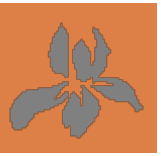
UNIT 11
LATCHES AND FLIP-FLOPS



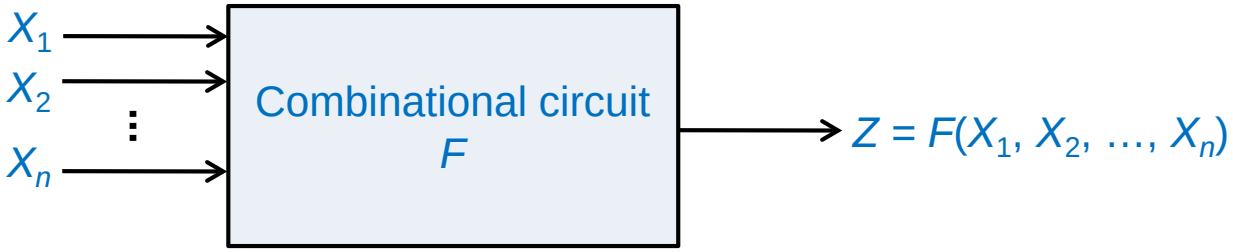
Spring 2011

Latches and Flip-Flops

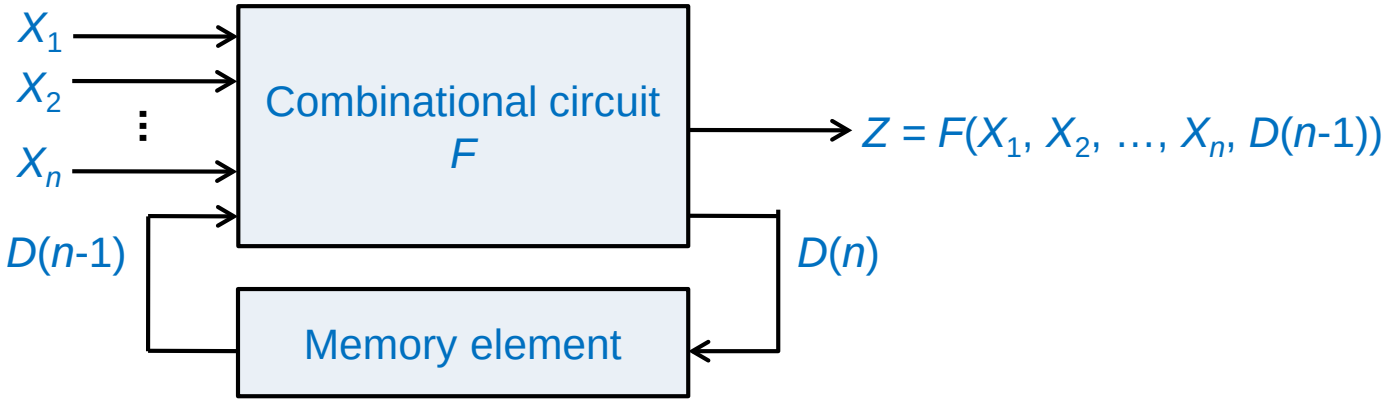
- **Contents**
  - Set-Reset latch
  - Gated D latch
  - Edge-triggered D flip-flop
  - S-R flip-flop
  - J-K flip-flop
  - T flip-flop
  - Flip-flops with additional inputs
- **Reading**
  - Unit 11

Recap: Two Types of Switching Circuits

- **Combinational circuits (memoryless)**
  - Outputs depend only on present inputs

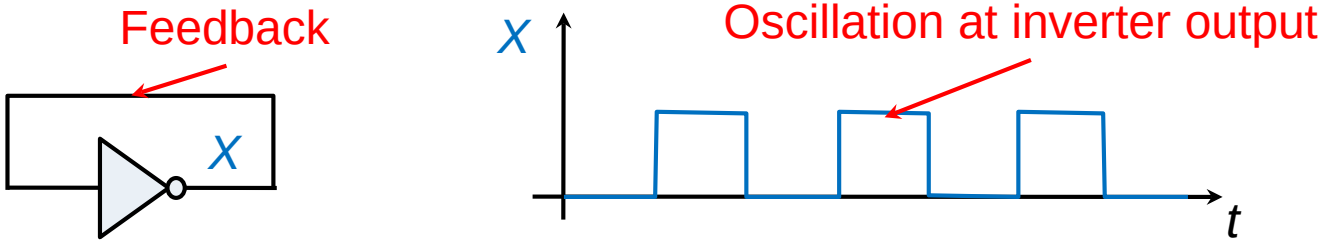


- **Sequential circuits (with memory)**
  - Outputs depend on both present & past inputs
  - In general, sequential ckts = combinational ckts + memory

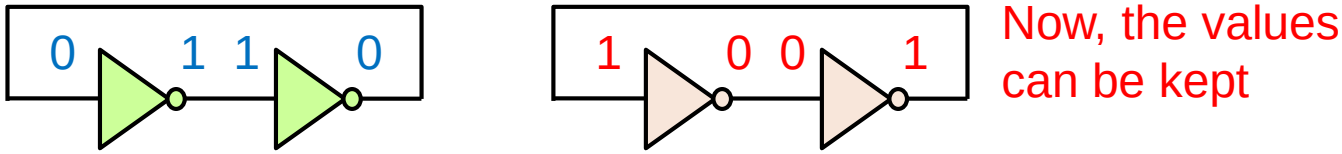


How to Remember the Past?

- **Feedback: the output of one of the gates is connected back into the input of another gate in the ckt so as to form a closed loop**
  - e.g., inverter with feedback
    - Q: How fast does the circuit oscillate?
    - A: Determined by the propagation delay of the inverter



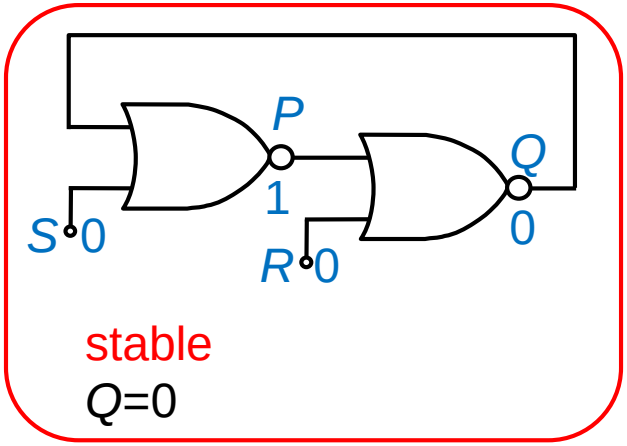
  - e.g., a feedback loop with two inverters
    - Two stable states
    - Latch: basic memory unit (store 1 bit)



