Northbridge Architecture of AMD's Griffin Microprocessor Family

Demands for mobility and portability have driven microprocessor and system manufacturers to innovate in the areas of power and thermal management and efficiency. Cost and space, meanwhile, have driven manufacturers toward unified memory architectures for CPU and graphics, further complicating power management. AMD's Griffin microprocessor, optimized for mobile computing platforms, introduces several northbridge innovations to manage power without diminishing performance.

Jonathan Owen

Maurice Steinman

AMD

●●●●●● Users buy computers, not processors, and their expectations pertain to the final product: the system. Nonetheless, the processor has a big impact on system characteristics. Over time, customers want systems to become cheaper, smaller, and more reliable, and have longer battery life. Users expect these characteristics to improve without adversely affecting performance or slowing the rate of performance increases: Baseline peak performance is a given. However, for many mobile workloads (word processing, e-mail, and so on), running the system in a peak-performance state is wasteful and burns power without improving the user experience. The goal is to operate in a peak-performance state when it's useful, and in a low-power state when it's not. This flexibility is the key to supporting both high performance and low power in the same processor.

AMD's next-generation mobile microprocessor, codenamed Griffin (see Figure 1), meets these challenges thanks to several innovations to the Griffin northbridge that balance performance and power.

Architectural challenges for northbridges in mobile platforms

The northbridge in x86 systems is the component that connects the CPU cores to memory and I/O devices, and maintains system memory coherency among requesters. AMD processors integrate the northbridge and memory controller into the processor chip, lowering CPU latency to memory and allowing higher CPU performance.

Unified memory architecture (UMA) is a standard way to reduce the cost of graphics subsystems. In a UMA platform, the graphics chipset does not have dedicated DRAM for rendering or frame buffer memory. Instead, it accesses system memory through the northbridge for those purposes. This leads to several architectural challenges:

- Graphics must have a high-bandwidth, low-latency link through the northbridge to memory. The challenge is managing the power of the high-performance I/O link between the graphics chipset and the processor.

- Because the memory controller is shared between the CPU and the graphics unit, the northbridge must arbitrate between them, and their access characteristics are quite different.
- The system must periodically refresh the display. This involves retrieving the screen contents from system memory over the high-speed link, and display refresh must continue to occur even when the CPU is idle. Consequently, the graphics path to memory must remain active, which interferes with placing the processor into a low-power state.

In the Griffin architecture, the northbridge balances the performance demands of the CPU cores and graphics with the energy expended to accomplish these tasks. The underlying philosophy is to expend energy in a manner that is commensurate with the performance delivered, and reduce energy consumption on non-performance-sensitive tasks. Compared to current-generation AMD mobile processors, Griffin includes several innovations to address these power-performance challenges:

