Logic Array Blocks and Adaptive Logic Modules in Cyclone V Devices 1

2014.01.10

CV-52001 Subscribe Send Feedback

This chapter describes the features of the logic array block (LAB) in the Cyclone® V core fabric.

The LAB is composed of basic building blocks known as adaptive logic modules (ALMs) that you can configure to implement logic functions, arithmetic functions, and register functions.

You can use a quarter of the available LABs in the Cyclone V devices as a memory LAB (MLAB).

The Quartus® II software and other supported third-party synthesis tools, in conjunction with parameterized functions such as the library of parameterized modules (LPM), automatically choose the appropriate mode for common functions such as counters, adders, subtractors, and arithmetic functions.

This chapter contains the following sections:

- LAB
- ALM Operating Modes

Related Information

Cyclone V Device Handbook: Known Issues
Lists the planned updates to the *Cyclone V Device Handbook* chapters.

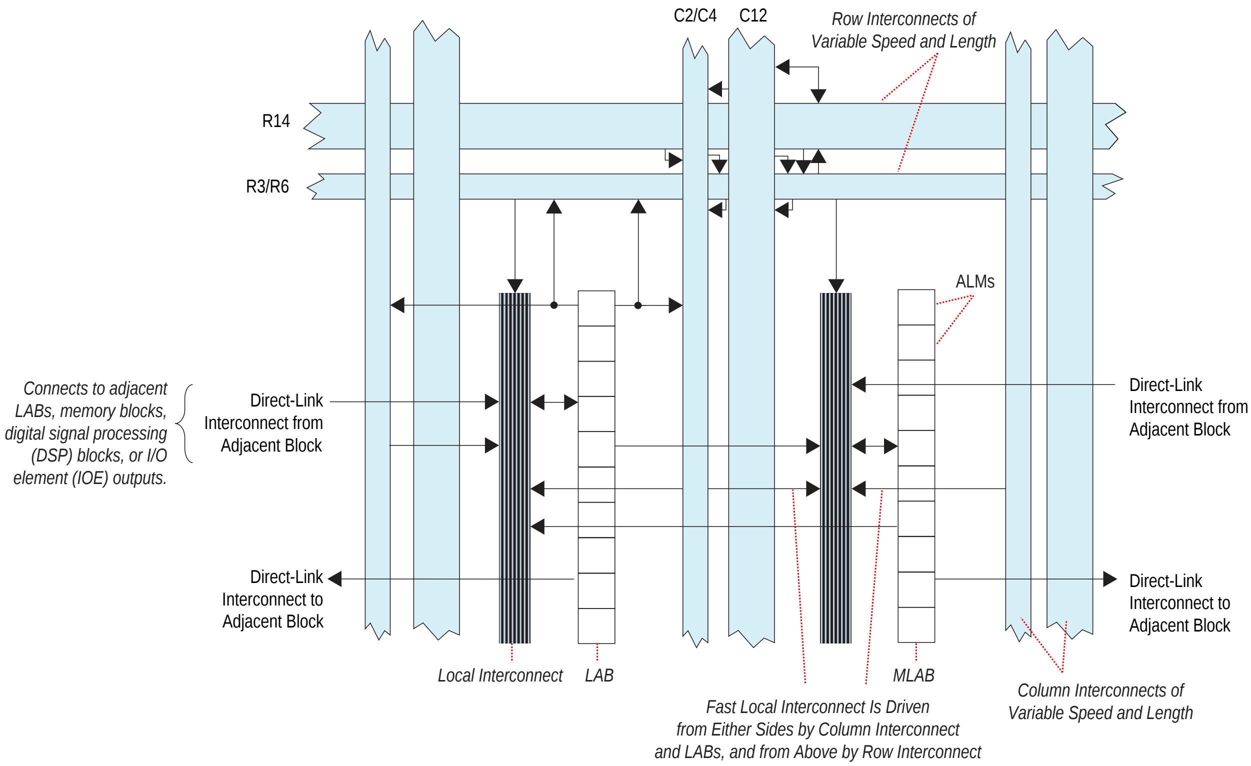
LAB

The LABs are configurable logic blocks that consist of a group of logic resources. Each LAB contains dedicated logic for driving control signals to its ALMs.

MLAB is a superset of the LAB and includes all the LAB features.

