



# インテル® Stratix® 10 デバイスのロジック・ アレイ・ブロックおよびアダプティブ・ロジック・モ ジュール・ユーザーガイド



## 目次

---

<b>1</b>	<b>インテル® Stratix® デバイスの LAB および ALM の概要</b>	<b>3</b>
<b>2</b>	<b>HyperFlex レジスター</b>	<b>4</b>
<b>3</b>	<b>インテル Stratix デバイスの LAB および ALM のアーキテクチャーと機能</b>	<b>5</b>
3.1	LAB	5
3.1.1	MLAB	6
3.1.2	ローカル・インターコネクトおよびダイレクトリンク・インターコネクト	6
3.1.3	キャリアチェーンのインターコネクト	7
3.1.4	LAB コントロール信号	8
3.2	ALM	9
3.2.1	ALM リソース	9
3.2.2	ALM 出力	10
3.2.3	ALM 動作モード	11
<b>4</b>	<b>インテル Stratix デバイスの LAB および ALM ユーザーガイドの改訂履歴</b>	<b>18</b>



## 1 インテル® Stratix® デバイスの LAB および ALM の概要

---

LAB ( ロジック・アレイ・ブロック ) は、ALM ( アダプティブ・ロジック・モジュール ) として知られる基礎的なビルディング・ブロックで構成されています。LAB を構成することで、ロジック・ファンクション、演算ファンクション、およびレジスター・ファンクションが実装可能となります。

また、インテル® Stratix® デバイスで使用可能な LAB の 4 分の 1 を MLAB ( メモリー LAB ) として使用することができます。なお、一部のデバイスはより高い MLAB 比率を有します。

インテル Quartus® Prime ソフトウェアおよびサポートされるサードパーティーの合成ツールは、カウンター、加算器、減算器、および演算ファンクションなどの一般的な機能に適切なモードを自動的に選択します。

### 関連情報

#### HyperFlex コア・アーキテクチャー、インテル Stratix デバイスの概要

Hyper-Register および HyperFlex コア・アーキテクチャーについて詳しい情報を提供します。Hyper-Register は、LAB の入力および出力へ接続された配線セグメントを含む、コア・ファブリック全体の各インターコネクト配線セグメントで使用可能な追加のレジスターです。



## 2 HyperFlex レジスター

---

インテル Stratix 10 デバイスファミリーは、インテル Hyperflex™ コア・アーキテクチャーを搭載しています。

インテル Stratix 10 デバイスの LAB はリタイミングを容易にするために設計された インテル Hyperflex レジスターとその他の機能が含まれています。インテル Hyperflex レジスターは ALM と キャリーチェーンで使用可能です。インテル Stratix 10 デバイスの ALM 接続の詳細の図で示されるように、インテル Hyperflex レジスターは実効なパス遅延を増減するために同期クリアーおよびクロックイネーブル入力に配置されます。リタイミング中、すべての インテル Hyperflex レジスターは有効になり、インテル Quartus Prime ソフトウェアにより制御されます。

---

Intel Corporation. 無断での引用、転載を禁じます。Intel、インテル、Intel ロゴ、Altera、ARRIA、CYCLONE、ENPIRION、MAX、NIOS、QUARTUS および STRATIX の名称およびロゴは、アメリカ合衆国および/またはその他の国における Intel Corporation の商標です。インテルは FPGA 製品および半導体製品の性能がインテルの標準保証に準拠することを保証しますが、インテル製品およびサービスは、予告なく変更される場合があります。インテルが書面にて明示的に同意する場合を除き、インテルはここに記載されたアプリケーション、または、いかなる情報、製品、またはサービスの使用によって生じるいっさいの責任を負いません。インテル製品の顧客は、製品またはサービスを購入する前、および、公開済みの情報を信頼する前には、デバイスの仕様を最新のバージョンにしておくことをお勧めします。

\*その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

ISO  
9001:2008  
登録済

### 3 インテル Stratix デバイスの LAB および ALM のアーキテクチャーと機能

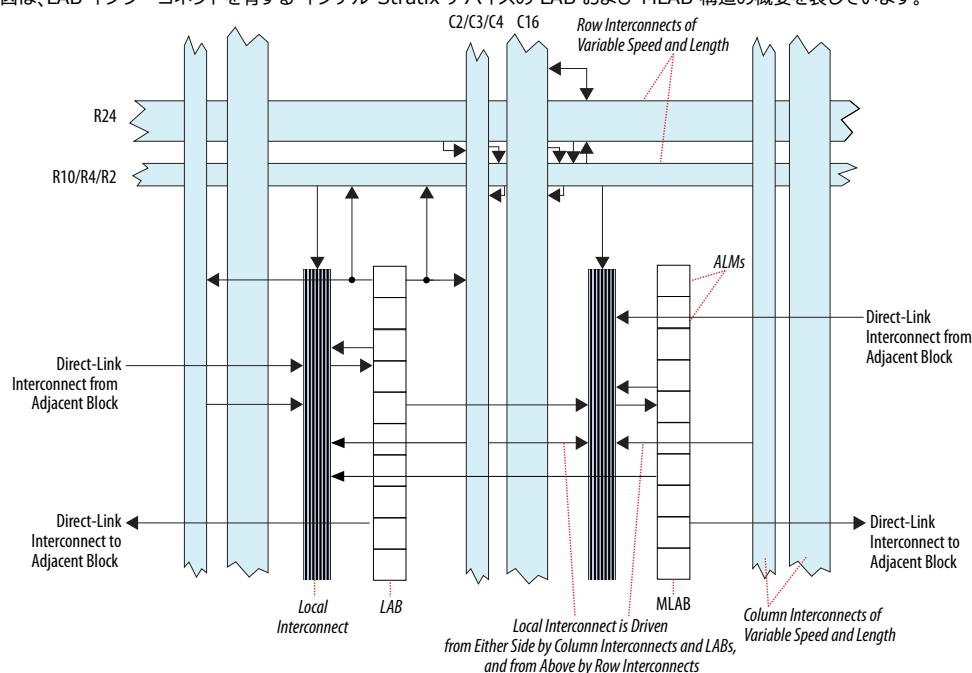
以下の項では、インテル Stratix デバイスの LAB および ALM について解説します。

