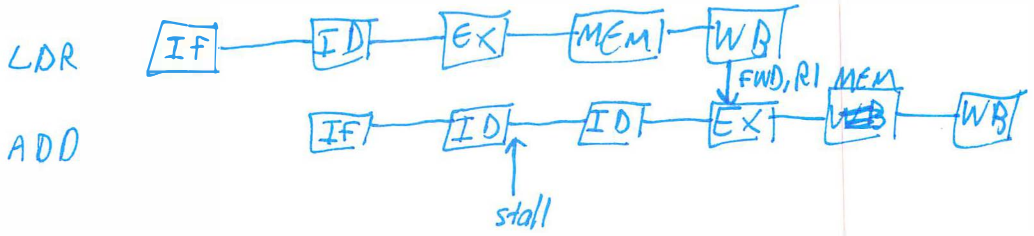
Name Answers

CE-1921-11 – Dr. Durant – Quiz 8
Spring 2017, Week 9

1. (2 points) Write a sequence of assembly instructions that generates a load-use (LDR) hazard.
2. (4 points) Describe a solution to the load-use hazard that ensures that the processor executes the above sequence properly. (−1) 2 stalls but ans. correct
3. (4 points) Draw a pipeline in-flight diagram for your sequence of instructions, illustrating key details (e.g., stalling, flushing, and/or forwarding) of how the hazard is resolved. (−1.5) show fwd (−1/4) stall @ IF not ID

(1) LDR R1,[R2,#0]
ADD R3,R1,R4

(2) Stall ADD @ decode, injecting a bubble.
Then, when ADD reaches EX, the R1 value is available for forwarding.

(3)


Name *Answers*

CE-1921-21 – Dr. Durant – Quiz 8
Spring 2017, Week 9

1. (2 points) Write a sequence of assembly instructions that generates a control hazard, specifically one with a conditional branch (assume it is ultimately taken for illustration purposes below).
2. (4 points) Describe a solution to the above hazard that ensures that the processor executes the above sequence properly.
3. (4 points) Draw a pipeline in-flight diagram for your sequence of instructions, illustrating key details (e.g., stalling, flushing, and/or forwarding) of how the hazard is resolved.

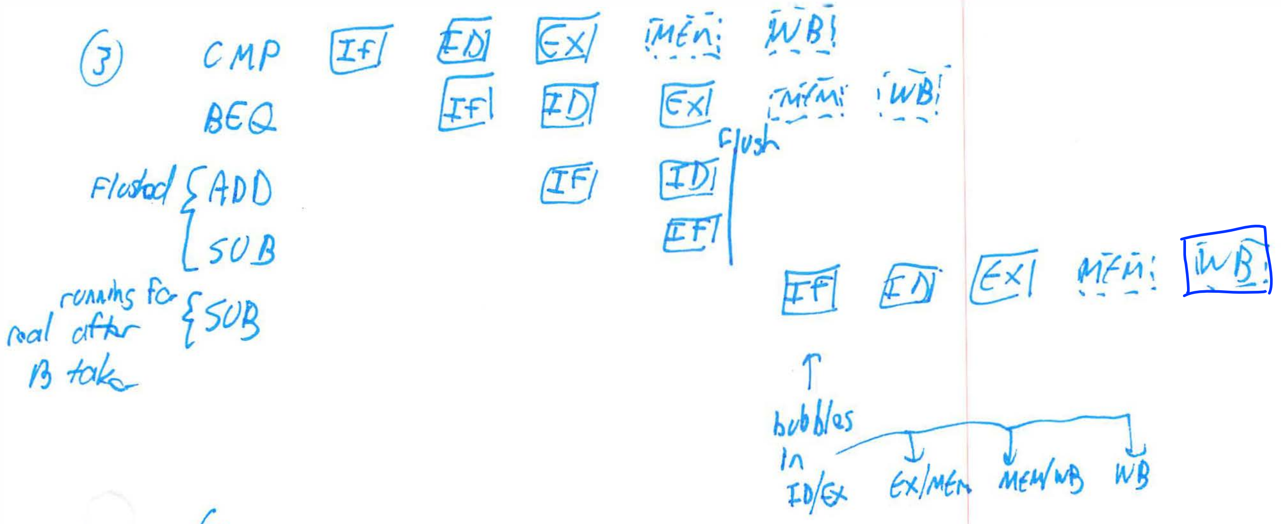
-1) flush early (NZVC cmt.)

-2½) Mis Id CTRL hazard as data hazard on flags // propose 2-cycle stall.

①
```
    CMP  R1,R2
    BEQ  LØ
    ADD  R3,R4,R5
LØ: SUB  R4,R5,RØ
```

② When B* reaches EX, flush skipped instructions if B taken.

③



Common Error

Note: EX owns the flags. S-type instructions, including CMP, update NZVC flags in EX. So, there is no hazard between CMP + BEQ. (If there were a hazard, say if flag wr. happened @ WB, it would be a data hazard.)