

# Xtensa 9 Customizable Processor

## Small, Ultra-Low-Power 32-bit Embedded Controller

Product Brief

### FEATURES

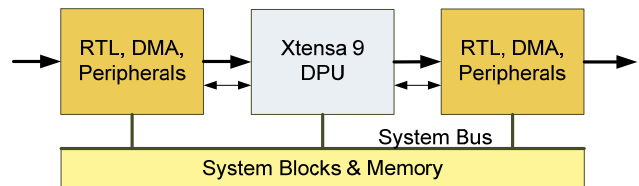
- Highly efficient, small, ultra- low-power core with a 32-bit modern architecture and 5-stage pipeline
- Wide range of configurable options
- Local memories configurable up to 8MB with option for parity or ECC
- Hardware Prefetch Unit reduces long memory latencies
- Extend with application-specific instructions, execution units and register files
- 32-bit wire input and 32-bit wire output GPIO Port option for peripheral control and monitoring
- 2x32-bit Queue FIFO interface option for data streaming, bypassing the system bus
- Optional floating point unit plus double-precision floating point acceleration
- Wide range of operating system support including Linux
- A complete software tool chain is automatically generated for each core

### BENEFITS

- Extremely efficient base architecture that is smaller, lower power, and has better code density than most other 32-bit embedded controllers
- Application-specific instruction extensions provide orders of magnitude performance improvements over traditional CPUs, in many cases eliminating the need to develop RTL blocks
- Lower verification effort with pre-verified, correct-by-construction, RTL generation
- Post-silicon programmability to extend the life of any design
- Highly accurate, high-speed system simulation models automatically created for software development.
- Develop, simulate, debug and profile in one IDE

### Optimized for Low Power

The Xtensa 9 processor, in its smallest configuration, is just 0.024 mm<sup>2</sup> with 12 uW/MHz average dynamic power post place & route in 40 LP process technology at 60 MHz.



**Xtensa 9: an embedded controller with flexible, extremely fast I/O that bypasses the system bus**

### Ideal for Control in the SOC Dataplane

Tensilica's Xtensa® 9 DPU (dataplane processing unit) is an exceptional controller built on Tensilica's unique Xtensa technology. It can easily be extended to out-perform any other embedded processor for a specific application using a combination of configurable options and custom instructions.

By embedding functionality right into the processor datapath itself, designers can use the Xtensa 9 DPU to not only perform control functions, but also some of the finite state machine tasks that manage RTL blocks and some of the RTL logic as well. This makes for a smaller, more efficient chip design, and it significantly reduces the verification challenges associated with new RTL designs.

### Create an Optimized DPU in Minutes

The Xtensa 9 processor is unlike other conventional embedded processors cores—the system designer can mold the processor to fit the target application. By selecting and configuring pre-defined elements of the architecture and by inventing completely new instructions and hardware execution units, the Xtensa 9 processor can deliver performance levels that are orders of magnitude faster than other 32-bit cores. And you can do this in a fraction of the time it takes to develop and verify an RTL-based solution.

Designers can define new instructions utilizing the Tensilica Instruction Extension (TIE) methodology, adding Verilog-like descriptions of datapaths, execution units, and register files that can deliver performance, area, and power characteristics approaching that of custom logic design.

Tensilica's Xtensa 9 DPU was designed from the start to be a basic building block in SOC designs.



Dataplane. DPU. Differentiate.



---

## Feature Overview

### Backwards-compatible ISA since 1999

- Fundamentally architected for extensibility
- Base instruction set of 80 instructions
- Run application code written back in 1999
- All optional blocks are still available
- Any differentiating designer-defined logic can be re-used today

### Optional pre-defined execution units

- 32-bit multiplier and/or 16-bit multiplier and MAC
- Integer divide
- Single-precision floating point unit
- Double-precision floating point acceleration

### Differentiate with designer-defined functions

- Make your specific algorithm run even more efficiently by adding the instructions it needs
- Development tools automatically adapt for full support

### Natural connectivity with RTL blocks

- 2x32-wire GPIO ports for peripheral control and monitoring
- 2x32-bit queue interfaces to FIFOs for data streaming into and out of the processor
- Co-simulation with RTL down to the pin level in SystemC

### Highly configurable interfaces

- Choice of 32-, 64- or 128-bit wide processor interface (PIF) to system bus
- Hardware prefetch unit
- Optional high-speed Xtensa Local Memory Interface (XLMI)
- Write buffer: selectable from 1-32 entries
- Optional AMBA AXI and AHB-Lite bridges with synchronous or asynchronous clocking
- Choice of 1-, 2- or 4-way cache and/or local memories
- Up to 32 interrupts

### Multi-core design style support

- Multi-core system modeling and SystemC co-simulation out-of-the-box, fully supported within the Xtensa Xplorer IDE
- Homogenous and heterogeneous subsystems supported
- Inter-core on-chip debug with break-in/out control
- Optional 16-bit processor ID
- Conditional store option and synchronization library provide shared memory semaphore operations and the “release consistency model” of memory access ordering

