



Welcome to The Hardware Lab!

Fall 2018

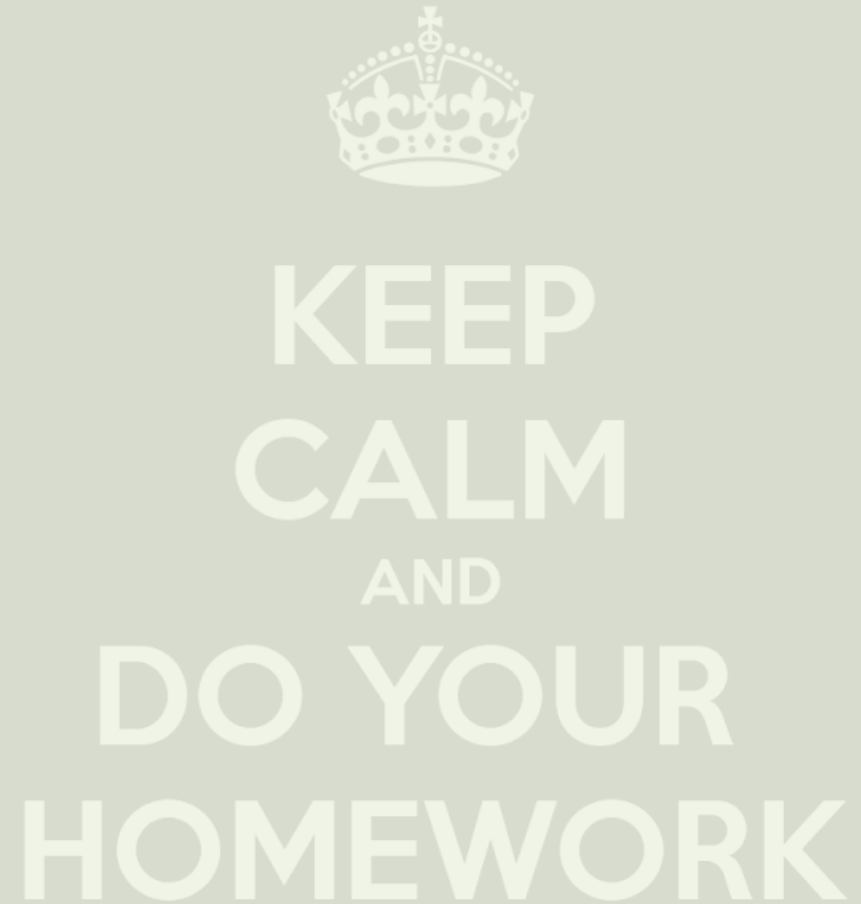
Lab 4: Finite State Machines

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 **11/1/2018. In class.**
 - Only demonstration is necessary. Nothing to submit.
- Advanced questions (5%)
 - Group assignment
 - ILMS submission due on **11/8/2018. 23:59:59.**
 - Demonstration on your FPGA board (**In class**)
 - Assignment submission (**Submit to ILMS**)
 - Source codes and testbenches
 - Lab report in PDF

