



ALL PROGRAMMABLE™

XAPP1240 (v1.0) 2015 年 4 月 17 日

20 ビット幅のオーバーサンプル データをベースとするクロックおよびデータリカバリ ユニット

著者 : Paolo Novellini、Antonello Di Fresco、Giovanni Guasti

概要

マルチサービス ネットワークには、広範な入力データ レートで動作可能なトランシーバーが必要です。高速シリアル I/O は、動作データ レートに下限があるため、低速クライアント信号へのインターフェイスが容易ではありません。このアプリケーション ノートで説明する NIDRU (Non-Integer Data Recovery Unit) は、データ レートの下限を 0Mb/s に引き下げることによって、専用の高速トランシーバーを真のマルチレート シリアル インターフェイスに対応できる有効なソリューションにします。NIDRU は、ザイリンクスの 7 シリーズおよび UltraScale™ デバイス用に設計されています。NIDRU の動作設定 (データ レート、ジッター帯域幅、入力 ppm 範囲、ジッター ピーキング) は動的にプログラム可能なため、ビットストリームのリロードやパーシャル リコンフィギュレーションは必要ありません。同期する外部基準クロックで動作している場合、NIDRU では小数値のオーバーサンプリング レートがサポートされます。したがって、セットアップされているチャンネル数に関係なく、またすべてのチャンネルが異なるデータ レートで動作している場合であっても、必要なクロック ツリーは 1 つのみ (BUFG または BUFG_GT) です。

抽出されたデータは、ユーザー アプリケーションへ同時に転送されます。このパス幅はユーザーがプログラムできるため、8 ビットまたは 10 ビットのいずれのユーザー アプリケーションへも容易に接続できます。

概要

このアプリケーション ノートは、次の 3 つの内容で構成されています。

- [「NIDRU のブロック図およびピン配置」](#)
NIDRU ラッパーの構造およびピン配置について説明しています。
- [「NIDRU の使用モデル」](#)
NIDRU のポートおよび属性の設定方法について説明しています。
- [「DRU のシミュレーション」](#)
DRU のシミュレーション用に設計されたテストベンチについて説明しています。

このアプリケーション ノートでは、実例として Fast Ethernet (125Mb/s ± 100ppm) および OC3/STM1 (155.520Mb/s ± 20ppm) を使用します。

