

L4: Sequential Building Blocks

(Flip-flops, Latches and Registers)

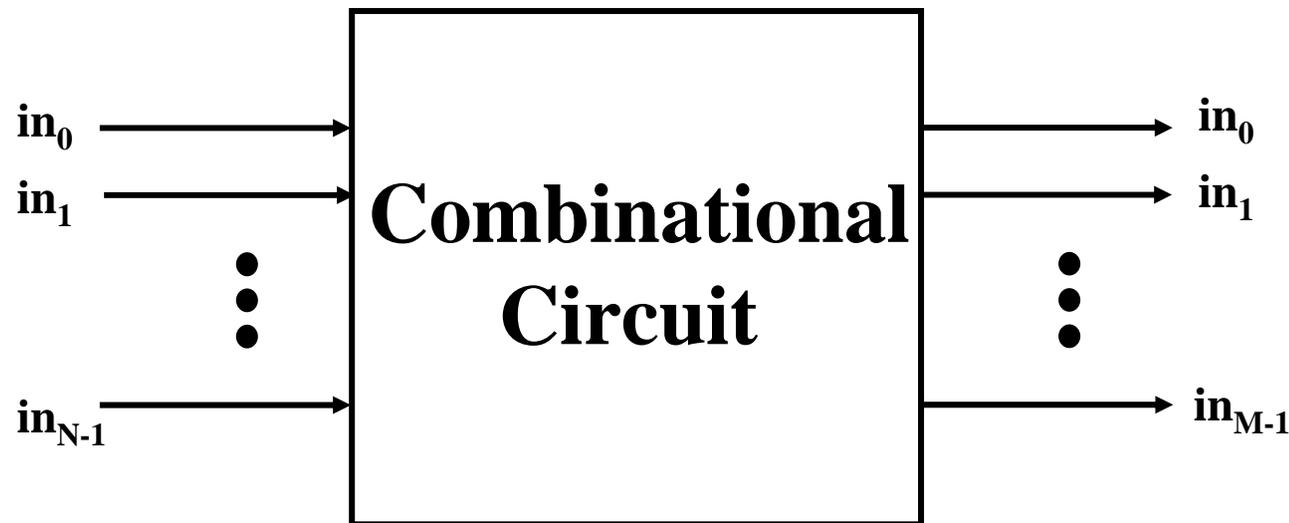


Acknowledgements:

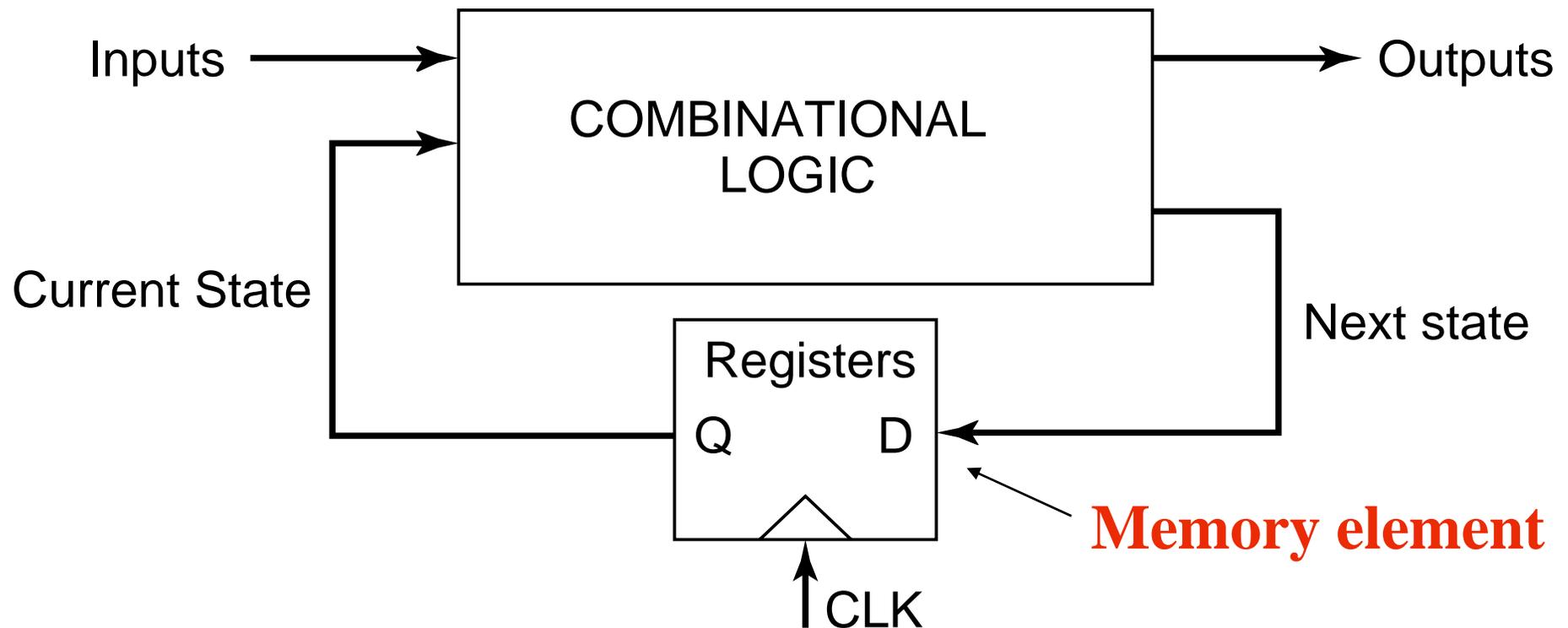
Materials in this lecture are courtesy of the following sources and are used with permission.

Prof. Randy Katz (Unified Microelectronics Corporation Distinguished Professor in Electrical Engineering and Computer Science at the University of California, Berkeley) and Prof. Gaetano Borriello (University of Washington Department of Computer Science & Engineering) from Chapter 2 of R. Katz, G. Borriello. *Contemporary Logic Design*. 2nd ed. Prentice-Hall/Pearson Education, 2005.

J. Rabaey, A. Chandrakasan, B. Nikolic. *Digital Integrated Circuits: A Design Perspective*. Prentice Hall/Pearson, 2003.

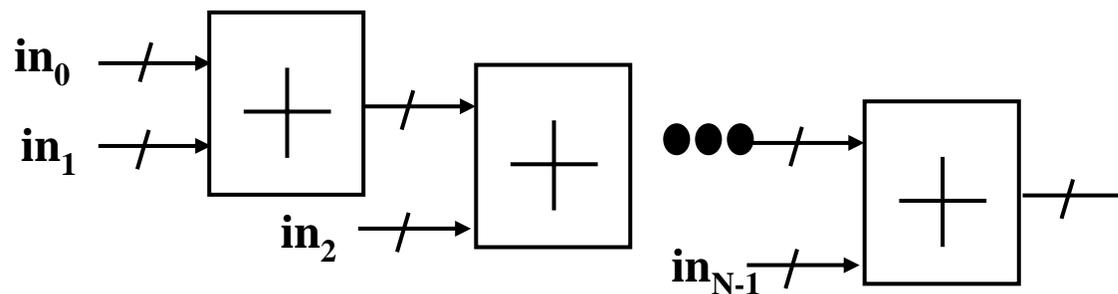


- **Combinational logic circuits are memoryless**
- **No feedback in combinational logic circuits**
- **Output assumes the function implemented by the logic network, assuming that the switching transients have settled**
- **Outputs can have multiple logical transitions before settling to the correct value**

