Intel Agilex® 7 Logic Array Blocks and Adaptive Logic Modules User Guide

Contents

1. Intel Agilex® 7 LAB and ALM Overview

The logic array block (LAB) is composed of basic building blocks known as adaptive logic modules (ALMs). You can configure the LABs to implement logic functions, arithmetic functions, and register functions.

You can use half of the available LABs in the Intel Agilex® 7 devices as memory LABs (MLABs). Certain devices may have a higher MLAB ratio.

The Intel® Quartus® Prime software and other supported third-party synthesis tools automatically choose the appropriate mode for common functions such as counters, adders, subtractors, and arithmetic functions.

Related Information

Intel Hyperflex™ Core Architecture, Intel Agilex 7 FPGAs and SoCs Device Overview
Provides more information about Hyper-Registers and the Intel Hyperflex™ core architecture. Hyper-Registers are additional registers available in every interconnect routing segment throughout the core fabric, including the routing segments connected to the LAB inputs and outputs.

intel.

2. Intel Hyperflex™ Register

The Intel Agilex 7 device family is based on the Intel Hyperflex™ core architecture.

The Intel Agilex 7 LAB contains Intel Hyperflex registers and other features designed to facilitate retiming. Intel Hyperflex registers are available in ALMs and carry chains. As shown in the Intel Agilex 7 ALM Connection Details figure, the Intel Hyperflex registers are located on the synchronous clear and clock enable inputs to increase or reduce path delay. All the Intel Hyperflex registers can be enabled and are controlled by the Intel Quartus Prime software during retiming.

intel.

3. Intel Agilex 7 LAB and ALM Architecture and Features

3.1. 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. The MLAB is a superset of the LAB and includes all the LAB features. There are a total of 10 ALMs in each LAB, as shown in the *Intel Agilex 7 LAB and MLAB Structure* figure.

