Using Soft Multipliers with Stratix & Stratix GX Devices

November 2002, ver. 2.0 Application Note 246

Introduction

Traditionally, designers have been forced to make a tradeoff between the flexibility of digital signal processors and the performance of ASICs and application-specific standard product (ASSPs) digital signal processing (DSP) solutions. The Altera® DSP solution eliminates the need for this tradeoff by providing exceptional performance combined with the flexibility of Stratix™ and Stratix GX FPGAs.

Stratix and Stratix GX devices include embedded high-performance multiplier-accumulators (MACs) in dedicated DSP blocks. The DSP blocks can operate at data rates above 300 million samples per second (MSPS), making Stratix and Stratix GX FPGAs ideal for high-speed DSP applications. In addition to the dedicated DSP blocks, designers can use the TriMatrix™ memory blocks to implement variable depth/width, high-performance multipliers. In this instance, TriMatrix memory blocks are used as a look-up table (LUT), which contains all possible results from multiplication of input data to constant coefficients.

There are four different soft multiplier modes of operation:

- Parallel Multiplications - Multiple memories produce one multiplication result with each clock cycle (e.g., high speed data scaling)
- Semi-Parallel Multiplications - Each memory produces one multiplication with multi-cycle operation (e.g., coefficient update of least mean squares (LMSs), coefficient update of equalizer)
- Sum of Multiplications - One memory or group of memories produce the sum of multiplications (e.g., finite impulse response (FIR), discrete cosine transform (DCT))
- Hybrid Multiplications - Combination and optimization of semi-parallel and sum of multiplications modes of operation. Ideal for a complex number of multiplications (e.g., complex fast Fourier transform (FFT), infinite impulse response (IIR))

This application note covers the sum of multipliers mode of operation.

Stratix & Stratix GX Memory & DSP Blocks

The TriMatrix memories consist of three types of RAM blocks: M512, M4K, and M-RAM blocks. M512 and M4K RAM blocks are memory blocks with a maximum width of 18 and 36 bits, respectively, and a maximum performance of approximately 300 MHz (ideal for soft multipliers). You can use these memory blocks for DSP applications that are multiplier intensive, such as imaging and mobile wireless technologies where the data word size does not fit within the standard 8-, 16-, or 32-bit widths.

Tables 1 and 2 show the number of Stratix and Stratix GX TriMatrix M4K memory blocks.

Table 1. Stratix TriMatrix Memory Blocks

Feature	EP1S10	EP1S20	EP1S25	EP1S30	EP1S40	EP1S60	EP1S80	EP1S120
M512 RAM (32 × 18 bits)	94	194	224	295	384	574	767	1,118
M4K RAM (128 × 36 bits)	60	82	138	171	183	292	364	520
M-RAM (4K × 144 bits)	1	2	2	4	4	6	9	12
Total RAM bits	920,448	1,669,248	1,944,576	3,317,184	3,423,744	5,215,104	7,427,520	10,118,016

Table 2. Stratix GX M4K Memory Blocks

Feature	EP1SGX10C EP1SGX10D	EP1SGX25C EP1SGX25D EP1SGX25F	EP1SGX40D EP1SGX40G
M512 RAM (32 × 18 bits)	94	224	384
M4K RAM (128 × 36 bits)	60	138	183
M-RAM (4K × 144 bits)	1	2	4
Total RAM bits	920,448	1,944,576	3,423,744

Basics of DSP Operation

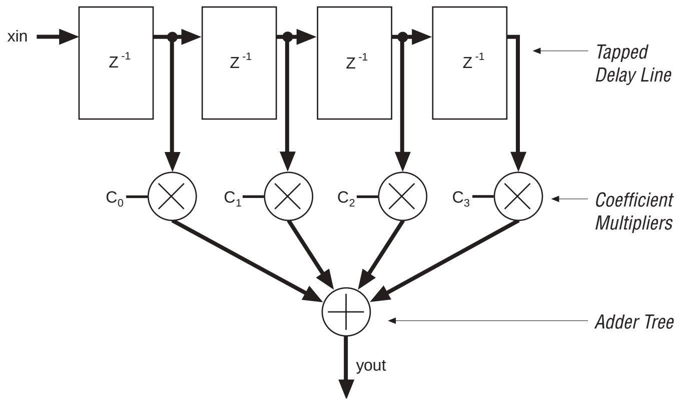
DSP is an arithmetic-intensive technology. To achieve high-speed signal processing, arithmetic operation must be accelerated.

