



XAPP476 (v1.0) 2003 年 7 月 10 日

## Spartan-3 FPGA 用の BSDL モデル

### 概要

BSDL (Boundary Scan Description Language) ファイルは、IEEE 1149.1 (JTAG) に準拠するザイリンクス デバイスの全パーツおよびパッケージに対して提供されています。このファイルをサードパーティのバウンダリ スキャン ツールに読み込むと、テスト ベクタを作成してテストを実行できます。また、ザイリンクスのプログラム ソフトウェアでもバウンダリ スキャンでデバイスをコンフィギュレーションする場合に BSDL ファイルを使用します。

### バウンダリ スキャンの概要

バウンダリ スキャン テストは、未接続ピンやショート ピンなどのボード レベルの接続問題を確認するために実行します。このテストにより、特に多数のボール グリッドを含むパッケージでは検出できないこともあった製造問題やレイアウト問題を素早く検出できるようになりました。最近では、バウンダリ スキャンは、デバイスのコンフィギュレーション手段の 1 つとして、ザイリンクスをはじめとするさまざまな PLD ベンダによって使用されています。

### IEEE 標準

JTAG (Joint Test Action Group) は IEEE 1149.1 標準の別名であり、バウンダリ スキャンの方法を定義しています。JTAG 準拠のデバイスには、ステート マシンと複数のレジスタを含む専用ハードウェアが含まれ、バウンダリ スキャンが実行できるようになっています。この専用ハードウェアにより、TDI (Test Data In)、TDO (Test Data Out)、TMS (Test Mode Select)、TCK (Test Clock) の 4 つの専用信号で提供される命令およびデータが解釈されます。JTAG 準拠のデバイスでは、TDI および TMS 信号の命令およびデータが解釈され、TDO 信号に駆動されます。TCK 信号はプロセスのクロック用に使用されます。

IEEE 1532 は、IEEE 1149.1 JTAG 標準を基に作成されています。IEEE 1532 を使用すると、プログラマブル ロジック デバイスのコンフィギュレーションにおいて柔軟性がさらに増します。この標準により、複数デバイスの同時プログラム、シリコンの拡張機能を使用したプログラム回数の削減、管理をさらに容易にする強力なシステムの作成が可能です。この標準では、インシステム プログラマブル ロジック デバイスをコンフィギュレーションするのに必要な次の 3 つの要素が定義されています。

- コンフィギュレーション用デバイス アーキテクチャのコンポーネント
- アルゴリズム記述のフレームワーク
- コンフィギュレーション データ ファイル

IEEE 1532 JTAG 標準の詳細については、[http://www.xilinx.co.jp/xlnx/xil\\_prodcats/product.jsp?title=isp\\_standards\\_specs#1532](http://www.xilinx.co.jp/xlnx/xil_prodcats/product.jsp?title=isp_standards_specs#1532) を参照してください。

### バウンダリ スキャン ツール

バウンダリ スキャン テストには、特定の装置とソフトウェアが必要です。テスト ベクタを作成するには、バウンダリ スキャン テスト用のソフトウェアを使用します。このテスト ベクタは通常 PC に接続されたテスト ポッドを使用してバウンダリ スキャン チェーンに渡されます。

バウンダリ スキャン テストのベクタを作成するには、テスト ソフトウェアに次のバウンダリ スキャン チェーンの情報を読み込む必要があります。

© 2003 Xilinx, Inc. All rights reserved. すべての Xilinx の商標、登録商標、特許、免責事項は、<http://www.xilinx.co.jp/legal.htm> にリストされています。他のすべての商標および登録商標は、それぞれの所有者が所有しています。すべての仕様は通知なしに変更される可能性があります。

保証否認の通知 : Xilinx ではデザイン、コード、その他の情報を「現状有姿の状態」で提供しています。この特徴、アプリケーションまたは規格の一実施例としてデザイン、コード、その他の情報を提供しておりますが、Xilinx はこの実施例が権利侵害のクレームを全く受けないということを表明するものではありません。お客様がご自分で実装される場合には、必要な権利の許諾を受ける責任があります。Xilinx は、実装の妥当性に関するいかなる保証を行なうものではありません。この保証否認の対象となる保証には、権利侵害のクレームを受けないことの保証または表明、および市場性や特定の目的に対する適合性についての黙示的な保証も含まれます。

