

Tensilica White Paper

**Low-Power, Low-Overhead, High-Fidelity
Digital Sound for SOCs:**

Tensilica's HiFi 2 Audio Engine

May, 2007

Executive Summary

High-quality audio creates an immersive experience that excites buyers and spurs the purchase of consumer products such as home theater and PC sound systems, flat-panel televisions, handheld and console video games, portable music and video players, and mobile telephone handsets. As a result, digital audio has rocketed to the top of the critical features list for all sorts of products over the past several years. At the same time, the number of digital-audio codecs (coders and decoders) and audio-enhancement programs has exploded and most consumer products must support multiple codecs and offer a broad range of audio-enhancement features. All of these factors have resulted in a high demand for a flexible, high-performance, low-power audio engine that adds sound to an SOC's design with the least amount of design effort and a small on-chip footprint. Tensilica's HiFi 2 Audio Engine was carefully crafted to meet these requirements for the broadest possible range of consumer products. The HiFi 2 Audio Engine is available as an audio-extension package for the Xtensa LX2 configurable processor core and incorporated into the pre-configured Diamond 330HiFi processor core (both processor cores are fully synthesizable).

Why use programmable SOC hardware for audio?

The SOCs developed for the original digital music players didn't use programmable MP3 decoders, they used dedicated hardware because that approach resulted in the lowest gate count. However, for products that must handle multiple digital-audio codecs, the need for multiple hardware codecs quickly grows the gate count and a programmable approach becomes far more attractive. Consequently, nearly all new SOC designs incorporating audio employ some form of programmable engine to run the audio codecs.

In addition to supporting multiple audio codecs, the programmable approach reduces design risk because future changes in either the codec standard or in the product's definition can be accommodated through a firmware change instead of a silicon spin. In addition, the programmable nature of a processor-based approach makes it far easier to perform application-specific pre- and post-processing such as sample-rate conversion, multi-band frequency equalization, speaker virtualization, and parsing (audio content extraction from a file using various container formats) to the SOC's design.

Why not just use a RISC processor or DSP for audio?

General-purpose processors are often used to implement audio codecs because these processors can do anything that can be expressed in a program given enough execution cycles. With a sufficiently fast clock, general-purpose processors can implement multiple audio codec algorithms and they are therefore suitable for multipurpose devices that must accommodate several digital-audio standards. However, general-purpose processors are not especially efficient engines for running audio codecs, which means that they must run

at higher clock rates to deliver real-time, multichannel audio. In turn, the higher clock rates cause these processors to consume more energy than do processors with audio-specific instruction sets. Higher power dissipation presents a real problem for portable, battery-powered devices.

Because they have features that accelerate the execution of audio-specific code such as hardware MACs, DSPs generally run audio codecs more efficiently than do general-purpose processors. However, DSPs have specialized and irregular architectures that make them poor targets for C compilers and therefore unattractive for implementing control-oriented tasks. Consequently, DSPs (and closely related application-specific, DSP-based data engines such as ARM's AudioDE) must be paired with a general-purpose control processor to create a design platform with sufficient capability and flexibility to be used for today's consumer products.

Dual-processor configurations that split the audio-processing duties across more than one processor or a processor and a hardware offload block create their own design problems because the audio-processing tasks must be apportioned to each processor and the audio operations running on each processor must be coordinated, which increases software-development complexity because the two different processors (usually a general-purpose RISC processor and a DSP) must be programmed with dissimilar software-development tools and use of different programming tools further complicates software development.

System developers implementing audio codecs and other audio-processing software in SOCs have another design alternative: application-tailored RISC processors, which start out as general-purpose processors capable of running any software program. Processor tailoring can add audio-specific extensions to the basic RISC processor architecture including new instructions, registers, and data types. These extensions accelerate code execution for the target audio applications (codecs and sound-shaping algorithms) and similar applications while allowing the processor to remain a good C compiler target. Tensilica's HiFi 2 Audio Engine is a second-generation set of processor extensions specifically designed for digital-audio applications.

The HiFi and the improved HiFi 2 Audio Engines

Tensilica's successful, first-generation HiFi Audio Engine, based on the company's configurable, 32-bit Xtensa V processor core, was an example of a configurable processor that had been optimized for audio applications with extra instructions and registers to accelerate the execution of audio-codec software. The original HiFi Audio Engine was designed into mobile phones and personal media players and these products are now in production.

