



LUND  
UNIVERSITY

# Introduction to Structured VLSI Design

---

## Lab1 – State Machine Modeling

MOHAMMAD ATTARI    SEPTEMBER 4, 2019



LUND  
UNIVERSITY

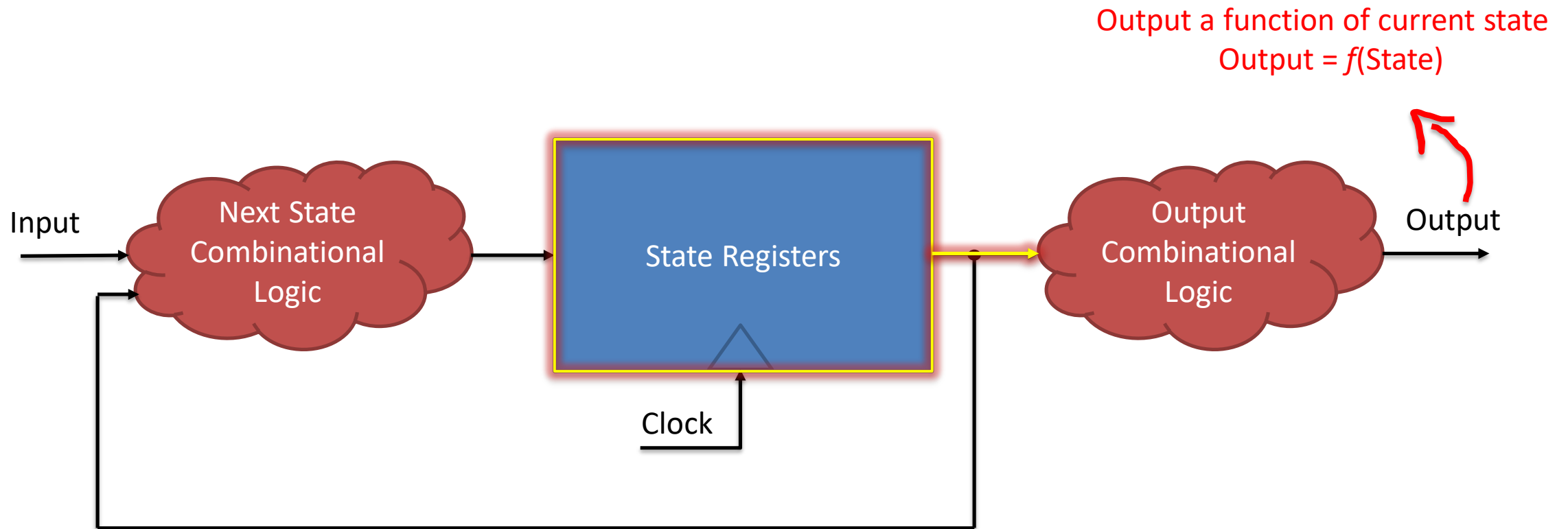
# Learning Outcomes

---

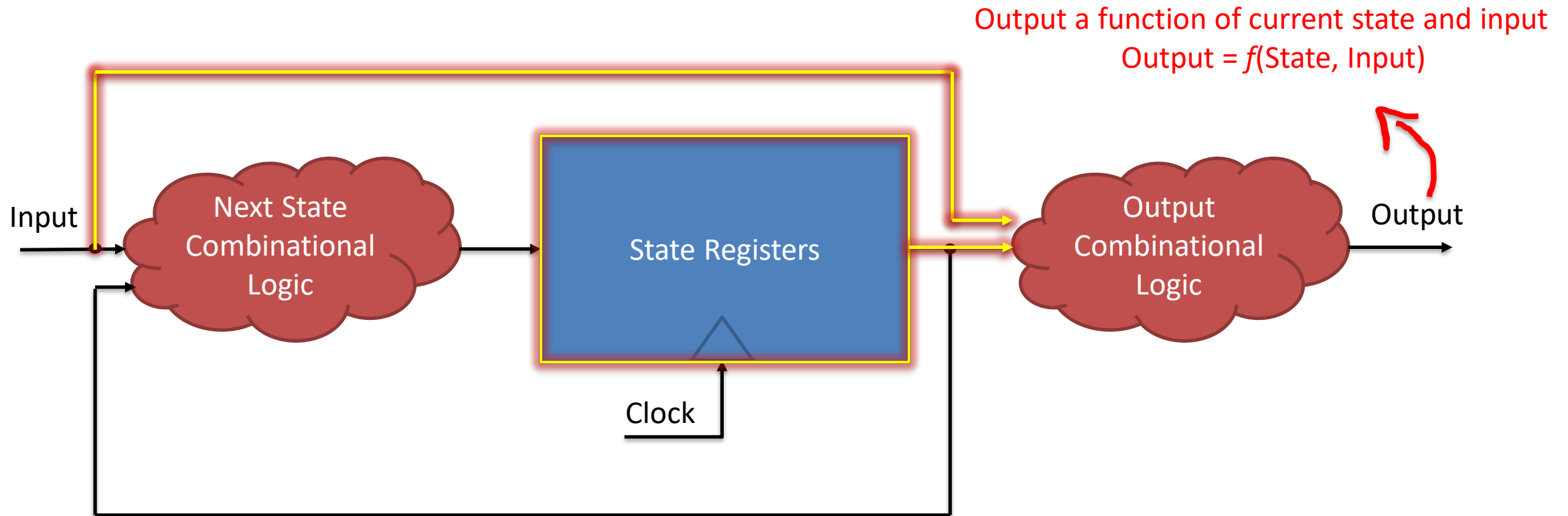
- VHDL modelling for synthesis
  - Registers (state)
  - Combinational logic
  - Latches
- State machines
  - Mealy
  - Moore
- Testbench
- Xilinx Vivado