Figure 1-1: LAB Structure and Interconnects Overview in Cyclone V Devices

This figure shows an overview of the Cyclone V LAB and MLAB structure with the LAB interconnects.

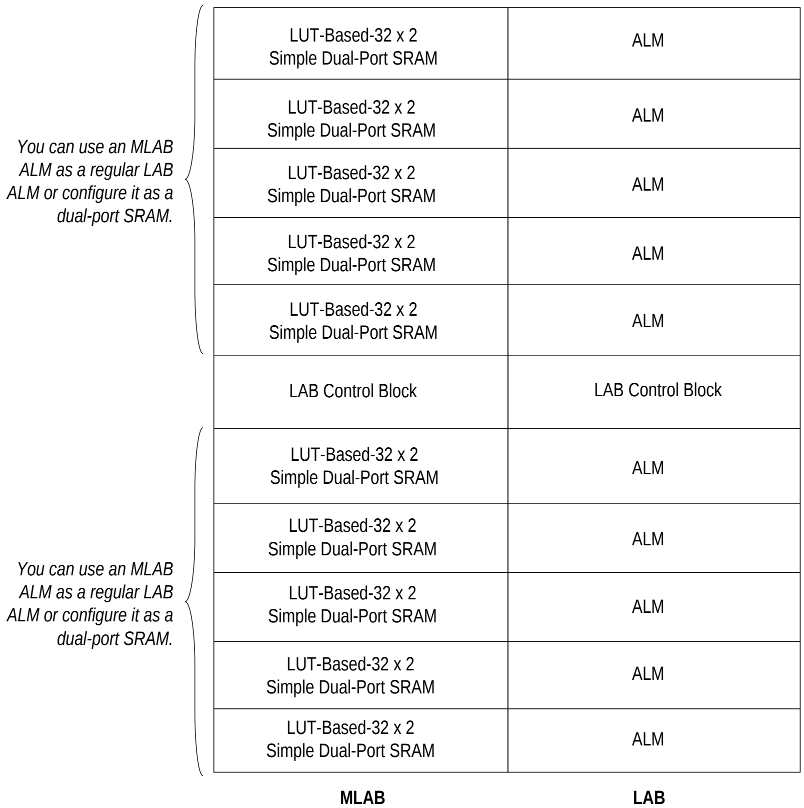


MLAB

Each MLAB supports a maximum of 640 bits of simple dual-port SRAM.

You can configure each ALM in an MLAB as a 32 x 2 memory block, resulting in a configuration of 32 x 20 simple dual-port SRAM block.

Figure 1-2: LAB and MLAB Structure for Cycone V Devices



Local and Direct Link Interconnects

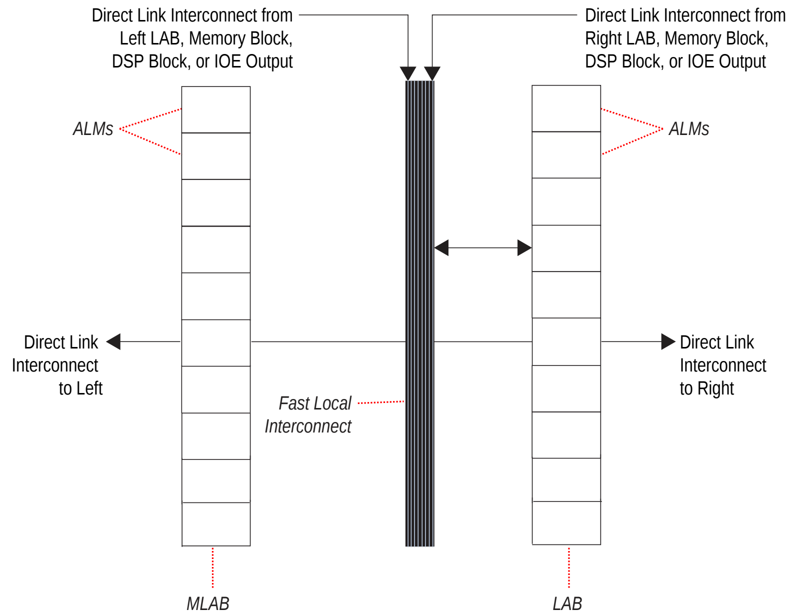
Each LAB can drive 30 ALMs through fast-local and direct-link interconnects. Ten ALMs are in any given LAB and ten ALMs are in each of the adjacent LABs.

