# SEQUENTIAL NETWORKS

- CANONICAL FORM OF SEQUENTIAL NETWORKS
- LATCHES AND EDGE-TRIGGERED CELLS. D FLIP-FLOP
- TIMING CHARACTERISTICS
- ANALYSIS AND DESIGN OF CANONICAL NETWORKS
- SR, JK and T FLIP-FLOP
- ANALYSIS OF FLIP-FLOP NETWORKS
- DESIGN OF FLIP-FLOP NETWORKS. EXCITATION FUNCTIONS
- SPECIAL STATE ASSIGNMENTS: ONE-FLIP-FLOP-PER-STATE AND SHIFTING REGISTER

# CANONICAL FORM OF SEQUENTIAL NETWORKS (Huffman-Moore)

STATE-TRANSITION FUNCTION $s(t+1) = G(s(t), x(t))$

OUTPUT FUNCTION $z(t) = H(s(t), x(t))$

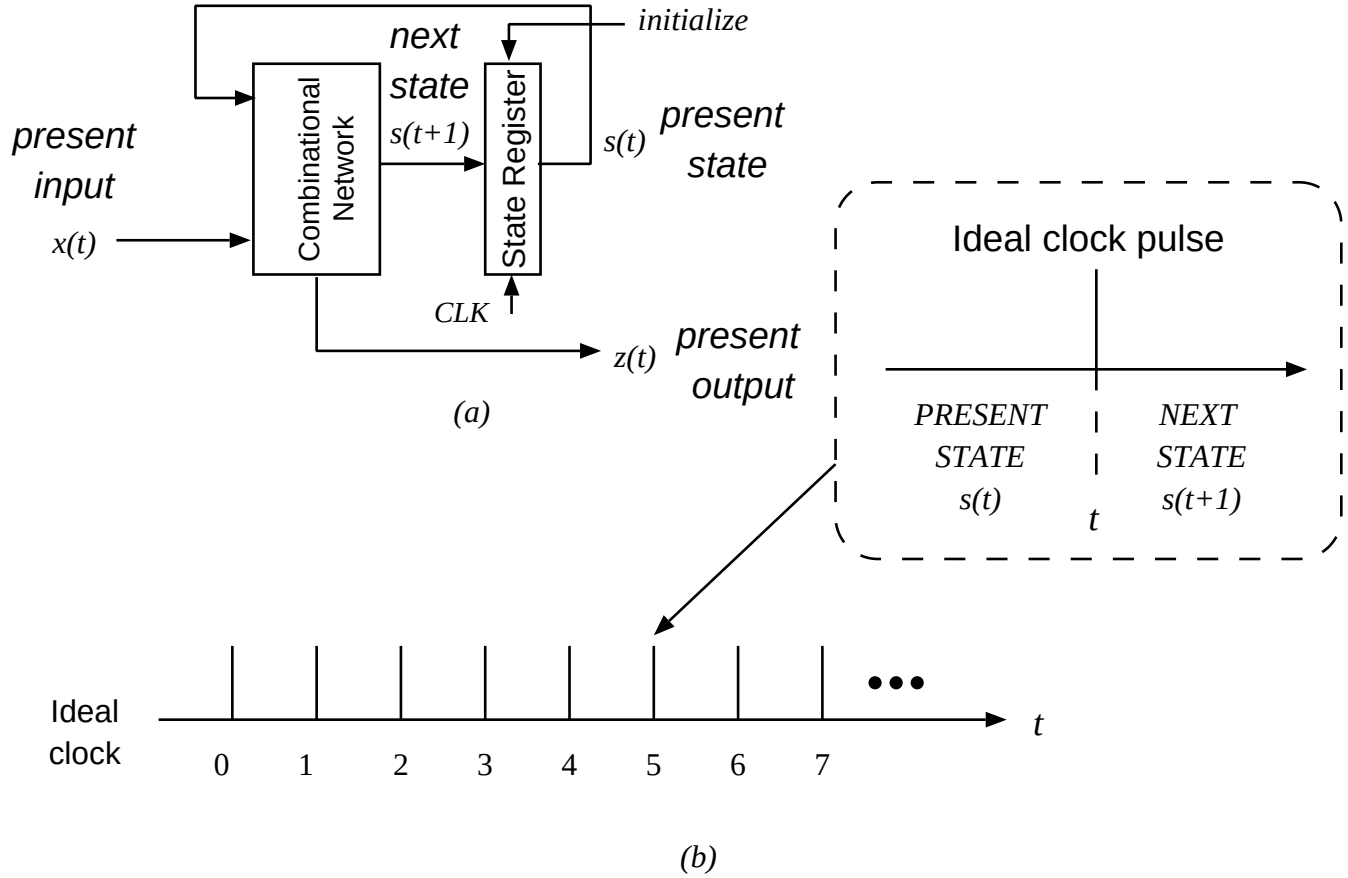



Figure 8.1: a) CANONICAL IMPLEMENTATION OF SEQUENTIAL NETWORK. b) IDEAL CLOCK SIGNAL AND ITS INTERPRETATION.

# MEALY AND MOORE MACHINES

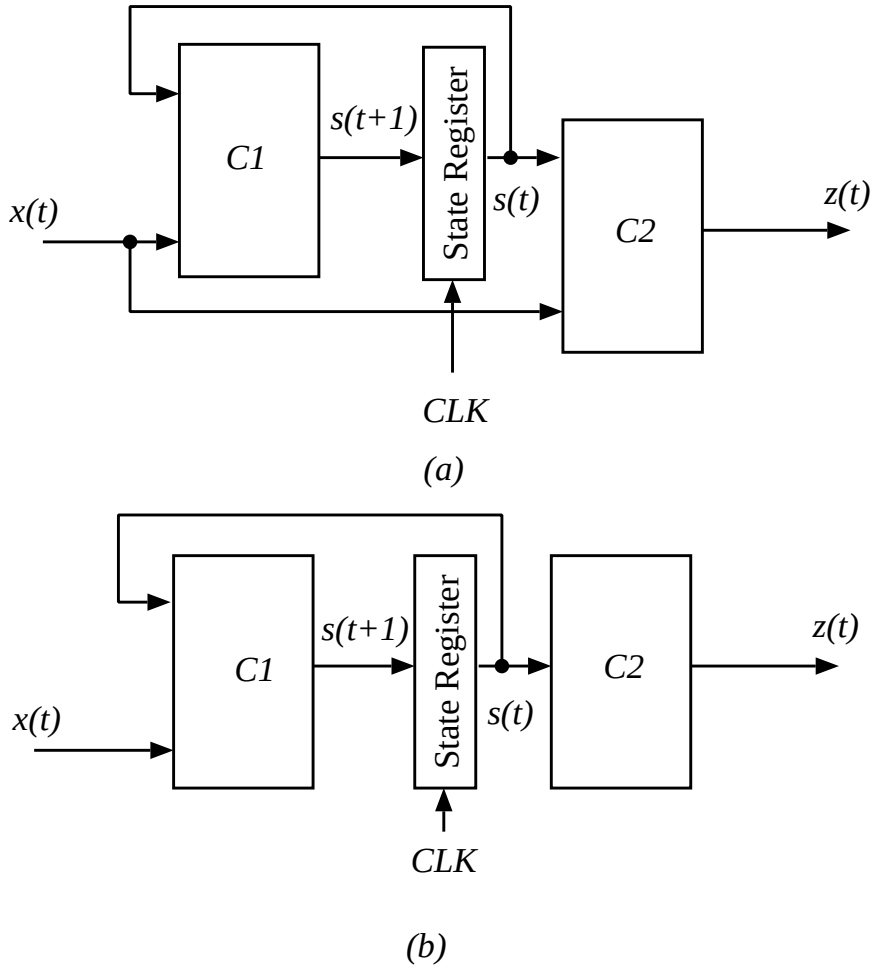



Figure 8.2: CANONICAL IMPLEMENTATIONS: a) MEALY MACHINE. b) MOORE MACHINE.

# HIGH-LEVEL AND BINARY IMPLEMENTATIONS

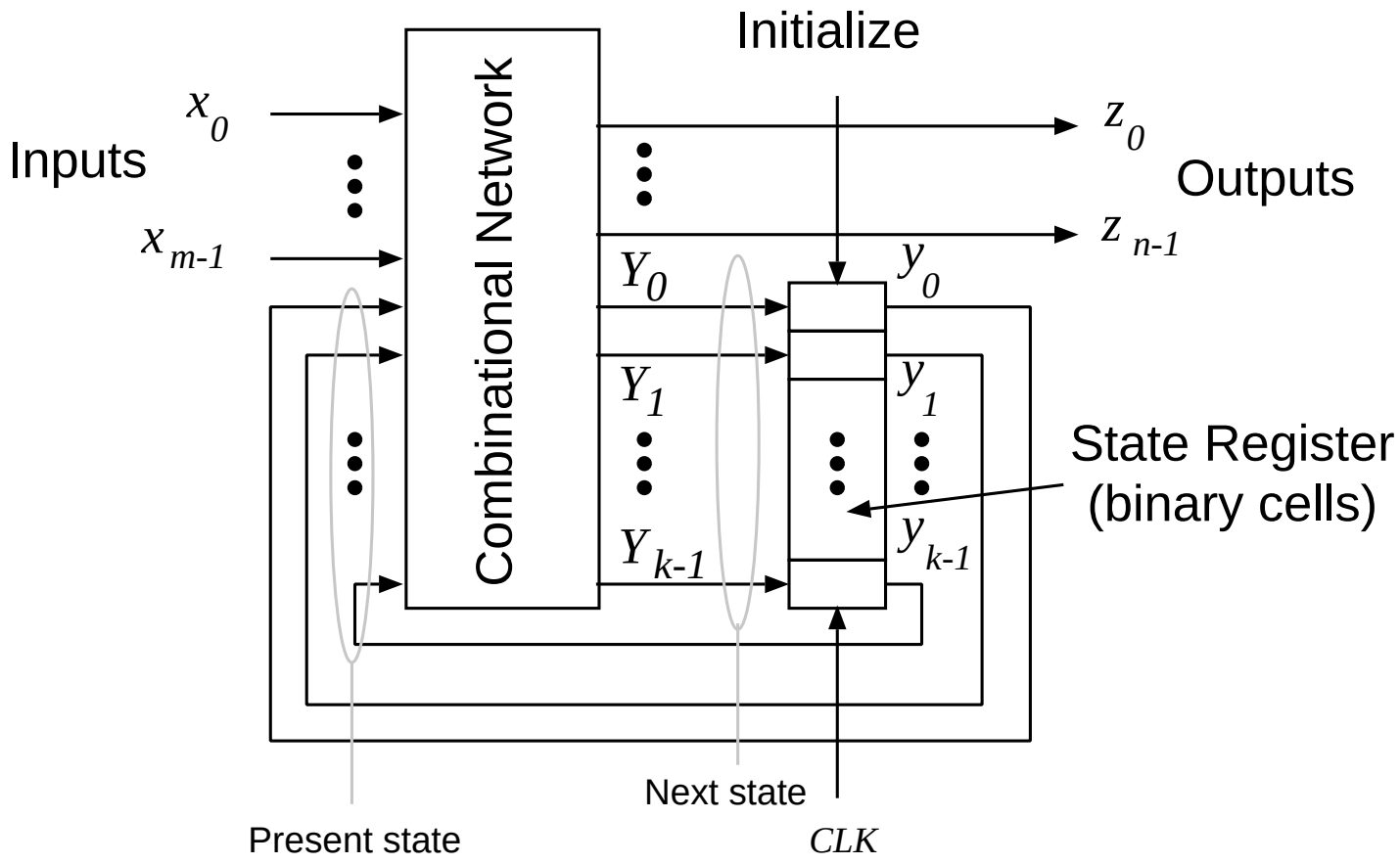



Figure 8.3: CANONICAL IMPLEMENTATION WITH BINARY VARIABLES.

# EXAMPLE 8.1

INPUT: $\underline{x}(t) = (x_1, x_0), \quad x_i \in \{0, 1\}$

OUTPUT: $z(t) \in \{0, 1\}$

STATE: $\underline{y}(t) = (y_3, y_2, y_1, y_0), \quad y_i(t) \in \{0, 1\}$

INITIAL STATE: $\underline{y}(0) = (0, 0, 0, 0)$

FUNCTION: THE TRANSITION AND OUTPUT FUNCTIONS

$$
\begin{aligned}
Y_3 &= y_2 x_1' x_0 \\
Y_2 &= (y_1 + y_2) x_0' + y_3 x_1 \\
Y_1 &= (y_0 + y_3) x_1' x_0 + (y_0 + y_1) x_1 \\
Y_0 &= (y_0 + y_3) x_0' y_1 x_1' x_0 + y_2 x_1 \\
z &= y_3 + y_2 + y_1 + y_0
\end{aligned}
$$

# EXAMPLE 8.1 (cont.)

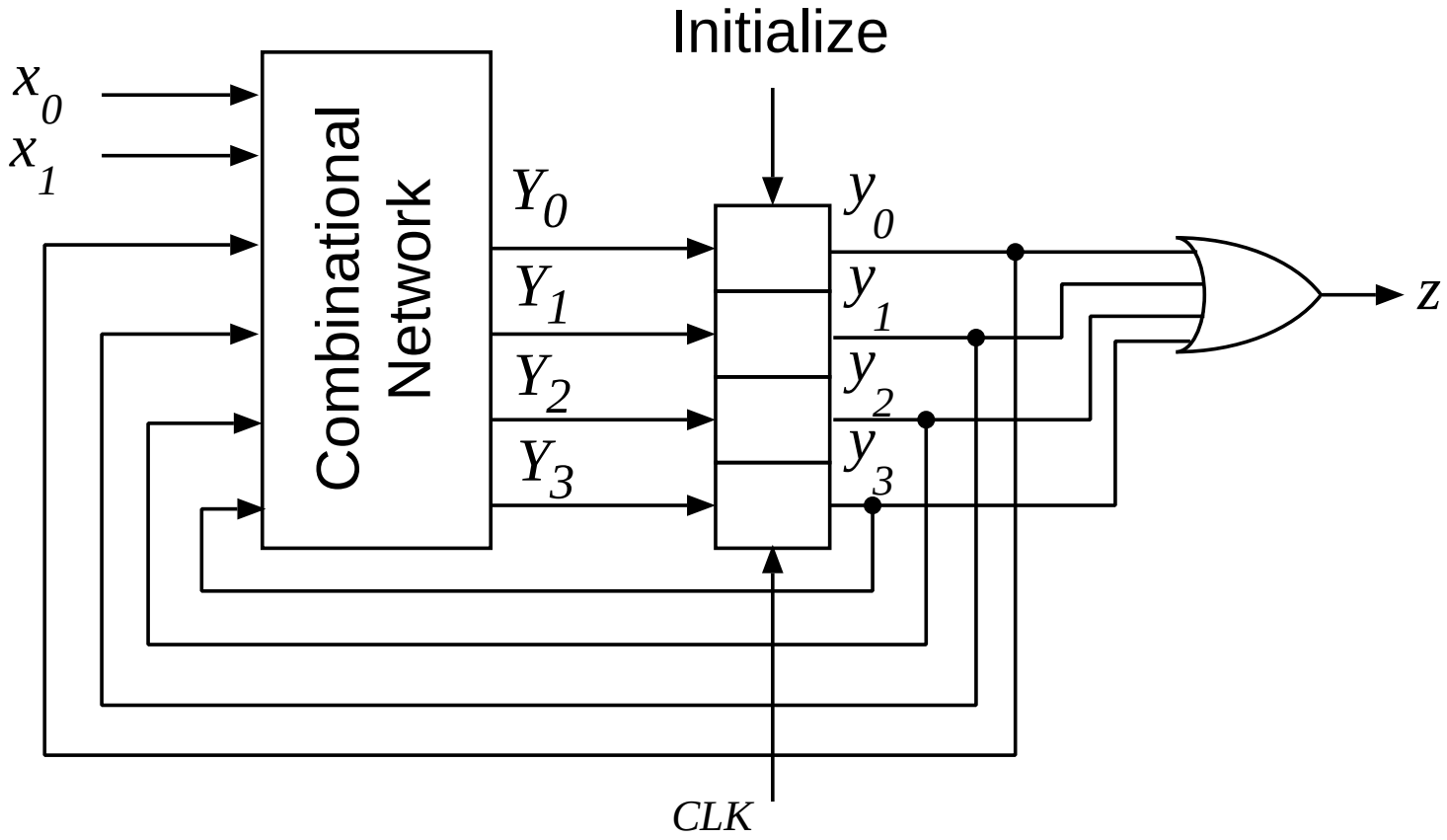



Figure 8.4: CANONICAL NETWORK FOR EXAMPLE 8.1.

