Reed Electronics Group

MICROPROCESSOR REPORT

www.MPRonline.com

THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

ARC 601 Gets Small

Virage Logic Introduces Tiny 32-Bit Microcontroller Core

By Tom R. Halfhill {1/19/10-01}

It's a race to the bottom—in a good way. The trend of replacing 8- and 16-bit microcontrollers with faster 32-bit devices has processor vendors rushing to shrink their cores to tinier dimensions. The smaller the core, the smaller the compromise in power consumption and cost when developers leave their 8- and 16-bit chips behind.

The latest entry in this race is the ARC 601. It's the first new processor core Virage Logic has introduced since acquiring ARC International in September 2009. Although the ARC 601 is a relatively minor variation of the five-year-old ARC 605, it affirms that Virage Logic is committed to the ARC product line and isn't retreating before market leader ARM. (See *MPR 9/14/09-01*, "Summer Shopping Spree.")

Indeed, the competition for tiny 32-bit licensable processor cores is growing. The ARC 601 closely follows introductions of small cores from ARM (Cortex-M0), MIPS Technologies (MIPS32 M14K), and Tensilica (Xtensa 8). Additional competition will come from Cambridge Consultants (XAP5a) and Cortus (APS3). The field has become so lively that last year we published a detailed comparison of tiny 32-bit cores. (See *MPR 5/11/09-01*, "Itty-Bitty 32-Bitters.")

With a minimum configuration of only 12,600 gates, the synthesizable ARC 601 will be a contender in this flyweight class. It's based the ARC 600 microarchitecture and is most closely related to the ARC 605 preconfigured core. The ARC 601 supports the ARCompact instruction-set architecture (ISA), which can switch modelessly between 16- and 32-bit instructions to use less memory than a conventional 32-bit RISC instruction set. The new processor works with ARChitect-2—a graphical processor-configuration tool—and ARC's Metaware software-development tools. A key advantage of the ARC 601 over the ARM Cortex-M0 is a configurable ISA, which allows developers to add application-specific instructions and extensions.

Virage Logic announced the ARC 601 on December 15 and is already shipping to licensees. The first publicly announced licensee is Phison Electronics of Taiwan, which will use the ARC 601 to replace 8051 cores in flash-memory controllers.

With sales of ARM-based microcontrollers rising and their prices falling well below $1, it's the right time for Virage Logic to claim a share of this high-volume market. Thousands of 8- and 16-bit designs will migrate to 32 bits in the years to come.

Shrinking With Age

ARC introduced the ARC 600 configurable processor in 2003 and the ARC 605 preconfigured core in 2005. Until now, the ARC 605 was the smallest member of this family. Although a base configuration of the ARC 605 is about 17,725 gates—already pretty small for a 32-bit processor—the 12,600-gate ARC 601 is about 29% smaller. After synthesis for TSMC's 90nm-G process, the ARC 601 is 0.089mm^2. The ARC 605 is 0.12mm^2, a difference of about 25%.

Yet, fundamentally, the ARC 601 and ARC 605 are nearly the same. (See *MPR 12/15/03-01*, "ARC Alters Trajectory," and *MPR 3/14/05-02*, "ARC's Preconfigured Cores.") Both processors have five-stage pipelines and fully support the 16/32-bit ARCompact ISA. Software written for the

