# hw2

Jonathan Kloepfer

September 2021

## 1 1.5 Consider three different processors P1,P2 and P3 executing the same instruction set. P1 has a 3GHz clock rate and a CPI of 1.5. P2 has a 2.5GHZ clock rate and a CPI of 1.0. P3 has 4.0GHz clock rate and has a CPI of 2.2.

(a) Which processor has the highest performance expressed in instructions per second?

IPS=$\frac{ClockRate}{CPI}$
P1: 3GHz / 1.5 = $2 * 10^9$ instructions per second
P2: 2.5GHz / 1.0 = $2.5 * 10^9$ instructions per second
P3: 4GHz / 2.2 = $1.82 * 10^9$ instructions per second

Answer P2 has the best performance according to instructions per second because its rating is higher.

(b) If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions?
Cycles:
CPU Time=$\frac{CPUClockCycles}{ClockRate}$
CPU Clock Cycles=Clock Rate * Time
P1: 3GHz * 10 = $3 * 10^{10}$ cycles
P2: 2.5GHz * 10 = $2.5 * 10^{10}$ cycles
P3: 4GHz * 10 = $4 * 10^{10}$ cycles
Number of instructions:
Instruction Count= $\frac{Cycles}{CPI}$ P1: 3GHz * 10 / 1.5 = $2 * 10^{10}$ instructions
P2: 2.5GHz * 10 / 1.0 = $2.5 * 10^{10}$ instructions
P3: 4GHz * 10 / 2.2 = $1.82 * 10^{10}$ instructions

(c) We are trying to reduce the execution time by 30% but this leads to an increase of 20% in the CPI. What clock rate should we have to get this

time reduction?
P1: Execution time * 0.7 = $\frac{(Numofinstructions*CPI*1.2)}{(NewClockrate)}$
New Clock Rate=$\frac{CPUClockCycles*1.2}{CPUTime*0.7}$
New Clock Rate=1.71* Clock Rate
P1: 3GHz * 1.71 = 5.13 GHz
P2: 2.5GHz * 1.71 = 4.27 GHz
P3: 4GHz * 1.71 = 6.84 GHz

## 2 1.7 Consider two different implementations 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 2.5GHz and CPIs 1, 2, 3, and 3, and P2 with a clock rate of 3GHz and CPIs of 2, 2, 2 and 2. Given a program with a dynamic instruction count of 1.0*$10^6$ divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D which is faster P1 or P2?

Time=instructions*CPI/clock rate
Add each instruction multiplied by its percentage multiplied by its CPI and divide by clock rate
P1:0.00104s
P2:.000666s
P2 is faster because it has a lower time.

(a) What is the global CPI for each implementation? CPI=CPU Time * Clock Rate/Instruction Count
P1: 2.6
P2: 1.998

(b) Find the clock cycles required in both cases?
Clock cycles=CPI*Instruction Count
P1: $2.6 * 10^6$
P2: $1.998 * 10^6$

## 3 1.8 Compilers can have a profound impact on the performance of an application. Assume that for a program, compiler A results in a dynamic instruction count of 1.0E9 and has an execution time of 1.1 s, while compiler B results in a dynamic instruction count of 1.2E9 and an execution time of 1.5s

(a) Find the average CPI for each program given that the processor has a clock cycle time of 1 ns? CPU time= Instruction count * CPI * Clock Cycle Time CPI=CPU Time/Instruction count *Clock Cycle Time A:1.1 B:1.25

(b) Assume the compiled programs run on two different processors. If the execution times on the two processors are the same, how much faster is the clock of the processor running compiler A's code versus the clock of the processor running compiler B's code? Clock rate (Frequency) = Instruction count $\times$ CPI / CPU time Ratio=(Instruction count(B) $\times$ CPI(B)) / (Instruction count (A) $\times$ CPI(A)) Processor A has a clock speed 1.37 time higher.

(c) A new compiler is developed that uses only 6.0E8 instructions and has an average CPI of 1.1. What is the speedup of using this new compiler versus using compiler A or B on the original processor? Atime/NewTime=Instruction Count* CPI(A or b)/(Instruction Count * CPI) A: 1.67 times faster B: 2.27 times faster

## 4 1.10 Assume for arithmetic, load/store, and branch instructions, a processor has CPIs of 1, 12, and 5, respectively. Also assume that on a single processor a program requires the execution of 2.56E9 arithmetic instructions, 1.28E9 load/store instructions, and 256 million branch instructions. Assume that each processor has a 2 GHz clock frequency. Assume that, as the program is parallelized to run over multiple cores, the number of arithmetic and load/store instructions per processor is divided by 0.7 x p (where p is the number of processors) but the number of branch instructions per processor remains the same.

