

Computer Organization 104-2

Lab 6: Multi-Core

Due: 2016/06/20 23:59:59

1. Goal

- Use the CPU in lab5 to implement a multi-core computing environment.

2. Requirement

- Use LAB5 pipelined CPU to implement **multi-core CPU**
 - Must to implement **Hazard Detection** and **Forwarding Unit**
 - Need to forward data if instructions have data dependency
 - Need to stall pipelined CPU if it detects load-use
- Use the following instruction sets to do matrix multiplication

ADD	MULT	ORI	SLL	BGT	J	LW
ADDI	AND	SLT	BEQ	BNEZ	JAL	SW
SUB	OR	SLTI	BNE	BGEZ	JR	

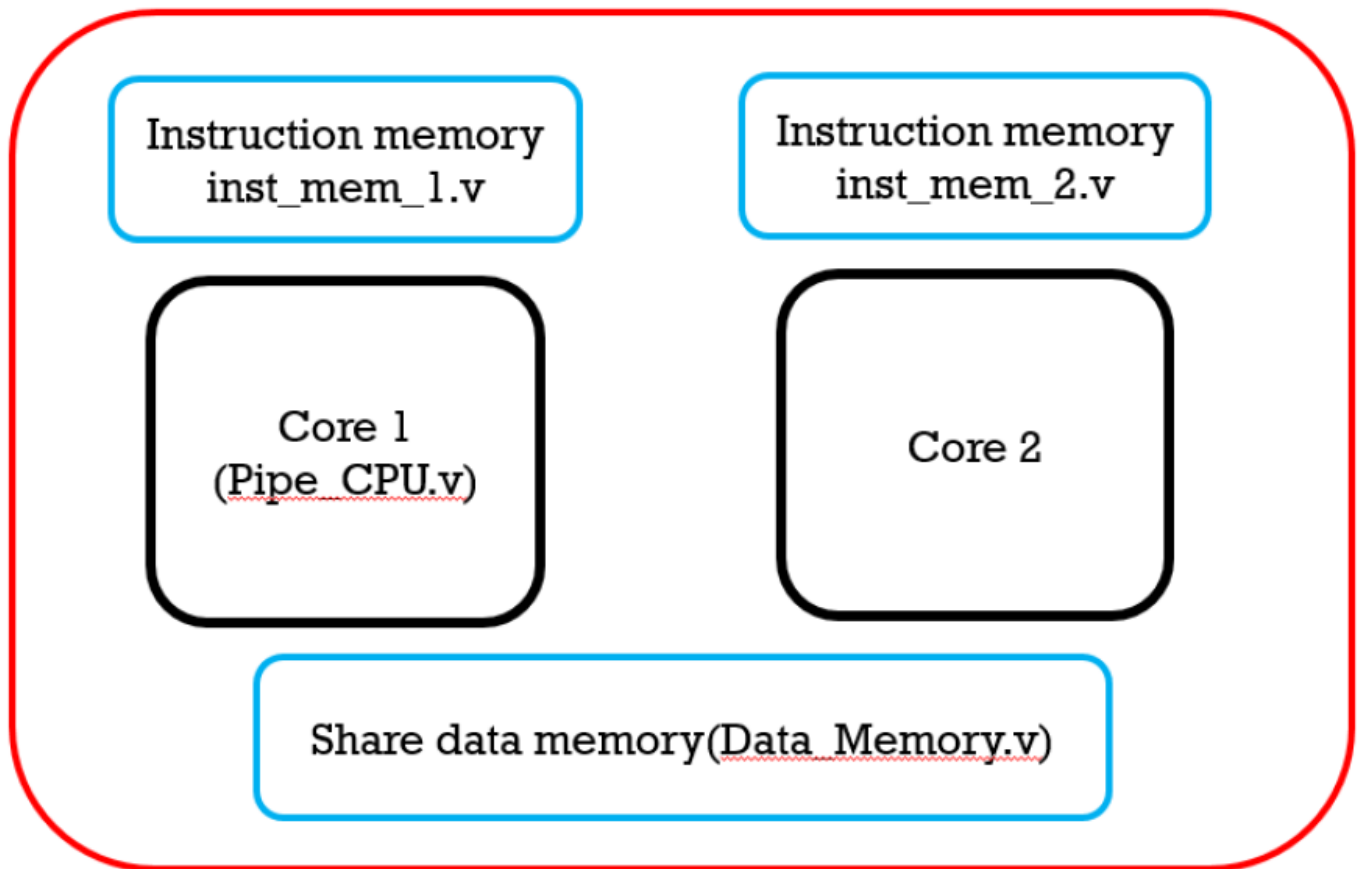
- Initial the matrix content in Data_Memory.v like the following graph
 - You can modify the in/output port of Data_Memory.v by yourself
 - **mem[address](value)**
 - TAs will modify the initial value when grading LAB6

$$\begin{bmatrix} \text{mem}[0](1) & \text{mem}[1](3) & \text{mem}[2](-100) \\ \text{mem}[3](2) & \text{mem}[4](-2) & \text{mem}[5](10) \\ \text{mem}[6](3) & \text{mem}[7](1) & \text{mem}[8](0) \end{bmatrix} * \begin{bmatrix} \text{mem}[9](3) & \text{mem}[10](4) \\ \text{mem}[11](-2) & \text{mem}[12](5) \\ \text{mem}[13](-1) & \text{mem}[14](6) \end{bmatrix} = \begin{bmatrix} \text{mem}[15] & \text{mem}[16] \\ \text{mem}[17] & \text{mem}[18] \\ \text{mem}[19] & \text{mem}[20] \end{bmatrix}$$

- Submit your machine code and assemble code
 - File name:
 - ◆ LAB6_machine_1.txt / LAB6_ASM_1.txt (for core 1)
 - ◆ LAB6_machine_2.txt / LAB6_ASM_2.txt (for core 2)
 - No grades will get if the file name is wrong
- Write your own testbench called TESTBENCH.v
- Answer the following questions in your report:
 - How do you separate the program for the two cores?
 - Assumed that programmers do not know the platform architecture (i.e. single core or multi-core) how can programmers manage their program partition?
 - Assumed that each core has private cache. If core1 write a new data at address 0x123, how could core 2 get the new data from 0x123? (hint: coherence)
 - Draw the detail architecture and describe your design in your report.
 - Say something if you want
- It's a no team assignment. Please attach your names and student IDs as comments in the top of each file. The assignment you upload on e3 must have the form of "student_ID_LAB6.rar".

3. Architecture Diagram

TESTBENCH.v



4. Grading Policy

- Total source: 110pts
 - Design: 100 pts, Report: 10 pts.
 - ※ **Any Plagiarism will be punished with a null score!**
- **Delay: 10%off/day**

7. Hand in your assignment

Please upload the assignment to the E3.

Put all of .v source files and report into same compressed file. (Use your student ID to be the name of your compressed file)