# Moore Machine

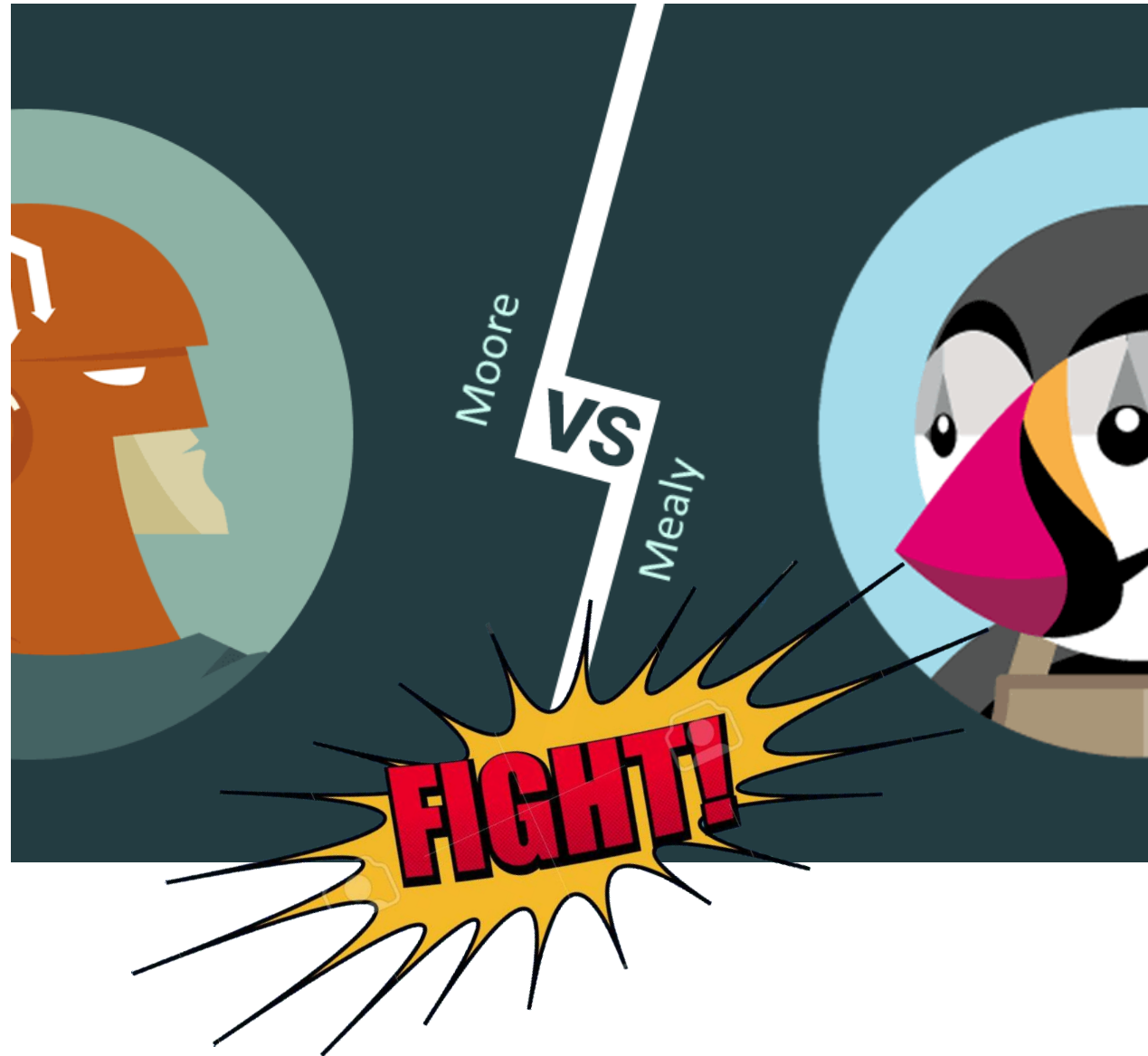


# Mealy Machine



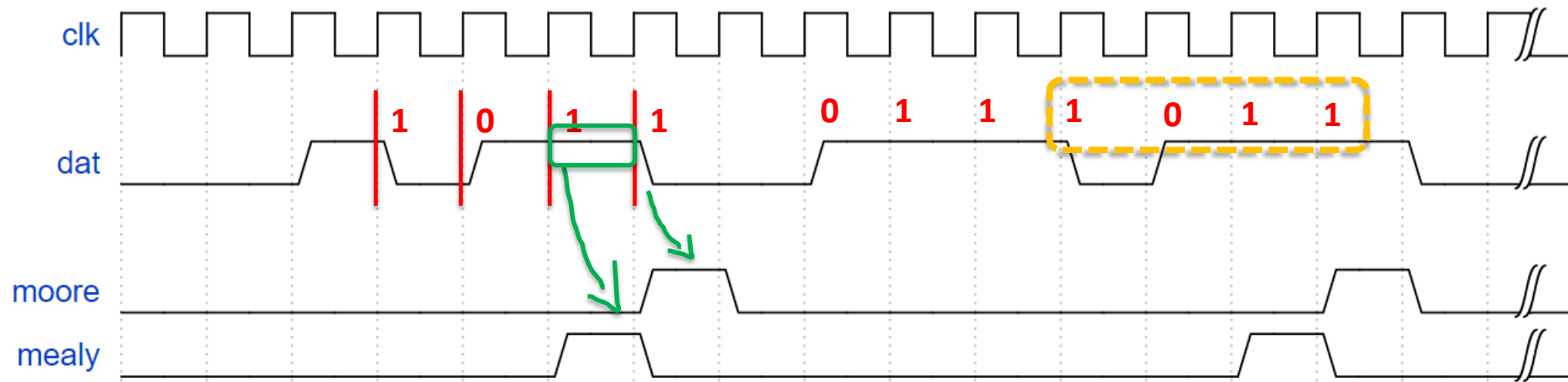
# Moore vs. Mealy

- Mealy
  - Fewer states
  - React faster
- Moore
  - Safer
  - Synchronous

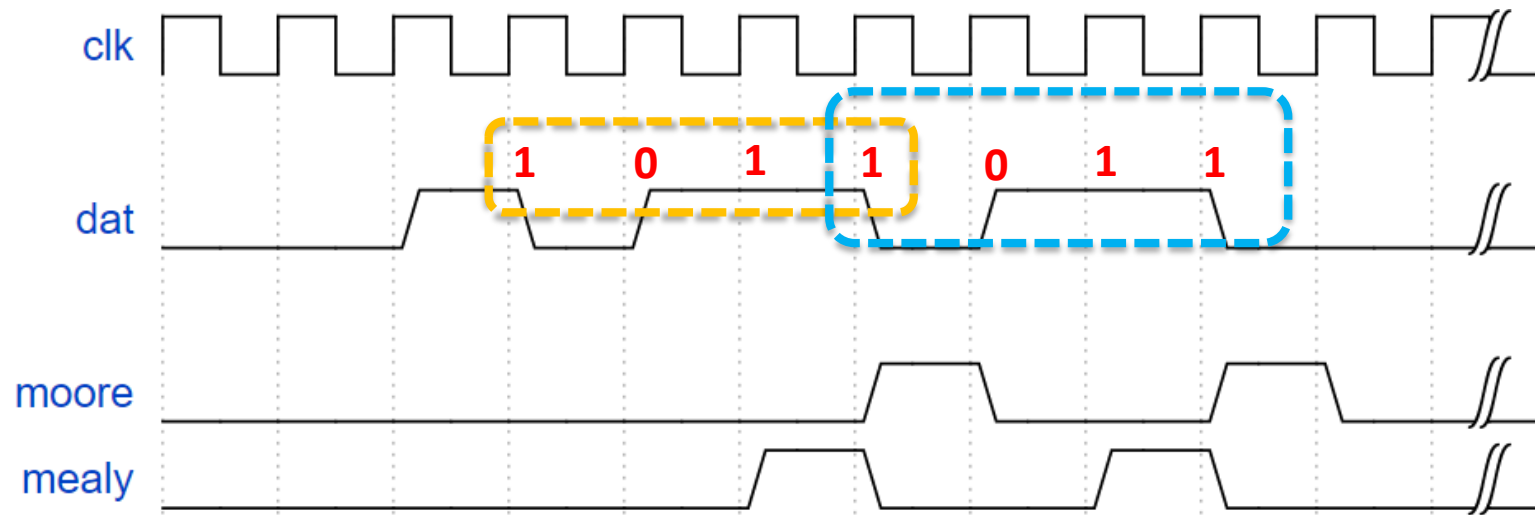


# Sequence Detector

- Detect the sequence **1011**



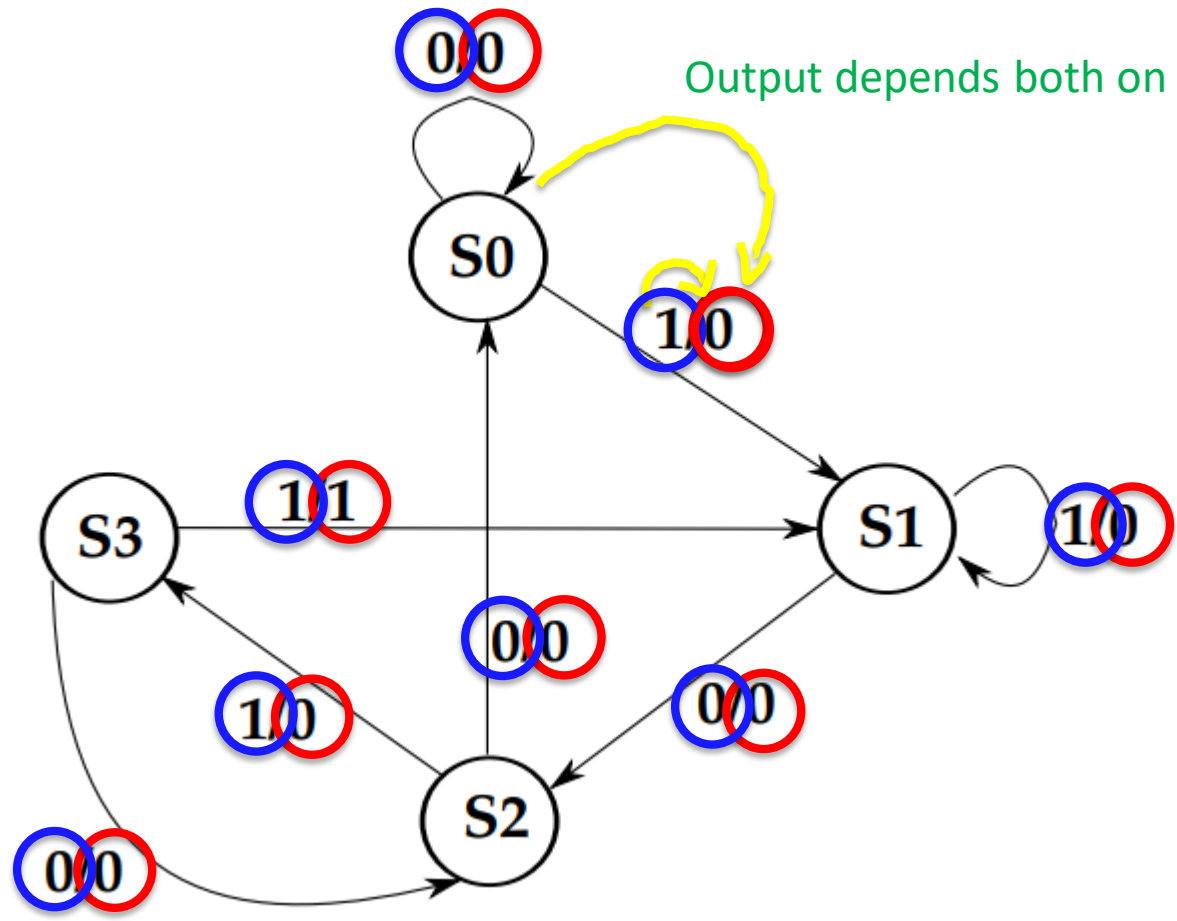
# Sequence in Sequence



# Mealy State Machine

input

output



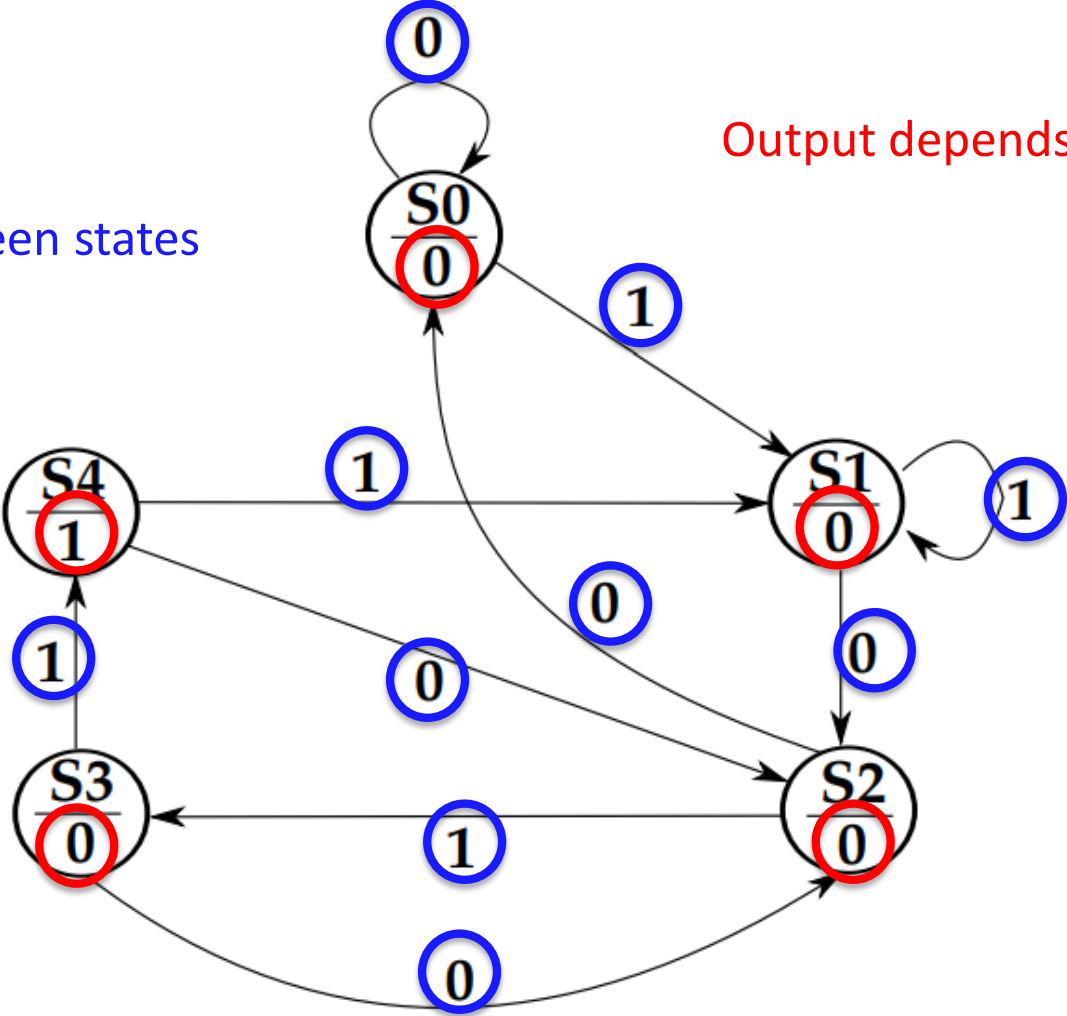
Output depends both on state and input



