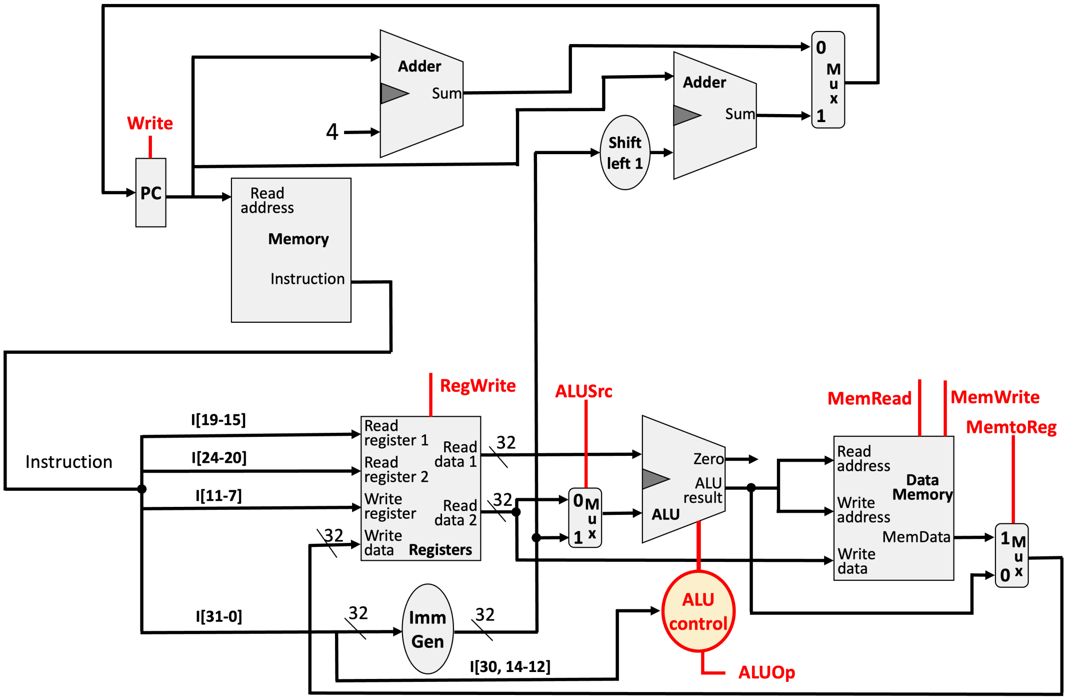

Figure 1: A diagram for a single-cycle datapath.

Question 4 (20 points): In this question we will examine the design of a datapath for a RISC-V processor.

1. (**5 points**) Figure 1 contains a block diagram for a single-cycle datapath design for a RISC-V processor. This diagram shows two separate memories: an instruction memory and a data memory. However, we know that a computer has only one memory. If we change the datapath design so that if features a single memory, there would be a structural hazard in the pipeline: the memory has to be accessed for both instruction fetching and data access in the same cycle. What is the solution adopted in modern computers to enable the use of a single memory while avoiding this structural hazard?

   The first level of cache, L1, is split into two separate caches, a data L1 and an instruction L1, which can be accessed in parallel during a single cycle.

2. (**5 points**) For the single-cycle datapath show in Figure 1, state what operation is performed in the ALU, and what happens with the result produced by the ALU, for the execution of the following instructions:

`sub s0, s1,s2 :`

The ALU subtracts the value in `s2` from the value in `s1`. the result is written into register `s0`

`sb t1, 8(t0) :`

The constant 8 is sign-extended to 32 bits. The ALU adds this value to the value in `t0`. The result is the memory address accessed by the `sb` instruction and it is sent to the memory as an address.

`beq t2, t3, label :`

The ALU subtracts the value in `t3` from the value in `t2`, the `zero` output of the ALU indicates if the branch should be taken or not taken.

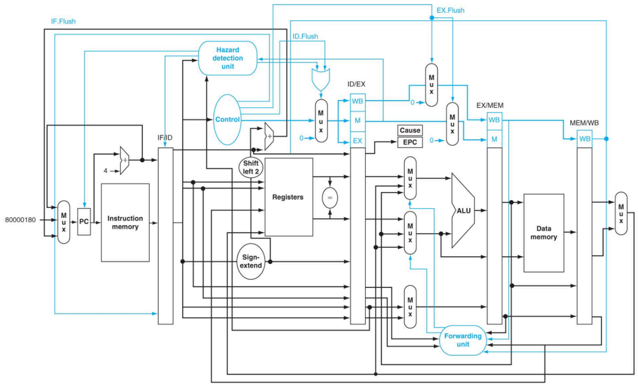

Figure 2: A diagram for pipelined datapath.

3. (**5 points**) Figure 2 shows the diagram from a possible design for a five-cycle pipeline for a RISC-V processor. However, with the design as presented, there is an issue with the specification of destination register for instructions that change the value of a register. Explain what the issue is and how it can be corrected.

The specification of the register to write is coming from the wrong instruction. Must carry the five bits specifying the register to write along the pipeline with the instruction. Thus, the ”write register” bits must come from the MEM/WB inter-stage register.

4. (**5 points**) The diagram in Figure 2 does not show the paths needed for forwarding. Assume that the following sequence of instructions are executed in this pipeline:

```
...
lw t0, 16(t2)
add t1, t3, t5
sub t0, t1, t0
...
```

Explain the forwardings that are needed to prevent delays in the execution of these instructions. What are all the changes to the design in Figure 2 that are needed to execute these instructions without delay. You can explain with a combination of text and drawing in the figure.

The value of `t0` computed by the `lw` instruction must be forwarded to one of the inputs of the ALU. The value of `t1` computed by the `add` instruction must be forwarded to the other input of the ALU. A new mux is needed in the top input of the ALU. A new input is needed in the bottom input of the ALU. A forwarding control logic is needed to detect the need for forwarding and issue the correct control signal to these muxes.