Computer Architecture Lab
Assignment 5

Om Patil (200010036)
Hrishikesh Pable (200010037)

October 2022

1 Introduction

This assignment involves developing a pipelined processor simulator which also considers the memory latency associated with memory access and ALU operations. We do this using a discrete event simulation model.

2 Benchmarks

Benchmarks for the provide sample programs in the current processor simulator are as follows,

Program	Instuctions		Stalls	Incorrect Branches	Cycles	IPC	Frequency (GHz)
	Static	Dynamic					
Descending	21	277	159	88	15325	0.0181	1
Even Odd	9	6	9	0	254	0.0236	1
Fibonacci	21	78	60	16	3926	0.0199	1
Palindrome	16	49	72	7	2311	0.0212	1
Prime	16	29	21	5	1402	0.0207	1

This is for the following latency configuration,

- MainMemory - 40 Cycles
- ALU - 1 Cycle
- Multiplier - 4 Cycles
- Divider - 10 Cycles

The same benchmark when all latency are assumed to be zero is as follows,

Program	Instuctions		Stalls	Incorrect Branches	Cycles	IPC	Frequency (GHz)
	Static	Dynamic					
Descending	21	277	159	88	721	0.384	1
Even Odd	9	6	9	0	20	0.300	1
Fibonacci	21	78	60	16	189	0.413	1
Palindrome	16	49	72	7	143	0.343	1
Prime	16	29	21	5	67	0.433	1

2.1 Static Instructions

This metric measures the number of static instruction present in the program.

2.2 Dynamic Instructions

This metric measures the number of dynamic instructions executed during the simulation of the program on the processor.

2.3 Stalls

The number of times a data dependency occurs in the pipeline. This implies that 1 bubble was created post the OF stage.

2.4 Incorrect Branches

The number of dynamic branch instructions for which the branch taken was incorrect, and 2 bubbles had to be created in the pipeline at the OF and IF stages, invalidating the instructions present in them.

2.5 Cycles

This metric measures the number of processor cycles required to complete execution of the program.

2.6 IPC

This metric measures the Instructions Executed per Cycle during the simulation.

2.7 Frequency

This metric measures the frequency in GHz of instructions executed in the simulation. We currently assume that all stages take exactly 1 nanosecond to execute.

3 Observations

The introduction of memory access latency and ALU execution latency has lead to the same 5 programs taking a much larger number of cycles to execute. The IPC has reduced drastically due to this rise in the number of cycles. As expected the latency has no effect on the number of stalls and incorrect branches.

It can also be observed that the IPC of programs with larger number of memory accesses such as `Descending` and `Fibonacci` is hit harder by the introduction of large memory latency. Programs with relatively low number of memory accesses such as `Palindrome` exhibit the exact inverse trend.

The frequency of the clock is 1 GHz since it takes 1 ns to execute one cycle, ie. the execution of the 5 stages run in parallel.