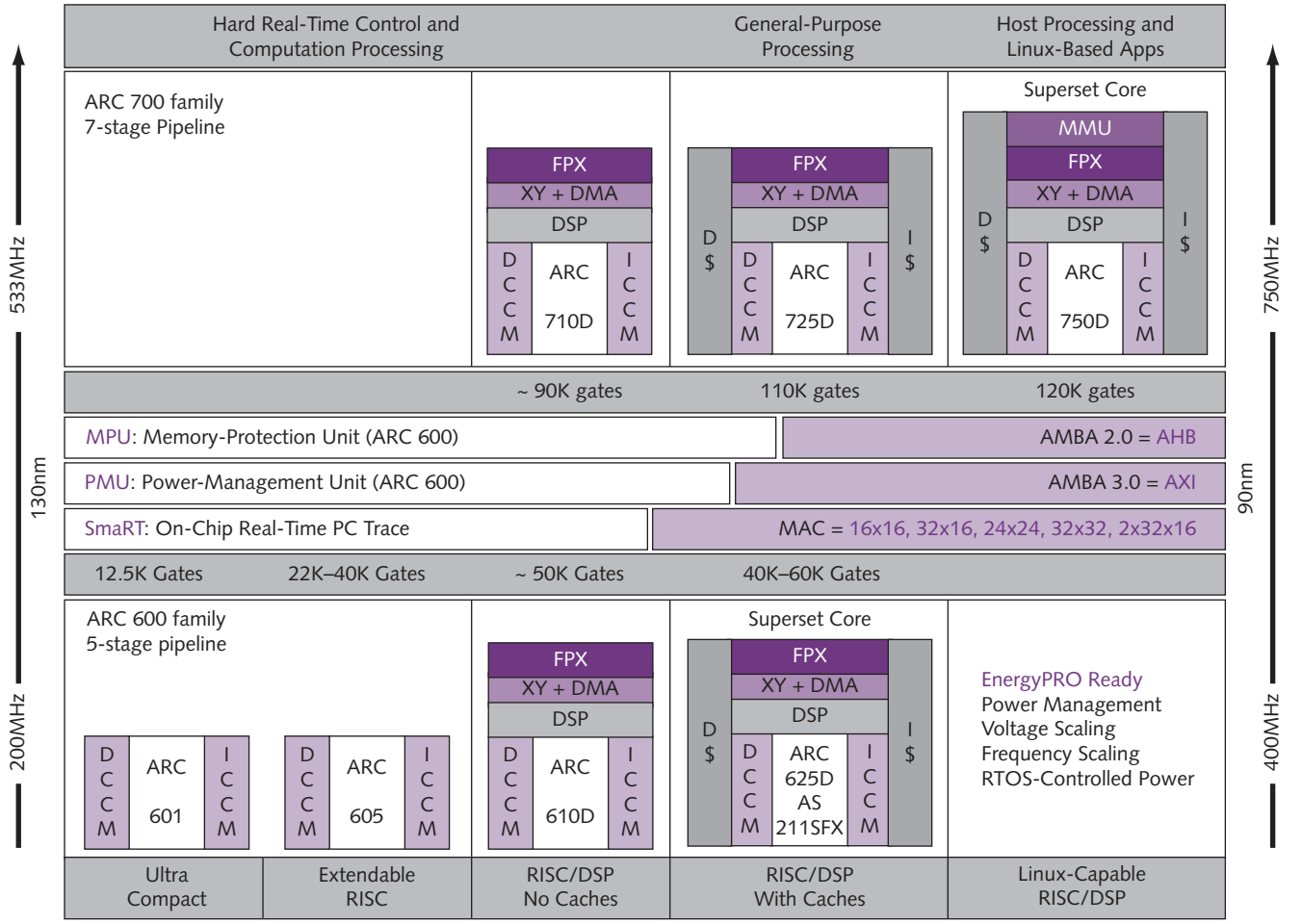

Figure 1. Two closely related microarchitectures form the basis of the ARC microprocessor product line. The ARC 600 is the smallest microarchitecture and is intended primarily for microcontrollers and deeply embedded applications. The ARC 700 is more capable and optionally supports virtual-memory operating systems. Both families are based on the ARCompact instruction-set architecture, which allows programmers to freely mix 16- and 32-bit instructions. All these processors are highly configurable, and all share the same hardware- and software-development tools.

ARC 601 will be upward compatible with other ARC 600 processors and with the ARC 700 family as well. Downward compatibility is possible if there are no dependencies on extended features. Figure 1 illustrates the ARC 600 and ARC 700 product lines.

In contrast, ARM's Cortex-M0 is upward compatible with other Cortex-M cores, but not downward compatible, because the Cortex-M0 supports only six 32-bit instructions. To improve code density and lure developers away from their memory-stingy 8- and 16-bit processors, the Cortex-M0 supports mostly 16-bit Thumb and Thumb-2 instructions.