### Complete hardware implementation and verification flow support

- Automatic generation of RTL and tailored EDA scripts for leading-edge process technologies, including physical synthesis and 3D extraction tools
- Auto-insertion of fine-grained clock gating for ultra-low power
- Hardware emulation support including automated FPGA netlist generation
- Comprehensive diagnostic test bench
- Formal verification support for designer-defined functions

### High-speed, high-accuracy system simulation models automatically created

- High-speed instruction-accurate simulator for software development
- Pipeline-modeling, cycle-accurate Xtensa instruction set simulator
- Xtensa SystemC (XTSC) transaction-level modeling (TLM) support, including out-of-the-box multi-core simulation
- Hardware co-simulation with RTL in SystemC with Tensilica's pin-level XTSC

### Integrated design environment

- Create, simulate, debug and profile whole designs in one tool—Xtensa Xplorer is a high productivity IDE
- Ninth generation software development tools target each processor. The advanced Xtensa C/C++ compiler (XCC) includes optimizations for base, optional and designer-defined instructions
- Increase productivity with multi-core subsystem design and simulation support
- Custom data display formatting for easy debug of vector and fixed-point data types as well as bit-mapped status and control
- Use Mentor Graphics Nucleus+, Express Logic's ThreadX, Micrium's uC/OS-II, Tata Elxsi's Ro-SES or the Linux operating systems



## Efficient Base Architecture

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, six special purpose registers, and 80 base instructions, including improved 16- and 24-bit (rather than 32-bit) RISC instruction encoding. Key features include:

- A wide range of configurable options to ensure you get just the logic you need to meet your functional and performance requirements
- Modelessly intermix 16- or 24-bit instructions for lowest code and performance overhead.
- Efficient 5-stage pipeline
- Local memories configurable up to 8MB with optional parity or ECC
- Caches up to 32KB and up to 4-way set associativity on both Instruction and Data sides
- Automated fine-grained clock gating throughout processor for ultra-low power solutions