(a) Find the total execution time for this program on 1, 2, 4, and 8 processors, and show the relative speedup of the 2, 4, and 8 processor result relative to the single processor result.
Execution time= Clock Cycles/Clock Rate 1: 9.6s
$((2.56 * 10^9) + (12 * 1.28 * 10^9) + (5 * 2.56 * 10^8))/(2 * 10^9)$
2: 7.04s
you gain 2.56 seconds or 2 processors is 1.36 times faster.
$((2.56 * 10^9)/1.4 + (12 * 1.28 * 10^9)/1.4 + (5 * 2.56 * 10^8))/(2 * 10^9)$
4: 3.84s
You gain 5.76 seconds or 4 processors is 2.5 times faster than 1.
$((2.56 * 10^9)/2.8 + (12 * 1.28 * 10^9)/2.8 + (5 * 2.56 * 10^8))/(2 * 10^9)$
8: 2.24s
You gain 7.36 seconds or 8 processors is 4.28 times faster than 1.
$((2.56 * 10^9)/5.6 + (12 * 1.28 * 10^9)/5.6 + (5 * 2.56 * 10^8))/(2 * 10^9)$

(b) If the CPI of the arithmetic instructions was doubled, what would the impact be on the execution time of the program on 1, 2, 4, or 8 processors? It makes execution time go up. And makes multi core processing more effective because it allows more higher CPI instructions to be divide among more processors. 1: 10.88s
$((2.56 * 10^9 * 2) + (12 * 1.28 * 10^9) + (5 * 2.56 * 10^8))/(2 * 10^9)$
2: 7.95s
you gain 2.93 seconds or 2 processors is 1.36 times faster.
$((2.56 * 10^9 * 2)/1.4 + (12 * 1.28 * 10^9)/1.4 + (5 * 2.56 * 10^8))/(2 * 10^9)$

4: 4.29s
You gain 6.58 seconds or 4 processors is 2.5 times faster than 1.
$((2.56 * 10^9 * 2)/2.8 + (12 * 1.28 * 10^9)/2.8 + (5 * 2.56 * 10^8))/(2 * 10^9)$
8: 2.46s
You gain 8.411 seconds or 8 processors is 4.407 times faster than 1.
$((2.56 * 10^9 * 2)/5.6 + (12 * 1.28 * 10^9)/5.6 + (5 * 2.56 * 10^8))/(2 * 10^9)$

(c) To what should the CPI of load/store instructions be reduced in order for a single processor to match the performance of four processors using the original CPI values?
$((2.56 * 10^9) + (CPI * 1.28 * 10^9) + (5 * 2.56 * 10^8))/(2 * 10^9)$=$((2.56 * 10^9)/2.8 + (12 * 1.28 * 10^9)/2.8 + (5 * 2.56 * 10^8))/(2 * 10^9)$
Load and stores CPI value would have to become 3.

## 5 1.13 Section 1.11 cites as a pitfall the utilization of a subset of the performance equation as a performance metric. To illustrate this, consider the following two processors. P1 has a clock rate of 4 GHz, average CPI of 0.9, and requires the execution of 5.0E9 instructions. P2 has a clock rate of 3 GHz, an average CPI of 0.75, and requires the execution of 1.0E9 instructions.

(a) One usual fallacy is to consider the computer with the largest clock rate as having the largest performance. Check if this is true for P1 and P2.
CPU time=instruction count* cpi / Clock Rate
P1:1.125s
P2:.25s
The statement is not true for P1 and P2 because P2 has a faster execution time so P1 cant be objectively better.

(b) Another fallacy is to consider that the processor executing the largest number of instructions will need a larger CPU time. Considering that processor P1 is executing a sequence of 1.0E9 instructions and that the CPI of processors P1 and P2 do not change, determine the number of instructions that P2 can execute in the same time that P1 needs to execute 1.0E9 instructions.
instruction count= CPU time * clock rate/cpi
P1 time:.225s
P2 instructions:$9 * 10^8$ instructions

## 6 1.15 Assume a program requires the execution of 50 * $10^6$ FP instructions, 110 * $10^6$ INT instructions, 80 * $10^6$ L/S instructions, and 16 * $10^6$ branch instructions. The CPI for each type of instruction is 1, 1, 4, and 2, respectively. Assume that the processor has a 2 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?
Original time=$((50*10^6)+(110*10^6)+(80*10^6*4)+(2*16*10^6))/(2*10^9)$
Original time=.256s
There is no change to the CPI that can be made to make the code run 2 times as fast because FP instructions are not a substantial enough part of the original time.

(b) By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?
We must divide the current CPI of L/S instructions by 5
Solved using Amdhal's Law

(c) By how much is the execution time of the program improved if the CPI of INT and FP instructions is reduced by 40% and the CPI of L/S and Branch is reduced by 30%? you gain 0.048s.