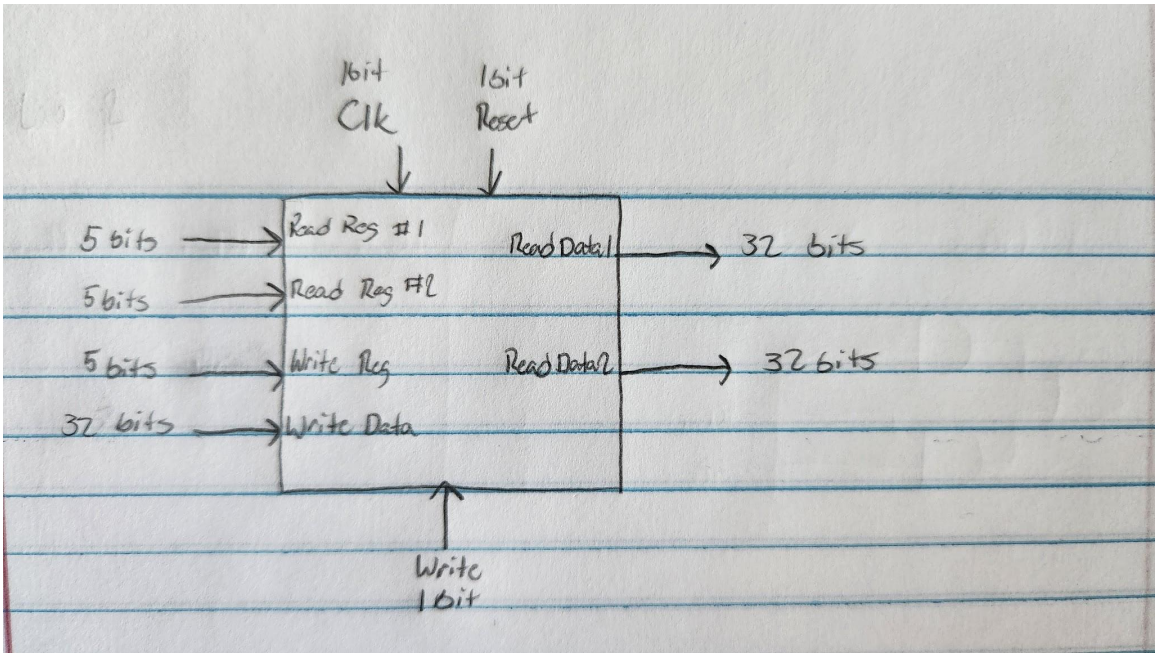
CprE 381, Computer Organization and Assembly-Level Programming

Lab 2 Report

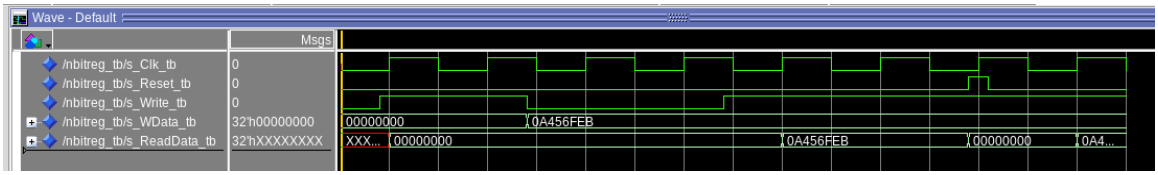
Student Name Zach Scurlock

Submit a typeset pdf version of this on Canvas by the due date. Refer to the highlighted language in the lab document for the context of the following questions.

[Part 2 (a)] Draw the interface description for the MIPS register file. Which ports do you think are necessary, and how wide (in bits) do they need to be?



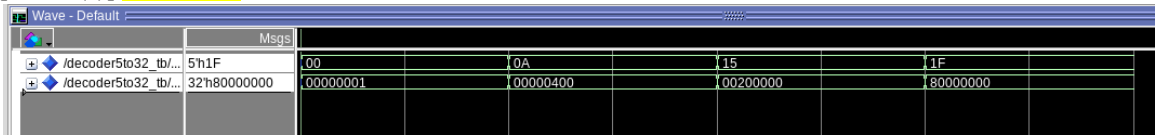
[Part 2 (c)] Waveform.



[Part 2 (d)] What type of decoder would be required by the MIPS register file and why?

A 5 to 32 bit decoder would be required because there are 32 different registers to choose from.

