Welcome to The Logic Design Lab!

Fall 2021

Moore and Mealy Conversion

Prof. Chun-Yi Lee

Department of Computer Science

National Tsing Hua University

Agenda

- Announcement
- Moore and Mealy conversion
  - Moore to Mealy conversion
  - Mealy to Moore conversion
- Common Questions
- Datapath Components

Today's class will help you:

1. Understand that Moore and Mealy machines may interchangeable
2. Understand how to convert a Moore machine to a Mealy machine
3. Understand how to convert a Mealy machine to Moore machine

Announcements

- Lab 4
  - Lab 4 advanced questions due on **11/11/2021 (Thu)**
  - Lab 4 demonstration on **11/18/2021 (Thu) (FPGA)**
  - **Please read the lab description and specification carefully**
  - **Compile your codes again before merging yours with your teammates**
  - **Please follow the template I/Os and submit your .v files**

- Final project
  - You can begin working on your final projects from now
  - Please avoid copying the works from past years

Final Project

- Use your FPGA to implement creative and interesting works

- Final project proposal due on **12/13/2021 (Mon)**
- Final project report due on **1/14/2022 (Fri)**

- Final exam
  - **12/9/2021 (Thu)**
  - 2 hours

Final Project Presentation

- Final project demonstration on **1/13/2022 (Thu)** and **1/14/2022 (Fri)**
  - Around 10 minutes per team

- **No restriction on your presentation style**
  - Be creative!
  - What is special and new in your project
  - Key features

- Order of presentation
  - We will randomly decide the order
  - Let us know if you have constraints

Final Project Report

- Final project report
  - Due on **1/14/2022, 23:59pm (Fri)**
  - Block diagrams and state transition diagrams
  - **Detailed explanation**

- Report contents
  - Introduction
  - Motivation
  - System specification
  - Experimental results
  - Conclusion
  - …And any other sections that you would like to include

Final Project Award

- **Category:** Difficulty and completeness
  - Graded by me and the TAs
    - Difficulty (**35%**)
    - Completeness (**30%**)
    - Source code coding style (**15%**) (Correctness, usage, comments, etc.)
    - Report (**20%**)
  - First place: **NTD $6,000**
  - Second place: **NTD $3,000**
  - Third place: **NTD $1,500**

- **Category:** Best creativity
  - Voted by everyone in the class
  - **NTD $1,000**

- **Category:** Deep learning models in FPGA
  - Extra **bonus points** and **cash rewards**. To be announced

Final Exam

- Dates and time
  - **12/9/2021 (Thu)**
  - **3:30pm ~ 5:30pm (2 hours)**

- Course contents
  - Verilog design questions
  - Logic design concepts
  - Lecture & lab contents

Agenda

- Announcement
- Moore and Mealy conversion
  - Moore to Mealy conversion
  - Mealy to Moore conversion
- Common Questions
- Datapath Components

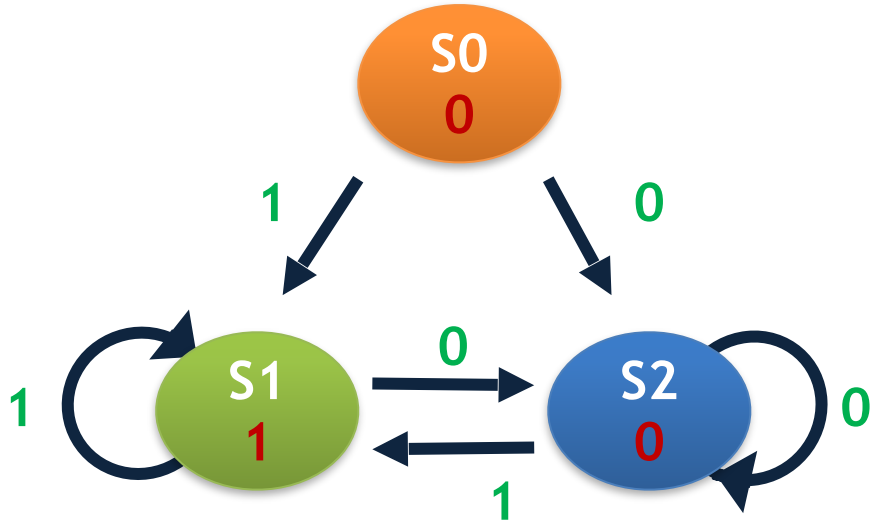
Today's class will help you:

1. Understand that Moore and Mealy machines may interchangeable
2. Understand how to convert a Moore machine to a Mealy machine
3. Understand how to convert a Mealy machine to Moore machine

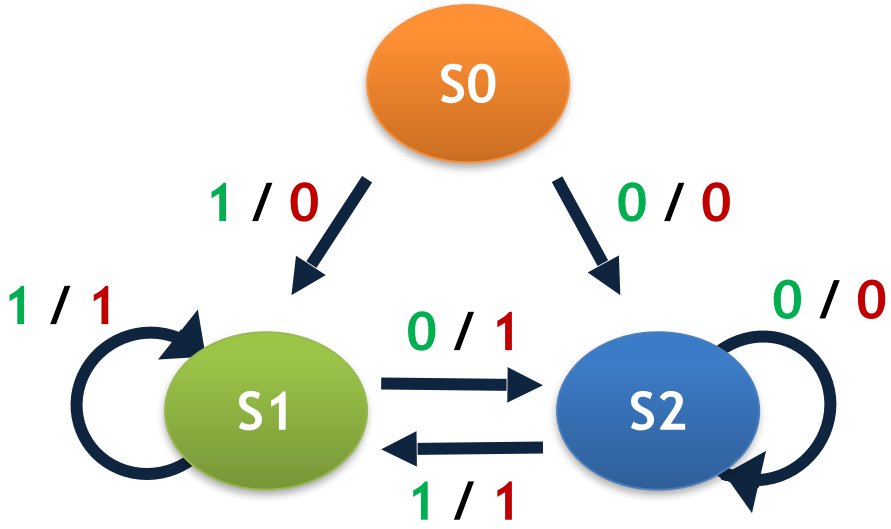
Moore vs. Mealy Machines

- **Moore machine:** **outputs** depend on **current states** only
- **Mealy machine:** **outputs** depend on **current states** and **inputs**

