

# Spartan-6 FPGA コンフィギュレーション

## ユーザー ガイド

UG380 (v2.2) 2010 年 7 月 30 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 2009–2010 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. PCI, PCI Express, PCIe, and PCI-X are trademarks of PCI-SIG. All other trademarks are the property of their respective owners.

本資料は英語版 (v2.2) 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。

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](mailto:jpn_trans_feedback@xilinx.com) までお知らせください。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。

## 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	説明
2009 年 6 月 24 日	1.0	初版リリース

日付	バージョン	説明
2010 年 2 月 17 日	2.0	<p>この文書全体を通じて REBOOT コマンドを IPROG コマンドに変更。</p> <p>第 1 章: 「高速度を重視したオプション」で、コンフィギュレーション データのサイズを 3.6Mb (XC6SLX16) に変更。20 ページの「集積度の異なる FPGA への移行」で、必要なコンフィギュレーション メモリのサイズを 2.6Mb (XC6SLX9) および 3.6Mb (XC6SLX16) に変更。「不正な複製から FPGA ビットストリームの保護するには」で、Spartan-6 デバイスが AES 復号化ロジックを使用していることを明示。</p> <p>第 2 章: 表 2-1 の後の「注意」を削除。図 2-2、図 2-3、図 2-6、図 2-7、図 2-12、図 2-20 で、VCCO_2 の抵抗を 2.4kΩ に変更。V<sub>FS</sub> および V<sub>BATT</sub> ポートを追加。SUSPEND ピンを追加。それぞれの図の注記で、末尾に 4 つの説明を追加。図 2-2 および図 2-6 で、Spartan-6 FPGA VCCO_2 およびプラットフォーム フラッシュ PROM V<sub>CCO</sub> 電源入力に関する注記から「2.5V または 3.3V」という記述を削除。図 2-12 の注記 12 および図 2-20 の注記 10 に PLL ロック待機の記述を追加。図 2-2 で、PROGRAM_B のプルアップ電源を VCCO_2 に変更。図 2-4 からスレーブ DIN を削除。「SelectMAP コンフィギュレーション インターフェイス」の最初の段落に SelectMAP が使用できないデバイスに関する記述を追加。表 2-3 の BUSY の説明にトグルに関する記述を追加。図 2-6 で、PROGRAM_B に 4.7kΩ のプルアップ抵抗を追加。図 2-6 の注記 14 に BUSY の記述を追加。図 2-7 の注記 2 に「コンフィギュレーション中」の記述を追加。表 2-6 および表 2-7 の配置を変更。表 2-6 で、Winbond の SPI フラッシュの記述を削除。「CSI_B」の最初の段落を変更。「RDWR_B」を改訂。図 2-9 の注記 1 で、CSI_B が同期ワード中にディアサートできないことを明示。図 2-12 で、3.3V を VCCO_2 に変更。「マスタ BPI コンフィギュレーション インターフェイス」で、BPI インターフェイスがサポートされないデバイスおよびパッケージを更新、CSG225 パッケージに A22 および A23 がないことを明示、トップブート パラレル NOR フラッシュの記述を追加。表 2-7 で、HDC および LDC の説明から BYTE# ポートの記述を削除。図 2-20 で、VCCO_1 および BYTE# を VCCO_1 に接続。プルアップ抵抗を FCS_B、FOE_B、FWE_B に追加。図 2-20 に注記 5 および 6 を追加。「コンフィギュレーション クロック (CCLK) のボード レイアウト」で、CCLK に関してダブル クロッキングを回避するため反射が起きないようにするという内容のメモを削除。</p> <p>第 4 章: 「ICAP_SPARTAN6」の第 1 段落の末尾の文を変更。</p> <p>「STARTUP_SPARTAN6」の第 1 段落で、EOS をコンフィギュレーションに変更。</p> <p>第 5 章: この章全体を通して、ロック状態の待機に DCM だけでなく PLL の記述も追加。表 5-1 で、V<sub>FS</sub>、V<sub>BATT</sub>、RFUSE の行を追加、メモ 4 を追加、ピン名の CMP_CS_B を CMPCS_B に変更し、関連する説明を更新。「コンフィギュレーション中の FPGA I/O ピンの設定」を第 1 章から、「多目的コンフィギュレーション ピン (Persist) の予約」を第 2 章から移動。「コンフィギュレーション中の FPGA I/O ピンの設定」で、すべてのユーザー I/O ピンにオプションのプルアップ抵抗があることを明示。表 5-2 でメモ 3 を追加。表 5-3 で、メモ 1 を追加し、メモ 2 を改訂。表 5-5 で、「コンフィギュレーション ビットの総数」列の値を変更。「デバイスの電源投入 (手順 1)」で、第 2 段落および第 3 段落を変更、第 4 段落に -4 の記述を追加。表 5-11 で、V<sub>FS</sub> および VCCO_5 を追加、V<sub>FS</sub> および V<sub>BATT</sub> の説明を変更、「値」列および「単位」列を削除、メモ 1、4、5 を追加、メモ 2 を更新して V<sub>FS</sub> の記述を追加。図 5-4 の下の 2 番目の段落を変更。「デバイス ID の確認 (手順 5)」の最後の段落を変更。「スタートアップ (手順 8)」の最初の段落で、シーケンシャル ステート マシンのクロックの記述を追加。表 5-17 で、DCM_LOCK の説明を改訂、メモ 3 の記述を「スタートアップ (手順 8)」の本文に移動。表 5-17 の後に新しい段落を追加。「暗号化キーの読み込み」で、プログラミング ケーブルの種類を明示、最後の段落の末尾の文を変更。「暗号化されたビットストリームの読み込み」の第 4 段落および第 5 段落の変更。</p>

日付	バージョン	説明
2010 年 2 月 17 日 (続き)	2.0	<p>「eFUSE」を追加。表 5-22 で、「ビット総数」列の値を変更。表 5-30 で、GENERAL2 および GENERAL4 の説明を改訂。「ブート履歴ステータスレジスタ (BOOTSTS)」で、レジスタのリセット方法の説明を変更。表 5-48 で、ビット 2 および 8 を「予約済み」に変更。図 5-16 で、DOUT と DIN の間にバッファを追加。図 5-16 の前に、新しいバッファに関する説明を追加。「ビットストリーム圧縮」を追加。</p> <p>第 6 章：第 1 段落を変更。表 6-1 で、手順 6 および 12 の「コンフィギュレーションデータ [15:0]」の値を変更。表 6-1 の下の最初の文で、手順の番号を変更。図 6-2 の前の段落で、SelectMAP のデータ順序に関する文を追加。図 6-2 で、タイミング図を変更。</p> <p>第 7 章：「マルチブートの概要」で、最後の段落を変更、「注意」を削除。「フォールバック動作」で、多くの記述を変更。「ICAP_SPARTAN6 を使用した再起動」で、最初の段落の「次のビットストリーム」を「MultiBoot ビットストリーム」に変更。コマンドシーケンスの手順 2 を変更。表 7-1 で、同期ワードの値を入れ替え、「説明」列の記述を変更、メモ 1 および 2 を追加。「ウォッチドッグ タイマ」で、始めの 3 つの段落の最初の文を変更。</p> <p>第 8 章：142 ページで、1 番目の簡条書きのスライスをフレームに変更、4 番目の簡条書きを改訂、トランシーバ DRP がマスクされないことを記述した簡条書きを削除。</p> <p>第 9 章：表 9-1 を変更。</p>
2010 年 2 月 22 日	2.1	<p>「ビットストリーム暗号化」で、サポートされているデータ幅を x1 および x8 に変更。「暗号化されたビットストリームの読み込み」の第 3 段落で、コンフィギュレーションビットストリームがデータ幅 x1 または x8 のコンフィギュレーション モードで提供できることを明示。SPI x2 および x4、BPI x16、SelectMAP x16 のバス幅は暗号化されたビットストリームでサポートされていないことを明示。</p>

日付	バージョン	説明
2010 年 7 月 30 日	2.2	<p>図 2-2、図 2-3、図 2-6、図 2-7、図 2-12、図 2-20 で、DONE および VCCO_2 間で接続されるプルアップ抵抗の値を 2.4kW から 330W に変更。図 2-3 および 図 2-6 で、INIT_B および VCCO_2 間で接続されるプルアップ抵抗の値を 2.4kW から 4.7kW に変更。図 2-6 で、RDWR_B および CSI_B ポートを FPGA (GND に接続) に追加。「マスター モード」に、コンフィギュレーション クロック周波数について記載した第二段落および第三段落を追加。「SelectMAP コンフィギュレーション インターフェイス」に、SelectMAP についての注意点を追加。表 2-3 の RDWR_B の項目で、V<sub>REF</sub> についての記述を追加。「CSI_B」の最初の段落で、CSI_B を同期ワードの途中でディアサートしないようにすることを記載。「マスター BPI コンフィギュレーション インターフェイス」の第一段落で、箇条書きを追加して段落の構成を変更。この箇条書きで、XC6SLX25/T デバイスから BPI コンフィギュレーション インターフェイスのサポートが削除されたことを記載。図 2-22、図 2-23、図 2-24 で、「VCCO_0」を「VCCO_2」に変更。「電源供給」の第二段落を変更。表 5-2 で、「Suspend 機能が未使用の場合」という記述および注記 4 を追加。「コンフィギュレーション ピン」の第一段落で、表の参照先を表 5-4 から表 5-3 に変更。表 5-3 のタイトルに「多目的」を追加。「デバイスの電源投入 (手順 1)」の第二段落で、「LVCMOS25 8 mA SLOW」を「LVCMOS 8 mA SLOW」に変更。表 5-12 で、CCLK 出力遅延のシンボルを「T<sub>ICCK</sub>」から「T<sub>BPIICCK</sub> または T<sub>SPICCK</sub>」に変更し、注記 2 を追加。図 5-4 に続く段落で、「V<sub>POR</sub>」を「推奨動作電圧値」に変更。「スタートアップ (手順 8)」で、第四段落を追加して、LCK_CYCLE オプションを指定することで DCM および PLL のスタートアップまで待機することについて記載。図 5-13 のタイトルから「DSP」を削除。117 ページの「ビットストリームの圧縮」に、大きな利点について 3 つの箇条書きを追加。「フォールバック動作」の第一段落で、「ウォーム ブート」を「マルチブート」に変更。「フォールバック動作」の第四段落で、ビットストリームを自動的に生成する方法について記載。表 7-1 の注記 2 に文章を追加。「LCK_Cycle に必要な追加のメモリ空間」のセクション タイトルおよび本文で、「DCM_WAIT」を「LCK_Cycle」に変更。「POST_CRC_FREQ」制約の記述で、リストされている可能な値から「66」を削除。「構文例」から NCF の構文例を削除。図 9-4 で、「BPI UP」を「BPI」に変更。図 9-4 についての説明の 7 項目目で、「BPI UP、または BPI Down」を「または BPI」に変更。</p>



# 目次

---

改訂履歴.....	2
<b>このユーザー ガイドについて</b>	
内容 .....	13
その他の資料.....	13
その他のリソース .....	14
<b>第 1 章：コンフィギュレーションの概要</b>	
概要 .....	15
設計に関する考慮事項.....	16
FPGA コンフィギュレーション データ ソース.....	16
マスター モード .....	16
スレーブ モード .....	17
JTAG 接続 .....	18
基本的なコンフィギュレーション ソリューション.....	18
低コストを重視したソリューション .....	19
高速動作を重視したオプション.....	19
PCI リンクの有効化要件への準拠 .....	20
単独および複数のコンフィギュレーション イメージ .....	20
マルチブート/セーフ アップデート .....	20
I/O 電圧要件.....	20
不揮発性データの格納 .....	20
集積度の異なる FPGA への移行 .....	21
製品寿命.....	21
不正な複製から FPGA ビットストリームを保護する .....	22
同じコンフィギュレーション ビットストリームを複数の FPGA に読み込む .....	22
コンフィギュレーションを決定する要素 .....	22
<b>第 2 章：コンフィギュレーション インターフェイスの基本</b>	
JTAG インターフェイス.....	23
シリアルコンフィギュレーション インターフェイス.....	24
マスター シリアル .....	26
スレーブ シリアルコンフィギュレーション.....	27
シリアルコンフィギュレーション データ タイミング.....	29
SelectMAP コンフィギュレーション インターフェイス.....	30
単一デバイスの SelectMAP コンフィギュレーション .....	32
プラットフォーム フラッシュ PROM の SelectMAP コンフィギュレーション.....	32
マイクロプロセッサによる SelectMAP コンフィギュレーション.....	35
SelectMAP データ読み込み.....	36
CS1_B .....	36
RDWR_B.....	37
CCLK.....	37
SelectMAP データの連続読み込み .....	37
SelectMAP データの不連続読み込み .....	39
SelectMAP のデータ順 .....	40
SPI コンフィギュレーション インターフェイス .....	41
マスター SPI ベンダー自動検出およびエラー処理機能 .....	46
マスター SPI のタイミング波形 .....	47
マスター SPI デュアル (x2) およびクアッド (x4) 読み出しコマンド .....	48
パワーオン シーケンスに関する注意事項 .....	49
SPI シリアル デイジー チェーン .....	50

マスター BPI コンフィギュレーション インターフェイス .....	50
パワーオン シーケンスに関する注意事項 .....	55
マスター モードの外部コンフィギュレーション クロック .....	56
コンフィギュレーション クロック (CCLK) のボード レイアウト .....	57

### 第 3 章：バウンダリ スキャンおよび JTAG コンフィギュレーション

概要 .....	61
IEEE 1149.1 を使用した Spartan-6 デバイスのバウンダリ スキャン .....	61
テスト アクセス ポート (TAP) .....	61
バウンダリ スキャン タイミング パラメーター .....	62
Spartan-6 デバイスでのバウンダリ スキャンの使用 .....	63
設計に関する考察事項 .....	64
JTAG 信号の配線 .....	64
電源供給 .....	65
バウンダリ スキャンによるコンフィギュレーション .....	65

### 第 4 章：ユーザー プリミティブ

BSCAN_SPARTAN6 .....	67
ICAP_SPARTAN6 .....	68
STARTUP_SPARTAN6 .....	69
DNA_PORT .....	69
SUSPEND_SYNC .....	70
POST_CRC_INTERNAL .....	71

### 第 5 章：コンフィギュレーションの詳細

コンフィギュレーション ピン .....	73
コンフィギュレーション中の FPGA I/O ピンの設定 .....	74
多目的のコンフィギュレーション ピン (Persist) の予約 .....	76
コンフィギュレーション データ ファイルの形式 .....	77
ビットストリームの概要 .....	77
同期ワード/バス幅の自動検出 .....	78
PROM ファイルの生成 .....	79
シリアル デイジー チェーン用の PROM ファイル .....	79
SelectMAP コンフィギュレーション用の PROM ファイル .....	80
SPI/BPI コンフィギュレーション用の PROM ファイル .....	80
ビットのスワップ .....	80
パラレル バスのビット順 .....	81
コンフィギュレーションの遅延 .....	82
コンフィギュレーション シーケンス .....	83
セットアップ (手順 1 - 3) .....	83
デバイスの電源投入 (手順 1) .....	83
コンフィギュレーション メモリのクリア (手順 2、初期化) .....	85
モード ピンのサンプル (手順 3) .....	86
ビットストリームの読み込み (手順 4 ~ 7) .....	86
同期化 (手順 4) .....	86
デバイス ID の確認 (手順 5) .....	87
コンフィギュレーション データ フレームの読み込み (手順 6) .....	88
CRC (Cyclic Redundancy Check) (手順 7) .....	88
スタートアップ (手順 8) .....	89
ビットストリーム暗号化 .....	91
AES の概要 .....	92
暗号化したビットストリームの作成 .....	92
暗号化キーの読み込み .....	92
暗号化ビットストリームの読み込み .....	92



ビットストリームの暗号化および内部コンフィギュレーション アクセス	
ポート (ICAP) .....	93
V <sub>BATT</sub> .....	93
eFUSE .....	94
eFUSE レジスタ .....	94
eFUSE 制御レジスタ (FUSE_CNTL) .....	94
JTAG 命令 .....	96
VFS ピン .....	96
RFUSE ピン .....	97
VCCAUX ピン .....	97
コンフィギュレーション メモリ フレーム .....	97
コンフィギュレーション パケット .....	98
パケットのタイプ .....	98
タイプ 1 パケット .....	99
タイプ 2 パケット .....	99
コンフィギュレーション レジスタ .....	100
CRC レジスタ .....	101
FAR_MAJ レジスタ .....	102
FAR_MIN レジスタ .....	102
FDRI レジスタ .....	102
FDRO レジスタ .....	102
MASK レジスタ .....	102
EYE_MASK レジスタ .....	102
LOUT レジスタ .....	103
CBC_REG レジスタ .....	103
IDCODE レジスタ .....	103
CSBO レジスタ .....	103
コマンド レジスタ (CMD) .....	103
制御レジスタ 0 (CTL) .....	104
ステータス レジスタ (STAT) .....	105
コンフィギュレーション オプション レジスタ (COR1 および COR2) .....	106
サスペンド レジスタ (PWRDN_REG) .....	107
フレーム長レジスタ .....	108
マルチフレーム書き込みレジスタ .....	108
コンフィギュレーション ウォッチドッグ タイマー レジスタ .....	108
HC_OPT_REG レジスタ .....	108
GENERAL レジスタ 1、2、3、4、および 5 .....	109
MODE レジスタ .....	110
CCLK_FREQ レジスタ .....	111
PU_GWE レジスタ .....	111
PU_GTS レジスタ .....	111
ブート履歴のステータス レジスタ (BOOTSTS) .....	111
SEU_OPT レジスタ .....	112
ビットストリームの構成 .....	112
デフォルトの初期コンフィギュレーション プロセス .....	113
Spartan-6 FPGA のデバイス固有 ID (Device DNA) .....	113
ID 値 .....	114
動作 .....	114
ID メモリの仕様 .....	115
ID の拡張 .....	115
JTAG によるデバイス ID へのアクセス .....	116
iMPACT によるデバイス ID へのアクセス .....	116
ビットストリームの圧縮 .....	116

