The Linley Group

MICROPROCESSOR REPORT

www.MPRonline.com

THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

AMD'S BOBCAT SNARLS AT ATOM

Low-Power x86 Core Gives AMD Teeth in Mobile-PC Fight

By Tom R. Halfhill {8/30/10-02}

For two years, Intel's Atom processors have utterly dominated the low-power x86 market, winning designs in the vast majority of netbooks while gaining market share in other segments as well. Atom has almost totally eclipsed Via Technologies' Centaur processors, which pioneered the concept of a smaller and simpler x86. Meanwhile, AMD has been virtually AWOL. Athlon Neo runs much hotter than Atom, and AMD's other x86 processors are optimized for high performance in servers, desktops, and mainstream notebooks.

Now, AMD is clawing back. Its newest CPU core, code-named Bobcat, should beat Atom in single-thread performance at similar sub-watt power levels. As Figure 1 shows, Bobcat follows Atom's lead by discarding some of the complexity of post-Pentium designs in favor of a simpler dual-issue microarchitecture. Yet, like Atom, Bobcat retains enough performance to run heavyweight PC operating systems like Windows 7. AMD estimates that Bobcat will deliver 90% of the performance of today's mobile-PC processors in half the die area.

Bobcat goes one step beyond Atom: it's almost entirely synthesizable and is portable to multiple fabrication processes at independent foundries. Indeed, the first Bobcat implementation will be manufactured in a 40nm-G bulk-CMOS process at TSMC, not in the 45nm SOI (silicon-on-insulator) process in which other AMD x86 chips are fabricated by GlobalFoundries, AMD's fab spinoff. Bobcat's portability raises the intriguing possibility of market-specific SoCs, perhaps designed by companies other than AMD.

Moreover, Bobcat heralds another type of integration—one that AMD has promised since acquiring ATI in 2006. The first Bobcat chip, code-named Ontario, combines two Bobcat CPUs with an ATI graphics processor on the same die. It will be the first shipping product to emerge from AMD's long-delayed Fusion project. AMD calls the integrated CPU-GPU chip an accelerated processing unit (APU). Ontario is scheduled to ship in production volumes next quarter and should appear in systems early next year.

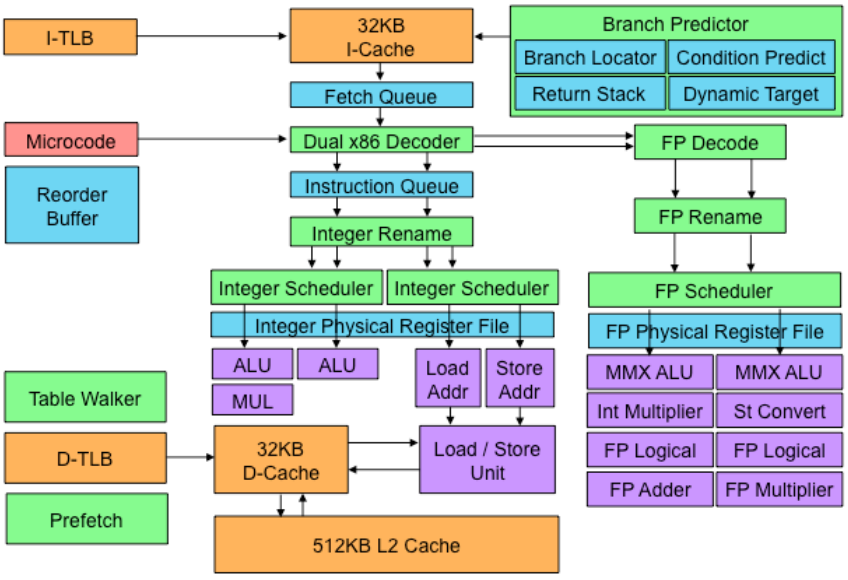

Figure 1. AMD Bobcat block diagram. In some respects, this dual-issue CPU core is the simplest AMD x86 microarchitecture since the mid-1990s. Yet it has many power-saving and performance features—such as out-of-order execution—not found in PC processors of that era. (Source: AMD)

(The first Fusion chip was supposed to be "Llano," a quad-core PC processor, but problems related to its new 32nm fabrication process have postponed production until 1H11.)