NIDRU のブロック図およびピン配置

このセクションでは、NIDRU ラッパーの構造およびピン配置について説明します。図 1 に、ラッパー構造を示します。関連ポートのみ表示しています。

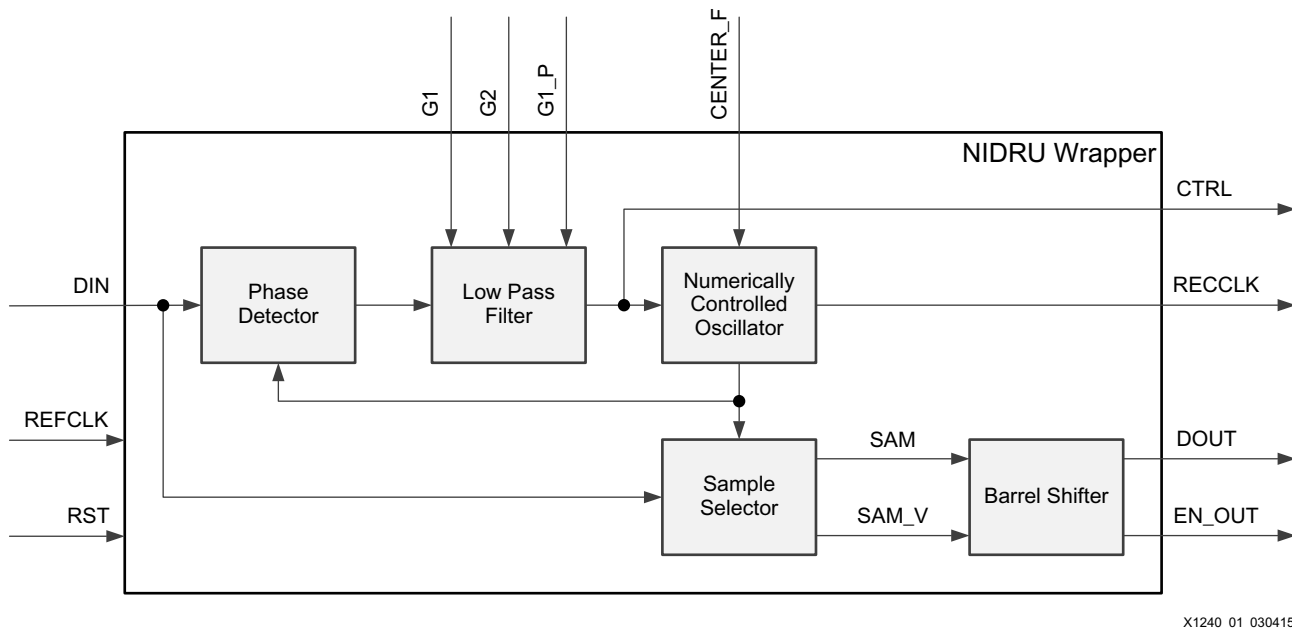


図 1: NIDRU ラッパーのブロック図

DIN ポートでは、SelectIO™ インターフェイスまたは Lock-to-Reference モードに設定された SerDes からオーバーサンプルされた未処理の 20 ビット幅データを受信します。NIDRU のビット順序変換は SerDes と同じで、最初のビットが LSb になります¹⁾。Phase Detector (PD) では入力データの遷移が監視され、内部の NCO (数値制御型オシレーター) の位相と継続的に照合されます。デジタル エラー信号が PD で生成されて Low Pass Filter (LP) でフィルタリングされ、これを使用して NCO 周波数が調整されて位相エラーが最小化します。このようにして、NIDRU の PLL (位相ロックループ) が機能します ([参照 1] および [参照 2])。Sample Selector (SS) ブロックは、NCO の出力に基づいて、アイの中央に最も近いサンプル値を選択します。REFCLK クロックの 1 サイクル間には、最大 10 の有効なサンプル値があり、SS によって SAM 出力に配置されます。SAMV は、各クロック サイクルで SAM に配置された有効なサンプル値の数を示します。NIDRU とユーザーアプリケーション間の接続をシンプルにするため、ラッパー内に Barrel Shifter (パレルシフター) があります。このブロックの出力データ幅は WDT_OUT 属性で指定できます。NIDRU ラッパー内のすべてのブロックは REFCLK に同期します。

1. Virtex®-4 デバイス以降、すべての SerDes は、ここで説明するとおり NIDRU と同じ順序規則に従います。Virtex-II Pro デバイスの SerDes では、異なるビット順序を採用しています。

NIDRU は、入力されるデータを同時に処理するため、通常、1 クロック サイクルで 2 ビット以上の出力を生成します。動作周波数 (REFCLK) と入力データ レートの関係を使用する式 1 に基づいて、1 クロック サイクルで処理される最大ビット数 (N_{MAX}) を算出できます。

$$N_{MAX} = \text{truncate} \left[\frac{f_{DIN}}{f_{REFCLK}} \right] + 1 \quad \text{式 1}$$

ここでは、 f_{REFCLK} とオーバーサンプリング レートの 3 つのユーザー コンフィギュレーション例を示し、それぞれの結果 (N_{MAX}) を示します。

- $f_{REFCLK} = 125\text{MHz}$ の Fast Ethernet および 2.5Gb/s のオーバーサンプリング レートの場合 : $N_{MAX} = 2$
- $f_{REFCLK} = 125\text{MHz}$ の STM1 および 2.5Gb/s のオーバーサンプリング レートの場合 : $N_{MAX} = 2$
- $f_{REFCLK} = 155.52\text{MHz}$ の Fast Ethernet および 3.1Gbit/s のオーバーサンプリング レートの場合 : $N_{MAX} = 1$

N が 1 より大きい場合は、バレルシフターを使用することによって、固定幅の FIFO へ DRU を簡単にインターフェイスできます。N が 1 の場合、ユーザー アプリケーションは 1 ビット幅のみとなるため、バレルシフターは必要ありません。