# CLOCK

- CLOCK PERIOD $T$
- CLOCK FREQUENCY $f = 1/T$
- (CLOCK) PULSE WIDTH $t_w$

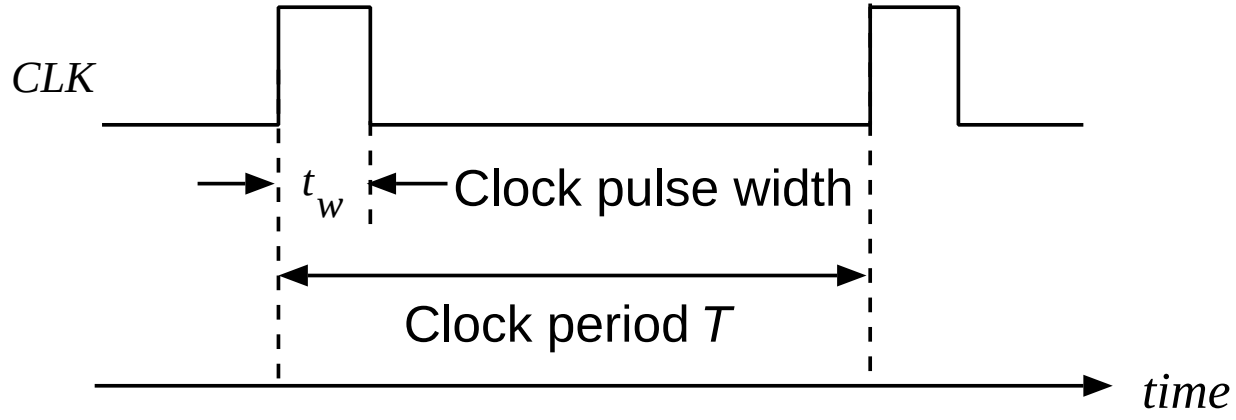



Figure 8.5: PULSE WIDTH AND CLOCK PERIOD.

# GATED LATCH - FIRST TRY

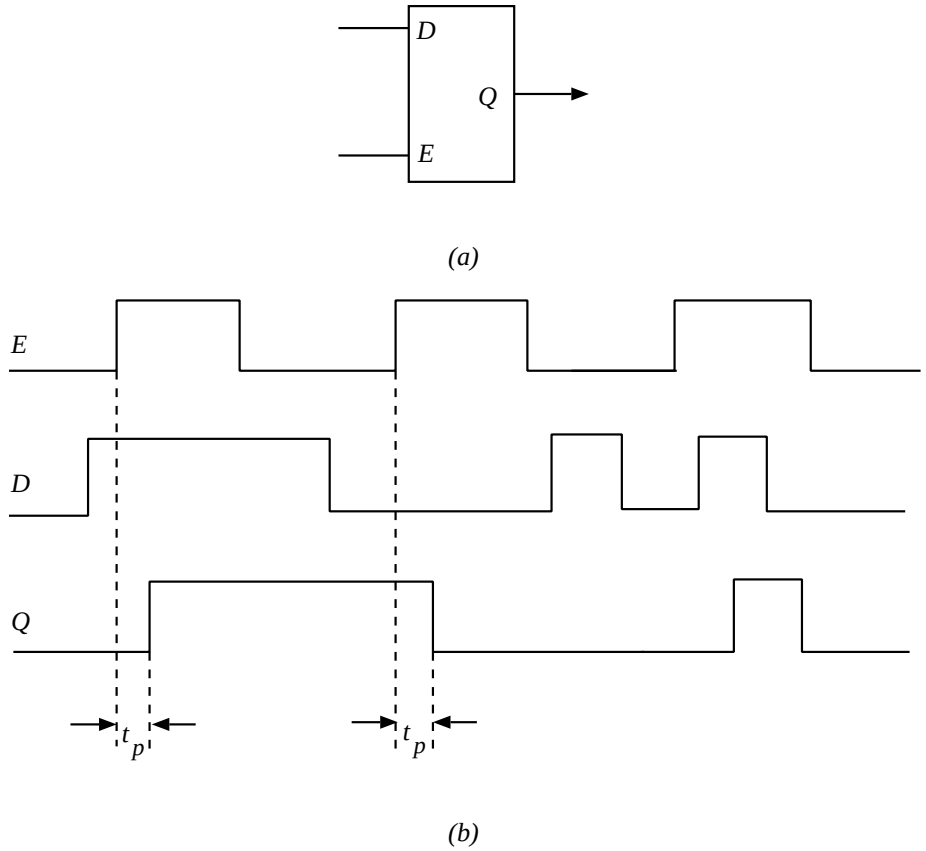
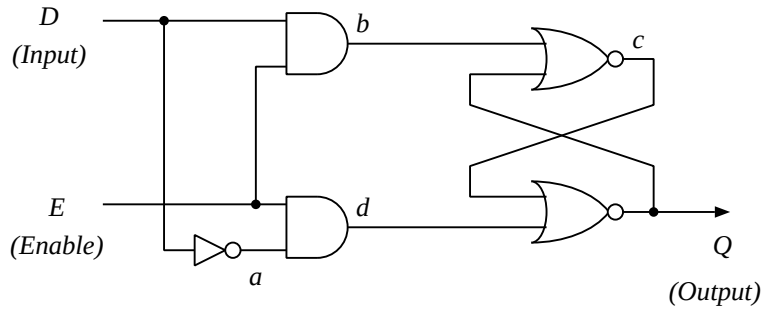



Figure 8.6: a) GATED-LATCH. b) TIMING BEHAVIOR.

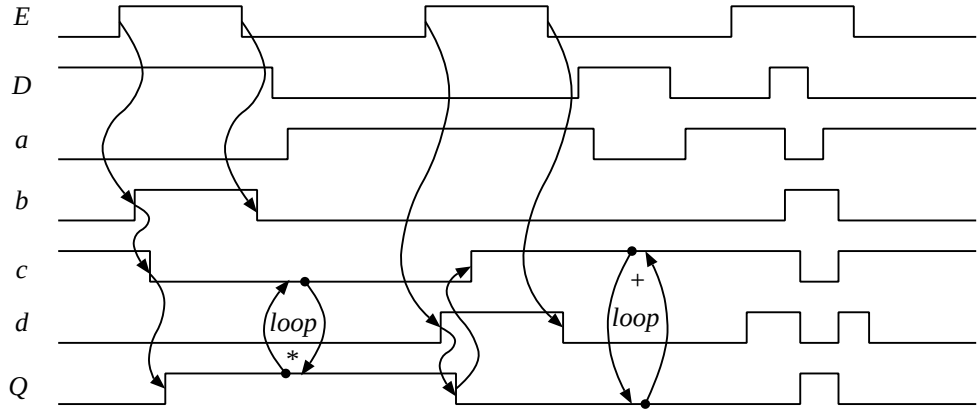
$$Q(t + t_p) = D(t) \cdot E(t) + Q(t) \cdot E'(t)$$

- LEVEL-SENSITIVE: when $E = 1$ then $Q = D$

# NOR-NOR LATCH




(a)




(b)

Figure 8.7: a) IMPLEMENTATION OF GATED-LATCH WITH NOR GATES. b) TIMING DIAGRAM.

# LIMITATIONS OF GATED-LATCH

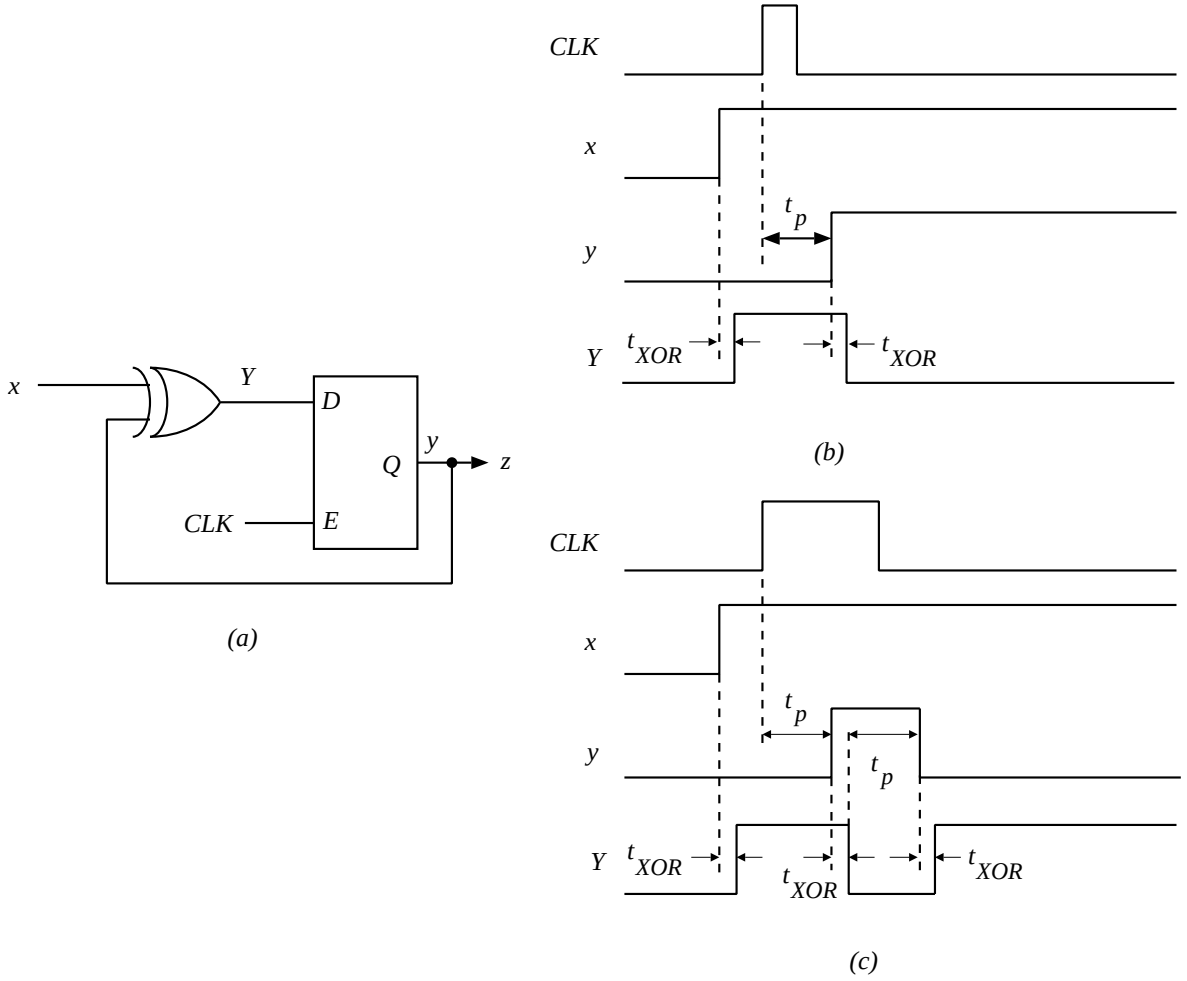



Figure 8.8: a) SEQUENTIAL NETWORK. b) CORRECT TIMING BEHAVIOR. c) INCORRECT TIMING BEHAVIOR.

# A SOLUTION: EDGE-TRIGGERED CELL

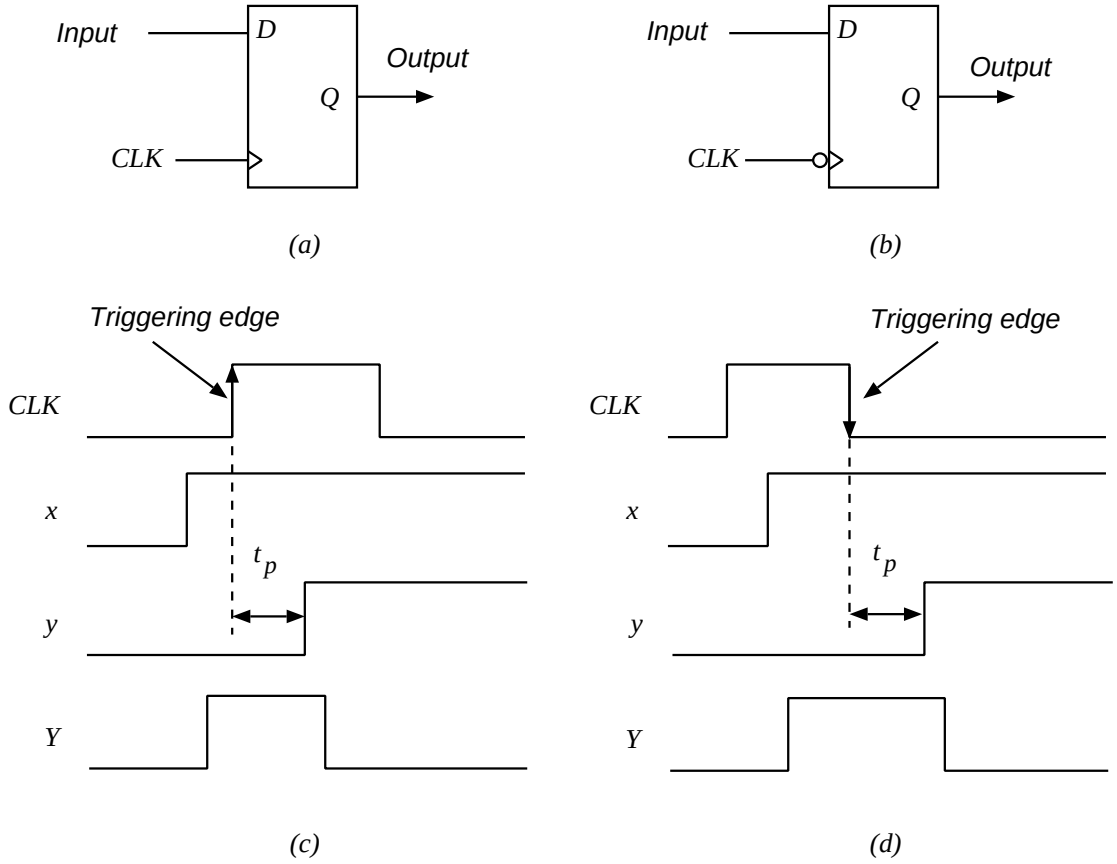



Figure 8.9: EDGE-TRIGGERED CELL: a) LEADING-EDGE-TRIGGERED CELL. b) TRAILING-EDGE-TRIGGERED CELL. c) LEADING-EDGE-TRIGGERED CELL IN NETWORK OF Figure 8.8. d) TRAILING-EDGE-TRIGGERED CELL IN NETWORK OF Figure 8.8.