Initially, AMD plans to sell Bobcat chips for netbooks, notebooks, and some embedded applications. Unlike Intel, AMD isn't yet pursuing tablets and smartphones, two fast-growing product categories already dominated by ARM. Nor is AMD making much noise about consumer electronics, a market largely ruled by ARM, MIPS, and Power Architecture. At the other end of the spectrum, AMD isn't ready to promote Bobcat for low-power servers, another market segment with great potential. (See *MPR 6/21/10-01*, "New Processors Target Data Centers.")

To enter those wider markets, AMD must further improve Bobcat's power efficiency and prove its performance. Atom has a two-year head start and is reaching all those markets first. AMD, as usual, could offer an alternative after Intel opens the door.

Bobcat Veers from Bulldozer

As expected, Bobcat differs markedly from "Bulldozer," AMD's other new x86 core. (See *MPR 8/30/10-01*, "AMD Bulldozer Plows New Ground.") Bulldozer is a more powerful design that will replace the current K10 microarchitecture in AMD's Phenom-II PC processors and Opteron server processors. Whereas Bulldozer's basic building block is a dual-core cluster with shared resources, Bobcat is a traditional single-core CPU. And whereas Bulldozer issues four instructions per clock cycle—an improvement over the three instructions of current AMD processors—Bobcat pares back to dual issue.

Table 1 compares these CPUs to show the new breadth of AMD's x86 microarchitectures. In most respects, Bobcat is downsized to save power and die area. It does, however, make some improvements over today's K10 designs, such as more-flexible instruction reordering. It also has better branch prediction—even better than Bulldozer's.

	Bobcat	Bulldozer	Athlon Neo
Target Markets	Notebooks, netbooks, embedded	PCs, servers	Low-end notebooks
Building Block	CPU core	Dual-CPU cluster	CPU core
Instr Decoding	2 x86 per cycle	4 x86 per cycle	3 x86 per cycle
Instruction Issue	2 per cycle	4 per cycle	3 per cycle
Instruction Ordering	Full reordering	Full reordering	Limited reordering (no load/store)
Threads	1 per CPU	1 per CPU	1 per CPU
ALU Pipeline	16 stages	Not disclosed	12 stages
L1 Cache (I + D)	32KB + 32KB with parity	32KB + 16KB per CPU with parity	64KB + 64KB
L2 Cache	512KB, 16-way, ECC, half speed	Variable size, shared, ECC, full speed	512KB, ECC, full speed
FPU / SSE	2x 64 bits per CPU, 4 SP ops per cycle	2x 128 bits per cluster, 16 SP ops per cycle, FMAC instruction	2x 128 bits per CPU, 8 SP ops per cycle
Instruction TLB (page sizes)	512 (4KB) + 8 (2MB)	L1: 72 (mixed size) L2: 512 (4KB)	L1: 32 (4KB) + 16 (2MB) L2: 512 (4KB)
Data TLB (page sizes)	L1: 40 (4KB) + 8 (2MB) L2: 512 (4KB) + 64 (2MB)	L1: 32 (4KB–1GB) L2: 1,024 (4KB–1GB)	L1: 48 (4KB) + 48 (2MB) + 8 (1GB) L2: 512 (4KB) + 128 (2MB)
x86 Extensions	SSE1–SSE4a, x86-64, AMD-V	SSE1–SSE4.2, AVX, FMAC-4, x86-64, AMD-V	SSE1–SSE3, x86-64, AMD-V
Lowest Pwr State	C6	C6	C1E
Foundry Portable?	Yes	No	No
IC Process	40nm-G CMOS TSMC	32nm HKMG GlobalFoundries	65nm SOI GlobalFoundries
Production	4Q10 (est)	2011 (est)	1Q09

Table 1. Key parameters for AMD's Bobcat, Bulldozer, and Athlon Neo. All are 64-bit x86-compatible CPU cores. The most notable difference is that Bulldozer is an indivisible dual-core cluster that rules out single-core implementations. The first Bobcat chip (code-named Ontario) will have two CPU cores, but some later chips will save power by having only one CPU. (Source: AMD)

Like all modern x86 processors, Bobcat translates complex x86 instructions into simpler RISC-like operations for streamlined execution. It starts by fetching up to 32 bytes per clock cycle from the 32KB instruction cache, which is parity protected, has 64-byte lines, and is two-way set associative. An x86 instruction can vary in length from 8 bits to 120 bits (or even longer in weird cases) but averages about 20 bits. By fetching up to 32 bytes per cycle, Bobcat is almost always sure to grab two or more instructions—the minimum required to keep the dual integer pipelines busy.