Figure 1.   Intel Agilex 7 LAB Structure and Interconnects Overview

This figure shows an overview of the Intel Agilex 7 LAB and MLAB structure with the LAB interconnects.
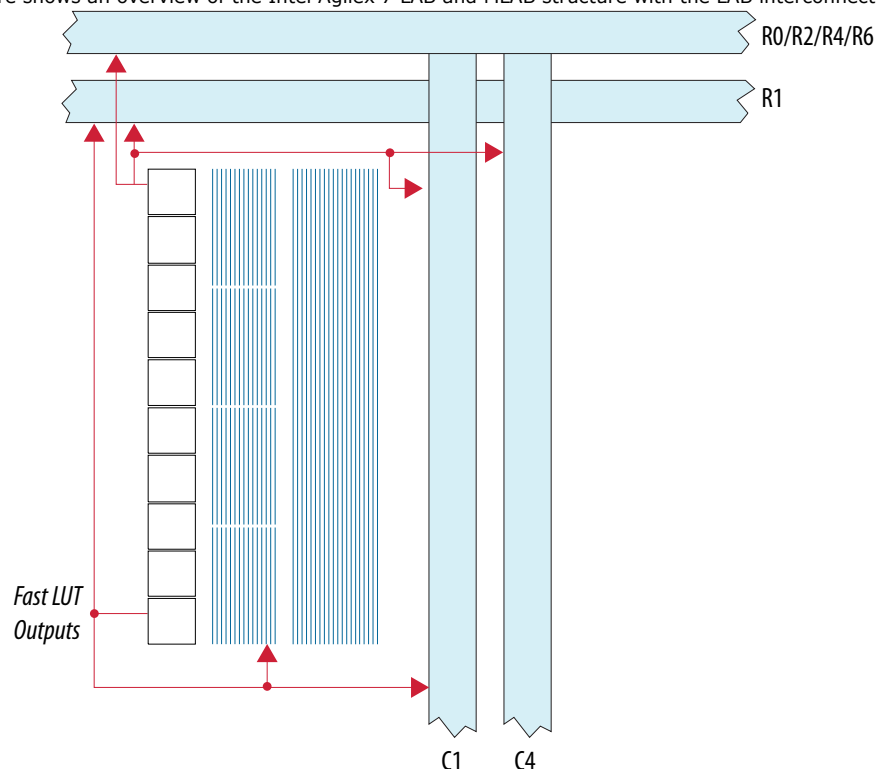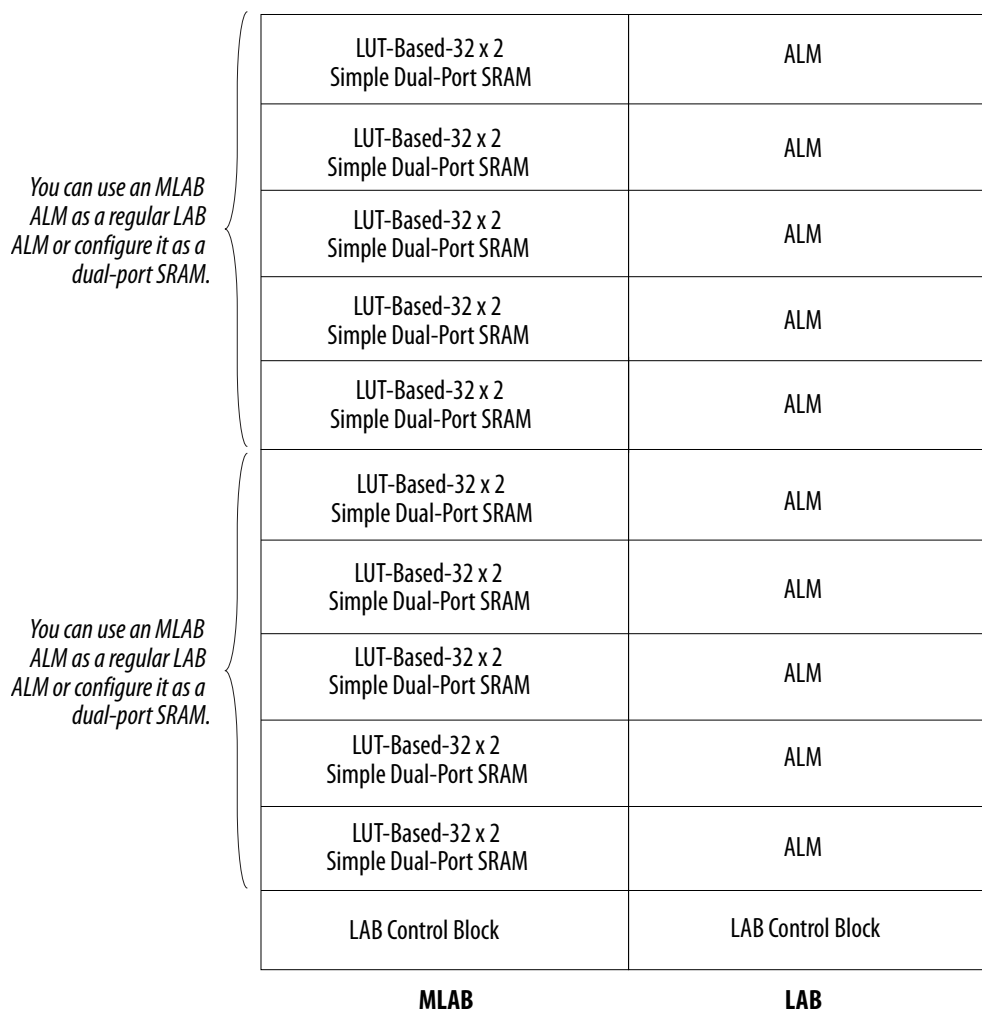

Related Information

MLAB on page 5

3.1.1. 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 (depth) x 2 (width) memory block, resulting in a configuration of 32 (depth) x 20 (width) simple dual-port SRAM block.

**ISO
9001:2015
Registered**

Figure 2.   Intel Agilex 7 LAB and MLAB Structure

LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LUT-Based-32 x 2 Simple Dual-Port SRAM	ALM
LAB Control Block	LAB Control Block

You can use an MLAB ALM as a regular LAB ALM or configure it as a dual-port SRAM.

You can use an MLAB ALM as a regular LAB ALM or configure it as a dual-port SRAM.

MLAB   **LAB**

3.1.2. Local and Direct Link Interconnects

Each LAB can drive out 60 ALM outputs. A subset of these can directly drive LAB inputs, but any connection to a different row or column must use at least one general-purpose routing wire.

The local interconnect drives the ALM inputs. ALM outputs, as well as column and row interconnects drive the local interconnect.