Many digital systems use signal filtering to remove unwanted noise to provide spectral shaping, or to perform signal detection or analysis. Filters are used with communication applications such as band selection, low-pass filtering, and video convolution functions. Two types of filters that provide these functions are FIR and IIR filters. You can use FIR filters in systems that require a linear phase and have an inherently stable structure. You can use IIR filters in systems that can tolerate phase distortion. Typical filter applications include signal pre-conditioning, band selection, and low-pass filtering.

FIR Filter Architecture

The structure of a FIR filter is a weighted, tapped delay line. The filter design process involves identifying coefficients that match the frequency response specified for the system. The coefficients determine the response of the filter. By changing the coefficient values or by adding more coefficients to the filter, you can change the signal frequencies, which pass through the filter. Figure 1 shows the basic FIR filter structure.

Figure 1. Basic FIR Filter Structure

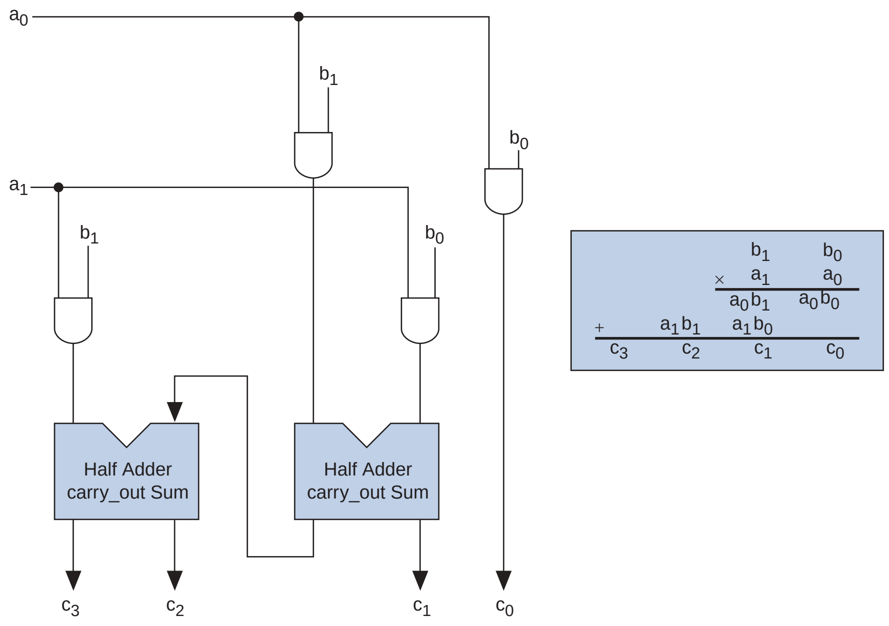


The FIR filter function (or many other DSP functions) is based on a MAC operation. The input data shifts into the shift register. The output of each register, which is called a tap, is the parallel input to the multiplier. Each parallel input is multiplied by a coefficient (C_n) as it is presented (see Figure 1).

MAC Function

The base of many DSP algorithms is a MAC function. The MAC function is represented by multiplication of a multiplier to a multiplicand, which implies that each element of the multiplier is multiplied by each bit of the multiplicand. The partial product of each multiplication is accumulated according to the weight of the partial product (weight indicates the location of a bit corresponding to other bits). For example, if a partial product of bits 4 through 7 is added to a partial product of bits 0 through 3, the partial product of 4 through 7 is shifted according to their weight and then accumulated to the partial product of previous stages. Figure 2 shows a simple 2×2 multiplication of multiplier a_1a_0 to multiplicand b_1b_0.

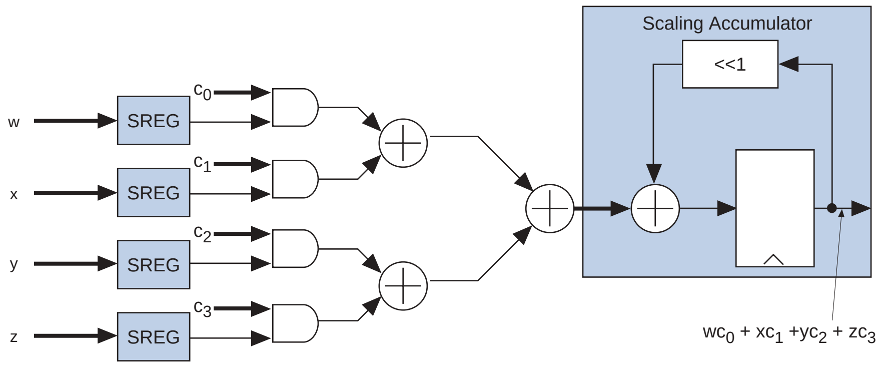
Figure 2. Multiplication of Two 2-Bit Numbers



