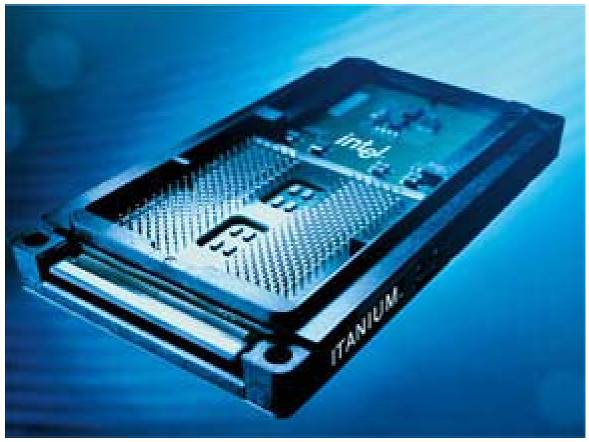
**CPU-Table of Contents**

**Introduction to the Processor (CPU)**

Many people consider the CPU (Central Processing Unit) to be the brains of the computer. This analogy is very loose because, for the most part, the CPU cannot keep data stored inside it like a brain. In contrast, it is used to process much of the information needed by the computer, just like our brain thinks and processes information and gives orders to our other body parts.




Intel's Next generation processor, the Itanium, featuring 64-bit architecture.
© Intel Corp, 2001

Over the past few years, we have seen the CPU MHz speeds go from 100 MHz to over 2 GHz (1000 MHz = 1 GHz). This is one reason that people need to learn about a CPU. Many people would expect a 1.8 GHz Intel Pentium 4 to be much faster than a 1.4 GHz AMD Athlon because its speed is 0.4 GHz faster. In truth, not only is the Intel Pentium 4 up to three times more expensive than the AMD Athlon, it is either much slower or neck to neck in most "Real world tests", which compares the amount of times that it takes each CPU to perform a certain task.

With this information, you know that you should not judge a computer by the "speed ratings". But if one CPU goes at a faster MHz rate and is slower, what does determine the speed of the CPU? There are a variety of factors, but we will show you the main parts of a CPU, and what they are used for.

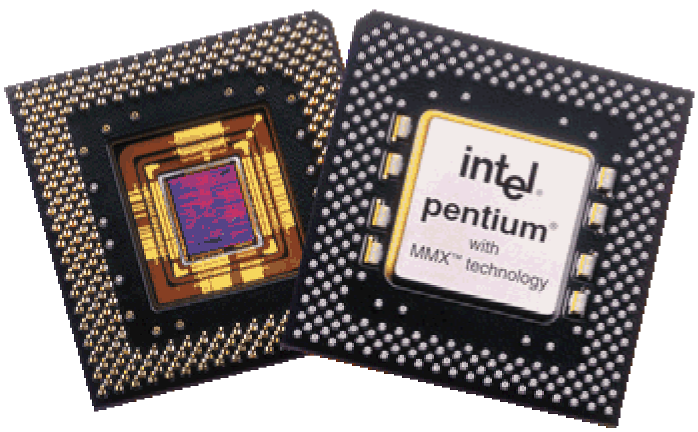
## CPU Basics

When looking at a CPU, there are a few basic things that we should know about it:

- A CPU has four basic tasks that it performs. They are **Fetch, Decode, Manipulate** and **Output**.
- Speed rating, although not accurate, is almost always measured by MHz.
- The CPU speed is determined by a combination of raw MHz as well as design and other features such as the FPU of the chip.

Before we get into details about how it works, we must remember that, like many other computer parts, the CPU is comprised of millions of logic gates embedded into it which then are used to complete a variety of different operations. The size of the CPU core, the part with the logic gates, can be as small as the size of a smaller coin.

The gates are used with a clock that regulates the speed at which the CPU is fed data. The speed at which it does this is measured in Hz (amount of clock pulses in one second), MHz (about 1 million Hz) and GHz (about 1000 MHz). If there was no clock to regulate the data flow, the CPU would be unorganized and useless. The clock does a similar thing for the CPU as traffic lights do for the traffic. It makes everything organized and tells when the data should pass through, and when it should not.



