Sequential Circuits (I)

What is a Sequential Circuits

- **Combinational circuits** are often referred to as “memoryless” circuits, since their output depends only on their current input and no history of prior inputs is related.
- The current output of a sequential circuit depends not only on the current input, but also on the current state of the circuit.
- Although we can use truth table to represent the current statue and next state based on the input of a sequential circuit. It’s less straightforward to represent the transition between states.
- So, we introduce State Machine to better represent the state transition of sequential circuits.

- A **state machine** is a graphic way of representing a time/history dependent process.
- It uses circles to represent states and edges to represent the transition from a state to another.
- A simple example of using state machine is to represent open and close a door.
  - If a door is currently open and you pull the door, it’s still open.
  - If a door is currently open and you push the door, it’s closed.
  - If a door is currently closed and you push the door, it’s still closed.
  - If a door is currently closed and you pull the door, it’s open.

- In this example, the door has two states: open and closed. The actions, pull and push, can change the state of the door depending on the door’s current state.
- We can draw a state machine for this example.

![State Machine Diagram](image)

- The difference between this example and the sequential circuits is that sequential circuits usually have outputs.
- Therefore, we need to introduce outputs into the state machines.
Moore Machine
- Moore machines let the output associate with the state.
- The input in Moore machines decides the next state.
- The output in Moore machines only depends on the state you transit into, not on how you got into the state (the input value).

Suppose there is a LED light on the top of the door to indicate the door is open or not in the dark. If the door is open, the LED is on. Otherwise, it’s off. We can update the above state machine to the following Moore machine.

**Example**
- Design a controller for an elevator.
- The elevator can be at one of the four floors: Ground, First, Second, and Third.
- There is a button that controls the elevator, and it has two possible values: up and down.
- The elevator goes up one floor every time you set the button to “up”, and goes down one floor every time you set the button to “down”.
- There are two lights in a row in the elevator that indicate the current floor. Both lights off (00) indicates the ground floor; The left light off and right light on (01) indicates the first floor; The right light off and left light on (10) indicates the second floor; Both lights on (11) indicates the third floor.
- Each time, the controller checks the current floor and current input before changing floors and lights.
- Draw a Moore machine diagram for the specification.