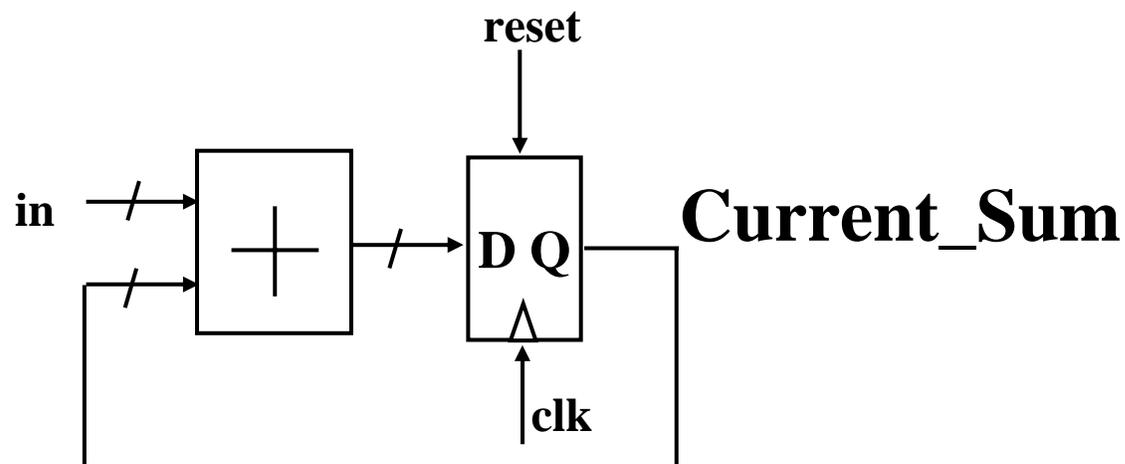


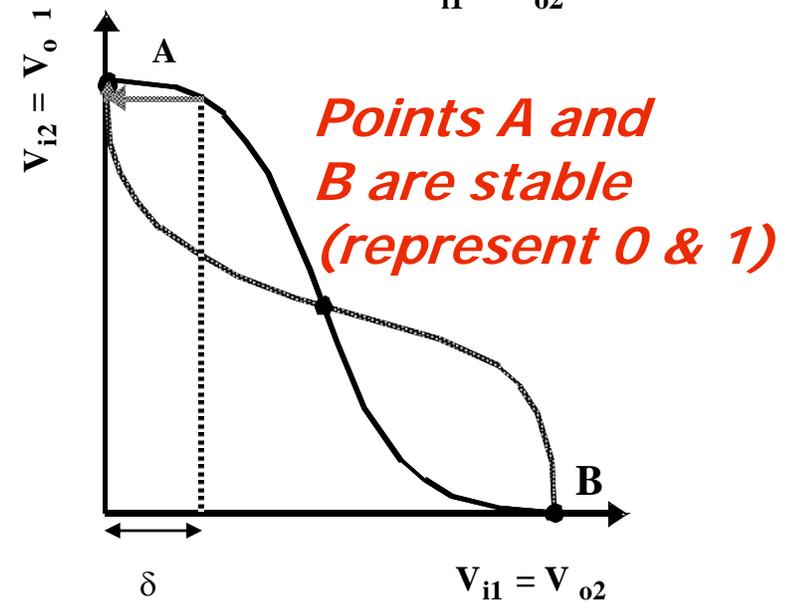
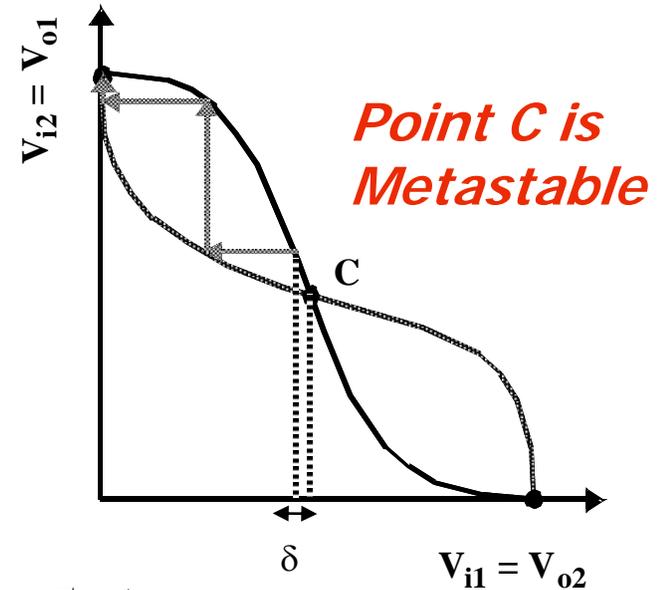
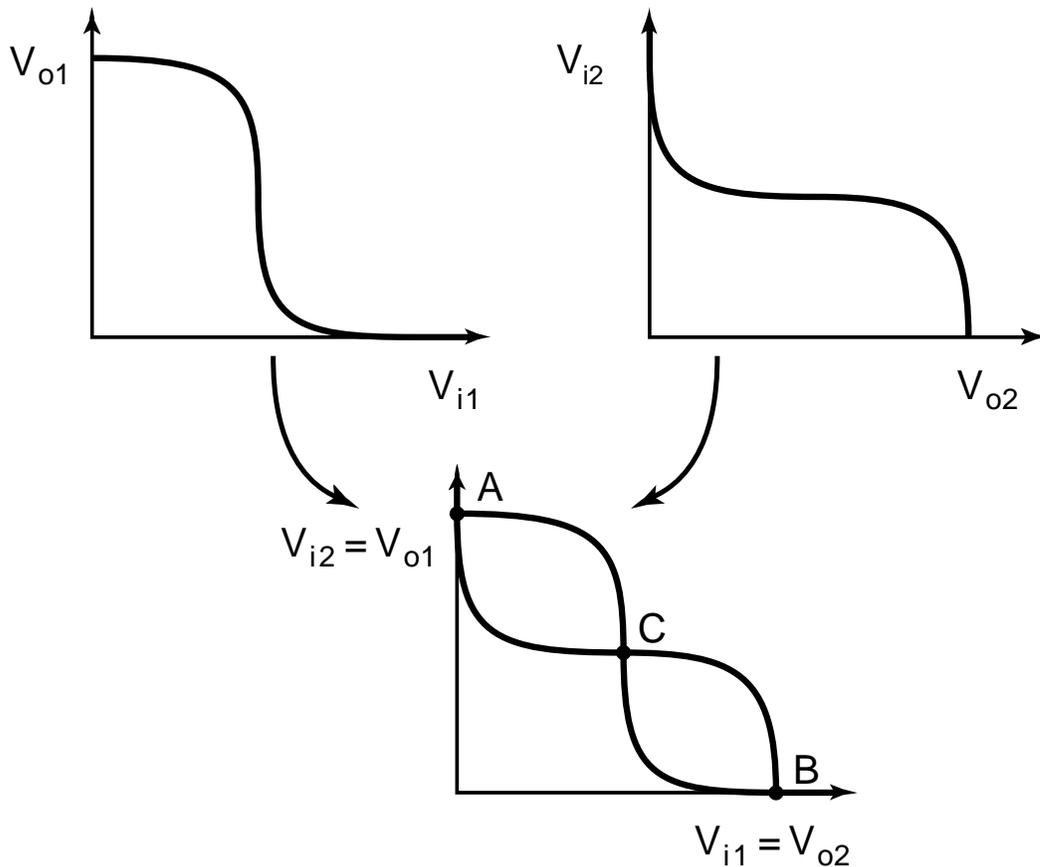
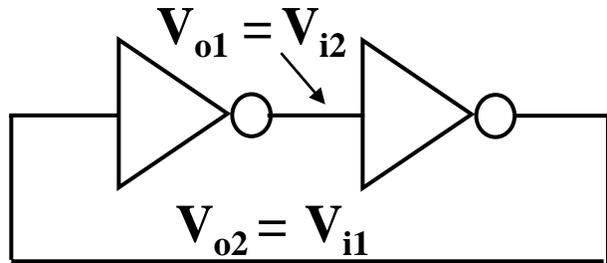
- **Sequential circuits have memory (i.e., remember the past)**
- **The current state is “held” in memory and the next state is computed based the current state and the current inputs**
- **In a synchronous systems, the **clock signal** orchestrates the sequence of events**

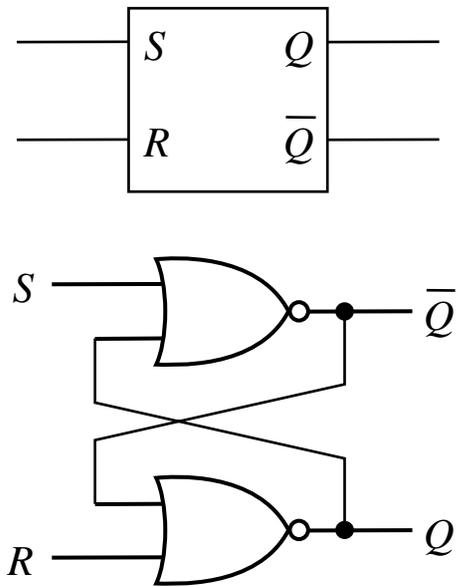
Adding N inputs (N-1 Adders)



Using a sequential (serial) approach

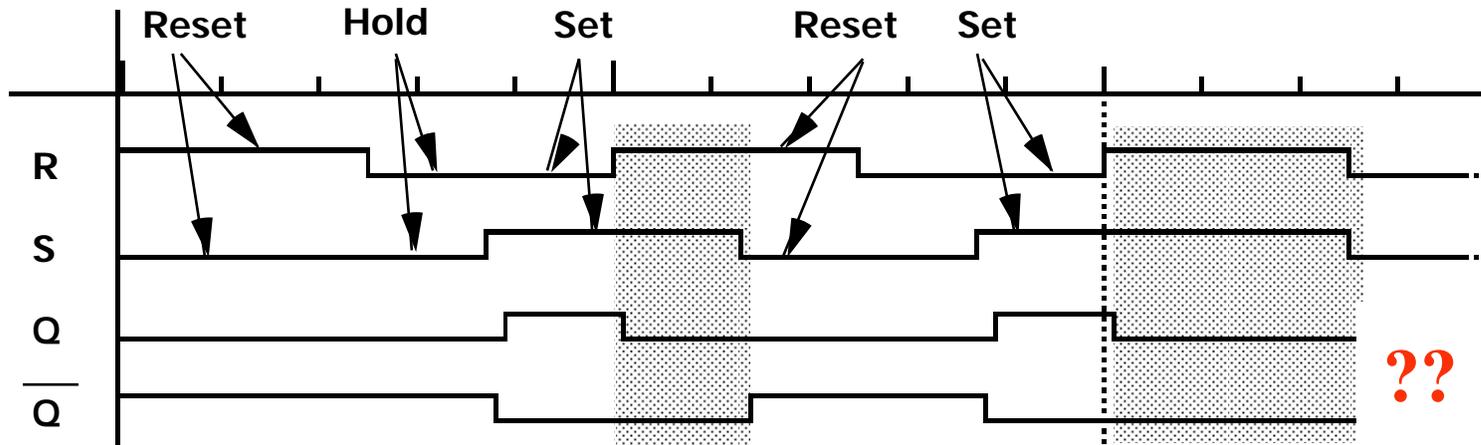
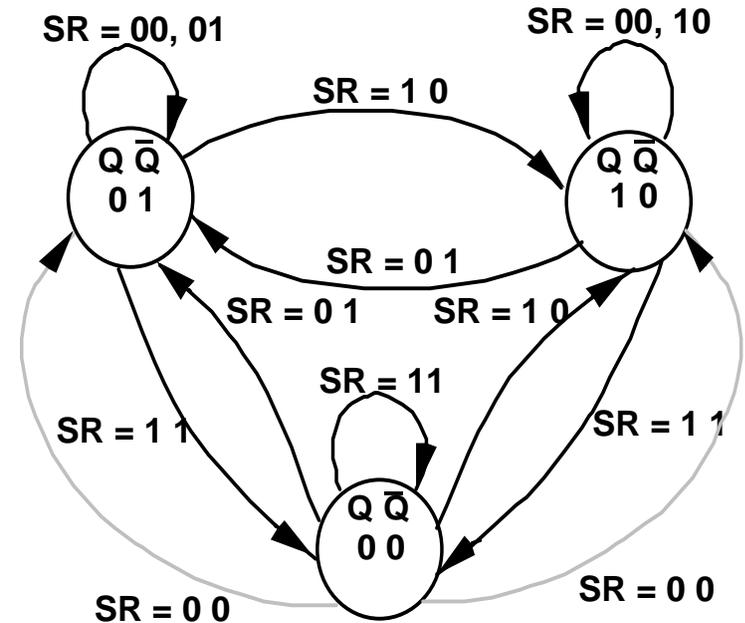