- Griffin is the first mobile processor family to implement the HyperTransport 3 link protocol.[1] This is a high-bandwidth, low-latency interconnect, with power management features to reduce power during idle or low-load conditions.
- Griffin implements a high-performance memory controller to provide sufficient bandwidth for both graphics and CPU accesses, and provides a dedicated internal virtual channel between the HyperTransport interface and the memory controller to meet the unique performance requirements of display-refresh traffic.
- Griffin provides multiple power planes to manage the power and performance of the CPU cores independently of the memory system demands made by graphics. Each CPU core has a dedicated plane; single-threaded workloads need not pay a power penalty
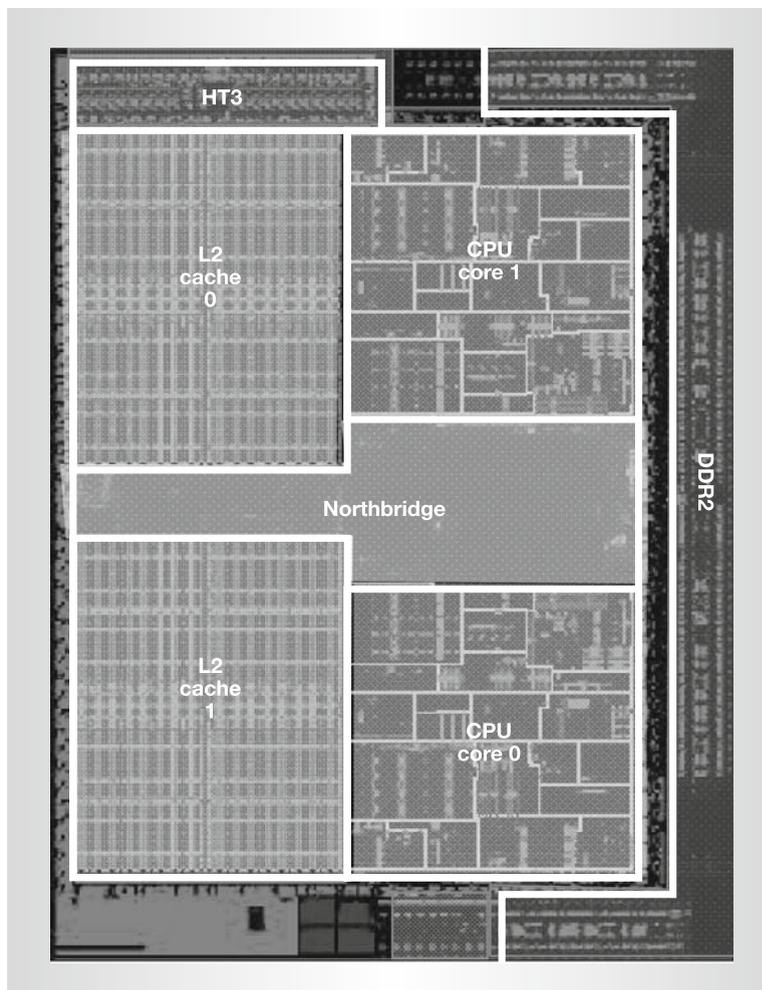

Figure 1. Die photo of Griffin, AMD's next-generation mobile processor, which links 64-bit multicore CPU technology with a new, mobile-optimized northbridge.

due to a powered-on idle core. The northbridge has its own power plane, giving graphics access to memory when all CPU cores are in a low-power state, and letting the northbridge run at a lower voltage than the CPU cores when they are in a high-power state.
- Support for fast frequency changes in the CPU cores and northbridge allow lightweight frequency adjustment to tune dynamic power. Reduction in operating frequency can also enable reduction in voltage to reduce power even further.