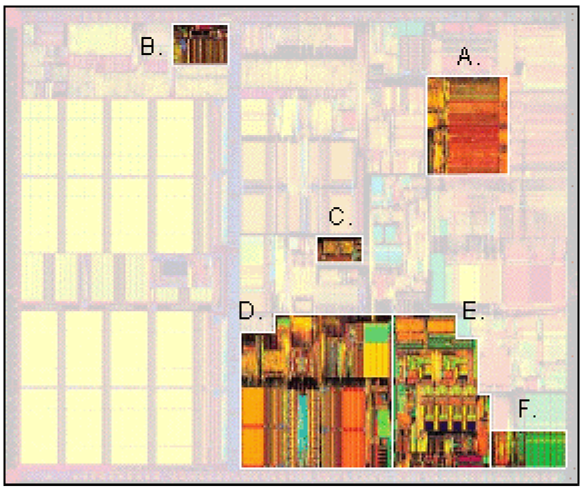

Aside from the high MHz speed, Intel's MMX technology made the Pentium a dominating CPU for the PC
© Intel Corp, 2001

**CPU Processing**

As you could guess, if you read the binary part of the website, the CPU processes only binary data. This means that all of the data, such as numbers, letters, colors, or actions to be performed, are stored in bits and bytes.

To process data, a CPU really only needs 6 basic parts to do its job:

| | |
|---|---|
| **Instruction Pointer** | This tells the CPU where the instruction or data is stored. Before a CPU can process data, it needs to know where the information should be taken from. |
| **Instruction Fetch** | Takes the instruction and data from the part of the memory that the instruction pointer specifies. |
| **Instruction Decoder** | Takes the instruction from Fetch, and decodes it so that it can be used by other parts of the CPU or the Computer. |
| **Register** | One of the more basic parts of memory where the data is stored. It can store data which has been processed or will be processed. |
| **ALU** | Arithmetic Logic Unit- used by the CPU to perform the calculations; completes the functions that the instructions tell it to perform. |
| **Control Unit** | It specifies when the different operations on the CPU are to be done. For example, it tells the instruction fetch when to fetch data, or the instruction decoder when to decode data. |




A. ALU - Arithmetic Logic Unit - used by the CPU to perform the calculations; completes the functions that the instructions tell it to perform.

B. Clock - acts as a regulator for the CPU to keep the flow of data synchronized.

C. Control Unit - specifies when the different operations on the CPU are to be done. For example, it tells the instruction fetch when to fetch data, or the instruction decoder when to decode data.

D. Instruction Decoder - takes the instruction from Fetch, and decodes it so that it can be used by other parts of the CPU or the Computer.

E. Instruction Pointer - this tells the CPU where the instruction or data is stored. Before a CPU can process data, it needs to know where the information should be taken from.

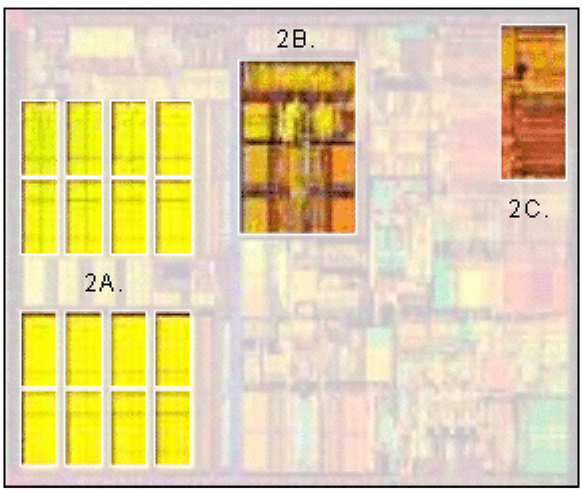
F. Instruction Fetch - takes the instruction and data from the part of the memory that the instruction pointer specifies.