With the subsequent introduction of the Xtensa LX and LX2 processor architectures, it became possible to create an improved, second-generation set of audio extensions that deliver even better audio performance by exploiting the added parallelism delivered by the Xtensa LX architecture through flexible-length instruction extensions (FLIX) and by

the associated, vectorizing C/C++ compiler (XCC), which automatically exploits the added parallelism made available through the FLIX hardware extensions. FLIX instruction extensions allow processors based on the Xtensa LX architecture to issue multiple independent operations each clock cycle. Like all Xtensa instructions, FLIX instructions are first class citizens of the processor’s instruction set and are recognized and used by all of the processor’s software-development tools including its C/C++ compiler, assembler, debugger, and instruction-set simulator.

Tensilica’s HiFi 2 Audio Engine extensions comprise a set of more than 300 new audio-specific operations paired with two new audio-specific register files: an 8-entry file named “P” with 48 bits/entry (each entry can hold two 24-bit audio values) and a 4-entry file named “Q” with 56-bit entries. The 56-bit values in the Q register file are generated from a set of instructions that control a dual multiplier/accumulator (MAC). Each of the two pipelined multipliers can perform a 24x24-bit or a 32x16-bit multiplication with a throughput of one multiplication per multiplier per cycle. The 56-bit results obtained from the two multipliers are accumulated in the Q registers. (Note: the 32x16-bit operation mode for the multipliers is particularly helpful for high-precision arithmetic.)

300 Audio-Specific HiFi 2 Instructions

The 300 HiFi 2 operation extensions are grouped as shown in Table 1.

HiFi 2 Operations	
1	Loads and Stores
2	Single/Dual Multiply with 56-bit Accumulator
3	Scalar and 2-Way SIMD ALU Operations
4	Variable/Immediate Shifts
5	Convert/Round/Truncate/Saturate Operations
6	Huffman Encode/Decode and Bit-Stream Support

Table 1: The HiFi 2 Audio Engine’s Five Operation Groups

- Group 1 includes the operations that load data from and store data to the P and Q register files. These operations support immediate and indexed addressing modes with and without automatic address register updating.
- Group 2 instructions drive the HiFi 2 Audio Engine’s dual MAC. Audio codec software uses these instructions to perform audio-stream transforms between the time and frequency domains and for windowing and frequency-band splitting, sample-rate conversion, and special audio effects such as reverb and three-dimensional sound simulation.

- Group 3 operations perform scalar arithmetic and Boolean functions on 56-bit words stored in the Q register file as well as 2-way SIMD arithmetic and Boolean functions on the 48-bit (paired 24-bit) words stored in the P register file.
- Group 4 instructions include shift operations used for normalization, which maximizes the dynamic range of the HiFi 2 Audio Engine's fixed-point calculations.
- Group 5 operations include conversion, rounding, truncation, and saturation functions for the P and Q register files. Operations in this group also permit data exchanges between the P and Q register files and the processor's base 32-bit register file.
- Group 6 includes Huffman encoding and decoding operations (used for noiseless coding and decoding) and bit-stream support operations (used for efficient bit stream packing and unpacking).

Very Un-RISC-Like but a Good Compiler Target

Adding 300 audio-specific instructions to a RISC processor makes the compiled target application code very efficient—which was a primary goal in the design of the HiFi 2 Audio Engine—but it is a very un-RISC-like concept and it would be prohibitively expensive, essentially impractical, to perform such application-specific processor tailoring without using an automated tool like Tensilica's Xtensa Processor Generator for constructing the processor cores and associated software-development tools.

A detailed examination of all 300 HiFi 2 Audio Engine operations in the five operation groups is available in the technical documentation but a quick look at the HiFi 2 Audio Engine's MAC operation group illustrates the flexibility of the Xtensa LX architecture's FLIX-format instructions. Figure 1 illustrates the instruction format of the HiFi 2 Audio Engine.

