Flip-Flops, Registers, Counters, and a Simple Processor

Chapter7

Chapter Objectives

- Logic circuits that can store information
- Flip-flops, which store a single bit
- Registers, which store multiple bits
- Shift registers, which shift the contents of the register
- Counters of various types

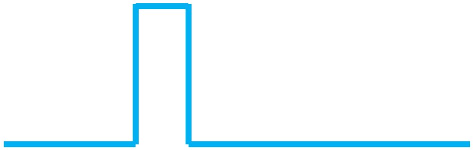
Contents

1. Basic Latch
2. Gated SR Latch
3. Gated D Latch
4. Master-Slave and Edge-Triggered D Flip-Flops
5. T Flip-Flop
6. JK Flip-Flop
7. Summary of Terminology

Contents

8. Registers
9. Counters
10. Reset Synchronization
11. Other types of Counters

Need for Memory



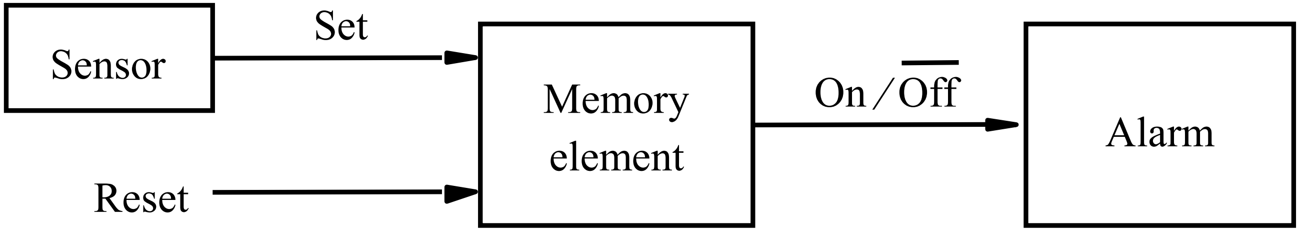


Figure 7.1. Control of an alarm system.

The circuit requires a memory element to remember that the alarm has to be active until the Reset signal arrives.

Simple Memory Element

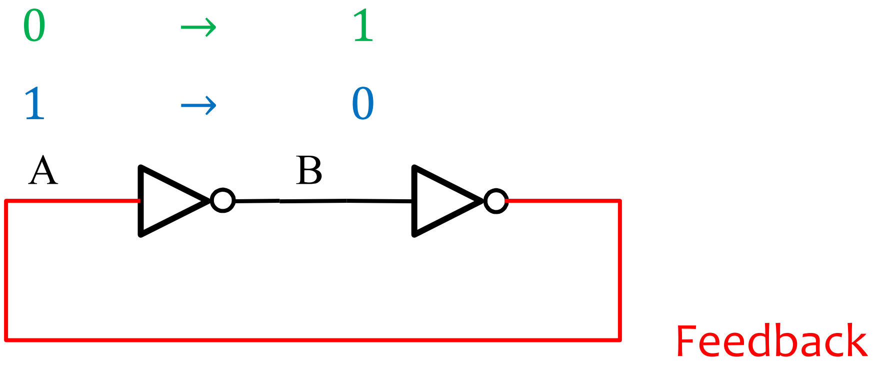


Figure 7.2. A simple memory element.

This circuit is not useful, because it lacks some practical means for changing its state.

A controlled memory element

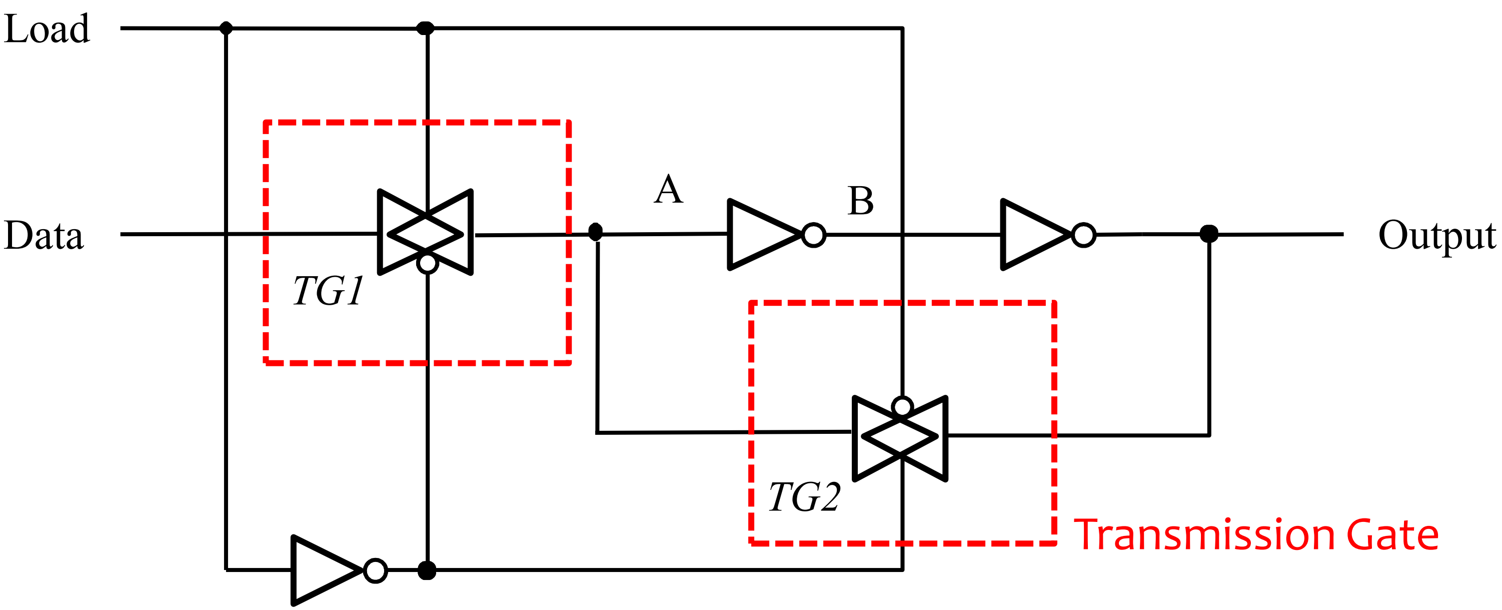
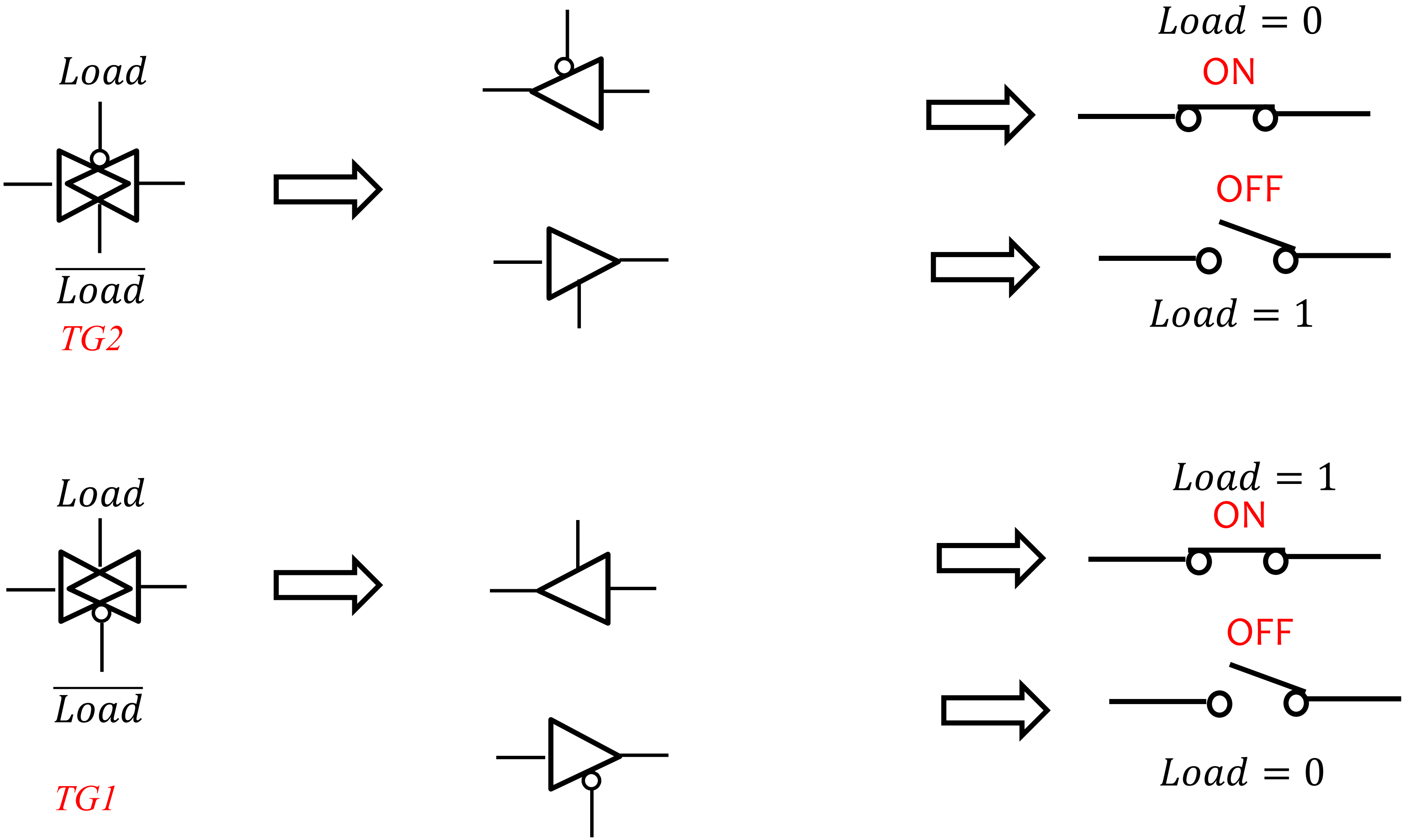


Figure 7.3. A controlled memory element.

A controlled memory element



$Load$

$\overline{Load}$

TG2

$Load = 0$

ON

OFF

$Load = 1$

$Load$

$\overline{Load}$

TG1

$Load = 1$

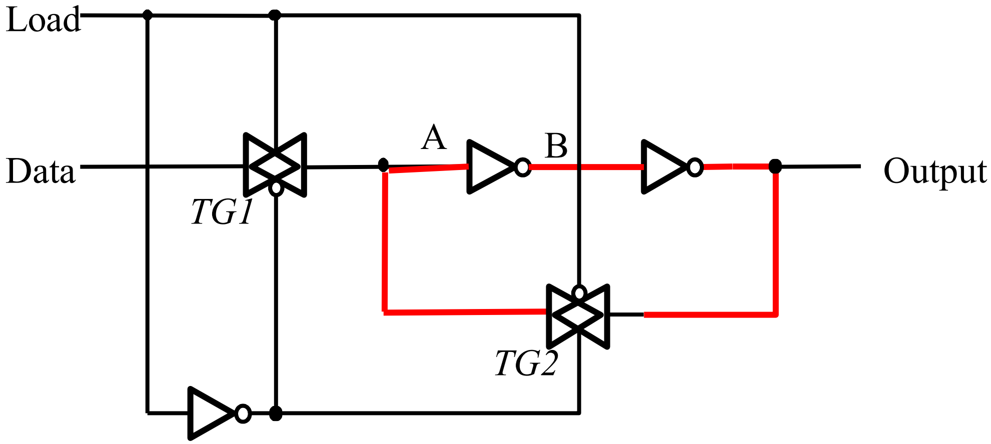
ON

OFF

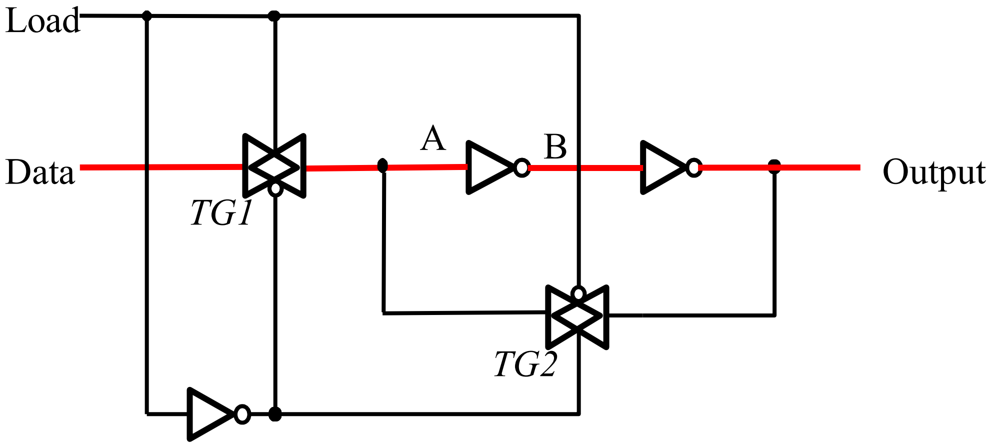
$Load = 0$

A controlled memory element

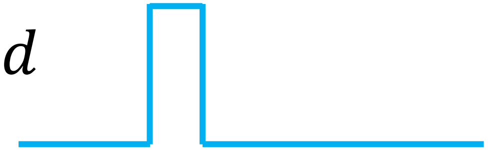
$Load = 0$



$Load = 1$



$Load$



BASIC LATCH

A memory element with NOR gates

NOR gate

x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Set	Reset	Q
0	0	0/1
0	1	0
1	0	1
1	1	0

Truth table

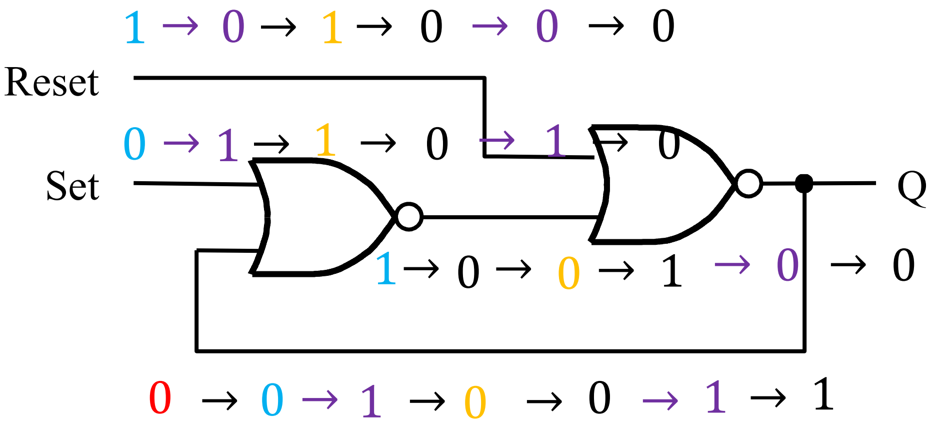
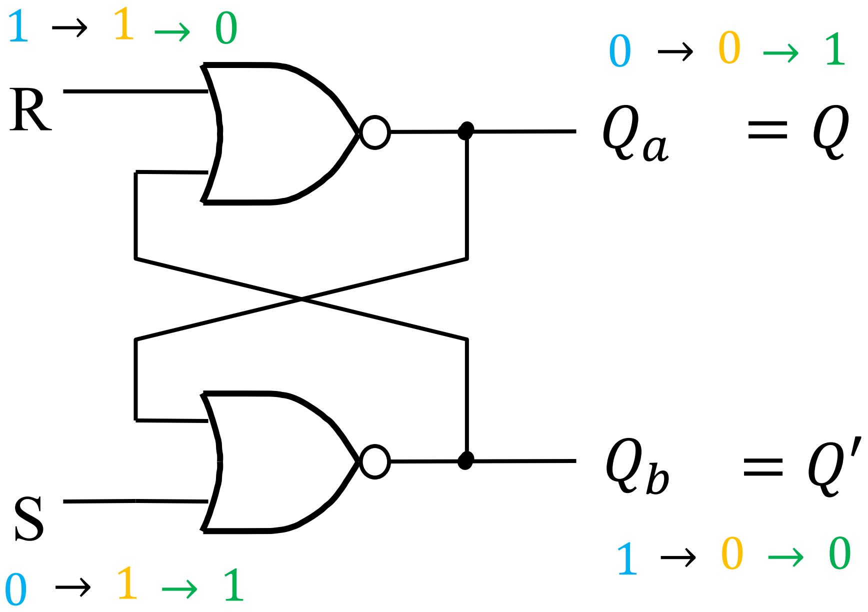


Figure 7.4. A memory element with NOR gates.

A latch built with NOR gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

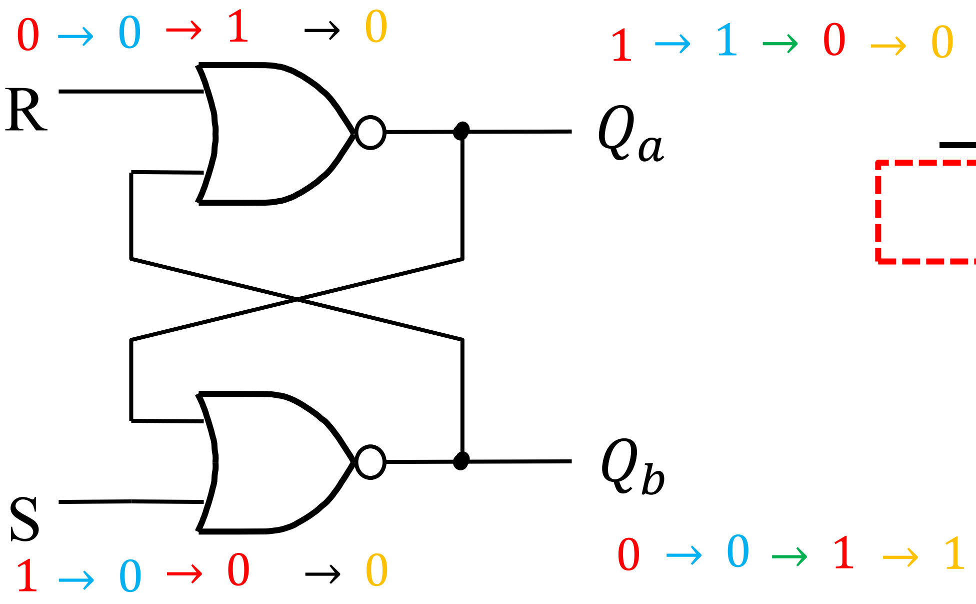
(b) Truth table

NOR gate

x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Figure 7.5. A latch built with NOR gates.

A latch built with NOR gates



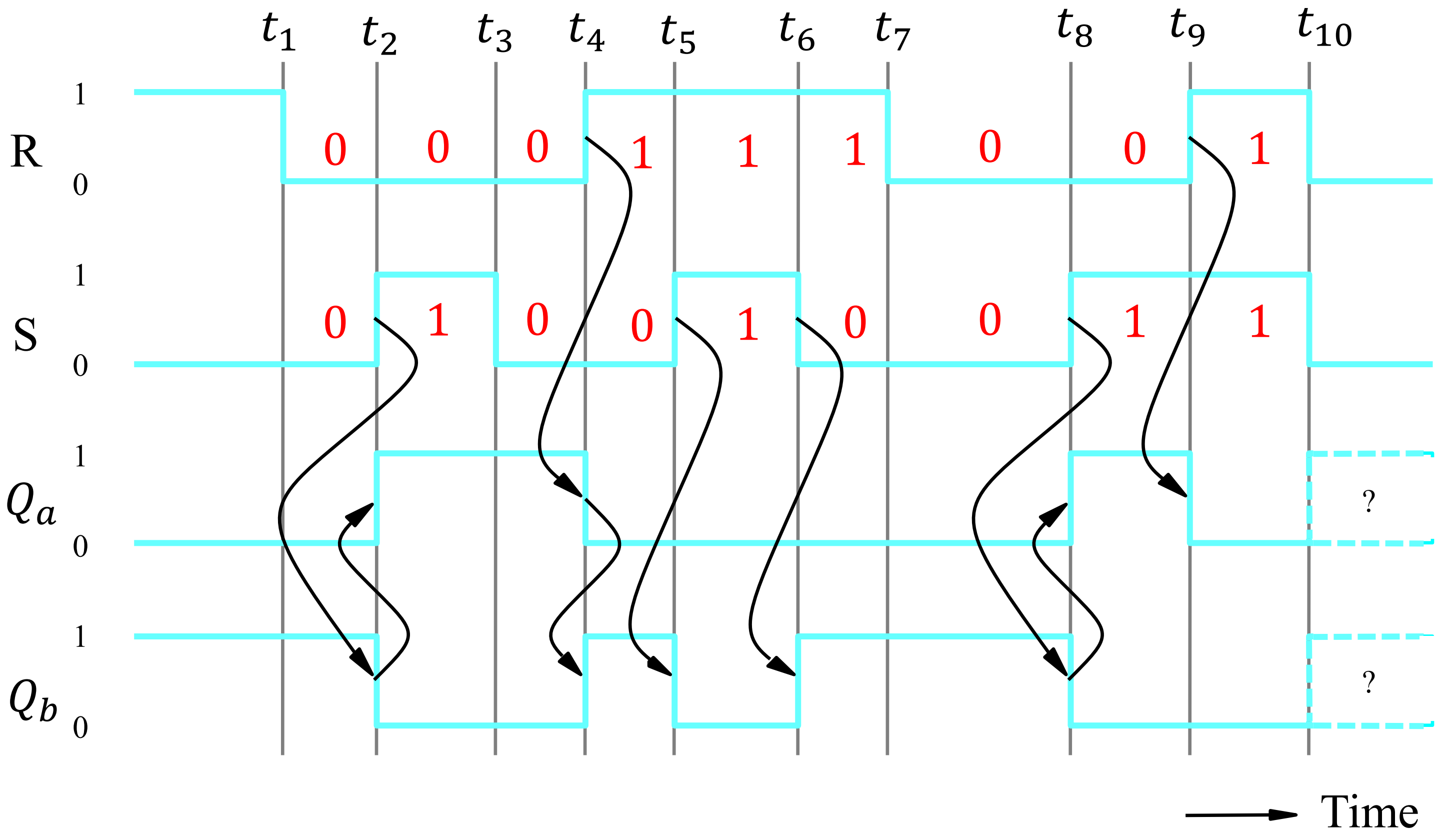
(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Truth table

Figure 7.5. A latch built with NOR gates.

A latch built with NOR gates

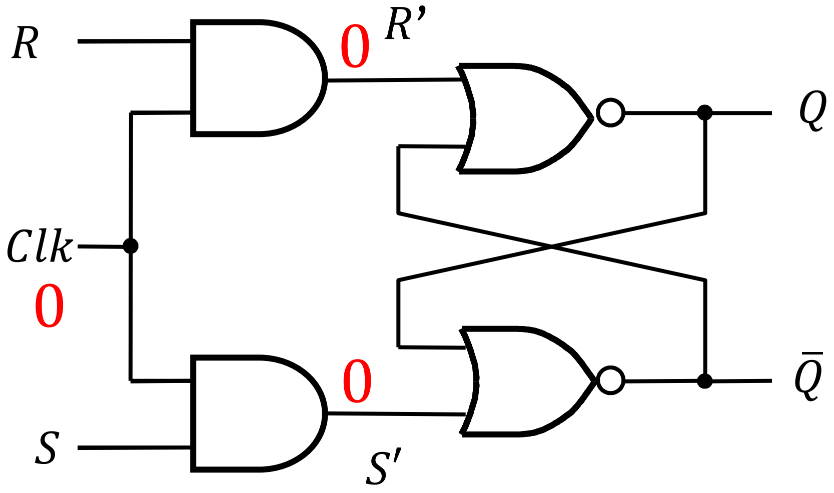


(c) Timing diagram

Figure 7.5. A latch built with NOR gates.

GATED SR LATCH

Gated SR latch



(a) Circuit

Clk	S	R	$Q(t+1)$
0	×	×	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	×

(b) Truth table

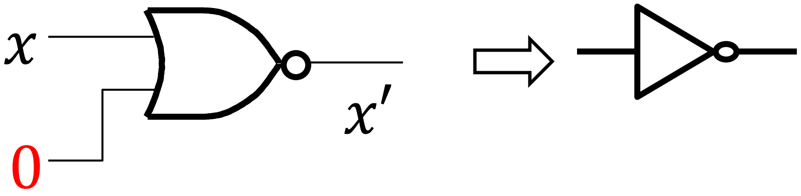
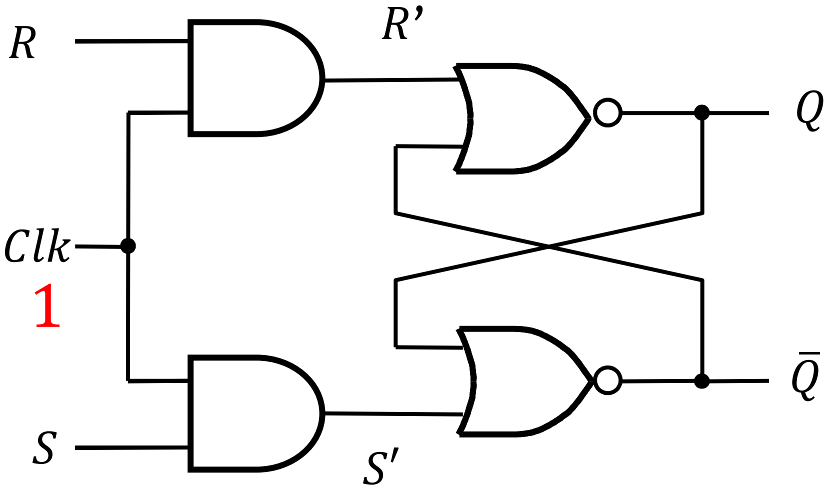


Figure 7.6. Gated SR latch.

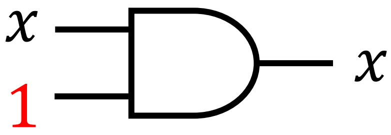
Gated SR latch



(a) Circuit

Clk	S	R	$Q(t+1)$
0	×	×	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	× (inhibited)

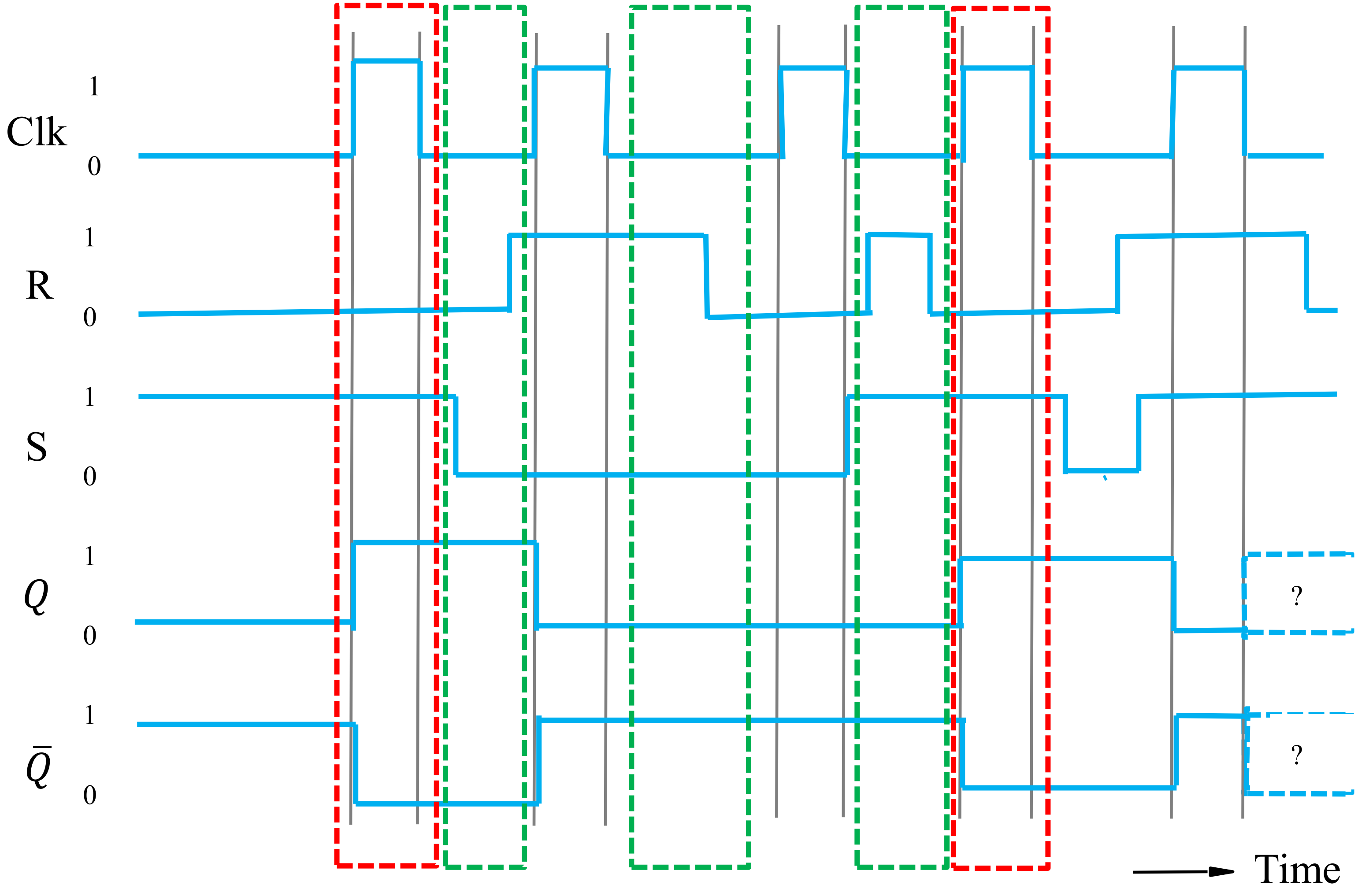
(b) Truth table



$R' = R, S' = S$

Figure 7.6. Gated SR latch.

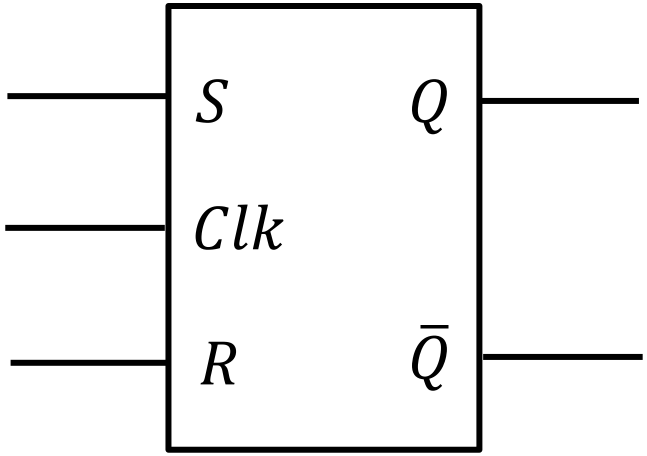
Gated SR latch



(c) Timing diagram

Figure 7.6. Gated SR latch.

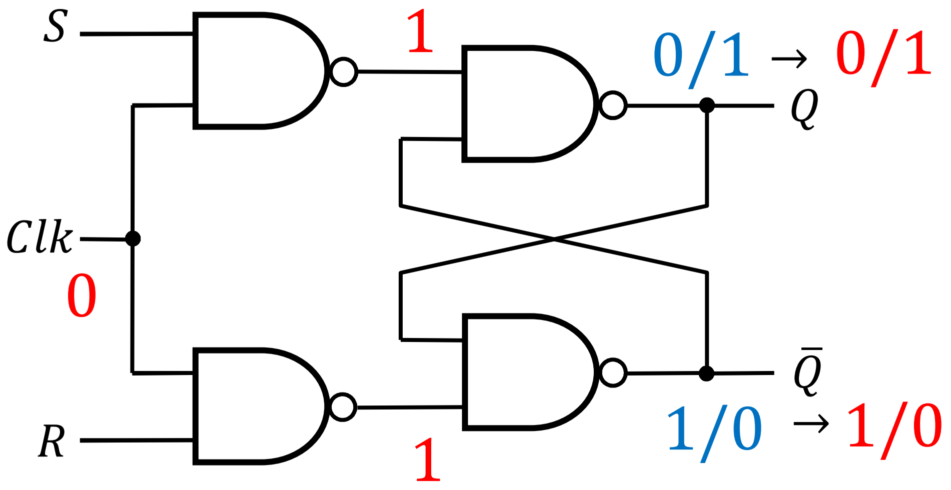
Gated SR latch



(d) Graphical symbol

Figure 7.6. Gated SR latch.

