



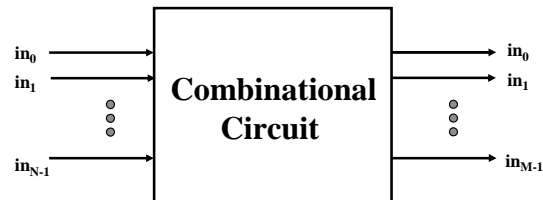
L5: Construction and Analysis of Sequential Building Blocks



Lecture material derived from R. Katz, "Contemporary Logic Design", Addison Wesley Publishing Company, Reading, MA, 1993.



Combinational Logic Review



No feedback in combinational circuits

- Combinational logic circuits are memoryless
- No feedback from inputs and outputs
- Output assumes the function implemented by the logic network, assuming that the switching transients have settled

A Sequential System

- 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

L5: 6.111 Spring 2003 Introductory Digital Systems Laboratory 3

A Simple Example

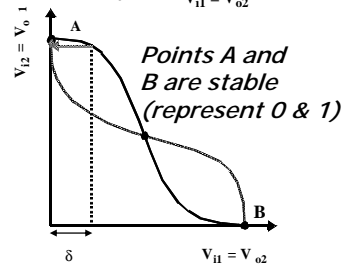
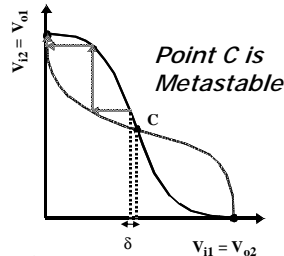
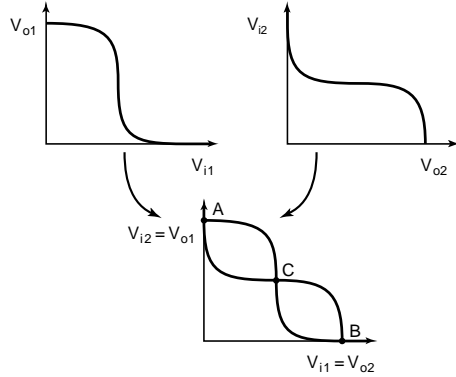
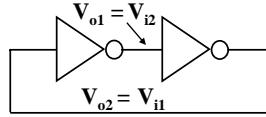
Adding N inputs (N-1 Adders)

Using a sequential (serial) approach

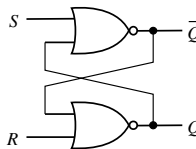
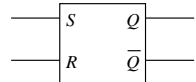
L5: 6.111 Spring 2003 Introductory Digital Systems Laboratory 4



Implementing State: Bi-stability

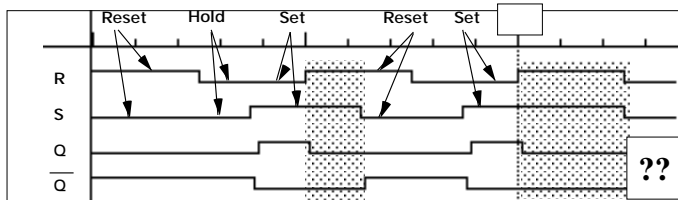
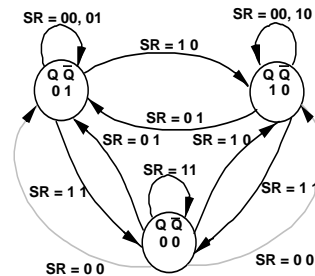