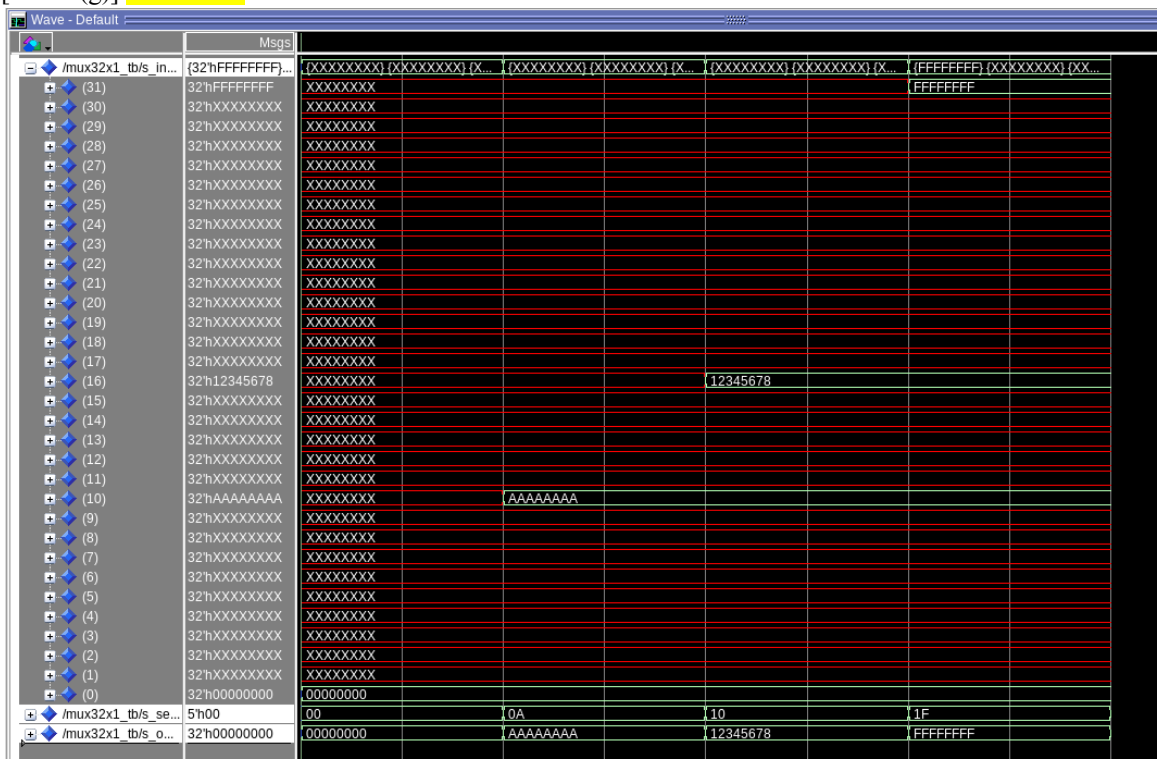
[Part 2 (e)] Waveform.



