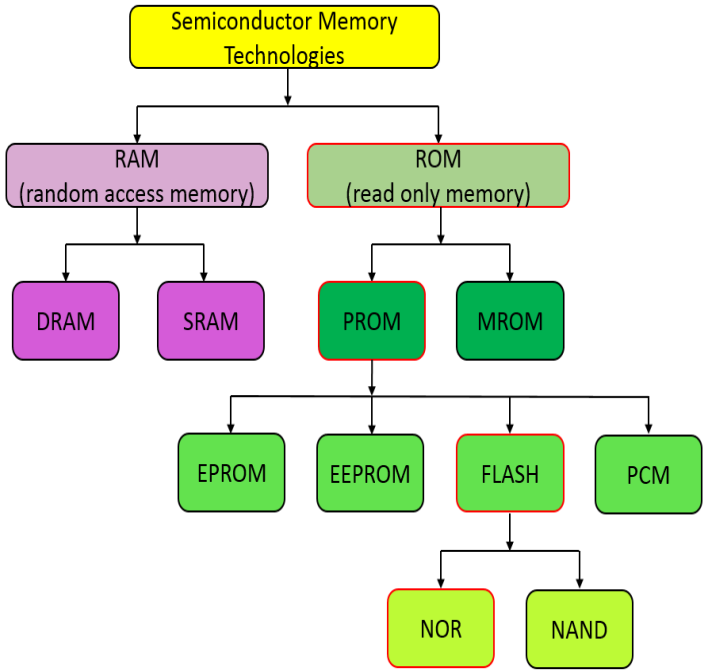
SRAM

SRAM is stands for ***Static Random Access Memory***. This form of semiconductor memory gains its name from the fact that, unlike DRAM, the data does not need to be refreshed dynamically. These semiconductor devices are able to support faster read and write times than DRAM (typically 10 ns against 60 ns for DRAM), and in addition its cycle time is much shorter because it does not need to pause between accesses.

However they consume more power, they are less dense and more expensive than DRAM. As a result of this SRAM is normally used for caches, while DRAM is used as the main semiconductor memory technology.

Flash memory

Flash memory may be considered as a development of EEPROM technology. Data can be written to it and it can be erased, although only in blocks, but data can be read on an individual cell basis. To erase and re-program areas of the chip, programming voltages at levels that are available within electronic equipment are used. It is also non-volatile, and this makes it particularly useful. As a result Flash memory is widely used in many applications including memory cards for digital cameras, mobile phones, computer memory sticks and many other applications.



PCM- Phase change RAM
MROM- Magneto-Resistive ROM

Flash memory stores data in an array of memory cells. The memory cells are made from floating-gate MOSFETS (known as FGMOS). These FG MOSFETs (or FGMOS in short) have the ability to store an electrical charge for extended periods of time (2 to 10 years) even without a connecting to a power supply.

Disadvantages of Flash Memory

1. Higher cost per bit than hard drives
2. Slower than other forms of memory
3. Limited number of write / erase cycles
4. Data must be erased before new data can be written
5. Data typically erased and written in blocks

PCM

This type of semiconductor memory is known as ***Phase change Random Access Memory***, P-RAM or just Phase Change memory, PCM. It is based around a phenomenon where a form of chalcogenide glass changes is state or phase between an amorphous state (high resistance) and a polycrystalline state (low resistance). It is possible to detect the state of an individual cell and hence use this for data storage. Currently this type of memory has not been widely commercialized, but it is expected to be a competitor for flash memory.

Semiconductor memory technology is developing at a fast rate to meet the ever growing needs of the electronics industry. Not only are the existing technologies themselves being developed, but considerable amounts of research are being invested in new types of semiconductor memory technology. In terms of the memory technologies currently in use, SDRAM versions like DDR4 are being further developed to provide DDR5 which will offer significant performance improvements. In time, DDR5 will be developed to provide the next generation of SDRAM.

Semiconductor Main Memory Organization (ROM)

- ROM (Read Only Memory) It is a read only memory.
- We can't write data in this memory.
- It is non-volatile memory i.e. it can hold data even if power is turned off.
- Generally, ROM is used to store the binary codes for the sequence of instructions you want the computer to carry out and data such as look up tables.
- This is because this type of information does not change.
- It is important to note that although we give the name RAM to static and dynamic read/write memory devices that does not mean that the ROMs that we are using are also not random access devices.
- In fact, most ROMs are accessed randomly with unique addresses.
- There are four types of ROM : Masked ROM, PROM, EPROM and EEPROM.

SEMICONDUCTOR MAIN MEMORY ORGANIZATION (ROM)

- The address on the address lines (A0 and A1) is decoded by 2 : 4 decoder.
- Decoder selects one of the four rows making it logic 0 (o/p is active low).
- The inverter connected at the output of decoder inverts the state of selected row (i.e. logic 1).
- Therefore, each output data line goes to logic 0 if a gate of MOS transistor is connected to row select lines.
- When gate of the MOS transistor is connected to the selected row, MOS transistor is turned on.
- This pulls the corresponding column data line to logic 0.

