KTU NOTES

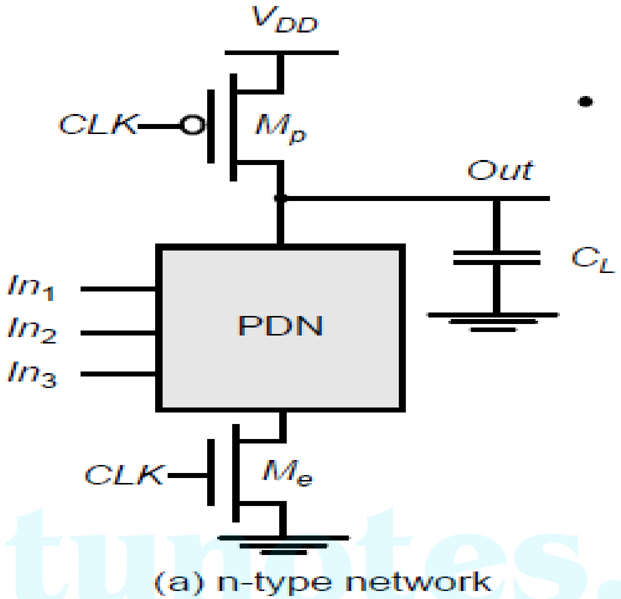
ECT 304: VLSI Circuit Design

Module 3

Dynamic Logic Design

- Static CMOS logic with a fan-in of N requires 2N devices.
- The pseudo-NMOS logic style requires only N+1 transistors to implement an N input logic gate, but unfortunately it has static power dissipation.
- An alternate logic style called dynamic logic obtains a similar result, while avoiding static power consumption.
- With the addition of clock input, it uses a sequence of ***pre-charge*** and conditional ***evaluation*** phases.

Dynamic Logic: Basic Principles



.52 Basic concepts of a dynamic gate.

- The PDN (pull-down network) is constructed exactly as in complementary CMOS.
- The operation of this circuit is divided into two major phases:
  - Precharge phase
  - Evaluation phase
- The mode of operation is determined by the clock signal 'CLK'.

Dynamic Logic: Basic Principles

- ***Precharge:***
  - When CLK = 0, the output node 'Out' is precharged to VDD by the PMOS transistor Mp.
  - During that time , the evaluate NMOS transistor Me is off, so that the pull-down path is disabled.
  - So there is no static power dissipation.

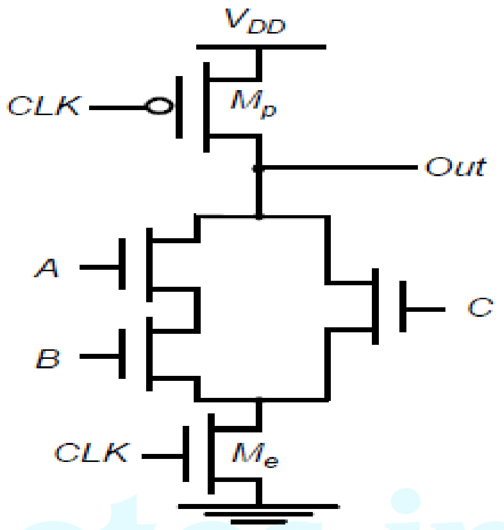
- ***Evaluation:***
  - For CLK = 1, the pre-charge transistor Mp is off, and the evaluation transistor Me is turned on.
  - The output is conditionally discharged based on the input values and the pull-down topology.
  - If the inputs are such that the PDN conducts, then a low resistance path exists between ***Out*** and ***GND*** and the output is discharged to ***GND***

Dynamic Logic: Basic Principles

- ***Evaluation:***
  - If the PDN is turned off, the pre-charge value remains stored on the output capacitance CL , which is a combination of junction capacitances, the wiring capacitance, and the input capacitance of the fan-out gates.
  - During evaluation phase, the only possible path between the output node and a supply rail is to *GND*.
  - Once ***Out*** is discharged, it cannot be charged again until the next pre-charge operation.
  - The inputs of the gate can thus make at most one transition during evaluation.
  - The output can be in the high impedance state during the evaluation period if the pull-down network is turned off. This does not happen in static CMOS.

Dynamic Logic: Basic Principles

- ***<u>Evaluation:</u>***
  - Consider an example:



(b) Example

  - During pre-charge phase (CLK = 0), the output is pre-charged to VDD, regardless of the input values, because the evaluation device is off.
  - During evaluation (CLK = 1), a conducting path is created between ‘Out’ and the ‘GND’, if (and only if) A.B+C is TRUE.
  - Otherwise, the output remains at the pre-charged state of VDD.
  - The following function is thus realised :

$$Out = \overline{CLK} + \overline{(A.B + C)}.CLK$$

Dynamic Logic: Advantages

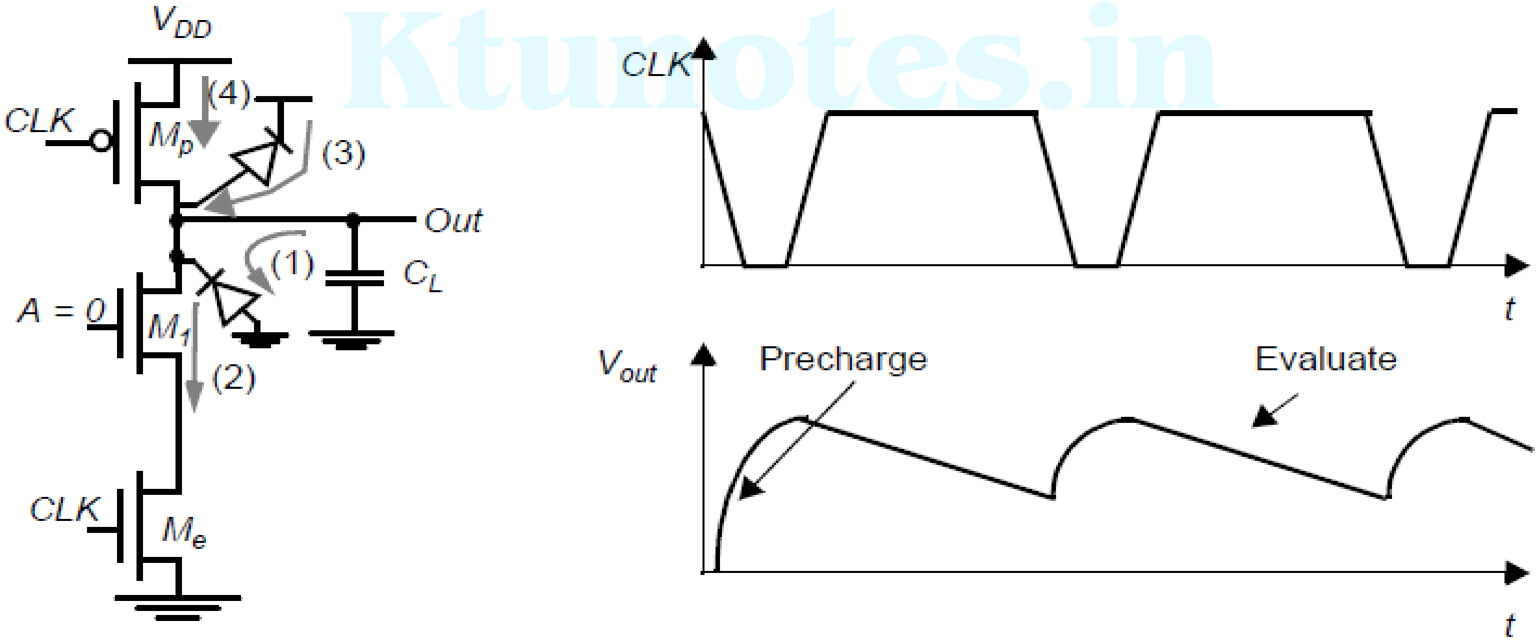
- The number of transistors required for a circuit with fan-in N is ***(N+2),*** in contrast to ***2N*** in case of static CMOS circuits.
- It is non-ratioed. i.e, the sizing of PMOS pre-charge device is not important for realising proper functionality of the gate.
- The load capacitance is about ***50%***less than static CMOS and is closer to nMOS circuits. Thus the speed of operation is faster than that of the static CMOS circuits.
- Lower static power dissipation.
- No short circuit power dissipation.
- No glitching power dissipation.

Dynamic Logic: Disadvantages

- Charge Leakage problem
- Charge sharing problem
- Clock Skew problem
- Reduction in voltage levels in cascaded blocks

Dynamic Logic: Charge Leakage

- The operation of a dynamic gate depends on the storage of information in the form of charge on the MOS capacitors.
- But this charge gradually leaks away due to leakage currents, eventually resulting in a malfunctioning of the gate.



(a) Leakage sources

(b) Effect on waveforms

Figure 6.56 Leakage issues in dynamic circuits.

Dynamic Logic: Charge Leakage

- Source 1 and 2 are the ***reverse-biased diode*** and the ***sub-threshold leakage*** of the NMOS pull-down device M1, respectively.
- Source 3 and 4 are the ***reverse-biased diode*** and the ***sub-threshold leakage*** of the PMOS pre-charge device Mp, respectively.
- ***Solution:***
  - Use of higher threshold voltage transistors.
  - Use weak PMOS (low W/L) transistors as pull-up device (Bleeder Transistor).

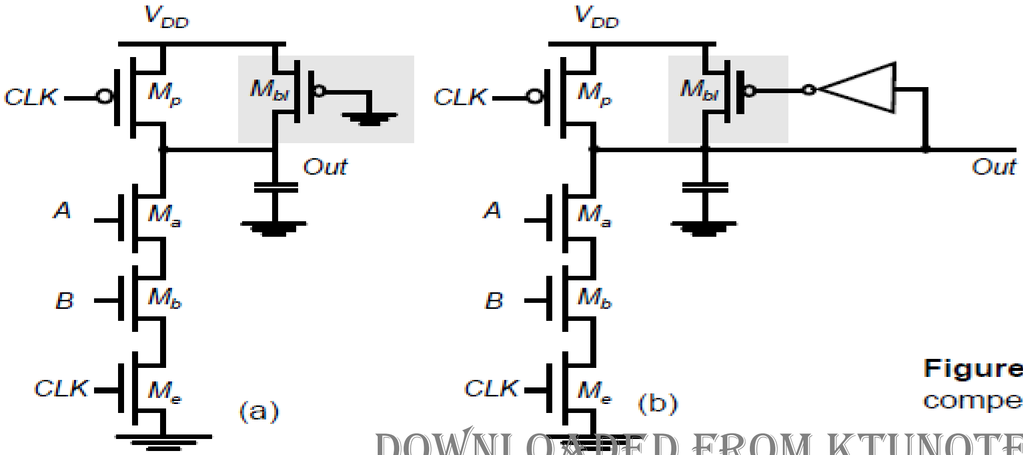
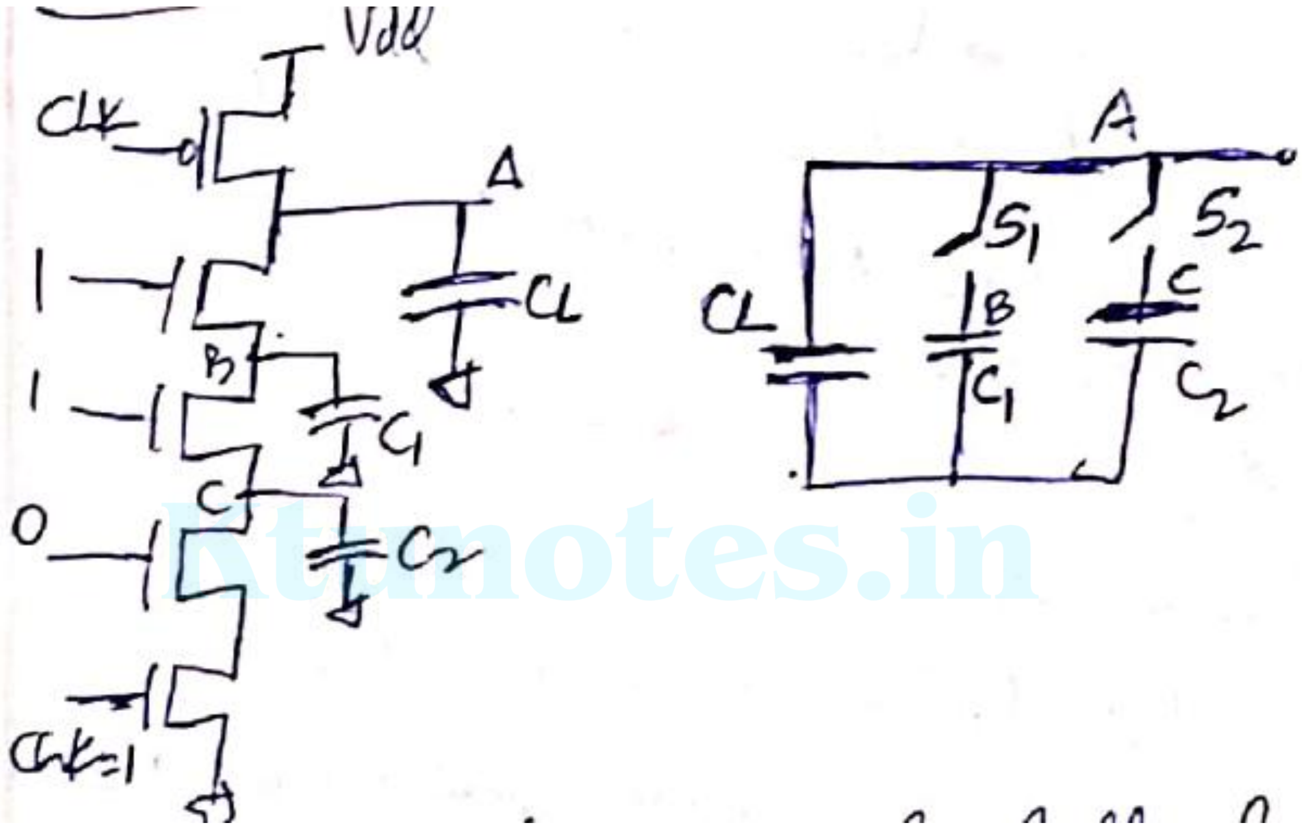


Figure 6.58 Static bleeders compensates for the charge-leakage.

Dynamic Logic: Charge Sharing



- Before the switches are closed, the charge on CL is given by QA = VDD.CL and charges at node B and C are QB = 0 and QC = 0.

Dynamic Logic: Charge Sharing

- After the switches are closed, there will be redistribution of charges based on charge conservation principles and the voltage VA at node A is given by,

  CL. VDD = (CL + C1 + C2)VA.

  i.e., VA = (CL/ (CL + C1 + C2). VDD

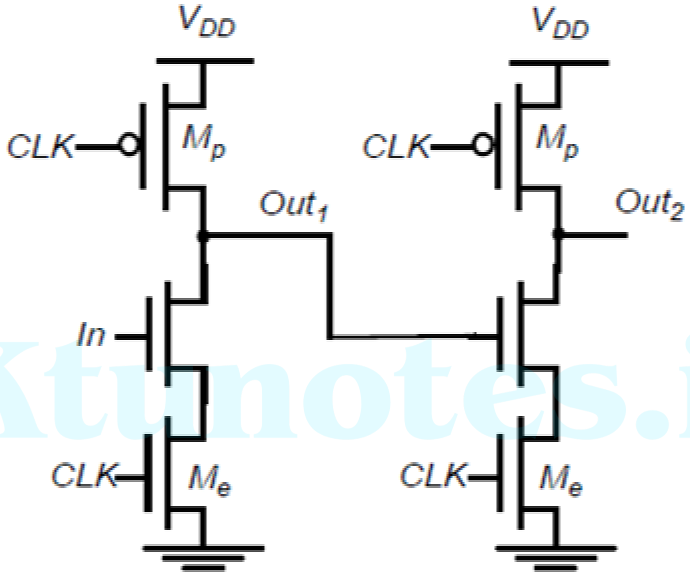
- ***Solution:***
- A weak PMOS is added as pull-up transistor. The transistor always remains ON and behaves like a pseudo-NMOS circuit during evaluation.

Dynamic Logic: Charge Sharing

- ***Solution:***
- Although there is static power dissipation due to this, during the evaluation phase, it helps to maintain the voltage by replenishing the charge loss due to leakage current or charge sharing .

Dynamic Logic: Clock Skew

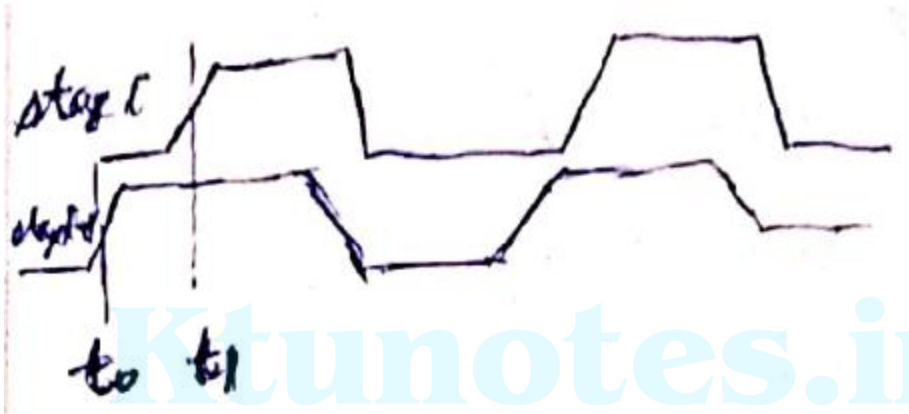
- It occurs when multi-stages are used (cascading) .



- Clock skew problem arises because of delay due to resistance and parasitic capacitances associated with the wire that carry the clock pulse and this delay is approximately proportional to the square of the wire length.

Dynamic Logic: Clock Skew

- Stage ‘i’ is in pre-charge phase and stage ‘i+1’ in evaluation phase.



- Clock skew results in hazards and race conditions.
- This problem can be overcome, if the output can be set to low during pre-charge.

Dynamic Logic: Cascading Issues

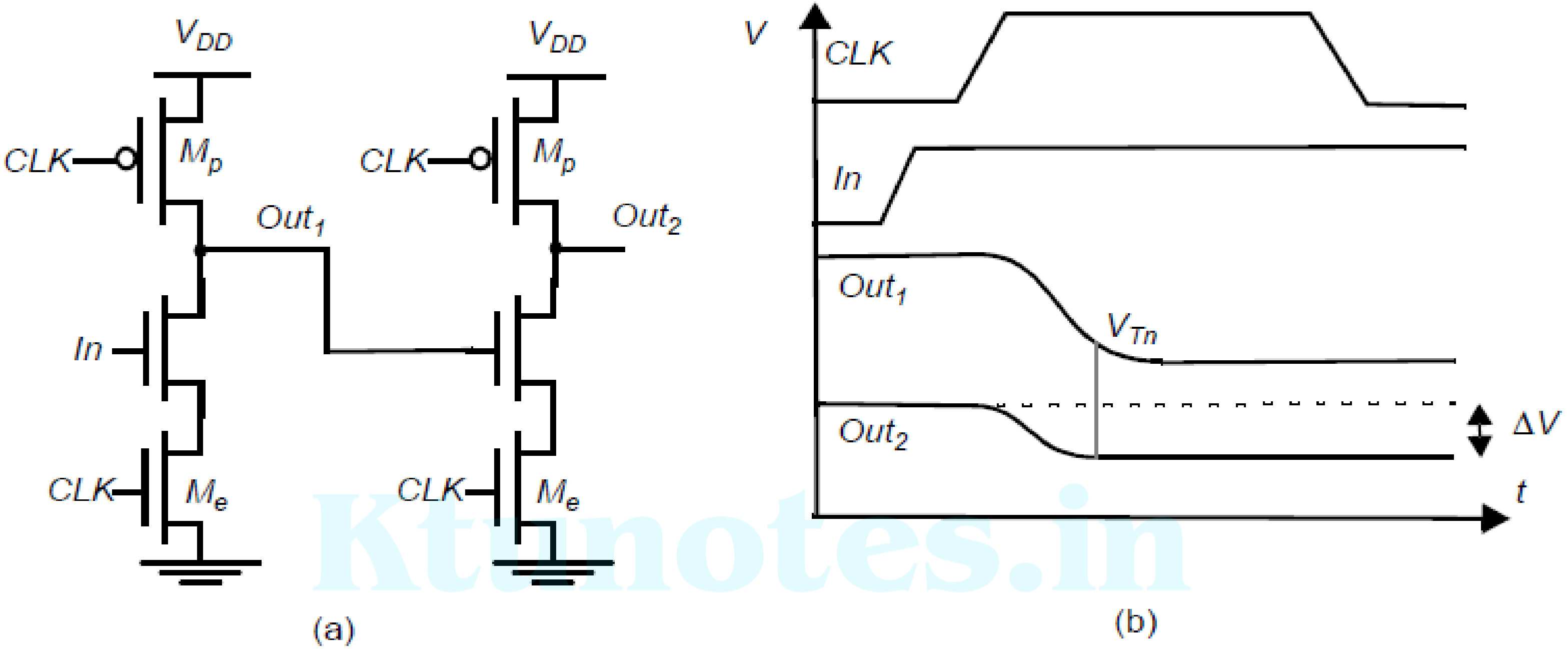


Figure 6.64 Cascade of dynamic *n*–type blocks.