....
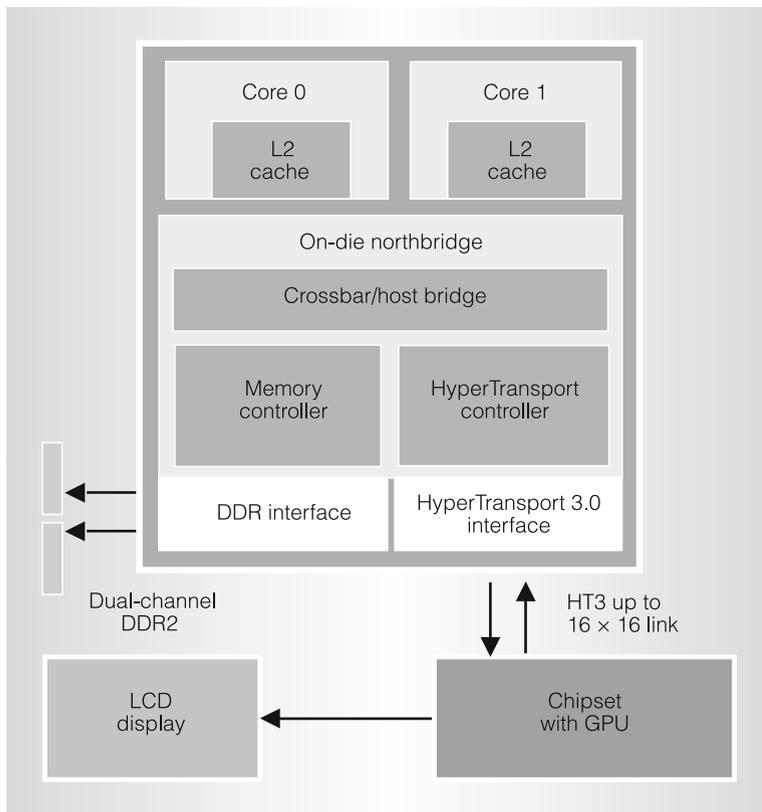
HOT CHIPS 19

Figure 2. Block diagram of Griffin.

Improved power efficiency obviously allows the processor to expend less energy for a given workload, but it also enables higher peak performance within a given power-thermal envelope at the system level. In the tightly constrained mobile environment, power and thermal considerations (rather than the maximum intrinsic speed of the CPU core) are frequently the limiting factor for performance.

Griffin overview

Griffin contains two high-performance 64-bit x86 CPU cores, each with a dedicated 1-Mbyte L2 cache, twice the size of the previous generation. The memory controller supports two channels of double-data-rate memory (DDR2-800), for up to 12.8 Gbytes/s of memory bandwidth and a total system memory size of up to 16 Gbytes. The HyperTransport 3 link is also a DDR bus, 16 bits wide in each direction and running at up to 2.2 GHz; this yields

4.4 Gbytes/s per wire, or a simultaneous peak bandwidth of 8.8 Gbytes/s to and from the chipset. Figure 1 shows a Griffin die photograph, with the major structures highlighted. Figure 2 shows a high-level block diagram.

Griffin memory controller

The two main memory request sources (CPU and graphics) have different access characteristics and requirements. For maximum performance, CPU traffic generally requires low latency. Graphics-rendering traffic is less sensitive to latency, but requires higher sustained bandwidth. Accordingly, AMD optimized the memory controller for minimum latency for CPU requests, and maximum bandwidth for graphics requests.

The memory controller tracks the page open/closed state of up to 16 banks per channel. It also implements two DRAM channels that can be grouped together into a single logical channel or ungrouped to operate independently, providing greater parallelism. Pages can be closed dynamically based on the bank access pattern. Requests are reordered to maximize page hits and minimize page conflicts. The memory controller tracks multiple priorities (both static and dynamic) to guarantee appropriate quality of service to the CPU, I/O, and graphics requesters.

The northbridge and memory controller are on a separate power plane and clock grid from the CPU cores. These components run at a lower frequency than the CPU cores, so they can also run at a lower voltage. Changes in the CPU voltage or frequency do not affect traffic to memory, so those can be freely optimized to match CPU load.

Griffin implements an advanced DRAM prefetching algorithm to reduce effective memory latency. The prefetcher is capable of tracking single-strided ($+n$, $+n$, $+n$) or double-strided ($+n$, $+m$, $+n$) patterns, with strides up to ± 4 cache lines, where a cache line is 64 bytes in size. Griffin can track prefetches in up to eight DRAM pages simultaneously. When an unopened page receives a request, Griffin allocates an entry to that page in the memory prefetch table

A → A + n → A + n + m → A + 2n + m → A + 2n + 2m

stride1   + n   + m   + n   + m
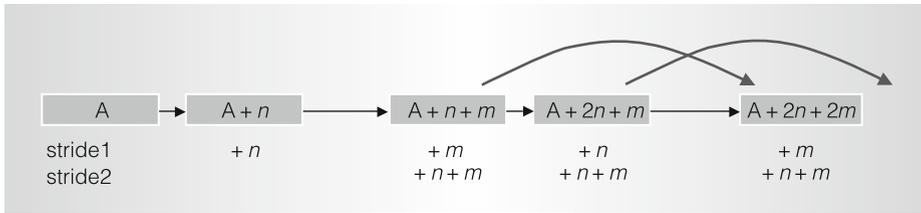stride2   + n + m   + n + m   + n + m

Figure 3. Prefetcher operating on a double-strided request stream.