In a departure from other AMD processors, Bobcat doesn't mark the instruction boundaries when prefetching code into the instruction cache. Instead, the processor scans the instruction stream while fetching from the cache and finds the boundaries on the fly. The instruction decoder scans up to 22 bytes per clock cycle and decodes one or two x86 instructions per cycle. On-the-fly scanning is a little slower than identifying the boundaries during prefetch, but it saves a little room in the cache by omitting the marker bits.

On average, Bobcat's decoder converts 89% of x86 instructions into a single RISC-like micro-op. These are the simpler x86 instructions that modern compilers favor. About 10% of x86 instructions are more complex and require two micro-ops. The remaining 1% of x86 instructions are pathological cases that require lengthy microcode execution, so Bobcat diverts them to the microcode engine that remains a legacy of all x86 processors. (AMD derived these percentages from real-world code streams; they don't represent percentages of the x86 instruction set.)

A Better Crystal Ball

Decoded micro-ops enter a queue to await scheduling; AMD hasn't disclosed the size of the queue. A reorder buffer (size also undisclosed) shuffles the instructions so the schedulers can dispatch two at a time to the function units. As usual with out-of-order processors, Bobcat has a pool of registers larger than the architectural register set, and instructions are temporarily assigned registers from this pool. Later, when the instruction retires, the processor renames the temporary registers to represent architectural (programmer-visible) registers. AMD hasn't disclosed the size of the rename-register pool.

In addition to supporting out-of-order execution, Bobcat's implementation of register renaming saves power, too. Instead of copying data from one physical register to another, the processor can simply change a pointer to rename the source register as the destination register. Bobcat's register pool maintains a mapping table for these pointers.

Bobcat's branch-prediction logic can predict the outcomes of two branches per clock cycle. It remembers the memory addresses of branch instructions it encounters and predicts return addresses before popping them from the stack. It also predicts addresses for register-indirect branches. Overall, AMD says Bobcat has a better crystal ball than either Bulldozer or existing AMD processors.

AMD improved the accuracy of Bobcat's branch predictor because the power-consumption penalty for wrong guesses is even worse than the performance penalty. The processor must flush the pipeline of partially completed instructions following the mispredicted branch and restart execution from that point, wasting all the power expended on them. In Bobcat's rather deep 16-stage pipeline, shown in Figure 2, the misprediction penalty is 13 clock cycles. The penalty is much worse if the rollback triggers a cache miss.

To save power, Bobcat uses clock gating in some elements of the branch predictor, activating the circuits only after encountering certain types of branches that require those elements. Further details remain undisclosed. AMD says the predictor uses a pattern-matching technique akin to neural nets, which implies an adaptive algorithm that learns by profiling the code. Simple predictors with branch history tables are adaptive, too, so don't read too much into the neural-net allusion.

Better Load/Store Reordering

For integer instructions, Bobcat has two dual-ported instruction schedulers, each capable of dispatching two micro-ops per clock cycle to the processor's function units. One scheduler feeds the dual ALUs, and the other feeds a load-address unit and a store-address unit.

Bobcat improves on existing AMD x86 processors by issuing load/store instructions out of order with the same flexibility as other instructions. Of course, true data dependencies always limit this potential, but Bobcat eliminates some false or resource-bound dependencies that restrict other AMD designs. Loads can bypass stores and other loads, and stores can bypass loads. A "hazard predictor" anticipates and avoids data dependencies. The pipeline can forward stores to later stages so their results are available to instructions executing out of order.

Floating-point instructions have their own scheduler, which can issue two operations per cycle to a pair of 64-bit

