Quiz#1 for Computer Architecture, Spring 2017, Total Points = 100pts

1. The following table holds MIPS assembly code fragments (note that **$0 = $zero = 0**)

```
LOOP:                slt    $t2,   $zero, $t1
                     bne    $t2,   $zero, ELSE
                     j      DONE
ELSE:                addi   $s2,   $s2,   4
                     subi   $t1,   $t1,   1
                     j      LOOP
DONE:
```

   a. (8pts) Assume that the initial values of **$t1 = 10** (decimal 10), and **$s2 =0**. What is the value of **$s2** after executing this code segment? 40

2. **(20 pts)** The following MIPS code is used to compute **$v0** from integers **A=8** and **B=10.** Assume that **$a0** and **$a1** initially contain the integer values of A and B, respectively. **$v0** is used for storing the output.

```
Start:               add    $t0,   $zero,  $zero
Loop:                beq    $a1,   $zero,  Finish
                     add    $t0,   $t0,    $a0
                     sub    $a1,   $a1,    2
                     j      Loop
Finish:              sll    $t0,   $t0,    3
                     add    $v0,   $t0,    $zero
```

   (a) (10pts) What is the value of **$v0** after executing this code segment? 320

   (b) (10pts) How many instructions are executed in this code segment? 24

3. (56pts, 2pt each) Based on the function of the seven control signals and the datapath of the single-cycle MIPS CPU in Figure 1, complete the setting of the control lines in the table (use 0, 1, and X (don't care) ONLY) for each MIPS instruction. Note that X (don't care) can help to reduce the implementation complexity. Hence, you NEED TO put X whenever possible.

Instruction	Branch	ALUSrc	RegWrite	RegDst	MemtoReg	Memory Write	Memory Read	ALUOp1	ALUOp0
sw	0	1	0	X	X	1	0/x	0	0
add	0	0	1	1	0	0	0/x	1	0
beq	1	0	0	X	X	0	0/x	0	1
lw	0	1	1	0	1	0	1	0	0

4. Suppose that the operation time for the major function block in Figure 1 is as follows: (a) Memory access (Read or Write): 12ns. (b) ALU operation or adder unit: 6 ns. (c) Register file access (Read or Write): 4 ns. The delay of Multiplexors and basic logic gates are ignored.

   - (16pts, 4pts each) Calculate the minimum execution time (i.e., the critical path) for each MIPS instruction based on "single-clock-cycle" implementation scheme

R-type instruction (e.g., `add $s1,$s2,$s3`): 14 ? ns	Branch (e.g., `beq $s1,$s2,Exit`): 10 ? ns
Store word (e.g., `sw $s1,8($s2)`): 22 ? ns	Load word (e.g., `lw $s1, 8($s2)`): 26 ? ns

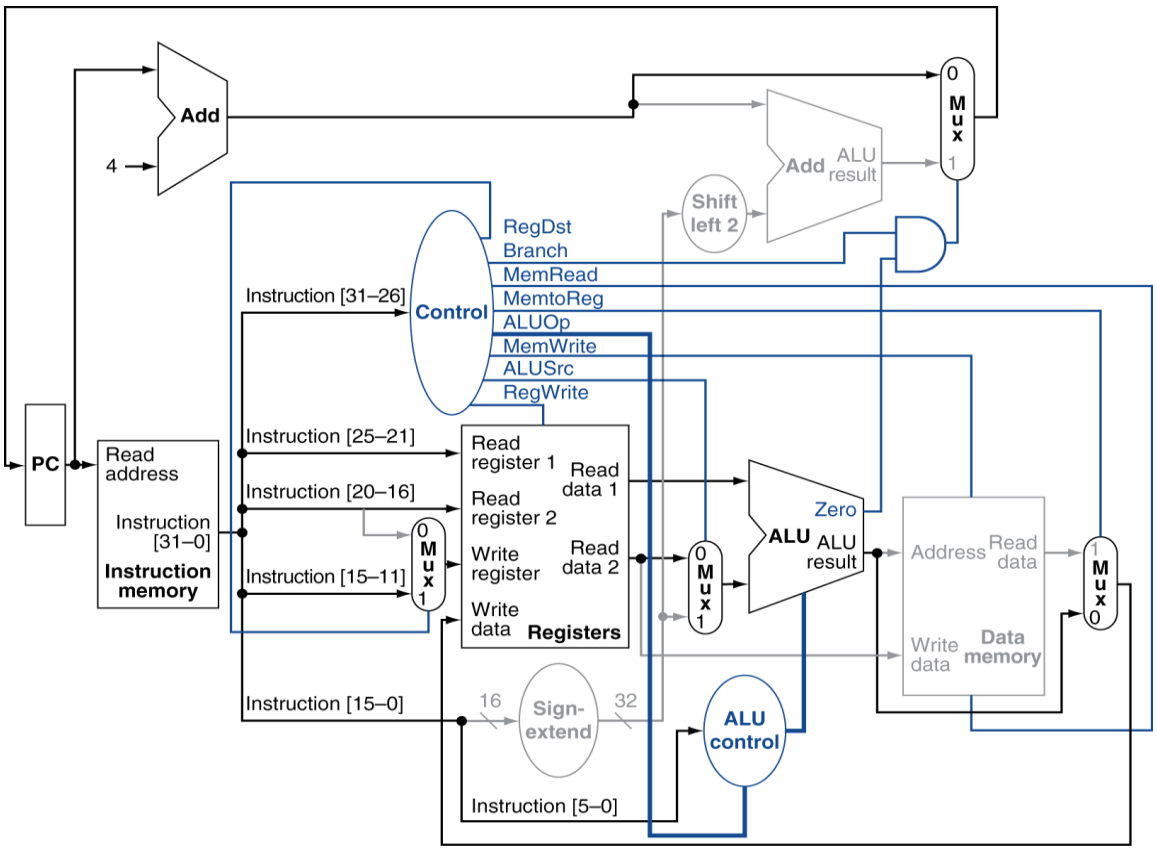

Figure 1: Single-cycle MIPS Design