Case Study
Intel®

Boosting Long Term Evolution (LTE) Application Performance with Intel® System Studio

"Intel® System Studio provides several key components to facilitate the product development, performance tuning and code debugging. It is a convenient, efficient and powerful tool for real time application development in wireless segment."

– Peng Cao,
Project Manager,
CID Wireless Shanghai Team

Challenge:

- Deliver high performance code for time-critical tasks in LTE wireless communication applications.
- Implement sophisticated algorithms to increase telecommunication network efficiency by leveraging general purpose x86 processor power.
- Build stable and efficient code that is future proof. The software needs to be able to migrate to future generations of Intel processors and chipsets to take advantage of new silicon architectures.

Solution:

- Intel® System Studio presents a wide variety of tools within a single tool chain for signal processing to application processing to obtain great computing performance, short development cycle and product simplification.

Benefit:

- Achieving the performance requirement for a LTE reference design code.
- Great efficiency for development, performance tuning, and debugging.
- Approximately six-fold increase in processing performance compared to the plain C code implementation.

How Intel System Studio was applied to deliver high performance code for the LTE application

In order to increase the speed and efficiency of telecommunications networks, modern wireless communication standards such as Long Term Evolution (LTE) apply sophisticated algorithms and demand high performance computation to process signal data. Their algorithms often include several computationally-intensive parts such as Fast Fourier Transform (FFT), multiple-input and multiple-output (MIMO), channel estimation, and turbo decoding. These operations are required to complete within a limited time slot, and their implementations need to be well optimized to utilize the full hardware computing capabilities.

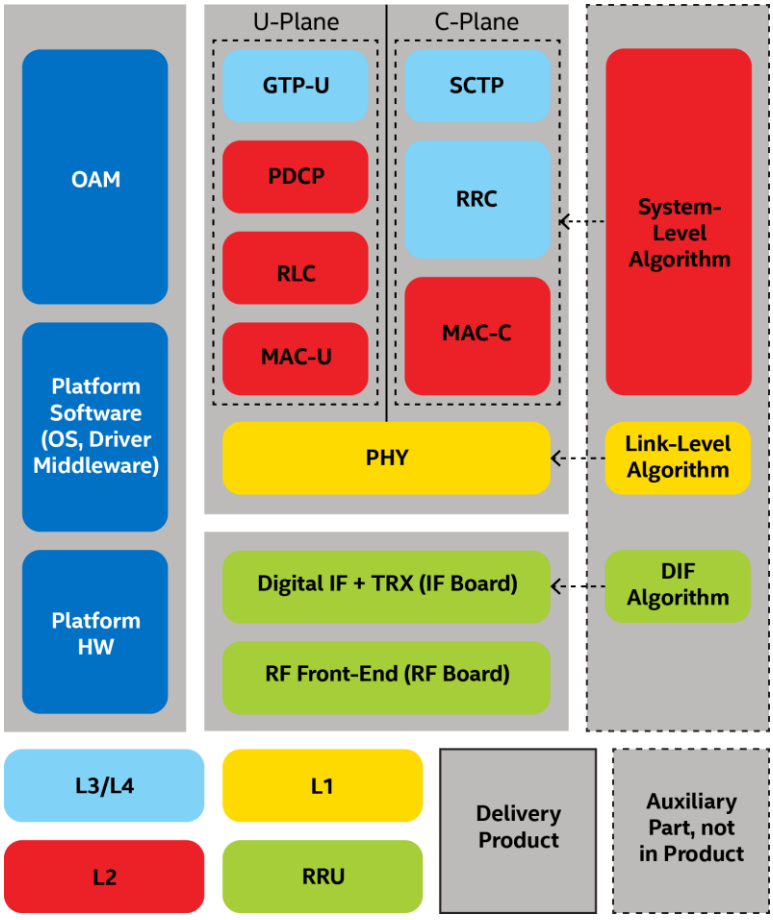

Figure 1: Evolved Node B(eNodeB) functionality example

Figure 1 shows the typical functionality partition of the Evolved Node B(eNodeB) side in a LTE network. Among all these functional modules, The L1 (physical layer) part has many real-time computation-intensive modules such as DFT, IDFT, FFT, and MIMO. With the support of Intel System Studio, these real time modules can be implemented in pure software mode and run on the latest Intel Architecture while meeting real time constraint and providing sufficient throughput. The code base guarantees that Intel Architecture has the capability and flexibility to implement and deploy LTE eNodeB functionality in the real world. Intel Communications Infrastructure Division team and Intel Labs China developed a reference design platform that provided the L1 Common Public Radio Interface connection and functions of next-generation mobile LTE radio access network with Intel Architecture. By using Intel System Studio, the reference code demonstrates considerable performance improvement. For example, the MIMO functions optimized with Intel® Advanced Vector Extensions 2 (Intel AVX2) instructions called from Intel® Integrated Performance Primitives (Intel® IPP) function provide nearly six-fold performance gain over the plain C implementation.