1. バウンダリ スキャン チェーンの構造 (デバイス数、タイプなど)  
チェーン構造は、ユーザーが指定するか、バウンダリ スキャン ソフトウェアで自動的に検出されます。
2. 各デバイスのバウンダリ スキャン アーキテクチャ (IR 幅、OPCODE、I/O 数、各 I/O のビヘイビア)  
各デバイスのバウンダリ スキャン アーキテクチャは、BSDL ファイルで定義されます。
3. デバイスの I/O 同士の接続方法  
この情報は、通常ボード レベルのネットリストから抽出されます。

## BSDL ファイル

JTAG 準拠デバイスの製造元からは、必ずそのデバイスの BSDL ファイルが提供されます。BSDL ファイルには、デバイスの各ピンの機能に関する情報が含まれます (I/O やパワー、グランドなど)。ザイリンクスの BSDL ファイルの拡張子は .bsd です。

ザイリンクス デバイスの BSDL ファイルは開発システムまたはザイリンクスの Web サイト ([http://www.xilinx.co.jp/support/sw\\_bsd.html](http://www.xilinx.co.jp/support/sw_bsd.html)) から入手できます。ザイリンクス以外の製造元の BSDL ファイルについては、その企業の Web サイトを参照してください。

ザイリンクス製品用には、IEEE 1532 BSDL ファイルのプロトタイプ版も用意しています。これらのファイルは、開発システムまたは <http://www.xilinx.co.jp/isp/1532download.htm> から入手可能です。

IEEE 1149.1 の BSDL ファイルは <device\_name>.bsd のように表示されます。

具体例: xc3s50\_pq208.bsd

IEEE 1532 の BSDL ファイルは <device\_name>\_1532.bsd のように表示されます。

具体例: xc3s50\_pq208\_1532.bsd

プロトタイプ版の IEEE 1532 標準の BSDL ファイルは IEEE 1149.1 標準の BSDL ファイルの代わりには使用できません。また、この 2 つのファイルを同時に使用することはできません。

## BSDL ファイルの構成

BSDL ファイルには、JTAG 準拠デバイスのバウンダリ スキャン アーキテクチャが VHDL で記述されています。BSDL ファイルは、主に次の 8 セクションで成り立っています。

### 1. entity 宣言

entity 宣言は、その BSDL ファイルで記述されるデバイス名を表す VHDL 文です。

xc3s50\_pq208.bsd ファイルの例

```
entity XC3S50_PQ208 is
```

### 2. generic パラメータ

generic パラメータでは、パッケージが指定されます。

xc3s50\_pq208.bsd ファイルの例

```
generic (PHYSICAL_PIN_MAP : string := "PQ208" );
```

### 3. ポートのロジック情報

この部分では、デバイスの全パッドがリストされます。ピンが入力の場合は「in bit;」、出力の場合は「out bit;」、双方向の場合は「inout bit;」、バウンダリ スキャンでは使用できない場合は「linkage bit;」と表示されます。

xc3s50\_pq208.bsd ファイルの例

```
port (
  GND: linkage bit_vector (1 to 28);
  CCLK_P104: inout bit;
  DONE_P103: inout bit;
  HSWAP_EN_P206: in bit;
  M0_P55: in bit;
  M1_P54: in bit;
  M2_P56: in bit;
  PROG_B: in bit;
  TCK: in bit;
  TDI: in bit;
  TDO: out bit;
  TMS: in bit;
  VCCAUX: linkage bit_vector (1 to 8);
  VCCINT: linkage bit_vector (1 to 4);
  VCCO0: linkage bit_vector (1 to 2);
  IO_P2: inout bit; -- PAD124
  IO_P3: inout bit; -- PAD123
```

