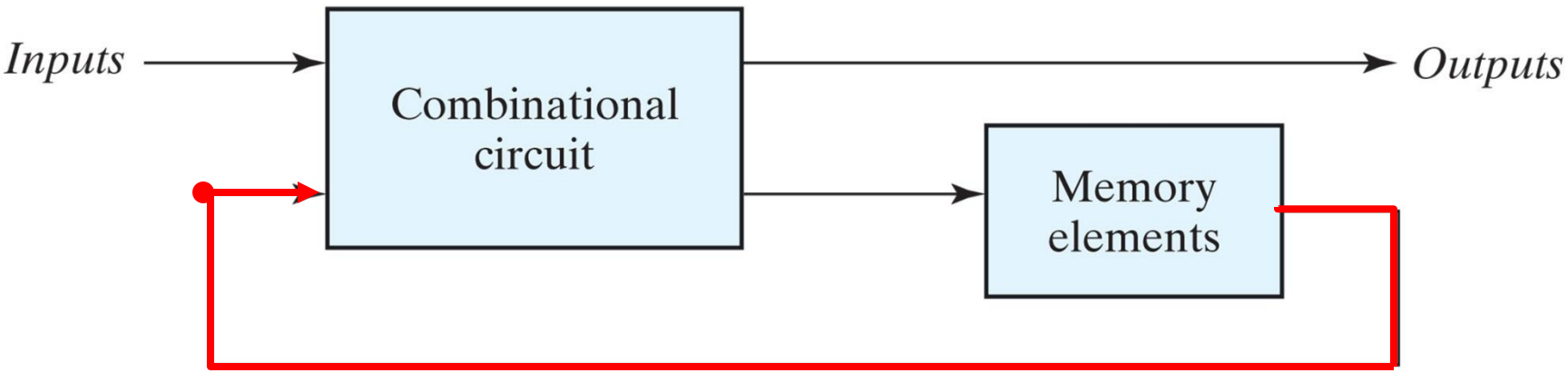
Chapter 5

Sequential Circuits

- Combinational vs Sequential
- Its all about feedback
- What is feedback?
  - The output of a circuit acts as an input to the circuit

Feedback



- Combinational
  - No feedback
  - Output depends solely on the inputs
- Sequential
  - Feedback
  - Current state depends on previous state
- State
  - Output at a given time
  - State is tied to a specific point in time

