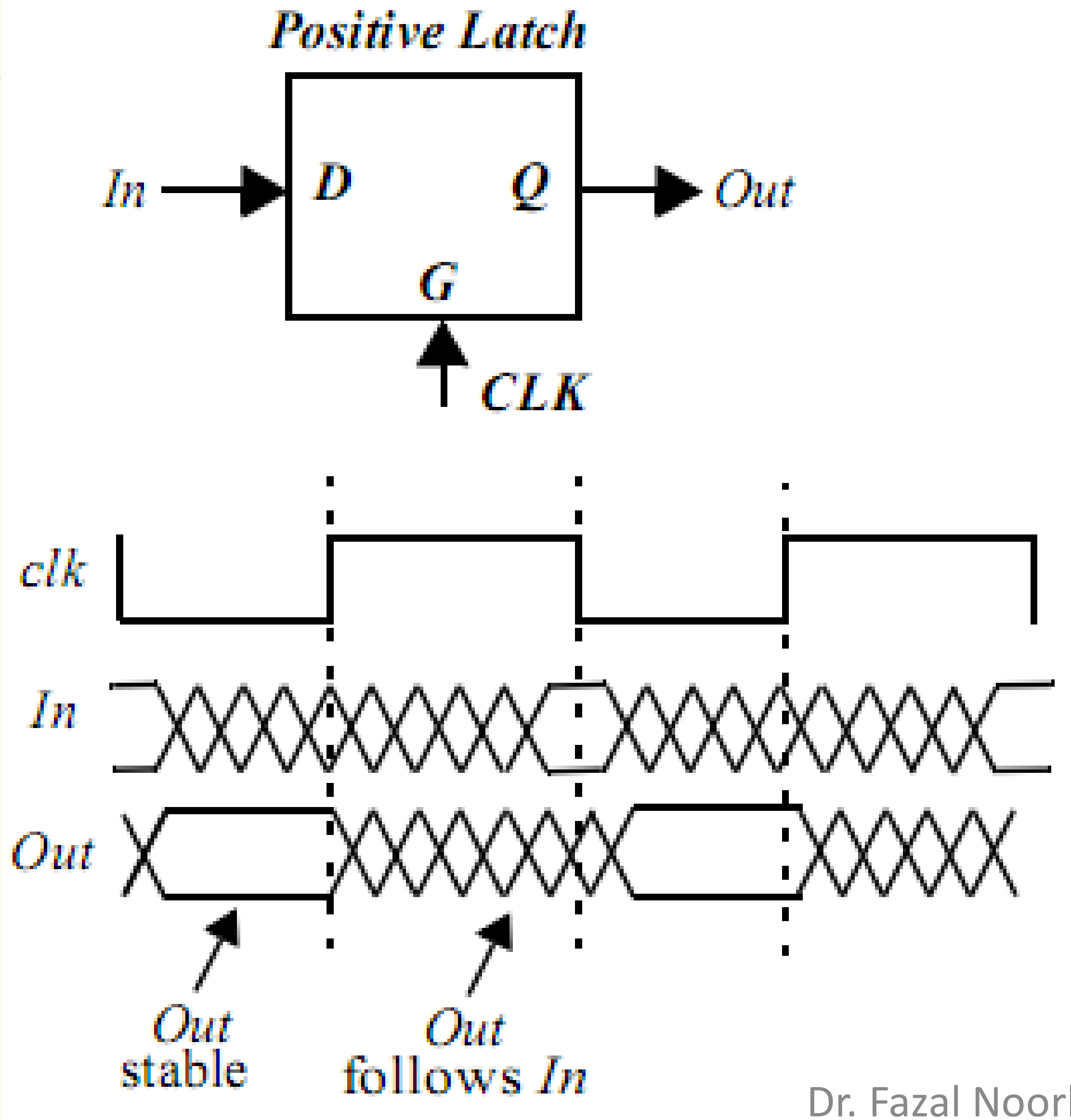
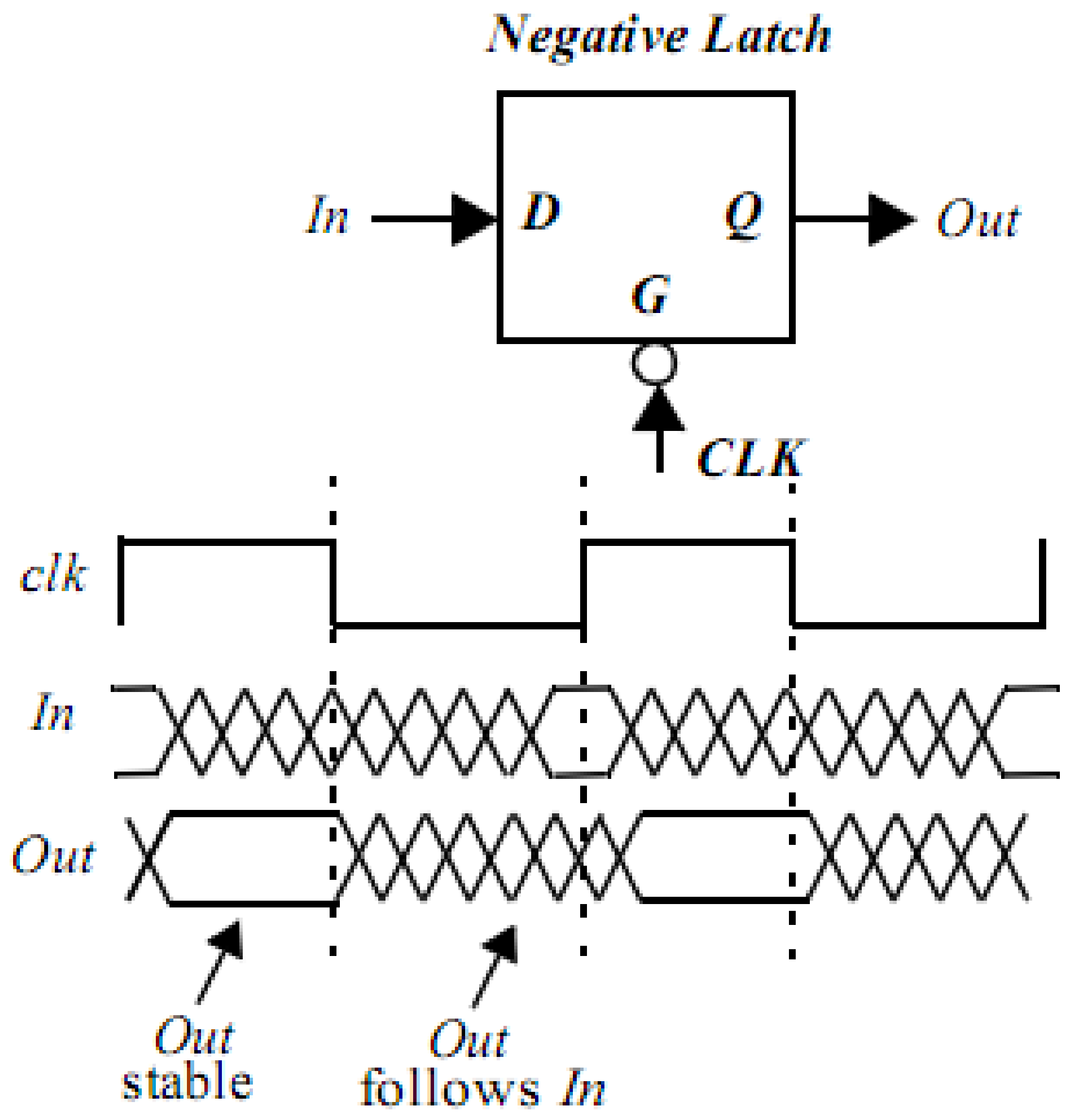
CO-4

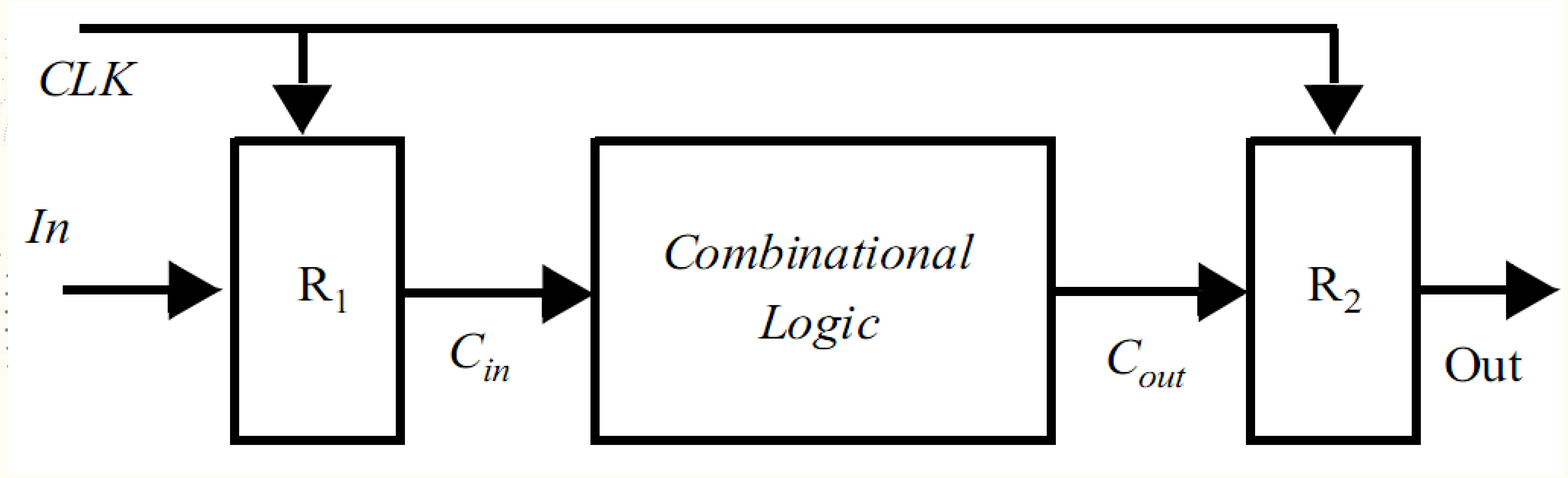
Dr. Fazal Noorbasha

Timing of positive and negative latches



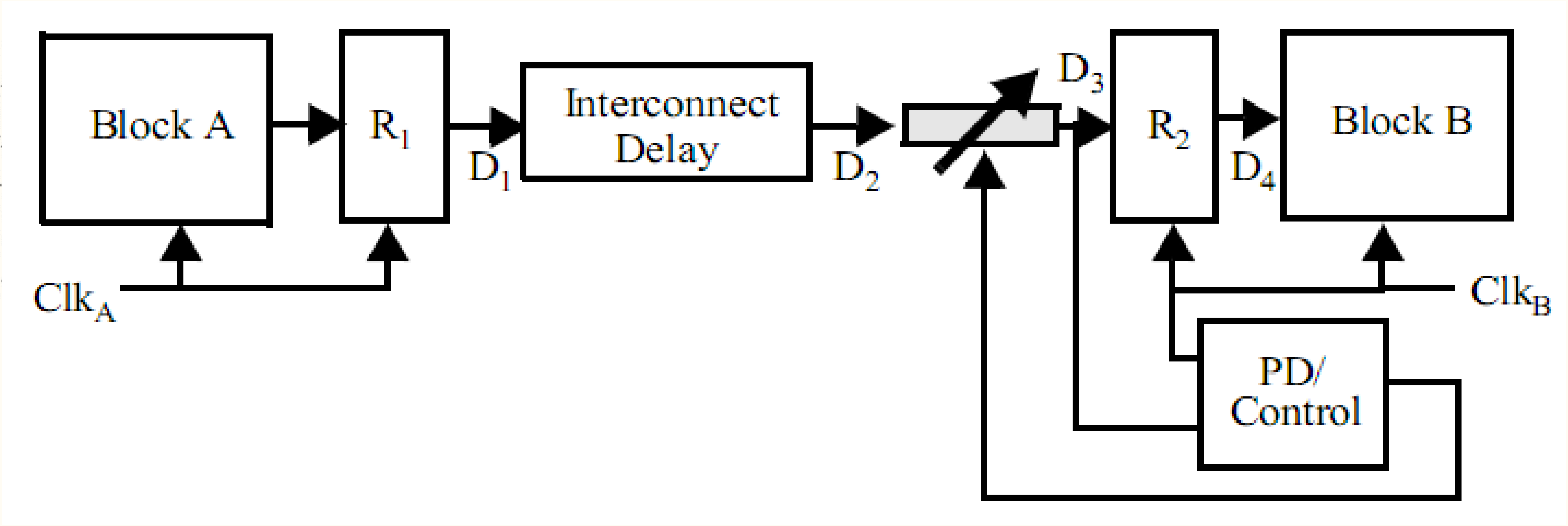


Dr. Fazal Noorbasha



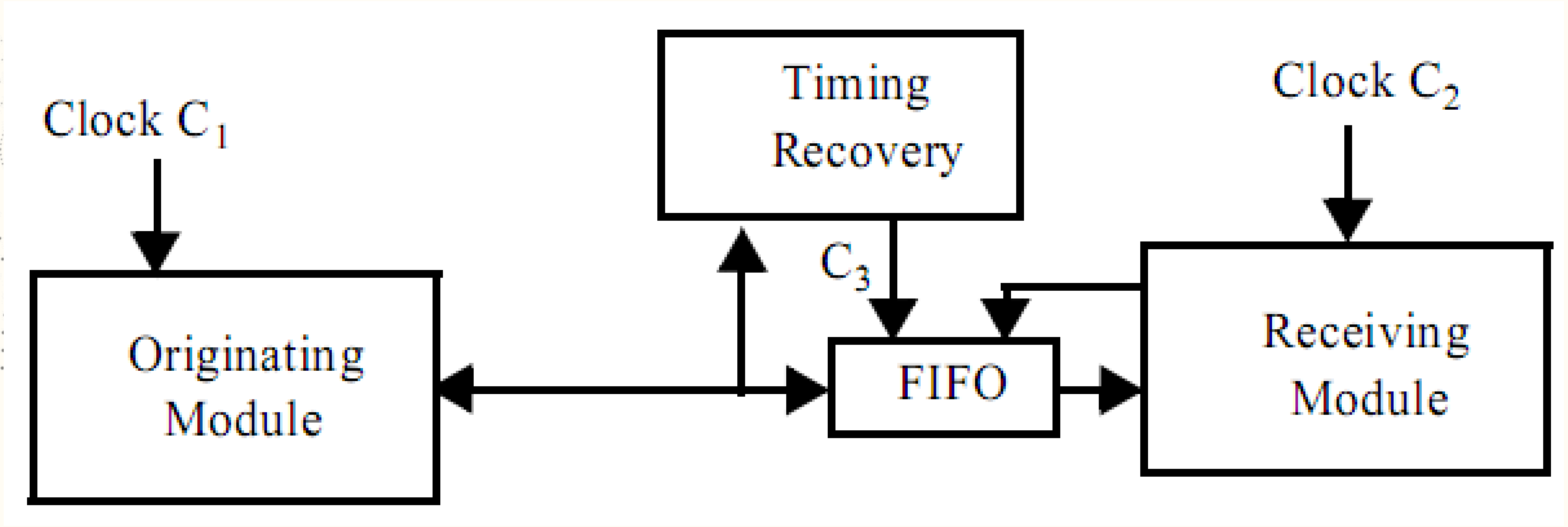
Synchronous interconnect methodology.

A synchronous signal is one that has the exact same frequency, and a known fixed phase offset with respect to the local clock. In such a timing methodology, the signal is "synchronized" with the clock, and the data can be sampled directly without any uncertainty.



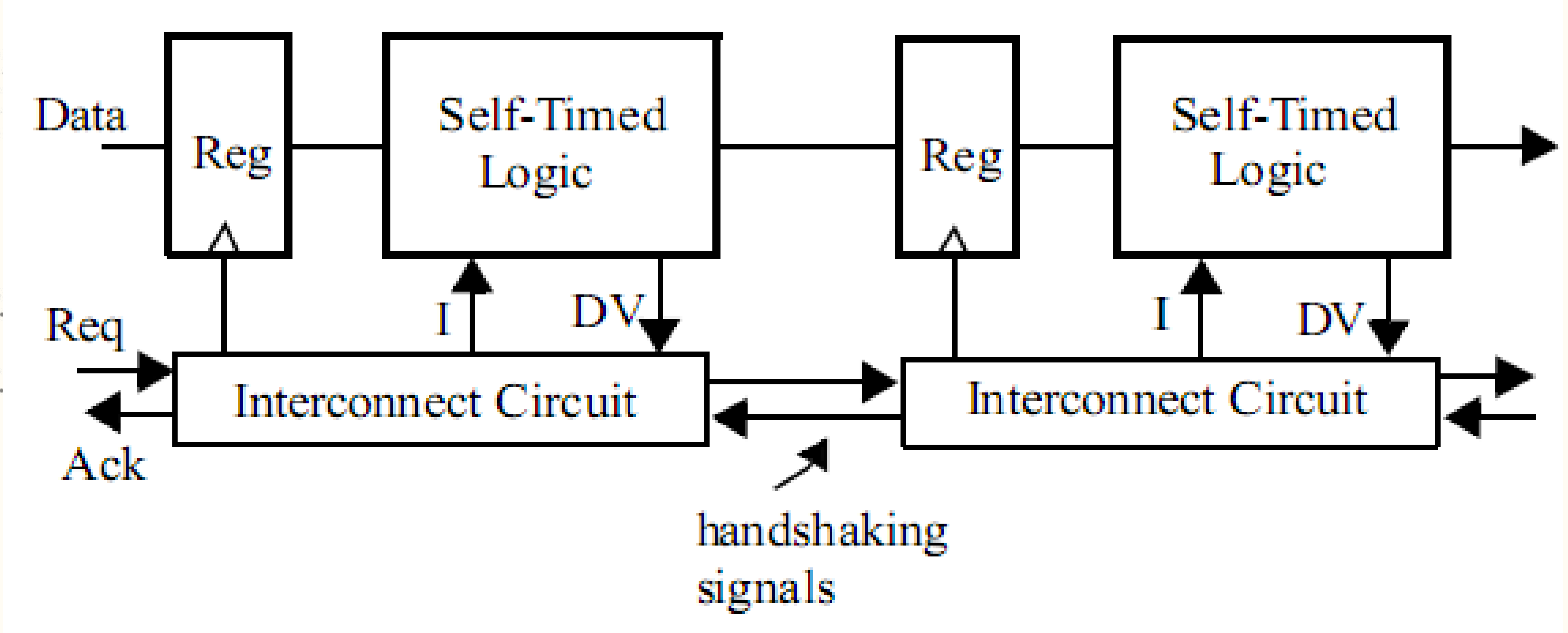
Mesochronous communication approach using variable delay line.

A mesochronous signal is one that has the same frequency but an unknown phase offset with respect to the local clock ("meso" from Greek is middle).



Plesiochronous communications using FIFO.

A plesiochronous signal is one that has nominally the same, but slightly different frequency as the local clock ("plesio" from Greek is near).



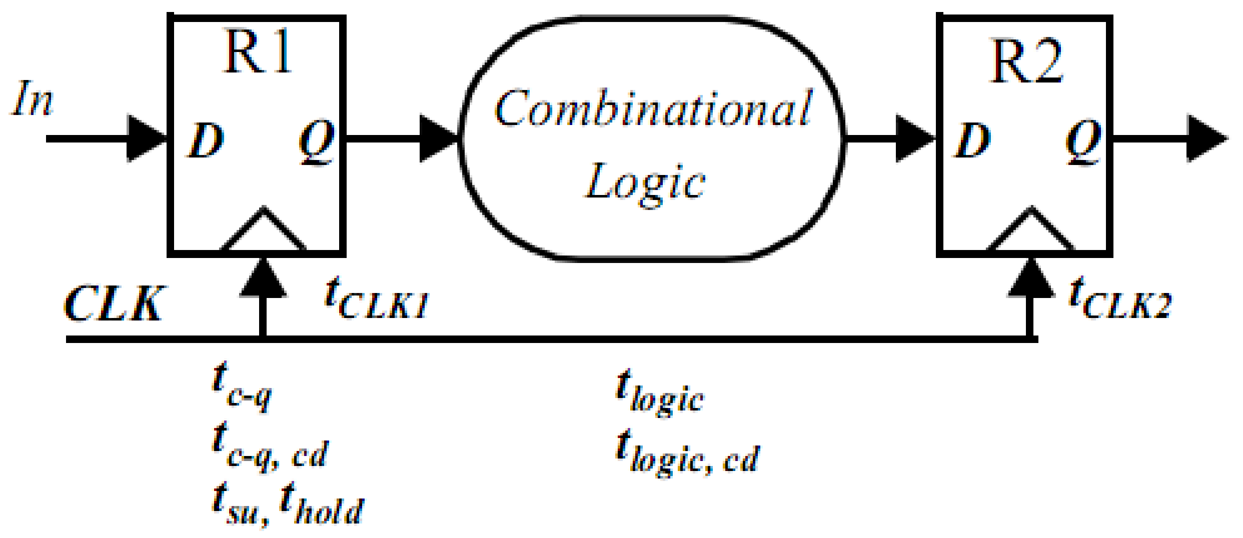
Asynchronous design methodology for simple pipeline interconnect

Asynchronous signals can transition at any arbitrary time, and are not slaved to any local clock. As a result, it is not straightforward to map these arbitrary transitions into a synchronized data stream.

Synchronous Design—An In-depth Perspective

Synchronous Timing Basics

The following timing parameters characterize the timing of the Sequential circuit.



- The contamination (minimum) delay $t_{c\text{-}q,cd}$, and maximum propagation delay of the register $t_{c\text{-}q}$.
- The set-up (t_{su}) and hold time (t_{hold}) for the registers.
- The contamination delay $t_{logic,cd}$ and maximum delay t_{logic} of the combinational logic.
- t_{clk1} and t_{clk2}, corresponding to the position of the rising edge of the clock relative to a global reference.

$$T > t_{c-q} + t_{logic} + t_{su}$$

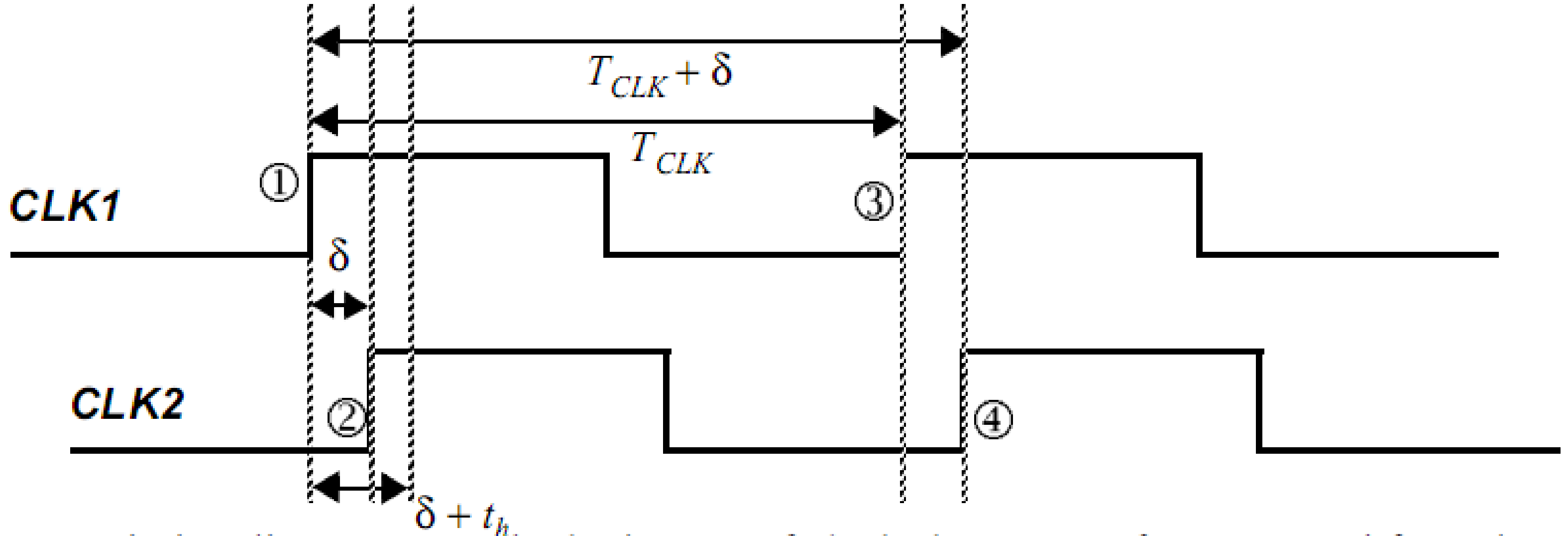
$$t_{hold} < t_{c-q,cd} + t_{logic,cd}$$

Clock Skew

The spatial variation in arrival time of a clock transition on an integrated circuit is commonly referred to as *clock skew*. The *clock skew* between two points *i* and *j* on a IC is given by $\delta(i,j) = t_i - t_j$, where t_i and t_j are the position of the rising edge of the clock with respect to a reference.