## 第 6 章：リードバックおよび

### コンフィギュレーションの検証

リードバックを実行するためのデザインの準備 .....	119
-----------------------------	-----

リードバック コマンド シーケンス.....	119
SelectMAP インターフェイスからのコンフィギュレーションレジスタへのアクセス.....	120
コンフィギュレーションレジスタの読み出し手順 (SelectMAP).....	120
コンフィギュレーションメモリの読み出し手順 (SelectMAP).....	122
JTAG インターフェイスからのコンフィギュレーションレジスタへのアクセス.....	124
コンフィギュレーションレジスタの読み出し手順 (JTAG).....	126
コンフィギュレーションメモリの読み出し手順 (IEEE Std 1149.1 JTAG).....	127
リードバックデータの検証.....	132

## 第 7 章：リコンフィギュレーションおよびマルチブート

マルチブートの概要.....	135
フォールバック マルチブート.....	136
フォールバック動作.....	136
IPROG リコンフィギュレーション.....	138
ICAP_SPARTAN6 を使用したリブート.....	138
フォールバックおよび IPROG リコンフィギュレーションに関連するステータスレジスタ.....	139
ウォッチドッグ タイマー.....	140
マルチブート イメージ間に必要なデータ空間.....	140
フラッシュ セクター、ブロック、ページの境界.....	140
LCK_Cycle に必要な追加のメモリ空間.....	141

## 第 8 章：リードバック CRC

POST_CRC 制約.....	144
POST_CRC.....	144
POST_CRC_INIT_FLAG.....	145
POST_CRC_ACTION.....	145
POST_CRC_FREQ.....	145
構文例.....	146
POST_CRC.....	146
POST_CRC_INIT_FLAG.....	146
POST_CRC_ACTION.....	146
POST_CRC_FREQ.....	146

## 第 9 章：アドバンス コンフィギュレーションインターフェイス

シリアル デイジー チェーン.....	147
混在シリアル デイジー チェーン.....	148
シリアル デイジー チェーンのガイドラインおよびデザインの考察.....	149
スタートアップ シーケンス (GTS).....	149
アクティブ DONE ドライバー.....	149
すべての DONE ピンを接続.....	149
DONE ピンの立ち上がり時間.....	149
ビットストリームのフォーマット.....	149
ギャング シリアル コンフィギュレーション.....	150
複数デバイスの SelectMAP コンフィギュレーション.....	152
パラレル デイジー チェーン.....	153
ギャング SelectMAP.....	154
SelectMAP の ABORT.....	155
コンフィギュレーションの ABORT シーケンス.....	155
リードバックの ABORT シーケンス.....	156
ABORT ステータスワード.....	156
ABORT 後にコンフィギュレーションまたはリードバックを再開.....	157
SelectMAP リコンフィギュレーション.....	158

## 第 10 章 : アドバンス JTAG コンフィギュレーション

概要 .....	159
JTAG コンフィギュレーション/リードバック .....	160
TAP コントローラーおよびアーキテクチャ .....	160
バウンダリ スキャン アーキテクチャ .....	163
バウンダリ スキャン レジスタ .....	163
命令レジスタ .....	164
BYPASS レジスタ .....	166
ID (IDCODE) レジスタ .....	166
JTAG コンフィギュレーション レジスタ .....	166
USERCODE レジスタ .....	166
USER1、USER2、USER3、USER4 レジスタ .....	166
Spartan-6 デバイスでのバウンダリ スキャンの使用 .....	167
バウンダリ スキャンによるコンフィギュレーション .....	167
スタートアップおよびシャットダウン シーケンス (JTAG) .....	170



# このユーザー ガイドについて

---

このユーザー ガイドでは、Spartan®-6 FPGA のコンフィギュレーションについて説明します。  
Spartan-6                      FPGA                      ファミリの最新の資料は、ザイリンクスのウェブサイト  
<http://japan.xilinx.com/support/documentation/spartan-6.htm> から参照できます。

## 内容

このユーザー ガイドは、次の各章から構成されています。

- 第 1 章「コンフィギュレーションの概要」
- 第 2 章「コンフィギュレーション インターフェイスの基本」
- 第 3 章「バウンダリ スキャンおよび JTAG コンフィギュレーション」
- 第 4 章「ユーザー プリミティブ」
- 第 5 章「コンフィギュレーションの詳細」
- 第 6 章「リードバックおよびコンフィギュレーションの検証」
- 第 7 章「リコンフィギュレーションおよびマルチブート」
- 第 8 章「リードバック CRC」
- 第 9 章「アドバンス コンフィギュレーション インターフェイス」
- 第 10 章「アドバンス JTAG コンフィギュレーション」

## その他の資料

次の資料も、<http://japan.xilinx.com/support/documentation/spartan-6.htm> からダウンロードできます。

- 『Spartan-6 ファミリ概要』  
Spartan-6 ファミリのファミリの特長と製品群の概要を説明しています。
- 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』  
Spartan-6 ファミリの DC 特性およびスイッチ特性の仕様が記載されています。
- 『Spartan-6 FPGA パッケージおよびピン配置仕様』  
デバイス/パッケージの組み合わせおよび最大 I/O 数の表、ピン定義、ピン配置表、ピン配置図、機械的図面、温度仕様が記載されています。
- 『Spartan-6 FPGA SelectIO リソース ユーザー ガイド』  
Spartan-6 の各デバイスで使用可能な SelectIO™ リソースについて説明しています。

- 『Spartan-6 FPGA クロック リソース ユーザー ガイド』  
Spartan-6 の各デバイスで使用可能な DCM や PLL などのクロッキング リソースについて説明しています。
- 『Spartan-6 FPGA ブロック RAM リソース ユーザー ガイド』  
Spartan-6 デバイスのブロック RAM の機能について説明しています。
- 『Spartan-6 FPGA コンフィギャブル ロジック ブロック ユーザー ガイド』  
Spartan-6 の各デバイスで使用可能なコンフィギャブル ロジック ブロック (CLB) の機能について説明しています。
- 『Spartan-6 FPGA メモリ コントローラー ユーザー ガイド』  
Spartan-6 FPGA のメモリ コントローラー ブロックについて説明しています。メモリ コントローラー ブロックは、Spartan-6 FPGA とよく使用するメモリ規格を接続する際のインターフェイスを簡略化するエンベデッド マルチポート メモリ コントローラーです。
- 『Spartan-6 FPGA GTP トランシーバー ユーザー ガイド』  
Spartan-6 LXT FPGA で使用可能な GTP トランシーバーについて説明しています。
- 『Spartan-6 FPGA DSP48A1 スライス ユーザー ガイド』  
Spartan-6 FPGA の DSP48A1 スライスのアーキテクチャについて説明し、コンフィギュレーション例も記載しています。
- 『Spartan-6 FPGA PCB デザインおよびピン配置ガイド』  
PCB およびインターフェイス レベルのデザインを決定する方法に焦点を当てた Spartan-6 デバイスの PCB デザイン情報を提供します。
- 『Spartan-6 FPGA パワー マネージメント ユーザー ガイド』  
Spartan-6 デバイスでのハードウェアによるさまざまなパワー マネージメントについて、Suspend モードに焦点をあてて説明しています。

## その他のリソース

その他の資料は、ザイリンクスのウェブサイトを参照してください。

<http://japan.xilinx.com/support/documentation/index.htm>

シリコンやソフトウェア、IP に関するアンサー データベースを検索したり、テクニカル サポートのウェブ ケースを開く場合は、次のウェブサイトにアクセスしてください。

<http://japan.xilinx.com/support>

# コンフィギュレーションの概要

---

## 概要

Spartan®-6 FPGA デバイスは、アプリケーション固有のコンフィギュレーション データ (ビットストリーム) を内部メモリに読み込んでコンフィギュレーションを行います。コンフィギュレーションの方法は、Spartan-6 FPGA 自身が外部不揮発性メモリ デバイスからビットストリームを読み込む方法と、マイクロプロセッサ、DSP プロセッサ、マイクロコントローラー、PC、ボード テスターなどから読み込む方法があります。いずれの方法でも、コンフィギュレーションに使用するデータパスには大きく 2 つの種類があります。1 つはシリアル データパスで、デバイス ピン要件を最小限に抑えたい場合に使用します。もう 1 つは 8 ビットまたは 16 ビットのデータパスで、高速パフォーマンス、業界標準のインターフェイスへの高速アクセスに使用され、プロセッサや x8 または x16 パラレル フラッシュ メモリなどの外部データ ソースに理想的です。

プロセッサやプロセッサ ペリフェラルのように、ザイリンクス FPGA は、インシステム、オンディマンドで、何度でも再プログラムできます。

ザイリンクス FPGA コンフィギュレーション データは CMOS コンフィギュレーション ラッチ (CCL) に格納されるため、いったん電源を切断するとコンフィギュレーションが必要です。ビットストリームは、毎回専用のコンフィギュレーション ピンからデバイスに読み込まれます。これらのコンフィギュレーション ピンは、次のようなコンフィギュレーション モードでインターフェイスとして機能します。

- JTAG コンフィギュレーション モード
- マスター シリアル/SPI コンフィギュレーション モード (x1、x2、x4)
- スレーブ シリアル コンフィギュレーション モード
- マスター SelectMAP/BPI コンフィギュレーション モード (x8、x16)
- スレーブ SelectMAP コンフィギュレーション モード (x8、x16)

コンフィギュレーション モードの詳細は、[第 2 章「コンフィギュレーション インターフェイスの基本」](#)を参照してください。

コンフィギュレーション モードは、モード入力ピン M[1:0] でレベルを設定して選択します。M1 および M0 モード ピンは、一定した DC 電圧レベルで設定する必要があります。これは、プルアップまたはプルダウン抵抗 (2.4kΩ) を使用するか、GND または VCCO\_2 に直接接続して設定します。モード ピンは、コンフィギュレーション中またはコンフィギュレーション前にトグルできませんが、コンフィギュレーション後はトグル可能です。モード ピンの設定オプションは、[第 2 章「コンフィギュレーション インターフェイスの基本」](#)を参照してください。

「マスター」および「スレーブ」という表現は、コンフィギュレーション クロック (CCLK) の方向を示します。

- マスター コンフィギュレーションモードでは、Spartan-6 デバイスは内部オシレーターからの CCLK を駆動するか（デフォルト）、または外部マスター クロック ソースの GCLK0/USERCCLK を駆動します (オプション)。周波数を選択するには、BitGen オプションの **-g ConfigRate** を内部オシレーターに対して使用します。詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) の「BitGen」を参照してください。コンフィギュレーション完了後は、次の条件のいずれかが満たされている場合を除き、オシレーターはオフになります。
  - SEU 検出が使用されています。
  - STARTUP プリミティブで CFGMCLK が接続されています。
  - 内部クロック ソースが Suspend モードで選択されています (オシレーターは WAKWUP シーケンス中にのみオン)。
  - 暗号化が有効になっています。

CCLK は多目的ピンです。コンフィギュレーション前はオンチップのプルアップ抵抗がありません。コンフィギュレーション後は PERSIST が使用されている場合を除き、ユーザー ピンとなります。

- スレーブ コンフィギュレーション モードのとき、CCLK は入力です。

JTAG/バウンダリ スキャン コンフィギュレーション インターフェイスは、モード ピンの設定にかかわらず、常に使用可能です。

## 設計に関する考慮事項

効率の良いシステムを構築するには、どの FPGA コンフィギュレーション モードがシステム要件に合っているのかを検討することが重要です。いずれのコンフィギュレーション モードでも、コンフィギュレーション専用の FPGA ピンだけでなくその他のピンも、一時的にコンフィギュレーションに使用可能です。コンフィギュレーションが完了すると、専用ピンでないピンは汎用ピンとなります。詳細は、[第 5 章「コンフィギュレーションの詳細」](#)を参照してください。

同様に、使用するコンフィギュレーション モードにより、FPGA I/O バンクの電圧が制限されることもあります。コンフィギュレーション オプションはいくつかあり、柔軟性がありますが、各システムに最適なソリューションがあるのが一般的です。最適なコンフィギュレーション オプションを選択するには、全体的な設定、速度、コスト、複雑さといった要因を考慮する必要があります。

## FPGA コンフィギュレーション データ ソース

Spartan-6 FPGA は柔軟性を最大限に考慮して設計されています。コンフィギュレーション データは、FPGA 自身が PROM から自動的に読み込むことも、プロセッサやマイクロコントローラーなどの外部インテリジェント デバイスを使用して FPGA にダウンロードすることもできます。

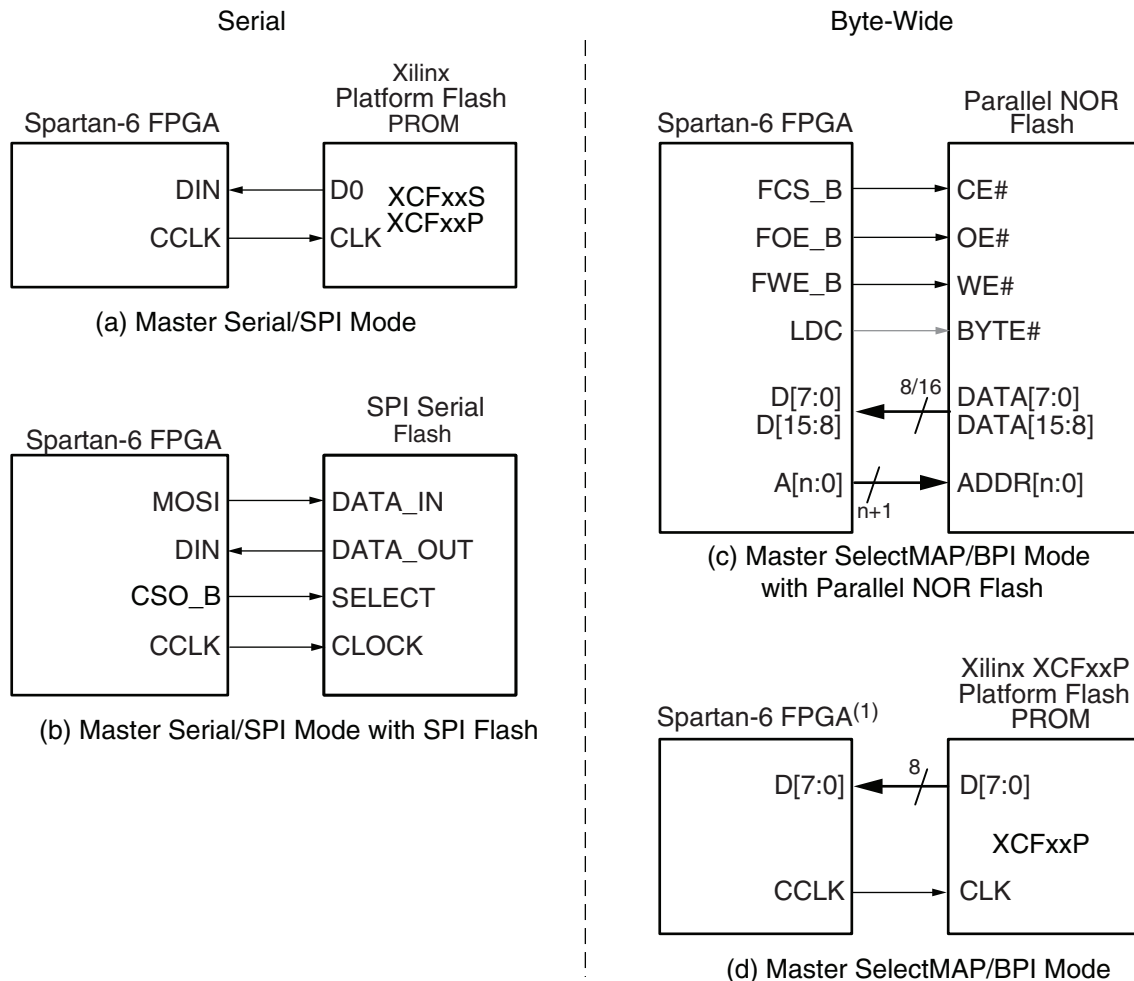
### マスター モード

FPGA の自動コンフィギュレーション モードは「マスター」モードと呼ばれています ([図 1-1](#) 参照)。このモードでは、FPGA コンフィギュレーション データをさまざまなタイプの不揮発性メモリに格納します。マスター モードでは、通常、コンフィギュレーション ビットストリームは FPGA と同じボード上の外部不揮発性メモリにあります。FPGA により、CCLK と呼ばれるコンフィギュレーション クロック信号 (内部オシレーターからのクロック信号、または外部マスター クロックソースの GCLK0/USERCCLK) が提供され、またコンフィギュレーション プロセスが制御されます。



コンフィギュレーション クロック周波数は、**bitgen -g configurate** オプションを使用してマスター モードで制御可能です。デフォルト値は、2MHz です。

選択したオプションに関係なく、マスター モードでのコンフィギュレーション クロックは 1MHz から開始します。FPGA はビットストリーム内でクロック動作を実行するため、コンフィギュレーション レート設定を読み出し、適宜変更します。



Note: The remaining Spartan-6 FPGAs support XCFxxP Platform Flash PROMs via Master SelectMAP mode.

The master serial and the master SPI configuration modes are combined and use the same mode selection.

The master SelectMAP and the master BPI configuration modes are combined and use the same mode selection.