Distributed Arithmetic

Distributed arithmetic is a method of performing multiplication by distributing the operation over many LUTs. Figure 3 shows the four-product MAC function that uses sequential shift and add to multiply four pairs, and then sum their partial product to obtain a final result. Each multiplier forms partial products by multiplying the coefficient by one bit of the data at a time using an AND gate.

Figure 3. Distributed Arithmetic with Four Constant Multiplicands



At the end of the process, these partial products (of a particular bit) are summed up and the final stage performs the final shift-accumulate at the scaling accumulator.

The scaling accumulator shifts the sums of partial products accordingly and then sums the result. The distributed-arithmetic circuit simultaneously performs four multiplications and sums the results when all of products are completed.

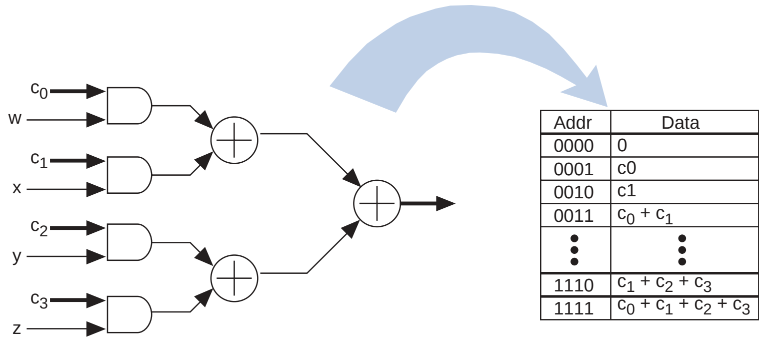
DSP Applications with Stratix & Stratix GX Devices

You can use Stratix and Stratix GX RAM blocks to implement DSP applications. Specifically, you can use the M512 and M4K RAM blocks, with 32 × 18-bit and 128 × 36-bit capacity, as LUTs to store the multiplication result of multiplier to multiplicand.

Distributed Arithmetic in LUT

Figure 4 shows how to implement distributed arithmetic using LUTs. The combined product and adder tree can be reduced to an LUT. The LUT contains the sums of constant coefficients for all possible input combinations to the LUT. Finally, the sums of the bits from the LUTs are added together, keeping in mind that different coefficient multiplications have different weight. Therefore, some shifting is required before the bit sums are performed.

Figure 4. Four-Bit Multiplication to Constant Coefficient Note (1)



Addr	Data
0000	0
0001	c0
0010	c1
0011	$c_0 + c_1$
⋮	⋮
1110	$c_1 + c_2 + c_3$
1111	$c_0 + c_1 + c_2 + c_3$

Note to Figure 4:

(1) c_0 to c_3 are constant coefficients.

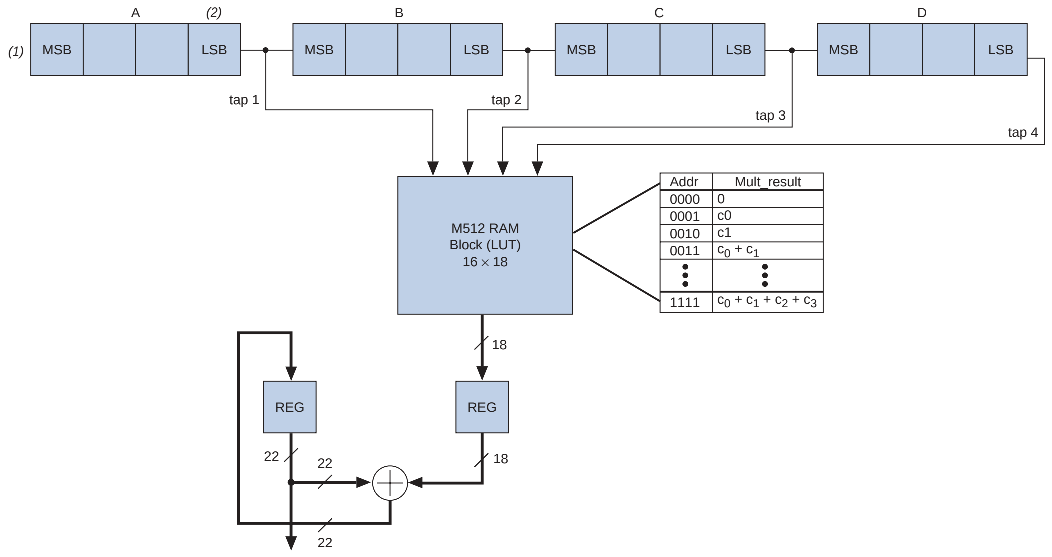
LUT Implementation in M512 & M4K RAM Blocks

