Welcome to The Logic Design Lab!

Fall 2020

Lab 4: Finite State Machines

Prof. Chun-Yi Lee

Department of Computer Science

National Tsing Hua University

Agenda

- Lab 4 Outline
- Lab 4 Basic Questions
- Lab 4 Advanced Questions

Lab 4 Outline

- Basic questions (1.5%)
  - Individual assignment
  - Due on 10/29/2020. In class.
  - Please submit your codes to ILMS after the demonstration.

- Advanced questions (5%)
  - Group assignment
  - Demonstration on your FPGA board (In class)
    - Submit your FPGA codes to ILMS by 11/12/2020. 15:00:00.
  - Assignment submission (Submit to ILMS)
    - ILMS submission due on 11/12/2020, 23:59:59.
    - Source codes and testbenches
    - Lab report in PDF

Lab 4 Rules

- Please note that grading will be based on **NCVerilog**
- You can use **ANY** modeling techniques

- If not specifically mentioned, we assume the following SPEC
  - **CLK** is **positive edge triggered**
  - Synchronously reset the Flip-Flops when **RESET == 1'b0**

Lab 4 Submission Requirements

- Source codes and testbenches
  - Please follow the templates EXACTLY
  - We will test your codes by TAs' testbenches
- Lab 4 report
  - Please submit your report in a single PDF file
  - Please draw the **block diagrams** and **state transition diagrams** of your designs
  - Please explain your designs in detail
  - Please list the contributions of each team member clearly
  - Please explain how you test your design
  - What you have learned from Lab 4

