Chapter #6: Sequential Logic Design

6.3 -- Flip-Flop Types
Realing Circuits with Different Kinds of FFs

Choosing a Flipflop

R-S Clocked Latch:
used as storage element in narrow width clocked systems
its use is not recommended!
however, fundamental building block of other flipflop types

J-K Flipflop:
versatile building block
can be used to implement D and T FFs
usually requires least amount of logic to implement \( f(\text{In},Q,Q+) \)
but has two inputs with increased wiring complexity

because of 1's catching, never use master/slave J-K FFs
edge-triggered varieties exist

D Flipflop:
minimizes wires, much preferred in VLSI technologies
simplest design technique
best choice for storage registers

T Flipflops:
don't really exist, constructed from J-K FFs
usually best choice for implementing counters

Preset and Clear inputs highly desirable!!
Realizing Circuits with Different Kinds of Flipflops

Characteristic Equations

R-S: \[ Q^+ = S + \overline{R} \overline{Q} \]

D: \[ Q^+ = D \]

J-K: \[ Q^+ = J \overline{Q} + K \overline{Q} \]

T: \[ Q^+ = T \overline{Q} + T \overline{Q} = T \oplus Q \]

Derived from the K-maps for \( Q^+ = f(\text{Inputs}, Q) \)

E.g., \( J=K=0 \), then \( Q^+ = Q \)
\( J=1, K=0 \), then \( Q^+ = 1 \)
\( J=0, K=1 \), then \( Q^+ = 0 \)
\( J=1, K=1 \), then \( Q^+ = Q \)

Implementing One FF in Terms of Another
Realizing Circuits with Different Kinds of Flipflops

**Design Procedure**

**Excitation Tables:** What are the necessary inputs to cause a particular kind of change in state?

<table>
<thead>
<tr>
<th>Q</th>
<th>Q^+</th>
<th>R</th>
<th>S</th>
<th>J</th>
<th>K</th>
<th>T</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>X</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>X</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Realizing Circuits with Different Kinds of Flipflops

Procedure for Realizing a Flip-Flop (#1) using another type of Flip-Flop (#2)

(a) Start with K-map of FF #1 ===> \( Q^+ = f(\text{inputs of FF #1, Q}) \)

(b) Create K-map for each input of FF #2 with the same inputs used in part (a)

(c) Fill in K-maps with appropriate values of FF#2 inputs to cause the same state changes as in the K-map of part (a).
Implementing D FF with a J-K FF:

#1 -- D FF ===> inputs = D  
#2 -- J-K FF ===> inputs = J,K

a) Start with K-map of \( Q^+ = f(D, Q) \)

\[
\begin{array}{c|c|c}
D & 0 & 1 \\
\hline
Q & 0 & 1 & 1 \\
\hline
Q^+ & D & 0 & 1 \\
\end{array}
\]

b) Create K-maps for J and K with same inputs (D, Q)

c) Fill in K-maps with appropriate values for J and K to cause the same state changes as in the original K-map

E.g., \( D = Q = 0, Q^+ = 0 \) then \( J = 0, K = X \)

\[
\begin{array}{c|c|c}
D & 0 & 1 \\
\hline
Q & 0 & 1 & 1 \\
\hline
J & X & X \\
\end{array}
\]  
\[
\begin{array}{c|c|c}
D & 0 & 1 \\
\hline
Q & 0 & 1 & X \\
\hline
K & X & D \\
\end{array}
\]
Realizing Circuits with Different Kinds of Flipflops

D implemented with J-K

![Diagram of J-K flipflop implementation for D]

© R.H. Katz   Transparency No. 13-7
Realizing Circuits with Different Kinds of Flipflops

Design Procedure (Continued)

Implementing J-K FF with a D FF:

a) K-Map of $Q^+ = F(J, K, Q)$

\[
\begin{array}{cccc}
  & J & K & Q^+ \\
 J & 00 & 01 & 11 & 10 \\
 Q & 0 & 0 & 0 & 1 \\
 & 1 & 0 & 0 & 1 \\
\end{array}
\]

$Q^+ = JQ' + K'Q$

b, c) Revised K-map using D's excitation table.

It's the same! That is why design procedure with D FF is simple!

Resulting equation is the combinational logic input to D to cause same behavior as J-K FF. Of course it is identical to the characteristic equation for a J-K FF.
Realizing Circuits with Different Kinds of Flipflops

J-K implemented with D

[Diagram of J-K flipflop implemented with D flipflops]