NOR-based Set-Reset (SR) Flipflop



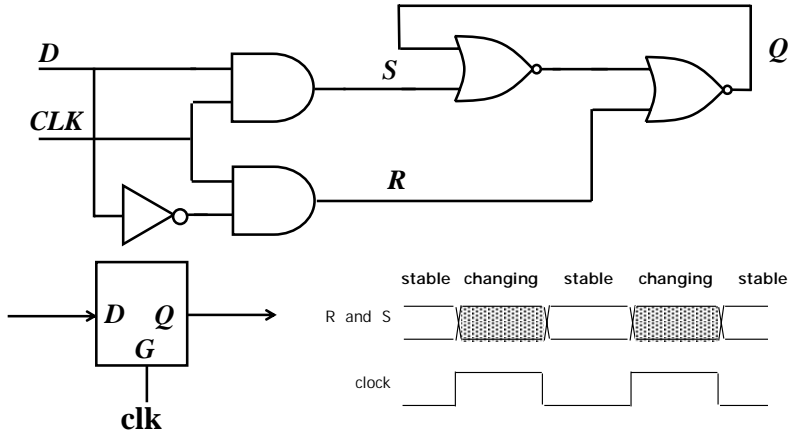
S	R	Q	\bar{Q}
0	0	Q	\bar{Q}
1	0	1	0
0	1	0	1
1	1	0	0

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

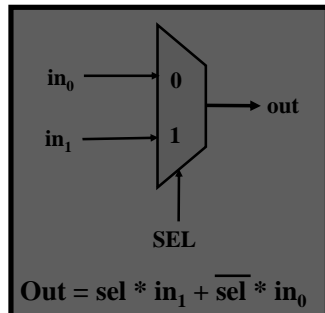
Making a Clocked Memory Element: Positive D-Latch



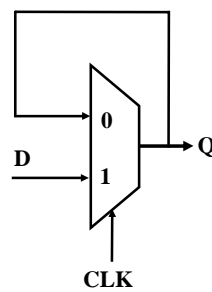
- A Positive D-Latch: Passes input D to Q when clk is high and holds state when clock is low (i.e., ignores input D)
- A Latch is level-sensitive: invert clock for a negative latch

Multiplexor based positive & negative latch

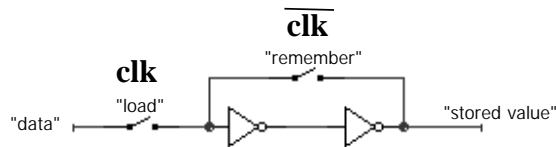
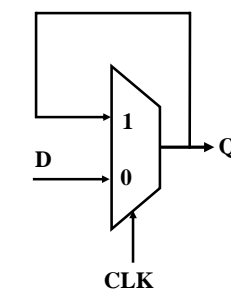
2:1 multiplexor



Positive Latch

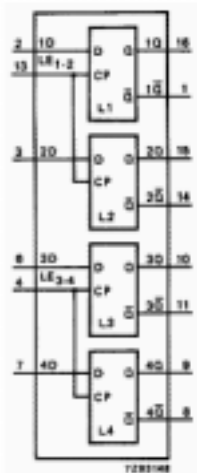


Negative Latch





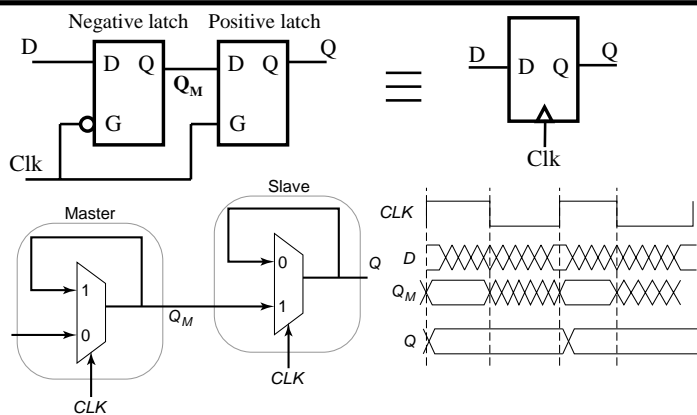
74HC75 (positive Latch)



OPERATING MODES	INPUTS		OUTPUTS	
	LE _{n-n}	nD	nQ	n \bar{Q}
data enabled	H	L	L	H
data latched	L	X	q	\bar{q}



Building an Edge-Triggered Register



■ Master-Slave Register

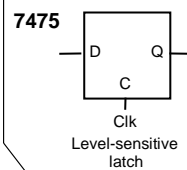
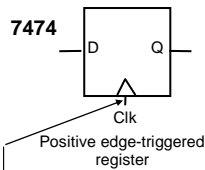
- Use negative clock phase to latch inputs into first latch
- Use positive clock to change outputs with second latch

