Sequential Logic Elements

In this chapter you will learn about
• Logic circuits that can store information;
• Basic cells, latches, and flip-flops;
• State diagrams;
• Design techniques for circuits that use flip-flops.
Circuit Types

- **Combinational** – output depends only on the input.
- **Sequential** – output depends on input and past behavior
  - Requires use of storage elements;
  - Contents of the storage elements is called *state*;
  - Circuit goes through a sequence of states as a result of changes in inputs.
- **Synchronous** – controlled by a clock.

Clock Signals

![Clock Signals Diagram](image)

Figure 7-1
Clock signals: (a) active high; (b) active low.
A Bistable Memory Element

- Bistable – possessing two stable states.

A Set/Reset (SR) Memory Element

- Called a Basic Cell;
- NOR centered Basic Cell:
  - Circuit (a), Function table (b).
- Inputs are active when they are high;
- Blocking side inputs.
Typical Operation of a Basic Cell

- Reset, clear.
- Set, preset.

NOR-Centered (Reset Dominant) Basic Cell

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Action</th>
<th>Q_{n+1}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>hold</td>
<td>Q_{n}</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>reset</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>set</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>reset</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inputs</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 → 0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 → 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1 → 0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1 → 1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

$Q_{n+1} = \overline{S} + R$

State Diagram
NAND-Centered (Set Dominant) Basic Cell

- Inputs are active low (when they are "asserted");
- Operation table indicates assertion, not voltage, levels.

**Circuit**

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Qn</th>
<th>Qn+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>hold</td>
<td>Qn</td>
</tr>
<tr>
<td>0 1</td>
<td>reset</td>
<td>0</td>
</tr>
<tr>
<td>1 0</td>
<td>set</td>
<td>1</td>
</tr>
<tr>
<td>1 1</td>
<td>set</td>
<td>1</td>
</tr>
</tbody>
</table>

**Operation Table**

<table>
<thead>
<tr>
<th>S R</th>
<th>Action</th>
<th>Qn+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>hold</td>
<td>Qn</td>
</tr>
<tr>
<td>0 1</td>
<td>reset</td>
<td>0</td>
</tr>
<tr>
<td>1 0</td>
<td>set</td>
<td>1</td>
</tr>
<tr>
<td>1 1</td>
<td>set</td>
<td>1</td>
</tr>
</tbody>
</table>

**Excitation Table**

<table>
<thead>
<tr>
<th>Qn→Qn+1</th>
<th>Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 → 0</td>
<td>0</td>
</tr>
<tr>
<td>0 → 1</td>
<td>1</td>
</tr>
<tr>
<td>1 → 0</td>
<td>0</td>
</tr>
<tr>
<td>1 → 1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Combined Form of the Basic Cell**

**NAND-centered**

**Excitation Table**

<table>
<thead>
<tr>
<th>Qn→Qn+1</th>
<th>Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 → 0</td>
<td>0</td>
</tr>
<tr>
<td>0 → 1</td>
<td>1</td>
</tr>
<tr>
<td>1 → 0</td>
<td>0</td>
</tr>
<tr>
<td>1 → 1</td>
<td>1</td>
</tr>
</tbody>
</table>

**NOR-centered**

**Excitation Table**

<table>
<thead>
<tr>
<th>Qn→Qn+1</th>
<th>Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 → 0</td>
<td>0</td>
</tr>
<tr>
<td>0 → 1</td>
<td>1</td>
</tr>
<tr>
<td>1 → 0</td>
<td>0</td>
</tr>
<tr>
<td>1 → 1</td>
<td>1</td>
</tr>
</tbody>
</table>
Designing Latches - A Model

- Latch - a logic circuit that transfers the input state to the output state when the clock signal is high and latches and holds the input when the clock signal goes low.

Excitation Table

<table>
<thead>
<tr>
<th>Q&lt;sub&gt;0&lt;/sub&gt; → Q&lt;sub&gt;0+1&lt;/sub&gt;</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 → 0</td>
<td>0</td>
<td>ꞏ</td>
</tr>
<tr>
<td>0 → 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1 → 0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1 → 1</td>
<td>ꞏ</td>
<td>0</td>
</tr>
</tbody>
</table>

Combined Form

Design of a Clocked D (Data) Latch
Design of a Clocked D Latch

Block Diagram

Next State Logic

Set

Reset

Q(H)

D

Clk

Truth Table

Inputs

CLK

D

S

R

Qn

Qn+1

<table>
<thead>
<tr>
<th>Clk</th>
<th>D</th>
<th>On</th>
<th>Qn+1</th>
<th>Set</th>
<th>Reset</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Equations

Set = clk \cdot D
Reset = clk \cdot \overline{D}

State Diagram

Clocked D Latch

Figure 7-12

D latch: (a) circuit design using NAND gates; (b) function table; (c) logic symbol.

