**CS210: Computer architecture**

**Lab 09:Study of MIPS multi- Cycle Processor implementation**

**Name**: Amit kumar yadav

**Roll No**: 2101CS09

**Email**: amit\_2101cs09@iitp.ac.in

**Task 1:**

**Study the given multi-cycle implementation of the processor, and identity error in the design (if any). Demonstrate with examples that the design implements jump, addi, r-type, beq, lw and sw correctly. For each example you can create separate library component and load memory content file. Enter the examples used in the record with relevant information for demonstration.**

(2**0 points)**

Your Code here;

**For addi:**

.text

addi $t0, $0, 5

**For jump:**

.text

j label

label:

**For add r-type**

.text

addi $t0, $0, 10

addi $t1, $0, 5

add $t2, $t1, $t0

**For lw-sw**

.text

addi $t0, $0, 10

sw $t0 ,0 ($t1)

lw $t2, 0 ($t1)

**For beq**

.text

addi $t0,$0,10

addi $t1,$0,10

beq $t1,$t0,Done

add $t2,$t1,$t0

Done:

**Task 2:**

Run the following code in the given MIPS multi-cycle design. How many cycles are required to run the code? What is the CPI of this program?

# MIPS assembly code

add $s0, $0, $0

add $s1, $0, $0

addi $t0, $0, 10

loop:slt $t1, $s0, $t0

beq $t1, $0, done

add $s1, $s1, $s0

addi $s0, $s0, 1

j loop

done:

Ans:

Your assembled code here

v2.0 raw

00008020

00008820

2008000a

0208482a

11200003

02308820

22100001

08100003

**CPI**

CPI = Number of Cycles /Number of Instructions

Number of Cycles = 199

Number of Instructions = 55

CPI = 199/55 = 3.61

**CPI = 3.61**

**Task 3:**

***Write ASCII values of your name full name to the memory using appropriate instruction, and write program to find the sum of those values using the given Multi-cycle MIPS design.* Enter the machine code in hex in the r*ecord* for demonstration*.***

(4**0 points)**

v2.0 raw

20020048

ac420001

20420049

20030049

ac630004

20420052

20030052

ac630005

20420045

20030045

ac630006

2042004E

2003004E

ac630007

ac420010

**Task 3:**

**Implement a new instruction in the given multi-cycle design and test using appropriate program. Add the new design as a library component.**

(4**0 points)**

**Submission :**  Demonstrate to TA

.text

addi $t0, $0, 10

addi $t1, $0, 5

slt $t2, $t1, $t0

v2.0 raw

2008000a

20090005

0128502a