Module -3

THE MEMORY SYSTEM

(Topics to study for Internals-2

Basic Concepts

- The maximum size of the Main Memory is determined by its addressing scheme.
- A 16-bit computer that generates 16-bit addresses is capable of addressing upto 2^{16} =64K memory locations.
- If a machine generates 32-bit addresses, it can access upto 2^{32} = 4G memory locations.
- This number represents the size of *address space* of the computer.

Contd - Word-address and Byte-address

- If the smallest addressable unit of information is a memory word, the machine is called word-addressable.

- If individual memory bytes are assigned distinct addresses, the computer is called byte-addressable

- A word-address assignment for a 32-bit computer:

Word Address	**Byte Address**			
0	0	1	2	3
4	4	5	6	7
8	8	9	10	11
.		…..		

Contd – Measure of speed of memory unit

- **Memory Access Time -** the time between Read/Write and MFC

It is the time that elapses between the initiation of an operation and the completion of that operation

- **Memory Cycle Time -** the time between two successive Read/Write operations

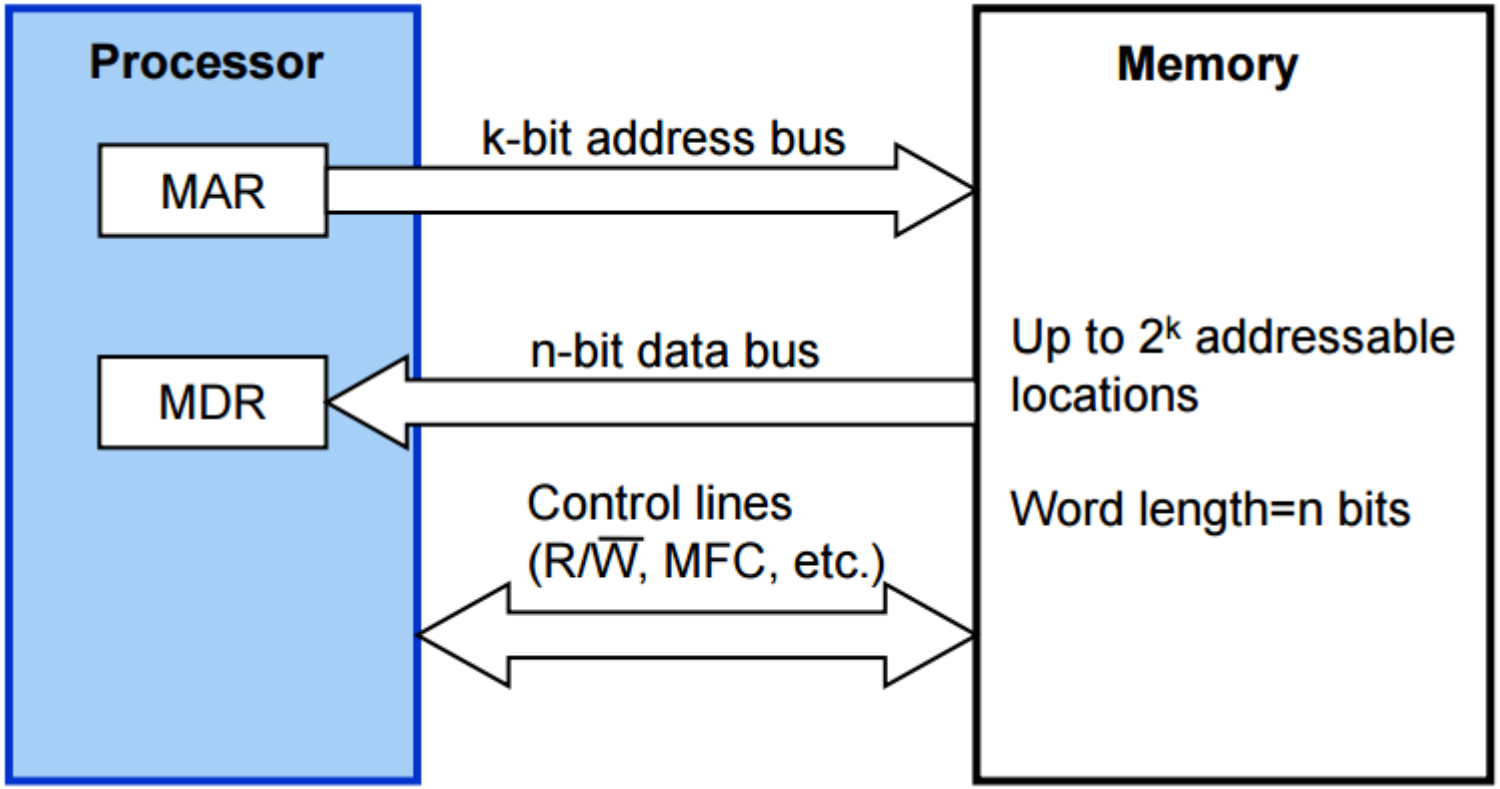
It is the minimum time delay required between the initiations of two successive memory operations

- The cycle time is usually slightly longer than the access time.

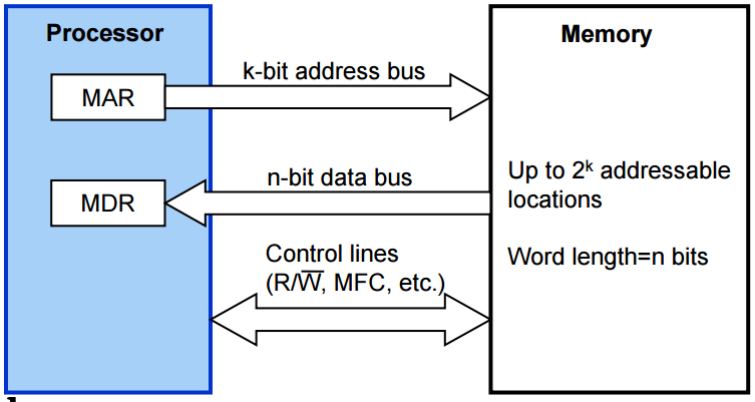
Contd.. - Random Access Memory (RAM)

- Access time for any location is independent of the location's address.
- Any location can be accessed in some fixed amount of time
- i.e access time for all locations is same
- In serial access storage devices (magnetic tapes & disks) access time is different for different locations

CPU-Main Memory Connection



Memory Read operation :



- CPU loads the address into MAR
- R/W' is set to 1
- data from the MM is placed on the data bus
- set MFC (memory function complete) to 1
- Copy the data from data lines to MDR

Memory Write operation :

- CPU loads the address into MAR and data to MDR
- R/W' is set to 0
- MM control circuitry loads the data into appropriate location
- sets MFC to 1