The ARC 601 achieves the same goal with its ARCompact ISA, a unified 16/32-bit instruction set that enables a preponderance of 16-bit operations. Virage Logic claims that ARCompact can reduce code size by as much as 40%. Keep in mind that a 50% reduction would be ideal if every 32-bit instruction were replaced with a 16-bit equivalent. (See the sidebar, "ARCompact: An Elegant 16/32-Bit ISA," in *MPR 2/18/03-06*, "Soft Cores Gain Ground.")

Last November, MIPS Technologies announced a similar 16/32-bit ISA called microMIPS, claiming a 35% typical reduction in code size. The first processors to use the microMIPS ISA will be the MIPS32 M14Kc and MIPS32 M14K, scheduled to ship in February. The M14K is the smaller of these cores and will compete with the ARC 601. (See *MPR 11/16/09-01*, "MicroMIPS Crams Code.")

MicroMIPS has much in common with ARCompact, so it's a tossup which 16/32-bit ISA is better. For C programmers, the deciding factor may be the efficiency of the compiler, not the merits of the ISA. ARC's Metaware division has spent eight years refining its development tools for ARCompact. MicroMIPS is brand-new, with a GNU-based tool chain from CodeSourcery.

For backward compatibility, the first microMIPS processors have an extra instruction decoder that supports the long-established MIPS32 ISA, too. ARM's stripped-down ISA for the Cortex-M0 will probably beat both ARCompact and microMIPS in code density, at the cost of 32-bit compatibility.

Dieting to Lose Gates

To transform the 17,725-gate ARC 605 into the 12,600-gate ARC 601, Virage Logic further optimized the five-year-old design, made some standard features optional, and removed the possibility of adding some optional features. Frankly, there wasn't much fat in the ARC 605 to begin with, so the diet hasn't altered the ARC 601 beyond recognition.

Neither processor supports caches. Instead, as an option, they have closely coupled memories for keeping instructions and data near the core. These memories serve much the same purpose as caches while requiring no extra logic for caching schemes. With the ARC 601, instruction memory can range in size from 512 bytes to 512KB and is buffered with a two-slot prefetch queue. Data memory can range in size from 512 bytes to 256KB. Both memories are single-ported SRAM arrays with 32-bit I/O interfaces. For very small designs, these local memories can be omitted altogether.

Closely coupled memories are generally preferred over caches for real-time applications, because cache behavior is nondeterministic. The trade-off is that programmers shoulder more responsibility—no cache logic is working in the background to keep the memories filled with relevant instructions and data. Most other processors in this class lack support for caches, too. (Exceptions are the Cortus APS3 and Tensilica Xtensa 8.)

Some optional features omitted from the base configuration of the ARC 601 are a 32-× 32-bit multiplier, a normalize instruction, a swap instruction, a barrel shifter, and a cyclic redundancy check (CRC) instruction. Developers who weep at the loss of these features can restore them with a few mouse-clicks in the ARChitect-2 configuration tool. Another check-box option determines whether the ARC 601 uses little-endian or big-endian memory addressing.

The ARC 601's 32-bit interface for system I/O is a rather generic Basic Virtual Component Interface (BVCI). This simple interface keeps the base configuration of the ARC 601 small, but many developers will spend a few thousand gates to add an AMBA-compatible AHB or AXI bus bridge.

The ARC 601 is not alone in this regard. Except for the ARM Cortex-M0, all other processors in this class have native system interfaces that also require an optional bridge to AMBA or something else. Only the Cortex-M0 has a 32-bit AMBA AHB-Lite bus as a standard feature. Some developers prefer a simple I/O interface for deeply embedded designs that have few on-chip peripherals.

Limited Interrupt Capability

Another pared-down feature of the ARC 601 is the interrupt system. It supports 8 or 16 interrupts (configurable), with only two priority levels. All other processors in this class can support at least 32 interrupts and at least four priority levels. The MIPS M14K and Cortus APS3 lead the pack by supporting up to 256 interrupts. The ARC 601's interrupt system is sufficient for many deeply embedded real-time applications, but some applications will exceed its capabilities.

