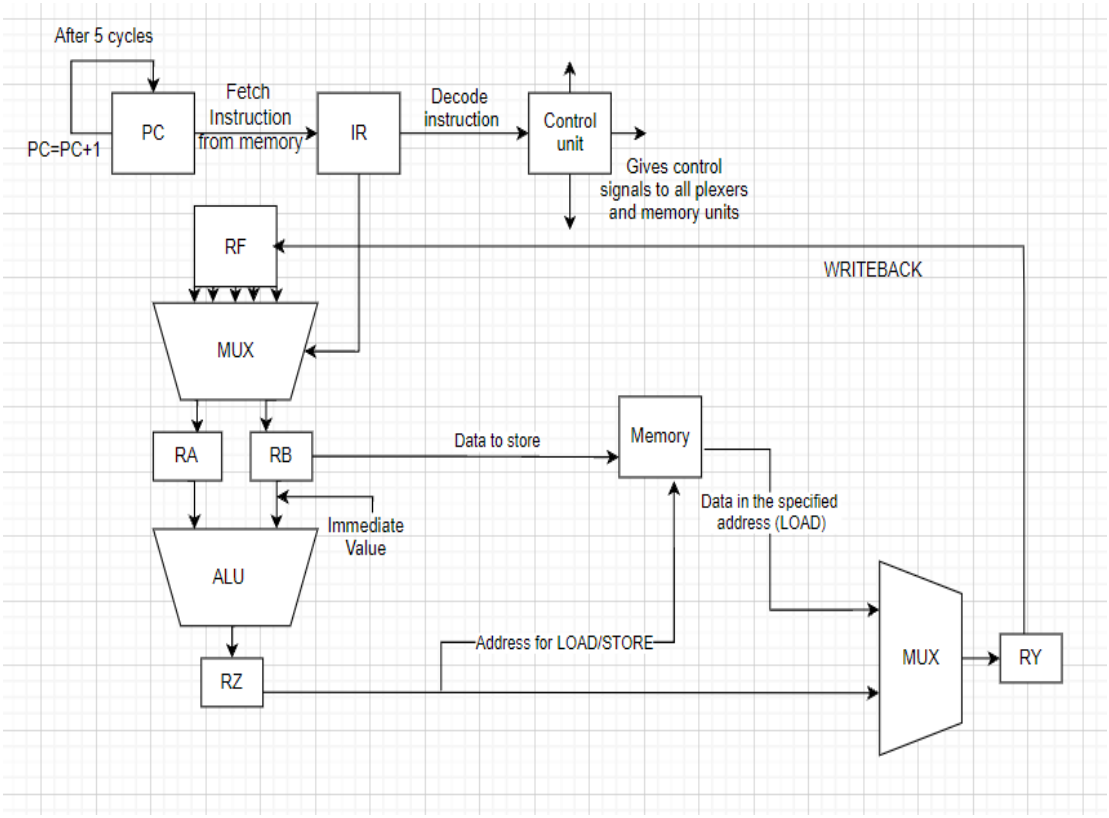