G. Instruction Fetch - takes the instruction and data from the part of the memory that the instruction pointer specifies.

## CPU Enhancements

Although the CPU needs only a few parts to do its job, it has been modified to increase the performance. The enhancements are made in order to process data faster. One of the weaknesses of the basic CPU was that it was not doing any processing during the time that it fetched an instruction from the computer memory. To reduce this, a memory storage area was created inside the CPU called Cache or L1 type Cache. The Cache is very fast, and is used to store information that the CPU needed to access the most. Level 2 Cache was also created for the CPU. It was similar to L1 Cache.

Now that the CPU has to wait much less for the data to be processed, the speed at which the data can be processed had to be enhanced. To do this, multiple ALUs (Arithmetic Logic Unit) were placed on the CPU, so that many more calculation was done each clock cycle. In addition, the FPU (Floating Point Unit) was added. The FPU is similar to the ALU, except it acts as a sort of specialist. It could handle extremely large and extremely small numbers much better than the FPU, allowing greater processing speeds. Also, it can simultaneously process multiple graphics and sounds at a time.




2A. Cache - memory inside the CPU used to temporarily store data.

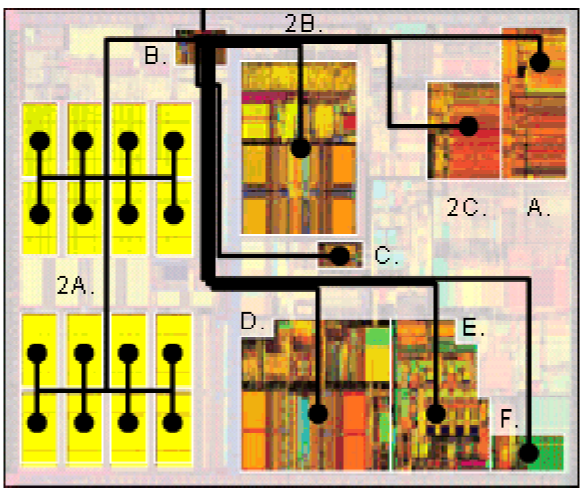
2B. L2 Cache - memory inside the CPU used to store the most common data used by the CPU, reducing the waiting time for the CPU.

2C. FPU - similar to the ALU except it handles large and small numbers extremely well. It is also capable of processing multiple sounds and graphics at the same time.

**What Happens to Instructions, Step By Step**

Using the parts of the CPU displayed on the previous page, we will trace the steps that an instruction goes through in the CPU. First, the instruction pointer tells the instruction fetch where in the memory the instruction is. The fetch takes the instruction and gives it to the decoder, which determines the steps that are necessary to fulfill the instructions. The information is then sent to the ALU, which performs the instructions that need to be performed. This includes adding, subtracting, or manipulating the data further. Finally, the instructions are sent out into the computer where they are needed.

This process continues whenever a CPU needs to do anything with any information, but it happens at an extremely fast pace. To make sure that everything happens at the right time, a clock generator is used to regulate the flow of data. The pulses that the clock generator sends are measured in the amount of pulses per second, or Hz. Today's computers have clocks that pulse millions of times per second, or MHz.; If two CPUs have every part that is the same, and the clocks were different speeds, the faster-clocked CPU would perform tasks faster. Since CPUs are not all the same, the efficiency of the CPU parts matter just as much, if not more, than the speed of the clock generator.




Using the parts of the CPU displayed on the previous page, we will trace the steps that an instruction goes through in the CPU.

First, the instruction pointer tells the instruction fetch where in the memory the instruction is.

The fetch takes the instruction and gives it to the decoder, which determines the steps that are necessary to fulfill the instructions.

The information is then sent to the ALU, which performs the instructions that need to be performed. This includes adding, subtracting, or manipulating the data further.