Gated SR latch with NAND gates



Clk	S	R	$Q(t+1)$
0	×	×	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	×

Truth table

NAND gate

x	y	$(xy)'$
0	0	1
0	1	1
1	0	1
1	1	0

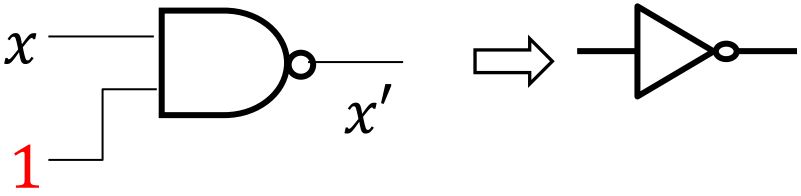
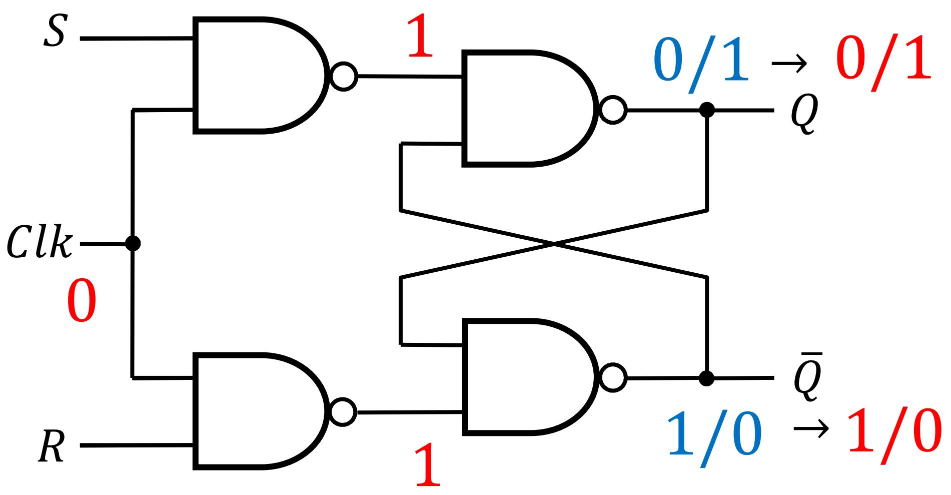


Figure 7.7. Gated SR latch with NAND gates.

Gated SR latch with NAND gates



Clk	S	R	$Q(t+1)$
0	×	×	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	×

Truth table

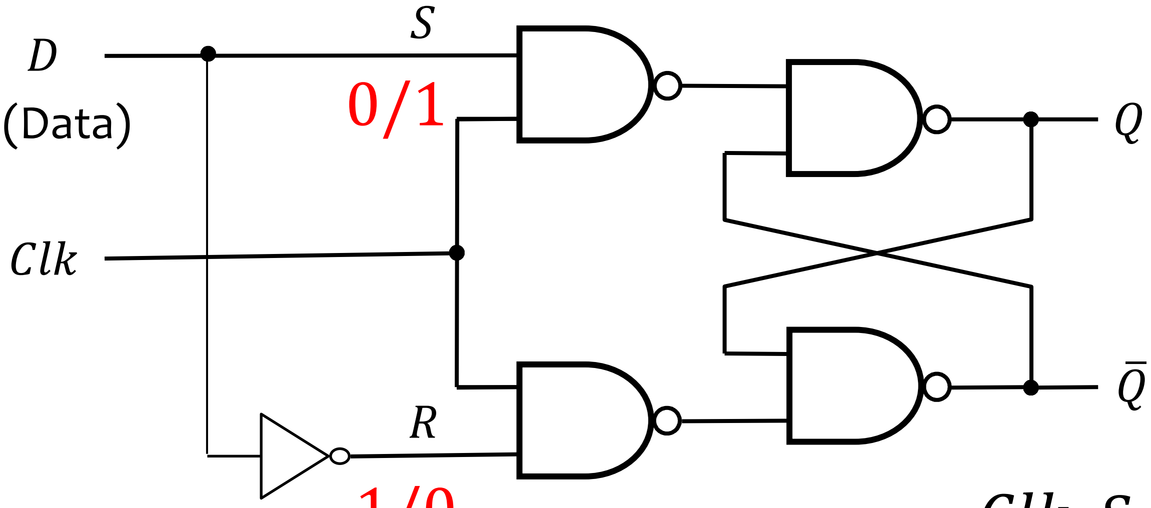
NAND gate

x	y	$(xy)'$
0	0	1
0	1	1
1	0	1
1	1	0

Figure 7.7. Gated SR latch with NAND gates.

GATED D LATCH

Gated D latch



(a) Circuit

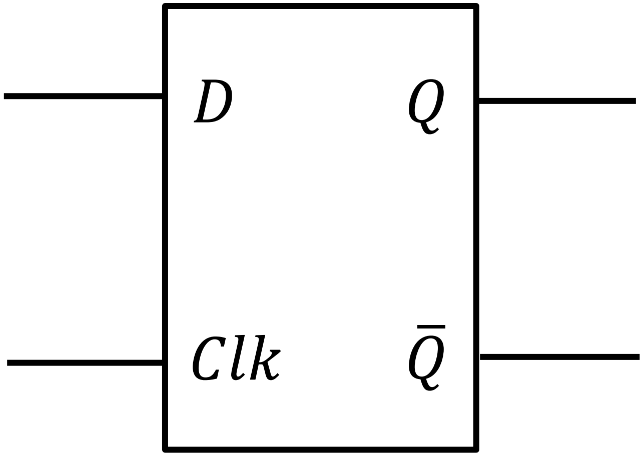
Clk	S	R	$Q(t+1)$
0	×	×	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	×

Clk	D	$Q(t+1)$
0	×	$Q(t)$ (no change)
1	0	0
1	1	1

(b) Characteristic table

Figure 7.8. Gated D latch.

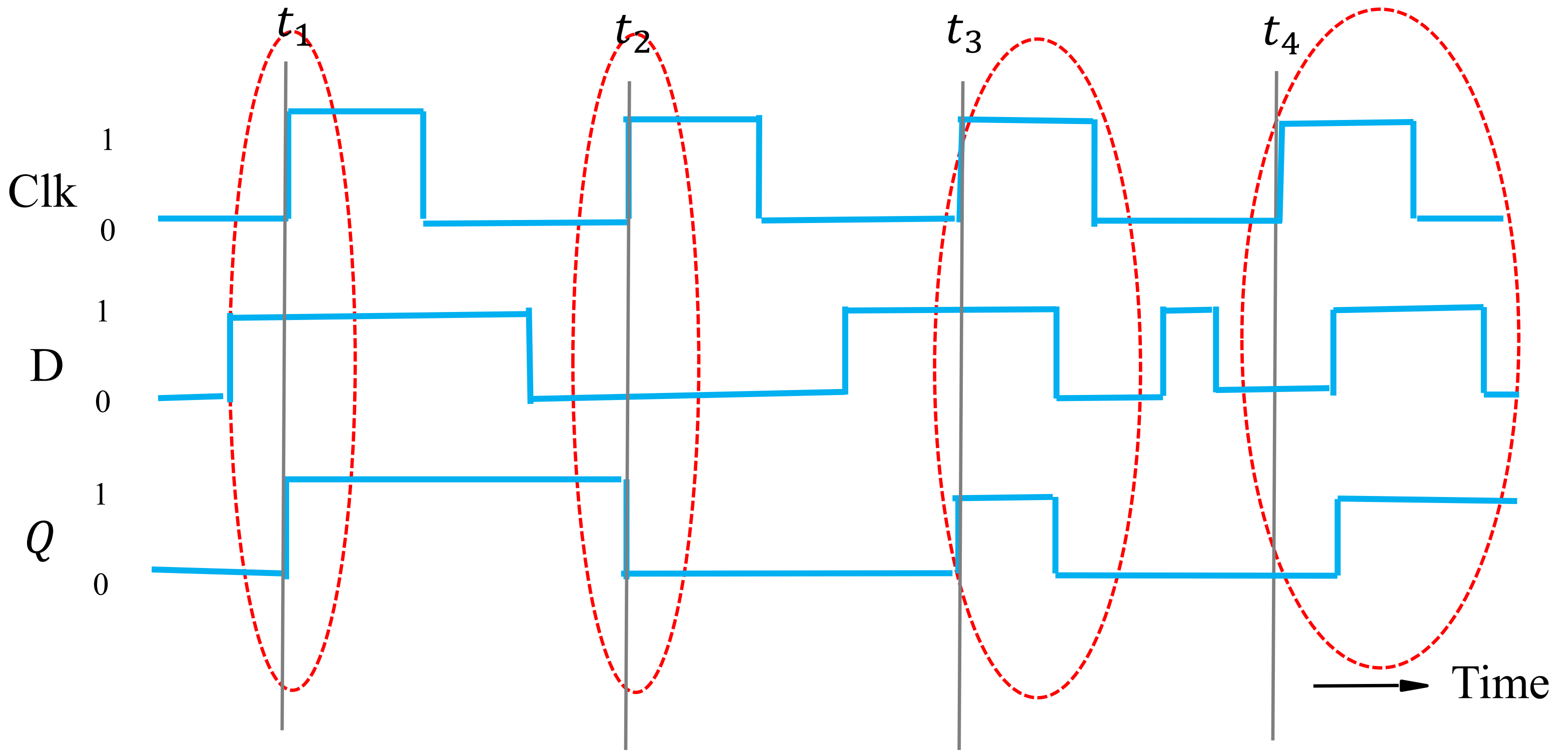
Gated D latch



(d) Graphical symbol

Figure 7.8. Gated D latch.

Gated D latch

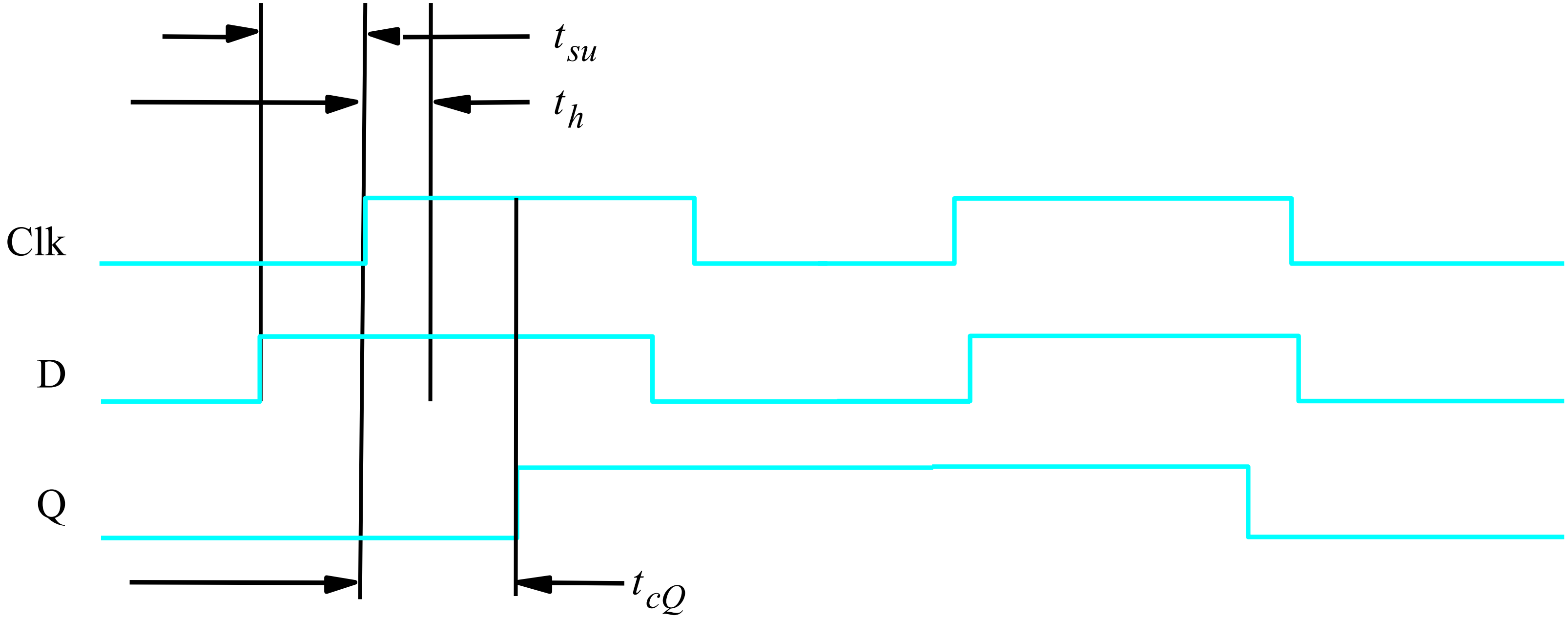


(c) Timing diagram

Figure 7.8. Gated D latch.

Setup and hold times

The minimum time that the D signal must be stable prior to the negative edge of the clock signal is called the setup time t_{su} of the latch.

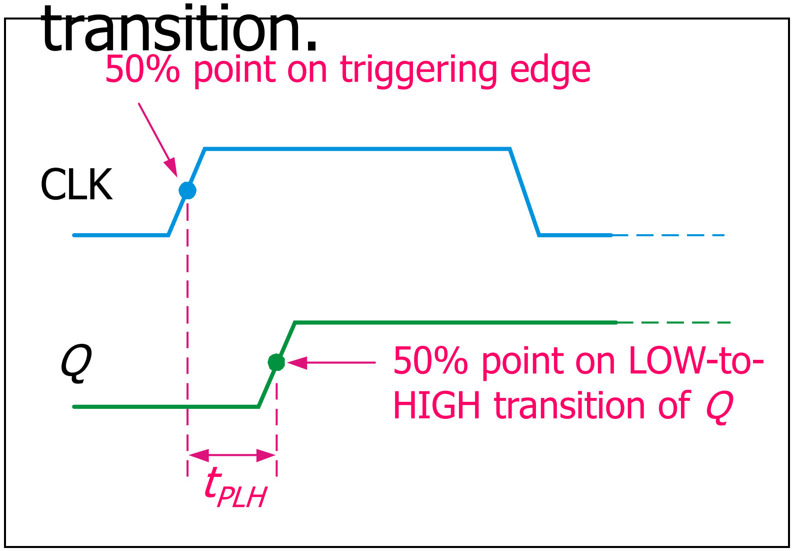


The minimum time that the D signal must remain stable after the negative edge of the clock signal is called the hold time t_h of the latch.

Figure 7.9. Setup and hold times.

Flip-flop Characteristics

Propagation delay time is specified for the rising and falling outputs. It is measured between the 50% level of the clock to the 50% level of the output transition.

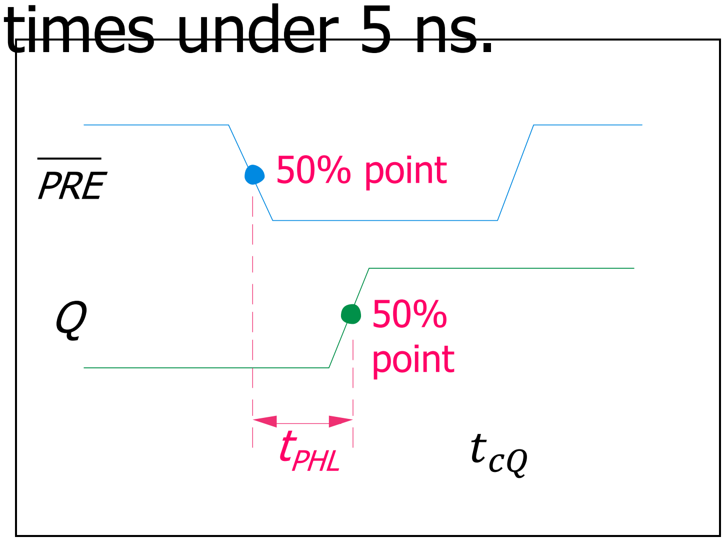

50% point on triggering edge
CLK
Q
50% point on LOW-to-HIGH transition of *Q*
t_{PLH}

The typical propagation delay time for the 74AHC family (CMOS) is 4 ns. Even faster logic is available for specialized applications.

Summar

Flip-flop Characteristics

Another **propagation delay time** specification is the time required for an *asynchronous* input to cause a change in the output. Again it is measured from the 50% levels. The 74AHC family has specified delay times under 5 ns.

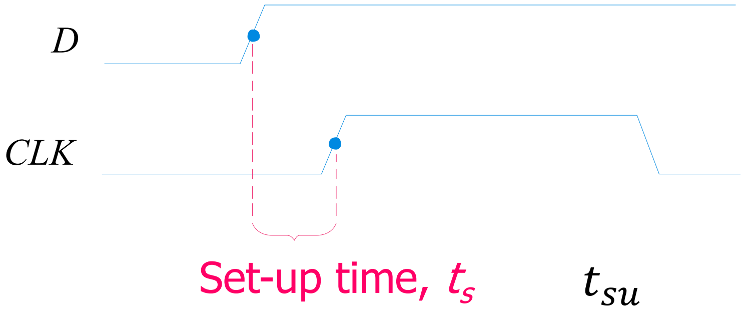


Summar

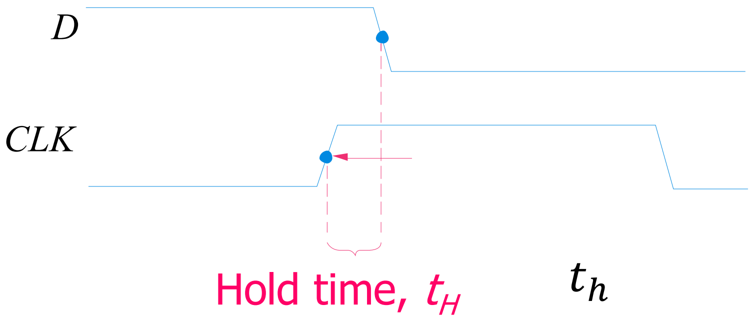
Flip-flop Characteristics

Set-up time and **hold time** are times required before and after the clock transition that data must be present to be reliably clocked into the flip-flop.

Setup time is the minimum time for the data to be present *before* the clock.

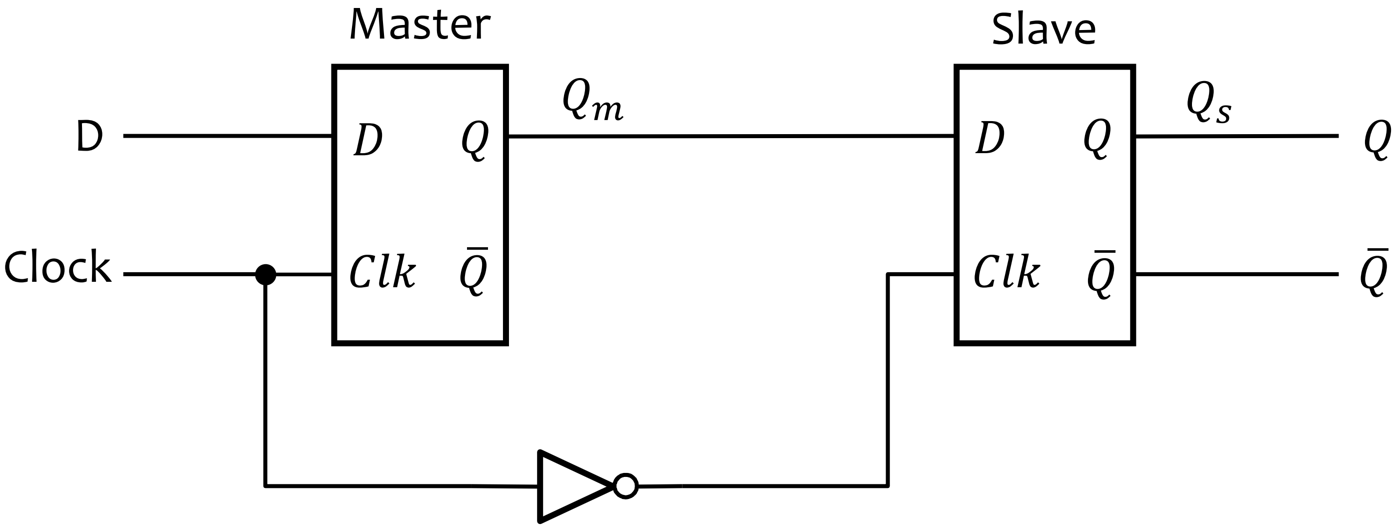


Hold time is the minimum time for the data to *remain* after the clock.



MASTER-SLAVE AND EDGE-TRIGGERED D FLIP-FLOPS

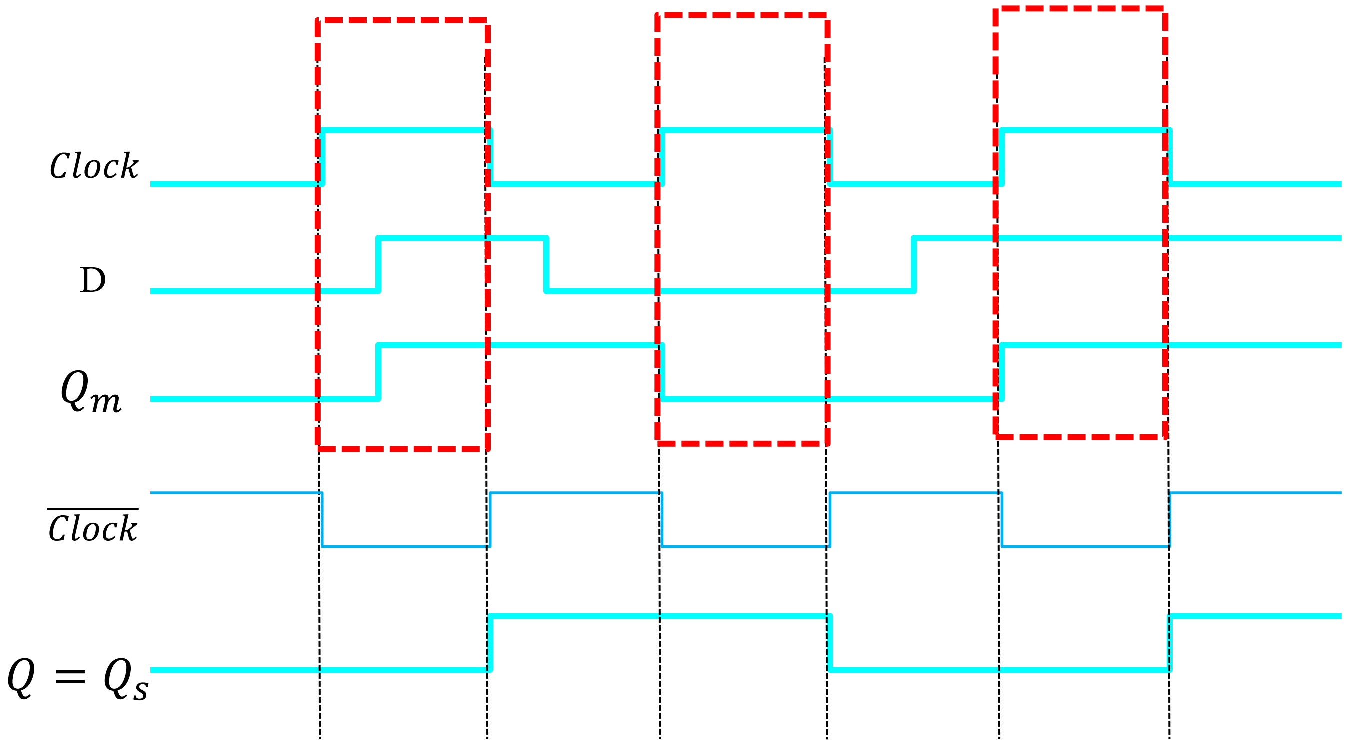
Master-slave D flip-flop



(a) Circuit

Figure 7.10. Master-slave D flip-flop.

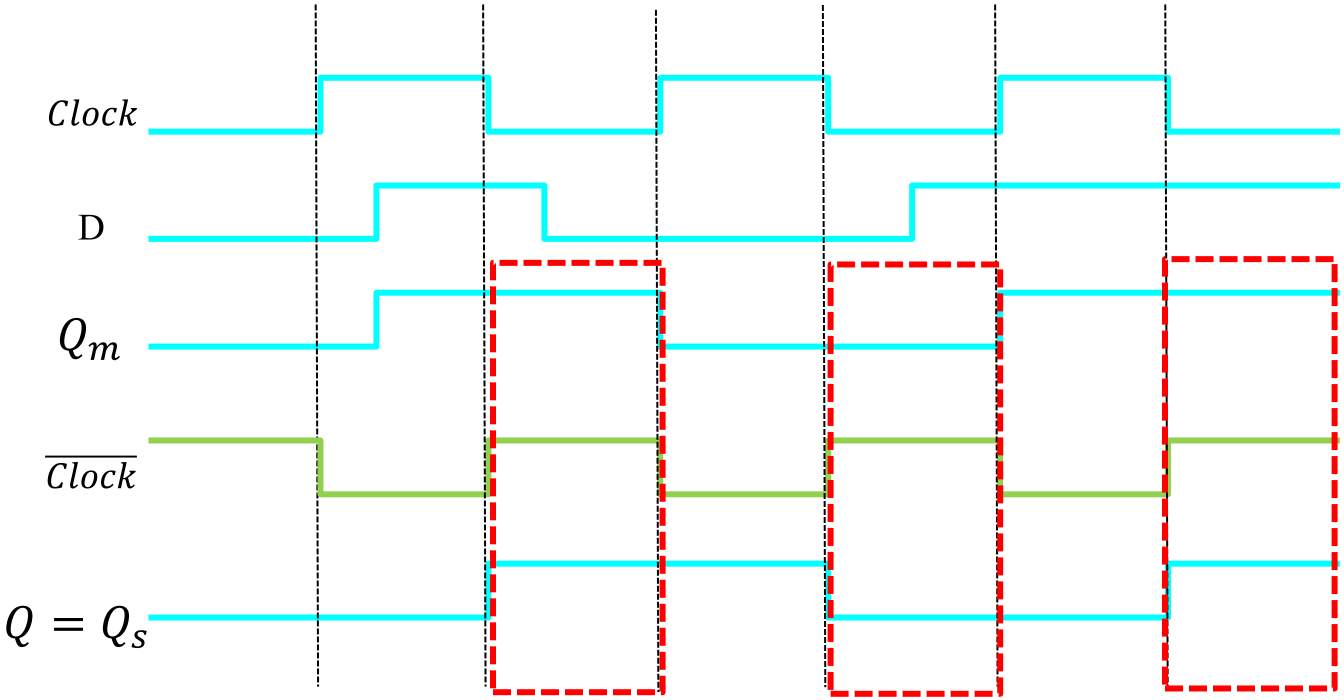
Master-slave D flip-flop



(b) Timing diagram

Figure 7.10. Master-slave D flip-flop.

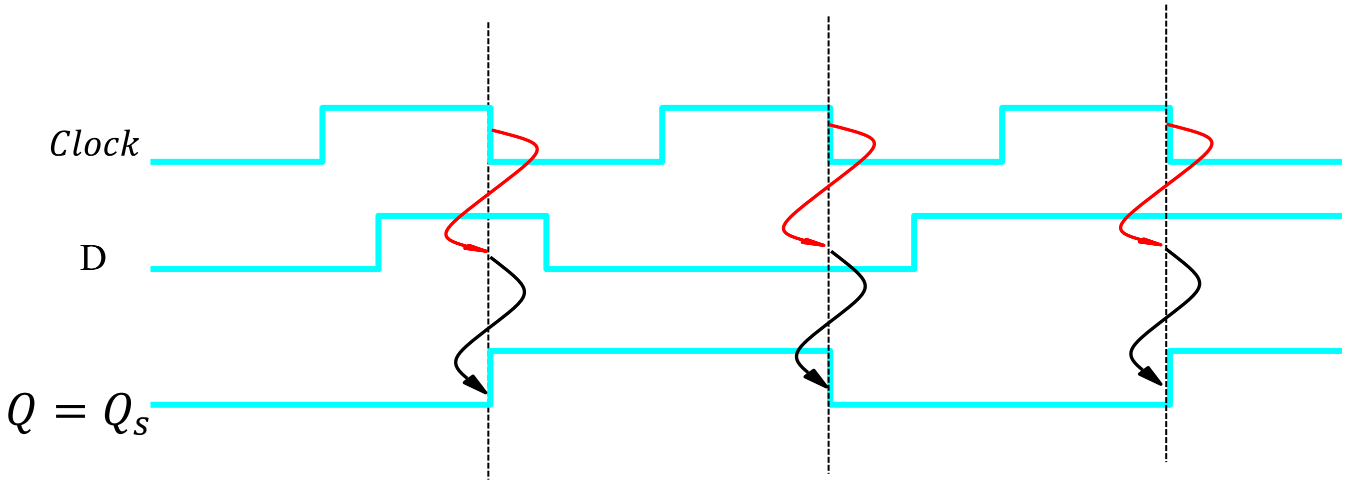
Master-slave D flip-flop



(b) Timing diagram

Figure 7.10. Master-slave D flip-flop.

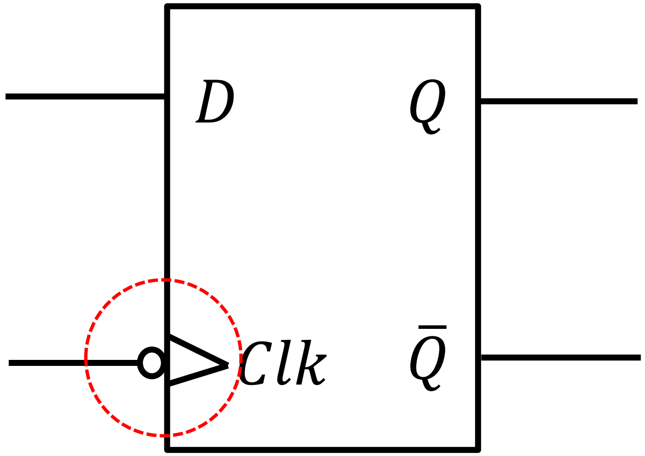
Master-slave D flip-flop



(b) Timing diagram

Figure 7.10. Master-slave D flip-flop.

Master-slave D flip-flop

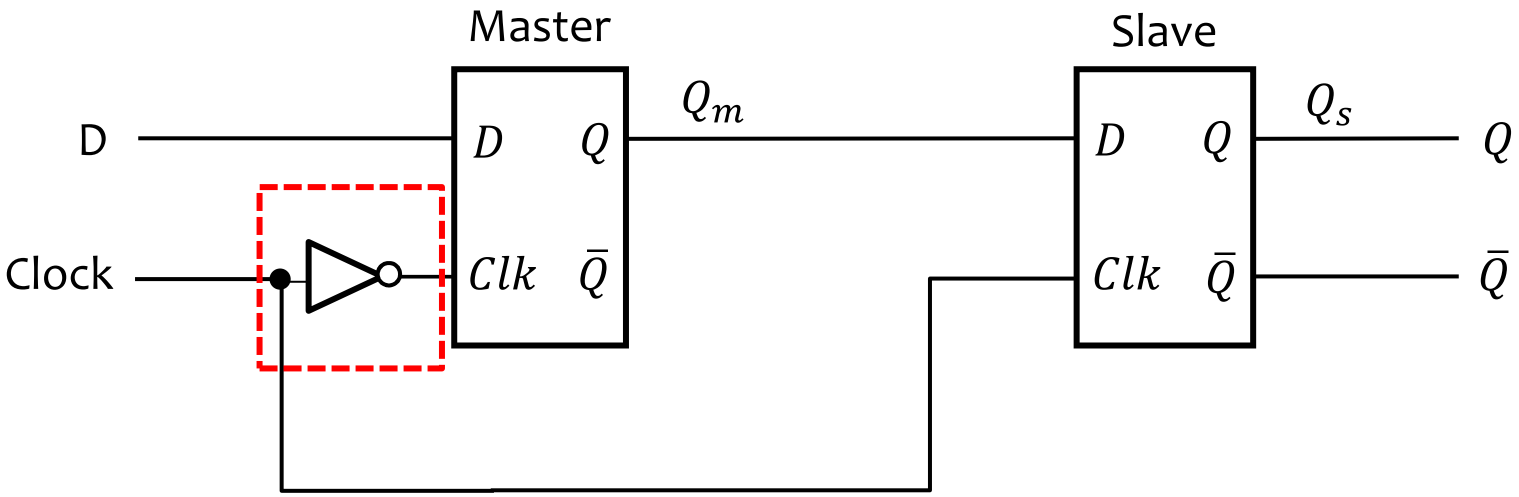


(c) Graphical symbol

Figure 7.10. Master-slave D flip-flop.

