Exercise 8.2:

Using the notation given on Figure 8.15 of the textbook we obtain the following network parameters:

Combination Networks’ delays:

\[ d_1_x = 4 \times t_p(\text{gate}) = 2 \text{ms} \]
\[ d_2 = 1 \times t_p(\text{gate}) = 0.5 \text{ms} \]

Network set-up time:

\[ t_{\text{set}}(\text{net}) = \frac{t_{\text{cell}}}{2} = t_{\text{set}}(\text{cell}) + d_1_x = 1 + 2 = 3 \text{ms} \]

Network hold time:

\[ t_h(\text{net}) = t_h(\text{cell}) = 0.5 \text{ms} \]

Network propagation delay:

\[ t_p(\text{net}) = t_p(\text{cell}) + d_2 = 3 + 0.5 = 3.5 \text{ms} \]
Exercise 8.8

We need a 3-bit vector to represent the six states and a 2-bit vector to represent the output. Let us define the following encoding:

<table>
<thead>
<tr>
<th>$y_2y_1y_0$</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>A</td>
</tr>
<tr>
<td>001</td>
<td>B</td>
</tr>
<tr>
<td>010</td>
<td>C</td>
</tr>
<tr>
<td>011</td>
<td>D</td>
</tr>
<tr>
<td>100</td>
<td>E</td>
</tr>
<tr>
<td>101</td>
<td>F</td>
</tr>
</tbody>
</table>

From the state table and the encoding we get the following K-maps.

The corresponding switching expressions are

$Y_0 = y_0$

$Y_1 = x' y_2 y_1 y_0 + x' y_1 y_0 + x y_2 y_0 + x y_1 y_0$

$Y_2 = x' y_2 y_1 y_0 + x' y_1 y_0 + x y_2 y_0 + x y_2 y_1 y_0$

$x_1 = y_2 y_0 + x y_0$

$x_0 = x' y_1 y_0 + y_2 y_0 + x y_0$

The sequential network is shown in Figure 8.8 on page 128.
Figure 8.8: Sequential network for Exercise 8.8
Exercise 8.16 From the network we obtain the following table, based on the expressions below:

\[ J_A = K_A = xQ_B \]
\[ J_B = K_B = x \]

<table>
<thead>
<tr>
<th>PS (Q_AQ_B)</th>
<th>Input (x = 0)</th>
<th>Input (x = 1)</th>
<th>Input (x = 0)</th>
<th>Input (x = 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0000</td>
<td>0011</td>
<td>00</td>
<td>01</td>
</tr>
<tr>
<td>01</td>
<td>0000</td>
<td>1111</td>
<td>01</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>0000</td>
<td>0011</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>11</td>
<td>0000</td>
<td>1111</td>
<td>11</td>
<td>00</td>
</tr>
</tbody>
</table>

\(J_AK_AJ_BK_B\) NS

The outputs are expressed as:

\[ z_3 = Q_AQ_B \]
\[ z_2 = Q_AQ'_B \]
\[ z_1 = Q'_AQ_B \]
\[ z_0 = Q'_AQ'_B \]

Giving the following names to the states:

<table>
<thead>
<tr>
<th>State Name</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S_0)</td>
<td>00</td>
</tr>
<tr>
<td>(S_1)</td>
<td>01</td>
</tr>
<tr>
<td>(S_2)</td>
<td>10</td>
</tr>
<tr>
<td>(S_3)</td>
<td>11</td>
</tr>
</tbody>
</table>

we get the transition table:

<table>
<thead>
<tr>
<th>PS (Q_AQ_B)</th>
<th>Input (x = 0)</th>
<th>Input (x = 1)</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S_0)</td>
<td>(S_0)</td>
<td>(S_1)</td>
<td>0</td>
</tr>
<tr>
<td>(S_1)</td>
<td>(S_1)</td>
<td>(S_2)</td>
<td>1</td>
</tr>
<tr>
<td>(S_2)</td>
<td>(S_2)</td>
<td>(S_3)</td>
<td>2</td>
</tr>
<tr>
<td>(S_3)</td>
<td>(S_3)</td>
<td>(S_0)</td>
<td>3</td>
</tr>
</tbody>
</table>