Moore machine

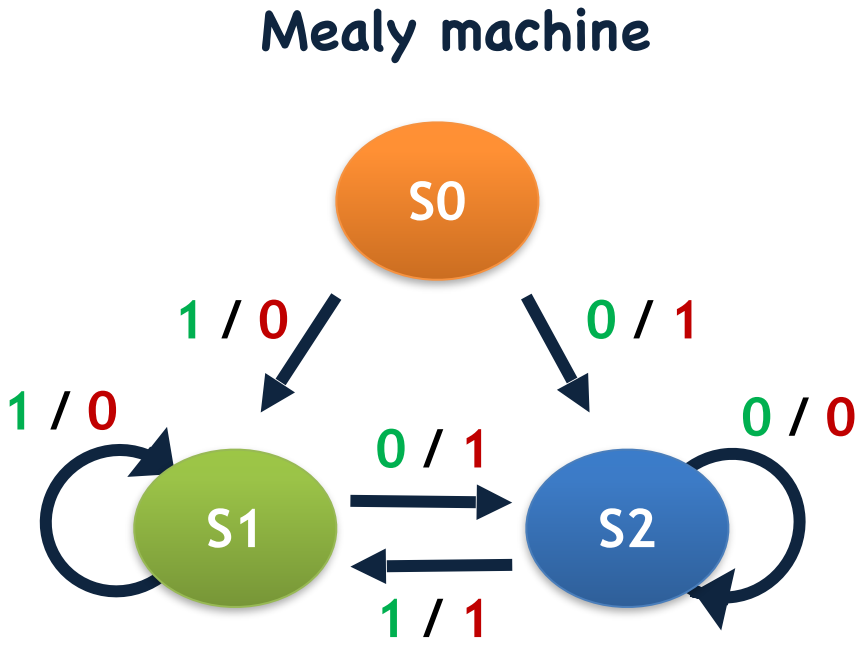


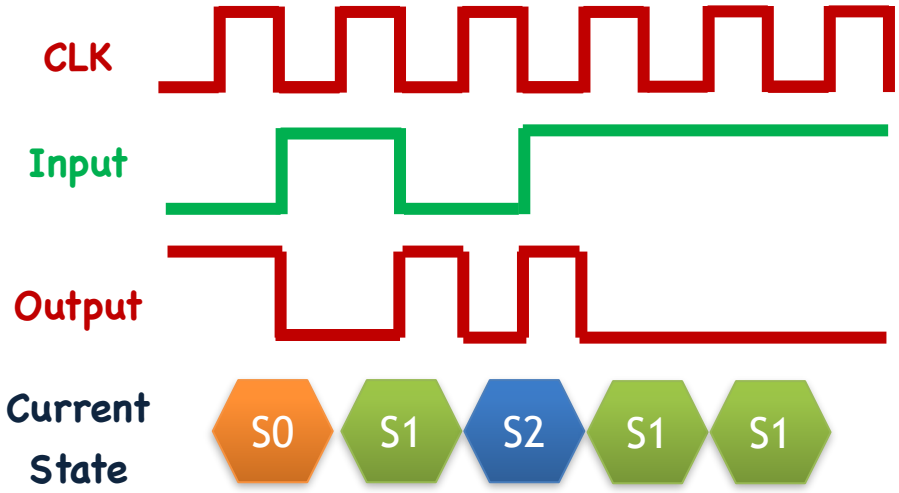
Mealy machine



Mealy Machine Input Changes

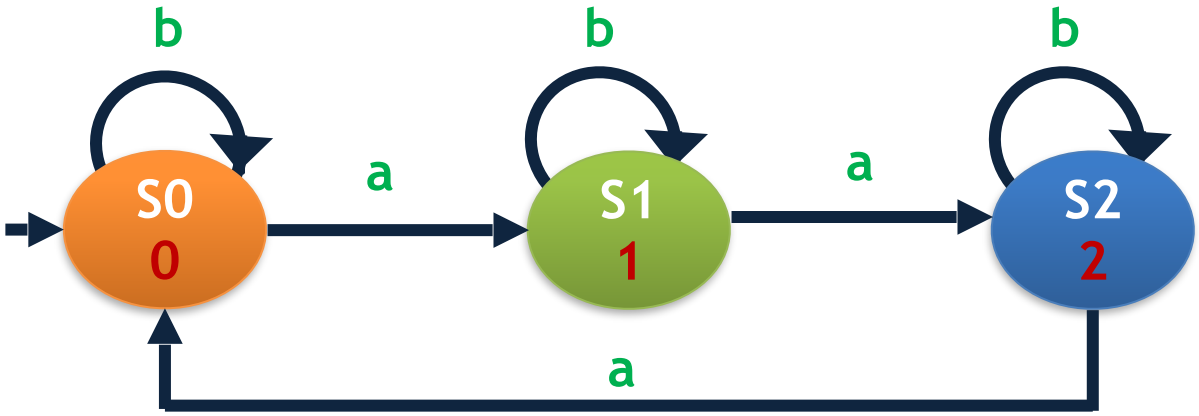
- Outputs depend on inputs and current state
- Inputs change may cause outputs change immediately
- State changes only at cock edges



