Computer Architecture DAT105 Exercise Session 1

Piyumal Ranawaka
piyumal@chalmers.se

Chalmers University of Technology, Sweden

September 12, 2021

Table of Contents

Table of Contents

Problem 1.1

Two Possible improvements

1. First
   - Floating Point Arithmetical Unit
   - Reduce the time taken by each floating-point instruction by a factor 10
2. Second
   - Add more first-level data cache
   - Reduce the time taken by each load instruction by a factor 2

F_{fp} = Fraction of execution time spent in floating-point instructions
F_{ls} = Fraction of execution time spent in Load/Store instructions.

The executions of these two sets of instructions are non overlapping in time.

Problem 1.1 (Part a)

Using Amdahl's speedup, what should be the relation between the fractions F_{fp} and F_{ls} such that the addition of the floating-point units is better than the addition of cache space.

Amdahl's speedup

$$Speedup = \frac{1}{1 - F + \frac{F}{S}} \tag{1}$$

- F - Fraction of execution time that can be improved
- S - Improvement Factor

Problem 1.1 (Part a)

For floating-point units improvement.

$$Speedup(fp) = \frac{1}{1 - F_{fp} + \frac{F_{fp}}{S}} \tag{2}$$

$$Speedup(fp) = \frac{1}{1 - F_{fp} + \frac{F_{fp}}{10}} \tag{3}$$

For cache space improvement.

$$Speedup(ls) = \frac{1}{1 - F_{ls} + \frac{F_{ls}}{S}} \tag{4}$$

$$Speedup(ls) = \frac{1}{1 - F_{ls} + \frac{F_{ls}}{2}} \tag{5}$$

Problem 1.1 (Part a)

$$Speedup(fp) > Speedup(ls) \tag{6}$$

$$\frac{1}{1 - F_{fp} + \frac{F_{fp}}{10}} > \frac{1}{1 - F_{ls} + \frac{F_{ls}}{2}} \tag{7}$$

$$\frac{F_{fp}}{F_{ls}} > \frac{5}{9} \tag{8}$$

Problem 1.1 (Part b)

Suppose that, instead of being given the values of fractions F_{fp} and F_{ls}, you are given the fraction of floating-point instructions and the fraction of Loads and Stores instructions. You are also given the average number of cycles taken by floating-point operations and Loads/Stores.

Q1. Can you still find out which improvement is better based on these numbers? Explain why and how.

Q2. Can you still estimate the maximum speedups for each improvement using Amdahl's law? Why?

Problem 1.1 (Part b.1)

Q1. Can you still find out which improvement is better based on these numbers? Explain why and how.

We are only given fraction on floating point and load-store instructions to total instructions i.e. FI_{fp}, FI_{ls}.
We are also given the average number of cycles for floating point and load store operations or in other words cycle per instructions (CPI) i.e. CPI_{fp} , CPI_{ls}.

$$F_{fp} = \frac{T_{fp}}{T_{Total}} \tag{9}$$

$$F_{ls} = \frac{T_{ls}}{T_{Total}} \tag{10}$$

$$\frac{F_{fp}}{F_{ls}} = \frac{\frac{T_{fp}}{T_{Total}}}{\frac{T_{ls}}{T_{Total}}} \tag{11}$$

$$\frac{F_{fp}}{F_{ls}} = \frac{T_{fp}}{T_{ls}} \tag{12}$$

Problem 1.1 (Part b.1)

the execution time can be given as

$$T_{fp} = I_{fp} \times CPI_{fp} \times T_c \tag{13}$$

$$T_{ls} = I_{ls} \times CPI_{ls} \times T_c \tag{14}$$

Dividing two equations

$$\frac{T_{fp}}{T_{ls}} = \frac{I_{fp} \times CPI_{fp} \times T_c}{I_{ls} \times CPI_{ls} \times T_c} = \frac{I_{fp} \times CPI_{fp}}{I_{ls} \times CPI_{ls}} \tag{15}$$

$$\frac{T_{fp}}{T_{ls}} = \frac{I_{fp}}{I_{ls}} \times \frac{CPI_{fp}}{CPI_{ls}} \tag{16}$$

Problem 1.1 (Part b.1)

$$\frac{F_{fp}}{F_{ls}} = \frac{T_{fp}}{T_{ls}} = \frac{I_{fp}}{I_{ls}} \times \frac{CPI_{fp}}{CPI_{ls}} \tag{17}$$

We have already established that $\frac{F_{fp}}{F_{ls}} > \frac{5}{9}$, so we can use above equation and still find out which improvement is better.