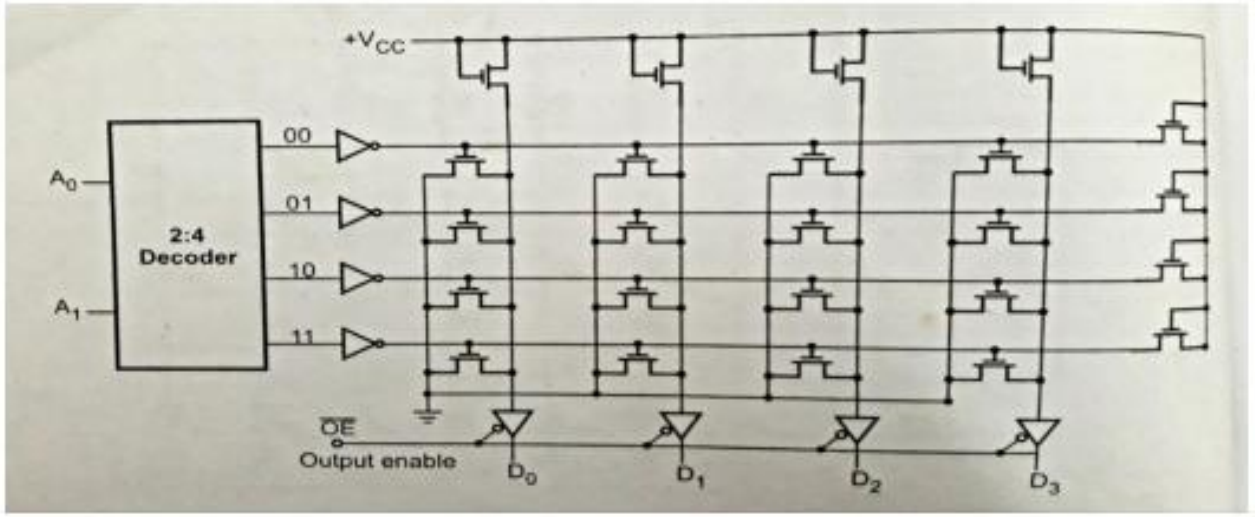


Figure 6.4 Simple four half-byte ROM

SEMICONDUCTOR MAIN MEMORY ORGANIZATION (ROM)

- Figure 6.5 shows four byte PROM.
- It has diodes in every bit position; therefore, the output is initially all 0s.
- Each diode, however has a fusible link in series with it.
- By addressing bit and applying proper current pulse at the corresponding output, we can blow out the fuse, storing logic 1 at the bit. It is necessary to pass around 20 to 50 mA of current for period 5 to 20 us.
- The blowing of fuses according to the truth table is called programming of ROM.
- The user can program PROMs with special PROM programmer.

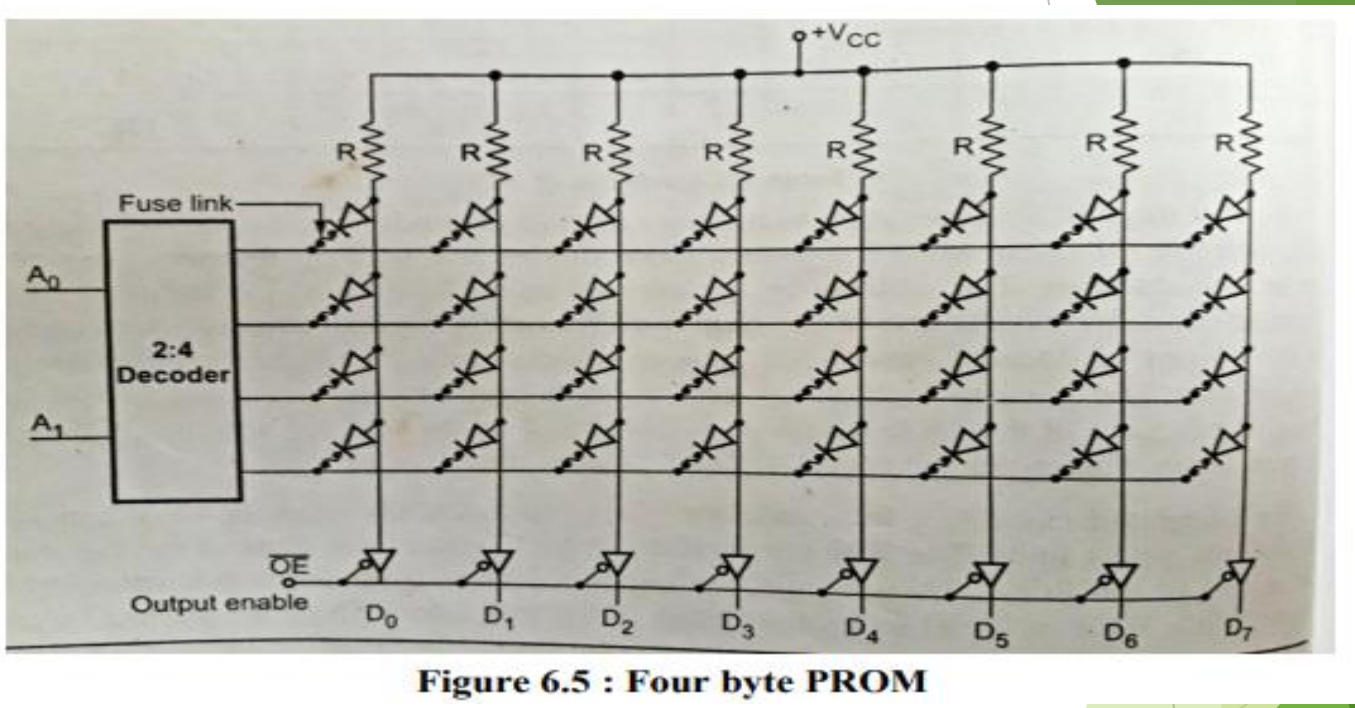


Figure 6.5 : Four byte PROM

Semiconductor Main Memory Organization (ROM)

