

# MAX 10 FPGA デバイスのアーキテクチャ

2016.08.11

M10-ARCHITECTURE



更新情報



フィードバック

MAX<sup>®</sup> 10 デバイスは、以下の要素で構成されています。

- ロジック・アレイ・ブロック (LAB)
- アナログ-デジタル・コンバータ (ADC)
- ユーザー・フラッシュ・メモリ (UFM)
- エンベデッド乗算器ブロック
- エンベデッド・メモリ・ブロック (M9K)
- クロックおよびフェーズ・ロック・ループ (PLL)
- 汎用 I/O
- 高速 LVDS I/O
- 外部メモリ・インタフェース
- コンフィギュレーション・フラッシュ・メモリ (CFM)

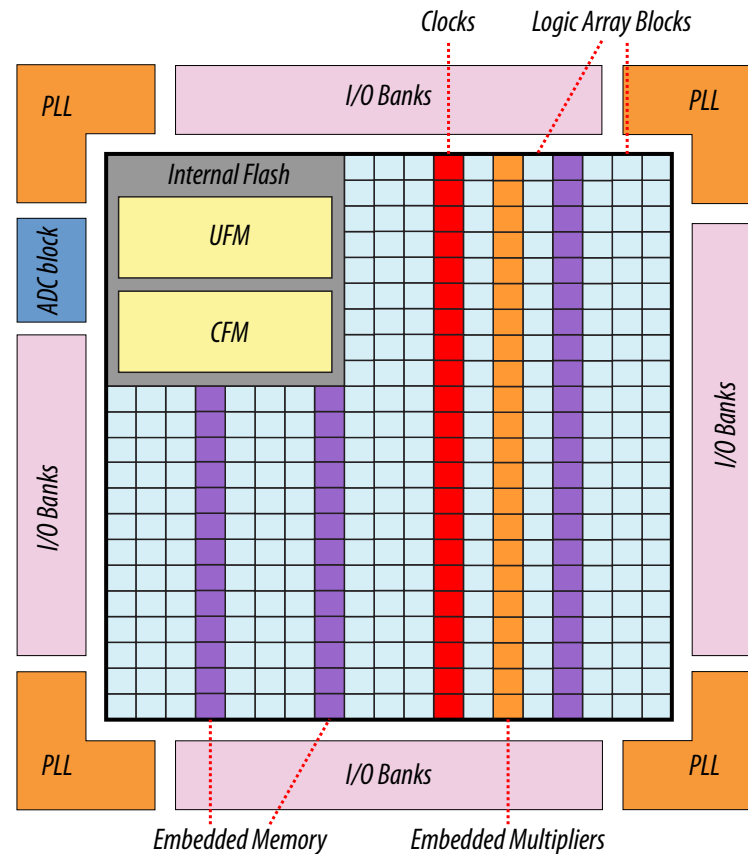
© 2016 Intel Corporation. All rights reserved. Intel, the Intel logo, Altera, Arria, Cyclone, Enpirion, MAX, Megacore, NIOS, Quartus and Stratix words and logos are trademarks of Intel Corporation in the US and/or other countries. Other marks and brands may be claimed as the property of others. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel 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 Intel. Intel 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  
登録済

**ALTERA**  
now part of Intel

図 1: MAX 10 デバイスの一般的なデバイス・フロアプラン

- 各ブロックの数量と場所は、MAX 10 デバイスごとに異なります。
- 一部の MAX 10 デバイスには特定のブロックが含まれない場合もあります。



#### 関連情報

- [MAX 10 デバイス・データシート](#)  
MAX 10 デバイスの仕様と性能についての詳細な情報を提供します。
- [MAX 10 FPGA デバイスの概要](#)  
MAX 10 デバイスの最大リソースについての詳細な情報を提供します。

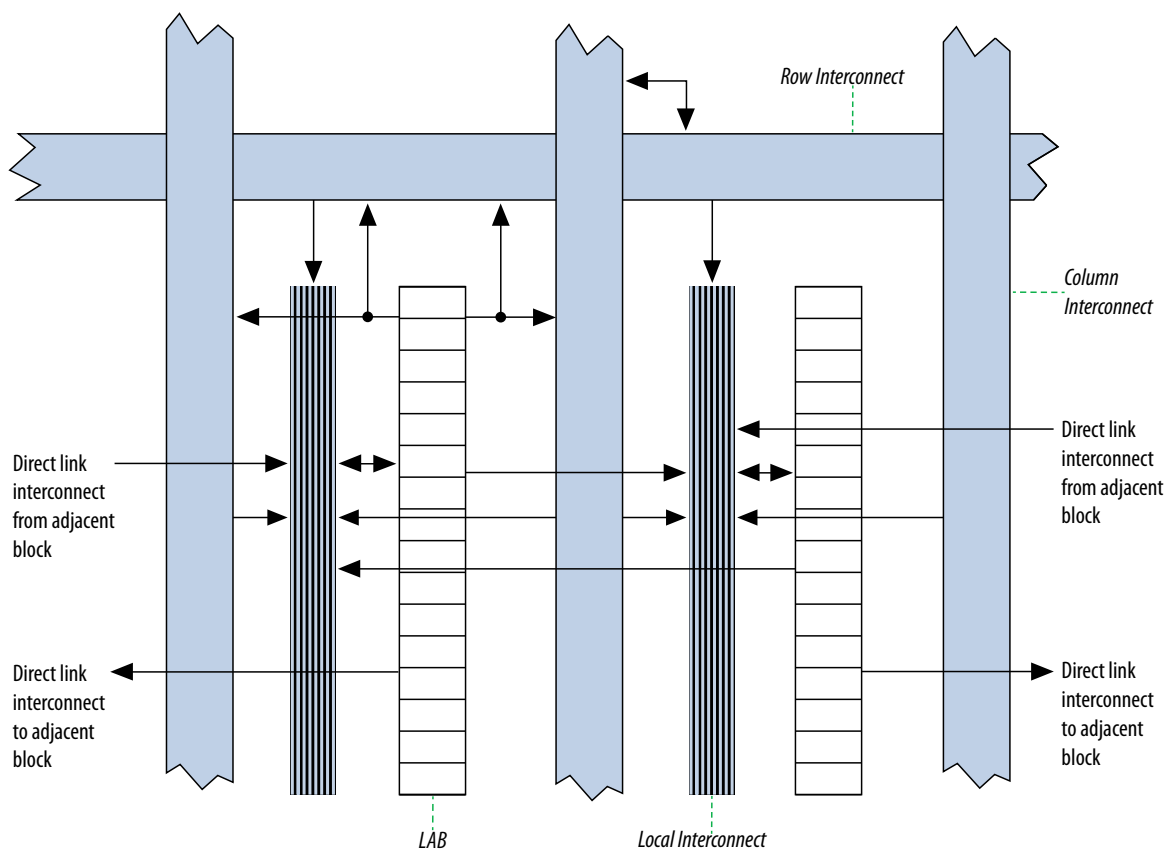
## ロジック・アレイ・ブロック

LAB は、ロジック・リソースのグループで構成されるコンフィギュレーション可能なロジック・ブロックです。

各 LAB は、以下の要素で構成されています。

- 16 個のロジック・エレメント (LE)— MAX 10 デバイスで最小ロジック・ユニット
- LE キャリー・チェーン—LAB 内の各 LE を通してシリアルに伝播されるキャリー・チェーン
- LAB コントロール信号—LAB 内の LE にコントロール信号を駆動するための専用ロジック
- ローカル・インタコネクト—同じ LAB 内の LE 間で信号を転送します。
- レジスタ・チェーン—LAB 内で、ある LE レジスタの出力を隣接する LE レジスタに転送します。

図 2: MAX 10 デバイスの LAB 構造



Quartus® Prime のコンパイラは関連するロジックを 1 つの LAB または隣接する複数の LAB に保存するため、ローカルおよびレジスタ・チェーン接続を使用して、性能と面積効率を確保することができます。

## LAB インタコネクト

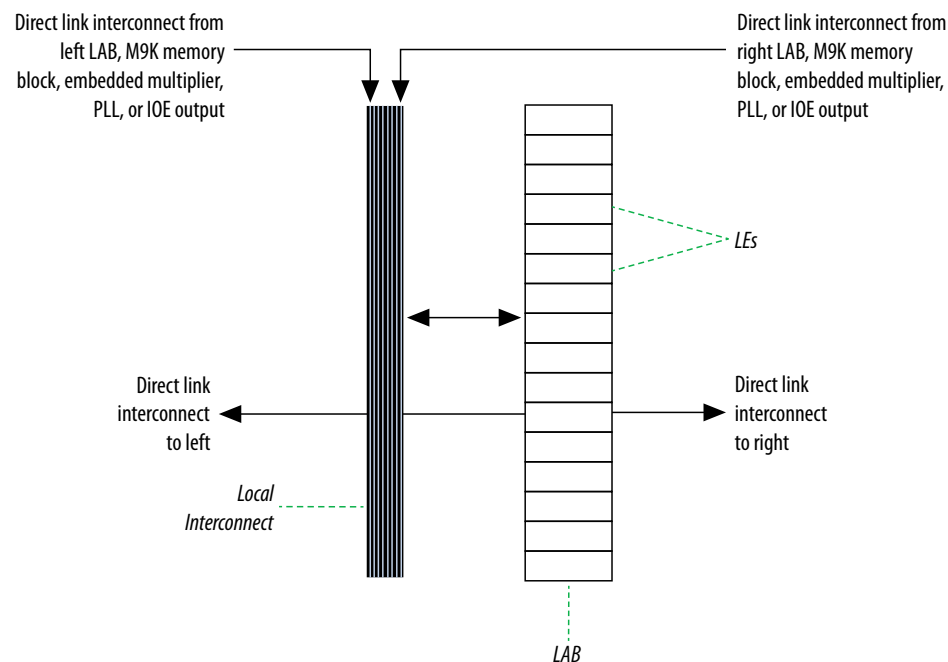
LAB ローカル・インタコネクトは、同じ LAB のカラムおよびロウ・インタコネクトと LE 出力によって駆動されます。

ダイレクト・リンク接続は、ロウおよびカラム・インタコネクトの使用を最小限に抑え、性能と柔軟性を向上させます。ダイレクト・リンク接続により、左右の隣接エレメントが、LAB のローカル・インタコネクトを駆動できます。この説明に該当するエレメントは以下のとおりです。

- LAB
- PLL
- M9K エンベデッド・メモリ・ブロック
- エンベデッド乗算器

各 LE は、ローカルおよびダイレクト・リンク・インタコネクトにより最高 48 個の LE を駆動できます。

図 3: MAX 10 デバイスの LAB ローカルおよびダイレクト・リンク・インタコネクト



## LAB コントロール信号

各 LAB には、LE 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ています。

コントロール信号には、次のものがあります。

- 2つのクロック信号
- 2つのクロック・イネーブル信号
- 2つの非同期クリア信号
- 1つの同期クリア信号
- 1つの同期負荷信号

図 4: MAX 10 デバイスの LAB ワイドのコントロール信号

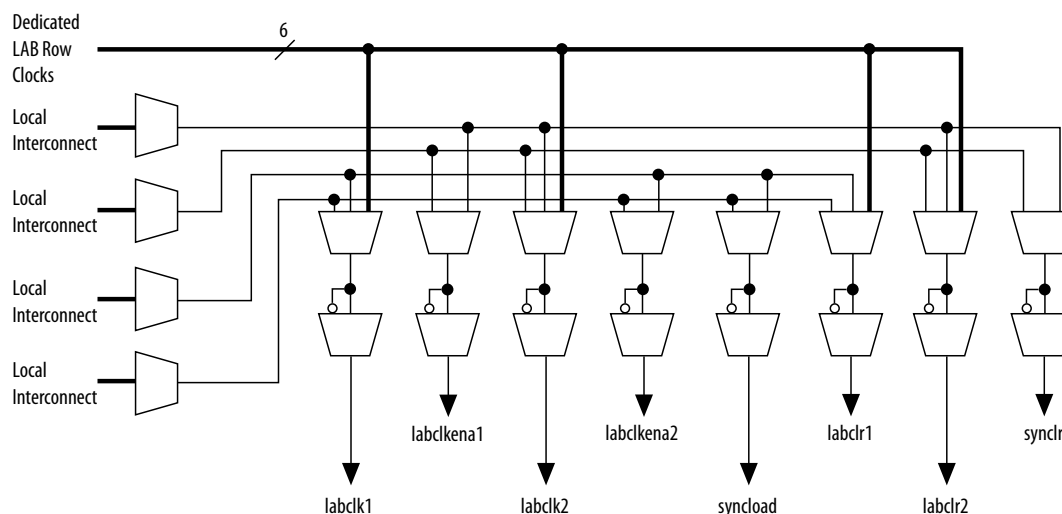


表 1: MAX 10 デバイスのコントロール信号の説明

コントロール信号	説明
labclk1 labclk2	<ul style="list-style-type: none"><li>各 LAB は 2 つのクロック信号を使用できます。各 LAB のクロック信号とクロック・イネーブル信号は、リンクされます。たとえば、labclk1 信号を使用する特定の LAB 内に含まれるすべての LE は、labclkena1 信号も使用します。</li><li>LAB がクロックの立ち上がりエッジと立ち下がりエッジの両方を使用する場合は、両方の LAB ワイドのクロック信号も使用します。</li><li>LAB ロウ・クロック [5..0] と LAB ローカル・インタコネクトは、LAB ワイドのクロック信号を生成します。MultiTrack インタコネクト固有の低スキューにより、データ分配に加えクロックとコントロール信号も分配できます。</li></ul>
labclkena1 labclkena2	<ul style="list-style-type: none"><li>各 LAB は 2 つのクロック・イネーブル信号を使用できます。各 LAB のクロック信号とクロック・イネーブル信号は、リンクされます。たとえば、labclk1 信号を使用する特定の LAB 内に含まれるすべての LE は、labclkena1 信号も使用します。</li><li>クロック・イネーブル信号をデアサートすると、LAB ワイドのクロック信号はオフになります。</li></ul>
labclr1 labclr2	非同期クリア信号: <ul style="list-style-type: none"><li>レジスタのクリア信号のロジックをコントロールする LAB ワイドのコントロール信号。LE は、非同期クリア・ファンクションを直接サポートします。</li><li></li></ul>