#### 3.1 LAB

LAB とは、ロジックリソースのグループで構成されるコンフィグレーション可能なロジックブロックのことを指します。各 LAB には ALM にコントロール信号を駆動するための専用ロジックが含まれます。MLAB は LAB のスーパーセットであり、LAB の機能をすべて備えています。インテル Stratix デバイスの LAB および MLAB 構造の図で示すように、各 LAB には計 10 個の ALM が装備されています。

図 -1: インテル Stratix デバイスの LAB 構造およびインターコネクットの概要

この図は、LAB インターコネクットを有する インテル Stratix デバイスの LAB および MLAB 構造の概要を表しています。



#### 関連情報

[MLAB \(6 ページ\)](#)

### 3.1.1 MLAB

各 MLAB は最大 640 ビットのシングル・デュアルポート SRAM をサポートしています。MLAB の各 ALM は、32 ( 深さ ) x 2 ( 幅 ) のメモリーブロックとしてコンフィグレーションでき、32 ( 深さ ) x 20 ( 幅 ) のシングル・デュアルポート SRAM ブロックのコンフィグレーションとなります。

図 -2: インテル Stratix デバイスの LAB および MLAB 構造

MLAB ALMをレギュラー LAB ALMとして使用 するか、デュアル ポート SRAMとして コンフィグレーション できます。	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
MLAB ALMをレギュラー LAB ALMとして使用 するか、デュアル ポート SRAMとして コンフィグレーション できます。	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
MLAB		LAB

### 3.1.2 ローカル・インターコネクトおよびダイレクトリンク・インターコネクト

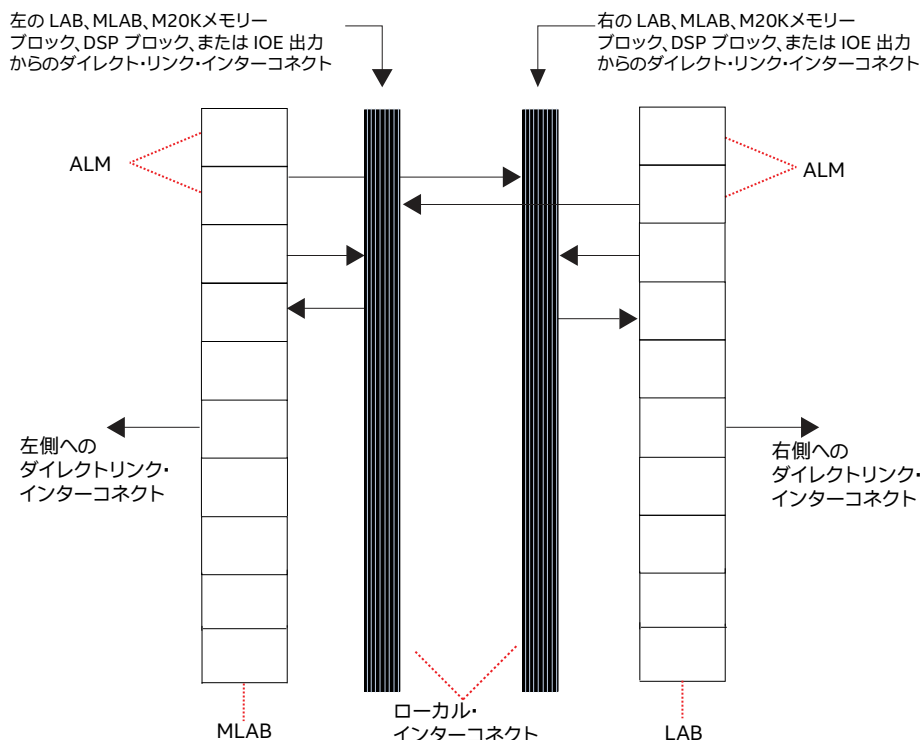
各 LAB は 40 ALM 出力をドライブアウトすることができます。2 つのグループの 20 ALM 出力は、ダイレクトリンク・インターコネクトを介して隣接する LAB を直接駆動することができます。

