

# ChipScope Pro チュートリアル

## ChipScope Pro Analyzer での IBERT コアの使用

UG811 (v 13.1) 2011 年 4 月 1 日



Xilinx is disclosing this user guide, manual, release note, and/or specification (the “Documentation”) to you solely for use in the development of designs to operate with Xilinx hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Xilinx expressly disclaims any liability arising out of your use of the Documentation. Xilinx reserves the right, at its sole discretion, to change the Documentation without notice at any time. Xilinx assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. Xilinx expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information. THE DOCUMENTATION IS DISCLOSED TO YOU “AS-IS” WITH NO WARRANTY OF ANY KIND. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

© Copyright 2011 Xilinx, Inc. XILINX, the Xilinx logo, Virtex, Spartan, ISE, 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.

本資料は英語版 (v 13.1) を翻訳したもので、内容に相違が生じる場合には原文を優先します。

資料によっては英語版の更新に対応していないものがあります。

日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。

## 改定履歴

The following table shows the revision history for this document.

日付	改訂内容
2011 年 4 月 1 日	13.1 で初期リリース

# 目次

---

改定履歴.....	2
<b>ChipScope チュートリアル : ChipScope Pro Analyzer での IBERT コアの使用</b>	
概要 .....	5
要件 .....	5
設定 .....	5
デザインの説明 .....	6
手順 1 : IBERT デザインのプロジェクト作成、カスタマイズおよび生成 .....	8
手順 2 : ChipScope Pro Analyzer でのデザインの確認 .....	10
手順 3 : スイープ テストの実行 .....	13
<b>その他のリソース</b>	
ザイリンクス リソース .....	19
ChipScope 資料.....	19
ボード資料.....	19



# ChipScope チュートリアル : ChipScope Pro Analyzer での IBERT コアの使用

---

## 概要

このチュートリアルでは、次を実行します。

- スタンドアロンの **CORE Generator™** ツールを使用して **Integrated Bit Error Ratio Tester (IBERT)** コアをカスタマイズおよび生成します。
- **ChipScope™ Pro Analyzer** を使用してみます。これには、**ChipScope Pro Analyzer** へのビットストリームファイルのインポート、デバイスのコンフィギュレーション、**IBERT/トランシーバー IP** コアの確認が含まれます。
- スweep テストを実行してトランシーバー チャネルを最適化し、ソフトウェア バージョン 13.1 に含まれる **IBERT** スweep プロットの GUI 機能を使用してデータのプロットを描画します。

## 要件

Xilinx® ISE® ツール フローの基本的な知識

## 設定

### 必要なパーツ

次のソフトウェアおよびハードウェアが必要です。

- ザイリンクス ISE Design Suite 13.1 (Logic、DSP、Embedded、または System Edition)
- ML605 ボード
- ML605 ボードと一緒に提供される JTAG USB ケーブル
- SMA (SubMiniature バージョン A) ケーブル 2 本

### ボードとケーブルの接続

1. ボードの USB JTAG コネクタから USB ケーブルをコンピューター システムに接続します。
2. 2 本の SMA ケーブルを次のように接続します。
  - a. 1 本の SMA ケーブルを J28 から J26 に接続
  - b. もう 1 本の SMA ケーブルを J29 から J27 に接続

