64-Bit CPUs: AMD Hammer vs. Intel IA-64

Rui Martins

Departamento de Informática, Universidade do Minho
4710 - 057 Braga, Portugal
ruiaugusto@netcabo.pt

Abstract. The AMD Hammer processor architecture is designed to provide a migration path from IA-32 to 64-bit applications. This communication presents an overview of its architecture, stressing the integrated memory controller and the high-speed scalable system bus, aiming a high performance multi-way processing solution. It also shows some implementations differences between the Hammer and Itanium processors, namely in their 32-bit backwards compatibilities, and on their performance ratings.

1 Introduction

Current chipsets manufactured by AMD and Intel are based on twenty year old x86 ISA architectures. For some time now, the discussion over making a manufacturing move to a 64-bit based architecture has been a highly contested strategy within these competing companies.

AMD is currently developing the Hammer, a real IA-32 extension. The Hammer chips will extend the x86 instruction set by introducing a 64-bit extension called long mode as well as register extensions.

The 64-bit mode supports 64-bit code through the addition of eight general-purpose registers and widens them along with the instruction pointer. The 64-bit mode also adds eight 128-bit floating-point registers. It supports a x86 legacy mode, which preserves compatibility with 16- and 32-bit applications.

Hammer has many features that are also present in modern RISC processors: it supports out-of-order execution, has multiple floating-point units, and can issue up to 9 instructions simultaneously. Like in RISC processors, there is a branch prediction table assisting in branch prediction.

The floating-point part of the processor contains three units: a floating store unit that stores results to the load/store queue unit and floating add and multiply units that can work in superscalar mode, resulting in two floating-point results per clock cycle.

The strengths of Hammer are that it has an on-chip memory controller (which reduces latency), a very strong floating point unit and a HyperTransport system bus. It allows a "glueless" connection of several processors to form multi-processor systems with very low memory latencies.

On the downside, the ClawHammer (desktop CPU version) will have only 256 KB L2 cache, a slightly slower L1 cache, and a relatively slow (1.5-2.0GHz) clock rate.

On the other hand, Intel has developed the IA-64, an EPIC (Explicitly Parallel Instruction Computer) design technology. Intel's Itanium chips can run 32-bit applications, but Itanium chips are not slated to include legacy processor cores dedicated to running 32-bit applications. Itanium processors will modify 32-bit instructions to run the applications.

To properly address all these subjects, this communication is organized in four sections: the first deals with AMD Hammer architecture and symmetric multiprocessing, sections 3 and 4 present some implementation differences between the Hammer and the Itanium processors, and the last one concludes by exposing some of the author's remarks on the advantages, the challenges and the future of these two technologies.

2 AMD's Hammer Architecture

The Hammer processor consists of the CPU core, L1 and L2 caches, and an onboard North Bridge as illustrated in figure 1. The North Bridge is based on AMD's new bus technology, HyperTransport. The North Bridge is supposed to have a total bandwidth of 19.2 GBps across three HyperTransport interfaces. Among the 19.2 GBps of total bandwidth, the North Bridge provides 6.25 GBps of bandwidth for IO devices (i.e., the PCI, AGP, and USB buses) and 2.7 GBps for the memory controller (illustrated in fig.3), which leaves 10.25 GBps for interprocessor communications.

