

Lecture 6

*Lecturer: Emery Berger**Scribe: Arpita Raveendran, Patrick Pegus*

6.1 Architecture

We began the discussion about architectures with a description of the timeline of processors (chips) released over time.

Intel released an 8-bit processor, in 1974, called 8080. It had no cache. Having no cache was actually advantageous:

1. We knew exactly how long each instruction in the assembly code would take, so the programs had a deterministic execution time.
2. Simplified Performance analysis.
3. Simplified Optimization analysis.
4. Instruction scheduling.

However, the maximum RAM size was only 64K.

So, After 8080, Intel wanted to make chips which accommodate larger address sizes, which led to the release of 8088 in 1979(The chip used in IBMs PC). 8088 (a variant of 8086) had 640 K of RAM, along with some rudimentary memory protection.

IBM continued its development of faster, better chips with bigger word-sizes: 80286 in 1982 (16 bit micro-processors), 80386 in 1985 (32 bit microprocessor), 80486 in 1989.

The x-86 architecture was developed by Intel in the 8086 series. It was primarily a CISC architecture.

Way after the paper favoring RISC over CISC written by Patterson, Intel invented a separate chip called 8087 (A math floating point co-processor) which could be plugged onto the mother board alongside the 8086. This x87 architecture would work alongside x86. Instead of hacking into the software to optimize things, the 8087 co processor was included to perform operations with complex instructions. At that time, the problem was that the processor and co-processor communicated slowly as they did not share cache. Nowadays, they are closely integrated. Continuing the timeline, Intel went on to develop the Pentium microprocessors.

On a side note, we also discussed a bit of Operating Systems: IBMs PC worked a DOS 1.0 OS. It was a single user, flat file system with no file permissions. Lisa was a PC developed by Apple, in the early 1980s, with a 16-bit chip, lots of RAM and also was the first PC with a GUI. However, it was too expensive and a toned down version of that was the MACINTOSH, released in 1985. Around 1986, Windows 3.1 was released which dominated PCs for 15 to 20 years. Windows 95 followed.

6.2 CISC vs. RISC

1. CISC Complex Instruction Set Computer. A current example is x86.
2. RISC Reduced Instruction Set Computer. A current example is ARM.
3. There is no technical definition of either, but RISCs have a uniform instruction size while CISCs are often variable.
4. With CISCs, it is difficult to differentiate between code and data, which is inherently insecure. Some examples of exploits are:

Drive-By malware Visit a web page with your browser. The browser downloads JavaScript with embedded x86 code. Computer executes code through buffer overflow.

Morris worm Self-replicating malware created by Robert Morris took down the internet in 1988. Morris was convicted of a felony and is now a professor at MIT.

5. x86 is by the most well known CISC.
 - (a) Examples of complex x86 instruction sets:
 - SIMD** Single Instruction Multiple Data such as vector operations
 - MIMD** Multiple Instruction Multiple Data
 - (b) Why has Intel's x86 continued to be so popular?
 - i. Intel makes really good silicon. The n in their n nm process continues to decrease, which is the distance between traces on their chips. This makes their chips faster, regardless of instruction set.
 - ii. They started from scratch, teaming up with HP to make the Itanium, which is RISC like. It used very long instruction words (VLIWs) and relied on compiler optimization, which did not work.
 - iii. Although it easier to write a compiler for reduced instruction sets, compiler writers are an insignificant portion of a processor maker's market.
 - iv. Intel's complex instruction set is only an interface whose operations are implemented in RISC like micro-ops or hardware. This is not a leaky abstraction. It allows Intel to introduce a complex instruction implemented in micro-ops and switch to a hardware implementation in later generations. There is a compiler on chip.
6. Coprocessor evolution
 - (a) The industry often creates discrete processors for specialized tasks that are eventually incorporated into the CPU.
 - (b) Examples:
 - FPU** The Intel 8087 math coprocessor or floating point unit was an optional processor that shared memory with the 8088 CPU and was located in a socket next to it.
 - GPU** Graphics cards contain GPUs, but GPUs are often now included in the CPU.
 - Other examples** Translation lookaside buffer (TLB) and vector machines
7. Loop unrolling is process of replacing jumps with repeated code. It results in less condition checking and more instruction level parallelism extraction.