

はじめに

MachXO2™ デバイスは、ディスプレイ・インターフェイス (7:1 LVDS) やメモリ・インターフェイス (LPDDR、DDR、DDR2) などの様々な I/O インターフェイスに対応できます。これらのインターフェイスを使用するアプリケーションに対応するため、MachXO2 デバイスのアーキテクチャは、大規模 FPGA で一般的に使用される高度なクロッキング機能を含めた設計になっています。設計者はこれらの機能を利用して、クロック生成、クロックスキューの最小化、性能の改善、及び電力消費の管理ができます。

本テクニカルノートでは、MachXO2 デバイス内で使用できるクロックリソースについて説明します。MachXO2 デバイスで使用可能なプライマリクロックやエッジクロック、クロック分周器、sysCLOCK™ PLL、DCC (Dynamic Clock Control) エlement、セカンダリ高ファンアウトネット、及び内蔵オシレータについては詳細を記述します。

各 MachXO2 デバイスの PLL、エッジクロック、及びクロック分周器の数を表 13-1 に示します。表 13-1.

表 13-1. PLL、エッジクロック、及びクロック分周器の数

パラメータ	説明	XO2-256	XO2-640	XO2-640U	XO2-1200	XO2-1200U	XO2-2000	XO2-2000U	XO2-4000	XO2-7000
PLL 数	汎用 PLL	0	0	1	1	1	1	2	2	2
エッジクロック数	高速動作用途用エッジクロック	0	0	4	4	4	4	4	4	4
クロック分周器数	DDR 用途用クロック分周器	0	0	4	4	4	4	4	4	4

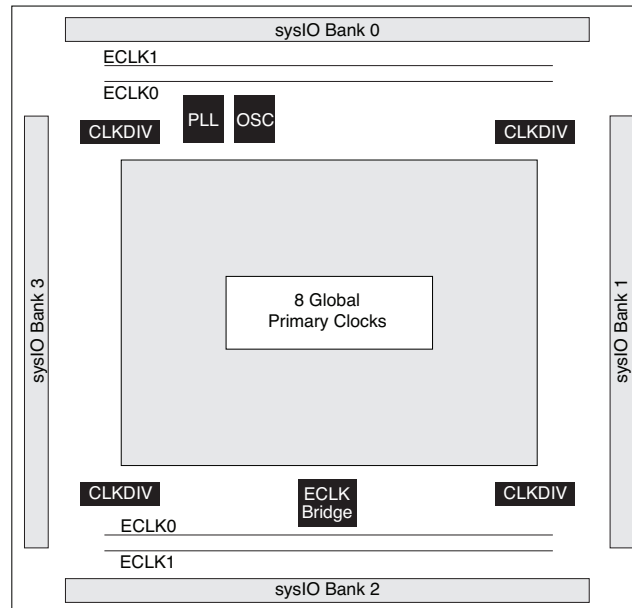
クロック / 制御分配ネットワーク

MachXO2 デバイスは、8 本のグローバル・プライマリクロックと 8 本のセカンダリ高ファンアウトネットの形式で、グローバルクロックの分配を提供します。MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスの上辺と下辺には、2 つのエッジクロックがあります。他のクロックソースとしては、クロック入力ピンと内部ノード、PLL、クロック分周器、及び内蔵オシレータがあります。

MachXO2 トップレベル・ビュー

デバイスの主なクロックリソースのトップレベル・ビューを図 13-1 に示します。

図 13-1. MachXO2 のクロック構造 (MachXO2-1200)



プライマリクロック

MachXO2 デバイスには 8 本のグローバル・プライマリクロックがあります。プライマリクロック・ネットワークは、チップ内の高ファンアウト信号用に低スキューのクロック分配パスです。2 本のプライマリクロックには、DCMA (Dynamic Clock Mux、ダイナミッククロック・マルチプレクサ) 機能があり、2 系統のクロックソースを切り替えることができます。

プライマリクロックのソースには、以下があります。

- ・ 専用クロックピン
- ・ PLL 出力
- ・ CLKDIV 出力
- ・ 内部ノード

DCMA (Dynamic Clock Mux)

MachXO2 デバイスには、2 つの DCMA (ダイナミッククロック・マルチプレクサ) があり、2 系統の独立したプライマリクロック信号をダイナミックに切り替える設計が可能です。DCMA の出力はプライマリクロック分配ネットワークに供給されます。DCMA への入力は、プライマリクロック・ネットワークで使用可能なクロックソースのいずれも可能です。

DCMA はマルチプレクサ機能を備えた単純なクロックバッファです。切り替え時にクロック信号が同期されないため、グリッチが発生する可能性があります。

DCMA プリミティブ定義

プリミティブは、本セクションの定義に沿ってデザインのソースコードでインスタンス化できます。図 13-2 と表 13-2 に DCMA の定義を示します。

図 13-2. DCMA プリミティブ・シンボル

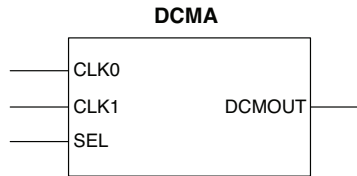


表 13-2. DCMA プリミティブのポート定義

ポート名	I/O	記 述
CLK0	I	クロック入力ポート 0 ～ デフォルト
CLK1	I	クロック入力ポート 1
SEL	I	ポート選択 - SEL=0 で CLK0 - SEL=1 で CLK1
DCMOUT	O	クロック出力ポート

VHDL ソースコードでの DCMA の宣言

ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネント宣言

```
component DCMA
port
  CLK0 : in std_logic;
  CLK1 : in std_logic;
  SEL  : in std_logic;
  DCMOUT : out std_logic;
end component;
```

DCMA インスタンス化

```
I1: DCMA
port map (CLK0 => CLK0,
  CLK1 => CLK1,
  SEL => SEL,
  DCMOUT => DCMOUT);
```

Verilog ソースコードでの DCMA の使用法

コンポーネント宣言

```
module DCMA (CLK0, CLK1, SEL, DCMOUT);

input CLK0;
input CLK1;
input SEL;
output DCMOUT;

endmodule
```

DCMA インスタンス化

```
DCMA I1 (.CLK0 (CLK0);
         .CLK1 (CLK1);
         .SEL (SEL);
         .DCMOUT (DCMOUT));
```

DCCA (Dynamic Clock Control)

MachXO2 デバイスには、各プライマリクロック・ネットワークに使用できる DCCA（ダイナミッククロック制御）機能があります。必要に応じて、DCCA では各プライマリクロックをコアロジックからディセーブルできます。これにより、クロック及びそれに対応するロジックが動作不要な場合にデザイン内で動作停止され、節電できます。

DCCA プリミティブ定義

DCCA プリミティブは、このセクションの定義に従ってデザインのソースコードでインスタンス化できます。図 13-3 と表 13-3 に DCCA の定義を示します。

図 13-3. DCCA プリミティブ・シンボル

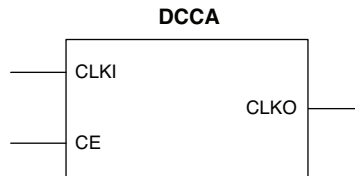


表 13-3. DCCA プリミティブのポート定義

ポート名	I/O	記 述
CLKI	I	クロック入力ポート
CE	I	クロックイネーブルポート - CE = 0 ~ ディセーブル - CE = 1 ~ イネーブル
CLKO	O	クロック出力ポート

VHDL ソースコードでの DCCA の宣言

ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネント宣言

```
component DCCA
  port (CLKI      : in std_logic;
        CE        : in std_logic;
        CLKO      : out std_logic);
end component;
```

DCCA インスタンス化

```
I1: DCCA
  port map ( CLKI => CLKI;
             CE   => CE;
             CLKO => CLKO);
end component;
```

Verilog ソースコードでの DCCA の使用法

コンポーネント宣言

```
module DCCA (CLKI, CE, CLKO);  
  
    input CLKI;  
    input CE;  
    output CLKO;  
  
endmodule
```

DCCA インスタンス化

```
DCCA I1 (.CLKI (CLKI),  
         .CE    (CE),  
         .CLKO  (CLKO));
```

エッジクロック

MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスの上辺と下辺には、2つのエッジクロック・リソースがあります。注入時間が短く、低スキューのこれらのエッジクロックは、I/O レジスタのクロックに使用されます。エッジクロック・リソースは、高いファンアウト能力を持つ高速 I/O インターフェイス用に設計されています。ECLK の位置と接続の詳細については、付録 B を参照してください。

エッジクロックのソースには以下があります（日本語訳註：エッジクロックに載せるためには後述の ECLKSYNCA プリミティブを明示的にインスタンスする必要があります。付録 B の図も参照）。

- ・ 専用クロックピン
- ・ PLL 出力
- ・ 内部ノード

エッジクロック・ブリッジ

MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスには、デバイス内の ECLK 間の通信強化に使用されるエッジクロック・ブリッジも装備されています。このブリッジにより、デバイス下辺への入力デバイス上辺のエッジクロックを、最小限のスキューでドライブできます。エッジクロック・ソースは、エッジクロック・ブリッジ経由でエッジクロックに接続することも、最短経路を使用して直接接続することもできます。

エッジクロック・ブリッジの主な使用目的は、DDR または 7:1 LVDS ビデオなどの高速データインターフェイスです。エッジクロック・ブリッジの使用法の詳細については、“TN1203, Implementing High-Speed Interfaces with MachXO2 Devices (MachXO2 デバイスによる高速インターフェイスの実装)” を参照してください。

エッジクロック・ブリッジにはクロック選択マルチプレクサがあり、2 系統の異なるクロックソースをエッジクロックごとにデザインで切り替えることができます。このクロック選択マルチプレクサは、ECLKBRIDGECS プリミティブを使用して設計されています。エッジクロック・ブリッジのブロック図は付録 B に示してあります。

ECLKBRIDGECS プリミティブ定義

ECLKBRIDGECS プリミティブは、このセクションの定義に沿ってデザインのソースコードでインスタンス化できます。デザインには、必要に応じて ECLKBRIDGECS プリミティブのインスタンス化を 2 つまで含めることができます。図 13-4 と表 13-4 に ECLKBRIDGECS の定義を示します。

図 13-4. ECLKBRIDGECS プリミティブ・シンボル

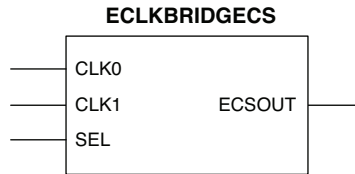


表 13-4. ECLKBRIDGECS プリミティブのポート定義

ポート名	I/O	記 述
CLK0	I	クロック入力ポート 0 ～ デフォルト
CLK1	I	クロック入力ポート 1
SEL	I	ポート選択 - SEL=0 で CLK0 - SEL=1 で CLK1
ECSOUT	O	クロック出力ポート

VHDL ソースコードでの ECLKBRIDGECS の宣言 ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネント宣言

```
component ECLKBRIDGECS
  port ( CLK0      :    in std_logic;
         CLK1      :    in std_logic;
         SEL       :    in std_logic;
         ECSOUT    :    out std_logic);
end component;
```

ECLKBRIDGECS インスタンス化

```
I1: ECLKBRIDGECS
  port map ( CLK0    => CLK0;
             CLK1    => CLK1;
             SEL     => SEL;
             ECSOUT  => ECSOUT);
```

Verilog ソースコードでの ECLKBRIDGECS の使用法 コンポーネント宣言

```
module ECLKBRIDGECS (CLK0, CLK1, SEL, ECSOUT);

  input CLK0;
  input CLK1;
  input SEL;
  output ECSOUT;

endmodule
```

ECLKBRIDGECS インスタンス化

```
ECLKBRIDGECS I1 ( .CLK0 (CLK0),
                  .CLK1 (CLK1),
                  .SEL   (SEL),
                  .ECSOUT (ECSOUT));
```

ECLKSYNCA (Edge Clock Synchronization)

MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスには、ECLKSYNCA (エッジクロック同期制御) も集積されています。この機能により、必要に応じて各エッジクロックをコアロジックからディセーブルにできます。設計者はこの機能を使用して、必要な場合にエッジクロックをイベントまたは外部信号と同期できます。また、クロック及びそれに対応するロジックをダイナミックにディセーブルにし、節電アプリケーションを設計することもできます。^{注1}

ECLKSYNCA プリミティブ定義

ECLKSYNCA プリミティブは、このセクションの定義に従ってデザインのソースコードでインスタンス化できます。図 13-5 と表 13-3 に ECLKSYNCA の定義を示します。

図 13-5. ECLKSYNCA プリミティブ・シンボル



表 13-5. ECLKSYNCA プリミティブのポート定義

ポート名	I/O	記 述
ECLKI	I	クロック入力ポート
STOP	I	エッジクロックを停止する制御信号 STOP=0、クロックがアクティブ STOP=1、クロックがオフ
ECLKO	O	クロック出力ポート

VHDL ソースコードでの ECLKSYNCA の宣言

ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネント宣言

```
component ECLKSYNCA
port
    ( ECLKI  :in std_logic;
      STOP   :in std_logic;
      ECLKO  :out std_logic);
end component;
```

1. 英語版にある R1 に関する記述は省略

ECLKSYNCA インスタンス化

```
I1: ECLKSYNCA
port map (  ECLKI => ECLKI,
            STOP  => STOP,
            ECLKO => ECLKO);
```

Verilog ソースコードでの ECLKSYNCA の使用法

コンポーネント宣言

```
module ECLKSYNCA (ECLKI, STOP, ECLKO);

input ECLKI;
input STOP;
output ECLKO;

endmodule
```

ECLKSYNCA インスタンス化

```
ECLKSYNCA I1      (.ECLKI (ECLKI);
                   .STOP  (STOP);
                   .ECLKO (ECLKO));
```

セカンダリ高ファンアウトネット

MachXO2 デバイスにはクロックや制御、または高ファンアウトデータの各信号に使用できる、8 本のセカンダリ高ファンアウトネットがあります。これらのセカンダリネットは、一般配線リソースよりも注入遅延とスキューが小さい、柔軟なグローバル・クロックリソースです。セカンダリネットは主に、クロックイネーブル (CE)、ローカルセット / リセット (LSR)、及びリセット (RST) アプリケーションなどのグローバル高ファンアウト制御信号用です。8 本のセカンダリネットは、独立した制御入力またはクロック入力を 4 本、独立したデータ入力を 4 本持つことができます。

セカンダリネットのソースには以下があります。

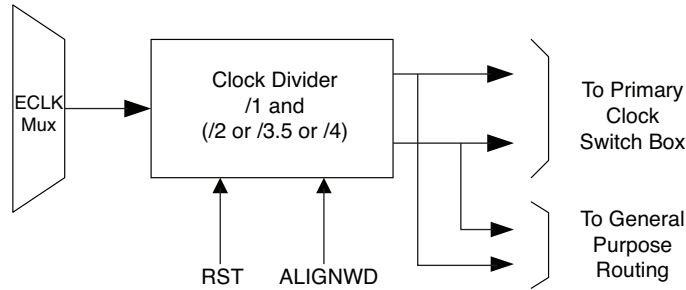
- ・ 専用クロックピン
- ・ 内部ノード

クロック分周器 (CLKDIVC)

MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスには、4 つのクロック分周器があります。MachXO2-256 及び MachXO2-640 デバイスには、クロック分周器はありません。クロック分周器には 2 本の出力があります。1 本は入力クロックと同じ周波数で、もう 1 本は入力クロックを 2、3.5、または 4 分周したものです。どちらの出力も、入出力間遅延は同一です。クロック分周器への入力エッジクロック・マルチプレクサからの出力です。クロック分周器の出力はプライマリクロック・ネットワークをドライブし、汎用配線またはセカンダリクロックにも使用できます。

図 13-6 にクロック分周器のブロック図を示します。

図 13-6. MachXO2 のクロック分周器



CLKDIVC プリミティブ定義

CLKDIVC プリミティブは、このセクションの定義に沿ってデザインのソースコードでインスタンス化できます。図 13-7 と Tables 13-6 及び 13-7 に CLKDIVC の定義を示します。

図 13-7. CLKDIVC プリミティブ・シンボル

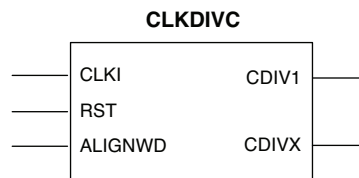


表 13-6. CLKDIVC プリミティブのポート定義

ポート名	I/O	記 述
CLKI	I	クロック入力
RST	I	リセット入力。非同期で全ての出力を強制的に Low にする： RST = 0、クロック出力の出力がアクティブ RST = 1、クロック出力の出力がオフ
ALIGNWD	I	ワードアラインメント用に信号を使用： ALIGNWD = 0、不使用時 詳細については、"TN1203, Implementing High-Speed Interfaces with MachXO2 Devices (MachXO2 デバイスによる高速インターフェイスの実装)" を参照
CDIV1	O	出力ポートを 1 分周した出力。RST=1 の時はトグルせず、Low か High のまま
CDIVX	O	出力ポートを 2、3.5、または 4 で分周

