

CS330 Architecture and Organization  
Assignment Chapter 4

Jingbo Wang

**Problem 1** — (Sections 4.1 and 4.4) Consider the following instruction:

Instruction: AND Rd, Rs, Rt

Interpretation:  $\text{Reg}[\text{Rd}] = \text{Reg}[\text{Rs}] \text{ AND } \text{Reg}[\text{Rt}]$

- (a) (3 points) What are the values of control signals (RegWrite, ALU operation, MemRead, MemWrite, Branch) generated by the control unit in Figure 4.2 on p. 247 for the above instruction?
- (b) (3 points) Which resources (blocks) perform a useful function for this instruction?
- (c) (3 points) Which resources (blocks) produce outputs, but their outputs are not used for this instruction? Which resources produce no outputs for this instruction?

*Answer:*

- (a) RegWrite: 1  
ALU operation: add  
MemRead: 0  
MemWrite: 0  
Branch: 0
- (b) All except Data Memory and Branch.
- (c) Outputs that are not used: Branch  
No outputs: Data Memory

**Problem 2** — (Section 4.1) The basic single-cycle MIPS implementation in Figure 4.2 can only implement some instructions. New instructions can be added to an existing Instruction Set Architecture (ISA), but the decision whether or not to do that depends, among other things, on the cost and complexity the proposed addition introduces into the processor datapath and control. Consider the new instruction (that uses registers Rs and Rd) for both the base address and offset to load a word from data memory:

Instruction LWI Rt, Rd(Rs)

Interpretation:  $\text{Reg}[\text{Rt}] = \text{Mem}[\text{Reg}[\text{Rd}] + \text{Reg}[\text{Rs}] ]$

- (a) (2 points) Is this an R-type, I-type, J-type instruction, or do we need to invent a new instruction type to implement LWI?
- (b) (3 points) Which existing blocks (if any) can be used for this instruction?
- (c) (3 points) Which new functional blocks (if any) do we need for this instruction?
- (d) (5 points) What are the values of control signals (RegWrite, ALU operation, MemRead, MemWrite, Branch) generated by the control unit in Figure 4.2? What new signals do we need (if any) from the control unit to support this instruction?

*Answer:*

- (a) It is R-type, because it uses Rt, Rd, Rs in the interpretation.
- (b) Blocks can be used:
  1. Data Memory: MemRead,
  2. RegWrite,
  3. ALU operation: add
- (c) Add a third input to the existing ALU for  $\text{Reg}[\text{Rd}] + \text{Reg}[\text{Rs}]$ .
- (d) RegWrite: 1  
 ALU operation: add  
 MemRead: 1  
 MemWrite: 0  
 Branch: 0  
 We do not need to add another block in this situation.

**Problem 3** — (Section 4.3) Refer to Figure 4.11 on p. 258 of the text. In your answers to these questions, refer to the parts of the processor by these names:

- Instruction memory
- Data memory
- Register file
- Sign extend unit
- Shift-left 2 unit
- ALU (the full-capability ALU located between the Register file and Data memory)
- Add PC+4 (located in upper-left of diagram)
- Add PC offset (located in upper-right of diagram)

Determine which parts of the processor are used when executing the following instructions.

- (a) (4 points) `add $t0, $t1, $t2`
- (b) (4 points) `addi $s0, $s1, -1`
- (c) (4 points) `beq $sp, $gp, label0` (Assume the branch does **not** take place.)
- (d) (4 points) `bne $t0, $t1, label1` (Assume the branch takes place.)
- (e) (4 points) `sw $t0, 1024($s0)`

*Answer:*

- (a) Instruction memory, Register file, ALU, Add PC+4.
- (b) Instruction memory, Register file, Sign extend unit, ALU, Add PC+4.
- (c) Instruction memory, Register file, Sign extend unit, ALU, Add PC+4,
- (d) Instruction memory, Register file, Sign extend unit, Shift-left 2 unit, ALU, Add PC+4.
- (e) Instruction memory, Data memory, Register file, Sign extend unit, ALU, Add PC+4, Add PC offset.

**Problem 4** — (Section 4.3) Suppose the processor is performing the instruction `add $t0, $t1, $t2`.

1. (2 points) Which input should the `ALUSrc` mux select: Read data 2, Sign extension unit, or is this a don't care?
2. (2 points) Which input should the `MemtoReg` mux select: Read data, ALU Result, or is this a don't care?
3. (2 points) Which input should the `PCSrc` mux select: Add PC+4, Add PC offset, or is this a don't care?

*Answer:*

- (1) Read data 2
- (2) ALU Result
- (3) Add PC+4,

**Problem 5** — (Section 4.3) Suppose the processor is performing the instruction `lw $t0, -4($s0)`.

1. (2 points) Which input should the `ALUSrc` mux select: Read data 2, Sign extension unit, or is this a don't care?
2. (2 points) Which input should the `MemtoReg` mux select: Read data, ALU Result, or is this a don't care?
3. (2 points) Which input should the `PCSrc` mux select: Add PC+4, Add PC offset, or is this a don't care?

*Answer:*

- (1) Sign extension unit
- (2) Read data
- (3) Add PC+4,

**Problem 6** — (Section 4.3) Suppose the processor is performing the instruction `bne $a0, $zero, loop`.

1. (2 points) Which input should the `ALUSrc` mux select: Read data 2, Sign extension unit, or is this a don't care?
2. (2 points) Which input should the `MemtoReg` mux select: Read data, ALU Result, or is this a don't care?
3. (2 points) Which input should the `PCSrc` mux select: Add PC+4, Add PC offset, or is this a don't care?

*Answer:*

- (1) Read data 2
- (2) don't care
- (3) add PC offset