

### はじめに

PLL (Phase-locked loop) はいくつかの分周カウンタおよび遅延エレメントを使用して、周波数合成と位相シフトを実行します。Stratix® および Stratix® GX の enhanced PLL では、これらのカウンタと遅延エレメントをリアルタイムでコンフィギュレーション可能です。設計者は、FPGA 全体をリコンフィギュレーションすることなく、出力クロック周波数と時間遅延をリアルタイムで変更することができます。

PLL リコンフィギュレーション機能は、異なる周波数で動作するアプリケーションで役立ちます。プロトタイプ環境でも容易にリアルタイムで PLL 出力周波数をスイープし、クロック遅延を調整することができます。例えば、テスト・パターンを生成するシステムは、被試験ユニットに応じて、50MHz または 100MHz のいずれかでパターンを生成および送信します。PLL コンポーネントのリアルタイム・リコンフィギュレーションにより、20 $\mu$ s 以内 (PLL のロック時間を除く) に、このような 2 つの出力周波数を切り替えることができます。この機能を使用して、出力クロックの遅延を変更することにより、リアルタイムでクロック - 出力間 ( $t_{CO}$ ) の遅延を調整することも可能です。この方法を使用すれば、新しい PLL 設定でプログラミング・ファイルを再生成する必要はありません。

本書では Stratix および Stratix GX enhanced PLL での、リアルタイム PLL リコンフィギュレーションの実装について解説し、PLL パラメータの選択方法を説明します。このアプリケーション・ノートでは、Quartus® II ソフトウェアでの PLL リコンフィギュレーション機能の実装方法についても説明しています。

### PLL リコンフィ ギュレーション・ハード ウェアの実装

Stratix デバイスの enhanced PLL は、リアルタイムの PLL リコンフィギュレーションをサポートします。以下の PLL コンポーネントはリアルタイムでコンフィギュレーション可能です。

- プリスケール・カウンタおよび遅延エレメント ( $n, \Delta t_n$ )
- フィードバック・カウンタおよび遅延エレメント ( $m, \Delta t_m$ )
- ポスト・スケール出力カウンタおよび遅延エレメント ( $g, \Delta t_g, l, \Delta t_l, e, \Delta t_e$ )

チャージ・ポンプ電流、ループ・フィルタ・コンポーネント、および電圧制御発振器 (VCO) タップを使用した位相シフトは動的に調整することはできません。

図 1 はリコンフィギュレーション回路のブロック図を示します。

図 1. PLL リンコンフィグレーション回路のブロック図

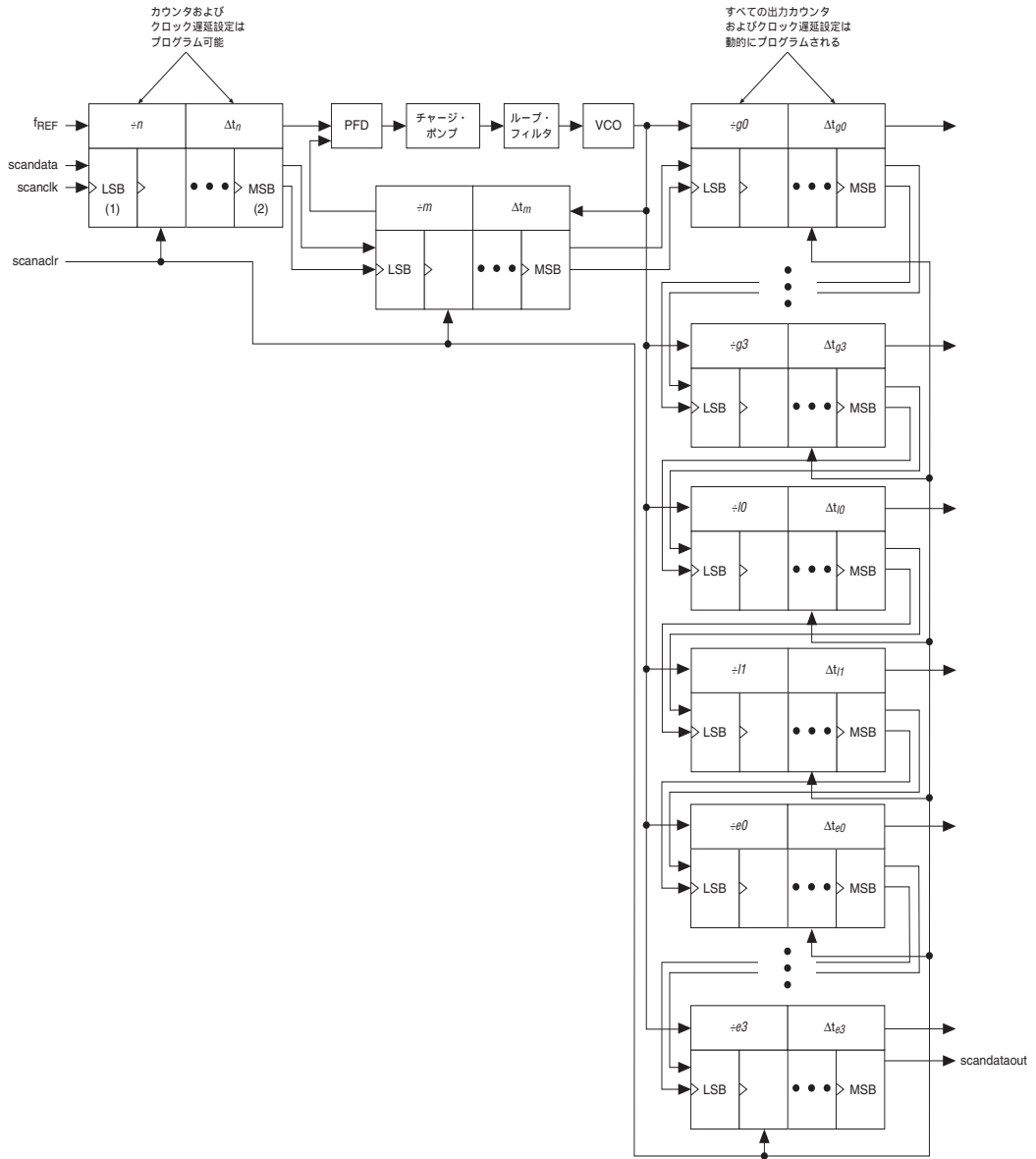


図 1 の注 :

- (1) LSB: 最下位ビット
- (2) MSB: 最上位ビット

図 1 に示すとおり、PLL カウンタと遅延エレメントは、新しい設定をシリアル・シフト・レジスタ・チェーンまたはシフト・レジスタにシフトすることによって動的に調整できます。シリアル・データは `scandata` ポートを経由してシフト・レジスタに入力され、`scanclk` によってクロック駆動されます。データの最終ビットがクロックされた後、PLL コンフィギュレーション・ビットはスキャン・レジスタ内のデータで同期的に更新されます。カウンタと遅延エレメントの設定は同期的に更新されますが、これらの設定は個々のカウンタ・クロック周波数に基づいて一度に 1 カウンタずつ更新されます。`scanclk` の最大入力シフト・クロック・レートは 100MHz です。`scanaclr` を使用して、シフト・レジスタ内のレジスタを非同期的にクリアすることも可能です。このクリア信号の最小パルス幅は 5ns です。これらの信号はすべて FPGA ロジック・アレイまたは I/O ピンでドライブできます (表 1 を参照)。

表 1. リアルタイム PLL リコンフィギュレーション・ポート

PLL ポート名	説明	ソース	出力先
<code>scandata</code>	シリアル入力データ・ストリーム	ロジック・アレイまたは I/O ピン	PLL リコンフィギュレーション回路
<code>scanclk</code>	シリアル・クロック入力信号	ロジック・アレイまたは I/O ピン	PLL リコンフィギュレーション回路
<code>scanaclr</code>	シリアル・シフト・レジスタ・チェーン用のアクティブ High、非同期クリア信号	ロジック・アレイまたは I/O ピン	PLL リコンフィギュレーション回路
<code>scandataout</code>	シフト・レジスタ内の最後のシフト・レジスタの出力	PLL リコンフィギュレーション回路	ロジック・アレイまたは I/O ピン

すべての PLL カウンタには 20 のコンフィギュレーション・ビットがあり、すべての遅延エレメントには 4 つのコンフィギュレーション・ビットがあります。カウンタは、プリスケールおよびフィードバックまたはポスト・スケールを行うことができます。

## プリスケールおよびフィードバック・カウンタ ( $n$ , $m$ )

プリスケール・カウンタ  $n$ 、およびフィードバック・カウンタ  $m$  は、2 種類の分周設定を切り替えることによって拡散スペクトラムを実装できます。これらのカウンタの範囲は 1 ~ 512 です。したがって、標準カウント値と拡散スペクトラム値には、それぞれ 9 ビットずつ合計 18 ビットのコンフィギュレーション・ビットが必要です。2 ビットの追加コンフィギュレーション・ビットを使用して、標準カウンタと拡散カウンタをバイパスする (すなわち 1 分周する) ことができます。拡散スペクトラムを使用する場合、正しく動作させるには、これら 2 ビットのバイパス・ビットを同じ値に設定する必要があります。この設定により、カウンタ・コンフィギュレーション・ビットの総数は 20 ビットになります。拡散スペクトラムを使用しない場合、デバイスは標準カウント値を使用し、拡散スペクトラム・カウント値とバイパス・ビットを無視します。