The local interconnect can drive ALMs in the same LAB using column and row interconnects and ALM outputs in the same LAB.

Neighboring LABs, MLABs, M10K blocks, or digital signal processing (DSP) blocks from the left or right can also drive the LAB's local interconnect using the direct link connection.

The direct link connection feature minimizes the use of row and column interconnects, providing higher performance and flexibility.

Figure 1-3: LAB Fast Local and Direct Link Interconnects for Cyclone V Devices



LAB Control Signals

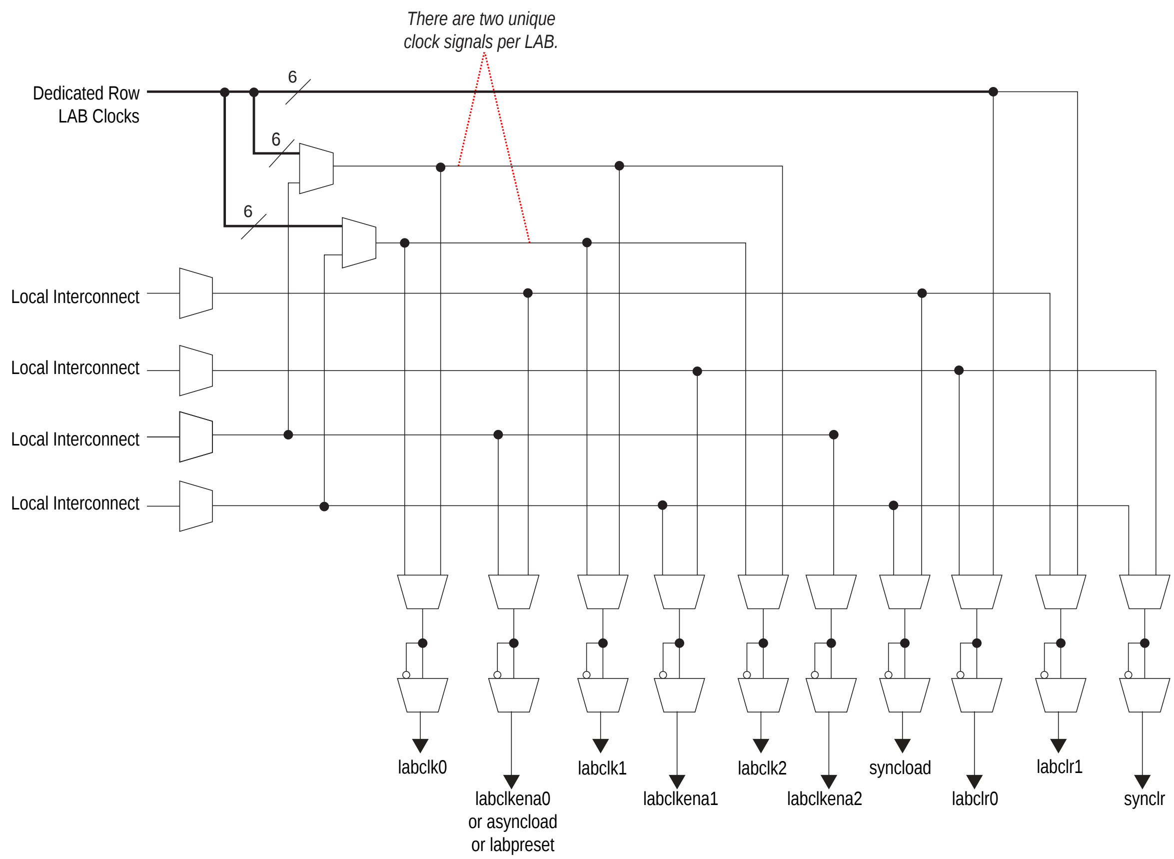
Each LAB contains dedicated logic for driving the control signals to its ALMs, and has two unique clock sources and three clock enable signals.

The LAB control block generates up to three clocks using the two clock sources and three clock enable signals. Each clock and the clock enable signals are linked.

De-asserting the clock enable signal turns off the corresponding LAB-wide clock.

