Problem 1: The MIPS program below copies a region of memory and runs on the illustrated implementation. In the sub-problems below use only the bypass connections shown in the illustration.

(*a*) Show a pipeline execution diagram for the code running on the illustrated implementation for two iterations.

(*b*) Compute the CPI and the rate at which memory is copied in bytes per cycle assuming a large number of iterations.

- Don't forget, when computing the number of cycles per iteration be sure not to count a cycle more, or less, than once.

```
LOOP:
 lw $t0, 0($a0)
 sw 0($a1), $t0
 addi $a0, $a0, 4
 bne $a0, $a2 LOOP
 addi $a1, $a1, 4
```
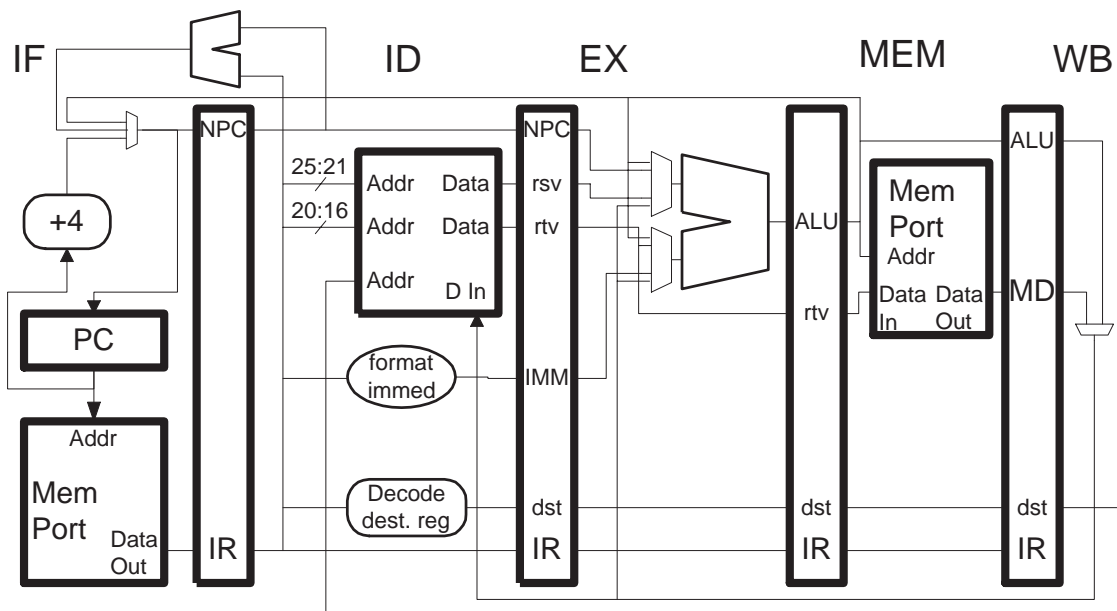

Problem 2: Execution should be inefficient in the problem above.

(*a*) Add **exactly** the bypass connections needed so that the program above executes as fast as possible.

- Don't forget that branch uses ID-stage comparison units.

- Don't forget the store.

(*b*) Show a pipeline execution diagram of the code on the improved implementation.

(*c*) For each bypass path that you've added show the cycles in which it will be used by writing the cycle number near the bypass path. If a bypass path goes to several places (for example, both ALU muxen) put the cycle number at the place(s) that use the signal.

(*d*) Re-compute the CPI and the rate at which memory is copied.