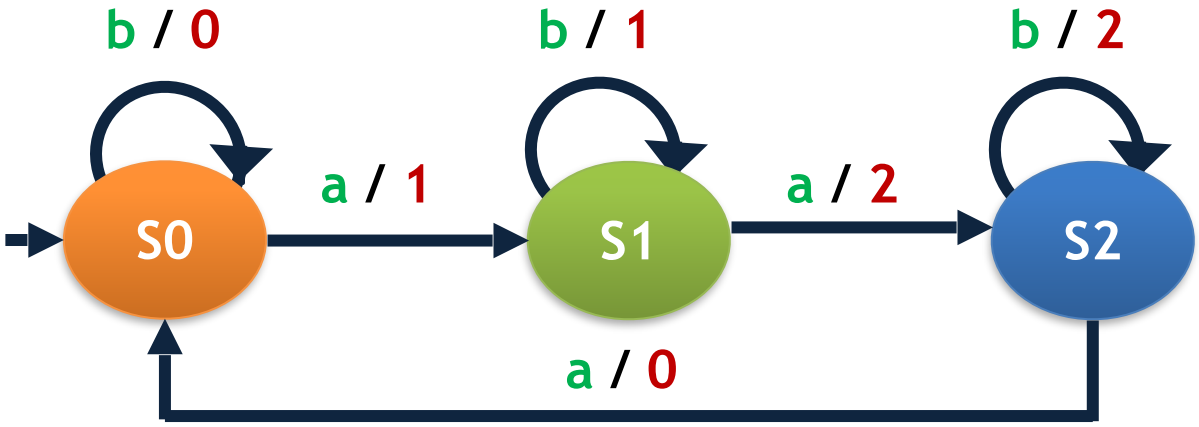


Moore to Mealy Conversion

Moore machine



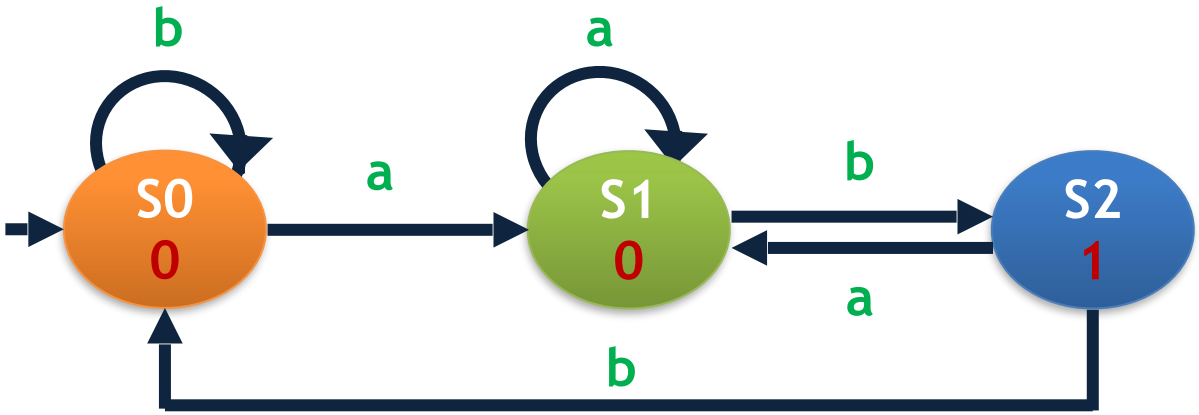
Mealy machine



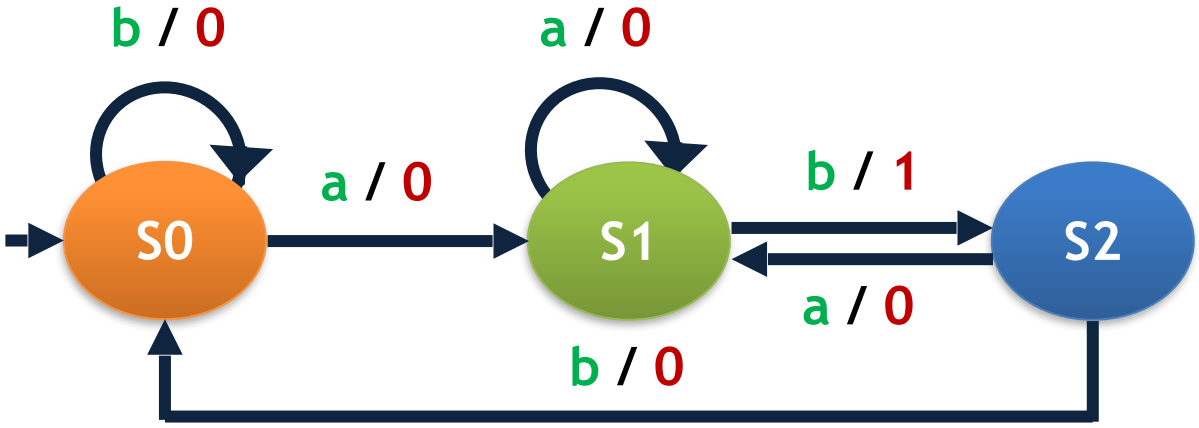
*Pictures revised from www.youtube.com/watch?v=vcNRb2cj1GI for education purpose only

Moore to Mealy Conversion

Moore machine



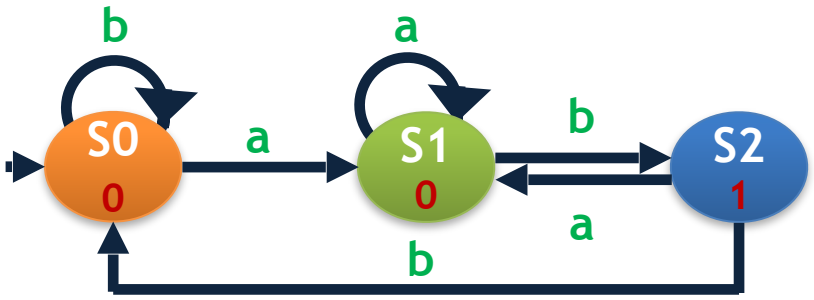
Mealy machine



*Pictures revised from www.youtube.com/watch?v=vcNRb2cj1GI for education purpose only

Moore to Mealy Conversion

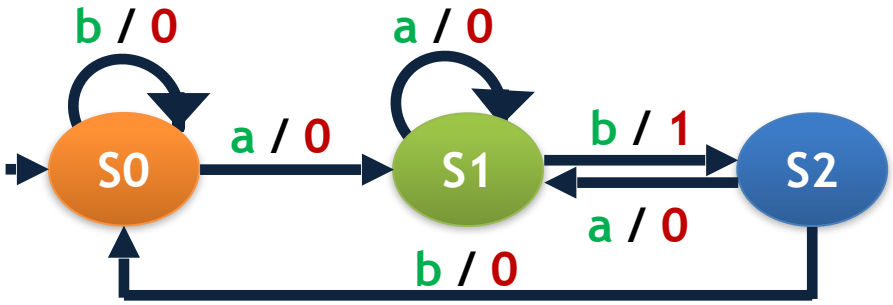
Moore machine



	a	b	output
S0	S1	S0	0
S1	S1	S2	0
S2	S1	S0	1

Next state

Mealy machine



	a	b
S0	S1, 0	S0, 0
S1	S1, 0	S2, 1
S2	S1, 0	S0, 0

Next state, Output

*Pictures revised from www.youtube.com/watch?v=vcNRb2cj1GI for education purpose only

Agenda

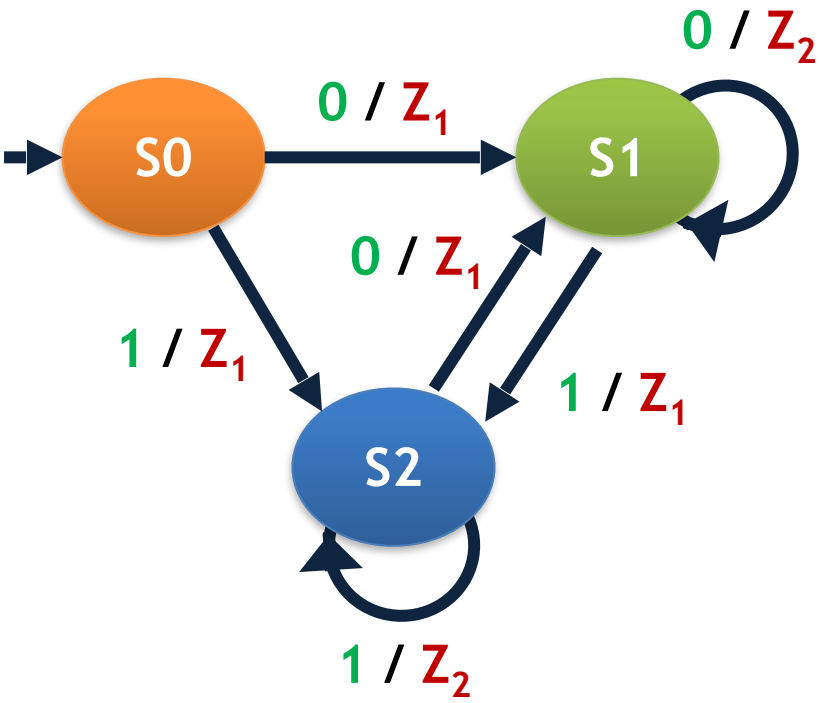
- Announcement
- Moore and Mealy conversion
  - Moore to Mealy conversion
  - Mealy to Moore conversion
- Common Questions
- Datapath Components

Today's class will help you:

1. Understand that Moore and Mealy machines may interchangeable
2. Understand how to convert a Moore machine to a Mealy machine
3. Understand how to convert a Mealy machine to Moore machine