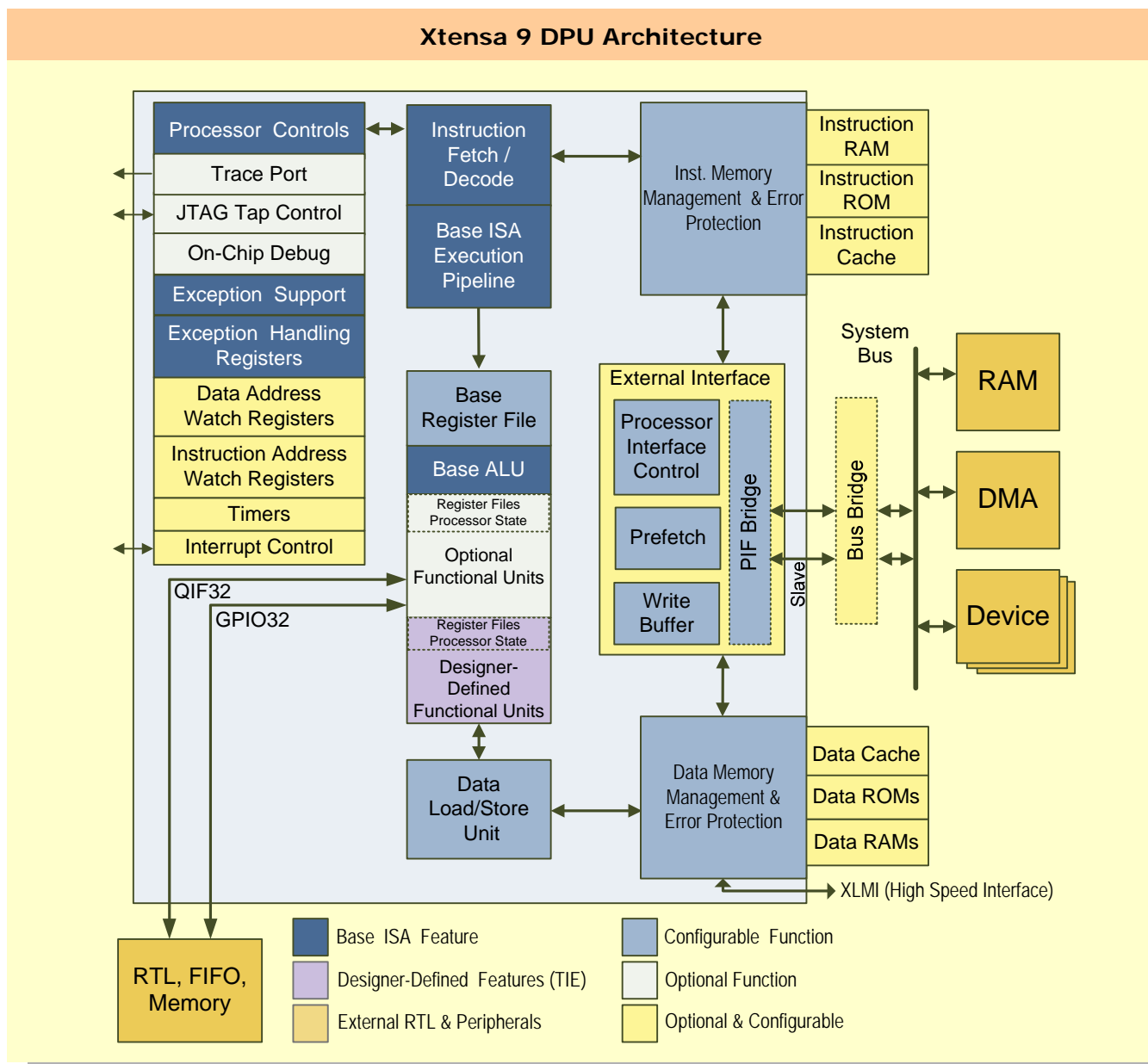


Figure 1. Xtensa 9 DPU showing standard, optional, and designer-defined blocks



---

## Base Instruction Set Compatibility

Configurability of a Tensilica processor core never compromises the underlying base Xtensa instruction set architecture (ISA), thereby ensuring availability of a robust ecosystem of third party application software and development tools. All configurable, extensible Xtensa processors are always compatible with major operating systems, debug probes, and ICE solutions. For each processor, the automatically generated complete software development toolchain includes an advanced Integrated Development Environment (IDE) based on the ECLIPSE framework, a world-class compiler, a cycle-accurate SystemC-compatible instruction set simulator, and the full industry standard GNU Toolchain.

Tensilica uses an ISA that has been backwards compatible since its introduction in 1999. It uses a base instruction set of 80 instructions and was fundamentally architected for extensibility. Designers can run application code written back in 1999 and it will run on the Xtensa 9 processor today. Any differentiating designer-defined instructions from earlier designs can be re-used today.

## Smaller Code Size

The Xtensa 9 DPU can modelessly issue 24-bit and 16-bit instructions, leading to 25-50% better code density and, therefore, smaller memories than mixed 32- and 16-bit architectures. Since memories typically dominate SOC area, this code density advantage translates into significant SOC area savings.

## Powerful Base ISA

The Xtensa ISA includes powerful compare-and-branch instructions and zero-overhead loops, which allow the compiler to generate tight, optimized loops. It also provides bit manipulations including funnel shifts and field-extract operations that are critical for applications such as networking that process the fields in packet headers and perform rule-based checks.

## Extensible ISA

One of the fundamental technology innovations in the Xtensa processor is the ability to easily and seamlessly add new instructions into the processor's datapath. The associated C data types, software tool chain support and the EDA scripts required to synthesize the processor are all generated automatically, just as if they had been there from the start. The specification of this new datapath and associated instructions and C data types is written in the Tensilica Instruction Extension (TIE) language, which is explained in more detail in a later section.

## Highly Configurable Functionality

Select from click-box options to add functionality to your processor and evaluate performance improvements in a matter of minutes. Basic interface options include:

- Processor interface (PIF)
  - Width: 32/64/128-bit
  - Optional "no PIF" configuration
- Optional AMBA AXI and AHB-Lite bridges with synchronous or asynchronous clocking
- Optional 16-bit processor ID
- Inbound DMA option
- XLMI high-speed local interface
- Big-Endian/Little-Endian byte ordering
- On-chip debug port (IEEE 1149.1 compliant)
- Trace port signals
- Up to 32 interrupts with up to 7 levels of priority plus a separate Non-Maskable Interrupt level
- Write buffer: selectable from 1 to 32 entries
- 2x32-wire GPIO ports for direct control and monitoring of peripherals
- 2x32-bit queue interfaces for streaming data into and out of the processor via FIFOs
- Single 16-bit MAC (multiply accumulator)
- 16- or 32-bit multipliers
- Low-area integer divider
- Single precision floating point unit
- Double precision floating point accelerator

Memory subsystem options include:

- Up to 2 local instruction and data RAMs and ROMs up to 8 Mbytes each
- Local data and instruction caches
  - Up to 4-way set associative
  - Up to 32 KB
  - Write-back and write-through cache write policy
- 4-way cache plus local memories
- Memory management options
- Region protection
- Region protection with translation
- Memory Management Unit (MMU) with Translation Look Aside Buffers (TLBs), includes no-execute bit security support
- MMU for the Linux operating system