Feature	TSMC 130nm-G	TSMC 90nm-G	TSMC 65nm-GP
Frequency (F_{Max})	233MHz	408MHz	533MHz
Core Area	0.175mm^2 (100MHz)	0.089mm^2 (100MHz)	0.039mm^2 (100MHz)
Dynamic Power	0.038mW / MHz	0.017mW / MHz	0.013mW / MHz
Power @ F_{Max}	8.85mW	6.93mW	6.92mW

Table 1. ARC 601 metrics in three TSMC fabrication processes. The maximum clock-frequency numbers assume worst-case process and operating conditions. The core-area numbers assume a target frequency of 100MHz, not the maximum clock frequencies in the table. Core areas assume 80% utilization with Virage Logic's SVT High-Density cell library. (Data source: Virage Logic.)

The ARC 601 has no provisions for adding ARC's configurable DSP extensions. Those extensions include various multiply-accumulate (MAC) instructions, saturating math, and separate X/Y data memories closely coupled to the core. If the target application requires signal processing, the next step on the product ladder is the ARC 610D, a preconfigured ARC 600 core that includes those extensions. The ARC 610D is about three times larger than the ARC 601. (See *MPR 3/14/05-02*, "ARC's Preconfigured Cores.")

All these economies keep the ARC 601 small and power-stingy, as Table 1 shows. The performance metrics in this table assume a "standard template" configuration of the ARC 601: a little-endian base core with eight interrupts, 64KB local instruction memory, and 2KB local data memory. The processor was synthesized using Virage Logic's Standard Voltage Threshold (SVT) High-Density cell library for three fabrication processes: TSMC's 130nm-G, 90nm-G, and 65nm-GP.

Obviously, this is a very small, low-power microprocessor core. At these dimensions, the difference between the 32-bit ARC 601 and a typical 8- or 16-bit processor verges on insignificant. On-chip peripheral logic and memory, not the processor, will dominate the design.

The only alarm bell in Table 1 is that migrating a design from 90nm to 65nm doesn't buy much reduction in dynamic power consumption, unless the clock speed is restrained. Of course, this phenomenon isn't unique to the ARC 601—it's a consequence of the diminishing returns from deep-submicron CMOS. The sweet spot looks like 90nm.

Formidable Competitor: Cortex-M0

The toughest competitor in this market is ARM's Cortex-M0, which is attracting customers faster than any previous ARM processor. (See *MPR 3/2/09-01*, "ARM's Smallest Thumb.") Since its debut in February 2009, ARM's smallest processor core has been licensed by 15 companies—almost half of them first-time ARM customers. So far, ARM has released the names of five licensees: Austriamicrosystems, Chungbuk Technopark, Melfas, NXP Semiconductor, and Triad Semiconductor.

The only household name in that group is NXP (formerly Philips Semiconductor), which began shipping its first 15 Cortex-M0 microcontrollers in December. NXP recently told *EETimes* that customers are clamoring for even more variations of this basic microcontroller design. ARM says other top semiconductor vendors have licensed the Cortex-M0, but they're not ready to make public announcements.

Interestingly, Virage Logic has a relationship with NXP, too. Last November, Virage Logic acquired significant assets from NXP, including CMOS libraries, peripheral blocks, infrastructure for SoCs, 25 families of patents, other semiconductor intellectual property (IP), and about 150 employees. In addition, Virage Logic and NXP have forged a multiyear alliance to develop, share, and license more IP. The alliance raised the possibility that NXP might introduce some ARC-based microcontrollers to supplement its ARM-based product line. It's common for microcontroller vendors to offer multiple CPU architectures.

However, NXP told *Microprocessor Report* that ARC-based microcontrollers aren't on the roadmap. Geoff Lees, NXP's vice president and general manager for the microcontroller product line, says his company is fully committed to ARM for 32-bit designs. Lees cited ARM's rapidly growing Cortex family and ARM's unmatched ecosystem of development tools and software. NXP's new-found devotion to ARM is another sign that the Cortex family is gaining traction in the marketplace.

Despite an alliance with Virage Logic, NXP has no plans to make ARC-based microcontrollers.