Lab 4 Rules

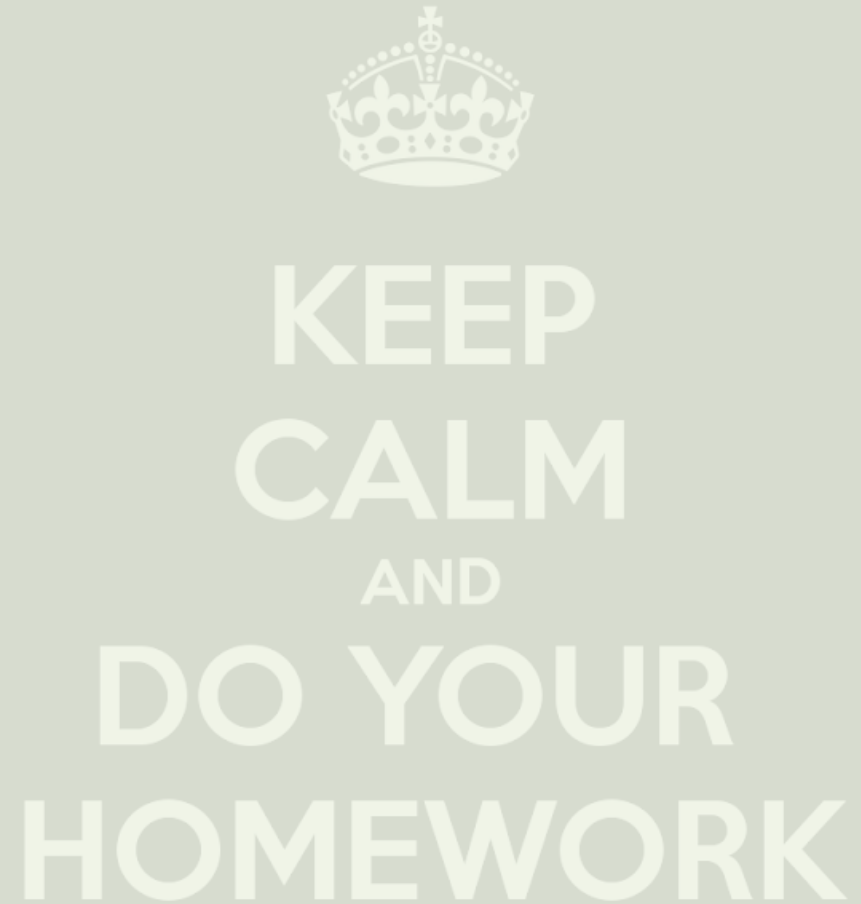
- You can use **ANY** modeling techniques
- If not specifically mentioned, we assume the following SPEC
 - **CLK** is **positive edge triggered**
 - Synchronously reset the Flip-Flops when **RESET == 1'b0**

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

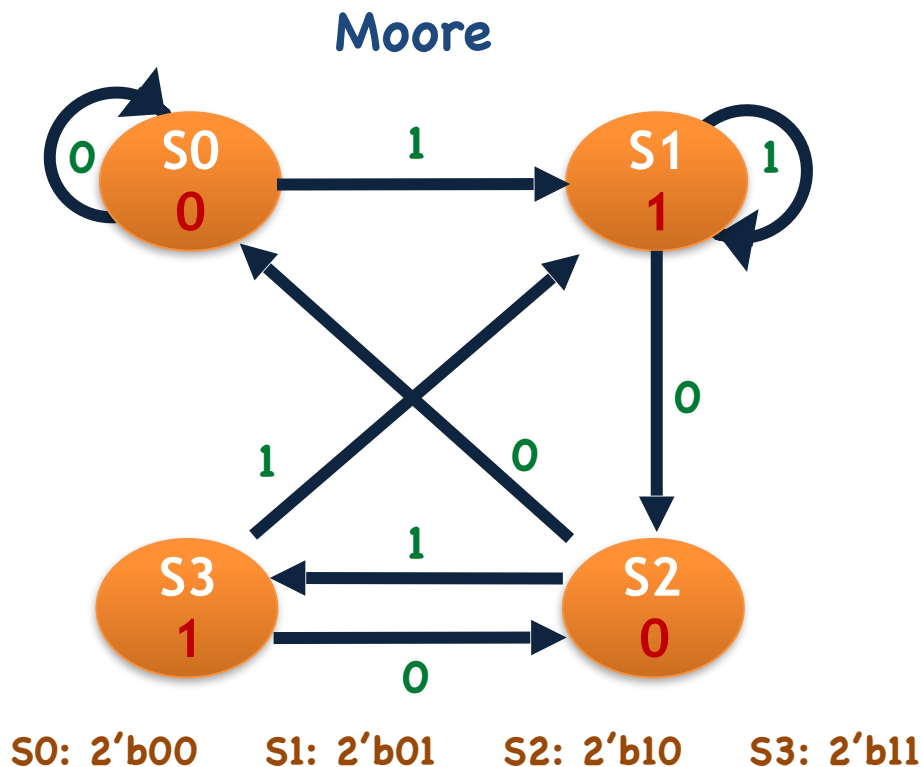


Basic Questions

- Individual assignment
- Verilog questions (due on 11/1/2018. In class.)
 - Moore machine
 - Mealy machine
- Demonstrate your work by waveforms