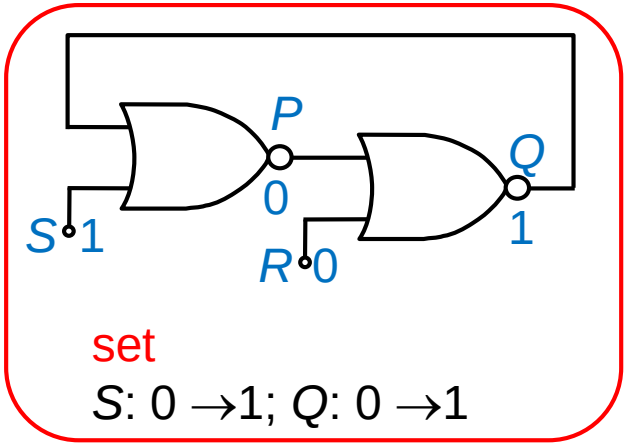
Set-Reset Latch

S-R latch

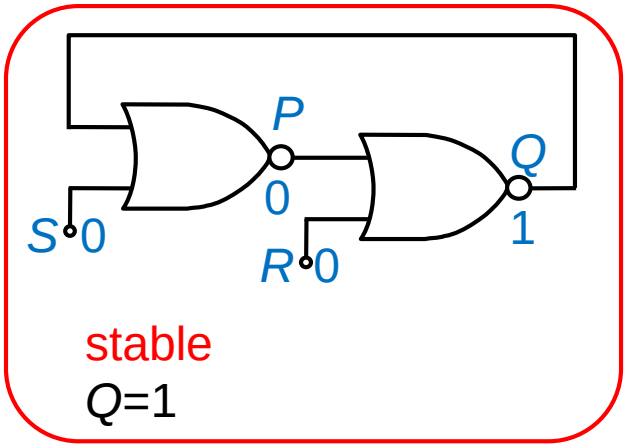
Set-Reset (S-R) Latch (1/2)



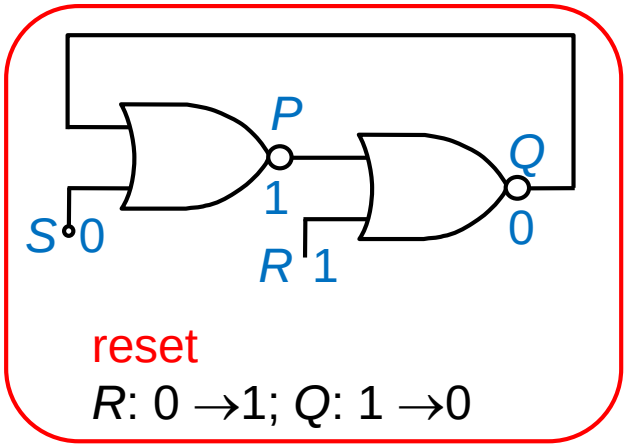
stable
Q=0



set
S: 0 →1; Q: 0 →1



stable
Q=1

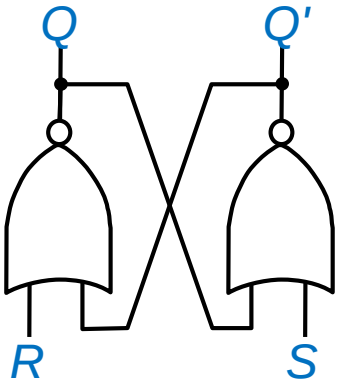


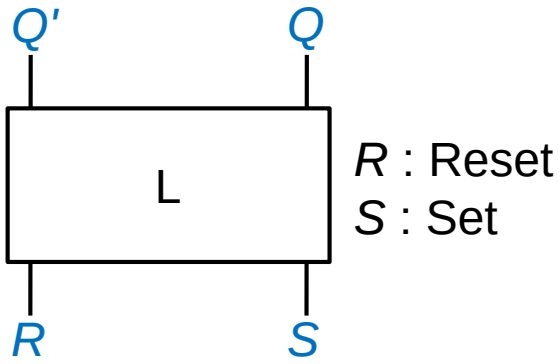
reset
R: 0 →1; Q: 1 →0

$P = Q'$

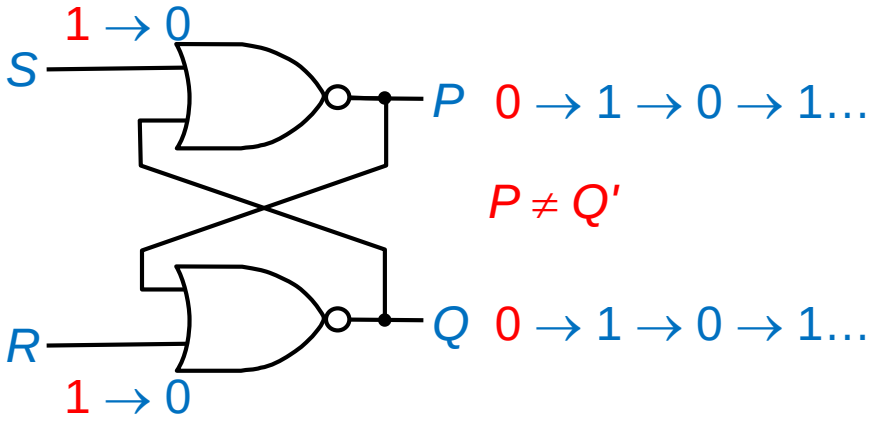
S-R Latch (2/2)

- **Cross-coupled form**



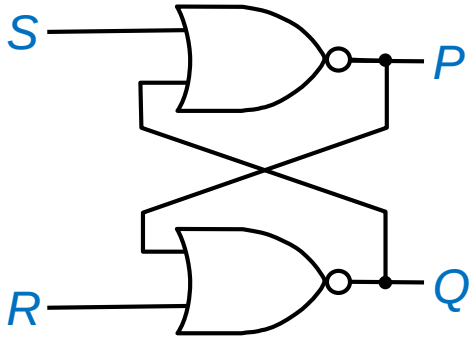


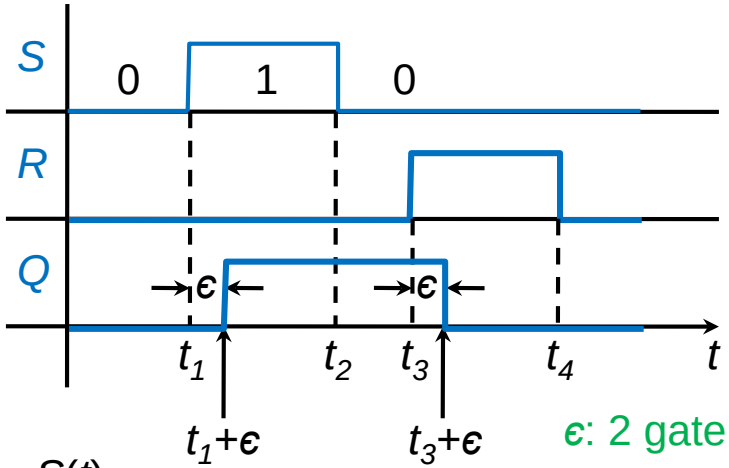
- ***S* = *R* = 1 ⇒ Unstable! Not allowed!!**