# Moore State Machine

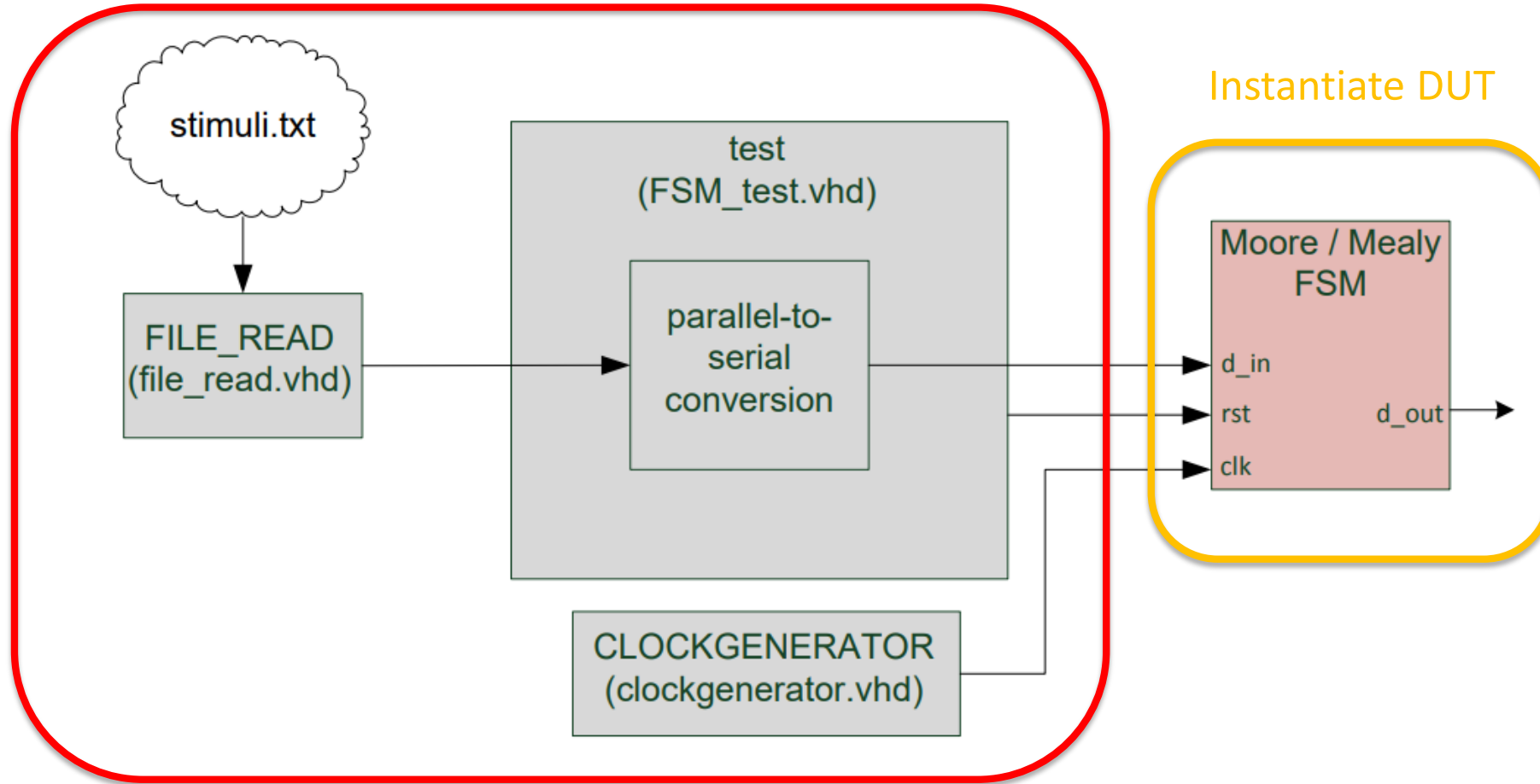
Input causes transition between states

Output depends only on state



# Testbench

All provided as templates



# Vivado Design Suite

The screenshot displays the Vivado Design Suite interface during a behavioral simulation. The main window shows a timing diagram for a functional simulation of a test bench. The simulation is titled "SIMULATION - Behavioral Simulation - Functional - sim\_1 - test".

**Project Manager (Left Panel):**

- PROJECT MANAGER
  - Settings
