Lab. Assignment-5(a)

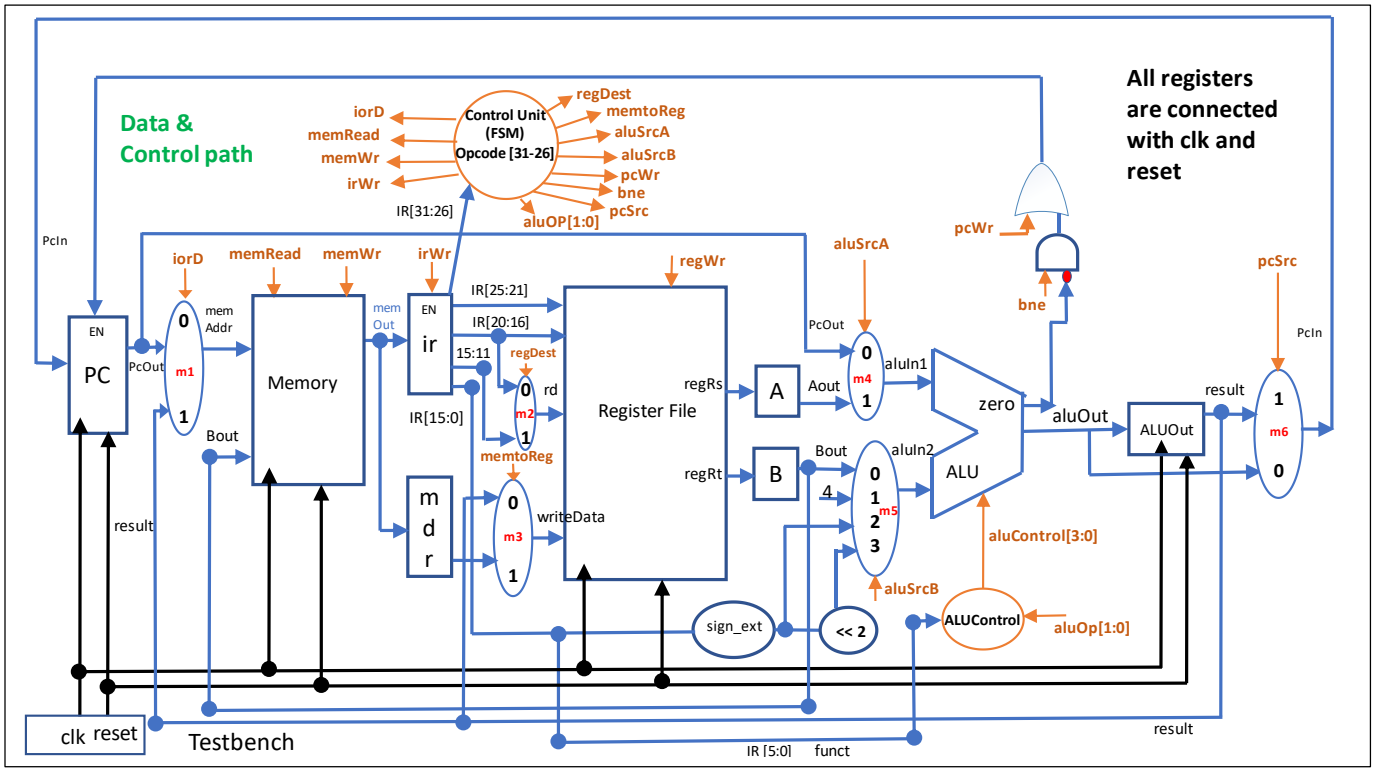
Computer Architecture (CS F342)
Semester-I, 2022-23
Department of Computer Science and Information Systems (CSIS)
BITS-Pilani, K K Birla Goa Campus, Goa, India.

