

FLANKED ON EITHER SIDE
OF THE PERFORMANCE,
PRICE, AND POWER
CURVE BY 8- AND 32-BIT
PROCESSORS, CAN 16-BIT
PROCESSORS SURVIVE?



PUTTING THE SQUEEZE ON 16-BIT PROCESSORS

BY ROBERT CRAVOTTA • TECHNICAL EDITOR

The start of the much-anticipated death of 16-bit processors is upon us. Because of insufficient forecasted demand, March 30, 2007, is the final date that Intel will accept orders for its MCS51, MCS251, MCS96, 80X18X, 80X38X, 80X486DXX, and i960 microprocessors; Intel will ship the last of these devices by Sept 28, 2007 (**Reference 1**). It is only a matter of time before the other semiconductor companies producing 16-bit processors follow. Or is it? Are 16-bit processors destined to become a footnote in the history books?

The much-heralded death of the 8-bit processor has failed to materialize, and it seems that almost everyone has acknowledged that 8-bit processors will continue to enjoy a robust place in the market. The market for 8-bit processors has not stagnated, but, rather, it continues to find new life as the smallest package, lowest power consumption, or lowest cost devices find their way into applications that were just a few years ago not economically or technically feasible. Two 2006 surveys queried embedded-system designers about their choice of processors. Even though the percentages and distribution of processor choice differ between each survey, each one showed growth in the number of respondents saying that they were using 8- and 16-bit processors in their designs (**references 2 and 3**).

As much as high-end 8-bit processors are adding capabilities and features that compete with low-end 16-bit processors, it is the falling edge of low-end 32-bit processors reaching down into the 16-bit price range that some people have identified as the largest threat to the long-term viability of 16-bit architectures. Tensilica's Strategic Marketing Manager Steve Leibson made just such a case last year (**Reference 4**). The main point of the case is that the cost differential between 16- and 32-bit architectures continues to shrink as part of the total system cost. The point of the case continues by pointing out that there is no feature that a 16-bit architecture can implement that a 32-bit architecture cannot also implement eventually, at a nearly zero system or silicon cost differential. This fact is the crux of countless presentations from companies pitching their 32-bit archi-

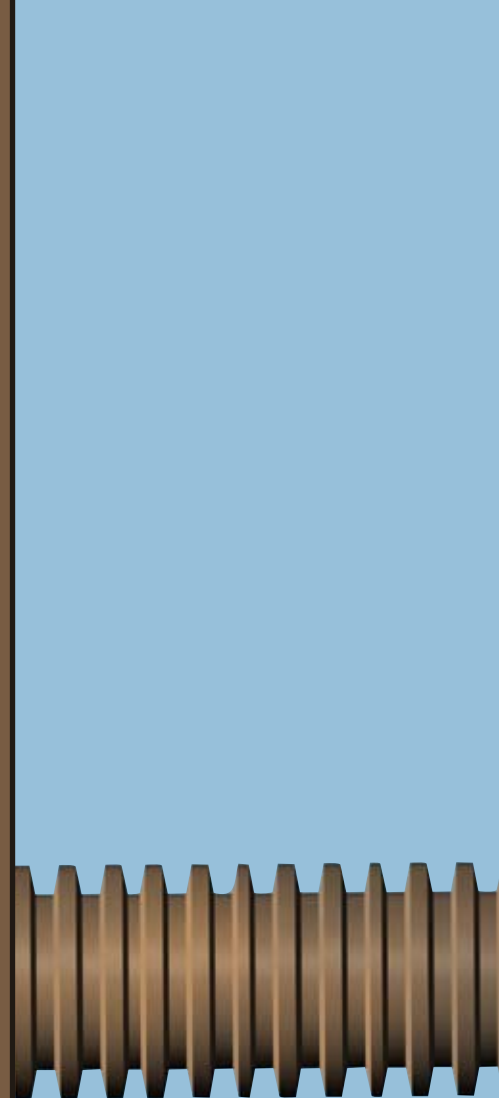
tectures—companies that do not offer 16-bit products.

