FIR Compiler MegaCore Function

Solution Brief 41 **June 1999, ver. 1**

Target Applications:
Cellular base stations, spread-spectrum communications, set-top boxes, and several other digital signal processing (DSP) applications

Family:
APEX™ 20K, FLEX® 10K, FLEX 8000, and FLEX 6000

Ordering Code:
PLSM-FIR

Vendor:

101 Innovation Drive
San Jose, CA 95134
http://www.altera.com
Tel. (408) 544-7000

Features

- Fully integrated finite impulse response (FIR) filter development environment
- Highly optimized for Altera® device architectures
- Supports parallel or serial arithmetic architectures
- Supports any number of taps
- Includes a built-in coefficient generator
- Imports integer or floating-point coefficients from third-party tools
- Supports multiple coefficient scaling algorithms
- Supports coefficient widths from 4 to 32 bits of precision
- Supports signed or unsigned input data, with widths from 4 to 32 bits wide
- Permits user-selectable output precision via rounding and saturation
- Determines symmetry and selects appropriate architecture automatically
- Creates MATLAB Simulink, VHDL, and Verilog HDL simulation models
- Generates Quartus™ and MAX+PLUS® II vector files
- Includes an impulse, step function, and random input testbed
- Provides dynamic resource estimates

General Description

Many digital systems use signal filtering to remove unwanted noise, to provide spectral shaping for communications channels, or to perform signal detection or analysis. FIR filters are used in systems that require a linear phase and have an inherently stable structure. Typical filter applications include signal preconditioning, band selection, and low-pass filtering.

The filter design process involves identifying coefficients that match the frequency response specified for the system. The coefficients determine the structure of the filter. You can change which signal frequencies pass through the filter by changing the coefficient values or adding more coefficients.

A fully parallel, pipelined FIR filter implemented in a programmable logic device (PLD) can operate at data rates above 100 megasamples per second (MSPS), making PLDs ideal for high-speed filtering applications. The FIR compiler MegaCore™ function has an interactive wizard-driven interface that allows you to create custom FIR filters easily. The wizard outputs simulation files for use with third-party tools, including MATLAB Simulink. The FIR compiler MegaCore function speeds up the design cycle by:

- Finding the coefficients needed to design custom FIR filters.
- Generating clock-cycle-accurate FIR filter models (also known as bit-true models) in the Verilog HDL and VHDL languages, and for the MATLAB environment (M-files and Model Files).
- Automatically generating the code required for the Quartus or MAX+PLUS II software to synthesize high-speed, area-efficient FIR filters of various architectures.
- Creating standard test vectors (i.e., impulse, step, and random input) to test the FIR filter's response.

Functional Description

You can run the FIR compiler wizard using the Quartus or MAX+PLUS II MegaWizard™ Plug-In Manager. After you set the function's parameters, the wizard generates a customized function that can be instantiated in your design file. Table 1 describes the options for the FIR compiler wizard.

Table 1. FIR Compiler Wizard Options

Page	Description
Input Data Type	The width of the input data bus (from 4 to 32 bits wide). Specifies a signed or unsigned bus.
Coefficients	The filter coefficients can be read from a file or generated using the FIR compiler wizard. In both cases, you can scale the coefficients and indicate the precision bits. The wizard detects the filter symmetry and selects an appropriate architecture. The function lets you specify the sample rate (either in Hertz or in relation to the Nyquist rate), the number of taps, and cut-off frequencies. The function supports low-pass, high-pass, band-pass, and band-reject filters. The function also supports rectangular, Hanning, Hamming, and Blackman windows. As you change the coefficient settings, you can view the frequency and response of the filter dynamically.
Limiting Precision	The function determines the output bit width based on the actual coefficient values and the input bit width. You can reduce your filter's precision by removing bits from the most significant bit (MSB) via truncation or saturation, or from the least significant bit (LSB) via truncation or rounding.
Architecture	You can indicate whether the filter is parallel or serial, and the number of channels for the filter. APEX 20K and FLEX 10K devices contain embedded system blocks (ESBs) and embedded array blocks (EABs), respectively, which are ideal for use with serial filters. Both parallel and serial filters allow pipelining, which lets you tradeoff between area and speed.
Simulation Output Files	The function generates several types of simulation files, including MAX+PLUS II Vector Files (**.vec**), MATLAB and Simulink models, MATLAB testbench files, Verilog HDL models, and VHDL output files.

Figure 1 shows the coefficient generator and MegaWizard Plug-In for the FIR compiler MegaCore function. From the coefficient generator, you can specify the function's sample rate, number of taps, cut-off frequency, filter type, and window method.