- Erasable Programmable ROMs use MOS circuitry.
- They store 1s and 0s as a packet of charge in a buried layer of the IC chip. EPROMs can be programmed by the user with a special EPROM programmer.
- The important point is that we can erase the stored data in the EPROMs by exposing the chip to ultraviolet light through its quartz window for 15 to 20 minutes, as shown in the figure 6.6

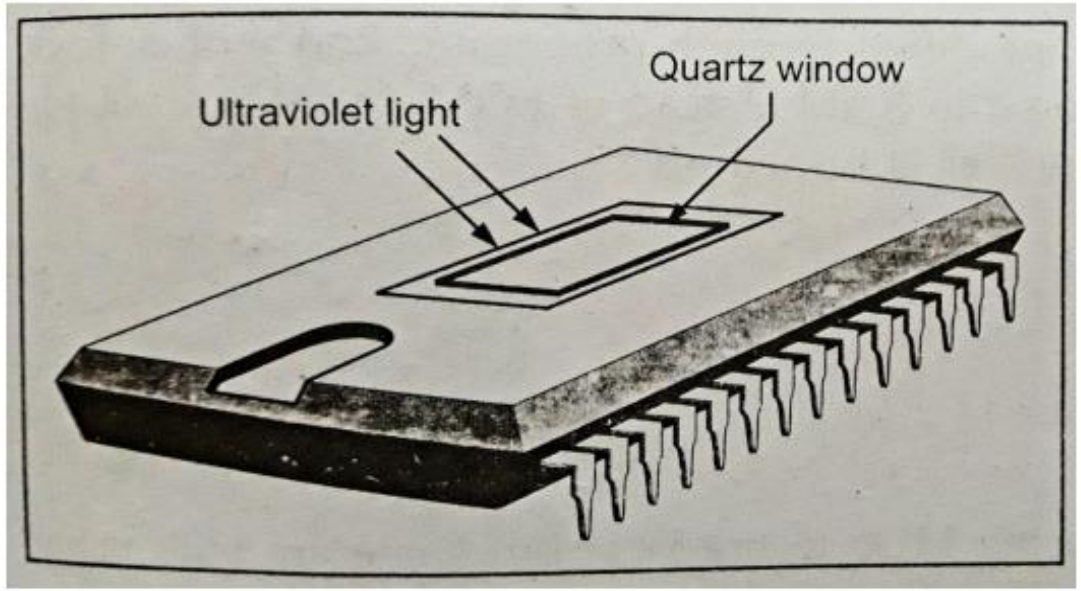


Figure 6.6: EPROM

https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/memory-types-technologies.php

Semiconductor Main Memory Organization (RAM)

- Unlike ROM, we can read from or write into the RAM, so it is often called read/write memory.
- The numerical and character data that are to be processed by the computer change frequently.
- These data must be stored in type of memory from which they can be read by the microprocessor, modified through processing and written back for storage.
- For this reason, they are stored in RAM instead of ROM.
- But it is a volatile memory, i.e. it cannot hold data when power is turned off. There are two types of RAMs:
- Static RAM
- Dynamic RAM

https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/memory-types-technologies.php

Semiconductor Main Memory Organization (RAM)

Static RAM

- Most of the static RAMs are built using MOS technology, but some are built using bipolar technology
- TTL RAM CELL Figure 6.7 shows a simplified schematic of a bipolar memory cell.
- The memory cell is implemented using TLL (Transistor – Transistor-Logic) multiple emitter technology.
- It stores 1 bit of information.
- It is nothing but a flip-flop.
- It can store either 0 or 1 as long as power is applied, and it can set or reset to store either 1 or 0, respectively.

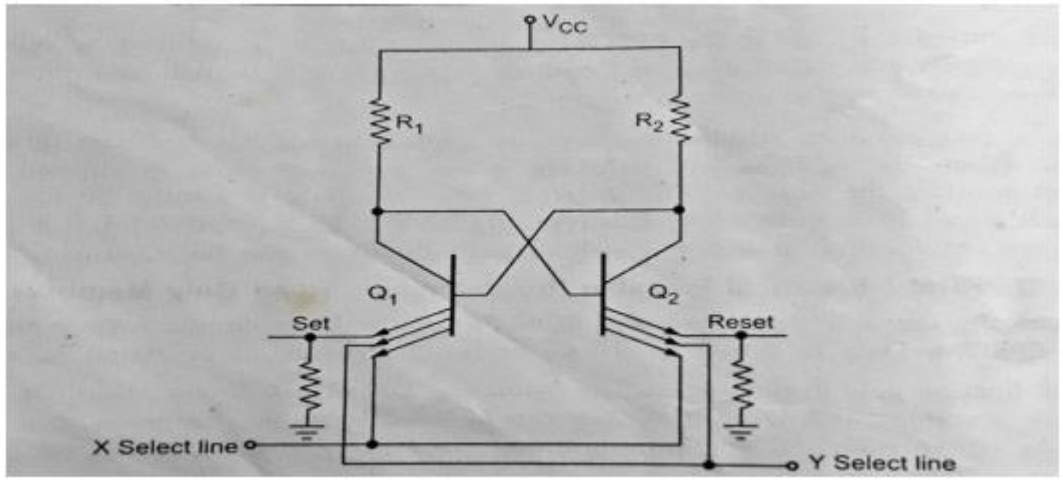


Figure 6.7 TTL RAM CELL

Semiconductor Main Memory Organization (RAM)

