Computer Organization 2016

HOMEWORK II

Due date: 2016/4/21 23:59

The goal of this homework is to let the students be familiar with the MIPS instruction set and Verilog, a hardware description language (HDL) used to model electronic systems. In this homework, you need to implement a basic MIPS CPU. Follow the instruction table in this homework and satisfy all the homework requirements. Please use the benchmark provided by TA to verify your CPU correctly. The verification tool is Modelsim. You will learn how to use Modelsim in Lab 2.

General rules for deliverables

- This homework needs to be completed by INDIVIDUAL student. If your code is copied, you will not get any scores.
- Compress all files into a single zip file, and upload the compressed file to Moodle.
  - The file hierarchy

F740XXXXX(your id)(folder)

SRC(folder) * Store your source code

F740XXXXX.docx(your Project Report)

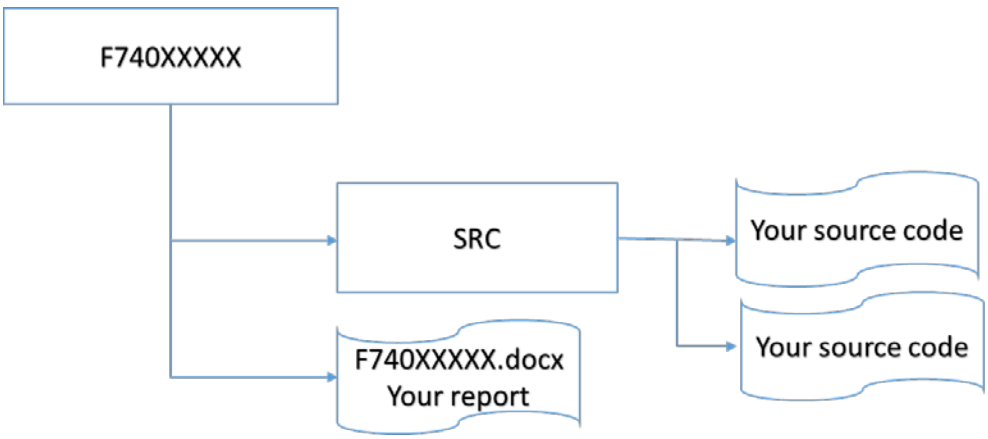


Fig.1 File hierarchy for homework submission

- **Important!** AVOID submitting your homework in the last minute. Late submission is not accepted.
- You should finish all the requirements (shown below) in this HW and Project report.
- Please use Project Report template on Archive to finish your Project report. HW2 Archive on Course website:
  Computer Organization 2016 course website

Exercise

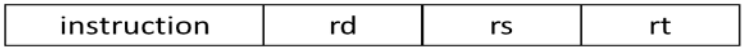
You need to implement a CPU that can execute all the instructions shown in the *MIPS ISA* section. In addition, you need to verify your CPU by using Modelsim. Note that, TAs will provide a simple architecture for the CPU and you just need to implement some incomplete/missing modules, e.g. ALU, control unit, regfile, etc.

Please finish all the modules, use the benchmark provided by TAs to verify the CPU, take a snapshot, and finally explain the snapshot, including the wires, signals …… in your report.

MIPS ISA

R Type

Assembler Syntax

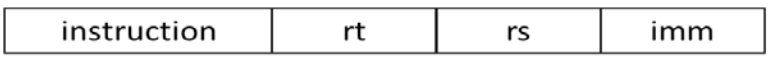


Machine code Format

opcode	Mnemonics	SRC1	SRC2	DST	funct	Description
000000	nop	00000	00000	00000	000000	No operation
000000	add	$Rs	$Rt	$Rd	100000	Rd = Rs + Rt
000000	sub	$Rs	$Rt	$Rd	100010	Rd = Rs – Rt
000000	and	$Rs	$Rt	$Rd	100100	Rd = Rs & Rt
000000	or	$Rs	$Rt	$Rd	100101	Rd = Rs | Rt
000000	xor	$Rs	$Rt	$Rd	100110	Rd = Rs ^ Rt
000000	nor	$Rs	$Rt	$Rd	100111	Rd = ~(Rs | Rt)
000000	slt	$Rs	$Rt	$Rd	101010	Rd = (Rs < Rt)?1:0
000000	sll		$Rt	$Rd	000000	Rd = Rt << shamt
000000	srl		$Rt	$Rd	000010	Rd = Rt >> shamt
000000	jr	$Rs			001000	PC=Rs