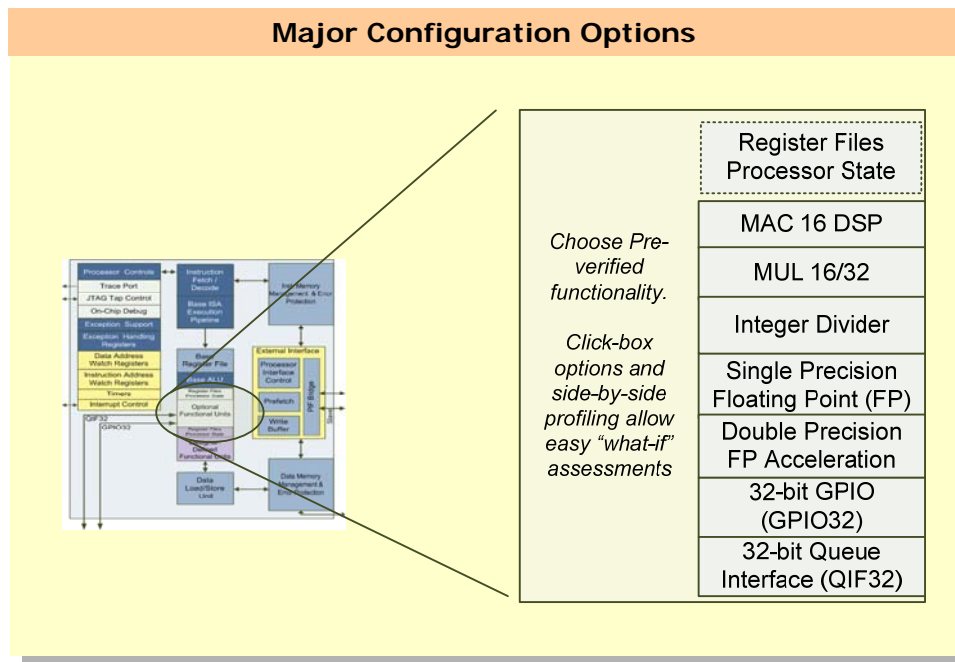
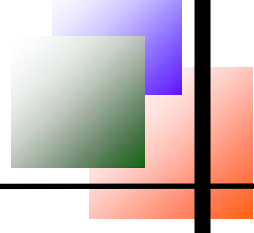


Figure 2. Tensilica offers pre-verified major configuration options for the Xtensa 9 DPU

### Configuration Options Bypass the Bus for Fast, Efficient Data I/O

Two configuration click-box options allow Xtensa 9 processors to very quickly communicate data, control or status information with RTL blocks or other Xtensa processors.

The GPIO32 configuration option adds two 32-wire ports to the Xtensa 9 processor (one input, one output) to quickly control and monitor peripherals or other logic in the system.

The QIF32 configuration option adds two 32-bit queue interfaces for FIFO-like data streaming into and out of the processor. The input queue functions with a familiar pop/empty/data interface to external logic while the output queue presents a similar push/full/data interface. All interactions with the Xtensa 9 processor pipeline are automatically implemented when the option is selected.

These options are accessed as registers in the processor, so no separate load/store is required to operating on the data.

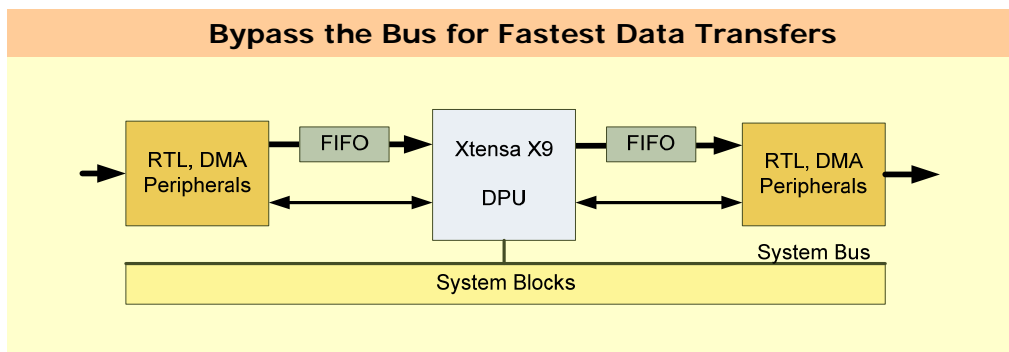


Figure 3. GPIO ports and queue interfaces allow direct data transfers for the fastest connections to RTL blocks or other Xtensa processors



## Extensibility Unlocks the True Power of Xtensa 9

Most embedded processors offer fixed hardware functionality with options for memory size, cache size, and bus interface. Performance is proportional to the clock speed. Beyond that, application code optimization effort or a move to the next processor in the roadmap is required. Tensilica offers something different—the opportunity to optimize the processor itself using Tensilica’s TIE language.

Tensilica’s TIE language is used to describe new instructions, registers and execution units that are then automatically added to the Xtensa 9 processor. TIE is a Verilog-like language used to describe desired instruction mnemonics, operands, encoding, and execution semantics. The TIE files are inputs to the Xtensa Processor Generator. The Generator automatically builds the processor and the complete software tool chain that incorporates all configuration options and new TIE instructions. The base instruction set remains for maximum compatibility with third party development tools and operating systems.

The TIE language unlocks the true power of Xtensa 9. It allows designers to get orders of magnitude performance increases in their processors and create differentiated processors.

## Flexibility to Add Just What You Need

Just as the designer can choose from a set of predefined functional options to improve processor performance, the designer can now create instructions that can speed up standard or proprietary algorithms. Using the tools provided, application hot spots can be identified and additional logic created to process these hot spots more efficiently, without the need to increase the clock frequency or re-write the software.