このダイレクトリンクによる接続機能により、ロウおよびカラム・インターコネクトの使用が最小限に抑えられ、さらに高い性能と柔軟性が提供されます。



ローカル・インターコネクトは ALM 入力を駆動します。ALM 出力と同様に、カラムとロウのインターコネクトもローカル・インターコネクトを駆動します。隣接する LAB、MLAB、M20K ブロック、あるいは左右のデジタル信号処理 (DSP) ブロックも、ダイレクトリンク接続を使用して LAB のローカル・インターコネクトを駆動することができます。

図 -3: インテル Stratix デバイスの LAB ローカル・インターコネクトおよびダイレクトリンク・インターコネクト

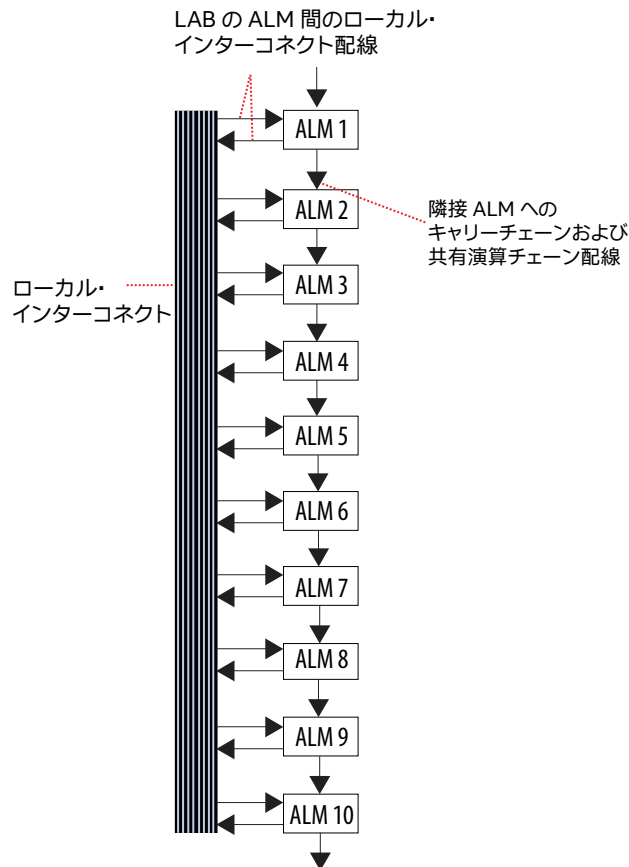


### 3.1.3 キャリーチェーンのインターコネクト

ALM 間には専用のキャリーチェーン・パスが存在します。インテル Stratix デバイスは、演算機能が効率的となるようキャリーチェーンを配線するために、LAB 内部に拡張されたインターコネクト構造を有しています。こうした ALM 間の接続はローカル・インターコネクトをバイパスします。

インテル Hyperflex レジスタはキャリーチェーンに追加され、一連の LAB 間で柔軟なリタイミングが可能になり、インテル Quartus Prime コンパイラはこれらのリソースを自動的に利用して使用率とパフォーマンスを向上させます。

図 -4: キャリーチェーンのインターコネクト



### 3.1.4 LAB コントロール信号

各 LAB は、LAB 内部の ALM レジスターを駆動するシングルクロックをサポートします。LAB は、ALM レジスター用の 2 つの固有のクロックイネーブル信号および追加のクリアー信号をサポートしています。

さらに、それぞれの LAB コントロール・ブロックは、ローカル・インターコネクトの Hyper-Register 用のクロック信号、および ALM 入力に配置された Hyper-Register 用の追加のクロックを駆動します。

LAB ロウ・クロック [5..0] と LAB ローカル・インターコネクトは LAB ワイドのコントロール信号を生成します。低スキュー・クロック・ネットワークは、ロウ・クロック [5..0] ヘグローバル信号を分配します。MultiTrack インターコネクトは、配線の効率化を図るため、長さと速度が異なる最適性能の連続配線ラインで構成されます。