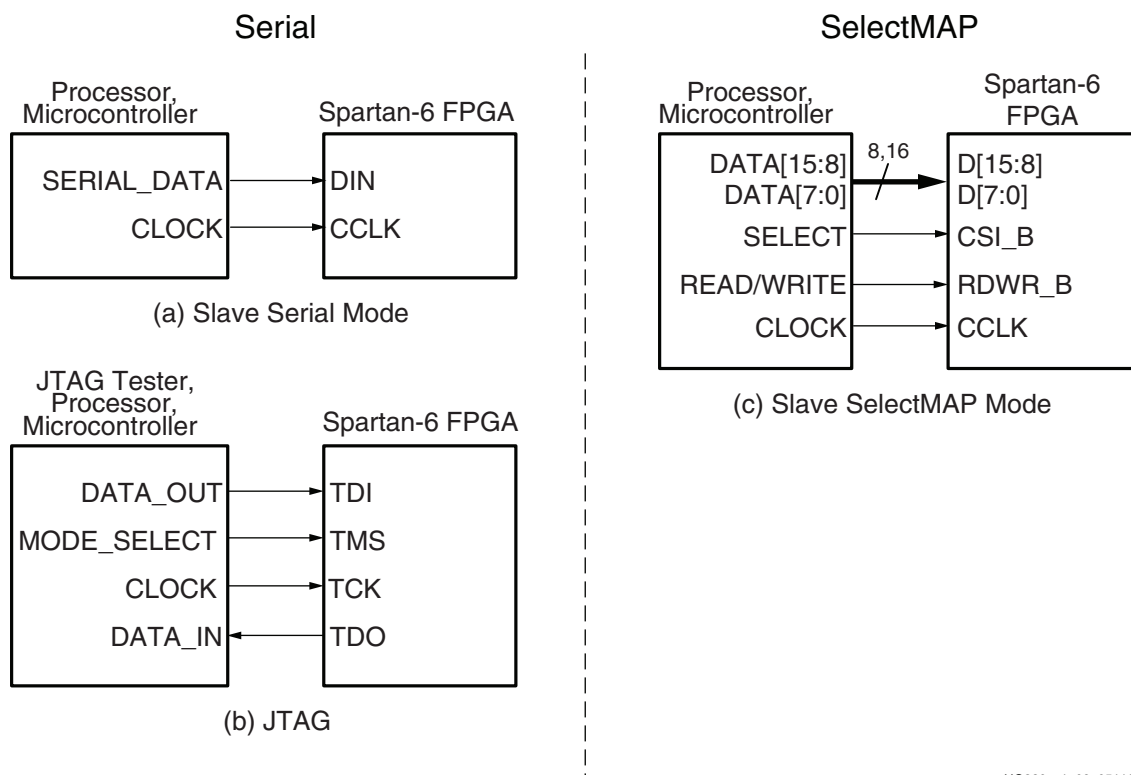
UG380\_c1\_01\_060109

図 1-1 : マスター コンフィギュレーション モード

## スレーブ モード

FPGA のコンフィギュレーションを外部から制御するコンフィギュレーション モードを「スレーブ」モードと総称します。この場合も、データパスはシリアルまたはバイト幅のどちらも使用できます。スレーブ モードでは、プロセッサ、マイクロコントローラー、DSP プロセッサ、テスターなどの外部インテリジェント デバイスを使用し、FPGA にコンフィギュレーション データをダウンロードします (図 1-2 参照)。このスレーブ コンフィギュレーションの利点は、FPGA ビットストリームをシステムのほぼどこにでも格納できることです。たとえば、オンボードのフラッシュ メモ

りにプロセッサのコードと一緒に格納することもできます。また、ハード ディスクに保存したり、ネットワークやブリッジ接続を使用してネットワーク上に保存しておくことも可能です。



UG380\_c1\_02\_051109

図 1-2：スレーブ コンフィギュレーション モード

スレーブ SelectMAP モードは、単純な x8 または x16 ビット幅のプロセッサ ペリフェラル インターフェイスで、チップ セレクト入力と読み出し/書き込み制御入力を含みます。スレーブ シリアルモードは、クロックとシリアル データ入力のみで構成される単純なモードです。

## JTAG 接続

4 本の信号線で構成される JTAG インターフェイスは、オンボード テスターとデバッグ用ハードウェアによく使用されます。Spartan-6 FPGA に対応した下記のザイリンクス プログラミング ケーブルも、プロトタイプ ダウンロードおよびデバッグ用に JTAG インターフェイスを使用します。最終的にアプリケーションで使用するコンフィギュレーション モードにかかわらず、JTAG コンフィギュレーションパスを含めておくことでデザイン開発が容易になります。第 3 章「バウンダリ スキャンおよび JTAG コンフィギュレーション」も参照してください。

- プラットフォーム ケーブル USB II  
<http://japan.xilinx.com/products/devkits/HW-USB-II-G.htm>
- パラレル ケーブル IV  
<http://japan.xilinx.com/products/devkits/HW-PC4.htm>

## 基本的なコンフィギュレーション ソリューション

基本オプションには、ザイリンクス プラットフォーム フラッシュ PROM またはサードパーティ SPI PROM を使用したマスター シリアル モードがあります。これらのソリューションでは、最少

数の FPGA ピンを使用し、柔軟性のある I/O 電圧がサポートされています。また SPI PROM はザイリンクス JTAG ベースのプログラミング ソフトウェアである iMPACT でサポートされています。詳細は、ISE ヘルプの「iMPACT ヘルプ」を参照してください。

[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx11/isehelp\\_start.htm](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx11/isehelp_start.htm)

## 低コストを重視したソリューション

どのコンフィギュレーション方法が最も低コストになるかは、アプリケーションによって異なります。

- システムに使用可能な不揮発性メモリが既にある場合、ビットストリーム イメージをシステムメモリに格納できます。また、ハードドライブに保存したり、ネットワーク接続を介してダウンロードできます。その際には、ダウンロード モードをスレーブ SelectMAP モード、スレーブ シリアル モード、または JTAG のいずれかにする必要があります。
- アプリケーションで不揮発性メモリが必要な場合、メモリは統合可能です。たとえば、FPGA コンフィギュレーションビットストリームはボードのプロセッサコードと共に格納できます。使用するプロセッサが FPGA にエンベデッドされている [MicroBlaze™](#) の場合、FPGA コンフィギュレーション データと MicroBlaze プロセッサのコードを同じ不揮発性メモリ デバイスに格納できます。
- Spartan-6 FPGA には SPI シリアル フラッシュメモリおよびパラレル NOR フラッシュメモリから直接コンフィギュレーションするオプションもあります。詳細は、[第2章「コンフィギュレーション インターフェイスの基本」](#)を参照してください。その他にもアプリケーションノート [XAPP973](#)『Virtex-5 FPGA での BPI PROM の間接プログラム』およびアプリケーションノート [XAPP974](#)『Spartan-3A FPGA で SPI シリアル Flash PROM の間接プログラム』を参照してください。

## 高速動作を重視したオプション

アプリケーションによっては、ロジックを短時間で動作可能にする必要があります。FPGA のコンフィギュレーション時間は、各モードや方法によって異なります。コンフィギュレーション時間とは、コンフィギュレーション自体にかかる時間と初期化に必要な時間を合計したもので、デバイスのサイズおよびコンフィギュレーションロジックの速度に依存します。たとえば、4ビットデータバスを使用して 33MHz でコンフィギュレーションする場合、Spartan-6 XC6SLX16 FPGA が 3.6Mb のコンフィギュレーションデータを読み込むのに約 28ms 要します。

- 同一クロック周波数で比較すると、パラレルコンフィギュレーションモードは複数ビットが1度にプログラムされるためシリアルモードよりも高速です。
- デジタイズチェーン接続した複数の FPGA をコンフィギュレーションするより、1つの FPGA をコンフィギュレーションするほうが短時間ですみます。複数の FPGA を使用したデザインでコンフィギュレーション時間を短縮するには、個々の FPGA を並行してコンフィギュレーションする必要があります。
- マスターモードでは、CCLK コンフィギュレーションクロック信号が FPGA 内部で生成されます。デフォルトでは、CCLK 周波数は低く設定されていますが、ConfigRate ビットストリームオプションを使用して高くできます。サポートされる最大 CCLK 周波数は、接続されている不揮発性メモリの読み出し速度の仕様に依存します。高速メモリを使用すると、より高速なコンフィギュレーションが可能です。FPGA の CCLK 出力周波数はプロセス、電圧、および温度によって変動します。コンフィギュレーションレートの最速値は、『Spartan-6 FPGA データシート』に記載されているように、CCLK 周波数の最小値によって異なります。ボード上に外部クロックがある場合、ザイリンクスプラットフォームフラッシュを使用したスレーブモードで FPGA をコンフィギュレーション可能です。

- ボード上に外部クロックがある場合、マスター モードのコンフィギュレーション中に外部クロック ソースを接続して使用する機能があります。ザイリンクス プラットフォーム フラッシュを使用しながら、外部クロック ソースを用いて FPGA をスレーブ モードでコンフィギュレーションすることも可能です。コンフィギュレーション中に外部クロック ソースを使用することで、マスターおよびスレーブの両モードでコンフィギュレーション時間を予測可能にできます。

## PCI リンクの有効化要件への準拠

PCI™ ローカル バス仕様のリビジョン 3.0 (「PCI 仕様」) では、さまざまな電源およびリセットの要件が定義されています。これらの要件によって、FPGA のインプリメンテーションで、長期にわたり信頼性およびボードの相互運用性を保つために解決しなければならない新たなデザインの課題が生じます。PCI アプリケーションでリンクがアクティブになるまでの時間を考慮し、指定時間内に FPGA のコンフィギュレーションを完了することが重要です。サードパーティのフラッシュ デバイスには、こうした時間的制約を満たしていないものが多くあります。

## 単独および複数のコンフィギュレーション イメージ

FPGA アプリケーションでは、システムに電源投入されるときにのみ FPGA が読み込まれるのが一般的です。

しかし、アプリケーションによっては、システム動作中に、ファンクションごとに異なる FPGA ビットストリームを使用して FPGA を複数回読み込む必要があるものがあります。たとえば、POST (Power On Self Test) をインプリメントするためのビットストリームを FPGA に読み込んだ後、最終アプリケーションで 2 つ目のビットストリームが読み込み可能です。多くのテスト装置アプリケーションでは、ハードウェア補助によるテストを実行するため異なるビットストリームが FPGA に読み込まれます。この方法を使用すると 1 つの小型 FPGA で大型の ASIC やゲート アレイ デバイスと同等の機能をインプリメントできます。

詳細は、第 7 章「リコンフィギュレーションおよび マルチブート」を参照してください。

## マルチブート/セーフ アップデート

高度なアプリケーションでは複数のビットストリーム イメージを格納できます。イメージの 1 つをユーザー アプリケーションでアップグレードし、リアルタイム システムのアップグレードを実行できます。ブート エラーが発生した場合は、初期イメージからシステムを回復させることも可能です。

## I/O 電圧要件

選択した FPGA コンフィギュレーション モードにより、FPGA アプリケーションに対する制約が生じます。特にコンフィギュレーション バンクに使用可能な I/O 電圧が制限されています。

たとえば、SPI や BPI モードでは、通常 3.3V デバイス (それよりも低い電圧は許容) であるサードパーティのフラッシュ メモリ コンポーネントが使用されます。つまり、メモリに接続されているバンクの I/O 電圧は入力電圧に準ずる必要があります。

## 不揮発性データの格納

一部の FPGA アプリケーションでは、データが外部不揮発性メモリに格納されます。Spartan-6 FPGA はこのようなアプリケーション用に有効な機能を提供します。

- Spartan-6 FPGA は外部シリアル (SPI) またはパラレル フラッシュ PROM (BPI) から直接コンフィギュレーションできます。
- フラッシュ PROM のアドレス、データ、制御ピンは、コンフィギュレーション中 FPGA から借用できます。コンフィギュレーション後、これらのピンは元に戻されて FPGA の読み出し/書き込み用に使用されます。
- FPGA コンフィギュレーション ビットストリームおよびアプリケーションの不揮発性データで同じ PROM を共有でき、全体的なシステム コストを削減します。

## 集積度の異なる FPGA への移行

Spartan-6 FPGA のパッケージのフットプリントおよびピン配置は、同一ファミリ内で集積度の異なるデバイス間の移行ができるよう設計されています。

FPGA アプリケーションではほかの不揮発性データをフラッシュ メモリに格納できるため、大型のストレージ デバイスが必要です。

異なる集積度のデバイス間のデザイン移行をサポートするには、ターゲット パッケージの最大デバイスに対応できるだけの十分なコンフィギュレーション メモリが必要です。たとえば、Spartan-6 XC6SLX9 デバイスを使用している場合、2.6Mb のコンフィギュレーション メモリが必要です。Spartan-6 XC6SLX16 デバイスの場合は、3.6Mb 必要です。

ダウンロードでコンフィギュレーションするアプリケーションでは、圧縮されていない FPGA ビットストリームの最大予測量に十分なメモリ容量が必要です。

FPGA が自動的にコンフィギュレーションするアプリケーションでは、PROM のフットプリントおよびそれに対応した FPGA コンフィギュレーション モードを使用して移行を簡単にすることができます。たとえば、ザイリンクス プラットフォーム フラッシュでは、XCFxxS シリアル ファミリを使用して 1Mb から 4Mb へ、XCFxxP パラレル ファミリを使用して 8Mb から 32Mb へ移行できます。1 つのアプリケーションでフラッシュが 2 つ使用されている場合は、プラットフォーム フラッシュのサブファミリごとに 2 つの異なるフットプリントを使用する必要があります。XCFxxP フラッシュ ファミリには 1.8V のコア電源電圧入力が必要で、XCFxxS には 3.3V が必要です。どちらのファミリでも 3.3V I/O が提供されます。

SPI シリアル フラッシュ ベンダーは幅広い移行範囲に対応できますが、マルチ パッケージ フットプリントが必要です。たとえば、Atmel DataFlash SPI シリアル フラッシュ ファミリでは、JEDEC および EIAJ 版の 8 ピン SOIC パッケージと 8 コネクタの CASON パッケージに対応する 1 つのフットプリントを使用し、1 ~ 64Mb の範囲で移行がサポートされています。Numonyx SPI シリアル フラッシュは、8 ピンおよび 16 ピンを組み合わせた SOIC フットプリントを使用するほかに、複数の SPI フラッシュ ベンダーのデバイスとの互換性もあります。

同様に、パラレル フラッシュでも、一般的な、複数のベンダーが提供するパッケージ フットプリントで広範囲にわたる集積度に対応できます。ここでは例を挙げて概説するにとどまりますが、詳細は使用しているフラッシュの仕様を参照してください。

## 製品寿命

アプリケーションの製品寿命を考慮する必要があります。一般的に、ベンダーから提供されているメモリはザイリンクスのプラットフォーム フラッシュ PROM と比較すると製品寿命が短くなっています。たとえば、5 年以上製造される産業向けアプリケーションを設計する場合、ザイリンクスのプラットフォーム フラッシュ PROM の製品寿命の方が長くなります。

製品寿命の短いものであれば、コスト面や種類の多さから見て、ベンダーが提供するメモリを使用した方が利点が多い場合もあります。

## 不正な複製から FPGA ビットストリームを保護する

プロセッサ コードと同様に、FPGA の機能を定義するビットストリームは電源投入時に FPGA に読み込まれます。結果として、他企業がこのビットストリームを入手してデザインを不正に複製してしまう可能性もあります。

プロセッサのように、FPGA ビットストリームとそこに埋め込まれている IP コアを保護する方法はいくつかあります。中でも、独自の Device DNA を用いる「認証」と呼ばれる手法が最も効果的です。詳細は、[第 5 章「コンフィギュレーションの詳細」](#)で説明します。また、XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスには、オンチップの AES (Advanced Encryption Standard) 復号化ロジックがあり、デザインを高度なセキュリティで保護します。

## 同じコンフィギュレーション ビットストリームを複数の FPGA に読み込む

通常、システム内では 1 つの FPGA に 1 つのコンフィギュレーション ビットストリームがあります。コンフィギュレーション デイジー チェーンを利用すると、1 つのコンフィギュレーション PROM に複数の異なる FPGA ビットストリーム イメージを格納できます。しかし、アプリケーションにあるすべての FPGA のパーツ番号とビットストリームが同じ場合、ビットストリーム イメージは 1 つだけでかまいません。また、同じビットストリームを、複数の同一 FPGA に読み込むことができるギャング コンフィギュレーションもあります。

## コンフィギュレーションを決定する要素

システムに最適なコンフィギュレーション ソリューションを決定する要素は数多く存在し、さまざまな詳細を考慮する必要があります。適切なコンフィギュレーションを選択しておく、後で対処しなくてはならない問題が少なくなります。

特に、コンフィギュレーション専用ピンとコンフィギュレーション後に再利用可能なピンの違いを理解しておくことが必要です。詳細は、コンフィギュレーションに関するセクションを参照してください。

また、データ ファイル形式およびビットストリームのサイズも考慮する必要があります。ビットストリームのサイズはデバイス サイズによって異なり、ビットストリームの生成にも複数の形式があります。

FPGA はコンフィギュレーション中、内部メモリの初期化から I/O の有効化まで、ある一定のシーケンスを実行します。このプロセスはコンフィギュレーション シーケンスと呼ばれます。電源投入から FPGA コンフィギュレーションの完了とスタートアップまでのタイミングを理解するには、このシーケンスとサブシーケンスの理解が必要です。

Spartan-6 LX75、LX75T、LX100、LX100T、LX150、および LX150T の FPGA には、AES 暗号化などの高度なセキュリティ保護機能があります。この機能は、ビットストリームの保護に非常に有益です。

詳細は、[第 5 章「コンフィギュレーションの詳細」](#)を参照してください。



## コンフィギュレーション インターフェイスの基本

この章では、Spartan®-6 FPGA デバイスで最も頻繁に使用されるコンフィギュレーション ソリューションについて説明します。いくつかの手法を挙げ、適切な接続、終端、信号定義、基本的なタイミングについて説明します。第 9 章「アドバンス コンフィギュレーション インターフェイス」では、エラー回復の詳細や、ここで簡単にまとめたものをさらに詳細に説明しています。

Spartan-6 デバイスは、Extended Spartan-3A ファミリでサポートされているコンフィギュレーション モードにすべて対応しています。ただし、Extended Spartan-3A ファミリには、コンフィギュレーション モードを定義するモード ピンが M[2:0] と 3 つあるのに対して、Spartan-6 デバイスでは M[1:0] の 2 つのみです。モードピンの説明は、表 2-1 を参照してください。インターフェイスのタイミング情報の詳細は、データシート [DS162](#)『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』を参照してください。

表 2-1 : Spartan-6 FPGA のコンフィギュレーション モード

コンフィギュレーション	M[1:0]	バス幅	CCLK の方向
マスター シリアル/SPI	01	1、2、4 <sup>(1)</sup>	出力
マスター SelectMAP/BPI <sup>(2)</sup>	00	8、16	出力
JTAG <sup>(3)</sup>	xx	1	入力 (TCK)
スレーブ SelectMAP <sup>(2)</sup>	10	8、16	入力
スレーブ シリアル <sup>(4)</sup>	11	1	入力

注記：

- デュアルおよびクアッド SPI モードを使用します。
- パラレルコンフィギュレーション モードのバスは、コンフィギュレーション ロジックによって自動的に検出されます。
- Spartan-6 デバイスには、モード ピンの設定にかかわらず FPGA で常に使用可能な、4 線式の JTAG (IEEE 1149.1 規格) ポートもあります。
- デフォルト設定は、モード ピンの内部プルアップ終端によります。

