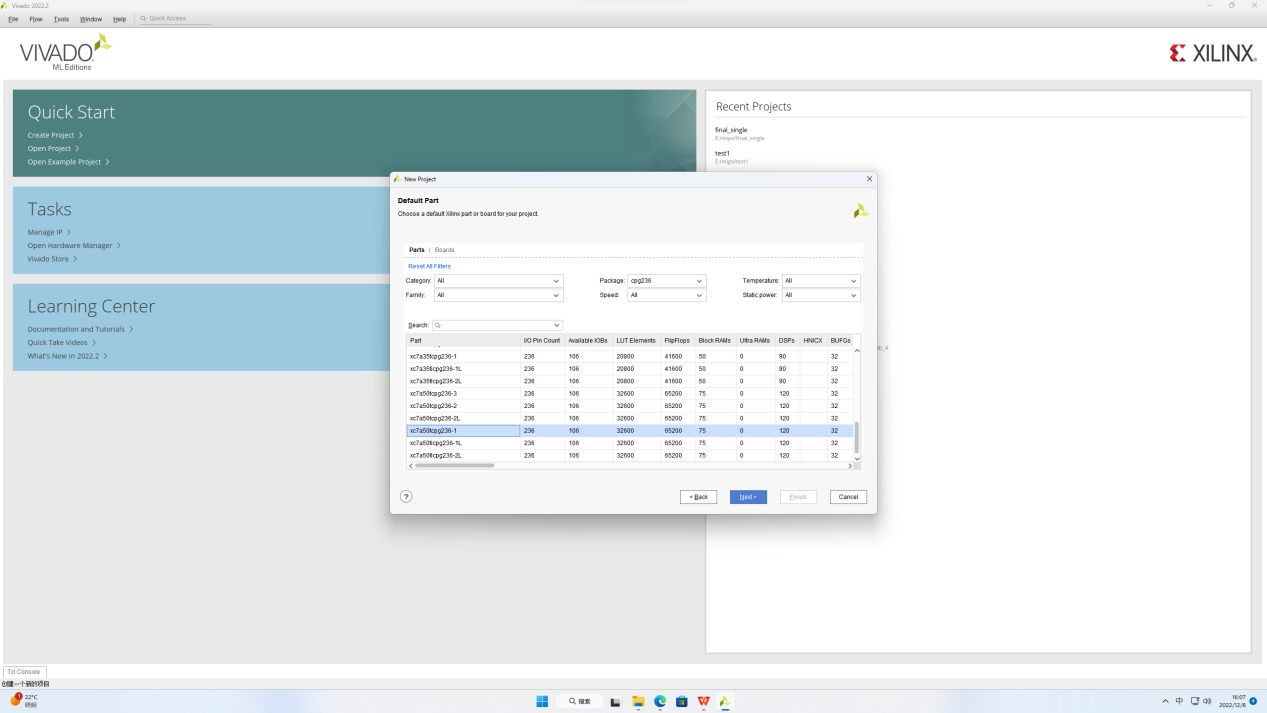
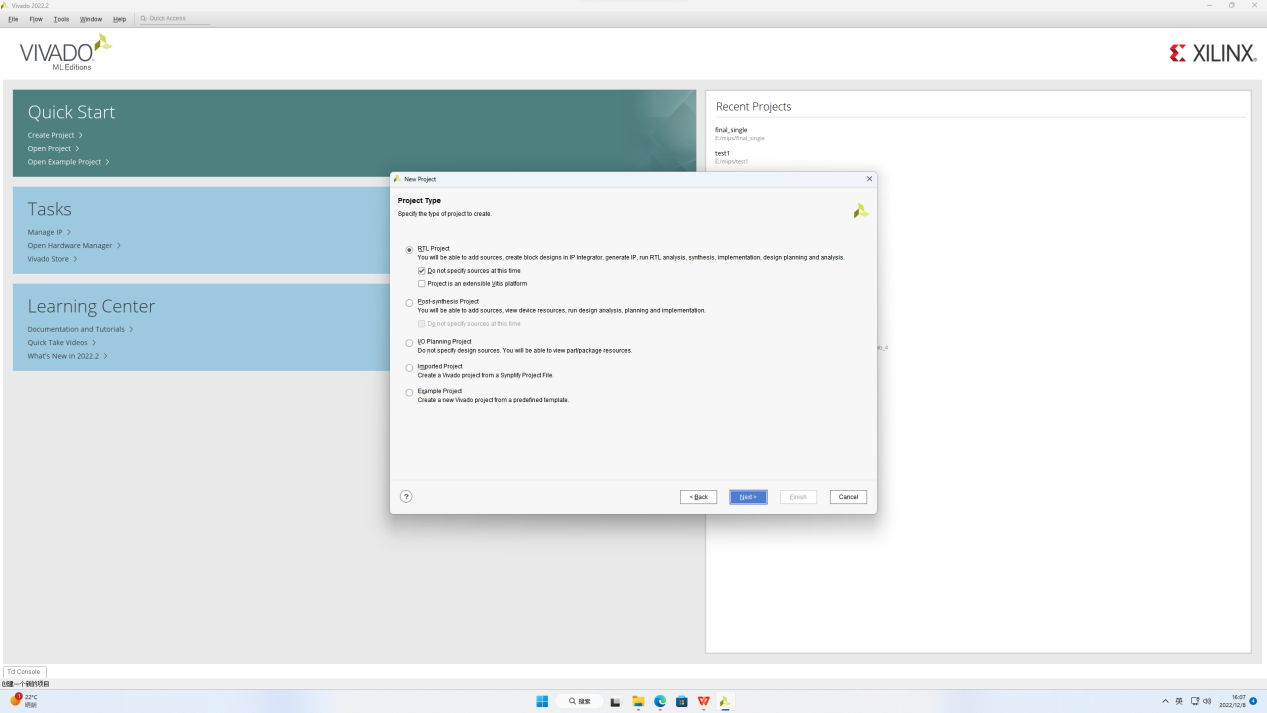
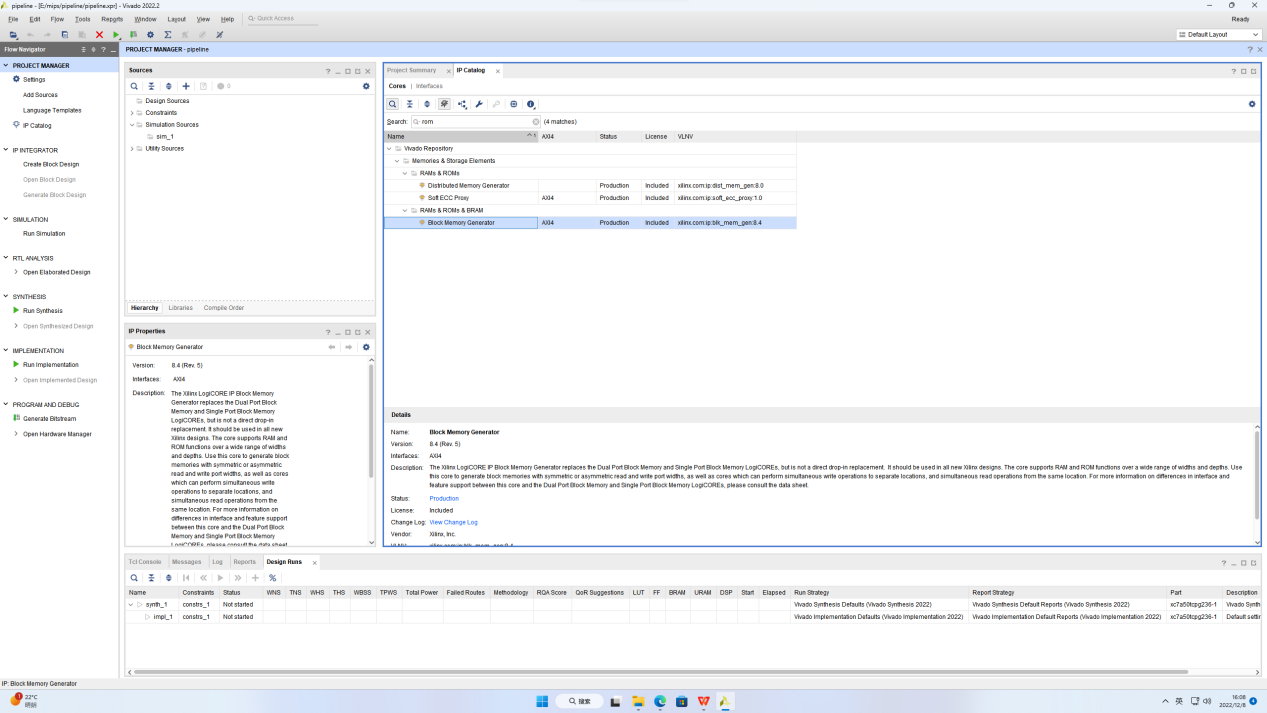
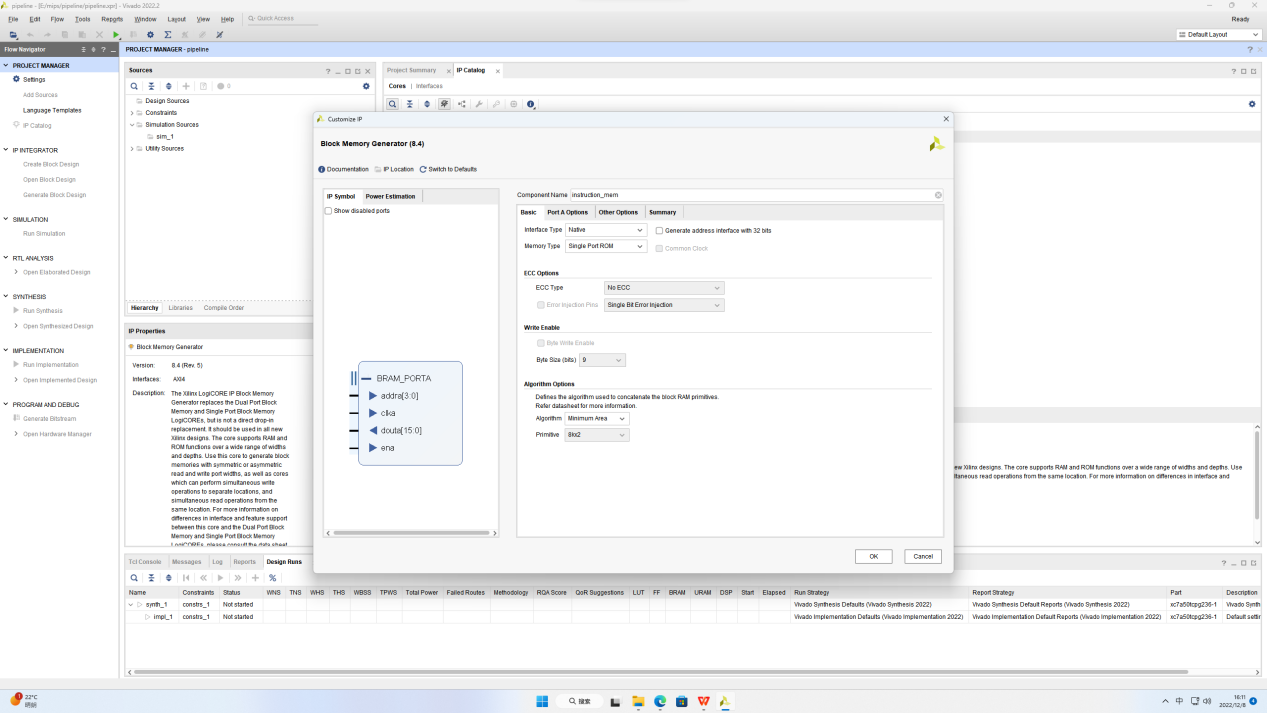
New project:



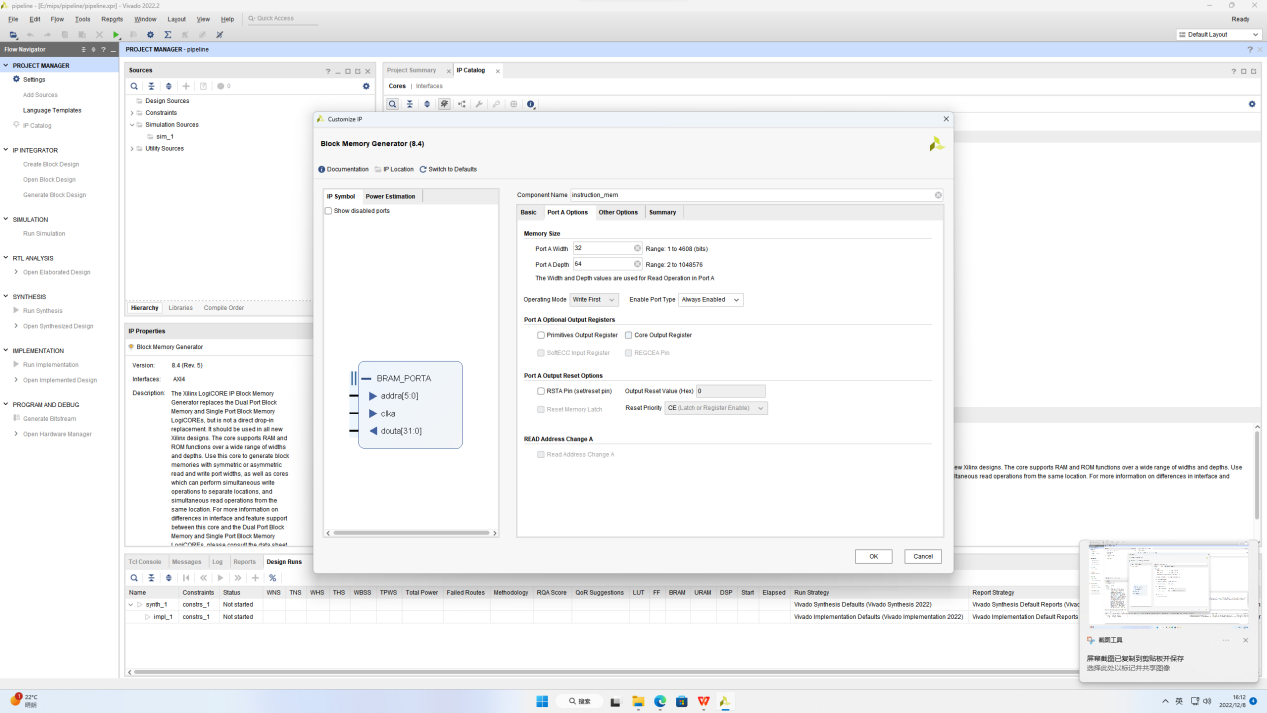
Generate IP CORE:

Instruction\_mem: use single port Rom because we use coe document to initialize instruction memory and there are no needs to write instruction\_mem;

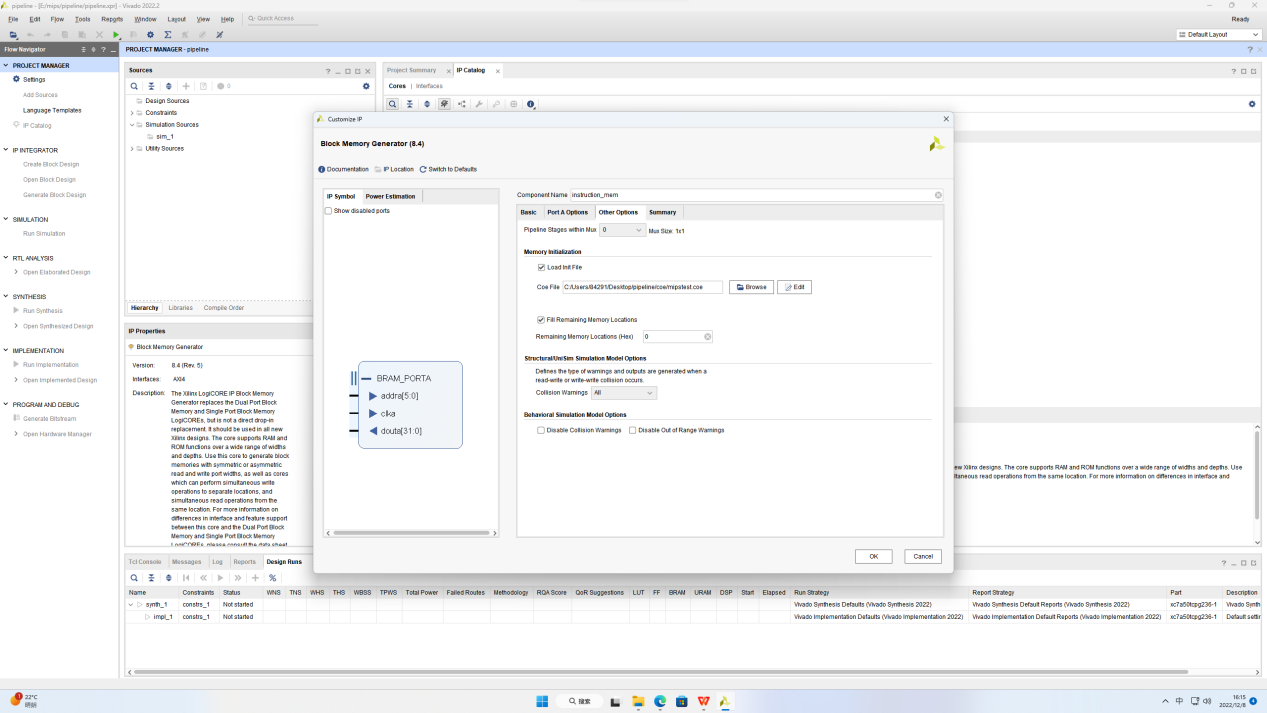


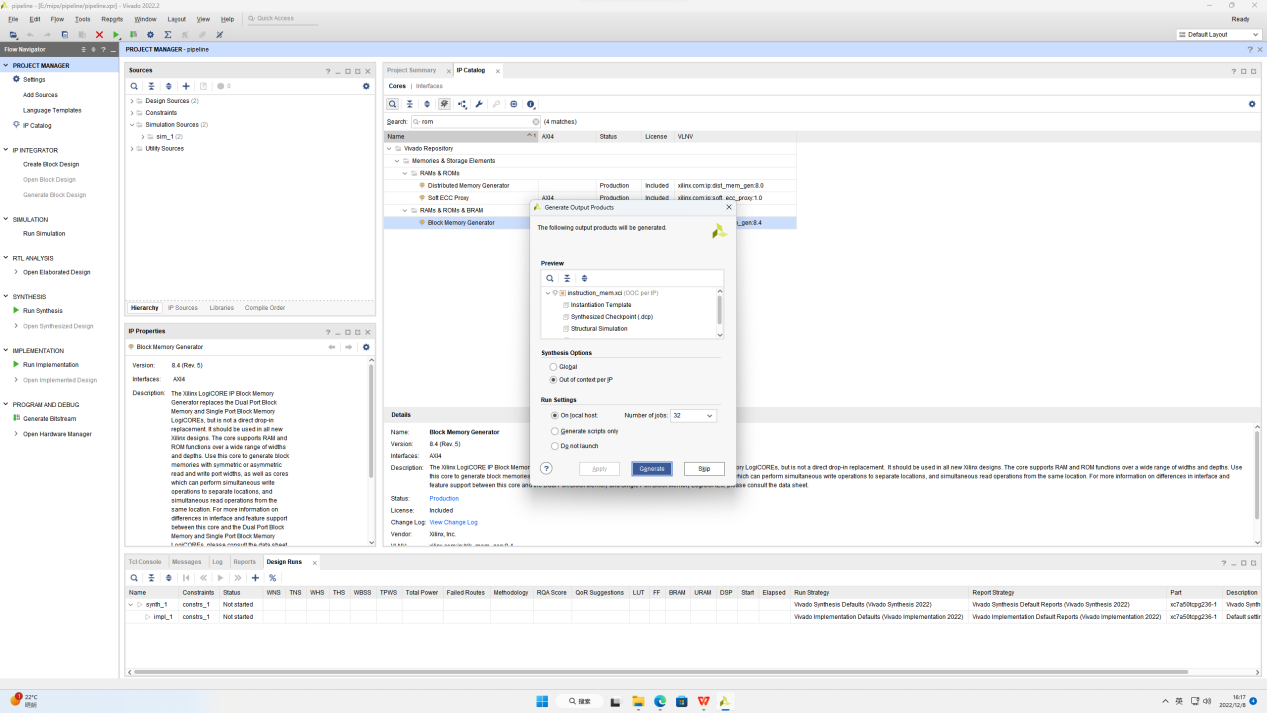


Don’t choose Primitives Output Register because this will make instruction\_mem take an extra cycle to output and we need to set the Enable Port Type to Always Enable thus reducing one of the required enable signals.