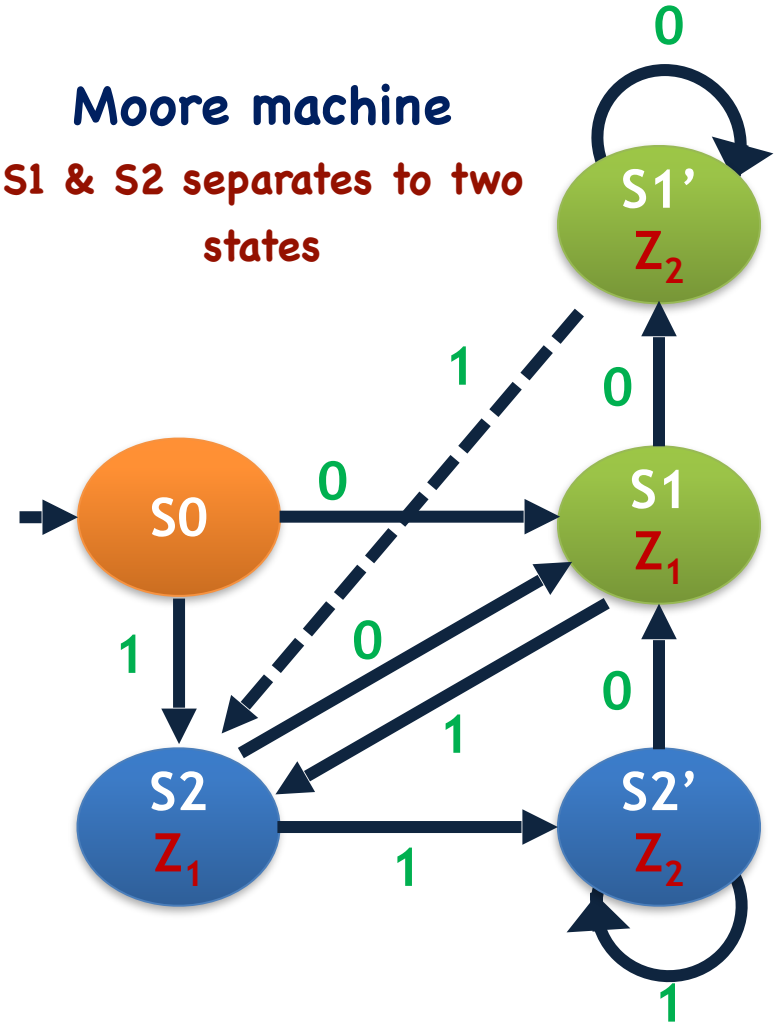
Mealy to Moore Conversion

Mealy machine



Moore machine

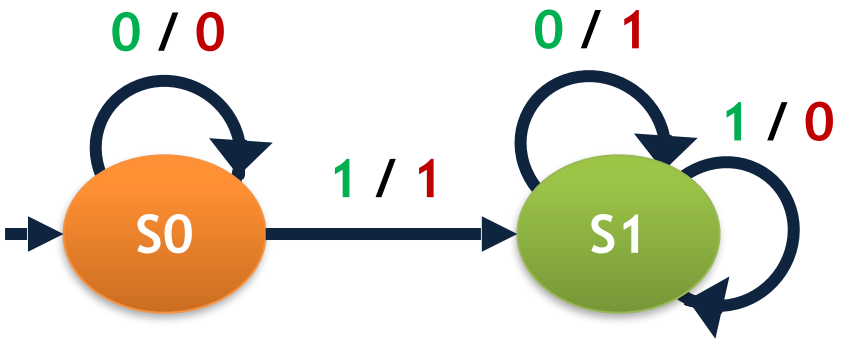
S1 & S2 separates to two states



*Pictures revised from www.youtube.com/watch?v=vcNRb2cj1GI for education purpose only

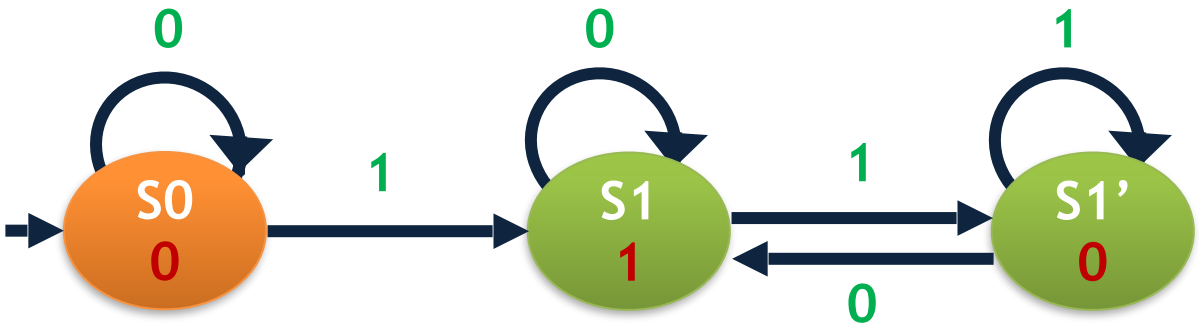
Mealy to Moore Conversion

Mealy machine



Moore machine

S1 separates to two states



*Pictures revised from www.youtube.com/watch?v=vcNRb2cj1GI for education purpose only

State Changes in Conversion

- Moore machine to Mealy machine
  - Number of states does not change
- Mealy machine to Moore machine
  - **May need to split the states**
  - Number of states increased
  - A Mealy machine with **N** states and **M** outputs may lead a Moore machine with a maximum of (**N x M**) states
  - In the previous example, **N** = 2 and **M** = 2

Agenda

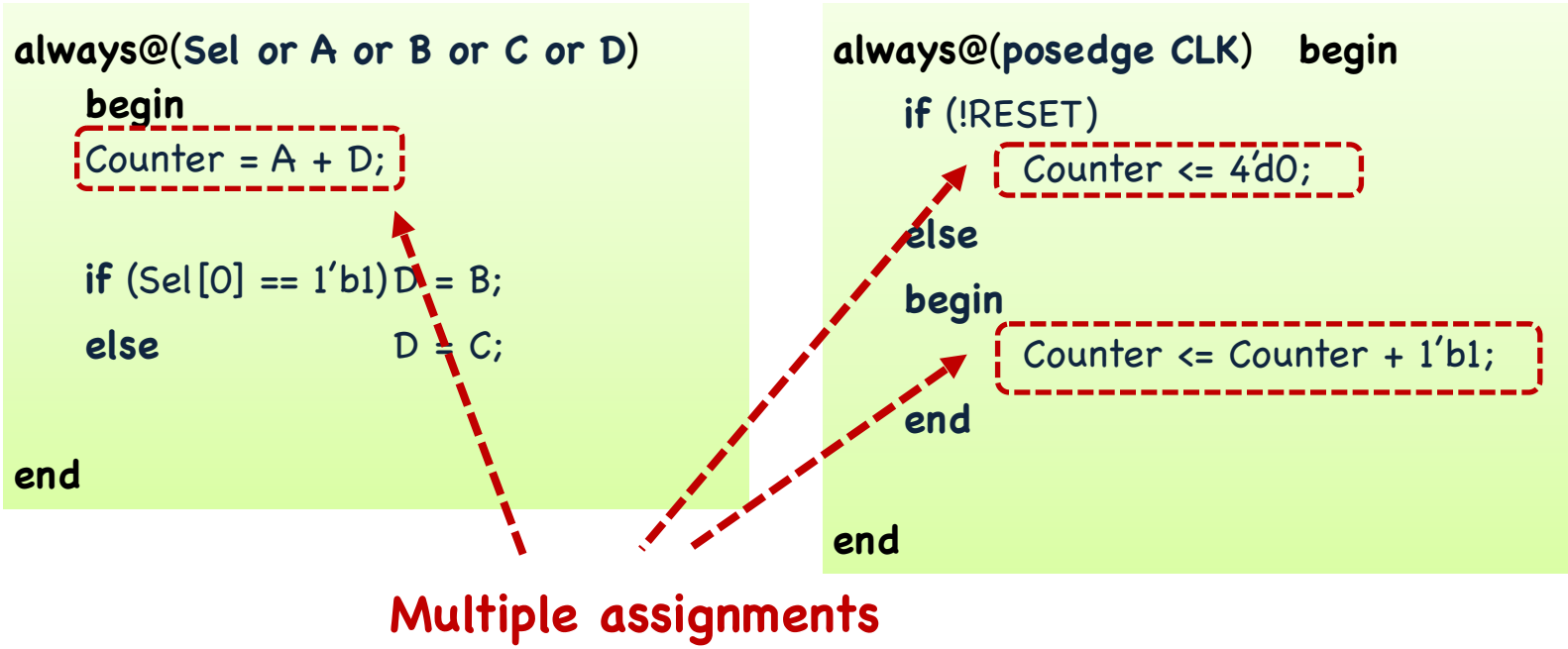
- Announcement
- Moore and Mealy conversion
  - Moore to Mealy conversion
  - Mealy to Moore conversion
- Common Questions
- Datapath Components