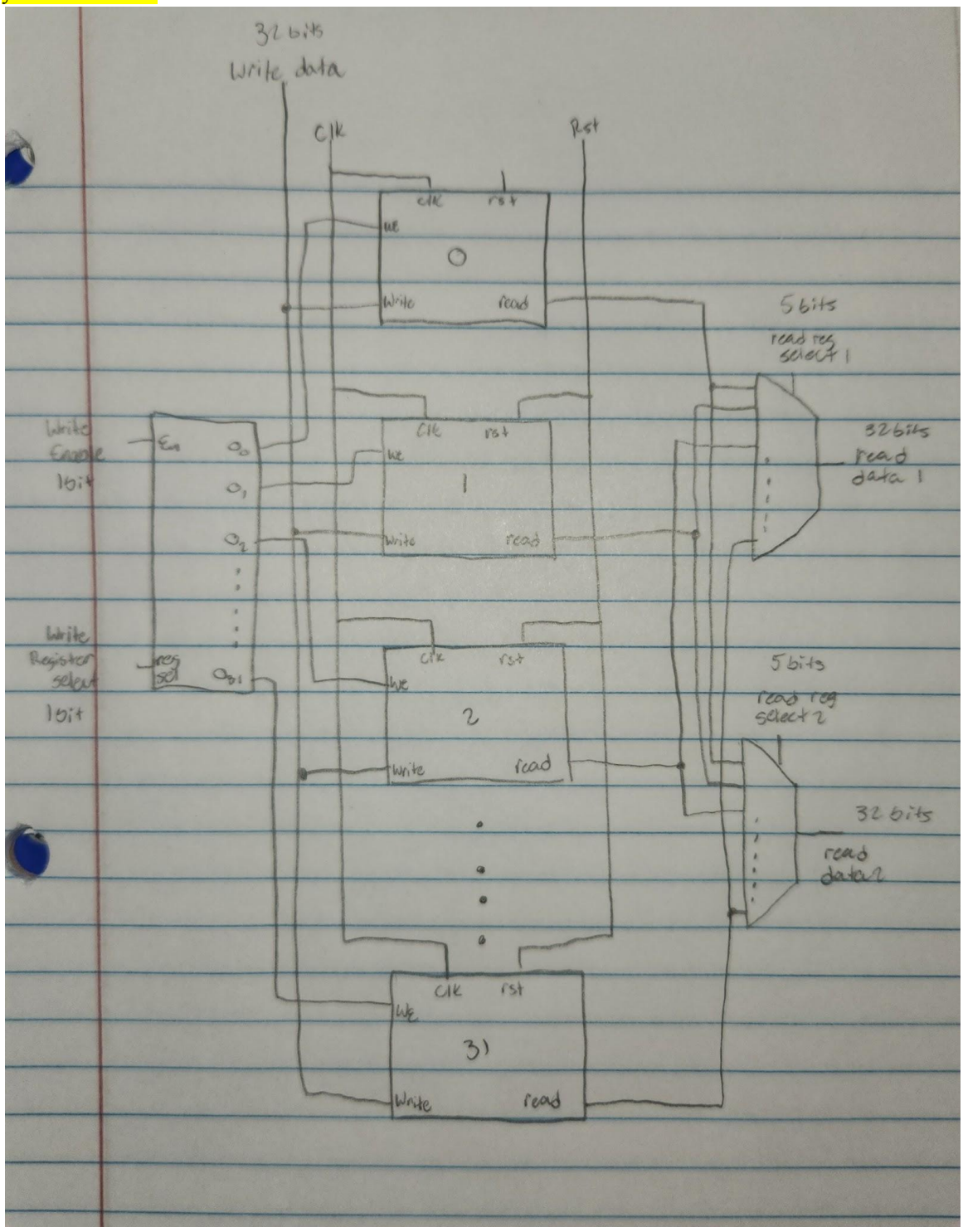
[Part 2 (f)] In your write-up, describe and defend the design you intend on implementing for the next part.

I decided to make a new data type that serves as a 2D array. This will make it easier to implement because it will save time and use fewer lines.

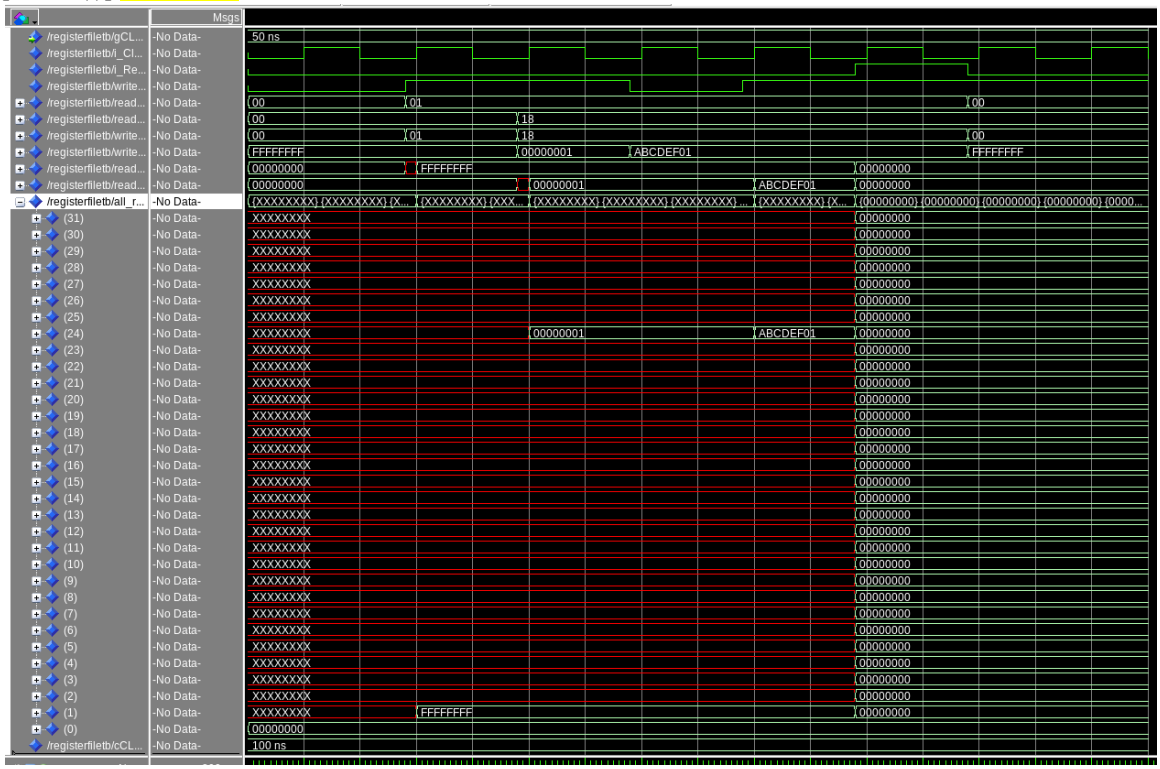
[Part 2 (g)] Waveform.



