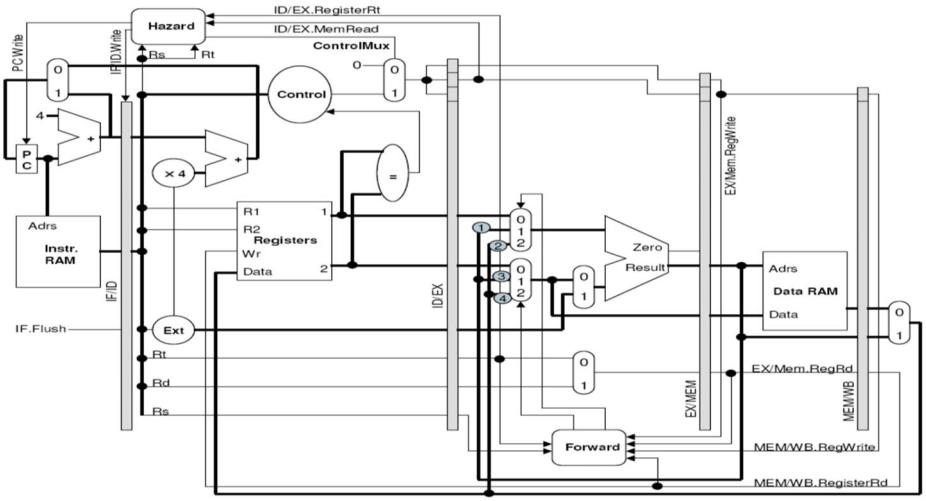
Problem 4.




```
apple:
     lw     $t0, 0($a1)
     mul    $t0, $t0, $a2
     lw     $t1, 0($a0)
     add    $t0, $t0, $t1
     sw     $t0, 0($a0)
     addi   $a3, $a3, -1
     addi   $a0, $a0, 4
     addi   $a1, $a1, 4
     bne    $a3, $zero, apple
```

Part (a) Assume that we run the "apple" code on the pipelined datapath shown above. Find the number of cycles needed to execute this code, accounting for all possible stalls and flushes. Assume that $a3 is initially set to 100. Show your work by filling the table below. Assume branches are executed as predicted not taken. That means, instruction(s) following the beq is/are flushed if the branch is taken (branch condition satisfied).

Inst	iter	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2
lw	N	F	D	E	M	W																	
mul	N		F	D	D	E	M	W															
lw	N				F	D	E	M	W														
add	N					F	D	D	E	M	W												
sw	N							F	D	E	M	W											
addi	N								F	D	E	M	W										
addi	N									F	D	E	M	W									
addi	N										F	D	E	M	W								
bne	N											F	D	E	M	W							
lw	N+1													F	D	E	M	W					

12 clock cycles per loop × 100 loops + 3 clock cycles (finish bne) = 1203 CC

Part (b) Show how you can re-arrange the instructions in the code above to eliminate as many stall cycles as possible. You do not need to reduce the number of flushes.

```
apple:
     lw   $t0, 0($a1)
     lw   $t1, 0($a0)
     mul  $t0, $t0, $a2
     add  $t0, $t0, $t1
     sw   $t0, 0($a0)
     addi $a3, $a3, -1
     addi $a0, $a0, 4
     addi $a1, $a1, 4
     bne  $a3, $0, apple
```

Problem 5

Datapath shows fifth cycle of executing this program. Fill in the ten remaining values, marked with a **?** symbol, in the EX and MEM stages. Assume that registers initially contain their number plus 100: $2 contains 102, $8 contains 108, etc.