Today's class will help you:

1. Understand that Moore and Mealy machines may interchangeable
2. Understand how to convert a Moore machine to a Mealy machine
3. Understand how to convert a Mealy machine to Moore machine

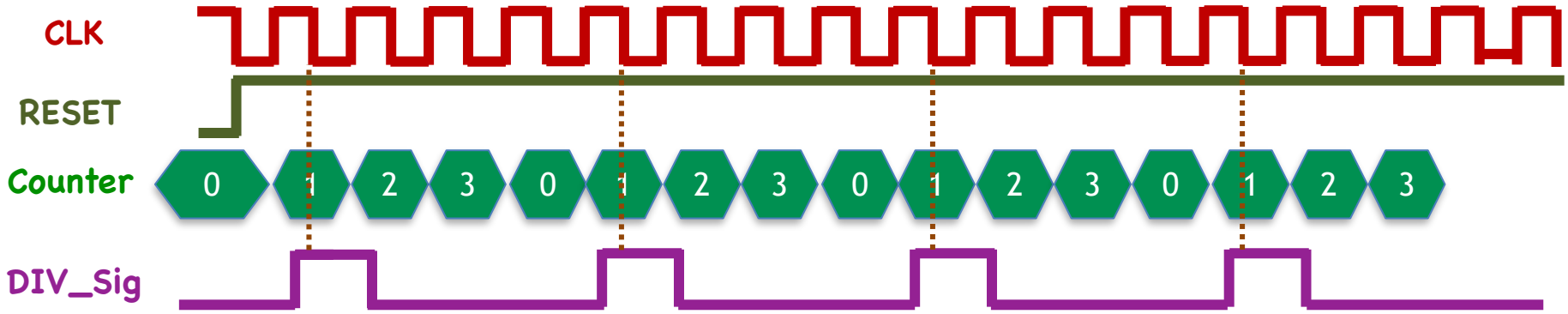
Why Can't I Configure FPGA…?

- Simulation passed, configuration failed
- Multiple assignments to **THE SAME** variable in **DIFFERENT** always blocks
- A variable can only be assigned in one single always block



Low Frequency Signals

- **Clock division in practical implementations**
  - All D-Flip Flops are triggered by the system clock signal
  - If clock division is required, use counter to create another signal called DIV_Sig
  - When DIV_Sig == 1'b1, perform your operations




```
always @ (posedge CLK) begin
        if (DIV_Sig == 1'b1)
                // Your operations here
        else
                // Your operations here
end
```

Delay Control Operator

- # followed by units of time
- Specify the delay in terms of units specified by `timescale
- **NOT synthesizable**, only used in testbench
- In real circuits, delay is realized by buffers (two inverters)

```
…
always
  begin
    #0    clock = 0;
    #50   clock = 1;
    #50;
  end
…
```

```
…
always
  begin
    #clock_period/2;
    clock = ~clock;
  end
…
```

parameter

clock_period

*Pictures revised from Prof. Ting-Chi Wang's 2013 slides for education purpose only

Agenda

- Announcement
- Moore and Mealy conversion
  - Moore to Mealy conversion
  - Mealy to Moore conversion
- Common Questions
- Datapath Components

Today's class will help you:

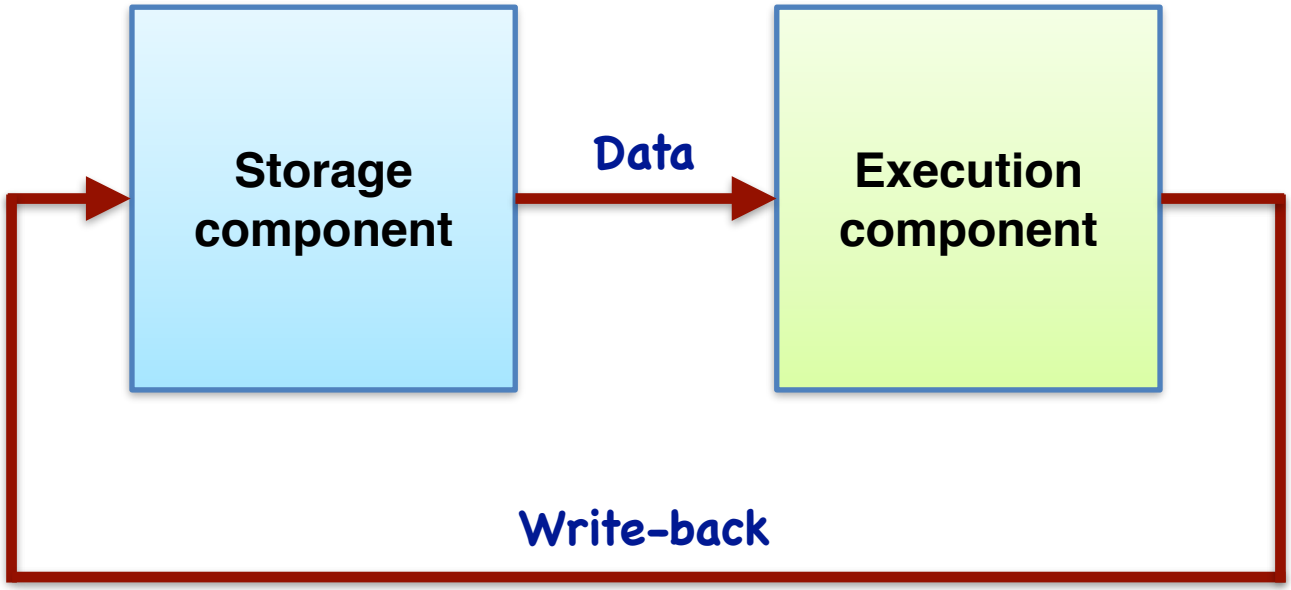
1. Understand that Moore and Mealy machines may interchangeable
2. Understand how to convert a Moore machine to a Mealy machine
3. Understand how to convert a Mealy machine to Moore machine

Datapath Components

- Design a system that contains control, execution, and storage components
  - **You have already learned all of these components in this semester**
- Data are stored in **storage components**
  - Memory
  - Cache
  - Registers
- Data are processed in **execution components**
  - Arithmetic components (adders, multipliers, subtractors, etc.)
  - Logical operation components (AND, OR, NOT, etc.)
  - Shifters and comparators
- Data flow is determined by **control components**