ボードのコネクタの位置は、図 1 に示すとおりです。

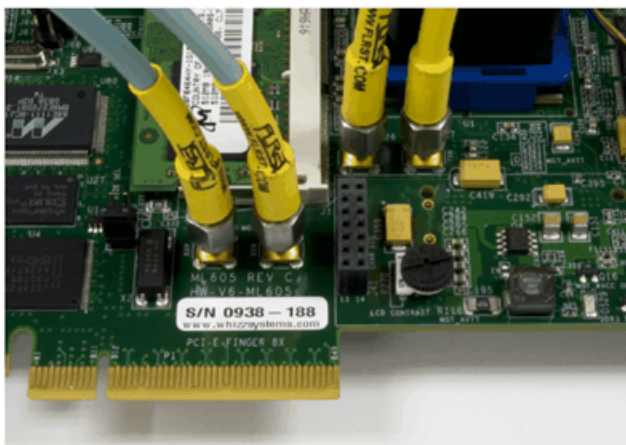


図 1 : SMA ケーブルの接続

3. ML605 の電源をオンにします。

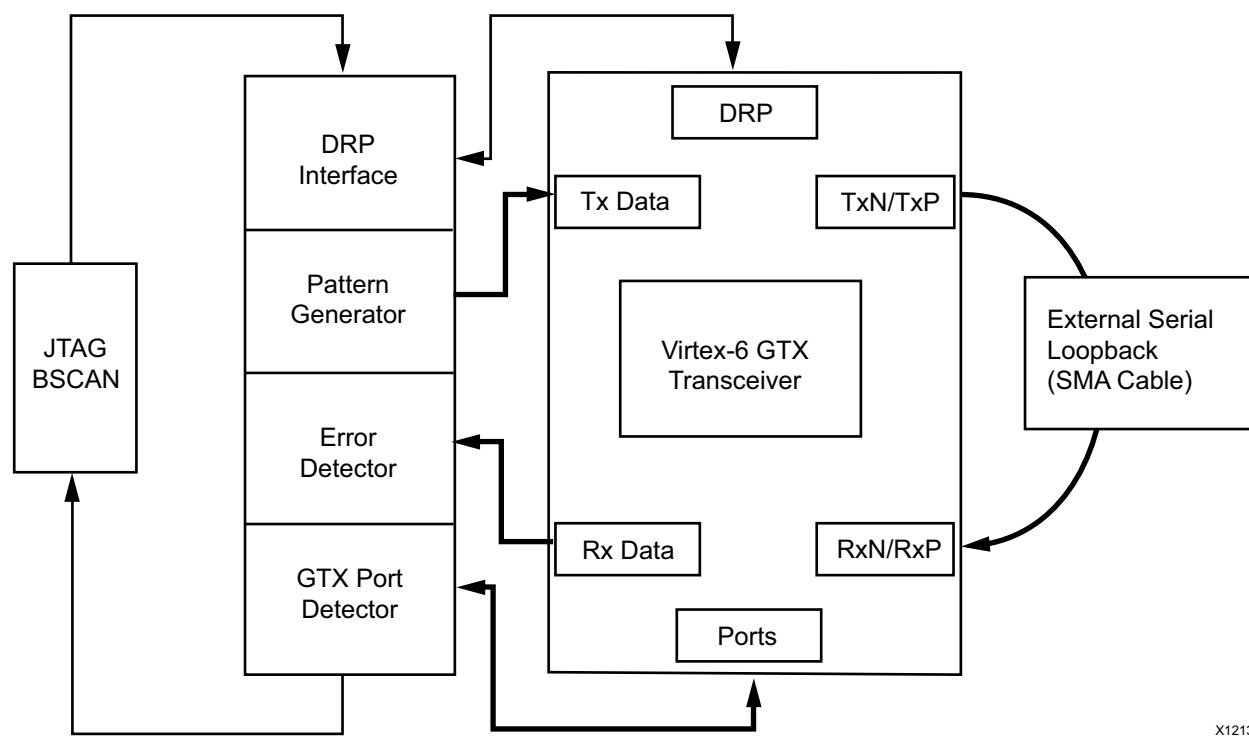
## デザインの説明

ChipScope Pro Analyzer IBERT コアをカスタマイズしておく、さまざまなザイリンクスリンクスデバイスのトランシーバー機能を評価および監視するために使用できます。このチュートリアルでは、Virtex®-6 GTX トランシーバーに焦点を置いて説明します。このデザインには、FPGA ロジックにインプリメントされたパターン ジェネレーターおよびパターン チェッカーのほか、GTX トランシーバーのポートと DRP (ダイナミック リコンフィギュレーション ポート) 属性へのアクセスが含まれます。また、通信ロジックが含まれるので、JTAG を使用してランタイム時にデザインにアクセスできます。IBERT コアは、すべてを含むデザインです。生成すると、ビットストリーム生成を含めたインプリメンテーション フロー全体が実行されます。

IBERT デザインは、ザイリンクスの CORE Generator ツールでのカスタマイズ設定に従って自動的に生成されるので、このチュートリアル専用のデザイン例が追加で必要になることはありません。

図 2 は、Virtex-6 トランシーバーを含む IBERT Virtex-6 GTX コアのインターフェイス間の関係を示すブロック図です。

- **DRP インターフェイスと GTX ポート レジスター** : IBERT を使用すると、GTX トランシーバーのポートおよび属性をユーザーが柔軟に変更できるようにもなります。ダイナミック リコンフィギュレーション ポート (DRP) が含まれるので、IBERT コアに含まれる GTX トランシーバーの属性をランタイム ソフトウェアにより監視および変更できます。必要な場合は、読み出しおよび書き込み可能なレジスターも含まれます。これらは、GTX トランシーバーのポートに接続されます。これらすべては、ChipScope Analyzer ツールを使用してランタイム時にアクセスできます。
- **パターン ジェネレーター** : IBERT デザインでイネーブルになった各 GTX トランシーバーには、パターン ジェネレーターとパターン チェッカーの両方が含まれます。パターン ジェネレーターはトランスミッタを介してデータを送信します。
- **エラー検出器** : IBERT デザインでイネーブルになった各 GTX トランシーバーには、パターン ジェネレーターとパターン チェッカーの両方が含まれます。パターン チェッカーはレシーバーからのデータを受信し、それを内部で生成されたパターンと比較します。



X12133

図 2 : IBERT デザイン フロー

## 手順 1 : IBERT デザインのプロジェクト作成、カスタマイズおよび生成

1. ザイリンクスの CORE Generator ツールを起動します。
2. [File] → [New Project] をクリックし、プロジェクト名を IBERT\_GTX\_coregen にして保存します (図 3)。

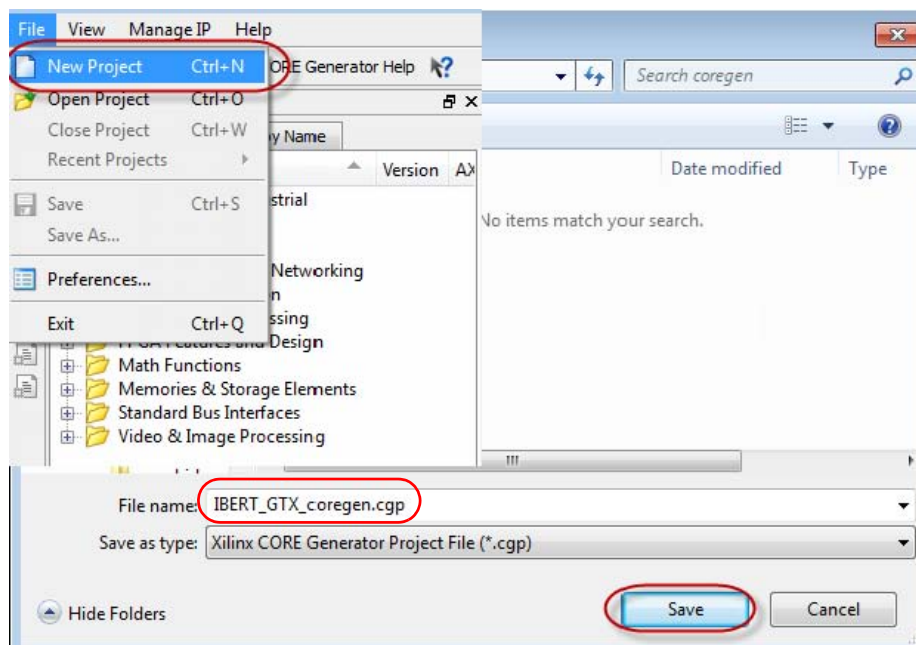


図 3 : CORE Generator ツールでのプロジェクトの作成と保存

3. プロジェクトを保存すると、[Project Options] ダイアログ ボックスが表示されます。
  - a. [Part] オプションを選択して、次を設定します。
    - [Family] を [Virtex 6] に設定
    - [Device] を [-xc6vlx240t] (ML605 ボードのデバイス) に設定
    - [Package] を [ff1156] に設定
    - [Speed Grade] を [-1] に設定
  - b. その他すべてのプロジェクト オプションはデフォルトのままにします。
  - c. [Apply] をクリックしてから [OK] をクリックします。
4. 生成する IBERT IP コアを選択します。  
[IP Catalog] ペインで [Debug & Verification] → [ChipScope Pro] → [IBERT Virtex6 GTX (ChipScope Pro-IBERT)] をダブルクリックします。
5. コアのボード コンフィギュレーションを設定するダイアログ ボックス (図 4) の [Board Configuration Settings] ドロップダウン メニューから [ml605 bank116fmc1pcsfpsmasgmii] を選択します。