## ポスト・スケール・カウンタ ( $g$ , $l$ , $e$ )

$g$ 、 $l$ 、および  $e$  ポスト・スケール・カウンタは拡散スペクトラムは実装せず、プログラム可能なデューティ・サイクルを実装します。各カウンタには9ビットの High 時間設定と9ビットの Low 時間設定があります。デューティ・サイクルは、出力 High 時間と Low 時間の2つを合計した全サイクル時間に対するいずれか一方の時間の比率です。加えて、これらのカウンタには2つのコントロール・ビット (RBYPASS と RSELODD) があり、コンフィギュレーション・ビットの総数は20ビットになります。

RBYPASS ビットを1に設定すると、カウンタがバイパスされ分周比は1になります。このビットを0に設定すると、VCO 出力周波数の有効分周比を計算するために、High 時間および Low 時間カウンタが追加されます。例えば、ポスト・スケール分周係数が10の場合、High カウント値と Low カウント値はそれぞれ5と5に設定して、50-50% のデューティ・サイクルを達成できます。他方、High カウント値と Low カウント値に4と6を設定すれば、40-60% のデューティ・サイクルで出力クロックが生成されます。

RSELODD ビットは、50% のデューティ・サイクルでの VCO 出力周波数に対する奇数の分周係数を示します。例えば、ポスト・スケール分周係数が3の場合、High および Low 時間カウント値をそれぞれ2と1に設定すれば、この分周を実現できます。これによっても33 (Low 時間) 対 67% (High 時間) のデューティ・サイクルが得られます。ただし、デザインで50-50% のデューティ・サイクルが要求される場合は、RSELODD コントロール・ビットを1に設定すれば、奇数分周比に対してこのデューティ・サイクルを実現できます。PLL は VCO 出力クロックの立ち下がりエッジで出力クロックを High から Low に遷移させることによって、このデューティ・サイクルを実装します。

## 遅延エレメント

コンフィギュレーション可能なすべての遅延エレメントは同一であり、それらの設定値を表 2 に示します。遅延エレメントは 250ps 単位で最大 3ns までの遅延を追加できます。Stratix enhanced PLL 内部のすべてのカウンタには、関連する遅延エレメントがあります。例えば、 $\Delta t_m$  の遅延エレメントは、フィードバック・カウンタに関連付けられ、すべての PLL クロック出力を進める（実質的に負の遅延を追加する）のに使用できます。

遅延エレメントのビット設定				遅延 (ns) (1)
3	2	1	0	
0	0	0	0	0.00
0	0	0	1	0.25
0	0	1	0	0.50
0	0	1	1	0.75
0	1	0	0	1.00
0	1	0	1	1.25
0	1	1	0	1.50
0	1	1	1	1.75
1	0	0	0	2.00
1	0	0	1	2.25
1	0	1	0	2.50
1	0	1	1	2.75
1	1	(2)	(2)	3.00

### 表 2 の注 :

- (1) 実際の遅延は評価中であり、デバイスのスピード・グレードによって変動します。
- (2) このエントリの設定値は無関係です。

## スキャン・チェーンの順序

シフト・レジスタの長さは PLL ごとに異なります。enhanced PLL 5 および 6 には 12 個のカウンタと遅延エレメント、および 1 個の 289 ビットのシフト・レジスタがあります。シフト・レジスタ内の最後のレジスタは転送イネーブル・ビットを保持します。図 2 にこれらの PLL 内の PLL コンポーネントにおけるシフト・レジスタの順序を示します。enhanced PLL 11 および 12 にあるのは 8 個のカウンタと遅延エレメント、および 1 個の 193 ビットのシフト・レジスタです。図 3 にこれらの PLL 内のコンポーネントにおけるシフト・レジスタの順序を示します。カウンタ設定用のスキャン・レジスタには、 $n$ 、 $m$ 、 $g_0$ 、および  $g_1$  のマークが付けられており、また遅延エレメント用スキャン・レジスタには  $\Delta t_m$ 、 $\Delta t_n$ 、 $\Delta t_{g_0}$ 、および  $\Delta t_{g_1}$  のマークが付けられています。チェーン内のすべてのレジスタで、MSB が最初にシフトされ、LSB が最後にシフトされます。

図 2. enhanced PLL 5 および 6 のシフト・レジスタの順序

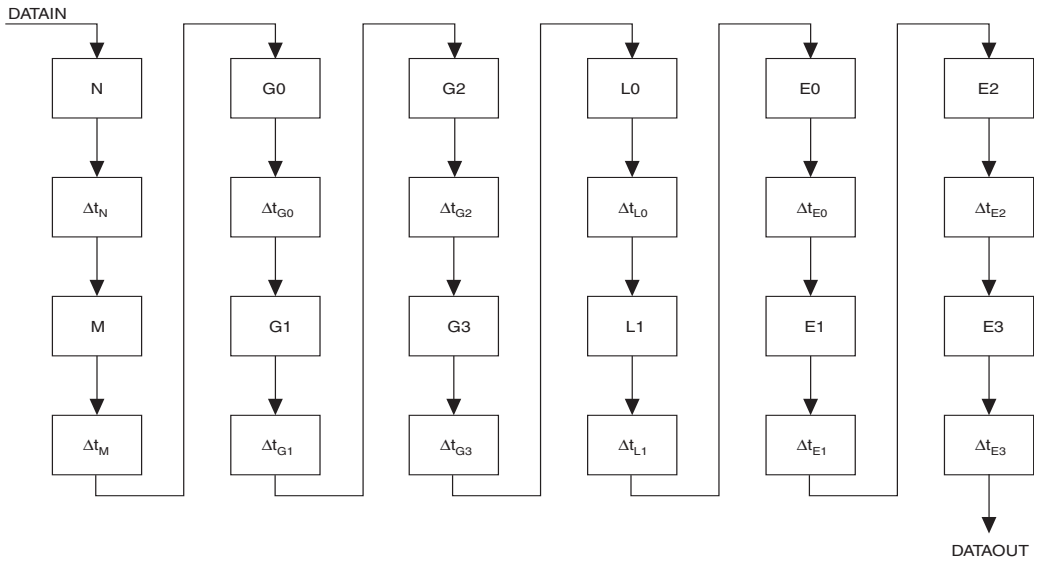
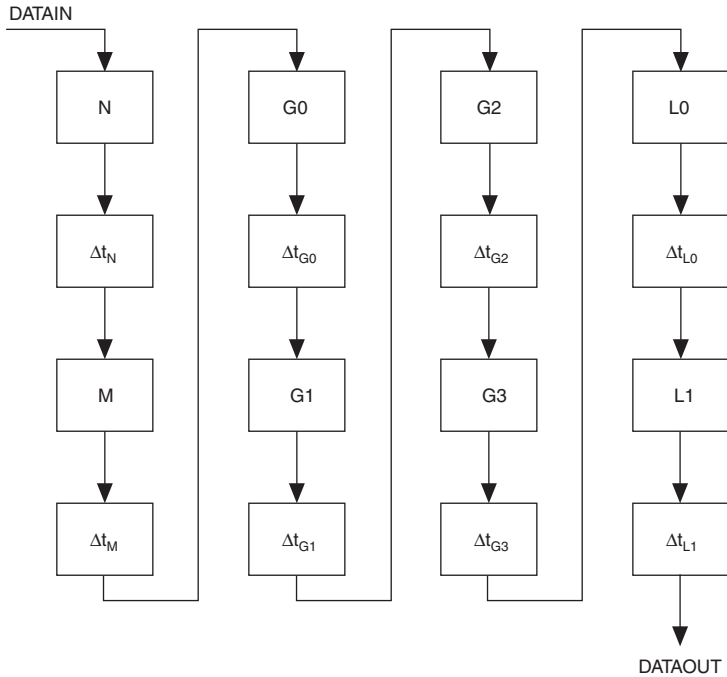


図 3. enhanced PLL 11 および 12 のシフト・レジスタの順序



## PLL カウンタのバイパス

PLL カウンタをバイパスすると、乗算 ( $m$  カウンタ) または除算 ( $n$ ,  $g$ ,  $l$ , または  $e$  カウンタ) 係数は 1 になります。

$m$  または  $n$  カウンタをバイパスする唯一の方法は、該当するカウンタのバイパス・ビットを 1 に設定し、カウンタの LSB を 0 に設定することです。 $m$  および  $n$  カウンタの場合のみ、バイパス・ビットを High に設定し、次にカウント値を 1 (または最下位桁が 1 の任意の値) に設定すると、カウンタがディセーブルされ PLL が動作不能になります。表 3 に  $m$  および  $n$  カウンタの設定を示します。

