

Cyclone V器件中的逻辑阵列模块与自适应逻辑模块

1

2013.05.06

CV-52001

✉ 订阅 💬 反馈

本章节介绍了Cyclone® V核心架构中逻辑阵列模块(LAB)的功能特性。

LAB是由称作自适应逻辑模块(ALM)的基本构造模块组成,通过配置这些模块能够实现逻辑功能、算术功能以及寄存器功能。

您可以将Cyclone V器件中1/4的LAB用作存储器LAB (MLAB)。

Quartus® II软件和所支持的第三方综合工具,与参数化功能(例如参数化模块库(LPM))一起,对常用功能(例如:计数器、加法器、减法和算术功能)自动选择相应的模式。

本章节涵盖以下两方面内容:

- LAB
- ALM操作模式

相关链接

[Cyclone V器件手册:已知问题](#)

列出了Cyclone V器件手册章节的计划更新。

LAB

LAB是由一组逻辑资源组成的可配置逻辑模块。每个LAB均包含专用逻辑,用来将控制信号驱动到ALM中。

MLAB是LAB的超集,包含了LAB的所有特性。

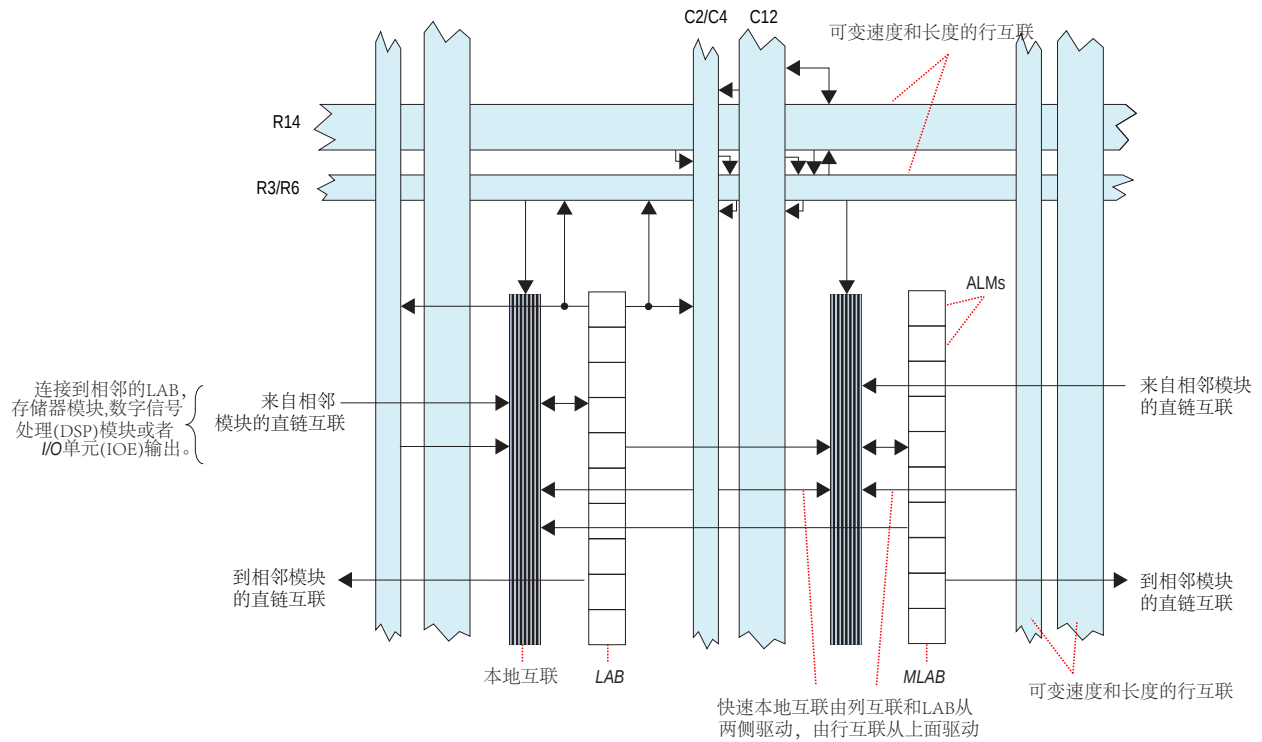
© 2013 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. 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.