- During pre-charge phase, the outputs of both inverters are pre-charged to VDD.
- Assume that the primary input '*In*' makes a 0 to 1transition .

Dynamic Logic: Cascading Issues

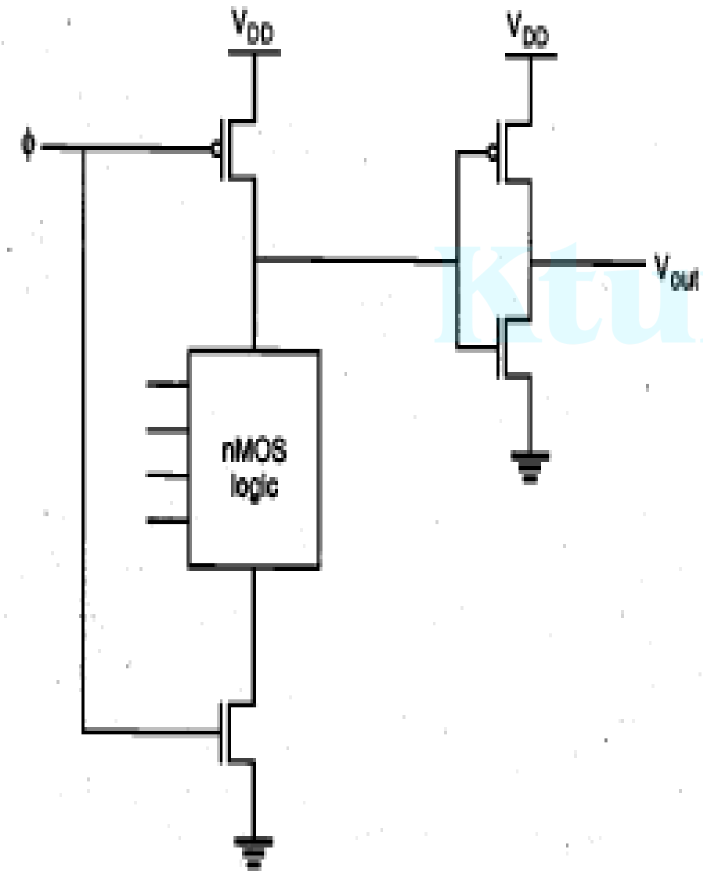
- On the rising edge of the clock, output ‘*Out1*’ starts to discharge.
- The second output should remain in the pre-charged state of VDD as its expected value is 1 (*Out1* transitions to 0 during evaluation).
- Due to the propagation delay for the input to discharge *Out1* to *GND*, the second output also starts to discharge.
- As long as the *Out1*, exceeds the switching threshold of the second gate (approx. VT,n), a conducting path exists between *Out2* and *GND*, and charge is lost at *Out2*.
- Thus *Out2* can be at an intermediate voltage. This leads to reduced noise margins and potential malfunctioning.

Dynamic Logic: Cascading Issues

- The cascading problem arises because outputs of the each gate – and hence the inputs of the next stages – are pre-charged to 1.
- The solution for cascading problems is to set all inputs to 0 during pre-charge phase.

Domino Logic

- It consists of an n-type dynamic logic block followed by a static inverter.



Generalized circuit diagram of a domino CMOS logic gate.

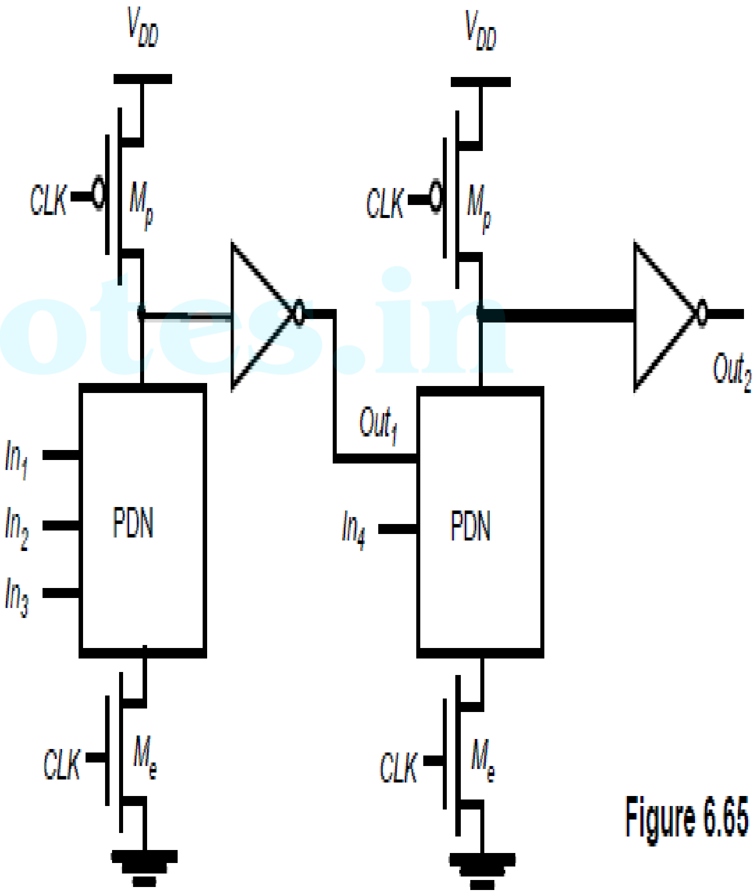


Figure 6.65 DOMINO CMOS logic.

Domino Logic

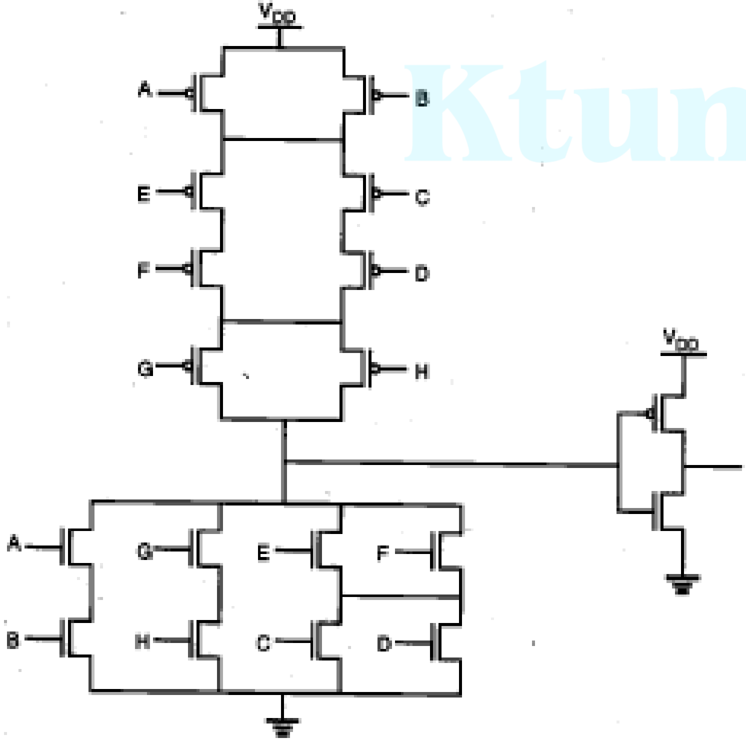
- During pre-charge, the output of the n-type dynamic gate is charged up to VDD and the output is set to 0.
- During evaluation, the dynamic gate conditionally discharges, and the output of the inverter makes a conditional transition from 0 →1.
- Regardless of the input voltages applied to the dynamic CMOS stage, it is not possible for the buffer output to make a 1→0 transition during the evaluation phase.
- Consider a chain of Domino gates. During pre-charge, all inputs are set to 0.

Domino Logic

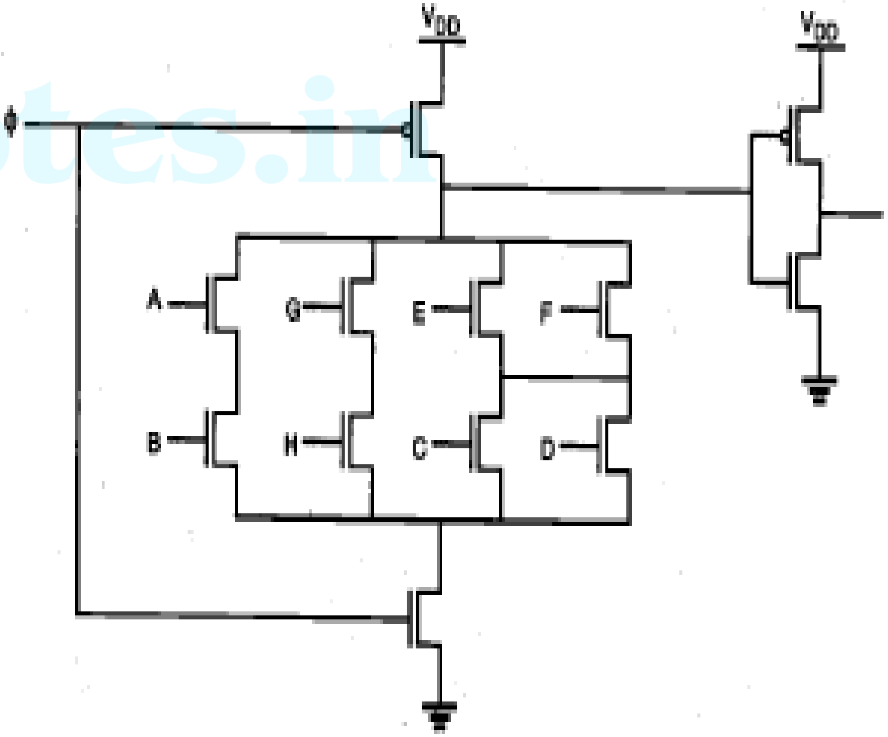
- During evaluation, the output of the first Domino block either stays at 0 or makes a 0→1 transition, affecting the second gate.
- This effect might ripple through the whole chain, one after the other, similar to a line of falling dominoes – hence the name.

Domino Logic

- Domino CMOS gates allow a significant reduction in the number of transistors required to realise any complex Boolean functions.
- Consider the function Z = AB + (C+D) (E+F) + GH.



Conventional CMOS



Domino CMOS

Domino Logic - Advantages

- Lower power consumption.
- Reduced chip area.
- Higher speed of operation (only rising edge delay)
- No short-circuit power dissipation.
- No glitching power dissipation.

Domino Logic - Limitations

- Each gate requires an inverting buffer.
- All the gates are non-inverting in nature.
- Higher switching activity.

Domino Logic – Charge Sharing

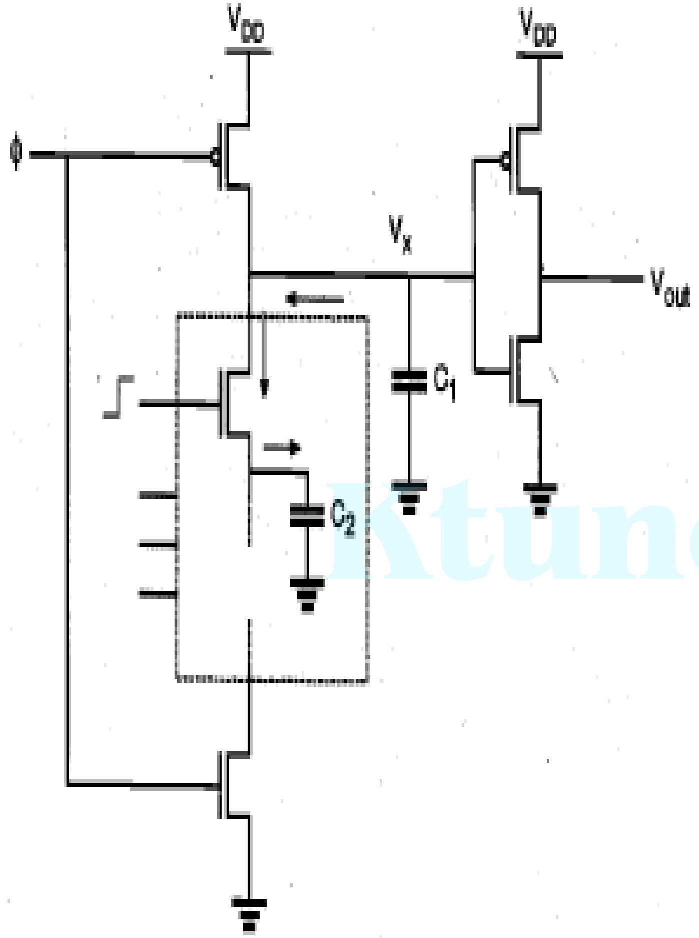


Figure 9.32. Charge sharing between the output capacitance C_1 and an intermediate node capacitance C_2 during the evaluation cycle may reduce the output voltage level.

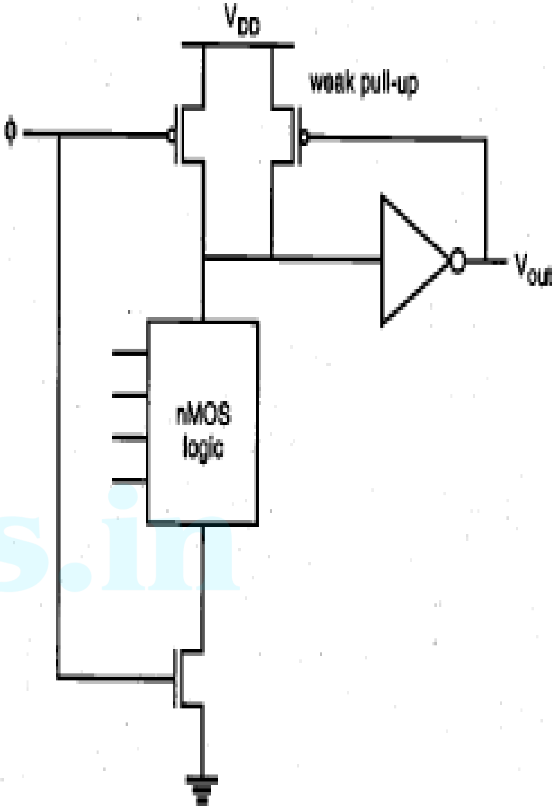
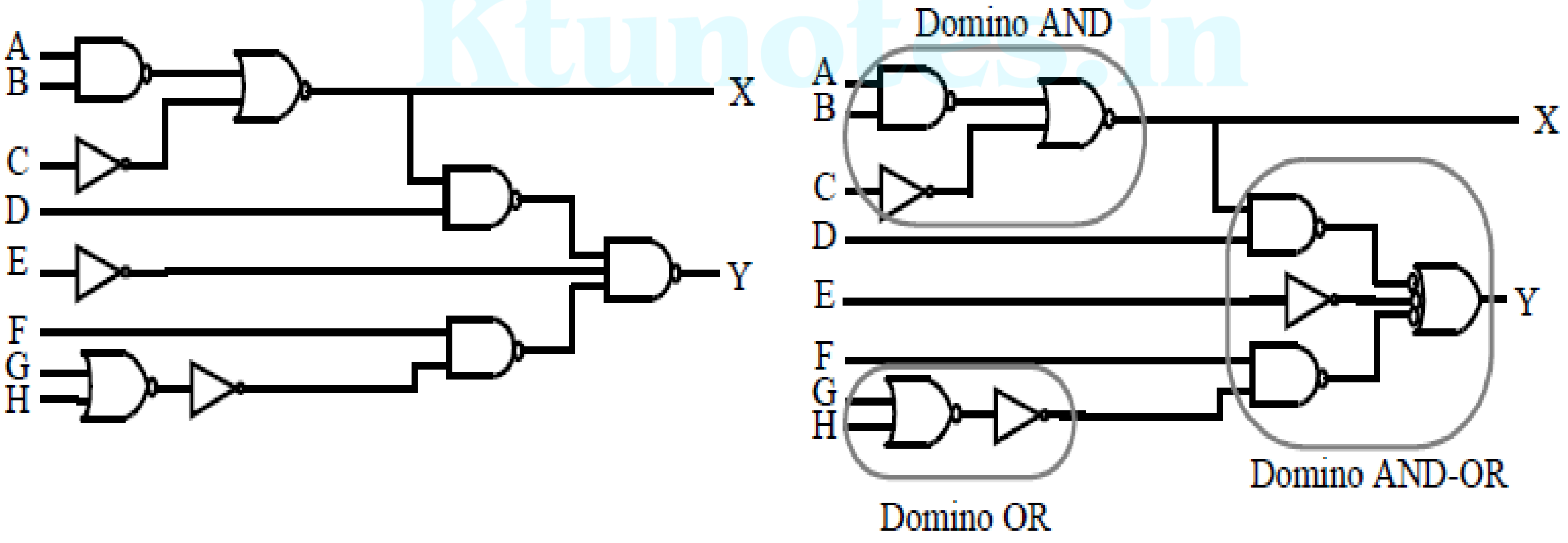


Figure 9.33. A weak pMOS pull-up device in a feedback loop can be used to prevent the loss of output voltage level due to charge sharing.

Domino Logic

- ***Dealing with Non-Inverting Property of Domino Logic:***
  - A major limitation in Domino Logic is that, only non-inverting logic can be implemented.
  - This can be dealt with – reorganising the logic using simple Boolean transforms such as De Morgan’s Law. But , it is not always possible .



(a) before logic transformation

(b) after logic transformation

Figure 6.67 Restructuring logic to enable implementation using non-inverting Domino Logic.

Domino Logic - Variations

- Different types of Dominos are – ***Multiple Output Domino Logic, Compound Domino*** etc.