表 3. $m$ および $n$ カウンタの設定	
ビット [9..0] 設定	説明
1xxxxxxxxx0	PLL カウンタをバイパス
1xxxxxxxxx1	PLL カウンタはディセーブル
0xxxxxxxxxx	PLL カウンタはバイパスもディセーブルもされない (1)
0000000000	PLL カウンタはカウント値 512 に設定される

### 表 3 の注:

- (1) PLL カウンタはバイパス・ビット (ビット 9) が 0 に設定されているので、バイパスもディセーブルもされません。



カウンタの LSB は右側です。

$m$  または  $n$  を 1 に設定すると、Quartus II ソフトウェアは常に標準カウント設定 (000000000) を選択し、バイパス・ビットを 1 に設定します (上記の標準カウント値は LSB を右側にして表示されています)。

10 個の出力カウンタ ( $g$ ,  $l$ , または  $e$ ) のいずれかをバイパスするように設定するには、バイパス・ビットを 1 に設定します。他のビットの値は無視されます。

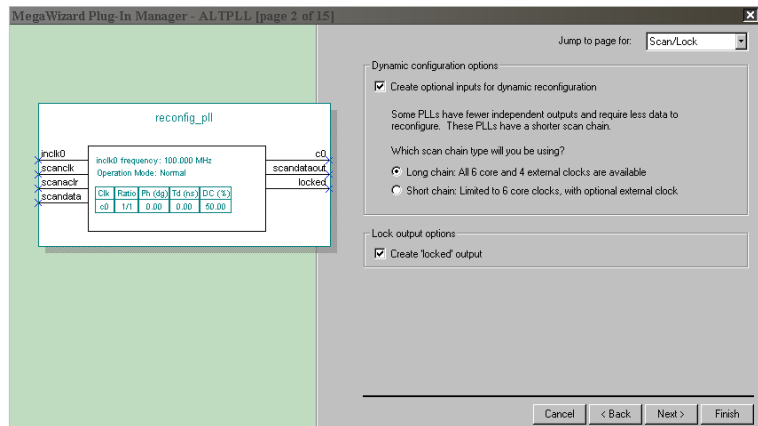
## Quartus II ソフトウェアでリコンフィギュレーション可能な PLL の実装

Quartus II altp11 MegaWizard® Plug-In Manager で、PLL 入力 / 出力周波数および位相 / 時間シフトを指定できます。これらのパラメータに基づいて、Quartus II ソフトウェアは PLL の内部設定を取り込みます。SRAM オブジェクト・ファイル (.sof) は、これらの内部設定を格納し、PLL はコンフィギュレーションおよびパワーアップ後にこれらの設定を使用します。Quartus II ソフトウェアで altp11\_reconfig メガファンクションを使用して、PLL リコンフィギュレーションを実装します。

## altpll メガファンクション

altpll MegaWizard Plug-In Manager を使用して、PLL リンク構成回路をイネーブルにすることができます (図 4 を参照)。enhanced PLL 5 および 6 を使用するとき、これらの PLL には 4 個の外部クロック出力カウンタがあるので、altpll MegaWizard Plug-In の 2 ページで、“Long chain: All 6 core and 4 external clocks are available” を選択します。PLL 11 および 12 を使用するときには、これらの PLL には 4 つの外部クロック出力がないので、“Short chain: Limited to 6 core clocks, with optional external clock” を選択します。リンク構成回路をイネーブルすると、altpll メガファンクションに、scanclk、scandata、scandataout、scandataclr および scandataout ポートが追加されます。

図 4. Quartus II MegaWizard Plug-In Manager での PLL リンク構成のイネーブル

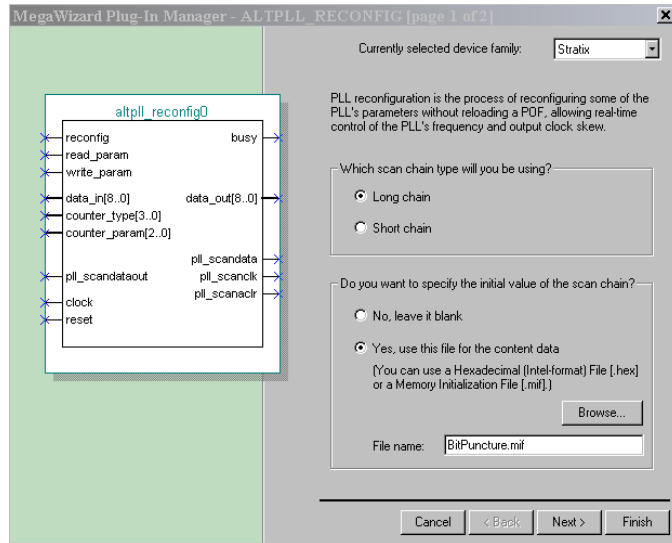


## altpll\_reconfig メガファンクション

altpll\_reconfig メガファンクションは、Stratix または Stratix GX enhanced PLL を動作中にリンク構成するための簡単で効率的な方法を提供します。altpll\_reconfig MegaWizard Plug-In Manager を使用して、Stratix または Stratix GX enhanced PLL をリンク構成することができます。

図 5 に、altpll\_reconfig メガファンクション用の MegaWizard インタフェースを示します。このメガファンクションには、Quartus II ソフトウェアで `libraries\megafuncions\gates\altpll_reconfig` を通じてアクセスすることができます。

図 5. MegaWizard Plug-In Manager インタフェース



ロング・チェーン（PLL 5 および 6、289 ビット）またはショート・チェーン（PLL 11 および 12、193 ビット）を指定することができます。また、scandata ビットを PLL スキャン・レジスタ内に初期設定するためのメモリ初期化ファイル（.mif）または 16 進（.hex）ファイルを指定することができます。

以下のステップを使用して、altpll\_reconfigメガファンクションを実装します。

1. 乗算 / 除算係数を選択して altpll メガファンクションを作成し、リアルタイム・リコンフィギュレーションを使用中として指定します。
2. altpll\_reconfig メガファンクションを作成し、前のステップで作成された altpll メガファンクションに合わせてシフト・レジスタのサイズ（ロング・チェーンまたはショート・チェーン）を指定します。
3. Quartus II ソフトウェアでデザインをコンパイル（または合成）して、PLL シフト・レジスタの初期状態またはデフォルト状態を表す MIF を生成します。
4. PLL リコンフィギュレーション中に変更されるカウンタ / 遅延設定に対応するために MIF を変更するか、または MIF を PLL の初期状態を表すのに使用します。

5. `altpll_reconfig` MegaWizard Plug-In Manager の 1 ページで、**Browse** ボタンをクリックし、修正された MIF を選択します。また、`altpll_reconfig` ファンクション・インスタンスを編集して、前のステップで修正された MIF に `init_scan_file` パラメータを指定します。

Quartus II ソフトウェアは、7 ページの「PLL カウンタのバイパス」に示すとおり、 $m$  および  $n$  カウンタをバイパスするためのガイドラインと同じガイドラインに従います (すなわち、 $m$  および  $n$  カウンタをバイパスするには、9 ビットの標準カウント・ビットは 0 に設定され、バイパス・ビットは 1 に設定されます)。カウンタ設定をリコンフィギュレーションするために MIF を修正する際は注意してください。無効な設定 (バイパス・ビット = 1、 $m$  および  $n$  カウンタの LSB = 1) を行うと、リコンフィギュレーション後に PLL カウンタがディセーブルされます。

`altpll_reconfig` メガファンクションは、PLL を適切にリコンフィギュレーションするためのすべての制御動作を管理するステート・マシンを備えています。これらの動作には、289 番目の立ち下がりエッジでの `scanclk` のスイッチ・オフ、`scanclk` 信号の立ち下がりエッジでの `scandata` のクロック・インが含まれ、`scanackr` ポートでの初期クリアを提供し、以降のリコンフィギュレーションの間にスキャン・レジスタをクリアします。`altpll_reconfig` メガファンクションによって、PLL のリコンフィギュレーションのプロセスがはるかに簡単になります。

## MIF の生成

コンパイラはリコンフィギュレーションをイネーブルするようにコンフィギュレーションされた PLL に対して、スキャン・レジスタの初期状態を表す MIF を生成します。次にこの MIF は、`altpll_reconfig` メガファンクションのスキャン・チェーン・キャッシュ (すなわち、`init_mif_file` パラメータ) を初期化できます。

MIF は PLL がリコンフィギュレーションを使用する場合のみ生成され、`scanclk` ポートが接続されているかどうかによって決定されます。生成されたファイルの名前は PLL のインスタンス名に由来して生成され、PLL サマリ・セクションにあるコンパイル・レポートで見つけることができます。

## ポートおよびパラメータ

以下のセクションで、`altpll_reconfig` メガファクションのパラメータ、入力ポート、および出力ポートを説明します。表 4 にポート・パラメータを示します。表 5 に入力ポートを示します。表 6 に `counter_type[3..0]` ポートの仕様、表 7 に `counter_param[2..0]` の仕様を示します。表 8 に出力ポートを示します。