表 1 では、NIDRU のコンフィギュレーション属性について説明します。NIDRU のポートについては、表 2 で説明しています。

表 1 : NIDRU のコンフィギュレーション属性

属性名	種類/範囲	説明	コメント
S_MAX	整数 1 ~ 10	1 クロック サイクルで抽出される最大サンプル数の予想値	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。 S_MAX を 10 を設定すると、あらゆるケースに対応できます。
MASK_CG	STD LOGIC VECTOR 15 ~ 0	生成された係数の数値的な精度	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。 MASK_CG をすべて 1 に設定すると、NIDRU が最高精度を使用します。
MASK_PD	STD LOGIC VECTOR 15 ~ 0	PD 計算の数値的な精度	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。 MASK_PD をすべて 1 に設定すると、NIDRU が最高精度を使用します。
MASK_VCO	STD LOGIC VECTOR 15 ~ 0	NCO 出力の数値的な精度	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。 MASK_VCO をすべて 1 に設定すると、NIDRU が最高精度を使用します。
WDT_OUT	整数 2 ~ 40	出力データ幅	バス DT_OUT の出力幅を示します。

表 2 では、NIDRU の各ポートについて説明します。NIDRU のコンフィギュレーション属性については、表 1 で説明しています。

表 2 : NIDRU のポート

ピン名	タイプ	説明	コメント
DT_IN(19:0)	I	SerDes または SelectIO インターフェイスからの入力データ	ビット 0 が最後に配置されます。
CENTER_F(36:0)	I	NIDRU の動作中心周波数	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。
EN	I	イネーブル	NIDRU のすべてのプロセスを有効にします。
G1(4:0)	I	ダイレクト ゲイン	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。
G2(4:0)	I	インテグラル ポスト ゲイン	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。
G1_P(4:0)	I	インテグラルプリ ゲイン	このポートのコンフィギュレーションについては、5 ページの「NIDRU の使用モデル」を参照してください。
CLK	I	クロック	NIDRU プロセスすべてのクロックです。
PH_OUT	O	NCO 位相出力	デバッグ出力です。
INTEG(31:0)	O	インテグラル ブランチ出力	
DIRECT(31:0)	O	ダイレクト ブランチ出力	
CTRL(31:0)	O	NCO 制御信号	
AL_PPM	O	PPM 警告	1 にセットされた場合、入力周波数が NIDRU で設定された範囲を超えたことを示します。
RST	I	リセット	インテグラル パス以外の NIDRU をリセットします。
RST_FREQ	I	インテグラル パスのリセット	NIDRU のインテグラル パスをリセットします。
EN_INTEG	I	インテグラル パスのイネーブル	デバッグ入力であり、1 に設定します。
PH_EST_DIS	I	位相エラー推定メソッド	デバッグ入力であり、0 に設定します。
RECCLK(19:0)	O	リカバリ クロック	リカバリ クロックであり、TX SerDes または SelectIO インターフェイスでシリアライズされます。その順序として、ビット 0 が最初にシリアライズされます。
VER(5:0)	O	バージョン	NIDRU のバージョンを示します。
SAMV(3:0)	O	出力されたサンプル数	各クロック サイクルごとに、NIDRU は抽出されたビット数をレポートします。SAMV は、ラッパ内のバレルシフターに接続されます。
SAM(9:0)	O	サンプル出力	各クロック サイクルごとに、NIDRU は抽出された SAMV ビットをレポートします。これは、SAM の下位部分に配置されます。SAM は、ラッパ内のバレルシフターに接続されます。
DOUT	O	出力データ	ユーザー アプリケーションの出力データを示します。DOUT の幅は、WDT_OUT 属性で指定可能です。
EN_OUT	O	出力データの Valid 信号	DOUT のデータが有効な場合、NIDRU は EN_OUT を 1 に設定します。

NIDRU の使用モデル

このセクションでは、ユーザー アプリケーション要件に基づいた NIDRU のポートおよび属性の指定方法について使用モデルを用いて説明します。アルゴリズムによってハードウェア パラメーター G1、G2、G1_P、および CENTER_F がどのように決定されるかを示します。