You can use the Stratix and Stratix GX M512 or M4K RAM memory blocks as a LUTs to implement multiplication for DSP applications. All possible combinations of a multiplicand summation are calculated and stored in the M512 or M4K RAM block as an LUT. As a result, each address in the memory blocks represent a unique multiplication result.

Each multiplier's *n*-data bits load into a shift register at the data rate of clock / *n*-data bits. The shift register's data is the input, which point to an address location in the M512 or M4K RAM block. The output of the RAM block indicates the multiplication result for a specific bit, at each clock cycle.

Figure 5 shows the RAM LUT implementation of four 4-bit data inputs and up to 16-bit constant coefficients. This implementation takes four clock cycles to complete multiplication operation by adding partial products. At each cycle, the addition of partial products will generate one extra bit as the carry on bit. By the end of fourth cycle, the final addition of partial products will generate a 22-bit output.

Figure 5. Four-Tap FIR Filter Implementation Using M512 RAM Blocks as LUTs



Addr	Mult_result
0000	0
0001	c0
0010	c1
0011	$c_0 + c_1$
⋮	⋮
1111	$c_0 + c_1 + c_2 + c_3$

Notes to Figure 5:

(1) MSB: most significant bit.

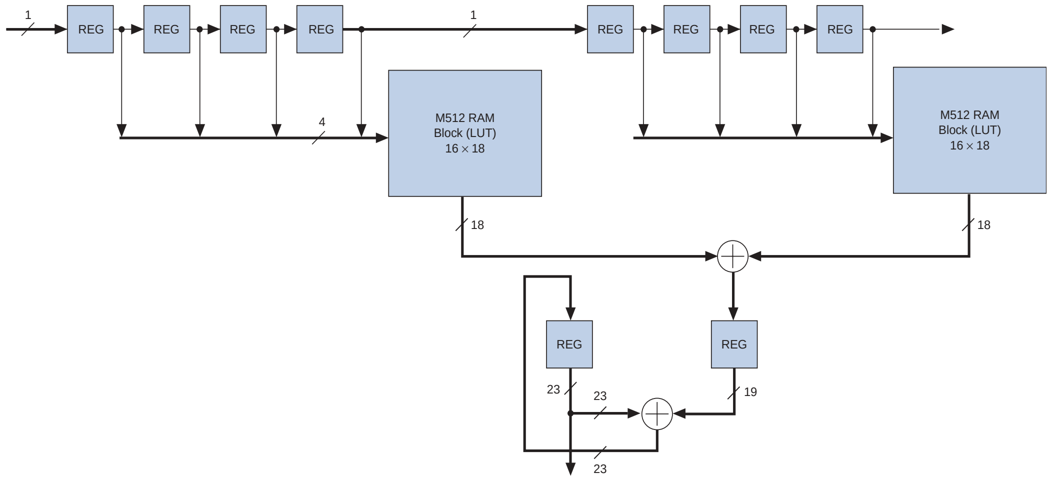
(2) LSB: least significant bit.

For example, Figure 5 shows four 4-bit data inputs called A, B, C, and D. The input data width determines the length of the shift register (e.g., a 4-bit shift register for 4-bit data). The outputs from the shift registers are called taps (the number of taps is equal to the number of input data). Since the size for Stratix and Stratix GX M512 RAM blocks is 32 × 18 bits, the maximum taps for each M512 RAM block is five (2^5 = 32 addresses). Depending on the number of inputs, coefficients, taps, and the required frequency, the number of RAM blocks that are used will vary. The example in Figure 5 requires only one M512 RAM block.