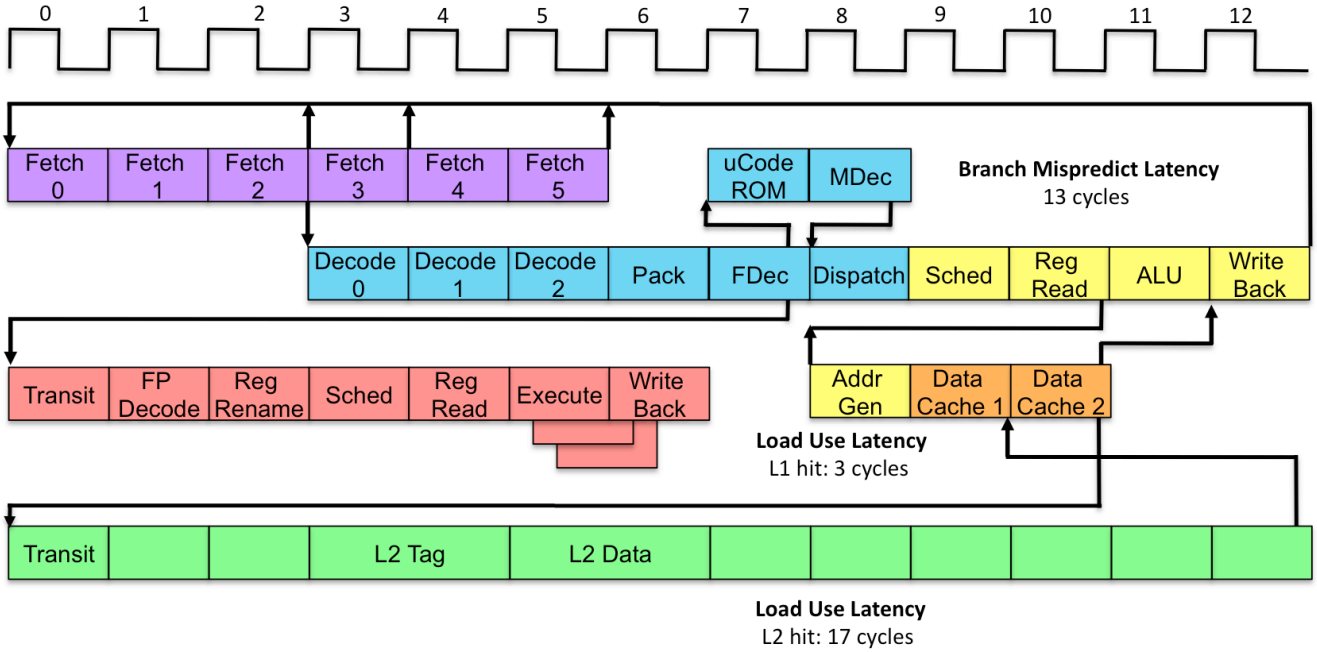

Figure 2. Bobcat pipeline diagram. The basic integer pipeline is 16 stages deep, including nine stages dedicated to fetching x86 instructions and decoding them into simpler RISC-like micro-ops. Today's AMD processors have 12-stage integer pipelines. Bobcat's additional stages support its more flexible instruction reordering and scheduling. (Source: AMD)

execution pipelines. (These pipelines can also handle 80-bit extended-precision operations, which are comparatively rare.) One pipeline has a multiplier, and the other has an adder; each is capable of executing one 64-bit operation or two 32-bit operations per cycle. Both pipelines can execute logical instructions, MMX instructions, and SSE1–SSE4a instructions. Bobcat lacks, however, the latest AVX extensions to the x86 architecture and the fused multiply-accumulate (FMAC) instructions in Bulldozer.

To save power, the FPU dedicates physical registers to floating-point, MMX, and SSE instructions instead of sharing the x87 FPU stack among them. Although additional registers require more logic, they save power by eliminating clumsy stack operations. Similarly, Bobcat saves power by using nonshifting queues throughout the pipelines—adjustable pointers allow random access to data in the queues, eliminating costly shuffles.

Tag, You're Hit

Completed instructions write their results back to the rename-register pool and L1 data cache. Bobcat can retire two instructions per cycle, even if they are complex x86 instructions. The data cache is 32KB with 64-byte lines and is eight-way set associative. Like the instruction cache, it's parity protected. To save power, Bobcat's cache arrays are clock gated. Before reading the cache, the processor checks the data tags and won't activate the arrays unless there's a hit.

Bobcat's 32KB L1 caches may seem small for a modern x86 processor, but they're not out of line with other low-power x86 CPUs. The load-use latency for Bobcat's L1 cache is three clock cycles—the same as current AMD x86 processors. AMD says that enlarging the L1 caches would insert too much additional logic and wiring into critical paths, making it impossible to maintain a three-cycle latency. To some extent, instruction reordering—especially Bobcat's flexibility to bypass loads and stores—mitigates the latency of L1 cache misses.

