Branch

To simulate data path for Branch instruction. Integrate Program Counter (PC), Instruction Memory (IM), Instruction Decoder (ID), Register File (RF), Arithmetic Logical Unit (ALU), Data Memory (DM) module developed in previous project. We are defining Branch Control Unit as well. Below are the steps involved:

a. Modify Sign Extend Unit to support branch instructions and define Branch Adder Unit, ShiftLeft2 Unit and Branch Mux Unit
b. Integrate these unit with previous lab unit to support branch data path.

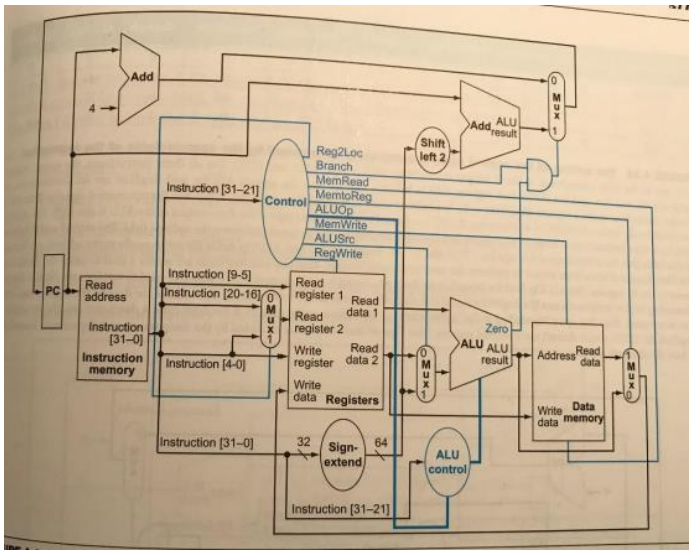


Figure shows the data path for R ,D and Branch type instruction.

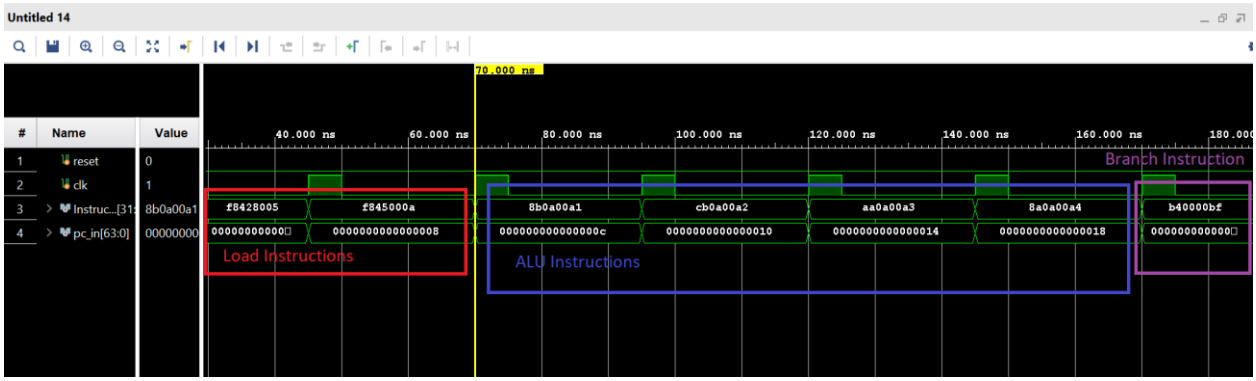
Operation parameters:

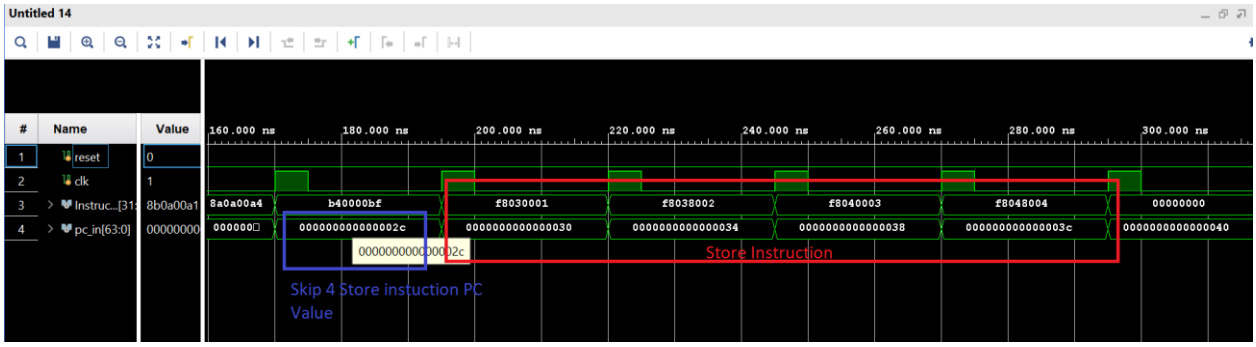
- Input clk 1 bit
- Input reset 1 bit

Module Name: PCIMIDwithRFALUDM

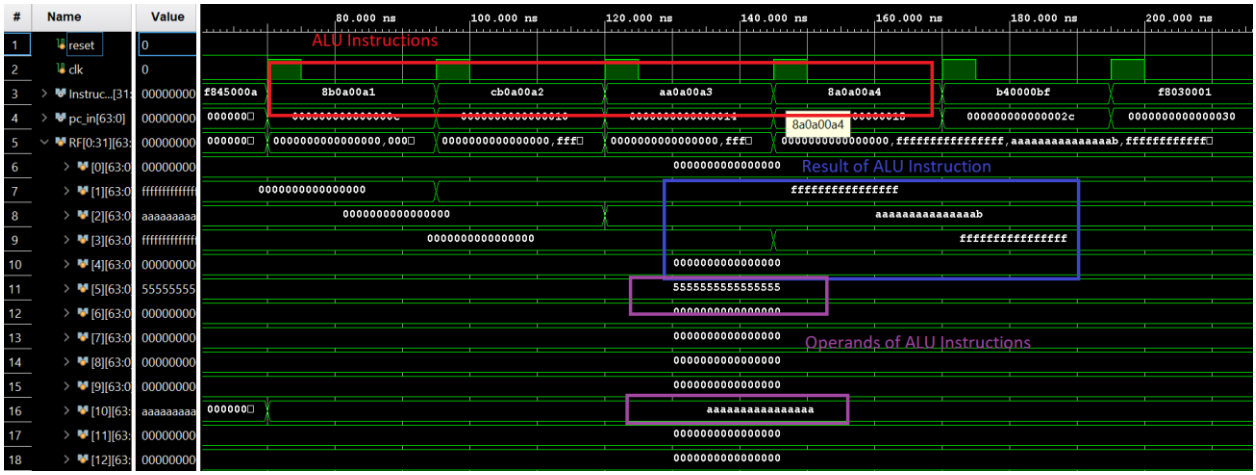
Test Module Name: PCIMIDwithRFALUDM_tb

Below screen shot show the waveform for parameters of datapath:

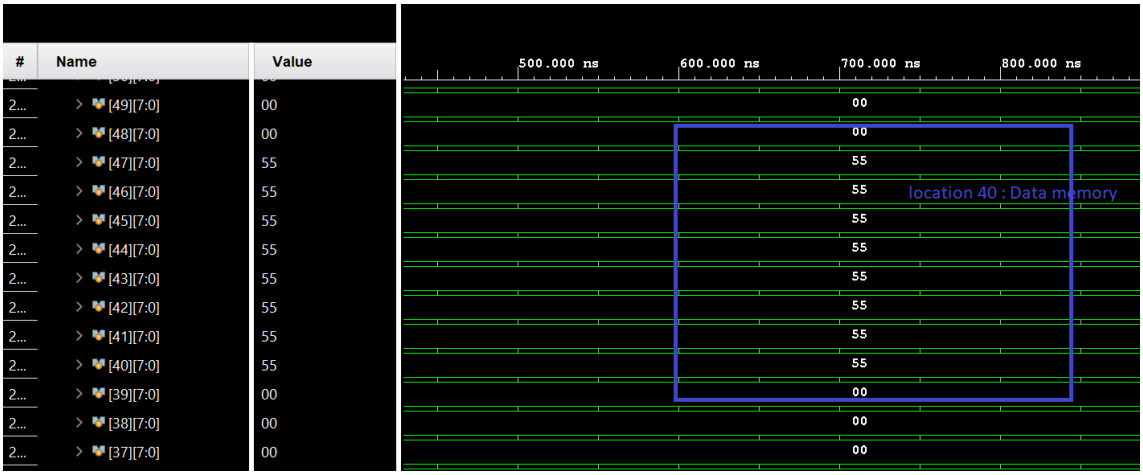


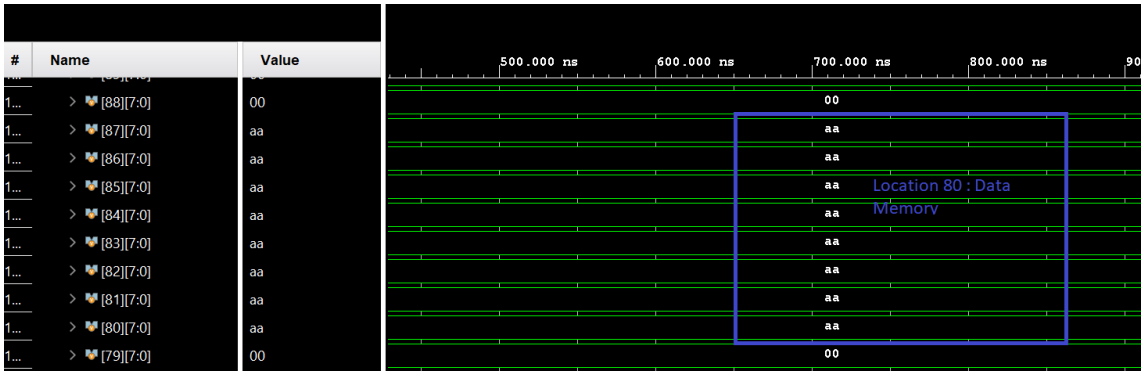


Here we are storing 15 instruction into the instruction memory and using program counter unit we are reading one instruction in each cycle. These are load, store, Arithmetic and Branch instructions. If you see due to Branch instruction, we skip 4 instructions execution.