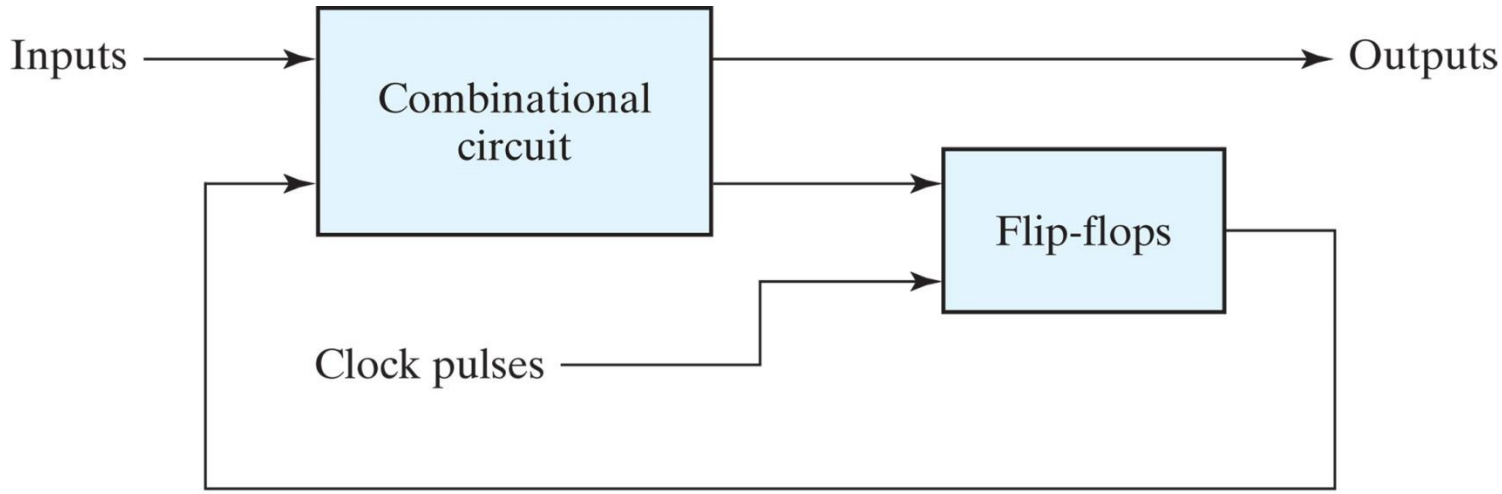
Sequential Circuits

- Feedback allows for the creation of a storage element.
  - Memory
  - Store, retain, and retrieve information

- Thus a sequential circuit is specified by a time sequence of inputs, outputs, and internal states.
- Two main types of sequential circuits
  - Synchronous
  - Asynchronous

- Synchronous
  - Behavior of the circuit is defined by the knowledge of its signals at a discrete points in time
- Asynchronous
  - Behavior depends on the inputs at ANY instant in time AND the order in which the inputs change

- Asynchronous
  - Can be thought of as a combinational circuit with feedback
  - Can be become unstable due to feedback among gates
- Synchronous
  - Uses a clock generator or clock
  - Determines when computational activity will occur within the circuit.



(a) Block diagram

(b) Timing diagram of clock pulses

Storage Elements

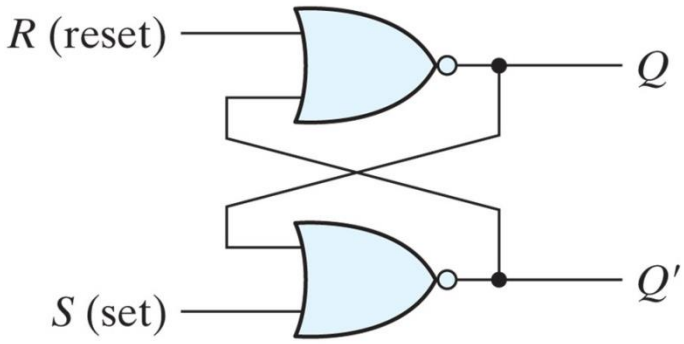
- Latches vs Flip-flops
  - Latch - A change in signal level changes output
  - Flip-flop – Controlled by the transition from high to low or low to high NOT the level.

Latches

- Useful for storing binary information and for the design of asynchronous circuits
- Not practical for use as storage elements in sequential circuits

SR Latch

- SR – Set/Reset



(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$)
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$)
1	1	0	0	(forbidden)

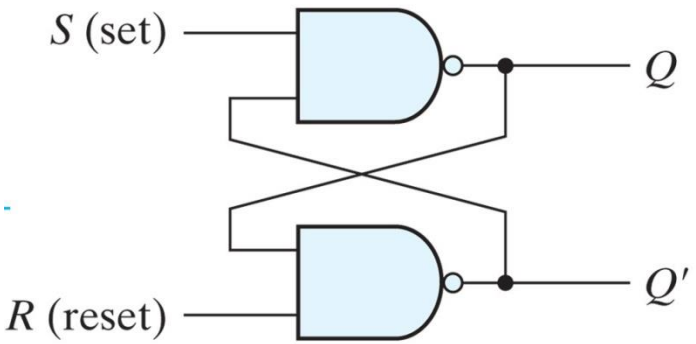
(b) Function table

Q = 1 & Q'=0
Latch is set

Note: 00 results in no change on the output.