The L2 cache is 512KB with 64-byte lines and is 16-way set associative. It's ECC protected and saves power by running at half the CPU clock frequency. Consequently, hits on the L2 cache have a load-use latency of 17 clock cycles compared with about 11 cycles for AMD processors with full-speed L2 caches. In another power-saving measure, Bobcat's I/O bus unit supports only two outstanding loads and eight outstanding stores. Other AMD processors support eight outstanding load/store transactions and two outstanding instruction fetches. To partially compensate, the bus unit has buffers for evicted data, line fills, and combined writes, but AMD isn't disclosing the buffer sizes.

Bobcat maintains separate translation lookaside buffers (TLBs) for instructions and data, including two data TLBs. Overall, Bobcat has smaller TLBs than existing AMD processors and lacks their two-level instruction TLB, but its resources are adequate for systems that will likely have less memory than mainstream PCs and servers.

Bobcat Should Beat Atom

Only two other x86 CPUs are in this low-power class: Intel's Atom and Via's Nano (code-named Isaiah). Atom is by far the market leader, reaping about 99% of the volume. AMD has a chance to take some market share away from Intel, because Bobcat should exceed Atom's single-thread performance in instructions per cycle, judging from their microarchitectures.

As Table 2 shows, Bobcat differs from Atom in two important ways: it dynamically reorders instructions, and it's single threaded. Atom always executes instructions in their original program order and can manage two hardware threads, although some models disable Hyper-Threading to save power (or for marketing reasons). Otherwise, Bobcat and Atom are similar designs with 16-stage integer pipelines, dual-issue superscalar execution, and like-sized caches. (See *MPR 4/7/08-01*, "Intel's Tiny Atom.")

AMD is making an interesting tradeoff between instruction reordering and threading. Both techniques improve throughput while adding unwanted complexity to a processor, so the performance gain must outweigh the additional overhead. Both techniques can improve throughput by approximately the same amount—we estimate 20–30%, although the gains depend greatly on the application code.

Bobcat favors single-thread performance. Out-of-order execution boosts instruction-level parallelism within a single thread, whereas multithreading boosts data- or task-level parallelism among two or more threads. Although finding enough work to keep an additional thread busy isn't difficult for today's multitasking PCs, reordering improves throughput with virtually all code, all the time. Also, reordering can scale better as the number of CPUs increases, at least in client systems.

In theory, multithreading is better at hiding the long latencies of memory operations, because the processor can suspend a stalled thread and switch to a different instruction stream. Although Bobcat can sometimes bypass a stalled load by executing other instructions out of order, the window for reordering instructions within a single thread is relatively small, so Bobcat can't postpone a stalled load for as long as Atom can. Also, Bobcat's half-speed L2 cache worsens the load-use latency.

There may be another reason why Bobcat favors instruction reordering over multithreading. AMD has never designed a multithreaded processor, although Bulldozer's dual-CPU cluster with shared resources comes close. AMD has much more experience with reordering, going back to the K5 microarchitecture of the mid-1990s. Although we're sure multithreading isn't beyond AMD's talents, the Bobcat team was probably more comfortable using a mature technique that has been thoroughly field tested over the past 15 years. Existing compilers are well adapted to instruction reordering, too.

Another performance factor is clock frequency. Unfortunately for our analysis, AMD hasn't yet announced

clock speeds or other details about Ontario chips. Having similar 16-stage pipelines and comparable 40nm/45nm fabrication, Bobcat and Atom should be able to reach similar clock frequencies. Ontario will be manufactured in TSMC's 40nm-G bulk-CMOS process, whereas Intel manufactures Atom in a proprietary 45nm high-*k* metal-gate process. Atom currently peaks at 2.13GHz, although most netbook chips run at 1.6GHz to 1.8GHz. (Atom can probably surpass 2.13GHz, because Via's Nano is nearly as fast in a 65nm process.)

Nano Outguns Bobcat and Atom

Via's Nano can execute three x86 instructions per cycle, outgunning the dual-issue execution of both Bobcat and Atom. Via hasn't disclosed the depth of Nano's pipeline, but it's about 16 stages—the same as Bobcat and Atom. Like Bobcat, Nano is a single-threaded machine that executes instructions out of order. Nano's three-way issue gives Via a single-thread throughput advantage over both rivals. (See *MPR 3/10/08-01*, "Via's Speedy Isaiah.")