  - Add Sources
  - Language Templates
  - IP Catalog
  - IP INTEGRATOR
    - Create Block Design
    - Open Block Design
    - Generate Block Design
  - SIMULATION**
    - Run Simulation
  - RTL ANALYSIS
    - Open Elaborated Design
  - SYNTHESIS
    - Run Synthesis
    - Open Synthesized Design
  - IMPLEMENTATION
    - Run Implementation
    - Open Implemented Design
  - PROGRAM AND DEBUG
    - Generate Bitstream
    - Open Hardware Manager

**Scope (Left Panel):**

Name	Design Unit	Block Type
test	test(behavior)	VHDL En...
sequence_decoder(...)	sequence_decoder(...)	VHDL En...
FILE_READ(BEHAVE)	FILE_READ(BEHAVE)	VHDL En...
CLOCKGENERATOR(...)	CLOCKGENERATOR(...)	VHDL En...

**Objects (Left Panel):**

Name	Value	Date
s_in	0	Logi
clk	1	Logi
rst	1	Logi
d_out1	0	Logi
d_out2	U	Logi
flag	0	Logi
count[3:0]	3	Arra
inputbus[11:0]	56c	Arra
inputbus_temp[...]	5b0	Arra
M1[11:0]	b60	Arra

**Timing Diagram (Center):**

The timing diagram shows the simulation results for various signals over time. The time axis ranges from 130 ns to 190 ns. A vertical yellow line marks the time 162.227400 ns. The signals shown are:

- s\_in: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- clk: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- rst: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- d\_out1: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- d\_out2: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- flag: Logic signal, high at 130 ns, low at 135 ns, high at 140 ns, low at 145 ns, high at 150 ns, low at 155 ns, high at 160 ns, low at 165 ns, high at 170 ns, low at 175 ns, high at 180 ns, low at 185 ns, high at 190 ns.
- count[3:0]: Counter signal, values: 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, 0, 1, 2.
- inputbus[11:0]: Bus signal, values: 5e5, bcb, 797, 22f, e5f, cbf, 97f, 2ff, 5ff, bff, 7ff, fff, e45, e45.
- inputbus\_temp[11:0]: Bus signal, values: e5f, bcb, 797, 22f, e5f, cbf, 97f, 2ff, 5ff, bff, 7ff, fff, e45, e45.
- M1[11:0]: Bus signal, values: bcb, 797, 22f, e5f, cbf, 97f, 2ff, 5ff, bff, 7ff, fff, e45, c6b.