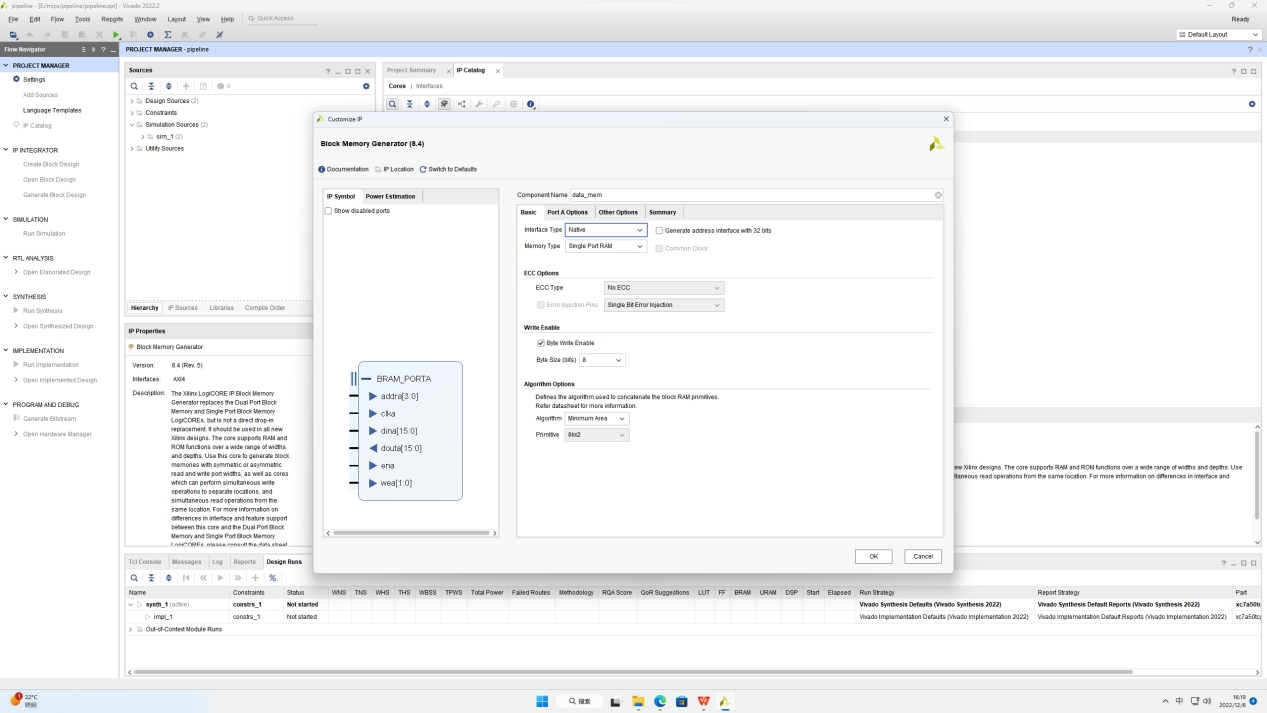


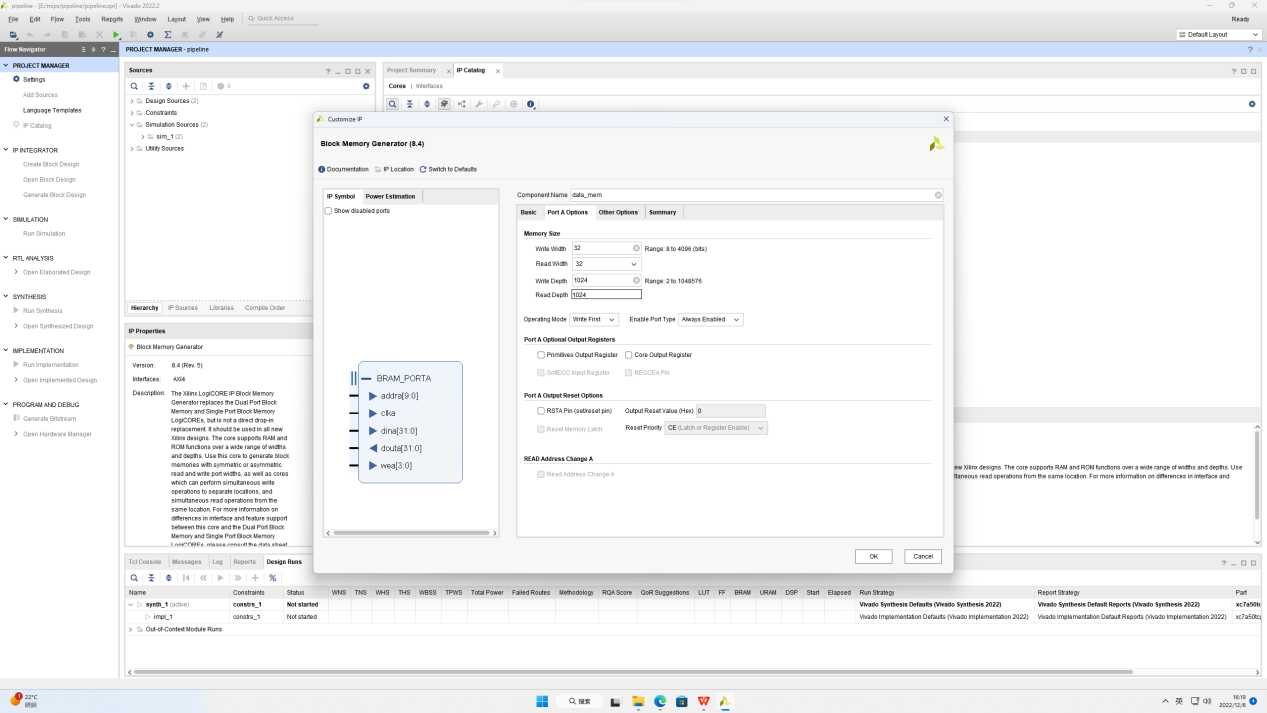
Load Init File using coe file;