■ View pair as one basic unit

- master-slave flip-flop twice as much logic



Latches vs. Edge-Triggered Register

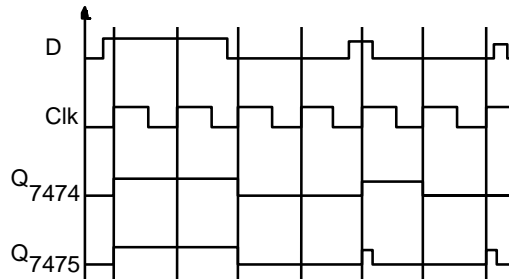


Bubble here for negative edge triggered register

Edge triggered device sample inputs on the event edge

Transparent latches sample inputs as long as the clock is asserted

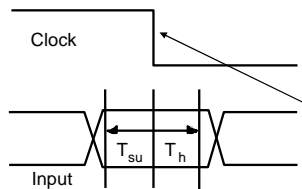
Timing Diagram:



Behavior the same unless input changes while the clock is high



Important Timing Parameters



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)

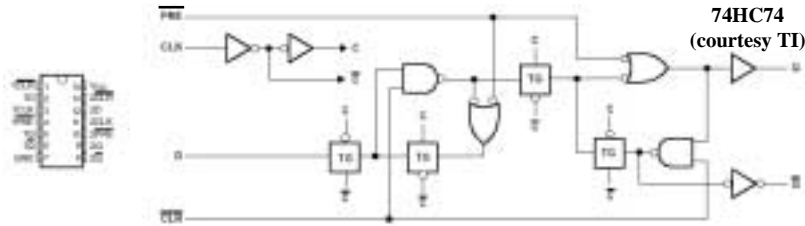
Minimum time after the clocking event during which the input must remain stable

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

74HC74 (Positive Edge-Triggered Register)



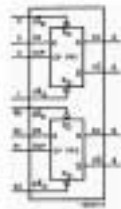
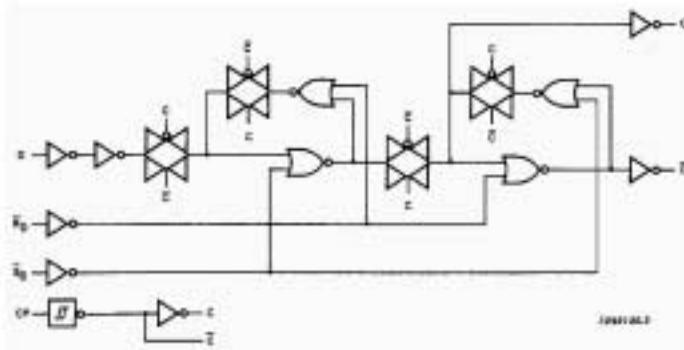
FUNCTION TABLE

INPUTS				OUTPUTS	
PRE	CLR	CLK	D	Q	\bar{Q}
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H ¹	L ¹
H	H	T	H	H	L
H	H	T	L	L	H
H	H	L	X	Q _p	\bar{Q}_p

D-FF with preset and clear

Type	Clock frequency	V _{CC} (V)	t _{PROP} (ns)		t _{SETUP} (ns)		LIMIT
			MIN	MAX	MIN	MAX	
Type 1	PRE or CLR active	2.0	0	0	0	0	400
		4.5	0	0	0	0	20
		5.0	0	0	0	0	20
Type 2	CLK edge or low	2.0	100	100	100	100	10
		4.5	10	20	10	10	10
		5.0	10	20	10	10	10
Type 3	Data	2.0	100	100	100	100	10
		4.5	20	20	20	20	10
		5.0	17	20	20	20	10
Type 4	PRE or CLR inactive	2.0	0	0	0	0	10
		4.5	0	0	0	0	10
		5.0	0	0	0	0	10