# IMPLEMENTATION: MASTER-SLAVE CELL

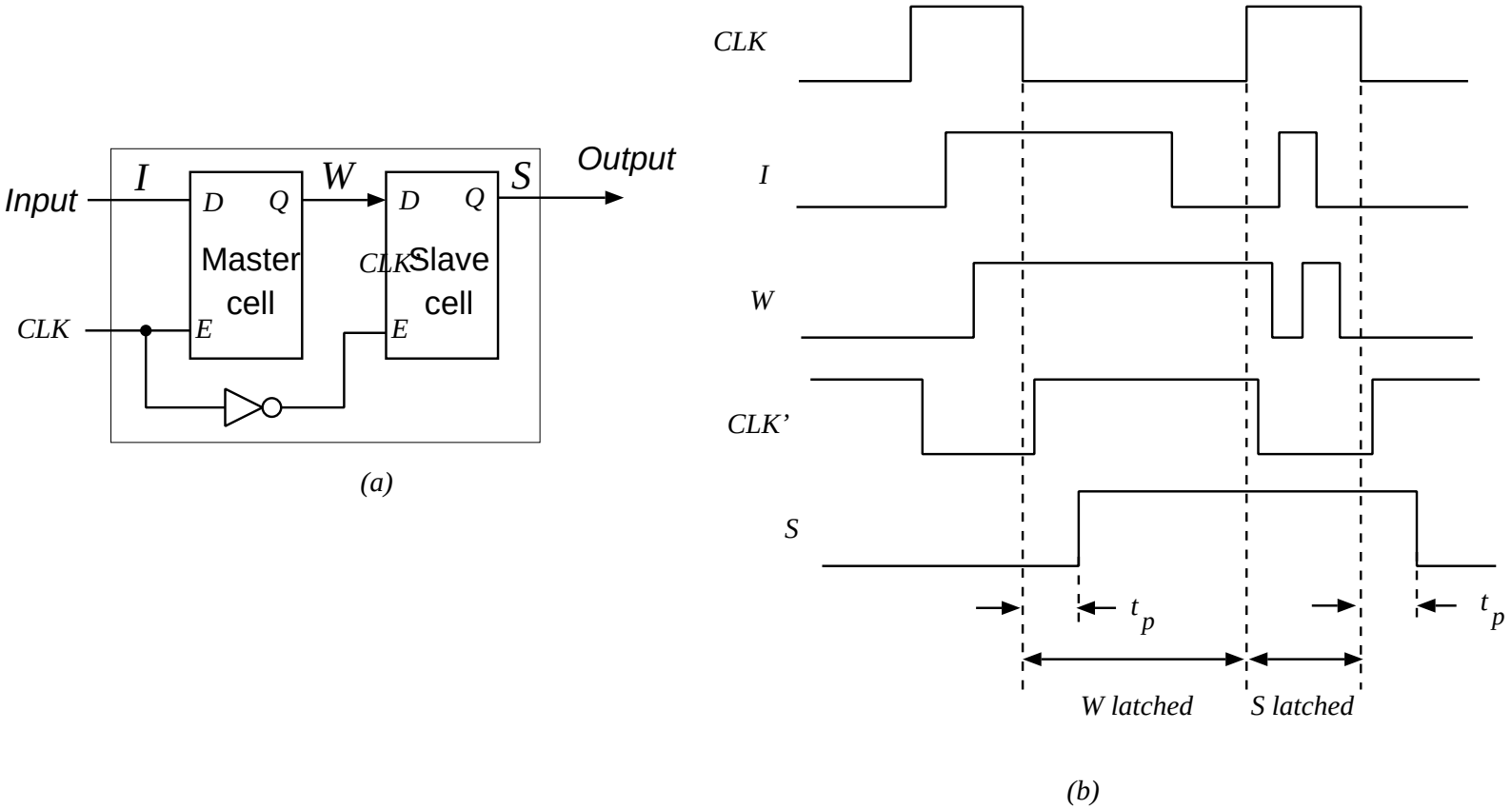



Figure 8.10: a) MASTER-SLAVE IMPLEMENTATION OF TRAILING-EDGE-TRIGGERED CELL. b) MASTER-SLAVE STATE CHANGE PROCESS.

# PRACTICAL BASIC CELL: D flip-flop

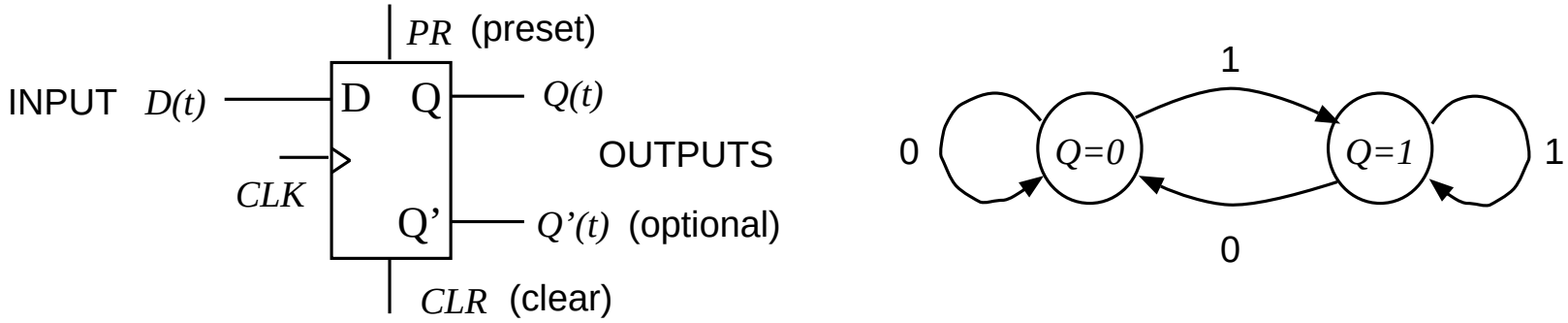



Figure 8.11: D FLIP-FLOP AND ITS STATE DIAGRAM.

| $PS = Q(t)$ | $D(t)$ | |
|---|---|---|
| | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| | $NS = Q(t+1)$ | |

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

# TIMING PARAMETERS OF A BINARY CELL

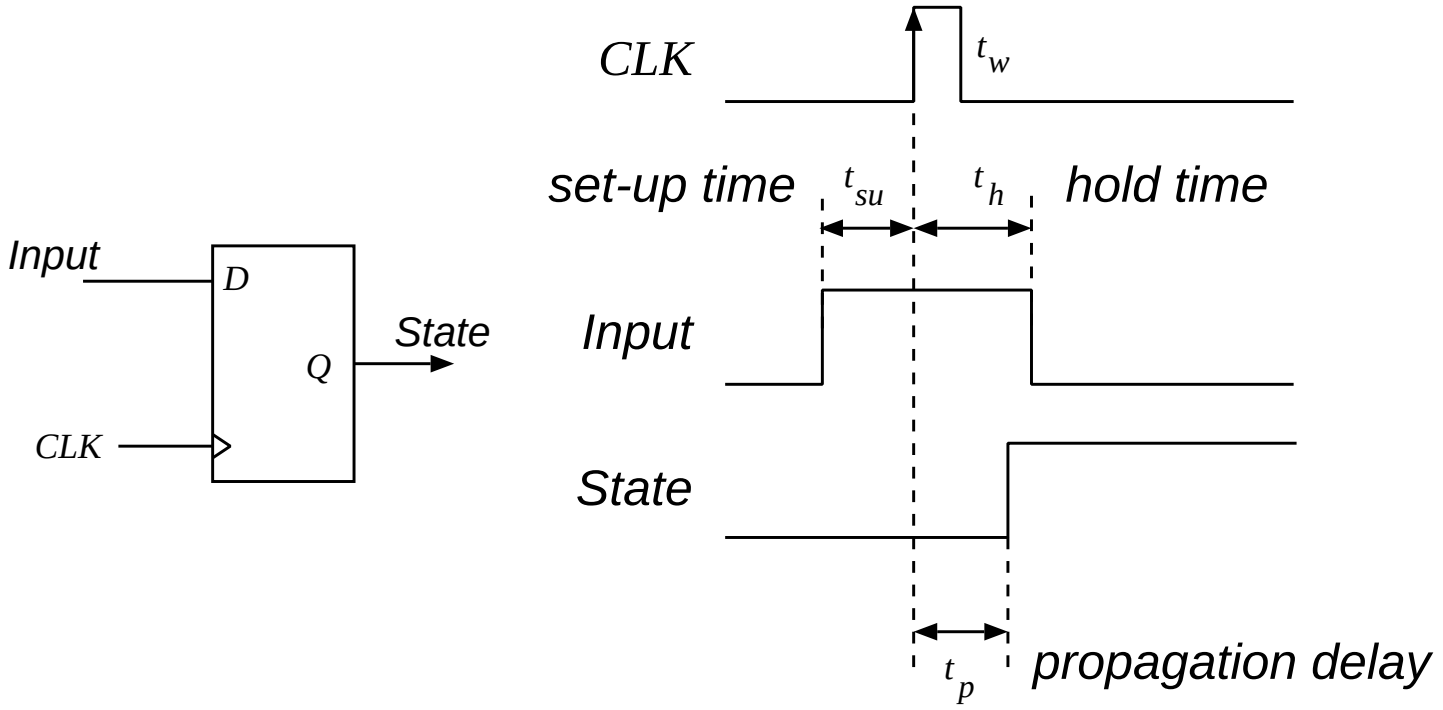



Figure 8.12: TIME BEHAVIOR OF CELL.

# CHARACTERISTICS OF A CMOS D flip-flop

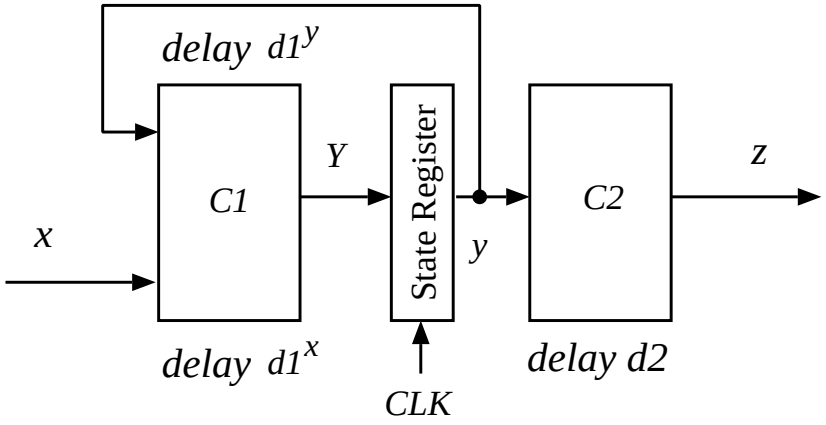
| Delays | | | | | Input factor | Size |
|---|---|---|---|---|---|---|
| $t_{pLH}$ [ns] | $t_{pHL}$ [ns] | $t_{su}$ [ns] | $t_h$ [ns] | $t_w$ [ns] | [std. loads] | [equiv. gates] |
| $0.49 + 0.038L$ | $0.54 + 0.019L$ | 0.30 | 0.14 | 0.2 | 1 | 6 |

$L$: output load of the flip-flop

- THIS FLIP-FLOP HAS ONLY THE UNCOMPLEMENTED OUTPUT

# TIMING CHARACTERISTICS OF SEQUENTIAL NETWORKS

- NETWORK SET-UP TIME: $t_{su}^{x}(net) = d1^{x} + t_{su}(cell)$



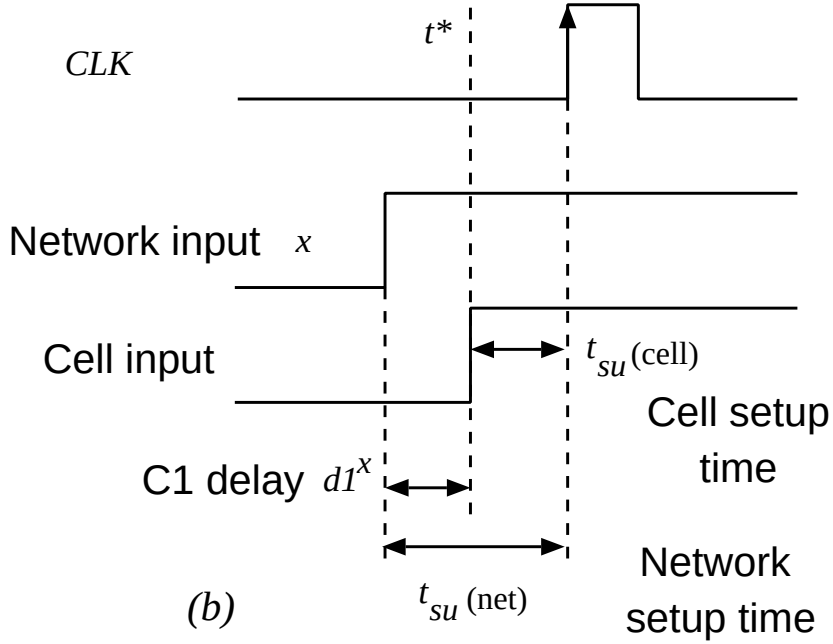




Figure 8.13: TIMING FACTORS IN SEQUENTIAL NETWORKS: a) THE NETWORK. b) NETWORK SET-UP TIME.

# TIMING FACTORS (cont.)

- NETWORK HOLD TIME: $t_h(net) = t_h(cell)$

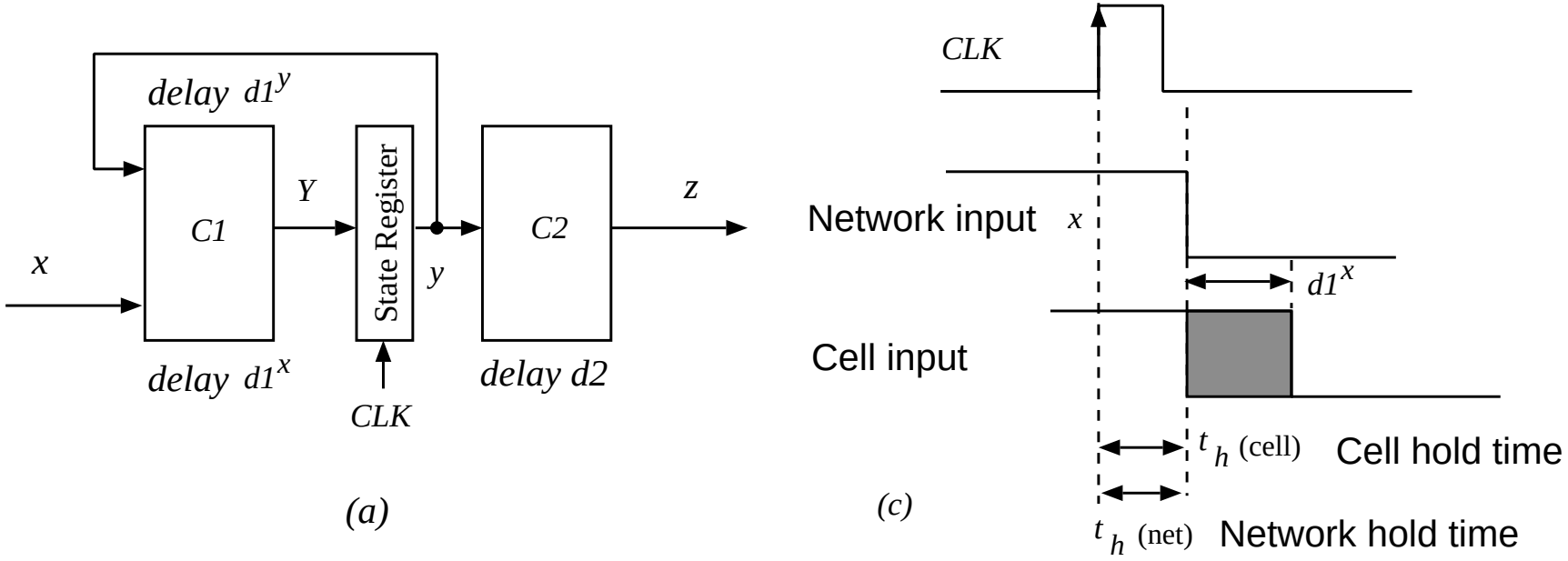



Figure 8.14: TIMING FACTORS IN SEQUENTIAL NETWORKS: a) THE NETWORK. c) NETWORK HOLD TIME.

