MEASURABLE ADVANTAGE™

Stratix 10: The Most Powerful, Most Efficient FPGA for Signal Processing

FPGAs enjoy a well-deserved reputation for highly parallel, high-throughput digital signal processing (DSP). This capability has been steadily increasing over past generations of FPGA devices. However, occasionally a revolutionary, rather than evolutionary, new product is introduced. Altera's new Stratix® 10 FPGA and SoC family certainly fits that description. Stratix 10 devices deliver up to 23 TMACs of fixed-point performance and up to 10 tera floating point operations per second (TFLOPS) of single-precision floating-point performance making these devices the highest performance DSP devices with a fraction of the power of alternative solutions, such as graphic processing units (GPUs) and dedicated DSP. Stratix 10 customers can expect to see up to an order of magnitude improvement in giga floating point operations per second (GFLOPS)/Watt in their actual designs compared to competitive GPU solutions.

This new device combines a number of innovations to deliver this breakthrough performance. In addition to a high number of DSP resources, Stratix 10 devices include densities from 500 kLE to 5.5 MLE and a large amount of on-chip memory all in a monolithic core fabric design, fabricated on Intel's 14 nm Tri-Gate process, industry's most advanced semiconductor process technology. The Stratix 10 device includes the ground breaking HyperFlex™ core architecture to enable a 2X increase of core clock frequency (f_{MAX}) for FPGA designs across a wide range of applications, compared to previous generation high-performance FPGA products. The Stratix 10 core hard blocks, such as memory and DSP blocks, have been designed to support up to 1 GHz operation, taking advantage of the new HyperFlex capabilities. The DSP architecture continues native support for 18 bit and 27 bit fixed point, with 64 bit accumulators, the largest in any FPGA. For even more dynamic range, the native floating-point architecture first introduced in the Arria® 10 device family is extended to the Stratix® 10 device family, supporting IEEE 754 single-precision floating point using dedicated hardened circuitry. This new capability offers designers the ability to implement algorithms in floating point with the same performance and power 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 the Altera's Generation 10 FPGAs. The award-winning Altera variable-precision DSP block includes a single-precision adder and single-precision multiplier in every DSP block. The existing midrange Arria 10 FPGAs are rated from 140 GFLOPS to 1.5 TFLOPS across the 20 nm family. Altera's new 14 nm Stratix 10 FPGA family, with over 10 thousand floating-point operators built into these hardened DSP blocks will use the same architecture, the highest degree of floating-point computational parallelism ever achieved in a single device.

Also new is the DSP block performance. With Stratix 10 devices, all fixed-point modes operate at a sustained 1 GHz frequency and all floating-point modes operate at a sustained 800 MHz frequency. These remarkable clock rates, coupled with the extreme density of DSP blocks enabled by 14 nm Tri-Gate process technology, are what drives the 11.5 TMAC (or 23 TMAC when using pre-adder) and 9.3 TFLOPS of peak performance of the 2.8 MLE Stratix 10 FPGA family member. Power efficiency is also unprecedented – estimated at 80 GFLOPS per Watt, a fraction of competing GPU solutions.

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 utilized.

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 like highly parallel fast Fourier transform (FFT) or finite impulse response (FIR) filter implementations. The structures are designed to maximize 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 math functions. Approximately 70 math.h library functions, compliant to Open Computing Language (OpenCL™) 1.2, 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 high-utilization 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. Without native floating-point support, 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. The design must be first hand converted to fixed-point conversion, which requires an engineer experienced in numerical analysis and stability. Even then, the implementation will likely not have the same numerical accuracy as the simulation. Any later changes in the algorithm must be hand ported again and any steps taken to optimize the fixed-point algorithm in the system are not reflected in the simulation. 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 a problem with the algorithm itself. Isolating the problem can be quite difficult. All of these issues can be eliminated, or largely mitigated by using Altera's floating-point FPGAs.

Comparison to GP-GPUs

The natural competition to the Altera floating-point FPGA capabilities is not other competitors' FPGAs, but general-purpose graphic processing units (GP-GPUs). Competing FPGA vendors' soft floating-point implementations, using logic to implement the complex floating-point circuitry, is simply not competitive or area efficient. The appropriate analogy would be the FPGAs of years ago, which lacked hard multipliers trying to compete against modern FPGA architectures with DSP blocks.

However, several years ago, GPU vendors incorporated floating-point into their computational units, achieving great degrees of floating-point processing, with levels of floating-point performance exceeding 1 TFLOP. These devices became known as GP-GPUs, 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 GPUs, there are major differences in how the algorithms are implemented. GP-GPUs use a "fine grained" 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.

Altera FPGAs use a "course grained" architecture where the thousands of computational units are arranged into typically a few dozen highly pipelined structures, operating on vectors. An FFT core or Cholesky decomposition core would be an example. Each of these cores produce a vector wide of output data each clock cycle, with the vector width determined by the designer.

GP-GPUs 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 PCI Express® (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. GP-GPUs often have extensive libraries, and are available as plug-in server cards.

FPGAs are relatively new to high-performance computing, but have compelling advantages. First, due to the coarse grained 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 datacenter acceleration, or more embedded applications such as radar processing.

Second, FPGAs have a much better GFLOPS/W capability than GP-GPUs, and this can be critical in applications that are not environmentally controlled – such as avionics for example. This also means for a given power budget, the FPGA can typically perform far more computations than a GP-GPU.

Third, the FPGA has 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 in communication over fiber optic or backplane links to other system components. In fact, Altera has specifically added options of data streaming to their OpenCL tools, in compliance with 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 math functions or an FFT core can utilize Altera megafunctions and MegaCore® intellectual property (IP) cores, which are available today.

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

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

Acknowledgements

Michael Parker, Principal DSP Planning Manager, Altera Corporation