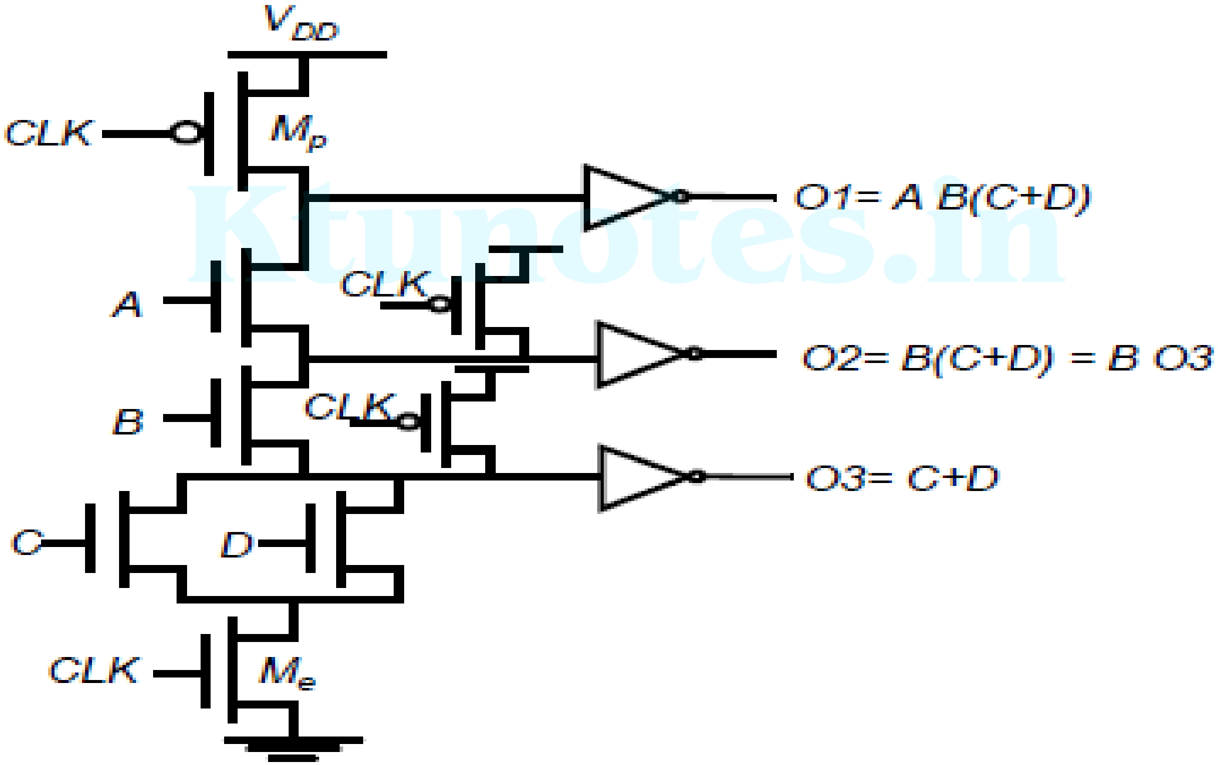


Figure 6.69 Multiple output Domino

Domino Logic - Variations

- O1 = (ABC)’ ; O2 = (DEF)’ and O3 = (GH)’ and
- O = ABCDEF + GH

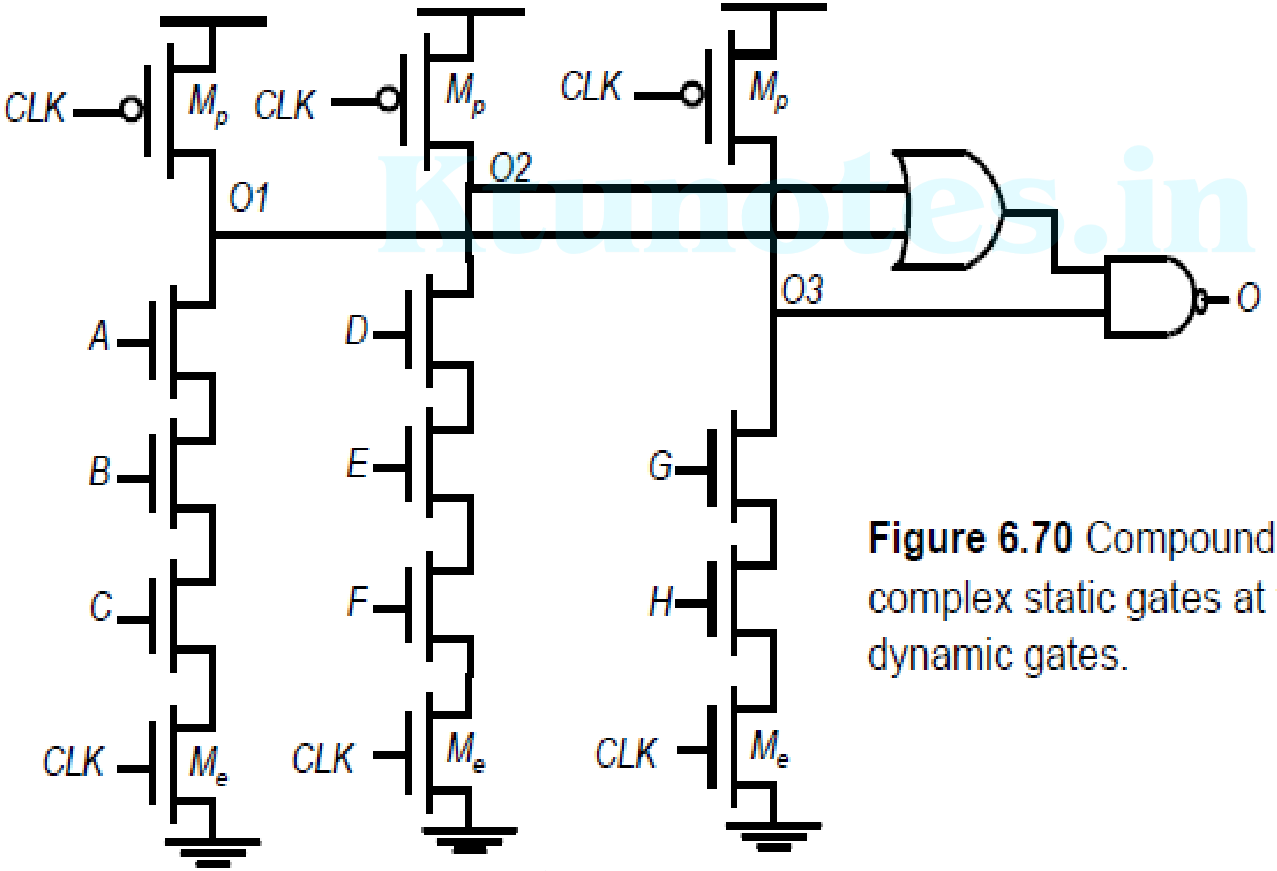


Figure 6.70 Compound Domino logic uses complex static gates at the output of the dynamic gates.

NP – Domino or NORA CMOS Logic

- In domino CMOS logic gates, all logic operations are performed by the nMOS transistors acting as pull-down networks, while the role of pMOS transistors is limited to pre-charging the dynamic nodes.
- As an alternative and a complement to nMOS-based domino CMOS logic, we can construct dynamic logic gates using pMOS transistors as well.
- NP – CMOS consists of alternating nMOS and pMOS stages.

NP – Domino or NORA CMOS Logic

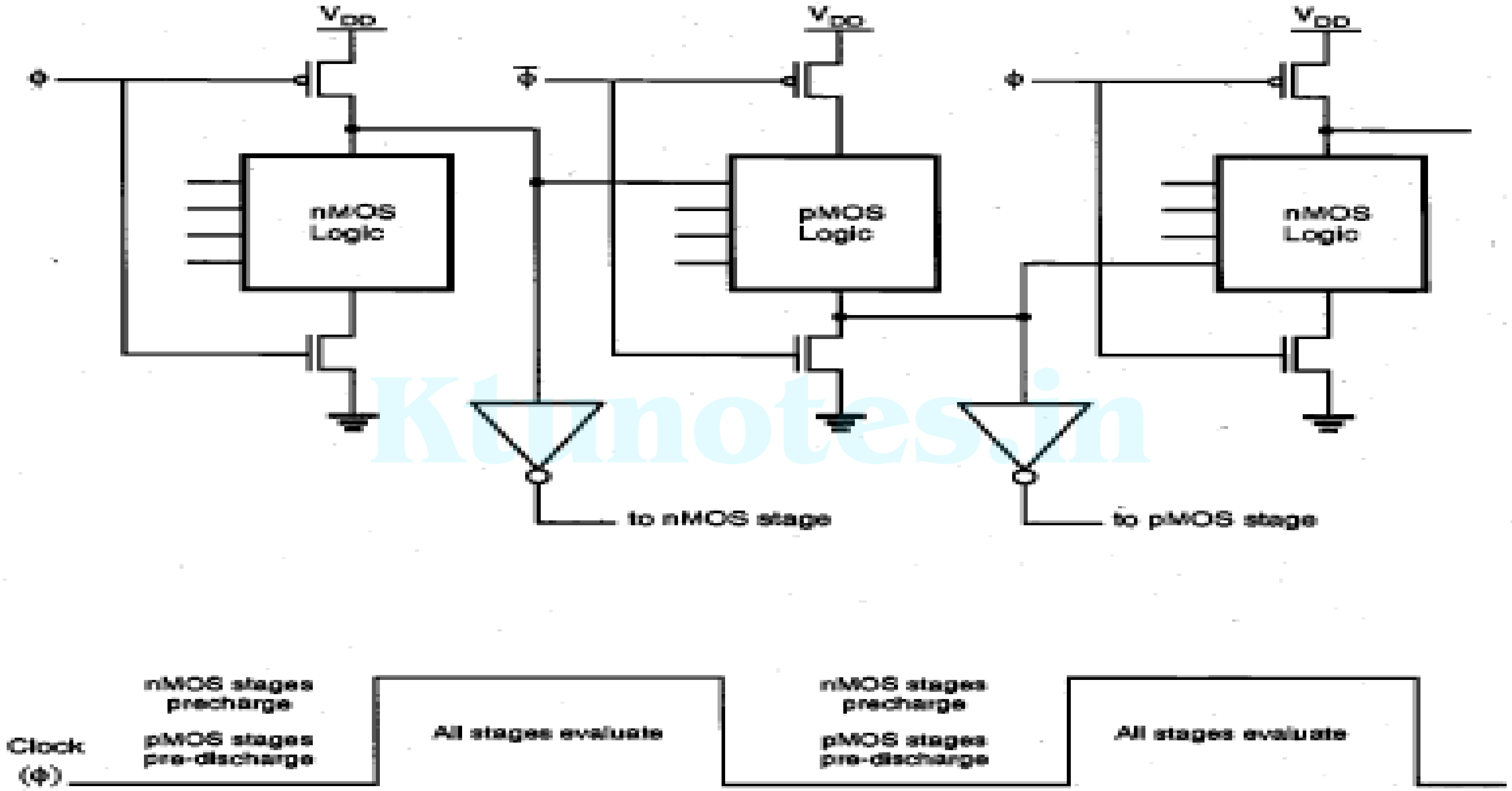


Figure 9.37. NORA CMOS logic consisting of alternating nMOS and pMOS stages, and the scheduling of precharge/evaluation phases.

NP – Domino or NORA CMOS Logic

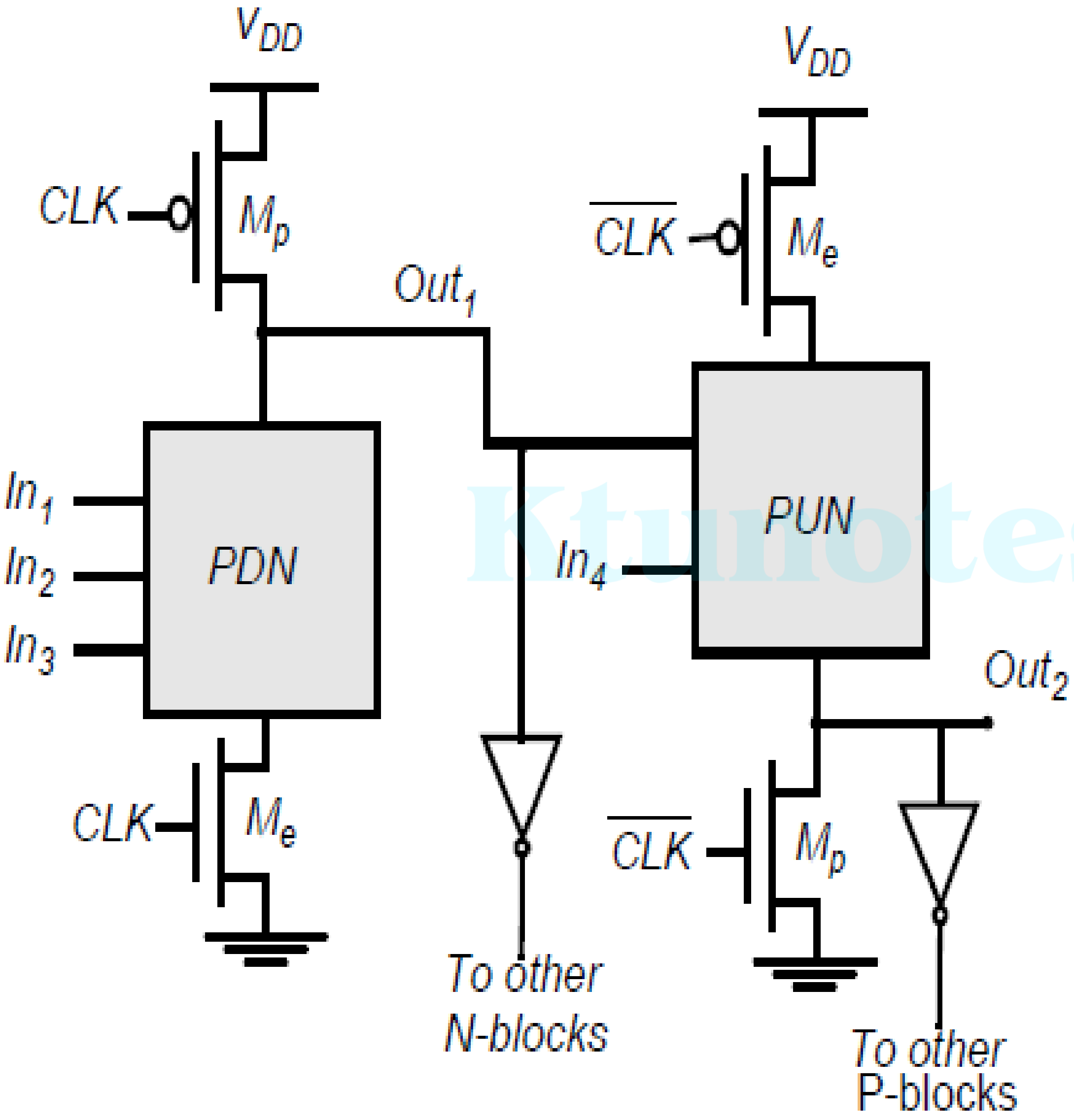


Figure 6.71 The *np*-CMOS logic circuit style.

NP – Domino or NORA CMOS Logic

- The pre-charge and evaluate timing of nMOS logic stage is accomplished by the clock signal CLK (ϕ), whereas the pMOS logic stages are controlled by the inverted clock signal (CLK)’ or (ϕ)’.
- ***<u>Working:</u>***
- When the clock signal is low, the output nodes of nMOS logic blocks are pre-charged to VDD through the pMOS pre-charge transistors.
- Whereas, the output nodes of pMOS logic blocks are pre-discharged to 0V through the nMOS discharge transistors driven by (ϕ)’.

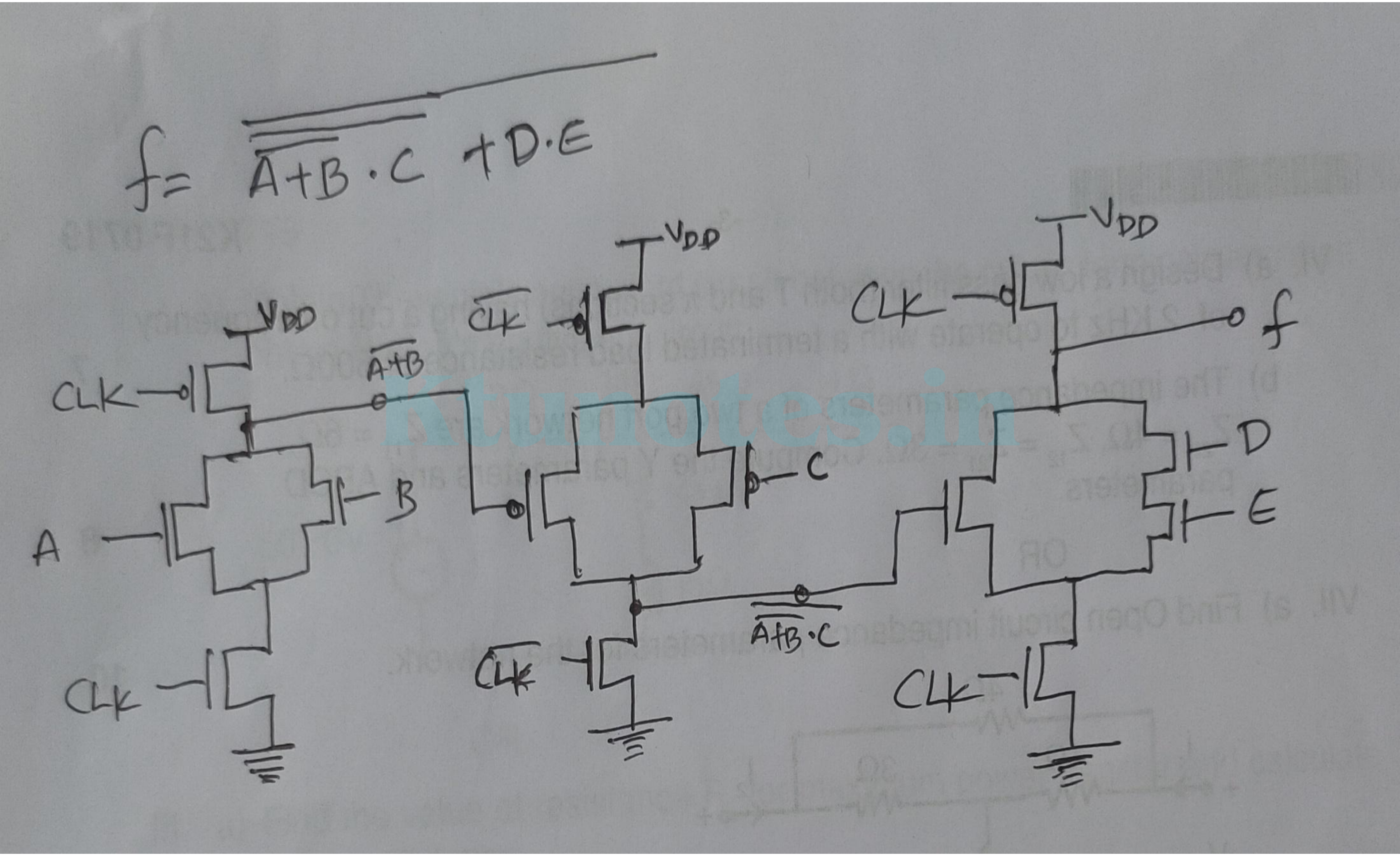
NP – Domino or NORA CMOS Logic

- ***Working:***
- When the clock signal makes a low-to-high transition (Inverted clock signal makes a high-to-low transition at the same time), all cascaded nMOS and pMOS logic stages evaluate one after the other.