コントロール信号	説明
syncload	同期負荷および同期クリア信号: <ul style="list-style-type: none"> <li>カウンタなどのファンクションを実装するために使用できます。</li> <li>LAB 内のすべてのレジスタに影響を及ぼす LAB ワイドのコントロール信号</li> </ul>
syncclr	

コントロール信号は、一度に 8 つまで使用できます。レジスタ・パッキングと同期負荷を同時に使用することはできません。

各 LAB には非グローバル・コントロール信号を 4 つまで含めることができます。追加の LAB コントロール信号は、グローバル信号である限り使用できます。

レジスタのプリセット信号のロジックをコントロールするための LAB ワイドの非同期負荷信号は、提供されていません。レジスタは、NOT ゲート・プッシュバック・テクニックでプリセットできます。MAX 10 デバイスは、プリセットまたは非同期クリア信号のいずれかのみをサポートします。

MAX 10 デバイスは、クリア・ポートに加え、デバイス内のすべてのレジスタをリセットするチップ・ワイドのリセット・ピン (DEV\_CLRn) も提供しています。このピンは、Quartus Prime ソフトウェアでコンパイル前に設定したオプションにより、コントロールされます。このチップ・ワイドのリセットにより、他のすべてのコントロール信号が無効になります。

## ロジック・エレメント

LE は、MAX 10 デバイス・ファミリのアーキテクチャの最小ロジック・ユニットです。LE はコンパクトで、ロジックを効率的に使用する高度な機能を提供します。

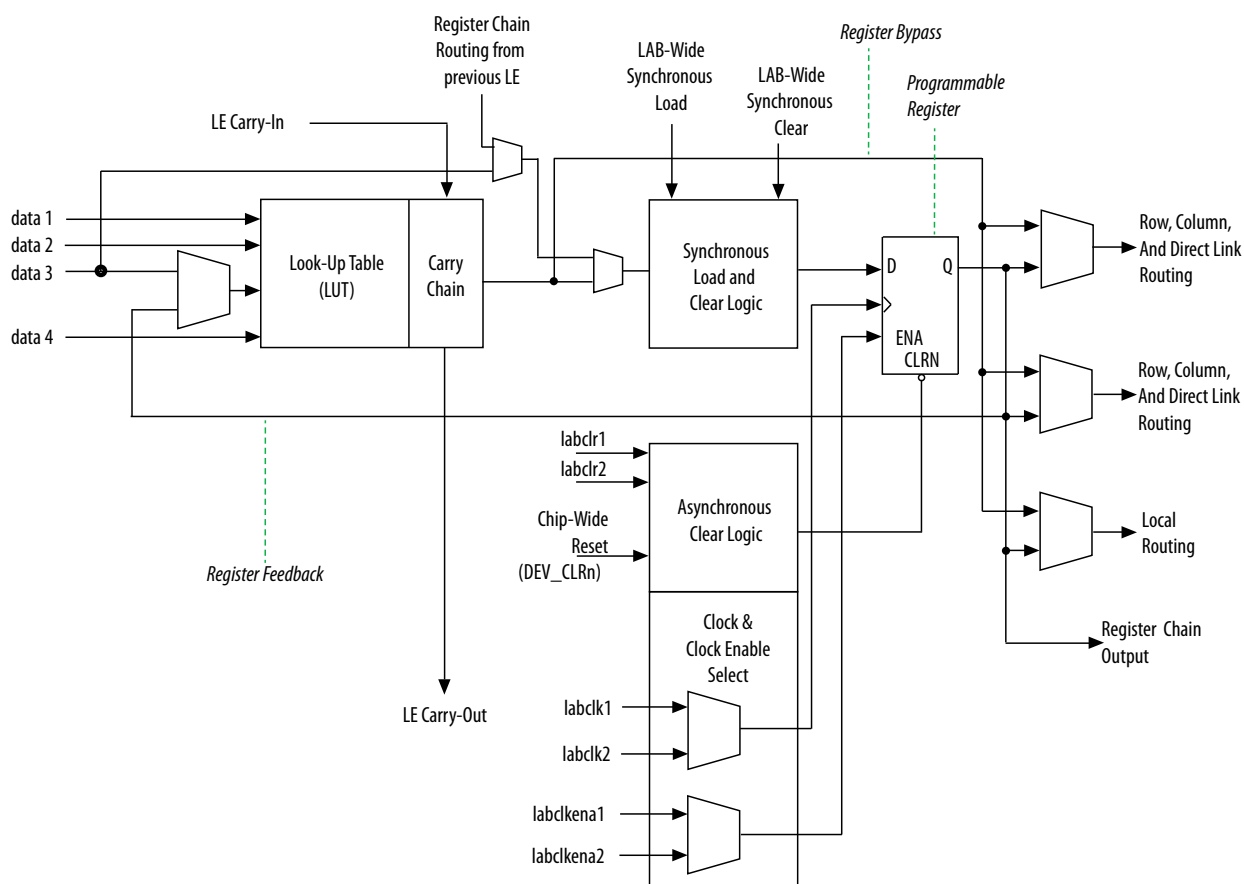
各 LE には以下の機能があります。

- 4 つの変数を使用する任意の機能を実装できる 4 入力ルック・アップ・テーブル (LUT)
- プログラマブル・レジスタ
- キャリー・チェイン接続
- レジスタ・チェイン接続
- 以下のインタコネクトを駆動する機能
  - ローカル
  - ロウ
  - カラム
  - レジスタ・チェイン
  - ダイレクト・リンク
- レジスタ・パッキングのサポート
- レジスタ・フィードバックのサポート

### LE の機能

LE には、複数の機能を可能にする入力、出力、およびレジスタが含まれています。

図 5: MAX 10 デバイスの LE ハイレベル・ブロック図。



## LE の入力

LE の動作モードであるノーマル・モードと演算モードの両方のモードでは、LE への 6 つの入力を利用することができます。各入力は、必要なロジック・ファンクションを実装するため、異なる目的地に向けられます。以下のものが LE 入力となります。

- LAB ローカル・インタコネクトからの 4 つのデータ入力
- 前の LE キャリー・チェーンからの LE キャリー・イン
- レジスタ・チェーン接続

## LE の出力

各 LE には、以下の 3 つの出力があります。

- 2 つの LE 出力は、カラムまたはロウとダイレクト・リンク・ルーティング接続を駆動します。
- 1 つの LE 出力は、ローカル・インタコネクト・リソースを駆動します。

MAX 10 デバイスでは、レジスタ・パッキング機能がサポートされています。レジスタ・パッキングにより、LUT またはレジスタ出力は、3 つの出力を個別に駆動します。この機能により、関

連性のないファンクションに対してレジスタと LUT を使用し、デバイスの使用効率を改善できます。

レジスタ・パッキングを使用する場合は、LAB ワイドの同期負荷コントロール信号を使用することはできません。

### レジスタ・チェイン出力

各 LE には、同じ LAB のレジスタをカスケード接続できるレジスタ・チェイン出力があります。この機能により LAB 間の接続が高速化されます。また、以下の動作を実現することにより、ローカル・インタコネクト・リソースを最適化できます。

- 組み合わせファンクションに LUT を使用
- 関連性のないシフト・レジスタ実装にレジスタを使用

### プログラマブル・レジスタ

各 LE のプログラマブル・レジスタをコンフィギュレーションすると、D、T、JK、または SR フリップフロップ動作を実現できます。各レジスタには以下の入力があります。

- クロック—グローバル・クロック・ネットワークを使用する信号、汎用 I/O ピン、または内部ロジックによって駆動できます。
- クリア—グローバル・クロック・ネットワークを使用する信号、汎用 I/O ピン、または内部ロジックによって駆動できます。
- クロック・イネーブル—汎用 I/O ピンまたは内部ロジックによって駆動できます。

組み合わせファンクションの場合、LUT 出力はレジスタをバイパスし、LE 出力に直接駆動されます。

### レジスタ・フィードバック

レジスタ・フィードバック・モードでは、同じ LE の LUT にレジスタ出力をフィードバックすることができます。これにより、レジスタは、フィッティングを改善する別のメカニズムを提供する独自のファンアウト LUT で確実にパッキングされます。LE は、LUT 出力の登録バージョンと非登録バージョンをドライブ・アウトすることもできます。

### LE 動作モード

MAX 10 デバイスの LE は、2 つのモードで動作します。

- ノーマル・モード
- 演算モード

これらの動作モードでは、LE リソースが異なる形で使用されます。どちらの LE モードにも、利用可能な 6 つの入力と LAB ワイドの信号があります。

Quartus Prime ソフトウェアは、パラメータ化されたモジュールのライブラリ (LPM) ファンクションなどのパラメータ化されたファンクションと組み合わせて、カウンタ、加算器、減算器、演算ファンクションなどの一般的なファンクションに適したモードを自動的に選択します。

最適な性能を実現するために使用する LE 動作モードを指定する専用ファンクションも作成できます。

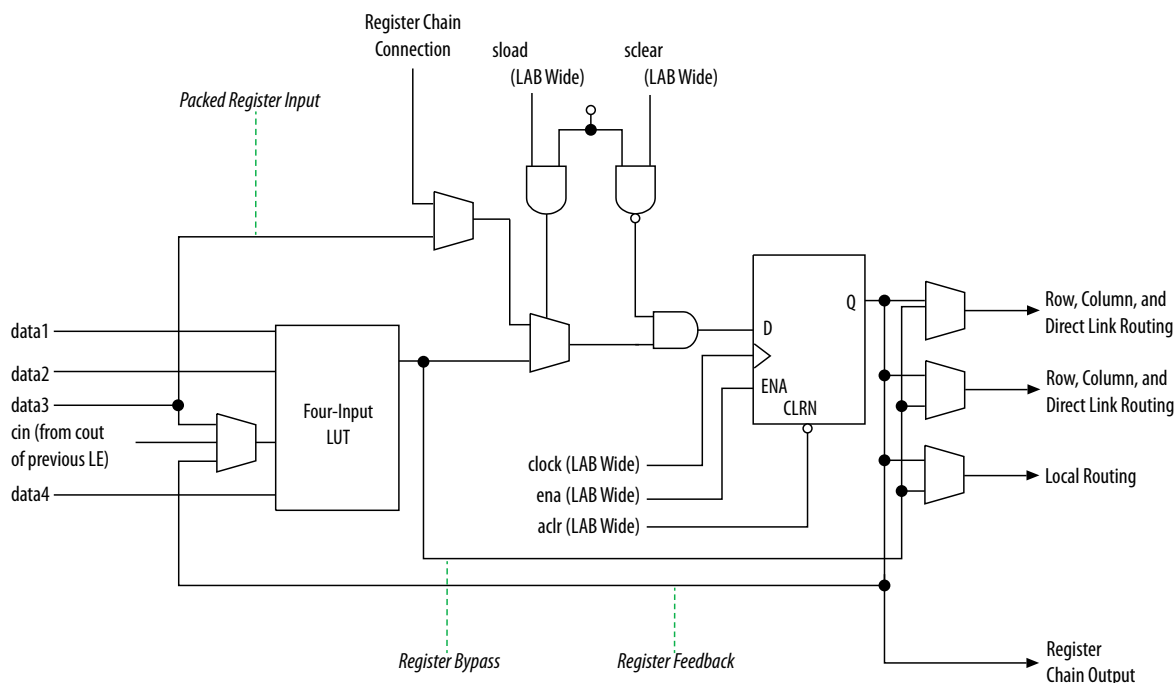


## ノーマル・モード

ノーマル・モードは、一般的なロジック・アプリケーションと組み合わせファンクションに適しています。

ノーマル・モードでは、LAB ローカル・インタコネクトからの4つのデータ入力が4入力 LUT に入力されます。Quartus Prime のコンパイラは、LUT への入力の1つとして、キャリー・イン (cin) または data3 信号を自動的に選択します。ノーマル・モードの LE は、パッキングされたレジスタとレジスタ・フィードバックをサポートします。

