

COMPUTER ARCHITECTURE & ASSEMBLY LANGUAGE HW 1 - Adil Hydari

February 8, 2024

Problem 1. Consider two different implementation of the same instruction set architecture. The instruction can be divided into four classes according to their CPI (class A, B, C and D). P1 with clock rate of 3 GHz and CPIs of 1, 4, 3 and 2, and P2 with a clock 2.5 GHz and CPIs of 2, 3, 2 and 1. Given a program with a dynamic instruction count of $1.0E6$ instructions divided into classes as follows: 20% class A, 35% class B, 20% class C and 25% class D, which implementation is faster?

- a) What is the global CPI for each implementation?
- b) Find the clock cycle required in both cases.

Solution:

a)

$$\text{Global CPI} = (1 \times 0.20) + (4 \times 0.35) + (3 \times 0.20) + (2 \times 0.25)$$

$$\text{Global CPI} = 0.2 + 1.4 + 0.6 + 0.5 = 2.7$$

$$\text{Clock Cycles} = 1^6 \times 2.7 = 2.7^6$$

b)

$$\text{Global CPI} = (2 \times 0.20) + (3 \times 0.35) + (2 \times 0.20) + (1 \times 0.25)$$

$$\text{Global CPI} = 0.4 + 1.05 + 0.4 + 0.25 = 2.1$$

$$\text{Clock Cycles} = 1^6 \times 2.1 = 2.1^6$$

c)

The second implementation is the faster one as it requires fewer clock cycles to execute the same number of instructions, even though it has a lower clock speed.

Problem 2. Assume a program requires the execution of 90×10^6 FP (floating point) instructions, 120×10^6 INT (integer) instructions, 80×10^6 Load/Store (L/S) instructions and 20×10^6 branch instructions. The CPI for each type of instruction is 2, 2, 3 and 2, respectively. Assume that the processor has a 3 GHz clock rate.

a) By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

b) By how much must we improve the CPI of L/S instruction if we want the program to run two times faster?

c) By how much is the execution time of the program improved if the CPI of INT and FP instructions are reduced by 30% and the CPI of L/S and Branch is reduced by 20%?

Solution:

Cycles)

$$90 * 10^6 * 2 + 120 * 10^6 * 2 + 80 * 10^6 * 3 + 20 * 10^6 * 2 = 7 * 10^8$$

a)

Without given a direct target for the exact amount of clock cycles we want to achieve, we can assume that if we multiply the amount of clock cycles that it takes to execute the FP instructions by some value $CPI_{FP_{new}}$ instead of 2 and set the whole expression equal to $3.5 * 10^8$ we will get our answer.

$$90 * 10^6 * CPI_{FP_{new}} + 120 * 10^6 * 2 + 80 * 10^6 * 3 + 20 * 10^6 * 2 = 3.5 * 10^8$$
$$x = -1.8$$

Based on this calculation, we would need -1.8 clock cycles for the execution time to be halved. As such, it is impossible for the CPI value of the FP instructions to be changed to any value that would allow the execution time to be halved.

b)

In order to improve the execution speed program's Load/Store instructions for it to run 2x faster, we would need to do the same

type of calculation we did for A.

$$90 * 10^6 * 2 + 120 * 10^6 * 2 + 80 * 10^6 * \text{CPI}_{\text{LSnew}} + 20 * 10^6 * 2 = 3.5 * 10^8$$

$$x = -1.375$$

Based on this calculation, we can see we would need negative clock cycles in order for the execution time to be halved, as such there is no CPI value for L/S instructions that would halve execution time.

c)

Again using the same equation for A and B:

$$\text{CPI}_{\text{FP}} = \text{CPI}_{\text{INT}} = 2 * 0.7 = 1.4$$

$$\text{CPI}_{\text{LS}} = 3 * 0.8 = 2.4$$

$$\text{CPI}_{\text{BRANCH}} = 1.6$$

$$90 * 10^6 * 1.4 + 120 * 10^6 * 1.4 + 80 * 10^6 * 2.4 + 20 * 10^6 * 1.6 = 5.18 * 10^8$$

$$\text{Execution time: } \frac{5.18 * 10^8}{3 * 10^9} = \frac{518}{3000} \text{ seconds}$$

$$\text{Improvement factor: } 700/518 = 1.35$$

According to these calculations, we can see that we had an overall improvement in execution time of 35%

Problem 3. One fallacy students and researchers often make is expecting to improve the overall performance of a computer just by improving only one aspect of the computer. Instructions are classified as: Integer, Floating Point, Load/Store, Branch, in this fictitious processor. Consider a computer running a program that requires 250s, with 70s spent executing Floating Point (FP) instructions, 80s executed Load/Store instructions, and 50s spent executing branch instructions.