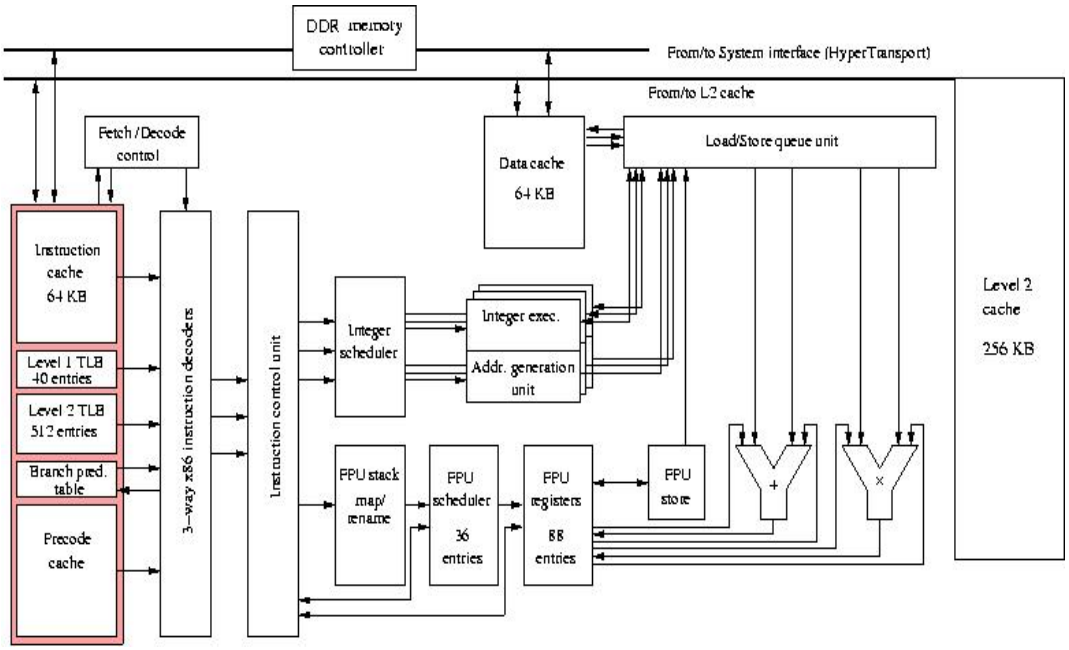

Fig. 1. Block diagram of AMD Opteron processor (Courtesy of AMD Corporation**)**

The memory interface supports PC2100, PC2400, and PC2700 DDR memory (illustrated in figure 3) with up to eight registered DIMMS. Using current memory density, this gives a Hammer CPU 16 GB of RAM at 2.7 GBps, which is a arrangement when compared with 1.6 GBps for a single-channel Rambus or 3.2 GBps for a dual-channel Rambus.

The system has support for 40-bit physical and 48-bit virtual memory addressing, which translates to a system with capabilities of handling 1 terabyte of RAM and 256 terabytes of virtual memory. This is important due to the use of coherent memory, where the RAM attached to each processor is available to the other processors in the array. Latency will be much higher than local memory, but AMD says it will seem as though there is a DRAM page conflict on normal memory, meaning the data needed was thought to be in memory but it wasn't. Latency shrinks as the CPU speed increases. Also, the interprocessor HyperTransport connection will become faster, since it is an integrated component.

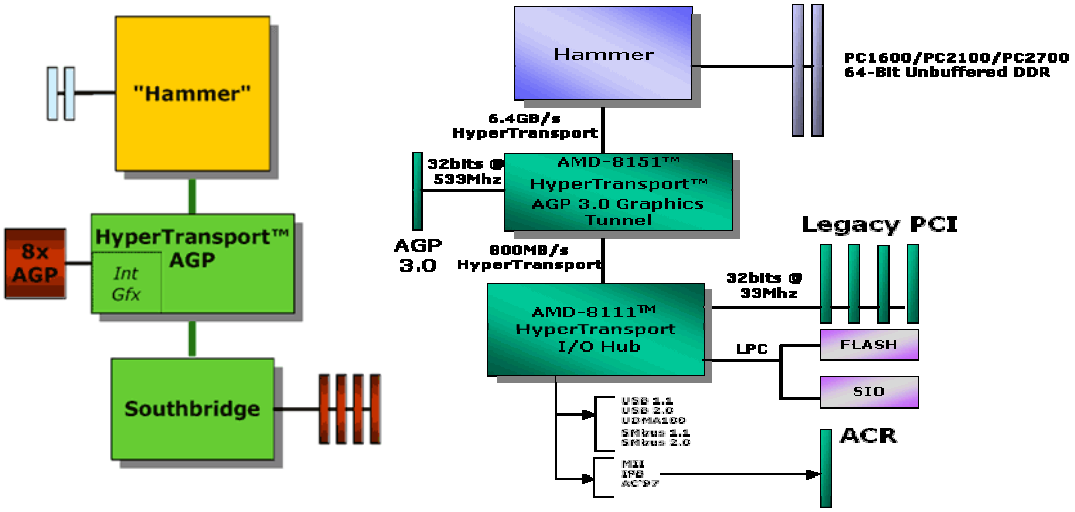

Fig. 2. AMD HyperTransport (Courtesy of AnandTech and X-bits Lab Corporations)