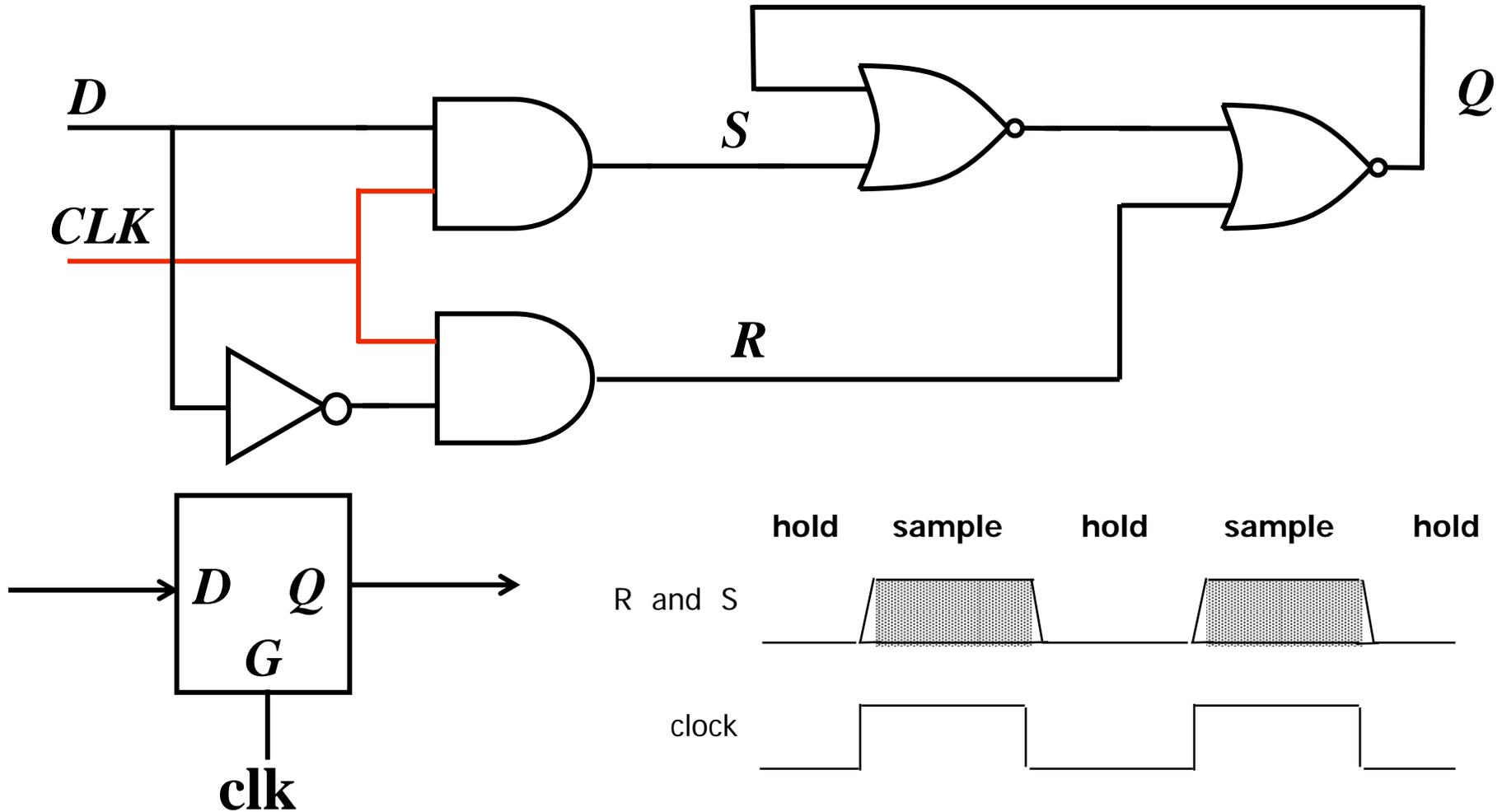
S	R	Q	\bar{Q}
0	0	Q	\bar{Q}
1	0	1	0
0	1	0	1
1	1	0	0

Forbidden State



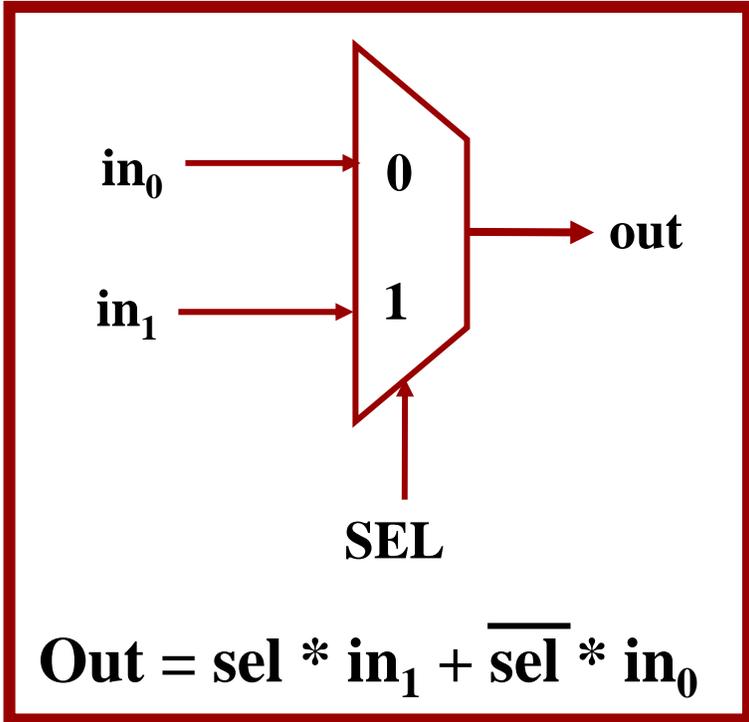
- Flip-flop refers to a bi-stable element (edge-triggered registers are also called flip-flops) – this circuit is not clocked and outputs change “asynchronously” with the inputs

Making a Clocked Memory Element: Positive D-Latch

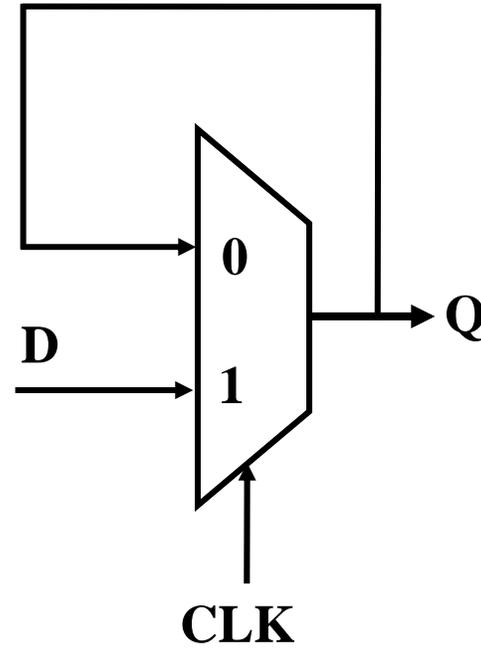


- **A Positive D-Latch:** Passes input D to output Q when CLK is high and holds state when clock is low (i.e., ignores input D)
- **A Latch is level-sensitive:** invert clock for a negative latch

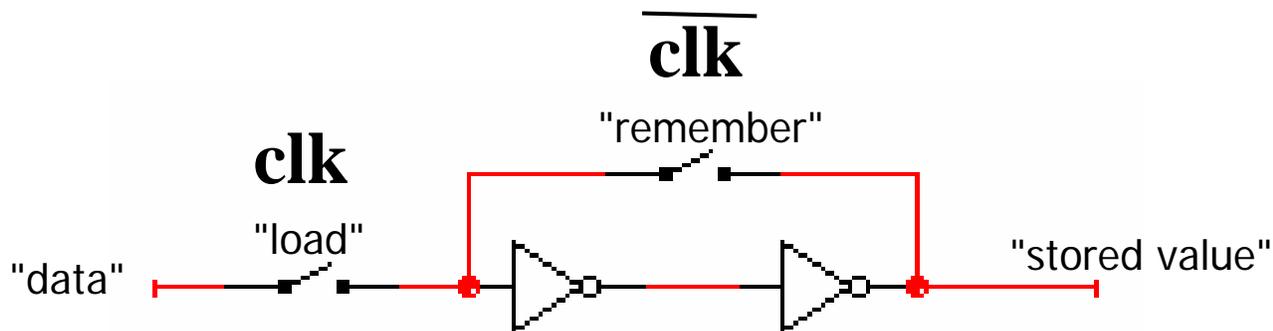
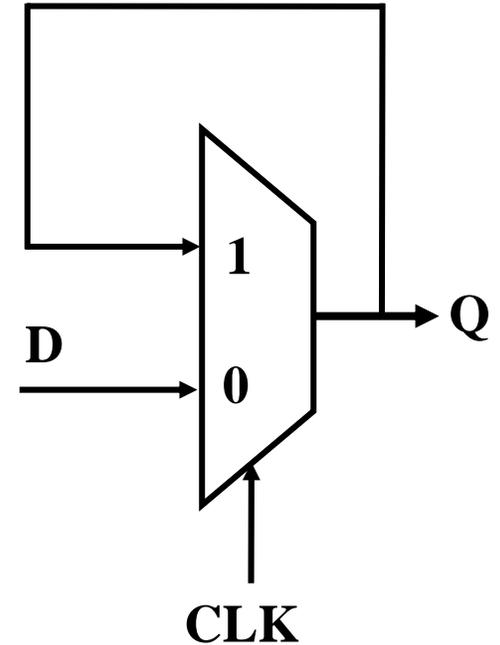
2:1 multiplexor

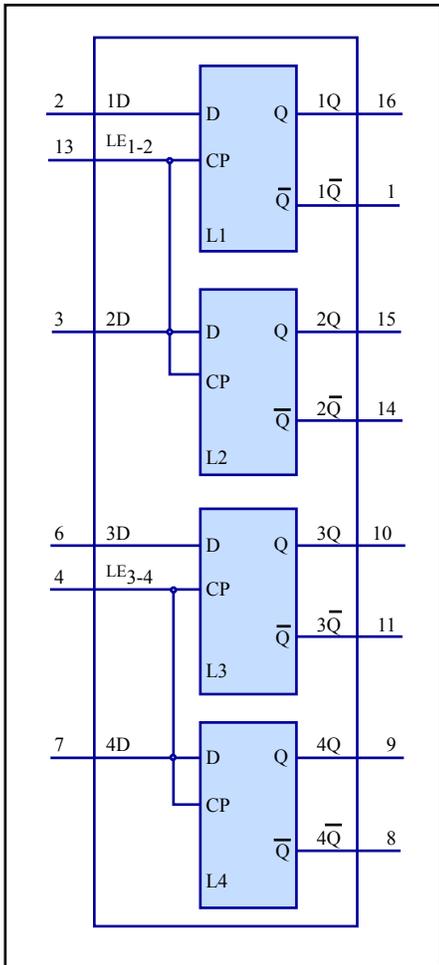


Positive Latch



Negative Latch





Operating Modes	Inputs		Outputs	
	LE _{n-n}	nD	nQ	n \bar{Q}
Data Enabled	H	L	L	H
	H	H	H	L
Data Latched	L	X	q	\bar{q}

Figures by MIT OpenCourseWare.