NP – Domino or NORA CMOS Logic

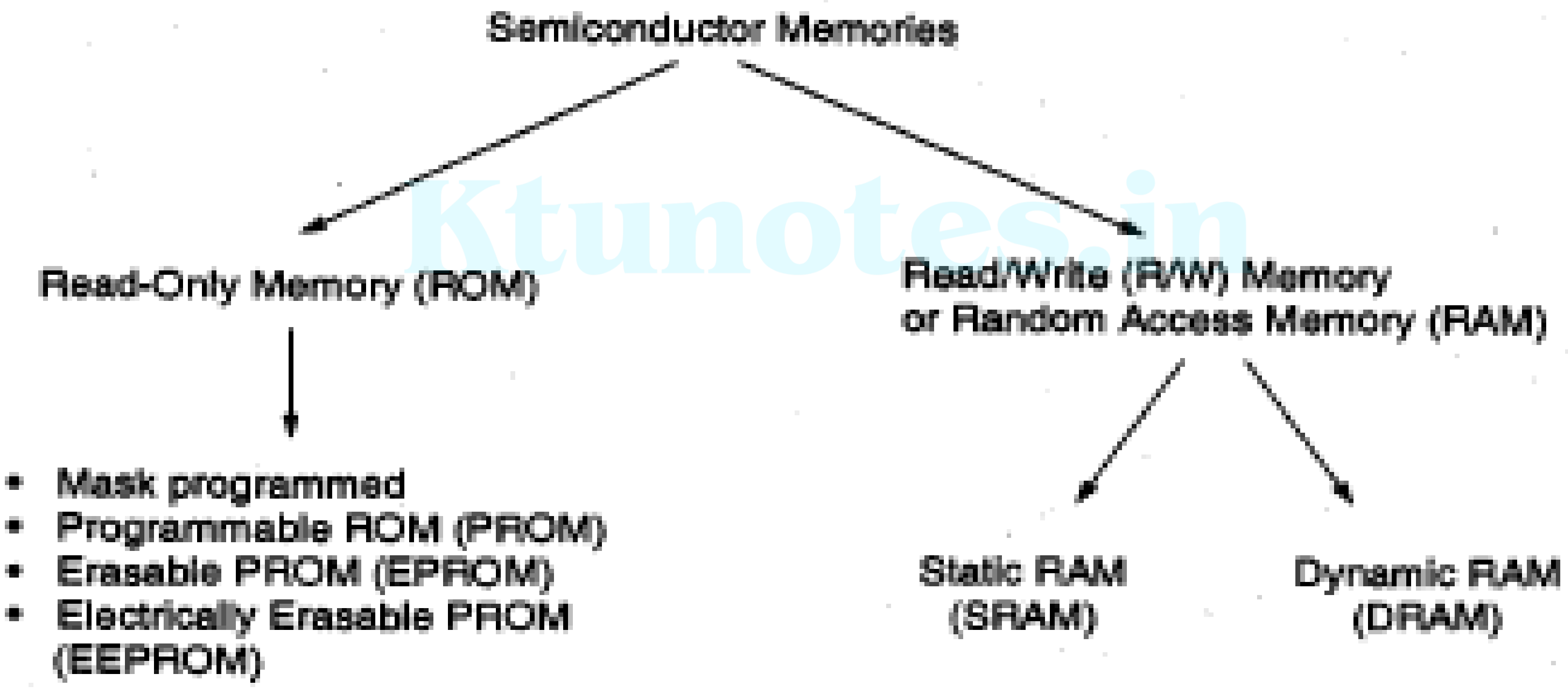
- ***Advantages:***
- A static CMOS inverter is not required at the output of every dynamic logic stage.
- NORA logic allows pipelined system architecture.
- ***Disadvantages:***
- Charge sharing problem.
- Charge leakage problem.
- P-tree blocks are slower than the n-tree modules, due to lower current drive of the pMOS transistors in the logic network.

NP – Domino or NORA CMOS Logic


$f = \overline{\overline{\overline{A+B}\cdot C} + D\cdot E}$
VDD
CLK
$\overline{A+B}$
A
B
CLK
VDD
$\overline{CLK}$
C
$\overline{CLK}$
$\overline{\overline{A+B}\cdot C}$
VDD
CLK
f
D
E
CLK

Semiconductor Memories

- An overview of the different memory types and their classifications can be shown as:



Semiconductor Memories

Read-Only Memory (ROM)

- Mask programmed
- Programmable ROM (PROM)
- Erasable PROM (EPROM)
- Electrically Erasable PROM (EEPROM)

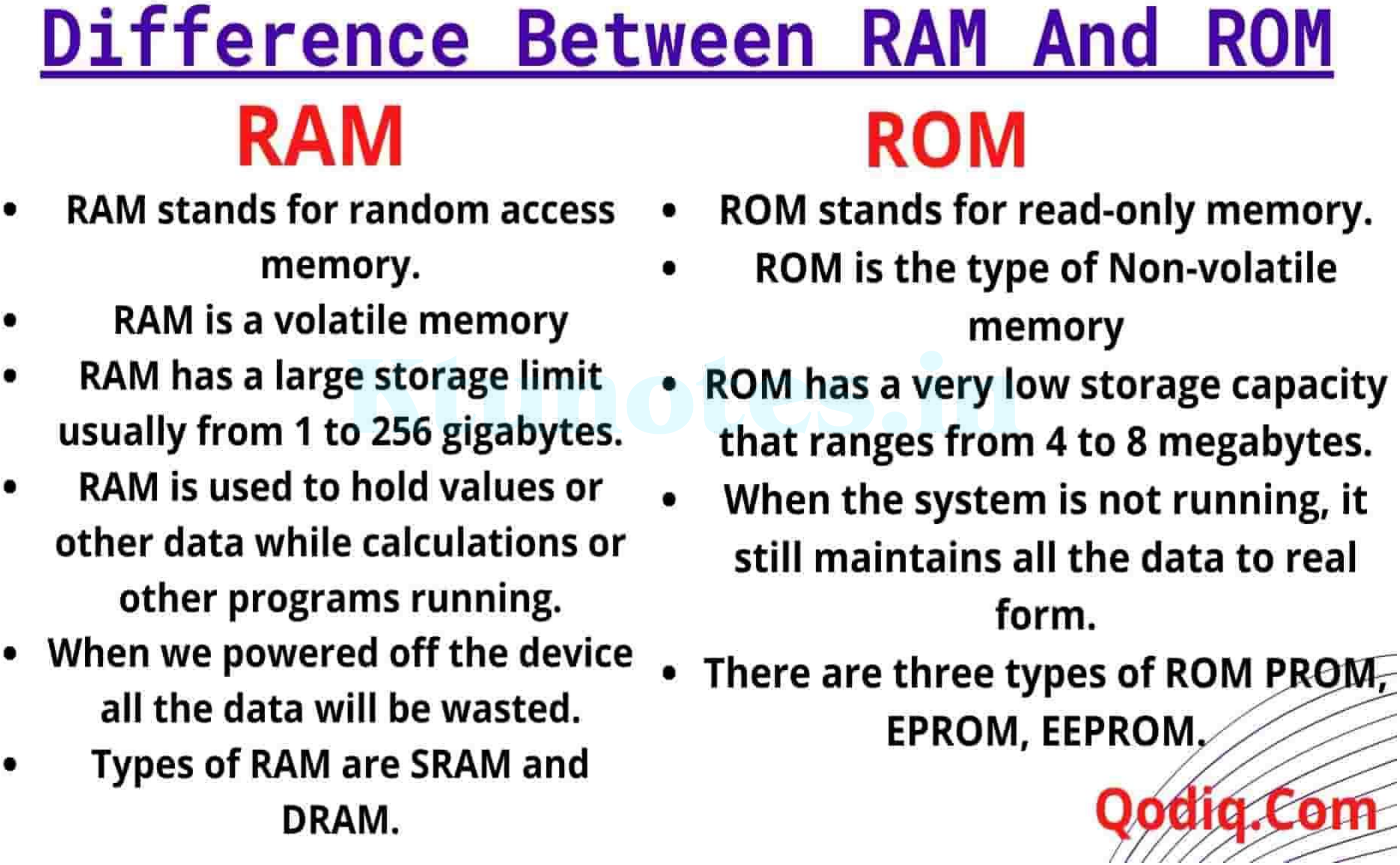
Read/Write (R/W) Memory or Random Access Memory (RAM)

Static RAM (SRAM)

Dynamic RAM (DRAM)

gure 10.1. Overview of semiconductor memory types.

Semiconductor Memories



Difference Between RAM And ROM

RAM

- RAM stands for random access memory.
- RAM is a volatile memory
- RAM has a large storage limit usually from 1 to 256 gigabytes.
- RAM is used to hold values or other data while calculations or other programs running.
- When we powered off the device all the data will be wasted.
- Types of RAM are SRAM and DRAM.

ROM

- ROM stands for read-only memory.
- ROM is the type of Non-volatile memory
- ROM has a very low storage capacity that ranges from 4 to 8 megabytes.
- When the system is not running, it still maintains all the data to real form.
- There are three types of ROM PROM, EPROM, EEPROM.

Qodiq.Com

Semiconductor Memories

- A typical memory array organisation is as below:

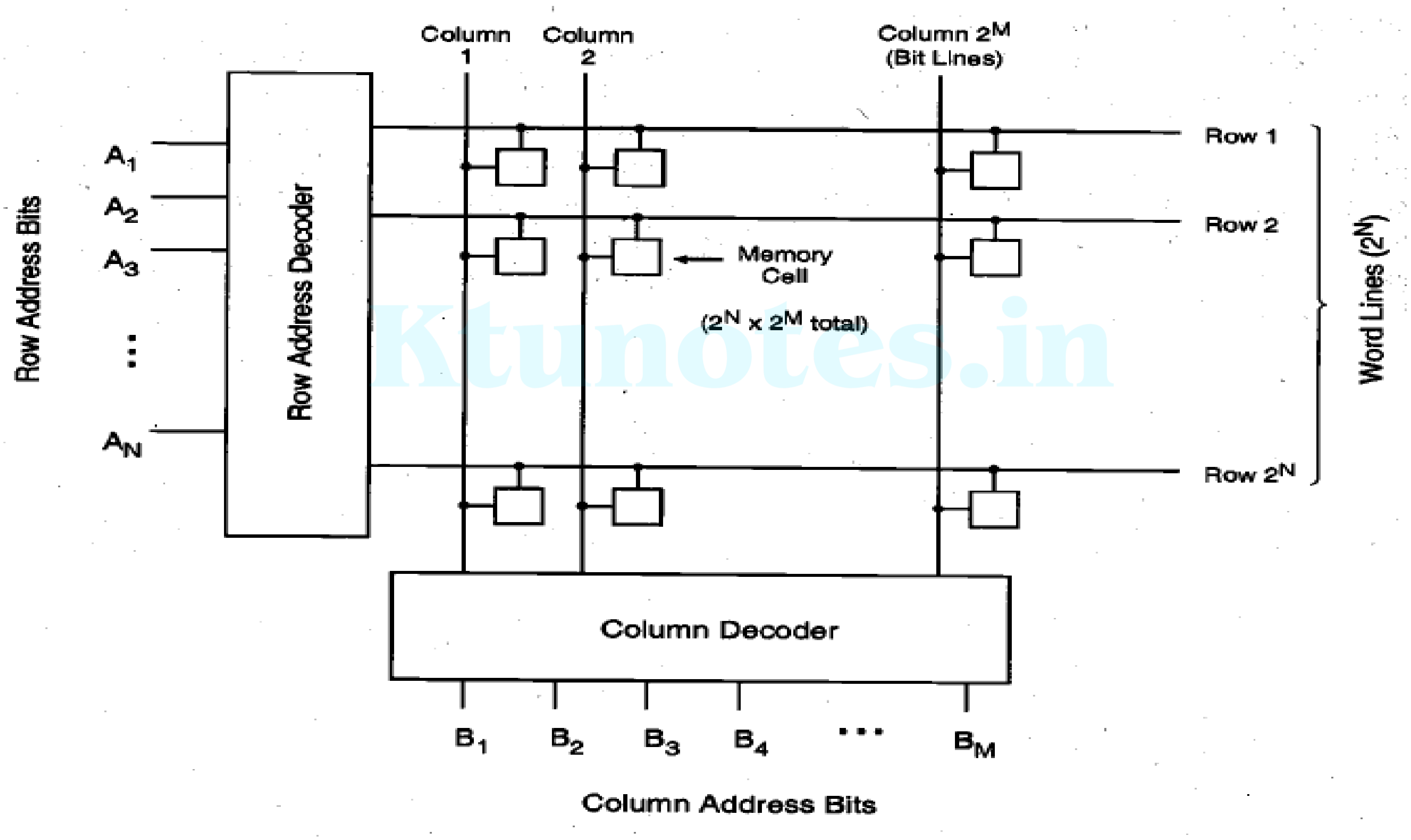


Figure 10.2. **Typical random-access memory array organization.**

Semiconductor Memories

- The data storage structure, or core consists of individual memory cells arranged in an array of horizontal rows and vertical columns.
- Each cell is capable of storing ***one bit of binary information***.
- Each memory cell shares a common connection with the other cells in the same row, and another common connection with the other cells in the same column.
- There are 2^N rows, also called ***word lines***, and 2^M columns, also called ***bit lines***. Thus the total number of memory cells in this array is $2^M x 2^N$.

Semiconductor Memories

- To access a particular memory cell, i.e., a particular data bit in this array, the corresponding *bit line* and the corresponding *word line* must be activated (selected).
- The row and column selection operations are accomplished by row and column decoders., respectively.
- The row decoder circuit selects one out of 2^N word lines according to an N-bit row address, while the column decoder circuit selects one out of 2^M bit lines according to an M-bit column address.

Semiconductor Memories

- Once a memory cell or a group of memory cells are selected, a data read and/or a data write operation may be performed on the selected single bit or multiple bits on a particular row.
- The column decoder circuit serves the double duties of selecting the particular columns and routing the corresponding data content in a selected row to the output.
- Here, the individual memory cells can be accessed for data read and/or data write operations in random order, independent of their physical locations in the memory array. Thus called ***Random Access Memory (RAM).***

Read Only Memory (ROM)

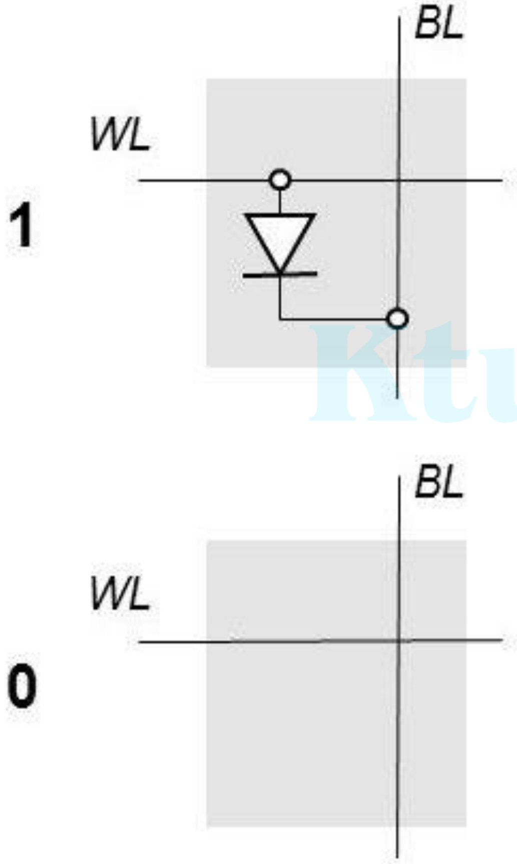
- Read only memories are used to store constants, control information and program instructions in digital systems.
- It is considered as components that provide a fixed, specified binary output for every binary input.
- It is a simple combinational Boolean network, which produces a specified output value for each input combination, i.e., for each address.
- Thus storing binary information at a particular address location can be achieved by the presence or absence of a data path from the selected row (word line) to the selected column (bit line).

Read Only Memory (ROM)

- ROM holds programs and data permanently even when the computer is switched off.
- Data can be read by the CPU in any order, so ROM is also direct access.
- The contents of ROM are fixed at the time of manufacture.

Approaches for implementing ROM cells

- ***Diode ROM:***



Diode ROM

• Bit line (BL) is clamped to ground
•WL enabled – diode turn on
•Results value 1 on bit line

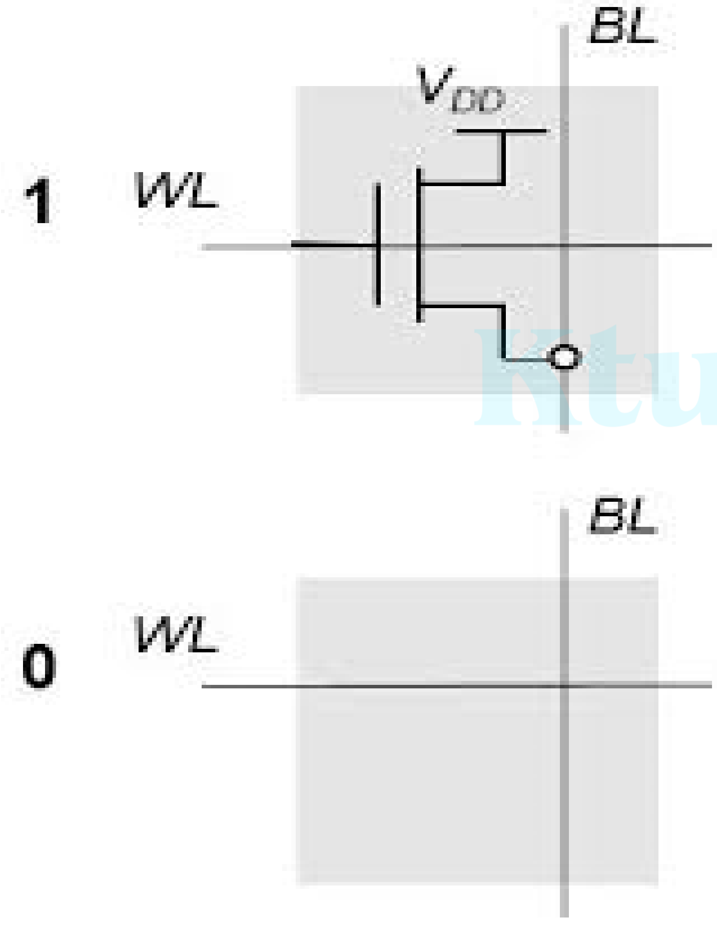
•No physical connection between WL & BL
•Value on BL is low independent of value on WL

•Presence or absence of diode determines the value on ROM cell
•Presence - cell storing 1
•Absence - cell storing 0

Disadvantages:
•It does not isolate bit line (BL) from word line(WL).
•It is only used in small memories.

Approaches for implementing ROM cells

- ***MOS ROM 1:***



MOS ROM 1

•Diode is replaced by gate-source connection of NMOS transistor, whose drain is connected to supply voltage.

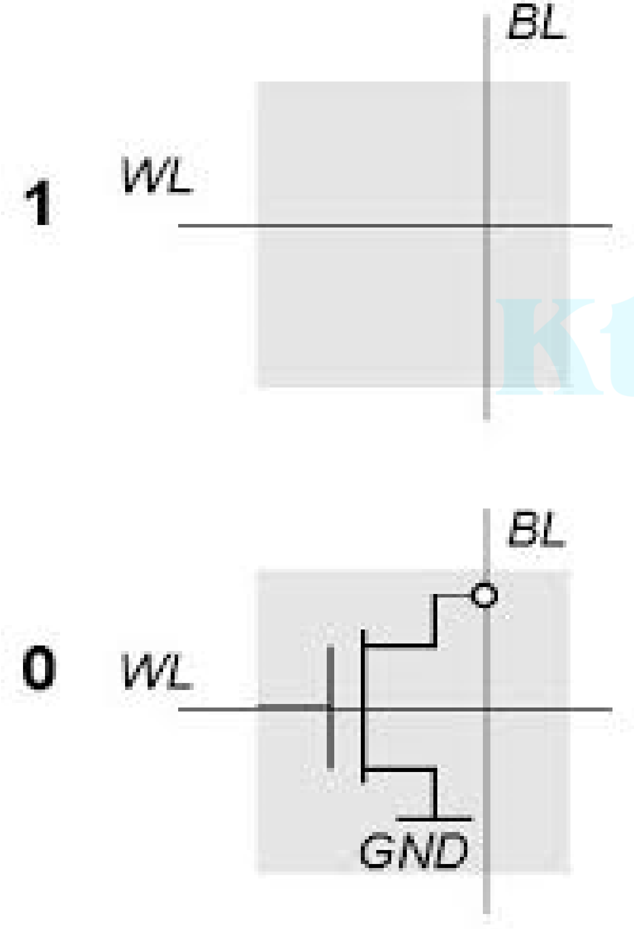
•Working is same as diode ROM

•Output current is provided by MOS transistor in the cell.

Disadvantages:
•More complex cell.
•It needs large area.
•Extra supply contact must be distributed to every cell.

Approaches for implementing ROM cells

- ***MOS ROM 2:***



MOS ROM 2

•Bit line is resistively clamped to the supply voltage, or the default value of the output must equal 1.