パラメータ名	説明
SCAN_CHAIN	メガファクションが制御するシフト・レジスタの長さを定義します。このパラメータは、 <code>long</code> または <code>short</code> に設定できます。このパラメータが <code>long</code> に設定された PLL は、4 個の追加外部カウンタと 289 ビットのコンフィギュレーションを備えています。このパラメータが <code>short</code> に設定された PLL は、193 ビットのコンフィギュレーションを備えています。
SCAN_INIT_FILE	シフト・レジスタ・キャッシュの初期値として使用される、 <code>.mif</code> または <code>.hex</code> ファイルの名前。このファイルは、PLL の <code>SCAN_CHAIN</code> の長さに応じて、193 ビットまたは 289 ビットのいずれかになります。データ・ビットのフォーマットは、表 9 に指定されるシフト・レジスタのフォーマットに従ってなければなりません。フォーマットが指定されていない場合は、キャッシュが未知の状態に初期化されることがあります。

ポート名	必須	説明
<code>clock</code>	Yes	個々のパラメータをロードするため、およびリコンフィギュレーション中に PLL をドライブするために使用されるクロック入力。このポートは有効なクロックに接続されていなければなりません。
<code>data_in[]</code>	No	パラメータの書き込み時にパラメータ・データを供給可能な 9 ビット・バス。9 ビットの一部しか使用しないパラメータもあります。この場合、ビット 0 から必要ビット数までのビットのみ使用されます(例えば、遅延エLEMENT値はビット [3..0] を使用し、ビット [8..4] は無視されます)。このバスは未接続状態の場合は、デフォルトで 0 になります。
<code>counter_type[]</code>	No	更新するカウンタ・タイプを選択する 4 ビット・バス。表 6 に、各 <code>counter_type</code> 値に指定されるカウンタを決定するためのマッピングを示します。  2 番目の 2 ビットは、 <code>g</code> 、 <code>l</code> および <code>e</code> カウンタのカウンタ番号を示します。また、外部の [e3..e0] カウンタは、ロング・シフト・レジスタ PLL (PLL 5 と 6) に対してのみ有効です。
<code>counter_param[]</code>	No	更新するカウンタ・タイプを選択する 3 ビット・バス。表 7 に各パラメータ・タイプへのマッピングを示します。対応するパラメータのビット幅は次のように定義されます。  最初の 2 ビットでビット幅を決定します(すなわち、00x=9、01x=4、10x=1)。

表 5. 入力ポート ( 2 / 3 )

ポート名	必須	説明
read_param	No	<p>counter_type[] および counter_param[] で指定されるパラメータがキャッシュから読み出され、dataout[] に供給されなければならないことを示す信号。読み出されて data_out[] に設定されるビット数は、前述のとおりパラメータ・タイプによって決まります。この信号は立ち上がりクロック・エッジでサンプリングされ、この時点でキャッシュからのパラメータ値の読み出しが開始されます。また、この信号は 1 クロック・サイクル間のみアサートし、後続クロック・サイクルでのパラメータの再読み出しを防止しなければなりません。</p> <p>read_param がアサートされて読み出されると直ぐに、busy 信号がアクティブになります。パラメータの読み出し中、busy 信号はアサートされたままです。busy 信号が非アクティブになると、dataout[] が有効になり、次のパラメータのロードを開始できるようになります。ビジーの間、dataout[] は無効です。</p>
write_param	No	<p>counter_type[] および counter_param[] で指定されるパラメータを、data_in[] で指定される値でキャッシュに書き込まなければならないことを示す信号。data_in[] から読み出されるビット数は、前述のとおりパラメータ・タイプによって決まります。この信号は立ち上がりクロック・エッジでサンプリングされ、この時点でキャッシュへのパラメータ値の書き込みが開始されます。また、この信号は 1 クロック・サイクル間のみアサートし、後続クロック・サイクルでのパラメータの再書き込みを防止しなければなりません。</p> <p>write_param が読み出されてアサートされると直ぐに、busy 信号がアクティブになります。パラメータの書き込み中、busy 信号はアサートされたままで、datain[] への入力は無視されます。busy 信号が非アクティブになると、デバイスは次のパラメータの書き込みを開始します。</p>

表 5. 入力ポート ( 3 / 3 )

ポート名	必須	説明
reconfig	Yes	<p>現在のキャッシュに指定された PLL 設定で PLL をリコンフィギュレーションしなければならないことを示す信号。デバイスはこの信号を立ち上がりクロック・エッジでサンプリングし、この時点でキャッシュされた設定の PLL へのロードが開始されます。また、この信号は 1 クロック・サイクル間のみアサートし、リコンフィギュレーションの完了後に PLL がリロードされないようにしなければなりません。</p> <p>reconfig 信号が読み出されると直ぐに、busy 信号がアクティブになります。PLL がリコンフィギュレーションされている間、busy 信号はアサートされたままです。busy 信号が非アクティブになると、以前と同様にパラメータを変更できます。</p> <p>また、リコンフィギュレーション中およびリコンフィギュレーション後に、シフト・レジスタのデータ・キャッシュは変化しません。これによって、1 つのパラメータの変更中に他のすべてのパラメータ値は以前の値で設定されるため、1 回に 1 つのパラメータだけを変更しながら複数回のリコンフィギュレーションが可能になります。</p>
reset	Yes	<p>ステート・マシンを初期化して有効状態にするために使用される非同期リセット入力。ステート・マシンは初回使用前にリセットする必要があります。リセットされていない場合、状態が有効かどうかは保証されません。このポートは接続されていなければなりません。</p>
pll_scandataout	No	<p>altpllメガファンクションのscandataoutポートによってドライブされる入力信号。scandataout 信号は、PLL がリコンフィギュレーションを完了すると High になります。したがって、この信号は PLL のリコンフィギュレーションが完了するまで busy 信号を High のまま保持するために、altpll_reconfig メガファンクションの内部で使用されます。</p>

カウンタの選択	counter_type[3..0]	
	2 進数	16 進数
<i>n</i>	00 00	0
<i>m</i>	00 01	1
(不正値)	00 10	2
(不正値)	00 11	3
<i>g0</i>	01 00	4
<i>g1</i>	01 01	5
<i>g2</i>	01 10	6
<i>g3</i>	01 11	7
<i>h0</i>	10 00	8
<i>h1</i>	10 01	9
(不正値)	10 10	A
(不正値)	10 11	B
<i>e0</i>	11 00	C
<i>e1</i>	11 01	D
<i>e2</i>	11 10	E
<i>e3</i>	11 11	F

表 7. counter_param[2..0] の設定			
カウンタの選択	counter_param[2..0]		幅
	2 進数	16 進数	
標準カウント ( <i>m</i> および <i>n</i> に対応)	000	0	9
拡散カウント ( <i>m</i> および <i>n</i> に対応)	001	1	9
High サイクル・カウント ( <i>g</i> , <i>l</i> , および <i>e</i> に対応)	000	0	9
Low サイクル・カウント ( <i>g</i> , <i>l</i> , および <i>e</i> に対応)	001	1	9
遅延エレメントの設定	010	2	4
(不正値)	011	3	
カウンタ・バイパス・ビット	100	4	1
カウンタ奇数分周ビット (スペクトラム拡散を使用しない場 合、 <i>m</i> および <i>n</i> は 0 であること)	101	5	1
拡散カウント・バイパス・ビット (スペクトラム拡散使用時のみ有 効)	101	5	1
(不正値)	110	6	
(不正値)	111	7	

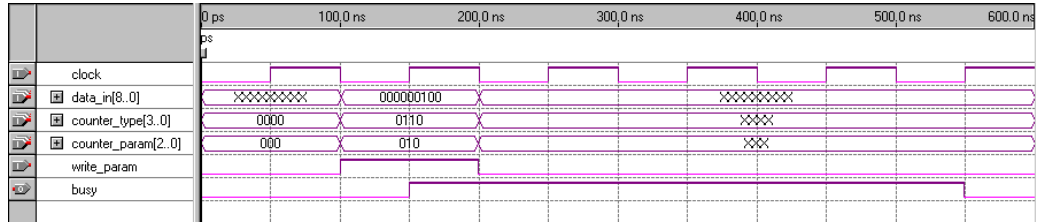
ポート名	必須	説明
data_out[]	No	ユーザがパラメータ・データをリード・バックすることができる9ビット・バス。パラメータ値は、counter_type[] 値と counter_param[] 値を使用し、read_param 信号をアサートすると要求され、その時点でパラメータ値はキャッシュからロードされてこのバス上にドライブされます。このデータは busy 信号がデアサートされると有効になります。
busy	No	この信号は、read_param、write_param、または reconfig のいずれかがアサートされると High になり、特定の動作が完了するまで High になったままです。ステート・マシンはこの信号がアサートされている間は入力を無視し、この信号をデアサートするまでマシンを変更することはできません。これは変更（読み出し/書き込み動作）を行うことができるのは busy 信号がデアサートされている間だけであることを意味します。
pll_scanclk	Yes	リコンフィギュレーションされるPLLのscanclkポートをドライブする信号
pll_scanaclr	Yes	リコンフィギュレーションされるPLLのscanaclrポートをドライブする信号
pll_scandata	Yes	リコンフィギュレーションされるPLLのscandataポートをドライブする信号