Cortex-M0 Uses Less Power

In most respects, the ARC 601 will labor to compete with the Cortex-M0. Despite having a deeper pipeline (five stages vs. three), the ARC 601 is actually a little slower than the Cortex-M0 (533MHz vs. 550MHz) when fabricated in TSMC's 65nm-GP process. Oddly, the reverse is true in 65nm-LP: the ARC 601 is 31% faster (354MHz vs. 270MHz.)

ARM theorizes that the Cortex-M0's speed advantage in 65nm-GP is due to larger transistors in that particular implementation. For 65nm-GP, ARM synthesized the core using a 12-track cell library (which has taller cells, hence room for larger transistors), whereas the 65nm-LP implementation uses a 10-track library.

All things being equal, the ARC 601 should run faster than the Cortex-M0, thanks to a deeper pipeline. The ARC 601 executes more Dhrystone mips per megahertz (1.2Dmips vs. 0.9Dmips), so it should deliver slightly more throughput at any clock frequency. ARM optimized the Cortex-M0 primarily for area and power, not for speed.

Consequently, in area and power, the Cortex-M0 looks better than the ARC 601. In TSMC's 65nm-GP process, ARC estimates 0.013mW per megahertz, whereas ARM estimates 0.0075mW per megahertz—about half the power. It's a big enough difference to give ARM the advantage in power/performance efficiency (120Dmips per milliwatt vs. 92.3Dmips/mW), despite the ARC 601's advantage in Dmips per megahertz.

A Puzzling Size Discrepancy

Comparing the sizes of these cores is tricky. At first glance, the ARC 601 and Cortex-M0 appear to be nearly the same size: 12,600 gates vs. 12,000 gates. However, as Table 1 shows, the ARC 601 will be 0.089mm^2 when synthesized for 90nm-G and a target clock speed of 100MHz. ARM says the Cortex-M0, under the same conditions, will be only 0.042mm^2—less than half the size.

Why is the ARC 601 larger, despite a similar gate count? ARC has no explanation. Both companies say they are estimating gate counts using NAND2×1-equivalent gates—the smallest type of NAND2 gate—which yield the highest theoretical count. The difference may be attributable to routing efficiency or the cell libraries. (Such discrepancies aren't uncommon. See the sidebar, "Gate Count? Depends Who's Counting" in *MPR 5/11/09-01*, "Itty-Bitty 32-Bitters.")

The ARC 601 does have two things the Cortex-M0 can't match: the option of closely coupled memories and a configurable ISA. The Cortex-M0 offers neither caches nor local memories, and no ARM processor allows developers to extend the ISA with custom instructions.

Specifications aside, the main advantage of the Cortex-M0 over all competitors is that it's from ARM. It's the smallest incarnation of the world's most popular 32-bit microprocessor architecture. However, the Cortex-M0's pedigree is diluted somewhat by its oddball ISA, which discards almost all 32-bit instructions in favor of 16-bit Thumb and Thumb-2 instructions. It's an ARM processor, all right, but it won't run existing 32-bit ARM code. That difference opens the door, at least a crack, for competing processors.

Xtensa 8 More Configurable

Another strong competitor is Tensilica's new Xtensa 8 core. Introduced in October, it's the latest version of Tensilica's configurable-processor architecture. It has few changes over Xtensa 7—mostly some new options. That's not a bad thing, because it was well suited for its target market to begin with. (See the sidebar, "Tensilica Debuts Xtensa 8 Core," in *MPR 11/30/09-01*, "Tensilica Tweaks Xtensa.")

The base configuration of the Xtensa 8 processor is a little larger than the ARC 601, but the difference isn't important. The main advantage of Xtensa 8 is that it's even more configurable than the ARC 601. Developers can add L1 caches, larger local memories (up to 4MB), an optional memory-management unit (MMU), and a wider system bus (up to 128 bits).

Simple check-box options in Tensilica's configuration tool let developers add 32 pairs of GPIO ports or 32-bit queued I/O ports that bypass the system bus. (The latter ports are ideal for adding coprocessors and custom logic.) Another check-box option is an AHB-Lite or AMBA-3 AXI bridge with synchronous or asynchronous bus clocking.