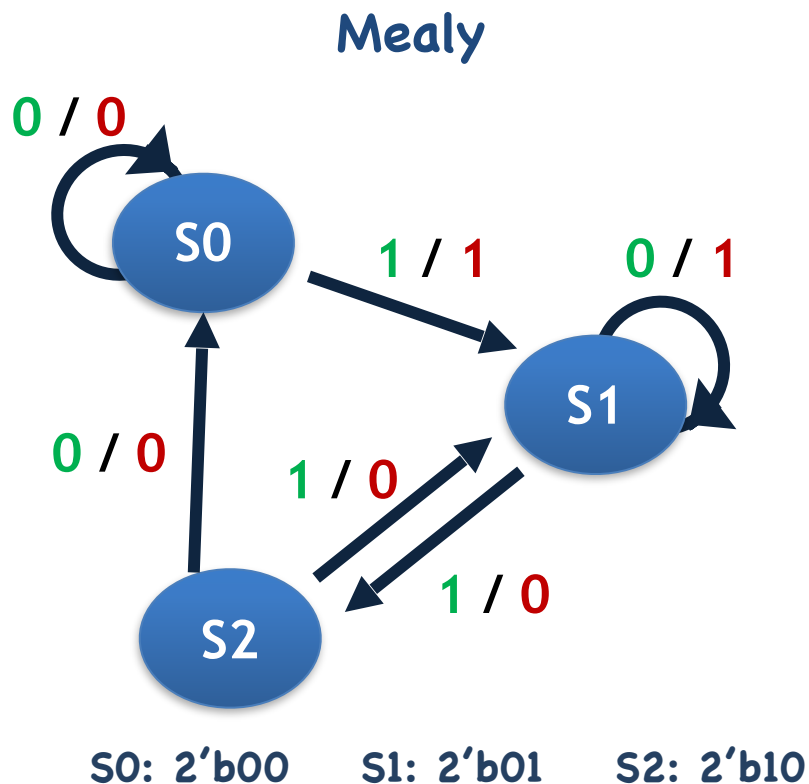
Verilog Question 1

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



Verilog Question 2

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



Agenda

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

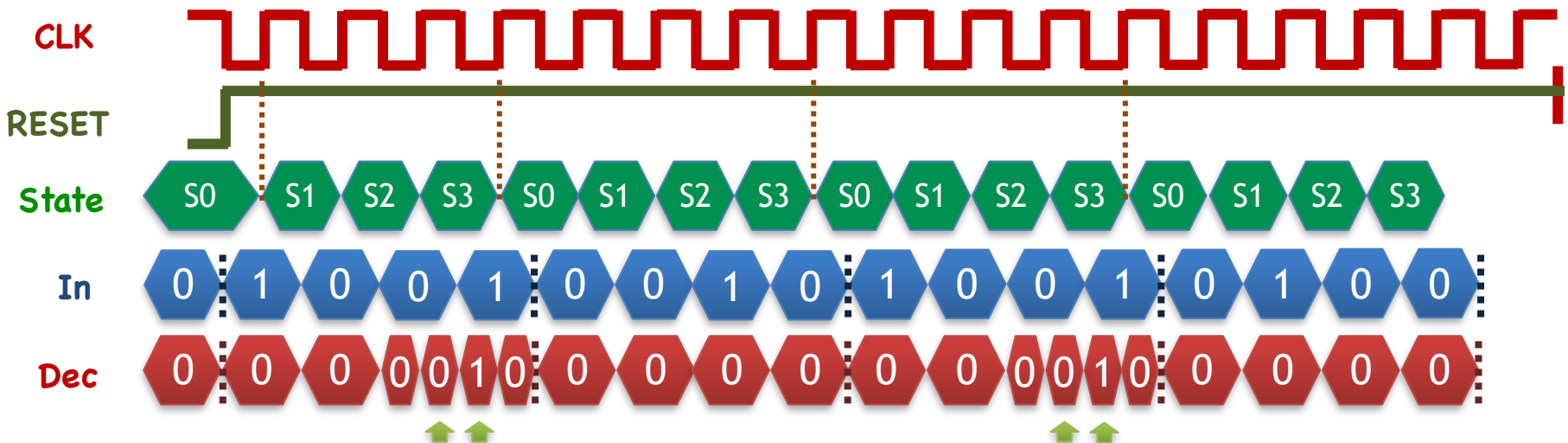
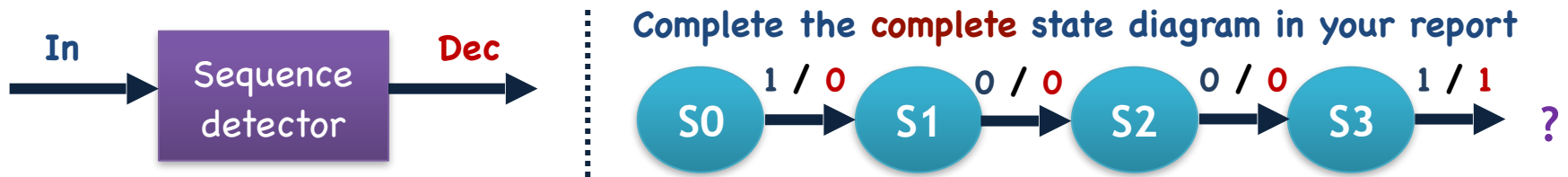


Advanced Questions

- Group assignment
- Verilog questions (due on 11/8/2018. 23:59:59.)
 - Mealy machine sequence detector