### altpll\_reconfig を使用した g2 カウンタ遅延エレメントのリコンフィギュレーション

以下のステップを実行することによって、スキャン・チェーン・キャッシュ内のカウンタ g2 に 1.0ns の遅延設定パラメータを設定します（図 6 を参照）。

- data\_in[] に 0100 を設定して 1.0ns の遅延を指定します。
- counter\_type[] に 0110 を設定して、カウンタ g2 を選択します。
- counter\_param[] に 010 を設定して、遅延設定パラメータを選択します。
- write\_param を 1 クロック・サイクル間アサートして、t = 150ns でシフト・レジスタ・キャッシュへのデータのロードが開始されるようにします。
- t = 150ns のデータ・ロード開始時にステート・マシンによって busy 信号がアサートされ、データ・ロードが正常に完了する t = 550ns の時点まで持続します。

図 6. カウンタ g2 に対する遅延設定パラメータ 1.0ns の設定

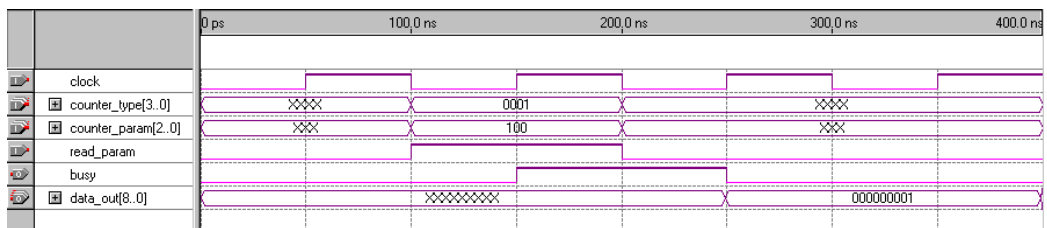


### カウンタ m のカウンタ・バイパス・パラメータ値の読み出し

図 7 に、カウンタ  $m$  のカウンタ・バイパス・パラメータを読み出す方法を示します。カウンタ  $m$  のカウンタ・バイパス・パラメータを読み出すには、以下のステップを実行します。

- counter\_type[] に 0001 を設定して、カウンタ  $m$  を選択します。
- counter\_param[] に 100 を設定して、カウンタ・バイパス・パラメータを選択します。
- read\_param を 1 クロック・サイクル間アサートし、 $t=150\text{ns}$  で data\_out[] レジスタへのデータのロードが開始されるようにします。
- $t=150\text{ns}$  のデータ・ロード開始時にステート・マシンによって busy 信号がアサートされ、データのロードが完了する  $t=250\text{ns}$  の時点まで持続します。ここで、data\_out[] が有効になり、data\_out[0] にカウンタ・バイパス・ビットの値を保持します（この例では、カウンタ・バイパス・ビットには 1 の値がありました）。

図 7. カウンタ m のカウンタ・バイパス・パラメータ値の読み出し

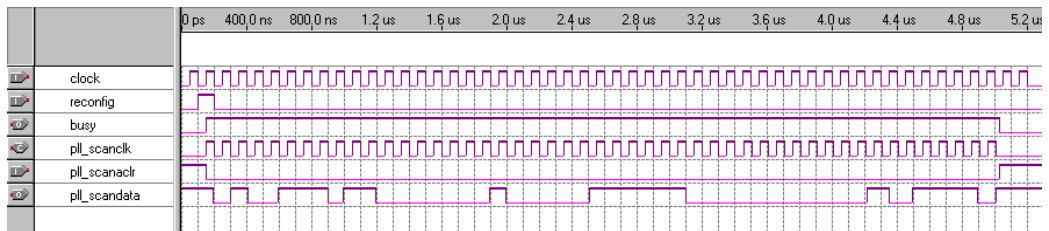



## シフト・レジスタ・キャッシュ内の現在のパラメータによる PLL のリコンフィギュレーション

図 8 に、シフト・レジスタ・キャッシュ内のデータを用いて PLL をリコンフィギュレーションする方法を示します。シフト・レジスタ・キャッシュ内のデータを用いて PLL をリコンフィギュレーションするには、以下のステップを実行します。

- reconfig を 1 クロック・サイクル間アサートして、 $t = 150\text{ns}$  でシフト・レジスタ・キャッシュ内のデータが PLL スキャン・ポートを通して PLL にロード開始されるようにします。
- $t = 0.150\mu\text{s}$  でのデータ・ロード開始時に、ステート・マシンによって busy 信号がアサートされ、PLL がリコンフィギュレーションを終了する時点の  $t = 5.000\mu\text{s}$  まで持続します。
- pll\_scanclock、aclr、および data 信号は、PLL をドライブし、シフト・レジスタ・キャッシュに格納されたパラメータを用いて PLL をリコンフィギュレーションします。図 8 に示す pll\_scandata 波形は単なる例であり、実際のデータを示すものではありません（現実の波形は 193 ビット以上のデータで形成されます）。

図 8. シフト・レジスタ・キャッシュ内の現在のパラメータによる PLL のリコンフィギュレーション



 この例では、reset ポートと pllscandata\_out ポートは使用していません。

## PLL コンフィ ギュレーション・スキャン・ レジスタ・ ビット・マップ

経験豊富な PLL ユーザは、ハードウェア実装セクションで詳述されている情報に基づいて、カウンタ設定および遅延エレメント設定を手動で選択することもできます。異なるカウンタおよび遅延エレメントに対して個別のコンフィギュレーション・ビット設定を決定した後、表 9 に掲載されているとおりビットを配置してください。この表ではシフト・レジスタのビット・マップを示します。ビット 0 はシフト・レジスタにシフトされる最後のビットです。ビット 288 は、enhanced PLL 5 および 6 にシフト・インされる最初のビット、ビット 192 は PLL 11 および 12 の最初のビットです。

表 9. PLL コンフィギュレーション・シフト・レジスタ・ビット・マップ ( 1 / 2 ) 注 (1), (2), (3)

PLL シフト・レジスタ・ビット・マップ									PLL パラメータ	サイズ (ビット)
LSB								MSB		
0	1	2	3	4	5	6	7	8	<i>n</i> カウンタ標準カウント	9
								9	<i>n</i> カウンタ・バイパス・ビット	1
10	11	12	13	14	15	16	17	18	<i>n</i> カウンタ拡散カウント	9
								19	<i>n</i> カウンタ・スペクトラム拡散バイパス・ビット	1
					20	21	22	23	<i>n</i> 遅延エレメント設定	4
24	25	26	27	28	29	30	31	32	<i>m</i> カウンタ標準カウント	9
								33	<i>m</i> カウンタ・バイパス・ビット	1
34	35	36	37	38	39	40	41	42	<i>m</i> カウンタ拡散カウント	9
								43	<i>m</i> カウンタ・スペクトラム拡散バイパス・ビット	1
					44	45	46	47	<i>m</i> 遅延エレメント設定	4
48	49	50	51	52	53	54	55	56	<i>g0</i> カウンタ High サイクル・カウント	9
								57	<i>g0</i> カウンタ・バイパス・ビット	1
58	59	60	61	62	63	64	65	66	<i>g0</i> カウンタ Low サイクル・カウント	9
								67	<i>g0</i> カウンタ奇数分周ビット	1
					68	69	70	71	<i>g0</i> 遅延エレメント設定	4
72	73	74	75	76	77	78	79	80	<i>g1</i> カウンタ High サイクル・カウント	9
								81	<i>g1</i> カウンタ・バイパス・ビット	1
82	83	84	85	86	87	88	89	90	<i>g1</i> カウンタ Low サイクル・カウント	9
								91	<i>g1</i> カウンタ奇数分周ビット	1
					92	93	94	95	<i>g1</i> 遅延エレメント設定	4
96	97	98	99	100	101	102	103	104	<i>g2</i> カウンタ High サイクル・カウント	9
								105	<i>g2</i> カウンタ・バイパス・ビット	1
106	107	108	109	110	111	112	113	114	<i>g2</i> カウンタ Low サイクル・カウント	9
								115	<i>g2</i> カウンタ奇数分周ビット	1
					116	117	118	119	<i>g2</i> 遅延エレメント設定	4
120	121	122	123	124	125	126	127	128	<i>g3</i> カウンタ High サイクル・カウント	9
								129	<i>g3</i> カウンタ・バイパス・ビット	1