Next-State Equation

- **Timing diagram**





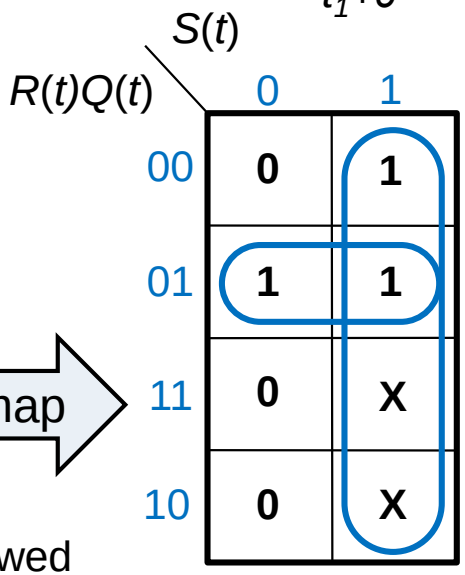
ϵ: 2 gate delay

- **Operation**

$S(t)$	$R(t)$	$Q(t)$	$Q(t+\epsilon)$	
0	0	0	0	Unchanged
0	0	1	1	
0	1	0	0	Reset to 0
0	1	1	0	
1	0	0	1	Set to 1
1	0	1	1	
1	1	0	–	Inputs not allowed
1	1	1	–	

Latches & FFs

K-map



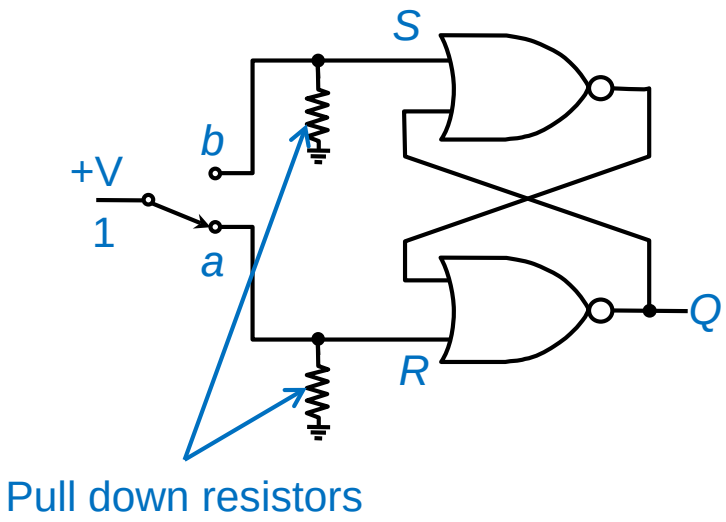
$Q(t+\epsilon)=S(t)+R'(t)Q(t)$

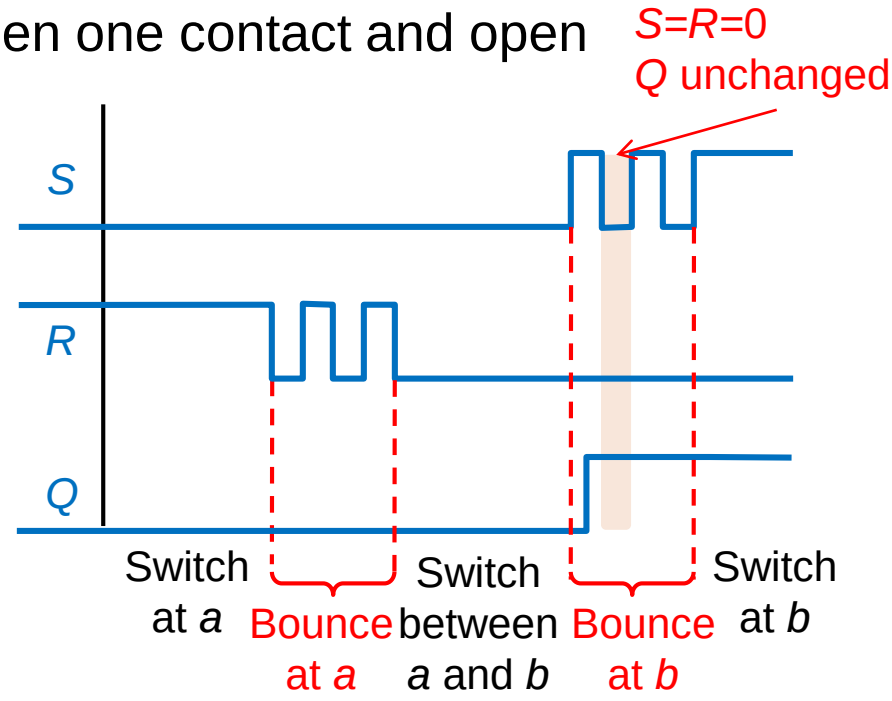
Next-state equation:
(Characteristic equation)

$Q^+ = S+R'Q$
under $SR=0$
(S=1, R=1 not allowed)

Application: Switch Debouncing

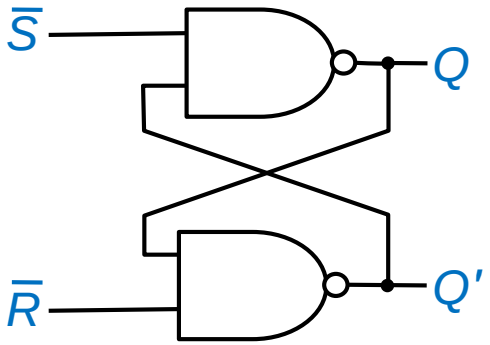
- **When a mechanical switch is opened or closed, switch contacts tend to vibrate before settling down ⇒ Debounce with S-R latch**
  - e.g, when the switch is flipped from *a* to *b*…
  - Work only with a "double throw" switch
    - Double throw: switch between two contacts
    - Single throw: switch between one contact and open

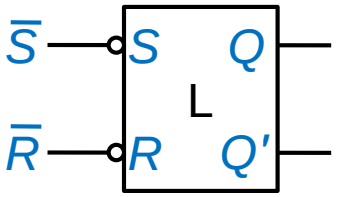




Alternative Form with NAND-Gates

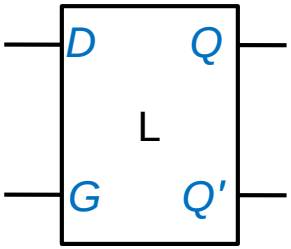
- **$\bar{S}$-$\bar{R}$ latch: active-low inputs for *S* & *R***