Agenda

- Lab 4 Outline
- Lab 4 Basic Questions
- Lab 4 Advanced Questions

Basic Questions

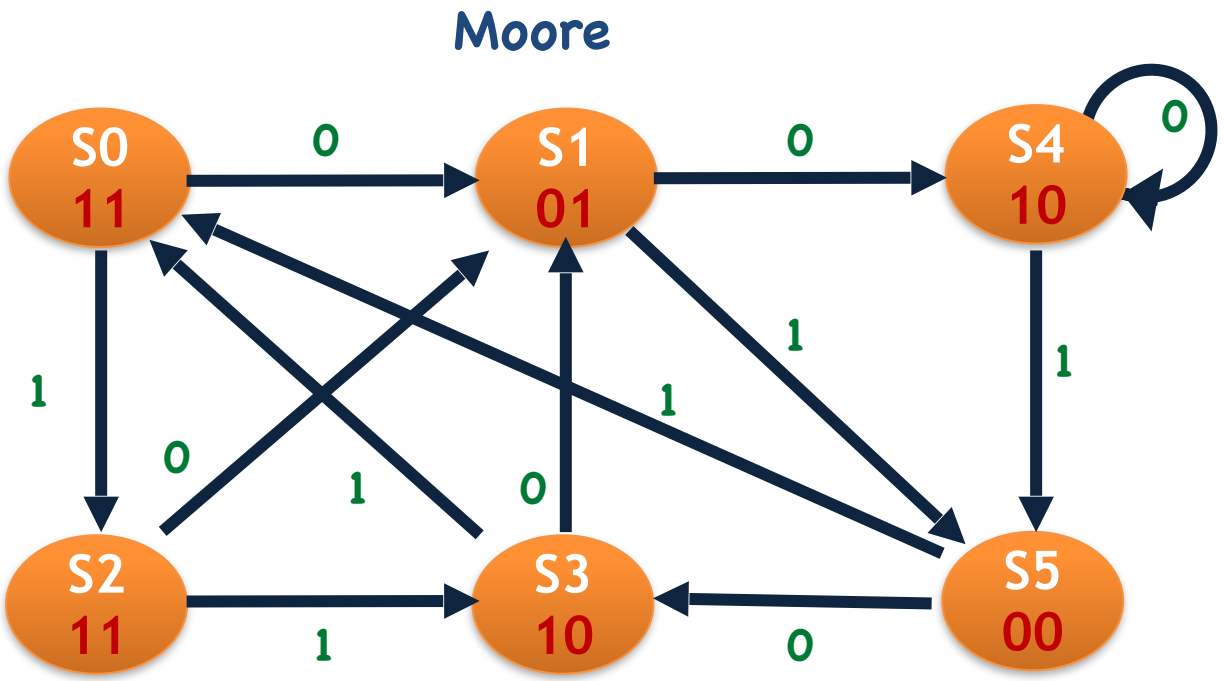
- Individual assignment

- Verilog questions (due on 10/29/2020. In class.)
  - Moore machine
  - Mealy machine

- Demonstrate your work by waveforms
- Please upload your codes to ILMS after demonstration
  - Otherwise, you will not have the scores for the basic questions

Verilog Question 1

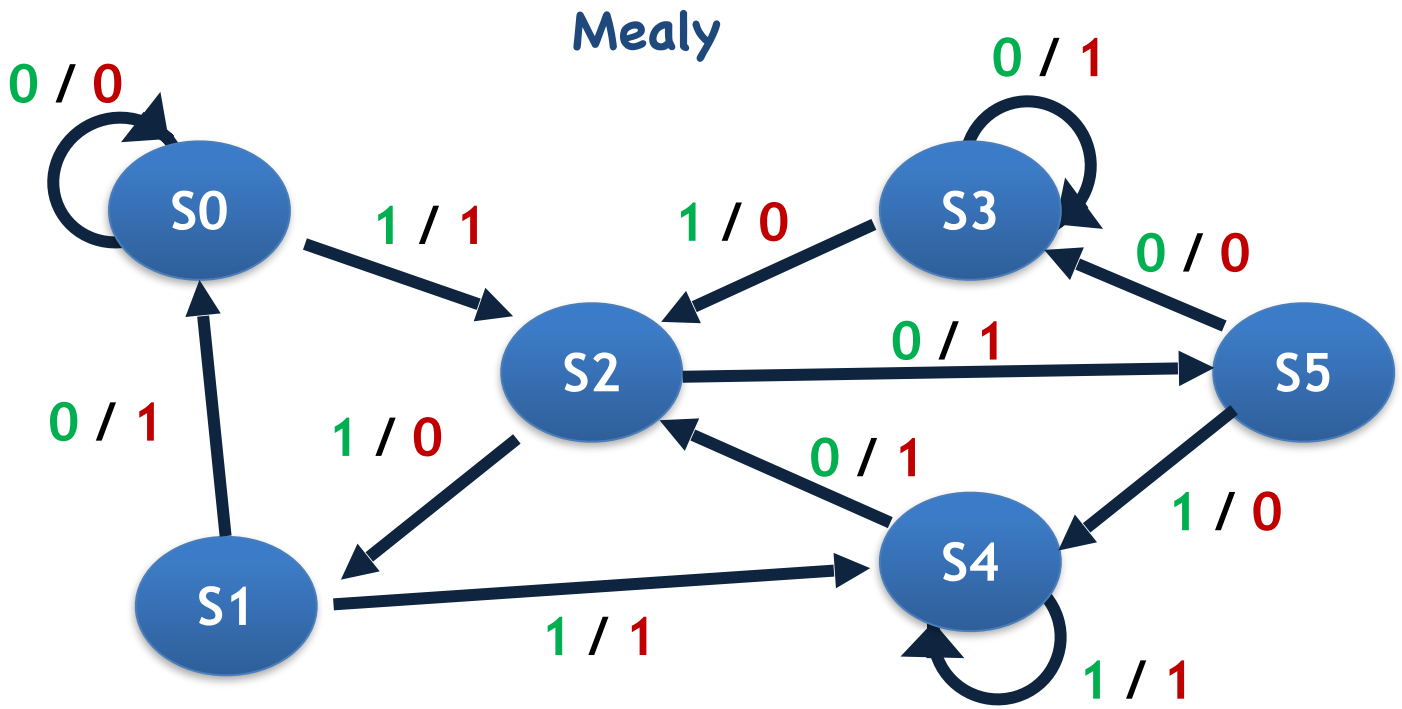
- Moore machine
  - **Green** represents input, while **red** represents output
  - Output your **current state** as well
  - When RESET == 1'b0, State = S0



