Computer Organisation

MODULE –I

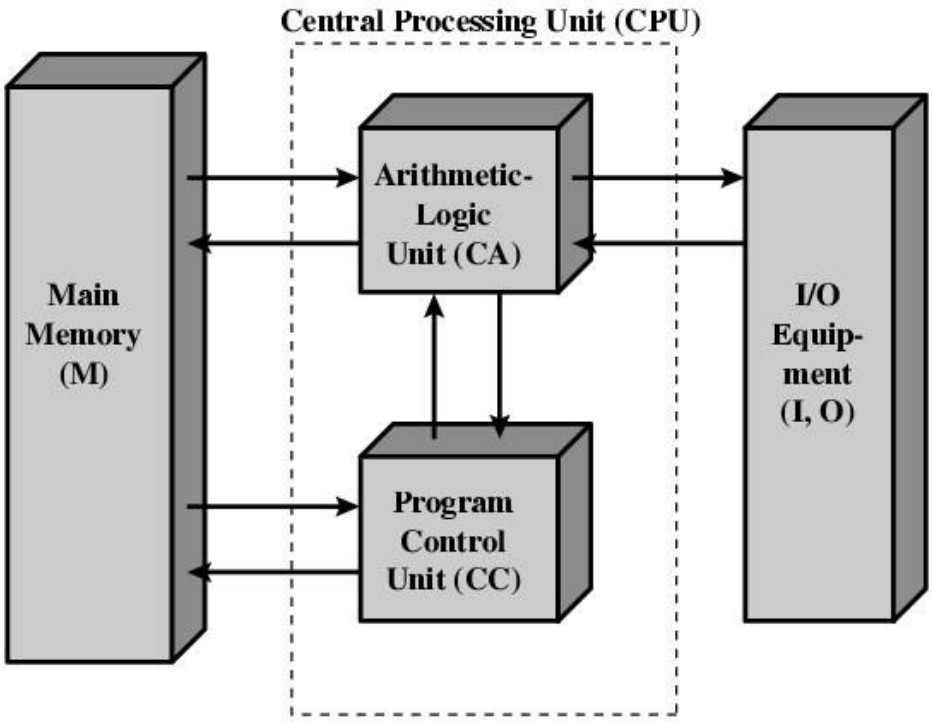
CO1 Illustrate the basic operational concepts and memory of a computer system

Architecture & Organization

- Architecture is those attributes visible to the programmer
  - —Instruction set, number of bits used for data representation, I/O mechanisms, addressing techniques.
  - —e.g. Is there a multiply instruction?
- Organization is how features are implemented
  - —Control signals, interfaces, memory technology.
  - —e.g. Is there a hardware multiply unit or is it done by repeated addition?

Functional units

Structure of von Neumann machine



A computer consists of the following functionally independent main parts:

a. Input unit
b. Memory
c. Central Processing Unit(CPU) that contains Arithmetic and Logic Unit, Control Unit and Registers
d. Output unit

Input unit -The **input unit** accepts coded information from human operators using devices such as keyboards, or from other computers over digital communication lines.

Memory unit - Memory stores programs and data.

There are two classes of storage, called primary and secondary.

Primary Memory : Primary memory, or main memory, is a fast memory that operates at electronic speeds. Programs must be stored in this memory while they are being executed. Instructions and data can be written into or read from the memory under the control of the processor

Cache Memory: Cache is a smaller, faster memory unit that hold sections of a program that are currently being executed, along with any associated data.

Secondary Storage :This is less expensive, permanent secondary storage that is used when large amounts of data and many programs have to be stored, particularly for information that is accessed infrequently.

Arithmetic and Logic Unit(ALU) – It carries out arithmetic and logical operations

Output Unit- Its function is to send processed results to the outside world. Eg: Printer

Control Unit- It controls and coordinates the activities of all the other units with the help of control signals.

Basic operational concepts

The activity in a computer is governed by ***instructions*.** To perform a given task, an appropriate program consisting of a ***list of instructions*** is stored in the memory. Individual instructions are brought from the *memory into the processor*, which executes the specified operations. Data to be used as instruction *operands* are also stored in the memory.

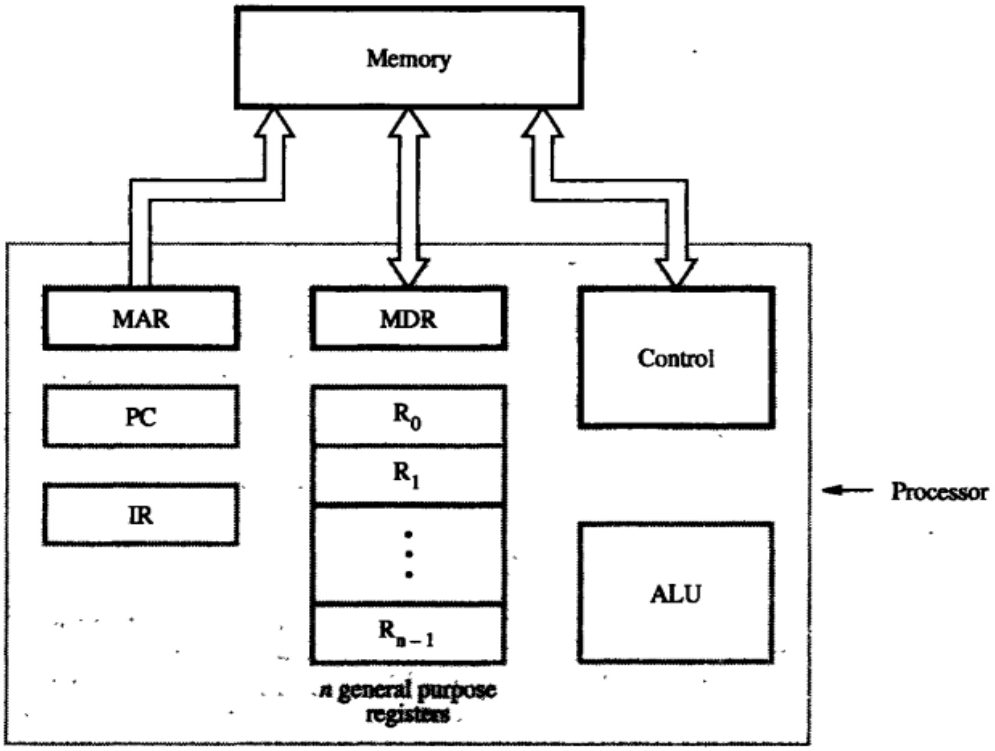
A typical instruction might be

Load R2, LOC

This instruction reads the contents of a memory location whose address is represented symbolically by the label LOC and loads them into processor register R2. The original contents of location LOC are preserved, whereas those of register R2 are overwritten.

Execution of this instruction requires **several steps**. First, the instruction is fetched from the memory into the processor. Next, the operation to be performed is determined by the control unit. The operand at LOC is then fetched from the memory into the processor. Finally, the operand is stored in register R2.