Figure 1. FIR Compiler Coefficient Generator

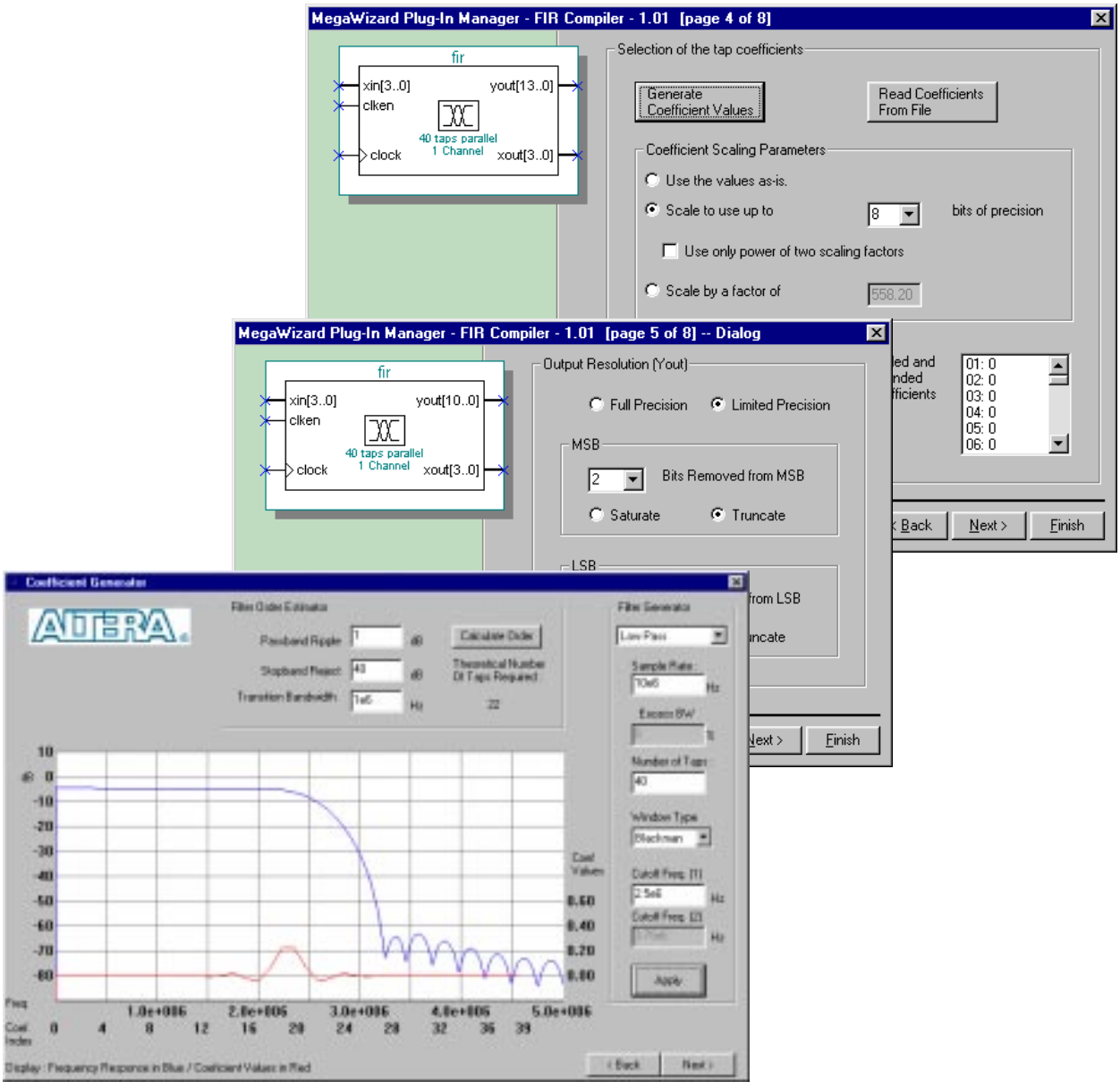
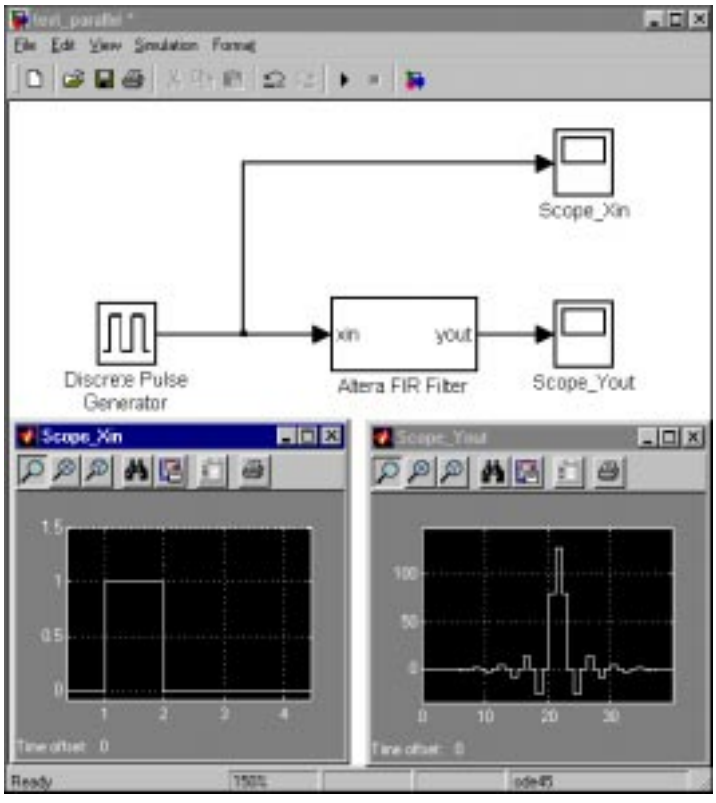


Figure 2 shows the MATLAB Simulink interface for the FIR compiler function.

Figure 2. System-Level Simulation with the MATLAB Simulink Interface



Performance

Table 2 describes the logic element (LE) requirements for the FIR compiler MegaCore function.

Table 2. FIR Compiler Performance

Device	Speed Grade	Parameters	Utilization		Performance (MHz)
			LEs	EABs	
FLEX 10KE	-1	17-tap fully parallel filter	879	0	82
		19-tap fully parallel filter	1,260	0	101
		79-tap serial filter	761	5	69

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com