図 6: MAX 10 デバイスのノーマル・モードでの LE の動作

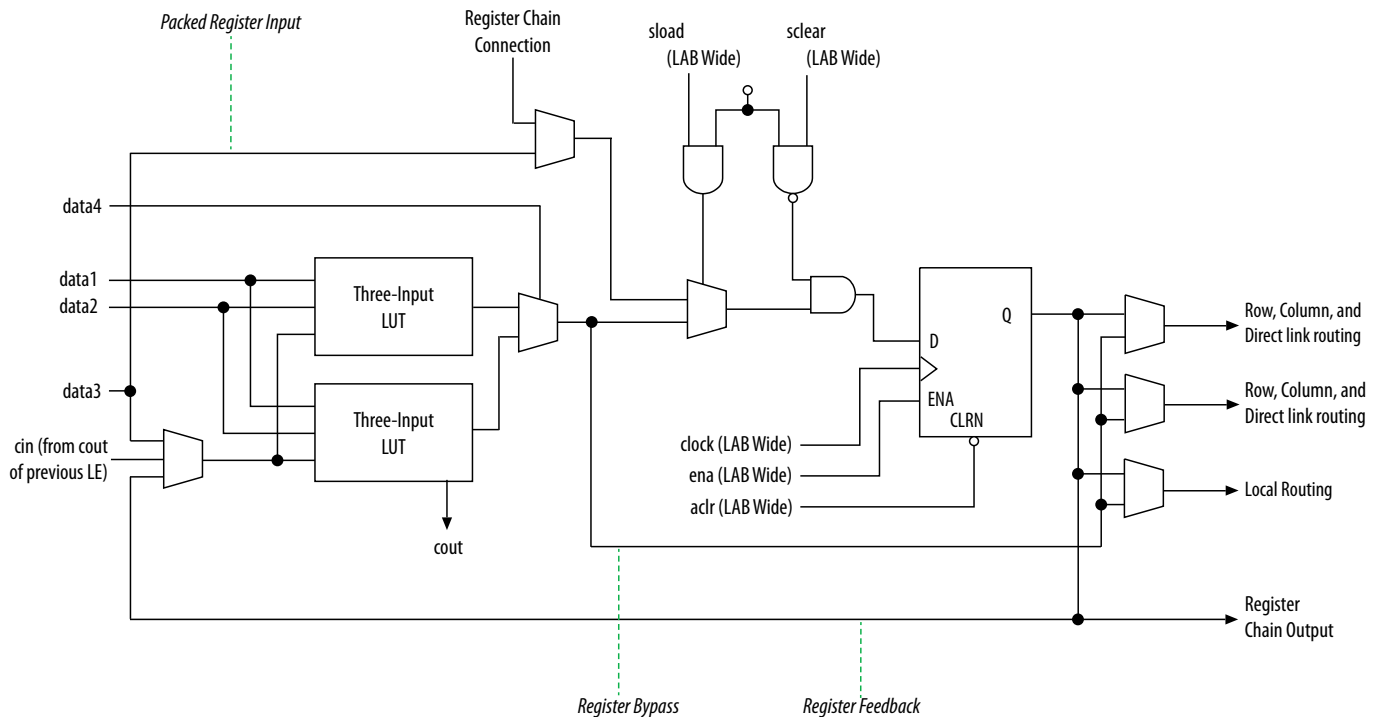


## 演算モード

演算モードは、加算器、カウンタ、アキュムレータ、およびコンパレータの実装に最適です。

演算モードの LE は、2 ビットのフル加算器と基本的なキャリー・チェインを実装します。演算モードの LE は、LUT 出力の登録バージョンと非登録バージョンをドライブ・アウトすることができます。LE を演算モードで使用する場合は、レジスタ・フィードバックとレジスタ・パッキングがサポートされます。

**図 7: MAX 10 デバイスの演算モードでの LE の動作**



## キャリア・チェーン

Quartus Prime のコンパイラは、デザイン処理中にキャリー・チェーン・ロジックを自動的に作成します。デザイン・エントリ中にキャリー・チェーン・ロジックを手動で作成することもできます。LPM ファンクションなどのパラメータ化されたファンクションは、適切なファンクションのキャリー・チェーンを自動的に利用します。Quartus Prime のコンパイラは、同じカラムの LAB を自動的にリンクすることによって、16 LE より長いキャリー・チェーンを作成します。

拡張フィッティングに対しては、長いキャリア・チェーンが垂直に実行されるので、ダイレクト・リンク・インタコネクトを通した M9K メモリ・ブロックまたはエンベデッド乗算器への高速水平接続が可能です。たとえば、M9K メモリ・ブロックのカラムの横にある LAB カラムに長いキャリア・チェーンが含まれるデザインの場合、すべての LE 出力がダイレクト・リンク・インタコネクトを通して隣接する M9K メモリ・ブロックにフィードすることができます。

キャリア・チェインが水平方向である場合、M9K メモリ・ブロックのカラムに隣接していない LAB は、他のロウまたはカラム・インタコネクトを使用して、M9K メモリ・ブロックを駆動します。

キャリア・チェインは、カラム全体にわたって続きます。

## エンベデッド・メモリ

MAX 10 エンベデッド・メモリ・ブロックは、高スループットの packets 処理、エンベデッド・プロセッサ・プログラム、エンベデッド・データ・ストレージなどのアプリケーション向けに最適化されています。

MAX 10 エンベデッド・メモリ構造は、9,216 ビット（パリティ・ビットを含む）のブロックで構成されています。RAM、ROM、シフト・レジスタ、FIFO などのさまざまなメモリ機能を提供するために、各 M9K ブロックは異なる幅およびコンフィギュレーションで使用することができます。

MAX 10 エンベデッド・メモリは、以下の一般的な機能をサポートします。

- ブロックあたり 8,192 メモリ・ビット（パリティ・ビットを含めた場合、ブロックあたり 9,216 ビット）
- 各ポートの独立した読み取りイネーブル（*rden*）信号および書き込みイネーブル（*wren*）信号
- M9K メモリ・ブロックが 2 つの 4.5 K シングル・ポート RAM に分割されたパック・モード
- 可変ポート・コンフィギュレーション
- すべてのポート幅におけるシングル・ポート・モードおよびシングル・デュアル・ポート・モードのサポート
- トゥルー・デュアル・ポート（1 回の読み取りと 1 回の書き込み、2 回の読み取り、あるいは 2 回の書き込み）動作
- 書き込み中のデータ入力マスキングのバイト・イネーブル
- 各ポート（ポート A とポート B）の 2 つのクロック・イネーブル・コントロール信号
- RAM および ROM モードでメモリの内容をプリロードするための初期化ファイル

#### 関連情報

[MAX 10 エンベデッド・メモリ・ユーザーガイド](#)

## エンベデッド乗算器

エンベデッド・マルチプライヤ・ブロックは、アプリケーションのニーズに応じて 2 つの動作モードのいずれかで使用することができます。

- 1 つの 18 ビット x 18 ビット・マルチプライヤ
- 最大 2 個の 9 ビット x 9 ビット独立マルチプライヤ

また、MAX 10 デバイスのエンベデッド・マルチプライヤを使用して、乗算器・加算器および乗算アキュムレータ機能を実装することができます。機能の乗算器部分は、エンベデッド・マルチプライヤを使用して実装されます。加算器またはアキュムレータ機能はロジック・エレメント (LE) に実装されます。

#### 関連情報

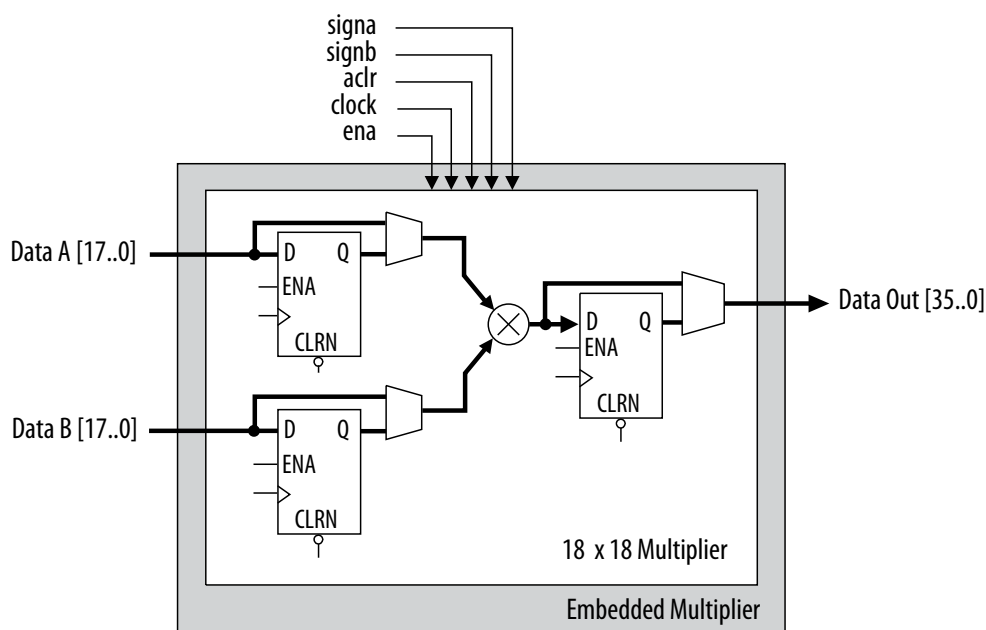
[MAX 10 エンベデッド・マルチプライヤ・ユーザーガイド](#)

## 18 ビット乗算器

各エンベデッド・マルチプライヤをコンフィギュレーションして、10~18 ビットの入力幅の 18 x 18 マルチプライヤをサポートすることができます。

次の図は、18 ビット・マルチプライヤをサポートするようコンフィギュレーションされたエンベデッド・マルチプライヤを示しています。

図 8: 18 ビット・マルチプライヤ・モード



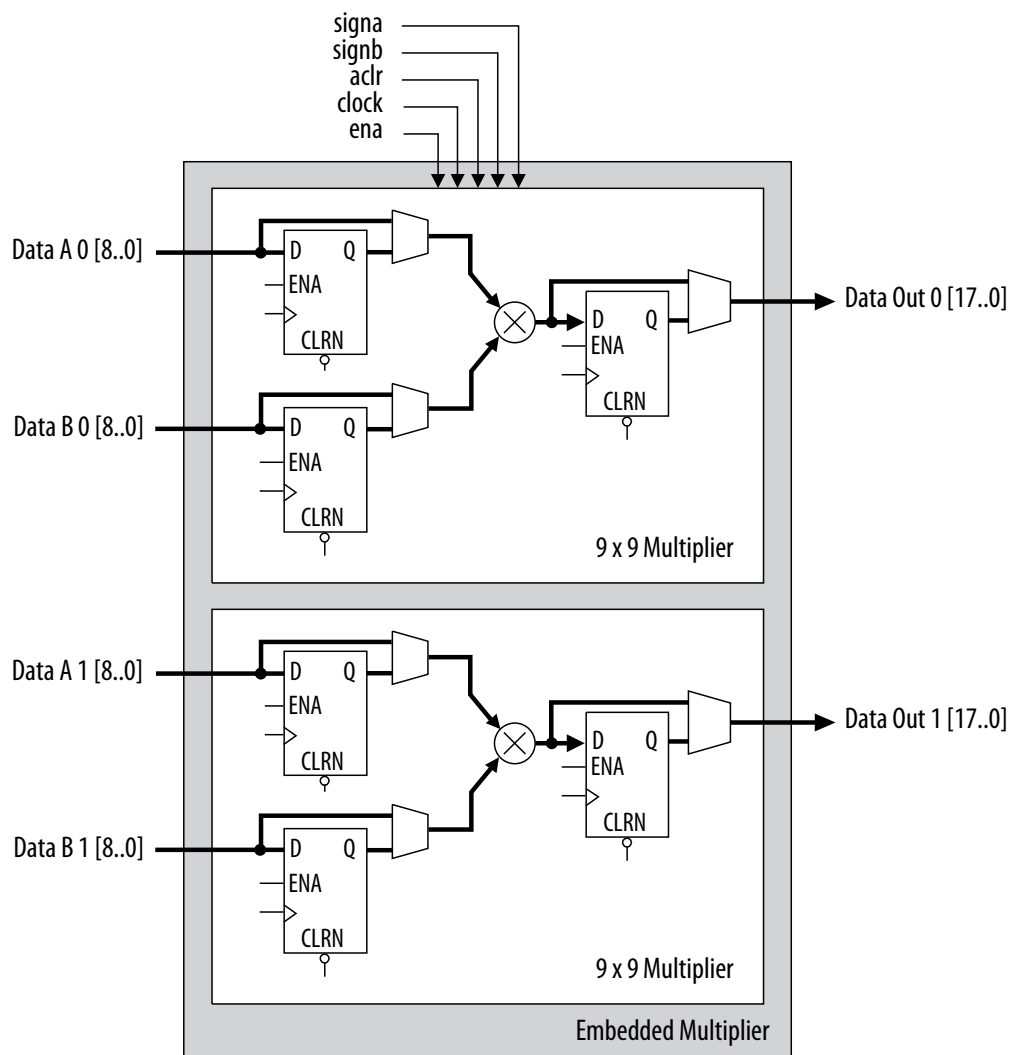
18 ビット・マルチプライヤの入力と結果は、すべてレジスタを介して個別に送信されます。マルチプライヤの入力は、符号付きの整数、符号なしの整数、またはこの2つの組み合わせを受け入れることができます。また、`signa` および `signb` 信号をダイナミックに変更し、専用の入力レジスタを介して送信することができます。