Operation:

- The X select and Y select input lines select a cell from matrix.
- The Q1 and Q2 are cross coupled inverters, hence one is always OFF while the other is ON. A "1" is stored in the cell if Q1 is conducting and Q2 is OFF.
- A "0" is stored in the cell if Q2 is conducting and Q1 is OFF.
- The state of the cell is changed to a "0" by pulsing a HIGH on the Q1 (SET) emitter.
- This turns OFFQ1.
- When Q1 is turned OFF, Q2is turned ON. As long as Q2 is ON, its collector is LOW and Q1 is held OFF.

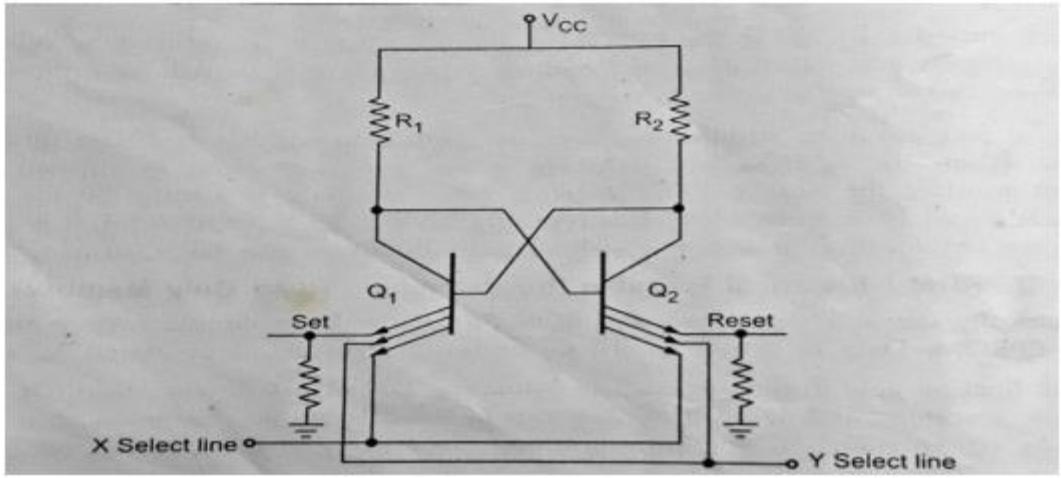


Figure 6.7 TTL RAM CELL

Semiconductor Main Memory Organization (RAM)

MOS Static RAM CELL

- Figure 6.9 shows a simplified schematic of MOS static RAM cell.
- Enhancement mode MOSFET transistors are used to make this RAM cell.
- It is very similar to TTL cell discussed earlier.

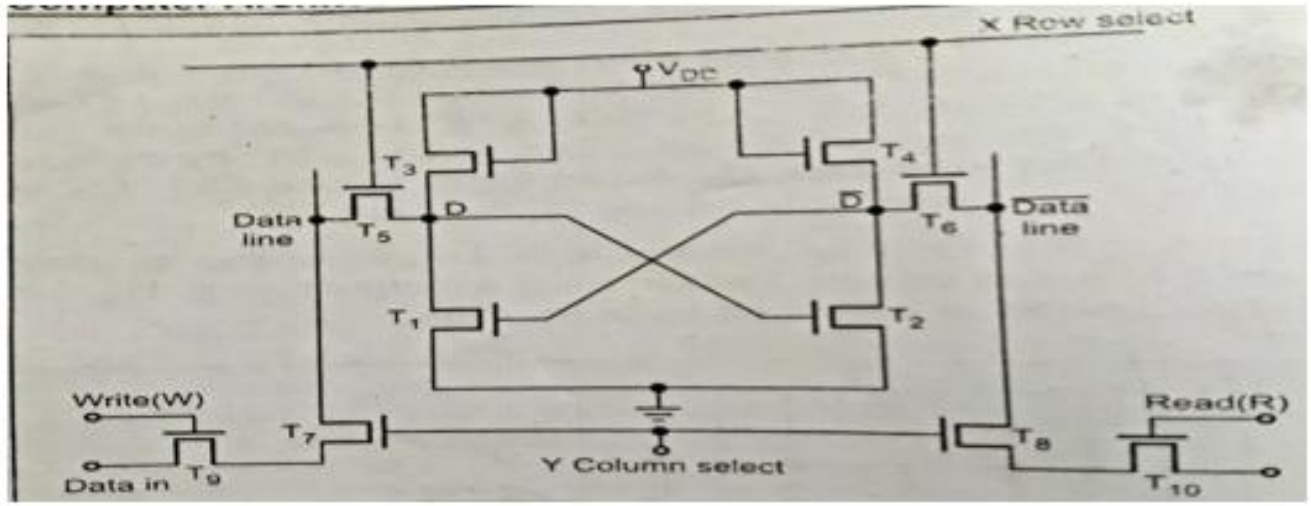


Figure 6.9 MOS static RAM cell

Semiconductor Main Memory Organization (RAM)