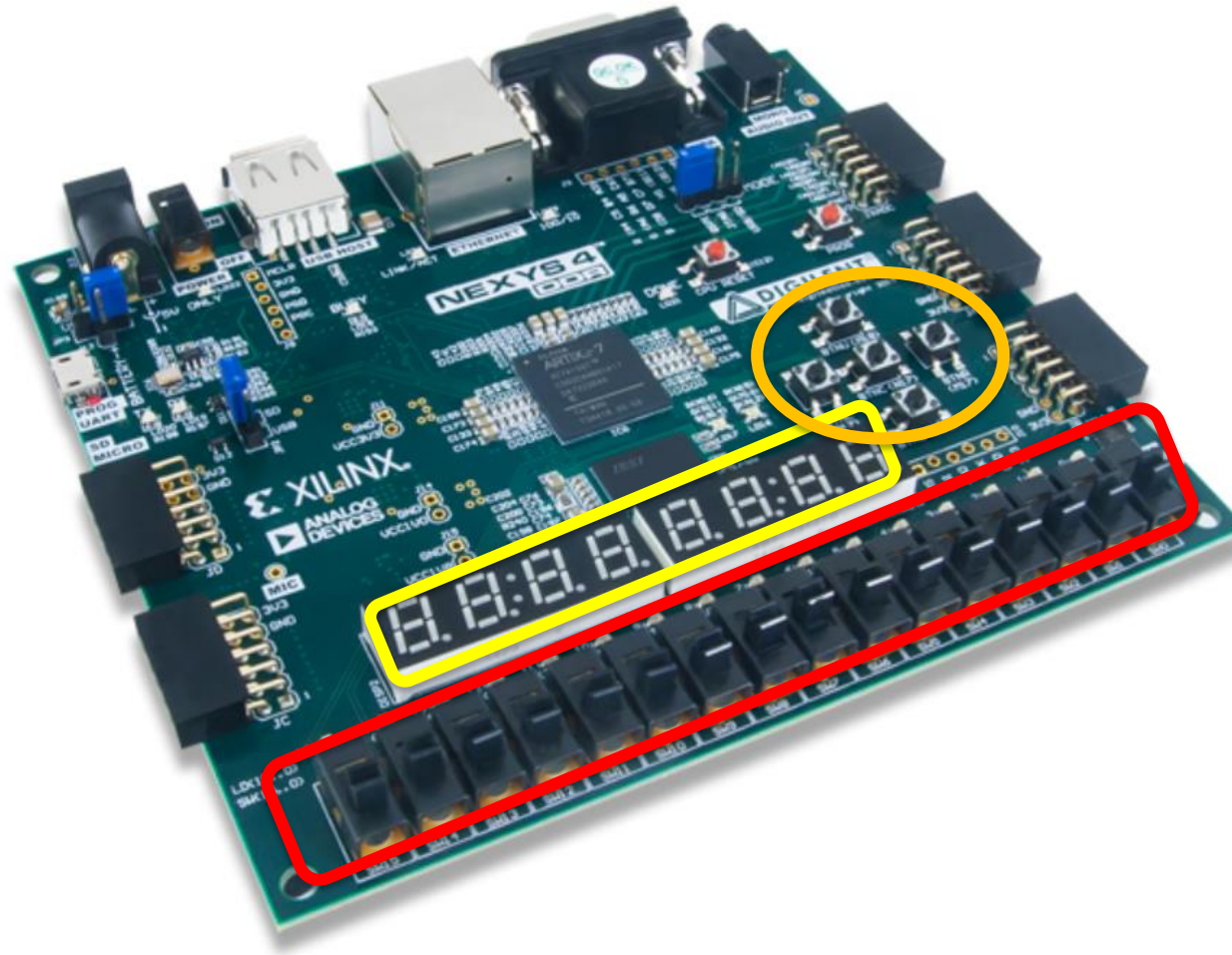
**Tcl Console (Bottom):**

```
run 20000 ns
run 20000 ns
run 20000 ns
run 20000 ns
```



# Nexys 4 FPGA (Labs 2-5)

---



# Lab1 Assignment

---

- Draw the state diagrams
  - Cross check
- Implement FSM in VHDL
- Show simulation
- Demonstrate understanding