Intel® System Studio Solution

Intel System Studio provides a comprehensive and integrated tool suite that help developers deliver the next generation of power efficient and high performance embedded applications.

Intel System Studio includes several components for code developing, debugging, and performance tuning:

Intel IPP, the optimized building block libraries:

Delivering the efficient multiple-input and multiple-output (MIMO) code is an important part of the application. This algorithm spreads the same total transmit power over the antennas to achieve an array gain, enabling reliable operation, low energy consumption, and high data rates within limited bandwidth. The most computationally intensive kernels of these operations are based on fast small matrix computations. Intel IPP provides ready-to-use and efficient functions for the MIMO algorithm. The Intel IPP MIMO functions are simple interfaces which take a receiver signal as the input and return an estimated transmit

signal so that a minimum mean square error is achieved. By calling such functions, users do not need to care about the low level code tuning to acquire the high efficient code.

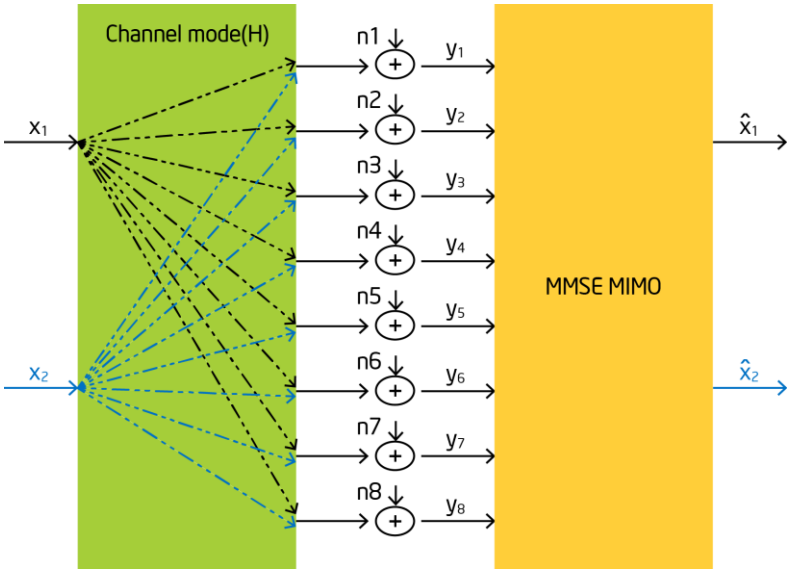

Figure 2: Multiple-input and Multiple-output

The Intel CID Wireless software teams benchmarked their reference software using Intel IPP, which showed considerable speedups: They compared pure C implementation code with the IPP functions. The test was performed on Intel Xeon® E5-2630L processors and measured several Intel IPP MIMO functions. Compared with pure C implementation code, the IPP functions provide about six-fold performance gain due to the optimization of the Intel Advanced Vector Extensions 2 (Intel AVX2) instructions.