And, like the ARC 601, Xtensa 8 has a customizable ISA, so developers can create application-specific instructions and extensions.

Although Xtensa is a 32-bit RISC architecture, its standard instruction length is 24 bits, not 32 bits. Additional instructions are 16 bits long. As a result, software written for Xtensa processors can match or exceed the code density of rival 16/32-bit instruction sets.

Virage Logic claims the ARC 601 beats Xtensa 8 in performance and power consumption, but we can't make apples-to-apples comparisons. Virage Logic provides performance metrics for TSMC's 130nm-G, 90nm-G, and 65nm-GP processes, whereas Tensilica provides metrics for TSMC's 45nm-GS and 40nm-LP processes.

A speed-optimized Xtensa 8 core in 45nm-GS can reach 1.03GHz, nearly twice the maximum clock speed of the ARC 601 in 65nm-GP. When optimized for area and power, the same Xtensa 8 configuration in 40nm-LP can stroll at 60MHz while dissipating less than one milliwatt. Allowing for differences in fabrication processes, we judge the ARC 601 and Xtensa 8 to be closely matched.

New Competition From MIPS

The ARC 601 fares better against the new MIPS32 M14K processor, which supersedes the MIPS32 M4K core. That's not surprising, because MIPS designed the M14K mainly to compete with faster processors like the ARM Cortex-M3, not stripped-down cores like the Cortex-M0 and ARC 601. (See *MPR 11/16/09-01*, "MicroMIPS Crams Code.")

Cambridge Consultants and Cortus have pursued the 8- and 16-bit replacement market for years.

Nevertheless, the M14K is a serious contender for 32-bit microcontrollers. Both ARC and MIPS provide performance metrics for their processors in TSMC's 90nm-G process, so we can make some valid comparisons. In 90nm-G, the ARC 601 can reach 408MHz while dissipating only 6.93mW, or 0.017mW per megahertz. In the same process, the M14K reaches only 322MHz while dissipating 35.4mW, or 0.11mW per megahertz. Note the dramatic difference in power consumption—the ARC 601 is approximately an order of magnitude lower. But because the M14K delivers more throughput (1.5Dmips per megahertz vs. 1.2Dmips/MHz), it should match the ARC 601's performance, even at slower clock speeds.

A few features may compensate for the M14K's power handicap. Developers can easily add multiple sets of general-purpose registers (up to 16 sets of 32 GPRs). This option enables faster context switching, because the processor needn't dump and restore the registers each time. Local instruction and data memories can be larger than the ARC 601's (up to 4MB), and an MMU supports two execution modes (user and supervisor), whereas the ARC 601 supports only one mode. The M14K supports either a unified 32-bit interface or separate 32-bit interfaces for local instruction and data memories, whereas the ARC 601 has only a single 32-bit system interface. The M14K also has the option of a separate coprocessor interface.

If a design must accommodate lots of interrupts, the M14K has a clear advantage. It supports up to 256 interrupts with 256 priority levels, whereas the ARC 601 supports only 8 or 16 interrupts with two priorities. Overall, the ARC 601 core is a better choice for the smallest possible low-power designs. The M14K core is better suited for designs that demand more sophistication.

More Competitors From Europe

ARC faces additional competition from two lesser-known companies based in England and France: Cambridge Consultants and Cortus. Both companies have been pursuing the 8- and 16-bit replacement market for several years.

The Cambridge Consultants XAP5a is the most unusual processor in this group. It's really a hybrid 16/32-bit architecture. It has a 16-bit bus, only four 32-bit registers, and instructions that are 16, 32, or 48 bits long. Despite those economies, it's a few thousand gates larger than the ARC 601, though it synthesizes to a slightly smaller size (0.037mm^2 vs. 0.039mm^2 in 65nm-GP). It's also much slower—a two-stage pipeline limits the maximum clock speed to 220MHz in 65nm-GP, compared with 533MHz for the ARC 601.

