

Cyclone® IV devices include a combination of on-chip resources and external interfaces that help increase performance, reduce system cost, and lower the power consumption of digital signal processing (DSP) systems. Cyclone IV devices, either alone or as DSP device co-processors, are used to improve price-to-performance ratios of DSP systems. Particular focus is placed on optimizing Cyclone IV devices for applications that benefit from an abundance of parallel processing resources, which include video and image processing, intermediate frequency (IF) modems used in wireless communications systems, and multi-channel communications and video systems.

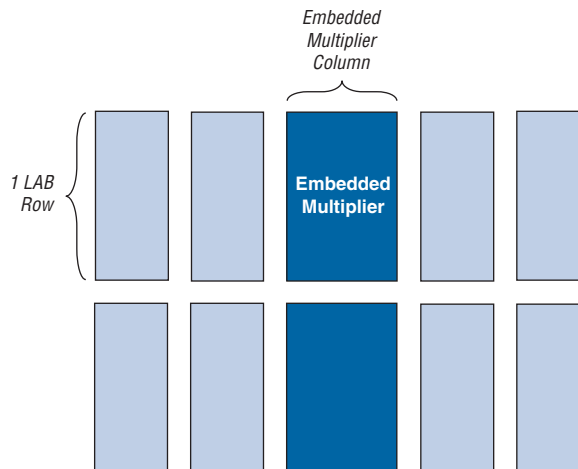
This chapter contains the following sections:

- “Embedded Multiplier Block Overview” on page 4-1
- “Architecture” on page 4-2
- “Operational Modes” on page 4-4

## Embedded Multiplier Block Overview

Figure 4-1 shows one of the embedded multiplier columns with the surrounding logic array blocks (LABs). The embedded multiplier is configured as either one  $18 \times 18$  multiplier or two  $9 \times 9$  multipliers. For multiplications greater than  $18 \times 18$ , the Quartus® II software cascades multiple embedded multiplier blocks together. There are no restrictions on the data width of the multiplier, but the greater the data width, the slower the multiplication process.

**Figure 4-1. Embedded Multipliers Arranged in Columns with Adjacent LABs**



© 2010 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at [www.altera.com/common/legal.html](http://www.altera.com/common/legal.html). Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.



Table 4–1 lists the number of embedded multipliers and the multiplier modes that can be implemented in each Cyclone IV device.

**Table 4–1. Number of Embedded Multipliers in Cyclone IV Devices**

Device Family	Device	Embedded Multipliers	9 × 9 Multipliers <sup>(1)</sup>	18 × 18 Multipliers <sup>(1)</sup>
Cyclone IV GX	EP4CGX15	0	0	0
	EP4CGX22	40	80	40
	EP4CGX30	80	160	80
	EP4CGX50	140	280	140
	EP4CGX75	198	396	198
	EP4CGX110	280	560	280
	EP4CGX150	360	720	360
Cyclone IV E	EP4CE6	15	30	15
	EP4CE10	23	46	23
	EP4CE15	56	112	56
	EP4CE22	66	132	66
	EP4CE30	66	132	66
	EP4CE40	116	232	116
	EP4CE55	154	308	154
	EP4CE75	200	400	200
	EP4CE115	266	532	266

**Note to Table 4–1:**

(1) These columns show the number of 9 × 9 or 18 × 18 multipliers for each device.

In addition to the embedded multipliers in Cyclone IV devices, you can implement soft multipliers by using the M9K memory blocks as look-up tables (LUTs). The LUTs contain partial results from the multiplication of input data with coefficients that implement variable depth and width high-performance soft multipliers for low-cost, high-volume DSP applications. The availability of soft multipliers increases the number of available multipliers in the device.



For more information about M9K memory blocks, refer to the *Memory Blocks in Cyclone IV Devices* chapter.



For more information about soft multipliers, refer to *AN 306: Implementing Multipliers in FPGA Devices*.