The figure above, on the left side shows that the AGP 8X controller is the only other chip that the chipset manufacturer has to provide outside the South Bridge. The AGP 8X controller connects to the Hammer processor via a HyperTransport link. On the right side the figure shows a system which is a result of a combination of different tunnels and controllers (AMD-8151 graphics AGP tunnel is an AGP bus controller; AMD-8111 input/output tunnel, unlike AMD-8151 and AMD-8131, only has one 8bit 400MB/sec HyperTransport bus controller).

The level 1 (L1) and level 2 (L2) caches (illustrated in fig. 3) have not been specified to date, as the processor has still to go into final silicon production. Some sources indicate that the L1 cache will be 64 KB, although whether this is the combined data and instruction cache or the size of each is not clear. The current Athlon has a 64-KB/64-KB L1 cache, which is likely to be the case on the Hammer.

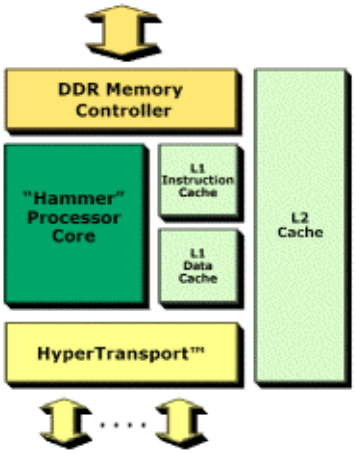

Fig. 3. Overview of the AMD Hammer CPU (Courtesy of AnandTech Corporation)

Between the L1 and L2 caches there is a trace lookup buffer (TLB). This device guesses which data the processor will need. Since this occurs independently of the normal schedulers and data management scheme, it increases the effective performance of the caches. L2 cache is expected to be 1 MB, four times more than the Athlon or P4. Larger L2 caches will reduce the number of page misses and avoid the additional latencies. Once data gets past the caches, it reaches three series of instruction decoders. These decoders take the rather bulky x86 instructions and break them into more efficient micro operations. In the

Hammer's case, these three decoders drop their instructions into an additional four schedulers: three feeding the integer arithmetic logic units (ALUs) and one far more complicated scheduler feeding the floating-point units (addition, multiplication, and miscellaneous functions that handle the multimedia extensions). Thus, for every cycle, the processor could perform four operations: three integer and one floating point. [1] [2]

2.1 Symmetric Multiprocessing

Hammer was designed for multiprocessor systems. The combination of integrated memory controller and bus interface turns it into a "glueless" processor. Other processors need "glue" logic to handle the relationship between processors, the memory controllers, and the system buses to ensure that all components get a turn. Giving each CPU the interface it needs eliminates conflicts. The limit of processors will be based on the complexity of the HyperTransport interface of each processor. SMP arrays expecting on more complex interaction between processors will need extra interprocessor bandwidth. In theory, all Hammer systems could be equipped with sufficient bandwidth to handle eight-way systems. In reality, only the SledgeHammer variants will have support for more than two processors. The figure below shows a four-way system with a bisection bandwidth of 12.8 GB/s. [3]

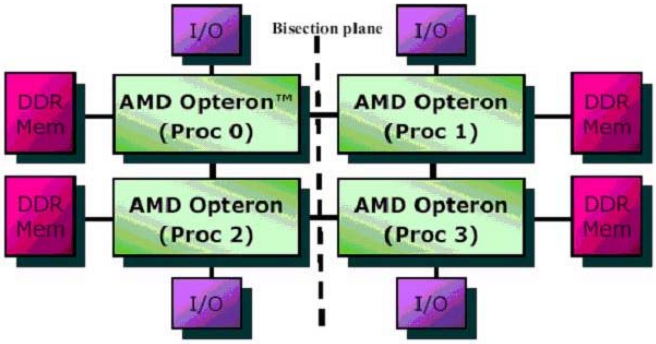

Fig. 4. Quad processor system topology (Courtesy of AnandTech Corporation)

3 Current 32-bit ISA Limitations

Naturally, given a 32-bit register, memory-addressing facilities are limited to four gigabytes. Precision and large data computations are also restricted by such architectures. This is most evident in CAD software and high-end servers. In addition a large portion of the x-86 ISA is obsolete. Modern software has no utility for them.