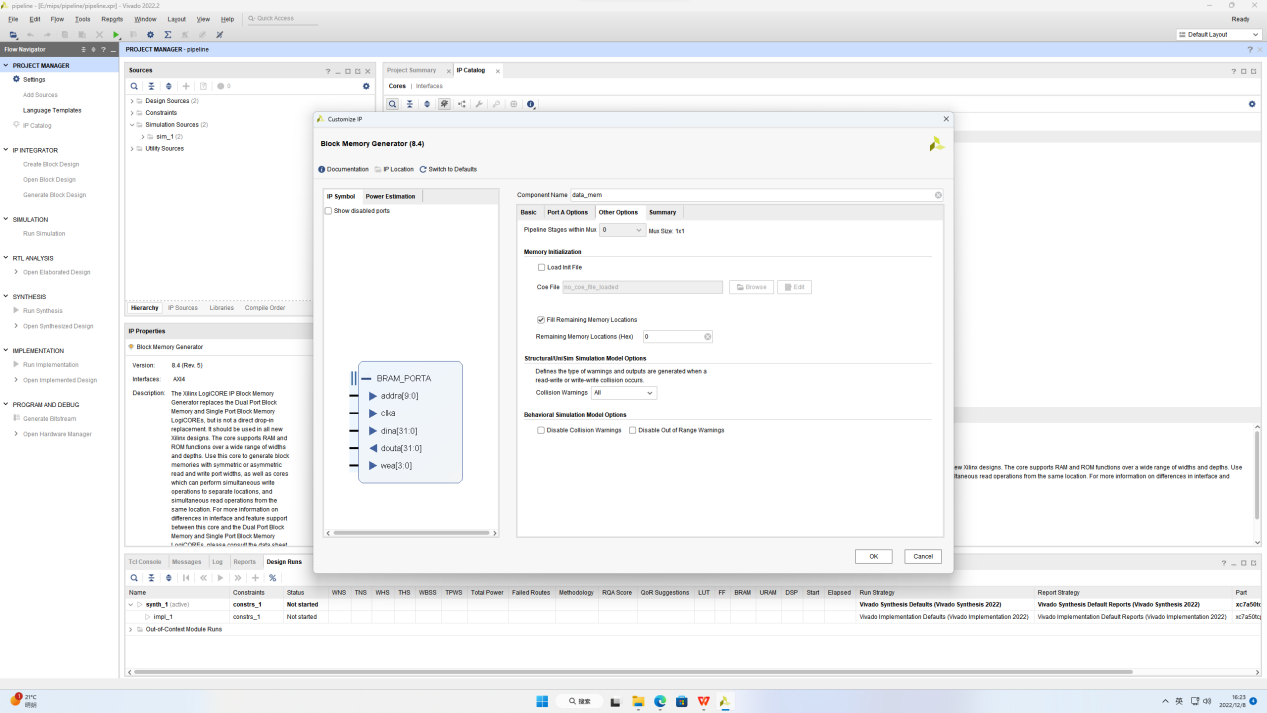




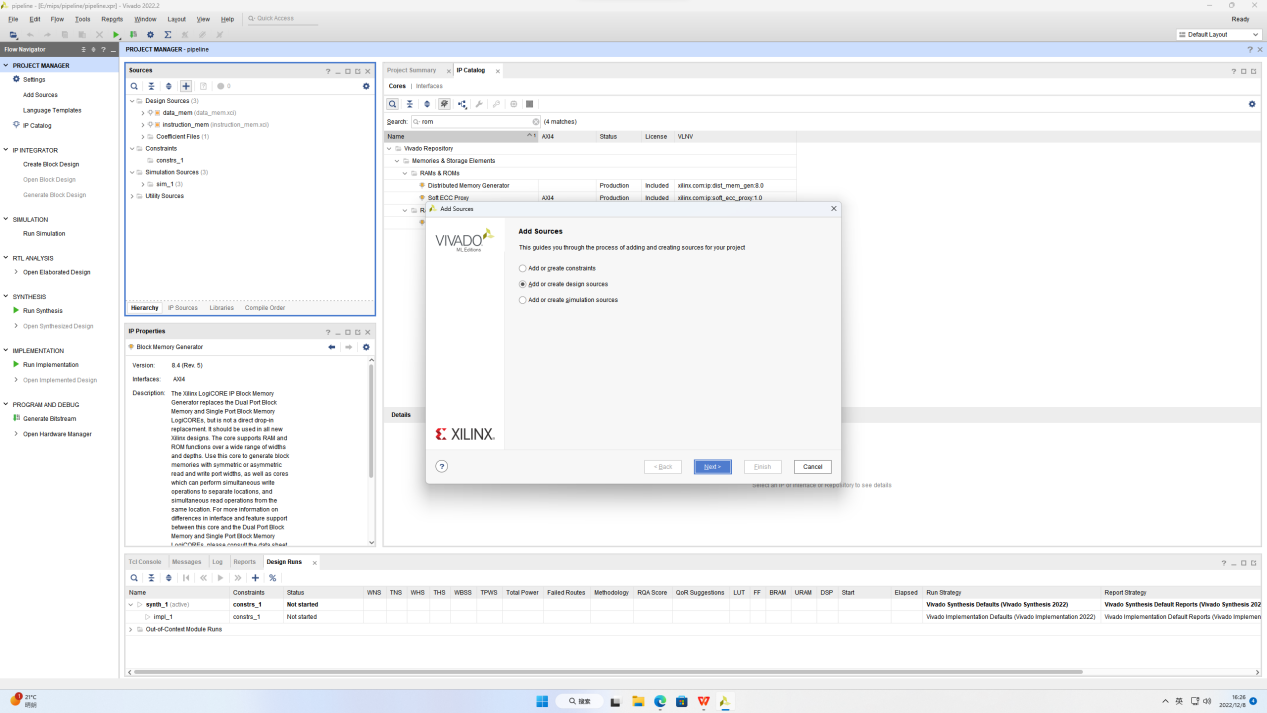
Data\_mem: Use single port ram because we need to write data\_mem;

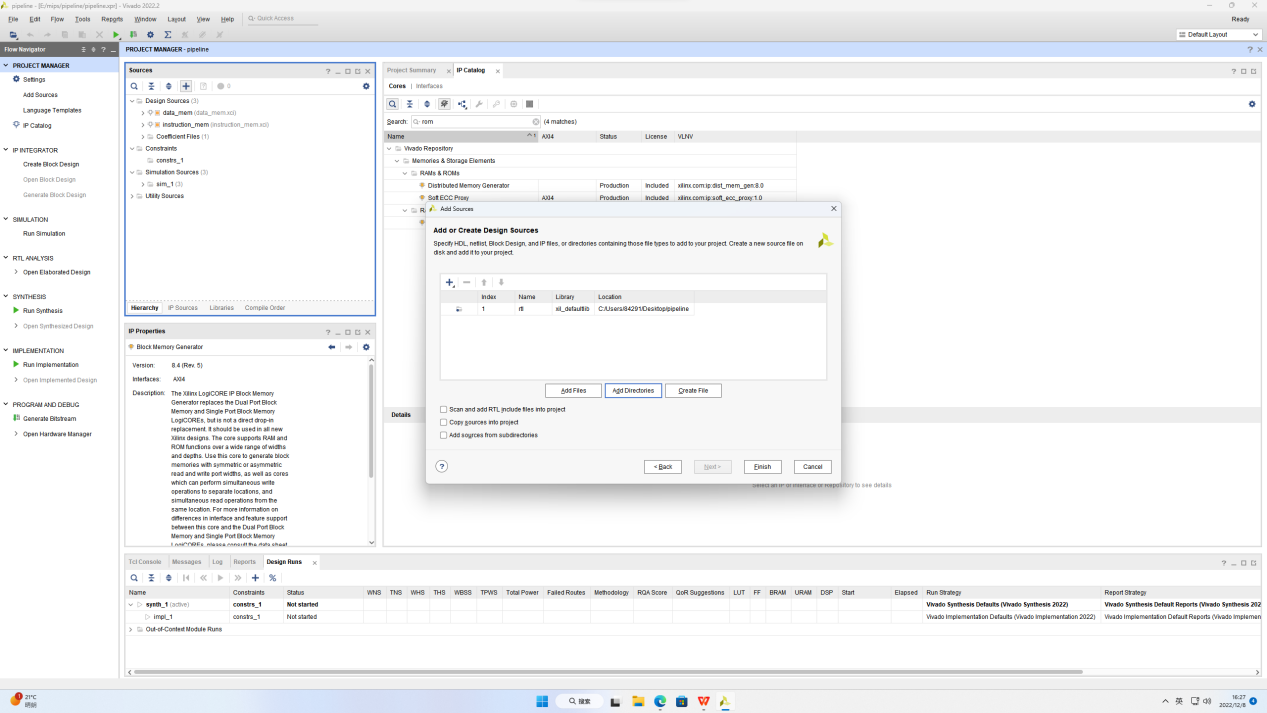






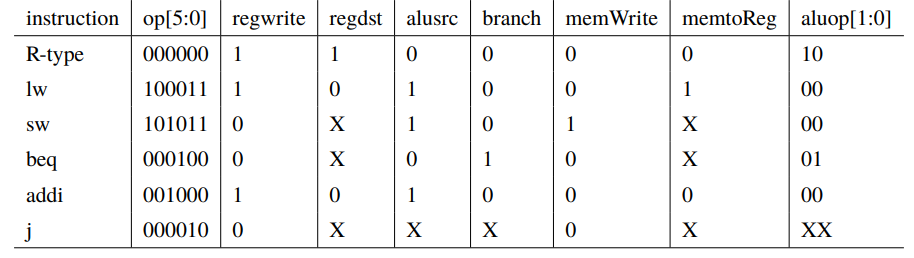
Add design source:



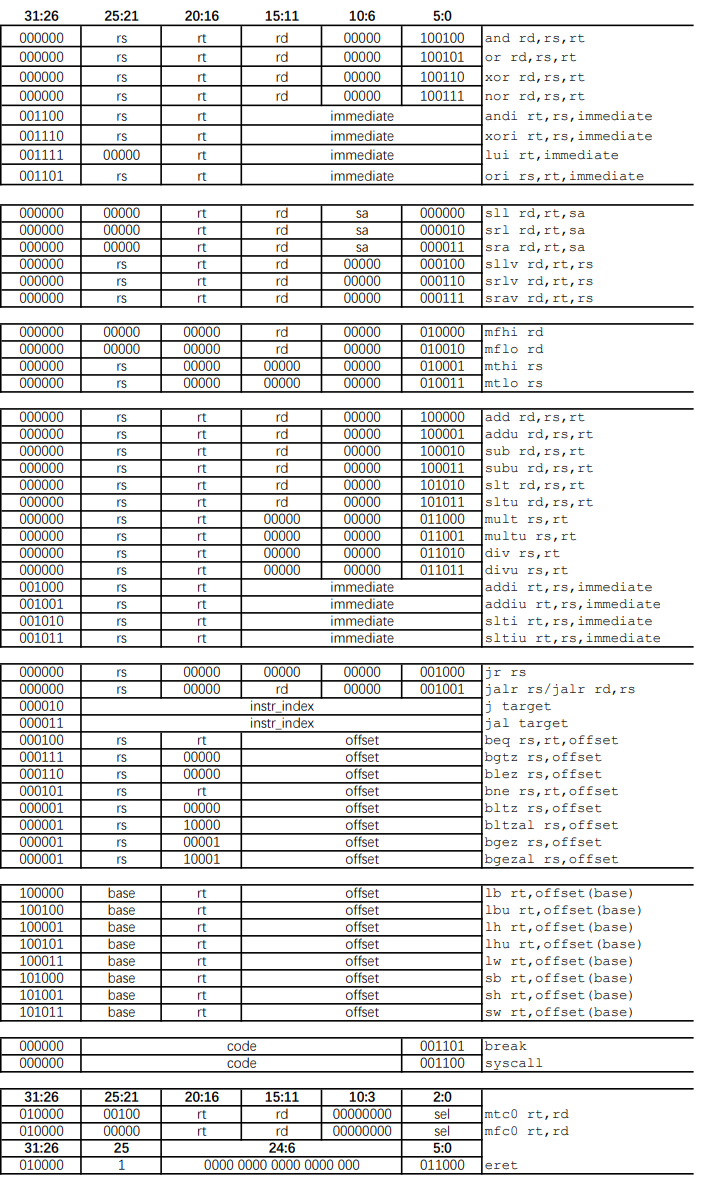


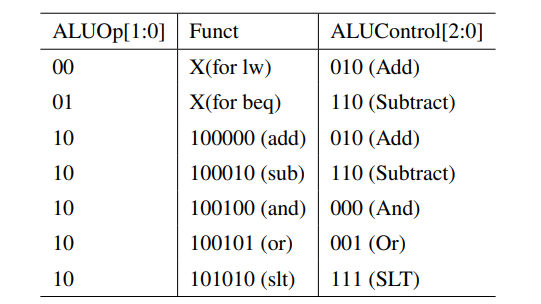
Alu and control:

The instructions involved:

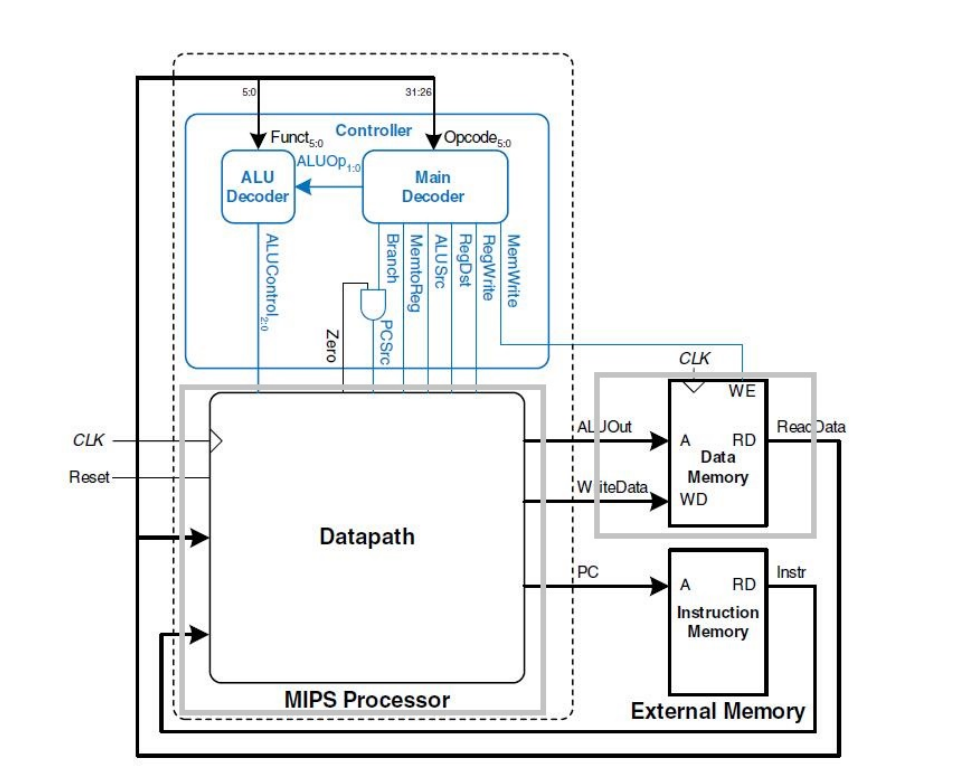


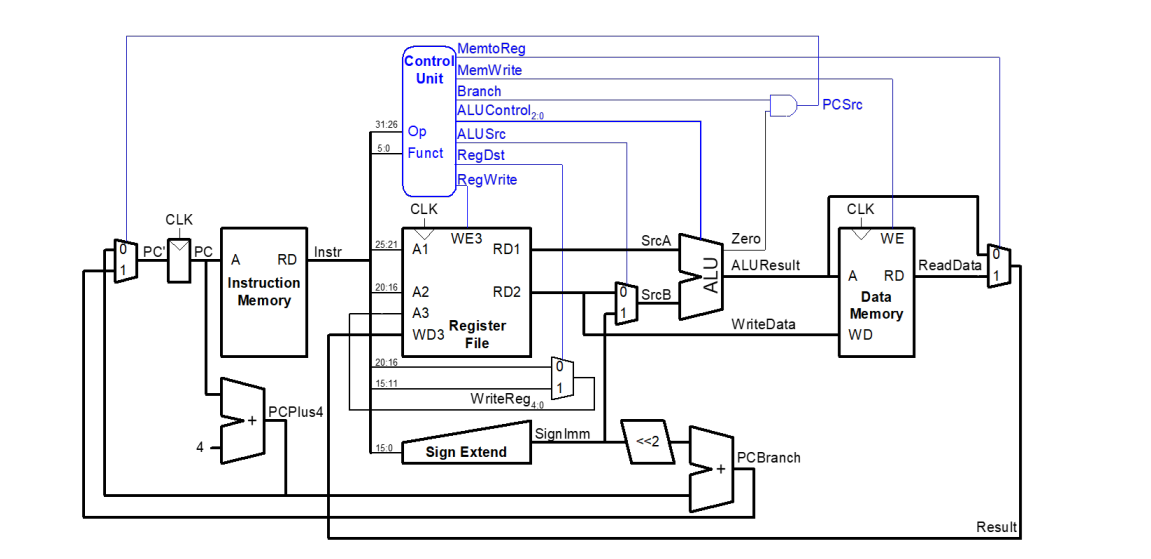
MIPS instruction machine code:

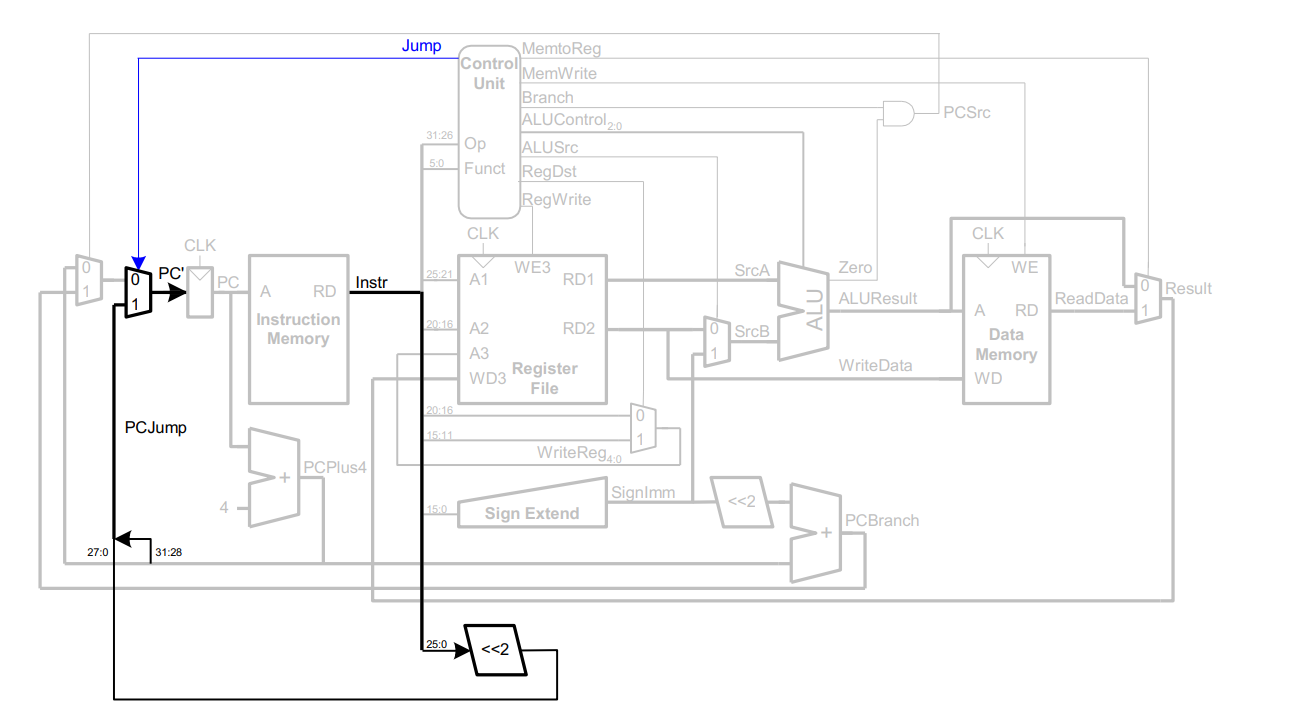




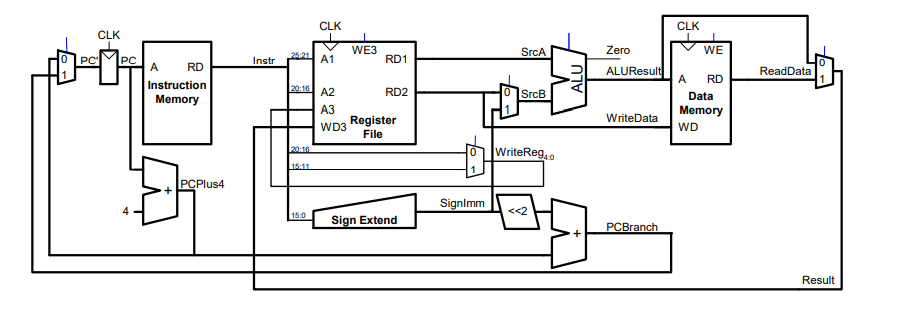
Single-cycle:

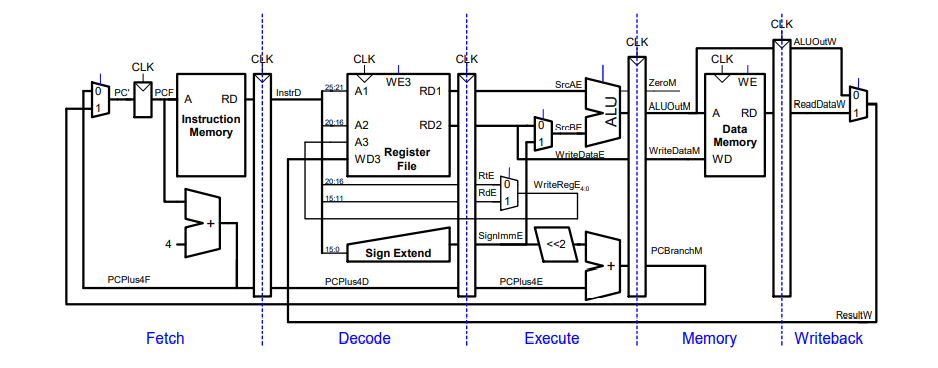


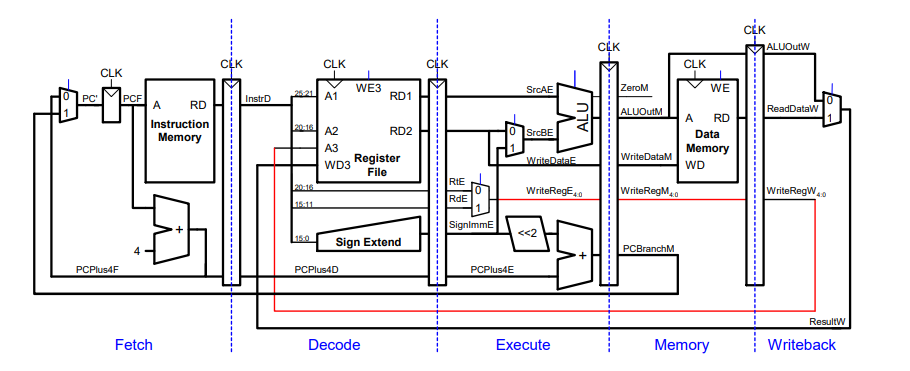


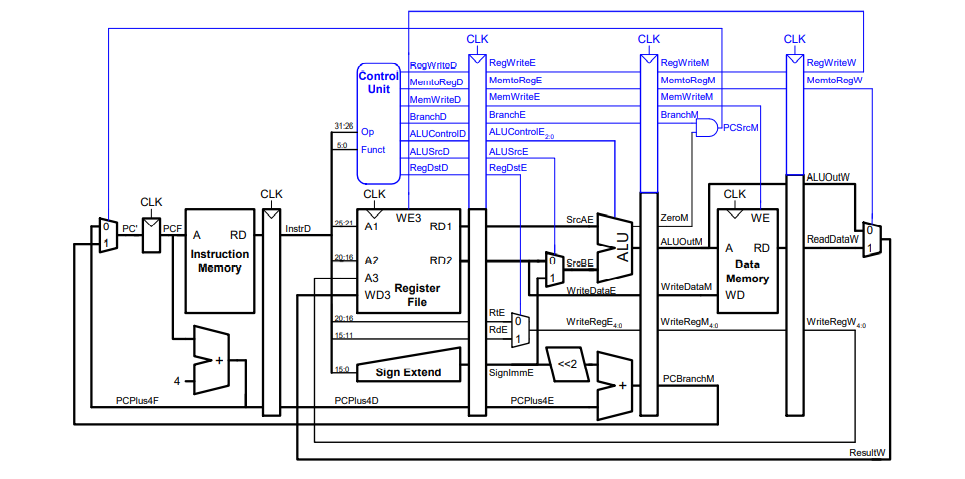


Single cycle to pipeline:



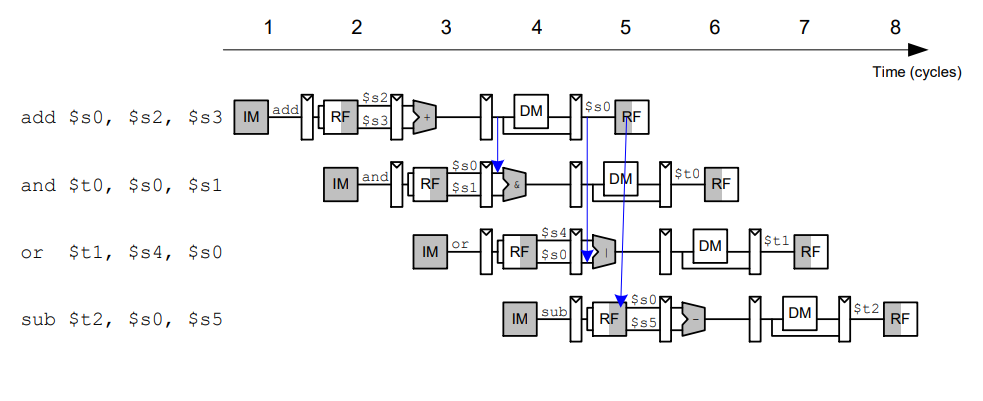
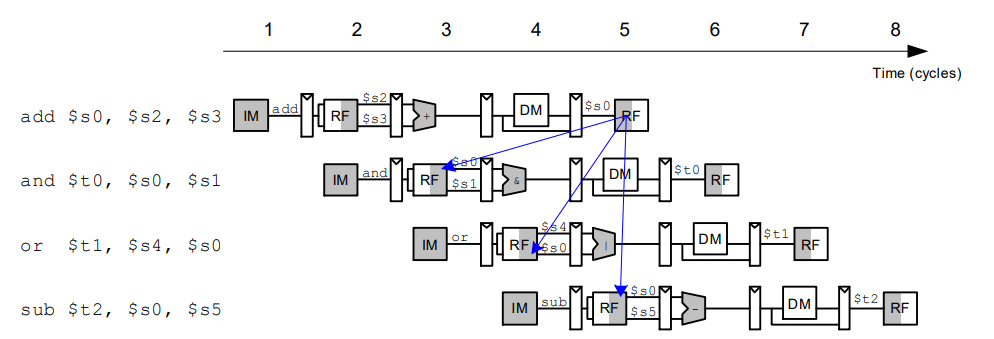




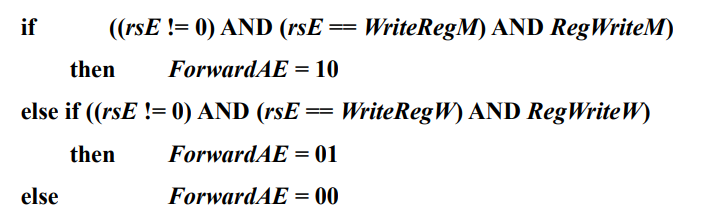


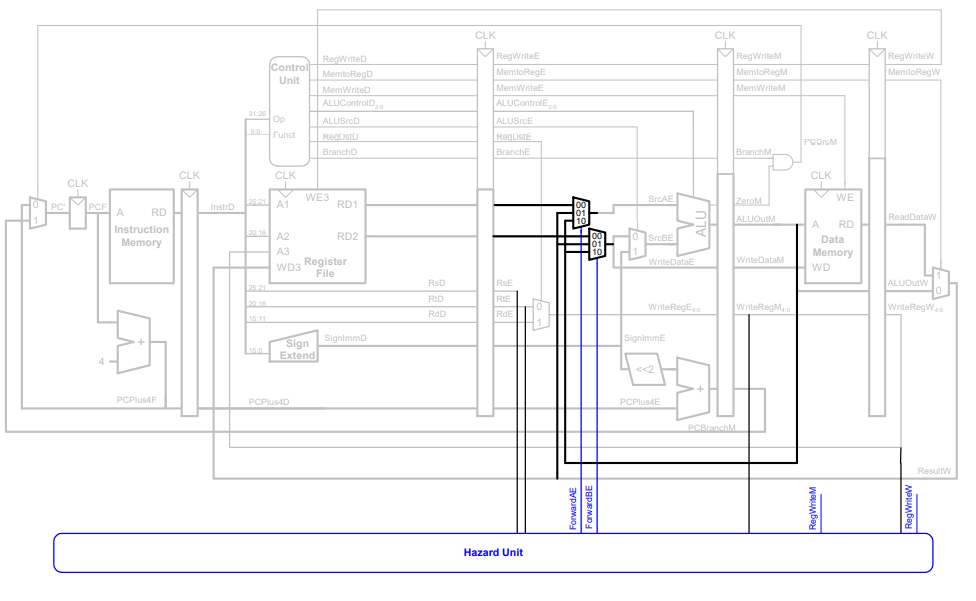
The structural hazard is solved by Harvard Architecture;

Data hazard:

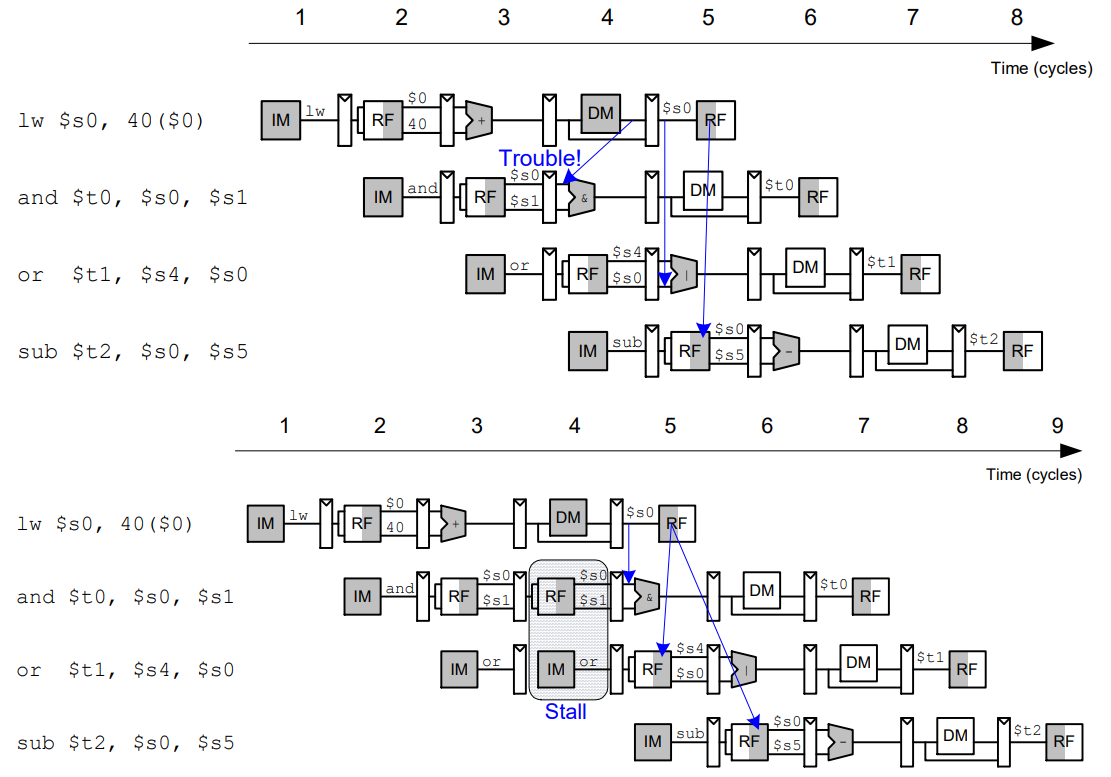


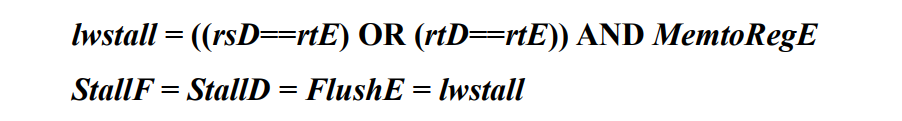
When the rd in the EX/MEM pipeline register and the MEM/WB pipeline register is the same as the source operand in the EX stage, the rd in the EX/MEM stage is relatively new and should be used preferentially:

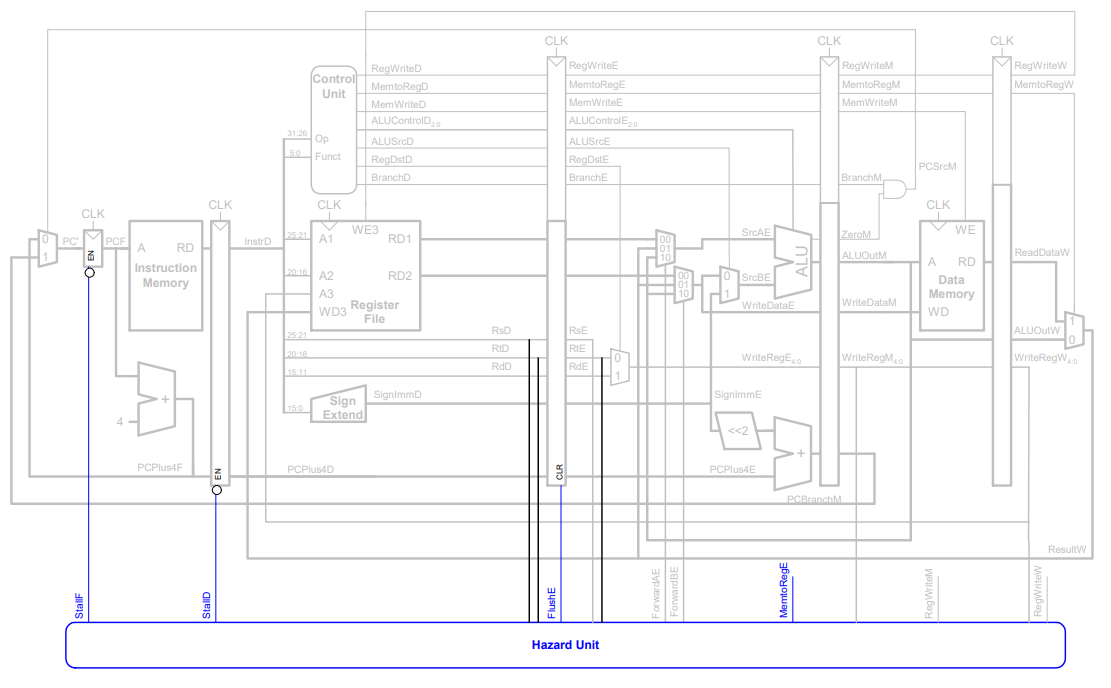




Pipeline stall:

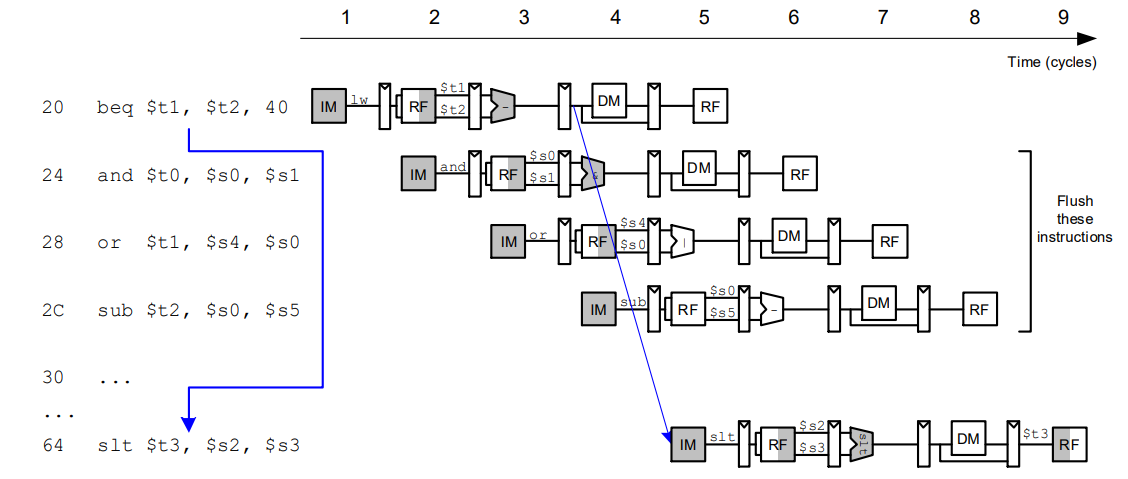


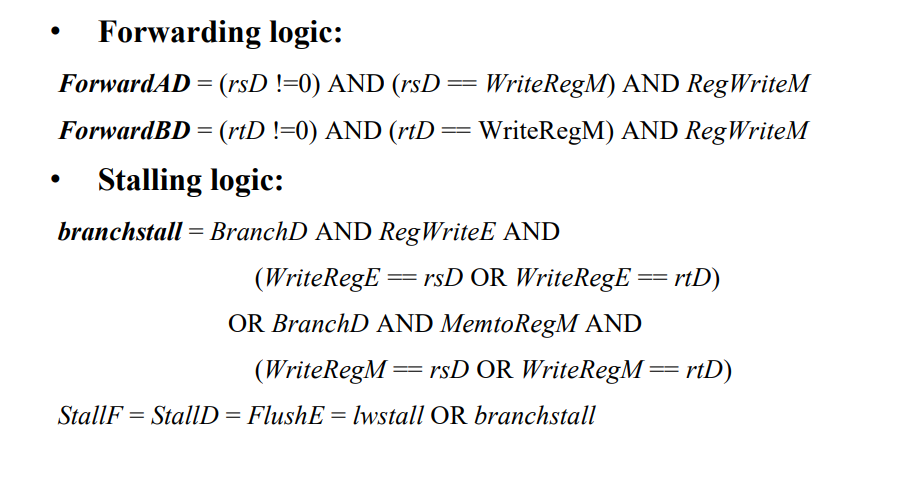
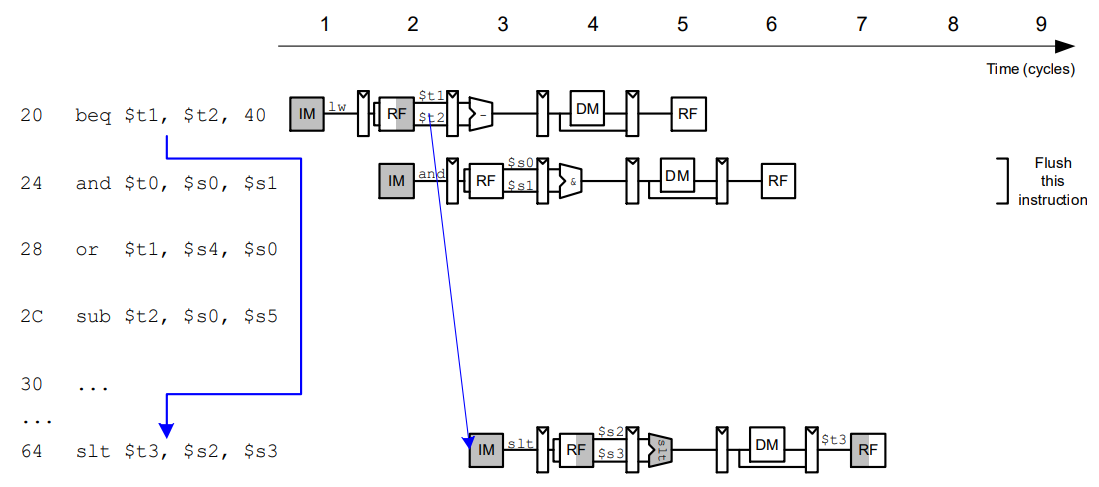


