# MOTOROLA ENHANCES STARCORE DSP

## SC140e Core Offers New Instructions, Caches, and Task Protection

*By Tom R. Halfhill {10/20/03-01}*

........................................................................................................................

Some of Motorola's latest 3G mobile phones will use an enhanced StarCore DSP that the company revealed at **Microprocessor Forum 2003**. The new SC140e core has several advanced features not found in other StarCore DSPs, including a new memory subsystem

and a user-level privilege mode. Motorola says the enhancements will eventually appear in a future architecture from StarCore LLC, a spinoff formed last year by Motorola, Infineon, and Agere (formerly Lucent).

Two related announcements accompanied Motorola's introduction of the SC140e. A day earlier, StarCore LLC announced the first two licensable StarCore DSP cores, the SC1200 and the SC1400. (See text box, "StarCore LLC Offers Soft DSPs.") Previously, only Agere and Motorola could design chips around StarCore DSP cores; now, anyone can license the synthesizable cores to design an SoC. The SC1200 and SC1400 are based on the same SC1000 architecture as Motorola's SC140 and SC140e. Indeed, Motorola's SC140 and StarCore's SC1400 are essentially two different names for the same core, which was introduced in 1999. (See *MPR 5/10/99-03*, "StarCore Reveals Its First DSP.")

On the second day of Microprocessor Forum, Motorola announced the Jupiter architecture, a "convergence platform" for wireless communicators that incorporate PDA functions. The foundation of the Jupiter architecture is an SC140e-based single-core modem chip and an ARM1136JF-S application processor. (*MPR* will report on the Jupiter architecture in a future issue.)

Although Motorola's new SC140e is a synthesizable DSP core like StarCore LLC's SC1200 and SC1400, Motorola has no plans at this time to broadly license it as intellectual property (IP). Instead, Motorola will use the SC140e in its own chips—like the Jupiter-platform modem

chip—and will design SC140e-based SoCs on demand for customers.

In about a year, the SC140e's enhanced features will find their way into the next-generation StarCore DSP architecture, the SC2000, which StarCore LLC will make available as licensable IP. Until then, system designers who want the SC140e's features will either have to hire Motorola to design an SoC to their specifications or buy the Jupiter modem chip. Motorola may introduce additional standard parts based on the SC140e as well.

### SC140e Has Microcontroller Features

Dan Tamir, core architecture manager of Motorola's DSP Platforms Development Center, presented the SC140e at Microprocessor Forum. Tamir described an improved DSP core that can perform some microcontroller functions to reduce the overall chip count and power consumption of a system. The improvements are particularly useful for battery-dependent mobile products, although the SC140e is powerful enough for network-infrastructure applications, too. In addition, the SC140e is binary-compatible with software written for Motorola's existing SC100-architecture cores, including the SC140. (StarCore LLC refers to the SC100 architecture as the SC1000, and to the SC140 microarchitecture as the SC1400.)

New to the SC140e are about a dozen additional instructions; instruction and data caches; provisions for tightly coupled local memory; a memory-management unit

(MMU); an address-translation lookup table with memory protection; and a user-level privilege mode. The new memory subsystem is much more sophisticated, and software programmers should appreciate the ability to run a multitasking real-time operating system (RTOS) that enforces memory protection and user/supervisor privileges. Taken together, these features can dramatically improve the performance, security, and reliability of an embedded system.

At its heart, the SC140e is almost identical to the existing SC140/SC1400, which *MPR* fully described in the May 1999 article referenced earlier. It's a 16-bit fixed-point DSP with a VLIW architecture optimized for communications. It has four ALUs, four multiply-accumulate (MAC) units, four bit-field units, two address-arithmetic units, one bit-manipulation unit, and a branch unit. There are 51 programmer-visible registers, including 16 general-purpose data registers (which also serve as accumulators), all 40 bits wide.

By executing six instructions per cycle—which could include as many as 10 basic operations, counting the multiplies and adds of four MACs—the SC140/SC140e can deliver 1,200 million MACs per second (MMACS) or 3,000 native mips at a nominal frequency of 300MHz, which is achievable in a 0.18-micron process. In a more advanced fabrication process, the core can reach higher clock frequencies, although lower frequencies and voltages may be desirable to save power in battery-operated products. The SC140e in Motorola's Jupiter-platform modem chip will be clocked at only 200MHz, even though it's fabricated in the latest 90-nanometer (nm) process.