ISO
9001:2008
Registered



图1-1: Cyclone V器件中的LAB结构和互联概况

此图显示了基于LAB互联的Cyclone V LAB和MLAB结构。

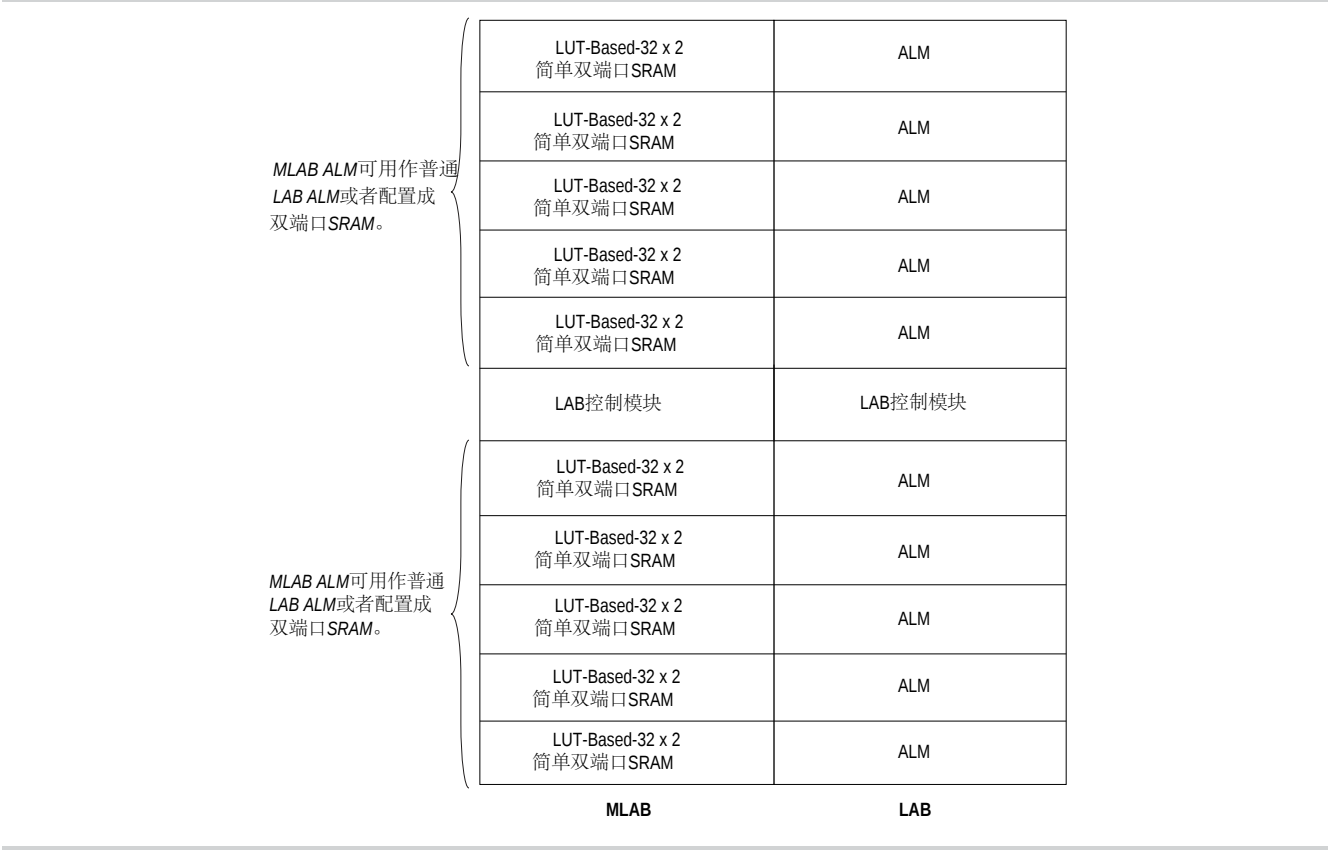


MLAB

每个MLAB均支持最大640 bit的简单双端口SRAM。

您可以将MLAB中的每一个ALM配置成32 x 2存储器模块，生成一个32 x 20简单双端口SRAM模块。

图1-2: Cyclone V 器件的LAB和MLAB结构



本地和直链互联(Local and Direct Interconnects)