Connection of the memory to the processor



- The CPU exchanges data with memory.
- For this purpose, it uses of two internal registers
  - —A memory address register (MAR), which specifies the address in memory for the next read or write
  - —A memory buffer register (MBR), which contains the data to be written into memory or receives the data read from memory.
- Program counter (PC) holds the address of the instruction to be fetched next
- The fetched instruction is loaded into instruction register (IR). An *instruction register* (IR) holds the *instruction* currently being executed.

- A memory module consists of a set of locations, having address in a sequential manner. Each location contains a binary number that can be an instruction or data.
- The n general purpose registers, R_0 through R_{n-1} used for temporary storage of data

Basic operational steps

1. Instrcution Fetch

- Execution of the program starts when PC is set to point to the first instruction of the program.
- The content of PC is transferred to MAR and Read control signal is send to memory
- The content of the memory location is loaded on to MDR via data bus.
- The content of MDR(instruction) is transferred to IR.

2. Instruction Decode – meaning of the instruction is identified

3. Data Fetch/ Address Calculation – The data can be either in a general purpose register or in the memory. If it is at a memory location, Read operation is needed to get the data from the memory to MDR.

4. Instruction Execution – The desired operation is carried out**.**

5. Write Back – The result is stored into destination (can be either processor register or memory location)

BUS INTERCONNECTION

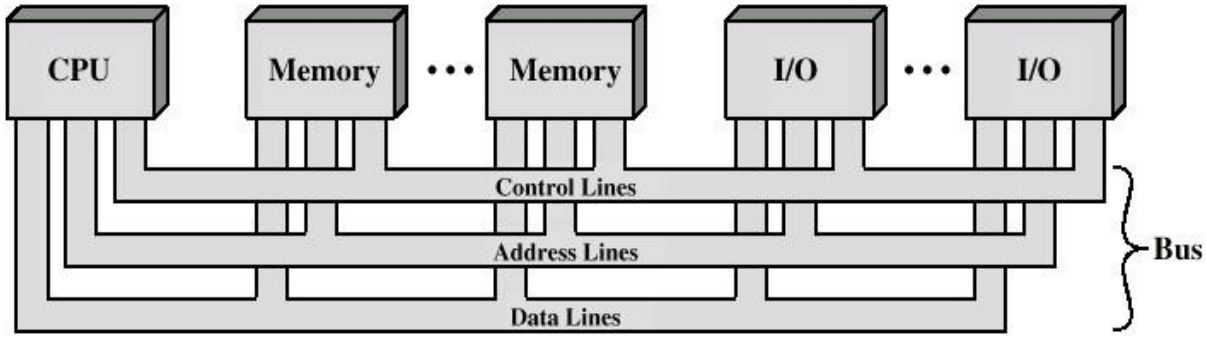
- A bus is a communication pathway connecting two or more devices.
- A bus is a shared transmission medium.
- Typically, a bus consists of multiple communication pathways, or lines.
- Each line is capable of transmitting signals representing binary 1 and binary 0.
- Computer systems contain a number of different buses that provide pathways between components at various levels of the computer system hierarchy.
- A bus that connects major computer components (processor, memory, I/O) is called a **system bus**.

Bus Structure

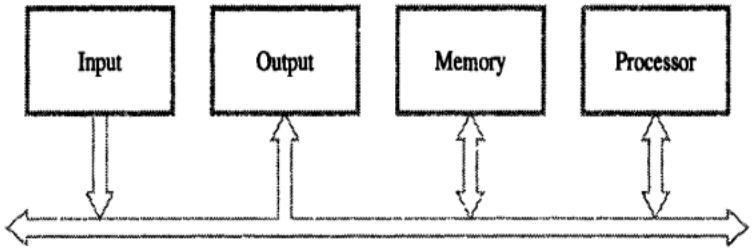
- Bus designs can be classified into three functional groups **data, address, and control** lines.

- **Data Bus**
  - —The data lines provide a path for moving data (data & instructions) among system modules.
  - —The data bus may consist of 32, 64, 128, or even more separate lines
  - —The number of lines is referred to as the width of the data bus.
  - —The number of lines determines how many bits can be transferred at a time.
  - —The width of the data bus is a key factor in determining overall system performance
- **Address bus**
  - The address lines are used to identify the source or destination of the data on the data bus.
  - The width of the address bus determines the maximum possible memory capacity of the system.
  - The address lines are generally also used to address I/O ports.
- **Control Bus**
  - The control lines are used to control the access to and the use of the data and address lines.
  - Control signals transmit both command and timing information among system modules.
  - Timing signals indicate the validity of data and address information.
  - Command signals specify operations to be performed.
  - Typical control lines include
    - —Memory read/write signal
    - —Interrupt request
    - —Clock signals

Bus Interconnection Scheme



The simplest way to interconnect functional units is to use a *single bu.* All units are connected to this bus. Because the bus can be used for only one transfer at a time, only two units can actively use the bus at any given time.

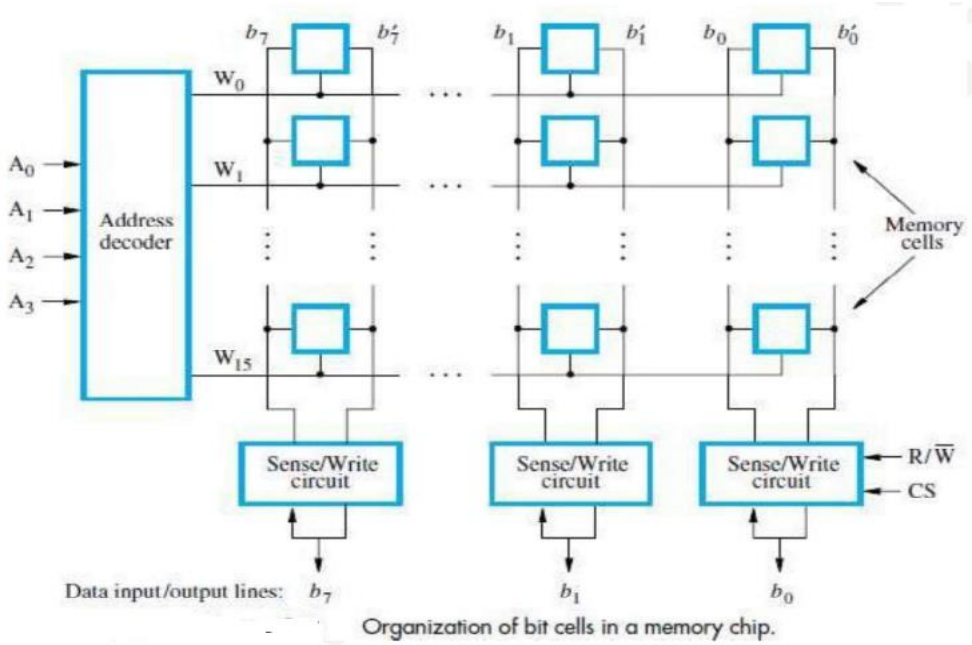


Single-bus structure.