Figure 1-4: LAB-Wide Control Signals for Cyclone V Devices

This figure shows the clock sources and clock enable signals in a LAB.



ALM Resources

One ALM contains four programmable registers. Each register has the following ports:

- Data
- Clock
- Synchronous and asynchronous clear
- Synchronous load

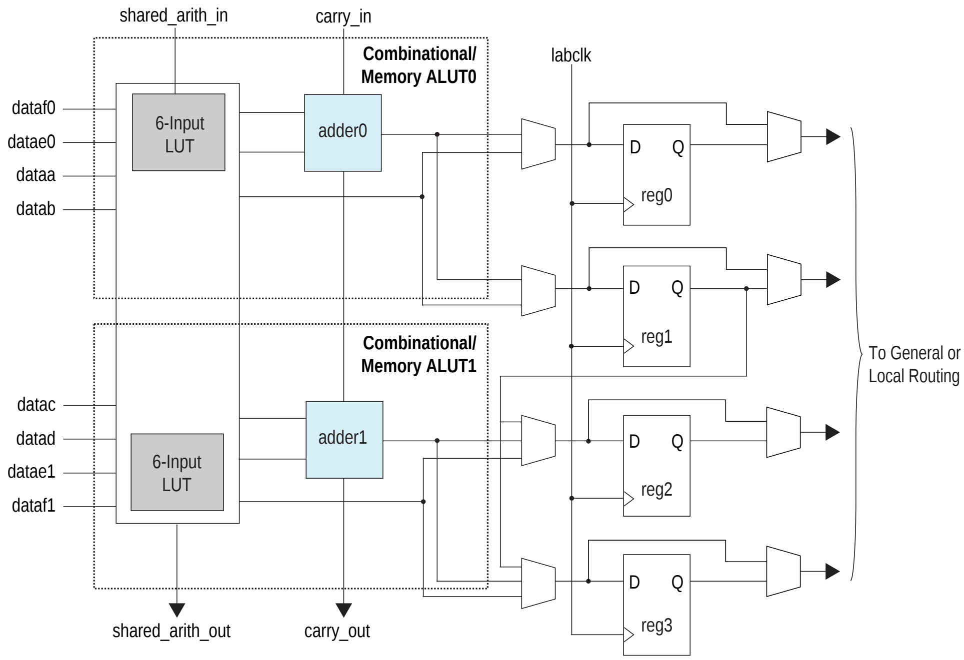
Global signals, general-purpose I/O (GPIO) pins, or any internal logic can drive the clock and clear control signals of an ALM register.

GPIO pins or internal logic drives the clock enable signal.

For combinational functions, the registers are bypassed and the output of the look-up table (LUT) drives directly to the outputs of an ALM.

Note: The Quartus II software automatically configures the ALMs for optimized performance.