Master-slave D flip-flop

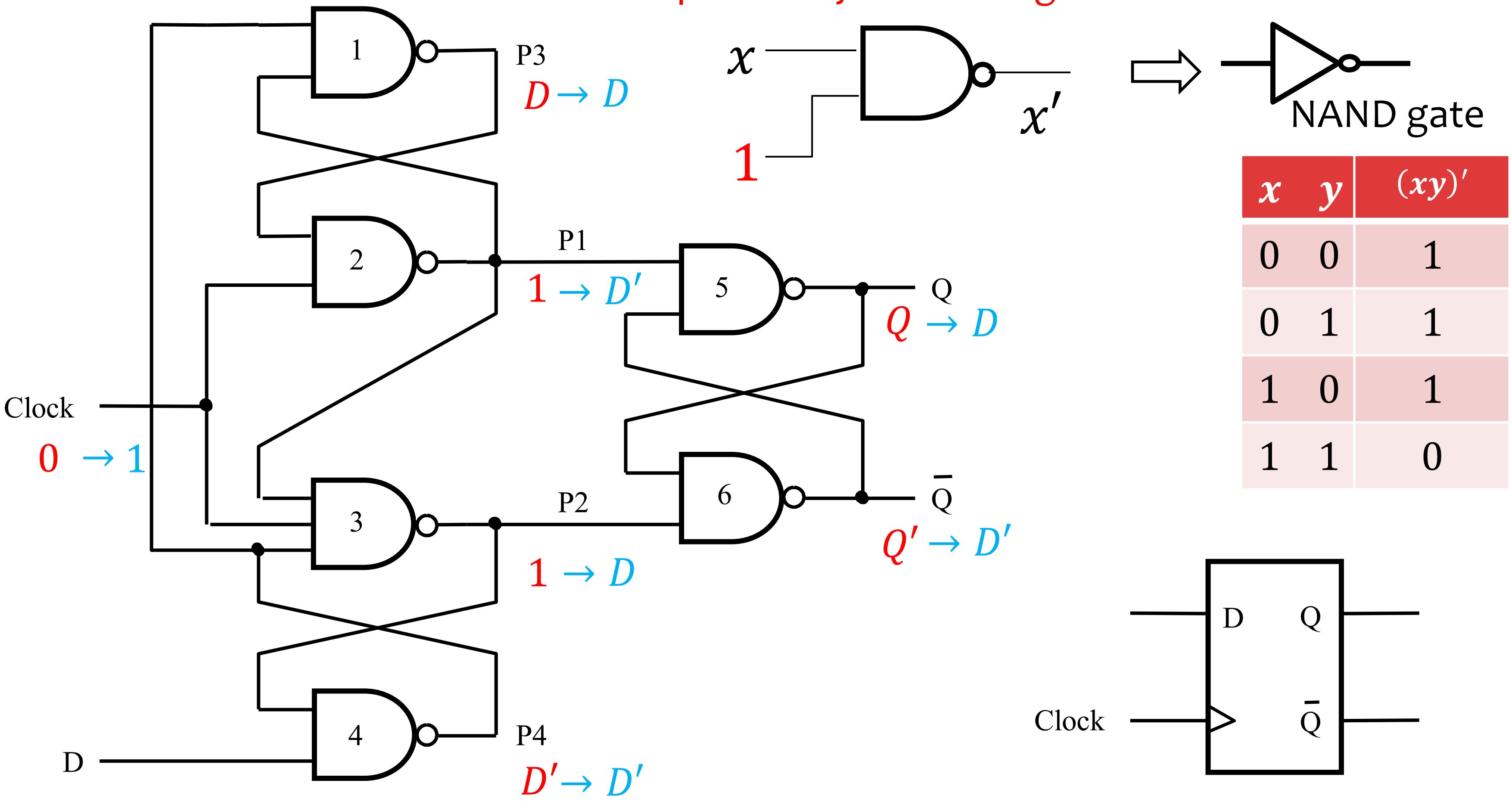
Positive edge triggered D flip-flop



(a) Circuit

A positive-edge-triggered D flip-flop

It requires only six NAND gates and fewer transistors



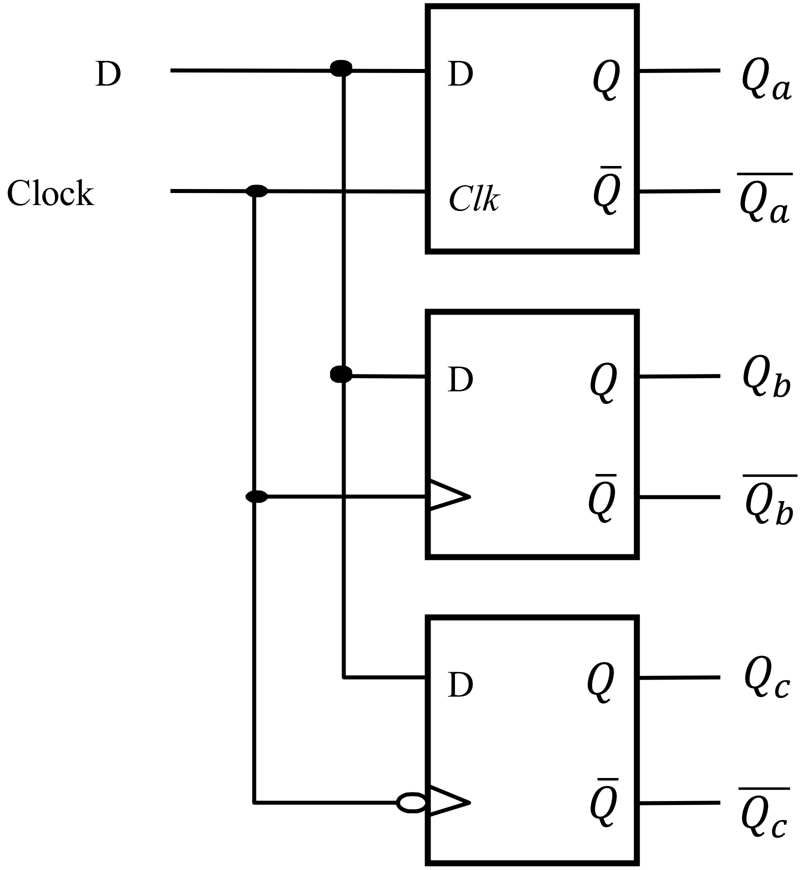
x	y	$(xy)'$
0	0	1
0	1	1
1	0	1
1	1	0

(a) Circuit

(b) Graphical symbol

Figure 7.11. A positive-edge-triggered D flip-flop.

Level-Sensitive versus Edge-Triggered Storage Elements



Level-sensitive

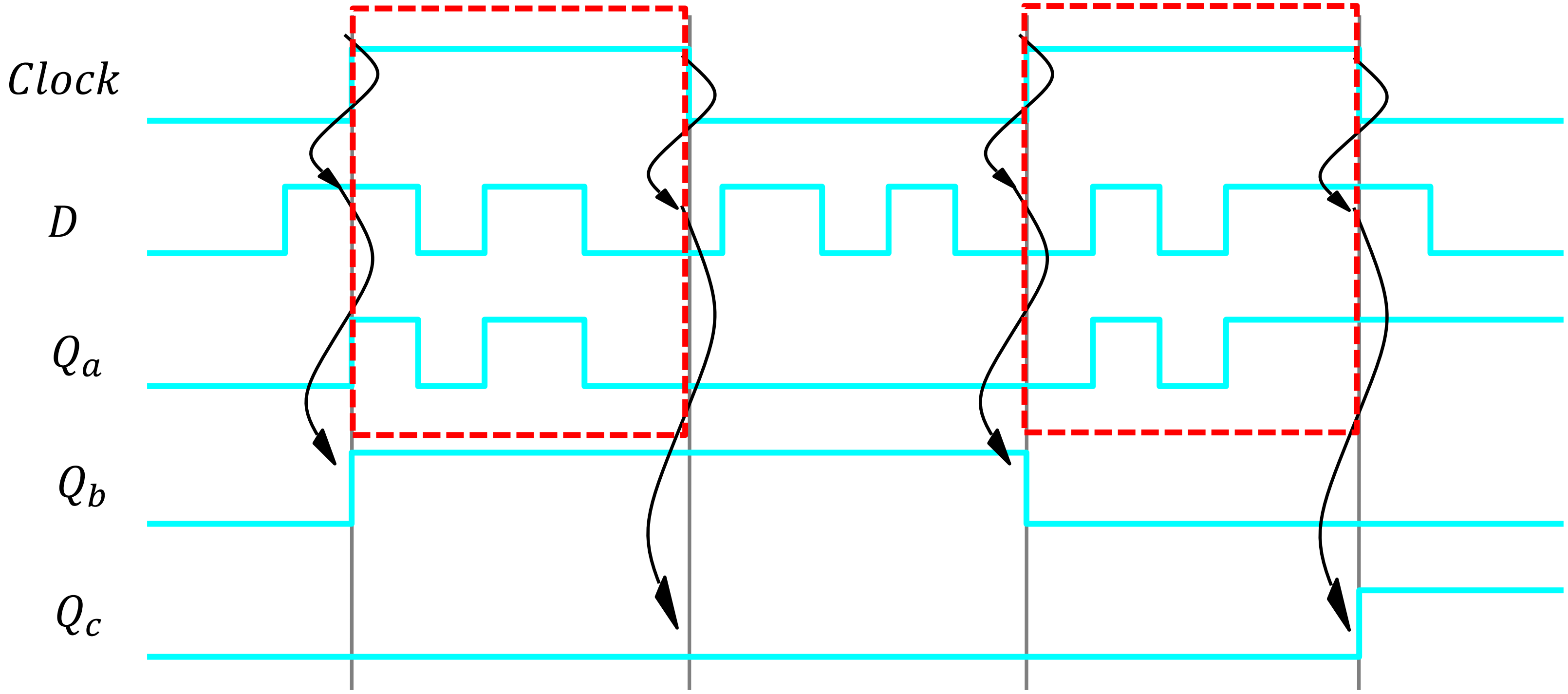
Positive-edge triggered

Negative-edge triggered

(a) Circuit

Figure 7.12. Comparison of level-sensitive and edge-triggered D storage elements.

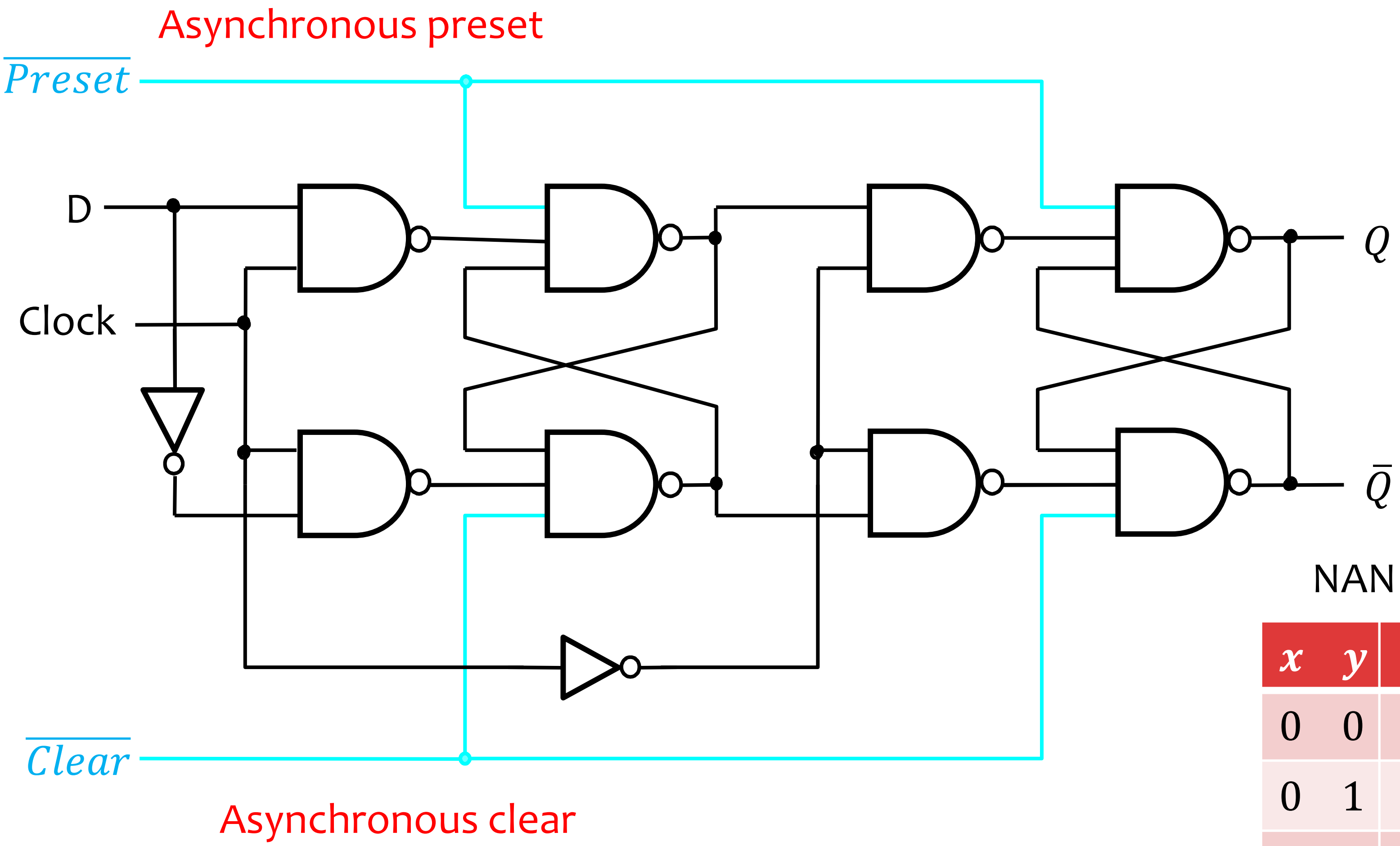
Level-Sensitive versus Edge-Triggered Storage Elements



(b) Timing diagram

Figure 7.12. Comparison of level-sensitive and edge-triggered D storage elements.

D flip-flop with *Clear* and *Preset*



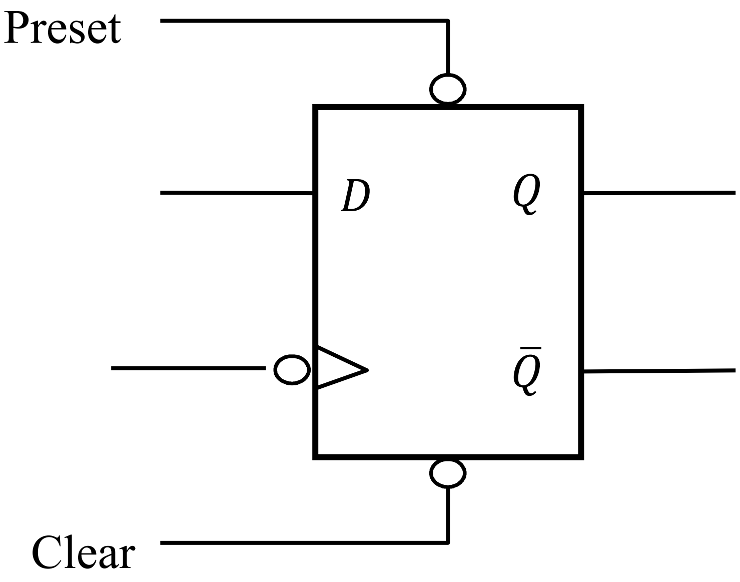
(a) Circuit

NAND gate

x	y	$(xy)'$
0	0	1
0	1	1
1	0	1
1	1	0

Figure 7.13. Master-slave D flip-flop with *Clear* and *Preset*.

D flip-flop with *Clear* and *Preset*



(b) Graphical symbol

Figure 7.13. Master-slave D flip-flop with *Clear* and *Preset*.

Synchronous reset for a D flip-flop

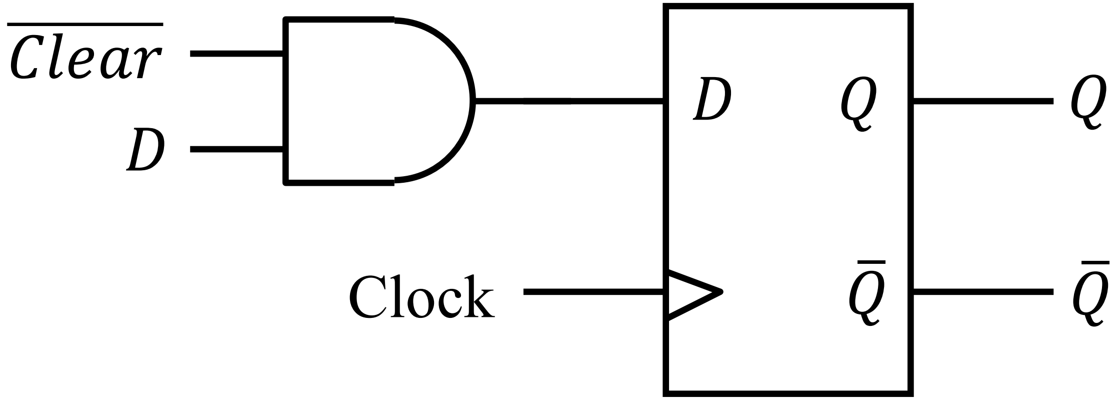


Figure 7.15. Synchronous reset for a D flip-flop.

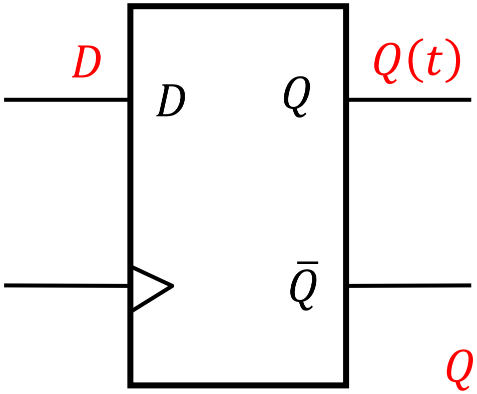
T FLIP-FLOP

D flip-flop

D	$Q(t+1)$
0	0
1	1

$Q(t+1) = D$

(b) Truth table



$Q(t+1) = D$

(c) Graphical symbol

$Q(t+1)$: Next state

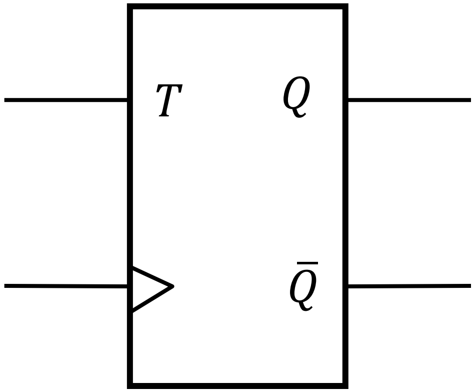
$Q(t)$: Present state

T flip-flop

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

$$Q(t+1) = T'Q(t) + TQ(t)'$$

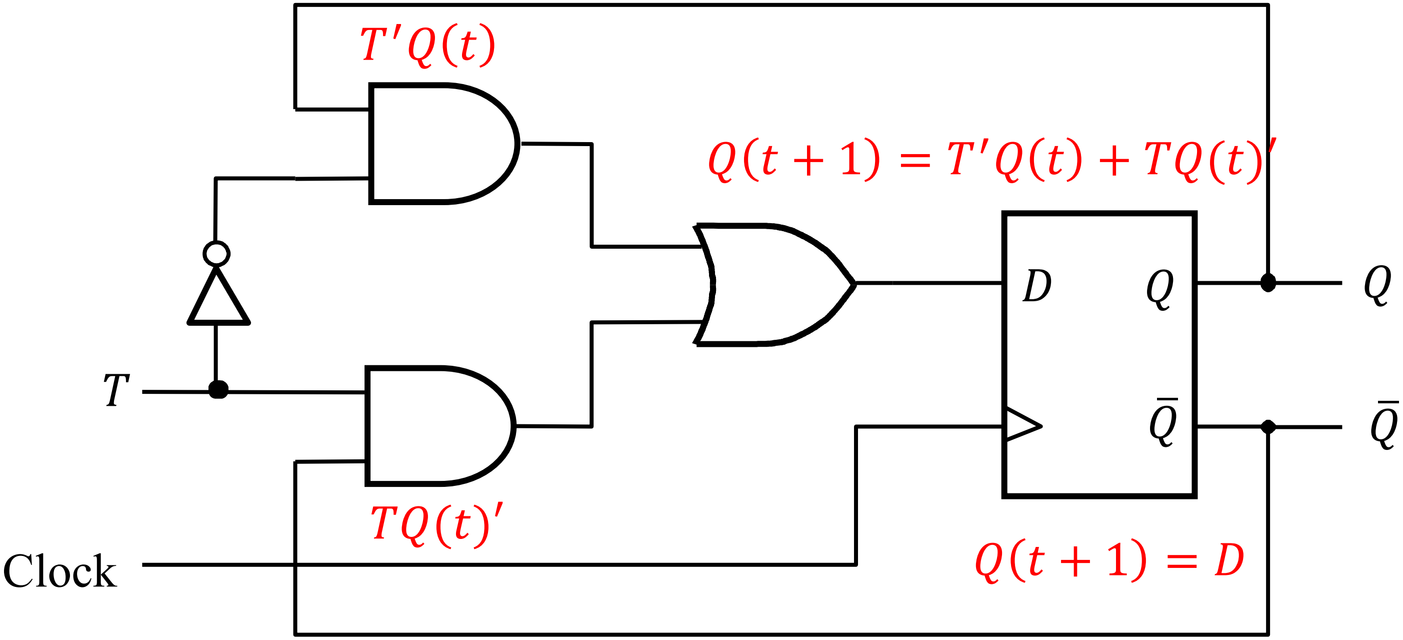
(b) Truth table



(c) Graphical symbol

Figure 7.16. T flip-flop.

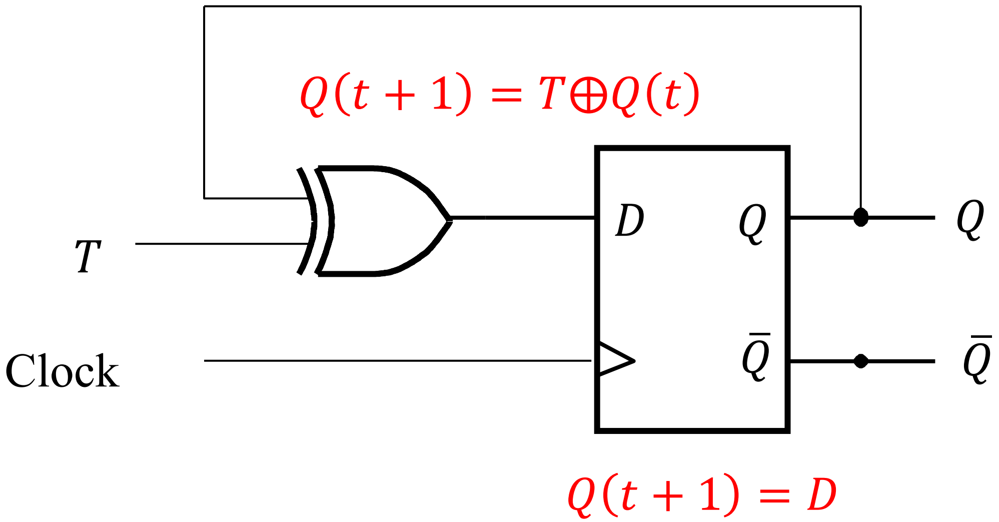
T flip-flop



(a) Circuit

Figure 7.16. T flip-flop.

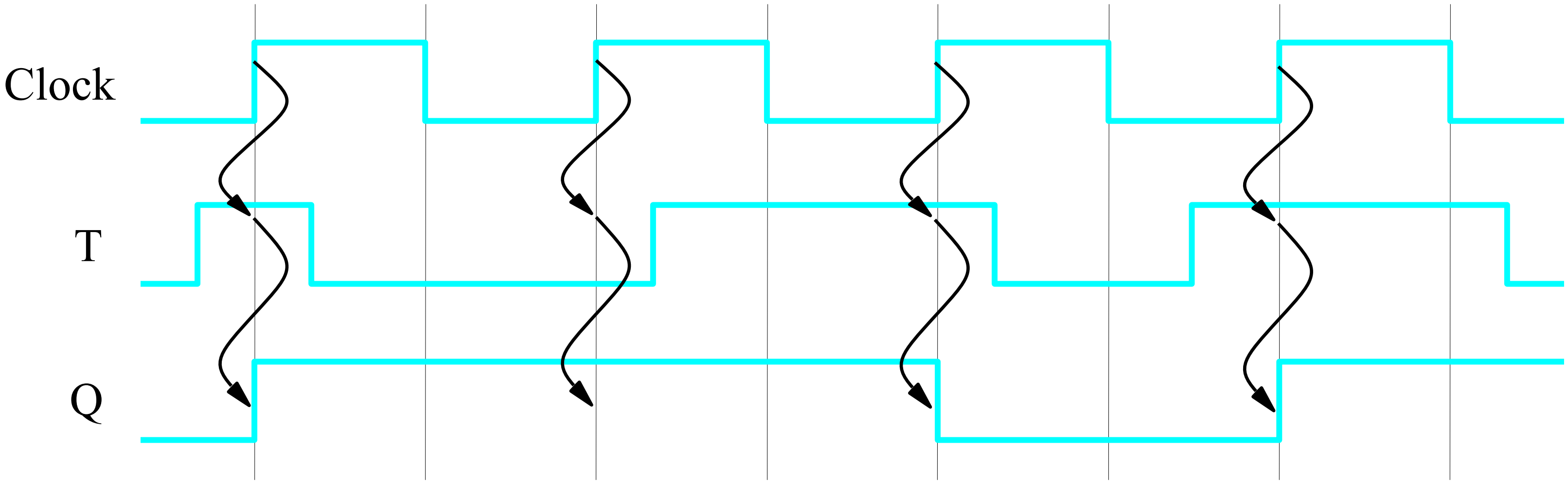
T flip-flop



(a) Circuit

Figure 7.16. T flip-flop.

T flip-flop



(d) Timing diagram

Figure 7.16. T flip-flop.

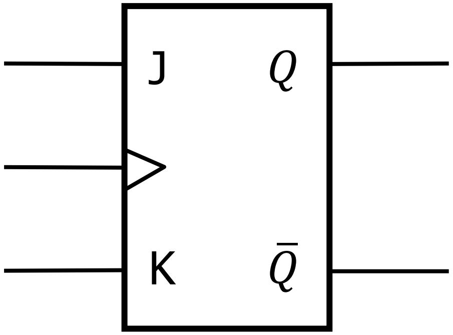
JK FLIP-FLOP

JK flip-flop

J	K	$Q(t+1)$	
0	0	$Q(t)$	No change
0	1	0	
1	0	1	
1	1	$Q'(t)$	Toggle

$$Q(t+1) = J'K'Q(t) + JK' + JKQ(t)'$$

(b) Truth table



(c) Graphical symbol

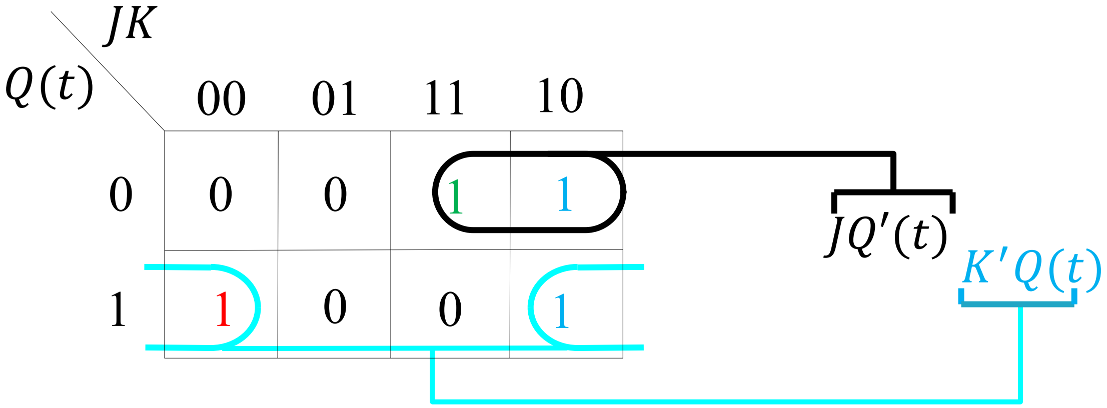
⬚ T-F/F

Figure 7.17. JK flip-flop.

JK flip-flop

$Q(t+1) = J'K'Q(t) + JK' + JKQ'(t)$

$JK' = JK'Q'(t) + JK'Q(t)$

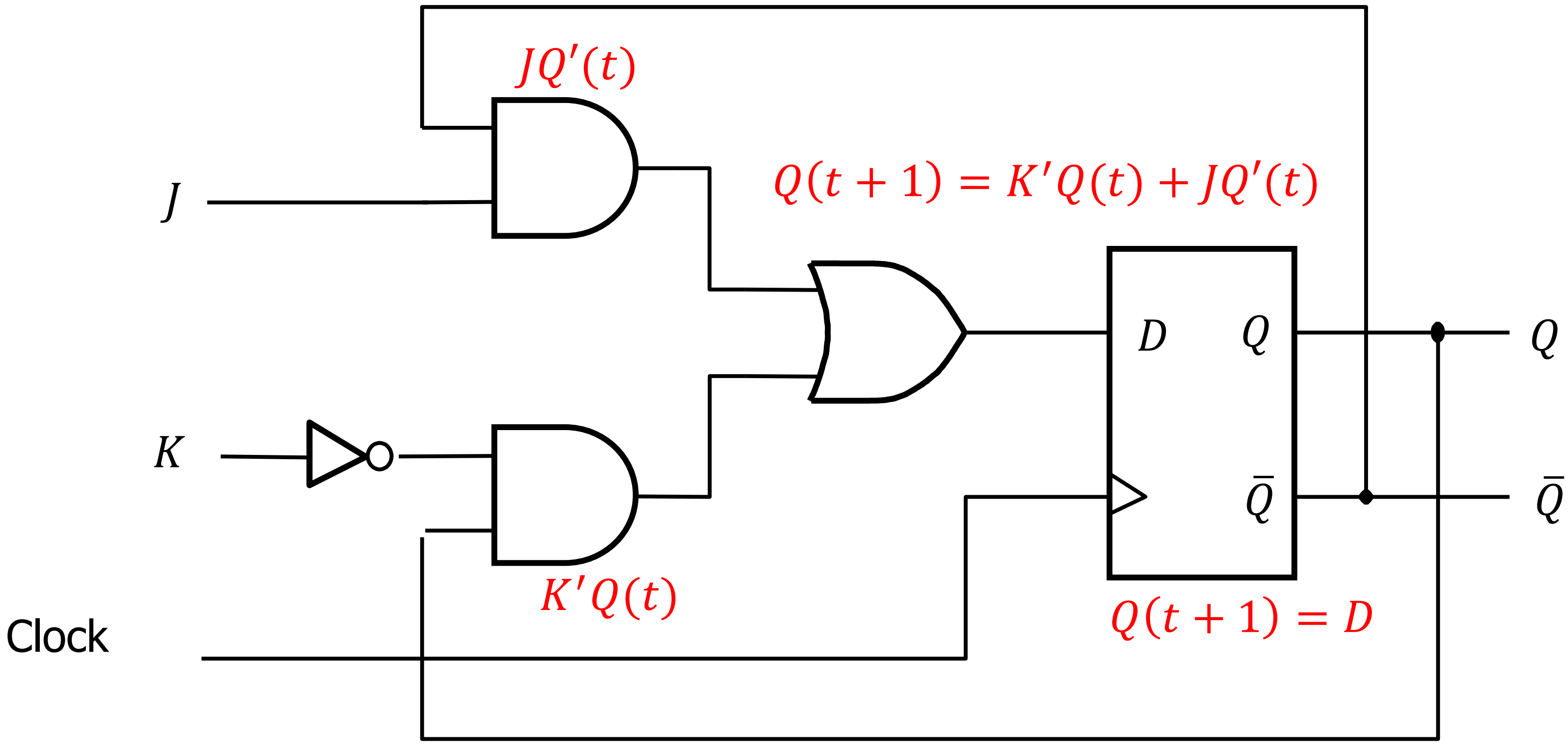


$Q(t)$ \ JK	00	01	11	10
0	0	0	1	1
1	1	0	0	1

$$Q(t+1) = K'Q(t) + JQ'(t)$$

Figure 7.17. JK flip-flop.

