Welcome to The Hardware Lab!

Fall 2020

Lab 6: Peripheral Components: VGA, Mouse, and Dual FPGA

Prof. Chun-Yi Lee

Department of Computer Science

National Tsing Hua University

Agenda

- Lab 6 Outline
- Lab 6 Basic Questions
- Lab 6 Advanced Questions

Lab 6 Outline

- Basic questions (2%)
  - Individual assignment
  - Due on 12/3/2020. Demonstration on your FPGA board (In class)
  - Only demonstration is necessary. Nothing to submit.

- Advanced questions (5%)
  - Group assignment
  - ILMS submission due on 12/17/2020. 23:59:59.
    - Submit your FPGA codes to ILMS by 12/17/2020. 15:00:00.
  - Demonstration on your FPGA board (In class)
  - Assignment submission (Submit to ILMS)
    - Source codes and testbenches
    - Lab report in PDF

Lab 6 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 6 Submission Requirements

- Source codes and testbenches
  - Please follow the templates EXACTLY
  - We will test your codes by TAs' testbenches
- Lab 6 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 6

Agenda

- Lab 6 Outline
- Lab 6 Basic Questions
- Lab 6 Advanced Questions

Basic Questions

- Individual assignment
- FPGA demonstration (due on 12/3/2020. In class.)
  - VGA sample code
  - Mouse sample code
- Demonstrate your work by FPGA

Basic FPGA Demonstration 1

- **VGA sample codes**
  - Please implement the keyboard sample codes 1 & 2 released on ILMS

- **Mouse sample codes**
  - Please implement the mouse sample code released on ILMS

Agenda

- Lab 6 Outline
- Lab 6 Basic Questions
- Lab 6 Advanced Questions

Advanced Questions

- Group assignment
- FPGA demonstration (due on 12/17/2020. In class.)
  - Dual FPGA communication
  - The slot machine
  - The car