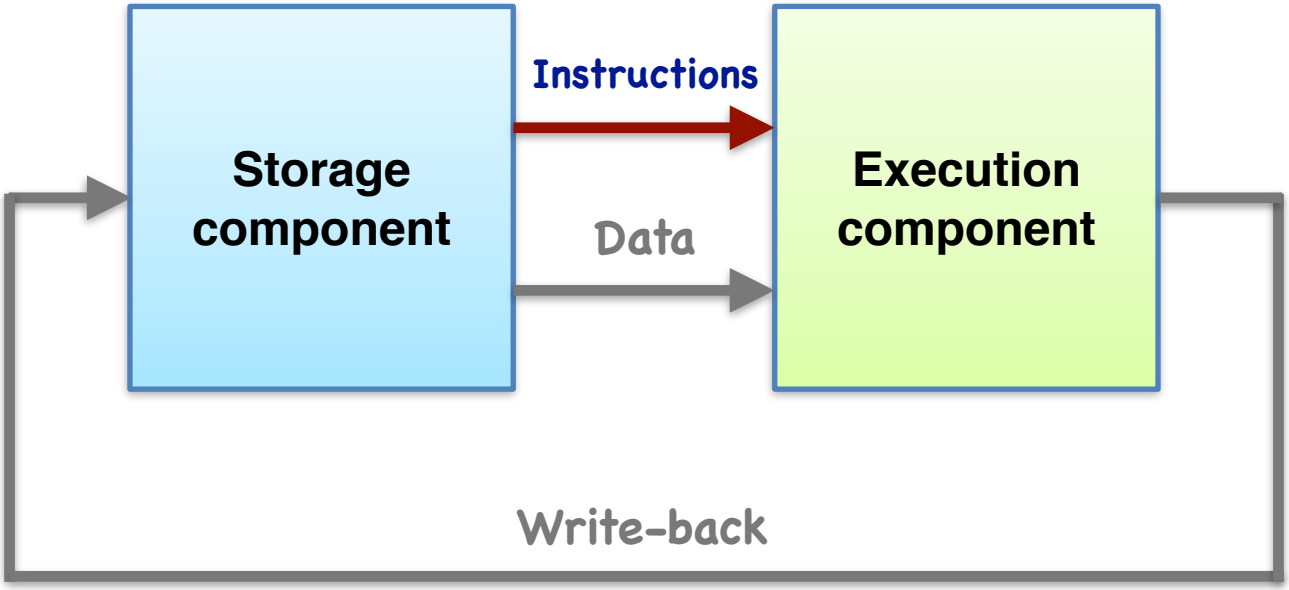
Simple Datapath Structure

- Storage components provides data to be processed
- Execution component process the data
- The result is stored back to the storage component



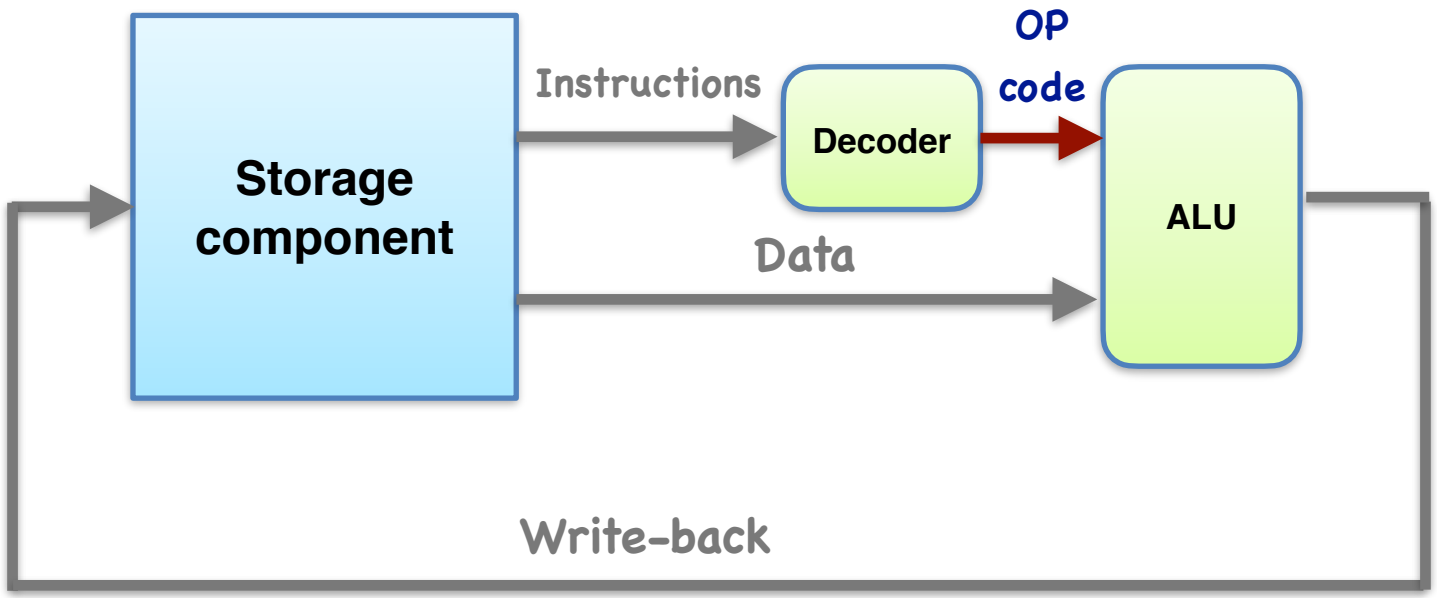
Instructions Comes In

- Instructions tells the execution component what operations to execute
- Depending on the instructions, the execution component selects the appropriate sub-units to run



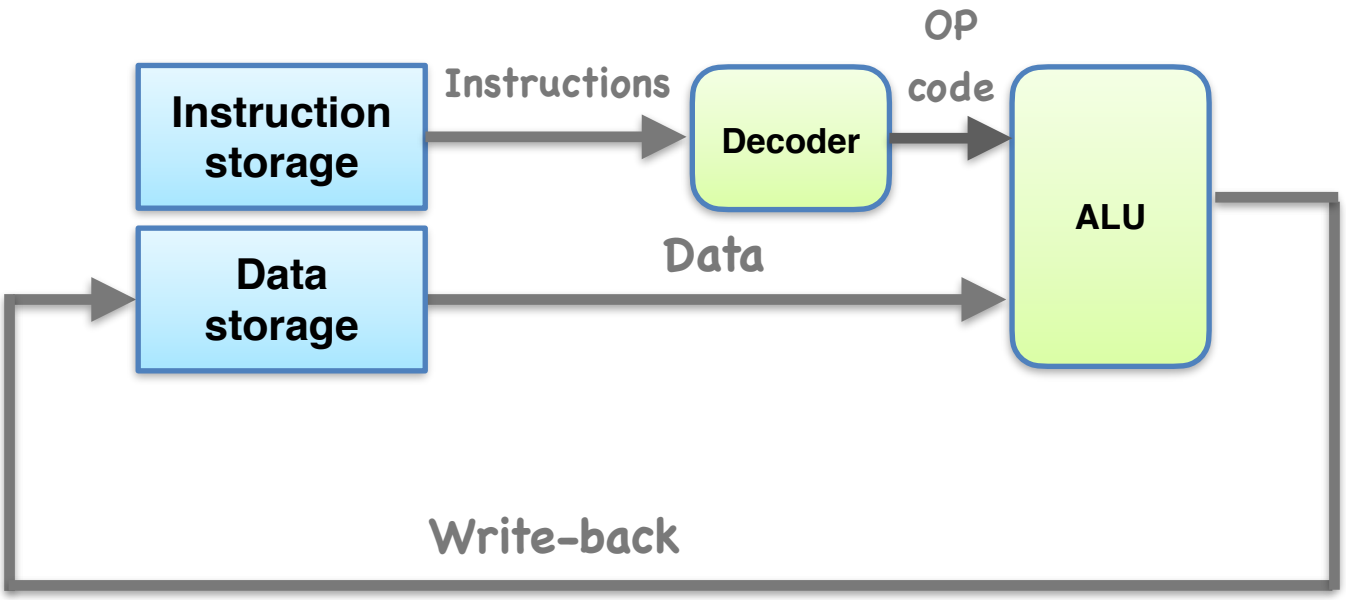
Decoder Comes In

- ALU stands for **A**rithmetic **L**ogic **U**nit
- Instructions are decoded by an decoder (Lab 1)
- The decoder generates an OP code for ALU (Lab 2)
- The ALU contains adders, multipliers, shifters, etc for execution (Lab 1 & 2)



