MICROPROCESSOR REPORT
VOLUME 12, NUMBER 16
DECEMBER 7, 1998
THE INSIDERS' GUIDE TO MICROPROCESSOR HARDWARE

TigerSHARC Sinks Teeth Into VLIW

Analog Devices' High-End DSP Challenges Texas Instruments, StarCore

by Ole Wolf and Jeff Bier,
Berkeley Design Technology, Inc.

Keeping pace with the accelerating march of architectural innovation in DSPs, Analog Devices (ADI) unveiled its third-generation floating-point DSP, dubbed TigerSHARC, at the recent Microprocessor Forum. There, architect Jose Fridman described a complex, high-performance VLIW-based design incorporating unusually extensive single-instruction, multiple-data (SIMD) capabilities. Unlike its predecessors, which are primarily aimed at applications demanding floating-point arithmetic, TigerSHARC has excellent fixed-point capabilities and is better described as a 16-bit fixed-point DSP with floating-point support than as a floating-point DSP.

SHARC-Infested Waters

In recent years, Analog Devices has been the standard-bearer for floating-point DSPs. Motorola and Lucent Technologies have dropped out of floating-point DSPs entirely, and Texas Instruments has let its floating-point products lie fallow while focusing on higher-volume fixed-point devices. ADI, however, has had significant success with its SHARC family of floating-point DSPs. Although high-performance general-purpose processors are often faster than DSPs on floating-point DSP tasks, SHARC offers developers an attractive mix of DSP-oriented features, including a large amount of on-chip SRAM, extensive interprocessor-communication support, and solid DSP application-development infrastructure.

Earlier this year, TI announced its first new floating-point architecture in eight years, the high-performance TMS320C67xx (see MPR 9/14/98, p. 18). A VLIW-like design, the 'C67xx is based on the fixed-point 'C62xx. The 167-MHz dual-multiplier 'C67xx, which began sampling in September, is far faster than ADI's existing ADSP-2106x SHARCs, which top out at just 60 MHz.

ADI quickly countered with the announcement of its second-generation floating-point architecture, the ADSP-2116x, dubbed Hammerhead. The '2116x promises to boost SHARC clock rates to 100 MHz, and it increases throughput via the addition of a second set of execution units. The second set of units can be used only in a SIMD fashion, in lock-step with the first set. BDTI's benchmarks indicate that Hammerhead doesn't approach the performance of the 'C67xx. Hammerhead does, however, maintain assembly source-code compatibility with the first-generation SHARC. (For maximum performance, though, code must be rewritten to use the SIMD features.) The first Hammerhead devices were initially expected to sample in November, but sampling has now slipped into early 1999.

Despite the fact that its second-generation silicon isn't yet available, ADI is charging ahead with its third-generation TigerSHARC, predicting that initial samples will be available by mid-1999 at a speed of 250 MHz. As Table 1 shows, TigerSHARC's projected floating-point multiply-accumulate throughput is significantly better than that of previous floating-point DSPs. In addition, its projected 16-bit fixed-point multiply-accumulate throughput is impressive: an order of magnitude faster than that of its ADI predecessors and four times faster than that of the TI 'C62xx at the same clock rate. TigerSHARC is scheduled to sample earlier than the Lucent/Motorola StarCore 440 (see MPR 10/26/98, p. 22), but nonetheless its fixed-point multiply-accumulate throughput is projected to be almost twice as fast. TigerSHARC's complex architecture will pose new challenges for programmers and code-generation tools, however.