I Type

Assembler Syntax



Machine code Format

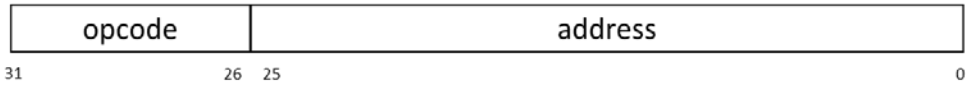
opcode	Mnemonics	SRC1	DST	SRC2	Description
001000	addi	$Rs	$Rt	imm	Rt = Rs + imm
001100	andi	$Rs	$Rt	imm	Rt = Rs & imm
001010	slti	$Rs	$Rt	imm	Rt = (Rs < imm) ? 1 : 0
000100	beq	$Rs	$Rt	imm	If(Rs == Rt) PC=PC+4+imm
000101	bne	$Rs	$Rt	imm	If(Rs != Rt) PC=PC+4+imm
100011	lw	$Rs	$Rt	imm	Rt = Mem[Rs + imm]
101011	sw	$Rs	$Rt	imm	Mem[Rs + imm] = Rt

J Type

Assembler Syntax

instruction	Target(label)

Machine code Format



opcode	Mnemonics	Address	Description
000010	j	jumpAddr	PC = jumpAddr
000011	jal	jumpAddr	R[31] = PC + 8 ; PC = jumpAddr

CPU Datapath

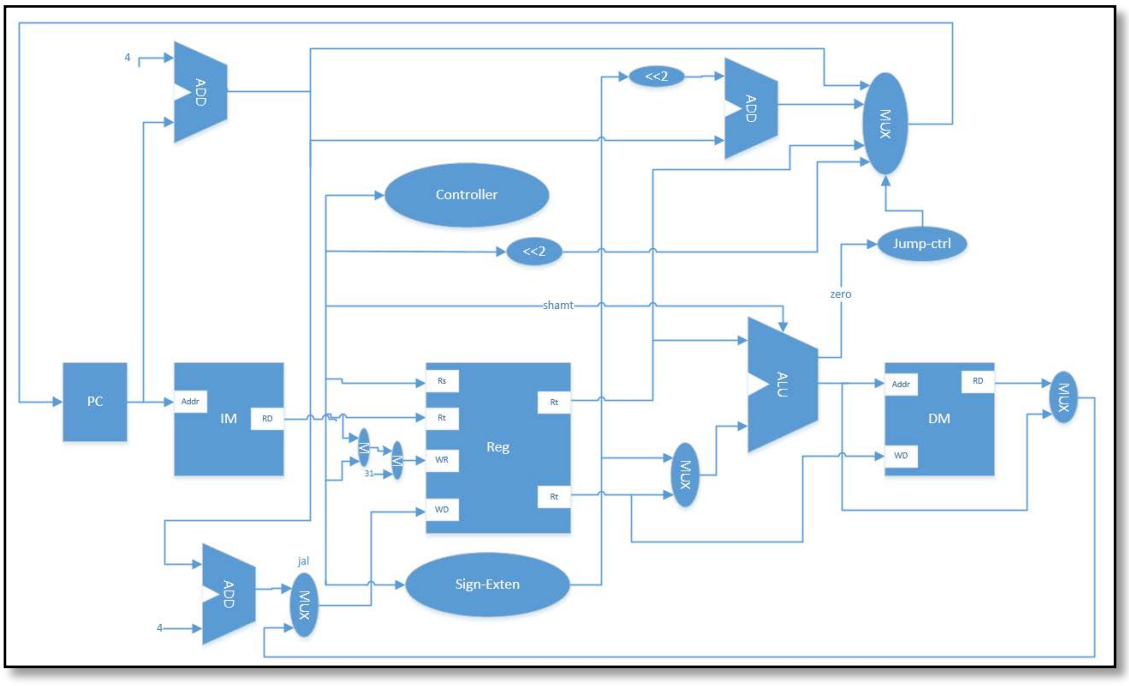


Fig.2 Single cycle CPU DataPath reference

Homework Requirements

1. Please implement these modules:
   I. Controller - **Controller.v**
   II. ALU - **ALU.v**
   III. Sign-Extension - **sign_extend.v**
   IV. Multiplexers - **Mux2to1.v, Mux4to1.v**
2. Verify your CPU with the benchmark and take a snapshot (e.g. Fig.3)

