Sequential Circuits

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Circuits

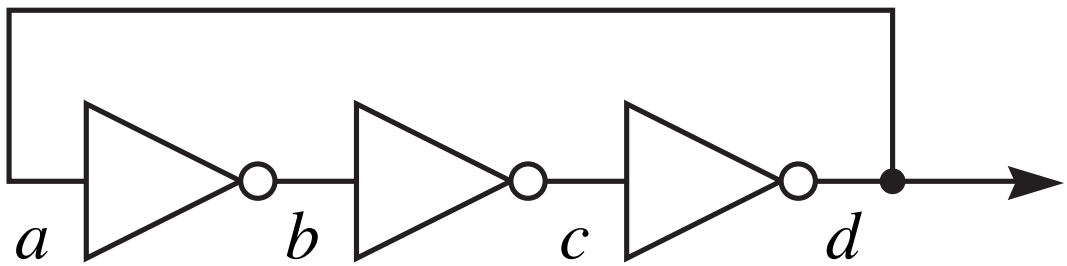
- Combinational circuit
  - The output depends only on the input
- Sequential circuit
  - Has a state
  - The output depends not only on the input but also on the state the circuit is in

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

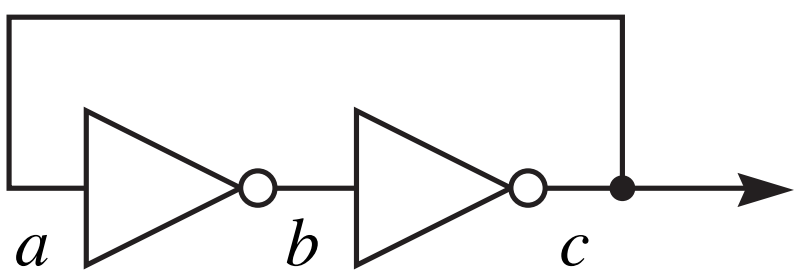
Sequential circuit

- Constructed from standard gates, but with one or more *feedback* connections
- An *unstable state* is one that will change a few gate delays later because of the feedback connection
- A *stable state* is one that will persist indefinitely until the input changes

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) An unstable circuit.



(b) A stable circuit.

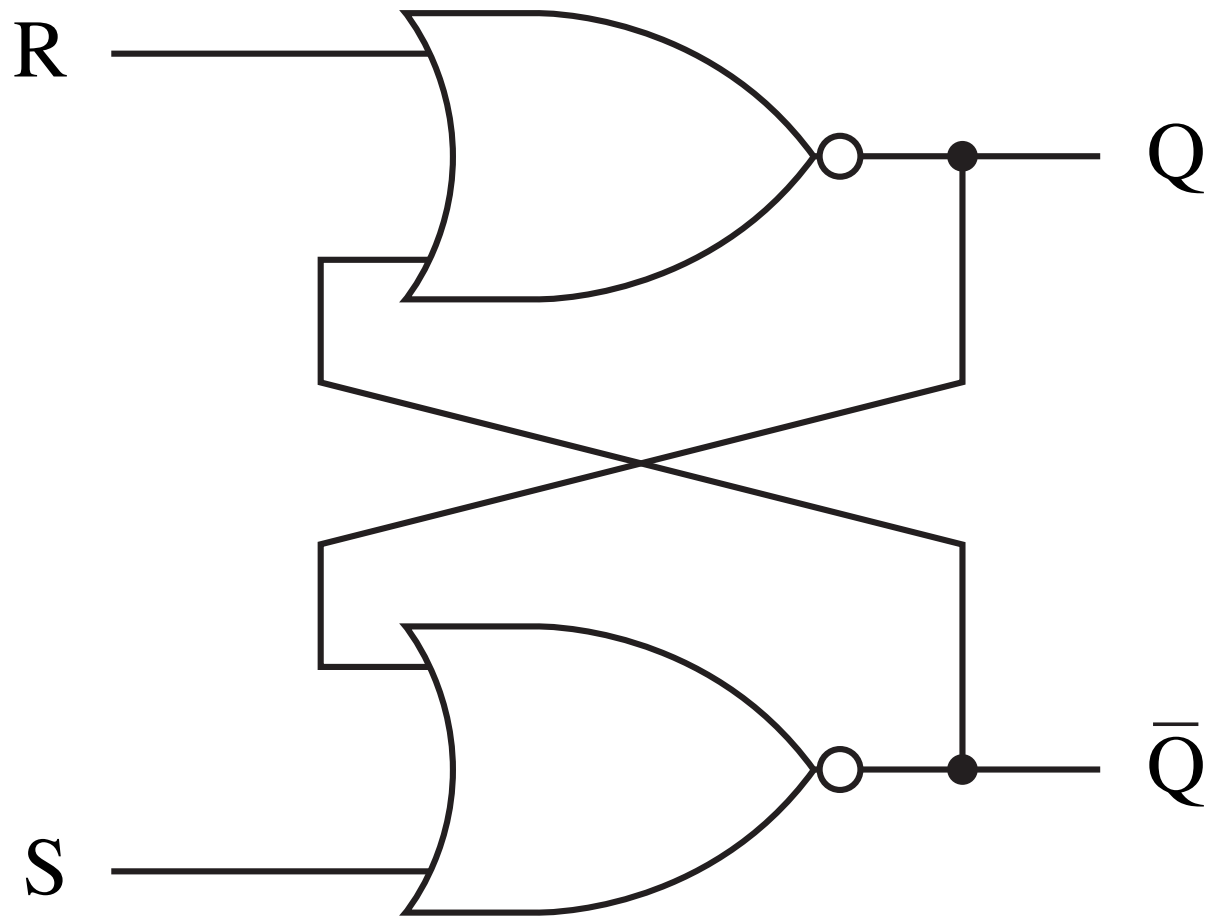
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

SR Latch

- The SR latch has two stable states
- When SR = 00, output Q can be 0 or 1 depending on the state of the latch
- S = 1 sets ouput Q to 1
- R = 1 resets output Q to 0

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.2



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.3

Time	S	R	Q	$\bar{Q}$	Stability
Initial	0	0	0	1	Stable
0	1	0	0	1	Unstable
T_g	1	0	0	0	Unstable
$2T_g$	1	0	1	0	Stable

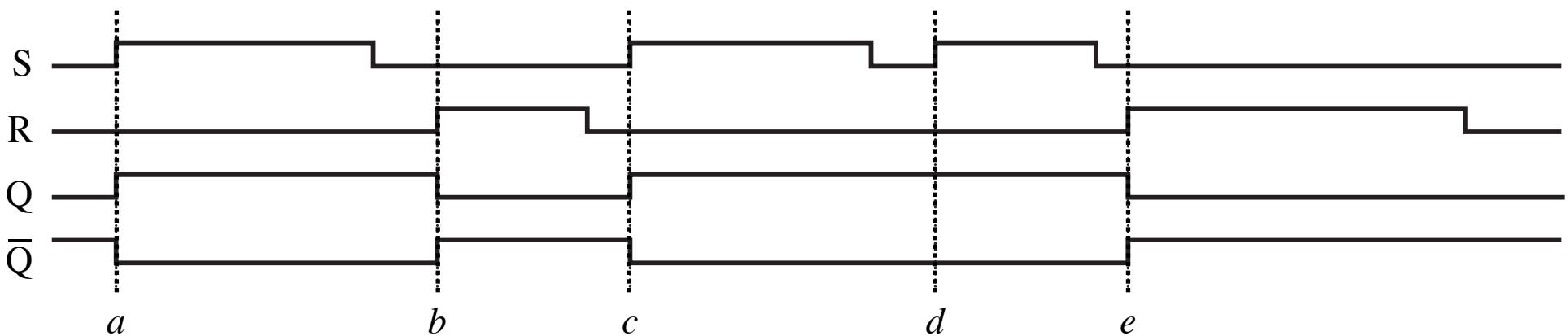
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.4

Time	S	R	Q	$\overline{Q}$	Stability
Initial	1	0	1	0	Stable
0	0	0	1	0	Stable

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.5



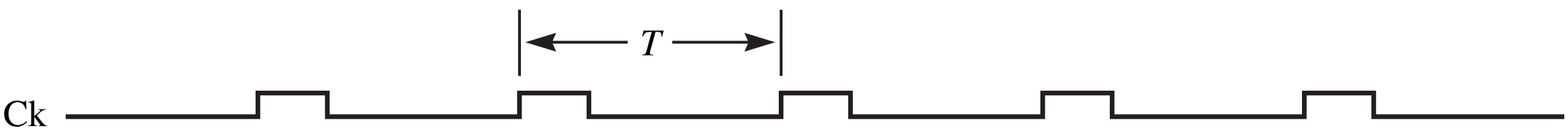
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

System clock

- Controls the state transitions of all the sequential circuits to happen at the same time
- Sequence of regularly spaced pulses with period *T*

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.6



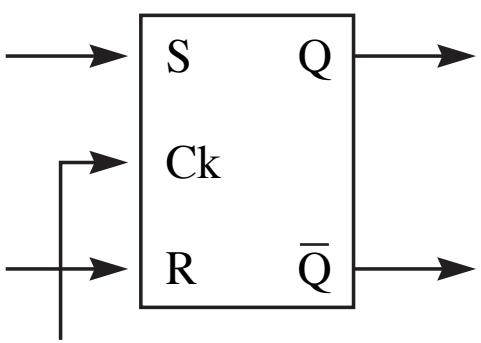
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Clocked SR flip-flop

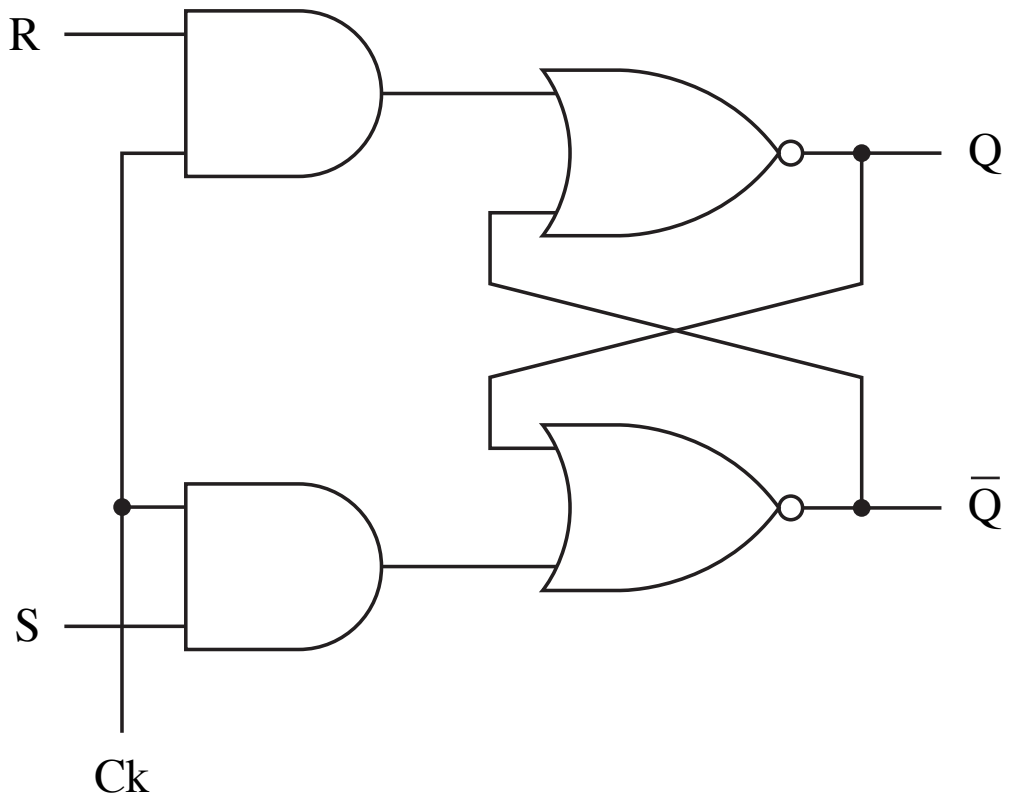
- Two AND gates that act as an enable
- Only when Ck is high can the S and R inputs affect the state of the flip-flop
- The effect is to digitize the time axis

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.7



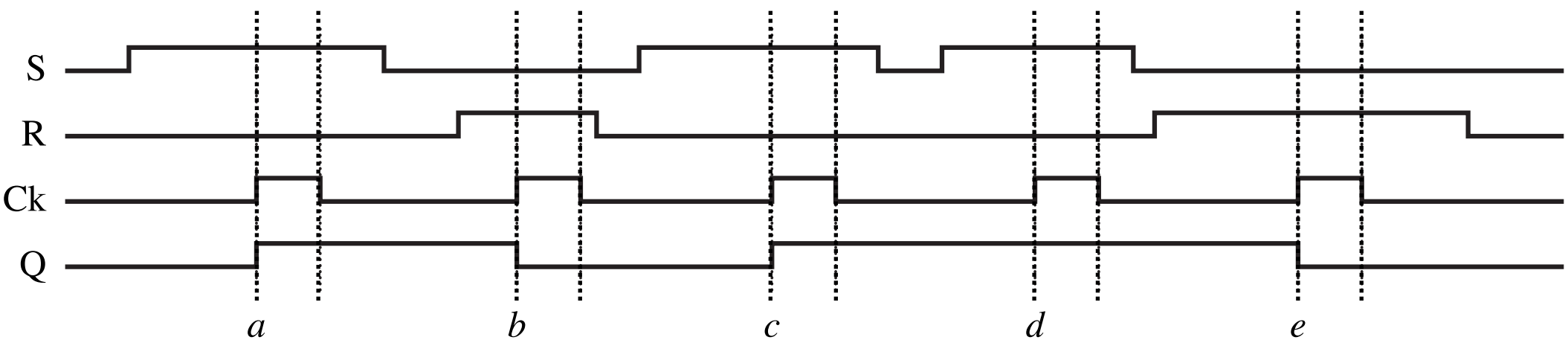
(a) Block diagram.



(b) Implementation.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.8



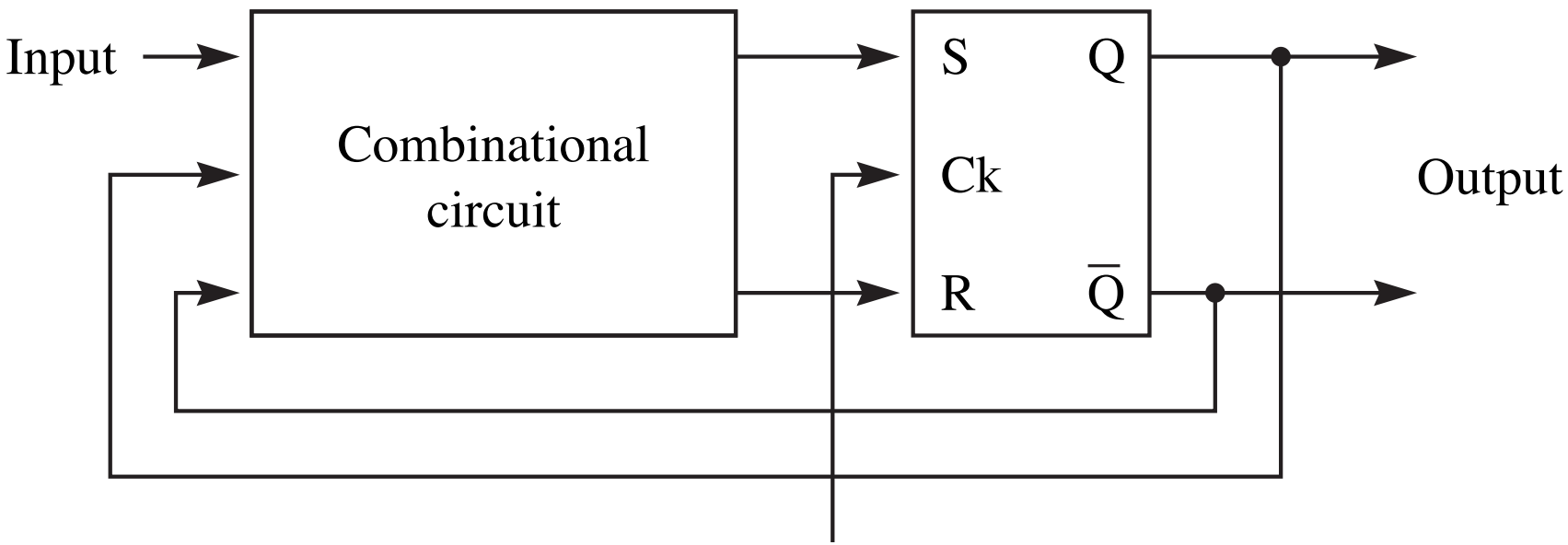
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

The feedback problem

- Flip-flops are often used in circuits with feedback connections (in addition to the internal feedback in the latch)
- Therefore, unstable states are possible
- There are two design solutions to the feedback problem
  - Edge-triggered flip-flops
  - Master-slave flip-flops

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.9



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Master-slave SR flip-flop

- Solves the instability problem caused by possible external feedback
- Input goes to the master latch first, and then from the master to the slave, in four steps
- The *threshold* of a gate is the value of the input signal that causes the output to change
- Engineers can make gates with thresholds a little above or a little below the average value between 0 and 1