Send Feedback

Figure 3. **Intel Agilex 7 LAB Local and Direct Link Interconnects**
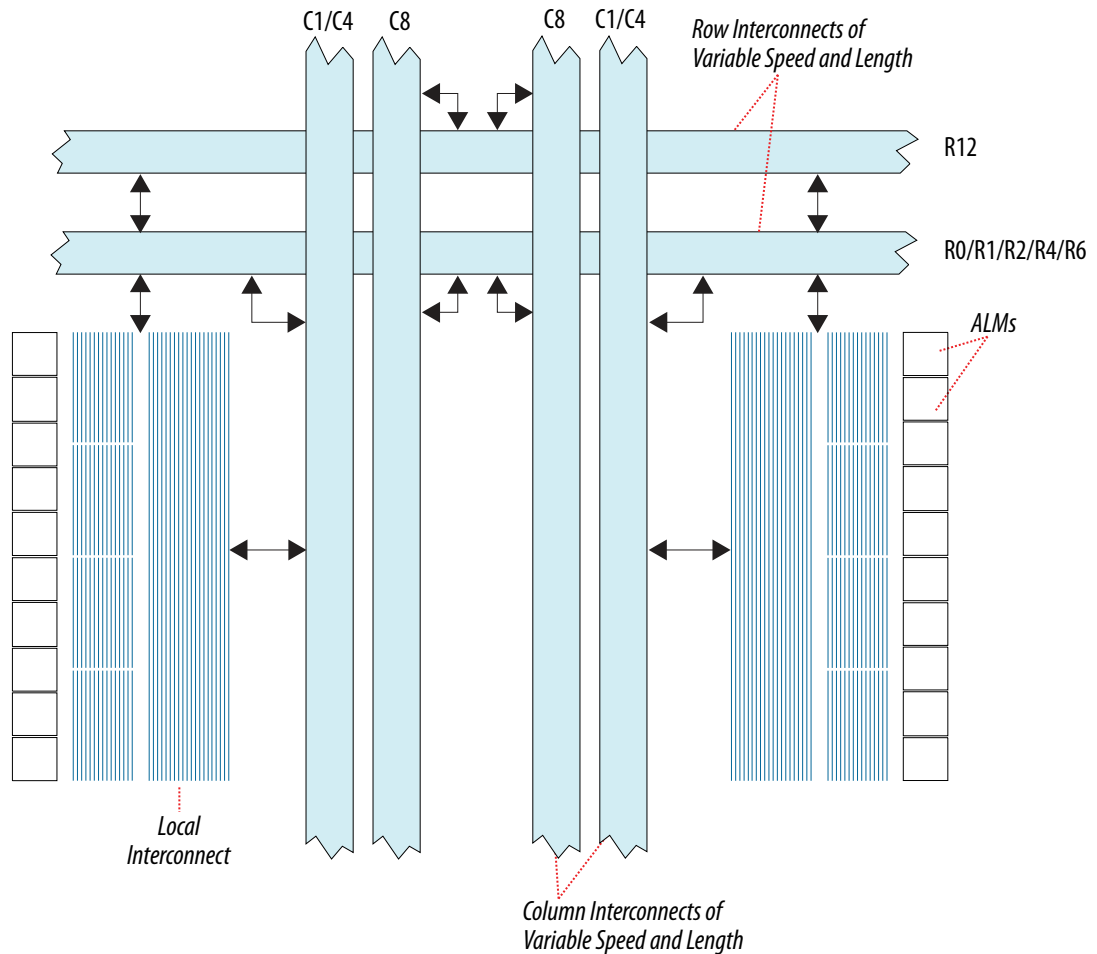

3.1.3. Carry Chain Interconnects

There is a dedicated carry chain path between the ALMs. Intel Agilex 7 devices include an enhanced interconnect structure in LABs for routing carry chains for efficient arithmetic functions. These ALM-to-ALM connections bypass the local interconnect.

The Intel Hyperflex registers are added to the carry chain to enable flexible retiming across a chain of LABs and the Intel Quartus Prime Compiler automatically takes advantage of these resources to improve utilization and performance.

