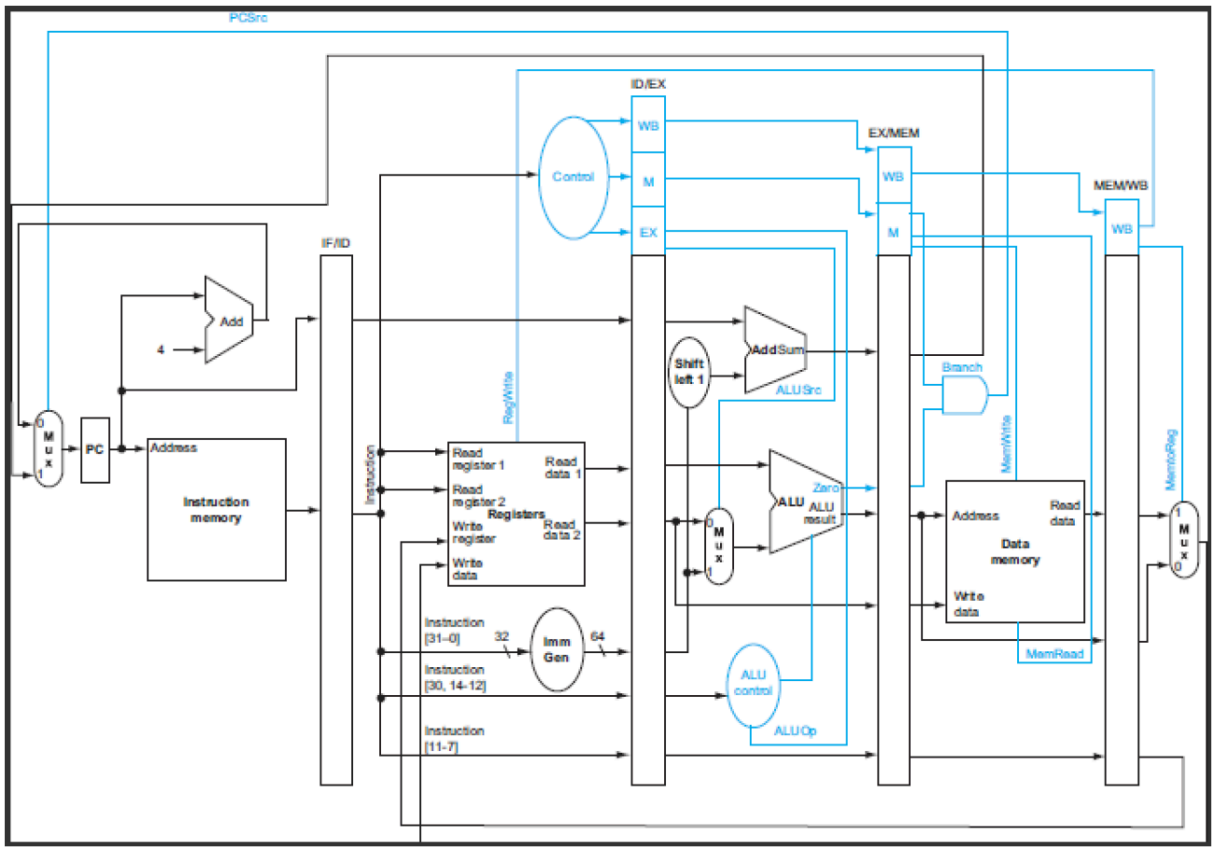

Figure 1: A diagram for a pipelined datapath.

Question 3 (15 points): Figure 1 contains a block diagram for a possible implementation of a five-stage pipeline similar to the one that we studied in class.

Assume that the following sequence of instructions execute in this pipeline:

```
0x000F 4000   lw     t0, 0(s1)
0x000F 4004   add    t1, t2, t3
0x000F 4008   sub    t4, t0, t1
0x000F 400C   sll    t5, t4, 4
0x000F 4010   sw     t0, 0(s5)
0x000F 4014   sra    t5, t5, 5
0x000F 4018   bne    t5, zero, loop
```

1. (**4 points**) During the execution of the sequence of instructions above, is there a need to insert any delay ("bubble") in the pipeline for the correct execution of the instructions? Explain

   The only need for a delay would be because of load-use delay. However, the register `t0` that receives the value of the load is only needed in the `sub` instruction. By that time the value is available for forwarding.

2. (**4 points**) Consider the clock cycle in which the `lw` instruction is in the write-back stage. Is there any forward that is needed in that clock cycle? If yes, state which forwarding is needed.
The `sub` instruction is in the execution stage, it uses `t0` which needs to be forwarded from the `MEM/WB` register to the `ALU` and it uses `t1` which needs to be forwarded from the `EX/MEM` stage to the ALU.

3. (**4 points**) Assume that during the execution of this sequence of instructions, the value in `s5` is 0x8000 0007. Would the execution of this sequence generate any exceptions in the pipeline? Explain your answer. If any exception is generated, explain how the exception would be handled in the pipeline. Make sure to specify in which stage of the pipeline exceptions are generated and what will happen with each of the instructions in the sequence.

   The `sw` to an odd address will cause an exception when the `sw` reaches the memory stage.
   The following instructions will execute to completion: `lw`, `add`, `sub`, `sll`.
   `sw`, `sra`, and `bne` instructions will be transformed into NOP. The address of the `sw` instruction will be stored in the `EPC` and the instruction from address `0x80000180` will be fetched.

4. (**3 points**) The `Mux` on the far left side of the Figure 1 has two inputs. The output of that `Mux` is sent to the `PC`. Explain what is the role of these two inputs, There is a third value that may be sent to the `PC`, but this value is not shown in this diagram. When is this third input used?

   - The top entry is `PC+4` and it is selected when the next sequential instruction is to be executed.
   - Branch Target is the lower input, this input is selected when the instruction is a branch and the branch is taken.
   - The address of the exceptions handler, `0x8000 0180`, is assigned to the `PC` when an exception occurs and the exception handler needs to be invoked.