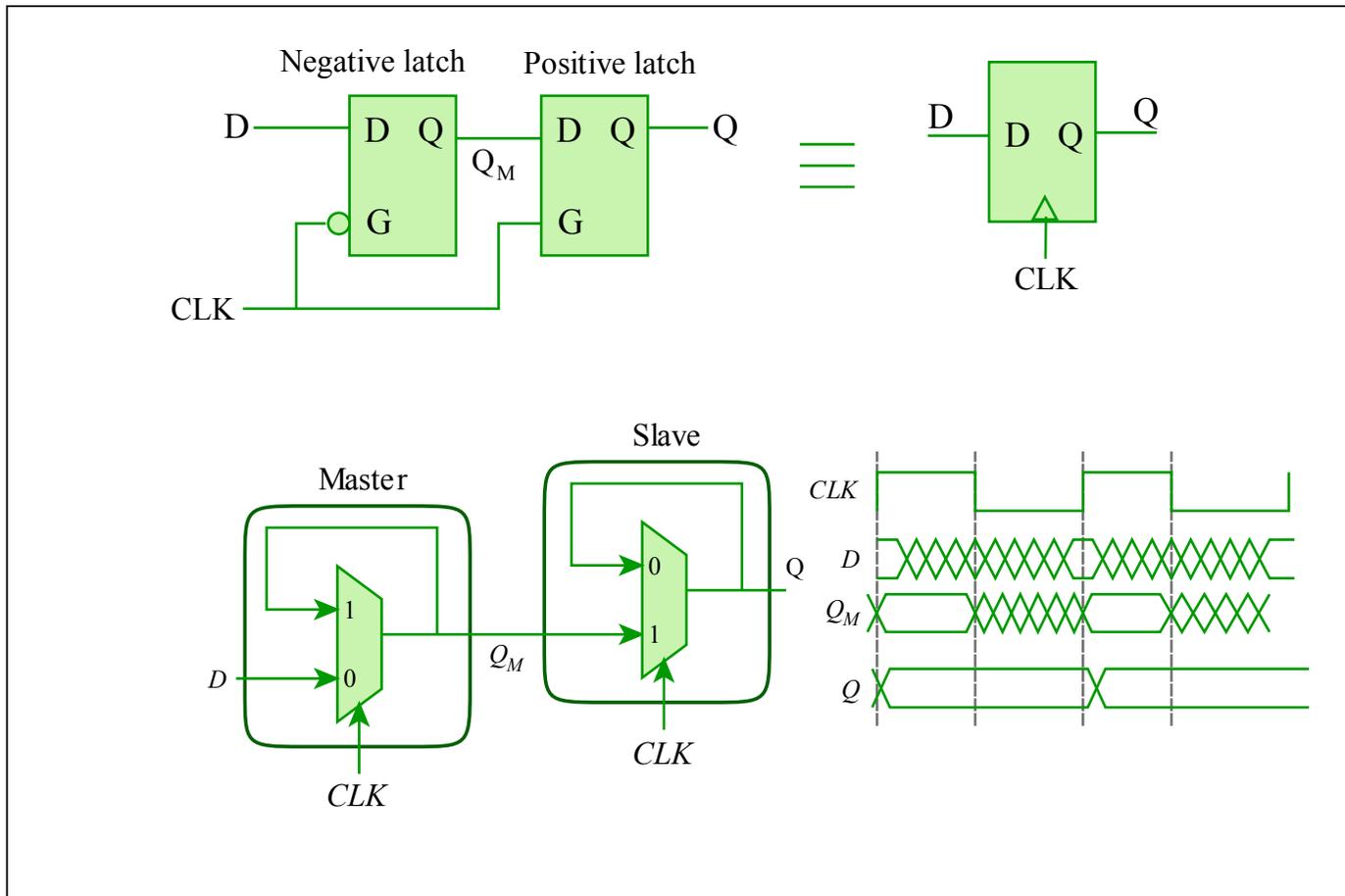


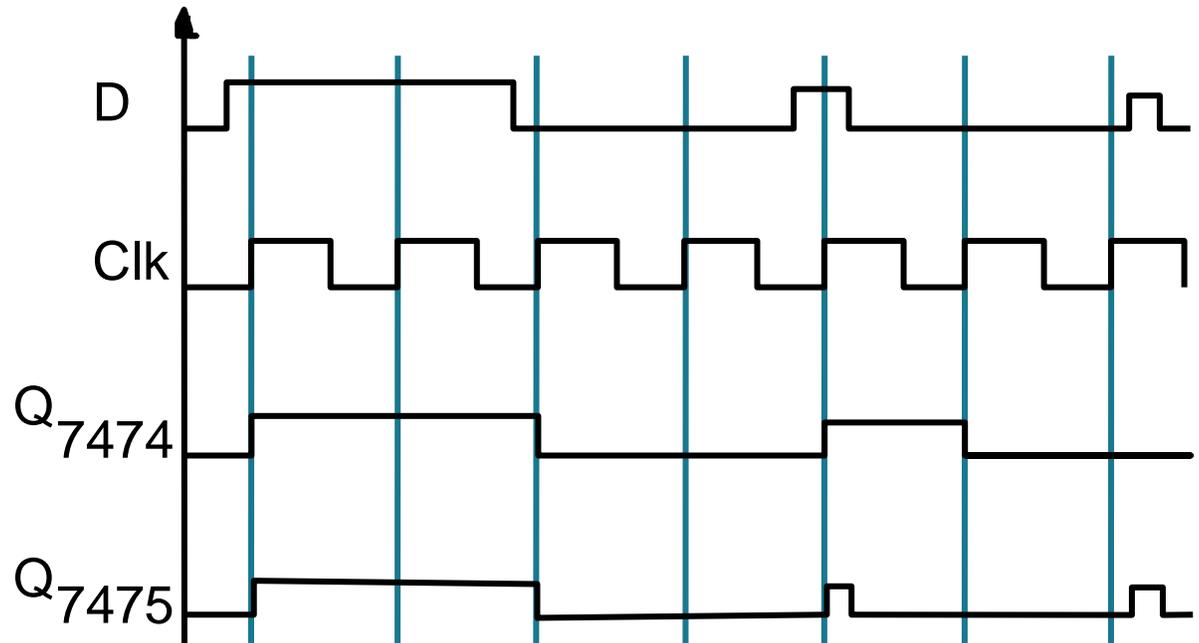
Image by MIT OpenCourseWare.

- **Master-Slave Register**
 - Use negative clock phase to latch inputs into first latch
 - Use positive clock to change outputs with second latch
- **View pair as one basic unit**
 - master-slave flip-flop twice as much logic

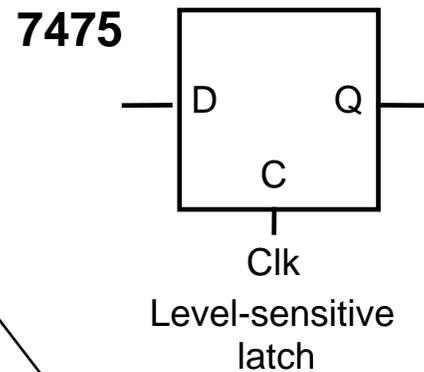
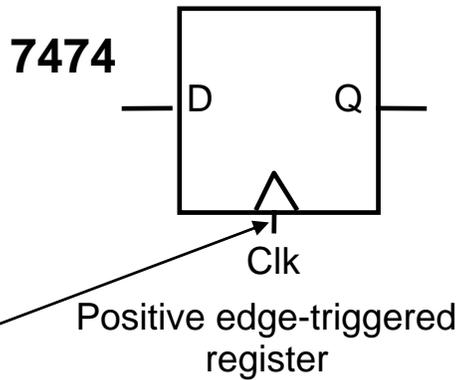
Edge triggered device sample inputs on the event edge

Transparent latches sample inputs as long as the clock is asserted

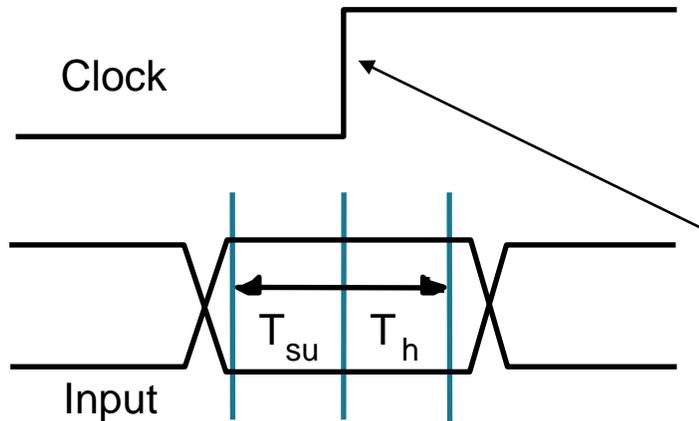
Timing Diagram:



Behavior the same unless input changes while the clock is high



Bubble here for negative edge triggered register



Clock:

Periodic Event, causes state of memory element to change

memory element can be updated on the: rising edge, falling edge, high level, low level

Setup Time (T_{su})

Minimum time before the clocking event by which the input must be stable

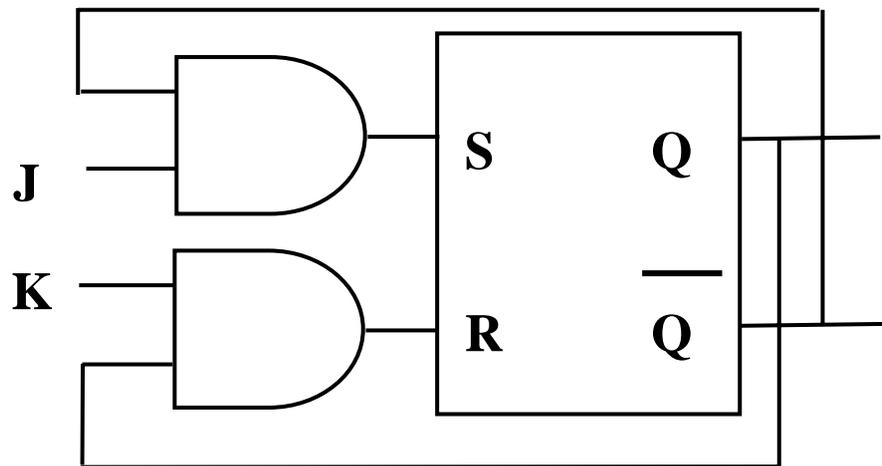
Hold Time (T_h)

Minimum time after the clocking event during which the input must remain stable

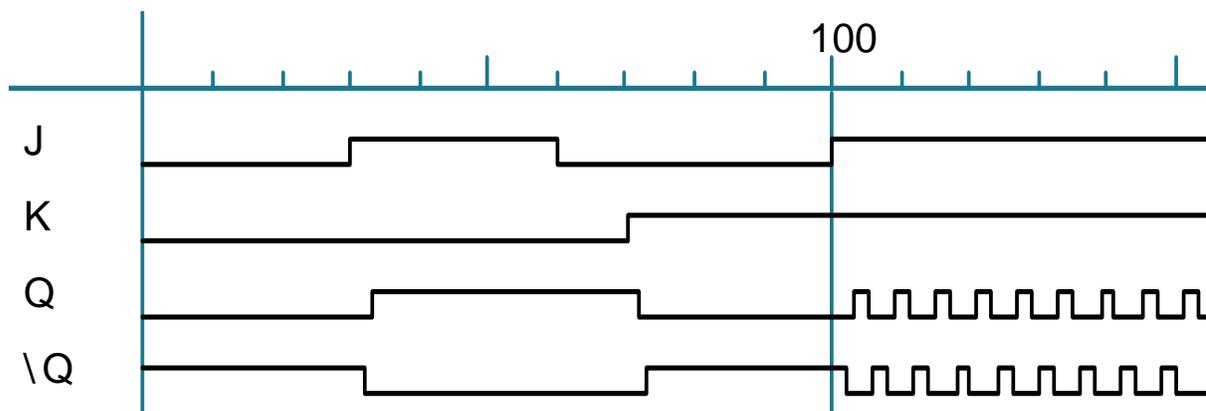
Propagation Delay (T_{cq} for an edge-triggered register and T_{dq} for a latch)

Delay overhead of the memory element

There is a timing "window" around the clocking event during which the input must remain stable and unchanged in order to be recognized



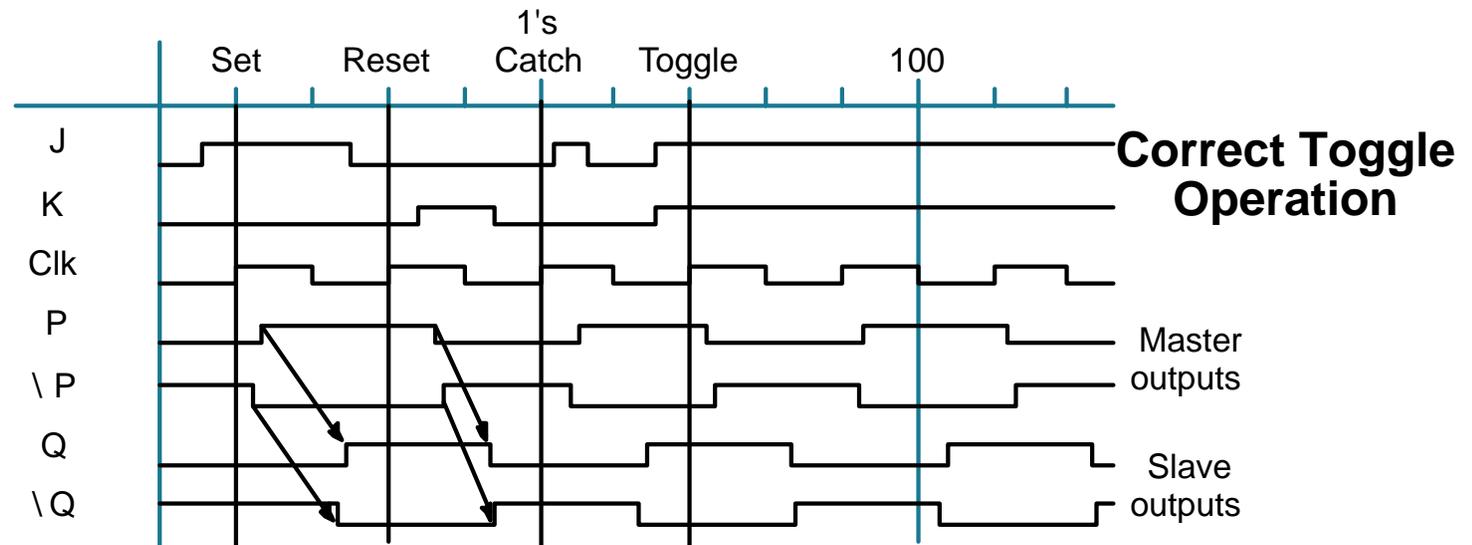
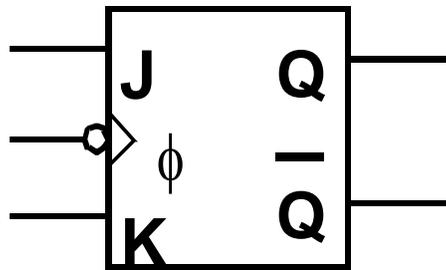
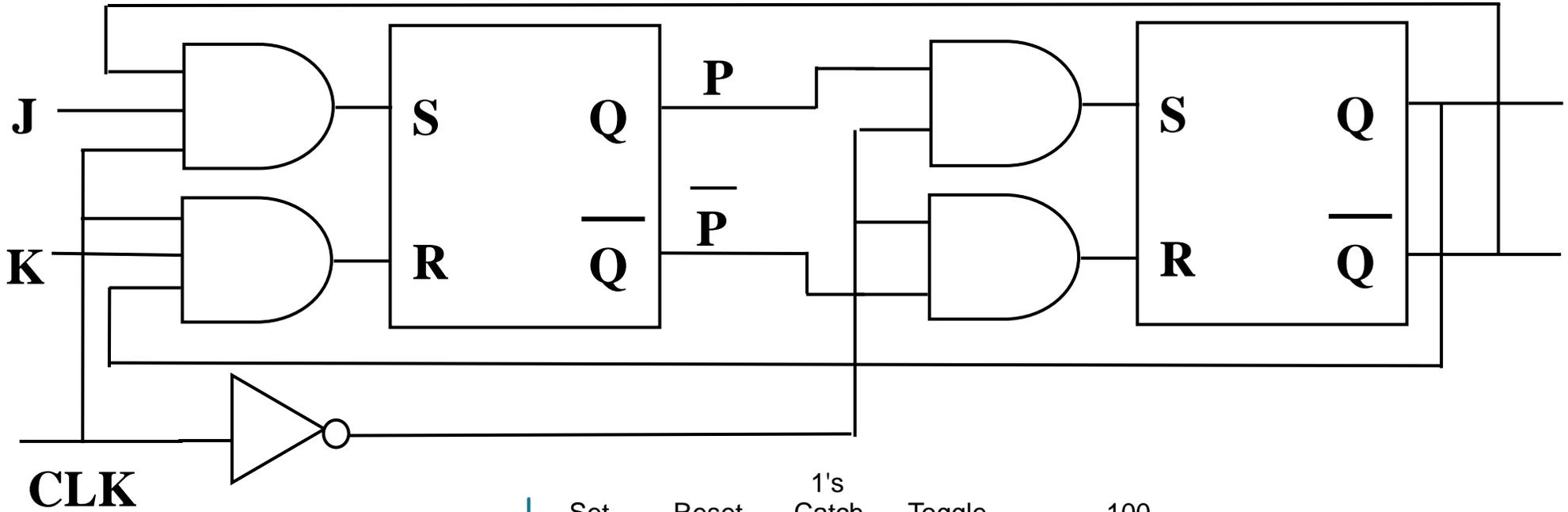
J	K	Q+	\overline{Q} +
0	0	Q	\overline{Q}
0	1	0	1
1	0	1	0
1	1	\overline{Q}	Q