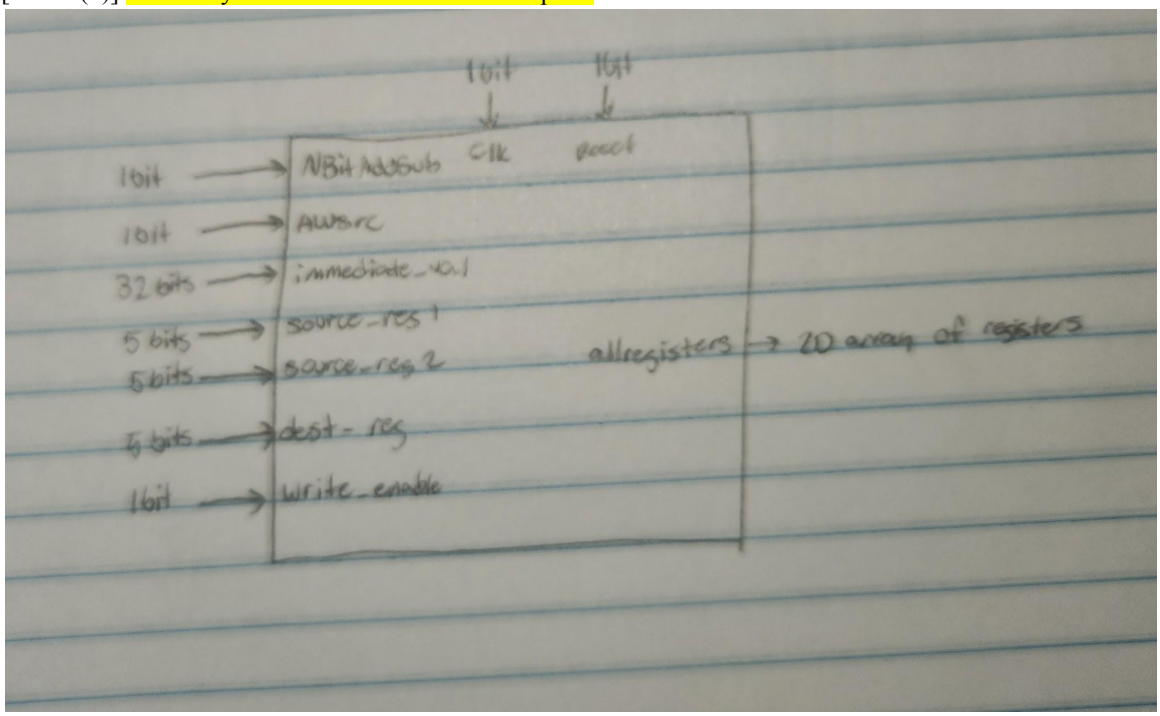
[Part 2 (h)] Draw a (simplified) schematic for the MIPS register file, using the same top-level interface ports as in your solution describe above and using only the register, decoder, and mux VHDL components you have created.



