LSU EE 4720 Homework 4 Due: 1 April 2020

It's up to all of us: $r > 2\,\mathrm{m} \Rightarrow R_e < 1$ where r is the radius of the largest circle with you at the center and containing only people in your household, and R_e is the effective reproduction number, the number of people infected by an infected person.

Problem 1: Appearing below is the code fragment from Homework 3.

```
# Cycle          0     1     2     3     4     5     6
addi R2, r0, 0   IF    ID    EX    ME    WB
mul R1, R2, r3         IF    ID    EX    ME    WB
add r4, R2, R1               IF    ID    EX    ME    WB
# Cycle          0     1     2     3     4     5     6
```

(*a*) Does this code fragment look like it was compiled with optimization on?

If your answer is something like "yes, it could be part of optimized code" then explain why you think it so and provide any missing context. (Do not change or re-arrange the three instructions above.)

If your answer is something like "no, it does not appear optimized" then show what the code would look like after optimization. *Hint: A correct answer can start with either "Yes it does" or "No it doesn't". The "No" answer is straightforward.*

Problem 2: MIPS does not appear to have a `muli` instruction.

(*a*) Comment on the following:

> *MIPS has a* `mul` *instruction but does not have a* `muli` *instruction because, as the solution to Homework 2 shows, the additional hardware for* `muli` *(beyond that used for* `mul`*) would be too costly.*

Is the statement above reasonable or unreasonable? Explain.

(*b*) Show the encoding of MIPS instruction `mul r1, r2, r3`. Show all 32 bits of the instruction, divided into fields (each field can be shown in the radix of your choice). (The MIPS ISA manuals are linked to the course Web page. Instruction encodings are in Volume II.)

(*c*) Some possible reasons that there is no `muli` instruction in MIPS is that either there are no Format-I opcodes available (they are all used by other instructions) or that the few remaining opcodes are being kept in reserve for a better instruction than a `muli`.

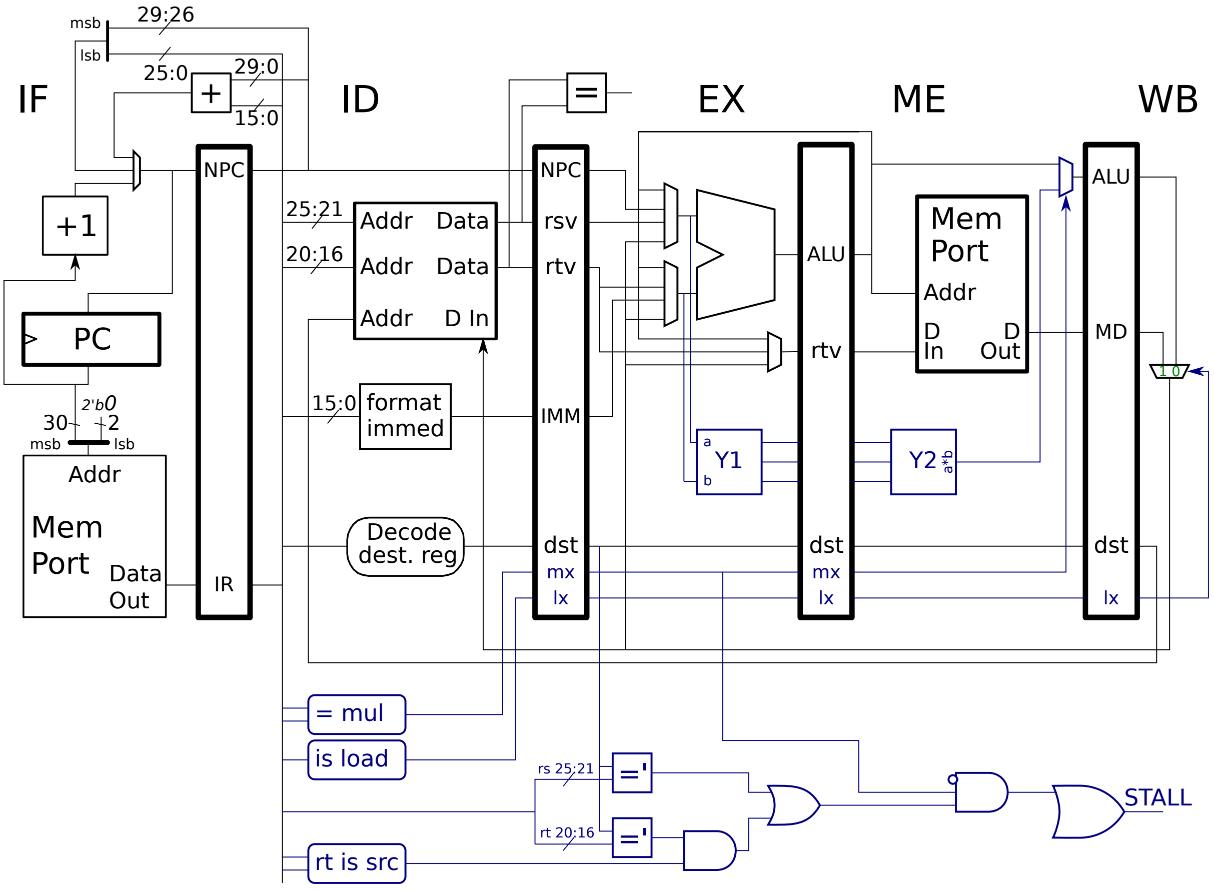
Based on the MIPS Architecture Manuals (they are linked to the course references page) how many opcodes are available for new Format-I instructions? The easy way to solve this is to find the right table. The hard way to solve this is to go through the 144 or so pages of instruction descriptions. *Hint: Look in volume I.*

Problem 3: Perhaps you saw this coming: Time to add `muli` to MIPS.