On each clock cycle, the LSB of each shift register simultaneously shifts out as inputs to the RAM block. The longer the shift register cycle, the more clock cycles required to shift out all the data to the RAM block. On the first cycle, the LSB of data cells A, B, C, and D are multiplied by all four bits of the coefficients. It will take four clock cycles to multiply the 4-bit inputs to the 16-bit coefficients.

If the FIR filter has more than five taps, you must use multiple M512 blocks or M4K RAM blocks (which are larger memory blocks) to complete multiplication. The reason for this requirement is that M512 RAM blocks have a maximum of 32 words (18 bits) per RAM block which is equal to 2^5 addresses, while M4K RAM blocks have a maximum of 128 words (36 bits) per RAM block which is equal to 2^7 addresses. The outputs of RAM blocks is shift-accumulated according to their weight, and provides the final multiplication result. Figure 6 shows the multiplication of eight 4-bit data inputs to a 16-bit constant coefficient in two M512 RAM blocks. The output from the RAM block is an 18-bit output. Since it takes four clock cycles to complete the multiplication operation (by adding the partial products), the final output after the forth cycle will be a 23-bit output.

Figure 6. Using Multiple M512 RAM Blocks for an 8-Tap FIR Filter

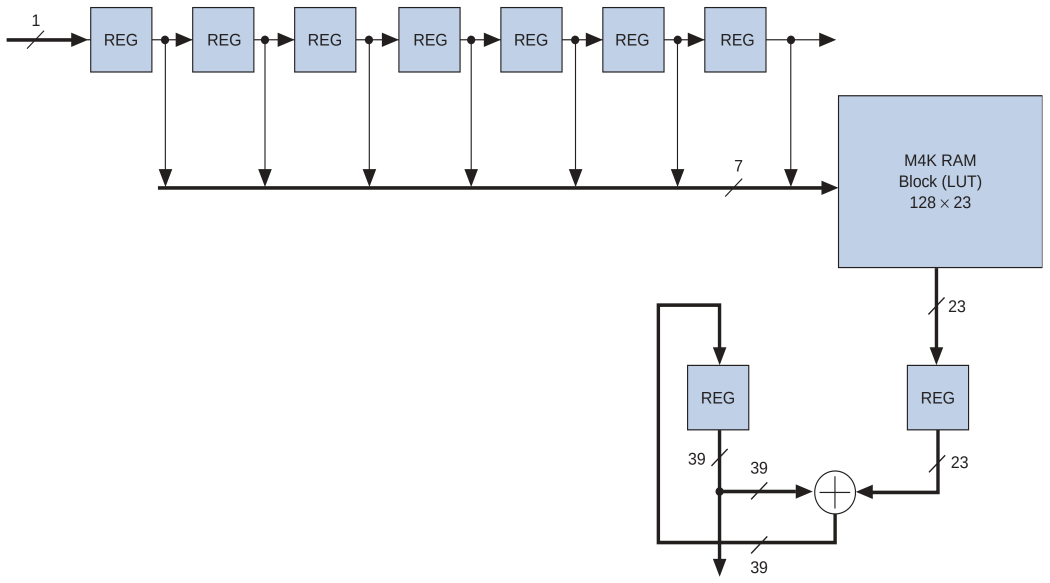


Since the application's performance is determined by the length of the shift register (MSPS = system clock / N, where N equals the width of input data bus) for filters that require high performance, Altera recommends that you split the shift register into smaller shift registers. This technique increases the performance, but uses more RAM blocks. See Table 1 on page 2 for the total number of TriMatrix memory blocks.

Also, if the FIR filter requires a coefficient larger than 16 bits, you can use multiple M512 RAM blocks or use M4K RAM blocks (M4K RAM blocks can perform multiplication up to a 34-bit coefficient). RAM block outputs are shifted and accumulated in a scaling accumulator to add up the partial products together and to obtain a final multiplication result.

Figure 7 shows multiplication of seven 16-bit data inputs to a 20-bit constant coefficient in one M4K RAM block. The 128 addressed lines correspond to seven data inputs in a M4K RAM block. Performing seven 16-bit × 20-bit multiplications will generate a 23-bit output from a M4K RAM block. Since it takes 16 clock cycles to complete adding the partial products, and at each partial product addition one bit is added to the total number of output bits, the final output is 39 bits.

Figure 7. Using a M4K RAM Block for a 7-Tap FIR Filter



Even though this technique is for a multiplication of constant coefficients, to change the coefficients, you can rewrite the RAM blocks since Stratix and Stratix GX memory blocks are dual-port RAM. This operation is useful for adaptive FIR filters.

