CS 311, Assignment 4 - Report

Abhishek Raj (180010002), Harsh Raj (180010017), Rishit Saiya (180010027)

October 29, 2020

1 Abstract

In this assignment, we updated our ***simulate*** function to a loop implements following algorithm:

```
while (not end of simulation) {
    performRW
    performMA
    performEX
    performOF
    performIF
    increment clock by 1
}
```

Here in, each stage we operated on the output generated by the previous stage in the prior cycle. If the input latches to the stage that has invalid content, then stage must do nothing and continue.

2 Statistics Table

The following table (Table 1) touches upon the number of cycles taken by each benchmark program, the number of times the OF stages that needed to stall because of a data hazard & the number of times an instruction on a wrong branch path entered the pipeline.

3 Interpretation and Comments

We see that the number of cycles taken by each program with pipeline implementation dropped substantially to half of the program without the pipeline. In an ideal case, it may reduce to one-fifth of total number of cycles. While in case of data hazards, we additionally add NOP instructions to avoid hazards which inturn increase the total number of cycles.

Similarly, the number of wrong branch instructions and number of OF instruction stalls are also affected on the same lines with above reasons to avoid hazards.

File Name	No.of Cycles	No. of Wrong Branch Instructions	No. of OF Stalls
descending.out	658	220	126
evenorodd.out	19	4	10
fibonacci.out	157	36	44
palindrome.out	124	18	51
prime.out	79	28	19

Table 1: Statistics Table