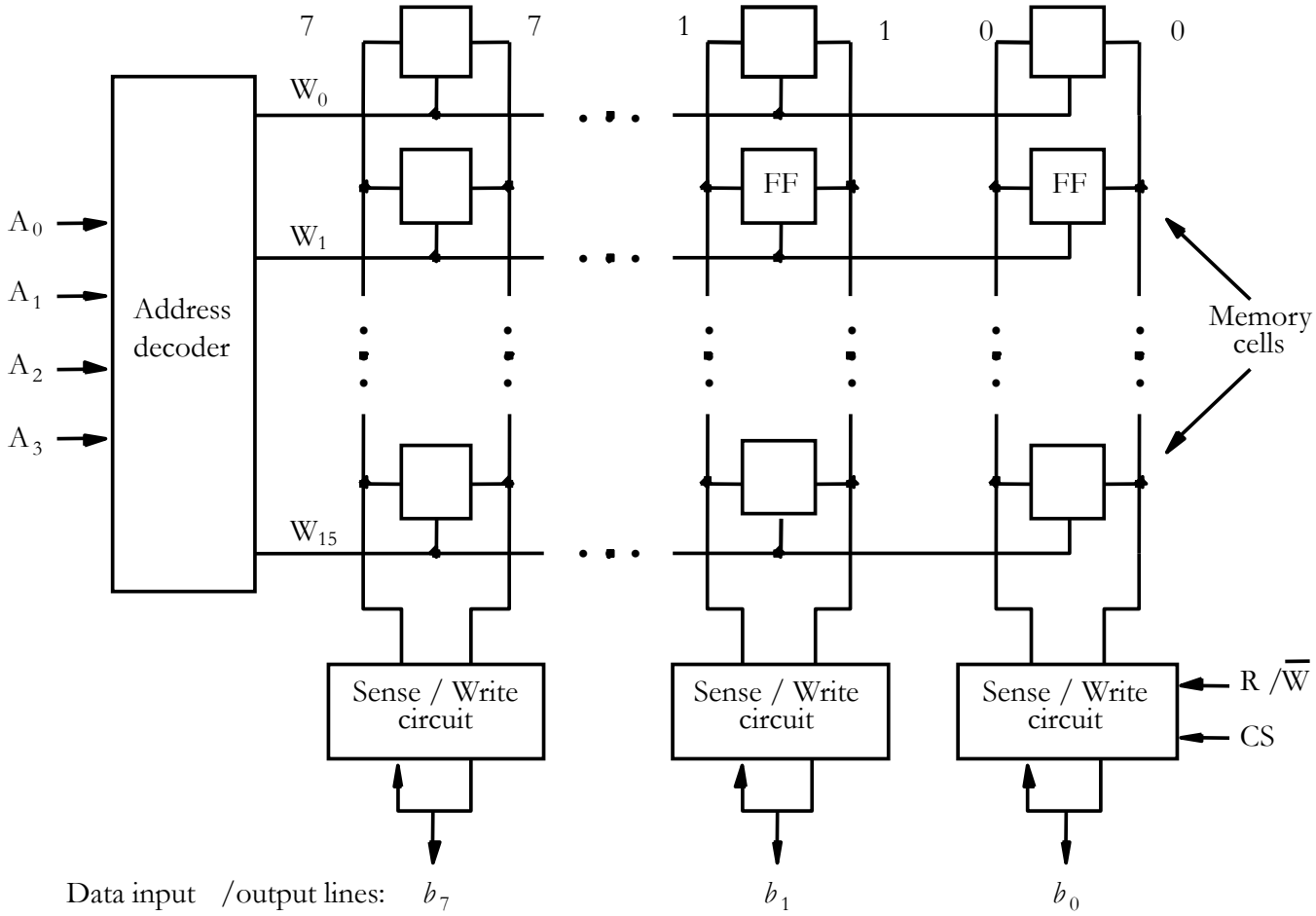
Internal Organization of Semiconductor Memory Chips

- A memory cell is capable of storing one bit of information
- Cells are organized in the form of an array
- A row of cells constitutes a **memory word**
- All cells of a row are connected to the **word line**
- word line is driven by the address decoder

Contd..

- Cells in each column are connected to a sense/write circuit by two lines known as **bit lines**
- sense/write circuits are connected to the data input/output lines
- During a READ operation Sense/Write circuits sense, or read, the information stored in the cells selected by a word line
- During a WRITE operation, they receive input information and store it in the cells of the selected word

Internal organization of a 16x8 memory chip



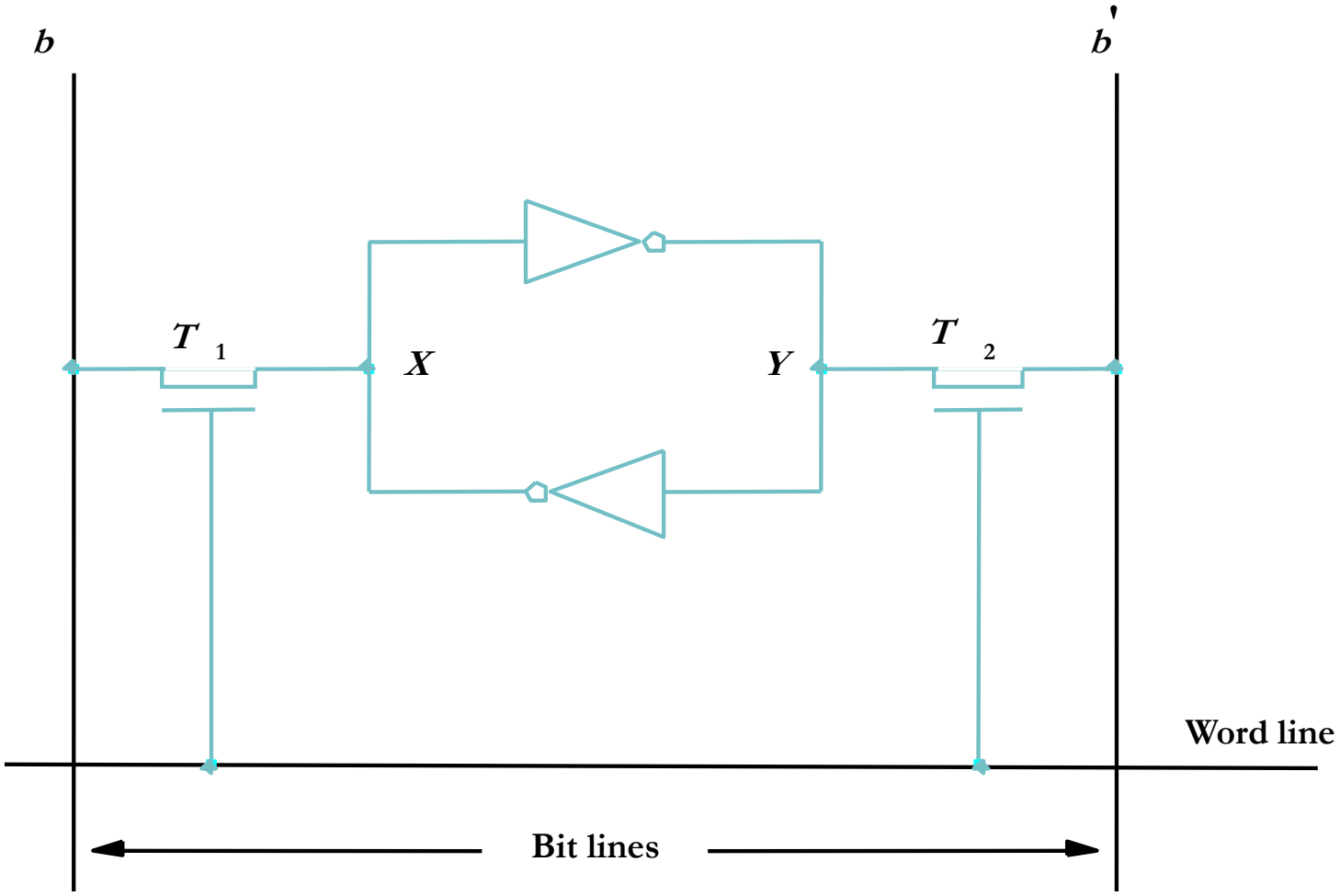
Contd..