# TIMING FACTORS (Cont.)

- NETWORK PROPAGATION DELAY: $t_p(net) = t_p(cell) + d2$

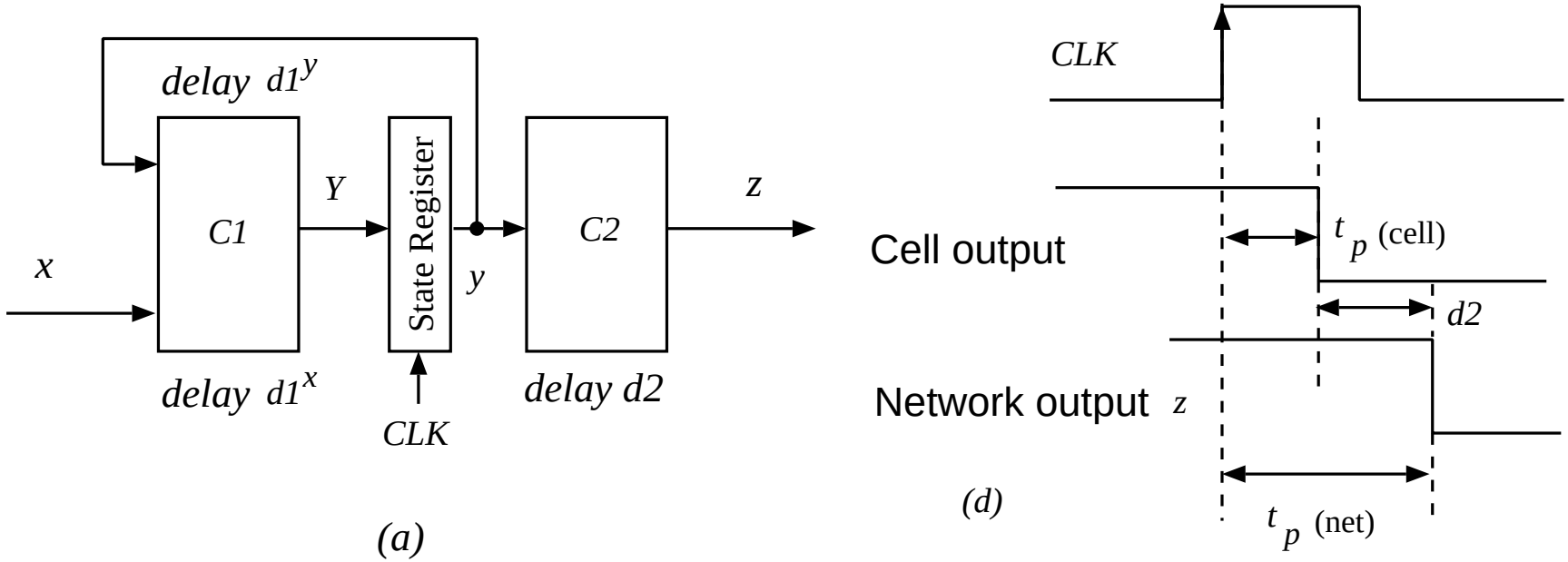



Figure 8.14: TIMING FACTORS IN SEQUENTIAL NETWORKS: a) THE NETWORK. d) NETWORK PROPAGATION DELAY.

# MAXIMUM CLOCK FREQUENCY

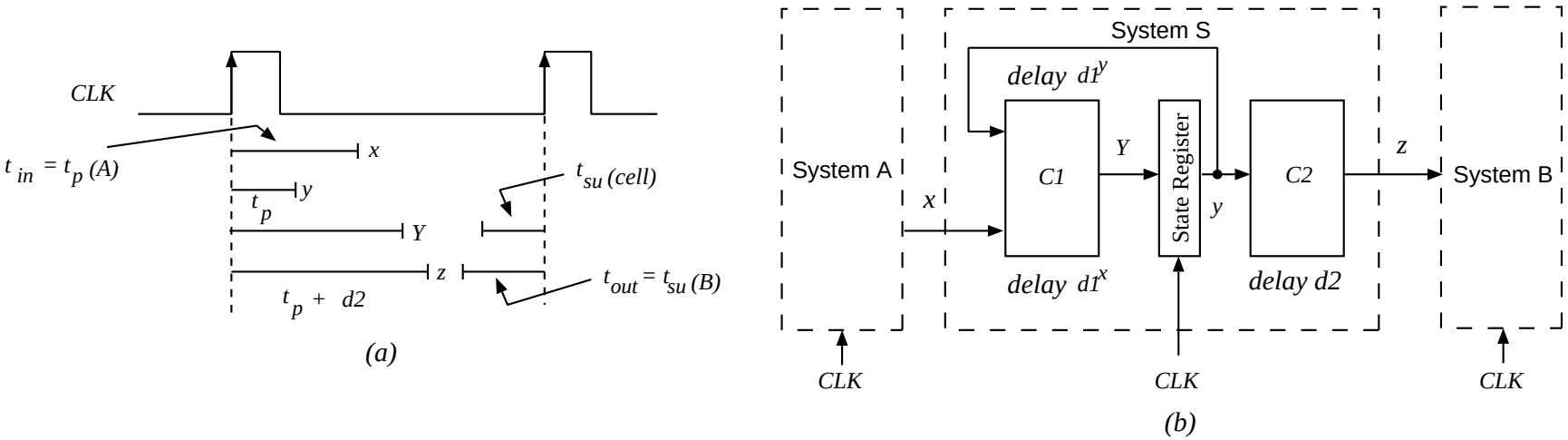
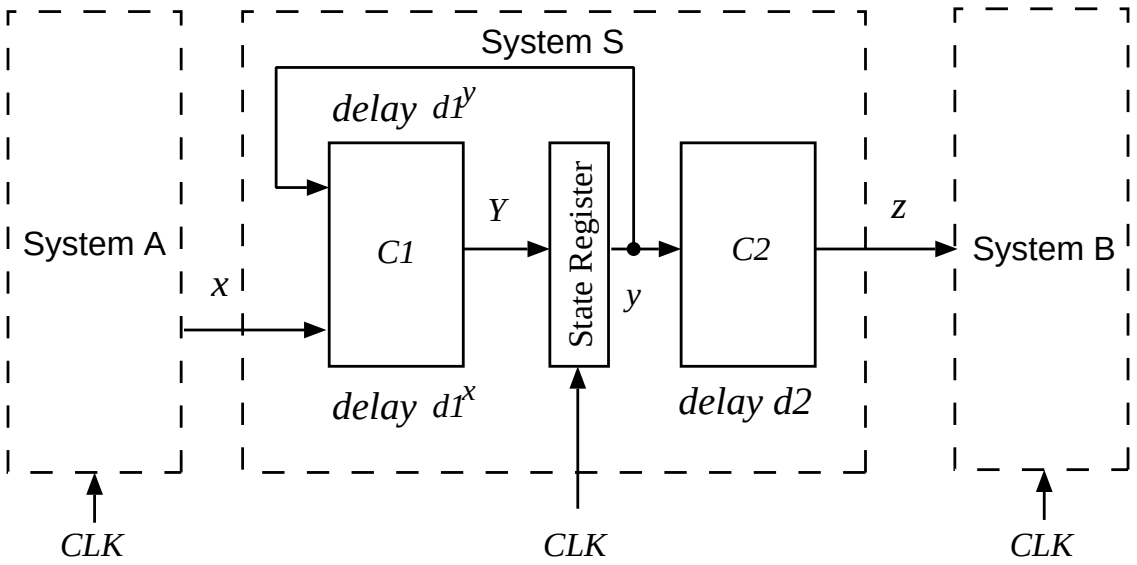
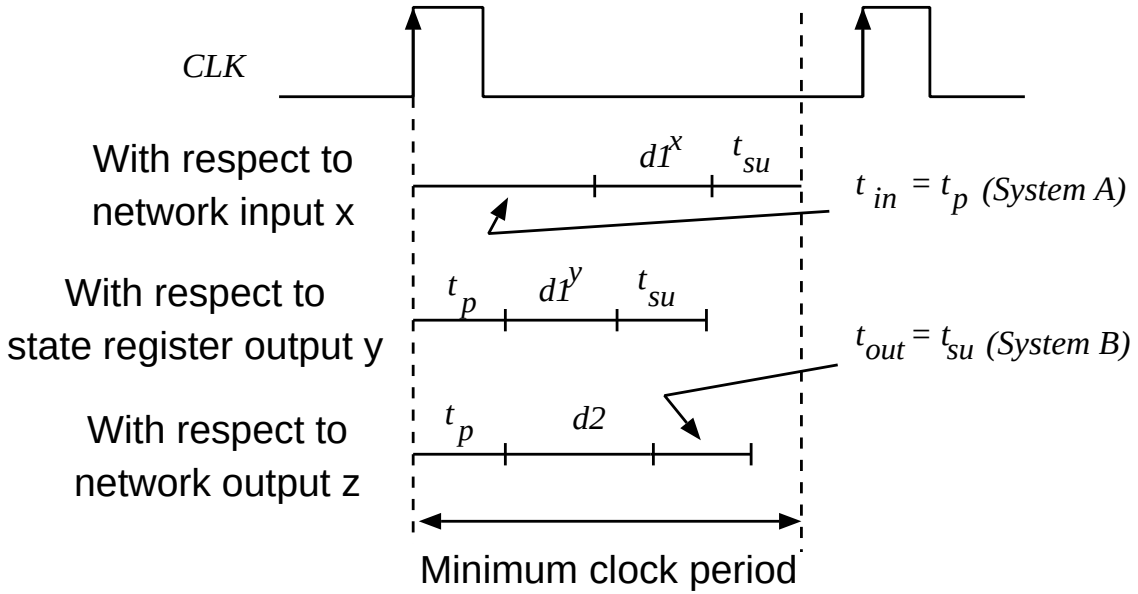



Figure 8.15: MAXIMUM CLOCK FREQUENCY: a) CLOCK PERIOD AND SIGNAL DELAYS. b) THE NETWORK.

- $t_{in}$ - TIME BETWEEN TRIGGERING EDGE OF CLOCK AND STABILIZATION OF INPUT $x$
- $t_{out}$ - TIME BETWEEN STABILIZATION OF OUTPUT $z$ AND NEXT CLOCK TRIGGERING EDGE




(b)




(c)

Figure 8.15: MAXIMUM CLOCK FREQUENCY: b) THE NETWORK. c) MINIMUM CLOCK PERIOD.

# MAXIMUM CLOCK FREQUENCY (cont.)

$$T_{\min} = 1/f_{\max}$$

$$T_{\min} = \max[(t_{in}+t^{x}_{su}(net)),\ (t_p(cell)+t^{y}_{su}(net)),\ (t_p(net)+t_{out})]$$

$$t_h(cell) \leq t_p(cell)$$

$$T_{\min} = \max[(t_{in}+d1^{x}+t_{su}(cell)),\ (t_p(cell)+d1^{y}+t_{su}(cell)),\ (t_p(cell)+d$$

# EXAMPLE 8.3

DETERMINE THE MAXIMUM CLOCK FREQUENCY

$$d1^x = d1^y = 2.5ns$$
$$d2 = 3ns$$
$$t_{su} = 0.3ns$$
$$t_p = 1ns$$
$$t_{in} = 2ns$$
$$t_{out} = 3ns$$

THE MINIMUM CLOCK PERIOD

$$T_{\min} = \max[(2 + 2.5 + 0.3),\ (1 + 2.5 + 0.3),\ (1 + 3 + 3)] = 7[\text{ns}]$$

THE MAXIMUM FREQUENCY

$$f_{\max} = \frac{1}{7 \times 10^{-9}} \approx 140(\text{MHz})$$

# CLOCK SKEW

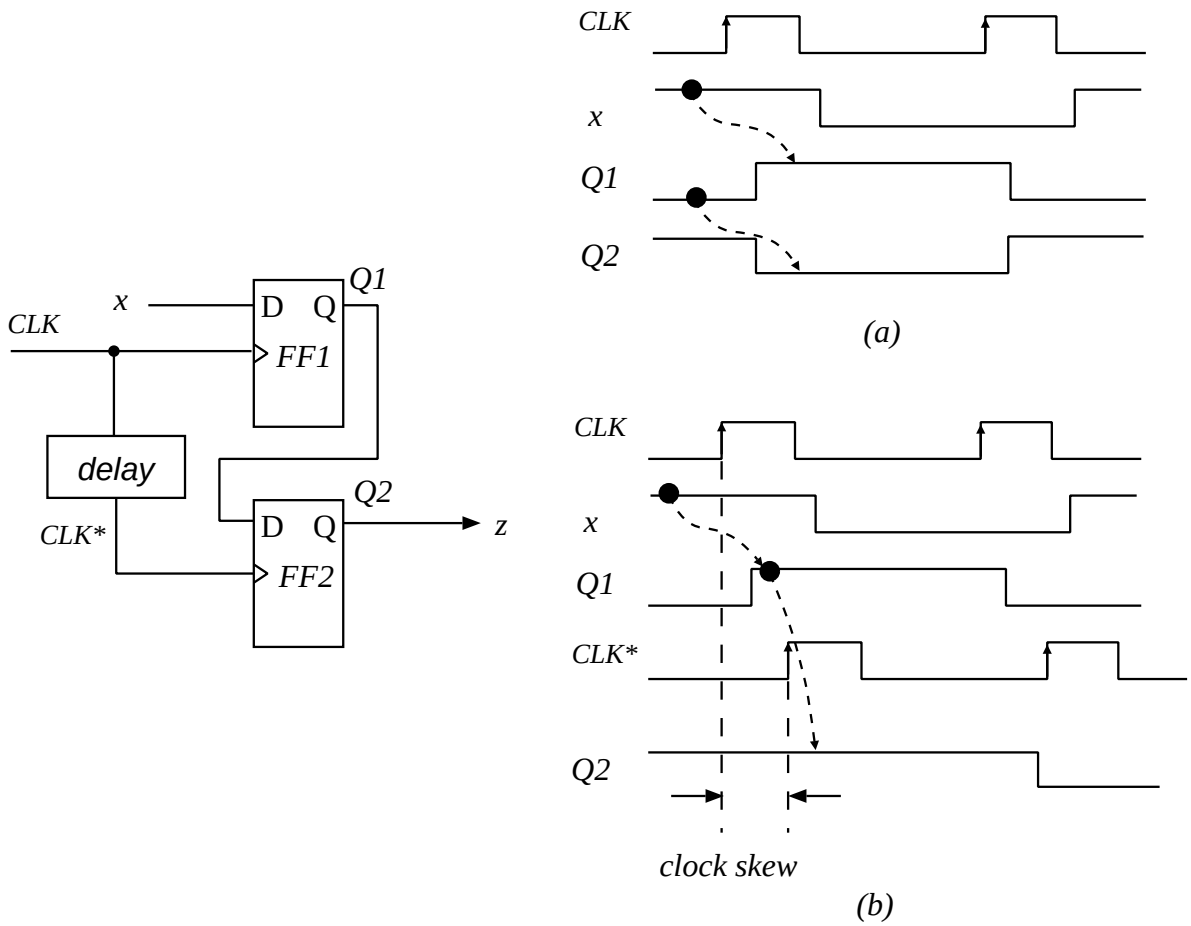



Figure 8.16: a) NETWORK BEHAVIOR WITHOUT CLOCK SKEW. b) NETWORK BEHAVIOR WITH INADMISSIBLE CLOCK SKEW.

# ANALYSIS OF CANONICAL SEQUENTIAL NETWORKS

1. ANALYZE COMBINATIONAL NETWORK
   DETERMINE THE TRANSITION AND OUTPUT FUNCTIONS

2. DETERMINE HIGH-LEVEL SPECIFICATION OF STATE DESCRIPTION OUTPUT FUNCTIONS.

3. IF DESIRED (OR REQUIRED), DETERMINE TIME BEHAVIOR

# EXAMPLE 8.4: ANALYSIS

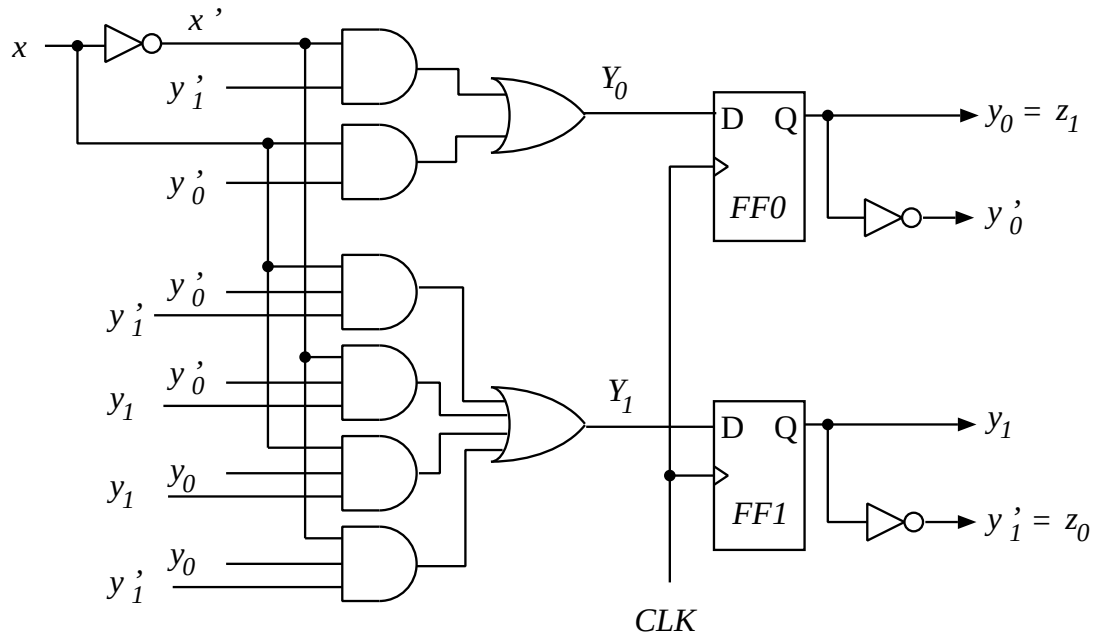



Figure 8.17: SEQUENTIAL NETWORK IN Example 8.4.