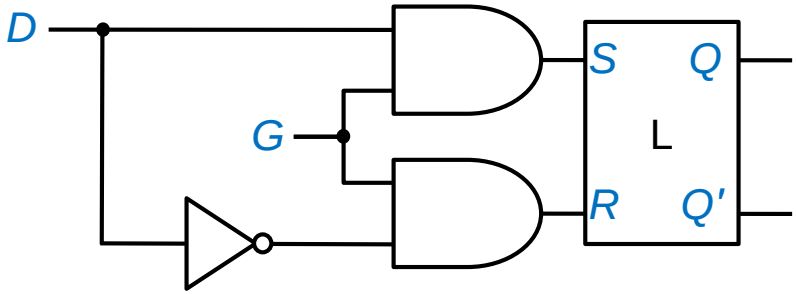


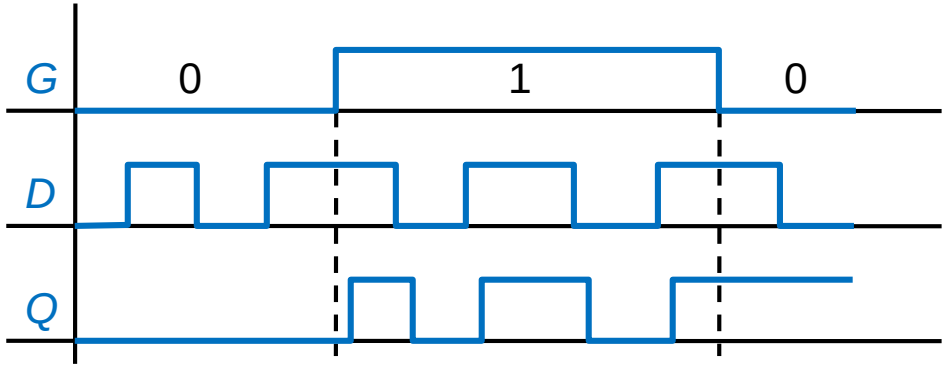
$\bar{S}$	$\bar{R}$	Q	Q^+	
1	1	0	0	Unchanged
1	1	1	1	
1	0	0	0	Reset to 0
1	0	1	0	
0	1	0	1	Set to 1
0	1	1	1	
0	0	0	–	Inputs not allowed
0	0	1	–	

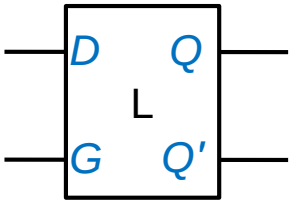
Gated D Latch



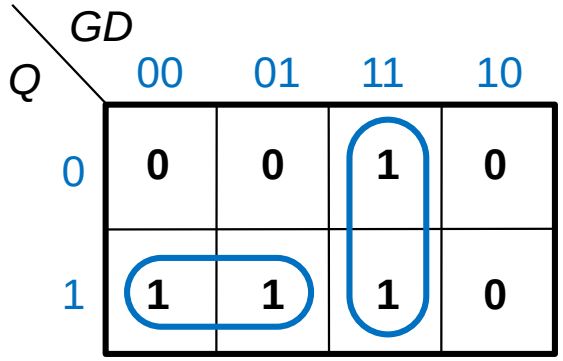
Gated D Latch





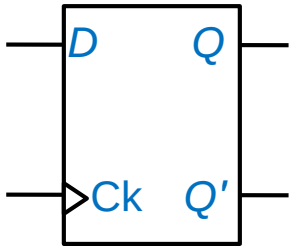


G	D	Q	Q^+	
0	0	0	0	Unchanged
0	0	1	1	$Q^+ = Q$
0	1	0	0	
0	1	1	1	
1	0	0	0	Update
1	0	1	0	$Q^+ = D$
1	1	0	1	
1	1	1	1	



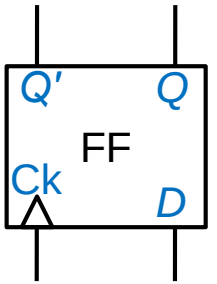
$Q^+ = G'Q + GD$

Edge-Triggered D Flip-Flop

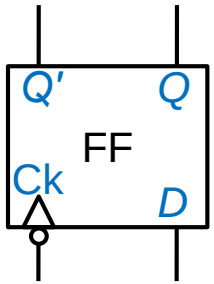


Edge-Triggered D Flip-Flops

- **Output changes are aligned with clock edges**
  - Positive (rising-edge) trigger
  - Negative (falling-edge) trigger
  - Symbol