#### 3.1.4.1 ロジック・コントロールのクリアー

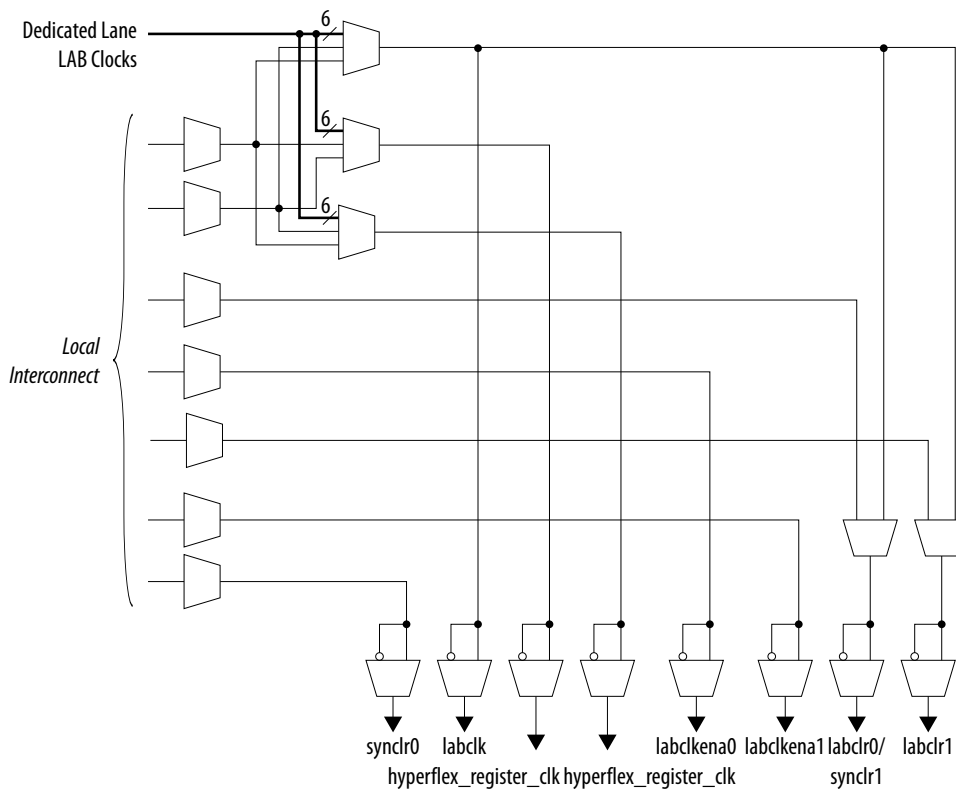
レジスターのクリアー信号のロジックは、LAB ワイド信号によって制御されます。ALM は同期クリアーおよび非同期クリアーの両方を直接サポートします。各 LAB は、クリアー信号の合計が 3 個を超えない限り、最大 2 つの同期クリアー信号と 2 つの非同期クリアー信号をサポートします。





インテル Stratix デバイスは、デバイス内のすべてのレジスターをリセットするデバイスワイドのリセットピン (DEV\_CLRn) を提供します。DEV\_CLRn ピンはコンパイルの前に インテル Quartus Prime ソフトウェアでイネーブリングすることができます。このデバイスワイドのリセット信号は、他のすべてのコントロール信号よりも優先されます。

図 -5: インテル Stratix デバイスの LAB ワイド・コントロール信号



## 3.2 ALM

次の項では、ALM リソース、ALM 出力、および ALM 動作モードについて説明します。

### 3.2.1 ALM リソース

各 ALM には、2 つの組み合わせアダプティブ LUT (ALUT)、2 ビットの全加算器、および 4 つのレジスター間で分割できる多様な LUT ベースのリソースが含まれています。

2 つの組み合わせ ALUT の最大 8 入力を使用して、1 つの ALM で 2 つの機能のさまざまな組み合わせを実装することができます。この適応性により、ALM は 4 入力 LUT アーキテクチャーとの完全な下位互換性を有します。1 つの ALM で、8 入力関数のサブセットを実装することも可能です。

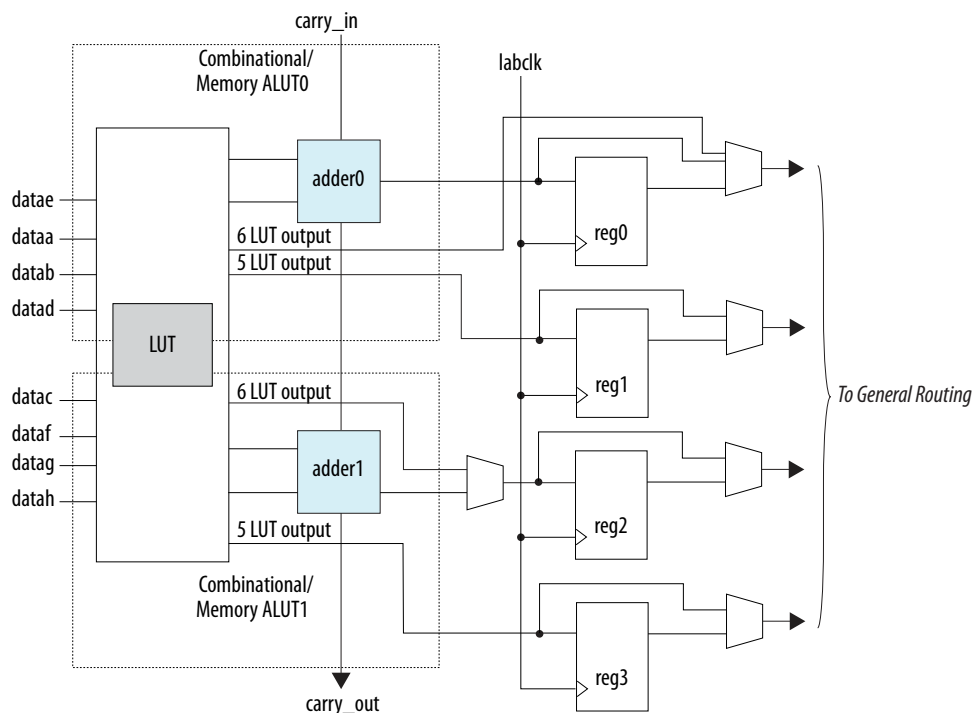
1 つの ALM には 4 つのプログラマブル・レジスターが含まれます。各レジスターそれぞれ、次のポートを有します。