表 13-7. CLKDIVC プリミティブの属性定義

名 称	記 述	値	デフォルト値
GSR	GSR イネーブル	ENABLED, DISABLED	DISABLED
DIV	CLK 分周器	2.0, 3.5 or 4.0	2.0

ALIGNWD 入力の使用目的は、DDR または 7:1 LVDS ビデオなどの高速データインターフェイスです。ALIGNWD の使用方法の詳細については、"TN1203, Implementing High-Speed Interfaces with MachXO2 Devices (MachXO2 デバイスによる高速インターフェイスの実装)" を参照してください。

VHDL ソースコードでの CLKDIVC の宣言

ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネントと属性の宣言

```
component CLKDIVC
generic (DIV : string;
        GSR : string);
port  (RST      : in  std_logic;
       CLKI     : in  std_logic;
       ALIGNWD  : in  std_logic;
       CDIV1    : out std_logic;
       CDIVX    : out std_logic);
end component;
```

CLKDIVC インスタンス化

```
I1: CLKDIVC
generic map ( DIV => "2.0",
              GSR => "DISABLED")
port map (   RST => RST,
             CLKI => CLKI,
             ALIGNWD => ALIGNWD,
             CDIV1 => CDIV1,
             CDIVX => CDIVX);
```

Verilog ソースコードでの CLKDIVC の使用

コンポーネントと属性の宣言

```
module CLKDIVC (RST, CLKI, ALIGNWD, CDIV1, CDIVX);

parameter DIV = "2.0";           // "2.0", "3.5", "4.0"
parameter GSR = "DISABLED";      // "ENABLED", "DISABLED"

input RST;
input CLKI;
input ALIGNWD;
output CDIV1;
output CDIVX;

endmodule
```

CLKDIVC インスタンス化

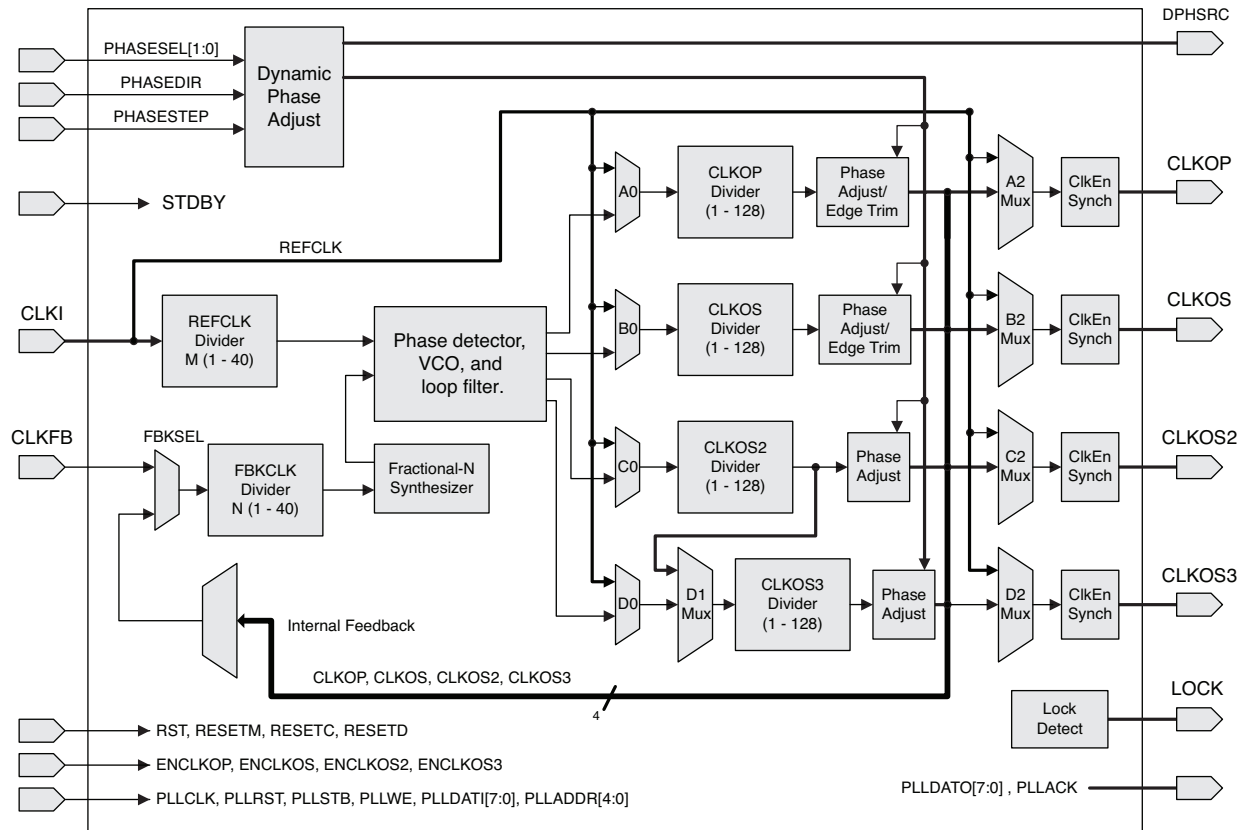
```
defparam I1.DIV = "2.0";
defparam I1.GSR = "DISABLED";

CLKDIVC I1 ( .RST (RST);
             .CLKI (CLKI);
             .ALIGNWD (ALIGNWD);
             .CDIV1 (CDIV1);
             .CDIVX (CDIVX));
```

sysCLOCK PLL

MachXO2 の PLL は、クロック注入遅延除去や周波数生成、及び位相調整などの機能を提供します。XO2-640U、1200/U および 2000 各デバイスには PLL は一個、XO2-2000U、4000 および 7000 には二個集積されています。二個集積されているデバイスで、オンチップ・フラッシュメモリのバックグラウンド・プログラミングを実装する場合、デバイス内右側にある RPLL と呼んでいる PLL の動作に留意が必要です。より詳細については TN1204 (MachXO2 Programming and Configuration Usage Guide) を参照ください。図 13-8 に MachXO2 の PLL のブロック図を示します。

図 13-8. MachXO2 PLL のブロック図



機能説明

PLL 分周器ブロック

入力クロック (CLKI) 分周器：CLKI 分周器は PLL ブロックへの入力クロック周波数の制御に使用されます。分周器設定は出力クロックの分周値に直接関連しています。入力と入力分周器の出力は、データシートで規定された位相検出器の周波数範囲内に入っている必要があります。

フィードバックループ (CLKFB) 分周器：CLKFB 分周器は、フィードバック信号の分周に使用されます。分周されたフィードバックをスピードアップし、PLL ブロックへの入力周波数に合わせる必要があるため、実質的には出力クロックが倍されます。分周されたフィードバック周波数が入力周波数と等しくなるまで、PLL ブロックは出力周波数を増加します。フィードバック分周器の出力は、MachXO2 ファミリー・データシートで規定された位相検出器の周波数範囲内に入っている必要があります。

出力クロック分周器 (CLKOP、CLKOS、CLKOS2、CLKOS3)：出力クロック分周器は、ジッタを最少化するために VCO 周波数を 400 ~ 800MHz に上げる目的で使用することもあります。各出力分周器は他の分周器から独立し、デフォルトではそれぞれ VCO をソースとして使用します。各出力分周器に設定できる値は

1 ~ 128 です。CLKOS や CLKOS2、または CLKOS3 分周器を従属接続することで、必要に応じてより低い周波数の出力を生成できます。

位相調整 (スタティックモード) : CLKOP、CLKOS、CLKOS2、及び CLKOS3 の出力は、入力クロックに対して相対的に位相調整できます。位相調整は 45° ステップで行えます。フィードバックとして選択されたクロック出力では、スタティック位相調整機能を使用できません。

位相調整 (ダイナミックモード) : PHASESEL、PHASEDIR、及び PHASESTEP の各ポートを使用して、ダイナミックモードで位相調整を制御することもできます。フィードバックとして選択されたクロック出力では、ダイナミック位相調整機能を使用できません。詳細については、本書の "ダイナミック位相調整" セクションを参照してください。

エッジトリム調整 (スタティックモード) : エッジトリム調整機能により、CLKOP 及び CLKOS ポートを細かく調整できます。

PLL の機能

スタンバイモード

MachXO2 PLL にはスタンバイモードがあり、デザイン内で PLL が動作不要な場合に PLL をスタンバイ状態にして電力を節約できます。デザインの必要性に応じて、PLL を完全にパワーダウンしたり、一部分のみをパワーダウンできます。

フラクショナル N 周波数生成

MachXO2 PLL にはフラクショナル N 周波数生成機能があり、ユーザが入力周波数の整数倍ではない出力クロックを生成できます。ユーザはフラクショナル N 分周器として、0 ~ 65535 の範囲の値を入力できます。この値は 65536 で分周されてから、その結果がフィードバック分周器に加えられます。平均有効フィードバック除数がこの値と等しくなるよう、MASH デルタシグマ変調方式が使用されています。目標周波数に対する偏差を小さくするためにフラクショナル N を使用することもできます。

WISHBONE ポート

MachXO2 PLL には WISHBONE ポート機能があり、PLL 設定をユーザロジックからダイナミックに変更できます。この機能を使用するときは、デザイン内で EFB ブロックもインスタンス化し、WISHBONE ポートにアクセス可能にしておく必要があります。正しいシミュレーションと動作のためには、PLL の WISHBONE ポートは EFB ブロックの同ポートに接続しなければなりません。WISHBONE ポートの使用法については、付録 D で詳しく説明します。

PLL の入力と出力

CLKI 入力

CLKI 信号は PLL の基準クロックです。PLL が正しく動作するためには、CLKI 入力がデータシートの仕様に適合している必要があります。CLKI 信号は専用のデュアルファンクション I/O ピン、任意の I/O ピン、または配線から引き出すことができます。専用のデュアルファンクション I/O ピンは低スキューの入力パスであり、PLL のソースとして推奨されます。基準クロックは入力 (M) 分周器によって分周され、PLL の位相検出器への 1 方の入力になります。上下辺バンクにある GPLL 用と PCLK ピンは PLL 入力に直接接続されています。左右辺バンクの PCLK 入力はプライマリクロック配線を介して PLL 入力に接続されます。

CLKFB 入力

CLKFB 信号は PLL へのフィードバック信号です。フィードバック信号は、正しい周波数、位相、またはその他の特性を維持するために出力クロックの調整が必要かどうかの判定用に用いられます。CLKFB 信号はプライマリクロック・ネットか専用の 2 目的 I/O ピン、または配線から引き出すか、もしくは出力クロック分周器から直接引き出すことができます。外部フィードバックを使用することで、設計者は基板レベルの

クロック位相を補正できます。フィードバック・クロック信号はフィードバック (N) 分周器によって分周され、PLL の位相検出器への入力を作成されます。バイパスされた PLL 出力をフィードバック信号として使用することはできません。

RST 入力

PLL のリセットは 2 つの条件で発生します。まず電源投入時に、コンフィグレーション・ブロックからの内部パワーアップ・リセット信号が PLL をリセットします。またユーザ制御の PLL リセット信号 RST は、PLL モジュールの一部として用意されます。RST 信号は内部的に生成されたリセット機能または、I/O ピンからドライブできます。この RST 信号は PLL コア (VCO、位相検出器、及びチャージポンプ) と出力分周器をリセットし、出力を Low にします。出力がバイパスモードの場合も同様です。

RST 信号がネゲートされると、PLL はロックイン・プロセスを開始し、PLL ロックの完了には t_{LOCK} 時間がかかります。図 13-9 に RST 入力のタイミング図を示します。RST 信号はアクティブ High で、オプションです。

RST 入力は入力分周器 (M 分周器) をリセットしません。M 分周器をリセットしないのは、外部で使用されているクロックが基準クロックと同期している場合があるためです。その場合、外部クロックと M 分周クロック (PLL の同期先) の間には、状態の関連性があります。PLL をリセットする場合、ユーザはこの関連性を維持する必要があります。その状況では、M 分周器をリセットすることなく PLL をリセットするために RST が使用されます。

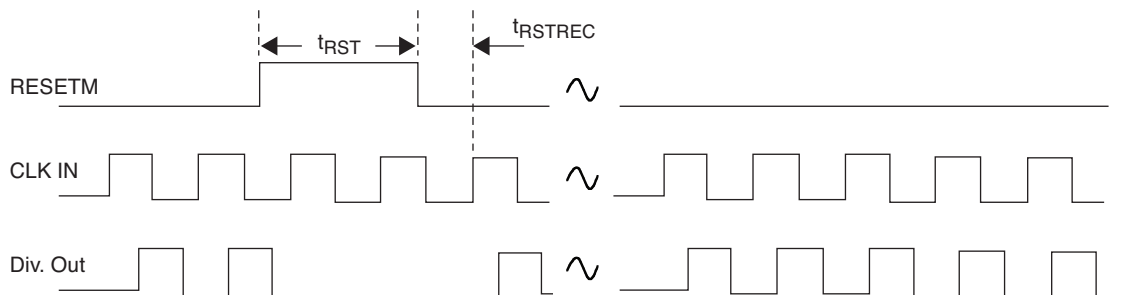
RESETM 入力

ユーザ制御の PLL リセット信号 RESETM が、PLL モジュールの一部として用意されています。RESETM 信号は、ユーザ回路で生成されたリセット機能または I/O ピンからドライブできます。RESETM 信号は PLL コアをリセットし (RST と同様)、M 分周器を含む全ての分周器をリセットします。その結果、PLL がバイパスモードになっている場合を含めて、出力はグランドレベルになります。

RESETM 信号がネゲートされると、PLL はロックイン・プロセスを開始し、PLL ロックの完了には t_{LOCK} 時間がかかります。図 13-9 に RESETM 入力のタイミング図を示します。RESETM 信号はアクティブ High で、オプションです。

ユーザが PLL 出力を外部クロックソースと同期する場合は、RESETM 信号を使用して PLL をリセットできます。

図 13-9. RST と RESETM のタイミング図



RESETC 入力

ユーザ制御の PLL リセット信号 RESETC は、PLL モジュールの一部として提供されます。RESETC 信号は、内部的に生成されたリセット機能または I/O ピンでドライブできます。この RESETC 信号は CLKOS2 出力分周器のみをリセットします。その結果、出力がバイパスモードの場合を除いて、CLKOS2 出力はグランドに接続されます。この出力がクロック分周器としてバイパスモードになっている場合は、RESETC 信号によってリセットされます。RESETC 信号を使用すると、CLKOS2 出力を外部クロック信号と同期できます。

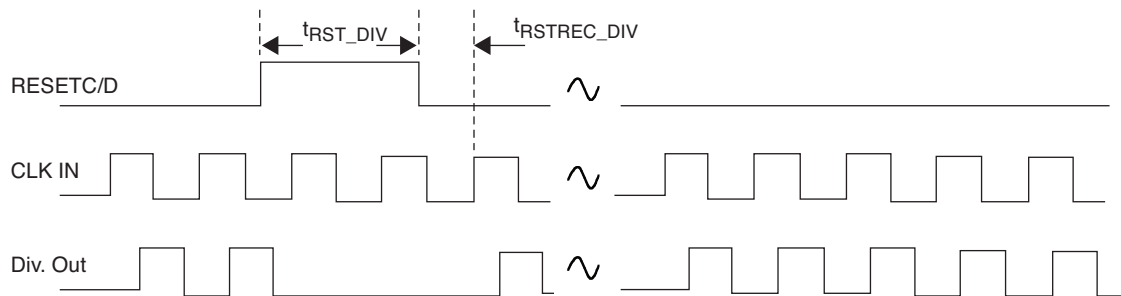
RESETC 信号がネゲートされてから、次のクロックエッジが CLKOS2 出力分周器をトグルするまでには、 t_{RSTREC_DIV} の時間遅延があります。図 13-10 に RESETC 入力のタイミング図を示します。CLKOS2 出力がフィードバックパス内で使用されていない場合、RESETC 信号は PLL ループに影響を与えません。CLKOS2 出力がフィードバックパス内で使用されている場合は、RESETC 信号ではなく RST または RESETM 信号を使用して PLL をリセットすることを推奨します。RESETC 信号はアクティブ High で、オプションです。

RESETD 入力

ユーザ制御の PLL リセット信号 RESETD は、PLL モジュールの一部として提供されます。RESETD 信号は、内部的に生成されたリセット機能または I/O ピンでドライブできます。この RESETD 信号は CLKOS3 出力分周器のみをリセットします。その結果、出力がバイパスモードの場合を除いて、CLKOS3 出力はグランドに接続されます。この出力がクロック分周器としてバイパスモードになっている場合は、RESETD 信号によってリセットされます。RESETD 信号を使用すると、CLKOS3 出力を外部クロック信号と同期できます。