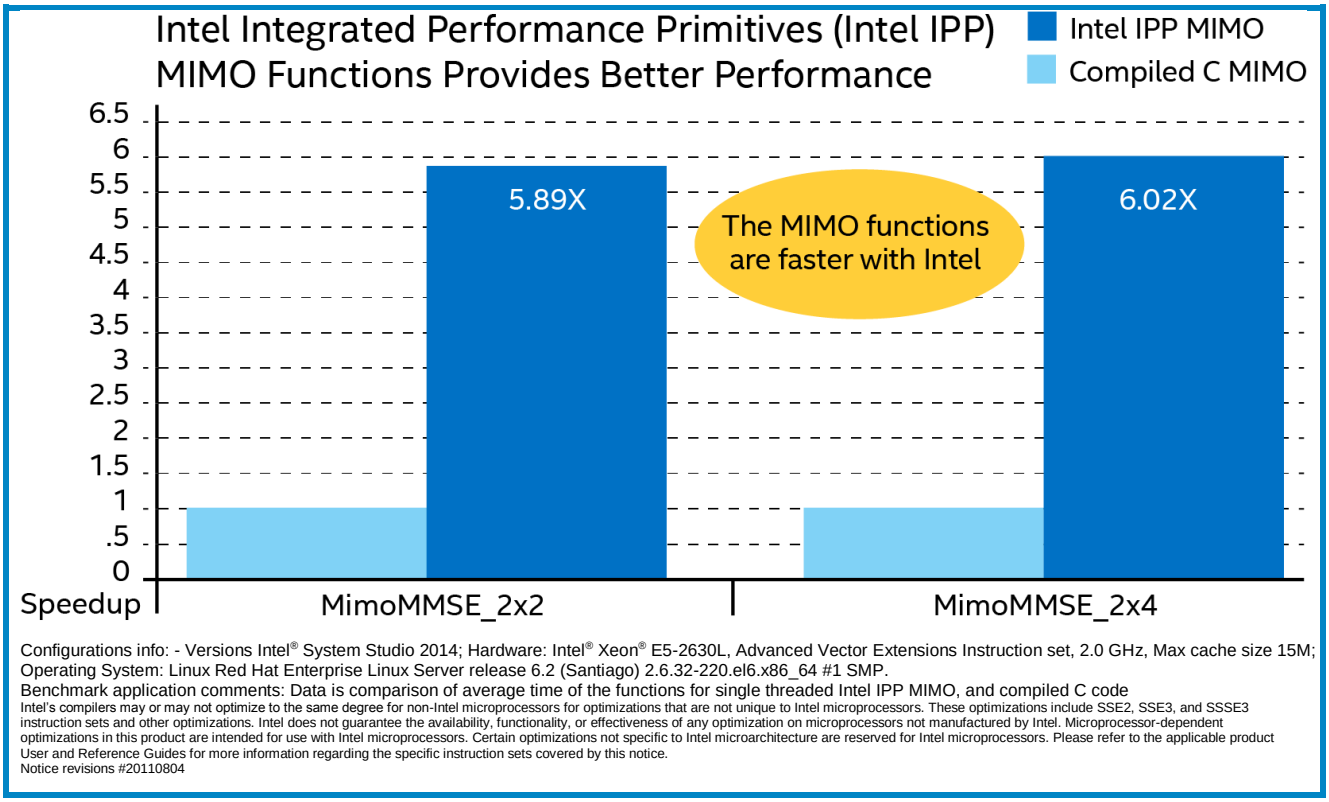

Figure 3: The MIMO functions performance comparison between C code and Intel IPP functions.

In addition to the MIMO functions, Intel IPP provides a diverse range of other signal processing functions including Discrete Fourier Transform, single filtering, convolution, and sampling. These functions provide easy and high efficient ways to process the signal data for the embedded applications. The software developers only need to focus on the implementation of the high level functionality, while Intel IPP provides the low level high optimized building blocks for high performing code.

The leading compiler for code vectorizaton: High performance wireless communication code needs to utilize Single Instruction Multiple Data (SIMD) instructions to process data in parallel. Multiple data values are loaded into the SIMD registers, and then operations are performed on all data elements at once. As one of the components of Intel System Studio, the Intel® C++ Compiler offers a rich set of machine independent and machine specific optimizations to maximize performance. Intel C++ compiler can generate the vectorized code automatically, and also supports programming for intrinsics which allow developers to write their own implementation with SIMD instructions directly.

The Intel wireless eNodeB signal processing reference design code was compiled with the Intel C++ Compiler, providing a substantial performance increase due to autovectorization. For example, the major data type in the reference design code is 16-bit fixed point. With the Intel AVX2 instructions, each operation can handle 16 data values instead of 1 data value for scalar, as shown in the Figure 4.