Vendor	Analog Devices			Lucent/Mot	Texas Instruments	
Device	'2106x	'2116x	N/A	N/A	'C62xx	'C67xx
Nickname	Sharc	Hammerhead	TigerSharc	StarCore 440	VelociTI	VelociTI
Availability	1994	1Q99	mid-99	1Q00	1Q97	3Q98
Compatibilty w/Predecessor	Object code	Assembly source	None	None	None	Binary w/ 320C62xx
Architecture	DSP	DSP+SIMD	VLIW+SIMD	VLIW	VLIW	VLIW
Clock Speed	60 MHz	100 MHz	250 MHz	300 MHz	250 MHz*	167 MHz
16b MACs/s	60 million	200 million	2,000 million	1,200 million	500 million	333 million
FP MACs/s	60 million	200 million	500 million	–	–	333 million

Table 1. Key attributes of all three generations of ADI's Sharc DSPs compared with Lucent/Motorola's forthcoming offering and TI's high-performance DSPs. *Currently available TMS320C62xx devices run at 200 MHz. (Source: vendors)

The Third Generation

TigerSHARC is aimed at telecommunications infrastructure applications, such as cellular telephone base stations and xDSL central-office equipment. But, as of this writing, ADI is withholding the details on the first TigerSHARC device and has specified details of only the TigerSHARC core.

As illustrated in Figure 1, the TigerSHARC architecture contains a program control unit, two computation units, two address generators, memory, various peripherals, and a DMA controller. With its VLIW architecture, TigerSHARC is capable of executing up to four instructions in a single cycle, and its SIMD features enable it to perform arithmetic operations on multiple 32-bit floating-point values or multiple 32-, 16-, or 8-bit fixed-point values.

Two Sets of Teeth

TigerSHARC's two computation units can be either treated independently or used in a SIMD fashion by a single instruction. The two computation units each contain a multiplier, an ALU, and a shifter. These three function units use 32- or 64-bit inputs and support both floating-point and fixed-point data. Each computation unit has an associated register file with thirty-two 32-bit data registers. For 64-bit inputs, two 32-bit registers are concatenated to form one 64-bit value. Data transfers and some outputs from the multiplier can be up to 128 bits wide; 128-bit destinations are formed by concatenating four consecutive 32-bit registers.

With two computation units that can optionally be controlled by a single instruction, TigerSHARC is a SIMD machine, where a single instruction can specify an arithmetic operation that is carried out on several data elements. However, two separate computation units are not enough for TigerSHARC. Employing an unusual hierarchical SIMD approach, TigerSHARC treats each register as either one 32-bit IEEE-754 floating-point value or one 32-bit, two 16-bit, or four 8-bit fixed-point values. Although increasingly common in general-purpose and embedded processors, such data-type agility is unusual among DSP processors and will be valuable as DSPs are increasingly called upon to handle diverse media types, for example, images and video as well as audio.

Each of TigerSHARC's computation units can perform two 32 × 32 → 64-bit fixed-point multiply-accumulates in a single cycle, using two operands each made up of two concatenated 32-bit registers. Thus, using both computation units, TigerSHARC can perform four 32 × 32 → 64-bit fixed-point multiply-accumulate operations in a single cycle. Alternatively, TigerSHARC can perform two 32-bit floating-point MAC operations per cycle.

In fixed-point DSP applications, the most common word width is 16 bits. With four 16-bit fixed-point elements concatenated in two 32-bit registers, one computation unit can in a single cycle perform four 16 × 16 → 32-bit multiply-accumulate operations (with 8 guard bits each to avoid overflow)—twice as many as any currently available fixed- or floating-point DSP can perform. If an instruction uses both computation units, it can perform eight 16 × 16 → 32-bit multiply-accumulate operations in a single cycle—four times as many as any current DSP.

The fact that TigerSHARC uses SIMD features at two levels—two separate computation units that each operate on SIMD operands—is unusual and potentially confusing. Figure 2 illustrates how the two SIMD computation units divide registers into different data sizes.

TigerSHARC is the first of the new wave of VLIW-based DSPs to provide extensive SIMD capabilities. This approach provides greater parallelism than that of its StarCore and Texas Instruments competitors, but at a cost. Some applications cannot make effective use of the data parallelism offered by SIMD operations. In addition, even for applications that can use SIMD, TigerSHARC programmers will face a new level of complexity when optimizing code.

Voracious Data Appetite

To feed its dual computation units, TigerSHARC uses two data address generators, called integer ALUs, and two 128-bit data buses to transfer up to 256 bits of data per cycle between the computation units and memory. Each address generator has a 32-entry address register file and is capable of modulo addressing and bit reversal.