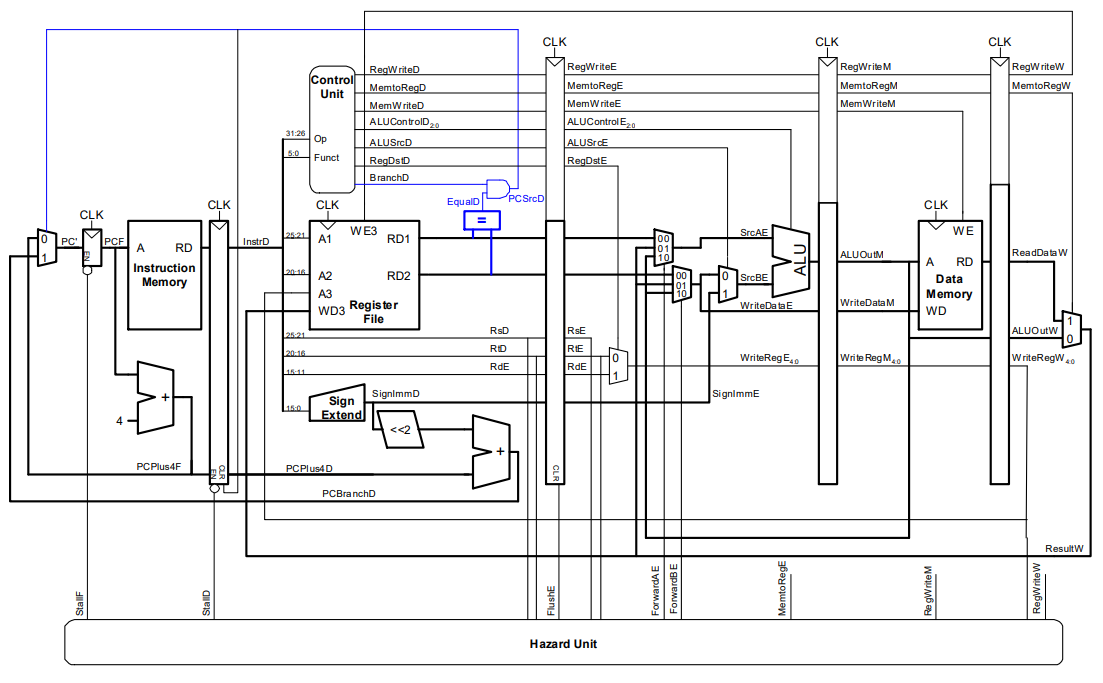


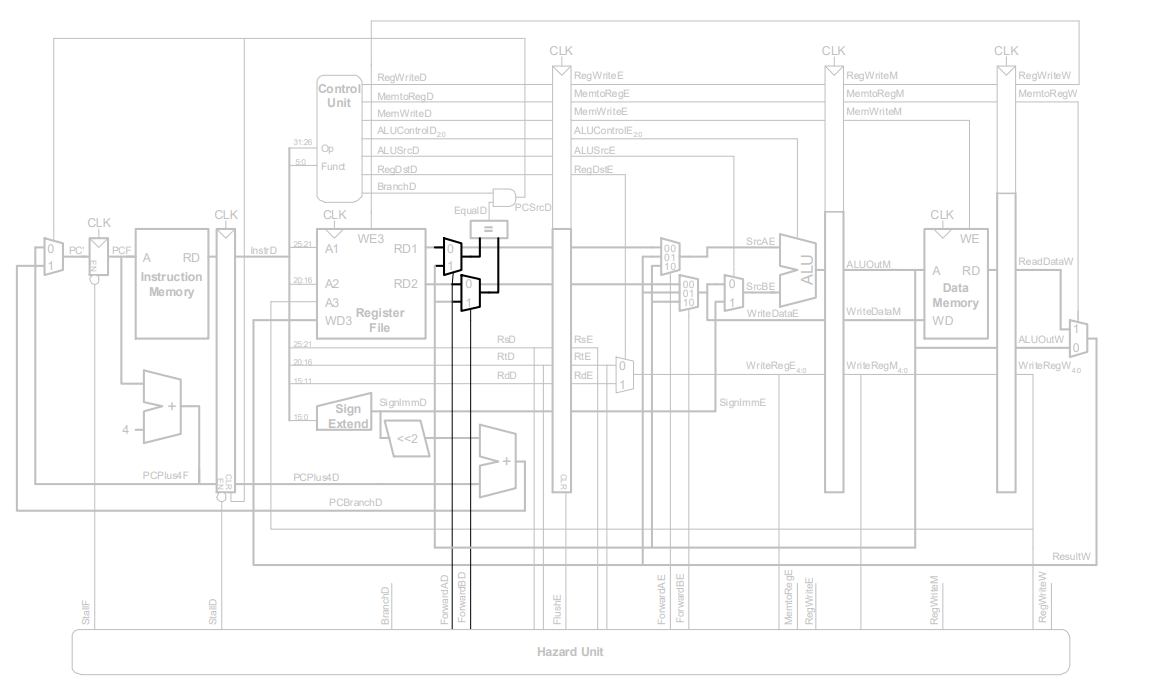


Control hazard:









The design above does not contain the Jump instruction pipeline data path, in fact, Jump instructions and Beq are carried out in ID phase, through pcsrcD\_flush (= pcsrcD | jump) as IF/ID pipeline register the clear signal, it can be as normal execution as Beq instructions.

This is the end of the design, but it has many drawbacks, such as when the instructions are as follows:

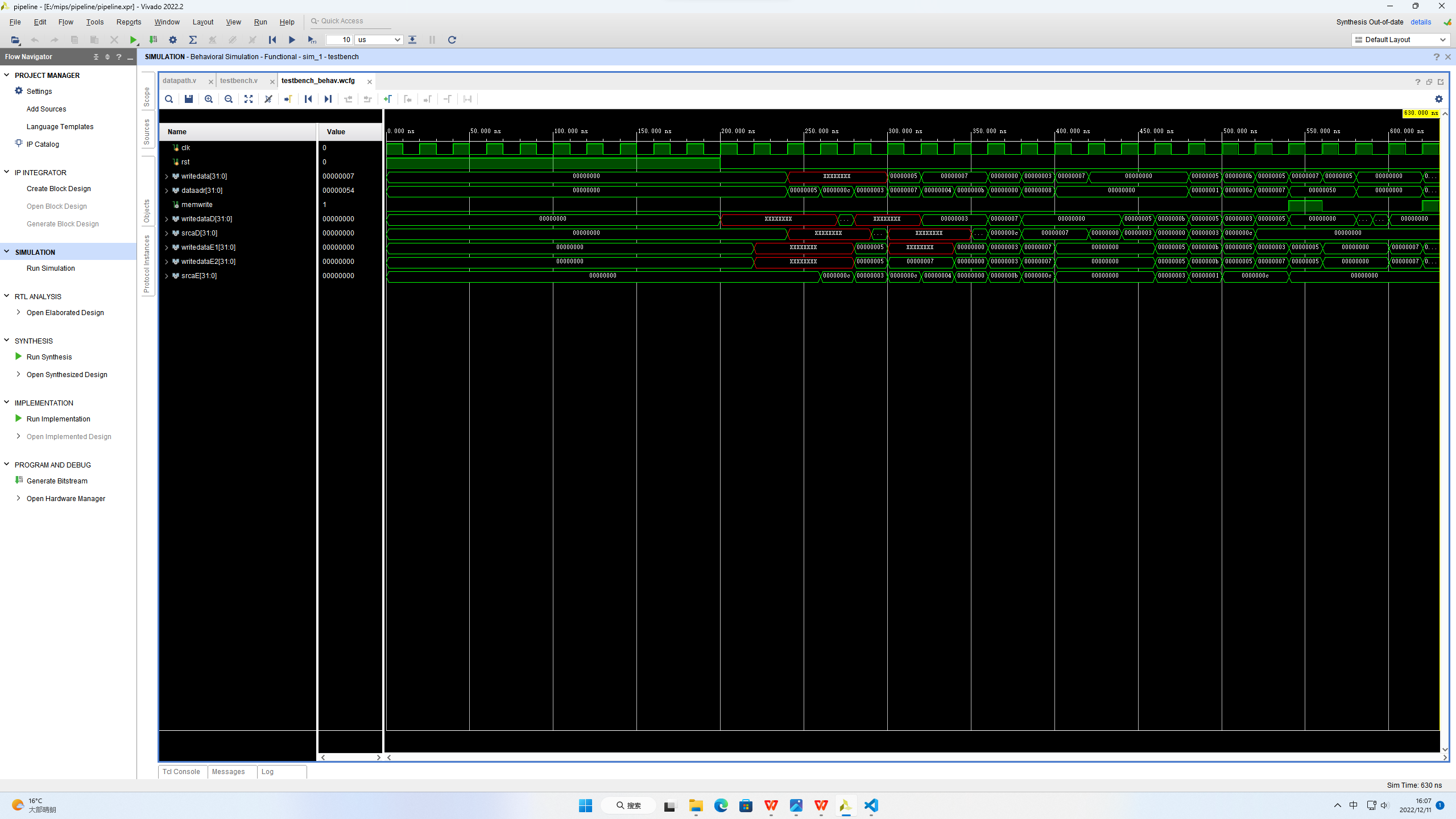
lw $2, 80($0)

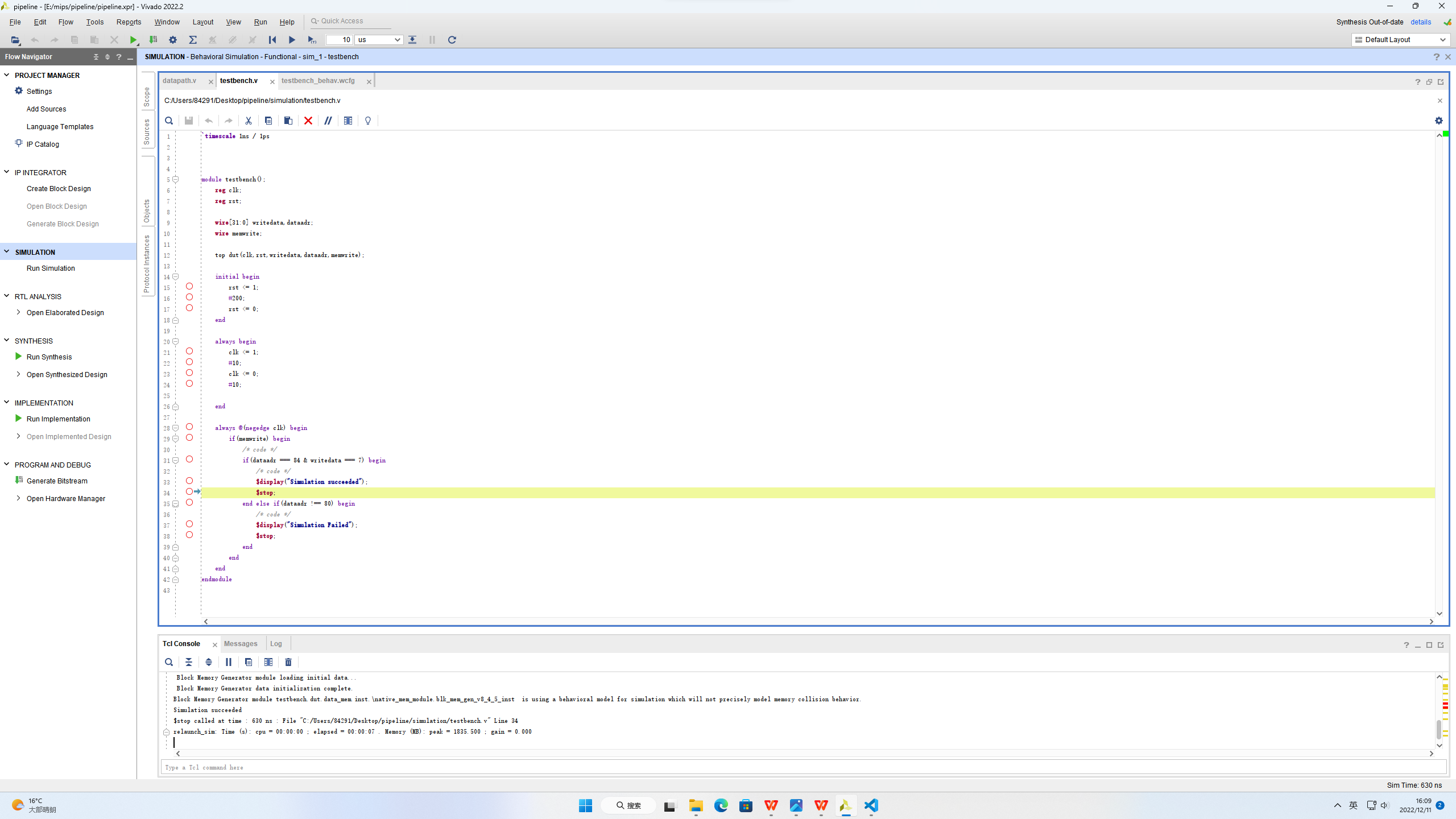
beq $2, $0, end

For normal execution, the CPU should generate two stalls until the lw instruction is in the MEM stage and the beq instruction is executed. This CPU cannot generate the above stall correctly.

Simulation:







Synthesis:

