Chapter 6  Sequential Circuits —
circuits with feedback.

Examples:  FSM

D-type latch

output a function of not only present inputs
but also past inputs (reflected in the current state)
Cross-coupled NOR gates

\[
\begin{array}{c|c|c}
\text{NOR} & 00 & 10 \\
01 & 10 & 00 \\
11 & 00 & 00 \\
\end{array}
\]

so if both \( R \) and \( S = 0 \) then
\[ \text{Cross-coupled NORs } \Rightarrow \text{ stable latch.} \]

what happens if \( R \) or \( S \) or both become \( \neq 0 \)?

\[
\begin{array}{c}
R \quad Q \quad 0 \rightarrow 1 \\
S \quad \bar{Q} \quad 1 \rightarrow 0 \\
0 \rightarrow 1 \rightarrow 0
\end{array}
\]

Asynchronous State Transition Diagram

\[
\text{SR}=00 \quad \text{SR}=01 \quad \text{SR}=10
\]

\[
\begin{array}{c|c}
\text{SR} & \text{Q} \\
00 & \text{Hold} \\
01 & 0 \\
10 & \text{forbidden} \\
\end{array}
\]
J-K Latch

To eliminate the forbidden state.

\[
\begin{array}{c}
J & K & Q(t) & Q(t+\Delta) \\
0 & 0 & 0 & 0 \\
0 & 1 & 0 & \text{hold} \\
1 & 0 & 0 & \text{reset} \\
1 & 1 & 0 & \text{set} \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \text{toggle} \\
\end{array}
\]

J = "run"
K = "kill"
Level Sensitive RS Latch

R

S

CLK

Q

\( \overline{Q} \)

(It is generally not a good idea to AND the clock. This circuit is a functional depiction of the actual circuit.)

Edge-triggered FF version also possible.
Using JK (SR) flip-flops

With d-type FF state elements, new state is computed based on inputs & present state bits, & reloaded each cycle.

With JK FF state elements, inputs are used to determine conditions under which to set or reset state bits.

Example: 1-bit serial adder (assume LSB first)

\[
\begin{array}{c}
\uparrow A \\
\hline \\
\uparrow B \\
\hline
\end{array}
\]

With D-type FF for carry

With JK FF for carry

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>Cin</th>
<th>S</th>
<th>Cout</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</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>
</tr>
<tr>
<td>0</td>
<td>1</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>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Cany kill \( ab \)

Cany generate \( ab \)
## Storage Element Taxonomy

<table>
<thead>
<tr>
<th>Synchronous</th>
<th>Asynchronous</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level-sensitive</td>
<td>Edge-triggered</td>
</tr>
<tr>
<td>✔️ D-type</td>
<td>✔️</td>
</tr>
<tr>
<td>✔️ JK-type (SR-type)</td>
<td>✔️</td>
</tr>
<tr>
<td>&quot;latch&quot;</td>
<td>&quot;FF&quot;</td>
</tr>
</tbody>
</table>

**Ex:**
- **D-type**
  - ![D-type Diagram](image)
  - "level sensitive" "transparent"
  - "edge-triggered"

- **JK-type**
  - ![JK-type Diagram](image)
  - "asynchronous" "unclocked"