

Implementing digital signal processing (DSP) datapaths with different performance, precision, intellectual property (IP), and development flows is challenging and labor-intensive. As more and more high-performance DSP datapaths are implemented on FPGAs, Altera has developed a complete DSP solutions portfolio at 28 nm to address these challenges and speed up the design cycle for FPGA-based applications. This white paper discusses the different components of this portfolio and how they come together to accelerate the implementation of a DSP design.

Introduction

Although signal processing is usually associated with digital signal processors, it is becoming increasingly evident that FPGAs are taking over as the platform of choice in the implementation of high-performance, high-precision signal processing. Accordingly, FPGA vendors are beginning to include hard multipliers and DSP blocks within their core silicon architecture. IP cores are also provided to assist traditional functions such as finite impulse response (FIR) and fast Fourier transforms (FFTs).

As a result, a wide range of applications are now relying on FPGAs as the key signal processing platform. These applications, shown in [Figure 1](#), share one thing in common—the performance requirements exceed the capabilities of a traditional programmable digital signal processor.

Figure 1. Different Applications Need Different Performance, Precision, IP, and Tools



These systems not only have different performance and precision requirements, but also different design and development flows. For example, video processing requires 9- to 10-bit precision, with some high-end designs needing a 16-bit color depth. These designs are generally created in a HDL design flow, with video- and image-processing IP functions increasingly utilized to speed up the development flow. On the other side of the spectrum, military radar designs require the highest DSP performance and floating-point precision to get the highest dynamic range. Many of these designs are modeled in the popular MATLAB and Simulink tools, along with floating-point functions that are optimized for the FPGA architecture.

Altera's Total 28-nm DSP Portfolio

The biggest challenge faced by FPGA vendors is in providing a complete DSP solution portfolio—one that not only includes a DSP silicon architecture that is configurable, but also a range of tools, IP, and building blocks that can help designers to quickly and efficiently complete the implementation of their algorithms. To support the 28-nm Stratix® V, Arria V, and Cyclone V FPGAs, Altera offers a total DSP portfolio, which, as illustrated in Figure 2, comprises a variable-precision DSP architecture, the DSP Builder Advanced Blockset, a video design framework, and a comprehensive suite of floating-point IP.

Figure 2. Industry's First 'Total' DSP Portfolio



Variable-Precision DSP Architecture

The basic principle behind Altera's DSP solutions portfolio is the recognition that one size does not fit all, that it is necessary to understand the diverse needs and preferences of customers in the design and development environment. Signal processing applications have different precision requirements and different precision levels at different stages of the signal processing data-paths. For example, video broadcast applications can efficiently use multipliers ranging from 9x9 to 18x18. Other applications, such as wireless and medical systems, that develop complex, multi-

channel filters, require a higher precision as there is a need to maintain data precision after each stage of the filter. Apart from these, there are also applications in the military, test, and high-performance computing industries that demand both performance and precision, sometimes requiring single- or even double-precision floating-point to implement complex matrix operations and FFTs.

To address the precision requirements of various DSP applications on the entire spectrum, Altera architected the industry’s first variable-precision DSP block. It is the first DSP block in the market to have two native precision modes, 18-bit precision mode and high-precision mode, illustrated in Figure 3 and Figure 4. This unique feature provides backward compatibility with previous 40-nm DSP blocks, as well as efficient support for emerging signal processing applications of higher precisions. In addition, the variable precision blocks for Stratix V, Arria V, and Cyclone V devices are optimized for various applications.