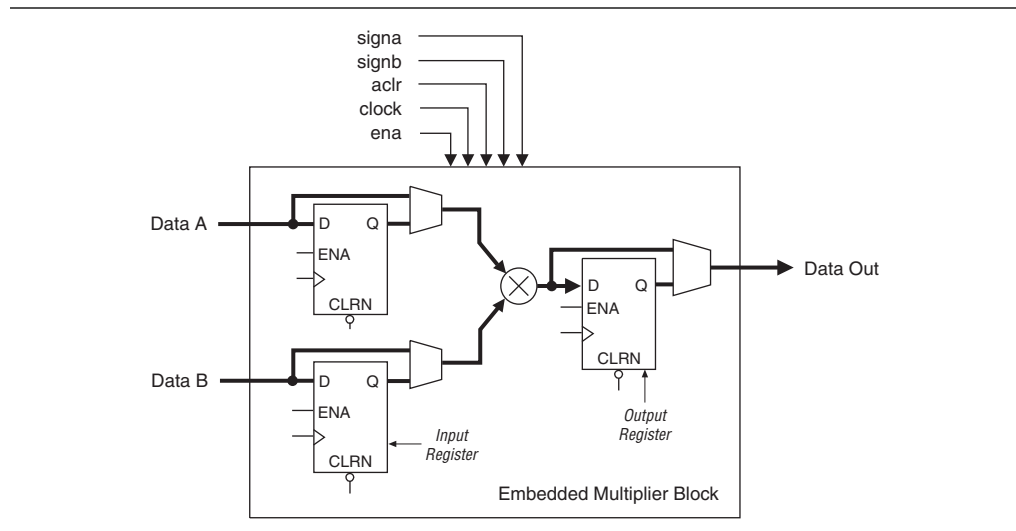
## Architecture

Each embedded multiplier consists of the following elements:

- Multiplier stage
- Input and output registers
- Input and output interfaces

Figure 4-2 shows the multiplier block architecture.

**Figure 4-2. Multiplier Block Architecture**



## Input Registers

You can send each multiplier input signal into an input register or directly into the multiplier in 9- or 18-bit sections, depending on the operational mode of the multiplier. You can send each multiplier input signal through a register independently of other input signals. For example, you can send the multiplier Data A signal through a register and send the Data B signal directly to the multiplier.

The following control signals are available for each input register in the embedded multiplier:

- clock
- clock enable
- asynchronous clear

All input and output registers in a single embedded multiplier are fed by the same clock, clock enable, and asynchronous clear signals.

## Multiplier Stage

The multiplier stage of an embedded multiplier block supports  $9 \times 9$  or  $18 \times 18$  multipliers, as well as other multipliers between these configurations. Depending on the data width or operational mode of the multiplier, a single embedded multiplier can perform one or two multiplications in parallel. For multiplier information, refer to “Operational Modes” on page 4-4.


Each multiplier operand is a unique signed or unsigned number. The *signa* and *signb* signals control an input of a multiplier and determine if the value is signed or unsigned. If the *signa* signal is high, the Data A operand is a signed number. If the *signa* signal is low, the Data A operand is an unsigned number.

Table 4–2 lists the sign of the multiplication results for the various operand sign representations. The results of the multiplication are signed if any one of the operands is a signed value.

**Table 4–2. Multiplier Sign Representation**

Data A		Data B		Result
signa Value	Logic Level	signb Value	Logic Level	
Unsigned	Low	Unsigned	Low	Unsigned
Unsigned	Low	Signed	High	Signed
Signed	High	Unsigned	Low	Signed
Signed	High	Signed	High	Signed

Each embedded multiplier block has only one *signa* and one *signb* signal to control the sign representation of the input data to the block. If the embedded multiplier block has two  $9 \times 9$  multipliers, the *Data A* input of both multipliers share the same *signa* signal, and the *Data B* input of both multipliers share the same *signb* signal. You can dynamically change the *signa* and *signb* signals to modify the sign representation of the input operands at run time. You can send the *signa* and *signb* signals through a dedicated input register. The multiplier offers full precision, regardless of the sign representation.

