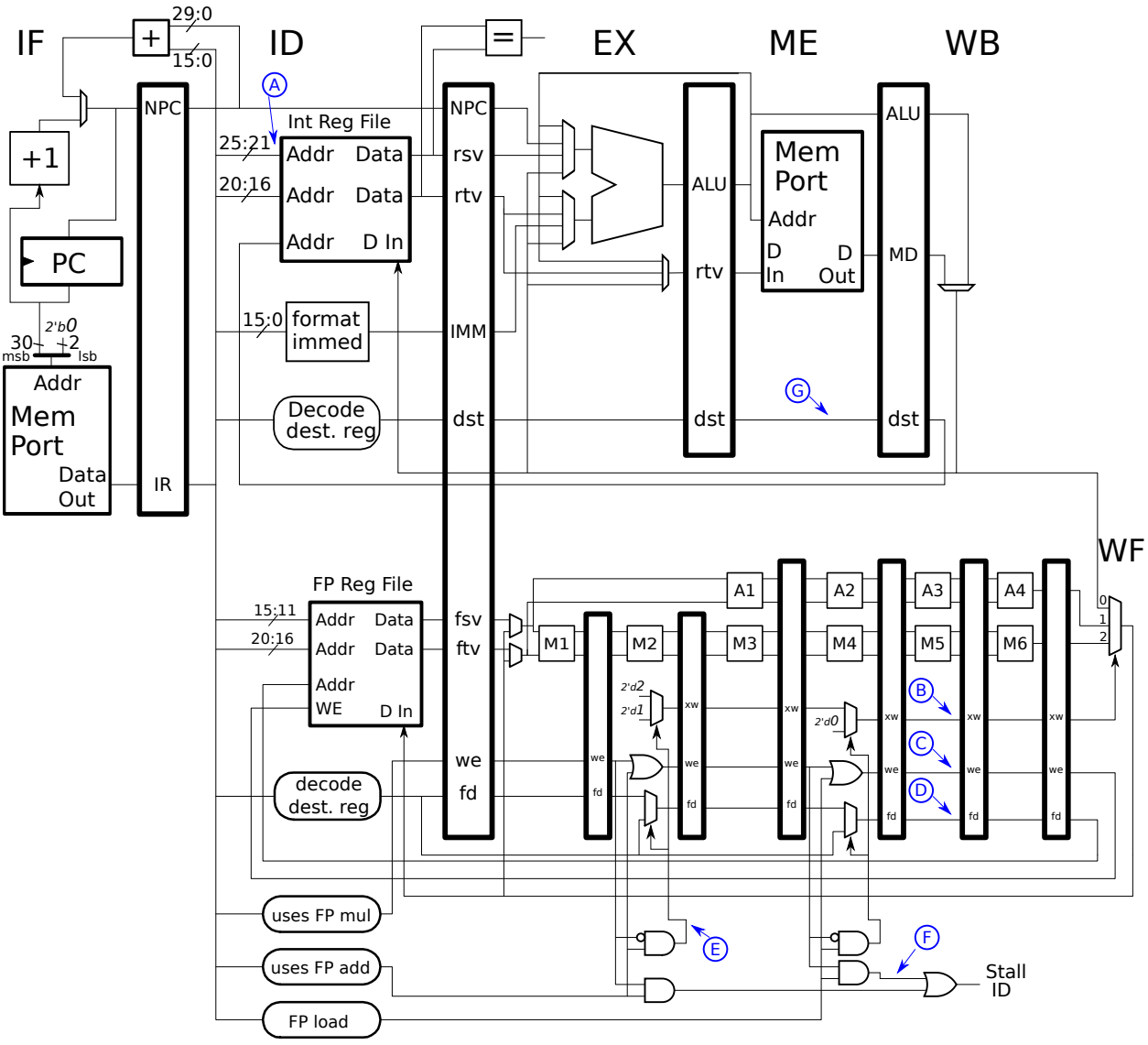
LSU EE 4720 **Homework 6** **Due: 23 April 2018**

Problem 1: Solve 2014 Homework 4 Problem 1.

Problem 2: Appearing on the next page is a MIPS implementation and the execution of some code on that implementation.



(*a*) Wires in the implementation (on the previous page) are labeled in blue. Show the values on those wires each cycle that they are affected by the executing instructions. The values for label A are already filled in.

```
LOOP: # Cycle       0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23
 lwc1 f0, 0(r1)     IF ID EX ME WF
 mul.s f1, f0, f10     IF ID -> M1 M2 M3 M4 M5 M6 WF
 add.s f2, f2, f1         IF -> ID -------------> A1 A2 A3 A4 WF
 lwc1 f0, 4(r1)                 IF -------------> ID EX ME WF
 mul.s f1, f0, f11                                IF ID -> M1 M2 M3 M4 M5 M6 WF
 add.s f2, f2, f1                                    IF -> ID -------------> A1 A2 A3 A4 WF
 bne r2, r1 LOOP                                           IF -------------> ID EX ME WB
 addi r1, r1, 8                                                              IF ID EX ME WB


# Cycle             0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23

A                      1                           1                           2   1 # Sample

B

C

# Cycle             0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23

D

E

# Cycle             0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23

F

G

# Cycle             0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23
```

(*b*) Schedule the code above so that it suffers fewer stalls. Register numbers can be changed but in the end the correct value must be in register `f2`. It is okay to add a few instructions before and after the loop. Each iteration must do the same amount of work as the original code.

Show a pipeline execution diagram for two iterations.

Problem 3: Design control logic for the lower `M1`-stage bypass multiplexor. Note that this is fairly easy since the mux has two inputs, so it's only necessary to detect the dependence.

An SVG source for the FP diagram can be found at:
`http://www.ece.lsu.edu/ee4720/2018/mpipei-fp-by.svg`.