Philips 74HC74 Implementation

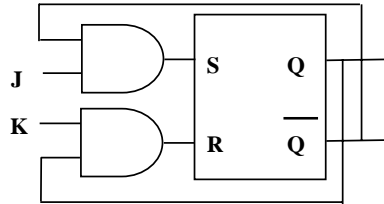


INPUTS				OUTPUTS	
\bar{P}_0	\bar{P}_1	CP	D	Q	\bar{Q}
L	H	0	0	H	L
L	L	X	0	L	H
L	L	X	0	H	L

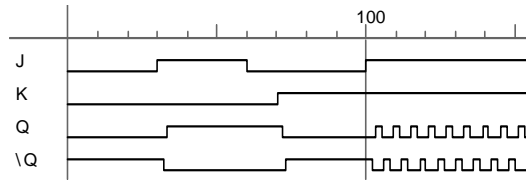
INPUTS			OUTPUTS	
\bar{P}_0	\bar{P}_1	CP	Q _{n+1}	Q _n
H	H	1	L	L
H	H	1	H	H



The J-K Flip-Flop



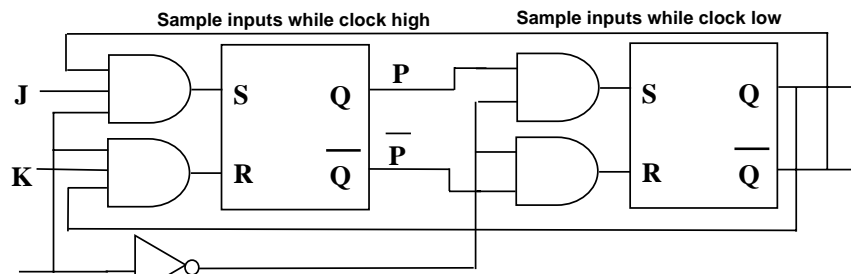
J	K	Q+	Q+
0	0	Q	Q
0	1	0	1
1	0	1	0
1	1	Q	Q



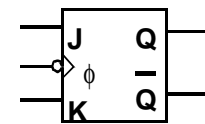
- Eliminate the forbidden state of the SR Flip-flop
- Use output feedback to guarantee that R and S are never both one



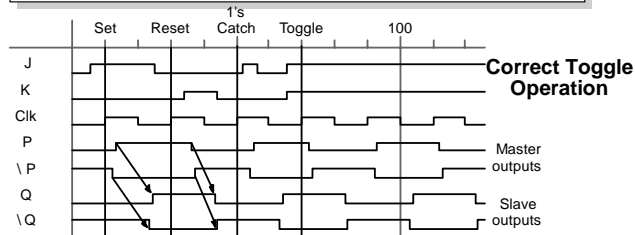
J-K Master-Slave Register



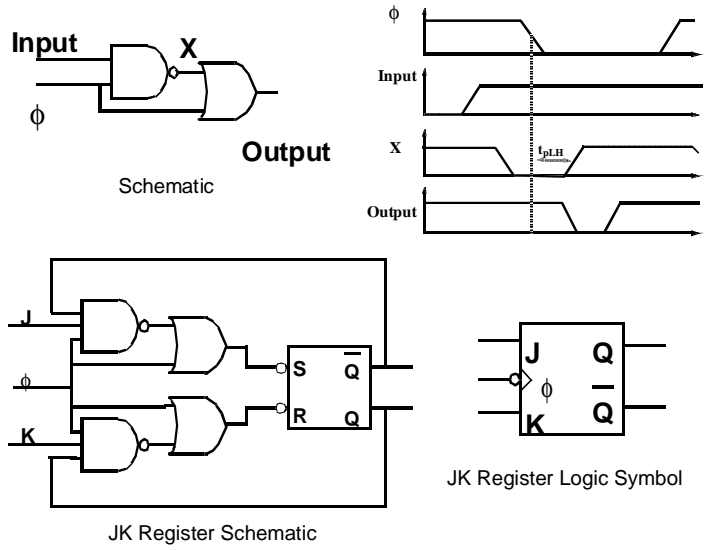
Uses time to break feedback path from outputs to inputs!