 When the *signa* and *signb* signals are unused, the Quartus II software sets the multiplier to perform unsigned multiplication by default.

## Output Registers

You can register the embedded multiplier output with output registers in either 18- or 36-bit sections, depending on the operational mode of the multiplier. The following control signals are available for each output register in the embedded multiplier:


- clock
- clock enable
- asynchronous clear

All input and output registers in a single embedded multiplier are fed by the same clock, clock enable, and asynchronous clear signals.

## Operational Modes

You can use an embedded multiplier block in one of two operational modes, depending on the application needs:

- One  $18 \times 18$  multiplier
- Up to two  $9 \times 9$  independent multipliers

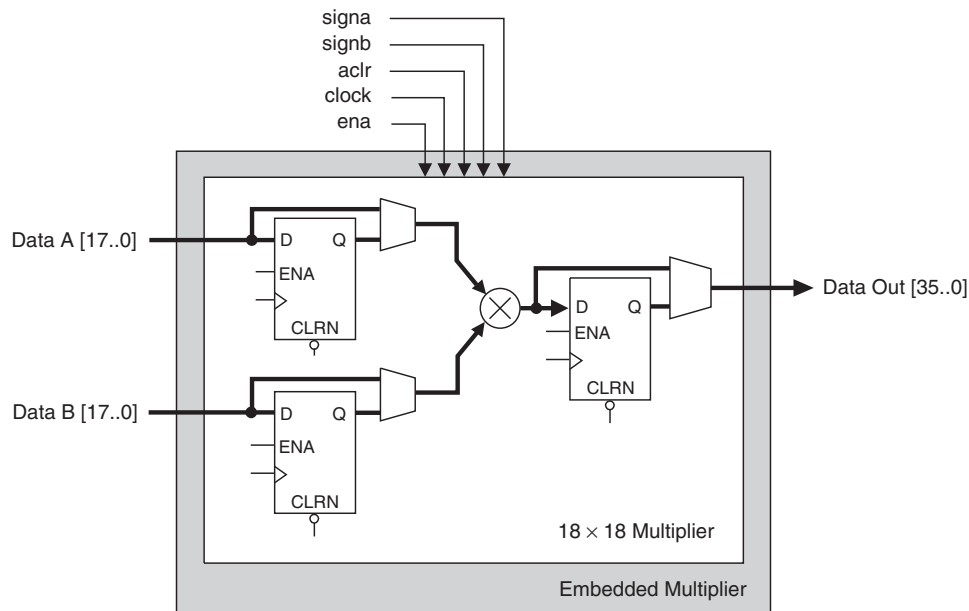
 You can also use embedded multipliers of Cyclone IV devices to implement multiplier adder and multiplier accumulator functions, in which the multiplier portion of the function is implemented with embedded multipliers, and the adder or accumulator function is implemented in logic elements (LEs).

## 18-Bit Multipliers

You can configure each embedded multiplier to support a single  $18 \times 18$  multiplier for input widths of 10 to 18 bits.

Figure 4-3 shows the embedded multiplier configured to support an 18-bit multiplier.