•Presence or absence of transistor determines the value on ROM cell

•Presence - cell storing 0

•Absence - cell storing 1

Disadvantages:

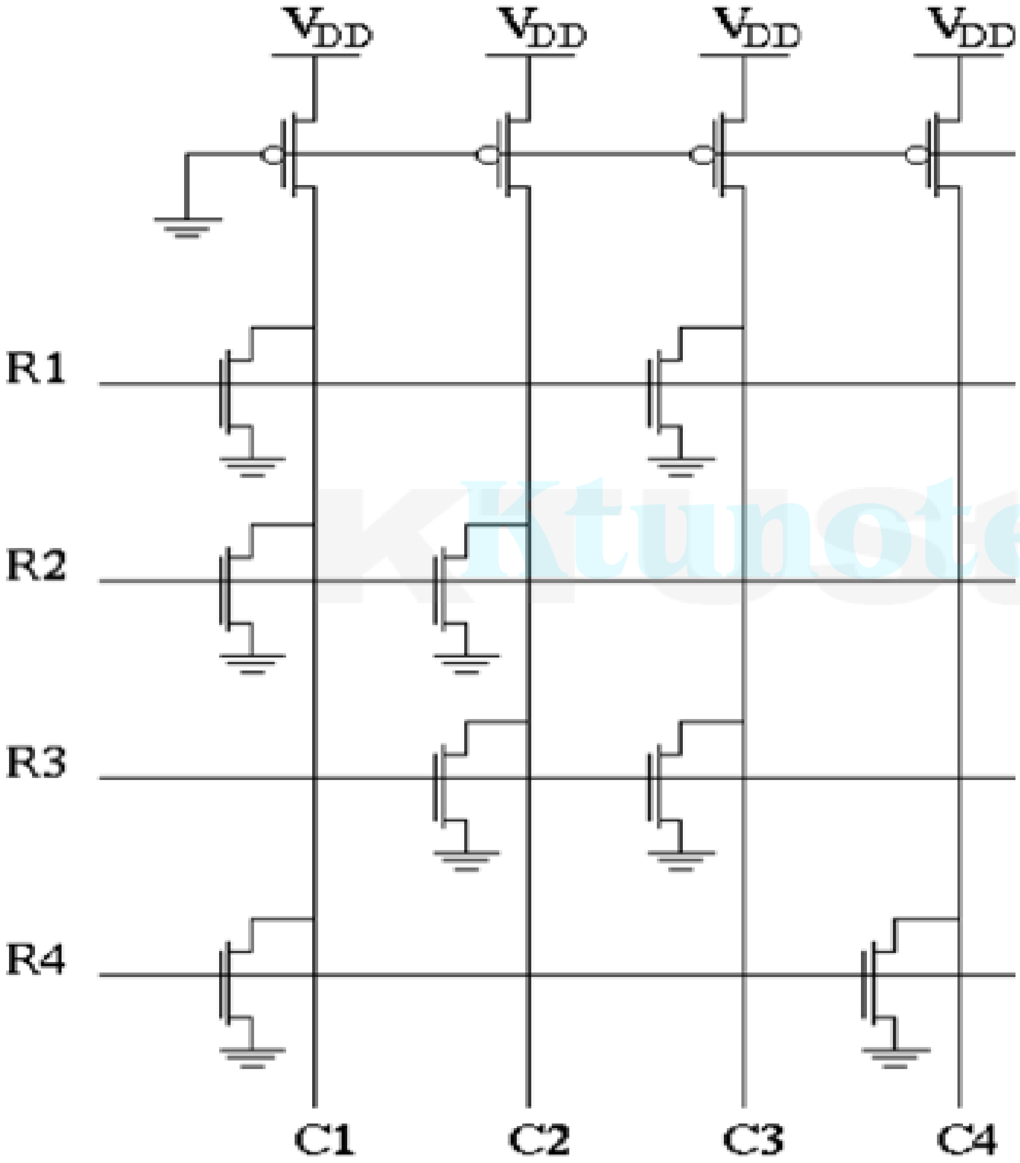
•More complex.

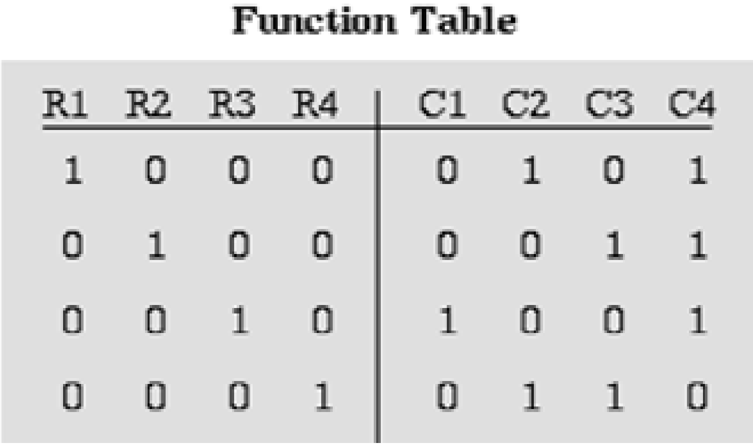
•It needs large area.

ROM cell array

- Different types of implementation of ROM arrays are:
- NOR-based ROM array
- NAND-based ROM array
- OR-based ROM array

4x4 MOS NOR ROM cell array




Function Table

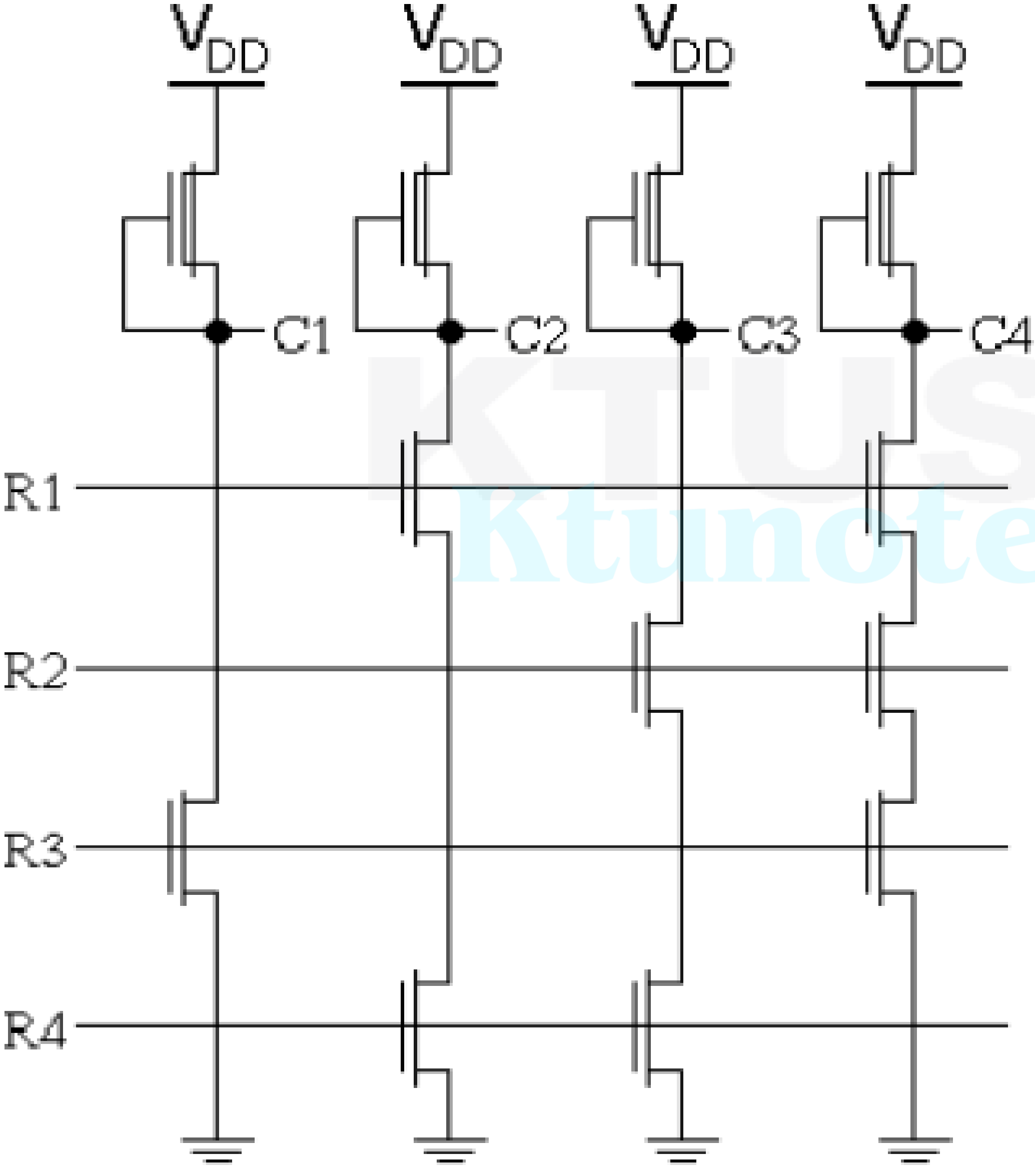
R1	R2	R3	R4	C1	C2	C3	C4
1	0	0	0	0	1	0	1
0	1	0	0	0	0	1	1
0	0	1	0	1	0	0	1
0	0	0	1	0	1	1	0

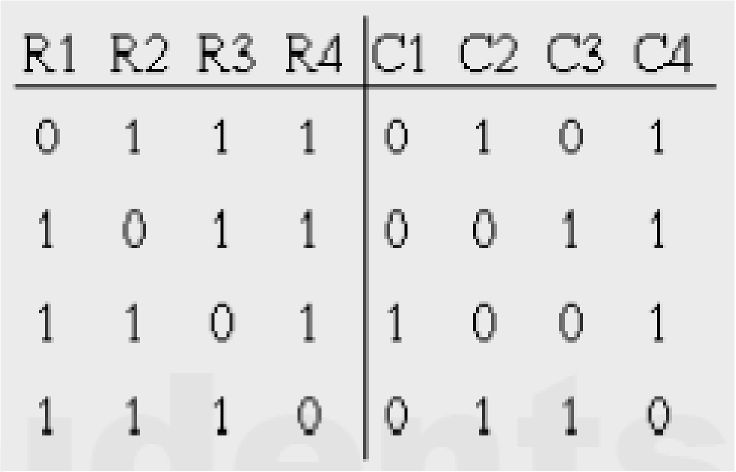
- Each column consists of pseudo-NMOS NOR gate driven by some of the row signals. i.e., the word line.

4x4 MOS NOR ROM cell array

- Only one word line is activated (selected) at a time by raising its voltage to VDD, while all other rows are held at a low voltage level.
- If an active transistor exists at the cross point of a column and the selected row, the column voltage is pulled down to the logic low level by that transistor.
- If no active transistor exists at the cross point, the column voltage is pulled high by the PMOS load device.
- Thus, a logic “1” –bit is stored as the absence of an active transistor, while a logic “0”-bit is stored as the presence of an active transistor at the cross-point.

4x4 MOS NAND ROM cell array





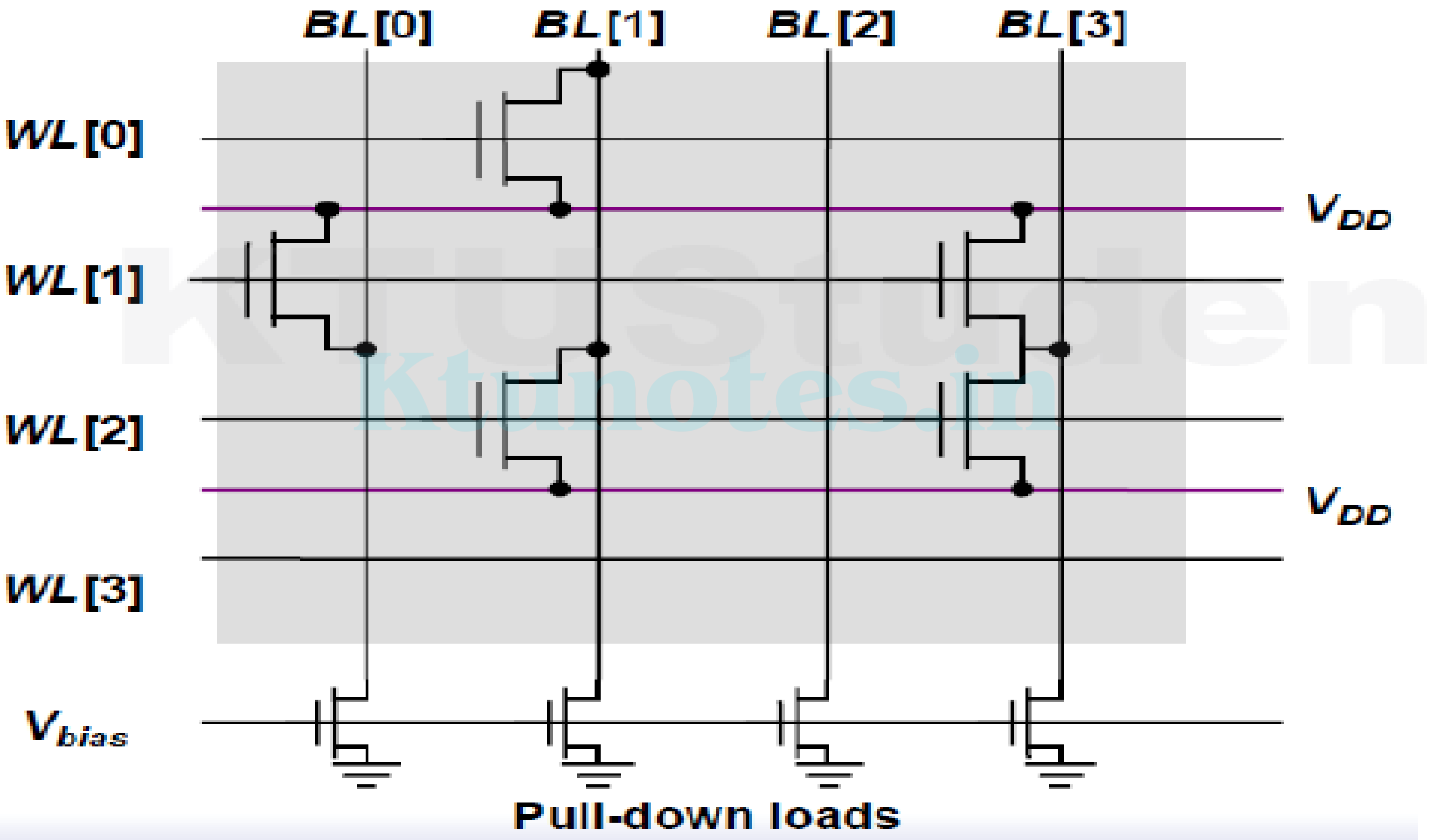
R1	R2	R3	R4	C1	C2	C3	C4
0	1	1	1	0	1	0	1
1	0	1	1	0	0	1	1
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	0

- Each bit line consists of a depletion-load NAND gate, driven by some of the row signals, i.e., the word lines.

4x4 MOS NAND ROM cell array

- In normal operation, all word lines are held at the logic HIGH voltage level except for the selected line, which is pulled down to logic LOW level.
- If a transistor exists at the cross point of a column and the selected row, that transistor is turned off and column voltage is pulled HIGH by the load device.
- On the other hand, if no transistor exists (shorted) at that particular cross point, the column voltage is pulled LOW by the other NMOS transistor in the multi-NAND structure.
- Thus, a logic "1"-bit is stored by the presence of a transistor that can be deactivated, while a logic "0"-bit is stored by a shorted or normally ON transistor at the cross-point.

4x4 OR ROM cell array



4x4 OR ROM cell array

- Function Table

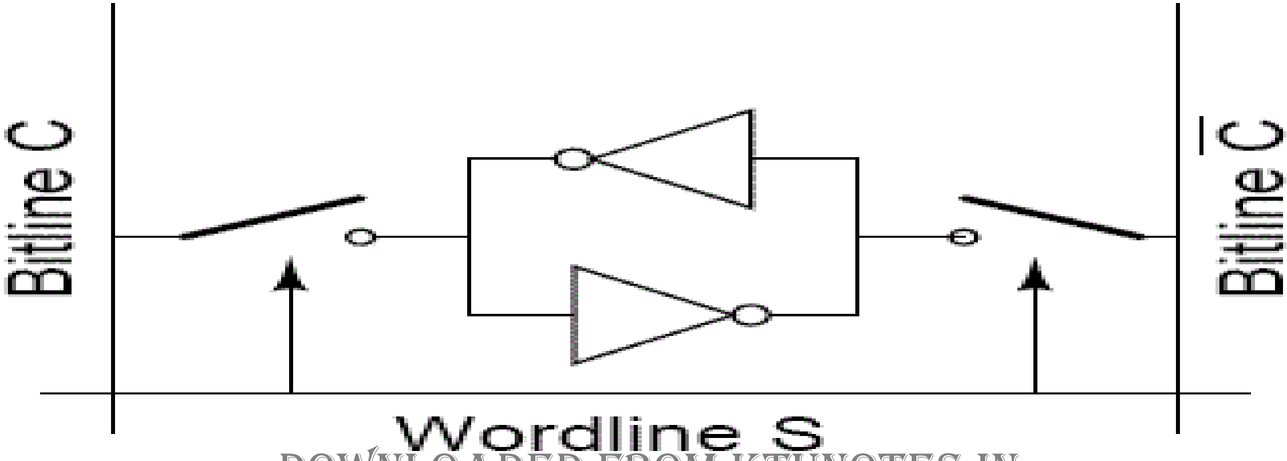
R0	R1	R2	R3	C0	C1	C2	C3
1	0	0	0	0	1	0	0
0	1	0	0	1	0	0	1
0	0	1	0	0	1	0	1
0	0	0	1	0	0	0	0

Static Read-Write Memory (SRAM)

- The memory circuit is said to be static if the stored data can be retained indefinitely, as long as the power supply is on, without any need for periodic refresh operation.
- The data storage cell, i.e., the one-bit memory cell in the static RAM arrays, invariably consists of a simple Latch circuit with two stable operating points.
- Depending on the preserved state of the two inverter latch circuit, the data being held in the memory cell will be interpreted either as logic ‘0’ or as logic ‘1’.

Static Read-Write Memory (SRAM)

- To access (read and write) the data contained in the memory cell via a bit line, we need at least one switch, which is controlled by the corresponding word line.
- Usually, two complementary access switches consisting of NMOS pass transistors are implemented to connect the 1-bit SRAM cell to the complementary bit lines (column).