## JTAG インターフェイス

JTAG 向けの特定モードはありませんが、デバイスへの電源投入後に使用可能なコンフィギュレーション インターフェイスとして JTAG インターフェイスがあります。詳細は、第 3 章「バウンダリ スキャンおよび JTAG コンフィギュレーション」を参照してください。

## シリアル コンフィギュレーション インターフェイス

シリアル コンフィギュレーション モードでは、CCLK の 1 サイクルで 1 コンフィギュレーション ビットを読み込み、FPGA をコンフィギュレーションします。

- マスター シリアル モードでは、CCLK は出力です。
- スレーブ シリアル モードでは、CCLK は入力です。

シリアル コンフィギュレーションのシミュレーション モデルが利用できます。詳細は、『合成/シミュレーション デザイン ガイド』([UG626](#)) を参照してください。

[図 2-1](#) に、Spartan-6 FPGA の基本的なシリアル コンフィギュレーション インターフェイスを示します。

シリアル モードで FPGA をコンフィギュレーションするには次の 4 つのモードがあります。

- マスター シリアル コンフィギュレーション
  - 一般的な構成では、プラットフォーム フラッシュ (XCFxxP) などの PROM を使用します。
- スレーブ シリアル コンフィギュレーション
  - 一般的な構成では、データおよびクロックを提供するプロセッサを使用します。
- シリアル デイジーチェーン コンフィギュレーション
  - PROM またはプロセッサから、複数の FPGA を順に異なるイメージでコンフィギュレーションします ([第 9 章「アドバンス コンフィギュレーション インターフェイス」](#) 参照)。
- ギャング シリアル コンフィギュレーション
  - PROM またはプロセッサから、複数の FPGA を並行して同じイメージでコンフィギュレーションします ([第 9 章「アドバンス コンフィギュレーション インターフェイス」](#) 参照)。

この章では、マスターおよびスレーブ シリアル コンフィギュレーションについて説明しています。デイジー チェーンおよびギャング コンフィギュレーションについては[第 9 章「アドバンス コンフィギュレーション インターフェイス」](#)を参照してください。

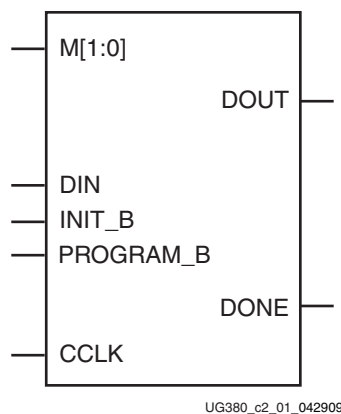


図 2-1 : Spartan-6 FPGA のシリアル コンフィギュレーション インターフェイス



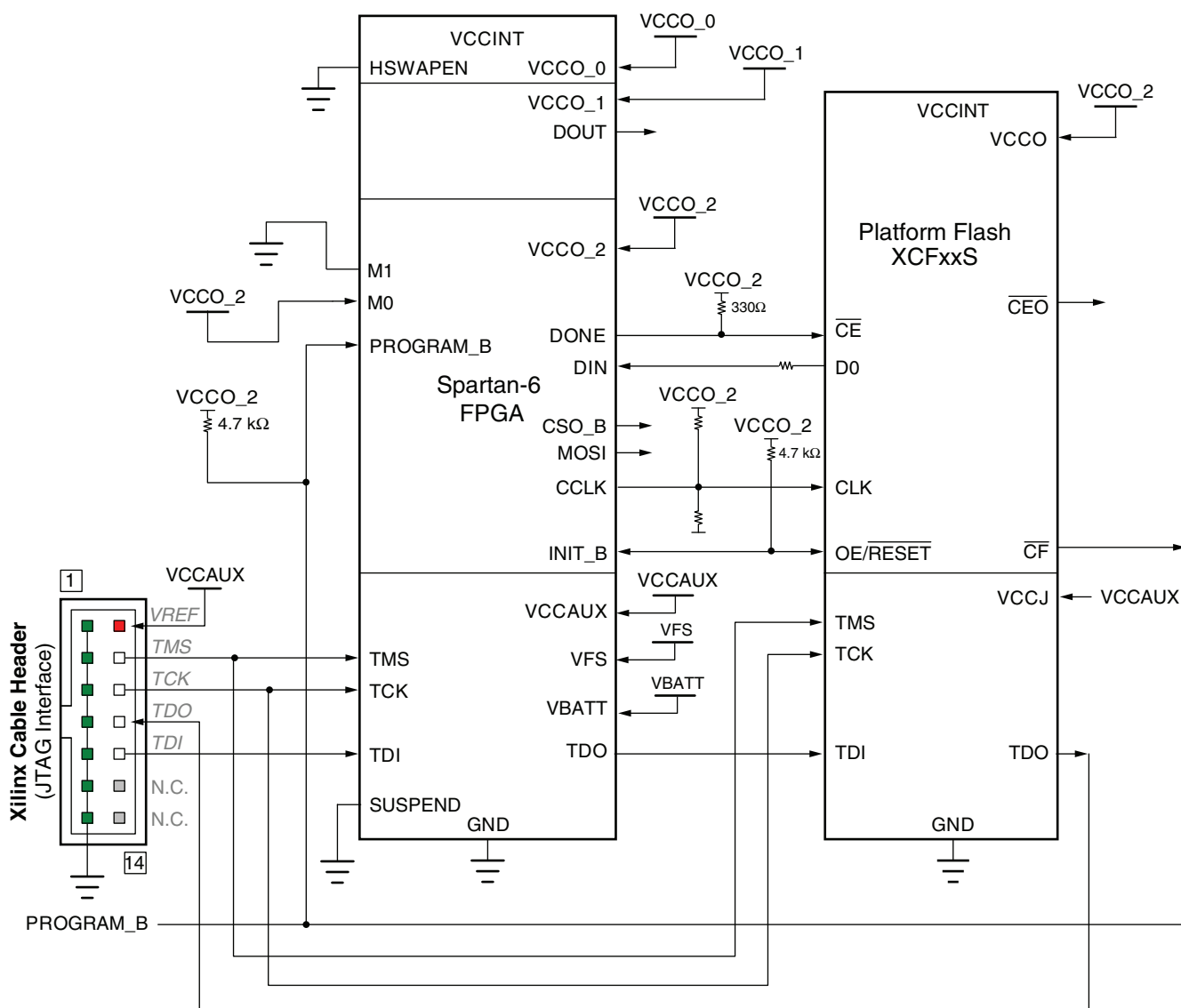
表 2-2 で、シリアル コンフィギュレーション インターフェイスの各ピンについて説明します。

表 2-2 : Spartan-6 FPGA のシリアル コンフィギュレーション インターフェイスで使用するピン

ピン名	タイプ	専用/多目的	説明
M[1:0]	入力	多目的	コンフィギュレーション モードを決定するモード ピンです (表 2-1 参照)。
CCLK	入力または出力	多目的	JTAG を除く、すべてのコンフィギュレーション モードでのコンフィギュレーション クロック ソースです (64 ページの「設計に関する考察事項」参照)
DIN	入力	多目的	シリアル コンフィギュレーション データ入力で、CCLK の立ち上がりエッジに同期します。
DOUT	出力	多目的	デジizer チェーンの下位デバイスに対するシリアル データ出力です。CCLK の立ち下がりエッジでデータを供給します。
DONE	双方向、オープンドレイン、またはアクティブ	専用	コンフィギュレーションの完了を示すアクティブ High の信号です。 0 = FPGA コンフィギュレーション未完了 1 = FPGA のコンフィギュレーション完了 ソフトウェア設定の詳細は、『コマンド ライン ツール ユーザー ガイド』(UG628) の「BitGen」を参照してください。
INIT_B	入力または出力、オープンドレイン	多目的	モード ピンのサンプル前は入力となり、Low に保持することでコンフィギュレーションを遅延させることができます。 モード ピンのサンプル後はオープン ドレインのアクティブ Low 入力となり、コンフィギュレーション中の CRC エラーの有無を示します。 0 = CRC エラー 1 = CRC エラーなし SEU 検出が有効になっている場合、リードバック CRC エラーが検出されると、オプションで INIT_B が Low に駆動されます。
PROGRAM_B	入力	専用	アクティブ Low の非同期フルチップ リセットです。

## マスター シリアル

図 2-2 に示すように、マスター シリアル モードでは、ザイリンクスのプラットフォーム フラッシュ PROM から FPGA をコンフィギュレーションできます。



Refer to the Notes following this figure for related information.

UG380\_c2\_02\_070210

図 2-2：マスター シリアル モードのコンフィギュレーション

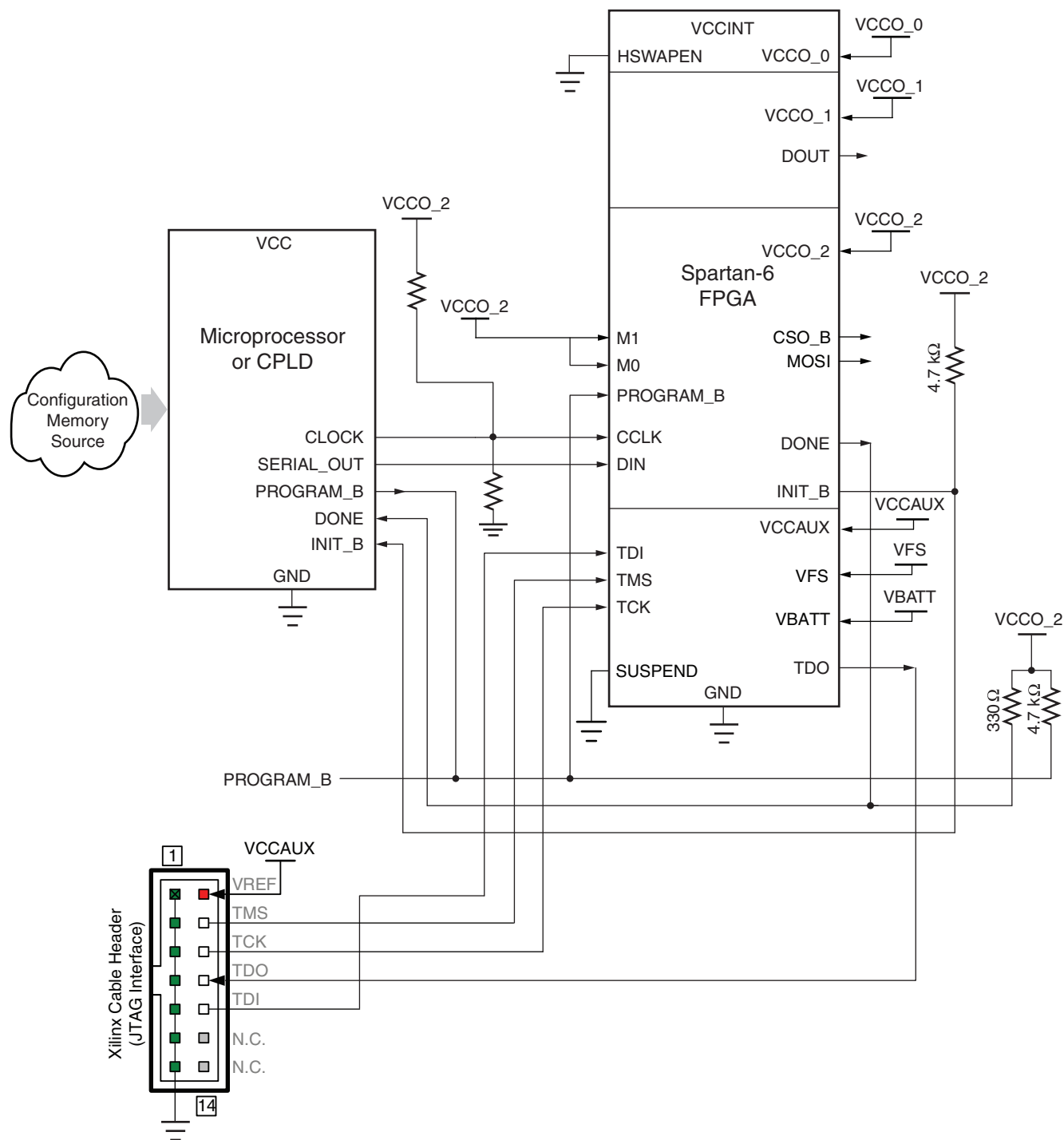
図 2-2 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、74 ページの表 5-2 を参照してください。
2. デイジー チェーン コンフィギュレーション モードでは、DOUT を下位 FPGA の DIN に接続してください。
3. CCLK ネットにはテブナン並列終端が必要です。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。

4. マスターシリアルおよびマスター SPI はどちらも同じモード ピンから有効になります。このため SPI 制御ピンである CSO\_B および MOSI はコンフィギュレーション中にトグルします。
5. Spartan-6 FPGA の VCCO\_2 およびプラットフォーム フラッシュ PROM の V<sub>CCO</sub> は同じ電圧である必要があります。
6. DONE ピンは、デフォルトでは内部プルアップ抵抗を持つオープン ドレイン出力です。外部プルアップ抵抗を追加することを推奨します。DONE ピンには、BitGen の **-g DriveDone** オプションを使用して有効にできる、プログラム可能でアクティブなドライバーがあります。
7. INIT\_B ピンは双方向オープン ドレイン ピンです。外部プルアップ抵抗の使用を推奨します。
8. BitGen スタートアップ クロック設定では、CCLK をシリアル コンフィギュレーションに設定する必要があります。これはソフトウェアによりデフォルトで設定されます。詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) を参照してください。
9. この図の PROM は、1 つまたは複数のザイリンクス PROM の場合を示しています。複数のザイリンクス PROM をカスケード接続することで、データ格納範囲全体が拡大できます。詳細は、『Platform Flash PROM ユーザー ガイド』([UG161](#)) を参照してください。
10. BIT ファイルは、PROM へ格納する前に PROM ファイルに変換する必要があります。iMPACT を使用して必要なファイルを生成する場合は、[79 ページ](#)の「[PROM ファイルの生成](#)」を参照してください。
11. 一部のザイリンクス PROM では、リセット ピンの極性がプログラム可能です。この設定で PROM を使用する場合は、**RESET** をアクティブ Low に設定してください。
12. マスター シリアル モード コンフィギュレーションは、プラットフォーム フラッシュ XCFS および XCFP PROM にのみ使用できます。
13. CSI\_B や RDWR\_B などの未使用の専用コンフィギュレーション ピンは、このモードではどのコンフィギュレーション ロジックにも接続されないため、フロートにしておくか GND に接続できます。CSI\_B および RDWR\_B は多目的ピンです。
14. V<sub>FS</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、[94 ページ](#)の「[eFUSE](#)」を参照してください。
15. V<sub>BATT</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供される、AES キーの格納用の電源です。AES 暗号が未使用の場合、V<sub>BATT</sub> は V<sub>CCAUX</sub> または GND に接続するか、未接続のままでもかまいません。
16. VCCO\_2 が 1.8V の場合、V<sub>CCAUX</sub> は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、V<sub>CCAUX</sub> は 2.5V または 3.3V にできます。
17. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

## スレーブ シリアル コンフィギュレーション

スレーブ シリアル コンフィギュレーションは、通常、シリアル デイジー チェーン接続した複数デバイスをコンフィギュレーションするとき、または外部マイクロプロセッサ/CPLD から単独デバイスをコンフィギュレーションするとき使用します ([図 2-3](#) 参照)。スレーブ シリアル コンフィギュレーションで考慮すべき点は、CCLK の方向を除いて、マスター シリアル コンフィギュレーションと同様です。CCLK はデータを提供する外部クロック ソースから駆動する必要があります ([29 ページ](#)の「[シリアル コンフィギュレーション データ タイミング](#)」参照)。



Refer to the Notes following this figure for related information.