NS

The state diagram for the given network is presented in Figure 8.19, and corresponds to a modulo-4 counter with decoded output.
Exercise 8.22

Since the excitation function of a D flip-flop is

\[ D(t) = Q(t+1) \]

the \( D \) input for each flip-flop implementation corresponds to its characteristic function. That is:

- for a SR flip-flop
  \[ D(t) = Q(t+1) = S + R'Q(t) \]

- for a T flip-flop
  \[ D(t) = Q(t+1) = T \oplus Q(t) \]

- for a JK flip-flop
  \[ D(t) = Q(t+1) = Q(t)J' + Q(t)'J \]

The corresponding networks are shown in Figures 8.25.
Figure 8.25: Networks of Exercise 8.22
Exercise 8.26
The state corresponds to the count. That is,

\[ s(t + 1) = (s(t) + 1) \mod 3 \]

Using a radix-2 representation for the count we get the following state table

<table>
<thead>
<tr>
<th>PS</th>
<th>Input</th>
<th>( x = 0 )</th>
<th>( x = 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>00</td>
<td>01</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Since the excitation function of a \( SR \) flip-flop is

<table>
<thead>
<tr>
<th>PS</th>
<th>NS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
</tr>
</tbody>
</table>

we get the following switching expressions

\[ S_2 = xQ_1 \]
\[ S_1 = xQ_2 Q_1' \]
\[ R_2 = xQ_2 \]
\[ R_1 = xQ_1 \]

The output is obtained directly from the state register. The sequential network is shown in Figure 8.29.

![Diagram](image)

Figure 8.29: Network for Exercise 8.26
Exercise 8.30 The modulo-5 counter presented in Example 8.8 was implemented using T-type flip-flops. For this problem we need to synthesize different functions for D-type flip-flops. The transition table, though, is exactly the same.

<table>
<thead>
<tr>
<th>PS</th>
<th>Input</th>
<th>x = 0</th>
<th>x = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q_dQ_1Q_0$</td>
<td>000</td>
<td>000</td>
<td>001</td>
</tr>
<tr>
<td></td>
<td>001</td>
<td>001</td>
<td>010</td>
</tr>
<tr>
<td></td>
<td>010</td>
<td>010</td>
<td>011</td>
</tr>
<tr>
<td></td>
<td>011</td>
<td>011</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>000</td>
</tr>
<tr>
<td></td>
<td>NS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Since the D-type flip-flops has as inputs the bits of the next state (shown in the transition table), the K-maps are:

\[
\begin{array}{cccc}
  \begin{array}{c}
    x = 0 \\
    Q_0
  \end{array} & \begin{array}{c}
    0 \\
    1 \\
    0
  \end{array} & \begin{array}{c}
    0 \\
    - \\
    1
  \end{array} & \begin{array}{c}
    0 \\
    - \\
    1
  \end{array} \\
\end{array}
\]

\[
\begin{array}{cccc}
  \begin{array}{c}
    x = 1 \\
    Q_0
  \end{array} & \begin{array}{c}
    0 \\
    1 \\
    1
  \end{array} & \begin{array}{c}
    0 \\
    - \\
    -
  \end{array} & \begin{array}{c}
    0 \\
    1 \\
    1
  \end{array} \\
\end{array}
\]

The inputs of the D-type flip-flops are:

\[
\begin{align*}
  D_2 &= xQ_1Q_0 + x'Q_2 \\
  D_1 &= xQ_2Q_0 + Q_1Q_2 + x'Q_1 \\
  D_0 &= xQ_2Q_0 + x'Q_0
\end{align*}
\]

The gate network for the sequential system is presented in Figure 8.34. Comparing Figure 8.34 with Figure 8.28 of the text we can easily see that this implementation uses more gates than the one done with T-type flip-flops.