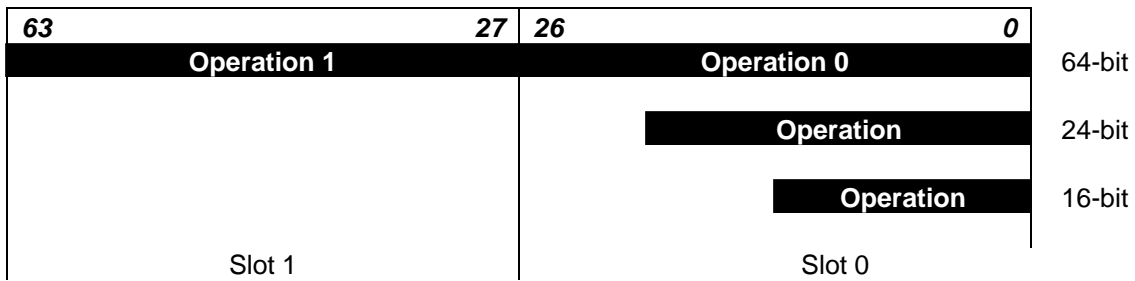


Figure 1: The HiFi 2 Audio Engine’s 16-, 24-, and 64-bit Instruction Formats

The HiFi 2 Audio Engine’s MAC instructions can include the following functions and very complex instructions can be built from these MAC primitives:

- Single or dual multiplication
- Fractional or integer arithmetic
- 24x24-bit, 16x16-bit, or 32x16-bit operands
- Overwrite, add, or subtract accumulation with or without saturation
- Signed or unsigned arithmetic

The base Xtensa LX RISC instructions use Slot 0 and are either 16 or 24 bits wide. The HiFi 2 Audio Engine instructions generally use Slot 0 for its load and store operations and for a few audio-processing DSP instructions that access to the processor’s base 32-bit register file. The HiFi 2 Audio Engine uses Slot 1 for most DSP operations.

Processors based on the HiFi 2 Audio Engine and the Xtensa LX architecture modellessly handle variable-width instruction streams. This feature allows the associated C/C++ compiler’s code generator to select and freely intermix differently sized instructions to minimize the size of the compiled code. The compiler automatically selects the smallest instruction that will perform the required operation, which results in very compact code and avoids the code bloat generally associated with multi-operation VLIW processors.

Figure 2 provides a simplified view the HiFi 2 Audio Engine’s register files and datapath, organized visually to match the 2-slot FLIX instruction format shown in Figure 1. The 16- and 24-bit instructions share the base RISC processor’s execution hardware (registers and pipeline) with operations in the HiFi 2 Audio Engine’s lower operation slot (called Slot 0). The rest of the HiFi 2 Audio Engine’s datapath extensions occupy Slot 1.