Q = 0 & Q'=1
Latch is reset

- Can also build with NAND Gates

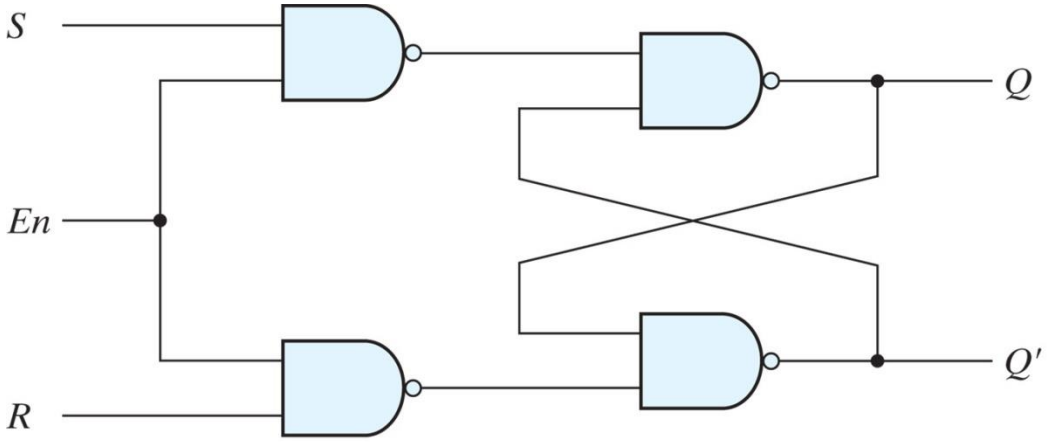


(a) Logic diagram

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	(after $S = 1, R = 0$)
0	1	1	0	
1	1	1	0	(after $S = 0, R = 1$)
0	0	1	1	(forbidden)

(b) Function table

SR-Latch with Enable



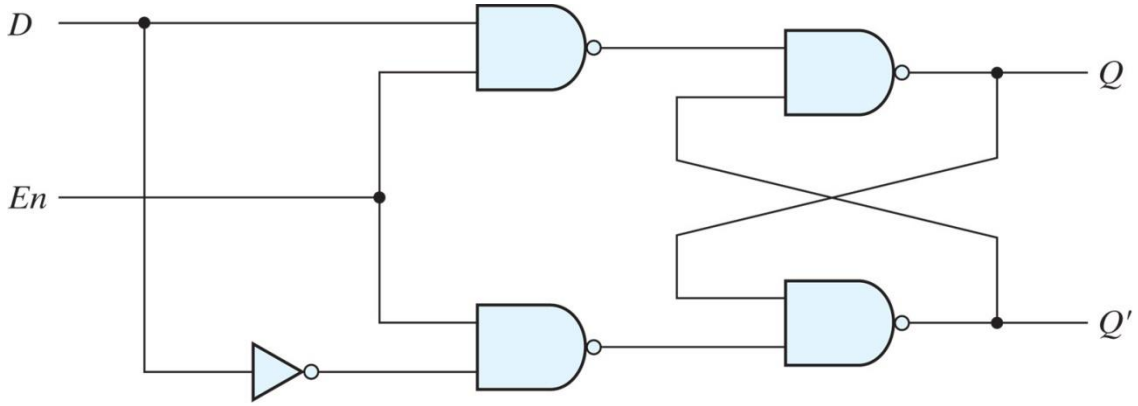
(a) Logic diagram

En	*S*	*R*	Next state of *Q*
0	X	X	No change
1	0	0	No change
1	0	1	$Q = 0$; reset state
1	1	0	$Q = 1$; set state
1	1	1	Indeterminate

(b) Function table

D-Latch

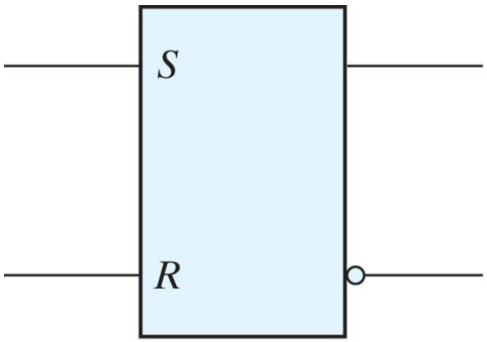
- Eliminates the indeterminate condition of the SR-Latch



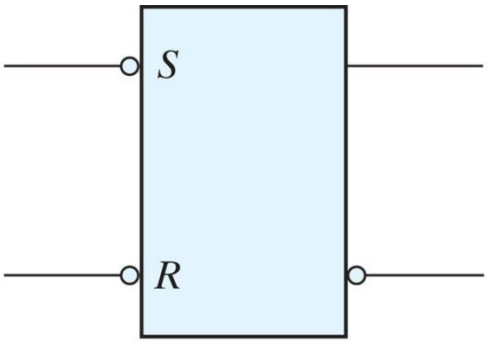
(a) Logic diagram

En	*D*	Next state of *Q*
0	X	No change
1	0	$Q = 0$; reset state
1	1	$Q = 1$; set state