Figure 3. Arria V and Cyclone V 18-Bit Precision and High-Precision Modes

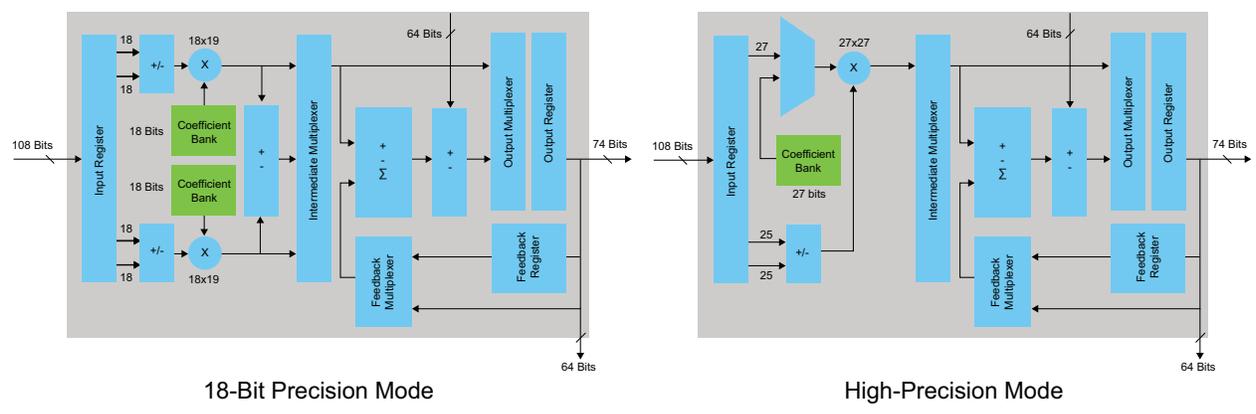
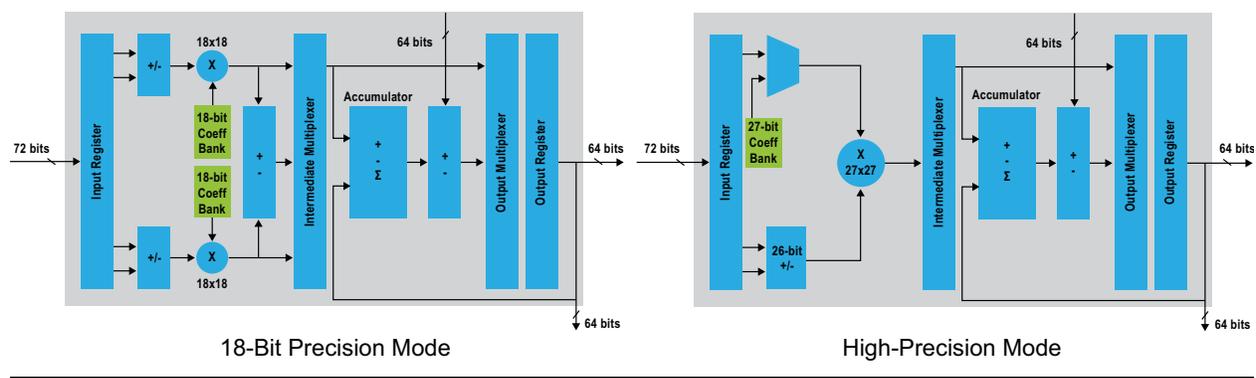


Figure 4. Stratix V 18-bit Precision and High-Precision Modes



A single variable-precision DSP block at 28nm can support precisions ranging from 9x9 to 27x27. In addition, the precision of each block within a device can be independently configured to support bit growth in various designs such as FIR and FFTs. This block is called “variable” because its precision is configurable by the customer on a block-by-block basis. This is a powerful new concept because FPGAs traditionally force the designer to adapt the algorithm to the block architecture, which results in either a suboptimal implementation or the need to modify the algorithm.

Legacy fixed-precision DSP architectures can support only one precision. As such, the designer either wastes resources when the precision requirement of the algorithm is lower, or settles for lower performance by cascading multiple blocks when the precision requirement is higher. In such a situation, only a DSP block with configurable precision is able to provide system performance within stringent cost and power budgets.