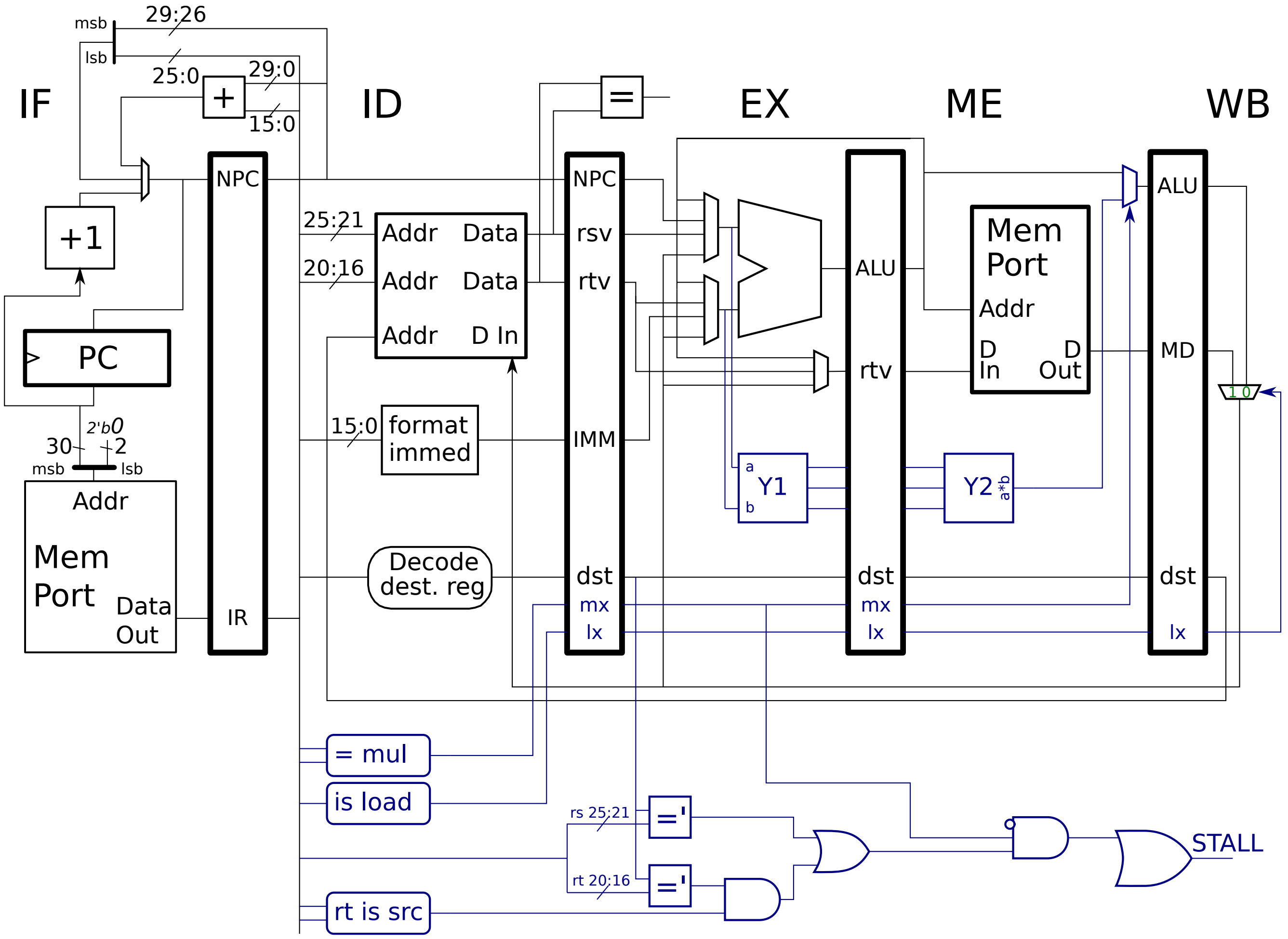
(*a*) Show how a Format-R `muli` instruction with a ten-bit immediate might be defined using unused fields in the Format-R encoding. Make up your own function field value, but try to pick one that's unused. (See the previous problem.) Show how `muli r1, r2, 43` might be encoded for your `muli` definition.

(*b*) Modify the hardware below (there's a copy on the next page) to implement this new instruction. The modified hardware should provide the immediate needed by `muli`. Show datapath **but not** control logic. Of course, any changes should not break existing instructions.

Pay attention to cost and performance. This can easily be solved by adding a mux in the `ID` stage. *Hint: The solution is not much more than a mux. Be sure to carefully label the inputs.*



The SVG source for the diagram below is available at `https://www.ece.lsu.edu/ee4720/2020/hw03-p2.svg`.


IF
ID
EX
ME
WB
msb
lsb
29:26
25:0
29:0
15:0
+
=
NPC
+1
PC
2'b0
30
2
msb
lsb
Addr
Mem Port
Data Out
IR
25:21
20:16
Addr
Data
Addr
Data
Addr
D In
15:0
format immed
Decode dest. reg
NPC
rsv
rtv
IMM
dst
mx
lx
ALU
rtv
dst
mx
lx
Mem Port
Addr
D In
D Out
ALU
MD
dst
lx
1 0
a
b
Y1
Y2
a*b
= mul
is load
rt is src
rs 25:21
rt 20:16
='
='
STALL