Instructions:

- Print your name in the space provided below.
- This examination is closed book and closed notes, aside from the permitted one-page formula sheet and the MIPS reference card. No calculators or other computing devices may be used.
- Answer each question in the space provided. If you need to continue an answer onto the back of a page, clearly indicate that and label the continuation with the question number.
- If you want partial credit, justify your answers, even when justification is not explicitly required.
- There are 6 questions, priced as marked. The maximum score is 100.
- When you have completed the test, sign the pledge at the bottom of this page and turn in the test.
- Note that either failing to return this test, or discussing its content with a student who has not taken it is a violation of the Honor Code.

Do not start the test until instructed to do so!

Name _____
printed

Pledge: On my honor, I have neither given nor received unauthorized aid on this examination.

_____
signed

For questions 1 – 4, feel free to refer to the copy of Fig 5.24 from P&H that was distributed along with the test.

1. Note the list of MIPS instructions that the given datapath supports.

   a) [12 points] For which of those instructions is the component labeled Sign extend needed? For each such instruction, describe carefully how that component is used in the execution of the instruction.

   Sign extension is needed for instructions that include a 16-bit immediate field that must be passed to a 32-bit ALU in order to be added to another 32-bit value to compute a memory address. That applies with three of the supported instructions: lw, sw and beq.

   b) [12 points] For which of those instructions is the right-most component labeled Shift left 2 needed? For each such instruction, describe carefully how that component is used in the execution of the instruction.

   Shift left 2 multiplies its input value by 4. The right-most unit takes its input from the sign extension unit, and is used during execution of the beq instruction, to convert the immediate value in the instruction to a word-based offset from the updated program counter value.

2. [12 points] For which of the supported instructions should the RegDst control line to the multiplexor between the instruction memory and the register file be set to 1? For each such instruction, explain why this setting is necessary.

   The RegDst control line determines whether the number of the register to be written is taken from bits 20-16 or bits 15-11 of the current instruction. Note that bits 20-16 are also used to specify the number of the second register to be read; when RegDst is set to 1, the instruction will read from one register (specified by instruction bits 25-21) and write to another (specified by instruction bits 20-16).

   This is necessary for the lw instruction, since it reads the address base value from one register and stores the value retrieved from memory into another register.

3. [12 points] For which of the supported instructions should the RegWrite control line to the register file be set to 1? For each such instruction, explain why this setting is necessary.

Setting RegWrite to 1 causes the register file to write the value at Write data to be stored into the register specified at Write register. The R-type instructions, beq, and lw write a value back into the register file, so RegWrite must be set to 1 for those.

4. [12 points] Consider the two multiplexors in the upper right corner of the datapath diagram. Is it logically possible to combine these into a single multiplexor (obviously taking more than two inputs)? If not, explain why not. If yes, draw a simple diagram of the combined multiplexor, label the inputs clearly, and explain how the control bit(s) for it would be set.

Yes, since the output from the first MUX is passed to the second one. We would need a single MUX with three input lines, and therefore with two control bits.

5. The datapath supports execution of the instructions shown in the table below. Some part of the control logic must determine which instruction is to be executed, from the `opcode` field of the machine instruction. Of course, for the R-type instructions, the specific instruction will be determined from the `funct` field of the instruction by the ALU control logic module.

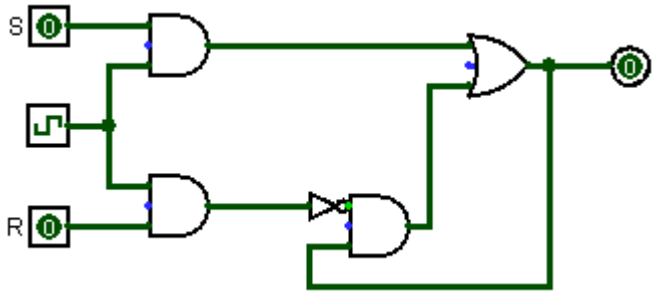
   The table below shows the opcode fields for the 9 relevant MIPS instructions, and one way of mapping those to a set of output bits that would indicate which particular instruction has been received:

	Opcode						Output Bits		
Instruction	c5	c4	c3	c2	c2	c1	o2	o1	o0
ADD	0	0	0	0	0	0	0	0	0
SUB	0	0	0	0	0	0	0	0	0
AND	0	0	0	0	0	0	0	0	0
OR	0	0	0	0	0	0	0	0	0
SLT	0	0	0	0	0	0	0	0	0
LW	1	0	0	0	1	1	0	0	1
SW	1	0	1	0	1	1	0	1	0
BEQ	0	0	0	1	0	0	0	1	1
J	0	0	0	0	1	0	1	0	0

   a) [16 points] Determine a minimal set of opcode bits needed to distinguish which of the 5 <u>different</u> output bit patterns should be produced. In other words, identify which specific columns of the opcode section of the table you would use in computing the output bits.

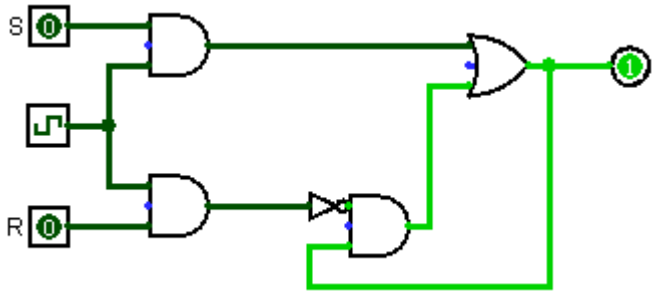
   b) [12 points] Describe precisely the procedure you would use the reduced table you identified in the previous question to construct a set of Boolean function for each output bit. Simply writing down that function is unacceptable (and unnecessary).

6. The circuit shown below is a clocked SR-latch:



a) [6 points] Suppose that the input S is set to 1 (high). Describe precisely how the output value will change as the clock cycles from low to high and back to low.

b) [6 points] Suppose that the circuit is in the state shown below:



Suppose that the input R is set to 1. Describe precisely how the output value will change as the clock cycles from low to high and back to low.