Clock skew is caused by static path-length mismatches in the clock load and by definition skew is constant from cycle to cycle. That is, if in one cycle *CLK2* lagged *CLK1* by δ, then on the next cycle it will lag it by the same amount. It is important to note that clock skew does not result in clock period variation, but rather phase shift.

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in



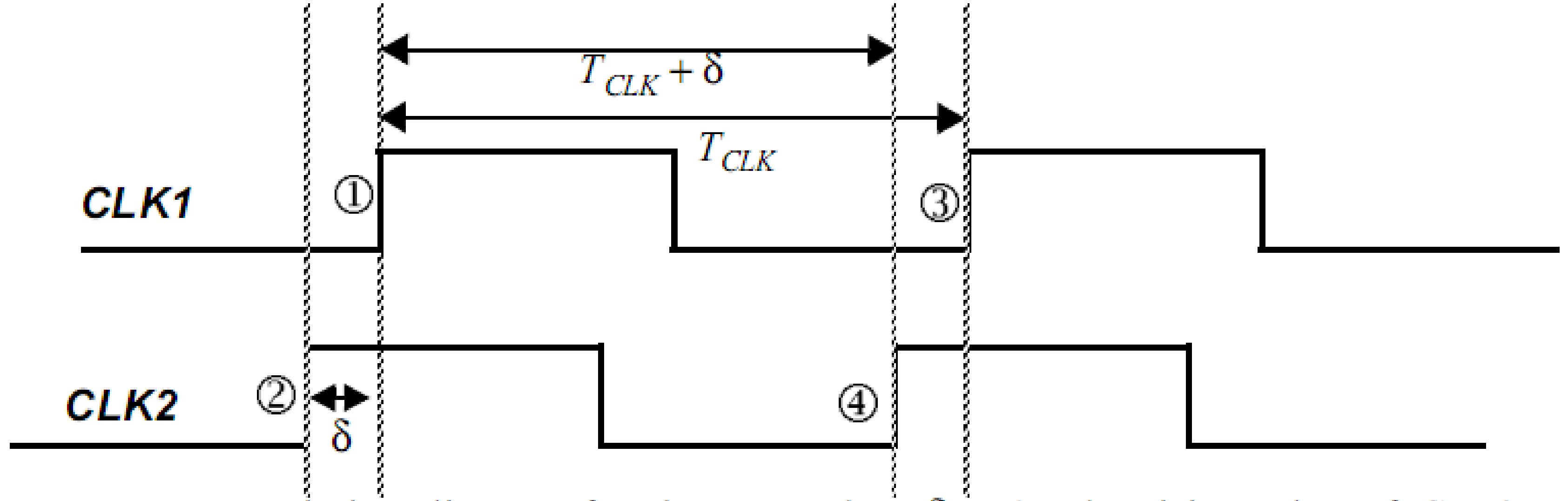
Timing diagram to study the impact of clock skew on performance and funcationality. In this sample timing diagram, $\delta > 0$.

$$T > t_{c-q} + t_{logic} + t_{su}$$

$$t_{hold} < t_{c-q,\,cd} + t_{logic,\,cd}$$

$$T + \delta \geq t_{c-q} + t_{logic} + t_{su}$$

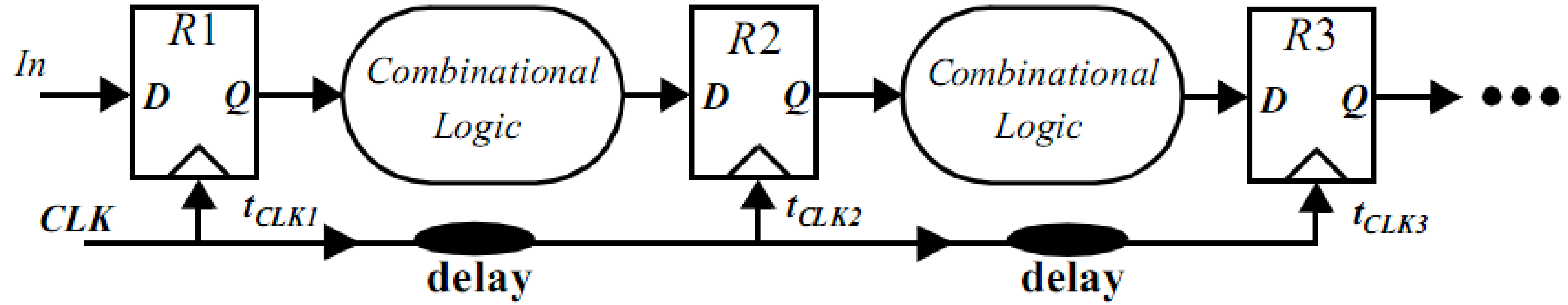
$$\delta + t_{hold} < t_{(c-q,\,cd)} + t_{(logic,\,cd)}$$



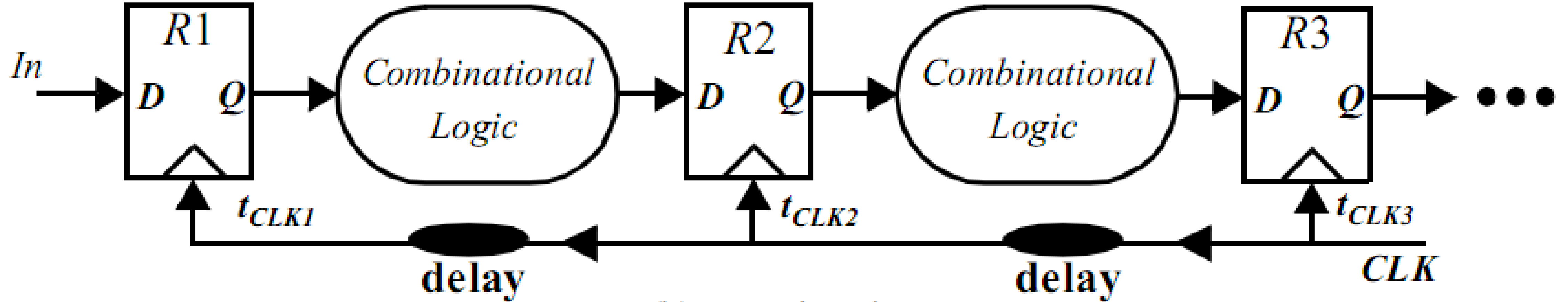
Timing diagram for the case when $\delta < 0$. The rising edge of $CLK2$ arrives earlier than the edge of $CLK1$.

$$T \geq t_{c-q} + t_{logic} + t_{su} - \delta$$

$$\delta < t_{(c-q,\,cd)} + t_{(logic,\,cd)} - t_{hold}$$

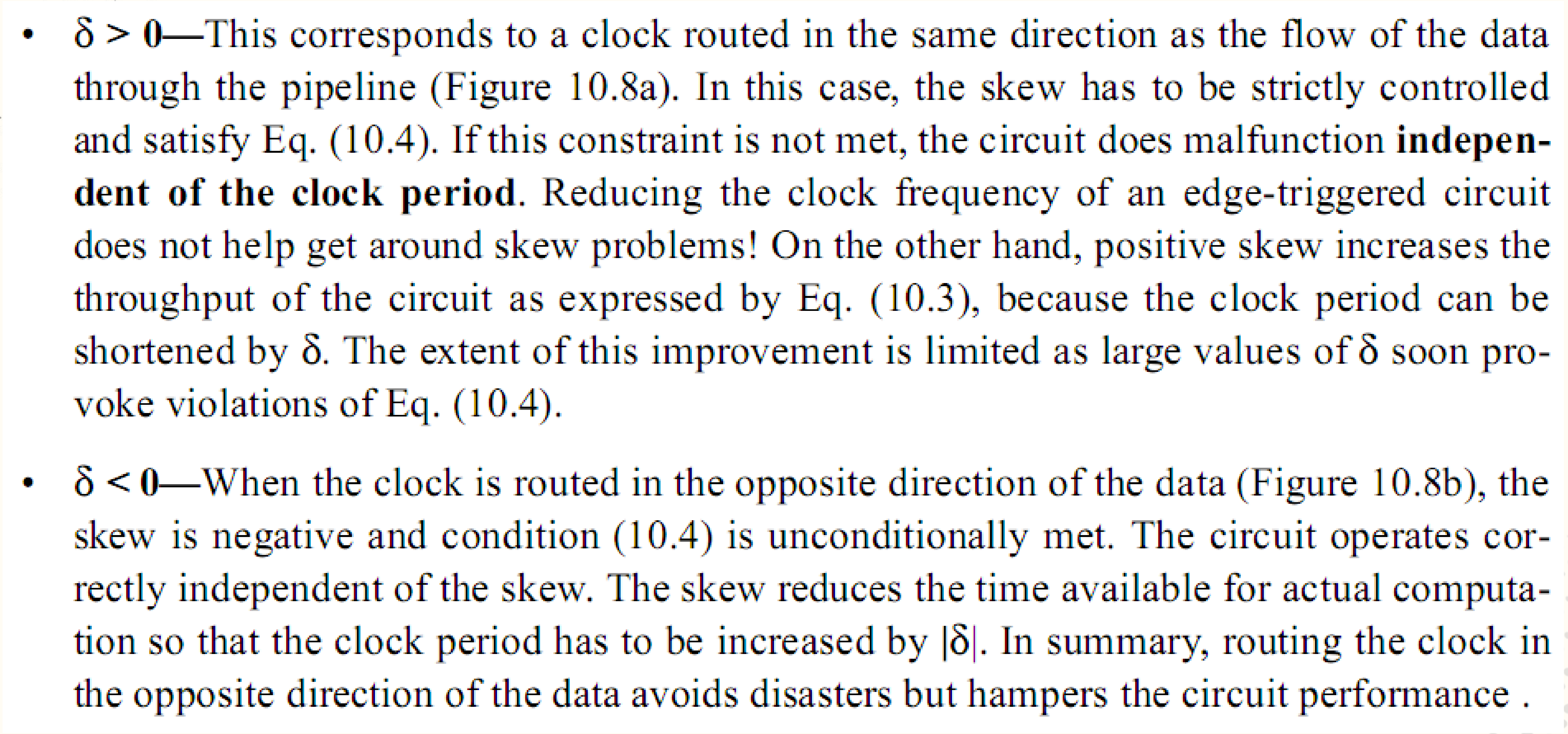


(a) Positive skew



(b) Negative skew

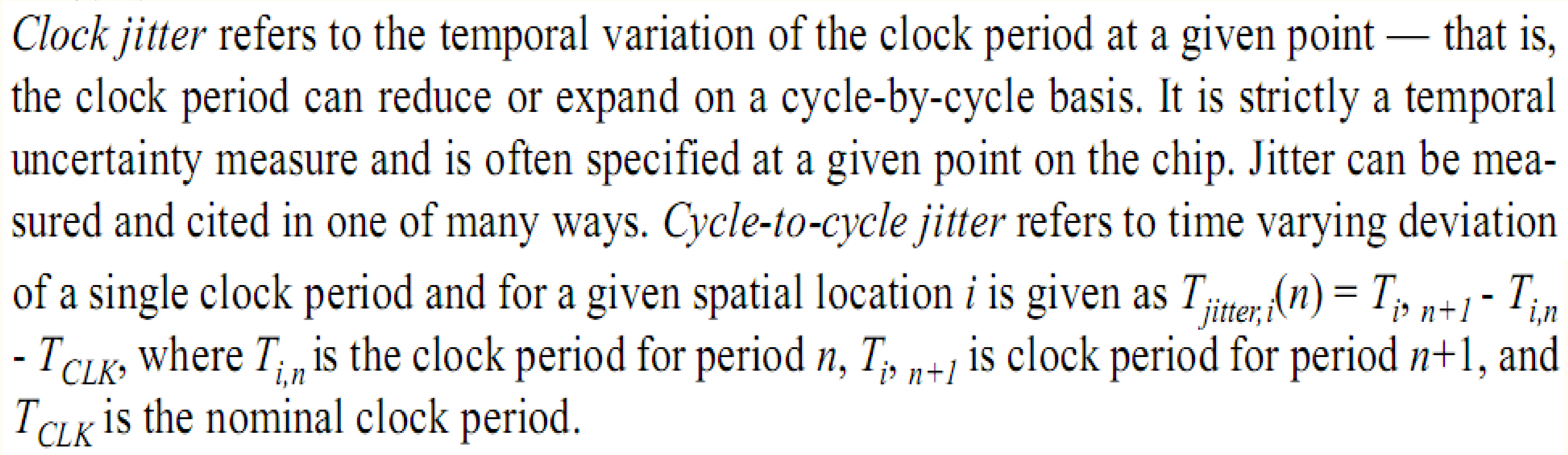
Positive and negative clock skew.



- $\delta > 0$—This corresponds to a clock routed in the same direction as the flow of the data through the pipeline (Figure 10.8a). In this case, the skew has to be strictly controlled and satisfy Eq. (10.4). If this constraint is not met, the circuit does malfunction **independent of the clock period**. Reducing the clock frequency of an edge-triggered circuit does not help get around skew problems! On the other hand, positive skew increases the throughput of the circuit as expressed by Eq. (10.3), because the clock period can be shortened by δ. The extent of this improvement is limited as large values of δ soon provoke violations of Eq. (10.4).

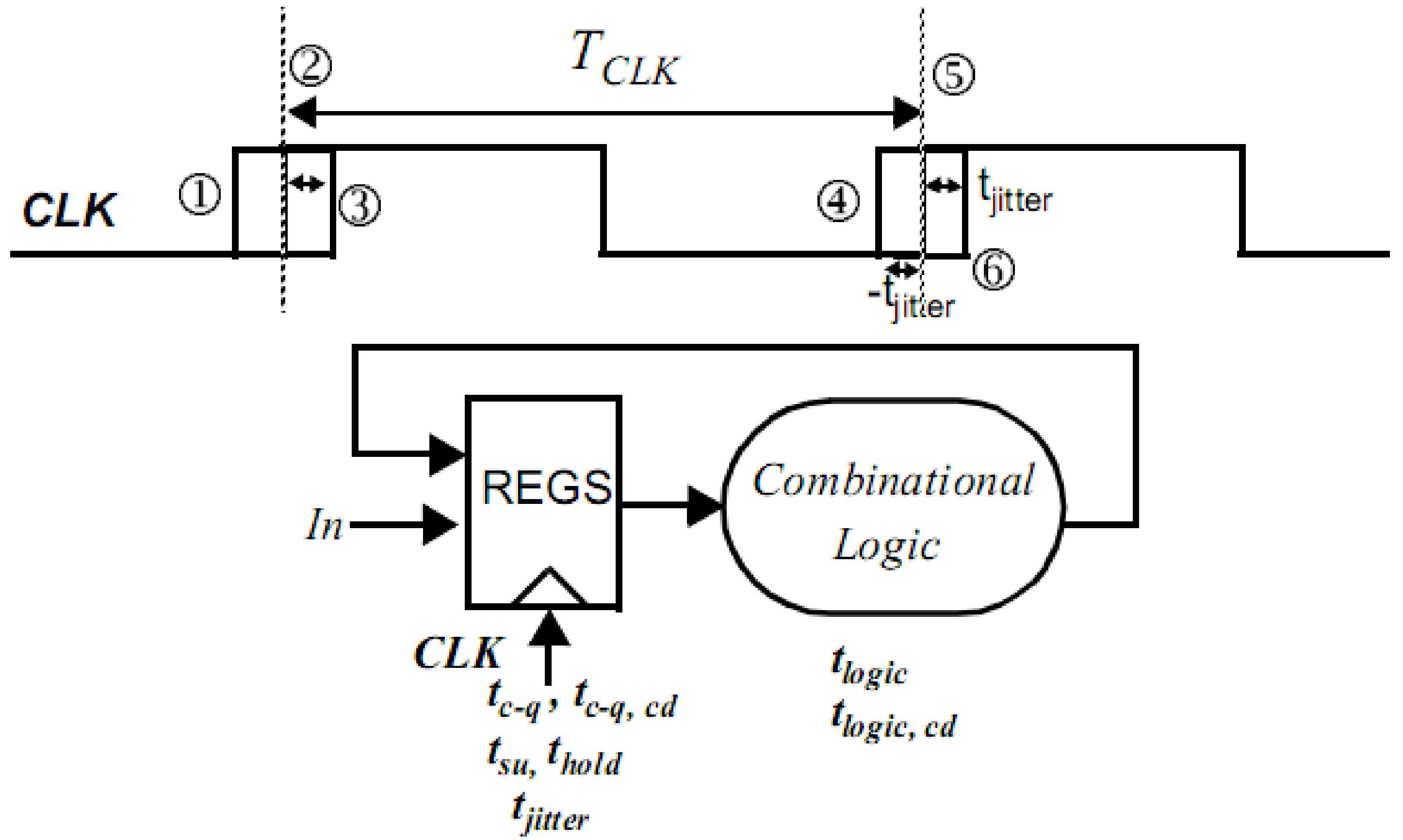
- $\delta < 0$—When the clock is routed in the opposite direction of the data (Figure 10.8b), the skew is negative and condition (10.4) is unconditionally met. The circuit operates correctly independent of the skew. The skew reduces the time available for actual computation so that the clock period has to be increased by $|\delta|$. In summary, routing the clock in the opposite direction of the data avoids disasters but hampers the circuit performance .

Clock Jitter



Clock jitter refers to the temporal variation of the clock period at a given point — that is, the clock period can reduce or expand on a cycle-by-cycle basis. It is strictly a temporal uncertainty measure and is often specified at a given point on the chip. Jitter can be measured and cited in one of many ways. *Cycle-to-cycle jitter* refers to time varying deviation of a single clock period and for a given spatial location *i* is given as $T_{jitter,i}(n) = T_{i,\ n+1} - T_{i,n} - T_{CLK}$, where $T_{i,n}$ is the clock period for period n, $T_{i,\ n+1}$ is clock period for period $n+1$, and T_{CLK} is the nominal clock period.

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in



Circuit for studying the impact of *jitter* on performance.

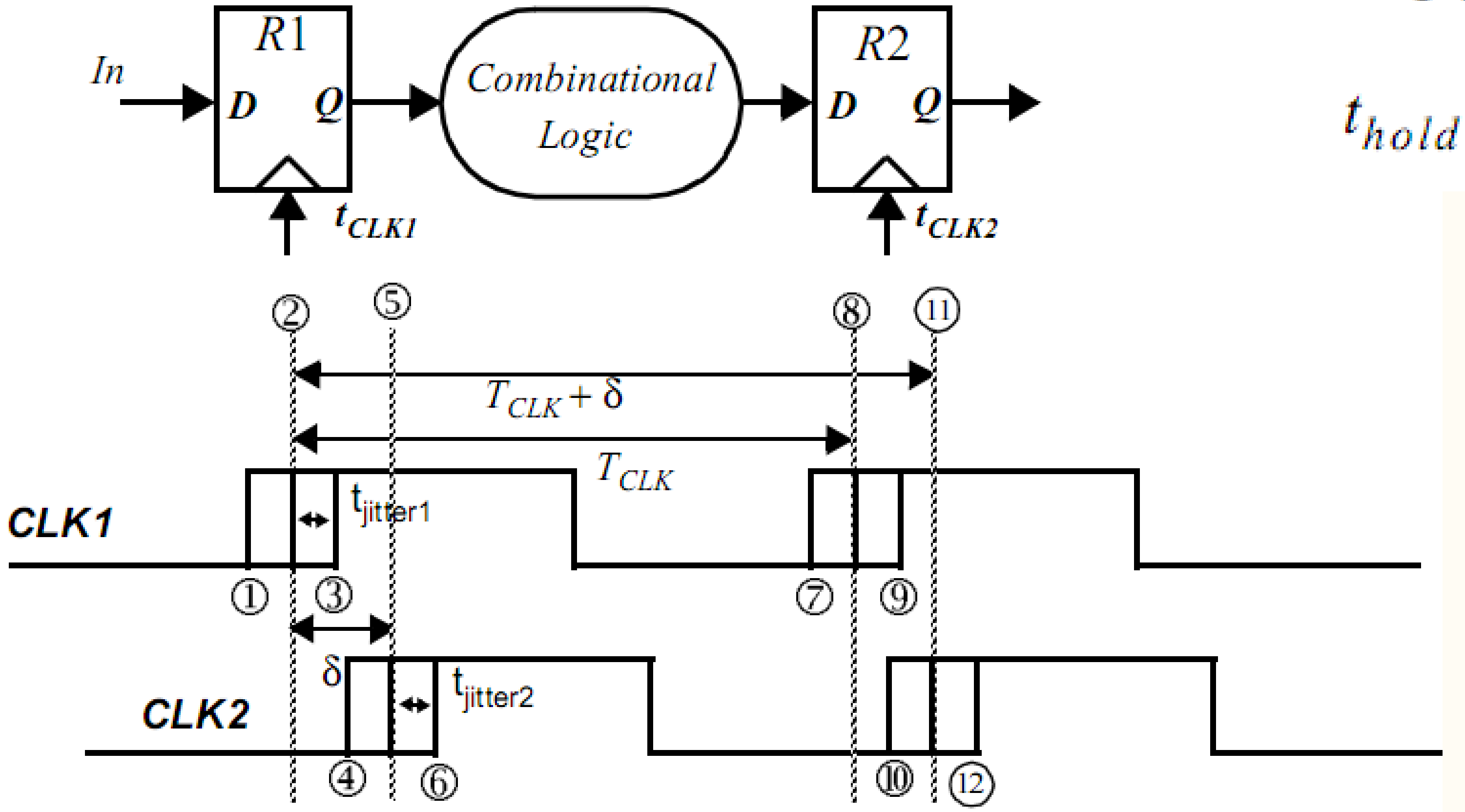
As a result, the total time available to complete the operation is reduced by 2 t_{jiiter} in the worst case and is given by

$$T_{CLK} - 2t_{jitter} \geq t_{c-q} + t_{logic} + t_{su} \quad \text{or} \quad T \geq t_{c-q} + t_{logic} + t_{su} + 2t_{jitter}$$

Impact of Skew and Jitter on Performance

$$T > t_{c-q} + t_{logic} + t_{su}$$

$$t_{hold} < t_{c-q,cd} + t_{logic,cd}$$



Sequential circuit to study the impact of skew and jitter on *edge-triggered* systems. In this example, a positive *skew* (δ) is assumed.