Motorola hasn't publicly estimated the power consumption of the SC140e, but the core is almost identical to the SC140, so power levels should be similar. When manufactured in a 0.18-micron process, the SC140 core consumes about 350mW at 300MHz. The SC140e should do much better in a 0.13-micron or 90nm process.

Motorola tinkered with the innards of this four-year-old, but still-impressive, DSP core as little as possible. The most significant difference between the microarchitectures of the SC140 and SC140e is the new memory subsystem, which supports multilevel instruction and data caches, tightly coupled on-chip memory, and an MMU. Figure 1 shows a block diagram of the SC140e.

## Unusual Cache-Locking Scheme

The SC140e memory subsystem supports a three-level hierarchy, including L1 caches, on-chip L2 caches, on-chip L1/L2 tightly coupled memories, and off-chip L3 memory. L1 caches can vary in size over a wide range, although 16K to 64K are likely sizes.

Despite the extra silicon required by caches and memories, Motorola views the new subsystem as a way to save power while boosting performance. Motorola says that in handset applications, the SC140e's active power drops about 20% with L1 caches, compared with a cacheless SC140e core. At the same time, the core's clock frequency can rise about 10–20%, because the memory interface becomes a critical path in a cacheless SC140e system, especially when processing data-intensive communication streams. Motorola bases these estimates on cycle-accurate simulations, using actual application workloads.

Caches always raise the problem of nondeterministic behavior in a system that must respond quickly to real-time events. In the SC140e, Motorola offers two solutions. First, the memory subsystem supports two levels of on-chip memory, named M1 and M2 to distinguish them from the L1 and L2 caches. Programs can store critical code and data in these memories, which are not divided into X and Y regions like the data memories in many other DSPs.

M1 memory, like an L1 cache, is accessible in a single clock cycle. M2 memory is accessible in multiple clock cycles; as with an L2 cache, the actual number will vary according to the chip design. However, programmers have full control over the contents of the M1 and M2 memories; unlike the caches, they are not managed by the processor.

Tightly coupled on-chip memory is a welcome feature, but it's common in many other processors. What is more innovative is Motorola's approach to cache locking in the SC140e. Usually, programmers can lock specific regions of a cache to stop the processor from evicting critical data or instructions needed by a
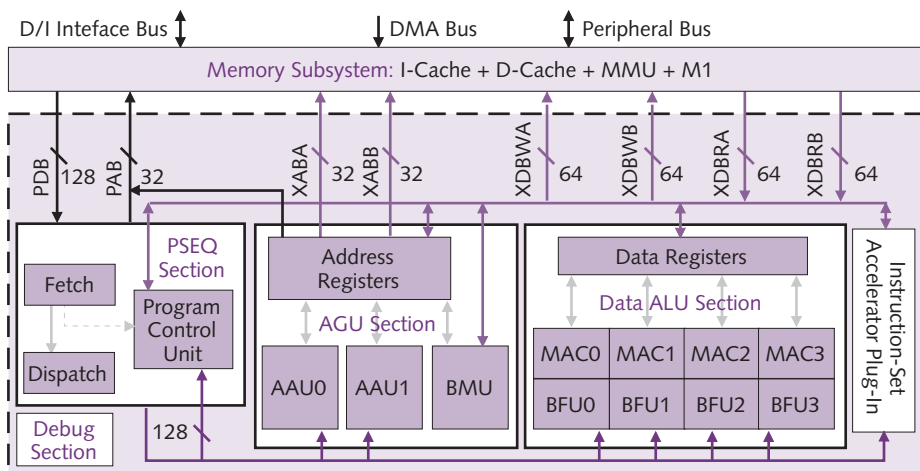


**Figure 1.** SC140e block diagram. The central part of the DSP core—bounded by the dotted lines—is almost unchanged from the existing SC140. Motorola bolted on a new memory subsystem and made some minor changes to support the new user-level privilege mode and task-protection features. At right, the instruction-set accelerator plug-in is a coprocessor interface for adding customer-defined instructions and registers.

real-time task or interrupt. Often, the programmer can lock a portion of the cache on a line-by-line basis. With the SC140e, programmers can lock multiple set-associative ways, which can hold the most-recently-used information from that particular task. Programmers can prevent a task from using the cache ways allocated to other tasks. Furthermore, the programmer allocates the ways by specifying the lower and upper boundaries of the region, which is presorted according to the cache's least-recently-used replacement policy.