図 2 に NIDRU の一般的なクロッキング構造を示します。2つのクロック ドメイン(リモートとローカル)を明示しています。周波数は、各クロック ドメインに括弧で示しています。



重要: NIDRU クロックは、常に PHY クロックにロックされるため、同じクロック ドメイン内にあります。ほとんどの場合、この2つのクロック比率は1となります。

図 2 の概要図では、分周/通倍機能は主に SerDes ブロック内部で実行されますが、その他に PLL で実行または NIDRU の EN 信号を使用する一般的な分周回路でも可能です。

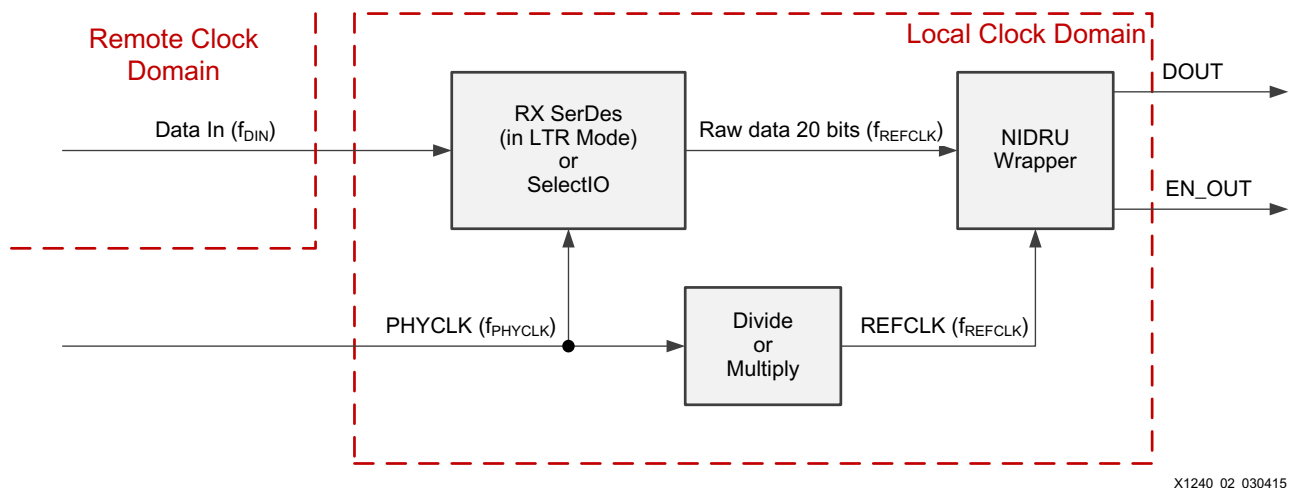


図 2 : NIDRU のクロッキング構造

EN_OUT は、ローカル クロックに同期します。ただし、NIDRU で EN_OUT が 1 にセットされている場合のレートは NIDRU で実行される位相伝達関数でフィルタリングされたリモート クロック ドメインにロックされます。

NIDRU の主な特長は、リモート クロック ドメインとローカル クロック ドメインの比率を小数値にできることであり、これは CENTER_F を使用して指定されます。

NIDRU コンフィギュレーション

このセクションでは、入力されるデータレートと基準クロックの周波数を有効な NIDRU コンフィギュレーションに変換する方法を説明します。

ユーザー コンフィギュレーションでは、次の仕様を定義します。

- 入力されるデータレートとそれに伴う偏差 ($f_{DIN} \pm \text{ppm}$)
- 有効な基準クロック周波数とそれに伴う偏差 ($f_{REFCLK} \pm \text{ppm}$)

f_{DIN} を指定すると、 f_{REFCLK} を有効範囲内で選択できます。上限値は、ターゲット デバイスのタイミング要件を満たすよう必然的に決定されるため、デバイス グレードやスピード グレードによって異なります。 f_{REFCLK} の下限値は、PHY によって制限される場合があります。たとえば、一般的に SerDes は基準クロックの最小周波数を規定します。SelectIO インターフェイスでは、一般的に f_{REFCLK} の下限が制限されることはありません。