- データ入力
- データ出力
- クロック
- クロックイネーブル
- 同期クリアー
- 非同期クリアー

グローバル信号、汎用 I/O (GPIO) ピン、あるいは任意の内部ロジックは、ALM レジスターのクロックイネーブル信号、クロック信号、および非同期クリアー・コントロール信号または同期クリアー・コントロール信号を駆動することができます。クロックイネーブル信号は同期リセット信号よりも優先されます。

組み合わせ機能では、レジスターがバイパスされ、LUT ( ルックアップ・テーブル ) と加算器の出力が ALM の出力に直接駆動します。

図 -6: インテル Stratix デバイスの ALM 上位レベルのブロック図



### 3.2.2 ALM 出力

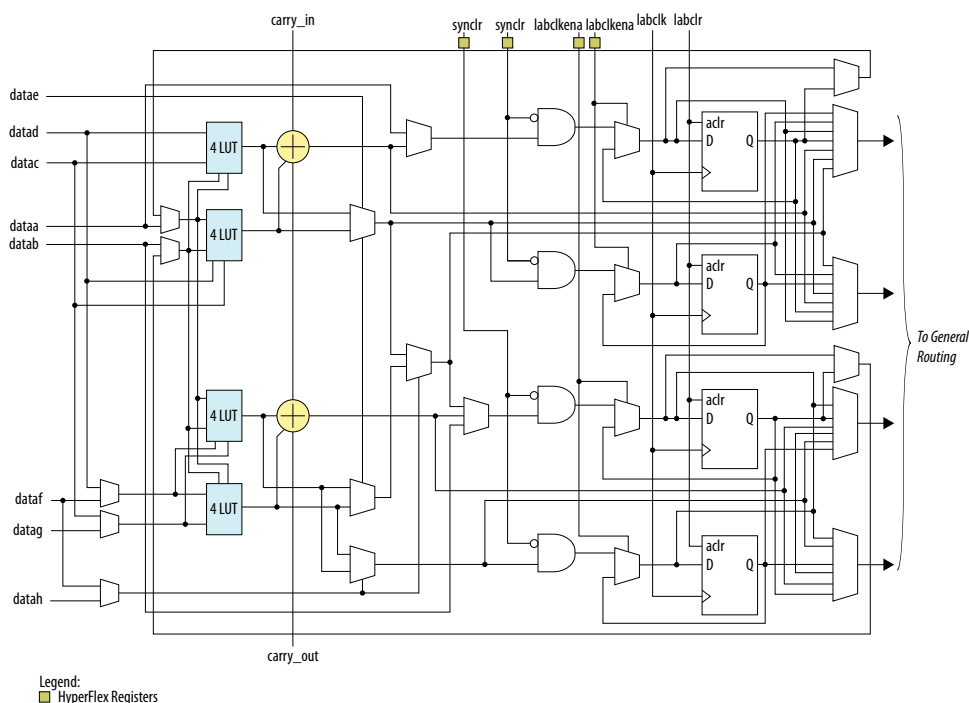
各 ALM における一般配線出力は、ローカル、ロウ、およびカラム配線リソースを駆動します。4 つの ALM 出力はカラム、ロウ、あるいはダイレクトリンク配線接続を駆動することができます。

LUT、加算器、またはレジスター出力は ALM 出力を駆動することができます。LUT または加算器と ALM レジスターは同時に ALM からドライブアウトすることができます。

レジスタパッキングは、まったく別のレジスタと組み合わせロジックを 1 つの ALM 内にパッキングすることでデバイスの稼働率を向上させます。フィッティングを向上させるための別のメカニズムは、レジスタ出力を同じ ALM の LUT 内にフィードバックできるようにし、レジスタが独自のファンアウト LUT でパッキングされるようにすることです。ALM は、LUT または加算器の出力のラッチされたバージョンとラッチされていないバージョンをドライブアウトすることもできます。

次の図は インテル Stratix 10 デバイスの ALM 接続を示しています。インテル Quartus Prime の Resource Property Editor では、ALM 接続全体を簡素化できます。配線の一部は インテル Quartus Prime ソフトウェアによって内部的に配線されます。