(MPT). If the table is full, Griffin overwrites the least recently used entry.

When an open page receives a request, the address difference to the previous request (delta) is calculated and stored in the stride1 register. Subsequent requests not only continue to update the stride1 register with each delta, but also update the stride2 register with the sum of the previous two deltas. The stride2 register is then used to predict future request addresses, looking two addresses ahead. Figure 3 shows an example of a double-strided request pattern and the values stored in the stride registers when each request is received. Once both stride registers are loaded, prediction begins. When the prefetcher has made a sufficient number of correct predictions to establish confidence that it has identified valid strides, it begins issuing requests to further predicted addresses so data is already available when those requests are subsequently received, hiding the DRAM access latency.

AMD has characterized the performance benefit of the DRAM prefetcher using the company's internal benchmark suite for assessing CPU performance. The benchmark suite includes three classes:

- digital media—workloads such as photo and video editing;

- office productivity—word processing, spreadsheets, and e-mail; and
- games—PC games with rich video content.

Table 1 summarizes the benchmark results.

In a UMA system, display-refresh traffic to redraw the display is an ongoing load. Depending on the size of the system's display(s) and the number of surfaces being rendered, such traffic can consume a considerable amount of the available system memory bandwidth (multiple gigabytes per second). Because the display controller in the graphics unit contains some display buffering, there is some tolerance to latency, but the system must provide a minimum sustainable bandwidth at a maximum guaranteed latency to avoid display buffer underrun. To optimize overall memory and system performance, display-refresh traffic should be serviced at a rate that keeps the display buffer filled at a constant, safe level. This requires granting it some degree of priority when the buffer is draining, to prevent underrun in the presence of other traffic. However, delivering unnecessarily high priority to display traffic when the buffer is not in danger of underrunning creates an unneeded latency penalty to nondisplay traffic, such as CPU and

Workload group	Performance, DRAM prefetcher off (IPC*)	Performance, DRAM prefetcher on (IPC)	Performance increase (%)
Digital media	1.0268	1.0388	1.17
Office productivity	0.6072	0.6210	2.27
Games	0.6924	0.7049	1.81
Overall	0.7558	0.7690	1.75

Table 1. Performance benefit of DRAM prefetcher, benchmark results.

* IPC: instructions per cycle.

Table 2. Griffin cores' time spent in power states.

C-state	Core 0, time spent (%)	Core 1, time spent (%)
C0	4	1
C1	1	4
C1E (non-AltVID*)—*deep sleep*	10	10
C1E (AltVID*)—*deeper sleep*	85	85

* AltVID: core at minimum voltage value required for flip-flops to maintain state.

graphics rendering traffic and other I/O DMA activity.

The requirements of display-refresh traffic do not map well to the standard virtual channels afforded by the HyperTransport protocol. HyperTransport provides an *isochronous channel*, which is guaranteed to provide low latency because it always receives the highest priority through the northbridge and memory system. But, because it is designed for low bandwidth, using this channel for high-bandwidth display refresh would in general result in other requests being blocked for some period of time, adversely impacting system performance. Meanwhile, the display-refresh traffic would actually be experiencing lower latency than necessary, without any corresponding performance benefit.

The main HyperTransport channel is the *base channel*, which contains all other request traffic. This channel does not provide a latency guarantee, and display-refresh requests would be interleaved with other requests from I/O and the CPU with no provision for enforcing their completion in a bounded time.

