CSCI 120 Introduction to Computation
CPU speed (draft)

Saad Mneimneh
Visiting Professor
Hunter College of CUNY

1 Introduction

We will look at three main factors in determining the speed of the processor: Cache memory, pipelining, and the clock speed.

2 Cache memory

Let us now look at all kinds of memory that we have seen so far. We have:

- main memory (short term memory): this is used to hold data that will be needed by the program in the near future
- mass storage devices such as hard disks (long term memory): these are used to hold data that is unlikely to be needed in the near future, but that will eventually be needed at some point in time
- registers (very short term memory): these are used to hold the data immediately applicable to the current instruction being executed or to the operation at hand.

Figure 1 shows all three levels of memories with their placement relative to the CPU.

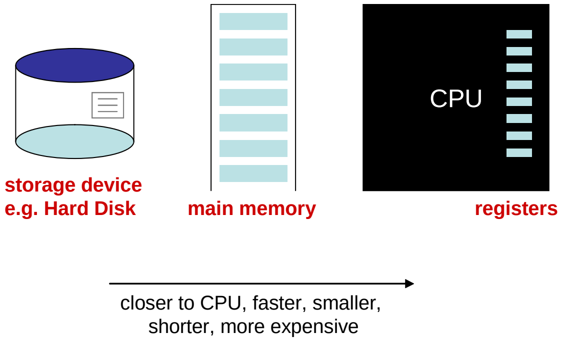

Figure 1: Memory levels

Most computers have an additional memory level called **cache memory** [1]. Cache memory is a high speed memory (but more expensive than RAM) located within the CPU itself (one may view it as being between main memory and the registers). It is usually few hundreds of KB. The computer attempts to keep a copy of the portion of main memory that is of current interest in the cache. Therefore, data transfers that would normally be made between registers and main memory are made between registers and cache memory. Any changes made to the cache are transferred collectively to main memory at a more opportune time. The result is that the CPU can execute instructions more rapidly because it is not delayed by the communication with main memory. This statement, however, needs clarification. It is conceivable for instance that the CPU must bring different portions of memory into cache very often (every few instructions). In this case, the use of cache is not justifiable, since the communication with memory is not reduced (in fact, may be increased). This usually does not happen because of the *locality* property. Programs and the data they use tend to be localized in main memory (the way they are loaded by the operating system). Even when there are jumps, the different portions of the program are close in memory.

The cache can be considered to be the CPU's *view of the world*, with the *real world* being main memory itself. With one CPU in the system, this *view of the world* is not problematic. With multiple CPUs, however, maintaining consistency between the *view of the world* (cache memory) and the *real world* becomes a problem. Each CPU will have its own cache. Therefore, a CPU may be accessing wrong information since it is not aware of any updates that other CPUs are performing (updates are local to their caches). Many cache-coherence protocols are used to solve this problem. Most of them include the capability for a CPU to **invalidate** the caches of other CPUs upon certain updates.

3 Pipelining and the machine cycle

Let us recall how instructions are performed by the CPU. First, an instruction is **fetched** from memory. Then the instruction is **decoded** by the control unit to determine its type and what the operands are. Finally, the instruction is **executed** by the ALU if needed. The three steps are repeated over and over until the program halts (or indefinitely). This periodic cycle, known as the machine cycle, is called the fetch-decode-execute cycle.

One way of speeding up the machine is by making faster fetch, decode, and execute steps. However, there is a limit imposed by the advances in technology and by physics too. For instance, bits cannot travel through wires faster than the speed of light!

Another way of speeding up the machine is by **pipelining** the machine cycle. The idea is the following. While an instruction is being decoded, the next one may be fetched, since the two steps use different circuitry. Similarly, while an instruction is being executed, the next one may be decoded. The effect of pipelining is illustrated below:

[1] The word cache means a safe place for hiding or storing things, from French *cacher*, which is to hide.

Without pipelining:

```
F  D  E  F  D  E  F  D  E  F  D  E  F  D  E
```

With pipelining:

```
F  D  E
   F  D  E
      F  D  E
         F  D  E
            F  D  E
```

Each instruction requires one more step with pipelining, compared to three more steps without pipelining. However, is it not always possible to pipeline the machine cycle perfectly as illustrated above, mainly because of Jump instructions. The next instruction to be fetch is not known until the conditional Jump executes! The standard solution for this problem is for the CPU to predict the result of the Jump and fetch the next instruction accordingly. If later on the CPU discovers that the prediction was wrong, it cancels the current instruction.

4 Clock speed

The processor relies on a small quartz crystal circuit called the **system clock** to control the timing of all CPU operations. The clock generates regular electronic pulses, or ticks, that set the operating pace of the different components of the CPU. Each tick is called a *clock cycle*. In the past, processors used one or more clock cycles to execute each intruction. Processors today can execute more than one instruction per clock cycle (depending on the type of instructions).

The clock speed is measured in the number of ticks per second, or *hertz* (Hz). Current personal computer processors have clock speeds in the Gigahertz (GHz) range. For instance, a processor operating at 2.33 GHz makes 2.33 billion ticks per seond (thus almost 2.33 billion instructions per second).

For even faster performance, several processors are now *dual-core*. A dual-core processor is a chip that has two separate processors. This is an example where cache coherence becomes important (the two processor may be accessing the same memory locations and, therefore, may have different views of the memory).

Note that the system clock is not the one that keeps track of the date and time. There is another battery-backed chip called the real-time clock that is responsible for that.

Here's a table showing few processors with their speeds.

Name	Date introduced/updated	Manufacturer	Clock speed
Pentium Exterme Edition (dual-core)	2005	Intel	3.2GHz
Pentium D (dual-core)	2005	Intel	2.8-3.2GHz
Pentium 4 with HT technology	2002/2005	Intel	2.4-3.8GHz
Pentium 4	2000/2005	Intel	1.3-3.8GHz
Pentium III	1999/2003	Intel	450MHz-1.4GHz
Celeron D	2004/2005	Intel	2.4-3.2GHz
Celeron	1998/2003	Intel	266MHz-2.8GHz
Athlon 64 X2 (dual-core)	2005	AMD	2.0-2.8GHz
Athlon 64 FX	2005	AMD	2.6-2.8GHz
Sempron	2004	AMD	1.5-2GHz
PowerPC (G1 to G5)	1994/2005	Motorola/IBM	60MHz-2.7GHz

5 Buying a laptop today

Here are some recommended features for a laptop:

- 1-2 GHz processor (dual-core an option)
- 512MB-2GB RAM
- 80-100 GB Hard Disk (SATA), 5400-7200 RPM
- USB 2.0
- Bluetooth (option)
- Internal wireless network card
- 64MB-256MB graphics card (memory on I/O controller for display)
- 14” SVGA/XVGA display (12” for portability)
- CD/DVD drive combo (not necessarily DVD recordable unless needed)
- Battery: get the smallest for light weight