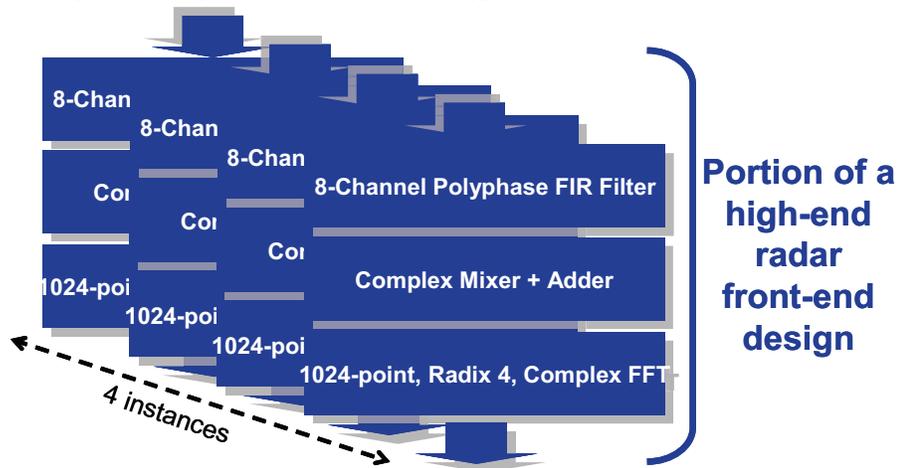
The increasing need for higher precision and complex multiplication operators in high-performance datapaths is also taken into consideration in the design of the variable-precision DSP block. To enable the cascading of multiple DSP blocks, the variable-precision block was designed with the industry’s only 64-bit cascade bus and adder. This design allows the implementation of large complex multipliers and floating-point signal processing functions with 50 percent fewer resources than the competing 18x25 architecture.

DSP Builder Advanced Blockset

Altera’s DSP Builder tool provides support for high-level, Simulink-based synthesis, timing-driven netlist optimizations, and a complete floating-point design flow for FPGAs. Netlist optimization is a unique feature of the DSP Builder tool that allows the designer to specify the desired f_{MAX} (clock frequency) and latency of the system and leave the rest of the work to the tool. The DSP Builder tool includes the necessary registers needed to increase the f_{MAX} of critical paths to meet latency constraints. As a result, no more time-consuming hand-tweaking of the HDL code is necessary as changes can be made with the push of a button.

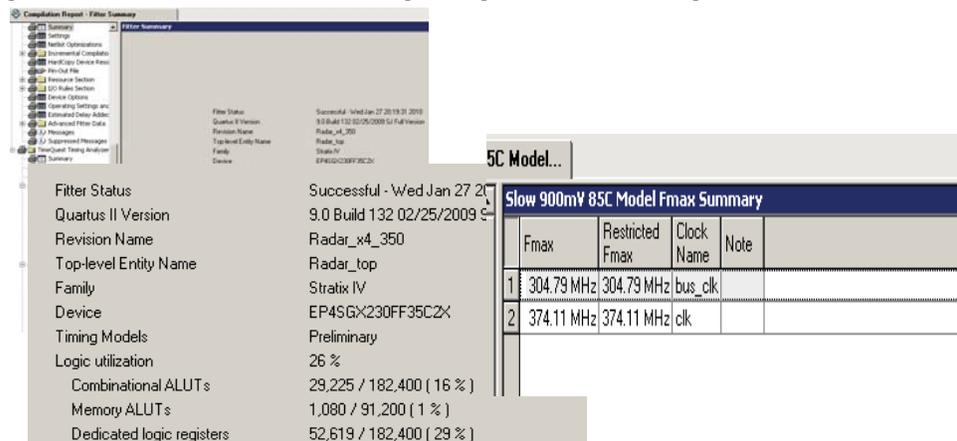
The resulting productivity gain can be best illustrated with an example radar design that meets timing at 350 MHz using the DSP Builder tool. [Figure 5](#) shows a portion of a radar design jointly developed by Altera and The MathWorks to be implemented in an Stratix V FPGA with a target f_{MAX} of 350 MHz.

Figure 5. Large DSP Design for a Radar Front-End Application



Typically, this f_{MAX} constraint can only be met by hand-tweaking the HDL code to add the necessary registers and resources. However, with the DSP Builder tool, designers now have an automated way of meeting the performance goal. The compilation report in Figure 6 shows the large design, comprising about 60K logic elements (LEs), achieving a system f_{MAX} greater than 350 MHz without the need for manual hand-tweaking of the HDL code.

Figure 6. Automatic Generation of a Large Design that Closes Timing at >350 MHz



This system f_{MAX} was achieved by following these steps:

1. The datapath was built in Simulink using building blocks from the DSP Builder library and simulated to make certain it conformed to the algorithm.
2. The f_{MAX} of the total system was set to 350 MHz in the Parameters file (.params) file in Simulink, signaling the DSP Builder tool to optimize the implementation for the specified performance. The system implementation constraints were added at a higher level of abstraction within the high-level Simulink design description.
3. After clicking “DSP Builder,” the Simulink design description was analyzed, and both a HDL code and a bitstream were generated for the Stratix V FPGA. The timing constraints (in this case, f_{MAX}) were incorporated. Pipeline registers and the correct amount of time-division multiplexing were automatically added to meet or even exceed the specified f_{MAX} .