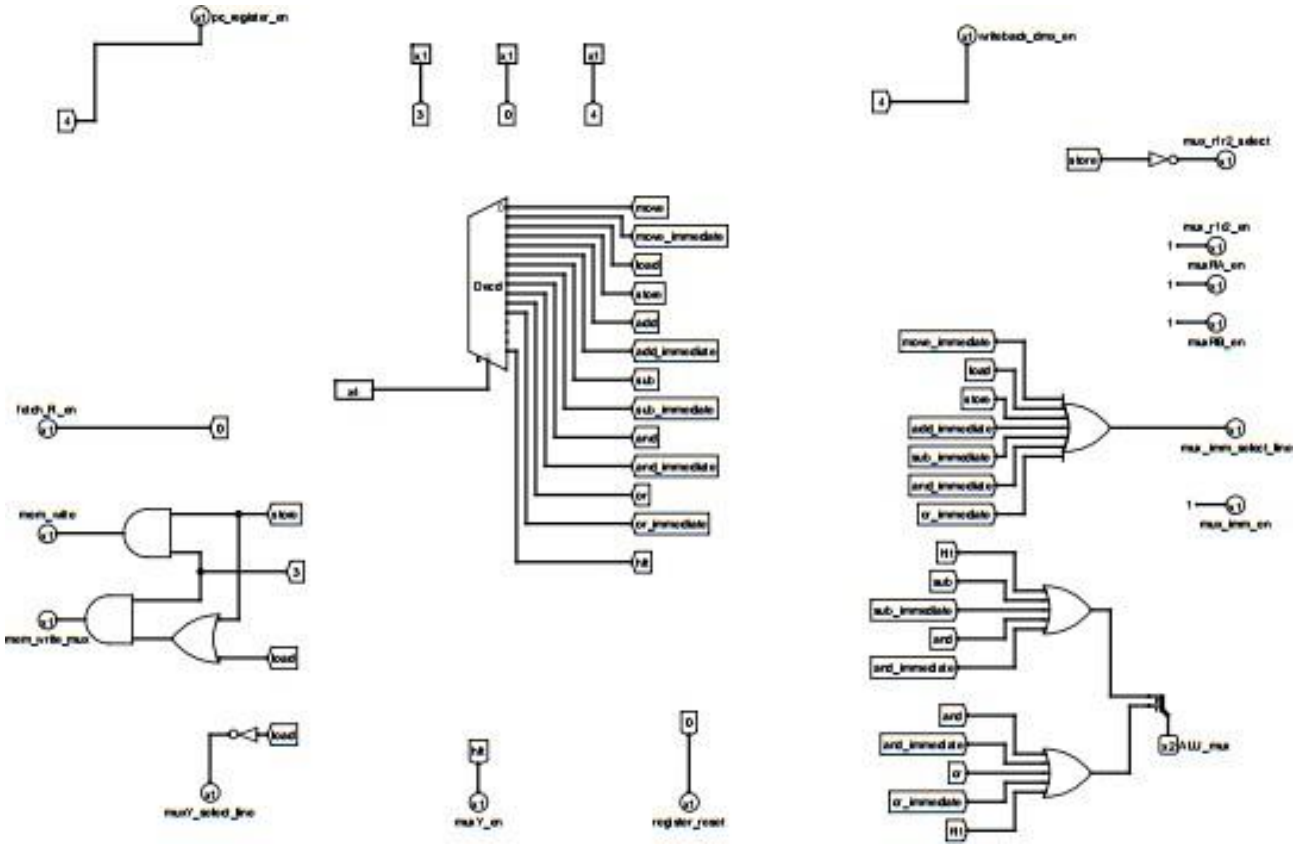
Block Level Implementation