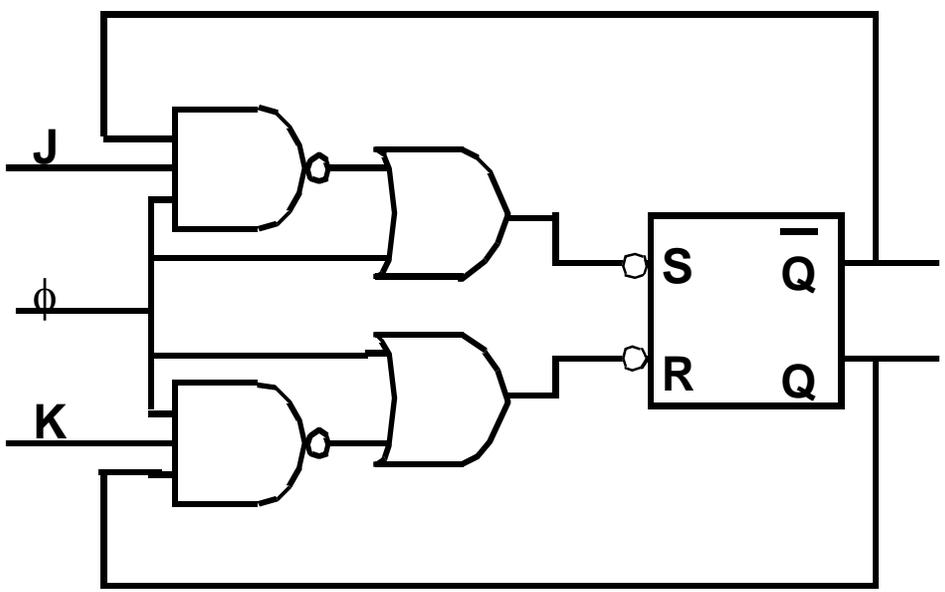
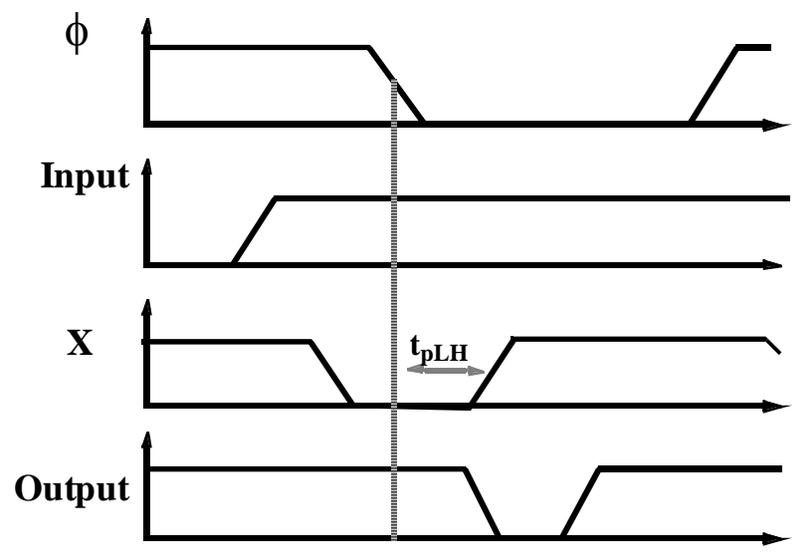
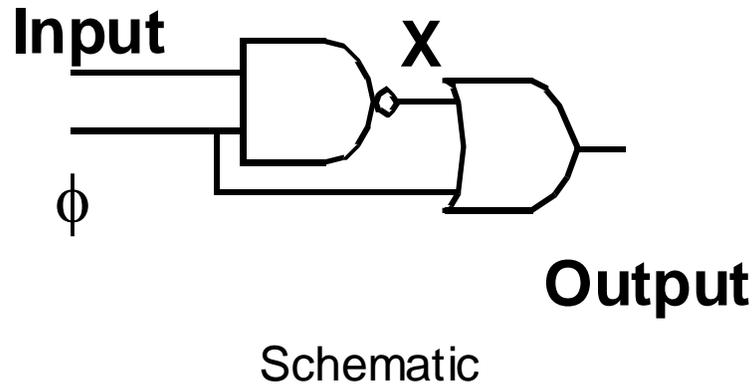
- Eliminate the forbidden state of the SR Flip-flop
- Use output feedback to guarantee that R and S are never both one

Sample inputs while clock high

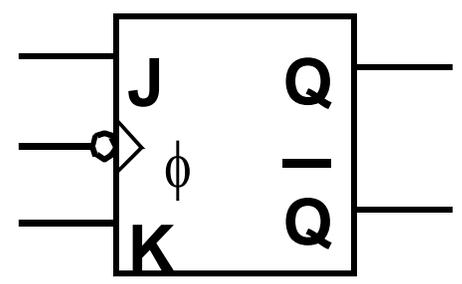
Sample inputs while clock low



Is there a problem with this circuit?



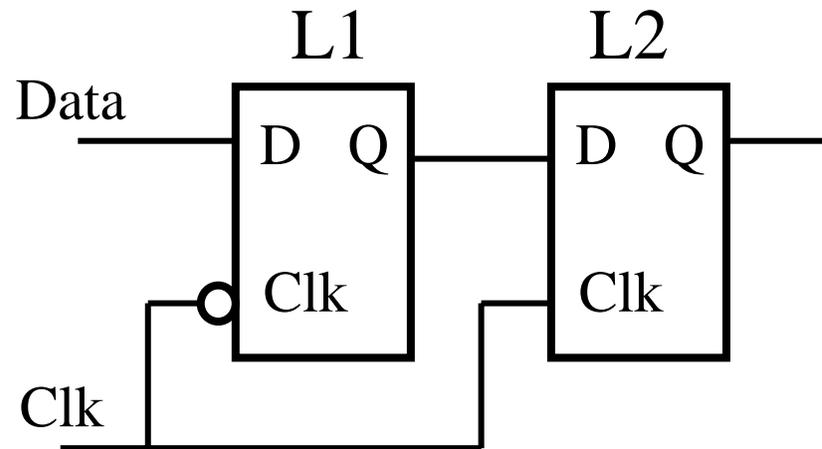
JK Register Schematic



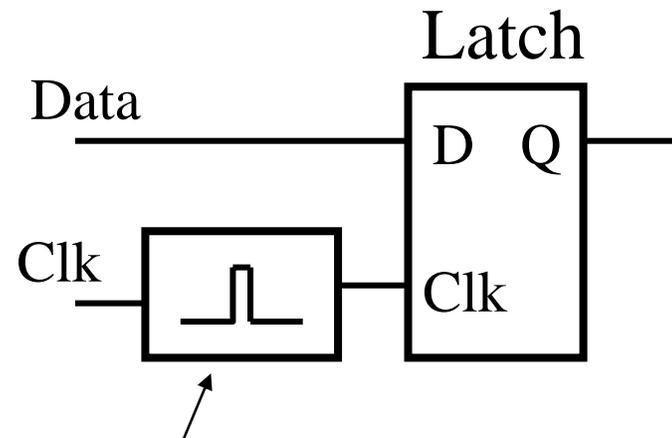
JK Register Logic Symbol

Ways to design an edge-triggered sequential cell:

Master-Slave Latches

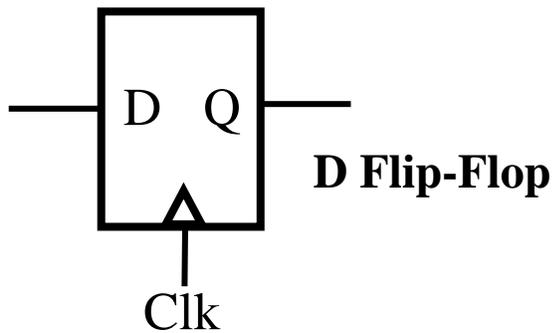


Pulse-Based Register

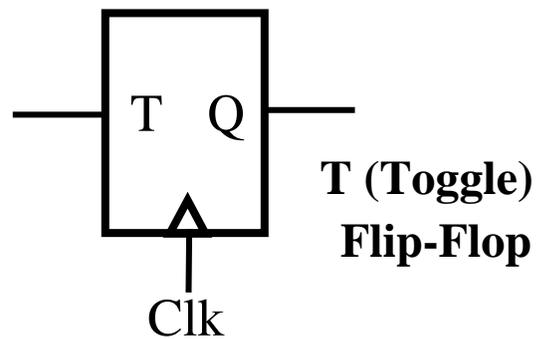
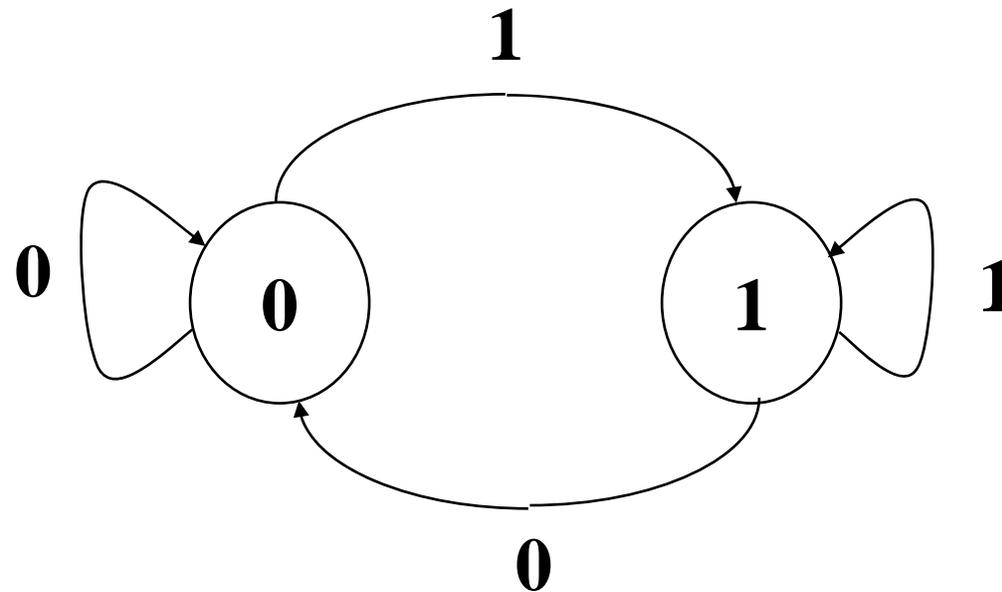