JK flip-flop



(a) Circuit

Figure 7.17. JK flip-flop.

SUMMARY OF TERMINOLOGY

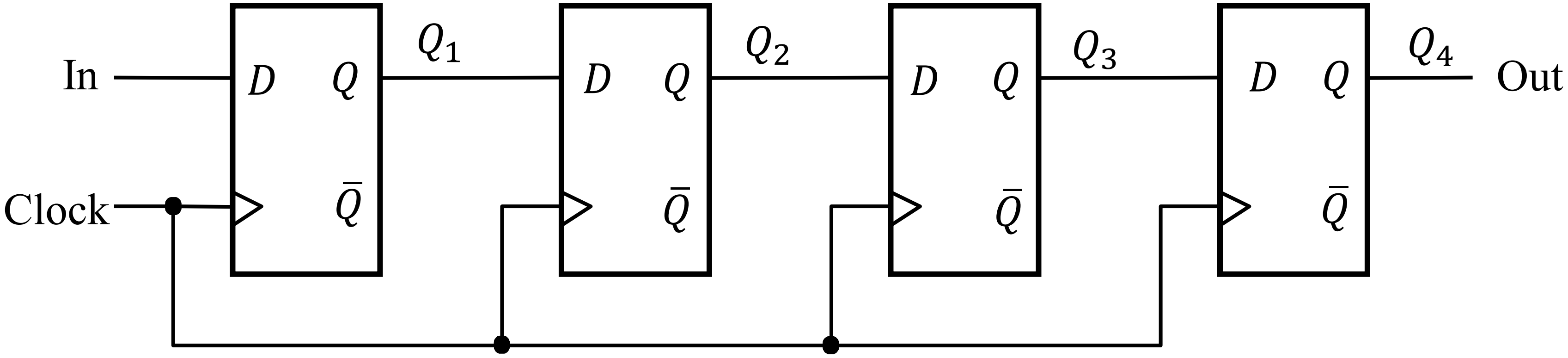
Summary of Terminology

- Basic latch is a feedback connection of two NOR gates or two NAND gates.
- Gated latch a basic latch that includes input gating and a control signal.
  - Gated SR latch uses the S and R inputs to set the latch to 1 or reset it to 0.
  - Gated D latch uses the D input to force the latch into a state that has the same logic value as the D input
- A flip-flop: Output state can be changed only on the edge of the controlling clock signal
  - Edge-triggered flip-flop is affected only by the input values present when the active edge of the clock occurs.
  - Master-slave flip-flop is built with two gated latches.

REGISTERS

Shift Register

When a set of n flip-flops is used to store n bits of information, such as n-bit number, we refer to these flip-flops as a register.



(a) Circuit

A register that provides the ability to shift its contents is called a shift register.

Figure 7.18. A simple shift register.

Shift Register

Input data

	In	Q_1	Q_2	Q_3	$Q_4 =$ Out
t_0	1	0	0	0	0
t_1	0	1	0	0	0
t_2	1	0	1	0	0
t_3	1	1	0	1	0
t_4	1	1	1	0	1
t_5	0	1	1	1	0
t_6	0	0	1	1	1
t_7	0	0	0	1	1

(b) A sample sequence

Figure 7.18. A simple shift register.

Parallel-access shift register

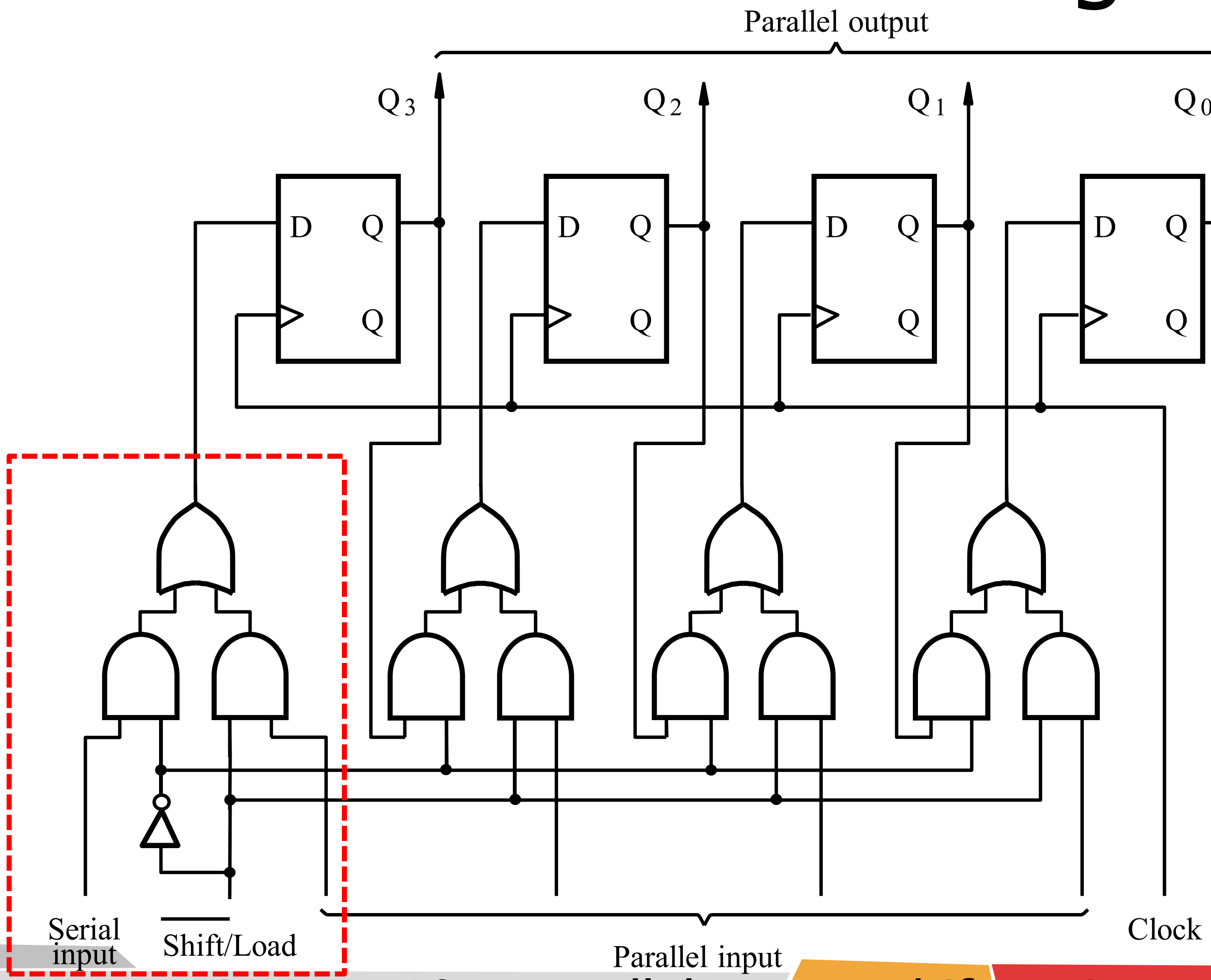


Figure 7.19. Parallel-access shift register.

Parallel-access shift register

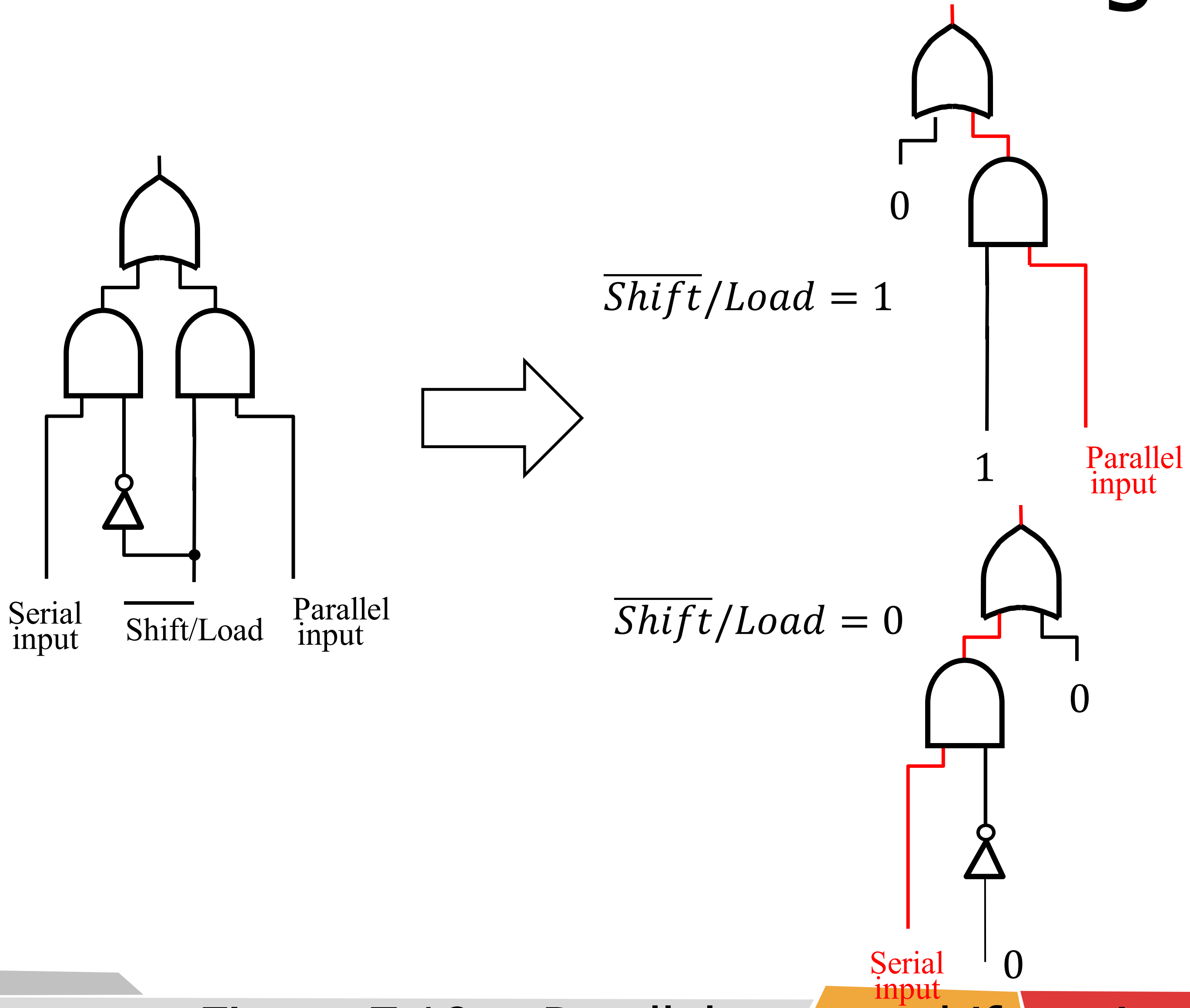


Figure 7.19. Parallel-access shift register.

Parallel-access shift register

Serial input

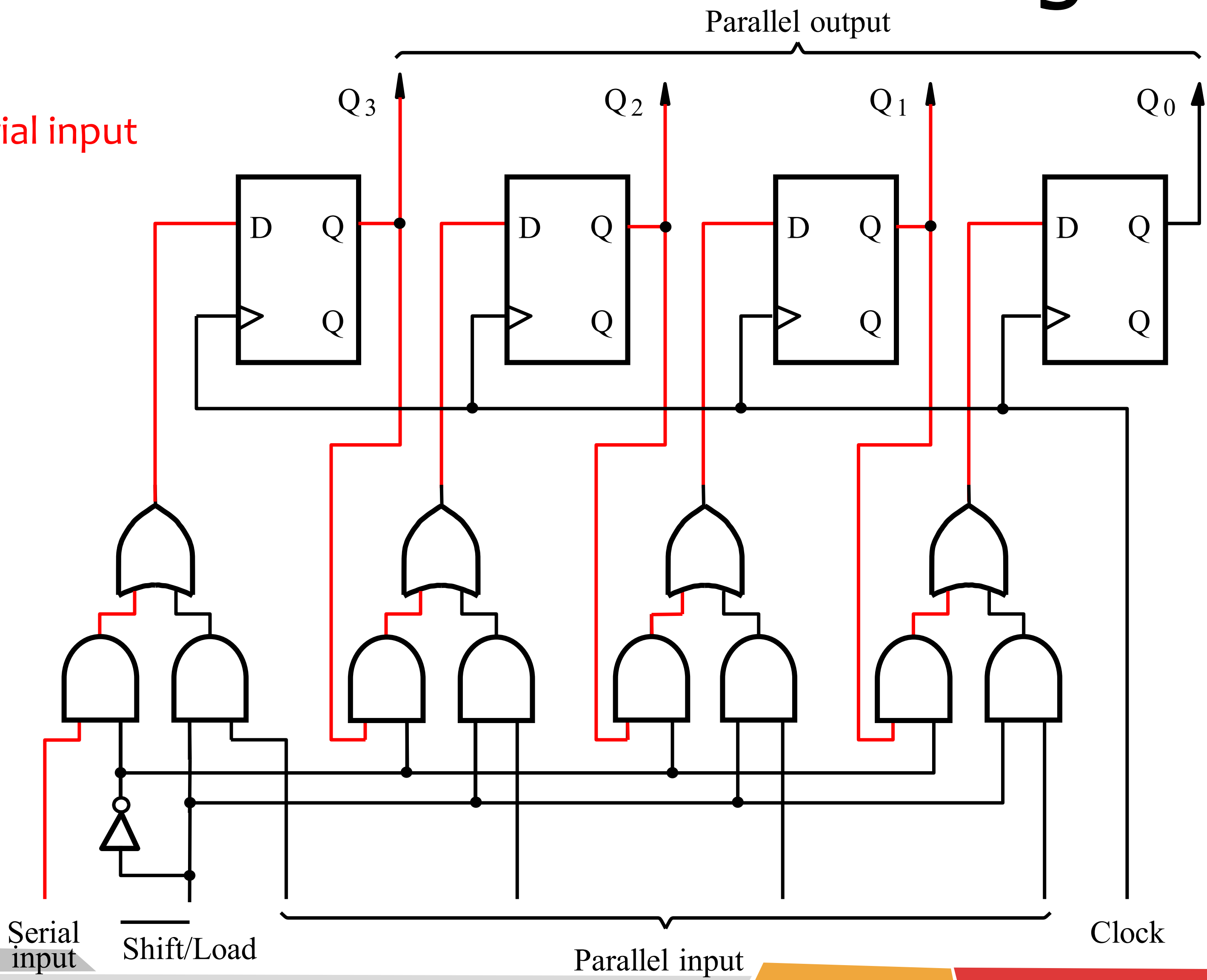


Figure 7.19. Parallel-access shift register.

Parallel-access shift register

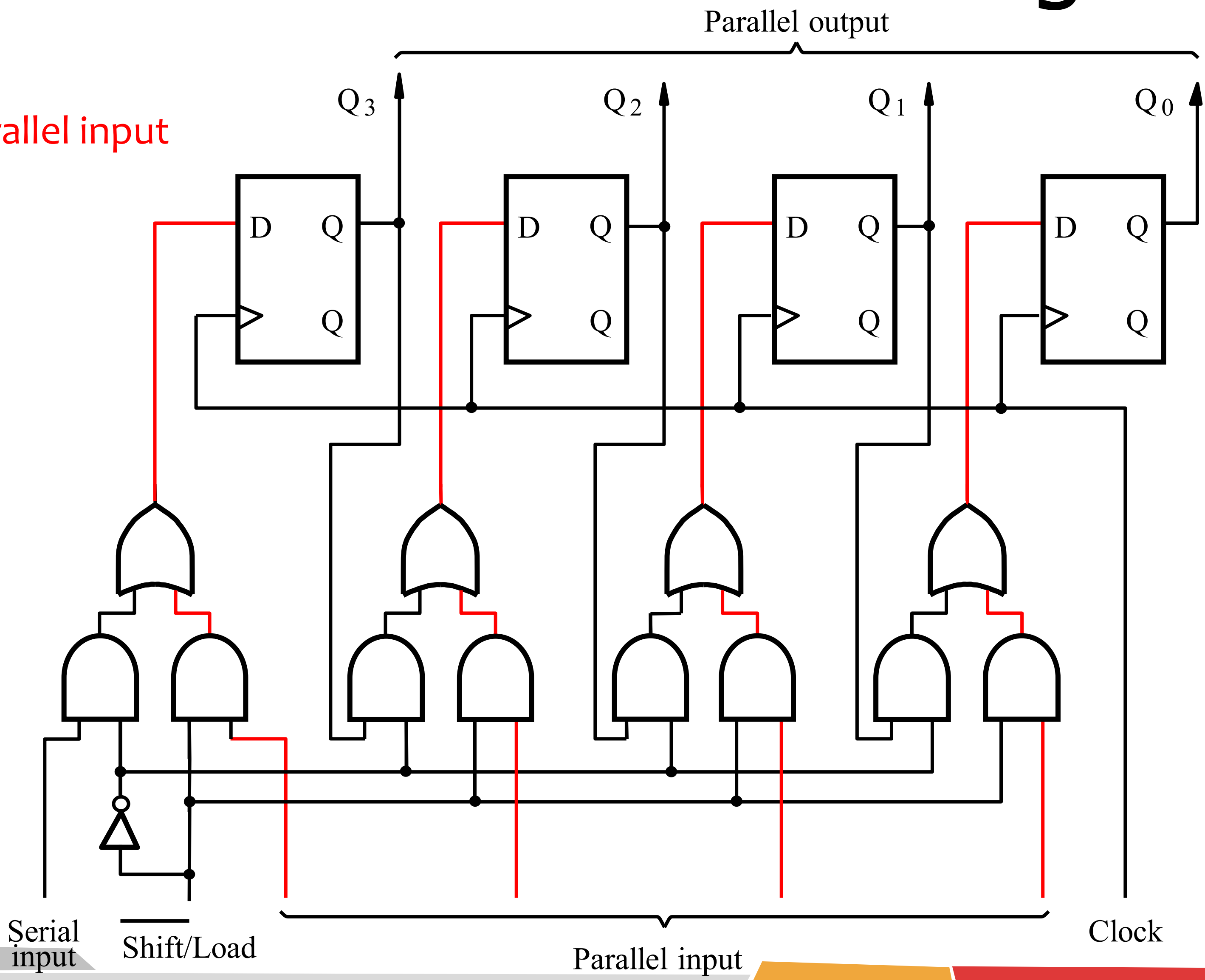
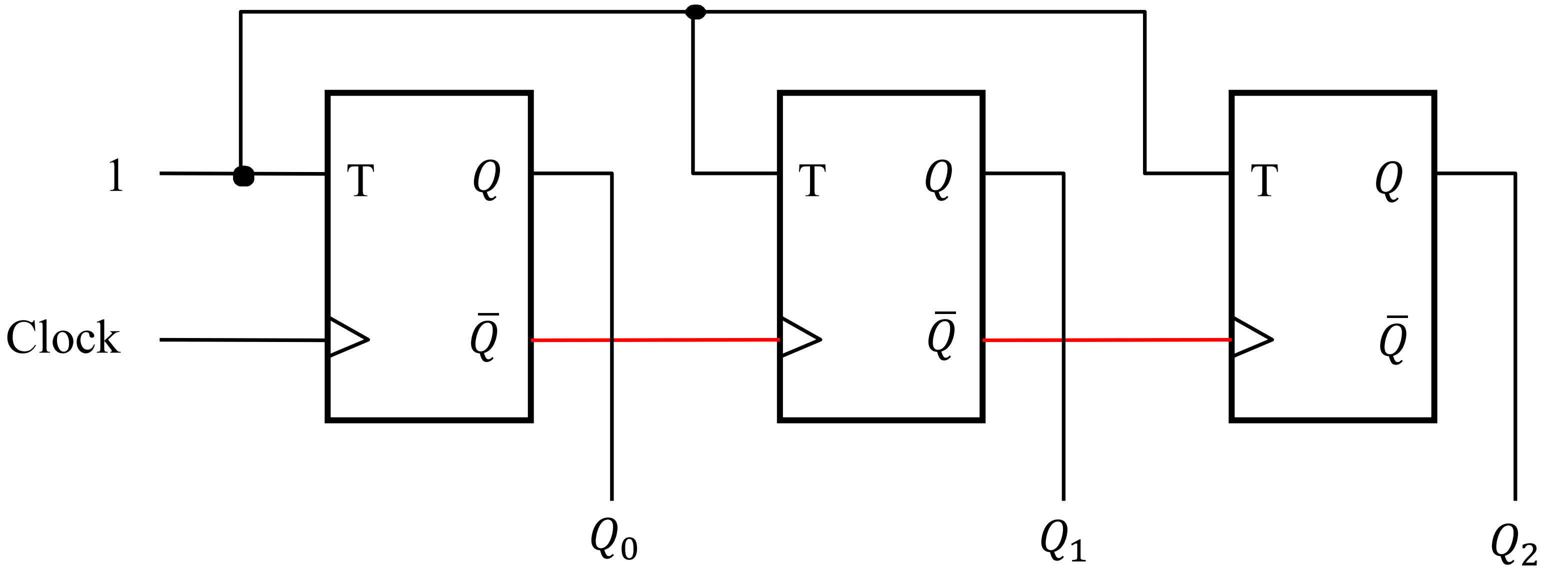


Figure 7.19. Parallel-access shift register.

COUNTERS

Asynchronous Counters

Up-Counter with T Flip-Flops

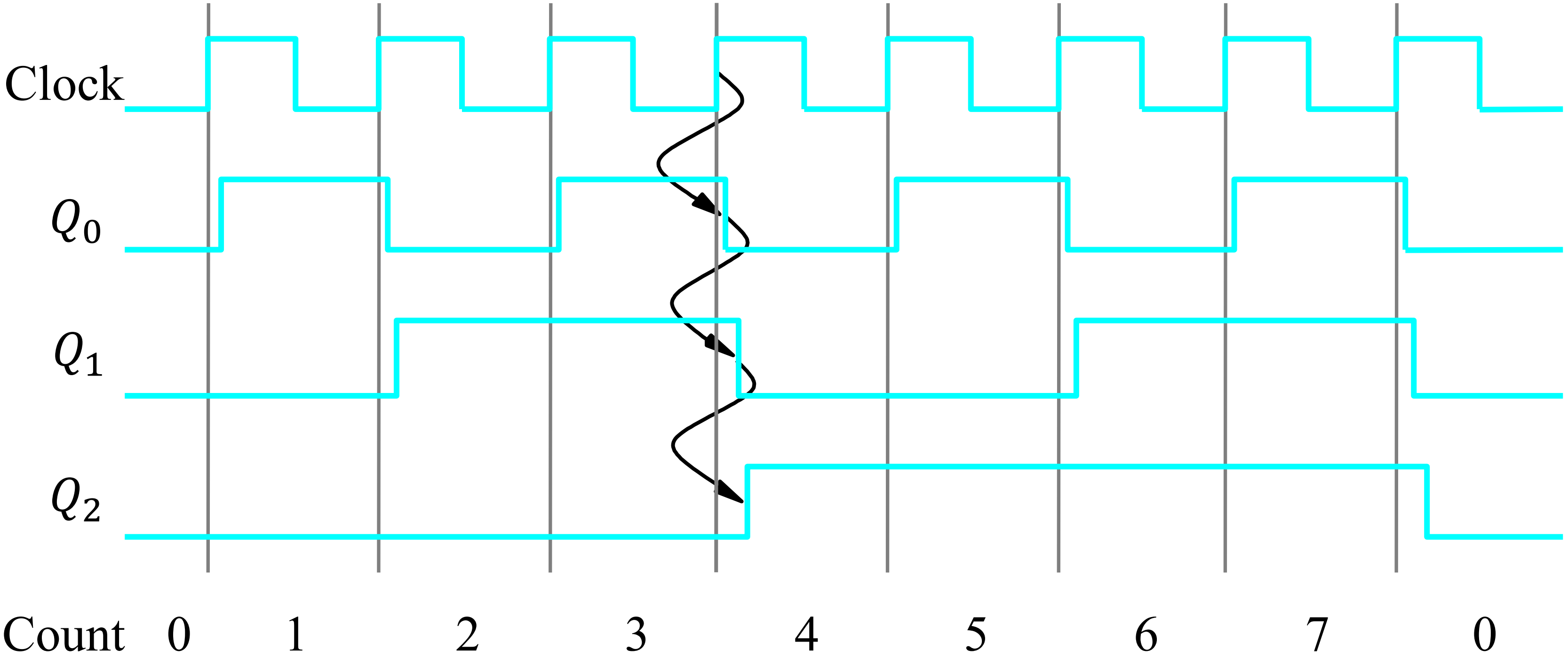


(a) Circuit

Figure 7.20. A three-bit up-counter.

Asynchronous Counters

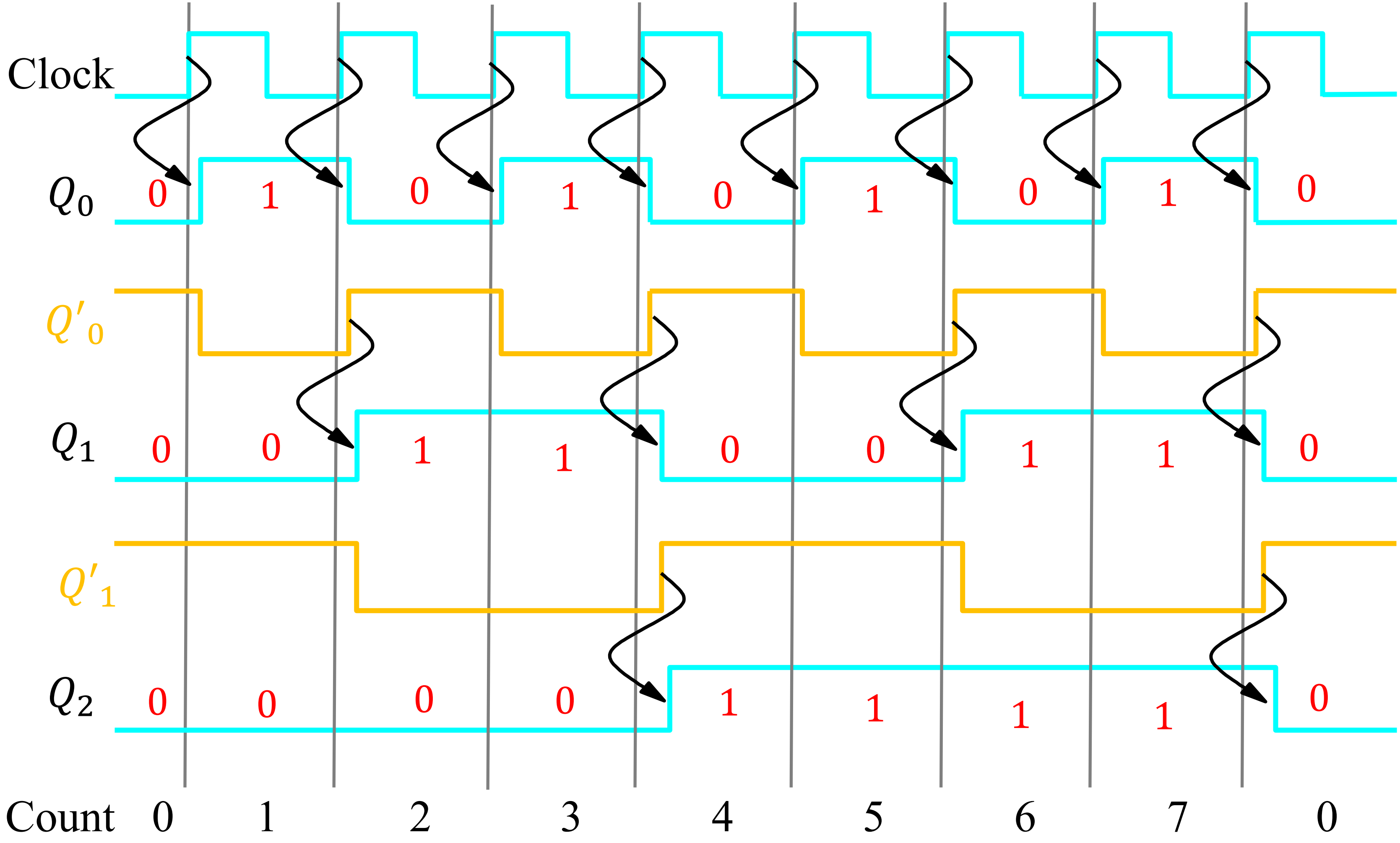
Up-Counter with T Flip-Flops



(b) Timing diagram

Figure 7.20. A three-bit up-counter.

Asynchronous Counters

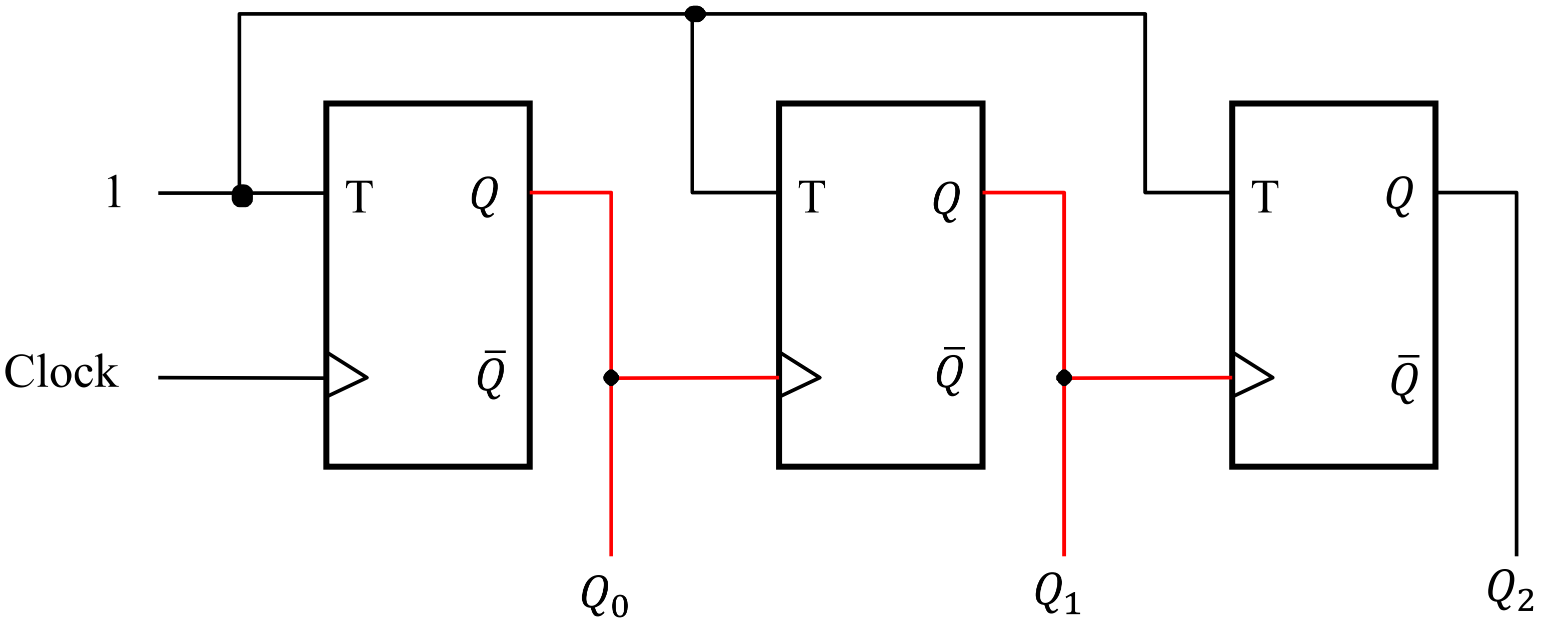


(b) Timing diagram

Figure 7.20. A three-bit up-counter.