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

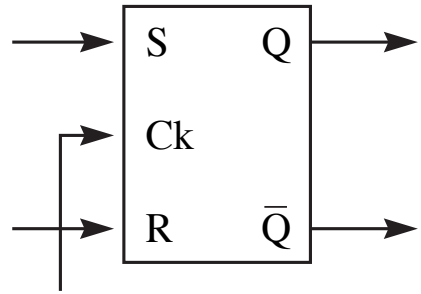
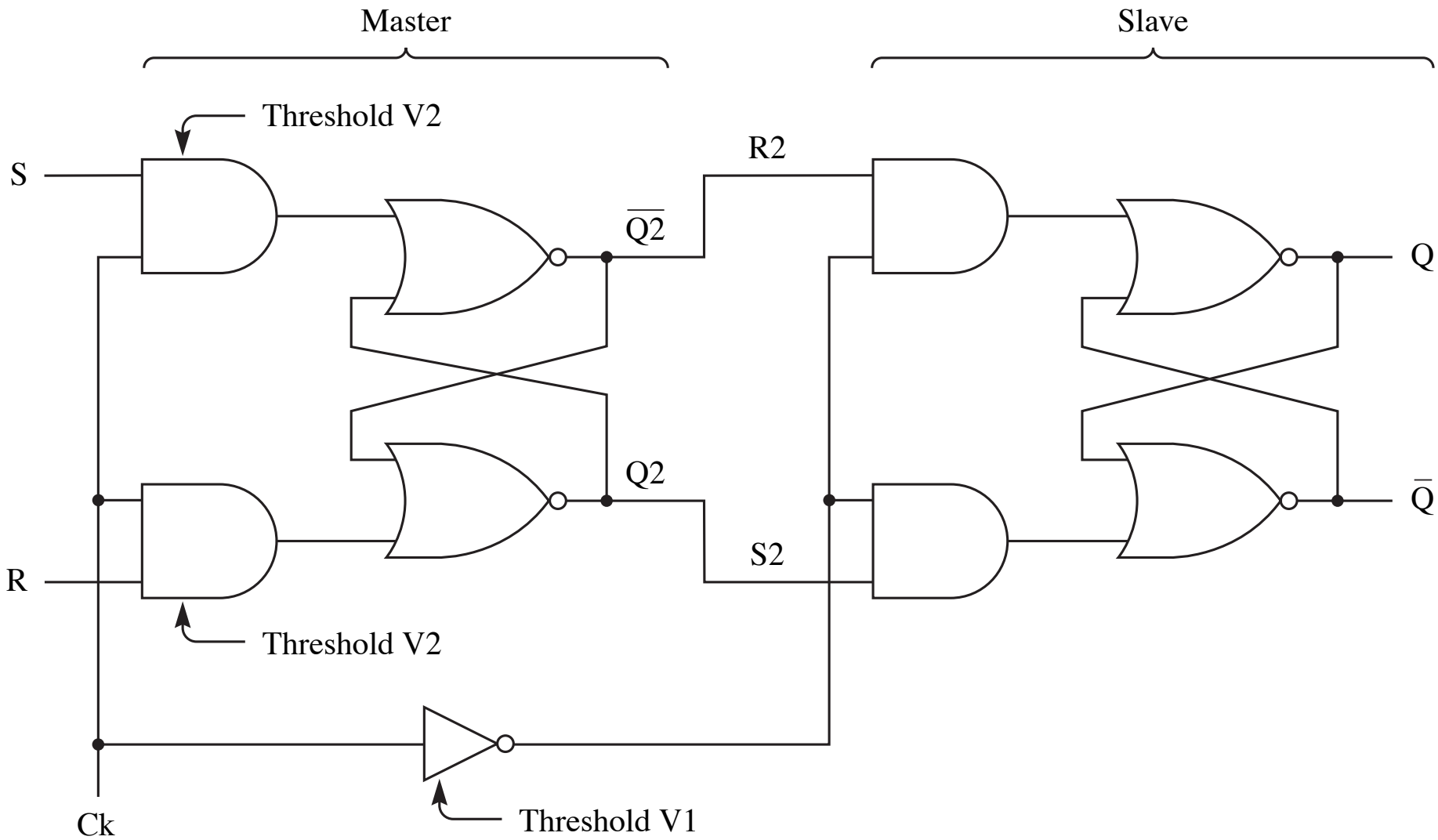


Figure 11.10

(a) Block diagram.



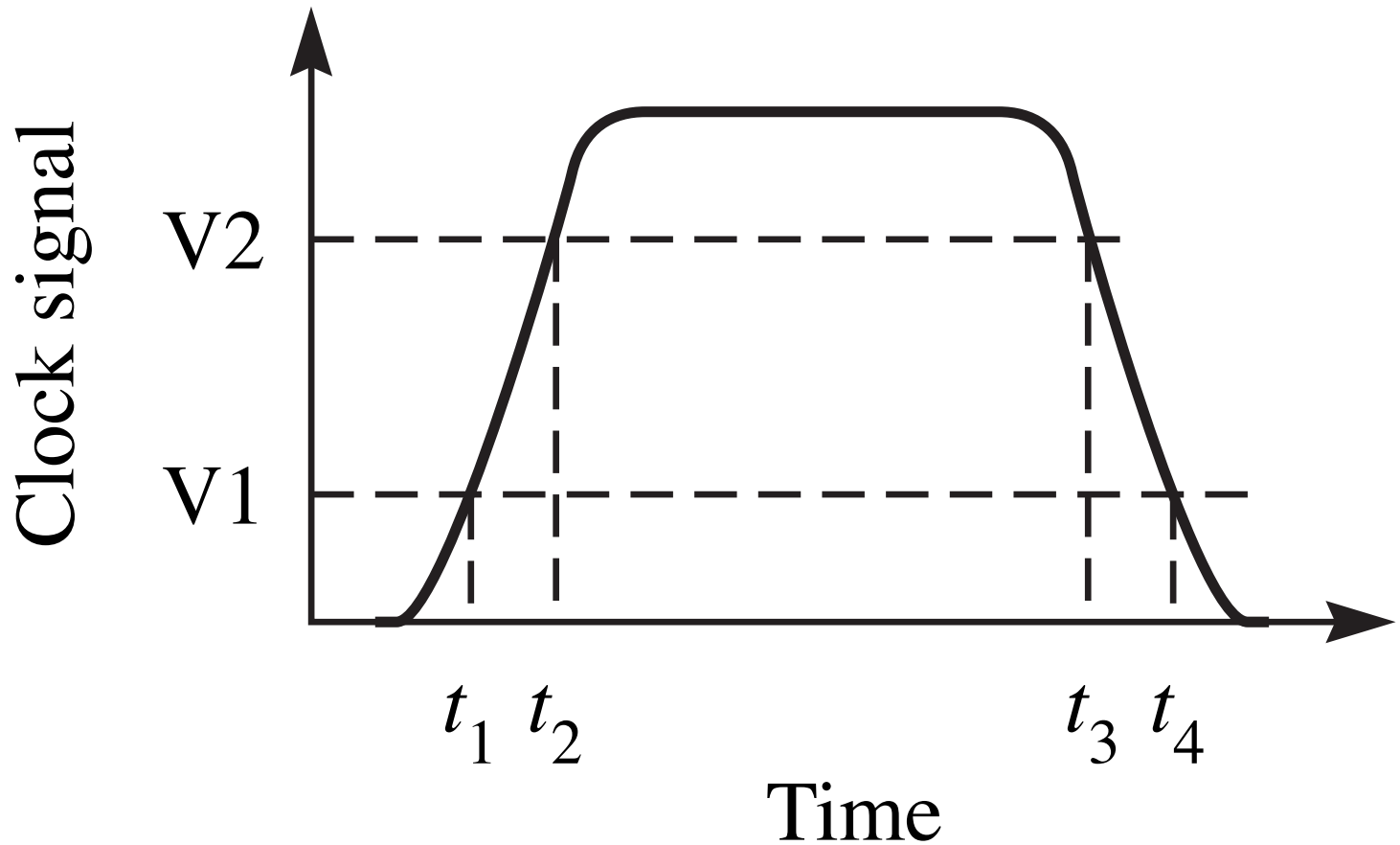
(b) Implementation.

Timing detail of a single Ck pulse

- t_1: Isolate slave from master
- t_2: Connect master to input
- t_3: Isolate master from input
- t_4: Connect slave to master

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.11



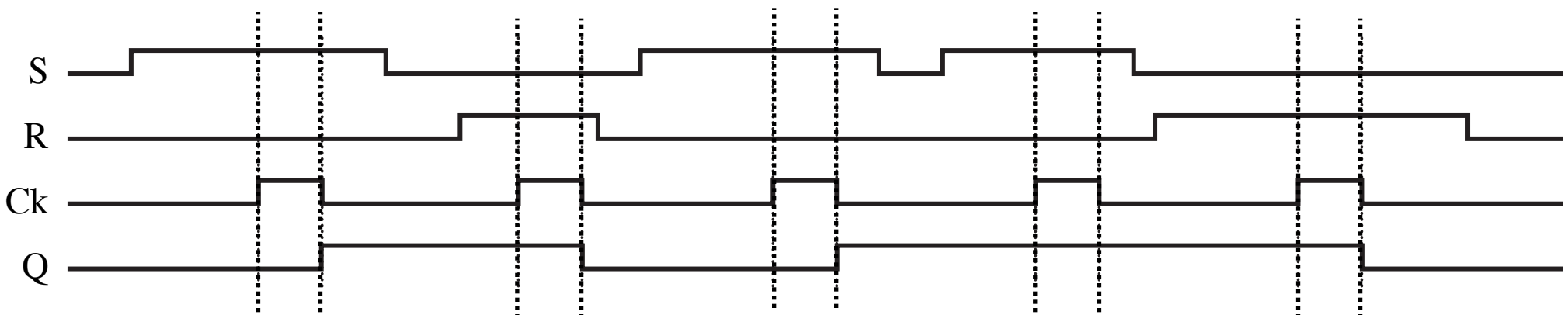
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Effect on timing

- The output changes on the *falling* edge of the Ck pulse and depends on the external SR input at that time

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.12



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Characteristic table

- A truth table is not adequate to describe a flip-flop, because its output depends on more than its input
- Given the inputs at time *t* and the state at time *t*, the characteristic table shows the state at time *t* + 1, that is, after one clock pulse

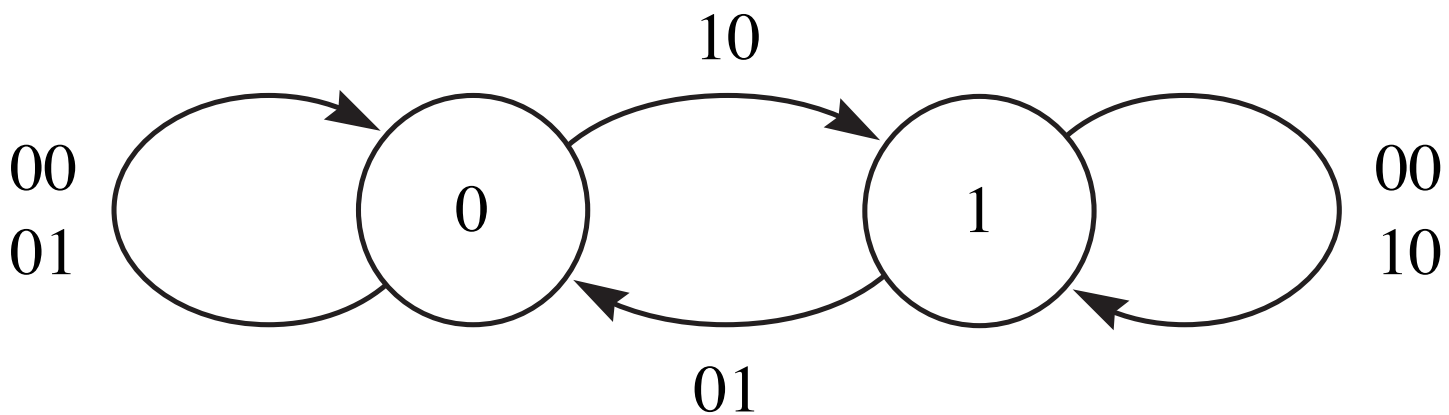
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.13

S(*t*)	R(*t*)	Q(*t*)	Q(*t* + 1)	Condition
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	–	Not defined
1	1	1	–	

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.14



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Four common flip-flops

- SR Set/reset
- JK Set/reset/toggle
- D Data or delay
- T Toggle

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Excitation table

- The excitation table is a design tool for constructing circuits from a given type of flip-flop
- Given the desired transition from Q(*t*) to Q(*t* + 1), what inputs are necessary to make the transition happen?

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.15

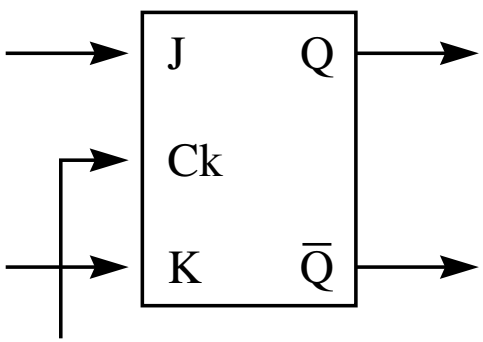
Q(t)	Q($t+1$)	S(t)	R(t)
0	0	0	×
0	1	1	0
1	0	0	1
1	1	×	0

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

JK flip-flop

- Resolves the undefined transition in the SR flip-flop
- When JK = 00, output Q can be 0 or 1 depending on the state of the latch
- J = 1 sets ouput Q to 1 (like S)
- K = 1 resets output Q to 0 (like R)
- JK = 11 toggles from one state to the other

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) Block diagram.

J(t)	K(t)	Q(t)	Q($t+1$)	Condition
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

(b) Characteristic table.

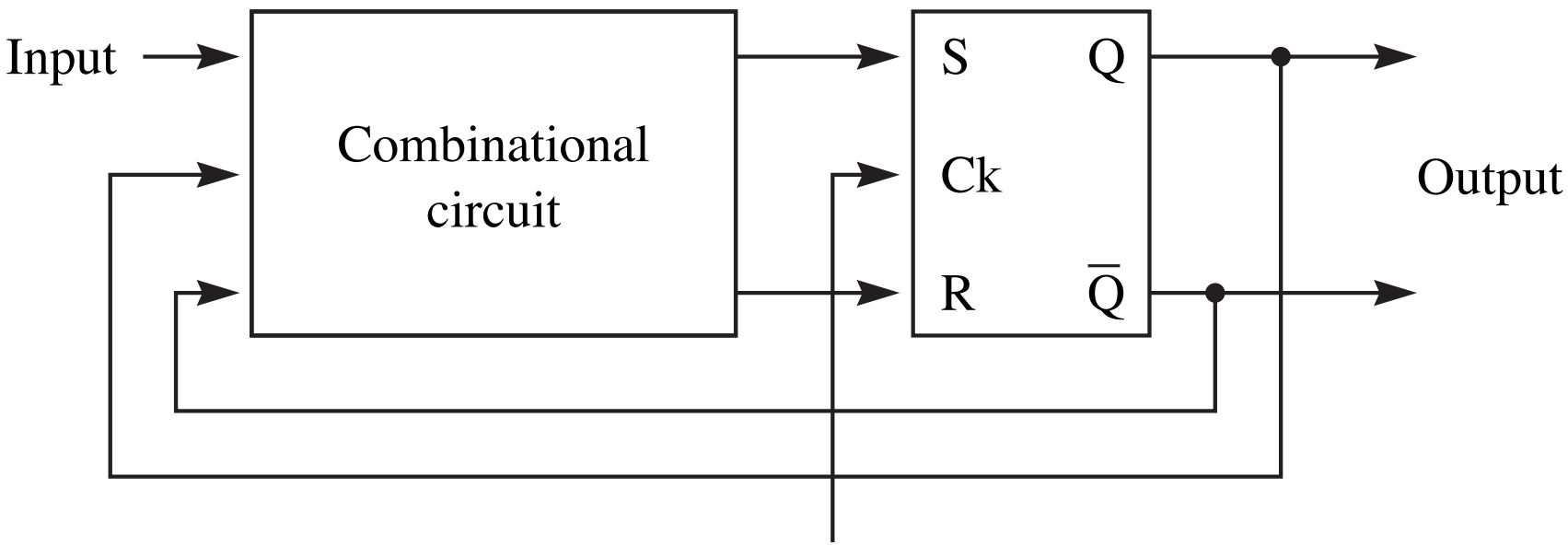
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

JK flip-flop design

- Must design a three-input two-output combinational circuit
- Inputs
  - $J(t)$, $K(t)$, $Q(t)$
- Outputs
  - $S(t)$, $R(t)$

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.9



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Design table

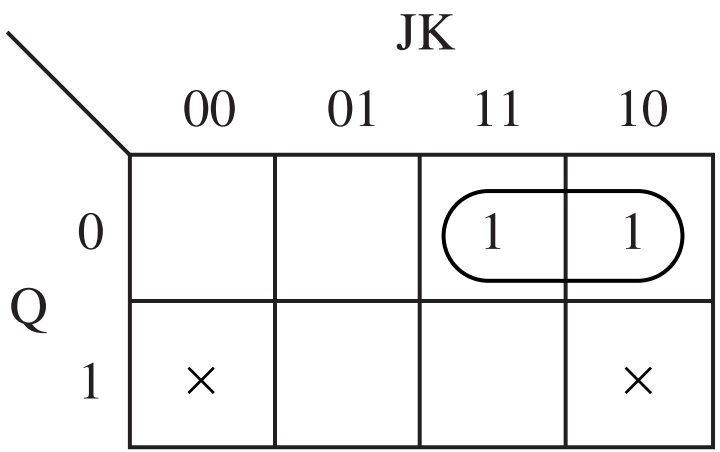
- Step 1: Given Q(t), J(t), and K(t), list the desired state after the transition Q(t + 1)
- Step 2: Given Q(t) and Q(t + 1), use the excitation table to list the required input for S(t) and R(t)
- Step 3: Use Karnaugh maps to design minimized two-level combinational circuits for S(t) and R(t)

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

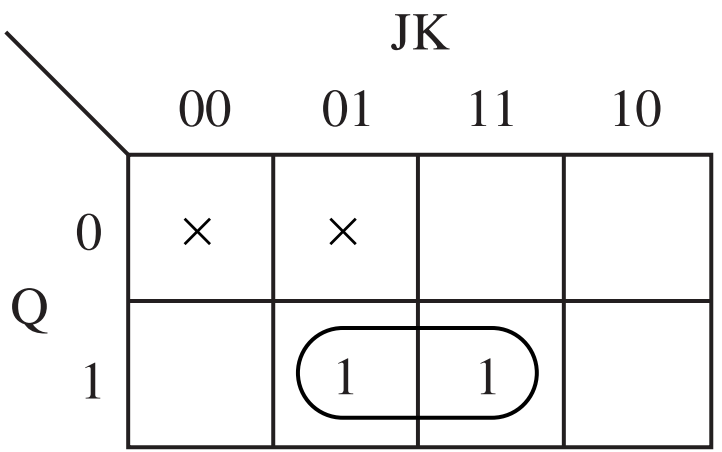
Figure 11.17

Q(*t*)	J(*t*)	K(*t*)	Q(*t* + 1)	S(*t*)	R(*t*)
0	0	0	0	0	×
0	0	1	0	0	×
0	1	1	1	1	0
0	1	0	1	1	0
1	0	0	1	×	0
1	0	1	0	0	1
1	1	1	0	0	1
1	1	0	1	×	0