- This chip consists of 16 words of 8 bits each
- Usually referred to as a 16 x 8 organization; stores 128 bits
- Requires 14 + 2 external connections

Static and Dynamic Memory

- **Static memory** : can store information as long as current flow to the cell is maintained.
- Also called Static RAM (SRAM)
- **Dynamic memory** : requires not only the maintaining of a power supply, but also a periodic "refresh" to maintain the information stored in them.
- Also called Dynamic RAM (DRAM)

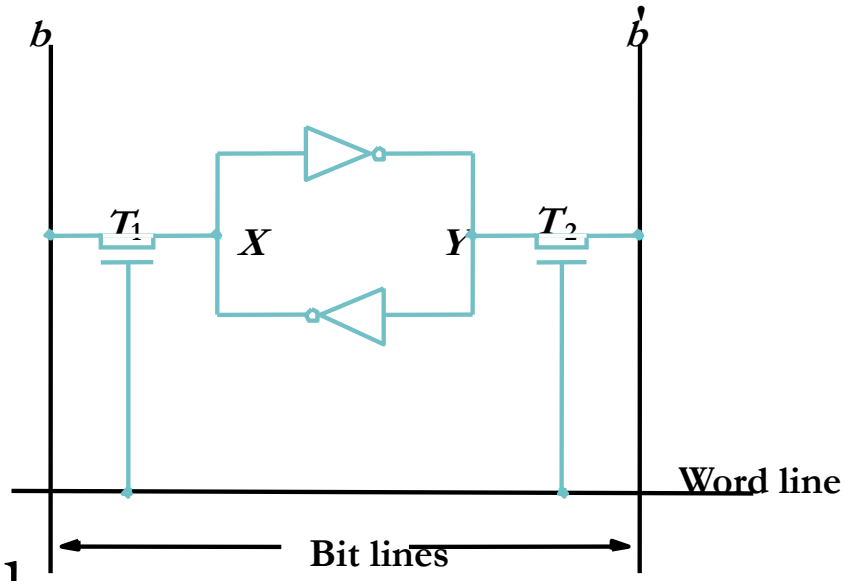
A static memory cell



Contd…

- 2 inverters form a latch
- T1 and T2 connect the latch to bit lines b and b’
- T1 and T2 act as switches; closed if the word line is high.
- If word line is low, T1 and T2 will be off ; latch retains its state.

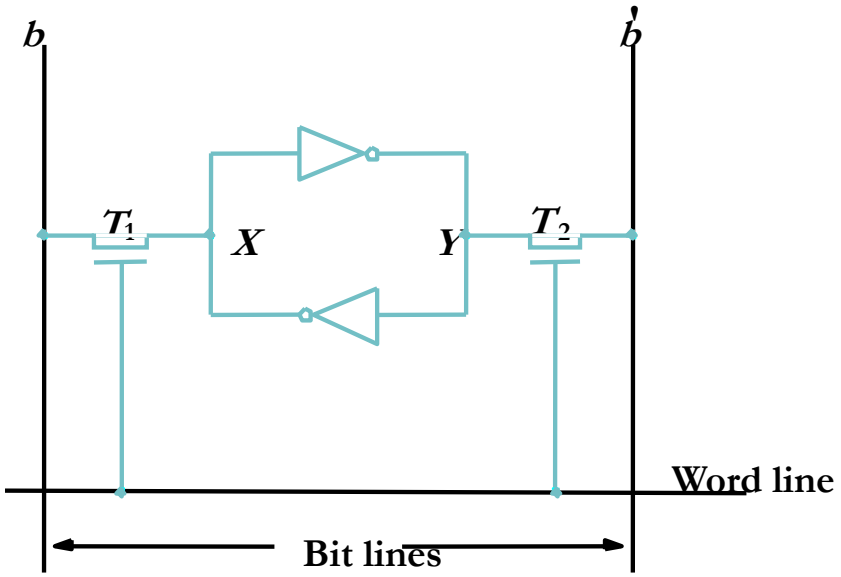
Contd..



Read operation:

- activate the word line
- Now, T1 and T2 are closed
- If the cell is in state 1, b becomes 1 and b' becomes 0 (and vice versa)
- Sense/Write circuit at the end of bit line set the o/p accordingly

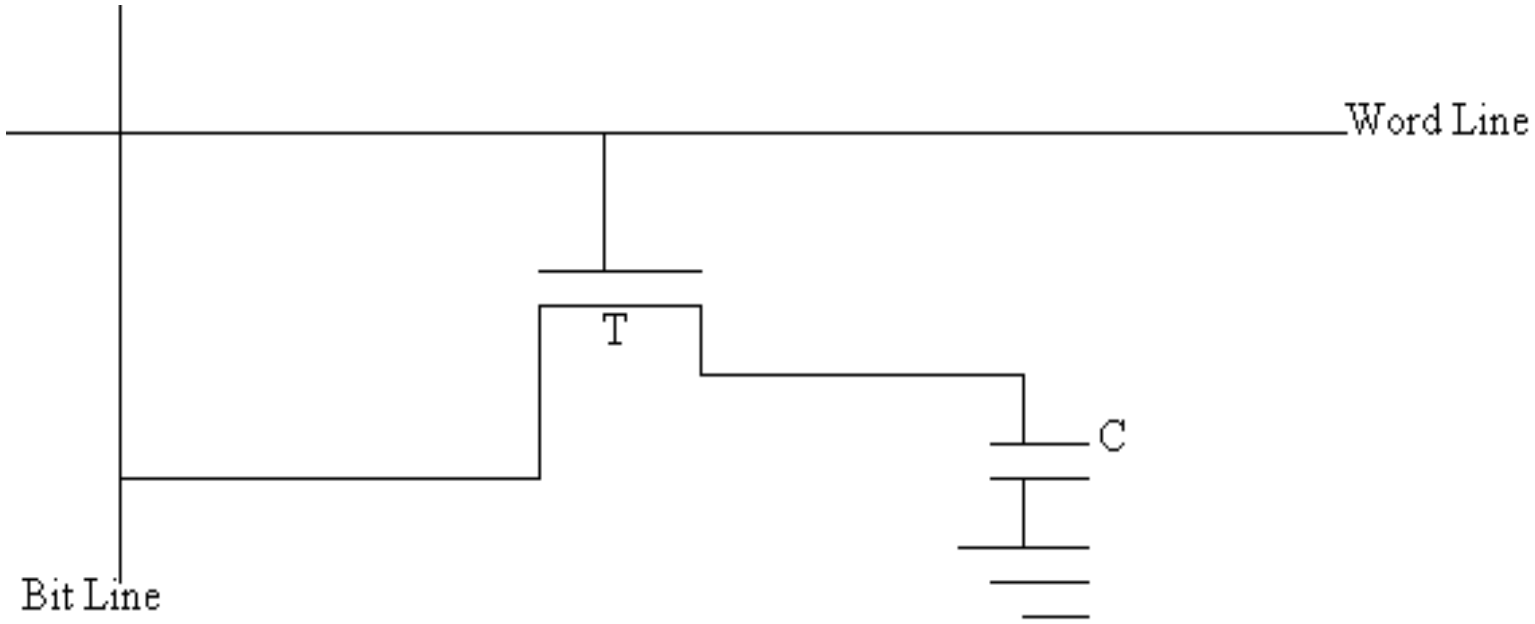
Contd..