- Here, T1 and T2 from the basic cross coupled inverters and T3 and T4 act as load resistors for T1 and T2 .
- X and Y lines are used for addressing the cell. When X and Y both are high, cell is selected.
- When X=1, T5 and T6 get ON and the cell is connected to the data and data line. When Y=1, T7 and T8 are made ON. Due to this, either read or write operation is possible.
- Write Operation : Write operation can be enabled by making W signal high. With write operation enable, if data-in signal is logic 1, node D is also at logic 1. This turns ON T2 and T1 is cutoff. If new data on data-in pin is logic 0, T2 will be cutoff and T1 will be turned ON.
- Read Operation : Read Operation can be enabled by mistake R signal high. With read operation enabled, T10 becomes ON. This connects the data output (Data) line to the data out and thus the complement of the bit stored in the cell is available at the output.

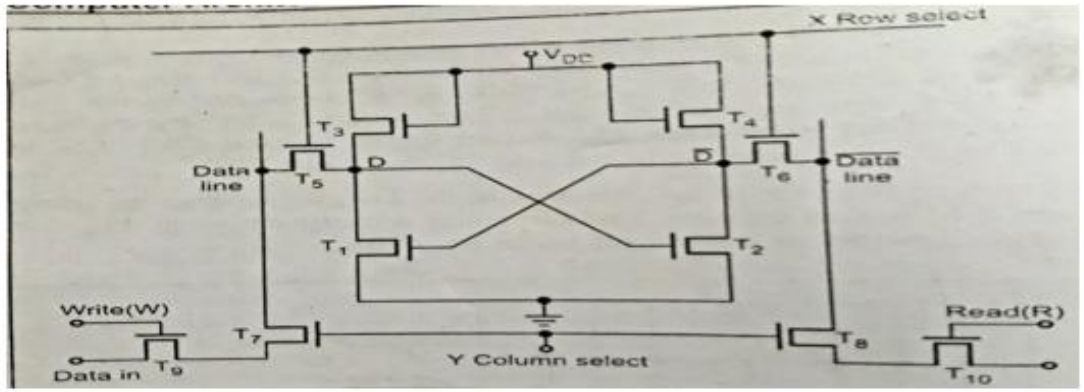


Figure 6.9 MOS static RAM cell

Semiconductor Main Memory Organization (RAM)

- Dynamic RAM stores the data as a charge on the capacitor.
- Figure 6.11 shows the dynamic RAM cell.
- A dynamic RAM contains thousands of such memory cells.
- When COLUMN (Sence) and ROW (Control) lines go high, the MOSFET conducts and charges the capacitor.
- When the COLUMN and ROW lines go low, the MOSFET opens and the capacitor retains its charge.
- In this way, it stores 1 bit.
- Since only a single MOSFET and capacitor are needed, the dynamic RAM contains more memory cells as compared to static RAM per unit area.
- The disadvantage of dynamic RAM is that it need refreshing of charge on the capacitor after every few milliseconds.
- This complicates the system design, since it requires the extra hardware to control refreshing of dynamic RAMs. In this type of cell, the transistor acts as a switch

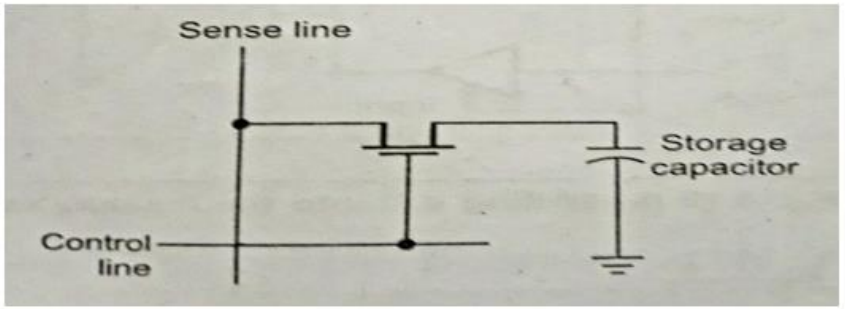


Figure 6.11 Dynamic RAM

Semiconductor Main Memory Organization (RAM)

Static RAM		Dynamic RAM
1	Static RAM contains less memory cells per unit area.	Dynamic RAM contains more memory cells as compared to static RAM per unit area.
2	It less access time hence faster memories	Its access time is greater than static RAMs.
3	Static RAM consists of number of flip-flops. Each flip-flop stores one bit.	Dynamic RAM stores the data as a charge on the capacitor. It consists of MOSFET and the capacitor for each cell.
4	Refreshing circuitry is not required.	Refreshing circuitry is required to maintain the charge on the capacitors after every few milliseconds. Extra hardware is required to control refreshing. This makes system design complicated.
5	Cost is more.	Cost is less.