最大 f_{DIN} は、式 2 で算出されるオーバーサンプリング レート (O_R) によって制限されます。

$$O_R = \frac{20 \cdot f_{REFCLK}}{f_{DIN}} \quad \text{式 2}$$

O_R は、少なくとも 2 より大きい値になる必要がありますが、ザイリンクスでは高周波数での十分なジッター耐性を備えるために $O_R \geq 3$ を推奨しています。

理論上の高周波ジッター耐性は、式 3 で示すように O_R と関連性を持ちます。

$$J_{(TOL-HF)} = \frac{1}{O_R} \quad \text{式 3}$$

式 3 が成立するすべての DRU には最適な $J_{(TOL-HF)}$ があります。

式 4 および式 5 を使用して、NIDRU のパラメーター (CENTER_F、 G_1 、 G_2) を計算します。式 4 では、CENTER_F を計算します。

$$CENTER_F = \frac{f_{DIN}}{f_{REFCLK}} \times 2^{32} \quad \text{式 4}$$

式 5 では、 G_1 および G_2 を計算します。

$$G_1 = G_2 \leq 32 - \text{roundup} \left[\log_2 \frac{2^{33} \cdot (\text{ppm}_{DIN} + \text{ppm}_{REFCLK}) \cdot f_{DIN} \cdot 10^{-6}}{f_{REFCLK}} \right] \quad \text{式 5}$$

G_1 と G_2 に同じ値を使用した場合、入力されるデータレートと基準クロックの両方の完全耐性範囲内で NIDRU が動作することが保証されます。 G_2 を減少させると、NIDRU の帯域幅が増加します。 G_2 を増加させると、常時自動ロックが保証されず、NIDRU がブルイン領域で動作することになるため推奨されません。

通常、 G_{1P} はシミュレーションで評価する必要があります。入力位相のステップを変更して、出力位相でリングングが無視できる条件になるまで、 G_{1P} 値を増加させます。 $G_1 = G_2$ の場合、 $G_{1P} = 16$ に設定すると無視できる範囲のリングングが保証されます。したがって、通常は $G_{1P} = 16$ に設定することが理想的です。

例

ここでは、 f_{DIN} および f_{REFCLK} の 3 つのユーザー コンフィギュレーション例を示し、それぞれ CENTER_F、 G_1 、 G_2 の結果を示します。

- Fast Ethernet ($f_{DIN} = 125\text{Mb/s} \pm 100\text{ppm}$ 、 $f_{REFCLK} = 125\text{MHz} \pm 100\text{ppm}$) :
CENTER_F = b00001000000000000000000000000000、 $G_1 = G_2 \leq 11$
- Fast Ethernet ($f_{DIN} = 125\text{Mb/s} \pm 10 \text{ ppm}$ 、 $f_{REFCLK} = 155.52\text{MHz} \pm 20\text{ppm}$) :
CENTER_F = b0000011001101110000101110010110101001、 $G_1 = G_2 \leq 11$
- OC3 ($f_{DIN} = 155.52\text{Mb/s} \pm 20\text{ppm}$ 、 $f_{REFCLK} = 125\text{MHz} \pm 100 \text{ ppm}$) :
CENTER_F = b0000100111110100000010100010100001110、 $G_1 = G_2 \leq 11$

PHY コンフィギュレーション

このセクションでは、PHY を適切に設定するための推奨条件を示します。

小数値のオーバーサンプラーが、PHY (通常は SelectIO インターフェイスまたは SerDes) からオーバーサンプリングしたデータを処理します。SerDes の場合、リファレンス モードにロックした状態にする必要があるため、次のようにポートを設定して、自動調整イコライザーを無効にしてください。

- RXLPMEN = 1
- RXLPMHFOVRDEN = 1
- RXLPMLFKLOVRDEN = 1
- RXOSOVRDEN=1

DRU のシミュレーション

TB_SIM_DRU_JITTER テストベンチは、同期および準同期シリアル入力を使用し、一般的なデータ レートで動作する NIDRU の性能をシミュレーションするためのものです。シミュレーションでは、表 3 に示す 6 つのケースを順に実行します。その他のケースとして、ユーザー アプリケーションのデータ レートを追加できます。