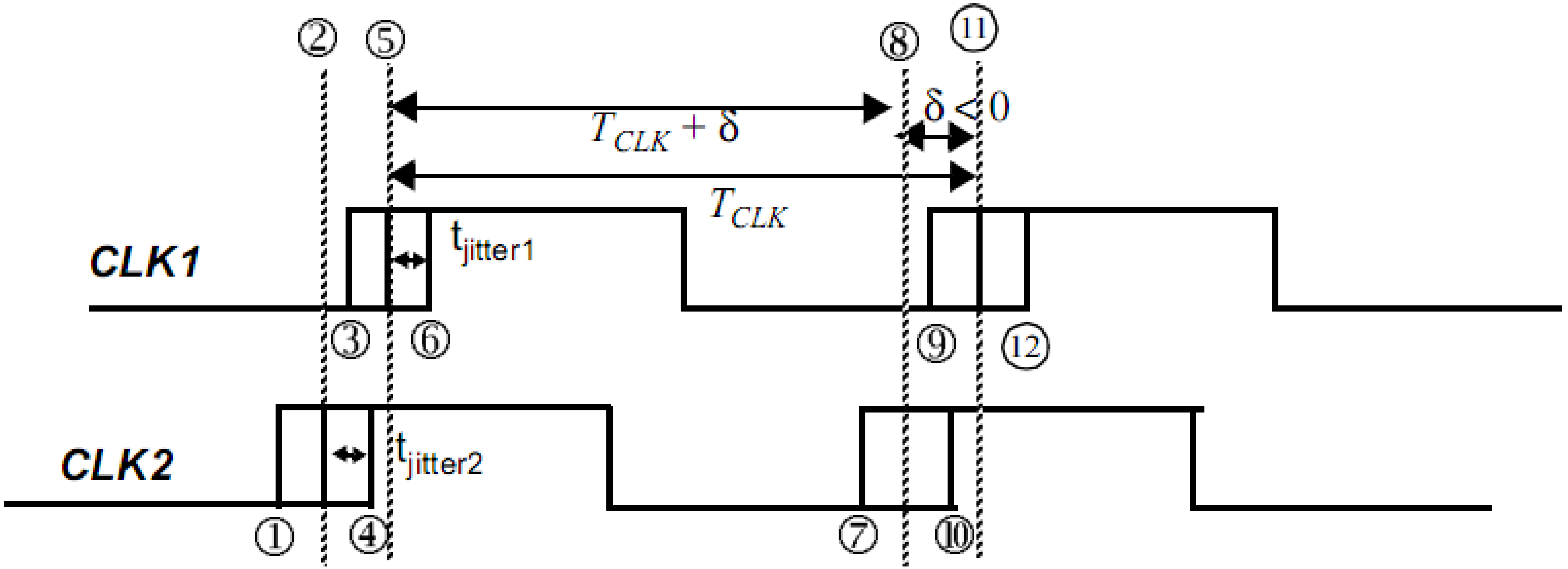
$$T_{CLK} + \delta - t_{jitter1} - t_{jitter2} \geq t_{c-q} + t_{logic} + t_{su}$$

or $$T \geq t_{c-q} + t_{logic} + t_{su} - \delta + t_{jitter1} + t_{jitter2}$$

$$\delta + t_{hold} + t_{jitter1} + t_{jitter2} < t_{(c-q,\,cd)} + t_{(\text{logic},\,cd)}$$

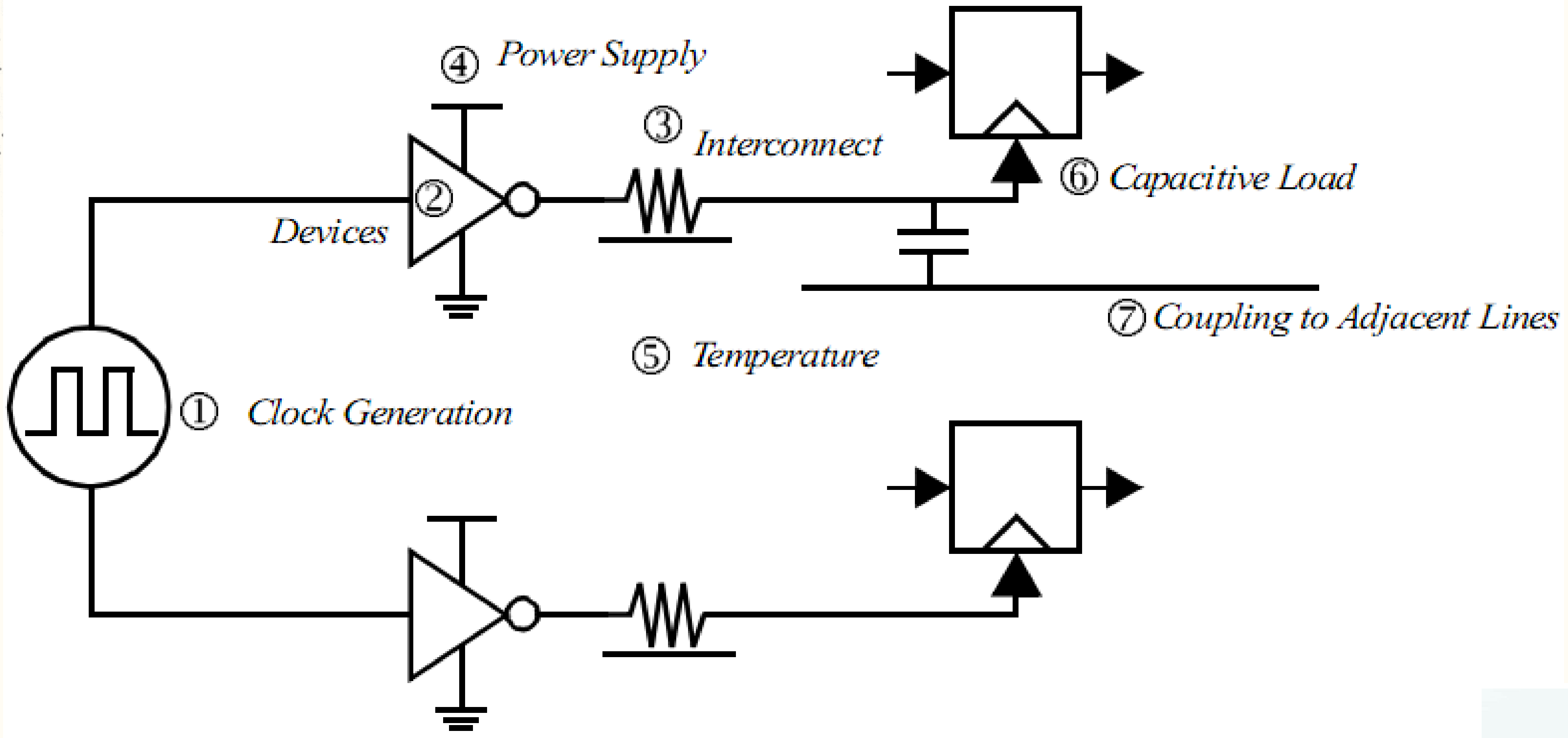
or

$$\delta < t_{(c-q,\,cd)} + t_{(\text{logic},\,cd)} - t_{hold} - t_{jitter1} - t_{jitter2}$$



Consider a negative clock *skew* (δ) and the *skew* is assumed to be larger than the *jitter*.

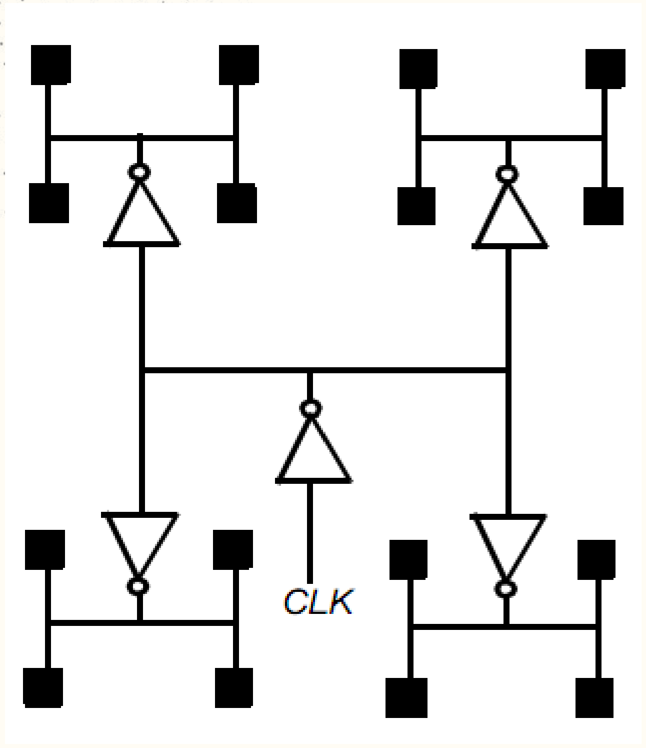
Sources of Skew and Jitter



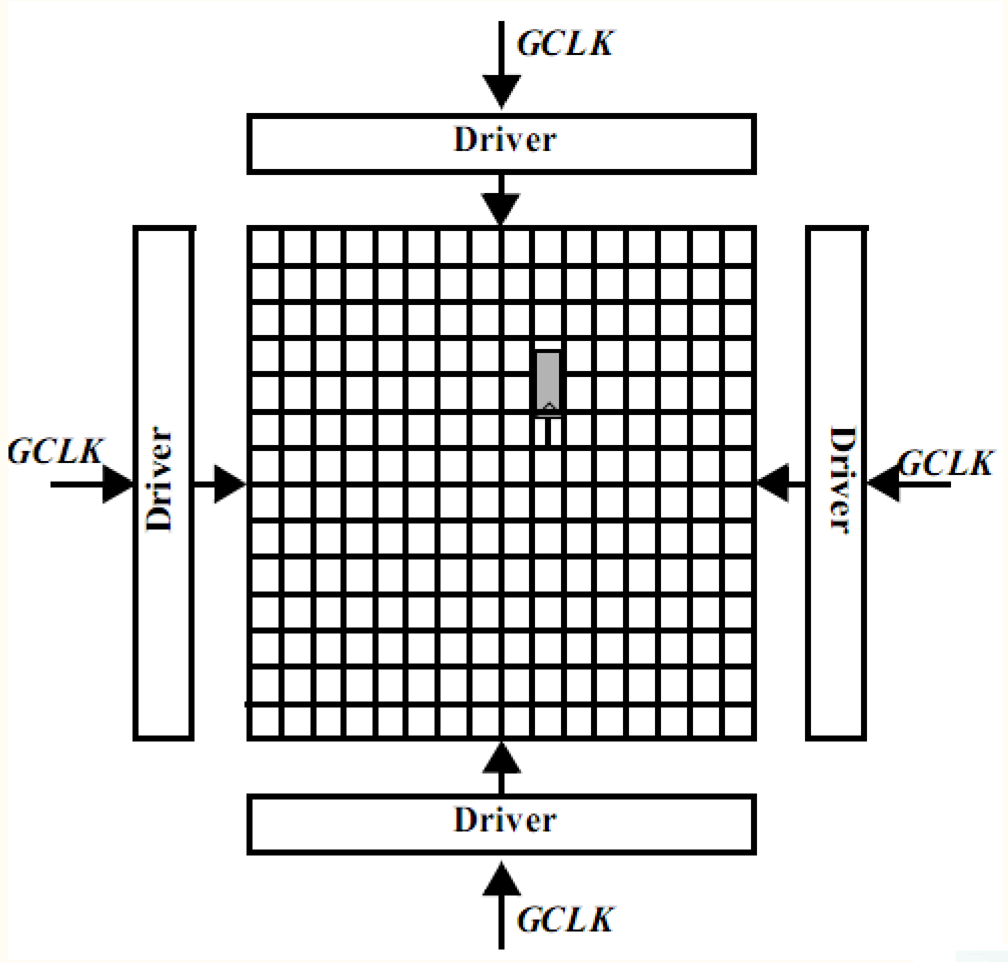
Skew and jitter sources in synchronous clock distribution.

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in

Clock-Distribution Techniques

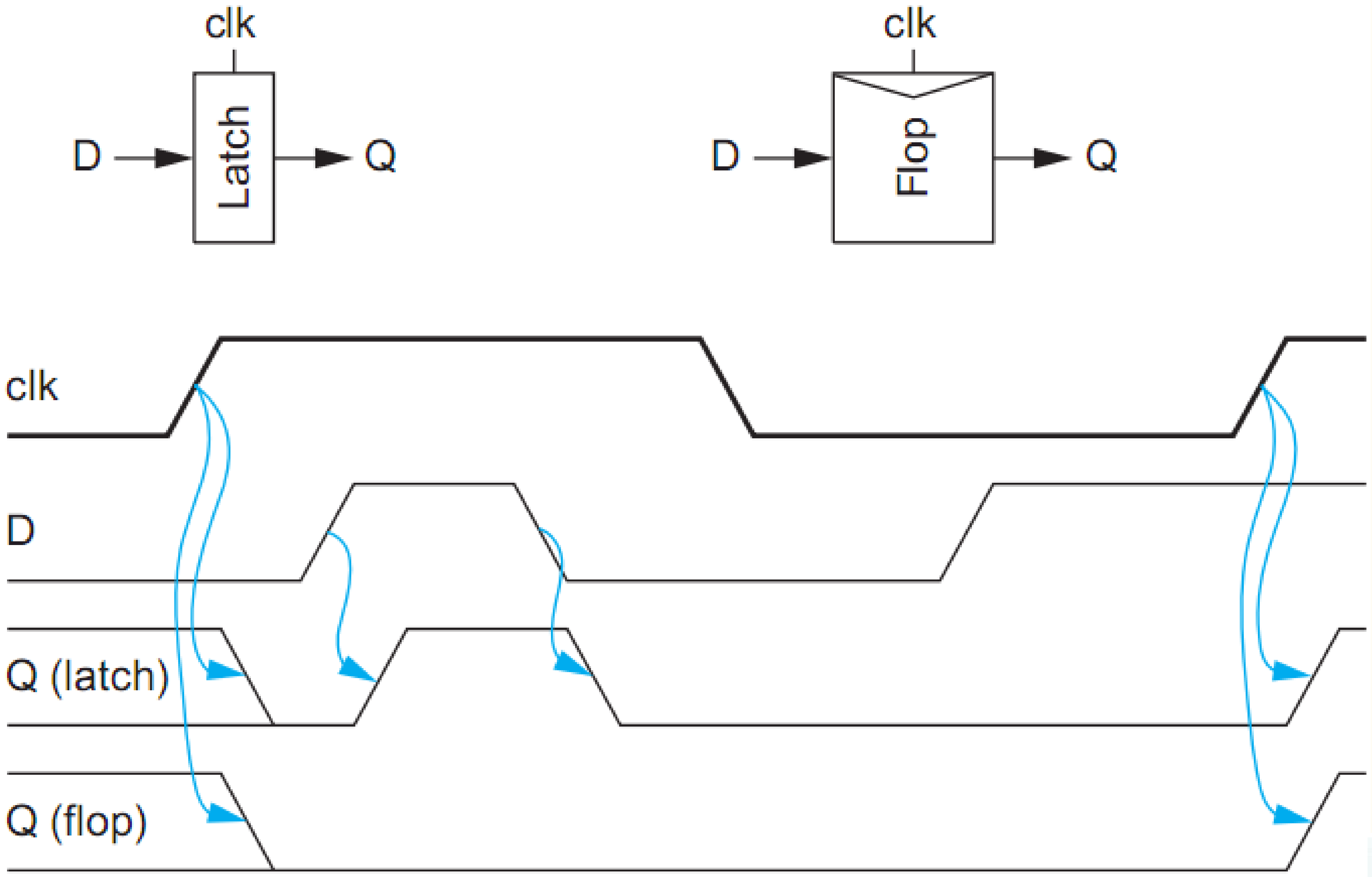


H-tree clock - distribution Network for 16 leaf nodes



Grid structures

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in

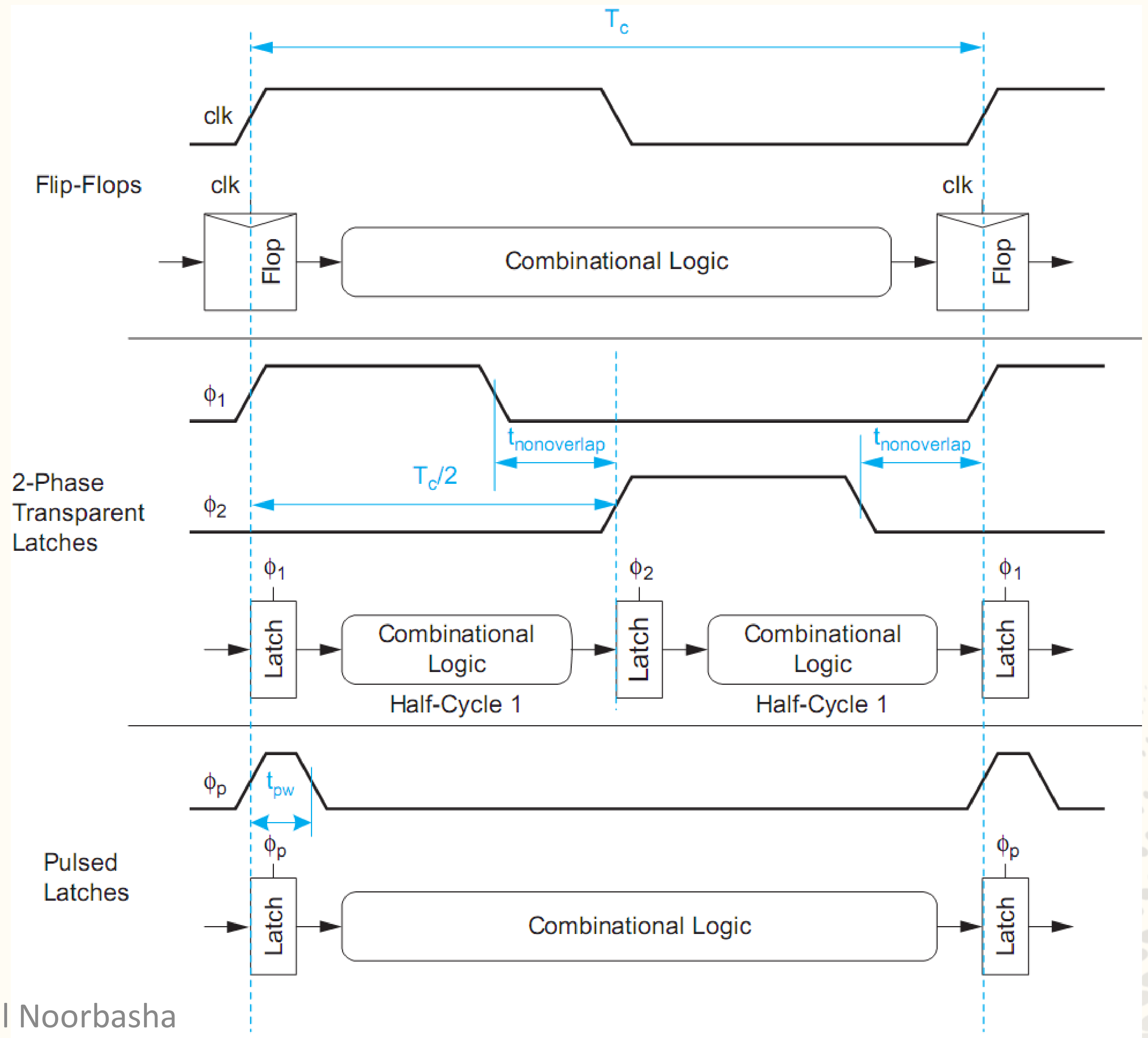


Latches and flip-flops

Dr. Fazal Noorbasha

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in

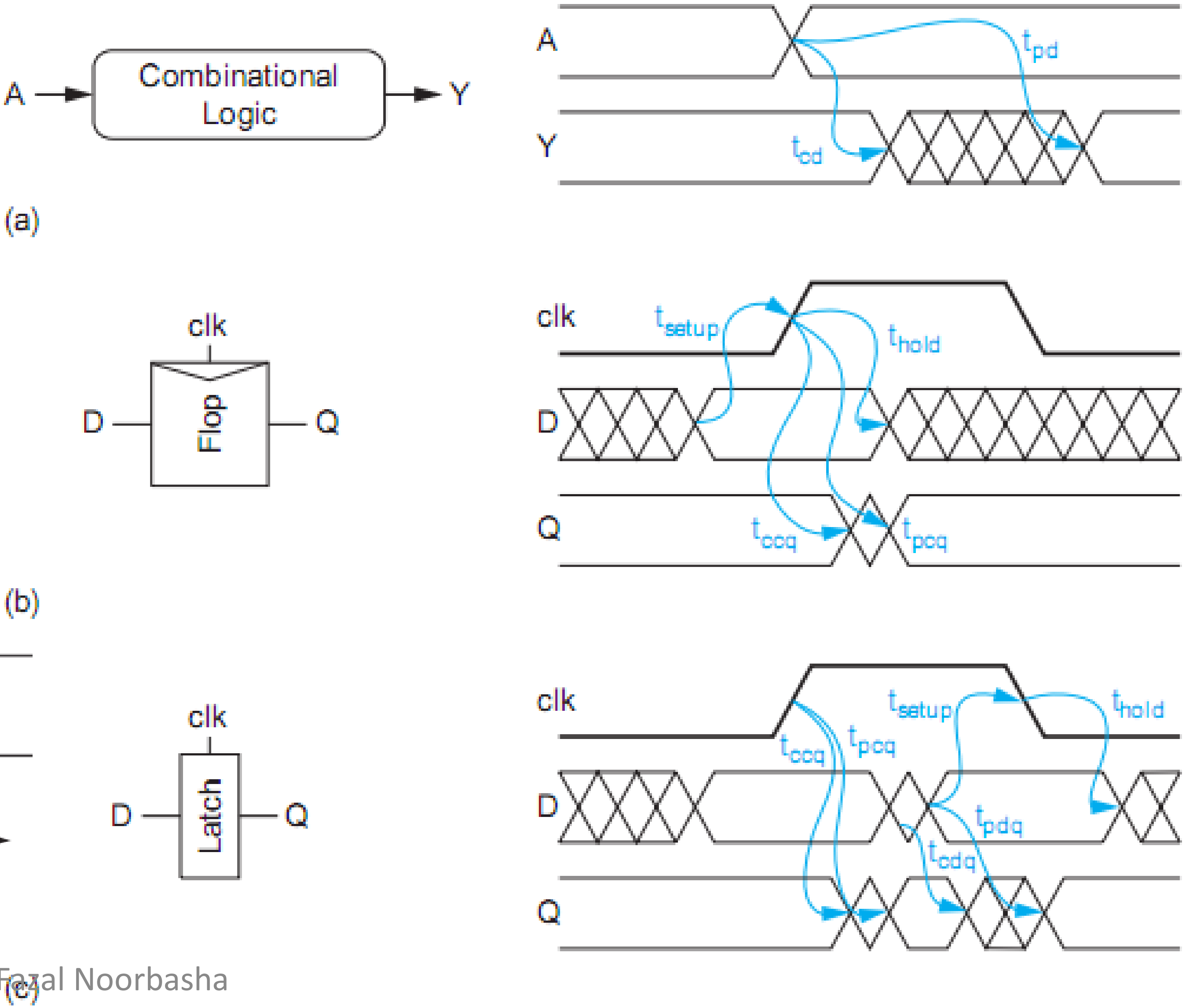
Static sequencing methods

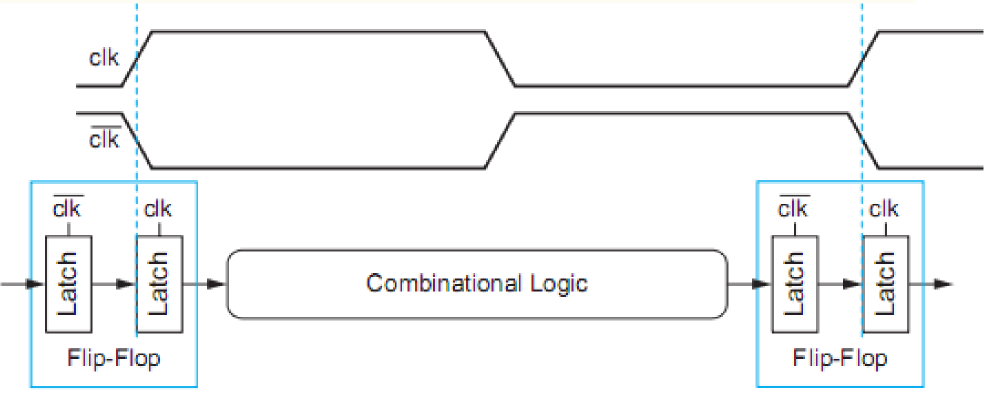


Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA

Sequencing element timing notation

Term	Name
t_{pd}	Logic Propagation Delay
t_{cd}	Logic Contamination Delay
t_{pcq}	Latch/Flop Clock-to-*Q* Propagation Delay
t_{ccq}	Latch/Flop Clock-to-*Q* Contamination Delay
t_{pdq}	Latch *D*-to-*Q* Propagation Delay
t_{cdq}	Latch *D*-to-*Q* Contamination Delay
t_{setup}	Latch/Flop Setup Time
t_{hold}	Latch/Flop Hold Time