**Figure 4.   Carry Chain Interconnects**
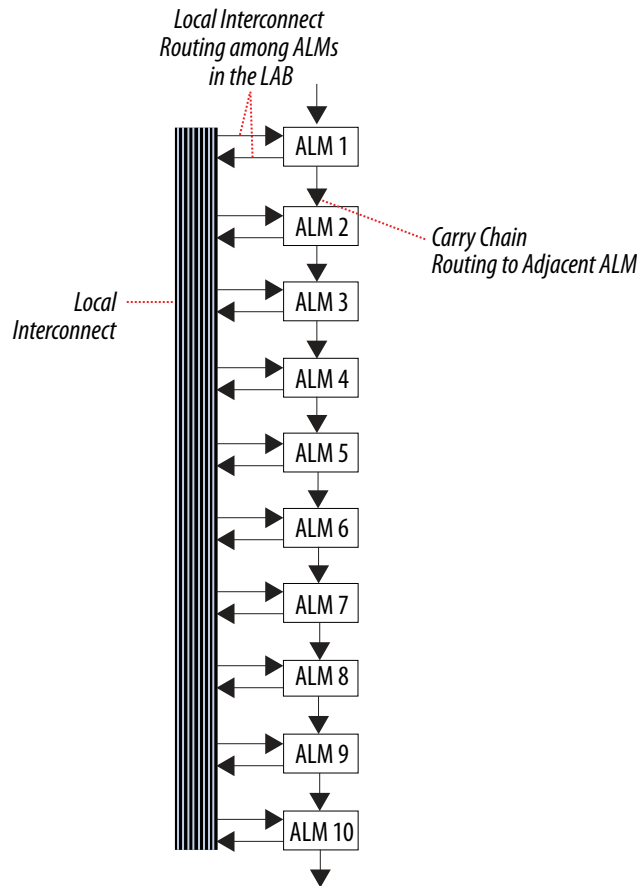
3.1.4. LAB Control Signals

There are two clock sources in each LAB control block, which generate two LAB clocks (`LABCLK[1:0]`) and two delayed LAB clocks (`LABCLK_Phi1[1:0]`) to drive the ALM registers and Hyper-Registers in the LAB. The LAB supports two unique clock enable signals, as well as additional clear signals, for the ALM registers.

The LAB row clocks [5..0] and LAB local interconnects generate the LAB-wide control signals. A low skew clock network distributes global signals to the row clocks [5..0]. The MultiTrack interconnect consists of continuous, performance-optimized routing lines of different lengths and speeds used for routing efficiency. The Intel Quartus Prime Compiler automatically routes critical design paths on faster interconnects to improve design performance and optimizes the device resources.

3.1.4.1. Clear Logic Control

LAB-wide signals control the logic for the ALM register's clear signal. The ALM register directly supports both a synchronous and an asynchronous clear. Each LAB supports one synchronous clear signal and two asynchronous clear signals.

intel.

Figure 5. **Intel Agilex 7 LAB-Wide Control Signals**
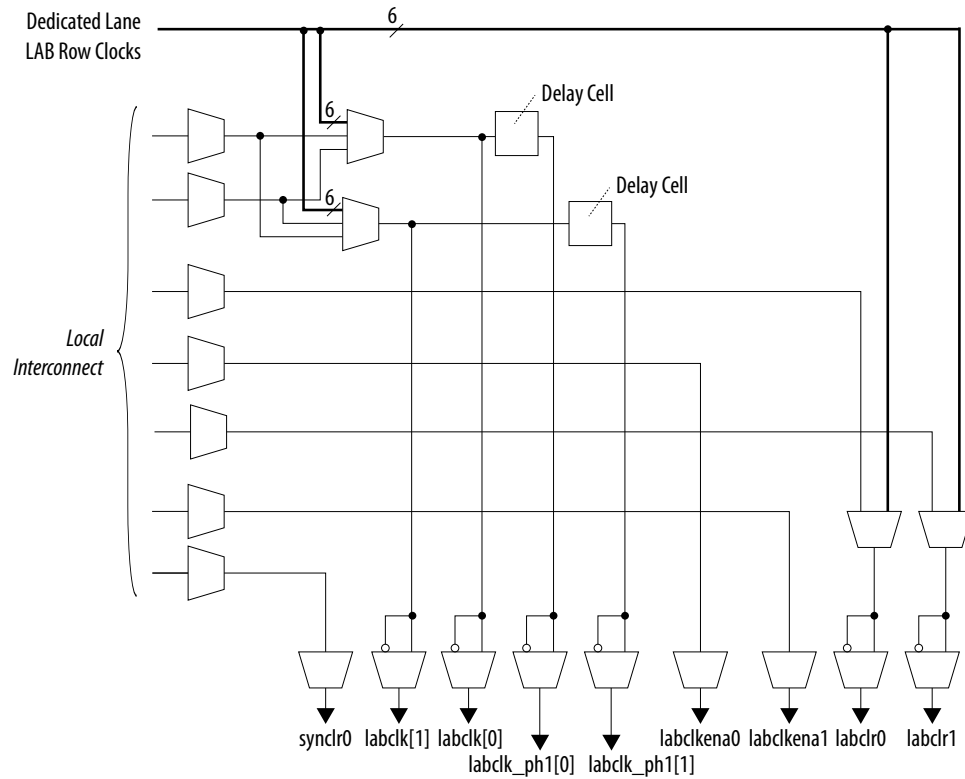
3.2. ALM

The following sections cover the ALM resources, ALM output, and ALM operating modes.

3.2.1. ALM Resources

Each ALM contains a variety of LUT-based resources that can be divided between two combinational adaptive LUTs (ALUTs), a two-bits full adder, and four registers.