Griffin meets the requirements of display-refresh traffic by separating that traffic into a dedicated virtual channel within the northbridge and the memory controller. That way, display-refresh requests can queue separately from other traffic, with their priorities through the northbridge and memory controller adjusted dynamically. The chipset can guarantee that these requests are able to reach the northbridge either by placing them in the isochronous channel on the HyperTransport link, or by statically reserving credits in the base channel for display-refresh traffic. In either

case, the northbridge separates out the display-refresh traffic from the other contents of the channel and queues it separately. This allows true isochronous requests (which require a relatively low bandwidth) and CPU requests, both of which are latency sensitive, to bypass the display-refresh requests in the northbridge; meanwhile the display-refresh requests can in turn bypass base-channel requests. At the memory controller, display-refresh requests are initially treated as low priority. However, pending display-refresh requests are aged, and their priority increases over time to provide the latency guarantee.

Dynamic performance scaling

Operating systems issue messages to hardware to indicate current CPU activity levels on a per-core basis. These are called C-states and P-states, according to the Advanced Configuration and Power Interface (ACPI) specification.[2] C0 indicates the CPU core is active; other C-states represent lower-power states with varying degrees of allowed recovery time. Within C0, the P-states represent the core's level of activity. (In a non-C0 state, the P-state just indicates the performance state that the core will return to upon reentry to C0.)

Table 2 lists the approximate amount of time Griffin spends in the various power states for the MobileMark 2005 battery-life benchmark running on Microsoft Windows XP.[3] In a mobile environment, the processor spends only a small fraction of the time in the highest-power core states. Also, the load is not spread evenly over the two cores; one is in an active state more than the other.

The northbridge is responsible for generating the clocks to the CPU cores. It does this with a single shared high-speed phase-locked loop (PLL) and programmable dividers for each core. The PLL frequency is fixed at the part's maximum rated frequency. The dividers are capable of fine-grained frequency steps, dividing the peak frequency by $n/2$, where n is any integer, providing a wide variety of performance points.

The northbridge adjusts the operating frequency of each core to match its

performance requirements as specified by its P-state. The northbridge can do this quickly by changing the divider, without requiring a PLL relock or interrupting CPU execution. Since such changes are lightweight, the operating system can afford to change the frequency frequently to tune the CPU performance to meet, but not wastefully exceed, the requirements and minimize dynamic power consumption.

Griffin implements a total of seven power planes with voltages coming from external regulators. Four of the planes are small, and are used to power the HyperTransport and DDR physical layer interface macros; they operate at fixed voltages. The other three are the main logic planes: one for each CPU core and one for the northbridge. Variable regulators under northbridge control generate the voltages on these planes. This allows the cores to be set to the minimum voltage (and, therefore, static power), supporting a particular operating frequency, and dynamically adjusted as that frequency changes with P-state on a per-core basis.

The benefit of voltage and frequency independence for the CPU cores is particularly evident for workloads that do not fully saturate both CPU cores. Figure 4 summarizes the percentage power savings achieved in the processor because of the independent CPU voltage and frequency planes for four different workloads: a game ("City of Heroes") and three different software DVD playback scenarios or applications.

If the operating system halts a core (changes its C-state from C0 to C1), the northbridge applies a large divisor to its clock, which greatly reduces the dynamic power. If both cores are halted, they are moved to the non-AltVID C1E state referred to as *deep sleep*. Then, their voltage can be reduced to the minimum value (AltVID) required for flip-flops to maintain state. This state is called *deeper sleep*. Coherent I/O traffic that needs to probe the CPU caches will force the cores back to the non-AltVID C1E state, but it is only necessary to go to the minimum frequency and voltage required to service them, not to the active P-state. By performing the minimum voltage ramp-up and ramp-down, the northbridge can complete the
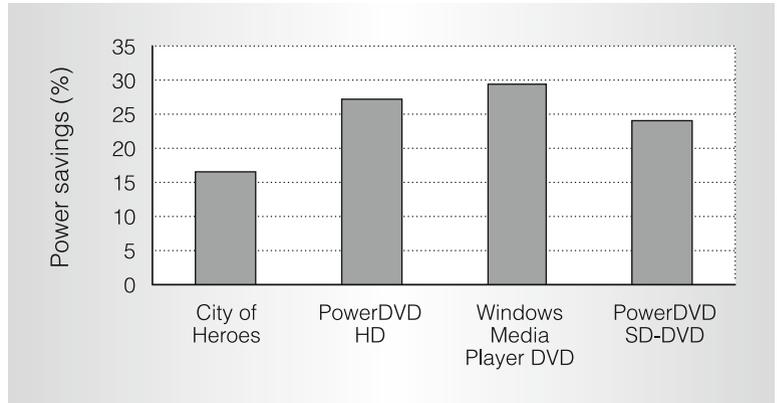