**メモ :** 図 4 に示すようなドロップダウン リストが表示されない場合は、前述の 2 番で別のデバイスを選択してしまっている可能性があります。

**メモ :** [Board Configuration Settings] ドロップダウン メニューには、[User Defined] と、上記で選択したものも含めて 4 つの ML605 ボードをターゲットとするコンフィギュレーション済みボード設定が含まれます。各コンフィギュレーション済みボード設定には、必要な設定がす



べて含まれています。これは、デザイン環境を素早く評価したり、一部のコンフィギュレーション済みのボード ファンクションを確認する場合などに便利です。また、これらの設定は [User Defined] 設定のテンプレートとしても使用できます。[User Defined] オプションを使用する場合は、クロック、ピン位置、プロトコル タイプなどのパラメータをユーザーがあらかじめ知っている必要があります。コンフィギュレーション済みボード設定を選択する場合は、その必要はありません。

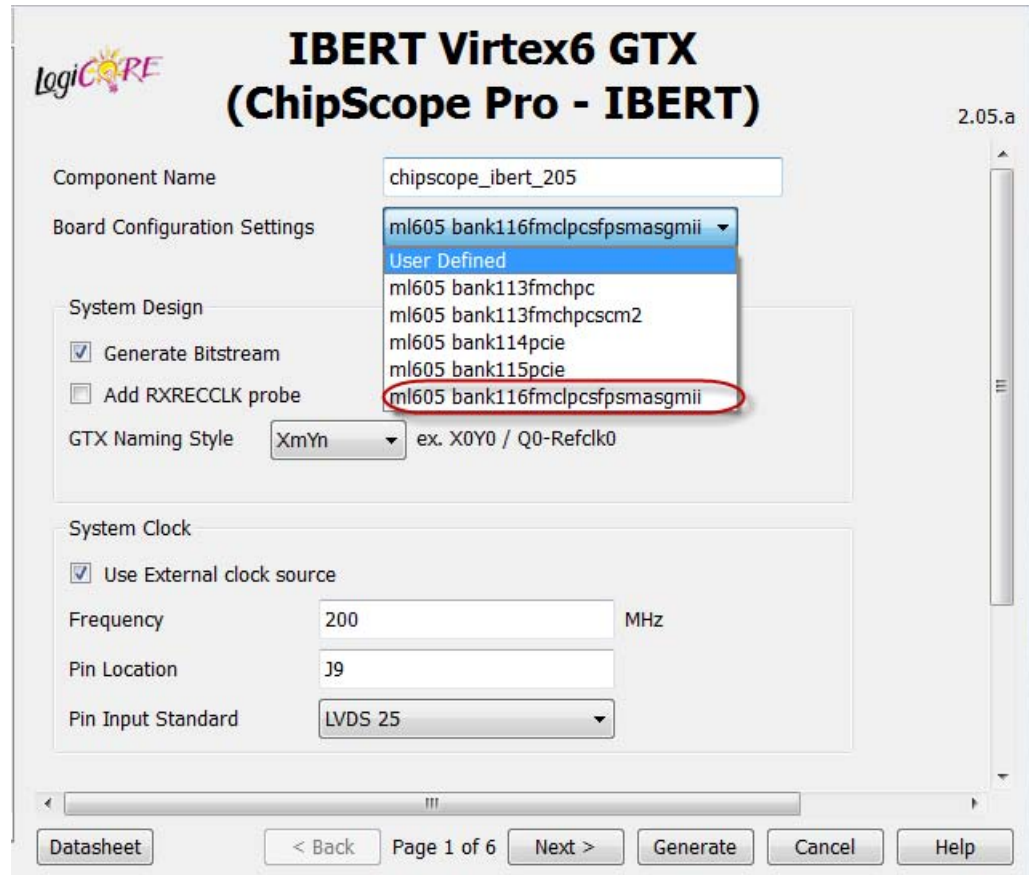


図 4 : [Board Configuration Settings] ダイアログ ボックス

6. [Next] をクリックし、コンフィギュレーション済みパラメーターを確認します。
7. 確認したら [Generate] をクリックして、IBERT デザインの生成を開始します。コアの生成には、約 1 分ほどかかることがあります。

ここまでで、CORE Generator ツールでプロジェクト ファイルを作成し、IBERT デザインをカスタマイズしてから生成しました。次は、ChipScope Pro Analyzer を使用してこのデザインを確認する方法について学びます。

## 手順 2 : ChipScope Pro Analyzer での設計の確認

ここでは、ChipScope Pro Analyzer を使用して手順 1 で作成した IBERT デザインを確認します。具体的には、ビット エラー数を確認しつつ、さまざまな入力パターンおよびループバック モードを使用して解析を実行します。

1. ChipScope Pro Analyzer を起動します。
2. メニューから [JTAG Chain] → [Xilinx Platform USB Cable] をクリックします。
3. [ChipScope Pro Analyzer [new project]] ダイアログ ボックスが表示されます。デフォルトの設定を使用してください。
4. [ChipScope Pro Analyzer] ダイアログ ボックスが表示されます。デフォルトの設定を使用してください。

### デバイスのコンフィギュレーション

1. ChipScope Pro Analyzer のメイン ウィンドウの [New Project] ペインで [DEV: 1 MyDevice1 (XC6VLX240T)] を右クリックし、[Configure] をクリックします。
2. ダイアログ ボックスで [Select New File] をクリックし、...\coregen\\*.bit ファイルを指定して [OK] をクリックして、ML605 にビット ファイルをダウンロードします。
3. ChipScope Pro Analyzer のメイン ウィンドウの [New Project] ペインで [DEV: 1 MyDevice1 (XC6VLX240T)] を展開表示し、[IBERT Console] をダブルクリックします。

**メモ：**この段階で、現在のプロジェクトの設定でコアを設定するかどうかを尋ねる [ChipScope Pro Analyzer - IBERT V6TX Project Settings] プロンプトが表示されることがあります。このプロンプトが表示されたら、[Yes] をクリックします。