S0: 3'b000 S1: 3'b001 S2: 3'b010 S3: 3'b011 S4: 3'b100 S5: 3'b101

Verilog Question 2

- Mealy machine
  - **Green** represents input, while **red** represents output
  - Output your **current state** as well
  - When RESET == 1'b0, State = S0



S0: 3'b000 S1: 3'b001 S2: 3'b010 S3: 3'b011 S4: 3'b100 S5: 3'b101

Agenda

- Lab 4 Outline
- Lab 4 Basic Questions
- Lab 4 Advanced Questions

Advanced Questions

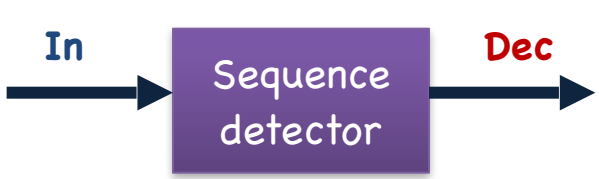
- Group assignment

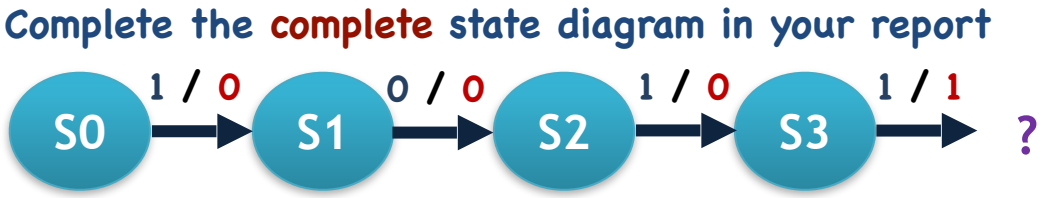
- Verilog questions
  - Source codes and the report due on 11/12/2020. 23:59:59.
  - Mealy machine sequence detector
  - Greatest common divisor

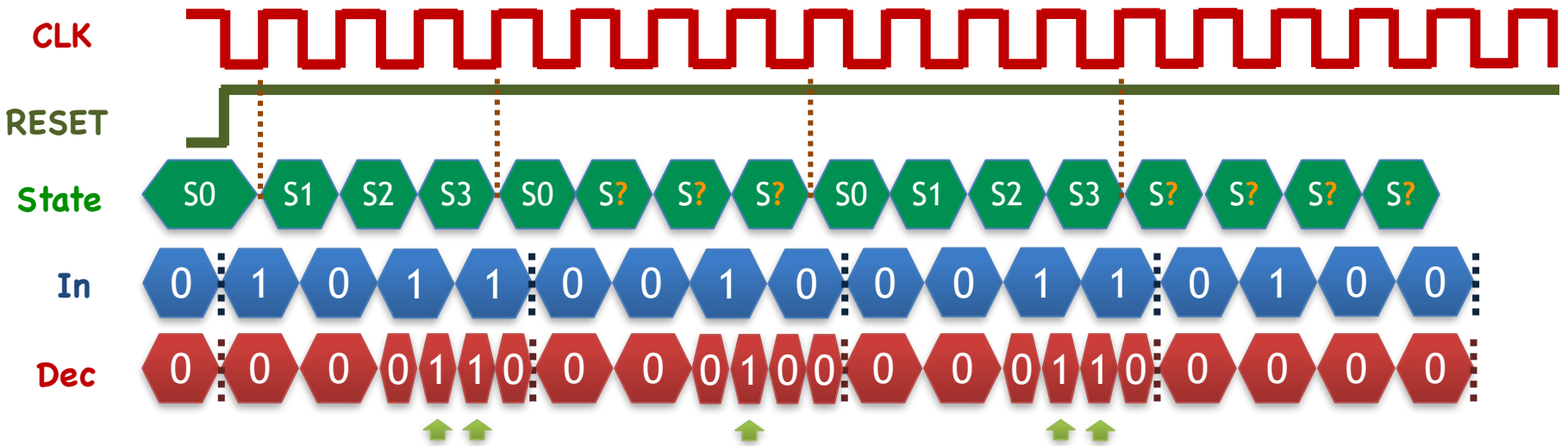
- FPGA demonstration (due on 11/12/2020. In class.)
  - Stopwatch
  - Please submit your source codes by 11/12/2020. 15:00:00.
  - We will only grade your demonstration based on your codes downloaded from ILMS.