The 2006 update of the *EDN* Microprocessor Directory (www.edn.com/microdirectory) lists 14 companies other than Intel that offer 16-bit microprocessors or microcontrollers to the public engineering community. *EDN's* DSP Directory (www.edn.com/dspdirectory) also includes many 16-bit DSPs. Although some of the companies with 16-bit products have not recently been publicly active with their 16-bit-product lines, many companies are continuing to invest in their 16-bit-product portfolios and are continuing to introduce 16-bit-product lines. Are these companies blindly betting millions of dollars to squeeze a little more out of a dying product line? Just what does the future hold for 16-bit processors?

At first glance, 8- and 32-bit processor architectures are squeezing out 16-bit architectures. The last few years have seen a resurgence of development effort by 8-bit-processor vendors. A previous article about 8- and 32-bit products explored why 8-bit processors best cover the low end of embedded applications in price, power, and package size (**Reference 5**). That article focuses on why 8-bit processors can fend off increasingly lower cost 32-bit devices. This article expands on the processing sweet spots that last year's processing-options article proposed (**Reference 6**).

THE SQUEEZE

Will 8- and 32-bit processors do an adequate job of overlapping and completely covering the applications that 16-bit processors best serve? To enable their 8-bit architectures to continue to support expanding requirements in leg-



acy 8-bit designs, 8-bit-processor manufacturers have been expanding the capabilities and features of their 8-bit architectures. Notable expansions are a larger addressable-memory map and higher clock rates. Although these devices may be able to emulate 16-bit data types, they do so at a performance and power penalty because they must perform multiple chained operations on the low and then the high byte of the operands that 16-bit architectures would otherwise implement as single instructions. A 16-bit processor operates faster and often at lower power than an 8-bit device for applications with processor-intensive 16-bit mathematical tasks.

This performance and power penalty to process wider data types is a driver for migrating from 8-bit to wider 16- or 32-bit architectures. This point is a key transition enabling 32-bit processors to compete with 16-bit processors; it is also central to the claim that 16-bit devices are doomed. If a design team must port its design to a new architecture, why not port it to one that has much less risk of requiring another port in the near future? In this case, weighing the size of the risk and the cost of performing another port becomes a trade-off in deciding whether to use an optimized 16-bit implementation or to skip to an adequate 32-bit implementation with room to grow. For

AT A GLANCE

- ▣ More embedded-system designs continue to include 8- and 16-bit processors.
- ▣ Some semiconductor companies are making serious efforts to supplant the 16-bit-processor market with lower end 32-bit devices.
- ▣ Just as 8- and 32-bit architectures are evolving, so too are 16-bit devices to remain competitive.
- ▣ Lower cost and power consumption will continue to be differentiators between 16- and 32-bit processors.
- ▣ A 16-bit sweet spot is any application that couples power consumption with data-intensive processing or peripheral-task-intensive control.
- ▣ Portable or distributed monitoring and control applications are emerging opportunities for 16-bit architectures.

designers using soft cores in FPGAs, the choices are limited (see sidebar “FPGAs’ 16-bit options”).

In addition to more complex control functions, other drivers pushing 8-bit legacy design to consider porting to larger architectures include added functions, especially network connectivity; the need to drive larger displays; and the desire or need to use an RTOS (real-time

operating system) for task management. TCP/IP (Transfer Control Protocol/Internet Protocol) network stacks require a hefty memory footprint, and there are few sources of TCP/IP stacks for 8- and 16-bit architectures. CMX Systems supports TCP/IP networking on 8- and 16-bit systems, but to support the smallest footprints, the stacks omit some protocol support. There is stronger RTOS support for 32-bit architectures than for 16-bit ones, mainly because systems implemented on 32-bit devices tend to be more complex and better benefit from RTOS support.

Last year, Luminary Micro introduced ARM Cortex-M3-based 32-bit processors at prices as low as \$1. This price places the processors squarely in competition with many 8- and 16-bit processors. In addition to Luminary Micro, Atmel and NXP offer low-cost ARM7-based processors that include features to ease the port effort from 8-bit designs. Some of these features, which are uncommon in 32-bit offerings, include bitwise manipulation, brownout detection, and power-on reset. The inclusion of these “8-bit” features in these devices plays to the point of 16-bit doomsayers that 32-bit devices can outplay the 16-bit architectures.

