ECE 2300
Digital Logic & Computer Organization
Spring 2018

More FSMs
Timing, Clocking
Announcements

• Prelim 1 graded
  – High: 60 (out of 60); Mean: 46.2; Median 46.8
  – You can view your exam during instructor OH today, TA OH tomorrow, or lab sections next week

• HW4 to be posted tonight
FSM for Toggle Flip-Flop

- Input: T
- Output: Q
- Current state: S
- Next state: S*

Transition/output table:

<table>
<thead>
<tr>
<th>S</th>
<th>S*</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T = 0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>T = 1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Moore state diagram:

S* = T’S + TS’
Q = S
Review: Pattern Detector Moore FSM

- Moore transition/output table
Minimized Equations for $S^*$ and $Out$

$$S_1^* = S_0 \cdot In + S_1 \cdot In$$

$$S_0^* = S_0^* \cdot In + S_1 \cdot In$$

<table>
<thead>
<tr>
<th>$S_1 \ S_0$</th>
<th>$S_1^* \ S_0^*$</th>
<th>$Out$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$In = 0$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$In = 1$</td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td>01</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>00</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$Out = S_1 \cdot S_0$</td>
<td></td>
</tr>
</tbody>
</table>
# Review: Pattern Detector Mealy FSM

### Mealy transition/output table

<table>
<thead>
<tr>
<th>$S_1, S_0$</th>
<th>$S_1^<em>, S_0^</em>, \text{Out}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>In = 0</td>
</tr>
<tr>
<td>0 0</td>
<td>0 0, 0</td>
</tr>
<tr>
<td>0 1</td>
<td>0 0, 0</td>
</tr>
<tr>
<td>1 0</td>
<td>0 0, 0</td>
</tr>
</tbody>
</table>

- Mealy transition/output table
Minimized Equations for $S^*$ and Out

$$S_1^* = S_0 \cdot \text{In} + S_1 \cdot \text{In}$$

$$S_0^* = S_1' \cdot S_0' \cdot \text{In}$$

$$\text{Out} = S_1 \cdot \text{In}$$

<table>
<thead>
<tr>
<th>$S1$, $S0$</th>
<th>$S1^<em>$, $S0^</em>$, Out</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\text{In} = 0$</td>
</tr>
<tr>
<td>0 0</td>
<td>0 0, 0</td>
</tr>
<tr>
<td>0 1</td>
<td>0 0, 0</td>
</tr>
<tr>
<td>1 0</td>
<td>0 0, 0</td>
</tr>
</tbody>
</table>
Traffic Light Controller

- 4-way intersection with traffic lights
- Opposing lanes sequence together
  - 20 seconds dwell on green
  - 5 seconds dwell on yellow
  - 25 seconds dwell on red
Four Scenarios

- E & W Green / N & S Red for 20 seconds
- E & W Yellow / N & S Red for 5 seconds
- E & W Red / N & S Green for 20 seconds
- E & W Red / N & S Yellow for 5 seconds
Traffic Light Controller States

• 10 states
  – E & W Green / N & S Red1 for 5 seconds
  – E & W Green / N & S Red2 for 5 seconds
  – E & W Green / N & S Red3 for 5 seconds
  – E & W Green / N & S Red4 for 5 seconds
  – E & W Yellow / N & S Red for 5 seconds
  – E & W Red / N & S Green1 for 5 seconds
  – E & W Red / N & S Green2 for 5 seconds
  – E & W Red / N & S Green3 for 5 seconds
  – E & W Red / N & S Green4 for 5 seconds
  – E & W Red / N & S Yellow for 5 seconds

Clock period is 5 seconds
Traffic Light Controller FSM

State (EW/NS)
Outputs

- G/R1
- G/R2
- G/R3
- G/R4
- Y/R
- R/G1
- R/G2
- R/G3
- R/G4

Lecture 10: 11
Factoring FSMs

• Break FSM into multiple communicating FSMs
• Simplifies large FSMs
• May result in fewer states
Traffic Light Controller Using 2 FSMs

- **Light Controller (LC) FSM has 4 states**
  - G/R, Y/R, R/G, R/Y
- **Timer FSM controls when the LC FSM advances to the next state**
  - Keeps LC in *Green* states for 4 clock cycles

**Diagram**

- Timer
- Light Controller
- EW
- NS

- **Next**: tells LC FSM to advance to next state
- **Green**: indicates the green light is currently on
Light Controller FSM