#### 4. パッケージ ピンのマップ

この部分では、チップ上のパッドがデバイス パッケージのピンにどのようにワイヤ接続されているかが表示されます。

xc3s50\_pq208.bsd ファイルの例

```
constant PQ208: PIN_MAP_STRING:=
  "GND: (P1, P8, P14, P25, P30, P41, P47, P53, P59, P66, " &
    " P75, P82, P91, P99, P105, P112, P118, P129, P134, P145, " &
    " P151, P157, P163, P170, P179, P186, P195, P202), " &
  "CCLK_P104: P104, " &
  "DONE_P103: P103, " &
  "HSWAP_EN_P206: P206, " &
  "M0_P55: P55, " &
  "M1_P54: P54, " &
  "M2_P56: P56, " &
  "PROG_B: P207, " &
  "TCK: P159, " &
  "TDI: P208, " &
  "TDO: P158, " &
  "TMS: P160, " &
  "VCCAUX: (P17, P38, P69, P89, P121, P142, P173, P193), " &
  "VCCINT: (P70, P88, P174, P192), " &
  "VCCO0: (P188, P201), " &
  "IO_P2: P2, " &
  "IO_P3: P3, " &
```

#### 5. use 文

use 文では、属性、タイプ、定数、BSDL ファイルで参照されるその他の情報を含む VHDL パッケージが呼び出されます。

xc3s50\_pq208.bsd ファイルの例

```
use STD_1149_1_1994.all;
```

#### 6. スキャン ポートの識別

この部分では、JTAG ピンの TDI、TDO、TMS、TCK、TRST が識別されます。TRST はオプションの JTAG ピンで、ザイリンクス デバイスでは使われていません。

xc3s50\_pq208.bsd ファイルの例

```
attribute TAP_SCAN_IN    of TDI : signal is true;
attribute TAP_SCAN_MODE of TMS : signal is true;
attribute TAP_SCAN_OUT  of TDO : signal is true;
attribute TAP_SCAN_CLOCK of TCK : signal is (33.0e6, BOTH);
```

## 7. TAP 情報

TAP 情報には、デバイスの JTAG ロジックの詳細な情報が表示されます。この情報には、IR 幅、OPCODE 命令、デバイスの ID コードなどが含まれます。これらの情報はデバイスによって異なります。

xc3s50\_pq208.bsd ファイルの例

```
attribute COMPLIANCE_PATTERNS of XC3S50_PQ208 : entity is
  "(PROG_B) (1)";
attribute INSTRUCTION_LENGTH of XC3S50_PQ208 : entity is 6;
attribute INSTRUCTION_OPCODE of XC3S50_PQ208 : entity is
  "EXTEST      (000000)," &
attribute INSTRUCTION_CAPTURE of XC3S50_PQ208 : entity is
  "XXXX01";
attribute IDCODE_REGISTER of XC3S50_PQ208 : entity is
  "XXXX" &      -- version
  "0001010" &   -- family
  "000001100" & -- array size
  "00001001001" & -- manufacturer
  "1";          -- required by 1149.1
```

## 8. バウンダリ レジスタ情報

この部分には、デバイスのバウンダリ スキャン セルのストラクチャが含まれます。デバイスの各ピンには最大で 3 つのバウンダリ スキャン セルが含まれ、各セルにレジスタとラッチがそれぞれ 1 つずつ含まれます。バウンダリ スキャン テスト ベクタは、これらのレジスタにロードされるか、これらのレジスタからスキャンされます。

xc3s50\_pq208.bsd ファイルの例

```
attribute BOUNDARY_REGISTER of XC3S50_PQ208 : entity is
  " 0 (BC_2, *, controlr, 1)," &
  " 1 (BC_2, IO_P161, output3, X, 0, 1, PULL0)," & -- PAD30
  " 2 (BC_2, IO_P161, input, X)," & -- PAD30
```

## BSDL ファイルの検証

ザイリンクスの BSDL ファイルの検証は、2 段階に分かれています。Preliminary (暫定) ファイルはザイリンクス標準の BSDL ファイルが自動的に生成されるプロセスで作成されます。このプロセスはスクリプト ベースで、デバイスのデザイン ファイル (アーキテクチャとピン配置がすべて記述された情報) から直接抽出されます。この Preliminary と記された BSDL ファイルは質が高く、その構文は常にテストされています。Final と記されたファイルは、構文およびハードウェア検証のテストをすべてパスしています。BSDL ファイルが各ピンの動作を正しく記述するために、ザイリンクスではサードパーティ (Intellitech 社) のバウンダリ スキャン ツールを使用し、BSDL ファイルに対して実際のシリコンを検証しています。