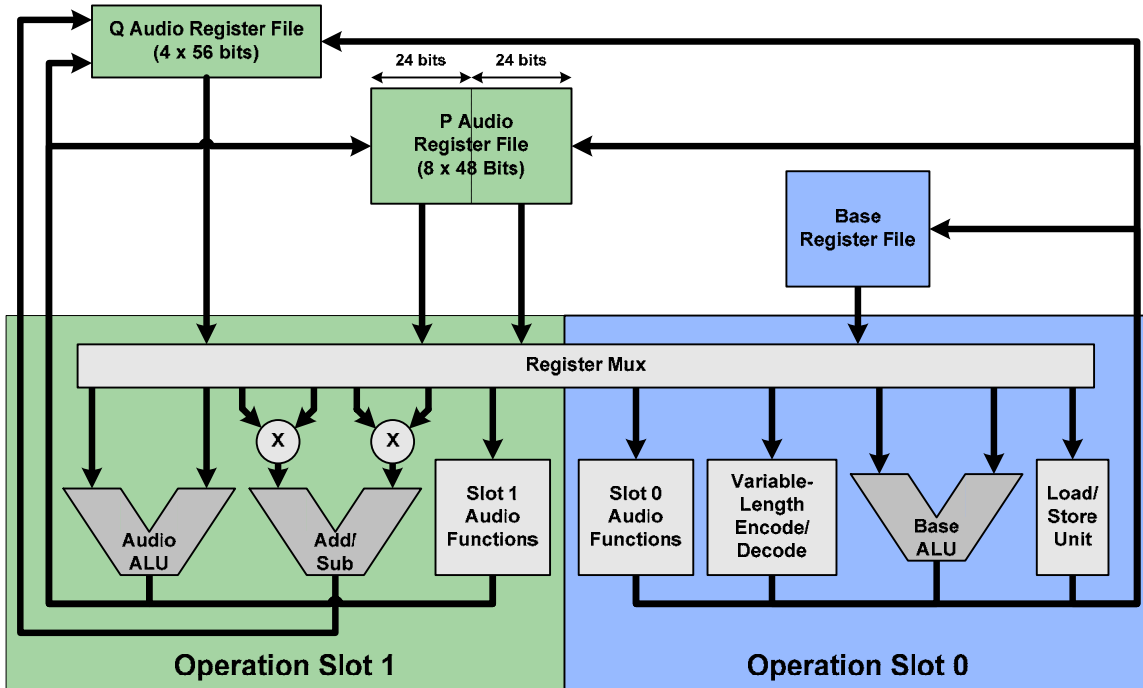


Figure 2: HiFi 2 Audio Engine Register Files and Datapath

The base RISC processor's and the HiFi 2 Audio Engine's 16- and 24-bit instructions only control resources assigned to Slot 0 while the HiFi 2 Audio Engine's 64-bit instructions can control all the processor resources in both slots. For 64-bit instructions, the operation residing in the lower part of the 64-bit instruction word controls processor resources assigned to Slot 0 and the operation in the upper part of the instruction word (labeled Operation 1 in Figure 1) controls resources assigned to Slot 1. It's important to note that processor cores based on the HiFi 2 Audio Engine architecture have two operation slots but there is only one processor, which keeps the programming model simple and allows the audio-enhanced processor to be a good target for the C/C++ compiler.

Experiments Guide HiFi 2 Audio Engine Development

The HiFi 2 Audio Engine’s designers experimented with various extensions and performance results with different software codecs guided architectural development. Because the Xtensa Processor Generator produces a tailored software-development tool suite for each new set of extensions in minutes, the designers found it was relatively easy to play “what if” games with the target application code and to objectively assess various extension alternatives along several simultaneous design dimensions.

For example, designers considered several alternative designs for the HiFi 2 Audio Engine’s MAC before finalizing the design. The HiFi 2 Audio Engine’s multipliers could have been designed to be capable of 24x24- or 32x16-bit operations and the final MAC unit could have been designed with one or two multipliers. A configuration allowing two 32x16-bit multipliers to be used as one 32x32-bit multiplier was also considered. Experiments with different MAC configurations produced the results shown in the Table below.

	Maximum Clock Rate (MHz)	Gate Count	Area (mm ²)
Single 24x24-bit MAC	299	88,569	0.98
Dual 24x24-bit MAC	289	100,860	1.12
Dual MAC supporting 24x24- and 32x16-bit operations	284	101,408	1.13
Dual MAC supporting 24x24-, 32x16-bit operations or a single 32x32-bit operation	270	110,012	1.22

Based on TSMC 130nm LV process, Artisan library, includes MUL32 Xtensa LX configuration option not used in the final HiFi 2 design

Table: HiFi 2 MAC options and experimental results

Based on these results, the designers of the HiFi 2 Audio Engine selected a dual-MAC configuration capable of 24x24-bit and 32x16-bit operations because this configuration provided the best audio-codec performance (the HiFi 2 Audio Engine’s MAC instructions are especially good for complex multiplications and for FIR filters) without seriously compromising the synthesized processor core’s maximum clock rate. Note that this maximum clock rate is many times higher than the actual clock rate needed to run any one of the software audio codecs. This processing headroom allows the HiFi 2 Audio Engine to handle several other audio and control tasks on the SOC while also running one or more audio codecs concurrently.

Industry's Broadest Line of Audio Codecs

Most consumer products need to run more than one audio codec (although not necessarily at the same time). Table 3 lists all of the proven audio codecs and other audio packages available for Xtensa and Diamond processor cores based on the HiFi 2 Audio Engine. Because each of these audio packages requires very little processing bandwidth, processor cores based on Tensilica's HiFi 2 audio engine can run these audio codecs at very low clock rates to save power. Alternatively, the processor can run several of these audio packages simultaneously at somewhat higher clock rates. The long list of audio software packages in Table 3 already leads the industry and Tensilica is committed to keeping this lead through the introduction of new codecs and other audio application software when needed.

All of the codecs listed in Table 3 are written in C. In fact, a primary design goal for the HiFi 2 Audio Engine was to create an audio platform that could be programmed in C while delivering the desired real-time performance at low processor clock rates. This approach opens software development to a much larger programming audience than for other vendors' audio solutions (which must be programmed in assembly language to achieve performance goals). Because many more programmers are familiar with C than with assembly-language coding, SOC design teams can draw on the much larger base of C/C++ programmers by using Tensilica's cores based on the HiFi 2 Audio Engine.