With up to eight inputs for the two combinational ALUTs, one ALM can implement various combinations of two functions. This adaptability allows an ALM to be completely backward-compatible with four input LUT architectures. One ALM can also implement a subset of eight input functions.

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

- Data in
- Data out
- Normal LAB clock
- Delayed LAB clock

- Clock enable

- Synchronous clear

- Asynchronous clear

Global signals, general-purpose I/O (GPIO) pins, or any internal logic can drive the clock enable signal, clock, and asynchronous or synchronous clear control signals of an ALM register. The clock enable signal has priority over synchronous reset signal.

For combinational functions, the registers are bypassed and the output of the look-up table (LUT) and adders drives directly to the outputs of an ALM. Two fast outputs are available for 6 LUT outputs and bottom 5 LUT outputs to bypass the output mux and connect to another LAB for critical path adjustment.

Figure 6.   Intel Agilex 7 ALM High-Level Block Diagram
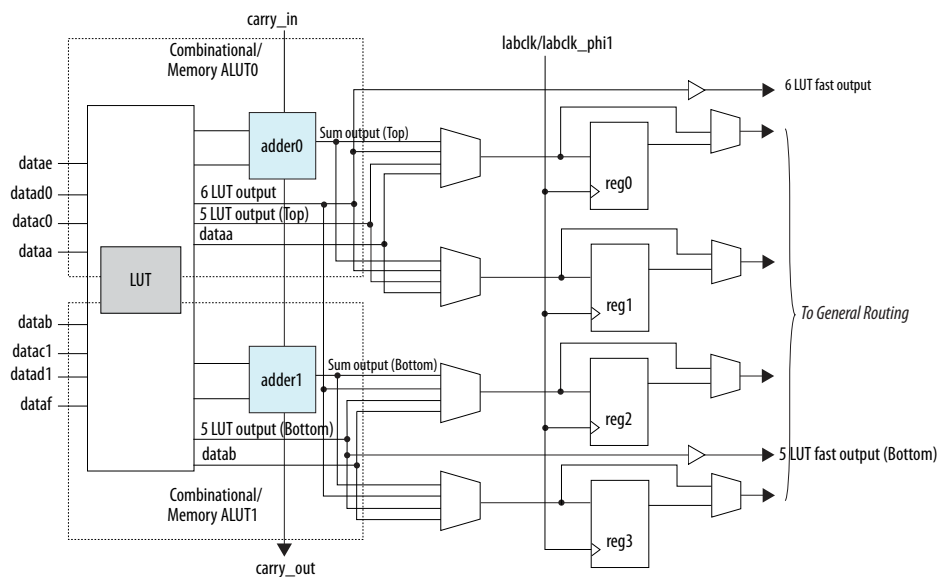

3.2.2. ALM Output

The general routing outputs in each ALM drive the local, row, and column routing resources. Six ALM outputs, including two fast output paths, can drive column, row, or direct link routing connections.

The LUT, adder, or register output can drive the ALM outputs. Both the LUT or adder and the ALM register can drive out of the ALM simultaneously.

Register packing improves device utilization by allowing unrelated register and combinational logic to be packed into a single ALM. The ALM can also drive out registered and unregistered versions of the LUT or adder output.

The following figure shows the Intel Agilex 7 ALM connectivity. In the Intel Quartus Prime Resource Property Editor, the entire ALM connection is simplified. Some routings will be routed internally by the Intel Quartus Prime software.

Figure 7.   Intel Agilex 7 ALM Connection Details
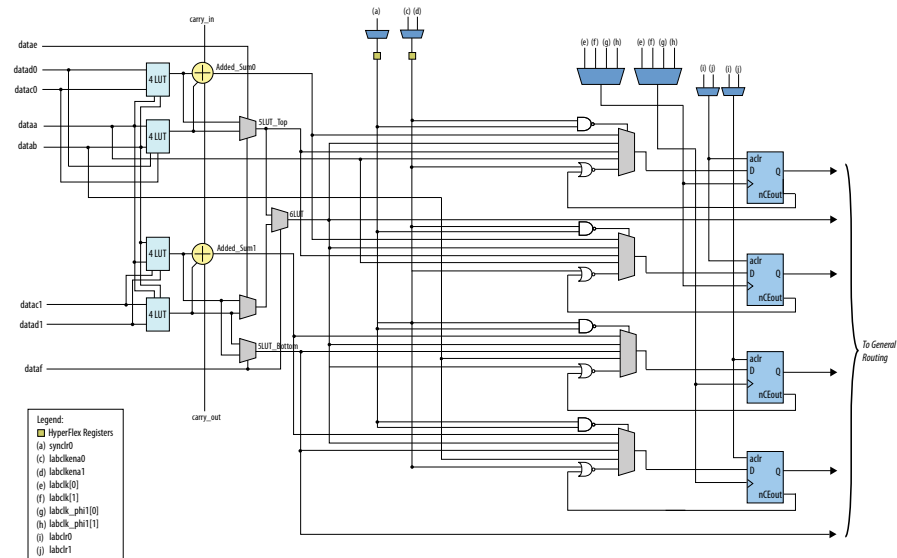
3.2.3. ALM Operating Modes