DSP Performance (MMAC)

Because digital signal processors are typically assigned MAC-intensive tasks, the DSP performance is related to MAC throughput. The unit for DSP speed is million multiply-accumulate operations per second (MMACS). Tables 3 and 4 show the throughput for DSP applications implemented in Stratix and Stratix GX RAM blocks for 16 × 16 multipliers, respectively. Tables 3 and 4 also show the total number of 16 × 16 multipliers that exist within Stratix and Stratix GX M512 and M4K blocks, respectively.

Table 3. 16 × 16 Multiplier in Stratix RAM Blocks *Notes (1), (2)*

Device	Number of Multipliers *(3)*			Performance (MMACS)		
	M512 Blocks	M4K Blocks	Total Blocks *(4)*	M512 Blocks	M4K Blocks	Total Blocks *(4)*
EP1S10	23	30	53	6,900	9,000	15,900
EP1S20	48	41	89	14,000	12,300	26,700
EP1S25	56	69	125	16,800	20,700	37,500
EP1S30	74	85	159	22,200	25,500	47,700
EP1S40	96	91	187	22,800	27,300	56,100
EP1S60	144	146	290	43,200	43,800	87,000
EP1S80	192	182	370	57,600	54,600	112,200
EP1S120	280	260	540	84,000	78,000	162,000

Table 4. 16 × 16 Multiplier in Stratix GX RAM Blocks *Notes (1), (2)*

Device	Number of Multipliers *(3)*			Performance (MMACS)		
	M512 Blocks	M4K Blocks	Total Blocks *(4)*	M512 Blocks	M4K Blocks	Total Blocks *(4)*
EP1S10C	23	30	53	6,900	9,000	15,900
EP1S10D	23	30	53	6,900	9,000	15,900
EP1S25C	56	69	125	16,800	20,700	37,500
EP1S25D	56	69	125	16,800	20,700	37,500
EP1S25F	56	69	125	16,800	20,700	37,500
EP1S40D	96	91	187	22,800	27,300	56,100
EP1S40G	96	91	187	22,800	27,300	56,100

Notes to Table 3:

(1) Both coefficient and input data width are 16 bits.

(2) These values are for four inputs. The number of multipliers will increase by approximately 50% with optimization of five inputs for M512 blocks and seven inputs for M4K blocks.

(3) The number of multipliers are normalized (divided by 16) to create throughput of one result per clock cycle.

(4) The total number of M512 and M4K blocks in a device.

To calculate the MAC throughput, use the following equation:

MMACS = (Frequency (MHz) × number of multipliers in RAM block)

For these calculations, assume that the operation frequency in M512 or M4K RAM blocks is approximately 300 MHz. It is also assumed that it will take 16 clock cycles to perform a multiplication operation (due to the width of the input data).

Software Implementation

The Altera FIR Compiler MegaCore® function generates FIR filters customized for Altera devices. You can use the FIR Compiler wizard interface to implement a variety of filter architectures, including fully parallel, serial, and multi-bit serial-fixed coefficient filters, and MAC-based and multi-cycle variable filters. The wizard also includes a coefficient generator to help you create filter coefficients.

FIR Compiler version 2.6.0 supports FIR filter implementation in Stratix and Stratix GX TriMatrix memory blocks. You can use this version of the FIR Compiler to implement soft multiplier-based FIR filters.

The FIR Compiler function speeds up the design cycle by:

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

For more information on the FIR Compiler function, refer to *FIR Compiler MegaCore Function User Guide*. For more information on FIR Filter and programmable logic device (PLD) implementation, refer to *AN 73: Implementing FIR Filters in FLEX Devices*.

Conclusion

You can use Stratix and Stratix GX DSP blocks to implement DSP applications. An alternative to DSP block implementation is the use of Stratix and Stratix GX TriMatrix blocks (M512 or M4K RAM blocks). This alternative is useful for designs that need more multipliers than are available using DSP blocks. This implementations is particularly useful when these multipliers are multiplied by a constant or a value that infrequently changes, as in an adaptive FIR filter.

Revision History

The information contained in *AN 246: Using Soft Multipliers with Stratix & Stratix GX Devices* version 2.0 supersedes information published in previous versions. The following change was made in *AN 246: Using Soft Multipliers with Stratix & Stratix GX Devices* version 2.0: added Stratix GX devices throughout the document.

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Literature Services:
lit_req@altera.com