Figure 4. Griffin's percentage power savings resulting from independent CPU voltage and frequency planes.

probes and return the core to the deeper-sleep state in the minimum possible time. This is called *AltVID popup*.

C- and P-state transitions affect only CPU core clocking and voltage. Because the northbridge is on a separate power plane and clock grid, northbridge power states and clocking—and, therefore, the graphics path to memory—are unaffected.

HyperTransport 3 is a high-bandwidth interconnect, necessary to support the bandwidth required by a DX10 UMA graphics chipset to the CPU's integrated memory controller. Griffin supports a 16-bit link in each direction running at a frequency of 2.2 GHz DDR, for a total raw bandwidth of 8.8 Gbytes/s each way. Although this is necessary to supply peak load, the system does not always require peak bandwidth: I/O devices and rendering engines are not always active, and display-refresh load depends on the screen's resolution and compression and can also be bursty.

The chipset, not the operating system, is in the best position to predict I/O load on a fine-grained basis. In a mobile UMA system, rendering and display-refresh requests from the graphics engine are the dominant component of I/O traffic. Because this is internal to the chipset, the chipset knows its activity level and bandwidth requirements for that load. The chipset is also the bridge to external I/O buses and can take into account power-state changes on those buses, which are triggered by the devices plugged into them. Chipset-

....
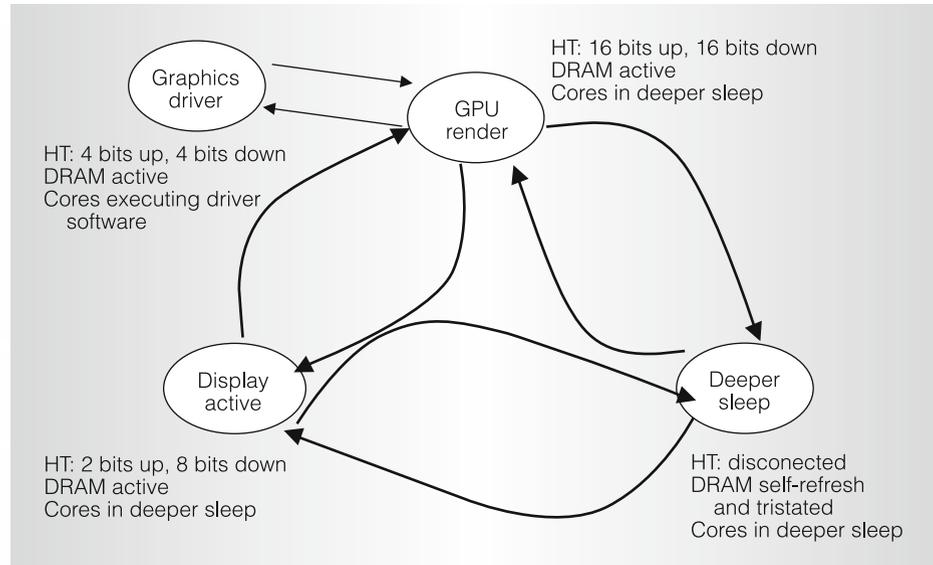
HOT CHIPS 19

Figure 5. Typical power-state transitions in a running system.

initiated power-state changes can take place over a granularity of microseconds, while operating system-initiated states change over milliseconds.

