Advanced Computer Architectures
Exercises

Christian Rossi

Academic Year 2023-2024

Abstract

The course topics are:

- Review of basic computer architecture: the RISC approach and pipelining, the memory hierarchy.
- Basic performance evaluation metrics of computer architectures.
- Techniques for performance optimization: processor and memory.
- Instruction level parallelism: static and dynamic scheduling; superscalar architectures: principles and problems; VLIW (Very Long Instruction Word) architectures, examples of architecture families.
- Thread-level parallelism.
- Multiprocessors and multicore systems: taxonomy, topologies, communication management, memory management, cache coherency protocols, example of architectures.
- Stream processors and vector processors; Graphic Processors, GP-GPUs, heterogeneous architectures.

Contents

CHAPTER 1

Exercise session I

1.1 Exercise one

Assessing the impact of modifications on performance:

1. Substituting a hardware component with a faster alternative.
2. Incorporating multiple parallel systems for executing independent tasks.

1.1.1 Solution

1. By scaling up: response time decreases, while throughput increases.
2. Through scaling out: throughput experiences an increase. Response time will only escalate if a queue was present, awaiting computing resources.

1.2 Exercise two

Let's consider two CPUs: CPU1 and CPU2. CPU1 operates with a clock cycle of $2\,ns$, while CPU2 has an operating frequency of $700\,MHz$. Given the frequencies of occurrence of instructions for both CPUs:

Operation type	**Frequency**	**CPU1 cycle**	**CPU2 cycle**
A	0.3	2	2
B	0.1	3	3
C	0.2	4	3
D	0.3	2	2
E	0.1	4	3

1. Calculate the average CPI for CPU1 and CPU2.
2. Determine which CPU is the fastest.

1.2.1 Solution

1. The CPI (Cycle Per Instruction) is calculated as:

$$\text{CPI} = \frac{\text{clock cycles}}{\text{instruction}}$$

The average CPI is then obtained by:

$$\sum_{i=1}^{n} \text{CPI}_i \cdot \text{F}_i$$

Where, $\text{F}_i = \frac{\text{I}_i}{\text{instruction count}}$. For CPU1:

$$\text{CPI}_1 = 0.3 \cdot 2 + 0.1 \cdot 3 + 0.2 \cdot 4 + 0.3 \cdot 2 + 0.1 \cdot 4 = 2.7$$

For CPU2:

$$\text{CPI}_2 = 0.3 \cdot 2 + 0.1 \cdot 3 + 0.2 \cdot 3 + 0.3 \cdot 2 + 0.1 \cdot 3 = 2.4$$

2. We have that:

$$\begin{aligned}
\frac{\text{EXE}_{\text{CPU1}}}{\text{EXE}_{\text{CPU2}}} &= \left(\frac{\text{IC}_1 \cdot \text{CPI}_1}{F_1}\right)\left(\frac{F_2}{\text{IC}_2 \cdot \text{CPI}_2}\right) \\
&= \frac{\text{IC}_1 \cdot \text{CPI}_1 \cdot F_2}{F_1 \cdot \text{IC}_2 \cdot \text{CPI}_2} \\
&= \frac{\text{CPI}_1 \cdot F_2}{F_1 \cdot \text{CPI}_2} \\
&= \frac{2.7 \cdot 700MHz}{2.4 \cdot 500MHz} \\
&= 1.575
\end{aligned}$$

Hence, CPU2 is approximately 1.575 times faster than CPU1.

1.3 Exercise three

The speed of image processing on FPGA is 2.86 times faster than on a CPU. The power consumption of an FPGA is $100\,W$, while that of the CPU is $30.85\,W$. We want to achieve a speedup of 2 with the addition of an FPGA.

1.3.1 Solution

To achieve a speedup of 2 with the addition of an FPGA, we use Amdahl's law:

$$S_{overall} = \frac{1}{(1 - F_{enhanced}) + \frac{F_{enhanced}}{S_{enhanced}}}$$

Given $S_{overall} = 2$, we solve for F_{enhanced}:

$$2 = \frac{1}{(1 - F_{enhanced}) + \frac{F_{enhanced}}{2.86}} \rightarrow F_{enhanced} = 0.768$$

Therefore, with 76.8% of the processing offloaded to the FPGA, a speedup of 2 can be achieved.

1.4 Exercise four

Consider the following assembly program:

```
LW      $1, OFF($2)
ADDI    $3, $1, 4
SUB     $4, $1, $2
ADDI    $2, $1, -8
SW      $4, OFF($2)
```

No optimizations are applied in the MIPS pipeline. The processor operates with a clock cycle of 2 ns.

- Draw the pipeline schema and highlight potential hazards.
- Illustrate the actual execution with stall cycles inserted.
- Calculate Instruction Count (IC), CPI, and MIPS.

1.4.1 Solution

1. The pipeline schema is:

Instruction	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
LW $1, OFF($2)	F	D	E	M	W				
FDDI $3, $1, 4		F	D	E	M	W			
SUB $4, $1, $3			F	D	E	M	W		
ADDI $2, $1, -8				F	D	E	M	W	
SW $5, OFF($2)					F	D	E	M	W

   The potential hazards are:

   - Instruction 1 writes to register $1, and instructions 2, 3, and 4 read from it.
   - Instruction 2 writes to register $3, and instruction 3 reads from it.
   - Instruction 4 writes to register $2, and instruction 5 reads from it.

2. The real execution with stall cycles inserted is:

Instruction	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
LW $1, OFF($2)	F	D	E	M	W										
ADDI $3, $1, 4		F	S	S	D	E	M	W							
SUB $4, $1, $3					F	S	S	D	E	M	W				
ADDI $2, $1, -8								F	D	E	M	W			
SW $5, OFF($2)									F	S	S	D	E	M	W

3. The performance metrics are:

$$\text{IC} = 5$$

$$\text{CPI} = \frac{CCs}{IC} = \frac{15}{5} = 3$$

$$\text{MIPS} = \frac{\text{clock frequency}}{CPI \cdot 10^6} = \frac{0.5 \cdot 10^9}{3 \cdot 10^6} = 166$$