## 9 ビット乗算器

各エンベデッド・マルチプライヤをコンフィギュレーションして、最大9ビットの入力幅の2つの9×9独立マルチプライヤをサポートすることができます。

次の図は、2つの9ビット・マルチプライヤをサポートするようコンフィギュレーションされたエンベデッド・マルチプライヤを示しています。

図 9: 9 ビット・マルチプライヤ・モード



9 ビット・マルチプライヤの入力と結果は、すべてレジスタを介して個別に送信されます。マルチプライヤの入力は、符号付きの整数、符号なしの整数、またはこの 2 つの組み合わせを受け入れることができます。

各エンベデッド・マルチプライヤ・ブロックは、ブロックへの入力データの符号表現を制御するにあたって 1 つの *signa* 信号と 1 つの *signb* 信号のみを有します。エンベデッド・マルチプライヤ・ブロックが 2 つの 9x9 マルチプライヤを有する場合、以下が適用されます。

- 両方のマルチプライヤの Data A 入力と同じ *signa* 信号を共有します。
- 両方のマルチプライヤの Data B 入力と同じ *signb* 信号を共有します。

## Clocking and PLL

MAX 10 デバイスは、グローバル・クロック・ネットワーク (GCLK) とフェーズ・ロック・ループ (PLL) をサポートしています。

クロック・ネットワークは、コアにクロック・ソースを提供します。クロック・ネットワークは、リセットやクリアなどの高ファンアウトのグローバル信号ネットワークで 사용할 ことができます。

PLL により、デバイス・クロック管理、外部システム・クロック管理、および I/O インタフェース・クロッキングに関する堅牢なクロック管理および合成が可能です。

### 関連情報

[MAX 10 クロック・ネットワークと PLL ユーザー・ガイド](#)

## グローバル・クロック・ネットワーク

GCLK はすべてのデバイスのクアドラントを供給し、デバイス全体を駆動します。I/O エレメント、ロジック・アレイ・ブロック (LAB)、専用マルチプライヤ・ブロック、および M9K メモリ・ブロックといったデバイス内のすべてのリソースは、クロック・ソースとして GCLK を使用することができます。これらのクロック・ネットワーク・リソースは、クロック・イネーブルやクリアなどの外部ピンによって供給されるコントロール信号に使用します。また、内部ロジックは、内部生成の GCLK と非同期クリア、クロック・イネーブル、またはその他の高ファンアウト・コントロール信号の GCLK を駆動することができます。

図 10: 10M02、10M04、および 10M08 デバイスの GCLK ネットワーク・ソース

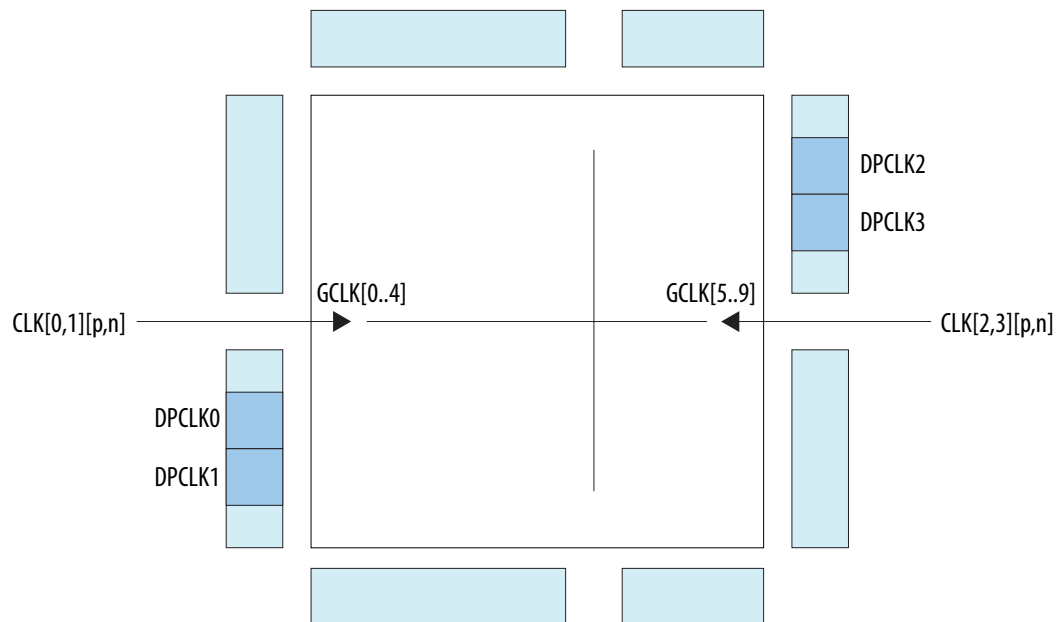
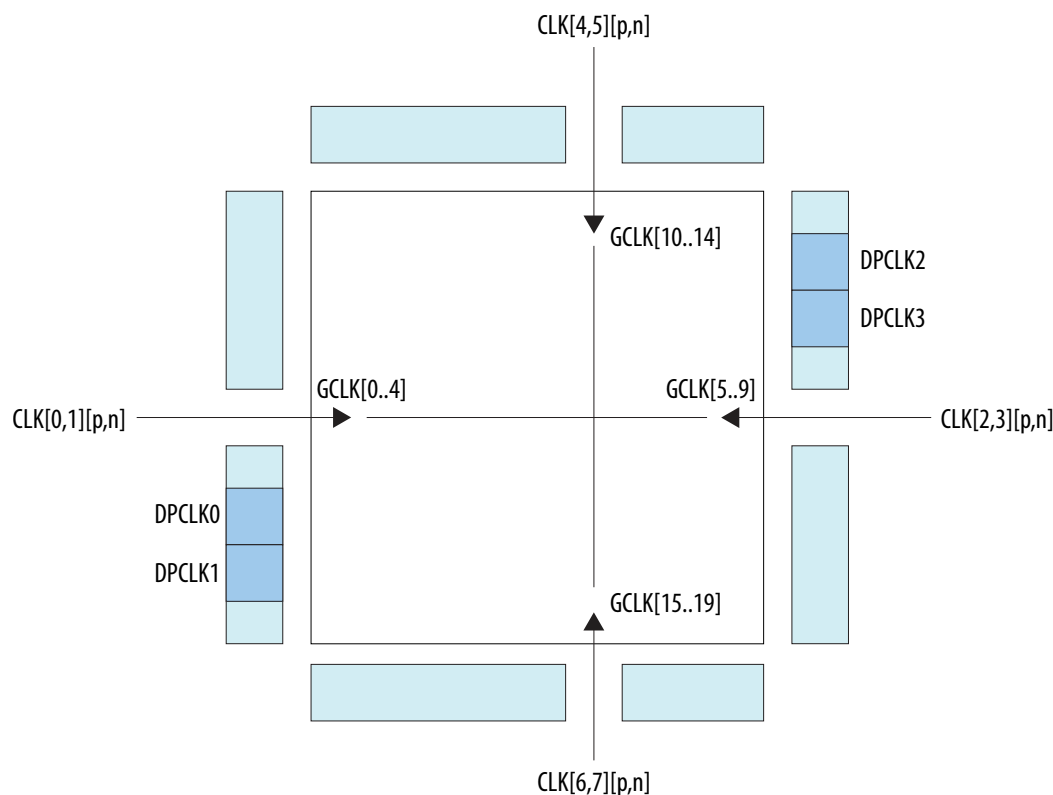


図 11: 10M16、10M25、10M40、および 10M50 デバイスの GCLK ネットワーク・ソース



## 内部オシレータ

MAX 10 デバイスは、クロック・マルチプレクサとディバイダを備えた内蔵リング・オシレータを有します。この内蔵リング・オシレータは最大 232 MHz で動作しますが、これには接続することはできません。この動作周波数はさらに低い周波数に分周されます。

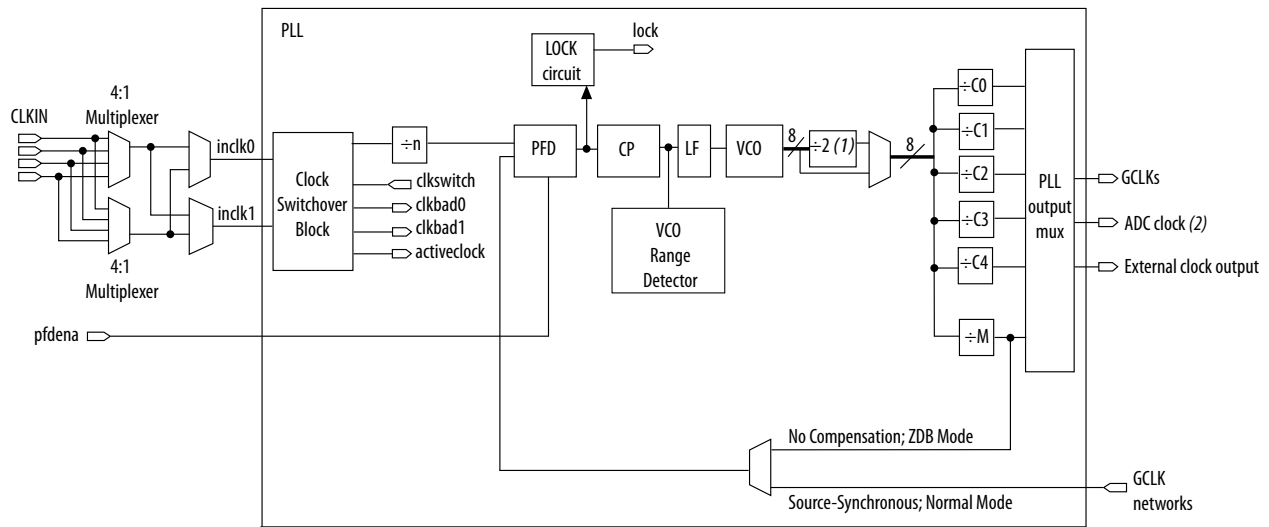
oscena 入力信号がアサートされると、オシレータが有効になり、clkout 出力信号を介して出力をロジック・アレイに配線することができます。oscena 信号が Low に設定される場合、clkout 信号は High のままです。この遅延は、TimeQuest timing analyzer を使用して解析することができます。

## PLL ブロックと位置

PLL の主な目的は、電圧制御オシレータ (VCO) の位相と周波数を入力基準クロックに同期させることです。

図 12: MAX 10 PLL のハイレベル・ブロック図

各クロック・ソースは、デバイスの PLL と同じ側にある 2 本または 4 本のクロック・ピンのいずれかから入力されます。



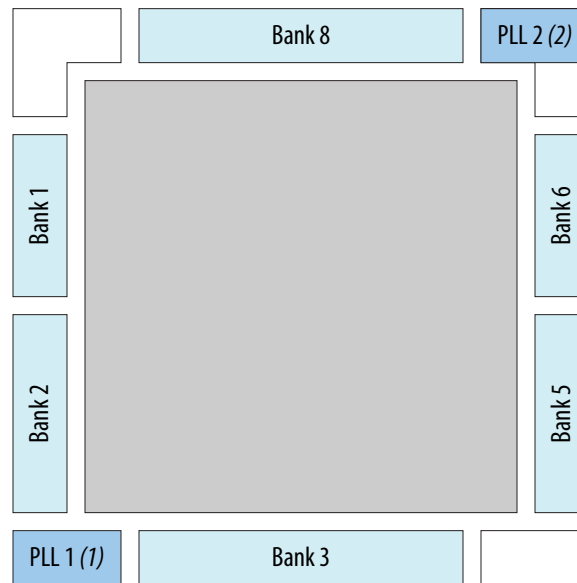
注:

- (1) これは VCO ポストスケール・カウンタです。
- (2) ADC クロックは、PLL1 と PLL3 の C0 カウンタによってのみ駆動することができます。

以下の図は、PLL の物理的な場所を示しています。各インデックスは、デバイス内の 1 つの PLL を表しています。PLL の物理的な位置は、Chip Planner の座標に対応します。Quartus Prime



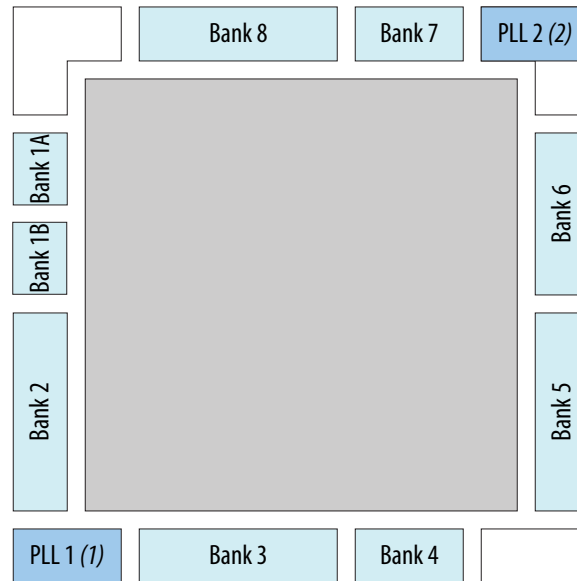
図 13: 10M02 デバイスの PLL の位置



注:

- (1) V36パッケージを除くすべてのパッケージで使用可能です。
- (2) U324およびV36パッケージでのみ使用可能です。

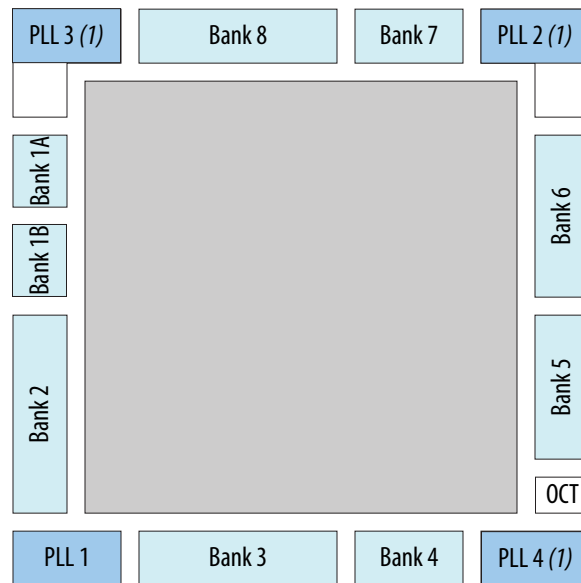
図 14: 10M04 および 10M08 デバイスの PLL の位置



注:

- (1) V81パッケージを除くすべてのパッケージで使用可能です。
- (2) F256, F484, U324, およびV81パッケージでのみ使用可能です。

図 15: 10M16、10M25、10M40、および 10M50 デバイスの PLL の位置



注:

(1) E144およびU169パッケージを除くすべてのパッケージで使用可能です。

## 汎用 I/O

MAX 10 デバイスの I/O システムは様々な I/O 規格をサポートしています。MAX 10 デバイスでは、I/O ピンはデバイス外周部の I/O バンクに配置されています。I/O ピンならびに I/O バッファは、いくつかのプログラム可能な機能を有します。

### 関連情報

[MAX 10 汎用 I/O ユーザー・ガイド](#)

## MAX 10 I/O バンクのアーキテクチャ

I/O エLEMENTは、各 I/O バンクに 4 つのモジュールのグループで配置されています。

- 高速 DDR3 I/O バンク — さまざまな I/O 規格と DDR3 を含むプロトコルをサポートします。これらの I/O バンクはデバイスの右側でのみ使用可能です。
- 高速 I/O バンク — さまざまな I/O 規格と DDR3 を除くプロトコルをサポートします。これらの I/O バンクはデバイスの上側、左側、下側で使用可能です。
- 低速 I/O バンク — デバイスの左上に配置されている低速の I/O バンクです。

I/O ピンのサポートについて詳しくは、使用しているデバイスのピンアウト・ファイルを参照してください。

## MAX 10 I/O バンクの配置

I/O バンクはデバイスの外周部に配置されています。

各デバイス・パッケージで使用可能なモジュラー I/O バンクについて詳しくは、それぞれのデバイスのピンアウト・ファイルを参照してください。

図 16: MAX 10 02 デバイスの I/O バンク (暫定版)

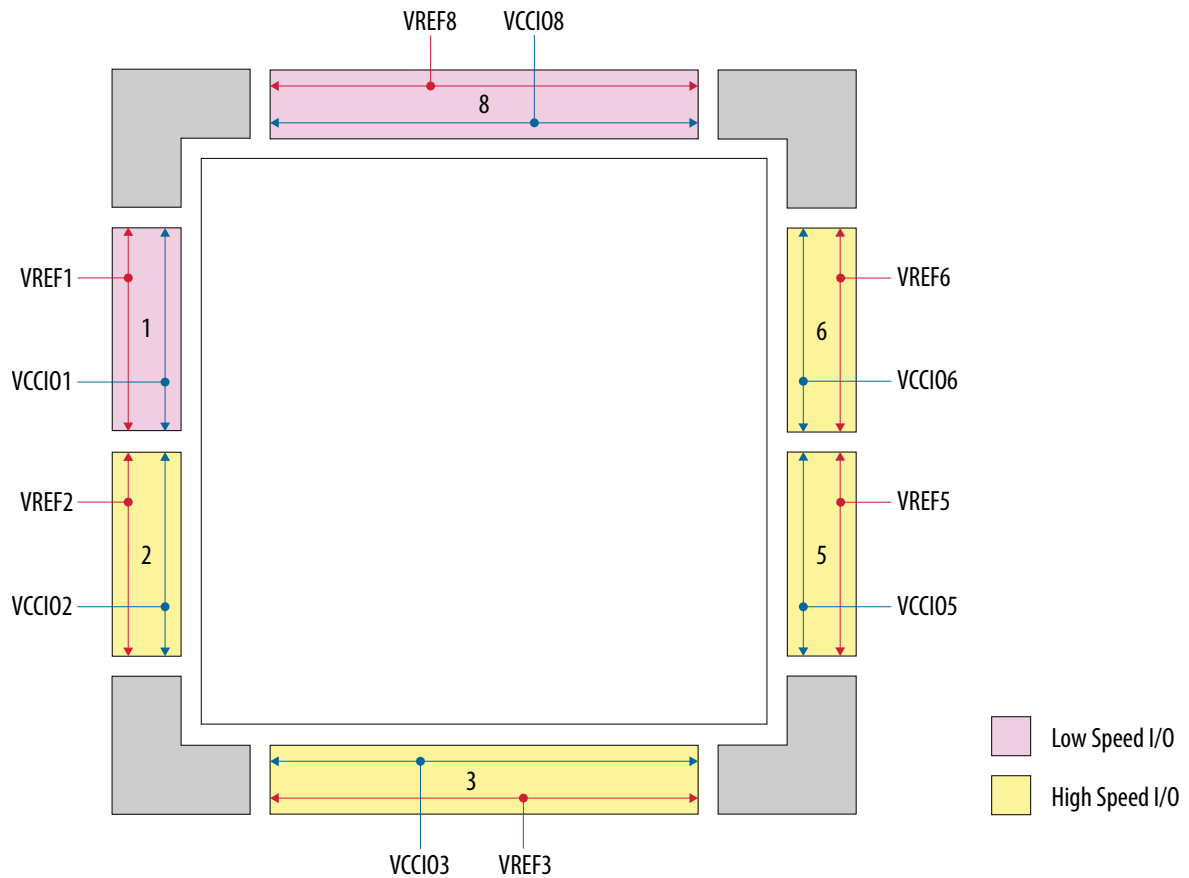


図 17: MAX 10 04 および 08 デバイスの I/O バンク (暫定版)

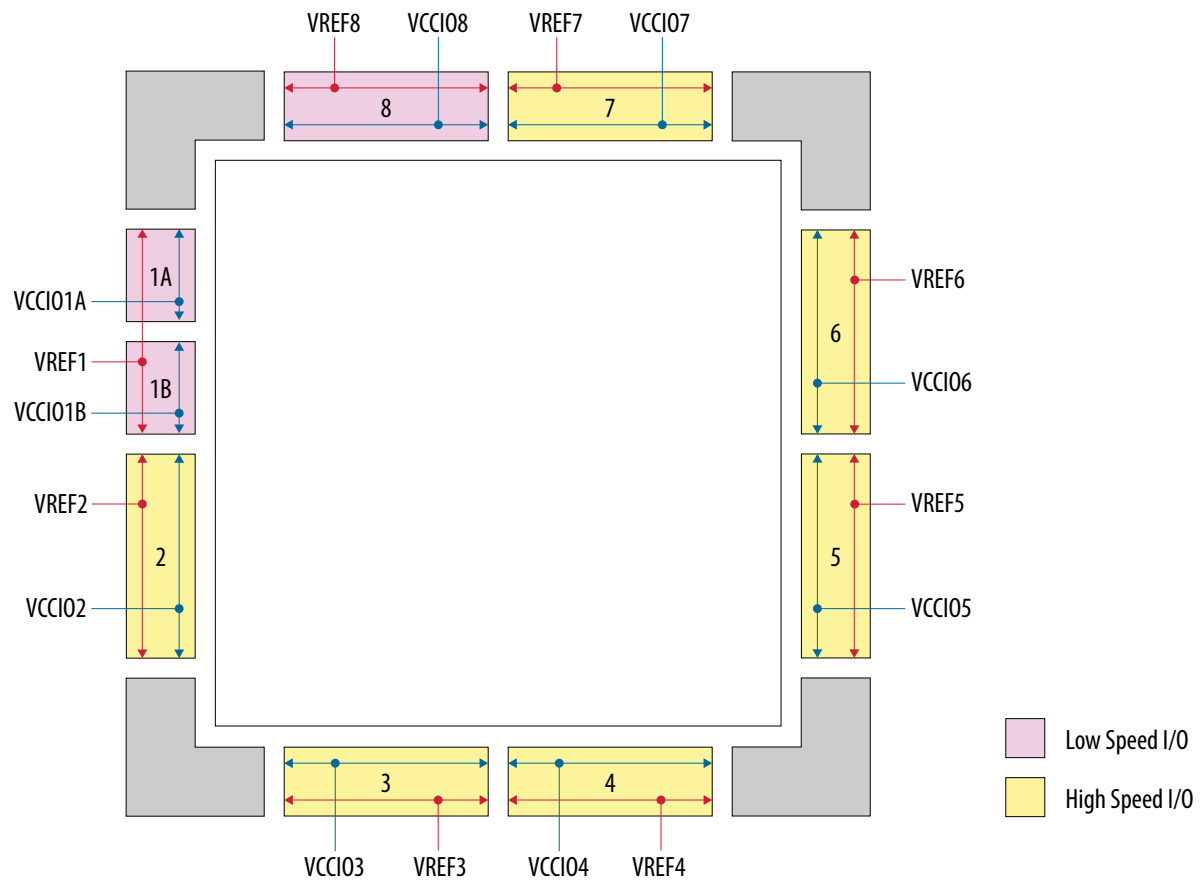
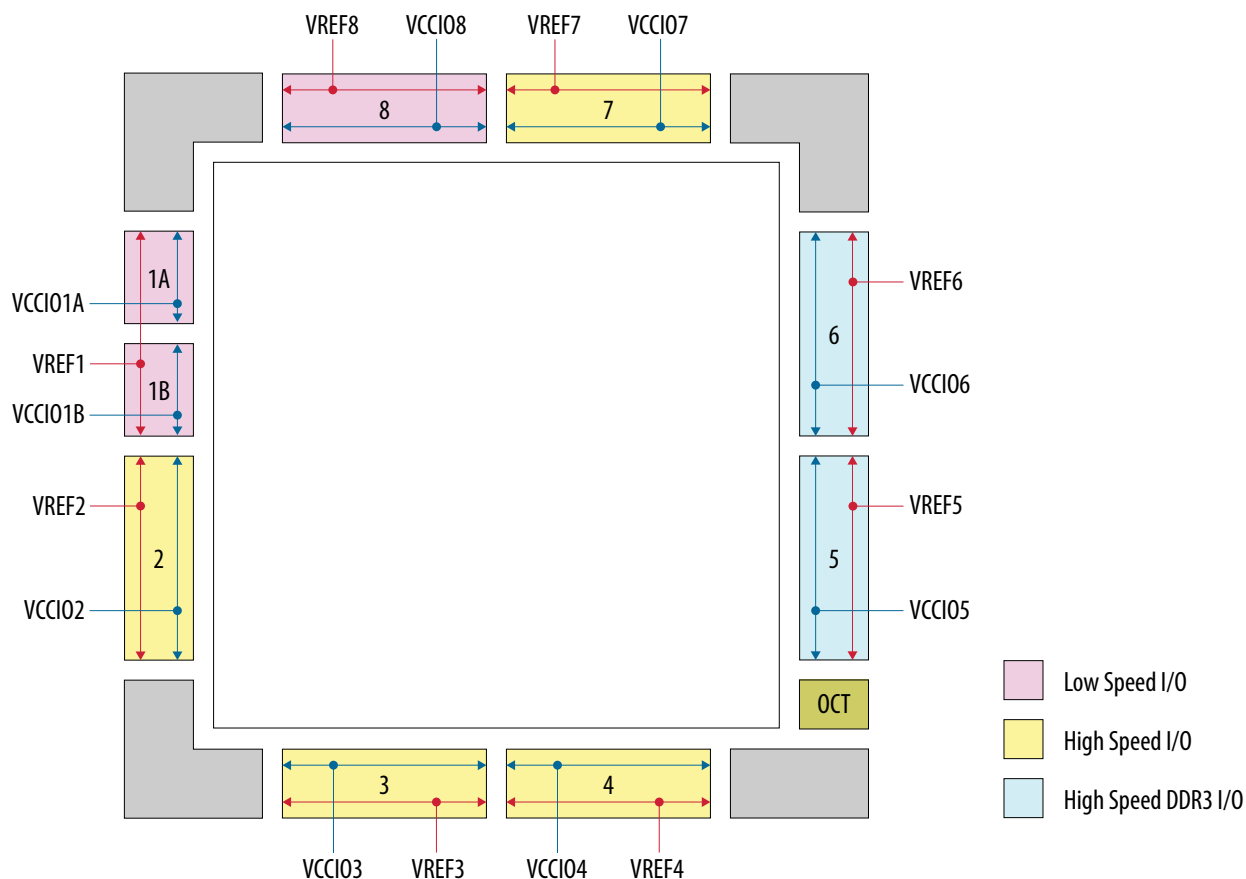