PLL シフト・レジスタ・ビット・マップ									PLL パラメータ	サイズ (ビット)	
LSB								MSB			
130	131	132	133	134	135	136	137	138	g3 カウンタ Low サイクル・カウント	9	
								139	g3 カウンタ奇数分周ビット	1	
						140	141	142	143	g3 遅延エレメント設定	4
144	145	146	147	148	149	150	151	152	l0 カウンタ High サイクル・カウント	9	
								153	l0 カウンタ・バイパス・ビット	1	
154	155	156	157	158	159	160	161	162	l0 カウンタ Low サイクル・カウント	9	
								163	l0 カウンタ奇数分周ビット	1	
					164	165	166	167	l0 遅延エレメント設定	4	
168	169	170	171	172	173	174	175	176	l1 カウンタ High サイクル・カウント	9	
								177	l1 カウンタ・バイパス・ビット	1	
178	179	180	181	182	183	184	185	186	l1 カウンタ Low サイクル・カウント	9	
								187	l1 カウンタ奇数分周ビット	1	
					188	189	190	191	l1 遅延エレメント設定	4	
192	193	194	195	196	197	198	199	200	e0 カウンタ High サイクル・カウン ト (4)	9	
								201	e0 カウンタ・バイパス・ビット (4)	1	
202	203	204	205	206	207	208	209	210	e0 カウンタ Low サイクル・カウン ト (4)	9	
								211	e0 カウンタ奇数分周ビット (4)	1	
					212	213	214	215	e0 遅延エレメント設定 (4)	4	
216	217	218	219	220	221	222	223	224	e1 カウンタ High サイクル・カウン ト (4)	9	
								225	e1 カウンタ・バイパス・ビット (4)	1	
226	227	228	229	230	231	232	233	234	e1 カウンタ Low サイクル・カウン ト (4)	9	
								235	e1 カウンタ奇数分周ビット (4)	1	
					236	237	238	239	e1 遅延エレメント設定 (4)	4	
240	241	242	243	244	245	246	247	248	e2 カウンタ High サイクル・カウン ト (4)	9	
								249	e2 カウンタ・バイパス・ビット (4)	1	
250	251	252	253	254	255	256	257	258	e2 カウンタ Low サイクル・カウン ト (4)	9	
								259	e2 カウンタ奇数分周ビット (4)	1	
					260	261	262	263	e2 遅延エレメント設定 (4)	4	
264	265	266	267	268	269	270	271	272	e3 カウンタ High サイクル・カウン ト (4)	9	
								273	e3 カウンタ・バイパス・ビット (4)	1	
274	275	276	277	278	279	280	281	282	e3 カウンタ Low サイクル・カウン ト (4)	9	
								283	e3 カウンタ奇数分周ビット (4)	1	
					284	285	286	287	e3 遅延エレメント設定 (4)	4	
								288	転送イネーブル・レジスタ (4), (5)	1	
									シフト・レジスタ長 (4)	289	

**表 9 の注 :**

- (1) シフト・レジスタにシフト・インされる最初のビットは、転送イネーブル・ビットです。このビットはロジック High に設定しなければなりません。
- (2) チェイン内のすべてのレジスタで、MSB が最初にシフトされ、LSB が最後にシフトされます。
- (3) enhanced PLL 11 および 12 の場合、シフト・レジスタは L1 遅延エレメントの設定で終わり、合計シフト・レジスタ長は 193 (1 ビットの転送イネーブル・レジスタ・ビットを含む) です。
- (4) これらの値はエンハンスド汎用 PLL 用で、8 つの外部出力しかありません (PLL 5 および 6)。
- (5) これは PLL 11 および 12 ではビット 192 です。

## デザインの考慮事項

PLL リコンフィギュレーション機能の以下の点を考慮する必要があります。

### カウンタおよび遅延エレメント

プリスケールおよびフィードバック・カウンタの設定 ( $m$ 、 $n$ ) を変更すると PLL の VCO 周波数が影響を受けるので、場合によっては PLL を基準クロックに再ロックさせる必要があります。遅延エレメントの設定 ( $\Delta t_m$ 、 $\Delta t_n$ ) を変更すると、基準クロックに対する出力クロックの位相関係が変化します。この場合にも PLL の再ロックが必要です。正確な影響はどの程度変化が大きいかにによって異なりますが、変更を行った場合は通常、再同期化が必要です。

$\Delta t_n$  遅延エレメントを追加すると、すべての PLL クロック出力が基準クロックに対して遅延します。この遅延エレメントを使用して、すべての出力を最大 +3.0ns 遅延させることができます。ポスト・スケール遅延エレメントで、さらに +3.0ns の遅延を追加して合計 +6.0ns にすることができます。ただし、どの 2 つの PLL 出力でも最大遅延差は 3.0ns だけです。

$\Delta t_m$  遅延エレメントを追加すると、すべての PLL クロック出力が基準クロックに対して進み、効果的に負の遅延を追加します ( $\Delta t_m$  遅延エレメントはフィードバック・パスにある)。この遅延エレメントを使用して、出力クロックを +3.0ns だけ進めることができます。

ループ・エレメント ( $m$ 、 $n$ 、 $\Delta t_m$ 、 $\Delta t_n$ ) に変更を加えるときは、CLKENA 信号を使用して、FPGA への PLL 出力をディセーブルし、システム・ロジックに影響を与える過周波数状態を除去する必要があります。

ポスト・スケール・カウンタまたは遅延エレメントを変更しても、PLL ロックや VCO 周波数には影響はありません。しかし、遅延エレメントの設定を大きく変更すると (250ps 増分量以上) 出力クロックにグリッチが発生する可能性があります。CLKENA 信号を使用するか、または遅延エレメントの設定を小さな増分量で変更してください。

出力クロック間の位相関係が要因の場合、ARESET 信号を使用して PLL を再同期化することを推奨します。これによってすべての内部 PLL カウンタがリセットされ、ロック・プロセスが再起動されます。PLL が再ロックすると、すべての出力クロックが正しい位相関係になります。PLL のリコンフィギュレーションまたはリセット中は、PLL のクロック出力をディセーブルして、システムの状態が変化しないようにすることができます。

### シフト・レジスタの制御動作

リコンフィギュレーション回路は、ロング・チェーンとショート・チェーンの両方とも同じように動作します。

25 ページの図 9 に、PLL リコンフィギュレーションに使用されるシフト・レジスタのブロック図を示します。クロックを直接制御できるので、データをシフトしないときはチェーンにクロックを供給しないようにする必要があります。これは、チェーン内に初期プログラミングからのデータがあるので、デバイスの初回電源投入時には重要です。チェーンのクリア前にクロックが供給されると、デバイスはチェーンの最終ビットに 1 をクロックする可能性があります。チェーンの最終ビット（転送イネーブル・ビット）に 1 がクロックされると常に、PLL はチェーン内のデータに関係なく自身のリコンフィギュレーションを開始します。最初に、チェーンがクリアされるまで scanclk が High または Low に保持されることを確認してください。最初から scanaclr が High に保持される場合は、問題なく scanaclr をトグルできます。また、チェーンがクリアされた後、scandata が Low の間はチェーンの状態を変化させないで scanclk をトグルすることができます。25 ページの図 10 に、代表的なリコンフィギュレーション波形を示します。

scanclk の 289 番目（または、ショート・チェーン PLL の場合は 193 番目）の立ち上がりエッジによって 289 番目（または 193 番目）のビットがクロック・インされた後、クロックは Low 状態に戻ってから、最後の負エッジ・フリップフロップをクロックしなければなりません。クロックが Low になった後、スキャン・レジスタから PLL 内のすべてのカウンタおよび遅延エレメントにビットを同期的にロードするための静止期間が必要です。t<sub>QUIET</sub> の間、scanclk と scanaclr は Low のままでなければなりません。altpll\_reconfig メガファンクションは、PLL のリコンフィギュレーションを正しく行うのに必要なこれらの動作を行うので、アルテラではこのメガファンクションを使用することを推奨しています。

ロジック・アレイから PLL への scanclk の配線遅延が、ロジック・アレイから PLL への scandata の配線遅延より大きい可能性があるため、正のホールド時間に対してデザインを保護する必要があります。scanclk の立ち上がりエッジから離れて scandata をクロックすると、1/2 サイクルのセットアップ時間と 1/2 サイクルのホールド時間をもたせることによって、正のホールド時間に対する保護を与えることができます。図 10 に scandata のクロッキング方式を示します。

配線に関するもう 1 つの問題が、scanac1r の遅延です。scanac1r の配線遅延は、scancl1k の配線遅延より長い場合も短い場合もあります。このため、scancl1k の立ち上がりエッジに近すぎる位置で scanac1r が High になった場合、どちらが先にチェーンに入るかは不定です。例えば、scanac1r の遅延が scancl1k の遅延より 1ns 長く、scancl1k が scanac1r の 0.5ns 後に立ち上がる場合、scanac1r は scancl1k の立ち上がりエッジをキャンセルします。しかし、scancl1k の遅延が scanac1r の遅延より 1ns 長く、scancl1k が scanac1r の 0.5ns 前に立ち上がる場合、チェーンに意図しない scancl1k の立ち上がりエッジが発生します。これを防止するには、scancl1k の立ち上がりエッジが scanac1r の立ち上がりエッジの 5ns 以内に発生しないようにする必要があります。