- **State (EW/NS)**
  - Green

- **Timer**
  - Next
  - Green

- **LC**
  - Next
  - Green

Diagram:

1. **G/R**
   - Green = 1
   - Next = 1

2. **R/G**
   - Green = 1
   - Next = 1

3. **R/Y**
   - Green = 0
   - Next = 0

4. **Y/R**
   - Green = 0
   - Next = 0
**NG**: next Green period

**G**: last Green period; wait for yellow light (Green=0) before counting

<table>
<thead>
<tr>
<th>State</th>
<th>Next</th>
<th>Green</th>
</tr>
</thead>
<tbody>
<tr>
<td>NG</td>
<td>Next = 0</td>
<td>Green = 0</td>
</tr>
<tr>
<td>NG1</td>
<td>Next = 0</td>
<td>Green = 0</td>
</tr>
<tr>
<td>G</td>
<td>Next = 1</td>
<td>Green = 1</td>
</tr>
<tr>
<td>NG2</td>
<td>Next = 0</td>
<td>Green = 0</td>
</tr>
<tr>
<td>NG3</td>
<td>Next = 0</td>
<td>Green = 0</td>
</tr>
</tbody>
</table>

**Timer FSM**
Traffic Light Controller Operation

Clk

Next

Green
LC state

Timer state

East

West

Lights

North

South

Lights

Lecture 10: 16
Moore vs. Mealy

• **States**
  – Mealy machines tend to have fewer states
    • Different outputs on arcs (up to $n^2$) rather than states ($n$)

• **Timing**
  – Mealy machines react faster to inputs
    • Combinational path between input & output may potentially lower clock frequency
    • Need to avoid combinational loop (asynchronous feedback) when two FSMs are communicating
  – Moore machines are safer to use
    • Outputs change at clock edge (always one cycle later)
    • Potentially higher clock frequency
FSM Discussions: True or False?

• The next state logic in Moore FSM only depends on current state?

• The next state logic in Mealy FSM is combinational?

• The function of a D flip-flop can be described with a Mealy state diagram?
Propagation Delay ($t_{pd}$)

- Time for change in input to change the output
- Typically specified between 50% points

Circuits have **minimum** and **maximum** propagation delays

- Minimum sometimes called the *contamination delay* and maximum the *propagation delay*
Timing Diagram

- Shows how outputs respond to changes in inputs over time

![Timing Diagram Diagram]

- $t_p$ of AND gate
Glitch (Hazard)

- **Unplanned momentary switching of an output**

- **Types of glitches**
  - Static 1-hazard: Input change causes output to go from 1 to 0 to 1 (should have stayed 1)
  
  \[
  \begin{array}{c}
  1 \\
  0 \\
  1 \\
  \end{array}
  \]

  - Static 0-hazard: Input change causes output to go from 0 to 1 to 0 (should have stayed 0)
  
  \[
  \begin{array}{c}
  0 \\
  1 \\
  0 \\
  \end{array}
  \]

  - Dynamic hazards: Input change causes a change from 0 to 1 to 0 to 1 or from 1 to 0 to 1 to 0 (there should be just one change)
  
  \[
  \begin{array}{c}
  0 \\
  1 \\
  0 \\
  1 \\
  \end{array}
  \]
  \[
  \begin{array}{c}
  1 \\
  0 \\
  1 \\
  0 \\
  \end{array}
  \]
Glitch Example

- Glitches are typically caused by unequal signal propagation delays through the circuit.

Assume $X$ and $Y$ are 1
$S$ changes from 1 to 0

Output signal should stay at 1, but shows a transient 0 value.
Timing Diagram Showing Glitch

X
Y
S
S'
S•Y
S'•X
F

glitch
Do Glitches Matter?

Glitches are fine, as long as they settle before the rising clock edge.

CLOCK must be glitch-free.
Sequential Circuit Timing

Lecture 10: 25
Glitch on F: No Problem

X
Y
S
S'
S•Y
S'•X
F
CLOCK

X
S
S'•X
S•Y
F
CLOCK

CLOCK

Lecture 10: 26
Synchronous Circuits

- On the triggering clock edge (clock tick), the input of a flipflop is transferred to the output and held.

- Must ensure the output of the combinational logic has settled before the next clock tick.
Before Next Class

- H&H 1.4, 3.4-3.5.5

Next Time

Timing Analysis
Binary Arithmetic