UG380\_c2\_03\_071910

図 2-3：スレーブ シリアル モードのコンフィギュレーション

図 2-3 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、74 ページの表 5-2 を参照してください。

2. デイジー チェーン コンフィギュレーション モードの場合、DOUT を下位 FPGA の DIN に接続してください。
3. CCLK ネットにはテブナン並列終端が必要です。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
4. DONE ピンは、デフォルトでは、内部プルアップ抵抗のあるオープンドレイン出力です。外部プルアップを追加することを推奨します。DONE ピンには、BitGen の **-g DriveDone** オプションを使用して有効にできる、プログラム可能でアクティブなドライバーがあります。
5. INIT\_B ピンは双方向オープンドレイン ピンです。外部プルアップ抵抗の使用を推奨します。
6. SPI 制御ピンである CSO\_B および MOSI はシリアル コンフィギュレーション中にトグルします。
7.  $V_{FS}$  は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、94 ページの「eFUSE」を参照してください。
8.  $V_{BATT}$  は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供される、AES キーの格納用の電源です。AES 暗号が未使用の場合、 $V_{BATT}$  は GND に接続するか、未接続のままでもかまいません。
9. VCCO\_2 が 1.8V の場合、 $V_{CCAUX}$  は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、 $V_{CCAUX}$  は 2.5V または 3.3V にできます。
10. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

## シリアル コンフィギュレーション データ タイミング

図 2-4 に、Spartan-6 デバイスのスレーブ シリアルおよびマスター シリアル モードにおけるコンフィギュレーション データのクロック シーケンスを示します。

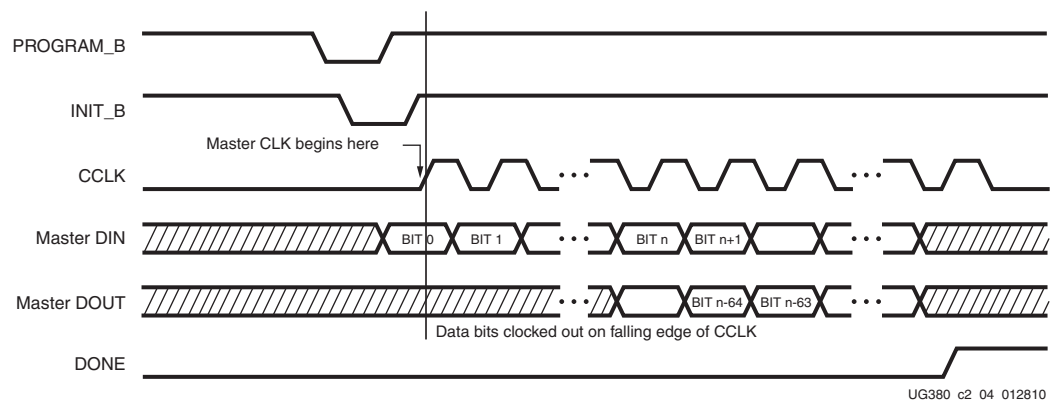


図 2-4 : シリアル コンフィギュレーションのクロック シーケンス

次は、図 2-4 について説明します。

1. ビット 0 は、最初のバイトの MSB を示します。たとえば、最初のバイトが 0xAA (1010\_1010) の場合、ビット 0 = 1、ビット 1 = 0、ビット 2 = 1 となります。
2. マスター コンフィギュレーション モードの場合、CCLK は矢印が示すように、モード ピンのサンプリングが完了するまで遷移しません。
3. スレーブ シリアル モードの場合、CCLK はフリーランニングになります。

## SelectMAP コンフィギュレーション インターフェイス

SelectMAP コンフィギュレーション インターフェイス (図 2-5 参照) には、Spartan-6 デバイスのコンフィギュレーション ロジックと接続するための 8 ビットまたは 16 ビットの双方向バスがあり、コンフィギュレーションおよびリードバックの両方に使用できます。詳細は、第 6 章「リードバックおよびコンフィギュレーションの検証」を参照してください。SelectMAP のバス幅は自動的に検出されます (78 ページの「同期ワード/バス幅の自動検出」参照)。SelectMAP コンフィギュレーションのシミュレーション モデルが利用できます。詳細は、『合成/シミュレーション デザイン ガイド』(UG626) を参照してください。

マスター SelectMAP モードでは CCLK は出力となります。スレーブ SelectMAP モードでは CCLK は入力となり、ボード上で外部ソースからまたは専用 GCLK ピンから供給されます。SelectMAP バスを使用し、1 つまたは複数の Spartan-6 デバイスをシリアルまたはパラレルにコンフィギュレーションできます。

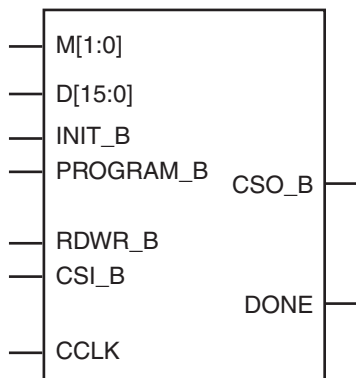
SelectMAP モードでの FPGA コンフィギュレーションには、次の方法があります。

- 単一デバイス マスター SelectMAP
- 単一デバイス スレーブ SelectMAP
  - 一般的な構成では、データおよびクロックを提供するプロセッサを使用します。
- 複数デバイス デイジー チェーン SelectMAP バス
  - PROM またはプロセッサから、複数の FPGA を順に異なるイメージでコンフィギュレーションします (第 9 章「アドバンス コンフィギュレーション インターフェイス」参照)。
- 複数デバイスのギャング SelectMAP
  - PROM またはプロセッサから、複数の FPGA を並行して同じイメージでコンフィギュレーションします (第 9 章「アドバンス コンフィギュレーション インターフェイス」参照)。

SelectMAP には次のような注意点があります。

- SelectMAP は、XC6SLX4 デバイス、あるいは TQG144 または CPG196 パッケージを使用するデバイスで使用できません。
- RDWR\_B は両用ピンです。これは、バンク 2 で  $V_{REF}$  にできますが、SelectMAP コンフィギュレーション モードでは  $V_{REF}$  として使用できません。

この章では、マスター SelectMAP およびスレーブ SelectMAP について説明しています。デイジーチェーンおよびギャング コンフィギュレーションの詳細は、第 9 章「アドバンス コンフィギュレーション インターフェイス」を参照してください。



UG380\_c2\_05\_042909

図 2-5 : Spartan-6 FPGA の SelectMAP コンフィギュレーション インターフェイス

表 2-3 で、SelectMAP コンフィギュレーション インターフェイスの各ピンについて説明します。

表 2-3 : Spartan-6 FPGA の SelectMAP コンフィギュレーション インターフェイスで使用するピン

ピン名	タイプ	専用/多目的	説明
M[1:0]	入力	多目的	コンフィギュレーション モードを決定するモードピンです (23 ページの表 2-1 参照)。
CCLK	入力および出力	多目的	JTAG を除く、すべてのコンフィギュレーションモードでのコンフィギュレーション クロック ソースです (57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」参照)。
D[15:0]	トライステート双方向	多目的	コンフィギュレーションおよびリードバック バスで、クロックは CCLK の立ち上がりエッジです (81 ページの「パラレル バスのビット順」参照)。
DONE	双方向、オープンドレインまたはアクティブ	専用	コンフィギュレーションの完了を示すアクティブ High の信号です。 0 = FPGA コンフィギュレーション未完了 1 = FPGA コンフィギュレーション完了
INIT_B	入力または出力、オープンドレイン	多目的	モード ピンのサンプル前は入力となり、Low に保持することでコンフィギュレーションを遅延させることができます。 モード ピンのサンプル後はオープンドレインのアクティブ Low 入力となり、コンフィギュレーション中の CRC エラーの有無を示します。 0 = CRC エラー 1 = CRC エラーなし SEU 検出機能が有効になっている場合、リードバック CRC エラーが検出されると、オプションで INIT_B が Low に駆動されます。
PROGRAM_B	入力	専用	アクティブ Low の非同期フルチップ リセットです。

表 2-3 : Spartan-6 FPGA の SelectMAP コンフィギュレーション インターフェイスで使用するピン

ピン名	タイプ	専用/多目的	説明
CSI_B	入力	多目的	アクティブ Low のチップ セレクトであり、SelectMAP データ バスを有効にします (36 ページの「 <a href="#">SelectMAP データ読み込み</a> 」参照)。 0 = SelectMAP データ バスは有効 1 = SelectMAP データ バスは無効
RDWR_B	入力	多目的	D[x:0] データ バスの方向を決定します (36 ページの「 <a href="#">SelectMAP データ読み込み</a> 」参照)。 0 = 入力 1 = 出力  CSI_B がディアサートされているときにのみ変更でき、それ以外の場合に変更すると ABORT が生じます (155 ページの「 <a href="#">SelectMAP の ABORT</a> 」参照)。V <sub>REF</sub> ピンとして使用できますが、SelectMAP コンフィギュレーション モードが使用できなくなります。
CSO_B	出力	多目的	パラレル デイジー チェーンのアクティブ Low のチップ セレクト出力です。FPGA が 1 つのアプリケーションでは使用されません。
BUSY	出力	多目的	リードバック中に使用します。コンフィギュレーション中はトグル可能です。

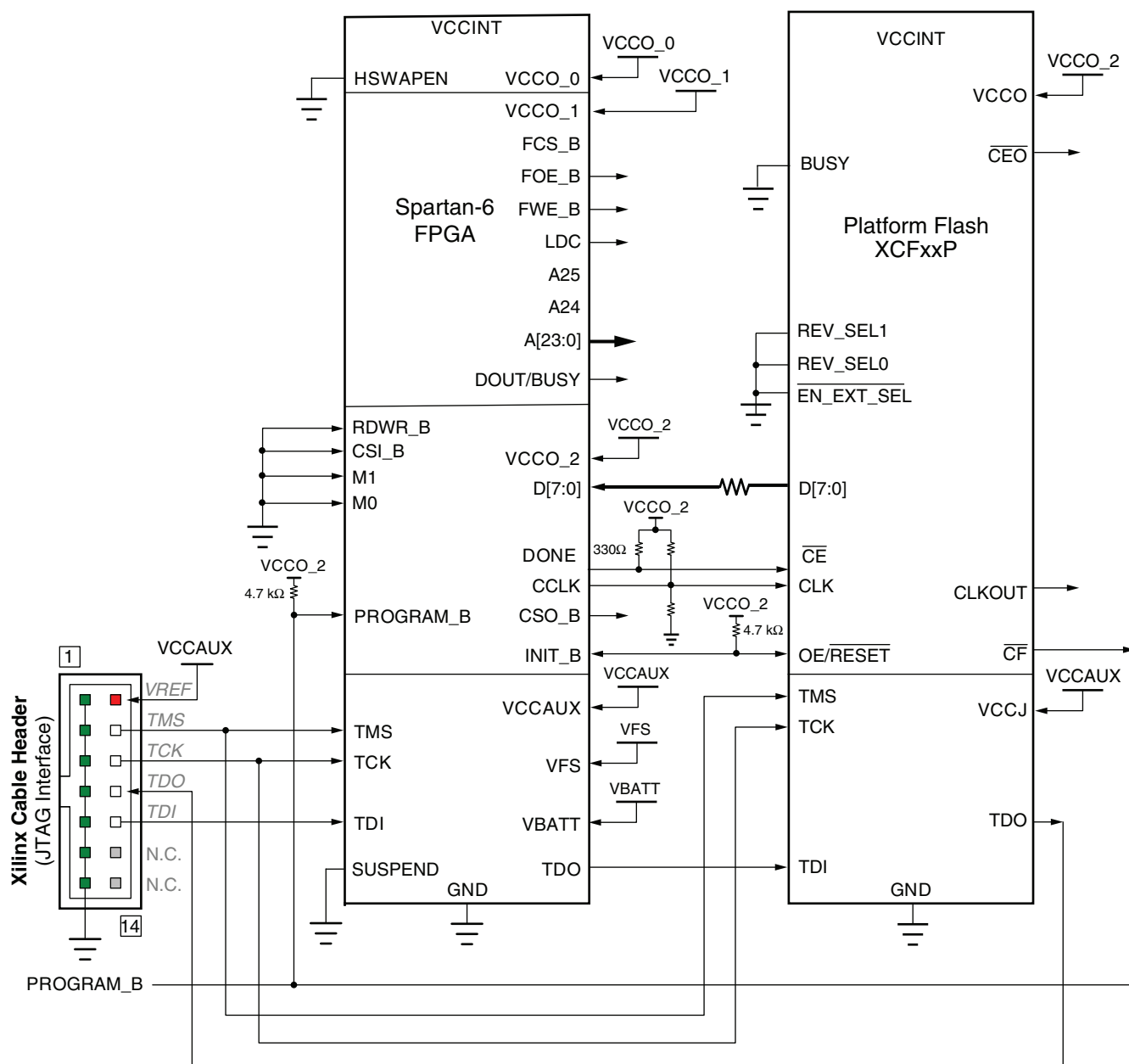
## 単一デバイスの SelectMAP コンフィギュレーション

このセクションでは、SelectMAP モードで 1 つのデバイスをコンフィギュレーションする方法について説明します。このコンフィギュレーションでは、FPGA をプラットフォーム フラッシュ PROM、あるいはマイクロプロセッサまたは CPLD に接続します。

### プラットフォーム フラッシュ PROM の SelectMAP コンフィギュレーション

SelectMAP モードで 1 つのデバイスをコンフィギュレーションするには、[図 2-6](#) に示すように、コンフィギュレーション PROM に直接接続するのが最も単純な方法です。この方法では、デバイスはマスター SelectMAP モードに設定され、連続的なデータの読み込みに対応できるように RDWR\_B および CS\_B ピンが GND に接続されています (36 ページの「[SelectMAP データ読み込み](#)」参照)。





Refer to the Notes following this figure for related information.

UG380 c2 06 071910

図 2-6：単一デバイスのマスター SelectMAP コンフィギュレーション

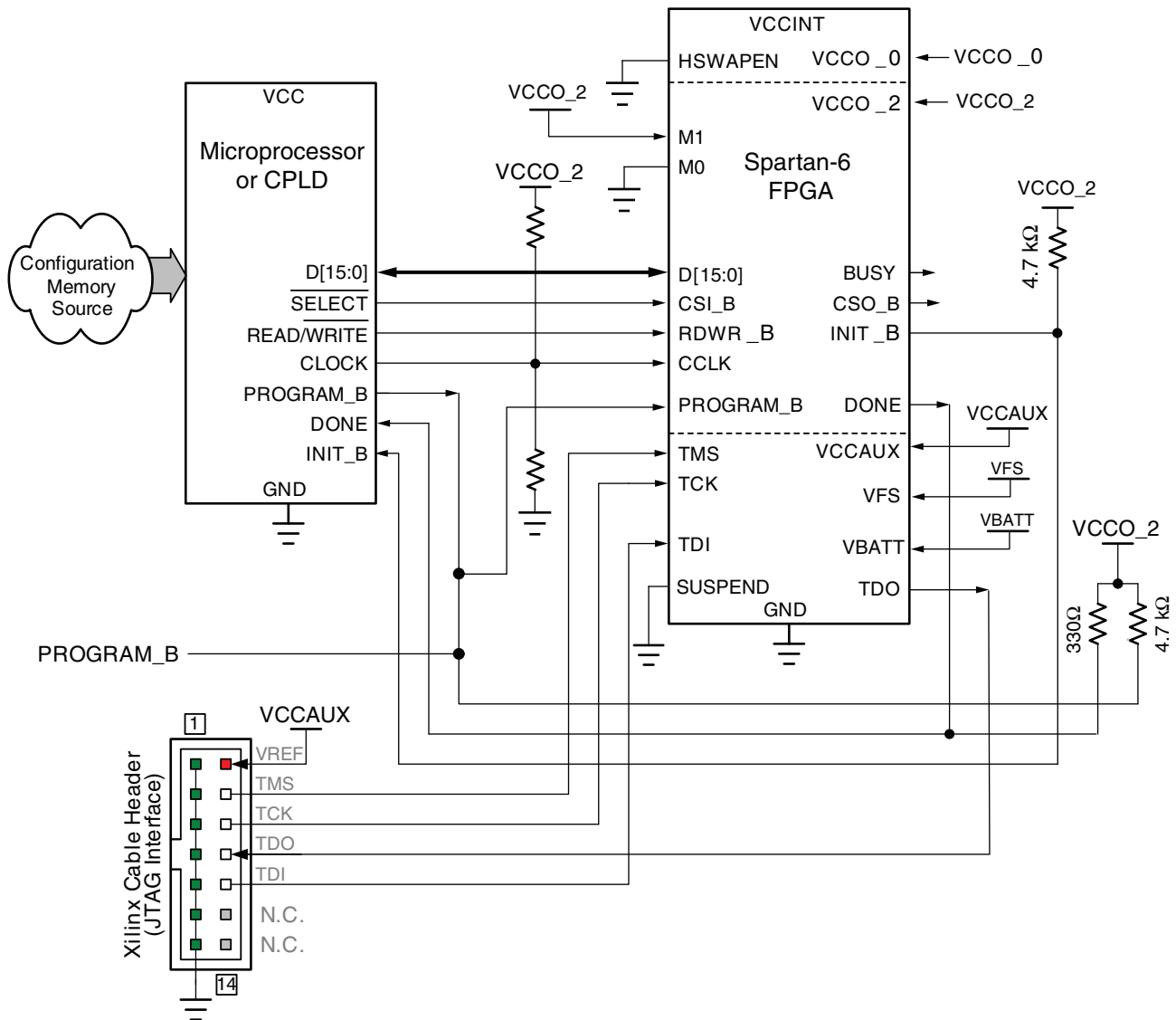
図 2-6 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、[74 ページの表 5-2](#) を参照してください。
2. TCCLK ネットにはテブナン並列終端が必要です。詳細は、[57 ページの「コンフィギュレーション クロック \(CCLK\) のボード レイアウト」](#) を参照してください。
3. DONE ピンは、デフォルトでは、内部プルアップを持つオープン ドレイン出力です。外部プルアップを追加することを推奨します。DONE ピンには、BitGen の **-g DriveDone** オプションを使用して有効にできる、プログラム可能でアクティブなドライバーがあります。

4. フラッシュから FPGA へのデータパスには、オーバーシュートを最小限に抑えるために直列抵抗の使用を検討してください。適切な抵抗値はシミュレーションで求めることができます。
5. INIT\_B ピンは、双方向オープンドレイン ピンです。外部プルアップ抵抗の使用を推奨します。
6. BitGen のスタートアップ クロック設定では、CCLK を SelectMAP コンフィギュレーションに設定する必要があります。
7. この図の PROM は、1 つまたは複数のザイリンクス PROM の場合を示しています。複数のザイリンクス PROM をカスケード接続することで、データの格納範囲全体を拡大できます。
8. BIT ファイルは、PROM へ格納する前に PROM ファイルに変換する必要があります。詳細は、[79 ページの「PROM ファイルの生成」](#)を参照してください。
9. 一部のザイリンクス PROM では、リセット ピンの極性がプログラム可能です。この設定で PROM を使用する場合、RESET をアクティブ Low に設定してください。
10. ザイリンクスの PROM は、パラレル モードに設定する必要があります。このオプションは一部のデバイスでのみ使用可能です。
11. SelectMAP モードで、ザイリンクスのコンフィギュレーション PROM から Spartan-6 デバイスをコンフィギュレーションする場合、RDWR\_B および CS\_B 信号は Low に接続可能です ([36 ページの「SelectMAP データ読み込み」](#)参照)。
12. マスター SelectMAP コンフィギュレーションの場合、データ バス幅は x8 または x16 が可能です。XCFxxP のデータ幅の最大値は x8 です。
13. プラットフォーム フラッシュ PROM SelectMAP コンフィギュレーションは、プラットフォーム フラッシュ XCFP PROM 専用です。プラットフォーム フラッシュ XCFS PROM ではシリアル コンフィギュレーション モードのみがサポートされています。
14. BUSY、FOE\_B、FCS\_B、および FWE\_B ピンを含むアドレス バス A[25:0] はコンフィギュレーション中にトグルします。これらの多目的ピンの動作は、コンフィギュレーション中にシステムで処理可能にしておく必要があります。
15. Spartan-6 FPGA の VCCO\_2 およびプラットフォーム フラッシュ PROM の V<sub>CCO</sub> は、同じ電圧である必要があります。
16. V<sub>FS</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、[94 ページの「eFUSE」](#)を参照してください。
17. V<sub>BATT</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、これらの多目的ピンの動作は、コンフィギュレーション中にシステムで処理可能にしておく必要があります。AES 暗号が未使用の場合、V<sub>BATT</sub> は GND に接続するか、未接続のままでもかまいません。
18. VCCO\_2 が 1.8V の場合、V<sub>CCAUX</sub> は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、V<sub>CCAUX</sub> は 2.5V または 3.3V にできます。
19. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

## マイクロプロセッサによる SelectMAP コンフィギュレーション

マイクロプロセッサまたは CPLD を使用し、1 つの Spartan-6 デバイスをコンフィギュレーションするカスタム アプリケーションでは、マスター SelectMAP モード (FPGA からの CCLK を使用) またはスレーブ SelectMAP モードを利用できますが (図 2-7 参照)、スレーブ SelectMAP モードの使用が推奨されます。スレーブ SelectMAP モードが推奨されます。マイクロプロセッサからのコンフィギュレーションの詳細は、アプリケーション ノート [XAPP502](#) 『マイクロプロセッサを使用した、スレーブ シリアルまたは SelectMAP モードでの FPGA コンフィギュレーション』() を参照してください。



Refer to the Notes following this figure for related information.