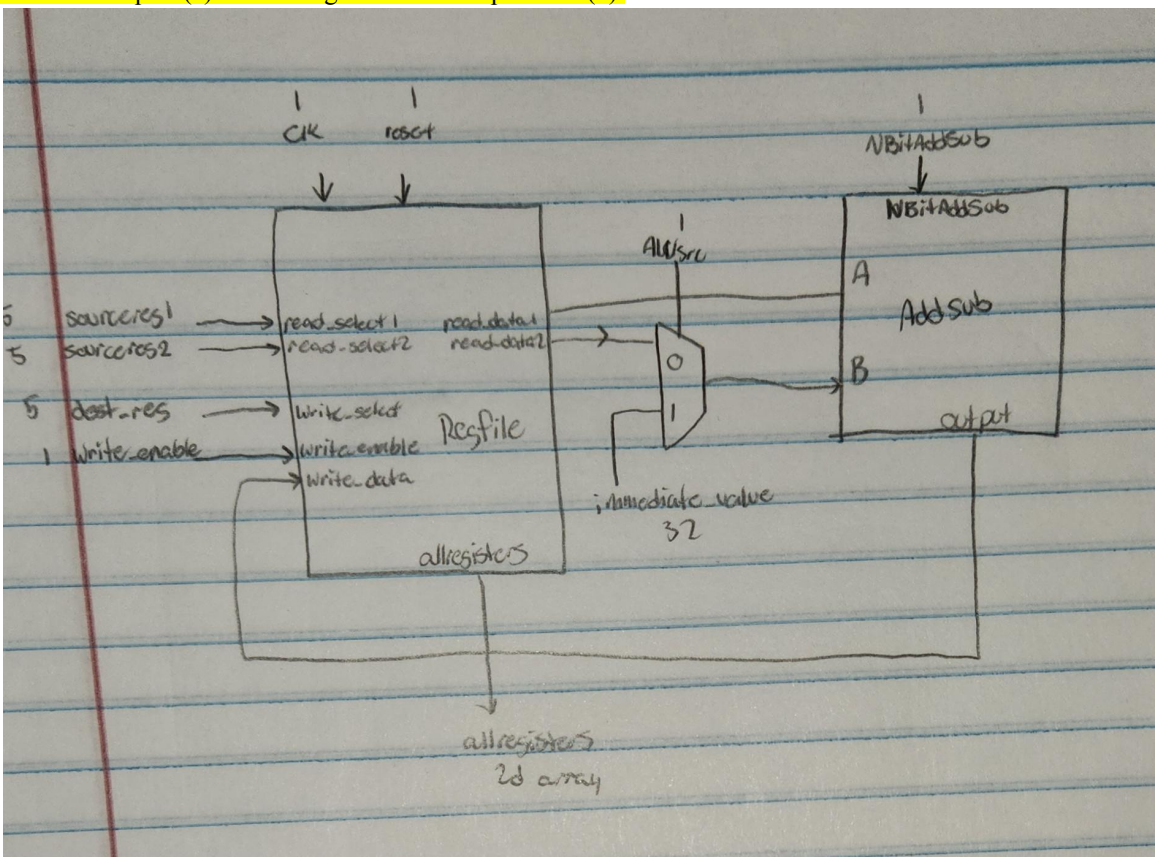
[Part 2 (i)] Waveform.



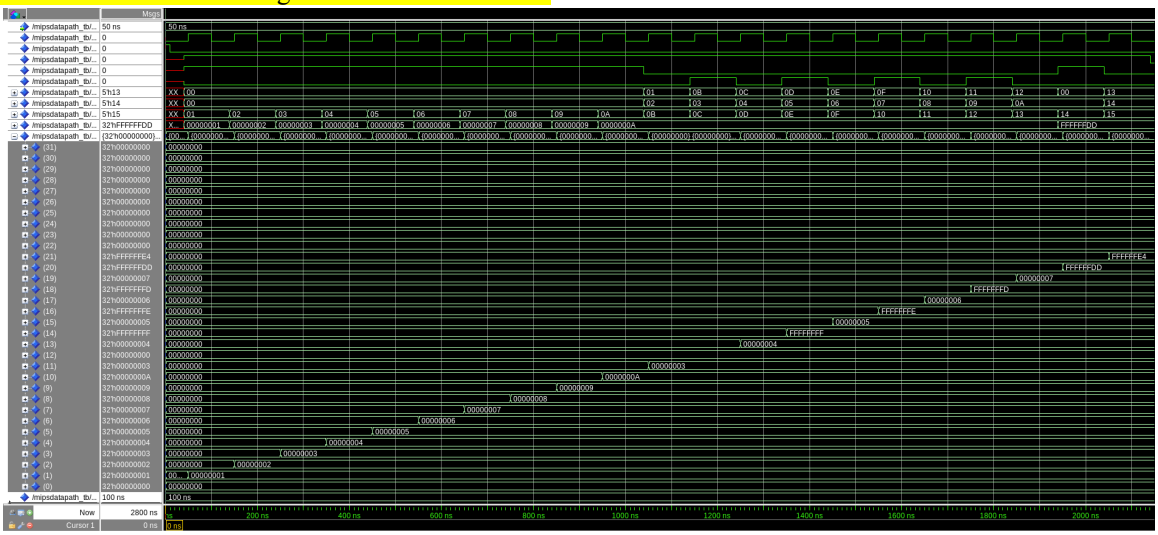
[Part 3 (b)] Draw a symbol for this MIPS-like datapath.



