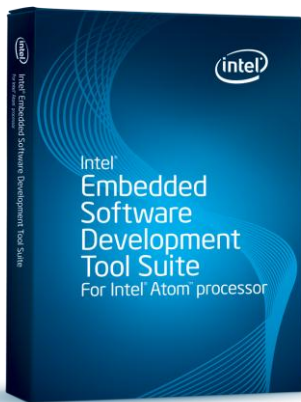




# Intel® Embedded Software Development Tool Suite For Intel® Atom™ processor

## Product Brief

Intel® Embedded Software Development Tool Suite  
for Intel® Atom™ processor



## Turbocharge your code for Intel® Atom™ processor-based embedded systems

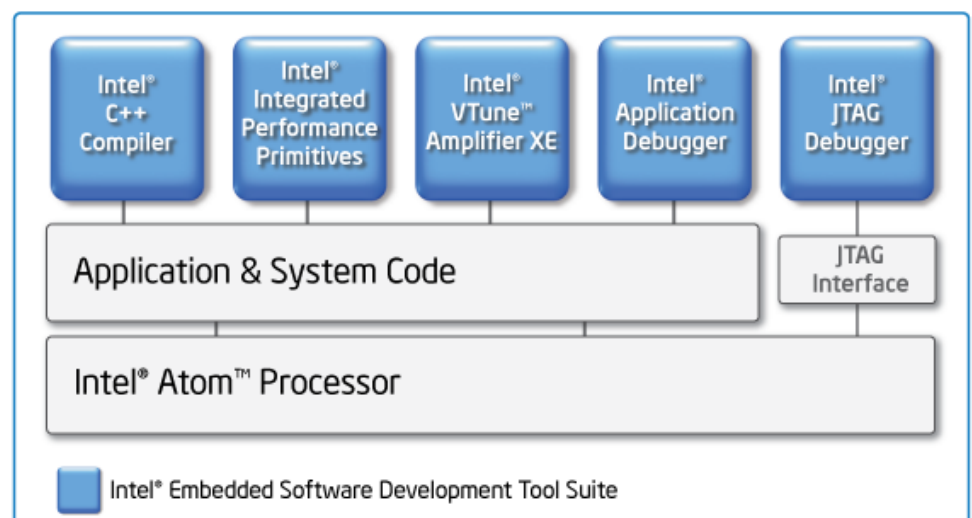
- High Performance Data, Media and Signal Processing Libraries
- High Performance C/C++ Compiler
- Powerful Application and JTAG-based Debug
- Advanced Performance Analysis Tool

Benefit from a comprehensive software development tools solution for your Intel® Atom™ processor-based embedded system designs and application software development. Coding, compiling, debugging and performance tuning made simple.

The Intel® Embedded Software Development Tool Suite for Intel® Atom™ processor is a complete solution that addresses embedded software development requirements for Intel® Atom™ processor-powered platforms such as embedded systems, tablets, netbooks, smartphones, IVI and other CE devices.

The Embedded Software Development Tool Suite covers the entire software development cycle: coding, compiling, debugging, and analyzing performance. All included tools are Linux\* hosted, are compatible with GNU tools, and support multiple Linux\* OS based targets.

- Intel® C++ Compiler for Linux\*
- Intel® Integrated Performance Primitives for Linux\*
- Intel® Application Debugger for Intel® Atom™ processor
- Intel® JTAG Debugger for Intel® Atom™ processor
- Intel® VTune™ Amplifier XE for Linux\*
- Compatibility and support for Linux\* based targets e.g. Yocto Project\*



# Features and Benefits

Feature	Benefits
<b>Intel® C++ Compiler</b>	<ul style="list-style-type: none"><li>• Boost performance with highly optimizing compiler for Intel® architecture</li><li>• Full support for Intel® Atom™ processor</li><li>• Compatible with GCC</li></ul>
<b>Intel® Integrated Performance Primitives Library</b>	<ul style="list-style-type: none"><li>• Highly optimized data, multimedia and signal processing functions</li><li>• Intel Atom processor optimized</li></ul>
<b>Intel® Application and JTAG Debuggers</b>	<ul style="list-style-type: none"><li>• Intel Atom processor and chipset support</li><li>• Kernel and low-level driver debug</li><li>• Application debug</li><li>• Thread grouping and thread specific run control</li><li>• Linux* OS aware debug</li><li>• Built-in flash memory tool</li><li>• Application and system wide execution trace support</li></ul>
<b>Intel® VTune™ Amplifier XE</b>	<ul style="list-style-type: none"><li>• Analyze and tune code running on target device</li><li>• No source code instrumentation required</li><li>• Advanced profiling for scalable performance tuning assistant</li><li>• Event based sampling on embedded target device</li><li>• Access to performance monitoring unit, with system-wide profiling capability</li></ul>
<b>Linux* Compatibility</b>	<ul style="list-style-type: none"><li>• Support of different embedded Linux* distributions utilizing a Linux 2.6.x and 3.0.x kernel</li><li>• Validated for Yocto Project* build Linux*-based target systems</li><li>• Compiler integration with Yocto Project* Application Development Toolkit</li><li>• Remote debug support for a Yocto* image running in virtual machine and on physical target device</li></ul>