State transition

$$Y_0 = x'y_1' + xy_0'$$
$$Y_1 = xy_0'y_1' + x'y_0'y_1 + xy_0y_1 + x'y_0y_1'$$

Output

$$z_0 = y_1'$$
$$z_1 = y_0$$

# EXAMPLE 8.4 (cont.)

- STATE-TRANSITION AND OUTPUT FUNCTIONS:

| $PS$ | Input | | |
|---|---|---|---|
| $y_1y_0$ | $x=0$ | $x=1$ | |
| 00 | 01 | 11 | 01 |
| 01 | 11 | 00 | 11 |
| 10 | 10 | 01 | 00 |
| 11 | 00 | 10 | 10 |
| | $Y_1Y_0$ | | $z_1z_0$ |
| | $NS$ | | Output |

- CODES:

| $x$ | $x$ |
|---|---|
| 0 | $a$ |
| 1 | $b$ |

| $z_1z_0$ | $z$ |
|---|---|
| 00 | $c$ |
| 01 | $d$ |
| 10 | $e$ |
| 11 | $f$ |

| $y_1y_0$ | $s$ |
|---|---|
| 00 | $S_0$ |
| 01 | $S_1$ |
| 10 | $S_2$ |
| 11 | $S_3$ |

# EXAMPLE 8.4 (cont.)

- HIGH-LEVEL SPECIFICATION:

  Input: $x(t) \in \{a, b\}$
  Output: $z(t) \in \{c, d, e, f\}$
  State: $s(t) \in \{S_0, S_1, S_2, S_3\}$
  Initial state: $s(0) = S_2$

  Functions: The state-transition and output functions

| $PS$ | $x(t) = a$ | $x(t) = b$ | |
|---|---|---|---|
| $S_0$ | $S_1$ | $S_3$ | $d$ |
| $S_1$ | $S_3$ | $S_0$ | $f$ |
| $S_2$ | $S_2$ | $S_1$ | $c$ |
| $S_3$ | $S_0$ | $S_2$ | $e$ |
| | $NS$ | | $z(t)$ |

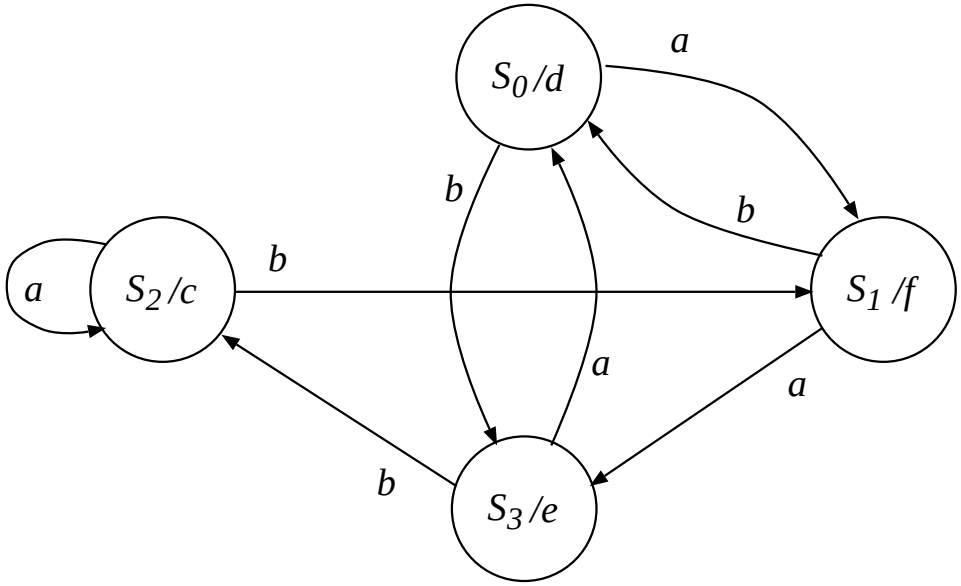



Figure 8.18: a) STATE DIAGRAM FOR SEQUENTIAL NETWORK.

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $x(t)$ | $a$ | $a$ | $b$ | $a$ | $b$ | $b$ | $a$ | $b$ | $a$ | $a$ | $b$ | $b$ | $b$ | $a$ | |
| $s(t)$ | $S_2$ | $S_2$ | $S_2$ | $S_1$ | $S_3$ | $S_2$ | $S_1$ | $S_3$ | $S_2$ | $S_2$ | $S_2$ | $S_1$ | $S_0$ | $S_3$ | $S_0$ |
| $z(t)$ | $c$ | $c$ | $c$ | $f$ | $e$ | $c$ | $f$ | $e$ | $c$ | $c$ | $c$ | $f$ | $d$ | $e$ | $d$ |

Figure 8.18: b) A sequence of input-output pairs.

# PROPAGATION DELAY $x$ to $z_0$:

| | | | |
|---|---|---|---|
| INPUT LOAD FACTORS: | $I_x$ | $=$ | $4$ |
| SET-UP TIME: | $t_{su}(net)$ | $=$ | $t_{pHL}(\text{NOT}) + t_{pHL}(\text{AND3})$ |
| | | | $+t_{pHL}(\text{OR4}) + t_{su}$ |
| | | $=$ | $(0.05 + 0.017 \times 3) + (0.18 + 0.018)$ |
| | | | $+(0.45 + 0.025) + 0.3$ |
| | | $=$ | $1.07$ [ns] |
| HOLD TIME: | $t_h(net)$ | $=$ | $0.14$ [ns] |
| PROPAGATION DELAY: | $t_p(z_0)$ | $=$ | $t_{pLH}(\text{FF}) + t_{pHL}(\text{NOT})$ |
| | | $=$ | $(0.49 + 0.038 \times 3)$ |
| | | | $+(0.05 + 0.017 \times (L + 3))$ |
| | | $=$ | $0.70 + 0.017L$ [ns] |
| | | | (load of NOT is $L + 3$, load of FF is 3) |
| SIZE: | | $=$ | $6 \times 2 + 2 + 3 + 2 \times 6 + 3 \times 1$ |
| | | $=$ | 32 equivalent gates. |

# DESIGN OF CANONICAL SEQUENTIAL NETWORKS

1. TRANSFORM THE TRANSITION AND OUTPUT FUNCTIONS

2. SPECIFY A STATE REGISTER TO ENCODE THE REQUIRED NUMBER OF STATES

3. DESIGN THE REQUIRED COMBINATIONAL NETWORK

# EXAMPLE 8.5: DESIGN

Input: $x(t) \in \{a, b, c\}$
Output: $z(t) \in \{0, 1\}$
State: $s(t) \in \{A, B, C, D\}$
Initial state: $s(0) = A$

Functions: The state-transition and output functions

| $PS$ | Input | | |
|---|---|---|---|
| | $x = a$ | $x = b$ | $x = c$ |
| $A$ | $C$,0 | $B$,1 | $B$,0 |
| $B$ | $D$,0 | $B$,0 | $A$,1 |
| $C$ | $A$,0 | $D$,1 | $D$,0 |
| $D$ | $B$,0 | $A$,0 | $D$,1 |
| | $NS,\ z$ | | |

# EXAMPLE 8.5 (cont.)

- CODING:

Input code

| $x$ | $x_1$ | $x_0$ |
|---|---|---|
| $a$ | 0 | 1 |
| $b$ | 1 | 0 |
| $c$ | 1 | 1 |

State code

| $s$ | $y_1$ | $y_0$ |
|---|---|---|
| $A$ | 0 | 0 |
| $B$ | 1 | 0 |
| $C$ | 0 | 1 |
| $D$ | 1 | 1 |

# EXAMPLE 8.5 (cont.)

- STATE-TRANSITION AND OUTPUT FUNCTIONS

| $PS$ | $x_1x_0$ | | |
|---|---|---|---|
| $y_1y_0$ | 01 | 10 | 11 |
| 00 | 01,0 | 10,1 | 10,0 |
| 10 | 11,0 | 10,0 | 00,1 |
| 01 | 00,0 | 11,1 | 11,0 |
| 11 | 10,0 | 00,0 | 11,1 |
| | $Y_1Y_0,\ z$ | | |
| | $NS$, Output | | |

$Y_1$:

| $y_1 \backslash x_0x_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 0 | 1 | 1 |
| 1 | - | 0 | 1 | 1 |
| 1 | - | 1 | 1 | 0 |
| 0 | - | 1 | 0 | 1 |

$y_0$

$Y_0$:

| $y_1 \backslash x_0x_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 1 | 0 | 0 |
| 1 | - | 0 | 1 | 1 |
| 1 | - | 0 | 1 | 0 |
| 0 | - | 1 | 0 | 0 |

$y_0$

$z$:

| $y_1 \backslash x_0x_1$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 0 | 0 | 1 |
| 1 | - | 0 | 0 | 1 |
| 1 | - | 0 | 1 | 0 |
| 0 | - | 0 | 1 | 0 |

$y_0$

- NEXT-STATE AND OUTPUT EXPRESSIONS

$$
\begin{aligned}
Y_1 &= y_1'x_1 + y_1x_1' + y_0'x_0' + y_0x_1x_0 \\
Y_0 &= y_0'x_1' + y_1'y_0x_1 + y_0x_1x_0 \\
z &= y_1'x_0' + y_1x_1x_0
\end{aligned}
$$

# EXAMPLE 8.5 (cont.)

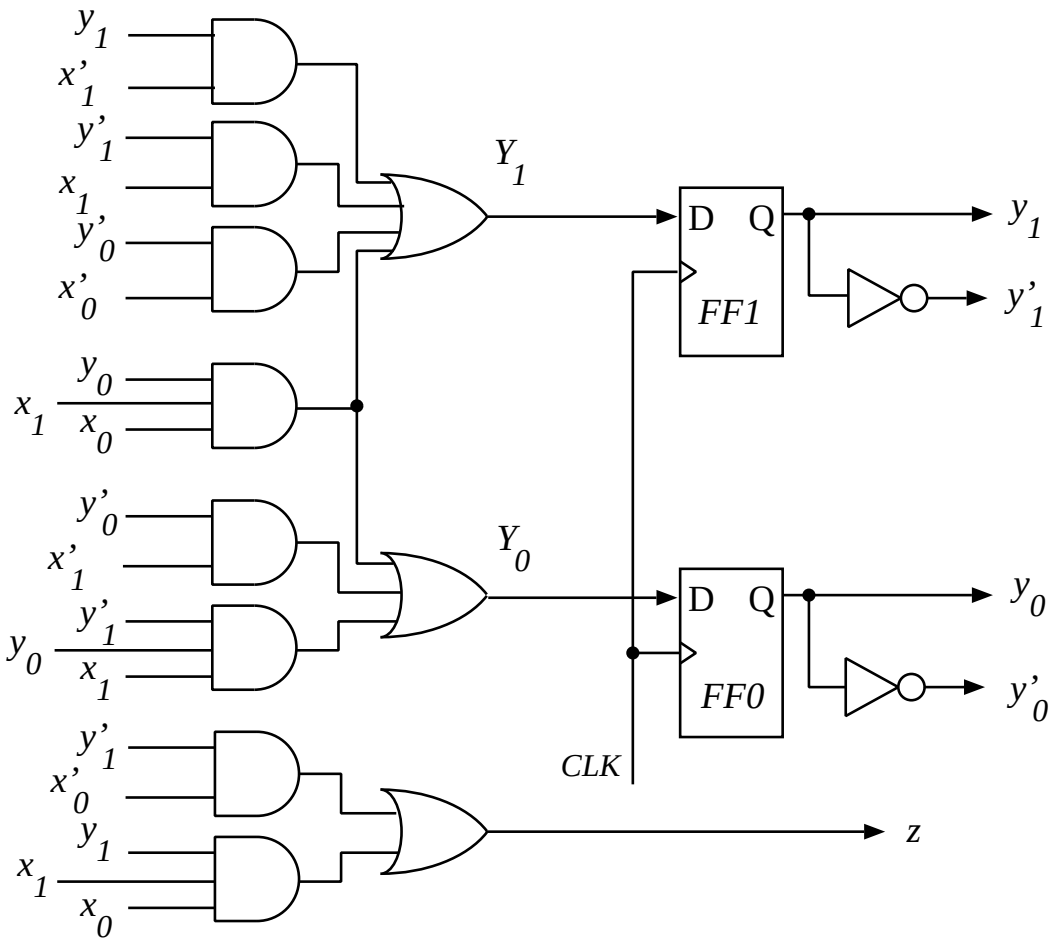
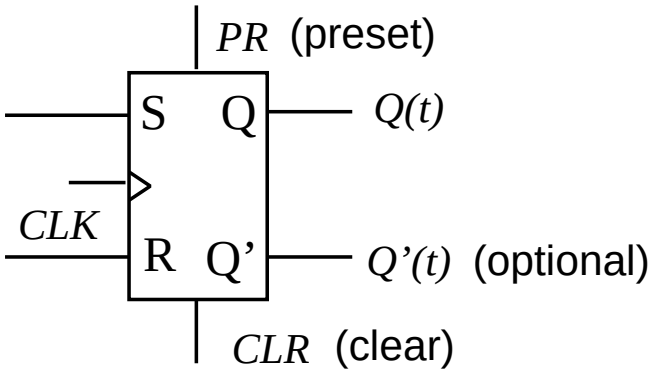



Figure 8.19: SEQUENTIAL NETWORK IN Example 8.5.

