Flip-flop: Digital memory unit

You cannot build a memory unit by combining gates:

How do you design a memory unit?

SR latch (set-reset latch)

\[
\begin{align*}
S=1, R=0 & \quad \rightarrow \quad Q=1 \\
S=0, R=1 & \quad \rightarrow \quad Q=0 \\
S=0, R=0 & \quad \rightarrow \quad Q \text{ remains (latch mode)} \\
S=1, R=1 & \quad : \quad \text{unused mode}
\end{align*}
\]

Implementation

NOR

\[
\begin{align*}
S & \quad \rightarrow \quad \neg Q \\
R & \quad \rightarrow \quad Q
\end{align*}
\]

Truth table

\[
\begin{array}{c|cc}
A & 0 & 1 \\
B & 0 & 1 & 0 & 0
\end{array}
\]

So when \( S=1, R=0 \) \( \Rightarrow \) \( Q=1, \neg Q=0 \)
when \( S=0, R=1 \) \( \Rightarrow \) \( Q=0, \neg Q=1 \)
when \( S=0, R=0 \) \( \Rightarrow \) \( \text{unchange} \)
when \( S=1, R=1 \) \( \Rightarrow \) \( Q=\neg Q=0 \) \( \text{unused mode} \)

Problem with simple SR flip-flop:
Gated SR latch

Only when \( G = \text{on} \) the above resume the SR-FF
When \( G = \text{off} \), SR-FF always in the hold-mode.

\( \Rightarrow \) System is updated only when clock is on.

Dtype flip-flop

Master works on raising edge of \( \text{CLK} \)
Slave works on the lowering edge of \( \text{CLK} \)

\( Q/\bar{Q} \) time for master to get ready
\( S/R \) time for slave to get ready
Does D flip flop have true memory?

JK Flip-Flop:

<table>
<thead>
<tr>
<th>J</th>
<th>K</th>
<th>action</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>unchange</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Q = 1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Q = 0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>toggle Q → ¬Q</td>
</tr>
</tbody>
</table>

Assume J = K = 1, Q = 0

1. D switches from 1 to 0
   → Q unchanged.

2. Q switches from 0 to 1
   → Assume J = K = 1.
   1. If Q = 0, D switches will change ⇒ Q = 1
   2. If Q = 1, D switches will change ⇒ Q = 0

Check for self-consistency yourself.