The Intel Agilex 7 ALM operates in any of the following modes:

- Normal mode
- Extended LUT mode
- Arithmetic mode

3.2.3.1. Normal Mode

Normal mode allows two functions to be implemented in one Intel Agilex 7 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. The Intel Quartus Prime Compiler automatically selects the inputs to the LUT. ALMs in normal mode support register packing.

The following figure shows a combination of different input connections for the LUT mode. In your design, the Intel Quartus Prime software may assign different input namings during compilation.

Figure 8. **ALM in Normal Mode**
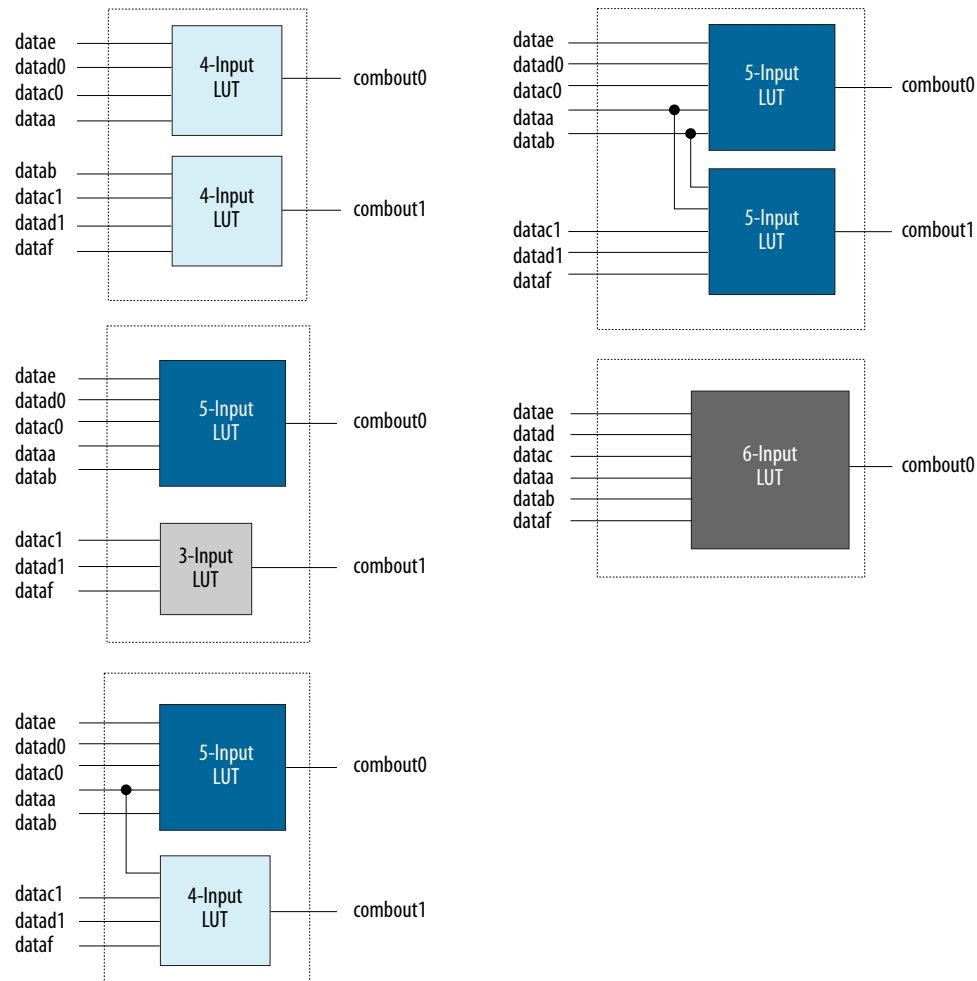
Combinations of functions with fewer inputs than those shown are also supported. For example, combinations of functions with the following number of inputs are supported.

- 4 and 3
- 3 and 3
- 3 and 2
- 5 and 2

For the packing of two 5-input functions into one ALM, the functions must have at least two common inputs. The common inputs are `dataa` and `datab`. The combination of a 4-input function with a 5-input function requires one common input (either `dataa` or `datab`).

