

Multi Core CPUs

Russell Hitchcock

In the computing industry there is a major shift afoot. Over the next few years you will see your networks overtaken by multi-core machines. This will affect many aspects of your network, from licensing and software development costs, to the entire focus of your network architecture. In this article I will help illustrate why the industry is moving in this direction and will explain some of the impacts it will have on you and your network.

Terminology

Terminology in the IT industry can be confusing. So, let's get something straight right off the bat. A multi-core CPU has two or more processing units on the same integrated circuit. This is different from the term "multi-chip" which refers to multiple integrated circuits packaged together. This is different still from the term "multi-CPU" which refers to multiple processors working together.

Advantages

So why would hardware designers want to put the CPUs on the same chip? Well, one great reason is that putting multiple cores on one integrated circuit in one package takes up less room on the printed circuit board than the equivalent amount of single core CPU packages. Another, less obvious, advantage is that since multiple cores on a single integrated circuit are physically close together the cache coherency is greatly improved.

Power savings can also be realised with multi-core processors. Since the cores are on the same chip, signals between the cores travel shorter distances. Also, multi-core CPUs typically run at a lower voltage and, since the power lost to a signal travelling over a wire is equal to the square of the voltage divided by the resistance in the wire, a lower voltage will result in less power loss.

Another possible area for power savings is with the clock speed. You see, multi-core CPUs can perform many more times the operations per second even while operating at a lower frequency. For example, the 16 core MIT RAW processor operates at 425 MHz and can perform over 100 times the number of operations per second as an Intel Pentium 3 running at 600 MHz. How does the frequency affect the power consumption of a CPU? Well, it's quite a complicated process, but a basic rule of thumb is that for every one percent increase in clock speed you will see a three percent increase in power consumption. That, of course, assumes that the other factors which affect power consumption have not been altered.

Multi-core CPUs also have the ability to share a bus interface as well as cache circuitry. Figure 1 shows a diagram of the Intel Core 2 dual, which features a shared L2 cache. This can result in significant space savings. According to Intel, the Core 2 dual CPU has up to a 4 MB shared L2 Cache.

Multi Core CPUs

Russell Hitchcock

Dual CPU Core Chip

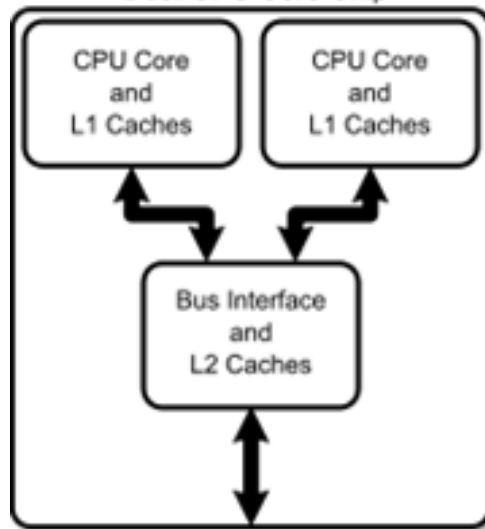


Figure 1: Diagram of Intel Core 2 Dual Processor

Tilera's Tile64

Speaking of a CPU's cache, Tilera's recently announced Tile64 (a 64 core processor) has a unique cache feature. The Tile64 utilises a mesh architecture, as shown in Figure 2. This mesh architecture allows the individual cores to do something quite unusual. When the core looks in its L2 cache and cannot find what it is looking for, it first looks in the L2 caches of the other cores in the mesh before requesting the data from main memory. This basically means that the mesh acts like an L3 cache.