Comparison Between SRAM and DRAM

Memory Chip Organization

- Large numbers of these cells are organized on a row and column basis chip to form a memory chip.
- Figure 6.15(a) shows the row and column organization for 6264 a high performance CMOS static RAM (SRAM).
- The chip has 13 address lines.
- The eight address lines are connected to the row decoder to indicate one of the 256 rows, and remaining five address lines are connected to the column decoder to indicate one of 32 columns.
- Where the decoded row and column cross, they select the desired individual memory cell. Simple, arithmetic shows that there are 256* 32,8192 crossings.
- The entire memory has eight such arrays.
- Therefore this memory has 8192* 8 memory cells.

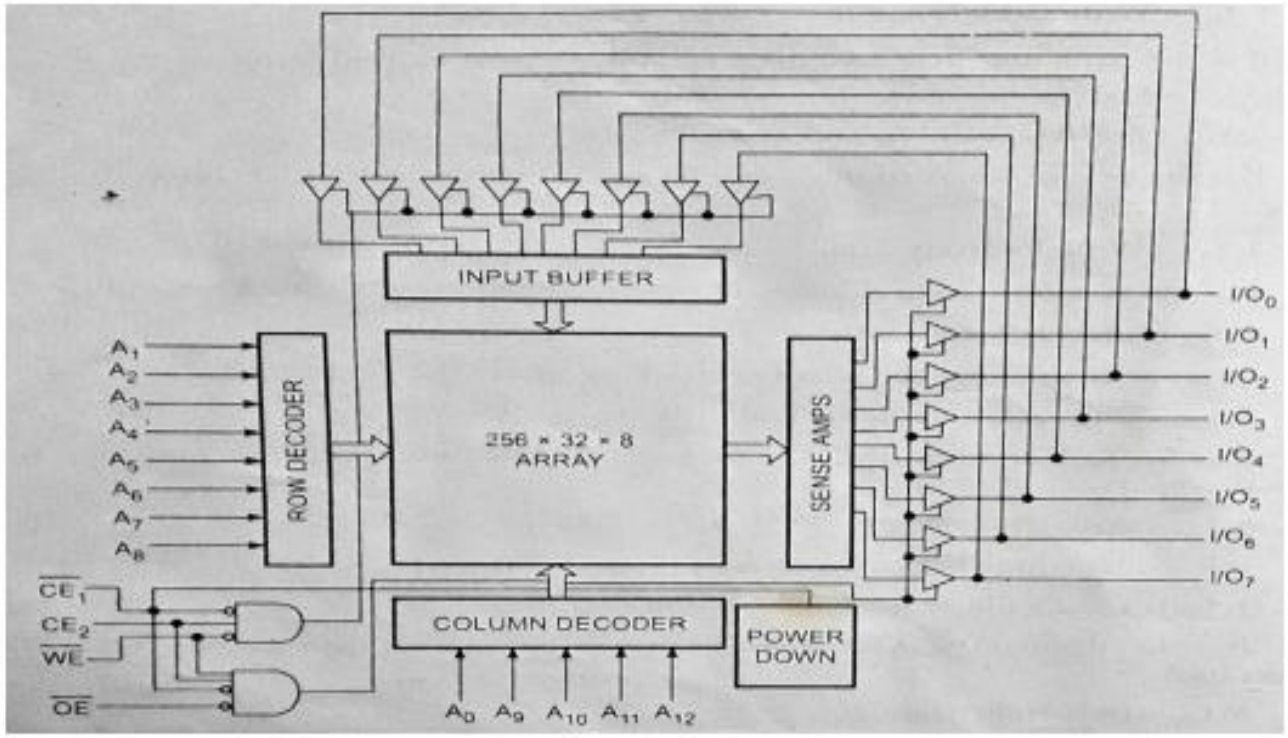
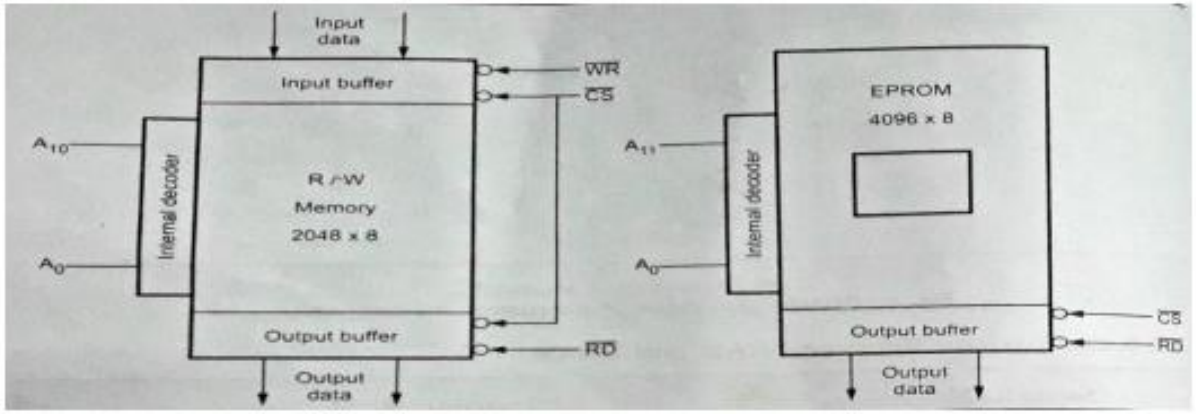