# SR FLIP-FLOP



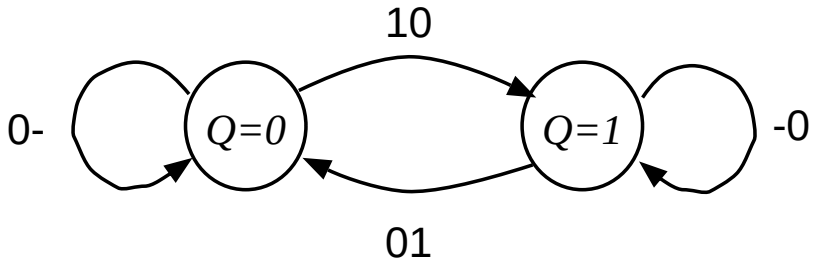
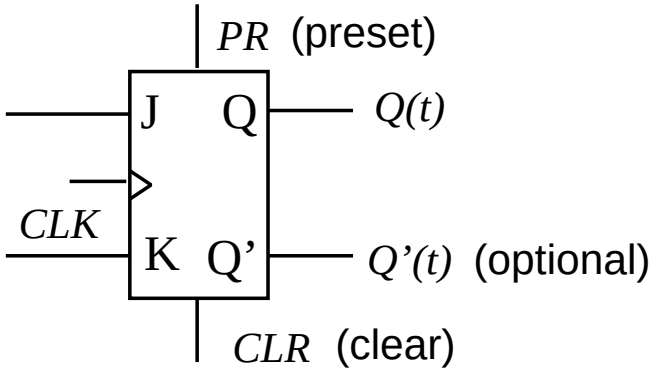




Figure 8.20: SR FLIP-FLOP AND ITS STATE DIAGRAM.

| $PS = Q(t)$ | $S(t)R(t)$ | | | |
|---|---|---|---|---|
| | 00 | 01 | 10 | 11 |
| 0 | 0 | 0 | 1 | - |
| 1 | 1 | 0 | 1 | - |
| | $NS = Q(t+1)$ | | | |

$$Q(t+1) = Q(t)R'(t) + S(t) \text{ restriction: } R(t) \cdot S(t) = 0$$

# JK FLIP-FLOP



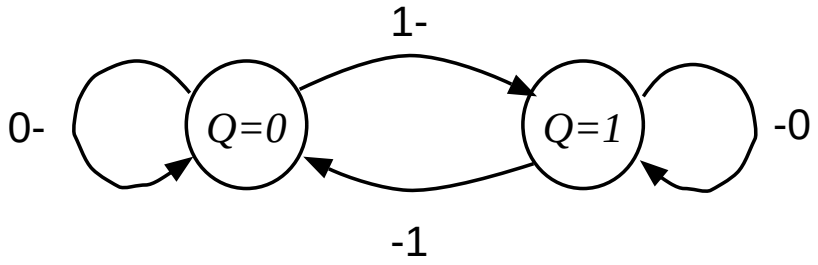
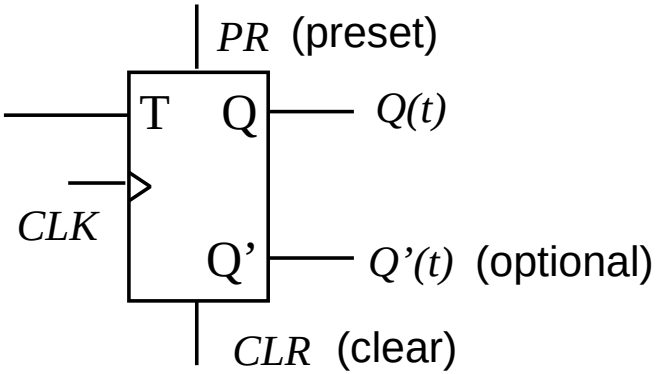




Figure 8.21: JK FLIP-FLOP AND ITS STATE DIAGRAM.

| $PS = Q(t)$ | $J(t)K(t)$ | | | |
|---|---|---|---|---|
| | 00 | 01 | 10 | 11 |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| | $NS = Q(t+1)$ | | | |

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

# T (Toggle) FLIP-FLOP



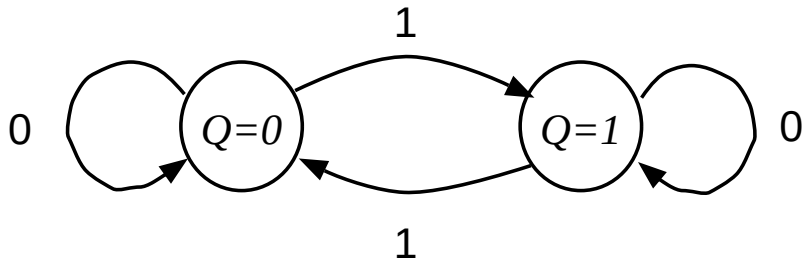




Figure 8.22: T FLIP-FLOP AND ITS STATE DIAGRAM.

| $PS = Q(t)$ | $T(t)$ | |
|---|---|---|
| | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 0 |
| | $NS = Q(t+1)$ | |

$$Q(t+1) = Q(t) \oplus T(t)$$

# IMPLEMENTING ONE FF TYPE WITH ANOTHER

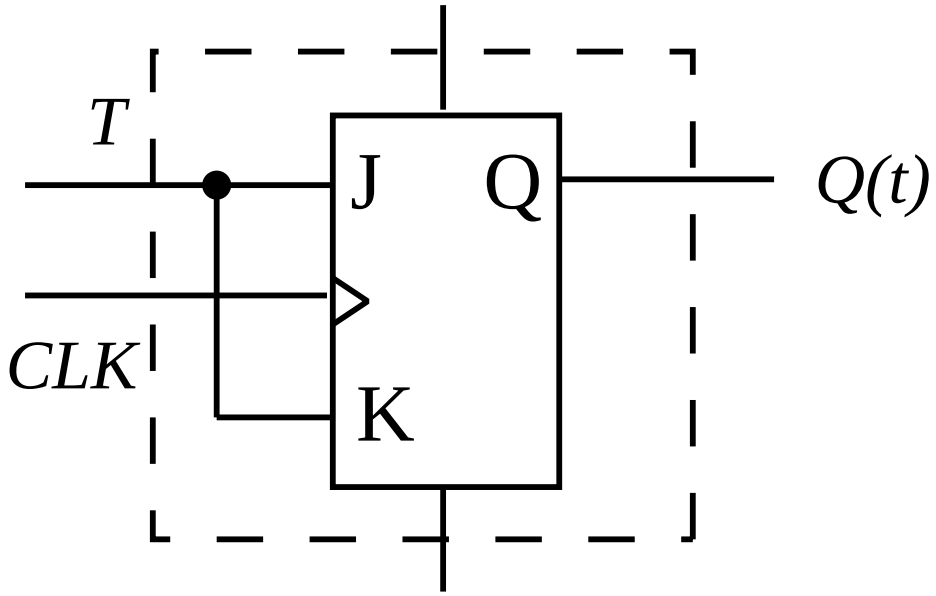



Figure 8.23: T FLIP-FLOP IMPLEMENTED WITH JK FLIP-FLOP.

# ANALYSIS OF NETWORKS WITH FLIP-FLOPS

1. OBTAIN THE TRANSITION FUNCTION OF THE NETWORK
   (a) DETERMINE THE INPUTS TO THE FLIP-FLOPS
   (b) USE THE TRANSITION FUNCTION OF THE FLIP-FLOPS TO DETERMINE THE NEXT STATE

2. OBTAIN THE OUTPUT FUNCTION

3. DETERMINE A SUITABLE HIGH-LEVEL SPECIFICATION

# CHARACTERISTICS OF A FAMILY OF CMOS FLIP-FLOPS

| FF type | Delays | | | | | Input factor | Size |
|---|---|---|---|---|---|---|---|
| | $t_{pLH}$ [ns] | $t_{pHL}$ [ns] | $t_{su}$ [ns] | $t_h$ [ns] | $t_w$ [ns] | [std. loads] | [equiv. gates] |
| D | $0.49 + 0.038L$ | $0.54 + 0.019L$ | 0.30 | 0.14 | 0.20 | 1 | 6 |
| JK | $0.45 + 0.038L$ | $0.47 + 0.022L$ | 0.41 | 0.23 | 0.20 | 1 | 8 |

$L$: output load of the flip-flop

These flip-flops have only uncomplemented outputs

# EXAMPLE 8.6: ANALYSIS

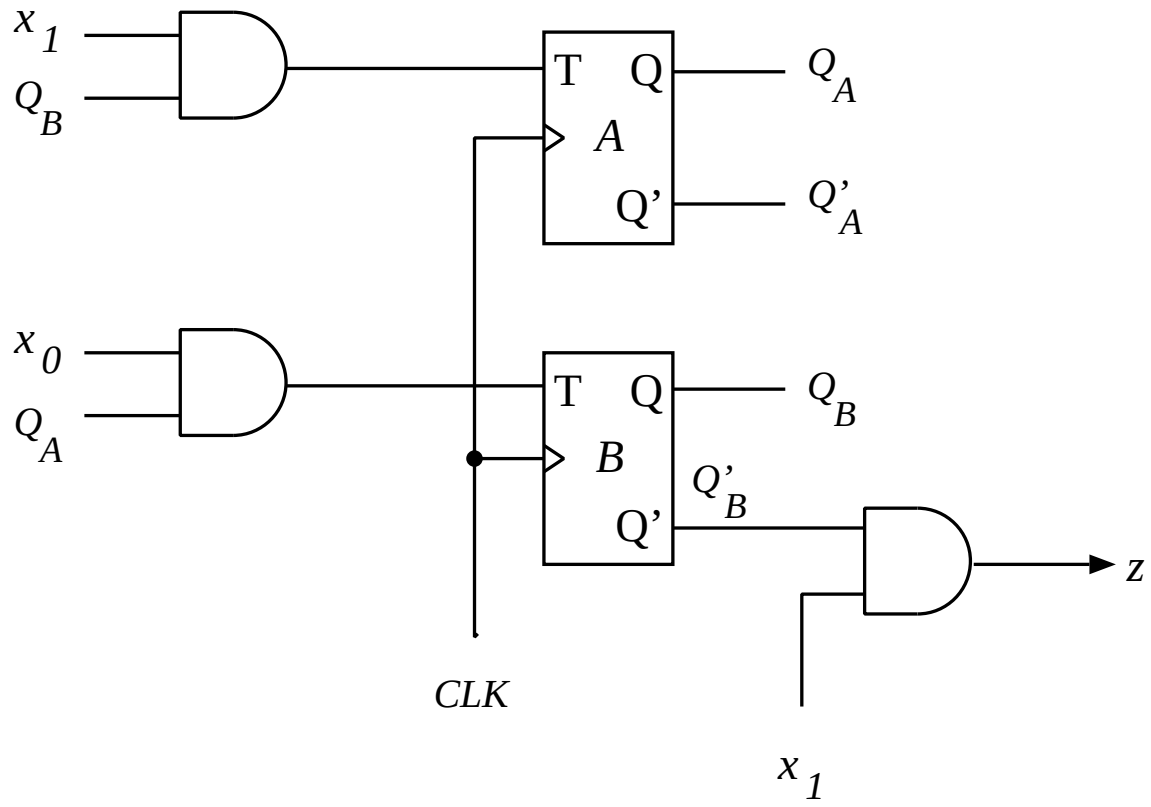



Figure 8.24: SEQUENTIAL NETWORK FOR Example 8.6.

$$
\begin{aligned}
T_A &= x_1 Q_B \quad Q_A(t+1) = Q_A(t) \oplus x_1 Q_B(t) \\
T_B &= x_0 Q_A \quad Q_B(t+1) = Q_B(t) \oplus x_0 Q_A(t) \\
z(t) &= x_1(t) Q'_B(t)
\end{aligned}
$$

# EXAMPLE 8.6 (cont.)

- STATE-TRANSITION AND OUTPUT FUNCTIONS

| $PS$ | Input | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $Q_AQ_B$ | $x_1x_0$ | | | | $x_1x_0$ | | | |
| | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 |
| 00 | 00 | 00 | 00 | 00 | 0 | 0 | 1 | 1 |
| 01 | 01 | 01 | 11 | 11 | 0 | 0 | 0 | 0 |
| 10 | 10 | 11 | 10 | 11 | 0 | 0 | 1 | 1 |
| 11 | 11 | 10 | 01 | 00 | 0 | 0 | 0 | 0 |
| | $Q_AQ_B$ | | | | $z$ | | | |
| | $NS$ | | | | Output | | | |

# EXAMPLE 8.6 (cont.)

- CODING:

| $Q_A$ | $Q_B$ | $s$ |
|---|---|---|
| 0 | 0 | $S_0$ |
| 0 | 1 | $S_1$ |
| 1 | 0 | $S_2$ |
| 1 | 1 | $S_3$ |

| $x_1$ | $x_0$ | $x$ |
|---|---|---|
| 0 | 0 | $a$ |
| 0 | 1 | $b$ |
| 1 | 0 | $c$ |
| 1 | 1 | $d$ |

# EXAMPLE 8.6 (cont.)

- HIGH-LEVEL DESCRIPTION:

  INPUT: $x(t) \in \{a, b, c, d\}$

  OUTPUT: $z(t) \in \{0, 1\}$

  STATE: $s(t) \in \{S_0, S_1, S_2, S_3\}$

  INITIAL STATE: $s(0) = S_0$

  Functions: the state-transition and output functions

| $PS$ | $x$ | | | | $x$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $a$ | $b$ | $c$ | $d$ | $a$ | $b$ | $c$ | $d$ |
| $S_0$ | $S_0$ | $S_0$ | $S_0$ | $S_0$ | 0 | 0 | 1 | 1 |
| $S_1$ | $S_1$ | $S_1$ | $S_3$ | $S_3$ | 0 | 0 | 0 | 0 |
| $S_2$ | $S_2$ | $S_3$ | $S_2$ | $S_3$ | 0 | 0 | 1 | 1 |
| $S_3$ | $S_3$ | $S_2$ | $S_1$ | $S_0$ | 0 | 0 | 0 | 0 |
| | $NS$ | | | | $z$ | | | |

# EXAMPLE 8.7: ANALYSIS

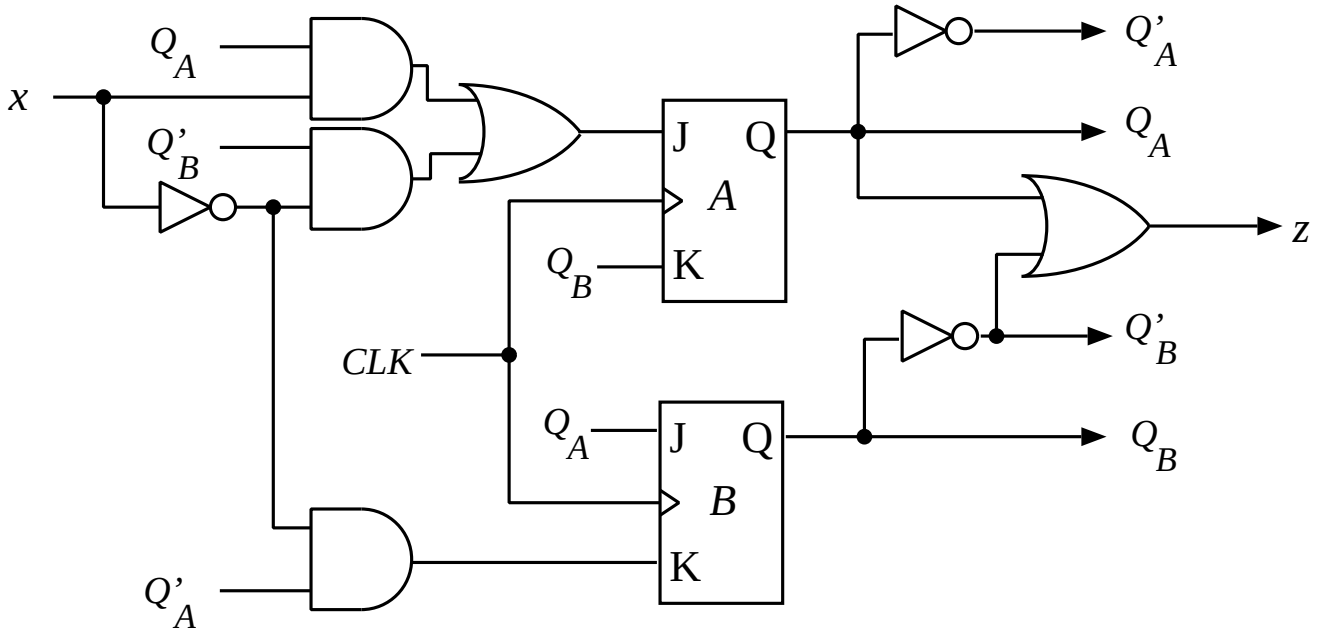



Figure 8.25: SEQUENTIAL NETWORK FOR Example 8.7

$$J_A = x'Q'_B + xQ_A \qquad K_A = Q_B$$
$$J_B = Q_A \qquad K_B = x'Q'_A$$

$$z = Q_A + Q'_B$$

# EXAMPLE 8.7 (cont.)

$$J_A = x'Q'_B + xQ_A \qquad K_A = Q_B$$
$$J_B = Q_A \qquad K_B = x'Q'_A$$

$$z = Q_A + Q'_B$$

$$\begin{aligned} Q_A(t+1) &= Q_A K'_A + Q'_A J_A \\ &= Q_A Q'_B + Q'_A(x'Q'_B + xQ_A) \\ &= Q'_B(Q_A + x') \end{aligned}$$

$$\begin{aligned} Q_B(t+1) &= Q_B K'_B + Q'_B J_B \\ &= Q_B(x + Q_A) + Q'_B Q_A \\ &= Q_B x + Q_A \end{aligned}$$

# EXAMPLE 8.7 (cont.)

- STATE-TRANSITION AND OUTPUT FUNCTIONS

| $PS$ | $NS$ | | Output |
|---|---|---|---|
| | $x=0$ | $x=1$ | $z$ |
| $Q_AQ_B$ | $Q_AQ_B$ | $Q_AQ_B$ | |
| 00 | 10 | 00 | 1 |
| 01 | 00 | 01 | 0 |
| 10 | 11 | 11 | 1 |
| 11 | 01 | 01 | 1 |