- **Single Bus Problems**
  - —Lots of devices on one bus leads to delay
  - —Most systems use multiple buses to overcome these problems

INTERNAL ORGANIZATION OF MEMORY CHIPS

Memory cells are usually .organized in the form of an array, in which each cell is capable of storing one bit of information.



Organization of bit cells in a memory chip.

Each row of cells constitutes a memory word, and all cells of a row are connected to a common line referred to as the *word line,* which is driven by the address decoder on the chip. The cells in each column are connected to a Sense/Write circuit by two *bit lines.* The Sense/Write circuits are connected to the data input/output lines of the chip.

The data input and the data output of each Sense/Write circuit are connected to a single bidirectional data line that can be connected to the data bus of a computer. Two control

lines, R/W[1] and CS, are provided in addition to address and data lines. The R/W[1] (Read/Write) input specifies the required operation, and the CS(Chip Select) input selects a given chip in a multichip memory system.

Read Operation

- Depending on the address given by the processor, the address decoder activates the appropriate word line.
- The Read/Write[1] control signal is 1.
- The Sense/Write circuit transfers the contents of all the selected cells are to the bit line and to the output data line

Write Operation

- Depending on the address given by the processor, the address decoder activates the appropriate word line.
- The Read/Write[1] control signal is 0.
- The Sense/Write circuit transfers the content of data input line to the bit line and then to the cells.

STATIC RAM (OR MEMORY)

• Memories consist of circuits capable of retaining their state as long as power is applied are known.

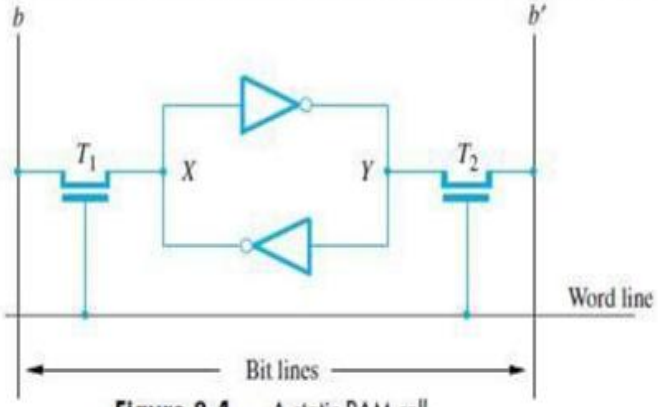


Figure 8.4 A static RAM cell.

• Two inverters are cross connected to form a latch (Figure 8.4).
• The latch is connected to 2-bit-lines by transistors T_1 and T_2.
• The transistors act as switches that can be opened/closed under the control of the word-line.
• When the word-line is at ground level, the transistors are turned off and the latch retain its state.

Read Operation

• To read the state of the cell, the word-line is activated to close switches T_1 and T_2.
• If the cell is in state 1, the signal on bit-line b is high and the signal on the bit-line b" is low.
• Thus, b and b" are complement of each other.
• Sense/Write circuit
  → monitors the state of b & b" and
  → sets the output accordingly.

Write Operation

- The state of the cell is set by
  - → placing the appropriate value on bit-line b and its complement on b" and
  - → then activating the word-line. This forces the cell into the corresponding state.
- The required signal on the bit-lines is generated by Sense/Write circuit.

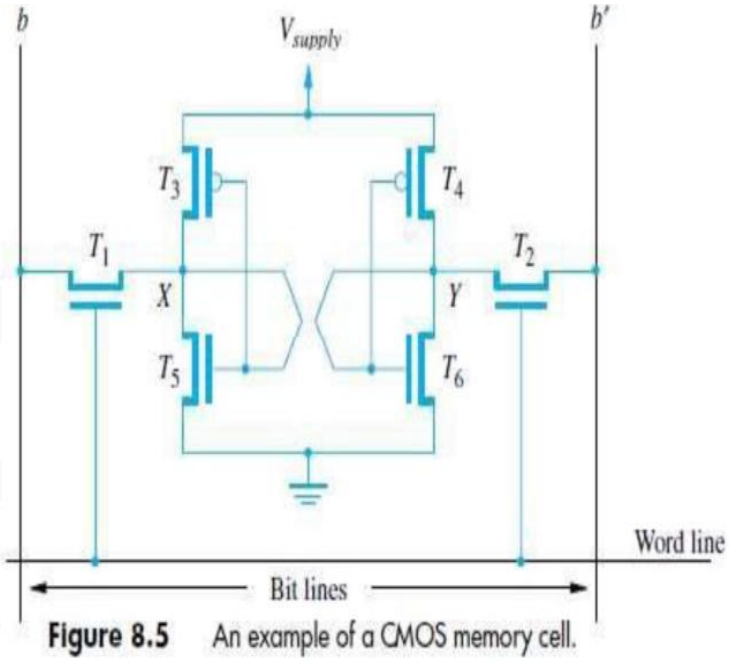


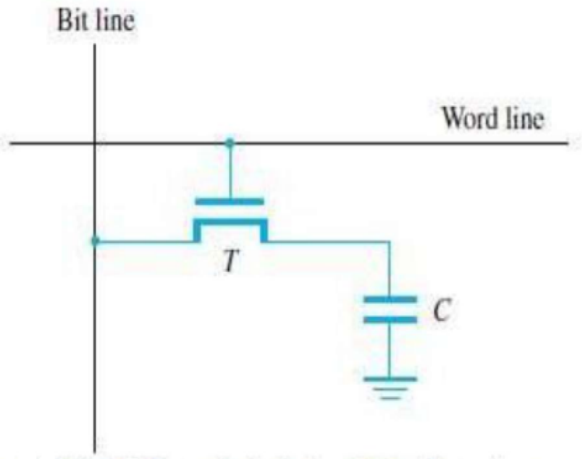
Figure 8.5 An example of a CMOS memory cell.

CMOS Cell

- Transistor pairs (T_3, T_5) and (T_4, T_6) form the inverters in the latch (Figure 8.5).
- In state 1, the voltage at point X is high by having T_5, T_6 ON and T_4, T_5 are OFF.
- Thus, T_1 and T_2 returned ON (Closed), bit-line b and b" will have high and low signals respectively.
- **Advantages:**
  1) It has low power consumption „." the current flows in the cell only when the cell is active.
  2) Static RAM"s can be accessed quickly. It access time is few nanoseconds.
- **Disadvantage:** SRAMs are said to be volatile memories „." their contents are lost when power is interrupted.

ASYNCHRONOUS DRAM

- Less expensive RAMs can be implemented if simple cells are used.
- Such cells cannot retain their state indefinitely. Hence they are called **Dynamic RAM (DRAM).**
- The information stored in a dynamic memory-cell in the form of a charge on a capacitor.
- This charge can be maintained only for tens of milliseconds.
- The contents must be periodically refreshed by restoring this capacitor charge to its full value.