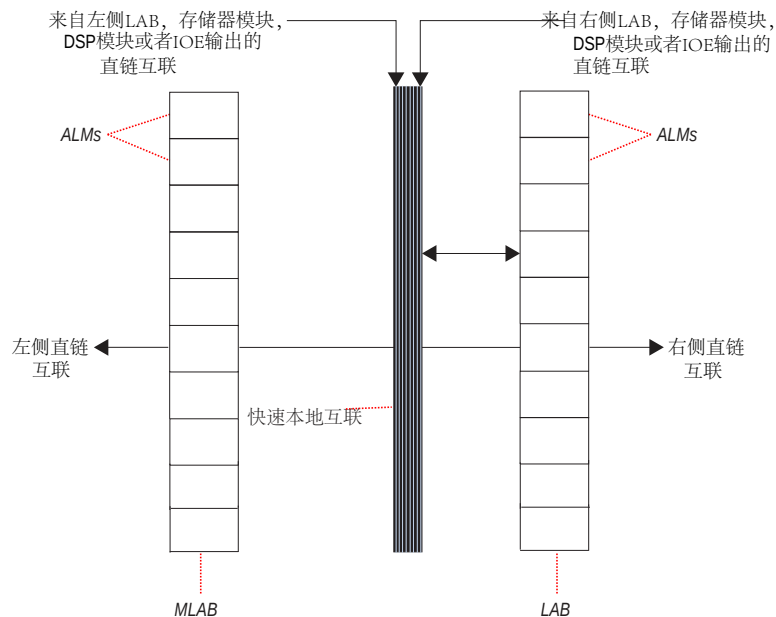
每个LAB能够通过快速本地和直链互联驱动30个ALM。10个ALM位于任意给定的LAB中，10个ALM位于每个相邻的LAB中。

LAB本地互联通过使用相同LAB中的行列互联以及ALM输出来驱动相同LAB中的ALM。

相邻的LAB，MLAB，M10K模块，或者左侧的数字信号处理(DSP)模块也能够通过直链连接来驱动LAB的本地互联。

直链互联功能最大限度地降低了行列互联的使用，从而提供了更高的性能和更大的灵活性。

图1-3: Cyclone V器件的LAB快速本地和直链互联



LAB控制信号

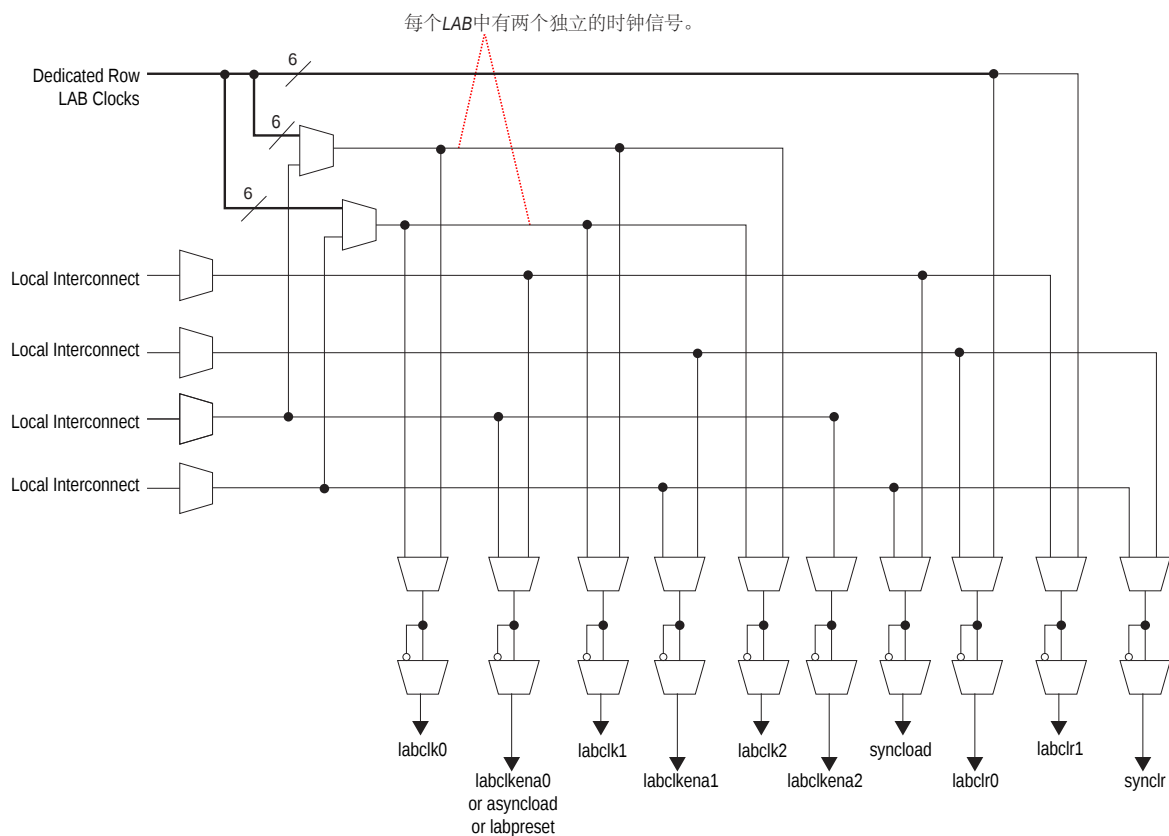
每个LAB均包含专用逻辑，将控制信号驱动到它的ALM中，并且包含两个独特的时钟源和三个时钟使能信号。