以下の手順を使用して、PLL をリコンフィギュレーションします。

1. CLKEN 信号を使用してすべての PLL 出力をディセーブルします。
2. scancl1k を開始する前に 5ns に 1 回 scanac1r をアサートします。
3. scandata ポートを通して、新しいカウンタと遅延エレメントの設定をスキャンします。
4. scancl1k が確実に 289 番目の立ち下がりエッジの後でオフになるようにし、静止期間が終わるまで待機します。
5. 再び scanac1r を 5ns 間アサートし、別のリコンフィギュレーションの前にすべてのスキャン・レジスタをクリアします。
6. ARESET を使用して PLL をリセットし、出力クロック間の位相関係を維持します。
7. 有効なクロックの検出後に PLL 出力を再びイネーブルします。

PLL 出力をディセーブルできない場合は、内部コンポーネントを徐々にかつインクリメンタルに変更します。例えば、基準入力クロックが 100MHz で、 $n$  カウンタが 5、 $m$  カウンタが 25 に設定されている場合、VCO は 500MHz で動作します。VCO 周波数を 600MHz に変更する場合は、 $n$  カウンタを 5 に、 $m$  カウンタを 30 に設定できます。フィードバック・カウンタ ( $m$ ) は 25 から 30 まで増分 1 で徐々に変更する必要があります。これによって過周波数状態（出力周波数が所要周波数より高い）が発生するリスクを低減し、ロック喪失状態を回避します。

PLL リコンフィギュレーション中またはリコンフィギュレーション後に、PLL がロックを喪失する場合は、リコンフィギュレーション・プロセス中に  $m$  および  $n$  カウンタの設定が変更された可能性があります。 $m$  および  $n$  カウンタ / 遅延エレメントの設定を変更すると、PLL がロックを喪失することがあります。例えば、入力クロック周波数が 350MHz、出力クロック周波数が 350MHz の場合、Quartus II ソフトウェアで  $m=2$  および  $n=2$ 、VCO 周波数 350MHz、 $k=1$  に設定すると、この周波数の組み合わせが得られます。

リコンフィギュレーション中にスキャン・ビットを設定して、 $m=n=4$  にしても、入力周波数と出力周波数は同じままです。しかし、この設定では  $m$  カウンタ値と  $n$  カウンタ値を変更する必要があるため、PLL はロックを喪失します。



PLL のリコンフィギュレーション中に誤ってこれらの設定が変更されないようにするための、 $m$  と  $n$  の値について詳しくは、『Quartus II ソフトウェアのコンパイル・レポート』を参照してください。

### リコンフィギュレーション・チェーンの説明

scanclk の最初の立ち上がりエッジで最初のビット、つまり転送イネーブル・ビットがスキャン・レジスタ 0 に登録され、scanclk の 289 番目の立ち上がりエッジで転送イネーブル・ビットがスキャン・レジスタ 288 に登録されます。次に、289 番目のクロックの立ち下がりエッジでこのビットがスキャン・レジスタ 289 に登録されます。転送イネーブル・ビットが登録されると、各カウンタの出力クロックに基づく新しいカウンタ設定がレジスタの並列セットに同期的にロードされます。新しい設定が PLL コンフィギュレーション・ビットにロードされると、reconfig\_done 信号が High になります。この信号は scanackr 信号と AND がとられ、リコンフィギュレーションが正常に実行された後で、転送イネーブル・ビット (289 番目のシフト・レジスタ) をリセットします。Quartus II ソフトウェア・バージョン 2.2 以降では、PLL リコンフィギュレーションがイネーブルされていると、altpll メガファンクションが scandataout ポートを自動的にイネーブルします。scandataout 信号が High になってから Low になり、PLL が新しい設定でリコンフィギュレーションされたことを通知します。さらに、PLL リコンフィギュレーション中に適切なビットがスキャンされてチェーンに送られたことを確認するには、以下に概説する手順に従います。

1. ARESET ポートを使用して PLL をリセットします。
2. scanclk をイネーブルし、scandataout ポートを通して、シフト・レジスタの内容を 1 ビットずつ読み出します。このポートは、Quartus II ソフトウェア・バージョン 2.2 で PLL リコンフィギュレーション機能を選択すると、自動的にイネーブルされます。

このため、PLL の scandata ポートに送られるスキャン・データ・ビット・ストリームにエラーやビット順序の誤りがないかチェックできます。

図 9. リンコンフィギュレーション・チェーンのブロック図 注(1)

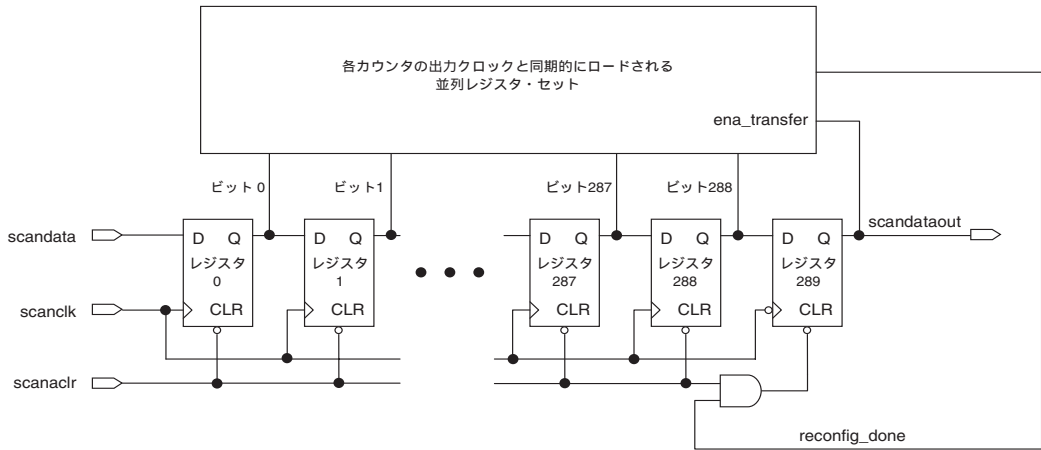


図 9 の注 :

- (1) この図は enhanced PLL (5、6) 用のロング・チェーンを示します。ショート・チェーン PLL (11、12) の場合は、287、288、289 を 191、192、193 に置き換えます。

図 10. EPLL リンコンフィギュレーション波形

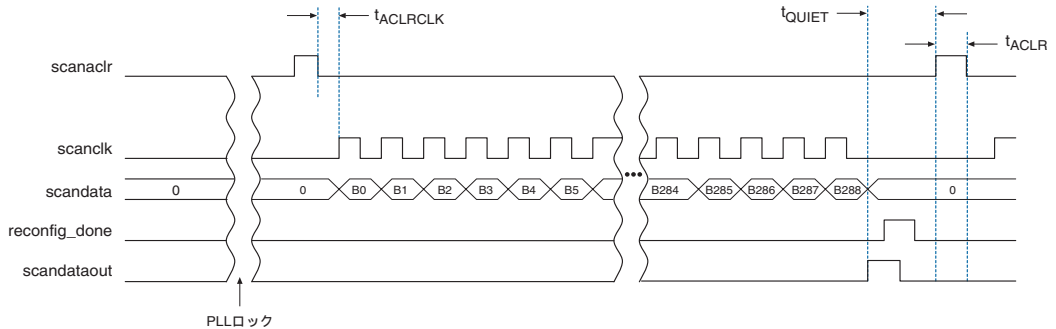



表 10 に PLL パラメータを示します。

表 10. PLL パラメータ	
パラメータ	値
$t_{MAX}$	scanclk 周波数は最大 100MHz が可能です。
$t_{ACLR}$	アルテラはスキャン・レジスタのリセットに最小 5ns の High パルス幅を推奨しています。
$t_{ACLRCLK}$	scanacclr の立ち下がりエッジから scanclk の立ち上がりエッジまでの時間。アルテラの推奨値は最小 5ns です。
$t_{QUIET}$	$t_{QUIET}$ の間は、scanclk と scanacclr がトグルしてはなりません。PLL コンフィギュレーション・ビットが新しいカウンタ設定に更新されるので、これらを両方とも Low に保持する必要があります。アルテラは、この期間を最小 50ns または最も遅い PLL カウンタのクロック周期の 2 倍のいずれか長い方に設定することを推奨しています。

## デザイン・アプローチ

PLL リコンフィギュレーション機能を使用するためのシフト・レジスタの設定方法は、以下を含めて多数あります。

- M512 ブロックのスキャン・ビット
- ロジック・エレメント (LE) のシリアル・シフト・レジスタ・チェーン
- altpll\_reconfig メガファンクション