- STATE CODING

| $Q_A$ | $Q_B$ | $S$ |
|---|---|---|
| 0 | 0 | $S_0$ |
| 0 | 1 | $S_1$ |
| 1 | 0 | $S_2$ |
| 1 | 1 | $S_3$ |

# EXAMPLE 8.7 (cont.)

- HIGH-LEVEL DESCRIPTION

  INPUT: $x(t) \in \{0, 1\}$

  OUTPUT: $z(t) \in \{0, 1\}$

  STATE: $s(t) \in \{S_0, S_1, S_2, S_3\}$

  INITIAL STATE: $s(0) = S_0$

  Functions: The state-transition and output functions

| $PS$ | Input | | |
|---|---|---|---|
| | $x = 0$ | $x = 1$ | |
| $S_0$ | $S_2$ | $S_0$ | 1 |
| $S_1$ | $S_0$ | $S_1$ | 0 |
| $S_2$ | $S_3$ | $S_3$ | 1 |
| $S_3$ | $S_1$ | $S_1$ | 1 |
| | $NS$ | | $z$ |

# EXAMPLE 8.7 (cont.)

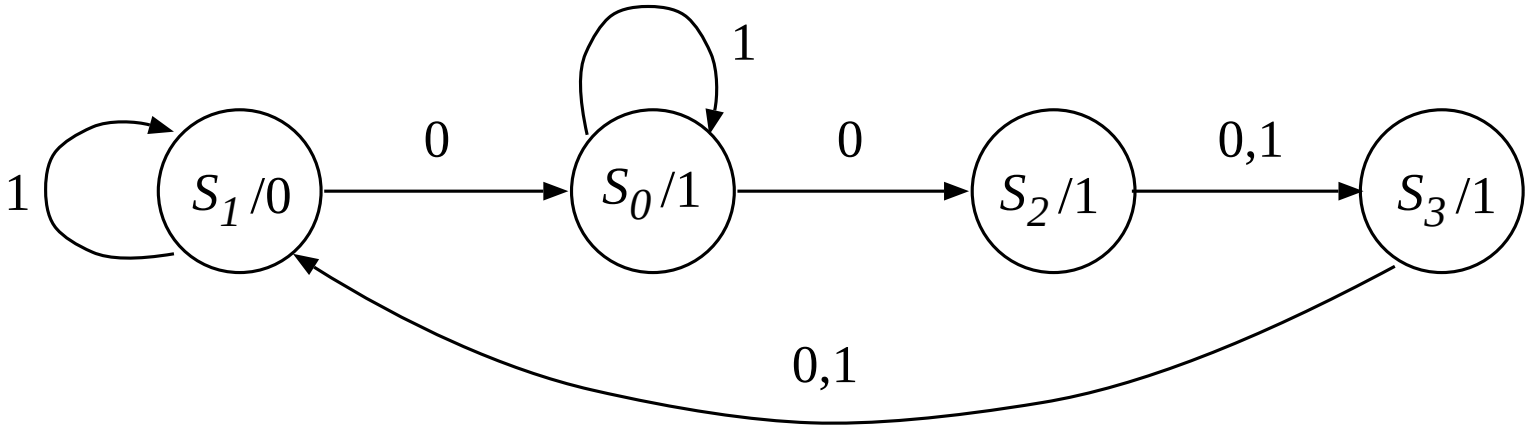



Figure 8.26: STATE DIAGRAM IN Example 8.7.

# OTHER CHARACTERISTICS (Example 8.7)

INPUT LOAD FACTOR: $I_x = 2$

SET-UP TIME: $t_{su}(net)$

$$\begin{aligned} &= t_{pLH}(\text{NOT}) + t_{pLH}(\text{AND}) + t_{pLH}(\text{OR}) \\ &\quad + t_{su}(FF) \\ &= (0.02 + 0.038 \times 2) + (0.15 + 0.037) \\ &\quad + (0.12 + 0.037) + 0.41 \\ &= 0.86 \text{ [ns]} \end{aligned}$$

HOLD TIME: $t_h(net) = 0.23$ [ns]

PROPAGATION DELAY: $t_p(net)$

$$\begin{aligned} &= t_{pHL}(\text{FF}) + t_{pLH}(\text{NOT}) + t_{pLH}(\text{OR}) \\ &= (0.47 + 0.022 \times 2) + (0.02 + 0.038 \times 2) \\ &\quad + (0.12 + 0.037L) \\ &= 0.73 + 0.037L \text{ [ns]} \end{aligned}$$

SIZE:

$$\begin{aligned} &= 3 + 2 \times 5 + 8 \times 2 \\ &= 29 \text{ equivalent gates} \end{aligned}$$

# DESIGN OF SEQUENTIAL FLIP-FLOP NETWORKS

- EXCITATION FUNCTION $E(Q(t), Q(t+1))$

| FROM | TO | INPUTS SHOULD BE |
|---|---|---|
| $Q(t) = 0$ | $Q(t+1) = 0$ | $S(t) = 0,\ R(t) = dc$ |
| $Q(t) = 0$ | $Q(t+1) = 1$ | $S(t) = 1,\ R(t) = 0$ |
| $Q(t) = 1$ | $Q(t+1) = 0$ | $S(t) = 0,\ R(t) = 1$ |
| $Q(t) = 1$ | $Q(t+1) = 1$ | $S(t) = dc,\ R(t) = 0$ |

# EXCITATION FUNCTIONS

## D flip-flop

| $PS$ | $NS$ | |
|---|---|---|
| | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| | $D(t)$ | |

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

## SR flip-flop

| $PS$ | $NS$ | |
|---|---|---|
| | 0 | 1 |
| 0 | 0- | 10 |
| 1 | 01 | -0 |
| | $S(t)$ | $R(t)$ |

## JK flip-flop

| $PS$ | $NS$ | |
|---|---|---|
| | 0 | 1 |
| 0 | 0- | 1- |
| 1 | -1 | -0 |
| | $J(t)$ | $K(t)$ |

## T flip-flop

| $PS$ | $NS$ |
|---|---|
| | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 0 |
| | $T(t)$ | |

$$T(t) = Q(t) \oplus Q(t+1)$$

# THE DESIGN PROCEDURE

1. OBTAIN A BINARY DESCRIPTION OF THE SYSTEM

2. SELECT THE TYPE OF FLIP-FLOP

3. DETERMINE THE INPUTS TO THE FLIP-FLOPS (use the excitation function)

4. DESIGN A COMBINATIONAL NETWORK

# EXAMPLE 8.8: DESIGN MODULO-5 COUNTER

- USE T FLIP-FLOPS

  Input: $x(t) \in \{0, 1\}$
  Output: $z(t) \in \{0, 1, 2, 3, 4\}$
  State: $s(t) \in \{S_0, S_1, S_2, S_3, S_4\}$
  Initial state: $s(0) = S_0$

  Functions: Counts modulo-5, i.e.,
  (0,1,2,3,4,0,1,2,3,4,0...),

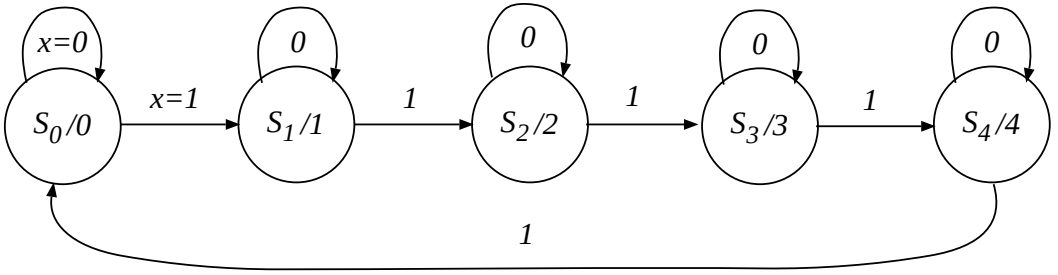



Figure 8.27: STATE DIAGRAM FOR Example 8.8.

# EXAMPLE 8.8 (cont.)

| $z$ | $z_2$ | $z_1$ | $z_0$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 2 | 0 | 1 | 0 |
| 3 | 0 | 1 | 1 |
| 4 | 1 | 0 | 0 |

| $PS$ | Input | | Input | |
|---|---|---|---|---|
| $Q_2Q_1Q_0$ | $x=0$ | $x=1$ | $x=0$ | $x=1$ |
| 000 | 000 | 001 | 000 | 001 |
| 001 | 001 | 010 | 000 | 011 |
| 010 | 010 | 011 | 000 | 001 |
| 011 | 011 | 100 | 000 | 111 |
| 100 | 100 | 000 | 000 | 100 |
| | $NS$ | | $T_2T_1T_0$ | |

DON'T CARES: 5, 6, AND 7

# EXAMPLE 8.8 (cont.)

## sm – STATE MAP

sm:

| $Q_2 \backslash x$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 2 | 3 |
| 3 | 6 | 7 |
| 2 | 4 | 5 |

($Q_1$ / $Q_0$)

$T_2$:

| $Q_2 \backslash x$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 3 | - | - |
| 2 | 0 | 1 |

($Q_1$ / $Q_0$)

$T_1$:

| $Q_2 \backslash x$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| 3 | - | - |
| 2 | 0 | 0 |

($Q_1$ / $Q_0$)

$T_0$:

| $Q_2 \backslash x$ | 0 | 1 |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 1 | 1 |
| 3 | - | - |
| 2 | 0 | 0 |

($Q_1$ / $Q_0$)

$$T_2 = xQ_2 + xQ_1Q_0$$
$$T_1 = xQ_0$$
$$T_0 = xQ_2'$$

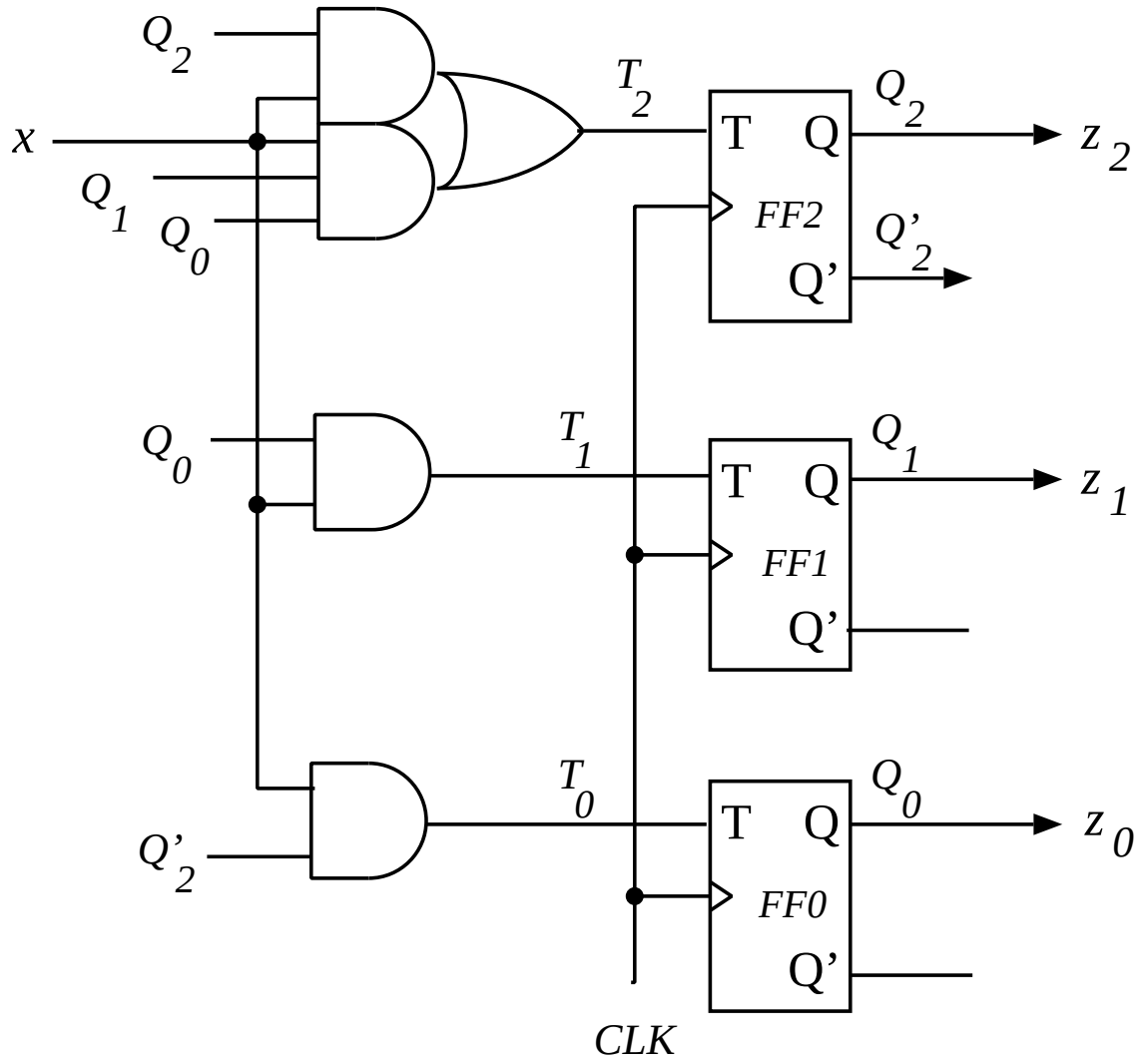



Figure 8.28: SEQUENTIAL NETWORK IN Example 8.8.

# EXAMPLE 8.9: DESIGN

Input: $\underline{x}(t) = (x_1, x_0),\ x_i \in \{0, 1\}$
Output: $z(t) \in \{0, 1\}$
State: $s(t) \in \{a, b, c, d\}$
Initial state: $s(0) = a$

Functions: The transition and output functions

| $PS$ | $x_1x_0$ | | |
|---|---|---|---|
| | 01 | 10 | 11 |
| $a$ | $b$,0 | $c$,1 | $c$,0 |
| $b$ | $a$,0 | $d$,1 | $d$,0 |
| $c$ | $d$,0 | $c$,0 | $a$,1 |
| $d$ | $c$,0 | $a$,0 | $d$,1 |
| | $NS,\ z$ | | |

# EXAMPLE 8.9 (CONT.)

| State | $Q_1Q_0$ |
|---|---|
| $a$ | 00 |
| $b$ | 01 |
| $c$ | 10 |
| $d$ | 11 |

| $PS$ | $x_1x_0$ | | |
|---|---|---|---|
| $Q_1Q_0$ | 01 | 10 | 11 |
| 00 | 01 | 10 | 10 |
| 01 | 00 | 11 | 11 |
| 10 | 11 | 10 | 00 |
| 11 | 10 | 00 | 11 |
| | $NS$ | | |

| $Q(t)$ | $Q(t+1)$ | $S$ | $R$ |
|---|---|---|---|
| 0 | 0 | 0 | - |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | - | 0 |

$S_1$:

| $Q_1 \backslash x_1x_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 0 | 1 | 1 |
| 1 | - | 0 | 1 | 1 |
| $Q_0$ | - | - | - | 0 |
| $Q_0$ | - | - | 0 | - |

$R_1$:

| $Q_1 \backslash x_1x_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | - | 0 | 0 |
| 1 | - | - | 0 | 0 |
| $Q_0$ | - | 0 | 0 | 1 |
| $Q_0$ | - | 0 | 1 | 0 |

$S_0$:

| $Q_1 \backslash x_1x_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 1 | 0 | 0 |
| 1 | - | 0 | - | - |
| $Q_0$ | - | 0 | - | 0 |
| $Q_0$ | - | 1 | 0 | 0 |

$R_0$:

| $Q_1 \backslash x_1x_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | - | 0 | - | - |
| 1 | - | 1 | 0 | 0 |
| $Q_0$ | - | 1 | 0 | 1 |
| $Q_0$ | - | 0 | - | - |

$$
\begin{aligned}
S_1 &= x_1 Q_1' \\
R_1 &= x_0' Q_1 Q_0 + x_1 x_0 Q_1 Q_0' \\
S_0 &= x_1' Q_0' \\
R_0 &= x_1' Q_0 + x_0' Q_1
\end{aligned}
$$

$z$:

| $Q_1$ \ $x_0$ | 0 | 0 | 1 |
|---|---|---|---|
| - | 0 | 0 | 1 |
| - | 0 | 1 | 0 |
| - | 0 | 1 | 0 |