Rising-edge trigger



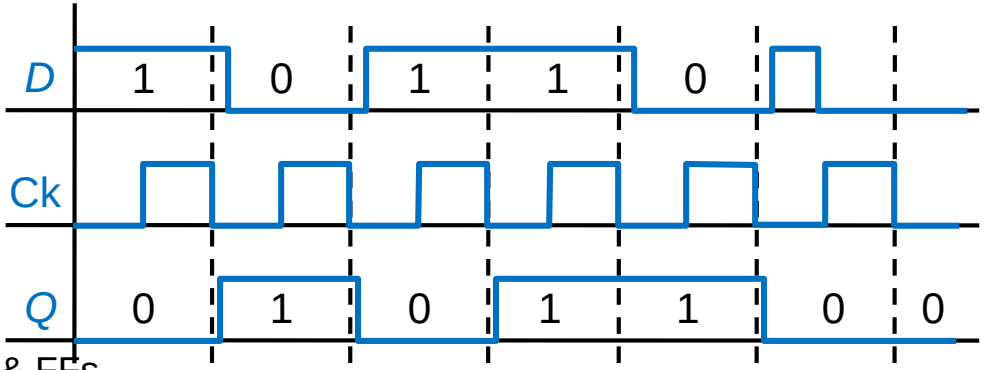
Falling-edge trigger

  - Truth table

D	Q	Q^+
0	0	0
0	1	0
1	0	1
1	1	1

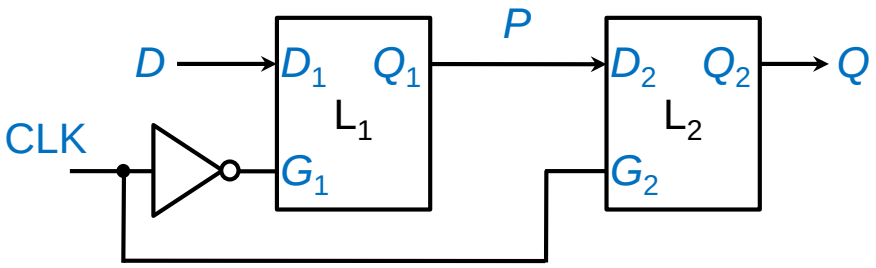
$Q^+ = D$

  - Timing diagram for falling-edge triggered D FF

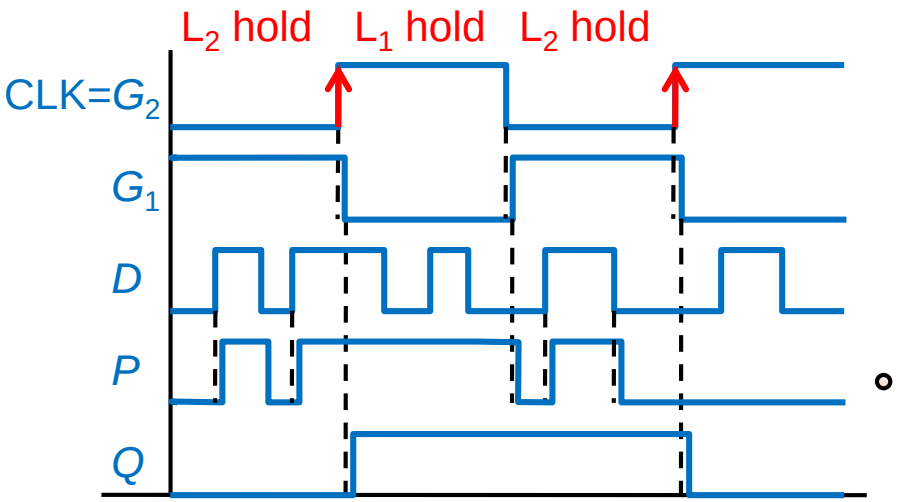


D FF – Rising Edge Trigger

- **Construct from 2 gated D latches**



- **Timing diagram (setup/hold time)**

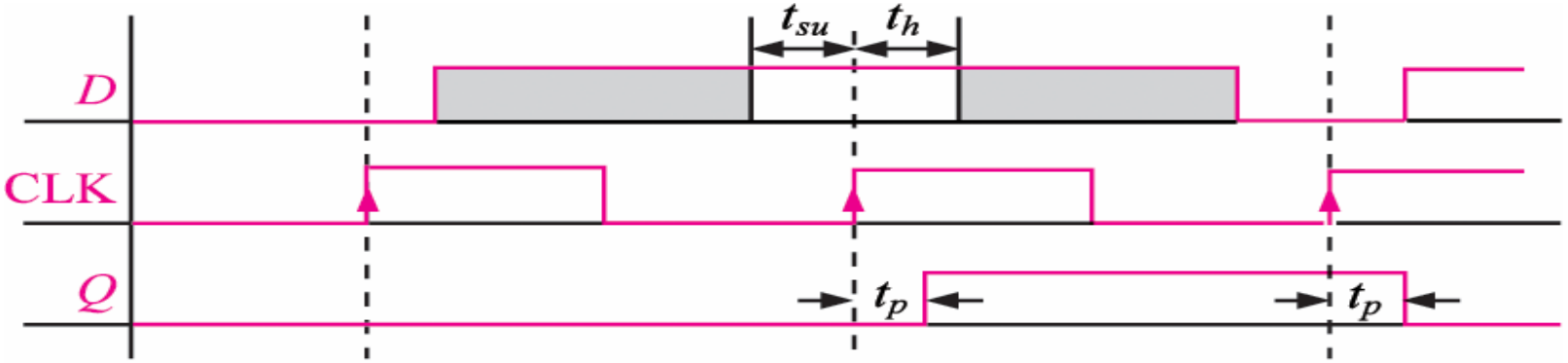


Q: What's the difference between a latch and a FF?

The value of a flip-flop' output won't change during a clock cycle.

Timing of a D FF (Setup Time/Hold Time)

- Because a flip-flop changes state only on the active edge of the clock, the *propagation delay* (t_p) *of a flip-flop* is the time between the active edge of the clock and the resulting change in the output.
- To function properly, the D input to an edge-triggered flip-flop must be held at a constant value for a period of time before and after the active edge of the clock.
- The amount of time that D must be stable before the active edge is called the *setup time* (t_{su}), and the amount of time that D must hold the same value after the active edge is the *hold time* (t_h).



Minimum Clock Period

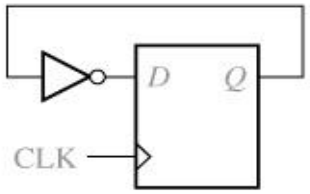
- **Determine the Minimum Clock Period**
  - Minimum clock period

    =(total delay of gates) + (delay of flip-flop + setup time)
- **Example: Suppose that**
  - Propagation delay of the inverter = 2*ns*,
  - Propagation delay of the D flip-flop = 5*ns*,
  - Setup time of the D Flip-flop = 3*ns*.

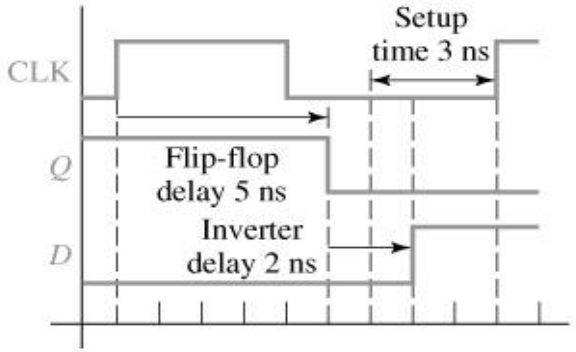
  Determine the minimum clock period of the circuit.