ケース 2 とケース 4 は、NIDRU が小数と整数の両方のオーバーサンプリング レートで動作できることを示しています。ケース 6 は、1Kb/s という非常に低いデータ レート (125K オーバーサンプリング レートと同等) で NIDRU が動作できることを示しています。

表 3: シミュレーション ケース

ケース番号	プロトコル	データ レート	基準クロック	オーバーサンプリング レート
1	プロプライエタリ	250Mb/s	125MHz	10
2	OC3	155.52Mb/s (+100ppm)	125MHz	16.075
3	SDI	270Mb/s (+100ppm)	148.5MHz	11
4	OC3	155.52Mb/s	155.52MHz	20
5	OC12	622.08Mb/s	125MHz	4.019
6	プロプライエタリ	1Kb/s	125MHz	2.5e6

シミュレーション スクリプトの実行手順は次のとおりです。

1. コマンド ウィンドウを開きます。
2. scripts ディレクトリに移動します。
3. modelsim.ini コンフィギュレーション ファイルを scripts ディレクトリにコピーします。
4. ModelSim を開きます。
5. ModelSim で run_sim_do スクリプトを実行します。



注記: このテストベンチは Modelsim のみでテストされていますが、NIDRU コアは次のシミュレーション ツールでも問題なく動作します。

- Vivado@ シミュレータ
- Mentor Graphics Modelsim
- Synopsys VCS

図 3 に、TB_SIM_DRU_JITTER テストベンチに実装されているアーキテクチャを示します。シミュレーション時間を最も短くするため、フル機能の SerDes の代わりに最適化された 20 ビットのデシリアライザーとシリアライザーが使用されています。また、同じ目的で SerDes トランスミッターの代わりに最適化された 20 ビットのシリアライザーが使用されています。

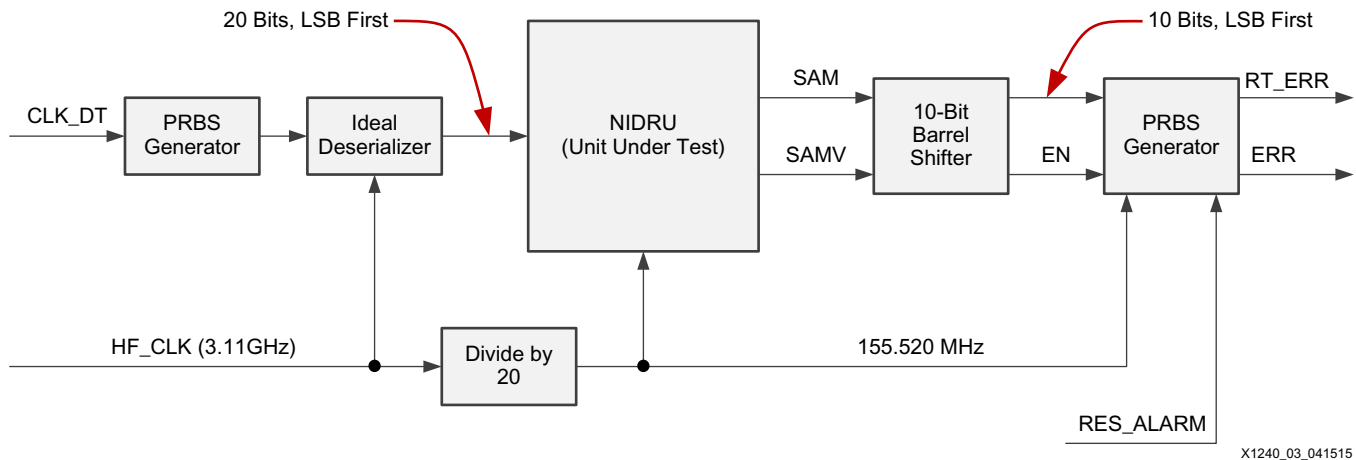


図 3 : TB_SIM_NIDRU のブロック図

テストベンチには、次の 2 つのクロック ドメインがあります。

- 配線のクロック ドメイン (HF_CLK に同期)
- DRU のクロック ドメイン (REFCLK に同期)