## Differentiate—Make a Processor That’s Uniquely Your Own

When processors have fixed hardware functionality and your competitors are using the same or similar processors, then differentiation is often limited to the algorithm implemented. Fixed processors are good at general-purpose computing, but not so good at any specific algorithm. Tensilica gives you the opportunity to differentiate at the hardware level and implement algorithms more efficiently by designing hardware that will accelerate your specific algorithm. This means that your design will be almost impossible to copy, as only your hardware will reach the performance required on the same software implementation.

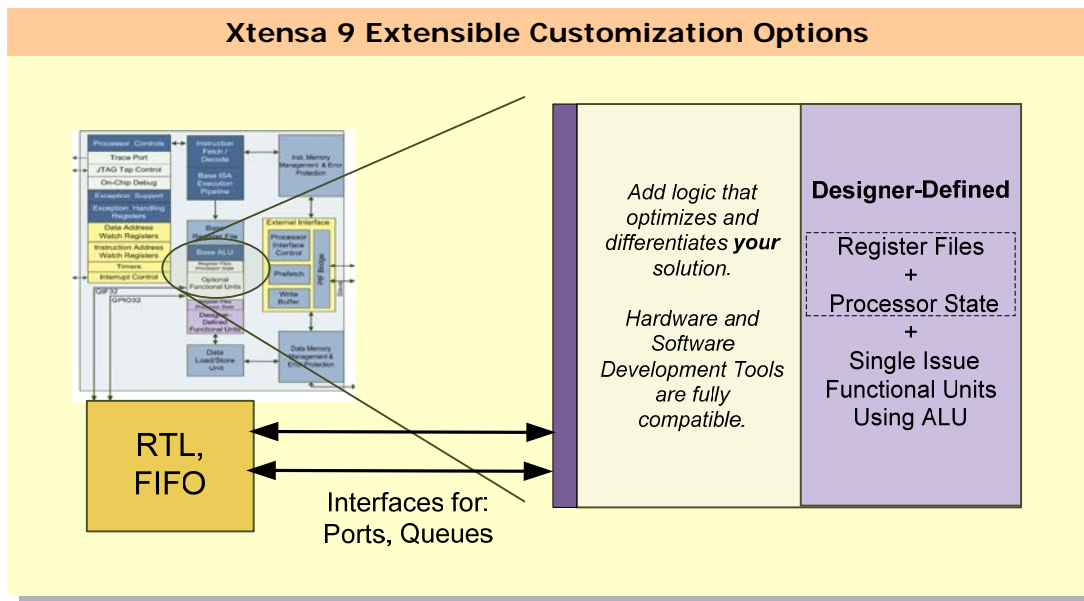


Figure 4. Xtensa 9 can be easily extended with custom instructions and direct interfaces to RTL blocks and FIFOs

## Rapid Design Development, Simulation, Debug and Profiling

The Xtensa Xplorer™ integrated design environment (IDE) serves as the graphical user interface (GUI) for the entire design experience. From Xtensa Xplorer, designers with existing application software code can profile their application, identify hot spots, decide on configuration options, and add new instructions and execution units to optimize performance and generate a new processor—all within a matter of hours. No other IP provider puts such flexibility directly into the hands of the designer with a tool that integrates software development, processor optimization, and multiple processor SOC architecture in one IDE.

Hardware designers now have new options for implementing algorithms. Interfaces can be added to the processor to offer direct, deterministic connectivity to SOC logic. With the GPIO port and queue interface options, designers can stream data into or out of the processor. This direct connectivity with the rest of the SOC offers great control and predictable bandwidth. The simple C programs needed to control the Xtensa processor can be written and debugged within the Xtensa Xplorer IDE.

The Xtensa Processor Generator creates a complete hardware design with matching software tools, including a mature, world-class compiler, a cycle-accurate SystemC-compatible instruction set simulator (ISS) and the full industry standard GNU Tool-chain.

