Assignment 3 Solutions
Instruction Set Architecture, Performance and Other ISAs (continued)

Due: May 1, 2014

Unless otherwise noted, the following problems are from the Patterson & Hennessy textbook (5th ed.).

1. Performance Processors:
   a) You can define any reasonable metric. One possible metric is the running time of the benchmark program. Running time = Instruction Count * Average CPI * Cycle Time.

	Compiler A	Compiler B
rAlpha	2.47×10^{-6} s	$2.20\times10^{-6}\,s$
c86	$1.26\times10^{-6}\,s$	$1.53\times10^{-6}\,s$

   c86 + Compiler A is the best.

   b) One possible solution is to define single core running time as metric.

	Compiler A	Compiler B
rAlphaX	$9.93\times10^{-7}\,s$	$9.26\times10^{-7}\,s$
c86x4	$7.69\times10^{-7}\,s$	$9.61\times10^{-7}\,s$

   c86x4 + Compiler A is the best.

   c) Using the metric to calculate:

rAlphaX	$4.71\times10^{5}\,s\frac{cycles\times core}{time(s)\times dollars^2}$
c86x4	$3.60\times10^{5}\,s\frac{cycles\times core}{time(s)\times dollars^2}$

   This metric shows that rAlphaX is better. You can give your arguments about whether this metric is useful or not. If you say this metric is useful, justify your answer. Otherwise, you should try to use b) to argue that this metric is not good enough: c86x4 has better performance for the same price.

   d) You should define a reasonable metric using power and performance, similar to energy delay product. One possible solution is:

$$latency\times power^2$$

2. Amdahl's Law:
   (1)

$$T_{old} = 4 + 14 + 2 + 12 + 2 = 34$$

$$Speedup = \frac{T}{0.85T} = \frac{1}{0.85}$$

$$T_{new} = \frac{T_{affected}}{Speedup} + T_{unaffected} = 0.85T_{affected} + T_{unaffected}$$

$$= 0.85(4 + 2 + 2) + (34 - 4 - 2 - 2) = 32.8ms$$

$$Speedup_{total} = \frac{T_{old}}{T_{new}} = \frac{34}{32.8} = 1.037$$

   (2) Use similar equation in (1).

$$T = 0.9 \times 14 + (34 - 14) = 32.6ms$$

$$Speedup_{total} = 1.043$$

   (3) Just practice again:

$$T = 0.9 \times 12 + (34 - 12) = 32.8ms$$

$$Speedup_{total} = 1.037$$

   (4) See this table:

Number of Processors	Computing Time	Computing Time Ratio	Routing Time Ratio
2	176		
4	96	0.545	1.181
8	49	0.510	1.308
16	30	0.612	1.294
32	14	0.467	1.045
64	6.5	0.464	1.13

3. MIPS ISA:
   (1) For R-Type instructions, you need to increase the size of rs, rt and rd bit field to 7 bits. Since there are more instructions, you need to expand opcode and (or) funct fields to encode them. Any reasonable answer is acceptable, but you should justify why your new format is enough for the increased number of instructions.

   (2) For I-Type instructions, you can change the instruction format based on your assumptions because we did not specify what kind of new instructions are included. Some

possible answers are: if there are instructions that use longer immediate constant, then we need to increase the immediate bit field. Notice that RISC architectures usually use fixed length instruction format. So I-type format is better to have the same length as R-type in (1). The basic acceptable answer should at least increase register bit field like what you did in (1).

(3) If you have increased instruction length, then it is possible for the program size to increase. If the new included instructions have more complex functions (like pseudo instructions), the programs using these new instructions may decrease the program size; more registers can also reduce load/store instructions; etc.

This is an open question. We will accept your answer as long as you can justify it.