LAB控制模块使用两个时钟源和三个时钟使能信号生成高达三个时钟。每个时钟和时钟使能信号都是相连的。

置低时钟使能信号会关闭相应的全LAB(LAB-wide)时钟。

图1-4: Cyclone V 器件的全LAB控制信号

此图显示了LAB中的时钟源和时钟使能信号。



ALM资源

一个ALM包含四个可编程寄存器。每个寄存器包含如下端口：

- 数据
- 时钟
- 同步和异步清零
- 同步加载

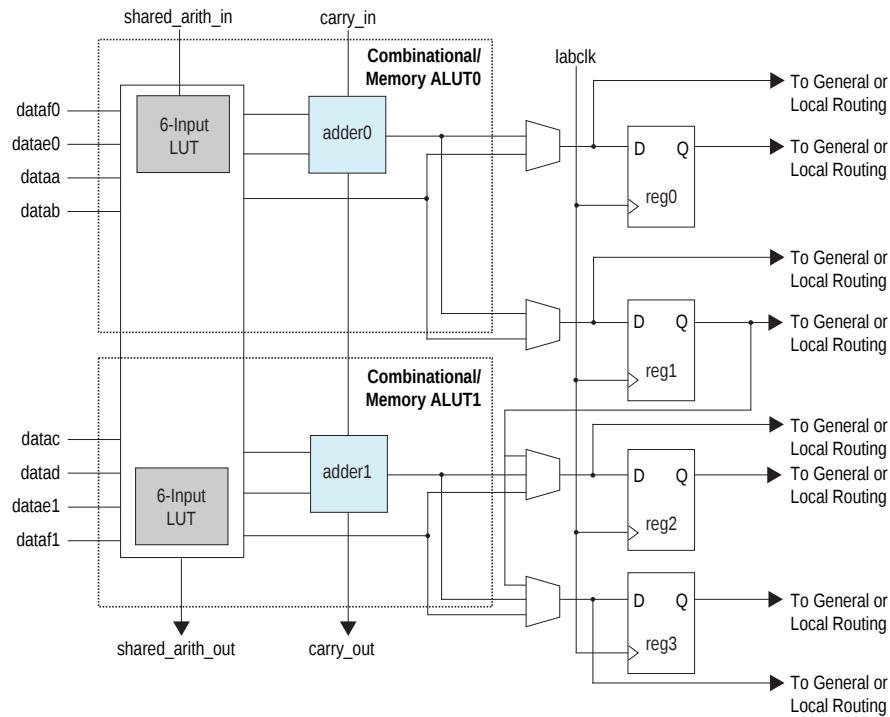
全局信号，通用I/O(GPIO)管脚或者任何内部逻辑都可以驱动ALM寄存器的时钟和清零控制信号。