図 18: MAX 10 16、25、40、50 デバイスの I/O バンク（暫定版）



## 高速 LVDS I/O

MAX 10 デバイス・ファミリは、LVDS I/O バンクと アルテラ・ソフト LVDS IP コアにより、高速 LVDS プロトコルをサポートしています。

MAX 10 デバイスは、コア・ファブリックのレジスタとロジックを使用して、LVDS の入出力インタフェースを実装します。

- LVDS トランスミッタとレシーバの場合、MAX 10 デバイスは I/O エLEMENT (IOE) 内のダブル・データ・レート I/O (DDIO) レジスタを使用します。このアーキテクチャにより、レシーバ入力スキュー・マージン (RSKM) またはトランスミッタのチャンネル間スキュー (TCCS) の面で性能を改善できます。
- LVDS シリアライザ/デシリアライザ (SERDES) の場合、MAX 10 デバイスはロジック・エレメント (LE) レジスタを使用します。

### 関連情報

[MAX 10 高速 LVDS I/O ユーザー・ガイド](#)

## MAX 10 高速 LVDS 回路

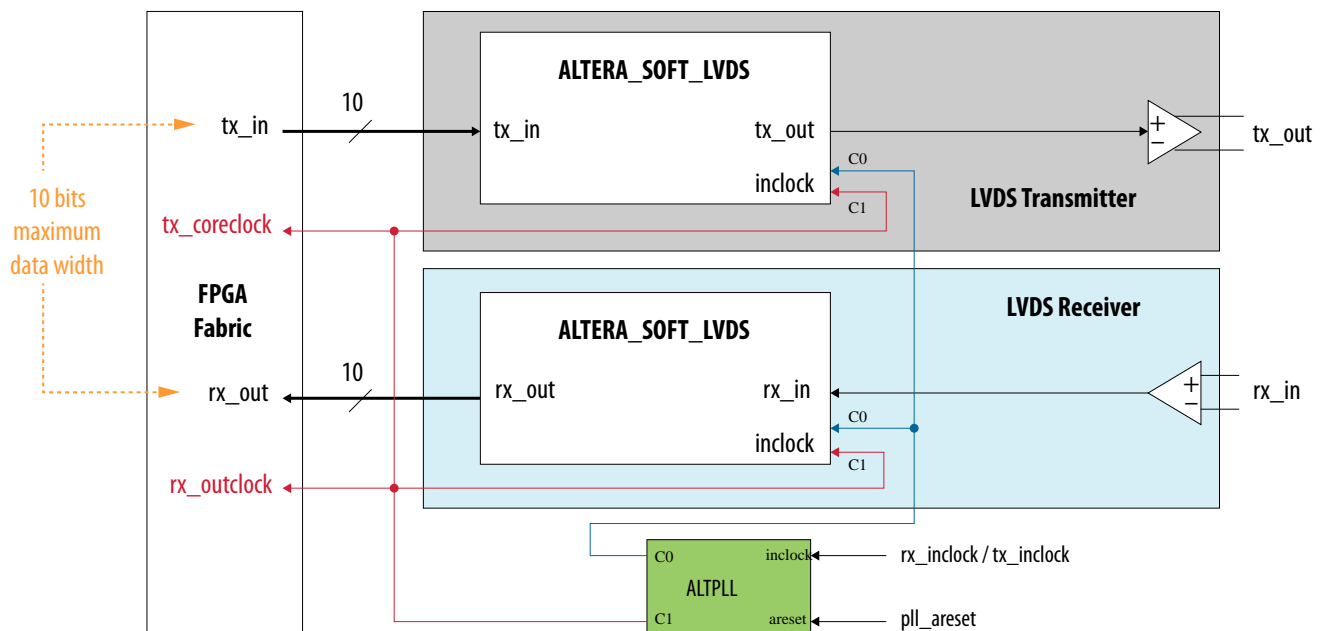
LVDS ソリューションは、MAX 10 デバイスの I/O エLEMENT とレジスタを使用します。アルテラ・ソフト LVDS IP コアは、コア・ロジックにシリアルライザとデシリアルライザをソフト SERDES ブロックとして実装します。

MAX 10 デバイスは専用のシリアライゼーションまたはデシリアライゼーション回路を備えていません。

- デバイスへの高速差動インタフェース実装には I/O ピンおよびコア・ファブリックを使用する
- MAX 10 ソリューションは送受信データのシリアル - パラレルおよびパラレル - シリアル変換を行うためにシフト・レジスタ、内部 PLL、I/O エレメントを使用する
- Quartus Prime ソフトウェアはコア・ファブリックに自動的に SERDES を構築するためにアルテラ・ソフト LVDS IP コアのパラメータ設定を使用する

### 図 19: ソフト LVDS SERDES

以下に、ソフト LVDS SERDES 回路のトランスミッタとレシーバ、およびトランシーバ・パスとレシーバ・パスのインタフェース信号を含むブロック図を示します。



## MAX 10 高速 LVDS I/O の位置

MAX 10 デバイスの I/O バンクでは、全ての I/O バンクで真の LVDS 入力とエミュレーション LVDS 出力をサポートしています。また、デバイス下側の I/O バンクでのみ真の LVDS 出力をサポートしています。

図 20: 10M02 デバイスの I/O バンクでの LVDS サポート

以下はシリコン・ダイの概略図です。各バンクに実際のバンク番号を示しています。LVPECLは、バンク 2 と 6 でのみサポートしています。

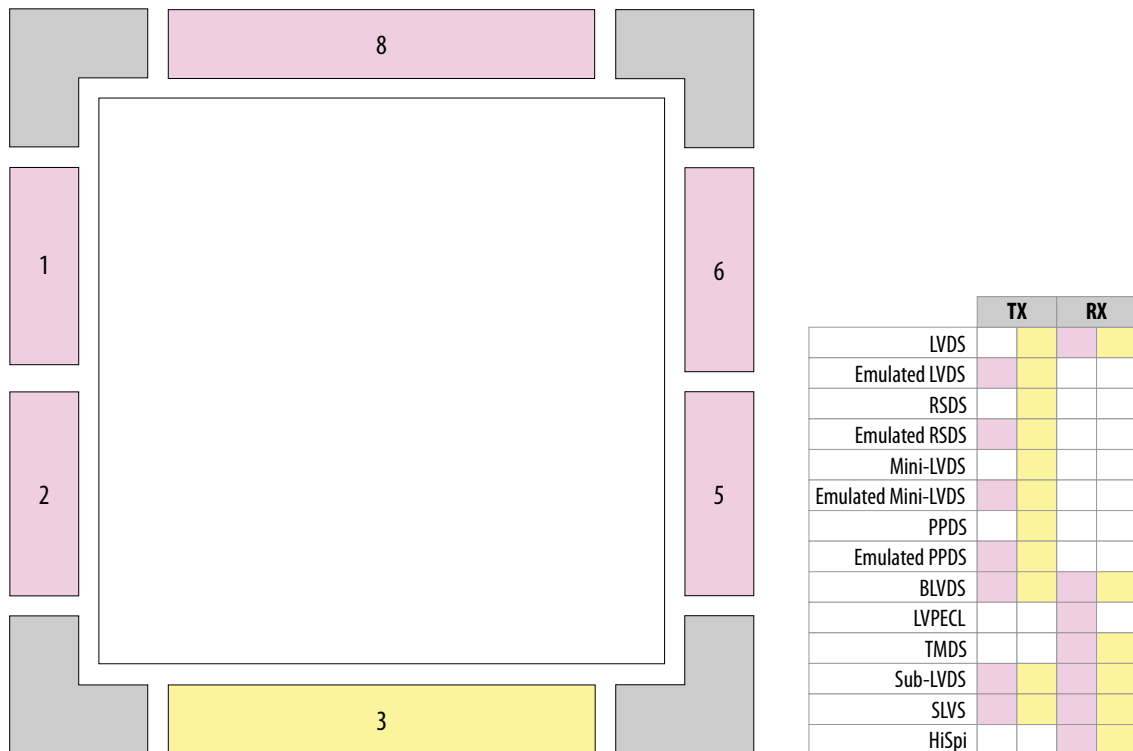


図 21: 10M04 および 10M08 デバイスの I/O バンクでの LVDS サポート

以下はシリコン・ダイの概略図です。各バンクに実際のバンク番号を示しています。LVPECL は、バンク 2 と 6 でのみサポートしています。

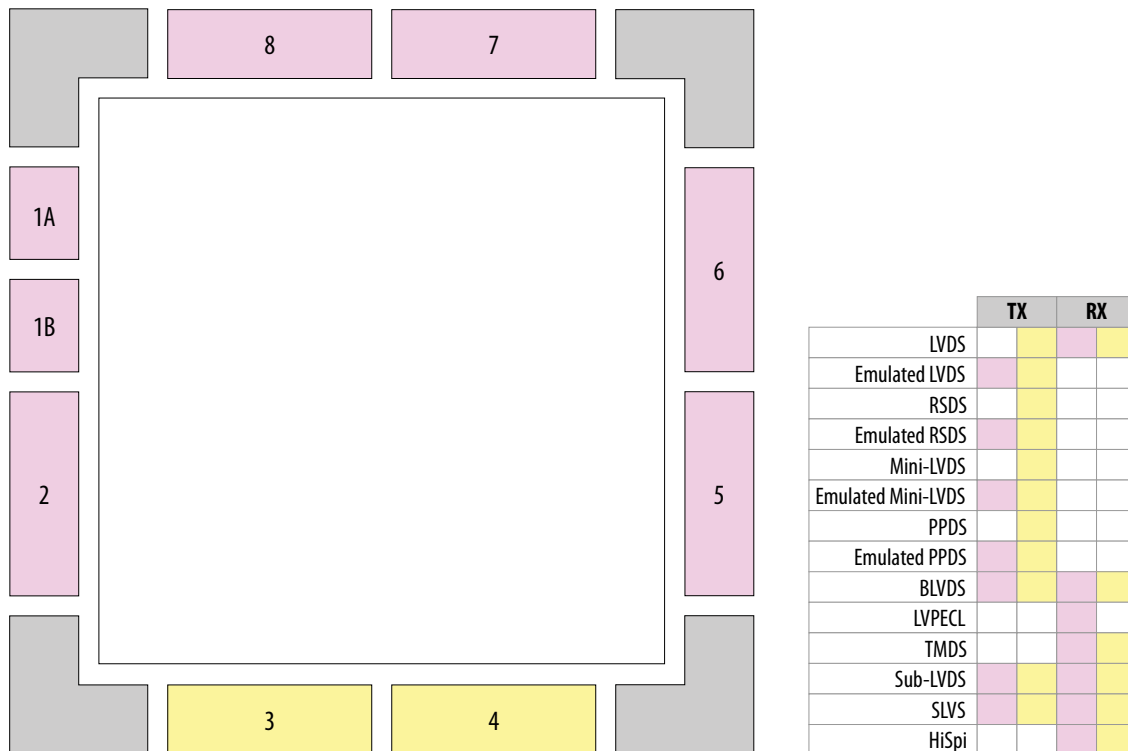
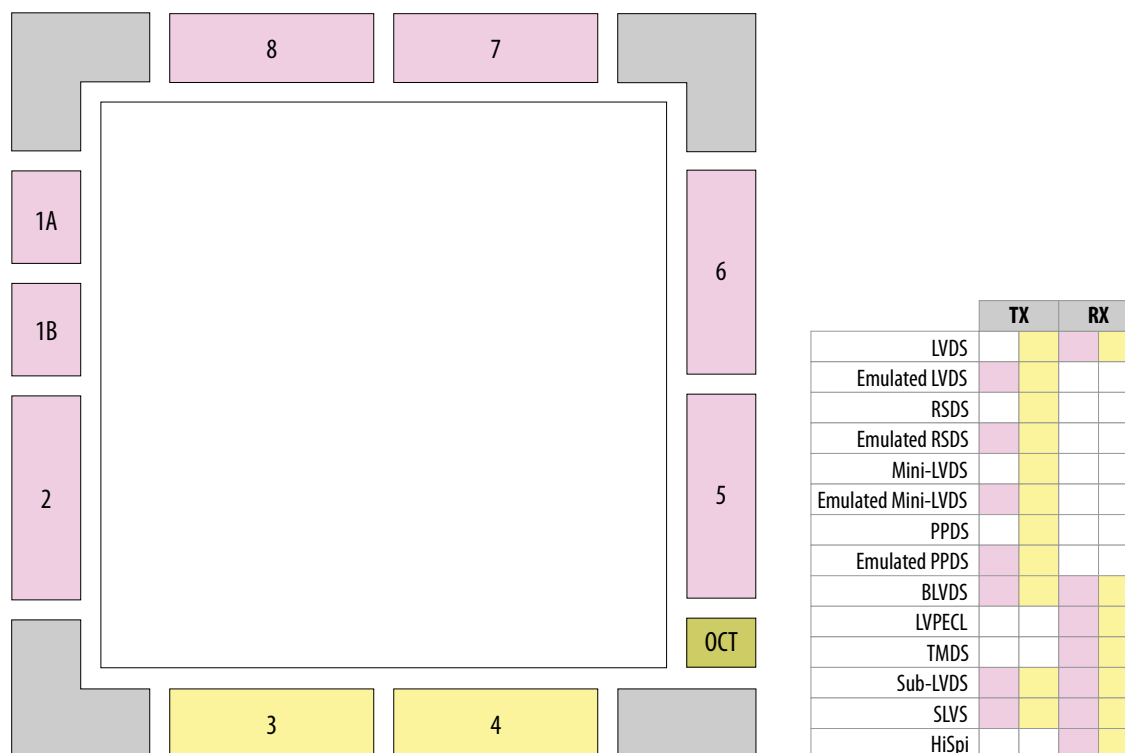