Asynchronous Counters

Down-Counter with T Flip-Flops

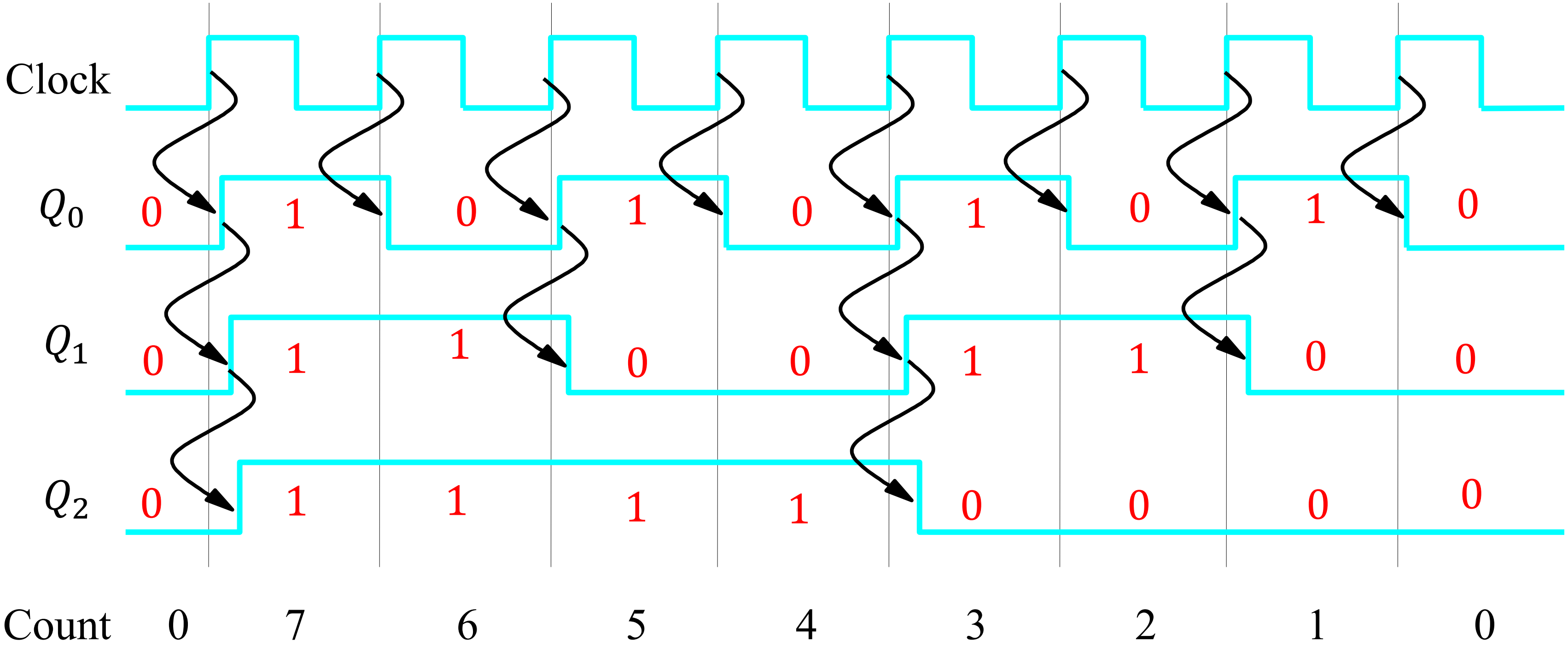


(a) Circuit

Figure 7.21. A three-bit down-counter.

Asynchronous Counters

Down-Counter with T Flip-Flops



Asynchronous counters are simple but not very fast

(b) Timing diagram

Figure 7.21. A three-bit down-counter.

Synchronous Counters

Synchronous Counter with T Flip-Flops

$T_0 = 1$

$T_1 = Q_0$

$T_2 = Q_0 Q_1$

$T_3 = Q_0 Q_1 Q_2$

$\vdots$

$T_n = Q_0 Q_1 \cdots Q_{n-1}$

Clock cycle	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

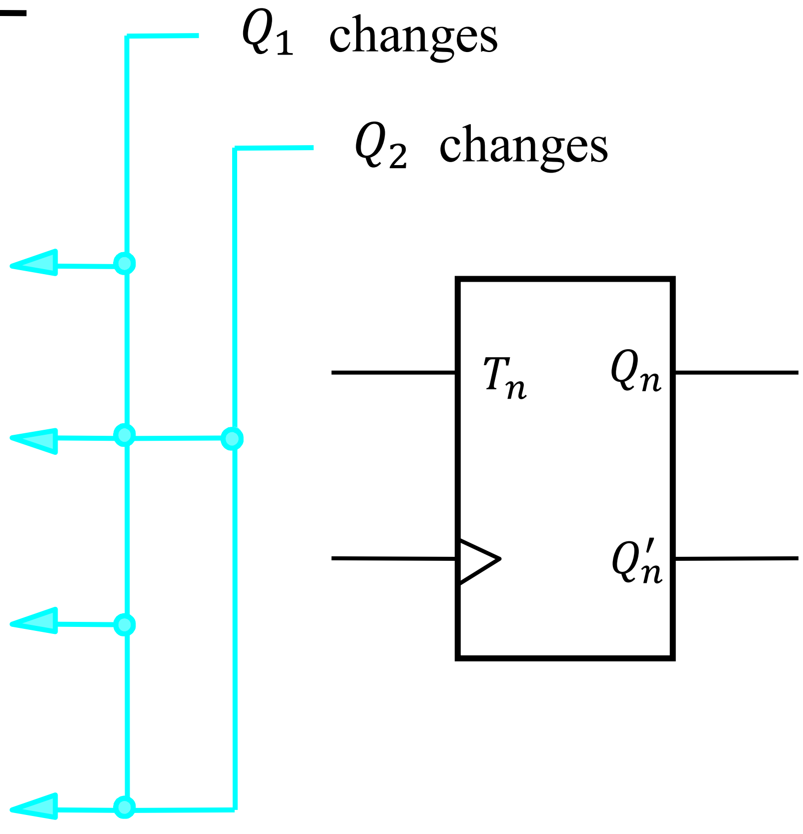
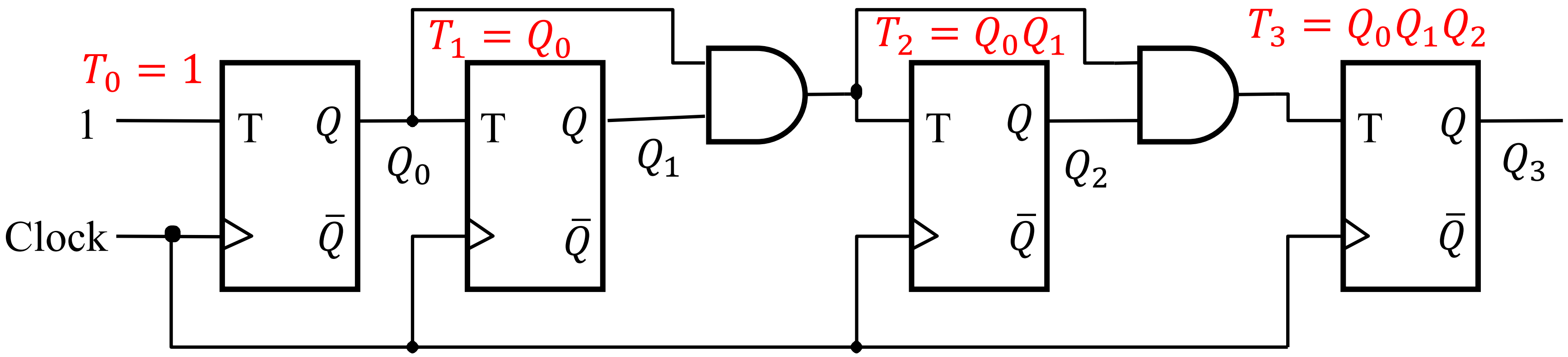


Table 7.1. Derivation of the synchronous up-counter.

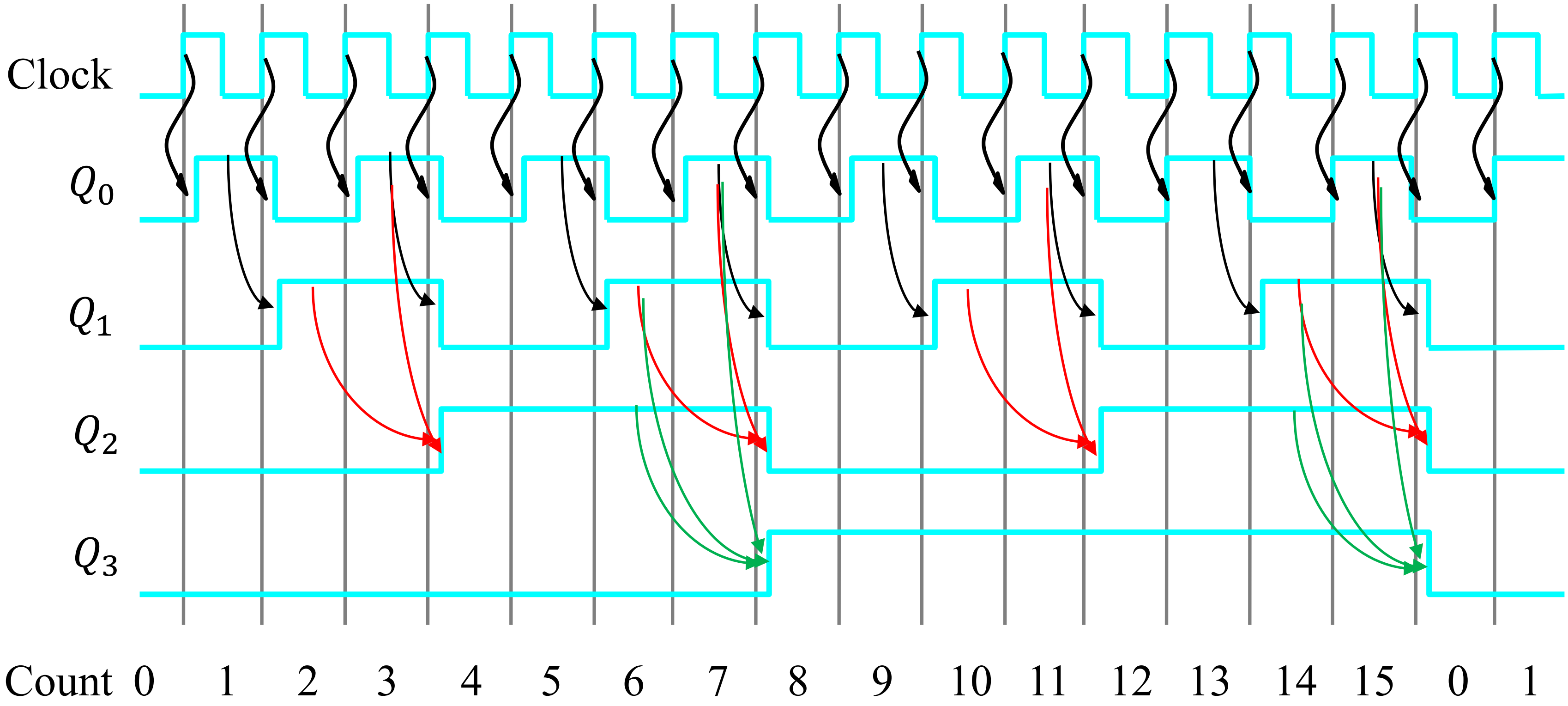
A four-bit synchronous up-counter



(a) Circuit

Figure 7.22. A four-bit synchronous up-counter.

A four-bit synchronous up-counter



(b) Timing diagram

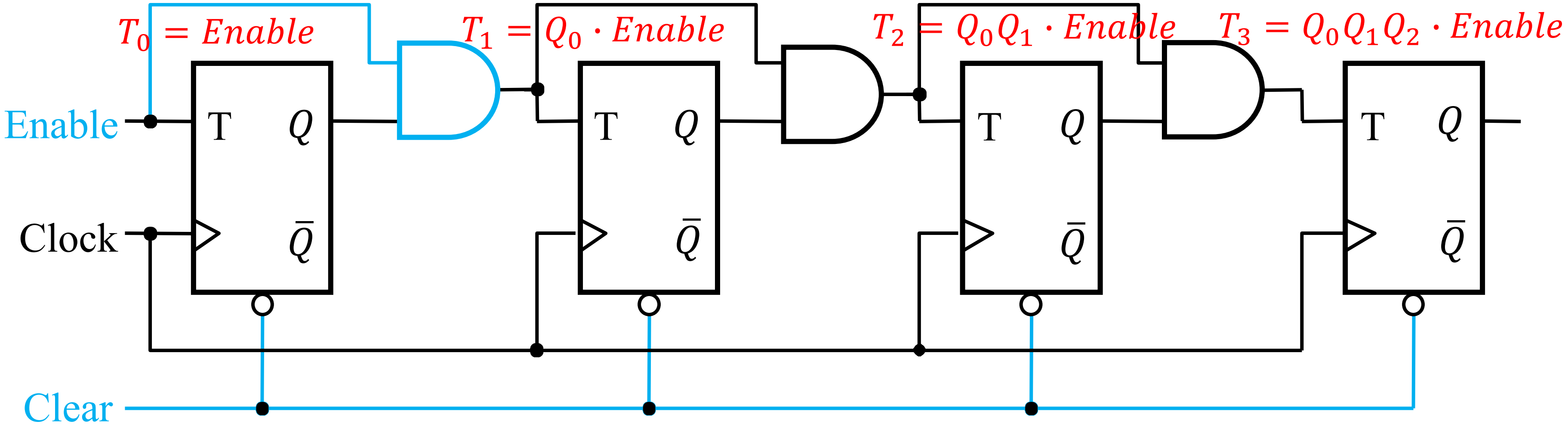
Figure 7.22. A four-bit synchronous up-counter.

Enable and Clear Capability

Enable control signal controls directly the T input of the first flip-flop.

If Enable = 0, then all T inputs will be equal to 0.

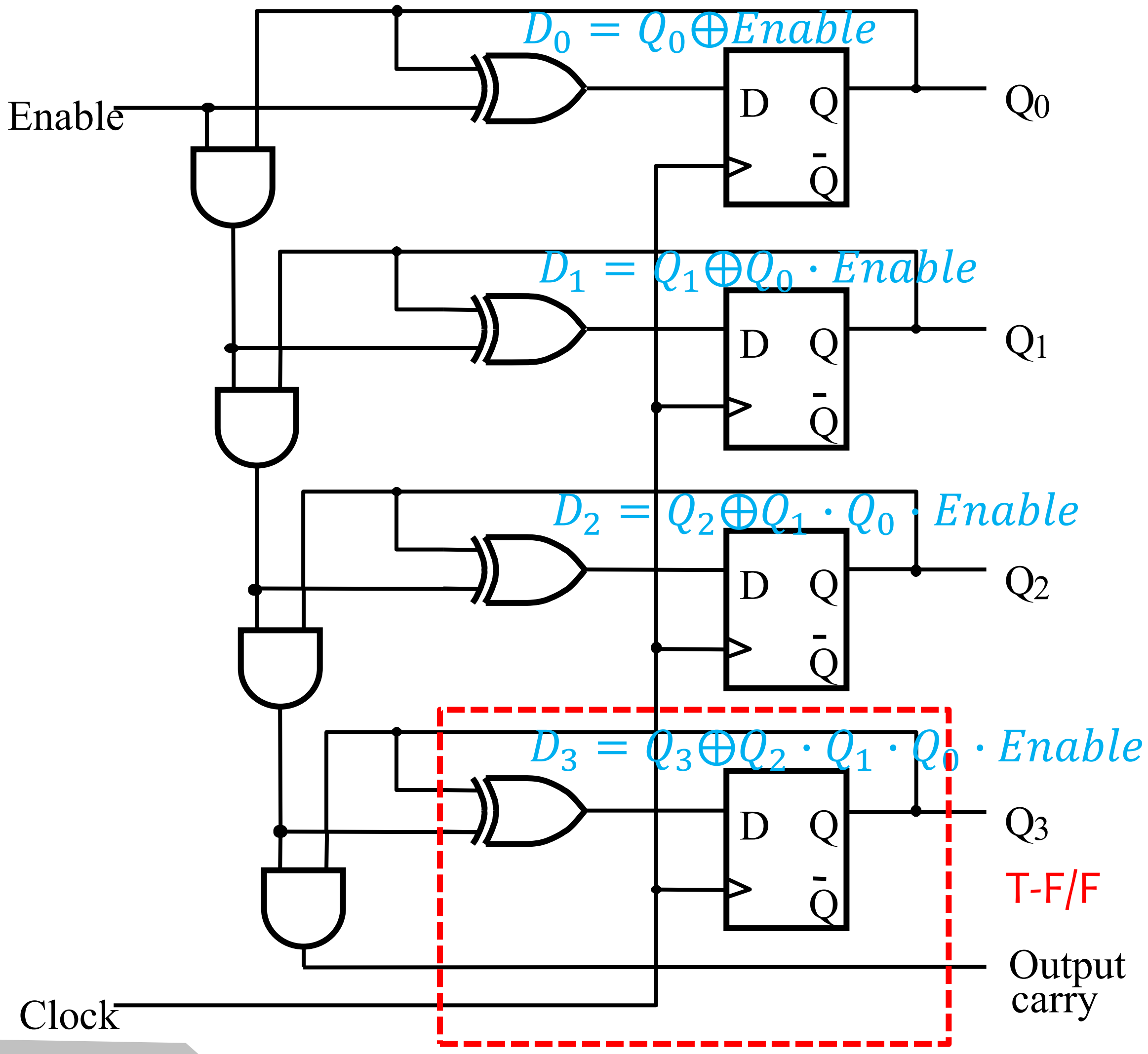
If Enable = 1, then the counter operates as expected.



To start with the count equal to zero, the clear inputs on all flip-flops can be tied together and driven by a Clear control input.

Figure 7.23. Inclusion of Enable and Clear capability.

Synchronous counter with D flip-flops



$$D_0 = Q_0 \oplus Enable$$

$$D_1 = Q_1 \oplus Q_0 \cdot Enable$$

$$D_2 = Q_2 \oplus Q_1 \cdot Q_0 \cdot Enable$$

$$D_3 = Q_3 \oplus Q_2 \cdot Q_1 \cdot Q_0 \cdot Enable$$

T-F/F $Q(t+1) = T \oplus Q(t)$

Figure 7.24. A four-bit counter with D flip-flops.

A counter with parallel-load capability

$Load = 0$

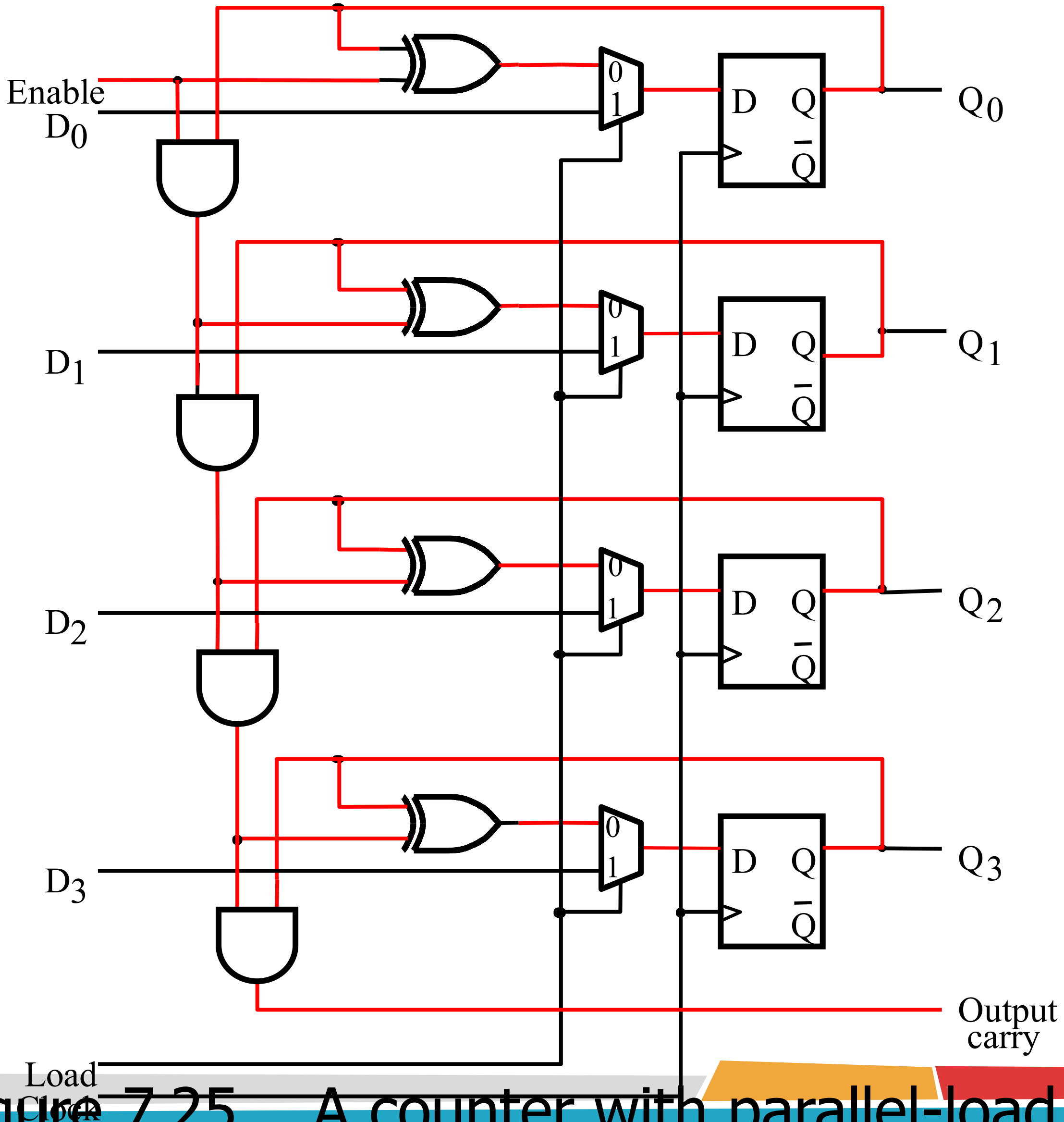


Figure 7.25. A counter with parallel-load capability.

A counter with parallel-load capability

$Load = 1$

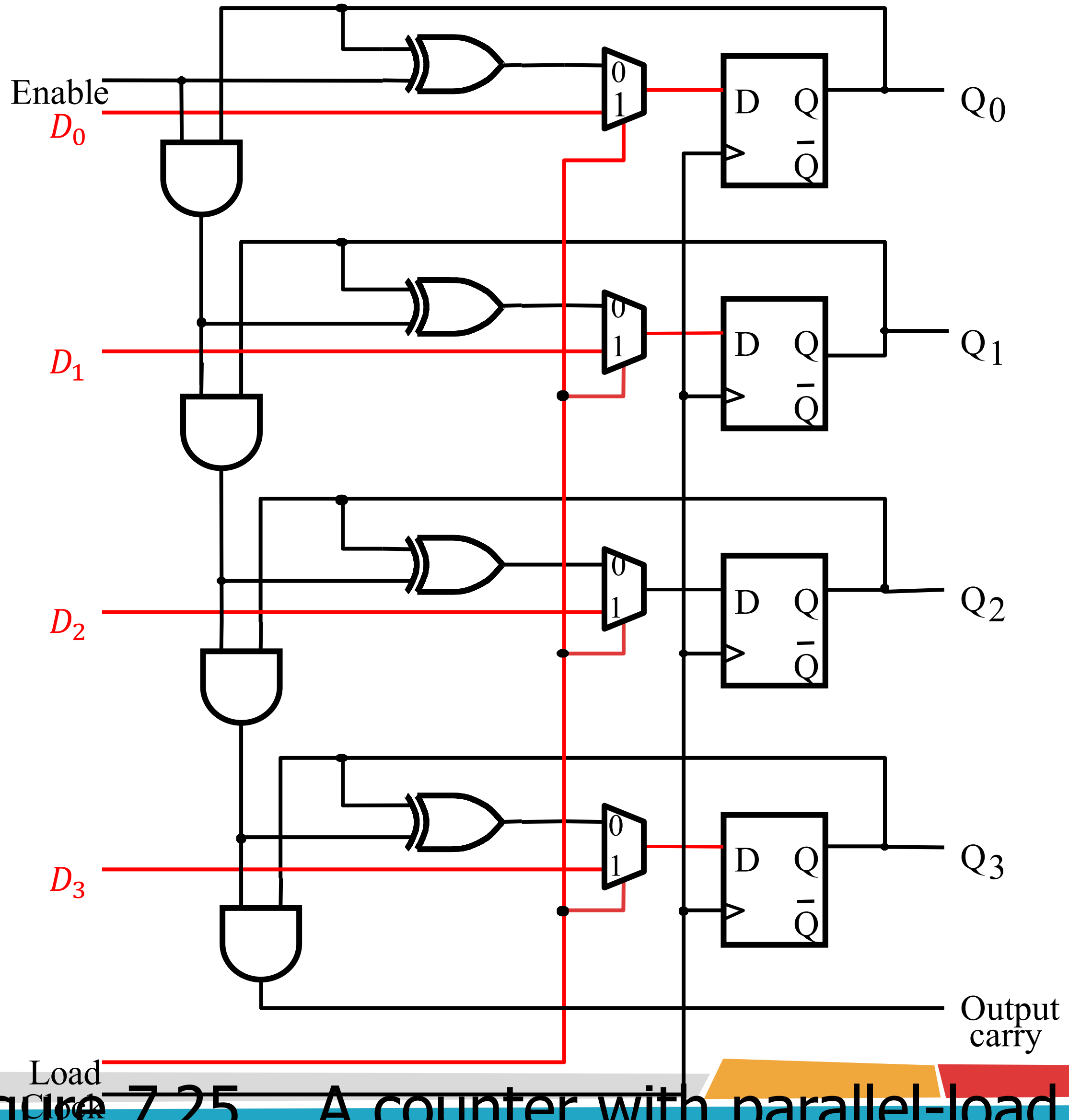
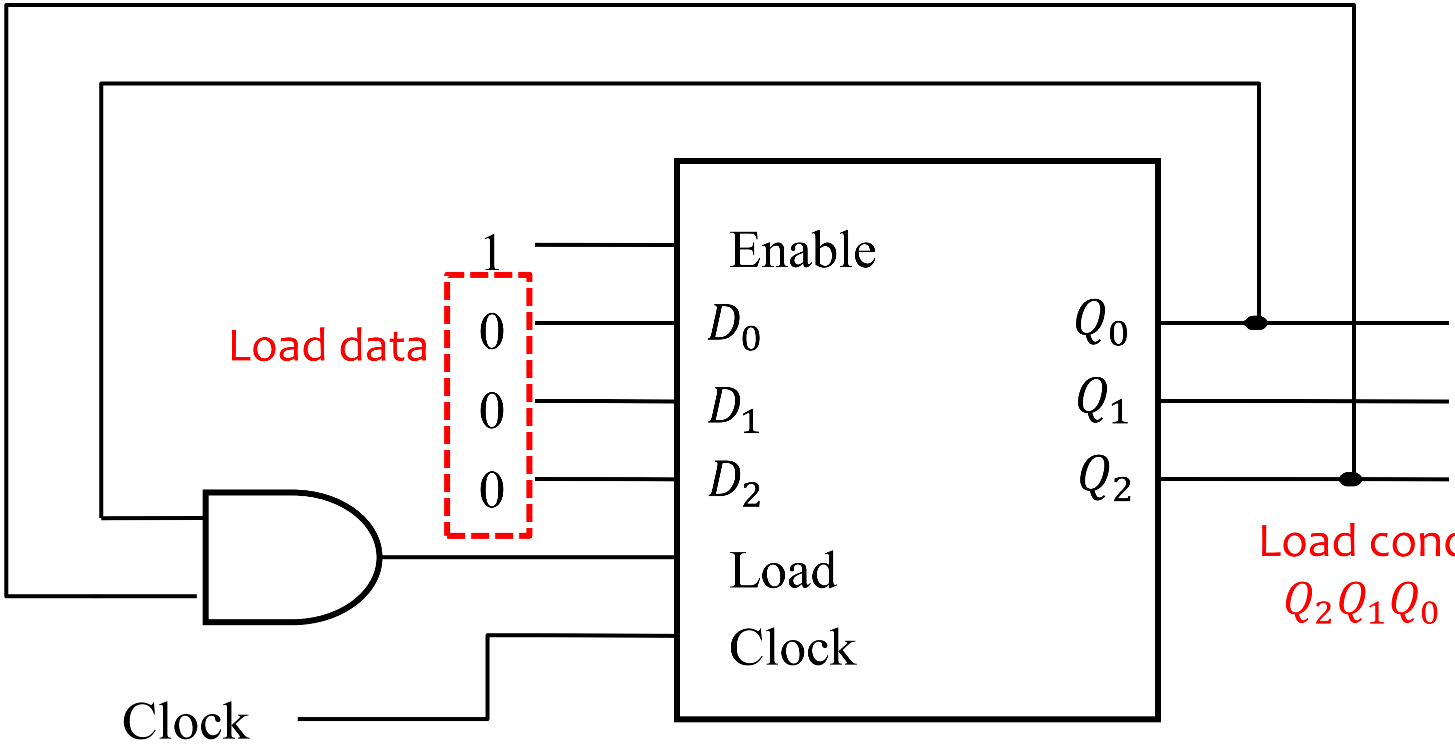


Figure 7.25. A counter with parallel-load capability.

RESET SYNCHRONIZATION

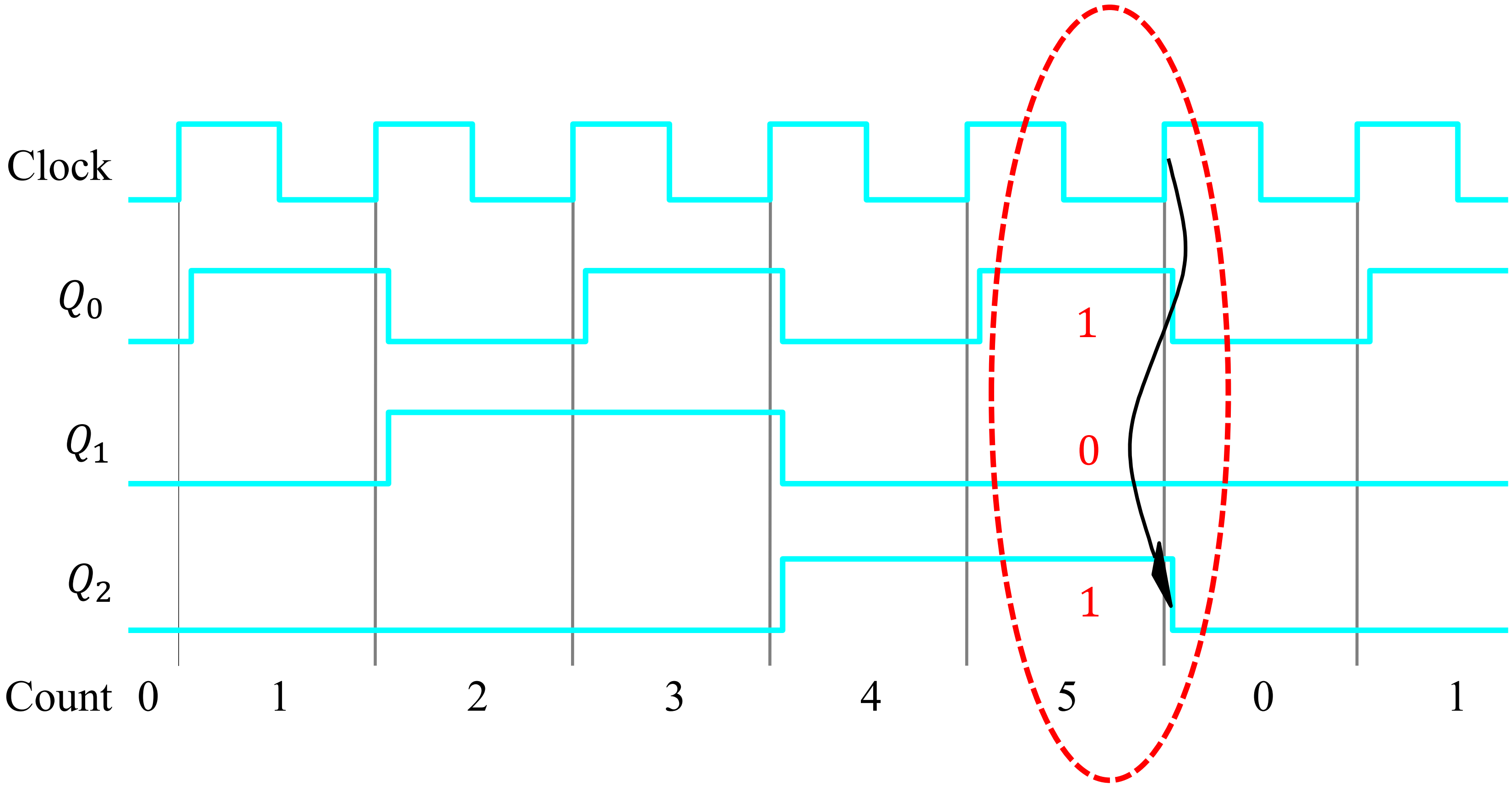
A modulo-6 counter with synchronous reset



(a) Circuit

Figure 7.26. A modulo-6 counter with synchronous reset.