Instruction size is another example of 32-bit architectures’ ability to play along

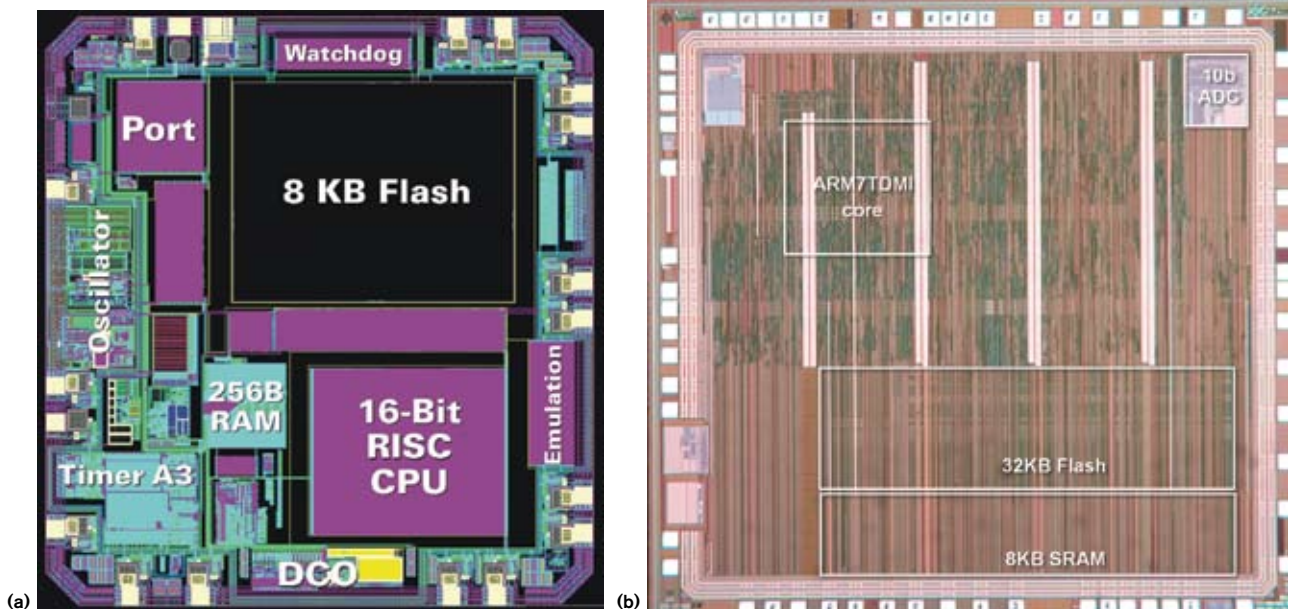


Figure 1 These die shots of a 16-bit (a, courtesy Texas Instruments) and a 32-bit (b, courtesy NXP) processor illustrate the relative silicon area of the processor core and on-chip memory (not to scale with each other).



with the advantages of 16-bit architectures. In general, 16-bit ISAs (instruction-set architectures) provide significantly better code density than purely 32-bit ISAs, and this feature manifests itself in a design as smaller requirements for program memory. The impact of the advantage becomes apparent when you compare the silicon area of any 8-, 16-, or 32-bit-processor core and the memory in many systems today (**Figure 1**). This advantage is pronounced enough that many contemporary 32-bit ISAs include a 16-bit instruction subset. The Cortex-M3 ISA goes a step further, supporting only the 16-bit Thumb-2 ISA. The inclusion of a 16-bit instruction subset in 32-bit architectures substantially reduces the code-density advantage that 16-bit devices enjoyed.

THEM'S FIGHTIN' WORDS

Just as 8- and 32-bit architectures are including features that let them better compete in the 16-bit-application area, so too have 16-bit architectures been changing. For example, some modern 16-bit processors, such as Freescale's MC9S12XE 16-bit microcontrollers, break the traditional 64-kbyte address-space limitation with support for a 1-Mbyte or more linear addressing space with no paging. The memory-to-memory addressing and 16 single-cycle, 16-bit registers in Texas Instruments' MSP430 family are examples of a modern 16-bit

architecture's addressing the accumulator bottleneck and limited register space in older 16-bit implementations. The 16- and 8-bit processors are including embedded debugging circuitry to assist developers.