Audio Packages for the HiFi 2 Audio Engine
Xtensa HiFi 2 Dolby Digital AC-3 Decoder
Xtensa HiFi 2 Dolby Digital AC-3 Consumer Encoder
Xtensa HiFi 2 Dolby Digital Plus Consumer Decoder
Xtensa HiFi 2 MP3 Decoder
Xtensa HiFi 2 MP3 Encoder
Xtensa HiFi 2 MPEG-4 aacPlus v2 Decoder
Xtensa HiFi 2 MPEG-4 aacPlus v2 Encoder
Xtensa HiFi 2 MPEG-4 aacPlus v1 Decoder
Xtensa HiFi 2 MPEG-4 aacPlus v1 Encoder
Xtensa HiFi 2 MPEG-2/4 AAC LC Decoder
Xtensa HiFi 2 MPEG-4 AAC LC Encoder
Xtensa HiFi 2 Ogg Vorbis Decoder
Xtensa HiFi 2 QSound microQ
Xtensa HiFi 2 SONiVOX Sound that Rocks
Xtensa HiFi 2 SRS TruSurround HD
Xtensa HiFi 2 SRS Xspace 3D
Xtensa HiFi 2 WMA Decoder
Xtensa HiFi 2 WMA Encoder
Xtensa HiFi 2 AMR Narrowband Speech Codec
Xtensa HiFi 2 AMR Wideband Speech Codec
Xtensa HiFi 2 G.729AB VoIP Speech Codec

Table 3: Currently available audio packages for processor cores based on Tensilica's HiFi 2 Audio Engine

HiFi 2 Audio Engine's Extensive Low Power Features

Through a combination of significantly lower power per MHz and architectural optimization of the HiFi 2 Audio Engine instruction set, the HiFi 2 Audio Engine delivers dramatic improvements in energy efficiency that result in increased battery life (more playing time) for portable and wireless applications. Based on TSMC's 65 nm LP process and a minimal HiFi 2 configuration including memory, dynamic and static power dissipation can be as low as 66 $\mu\text{W}/\text{MHz}$ and 69 μW respectively. Total power dissipation is only 0.45 mW while decoding a typical MP3 file at 5.7 MHz. For the Diamond 330HiFi core, power dissipation ranges from 0.089 to 0.352 mW/MHz in TSMC's 0.13G process and typical power dissipation is 0.148 mW/MHz in TSMC's 90 nm G process.

Tensilica's Xtensa LX2 configurable processor core and Tensilica's Diamond 330HiFi Audio Engine, which is based on the Xtensa LX2 core, have many features that facilitate low-power operation including functional clock gating and a variety of power-down and sleep modes. Lower power and energy consumption is a key reason for using configurable processor cores and instruction-set extension to run audio codecs. By adding appropriately tailored instructions to the processor's ISA, the configured processor executes the target application code in many fewer cycles. As a result, the processor core can execute the codec at a much reduced clock frequency, which in turn cuts both power dissipation and energy consumption.

Xtensa and Diamond processor cores have two levels of clock gating. The first level of clock gating is based on global conditions. For instance, the WAITI option allows an Xtensa processor to enter a sleep mode that turns off the clocks to nearly all of the processor's internal registers. An interrupt wakes the processor from sleep mode. In addition, the processor's RunStall signal can still be used to save power by allowing external logic to stall the processor pipeline and turn off the clock to many of the processor's registers.

The processors' second level of clock gating is functional clock gating. Xtensa and Diamond processor cores contain *hundreds* of functional blocks, identified through trillions of simulation cycles exercising all of the processor's instructions. Through analysis of these simulations, Tensilica's processor development team determined when each functional block needed to be clocked and when the block could be quiescent. As a result, all processor cores based on Tensilica's Xtensa LX2 architecture (including Diamond cores) dynamically clock only the needed functional units on a clock-by-clock basis. This approach achieves the lowest possible dynamic power and energy dissipation. For maximum power savings, both levels of clock gating should be used.

Lower Gate Count

Through careful engineering and by fully exploiting the features of the Xtensa LX architecture, Tensilica was able to implement the Xtensa HiFi 2 Audio Engine in approximately 44K gates when targeted to 200 MHz.

Performance to Spare

For high-performance applications, processor cores based on Tensilica's Xtensa LX2 processor core and the HiFi 2 Audio Engine can attain clock rates exceeding 320 MHz in TSMC's 90 nm G process, leaving plenty of headroom for other control, media, and signal-processing tasks. The efficient Xtensa HiFi 2 Audio Engine architecture requires only a small fraction of this available processor bandwidth to perform the audio decoding and encoding functions.

Industry's Broadest Range of Interfacing Alternatives

Most SOC audio solutions are bus centric because buses represent a conventional and well-understood approach to designing on-chip interconnect. All processors based on Tensilica's Xtensa LX2 and HiFi 2 Audio Engine architectures provide a range of bus-type interconnections including the Xtensa LX2 core's Processor Interface (PIF) bus, an AHB adapter for the PIF, and local-memory buses.