GPIO管脚或内部逻辑驱动时钟使能信号。

对于组合功能，寄存器被旁路，LUT的输出直接驱动到ALM的输出。

注意：Quartus II针对已优化的性能自动配置ALM。

图1-5: Cyclone V 器件的ALM高级结构图



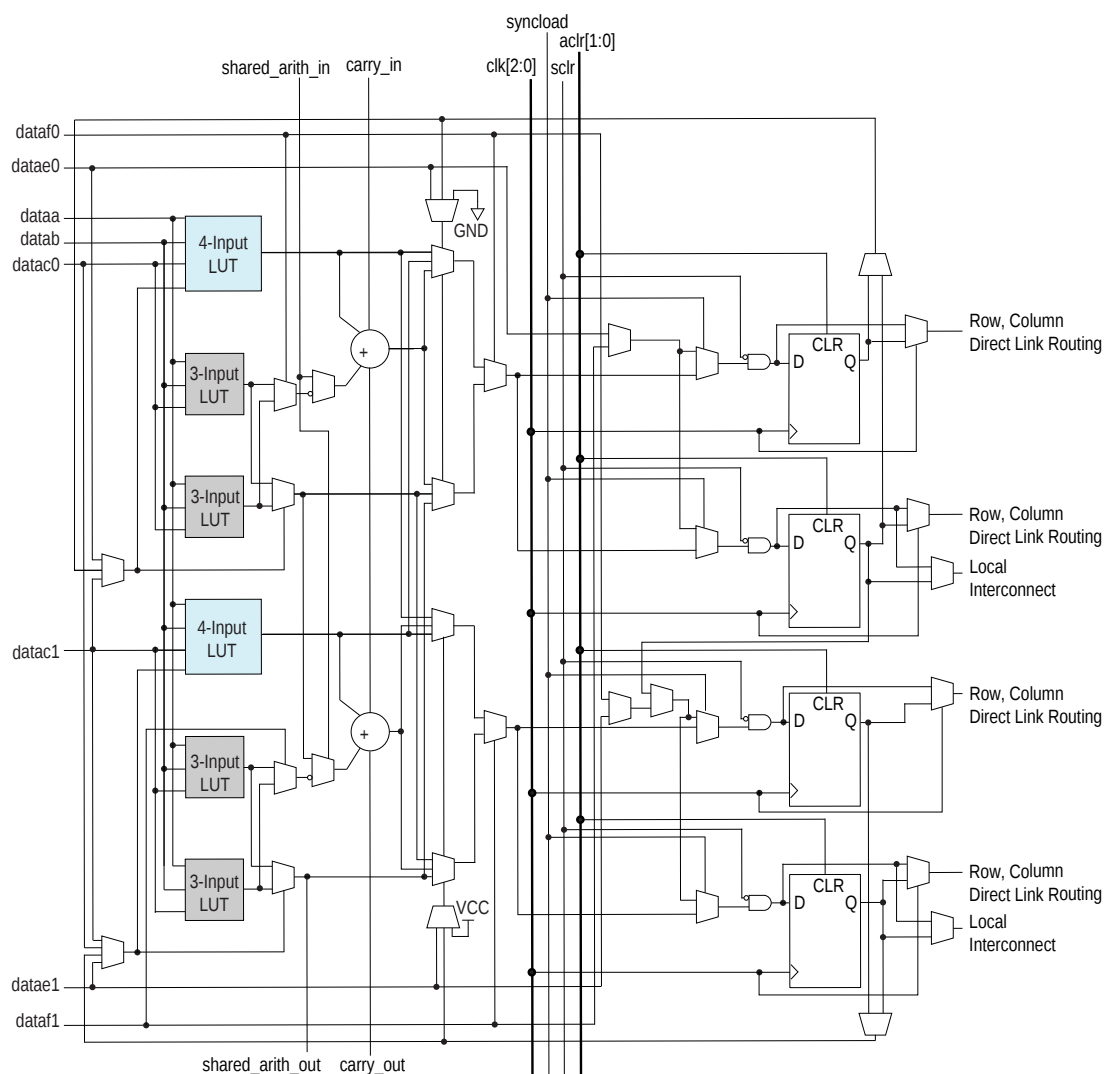
ALM输出

每个ALM中的通用布线输出驱动本地，行和列布线资源。两个ALM输出能驱动行、列或者直链布线连接，并且其中的一个ALM输出也能够驱动本地互联资源。

LUT，加法器或者寄存器输出能驱动ALM输出。LUT或加法器能够驱动一个输出，而寄存器驱动另一个输出。

寄存器封装(Register Packing)通过将无关的寄存器和组合逻辑封装在一个ALM中来提高器件利用率。改善布局布线的另一种机制是支持寄存器输出驱动回相同ALM的LUT中，使寄存器与其本身的扇出LUT封装在一起。ALM也能够驱动寄存的LUT以及未寄存的LUT或者加法器输出。