Static Read-Write Memory (SRAM)

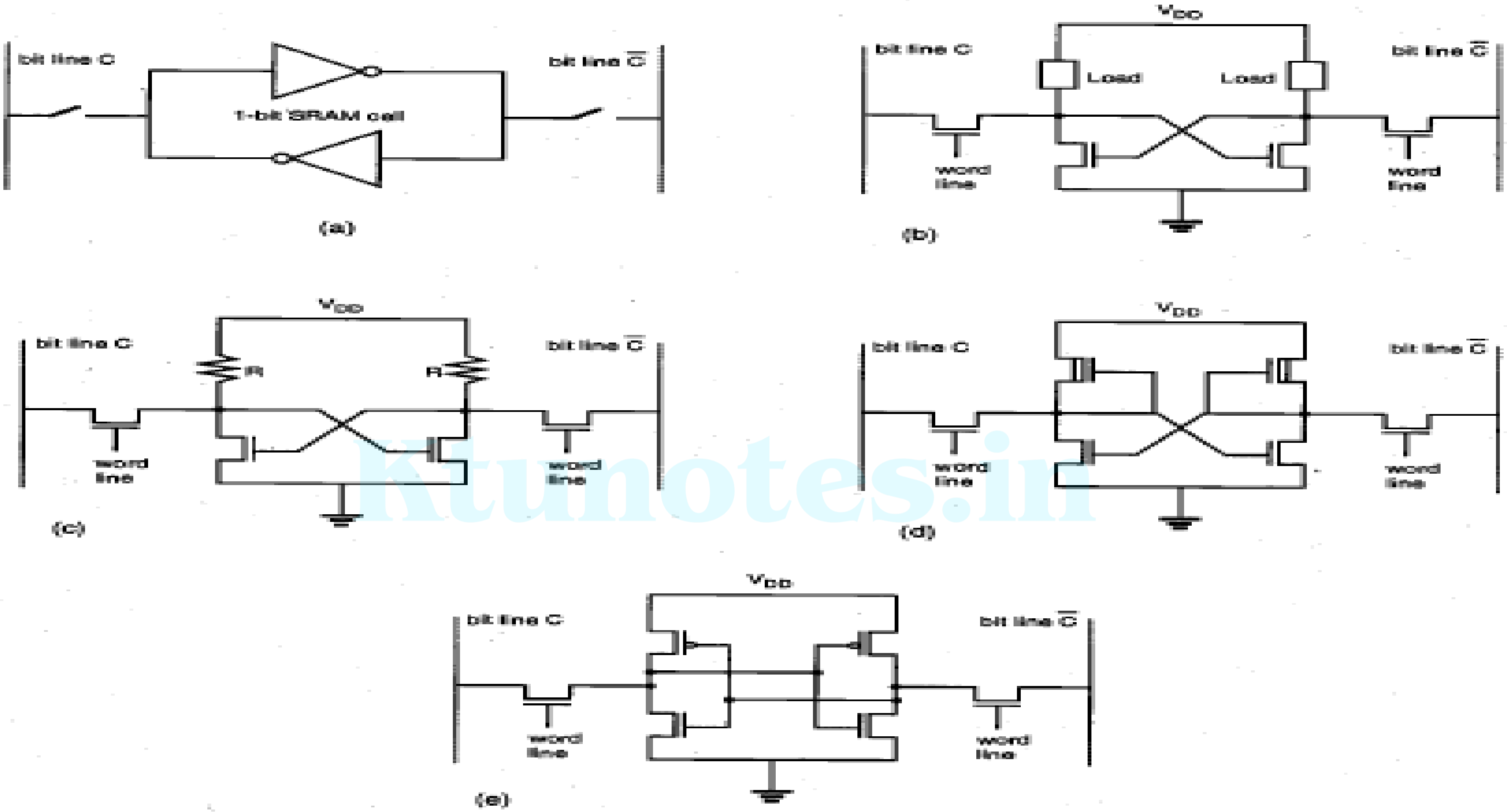
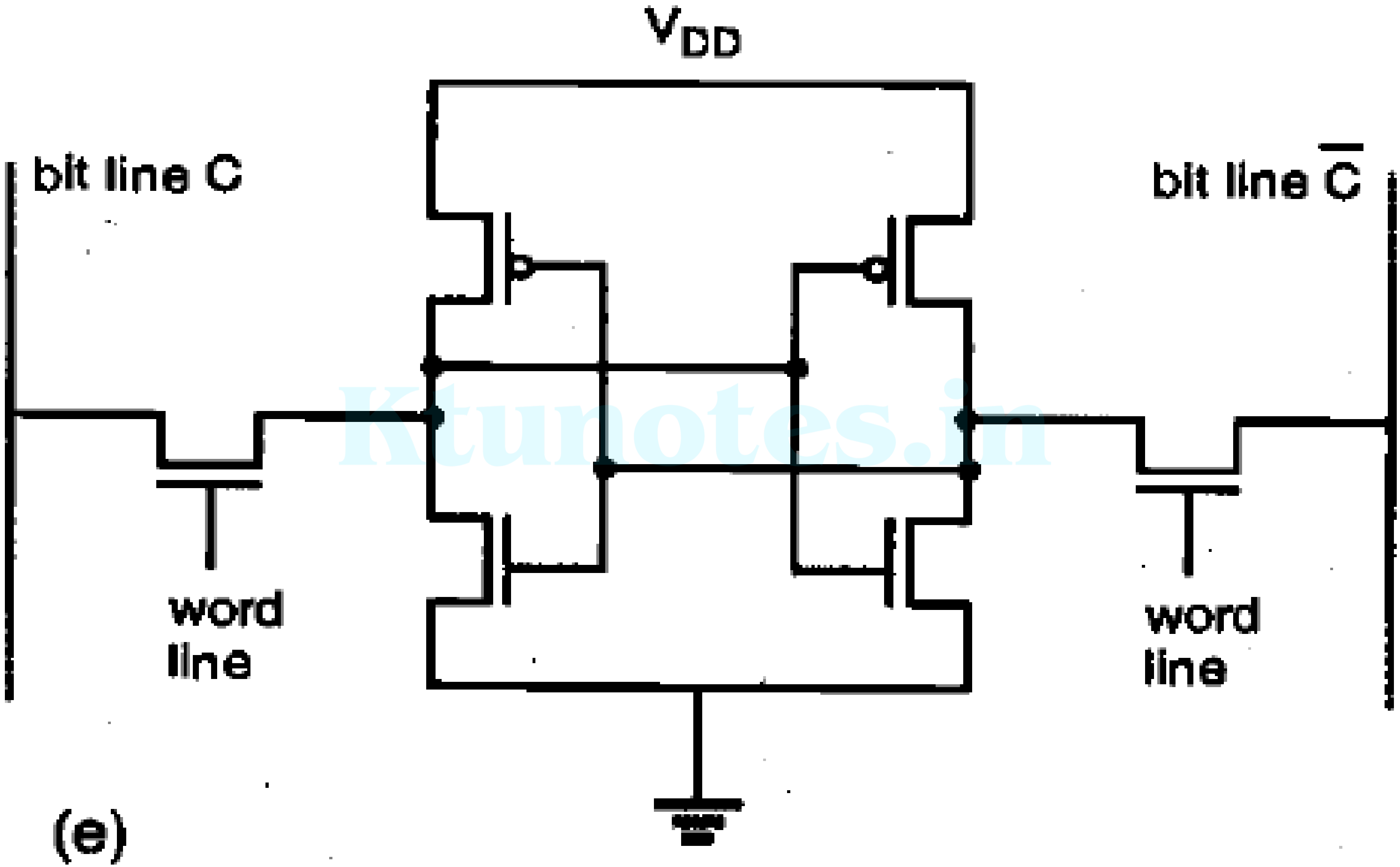
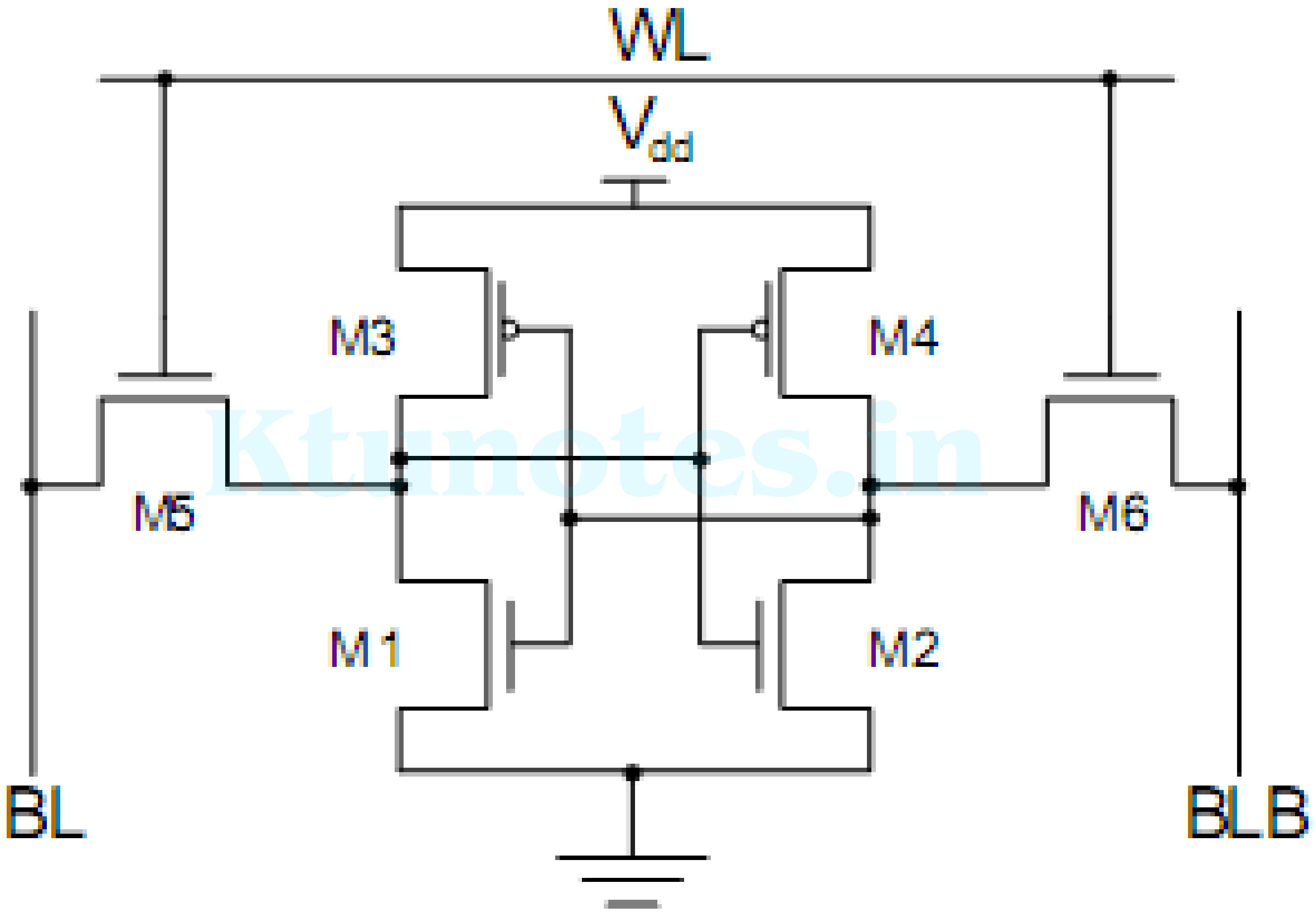


Figure 10.21. Various configurations of the static RAM cell. (a) Symbolic representation of the two-inverter latch circuit with access switches. (b) Generic circuit topology of the MOS static RAM cell. (c) Resistive-load SRAM cell. (d) Depletion-load nMOS SRAM cell. (e) Full CMOS SRAM cell.

CMOS SRAM Cell



CMOS SRAM Cell



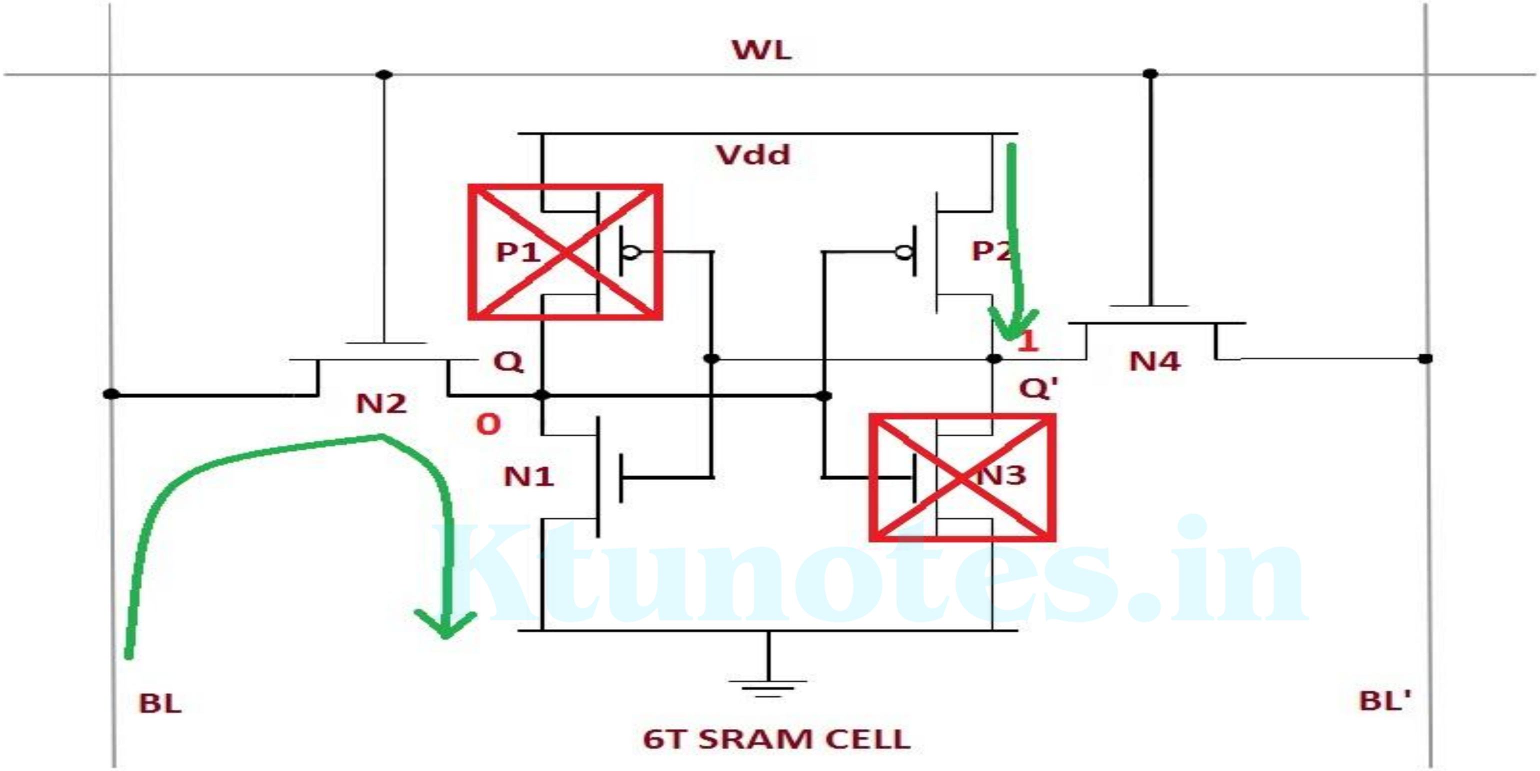
CMOS SRAM Cell

- The memory cell consists of simple CMOS inverters connected back to back, and two access transistors (M5 and M6).
- The access transistors are turned ON whenever a word line is activated for read or write operation, connecting the cell to the complementary bit line columns.
- SRAM has 3 operational modes:
  - ***Hold state:*** value of bit is stored in the cell for future use.
  - ***Write:*** 1 or 0 is fed to cell.
  - ***Read:*** value of stored bit is transmitted to the outside world.
- Pair of cross coupled inverters provides storage while access transistor provide read and write operation.

CMOS SRAM Cell

- Access transistors are controlled by word line signal WL that defines operational modes.
- WL = 0, both access transistors are OFF and the cell is in hold state.
- WL = 1. READ/WRITE
- In READ operation, bit and bit’ lines are output.
- In WRITE operation, bit and bit’ lines are inputs.

CMOS SRAM READ operation



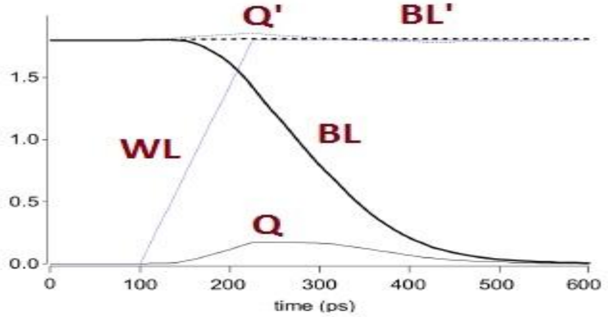
READ OPERATION: i) Precharge BL, BL' to HIGH

ii) Turn on WL

iii) BL or BL' will be pulled down to LOW depending on Q, Q'

iv) Eg. If Q = 0, Q'= 1, BL discharges through N2 - N1 - GND and BL' stays high. But Q bumps up slightly

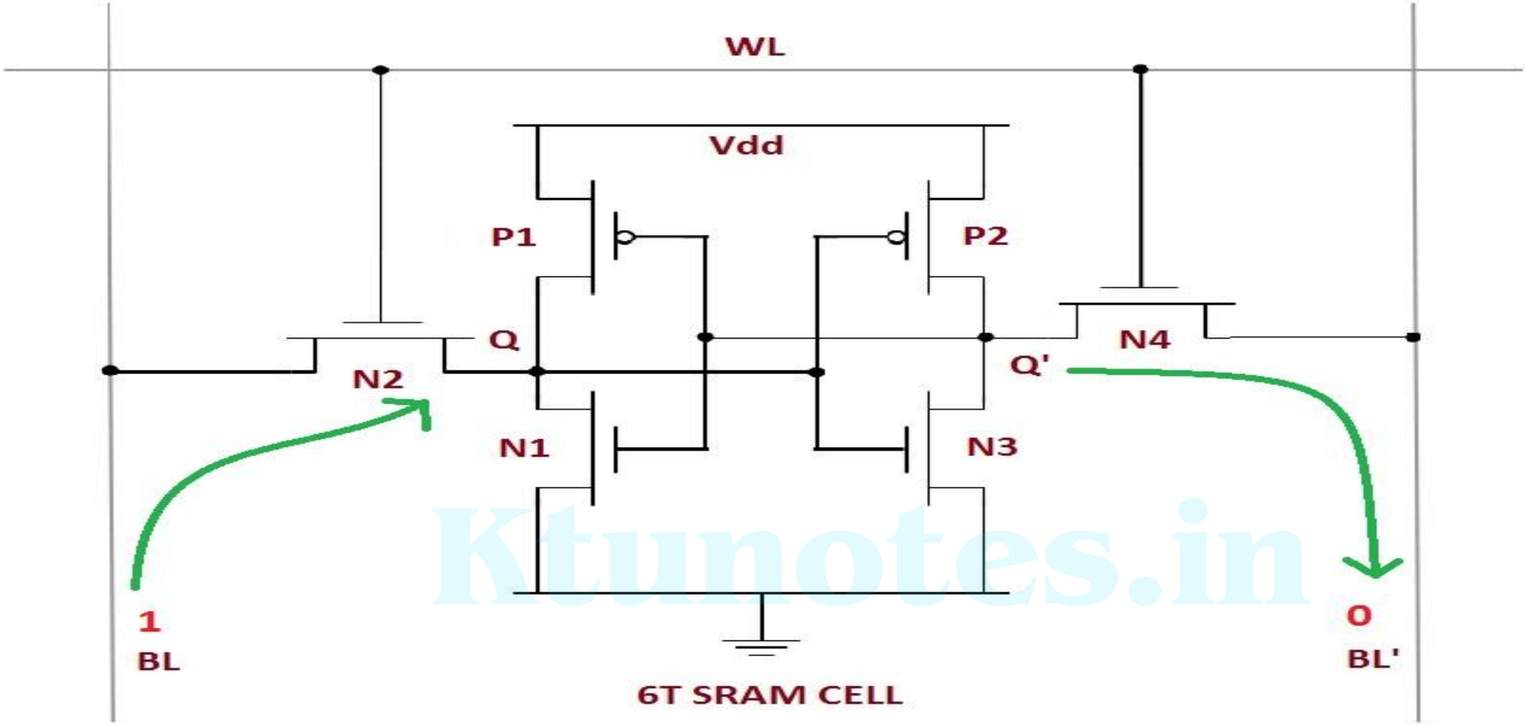
v) In order for Q to not flip N1 should be stronger than N2, i.e N1 >> N2



CMOS SRAM READ operation

- Assume that logic’0’ is stored in the cell.
- Both bit lines are pre-charged to VDD.
- The transistors N3 and P1are turned OFF, while the transistors N1 and P2 operate in linear mode.
- Read cycle is started by asserting WL enabling access transistor.
- During READ operation, value stored in Q and Q’ are transferred to the bit line by leaving BL’ in its pre-charged value and by discharging BL through N1 and N2.

