Guest Editor’s Introduction

Digital Signal Processing Microprocessors: Forward to the Past?

L. Robert Morris
Carleton University and DSPS Inc., Ottawa

In 1986, a number of new digital signal processing microchips were announced by semiconductor manufacturers whose names are familiar to IEEE Micro’s readers: Analog Devices, Motorola, National Semiconductor, NEC, Texas Instruments, and Philips/Signetics. What are DSP micros? What makes them different from the general-purpose micros offered by many of the same companies? What are their applications? Why should IEEE Micro’s readers be interested in such devices? This brief introduction and the articles which follow will attempt to answer most of these questions.

DSP micros share one feature: speed in the execution of certain algorithms. As was first noted in a 1983 survey,1 these processors are, in effect, reduced-instruction-set computers optimized for the fastest possible execution of addition, subtraction, multiplication, and shifting instructions. In early DSP micros especially, a reduced instruction set, which can be implemented in a small area of silicon, is accompanied by single-cycle multiplication and shifting, which are accomplished by devoting a relatively large area of silicon to an array multiplier and a barrel (or combinatorial) shifter. In contrast, most current general-purpose micros still effect such operations via multiple-cycle, microcoded instructions that make use of the arithmetic unit’s single-cycle, parallel-add and single-bit shift capability. Since integer multiplication and shifting are statistically unimportant for most programs that run on general-purpose micros, designers of such devices prefer to devote large areas of silicon to implementation of larger, more versatile instruction sets (sometimes including floating-point in the on-chip microcode), memory management, or cache memories.

The implication of the above—that fast integer multiplication and shifting are considered crucial to digital signal processing software—is correct. In fact, the software implementation of the most common digital signal processing algorithm, an n-tap finite-length impulse response (FIR) filter, essentially consists of n multiply/accumulates. These instructions are executed once for every signal sample that is input (at rates typically of 8 kHz and above). Most of the newer DSP micros can accomplish each multiply/accumulate in a single cycle of about 100 ns! This is one to three orders of magnitude faster than most general-purpose micros. For example, a 16-MHz 80386—a state-of-the-art micro which effects register-to-register 16-bit addition (ADD) in only 125 ns—requires about 1250 ns for a 16 x 16-bit multiplication (IMUL), and a 5-MHz 8088 requires 32,000 ns for the same instruction! Other important DSP algorithms—the fast Fourier transform, or FFT, for example—require many more addition/subtractions than multiplications, but even for these algorithms the relatively slow multiply on general-purpose processors represents a significant bottleneck.

The first DSP micro, the Intel 2920, appeared nearly a decade ago. It was followed by the AMD 2811, the NEC μPD7720, and, in 1982, the Texas Instruments TMS32010. While the 2811 and 7720 both had on-chip array multipliers, both were ROM-programmable only and had relatively small data and program address spaces. The 32010 was the first DSP micro that could execute instructions at full speed from an off-chip program RAM, and it could also accommodate a program nearly an order of magnitude larger than the 7720 could.

The articles in this issue will reveal both similarities and differences between DSP and general-purpose micros. For example, DSP micros employ many speed- and efficiency-related design strategies also employed in regular micros: pipelining of instructions, use of addressing modes that efficiently access relevant data structures (e.g., autoincrement and autodecrement modes for arrays and an indexed ad-
dressing mode for FFTs), and use of “clean” subroutine calling and address passing protocols. Differences include DSP micros’ use of the dual-bus Harvard architecture, which enables simultaneous fetching of instructions and data; special DSP-related addressing modes (e.g., index computation modulo an arbitrary number, automatic circular queue or free data move for FIR filters, and bit reversal for FFTs); extra addressing ALUs; and special interfaces to serve specific fields of application (e.g., serial interfaces for codecs in telecommunications).

How were DSP algorithms implemented in the pre-DSP micro era? In the early 1970’s, array processors—first fixed-point and then floating-point—were available for real-time execution of many audio-bandwidth DSP algorithms. These machines varied in cost from $10,000 to $50,000 and typically consisted of a rack-mounted unit weighing upwards of 100 lbs. and consuming about a kilowatt of power. These attributes generally limited the use of array processors to large laboratories and certainly precluded the inclusion of such machines as subcomponents in OEM systems. The data in Table 1 reveals an interesting fact: the 1976 array processors and the 1986 DSP micros have comparable FFT execution times, and the same holds for 1976 general-purpose minicomputers and 1986 general-purpose processors.
micros! Thus, today's DSP and general-purpose micros exhibit approximately the same performance as their decade-old ancestors. Further, comparisons of their architectures do not reveal that any startling changes have occurred since 1976.

What has occurred, of course, is a three-order-of-magnitude reduction in cost, size, weight, and power consumption. It is the combination of 1976 array processor performance with 1986 microchip attributes that has both quantitatively and qualitatively changed the extent to which theory can be applied to the practical solution of problems in signal processing, communications, and control, and in new disciplines such as artificial intelligence. In many cases, the computer simulation traditionally carried out as a precursor to system realization via hardwired logic can now become the cost-effective implementation via software on a DSP micro.

DSP micros are now on the verge of surpassing their array processor ancestors in architectural complexity and sophistication as well as in performance. Thus, the theme finally becomes "Forward to the Future." VLSI allows active device densities and signal propagation times not possible a decade ago. And, fortunately, semiconductor technologies have not yet hit a "brick wall" in terms of speed. Gallium arsenide (GaAs) transistors and high-electron-mobility transistors (HEMTs) in particular suggest that another "easy" order-of-magnitude improvement in performance is not unreasonable to anticipate, even with existing architectures. Although DSP devices having parallel and dataflow architectures have appeared, at present they have not achieved the user acceptance of more conventional "sequential" processors. This is partially due to the fact that the present DSP micro user anticipates that performance enhancements requiring neither changes to algorithms nor even changes to software will continue to appear due to clock-speed-related semiconductor progress alone!

We should discuss one other possible scenario.1 Note that the fastest general-purpose micros already approach the performance of the slowest DSP micros: a 16-MHz 80386 computes a 1K, complex, fixed-point FFT only 66 percent slower than a 20-MHz TMS32010 (see Table 1 again). With the newest versions of general-purpose micros already incorporating a DSP-like dual bus architecture (for example, the Motorola 680302), the obvious next step—integration of an array multiplier and a barrel shifter into general-purpose micros—cannot be far off. Since these two devices make possible fast floating-point multiplication and addition, respectively, and since floating-point performance "sells," most semiconductor manufacturers are on the verge of taking this step.

Although the resulting general-purpose micros will still lack some special instructions and architectural attributes that help in achieving maximum DSP performance, it is entirely conceivable—with GaAs technology already commercially viable in 19863,4—that by incorporating GaAs/HEMT transistors they can achieve a performance of 100 MIPS and upwards and make special-purpose DSP micros unnecessary in many DSP applications. 

References


L. Robert Morris is a professor of systems and computer engineering at Carleton University in Ottawa and is also president of DSPS Digital Signal Processing Software, Inc. His areas of interest include DSP algorithm/DSP micro architecture interactions, signal processing, and time/space optimization of programs for DSP micros. He is currently involved in a multiuniversity Canadian project for developing a DSP-micro-based, multichannel cochlear implant. He obtained a BASc in electrical engineering from the University of Toronto and a PhD in speech communications from the University of London (Imperial College), England.

Morris can be reached at Systems Engineering, MacKenzie Bldg., Room 377, Carleton University, Ottawa, Ont. K1S 5B6, Canada.

Reader Interest Survey

Indicate your interest in this article by circling the appropriate number on the Reader Interest Card.

High 195  Medium 196  Low 197