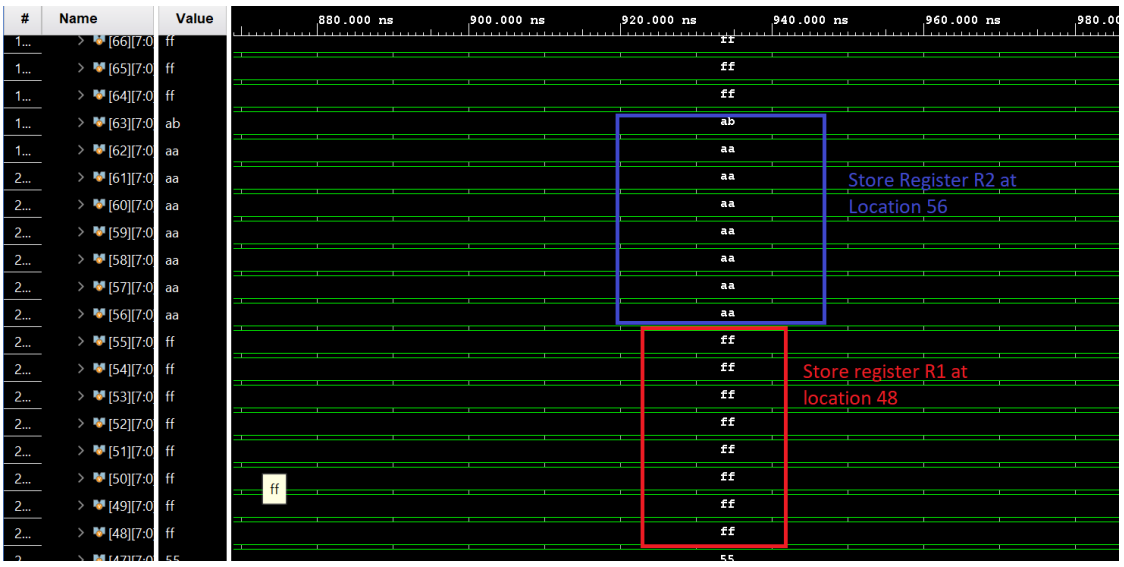


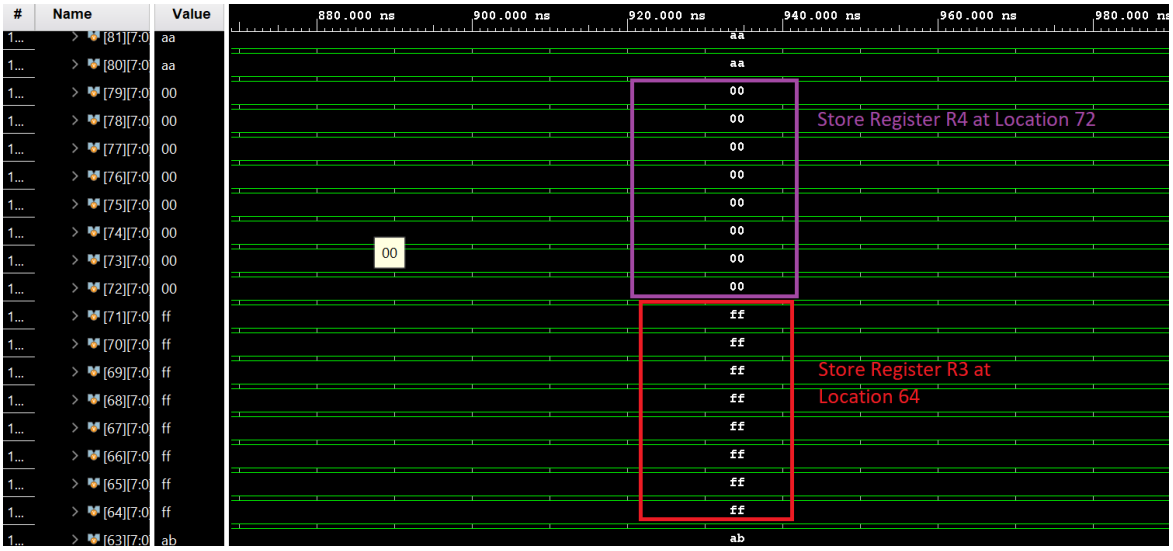
Here the Operands of Arithmetic instruction are read from registers and results are stored back to registers. Operands are in register 5, 10 while the arithmetic results are in register 1,2,3,4 where the results are of ADD, AND, OR, SUB operations respectively. This will support R type instruction.





Here we can see location 40 and 80 in Data memory. These values are loaded to register 5 and 10 in register file to perform arithmetic operations. This will support load Instructions.





Here we are storing register 1,2,3,4 back to data memory at location 48,56,64,72 respectively. This will support store instruction. Also, we are skipping the instruction which are storing register 1,2,3,4 to data memory at location 8,16,24,31 respectively using branch instruction.