Data is transferred between the computation units and on-chip memory in blocks of 32, 64, or 128 bits. The two 128-bit data buses deliver 8 Gbytes/s on a 250-MHz TigerSHARC; this bandwidth corresponds to sixteen 16-bit data words per cycle.

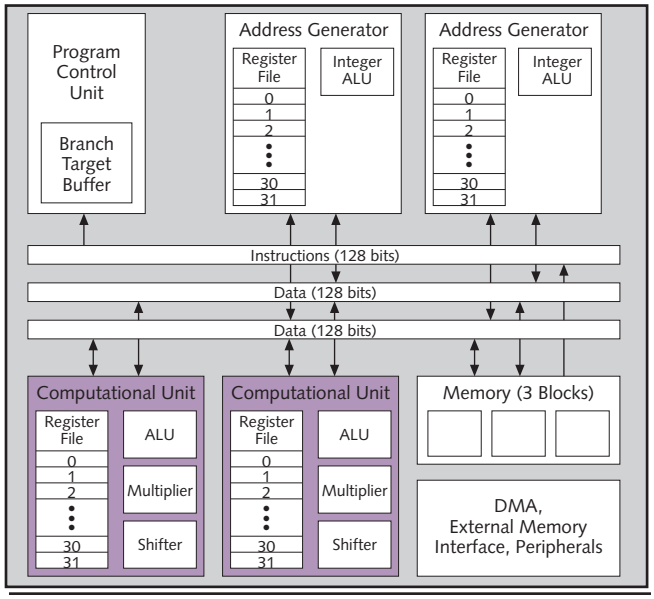

Figure 1. TigerSHARC's two computation units are connected to on-chip memory via two data buses that can transfer a total of 256 bits of data per cycle. Addresses are provided by two address generators. Address buses are not shown.

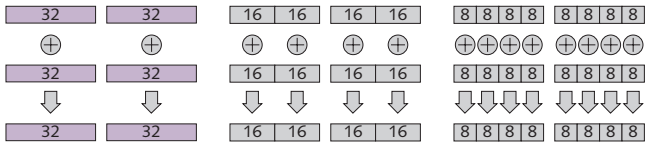

Figure 2. Each of the two TigerSHARC computation units operates on up to two 32-bit register pairs, where each register is treated as one 32-bit, two 16-bit, or four 8-bit fixed-point values. Alternatively, each computation unit can operate on 32-bit floating-point operands in non-SIMD fashion.

On-chip memory is divided into three banks: one for software and two for data. ADI will not disclose the amount of on-chip memory in the first TigerSHARC devices, but we expect that the vendor will continue to be generous with on-chip memory; the predecessor SHARC and Hammerhead devices include 68K to 512K of on-chip memory.

When moving 64-bit or 128-bit data, TigerSHARC transfers data from consecutive memory locations to consecutive data registers, or vice versa. The smallest amount of data that can be transferred is 32 bits. If TigerSHARC programs use word sizes of 8 or 16 bits in a DSP algorithm, they cannot access individual words; any load or store will transfer at least four 8-bit or two 16-bit words.

The chip includes a data alignment buffer and a short data alignment buffer that allow 64 or 128 bits of data to be transferred from (but not to) any memory location aligned on a 16-bit word boundary. TigerSHARC provides more flexibility than most processors with SIMD features, which often require that data be aligned at memory locations divisible by the size of the data transfer.

The chip does not, however, provide the kind of powerful data-shuffling instructions found in Motorola's AltiVec extensions for PowerPC (see MPR 5/11/98, p. 1), nor can it conditionally operate on only some subwords of a SIMD operand.

Chip in a Bottle?

TigerSHARC's on-chip memory bandwidth is impressive, to be sure, but at some point applications must communicate with the world outside the processor's pins. ADI has declined to provide details on the external memory interface of initial TigerSHARC devices, except to say that the off-chip memory bandwidth will be 800 Mbytes/s. This bandwidth is comparable to that of TI's current 'C6xxx devices. But this may not be adequate; because TigerSHARC's compute bandwidth and on-chip memory bandwidth are significantly greater than those of the 'C6xxx devices, TigerSHARC's external memory interface is at risk of becoming a severe performance bottleneck.

