Lab. Assignment-6 (a)

Computer Architecture (CS F342)
Semester-I, 2022-23
Department of Computer Science and Information Systems (CSIS)
BITS-Pilani, K K Birla Goa Campus, Goa, India.

Due date: Nov 21, 2022: 2:59 PM Marks: 20

Consider the 32-bit MIPS instruction model (instruction format is given below) with instructions: LW, SW, ADD, ADDI, and BNE. The instructions are executed on Pipelined MIPS microprocessor with full hazard detection unit. For this, one has to define the following modules. There is no need to define other variables and modify the other module except the below in the program.

File Name	Module	Marks
DataFwdUnit.v	forwardingUnit	5
HDU.v	hazardDetectionUnit	3
BTB.v	branchFSM	5
comparator.v	comparator	1
topModule.v	pipelinedDatapath	5
testbench.v	testbench	0.5
For attending the Lab-6 (a) **[attendance needs to be recorded]**		0.5

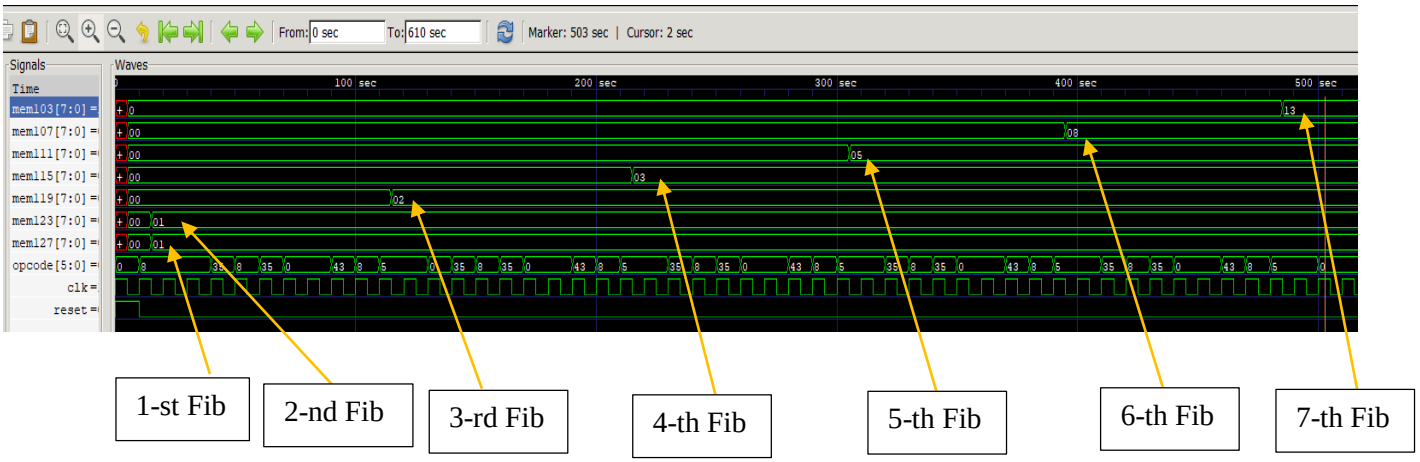
Please find the datapath of MIPS pipelined with hazards detection unit in **pipelined_HDU_MIPS.png**

MIPS has 32 registers and R0 holds zero value only.

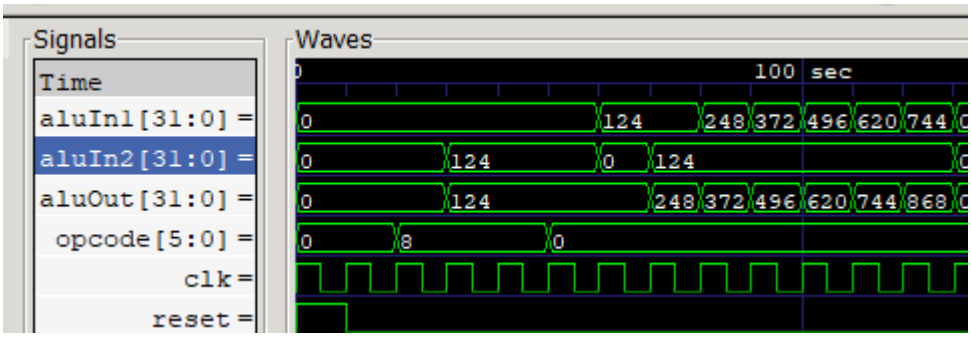
Instruction format:

R-Type	OPCODE						RS					RT					RD					SHAMT					FUNCT				
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I-Type	OPCODE						RS					RT/RD					IMMEDIATE														
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

The actual branch decision is taken from ID stage and predicted branch decision taken from IF stages. For predicting the branch decision, a 2-bit saturation counter, with initial state 2'b11, is used in BTB. In the template, a test program, stored in the memory, generates 7 Fibonacci numbers. The program is written using the above-mentioned instructions. NOP instruction is not used in the program. The first two Fibonacci numbers are stored in the memory at locations 124 and 120. The program calculates the first seven Fibonacci numbers, and the remaining five Fibonacci numbers can be stored in locations 116 onwards. The Big-endian format is considered to store the instruction and data. Memory is byte-addressable. The memory contains 128 locations. However, the program counter (PC) is 32-bit. Some bits of the PC are used to identify the memory locations. Similarly, for ALU's output. The program generates the following outputs:



To test the complicated data hazard, copy the **memInstr.v** with **copy_memInstr.v** and rename the **compMemInstr.v** with **memInstr.v**. Run the simulation and get the following output.



<u>Instructions for uploading the results</u>:

Download the *CA_Lab6a_Pipelined_HDU_MIPS_Template.zip* and rename the folder as <*CampusID*>_ *CA_Lab6a_PipelinedHDU_MIPS.* Output must be stored as <*CampusID*>_ *CA_Lab6a_PipelinedHDU_MIPS_output1.png* and <*CampusID*>_ *CA_Lab6a_PipelinedHDU_MIPS_output2.png* in the folder. Create a zip file <*CampusID*>_ *CA_Lab6a_PipelinedHDU_MIPS.zip* which contains all the file mention above and submit it in Quanta. Do not create archives in other formats (rar, tar.gz etc). Once uploaded on Quanta, remember to submit for grading. Do not leave it as a draft.

/*##

Note: Please don't upload the assignments, template file/solution and lab. manual on GitHub or others public repository.

Kindly remove them, if you have uploaded the previous assignments.

It violates the BITS's Intellectual Property Rights (IPR).

**/