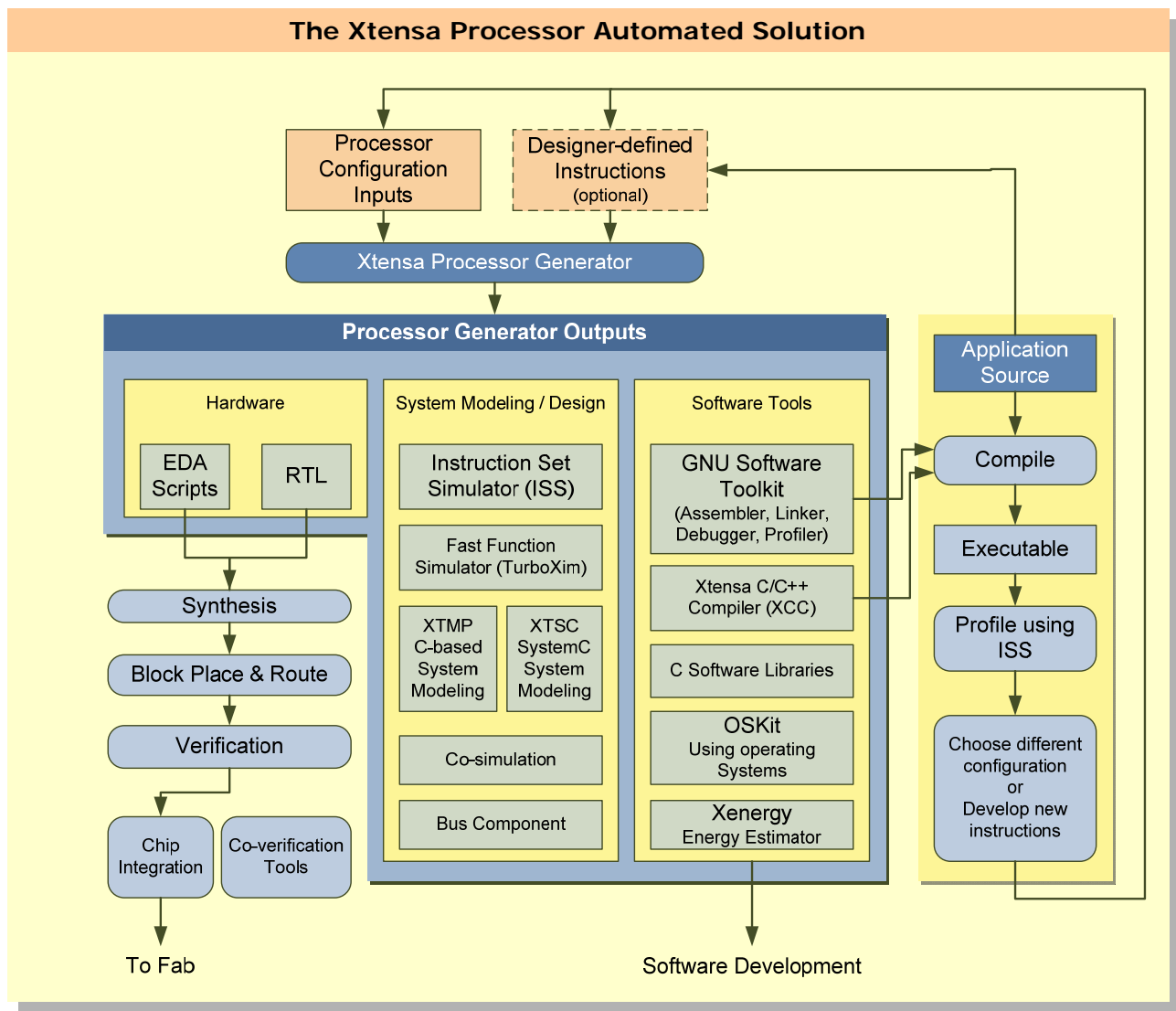
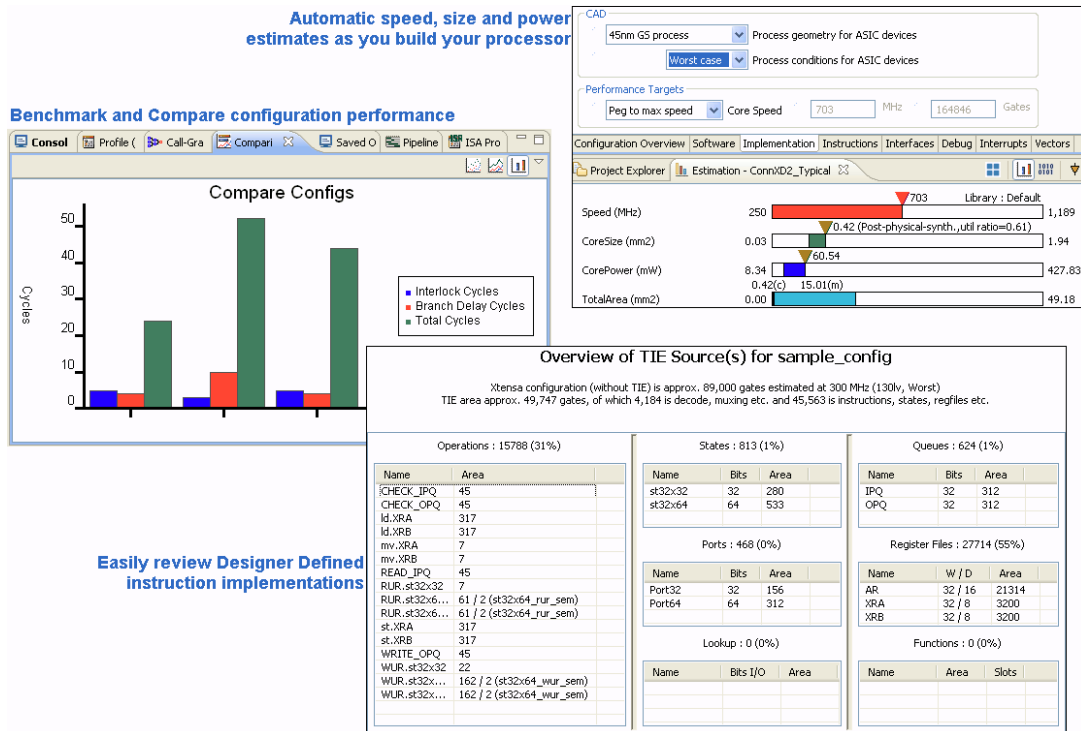