Problem 1.1 (Part b.2)

Q2. Can you still estimate the maximum speedups for each improvement using Amdahl's law? Why?

Answer: NO

Why ?

We just have the ratio of F_{fp}/F_{ls} and therefore determining F_{fp} or F_{ls} individually is impossible

Problem 1.1 (Part b.2)

If we need to use the following equations to derive F_{fp} and F_{ls}:

$$T_{fp} = I_{fp} \times CPI_{fp} \times T_c \tag{18}$$

$$T_{ls} = I_{ls} \times CPI_{ls} \times T_c \tag{19}$$

$$T_{Total} = I_{Total} \times CPI_{Total} \times T_c \tag{20}$$

We don't have have a way to get T_{total}.

if we are use the ratio of above equations to derive F_{fp} and F_{ls}:

$$F_{fp} = \frac{I_{fp} \times CPI_{fp}}{I_{Total} \times CPI_{Total}} \tag{21}$$

$$F_{ls} = \frac{I_{ls} \times CPI_{ls}}{I_{Total} \times CPI_{Total}} \tag{22}$$

We don't have average CPI for all type of instructions nor the total number of instructions.

Problem 1.1 (Part c)

Q: What are fractions F_{fp} and F_{ls} such that a speedup of 50% (or 1.5) is achieved for each improvement deployed separately?

$$Speedup = 1.5 \tag{23}$$

Consider the floating point improvement

$$Speedup(fp) = \frac{1}{1 - F_{fp} + \frac{F_{fp}}{10}} \tag{24}$$

$$1.5 = \frac{1}{1 - F_{fp} + \frac{F_{fp}}{10}} \tag{25}$$

$$F_{fp} = 0.3707 \tag{26}$$

Problem 1.1 (Part c)

Consider the cache improvement

$$1.5 = \frac{1}{1 - F_{ls} + \frac{F_{ls}}{2}} \tag{27}$$

$$F_{ls} = 0.67 \tag{28}$$

Problem 1.1 (Part d)

It is decided to deploy the floating point unit first and to add cache space later on. In the original workload, fractions F_{fp} and F_{ls} are 30% and 20% respectively. What is the maximum speedup obtained by upgrading to the floating point units? Assuming that this maximum speedup is achieved by the floating-point unit upgrade, what is the maximum speedup of the cache upgrade with respect to the floating point unit upgrade?

Problem 1.1 (Part d)

$$F_{fp} = 0.3$$

$$F_{ls} = 0.2$$

Maximum speed up for floating point unit.

$$Speedup = \frac{1}{1 - 0.3 + \frac{0.3}{10}}$$

$$Speedup = \frac{1}{0.73} = 1.37$$

Problem 1.1 (Part d)

The maximum speed up for the load store improvement is required to be calculated with respect to new execution time.

Recall the speed up equation .

$$T_{exe(new)} = \frac{T_{exe(old)}}{Speedup}$$

$$T_{exe(new)} = \frac{T_{exe(base)}}{1.37} = T_{exe(base)} \times 0.73$$

Problem 1.1 (Part d)

Fraction of time spent in load store with respect to new execution time.

$$F_{ls-new} = \frac{F_{ls}}{0.73} = \frac{0.2}{0.73} = 0.274$$

Therefore, the maximum speedup of the cache upgrade after the floating point unit upgrade is

$$Speedup = \frac{1}{1 - 0.274 + \frac{0.274}{2}} = 1.1587$$

Table of Contents

Problem 1.3 (a)

- Take the ratio of average execution times, S_1
- Take the arithmetic means of speedups, S_2
- Take the harmonic means of speedups, S_3
- Take the geometric means of speedups, S_4

Machines	Program 1	Program 2	Program 3
Base Machine	1 sec	10 msec	10 sec
Base + FP Unit	1 sec	2 msec	6 sec
Base + cache	0.7 sec	9 msec	8 sec

Problem 1.3 (a)

Calculate average execution time.