A single-transistor dynamic memory cell.

- In order to store information in the cell, the transistor T is turned „ON"
- The appropriate voltage is applied to the bit-line which charges the capacitor.
- After the transistor is turned off, the capacitor begins to discharge.
- Hence, info. stored in cell can be retrieved correctly before threshold value of capacitor drops down.
- During a read-operation,
  - → transistor is turned „ON"
  - → a sense amplifier detects whether the charge on the capacitor is above the threshold value.
    - If (charge on capacitor) > (threshold value) → Bit-line will have logic value „1".
    - If (charge on capacitor) < (threshold value) → Bit-line will set to logic value „0".

ASYNCHRONOUS DRAM DESCRIPTION

- The 4 bit cells in each row are divided into 512 groups of 8 (Figure 5.7).
- 21 bit address is needed to access a byte in the memory. 21 bit is divided as follows:
  1) 12 address bits are needed to select a row.
     i.e. A_{8-0} → specifies row-address of a byte.
  2) 9 bits are needed to specify a group of 8 bits in the selected row.
     i.e. A_{20-9} → specifies column-address of a byte.

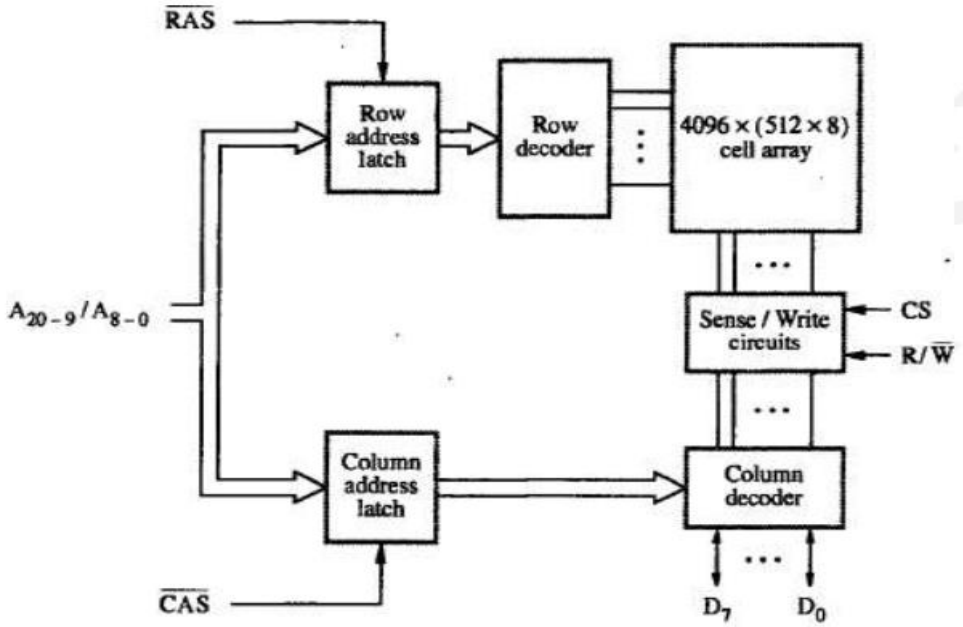


Figure 5.7 Internal organization of a 2M × 8 dynamic memory chip

- During Read/Write-operation,
  - → row-address is applied first.
  - → row-address is loaded into row-latch in response to a signal pulse on **RAS'** input of chip.
    (RAS = Row-address Strobe CAS = Column-address Strobe)
- When a Read-operation is initiated, all cells on the selected row are read and refreshed.
- Shortly after the row-address is loaded, the column-address is
  - → applied to the address pins &
  - → loaded into **CAS'.**
- The information in the latch is decoded.
- The appropriate group of 8 Sense/Write circuits is selected.
  R/W'=1(read-operation) → Output values of selected circuits are transferred to data-lines D_0-D_7.
  R/W'=0(write-operation) → Information on D_0-D_7 are transferred to the selected circuits.
- RAS" & CAS" are active-low so that they cause latching of address when they change from high to low.
- To ensure that the contents of DRAMs are maintained, each row of cells is accessed periodically.
- A special memory-circuit provides the necessary control signals RAS" & CAS" that govern the timing.
- The processor must take into account the delay in the response of the memory.

Fast Page Mode

- Transferring the bytes in sequential order is achieved by applying the consecutive sequence of column-address under the control of successive CAS" signals.
- This scheme allows transferring a block of data at a faster rate.
- The block of transfer capability is called as *fast page mode*.

SYNCHRONOUS DRAM

- The operations are directly synchronized with clock signal (Figure 8.8).
- The address and data connections are buffered by means of registers.
- The output of each sense amplifier is connected to a latch.
- A Read-operation causes the contents of all cells in the selected row to be loaded in these latches.
- Data held in latches that correspond to selected columns are transferred into data-output register.
- Thus, data becoming available on the data-output pins.

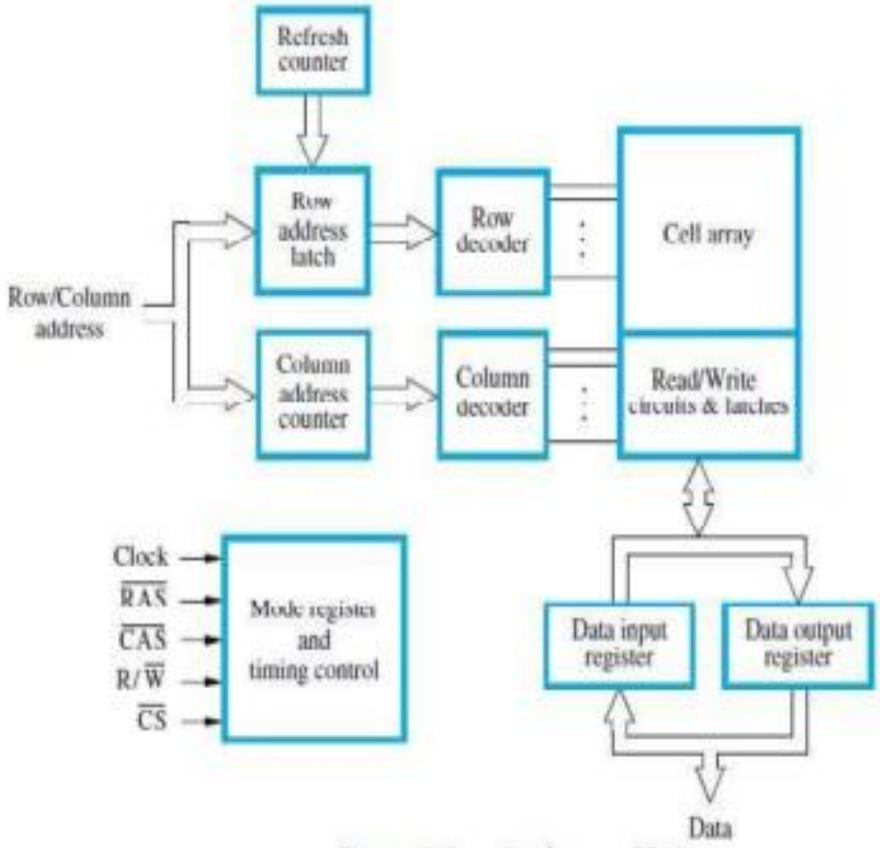


Figure 8.8 Synchronous DRAM.