By allowing programmers to specify which cache region should remain active for a particular task, Motorola's locking scheme stops the processor from replacing the critical code and data of that task with code and data from other tasks. SC140e caches are 16-way set-associative, and any task can reserve any block of ways within the cache. So, for example, a programmer could allocate an entire 16K cache to only one task (all 16 ways) or to as many as 16 tasks (one way, or 1K, per task). Figure 2 shows how a 16K cache might be equally divided among four tasks, with four ways (4K) per task.

Motorola believes this unusual locking scheme will preserve the SC140e's ability to respond in a timely fashion to real-time events, while still providing the benefits of a cache hierarchy. If the locking scheme proves inadequate for any reason, programmers can guarantee that a real-time task will have instant access to critical instructions or data by using the tightly coupled M1 or M2 memories instead of the caches.

### Task Protection for Safer Multitasking

Motorola has noticed a trend in mobile communicators to shift more system-control functions from the microcontroller to the DSP, which is forcing the RTOSs for DSPs to become more sophisticated. However, most DSPs lack the features needed to run a multitasking RTOS that protects memory against encroachments by different tasks and that enforces multiple levels of access privileges. The existing SC140 DSP core is typical: it has no memory protection, and although it has two execution modes, both modes have almost the same level of access. In effect, every task enjoys the unlimited privileges of a "supervisor."

The SC140e changes all that. For memory protection, the SC140e has an MMU and an address-translation lookup table that uses content-addressable memory (CAM). It's a little different from a translation lookaside buffer (TLB), which is a cache of page-table addresses. The SC140e performs the logical-to-physical address translation of a TLB, but it also checks a task's attempts to access memory against registers that indicate the permission level of the task. If a user-level program tries to access another task's memory

region without permission, the MMU triggers a memory exception.

Naturally, the memory protection would be ineffective if any task could modify the base and bound registers of protected memory, so the SC140e also adds a new access-privilege mode for user-level tasks. In user mode, the processor won't let a task modify the memory-protection registers, access another task's memory without permission, disable interrupts, or perform other operations beyond its privilege level.

As a result, the SC140e now has two distinct privilege modes: user and supervisor. Another mode, called normal mode, carries over from the SC140 to maintain compatibility with existing software. Normal mode has nearly the same privileges as supervisor mode but uses a different stack. In the SC140e, user mode and normal mode share the same stack. Supervisor mode can access both the normal stack and its own special stack, called the exception stack. New software written for the SC140e could use supervisor mode and user mode to the exclusion of normal mode, unless backward compatibility with the SC140 is an issue. Normal mode may still be useful for running such things as device drivers that need supervisor-level privileges while using a different stack.

The SC140e's 32-bit status register has a new protection-enable (PE) bit that indicates when the processor is running in user mode. The TRAP instruction, or any exception in user mode, requests a switch to supervisor mode. A mode switch clears the PE bit, sets another bit in the status register that distinguishes between supervisor and normal modes, and pushes the user-mode program counter and status register onto the exception stack.
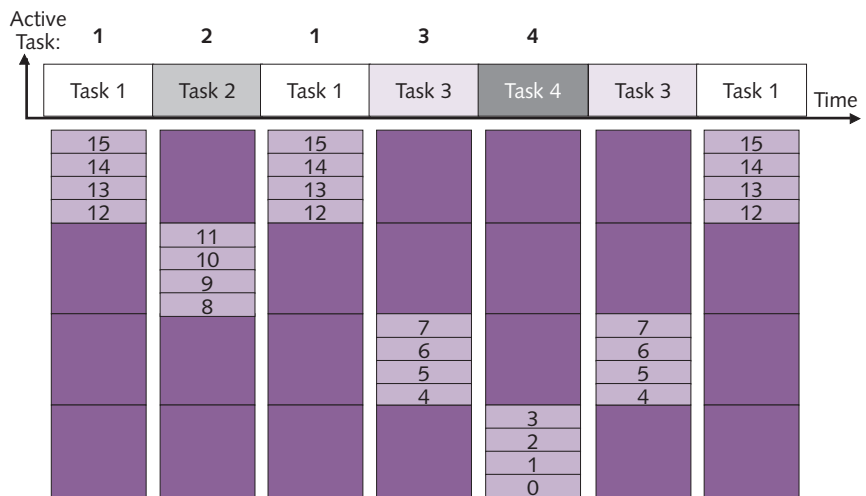


**Figure 2.** In this example, four independent tasks equally divide a 16K cache into four regions. Each region is a block of four ways, 1K per way. As the RTOS switches contexts from task to task, starting at the left with task 1, different portions of the cache become active. The portions of the cache allocated to other tasks are off limits to the current task. Many other allocations are possible, with a minimum of one way (1K in this example) per task.