UG380\_c2\_07\_062910

図 2-7: 単一デバイスのスレーブ SelectMAP コンフィギュレーション (マイクロプロセッサおよび CPLD を使用)

図 2-7 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、74 ページの表 5-2 を参照してください。

2. DOUT/BUSY はコンフィギュレーションまたはリードバック中に駆動できる出力です。
3. CCLK の終端については、[57 ページの「コンフィギュレーション クロック \(CCLK\) のボードレイアウト」](#)を参照してください。
4. この回路図は、アプリケーション ノート [XAPP502](#) 『マイクロプロセッサを使用した、スレーブ シリアルまたは SelectMAP モードでの FPGA コンフィギュレーション』からの抜粋です。多数ある可能なインプリメンテーションの 1 つです。
5. DONE ピンは、デフォルトでは内部プルアップ抵抗を持つオープン ドレイン出力です。外部プルアップを追加することを推奨します。DONE ピンには、BitGen の **-g DriveDone** オプションを使用して有効にできる、プログラム可能でアクティブなドライバーがあります。
6. INIT\_B ピンは双方向オープン ドレイン ピンです。外部プルアップ抵抗の使用を推奨します。
7. BitGen のスタートアップ クロック設定では、CCLK を SelectMAP コンフィギュレーションに設定する必要があります。
8. 1 つの FPGA のみをコンフィギュレーションする場合、かつリードバックが不要な場合は、CSI\_B および RDWR\_B 信号を GND に接続できます。
9. スレーブ SelectMAP コンフィギュレーションの場合、データ バス幅は x8 または x16 が可能です。
10. V<sub>FS</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、[94 ページの「eFUSE」](#)を参照してください。
11. V<sub>BATT</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供される、AES キーの格納用の電源です。AES 暗号が未使用の場合、V<sub>BATT</sub> は GND に接続するか、未接続のままでもかまいません。
12. VCCO\_2 が 1.8V の場合、V<sub>CCAUX</sub> は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、V<sub>CCAUX</sub> は 2.5V または 3.3V にできます。
13. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

## SelectMAP データ読み込み

SelectMAP インターフェイスでは、連続的または非連続的にデータを読み込むことが可能です。データ読み込みは、CSI\_B、RDWR\_B、および CCLK 信号で制御します。

### CSI\_B

チップ セレクト入力 (CSI\_B) は、同期ワードが検出された後に SelectMAP バスを有効にします。同期ワード検出前は、D[0:n] ピンが CSI\_B のステートに関係なく各クロック エッジでサンプルされます。CSI\_B は、同期ワードの途中でディアサートしないようにしてください。CSI\_B が High のとき、Spartan-6 デバイスは SelectMAP インターフェイスを無視し、入力データを保持したり、出力を駆動することはありません。D[0:n] はハイ インピーダンスになり、RDWR\_B は無視されます。

- CSI\_B = 0 の場合、デバイスの SelectMAP は有効
- CSI\_B = 1 の場合、デバイスの SelectMAP インターフェイスは無効

複数デバイスの SelectMAP コンフィギュレーションについては、[第 9 章「アドバンス コンフィギュレーション インターフェイス」](#)を参照してください。

SelectMAP インターフェイスでコンフィギュレーションされるデバイスが 1 つのみ、かつリードバックが不要な場合、またはギャング SelectMAP コンフィギュレーションが使用される場合は、[第](#)

9 章「アドバンス コンフィギュレーション インターフェイス」に示すように、CSI\_B 信号を GND に接続できます。

## RDWR\_B

RDWR\_B は Spartan-6 デバイスの入力であり、データ ピンが入力または出力かを選択します。

- RDWR\_B = 0 の場合、データ ピンは入力 (FPGA に書き込む)
- RDWR\_B = 1 の場合、データ ピンは出力 (FPGA から読み出す)

コンフィギュレーションの際は、RDWR\_B を書き込み制御に設定 (RDWR\_B = 0) しておく必要があります。リードバックを実行するには、CSI\_B がディアサートされている間に RDWR\_B を読み出し制御に設定 (RDWR\_B = 1) しておく必要があります。詳細は、第 6 章「リードバックおよびコンフィギュレーションの検証」を参照してください。リードバックが不要な場合、RDWR\_B は GND に接続するか、SelectMAP ABORT 時のデバッグに使用できます。

CSI\_B がディアサートされている場合、RDWR\_B 信号は無視されます。データ ピンの 3 つのステートの読み出し/書き込み制御は非同期です。CSI\_B がアサートされている間、RDWR\_B を読み出し (RDWR\_B = 1、リードバック) に設定すると、FPGA は CCLK と無関係に SelectMAP データをアクティブに駆動します。CSI\_B がアサートされている状態で RDWR\_B を変更すると、FPGA が非同期的に違反を検出して BUSY 信号を駆動するため、ABORT となります。ステータスレジスタは次の CCLK の立ち上がりエッジまで更新されません (155 ページの「SelectMAP の ABORT」参照)。

## CCLK

SelectMAP データ バスのすべての動作は、CCLK に同期しています。RDWR\_B が書き込み制御に設定されている場合 (RDWR\_B = 0、コンフィギュレーション)、FPGA は SelectMAP データ ピンを CCLK の立ち上がりエッジでサンプルします。RDWR\_B が読み出し制御に設定されている場合 (RDWR\_B = 1、リードバック)、FPGA は SelectMAP データ ピンを CCLK の立ち上がりエッジで更新します。

スレーブ SelectMAP モードでは、CCLK を停止することによってコンフィギュレーションを中断できます (39 ページの「SelectMAP データの不連続読み込み」参照)。

## SelectMAP データの連続読み込み

データの連続読み出しは、コンフィギュレーション コントローラーからコンフィギュレーション データ ストリームを中断なく供給可能なアプリケーションで実行されます。電源投入後、コンフィギュレーション コントローラーは、RDWR\_B 信号を書き込み制御 (RDWR\_B = 0) に設定し、CSI\_B 信号をアサート (CSI\_B = 0) して、デバイスの BUSY を Low 駆動させます (この遷移は非同期)。CSI\_B がアサートされる前に RDWR\_B を Low 駆動する必要があり、そうしなければ ABORT が発生します。(155 ページの「SelectMAP の ABORT」参照)。

デバイスは、CCLK の次の立ち上がりエッジでデータ ピンのサンプリングを開始します。コンフィギュレーションでは、バス幅が決定するまで D[0:15] ピンのみがサンプルされます。詳細は、78 ページの「同期ワード/バス幅の自動検出」を参照してください。バス幅が決定すると、同期幅を検出するために適切な幅のデータ バスがサンプルされます。同期ワードがデバイスに伝搬された後、コンフィギュレーションが開始します。

コンフィギュレーション ビットストリームの読み込み後に、デバイスはスタートアップ シーケンスを開始します。デバイスは、ビットストリームで指定されたスタートアップ シーケンスで、DONE 信号を High にアサートします (第 5 章の「スタートアップ (手順 8)」参照)。コンフィギュレーション コントローラーは、スタートアップ シーケンスが完了するまで、CCLK 信号を継続して送信す

る必要があります。(DONE が High になった後でも、CCLK 信号が数サイクル必要な場合があります (第 5 章の「スタートアップ (手順 8)」参照)。

CSI\_B および RDWR\_B 信号は、コンフィギュレーション後にディアサートできますが、アサートしたままでもかまいません。SelectMAP ポートがアクティブでなくなっているため、ここで RDWR\_B をトグルしても ABORT は生じません。図 2-8 に、SelectMAP コンフィギュレーションにおける連続的なデータ読み出しのタイミングを示します。

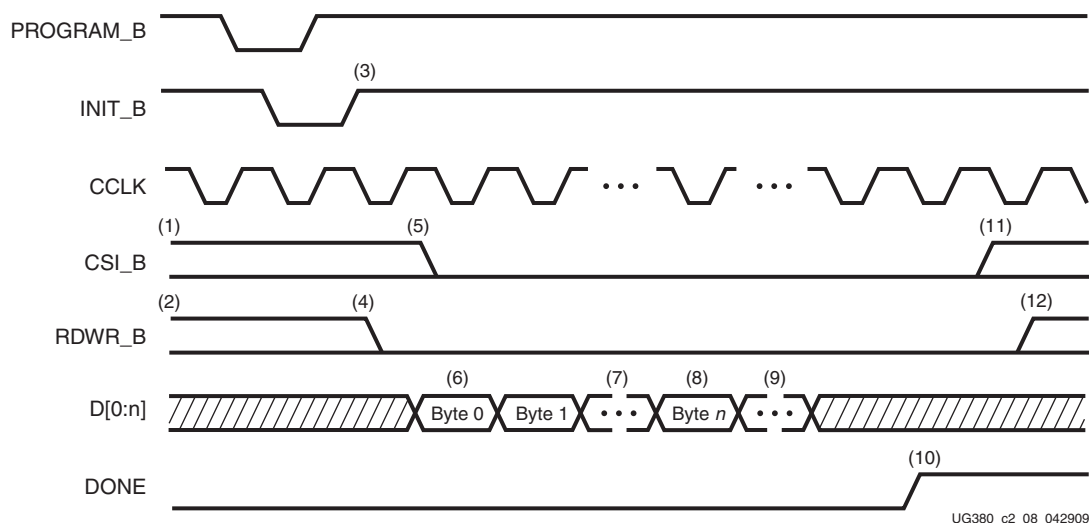


図 2-8 : x8 または x16 SelectMAP データの連続的な読み込み

図 2-8 について説明します。

1. SelectMAP バスにデバイスが 1 つしかない場合、CSI\_B 信号を Low に接続できます。CSI\_B が Low に接続されていないときは、随時アサート可能です。
2. リードバックが不要な場合は RDWR\_B を Low に接続できます。CSI\_B がアサートされた後、RDWR\_B をトグルしないようにしてください。トグルすると ABORT が生じます。詳細は、155 ページの「SelectMAP の ABORT」を参照してください。
3. モード ピンは、INIT\_B が High になったときにサンプルされます。
4. ABORT を回避するため、CSI\_B より前に RDWR\_B をアサートする必要があります。
5. CSI\_B をアサートして SelectMAP インターフェイスを有効にします。
6. CSI\_B がアサートされた後の CCLK の最初の立ち上がりエッジで、最初のバイトが読み込まれます。
7. 各 CCLK の立ち上がりエッジで 1 バイトのコンフィギュレーション ビットストリームが読み込まれます。
8. スタートアップ コマンドが読み込まれると、デバイスはスタートアップ シーケンスを開始します。
9. スタートアップ シーケンスは、最低でも CCLK の 8 サイクル間継続されます (第 5 章の「スタートアップ (手順 8)」参照)。
10. スタートアップ シーケンスで DONE ピンは High になります。スタートアップ シーケンスを完了するために追加の CCLK が必要な場合があります (第 5 章の「スタートアップ (手順 8)」参照)。
11. コンフィギュレーションが完了したら、CSI\_B 信号はディアサートできます。

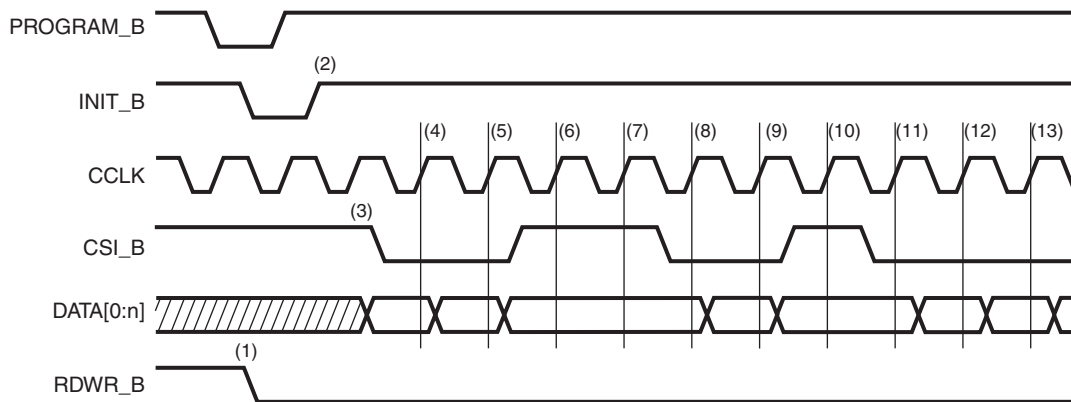
12. CSI\_B 信号がディアサートされたら、RDWR\_B をディアサートできます。

13. データ バスは x8 または x16 が可能です。

### SelectMAP データの不連続読み込み

データの不連続読み込みは、コンフィギュレーション コントローラーからコンフィギュレーション データを連続的に供給できないアプリケーションで使用します。たとえば、新しいデータをフェッチする間コンフィギュレーションが停止するようなコントローラーがこれに該当します。

コンフィギュレーションを一時停止させる方法は 2 つあります。1 つは CSI\_B 信号をディアサートする方法 (フリーランニング CCLK 手法、[図 2-9](#)) で、もう 1 つは、CCLK を停止する方法 (制御 CCLK 手法、[図 2-10](#)) です。



UG380\_c2\_09\_042909

図 2-9 : フリーランニング CCLK 手法を使用した SelectMAP データの不連続読み込み

[図 2-9](#) について説明します。

1. RDWR\_B を Low に駆動し、D[0:n] ピンをコンフィギュレーションの入力として設定します。リードバックが不要な場合は RDWR\_B を Low に接続できます。CSI\_B がアサートされた後、RDWR\_B をトグルしないようにしてください。トグルすると ABORT が生じます。詳細は、[155 ページの「SelectMAP の ABORT」](#)を参照してください。CSI\_B は、同期ワードの途中でディアサートできません。
2. INIT\_B が High にトグルすると、デバイスはコンフィギュレーション可能な状態となります。
3. ユーザーが CSI\_B 信号を Low にアサートして SelectMAP バスを有効にします。SelectMAP バ스에 デバイスが 1 つしかない場合、CSI\_B 信号を Low に接続できます。CSI\_B が Low に接続されていないときは、随時アサート可能です。
4. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。データ バス幅は x8 または x16 が可能です。
5. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。
6. ユーザーが CSI\_B をディアサートし、このバイトは無視されます。
7. ユーザーが CSI\_B をディアサートし、このバイトは無視されます。
8. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。
9. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。
10. ユーザーが CSI\_B をディアサートし、このバイトは無視されます。
11. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。



12. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。

13. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。

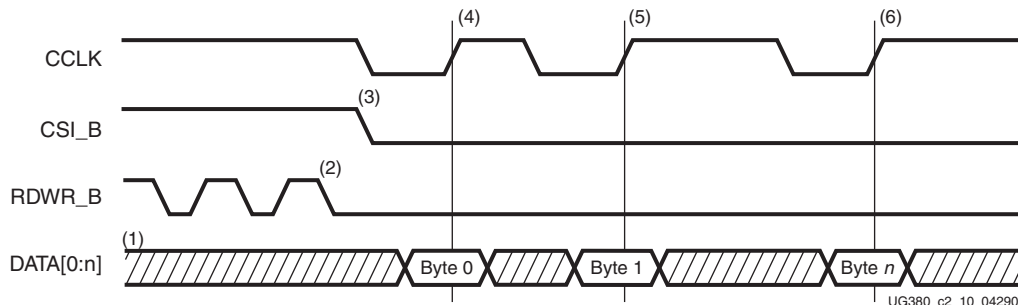


図 2-10： CCLK 制御手法を使用した SelectMAP データの不連続読み込み

図 2-10 について説明します。

1. CSI\_B がディアサートされている間、データ ピンはハイ インピーダンス状態です。データ バスは x8 または x16 が可能です。
2. CSI\_B がディアサートされている間、RDWR\_B はデバイスに影響を与えません。
3. ユーザーが CSI\_B をアサートします。デバイスは、CCLK の立ち上がりエッジでコンフィギュレーション データの読み込みを開始します。
4. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。
5. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。
6. CCLK の立ち上がりエッジで 1 バイトが読み込まれます。

## SelectMAP のデータ順

通常、SelectMAP コンフィギュレーションは、マイクロプロセッサまたは CPLD にあるユーザー アプリケーションを使用して駆動するか、場合によっては、その他の FPGA デバイスを使用して駆動します。これらのアプリケーションでは、コンフィギュレーション データ ファイル内のデータ 順と FPGA で予測されるデータ順の対応関係を理解しておく必要があります。

SelectMAP x8 モードの場合、CCLK の 1 サイクルで 1 バイトのコンフィギュレーション データが読み込まれ、各バイトの MSB は D0 ピンに現れます。D0 = MSB、D7 = LSB という規則は、その他のデバイスが採用しているものとは異なります。x16 モードの場合は、[81 ページの「パラレルバスのビット順」](#)を参照してください。この規則の違いが、カスタム コンフィギュレーション ソリューションを設計する際に混乱の元となることがあります。[表 2-4](#) に、16 進数値 0xABCD を SelectMAP データ バスに読み込む方法を示します。

表 2-4： SelectMAP 8 ビット モードのビット順

CCLK サイクル	16 進数	D0	D1	D2	D3	D4	D5	D6	D7
1	0xAB	1	0	1	0	1	0	1	1
2	0xCD	1	1	0	0	1	1	0	1