A modulo-6 counter with synchronous reset

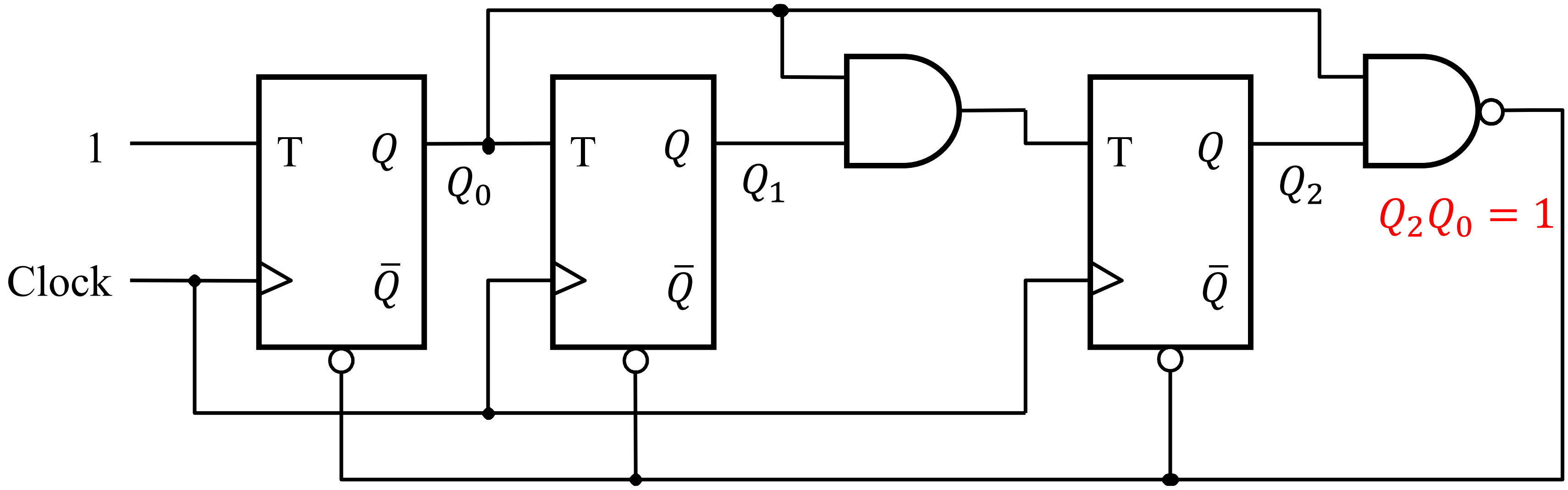


(b) Timing diagram

Load condition
$Q_2Q_1Q_0 = 101$

Figure 7.26. A modulo-6 counter with synchronous reset.

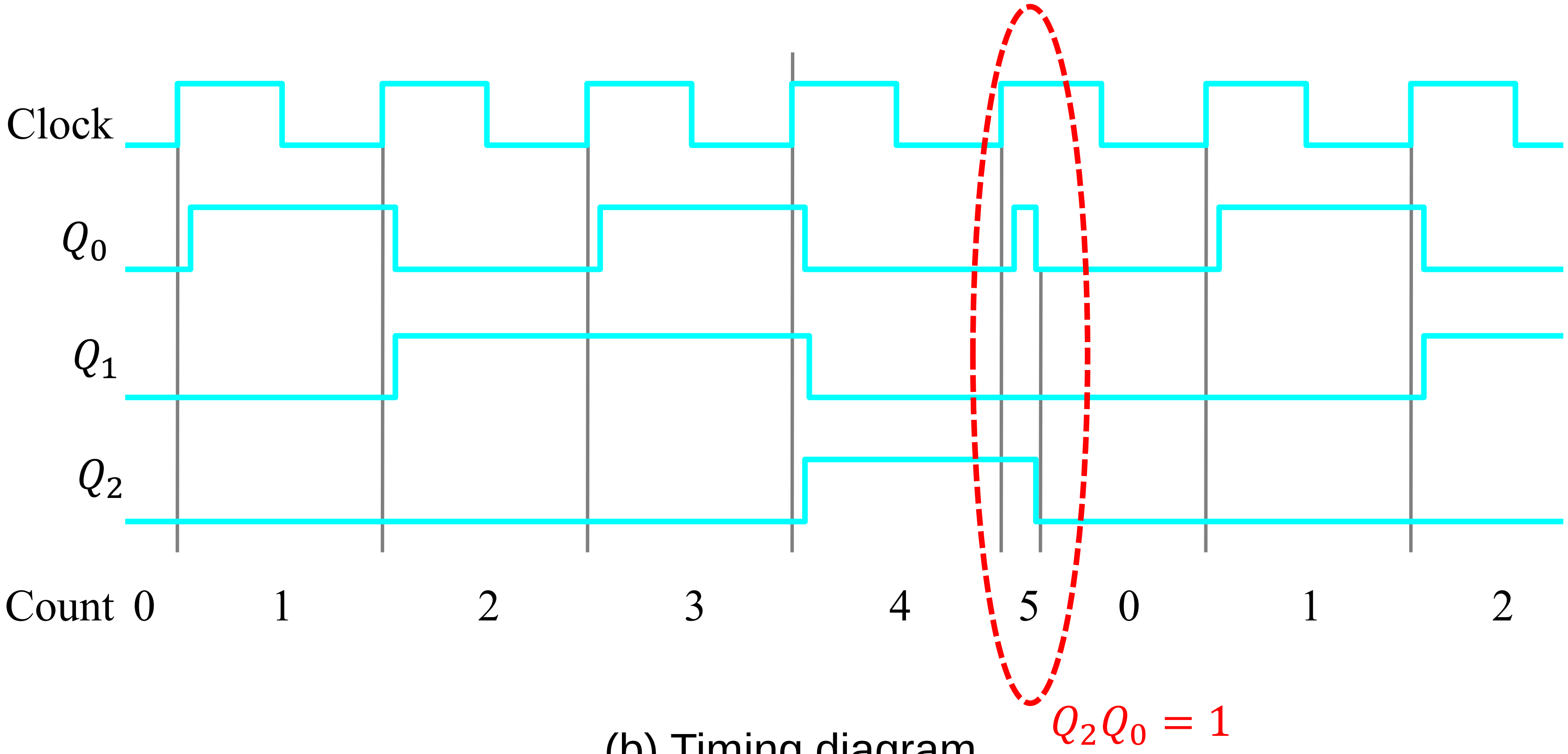
A modulo-6 counter with asynchronous reset



(a) Circuit

Figure 7.27. A modulo-6 counter with asynchronous reset.

A modulo-6 counter with asynchronous reset



(b) Timing diagram

Figure 7.27. A modulo-6 counter with asynchronous reset.

OTHER TYPES OF COUNTERS

BCD counter

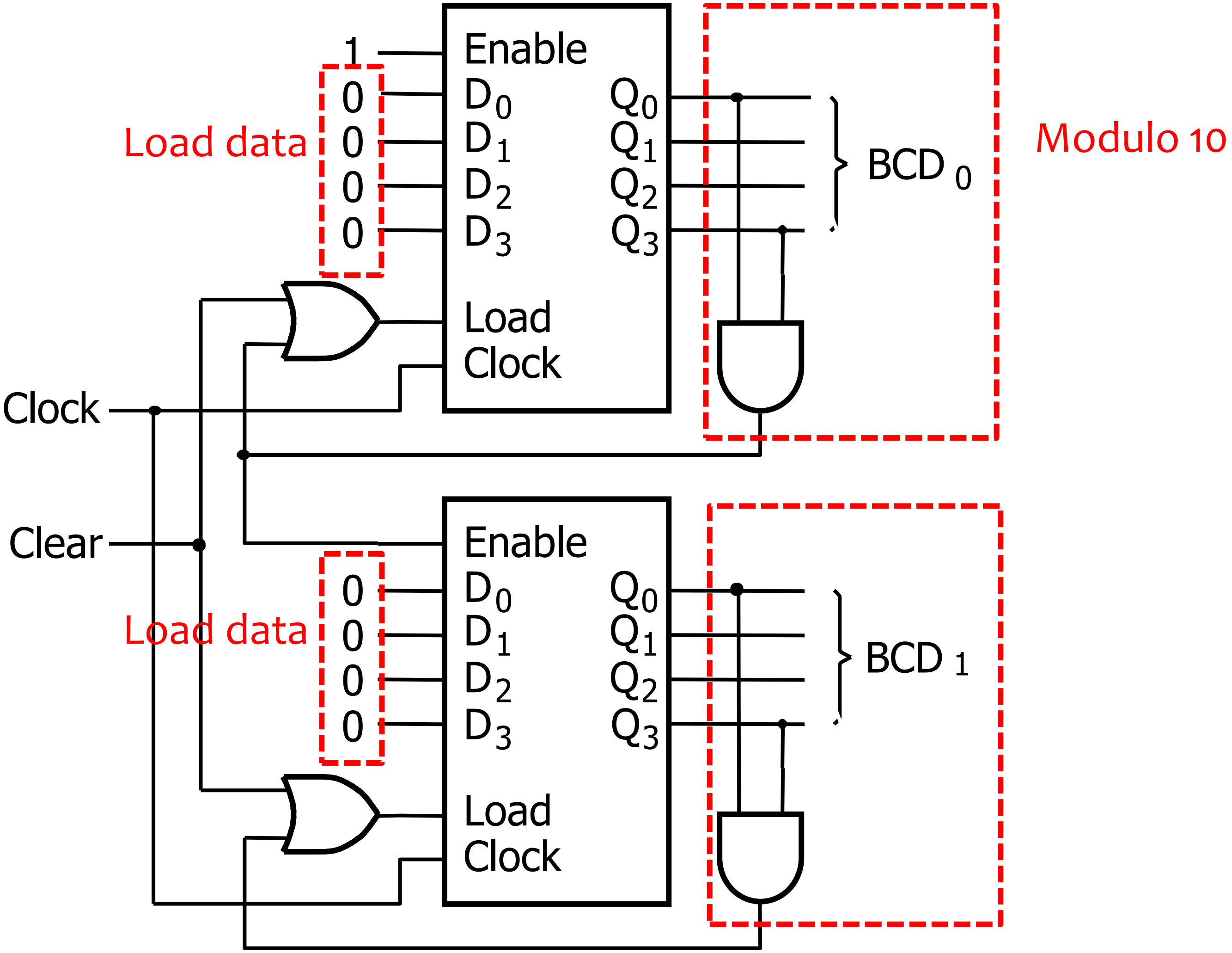
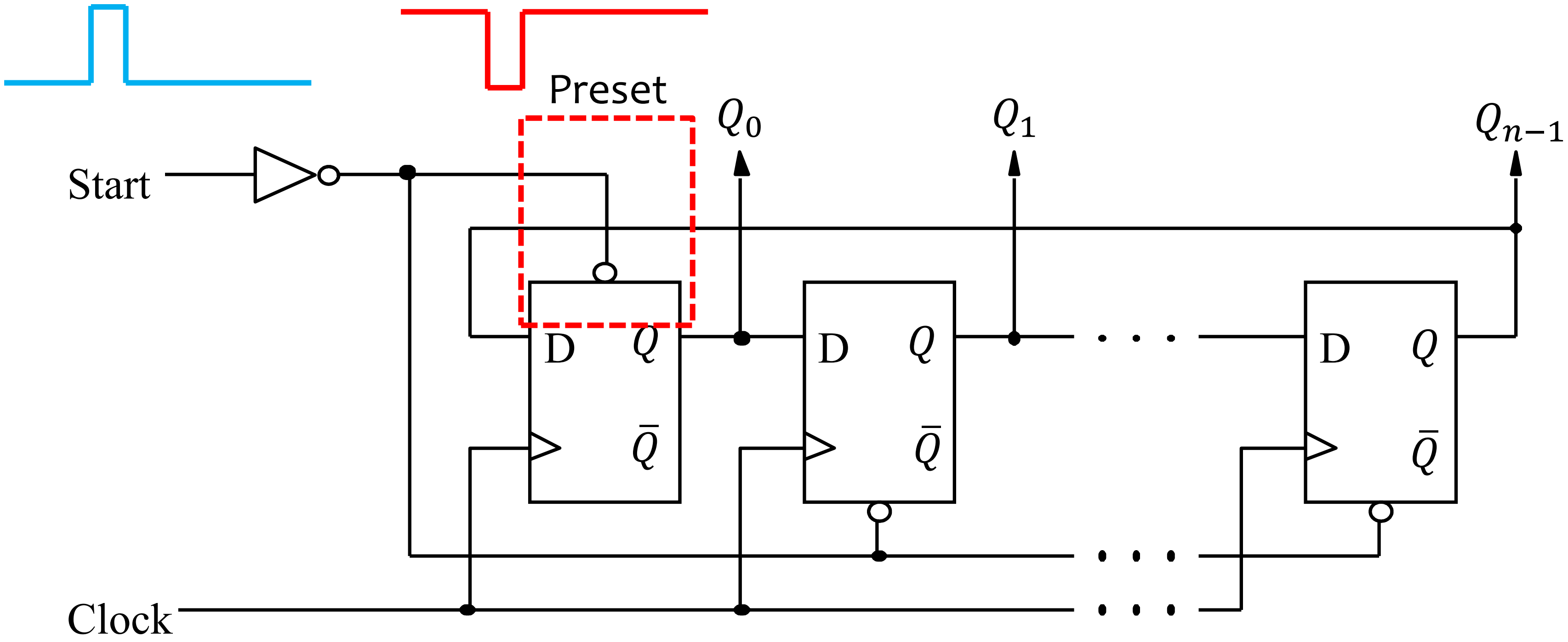


Figure 7.28. A two-digit BCD counter.

Ring counter



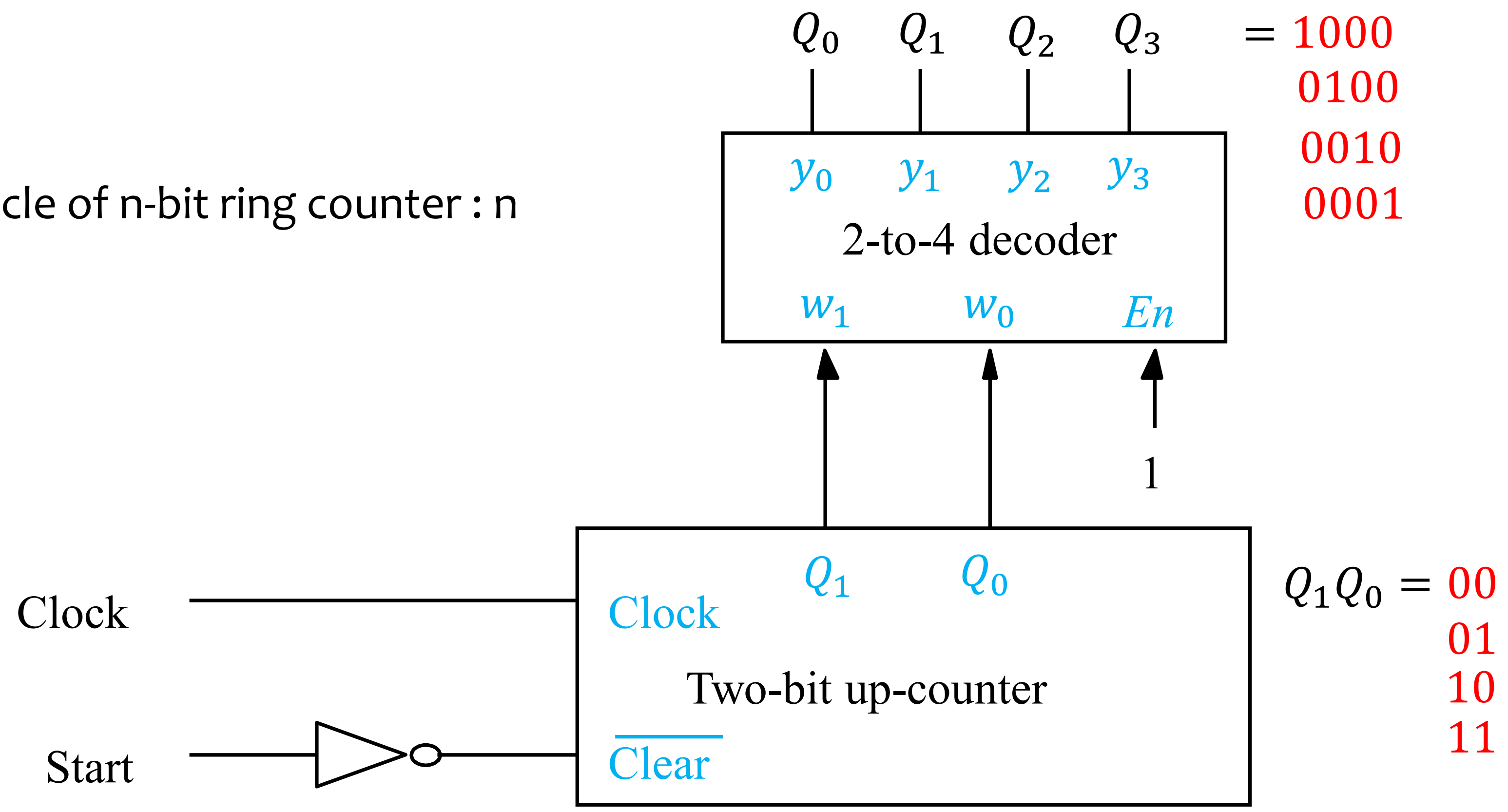
Initial state of n-bit counter: $Q_0Q_1 \cdots Q_{n-1} = 10 \cdots 0$

(a) An n-bit ring counter

Figure 7.29. Ring counter.

Ring counter

Cycle of n-bit ring counter : n



(b) A four-bit ring counter

Figure 7.29. Ring counter.

Johnson counter

Initial state of n-bit Johnson counter: $Q_0Q_1 \cdots Q_{n-1} = 00 \cdots 0$

$Q_0Q_1 \cdots Q_{n-1} = 00 \cdots 0$

$Q_0Q_1 \cdots Q_{n-1} = 10 \cdots 0$

$Q_0Q_1 \cdots Q_{n-1} = 11 \cdots 0$

$\vdots$

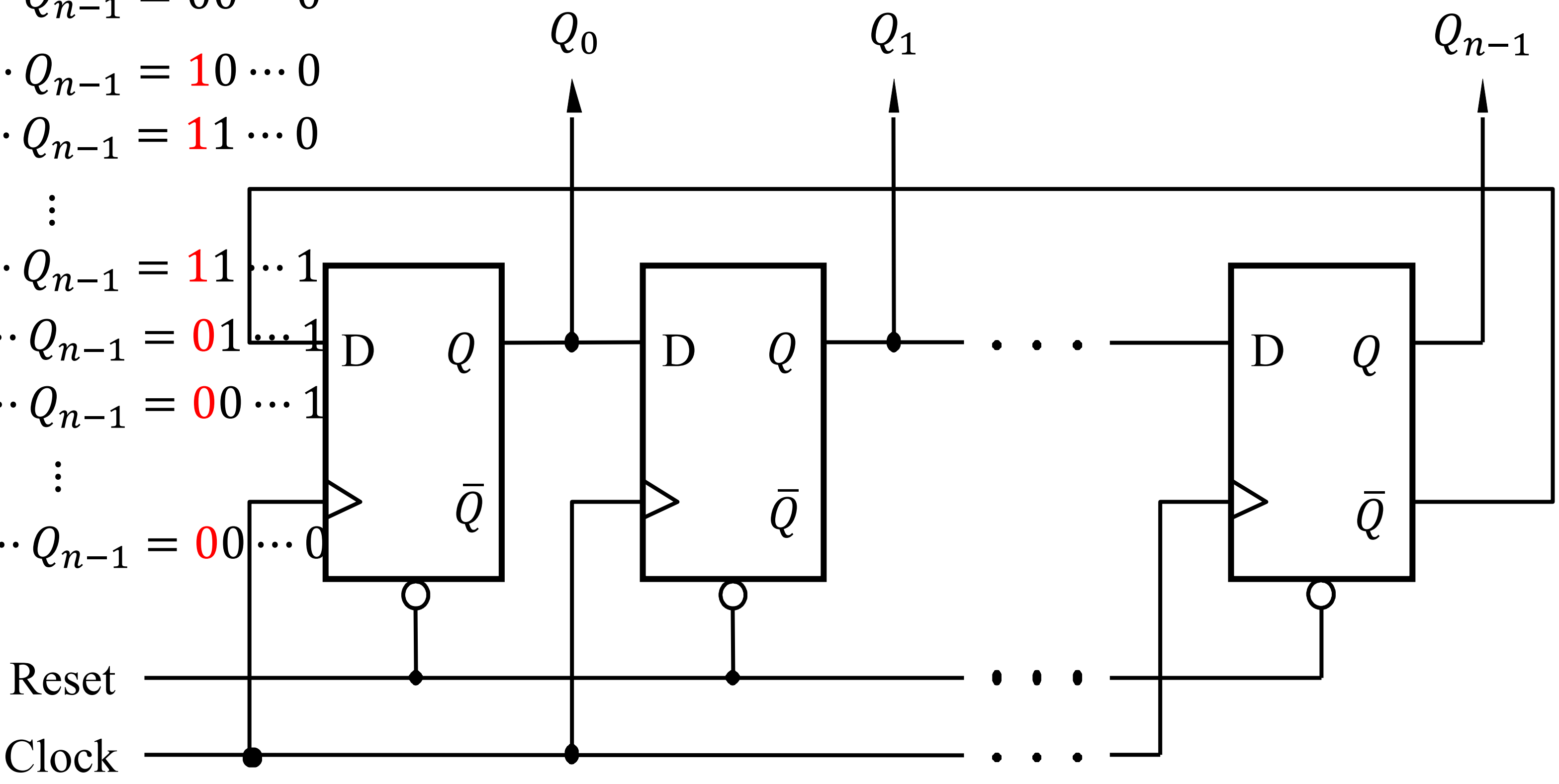
$Q_0Q_1 \cdots Q_{n-1} = 11 \cdots 1$

$Q_0Q_1 \cdots Q_{n-1} = 01 \cdots 1$

$Q_0Q_1 \cdots Q_{n-1} = 00 \cdots 1$

$\vdots$

$Q_0Q_1 \cdots Q_{n-1} = 00 \cdots 0$



Cycle of n-bit Johnson counter : 2n

Figure 7.30. Johnson counter.

DESIGN EXAMPLES

$R1_{in}, R2_{in}, \cdots Rk_{in}$: control signals when data is loaded into each register

$R1_{out}, R2_{out}, \cdots Rk_{out}$: only one of tri-state buffer enable control signal is asserted at a given time.

$R1$ to Rk : n-bit registers

: n-bit bus

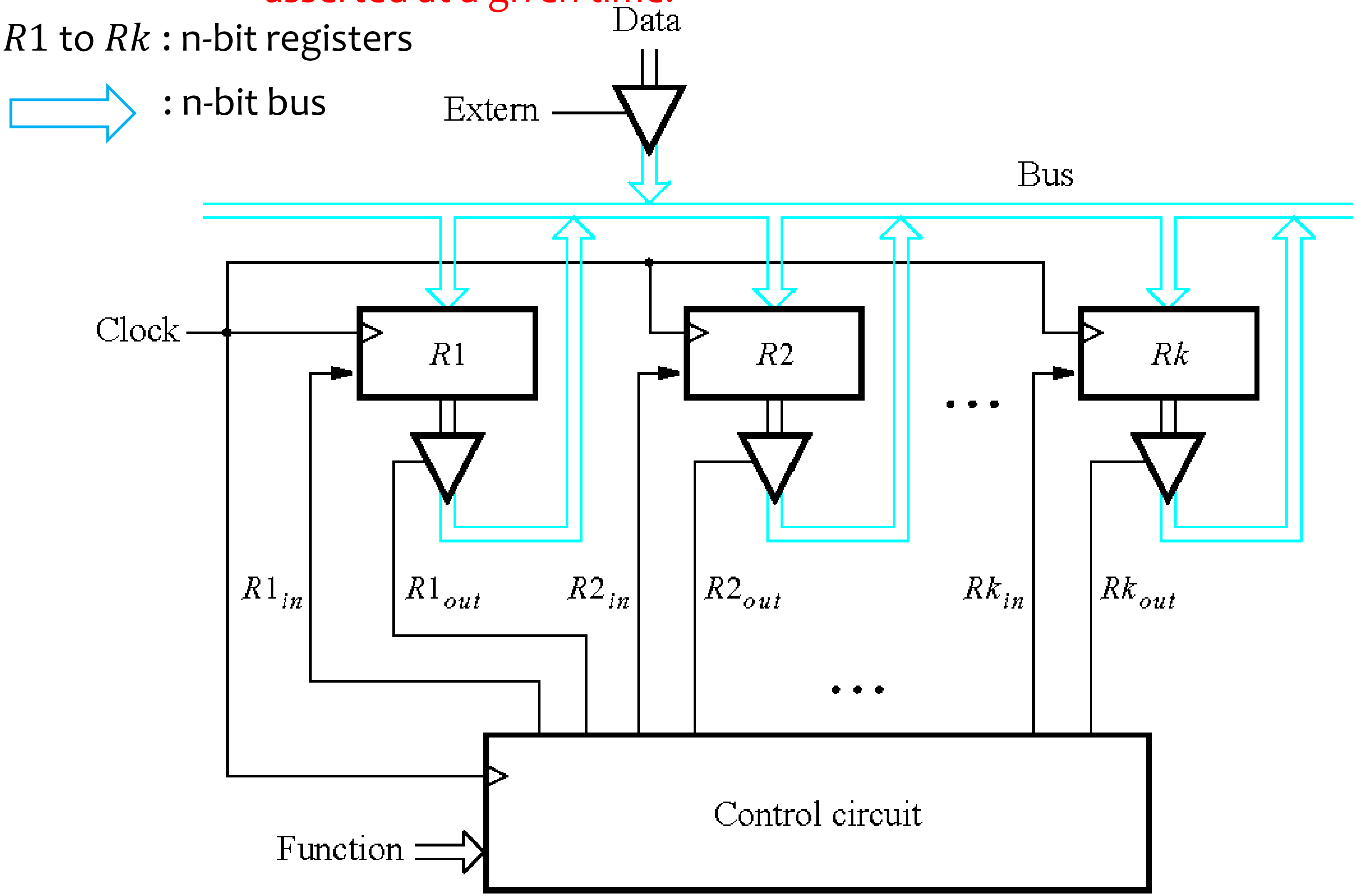
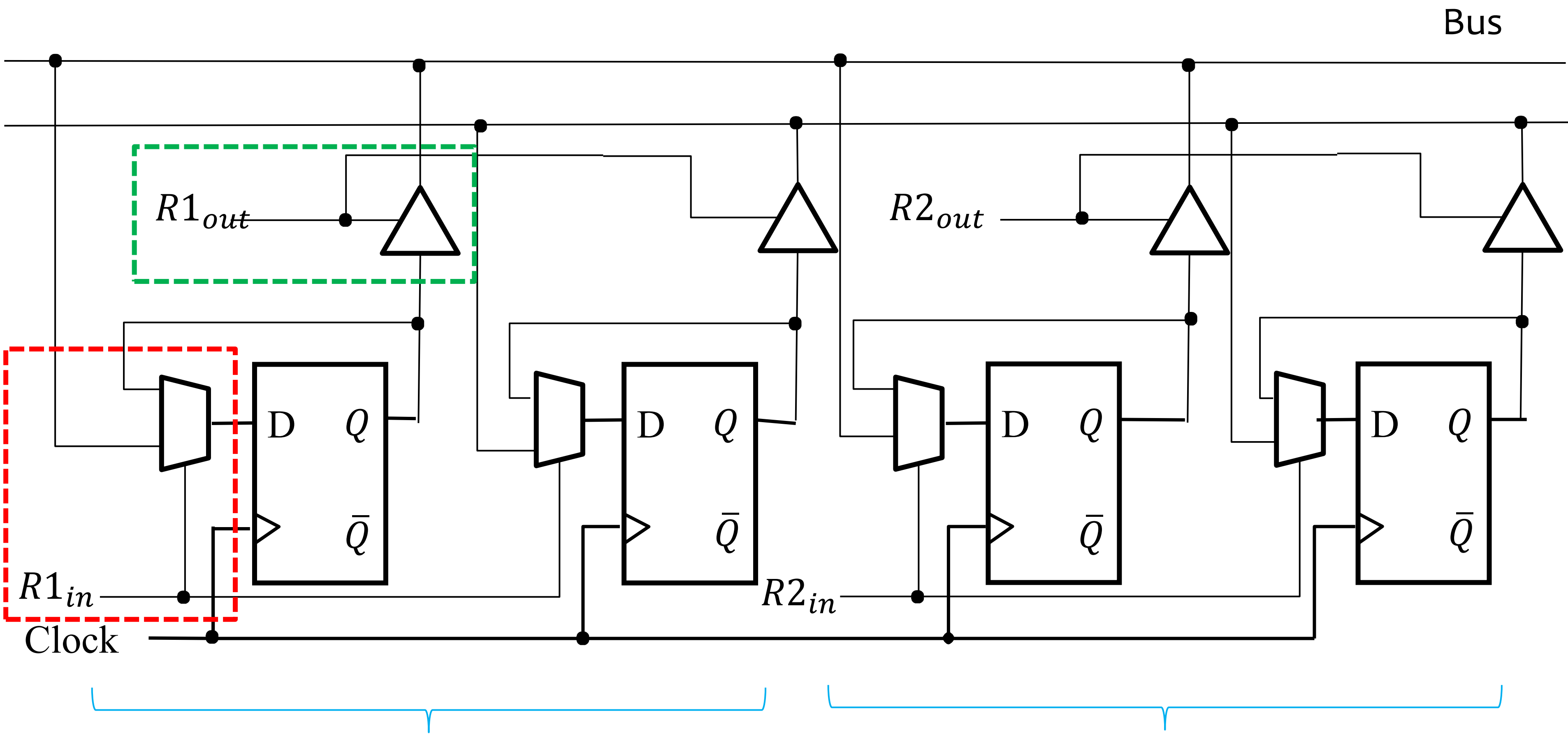


Figure 7.55. A digital system with k registers

Bus Structure



$R1_{in} = 0$: Remain present state

$R1_{in} = 1$: Read data bus

$R1_{out} = 0$: disconnect bus

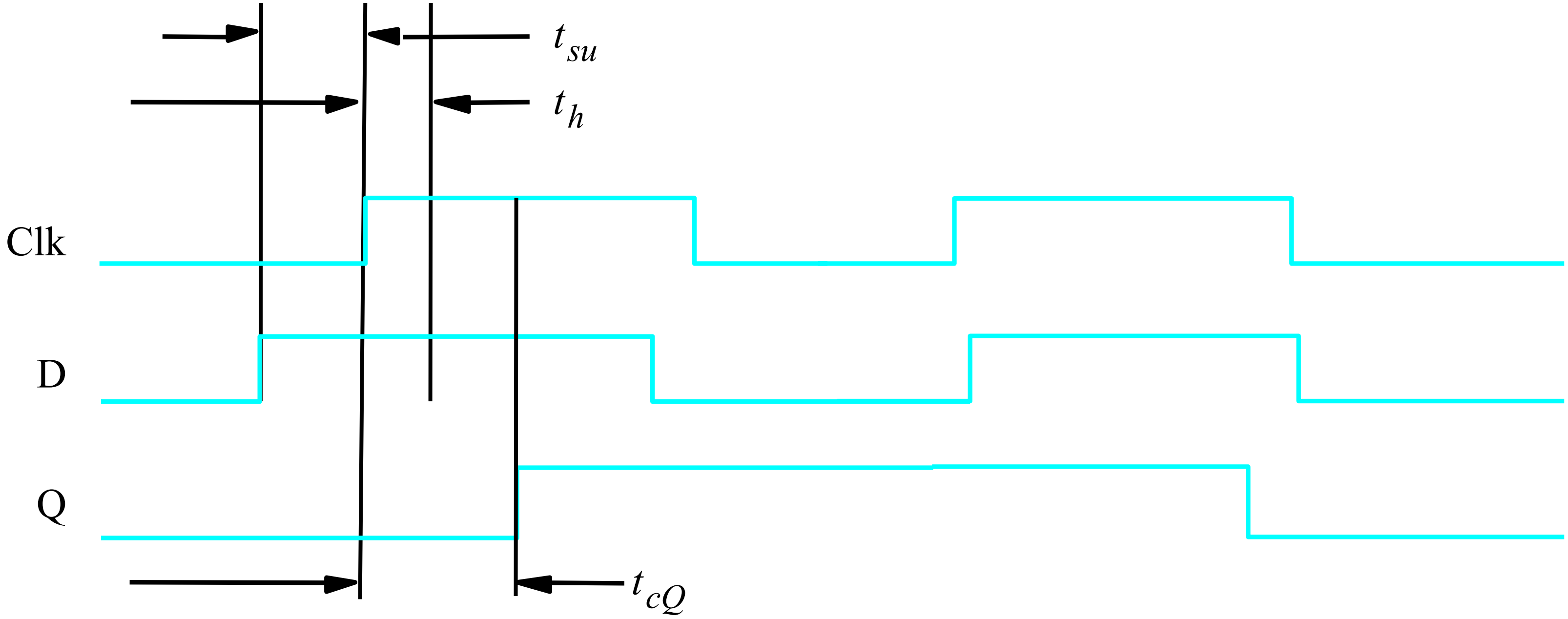
$R1_{out} = 1$: connect bus and push present state into bus

Figure 7.56. Details for connecting registers to a bus

TIMING ANALYSIS OF FLIP-FLOP CIRCUITS

Setup and hold times

The minimum time that the D signal must be stable prior to the positive edge of the clock signal is called the setup time t_{su} of the latch.