Figure 5. Tensilica's proven methodology automates the creation of customized processors and matching software tools

## Hardware Development



**Figure 6. Xtensa Xplorer can display valuable information including performance comparisons, instruction sizes, and processor size, area and power**

### For Hardware Development

Hardware designers can profile, compare and save many different processor configurations. Designers can use the ISS to simulate a single processor or, for multiple processor subsystems, designers can choose Tensilica's XTensa Modeling Protocol (XTMP) or Tensilica's XTensa SystemC (XTSC) modeling tools.

Xtensa Xplorer serves as the gateway to the Xtensa Processor Generator. Once a processor configuration is finalized, the Xtensa Processor Generator creates the automatically verified Xtensa processor to match all of the configuration options and extensions you have defined, in about an hour. The full software tool chain is also created that matches all processor modifications made. See the Processor Developer's Toolkit product brief for more information.

### Complete Hardware Implementation and Verification Flow Support

- Automatic generation of RTL and tailored EDA scripts for leading edge process technologies, including physical synthesis and 3D extraction tools
- Auto-insertion of fine-grained clock gating delivers ultra-low power

- Hardware emulation support including automated FPGA netlist generation
- Comprehensive diagnostic test bench
- Format verification support for designer-defined functions
- Pipeline-modeling, cycle-by-cycle accurate Xtensa instruction set simulator (ISS)
- System modeling capabilities with optional XTMP and XTSC simulation environments
- Multiple-processor on-chip debug capable with break-in/out control
- Hardware co-simulation in System C with Tensilica's pin-level XTSC connectivity to RTL
- XTSC transaction-level modeling support, including out-of-the-box multi-core co-simulation
- Xenergy™ energy estimation tool to optimize hardware and software for power





## Software Development

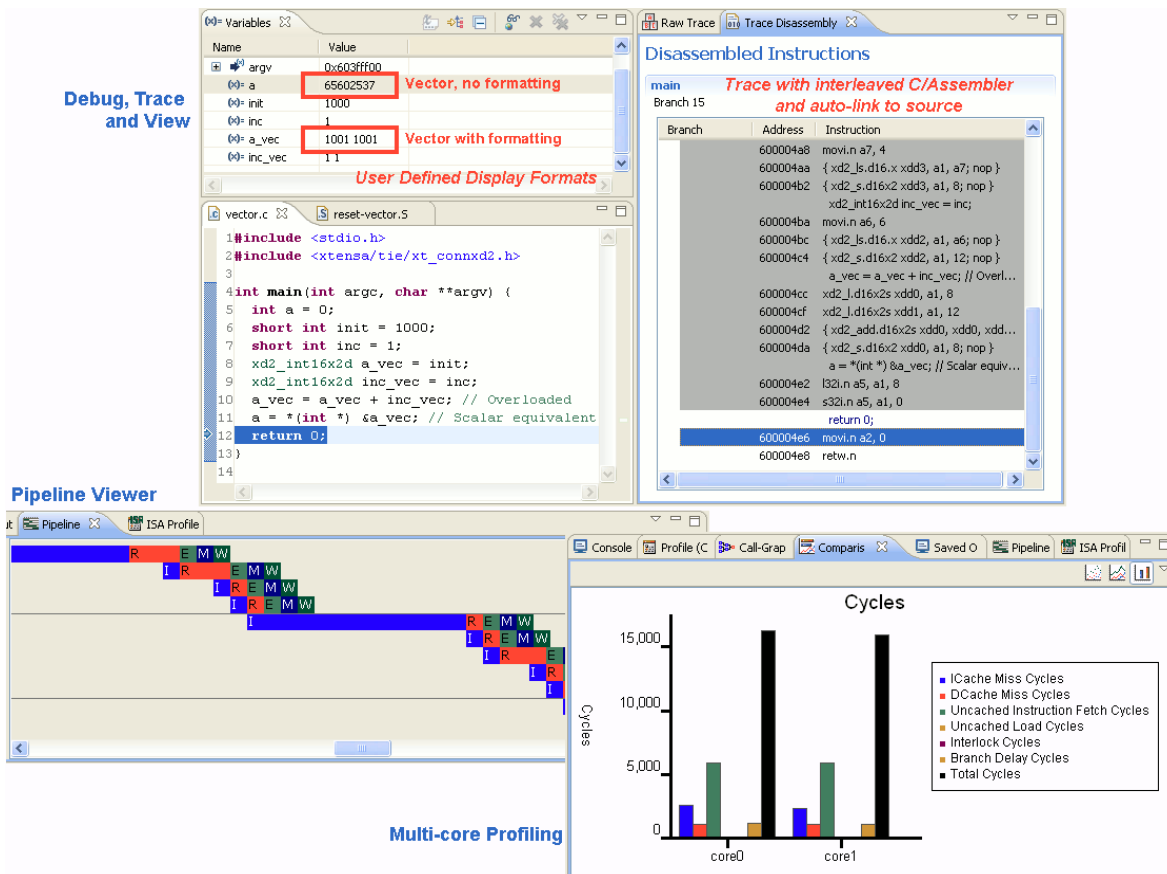


Figure 7. Xtensa Xplorer shows debug/trace, profiling of pipeline utilization, and a cycle comparison for a multiple core simulation

