tensilica®

xtensa

PRODUCT BRIEF

## Xtensa Processor Core

A configurable, extensible and synthesizable processor core, Tensilica® 's Xtensa® processor is the first microprocessor architecture designed specifically to address embedded System-On-Chip (SOC) applications. It was designed from the start to be a configurable architecture enabling designers to tailor each implementation to match the application requirements for the target SOC.

The Xtensa processor is unlike any other conventional embedded processor core – it changes the rules of the SOC game. Using Xtensa technology, the system designer molds the processor to fit the application by selecting and configuring predefined elements of the architecture and by inventing completely new instructions and hardware execution units that can deliver performance levels orders of magnitude faster than alternative solutions. The Xtensa Processor Generator also automatically generates a complete optimized software environment – including operating system support – for each processor configuration. The power and flexibility of the configurable Xtensa processor make it the ideal choice for all complex SOC designs.

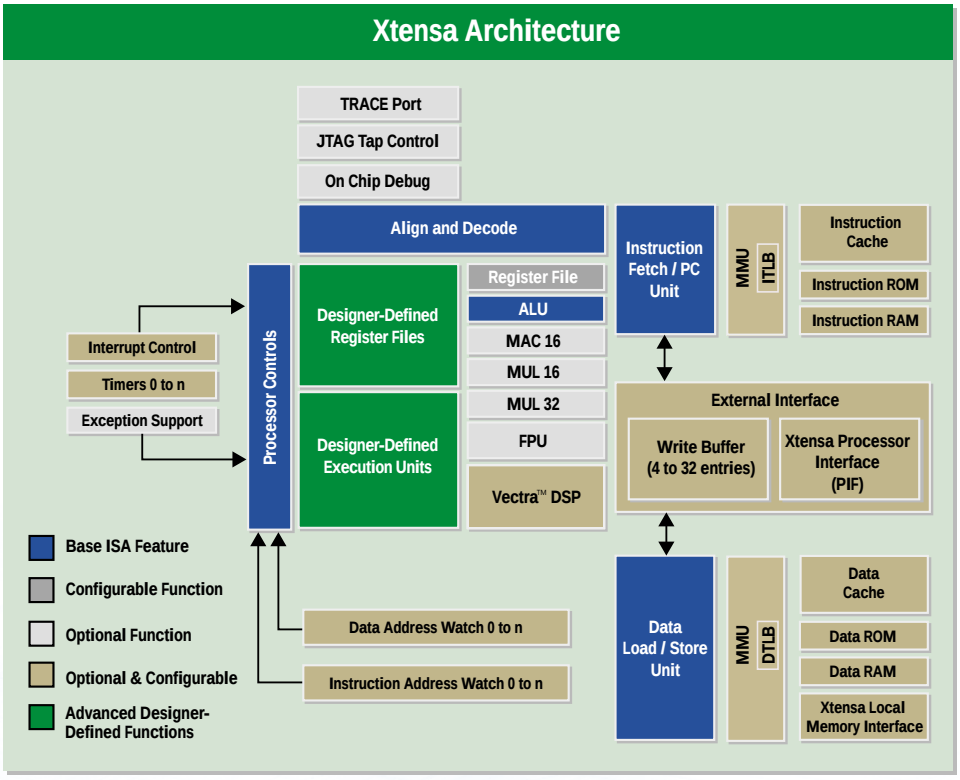



***Figure 1.*** *The Xtensa architecture consists of various configurable building blocks. CONFIGURABLE FUNCTION blocks are elements parameterized by the system designer. OPTIONAL FUNCTION blocks indicate elements available to accelerate specific applications. OPTIONAL AND CONFIGURABLE blocks are optional elements scalable to fit applications, including peripherals. ADVANCED DESIGNER-DEFINED FUNCTIONS are hardware execution units and registers added to the processor by the designer to accelerate specific algorithms for a given SOC design. Common in all configurations is the Base Instruction Set Architecture.*

## Unleash the Potential of System-On-Chip Designs with a Configurable Processor Solution

SOC design can provide higher performance, lower cost, smaller form factor and longer battery life through lower power. But these designs often experience bottlenecks in block-level integration and verification, hardware/software co-development and semiconductor portability. Until now, embedded SOC designers have had to develop solutions based around rigid processor cores that were designed for workstation architectures. The Xtensa solution provides a configurable microprocessor core that is quickly integrated with other system blocks and is easily adapted to the needs of today's high volume, high-performance embedded applications. The Xtensa core is particularly well suited for digital consumer, networking, office automation and wireless embedded SOC applications.