Due date: Oct 24, 2022: 3 AM Marks: 6

Consider the 32-bit MIPS instruction model (instruction format is given below) and, using Verilog HDL language, simulate the MIPS's instructions: LW, SW, ADD, ADDI, and BNE. 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
A.v	A_Reg	0.2
B.v	B_Reg	0.2
MDR.v	MDR_Reg	0.2
IR.v	IR_Reg	0.2
ALUOut.v	ALUOut_Reg	0.2
mux2_1_Data.v	mux2to1_32bits	0.2
mux4_1_Data.v	mux4to1_32bits	0.2
controller.v	controlCircuit	1.2
topModule.v	MIPS_multiCycle	1.7
testbench.v	testbench	0.2
For attending the Lab-5(a) **[attendance needs to be recorded]**		0.5

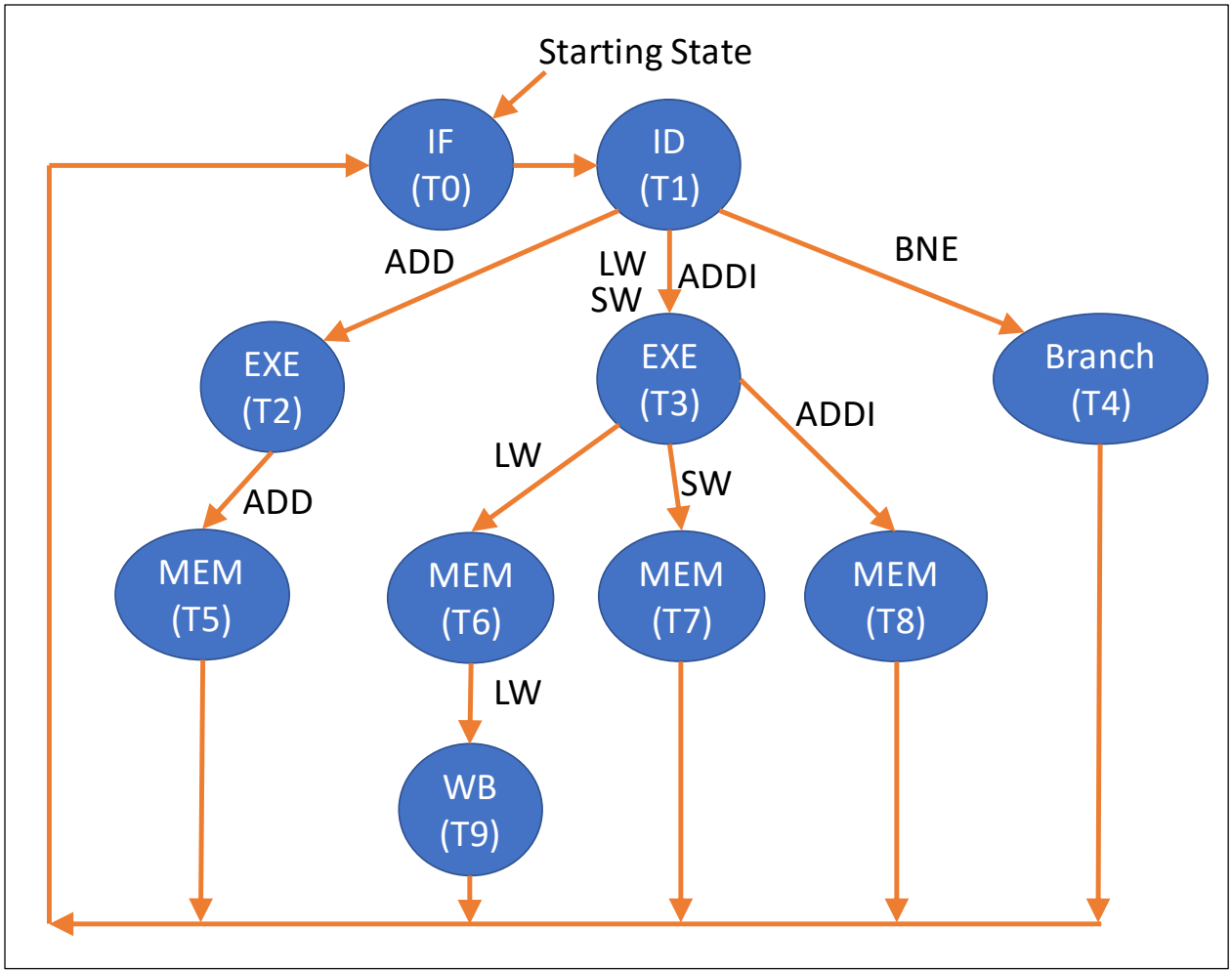
MIPS's multicycle model



Instruction format:

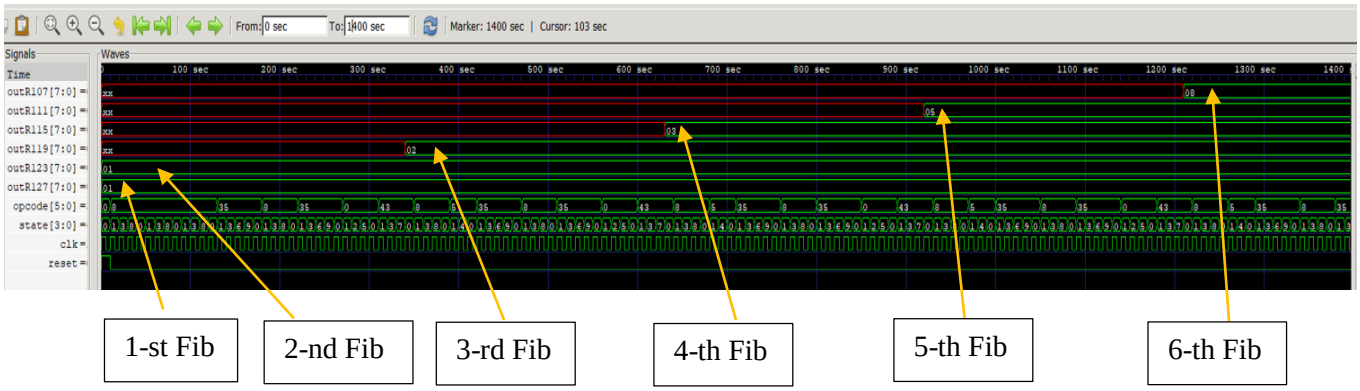
R-Type	OPCODE						RS					RT					RD					SHAMT					FUNCT					
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I-Type	OPCODE						RS					RT/RD					IMMEDIATE															
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

States' information for the Instructions: LW, SW, ADDI, ADD, and BNE:



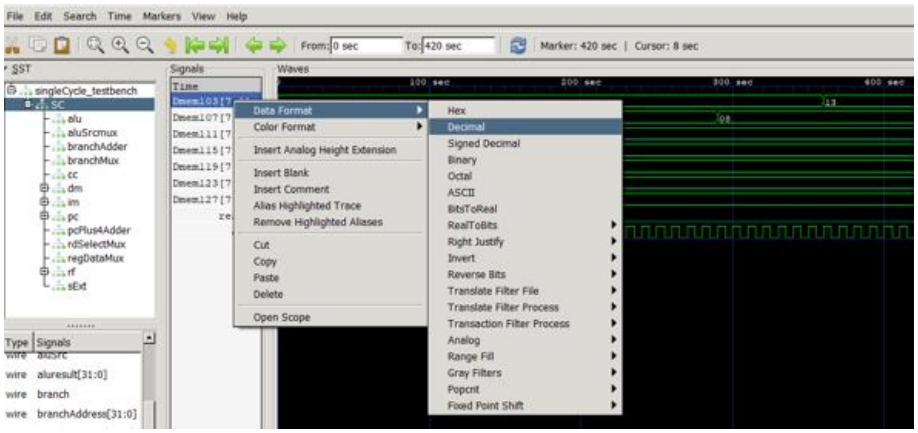
In the controlCircuit() module, the above-mentioned states are already defined. One has to use them properly to get the following output.

In the template, a test program, stored in the memory, generates 6 Fibonacci numbers. The program is written using the above-mentioned instructions. The first two Fibonacci numbers are stored in the memory at locations 124 and 120. The program calculates the first six Fibonacci numbers, and the remaining four 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. The program generates the following outputs:



****Better version of the image can be found in PPT.**

Following is changing procedure for signals' data format as decimal:



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

Download the *CA_Lab5a_MultiCycleMIPS_Template.zip* and rename the folder as *<CampusID>_CA_Lab5a_MultiCycleMIPS*. Output must be stored as *<CampusID>_ MultiCycleMIPS_output.png* in the folder. Create a zip file *<CampusID>_CA_Lab5a_ MultiCycleMIPS.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.

/*##

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).

**********/