## Performance

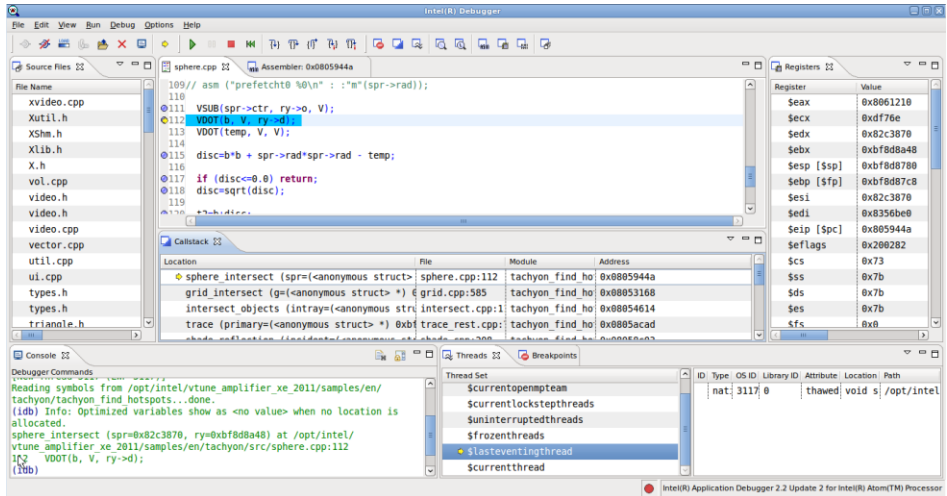
The highly optimized in-order scheduler and other improvements and optimizations targeting the latest Intel® Atom™ Processor generations in the Intel® C++ Compiler provide a significant performance advantage over other solutions in the industry. The highly optimized Intel® Integrated Performance Primitives (Intel® IPP) provide the same easy-to-use API as for other Intel® architectures, while at the same time being highly optimized for Intel Atom processors. Intel® VTune™ Amplifier XE helps identify performance bottlenecks and analyze performance data collected on the Intel Atom processor-based target device.

## Data, Multimedia and Signal processing with performance libraries

With Intel® Integrated Performance Primitives, application developers can focus on feature implementation rather than the optimization of application code for specialized operations. Intel IPP provides performance-optimized building-block functions for key software applications such as: multimedia playback/recording, editing, image processing, audio/speech/signal processing, and network data communications. Free code samples downloadable from Intel's website enhance the value of the Intel IPP functions by illustrating the implementation of multi-threaded application blocks such as video, audio, and speech codecs.