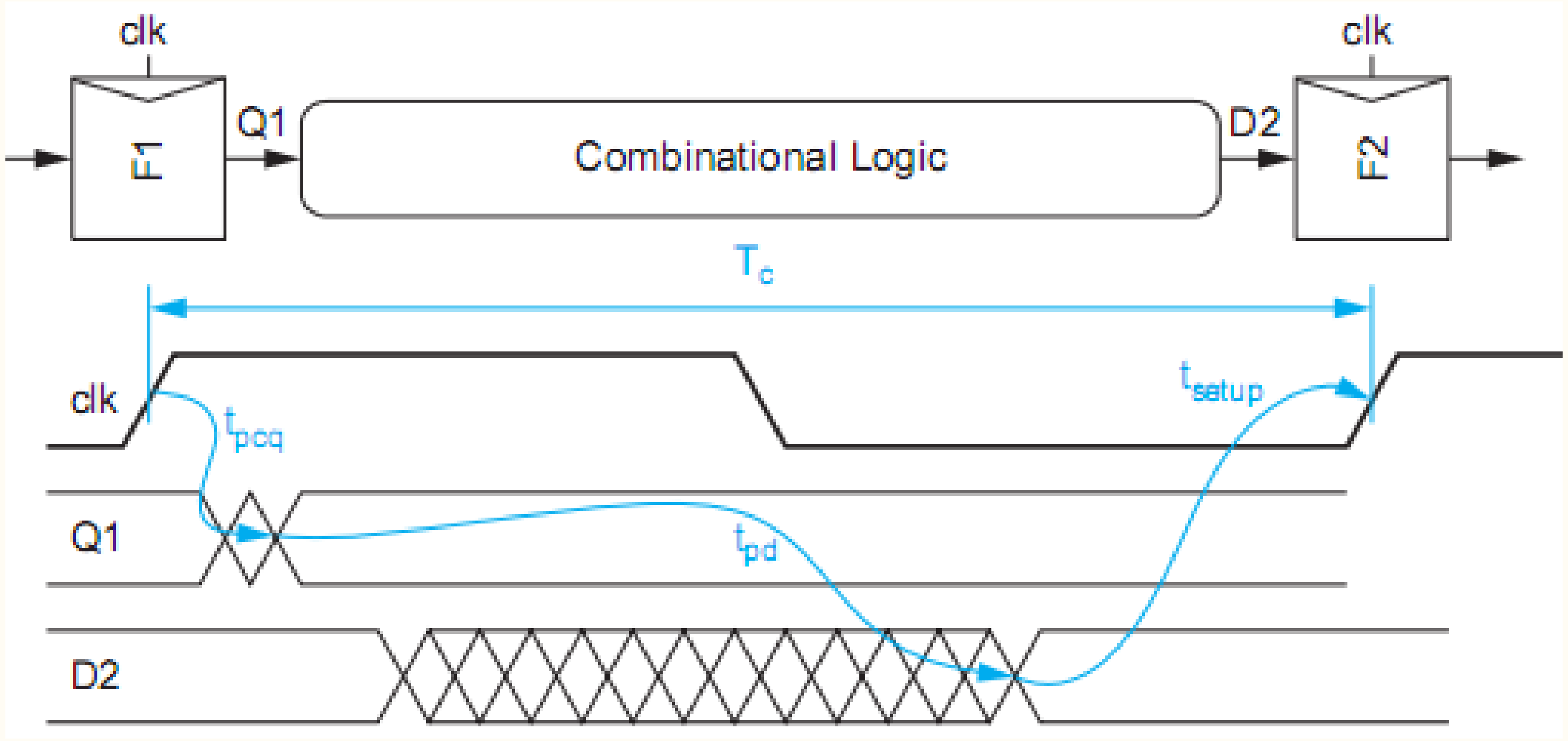


Dr. Fazal Noorbasha

Max-Delay Constraints

$$T_c \geq t_{pcq} + t_{pd} + t_{\text{setup}}$$

$$t_{pd} \leq T_c - \underbrace{\left(t_{\text{setup}} + t_{pcq}\right)}_{\text{sequencing overhead}}$$

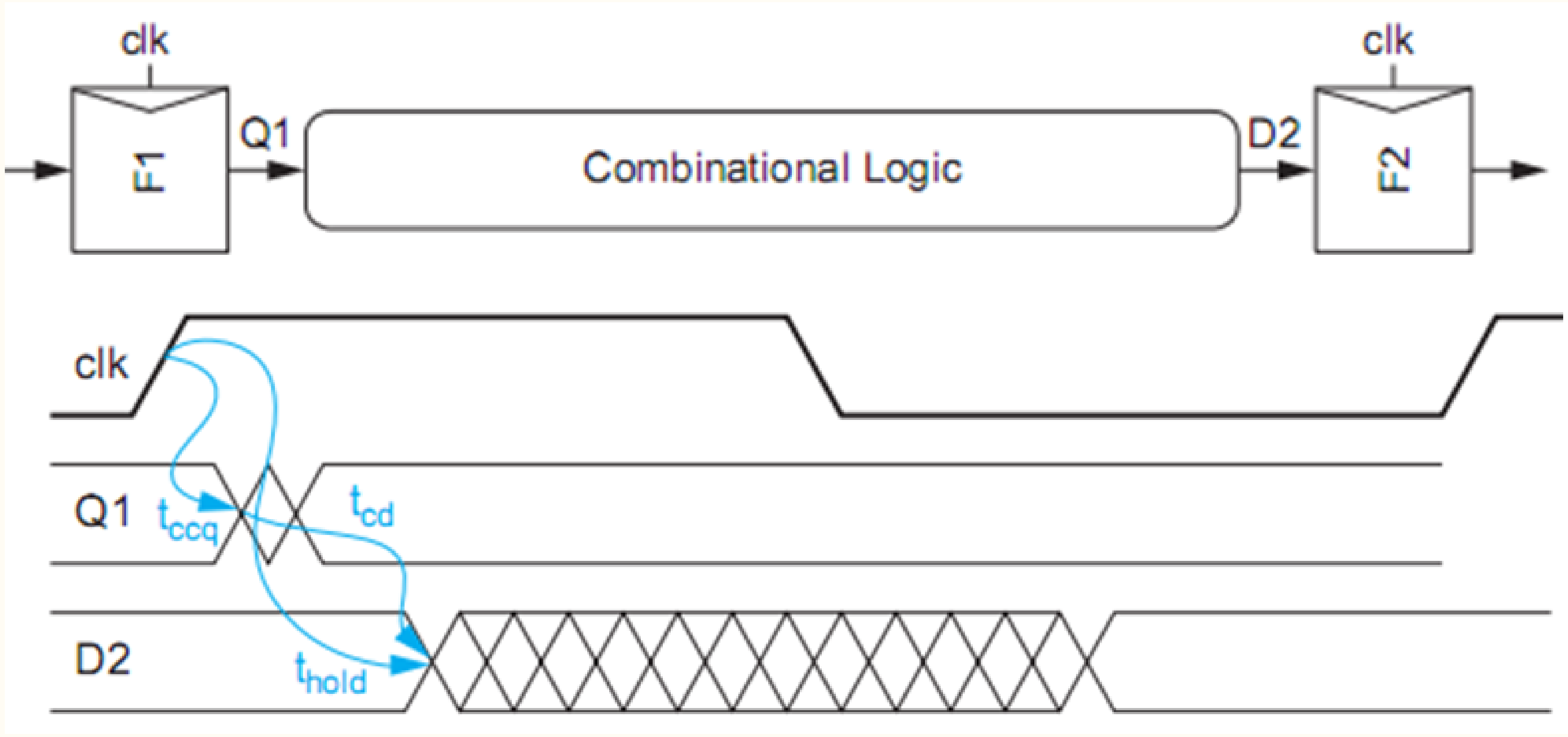


Flip-flop max-delay constraint

If the combinational logic delay is too great, the receiving element will miss its setup time and sample the wrong value. This is called a **setup time failure or max-delay failure**. It can be solved by redesigning the logic to be faster or by increasing the clock period.

Min-Delay Constraints

$$t_{cd} \geq t_{\text{hold}} - t_{ccq}$$



Flip-flop latch min-delay constraint

If the hold time is large and the contamination delay is small, data can incorrectly propagate through two successive elements on one clock edge, corrupting the state of the system. This is called a **race condition, hold-time failure, or min-delay failure**. It can only be fixed by redesigning the logic, not by slowing the clock.

Behavior of Bistable Elements

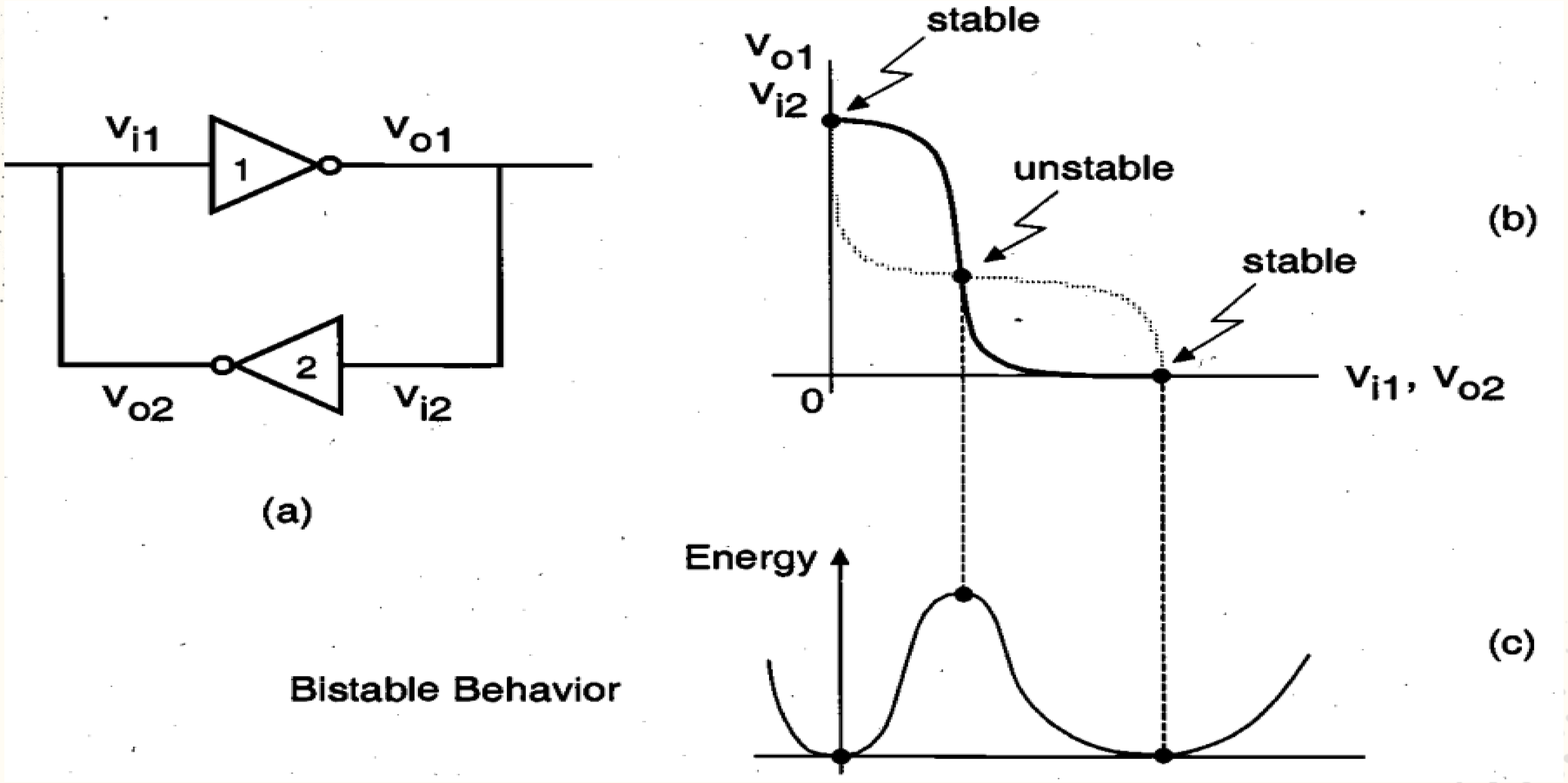
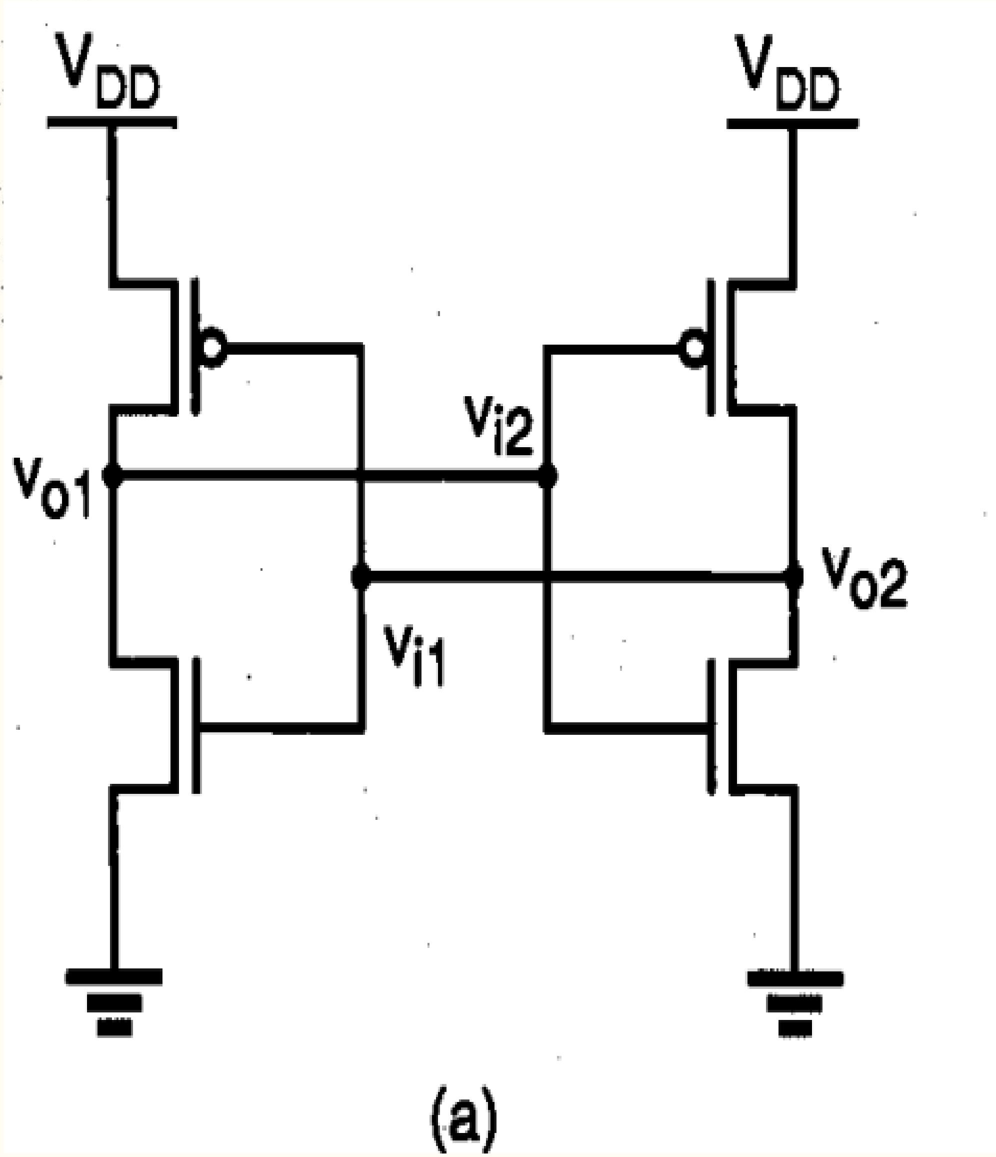


Fig: Static behavior of the two-inverter basic bistable element: (a) Circuit schematic. (b) Intersecting voltage transfer curves of the two inverters, showing the three possible operating points. (c) Qualitative view of the potential energy levels corresponding to the three operating points.

Behavior of Bistable Elements



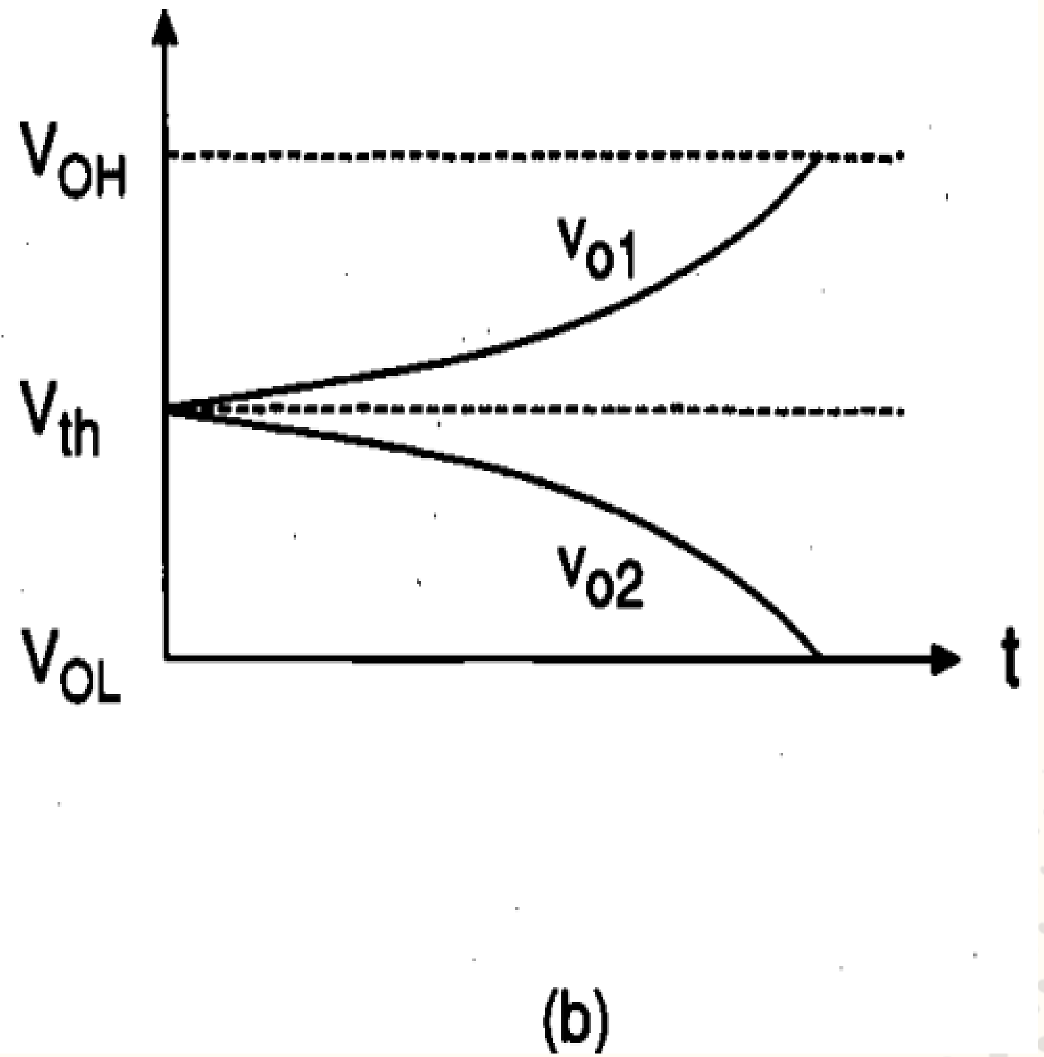
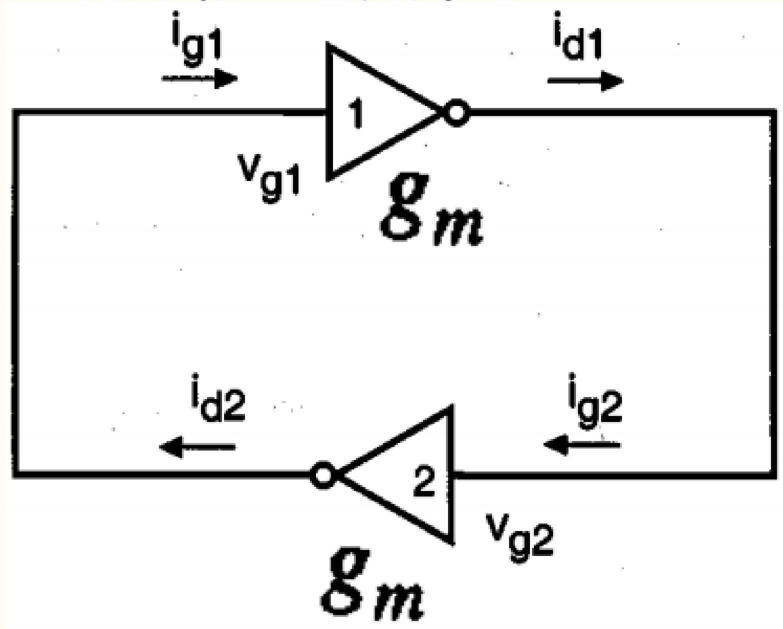


Fig: (a) Circuit diagram of a CMOS bistable element. (b) One possibility for the expected time-domain behavior of the output voltages, if the circuit is initially set at its unstable operating point.

$$i_{g1} = i_{d2} = g_m v_{g2}$$
$$i_{g2} = i_{d1} = g_m v_{g1} \tag{8.1}$$

Here, g_m represents the small-signal transconductance of the inverter. The gate voltages of both inverters can be expressed in terms of the gate charges, q_1 and q_2.



