Name: Id#

COE 301/ICS 233, Term 171

Computer Architecture & Assembly Language

Quiz# 5 Solution

Date: Tuesday, Nov. 28, 2017

# **Q1.** Consider the single-cycle datapath and control given below along with ALU design for the MIPS processor implementing a subset of the instruction set:

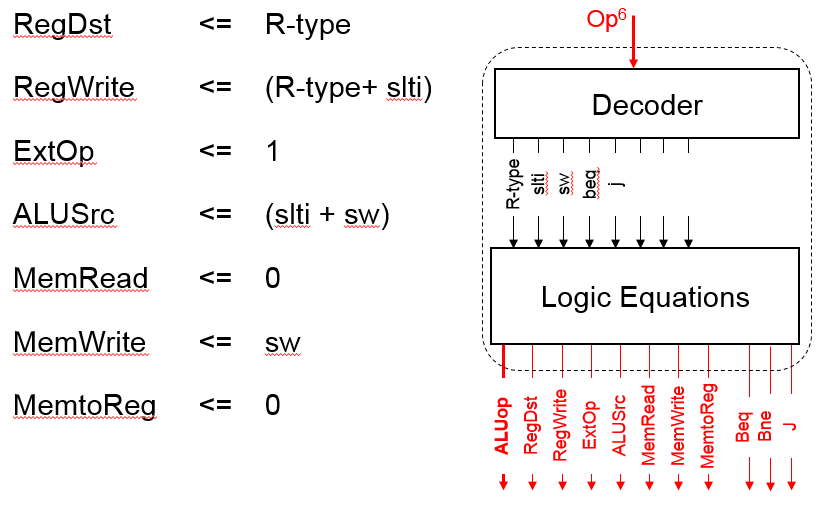


## 

## Show the control signals generated for the execution of the following instructions by filling the table given below: **(5 points)**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Op | RegDst | RegWrite | ExtOp | ALUSrc | ALUOp | Beq | Bne | J | MemRead | MemWrite | MemtoReg |
| R-type | 1 = Rd | 1 | x | 0=BusB | R-type | 0 | 0 | 0 | 0 | 0 | 0 |
| slti | 0 = Rt | 1 | 1=sign | 1=Imm | SLT | 0 | 0 | 0 | 0 | 0 | 0 |
| sw | x | 0 | 1=sign | 1=Imm | ADD | 0 | 0 | 0 | 0 | 1 | x |
| beq | x | 0 | x | 0=BusB | SUB | 1 | 0 | 0 | 0 | 0 | x |
| j | x | 0 | x | x | x | 0 | 0 | 1 | 0 | 0 | x |

## Excluding the ALUOp, Beq, Bne and J signals, show the design of the control unit for the control signals given in the table above based on the given instructions. Assume that the opcode of these instructions is a 6-bit opcode such that the opcode for R-type instructions is 0, the opcode for slti is 1, the opcode for sw is 2, and so on for the rest of the instructions. **(5 points)**



## Show the design of the Next PC block. **(4 points)**

## 

## We wish to add the following instructions to the MIPS single-cycle datapath. Add any necessary datapath modifications and control signals needed for the implementation of these instructions. Show only the **modified** and **added** components to the datapath.

* 1. sra **(3 points)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | Meaning | Format | | | | | |
| sra rd, rt, imm5 | rd= rt>>imm16 | Op6 = 0 | 0 | rt5 | rd5 | Imm5 | f6=3 |

For the sra instruction, examining the ALU one can see that the shift amont is coming through the A-input of the ALU and the operand to be shifted comes through the B input of the ALU. Thus, we need-to add a MUX on the A-input to select between the output of a register and the immediate values. This MUX needs to select only between the least significant 5 bits of BusA and bits 6 to 10 from Imm16. The modified part in the datapath is shown below:

**E**xt

A

L

U

ALUCtrl

BusA

BusB

ALUSrc

zero

Imm16

Imm32

**m**

**u**

**x**

0

1

0

**m**

**u**

**x**

0

1

0

Shifti

ExtOp

* 1. jr **(3 points)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instruction | Meaning | Format | | | | | |
| jr rs | PC=rs | op6 = 0 | rs5 | 0 | 0 | 0 | 8 |

For this instruction, the changes required in the datapath to implement it is to load the PC from BusA, which is driven by the RS field. Thus, we need to add a MUX to select the target address to be loaded in the PC either from the output of the MUX choosing between the address from NextPC block and incremented PC or from BusA. The required changes are shown below:

PC

00

+1

30

**m**

**u**

**x**

0

1

0

jr

**m**

**u**

**x**

0

1

0

PCSrc

BusA

Address from NextPC

## Assume that the propagation delays for the major components used in the datapath are as follows:

* + - Instruction and data memories: 120 ps
    - ALU and adders: 30 ps
    - Register file access (read or write): 14 ps
    - Main control: 8 ps
    - ALU control: 7 ps

Ignore the delays in the multiplexers, PC access, extension logic, and wires. What is the cycle time for the single-cycle datapath given above? **(3 points)**

Cycle Time = IM + max(Main Control+ALU Control, Register Reading) + ALU + DM + Register Write

= 120 ps + 15 ps + 30 ps + 120 ps+ 14 ps = 299 ps