This comparison assumes equal clock speeds. Although Nano is restrained by older 65nm CMOS technology, Via says the latest Nano 3000-series chips can reach 2.0GHz, nearly matching Atom's top speed of 2.13GHz in 45nm technology. Most Nano processors run at 1.0GHz to 1.8GHz, as do most Atom processors. Via has taped out a dual-core Nano in 40nm, but the company hasn't decided whether to make a single-core Nano in that process. Via says the dual-core 40nm chip will use about the same amount of power as the single-core 65nm chip.

According to Via's benchmarking, a 1.3GHz Nano 3000 is 43% faster than a 1.6GHz Atom N270 on PCMark 2005 v120. On 3DMark 2006, Via claims a 51% advantage. EEMBC's CoreMark yields different results. Third-party testers report that an 800MHz Nano L3050 scored 2,305 (2.9 CoreMarks per megahertz), a 1.0GHz Atom N450 scored 2,806 (2.8 CoreMarks per megahertz), and a 1.68GHz Atom N280 scored 5,353 (3.2 CoreMarks per megahertz). These third-party CoreMark tests have not been certified by EEMBC, however. (See *MPR 6/8/09-01*, "EEMBC's Dhrystone Killer.")

Power consumption is no contest. The Nano 3000 ranges from 5W at 1.0GHz to 25W at 1.8GHz (maximum thermal design power, or TDP). Intel quotes 2.5W TDP for the 1.6GHz Atom N270, a chip made virtually obsolete by Intel's lower-power "Pine Trail" Atom processors. So even the first-generation Atom has better performance per watt than Nano.

Nano is simply a larger design. In addition to three-issue superscalar execution, it has larger L1 caches than both Bobcat and Atom and a larger L2 cache than Bobcat. A few Atom processors match Nano's 1MB L2 cache, but most have only 512KB. Nano's standout feature is a beefier FPU with 128-bit data paths. It's capable of executing eight single-precision operations per cycle—twice as many as its rivals. Nano also has some proprietary security extensions (PadLock and Via Secure Mode), but they are more useful in embedded systems than in mainstream PCs, where software support for the extensions is rare.

Via emphasizes pin compatibility among its processors—an often overlooked feature that gets less attention from AMD and Intel. All Centaur processors sold for the past eight years are pin compatible, including the new dual-core Nano that recently taped out. Interestingly, both Ontario and Nano are manufactured by TSMC—a first for AMD x86 processors. With the dual-core Nano and dual-core Ontario both fabricated in the same 40nm process at the same foundry, direct comparisons will be easier, although

	AMD Bobcat	Intel Atom	Via Nano / Isaiah
Instruction Decoding	2 x86 per cycle	2 x86 per cycle	3 x86 per cycle
Instruction Issue	2 per cycle	2 per cycle	3 per cycle
Instruction Ordering	Out of order	In order	Out of order
Threading	1 thread	2 threads	1 thread
ALU Pipeline	16 stages	16 stages	~16 stages
L1 Cache (I + D)	32KB + 32KB with parity	36KB + 24KB	64KB + 64KB
L2 Cache	512KB, 16-way, ECC, half speed	512KB, 2- to 8-way	1MB, 32-way*, ECC
FPU / SSE	1x 64 bits 4 SP ops per cycle	2x 64 bits 4 SP ops per cycle	2x 128 bits 8 SP ops per cycle
Instr TLB Entries (page sizes)	512 entries (4KB) + 8 entries (2MB)	Unified 64-entry TLB + 16-entry micro-TLB per thread	192 entries*
Data TLB Entries (page sizes)	L1: 40 (4KB) + 8 (2MB) L2: 512 (4KB) + 64 (2MB)	Unified TLB	192 entries*
x86 Extensions	SSE1–SSE4a, x86-64, AMD-V	SSE1–SSE3, x86-64	SSE1–SSE4*, x86-64, virtualization, Via PadLock, Via Secure Mode
Lowest Power State	C6	C6	C6
Foundry Portable?	Yes	No	Yes
IC Process	40nm-G CMOS TSMC	45nm HKMG Intel	65nm CMOS TSMC
Production	4Q10 (est)	2Q08	3Q08