RESETD 信号がネゲートされてから、次のクロックエッジが CLKOS3 出力分周器をトグルするまでには、 t_{RSTREC_DIV} の時間遅延があります。図 13-10 に RESETD 入力のタイミング図を示します。CLKOS3 出力がフィードバックパス内で使用されていない場合、RESETD 信号は PLL ループに影響を与えません。CLKOS3 出力がフィードバックパス内で使用されている場合は、RESETD 信号ではなく RST または RESETM 信号を使用して PLL をリセットすることを推奨します。RESETD 信号はアクティブ High、オプションです。

図 13-10. RESETC と RESETD のタイミング図



ENCLKOP 入力

ENCLKOP 信号は、ユーザ信号によって CLKOP 出力をイネーブル及びディセーブルするために使用されます。これにより、CLKOP 出力が使用されていない場合に CLKOP 出力を停止することで電力を節約できます。また、この信号を使用して CLKOP をデザイン内の別の信号と同期することもできます。ENCLKOP 信号はオプションであり、ユーザが IPexpress™ で Clock Enable ポートオプションを選択した場合のみ使用可能になります。ENCLKOP 信号が要求されていない場合は、PLL がスタンバイモードのときを除き、CLKOP 出力は（PLL がインスタンス化されるときに）常にアクティブになります。ENCLKOP 信号はアクティブ High です。

ENCLKOS 入力

ENCLKOS 信号は、ユーザ信号によって CLKOS 出力をイネーブル及びディセーブルするために使用されます。これにより、CLKOS 出力が使用されていない場合に CLKOS 出力を停止することで電力を節約できます。また、この信号を使用して CLKOS をデザイン内の別の信号と同期することもできます。ENCLKOS 信号はオプションであり、この信号を使用できるのは、IPexpress で CLKOS 出力及び Clock Enable ポートオプションを選択して PLL を構成している場合のみです。PLL が CLKOS 出力イネーブルとして構成されていて、かつ ENCLKOS 信号が要求されていない場合は、PLL がスタンバイモードのときを除き、CLKOS 出力は常にアクティブになります。ENCLKOS 信号はアクティブ High です。

ENCLKOS2 入力

ENCLKOS2 信号は、ユーザ信号によって CLKOS2 出力をイネーブル及びディセーブルするために使用されます。これにより、CLKOS2 出力が使用されていない場合に CLKOS2 出力を停止することで電力を節約できます。また、この信号を使用して CLKOS2 をデザイン内の別の信号と同期することもできます。ENCLKOS2 信号はオプションであり、この信号を使用できるのは、IPexpress で CLKOS2 出力及び Clock Enable ポートオプションを選択して PLL を構成している場合のみです。PLL が CLKOS2 出力イネーブルとして構成されていて、かつ ENCLKOS2 信号が要求されていない場合は、PLL がスタンバイモードのときを除き、CLKOS2 出力は常にアクティブになります。ENCLKOS2 信号はアクティブ High です。

ENCLKOS3 入力

ENCLKOS3 信号は、ユーザ信号によって CLKOS3 出力をイネーブル及びディセーブルするために使用されます。これにより、CLKOS3 出力が使用されていない場合に CLKOS3 出力を停止することで電力を節約できます。また、この信号を使用して CLKOS3 をデザイン内の別の信号と同期することもできます。ENCLKOS3 信号はオプションであり、この信号を使用できるのは、IPexpress で CLKOS3 出力及び Clock Enable ポートオプションを選択して PLL を構成している場合のみです。ENCLKOS3 信号が要求されていない場合は、PLL がスタンバイモードのときを除き、CLKOS3 出力は常にアクティブになります。ENCLKOS3 信号はアクティブ High です。

STDBY 入力

STDBY 信号は、PLL が不要な場合に PLL を低電力スタンバイモードにするために使用されます。STDBY ポートを電源コントローラに接続すると、デバイスがスタンバイモードになった場合に PLL を低消費電力状態にできます。あるいは、スタンバイモードとは無関係に、ユーザロジックによって STDBY ポートをドライブできます。STDBY 信号はオプションであり、ユーザが IPexpress で Standby ポートオプションを選択した場合にのみ使用可能になります。STDBY 信号はアクティブ High です。

PHASESEL 入力

PHASESEL[1:0] 入力は、ダイナミック位相調整ポートの影響を受ける PLL 出力ポートの指定に使用されます。使用可能な設定は、本書の“ダイナミック位相調整”セクションに示します。PHASESEL 信号は PHASESTEP 信号がトグルされる前に安定している必要があります。PHASESEL 信号はオプションであり、ユーザが IPexpress で Dynamic Phase ポートオプションを選択した場合のみ使用可能になります。

PHASEDIR 入力

PHASEDIR 入力は、ダイナミック位相シフトを進めるかと遅らせるかのどの方向かの指定に使用されます。PHASEDIR = 0 の場合、位相シフトは現在のクロックから 1 ステップ遅れます。PHASEDIR = 1 の場合、位相シフトは現在のクロックから 1 ステップ進みます。PHASEDIR 信号は PHASESTEP 信号がトグルされる前に安定している必要があります。

PHASEDIR 信号はオプションであり、ユーザが IPexpress で Dynamic Phase ポートオプションを選択した場合のみ使用可能になります。

PHASESTEP 入力

PHASESTEP 信号は、PHASESEL 入力に指定されたクロック出力ポートのダイナミック位相調整を、PHASEDIR 入力に指定された方向に開始するために使用されます。PHASESTEP 信号はオプションであり、ユーザが IPexpress で Dynamic Phase ポートオプションを選択した場合のみ使用可能になります。

CLKOP 出力

CLKOP は sysCLOCK PLL の主クロック出力です。この信号はデフォルトで常に使用可能であり、チップのプライマリクロック・ネットワークに配線できます。CLKOP 出力は上下辺のエッジクロックにも配線でき

ます。CLKOP 出力はスタティックまたはダイナミックに位相シフト可能であり、デューティトリム調整機能と併せて使用することもできます。CLKOP 信号出力は、CLKOP 出力分周器から引き出すことも、PLL をバイパスすることもできます。CLKOP がバイパスモードの場合は、出力分周器をバイパスすることも、回路で使用することもできます。

CLKOS 出力

CLKOS 信号は sysCLOCK PLL のセカンダリクロック出力です。この信号はユーザが選択した場合に使用でき、デバイスのプライマリクロック・ネットワークに配線できます。CLKOS 出力は上辺及び下辺のエッジクロックにも配線できます。CLKOS 出力はスタティックまたはダイナミックに位相シフト可能であり、デューティトリム調整機能と併せて使用することもできます。CLKOS 信号出力は、CLKOS 出力分周器から引き出すことも、PLL をバイパスすることもできます。CLKOS がバイパスモードの場合は、出力分周器をバイパスすることも、回路で使用することもできます。CLKOS 信号はオプションです。

CLKOS 出力は、128 より大きい分周値が必要な低い周波数の出力にも対応できます。これは CLKOP と CLKOS 出力の分周器を従属接続することにより実現します。このような場合、CLKOP は単独の出力として用いることはできません。また、従属接続した出力を PLL のフィードバック入力として使用することもできません。

CLKOS2 出力

CLKOS2 信号は、sysCLOCK PLL で使用できるもう 1 本のセカンダリクロック出力です。この信号はユーザが選択した場合に使用でき、チップのプライマリクロック・ネットワークに配線できます。CLKOS2 出力は上下辺のエッジクロックに配線できません。CLKOS2 出力はスタティックまたはダイナミックに位相シフト可能ですが、デューティトリム調整機能はありません。CLKOS2 信号出力は、CLKOS2 出力分周器から引き出すことも、PLL をバイパスすることもできます。CLKOS2 がバイパスモードの場合は、出力分周器をバイパスすることも、回路で使用することもできます。CLKOS2 信号はオプションです。

CLKOS2 出力は、128 より大きい分周値が必要な低い周波数の出力にも対応できます。これは CLKOS と CLKOS2 出力の分周器を従属接続することにより実現します。このような場合、CLKOS は単独の出力として用いることはできません。また、従属接続した出力を PLL のフィードバック入力として使用することもできません。

CLKOS3 出力

CLKOS3 信号は、sysCLOCK PLL で使用できるもう 1 本のセカンダリクロック出力です。この信号はユーザが選択した場合に使用でき、チップのプライマリクロック・ネットワークに配線できます。CLKOS3 出力は上辺 / 下辺のエッジクロックに配線できません。CLKOS3 出力はスタティックまたはダイナミックに位相シフト可能ですが、デューティトリム調整機能はありません。CLKOS3 信号出力は、CLKOS3 出力分周器から引き出すことも、PLL をバイパスすることもできます。CLKOS3 がバイパスモードの場合は、出力分周器をバイパスすることも、回路で使用することもできます。CLKOS3 信号はオプションです。

CLKOS3 出力は、128 を超える出力分周器の値を必要とする、低い周波数出力にも対応しています。これは、CLKOS2 及び CLKOS3 出力分周器を従属接続することで実現されます。この用途で用いる場合、独立したクロック出力として CLKOS2 出力を使用することはできません。従属接続したクロック出力を PLL のフィードバック信号に使用することはできません。

DPHSRC 出力

DPHSRC 出力は、ダイナミック位相ポートと WISHBONE レジスタの、どちらがダイナミック位相調整機能の制御に使用されているかを示すために使用されます。ダイナミック位相ポートは、PHASESEL、PHASEDIR、及び PHASESTEP ポートです。DPHSRC 信号はオプションであり、ユーザが IPexpress で Dynamic Phase ポートオプションを選択すると使用可能になります。ユーザが Dynamic Phase ポートオプションを選択していない場合は、デフォルトで WISHBONE レジスタがダイナミック位相調整機能の設定に使用されます。

LOCK 出力

LOCK 出力は PLL のステータスに関する情報を提供します。デバイスの電源投入後、入力クロックが有効になると、PLL は規定されたロック時間内にロックを達成します。ロックすると PLL LOCK 信号がアサートされます。LOCK は、標準ロックモードとスティッキー・ロックモードのどちらにもできます。標準ロックモードでは、LOCK 信号は PLL のロックが確認されたときにアサートされ、ロックが外れたときにネゲートされます。スティッキー (sticky) ロックモードでは、LOCK 信号がいったんアサートされると、PLL リセットが与えられるまで LOCK 信号はアサートされたままになります。PLL のロックが外れた場合に PLL を基準クロックと再同期するために、PLL RST をアサートすることを推奨します。LOCK 信号は FPGA ロジックで使用し、設計者の要求に応じて RST 信号の生成回路を実装することができます。LOCK 信号はオプションであり、ユーザが IPexpress 内で PLL Lock 信号オプションを選択すると使用可能になります。^{注1}

WISHBONE ポート

WISHBONE ポートのリストは、使用法の説明と共に付録 D に示します。WISHBONE ポートはオプションです。

PLL 属性

PLL はソース制約とプリファレンス・ファイルによってパラメータ設定を可能にするいくつかの属性を利用します。以下のセクションでは、これらの属性とその使用法について説明します。

FIN

入力周波数は分周器設定に基づいて、規定された周波数範囲内の任意の値にできます。

CLKI_DIV, CLKFB_DIV, CLKOP_DIV, CLKOS_DIV, CLKOS2_DIV, CLKOS3_DIV

これらの分周器は各出力クロックの周波数を決定します。IPexpress を使用する場合、ユーザは無効な組み合わせを入力することはできません。有効な組み合わせは、入力周波数、分周器、及び PLL 仕様で決定されます。

CLKOP_DIV 値は、規定された範囲内で FVCO が最大になるように、FIN と CLKOP_FREQ に基づいて CLKI_DIV と CLKFB_DIV の値で計算されます。これは、CLKOP 出力をフィードバック信号に使用する場合です。別の出力をフィードバック信号に使用する場合は、対応する出力分周器をこの方法で計算します。

フィードバック信号に使用されない出力信号は、VCO 周波数及び所望の出力周波数に基づく出力分周器の値を使用します。この全ての分周器で可能な値は 1 から 128 ですが、場合によっては、PLL 仕様に違反するためにこの全範囲が許容されないことがあります。

FREQUENCY_PIN_CLKI, FREQUENCY_PIN_CLKOP, FREQUENCY_PIN_CLKOS, FREQUENCY_PIN_CLOS2, FREQUENCY_PIN_CLKOS3

これらの入力及び出力クロック周波数は分周器の値を決定します。

Frequency Tolerance – CLKOP, CLKOS, CLKOS2, CLKOS3

周波数偏差。必要な出力周波数が達成できない場合は、クロック出力の周波数偏差を入力してもかまいません。

1. 英語版にある R1 に関する記述は省略

MachXO2 PLL プリミティブ定義

PLL プリミティブは、このセクションの定義に従ってデザインのソースコードでインスタンス化できます。図 13-11 と表 13-8 に EHXPLLJ の定義を示します。

図 13-11. PLL プリミティブ・シンボル

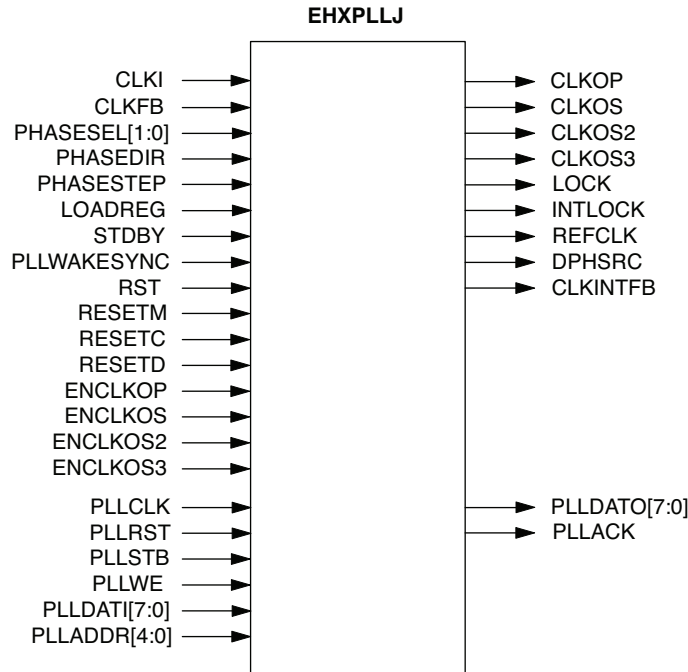


表 13-8. PLL プリミティブのポート定義

ポート名	I/O	記述
CLKI	I	PLL へのクロック入力
CLKFB	I	フィードバック・クロック
PHASESEL[1:0]	I	ダイナミック位相調整ポートの影響を受ける出力の選択
PHASEDIR	I	ダイナミック位相調整の方向
PHASESTEP	I	ダイナミック位相ステップ ~ トグルにより VCO 位相調整が 1 ステップシフト
LOADREG	I	ダイナミック位相ロード ~ トグルにより分周器の位相調整値を PLL にロード
CLKOP	O	プライマリ PLL 出力クロック (位相シフト調整あり)
CLKOS	O	セカンダリ PLL 出力クロック (位相シフト調整あり)
CLKOS2	O	セカンダリ PLL 出力クロック 2 (位相シフト調整あり)
CLKOS3	O	セカンダリ PLL 出力クロック 3 (位相シフト調整あり)
LOCK	O	PLL ロック、非同期信号。アクティブ High で PLL が入力とフィードバック信号にロックされていることを示す
INTLOCK	O	PLL 内部ロック、非同期信号。アクティブ High で PLL が内部フィードバックでロックしていることを示す ¹
REFCLK	O	基準クロック・マルチプレクサの出力
DPHSRC	O	ダイナミック位相ソース ~ ポートまたは WISHBONE がアクティブ
STDBY	I	PLL をパワーダウンスするスタンバイ信号
PLLWAKESYNC	I	PLL ウェイクアップ、同期。PLL ウェイクアップ時に PLL が内部パスからユーザ指定フィードバックパスへの切り替えをイネーブルする ¹
RST	I	M 分周器をリセットしない PLL リセット。アクティブ High
RESETM	I	M 分周器もリセットする PLL リセット。アクティブ High

表 13-8. PLL プリミティブのポート定義 (Continued)

ポート名	I/O	記述
RESETC	I	CLKOS2 出力分周器のみをリセットする。アクティブ High
RESETD	I	CLKOS3 出力分周器のみをリセットする。アクティブ High
ENCLKOP	I	CLKOP 出力のクロックイネーブル
ENCLKOS	I	CLKOS 出力のクロックイネーブル ~ CLKOS ポートがアクティブの場合のみ使用可
ENCLKOS2	I	CLKOS2 出力のクロックイネーブル ~ CLKOS2 ポートがアクティブの場合のみ使用可
ENCLKOS3	I	CLKOS3 出力のクロックイネーブル ~ CLKOS3 ポートがアクティブの場合のみ使用可
PLLCLK	I	PLL データバス、クロック入力
PLLSTB	I	PLL データバス、ストロブ信号
PLLWE	I	PLL データバス、ライトイネーブル信号
PLLADDR [4:0]	I	PLL データバス、アドレス
PLLDATI [7:0]	I	PLL データバス、データ入力
PLLDATO [7:0]	O	PLL データバス、データ出力
PLLACK	O	PLL データバス、アクノレッジ信号