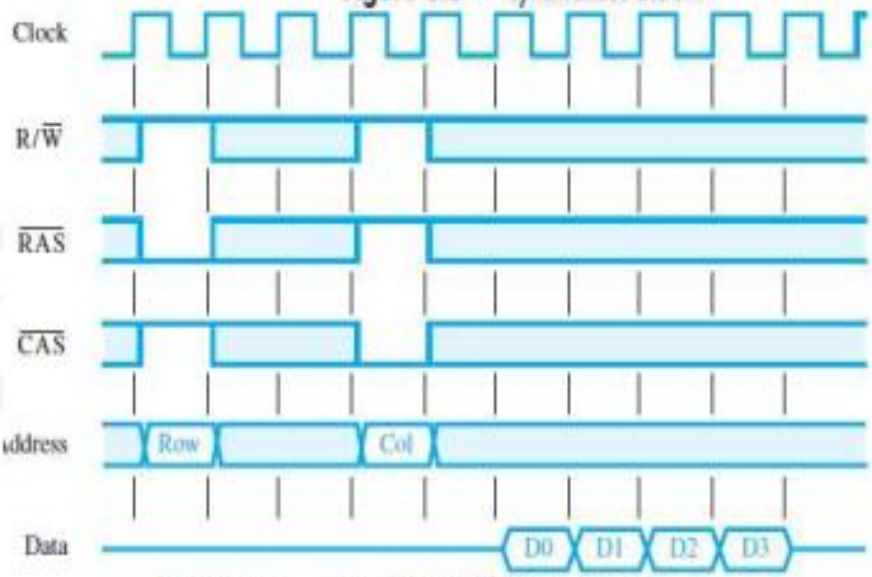


Figure 8.9 A burst read of length 4 in an SDRAM.

- First, the row-address is latched under control of RAS" signal (Figure 8.9).
- The memory typically takes 2 or 3 clock cycles to activate the selected row.
- Then, the column-address is latched under the control of CAS" signal.
- After a delay of one clock cycle, the first set of data bits is placed on the data-lines.
- *SDRAM automatically increments column-address to access next 3 sets of bits in the selected row.*

LATENCY & BANDWIDTH

• A good indication of performance is given by 2 parameters: 1) Latency 2) Bandwidth.

Latency

• It refers to the amount of time it takes to transfer a word of data to or from the memory.
• For a transfer of single word, the latency provides the complete indication of memory performance.
• For a block transfer, the latency denotes the time it takes to transfer the first word of data.

Bandwidth

• It is defined as the number of bits or bytes that can be transferred in one second.
• Bandwidth mainly depends on
    1) The speed of access to the stored data &
    2) The number of bits that can be accessed in parallel.

DOUBLE DATA RATE SDRAM (DDR-SDRAM)

• The standard SDRAM performs all actions on the rising edge of the clock signal.
• The DDR-SDRAM transfer data on both the edges (loading edge, trailing edge).
• The Bandwidth of DDR-SDRAM is doubled for long burst transfer.
• To make it possible to access the data at high rate, the cell array is organized into two banks.
• Each bank can be accessed separately.
• Consecutive words of a given block are stored in different banks.
• Such interleaving of words allows simultaneous access to two words.
• The two words are transferred on successive edge of the clock.

SRAM versus DRAM

→ Both volatile
    —Power must be continuously supplied to the memory to preserve the bit values

→ DRAM
    —Simpler to build, smaller
    —More dense (smaller cells = more cells per unit area)
    —Less expensive
    —Requires the supporting of refresh circuitry
    —used for large memory requirements
    —Used for main memory

→ SRAM
    —Faster
    —less dense and more expensive
    —No refresh circuitry needed
    —Used for cache memory (both on and off chip)

READ ONLY MEMORY (ROM)

- Both SRAM and DRAM chips are volatile, i.e. They lose the stored information if power is turned off.
- Many application requires non-volatile memory which retains the stored information if power is turned off.
- For ex:

  OS software has to be loaded from disk to memory i.e. it requires non-volatile memory.
- Non-volatile memory is used in embedded system.
- Since the normal operation involves only reading of stored data, a memory of this type is called ROM
  - **At Logic value '0'** → Transistor(T) is connected to the ground point (P).

    Transistor switch is closed & voltage on bit-line nearly drops to zero (Figure 8.11).
  - **At Logic value '1'** → Transistor switch is open.

    The bit-line remains at high voltage.

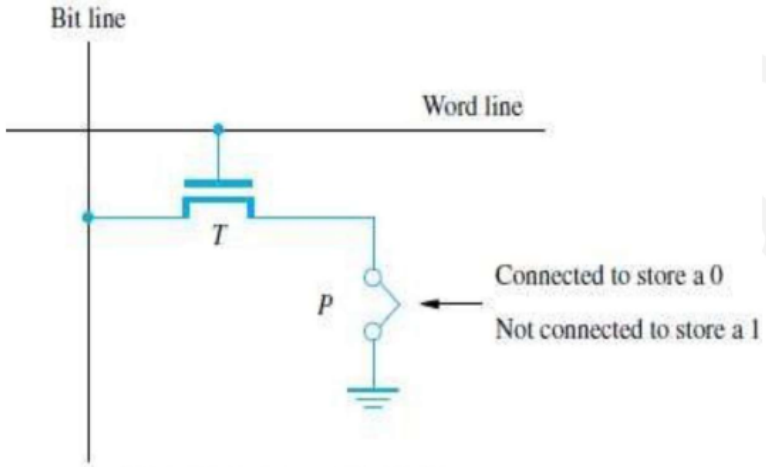


Figure 8.11 A ROM cell.

- To read the state of the cell, the word-line is activated.
- A Sense circuit at the end of the bit-line generates the proper output value.

TYPES OF ROM

- Different types of non-volatile memory are
  1) PROM
  2) EPROM
  3) EEPROM &
  4) Flash Memory (Flash Cards & Flash Drives)

PROM (PROGRAMMABLE ROM)

- PROM allows the data to be loaded by the user.
- Programmability is achieved by inserting a „fuse" at point P in a ROM cell.
- Before PROM is programmed, the memory contains all 0"s.
- User can insert 1"s at required location by burning-out fuse using high current-pulse.
- This process is irreversible.
- **Advantages:**
  1) It provides flexibility.
  2) It is faster.
  3) It is less expensive because they can be programmed directly by the user.