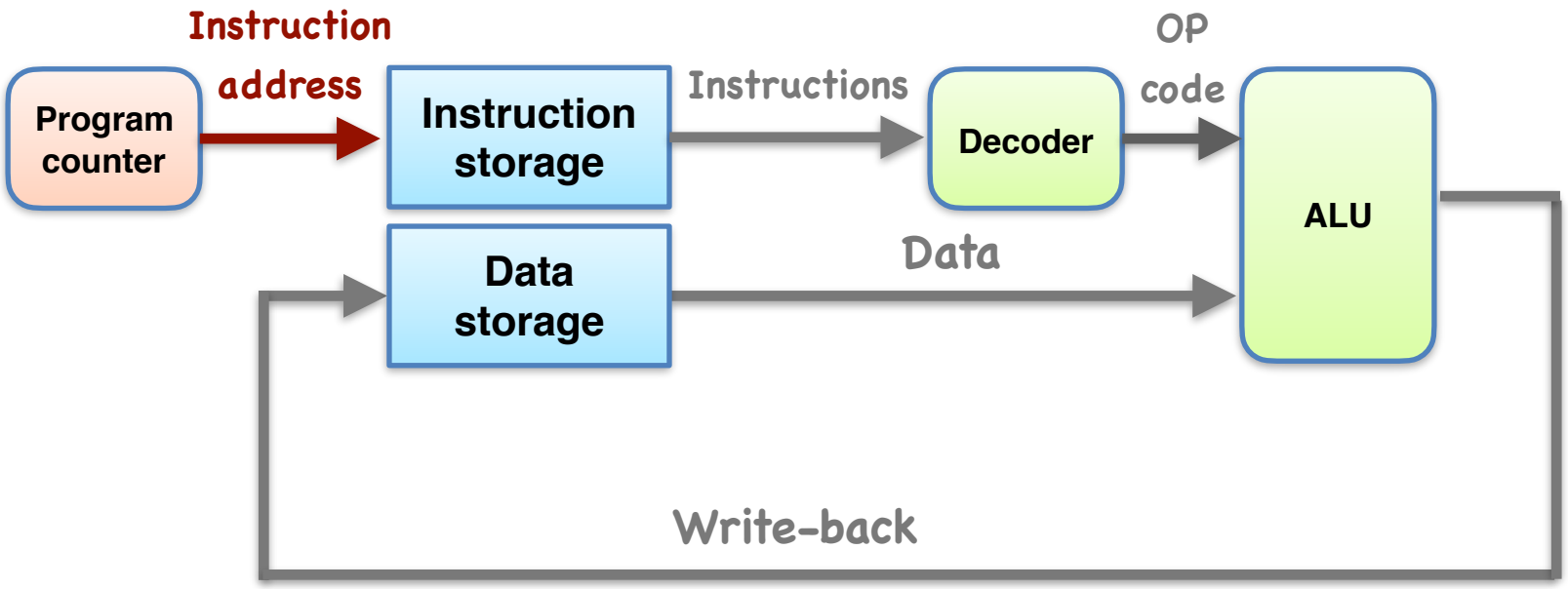
Separation of Storage Component

- The storage component can be splitted into instruction storage and data storage
- The instruction storage stores the instructions to be operated
- The data storage stores the data to be processed



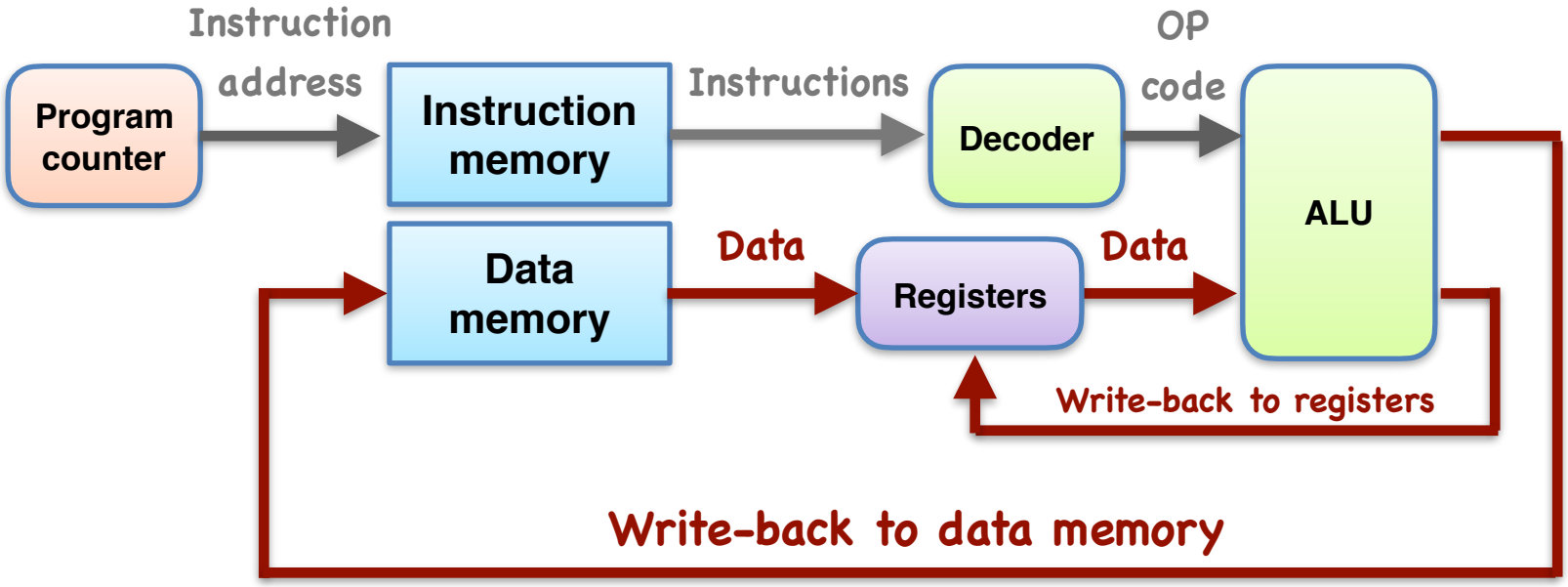
Program Counter

- Program counter (PC) provides the address of the instruction to be executed in the instruction storage
- Program counter updates the address to the next instruction every cycle
  - PC <= PC + 4 (4 bytes per instruction)
- The instructions stored in the instruction storage are called **a program**