1. PLLWAKWSYNC と INTLOCK プリミティブポートは、モジュール生成に IPexpress が用いられた場合に付加される。ポートはモジュール内で接続されていない。テスト結果ではこれらポートの使用は実効的な意義がないことを示す

ダイナミック位相調整

MachXO2 PLL は、ダイナミック位相調整ポートまたは WISHBONE インターフェイスによって、以下の方法でダイナミック位相調整を実現します。WISHBONE インターフェイスについては、付録 D で詳しく説明します。

ダイナミック位相調整機能を使用するには、PHASESEL[1:0]、PHASEDIR、PHASESTEP の各ポート / 信号を使用します。DPHSRC ポートも用意されており、位相調整の実装前に正しい信号ソース、プリミティブポート、または WISHBONE 信号が選択されていることを確認できます。ダイナミック位相調整ポートが選択されている場合のデフォルト設定では、プリミティブポートがダイナミック位相調整に使用されます。WISHBONE レジスタの MC1_DYN_SOURCE を使用して、ダイナミック位相調整のソースを必要に応じて WISHBONE インターフェイスから変更することもできます。ユーザがダイナミック位相ポートを GUI インターフェイスから選択していない場合は、WISHBONE 信号がダイナミック位相調整に使用されます。

CLKOP と CLKOS、CLKOS2、及び CLKOS3 の 4 本の出力クロックには全てダイナミック位相調整機能がありますが、調整できるのは一度に 1 系統の出力クロックだけです。表 13-9 に PHASESEL[1:0] 信号で設定する出力クロックの選択を示します。PHASESEL 信号は PHASESTEP 信号がトグルされる前に安定している必要があります。

表 13-9. PHASESEL 信号設定の定義

PHASESEL[1:0]	シフトされる PLL 出力
00	CLKOS
01	CLKOS2
10	CLKOS3
11	CLKOP

選択した出力クロック位相は、PHASEDIR ポートまたは信号の値に応じて進められるかまたは遅らされます。表 13-10 に使用可能な PHASEDIR 設定を示します。PHASEDIR 信号は PHASESTEP 信号がトグルされる前に安定している必要があります。

表 13-10. PHASEDIR 信号設定の定義

PHASEDIR	方 向
00	遅れ (lagging)
01	進み (leading)

PHASESEL と PHASEDIR の設定後、PHASESTEP 信号をトグルすることによって位相調整が行われます。PHASESTEP 信号のパルスごとに 1 ステップの位相シフトが発生します。PHASESTEP 信号パルスはロジックゼロ値から開始される必要があります、位相シフトは PHASESTEP 信号の立ち下がりエッジで開始されます。ステップサイズは次の式で表されます。

$$\text{ステップサイズ} = 45^\circ / \text{出力分周器 (値)}$$

1 ステップよりも大きい位相シフトが必要な場合は、PHASESTEP 信号の複数のパルスによって所望の位相シフトを発生できます。PLL で発生できる最小の位相シフトは 1 ステップサイズです。ダイナミック位相調整では、出力クロックを遅らせる場合はグリッチのない調整になりますが、出力クロックを進める場合はグリッチが発生することがあります。

周波数の計算

PLL はユーザボードに必要とされる周波数ソースがない場合に、所望周波数のクロックを生成するために用いることができます。生成される周波数は以下に示す等式で計算できます。

$$f_{\text{OUT}} = f_{\text{IN}} * N/M \quad (1)$$

$$f_{\text{VCO}} = f_{\text{OUT}} * V \quad (2)$$

$$f_{\text{PFD}} = f_{\text{IN}} / M = f_{\text{FB}} / N \quad (3)$$

ここで、

f_{OUT} は出力周波数

f_{IN} は入力周波数

f_{VCO} は VCO 周波数

f_{PFD} は PFD (位相検出器) 周波数

f_{FB} はフィードバック周波数

N はフィードバック分周器の値 (IPexpress GUI で示される整数値)

M は入力分周器の値 (IPexpress GUI で示される整数値)

V は出力分周器の値 (IPexpress GUI で示される整数値)

PLL へのフィードバック・ソースシンクロナスとして用いられるクロック出力信号に対しては、これら数式は真です。VCO 周波数がこれら等式から計算された場合、式 (2) によって残りの出力クロック信号を計算するために用いることができます。

上記等式は、出力とフィードバック両パスが等価な場合に成立します。そうでない場合は、二つの分周器を考慮にいれる必要があるため、等式 (1) はより複雑になります。

フラクショナル N シンセシス動作

MachXO2 sysCLOCK PLL は高精度（16 ビット）のフラクショナル N シンセシスに対応します。フラクショナル N 周波数生成では入力周波数の整数倍でない出力クロックの生成が可能です。フラクショナル N シンセシス・オプションは IPexpress GUI で Fractional-N Divider 部のイネーブル（Enable）ボックスをチェックして、その隣のボックスに 0 から 65535 の値を入力することで使用可能になります。このボックスに入力された値は、フィードバック分周器の分数部を構成するために 65536 で分周されます（やはり N 分周器と呼ぶ）。実効分周値は以下で与えられます。

$$N_{\text{eff}} = N + (F/65536) \quad (4)$$

ここで、

N は整数フィードバック分周器の値（IPexpress GUI に示す）

F は上に示すフラクショナル N シンセシスボックスに入る値

出力周波数は次式で与えられます。

$$f_{\text{OUT}} = (f_{\text{IN}}/M) * N_{\text{eff}} \quad (5)$$

ここで、

f_{OUT} は出力周波数

f_{IN} は入力周波数

M は入力分周器の値（IPexpress GUI に示す）

フラクショナル N シンセシスは、ユーザが入力した分数値を近似するために、シグマデルタ技法を用いることで動作します。従って、フラクショナル N シンセシス・オプションは、整数フィードバック分周器に比較して、PLL VCO と出力クロックのジッタが大きくなる結果となります。PLL ジッタ性能を必要以上に悪化させないように、フラクショナル N シンセシスは N/M 分周器比が 4 かそれ以上の場合にのみ使用することを推奨します。フラクショナル N 動作ジッタ値の規格はデータシートを参照して下さい。

低消費電力機能

MachXO2 PLL には、設計者がデザインの電力消費を最小限にできるいくつかの機能が内蔵されています。これにはダイナミック・クロックイネーブルと、スタンバイモードがあります。

ダイナミック・クロックイネーブル

ダイナミック・クロックイネーブル機能を使用すると、選択した出力クロックをデザイン内で使用していない期間にオフにできます。この機能を実現するため、各出力クロックには選択可能な独立した出力イネーブル信号があります。出力イネーブル信号は ENCLKOP と ENCLKOS、ENCLKOS2、及び ENCLKOS3 です。IPexpress GUI で Clock Enable ポートオプションを選択している場合、IPexpress GUI でイネーブルになっている CLKOP ポートとその他の全てのポートに対して、PLL モジュールのトップレベル・ポートに出力イネーブル信号を引き出すことができます。

出力が IPexpress GUI でイネーブルになっていない場合、選択したその出力信号のポートはモジュールに表示されず、その出力は非アクティブになります。

スタンバイモード

電力消費を最小限にするため、アプリケーションで PLL が必要とされていない場合に PLL をシャットダウンすることができます。その後 PLL が再び必要になった場合に PLL を再始動でき、PLL がフィードバック信号にロックするための短い遅延の後で、出力クロックが再びアクティブになります。このモードを実現するには、IPexpress GUI で Standby Ports オプションを選択します。これにより、PLL モジュールのトップレベルに STDBY と PLLWAKESYNC、及び INTLOCK 信号が引き出されます。PLL をスタンバイモードにすると、PLL がパワーダウンされ、全ての出力がディセーブルされます。

STDBY 信号が High にされ、出力が Low にドライブされると、PLL はスタンバイモードになります。STDBY ポートを電源コントローラに接続すると、デバイスがスタンバイモードになった場合に PLL を低消費電力状態にできます。あるいは、スタンバイモードとは無関係に、ユーザロジックによって STDBY ポートをドライブすることもできます。

STDBY 信号が Low にされると、PLL はスタンバイモードから起動します。スタンバイモードからウェイクアップする際に、スタンバイモードに入る前に選択されていた外部フィードバック信号に自動的にロックします。最大規定ロック時間は t_{LOCK} で、ロック状態を示すために LOCK 信号が High にアサートされます。

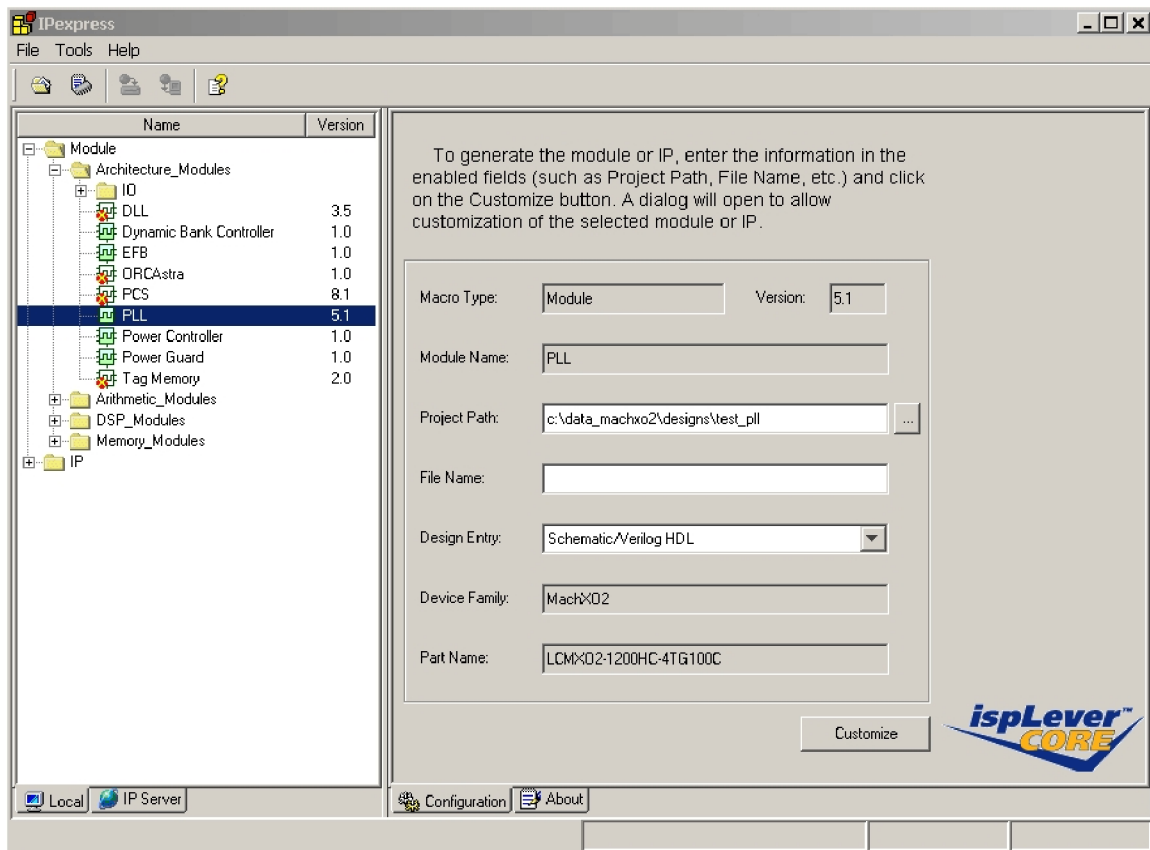
IPexpress を使用した PLL の設定

PLL の生成及びパラメータ設定には IPexpress を使用します。設計者は GUI を使用して PLL のパラメータを選択し、シミュレーションと論理合成フローで使用される HDL モデルを生成します。

図 13-12 に ispLEVER の IPexpress 内で PLL を選択した場合のメインウィンドウを示します。Lattice Diamond™ でこれに相当する画面の例については、付録 E の図 13-24 を参照してください。ispLEVER Project Navigator または Diamond から IPexpress を開くと、プロジェクト設定が自動的に入力されます。ispLEVER を使用する場合には入力が必要なエントリはファイル名だけです。Diamond を使用する場合は、ファイル名とモジュール出力タイプ（VHDL または Verilog）を入力する必要があります。

IPexpress をスタンドアロン・ツールとして開いた場合は、この画面に示す追加のパラメータを指定する必要があります。適切なモジュール名を入力した後、**Customize**（カスタマイズ）ボタンをクリックすると、図 13-13 に示す Configuration（構成）タブウィンドウが開きます。

図 13-12. PLL モジュールの IPexpress メインウィンドウ



Configuration タブ

Configuration（構成）タブには、ユーザがアクセス可能な属性の全リストが、デフォルト値が設定された状態で表示されます。完了後、**Generate**（生成）ボタンをクリックするとソースが生成されます。

Configuration Modes（コンフィグレーションモード）

Configuration タブでは、周波数モードまたは分周器モードで PLL をパラメータ設定できます。

- ・ **Frequency Mode**（周波数モード）：このモードでは、ユーザが入出力のクロック周波数を入力すると、IPexpress が分周器の設定値を計算します。入出力周波数を入力した後、Calculate（計算）ボタンをクリックすると、分周器の値と実際の周波数が表示されます。

ユーザが入力した出力周波数が実現できない場合は、最も近い周波数が “Actual（実際の値）” テキスト・ボックスに表示され、エラーメッセージが表示されます。偏差値をパーセント単位で入力することもできます。Calculate（計算）ボタンを押すと、計算結果が入力した偏差範囲内にある場合、計算は正確と見なされます。

入力した値が範囲外の場合は、Calculate（計算）ボタンを使用した後に値が赤で表示され、エラーメッセージが表示されます。

- ・ **Divider Mode**（分周器モード）：このモードでは、ユーザは入力周波数と分周値を設定します。ユーザは、MachXO2 ファミリー・データシートに規定された許容範囲内で、VCO の周波数が最大になる CLKOP 分周器値を選択します。

入力した値の組み合わせが無効な PLL 構成になる場合、値の変更を求めるプロンプトと、範囲外の値に対する推奨値がテキストボックスに表示されます。

図 13-13. MachXO2 PLL の Configuration (構成) タブ

Lattice FPGA Module -- PLL

Configuration | Generate Log

PLL

Estimated Resource Usage:

Configuration

☒ Frequency Mode
☐ Divider Mode

CLKI
 Frequency: 100
 Divider: 1

CLKFB
 FBK Mode: CLKOP
 FBK Divider: 1
 Fractional-N Divider (0 - 65535): 0
☐ Enable

Optional Port Selections
☐ Dynamic Phase Ports
☐ Clock Enable Ports
☐ Standby Ports

PLL Reset Options
☐ Provide PLL Reset
☐ Provide PLLM Reset
☐ PLL CLKOS2 Reset
☐ PLL CLKOS3 Reset

Lock Settings
☐ Provide PLL Lock signal
☐ PLL Lock is Sticky

Wishbone BUS
☐ Provide WB Ports
 Requires instantiation of EFB block

CLKOP
☐ Bypass
☐ Clock Divider
 Desired Frequency: 100
 Tolerance %: 0.0
 Divider: 8
 Actual Frequency:
 Static Phase Shift - degrees: 0
 Duty Trim Options:
☒ Rising Delay Multiplier: 0
☐ Falling

CLKOS
☐ Enable
☐ Bypass
☐ Clock Divider
 Desired Frequency: 100
 Tolerance %: 0.0
 Divider: 8
 Actual Frequency:
 Static Phase Shift - degrees: 0
 Duty Trim Options:
☒ Rising Delay Multiplier: 0
☐ Falling

CLKOS2
☐ Enable
☐ Bypass
☐ Clock Divider
 Desired Frequency: 100
 Tolerance %: 0.0
 Divider: 8
 Actual Frequency:
 Static Phase Shift - degrees: 0

CLKOS3
☐ Enable
☐ Bypass
☐ Clock Divider
 Desired Frequency: 100
 Tolerance %: 0.0
 Divider: 8
 Actual Frequency:
 Static Phase Shift - degrees: 0