图1-6: Cyclone V 器件的ALM连接明细



ALM操作模式

Cyclone V ALM运行在下面其中的一个模式中:

- 正常模式
- 扩展LUT模式
- 算术模式
- 共享算术模式

正常模式

在正常模式下，可以在一个Cyclone V ALM中实现两个功能，或者一个6输入功能。

来自LAB本地互联的多达八个数据输入是组合逻辑的输入。

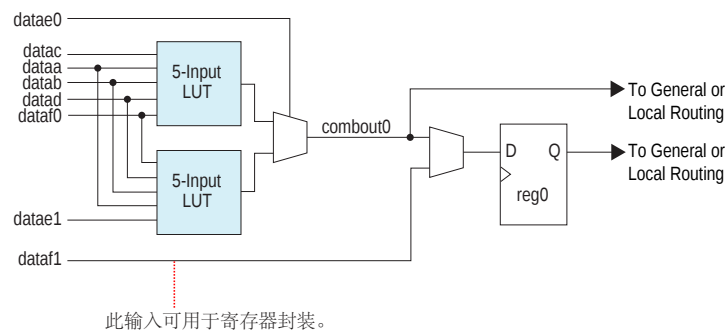
ALM能够支持某些完全独立的功能组合，以及具有共同输入的多种功能的组合。

扩展LUT模式(Extended LUT Mode)

在此模式中，如果7输入功能是未寄存的，那么未使用的第8个输入可用于寄存器封装。

符合模板的功能(如下图所示)经常作为Verilog HDL或VHDL代码中的“if-else”语句出现在设计中。

图1-7: Cyclone V器件的扩展LUT模式中所支持的7输入功能模板



算术模式

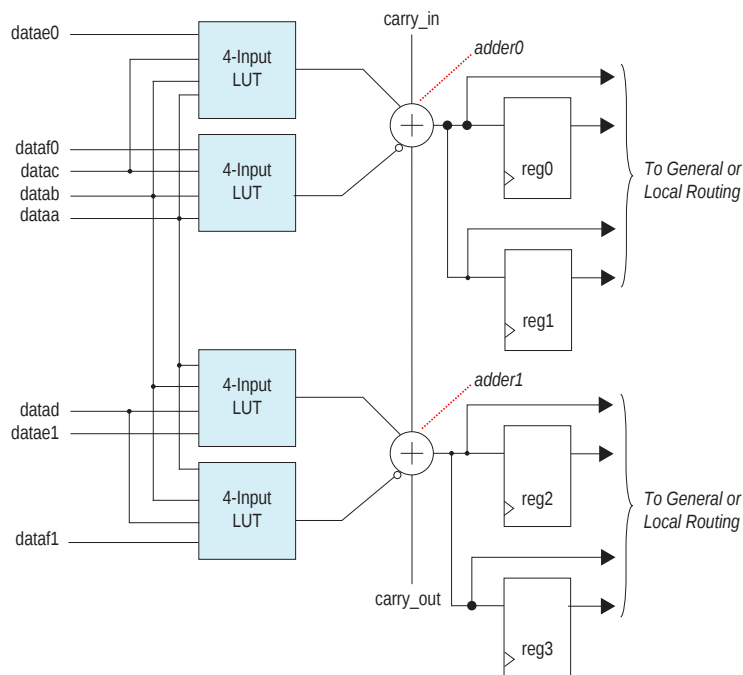
算术模式中的ALM使用两个4输入LUT以及两个专用全加法器。

专用加法器使LUT可用于执行预加器逻辑；因此，每一个加法器能够将两个4输入功能的输出相加。

ALM支持同时使用加法器的进位输出和组合逻辑输出。在此操作中，加法器输出被忽略。

对于使用此模式的功能，使用加法器与组合逻辑输出将节省高达50%的资源。

图1-8: Cyclone V器件的算术模式ALM



进位链(Carry Chain)

进位链提供了算术或者共享算术模式中专用加法器之间的快速进位功能。

Cyclone V器件中的2-bit进位选择功能使ALM中的进位链传播延迟减少了一半。进位链能够开始于LAB中的第一个ALM或者第五个ALM。最终的carry-out信号会被传输到ALM中，在此ALM中驱动到本地、行或列互联。