RTOS support can help simplify programming and the process of migrating from 8- to 16-bit microcontrollers. The options have been growing for those 16-bit designs that can benefit from an RTOS. RTOS support for 16-bit architectures has lagged behind that for 32-bit architectures, but it is happening. In addition to processor-vendor-provided kernel support, RTOSs supporting 16-bit architectures are available from third-party sources, such as CMX, Mentor Graphics (Nucleus), Micrium (μ C/OS-II), and FreeRTOS. To better support 16-bit RTOSs, these processors may include special on-chip registers, such as Fujitsu's User and System stack pointers, to provide additional RTOS support.

Memory protection is an important part of many embedded RTOSs, but 16-bit processors typically address system protection differently from 32-bit processors with an MPU (memory-protection unit). One example is Microchip's CodeGuard security that enables OEMs to divide and share three segments of on-chip memory with tiered levels of security for the boot, secure, and general segments. This segmentation allows design houses or algorithm ven-

dors to protect proprietary software in secure memory segments and permit a range of applications to access the algorithm from the other segments. Other examples of some system-protection or fail-safe features in 16-bit architectures are mechanisms to protect against accidental writes to flash program memory, traps for stack overflows, an independent clock source for watchdog timers, backup oscillators, and power brownout and power-on-reset supervisors. However, MPUs are not limited to 32-bit processors; Freescale's MC9S12XE 16-bit microcontroller family includes an integrated MPU.

Despite Intel's exit from the 16-bit-embedded-processor market, more than a dozen semiconductor companies still have active 16-bit-product lines. Many of them have processor-product lines that span 8-, 16-, and 32-bit architectures. Wayne Chavez, Freescale's automotive-product-marketing manager, points out, "16-bit processors are still a source of growth, and Freescale's processor strategy is one of intelligent overlap." Many of the semiconductor companies that have processor products that span all of the data widths share this sentiment. In general, these companies place a lot of value on providing designers the ability to scale up and down for the best price, performance, and peripheral mix by preserving the programming model and tools across processor architectures. Ultimately, the main concern is that one of their processors gets the design-in, and providing a scalable choice is part of their strategy to accomplish this goal.

Another concept, virtualization, is undergoing a resurgence from the old mainframe days. At a basic level, it means allowing the software-development model to not worry about whether it resides on a single processor or on multiple cores. It can enable a hard RTOS to operate with a general-purpose operating system on the same device. However, "Most embedded-software development still relies on an intimacy with the metal for best cost and efficiency," says Ryan Scott, marketing manager for industrial and multimarket microcontrollers at Infineon. Predictable or deterministic performance, lower power consumption, and a BOM (bill-of-materials) cost difference of

FPGAs' 16-bit options

Most FPGA companies offer and support 8- and 32-bit soft cores, but none currently support a 16-bit core. Mike Thompson, senior IP (intellectual-property)-product-marketing manager at Actel, says that the company periodically receives requests to support a 16-bit core but notes, "There is not enough traction to justify the additional cost to add and maintain a third architecture." Bob Garrett, senior marketing manager for Nios marketing at Altera, acknowledges that the company's Nios 1.0 core was originally a 16-bit core but that the currently supported configurations of the core are 32 bits.

As a side note, FPGAs implemented alongside 8- or 16-bit processors in general act as external logic, a high-speed state machine, or a bridge for emerging niche functions. Power consumption is a significant challenge facing FPGAs in applications using these smaller processor architectures. According to Ritesh Tyagi, director of marketing for the System LSI Business Unit at Renesas Technology America, states, "As microcontrollers integrate hard-block implementations of these emerging functions, the new microcontrollers are replacing the FPGA."



even a penny still matters for embedded-system designs that 8- and 16-bit processors best suit.

ESCAPING THE SQUEEZE

Low cost and low power consumption are important differentiators for 8-bit processors. Unsurprisingly, the same situation is true for 16-bit processors, especially when design teams perform a trade-off to migrate their design between 16- and 32-bit options. Designs that target 16-bit processors in general use less memory than those designs that would target a 32-bit processor. Despite the diminishing silicon differential between a 16- and a 32-bit core relative to the entire system, the appropriate amount of adjacent memory for each core size limits how close the price of these devices can converge.