Verilog Question 1

- **Mealy machine sequence detector**
  - 1-bit input **In** and 1-bit output **Dec**
  - When the four bit sequence is **1011**, **0011**, or **1010**, **Dec** is set to **1**
  - Re-detect the sequence **every four bits**
  - Please draw your state diagram in your report

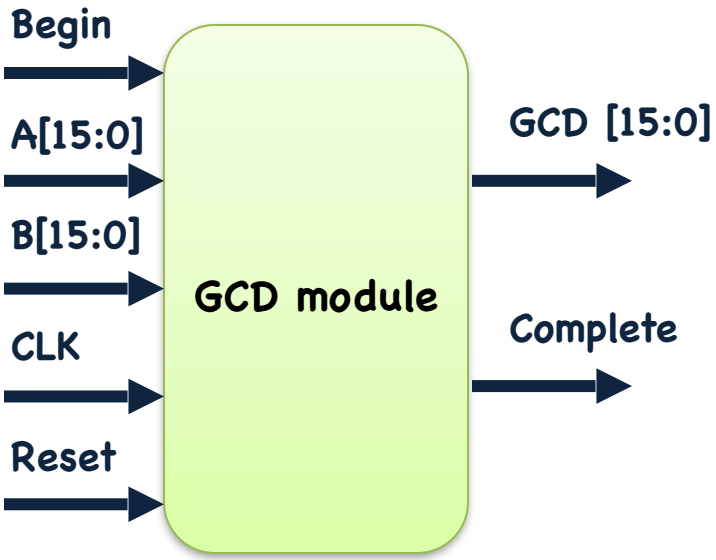






Verilog Question 2

- **Greatest common divisor**
- Calculate the greatest common divisor of two numbers **A** and **B**
- Block diagram and pseudo code are as follows
  - You **shall not** use **loop statements and modulus (%)** in your Verilog codes