Machines	Program 1	Program 2	Program 3	Average Execution Time
Base Machine	1	0.01	10	$(1+0.01+10)/3$
Base + FP Unit	1	0.002	6	$(1+0.002+6)/3$
Base + cache	0.7	0.009	8	$(0.7+0.009+8)/3$

Machines	Program 1	Program 2	Program 3	Average Execution Time
Base Machine	1	0.01	10	3.67
Base + FP Unit	1	0.002	6	2.334
Base + cache	0.7	0.009	8	2.903

Problem 1.3 (a)

S1 - Ratio of average execution time.

$$Ratio(Base + FP - Unit) = \frac{3.67}{2.334} = 1.57$$

$$Ratio(Base + cache) = \frac{3.67}{2.903} = 1.26$$

Machines	S1	S2	S3	S4
Base + FP Unit	1.57	?	?	?
Base + cache	1.26	?	?	?

FP Unit is better.

Problem 1.3 (a)

Calculate speed up over base machine

Machines	Program 1	Program 2	Program 3
Base Machine	1/1	10/10	10/10
Base + FP Unit	1/1	10/2	10/6
Base + cache	1/0.7	10/9	10/8

Machines	Program 1	Program 2	Program 3
Base Machine	1	1	1
Base + FP Unit	1	5	1.67
Base + cache	1.43	1.1	1.25

Problem 1.3 (a)

S2 - Arithmetic Mean.
Arithmetic Mean (Base + FP-unit)

$$= \frac{(1 + 5 + 1.67)}{3} = 2.55$$

Arithmetic Mean (Base + cache)

$$= \frac{1.43 + 1.1 + 1.25}{3} = 1.26$$

Machines	S1	S2	S3	S4
Base + FP Unit	1.57	2.55	?	?
Base + cache	1.26	1.26	?	?

Problem 1.3 (a)

S3 - Harmonic Mean.
Harmonic Mean (Base + FP Unit)

$$= \frac{3}{\frac{1}{1} + \frac{1}{5} + \frac{1}{1.67}} = 1.67$$

Harmonic Mean (Base + cache)

$$= \frac{3}{\frac{1}{1.43} + \frac{1}{1.11} + \frac{1}{1.25}} = 1.25$$

Machines	S1	S2	S3	S4
Base + FP Unit	1.57	2.55	1.67	?
Base + cache	1.26	1.26	1.25	?

Problem 1.3 (a)

S4 - Geometric Mean.
Geometric Mean (Base + FP Unit)

$$= \sqrt[3]{(1 \times 5 \times 1.67)} = 2.03$$

Geometric Mean (Base + cache)

$$= \sqrt[3]{1.43 \times 1.1 \times 1.25} = 1.26$$

Machines	S1	S2	S3	S4
Base + FP Unit	1.57	2.55	1.67	2.03
Base + cache	1.26	1.26	1.25	1.26

Problem 1.3 (b)

To remove the bias due to the difference in execution times, we first normalize the execution time of the base machine to 1, yielding the normalized execution times in Table.

Machines	Program 1	Program 2	Program 3
Base Machine	1	1	1
Base + FP Unit	1	0.2	0.6
Base + cache	0.7	0.9	0.8

Compute the four average speedups to the base machine for both improvements. Which conclusions would you draw about the best improvement if you were to consider each average speedup individually?

Problem 1.3 (b)

Calculate speed up over base machine and average execution time.

Machines	Program 1	Program 2	Program 3	Average Normalized Execution Time
Base Machine	1/1	1/1	1/1	(1+1+1)/3
Base + FP Unit	1/1	1/0.2	1/0.6	(1+0.2+0.6)/3
Base + cache	1/0.7	1/0.9	1/0.8	(0.7+0.9+0.8)/3

Machines	Program 1	Program 2	Program 3	Average Normalized Execution Time
Base Machine	1	1	1	1
Base + FP Unit	1	5	1.67	0.6
Base + cache	1.43	1.1	1.25	0.8

Speedup is same. Only average execution time is changed. So we only need to calculate the S1.