図 -7: インテル Stratix デバイスの ALM 接続の詳細



### 3.2.3 ALM 動作モード

インテル Stratix の ALM は、以下の 3 つのモードで動作します。

- ノーマルモード
- 拡張 LUT モード
- 演算モード

#### 3.2.3.1 ノーマルモード

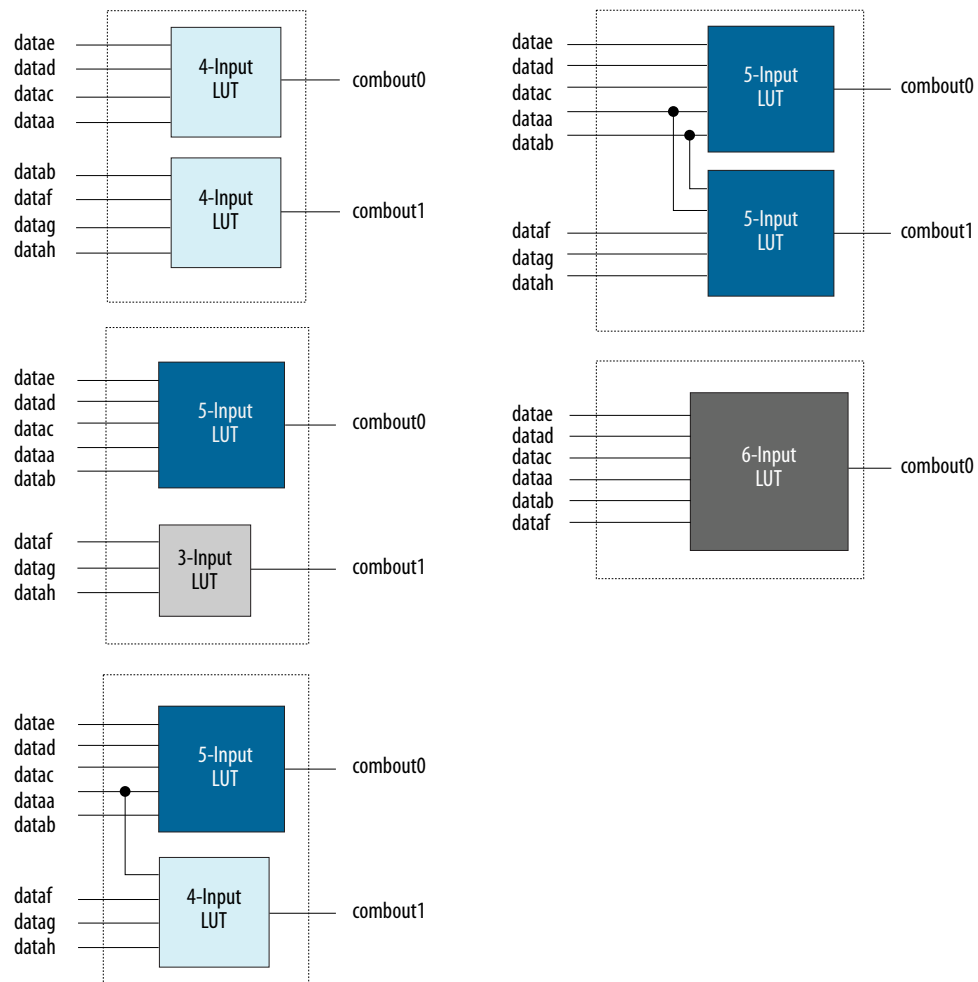
ノーマルモードでは、2 つのファンクションまたは最大 6 入力の 1 つのファンクションを 1 つの インテル Stratix デバイスの ALM に実装することができます。

LAB ローカル・インターコネクトからの最大 8 データ入力は、組み合わせロジックの入力になります。

ALM は、完全に独立したファンクションの特定の組み合わせおよび共通の入力を持つファンクションの様々な組み合わせをサポートできます。インテル Quartus Prime コンパイラーは LUT への入力を自動で選択します。ノーマルモードの ALM はレジスターパッキングをサポートします。

次の図は LUT モードでの異なる入力接続の組み合わせを示しています。デザインでは、インテル Quartus Prime ソフトウェアはコンパイル中に異なる入力名に割り当てる場合があります。

図 -8: ノーマルモードの ALM



なお、ここで示すものよりも入力数が少ないファンクションの組み合わせもサポートされています。たとえば、次の入力数を持つファンクションの組み合わせがサポートされます。

- 4 と 3
- 3 と 3
- 3 と 2
- 5 と 2



2 つの 5 入力ファンクションを 1 つの ALM にパッキングする際、これらのファンクションは少なくとも 2 つの共通入力を有している必要があります。共通入力は dataa と datab です。4 入力ファンクションと 5 入力ファンクションの組み合わせでは、1 つの共通入力 (dataa または datab のいずれか) が必要です。

使用頻度の低いデバイスでは、インテル Quartus Prime ソフトウェアを使用して 1 つの ALM に配置できるファンクションを別の ALM に実装することにより、最高のパフォーマンスを実現することができます。デバイスの使用率が高くなり始めると、インテル Quartus Prime ソフトウェアは自動的にインテル Stratix の ALM を最大限に活用します。インテル Quartus Prime のコンパイラは共通入力を使用するファンクションまたは完全に独立したファンクションを自動的に検索します。これらのファンクションを 1 つの ALM に配置することにより、デバイスのリソースを効率的に使用します。なお、ロケーション・アサインメントを設定することによってリソース使用量を手動でコントロールすることもできます。

