Nick Petro

04/08/2016

COE 0147

Lab 8 Book Problems

Partner: Kevin Gilboy

**Q4.1:**

Q4.1.1

RegWrite = 1

MemRead = 0

ALU MUX = 0

MemWrite = 0

ALU Op = 10 (AND)

Reg MUX = 1

Branch = 0

Q4.1.2

PC – used for keeping track of what instruction the program is on, and calculating the subsequent instruction address

Add – used to increment the PC

Instruction memory – used for storing instructions to be fetched

Registers (including Reg MUX) – used as temporary memory that values can be recalled from to have operations applied to them

ALU (including ALU MUX) – used as the unit that actually does the AND function

Q4.1.3

Data memory – There was no data memory used in performing the AND operation. It takes no input and produces no output for this instruction.

Branch – There was no jump performed for this operation. Since the branch control signal is 0, the branch does not produce an output.

**Q4.2:**

Q4.2.1

PC

Reg MUX

Registers

ALU MUX  
ALU

Memory

Q4.2.1

Nothing

Q4.2.1

Nothing

**Q4.3:**

Q4.3.1

With this improvement: I-mem, Regs, MUX, ALU, Data Mem, MUX  
Latency = 400+200+30+(120+300)+350+30 = 1430 ps

Without this improvement: I-mem, Regs, MUX, ALU, Data Mem, MUX  
 Latency = 400+200+30+120+350+30 = 1130 ps

Q4.3.2

Speedup =

Q4.3.3

Cost to performance with this improvement:   
(1000+200+500+(100+600)+2000+2\*30+3\*10)/.832 = 5396.6  
Compared to without improvement: 5396.6/3890 = 1.39  
  
Cost to performance without this improvement  
(1000+200+500+100+2000+2\*30+3\*10)/1 = 3890

Compared to without improvement: 3890/3890 = 1.00  
  
The obvious choice is to not add the improvement. Comparing the cost to performance ratios, the processor without the “upgraded” ALU performs better for a lower cost.

**Q4.4:**

Q4.4.1

Since the processor only needs to fetch consecutive instructions, the cycle time would ignore the Add unit (since the Add function takes less time than I-Mem, the addition would happen in parallel to I-Mem). Therefore the critical path would just be I-Mem = 200 ps.

Q4.4.2

I-Mem + SignExtend + ShiftLeft + Add + MUX = 200 + 15 + 10 + 70 + 20 = 315 ps

Q4.4.3

I-Mem + Regs + MUX + ALU + MUX = 200 + 90 + 20 + 90 + 20 = 420 ps  
*It can be noted that this critical path is the same path for determining the MUX control signal for the branch*

Q4.4.4

Jump instructions that are PC relative

Q4.4.5

It is not on the critical path of any operation since other units (such as I-Mem) take much longer to complete. The shift left unit only has a latency of 10 ps.

Q4.4.6

Since beq requires a much longer path than add would, beq is the critical path. Because of this, and because the shift left unit is not on the critical path of beq, the critical path time will not be effected.