## The Power of a Configurable RISC Core

The Xtensa 32-bit architecture features a compact instruction set optimized for embedded designs. The base architecture has a 32-bit ALU, up to 64 general-purpose physical registers, 6 special purpose registers and 80 base instructions, including improved 16- and 24-bit (rather than 32-bit) RISC instruction encoding. The Xtensa processor's advanced architecture allows designers to achieve significant code size reductions that result in higher code density and better power dissipation – key to saving cost in a highly integrated SOC ASIC. The Xtensa core's 16-and 24-bit encoding also provides powerful branch instructions such as combined compare-and-branch and zero-overhead loops, and bit manipulations including funnel shifts and field-extract operations.

## Create an Optimal Processor Core in Hours — RTL, Modeling Support, Software Chain

Profile the application software, configure the microprocessor core and add new instructions to optimize performance – all in a matter of hours.

The Xtensa Processor Generator assists designers in creating tailored, application-specific embedded processors quickly and reliably. Before committing to silicon, system designers can explore multiple architectures by making area, speed, power and code-density design tradeoffs based on real-time feedback from the processor generator.

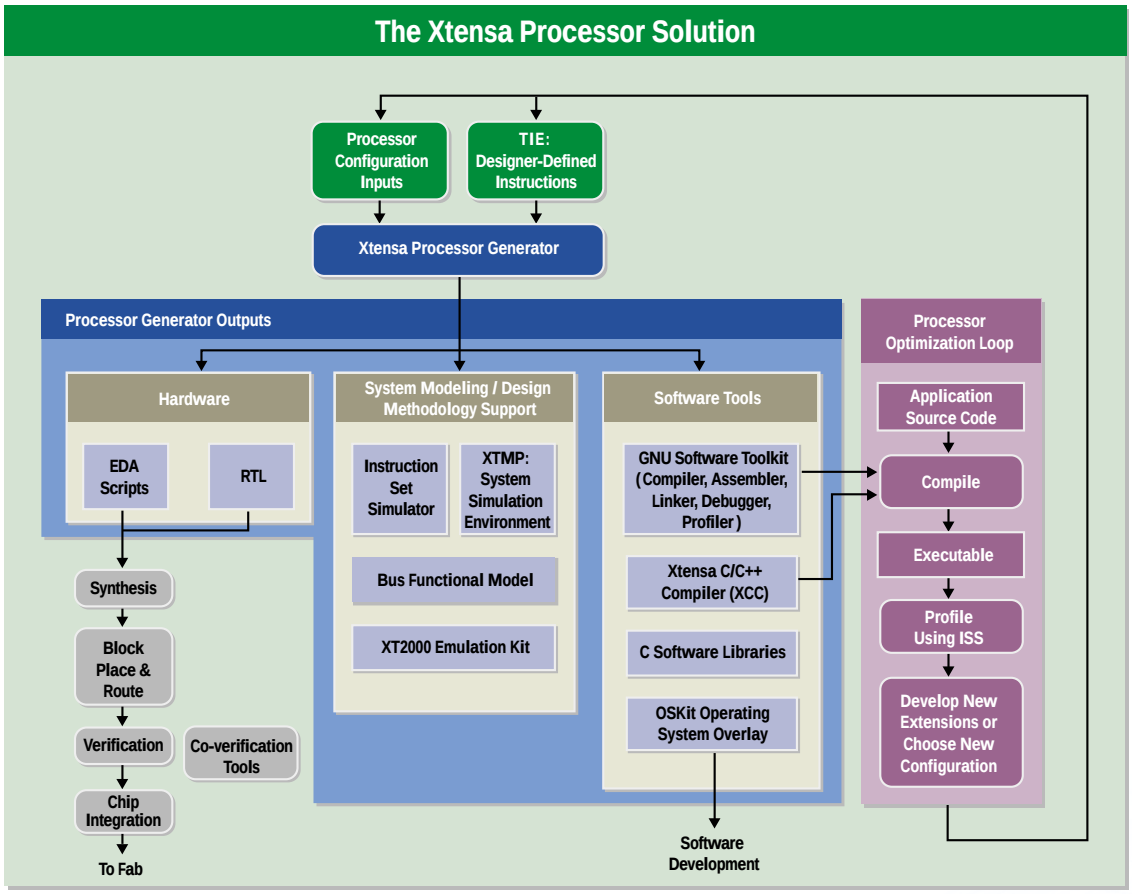



***Figure 2.*** *The Xtensa configurable processor core provides comprehensive hardware and software deliverables and support tools to tailor the embedded SOC to the exact needs of the application. The Xtensa solution includes: synthesizable RTL, full software development tool suite, synthesis/scan/P&R scripts, support for today's popular standard cell libraries and memory blocks, gate-level and RTL-level simulation/verification, bus functional model, and instruction-cycle accurate Instruction Set Simulation models, test bench for the core in RTL and extensive training courses.*

