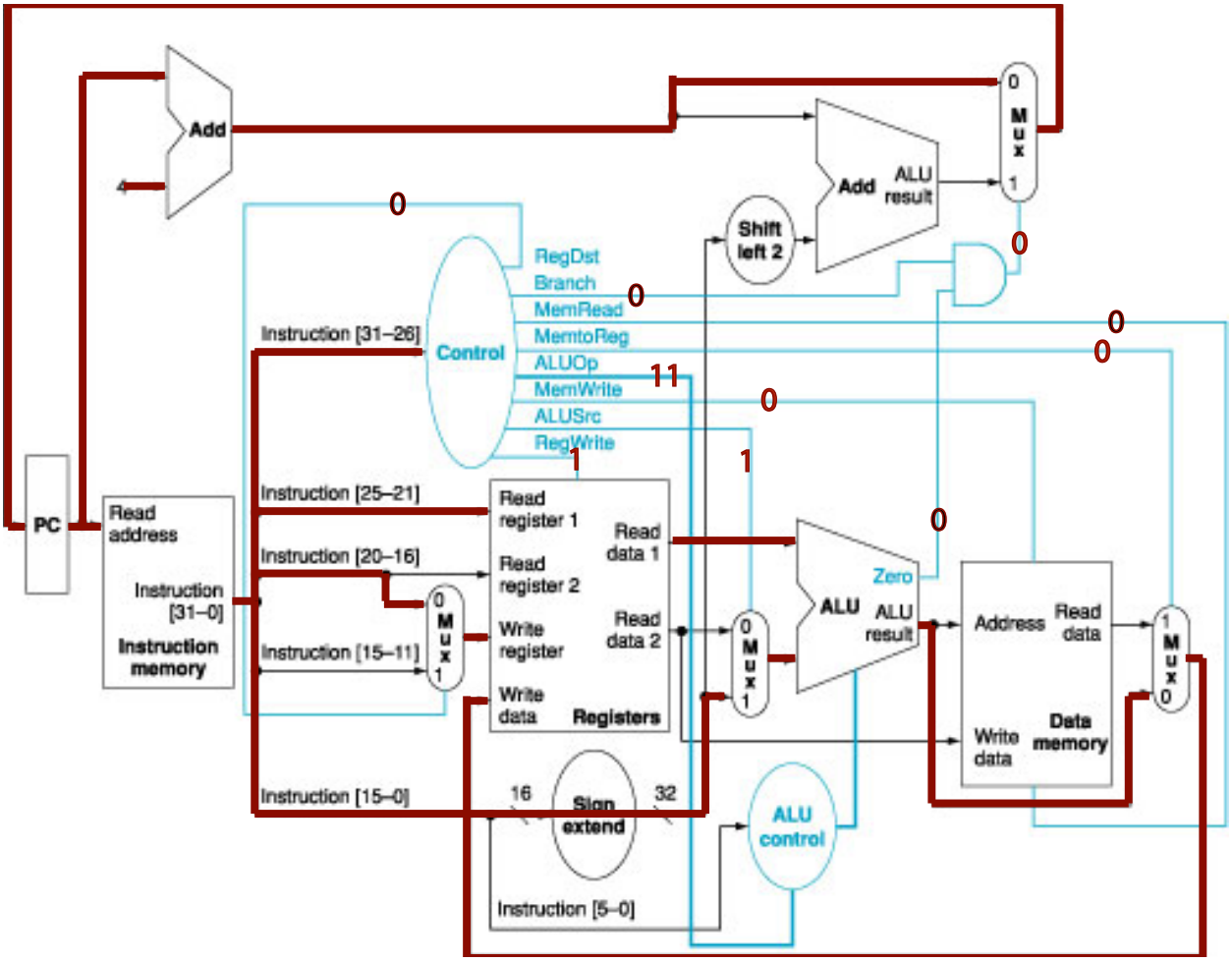
1a) 13.5*10^9 * 0.9 = cycles = 12.15*10^9
1b) 12.15*10^9/6*10^9 = 12.15/6 = 2.025 seconds