4. ループバック モードの初期設定を使用し、[IBERT Console] ウィンドウが表示されます (図 5)。

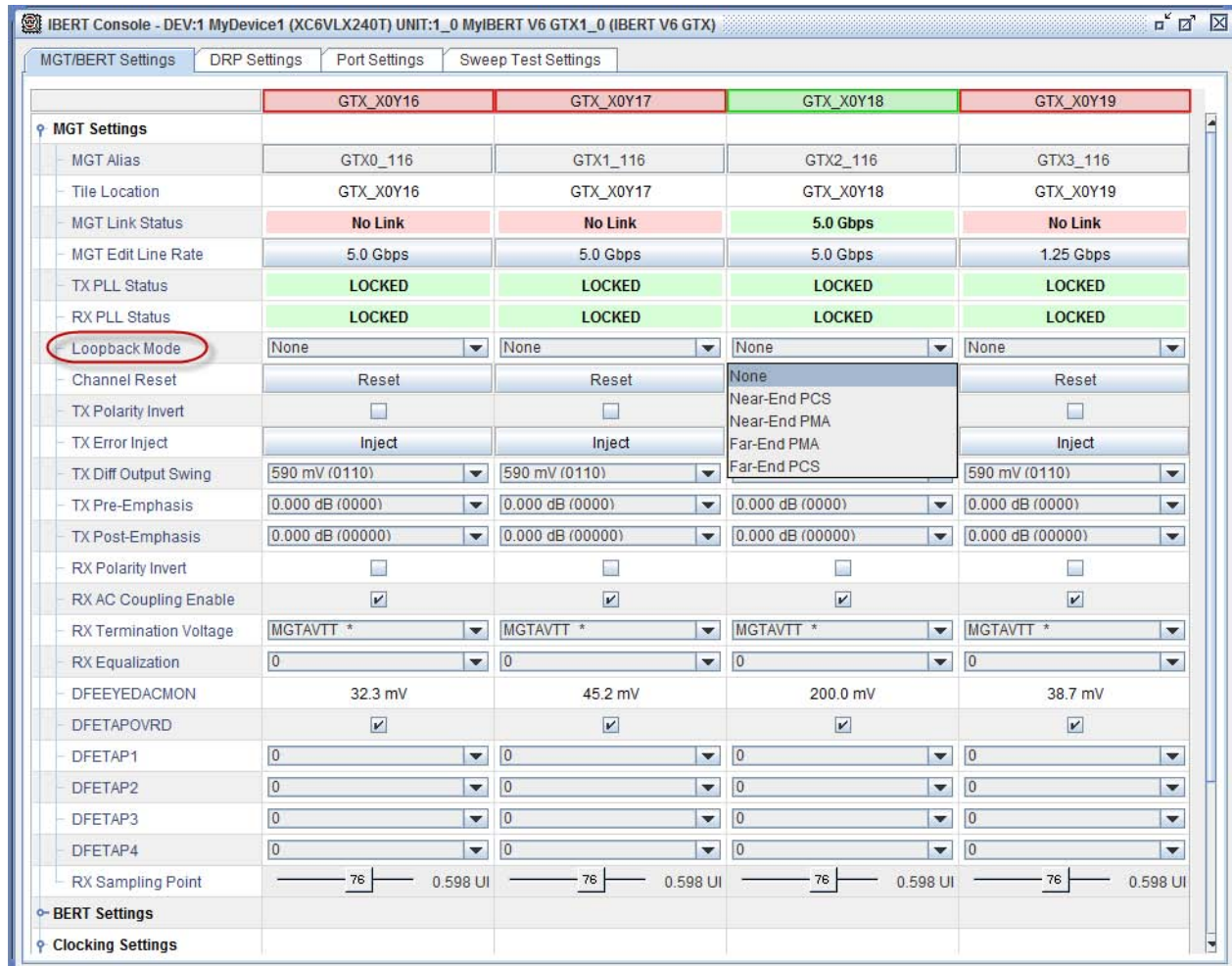


図 5 : Virtex-6 FPGA GTX トランシーバーの [IBERT Console] ウィンドウ

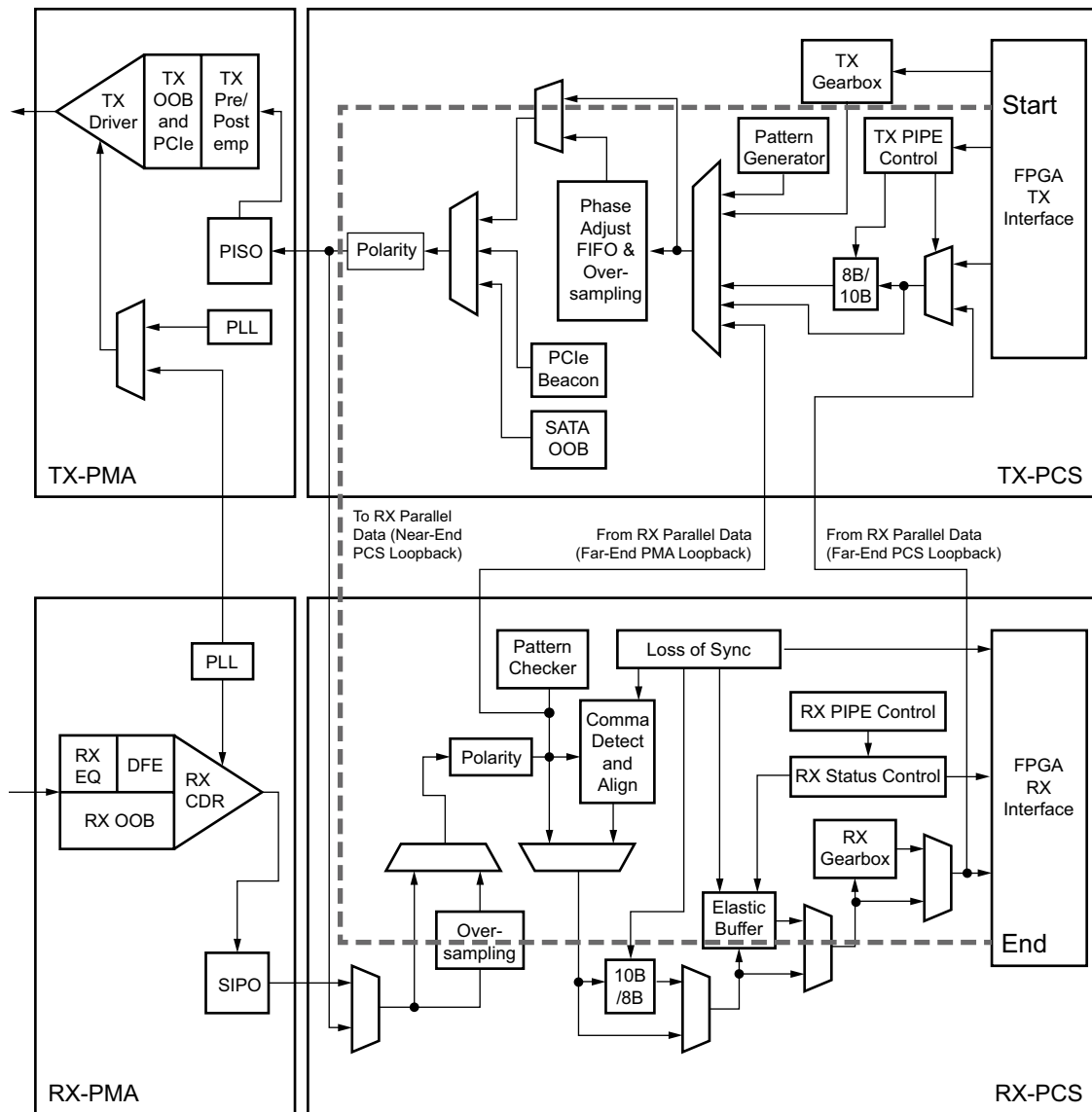
## [IBERT Console] ウィンドウの主な機能

コンソール ウィンドウの一番上には 4 つの設定タブが表示されます。それぞれのタブが表示されており、通常はダイナミックに制御可能なファンクションまたはステータス モニターとして機能するファンクションを記述する行が含まれます。

