Comp 2825 - Computer Architecture

Exercise Solutions – Week 1

Exercise 1 – What is a Computer?

C – Something that follows instructions

Computers also calculate, respond to user input, and store data – but what makes them different from calculators, day planners and other electronic devices is that they can be instructed to do different jobs. This makes them general-purpose and highly flexible.

Exercise 2 – What is an Operating System?

C – Microsoft Windows

Operating systems combine with hardware to provide a platform for running application programs. For the most part, an operating system doesn't actually do the job you need done – it provides the facilities that other programs need to do that job.

Exercise 3 – Implementation or Specification

Left-hand flowchart: A – Implementation

The flowchart shows an algorithm – a description of HOW to perform some job. This particular flowchart appears to be copying data from the keyboard to a printer, so the specification might be something along the lines of "typewriter simulator".

Right-hand photo: B – Specification

The photo shows a piece of electronic equipment - we can see which inputs (buttons) and outputs (the display) are required. The innards of this device (the "implementation") could use integrated circuits, transistors, or vacuum tubes and still look pretty much the same on the outside.

Exercise 4 – Pipeline

Latency: C – 2.667 ns

A computer with a 1.5GHz (1,500,000,000 cycles/second) clock takes 1 / 1,500,000,000 seconds for each clock cycle – that's 0.000 000 000 667 seconds, or 0.667ns per clock cycle. Each pipeline stage takes one clock cycle, which is 0.667ns. Since there are 4 pipeline stages that each instruction needs to pass through, the latency is 0.667ns x 4 stages = 2.667ns

Bandwidth: B – 1,500,000,000 instructions/sec (1500MIPS)

One new instruction can enter the pipeline every clock cycle. Since there are 1,500,000,000 clock cycles per second, this means that 1,500,000,000 instructions can be executed every second.

Exercise 5 – Superscalar Pipeline

Latency: D – 12 ns

A computer with a 500MHz (500,000,000 cycles/second) clock takes 1 / 500,000,000 seconds for each clock cycle – that's 0.000 000 002 seconds, or 2ns per clock cycle. Each pipeline stage takes one clock cycle, or 2 ns. No matter which path an instruction takes through the pipeline, it needs to pass through 6 stages, so the latency is 2ns x 6 stages = 12ns

Bandwidth: B – 1000 MIPS (1,000,000,000 instructions/sec)

Since there are two pipelines that instructions can pass through, the first two stages must be able to handle 2 instructions with each clock cycle (otherwise there wouldn't be any point in having the dual pipelines for stages S3-S6). If 2 instructions can enter the pipeline on every cycle, and if there are 500,000,000 cycles per second, then the bandwidth must be 2 x 500,000,000 or 1,000,000,000 instructions/second (1000 MIPS).

Exercise 6 – Superscalar Pipeline

Latency: C – 14 ns

A computer with a 500MHz (500,000,000 cycles/second) clock takes 1 / 500,000,000 seconds for each clock cycle – that's 0.000 000 002 seconds, or 2ns per clock cycle. Each instruction has to pass through the S1-S2-S3 stages, ONE of the S4 stages, and then the S5 stage. This means that the instruction takes 1 + 1 + 1 + 3 + 1 = 7 clock cycles to go through the pipeline. Since each clock cycle takes 2ns, the latency is 7 x 2ns = 14ns.

Bandwidth: A – 500 MIPS (500,000,000 instructions/sec)

Assuming that there's no contention for consecutive instructions to use the same S4 stage, a new instruction can enter the pipeline on every clock cycle. Since there are 500,000,000 cycles per second, this means that the pipeline can execute 500,000,000 instructions per second (500 MIPS).

Note that this would be the BEST performance. Depending on the mix of instructions, the S4 stage can become a bottleneck. For example, if there were 3 floating point instructions in a row, the S4 stage would not be able to process them quickly enough and the pipeline would have to be stalled while the calculations were done.