Write operation:

- Place appropriate value on b (and its complement on b')
- Activate the word line
- Now, the cell will attain the state of bit line b

A dynamic memory cell

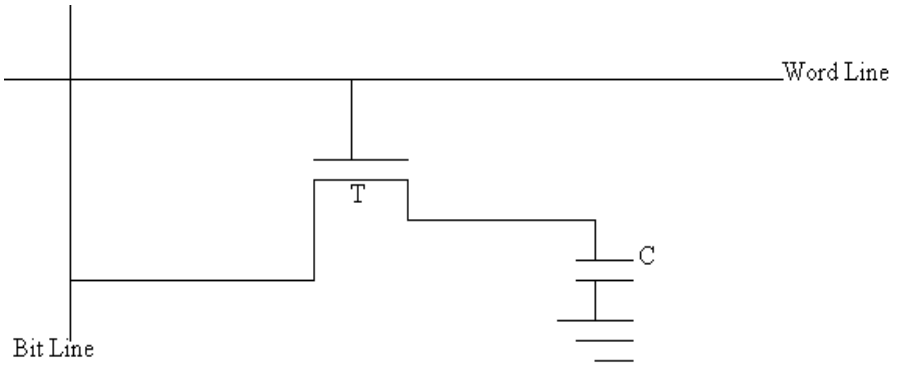


Information is stored in the form of charge on the capacitor

Contd..

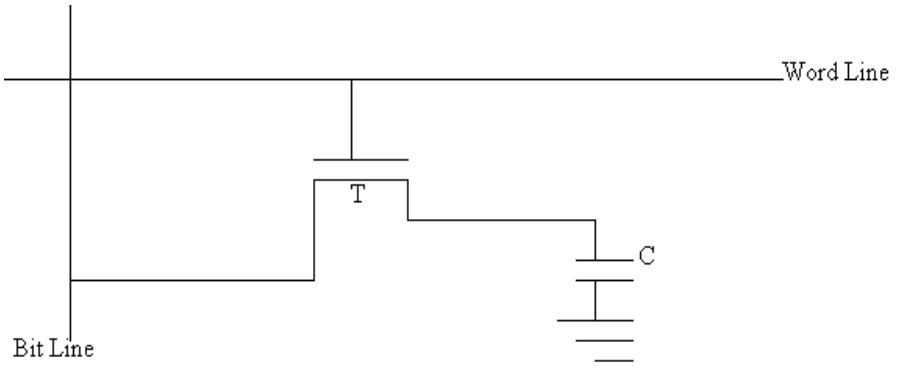
Write operation :

- Transistor T is turned on by activating the word line
- Appropriate voltage is applied to the bit line
- Now, the capacitor is charged to the level of voltage in the bit line



Contd. - Refreshing

- After the transistor is turned off, capacitor begins to discharge due to
  - its leakage resistance
  - transistor conducting a tiny current in the off state
- Hence the need of refreshing, to restore the capacitor's charge
- The information is read correctly only if the cell is read before the charge drops below a threshold value



Contd. - refreshing

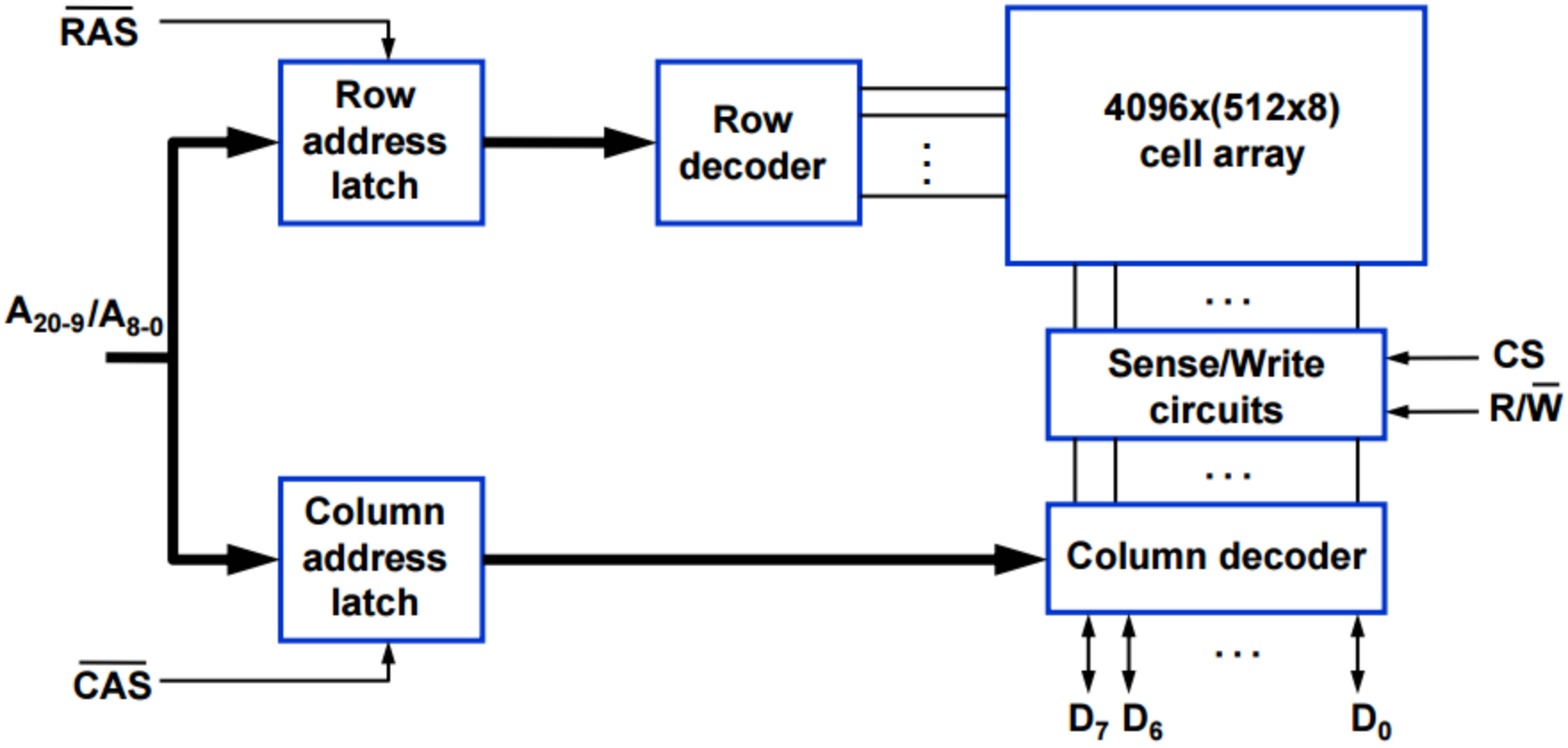
- During Read operation, the sense amplifier connected to the bit line checks whether the charge on the capacitor is above the threshold value.
- If so, a full voltage is placed in the bit line. This charges the capacitor to full voltage, representing a logic 1 .
- Else, voltage on the bit line is made 0, and now the capacitor will have no charge, representing a logic 0.
- Thus, reading a row refreshes all cells in that row

Asynchronous DRAMS

- **Timing of the memory device is controlled by a specialized memory controller circuit which generates RAS and CAS.**
- **No separate clock signal.**
- **This type of DRAM is known as asynchronous DRAM**

2M x 8 asynchronous dynamic memory chip

Explain the operation of a 16 Mega bit DRAM chip configured as 2Mx8 – 10 marks



Contd..

