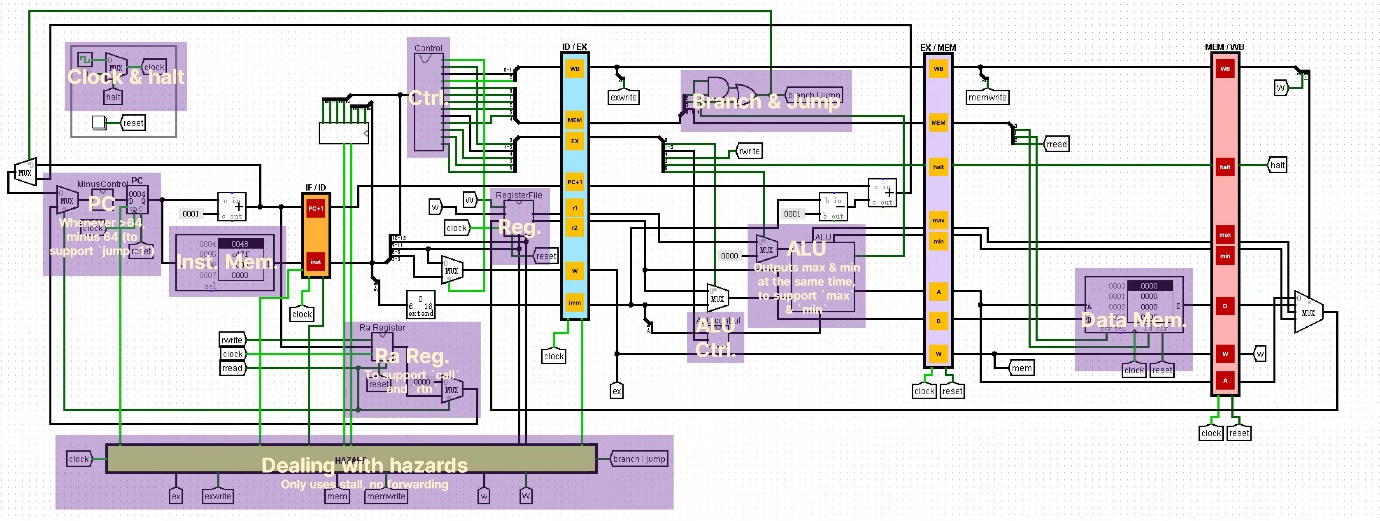
**Homework 3 Answer Sheet for the Bonus Question**

Please state the name and SID of all members of your group (you can add more rows if there are more than 5 members).

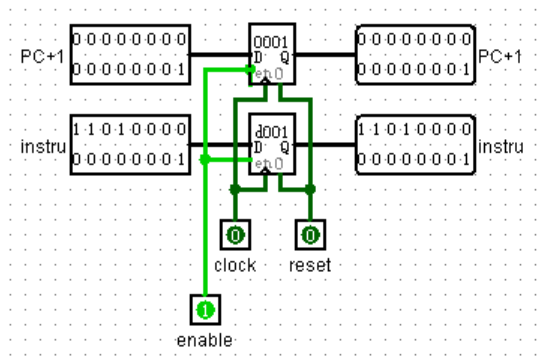
|  |  |  |
| --- | --- | --- |
| member | name | SID |
| #1 (contact person) | WU Xiaoqing | 56641363 |
| #2 | HUANG Jinyu | 56641166 |
| #3 | WU Jianrui | 56641885 |
| #4 | WANG Chenchen | 56344983 |
| #5 | DU Junye | 56641800 |

1. Please graphically explain the part of circuits included in each step of the pipeline.



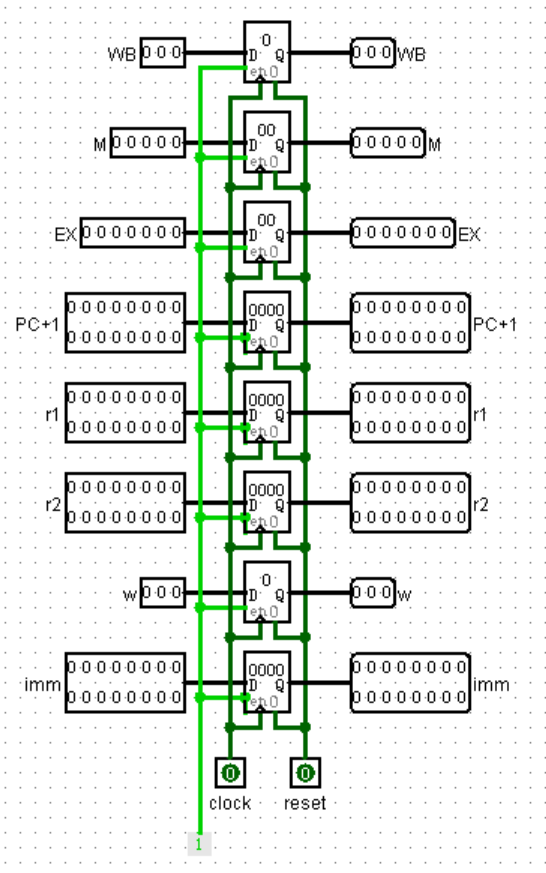
Stage 1: Instruction Fetch

In this step, CPU will fetch the instruction. The current PC is the index to instruction memory, and PC will be incremented at the end of circle. The instructions bits and PC are sent to pipeline register(IF/ID)