In many embedded-system applications, especially those suited to a smaller architecture width, there is often a significant opportunity for lower power consumption through low-power sleep and standby modes. However, increasing leakage current—the power consumption when the processor is on but idle—negates the advantages that smaller process geometries provide in silicon area for converging the cost between 16- and 32-bit processors. Clever clock gating does not get around the leakage-current penalty. The only way to avoid it is to use slower, less leaky transistors or to be able to completely shut down and provide power to the circuit on a demand basis, further increasing the complexity and cost of the 32-bit device. Just as with performing 16-bit arithmetic with an 8-bit processor, a mismatch of data size and processor width can adversely affect the amount of overall processing the core can perform and the amount of power it takes to perform those tasks when compared with an architecture width that matches the data width.

There is no 8051 or ARM equivalent for the 16-bit-processor market. This fact may play heavily in the assumptions foretelling the death of 16-bit architectures; however, this issue may not be significant because 16-bit devices should be able to stave off 32-bit processors and defend their existence with configurations that are highly tuned for specific

MORE AT EDN.COM

⊕ For a related article about processing-architecture sweet spots, go to www.edn.com/article/CA6298267.

⊕ To learn about 8- versus 32-bit processors, go to www.edn.com/article/CA502421.

⊕ Go to www.edn.com/070215cs and click on Feedback Loop to post a comment on this article.

applications. The question, then, could be: What are those applications? Currently, automotive and industrial-control applications are major market segments for current 16-bit processors.

A processing sweet spot for 16-bit devices is any application that couples sensitivity with power consumption and data-intensive processing or peripheral-task-intensive control that 8-bit products cannot as efficiently provide. Such applications include metering for electricity, water, and gas, as well as handheld products, such as small point-of-sale products for taking orders in restaurants, retail stores, or service establishments.

In recent years, a number of semiconductor companies have built hybrid processors, which the industry increasingly refers to as DSCs (digital-signal controllers). These devices combine characteristics of DSPs and microcontrollers in a single instruction stream. Generally, DSCs can continuously feed the arithmetic unit without stalling, and they can perform rapid context switching for peripheral-control functions. These devices also commonly employ intelligent or autonomous peripheral capabilities to avoid starving the arithmetic engine. Most of these devices employ a 16-bit architecture.

Emerging applications that 16-bit architectures and DSCs are well-positioned to support include portable, home-based medical equipment; portable monitoring equipment; and distributed industrial-monitoring and -control equipment, such as for smart buildings. Most of these applications will also support wireless- or wired-network connectivity. Only time will tell whether the 16-bit doomsayers will relent as the 8-bit doomsayers have done in recent years. **EDN**

REFERENCES

- 1 Intel Product Discontinuation Notice (PDN) number 106013-01.
- 2 Ganssle, Jack, "What Processor Is In Your Product?" *Doctor Dobb's Journal*, Oct 2, 2006, www.ddj.com/dept/embedded/193101234.
- 3 The William Baldwin Group, "Embedded Engineer Survey '06: Volume 2, What Engineers Do," September 2006, <https://william-baldwin.com/contents.htm> and www.eg3.com.
- 4 Leibson, Steve, "Rest-in-peace: The 16-bit processor," *Electronics Weekly*, April 27, 2006, www.electronicweekly.com/Articles/2006/04/27/38406/Rest-in-peace+The+16-bit+processor.htm.
- 5 Cravotta, Robert, "Reaching down: 32-bit processors aim for 8 bits," *EDN*, Feb 17, 2005, pg 31, www.edn.com/article/CA502421.
- 6 Cravotta, Robert, "Processing options," *EDN*, Jan 19, 2006, pg 42, www.edn.com/article/CA6298267.

FOR MORE INFORMATION

Actel
www.actel.com

Altera
www.altera.com

ARM
www.arm.com

Atmel
www.atmel.com

Ceva
www.ceva-dsp.com

CMX Systems
www.cmx.com

FreeRTOS
www.freertos.org

Freescale
www.freescale.com

Fujitsu
www.fujitsu.com

IAR Systems
www.iar.com

Infineon
www.infineon.com

Intel
www.intel.com

Lattice Semiconductor
www.latticesemi.com

Luminary Micro
www.luminarymicro.com

Mentor Graphics
www.mentor.com

Micrium
www.micrium.com

Microchip
www.microchip.com

National Semiconductor
www.national.com

NEC Electronics
www.necel.com

NXP Semiconductors
www.nxp.com

Renesas
www.renesas.com

Tensilica
www.tensilica.com

Texas Instruments
www.ti.com

Toshiba
www.toshiba.com

Xilinx
www.xilinx.com

