MEASURABLE ADVANTAGE™

The Industry's First Floating-Point FPGA

The FPGA has long been known for its massive digital signal processing (DSP) capabilities in fixed point. Designers can capitalize on this processing power even more with a recent Altera® technology breakthrough—the industry's first floating-point FPGA. The company's newest FPGAs now can natively support IEEE 754 single-precision floating point using dedicated hardened circuitry. This new capability offers designers the ability to implement their algorithms in floating point with the same performance and efficiency as fixed point. This has been achieved without any power, area, or density compromises, and with no loss of fixed-point features or functionality.

Floating-Point Performance and Features

The key technology lies at the core of Altera's Generation 10 FPGAs. The award-winning Altera variable-precision DSP blocks have now been enhanced to include a single-precision adder and single-precision multiplier in every DSP block. With thousands of floating-point operators built into these hardened DSP blocks, Arria® 10 FPGAs are rated from 140 GigaFLOPS (GFLOPS) to 1.5 TeraFLOPS (TFLOPS) across the 20 nm family. Altera's 14 nm Stratix® 10 FPGA family will use the same architecture, extending the performance range right up to 10 TFLOPS, the highest ever in a single device.

The floating-point computational units, both multiplier and adder, are seamlessly integrated with existing variable-precision fixed-point modes. This provides a 1:1 ratio of floating-point multipliers and adders, which can be used independently, as a mult-add, or as a mult-accumulator. Designers still have access to all the fixed-point DSP processing features used in their current designs, but for superior numerical fidelity and dynamic range, can easily upgrade all or part of the design to single-precision floating point as desired. Since all the complexities of IEEE 754 floating point are within the hard logic of the DSP blocks, no programmable logic is consumed, and similar clock rates as used in fixed-point designs can be supported in floating point, even when 100 percent of the DSP blocks are used.

Special vector modes are also supported by columns of floating-point DSP blocks operating in unison. These vector modes can be used to support typical linear algebra functions used in high-performance computing applications, as well as more traditional FPGA functions such as highly parallel fast Fourier transform (FFT) or finite impulse response (FIR) filter implementations. The structures are designed to maximize the use of both the floating-point multiplier and adder in each block, allowing the designer to achieve as close as possible to the peak GFLOPS rating of a given Altera FPGA.

Altera provides a comprehensive set of floating-point mathematical functions. Approximately 70 math.h library functions, compliant with the OpenCL™ 1.2 specification, are optimized for the new hardened floating-point architecture. These functions leverage the hard memory and DSP blocks in the FPGA, using almost no FPGA logic. This ensures consistent, low-latency, high f_{MAX} implementations, even in packed FPGA designs.

Productivity Benefits

Native floating-point support is of great significance to designers implementing complex, high-performance algorithms in FPGAs. All algorithm development and simulations are performed in floating point prior to building a system. Once the algorithm simulation is completed, there is typically a further 6-12 month effort to analyze, convert, and verify a floating-point algorithm in a fixed-point implementation. This amount of effort is often required to overcome three main problem areas.

First, the floating-point design must be converted manually to fixed point, which requires an experienced engineer. Even then, the implementation will likely not have the same numerical accuracy as the simulation.

Second, any later changes in the algorithm must be converted manually again. Also, any steps taken to optimize the fixed-point algorithm in the system are not reflected in the simulation.

Third, as problems arise during system integration and testing, the possible causes could be any of the following: an error-in-hand conversion process, a numerical accuracy problem, or the algorithm itself is just defective. Isolating the problem can be quite difficult. All of these issues can be eliminated by using Altera's floating-point FPGAs.

Comparison to GPGPUs

The natural competition to the Altera floating-point FPGA is not other competing FPGAs, but general-purpose graphics processing units (GPGPUs). The "soft" floating-point implementation offered by other FPGA vendors, using logic to implement the complex floating-point circuitry, is simply not competitive or efficient. The appropriate analogy would be the FPGAs of years ago without hard multipliers, trying to compete against modern FPGA architectures with DSP blocks.

However, several years ago, graphic processing unit (GPU) vendors incorporated floating point into their computational units, achieving great degrees of floating-point processing and levels of single-precision performance similar to Altera FPGAs. These devices became known as GPGPUs, as they are no longer just graphics engines but general-purpose computing accelerators.

While a common design flow, known as OpenCL, can be used for FPGAs and GPGPUs, there are major differences in how the algorithms are implemented. GPGPUs use a parallel processor architecture, with thousands of small floating-point mult-add units operating in parallel. The algorithm is broken up into tens of thousands of threads, which are mapped to the available computational units as the data is made available.

On the other hand, Altera FPGAs use a pipelined logic architecture where the thousands of computational units are arranged into typically into a streaming dataflow circuit, operating on vectors. An FFT core or Cholesky decomposition core would be an example. Each of these cores produces a vector wide of output data each clock cycle, with the vector width determined by the designer.

GPGPUs tend to operate efficiently on algorithms where the ratio of computation to I/O is very high. Since the host GPU must provide data over a PCIe® link to the GPU, the GPU can become data starved unless there is a high degree of calculations to be done on each data.

FPGAs are relatively new to high-performance computing, but have compelling advantages. First, due to the pipelined logic architecture, the latency for processing a given data stream is much lower than on a GPU. This can be a key advantage for some applications, such as financial trading algorithms.

Second, FPGAs have superior GFLOPS/W capability than GPGPUs, and this can be critical in applications that are not environmentally controlled, such as avionics. This also means that for a given power budget, the FPGA can typically perform far more computations than a GPGPU.

Third, the FPGA has an incredibly versatile and ubiquitous connectivity. The FPGA can be placed directly in the datapath and process the data as it streams through. For example, the FPGA can interface directly to the feeds of an array antenna and perform both fixed and floating-point processing, while communicating over fiber or backplane links with other system components. In fact, Altera has specifically added the option of data streaming to their OpenCL tools, which is in compliance with the OpenCL vendor extension rules.

Design Flows for Floating Point

Designers can access the floating-point FPGA features using a variety of design flows. For example, hardware designers who may just need a few floating-point mathematical functions or FFT cores can utilize the Altera MegaCore® functions which are available today.

For hardware or system engineers, Altera also offers a model-based flow using their DSP Builder Advanced Blockset and MathWorks' MATLAB and Simulink tools. This tool flow allows the engineer to design, simulate, and implement entirely within the MathWorks environment, and provides native support for vectors needed in linear algebra applications. Meanwhile, for GPU designers, as previously mentioned, OpenCL provides access to FPGAs without the need to become familiar with the FPGA architecture details.

All of these tool flows are available today and support most of Altera's FPGA families. Performing a recompile and targeting an Arria 10 FPGA using Altera's Quartus® II software version 14.1 will seamlessly map onto the hard floating-point DSP blocks, providing the huge benefits of a native floating-point FPGA.

OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos.

Acknowledgements: Michael Parker, Principal DSP Planning Manager, Altera Corporation