図 -9: ノーマルモードの 6 入力 LUT モード・ファンクション

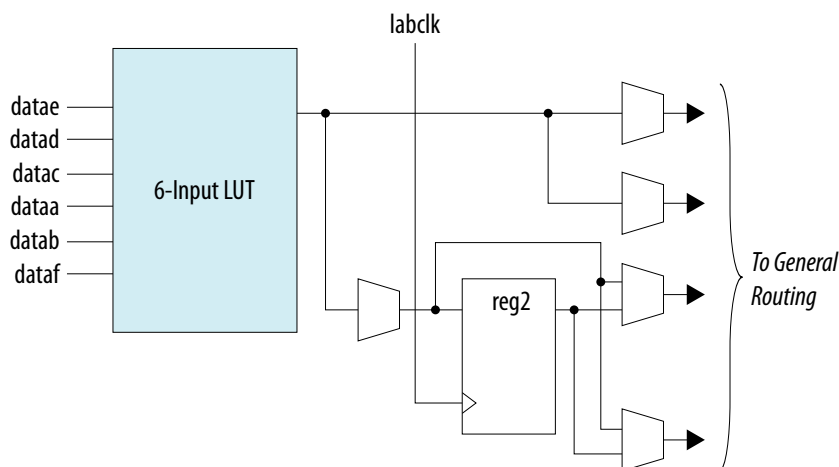
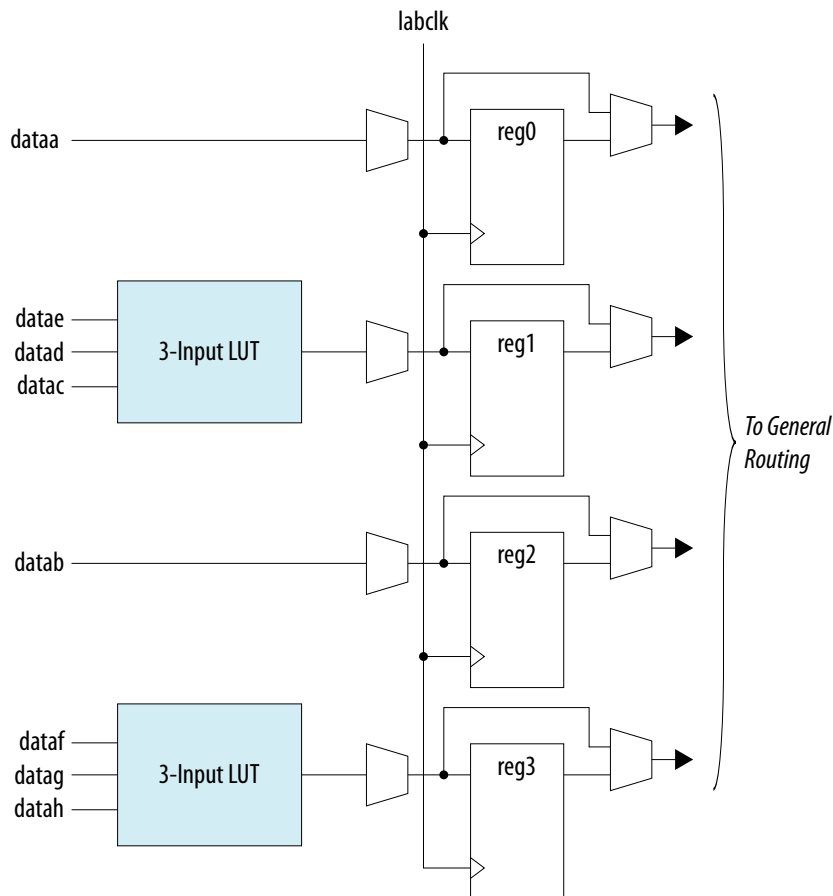


図 -10: ノーマルモードの 3 入力 LUT モード・ファンクション

レジスタパッキングには, dataa と datab が使用可能です。



注意: 3 アトム入力 LUT は、インテル Quartus Prime ソフトウェアがマップする物理 ALM データ入力に関係なく、dataa、datab および datac という名称がついています。

次の入力を使用して任意の 3~6 入力ファンクションを実装することができます。

- datae
- datad
- datac
- datag
- datah
- dataf
- dataa および datab—両方の LUT で dataa と datab が共有され、別の入力ファンクションを実装するための柔軟性を提供します。