CMOS SRAM WRITE operation



WRITE OPERATION: i) Drive BL, BL' with necessary values

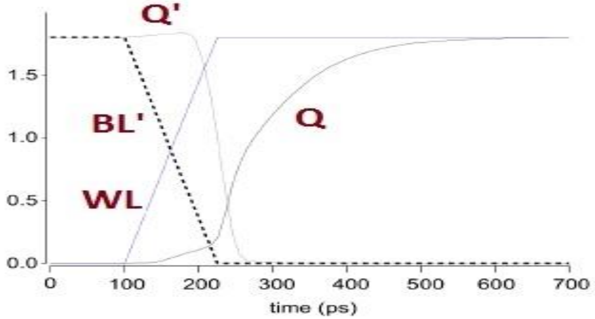
ii) Turn on WL

iii) Bit Lines overpower cell with new value

iv) Eg. Q = 0, Q' = 1 and BL = 1, BL' = 0. This forces Q' to low and Q to high

v) To overpower feedback inverter loop, N2 should be stronger than P1, i.e N2 >> P1

As Q starts to charge up to 1, output of inverter P2-N3 starts to discharge which in turn, makes P1 turn on. Thus the feedback inverters lock on to the values to be written



CMOS SRAM WRITE operation

- Assume that logic’0’ is stored in the cell.
- A ‘1’ is written in the cell by setting BL = 1 or BL’ = 0.
- This cause the flip-flop to change state.
  - Q = 0 and Q’ = 1
  - WL = 1
  - BL and BL’ are inputs
  - BL’ = 0 and BL = 1
  - Q = 1

CMOS SRAM WRITE operation

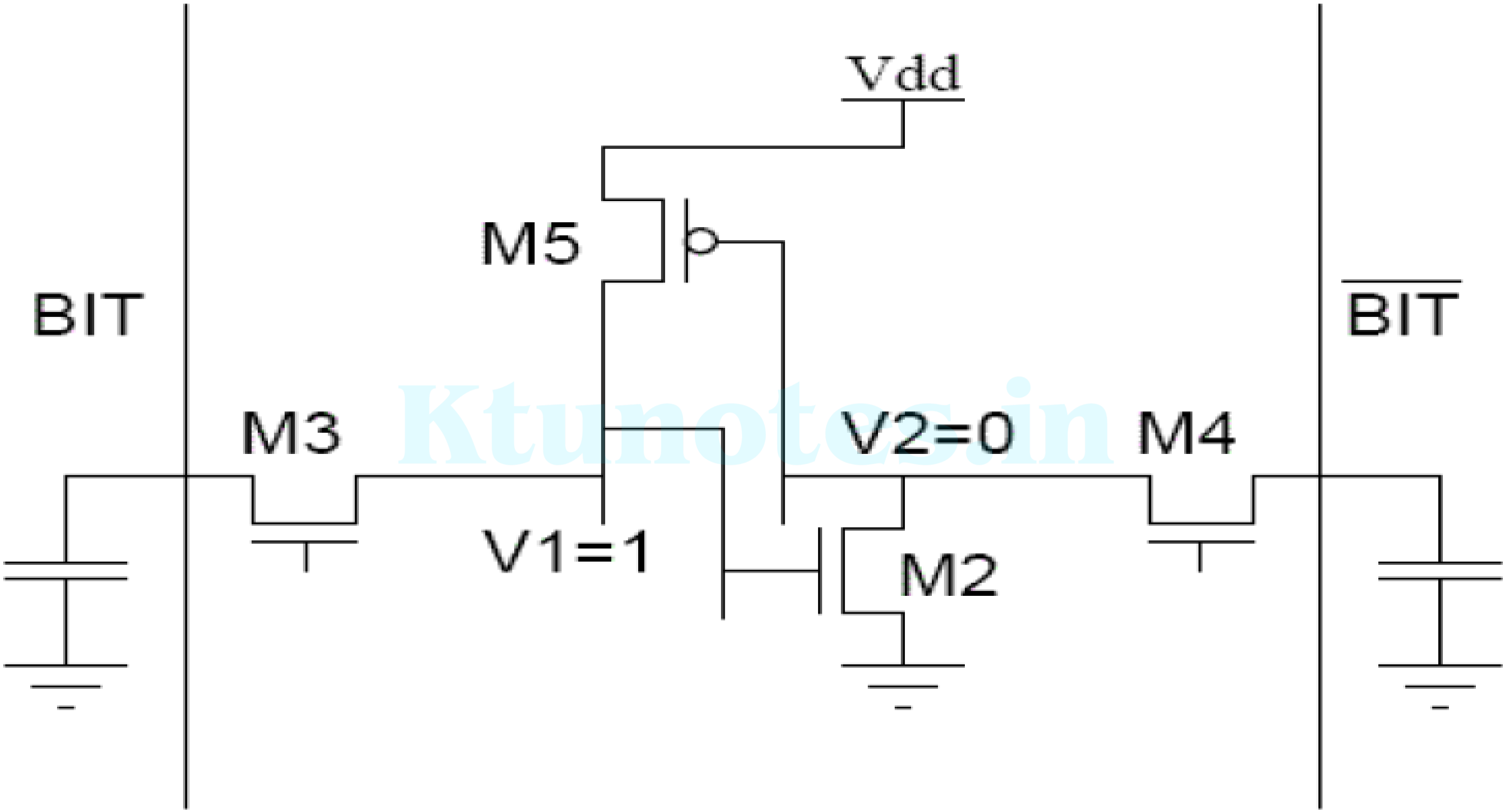


Fig 28.51: SRAM start of write '0'

CMOS SRAM

- ***Advantages:***
- Static power dissipation is very small.
- High noise immunity due to larger noise margins.
- The ability to operate at lower supply voltage.
- ***Disadvantages:***
- Large cell size

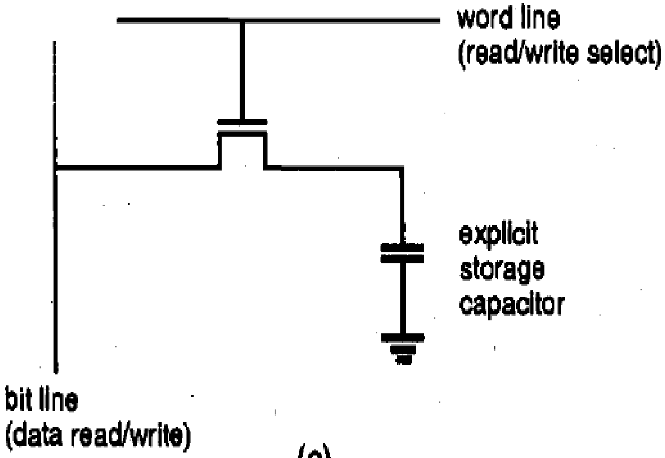
Dynamic Read-Write Memory (DRAM)

- The only problem with SRAM cell is that, it occupies a significant amount of silicon space.
- This problem is addressed in the **dynamic read-write memory (DRAM)**.
- In dynamic RAM, binary data is stored as charge in a capacitor.
- The memory cell consists of a storage capacitor and an access transistor.
- In DRAM, the binary data is stored as charge in the capacitor, where the presence and absence of charge determines the value of stored bit.

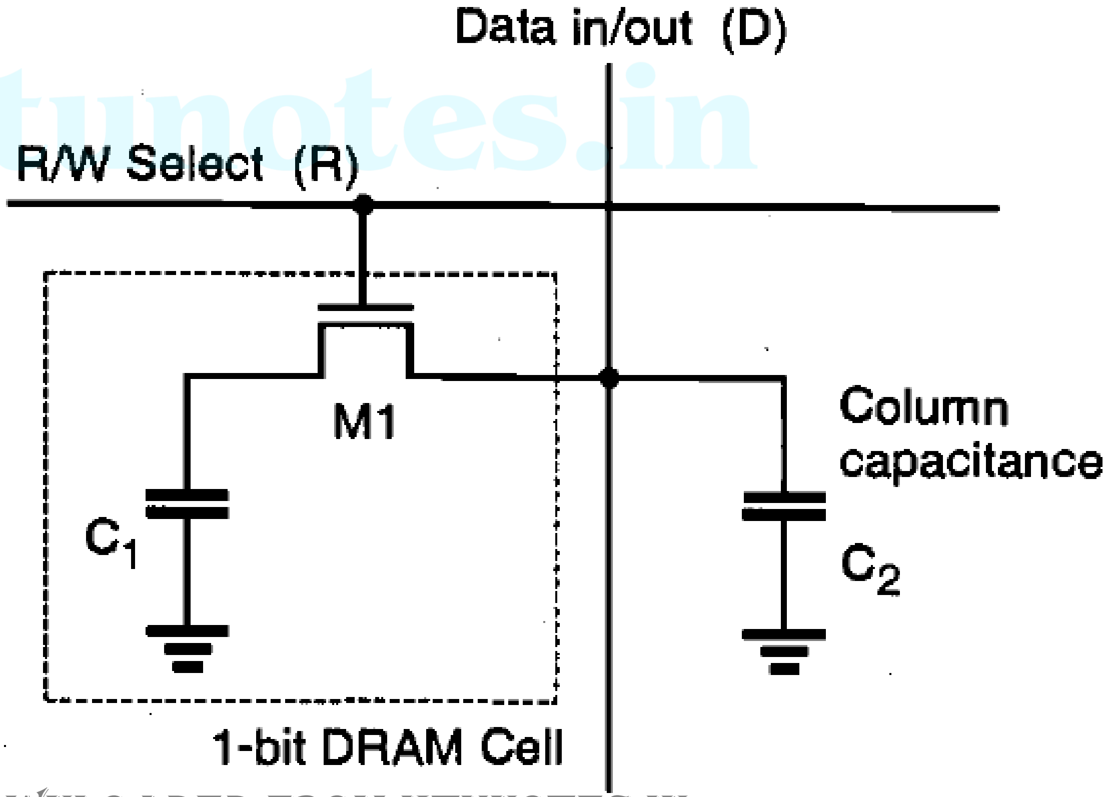
Dynamic Read-Write Memory (DRAM)

- But data in capacitor cannot be stored for a long time because a capacitor holds an electrical charge for a limited amount of time as the charge gradually drains away.
- DRAM cells requires a periodic refreshing of the stored data.
- The use of capacitor as the primary storage device enables the DRAM cell to be realized on a much smaller area.
- Access devices or switches are used for READ/WRITE operations.

One-Transistor (1-T) DRAM



(c)



One-Transistor (1-T) DRAM

- 1-T DRAM cell consists of ***one explicit storage capacitor*** (***C1***) and ***one access transistor*** (***M1***).
- Binary data are stored as the presence or absence of charge in the storage capacitor.
- Capacitor ***C2*** represents the much larger parasitic column capacitance associated with the word line.
- Charge sharing between this large capacitance and the very small storage capacitance plays a very important role in the operation of the 1-T DRAM cell.

One-Transistor (1-T) DRAM

- To consider ***read/write*** operations, we have to take into account a significant parasitic column capacitor ***C2*** associated with each column.
- The control line is controlled by the row address decoder.
- Before any operation is performed, each column capacitance is pre-charged high.
- The cell is selected for a ***read/write*** operation by asserting its word line high (R = 1). This connects the storage capacitance ***C1*** to the bit line (D).

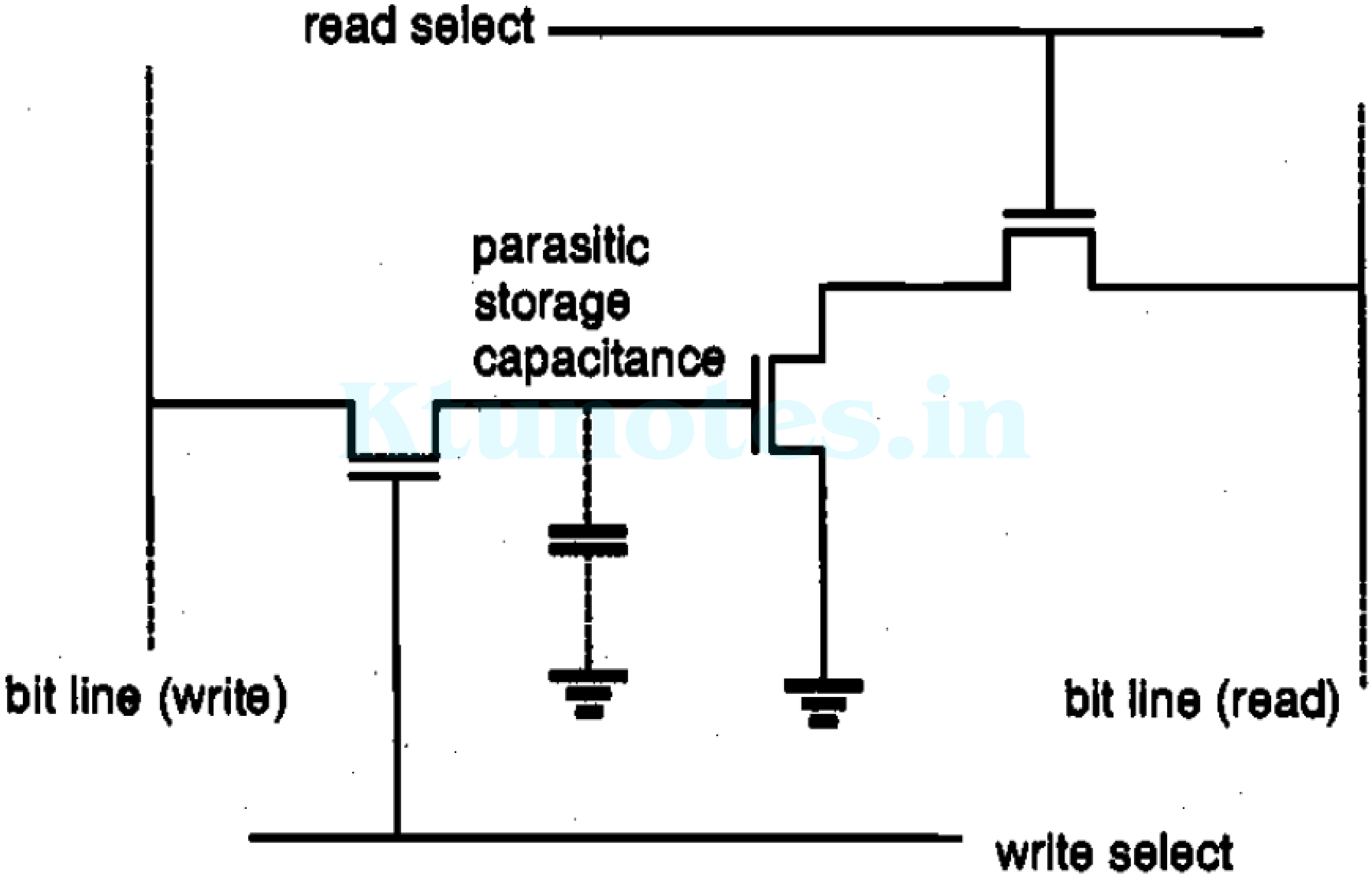
One-Transistor (1-T) DRAM

- The ***write operation*** is performed by applying either ***high or low*** voltage to the bit line, thus ***charging (write '1') or discharging (write '0')*** the capacitor through the access transistor.
- During ***read operation***, there is a flow of charges between the storage capacitor ***C1*** and the column capacitor ***C2***.
- As a result the column capacitor voltage either ***increases (read '1') or decreases (read'0')*** slightly. This difference can then be amplified by the sense amplifier.

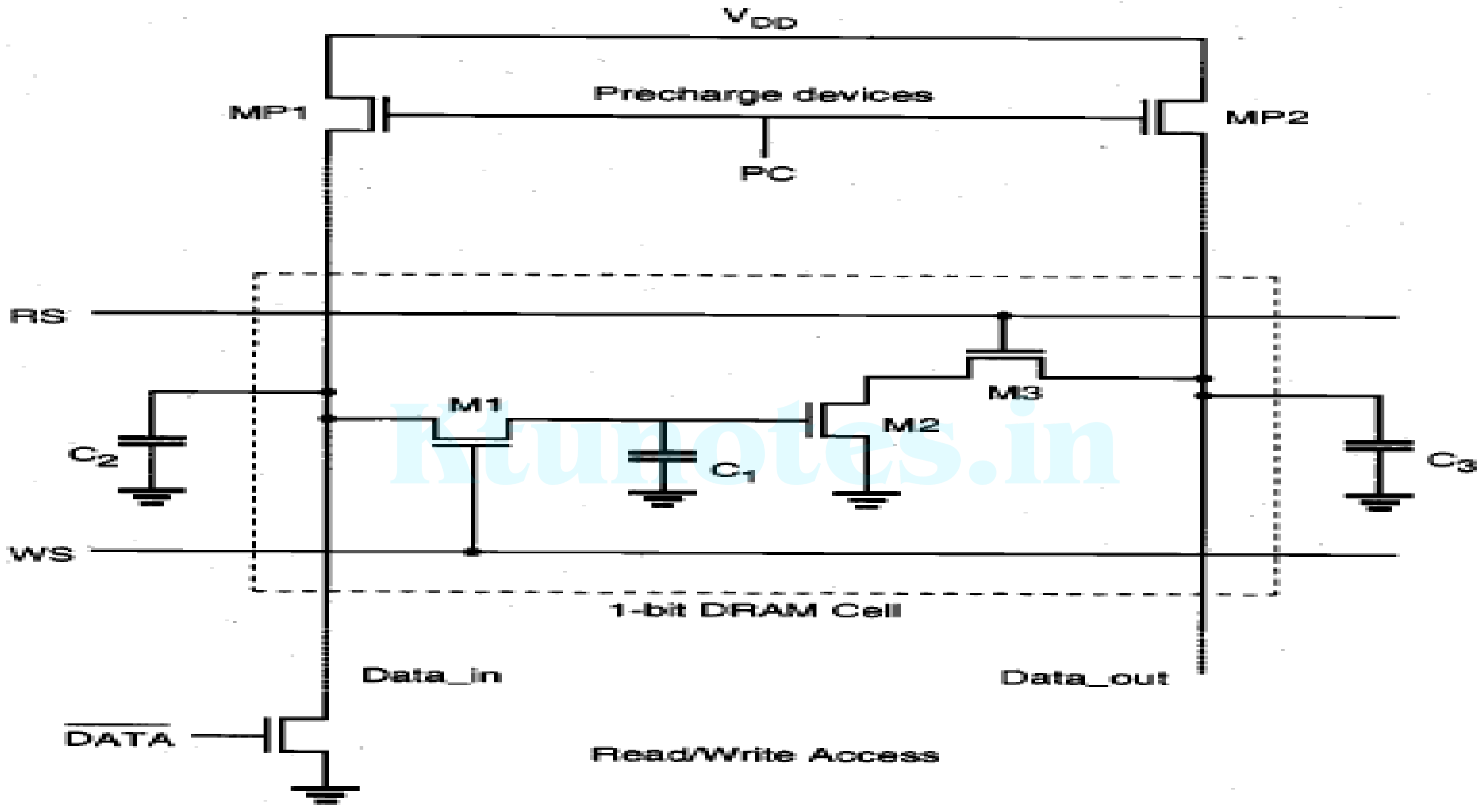
One-Transistor (1-T) DRAM

- Read operation destroys the charge stored on the storage capacitor ***C1*** (“***destructive read out***”).
- Therefore, the data must be restored (refreshed) each time the read operation is performed.

Three-Transistor (3-T) DRAM



Three-Transistor (3-T) DRAM



Three-transistor DRAM cell with the pull-up and read/write circuitry.