Calculate

☐ Import IPX to Diamond project

Generate Close Help

表 13-11. IPexpress GUI のユーザパラメータ

ユーザパラメータ名	記 述	範 囲	デフォルト
Frequency Mode	ユーザが所望の CLKI 及び CLKOP 周波数を入力	ON/OFF	ON
Divider Mode	ユーザが所望の CLKI 周波数及び分周器設定値を入力	ON/OFF	OFF
CLKI	周波数	10 ~ 400 MHz	100 MHz
	分周	1 ~ 40	1
CLKFB	フィードバックモード	CLKOP, CLKOS, CLKOS2, CLKOS3, INT_OP, INT_OS, INT_OS2, INT_OS3, UserClock	CLKOP
	フラクショナル N 分周器イネーブル	ON / OFF	OFF
	フラクショナル N 分周器	0 ~ 65535	0
Output Port Selections	ダイナミック位相ポート	ON / OFF	OFF
	クロックイネーブル・ポート	ON / OFF	OFF
	スタンバイポート	ON / OFF	OFF
PLL Reset Options	PLL リセットを与える	ON / OFF	OFF
	PLLM リセットを与える	ON / OFF	OFF
	CLKOS2 リセットを与える	ON / OFF	OFF
	CLKOS3 リセットを与える	ON / OFF	OFF
Lock Settings	PLL LOCK 信号を与える	ON / OFF	OFF
	PLL LOCK は “スティッキー”	ON / OFF	OFF
WISHBONE Bus	WISHBONE ポートを与える	ON / OFF	OFF
CLKOP	バイパス	ON / OFF	OFF
	クロック分周器 (バイパスモードのみ)	ON / OFF	OFF
	所望の周波数	3.125 ~ 400 MHz	100 MHz
	偏差 (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	分周器	1 ~ 128	8
	実際の周波数 (リードオンリ)	-	-
	静的位相シフト (度)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	00
	立ち上がりエッジトリム	ON / OFF	OFF
	立ち下がりエッジトリム	ON / OFF	OFF
	遅延乗数	0, 1, 2, 4	0
CLKOS	イネーブル	ON / OFF	OFF
	バイパス	ON / OFF	OFF
	クロック分周器 (バイパスモードのみ)	ON / OFF	OFF
	所望の周波数	0.024 - 400 MHz	100 MHz
	偏差 (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	分周器	1 ~ 128	8
	実際の周波数 (リードオンリ)	-	-
	静的位相シフト (度)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	00
	立ち上がりエッジトリム	ON / OFF	OFF
	立ち下がりエッジトリム	ON / OFF	OFF
	遅延乗数	0, 1, 2, 4	0

表 13-11. IPexpress GUI のユーザパラメータ (Continued)

ユーザパラメータ名	記 述	範 囲	デフォルト
CLKOS2	イネーブル	ON / OFF	OFF
	バイパス	ON / OFF	OFF
	クロック分周器 (バイパスモードのみ)	ON / OFF	OFF
	所望の周波数	0.024 to 400 MHz	100 MHz
	偏差 (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	分周器	1 ~ 128	8
	実際の周波数 (リードオンリ)	-	-
	静的位相シフト (度)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	00
CLKOS3	イネーブル	ON / OFF	OFF
	バイパス	ON / OFF	OFF
	クロック分周器 (バイパスモードのみ)	ON / OFF	OFF
	所望の周波数	0.024 - 400 MHz	100 MHz
	偏差 (%)	0.0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0	0.0
	分周器	1 ~ 128	8
	実際の周波数 (リードオンリ)	-	-
	静的位相シフト (度)	0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°	0°

IPexpress 出力

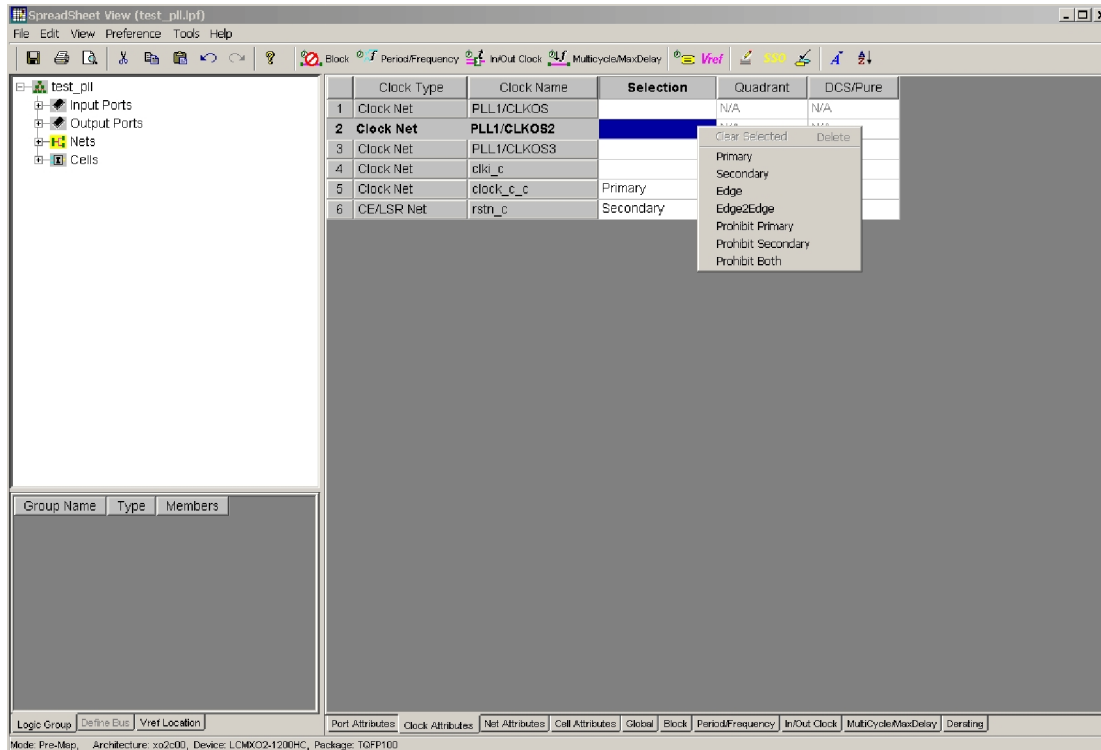
デザインで使用する重要な IPexpress の出力ファイルが 2 つあります。1 つは <module_name>.[v|vhd] ファイルです。これは IPexpress によって生成されユーザが命名したモジュールです。このファイルは、論理合成とシミュレーションの両フローで使用するのためのものです。他方はテンプレートファイル <module_name>.tpl.[v|vhd] で、これにはモジュールのインスタンス例が格納されています。このファイルはユーザがインスタンス例をコピー / ペーストして使用するために提供されるものであり、論理合成やシミュレーションのフローで直接使用するものではありません。

IPexpress は選択されたデータレート固有の属性を、PLL 用の HDL モジュールに設定します。これらの属性は簡単に変更できますが、PLL の性能を維持するには、必ず GUI を再実行して変更する必要があります。ツールフローのマップステージの後、PLL から生成されるクロックに自動的に制約を与えるため、FREQUENCY 設計制約がプリファレンス・ファイルに格納されます。

Pre-MAP Preference Editor の使用

クロックの設計制約をプリマップ・プリファレンスエディタ (Pre-MAP Preference Editor) で設定できます。図 13-14 にスクリーンショットの例を示します。クオドラント (Quadrant) 及び DCS/Pure 列は MachXO2 デバイスにはありません。プリマップ・プリファレンスエディタは ispLEVER® デザインプランナ・ツールの一部です。Diamond でこれに相当する機能は、単に "スプレッドシート・ビュー" と呼ばれます。

図 13-14. Pre-MAP Preference Editor の例



PLL 基準クロックスイッチ (PLLREFCS)

必要に応じて、MachXO2 PLL の基準クロックはオプションで、2 系統の異なるクロックソースを切り替えることができます。この機能を使用するには、デザイン内で PLLREFCS プリミティブがインスタンス化されている必要があります。PLLREFCS は PLL でのみ使用できます。

基準クロックを切り替えると、PLL のロックが一定期間外れることがあります。その場合、ロックの再獲得までに、MachXO2 ファミリー・データシートで規定された t_{LOCK} 時間かかることがあります。周波数の異なる基準クロック信号に切り替える場合は、PLL をリセットすることを推奨します。

PLLREFCS プリミティブは、このセクションの定義に従ってデザインのソースコードでインスタンス化できます。図 13-15 と表 13-12 に PLLREFCS の定義を示します。

図 13-15. PLLREFCS プリミティブ・シンボル

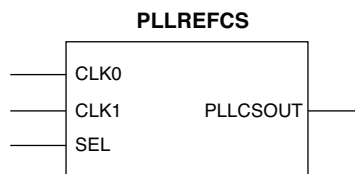


表 13-12. PLLREFCS プリミティブのポート定義

ポート名	I/O	記 述
CLK0	NO	CLK0
CLK1	NO	CLK1
SEL	NO	SEL - SEL = 0 CLK0 入力を選択されている - SEL = 1 CLK1 入力を選択されている
PLLCSOUT	NO	PLLCSOUT

内蔵オシレータ (OSCH)

MachXO2 デバイスには、デザイン内でクロックソースとして使用できる内蔵オシレータがあります。内蔵オシレータの精度は $\pm 5\%$ (公称値) です。このオシレータは、高精度のクロックを必要としないアプリケーションのクロックソースとして使用するためのものです。

透過 (transparent) モードのコンフィグレーション中、内蔵オシレータはユーザロジックに対してアクティブなまま供給されます。バックグラウンド・コンフィグレーション用に内部的にオシレータが用いられていれば、同様にファブリックへの供給が停止したり影響されることはありません。MachXO2 にはオシレータが一つしかありませんが、ユーザ用とコンフィグレーション用には異なる分周器とソースから得られるようになっています。

オシレータ出力は分周器を介して配線され、柔軟性のあるクロック周波数ソースを提供します。表 13-15 に、使用可能な出力周波数を示します。

OSCH Primitive Definition

OSCHOSCH プリミティブは、このセクションの定義に従ってデザインのソースコードでインスタンス化できます。図 13-16 と表 13-13 ~ 13-15 に OSCH の定義を示します。

図 13-16. OSCH プリミティブ・シンボル

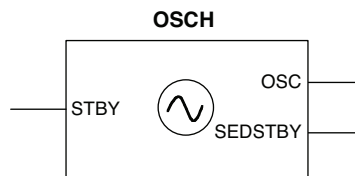


表 13-13. OSCH プリミティブ定義

ポート名	I/O	記 述
STDBY	I	スタンバイ ~ オシレータをスタンバイモードにパワーダウンする - STDBY = 0 : OSC 出力はアクティブ - STDBY = 1 : OSC 出力はオフ
OSC	O	クロック出力ポート
SEDSTDBY	O	スタンバイ ~ SED クロック ¹ をパワーダウン

1. この出力は、デバイスがスタンバイになると、オシレータがシャットダウンすることを SED ブロックに通知する。シミュレーション用としてのみ必要

表 13-14. OSCH プリミティブの属性定義

名 称	表 記	値	デフォルト
Nominal Frequency (MHz)	NOM_FREQ	2.08, 2.15, 2.22, ... 66.5, 88.67, 133.0 (完全なリストは表 13-13 を参照)	2.08 MHz

表 13-15. OSCH でサポートする周波数設定

2.08	4.16	8.31	15.65
2.15	4.29	8.58	16.63
2.22	4.43	8.87	17.73
2.29	4.59	9.17	19.00
2.38	4.75	9.50	20.46
2.46	4.93	9.85	22.17
2.56	5.12	10.23	24.18
2.66	5.32	10.64	26.60
2.77	5.54	11.08	29.56
2.89	5.78	11.57	33.25
3.02	6.05	12.09	38.00
3.17	6.33	12.67	44.33
3.33	6.65	13.30	53.20
3.50	7.00	14.00	66.50
3.69	7.39	14.78	88.67
3.91	7.82	15.65	133.00

NOM_FREQ 属性設定が表の値に一致している必要があります。一致していない場合、警告メッセージが表示され、その属性値は無視されます。

デフォルトでは、内部オシレータを明示的にインスタンスしていなくてもイネーブルされます。オシレータを使用する必要がない場合には、STDBY ポートを使用することでパワーダウンできます。このポートは、ユーザ信号または I/O ピンに接続できます。ユーザは、Wishbone バスの動作、SPI または I²C のパラメータ設定、SPI または I²C のユーザモード動作、バックグラウンド・フラッシュ・アップデート、SED などの各動作が必要な場合に、オシレータがオフになっていないことを確実にする必要があります。

VHDL ソースコードでの OSCH 宣言

ライブラリのインスタンス化

```
library machxo2;
use machxo2.all;
```

コンポーネントと属性の宣言

```
COMPONENT OSCH
-- synthesis translate_off
    GENERIC (NOM_FREQ: string := "2.56");
-- synthesis translate_on
    PORT (
        STDBY      :IN   std_logic;
        OSC         :OUT  std_logic;
        SEDSTDBY   :OUT  std_logic);
END COMPONENT;

attribute NOM_FREQ : string;
attribute NOM_FREQ of OSCinst0 : label is "2.56";
```

OSCH インスタンス化

```
begin
OSCInst0: OSCH
-- synthesis translate_off
    GENERIC MAP ( NOM_FREQ => "2.56" )
-- synthesis translate_on
    PORT MAP ( STDBY=> stdby,
               OSC=> osc_int
               SEDSTDBY=> stdby_sed
               );
```

Verilog ソースコードでの OSCH 宣言

```
// Internal Oscillator
// defparam OSCH_inst.NOM_FREQ = "2.08";// This is the default frequency
// defparam OSCH_inst.NOM_FREQ = "24.18";

OSCH OSCH_inst( .STDBY(1'b0), // 0=Enabled, 1=Disabled
// also Disabled with Bandgap=OFF
               .OSC(osc_clk),
               .SEDSTDBY()); // this signal is not required if not

// using SED
```