図 22: 10M16、10M25、10M40、10M50 デバイスの I/O バンクでの LVDS サポート

以下はシリコン・ダイの概略図です。各バンクに実際のバンク番号を示しています。LVPECLは、バンク 2、3、6、8 でのみサポートしています。



## 外部メモリ・インタフェース

MAX 10 デバイスは、幅広い外部メモリ規格とインタフェース接続することができます。

この機能により、画像処理、ストレージ、通信、一般的なエンベデッド・システムなどの幅広いアプリケーションで MAX 10 デバイスを使用することができます。

MAX 10 デバイスの外部メモリ・インタフェース・ソリューションは、以下の要素で構成されます。

- 外部メモリ・インタフェースをサポートする I/O エLEMENT。
- メモリ・インタフェースのコンフィギュレーションによりさまざまな外部メモリ・インタフェース規格をサポートするための UniPHY IP コア。

### 関連情報

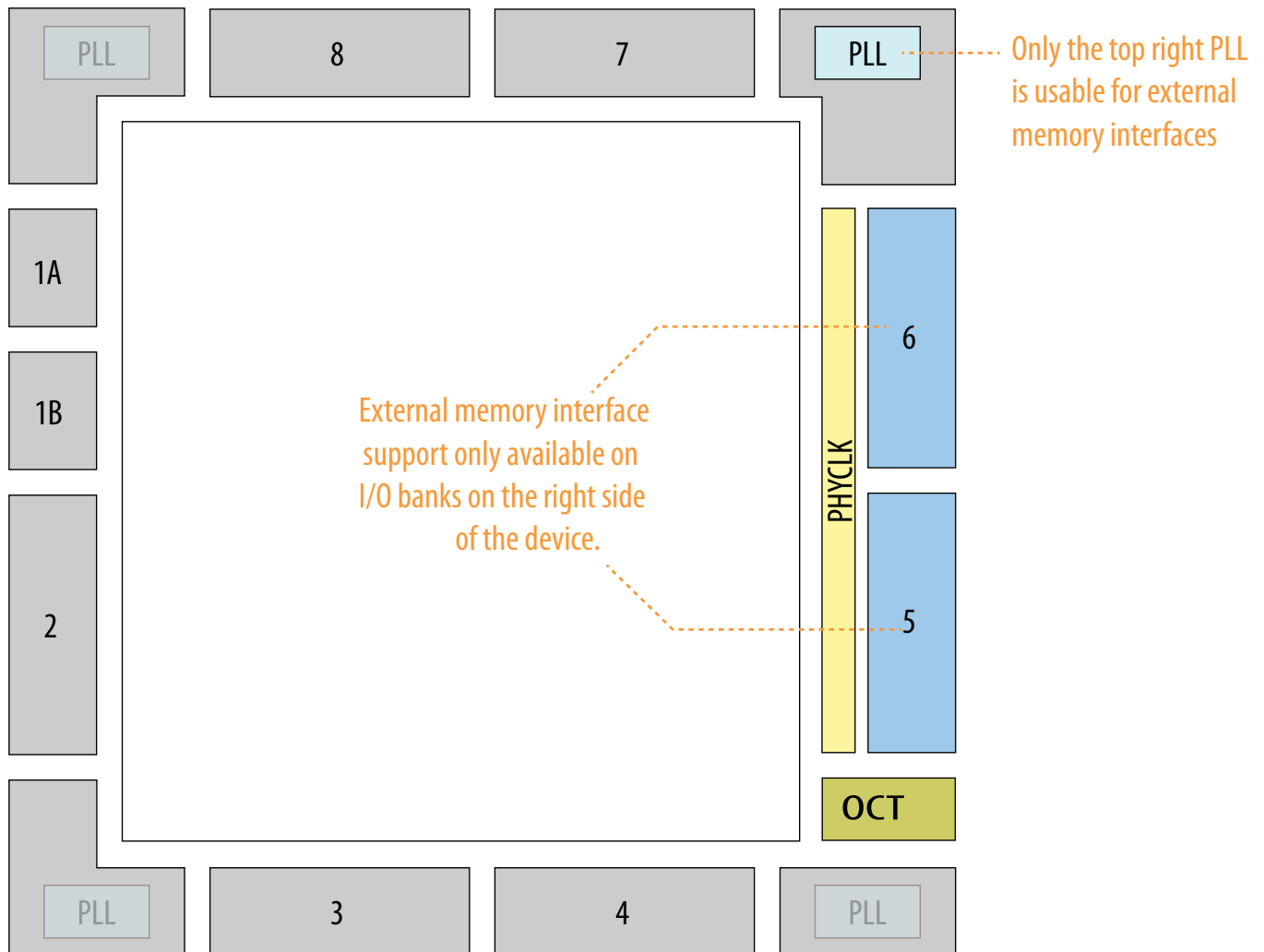
[MAX 10 外部メモリ・インタフェース・ユーザーガイド](#)

## MAX 10 外部メモリ・インタフェースの I/O バンク

MAX 10 デバイスの外部メモリ・インタフェースは、デバイス右側の I/O バンクだけでサポートされています。そのため、外部メモリ I/O ピンはすべて、デバイス右側の I/O バンクに配置する必要があります。

図 23: 外部メモリ・インタフェースの I/O バンク

次の図は、デバイス・パッケージの裏面図に相当するシリコン・ダイの上面図を表しています。



外部メモリ・インタフェースのサポートは 10M16、10M25、10M40、および 10M50 デバイスでのみ利用可能です。

## アナログ・デジタル・コンバーター

MAX 10 デバイスは、2つのアナログ・デジタル・コンバーター（ADC）まであります。ADC はオンダイ温度監視および外部アナログ信号変換のための組み込みの機能を備えた MAX 10 デバイスを提供します。

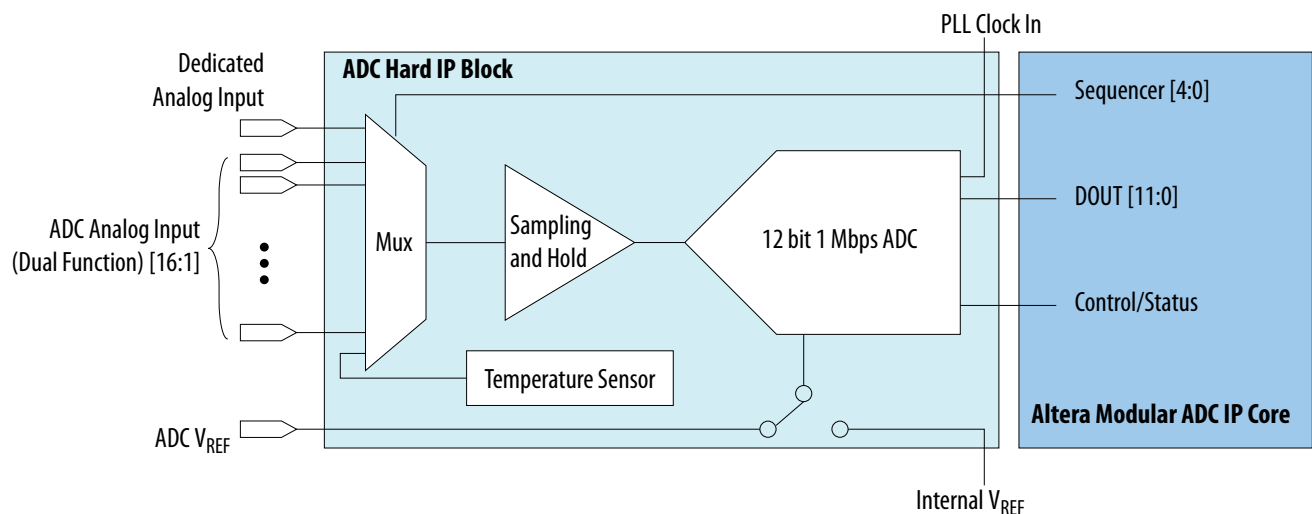
ADC ソリューションは、アルテラ・モジュラー ADC IP コアを介して MAX 10 デバイスのペリフェラルとソフト・ロジックでハード IP ブロックで構成されています。

ADC ソリューションは、情報処理、コンピューティング、データの送信、およびコントロール・システムのためのデジタル・データにアナログ量を変換する組み込みの機能を提供します。基本機能は、観察されているアナログ信号の 12 ビット・デジタル表現を提供する必要があります。

ADC ソリューションは、2つのモードで動作します。

- ノーマル・モード—1 秒 (Msps) あたり 1つのメガシンボルの累積しているサンプリングレートによって最高 18 回のシングル・エンドの外部のインプットに監視します。
- 温度検知モード—1 秒 (ksps) あたり最高 50 のキロシンボルのサンプリング・レートによって内部の温度データ入力を監視します。

図 24: MAX 10 デバイスの ADC Hard IP ブロック



### 関連情報

[MAX 10 アナログ-デジタル・コンバータ・ユーザー・ガイド](#)