## StarCore LLC Offers Soft DSPs

StarCore LLC—the yearling spinoff from Infineon, Lucent/ Agere, and Motorola—is shipping the first synthesizable StarCore-architecture DSPs offered as licensable intellectual property (IP). At least two customers, not yet announced, are building the cores into SoCs for 3G handsets.

Until now, the StarCore DSP architecture was the exclusive domain of Motorola and Agere (formerly Lucent), the two companies that formed the original StarCore partnership in 1998. Motorola and Agere jointly designed and introduced the StarCore architecture in 1999. Other companies could get StarCore DSPs only by purchasing the standard parts designed and manufactured by Motorola. After Motorola and Agere decided to spin off their partnership as a separate company, the new StarCore LLC began transforming the architecture into licensable IP.

StarCore LLC is offering six synthesizable variations of two basic microarchitectures, the SC1200 and SC1400. The only significant difference between the microarchitectures is the number of ALU/multiply-accumulate (MAC) units: the SC1200 has two and the SC1400 has four. The six variations of these two microarchitectures differ in the caches, interfaces, and peripheral controllers wrapped around the cores. Table 1 summarizes their features.

Both microarchitectures are software-compatible with existing StarCore DSPs. The lower-end cores are intended for mobile applications, primarily cellphones; the higher-end cores are suitable for infrastructure applications, such as base stations, as well as for cellphones. The synthesizable models, newly written in Verilog, are portable to virtually any fabrication process. StarCore LLC has not disclosed fees but is offering several different licenses, including single-project, term-limited, corporate, and ASIC licenses.

| Feature | StarCoreLLC SP1201 | StarCoreLLC SP1202 | StarCoreLLC SP1203 | StarCoreLLC SP1401 | StarCoreLLC SP1402 | StarCoreLLC SP1403 |
|---|---|---|---|---|---|---|
| DS Core Features | | | | | | |
| Architecure | SC1000 | SC1000 | SC1000 | SC1000 | SC1000 | SC1000 |
| Microarchitecture | SC1200 | SC1200 | SC1200 | SC1400 | SC1400 | SC1400 |
| ALU/MAC Units | 2 | 2 | 2 | 4 | 4 | 4 |
| Address Generation Units | 2 | 2 | 2 | 2 | 2 | 2 |
| Branch Units | 1 | 1 | 1 | 1 | 1 | 1 |
| Data Registers | 16 x 40 bits | 16 x 40 bits | 16 x 40 bits | 16 x 40 bits | 16 x 40 bits | 16 x 40 bits |
| Address Registers | 27 x 32 bits | 27 x 32 bits | 27 x 32 bits | 27 x 32 bits | 27 x 32 bits | 27 x 32 bits |
| Additional Features | | | | | | |
| On-Chip Emulation (Debug) | Yes | Yes | Yes | Yes | Yes | Yes |
| AHB Interface | -- | Yes | Yes | -- | Yes | Yes |
| Optional I-Cache (2-way) | -- | -- | 8K–64K | -- | -- | 8K–64K |
| Optional D-Cache (2-way) | -- | -- | 8K–64K | -- | -- | 8K–64K |
| Program ROM/RAM Interface | -- | 128 bits | 128 bits | -- | 128 bits | 128 bits |
| Data ROM/RAM Interface | -- | 128 bits** | 128 bits** | -- | 128 bits** | 128 bits** |
| SRAM Interface | -- | 128 bits** | 128 bits** | -- | 128 bits** | 128 bits** |
| DMA Controller Interface | -- | Yes | Yes | -- | Yes | Yes |
| Interrupt Controller | -- | Yes | Yes | -- | Yes | Yes |
| Clock Controller | -- | Yes | Yes | -- | Yes | Yes |
| Mapped Accelerator Interface* | -- | -- | 2 x 64 bits* | -- | -- | 2 x 64 bits* |
| Estimated Core Clock Frequencies | | | | | | |
| Low-Leak 0-13μm | | | | | | |
|   Worst-Case | | 150–200MHz | | | 150–190MHz | |
|   Typical | | 190–250MHz | | | 190–240MHz | |
| High-Perf 0-13μm | | | | | | |
|   Worst-Case | | 250–400MHz | | | 240–360MHz | |
|   Typical | | 310–500MHz | | | 300–450MHz | |
| Low-Leak 90nm | | | | | | |
|   Worst-Case | | 200–240MHz | | | 190–230MHz | |
|   Typical | | 250–300MHz | | | 240–290MHz | |
| High-Perf 90nm | | | | | | |
|   Worst-Case | | 340–470MHz | | | 330–450MHz | |
|   Typical | | 430–580MHz | | | 410–560MHz | |
| Availability | Now | Now | Now | Now | Now | Now |