DRU (Fast Ethernet の場合) は、Fast Ethernet の物理的仕様 [参照 1] に定義されている 200ppm という要件を超えますが、2 つのクロック ドメイン間を 250ppm 以内の周波数誤差で動作します。

PRBS (疑似ランダム バイナリ シーケンス) Generator は、CLK_DT の最大スピードで動作して、PRBS 15 パターンを生成できます。Ideal Deserializer (最適化されたデシリアライザー)、DRU、および PRBS チェッカーはすべて、ローカルの REFCLK ドメインで動作します。DRU と PRBS チェッカーは合成可能です。

テストベンチには、GTX トランシーバー インスタンスの代わりに、3.11GHz で動作する Ideal Deserializer が配置されています。これによって、結果の信頼性を保ちながらシミュレーション時間を短縮できます。Ideal Deserializer の代わりに、GTP または GTX トランシーバーの SmartModel を使用してフルシミュレーションを実行することも可能です。

PRBS Generator がシリアルの場合でも PRBS チェッカーはパラレル動作します。EN = 1 の場合のみ、DRU の 10 ビット出力が処理されます。

シミュレーション デバッグでは、RT_ERR (real-time error) 出力を使用できます。この信号は、エラーが検出されると High にセットされ、10 ビットのベクターが修正されると Low になります。ERR 出力はエラーが検出されると High に遷移し、ERR_RST でリセットされます。したがって、ERR は通常のハードウェア テストで使用されますが、RT_ERR はシミュレーションで有効です。

出力例を図 4 に示します。125Mb/s \pm 25ppm 入力、155.52MHz (24.88X) の基準クロックを使用して回復されます。制御信号が Low に遷移し、入力されるデータレートに合わせて VCO 周波数を調整します。/scripts/wave.do ファイルを使用し、波形を描画したり CTRL 信号を拡大することで DRU の線形反応を確認できます。図 4 の描画は一次応答のように見えますが、実際には非常に小さいオーバーシュートのある二次応答です。

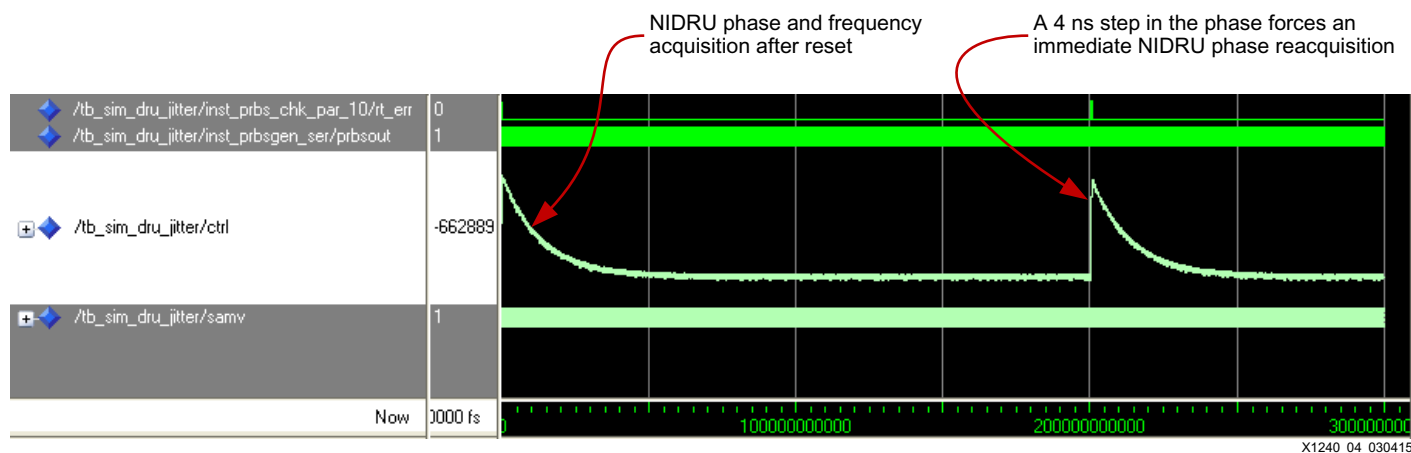


図 4 : シミュレーション出力の例

リソース

NIDRU は、加算器、乗算器、累算器、シフターなど効率に優れたストラクチャのみを使用して設計されています。アプリケーションによりませんが、NIDRU ブロックは 1 つの DSP を推論して合成できます。表 4 に、Kintex®-7 FPGA のリソース要件を示します。