Figure 6.15(a) shows the row and column organization for 6264

Memory structure and its Requirements

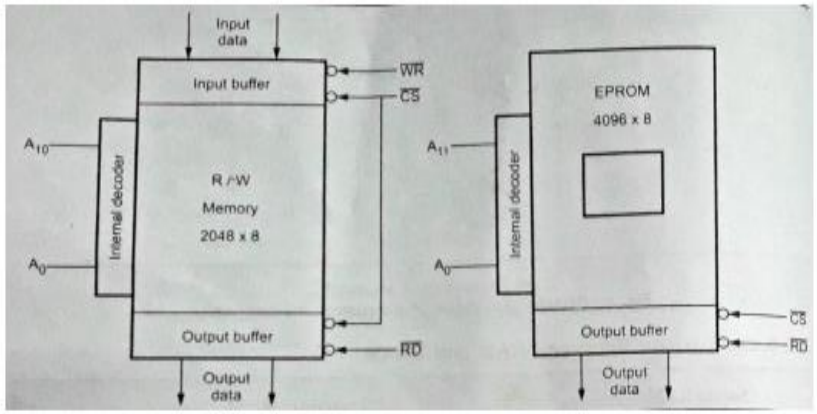
- Read/Write memories consists of an array of registers, in which each register has unique address.
- The size of the memory is N*M as shown in figure 6..13(a) where N is the number of registers and M is the word length, in number of bits.
- Example 1:
- If memory is having 12 address lines and 8 data lines, then Number of registers/memory locations= 2^N = 2^12 = 4096 Word length = M bit = 8 bit



6.13 (a) Logic diagram for RAM (b) Logic diagram for EPROM

Memory structure and its Requirements

- Example 2 : If memory has 8192 memory locations, then it has 13 address lines.
- The table 6.4 summarizes the memory capacity and address lines required for memory interfacing.
- Figure 6.13 (b) shows the logic diagram of a typical EPROM (Erasable Programmable Read-Only Memory) with 4096 (4K) registers.
- It has 12 address lines A0-A11, one chip select (CS), one Read control signal.
- Since EPROM is a read only memory, it does not require the (WR) signal.



6.13 (a) Logic diagram for RAM (b) Logic diagram for EPROM

Table 6.4 : Memory capacity and address lines required

Memory capacity	Address Lines Required
1K = 1024 memory locations	10
2K = 2048 memory locations	11
4K=4096 memory locations	12
8K = 8192 memory locations	13
16K = 16384 memory locations	14
32K = 32768 memory locations	15
64K = 65536 memory locations	16

Associative Memory

- Many data-processing applications require the search of items in a table stored in memory.
- They use object names or number to identify the location of the named or numbered object within a memory space.
- For example, an account number may be searched in a file to determine the holder's name and account status.
- To search an object, the number of accesses to memory depends on the location of the object and the efficiency of the search algorithm.
- The time required to find an object stored in memory can be reduced considerably if objects are selected based on their contents, not on their locations.
- ***A memory unit accessed by the content is called an associative memory or content addressable memory (CAM).***
- This type of memory is accessed simultaneously and in parallel on the basis of data content rather than by specific address or location.

Associative Memory

- The figure 6.18 shows the block diagram of an associative memory. It consists of memory array with match logic for m, n-bit words and associated registers.
- The argument register (A) and key register (K) each have n-bits per word.
- Each word in memory is compared in parallel with the contents of the argument register.