Edge-Triggered D Flip-Flop

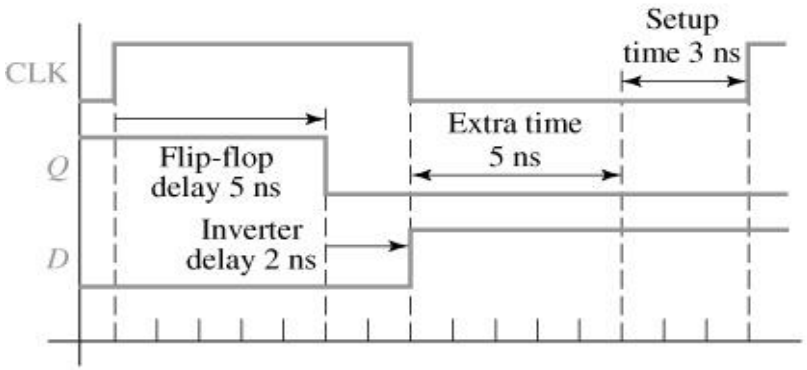
- **Example (cont.):**



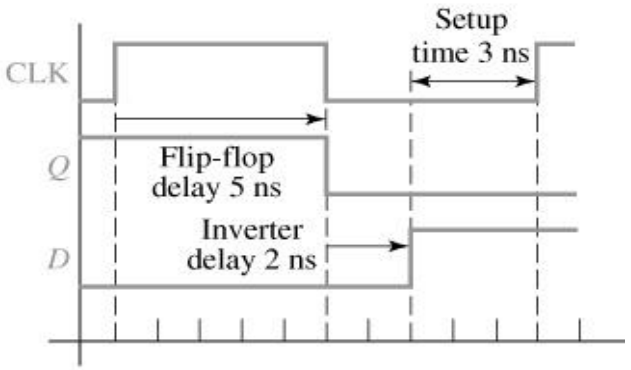
(a) Simple flip-flop circuit



(b) Setup time not satisfied

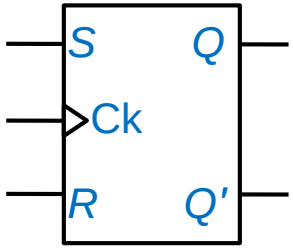


(c) Setup time satisfied

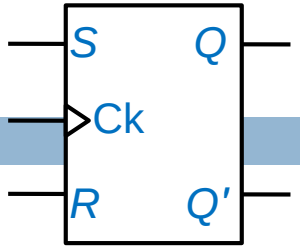


(d) Minimum clock period

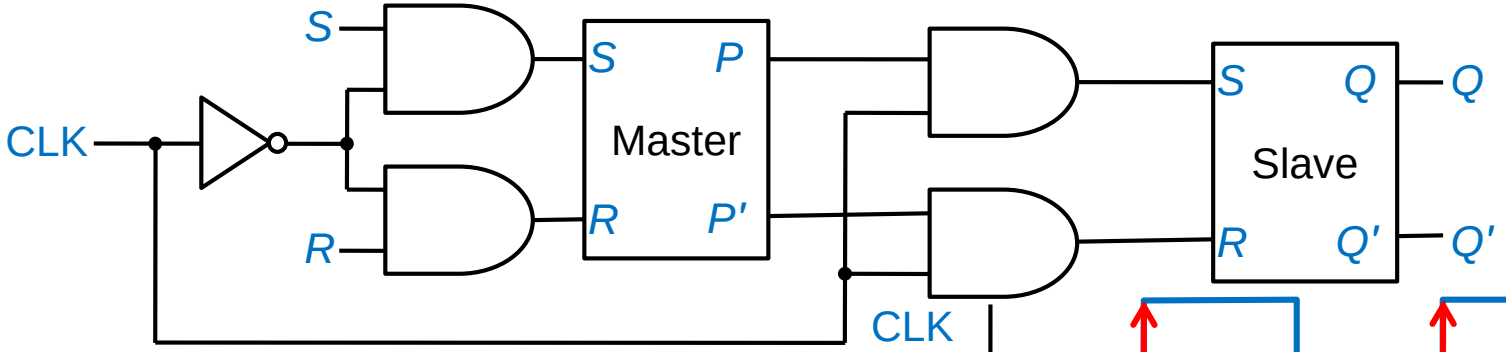
S-R Flip-Flop



S-R Flip Flop

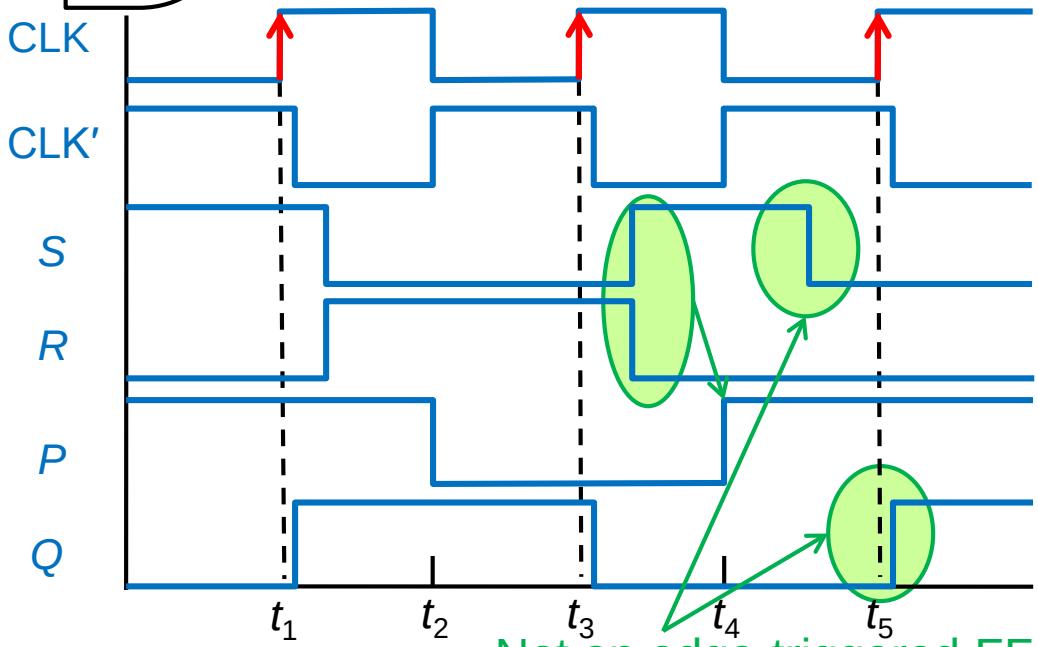


- **Output changes at clock edges**
- **Construct from 2 latches**



- **Operation**

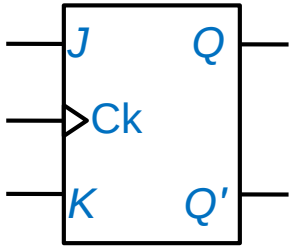
S	R	*operation*
0	0	No state change
0	1	Reset *Q* to 0 (after active Ck edge)
1	0	Set *Q* to 1 (after active Ck edge)
1	1	Not allowed