Figure 1-5: ALM High-Level Block Diagram for Cyclone V Devices



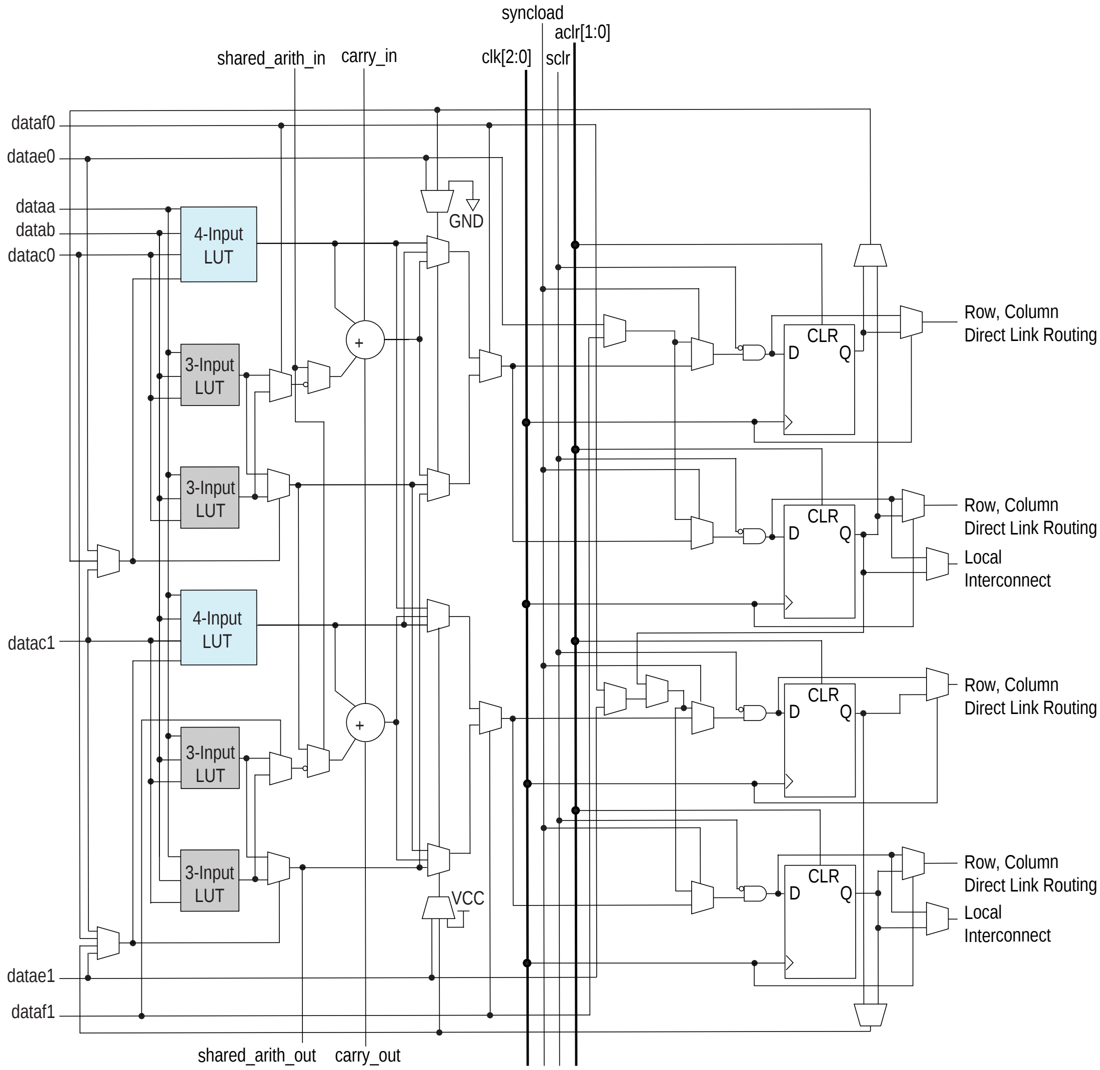
ALM Output

The general routing outputs in each ALM drive the local, row, and column routing resources. Two ALM outputs can drive column, row, or direct link routing connections, and one of these ALM outputs can also drive local interconnect resources.

The LUT, adder, or register output can drive the ALM outputs. The LUT or adder can drive one output while the register drives another output.

Register packing improves device utilization by allowing unrelated register and combinational logic to be packed into a single ALM. Another mechanism to improve fitting is to allow the register output to feed back into the look-up table (LUT) of the same ALM so that the register is packed with its own fan-out LUT. The ALM can also drive out registered and unregistered versions of the LUT or adder output.

Figure 1-6: ALM Connection Details for Cyclone V Devices



ALM Operating Modes

The Cyclone V ALM operates in any of the following modes:

- Normal mode
- Extended LUT mode
- Arithmetic mode
- Shared arithmetic mode

Normal Mode

Normal mode allows two functions to be implemented in one Cyclone V ALM, or a single function of up to six inputs.

Up to eight data inputs from the LAB local interconnect are inputs to the combinational logic.

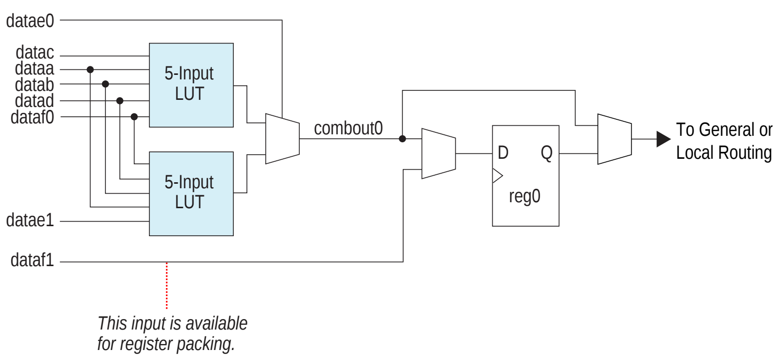
The ALM can support certain combinations of completely independent functions and various combinations of functions that have common inputs.