Table 2. Comparing AMD's Bobcat with Intel's Atom and Via Technologies' Nano. Ironically, given Via's history of emphasizing simple microarchitectures, Nano is the widest superscalar design in this group. It also has a more powerful FPU. This table omits power consumption because AMD has yet to release details of Ontario, the first Bobcat-based chip. *Nano 3000-series features; a page-directory cache supplements the TLBs. (Source: vendors)

Price & Availability

Two Bobcat CPU cores will debut in "Ontario," the code-name for an integrated CPU-GPU Fusion chip, which AMD calls an accelerated processor unit (APU). Ontario is scheduled to ship in production volumes in the fourth quarter and appear in systems early next year. AMD hasn't announced clock speeds, power consumption, or pricing. For more information about Bobcat, visit *http://budurl.com/HotChipsPressKit*.

Nano doesn't include a GPU. These comparisons must wait, however, until both companies release clock-speed and power-consumption specifications for their new processors.

Don't Forget the GPU

Ontario's integrated CPUs, GPU, and north-bridge functions will make platform-level comparisons of power consumption more important than ever. AMD hasn't disclosed Ontario's TDP, but we estimate it's less than 10W. If our estimate is close to accurate, the dual-CPU Ontario will be a major leap beyond the single-CPU Athlon Neo, which lacks a GPU and north-bridge functions but still has a TDP of 15W at 1.6GHz. Ontario still won't match Atom's platform-level power consumption, however, and its TDP will probably limit it to systems with active cooling.

Another important variable is graphics performance. AMD is deriving Ontario's integrated GPU from an ATI 5000-series "Evergreen" discrete GPU, so it will probably outperform the pedestrian GPUs in today's Atom and Nano chipsets. Superior graphics performance is more visible to users than slimmer advantages in general-purpose code throughput.

Intel isn't standing still in this regard. Reportedly, Cedar Trail will double the performance of the GMA3150 graphics processor in today's "Pineview" chipset. Nevertheless, we expect graphics to be a strong point of Ontario.

Nano has graphics-integrated chipsets as well. Via's VX900, which was introduced this year, has 2D graphics, 3D graphics (DirectX 9.0), high-fidelity audio, a DDR3 memory controller, and HD video decoding for Blu-ray, among other features. (A newer Via chipset, the VN1000, has additional features and supports DirectX 10.1, but it's intended for small desktop PCs.)

AMD says Ontario will support DirectX 11, which includes Microsoft's new DirectCompute API. DirectCompute lets programmers use the GPU for data-parallel processing in nongraphics applications. Ontario also supports two additional APIs for parallel processing—ATI Stream v2.2 and OpenCL 1.1. (See *MPR 12/22/08-01*, "AMD's Stream Becomes a River," and the sidebar, "OpenCL Tries to Standardize Parallel Programming.")

As Nvidia has also shown with its Cuda technology, offloading data-parallel tasks to the GPU can improve performance far beyond the reach of a general-purpose CPU. Of course, software developers must modify their code to use these APIs. At the Computex show in Taiwan last June, AMD announced the Fusion Fund, which subsidizes developers to write Fusion software. AMD also demonstrated two operational Fusion processors at Computex: Ontario and Llano (a quad-core PC processor based on the K10 microarchitecture).

Opening the Door for SoCs

Perhaps the most intriguing aspect of Bobcat is its potential in SoCs. The CPU core is fully synthesizable and portable to different fabrication processes at different foundries. Outsourcing production to TSMC makes sense, even after AMD's spinoff of GlobalFoundries (GF), because the GF 45nm SOI process is optimized for high performance, not low power. TSMC manufactures numerous low-power embedded processors in its 40nm G and LP processes.

Bobcat is the most portable x86 CPU yet created (not counting synthesizable versions of ancient x86 processors, like the 80186). Although a synthesized processor sacrifices some performance compared with a fully custom design, AMD says Bobcat isn't wholly the product of an RTL compiler. Engineers tuned the critical paths for speed and power efficiency, often increasing the number of high-threshold transistors to reduce static current leakage. These techniques are common when designing high-performance embedded processors.

Atom is largely synthesized, too, although for different reasons. As we reported in our previously cited 2008 article, 91% of Atom's CPU core comprises standard cells, augmented with some manual optimizations. Intel says the core's only fully custom digital block is the microsequencer engine that decodes rare or complex x86 instructions. Intel synthesized Atom to accelerate the project and make the CPU easier to integrate in market-specific SoCs for communications, networking, and consumer electronics. Intel plans to design, manufacture, and sell most of those SoCs.