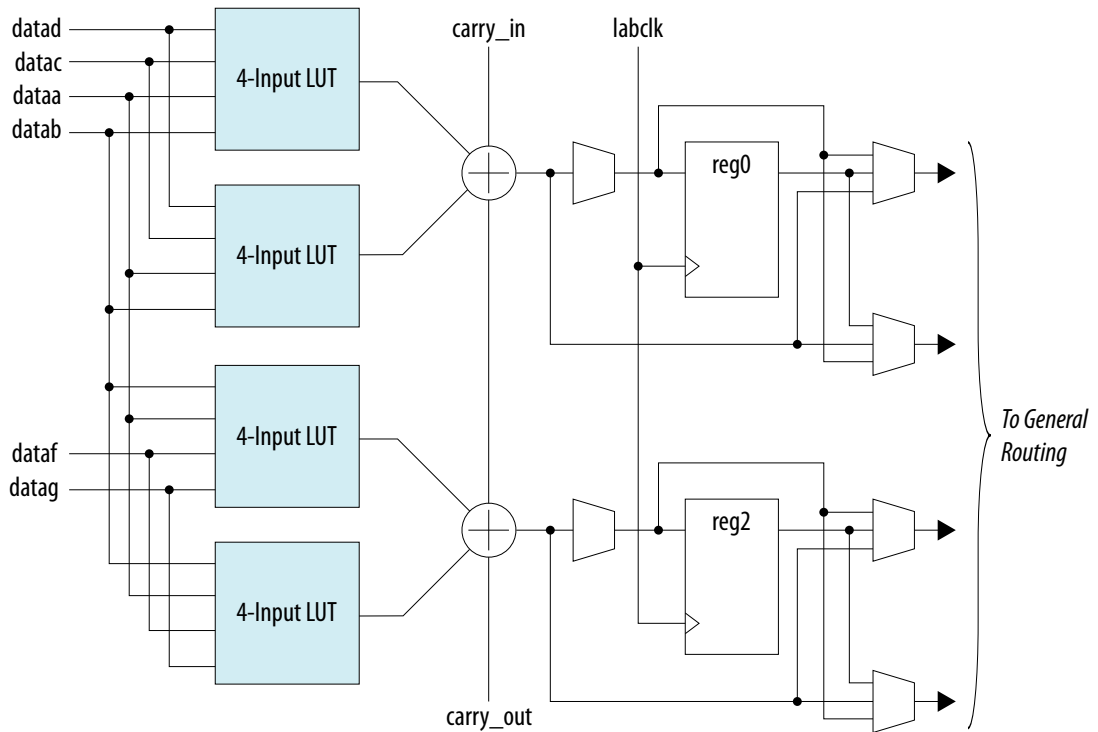
### 3.2.3.3 演算モード

演算モードの ALM は、2 個の専用全加算器と共に 2 組の 4 入力 LUT を使用します。この専用加算器を使用することで、LUT がプリアダーロジックの実行が可能となります。よって、各加算器は 2 個の 4 入力ファンクションの出力を追加することが可能です。

また、演算モードではクロックイネーブル、カウンタイネーブル、同期アップ / ダウン・コントロール、加算 / 減算コントロール、および同期ロードを提供します。

クリアおよびクロックイネーブル・オプションは、LAB 内のすべてのレジスターに影響する LAB ワイド信号です。これらの信号はレジスターごとに個別にディスエーブルまたはイネーブルすることができます。インテル Quartus Prime ソフトウェアは、カウンタによって使用されていないレジスターを自動的に他の LAB に配置します。

図 -12: インテル Stratix デバイスの演算モードの ALM



#### 3.2.3.3.1 キャリーチェーン

キャリーチェーンは、演算モードで専用加算器間的高速キャリー・ファンクションを提供します。

インテル Stratix デバイスの 2 ビット・キャリー選択機能は、ALM を使用してキャリーチェーンの伝播遅延を分割します。キャリーチェーンは、LAB 内の最初の ALM または 6 番目の ALM のいずれかで開始することができます。最後のキャリーアウト信号は ALM に接続され、そこでローカル、ロウ、カラムのいずれかのインターコネクต์に供給されます。

高ファンイン演算ファンクションが実装されている場合、デバイス内のある特定のエリアに配線が密集することを防止するため、LAB は次の LAB に接続する前に LAB の下半分のみを使用するキャリーチェーンをサポートすることができます。これにより、LAB 内の ALM の利用可能な上半分を使用し、ノーマルモードでより狭いファンイン・ファンクションを実装することが可能となります。最初の LAB 内の下部





に位置する 5 つの ALM を使用するキャリーチェーンは、カラム内で次の LAB の ALM の下半分に取り込みます。また、最初の LAB 内の下位 5 つの ALM を使用するキャリーチェーンは、カラム内で次の LAB の ALM の下半分に取り込みます。なお、LAB カラムと MLAB カラムの上半分は、バイパスすることが可能です。

インテル Quartus Prime のコンパイラーは、複数の LAB をリンクさせることにより、20 個以上の ALUT ( 演算モードでは 10 個の ALM) で構成される長いキャリーチェーンを作成します。フィッティングが向上されるよう、長いキャリーチェーンは垂直に実行されるため、TriMatrix メモリーおよび DSP ブロックへの水平方向の接続を高速化することができます。



## 4 インテル Stratix デバイスの LAB および ALM ユーザーガイドの改訂履歴

日付	バージョン	変更内容
2017 年 11 月	2017.11.06	<ul style="list-style-type: none"> <li>ノーマルモードの 6 入力 LUT モード・ファンクション</li> <li>項「インテル Stratix デバイスの LAB および ALM の概要」を更新。</li> <li>項「キャリアチェーン」の項を更新。</li> <li>インテル Stratix デバイスの ALM 上位レベルのブロック図を更新。</li> <li>インテル Stratix 10 デバイスの ALM 接続の詳細の図の Hyperflex レジスターを更新。</li> </ul>
2016 年 10 月	2016.10.31	初版

Intel Corporation. 無断での引用、転載を禁じます。Intel、インテル、Intel ロゴ、Altera、ARRIA、CYCLONE、ENPIRION、MAX、NIOS、QUARTUS および STRATIX の名称およびロゴは、アメリカ合衆国および/またはその他の国における Intel Corporation の商標です。インテルは FPGA 製品および半導体製品の性能がインテルの標準保証に準拠することを保証しますが、インテル製品およびサービスは、予告なく変更される場合があります。インテルが書面にて明示的に同意する場合を除き、インテルはここに記載されたアプリケーション、または、いかなる情報、製品、またはサービスの使用によって生じるいっさいの責任を負いません。インテル製品の顧客は、製品またはサービスを購入する前、および、公開済みの情報を信頼する前には、デバイスの仕様を最新のバージョンにしておくことをお勧めします。

\*その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

ISO  
9001:2008  
登録済