たとえば、プルダウン メニューから使用可能な設定のいずれか 1 つを選択すると、ループバックモードのファンクションをコンフィギュレーションできます。

特定の GTX トランシーバー チャネルのループバック モードを制御する [Loopback Mode] 設定を確認してください。このチュートリアルでは、次のループバック モードが使用されます。

- **[None]** : 内部ループバックは使用されません。
- **[Near-End PCS]** : 回路は近端 GTX トランシーバーに完全に含まれます。ループバック パスは、TX ファブリック インターフェースから開始して、PCS を通過し、GTX トランシーバーチャネルの PMA 側を通過せずに RX ファブリックに戻ります (図 6)。



x12134

図 6 : GTX のループバック パス

[IBERT Console] ウィンドウを確認すると、GTX\_X0Y18 トランシーバー チャンネルだけが TX および RX チャンネル間で 5Gbps のライン レート リンクを構築していることがわかります。これは、TX および RX 間でループバックされるのがこのチャンネルだけだからです。その他 3 つのチャンネルは、「No Link」と表示されます。GTX\_X0Y18 トランシーバーは、ML605 回路図 (図 7) に示すように、SMA ケーブルを通してループバック可能な ML605 BANK 116 の唯一のチャンネルです。

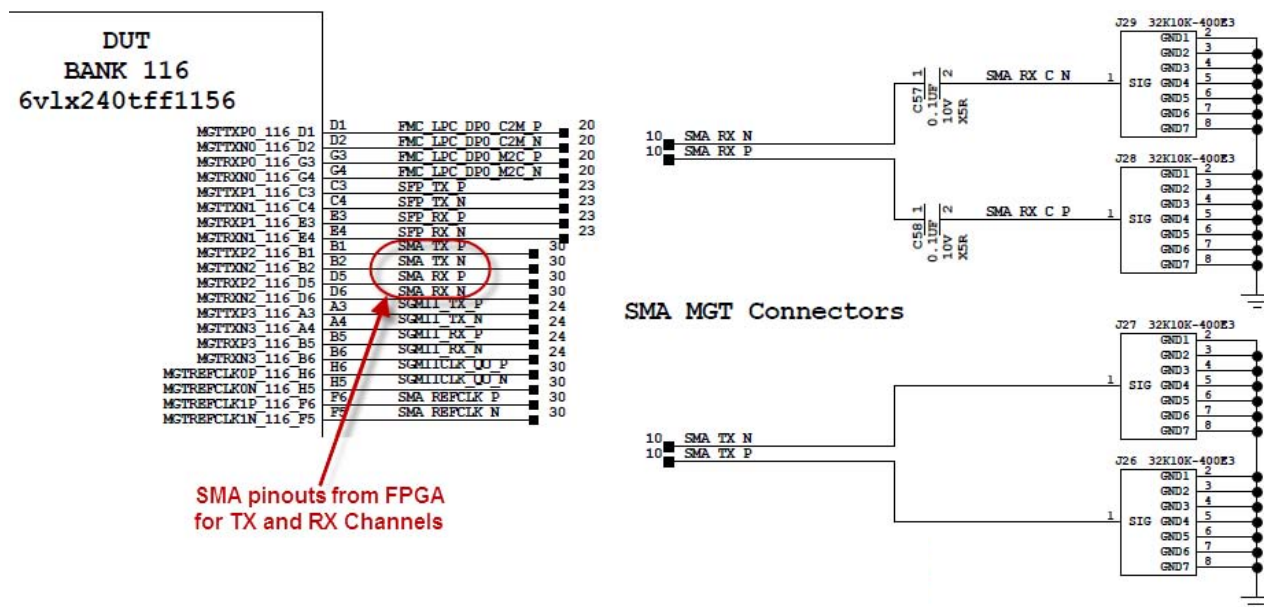


図 7 : SMA ケーブルを通るループバックを示す ML605 回路図

Virtex-6 GTX トランシーバーの [IBERT Console] ウィンドウに関する詳細は、  
<http://japan.xilinx.com/tools/cspro.htm> の『ChipScope Pro ソフトウェアおよびコア ユーザー ガイド』を参照してください。

## 手順 3 : スイープ テストの実行

### 概要

この最後の手順では、さまざまなトランシーバー チャンネル設定を使用してチャンネルのスイープ テストを実行します。

- [Sweep Test Settings] パネルを開いて、スイープ テストのパラメーターを確認します。
- スイープ テストの設定をします。
- スイープ テストを実行します。
- IBERT のスイープ プロットの GUI を使用してデータのプロットを描画します。

### [Sweep Test Settings] パネルを開いて、スイープ テストのパラメーターを確認

[IBERT Console] ウィンドウで [Sweep Test Settings] タブをクリックし、スイープ テストのパラメータを設定可能なパネルを開きます。

### [Sweep Test Settings] パネルのパラメータとオプション

[Sweep Test Settings] パネルが表示されます (図 8)。重要なオプションは、赤い円で囲んであります。

[Sweep Test Settings] パネルでは、さまざまなトランシーバー設定をスイープするチャンネル テストを設定できます。



TX および RX 設定は、同じ GTX トランシーバー用です。TX および RX の両方の設定のスイープは、トランシーバーが近端または外部ループバック モードの 1 つに設定されているときのみ機能します。RX パラメーターのスイープは、リンクの対応する TX エンドポイントが別のデバイスまたは同じデバイスに含まれる別のトランシーバーに含まれるときのみ実行できます。

[Sweep Test Settings] パネルには、次の 4 つのフィールドがあります。

- [Parameter Settings]
- [Sampling Point Region]
- [Test Controls]
- [Test Results]