Figure 7-13

Functional behavior of a D latch for various inputs.
Design of a Clocked Toggle (T) Latch

**Block Diagram**

**Truth Table**

<table>
<thead>
<tr>
<th>Clk T</th>
<th>Action</th>
<th>Qn+1</th>
<th>Inputs</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Hold</td>
<td>Qn</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>Hold</td>
<td>Qn</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>Toggle</td>
<td>(\overline{Q_n})</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Equations**

\[ Set = \text{clk} \cdot \overline{T} \cdot \overline{\overline{Q}} \]

\[ Reset = \text{clk} \cdot \overline{T} \cdot \overline{Q} \]

Design of a Clocked JK Latch

**Block Diagram**

**Truth Table**

<table>
<thead>
<tr>
<th>Clk J K</th>
<th>Action</th>
<th>Qn+1</th>
<th>Inputs</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>Hold</td>
<td>Qn</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 1</td>
<td>Hold</td>
<td>Qn</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 0</td>
<td>Reset</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 1</td>
<td>Set</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Equations**

\[ Set = \text{clk} \cdot J \cdot \overline{Q} \]

\[ Reset = \text{clk} \cdot K \cdot Q \]
Design of a Set-Dominant Clocked SR Latch

- Inputs – S, R, Clock, Q
- Outputs – Set, Reset

Block Diagram

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Action</th>
<th>( Q_{\text{next}} )</th>
<th>Inputs</th>
<th>S</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>hold</td>
<td>( Q_N )</td>
<td>0 → 0</td>
<td>0</td>
<td>φ</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>reset</td>
<td>0</td>
<td>0 → 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>set</td>
<td>1</td>
<td>1 → 0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>set</td>
<td>1</td>
<td>1 → 1</td>
<td>( \phi )</td>
<td>0</td>
</tr>
</tbody>
</table>

Excitation Table

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>clk</th>
<th>( Q_N )</th>
<th>Set</th>
<th>Reset</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>( \phi )</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>( \phi )</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>( \phi )</td>
<td>0</td>
</tr>
</tbody>
</table>

Equations

- Set = \( clk \cdot S \)
- Reset = \( clk \cdot \overline{R} \overline{S} \)
**SR Latch with Enable (Clock)**

![SR Latch with Enable (Clock) Diagram]

**Design of a Clocked JK Latch – Version II**

- Replace the basic cell with a D-latch as the memory element.

![Design of a Clocked JK Latch – Version II Diagram]

- Inputs – J, K, Clk, Q
- Output – D
Design of a Clocked JK Latch – Version II

Terminology

- Latches are often called *Transparent* because the output will follow the input as long as the clock signal is high.
- Flip-flops are *edge-triggered*
  - Positive-edge triggered (PET) is when action occurs on the rising edge of the clock signal;
  - Negative-edge triggered (NET) is when action occurs on the falling edge of the clock signal.
- Types of Flip-flops
  - SR (rarely used);
  - D (very, very common, 74HC74);
  - JK (hardly ever used, 74HC109);
  - Toggle (occasionally used by CAD programs).
Setup and Hold Times

• Setup time ($t_{SU}$) is the time interval preceding the active transition point of the CLK during which all data inputs must remain stable.
• Hold time ($t_{H}$) is the time interval following the active transition point of the CLK during which all data inputs must remain stable.
• See data sheet for 74HC74

PET Master-Slave D Flip-Flop

• QM follows the D input whenever CLK is low.
• When CLK goes high, QM is transferred to the output.
Positive-Edge-Triggered D Flip-Flop

![Circuit Diagram](image)

Figure 7-20
Commercial circuit for a positive-edge-triggered D flip-flop such as 74LS74.

PET D Flip-Flop with *Clear* and *Preset*

- Synchronous – transitions or actions occur in relation to the CLK signal;
- Asynchronous – transitions or actions are not related to the CLK signal.

![PET D Flip-Flop Diagram](image)

Figure 7-19
Positive-edge-triggered D flip-flop with preset and clear:
(a) logic symbol; (b) circuit design using NAND gates.
Level-Sensitive vs. Edge-Triggered

- Level-sensitive = latch
- Edge-triggered = flip-flop

Design a T Flip-Flop from a D Flip-Flop

- The memory element is now edge-triggered meaning the Clk signal is no longer part of the next-state logic.
Design a T Flip-Flop from a D Flip-Flop

![T Flip-Flop Circuit](image)

Design a JK Flip-Flop from a D Flip-Flop

![JK Flip-Flop Circuit](image)

Function Table

<table>
<thead>
<tr>
<th>J</th>
<th>K</th>
<th>Q_{(n+1)}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q_{(n)}</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Q_{(n)}</td>
</tr>
</tbody>
</table>

Graphical symbol
Summary of Terminology

- Basic cell – cross-coupled NAND/NOR.
- Gated latch – output changes only when $Clk$ is asserted
  - Gated SR latch;
  - Gated D latch;
  - Gated JK latch.
- Flip-flop – output changes only on $Clk$ edge
  - Master-slave;
  - Edge-triggered;
  - Three main types
    - D (very, very common, 74HC74);
    - JK (hardly ever used, 74HC109);
    - Toggle (occasionally used by CAD programs).

Sequential Logic Elements Summary

In this chapter you learned about
- Logic circuits that can store information;
- Basic cells, latches, and flip-flops;
- State diagrams;
- Design techniques for circuits that use flip-flops.