## The Xtensa Processor Generator Makes It Possible

The system designer, hardware or software developer uses the Web-based Xtensa Processor Generator interface to select the instruction set options, memory hierarchy, closely-coupled building blocks and external interfaces required by the application. The designer can also describe additional data-types, instructions and execution units using the Tensilica Instruction Extension (TIE) language. The Xtensa Processor Generator then produces both the complete synthesizable hardware design and the tailored software environment in a matter of hours (see Figure 2.) The synthesizable hardware can be immediately integrated into the remaining SOC design. It is easily ported to any fabrication process for the ultimate cost leverage. Software development, system-level simulation and tuning can also start immediately by using the profiler, various simulation models and overlays for supported RTOSes. The Xtensa Processor Generator allows quick processor configuration of many Xtensa core options, including:

- Execution Unit and ISA Options
  - Multipliers, 32 or 16-bit
  - DSP Engines
    - Single 16-bit MAC
    - SIMD Vectra™ DSP Engine
  - Floating Point Unit
- Interface Options
  - Processor Interface Width: 32/64/128-bit
  - Big-Endian/Little-Endian byte ordering
  - On-Chip Debug
  - Trace Port
  - XLMI high-speed local interface
  - Up to 32 interrupts
- Memory Subsystem Options
  - Memory Management Options
  - Local Data and Instruction Caches
    - Up to 4-way set associative
    - Up to 32 KB
    - Write-back and Write-through cache write policy
  - Separate RAM, ROM areas for Data, Instruction
- Design Support
  - Instruction Set Simulator and Xtensa Modeling Protocol
  - Bus Functional Model
  - RTOS Support packages for ATI Nucleus and WindRiver® Systems' VxWorks®

## Tensilica Instruction Extension (TIE) Language

The Tensilica Instruction Extension (TIE) language is used to describe new instructions and new registers and execution units that are then automatically added to the Xtensa processor. TIE is a Verilog-like language used to describe desired instruction mnemonics, operands, encoding and semantics. TIE files are inputs to the Xtensa Processor Generator. The Generator automatically builds a version of the Xtensa processor and the complete tool chain that incorporates the new TIE instructions.

Program optimization using TIE is first accomplished by identifying "hot spots"–performance-sensitive regions of application software. By utilizing the execution profiler, the designer is able to analyze the efficiency of an application program and evaluate where TIE can be used to accelerate the performance of the software. The designer can iterate a sequence of processor builds while profiling the new processors to weigh the benefit of adding instructions and TIE hardware. Aggressive use of parallelism and other techniques can often deliver 10X, 100X or even greater performance increases using the new TIE instructions (see Figure 3).

The programmer uses the added instructions in C-code as intrinsic function calls, and the Xtensa compilers automatically do all optimizations and scheduling of the assembly code.

## Software Development Tools Based on Industry Standard

The Xtensa software development environment consists of industry standard GNU tools. These include a C/C++ compiler (gcc), assembler, linker, and a debugger (gdb). Tensilica has ported and optimized these tools for the Xtensa instruction set architecture and used proprietary techniques to enable various instruction packages based on the particular hardware configuration generated. A graphical user interface to the debug environment is provided via the Data Display Debugger (DDD) utility.

The Xtensa software development environment is generated from the same database as the processor hardware description. This assures correctness and consistency by construction. Designers get a compiler, linker, assembler, and debugger tuned exactly for their hardware. The software tool chain is automatically updated and optimized to make use of the designer-defined instructions added during the hardware-generation process.