Small-signal input and output currents of the inverters.

$$v_{g1} = \frac{q_1}{C_g} \qquad v_{g2} = \frac{q_2}{C_g} \tag{8.2}$$

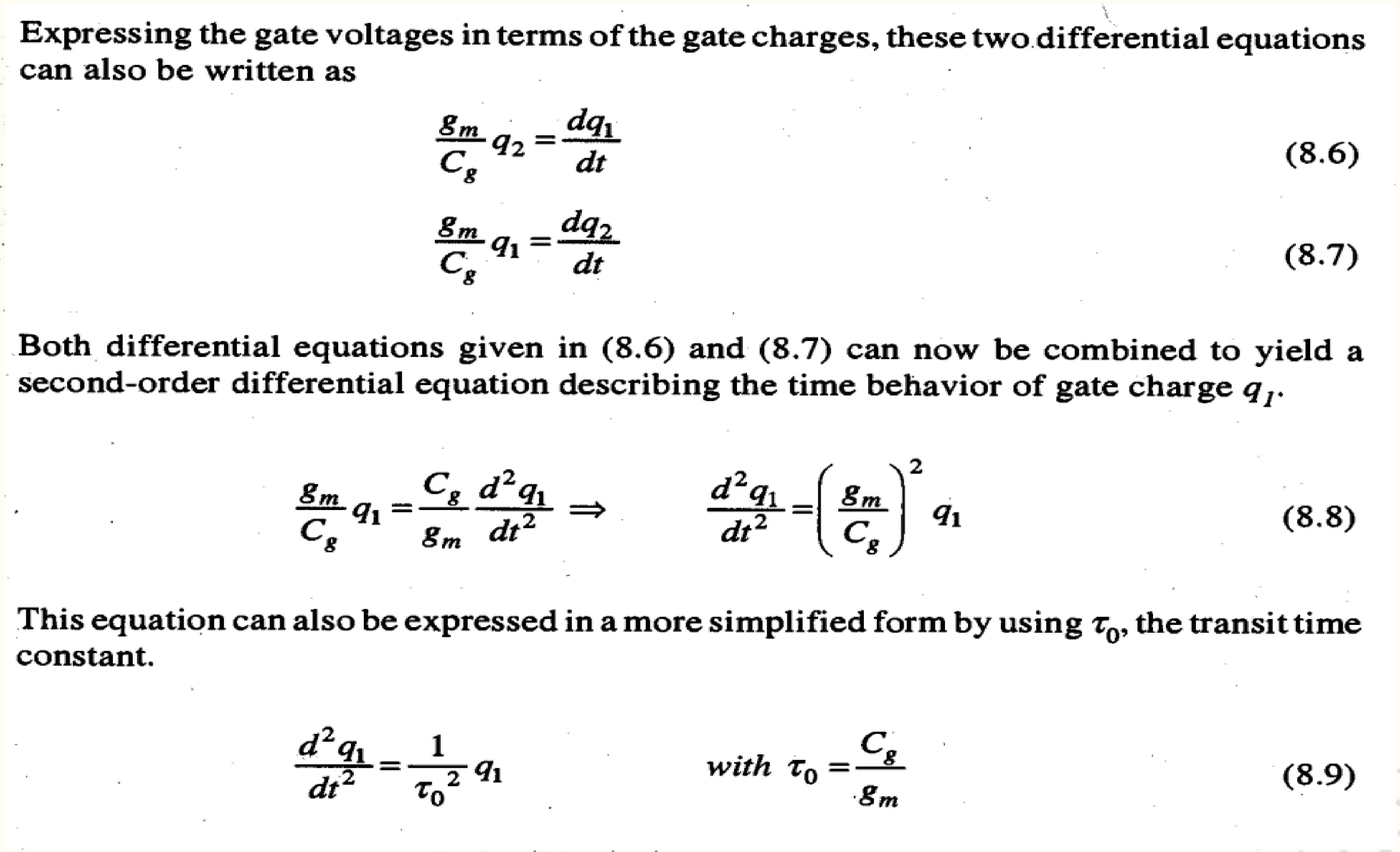
Note that the small-signal gate current of each inverter can be written as a function of the time derivative of its small-signal gate voltage, as follows.

$$i_{g1} = C_g \frac{dv_{g1}}{dt}$$
$$i_{g2} = C_g \frac{dv_{g2}}{dt} \tag{8.3}$$

Combining (8.1) with (8.3), we obtain:

$$g_m v_{g2} = C_g \frac{dv_{g1}}{dt} \tag{8.4}$$

$$g_m v_{g1} = C_g \frac{dv_{g2}}{dt} \tag{8.5}$$



Expressing the gate voltages in terms of the gate charges, these two differential equations can also be written as

$$\frac{g_m}{C_g} q_2 = \frac{dq_1}{dt} \tag{8.6}$$

$$\frac{g_m}{C_g} q_1 = \frac{dq_2}{dt} \tag{8.7}$$

Both differential equations given in (8.6) and (8.7) can now be combined to yield a second-order differential equation describing the time behavior of gate charge q_1.

$$\frac{g_m}{C_g} q_1 = \frac{C_g}{g_m} \frac{d^2 q_1}{dt^2} \Rightarrow \qquad \frac{d^2 q_1}{dt^2} = \left(\frac{g_m}{C_g} \right)^2 q_1 \tag{8.8}$$

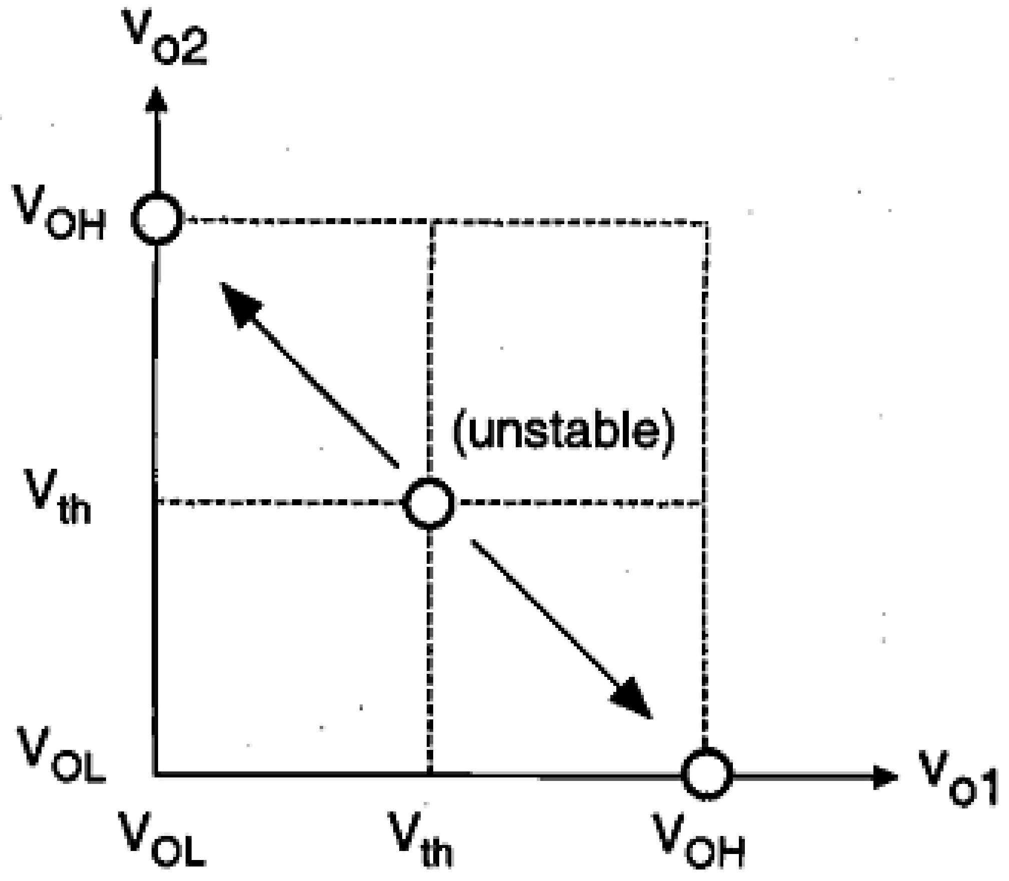
This equation can also be expressed in a more simplified form by using τ_0, the transit time constant.

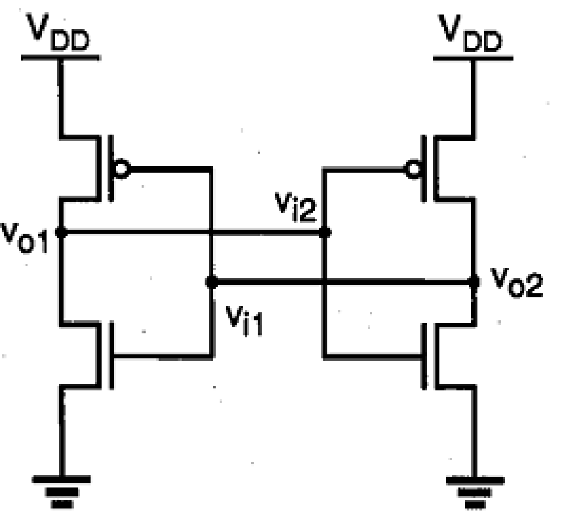
$$\frac{d^2 q_1}{dt^2} = \frac{1}{\tau_0{}^2} q_1 \qquad \textit{with} \;\; \tau_0 = \frac{C_g}{g_m} \tag{8.9}$$

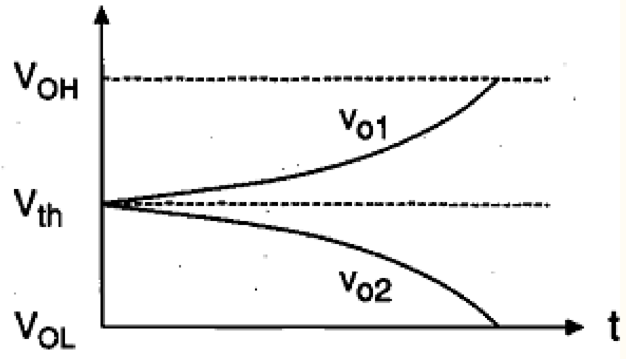
Phase-plane representation of the bistable circuit behavior.

v_{o1}: $V_{th} \rightarrow V_{OH}$ or V_{OL} .

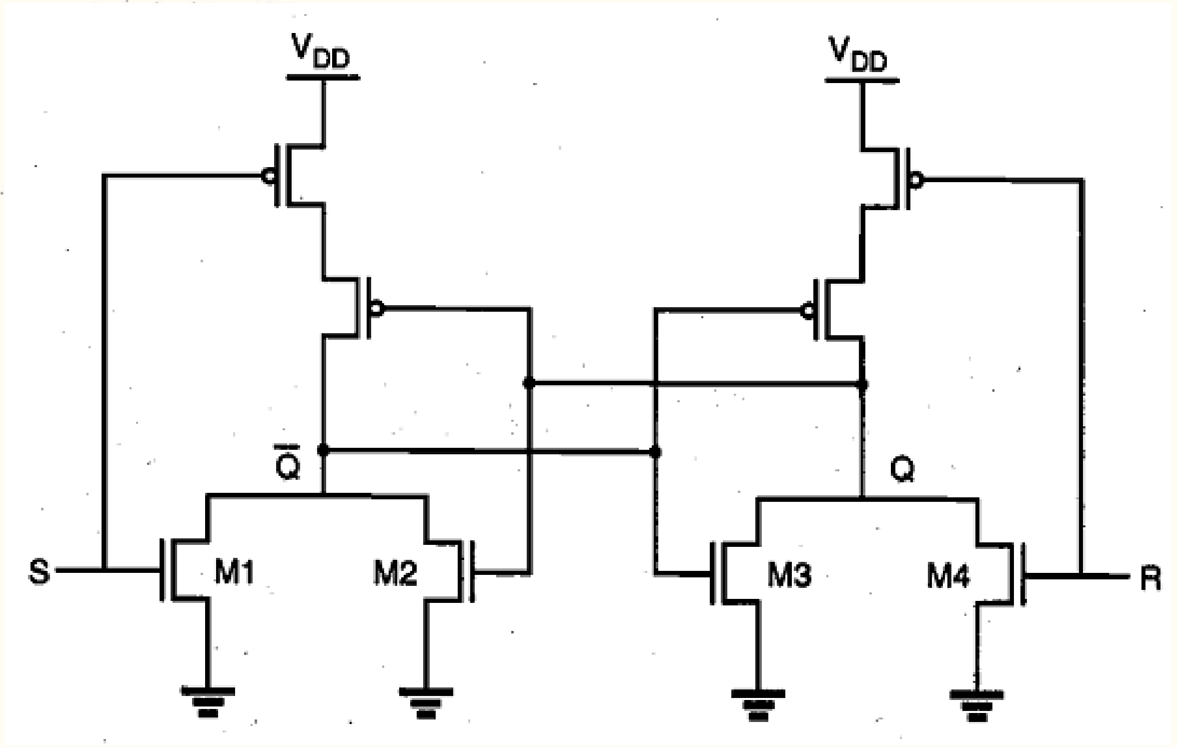
v_{o2}: $V_{th} \rightarrow V_{OL}$ or V_{OH}



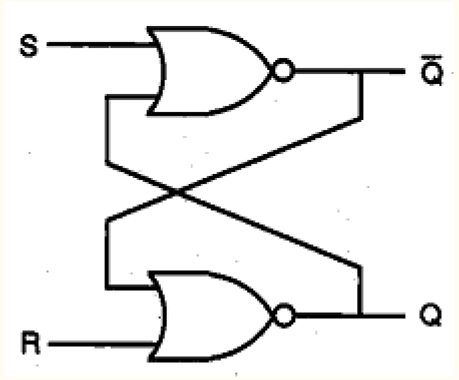




The SR Latch Circuit

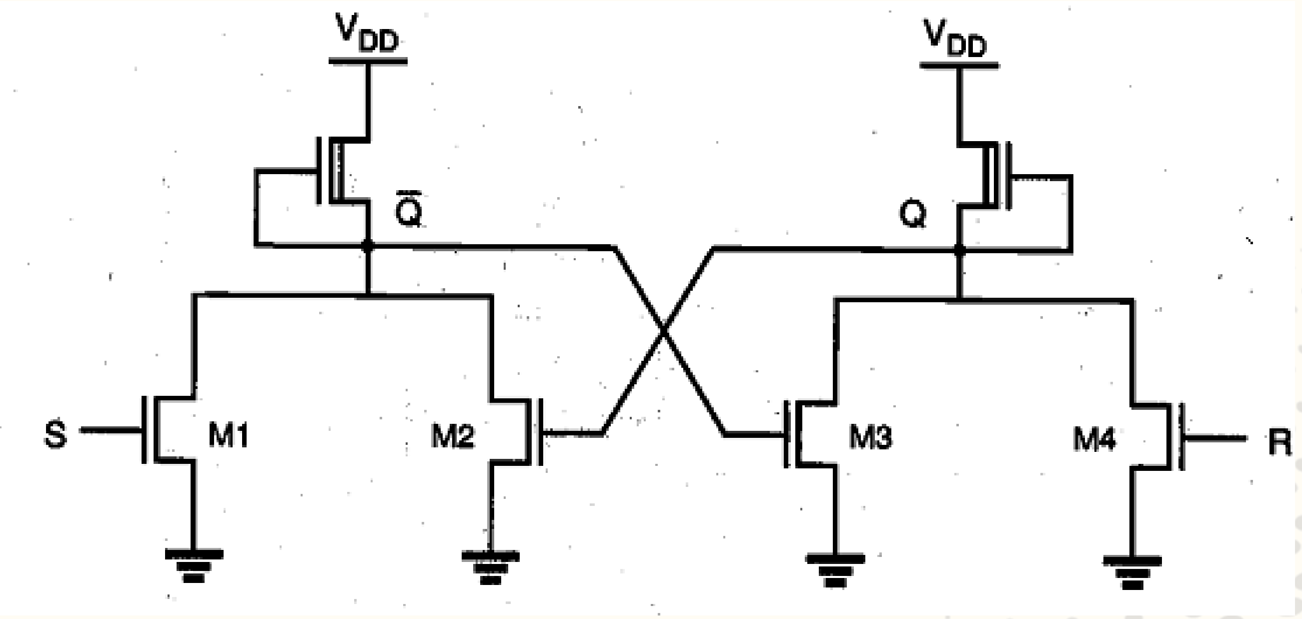


CMOS SR latch circuit based on NOR2 gates.



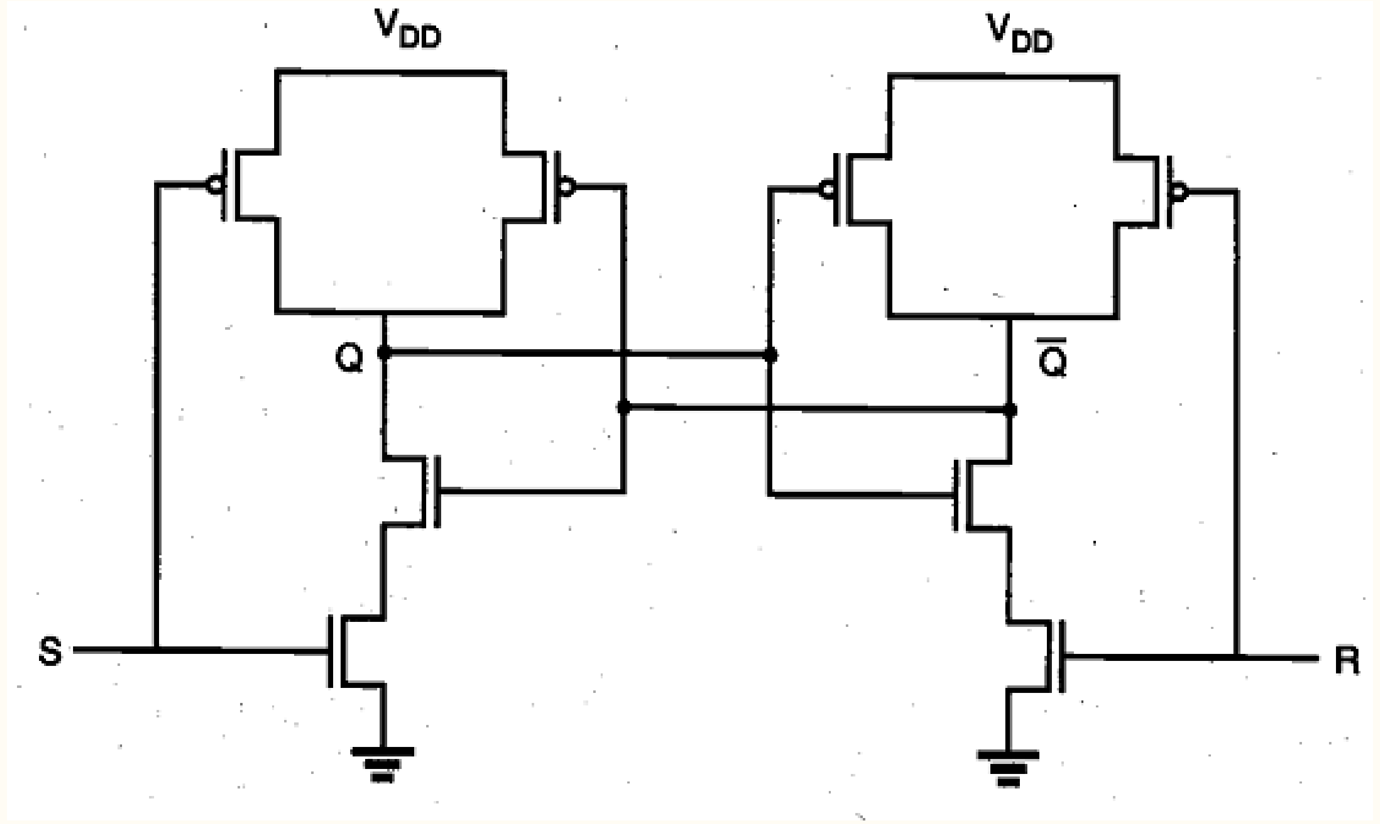
S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Operation
0	0	Q_n	$\overline{Q_n}$	hold
1	0	1	0	set
0	1	0	1	reset
1	1	0	0	not allowed

S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Operation
V_{OH}	V_{OL}	V_{OH}	V_{OL}	M1 and M2 on, M3 and M4 off
V_{OL}	V_{OH}	V_{OL}	V_{OH}	M1 and M2 off, M3 and M4 on
V_{OL}	V_{OL}	V_{OH}	V_{OL}	$M1$ and M4 off, M2 on, or
V_{OL}	V_{OL}	V_{OL}	V_{OH}	M1 and M4 off, M3 on

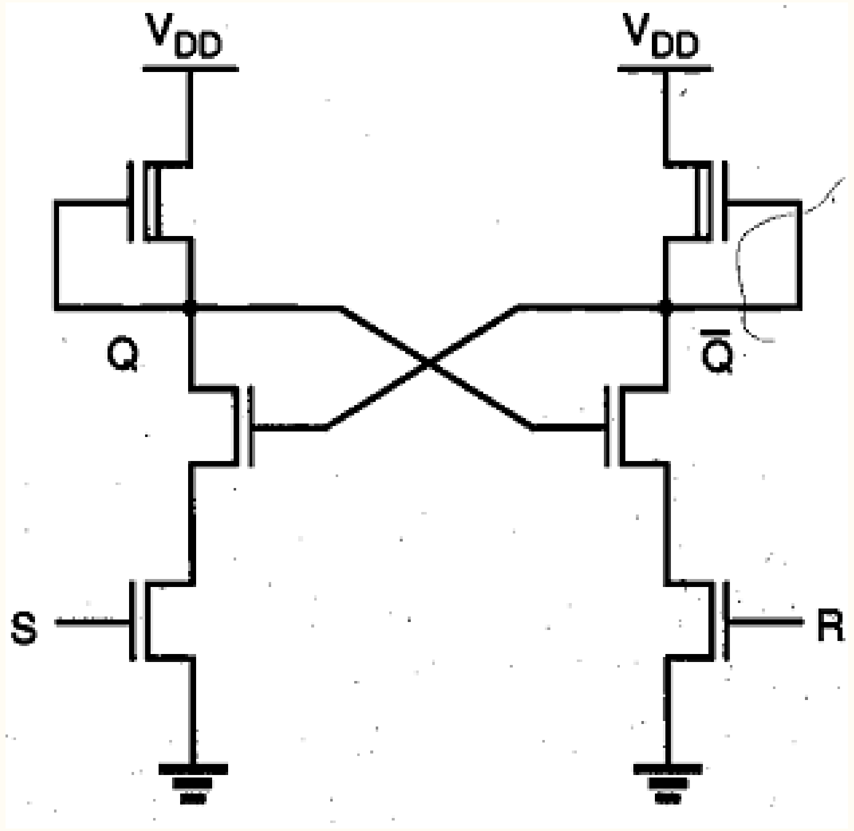


Dr. Fazal Noorbasha

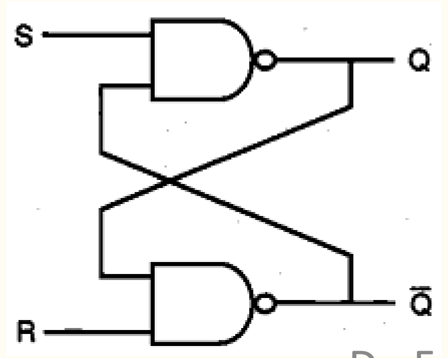
Depletion-load nMOS SR latch circuit based on NOR2 gates



CMOS SR latch circuit based on NAND2 gates.