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



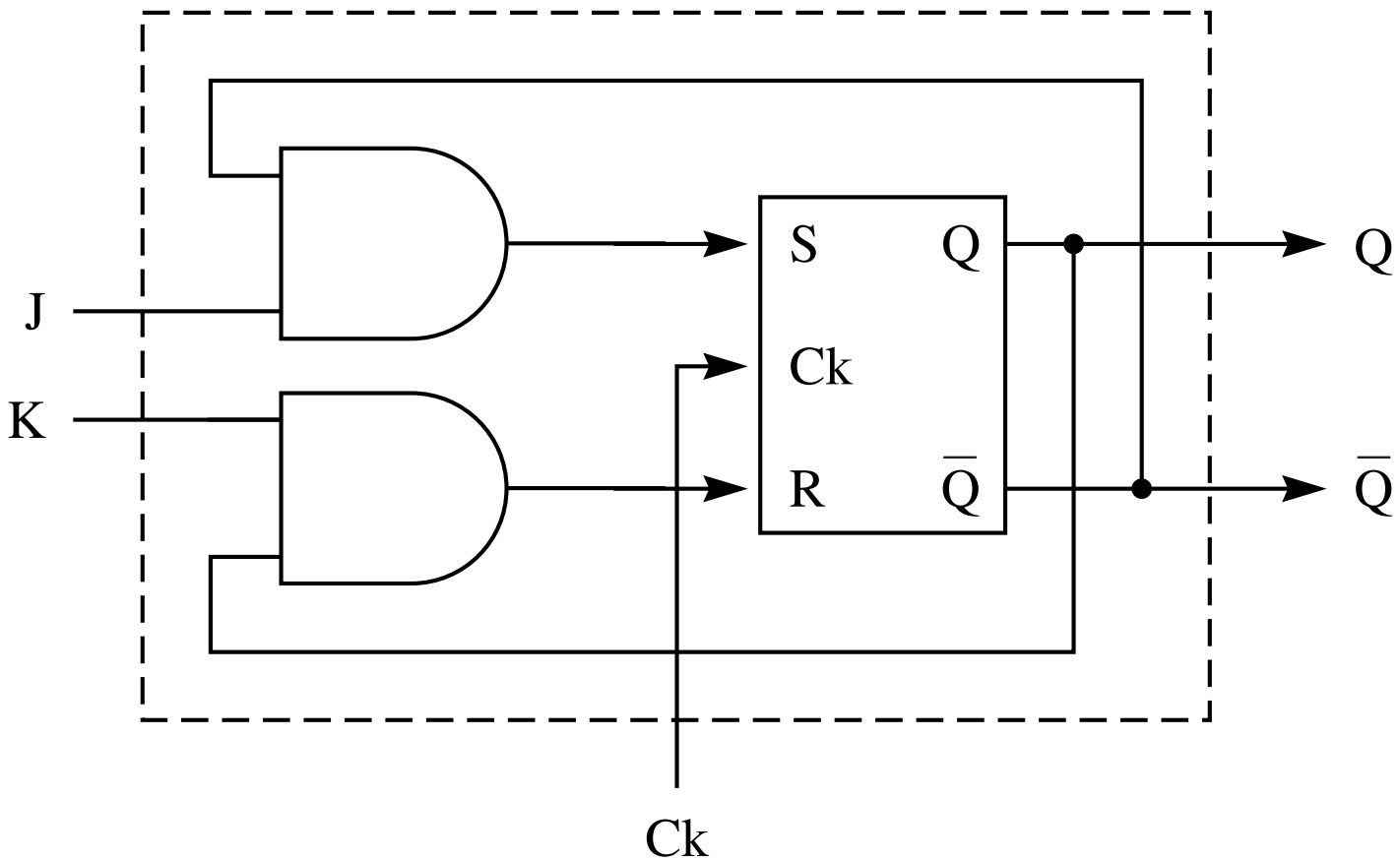
(a) Karnaugh map for S.



(b) Karnaugh map for R.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.19

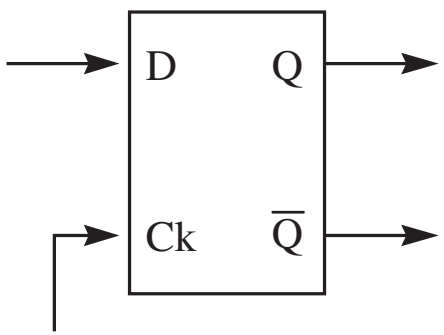


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

D flip-flop

- The "delay" or "data" flip-flop
- Only one input, D
- Regardless of the current state Q(*t*), the state after the clock pulse Q(*t* + 1) will be the same as D(*t*)

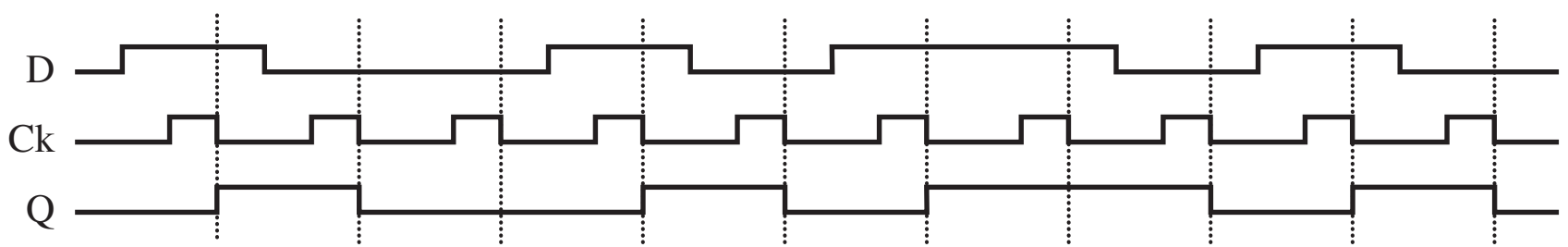
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) Block diagram.

D(*t*)	Q(*t*)	Q(*t* + 1)	Condition
0	0	0	Delay
0	1	0	
1	0	1	Delay
1	1	1	

(b) Characteristic table.

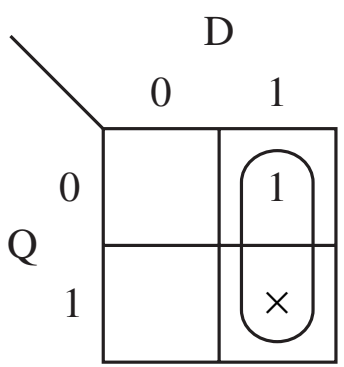


(c) A timing diagram.

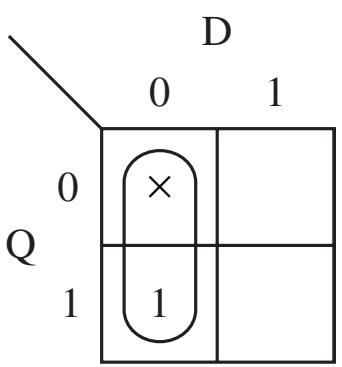
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Q(*t*)	D(*t*)	Q(*t* + 1)	S(*t*)	R(*t*)
0	0	0	0	×
0	1	1	1	0
1	1	1	×	0
1	0	0	0	1

(a) Design table.



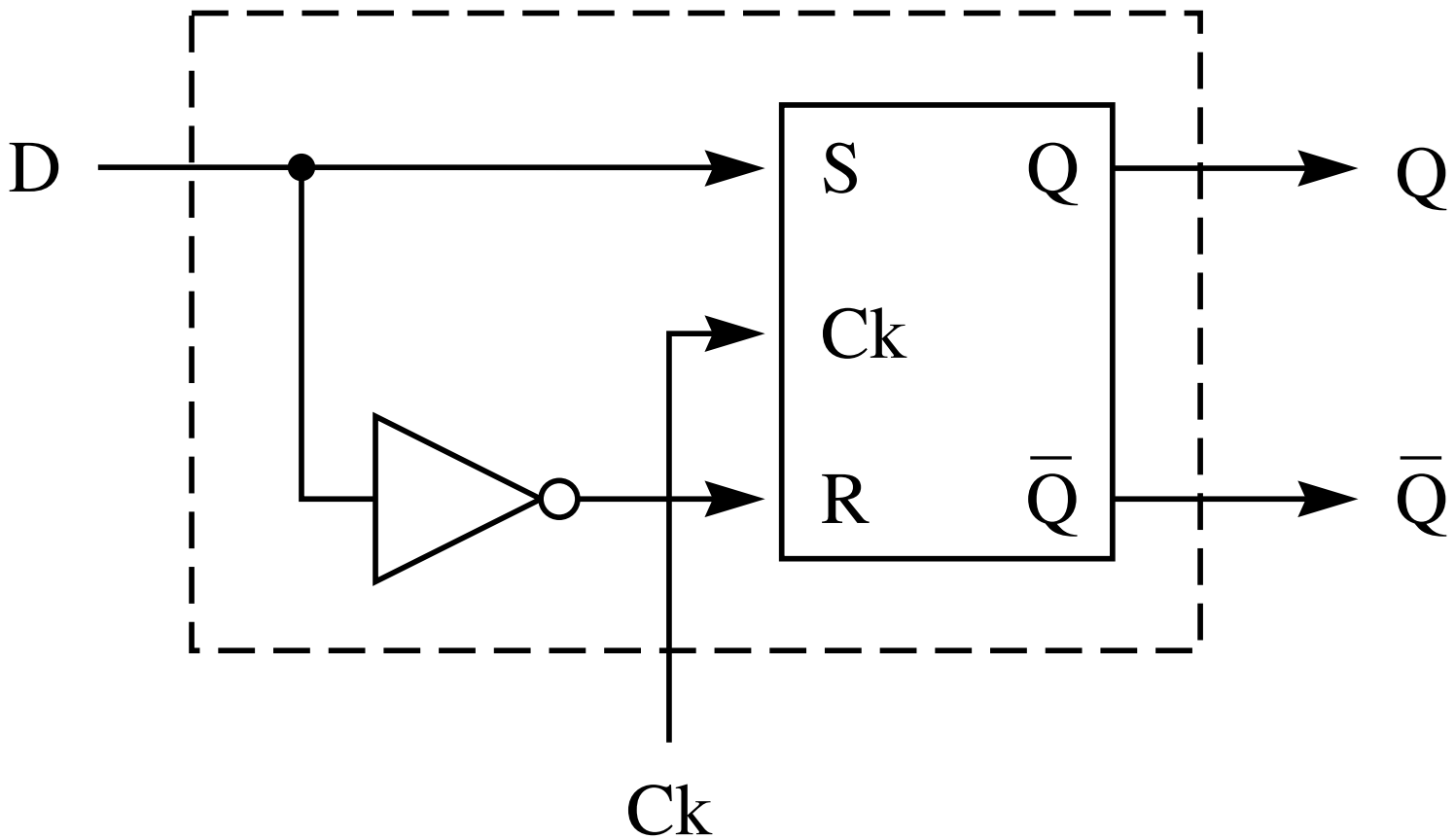
(b) Karnaugh map for S.



(c) Karnaugh map for R.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.22

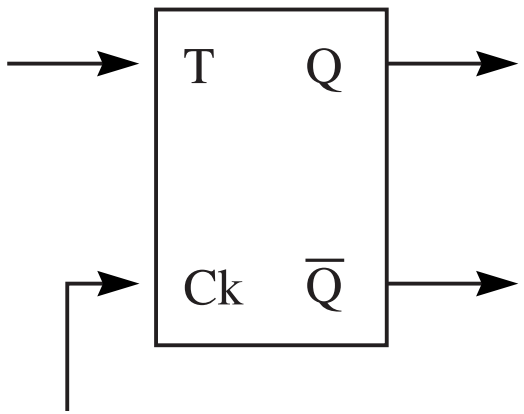


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

T flip-flop

- The "toggle" flip-flop
- Only one input, T
- If T = 0, the state remains unchanged
- If T = 1, the state toggles from 0 to 1 or from 1 to 0

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) Block diagram.

T(*t*)	Q(*t*)	Q(*t* + 1)	Condition
0	0	0	No change
0	1	1	
1	0	1	Toggle
1	1	0	

(b) Characteristic table.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Flip-flop design

- Any given flip-flop can be constructed from any other flip-flop with the right combinational circuit
- Use the excitation table for the flip-flop from which you are constructing the given flip-flop

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.24

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

(a) The JK flip-flop.

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

(b) The D flip-flop.

Q(*t*)	Q(*t* + 1)	T(*t*)
0	0	0
0	1	1
1	0	1
1	1	0

(c) The T flip-flop.

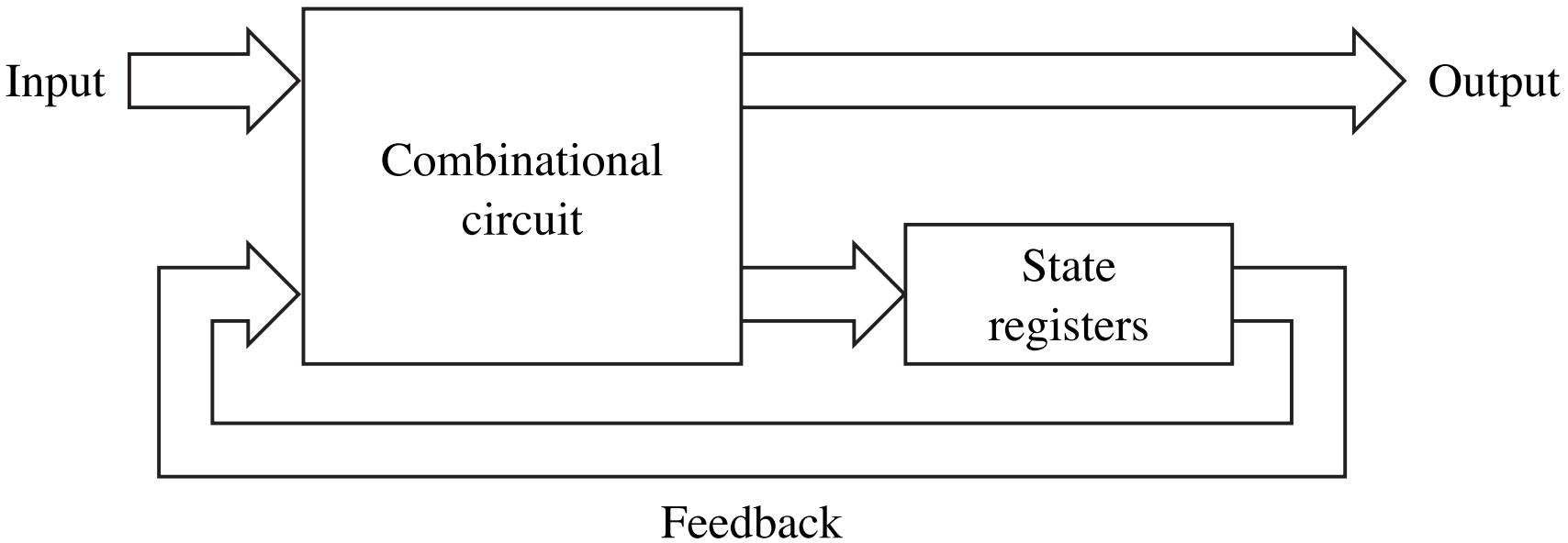
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

General sequential circuits

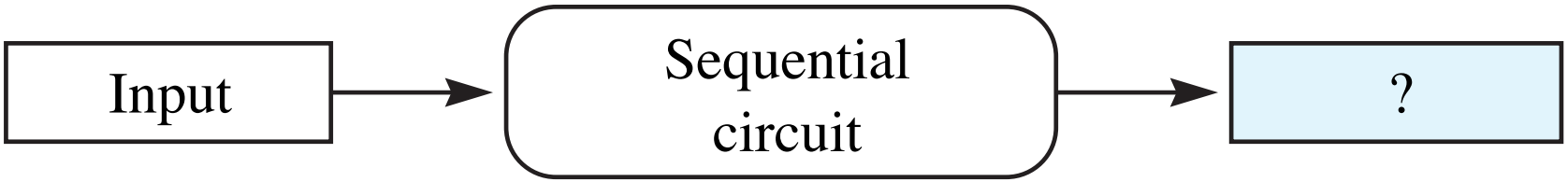
- A general sequential circuit is an interconnection of gates and flip-flops
- The flip-flops are called state registers
- The current state and current input determine the current output
- The current state and current input determine the next state, that is, the state after one Ck clock pulse

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

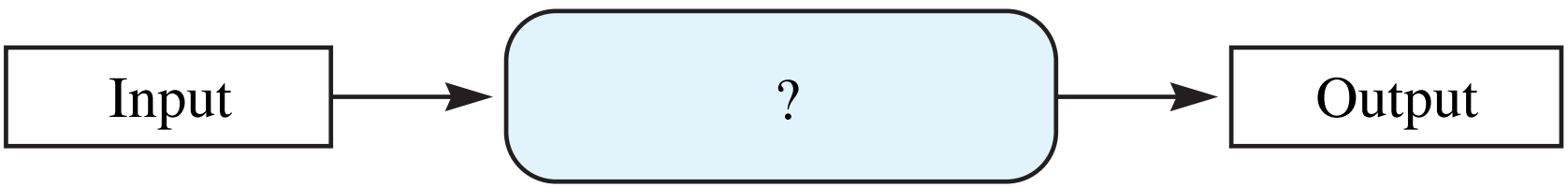
Figure 11.25



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) Analysis—The input and sequential circuit are given. The output is to be determined.