# Application and System Debug

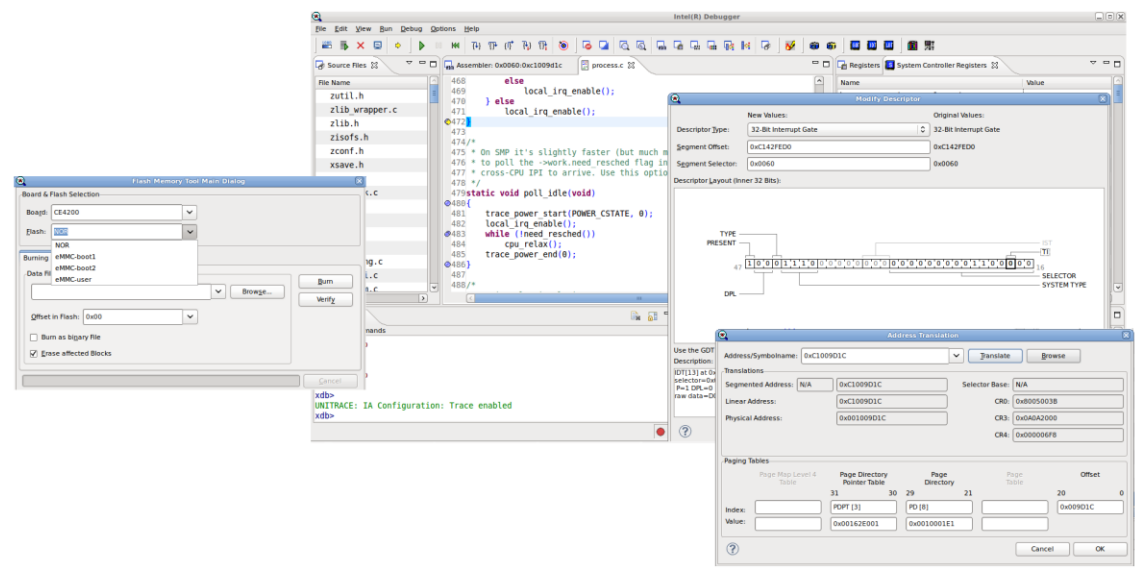
Intel's Debuggers for Intel® Atom™ processors support all aspects of application and system level debugging. Whether the issue to be isolated and fixed is in firmware, bootloader, low-level driver or the operating system kernel, or whether it is in a high-level C/C++ language application, Intel debuggers provide a solution to speed up bug resolution. Features such as full execution trace support, thread specific breakpoints and run-control, device register views, page translation table access and flash memory writer capabilities (requires JTAG hardware interface) provide more comprehensive visibility and control of the environment the application runs in. Software stack components such as shared objects, libraries, and applications can be debugged on the host development environment as well as remotely via TCP/IP.



Remote application cross-debugger with advanced thread-awareness

# JTAG Debug

The Intel® JTAG Debugger for Intel® Atom™ processor is the recommended debug solution for device manufacturers, embedded developers, and operating system vendors who work on e.g. low-level driver development and rely on kernel debugging capabilities. A JTAG debug solution does not require a running operating system on the device. It is therefore ideal for bootcode and firmware as well as kernel debugging while booting the OS. Through a JTAG interface the target hardware is connected to the debugger on the host system. The JTAG debugger allows in-depth access to IA-specific features e.g. execution trace support, as well as access to system-on-chip and chipset peripheral register values. This unique feature makes it especially valuable for driver and firmware development and debugging. The entire processor and peripheral registers are fully documented in the Intel® JTAG debugger solution.



System debugger with bitfield editor, memory configuration capability and integrated flashing tool

## Supported JTAG devices:

Third-party JTAG interface support available from Macraigor\*: Additional product details for the Macraigor\* usb2Demon\* hardware device at [www.macraigor.com/intel](http://www.macraigor.com/intel)

Intel® XDP3 JTAG interface (please contact Intel if you are a hardware manufacturer): [EmbeddedDevTools@intel.com](mailto:EmbeddedDevTools@intel.com)

# Application Debug

The Intel® Application Debugger for Intel® Atom™ processor supports all aspects of debugging, from low-level assembler debugging to high-level language C/C++ application debugging. The debugger provides access to hardware trace buffers which allow for full execution trace support to root cause complex runtime errors in embedded systems. The debugger's sophisticated filters enable task context aware tracing to only capture application specific issues. On the development host, the application debugger supports TCP/IP-based development and testing. The debugger can be used not only to debug applications that are running on actual Intel Atom processor-powered devices, but also for host development system-based debug before deployment to the targeted device inside the host's native runtime environment and inside a QEMU\* based virtual machine. For threaded development it provides the ability to define thread groups and apply breakpoints and run-control to specific application threads. Native testing and remote debugging of processes running on a virtual machine reduce time and simplify the development process. The GUI-driven application debugger supports execution trace unwinding allowing looking back at the history of an executed program, providing OS awareness and thread-aware debugging.

# Yocto Project\* and Intel® Software Tools

The Intel® Embedded Software Development Tool Suite is a set of highly optimizing software development tools, with powerful debuggers for in-depth system analysis. The tools are compatible with the GNU world and complement the standard open source GNU tools offering.

Furthermore, the Intel® C++ Compiler and Intel® Integrated Performance Primitives have been validated to be compatible with the Poky\* Linux based application cross-build environment used by the Yocto Project\* Application Development Toolkit. The tool suite installer permits automated integration of the Intel C++ Compiler into the Application Development Toolkit at install time and allows easy and seamless switching between the Intel Compiler and the GNU Compiler.