Last year, Intel announced an alliance with TSMC to design and manufacture Atom-based SoCs for third parties. This is the closest Intel has come to licensing the x86 since the early 1980s, when IBM convinced Intel to license AMD as a second source for processors in IBM PCs. But third parties weren't interested in the 45nm Atom, and Intel hasn't offered the 32nm design, so the Intel-TSMC initiative appears to be dead. (See *MPR 3/30/09-01*, "Intel Will Customize Atom.")

Now, with Bobcat, it's possible that AMD could pursue similar ventures. AMD may be too busy with PC and server processors to compete directly with SoCs from Intel and other semiconductor companies, but it could license Bobcat to third parties for SoC designs. The x86 is underrepresented in the embedded market, mainly because Intel won't license CPU cores or the architecture as ARM and other intellectual-property vendors do. Bobcat could fill that gap.

The only hitch might be AMD's all-important x86 license from Intel. Under the terms of that license, which are confidential, AMD may be forbidden to sublicense the x86 to other companies—especially if the chips would be manufactured at a foundry in which AMD has no substantial stake, such as TSMC. (Intel has said that fab ownership is a condition of AMD's x86 license; AMD retains a minority stake in GlobalFoundries.) Perhaps AMD could circumvent any licensing restrictions by selling chips to the SoC developers for resale.

Another possibility is that Intel's pending settlement with the Federal Trade Commission will change the rules. According to *EDN*, the FTC will require Intel to modify its patent licenses to give AMD (as well as Via and Nvidia) more freedom to enter joint ventures or merge with other companies. (The FTC action has spawned new rumors that Nvidia will acquire Via.)

Anything that moves Bobcat or Nano into independently designed SoCs would overcome a major drawback of the x86 in its growing battle with ARM. Without a licensable x86 core, third parties can't design the unique SoCs that some products need for differentiation (Apple's ARM-compatible A4 chip in the iPhone 4 and iPad is a prime example). Given Intel's reluctance to loosen its reins on the x86, a wider role for Bobcat could radically alter the embedded-processor market.

Bobcat Puts AMD in the Fight

Bobcat brings a much-needed low-power x86 design to AMD's product line. The clear trend in client computing is toward smaller, mobile systems. AMD's existing CPUs are adequate for desktop-replacement laptops and mainstream notebooks, but they're not power-efficient enough for the new breeds of thin-and-light notebooks and diminutive netbooks. Since Atom's debut in 2008, Intel has virtually monopolized these market segments. Via has some prominent customers (including Hewlett-Packard, IBM, Lenovo, and Samsung), but only in low volumes.

Bobcat puts AMD back into the fight. It is AMD's first true low-power x86 processor since Geode—a weak 1990s design acquired from National Semiconductor, which derived it from the Cyrix MediaGX chip of the mid-1990s. Bobcat is very competitive in its class, being capable of beating Atom's instructions-per-cycle performance on single threads. Thanks to the latest x86 virtualization extensions and error-protected caches, Bobcat is suitable for low-power servers as well.

By pairing two Bobcat CPUs with an ATI GPU in Ontario, AMD is introducing a complete low-power platform, not just a processor. Atom's initial weakness was its system chipset, which consumed much more power than the CPU while delivering lackluster graphics performance. Nvidia exploited this weakness by selling Atom-compatible chipsets with integrated graphics that surpassed Intel's offering. Pine Trail improved the Atom platform's power efficiency, and Cedar Trail will do even better. Bobcat will hit the market as a power-efficient and highly integrated platform from the start, but delays in the Fusion project have greatly reduced the advantage that AMD might have enjoyed.

Because AMD is starting with virtually zero market share in the netbook segment, it seems like Bobcat can hardly lose. Any slice of the pie is larger than no slice. But Via's experience urges caution. Despite seeding this market with its groundbreaking C-series processors and then delivering the sophisticated Nano, Via has gleaned minuscule market share while Intel reaps the bountiful harvest. Bobcat looks more competitive than Nano, and we think it could gain a 10–20% share of the netbook market. In other words, Bobcat could win the same amount of market share that AMD historically captures when fielding a product that's competitive with Intel. ♦