Lecture 8: Synchronous Sequential Logic Example

Matthew Shuman
April 26th, 2013

1 NES Controller Example of Lecture 7 — Not in Text

Below is the seven step design process for state machines.

1. Define the state machine requirements
2. State diagram
3. State assignment
4. State table
5. Minimizations
6. State Equations
7. Logic Diagram

Another example of a state machine design process is a block that sends signals to read a NES controller.

1. Define the state machine requirements
   The schematic in figure 1 shows the circuitry inside of a NES Controller[1]. The waveforms in figure 2 show the two input signals, Latch and Clock, needed by the controller and the expected output signal, Data.

1http://seb.riot.org/nescontr/
Figure 1: The buttons are active low, there are two communication inputs, and there is one communication output.

Figure 2: The expected pattern for inputting Clock and Data are shown, as well as the timing of how the button data is output.

2. State diagram
   Translate the information in step 1 into a state diagram as shown in figure 3.

Figure 3: This state diagram pictures the operation of the state machine described in step 1.
3. State assignment

Assign each state in the state diagram a binary representation.

<table>
<thead>
<tr>
<th>State</th>
<th>BinaryValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
</tr>
<tr>
<td>2</td>
<td>0010</td>
</tr>
<tr>
<td>3</td>
<td>0011</td>
</tr>
<tr>
<td>4</td>
<td>0100</td>
</tr>
<tr>
<td>5</td>
<td>0101</td>
</tr>
<tr>
<td>6</td>
<td>0110</td>
</tr>
<tr>
<td>7</td>
<td>0111</td>
</tr>
<tr>
<td>8</td>
<td>1000</td>
</tr>
<tr>
<td>9</td>
<td>1001</td>
</tr>
<tr>
<td>A</td>
<td>1010</td>
</tr>
<tr>
<td>B</td>
<td>1011</td>
</tr>
<tr>
<td>C</td>
<td>1100</td>
</tr>
<tr>
<td>D</td>
<td>1101</td>
</tr>
<tr>
<td>E</td>
<td>1110</td>
</tr>
<tr>
<td>F</td>
<td>1111</td>
</tr>
</tbody>
</table>

4. State table

<table>
<thead>
<tr>
<th>State</th>
<th>BinaryValue, Q_3Q_2Q_1Q_0</th>
<th>NextState</th>
<th>NextStateBinaryValue, Q_3^*Q_2^*Q_1^<em>Q_0^</em></th>
<th>Clock</th>
<th>Latch</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
<td>1</td>
<td>0001</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
<td>2</td>
<td>0010</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0010</td>
<td>3</td>
<td>0011</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0011</td>
<td>4</td>
<td>0100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0100</td>
<td>5</td>
<td>0101</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0101</td>
<td>6</td>
<td>0110</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0110</td>
<td>7</td>
<td>0111</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0111</td>
<td>8</td>
<td>1000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>1000</td>
<td>9</td>
<td>1001</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>1001</td>
<td>A</td>
<td>1010</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>1010</td>
<td>B</td>
<td>1011</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1011</td>
<td>C</td>
<td>1100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>1100</td>
<td>D</td>
<td>1101</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>1101</td>
<td>E</td>
<td>1110</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>1110</td>
<td>F</td>
<td>1111</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>1111</td>
<td>0</td>
<td>0000</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

5. Minimizations

Find and minimize equations for Q_3^*, Q_2^*, Q_1^*, Q_0^*, Clock, and Latch. That will be 6 four variable K-maps. The inputs are Q_3, Q_2, Q_1, and Q_0.

6. State Equations

Write down the 6 State Equations.

7. Logic Diagram

Put the logic gates for Q_3^*, Q_2^*, Q_1^*, and Q_0^* into the next state logic of figure 4. These outputs will go into the D inputs of 4 D flip flops. The logic gates for Clock and Latch go into the output logic of 4.
Figure 4: A typical template for a Moore State Machine.