Problem 1.3 (b)

Lets calculate the S1 - Ratio of average execution time.

$$Ratio(Base + FPUnit) = \frac{1}{0.6} = 1.67$$

$$Ratio(Base + cache) = \frac{1}{0.8} = 1.25$$

Machines	S1	S2	S3	S4
Base + FP Unit	1.67	2.55	1.67	2.03
Base + cache	1.25	1.26	1.25	1.26

In all cases, Base + FP units is still better than Base + cache.

Table of Contents

Problem 1.6

A baseline microprocessor is enhanced as follows.

1. The core is replicated 16 times to form a 16-way CMP.
2. A floating-point co-processor is added to each core. This co-processor speeds up all floating point operations in each core by a factor 4 and is attached to the core. While the floating point co-processor is active, the host core is idle and the presence of the co-processor does not affect instructions that are not part of floating point operations.

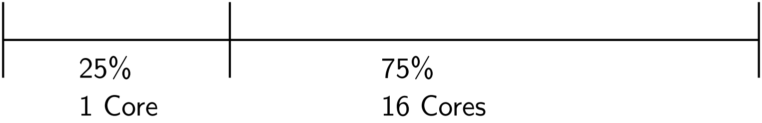
We observe the following on the NEW machine

1. The floating-point co-processor is used during 30% of the execution time of each core.
2. 25 % of the time only one core is active and 75 % of the time the sixteen cores are active.

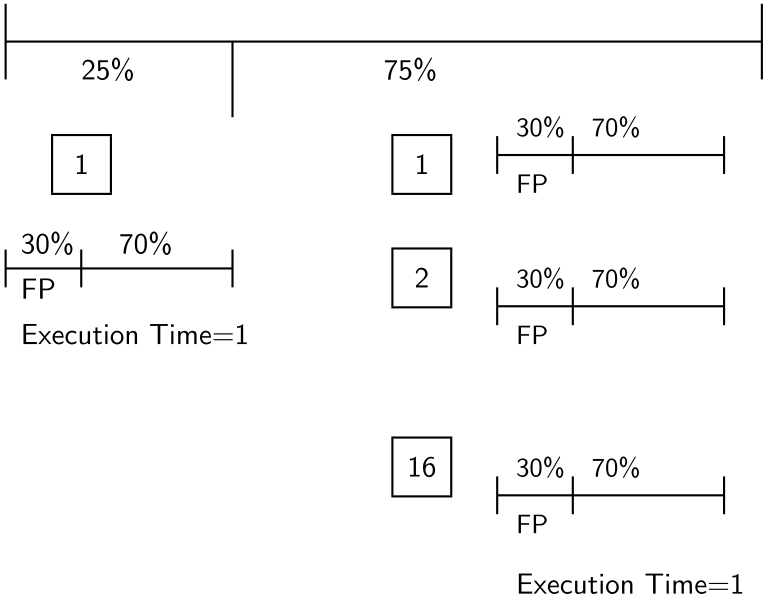
What is the speedup of the new machine over the base machine?

Problem 1.6

The given data is for the new machine.
We have to find the data for the base machine without the improvements.Execution Time New Machine = 1



Problem 1.6


25%
75%
1
30%
70%
FP
Execution Time=1
1
30%
70%
FP
2
30%
70%
FP
16
30%
70%
FP
Execution Time=1

Problem 1.6

Normalized execution time without the floating point unit enhancement assuming that execution time with enhancement is 1.

$$T_{base-nofp} = (0.3 \times FP_{factor}) + 0.7$$
$$T_{base-nofp} = 0.3 \times 4 + 0.7$$
$$T_{base-nofp} = 1.2 + 0.7$$
$$T_{base-nofp} = 1.9$$

Now consider the normalized complete execution time for base machine when only 1 core is available.(ie: no parallel execution)

$$T_{serial} = Serial(\%) \times T_{parallel} + Parallel(\%) \times (\#Cores) \times T_{parallel}$$
$$T_{base} = Serial(\%) \times T_{base-nofp} + Parallel(\%) \times (\#Cores) \times T_{base-nofp}$$
$$T_{base} = 0.25 \times 1.9 + 0.75 \times 16 \times 1.9$$
$$T_{base} = 0.475 + 22.8$$
$$T_{base} = 23.275$$

Problem 1.6

$$\text{Speedup} = \frac{T_{base}}{T_{new}}$$

$$\text{Speedup} = \frac{23.275}{1}$$

$$\text{Speedup} = 23.275$$