テクニカルサポート

e-mail: techsupport@latticesemi.com

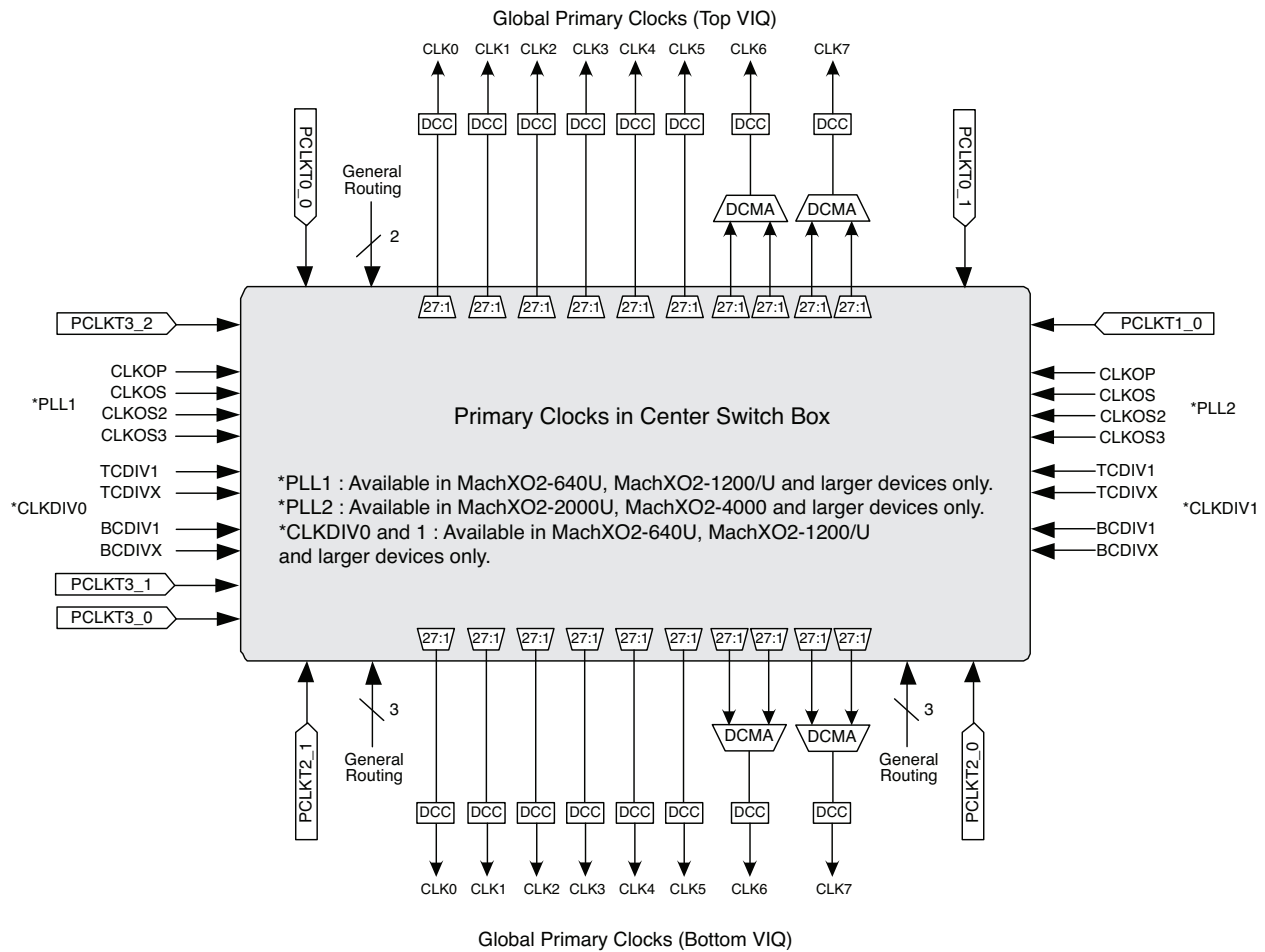
インターネット: www.latticesemi.com

日本語版改版履歴

日付	バージョン	ページ (新)	改訂内容
2010 年 11 月	01.0	-	初版
2011 年 1 月	01.1	-	超多ピン (“U”) デバイスについて更新
2011 年 4 月	01.2	2	MachXO2 クロッキング構造 (MachXO2-1200) 図 ~ バンク 4 を 3 に変更
		1. 11 2. 20-21 3. 30	1. MachXO2 PLL ブロック図を更新。WISHBONE ポート接続を明記。PLLWAKESYNC と INTLOCK 信号記述を削除。2. 周波数計算とフラクショナル N シンセシス章を追加。3. オシレータに Verilog インスタンス例を追加
		29	OSCH サポート周波数設定表を訂正
		39-44	PLL WISHBONE レジスタ記述を追加
2011 年 7 月	01.4	-	VHDL 記述例の訂正 (複数)
		31, 33, 34	付録 A/B 図のクロック名更新 (図 13-17 / 20 / 21)
2011 年 8 月	01.5	11	図 13-8 差し替え、
		37, 38	付録 D, PLL Wishbone 動作に関する記述の更新
2012 年 1 月	01.6	-	1. ドキュメントステータスをアドバンストから正規版に 2. ライブラリインスタンス化の VHDL コードでライブラリ宣言部を全てのマクロについて更新
2012 年 8 月	01.7 ~ 02.1	13	RST Input と RESETM 節、バイパス時の記述更新
		21	Fractional-N 節末尾追記
2012 年 9 月	2.1 - 2.2	13-28	内蔵オシレータ OSCH、記述追加
2013 年 6 月	2.3	13-9	表 13-6、CDIV1 に関する記述追加
		13-12	CLKI パラグラフ、最終 2 行を追加
		13-30	テクニカルサポート、ホットラインを削除
2014 年 6 月	2.4 - 2.5	13-11	冒頭のパラグラフに、バックグラウンド・プログラミングに関する注意を追記
		13-12	先頭行に CLKOS を追加して、修正
		13-14	CLKOS 出力、CLKOS2 出力項にそれぞれパラグラフ追加。低周波数対応の記述
		13-29	STDBY ポートについての記述を更新
		13-37	“USE_EDGE” および “EDGE2EDGE” 削除

付録 A. プライマリクロックのソース及び分配

図 13-17. MachXO2 プライマリクロック・ソースと分配



注 : MachXO2 には 8 本のグローバル・プライマリクロックがあります。各プライマリクロックは、プライマリクロックセンター・スイッチボックスの上辺と下辺にドライブ出力があります。上辺と下辺のドライブは、プライマリクロックごとに同じクロックソースを使用する必要があります。

図 13-18. MachXO2 プライマリクロック・マルチプレクサ ~ MachXO-640U 及び MachXO2-1200/U 以上のデバイス

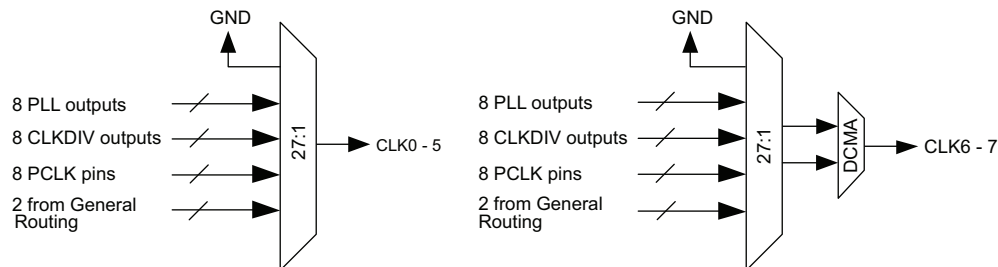
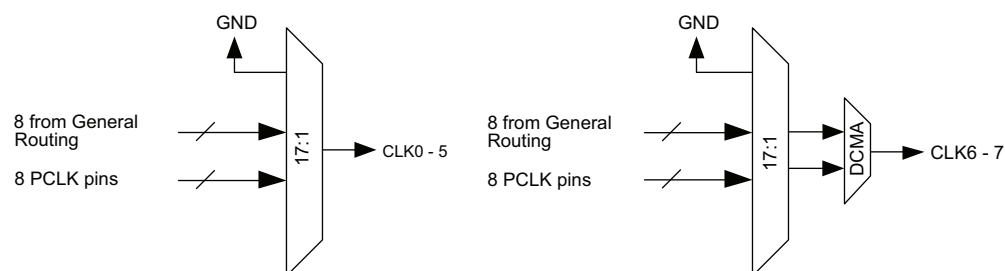
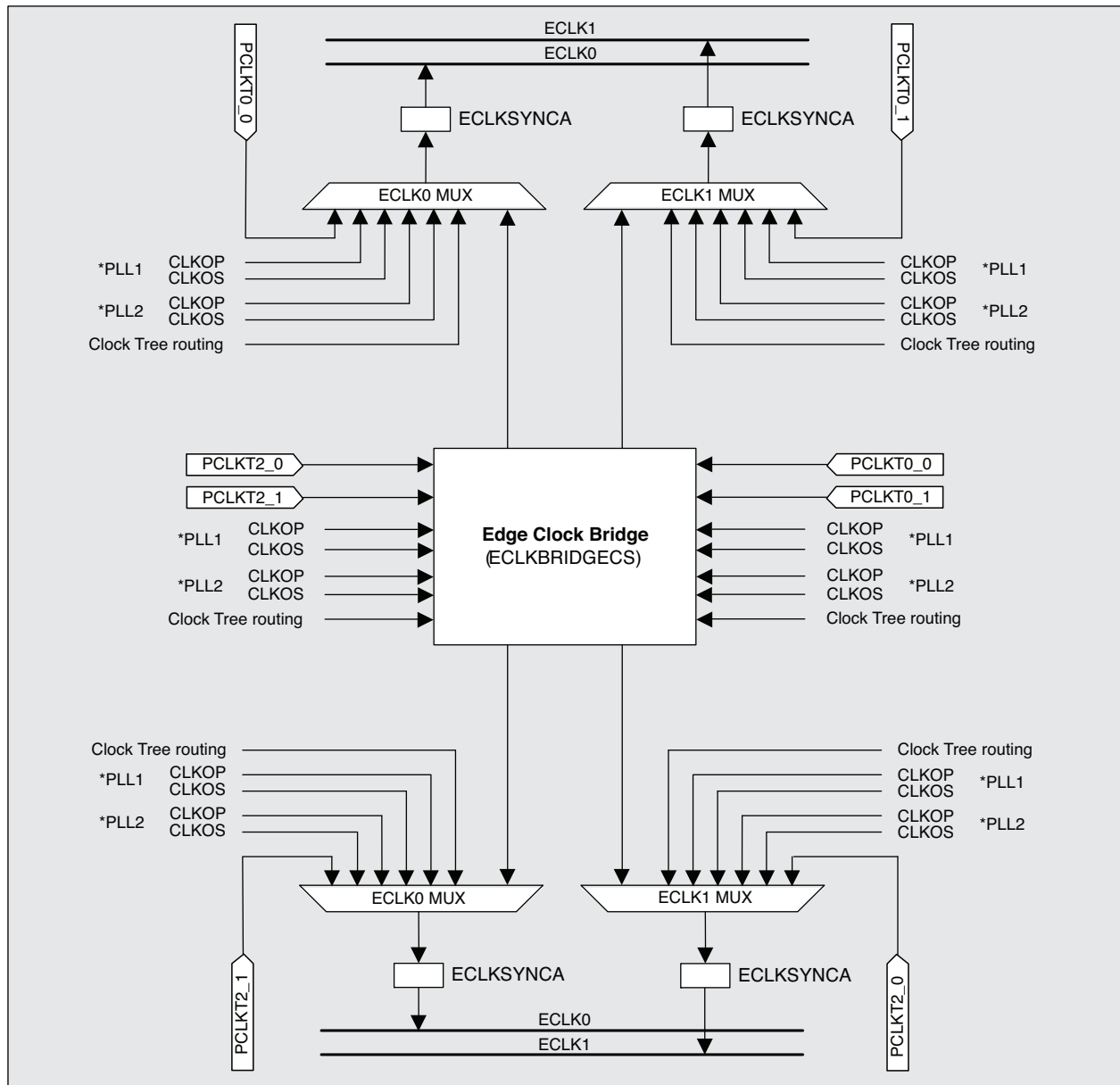


図 13-19. MachXO2 プライマリクロック・マルチプレクサ ～ MachXO2-256 及び MachXO2-640



付録 B. エッジクロック・ソースと接続

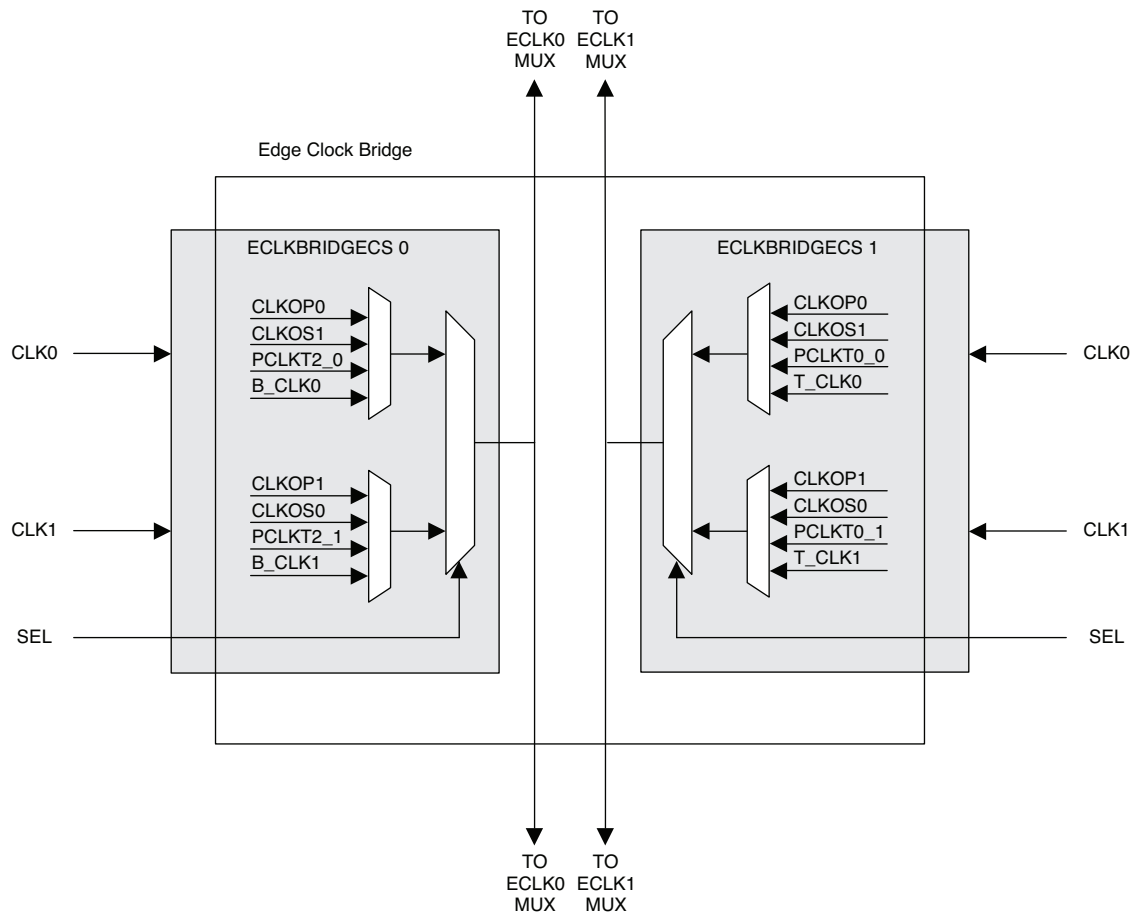
図 13-20. MachXO2 エッジクロック・ソースと接続



注：

1. MachXO2 でエッジクロック・リソースを持つのは、MachXO2-640U 及び MachXO2-1200/U 以上のデバイスのみです
2. エッジクロック・マルチプレクサ ECLK0 MUX 及び ECLK1 MUX は、ソフトウェアが使用する配線リソースです。これらのマルチプレクサには、ダイナミックな入力切り替えはありません。ドライバがエッジクロックをダイナミックに切り替えるには、ECLKBRIDGECS エlement がデザイン内でインスタンス化されている必要があります。

図 13-21. MachXO2 エッジクロック・ブリッジソースと接続



注：

1. エッジクロック・ブリッジにより、1本のクロック信号が上下辺両方のエッジクロックを最小限のスキューでドライブできます。クロックソースの切り替えが必要な場合もエッジクロック・ブリッジを使用できます。
2. エッジクロック・ブリッジリソースは、MachXO2-640U 及び MachXO2-1200/U 以上の規模のデバイスで使用できます。
3. エッジクロック・ブリッジを使用するには、デザイン内で ECLKBRIDGECS プリミティブがインスタンス化されている必要があります。エッジクロック・ブリッジを持つデバイスには、使用可能な ECLKBRIDGECS リソースが 2 つあります。

付録 C. クロックの設計制約

主なクロック設計制約を以下にいくつか紹介します。その他の設計制約及び詳細については、'ヘルプ' ファイルを参照してください。

FREQUENCY

次の物理設計制約は、clk1 というネットに 100 MHz の周波数を割り当てます。

```
FREQUENCY NET "clk1" 100 MHz;
```

次の設計制約は、各クロックドメインのホールドマージン値を指定します。

```
FREQUENCY NET "RX_CLKA_CMOS_c" 100.000 MHz HOLD_MARGIN 1 ns;
```

MAXSKEW

次の設計制約は、NetB というネットに 5 ns の最大スキューを割り当てます。

```
MAXSKEW NET "NetB" 5 NS;
```

MULTICYCLE

次の設計制約は、COMP A から COMP B (NET1) のパスの周期を 50 ns に緩和します。

```
MULTICYCLE "PATH1" START COMP "COMP A" END COMP "COMP B" NET  
"NET1" 50 NS ;
```

PERIOD

次の設計制約は、Clk1 というポートに 30 ns のクロック周期を割り当てます。

```
PERIOD PORT "Clk1" 30 NS;
```

PROHIBIT

次の設計制約は、bf_clk というクロックネットの配線にプライマリクロックを使用することを禁止します。

```
PROHIBIT PRIMARY NET "bf_clk";
```

次の設計制約は、bf_clk というクロックネットの配線にセカンダリ高ファンアウトネットを使用することを禁止します。

```
PROHIBIT SECONDARY NET "bf_clk";
```

PROHIBIT_BOTH

この設定を選択すると、デザインプランナが PROHIBIT PRIMARY NET net_name と PROHIBIT SECONDARY NET net_name の両方を生成します。

USE PRIMARY

指定されたネットの配線にプライマリクロック・リソースを使用します。

```
USE PRIMARY NET clk_fast;  
USE PRIMARY DCCA NET "bf_clk";  
USE PRIMARY PURE NET "bf_clk" QUADRANT_TL;
```

USE SECONDARY

指定されたネットの配線にセカンダリ高ファンアウトネットを使用します。

```
USE SECONDARY NET "clk_lessfast" QUADRANT_TL;
```

CLOCK_TO_OUT

この設計制約は、クロックに相対的な最大許容出力遅延値を指定します。

以下の 2 つの設計制約は CLKPORT と CLKNET の両方のキーワードを使用した例で、それぞれに対応するトレースレポートの範囲も示してあります。

CLKNET は PLL よりも前のパスのタイミング解析を停止するため、PLL 補正タイミング値は得られません。

```
CLOCK_TO_OUT PORT "RxAddr_0" 6.000000 ns CLKNET "pll_rxclk" ;
```

この設計制約では、以下のクロックパスの結果となります。

```
Clock path pll_inst/pll_utp_0_0 to PFU_33:
```

Name	Fanout	Delay (ns)	Site	Resource
ROUTE	49	2.892	ULPPLL.MCLK to	R3C14.CLK0 pll_rxclk

2.892 (0.0% logic, 100.0% route), 0 logic levels.				

CLKPORT が用いられると、クロックポート・リソースまでのタイミング解析を行うため、PLL 補正タイミング値が得られます。

```
CLOCK_TO_OUT PORT "RxAddr_0" 6.000000 ns CLKPORT "RxClk" ;
```

この設計制約では、以下のクロックパスの結果となります。

```
Clock path RxClk to PFU_33:
```

Name	Fanout	Delay (ns)	Site	Resource
IN_DEL	---	1.431	D5.PAD to	D5.INCK RxClk
ROUTE	1	0.843	D5.INCK to	ULPPLL.CLKIN RxClk_c
MCLK_DEL	---	3.605	ULPPLL.CLKIN to	ULPPLL.MCLK
pll_inst/pll_utp_0_0				
ROUTE	49	2.892	ULPPLL.MCLK to	R3C14.CLK0 pll_rxclk

8.771 (57.4% logic, 42.6% route), 2 logic levels.				