Extended LUT Mode

In this mode, if the 7-input function is unregistered, the unused eighth input is available for register packing.

Functions that fit into the template, as shown in the following figure, often appear in designs as “if-else” statements in Verilog HDL or VHDL code.

Figure 1-7: Template for Supported 7-Input Functions in Extended LUT Mode for Cyclone V Devices



Arithmetic Mode

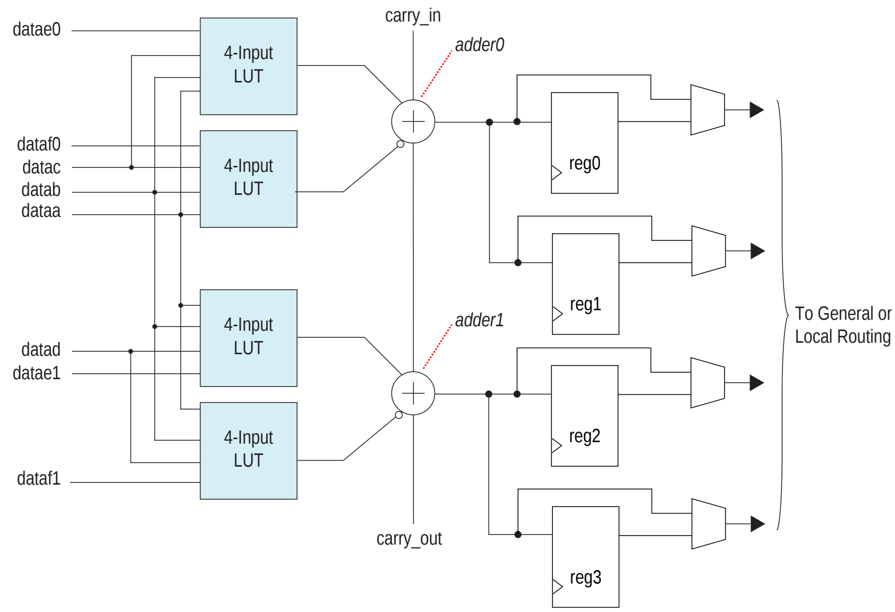
The ALM in arithmetic mode uses two sets of two 4-input LUTs along with two dedicated full adders.

The dedicated adders allow the LUTs to perform pre-adder logic; therefore, each adder can add the output of two 4-input functions.

The ALM supports simultaneous use of the adder’s carry output along with combinational logic outputs. The adder output is ignored in this operation.

Using the adder with the combinational logic output provides resource savings of up to 50% for functions that can use this mode.

Figure 1-8: ALM in Arithmetic Mode for Cycline V Devices



Carry Chain

The carry chain provides a fast carry function between the dedicated adders in arithmetic or shared arithmetic mode.

The two-bit carry select feature in Cyclone V devices halves the propagation delay of carry chains within the ALM. Carry chains can begin in either the first ALM or the fifth ALM in a LAB. The final carry-out signal is routed to an ALM, where it is fed to local, row, or column interconnects.

To avoid routing congestion in one small area of the device when a high fan-in arithmetic function is implemented, the LAB can support carry chains that only use either the top half or bottom half of the LAB before connecting to the next LAB. This leaves the other half of the ALMs in the LAB available for implementing narrower fan-in functions in normal mode. Carry chains that use the top five ALMs in the first LAB carry into the top half of the ALMs in the next LAB in the column. Carry chains that use the bottom five ALMs in the first LAB carry into the bottom half of the ALMs in the next LAB within the column. You can bypass the top-half of the LAB columns and bottom-half of the MLAB columns.