EPROM (ERASABLE REPROGRAMMABLE ROM)

- EPROM allows
  - → stored data to be erased and
  - → new data to be loaded.
- In cell, a connection to ground is always made at „P" and a special transistor is used.
- The transistor has the ability to function as
  - → a normal transistor or
  - → a disabled transistor that is always turned „off".
- Transistor can be programmed to behave as a permanently open switch, by injecting charge into it.
- Erasure requires dissipating the charges trapped in the transistor of memory-cells.
  This can be done by exposing the chip to ultra-violet light.
- **Advantages:**
  1) It provides flexibility during the development-phase of digital-system.
  2) It is capable of retaining the stored information for a long time.
- **Disadvantages:**
  1) The chip must be physically removed from the circuit for reprogramming.
  2) The entire contents need to be erased by UV light.

EEPROM (ELECTRICALLY ERASABLE ROM)

- **Advantages:**
  1) It can be both programmed and erased electrically.
  2) It allows the erasing of all cell contents selectively.
- **Disadvantage:** It requires different voltage for erasing, writing and reading the stored data.

FLASH MEMORY

- In EEPROM, it is possible to read & write the contents of a single cell.
- In Flash device, it is possible to read contents of a single cell & write entire contents of a block.
- Prior to writing, the previous contents of the block are erased.
  Eg. In MP3 player, the flash memory stores the data that represents sound.
- Single flash chips cannot provide sufficient storage capacity for embedded-system.
- **Advantages:**
  1) Flash drives have greater density which leads to higher capacity & low cost per bit.
  2) It requires single power supply voltage & consumes less power.
- There are 2 methods for implementing larger memory: 1) Flash Cards & 2) Flash Drives

  1) Flash Cards
  - ➢ One way of constructing larger module is to mount flash-chips on a small card.
  - ➢ Such flash-card have standard interface.
  - ➢ The card is simply plugged into a conveniently accessible slot.
  - ➢ Memory-size of the card can be 8, 32 or 64MB.
  - ➢ Eg: A minute of music can be stored in 1MB of memory. Hence 64MB flash cards can store an hour of music.

  2) Flash Drives
  - ➢ Larger flash memory can be developed by replacing the hard disk-drive.
  - ➢ The flash drives are designed to fully emulate the hard disk.
  - ➢ The flash drives are solid state electronic devices that have no movable parts.

  Advantages:
  1) They have shorter seek & access time which results in faster response.
  2) They have low power consumption. ∴ they are attractive for battery driven application.
  3) They are insensitive to vibration.

  Disadvantages:
  1) The capacity of flash drive (<1GB) is less than hard disk (>1GB).
  2) It leads to higher cost per bit.
  3) Flash memory will weaken after it has been written a number of times (typically at least 1 million times).

SPEED, SIZE COST

Characteristics	SRAM	DRAM	Magnetis Disk
Speed	Very Fast	Slower	Much slower than DRAM
Size	Large	Small	Small
Cost	Expensive	Less Expensive	Low price

Memory	Speed	Size	Cost
Registers	Very high	Lower	Very Lower
Primary cache	High	Lower	Low
Secondary cache	Low	Low	Low
Main memory	Lower than Seconadry cache	High	High
Secondary Memory	Very low	Very High	Very High

- The main-memory can be built with DRAM (Figure 8.14)
- Thus, SRAM"s are used in smaller units where speed is of essence.
- The Cache-memory is of 2 types:
  - **1) Primary/Processor Cache** (Level1 or L1 cache)
    - ➢ It is always located on the processor-chip.
  - **2) Secondary Cache** (Level2 or L2 cache)
    - ➢ It is placed between the primary-cache and the rest of the memory.

- The access time for main-memory is about 10 times longer than the access time for L1 cache

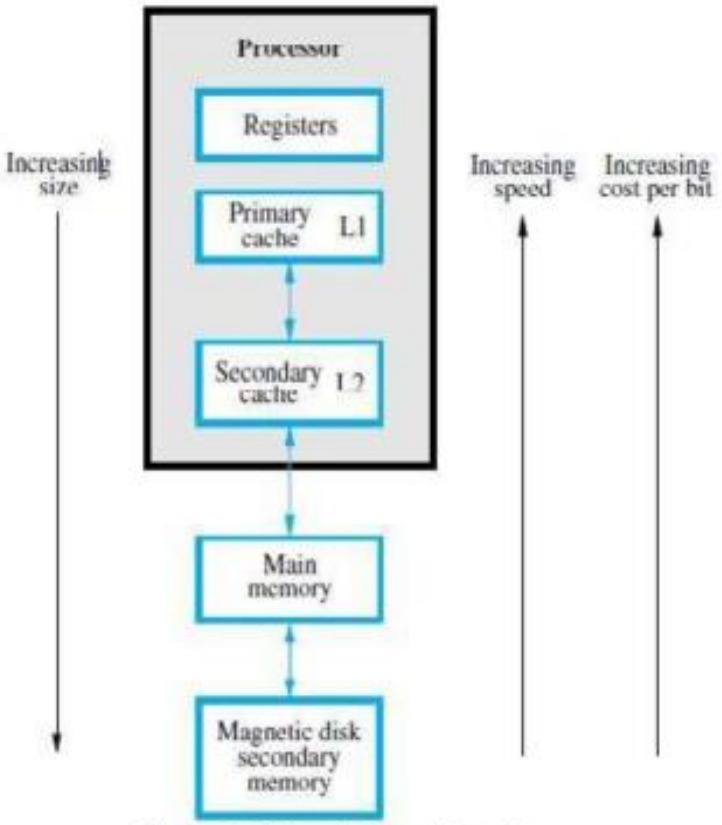


Figure 8.14 Memory hierarchy.

CACHE MEMORIES

- The effectiveness of cache mechanism is based on the property of „**Locality of Reference'.**

Locality of Reference

- Many instructions in the localized areas of program are executed repeatedly during some time period
- Remainder of the program is accessed relatively infrequently (Figure 8.15).
- There are 2 types:
  - **1) Temporal**
    - The recently executed instructions are likely to be executed again very soon.
  - **2) Spatial**
    - Instructions in close proximity to recently executed instruction are also likely to be executed soon.
- If active segment of program is placed in cache-memory, then total execution time can be reduced.
- **Block** refers to the set of contiguous address locations of some size.
- The cache-line is used to refer to the cache-block.

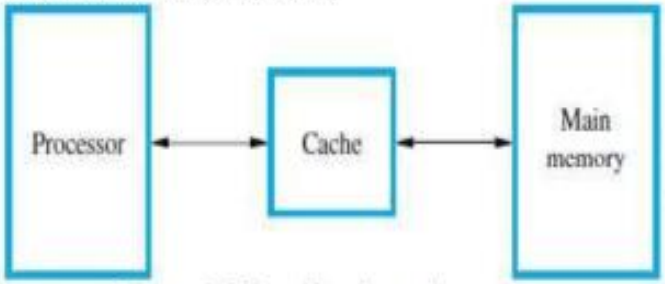


Figure 8.15 Use of a cache memory.

