Rodrigo Becerril Ferreyra
CECS 440 Section 02
Exercise 1
07 September 2021

Problem 1.1

Three different types of computers include the following:

- Personal computers (PCs)
- Server computers
- Embedded computers

Problem 1.3

A computer program written in C goes through the preprocessor which prepares the program for compilation, the compiler which turns the high-level code into assembly, the assembler which translates the compiler's output into machine (binary) code, and finally the linker, where external libraries can be added to the final program.

Problem 1.5

1. P2 has the highest performance in instructions per second, at 2.5×10^9:

$$\frac{2.5 \times 10^9 \text{ cyc}}{1 \text{ s}} \times \frac{1 \text{ inst}}{1 \text{ cyc}} = \frac{2.5 \times 10^9 \text{ inst}}{1 \text{ s}}$$

   P1 and P3 have an instructions per second rating of 2×10^9 and 1.818×10^9, respectively.

2.

In 10 seconds:	Number of Cycles	Number of instructions
P1	30×10^9	20×10^9
P2	25×10^9	25×10^9
P3	40×10^9	18.18×10^9

3. We should have a clockrate that is $1.2 \times 1.3 = 1.56x$ the rate of the original clock; that is, the clock should be 56% higher.

Problem 1.7

There are 1×10^5 class A instructions, 2×10^5 class B instructions, 5×10^5 class C instructions, and 2×10^5 class D instructions.

1. The number of cycles P1 will go through to complete the program is $1(1 \times 10^5) + 2(2 \times 10^5) + 3(5 \times 10^5) + 3(2 \times 10^5) = 2.6 \times 10^6$ cycles, and the number of cycles P2 will go through to complete the program is $2(1 \times 10^5) + 2(2 \times 10^5) + 2(5 \times 10^5) + 2(2 \times 10^5) = 2.0 \times 10^6$ cycles.

2. P1 will take $\frac{2.6 \times 10^6 \text{ cyc}}{2.5 \times 10^9 \text{ cyc/s}} = 1.04\,\text{ms}$ to complete the program, while P2 will take $\frac{2.0 \times 10^6 \text{ cyc}}{3 \times 10^9 \text{ cyc/s}} = 667\,\mu\text{s}$ to complete the same program. P2 is faster.

Problem 1.9

$$P = \frac{1}{2} C V^2 f$$
$$C = \frac{2P}{V^2 f}$$

where P is power, C is capacitive load, V is voltage, and f is frequency.

1. The Pentium 4 Prescott processor has an average capacitive load of

$$\frac{2(100\,\text{W})}{(1.25\,\text{V})^2 3.6\,\text{GHz}} = 35.6\,\text{nF}.$$

   The Core i5 Ivy Bridge processor has an average capacitive load of

$$\frac{2(70\,\text{W})}{(0.9\,\text{V})^2 3.4\,\text{GHz}} = 50.8\,\text{nF}.$$

2. The Pentium 4 Prescott processor idle power dissipation ratio is

$$10\,\text{W}/(10\,\text{W} + 90\,\text{W}) = 10\%.$$

   The Core i5 Ivy Bridge processor idle power dissipation ratio is

$$30\,\text{W}/(30\,\text{W} + 40\,\text{W}) \approx 42.9\%.$$

3. The voltage should also be reduced by 10% to keep the same leakage current.

Problem 1.13

1. The equation for CPU time is

$$t_{\text{CPU}} = \text{instcount} \times \text{CPI} \times \frac{1}{f}.$$

   Processor P1 has a CPU time of $(5 \times 10^9) \times (0.9) \times \frac{1}{4\,\text{GHz}} = 1.125\,\text{s}$. Processor P2 has a CPU time of $(1 \times 10^9) \times (0.75) \times \frac{1}{3\,\text{GHz}} = 0.25\,\text{s}$. Even though P1 has a higher clock rate, P2 beats P1 in total time spent.

2. P1 will take $0.0.225\,\text{s}$ to process 1×10^9 instructions. The number of instructions ι that P2 can process in the same amount of time is:

$$\begin{aligned} t_{\text{CPU}} &= \text{instcount} \times \text{CPI} \times \frac{1}{f} \\ 0.225\,\text{s} &= \iota \times \frac{0.75}{3\,\text{GHz}} \\ \iota &= 9 \times 10^8 \end{aligned}$$

   P2 can process 9×10^8 instructions in the same time that it takes P1 to process 10×10^8, which is less.

3. P1 has a MIPS rating of $\frac{4\,\text{GHz}}{0.9 \times 10^6} = 4444\,\text{s}^{-1}$, while P2 has a MIPS rating of $\frac{3\,\text{GHz}}{0.75 \times 10^6} = 4000\,\text{s}^{-1}$. P1 has a higher MIPS rating.

4. P1's MFLOPS rating is $\frac{0.4(5 \times 10^9)}{1.125 \times 10^6\,\text{s}} = 1778\,\text{s}^{-1}$, while P2 MFLOPS rating is $\frac{0.4(1 \times 10^9)}{0.25 \times 10^6\,\text{s}} = 1600\,\text{s}^{-1}$. P1 has a higher MFLOPS rating.

Problem 1.15

Amdahl's Law states that

$$t = o/n + u$$

where t is the execution time after improvement, o is the amount of time of the thing you're improving (before improvement), n is the amount of time you're improving it by, and u is the time of the unaffected component of the total time. The amount of time the processor takes to run all instructions at current specs as given in the problem is given in the table below.

Type of Instruction	Time (s)
FP	0.025
INT	0.055
L/S	0.16
branch	0.016
Total	0.256

1. Half the time it takes for the program to run by improving FP operations:

$$0.128\,\mathrm{s} = \frac{0.025\,\mathrm{s}}{n} + 0.231\,\mathrm{s}$$
$$-0.103\,\mathrm{s} = \frac{0.025\,\mathrm{s}}{n}$$
$$n = -0.243$$

   It is impossible to half the runtime because even if the FP operations were to disappear, the program would still run for 0.231 s.

2. Half the time it takes for the program to run by improving L/S operations:

$$0.128\,\mathrm{s} = \frac{0.16\,\mathrm{s}}{n} + 0.096\,\mathrm{s}$$
$$0.032\,\mathrm{s} = \frac{0.16\,\mathrm{s}}{n}$$
$$n = 5$$

   The CPI of the L/S instruction must be improved by 5x, meaning the CPI should be at 0.2 CPI.

3. The old, unimproved run time of the program is 0.256 s, and the new improved time obtained by improving FP and INT by 40%, and L/S and branch by 30%, is

$$(0.025 \cdot 0.6) + (0.055 \cdot 0.6) + (0.16 \cdot 0.7) + (0.016 \cdot 0.7) = 0.1712\,\mathrm{s}$$

   This means that there is an improvement in time of $\frac{0.256}{0.1712} \approx 1.49$.