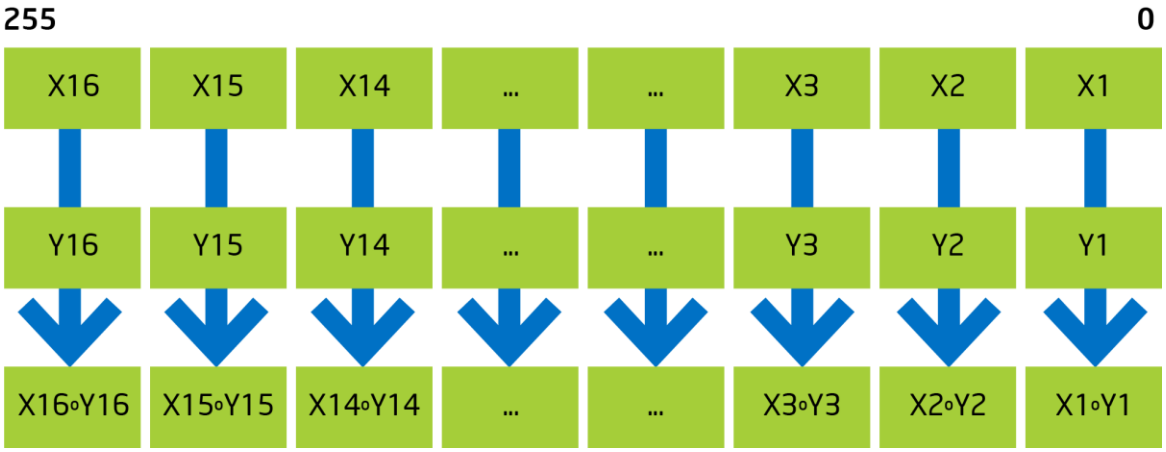

Figure 4: Processing data in parallel with Intel SIMD instructions

“The eNodeB signal processing reference design code is very sensitive to real time performance. Intel C++ Compiler maximizes our code performance to meet real time constraints in the application via support of high efficiency instructions from the latest Intel architecture. It’s an outstanding tool to demonstrate high performance, portability and scalability of Intel architecture in the Telecom area", said Peng Cao.

System performance and power analysis: For embedded applications, Intel® VTune™ Amplifier 2014 for Systems provides ways to identify and locate performance bottlenecks in the code. It allows the developer to analyze in-depth CPU, GPU and System-on-Chip (SOC) activities, and events. To understand how power is consumed in the application, the Intel® Energy Profiler can identify wake-up causes, timers triggered by application, and interrupts. The LTE eNodeB software teams used Intel® VTune™ Amplifier to collect and display software performance data, hardware events with call stacks, and also find hot spots even in small functions. VTune Amplifier shows the results and problem with the related source code, so it is easy to quickly understand the performance problem of the application.

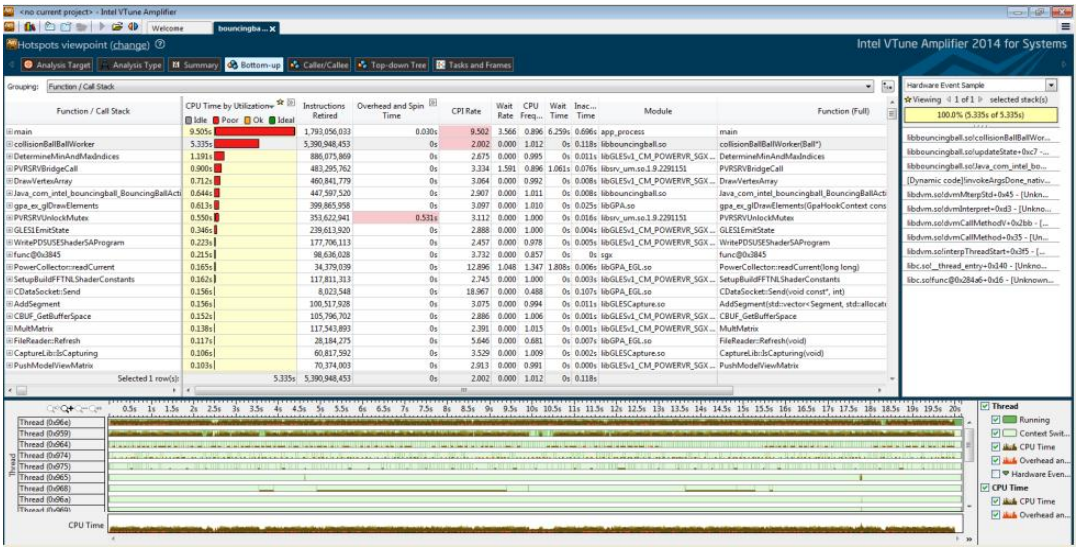

Figure 5: Intel® VTune™ Amplifier for Systems provides advanced performance data and visualized results to analyze the performance problems.

Conclusion

Intel System Studio components were used to optimize the LTE wireless software stack. The tool suite provides developers valuable tools to implement embedded solutions that are reliable, high performance, and power efficient. Intel System Studio provides a simple approach for software teams to deliver high performance LTE reference code with enhanced productivity.

- Speed-up development and testing with deep hardware and software insights.
- Enhance code stability using in-depth system-wide debuggers and analyzers.
- Boost power efficiency and performance using system-wide analyzers, compiler and libraries.

To learn more visit: http://intel.ly/system-studio

Software

Printed in USA 0414/BLK/CMD/PDF Please Recycle 330597-001US