3.1 AMD's Approach

The line behind AMD's approach to 64-bit processor design is that the limitation of the 32-bit x-86 ISA is a matter of current architecture implementation. Additionally, implementing an all-new ISA breaks compatibility with the multitude of existing software. Therefore, backwards compatibility with 32-bit ISA software is a required feature of the next generation of its processors. So, a few new registers were added to the register set, and the existing ones were extended. As illustrated in the two figures below, the 8 general purpose registers from IA-32/16 are extended to 64-bit and the number of registers is increased from 8 to 16. The number of SSE/SSE2 registers is also increased from 8 to 16. And most important, the IP now has a 64-bit option.

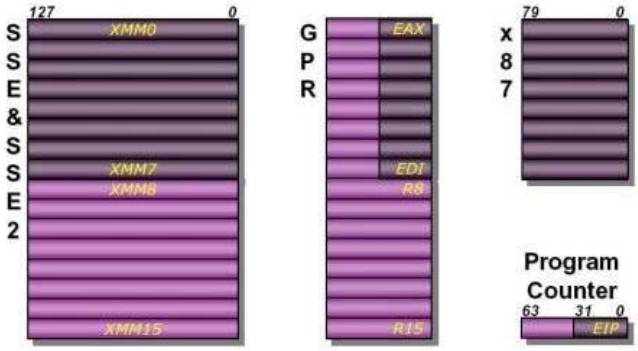

Fig. 5. X86-64 AMD programmer's model (Courtesy of AnandTech Corporation)

Fig. 6. Extension of EAX into RAX (Courtesy of AnandTech Corporation)

To support both 32bit and 64bit code and registers, the x86-64 architecture allows the processor to work in two modes: long mode with two sub-modes (64bit and compatibility modes) and legacy mode as illustrated in figure above. [4]

Mode		OS	Recompilation required	Default			
				Address length	Operand length	Additional registers	GPR size
Long Mode	*64bit*	64bit	Yes	64	32	Yes	64
	Compatibility mode		No	64		No	32
Legacy Mode		32bit or 64bit	No	32	32	No	32
				16	16		

Fig. 7. AMD's operation modes (Courtesy of X-bits lab Corporation)

3.2 Intel's Approach

The line behind Intel's approach to 64-bit processor design is that the hardware currently required to implement the x-86 ISA at a reasonable speed is far more complex than necessary. Similar to the previous move to RISC architectures, Intel hopes to establish the EPIC architecture paradigm. This solution is designed specifically to address simplicity of hardware design and to exploit parallelism. EPIC builds on the parallelism of RISC and newer-generation CISC by making of the use of parallel execution a necessary function. Modern code is complex, with many branches and conditions that a RISC or CISC chip generally struggles to manage. [5]

4 Dealing with Common Architecture Design Issues

Developers over the years have increased performance of their architectures in many ways. However, there are two issues that still have a large impact upon performance, even in today's processors. These issues are the problem of branch optimization and memory latency. When a CPU encounters a branch instruction, it has a number of possible paths that it can follow. The processor must stop and wait for the branch to evaluate before continuing. This wastes an enormous amount of CPU time. The second problem, states that

CPU speeds will always increase at a much faster rate than memory speeds. Thus whenever time memory is accessed, the CPU must again waste CPU time waiting. Both Intel and AMD have come up with techniques to minimize these problems.

4.1 Branch Optimization

To optimize code execution when a branch instruction is found, the CPU must wait until the branch result is computed. This wastes time that in turn decreases performance. A number of methods to overcome this problem were used in the past. However current research has indicated more efficient ways to solve the problem.

AMD's 64-bit Branching Solution. AMD's approach to the branching problem uses the well-known branch prediction concept. That is, the CPU guesses which branch to take. One problem with this approach is that an incorrect "guess" forces the entire pipeline to be purged and time is wasted.