The minimum time that the D signal must remain stable after the positive edge of the clock signal is called the hold time t_h of the latch.

Figure 7.9. Setup and hold times.

Timing Analysis of Flip-Flop Circuits

We wish to calculate the Maximum clock frequency : F_{max}

Assume that the flip-flop timing parameters

$t_{su} = 0.6\, ns, t_h = 0.4\, ns\ and\ 0.8 \leq t_{cQ} \leq 1.0\, ns$

Maximum period of clock signal $T_{min} = 1/F_{max}$

$$T_{min} = t_{cQ} + t_{NOT} + t_{su} = 1.0 + 1.1 + 0.6 = 2.7\, ns$$

$$F_{max} = \frac{1}{2.7} ns = 370.37\, MHz$$

Assume that logic gate propagation delay$= 1 + 0.1k$,where k is the number of inputs.

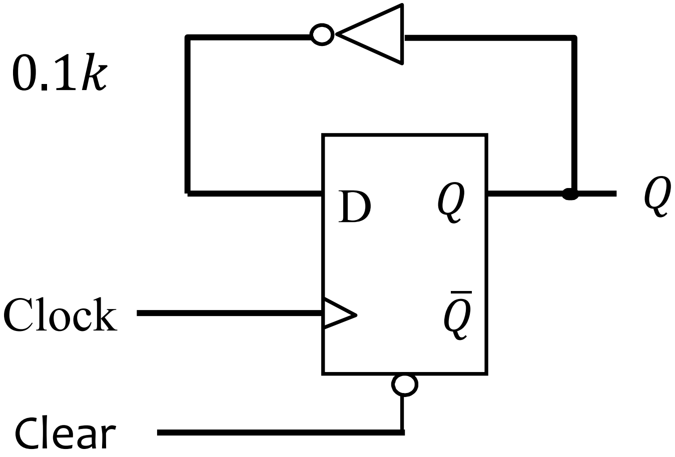


Figure 7.80. A simple flip-flop circuit.

Timing Analysis of Flip-Flop Circuits

We wish to calculate the maximum clock frequency for the counter.

Red line from Q_0 to Q_3 shows the longest path.

$$T_{min} = t_{cQ} + 3(t_{AND}) + t_{XOR} + t_{su}$$
$$= 1.0 + 3(1.2) + 1.2 + 0.6 = 6.4\,ns$$

$$F_{max} = \frac{1}{6.4}ns = 156.25\,MHz$$

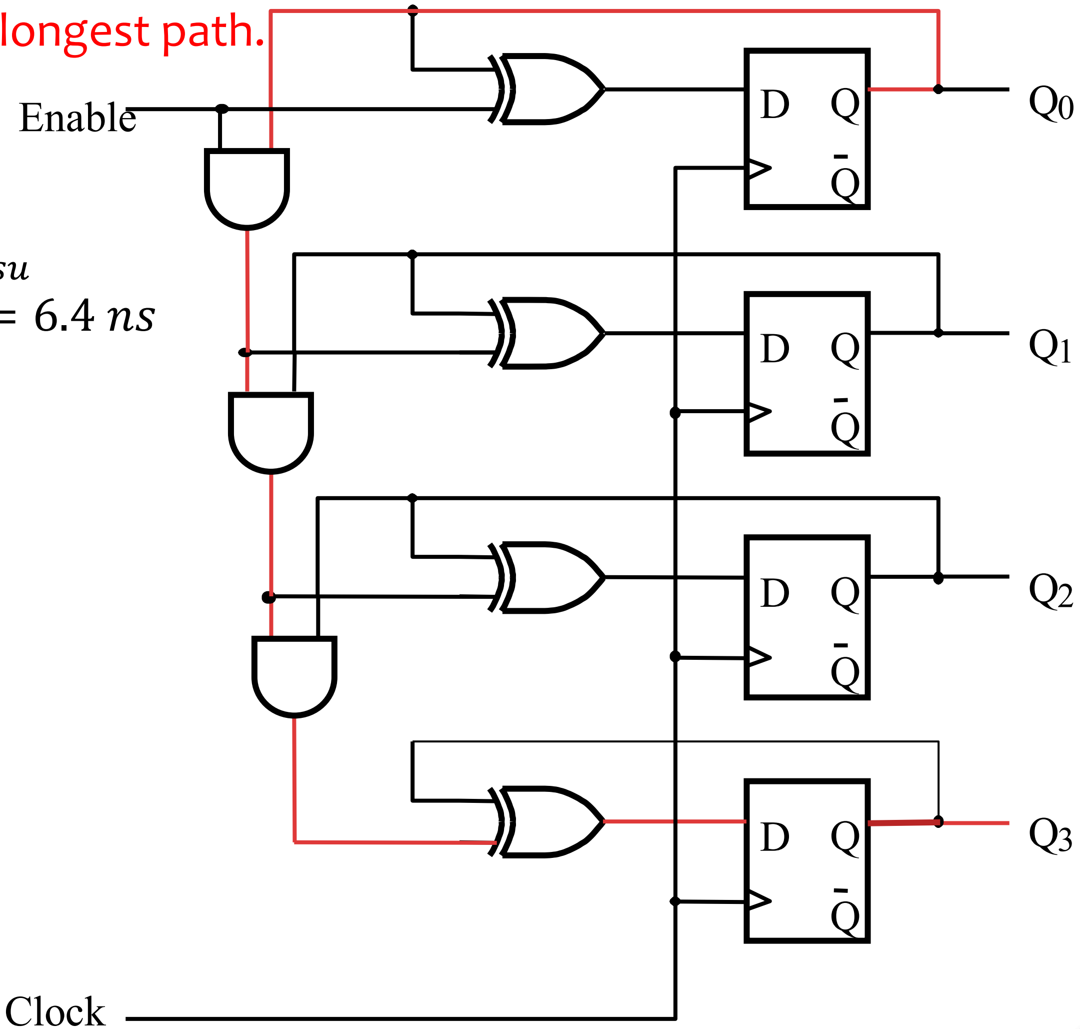
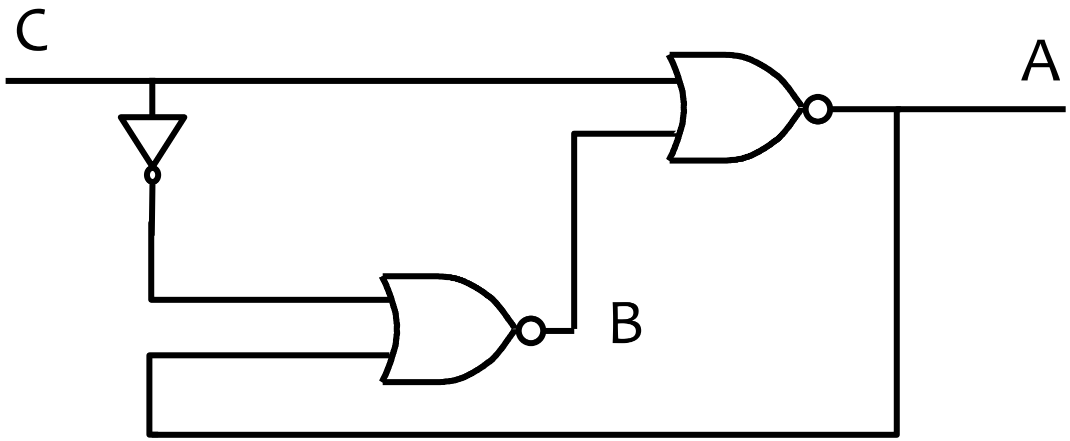


Figure 7.81. A 4-bit counter.

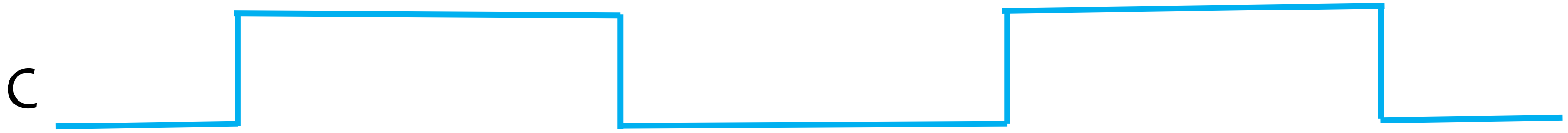
EXAMPLES OF SOLVED PROBLEMS

Example 7.13

Problem: Consider the circuit in Figure 7.82a. Assume that the input C is driven by a square wave signal with a 50% duty cycle. Draw a timing diagram that shows the waveforms at points A and B. Assume that the propagation delay through each Gate is Δ seconds.



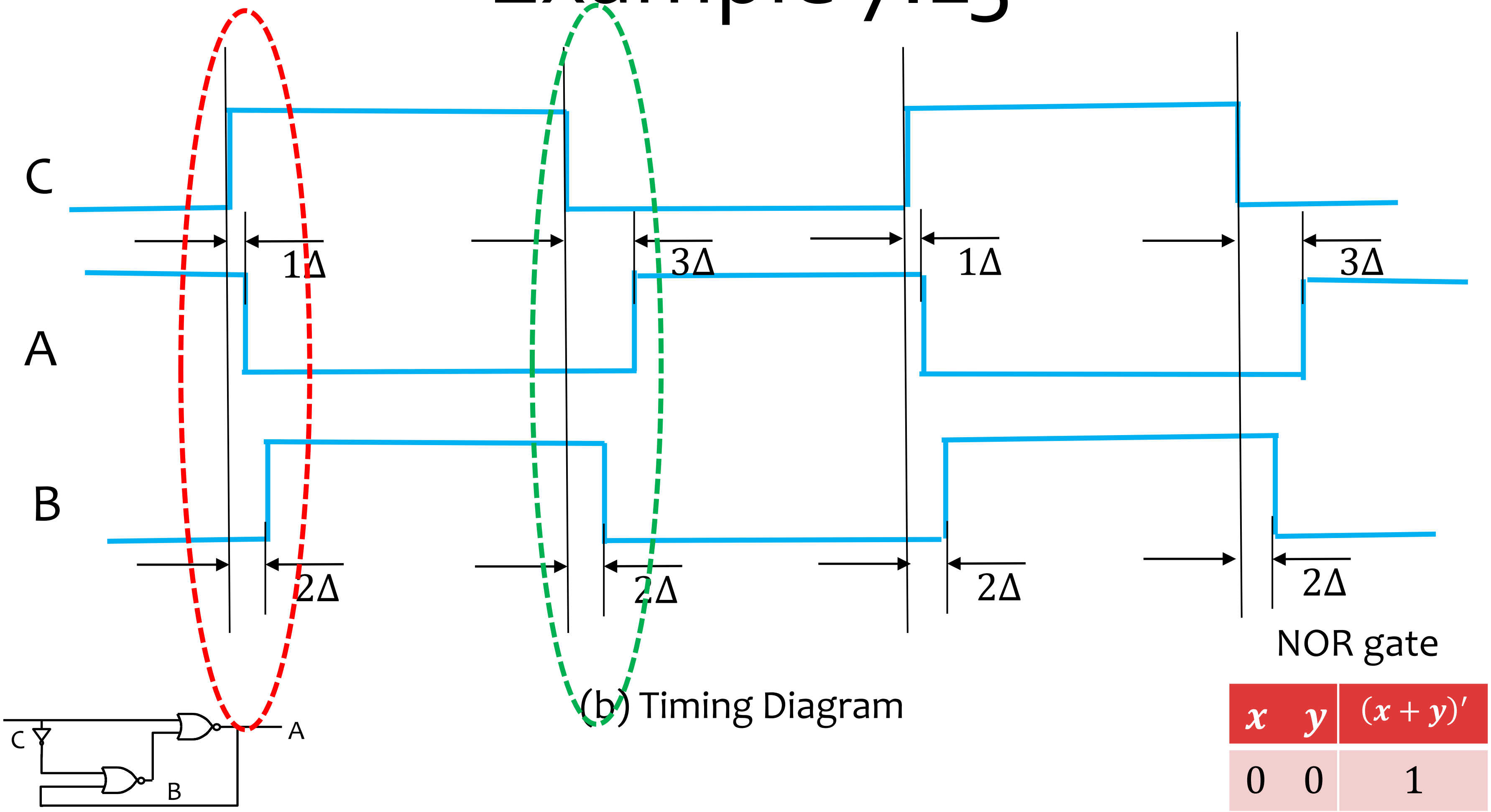
(a) Circuit



(b) Timing Diagram

Figure 7.82 Circuit for Example 7.13

Example 7.13



(a) Circuit

(b) Timing Diagram

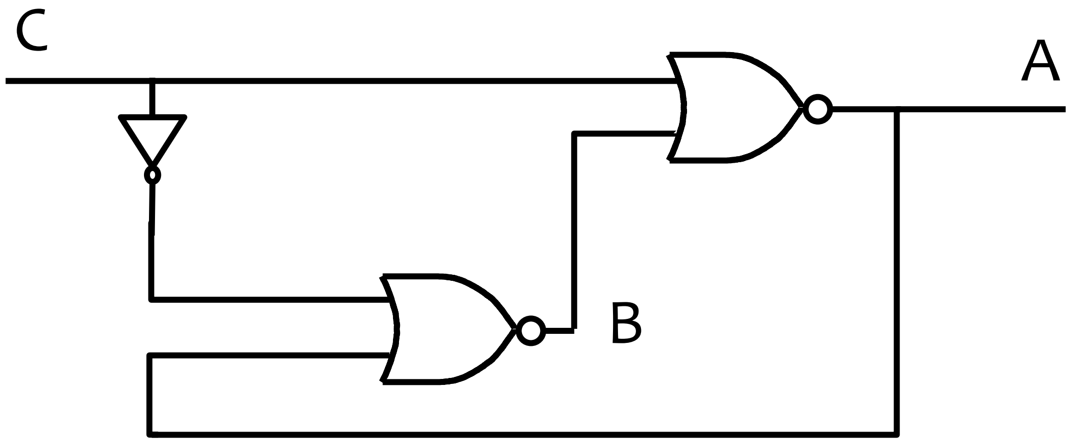
Figure 7.82 Circuit for Example 7.13

NOR gate

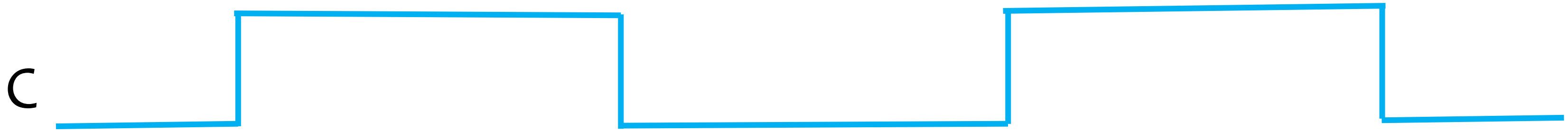
x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Example 7.13

Problem: Consider the circuit in Figure 7.82a. Assume that the input C is driven by a square wave signal with a 50% duty cycle. Draw a timing diagram that shows the waveforms at points A and B. Assume that the propagation delay through each Gate is Δ seconds.



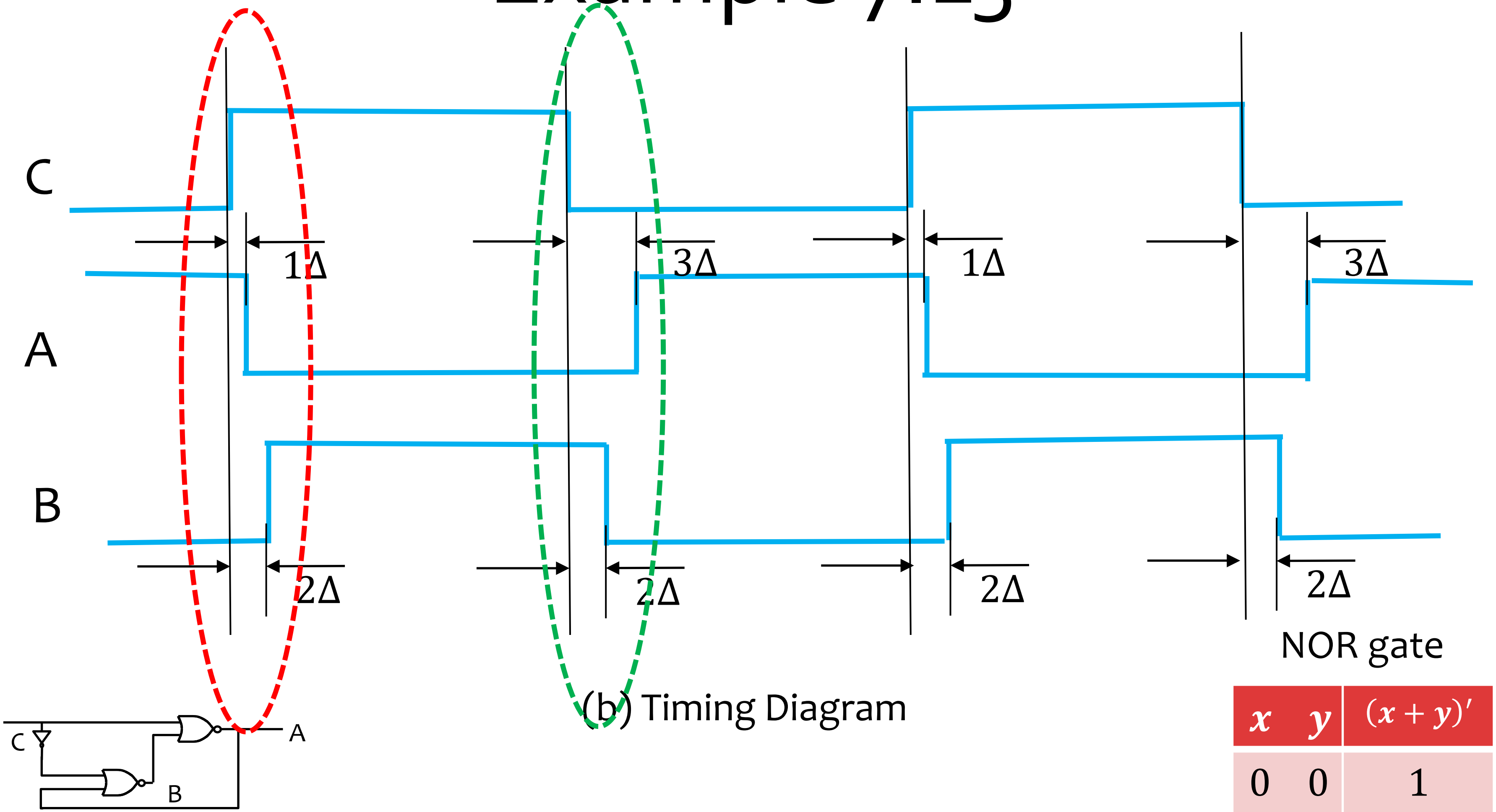
(a) Circuit



(b) Timing Diagram

Figure 7.82 Circuit for Example 7.13

Example 7.13



(b) Timing Diagram

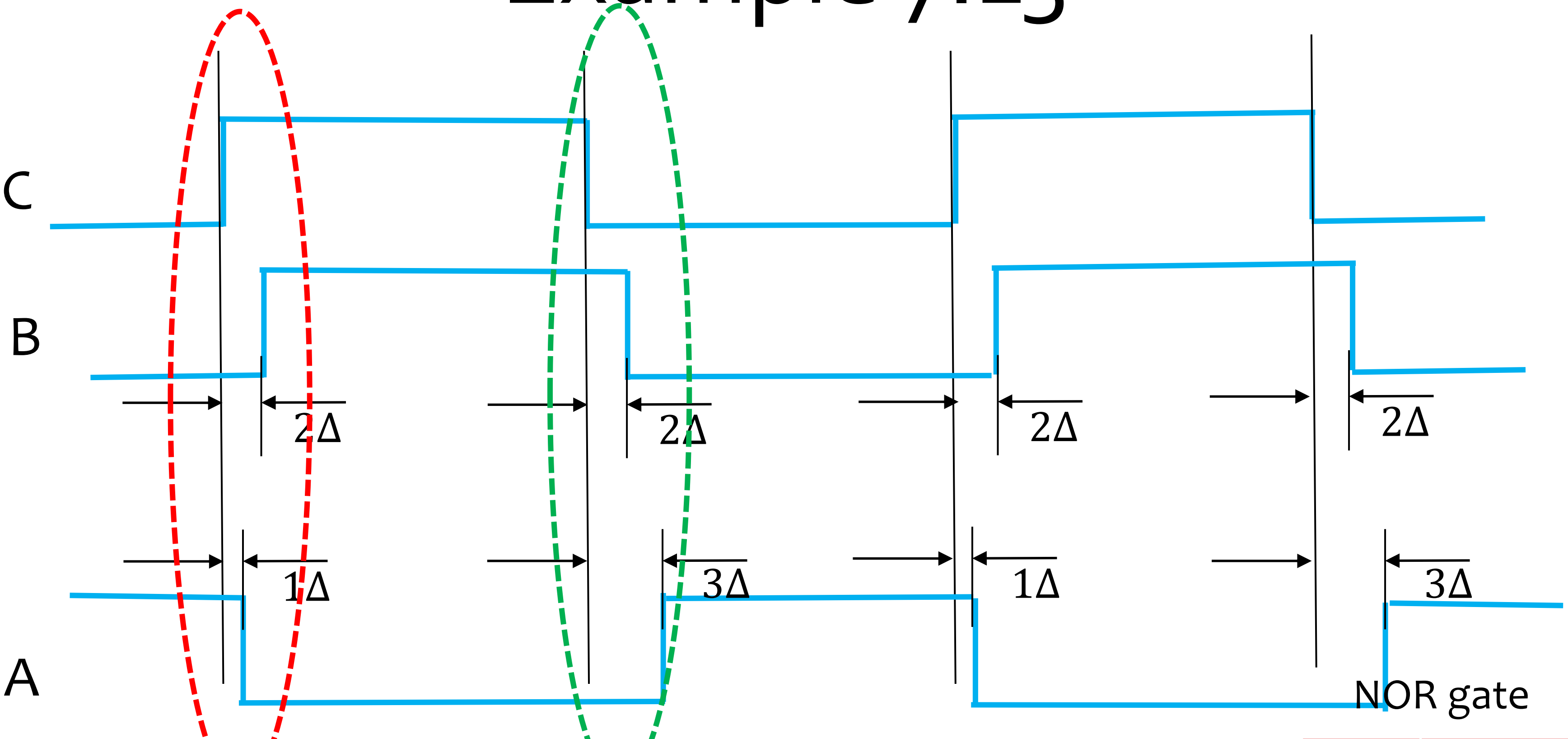
(a) Circuit

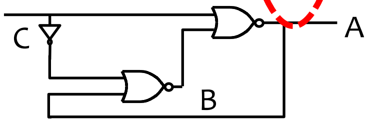
Figure 7.82 Circuit for Example 7.13

NOR gate

x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Example 7.13





(a) Circuit

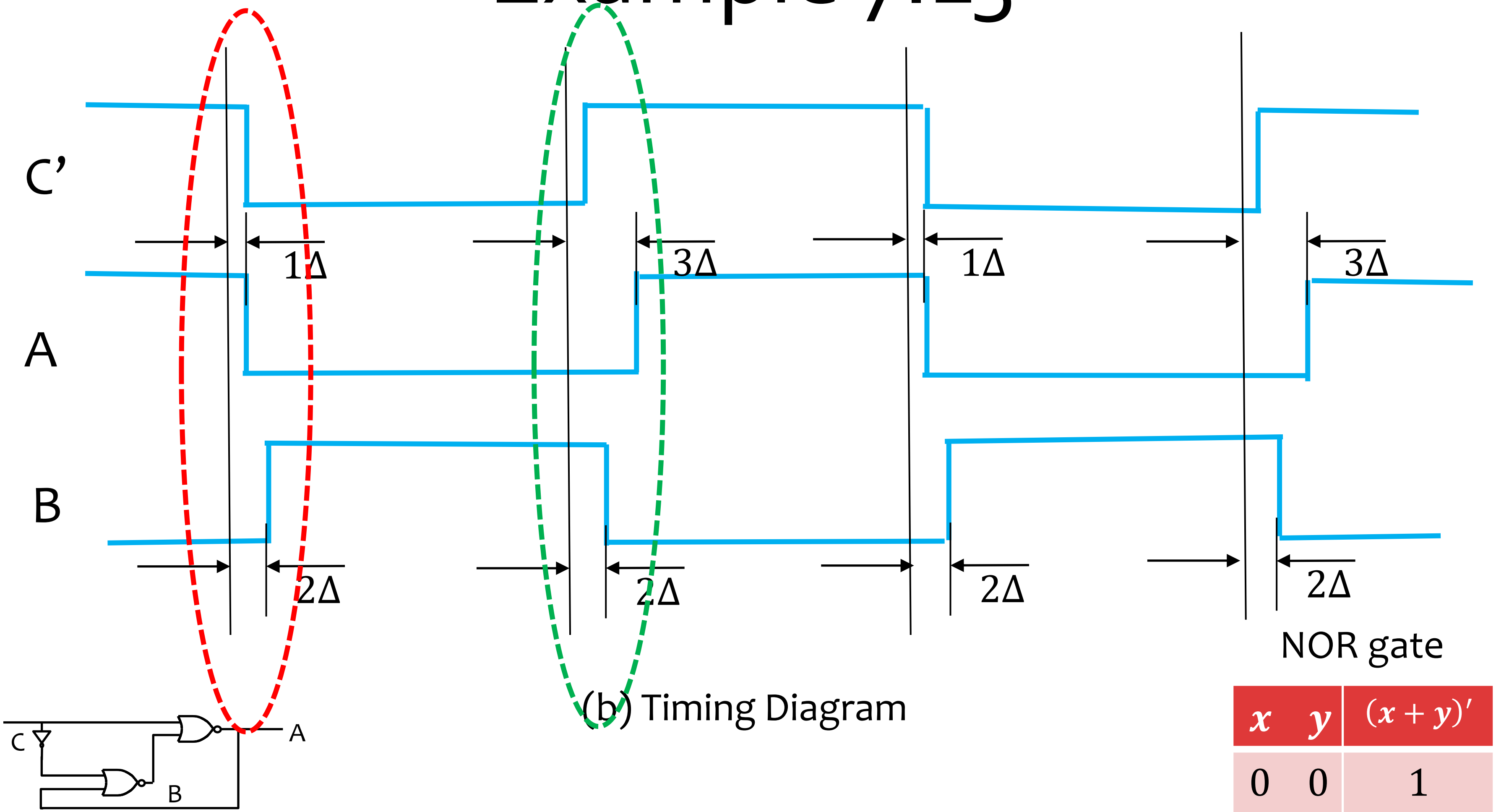
(b) Timing Diagram

Figure 7.82 Circuit for Example 7.13

NOR gate

x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Example 7.13



(b) Timing Diagram

(a) Circuit

Figure 7.82 Circuit for Example 7.13

NOR gate

x	y	$(x+y)'$
0	0	1
0	1	0
1	0	0
1	1	0

Example 7.14

Problem: Determine the functional behavior of the circuit in Figure 7.83. Assume that Input w is driven by a square wave signal.

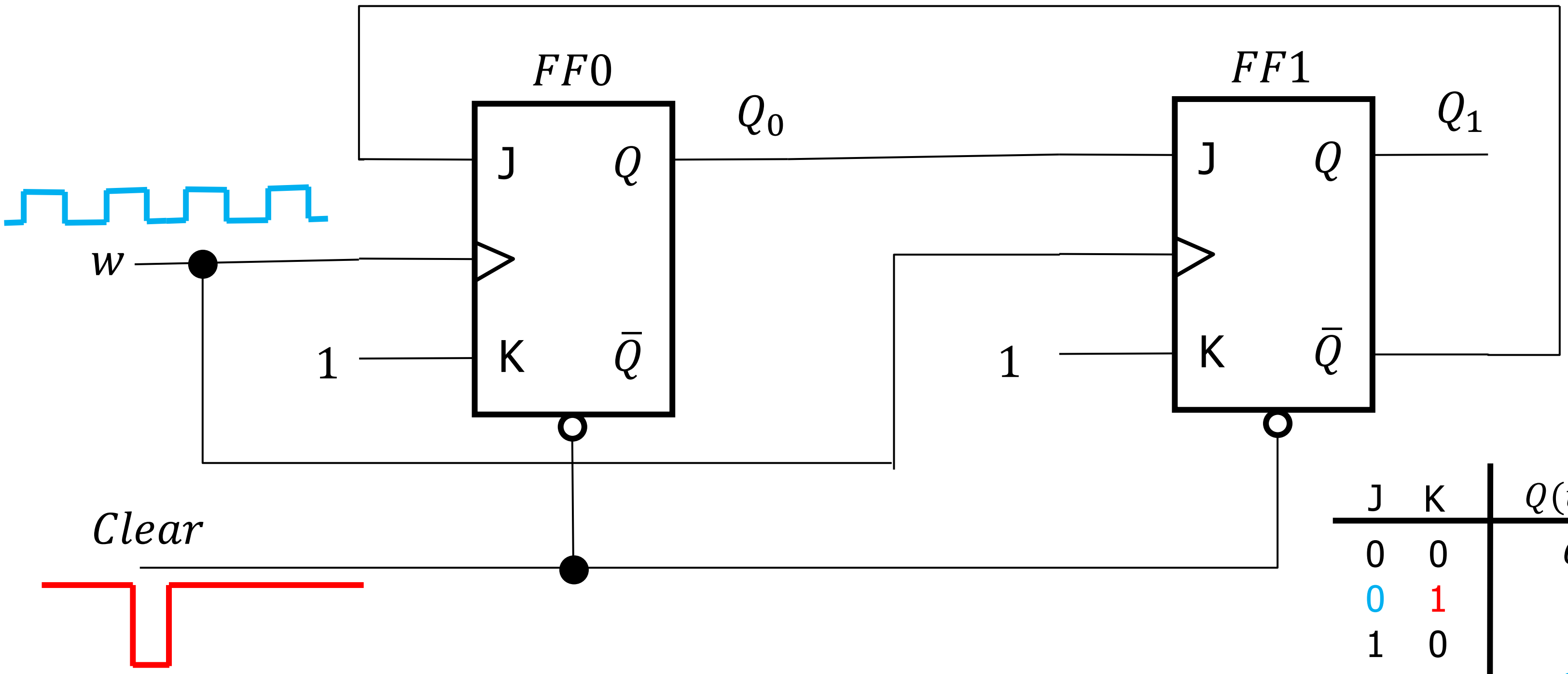


Figure 7.83 Circuit for Example 7.14

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

(b) Truth table

Example 7.14

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

Time interval	$FF0$			$FF1$		
	$J_0(Q_1')$	K_0	Q_0	$J_1(Q_0)$	K_1	Q_1
Clear	1	1	0	0	1	0
t_1	1	1	1	1	1	0
t_2	0	1	0	0	1	1
t_3	1	1	0	0	1	0
t_4	1	1	1	1	1	0

inverting

Figure 7.84 Summary of the behavior of the circuit in Figure 7.83

Example 7.15

Problem: Figure 7.70 shows a circuit that generates four timing control signals T_0, T_1, T_2, and T_3. Design a circuit that generates six signals T_0 to T_5.