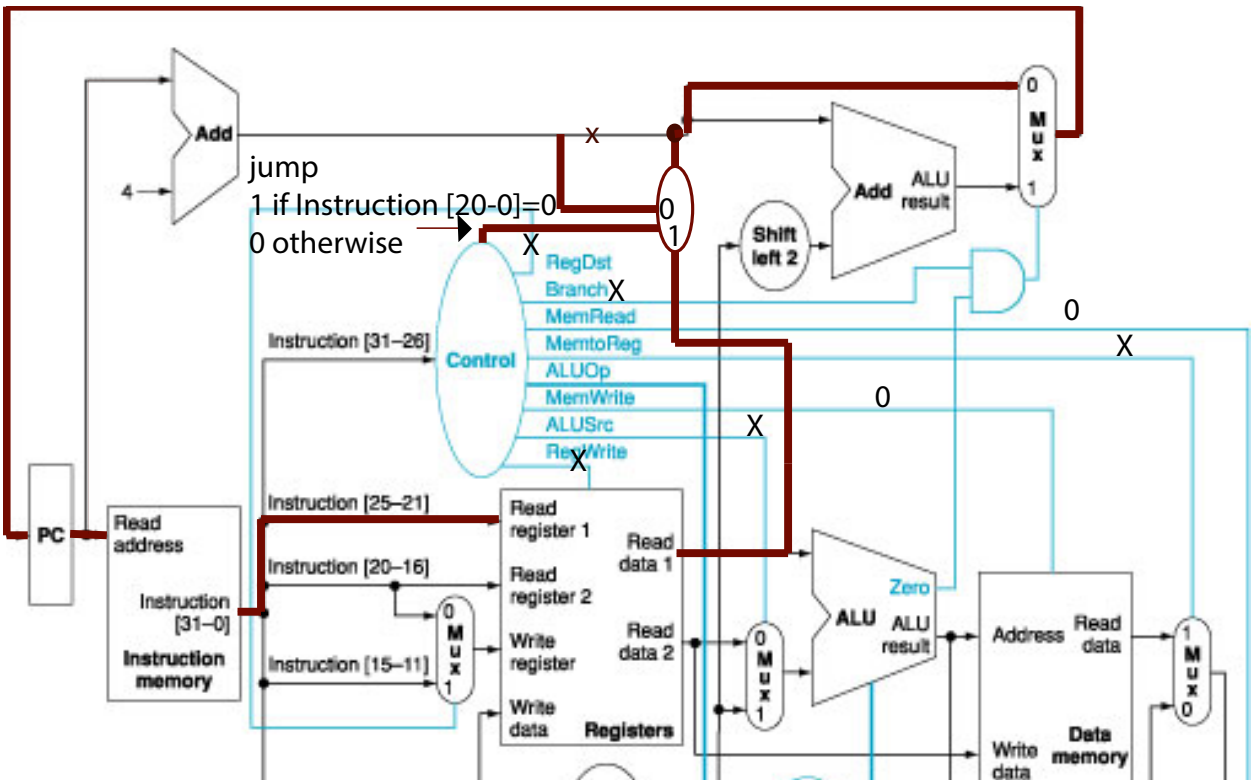
2a) I1=2.65
I2=2.7
I1 is faster at a ratio of 53/54
2b) I1=3.05
I2=2.65
I2 is faster at a ratio of 53/61
2c) I1=2.15
I2=2.8
I1 is faster at a ratio of 43/56
2d) C1: 5*10^9cycles per second/2.65 cycles per instruction= 1886792452 instructions per second.
C2: 5*10^9/3.05 = 1639344262 instructions per second.
C3: 5*10^9/2.15 = 2325581395 instructions per second. <———————
2e) C1: 4*10^9/2.7 = 1481481481 instructions per second.
C2: 4*10^9/2.65 = 1509433962 instructions per second. <———————
C3: 4*10^9/2.8 = 1428571428 instructions per second.
2f) C3 with I1.

3a) Branch = 1
No instruction except beq and j will work. They'll execute fine, but then they'll jump a strange Program Counter after if the result from the ALU happens to be 0. For example add $t0, $t1, $t2 works fine, except when $t1 = -$t2, in which case it will branch strangely.
3b) ALUOp1=1
No instruction except R-types will work. Instead of the intended operation being performed by the ALU, the operation will be determined by Instruction[5-0]. Some may work sometimes and not others.
3c) RegWrite=1
sw , beq, and j will not work. Some value will be written to some register for all of them, which is not supposed to happen.

4)


Add
4
0
RegDst
Branch
MemRead
MemtoReg
ALUOp
MemWrite
ALUSrc
RegWrite
Control
Instruction [31–26]
11
Shift left 2
Add
ALU result
M u x
PC
Read address
Instruction [31–0]
Instruction memory
Instruction [25–21]
Instruction [20–16]
Instruction [15–11]
Instruction [15–0]
Instruction [5–0]
Read register 1
Read register 2
Write register
Write data
Read data 1
Read data 2
Registers
16
Sign extend
32
ALU control
Zero
ALU
ALU result
Address
Read data
Write data
Data memory

5)


Add
4
jump
1 if Instruction [20-0]=0
0 otherwise
x
X
0
1
RegDst
Branch
MemRead
MemtoReg
ALUOp
MemWrite
ALUSrc
RegWrite
Control
Instruction [31–26]
Shift left 2
Add
ALU result
M u x
PC
Read address
Instruction [31–0]
Instruction memory
Instruction [25–21]
Instruction [20–16]
Instruction [15–11]
Read register 1
Read register 2
Write register
Write data
Read data 1
Read data 2
Registers
Zero
ALU
ALU result
Address
Read data
Write data
Data memory