[Part 3 (c)] Draw a schematic of the simplified MIPS processor datapath consisting only of the component described in part (a) and the register file from problem (1).



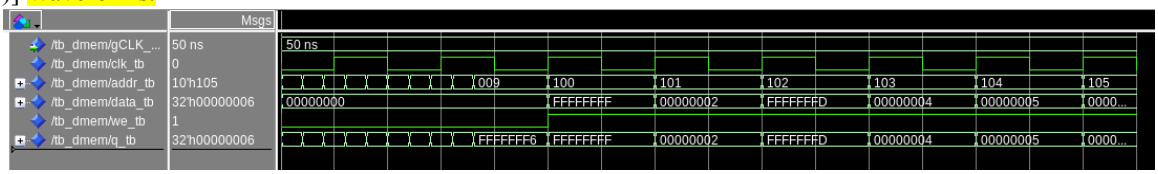
[Part 3 (d)] Include in your report waveform screenshots that demonstrate your properly functioning design. Annotate what the final register file state should be.



[Part 4 (a)] Read through the mem.vhd file, and based on your understanding of the VHDL implementation, provide a 2-3 sentence description of each of the individual ports (both generic and regular).

The generic port data width determines how many bits are stored at each memory address, and address width how many bits the memory addresses are. The “addr” input is the memory address the CPU is trying to extract data from. The “data” input is for writing data to memory, “we” is the write enable bit for memory, and “q” is the data output from the memory address in addr.

[Part 4 (c)] Waveforms.



[Part 5 (a)] What are the MIPS instructions that require some value to be sign extended? What are the MIPS instructions that require some value to be zero extended?

addi, subi, load, store, slti, sltiu

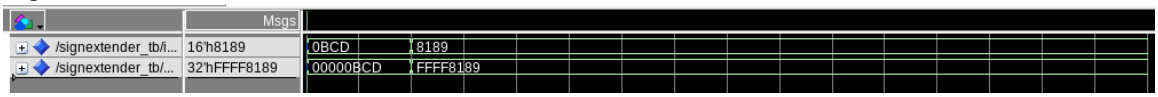
andi, ori,

[Part 5 (b)] what are the different 16-bit to 32-bit “extender” components that would be required by a MIPS processor implementation?

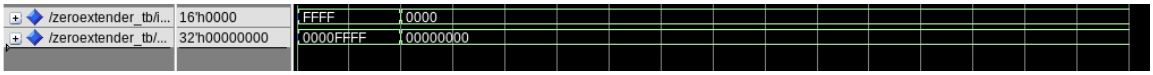
a sign extender and a zero extender

[Part 5 (d)] Waveform.