- Cells are organized in 4K x 4K array (4096 x 4096 bits= 16 mega bits)
- 4096 cells in each row are divided into 512 groups of 8 cells each
- So, each row stores 512 bytes
- 12 address bits are needed to address 4096 rows (i.e, 2^{12}=4096)
- 9 bits are needed to address 512 coloumns (2^9=512). A column is a group of 8 bits
- Hence, 21 bit address

Properties

- Read operation refreshes a cell

  Write operation overwrites a cell

- Refreshing is performed automatically

  Each row is accessed periodically (irrespective of read/write operations)

- Asynchronous memory – RAS and CAS govern the timing. No clock signal.

FAST PAGE MODE

- It is an operation to read consecutive columns in a row
- 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 transfer capability is called as **Fast Page Mode.**

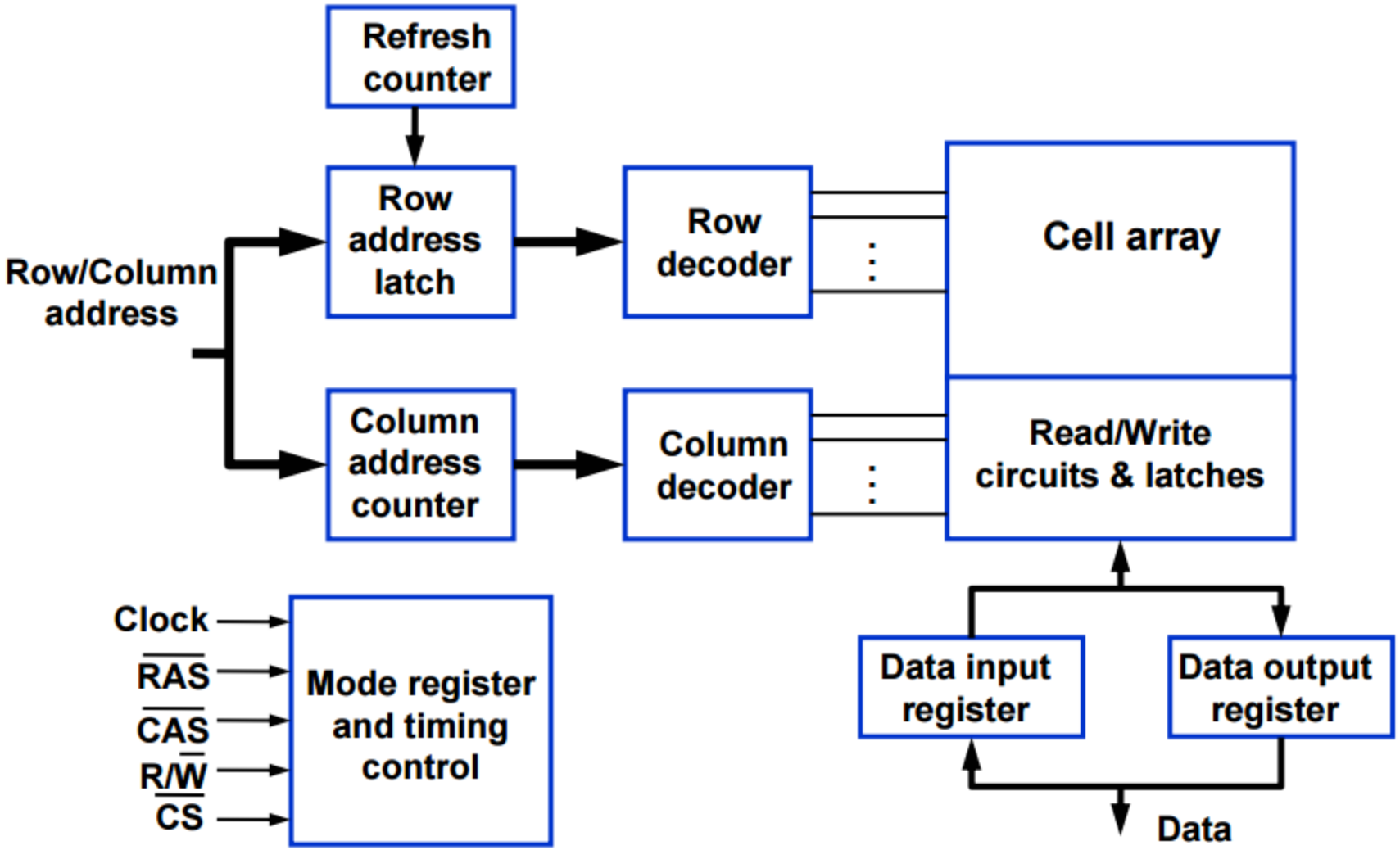
Contd..

- Row and column addresses are multiplexed on 12 pins. (hence no. of pins are reduced)

Read Operation

- row addr is applied. It is loaded to Row Address Latch when RAS (row address strobe) signal is active
- All the cells in the selected row are refreshed
- Now, the column address is applied. It is loaded to Column Address Latch when CAS signal is active