 - Greatest common divisor
- FPGA demonstration (due on 11/8/2018. In class.)
 - Stopwatch

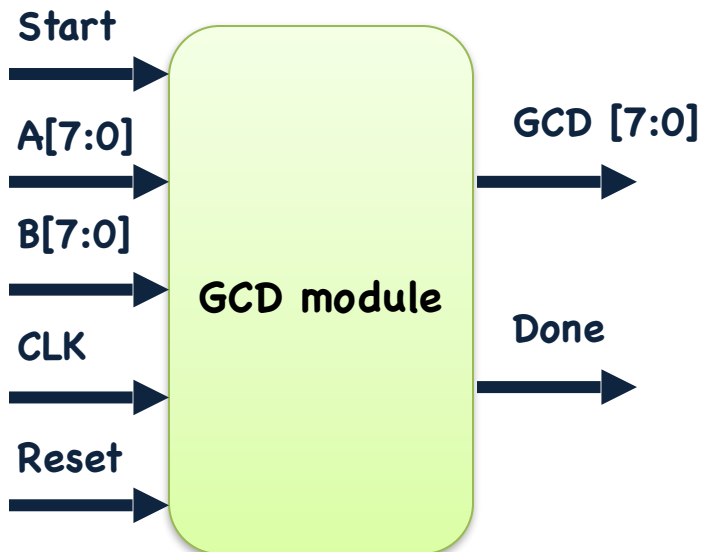
Verilog Question 1

- **Mealy machine sequence detector**
 - 1-bit input **In** and 1-bit output **Dec**
 - When the four bit sequence is either **1001**, **Dec** is set to 1
 - Redetect 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 follow
 - You **shall not** use **while statements and modulus (%)** in your Verilog codes