ザイリンクスの BSDL ファイルは、Intellitech 社の製品 Eclipse で BSDL 構文チェック (strict) を使用して 1149.1 標準に準拠しているかどうか確認されています。IEEE 1149.1b-1994 (BSDL 構文の標準) に記述されている意味チェックはすべて strict 解析を使用して実行されます。意味チェックが終了すると、各 I/O ピンに対するテストを記述した BSDL ファイルからテスト パターンが作成されます。ザイリンクスのデバイスとパッケージの組み合わせは、Intellitech 社の RCT (Reduced Contact Tester) でテストされます。このテスト パターンには、Test-Logic-Reset と TAP コントローラの動作、BYPASS/IDCODE/USERCODE 命令とレジスタ、すべての入力/出力/双方向/クロック ピンとコント

ロールセルに対するバウンダリレジスタのピンマップの検証が含まれます。最後に、デザインをダウンロードし、RCT テスタを使用してコンフィギュレーション後の検証が行われ、デバイスがプログラムされると、各デバイスが 1149.1 標準でテストされます。

## BSDLANno の使用

コンフィギュレーションをしていない Spartan-3 デバイスでは、できるだけバウンダリ スキャン テストを実行する必要があります。これは、コンフィギュレーションされていないデバイスの方が双方向の スキャン ベクタに対してすべての I/O を使用でき、テスト カバレッジが上がるためです。コンフィギュレーションが中断できない場合、および差動信号の標準が使用されている場合、バウンダリ スキャン テストはコンフィギュレーション後に実行してください。差動信号がザイリンクス デバイス間にある場合は、両方のデバイスともコンフィギュレーション前にテストできます。差動信号ペアの各サイドはそれぞれシングル エンド信号として動作します。

ザイリンクスの提供する BSDL ファイルには、コンフィギュレーションされる前のデバイスのバウンダリ スキャン ビヘイビアが記述されています。コンフィギュレーション後は、このバウンダリ スキャン ビヘイビアは変わります。たとえば、コンフィギュレーション前に双方向だった I/O ピンは、入力のみ、出力のみ、双方向、使用不可のいずれかに変わる可能性があります。バウンダリ スキャン テスト ベクタは通常 BSDL ファイルから抽出されます。このため、コンフィギュレーション後のザイリンクス デバイスでバウンダリ スキャン テストを実行する場合、BSDL ファイルをコンフィギュレーション後のバウンダリ スキャン ビヘイビアに合わせて変更する必要があります。

バウンダリ スキャン アーキテクチャはデバイスがコンフィギュレーションされると変更されます。これは、バウンダリ スキャン レジスタが I/O バッファとセンス増幅器の後ろに配置されているためです。ハードウェアがこのように配置されていると、バウンダリ スキャン ロジックはデザインで指定されている I/O 標準で動作するので、使用可能な I/O 標準の全範囲に渡ってバウンダリ スキャン テストを実行できます。

バウンダリ スキャン レジスタとパッドの接続は一部変更されることがあるため、バウンダリ スキャン アーキテクチャはデバイスのコンフィギュレーション後に変更されます。コンフィギュレーション後の BSDL ファイルをバウンダリ スキャン テスタに読み込んでから、テストを行ってください。このバウンダリ スキャン アーキテクチャが BSDL ファイルで変更されていないと、バウンダリ スキャン テストはエラーになります。

ザイリンクスの BSDLANno ユーティリティを使用すると、コンフィギュレーション後のテスト用に BSDL ファイルが自動的に変更されます。BSDLANno では、配線済み NCD ファイルから必要なデザイン情報が読み込まれ、コンフィギュレーション後のバウンダリ スキャン アーキテクチャに合わせた BSDL ファイルが作成されます。

次の構文を使用すると、BSDLANno でコンフィギュレーション後の BSDL ファイルが作成されます。

```
bsdlanno [options] infile[.ncd] outfile[.bsd]
```

infile は配線済み (PAR 後) の NCD デザイン ソース ファイルです。outfile[.bsd] はそのデザインの BSDL ファイルの出力ファイルです。拡張子 .bsd はオプションです。BSDLANno の詳細については、アンサー #15346 ([http://www.xilinx.co.jp/xlnx/xil\\_ans\\_display.jsp?i&getPagePath=15346](http://www.xilinx.co.jp/xlnx/xil_ans_display.jsp?i&getPagePath=15346)) を参照してください。