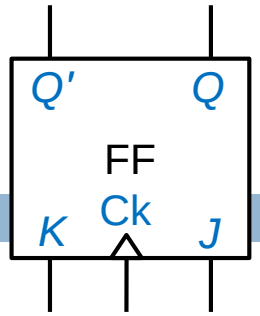
Not an edge-triggered FF

⇒ Do not allow SR change while CLK is low

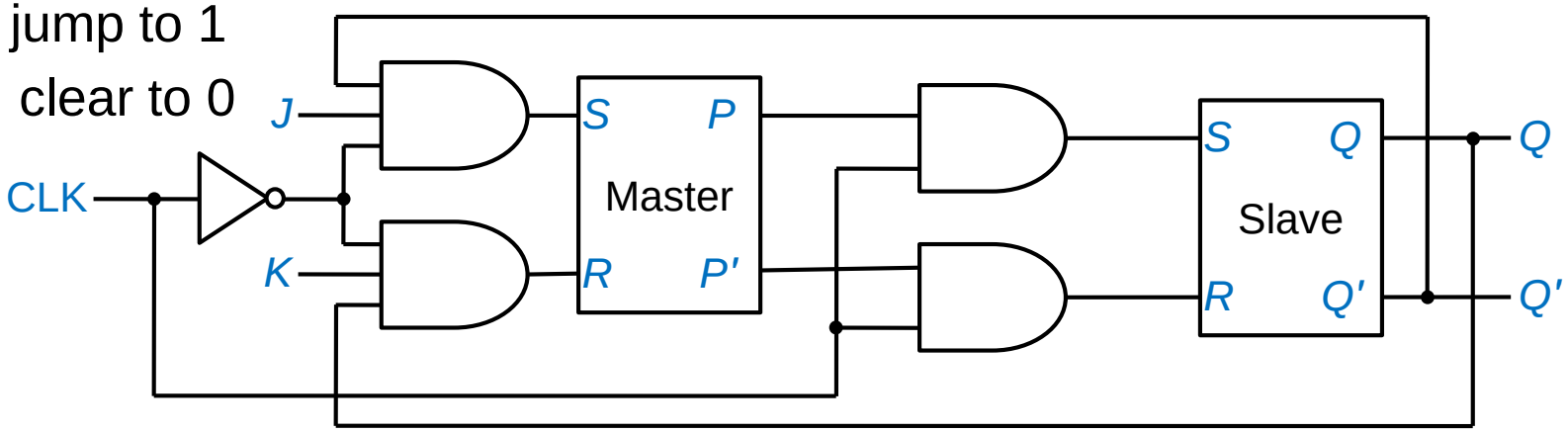
J-K Flip-Flop



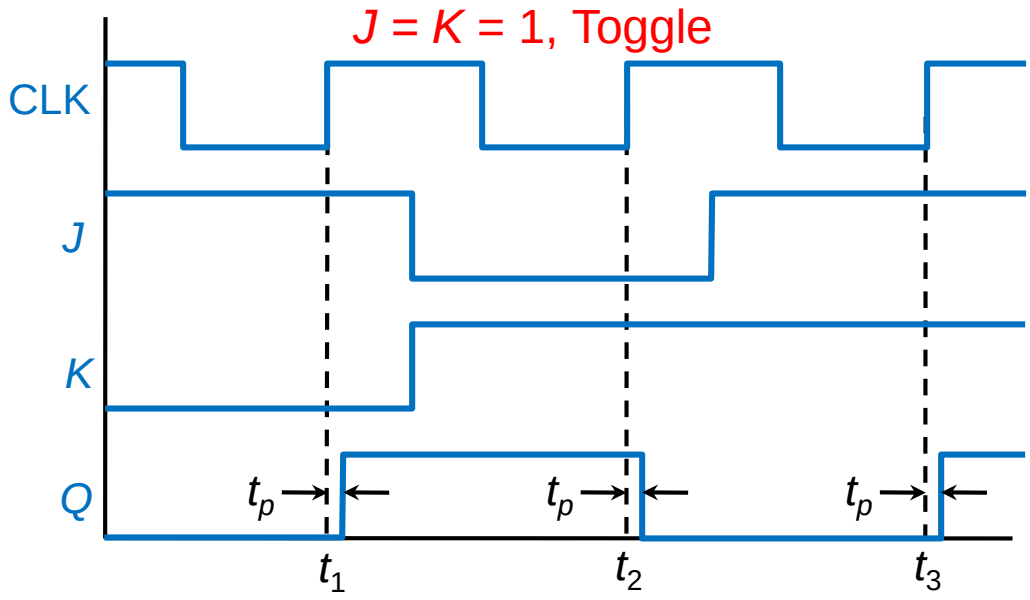
J-K Flip-Flop



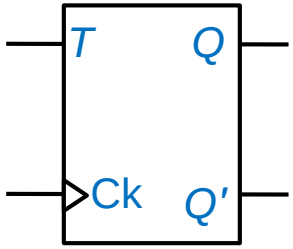
- **Extension of S-R FF**
  - *J*: jump to 1
  - *K*: clear to 0



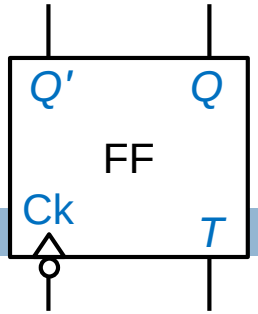
J	*K*	*Q*	Q^+	$Q^+ = JQ' + K'Q$
0	0	0	0	Unchanged
0	0	1	1	
0	1	0	0	Clear to 0
0	1	1	0	
1	0	0	1	Jump to 1
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	



T Flip-Flop

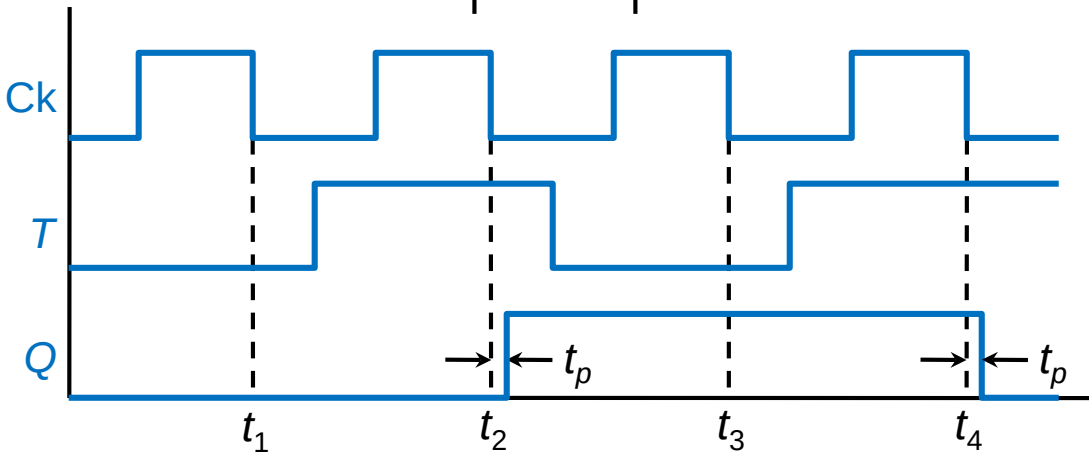


T Flip-Flop

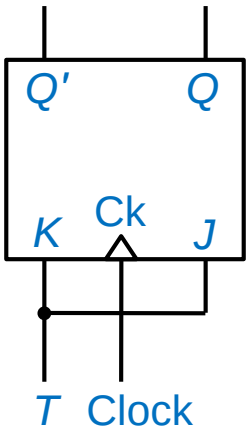


- **T: Toggle**

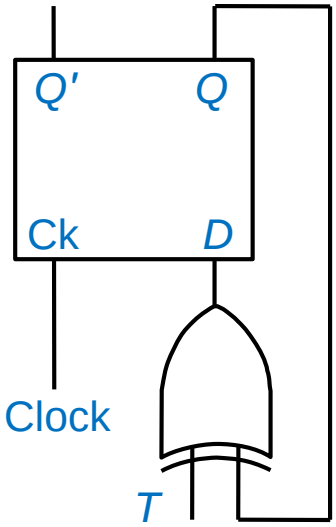
T	Q	$Q^+=T'Q+TQ'=T\oplus Q$	
0	0	0	Unchanged
0	1	1	
1	0	1	Toggle
1	1	0	



- **Implementation**



J-K FF based:
$J = K = 0$ Unchanged
$J = K = 1$ Toggle
$Q^+ = JQ' + K'Q$
$= TQ' + T'Q$