AMD has vastly improved the branch prediction unit of the Athlon in the Hammer. The branch target array has the same 2K entry limit and 12-entry return stack as the Athlon, but the unit itself has been improved. It has these branch selectors which are bits stored in the L1 cache that contain information about where branches in the code exist and what type of branches they are. These branch selectors also have an additional bit that can flag the branch as static thus allowing the processor to predict it statically. This helps prevent the global history counter, from becoming cluttered with unnecessary information since when the processor branched to a particular error code will not help predict any non-static branches in the code later on.

The final feature of the Hammer's branch prediction unit is a bit of logic called the Branch Target Address Calculator (BTAC) which is illustrated in figure below:

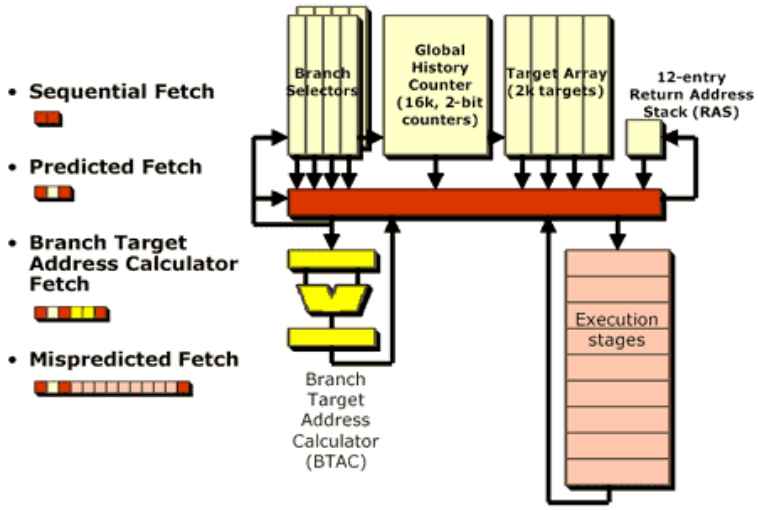

Fig. 8. Branch Target Address Calculator (Courtesy of AnandTech Corporation)

The Hammer isn't engineered in an entirely different manner; while it won't even attempt to extract the level of parallelism in the code that the Itanium does, what it will do is attempt to predict better the outcome of branches. In this case, the Hammer would calculate the direction a branch appears to be taking and use its Branch Target Address Calculator to actually calculate the branch. This little distraction only eats up around 5 clock cycles and improves dramatically the efficiency of the processor's ability to predict branches by removing some of the guesswork and actually calculating the direction and path of a branch. [3]

Intel's 64-bit Branching Solution.

Intel's approach to the branching problem proposes utilizing parallelism to execute all branches simultaneously, a technique they call predication. With predication, each block of code is marked and assigned to a predication register, which will eventually hold the results of the comparison. Once the comparisons can be evaluated, the predication registers are then examined to determine which branch was the correct one, and the results of the other branches are discarded. Moreover, Intel has worked closely with researchers in the area of compiler design and code-optimization. Compilers will be able to mark sections of code as explicitly parallel, and be able to support data to the CPU in 128-bit "safe" chunks.

It is important to notice what the Itanium does when faced with multiple conditional branches: it simultaneously evaluates various chunks of code, including both conditions of a branch, and at the end it chooses the "correct" data and discards what is useless, instead of predicting where the branch will take the CPU. One of the most elegant ways Intel had to deal with branch mispredict penalties is by the introduction of an execution trace cache; this cache stores instructions in their decoded form, in their order of execution, so that a branch mispredict will not result in another set of time consuming decoding steps. [5] [6]

4.2 Memory Latency

Memory latency is the cause of the large variance in CPU fetch and executes time. This is an issue in CPU design because the CPU is so much faster than memory. The move from 32 to 64-bit has only increased this gap. The CPU is now fed twice as much data, which it will process extremely fast, and spend the same amount of time waiting for more data.

AMD's 64-bit Latency Solution.