Depletion-load nMOS NAND-based SR latch circuit

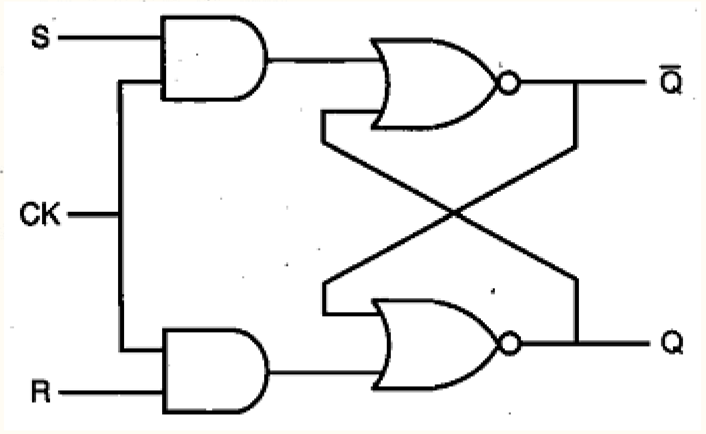


S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Operation
0	0	1	1	not allowed
0	1	1	0	set
1	0	0	1	reset
1	1	Q_n	$\overline{Q_n}$	hold

Dr. Fazal Noorbasha

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA

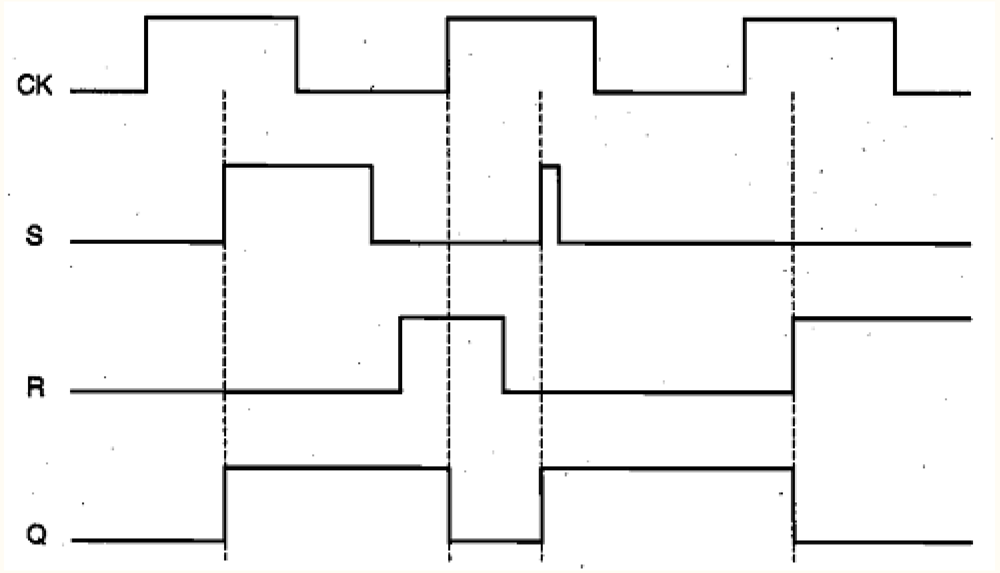
Clocked Latch and Flip-Flop Circuits



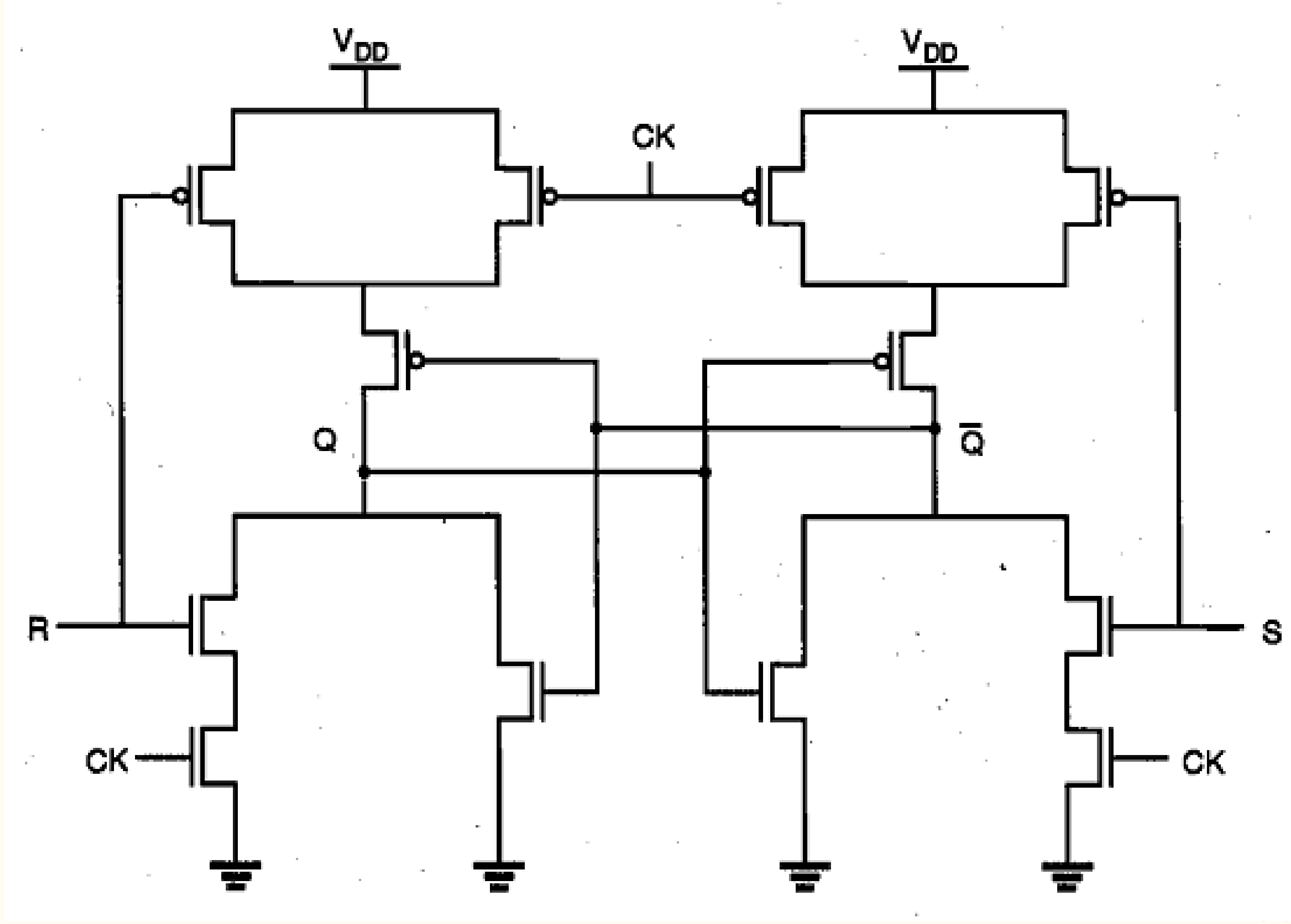
$\overline{[\,(CK \cdot S) + Q\,]}$

$\overline{[\,(CK \cdot R) + Q'\,]}$

Gate-level schematic of the clocked NOR-based SR latch

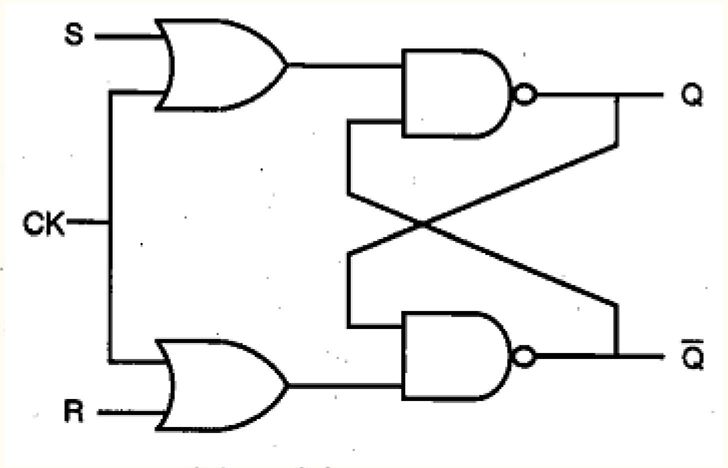


Sample input and output waveforms



AOI-based implementation of the clocked NOR-based SR latch circuit.

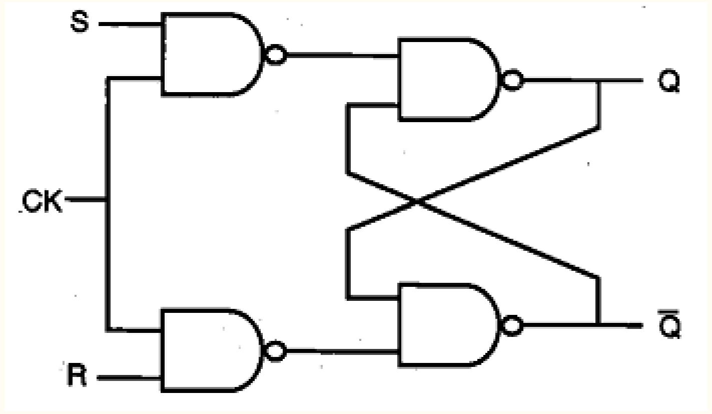
Dr. Fazal Noorbasha



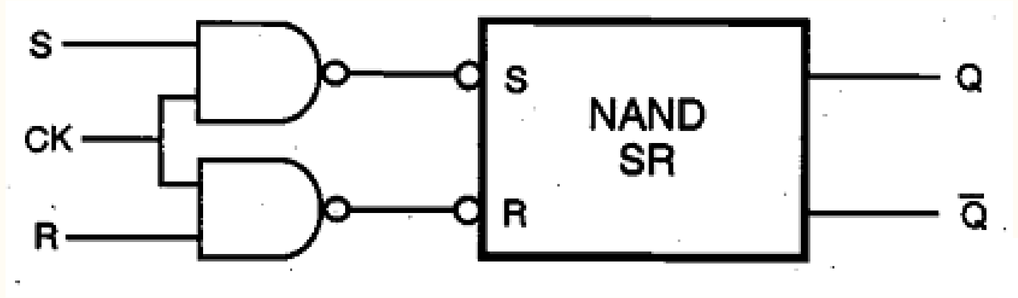
$\overline{[(CK + S) \cdot Q']}$

$\overline{[(CK + R) \cdot Q]}$

Gate-level schematic of the clocked NAND-based SR latch circuit, with active low inputs



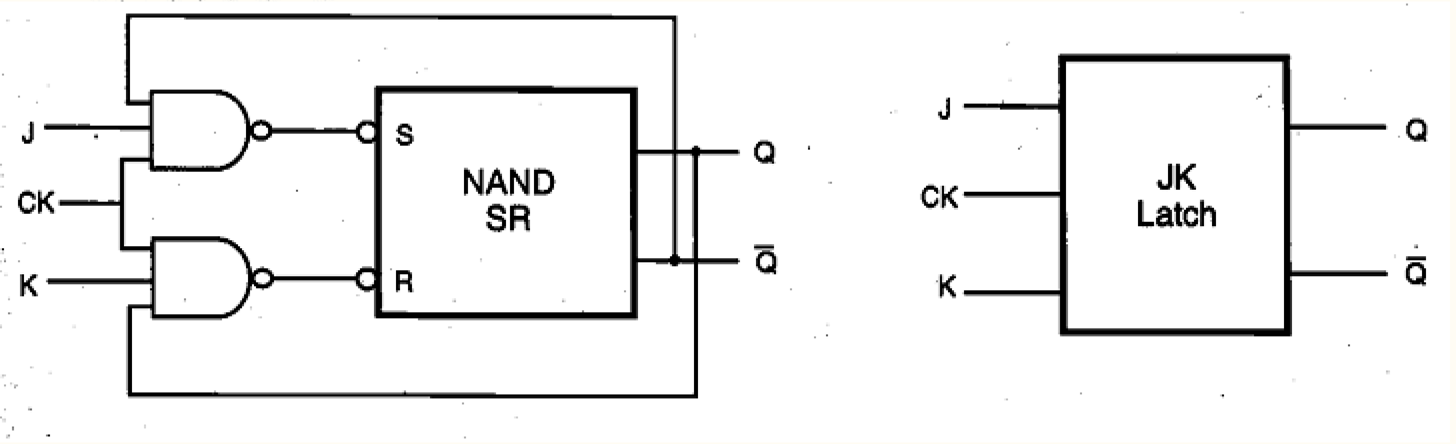
Gate-level schematic of the clocked NAND-based SR latch circuit, with active high inputs



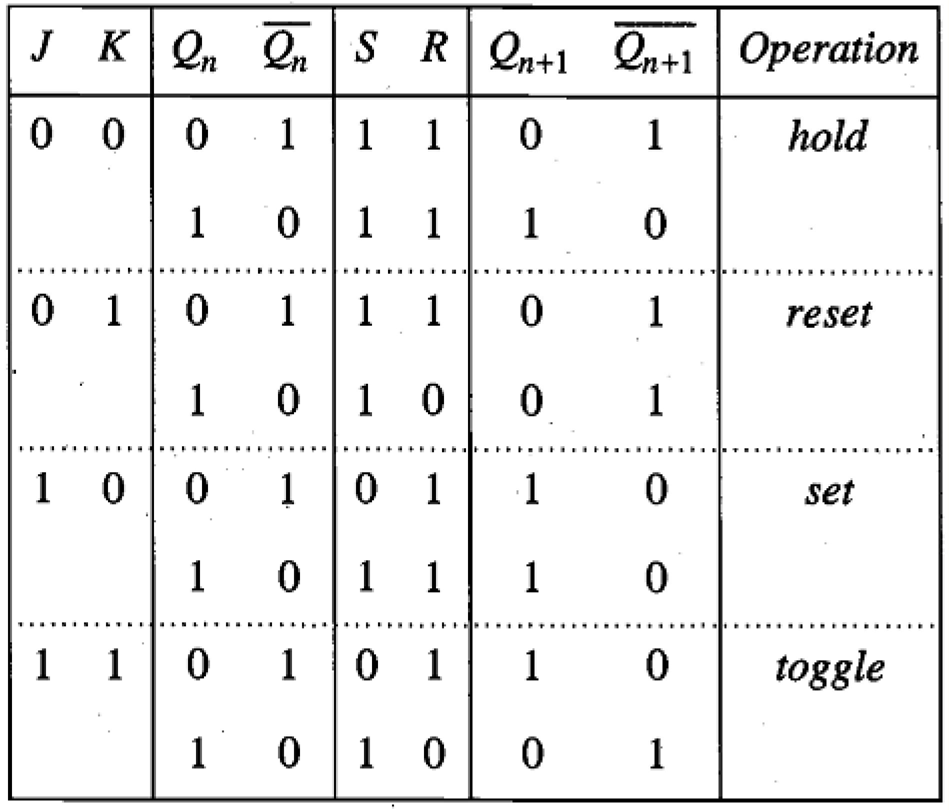
Partial block diagram representation

Dr. Fazal Noorbasha

Clocked JK Latch

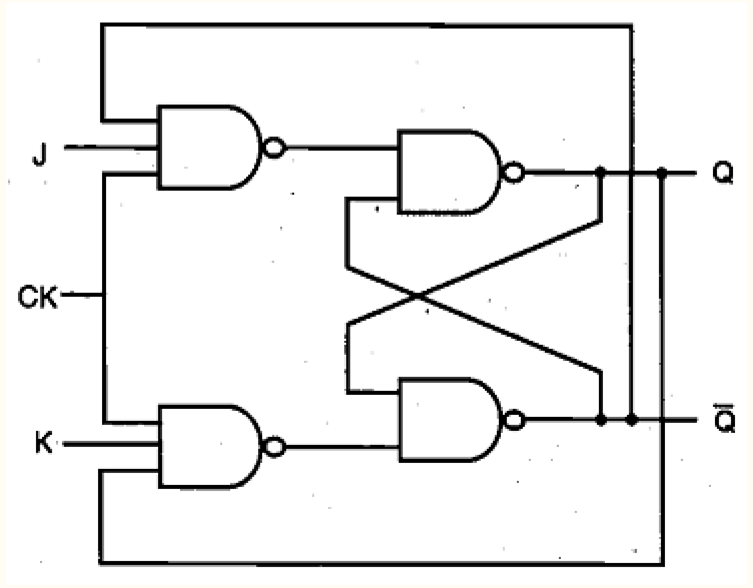


Gate-level schematic of the clocked NAND-based JK latch circuit.



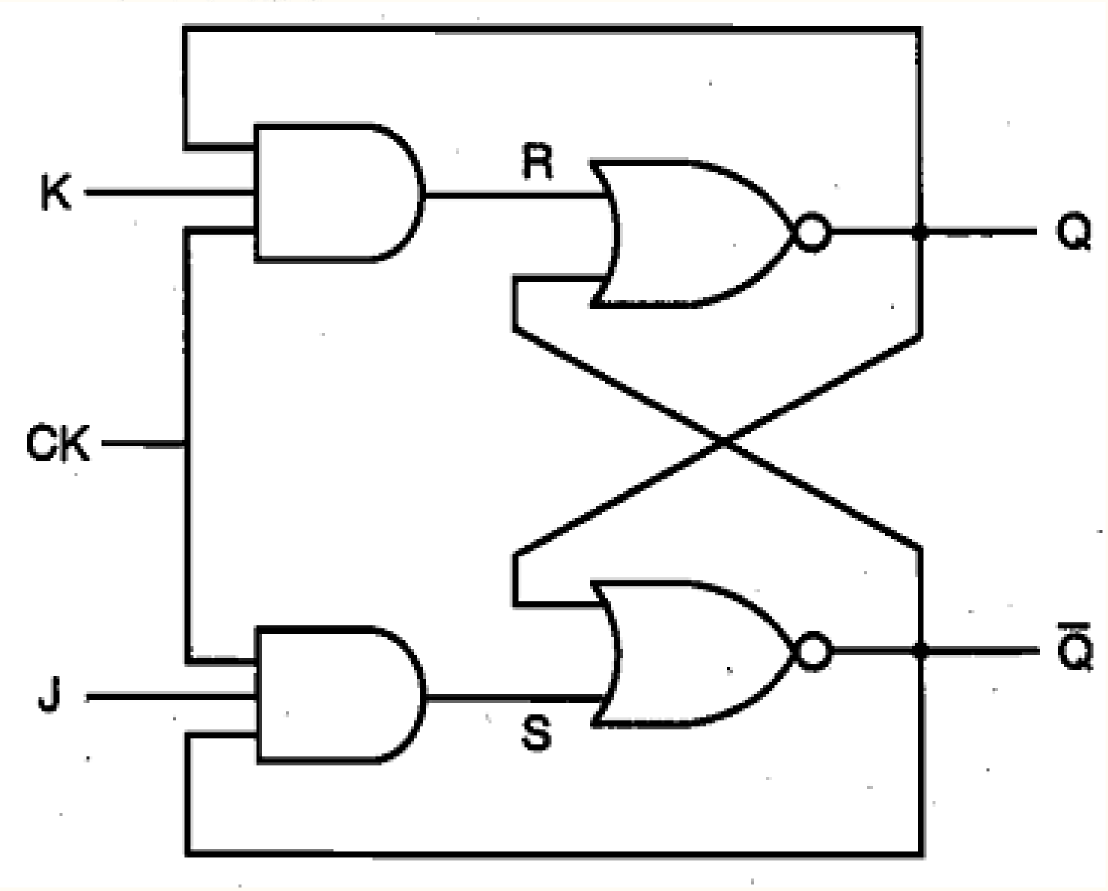
J	K	Q_n	$\overline{Q_n}$	S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Operation
0	0	0	1	1	1	0	1	*hold*
		1	0	1	1	1	0	
0	1	0	1	1	1	0	1	*reset*
		1	0	1	0	0	1	
1	0	0	1	0	1	1	0	*set*
		1	0	1	1	1	0	
1	1	0	1	0	1	1	0	*toggle*
		1	0	1	0	0	1	

Detailed truth table of the JK latch circuit.

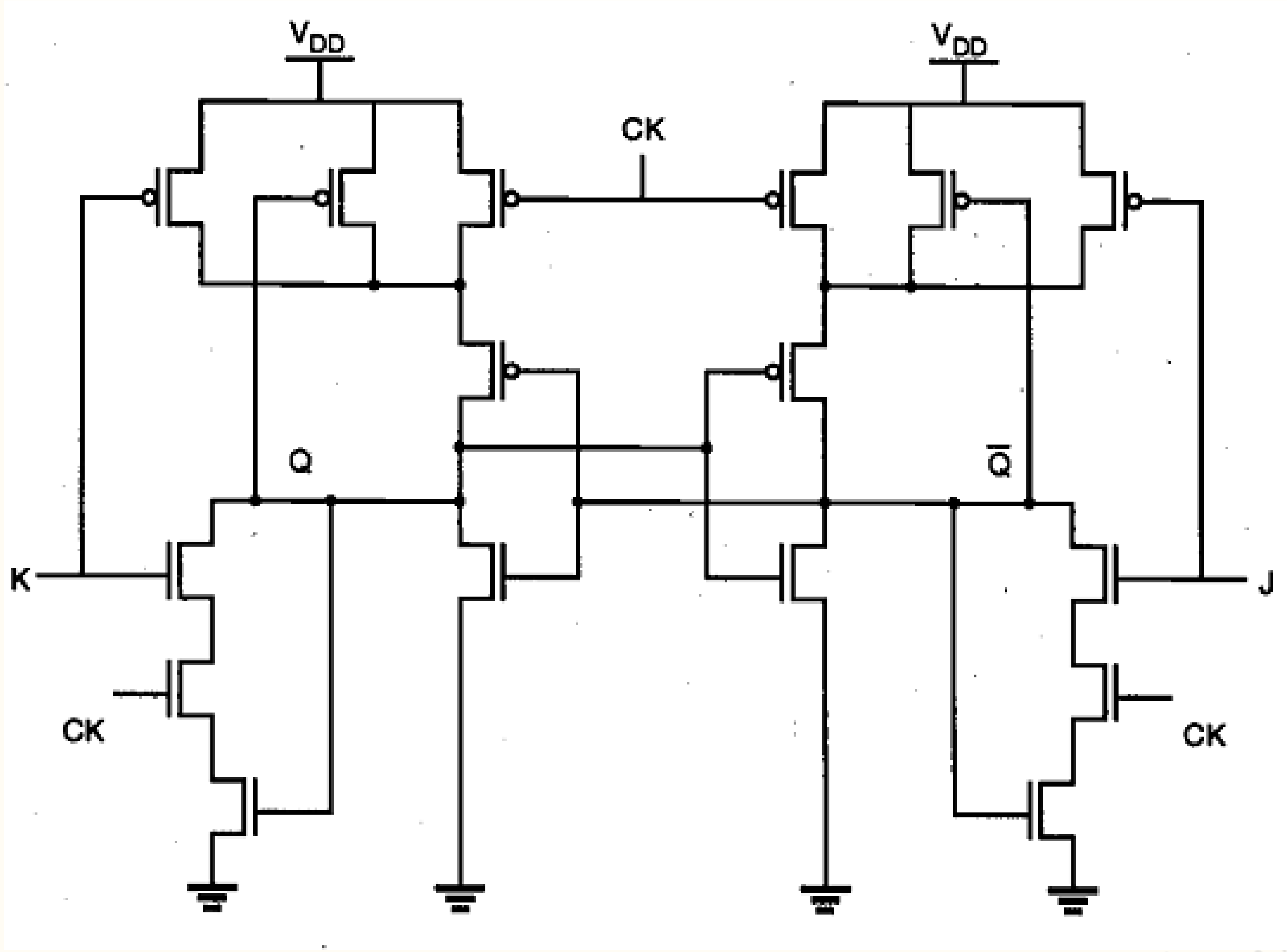


All-NAND implementation of the clocked JK latch circuit.

