ENCS437 **Quiz#3** **Nov 22, 2016**

Student Name: _____Solution_____Student ID: _____

Given the operation times as follow:

Instruction memory access time = 190 ps,
Register file read access time = 150 ps,
ALU delay (basic instructions) = 190 ps,
Data memory access time = 190 ps
Register file write access = 150 ps
ALU delay (multiply/divide) = 550

Assume the following instruction mix:
30% ALU, 15% multiply/divide, 15% load, 15% store, 15% branch, and 10% jump.

a) What is the total delay for each instruction class and the clock cycle for a **single cycle** MIPS implementation?
b) Assume we fix the clock cycle to 200 ps for a **multi-cycle** CPU, what is the average CPI? And speedup over a single cycle?
c) If we convert the multi-cycle into 5 stages **pipelined** CPU, what would be the execution time and speedup over the single cycle and the multi-cycle?
d) If the load instructions cause one stall to be inserted into the pipeline, and the multiply/divide instructions cause two stalls to be inserted, what would be the increase in the pipeline CPI over the ideal CPI? Re-calculate speedups of part (c)?

Solution:

Instruction Class	Instruction Memory	Register Read	ALU	Data Memory	Register Write	Total Delay (ps)
Basic ALU	**190**	**150**	**190**		**150**	**680**
Mul & Div	**190**	**150**	**550**		**150**	**1040**
Load	**190**	**150**	**190**	**190**	**150**	**870**
Store	**190**	**150**	**190**	**190**		**720**
Branch	**190**	**150**	**190**			**530**
Jump	**190**	**150**				**340**

ENCS437 **Quiz#3** **Nov 22, 2016**

Student Name: _____Student ID: _____

a. Max delay is when performing mult/div, thus for single cycle design, the clock cycle time is equal to **1040 ps**, i.e. clock rate = 0.96 MHz

b. For multi-cycle design with a clock period of 200 ps, i.e. clock rate = 5 MHz:

Basic ALU = 4 stages ➔ 4 cycles
Multiply & Divide = 4 stages but ALU requires three cycles ➔ 6 cycles
Load = 5 cycles
Store = 4 cycles
Branch = 3 cycles
Jump = 2 cycles

Average CPI = 0.3 * 4 + 0.15 * 6 + 0.15 * 5 + 0.15 * 4 + 0.15 * 3 + 0.1 * 2 = 4.1

Speedup of multi-cycle over single-cycle;
Sup(s → m) = (multi clk rate/signle clk rate)*(sigle CPI/multi CPI)
or
S(s → m) = ((single cycle execution time)*s-CPI)/((multi clk cyle time)*m-CPI)

= (1040 * 1) / (200 * 4.1) = **1.27**

In case you wanted to calculate average execution time;
Average CPU execution time = Cycle time*CPI*NumberOfInstructions
T = C x CPI x I
For any instruction, its execution time = 200ps * 4.1 = 820 ps

c. Ideal pipeline CPI = 1 running at 200 ps;

Execution time for any instruction = 200ps * 5 = 1000 ps
However Mult&Div requires three ALU cycles which means the pipeline should really be a seven stages pipeline, thus its execution time is equal to 1400 ps. The following speed up calculations hold true whether the pipe is 5 or 7 stages. This is because the pipeline CPI is independent of the number of stages.

Speedup pipeline over single cycle;
S(s →p) = (Ts*sCPI)/(Tp*pCPI) = (1040*1)(200*1) = **5.2**

Speedup pipeline over multi cycle;
S(m →p) = (Tm*mCPI)/(Tp*pCPI) = (200*4.1)(200*1) = **4.1**

d. None ideal pipeline;
pCPI" = 1 + pipeline cycles with stalls = 1 + (load stalls +div&multiply stalls)
= 1 + (0.15x1 + 0.15x2) = **1.45**

Speedup pipeline over single cycle;
S(s →p)* = (Ts*s-CPI)/(Tp*p-CPI") = (1040*1)(200*1.45) = **3.586**

Speedup pipeline over multi cycle;
S(m →p) = (Tm*m-CPI)/(Tp*p-CPI") = (200*4.1)(200*1.45) = **2.83**