AMD also approaches this issue similarly with an increased pipeline (twelve stages). It is AMD assurance that the time spent to access the memory controller is critical to performance. Therefore, AMD decided to integrate the memory controller directly into the CPU. When the CPU executes a read from system memory the command is first sent over the FSB to the North Bridge of the chipset, which then hands it off to its integrated memory controller. These initial steps alone house a number of potential bottlenecks. Although very rare (since FSBs and memory buses are usually kept somewhat in sync), it is possible for a lack of FSB bandwidth to slow down this part of the memory read. Much more likely are the inefficiencies within the North Bridge and its memory controller which would add costly delays to the data retrieval from memory. When the memory controller has the read request, it is sent over the memory bus to the memory and once the data is found it is sent back to the memory controller. The memory controller takes the data, hands it off to the FSB interface within the North Bridge and back to the CPU it goes. [3]

Intel's 64-bit Latency Solution.

The basis for Intel's new architecture is a technique they call speculation (also called speculative loading in EPIC). With speculation, memory addresses are fetched long before they are needed, even in the case of a branch instruction. This capability brings with it the possibility of an exception occurring. Intel adds a small amount of additional hardware to ensure there is a proper exception handling mechanism. In general, a special register called the check register is checked after a speculative memory load. If the load caused an

exception, setting this register will indicate it, and thus the exception is handled. Speculation enables nearly all of the chip's real estate to be dedicated to task execution, whereas large areas of RISC and CISC chips are dedicated to housekeeping functions, such as managing execution sequences. However, this advantage comes with a price. An EPIC chip still needs execution management to control scheduling, which is now done by software in the compiler rather than on the chip itself. This places much more responsibility on the compiler writers. [5], [6]

5 Conclusion

The original Itanium has not exactly paved the way for its successor. However, Itanium 2 will still be starting from effectively ground zero in terms of software support and user base. Meanwhile, AMD's Opteron will slowly gain market share, as this market is less susceptible to performance than software support reputation, perceived reliability and the force of habit. And it remains to be seen how quickly the big database vendors (Oracle, Sybase, Microsoft) will produce fully 64-bit x86-64 versions for the Opteron. However, in the workstation market, the Opteron can be a very effective weapon. The Opteron's platform is very scalable and the HyperTransport is a very elegant way of interconnecting the ASIC's making motherboard very flexible and less expensive. Running a 64-bit version of Windows, the Opteron can offer 4 GB to each 32-bit process without any performance hit. Being significantly improved over the Athlon MP, it is expected that the Opteron performs exceptionally well in workstation applications, and these two advantages might increase AMD's popularity in the workstation market. In the longer term, this might encourage workstation ISVs to develop and launch x86-64 versions of their software. In order to counter the threat of Opteron, Intel's will most likely opt to push Deerfield in the high-end workstation market, while a Xeon version of Tejas could smooth the migration from 32-bit x86 to 64-bit IA-64. Nevertheless, there is an important Window of opportunity for AMD in 2003. [6], [7], [8]

References

[1] Fred Weber & CTO, Computation Group – AMD http://www.amd.com/us-en/assets/content_type/DownloadableAssets/Opteron_MPF_2002_print.pdf (Microprocessor Forum 2002)

[2] Omid Rhamat,Thomas Pabst, AMD's Opteron Comes Down Hard, http://www6.tomshardware.com/cpu/20020424/index.html (April 24, 2002).

[3] Anand Lal Shimpi, AMD's Hammer Architecture-Making Sense of It All, http://www.anandtech.com/cpu/showdoc.html?i=1546&p=1 (November 20, 2001).

[4] Xbitlabs – Special Hardware Infocenter, A Glance at the Future: AMD Hammer Processors and x86-64 Technology, http://www.xbitlabs.com/cpu/hammer-preview/ (August 14, 2002)

[5] Hannibal, A Preview of Intel's IA-64, http://www.arstechnia.com/cpu/lq99/ia-64-preview-1.html (November 20, 2001).

[6] Ron E. Curry, IA-64 Architecture Delivers New Levels of Performance and Scalability, http://www.intel.com/ (November 20, 2001).

[7] John G. Spooner, AMD Hammering at 64-bit Desktops, http://zdnet.com.com/2100-11-502578.html (November 20, 2001).

[8] Rick C. Hodgin, Intel's Itanium, http://www.geek.com/procspec/features/itanium (November 20, 2001).