Griffin supports all the HyperTransport 3 methods of reducing power on less-than-fully utilized links, under chipset control. The link width can be reduced by powers of two in each direction independently, and the link frequency can also be reduced to match the link capabilities to the load. The link can be shut off entirely during periods of inactivity, or alternately disconnected and reconnected in a stutter mode. These transitions are made quickly in hardware, without involving or notifying the operating system, and do not affect the path between the CPU cores and memory. When the cores are halted, the chipset can use the same mechanism to signal DRAM to go into self-refresh and the northbridge to go into a shutdown state.

The combination of processor and chipset working together lets Griffin dynamically, and at fine granularity, match power to the system's current requirements for maximum power efficiency. Figure 5 represents a sample flow showing typical power-state transitions in a running system.

In the *graphics driver* state, the CPU might be active, preparing for the next scene to be rendered. The cores and memory are active, but the I/O load is low, so the HyperTransport link is only partially active.

When the setup completes, the system could pass to the *GPU render* state, where the GPU becomes active to generate the next frame. It might require peak I/O and memory bandwidth, while the CPU goes idle.

Once the frame is drawn, the GPU can go to a lower-power state, and the system might transition to the *display active* state. The I/O link must then provide just enough bandwidth to support display refresh, which the northbridge could reach by lowering the link width and/or frequency.

Finally, if display refresh is handled by stuttering, Griffin could completely shut down between bursts. The cores are in *deeper sleep*, the memory is in self-refresh, and the I/O link is disconnected. This is the minimum-power state and will persist until awakened by the chipset.

Table 3 summarizes the Griffin power optimization features. The following are the key principles:

- Tune the frequency and voltage of individual functional units to the minimum required to support the current load. Shut off what's not in use.

Table 3. Griffin power optimization features.

Feature	Attributes
Independent CPU core voltage planes and frequency selection	Matches power consumption to CPU performance delivered.
Separate voltage plane for on-die northbridge	Enables CPU deeper sleep with integrated graphics.
Dynamic HyperTransport 3 link power management	Matches power consumption to interconnect bandwidth delivered.
Autonomous hardware control of CPU core deeper-sleep state	Increases residency in CPU deeper-sleep state.
Autonomous hardware control of DRAM self-refresh state	Increases residency in DRAM sleep state.
CPU core deeper-sleep wakeup to service probes at lowest P-state	Increases residency in CPU deeper-sleep state.

- Software makes the best decisions about some power-state transitions, and hardware about others. Put the control where the best information is, and enable them to work together.
- Power-state transitions must be fast to be useful. If the system can't get out of a low-power state in time to service latency-sensitive requests, it can't safely go into it in the first place.

On-die thermal management

Previous generations of AMD processors had a single on-die thermal diode that had to be monitored by external circuitry (see Figure 6a). In the event of an over-temperature event, an external controller scanning the thermal monitor circuit via SMBus[4] would signal the processor to throttle using the ProcHot pin, which would reduce traffic and power, allowing the chip to cool.

Griffin integrates the thermal-monitoring circuit on-chip. Multiple sensors are placed at interesting points around the die for more precise measurement. The increase in precision allows the thermal trip point to be set less conservatively, increasing the part's operating range. The results are signaled to the embedded controller via the SideBand Thermal Sensor Interface (SB-TSI) bus, which implements a subset of the SMBus functionality, and control logic inside Griffin can assert ProcHot directly to signal an over-temperature event. In addition, a new MemHot signal lets DRAM flag an over-temperature condition. Signaling by any of these methods, including internal detection, can move Griffin to a lower-power state.