$\overline{[\,(CK \cdot K \cdot Q) + Q'\,]}$



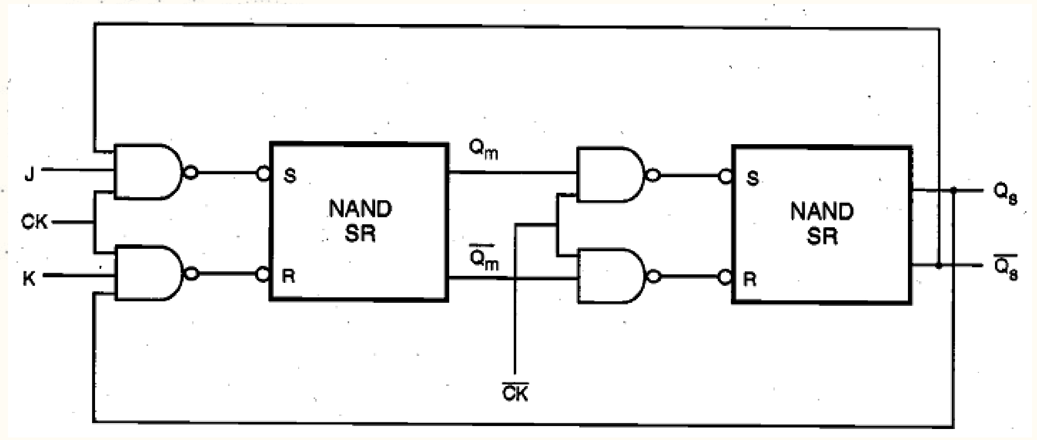
$\overline{[\,(CK \cdot J \cdot Q') + Q\,]}$



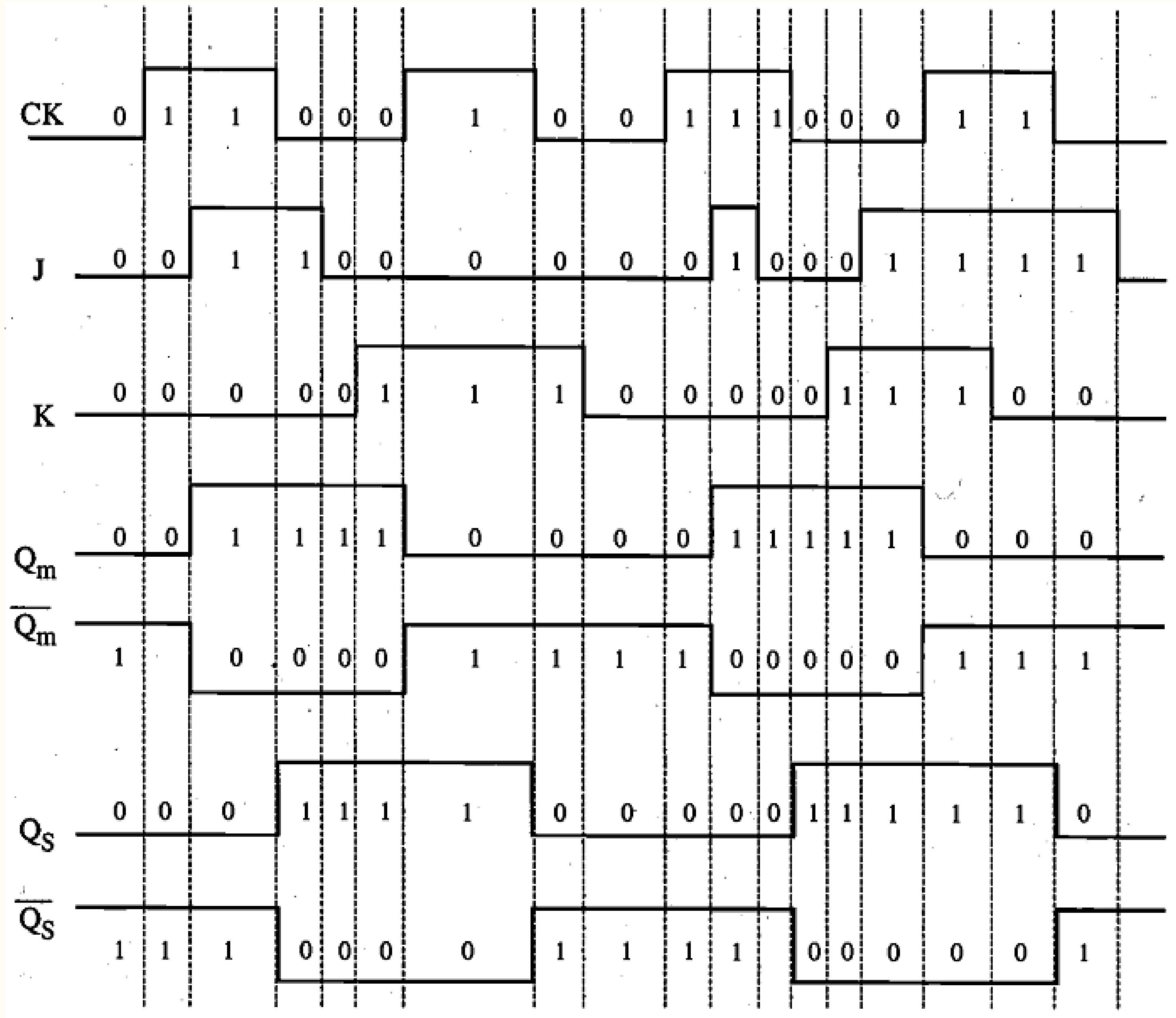
Gate-level schematic of the clocked NOR-based JK latch circuit.

CMOS AOI realization of the JK latch

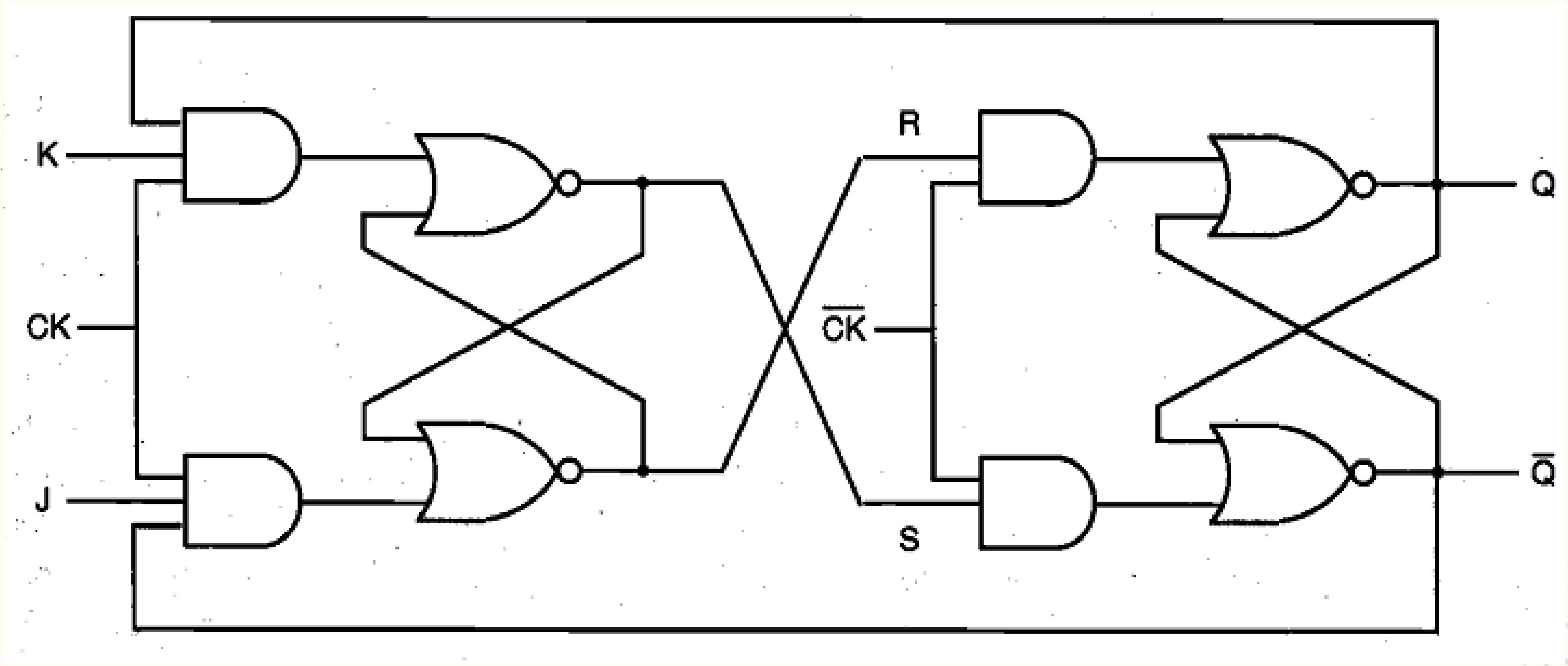
Master-Slave Flip-Flop



Master-slave flip-flop consisting of NAND-based JK latches

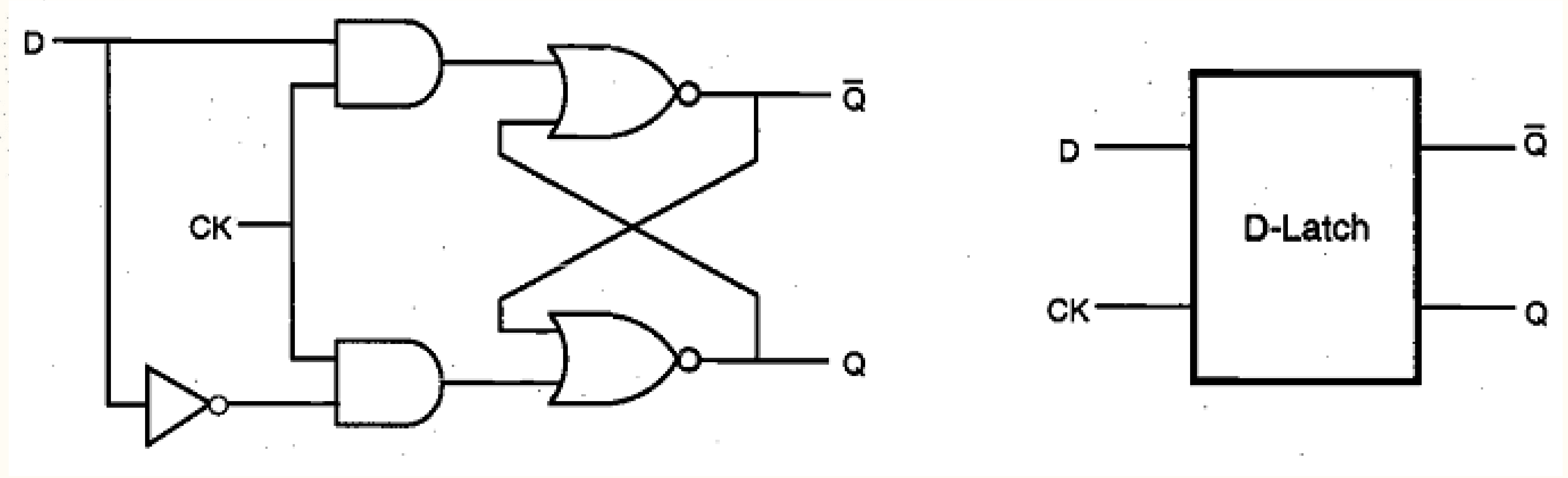


Sample input and output waveforms of the master-slave flip-flop circuit



NOR-based realization of the JK master-slave flip-flop

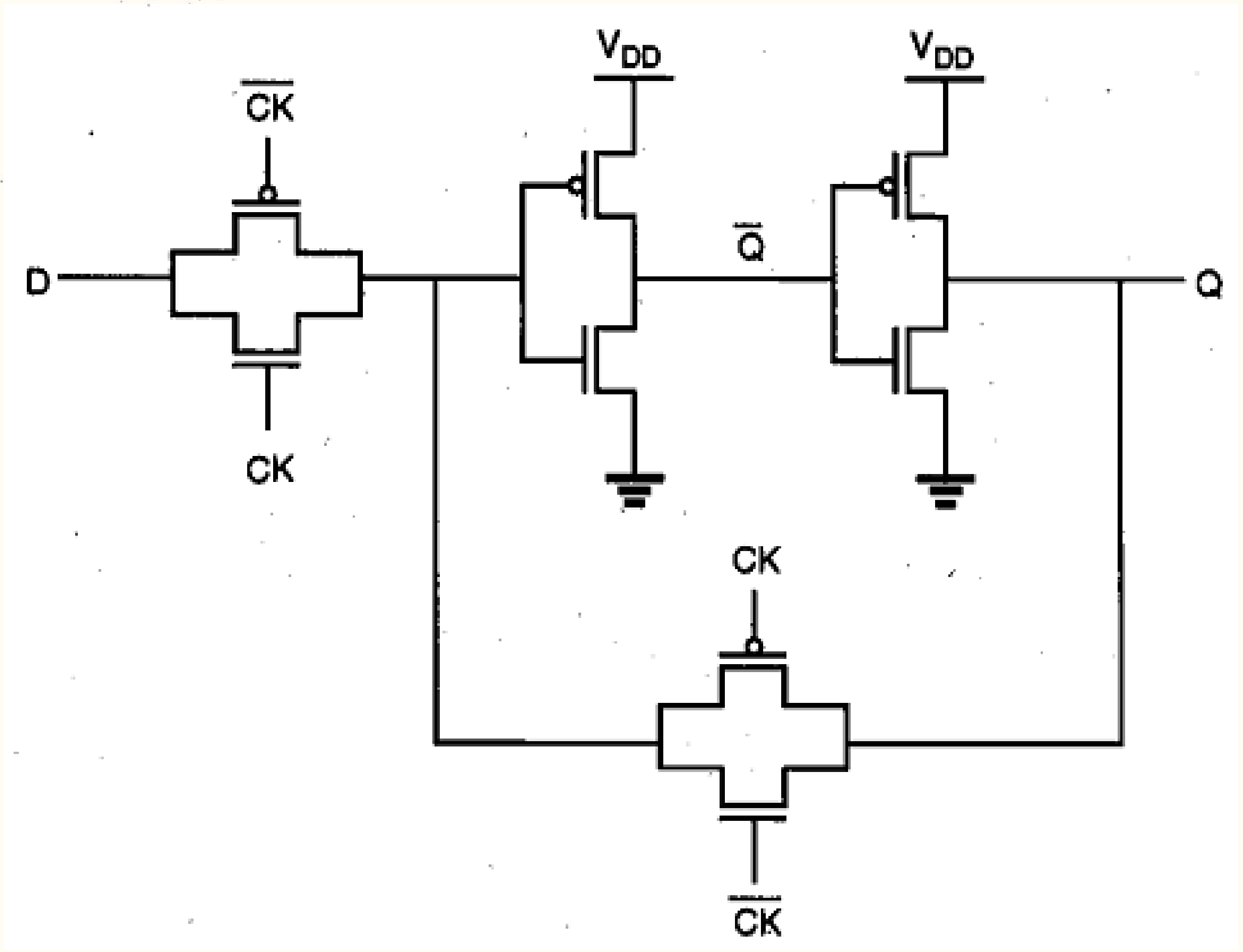
CMOS D-Latch and Edge-Triggered Flip-Flop



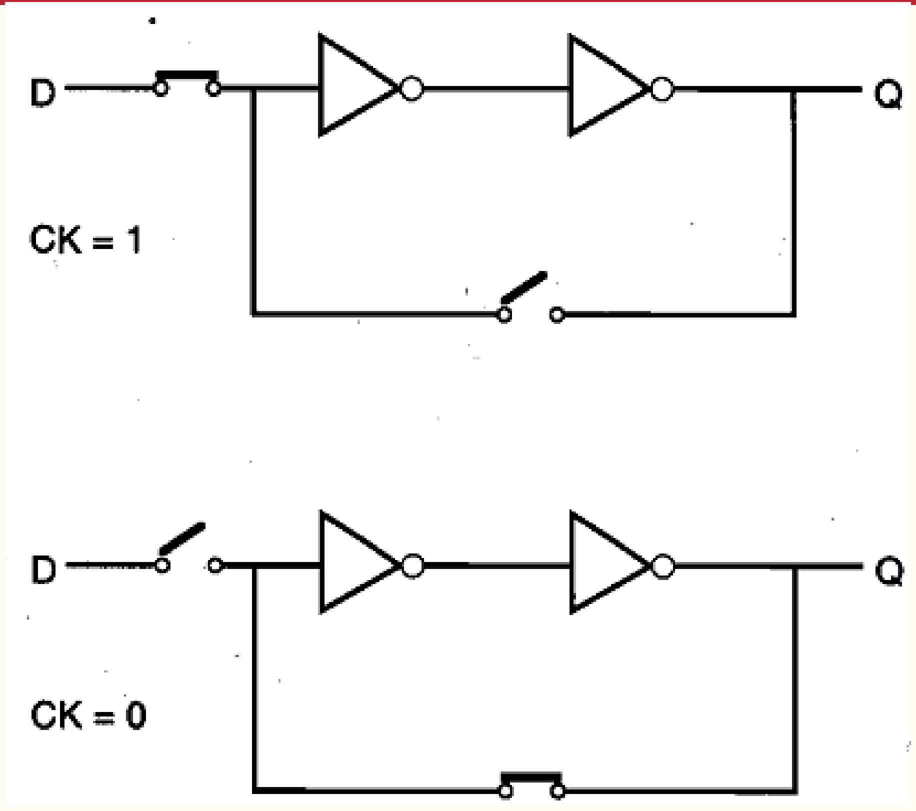
Gate-level schematic and the block diagram view of the D-latch

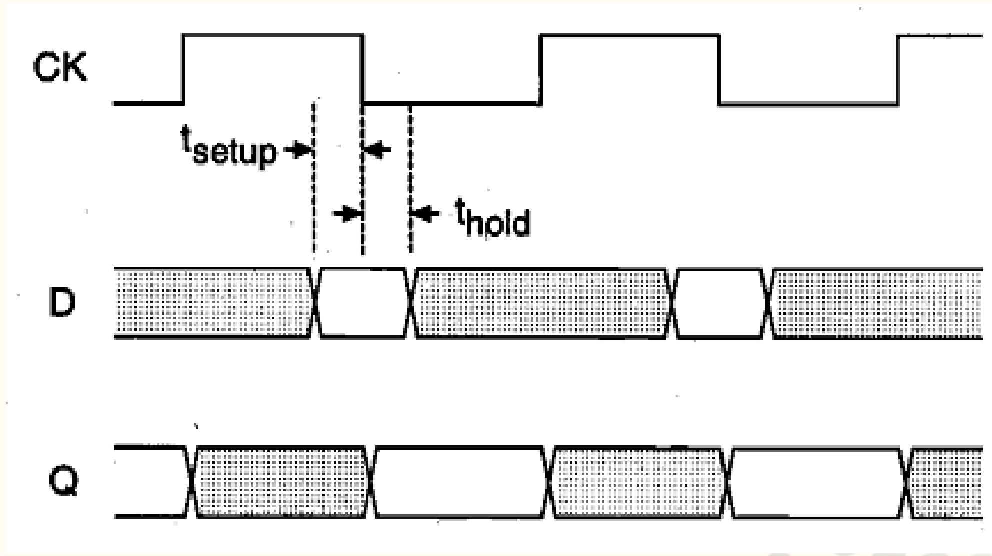
Dr. Fazal Noorbasha

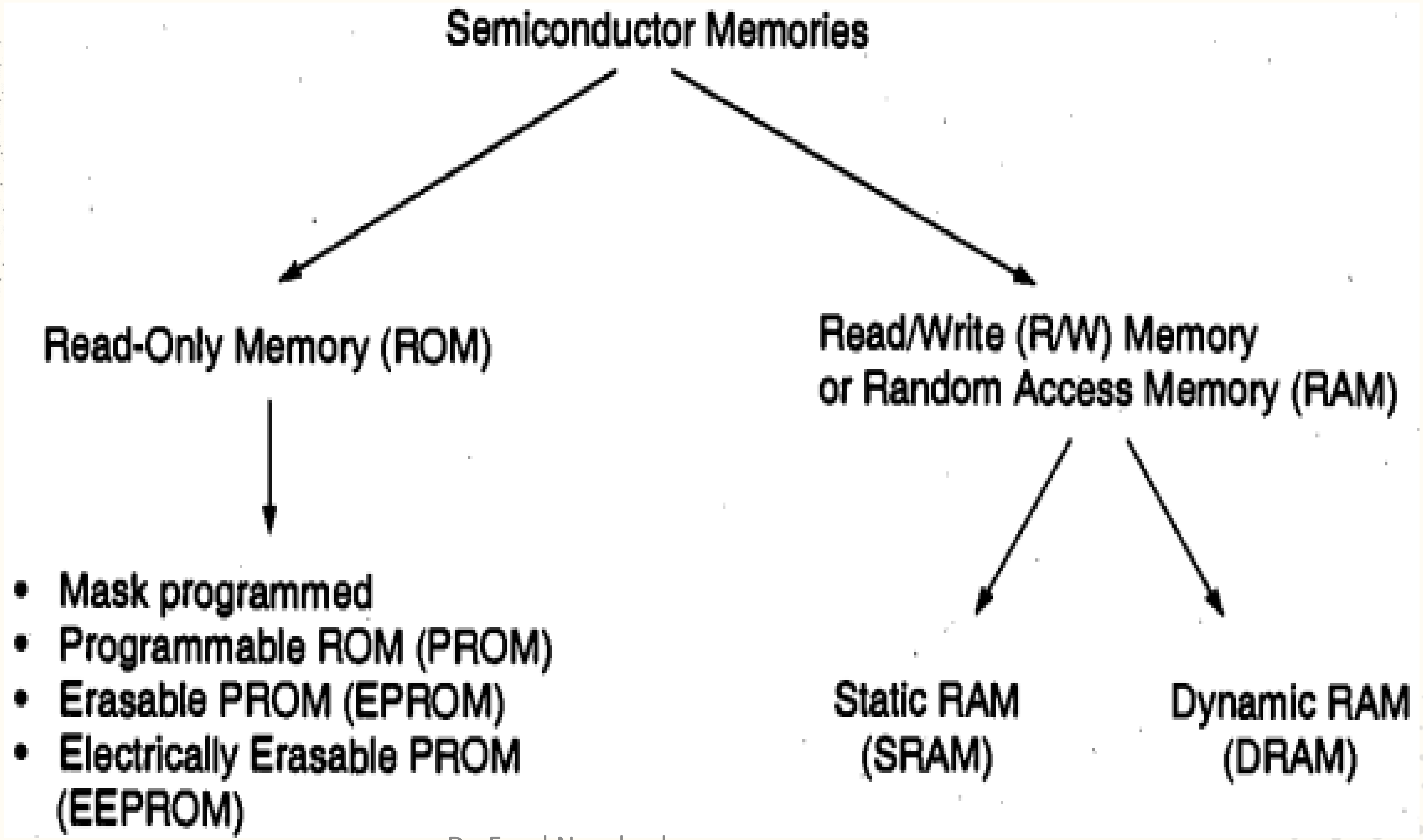
Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in