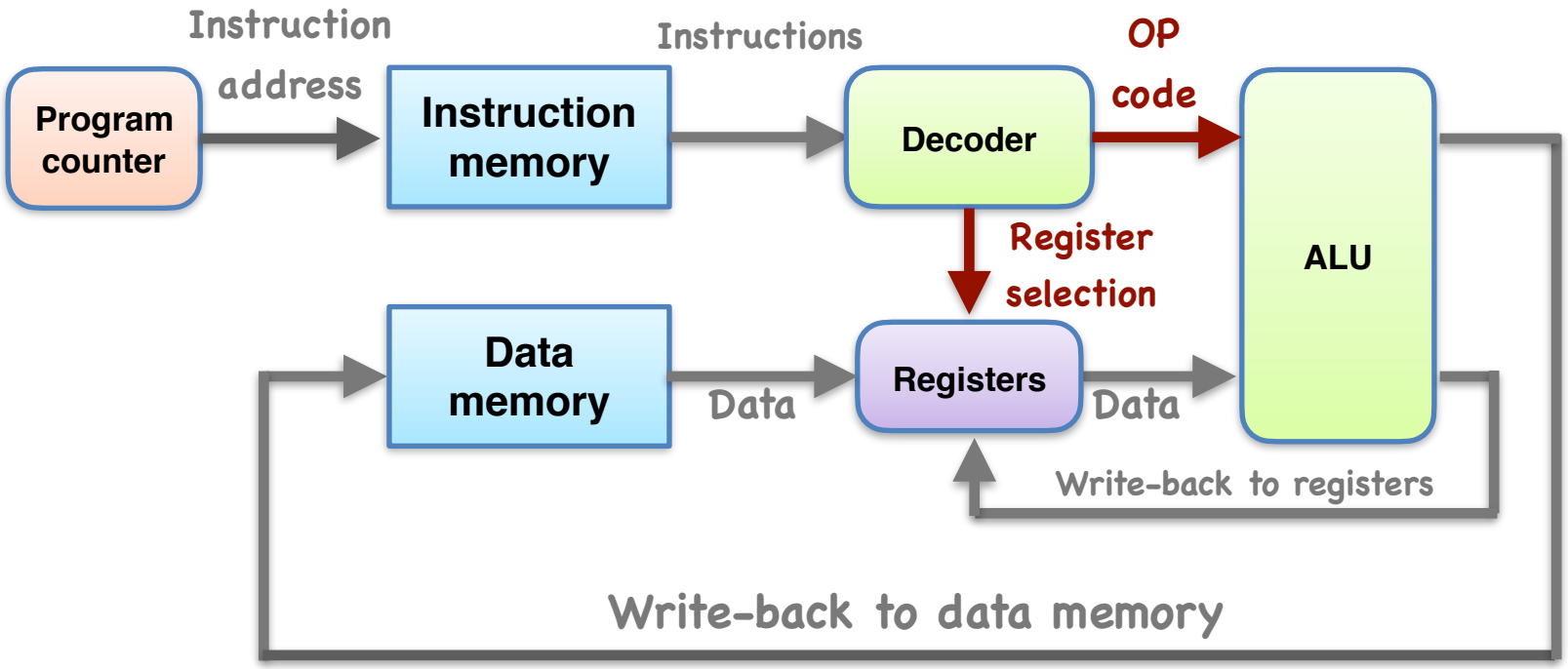
Registers for Local Storage

- Data storage is separated into registers and data memory
- Registers are small amount of storage elements constructed from DFFs
- Data can be loaded from the data memory to the registers
- Data can also be stored back to the registers from the ALU results



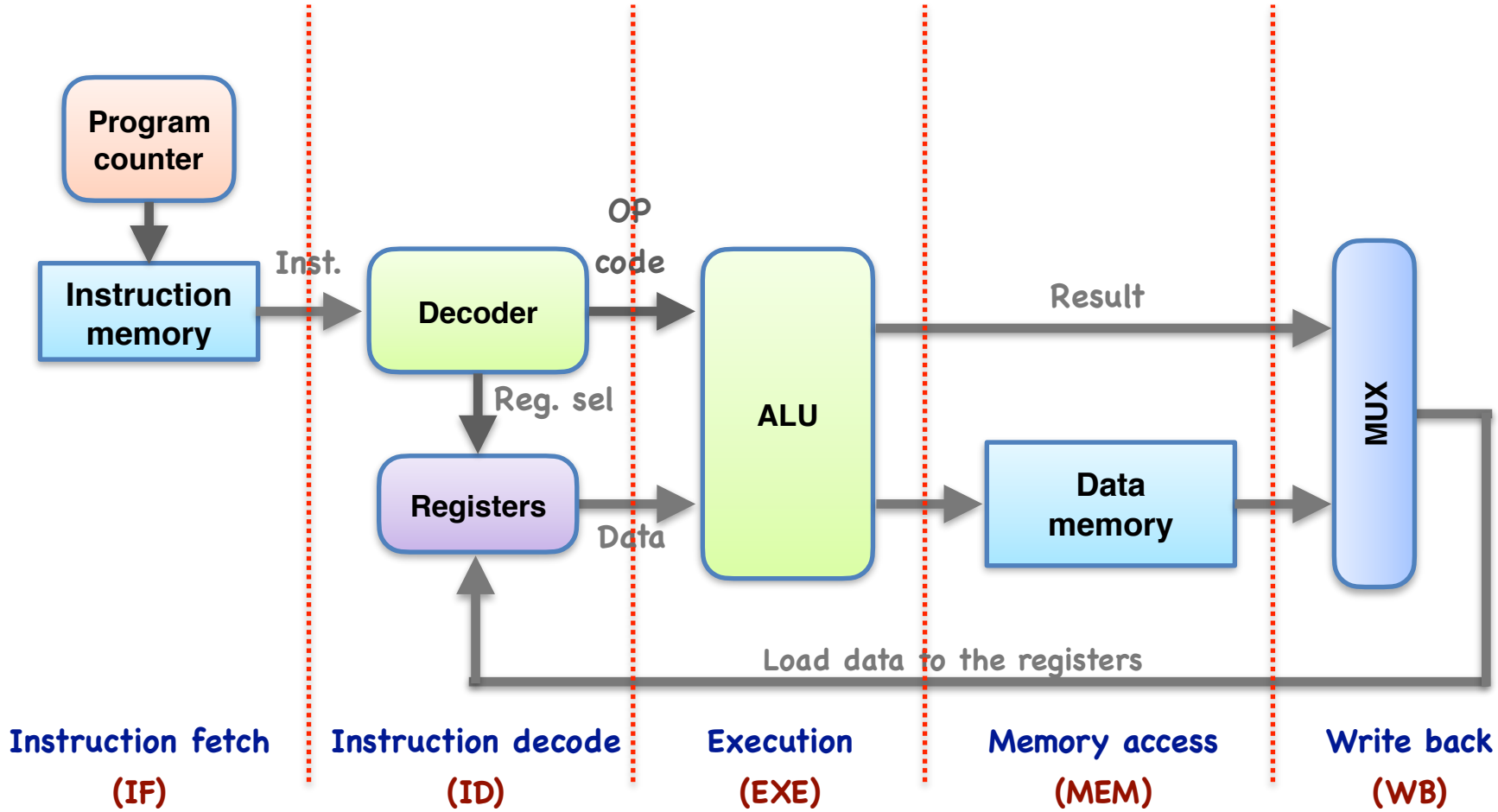
Selection of Registers

- The decoder generates two items: **OP code** and **register selection**
- The decoder selects the registers whose contents are to be fed into ALU



Five Stage Pipeline

- The datapath can be re-organized and separated into five stages
- This five stage pipeline structure is the basis of computer architecture




Thank you for your attention!
*Pier view taken at Hollywood, Los Angeles, California, USA.
This picture is taken by Chun-Yi Lee himself, who is also a fan of photography
Photo by C.-Y. Lee © 2013