(b) Design—The input and desired output are given. The sequential circuit is to be determined.

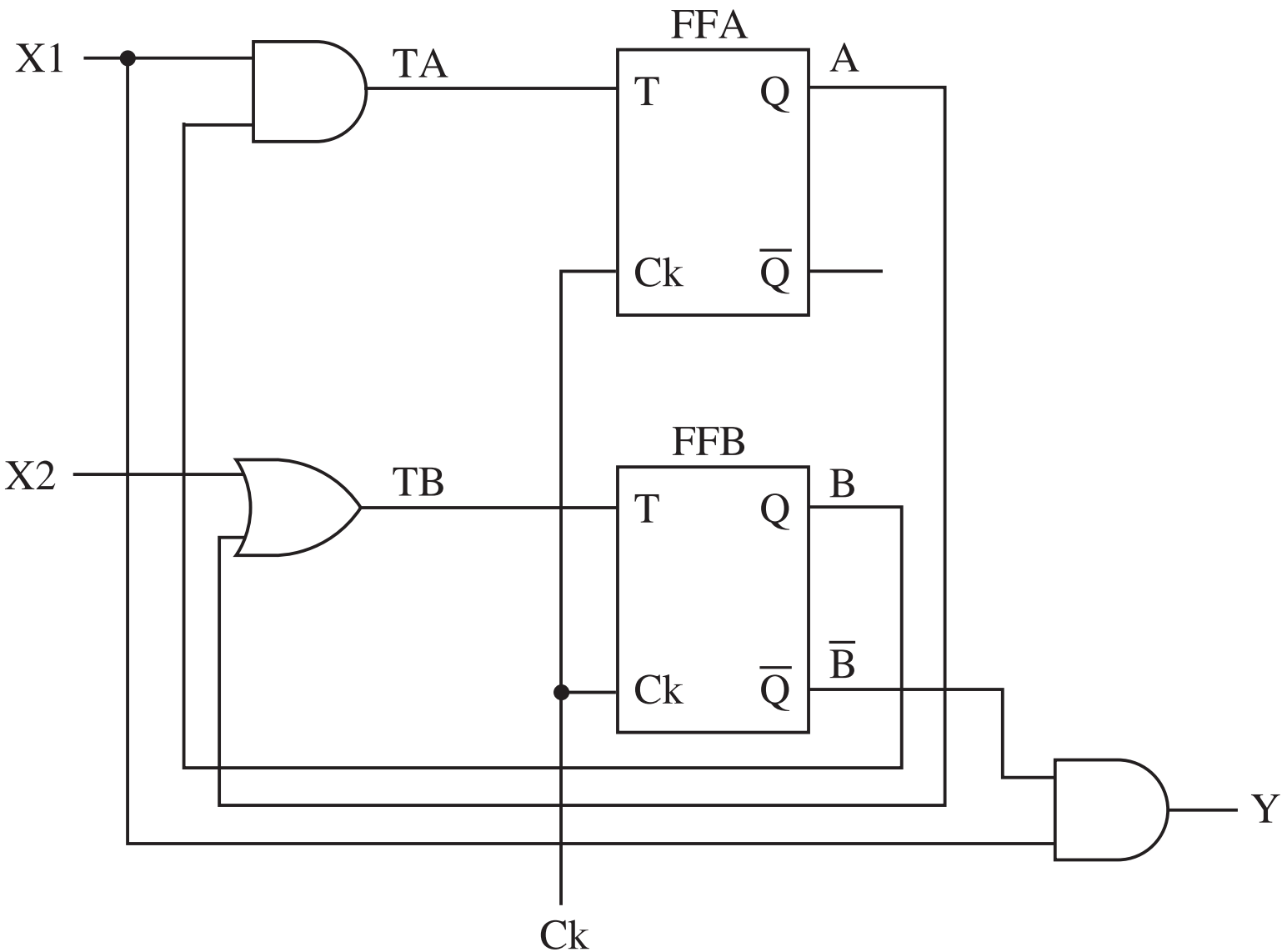
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Sequential analysis

- Step 1: List all possible combinations of current state and current input in an analysis table
- Step 2: For each combination, compute the output and the current inputs to the state registers
- Step 3: From the characteristic table, determine the next state and construct the state transition table and diagram

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.27



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.28

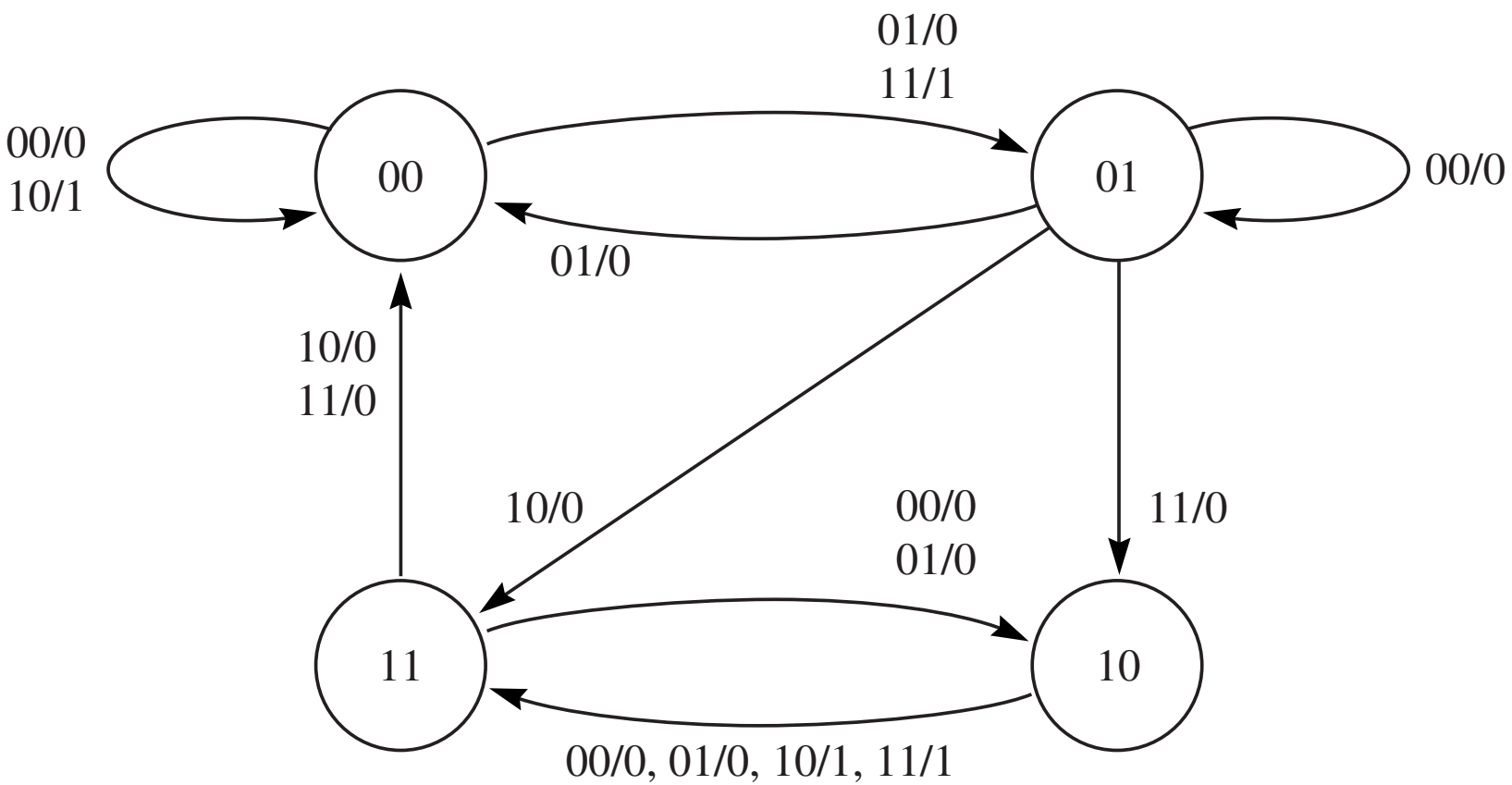
A(t)	B(t)	X1(t)	X2(t)	Y(t)	TA(t)	TB(t)	A(t + 1)	B(t + 1)
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	1
0	0	1	0	1	0	0	0	0
0	0	1	1	1	0	1	0	1
0	1	0	0	0	0	0	0	1
0	1	0	1	0	0	1	0	0
0	1	1	0	0	1	0	1	1
0	1	1	1	0	1	1	1	0
1	0	0	0	0	0	1	1	1
1	0	0	1	0	0	1	1	1
1	0	1	0	1	0	1	1	1
1	0	1	1	1	0	1	1	1
1	1	0	0	0	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	0	1	1	0	0
1	1	1	1	0	1	1	0	0

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.29

A(t) B(t)	X1(t) X2(t) 00	01	10	11
00	00, 0	01, 0	00, 1	01, 1
01	01, 0	00, 0	11, 0	10, 0
10	11, 0	11, 0	11, 1	11, 1
11	10, 0	10, 0	00, 0	00, 0
	A(t + 1) B(t + 1), Y(t)			

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



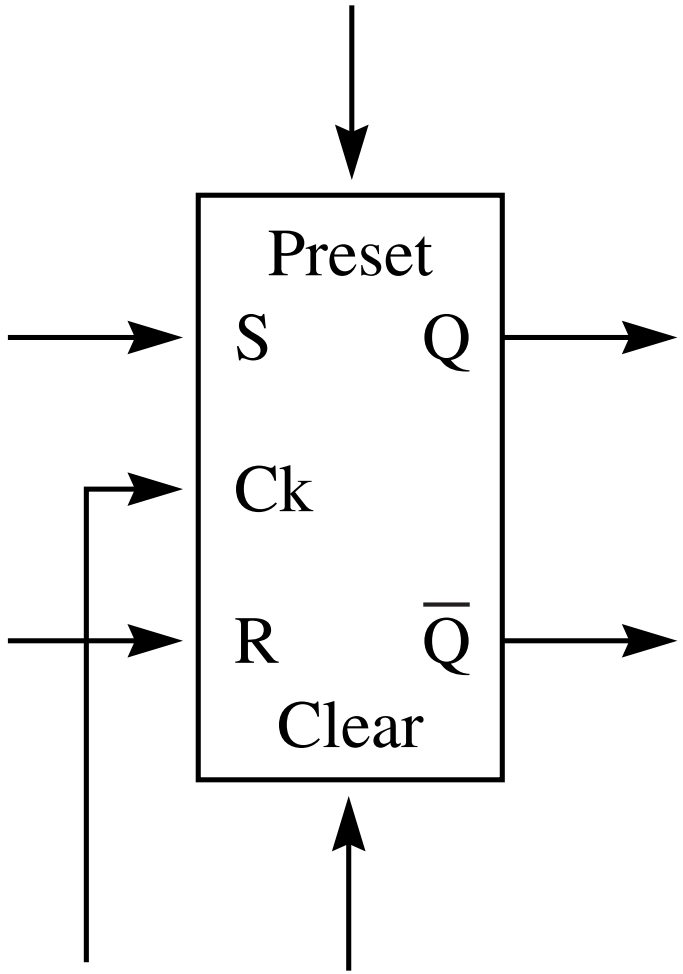
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Asynchronous inputs

- An *asynchronous* input changes the state of a flip-flop immediately without regard to Ck
- Preset sets Q to 1
- Clear clears Q to 0
- Used to initialize the state of a machine

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.31



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Sequential design

- Given the state transition diagram, the output, and the type of flip-flop to be used, design the combinational circuit
- Any unused input combinations or unused states are don't care conditions
- 2^n states are possible with n flip-flops

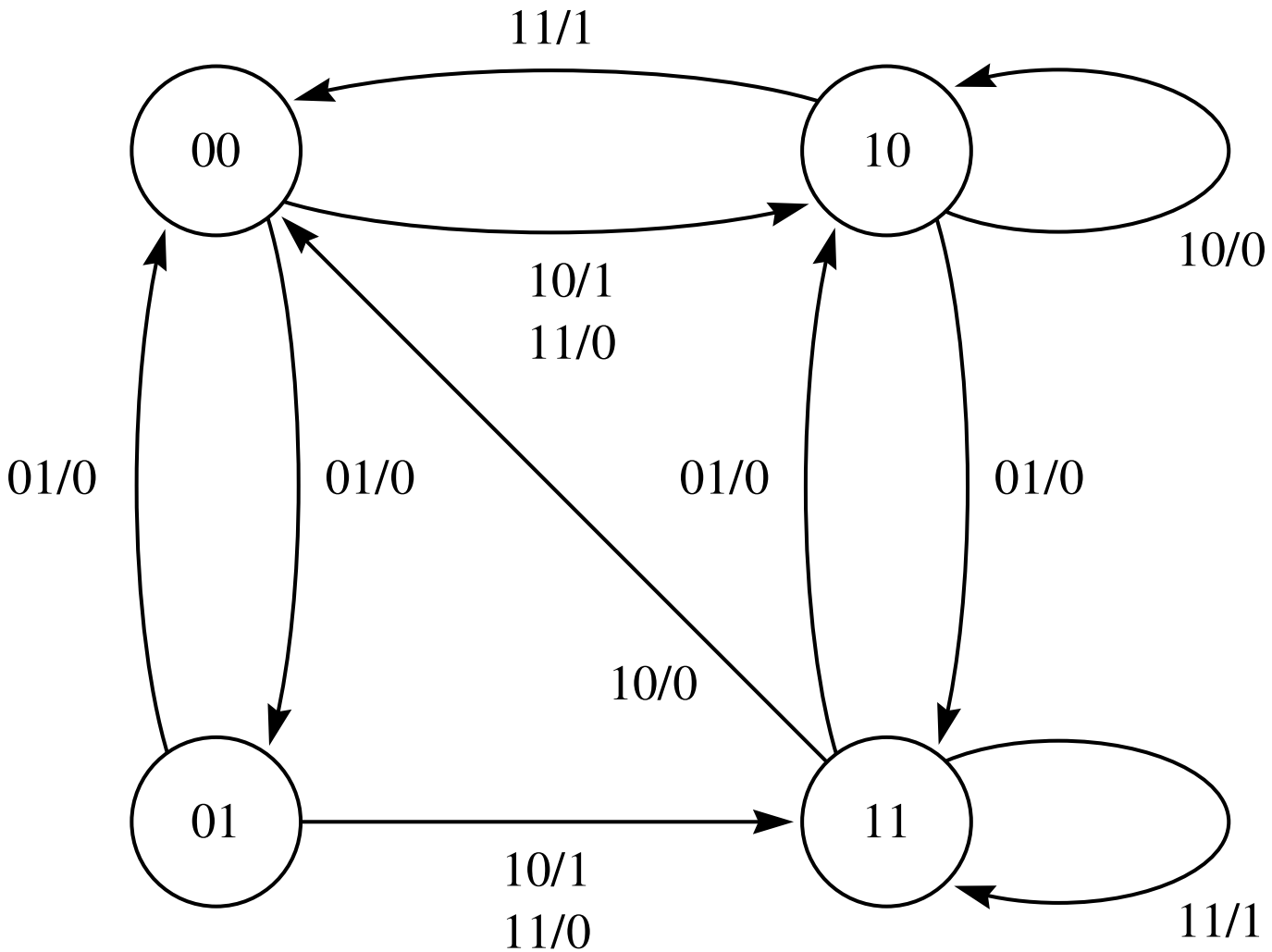
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Design steps

- Step 1: In a design table, list the initial state, input, and output, and from the transition diagram list the next state
- Step 2: Use the excitation table for the given type of flip-flop to determine the input required for the state registers
- Step 3: Use Karnaugh maps to design a minimized two-level circuit for each flip-flop input

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.32

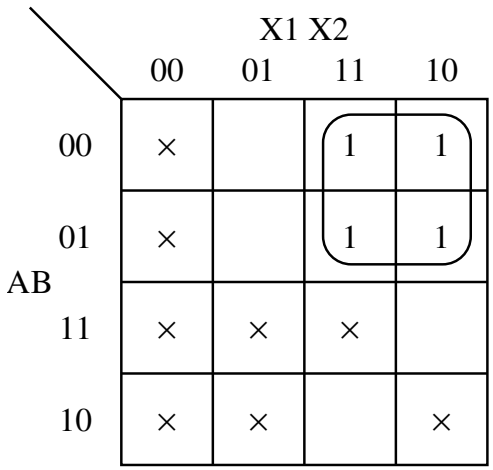


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

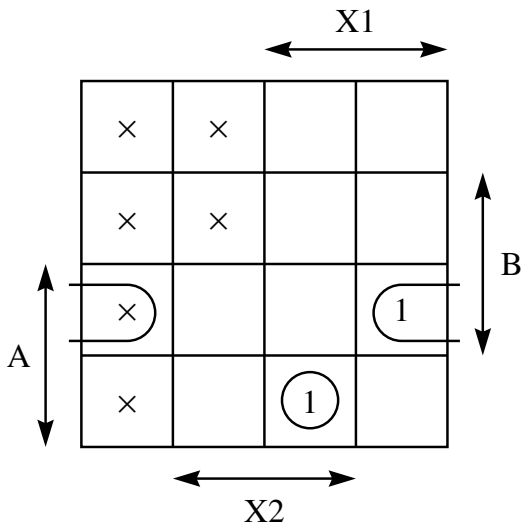
Figure 11.33

Initial state		Initial input		Initial output	Next state		Flip-flop input conditions			
							FFA		FFB	
A(t)	B(t)	X1(t)	X2(t)	Y(t)	A($t+1$)	B($t+1$)	SA(t)	RA(t)	SB(t)	RB(t)
0	0	0	1	0	0	1	0	×	1	0
0	1	0	1	0	0	0	0	×	0	1
1	1	0	1	0	1	0	×	0	0	1
1	0	0	1	0	1	1	×	0	1	0
0	0	1	1	0	1	0	1	0	0	×
0	1	1	1	0	1	1	1	0	×	0
1	1	1	1	1	1	1	×	0	×	0
1	0	1	1	1	0	0	0	1	0	×
0	0	1	0	1	1	0	1	0	0	×
0	1	1	0	1	1	1	1	0	×	0
1	1	1	0	0	0	0	0	1	0	1
1	0	1	0	0	1	0	×	0	0	×