The various circuit diagrams that we have utilized in implementing the main circuit of the 32-bit RISC Processor design are as follows:

- **5 Stage Pipeline Block Diagram:**

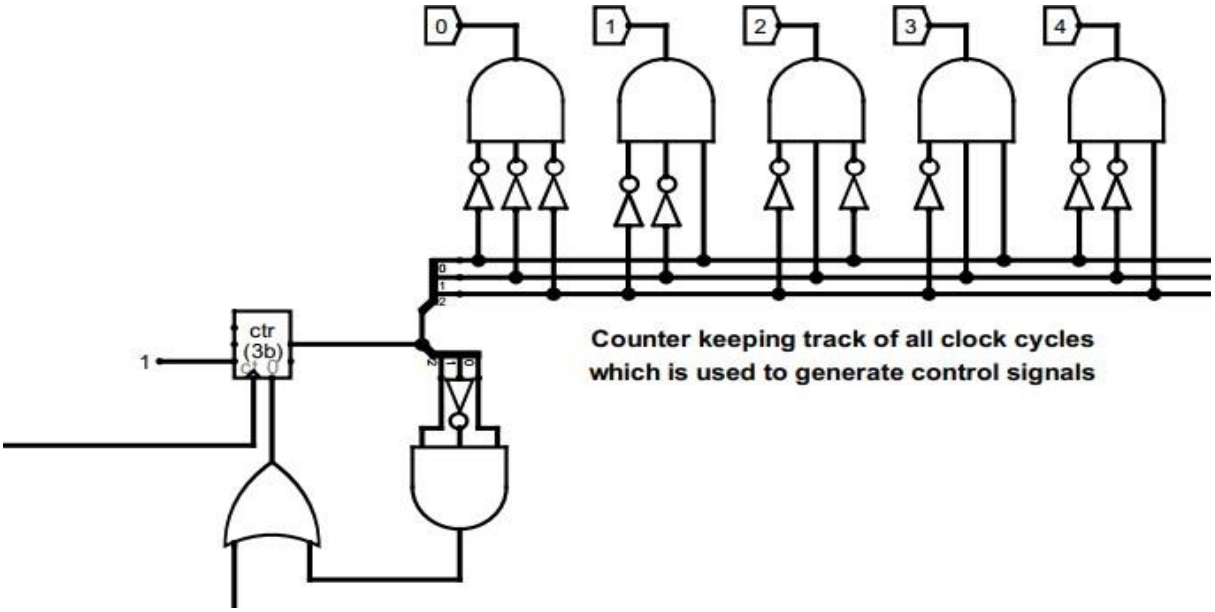


- **Control Unit :**



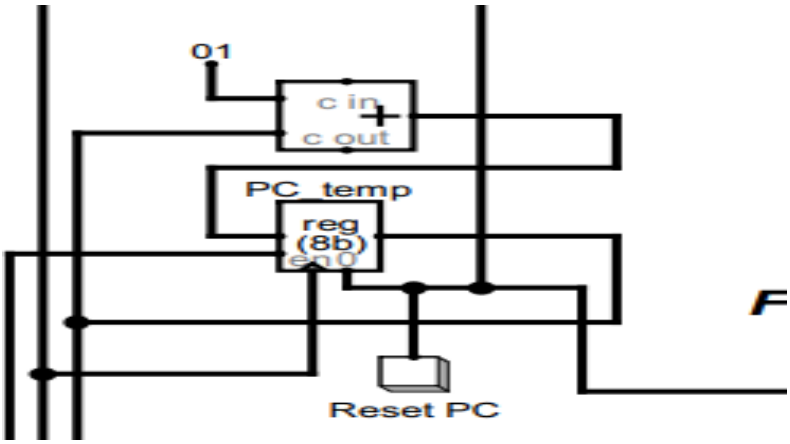
The control unit will generate all the signals that will control the flow of the whole circuit of the CPU design. All the select lines, enable lines of various MUX, registers and ALU etc. are controlled systematically by the control unit.

- **Stage Counter :**



In our RISC 32 bit processor design, every instruction is completed in exactly 5 stages, each of which completes in one clock cycle. The stage counter keeps the track of current stage and resets to stage 0 after the stage 4. This is utilized by the Control Unit to generate various signals depending on which stage the instruction execution is currently in.

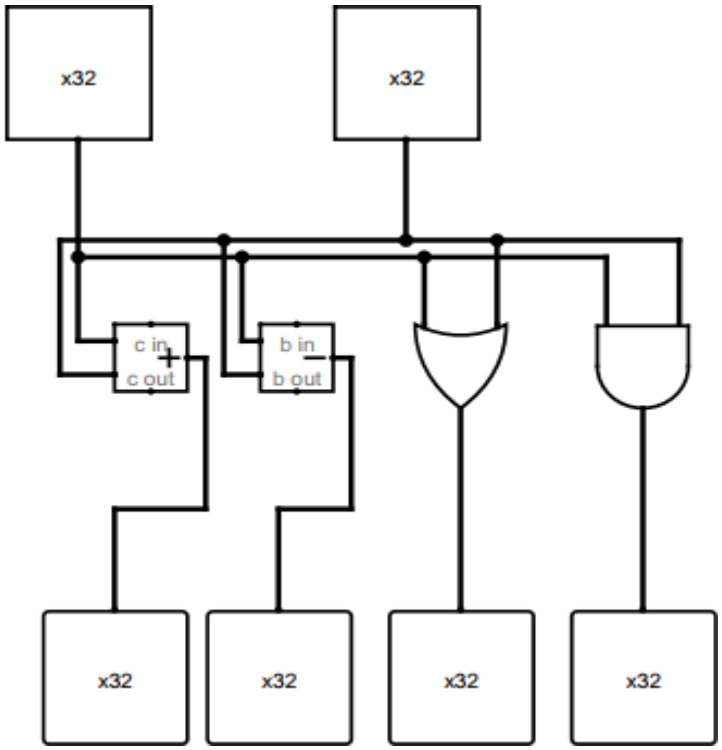
- **Program Counter :**



The program counter stores the address of the next instruction to be executed by the processor. Initially, it stores 00h (8 bit data address), then in every instruction, a control signal is produced in stage 4 which enables the 8 bit register and the address stored gets incremented by 1 by the adder circuit and fed to stage 0 of next instruction.

