IT5002 Computer Systems and Applications
Tutorial 3

Tutorial Questions

Questions 1 and 2 refer to the complete datapath and control design in lectures 7 and 8. For your convenience, the complete datapath is attached at the end of this tutorial.

1. Let us perform a complete trace to understand the working of the complete datapath and control implementation. Given the following three hexadecimal representations of MIPS instructions:

    i.   `0x8df80000: lw $24, 0($15)`
    ii. `0x1023000C: beq $1, $3, 12`
    iii. `0x0285c822: sub $25, $20, $5`

   For each instruction encoding, do the following:

   a. Fill in the tables below. The first table concerns with the various data (information) at each of the datapath elements, while the second table records the control signals generated. Use the notation $8 to represent register number 8, [$8] to represent the content of register number 8 and Mem(X) to represent the memory data at address X.

Registers File				ALU		Data Memory	
RR1	RR2	WR	WD	Opr1	Opr2	Address	Write Data

[Wr = Write; Rd = Read; M = Mem; R = Reg]

RegDst	RegWr	ALUSrc	MRd	MWr	MToR	Brch	ALUop	ALUctrl

   b. Indicate the value of the PC after the instruction is executed.

2. With the complete datapath and control design, it is now possible to estimate the latency (time needed for a task) for the various type of instructions. Given below are the resource latencies of the various hardware components (ps = picoseconds = 10^{-12} second):

Inst-Mem	Adder	MUX	ALU	Reg-File	Data-Mem	Control/ ALUControl	Left-shift/ Sign-Extend/ AND
400ps	100ps	30ps	120ps	200ps	350ps	100ps	20ps

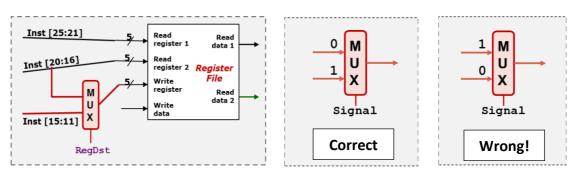Give the estimated latencies for the following MIPS instructions:

   (a) "SUB" instruction (e.g. `sub $25, $20, $5`)
   (b) "LW" instruction (e.g. `lw $24, 0($15)`)
   (c) "BEQ" instruction (e.g. `beq $1, $3, 12`)

What do you think the **cycle time** should be for this particular processor implementation?

Hint: First, you need to find out the **critical path** of an instruction, i.e. the path that takes the longest time to complete. Note that there could be several <u>parallel paths</u> that work more or less simultaneously.

3. [AY2013/14 Semester 2 Term Test #2]
   Mr. De Blunder made a *huge* mistake while making his own non-pipelined MIPS processor. He accidentally **swapped the two input ports for the RegDst multiplexer:**

For each of the following instructions, give:

   i. One example where the incorrect processor still gives the **right execution result.**
   ii. One example where the incorrect processor gives the **wrong execution result**.

If there is no suitable answer, please indicate "No Answer".

   (a) `add` (Addition)
   (b) `lw` (Load Word)
   (c) `beq` (branch-if-equal), provide the branch offset as immediate value.

4. Suppose the four stages in some 4-stage pipeline take the following timing: 2ns, 3ns, 4ns, and 2ns. Given 1000 instructions, what is the speedup (in two decimal places) of the pipelined processor compared to the non-pipelined single-cycle processor?

Datapath & Control