

HISTORY OF MICROPROCESSOR - I

The microprocessor, or CPU, as some people call it, is the brains of our personal computer. I'm getting into this history lesson not because I'm a history buff (though computers do have a wonderfully interesting past), but to go through the development step-by-step to explain how they work.



Well, not everything about how they work, but enough to understand the importance of the latest features and what they do for you. It's going to take more than one article to dig into the inner secrets of microprocessors. I hope it's an interesting read for you and helps you recognize computer buzzwords when you're making your next computer purchase.

1. Where Did CPUs Come From?

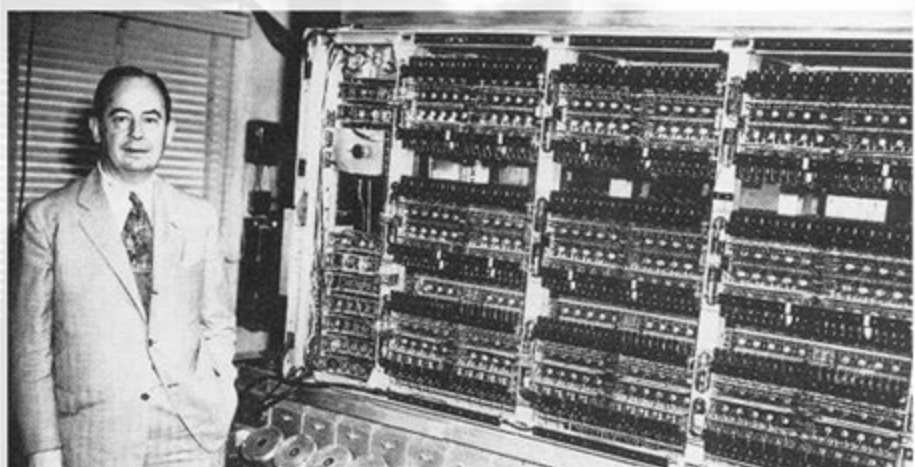


When the 1970s dawned, computers were still monster machines hidden in air-conditioned rooms and attended to by technicians in white lab coats. One component of a mainframe computer, as they were known, was the CPU, or Central Processing Unit. This was a steel cabinet bigger than a refrigerator full of circuit boards crowded with transistors.

Computers had only recently been converted from vacuum tubes to transistors and only the very latest machines used primitive integrated circuits where a few transistors were gathered in one package. That means the CPU was a big pile of equipment. The thought that the CPU could be reduced to a chip of silicon the size of your fingernail was the stuff of science fiction.

2. How Does a CPU Work?

In the '40s, mathematicians John Von Neumann, J. Presper Eckert and John Mauchly came up with the concept of the stored instruction digital computer. Before then, computers were programmed by rewiring their circuits to perform a certain calculation over and over. By having a memory and storing a set of instructions that can be performed over and over, as well as logic to vary the path of instruction, execution programmable computers were possible.



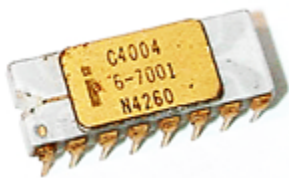
Mathematicians John Von Neumann, J. Presper Eckert and John Mauchly came up with the concept of the stored instruction digital computer.

The component of the computer that fetches the instructions and data from the memory and carries out the instructions in the form of data manipulation and numerical calculations is called the CPU. It's central because all the memory and the input/output devices must connect to the CPU, so it's only natural to keep the cables short to put the CPU in the middle. It does all the instruction execution and number calculations so it's called the Processing Unit.

The CPU has a program counter that points to the next instruction to be executed. It goes through a cycle where it retrieves, from memory, the instructions in the program counter. It then retrieves the required data from memory, performs the calculation indicated by the instruction and stores the result. The program counter is incremented to point to the next instruction and the cycle starts all over.

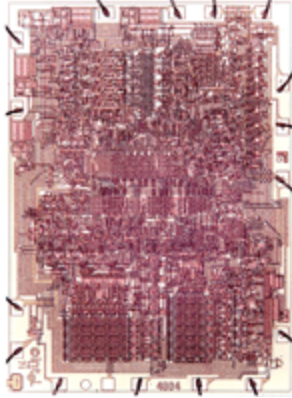
3. The First Microprocessor

In 1971 when the heavy iron mainframe computers still ruled, a small Silicon Valley company was contracted to design an integrated circuit for a business calculator for Busicom. Instead of hardwired calculations like other calculator chips of the day, this one was designed as a tiny CPU that could be programmed to perform almost any calculation.



The expensive and time-consuming work of designing a custom wired chip was replaced by the flexible 4004 microprocessor and the instructions stored in a separate ROM (Read Only Memory) chip. A new calculator with entirely new features can be created simply by programming a new ROM chip. The company that started this revolution was Intel Corporation. The concept of a general purpose CPU chip grew up to be the microprocessor that is the heart of your powerful PC.