Like previous SHARC devices, TigerSHARC will sport 14 channels of DMA. Because the external memory interface is rather slow compared with the chip's high compute bandwidth, TigerSHARC will have to rely heavily on DMA to help applications make maximum use of its limited off-chip memory bandwidth. In addition, program execution from on-chip memory will be vital, because the external memory can't keep up with the processor's instruction consumption rate. If TigerSHARC features as much on-chip memory as its predecessors, however, there should be little need for executing from external memory.

The company also declined to describe TigerSHARC's on-chip peripherals or multiprocessor capabilities. We expect that ADI will keep to its tradition, including enhanced versions of the peripherals found on '2106x and '2116x devices. Similarly, we expect that it will continue to provide extensive multiprocessor I/O support.

Slippery for Programmers

Recent processor designs by Texas Instruments and by the StarCore partners indicate that VLIW architectures are here to stay for DSP. Following its competitors' lead, TigerSHARC employs a VLIW architecture that can fetch and issue up to four 32-bit instructions in a single cycle. Two instructions can be issued to the same computation unit, enabling, for example, arithmetic operations to proceed in parallel with shift operations in each computation unit.

TigerSHARC breaks compatibility with its two SHARC predecessors. The new chip uses an algebraic instruction set, shown in Table 2, that has the same flavor as the original SHARC instructions. Programmers accustomed to coding for earlier SHARC devices should quickly become comfortable with the TigerSHARC instruction set. But, based on our own experience with Hammerhead and other SIMD architectures, we expect that programmers will find it difficult to use TigerSHARC's two computation units and hierarchical SIMD features efficiently.

Instructions are selected for parallel execution by the assembly-language programmer or the compiler. All TigerSHARC

Description	B	S	N	L	F
ALU Operations					
Add, Subtract	●	●	●	●	●
Add and Subtract	●	●	●	●	●
Add w/Carry or Subtract w/Borrow	●	●	●	●	
Absolute Value of Sum or Difference	●	●	●	●	●
Accumulate Absolute Sum/Diff	●	●			
Average	●	●	●	●	●
Sideways Sum	●	●			
Minimum or Maximum	●	●	●	●	●
Increment or Decrement	●	●	●	●	
Compare	●	●	●	●	●
Clip (saturate)	●	●	●	●	●
Absolute Value or Negate	●	●	●	●	●
Logical (and, or, xor, not, not-and)			●	●	
Expand (convert to higher precision), Expand Sum or Diff	●	●	●		
Compact (convert to lower precision), Compact Sum or Diff		●	●		
Merge	●	●			
Count Ones			●	●	

Description	B	S	N	L	F
Conversion					
Fixed Point to Floating Point with Optional Scaling			●	●	●
Floating Point to Fixed Point with Optional Scaling			●		●
Extract Mantissa or Exponent			●		●
Shifter/Bit Manipulation					
Logical or Arithmetic Shift	●	●	●	●	
Rotate			●	●	
Field Deposit/Extract			●	●	
Apply Mask			●	●	
Bit Set, Clear, Toggle, or Test			●	●	
Exponent Detection		●	●		
Count Leading Zeros and Ones			●		
Multiply/Divide					
Multiply		●	●		●
Multiply-Accumulate		●	●		
Complex Multiply-Accumulate		●			
Reciprocal Seed, Reciprocal Square-Root Seed					●

Description	B	S	N	L	F
Load/Store					
Normal Load/Store			●		●
Long Load/Store				●	
Quadword Load/Store	128 bits				
Flow Control					
Jump	—				
Jump to Subroutine	—				
Return from Subroutine or Interrupt Svc Routine	—				
System					
NOP	—				
Enter Low-Power Mode	—				
Invalidate Branch Target Buffer	—				
Trap	—				

Table 2. TigerSHARC instruction set summary (not exhaustive). B, S, N, L, and F indicate the operands supported: B = 8-bit ("byte"), S = 16-bit ("short"), N = 32-bit ("normal"), L = 64-bit ("long"), F = 32-bit (floating point).

instructions can be predicated. Unlike the 'C6xxx pipeline, TigerSHARC's pipeline is fully interlocked, meaning that unexpected results will not be produced as a result of pipeline effects. The TigerSHARC pipeline has eight stages, and although it is fully interlocked, it is quite visible to the programmer. All additions, multiplications, and data-load operations have single-cycle throughput but a two-cycle latency.

