HW 1 Solution

Note to Students:
If you have a different solution to a particular problem or if you have other things that you would like the TA to be aware of during grading, please contact the TA Jia Guo:

Problem 1

1.1) Clock cycle time is determined by the critical path, which for the given latencies happens to be to get the data value for the load instruction:
For example,
lw $t0, –4($sp)

I-type format, MIPS Instruction formats

31-26	25-21	20-16	15-0
opcode	rs: register src	rt: register dest	offset
lw	$sp	$t0	-4

Without improvement 450 + 250 + 150 + 550 + 20 + 250 = 1670 ps **(5 pts)**
With improvement 450 + 250 + 150 + 130 + 550 + 20 + 250 =1800 ps **(5 pts)**

Without improvement:

Operation	Location	Delay	Output
reading the instruction memory	I-Mem	450ps	Instruction[31-0]
reading the base register $sp	Regs	250ps	Read data 1
computing memory address $sp-4	ALU	150ps	ALU result
reading the data memory	D-Mem	550ps	Read data
choosing the read data –4($sp)	Mux	20ps	Mux output, when selecting channel 1
storing data back to $t0	Regs	250ps	Register t0

1.2) (1/0.85) * (1670/1800) = 1.092, so the speedup is 1.092, it is faster than original architecture. **(10 pts)**

1.3) The speedup comes from changes in clock cycle time and changes to the number of clock cycles we need for the program: We need x% fewer instructions, but cycle time is 1600 instead of 1480, so we have a speedup of (1/x)*(1670/1800) **>=** 1.3, x **<=** 0.714 so percentage = 1 - x **>=** 0.286 **(10 pts)**, which means that at least 28.6% instructions need to be eliminated.

Problem 2

2.1) **18 pts**

add **(3pts)**	addi **(3pts)**	bne **(3pts)**	lw **(3pts)**	sw **(3pts)**	nor **(3pts)**
35/270	45/270	40/270	70/270	40/270	40/270
12.96%	16.67%	14.81%	25.93%	14.81%	14.81%

2.2) The data memory is used by LW and SW instructions, so the answer is:

25.93% + 14.81% = 40.74% **(5 pts)**
100%**(2 pts)**

2.3) 16.67% + 14.81% + 25.93% + 14.81% = 72.22% **(5 pts)**

The sign-extend circuit is actually computing a result in every cycle. The input of the sign-extend circuit is needed for ADDI (to provide the immediate ALU operand), BNE (to provide the PC-relative offset), and LW and SW (to provide the offset used in addressing memory) **(5 pts)**

Problem 3

This instruction is AND r11, r8, r3
Note that AND is bit by bit.

3.1)

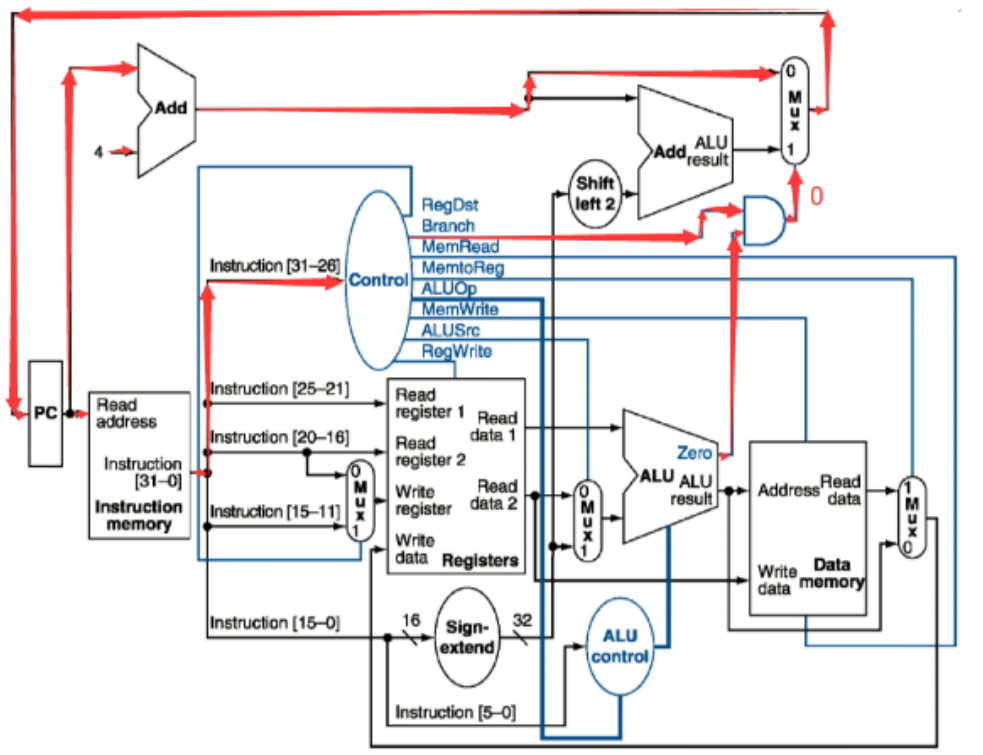
Sign-extend (3 pts)	**Shift-left-2 (2 pts)**
00000000000000000101100000100100	00000000000000001011000001001000 0

3.2)

ALUOp (3 pts)	**Instruction[5-0] (2 pts)**
10	100100

3.3)

New PC (2 pts)	**Path (3 pts)**
pc+4	PC to Add (PC+4) to branch Mux to PC



3.4)

WrReg Mux (2 pts)	ALU Mux (1 pts)	Mem/ALU Mux (1 pts)	Branch Mux (1 pts)
Instruction[15-11]: 11_{10} or 01011_2	79_{10}	99 And 79 = 67_{10}	PC+4

3.5)

ALU (2 pts)	Add (PC+4) (1 pts)	Add (Branch)(2 pts)
99 and 79	PC and 4	PC+4 and $00000000000000010110000010010000_2$

3.6)

Read Register 1 **(2 pts)**	Read Register 2 **(2 pts)**	Write Register **(2 pts)**	Write Data **(2 pts)**	RegWrite **(2 pts)**
8	3	11	67	1