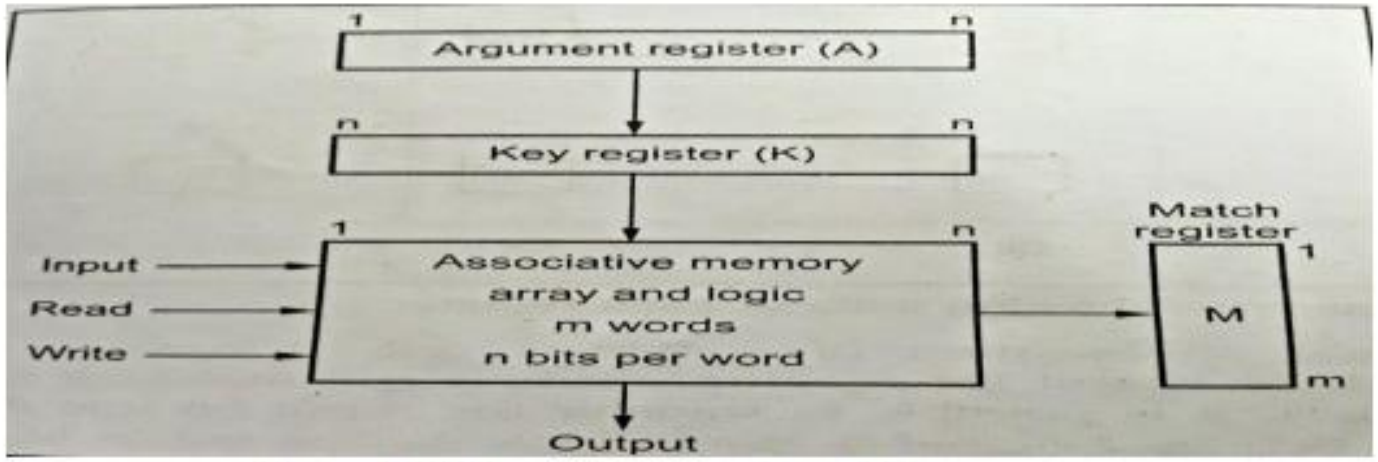


Figure 6.18 Block diagram of associative memory

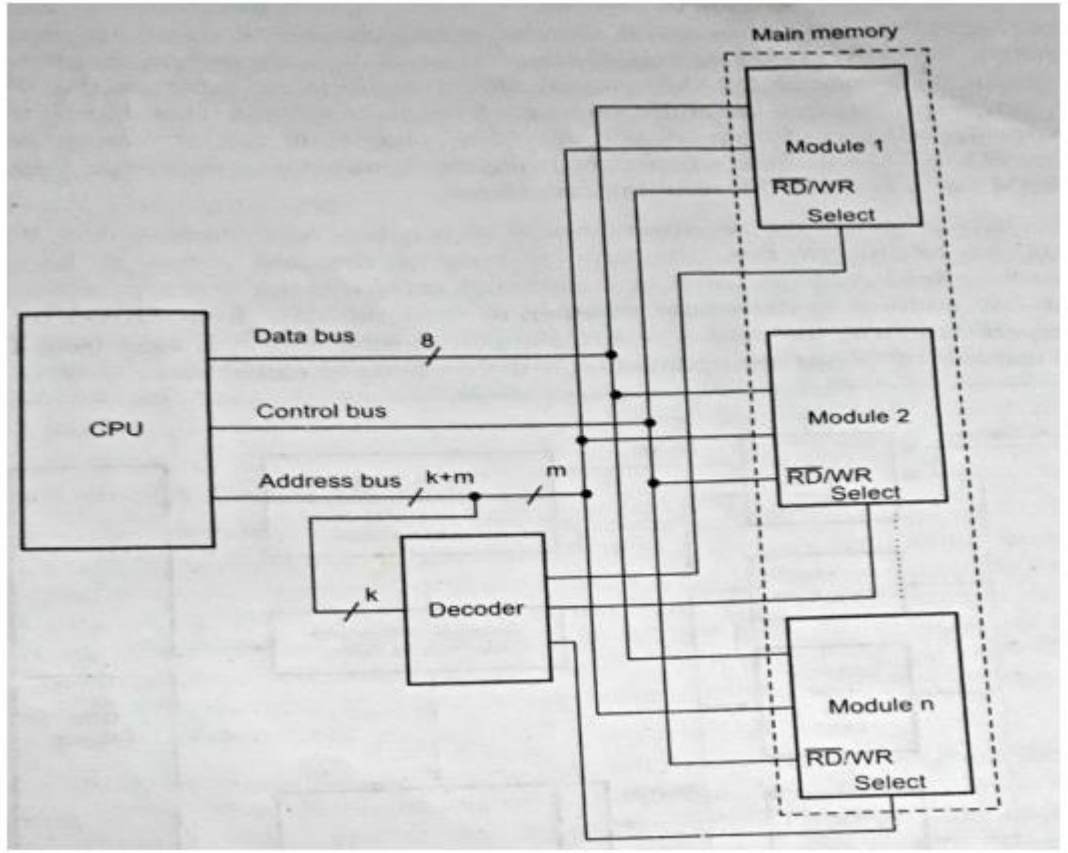
Associative Memory

- The words that match with the word stored in the argument register, set corresponding bits in the match register.
- Therefore, reading can be accomplished by a sequential access to memory for those words whose corresponding bits in the match register have been set.
- The key register provides a mask for choosing a particular field or bits in the argument word.
- Only those bits in the argument register having 1's in their corresponding position of the key register are compared.
- For example, if argument register A and the key register K have the bit configuration shown below.
- Only the three rightmost bits of A are compared with memory words because K has 1's in these positions.

A	11011	O1O
K	00000	111
Word 1	01010	010 match
Word 2	11011	100 no match

Associative Memory

- The memory access time is the bottleneck in the system and one way to reduce it is to use cache memory.
- Alternative technique to reduce memory access time is memory interleaving.
- In this technique, to reduce memory access time, it is divided into a number of memory modules and the address are arranged such that the successive words in the address space are placed in different modules.



Associative Memory

- Most of the times CPU accesses consecutive memory locations.
- In such situations address will be to the different modules.
- Since these modules can be accessed in parallel, the average access time of fetching word from the main memory can be reduced.
- ***The low-order k bits of the memory address are generally used to select a module, and the high-order m bits are used to access a particular location within the selected module.***
- In this way, consecutive address is located in successive modules.
- Thus, any component of the system that generates requests for access to consecutive memory system as a whole.

Associative Memory

- It is important to note that to implement the interleaved memory structure, there must be 2 modules; otherwise, there will be gaps of nonsexist locations in the memory address space.
- The effect of interleaving is substantial. However, it does not speed up memory operation by a factor equal to the number of the module.
- It reduces the effective memory cycle time by a factor close to the number of modules.
- Pipeline and vector processors often require simultaneous access to memory from two or more source.
- This need can be satisfied by interleaved memory.
- A CPU with instruction pipeline can also take the advantage of interleaved memory modules so that each segment in the pipeline can access memory independent of memory access from other segments.