Digital Design and Computer Organization

Finite State Machines - 1

Reetinder Sidhu

Department of Computer Science and Engineering

Digital Design and Computer Organization

Finite State Machines - 1

Reetinder Sidhu

Department of Computer Science and Engineering

Finite State Machines - 1

Course Outline

- Digital Design
  - Combinational logic design
  - Sequential logic design
    - ★ **Finite State Machines - 1**
- Computer Organization
  - Architecture (microprocessor instruction set)
  - Microarchitecure (microprocessor operation)

Concepts covered

- Finite State Machines
  - Mealy and Moore Machines

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)

- The CL block contains **only combinational logic**

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)

- The CL block contains **only combinational logic**
- The state block contains **only memory elements**

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)

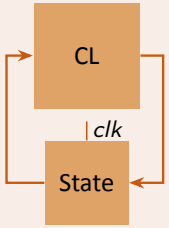
- The CL block contains **only combinational logic**
- The state block contains **only memory elements**
  - Memory elements always D flip-flops in this course

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)



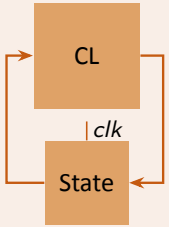
- The CL block contains **only combinational logic**
- The state block contains **only memory elements**
  - Memory elements always D flip-flops in this course
- Each arrow represents wires

Finite State Machines - 1

What is a Finite State Machine?

- The mathematical foundation for combinational logic circuits is Boolean functions. What about sequential logic circuits?

Finite State Machine (FSM)



- The CL block contains **only combinational logic**
- The state block contains **only memory elements**
  - Memory elements always D flip-flops in this course
- Each arrow represents wires

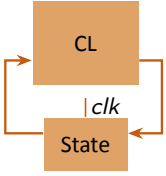
- Essentially **any sequential logic circuit** (from a simple counter to a complex microprocessor) can be represented as a Finite State Machine
- Fundamental concept in Computer Science and Engineering
  - Studied as Finite Automaton in Automata Theory course

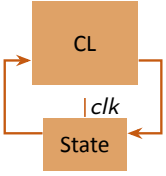
Finite State Machines - 1
Mealy and Moore FSMs

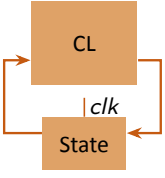
- What about input and output?

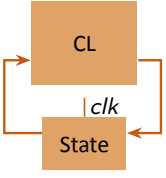
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities



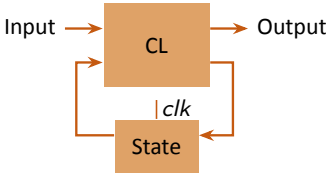


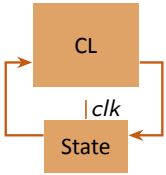


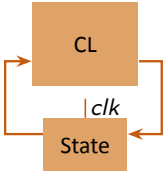


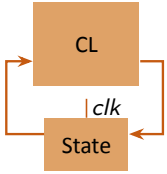
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities



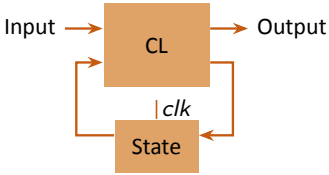


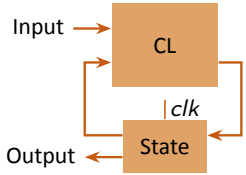


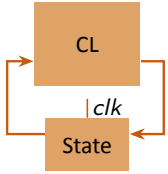


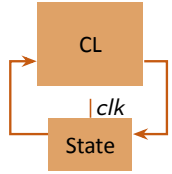
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities



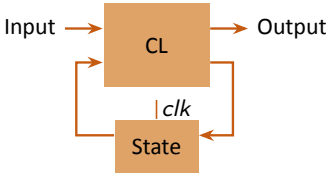


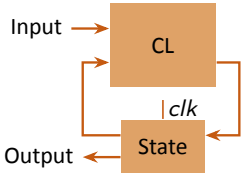


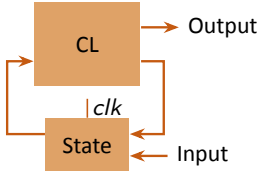


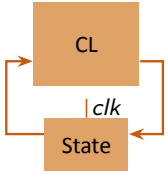
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities



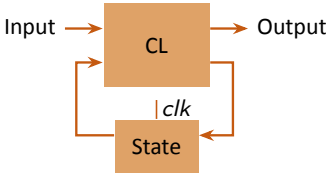


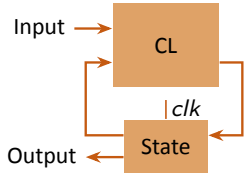


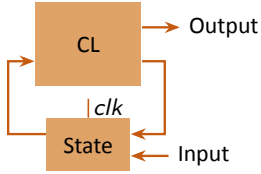


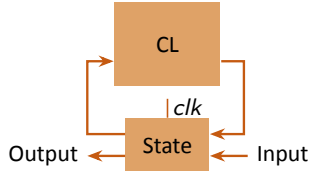
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities



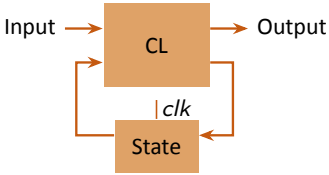


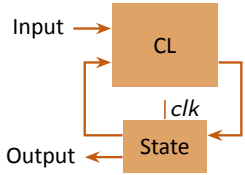


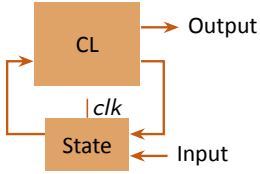


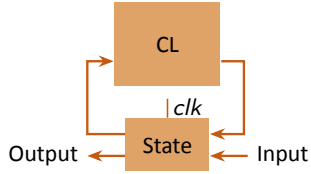
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities
  - Input to flip-flops not very useful



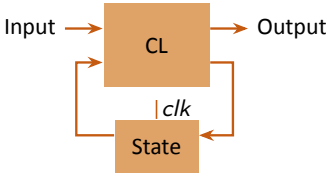


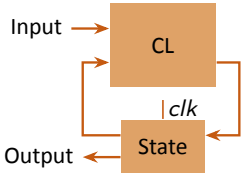




Finite State Machines - 1
Mealy and Moore FSMs

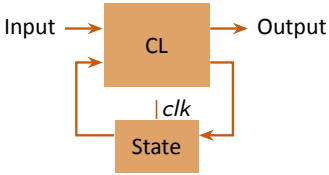
- What about input and output? Four possibilities
  - Input to flip-flops not very useful



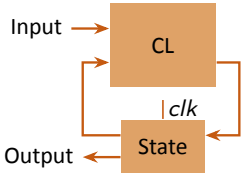


Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities
  - Input to flip-flops not very useful

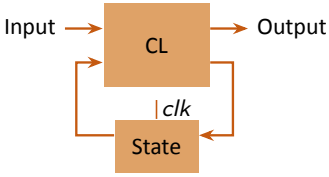


Core Mealy FSM

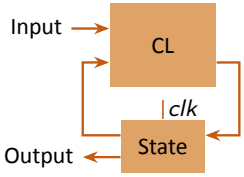


Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities
  - Input to flip-flops not very useful



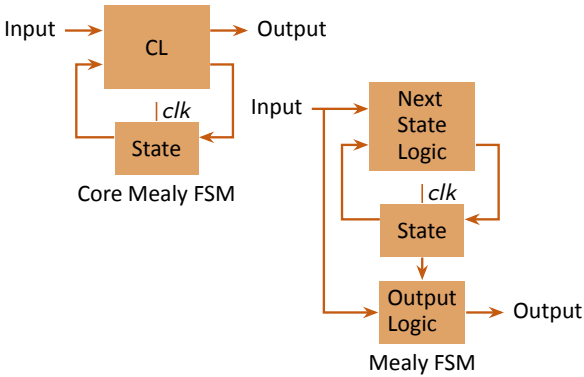
Core Mealy FSM



Core Moore FSM

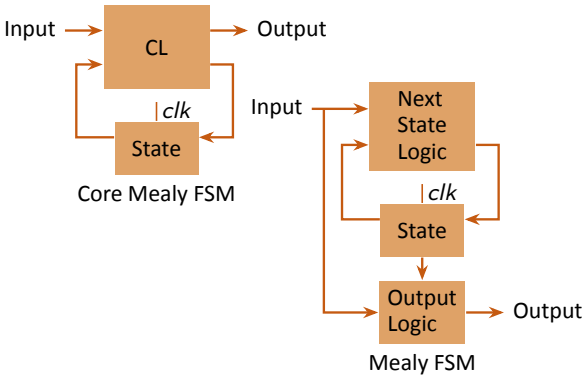
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities
  - Input to flip-flops not very useful



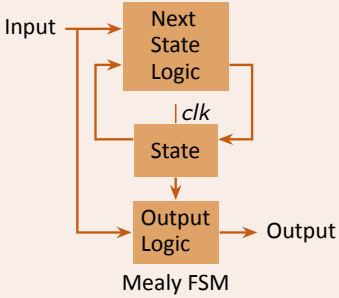
Finite State Machines - 1
Mealy and Moore FSMs

- What about input and output? Four possibilities
  - Input to flip-flops not very useful



Finite State Machines - 1
Mealy and Moore Finite State Machines

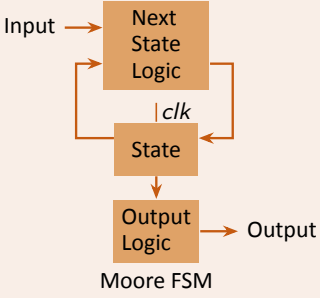
Mealy Machine



Mealy FSM

- Output depends on inputs and current state

Moore Machine



Moore FSM

- Output depends upon current state only

Finite State Machines - 1

Think About It

- Classify the following two logic circuits into Mealy machine, Moore machine or neither
  - Also partition the logic into next state, output and state blocks