ここでは、GTX\_X0Y18 トランシーバー チャンネルのスイープ テストを実行します。

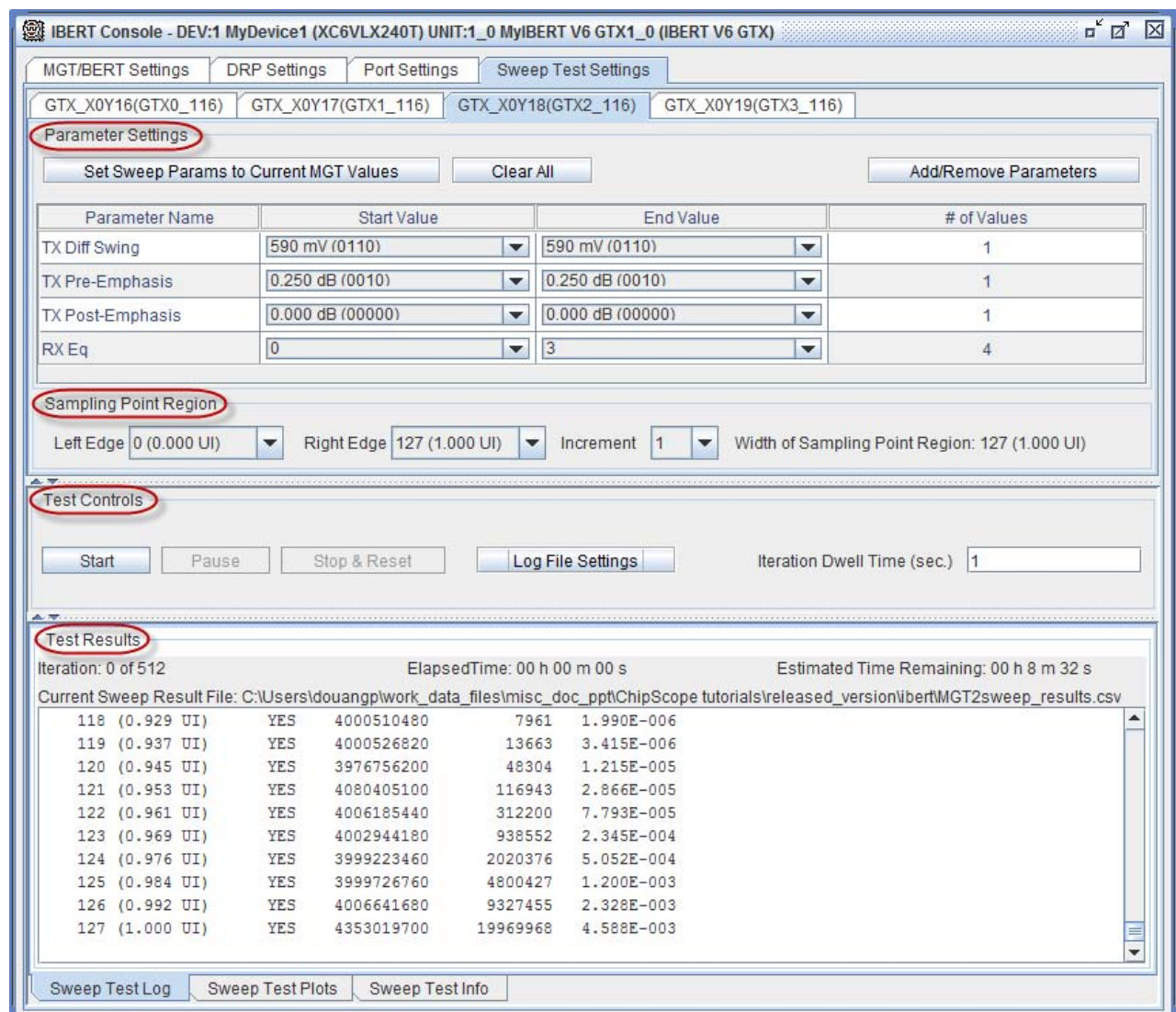


図 8 : [Sweep Test Settings] パネル

## スイープ テストの設定

1. [IBERT Console] ウィンドウで [Sweep Test Settings] タブがオンになっていることを確認します (図 8)。
2. [GTX\_X0Y18 (GTX2\_116)] トランシーバーを選択します。
3. [Add/Remove Parameters] をクリックします。[Add/Remove Ports/Attributes] ダイアログ ボックスが表示されます。
4. [Add/Remove Ports/Attributes] ダイアログ ボックスで次のパラメーターを選択し、次の順番どおりに並べます (図 9)。
  - [TX Diff Swing]
  - [TX Pre-Emphasis]
  - [TX Post-Emphasis]
  - [RX Eq]
5. [OK] をクリックして [IBERT Console] ウィンドウに戻ります。

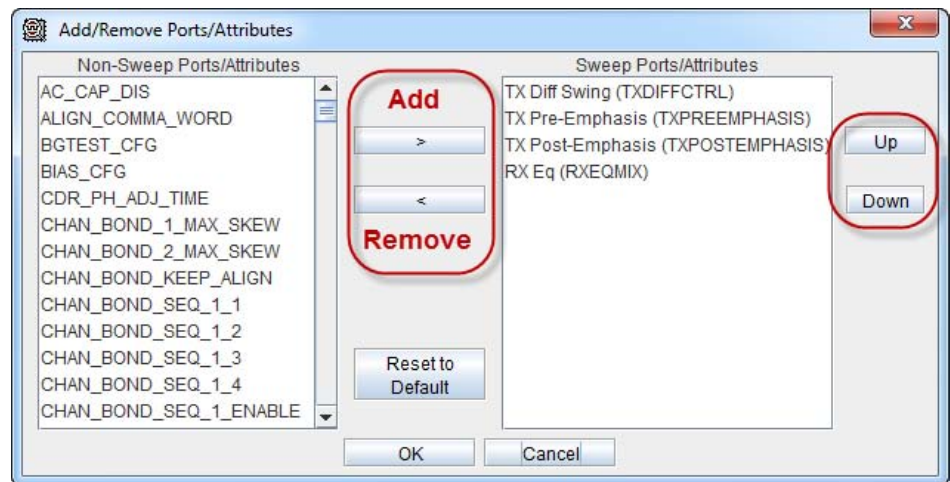


図 9：スイープ設定オプション

**メモ：** [Sweep Ports/Attributes] リストのパラメーターの順序は、パラメーターのスイープ頻度を示しています。リストの一番上の近いパラメータの値は、一番下に近いパラメータの値よりもスイープされる頻度が少なくなります。つまり、表上部のパラメータはスイープ アルゴリズムの外側ループにあり、表下部のパラメータはスイープ アルゴリズムの内側ループにあるということです。