In addition to these options, the SOC design team can use the unique features of the Xtensa LX2 architecture's FIFO-queue interfaces. Instruction extensions push data from the processor onto external queues, pop data from external queues into processor state registers or register files, or use the information from input queues directly in computations. These dedicated FIFO-queue interfaces provide direct connections between the processor core's internal execution pipeline and standard FIFO memories using a conventional 3-wire handshake. FIFO-based I/O is not a shared resource and is therefore immune to uncertain latencies caused by the inevitable contention encountered with shared-bus system architectures.

Queue interfaces allow Xtensa and Diamond processor cores to directly interface to external FIFO queues that have standard push- and pop-type interfaces and full-and-empty flow control. Compressed, streaming audio data can enter the audio processor via an input queue and decoded audio samples can be sent to the system's output DACs via an output queue. Because of this, FIFO-queue interfaces are good I/O choices in system designs that incorporate audio because the queue interfaces remove the continuous flow of audio data from other bus traffic, freeing valuable system-bus bandwidth at a very low hardware cost.

Component Audio for SOCs

Most SOC design teams working with on-chip audio simply need to add audio as a “drop-in” component. Audio is certainly one of the SOC’s important features but design teams generally need to add value by spending development time and resource on other product-specific features. Digital audio has become sufficiently standardized so that it can now be added to the SOC design as an off-the-shelf component. However, the only way to add audio as an SOC component is to select a complete, ready-to-use audio solution.

There are many facets to measuring the completeness of an audio solution. Tensilica’s HiFi 2 Audio Engine and Diamond 330HiFi audio processor core are complete solutions that offer:

- The industry’s broadest range of audio and voice codecs. Tensilica is committed to continually adding the latest audio and speech codecs needed by SOC design teams.
- Low-power operation. The HiFi 2 Audio Engine’s architecture allows existing and future codecs to operate at low clock rates and therefore at low power and with reduced energy consumption. Some competing audio solutions are optimized for one specific codec, say MP3, but perform poorly when running other audio codecs. Tensilica’s HiFi 2 Audio Engine with its 300 audio instructions and audio-specific register files is designed to run all audio and speech codecs efficiently.
- Easy design in. Many factors determine how easy it will be to design a vendor’s audio solution into an SOC. Among these are the flexibility of the core, the interface flexibility of the audio block, simulation support, and programming support.
 - Tensilica’s hybrid RISC/audio-DSP approach provides more flexibility than any other in the industry. A Diamond 330HiFi processor core or a configurable Xtensa LX2 core with HiFi 2 Audio Engine extensions can run Tensilica’s long list of supported audio and speech codecs as well as new codecs and, critically, any compiled C/C++ program. There simply is no more flexible approach.
 - All Tensilica processor cores offer a variety of interface options including conventional buses and FIFO-queue interfaces, which are unique in the industry. FIFO queue interfaces are extremely efficient relative to bus-based interfaces and never suffer from contention (as do shared buses), which makes FIFO-queue interfaces ideal for SOC audio applications.
 - All Tensilica processor cores incorporating HiFi 2 Audio Engine extensions come complete with an instruction-set simulator (ISS) that can perform in variety of system-simulation environments including SystemC. In addition, Tensilica offers a fast functional ISS called TurboXim that is

40 to 80 times faster than a conventional ISS. TurboXim is so fast that it runs popular audio codecs such as the MP3 decoder faster than real time.

- All Tensilica processor cores are accompanied by a comprehensive suite of programming and debugging tools carefully tailored to the processor. XCC, Tensilica's vectorizing C/C++ compiler, understands all of the parallelism built into the HiFi 2 Audio Engine architecture including the dual MACs, so the compiler's code generator produces very compact, highly efficient code that achieves performance goals at low processor clock rates. As a result, audio codecs and other audio software can be completely developed in C, which opens the coding task to a wider range of programmers and permits more rapid software development.

An Open Invitation

If your design team is responsible for incorporating audio features into your next SOC design, contact Tensilica for assistance. For more information on the unique abilities and features of the HiFi 2 Audio Engine, see www.tensilica.com, email sales@tensilica.com, or contact Tensilica directly at:

Tensilica, Inc.
3255-6 Scott Blvd.
Santa Clara, CA 95054
Phone: (408) 986-8000
FAX: (408) 986-8919