## ソフトウェア サポート

ザイリンクスでは、デバイス ファイルの作成およびデバイスのプログラムを実行するツールを複数提供しています。バウンダリ スキャン テスト機能は、複数のサードパーティ ベンダから入手可能です。詳細については、[www.xilinx.co.jp/support](http://www.xilinx.co.jp/support) の [ハードウェア] → [コンフィギュレーション ソリューション] を参照してください。

## iMPACT

iMPACT はザイリンクスの FPGA、CPLD、PROM すべてのコンフィギュレーションおよびプログラムを行うソフトウェア ツールです。ウィザードに従うだけで、簡単にコンフィギュレーション プロセス

のステップがすべて終了できます。iMPACT では SVF などの主な出力ファイル タイプがサポートされています。このコンフィギュレーション ソフトウェア iMPACT を使用すると、スレーブ シリアル、SelectMAP (スレーブ パラレル)、JTAG IEEE 1149.1 の 3 つのモードを使用してすべてのザイリンクス FPGA を簡単にコンフィギュレーションできます。サポートされるケーブルは、パラレル ケーブル IV と MultiPRO ケーブルです。

iMPACT の JTAG モードには、ケーブルの動作と JTAG チェーンの信頼性の両方をテストできる機能が含まれます。iMPACT でユーザー コード ロケーションへの書き込み、およびリード バックを数千回指定すると、チェーンの動作をテストできます。iMPACT ではこの動作で発生したエラーの回数がカウントされ、これを基に JTAG チェーンの信頼性およびノイズ耐性、ボード レイアウトなどが評価可能です。

## SVF ファイル

SVF (Serial Vector Format) は JTAG チェーンの動作をコンパクトに記述した業界標準のファイル形式です。SVF ファイルには、既にデバイスのプログラム情報がすべて含まれているため、デバイスに対する複雑な知識は必要ありません。SVF ファイルを作成する機能は、iMPACT に含まれています。詳細については、iMPACT のオンライン ヘルプを参照してください。

## IEEE 1532 プログラム用 J Drive Engine

ザイリンクスは、世界初の IEEE 1532 プログラム エンジンである J Drive Engine を開発しました。このエンジンと PC のパラレル ポートに接続されたシンプルなケーブルを使用すると、Xilinx IEEE 1532 に準拠した PLD が簡単にコンフィギュレーションできます。J Drive Engine にプログラムするデバイスのデータと 1532 の BSDL ファイルを読み込み、コマンド ラインで JTAG チェーンの PLD をコンフィギュレーションします。詳細については、[http://www.xilinx.co.jp/xlnx/xil\\_prodcatsystemsolution.jsp?title=isp\\_jdrivemain\\_page](http://www.xilinx.co.jp/xlnx/xil_prodcatsystemsolution.jsp?title=isp_jdrivemain_page) を参照してください。

## BSCAN\_SPARTAN3 マクロの使用

BSCAN\_SPARTAN3 を使用すると、Spartan-3 デバイスの BSCAN サイトへアクセスできます。このマクロは、内部バウンダリ スキャン チェーンを作成するのに使用します。Spartan-3 FPGA の場合、4 つの JTAG 専用ピン (TDI、TDO、TCK、TMS) が含まれます。バウンダリ スキャン用に標準的な JTAG を使用する場合は、これらの JTAG ピンをポートに接続するだけです。BSCAN\_SPARTAN3 シンボルのピンは、内部スキャン チェーンを駆動する特別な機能が必要な場合にのみ接続します。

Spartan-3 FPGA には、USER1 と USER2 命令を介してユーザーが定義可能な 2 本のスキャン チェーンがあります。これらの命令を使用すると、JTAG インターフェイスを介してデザインへアクセスできます。このオプションの USER1 と USER2 命令を使用する場合は、ソース コードで BSCAN\_SPARTAN3 マクロをインスタンス化してユーザー定義のスキャン チェーンに接続する必要があります。

## 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	改訂内容
07/10/03	1.0	初期リリース