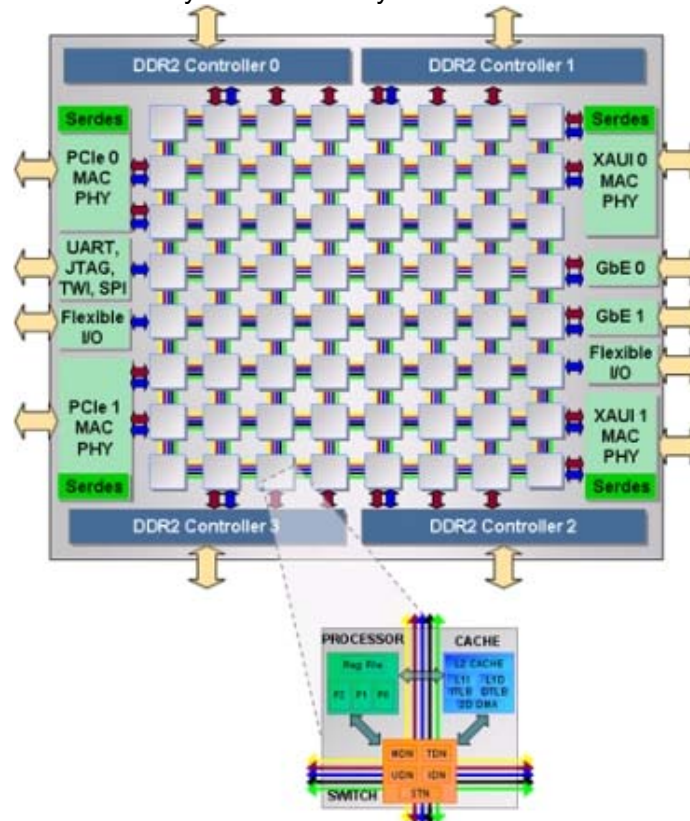


Figure 2: Diagram of the Tile64

Multi Core CPUs

Russell Hitchcock

Architecture

The unique cache feature of the Tile64 is an example of a major shift in computer architecture. Currently computers are centered around memory, with the processors accessing this memory. This requires a lot of communication overhead and is also a major bottleneck and limiting factor for speed of operation. With the adoption of multiple cores, the industry is moving towards a more processing and communication centric architecture. This new architecture is both faster and more efficient in its power usage.

The current memory-centric architecture is not capable of realising the full advantages of multiple cores. For example, a typical cache read takes only 10 percent of the energy needed to read an off-chip memory location. The speed of an off-chip read operation will also be limited by both the memory technology used and the connection medium used which is not typically scalable.

As multi-core processors become more common they will become more affordable. When this happens, software developers will begin to develop truly multi-threaded applications. This is when you will see a change in your networks. Your networks will likely change from being memory centric to processing and communication centric. Your networks of course won't likely be using dual or quad core processors (except perhaps with the user's computer). Your network equipment will have tens or hundreds of cores.

Multi-threading

But what's this about software developers? Well, another factor which limits the performance advantages of multi-core CPUs is the software which runs on it. For the average user, the largest performance gains seen when switching to a multi-core CPU is with improved multi-tasking. For example, with a multi-core CPU you will see a large improvement if you are watching a DVD while doing a virus scan. This is because, each application will be assigned to different cores.

If a user is running a single application on a multi-core machine there will likely not be significant performance advantages. This is because most applications are not truly multi-threaded. Applications may appear to be multi-threaded, for example a virus scan may start a new thread while the GUI runs in another thread. This is not true multi-threading. True multi-threading is when the bulk of the work is divided into threads. In the virus scan example, the GUI thread does very little work, while the virus scan thread does the heavy lifting and it is not capable of being divided up and sent to different cores.

Developing a truly multi-threaded application requires a lot of very difficult work. This obviously adds significant costs to the software design cycle. That is why the majority of software applications will not be developed as truly multi-threaded applications until the number of cores is significantly high so that multi tasking does not realise any performance gains. This is when the user will demand it.

Your networks are a slightly different story though. Routers are likely to be the first widely adopted machines with multiple cores as well as significant multi-threading. Servers will also see significant gains from multiple cores and multi-threading. Some of you now may be thinking, aren't these products already multi-core? Well, yes, many are. I'm talking about a significant jump in the amount of cores. Intel has promised to deliver an 80 core by 2011, this is what I am referring to.

Licensing

The next question you're likely to ask is, How does this affect my software licensing? Currently, this is a difficult question to answer. Many software companies will only require one license to run on multi-core CPUs. Of course, this usually only applies to CPUs with two or perhaps four cores. Microsoft has stated that they will continue to license their server software on a per processor basis and not on a per core basis. This certainly appears to be the way the industry is moving. However, we can only wait and see what software companies will do once we want to run their software on 80 cores.

Multi Core CPUs

Russell Hitchcock

While there are some unknowns related to licensing, and some definite disadvantages related to software development, the move towards multi-core processors is definitely a good thing. Over the last few years, you've probably noticed that CPU speeds have increased dramatically, while their performance has only increased marginally. These diminishing returns are the real motivation behind the move to multi-cores. Multi-core technology is the only technology which can truly deliver significant performance gains. I don't know about you, but I can hardly wait!