注記：

1. D[0:7] は SelectMAP モードのデータ ピンです。

このような一般的な規則と異なるデータ順でも問題なく扱えるアプリケーションもありますが、そうでない場合は、ソース コンフィギュレーション データ ファイルをビット スワップして、データ



ストリームの各バイト内のビット順を逆にした方が扱いやすくなることがあります。このようなアプリケーション用に、ザイリンクスの PROM ファイル生成ソフトウェアでは、ビット スワップした PROM ファイルが生成可能です (77 ページの「コンフィギュレーション ファイルの形式」参照)。

表 2-5 に、FPGA の x8 および x16 モードでのビット順序を示します。

表 2-5 : Spartan-6 FPGA のビット順

モード	ピン															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x16	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
x8									0	1	2	3	4	5	6	7

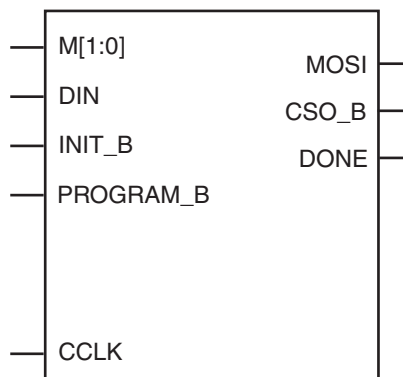
## SPI コンフィギュレーション インターフェイス

マスター シリアル ペリフェラル インターフェイス (SPI) (図 2-11 参照) では、SPI シリアル フラッシュを使用してコンフィギュレーション データを格納できます。Spartan-6 FPGA は、接続されている業界標準の SPI シリアル フラッシュ PROM から直接コンフィギュレーションされます。SPI は標準の 4 線式インターフェイスですが、さまざまな利用可能な SPI フラッシュ メモリで多様な読み出しコマンドおよびプロトコルが使用されています。図 2-12 は、データ幅が x1 または x2 の SPI コンフィギュレーションの接続を示したものです。x2 デバイスでは、D ピンが多目的ピンのデータ入出力ピンとして使用されるので、x1 モードの場合と接続は同様です。46 ページの図 2-13 に SPI x4 の接続を示します。データ ピンを 2 つ追加して 4 ビットのデータ インターフェイスとしています。デイジー チェーン コンフィギュレーションは、SPI x1 モードでのみ利用できます。表 2-5 に、マスター SPI モードでの SPI フラッシュ PROM との接続に使用する FPGA ピンを示します。

iMPACT プログラミング ソフトウェアには、間接プログラムを使用して SPI シリアル フラッシュをプログラムする機能があります。これにより、iMPACT ソフトウェアと SPI フラッシュを Spartan-6 デバイス経由で接続する新しい FPGA デザインがダウンロードされます。以前の FPGA メモリの内容は失われます。サポートされている SPI フラッシュ デバイスのリストは、[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx11/isehelp\\_start.htm](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx11/isehelp_start.htm) で

ISE® ソフトウェアのヘルプを開き、「デバイスのコンフィギュレーションおよびプログラム」の下にある iMPACT ヘルプから「SPI または BPI フラッシュ メモリの間接プログラムの概要」を参照してください。

詳細は、アプリケーション ノート [XAPP974](#) 『Spartan-3A FPGA を使用した SPI シリアル フラッシュ PROM の間接プログラム』を参照してください。



UG380\_c2\_11\_051909

図 2-11 : Spartan-6 FPGA の SPI コンフィギュレーション インターフェイス

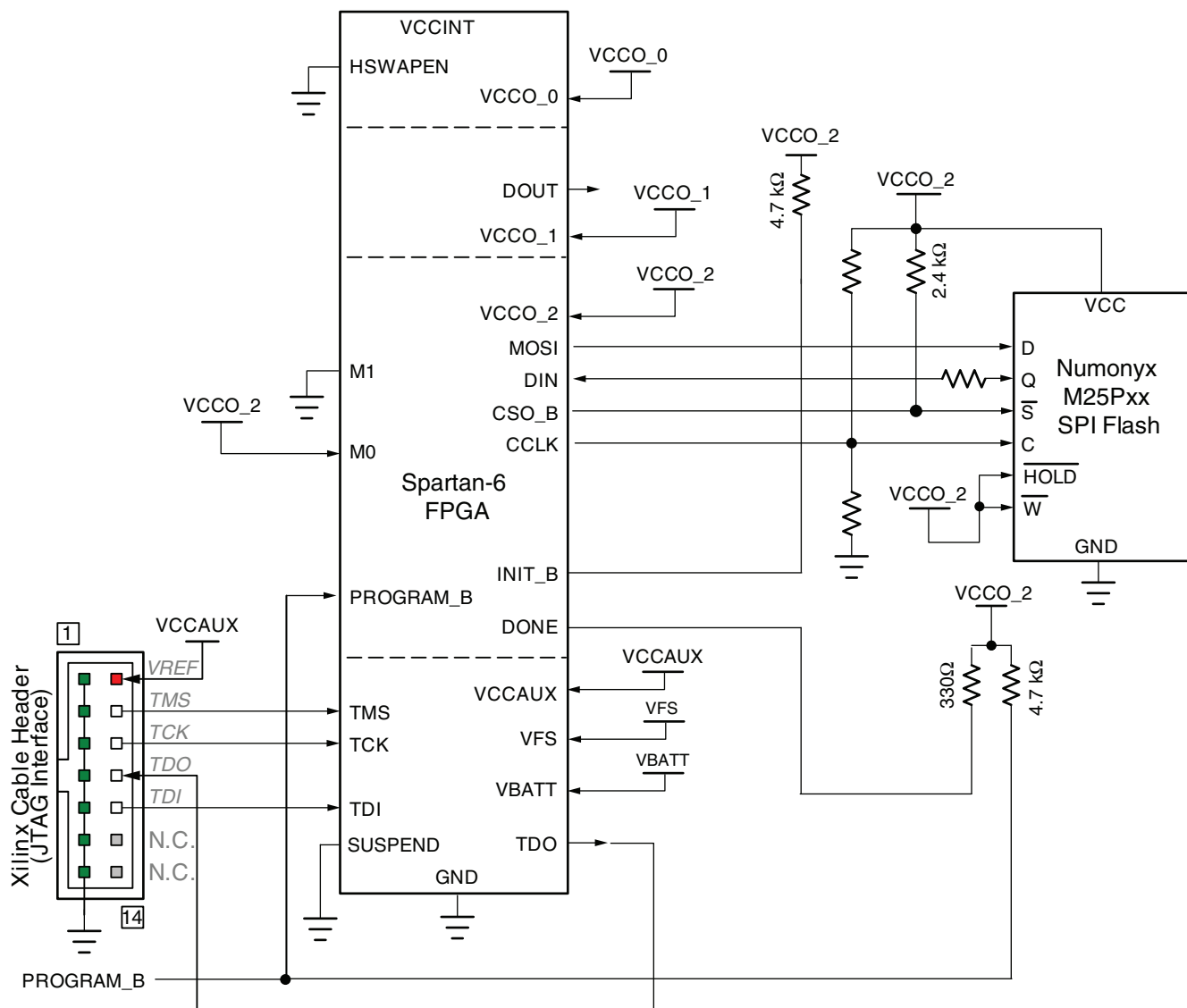
表 2-6 で、SPI コンフィギュレーション インターフェイス の各ピンについて説明します。

表 2-6 : Spartan-6 FPGA SPI コンフィギュレーション インターフェイスで使用するピン

ピン名	FPGA の方向	説明	コンフィギュレーション中	コンフィギュレーション後
HSWAPEN	入力	ユーザー I/O プルアップ制御 コンフィギュレーション中に Low の場合、すべての I/O ピンでプルアップ抵抗が、対応する I/O バンクの V <sub>CCO</sub> 入力に対して有効になります。 0 : コンフィギュレーション中プルアップ 1 : プルアップなし	コンフィギュレーション中 有効なレベルで駆動する	ユーザー I/O
M[1:0]	入力	モード セレクト FPGA コンフィギュレーション モードを選択します。	M[1:0] = 2'b01 INIT_B が High のときサンブルされる	ユーザー I/O
MOSI/ MISO[0]/ CSI_B	出力/入力	マスター FPGA シリアル データ出力およびマスター FPGA シリアル データ入力 SPI フラッシュ PROM のスレーブ データ入力ピンに接続します。	FPGA は SPI フラッシュ メモリに読み出しコマンドと、PROM のシリアル データ入力の開始アドレスを送信する	ユーザー I/O
DIN/D0/ MISO/ MISO[1]	入力	マスター FPGA シリアル データ入力およびスレーブ SPI フラッシュ出力 SPI フラッシュ PROM のスレーブ データ出力ピンに接続します。	FPGA は PROM のシリアル データ出力からシリアル データを受信する	ユーザー I/O
CSO_B	出力	マスター SPI チップ セレクト出力 アクティブ Low です。SPI フラッシュ PROM のスレーブ セレクト入力に接続します。	HSWAPEN_B = 1 の場合、この信号は外部プルアップ抵抗を介して V <sub>CCO</sub> に接続される	SPI フラッシュを無効にし、MOSI、DIN、および CCLK ピンを再度使用するため、コンフィギュレーション後、CSO_B を High に駆動する。オプションで、このピン、MOSI、DIN、および CCLK を再度使用して、SPI フラッシュとの通信を継続できる

表 2-6 : Spartan-6 FPGA SPI コンフィギュレーション インターフェイスで使用するピン (続き)

ピン名	FPGA の方向	説明	コンフィギュレーション中	コンフィギュレーション後
CCLK	出力	コンフィギュレーション クロック FPGA 内部オシレーターで生成されます。 SPI フラッシュ PROM のスレーブ クロック入力に接続します。	PROM のクロック入力を駆動する	ユーザー I/O 未使用の場合 High または Low に駆動する
DOUT	出力	シリアル データ出力 複数の FPGA デイジー チェーン コンフィギュレーションで使用されます。	1 つの FPGA を用いるデザインでは使用されない。 DOUT にはプルアップ抵抗が付けられ、アクティブには駆動しない。デイジー チェーン コンフィギュレーションでは、このピンは、チェーン接続されている次の FPGA の DIN 入力に接続される	ユーザー I/O
INIT_B	オープン ドレイン 双方向 I/O	初期化インジケータ アクティブ Low です。メモリの初期化プロセス (クリア) 中、コンフィギュレーションの開始時点で Low になります。メモリの初期化の終了時点でリリースされ、このときモード ピンがサンプルされます。	コンフィギュレーション中はアクティブ。電源投入後、SPI フラッシュ PROM が操作可能な状態になるまで 2ms 以上必要な場合は、PROM が準備完了になるまで INIT_B を Low に保持する	POST_CRC が有効ではない、または POST_CRC_FLAG が無効の場合はユーザー I/O。 INIT_B にプルアップ抵抗を使用
DONE	オープン ドレイン 双方向 I/O	FPGA コンフィギュレーション完了 コンフィギュレーション中はアクティブ Low です。FPGA でコンフィギュレーションが完了すると High になります。	Low の場合、FPGA がコンフィギュレーションされていないことを示す	専用。外部プルアップを介して High になる。High の場合、FPGA のコンフィギュレーションが完了していることを示す
PROGRAM_B	入力	FPGA のプログラム アクティブ Low です。500ns 以上 Low にアサートされると、FPGA はコンフィギュレーション メモリをクリアして DONE および INIT_B ピンをリセットし、PROGRAM_B が High になった後にコンフィギュレーション プロセスを再開します。	コンフィギュレーションを開始するには High にする必要がある	PROGRAM_B を Low に駆動してリリースし、FPGA を再度プログラムする。 PROGRAM_B を保持して FPGA の I/O ピンをハイインピーダンス状態にすると、SPI フラッシュ PROM ピンの直接プログラムが可能になる
MISO[3:2]	入力	マスター FPGA シリアル データ入力およびスレーブ SPI データ出力	Fast-Read Quad Output (高速読み出しクアッド出力) コマンドを用いる場合にのみ使用	ユーザー I/O



Refer to the Notes following this figure for related information.

UG380 c2 12 062510

図 2-12 : Spartan-6 FPGA の SPI コンフィギュレーション インターフェイス

図 2-12 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、74 ページの表 5-2 を参照してください。
2. デイジー チェーン コンフィギュレーション モードでは、DOUT を下位 FPGA の DIN に接続してください。
3. CCLK の終端については、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
4. フラッシュから FPGA へのデータパスには、オーバーシュートを最小限に抑えるために直列抵抗の使用を検討してください。適切な抵抗値はシミュレーションで求めることができます。
5. Spartan-6 FPGA の VCCO\_2 電源電圧は SPI デバイスの  $V_{CC}$  と同じ電圧である必要があります。

6. CSO\_B および MOSI は、CCLK の立ち下がりエッジで使用されます。
7. DIN は、CCLK の立ち上がりエッジで使用されます。
8. SPI フラッシュ側には、Write Protect や Hold などの追加ピンがあり、これらは FPGA コンフィギュレーションでは使用されません (読み出し専用)。SPI ベンダーの仕様に従って適切に切断する必要があります。
9. HSWAPEN が未接続のままか High に接続されている場合は、CSO\_B にプルアップ抵抗が必要です。
10. クロック ソースが内部オシレーターの場合、CCLK の周波数は BitGen の **ConfigRate** オプションを使用して調節できます。外部ソースが使用されている場合は、[56 ページの「マスターモードの外部コンフィギュレーション クロック」](#)を参照してください。
11. DONE ピンは、デフォルトでは内部プルアップを持つオープン ドレイン出力です。外部プルアップを追加することを推奨します。DONE ピンには、BitGen の **-g DriveDone** オプションを使用して有効にできる、プログラム可能なアクティブなドライバーがあります。
12. DONE ピンをリリースするスタートアップ サイクル前にデジタル クロック マネージャー (DCM) または PLL ロック待機が有効になっている場合、スタートアップ待機条件が満たされて DONE がリリースされるまで、FPGA はデータを出力し続けます。マルチブート コンフィギュレーション固有の注意点は、[140 ページの「マルチブート イメージ間に必要なデータ空間」](#)を参照してください。
13. [図 2-12](#) に Numonyx 社の SPI フラッシュを示します。  
詳細は、[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx11/isehelp\\_start.htm](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx11/isehelp_start.htm) で ISE® ソフトウェアのヘルプを開き、「デバイスのコンフィギュレーションおよびプログラム」の下にある iMPACT ヘルプから「SPI または BPI フラッシュ メモリの間接プログラムの概要」を参照して、サポートされているデバイスを確認してください。
14.  $V_{FS}$  は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、[94 ページの「eFUSE」](#)を参照してください。
15.  $V_{BATT}$  は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供される、AES キーの格納用の電源です。AES 暗号が未使用の場合、 $V_{BATT}$  は GND に接続するか、未接続のままでもかまいません。
16. VCCO\_2 が 1.8V の場合、VCCAUX は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、VCCAUX は 2.5V または 3.3V にできます。
17. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

Spartan-6 FPGA では、マスターシリアルモードでの SPI PROM の x4 コンフィギュレーションもサポートされています。詳細は、[図 2-13](#) を参照してください。

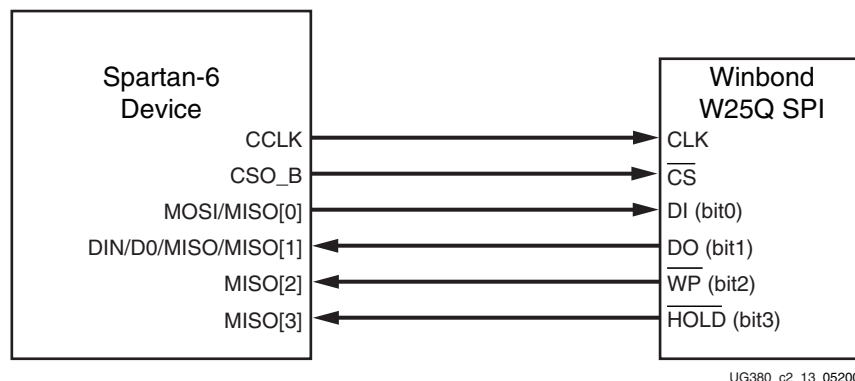


図 2-13： マスター シリアル クアッド ビット SPI コンフィギュレーション

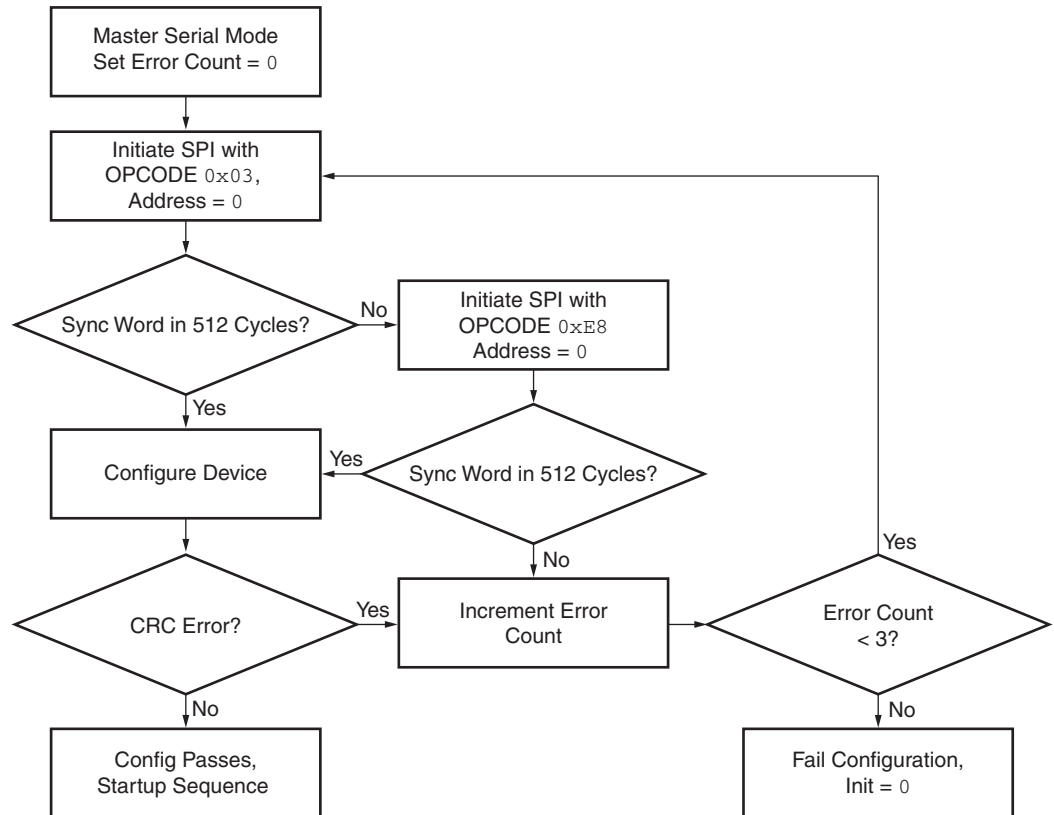
図 2-13 について説明します。

1. 図 2-13 に示す接続では、Winbond W25Q SPI シリーズ フラッシュ PROM が使用されています。クアッド出力を有効にするには、デバイスがクアッド出力モードで伝送が可能となる前に PROM の `status_reg[9]` の QE ビットを 1 に設定する必要があります。この設定は iMPACT ソフトウェアでのプログラム実行時に可能です。
2. CCLK は FPGA または外部クロック ソースから供給されます。
3. PROGRAM\_B、INIT\_B、および M0 ピンにはデフォルトでプルアップが付いています。
4. x4 のソフトウェア サポートには、BitGen で x4 機能を有効にする必要があります。
5. x4 出力を有効にするため SPI デバイスは特定のレジスタ設定でプログラムする必要があります。これは iMPACT で設定できます。