$Q_0$ / $x_1$

THE OUTPUT EXPRESSION IS

$$z = x_0' Q_1' + x_1 x_0 Q_1$$

# EXAMPLE 8.9 (cont.)

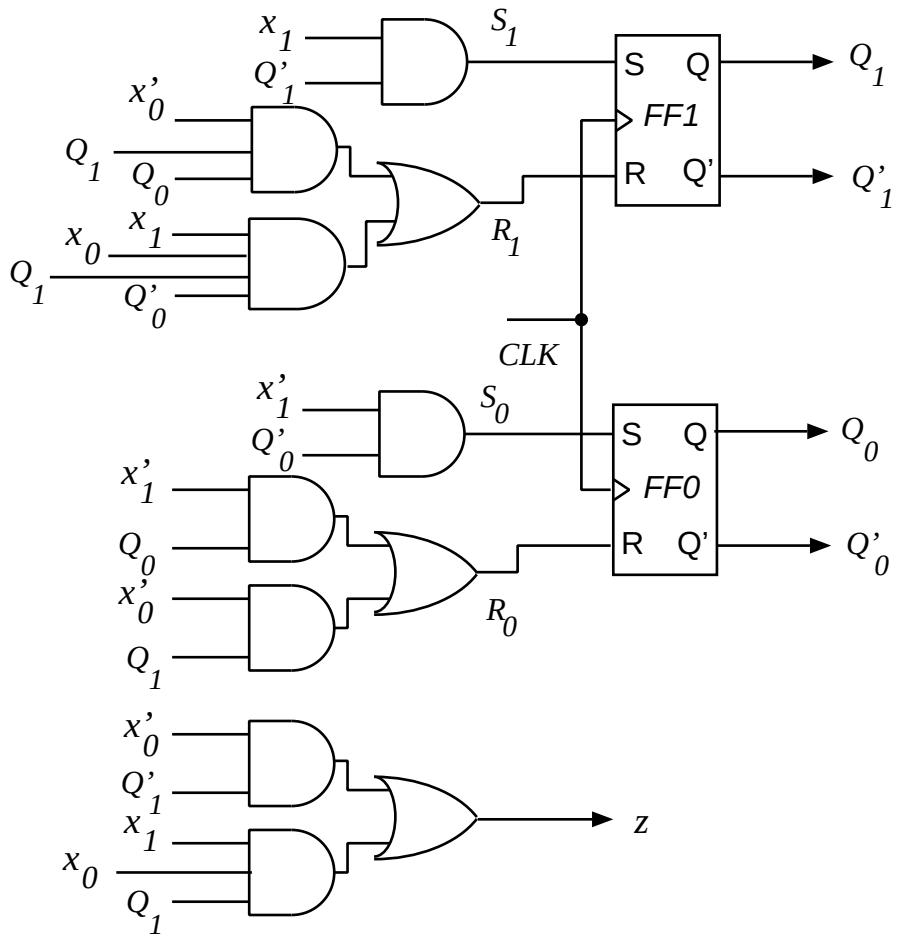



Figure 8.29: SEQUENTIAL NETWORK IN Example 8.9.

# SPECIAL STATE ASSIGNMENTS

- ONE FLIP-FLOP PER STATE

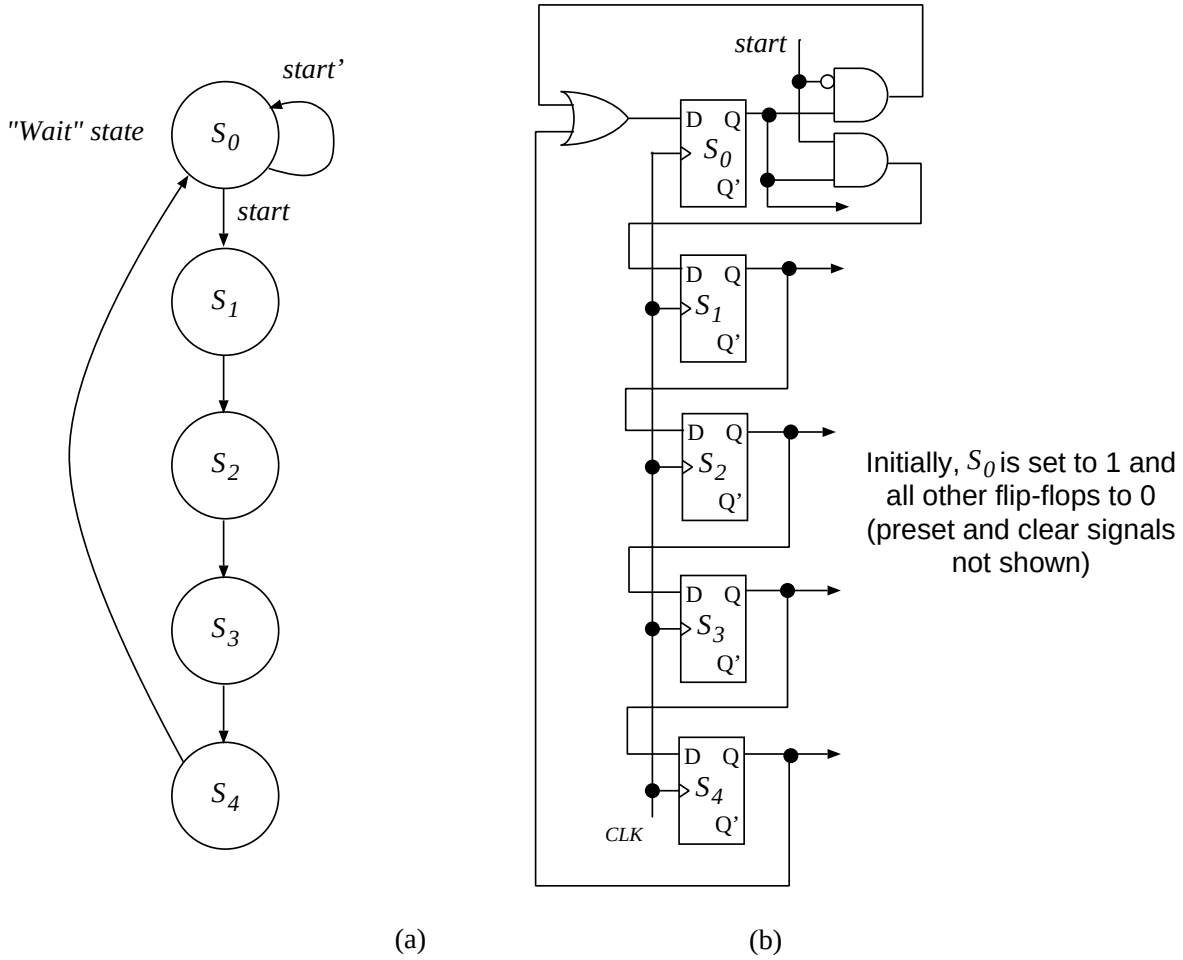



Figure 8.30: ONE FLIP-FLOP PER STATE APPROACH: a) STATE DIAGRAM. b) IMPLEMENTATION (Outputs omitted).

# PRIMITIVES FOR "one-flip-flop-per-state" APPROACH

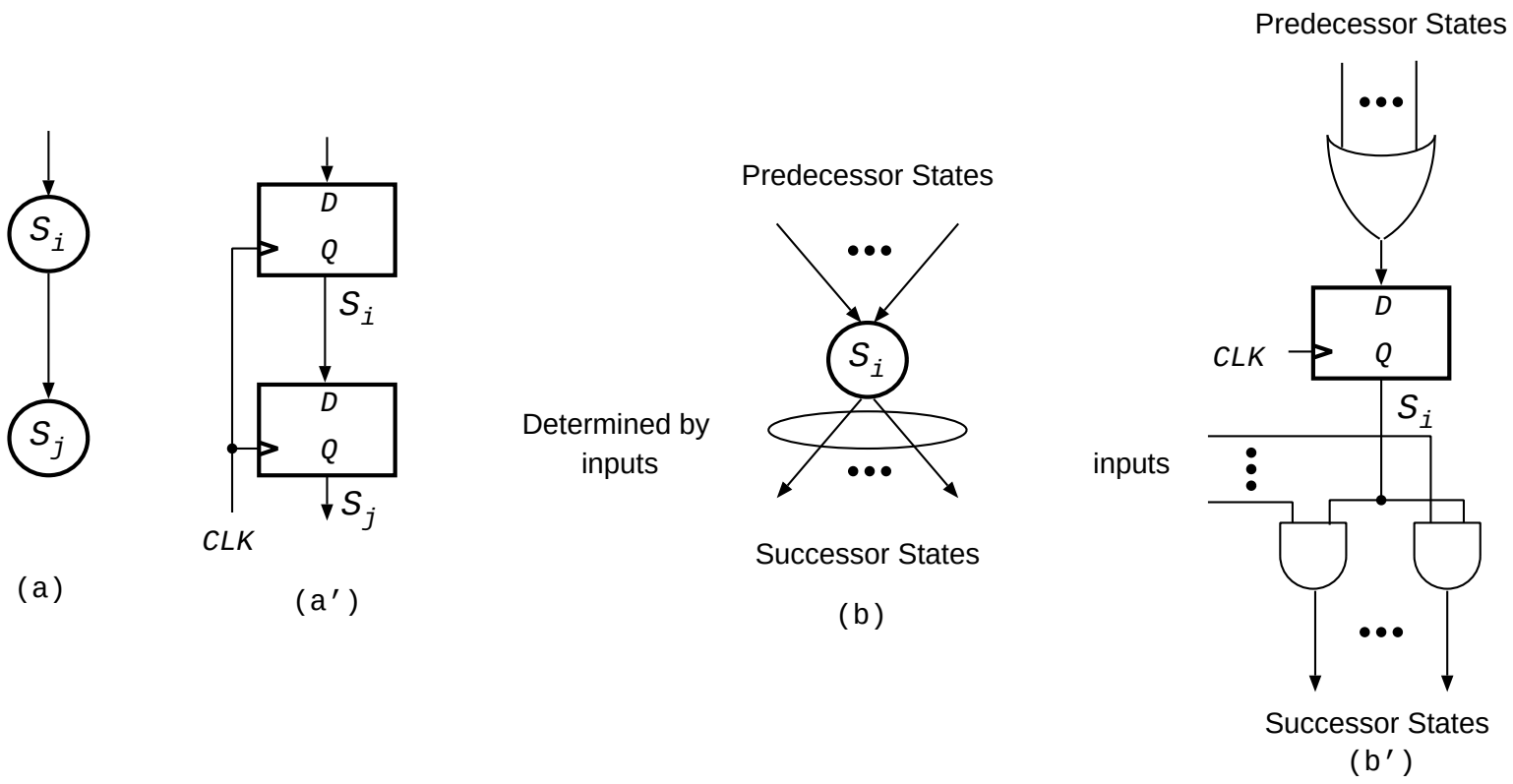



Figure 8.31: PRIMITIVES FOR THE "ONE-FLIP-FLOP-PER-STATE" APPROACH.

# CONTROLLER FOR SIMPLE VENDING MACHINE

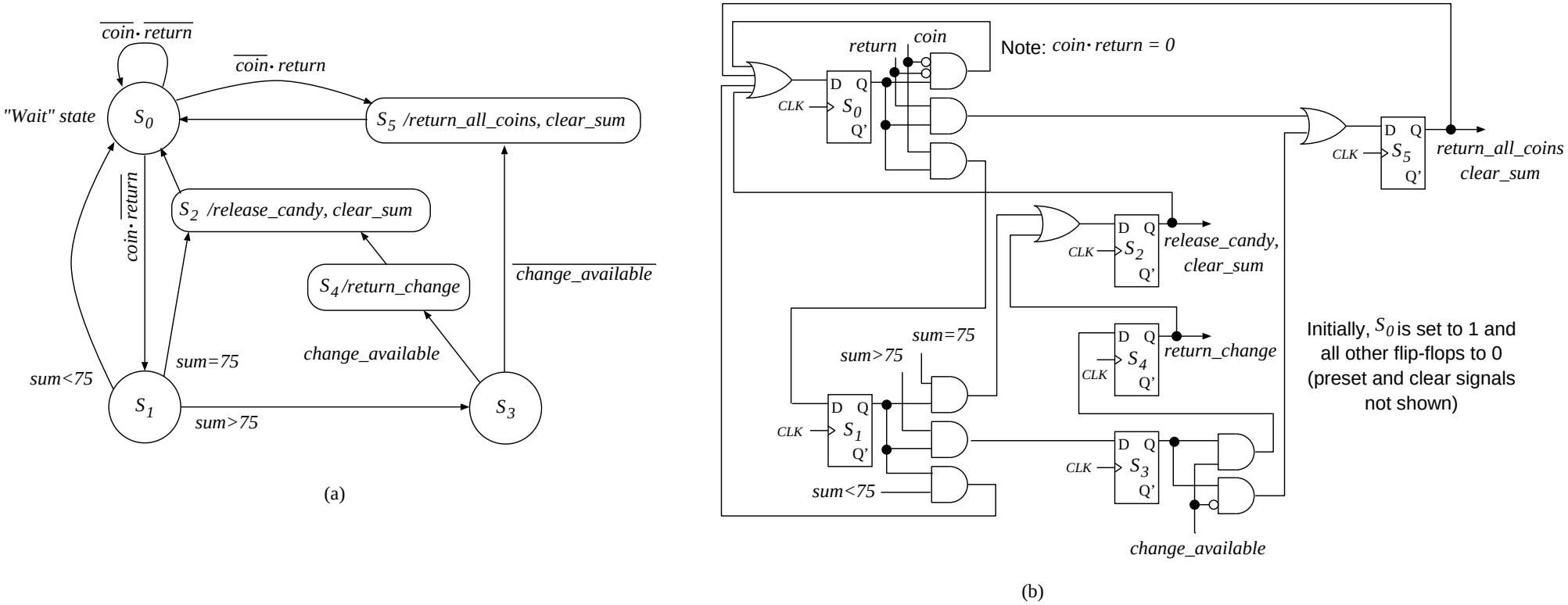



Figure 8.32: A ONE-FLIP-FLOP-PER-STATE IMPLEMENTATION OF A CONTROLLER FOR VENDING MACHINE: a) STATE DIAGRAM. b) IMPLEMENTATION.

# SHIFTING STATE REGISTER: Example 8.10

INPUT: $x(t) \in \{0, 1\}$

OUTPUT: $z(t) \in \{0, 1\}$

FUNCTION: $z(t) = \begin{cases} 1 \ \textbf{if} \ \ x(t-3, t) = 1101 \\ 0 \ \textbf{otherwise} \end{cases}$

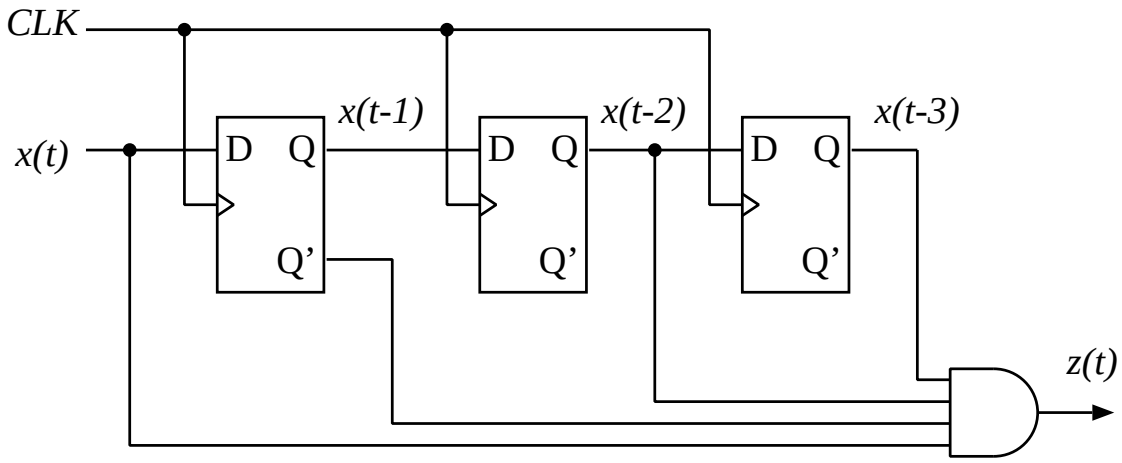



Figure 8.33: IMPLEMENTATION OF PATTERN RECOGNIZER IN EXAMPLE 8.10.