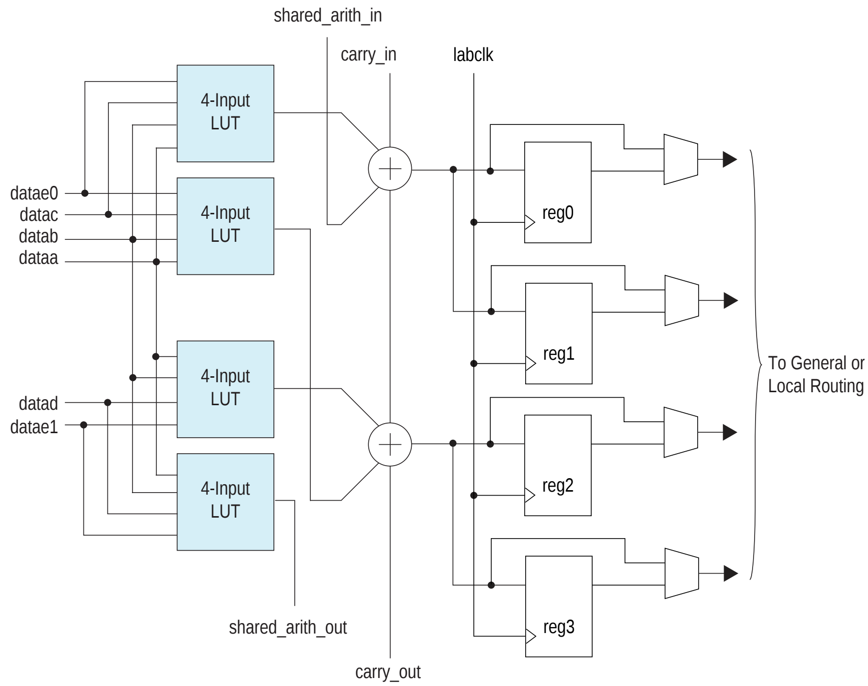
The Quartus II Compiler creates carry chains longer than 20 ALMs (10 ALMs in arithmetic or shared arithmetic mode) by linking LABs together automatically. For enhanced fitting, a long carry chain runs vertically, allowing fast horizontal connections to the TriMatrix memory and DSP blocks. A carry chain can continue as far as a full column.

Shared Arithmetic Mode

The ALM in shared arithmetic mode can implement a 3-input add in the ALM.

This mode configures the ALM with four 4-input LUTs. Each LUT either computes the sum of three inputs or the carry of three inputs. The output of the carry computation is fed to the next adder using a dedicated connection called the shared arithmetic chain.

Figure 1-9: ALM in Shared Arithmetic Mode for Cyclone V Devices



Shared Arithmetic Chain

The shared arithmetic chain available in enhanced arithmetic mode allows the ALM to implement a 3-input adder. This significantly reduces the resources necessary to implement large adder trees or correlator functions.

The shared arithmetic chain can begin in either the first or sixth ALM in a LAB.

Similar to carry chains, the top and bottom half of the shared arithmetic chains in alternate LAB columns can be bypassed. This capability allows the shared arithmetic chain to cascade through half of the ALMs in an LAB while leaving the other half available for narrower fan-in functionality. In every LAB, the column is top-half bypassable; while in MLAB, columns are bottom-half bypassable.

The Quartus II Compiler creates shared arithmetic chains longer than 20 ALMs (10 ALMs in arithmetic or shared arithmetic mode) by linking LABs together automatically. To enhance fitting, a long shared arithmetic chain runs vertically, allowing fast horizontal connections to the TriMatrix memory and DSP blocks. A shared arithmetic chain can continue as far as a full column.

Document Revision History

Date	Version	Changes
January 2014	2014.01.10	Added multiplexers for the bypass paths and register outputs in the following diagrams: • ALM High-Level Block Diagram for Cyclone V Devices • Template for Supported 7-Input Functions in Extended LUT Mode for Cyclone V Devices • ALM in Arithmetic Mode for Cyclone V Devices • ALM in Shared Arithmetic Mode for Cyclone V Devices
May 2013	2013.05.06	• Added link to the known document issues in the Knowledge Base. • Removed register chain outputs information in ALM output section. • Removed `reg_chain_in` and `reg_chain_out` ports in ALM high-level block diagram and ALM connection details diagram.
December 2012	2012.12.28	Reorganized content and updated template.
June 2012	2.0	Updated for the Quartus II software v12.0 release: • Restructured chapter. • Updated Figure 1–6.
November 2011	1.1	Minor text edits.
October 2011	1.0	Initial release.