 IF/ID

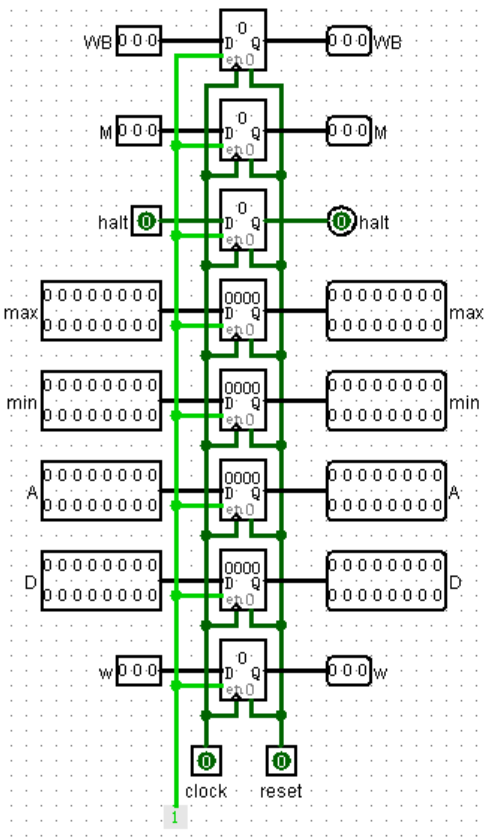
Stage 2: instruction decode & register file read

In this step, the instructions read from pipeline (IF/ID) are sent to decoder and generate control signals. PC and immediate number (offset) will be sent to pipeline register (ID/EX) directedly (PC+1/imm). Data handled by register files(r1/r2), write register (w) and other control information are also read into pipeline register (ID/EX).

ID/EX

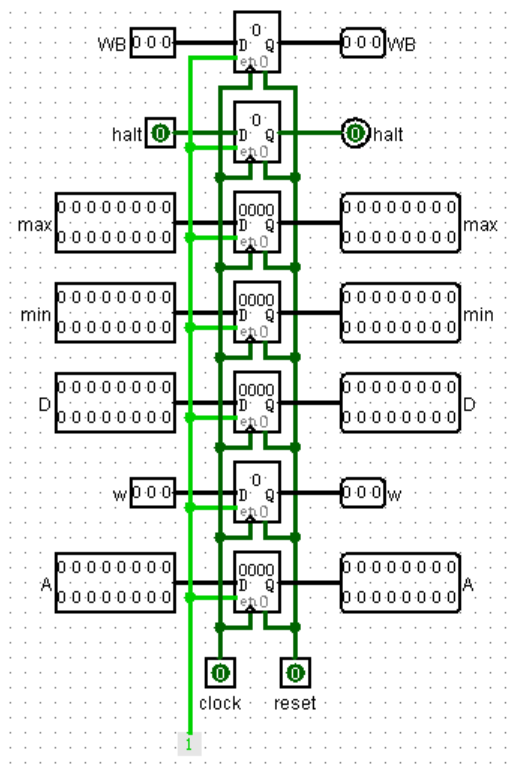
Stage 3: execute

In this circle, data are read from ID/EX pipeline register to get values and control bits, and CPU will perform ALU operation as well as compute targets (PC+4+offset) in case this is a branch. The result of ALU (max/min), memory related information (A/D), write register(w) and other control information are sent to pipeline register (IE/MEM)

EX/MEM

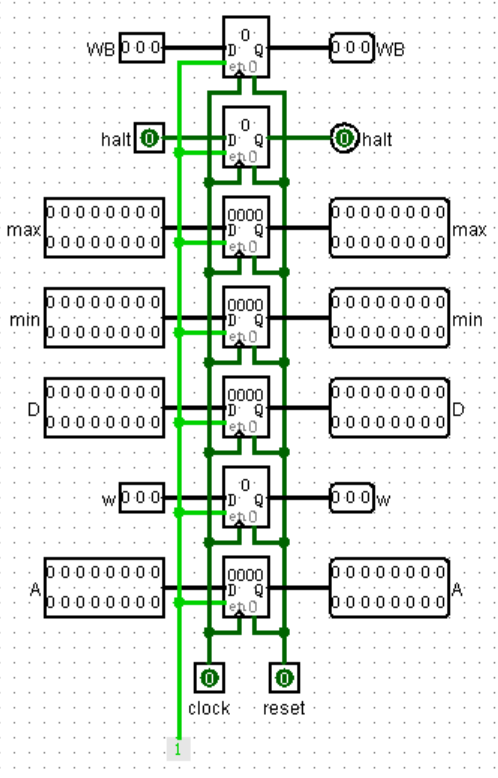
Stage 4: memory access

Read EX/MEM pipeline register to get values and control bits. Perform memory load/store if needed. The pass result of ALU (max/min), write register (w) and memory operation (A/D) and other control information are sent into pipeline register (MEM/WB)

MEM/WB

Stage 5: write back

In this step, read data from MEM/WB pipeline register to get values and control bits. Then select related value and write to register file

MEM/WB

1. Please explain how does your processor deal with different types of hazards.

In ID stage, if the register we are going to access is engaged by some other progresses, we will stall ID and flush IF until the other progresses finish.

If we have a branch or jump command, we will flush IF.