Synchronous DRAMS



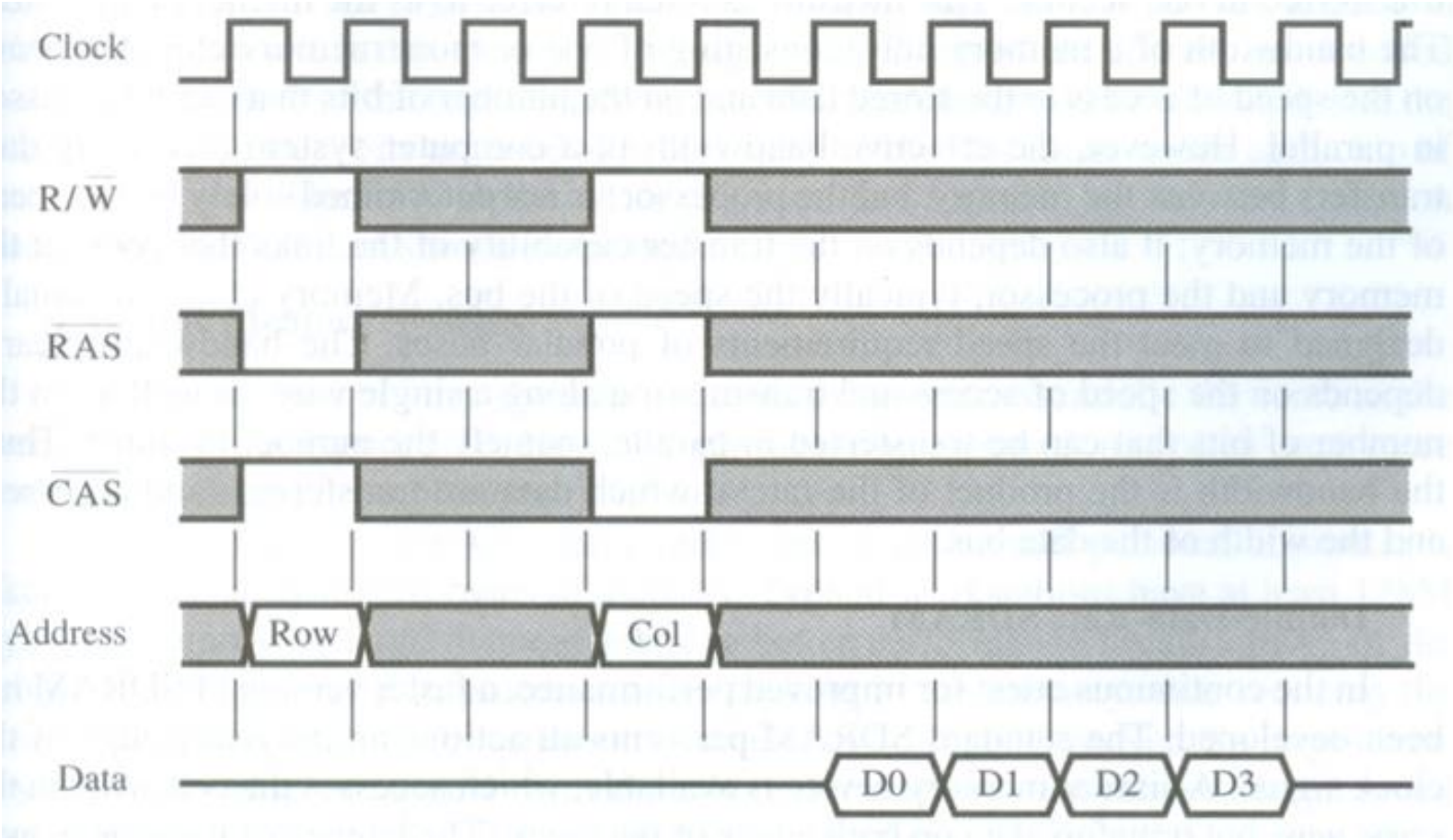
Synchronous DRAMS

- **Operation of SDRAM is directly synchronized with clock signal.**
- The outputs of the Read/Write circuits are connected to latches.
- During a Read operation, the contents of the cells in a row are loaded onto the latches.
- During a refresh operation, the contents of the cells are refreshed without changing the contents of the latches.
- Data held in the latches that correspond to the selected columns are transferred to the output.

Properties of SDRAM

- For a burst mode of operation, successive columns are selected using column address counter and clock.
- CAS signal need not be generated externally.
- A new data is placed during rising edge of the clock
- Mode of operation is programmable using mode register
- Burst mode : to read/write successive columns

Burst read of length 4 in an SDRAM



Cache Memory

- Main memory is very slow compared to the processor
- To reduce the time needed to access the necessary information Cache memory is used
- The cache mechanism is based on the property of programs called **locality of reference**

Contd..

- Most of the execution time of programs is spent on routines in which many instructions are executed repeatedly
- These instructions may constitute a simple loop, nested loops or few procedure that repeatedly call each other

- The main observation is that *many instructions in a few localized areas of the program are repeatedly executed and that the remainder of the program is accessed relatively infrequently*

- This property is called as **Locality of Reference**

Temporal aspect of LOR

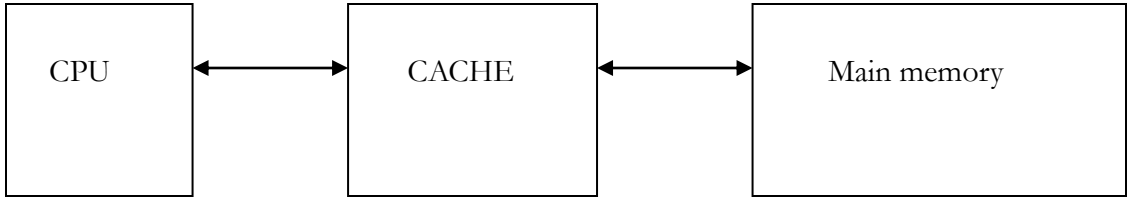
- Temporal aspect means that - recently executed instruction is likely to be executed again very soon

- The above aspect suggests that whenever an information item is first needed, this item should be brought to the cache where it will hopefully remain until it is needed again

Spatial aspect of LOR

- Spatial aspect means that – instructions in close proximity to a recently executed instruction are also likely to be executed very soon
- This suggests that instead of fetching just one item from the main memory to cache, it is useful to fetch several items that reside at adjacent address as well.
- A **block** is a set of contiguous addr. locations

Use of cache



- Read operation- copy the block containing the required word to cache. During program execution, the desired contents are directly read from the cache

Terms…

- Cache mapping – the correspondence between the main memory blocks & cache blocks
- Cache replacement – Done by replacement algorithms.
- Read Hit – the requested word exists in the cache. The requested word is read from the appropriate cache location
- Write Hit - the word to be written exists in the cache.

Contd..

- Two ways to write :
  1. Write-through – cache and main memory locations are updated simultaneously.

     Results in unnecessary memory-write operations when a word is updated frequently in the cache (but is simplest method)

  2. Write-back : main memory is updated later. Update only the cache location, and mark it as updated using dirty or modified bit

     Also results in unnecessary Write operations.

     During write-back all words of the block are written to memory even if a single word has been changed.

Mapping Functions

Assumptions:

- Cache has 128 blocks
- Each block has 16 words
- 4bits to address 16 words

- Main memory has 4K (4096) blocks
- 12 bits to address 4096 blocks

- Total 16 bits in the address field

Direct Mapping

- Mapping function:

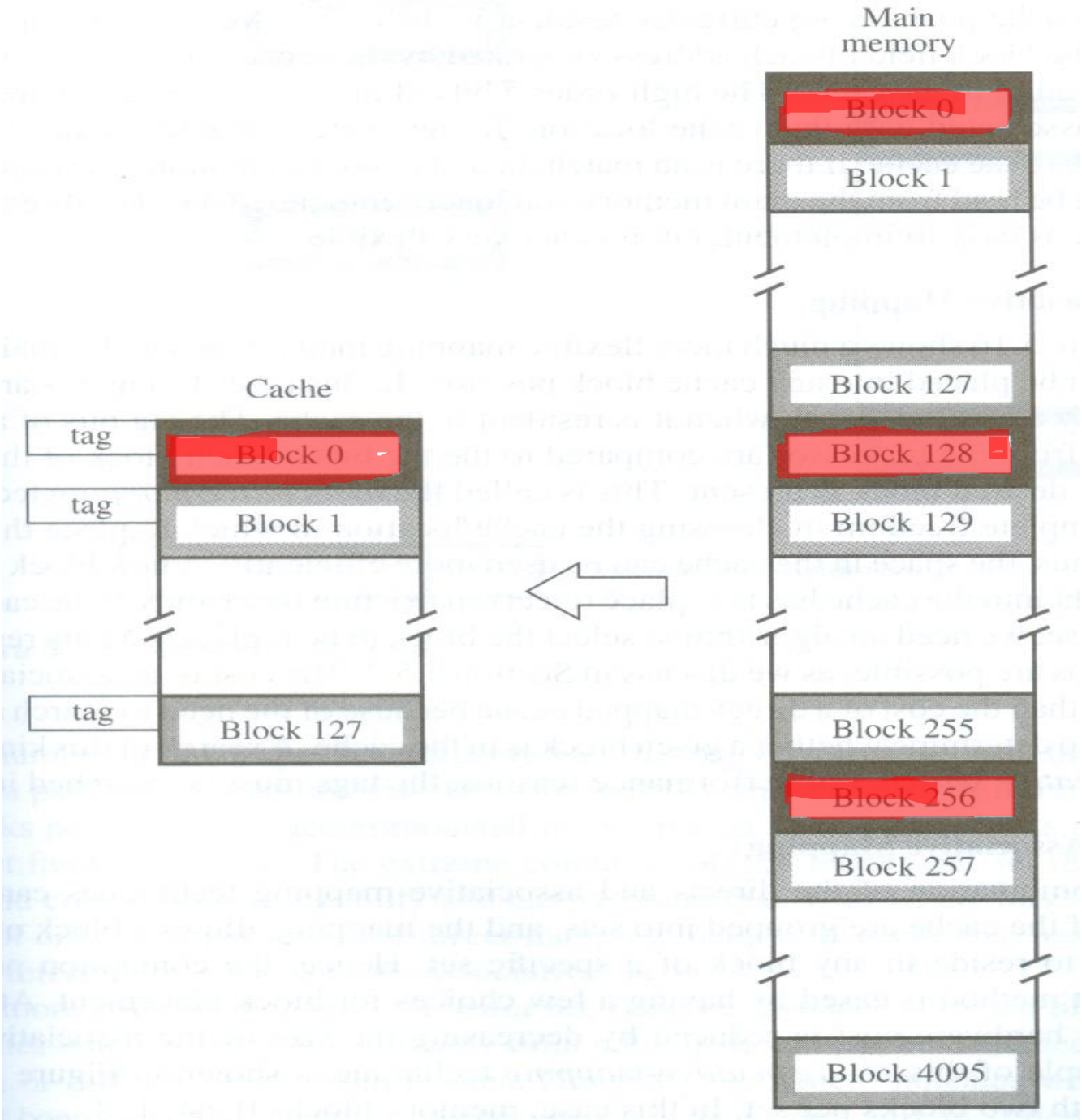
  (MM *Block address*) **MOD** (*Number of blocks in cache*)

- Block j of the main memory maps onto block j mod 128 of the cache.

- Blocks 0, 128, 256,… of main memory mapped to block 0 of cache

- Blocks 1, 129, 257,… of main memory mapped to block 1 of cache

- So, a total of 32 blocks of main memory will be mapped to each cache block (ie 4096/128)

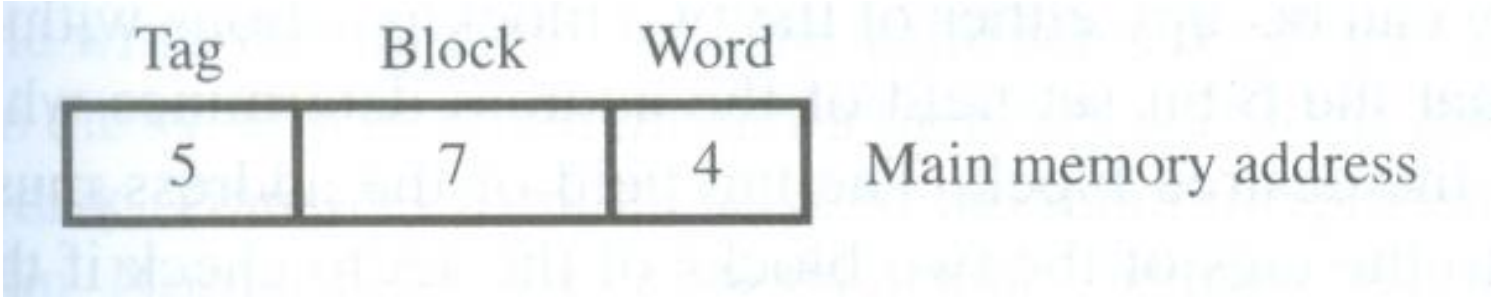

Main memory
Block 0
Block 1
Block 127
Block 128
Block 129
Block 255
Block 256
Block 257
Block 4095
Cache
tag
Block 0
tag
Block 1
tag
Block 127

Contd..

- Hence the contention….

- Uses a replacement algorithm

Placement of a memory block in cache



- 4 bit word address – addresses the required **word** in the block of memory
- 7 bit block address - addresses the cache **block** where this memory block will be stored
- 5 bit tag – Identifies one of the 32 blocks mapped on to a cache block