The designer can also efficiently run multiple “what-if” scenarios with the DSP Builder tool. To do so, all that is necessary is to change the f_{MAX} settings, latency settings, target device architecture, and even design parameters such as the number of channels, by editing the top-level parameter file in MATLAB and Simulink. Once satisfied with the performance, latency, and device utilization, the designer can either choose to use that HDL code for the datapath, or to further tweak the code to meet additional system goals. In either case, the implementation design cycle is reduced tremendously.

Video Design Framework

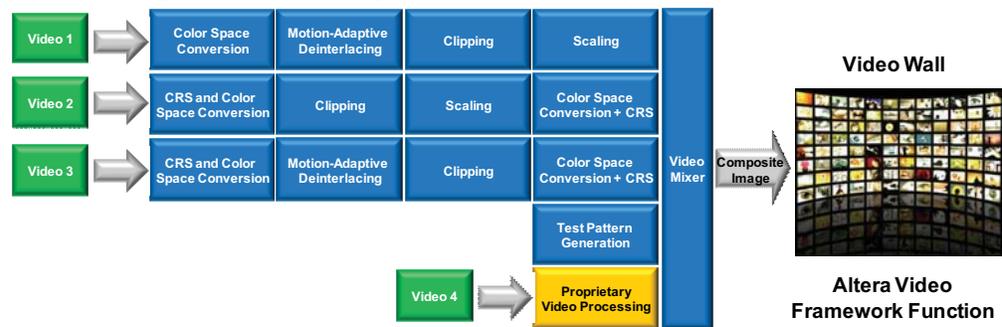
As the world of video makes a transition to 1080p high-definition (HD) resolutions, FPGAs are ideal platforms for video processing. Altera anticipated this transition nearly four years ago and invested in a video design framework, shown in Figure 7, that edged out Xilinx’s design tools to win the prestigious 2009 EDN Innovation Award.

Figure 7. FPGA Industry’s Only Video Design Framework



Altera’s video design framework is currently the only one in the market that includes 18 video functions, a streaming video interface standard, six hardware-verified reference designs, and a range of video development kits. To date, over 100 active customers are using this video design framework in their systems.

Figure 8 shows an example customer design using the Altera® video design framework. The end system is a video wall that incorporates multiple video sources, also known as a composite video. Such video walls are not only common in outdoor advertising monitors, but also in medical, military, and broadcast applications. As the individual videos come from different sources, they must be processed differently—some video sources need to be de-interlaced and scaled, others are progressive to begin with and need only be scaled, while some others may need to be custom processed. All the sources are then stitched together to form a composite image that is within the user’s control.

Figure 8. Using Altera's Video Design Framework to Develop a Custom Video Wall**Note:**

(1) Image: Apantac LLC

To build the rather complex video signal chain, the building blocks and the open-streaming interface of Altera's video design framework were used. Key MegaCore® functions from Altera's Video and Image Processing Suite can be linked together to create one video path, while the other path can be fully customized. Both video streams can then be alpha blended to create the composite video stream.