**Table 1.** Shorter signal paths give the simpler SC1200 cores a slight edge in clock frequency over the SC1400 cores, which have twice as many ALU/MAC units. *The MAI allows application-specific logic to appear as a memory-mapped peripheral and can transfer 64 bits at a time. **The two data buses and the direct-memory interface to SRAM can access up to 128 bits in the same line without contention. †StarCore LLC's estimates.

To switch back from supervisor mode to user mode, the SC140e restores the user-level state and executes the RTE (return from exception) instruction, which resets the PE bit. To prevent any mischief, the SC140e allows user-level tasks to see and modify only a copy of the status register. This also preserves compatibility with existing SC140 software.

## New Instructions Accelerate Multimedia

The StarCore SC1000 instruction-set architecture (ISA) is already well suited for data-intensive communications and multimedia processing. It has numerous instructions for bit-manipulation and compare operations, plus some algorithm-specific instructions for such things as Viterbi encoding. (For a large table of the most relevant instructions, see the May 1999 *MPR* article referenced earlier.)

The SC140e adds about a dozen new instructions to the existing ISA, including some instructions that handle smaller data types in 32-bit chunks. For example, one instruction loads four eight-bit operands into four different registers in parallel—ideal for byte-sized multimedia data. However, Motorola is not publicly releasing information about all the new instructions at this time. The instructions will probably be disclosed next year, when StarCore LLC absorbs them into the future SC2000 ISA.

Like existing SC1000-based DSP cores, the SC140e supports the instruction-set accelerator plug-in, a coprocessor interface for adding application-specific instructions and registers. It allows Motorola to extend the standard ISA for its own chips or for SoCs designed to customer specifications by Motorola. Extensions can be written in Verilog or VHDL and integrated with the SC140e core (which is written in Verilog).

Extension instructions cannot directly access core registers, but they can access extension registers and use special instructions to move operands between the core and extension registers. In other respects, extension instructions behave like standard instructions—including the ability to run in parallel with standard instructions in the same VLIW bundles. Software programmers can use extension instructions in assembly language, or by defining intrinsic functions with the Metrowerks CodeWarrior ANSI C compiler or a third-party compiler. (The Green Hills *MULTI* tools support StarCore DSPs with a C/C++ compiler.)

Unlike most DSPs, the SC140e is truly designed to be programmed in C or C++, not in assembly language. It's a statically scheduled VLIW DSP with variable-length instructions, variable-length instruction bundles, 16 function units, and an ISA with nearly 200 instructions, not counting the numerous addressing variations and optional instruction prefixes. Because of the architecture's complexity, Motorola expects programmers to reserve assembly language for tight inner loops and critical routines, relying on C for the bulk of the code. Some new instructions in the SC140e are intended to improve the performance of compiled code—like the four-byte parallel load, which a compiler can substitute for four single-byte loads.

## Competition From Analog Devices and Intel

Although MMUs, address translation, memory protection, and multilevel access privileges are common features in high-performance microprocessors and microcontrollers, they are rare in DSPs. The only other similar examples are members of the Analog Devices Blackfin family, which is based on the Micro Signal Architecture (MSA) jointly developed with Intel.

The charter member of the Blackfin family was the ADSP-21535, introduced by ADI in 2001. (See *MPR 7/30/01-01*, "Intel/ADI DSP Core Gets a Home.") On paper, at least, the 21535 looks less powerful than the SC140e. It's a 16-bit fixed-point DSP with two MAC units and two ALUs with 40-bit accumulators, whereas the SC140e is a 16-bit fixed-point DSP with twice as many MAC units and ALUs, eight times as many 40-bit accumulators, and additional bit-manipulation units not found in the ADI processor. Although the 21535 has memory protection, it does not perform address translation.