## For Software Development

The Xtensa Software Developer's Toolkit (SDK) provides a comprehensive collection of code generation and analysis tools that speed the software application development process. Tensilica's Eclipse-based Xtensa Xplorer GUI serves as the cockpit for the entire development experience and also provides powerful visualization tools to aid application optimization.

The entire Xtensa software development tool chain, along with simulation models, RTOS ports, optimized C-libraries, etc., are automatically generated by the Xtensa Processor Generator. This also ensures that all the software tools – such as the compiler, linker, assembler, debugger, and instruction set simulator – always match and are tuned exactly to any custom processor hardware.

## Complete Software Development Tools

- Mature, highly optimizing C/C++ compiler (XCC) that rivals hand-coded assembly applications on other processors

- GNU-based Assembler and Linker
- Pipeline-modeled, cycle-accurate ISS
- High-speed (40-80x) instruction accurate TurboXim™ simulator speeds software development
- XTMP and XTSC for multiple processor simulation and modeling
- Debug offers full GUI and command line support for single and multiple processor designs
- Profiling views processor pipeline utilization as well as time in functions across multiple processors. Allows "what if" comparisons
- Xenergy energy estimation tool helps the designer tune the software for energy consumption
- Support for major operating systems including Mentor Graphics' Nucleus Plus, Express Logic's ThreadX, Micrium's µC/OS-II, Sophia Systems' µITRON, and open-source Linux.



## Ideal for Applications Where Low Power is Critical

Power often is the key issue in an SOC design. Tensilica employs many techniques to reduce power consumption, both built in to the base hardware and into the configuration options, allowing more control over system and memory resources. Tensilica processors consistently consume less power than other licensable embedded CPUs at equivalent gate counts.

Tensilica automates the insertion of fine-grained clock gating for every functional element, including those defined by the designer. This automation gives the Xtensa 9 DPU a significant advantage over RTL design where manual, error-prone post-layout tuning of clock circuits is often required.

Accessing local memories is one of the highest power consuming activities. Tensilica has designed the Xtensa 9 processor to eliminate any unnecessary local memory interface activation if that memory is not directly addressed by the processor.

Tensilica automates the implementation of these energy saving techniques by the Xtensa Processor Generator.

The designer can configure the external data bus width and internal local memory data widths independently. This allows system-level power optimizations depending on whether the processor is constrained by external or internal instruction and data access.

Designers use Tensilica's Xenergy™ energy estimation tool to evaluate energy-related tradeoffs in the design process. The Xenergy tool can be used to optimize TIE hardware instructions and to fine tune the software application for the lowest energy usage.

## Specifications

Configuration	Post-Route Area ( $\mu\text{M}^2$ )	Clock Rate (MHz)	Power Dissipation (mW/MHz)
<b>Smallest*</b> —Synopsys library, TSMC 40LP, low-power flow	0.024	60	0.012
<b>Smallest*</b> —Synopsys library TSMC 40LP, high-speed flow	0.044	670	0.018
<b>Smallest*</b> —Synopsys library, TSMC 45GS, low-power flow	0.024	62	0.009
<b>Smallest*</b> —Synopsys library, TSMC 45GS, high-speed flow	0.044	1032	0.014
<b>106Micro**</b> —Synopsys library, TSMC 40LP, low-power flow	0.046	57	0.017
<b>106Micro**</b> —Synopsys library TSMC 40LP, high-speed flow	0.074	540	0.026
<b>106Micro**</b> —Synopsys library TSMC 45GS, low-power flow	0.045	57	0.016
<b>106Micro**</b> —Synopsys library TSMC 45GS, high-speed flow	0.074	907	0.019

\*Smallest—smallest configuration used by customers with only local instruction and data RAM interfaces and full clock gating.

\*\*106Micro—similar to Tensilica's Diamond Standard 106Micro with an iterative 32x32 multiplier, separate instruction and data memory interfaces, PIF, an interrupt controller with 15 interrupts at two priority levels, an integrated timer, on-chip debugging hardware, and embedded trace support.

## Tensilica, Inc.

3255-6 Scott Blvd., Santa Clara, CA 95054-3013, USA  
Tel: 1-408-986-8000 Fax: 408-986-8919 Website: [www.tensilica.com](http://www.tensilica.com)

©March 2011 Tensilica and Xtensa are registered trademarks of Tensilica, Inc.  
The Tensilica logo, Xenergy, Xplorer and TurboXim are trademarks of Tensilica, Inc. All other trademarks are the property of their respective owners.