Dual FPGA Communication

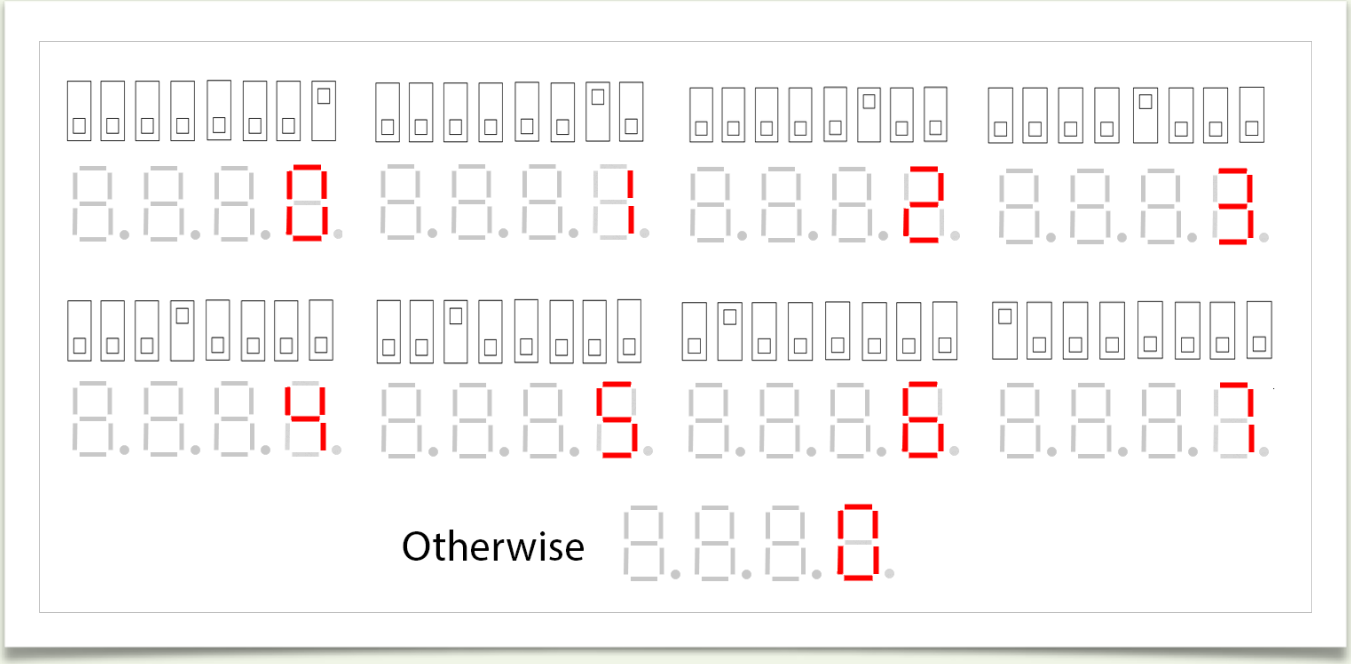
Requirements

- Please design a simple **FPGA-to-FPGA** communication protocol
- The protocol is required to fulfill the following requirements:
  - Use the Handshaking protocol described below to send a number from a **Master FPGA** to a **Slave FPGA**
    - [**Master** -> **Slave**] Request
    - [**Slave** -> **Master**] ACK
    - [**Master** -> **Slave**] Send data (number)
  - Your design should be demonstrable in an observable speed so that TAs can know whether your design is correct or not
  - Your design should be stable and should avoid signal loss

Dual FPGA Communication

Data Representation

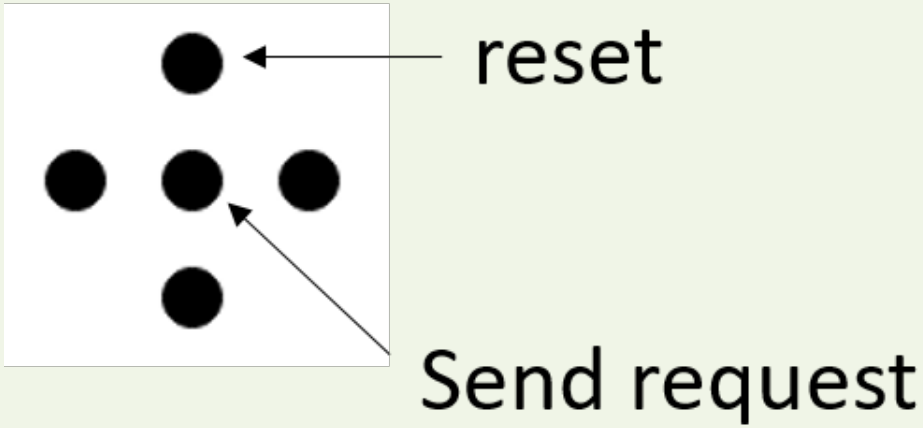
- For the **Master FPGA**, please use switches to represent numbers in one-hot form
- For the **Slave FPGA**, please display the numbers on your 7-segment displays
- Please illuminate LED[0] for at least 1 second when FPGA receive a request or an ACK
- Below are input and the corresponding 7-segment display



Dual FPGA Communication

Button Control

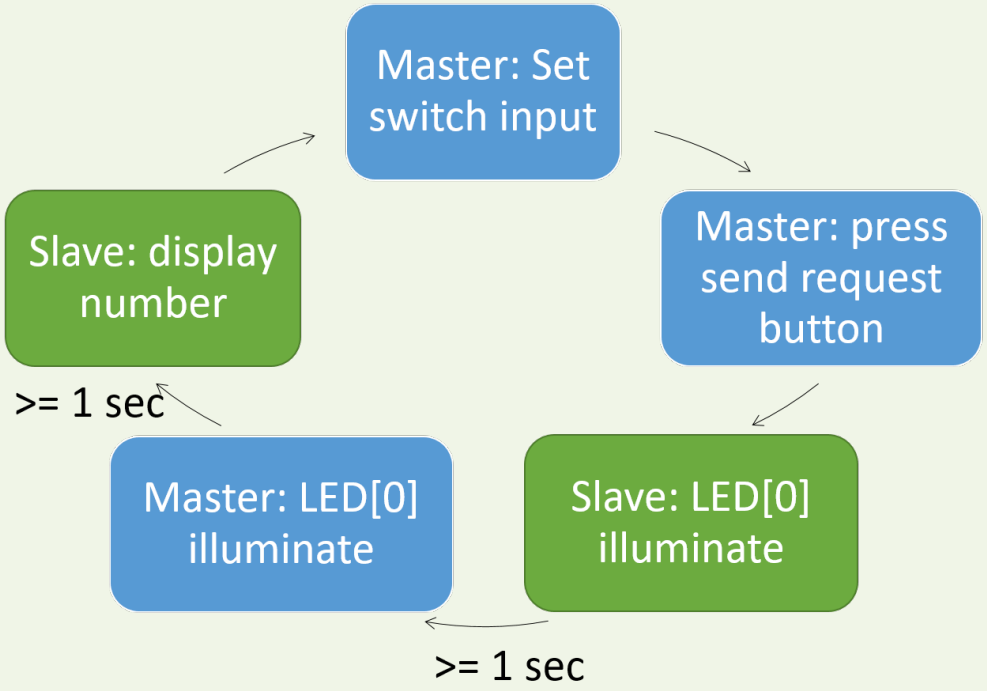
- The **UP** button is for **reset**, and the **MIDDLE** button is for **sending requests**
- The communication starts only after the **send request button** of the **Master FPGA** is pressed
- When the **Master FPGA** resets, it stop communicate with the **Slave FPGA** until the next **send request button** is pressed
- When the **Slave FPGA** resets, the 7-segment display **0** until next request
- The reset action of the two FPGA is independent of each other