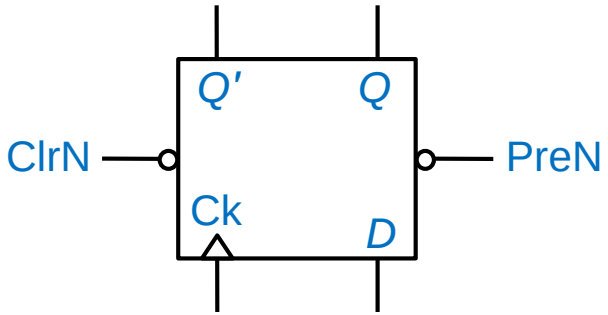


D FF based:
D input: $T\oplus Q$
$Q^+ = T\oplus Q$
$= TQ' + T'Q$

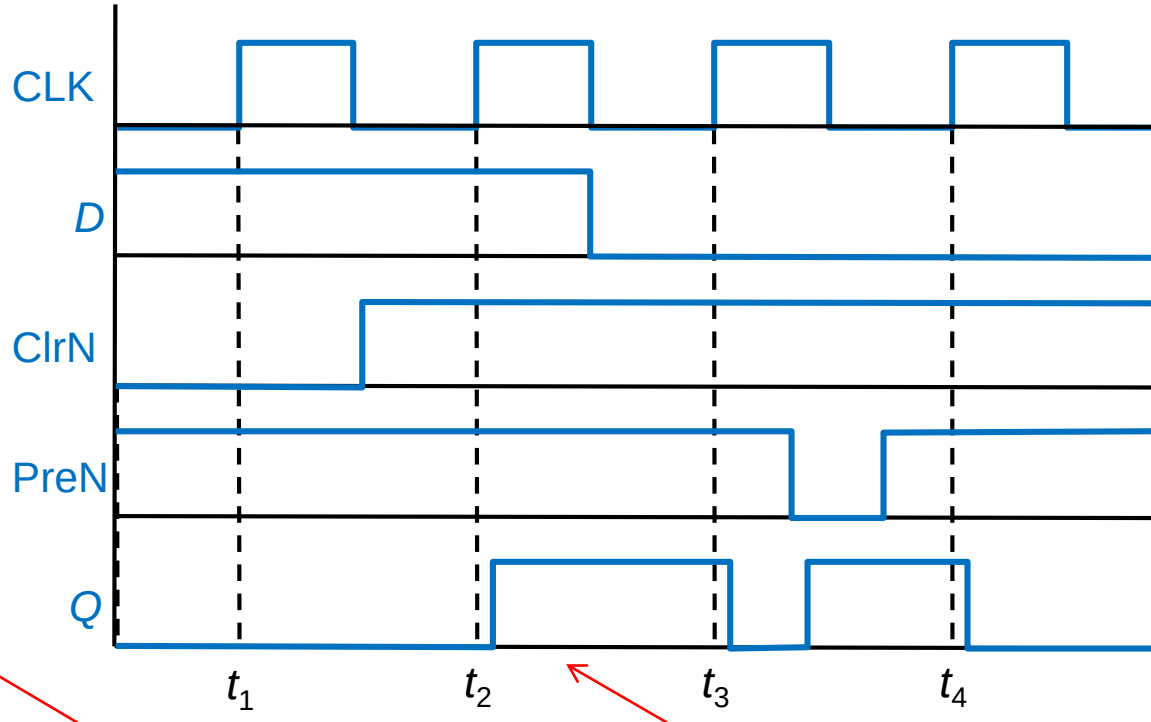
FFs with Additional Inputs

D FF with Clear and Preset

- **Use additional inputs to set FF to an initial state independent of the clock**
  - e.g., asynchronous Clear and Preset

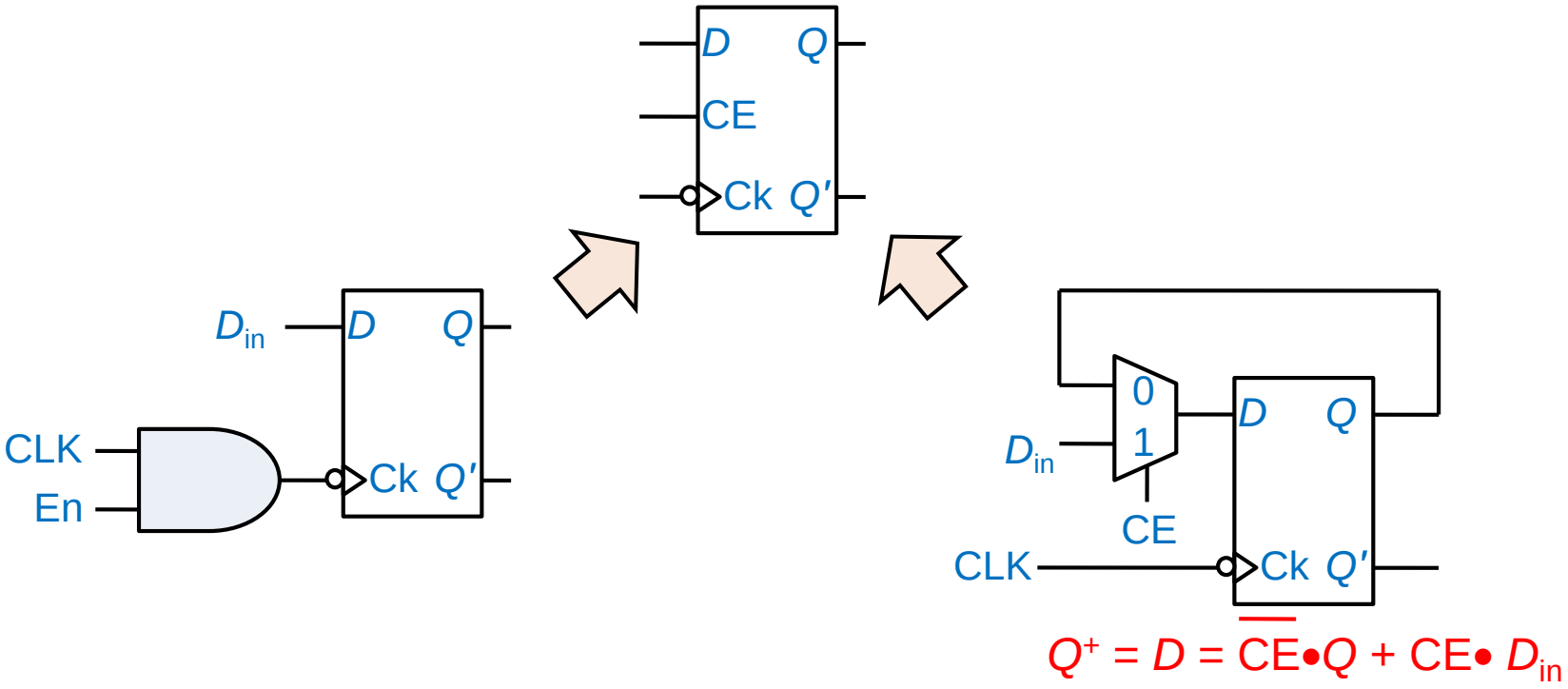


Ck	*D*	PreN	ClrN	Q^+
x	x	0	0	- (not allowed)
x	x	0	1	1
x	x	1	0	0
↑	0	1	1	0
↑	1	1	1	1
0,1,↓	x	1	1	*Q* (no change)



D FF with Clock Enable

- **We want some Flip-flops to hold existing data even though the data input may be changing**
  - Gated clock: gate the clock by clock enable (CE)



Summary

Characteristic Equations for Latches & FFs

Type	Q^+
S-R latch or FF	$Q^+ = S + R'Q \quad (SR = 0)$
Gated D latch	$Q^+ = G'Q + GD$
D FF	$Q^+ = D$
D-CE FF	$Q^+ = D \bullet CE + Q \bullet CE'$
J-K FF	$Q^+ = JQ' + K'Q$
T FF	$Q^+ = T \oplus Q$