Sign Extended



Zero Extended



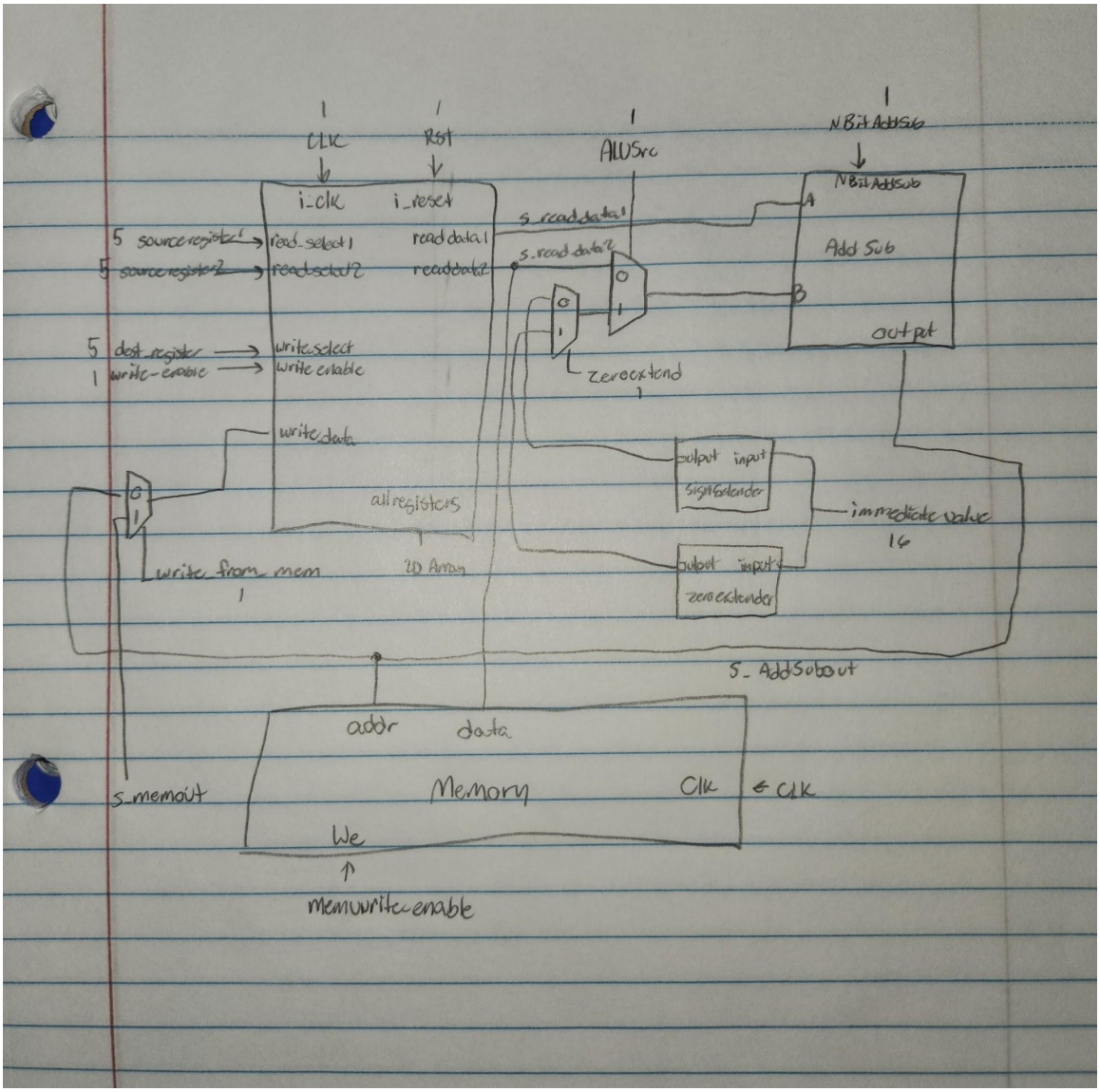
[Part 6 (a)] what control signals will need to be added to the simple processor from part 2? How do these control signals correspond to the ports on the mem.vhd component analyzed in part 3?

memwrite_en: 1 when writing to memory , else 0

zeroextend: 1 when immediate value needs zero extended, 0 for sign

write_from_mem: 1 when writing to regfile from memory, 0 when ALU

[Part 6 (b)] Draw a schematic of a simplified MIPS processor consisting only of the base components used in part 2, the extender component described in part 4, and the data memory from part 3.


1
CLK
1
RST
1
ALUSrc
1
NBitAddSub
i_clk
i_reset
NBitAddSub
A
Add Sub
B
output
5 source register1
5 source register2
read_select1
read_select2
read data1
read data2
s_read_data1
s_read_data2
5 dest_register
1 write_enable
write_select
write_enable
write_data
zeroextend
output input
signExtender
immediate value
16
all registers
write_from_mem
1
2D Array
output input
zeroExtender
5_AddSubout
addr
data
s_memout
Memory
Clk
CLK
We
memwrite_enable

[Part 6 (c)] Waveform.