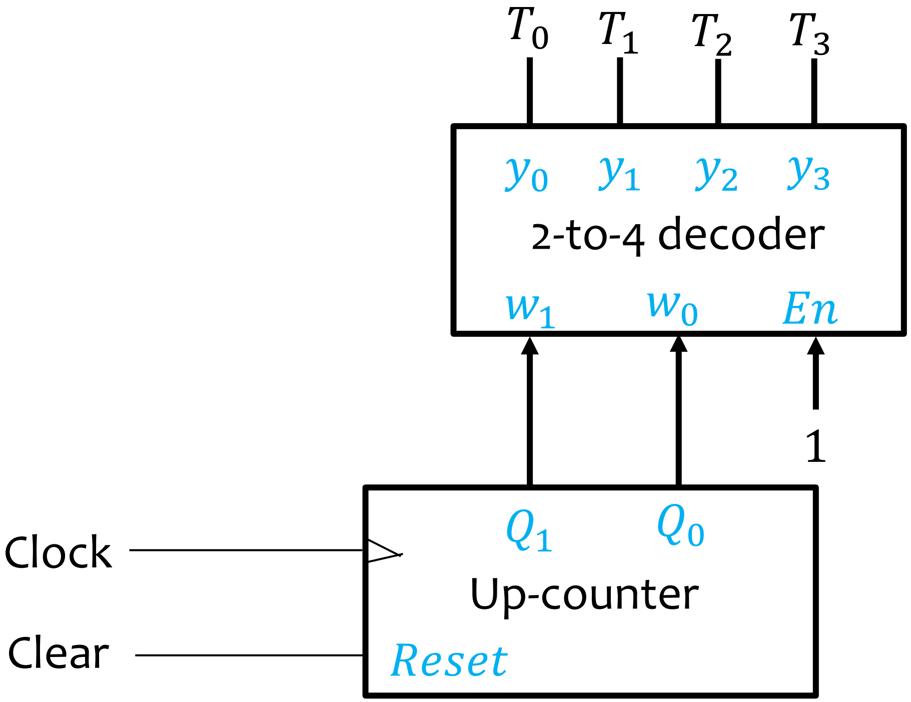
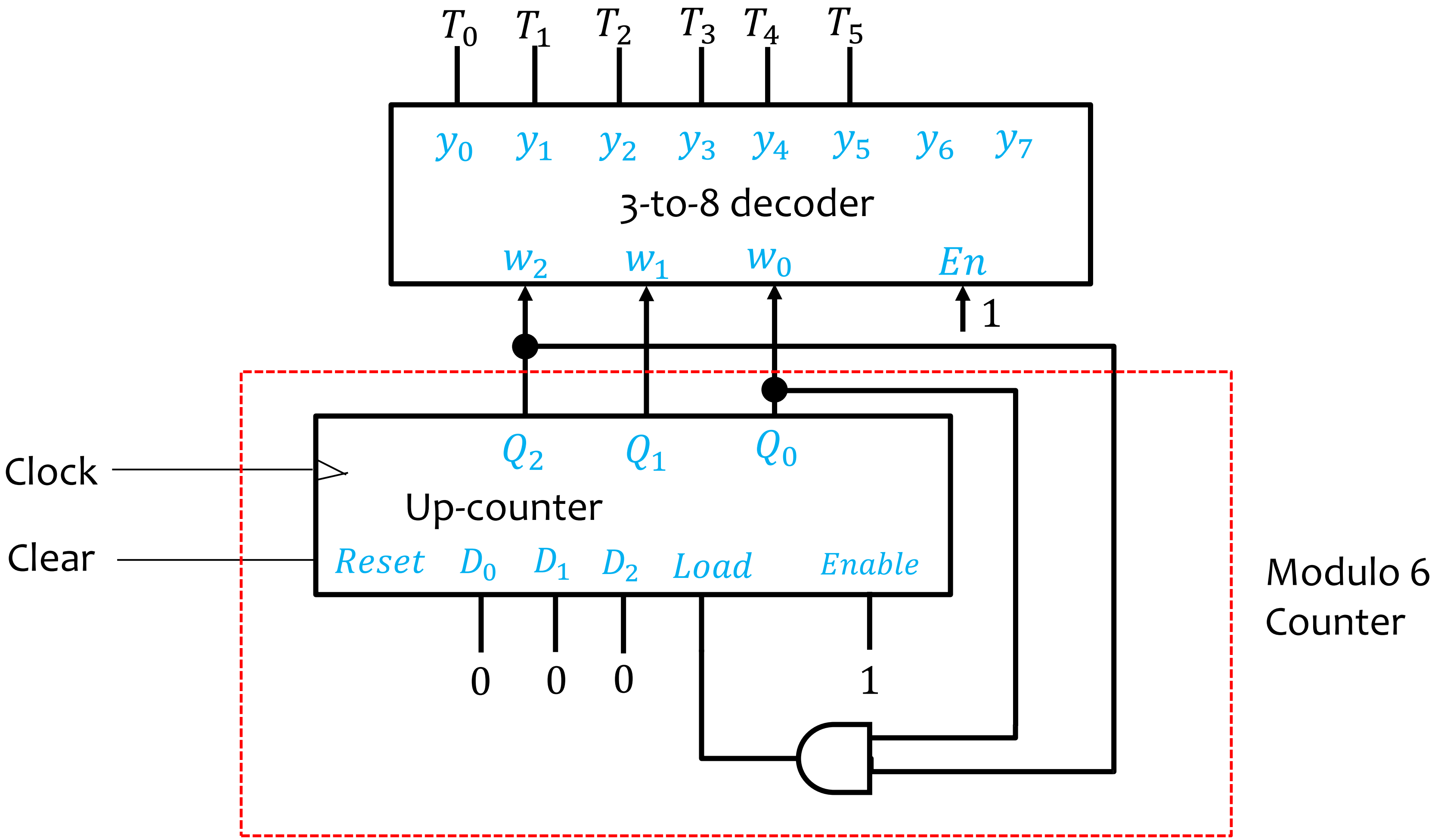


Figure 7.70 A port of the control for the processor

Example 7.15


T_0
T_1
T_2
T_3
T_4
T_5
y_0
y_1
y_2
y_3
y_4
y_5
y_6
y_7
3-to-8 decoder
w_2
w_1
w_0
En
1
Clock
Clear
Q_2
Q_1
Q_0
Up-counter
$Reset$
D_0
D_1
D_2
$Load$
$Enable$
0
0
0
1
Modulo 6
Counter

Example 7.15

Alternative design: 3-bit Johnson counter

Clock Cycle	Q_0	Q_1	Q_2	$Control\ signal$
0	0	0	0	$T_0 = Q_0'Q_2'$
1	1	0	0	$T_1 = Q_0Q_1'$
2	1	1	0	$T_2 = Q_1Q_2'$
3	1	1	1	$T_3 = Q_0Q_2$
4	0	1	1	$T_4 = Q_0'Q_1$
5	0	0	1	$T_5 = Q_1'Q_2$

$T_0 = Q_0'Q_1'Q_2'$

$010, 101: don't\ care\ term$

Example 7.15

$T_0 = Q_0'Q_2'$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	1	d	0	0
1	0	0	0	d

$T_1 = Q_0Q_1'$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	0	d	0	1
1	0	0	0	d

$T_2 = Q_1Q_2'$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	0	d	1	0
1	0	0	0	d

$T_3 = Q_0Q_2$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	0	d	0	0
1	0	0	1	d

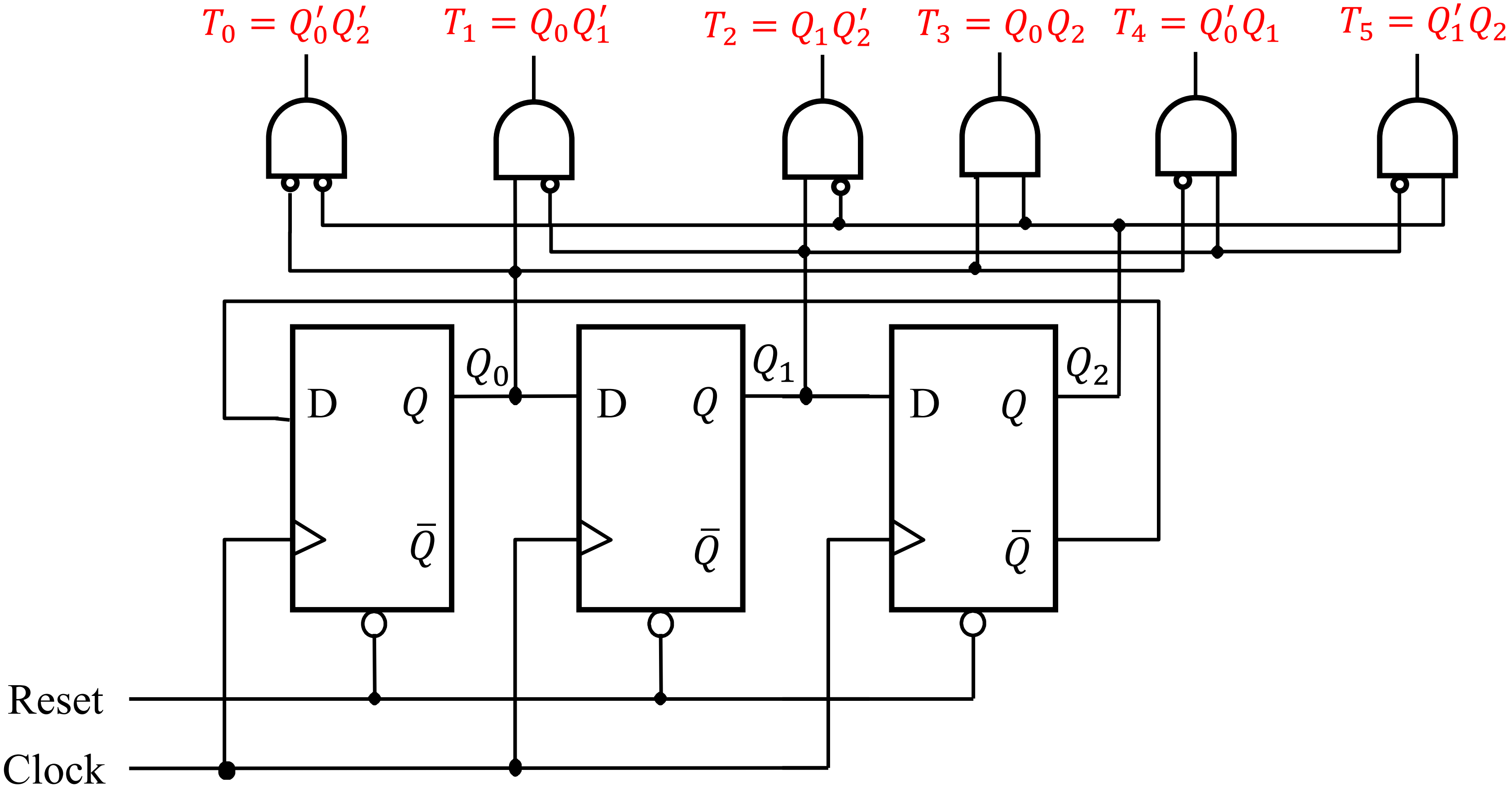
$T_4 = Q_0'Q_1$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	0	d	0	0
1	0	1	0	d

$T_5 = Q_1'Q_2$

$Q_2 \backslash Q_0Q_1$	00	01	11	10
0	0	d	0	0
1	1	0	0	d

Example 7.15


$T_0 = Q_0'Q_2'$
$T_1 = Q_0Q_1'$
$T_2 = Q_1Q_2'$
$T_3 = Q_0Q_2$
$T_4 = Q_0'Q_1$
$T_5 = Q_1'Q_2$
D
Q
$\bar{Q}$
Q_0
D
Q
$\bar{Q}$
Q_1
D
Q
$\bar{Q}$
Q_2
Reset
Clock

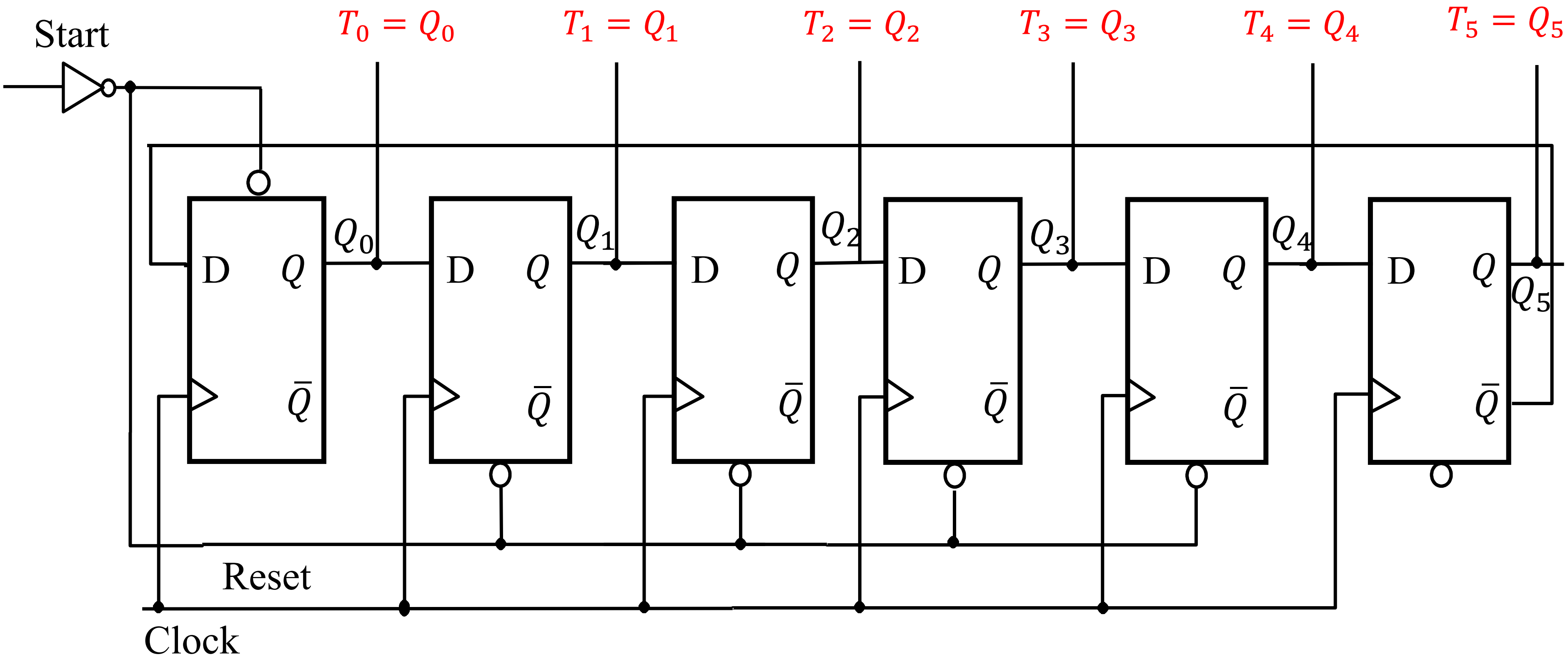
Example 7.15

Alternative design: 6-bit Ring counter

Clock Cycle	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5	$Control\ signal$
0	1	0	0	0	0	0	$T_0 = Q_0$
1	0	1	0	0	0	0	$T_1 = Q_1$
2	0	0	1	0	0	0	$T_2 = Q_2$
3	0	0	0	1	0	0	$T_3 = Q_3$
4	0	0	0	0	1	0	$T_4 = Q_4$
5	0	0	0	0	0	1	$T_5 = Q_5$

Example 7.15

Alternative design: 6-bit Ring counter

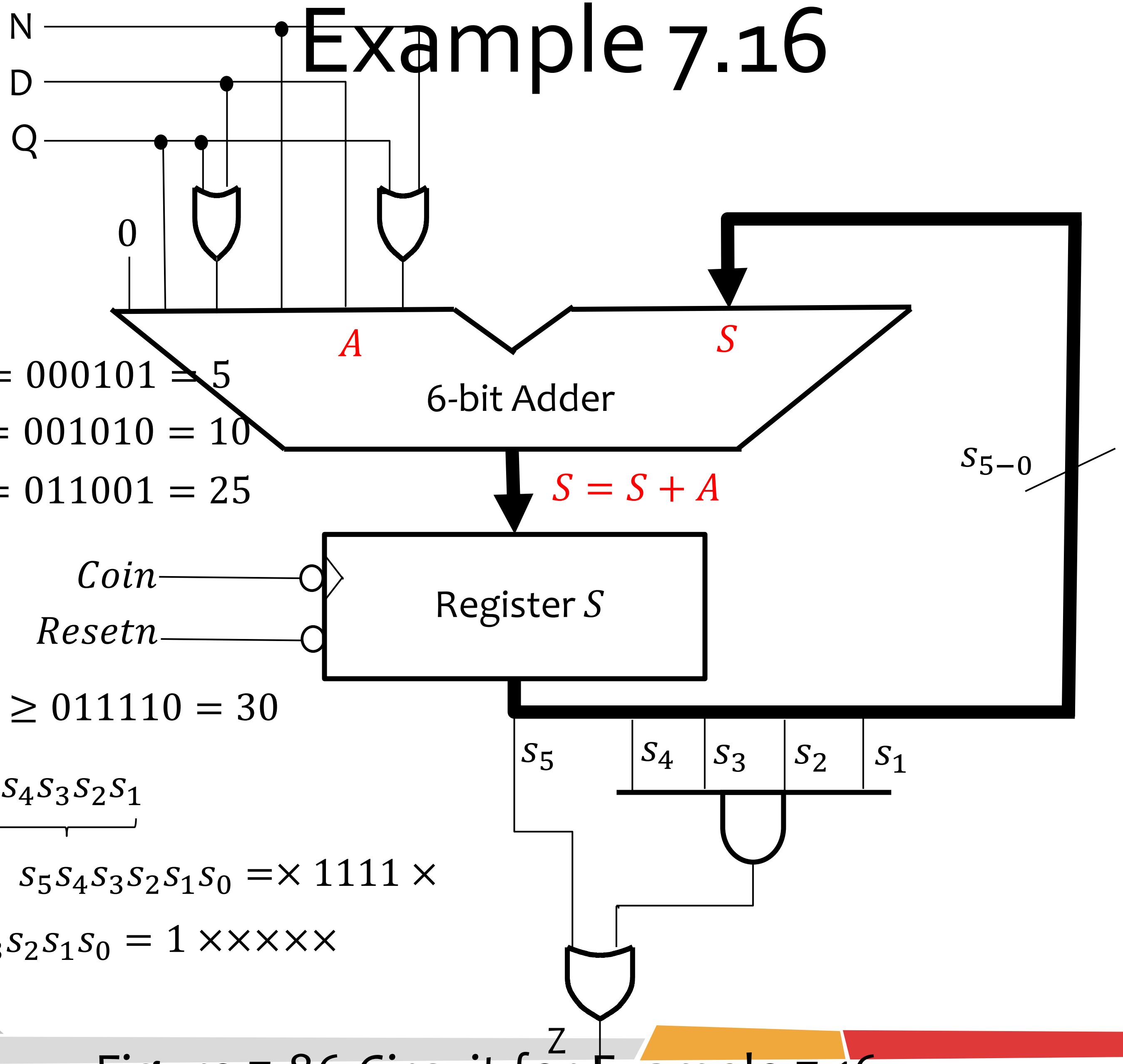


Example 7.16

Problem: Design a circuit that can be used to control a vending machine. The circuit has Five inputs: Q (quarter), D (dime), N (nickel), $Coin$, and $Resetn$. When a coin is deposited In the machine, a coin-sensing mechanism generates a pulse on the appropriate input (Q, D, or N). To signify the occurrence of the event, the mechanism also generates a pulse on the line $Coin$. The circuit is reset by using the $Resetn$ signal (active low). When at least 30 cents has been deposited, the circuit activates the output, Z. No change is given if the amount exceeds 30 cents.

Design the required circuit by using the following components: a six-bit adder, a six-bit register, and any number of AND, OR, and NOT gates.

Example 7.16



$N = 1, A = 000101 = 5$

$D = 1, A = 001010 = 10$

$Q = 1, A = 011001 = 25$

$Z = 1, \text{if } S \geq 011110 = 30$

$$Z = \underbrace{s_5}_{} + \underbrace{s_4 s_3 s_2 s_1}_{}$$

$s_5 s_4 s_3 s_2 s_1 s_0 = \times\, 1111 \times$

$s_5 s_4 s_3 s_2 s_1 s_0 = 1 \times\times\times\times\times$

Figure 7.86 Circuit for Example 7.16

Example 7.18

Problem: In section 7.15 we presented a timing analysis for the counter circuit in Figure 7.81. Redesign this circuit to reduce the logic delay between flip-flops, so that the circuit can operate at a higher maximum clock frequency.

Timing Analysis of Flip-Flop Circuits

We wish to calculate the maximum clock frequency for the counter.

Red line from Q_0 to Q_3 shows the longest path.

$$T_{min} = t_{cQ} + 3(t_{AND}) + t_{XOR} + t_{su}$$
$$= 1.0 + 3(1.2) + 1.2 + 0.6 = 6.4\ ns$$

$$F_{max} = \frac{1}{6.4ns} = 156.25\ MHz$$

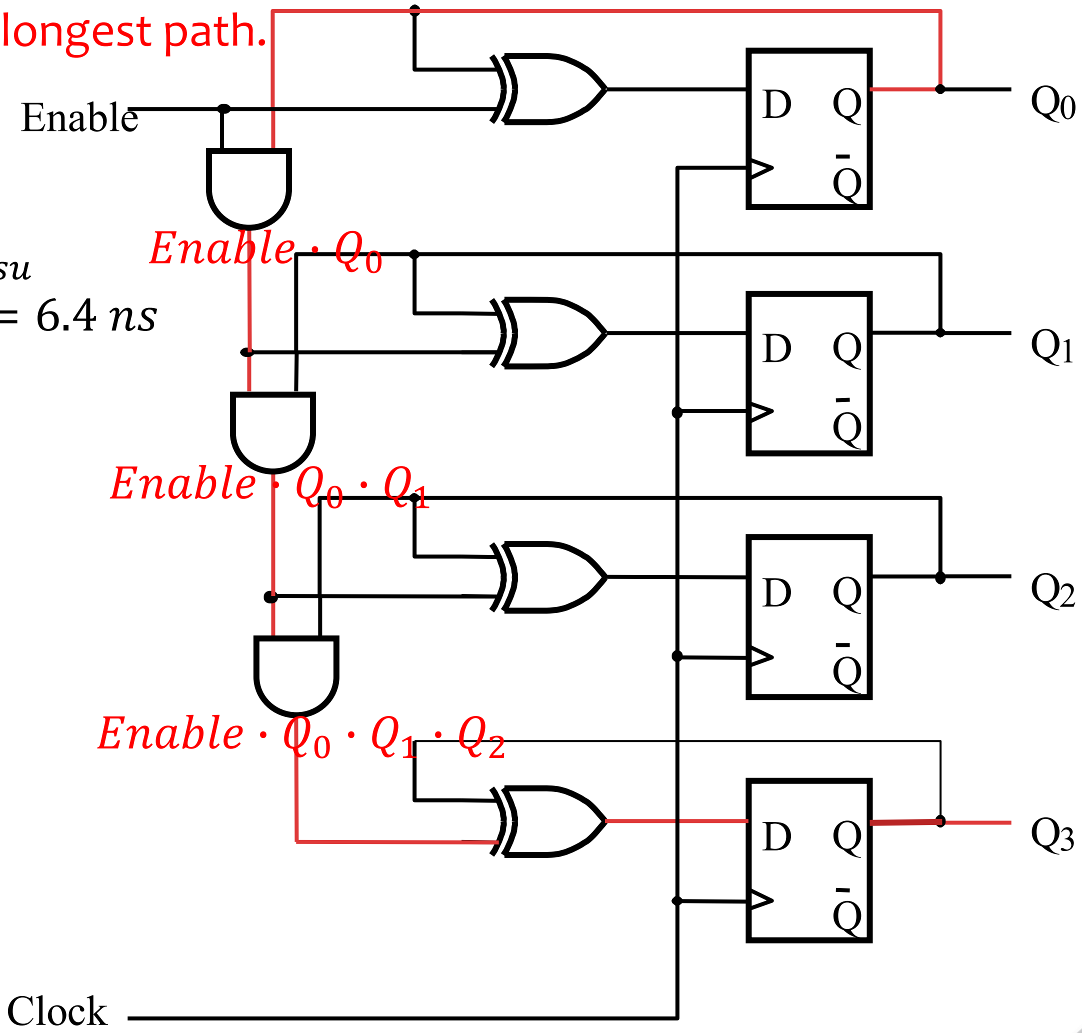

Enable
D
Q
Q̄
Q0
$Enable \cdot Q_0$
Q1
$Enable \cdot Q_0 \cdot Q_1$
Q2
$Enable \cdot Q_0 \cdot Q_1 \cdot Q_2$
Q3
Clock

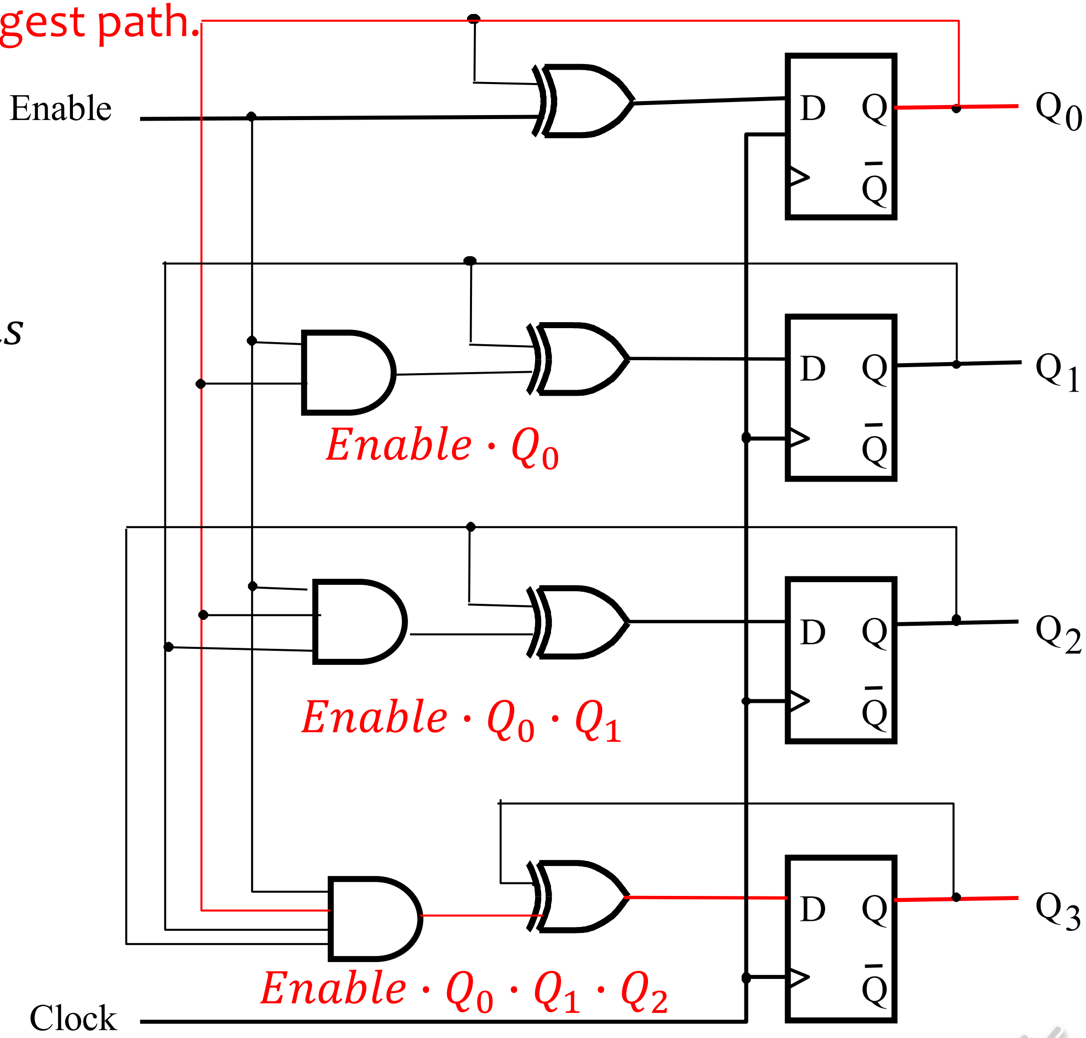
Figure 7.81. A 4-bit counter.

Example 7.18

Red line from Q_0 to Q_3 shows the longest path.

$$T_{min} = t_{cQ} + t_{AND} + t_{XOR} + t_{su}$$
$$= 1.0 + 1.4 + 1.2 + 0.6 = 4.2\ ns$$

$$F_{max} = \frac{1}{4.2ns} = 238.1\ MHz$$


Enable
D
Q
Q
Q_0
D
Q
Q
Q_1
$Enable \cdot Q_0$
D
Q
Q
Q_2
$Enable \cdot Q_0 \cdot Q_1$
D
Q
Q
Q_3
$Enable \cdot Q_0 \cdot Q_1 \cdot Q_2$
Clock

Summary

- Basic latch is a feedback connection of two NOR gates or two NAND gates.
- Gated latch a basic latch that includes input gating and a control signal.
  - Gated SR latch uses the S and R inputs to set the latch to 1 or reset it to 0.
  - Gated D latch uses the D input to force the latch into a state that has the same logic value as the D input
- A flip-flop: Output state can be changed only on the edge of the controlling clock signal
  - Edge-triggered flip-flop is affected only by the input values present when the active edge of the clock occurs.
  - Master-slave flip-flop is built with two gated latches.

Summary

- When a set of n flip-flops is used to store n bots of information, such as n-bit number is referred as a register.
- A register that provides the ability to shift its contents is called a shift register.
- Counters are circuits that can increment or decrement by 1, and are used in digital systems for many purposes.
- Asynchronous counters are simple, but not very fast.
- Counters have the capability with Enable, Clear, and Load.

Summary

- When the counter is reset on the active edge of the clock, this type of counter has a reset synchronous reset.
- There are other types of counters including BCD counter (Modulo 10 counter), ring counter, and Johnson counter.
- In the bus structure, each register is connected to the bus through an n-bit tri-state buffer.
- As timing analysis of flip-flop circuits, the maximum clock frequency is investigated based on the timing parameters such as set-up time, hold time, clock edge to the Q output time, and logic gate delay.