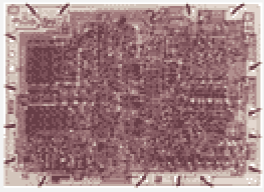
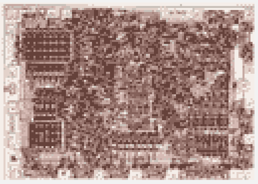
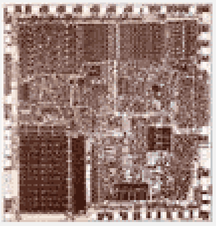
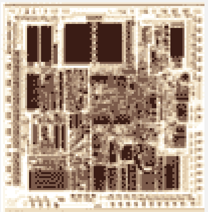
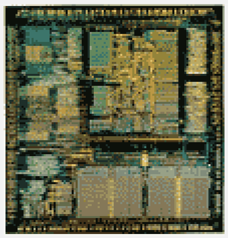
This process continues whenever a CPU needs to do anything with any information, but it happens at an extremely fast pace.

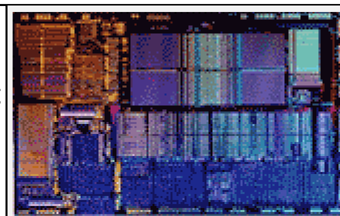
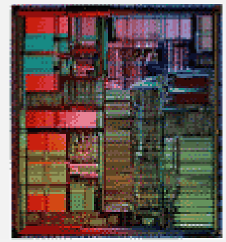
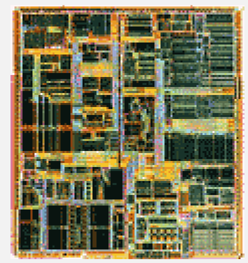
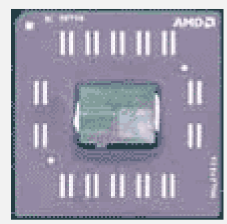
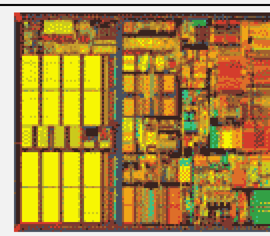
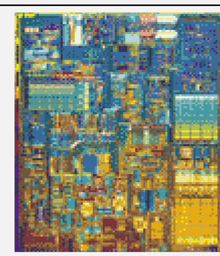
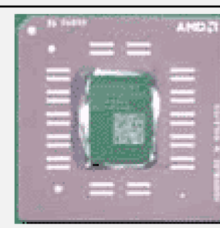
To make sure that everything happens at the right time, a clock generator is used to regulate the flow of data. The pulses that the clock generator sends are measured in the amount of pulses per second, or Hz.

Today's computers have clocks that pulse millions of times per second, or MHz. If two CPUs have every part that is the same, and the clocks were different speeds, the faster-clocked CPU would perform tasks faster. Since CPUs are not all the same, the efficiency of the CPU parts matters just as much if not more than the speed of the clock generator.

## CPU History

Here is a brief history of the important CPUs which were featured in PCs over the years. Although it has been dominated by Intel, we can see AMD coming out with some cutting-edge CPUs in the last few years.

| Processor | Description | Core Photo |
|---|---|---|
| **Intel 4004 (1971)** | Intel's first microprocessor. It was a breakthrough in computer technology, powering one of the first electronic calculators. |  |
| **Intel 8008 (1979)** | This is the first chip that was used in a PC. It could run at 4 MHz and supported up to 1 MB of system RAM. |  |
| **Intel 80186 (1980)** | The 186 was a very popular CPU. There are two versions, an 8-bit or 16-bit (the amount of bits allowed to be inputted each clock cycle). The 186 eventually reached a speed of 25 MHz, using 3 volts. |  |
| **Intel 80286 (1982)** | This is a 16-bit processor which supports up to 16 MB of RAM. It was the first processor to be able to multitask (run multiple programs at a time), but the operating systems at that time could not take advantage of it. The chip ran as high as 20 MHz. |  |
| **Intel 80386 (1988)** | This was a revolutionary chip for the PC industry. It was the first 32-bit processor, which meant it could use twice as much data on each clock cycle. The 386 was also capable of using 16 bytes of cache. Having speeds from 12.5 MHz to 33 MHz, it was a big step for the PC, and was very user-friendly. |  |