However, the 21535 is highly integrated, with 308KB of on-chip SRAM, two UARTs, a PCI controller, a USB controller, an SDRAM controller, two synchronous serial ports (SPORT), and other features. In contrast, the SC140e is a soft core for SoCs, not generally licensable, initially delivered only in the Jupiter-platform modem chip from Motorola. As yet, Motorola has released few details about the Jupiter modem chip.

Intel's first MSA-based processor is an XScale-family chip, the PXA800F, announced earlier this year. (See *MPR 3/10/03-01*, "The Sand in Manitoba.") It's not strictly a DSP. Instead, it's a dual-core chip that combines an MSA core with an ARM-compatible XScale core, plus a host of integrated peripherals and memories. Although Intel is keeping some details under wraps, the MSA core in the PXA800F doesn't seem to have the MMU and multilevel privilege



Motorola's SC140e hardware and software designers are based in Texas, Israel, and Romania. This is the main design team in Israel.

modes found in the 21535 and SC140e. Presumably the XScale core will handle system-control tasks.

Nevertheless, the PXA800F appears to be a direct competitor for the SC140e-based chips that Motorola is introducing for mobile communicators. It will be interesting to compare these chips when more details become available. The comparison would be more meaningful if Motorola would release certified EEMBC benchmarks for its StarCore DSPs, which the company hasn't yet done. EEMBC scores would also make it easier to compare StarCore DSPs to the popular DSPs from Texas Instruments, although TI's chips lack the task-protection features of the SC140e.

Motorola hints that, in addition to the Jupiter modem chip, it may deliver standard-part DSPs based on the SC140e. Doing that would make sense, because the first StarCore DSP from Motorola wasn't an SoC designed for a customer but an SC140-based standard part from Motorola, the MSC8101. (See *MPR 10/6/99-03*, "First StarCore DSP Targets Networking.") The MSC8101 is a highly integrated DSP with a PowerQuicc II coprocessor, a PowerPC interface, some on-chip peripherals, and 512KB of SRAM. It would be logical for Motorola to offer a similar standard part wrapped around the SC140e.

More-exotic implementations are possible. Motorola's new MSC8122, scheduled to sample in 1H04, integrates four SC140 (not SC140e) cores, 1.4MB of SRAM, and a host of peripherals. Initially, it will run at 300–400MHz in a 90nm process.

### Missing Link: RTOS Support

Overall, the SC140e significantly improves on the SC140 without drastically altering the original core design or ISA. Preserving backward compatibility with existing software

provides an easy migration path for current customers. The enhanced SC140e is a logical bridge to the next-generation StarCore SC2000 architecture, which StarCore LLC will probably announce in late 2004. If the SC2000 adopts the SC140e's enhancements, as Motorola promises, customers can get a head start on the architectural transition today.

In network-infrastructure applications, the SC140e's ability to manage and protect multiple partitions of memory is a valuable feature. Multichannel processing will be more secure and efficient, because the SC140e and a multitasking RTOS can handle several data channels in isolation from each other, dedicating independent regions of instruction and data memory to each channel. Other memory regions can be shared if different processes need to exchange data. A channel protocol can change at run time (from, say, V.92 modem data to G.711 telephone audio) while maintaining isolation from other channels. The new memory protection, user/supervisor privilege modes, and cache-locking features all contribute to these capabilities.

If an SC140e and a DSP RTOS can assume more of the system-control functions formerly managed by a microcontroller and a regular RTOS, it might be possible, in some designs, to eliminate the microcontroller altogether. That could reduce the system's chip count, minimize the amount of software running on the system, save power, conserve board space, and cut costs.

There will probably be a learning curve, however. Microcontrollers are generally easier to program than DSPs are, and microcontroller RTOSs are more plentiful and familiar. Indeed, the idea of a data-crunching DSP that can assume the command role in a system is such a novelty that RTOS support for the SC140e's new system-management capabilities is virtually nonexistent at this time. Motorola says multiple vendors will provide DSP RTOSs that fully support the SC140e, but the vendors aren't ready to publicly announce their support until after the publication date of this article. One possible candidate is the OSEck RTOS from OSE Systems, which currently supports Motorola's SC140-based MSC8101 DSP.

Perhaps the strongest endorsement of the SC140e is Motorola's decision to use it in the new Jupiter convergence platform, which is critical to Motorola's future product line of advanced mobile communicators. In colloquial terms, the company is eating its own dog food. That should make potential customers feel a little more comfortable about choosing the SC140e for their own products: the chip vendor is the taste-tester. ◇