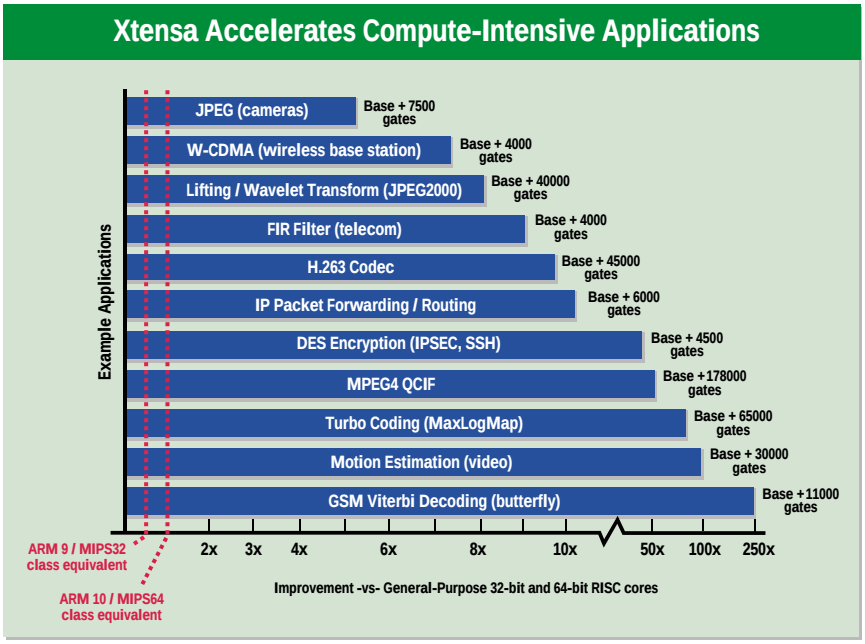



***Figure 3.*** *Designer-defined extensions can have a huge impact on system performance*

## Xtensa C/C++ Compiler

In addition to the standard GNU tool-chain, Tensilica offers the optional Xtensa C/C++ Compiler (XCC). XCC delivers a 20% to 40% average improvement in performance compared to gcc. XCC offers automatic vectorization of C-code for the Vectra DSP Engine. As in the standard Xtensa GNU-based software development tool suite, XCC is generated from the same database as the processor hardware description to assure correctness by construction.

## Xtensa IDE & RTOS Support and the OSKit

The Xtensa OSKit™ consists of configured overlays for leading commercially available real-time operating systems such as Accelerated Technology, Inc.'s Nucleus PLUS and Wind River® System's Tornado® for VxWorks®. The OSKit overlays ensure that all configured features of the architecture, including designer-defined instructions are fully supported by the standard RTOS runtime environment. In addition, leading 3rd party software vendors offer a variety of development environments and debuggers for the Xtensa core including Mentor Graphics® for Seamless™ Co-verification Environment and for XRAY® Debugger support.

## Instruction Set Simulator and Xtensa Modeling Protocol

The Xtensa Instruction Set Simulator (ISS) is a multiple-processor-capable instruction set simulator that is callable in the designer's system API. This means multiple ISSs can be instantiated in a C/C++ environment along with other system components for accurate subsystem-level simulation and analysis.

Xtensa Modeling Protocol (XTMP) is an environment for Xtensa processor subsystem modeling and simulation. It allows rapid assembly of system-level simulations of one or more Xtensa processors and various memories and building blocks.

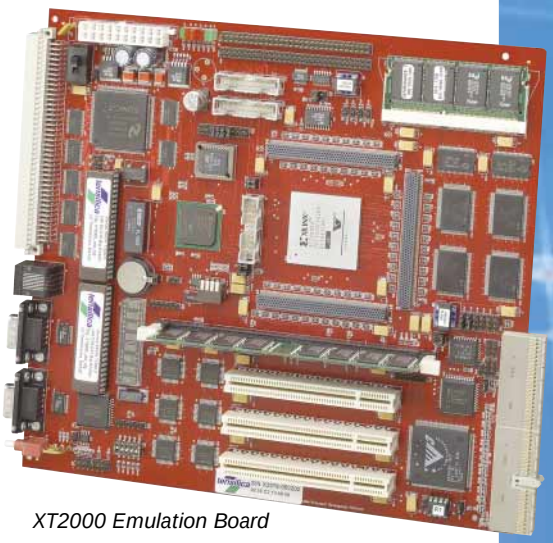
With the Xtensa ISS and XTMP, designers can rapidly build and simulate complete SOC subsystems comprised of multiple, heterogeneous Xtensa processors (see Figure 4).

## System Verification

Tensilica provides a robust suite of verification support tools to assist the designer's SOC verification methodology. From embedded RTL checkers/monitors in the Xtensa interfaces, to automated bus functional model generation which supports HW/SW co-verification, to scripting support for formal verification of TIE extensions, Tensilica provides the ingredients to fit your verification style.