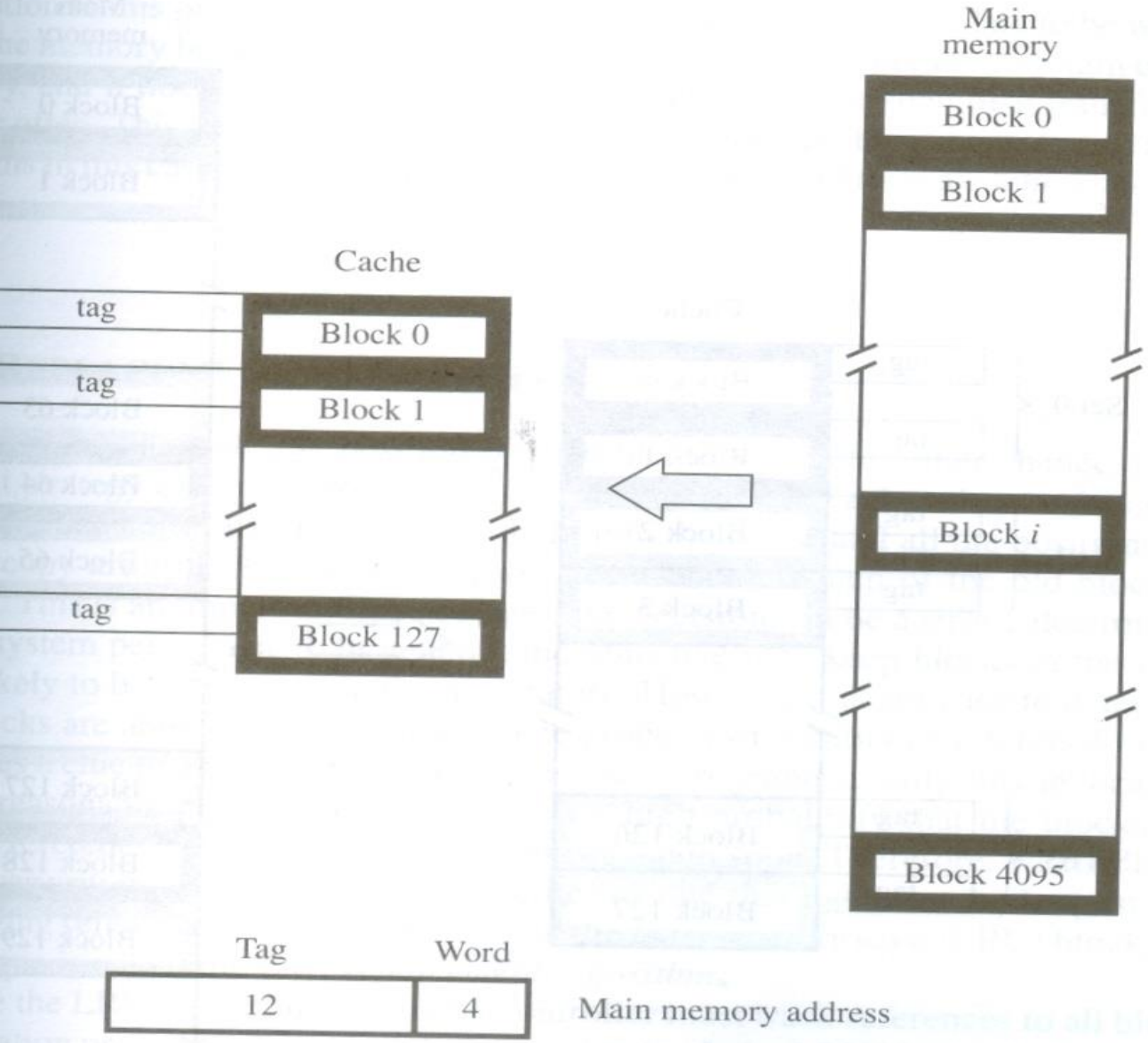
contd..

Accessing from cache:

- The 7 bits of cache block field of address generated by the processor points to a cache block
- Compare the high order 5 bits with the tag bits associated with that cache
- If they match, the desired word is in that block of the cache.
- Else read that block from MM & load into cache
- Easy to implement; very flexible

Associative Mapping

- According to this mapping a memory block can be placed into any cache block
- 12 tag bits to identify 4096 memory blocks
- The tag bits of the Addr received from the processor are compared to the tag bits of each block of the cache.
- Replacement – only if the cache is full
- Cost of searching is more



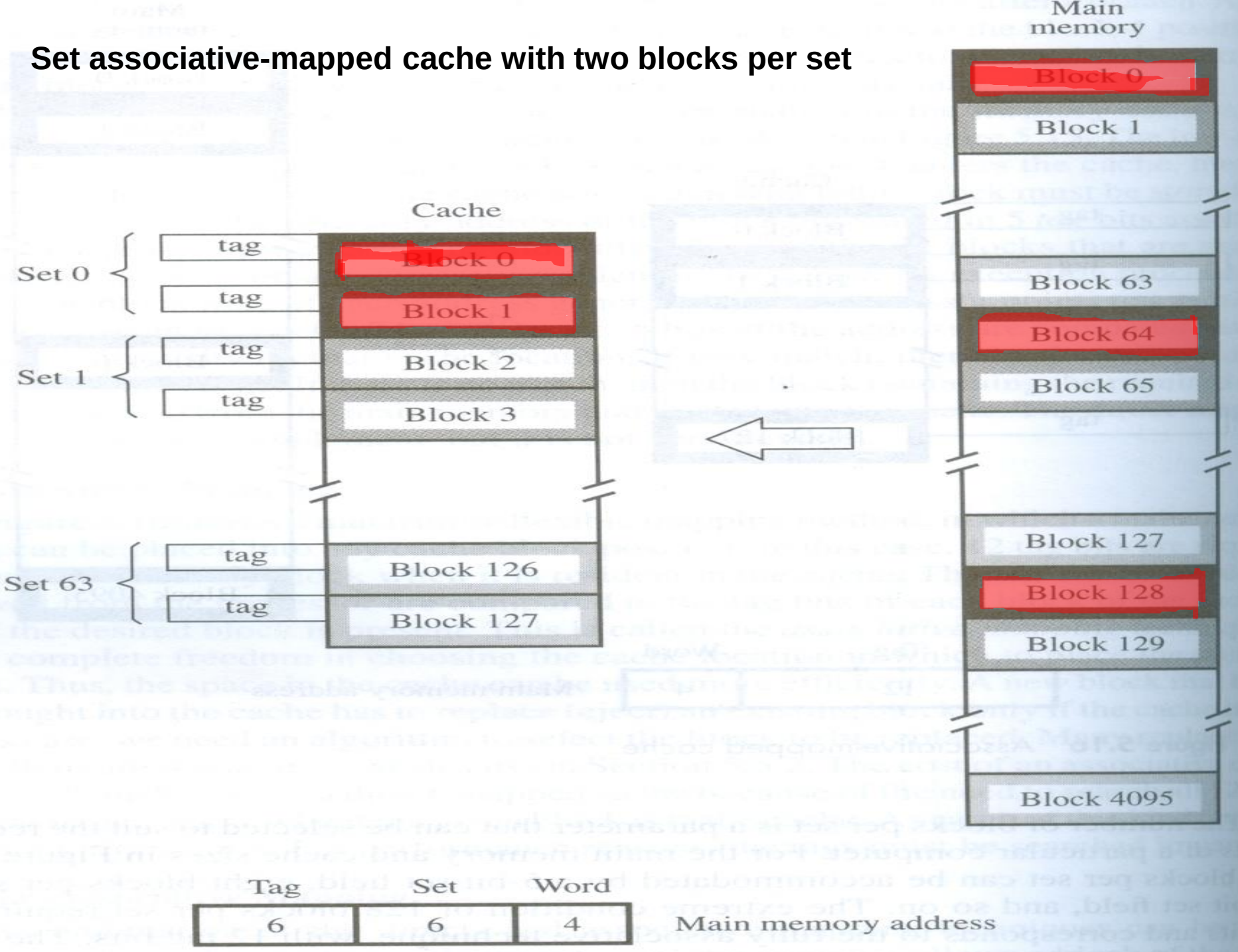
e 5.16 Associative-mapped cache.

Set Associative Mapping

- Blocks of the cache are grouped into sets
- Mapping function:

  (*block address*) **MOD** (*Number of sets in a cache*)
- A main memory block which is mapped to a set can reside in any of the blocks in the set
- Has a choice for block placement. Hence less contention.
- More h/w cost

Set associative-mapped cache with two blocks per set



Contd..

- 64 groups.
- Mapping function – j mod 64
- 0,64, 128, 192, 256… blocks of MM mapped to cache set 0. (And so on…)
- They can occupy either of the two blocks in that set
- 4096/64=64 blocks are mapped to a set . So, 6 bit tag field
- 64 sets. Hence 6 bit set field

Contd…

- Reduces the search and contention.