Dual FPGA Communication

Communication Process

- The whole communicate process is designed as below:

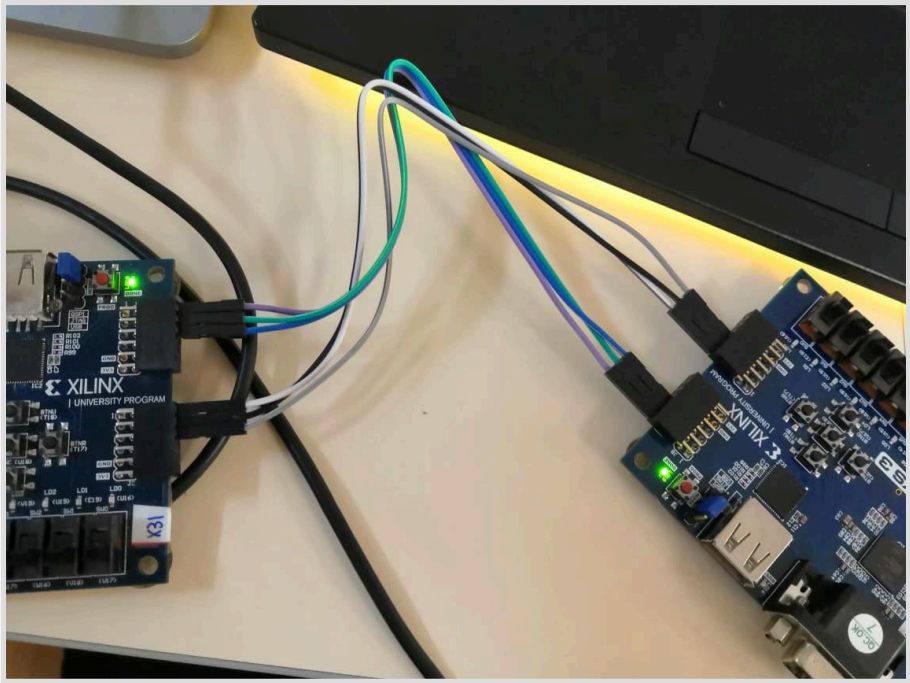


- The display on the Slave FPGA should be hold until the data of next request is received