| | | |
|---|---|---|
| **Intel 486 (1991)** | The 486 used much of the 386 architecture, but it added a math coprocessor, which made it much faster. It can go up to 120MHz. It also came in an SX version, which was cheaper to make because no math coprocessor was utilized. |  |
| **Intel Pentium (1993)** | The Pentium was available in speeds from 75MHz all the way up to 233MHz. It had an FPU, which allowed much grater performance. Internally, it had two 32-bit chips which split the work. The chip came with 16 KB of cache. |  |
| **Intel Pentium II (1997)** | Built with over 7.5 million transistors, this processor included MMX technology, used to process video and audio faster. This CPU used a cartridge to connect to the motherboard. It had a built in L2 Cache, making it a very fast processor. |  |
| **AMD Athlon (1999)** | The first processor that was faster than its Intel counterpart, the Athlon made history. It featured 256Kb of cache, as well as 3DNow instructions designed to improve the FPU. It uses a 266MHz bus, twice as fast as the Pentium III. AMD made history with the Athlon by being the first to break the 1GHz barrier. |  |
| **Intel Pentium III (1999)** | Performing similarly to the AMD Athlon, the Pentium III features 256Kb of cache. It comes in speeds from 450MHz all the way up to 1.13 GHz. |  |
| **Intel Pentium 4 (2001)** | This chip is actually much slower than its predecessor, but the MHz advantage it has over the Pentium III made it faster. It is especially fast on internet applications, although it cannot compete with its AMD counterpart in any other area. |  |
| **AMD Thunderbird (2001)** | Currently the best PC processor on the market, AMD dominates the world of 3D gaming as well as professional programs. It comes in first in almost every test, even though its clock is about 0.5GHz slower than the Pentium 4. |  |

## MHz Myth

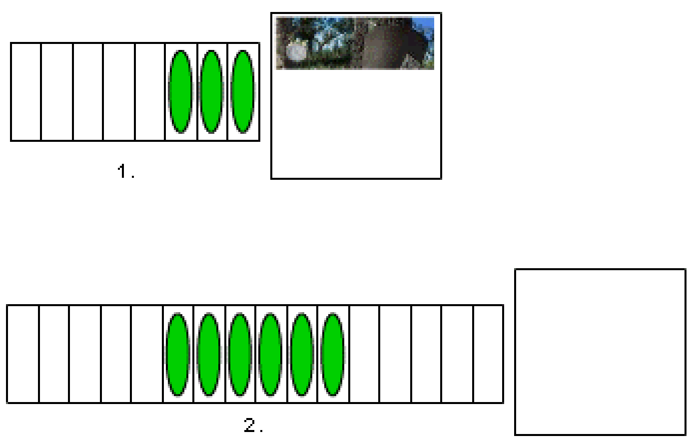
Over the history of the PC, one dominating factor that influences which computer to buy, is the speed of the CPU measured in MHz. This is a major reason why Intel has dominated the market for a very long time. The performance of a CPU does not depend solely on the MHz. The MHz speed, or Frequency, is merely one of the contributing factors.

For a CPU to be fast, it has to have a combination of fast frequency (MHz), efficiency in calculations (FPU and ALU), a bus as wide as possible (currently 32-bit, although Intel and AMD are already working on 64-bit models) as well as a pipeline which has to be as short and efficient as possible. The longer the pipeline, the more time it takes for the CPU to output data. The AMD Athlon has an excellent FPU, which gives its "slower" CPU an edge over the faster clocked Intel Pentium 4. The Athlon also features a shorter, more efficient pipeline, which makes it an excellent choice for personal and professional computers.