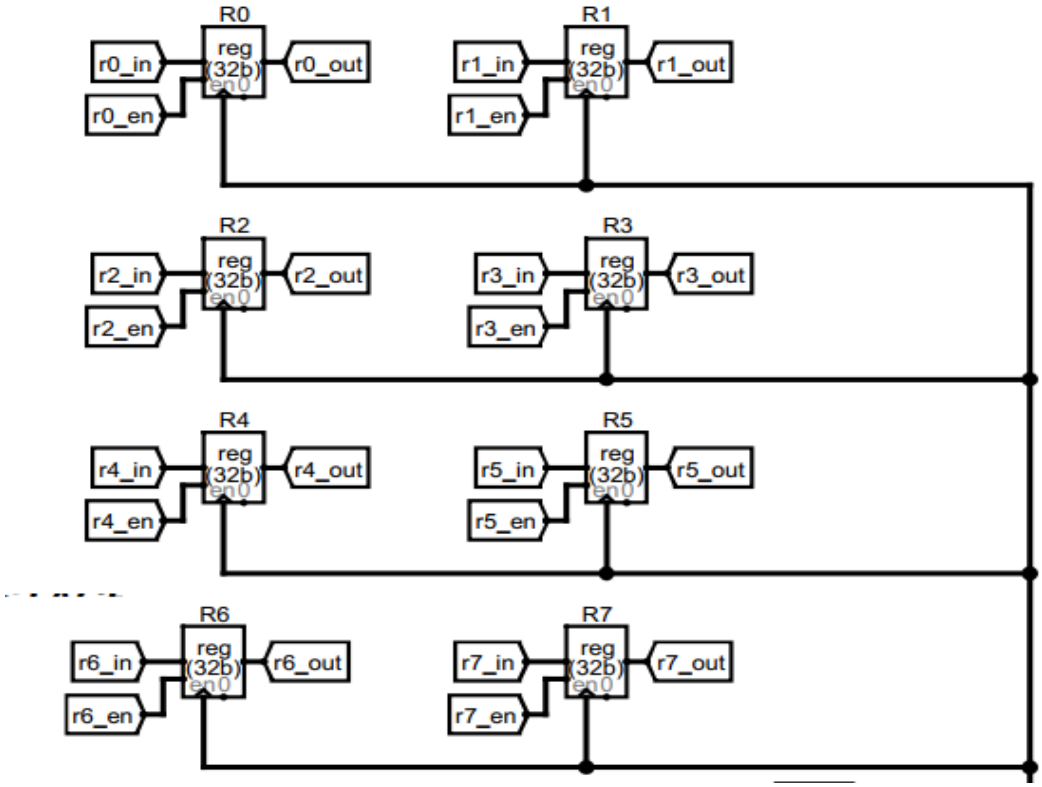
Reset PC button: Resets PC value to 00 and sets the counter value to 0. Also, this button can be used to restart the processor after using HLT command.

- **Arithmetic and Logical Unit :**



The ALU performs all sorts of arithmetic and logical operations required in the execution of the instruction. Our design will perform 4 basic operations : Add, Subtract, OR, AND. All the inputs and outputs in the ALU are 32 bit, so the 16-bit immediate values are 0 padded and extended to 32 bit. The Control unit will generate signal on the basis of OPCODE to perform the desired operation in the instruction.

- **Register File :**



The register file in our design consists of 8 32-bit registers (R0 – R7). The output lines of all the registers are connected to 2 Multiplexers which control flow of data from register file to Ra and Rb registers on the basis of Source register codes (R1 and R2) in the instruction. The input lines of all the registers are connected to a Demultiplexer which is used to select the register for writeback using the Destination Register Code (Rd).