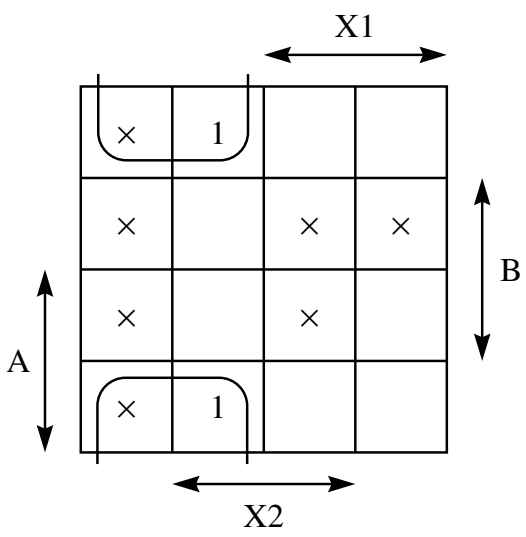
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



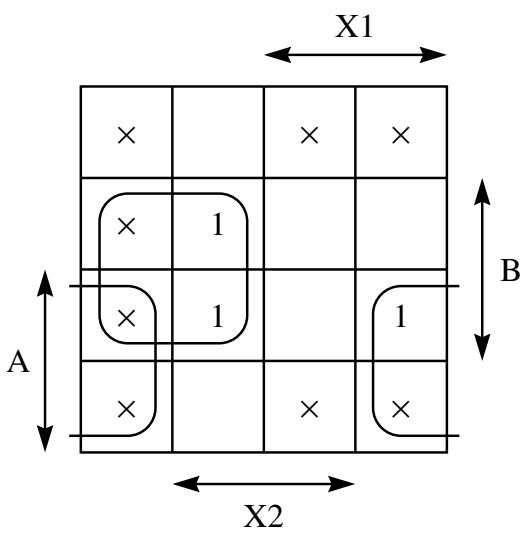
(a) SA = $\overline{A}$ X1



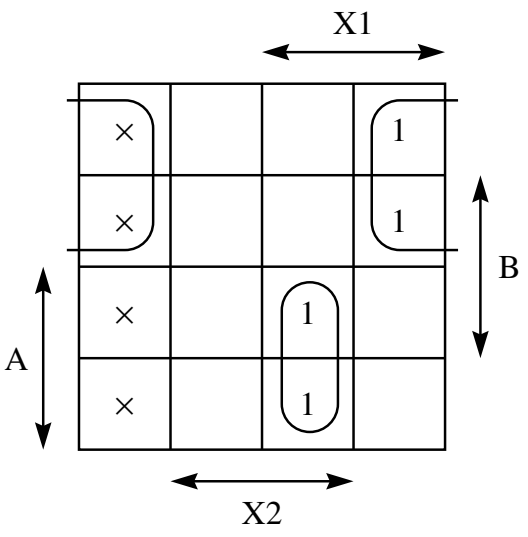
(b) RA = A B $\overline{X2}$ + A $\overline{B}$ X1 X2



(c) SB = $\overline{B}$ $\overline{X1}$



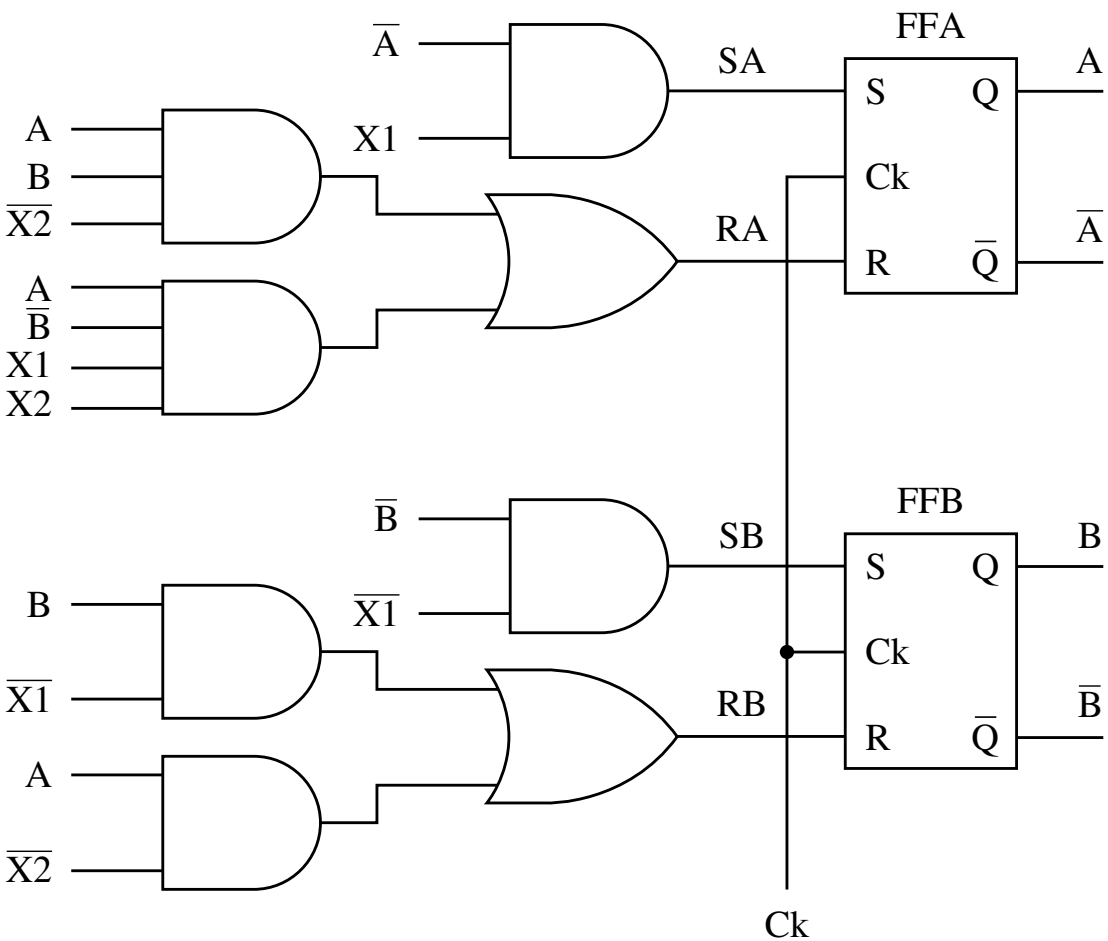
(d) RB = B $\overline{X1}$ + A $\overline{X2}$

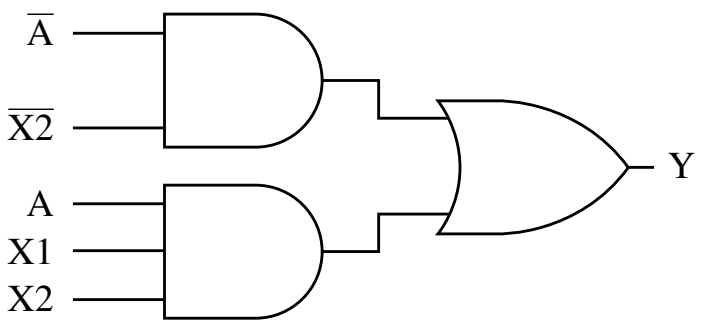


(e) Y = $\overline{A}$ $\overline{X2}$ + A X1 X2

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.35



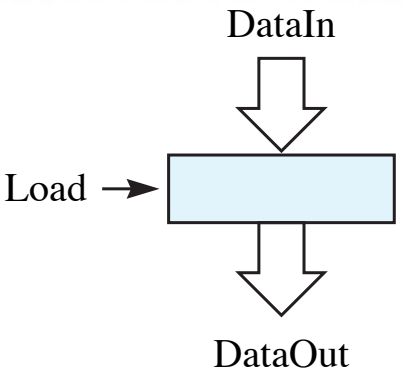


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

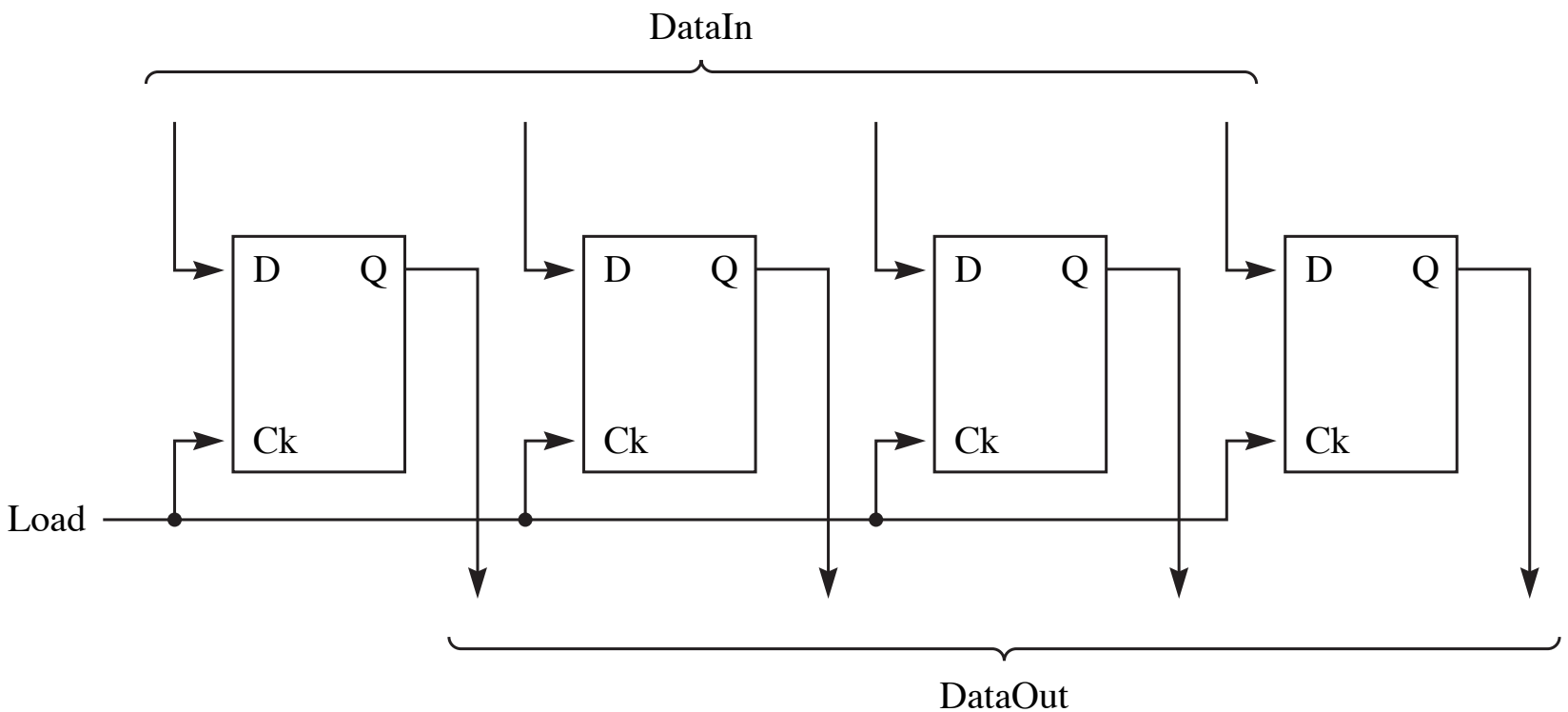
Register

- An example is the 16-bit accumulator in the Pep/8 CPU
- Constructed as an array of D flip-flops with a Load line that connects to each Ck input
- Data is clocked into the register in parallel

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) Block diagram.



(b) Implementation with D flip-flops.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Bus

- A *bus* is a group of wires connecting two subsystems
- With a *unidirectional* bus, data can flow in only one direction
- With a *bidirectional* bus, data can flow in both directions

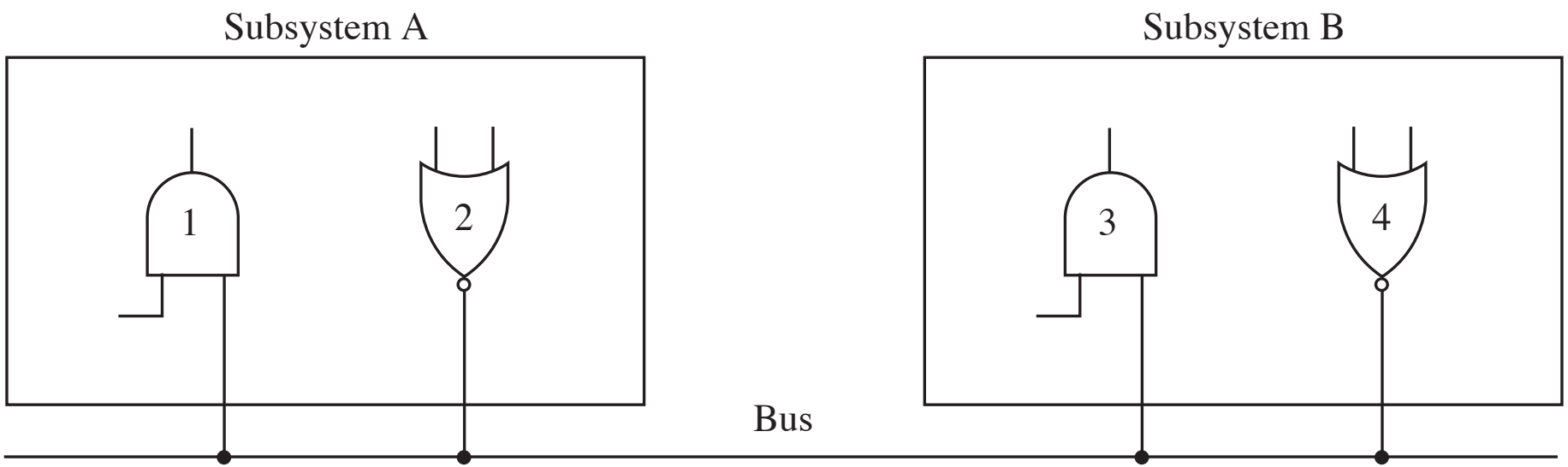
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Bidirectional bus

- Requires only half the number of wires between subsystems
- Problem: You can connect the inputs of two gates, but you cannot connect the outputs of two gates
- Solution: The tri-state buffer

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.37



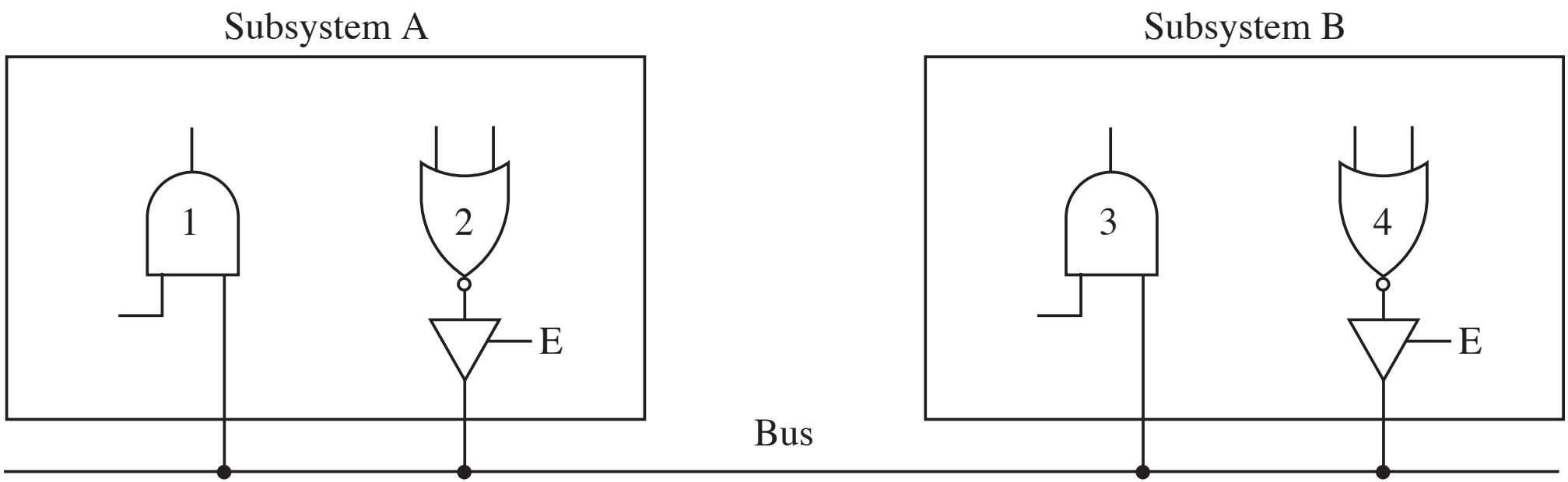
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.38

E	*a*	*x*
0	0	Disconnected
0	1	Disconnected
1	0	0
1	1	1

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.39

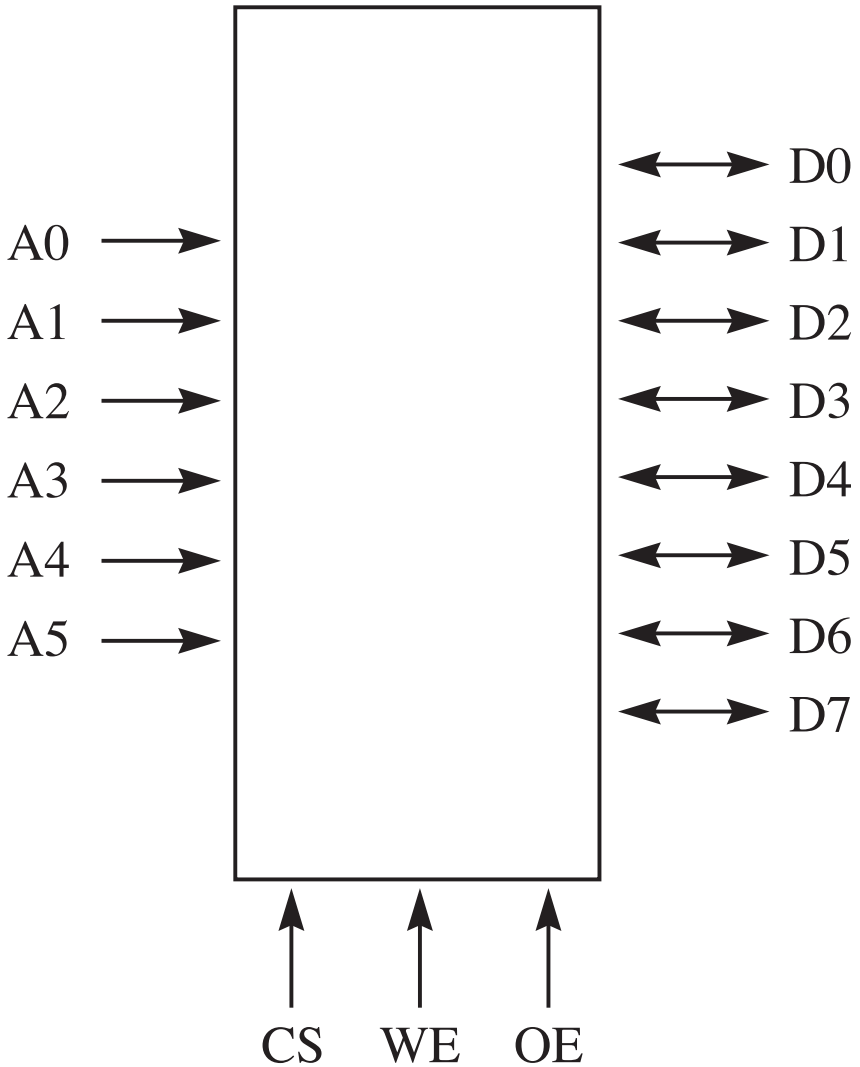


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

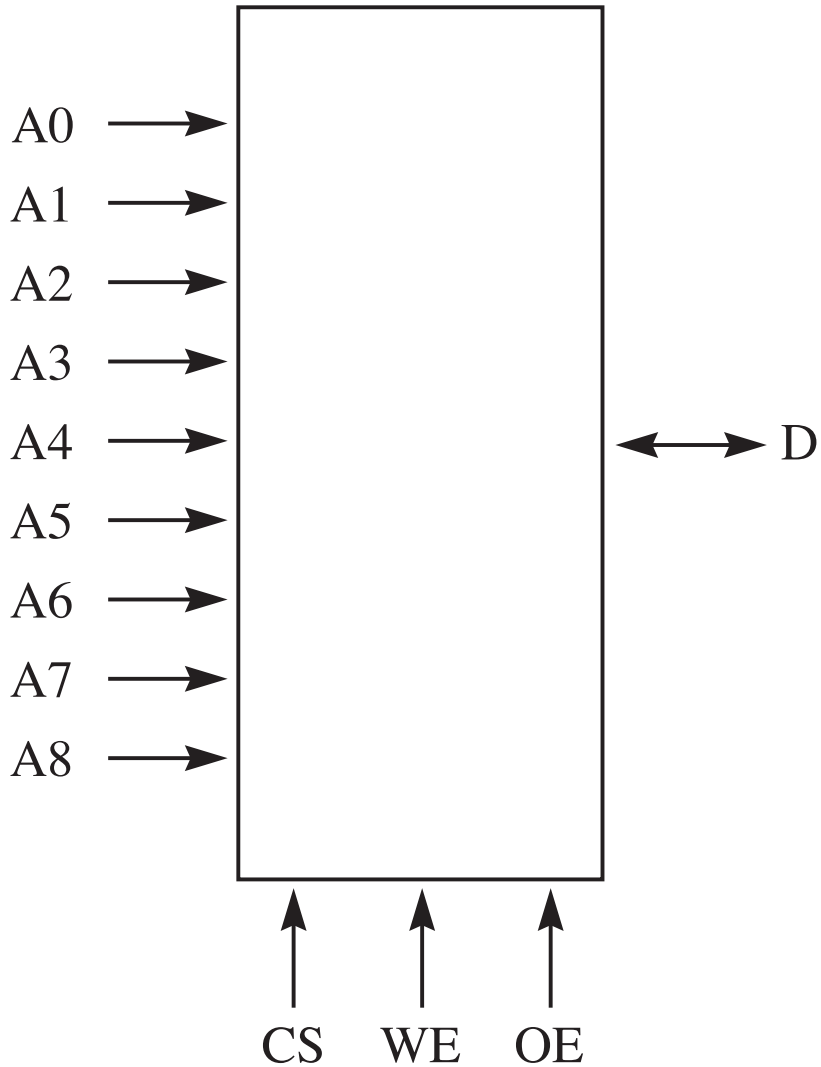
Memory subsystems

- CS: Chip select, to enable or select the memory chip
- WE: Write enable, to write or store a memory word to the chip
- OE: Output enable, to enable the output buffer to read a word from the chip

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(a) 64 × 8 bit memory chip.



(a) 512 × 1 bit memory chip.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Memory access

- To store a word (memory write)
  - Select chip by setting CS to 1
  - Put data and address on the bus and set WE to 1
- To retrieve a word (memory read)
  - Select chip by setting CS to 1
  - Put address on the bus, set OE to 1, and read the data on the bus

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

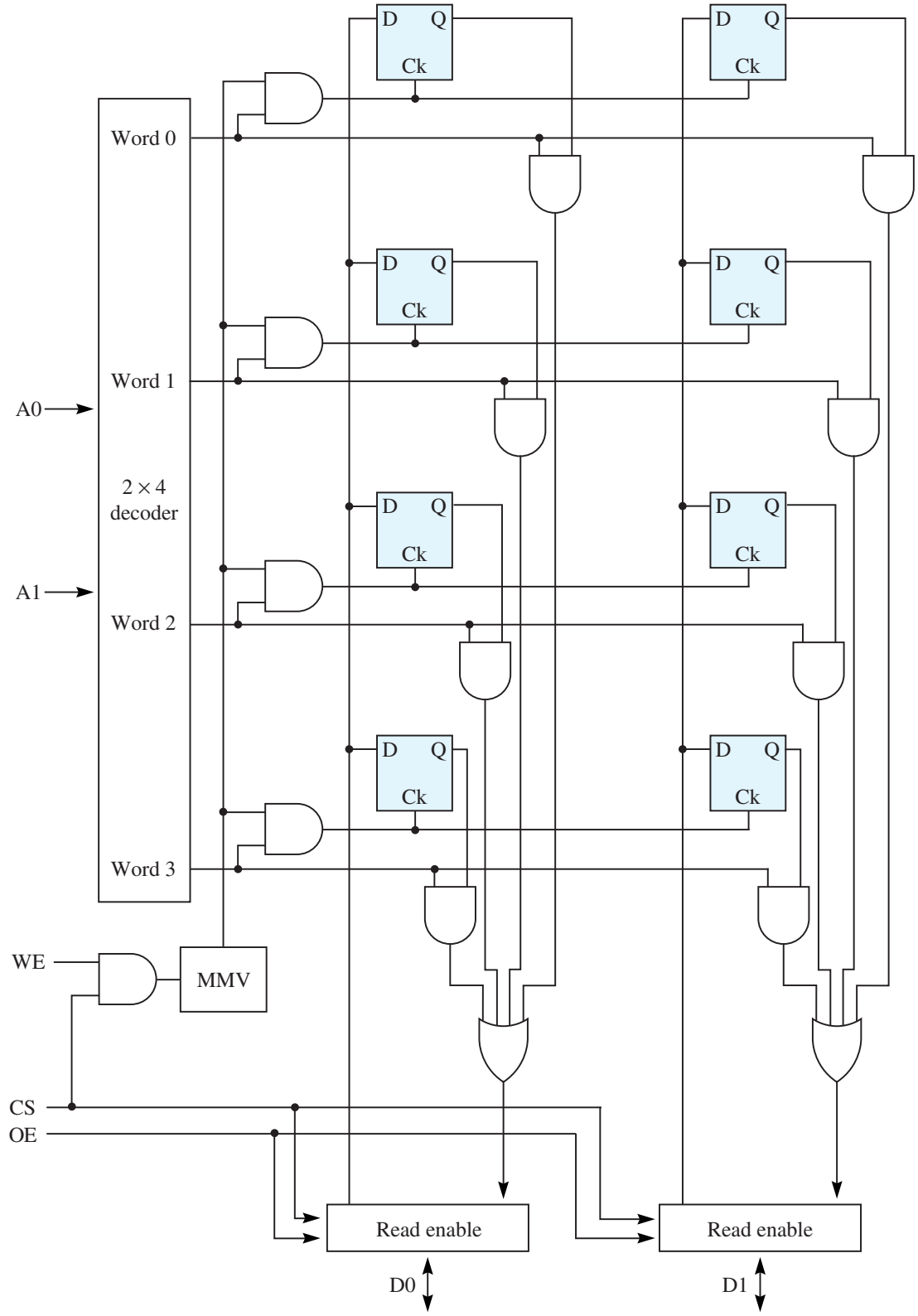
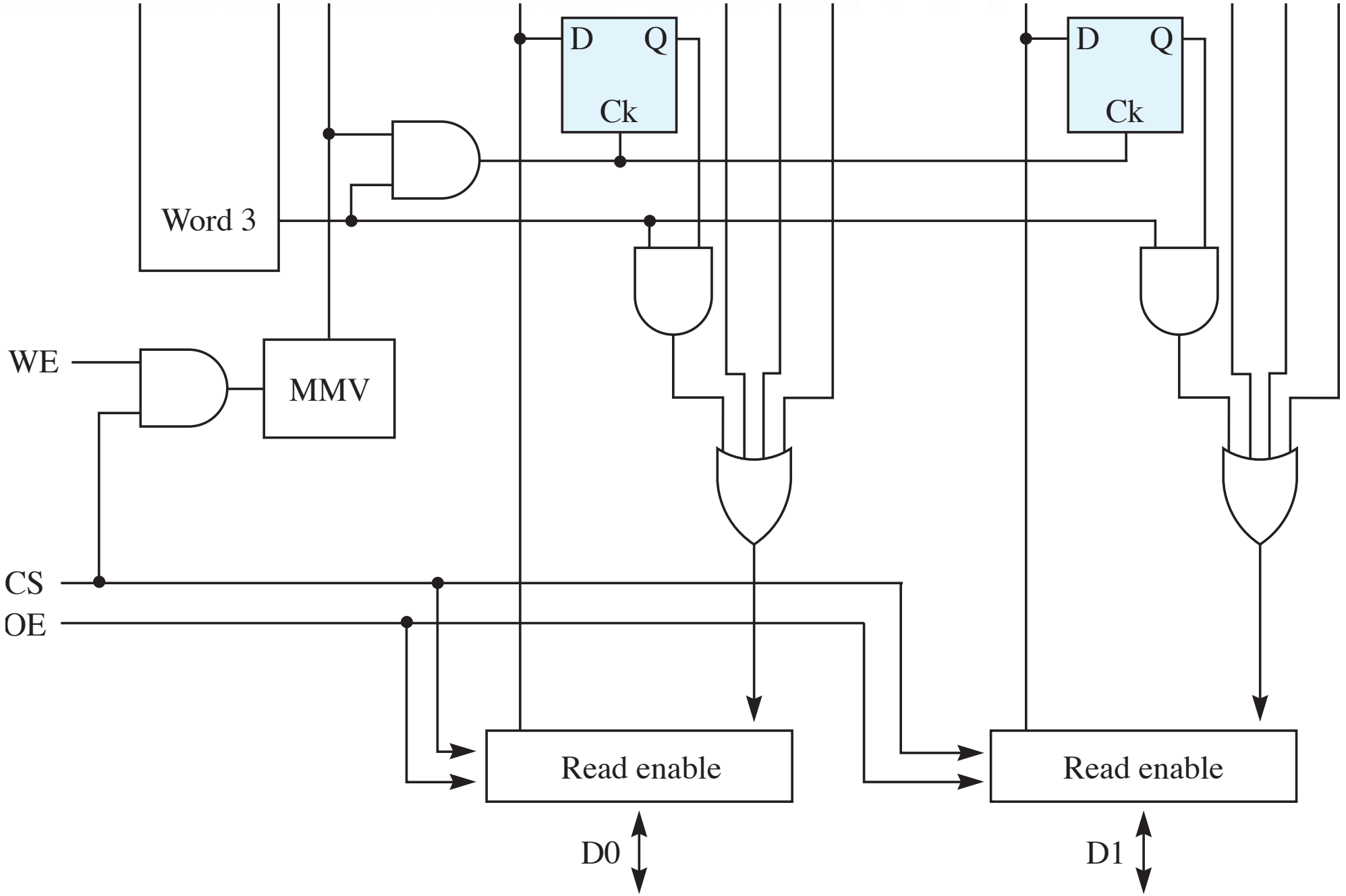

D Q
Ck
Word 0
Word 1
Word 2
Word 3
A0
A1
2 × 4
decoder
WE
MMV
CS
OE
Read enable
Read enable
D0
D1

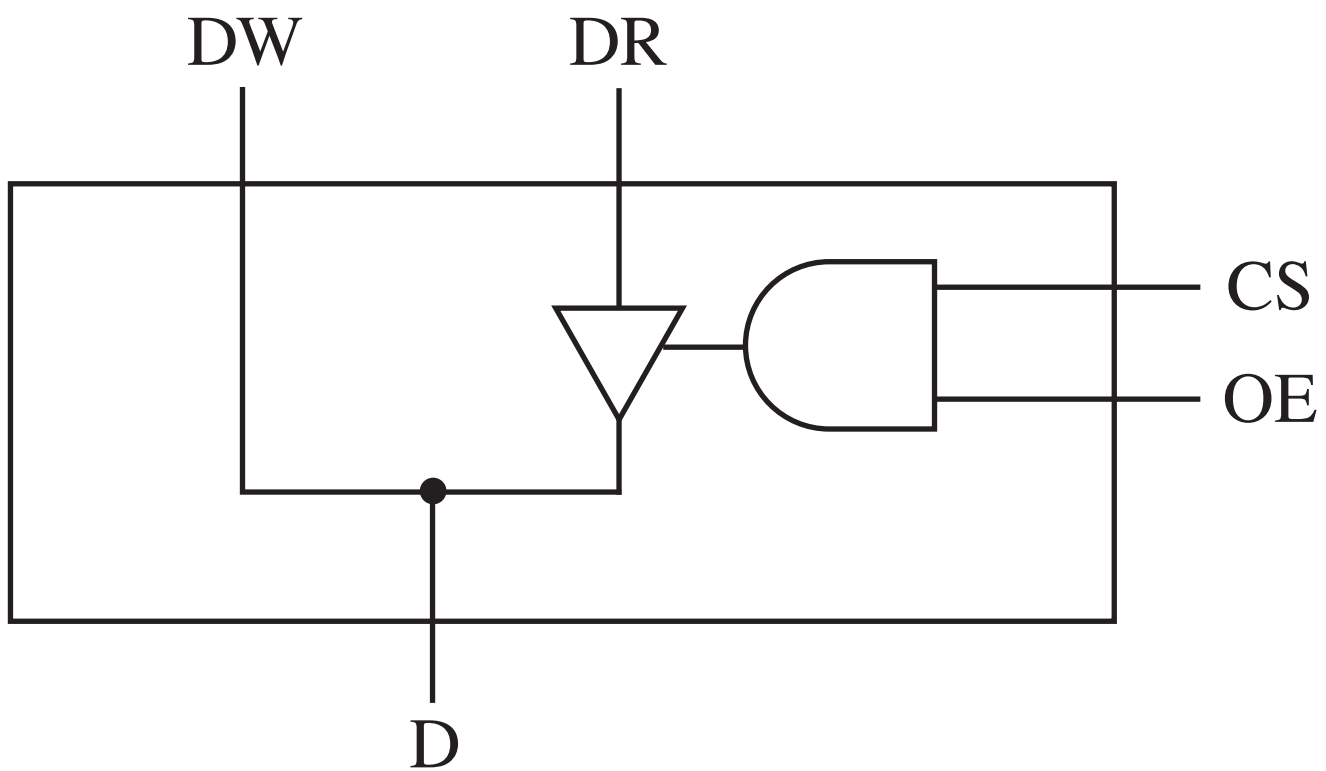
Figure 11.41

Figure 11.41
(Expanded)



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.42



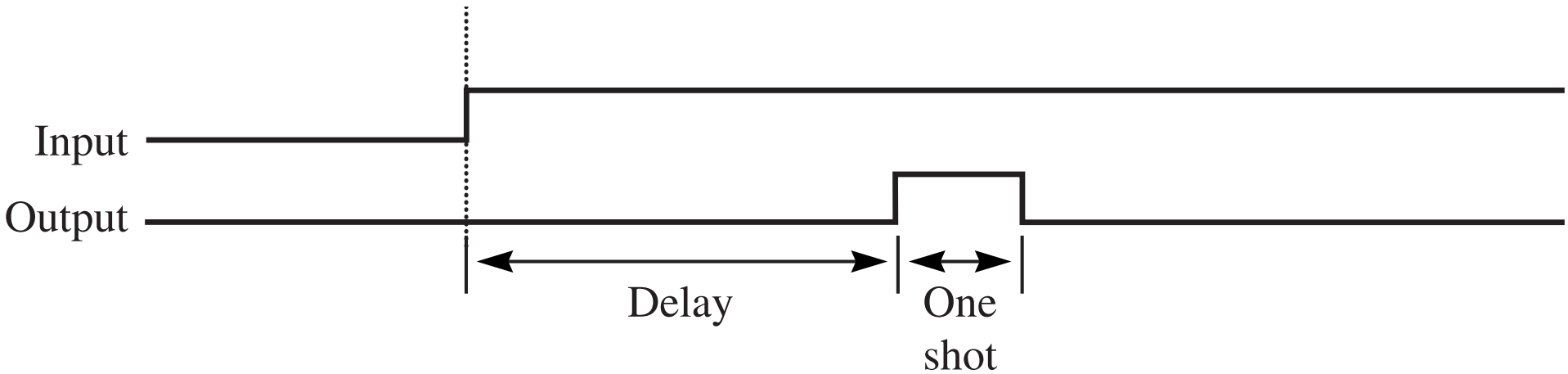
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.43