Figure 4-3. 18-Bit Multiplier Mode



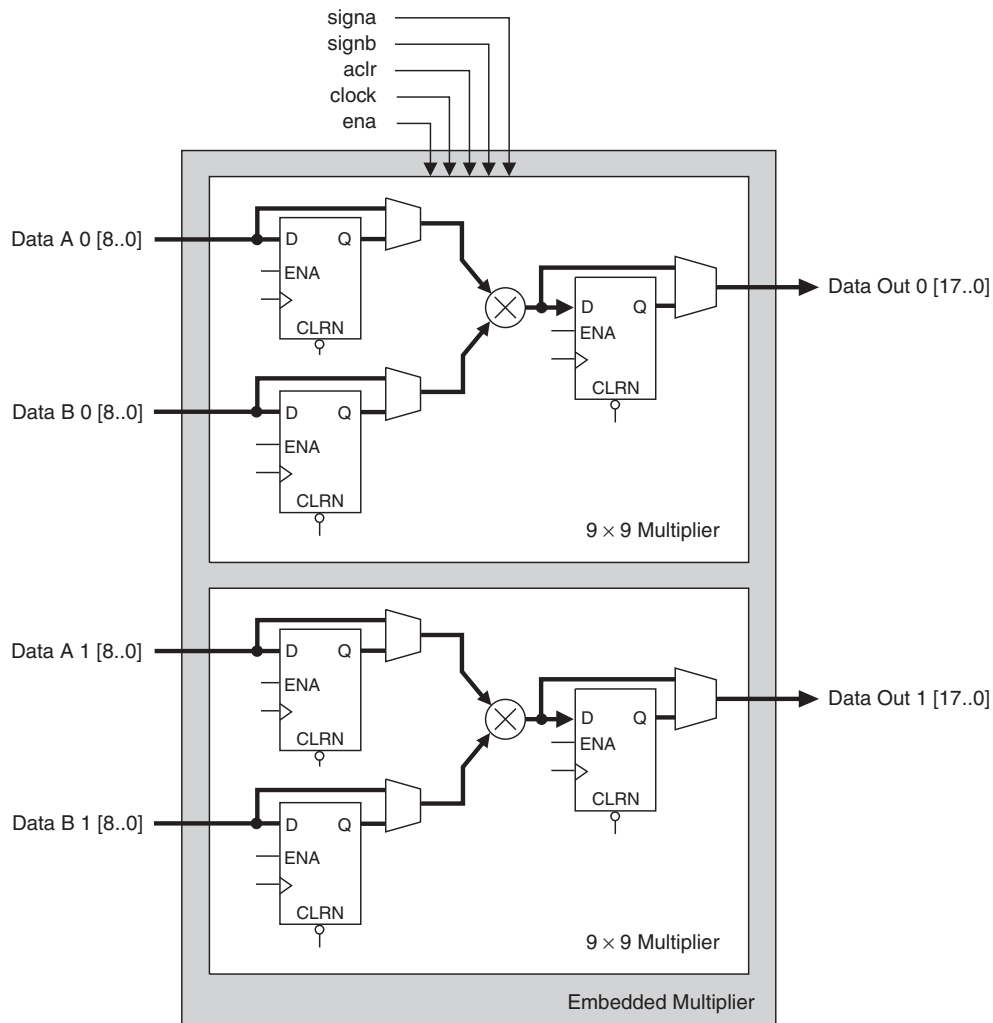
All 18-bit multiplier inputs and results are independently sent through registers. The multiplier inputs can accept signed integers, unsigned integers, or a combination of both. Also, you can dynamically change the *signa* and *signb* signals and send these signals through dedicated input registers.

## 9-Bit Multipliers

You can configure each embedded multiplier to support two  $9 \times 9$  independent multipliers for input widths of up to 9 bits.

Figure 4-4 shows the embedded multiplier configured to support two 9-bit multipliers.

**Figure 4-4. 9-Bit Multiplier Mode**



All 9-bit multiplier inputs and results are independently sent through registers. The multiplier inputs can accept signed integers, unsigned integers, or a combination of both. Two  $9 \times 9$  multipliers in the same embedded multiplier block share the same *signa* and *signb* signal. Therefore, all the Data A inputs feeding the same embedded multiplier must have the same sign representation. Similarly, all the Data B inputs feeding the same embedded multiplier must have the same sign representation.

## Document Revision History

Table 4-3 lists the revision history for this chapter.

**Table 4-3. Document Revision History**

Date	Version	Changes
February 2010	1.1	Added Cyclone IV E devices in Table 4-1 for the Quartus II software version 9.1 SP1 release.
November 2009	1.0	Initial release.

