CS451 Single Cycle Performance Homework

Troy Veldhuizen

1.) Consider a CPU. Suppose we were given the option of decreasing the cycle time by 20% in exchange for a 20% increase in the number of instructions. Is this a good trade-off?

$$\begin{aligned} t = n * p \quad & t' = n' * p' \\ t = 925n \quad & t' = (1.2)n * (0.8)925 \\ & t' = (1.2)n * 740 \\ & t' = 888n \end{aligned}$$

We can set the two equations equal to each other to see how many instructions each CPU can run in the same amount of time.

$$\begin{aligned} 925n &= 888n \\ 1.042n &= n \end{aligned}$$

We can see that the new cpu will run 4.2% more instructions in the same time as the old CPU, assuming the old CPU's clock period was 925ps. This is a good performance increase. This is a good tradeoff.

3.) Consider the single-cycle CPU with a 925ps clock period. Suppose that doubling the size of the register file from 32 to 64 registers would reduce the number of lw and sw instructions, but would increase the delay of the register file from 150 to 175. What percent of these instructions must be removed? Assume that loads and stores account for 35% of the original workload.

$$\begin{aligned} t = n * p \quad & t' = n' * p' \\ t = 925n \quad & t' = (n - (x * 0.35 * n))950 \\ & t' = (n - 0.35xn)950 \\ & t' = n(1 - 0.35x)950 \end{aligned}$$

Solve by setting t equal to t'.

$$\begin{aligned} 925n &= n(1-0.35x)950 \\ 925 &= (1-0.35x)950 \\ \frac{925}{950} &= 1-0.35x \\ \frac{925}{950}-1 &= -0.35x \\ \frac{-.0263}{-0.35} &= x \\ x &= 0.075 \end{aligned}$$

Greater than 7.5% of the lw and sw instructions need to be removed to increase the CPU's performance.

5.) Problem 2.43 from the Patterson and Hennessy textbook (3rd edition): As you know, MIPS is a "load-store" architecture. In contrast, the Intel architecture allows ALU operations to access memory. In this exercise, you will examine quantitatively the pros and cons of adding an "IA-32 style" addressing mode to MIPS. Consider adding a new instruction:

```
addm $t2, 100($t3)    #  $t2 = $t2 + Memory[$t3 + 100].
```

Assume that the new instruction will cause the cycle time to increase by 10%. Use the instruction frequencies given below:

- Arithmetic / Logic: 42%
- Loads: 24%
- Stores: 12%
- Branch and Jump: 22%.

(these values are for for the SPEC2000int and are given in Figure 2.48). Assume that the new instruction affects only the clock speed, not the CPI. What percentage of loads must be eliminated for a single-cycle machine with the new instruction to have at least the same performance as the single-cycle machine?

$$\begin{aligned} t = n*p \quad & t' = n'*p' \\ t = 925n \quad & t' = n'*1017.5 \\ & t' = n-(x*0.24*n)1017.5 \\ & t' = n-(0.24xn)1017.5 \\ & t' = n(1-0.24x)1017.5 \end{aligned}$$

Solve by setting t equal to t'.

$$\begin{aligned} 925n &= n(1-0.24x)1017.5 \\ 925 &= (1-0.24x)1017.5 \\ \frac{925}{1017.5} &= 1-0.24x \\ \frac{925}{1017.5}-1 &= -0.24x \\ \frac{-.091}{-0.24} &= x \\ x &= 0.380 \end{aligned}$$

The new new instruction must remove 38% of loads to have at least the same performance.

7.) Your company builds a single-cycle CPU that is very similar to that presented in the textbook, except its lw and sw instructions do not accept an offset parameter. This difference allows the ALU and the data memory to be organized in parallel. The lead architect is considering switching to a "Patterson and Hennessy" style CPU where the ALU and data memory are organized in series because (1) he estimates that the switch will reduce the number of instructions by 10% and (2) he has heard there has been a recent breakthrough that will significantly increase the speed of the ALU. Given that

- the current CPU's clock period is 1000ps (i.e., the one with the ALU and data memory in parallel),
- the ALU requires 200ps, and
- the data memory requires 250ps.

(a) Without any changes to the ALU, will the new design be faster or slower than the current design?

$$\begin{aligned} t &= n*p & t' &= n'*p' \\ t &= 1000n & t' &= n'*1200 \\ & & t' &= (0.9)n*1200 \\ & & t' &= 1080n \end{aligned}$$

We can set the two equations equal to each other to see how many instructions each CPU can run in the same amount of time.

$$\begin{aligned} 1000n &= 1080n \\ n &= 1.08n \end{aligned}$$

We can see that the old CPU design was 8% faster than the new old.

(b) How fast must the new ALU run in order for the new design to be faster than the current design?

The new CPU needs to run more than 8% faster. So the total clock period needs to be 1,104ps for the new design. The difference in clock period between the old and new design is 104ps. Because the ALU is the only difference on the critical path between the two designs, the new ALU needs to run at 104ps.

8.) Consider the following proposed instruction:incmp(increment and compare). This instruction increments the value of a register before comparing it. In other words, it performs an operation similar to this: if (++R1 != R2). It would be most useful in combining the condition check and increment in a typical for loop. (i.e., combining addi, R1, 1 and bne R1, R2, TOP OF LOOP into a single incmp instruction). Adding this instruction will remove some addi instructions at the cost of adding 30ps to the ALU time. Given that

- The current CPU's clock period is 925ps.
- The current CPU's ALU requires 200ps.
- Adding incmp to the ALU will increase its latency by 30ps.
- Branch statements comprise 11% of a typical workload.

(a) What percent of branch instructions must be consolidated with an addi into a single incmp instruction in order for the new CPU to have better performance than the current CPU?

$$
\begin{aligned}
t = n * p \quad & t' = n' * p' \\
t = 925n \quad & t' = (n - (x * 0.11 * n))955 \\
& t' = (n - 0.11xn)955 \\
& t' = n(1 - 0.11x)955
\end{aligned}
$$

Set t equal to t'.

$$
\begin{aligned}
925n &= n(1 - 0.11x)955 \\
925 &= (1 - 0.11x)955 \\
\frac{925}{955} &= 1 - 0.11x \\
\frac{925}{955} - 1 &= -0.11x \\
\frac{-0.0314}{-0.11} &= x \\
x &= 0.286
\end{aligned}
$$

We need to remove greater than 28.6% of the branch instructions to increase performance in our CPU.

(b) Suppose the CPU had a clock period of 725 instead of 925. Would you need to consolidate more or fewer branch instructions to "break even"? Explain how you can answer this question without doing any additional calculations.

The slowdown of 30ps on the ALU will have a greater effect on the faster clock period of 725ps. The percent loss of speed will be greater. A greater

percentage of branch instructions will need to be consolidated to recover this lost speed.