- By how much is the total time reduced if the time for the FP operations is reduced by 15%?
- By how much is the time for integer (INT) operations reduced if the total time is reduced by 15%?
- Can the total time be reduced by 15% by reducing only the time for the branch instructions?

Solution:

a)

The total time is reduced when the FP operation time is changed can be found using:

$$\text{Reduction in FP time} = 70\text{s} \times 15\% = 70 \text{ sec} \times 0.15 = 10.5 \text{ sec}$$

$$\text{New Time} = 250 \text{ sec} - 10.5 \text{ sec} = 239.5 \text{ sec}$$

b)

$$\text{Reduction in INT time} = 250 \text{ sec} \times 15\% = 250 \text{ sec} \times 0.15 = 37.5 \text{ sec}$$

c)

$$250 \text{ sec} \times 15\% = 37.5 \text{ sec}$$

$$\frac{37.5}{50} \times 100\% = 75\%$$

Based on these calculations, reducing the total time by 15% with only branch instructions is not possible without making improvements to the overall system performance.

Problem 4. Let register x5 hold the hex number 01a74cd0 hex and let x6 hold the hex number 00467b44 hex. If this information is enough, then answer the following three questions, else explain what extra information you may need to proceed.

- (1 pts) What are the contents of register x31 after the following instruction is executed? `addi x31, x5, 32`
- (1 pts) What would be the contents of register x28 after the following instruction is executed? `add x28, x5, x6`
- (1 pts) What would be the contents of register x29 after the following instruction is executed? `ld x29, 1056(decimal)(x30)`
- (1 pts) What would be the contents of register x29 after the following instruction is executed? `sd x28, 16 decimal(x29)`

Solution:

- $0x01a74cd0 + 0x20 = 0x01a74cf0$
- $0x01a74cd0 + 0x00467b44 = 0x01edb814$
- In order to perform this operation, we need to know the value that is contained within register x30, instead the only information we receive is that we will load the value into register x29 and we have an offset of 1056 (in decimal).
- The store double instruction stores a value from a register into memory. This operation will store the contents of x28 into the memory address of x29 with an offset of 16 (in decimal). However, `sd` does not alter the contents of x29 since we are essentially doing `x29+16` and only changing the memory address.

Problem 5. Find the top 3 super-computing centers in the world (High

Performance Computing Centers, focusing on high computational capacity and MIPS) and the top 3 data-centers (focusing on storage capacity, Cloud Computing, etc.) prevailing today. For each unit/center you document, search and find the following:

- 1) Location – and insight as to why this location was selected.
- 2) How many servers or cores are included
- 3) What are the highest number of FLOPS (Floating Point Instructions per second)
- 4) What is the highest MIPS or Instructions per second processed, or speed or performance
- 5) What is the total power used
- 6) What is the total power density per square foot
- 7) What is the vendor and/or the processor's technology used (i.e., POWER9 CPU and NVIDIA Tesla V100 Tensor Core GPUs etc.)

Solution:

1) Frontier - Oak Ridge National Laboratory (ORNL) Location: Tennessee, USA

Servers/Cores: 8,699,904 cores.

Highest FLOPS: Its peak performance is 1,679.82 petaflops.

Highest MIPS/Performance: Not Listed.

Total Power Used: 22.7 MW.

Power Density: 62.86 gigaflops/watt.

Vendor/Processor's Technology: 3rd Generation EPYC and AMD Instinct MI250X.

2) Aurora - Argonne National Laboratory Location: Illinois, USA

Servers/Cores: 4,742,808 cores.

Highest FLOPS: Its peak performance is 1,059.33 petaflops.

Highest MIPS/Performance: Not Listed.

Total Power Used: 24.6 MW.

Power Density: 30 teraFLOPS per node.

Vendor/Processor's Technology: Intel Max 1550 and Intel Xeon Max 9470 .

3) Eagle - Microsoft Location: Illinois, USA

Servers/Cores: 1,123,200 cores.

Highest FLOPS: Its peak performance is 846.84 petaflops.

Highest MIPS/Performance: Not Listed.

Total Power Used: Not Listed.

Power Density:N/A.

Vendor/Processor's Technology: Intel Xeon Platinum 8480C and Nvidia Hopper H100.