4. 4 Bits Isn't Enough

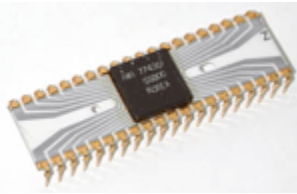


The original 4004 microprocessor chip handled data in four bit chunks. Four bits gives you sixteen possible numbers, enough to handle standard decimal arithmetic for a calculator. If it were only the size of the numbers we calculate with, we might still be using four bit microprocessors.

The problem is that there is another form of calculation a stored instruction computer needs to do. That is it has to figure out where in memory instructions are. In other words, it has to calculate memory locations to process program branch instructions or to index into tables of data.

Like I said, four bits only gets you sixteen possibilities and even the 4004 needed to address 640 bytes of memory to handle calculator functions. Modern microprocessor chips like the [Intel Pentium 4](#) can address 18,446,744,073,709,551,616 bytes of memory, though the motherboard is limited to less than this total. This led to the push for more bits in our microprocessors. We are now on the fence between 32 bit microprocessors and 64 bit monsters like the [AMD Athlon 64](#).

5. The First Step Up, 8 Bits



With a total memory address space of 640 bytes, the Intel 4004 chip was not the first microprocessor to be the starting point for a personal computer. In 1972, Intel delivered the 8008, a scaled up 4004. The 8008 was the first of many 8-bit microprocessors to fuel the home computer revolution. It was limited to only 16 Kilobytes of address space, but in those days no one could afford that much RAM.

Two years later, Intel introduced the 8080 microprocessor with 64 Kilobytes of memory space and increased the rate of execution by a factor of ten over the 8008. About this time, Motorola brought out the 6800 with similar performance. The 8080 became the core of serious microcomputers that led to the Intel 8088 used in the IBM PC, while the 6800 family headed in the direction of the Apple II personal computer.



6. 16 Bits Enables the IBM PC



By the late '70s, the personal computer was bursting at the seams of the 8 bit microprocessor performance. In 1979, Intel delivered the 8088 and IBM engineers used it for the first PC. The combination of the new 16 bit microprocessor and the name IBM shifted the personal computer from a techie toy in the garage to a mainstream business tool.

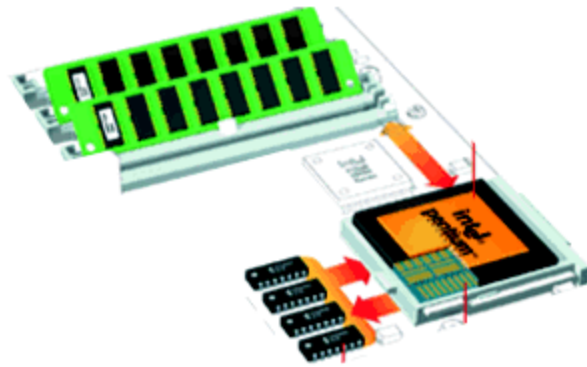


The major advantage of the 8086 was up to 1 Megabyte of memory addressing. Now, large spreadsheets or large documents could be read in from the disk and held in RAM memory for fast access and manipulation. These days, it's not uncommon to have a thousand times more than that in [a single 1 Gigabyte RAM Module](#), but back in that time it put the IBM PC in the same league with minicomputers the size of a refrigerator.

7. Cache RAM, Catching Up With the CPU

We'll have to continue the march through the lineup of microprocessors in the next installment to make way for the first of the enhancements that you should understand. With memory space expanding and the speed of microprocessor cores going ever faster, there was a problem of the memory keeping up.

Large low-powered memories cannot go as fast as smaller higher power RAM chips. To keep the fastest CPUs running full speed, microprocessor engineers started inserting a few of the fast and small memories between the main large RAM and the microprocessor. The purpose of this smaller memory is to hold instructions that get repeatedly executed or data that is accessed often.



This smaller memory is called cache RAM and allows the microprocessor to execute at full speed. Naturally, the larger the cache RAM the higher percentage of cache hits and the microprocessor can continue running full speed. When the program execution leads to instructions not in the cache, then the instructions need to be fetched from the main memory and the microprocessor has to stop and wait.

8. Cache Grows Up

The idea of cache RAM has grown along with the size and complexity of microprocessor chips. [A high-end Pentium 4](#) has 2 Megabytes of cache RAM built into the chip. That's more than twice the entire memory address space of the original 8088 chip used in the first PC and clones. Putting the cache right on the microprocessor itself removes the slowdown of the wires between chips. You know you are going fast when the speed of light for a few inches makes a difference!

9. Cache Splits Up

As I mentioned above, smaller memories can be addressed faster. Even the physical size of a large memory can slow it down. Microprocessor engineers decided to give the cache memory a cache. Now we have what is known as L1 and L2 cache for level one and level two. The larger and slower cache is L2 and is the usual size quoted in specifications for cache capacity. A few really high-end chips like the Intel Itanium II had three levels of cache RAM.

Beware that the sheer size of cache RAM or the number of layers are not good indications of cache performance. Different microprocessor architectures between Intel and AMD make it especially hard to compare their cache specifications. Just like Intel's super high clock rates don't translate into proportionately more performance, doubling of cache size certainly doesn't double the performance of a microprocessor. Benchmark tests are not perfect, but are a better indicator of microprocessor speed than clock rate or cache size specifications.