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

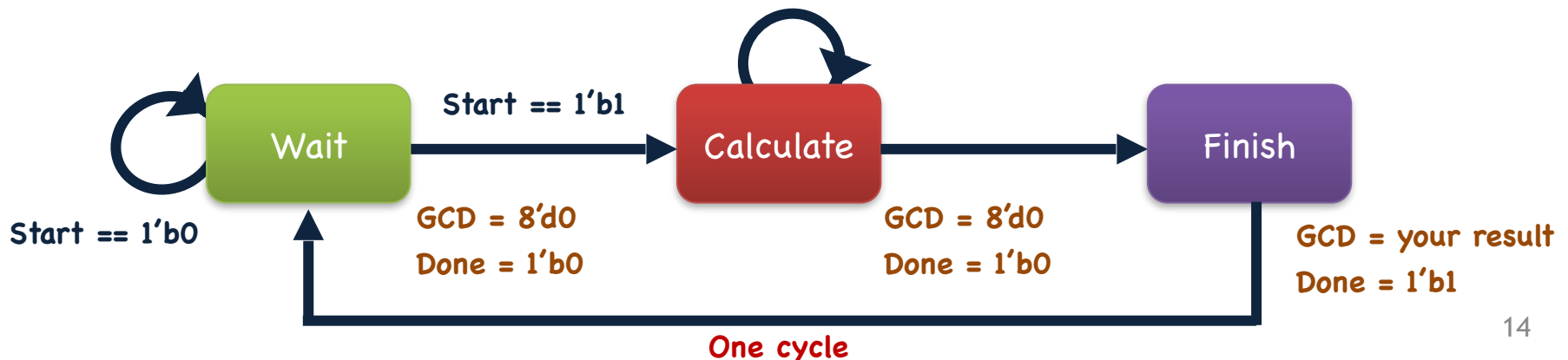
return a;

end

**GCD pseudo
code**

Verilog Question 2 (Cont'd)

- Three states are used: **Wait**, **Calculate**, and **Finish**
- **Wait state**
 - Wait for $\text{Start} == 1'b1$ (**one cycle**) to begin the operation
 - When $\text{Reset} == 1'b0$, reset the module to the **Wait state**
- **Calculate state**
 - Calculate the **subtraction operations once per cycle**
- **Finish state**
 - Output the GCD result for **one cycle**
 - $\text{Done} == 1'b1$ for **one cycle**

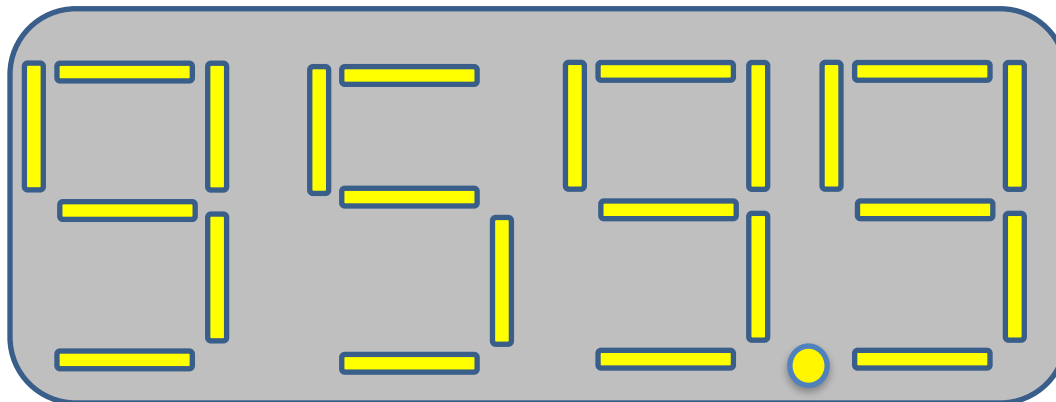


Advanced Questions

- Group assignment
- Verilog questions (due on 11/8/2018. 23:59:59.)
 - Mealy machine sequence detector
 - Greatest common divisor
- FPGA demonstration (due on 11/8/2018. In class.)
 - Stopwatch

FPGA Demonstration 1

- 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-button (**UP button, T18**) with debounce to start counting up
- Use another push-button (**RIGHT button, T17**) with debounce to reset
- Three FSM states: **RESET, WAIT, COUNT**
 - Please draw a state transition diagram in your report



Thank you for your attention!



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