6. [IBERT Console] ウィンドウで [Sweep Test Settings] タブをオンにし、それぞれのスイープ パラメータの開始値 (Start Value) と終了値 (Stop Value) を確認します。

パラメータ	Start Value	Stop Value
TX Diff Swing	590 mV	590 mV
TX Pre-Emphasis	0.25 dB (0.010)	0.25 dB (0.010)
TX Post-Emphasis	0.000 dB (0.000)	0.000 dB (0.000)
RX Eq	0	3

**メモ：** このチュートリアルでは、スイープされるパラメータは RX Eq (Receiver Equalizer) のみです。それ以外は定数値に設定されます。

7. [Sampling Point Region] が 0 (0.000 UI) から 127 (1.00 UI) で、反復ごとの滞留時間は 1 秒増加するように設定します。

メモ : [Sampling Point Region] は、アイとサンプル間の水平ポイントを示します。

8. [Sweep Test Settings] パネルの [Test Controls] フィールドで、[Iteration Dwell Time] が 1 になっていることを確認します。

## スイープ テストの実行と結果の確認

[Start] をクリックしてテスト データのスイープを開始します。テスト結果は、リアルタイムまたはログ ファイルで確認できます。

- テスト結果をリアルタイムで確認する場合 :

[IBERT Console] ウィンドウの一番下の [Sweep Test Log] タブをクリックし、スイープ テスト結果を確認します。これらのスイープ パラメーター設定の場合、合計 512 反復 (サンプル ポイントの合計 \* スイープ値の総数 =  $128 * 1 * 1 * 1 * 4 = 512$ ) になります。

- テスト結果をログ ファイルで確認する場合 :

テスト結果は、スイープ テスト結果ファイルにも書き込まれ、このファイルは後でオフラインで解析を実行する際に使用できます。[Test Controls] パネルの下に [Log File Settings] ボタンをクリックし、ファイルのディレクトリと名前を設定できるダイアログ ボックスを開きます。

## IBERT のスイープ プロットの GUI を使用してデータのプロットを描画

次に、複数のカーブを持つバスタブ プロットを作成し、RX Eq パラメーターの設定を変えてトランシーバーのマージンをグラフで比較します。バージョン 12.3 またはそれ以前のソフトウェアを使用する場合は、別のスプレッドシート ソフトウェア プログラムを使用し、前の手順のスイープ データ結果でグラフを描く必要がありますが、バージョン 13.1 からは、ChipScope Pro Analyzer の [IBERT Console] ウィンドウに IBERT スイープ プロットの GUI 機能が含まれるようになっていきます。

### IBERT スイープ プロットの GUI

IBERT スイープ プロットの GUI を使用すると、次が可能になります。

- スイープ データが使用可能になると、[IBERT Console] ウィンドウ内で直接バスタブ カーブのプロットが描画できるようになります。
- トランシーバーからのスイープ データを外部のスプレッドシート ソフトウェアを使用しなくても効率的に解析できます。
- スタンドアロン モードで動作するので、1 つまたは複数のスイープ データ ファイル (カンマ区切りフォーマット) を読み込むことができ、統合モードと同様の方法でデータを解析できます。スタンドアロン モードは、ボードにアクセスがない場合にスイープ データをオフラインで解析する際に便利です。統合モードの場合は、ライブ モードにローカルまたはリモートで接続しておく必要があります。



## バスタブ カーブのプロット

1. [Sweep Test Settings Panel] の一番下の [Sweep Test Plot] タブをクリックし、バスタブ カーブのプロットを描画します。

図 10 のようにバスタブ プロットが 4 つ表示されます。表示オプションでは、次を変更できます。

- 1 つまたは複数のプロットを描画
- 左右または BER マーカーを移動することで、各プロットのマージンを確認
- プロットの表示/非表示
- 線の色を指定または変更