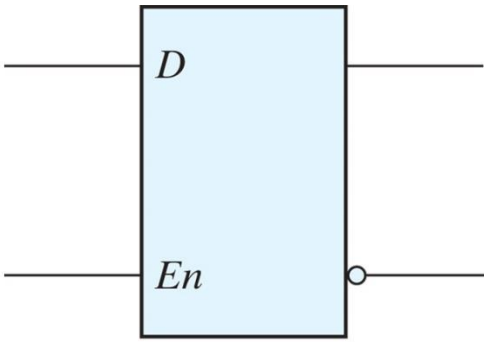
(b) Function table



SR



$\overline{S}\overline{R}$

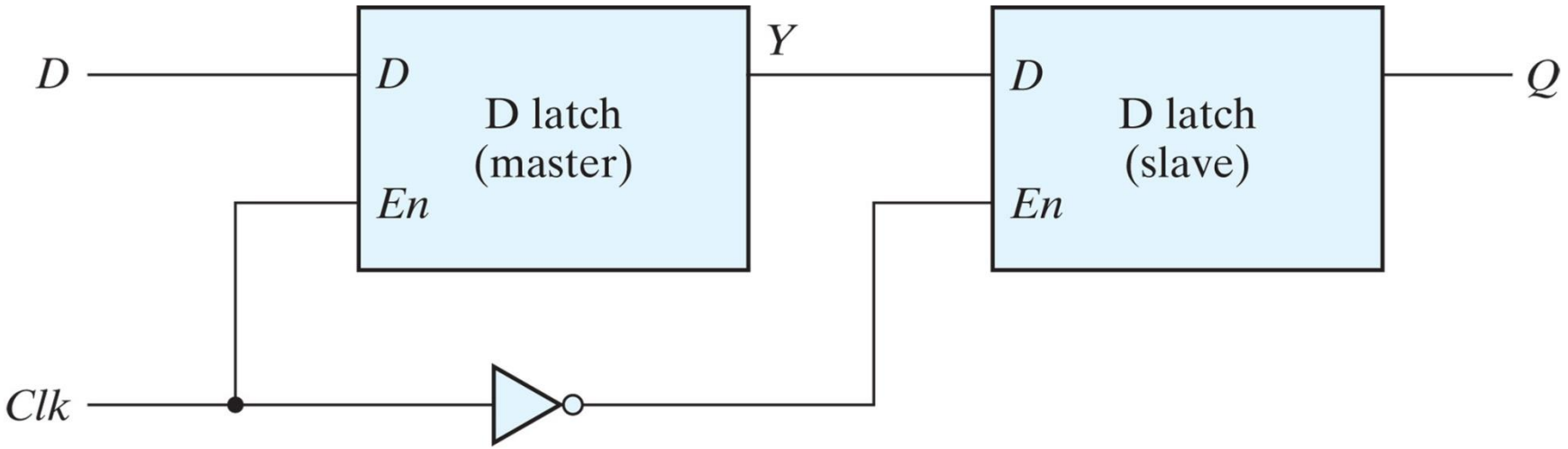


D

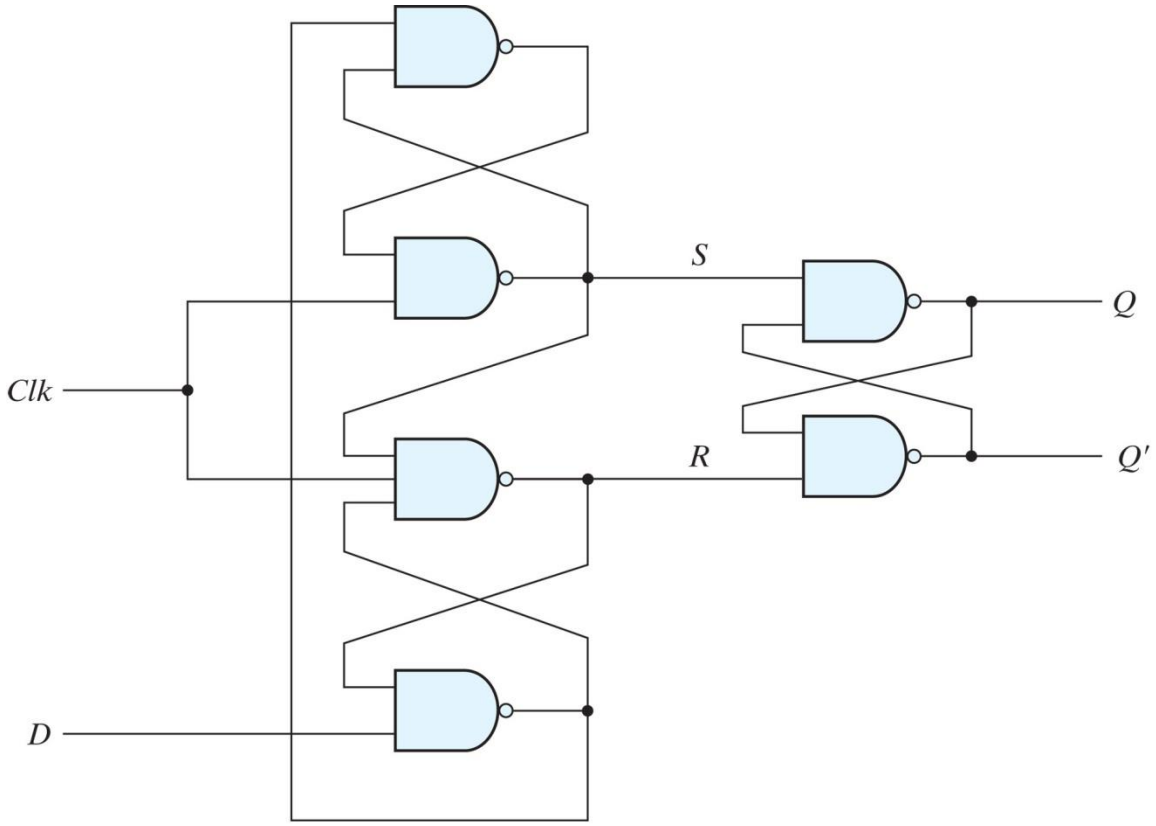
Flip-flops

- A binary storage device capable of storing 1 bit of binary information.
- Used in clocked sequential circuits
- State is changed by a change in the control input
  - Clock transition
  - A trigger