Cambridge Consultants argues that the XAP5a is more than fast enough to replace an 8- or 16-bit microcontroller in most applications, and that memory is the most important resource to conserve in small systems. The XAP5a's 16-bit interface will cut costs when compared with 32-bit interfaces on rival processors. These arguments have merit, especially in smaller designs that mainly use 16-bit datatypes.

Cortus is a French company that has specialized in small 32-bit processor cores since 2005. The Cortus APS3 is definitely small: at 9,500 gates for a minimum configuration, it's the smallest 32-bit processor core on the market. Like the Cortex-M0, it has only 16 GPRs. Previous versions of the APS3 had no provision for local instruction or data memories, but the latest incarnation allows each local memory to be as large as 4GB. For those who prefer caches, Cortus now offers an optional data cache to supplement the optional instruction cache. It even supports cache coherency for multiprocessing. Another new option is an MMU with a translation-lookaside buffer (TLB).

Cortus says the APS3 can exceed 300MHz in a 130nm-G process, surpassing the ARM and ARC cores in that process. Power consumption is only 0.024mW per megahertz, slightly lower than the ARC 601. Optional coprocessors make the APS3 customizable for specific applications. There's even a 16-bit fixed-point DSP coprocessor, unique in this class. Other options include a Harvard or von Neumann bus architecture and some free peripheral cores. Table 2 compares the ARC 601 with the licensable processor cores

Feature	Virage Logic ARC 601	ARM Cortex-M0	Cambridge XAP5a	Cortus APS3	MIPS MIPS32 M14K	Tensilica Xtensa 8
CPU ISA	ARCompact	ARMv6-M	XAP5	Cortus APS	MIPS32 R2	Xtensa
Arch. Width	32 bits	32 bits	16 / 32 bits	32 bits	32 bits	32 bits
Instr. Lengths	16, 32 bits	16, 32 bits	16, 32, 48 bits	16, 32 bits	16, 32 bits	16, 24 bits
GPRs	32 x 32 bits	16 x 32 bits	8 x 16 bits (4 x 32 bits)	16 x 32 bits	32 x 32 bits 1–16 sets	Up to 64 x 32 bits
Pipeline Depth	5 stages	3 stages	2 stages	5–7 stages	5 stages	5 stages
L1 Cache (I / D)	—	—	—	I-cache 0–2MB D-cache 0–4KB	—	Optional
Local Instr. RAM	512b–512KB	—	—	0–4GB	0–4GB	0–4MB
Local Data RAM	512b–256KB	—	—	—	0–4GB	0–4MB
Memory Management	Optional MPU	—	MPU	Optional MMU + TLB	MMU (Fixed map)	Optional MMU + TLB
Hardware Multiplier	Optional 32 x 32 bits	2 options 32 x 32 bits	32 x 16 bits	Optional 32 x 32 bits	Fast or small 32 x 16 bits	Optional 16 or 32 bits
External Interrupts	8 or 16	1–32	32	Up to 256 (vectored)	8 (vectored) 256 (external)*	Up to 32
Interrupt Priorities	2	4	4 or 16	16	256	6
Privilege Modes	1	1	3	1	2	1
Custom Extensions	Yes	—	Optional	Yes (coprocessors)	Optional (CorExtend)	Yes
System Interface	BVCI* 1 x 32 bits	AHB-Lite 1 x 32 bits	Native 1 x 16 bits	Native* 1 x 32 bits or 2 x 32 bits	Native MIPS* 1 or 2 x 32 bits	Native PIF* 32–128 bits
Coprocessor Interface	—	—	—	Native 1 x 32 bits	MIPS COP2 32 bits	Optional 2 x 32 bits
Core Frequency (F_{Max})	533MHz (65nm-GP)	550MHz (65nm-GP)	220MHz (65nm-GP)	>300MHz (130nm-G)	322MHz (90nm-G)	1.03GHz (45nm-GS)
Core Size (Base)	12.6Kgates	12Kgates	18Kgates	9.5Kgates	n/a	15Kgates
Dhrystone 2.1	1.2Dmips / MHz	0.9Dmips / MHz	0.93Dmips / MHz	0.88Dmips / MHz	1.5Dmips / MHz	1.28Dmips / MHz
Power (F_{Max})	0.013mW / MHz (65nm-GP)	0.0075mW / MHz (65nm-GP)	0.006mW / MHz (65nm-GP)	0.024mW / MHz (130nm-G)	0.11mW / MHz (90nm-G)	0.014mW / MHz (45nm-GS)
Power Efficiency (Dmips / mW)	92.3 (65nm-GP)	120 (65nm-GP)	155 (65nm-GP)	36.6 (130nm-G)	12.5 (90nm-G)	91.4 (45nm-GS)
Introduction	Dec 2009	Feb 2009	Jul 2008	Jul 2005	Feb 2010	Oct 2009

