Moore machine

a circuit in which the output depends only on the current state
Mealy machine

*a circuit in which the outputs depend on the inputs as well as the current state*
FSM timing characteristics

**MEALY**

- Input: ASYNC
- CL
- FFs: ASYNC
- Output: ASYNC

**MOORE**

- Input: $ASYNC$
- $!$
- "ASYNC
- SYNC
- Output: SYNC
Flip-flop timing requirements

- Flip-flops sample their inputs at each rising or falling clock edge

- The input data must be held stable for some time before and after the sample
A Mealy or Moore circuit?
An example Moore circuit

![Diagram of a Moore circuit](image)

### (a) Clock

<table>
<thead>
<tr>
<th>Present state</th>
<th>Inputs</th>
<th>Next state</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>A  X  Y  A  Z</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0  0  0  0  0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0  0  1  1  0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0  1  0  1  0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0  1  1  0  0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1  0  0  1  1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1  0  1  0  1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1  1  0  0  1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1  1  1  1  1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(b) State table
In class exercise

- Design a Mealy machine to identify when the sequence “3827” has occurred in a serial numerical input.

- Now design a Moore machine to do the same thing.
In class exercise: design a vending machine

• This vending machine will dispense a soda after the user has entered $.15

• Inputs: N, D (nickel, dime, quarter inserted)

• Output: R (release soda)
FSM design and implementation techniques

**Unused states:** extra state encodings (e.g., using 3 FFs to represent 6 states leaves 2 unused states) can be treated as “don’t care” values and used to simplify the combinational logic

   This reduces combinational logic, which means a faster clock.

**State minimization:** two states are equivalent if they transition to the same or equivalent states on the same inputs while producing the same outputs

   This can reduce the number of flip-flops.