Short pulse around clock edge

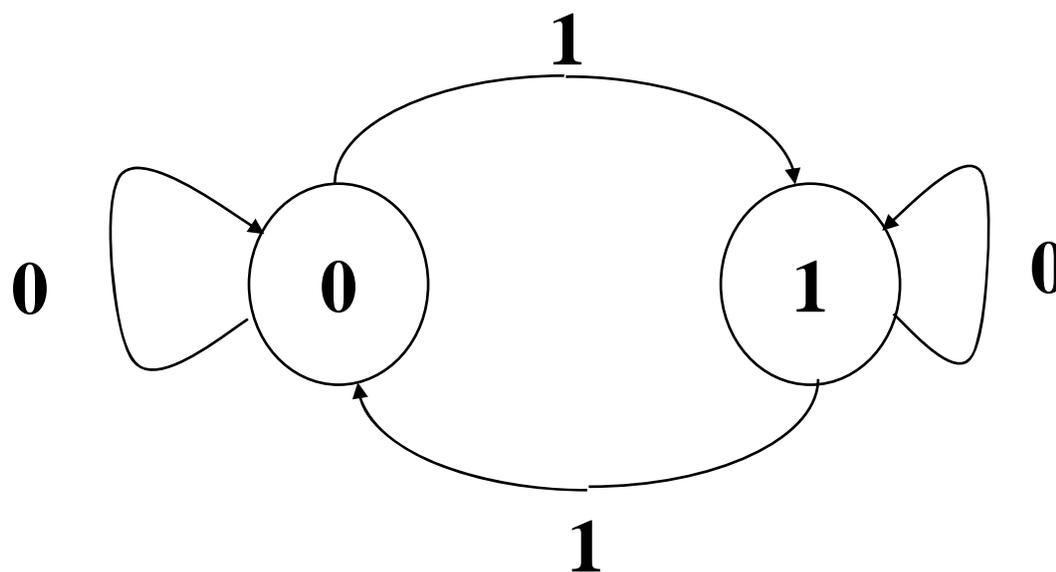
- **Pulse registers are widely used in high-performance microprocessor chips (Sun Microsystems, AMD, Intel, etc.)**
- **The can have a negative setup time!**



D	Q_N
0	0
1	1



T	Q_N
0	Q_{N-1}
1	$\overline{Q_{N-1}}$



Characteristic Equations

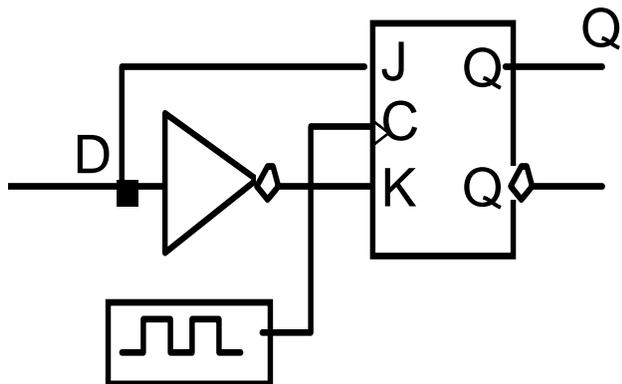
D: $Q_+ = D$

J-K: $Q_+ = J \bar{Q} + \bar{K} Q$

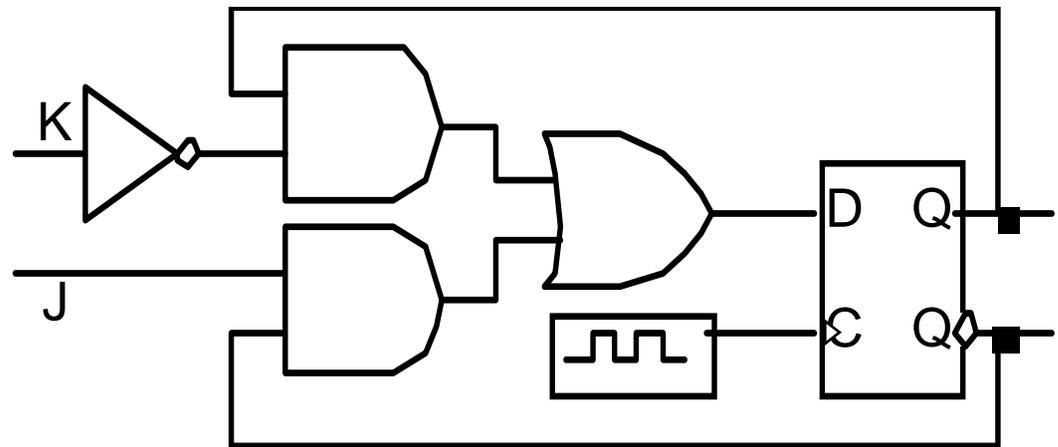
T: $Q_+ = T \bar{Q} + \bar{T} Q$

E.g., J=K=0, then $Q_+ = Q$
 J=1, K=0, then $Q_+ = 1$
 J=0, K=1, then $Q_+ = \bar{Q}$
 J=1, K=1, then $Q_+ = \bar{Q}$

Implementing One FF in Terms of Another



D implemented with J-K



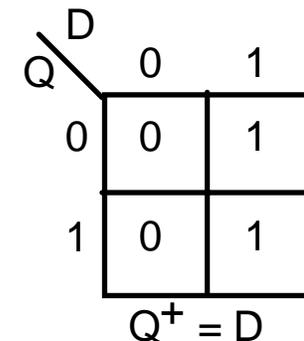
J-K implemented with D

Excitation Tables: What are the necessary inputs to cause a particular kind of change in state?

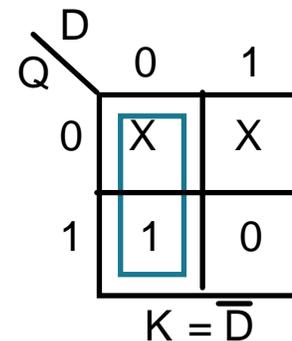
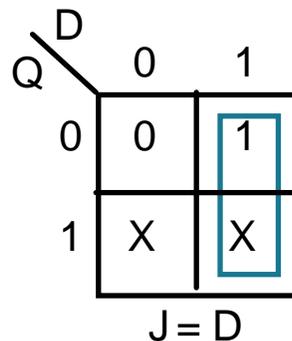
Q	Q+	J	K	T	D
0	0	0	X	0	0
0	1	1	X	1	1
1	0	X	1	1	0
1	1	X	0	0	1

Implementing D FF with a J-K FF:

- 1) Start with K-map of $Q^+ = f(D, Q)$
- 2) Create K-maps for J and K with same inputs (D, Q)
- 3) Fill in K-maps with appropriate values for J and K to cause the same state changes as in the original K-map



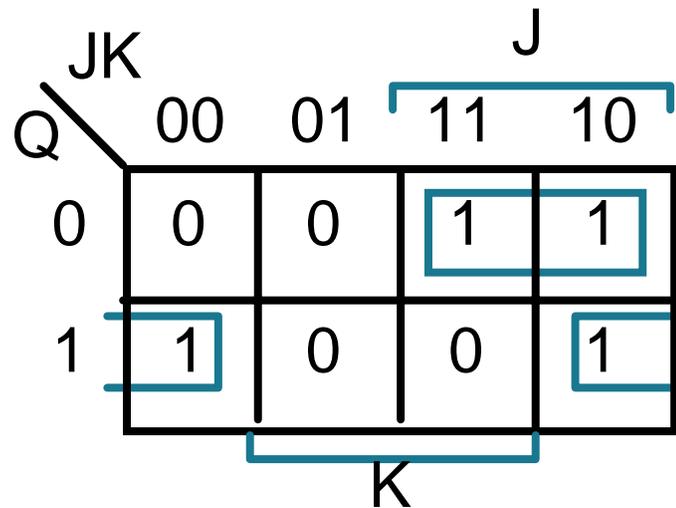
E.g., $D = Q = 0$, $Q^+ = 0$
then $J = 0$, $K = X$



Implementing J-K FF with a D FF:

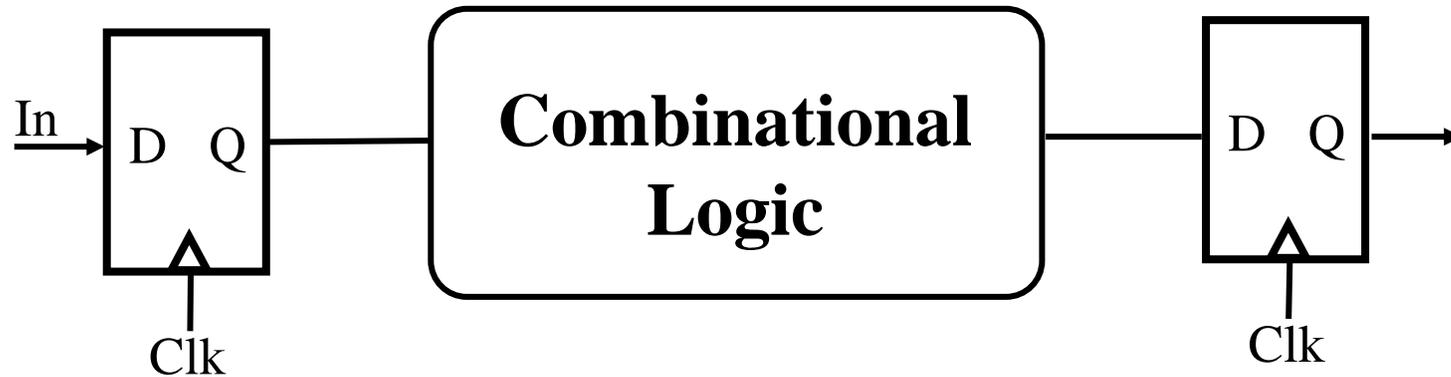
1) K-Map of $Q^+ = F(J, K, Q)$

2,3) Revised K-map using D's excitation table
 its the same! that is why design procedure with D FF is simple!



$$Q^+ = D = J\bar{Q} + \bar{K}Q$$

Resulting equation is the combinational logic input to D to cause same behavior as J-K FF. Of course it is identical to the characteristic equation for a J-K FF.