## Shorter Pipeline

To demonstrate how a CPU benefits from a shorter pipeline, here is a basic model of data being processed to make a picture. There are two pipelines (one long and one short) to illustrate the advantages of an efficient pipeline.

Pipeline Length




| |
|---|
| 1. Shorter Pipeline |
| 2. Longer Pipeline |
| The length of the pipeline directly affects the time it takes for the CPU to complete a certain task. You will soon see the advantages of having a shorter Pipeline. In this demo, we will load two pictures, same in size, with the two different sized pipelines to show you the advantages. |

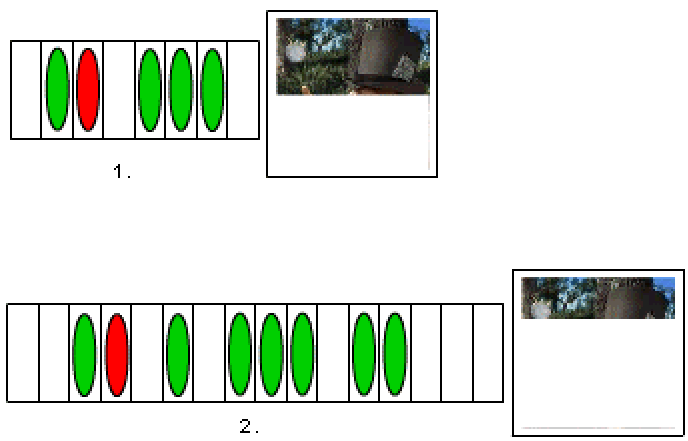
| |
|---|
| We see the data flowing through the pipeline. Since the first pipeline is shorter, the data will be processed quickly, while the longer pipeline struggles to finish its processing. |
| While the shorter pipeline has already completed the task of loading the picture, the longer pipeline has not finished processing any information, and no part of the picture can be seen. |
| Finally, the longer pipeline has finished loading the picture. We now see the big advantage of a shorter pipeline: EFFICIENCY. |

## Pipeline Tax

Now, you might be thinking, "Wait a minute, if the CPU with the shorter Pipeline runs at a slower pace, wouldn't the higher-clocked CPU eventually catch up to the slower CPU?". In a perfect world, this might be true, but just like everything, CPU's are not perfect.

When a CPU is fed data, there are sometimes branches of data which causes a pipeline to drain. Although this does not happen very often, you must remember that there are millions of pieces of data being fed to the CPU every second, and a large amount of the data can contain bad parts. To illustrate how this impacts a CPU with a longer pipeline more than a CPU with a shorter pipeline, we will once again be using two CPUs. The CPU with the longer pipeline will be running at 150% the clock speed of what the CPU with the shorter pipeline will be running.

Pipeline Tax




| |
|---|
| 1. Shorter Pipeline |
| 2. Longer Pipeline |
| This Flash will demonstrate Pipeline Tax. The CPU with the longer pipeline is 150\% the speed of the shorter pipeline. You will see why the slower CPU can finish its jobs faster than the faster CPU. NOTE: the red piece of data represents bad data, which causes the pipeline to drain. |
| The data starts flowing through both of the pipelines. |
| Even though the CPU with shorter pipeline is slower, it will start finishing the data first. When the bad data comes to the end, it will cause the whole pipeline to drain. This will not affect the shorter CPU as much, because it features a shorter pipeline. |
| The pipeline empties and starts where it left off... The longer pipeline will follow. |
| The second bad piece of data will soon drain the smaller pipeline. The longer pipeline is falling even more behind. |
| The higher clocked CPU is finally finished. Even though it was 50% faster in frequency, it still could not keep up with the slower, more efficient CPU. The margins are even bigger in real life because much more data gets processed, which |

| |
|---|
| shows the importance of an efficient pipeline. The smaller Pipeline CPU is faster because Pipeline tax does not affect it as drastically. |

We hope that you now have a bigger understanding of why the CPU speed cannot be measured in MHz.