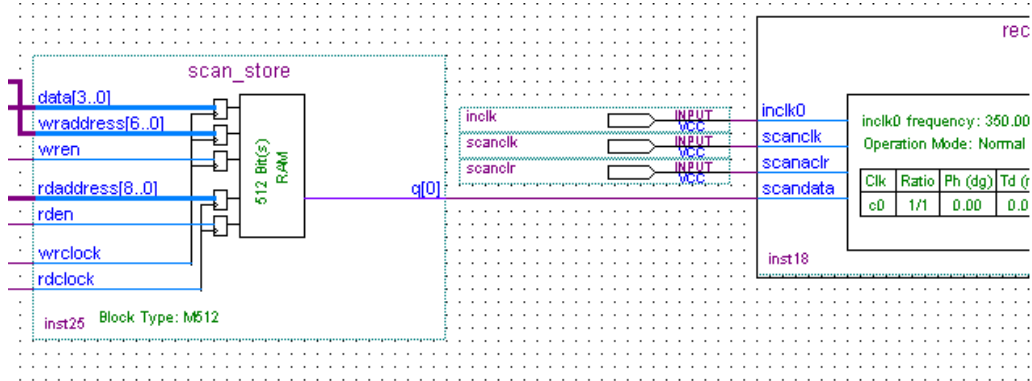
 アルテラは、シフト・レジスタの設定に altpll\_reconfig メガファンクションを使用することを推奨しています。

## メモリ内のスキャン・ビット

図 11 は、M512 メモリ・ブロック内にスキャン・ビットが置かれ、これらが転送イネーブル・ビットであるビット 288 から読み出されるデザインを示しています。

図 11 では、PLL カウンタに対応する遅延エレメント・ビットが 4 ビット幅なので、書き込みデータ・ポートは 4 ビット幅となっています。これらのビットは PLL の scandata ポートにシリアルに読み出されるので、読み出しアドレスは 9 ビット幅であり、デバイスは M512 メモリ・ブロックから 289 ビットを読み出すことができます。PLL への scanclk および scanacclr 入力は、ピンまたは内部ロジックから供給できます。入力がピンから供給される場合は、インテリジェント・コントローラ (マイクロプロセッサなど) が、scanclk の 289 番目の立ち下がりエッジの後に、scanclk をオフにするといった動作を制御することが必要です。詳しくは、21 ページの「デザインの考慮事項」を参照してください。

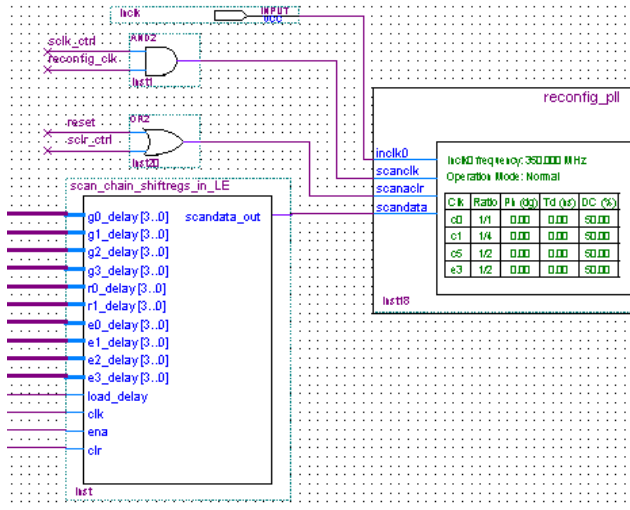
図 11. M512 メモリ・ブロックから読み出されるスキャン・データ・ビット



## LE でのシリアル・シフト・レジスタ・チェーン

scandata PLL ポートにビットをシフトするシリコン上の専用スキャン・レジスタをエミュレートする LE に、独自のシリアル・シフト・レジスタ・チェーンを構築することができます。このアプローチに従ったデザイン例を示します。図 12 に、サブ・デザイン (scan\_chain\_shiftregs\_in\_le) において LE のバンクから供給されるスキャン・データ・ビットを示します。289 番目の立ち下がりエッジの後でスキャン・クロックをオフにしたり、あるいは PLL リコンフィギュレーションの前で scanclr をアサートするカスタム・ロジックを構築することができます。

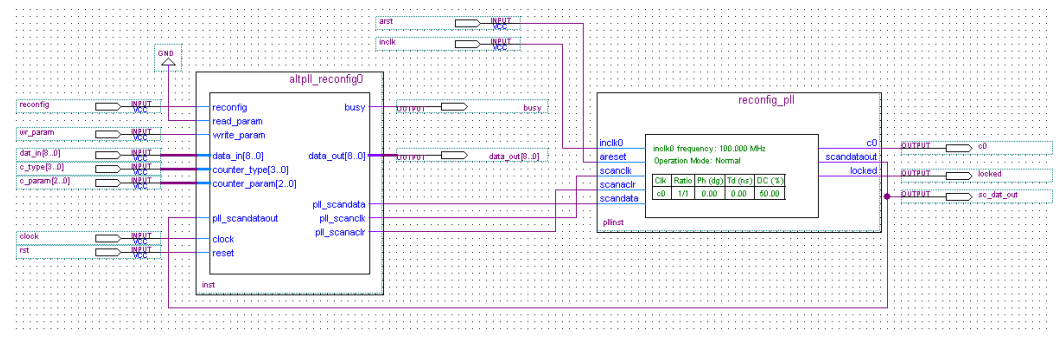
図 12. LE を使用して構築されたシフト・レジスタ・チェーンから供給されるスキャン・データ



## altpll\_reconfig メガファンクション

altpll\_reconfig メガファンクションは、PLLパラメータのリコンフィギュレーションを実装するインタフェースを提供します。このメガファンクションを使用して、PLL タイミングをコンフィギュレーションし、各パラメータを個別に変更することができます。さらに標準 MIF または HEX ファイルを使用してデフォルト・コンフィギュレーションを指定することも可能です。図 13 にこのメガファンクションを PLL に接続する方法を示します。

図 13. PLL に接続するメガファンクション



詳しくは、「例 3: altpll\_reconfig デザイン」を参照してください。

## まとめ

PLL リコンフィギュレーションは、システム設計者が動作中にクロック出力周波数とクロック遅延を変更するのに使用できる強力な機能です。PLL カウンタ設定および遅延エレメント設定を選択するときは、PLL のロック喪失、グリッチ、および位相関係などを十分に考慮する必要があります。アルテラは、PLL パラメータを徐々に変更するか、1 ステップの大幅な変更をインクリメンタルに行うことを推奨しています。また、PLL リコンフィギュレーション時間も標準 20 $\mu$ s 未満であり、動作モードを素早く切り替えることができます。Stratix PLL の柔軟性により、卓越したクロック管理システムが実現します。

## デザイン例

このセクションでは、デザインにリコンフィギュレーション可能な PLL を実装する例を示しています。

### 例 1: LE でのシフト・レジスタ

デザインを解凍し、Quartus II ソフトウェアでコンパイルします。デザインは、`rst` 信号をトグルすると (1  $\rightarrow$  0)、G0 カウンタがリコンフィギュレーションされて 250ps のシフトが生成されるように構成されています。`rst` 信号をプッシュ・ボタンに接続して、ボタンを何回かプッシュすると、0 ~ 3000ps までの範囲で 250ps のジャンプをインクリメンタルに行うことができます。

### 例 2: MIF での `altpll_reconfig` デザイン

デザインを解凍し、Quartus II ソフトウェアでコンパイルします。MIF は、g0 カウンタを 3 分周の初期設定から 6 分周にリコンフィギュレーションするように設定されます。これにより、出力周波数が効果的に 100MHz から 50MHz に変更されます。その他の設定またはカウンタをリコンフィギュレーションするには、最初に `altpll` MegaWizard Plug-In Manager でリコンフィギュレーションする適切なカウンタをイネーブルし、次に MIF 内の対応するスキャン・ビット (表 3 に示す) を変更します。

### 例 3: `altpll_reconfig` デザイン

デザインを解凍し、Quartus II ソフトウェアでコンパイルします。ベクタ波形ファイル (.vwf) は、g0 カウンタを最初に 4 分周して出力周波数を 75MHz にし、次に 6 分周して出力周波数を 50MHz にするように設定されます。このカウンタの初期設定は 3 分周で出力クロックは 100MHz です。その他の設定またはカウンタをリコンフィギュレーションするには、最初に `altpll` MegaWizard Plug-In Manager でリコンフィギュレーションする適切なカウンタをイネーブルし、次に対応する書き込みパラメータ (表 8 を参照) を設定します。



この例では、G0 の High および Low カウンタ値はそれぞれ 2 と 4 です。VCO は 300MHz に設定されているので、出力クロック周波数は  $300/(2+4) = 50\text{MHz}$  となります。しかし、high=2 および low=4 なので、50MHz の出力クロックのデューティ・サイクルは  $2/6$  と  $4/6$  (すなわち、33.33 対 66.66%) になります。50MHz の出力クロックを 50% のデューティ・サイクルで得るには、G0 カウンタで High を 3、Low を 3、そして RSELODD ビットを 1 に設定します (RSELODD の説明は、4 ページの「ポスト・スケール・カウンタ (g, l, e)」を参照してください)。