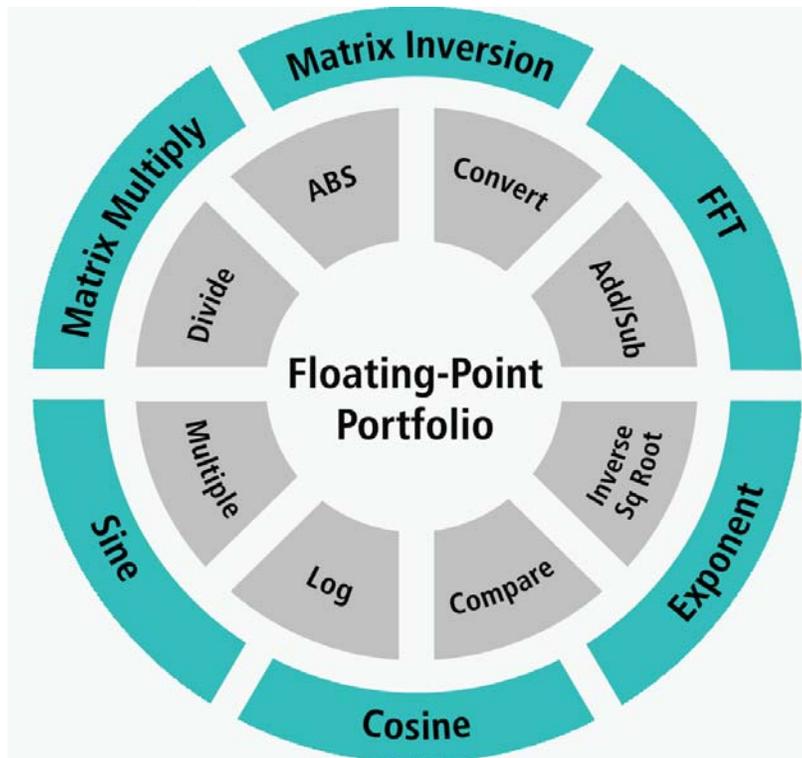
Comprehensive Floating-Point IP

In the high-performance DSP domain, floating-point signal processing is slowly but surely being seen as a way to increase dynamic range. Altera's internal research shows that almost half of high-performance DSP designs using FPGAs, such as advanced military space-time adaptive processing (STAP) radar, MIMO equalization for LTE channel cards, and high-performance computing boxes, require higher than 18-bit precision.

Floating-point processing generally involves mantissa multiplication, mantissa normalization and de-normalization, and exponent addition. While exponent addition and subtraction operations are straightforward, mantissa multiplication and normalization require higher than 24-bit precision multipliers. In order to perform these operations, traditional FPGA architectures that are limited to the 18x25 precision must be cascaded to implement a single-precision mantissa multiplication.

Altera's new variable-precision architecture can implement single-precision floating-point mantissa multiplications in a single block, thus allowing for a very high-performance design.

Figure 10. Altera's Floating-Point Portfolio



It is possible to achieve very high f_{MAX} and low latency for these functions as they are optimized for Altera device architectures. For large matrix multiplication functions such as 64x64, an f_{MAX} as high as 380 MHz can be obtained.

Summary

The “DSP in FPGAs” concept spans across different industries and different applications of different performance, precision, IP, and tool flow requirements. Because today’s FPGA vendors are expected to meet the customer’s need for a complete DSP portfolio that includes IP, tools, building block functions, and configurable DSP blocks to enable rapid design implementation and debug, Altera has developed a unique and differentiated “total” DSP solutions portfolio, in conjunction with the 28-nm Stratix V, Arria V, and Cyclone V FPGAs, to enable high-performance DSP designs for a wide range of markets and applications.

Further Information

- Stratix V FPGAs: Built for Bandwidth:
www.altera.com/products/devices/stratix-fpgas/stratix-v/stxv-index.jsp
- Literature: Arria V FPGAs:
<http://www.altera.com/products/devices/arria-fpgas/arria-v/arrv-index.jsp>
- Literature: Stratix V Devices:
www.altera.com/products/devices/stratix-fpgas/stratix-v/literature/stv-literature.jsp
- DSP Solutions:
www.altera.com/technology/dsp/dsp-index.jsp

- Altera’s Total 28-nm DSP Portfolio: Fastest Path to Highest Performance Signal Processing:
www.altera.com/b/28-nm-dsp-portfolio.html
- Webcast: “Accelerate your FPGA-Based DSP Designs”:
www.altera.com/education/webcasts/all/wc-2010-accelerate-fpga-dsp-designs.html
- Video and Image Processing (VIP) Suite MegaCore Functions:
www.altera.com/products/ip/dsp/image_video_processing/m-alt-vipsuite.html

Acknowledgements

- Suhel Dhanani, Sr. Manager, Embedded Marketing, Altera Corporation
- Jordon Inkeles, Senior Manager, Software and DSP Marketing, Altera Corporation

Document Revision History

Table 1 shows the revision history for this document.

Table 1. Document Revision History

Date	Version	Changes
April 2011	1.1	Added details for Arria V, Cyclone V, and variable-precision DSP blocks.
July 2010	1.0	Initial release.