## XT2000 Processor Emulation Kit

The XT2000 Emulation Kit (see photo) is a comprehensive, low cost development tool that uses a programmable logic device to emulate a specific Xtensa processor configuration. The emulation kit enables the developer to evaluate various processor configuration options and to initiate software development and debug early in the design cycle. System designers can specify, implement and debug a new processor configuration, including designer-defined instructions, in just hours.



*XT2000 Emulation Board*

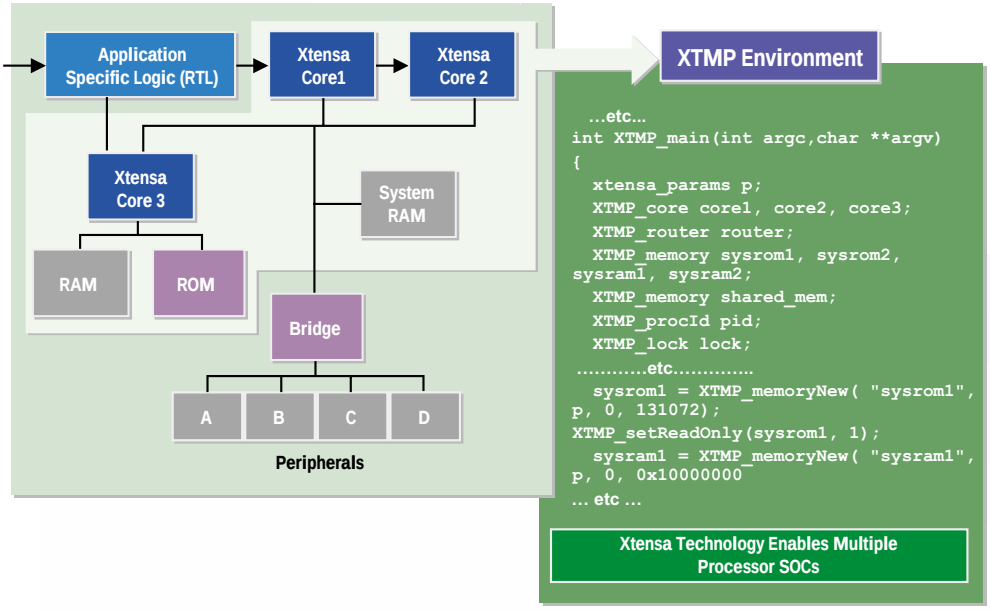



***Figure 4.*** *Xtensa Modeling Protocol (XTMP) environment for multiple-processor system simulation*

## FEATURE SUMMARY

The Xtensa processor offers complete and robust tools, and building blocks to develop the target embedded system-on-chip solution:

- Xtensa Processor Generator
  - Automatic and rapid generation of RTL and companion software development tools
- Tensilica Instruction Extension (TIE) language
  - Designer-defined instructions are easy to construct and are automatically integrated with the base processor.
- Optional units to fit the application
  - Multipliers, 16-bit MAC, FPU and SIMD Vectra DSP Engine blocks
- Memory Management Options
  - Region Protection
  - Region Protection with Translation
  - Memory Management Unit (MMU) with Translation Look Aside Buffers (TLBs)
- Configurable Interfaces
  - Processor Interface (PIF) width: 32 / 64 / 128-bit
  - Optional High Speed Xtensa Local Memory Interface (XLMI)
  - Inbound PIF request to XLMI and Data Ram
- Multiple-Processor Development and Debug Capabilities
  - MP capable, C / C++ callable Instruction Set Simulator (ISS)
  - System Modeling Capability: XTMP Simulation Software Environment for the ISS
  - Multi-Processor On-Chip Debug capability
- Software Development Tools
  - GNU C / C++ software development tool suite
  - High Performance Xtensa C Compiler (XCC)
  - Tailor the Xtensa core and get your software development tools in a matter of hours
- Robust EDA environment support
  - Standard or physical synthesis design flow
- Verification support
  - Comprehensive diagnostics for the Xtensa core and designer-defined TIE verification
- Simulation models
  - Accurate and robust set of models that track the configured processor core
- OSKit™ overlays for supported Real-time Operating Systems
  - Accelerated Technology, Inc.'s Nucleus
  - WindRiver Systems' VxWorks® for Tornado®