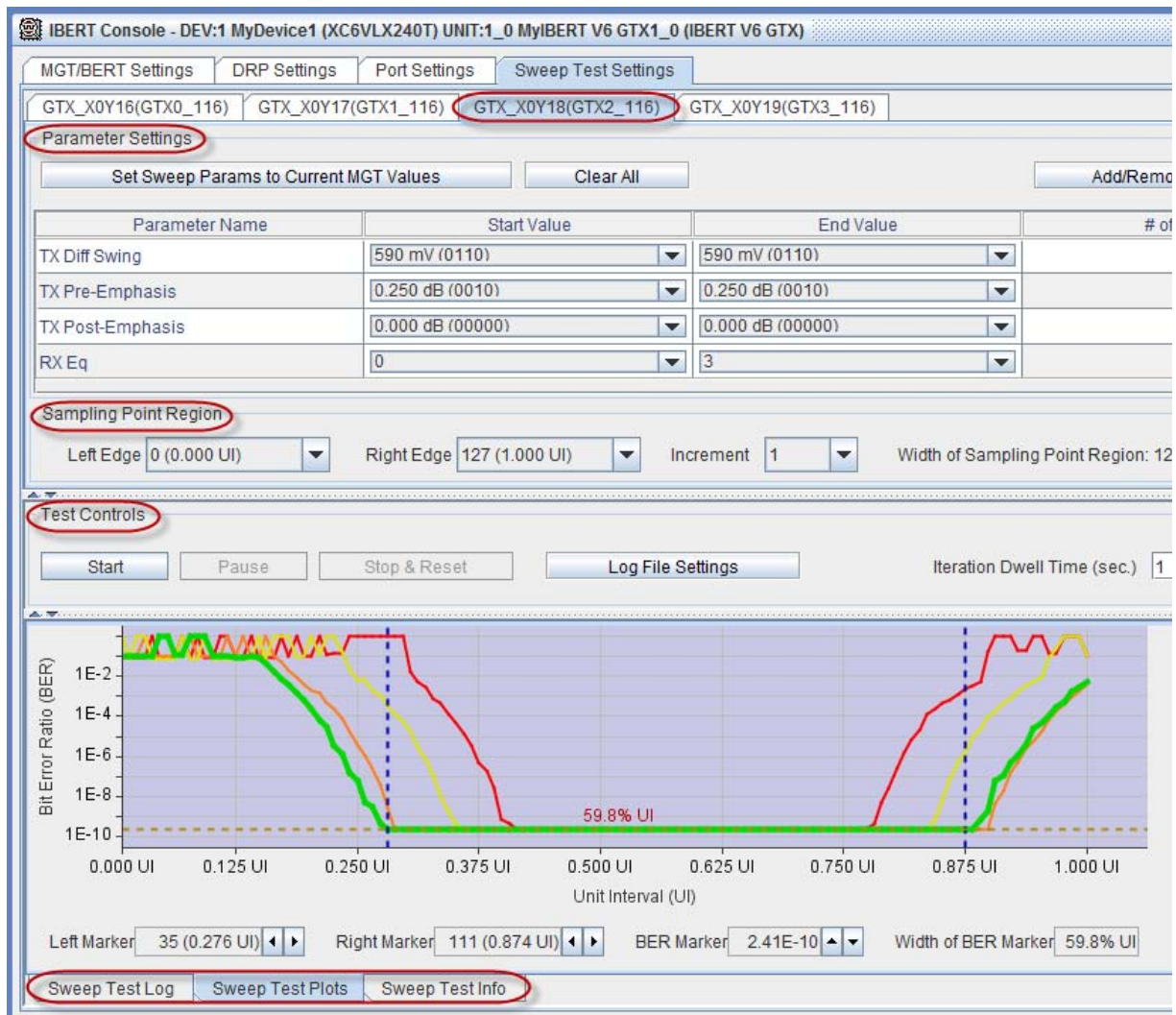


図 10 : スイープ テスト プロットの表示

2. [Sweep Test Info] タブ (図 11) をクリックし、アイ開口部およびマージンの一番大きいところを見つけます。

[Opening at Lowest BER Level] 列で並び替えることができます。このチュートリアルの場合、RX Eq パラメーターが 3 に設定された場合に、プロット 3 の開口部とマージンが一番大きいようです。

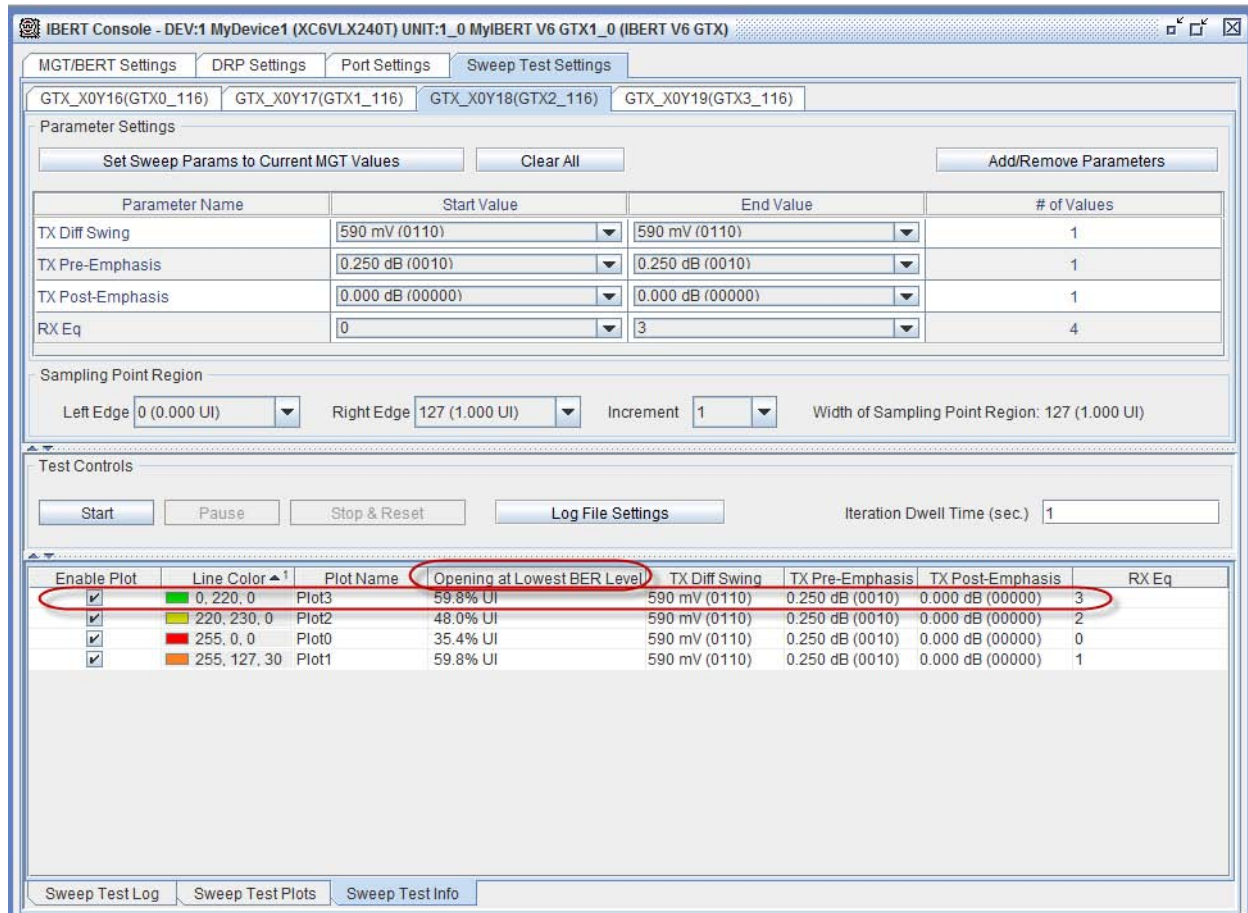


図 11 : [Sweep Test Info] タブ

スイープ テストの実行の詳細については、<http://japan.xilinx.com/tools/cspro.htm> から『ChipScope Pro ソフトウェアおよびコア ユーザー ガイド』を参照してください。

## その他のリソース

---

### ザイリンクス リソース

- ザイリンクス 資料 :  
<http://japan.xilinx.com/support/documentation>
- 用語集 :  
[http://japan.xilinx.com/support/documentation/sw\\_manuals/glossary](http://japan.xilinx.com/support/documentation/sw_manuals/glossary)
- ザイリンクス サポート :  
<http://japan.xilinx.com/support>

### ChipScope 資料

- 『ChipScope™ Pro ソフトウェアおよびコア ユーザー ガイド』(UG029) :  
[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx13\\_1/chipscope\\_pro\\_sw\\_cores\\_ug029.pdf](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx13_1/chipscope_pro_sw_cores_ug029.pdf)
- 『Using Xilinx ChipScope Pro ILA Core with Project Navigator to Debug FPGA Applications』(UG750) :  
[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx13\\_1/ug750.pdf](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx13_1/ug750.pdf)

### ボード 資料

- Spartan-6 FPGA SP601 評価キットの情報 :  
<http://japan.xilinx.com/products/devkits/EK-S6-SP601-G.htm>