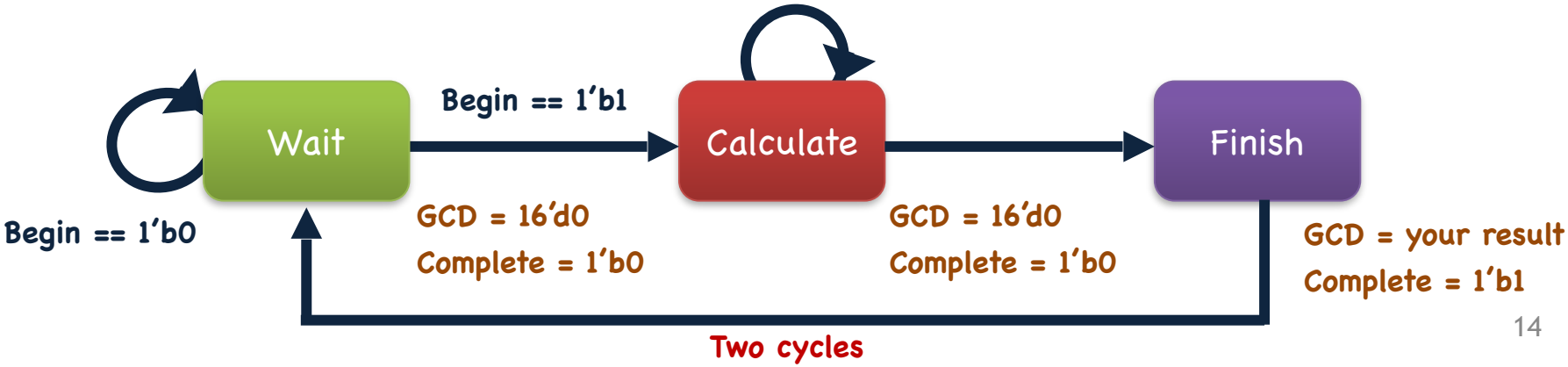


GCD pseudo code

```
Function gcd (a, b)
begin
    if (a == 0)
        return b;
    while (b != 0)
    // Do the following operation once per clock cycle
    begin
        if (a > b)
            a = a - b;
        else
            b = b - a;
    end
    return a;
end
```

Verilog Question 2 (Cont'd)

- Three states are used: **Wait**, **Calculate**, and **Finish**
- **Wait state**
  - Wait for Begin == 1'b1 (**one cycle**) to begin the operation (**and fetch the inputs**)
  - When Reset == 1'b0, reset the module to the **Wait state**
- **Calculate state**
  - Calculate the **subtraction operations once per cycle**
- **Finish state**
  - Output the GCD result for **two cycles**
  - Complete == 1'b1 for **two cycles**



Advanced Questions

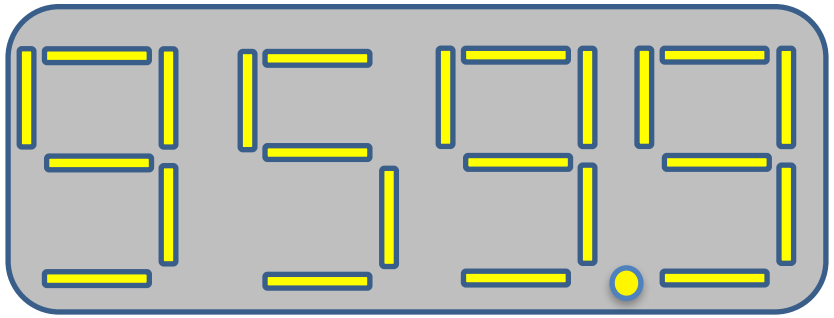
- Group assignment

- Verilog questions
  - Source codes and the report due on 11/12/2020. 23:59:59.
  - Mealy machine sequence detector
  - Greatest common divisor

- FPGA demonstration (due on 11/12/2020. In class.)
  - Stopwatch
  - Please submit your source codes by 11/12/2020. 15:00:00.
  - We will only grade your demonstration based on your codes downloaded from ILMS.

FPGA Demonstration

- Please design a stopwatch and display it on your 7-segment display
- The four digits represent:
  - Digit[3]: Minutes
  - Digit[2:1]: Seconds
  - Digit[0]: 0.1 Seconds
  - Dot: Separates the third and fourth digits
- Use a push-bottom (**LEFT button**) with debounce to start counting up
  - If the **LEFT button** is pushed, the stopwatch stops.
  - If the **LEFT button** is pushed again, the stopwatch resumes counting up.
- Use another push-button (**UP button**) with debounce to reset
- **When 9:59.9 is reached, the stopwatch is returned to 0** and **wait**.
- Three FSM states: **RESET, WAIT, COUNT**
  - **Please draw a state transition diagram in your report**



*The first Starbucks at Seattle, Washington, USA
This picture is taken by Chun-Yi Lee himself, who is also a fan of photography