## PERFORMANCE SUMMARY

| | |
|---|---|
| Processor Architecture: | 5-stage pipeline, high-performance, 32-bit RISC |
| Instruction Set: | Xtensa ISA with compact 16-bit and 24-bit encoding (no mode switching) |
| Clock Speed: | 350MHz in 0.13µ process<br>200MHz in 0.18µ process |
| Performance: | Tensilica customers have achieved 5X, 10X, and even 100X+ increases in performance in selected algorithms by extending the Xtensa processor with TIE |
| Size: | Approximately 25,000 gates – base processor<br>As low as 18,000 gates – task engine configuration |
| Power: | 0.1mW / MHz in 0.13µ process @ 1.0V<br>0.4mW / MHz in 0.18µ process @ 1.8V |

Clock speed, power, size and performance vary with configuration and choice of implementation technology

## Vectra™ DSP Engine

The Vectra Engine is a key Xtensa building block in a unified solution for System-On-Chip (SOC) designs. For the first time, a single processor architecture can be rapidly configured to satisfy each of the major requirements in embedded control and protocol, signal and image processing. The unified architecture provided by the Xtensa solution ensures a common foundation for software development, simulation and RTOS environments. The Vectra Engine provides world-leading data throughput through various 8, 16, and 24-bit fixed point SIMD configurations (see Table 1 & Figure 5).

**Vectra DSP Engine Configurations**

| | Vectra V1620-8 | Vectra V1620-4 | Vectra V1616-8 | Vectra V0810-8 | Vectra V3224-4 |
|---|---|---|---|---|---|
| Elements per vector A | 8 | 4 | 8 | 8 | 4 |
| Memory width of each element B | 16 | 16 | 16 | 8 | 32 |
| Register width of each element C | 20 | 20 | 16 | 10 | 24 |
| Number of MAC units D | 4 | 2 | 4 | 4 | 2 |
| Multiplier and Multiplicand width E | 16x16 | 16x16 | 16x16 | 8x8 | 24x24 |

***Table 1.*** *A Family of Vectra DSP Engines for the Xtensa Processor*

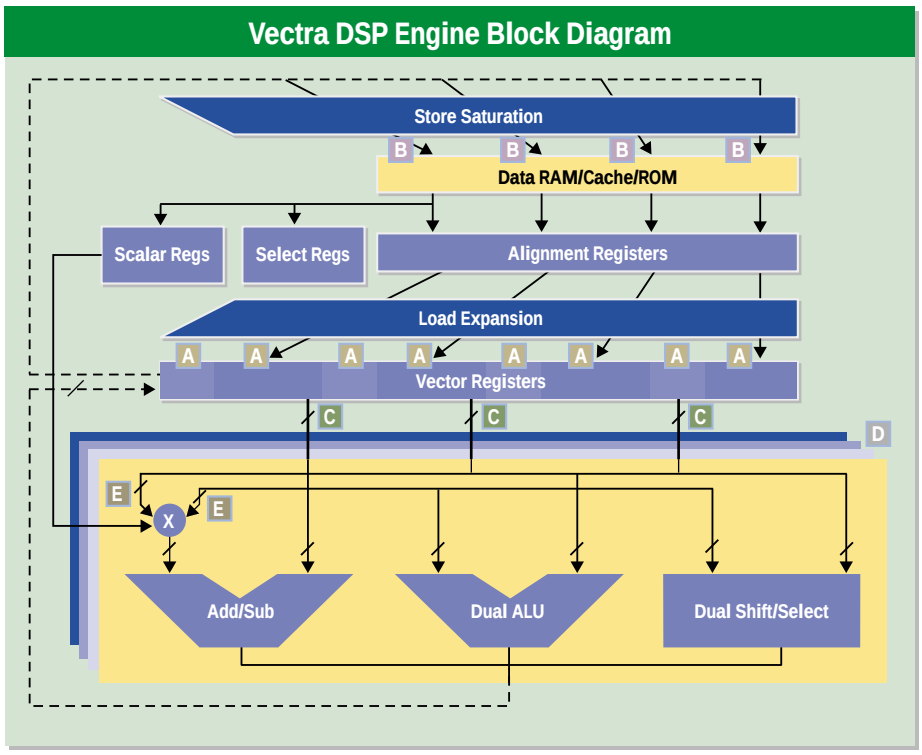



***Figure 5.*** *The Vectra engine's Vector/SIMD architecture is optimized for efficiency and performance*


**Tensilica, Inc.**
Corporate Headquarters
3255-6 Scott Boulevard
Santa Clara, CA 95054-3013
USA
Tel: 408.986.8000
Fax: 408.986.8919

GENERAL INFORMATION
info@tensilica.com

SALES INFORMATION
sales@tensilica.com


For additional information, please see our web site at:
www.tensilica.com