Dual FPGA Communication

Port Connection via Jumper

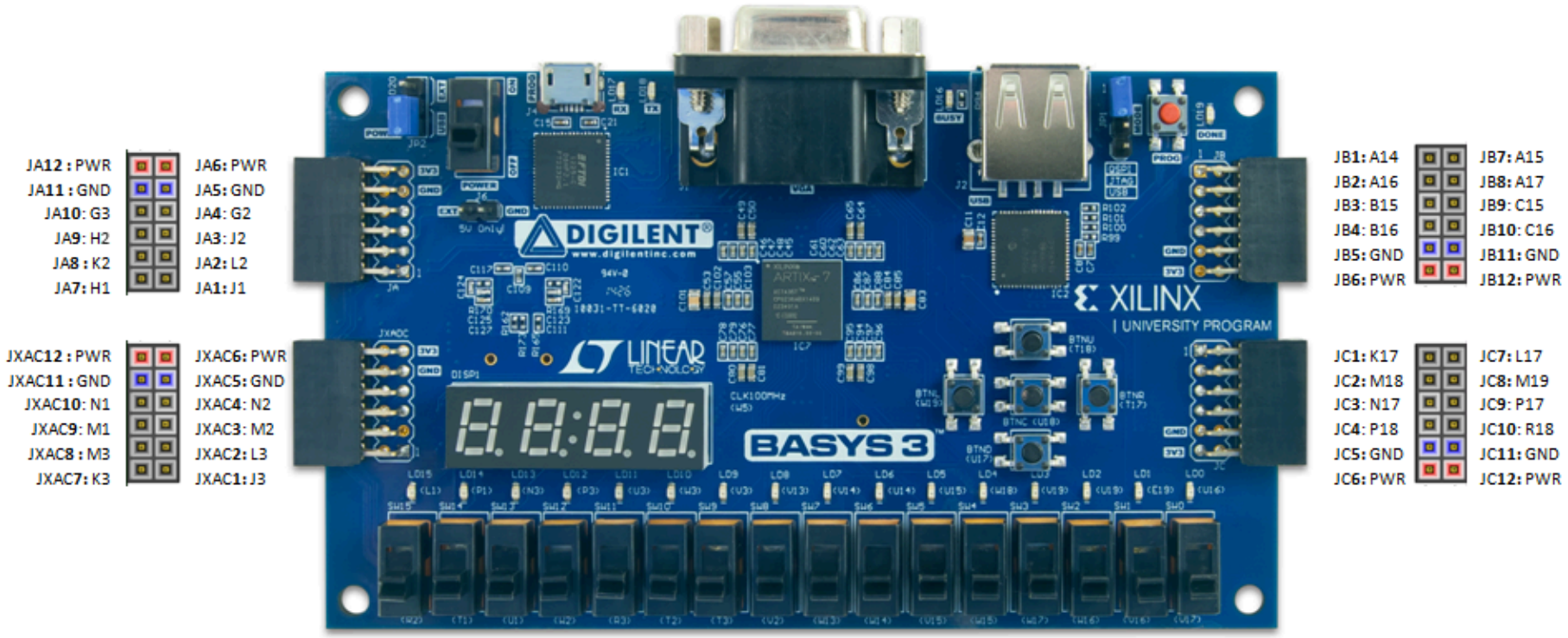
- A demonstration of the ports connection via jumpers (as defined in the XDC file) is provided below.
- In case that some ports are malfunctioned or failed to work corrected, you are also allowed to use the other ports, as long as the two FPGAs can communicate correctly according to our problem specifications.
  - A reference PMOD port mapping diagram is provided in the next page.



Dual FPGA Communication

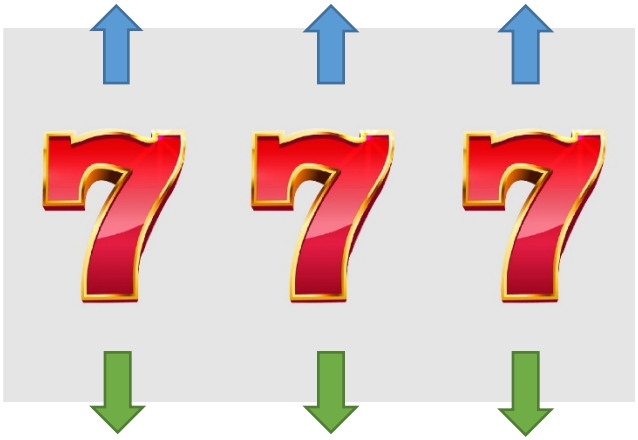
PMOD Reference Diagram

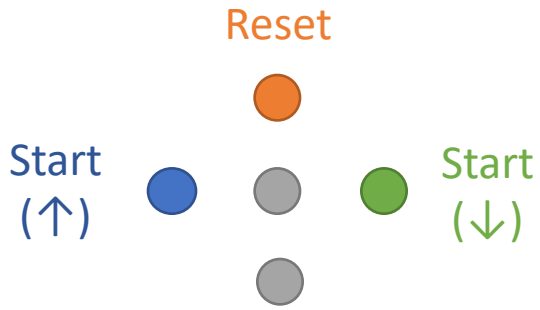
Basys3: Pmod Pin-Out Diagram



The Slot Machine

- The slot machine will run in a upward direction as you press "Start (↑)", and in a downward direction as you press "Start (↓)".
- Press "Reset" to reset the machine.
- Remember to add **debounce** and **one-pulse** circuits to your buttons.
- The moving behavior of each digit should be **the same** as that in the sample code.





The Car

- Please refer to another slide deck for the details.
- Make sure your car can run on the track correctly.
- Use ultrasonic sensor to detect the distance.
  - If distance < 40cm, stop the car.
- We will have two tracks.
  - In the basic track, we only care about its correctness.
  - In the bonus track, we will test its correctness and speed.

Thank you for your attention!

*Schloß Neuschwanstein, Germany
This picture is taken by Chun-Yi Lee himself, who is also a fan of photography