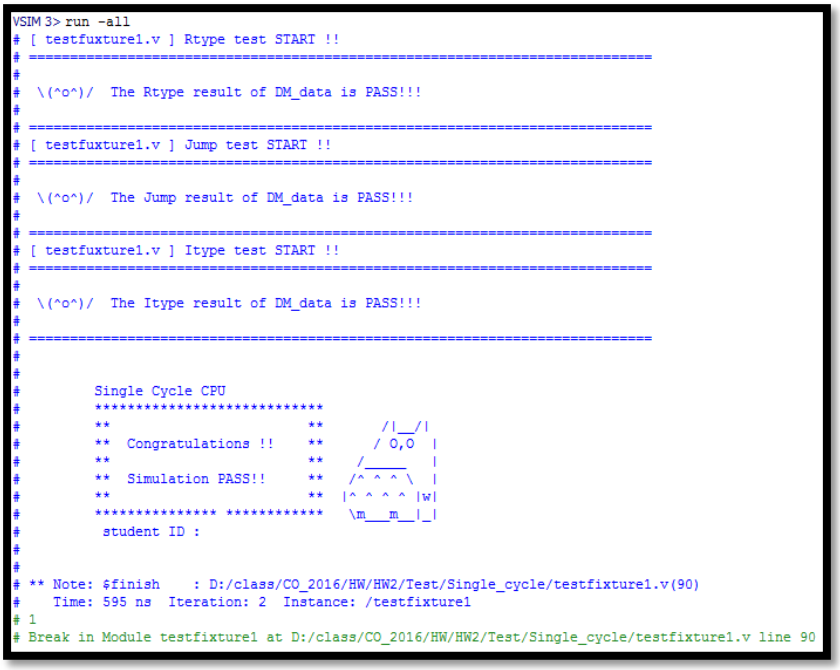


Fig.3 Simulation Successful snapshot

3. Take snapshot
   I. Using waveform to verify the execute results.
   II. Please annotate the waveform (as shown in Fig. 4)

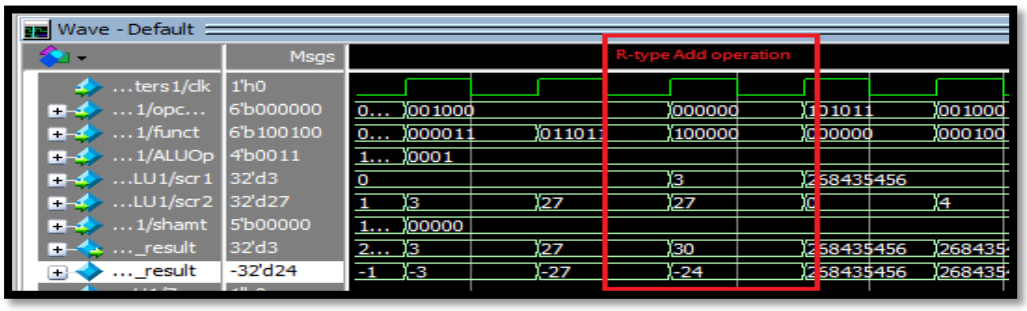


Fig.4 Instruction waveform snapshot

   III. At least list the following instructions in the waveform snapshot. Of course, listing more instructions is better.
      i. sub, and, nor, ,slt, srl, addi, beq, lw, j
4. Finish the Project Report.

5. Bonus
    I. Use Qtspin to compile your HW1 assembly program, and run the compiled machine code on the CPU you have developed in HW2.
    II. Take a snapshot of reg $t0 waveform and put your IM_data in src/tb1 folder as “FIB_IM_DATA.dat”.

TIPS

- **Please refer to the lab2 tutorial and build your project.**

Important

When you upload your file, please check you have satisfied all the homework requirements, including the **File hierarchy**, **Requirement file** and **Report format**.

If you have any questions, please contact us.