Three-Transistor (3-T) DRAM

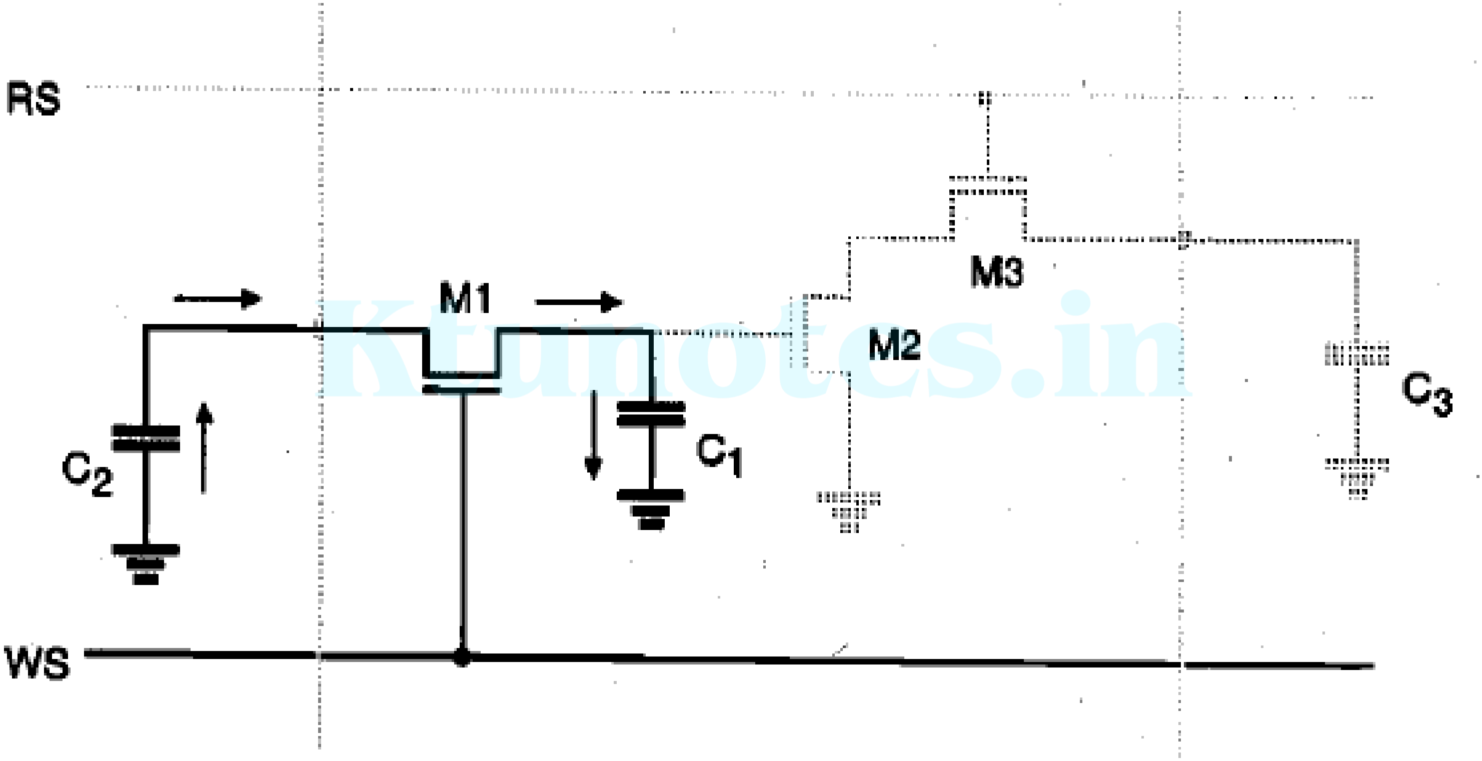
- This DRAM cell utilise a single transistor as the storage device and one transistor each for READ and WRITE access switches.
- The column pull up transistors and the column READ/WRITE circuitry is also there.
- Here the binary data is stored in the form of charge in the parasitic node capacitance ***C1***.
- The cell has 2 separate bit lines for “data read” and “data write”, and 2 separate word lines to control the access transistors.

Three-Transistor (3-T) DRAM

- The storage transistor ***M2*** is turned OFF or ON depending on the charge stored in storage capacitor ***C1*** and the pass transistors ***M1*** and ***M3*** act as access switches for data WRITE and READ operations.
- Selection lines for reading and writing must be separated because the stored charge on ***C1*** would be lost if ***M3*** is turned ON during reading.
- Two column capacitances ***C2*** and ***C3*** are at least one order of magnitude larger than the internal storage capacitance ***C1***.

Three-Transistor (3-T) DRAM

- ***Write "1" operation:***



Charge sharing between C_2 and C_1 during the write "1" sequence.

Three-Transistor (3-T) DRAM

- ***Write "1" operation:***
  - Pre-charge ***C2*** and ***C3***. (Charged to logic "1" level).
  - For the write "1" operation, the inverse data input (DATA') is at logic "0". Thus the data write transistor ***MD*** is turned OFF.
  - Set ***WS = 1***. Then ***M1*** turns ON, i.e., the charge on ***C2*** is shared with ***C1***.
  - Since the capacitance ***C2*** is very large compared to ***C1***, the storage node capacitance ***C1*** attains approximately same logic-high level as ***C2***.
  - After write operation (***WS = 0***), ***M1*** turns OFF. Since ***C1*** is charged up to logic "1", ***M2*** turns ON.

Three-Transistor (3-T) DRAM

- ***Read "1" operation:***

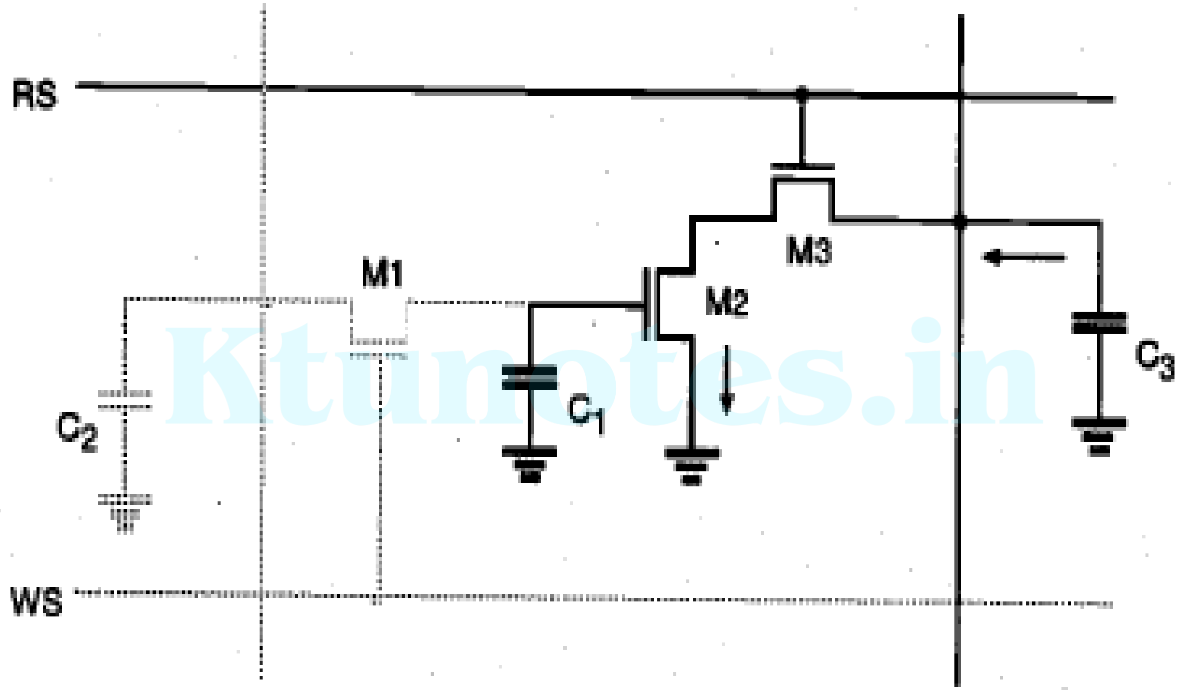


Figure 10.41. The column capacitance C_3 is discharged through the transistors M2 and M3 during :he read "1" operation.

Three-Transistor (3-T) DRAM

- ***Read "1" operation:***
  - Pre-charge ***C2*** and ***C3***. (Charged to logic ”1” level).
  - Set ***RS = 1***. Then ***M3*** turns ON.
  - As ***C1*** is having logic "1", ***M2*** is also ON.
  - Thus ***C3*** discharges through ***M2*** and ***M3*** and the falling column voltage is interpreted by the "data read" circuitry as a stored logic "1".
  - Here the charge stored in ***C1*** is not disturbed.

Three-Transistor (3-T) DRAM

- ***Write "0" operation:***

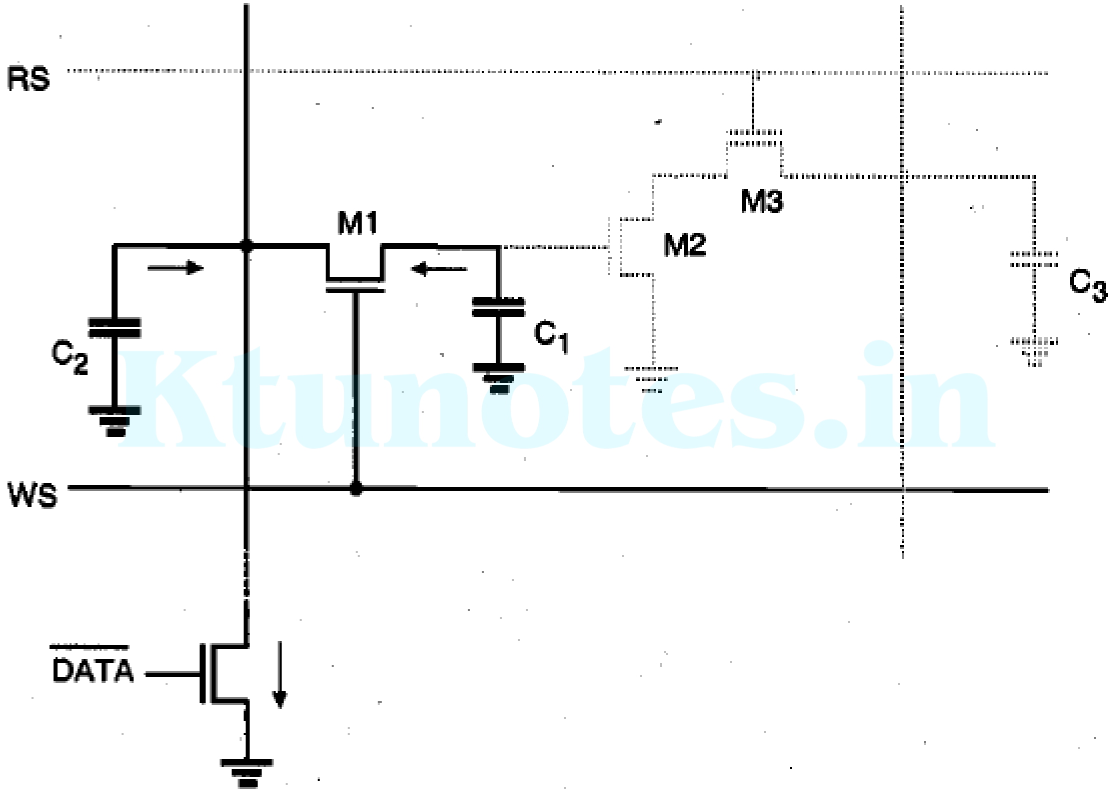


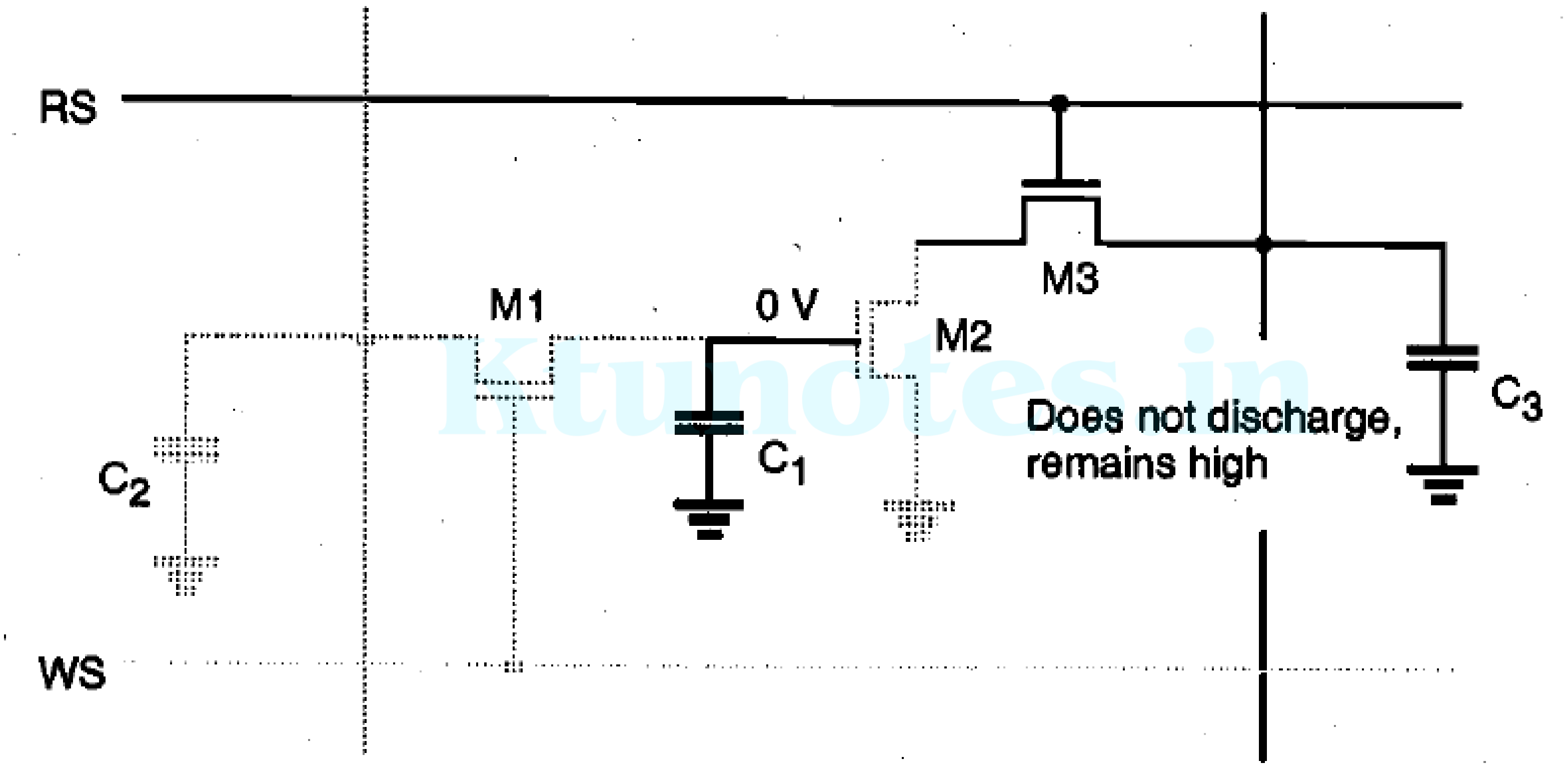
Figure 10.42. Both C_1 and C_2 are discharged via M1 and the data write transistor during the write "0" sequence.

Three-Transistor (3-T) DRAM

- ***<u>Write “0” operation:</u>***
  - Pre-charge ***C2*** and ***C3***. (Charged to logic ”1” level).
  - For the write “0” operation, the inverse data input (DATA’) is at logic “1”. Thus the data write transistor ***MD*** is turned ON.
  - Set ***WS = 1***. Then ***M1*** turns ON and voltage level on ***C1*** and ***C2*** are pulled to logic “0”, through ***M1*** and ***MD***.
  - After write operation (***WS = 0***), ***M1*** turns OFF. Since ***C1*** is discharged up to logic “0”, ***M2*** turns OFF.

Three-Transistor (3-T) DRAM

- ***Read "0" operation:***



).43. The column capacitance C_3 cannot discharge during the read "0" cycle.

Three-Transistor (3-T) DRAM

- ***Read "0" operation:***
  - Pre-charge ***C2*** and ***C3***. (Charged to logic ”1” level).
  - Set ***RS = 1***. Then ***M3*** turns ON.
  - As ***C1*** is having logic "0", ***M2*** is OFF.
  - Thus ***C3*** will not discharge through ***M2*** and ***M3*** as there is no conducting path and the logic-high level on the column voltage is interpreted by the "data read" circuitry as a stored logic "0"bit.
  - Here the charge stored in ***C1*** is not disturbed.

Three-Transistor (3-T) DRAM

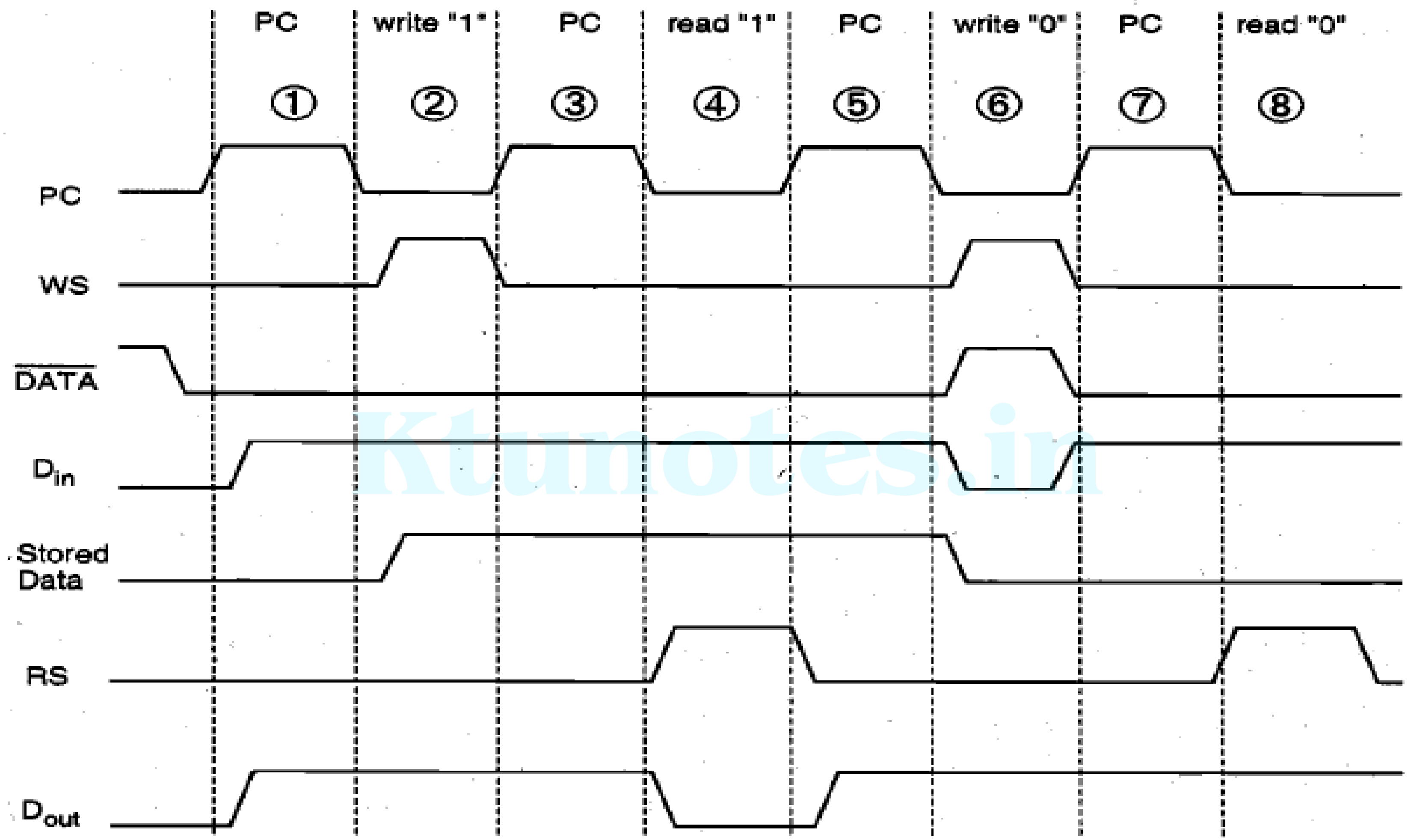


Figure 10.38. Typical voltage waveforms associated with the 3-T DRAM cell during four consecutive operations: write "1," read "1," write "0," and read "0."

Three-Transistor (3-T) DRAM

- ***Advantages:***
- Less number of transistors required.
- Does not dissipate any static power for data storage.
- Use of periodic pre-charge cycles instead of static pull-up further reduces the dynamic power dissipation.
- ***Disadvantages:***
- Additional peripheral circuitry required for scheduling the non-overlapping control signals and the refresh cycles (2 to 4 ms).