Register Timing Parameters

T_{cq} : worst case rising edge
clock to q delay

$T_{cq, cd}$: contamination or
minimum delay from
clock to q

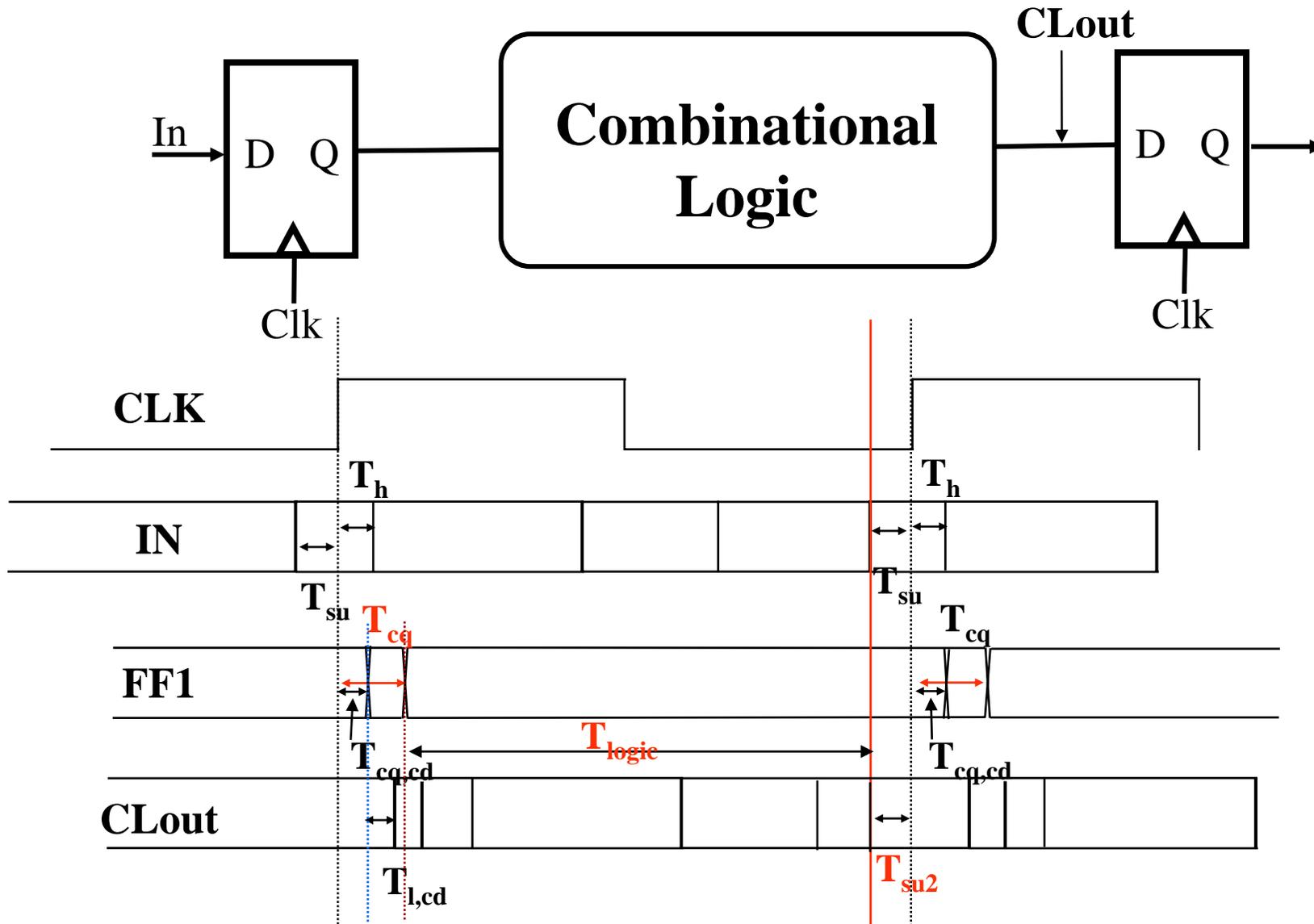
T_{su} : setup time

T_h : hold time

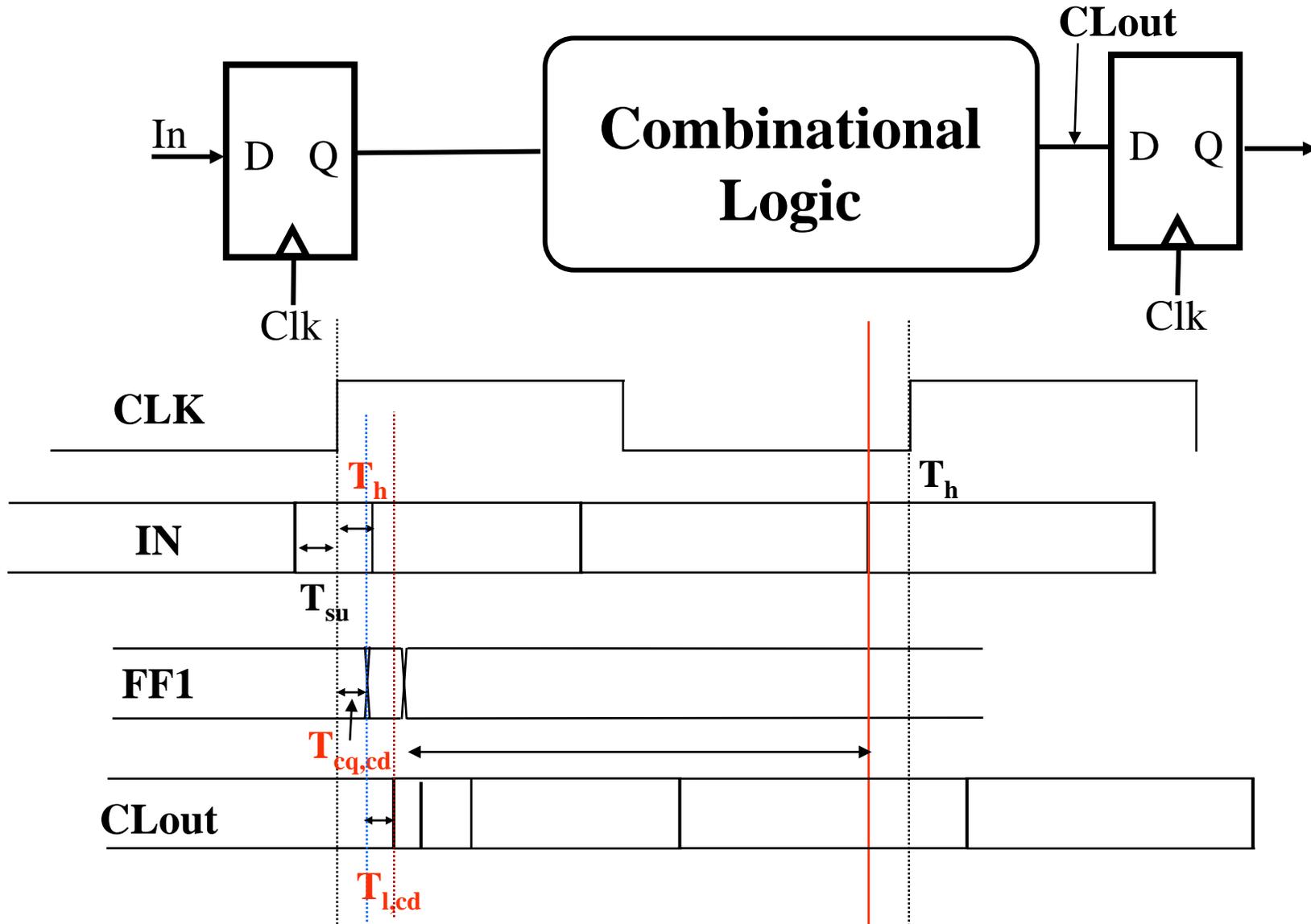
Logic Timing Parameters

T_{logic} : worst case delay
through the combinational
logic network

$T_{logic, cd}$: contamination or
minimum delay
through logic network

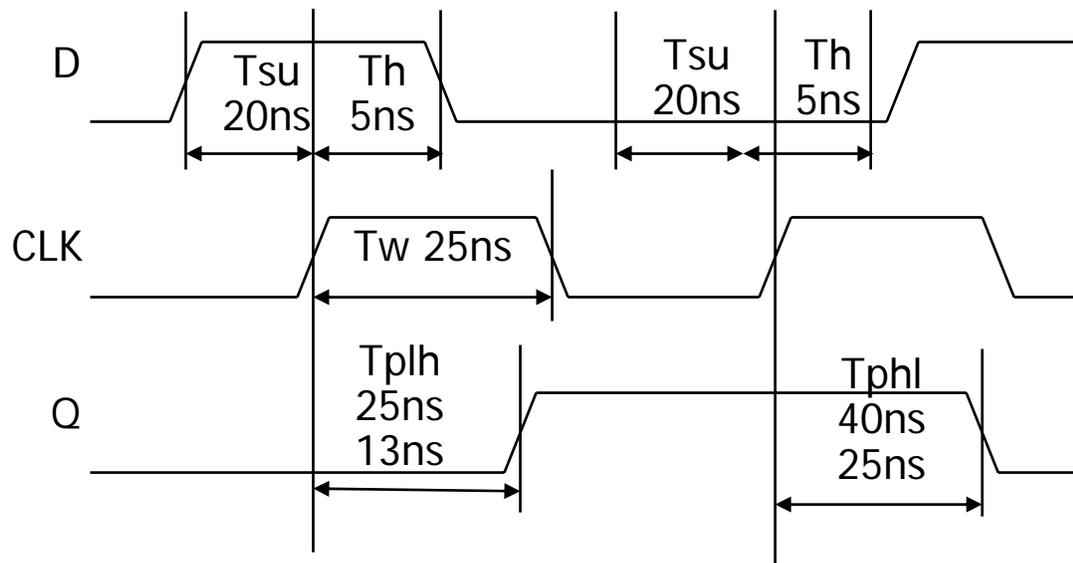


$$T > T_{cq} + T_{logic} + T_{su}$$



$$T_{cq,cd} + T_{logic,cd} > T_{hold}$$

■ Typical parameters for Positive edge-triggered D Register



all measurements are made from the clocking event that is, the rising edge of the clock

■ Shift-register