INPUT_SETUP

次の設計制約は、クロックネットに対して相対的な、入力ポートのセットアップ時間要件を指定します。

```
INPUT_SETUP PORT "datain" 2.000000 ns HOLD 1.000000 ns CLKPORT "clk"
PLL_PHASE_BACK ;
```

PLL_PHASE_BACK

次の設計制約は、直前のクロックエッジに基づいてトレースを計算する必要がある場合に、INPUT_SETUP と組み合わせて使用します。

この設計制約は、PLL 出力の位相調整の設定に役立ちます。負の位相調整機能は用意されていませんが、PLL_PHASE_BACK 設計制約は、負の位相調整機能が可能であるかのように動作します。

例えば：

CLKOS に -90° の位相調整が求められる場合、ユーザは位相を 270° に設定することで、INPUT_SETUP 設計制約を PLL_PHASE_BACK で設定できます。

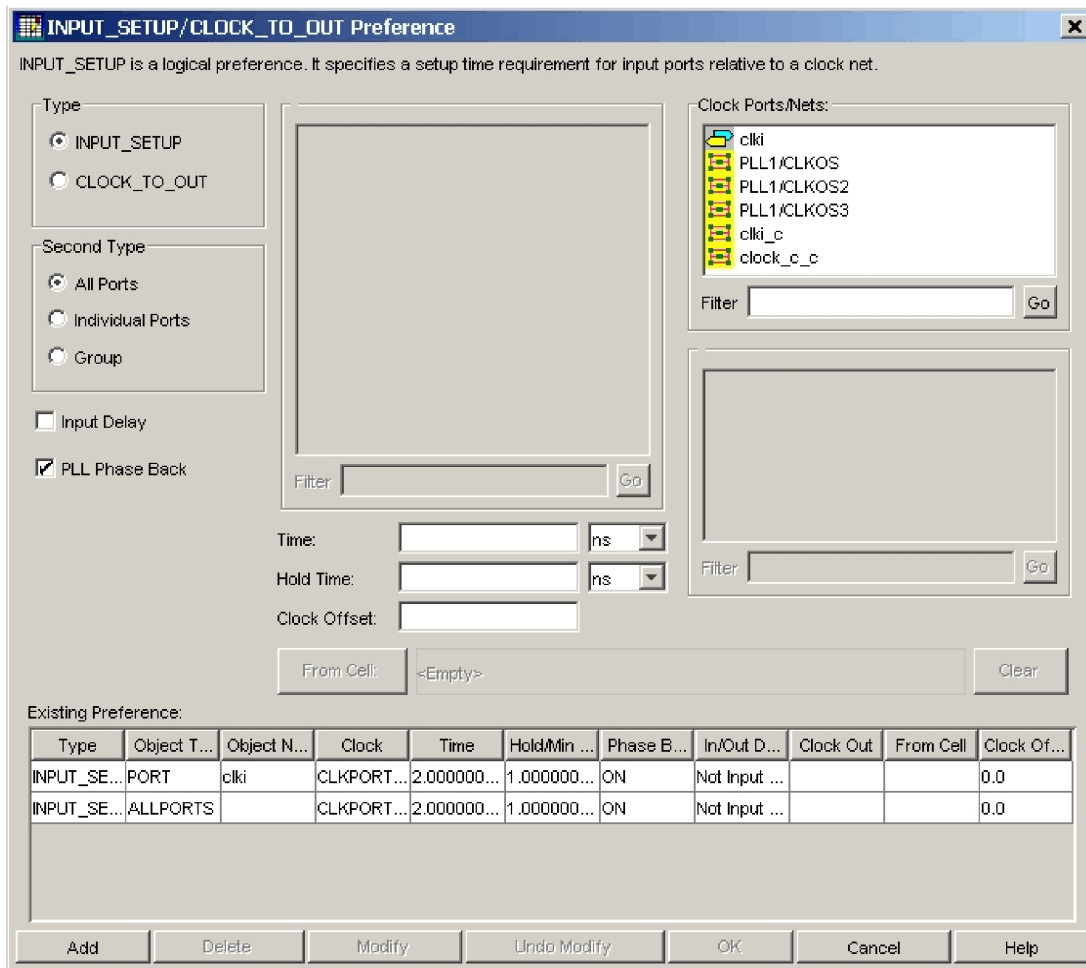
プリマップ・プリファレンスエディタでの PLL_PHASE_BACK の使用法

プリマップ・プリファレンスエディタを使用すると、PLL_PHASE_BACK 属性を設定できます。

4. デザインプランナ (Pre-Map) を開く
5. デザインプランナ・コントロール・ウィンドウで、**View → Spreadsheet View** の順に選択する
6. Spreadsheet View ウィンドウで、**Input_setup/Clock_to_out...** を選択する

負側 PLL 位相調整 (Phase Back) 機能の INPUT_SETUP/CLOCK_TO_OUT プリファレンス・ウィンドウ例を図 13-22 の示します。

図 13-22. Input_SETUP/CLOCK_to_OUT プリファレンス・ウィンドウ



INPUT_SETUP is a logical preference. It specifies a setup time requirement for input ports relative to a clock net.

Type

☒ INPUT_SETUP

☐ CLOCK_TO_OUT

Second Type

☒ All Ports

☐ Individual Ports

☐ Group

☐ Input Delay

☒ PLL Phase Back

Filter

Go

Time: ns

Hold Time: ns

Clock Offset:

From Cell: <Empty> Clear

Existing Preference:

Type	Object T...	Object N...	Clock	Time	Hold/Min ...	Phase B...	In/Out D...	Clock Out	From Cell	Clock Of...
INPUT_SE...	PORT	clki	CLKPORT...	2.000000...	1.000000...	ON	Not Input ...			0.0
INPUT_SE...	ALLPORTS		CLKPORT...	2.000000...	1.000000...	ON	Not Input ...			0.0

Add Delete Modify Undo Modify OK Cancel Help

付録 D. PLL WISHBONE バスの動作

EFB の WISHBONE バスを介して PLL の動作パラメータをダイナミックに変更できます。この機能を使用するためには、デザイン内で EFB ブロックをインスタンス化する必要があります。そしてユーザロジックの WISHBONE バスを EFB ブロックに接続します。EFB と PLL は PLL データバスの専用配線で接続されます。デザイン内で EFB ブロックを使用する方法の詳細については、“Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (ユーザフラッシュメモリとハードマクロ化制御機能の使用法) (TN1205)” を参照してください。

PLL モジュール上の PLL データバスは、この動作の機能シミュレーションに対応しています。シミュレーションを適切に行うには、HDL デザイン内で EFB ブロックの PLL バスに PLL データバスを接続する必要があります。表 13-16 に WISHBONE ポート及び対応する PLL バスポートを示します。

表 13-16. PLL WISHBONE のポート定義

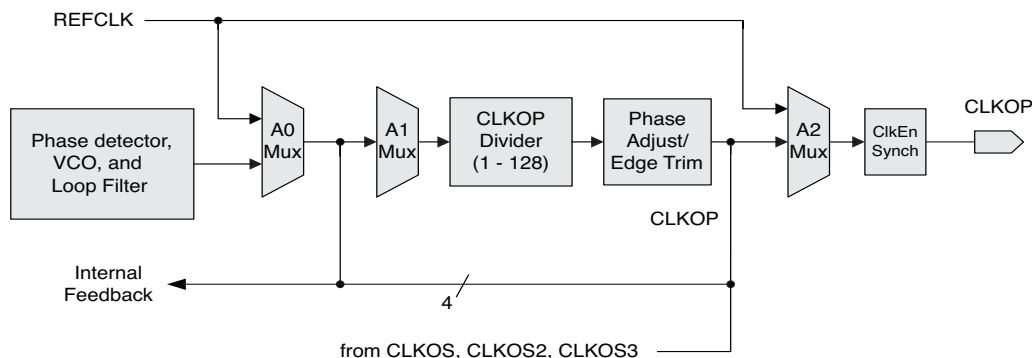
ポート名	I/O	記 述	EFB ポート名
PLLCLK	I	PLL データバス、クロック	pll_bus_o[16]
PLL_RST	I	PLL データバス、リセット。データバスのみをリセットし、レジスタ値はリセットしない	pll_bus_o[15]
PLLSTB	I	PLL データバス、ストロブ信号	pll_bus_o[14]
PLLWE	I	PLL データバス、ライトイネーブル信号	pll_bus_o[13]
PLLADDR [4:0]	I	PLL データバス、アドレス	pll_bus_o[12:8]
PLLDATI [7:0]	I	PLL データバス、データ入力	pll_bus_o[7:0]
PLLDATO [7:0]	O	PLL データバス、データ出力	pll_bus_i[8:1]
PLLACK	O	PLL データバス、アクノレッジ信号	pll_bus_i[0]

PLL アーキテクチャ

MachXO2 には、各種のアプリケーションに対応できるように、構成を柔軟に設定可能な 4 つの出力セクションがあります。IPexpress は一般的な PLL 構成のほとんどに対応できますが、さらに複雑なニーズを持つユーザは、より高度なサポートオプションを可能にする WISHBONE バスを使用して PLL 構成を変更できます。

4 つの PLL 出力セクションは、それぞれ同様の構成オプションを持ちます。各出力セクションには文字識別子が割り当てられ、CLKOP 出力セクションは A、CLKOS 出力セクションは B、CLKOS2 出力セクションは C、CLKOS3 出力セクションは D となっています。4 つの各出力セクションには、PLL 構成の制御に使用される 3 つの信号選択マルチプレクサがあります。A 出力セクションの図を図 13-23 に示します。B 出力セクションは、マルチプレクサのラベルが B0、B1、及び B2 となる以外は A セクションと同じです。C 及び D セクションは同様にマルチプレクサのラベルが C0、C1、C2、D0、D1、及び D2 となります。C 及び D セクションには位相調整ブロックはありますが、エッジトリム機能はありません。

図 13-23. PLL の CLKOP 出力セクション



PLL レジスタに対する EFB WISHBONE レジスタマップを表 13-17 に示します。MachXO2 ないで第二の PLL を用いる場合は、表の対応するアドレスに 0x20 を加算します。

表 13-17. PLL レジスタ用 EFB WISHBONE レジスタマップ

Reg.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	MC1_DIVFBK_FRAC[7:0]							
1	MC1_DIVFBK_FRAC[15:8]							
2	MC1_LOADREG	MC1_DELA[6:0]						
3	MC1_PLLPDN	MC1_DELB[6:0]						
4	MC1_WBRESET	MC1_DELC[6:0]						
5	MC1_USE_DESI	MC1_DELD[6:0]						
6	MC1_REFIN_RESET	MC1_DIVA[6:0]						
7	MC1_PLLRST_ENA	MC1_DIVB[6:0]						
8	MC1_MRST_ENA	MC1_DIVC[6:0]						
9	MC1_STDBY	MC1_DIVD[6:0]						
A	MC1_ENABLE_SYNC	MC1_PHIB[2:0]			MC1_INT_LOCK_STICKY	MC1_PHIA[2:0]		
B	MC1_DCRST_ENA	MC1_PHID[2:0]			MC1_RESERVED_2	MC1_PHIC[2:0]		
C	MC1_DDRST_ENA	MC1_SEL_OUTB[2:0]			MC1_INTFB	MC1_SEL_OUTA[2:0]		
D	MC1_LOCK[1:0]		MC1_SEL_OUTC[2:0]			MC1_SEL_OUTD[2:0]		
E	MC1_SEL_DIVA[1:0]		MC1_SEL_DIVB[1:0]		MC1_SEL_DIVC[1:0]		MC1_SEL_DIVD[1:0]	
F	MC1_CLKOP_TRIM[3:0]				MC1_CLKOS_TRIM[3:0]			
10	MC1_DYN_SOURCE	MC1_LOCK_SEL[2:0]			MC1_ENABLE_CLK[3:0]			
11	MC1_TRIMOS3_BYPASS_N	MC1_TRIMOS2_BYPASS_N	MC1_TRIMOS_BYPASS_N	MC1_TRIMOP_BYPASS_N	MC1_DYN_SEL[1:0]		MC1_DIRECTION	MC1_ROTATE
12	MC1_LF_RESGRND	MC1_SEL_REF1[2:0]			MC1_EN_UP	MC1_SEL_REF2[2:0]		
13	MC1_DIVFBK_ORDER[1:0]		MC1_CLKMUX_FB[1:0]		MC1_SEL_FBK[3:0]			
14	MC1_GMC_RESET	MC1_DIVREF[6:0]						
15	MC1_FORCE_VFILTER	MC1_DIVFBK[6:0]						
16	MC1_LF_PRESET	MC1_LF_RESET	MC1_TEST_ICP	MC1_EN_FILTER_OPAMP	MC1_FLOAT_ICP	MC1_GPROG[2:0]		
17	MC1_KPROG[2:0]			MC1_IPROG[4:0]				
18	MC1_GMC_PRESET	MC1_RPROG[6:0]						
19	MC1_GMCREF_SEL[1:0]		MC1_MFGOUT2_SEL[2:0]			MC1_MFGOUT1_SEL[2:0]		
1A	MC1_GMCSEL[3:0]				MC1_VCO_BYPASS_D0	MC1_VCO_BYPASS_C0	MC1_VCO_BYPASS_B0	MC1_VCO_BYPASS_A0
1B	MC1_RESERVED[4:0]					MC1_EN_PHI	MC1_DPROG[1:0]	
1C	RESERVED							LOCK_STS

注：レジスタ 0 ～ 11 はユーザがアクセス可能なレジスタ。残りのレジスタは Lattice 用、または読み取り専用アクセス用に予約済み

表 13-18. PLL WISHBONE レジスタ定義

レジスタ名	レジスタアドレス (Hex)	サイズ (Bits)	記 述	デフォルト値	ユーザアクセス	GUI アクセス
MC1_DIVFBK_FRAC[15:0]	0[7:0] 1[7:0]	16	フラクショナル N 分周器の値。実際の分周値はこの値を 65535 で割ったもの	0	Yes	Yes
MC1_LOADREG	2[7]	1	有効なのは MC1_DYN_PHASE=0 の場合のみ。 MC1_LOADREG ビットの立ち下がりエッジで分周器出力の位相シフトを開始するコマンド CLKOP の位相シフトは MC1_DIVA と MC1_DELA 値が同一でない時に実行。CLKOS の位相シフトは MC1_DIVB と MC1_DELB 値が同一でない時に、CLKOS2 の位相シフトは MC1_DIVC と MC1_DELC 値が同一でない時に、CLKOS3 の位相シフトは MC1_DIVD と MC1_DELD 値が同一でない時にそれぞれ実行される	0	Yes	N/A
MC1_PLLPDN	3[7]	1	PLL 未使用の時にパワーダウン。PLL が使用されている場合、ソフトウェアが自動的に '1' に、使用されていないときは '0' にセットする 0 = PLL をパワーダウン 1 = PLL はパワーアップ	1	Yes	Yes、自動
MC1_WBRESET	4[7]	1	Wishbone からの PLL リセット。RESETM ポート動作と等価 0 = PLL 通常動作 1 = PLL リセットがアクティブ	0	Yes	No
MC1_USE_DESI	5[7]	1	フラクショナル N 分周器を使用するかどうかを制御 divider is used. 0 = PLL 通常動作 1 = フラクショナル N 分周器を使用	0	Yes	Yes
MC1_REFIN_RESET	6[7]	1	入力基準クロックが PLLREFCS プリミティブを用いて切り替えられる時に PLL を自動的にリセットするかどうかを制御 0 = PLL をリセットしない 1 = 入力切り替え時に自動的に PLL をリセット	0	Yes	No
MC1_PLLRST_ENA	7[7]	1	PLLRESET ポートをイネーブル 0 = PLLRESET ポートは非アクティブ 1 = PLLRESET ポートはイネーブル	0	Yes	Yes
MC1_MRST_ENA	8[7]	1	RESETM ポートをイネーブル 0 = RESETM ポートは非アクティブ 1 = RESETM ポートはイネーブル	0	Yes	Yes
MC1_STDBY	9[7]	1	PLL の STDBY ポートをイネーブル 0 = STDBY ポートは非アクティブ 1 = STDBY ポートはイネーブル	0	Yes	Yes
MC1_ENABLE_SYNC	A[7]	1	CLKOP に対して同期して CLKOS, CLKOS2, CLKOS3 をディセーブル / イネーブルする 0 = 同期ディセーブル / イネーブルは非アクティブ 1 = 同期ディセーブル / イネーブルはアクティブ	0	Yes	No
MC1_DCRST_ENA	B[7]	1	RESETDC ポートをイネーブル ~ CLKOS2 リセット 0 = RESETDC ポートは非アクティブ 1 = RESETDC ポートはイネーブル	0	Yes	Yes
MC1_DDRST_ENA	C[7]	1	RESETDD ポートをイネーブル ~ CLKOS3 リセット 0 = RESETDD ポートは非アクティブ 1 = RESETDD ポートはイネーブル	0	Yes	Yes