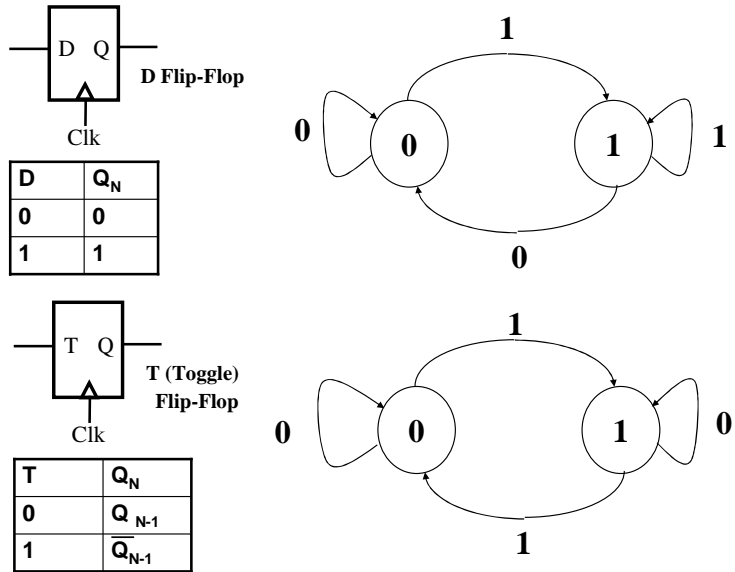
J-K Logic Symbol



Pulse Based Edge-Triggered J-K Register



D Flip-Flop vs. Toggle Flip-Flop





Realizing different types of memory elements



Characteristic Equations

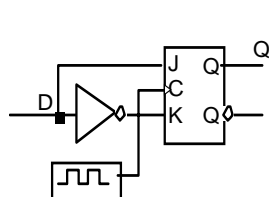
$$D: Q^+ = D$$

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

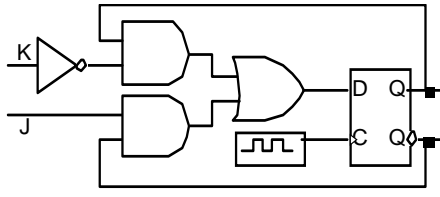
$$T: Q^+ = T\bar{Q} + \bar{T}Q$$

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



Design Procedure



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

		D	
		0	1
Q	0	0	1
	1	0	1

$Q^+ = D$

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

		D	
		0	1
Q	0	0	1
	1	X	X

$J = D$

		D	
		0	1
Q	0	X	X
	1	1	0

$K = \bar{D}$



Design Procedure (cont.)



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!

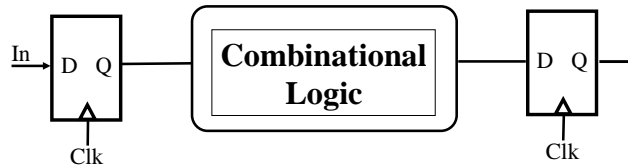
		J			
		00	01	11	10
Q	JK	00	01	11	10
	0	0	0	1	1
1	1	1	0	0	1
		K			

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

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.