For efficient DSP software, these latencies mandate software pipelining and loop unrolling, and they also complicate programming. In addition, these techniques increase the number of 32-bit instructions required to optimize a DSP task for speed, inflating program-memory usage compared with that of previous SHARC chips. This is a common problem among VLIW architectures, and TigerSHARC code is likely to be more compact than 'C6xxx software.

Branches nominally take three or six cycles to execute, and, due to TigerSHARC's interlocking pipeline, no instructions can execute while a branch instruction is pending. To ease this problem, TigerSHARC predicts branches with a 128-entry four-way set-associative branch target buffer (BTB). This enables the processor to effectively perform zero-overhead branching, provided that the direction and target address of the branch are correctly predicted. In addition, TigerSHARC provides a zero-overhead hardware-looping mechanism.

Tools Will Be Critical

TigerSHARC tools will consist of ADI's VisualDSP tool suite, including a cycle-accurate instruction-set simulator. The vendor states that tools are currently available to lead customers and will be broadly available in 1H99. ADI also plans to make available libraries of optimized DSP building-block functions.

With TigerSHARC's VLIW style of instruction execution and multilevel SIMD operations, few programmers will be willing and able to write optimized assembly code for the new architecture. Therefore, ADI's C compiler will be a pivotal element in the success of TigerSHARC. Compared with TI's 'C67xx, TigerSHARC's pipeline effects are more benign, but its SIMD operations and data-type agility pose serious challenges to efficient compiler code generation. ADI has plans to provide a vectorizing C compiler, but the company will face significant challenges in developing a truly efficient compiler capable of handling TigerSHARC's VLIW and SIMD architecture. Without such a compiler, though, application developers will be hard pressed to realize TigerSHARC's performance potential.

MICHAEL MUSTACCHI

Analog Devices' Jose Fridman describes the VLIW features in the new TigerSHARC DSP.

A Sea Change

Just two years ago, conventional DSP architectures were firmly entrenched, with the dominant DSP vendors focusing on minor architectural tweaks, increasing clock speeds, and on-chip integration to boost perform[ance and functionality. Today, all four major DSP processor vendors have committed to VLIW-oriented designs for the high ends of their product lines.

ADI's customers may initially be confused by the rapid emergence of two new generations of SHARC processors—one compatible with its predecessor, the other not. Despite surface appearances, though, the product lineup works well. The second-generation Hammerhead allows early SHARC users to significantly improve performance without completely rewriting their code. TigerSHARC, in contrast, promises leading-edge performance for customers willing to take the plunge with a completely new architecture.

Like the other recently introduced VLIW-based DSPs, TigerSHARC promises impressive performance gains over its predecessors but brings with it many new challenges. Primary among these will be the compilers and other code-generation tools, since the unprecedented complexity of the new architecture makes it difficult to access its performance potential through traditional hand-optimized assembly code. Although TigerSHARC's massive SIMD capabilities offer dazzling levels of parallelism and unprecedented data-type agility, ADI must deliver outstanding software tools—particularly compilers—to enable users to access the processor's capabilities. If ADI can deliver the necessary tools, TigerSHARC's performance should satisfy the most demanding fixed- and floating-point applications. M

Authors Ole Wolf and Jeff Bier are with Berkeley Design Technology, Inc., the DSP technology analysis and software development firm. Wolf and Bier are co-authors of Buyer's Guide to DSP Processors*, which is available from MDR.*

For More Information

For additional information about TigerSHARC, please visit *www.analog.com/dsp.* Initial TigerSHARC devices are scheduled to sample to customers in 1H99. ADI has not yet disclosed on-chip memory and peripheral configurations, pricing, packaging, or power consumption for these devices.