表 13-18. PLL WISHBONE レジスタ定義 (Continued)

レジスタ名	レジスタアドレス (Hex)	サイズ (Bits)	記 述	デフォルト値	ユーザアクセス	GUI アクセス
MC1_DELA[6:0]	2[6:0]	7	位相の粗調整時の CLKOP セクションの遅延値。ゼロ遅延のためには本値は MC1_DIVA[6:0] に等しいこと	7	Yes	Yes
MC1_DELB[6:0]	3[6:0]	7	位相の粗調整時の CLKOS セクションの遅延値。ゼロ遅延のためには本値は MC1_DIVB[6:0] に等しいこと	7	Yes	Yes
MC1_DELC[6:0]	4[6:0]	7	位相の粗調整時の CLKOS2 セクションの遅延値。ゼロ遅延のためには本値は MC1_DIVC[6:0] に等しいこと	7	Yes	Yes
MC1_DELD[6:0]	5[6:0]	7	位相の粗調整時の CLKOS3 セクションの遅延値。ゼロ遅延のためには本値は MC1_DIVD[6:0] に等しいこと	7	Yes	Yes
MC1_DIVA[6:0]	6[6:0]	7	CLKOP セクションの出力分周器設定で、“分周値 - 1” に等しい。	7	Yes	Yes
MC1_DIVB[6:0]	7[6:0]	7	CLKOS セクションの出力分周器設定で、“分周値 - 1” に等しい	7	Yes	Yes
MC1_DIVC[6:0]	8[6:0]	7	CLKOS2 セクションの出力分周器設定で、“分周値 - 1” に等しい	7	Yes	Yes
MC1_DIVD[6:0]	9[6:0]	7	CLKOS3 セクションの出力分周器設定で、“分周値 - 1” に等しい	7	Yes	Yes
MC1_PHIA[2:0]	A[2:0]	3	CLKOP に対する VCO 位相シフト (0 ~ 7) を選択。各タップは 45 度に相当	0	Yes	Yes
MC1_PHIB[2:0]	A[6:4]	3	CLKOS に対する VCO 位相シフト (0 ~ 7) を選択。各タップは 45 度に相当	0	Yes	Yes
MC1_PHIC[2:0]	B[2:0]	3	CLKOS2 に対する VCO 位相シフト (0 ~ 7) を選択。各タップは 45 度に相当	0	Yes	Yes
MC1_PHID[2:0]	B[6:4]	3	CLKOS3 に対する VCO 位相シフト (0 ~ 7) を選択。各タップは 45 度に相当	0	Yes	Yes
MC1_INT_LOCK_STICKY	A[3]	1	内部ロックがスティッキーかどうかを設定。スティッキーでは一度ロックすると、PLL がリセットかパワーダウンされるまで High のまま。PLL は内部ロックは使用しない 0 = 内部ロックは通常動作 1 = 内部ロックはスティッキー動作	1	Yes	Not used
MC1_RESERVED2	B[3]	1	未使用	N/A	N/A	N/A
MC1_SEL_OUTA[2:0]	C[2:0]	3	CLKOP 出力用 Mux A2 の選択値で、必要な場合は分周器を従属接続可能 000 = CLKOP を DIVA 出力にする 001 = CLKOP を DIVB 出力にする 010 = CLKOP を DIVC 出力にする 011 = CLKOP を DIVD 出力にする 100 = CLKOP を REFCLK 出力にする (クロック分周器を用いないバイパスモードと同じ) これ以外の値はラティス用に予約	000	Yes	No
MC1_SEL_OUTB[2:0]	C[6:4]	3	CLKOS 出力用 Mux B2 の選択値で、必要な場合は分周器を従属接続可能 000 = CLKOS を DIVB 出力にする 001 = CLKOS を DIVC 出力にする 010 = CLKOS を DIVD 出力にする 011 = CLKOS を DIVA 出力にする 100 = CLKOS を REFCLK 出力にする (クロック分周器を用いないバイパスモードと同じ) これ以外の値はラティス用に予約	000	Yes	No

表 13-18. PLL WISHBONE レジスタ定義 (Continued)

レジスタ名	レジスタアドレス (Hex)	サイズ (Bits)	記 述	デフォルト値	ユーザアクセス	GUI アクセス
MC1_SEL_OUTC[2:0]	D[5:3]	3	CLKOS2 出力用 Mux C2 の選択値で、必要な場合は分周器を従属接続可能 000 = CLKOS2 を DIVC 出力にする 001 = CLKOS2 を DIVD 出力にする 010 = CLKOS2 を DIVA 出力にする 011 = CLKOS2 を DIVB 出力にする 100 = CLKOS2 を REFCLK 出力にする (クロック分周器を用いないバイパスモードと同じ) これ以外の値はラティス用に予約	000	Yes	No
MC1_SEL_OUTD[2:0]	D[2:0]	3	CLKOS3 出力用 Mux D2 の選択値で、必要な場合は分周器を従属接続可能 000 = CLKOS3 を DIVD 出力にする 001 = CLKOS3 を DIVA 出力にする 010 = CLKOS3 を DIVB 出力にする 011 = CLKOS3 を DIVC 出力にする 100 = CLKOS3 を REFCLK 出力にする (クロック分周器を用いないバイパスモードと同じ) これ以外の値はラティス用に予約	000	Yes	No
MC1_INTFB	C[3]	1	PLL 内部フィードバックを初期 PLL ロック動作に使用する。INTLOCK と PLLWAKESYNC ポートと共に使用される。 本ビットは変更しないことを推奨 0 = PLL 内部フィードバックを使用しない 1 = PLL 内部フィードバックを使用する	0	Yes	No
MC1_LOCK[1:0]	D[7:6]	2	周波数ロック検出器の精度、或いは感度 00 = +/- 250 ppm 01 = +/- 1000 ppm 10 = +/- 4000 ppm 11 = +/- 16000 ppm	00	Yes	No
MC1_SEL_DIVA[1:0]	E[7:6]	2	DIVA (CLKOP) への入力 Mux A1 選択値。望む場合、分周器を従属接続可能 00 = MUX A0 出力 01 = DIVD (CLKOS3) 出力 10 = DIVB (CLKOS) 出力 11 = DIVC (CLKOS2) 出力	00	Yes	No
MC1_SEL_DIVB[1:0]	E[5:4]	2	DIVB (CLKOS) への入力 Mux B1 選択値。望む場合、分周器を従属接続可能 00 = MUX B0 出力 01 = DIVA (CLKOP) 出力 10 = DIVD (CLKOS3) 出力 11 = DIVC (CLKOS2) 出力	00	Yes	No
MC1_SEL_DIVC[1:0]	E[3:2]	2	DIVC (CLKOS2) への入力 Mux C1 選択値。望む場合、分周器を従属接続可能 00 = MUX C0 出力 01 = DIVA (CLKOP) 出力 10 = DIVB (CLKOS) 出力 11 = DIVD (CLKOS3) 出力	00	Yes	No
MC1_SEL_DIVD[1:0]	E[1:0]	2	DIVD (CLKOS3) への入力 Mux D1 選択値。望む場合、分周器を従属接続可能 00 = MUX D0 出力 01 = DIVA (CLKOP) 出力 10 = DIVB (CLKOS) 出力 11 = DIVC (CLKOS2) 出力	00	Yes	No

表 13-18. PLL WISHBONE レジスタ定義 (Continued)

レジスタ名	レジスタアドレス (Hex)	サイズ (Bits)	記 述	デフォルト値	ユーザアクセス	GUI アクセス
MC1_CLKOP_TRIM[3:0]	F[7:4]	4	CLKOP 出力トリミング制御。TRIM[3:0] のビット 3 が影響するエッジを設定 TRIM[3] = 0 立ち下がりエッジでトリムがアクティブ TRIM[3] = 1 立ち上がりエッジでトリムがアクティブ TRIM[2:0] はワンホット信号 TRIM[2:0] = 001 で 70 ps トリム TRIM[2:0] = 010 で 140 ps トリム TRIM[2:0] = 100 で 280 ps トリム	0000	Yes	Yes
MC1_CLKOS_TRIM[3:0]	F[3:0]	4	CLKOS 出力トリミング制御。TRIM[3:0] のビット 3 が影響するエッジを設定 TRIM[3] = 0 立ち下がりエッジでトリムがアクティブ TRIM[3] = 1 立ち上がりエッジでトリムがアクティブ TRIM[2:0] はワンホット信号 TRIM[2:0] = 001 で 70 ps トリム TRIM[2:0] = 010 で 140 ps トリム TRIM[2:0] = 100 で 280 ps トリム	0000	Yes	Yes
MC1_ENABLE_CLK[3:0]	10[3:0]	4	各 PLL 出力ポートのクロックイネーブル。本設定は対応するイネーブルポート信号と OR され、クロック出力イネーブルとなる。ソフトウェアが GUI 設定に基づいて自動的に設定する。 本ビットは変更しないことを推奨 xxx1 = CLKOP をイネーブル xx1x = CLKOS をイネーブル x1xx = CLKOS2 をイネーブル 1xxx = CLKOS3 をイネーブル	0001	Yes	Yes
MC1_LOCK_SEL[2:0]	10[6:4]	3	ロック検出器動作モード、通常かスティッキーか。後者は一度ロックすると、PLL がリセットかパワーダウンされるまで High のまま 000 = PLL ロックは通常動作 001 = PLL ロックはスティッキー動作 100 = もう一つの PLL ロック通常動作 これ以外は非対応モード	000	Yes	Yes
MC1_DYN_SOURCE	10[7]	1	Wishbone や外部ポートがダイナミック位相設定を制御するかどうかを指定 0 = Wishbone レジスタが制御する 1 = 外部ポートが制御する	1	Yes	間接
MC1_DIRECTION	11[1]	1	有効なのは MC1_DYN_PHASE=0 の場合のみ。MC1_ROTATE コマンドでダイナミック位相変化の方向を指定 0 = 位相を遅れ位相に回転 1 = 位相を進み位相に回転	0	Yes	非該当
MC1_ROTATE	11[0]	1	有効なのは MC1_DYN_PHASE=0 の場合のみ。現 VCO 位相から遅れ、又は進み位相への変更を開始するコマンド。位相は MC1_ROTATE ビットの負のエッジで変わる。各ステップは VCO 位相の 45 度に相当。(MC1_ROTATE は PHASESTEP 信号に等価)	0	Yes	非該当

表 13-18. PLL WISHBONE レジスタ定義 (Continued)

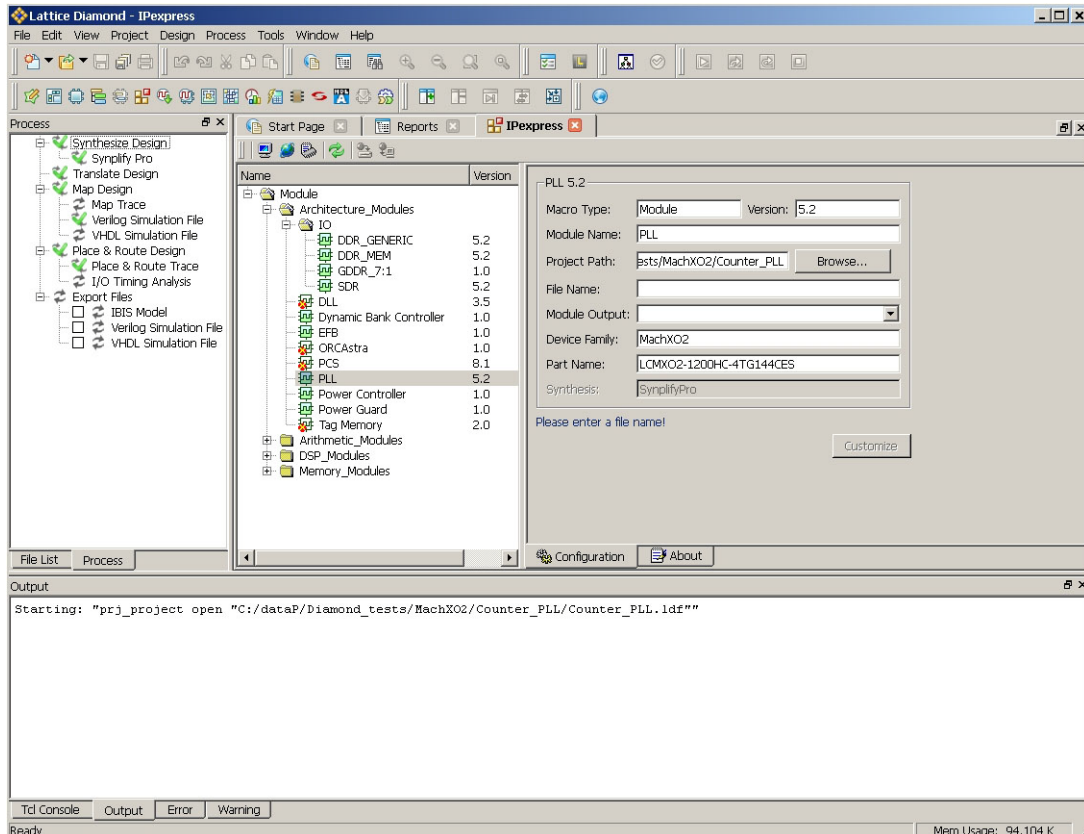
レジスタ名	レジスタアドレス (Hex)	サイズ (Bits)	記 述	デフォルト値	ユーザアクセス	GUI アクセス
MC1_DYN_SEL[1:0]	11[3:2]	2	有効なのは MC1_DYN_PHASE=0 の場合のみ。ダイナミック位相制御がどのポートで制御されるかを指定 00 = CLKOS をイネーブル 01 = CLKOS2 をイネーブル 10 = CLKOS3 をイネーブル 11 = CLKOP をイネーブル	00	Yes	非該当
MC1_TRIMOP_BYPASS_N	11[4]	1	CLKOP 出力トリム回路をバイパスするかどうかを設定 0 = トリム回路をバイパス 1 = トリム回路をバイパスしない	0	Yes	間接
MC1_TRIMOS_BYPASS_N	11[5]	1	CLKOS 出力トリム回路をバイパスするかどうかを設定 0 = トリム回路をバイパス 1 = トリム回路をバイパスしない	0	Yes	間接
MC1_TRIMOS2_BYPASS_N	11[6]	1	CLKOS2 出力トリムをバイパスするビット。CLKOS2 にはトリム制御がない。CLKOP や CLKOS 出力のトリムがアクティブの場合に CLKOP, CLKOS, CLKOS2, & CLKOS3 出力と遅延を等しくするためにダミーのトリム回路がある 0 = トリム回路をバイパス 1 = トリム回路をバイパスしない	0	Yes	間接
MC1_TRIMOS3_BYPASS_N	11[7]	1	CLKOS3 出力トリムをバイパスするビット。CLKOS3 にはトリム制御がない。CLKOP や CLKOS 出力のトリムがアクティブの場合に CLKOP, CLKOS, CLKOS2, & CLKOS3 出力と遅延を等しくするためにダミーのトリム回路がある 0 = トリム回路をバイパス 1 = トリム回路をバイパスしない	0	Yes	間接

付録 E. Lattice Diamond デザインソフトウェアでの MachXO2 デバイスの使用法

Lattice Diamond ソフトウェアで MachXO2 デバイスを使用する場合、本書の図 13-12 と 13-14 に示したスクリーンショットとは多少の違いがあります。図 13-13 は Diamond でも ispLEVER の場合と同じです。

IPexpress を使用して Diamond から PLL を構成する場合、ユーザはファイル名を指定し、モジュール出力タイプとして VHDL または Verilog も選択する必要があります。モジュール出力タイプの選択には、プルダウン選択ボックスを使用します。図 13-24 に Diamond の画面例を示します。

図 13-24. Diamond を使用した PLL モジュールの IPexpress メインウィンドウ



ファイル名と出力タイプを入力した後、**Customize** ボタンをクリックすることで Configuration タブ・ウィンドウが図 13-13 で示すようにオープンします。

Diamond を用いてプライマリやセカンダリ、またはエッジ各クロックに対するクロック制約を設定する場合、**Spreadsheet View** (スプレッドシート・ビュー) を開いて **Clock Resource** (クロックリソース) タブを選択します。次に、目的のクロック信号の選択ウィンドウ内を右クリックして、プルダウンメニューから該当するクロック設計制約を選択します。Diamond の画面例をを図 13-25 に示します。

図 13-25. Diamond を使用した Spreadsheet View でのクロック選択