当实现高扇进(high fan-in)算术功能时，要避免器件中一小块区域内的布线拥塞，连接到下一个LAB之前，LAB能够支持仅使用LAB的上半部或者下半部的进位链。这样，LAB中的ALM另半部可用于实现正常模式中较窄的扇进功能。使用第一个LAB中的上面五个ALM的进位链传输到列中下一个LAB中的ALM的上半部。使用第一个LAB中的下面五个ALM的进位链传输至列中下一个LAB中的ALM的下半部。您可以旁路LAB列的上半部和MLAB列的下半部。

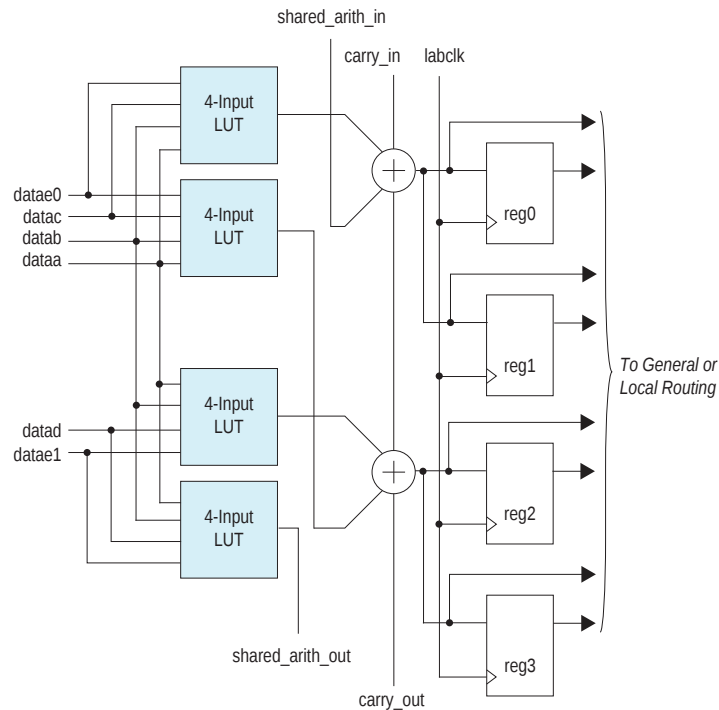
Quartus II Compiler通过自动将LAB连接在一起来创建超过20个ALM长的进位链(算术或者共享算术模式的10个ALM)。要增强布局布线,长进位链需要纵向运行，从而实现到TriMatrix存储器和DSP模块的快速水平连接。进位链能够持续运行直到全列。

共享算术模式

共享算术模式的ALM能够实现ALM中的3输入加法运算。

此模式通过四个4输入LUT配置ALM。每个LUT将计算三个输入的和，或者计算三个输入的进位。通过使用称作共享算术链的专用连接将进位计算的输出传送到下一个加法器。

图1-9: Cyclone V器件的共享算术模式ALM



共享算术链

增强算术模式中共享算术链使ALM能够实现3输入加法，这样可以显著降低用于实现大型加法器树或者相关器功能的必要资源。

共享算术链可以开始于LAB中的第一个或者第六个ALM。

与进位链类似，相隔LAB列中的共享算术链的上半部和下半部能够被旁路。这一性能使得共享算术链通过LAB中的一半ALM进行级联，而剩下的一半用于较窄扇进功能。在每个LAB中，列的上半部是可旁路的；而在MLAB中，列的下半部是可旁路的。

Quartus II Compiler通过自动将LAB连接在一起来创建超过20个ALM长的共享算术链(算术或者共享算术模式中的10个ALM)。要增强布局布线，需要长共享算术链纵向运行，从而快速水平连接到TriMatrix存储器和DSP模块。共享算术链能持续运行直到全列。

文档修订历史

日期	版本	修订内容
2013年5月	2013.05.06	<ul style="list-style-type: none"> 对Knowledge Base中的已知文档问题添加了链接。 删除了ALM输出部分的寄存器输出信息。 删除了ALM高级结构图和ALM链接明细图中的reg_chain_in和reg_chain_out端口。
2012年12月	2012.12.28	重组内容并更新模板。

日期	版本	修订内容
2012年6月	2.0	针对Quartus IIV12.0的发布进行了更新： <ul style="list-style-type: none">• 重构章节。• 更新了图 1 - 6。
2011年11月	1.1	少量文本编辑。
2011年10月	1.0	首次发布。