CS	OE	Operation
0	×	Disconnected
1	0	Disconnected
1	1	Connect DR to D

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.44



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Memory types

- SRAM: Static random access memory
- DRAM: Dynamic RAM
- ROM: Read-only memory
- PROM: Programmable ROM
- EPROM: Erasable PROM
- EEPROM: Electrically erasable PROM
- Flash memory: A type of EEPROM

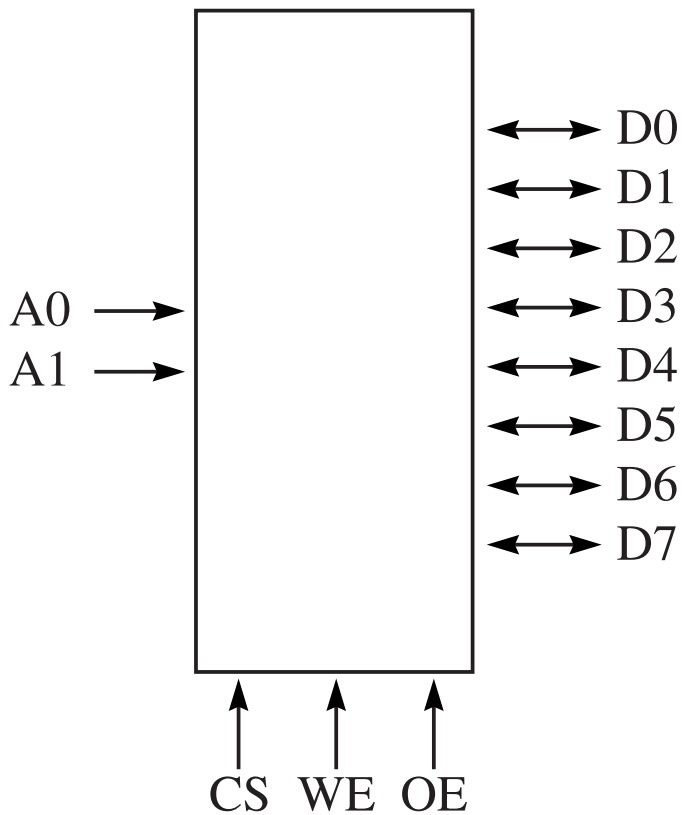
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Constructing memory subsystems

- Two design problems
  - How to combine several $n \times m$ chips to make an $n \times k$ module where k is greater than m
  - How to combine several $n \times m$ chips to make an $l \times m$ module where l is greater than n — the address decoding problem

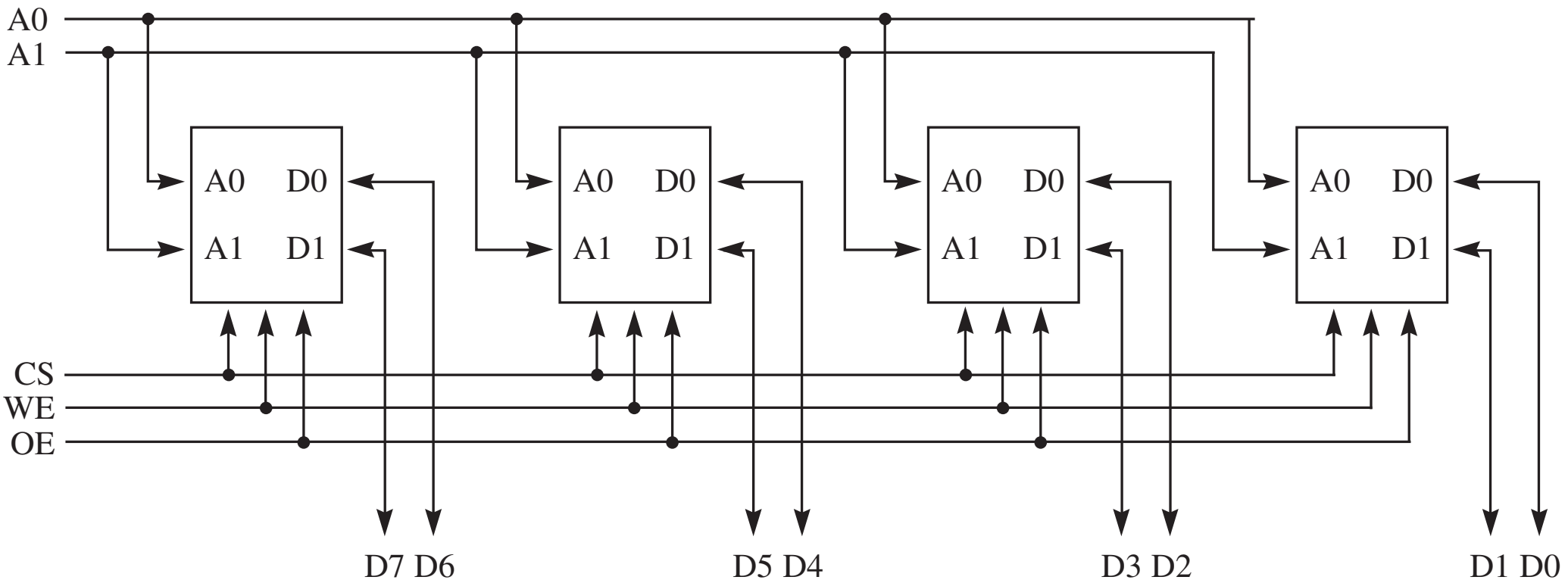
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.45(a)



(a) Block diagram.

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



(b) Implementation.

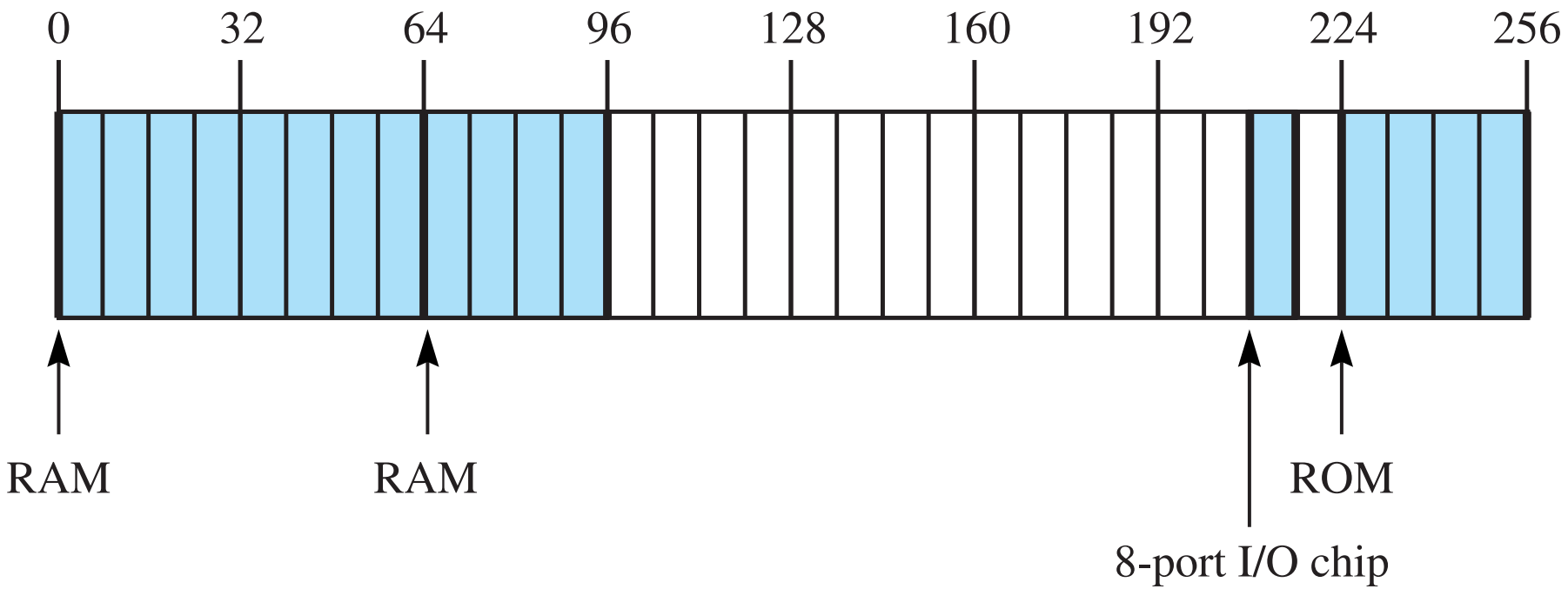
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Address decoding

- An example with 8 address lines and four chips in an address space of 256 bytes
  - 64-byte RAM at address 0
  - 32-byte RAM at address 64
  - 8-port I/O chip at address 208
  - 32-byte ROM at address 224

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.46

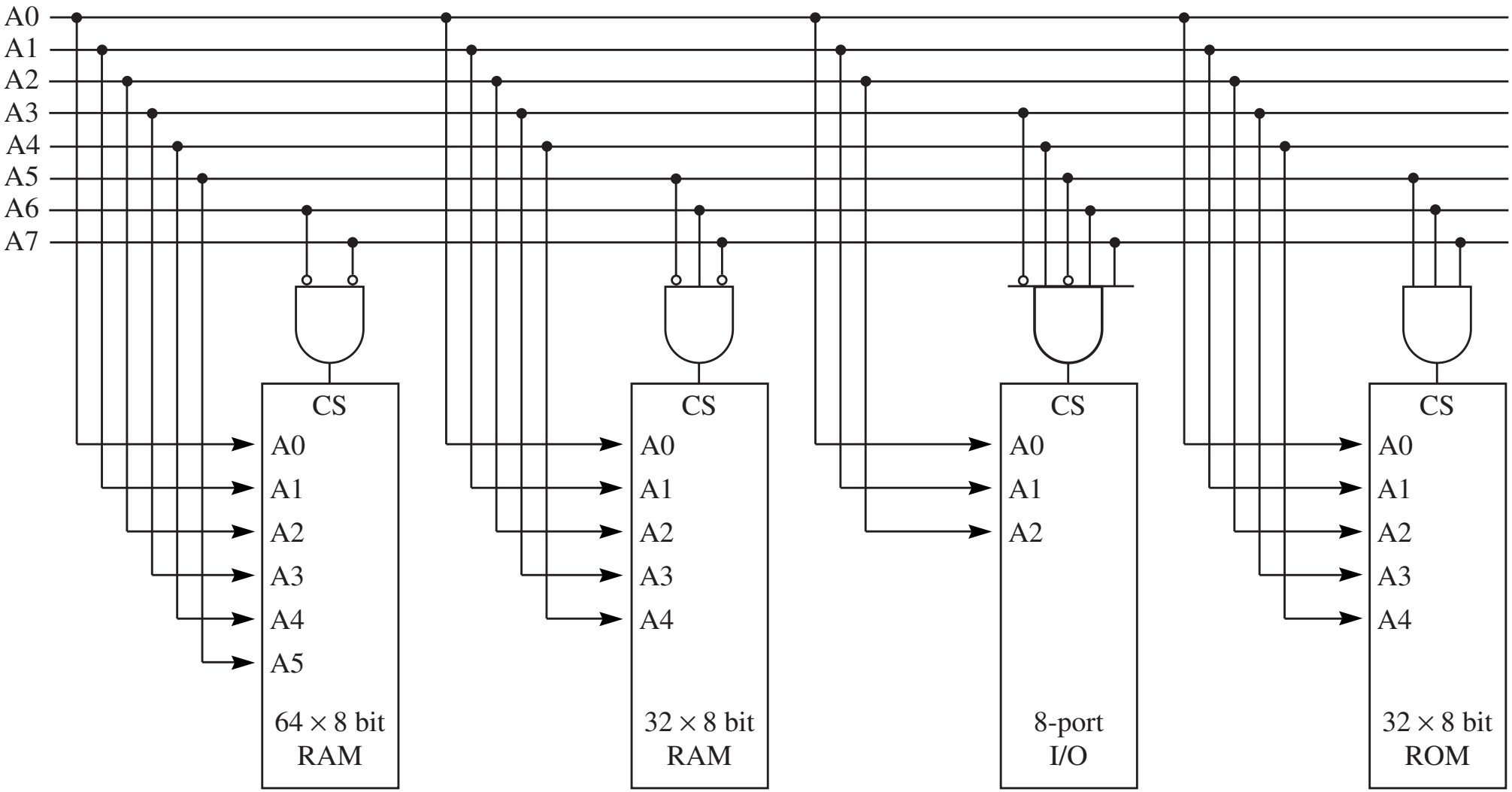


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.47

Device	64 × 8 RAM	32 × 8 RAM	8-port I/O	32 × 8 ROM
Minimum address	0000 0000	0100 0000	1101 0000	1110 0000
Maximum address	0011 1111	0101 1111	1101 0111	1111 1111
General address	00xx xxxx	010x xxxx	1101 0xxx	111x xxxx

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

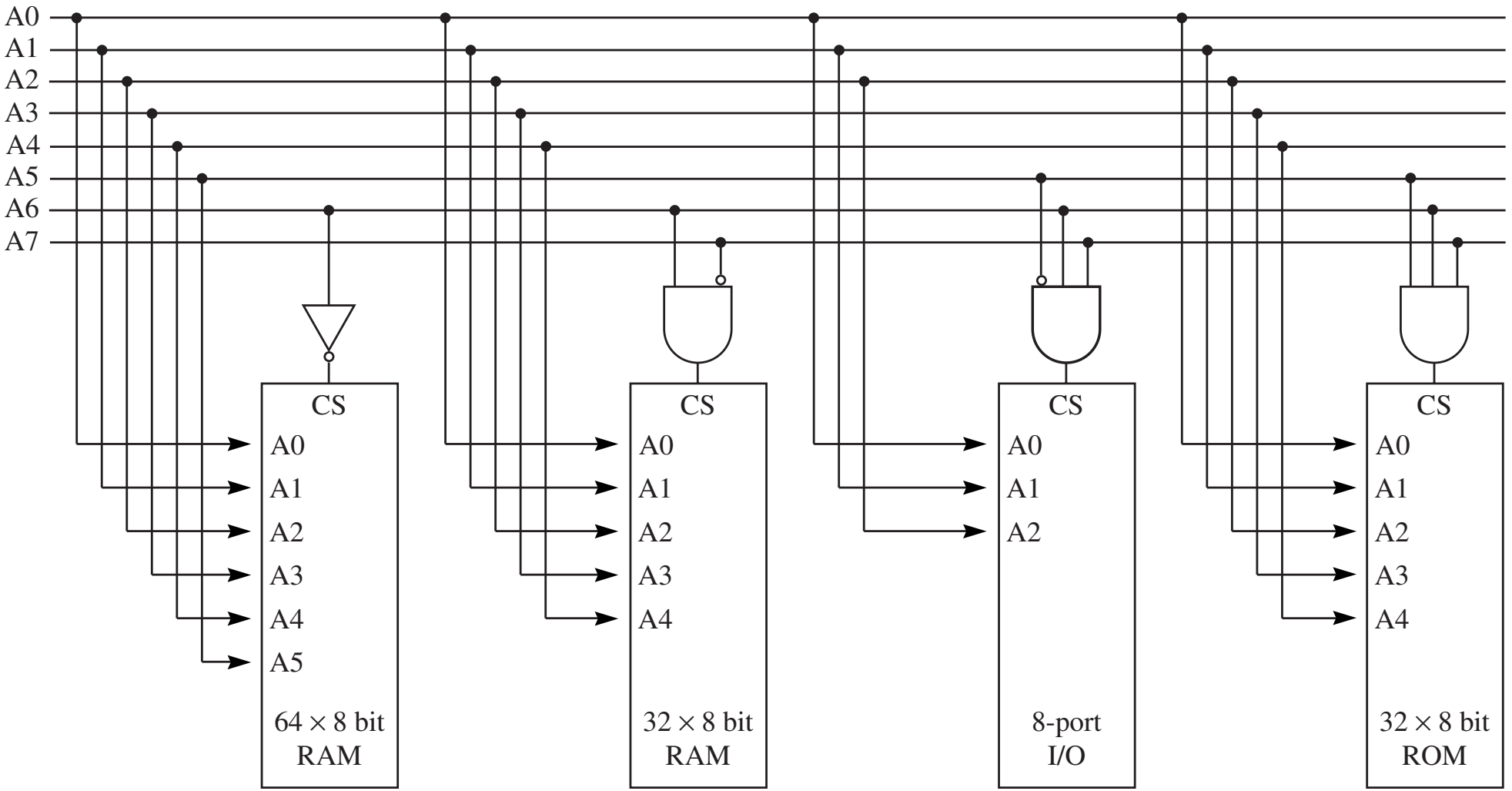


© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Partial address decoding

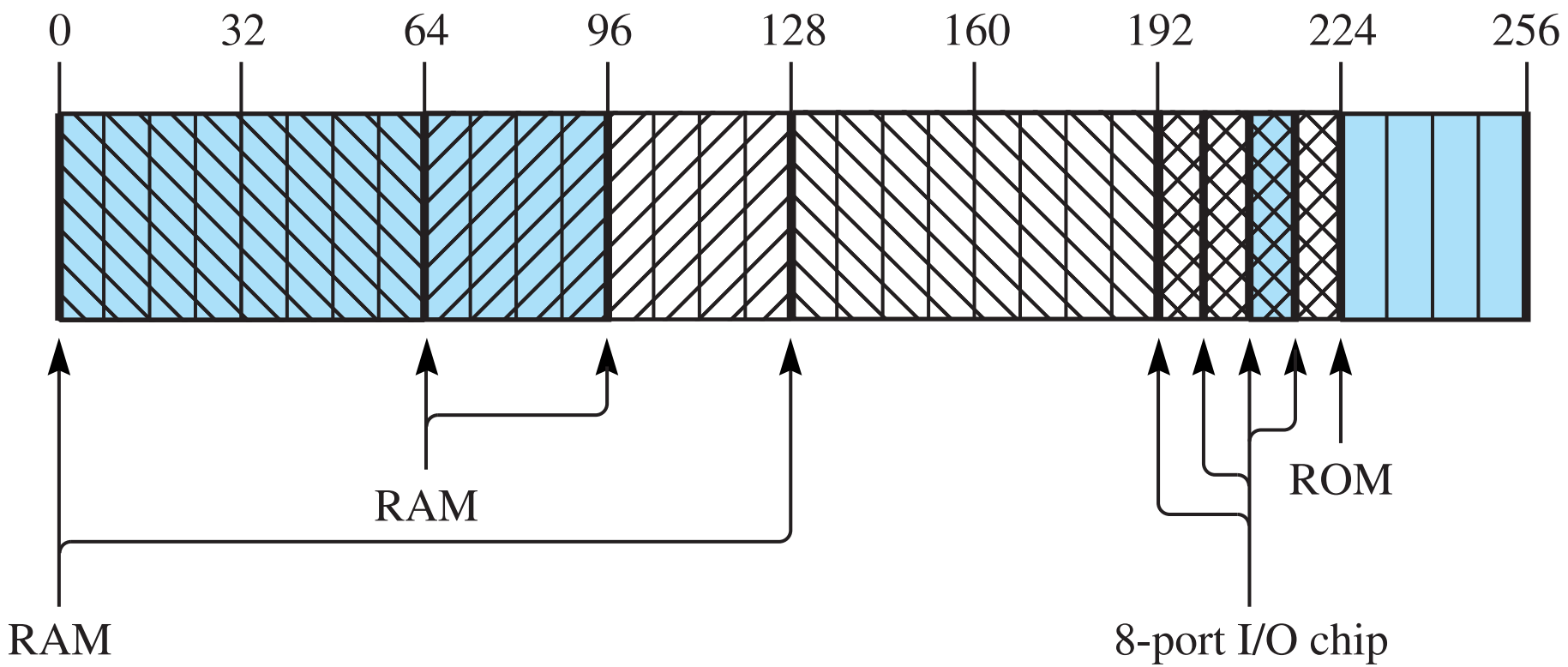
- 0 0 x x x x x x, 64×8 bit RAM
- 0 1 0 x x x x x, 32×8 bit RAM
- 1 1 0 1 0 x x x, 8-port I/O chip
- 1 1 1 x x x x x, 32×8 bit ROM

© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.50



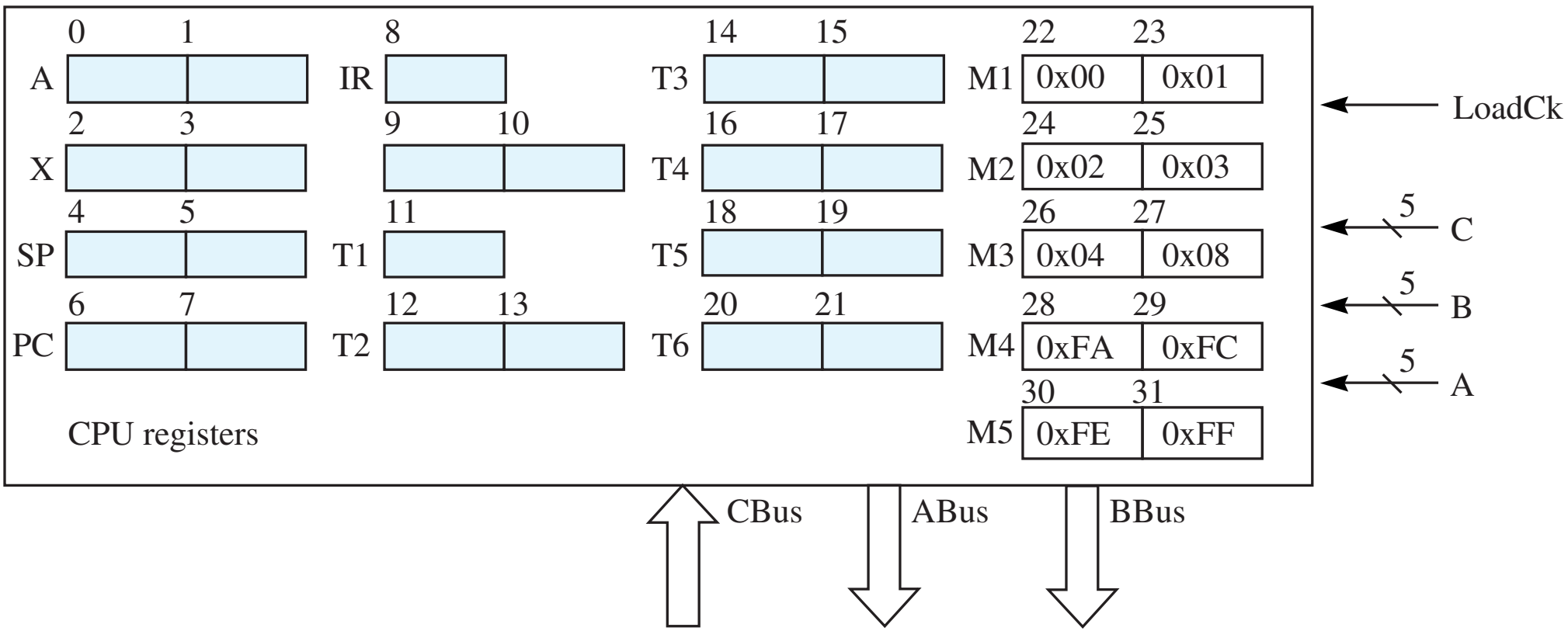
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Two-port register bank

- Implementation of the registers (accumulator, index register, etc.) in the Pep/8 CPU
- The data buses are unidirectional instead of bidirectional
- There are two output ports instead of one

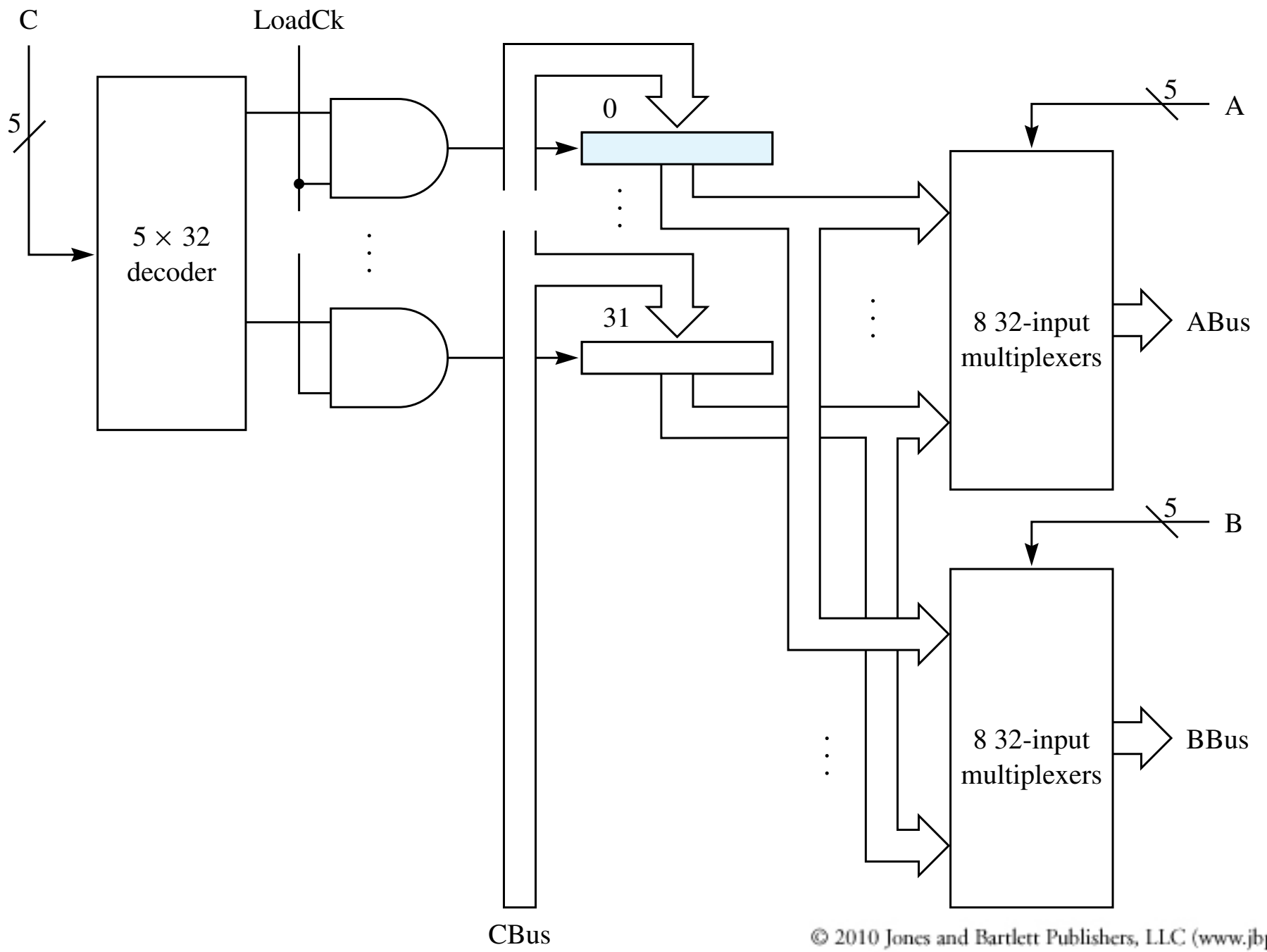
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.51



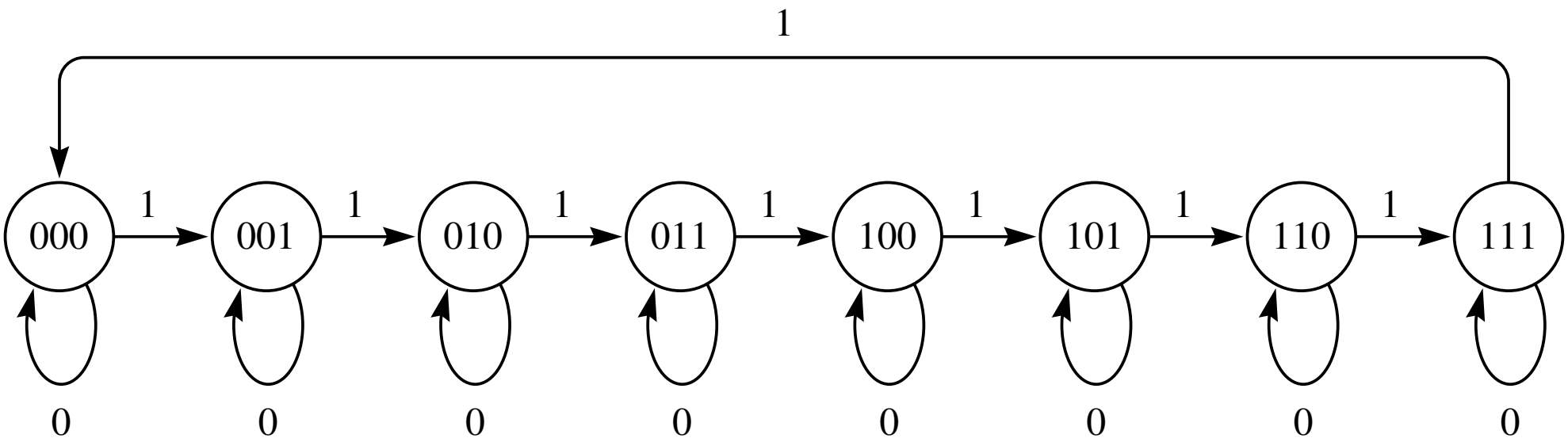
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.52



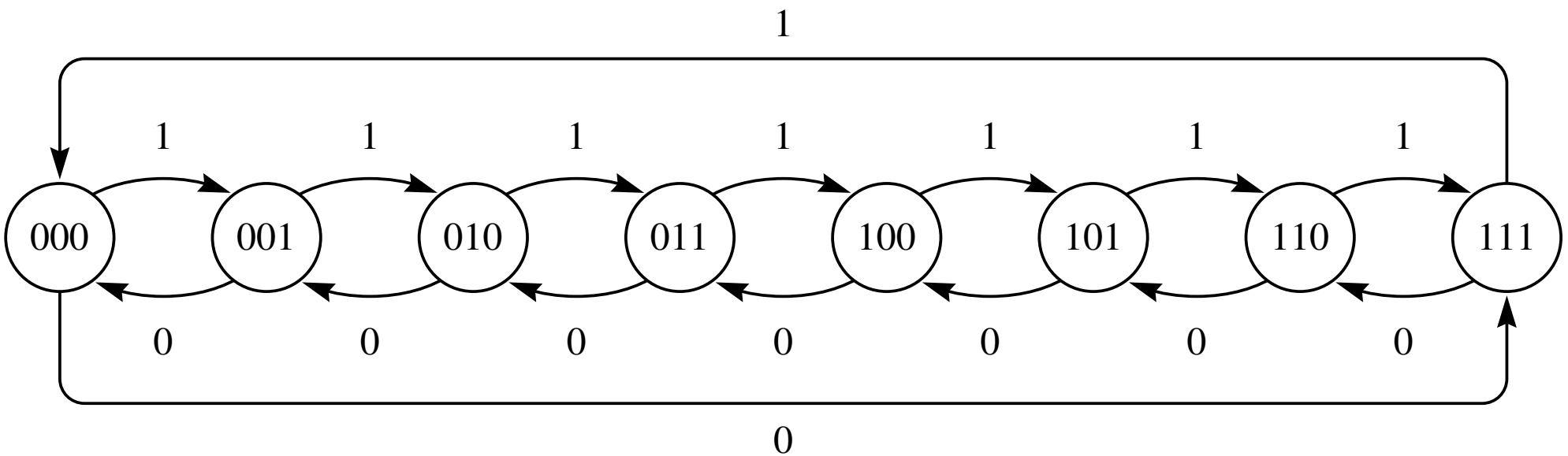
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.53



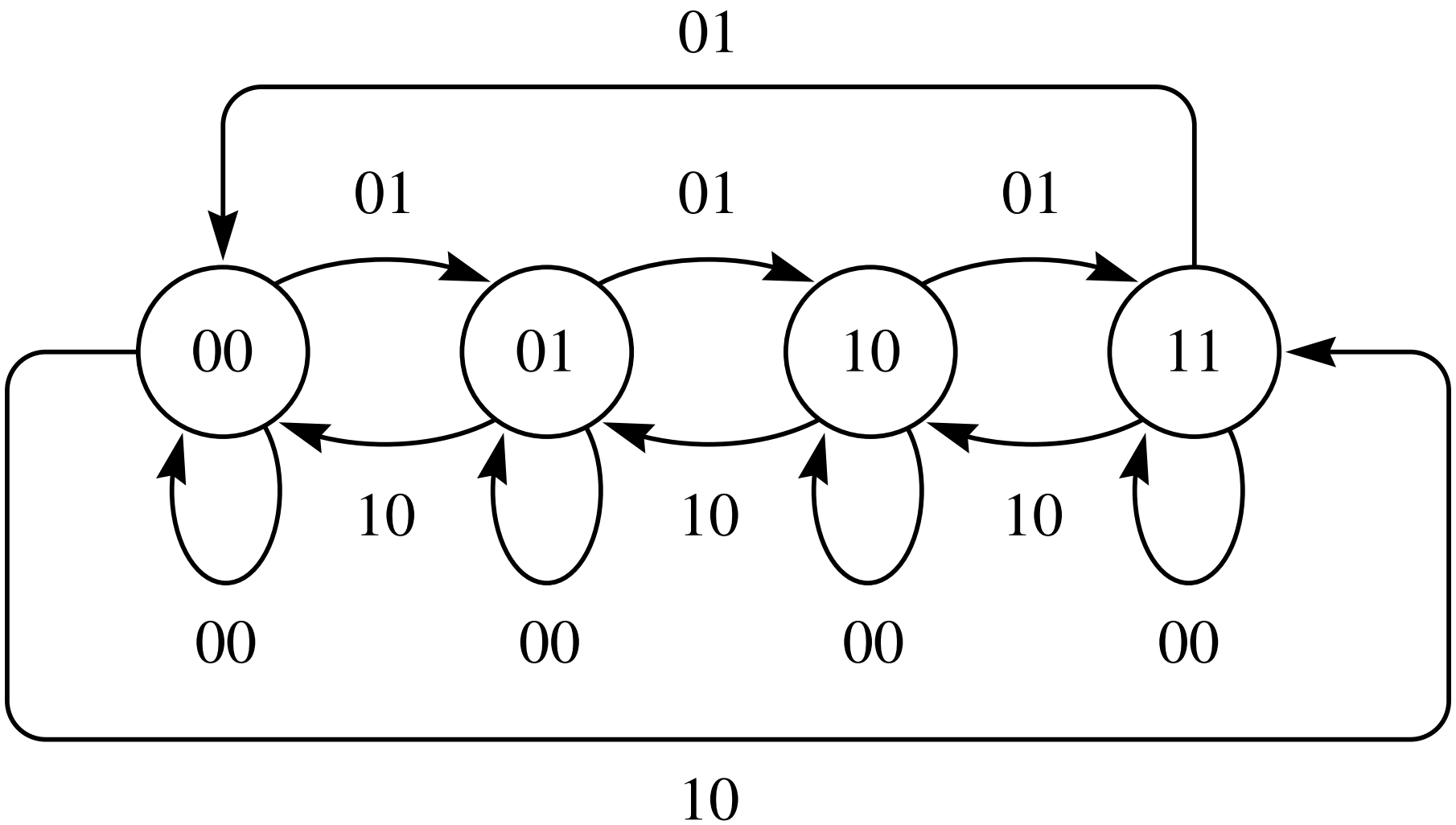
© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.54



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)

Figure 11.55



© 2010 Jones and Bartlett Publishers, LLC (www.jbpub.com)