表 4 : Kintex-7 FPGA で NIDRU に必要なハードウェア リソース

合成タイプ	フリップフロップ	LUT	DSP48E スライス数	BUFG ⁽¹⁾	Kintex-7 FPGA スライス ⁽²⁾⁽³⁾
DSP あり	755	1,350	1	1	414
DSP なし	795	1,681	0	1	518

注記 :

- 複数のチャンネルがセットアップされ、すべてが異なるデータレートで動作している場合でも、必要な BUFG は 1 つのみです。
- ここに示すリソース要件にパレル シフターは含まれていません。
- これらの結果は VIVADO Design Suite v2014.2 を使用して取得したものです。合成およびインプリメンテーションのストラテジはデフォルトを使用し、CLK 周期は 6.4ns に設定しています。

ソフトウェア要件

このデザインのソフトウェア要件は次のとおりです。

- Vivado Design Suite の v2014.2 またはそれ以降
- Mentor Graphics ModelSim ソフトウェアの v10.0c またはそれ以降 (シミュレーション用)

リファレンス デザイン

このアプリケーション ノートの [リファレンス デザイン ファイル](#) は、ザイリンクスのウェブサイトからダウンロードできます。

表 5 に、リファレンス デザインの詳細を示します。

表 5: リファレンス デザインの詳細

パラメーター	説明
全般	
開発者	Paolo Novellini、Antonello Di Fresco、Giovanni Guasti
ターゲット デバイス	ザイリンクス 7 シリーズ デバイスおよび UltraScale デバイス
ソース コードの提供	あり (一部暗号化)
ソース コードの形式	VHDL
既存のザイリンクス アプリケーション ノート/リファレンス デザイン、またはサードパーティ ソースからデザインへのコード/IP の使用	このリファレンス デザインでは XAPP884 のコードを使用
シミュレーション	
論理シミュレーションの実施	あり
タイミングシミュレーションの実施	N/A
論理シミュレーションおよびタイミング シミュレーションでのテストベンチの利用	あり
テストベンチの形式	VHDL
使用したシミュレータ/バージョン	ModelSim 10.0 c
SPICE/IBIS シミュレーションの実施	N/A
インプリメンテーション	
使用した合成ツール/バージョン	Vivado 合成
使用したインプリメンテーション ツール/バージョン	Vivado インプリメンテーション
スタティック タイミング解析の実施	あり
ハードウェア検証	
ハードウェア検証の実施	あり
使用したハードウェア	KC724 GTX トランシーバー特性評価ボード

参考資料

1. 『Phase-Locked Loops:Design, Simulation, and Applications』4 版 (1999 年) – 著者 : Best, Roland E、出版社 : McGraw-Hill Professional
2. 『Phaselock Techniques』3 版 (2005 年) – 著者 : Gardner, Floyd M、出版社 : Wiley-Interscience
3. IEEE 802.3 Ethernet 規格 : www.ieee802.org/3/

改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	内容
2015 年 4 月 17 日	1.0	初版

法的通知

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available “AS IS” and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx’s limited warranty, please refer to Xilinx’s Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx’s Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>.

© Copyright 2015 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

この資料に関するフィードバックおよびリンクなどの問題につきましては、jpn_trans_feedback@xilinx.com まで、または各ページの右下にある [フィードバック送信] ボタンをクリックすると表示されるフォームからお知らせください。フィードバックは日本語で入力可能です。いただきましたご意見を参考に早急に対応させていただきます。なお、このメールアドレスへのお問い合わせは受け付けておりません。あらかじめご了承ください。