Table 2. Feature comparison of the Virage Logic ARC 601, ARM Cortex-M0, Cambridge Consultants XAP5a, Cortus APS3, MIPS Technologies MIPS32 M14K, and Tensilica Xtensa 8 processors. All are licensable, synthesizable 32-bit embedded-processor cores for microcontrollers and deeply embedded applications. All are so small that other differences will probably matter more to developers. Clock frequencies are the maximums that vendors specify for a given fabrication process. Mind the process differences when comparing these specs—*MPR* was unable to obtain data for equivalent processes. *Optional AMBA AHB, AHB-Lite, or AMBA-3 AXI bridges are available for these native system interfaces.

it will most likely face when competing to replace 8- and 16-bit microcontroller designs.

Overall, we think the Cortus APS3 and Cambridge Consultants XAP5a are credible alternatives to the ARC 601, Cortex-M0, MIPS M14K, and Xtensa 8. The XAP5a claims the highest power efficiency in this group—155Dmips per milliwatt, according to the latest Dhrystone benchmarking by Cambridge Consultants. The Cortus APS3 has the smallest gate count for a base configuration. *MPR* covered the XAP5a and APS3 in detail last year. (See *MPR 5/11/09-01*, "Itty-Bitty 32-Bitters.")

One important factor we can't compare is cost, because processor-IP companies rarely disclose their licensing fees in public. Keep in mind that smaller companies are hungrier for business and more likely to offer bargains.

One-Stop Shopping

In summary, the ARC 601 is a minor addition to the ARC product line that competes favorably with other small 32-bit processor cores. In the past year, we've reported on similar new cores from ARM, MIPS, and Tensilica, so it's a busy market segment. There has been an unusual amount

of acquisition activity among microcontroller vendors, and prices of 32-bit controllers are falling into the realm of pocket change. Although 8- and 16-bit cores and controllers aren't anywhere near extinction, the long-term trend is clear.

Virage Logic is striving to differentiate by assembling a broader ecosystem. Last year's acquisition of ARC International added the central ingredient of 32-bit processor cores and some additional peripheral IP to Virage Logic's existing catalog. Already a purveyor of physical-IP libraries, Virage Logic is becoming as broad-based as ARM, which absorbed Artisan's physical IP a few years ago. (See *MPR 9/7/04-01*, "ARM Extends Its Reach.")

As we noted last September, Virage Logic wants to offer a one-stop shop. The strategy is to reap as much revenue as possible from chip designers by offering synthesizable processor cores, peripherals, and memory, plus cell libraries, development tools, a real-time operating system (ARC's own MQX), and miscellaneous middleware. Potentially, designers can save time and money while having more confidence that all the pieces will work together. (See *MPR 9/14/09-01*, "Summer Shopping Spree.")

Price & Availability

Virage Logic's ARC 601 embedded-processor core is available for licensing now. The synthesizable core is delivered in Verilog format with the ARChitect-2 processor-configuration tool. Some optional features incur no additional costs when added to the core with this tool. Extended features, such as AMBA bus bridges, are available separately. Virage Logic doesn't publish the licensing fees. For more information, visit: *www.viragelogic.com/render/content.asp?pageid=892*

It's a bold strategy, and Virage Logic is moving boldly, completing major acquisitions of ARC and a chunk of NXP in the past six months. Although ARM is successfully executing a similar strategy, previous attempts by ARC and MIPS to create a one-stop shop have failed. To succeed, Virage Logic must avoid their missteps and ensure that its expanding product line is as well integrated as the marketing promises.