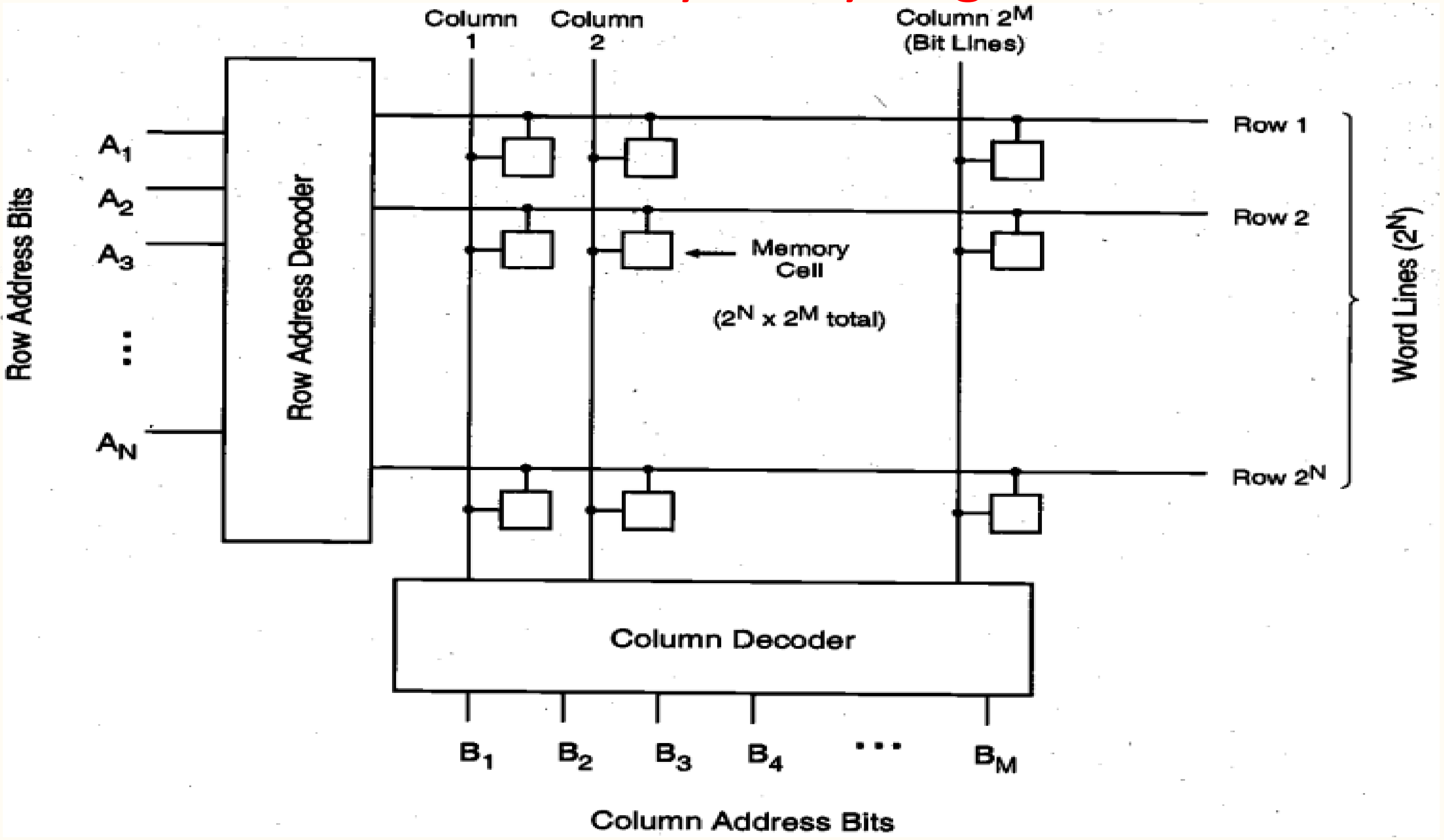
CMOS implementation of the D-latch (version 1)






Semiconductor Memories
Read-Only Memory (ROM)
Read/Write (R/W) Memory or Random Access Memory (RAM)
• Mask programmed
• Programmable ROM (PROM)
• Erasable PROM (EPROM)
• Electrically Erasable PROM (EEPROM)
Static RAM (SRAM)
Dynamic RAM (DRAM)

Random Access Memory Array Organization



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 1-bit of binary information. Also, 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. In this structure, there are 2 rows, also called word lines, and 2M columns, also called bit lines. Thus, the total number of memory cells in this array is 2M x 2N.

SRAM-READ and WRITE OPERATION

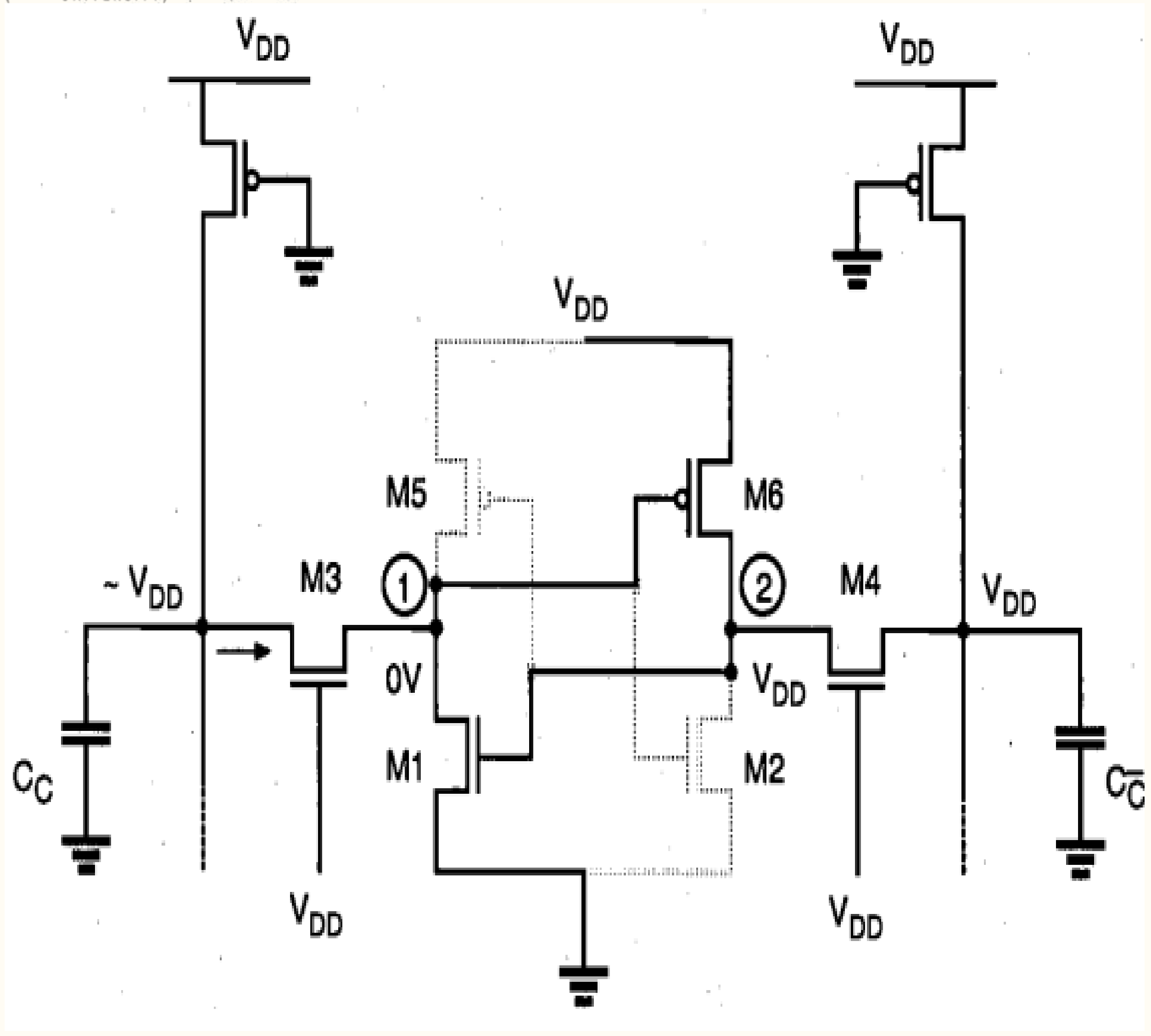


Fig (a): Voltage levels in the SRAM cell at the beginning of the "read" operation.

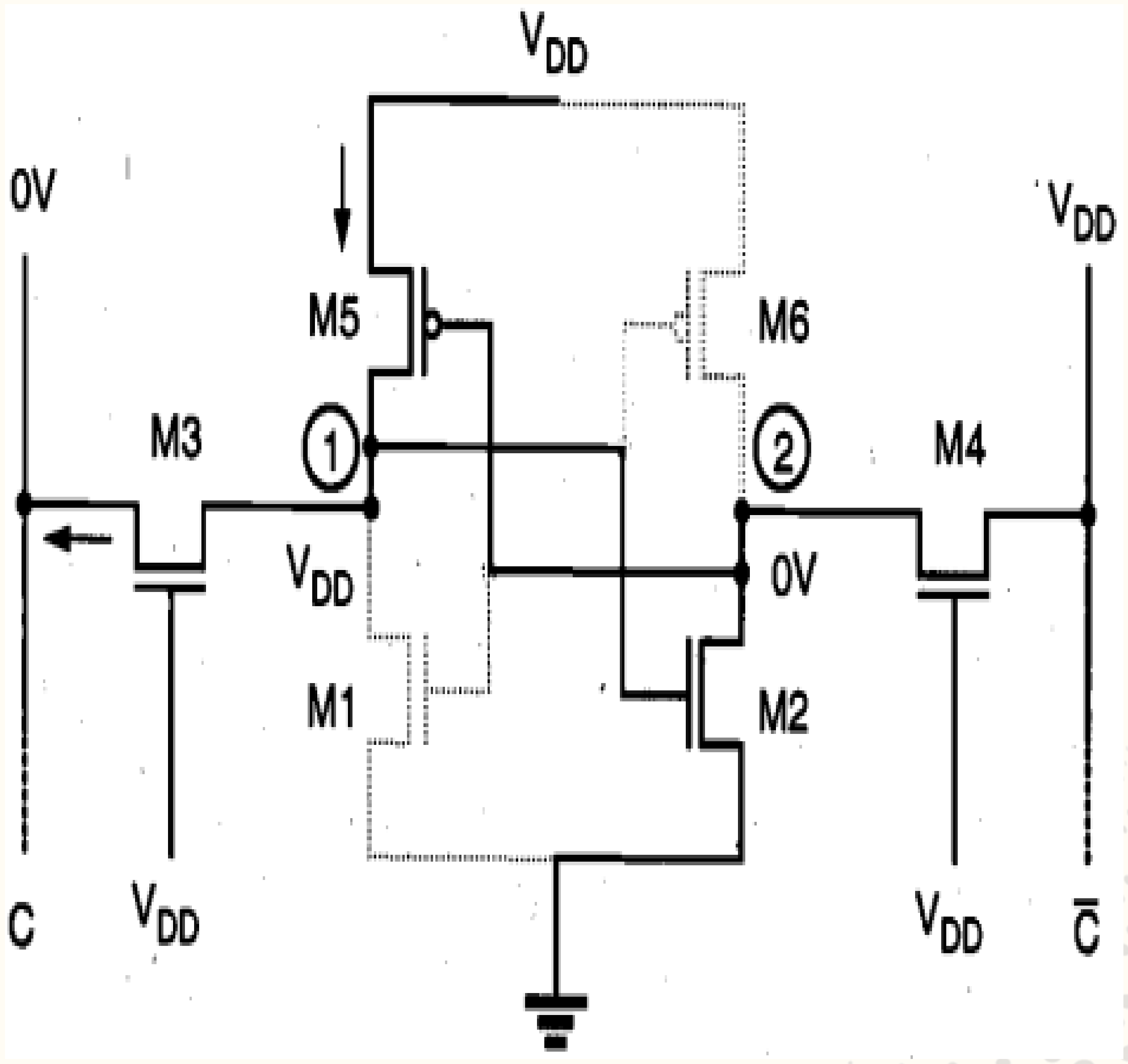


Fig (b): Voltage levels in the SRAM cell at the beginning of the "write" operation.

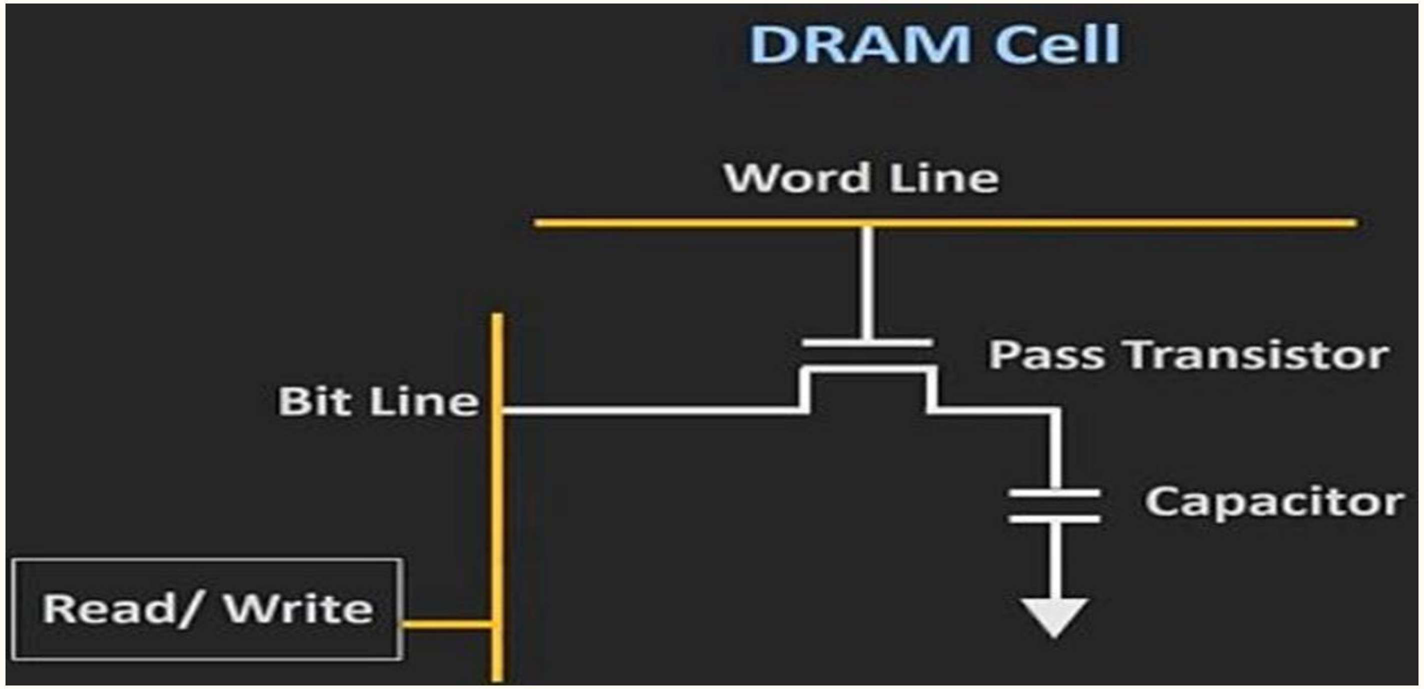
Dynamic Read-Write Memory (DRAM) Circuits

- Write Operation

1. WL = 1
2. Access transistors turned ON.
3. Apply voltage (Logic 1 = Vdd and Logic 0 =GND) to Bit Line.
4. Accordingly, capacitor will be charged to Vdd or discharged to GND.

- Hold Operation

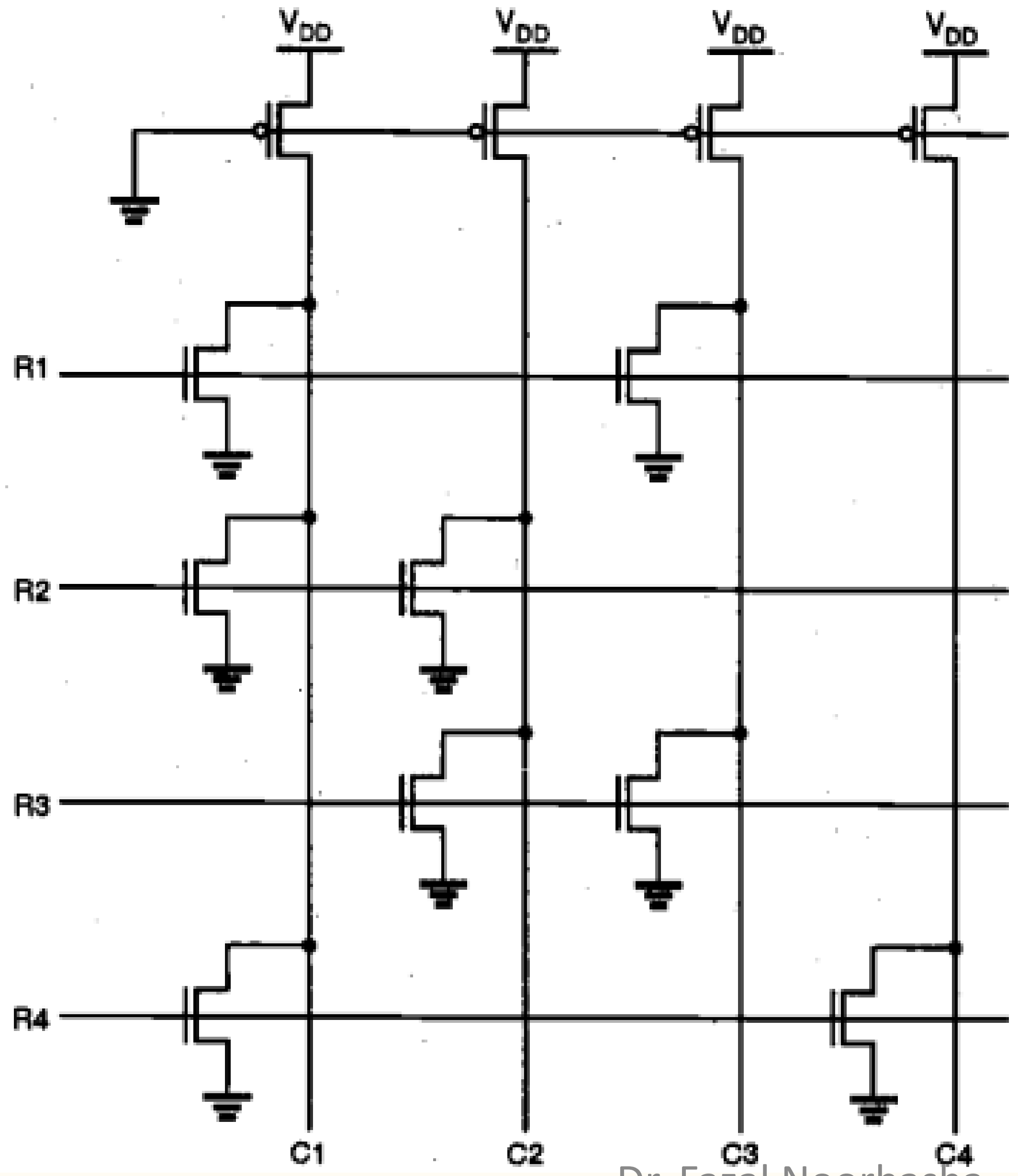
1. WL = 0
2. Access transistors turned OFF.
3. Charge is held on a capacitor.



- Read Operation

1. WL = 1
2. Access transistors turned ON.
3. Charge of the capacitor would get distributed with bit line capacitance.
4. This will change the bit line voltage as 1 or 0

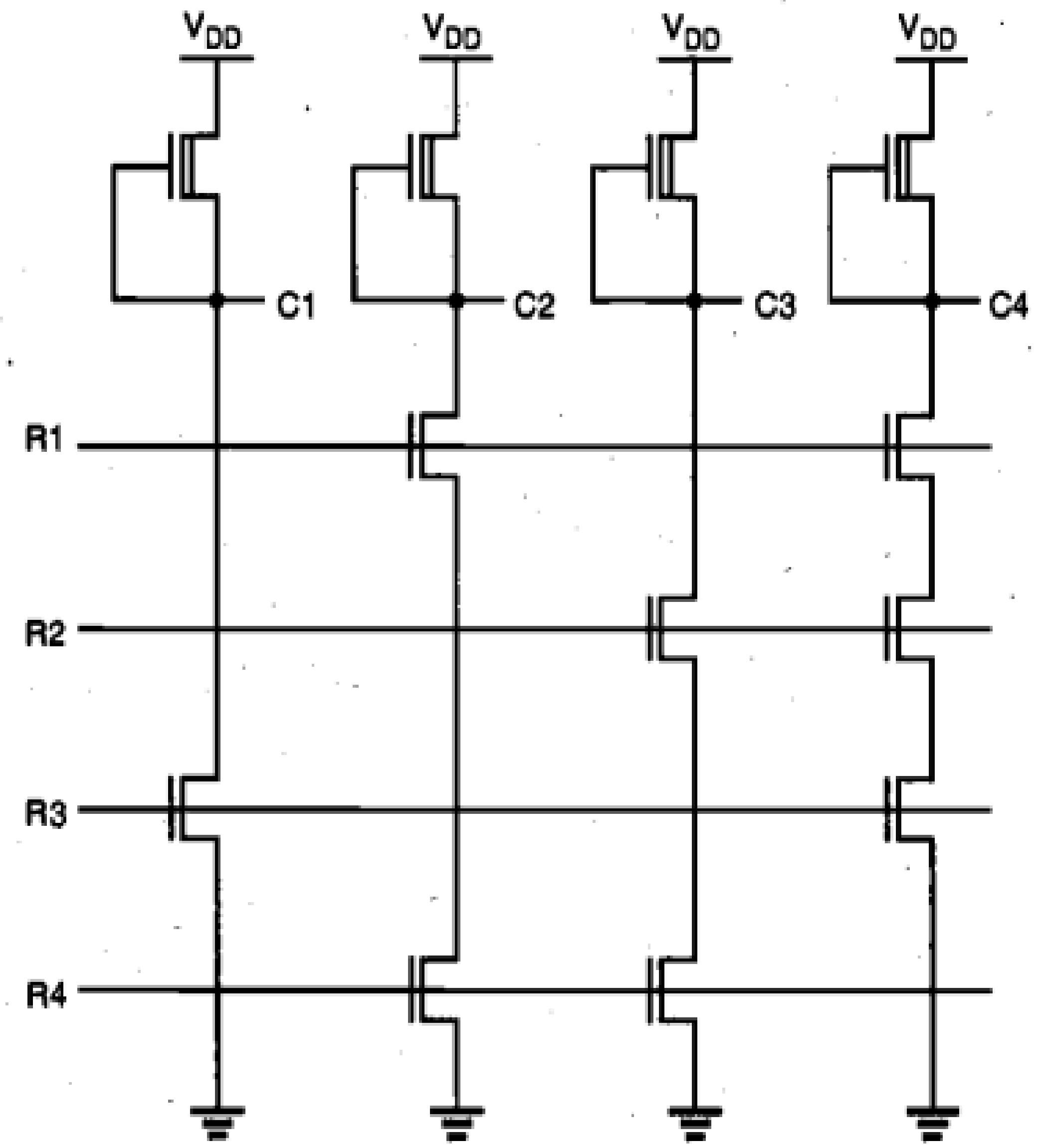
A 4-bit x 4-bit NOR-based ROM array



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

Dr. Fazal Noorbasha
Associate Professor - Dept. of ECE
Associate Dean - Academics
K L University
Guntur, AP – INDIA
E-Mail: fazalnoorbasha@kluniversity.in

A 4-bit x 4-bit NAND-based ROM 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