## ADC ブロックの位置

ADC ブロックは、MAX 10 デバイス外周部の左上の角に配置されています。

図 25: MAX 10 04 と 08 デバイスでの ADC ブロックの位置

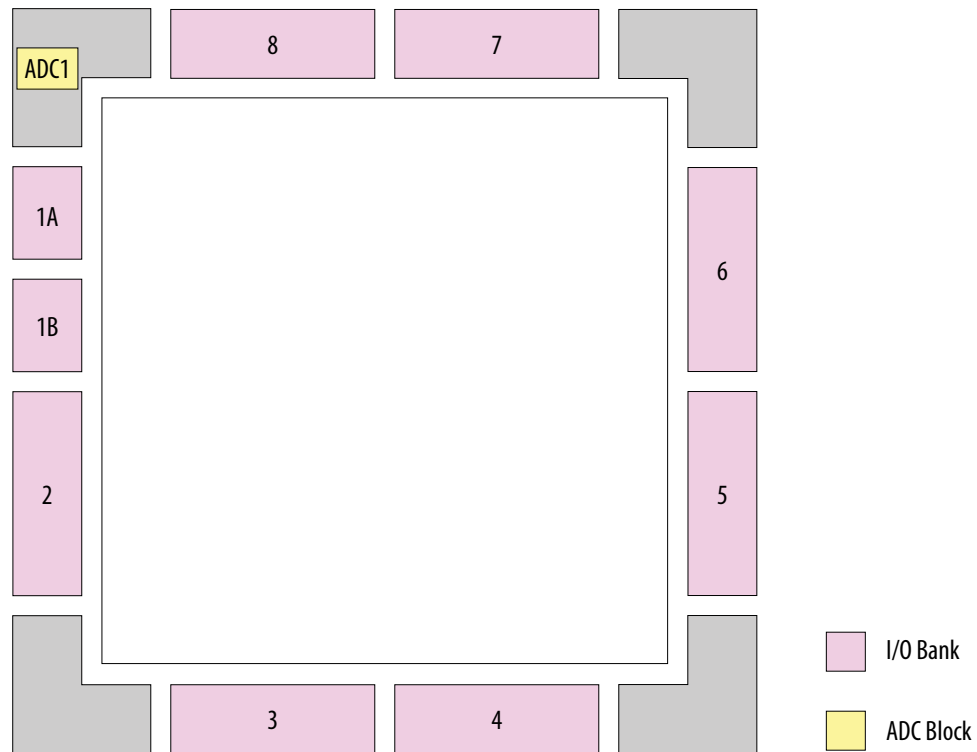


図 26: MAX 10 16 デバイスでの ADC ブロックの位置

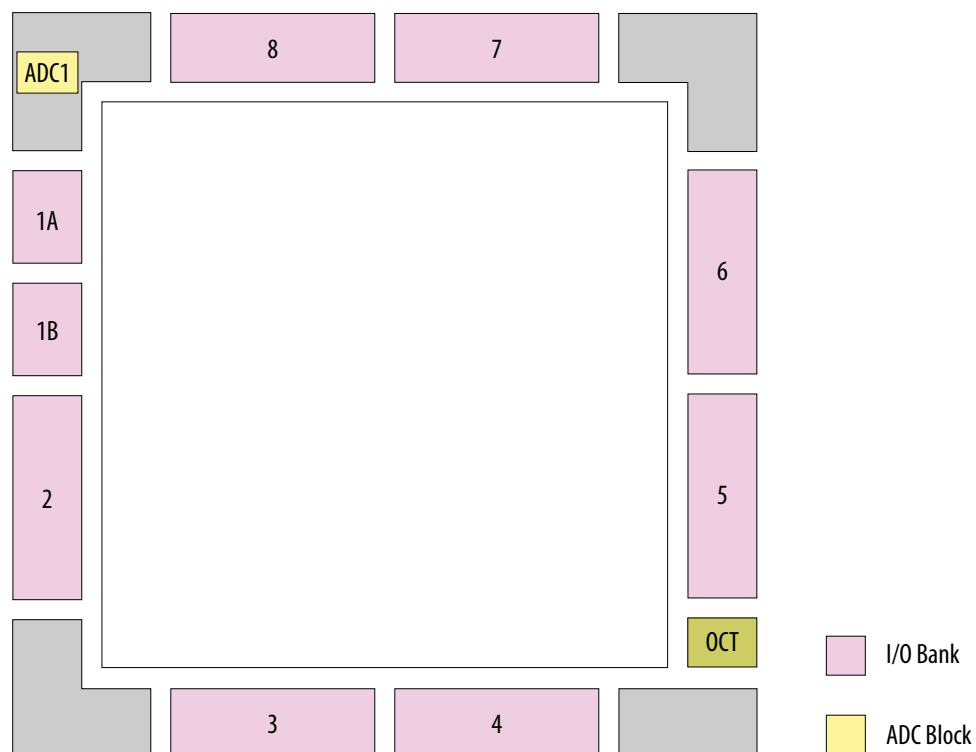
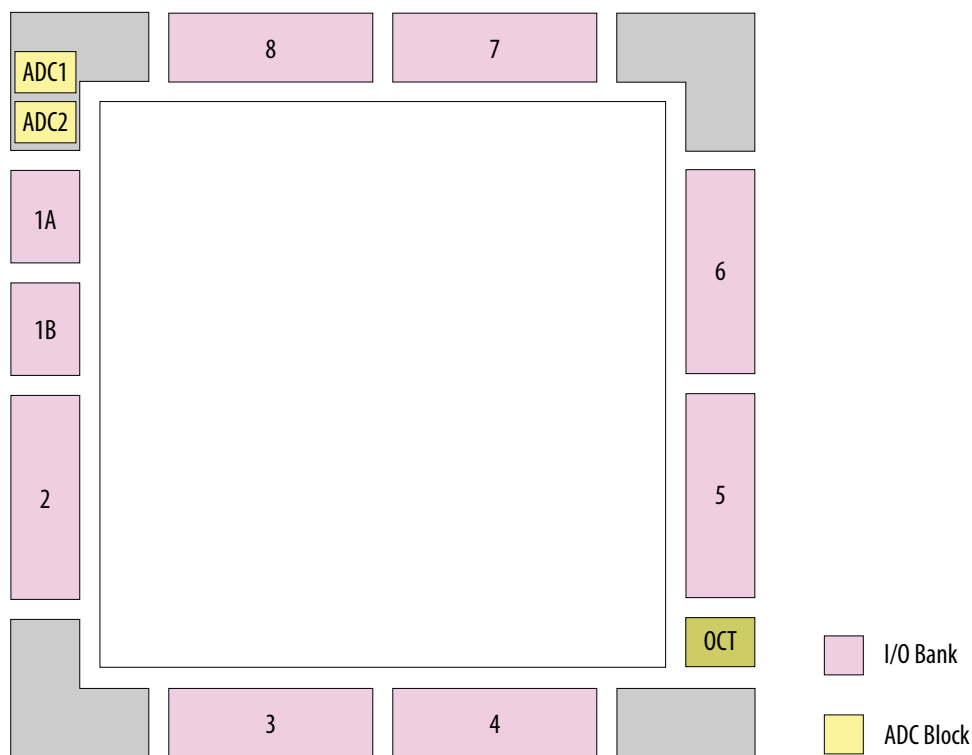


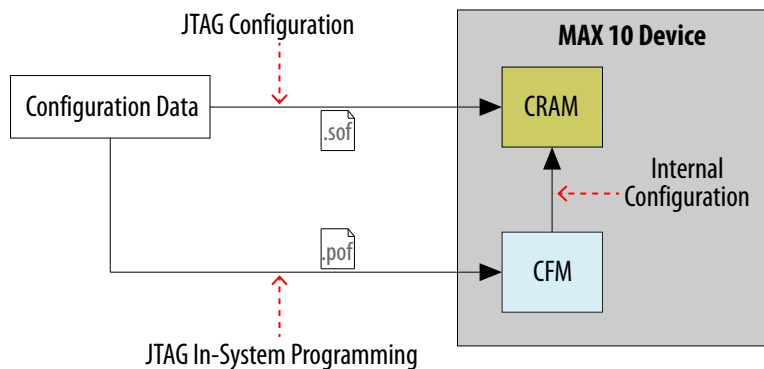
図 27: MAX 10 25、40 と 50 デバイスでの ADC ブロックの位置

これらのデバイスの E144 パッケージが備える ADC ブロックは 1 つのみです。



## コンフィギュレーション手法

図 28: MAX 10 デバイスの JTAG コンフィギュレーションおよび内部コンフィギュレーションの概要



### 関連情報

[MAX 10 FPGA コンフィギュレーション・ユーザー・ガイド](#)

## JTAG コンフィギュレーション

MAX 10 デバイスでは、JTAG 命令は内部コンフィギュレーション手法よりも優先されます。

JTAG コンフィギュレーション手法を使用して、JTAG インタフェースの TDI、TDO、TMS および TCK ピンを介して直接的にデバイスの CRAM をコンフィギュレーションすることができます。Quartus Prime ソフトウェアは SRAM オブジェクト・ファイル (.sof) を自動的に生成します。.sof は、ダウンロード・ケーブルと Quartus Prime ソフトウェア・プログラマを使用してプログラミングが可能です。

## 内部コンフィギュレーション

内部コンフィギュレーションを行う前に、コンフィギュレーション・データをコンフィギュレーション・フラッシュ・メモリ (CFM) にプログラミングしておく必要があります。CFM に書き込まれるコンフィギュレーション・データは、プログラム・オブジェクト・ファイル (.pof) の一部になります。JTAG インシステム・プログラミング (ISP) を使用して、.pof を内部フラッシュにプログラミングします。

内部コンフィギュレーション時には、MAX 10 デバイスは CFM からのコンフィギュレーション・データを CRAM にロードします。

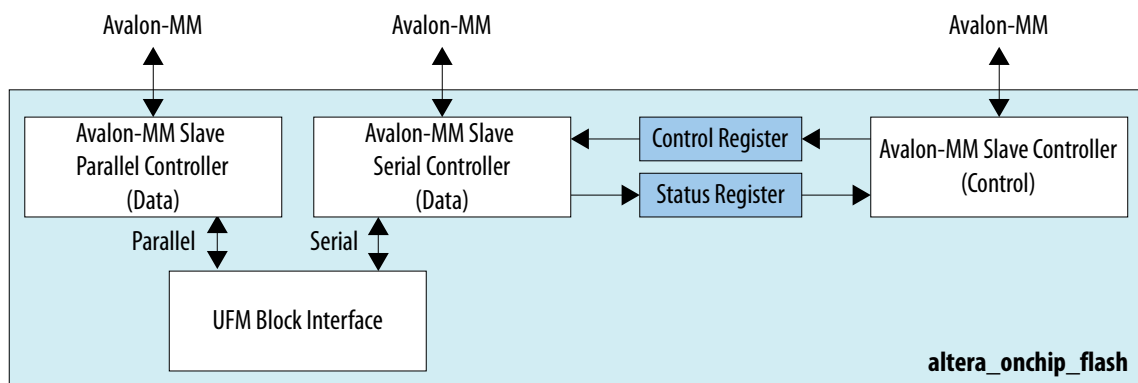
## ユーザー・フラッシュ・メモリ

アルテラの MAX 10 デバイスは、不揮発性情報を格納するユーザー・フラッシュ・メモリ (UFM) ブロックを備えています。

UFM は、MAX 10 デバイスで使用可能な内蔵フラッシュの一部です。

MAX10 デバイスの UFM アーキテクチャは、ソフトとハード IP の組み合わせです。Quartus II ソフトウェアでアルテラのオンチップ・フラッシュ IP コアを使用して、UFM にアクセスすることができます。

図 29: アルテラのオンチップ・フラッシュ IP のブロック図



この IP ブロックは、2つの Avalon-MM スレーブ・コントローラがあります。



- データ・フラッシュへのリードおよびライト・アクセスを提供する UFM ブロックのラップです。
- コントローラーライト動作のために必要とされるフラッシュのための CSR とステータス・レジスタです。

#### 関連情報

[MAX 10 ユーザー・フラッシュ・メモリ \(UFM\) ユーザー・ガイド](#)

## 電源管理

MAX 10 電源最適化の機能は以下のとおりです。

- シングル電源またはデュアル電源のデバイス・オプション
- パワー・オン・リセット (POR) 回路
- パワー・マネージメント・コントローラ手法
- ホット・ソケット

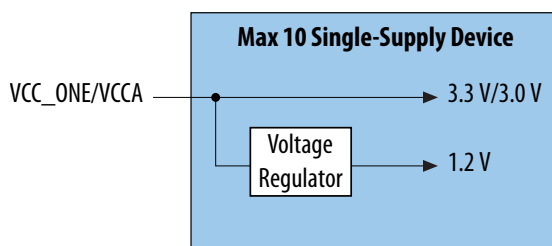
#### 関連情報

[電源管理ユーザー・ガイド](#)

## シングル電源デバイス

MAX 10 のシングル電源デバイスは、3.0 V または 3.3 V の外部電源いずれか一方のみを必要とします。外部電源を MAX 10 デバイスの VCC\_ONE と VCCA 電源ピンへの入力として供給します。次いで、この外部電源は MAX 10 シングル電源デバイスの内部電圧レギュレータによって 1.2 V に調節されます。この 1.2 V の電圧レベルは、コア・ロジックの動作に必要です。

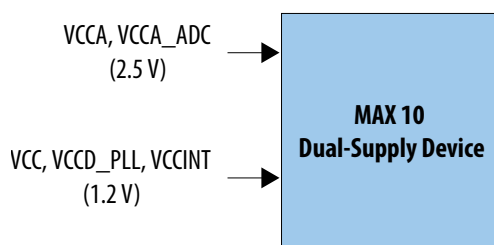
図 30: MAX 10 シングル電源デバイス



## デュアル電源デバイス

MAX 10 のデュアル電源デバイスは、デバイスのコア・ロジックと外周部の動作のために 1.2 V および 2.5 V を必要とします。

図 31: MAX 10 デュアル電源デバイス



## パワー・マネージメント・コントローラ手法

パワー・マネージメント・コントローラ手法により、ランタイム中にアプリケーションをスリープ・モードにすることができます。これにより、デザインの一部をオフにすることが可能になるので、ダイナミック電力の消費量を削減します。アプリケーションは、1ms 未満の高速ウェイクアップ時間で再度有効にすることができます。

## ホット・ソケット

MAX 10 デバイスは、ホット・プラグイン、またはホット・スワップとも呼ばれるホット・ソケットに対応しており、また外部デバイスの使用を伴わない電源シーケンスをサポートしています。これにより、システム・バスの動作やシステムに挿入されたボードに影響を与えることなく、システムの動作中にシステム内のボード上で MAX 10 デバイスの挿入または取り外しをすることができます。

ホット・ソケット機能は、異なる電圧レベルのデバイスが組み合わされた PCB 上で MAX 10 デバイスを使用する際に直面するいくつかの困難を取り除きます。

MAX 10 デバイスのホット・ソケット機能を用いると、適切なパワーアップ・シーケンスをボード上のデバイスごとに保障する必要がなくなります。MAX 10 デバイスのホット・ソケット機能は以下を提供します。

- 外部コンポーネントまたはボードの操作を伴わない、ボードやデバイスの挿入と取り外し
- あらゆるパワーアップ・シーケンスへのサポート
- ホット・インサクション中に I/O バッファがシステム・バスに影響を与えない

## MAX10 FPGA デバイス・アーキテクチャの改訂履歴

日付	バージョン	変更内容
2016 年 8 月	2016.08.11	エンベデッド・マルチプライヤより重複した内容を削除。

日付	バージョン	変更内容
2016 年 5 月	2016.05.13	<ul style="list-style-type: none"> <li>内蔵オシレータの構造に関する情報を追加。</li> <li>項を題名をクロック・ネットワークと PLL からクロッキングと PLL に変更。</li> <li>高速 LVDS 回路の情報を追加。</li> <li>パワー・マネージメント・コントローラ手法およびホット・ソケットの情報を追加。</li> </ul>
2015 年 5 月	2015.05.04	<ul style="list-style-type: none"> <li>「内部コンフィギュレーション」の図を削除。</li> <li>「コンフィギュレーション」の「MAX 10 デバイスの JTAG コンフィギュレーションおよび内部コンフィギュレーションの概要」の図を追加。</li> </ul>
2014 年 12 月	2014.12.15	<ul style="list-style-type: none"> <li>ユーザー・フラッシュ・メモリのアルテラのオンチップ・フラッシュ IP コア・ブロック図を更新。</li> <li>リンクを更新。</li> </ul>
2014 年 9 月	2014.09.22	初版。