The Intel® Application Debugger can remotely connect to any device running Yocto\* build Linux\* with the help of TCP/IP communication and a small user mode debug handler application on the target device. The same approach can also be used to debug an application running on a Yocto\* based virtual machine image.

The Intel® JTAG Debugger can be used to debug device drivers and the OS kernel of Yocto Project\* build Linux images running on a device with an Intel eXtended Debug Port (XDP).

The Sampling Collector for Intel® VTune™ Amplifier XE provides low overhead event based performance sampling, identifying performance bottlenecks and resource constraints as well as other potential issues affecting program efficiency. It achieves this through access to the performance monitoring unit (PMU), with a kernel module that can easily be rebuilt against the Yocto Project\* kernel of your choice.

# Performance Analysis and Tuning

Intel® VTune™ Amplifier XE provides an easy to use user Interface and its rich functionality makes it fast and easy to identify performance bottlenecks by e.g. providing a list of the most active functions. A click on a function name displays the source and shows the most time-consuming code statements. Furthermore, event-based sampling support for low-power Intel Atom processors permits the developer to determine the causes for execution stalls that impact performance.

The sampling collector driver for Intel® VTune™ Amplifier XE for event based sampling can be rebuilt within the target OS build environment on the host system accommodating software stack limitations that a developer can face on some small form factor target devices. A modifiable build configuration script is provided for this purpose.

Line	Source	CPU_CLK_UNHALTED CORE by Package	BR_INST_RETIRED, MISPRED by Package	L1I_MISSES by Package	RS_UOPS_DISPATCHED by Package	UOPS_RETIRED, ANY by Package
112	{					
113	/******					
114	// Try to defeat hardware prefetching by varying the stride					
115	int i(0), iteration_count(0);					
116						
117	do {	1,498,000,000	35,400,000		4,686,000,000	4,484,000,000
118	mem_array[j*mem_array_i_max+1] = *fill_value + 2;	6,442,000,000			18,874,000,000	15,992,000,000
119						
120	// Code to give the array accesses a non-uniform stride to defeat hardware prefetch					
121	if ((iteration_count % 3) == 0) j+=3;	14,098,000,000	46,000,000		44,940,000,000	39,432,000,000
122	else j+=2;	910,000,000	3,800,000		3,148,000,000	2,480,000,000
123	iteration_count++;					
124	} while (j < mem_array_j_max);	2,468,000,000	1,200,000		8,346,000,000	6,460,000,000
125	}					
126	/******					
127	/******					
128						
129	int iteration_count(0);					
130	for (int j = 0; j < mem_array_j_max; j++)					
131	{					

*Intel® VTune™ Amplifier XE identifies optimization opportunities by mapping source code to time spent, and chipset events like cache misses and branch mispredictions.*

Technical Specifications	
Host system OS support	Ubuntu* 10.04, 11.04 Fedora* 11, 13, 14 and Fedora 15 Please consult the release notes for a detailed overview of supported host platforms
Target system & OS support	Broad support of Intel® Atom™ processor variants (Zxxx, Nxxx, Dxxx, and Exxx series) Intel® Atom™ D2xxx & N2xxx series processors Intel® Atom™ D5xx, D4xx, D2xx series processors Intel® Atom™ E6xx series processors Intel® Atom™ N4xx, N3xx, N2xx series processors Intel® Atom™ Z6xx, Z5xx series processors Intel® Atom™ processor CE 42xx, and CE 41xx Intel® Media processor CE 31xx Linux* kernel 3.0.x and 2.6.x, Yocto Project* 1.0, 1.1 incl. real-time scheduler support, and MeeGo* 1.1, 1.2 compliant OS
Programming languages	Natively supports C and C++ development
System requirements	Please refer to <a href="http://www.intel.com/software/products/systemrequirements/">www.intel.com/software/products/systemrequirements/</a> for details on hardware and software requirements.
Support	All product updates, Intel® Premier Support services and Intel® Support Forums are included for one year. Intel Premier Support gives you confidential support, technical notes, application notes, and the latest documentation. Join the Intel® Support Forums community to learn, contribute, or just browse! <a href="http://software.intel.com/en-us/forums/software-development-toolsuite-atom/">http://software.intel.com/en-us/forums/software-development-toolsuite-atom/</a> .

**Download a trial version today**  
[www.intel.com/software/products/eval](http://www.intel.com/software/products/eval)

## Optimization Notice

Notice revision #20110804

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