Edge-Triggered D Flip-flop



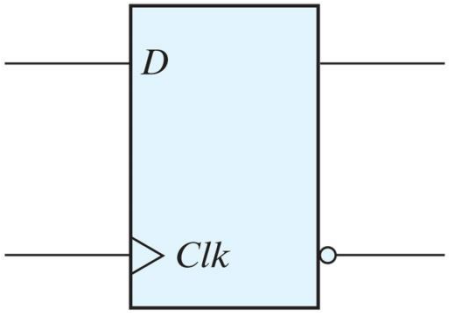
D-type Positive Edge Triggered F/F



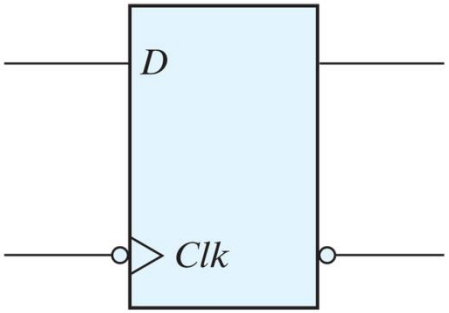
D-Flip-flop

D Flip-Flop

D	Q(t + 1)
0	
1	

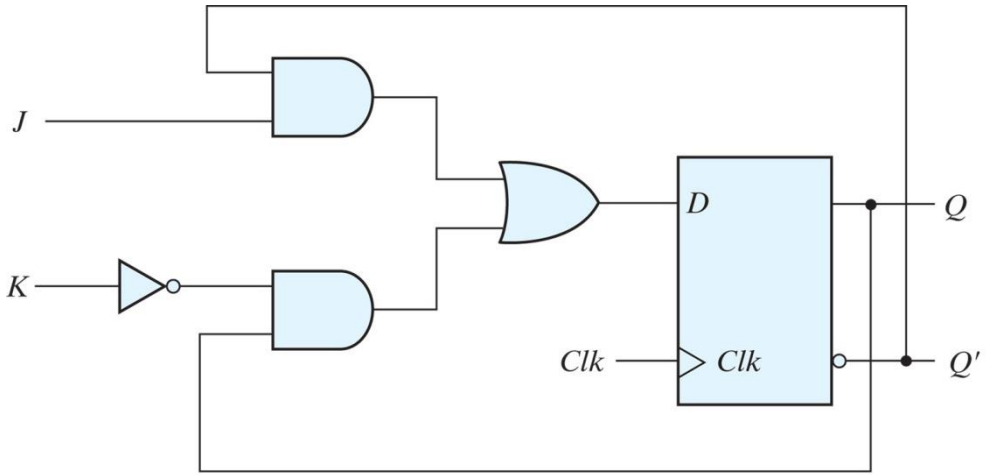


(a) Positive-edge



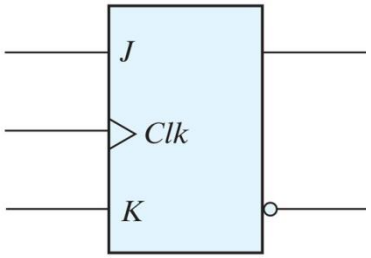
(a) Negative-edge

JK Flip-Flop

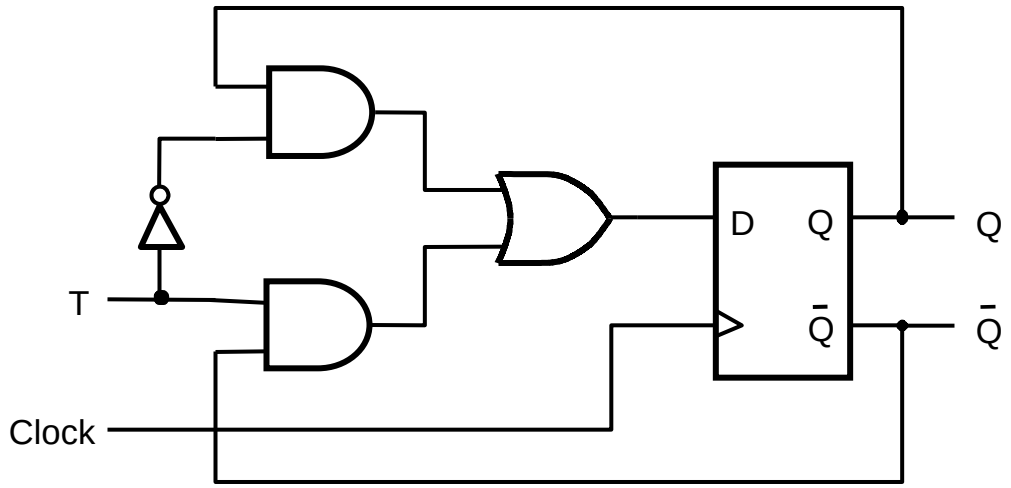


JK Flip-Flop

J	K	Q(t + 1)
0	0	
0	1	
1	0	
1	1	

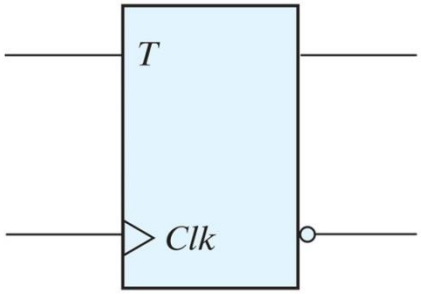


Toggle Flip-Flop



T Flip-Flop

T	Q(t + 1)
0	
1	



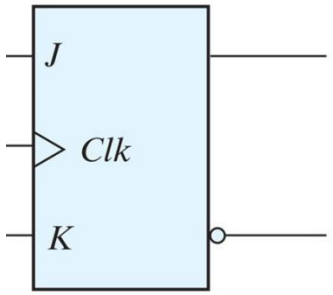
Make a T from a JK

JK Flip-Flop

J	K	Q(t + 1)
0	0	$Q(t)$
0	1	0
1	0	1
1	1	$Q'(t)$

T Flip-Flop

T	Q(t + 1)
0	$Q(t)$
1	$Q'(t)$



From a D Flip-flop

D Flip-Flop

D	Q(t + 1)
0	0
1	1

T Flip-Flop

T	Q(t + 1)
0	$Q(t)$
1	$Q'(t)$