Griffin represents an important step forward in AMD's mobile processor line, with significant advances in both performance and power-thermal management. The new northbridge, specifically optimized around UMA applications, min-
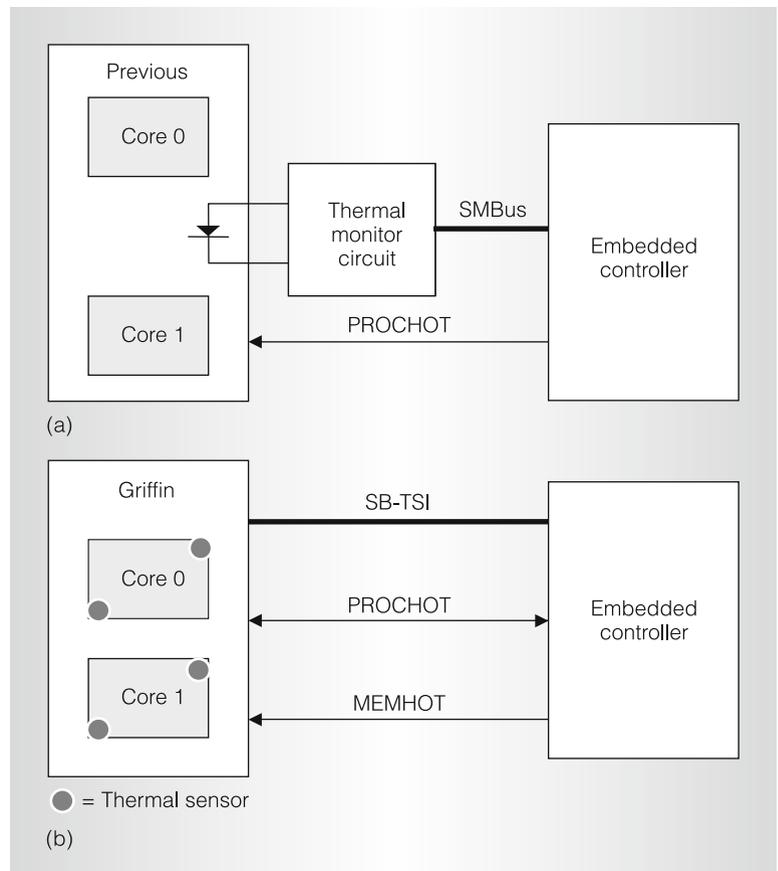

Figure 6. On-die thermal management of previous AMD processors (a) and of Griffin (b). Griffin's on-chip thermal monitoring is more precise, so the trip point can be set less conservatively.

imizes power without penalizing CPU or graphics performance.

The new level of coordination between the CPU and chipset introduced in Griffin lays the architectural groundwork for succeeding generations of mobile parts based on AMD's *Accelerated Computing* concept, which will integrate more of the chipset, including graphics, directly onto the processor die. MICRO

References

1. *HyperTransport I/O Link Specification*, revision 3.00c, HyperTransport Consortium, Sept. 2007, http://hypertransport.org/docucontrol/HTC20051222-0046-0023.pdf.

2. *Advanced Configuration and Power Interface Specification*, revision 3.0b, Oct. 2006, http://www.acpi.info/DOWNLOADS/ACPIspec30b.pdf.

3. *MobileMark 2005*, white paper, revision 1.0, Business Applications Performance Corp., May 2005, http://www.bapco.com/techdocs/MobileMark2005WhitePaper.pdf.

4. *System Management Bus (SMBus) Specification*, version 2.0, System Management Interface Forum, Aug. 2003, http://smbus.org/specs/smbus20.pdf.

Jonathan Owen is a principal member of technical staff at AMD, working in northbridge architecture and design for mobile CPUs. He has been a contributor to both the HyperTransport I/O Link and PCI Express Base specifications. Owen has an MEng in electrical engineering from Cornell University.

Maurice Steinman is an AMD Fellow, working in northbridge architecture and design for mobile CPUs. He has been working in the computer industry for more than 20 years since graduating from Rensselaer Polytechnic Institute with a BS in computer and systems engineering in 1986.

Direct questions and comments about this article to Rob Keosheyan, AMD Corporate Communications, 6800 Burleson Rd, Bldg. B312 MS 567, Austin, TX 78741; rob.keosheyan@amd.com.

For more information on this or any other computing topic, please visit our Digital Library at http://computer.org/csdl.