In a sparsely used device, functions that could be placed in one ALM may be implemented in separate ALMs by the Intel Quartus Prime software to achieve the best possible performance. As a device begins to fill up, the Intel Quartus Prime software automatically uses the full potential of the Intel Agilex 7 ALM. The Intel Quartus Prime Compiler automatically searches for functions using common inputs or completely

Intel Agilex® 7 Logic Array Blocks and Adaptive Logic Modules User Guide

intel.

independent functions to be placed in one ALM to make efficient use of device resources. In addition, you can manually control resource use by setting location assignments.

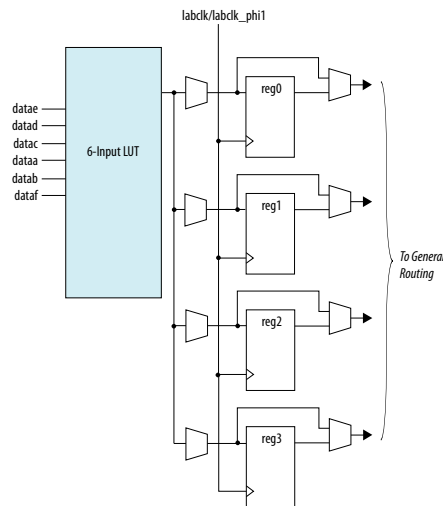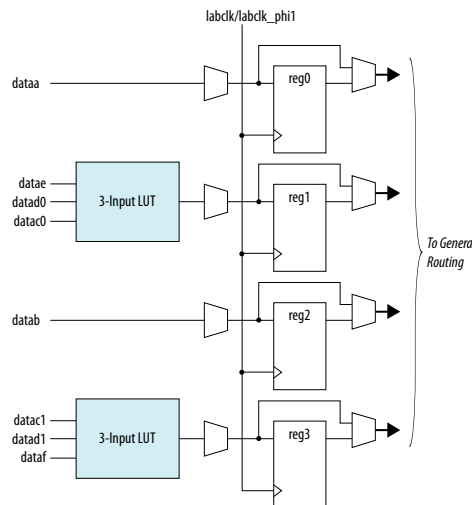**Figure 9.   6-Input LUT Mode Function in Normal Mode**

Figure 10.   3-Input LUT Mode Function in Normal Mode

`dataa` and `datab` are available for register packing.

You can implement any three to six input function using the following inputs:

- `datae`
- `datad0`
- `datac0`
- `datac1`

- `datad1`

- `dataf`

- `dataa` and `datab`—whereby `dataa` and `datab` are shared across both LUTs to provide flexibility to implement a different function in each LUT.

Both `dataa` and `datab` inputs support the register packing feature. If you enable the register packing feature, both `dataa` and `datab` inputs or either one of the inputs bypass the LUT and directly feed into the register, depending on the packed register mode used. For Intel Agilex 7 devices, the following types of packed register modes are supported:

- 5-input LUT with 1 packed register path

- 5-input LUT with 2 packed register paths

- Two 3-input LUTs with 2 packed register paths

The 3-input LUT with 2 packed register paths is illustrated in the 3-Input LUT Mode Function in Normal Mode figure. For Intel Agilex 7 devices, the 6-input LUT mode does not support the register packing feature.

3.2.3.2. Extended LUT Mode

Figure 11. Supported 8-Input Functions in the Extended LUT Mode
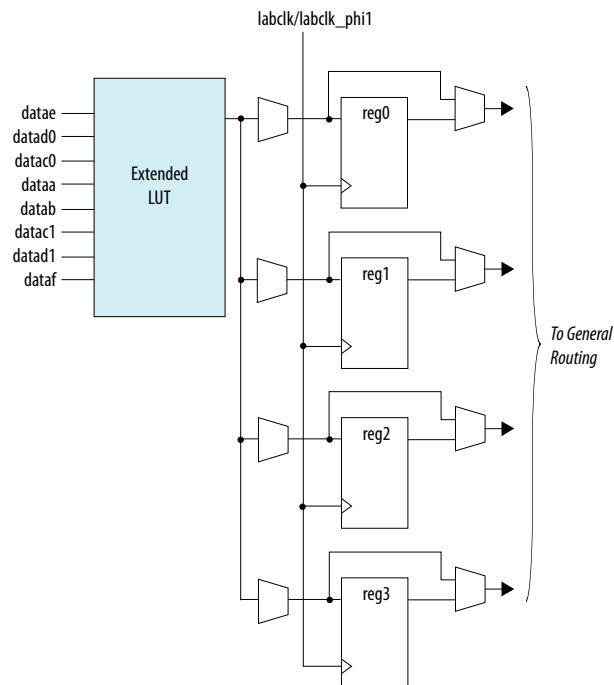
Certain 8-input functions can be implemented in a single ALM using all the LUT inputs:

- `datae`

- `datad0`

- `datac0`

- `dataa`

- `datab`

- `datac1`

- `datad1`

- `dataf`

In the 8-input extended LUT mode, the packed register mode is supported, provided that the packed register shares a `dataa` or `datab` input with the 8-input LUT.
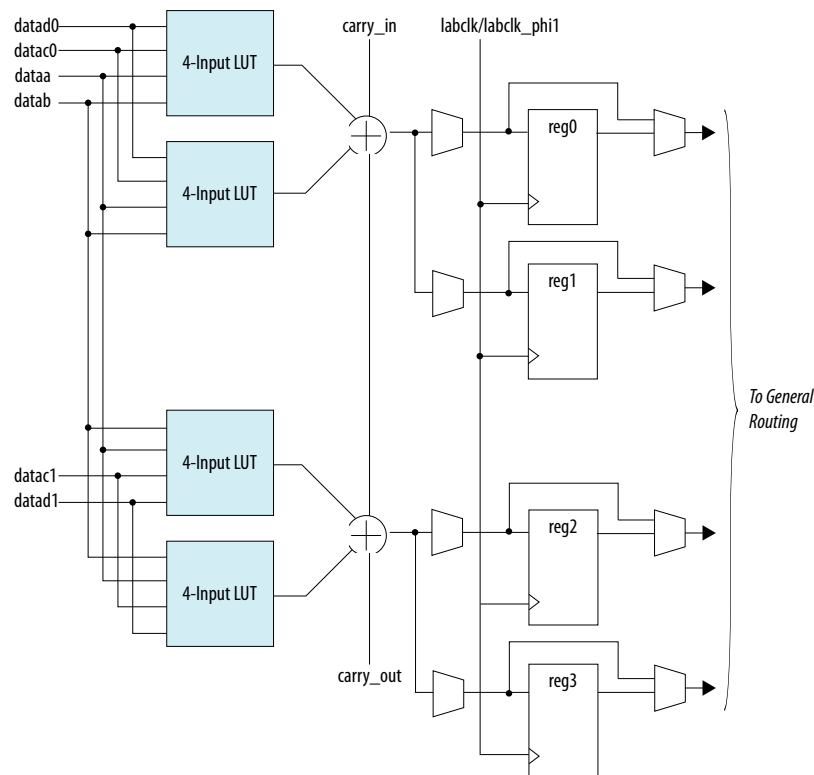
3.2.3.3. Arithmetic Mode

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.

Arithmetic mode also offers clock enable, counter enable, synchronous up and down control, add and subtract control, and synchronous clear.

The clear and clock enable options are LAB-wide signals that affect all registers in the LAB. You can individually disable or enable these signals for each pair of registers in an adaptive LUT (ALUT). The Intel Quartus Prime software automatically places any registers that are not used by the counter into other LABs.

Figure 12.   Intel Agilex 7 ALM in Arithmetic Mode

3.2.3.3.1. Carry Chain

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

The 2-bit carry select feature in Intel Agilex 7 devices splits the propagation delay of carry chains with the ALM. Carry chains can begin in either the first ALM or the sixth ALM in a LAB. The final carry-out signal is routed to an ALM, where it is fed to local, row, or column interconnects.

intel.

4. Document Revision History for the Intel Agilex 7 Logic Array Blocks and Adaptive Logic Modules User Guide

Document Version	Changes
2023.03.27	Updated product family name to "Intel Agilex 7".
2022.05.24	Updated Figure: *Intel Agilex 7 ALM High-Level Block Diagram*.
2019.11.14	Updated the description in the *LAB Control Signals* section.
2019.10.01	• Updated the details for the synchronous and asynchronous clear signals in the *Clear Logic Control* section. • Updated Figure: *Intel Agilex 7 LAB Structure and Interconnects Overview*. • Updated Figure: *Intel Agilex 7 LAB Local and Direct Link Interconnects*. • Updated Figure: *Intel Agilex 7 ALM Connection Details*. • Updated Figure: *Intel Agilex 7 LAB-Wide Control Signals*. • Updated Figure: *Intel Agilex 7 ALM High-Level Block Diagram*. • Removed the *Time Borrowing and Latch Modes* section.
2019.04.02	Initial release.