System Timing Parameters



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

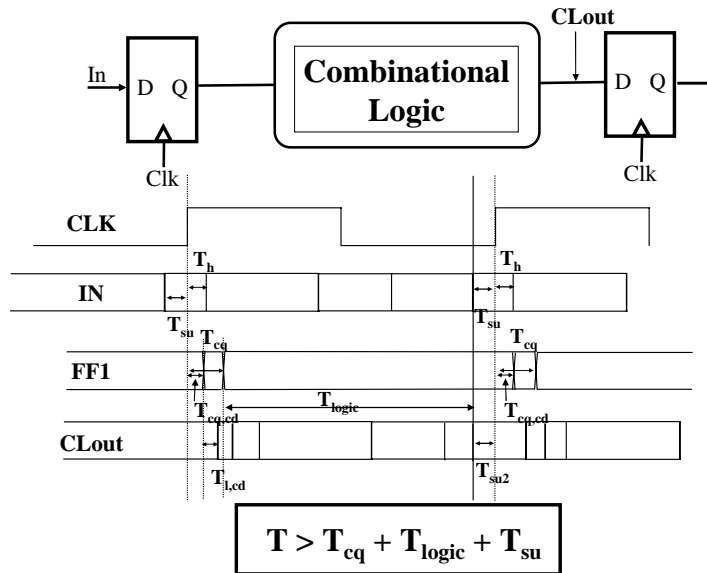
Logic Timing Parameters

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

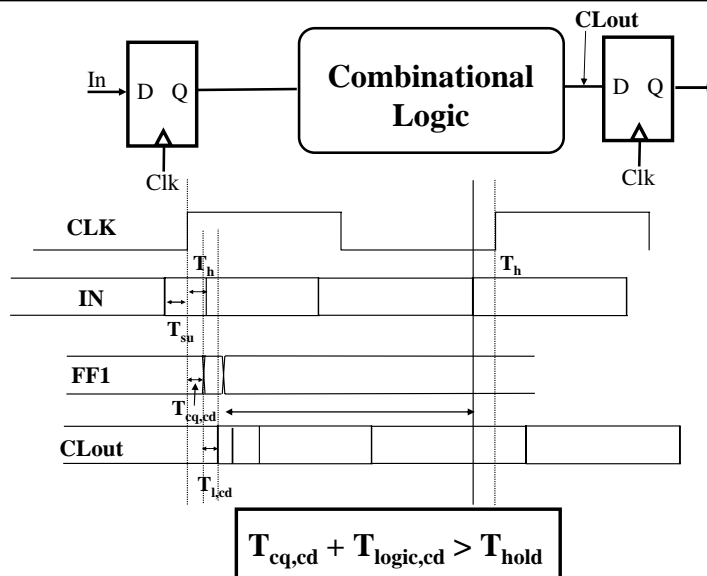
$T_{logic,cd}$: contamination or
minimum delay
through logic network



System Timing (I): Minimum Period



System Timing (II): Minimum Delay

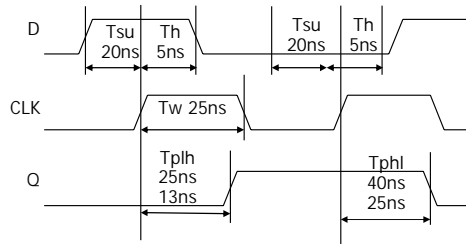




Shift-Register

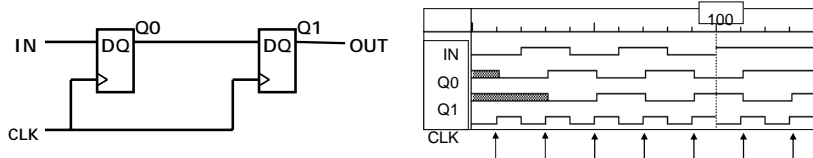


■ Typical parameters for Positive edge-triggered D Register

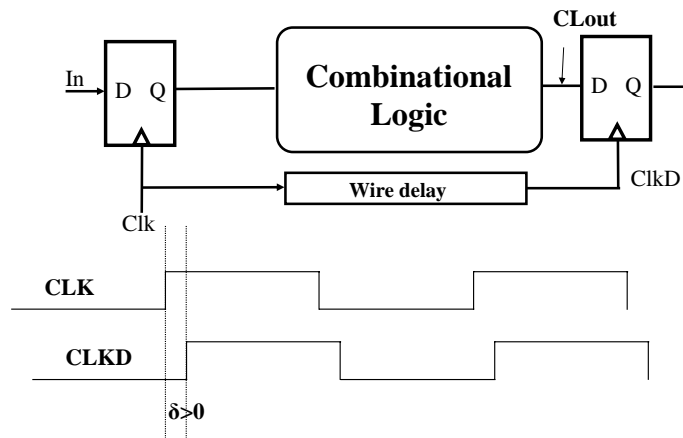


all measurements are made from the clocking event that is, the rising edge of the clock

■ Shift-register



Clocks are not perfect: Clock Skew



$$T > \underline{\hspace{2cm}}$$