- The Cache-memory stores a reasonable number of blocks at a given time.
- This number of blocks is small compared to the total number of blocks available in main-memory.
- Correspondence b/w main-memory-block & cache-memory-block is specified by mapping-function.
- Cache control hardware decides which block should be removed to create space for the new block.
- The collection of rule for making this decision is called the **Replacement Algorithm.**
- The cache control-circuit determines whether the requested-word currently exists in the cache.
- The write-operation is done in 2 ways: 1) Write-through protocol & 2) Write-back protocol.

Write-Through Protocol

- Here the cache-location and the main-memory-locations are updated simultaneously.

Write-Back Protocol

- This technique is to
  - → update only the cache-location &
  - → mark the cache-location with associated flag bit called **Dirty/Modified Bit.**
- The word in memory will be updated later, when the marked-block is removed from cache.

During Read-operation

- If the requested-word currently not exists in the cache, then **read-miss** will occur.
- To overcome the read miss, *Load-through/Early restart protocol* is used.

Load-Through Protocol

- The block of words that contains the requested-word is copied from the memory into cache.
- After entire block is loaded into cache, the requested-word is forwarded to processor.

During Write-operation

- If the requested-word not exists in the cache, then **write-miss** will occur.
  1) If **Write Through Protocol** is used, the information is written directly into main-memory.
  2) If **Write Back Protocol** is used,
     - → then block containing the addressed word is first brought into the cache &
     - → then the desired word in the cache is over-written with the new information.

MAPPING-FUNCTION

- Here we discuss about 3 different mapping-function:
  1) Direct Mapping
  2) Associative Mapping
  3) Set-Associative Mapping

DIRECT MAPPING

- The block-j of the main-memory maps onto block-j modulo-128 of the cache (Figure 8.16).
- When the memory-blocks 0, 128, & 256 are loaded into cache, the block is stored in cache-block 0.
  Similarly, memory-blocks 1, 129, 257 are stored in cache-block 1.
- The contention may arise when
  1) When the cache is full.
  2) When more than one memory-block is mapped onto a given cache-block position.
- The contention is resolved by
  allowing the new blocks to overwrite the currently resident-block.
- Memory-address determines placement of block in the cache.

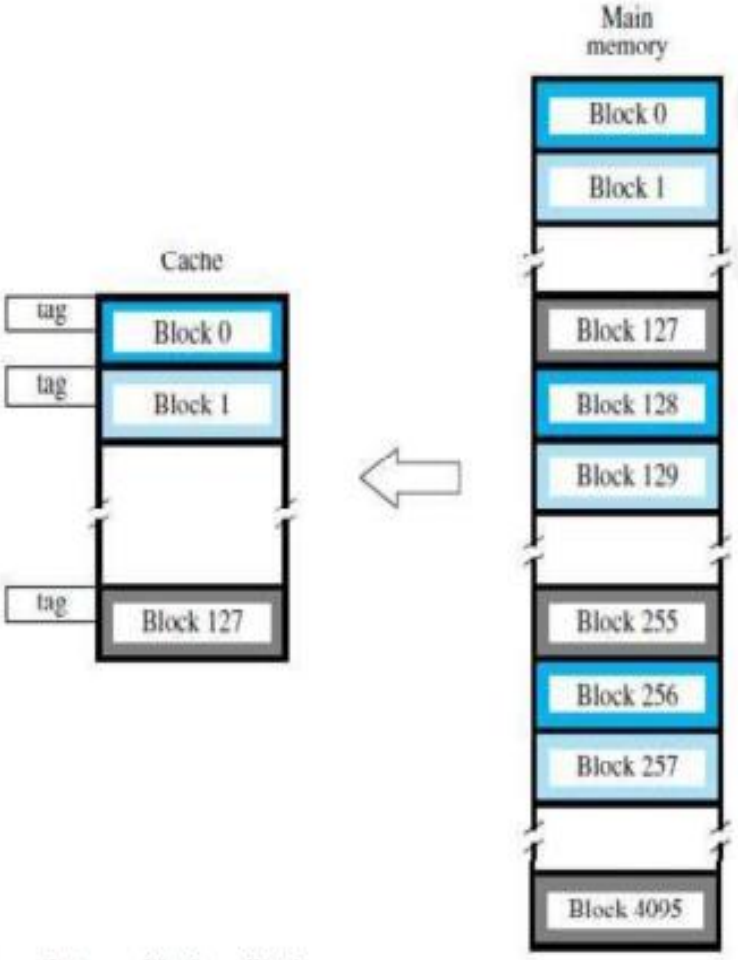


Figure 8.16 Direct-mapped cache.

- The memory-address is divided into 3 fields:

1) Low Order 4 bit field

- Selects one of 16 words in a block.

2) 7 bit cache-block field

- 7-bits determine the cache-position in which new block must be stored.

3) 5 bit Tag field

- 5-bits memory-address of block is stored in 5 tag-bits associated with cache-location.

- As execution proceeds,
  5-bit tag field of memory-address is compared with tag-bits associated with cache-location.
  If they match, then the desired word is in that block of the cache.
  Otherwise, the block containing required word must be first read from the memory.
  And then the word must be loaded into the cache.

ASSOCIATIVE MAPPING

- The memory-block can be placed into any cache-block position. (Figure 8.17).
- 12 tag-bits will identify a memory-block when it is resolved in the cache.
- Tag-bits of an address received from processor are compared to the tag-bits of each block of cache.
- This comparison is done to see if the desired block is present.

