

CS311-Lab4

Josyula V N Taraka Abhishek - 200010021

Karthik JP - 200010022

October 2022

1 Data

TEST CASE	Hash of state	Number of instructions	Number of cycles
descending.out	255541867	277	277
evenorodd.out	-224294686	6	6
fibonacci.out	-1518357572	78	78
palindrome.out	155317940	49	49
prime.out	-1414219998	29	29

1) Before Pipeline

TEST CASE	Hash of state	OF stalls	Wrong branch instructions in pipeline	Number of cycles
descending.out	255541867	126	220	658
evenorodd.out	-224294686	10	4	19
fibonacci.out	-1518357572	44	36	157
palindrome.out	155317940	51	18	124
prime.out	-1414219998	19	28	79

2) After Pipeline

2 Observations from data

- We observe that the hash of the state of processor is the same in both the cases so our program is functionally correct as observed from data.
- A large number of cycles are wasted from wrong branches, we can attempt to fix this using a branch predictor.
- The ideal throughput for a pipelined processor should be 5 times that of the previous case
- The pipelined processor performed 2.45 times better in fibonacci case