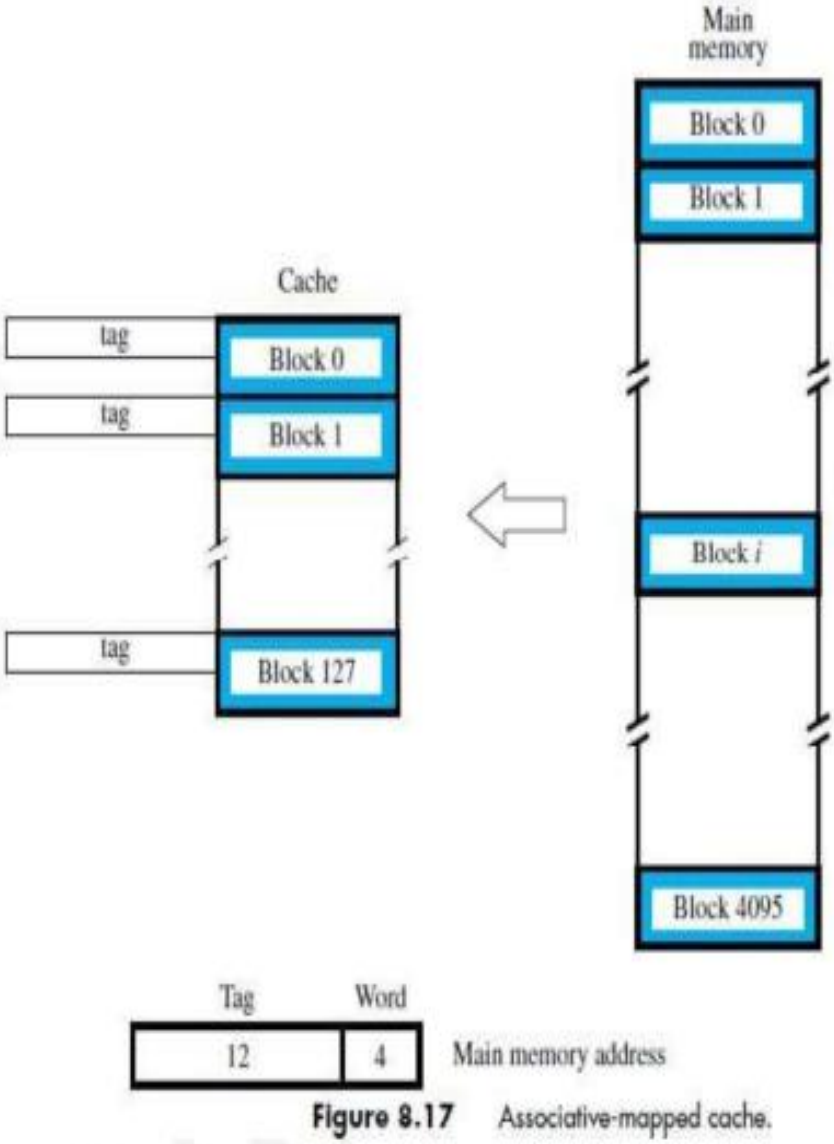


Figure 8.17 Associative-mapped cache.

- It gives complete freedom in choosing the cache-location.
- A new block that has to be brought into the cache has to replace an existing block if the cache is full.
- The memory has to determine whether a given block is in the cache.
- **Advantage:** It is more flexible than direct mapping technique.
- **Disadvantage:** Its cost is high.

SET-ASSOCIATIVE MAPPING

- It is the combination of direct and associative mapping. (Figure 8.18).
- The blocks of the cache are grouped into sets.
- The mapping allows a block of the main-memory to reside in any block of the specified set.
- The cache has 2 blocks per set, so the memory-blocks 0, 64, 128..... 4032 maps into cache set „0"
- The cache can occupy either of the two block position within the set.

6 bit set field

- Determines which set of cache contains the desired block.

6 bit tag field

- The tag field of the address is compared to the tags of the two blocks of the set.
- This comparison is done to check if the desired block is present.

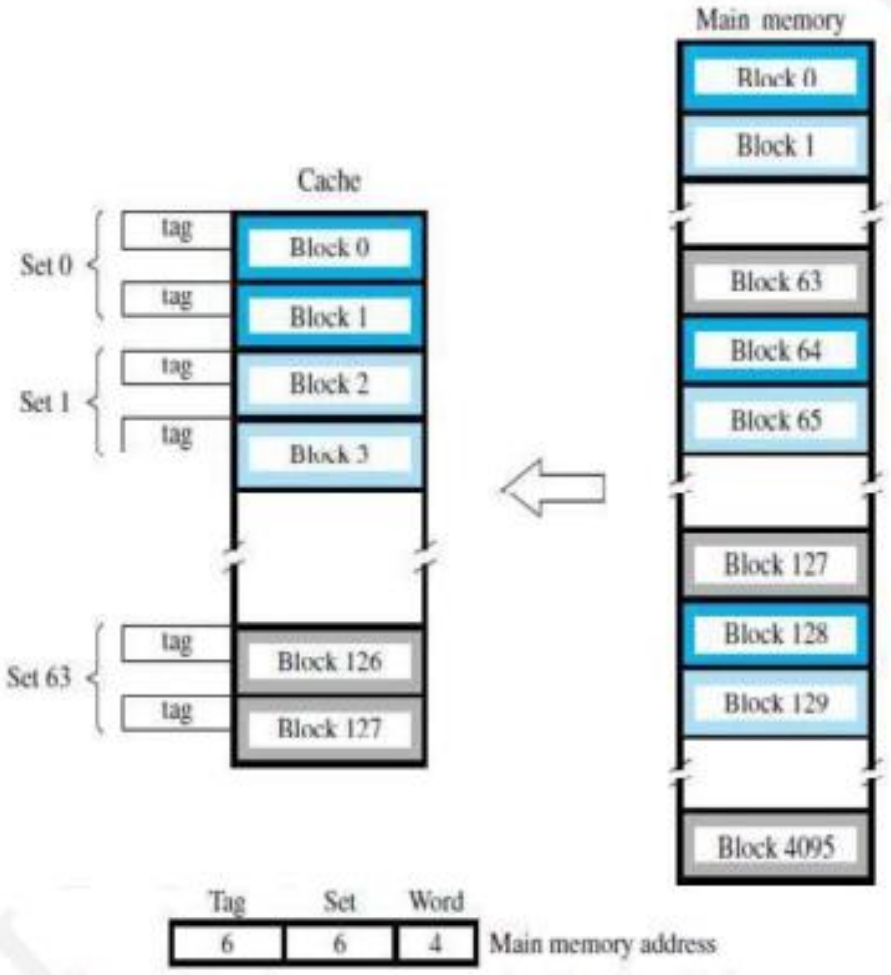


Figure 8.18 Set-associative-mapped cache with two blocks per set.

- The cache which contains 1 block per set is called **direct mapping.**
- A cache that has „k" blocks per set is called as "**k-way set associative cache**".
- Each block contains a control-bit called a **valid-bit.**
- The Valid-bit indicates that whether the block contains valid-data.
- The dirty bit indicates that whether the block has been modified during its cache residency.

Valid-bit=0 → When power is initially applied to system.

Valid-bit=1 → When the block is loaded from main-memory at first time.

- If the main-memory-block is updated by a source & if the block in the source is already exists in the cache, then the valid-bit will be cleared to "0".
- If Processor & DMA uses the same copies of data then it is called as **Cache Coherence Problem**.
- **Advantages:**

1) Contention problem of direct mapping is solved by having few choices for block placement.
2) The hardware cost is decreased by reducing the size of associative search.

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.

Direct mapping

- there is only one possible line for any particular block, and no choice is possible.

Replacement Algorithms Associative & Set Associative

- Least Recently used (LRU)
  - —Replace that block in the set that has been in the cache longest with no reference to it
- First in first out (FIFO)
  - —replace block that has been placed into cache first.
- Least frequently used
  - —replace block which has had fewest references
- Random
  - —pick a cache line at random