## マスター SPI ベンダー自動検出およびエラー処理機能

SPI 読み出しコマンドは、初期デバイス コンフィギュレーション向けの読み出しコマンド ループを用いて自動的に選択されます。図 2-14 に、このアルゴリズムを示します。SPI x2 および x4 アプリケーションでは、初期データの読み込みにこのシーケンスを使用します。最初のコマンド セットが FPGA に発行された後、この読み出しコマンドは Mode\_Reg に変わり、コンフィギュレーションは IPROG コマンドを使用して x2 または x4 モードに変わります。これらのモードを有効にするには、BitGen の `spi_buswidth` オプションで SPI x2 または x4 コマンドを設定する必要があります。

マルチブート アプリケーションの場合、この読み出しコマンドは、Mode\_Reg、General 2、General 4 レジスタに含まれるマルチブート設定と共に使用されるように手動で設定する必要があります。

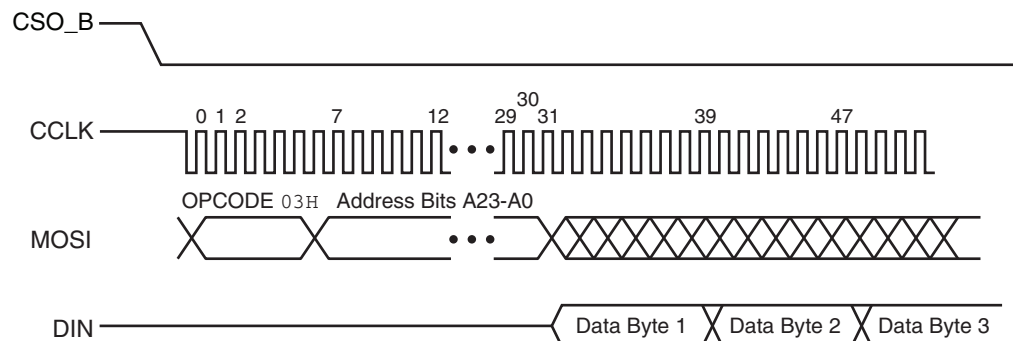


UG380\_c2\_14\_011310

図 2-14：初期コンフィギュレーション中の読み出しコマンド ループ

## マスター SPI のタイミング波形

図 2-15 に、デバイスが最初に発行する読み出しコマンドである、SPI 読み出し (Opcode = 03H) を示します。この読み出しコマンドで同期ワードが返されない場合は、E8h の次の読み出しコマンドがデバイスへ発行されます (図 2-16 参照)。



UG380\_c2\_15\_052009

図 2-15：マスター SPI のタイミング図 (Opcode = 03h)

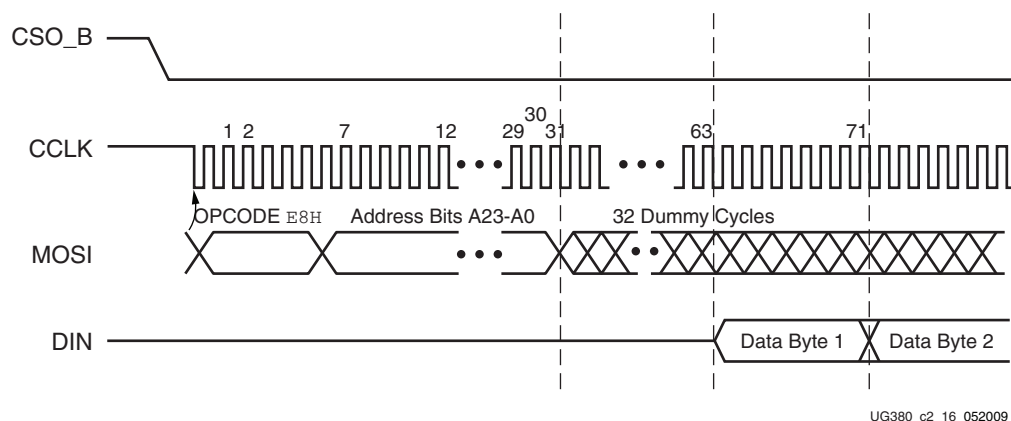


図 2-16： マスター シリアル SPI のタイミング図 (Opcode = E8h)

## マスター SPI デュアル (x2) およびクアッド (x4) 読み出しコマンド

Spartan-6 FPGA のマスター SPI コンフィギュレーション モードは、クアッド I/O SPI をサポートする Winbond W25Q フラッシュ メモリのデュアル (x2) およびクアッド (x4) ビット メモリの読み出しコマンドに対応しています。ソフトウェアでこのコンフィギュレーションを有効にするには、BitGen の `spi_buswidth` オプションを使用して SPI x2 または x4 の BIT ファイルを生成します。FPGA は始め x1 モードでブートするため、その後 x2 または x4 モードに切り替えます。

x2 モードでは、Fast-Read Dual Output (高速読み出しデュアル出力) (3Bh) 命令が発行されますが、これは、DO ピンだけでなく DO および DIO (MOSI) の 2 つのピンでデータが出力される点を除き、標準の Fast Read (高速読み出し) (0Bh) 命令と同じです。これにより、デュアル出力から標準 SPI デバイスの 2 倍のレートでデータを伝送できます。図 2-17 に、デュアル読み出しコマンド (3Bh) で Winbond SPI フラッシュを使用する際のマスター シリアル SPI コンフィギュレーション モードのタイミング図を示します。

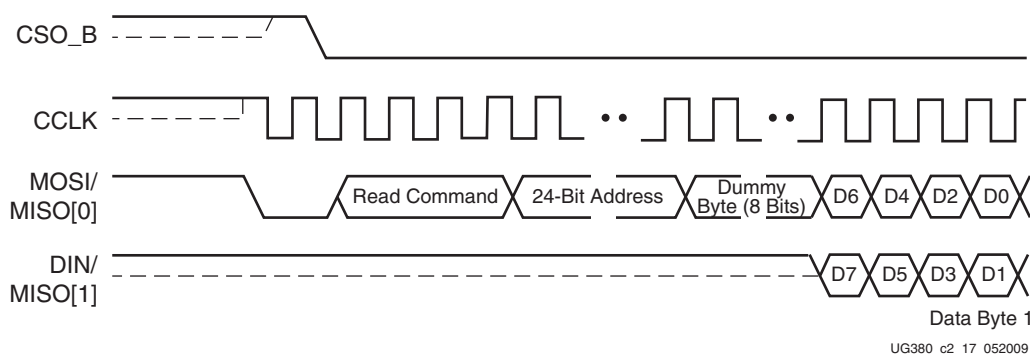


図 2-17： Winbond SPI デュアル読み出しコマンド (3Bh) のタイミング図

x4 モードでは、Fast-Read Quad Output (高速読み出しクアッド出力) (6Bh) 命令が発行されますが、これは、DO ピンだけでなく、4 つのピンでデータが出力される点を除き、標準の Fast Read (高速読み出し) (0Bh) 命令と同じです。これにより、クアッド出力から標準 SPI デバイスの 4 倍のレートでデータを伝送できます。図 2-18 に、クアッド読み出しコマンド (6Bh) で Winbond SPI フラッシュを使用する際のマスター シリアル SPI コンフィギュレーション モードのタイミング図を示します。



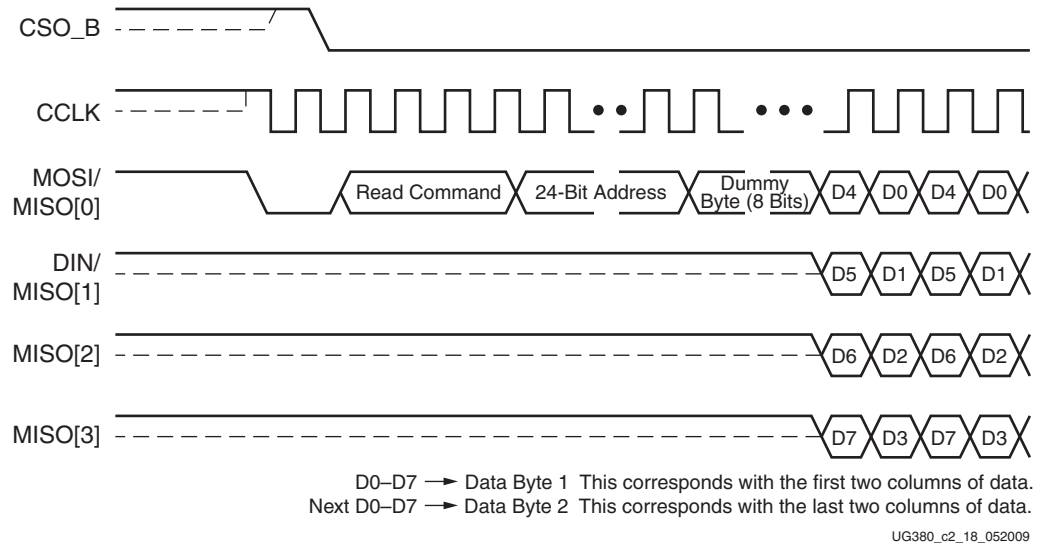


図 2-18 : Winbond SPI クアッド読み出しビット コマンド (6Bh) のタイミング図

## パワーオン シーケンスに関する注意事項

電源を投入すると、FPGA でコンフィギュレーションが自動的に開始します。FPGA がマスター シリアル SPI コンフィギュレーション モードの場合、CS0\_B を Low にアサートして SPI フラッシュを選択し、この SPI フラッシュへの読み出しコマンドを駆動します。SPI フラッシュは、CS0\_B が Low にアサートされて読み出しコマンドが送信される前に、コマンドを受信可能な状態になっている必要があります。

FPGA と SPI フラッシュには異なる電源レールから電源が供給でき、同じ電源を使用した場合でも異なる立ち上がり地点で応答することがあるため、FPGA と SPI フラッシュのパワーオン シーケンスと電源の立ち上がりには注意が必要です。パワーオン シーケンスや電源の立ち上がりによっては、FPGA が SPI フラッシュの前に、あるいは SPI フラッシュが FPGA の前に立ち上がることがあります。また、一部の SPI フラッシュ デバイスでは電源投入からデバイスが選択可能になるまでの最小時間 (通常は数ミリ秒) が指定されていることがあり、この間はデバイスを選択しないようにする必要があります。電源の立ち上がりがほぼ同時の多くのシステムでは、FPGA のパワーオン リセット時間 ( $T_{POR}$ ) を使用して FPGA の コンフィギュレーションの開始を遅らせ、SPI フラッシュの準備が完了するようにできます。一般にシステムを設計する際は、パワーオン シーケンス、電源の立ち上がり、FPGA パワーオン リセットのタイミング、および SPI フラッシュの電源投入のタイミングが、FPGA コンフィギュレーションの開始と SPI フラッシュの準備完了のタイミングの関係にどのように影響するかを考慮する必要があります。Spartan-6 FPGA の電源要件およびタイミングの詳細は、データシート [DS162](#) 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』を参照してください。SPI フラッシュの電源投入のタイミング要件は、SPI フラッシュのデータシートを参照してください。

FPGA がコンフィギュレーションを開始する前に SPI フラッシュを確実にコマンド受信可能な状態にするには、次のいずれかの手法でシステムを設計するようにします。

- FPGA のコンフィギュレーションが開始する前に SPI フラッシュに電源が投入されて非同期読み出しが可能な状態になるよう、パワーオン シーケンスを制御する
- 電源投入時から FPGA の PROGRAM\_B ピンを Low のまま保持して FPGA のコンフィギュレーションの開始を遅らせ、SPI フラッシュに電源が完全に投入されてコマンドを受信できるようになった後 PROGRAM\_B ピンを High にする

- 電源投入時から FPGA の INIT\_B ピンを Low のまま保持して FPGA のコンフィギュレーションの開始を遅らせ、SPI フラッシュがコマンドを受信できるようになった後 INIT\_B ピンを High にする

SPI フラッシュを使用した FPGA のコンフィギュレーション方法および iMPACT ソフトウェアを使用したシステム内の SPI プログラム方法については、アプリケーション ノート [XAPP951](#) 『SPI シリアル フラッシュを使用したザイリンクス FPGA のコンフィギュレーション』を参照してください。

## SPI シリアル デイジー チェーン

シリアル デイジー チェーン アプリケーションでは、最初のデバイスを SPI モードに設定し、すべての下位デバイスをスレーブ シリアル モードに設定できます。この場合、すべてのコンフィギュレーション ビットストリームは、1 つの SPI デバイス内部に格納可能です。マスターおよびスレーブ デイジー チェーンのビットストリーム形式は、完全に同一です。詳細は、[147 ページ](#)の「[シリアル デイジー チェーン](#)」を参照してください。

## マスター BPI コンフィギュレーション インターフェイス

[図 2-19](#) に示すマスター バイト幅ペリフェラル インターフェイス (BPI) では、Spartan-6 FPGA ファミリを業界標準の平行 NOR フラッシュからコンフィギュレーションできます ([図 2-20](#) 参照)。Spartan-6 FPGA では、最大 512Mb の平行 NOR フラッシュがサポートされ、最大 26 のアドレス信号を使用してアクセスできます。コンフィギュレーションで使用するフラッシュの最大サイズを制限するアドレス信号の数は、Spartan-6 デバイスおよびパッケージによって異なります。

BPI には次のような注意点があります。

- マスター BPI コンフィギュレーション インターフェイスがターゲットの場合、バンク 1 にあるメモリ コントローラー ブロック (MCB-M1) は使用できません。このデザインでは、MCB の多目的ピンまたは BPI コンフィギュレーション インターフェイスの多目的ピンのいずれかを使用できますが、両方は使用できません。
- XC6SLX25/T デバイスは BPI インターフェイスをサポートしていません。
- XC6SLX4 デバイスおよび TQG144 および CPG196 パッケージの Spartan-6 FPGA は BPI インターフェイスをサポートしません。
- CSG225 パッケージでは、アドレス コンフィギュレーション ピン A22 および A23 を使用できません。

iMPACT プログラミング ソフトウェアには、間接プログラムを使用してトップブート 平行 NOR フラッシュをプログラムする機能があります。Spartan-6 デバイスを介して iMPACT ソフトウェアからフラッシュ デバイスへの接続が可能になるよう、FPGA デザインに IP を追加する必要があります。サポートされている BPI デバイスのリストは、

[http://japan.xilinx.com/support/documentation/sw\\_manuals/xilinx11/isehelp\\_start.htm](http://japan.xilinx.com/support/documentation/sw_manuals/xilinx11/isehelp_start.htm) を参照し、ISE ソフトウェアのヘルプを開き、「デバイスのコンフィギュレーションおよびプログラム」の下にある iMPACT ヘルプから「SPI または BPI フラッシュ メモリの間接プログラムの概要」を参照してください。

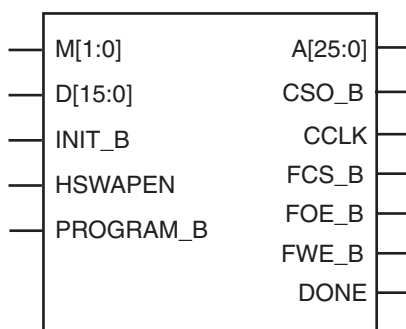
詳細は、アプリケーション ノート [XAPP973](#) 『Virtex-5 FPGA を使用した BPI PROM の間接プログラム』を参照してください。

FPGA は最大 26 のアドレス ラインを駆動して、接続された平行 フラッシュにアクセスします。このコンフィギュレーションでは、非同期読み出しモードのみが使用でき、FPGA がアドレスバスを駆動してこれに対してフラッシュ PROM がビットストリーム データを駆動します。サポー

トされているバス幅は x8 および x16 です。パラレル NOR フラッシュで x8 と x16 の両方のデータ幅がサポートされている場合、BYTE# 信号を必要な幅に応じて適切なレベルに接続することが必要です。78 ページの「同期ワード/バス幅の自動検出」で説明されているように、バス幅は自動で検出されます。

マスター BPI モードでパラレル NOR フラッシュ デバイスが使用されているとき、CCLK 出力はこのフラッシュ デバイスに接続されません。しかし、フラッシュ データは CCLK の立ち上がりエッジでサンプルされます。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。BPI に関連するタイミング パラメーターは、CCLK を基準クロックとします。

マスター BPI モードでは、アドレスは 0 から開始し、DONE ピンがアサートされるまで 1 ずつ増加します。アドレスが最大値 26'h3FFFFFF に達しても、コンフィギュレーションが完了していない場合 (DONE がアサートされていない)、カウンタはラップアラウンドし、0 からもう一度カウントを開始します。



UG380\_c2\_25\_121109

図 2-19 : Spartan-6 FPGA BPI コンフィギュレーション インターフェイス

表 2-7 で、BPI コンフィギュレーション インターフェイスのピンについて説明します。

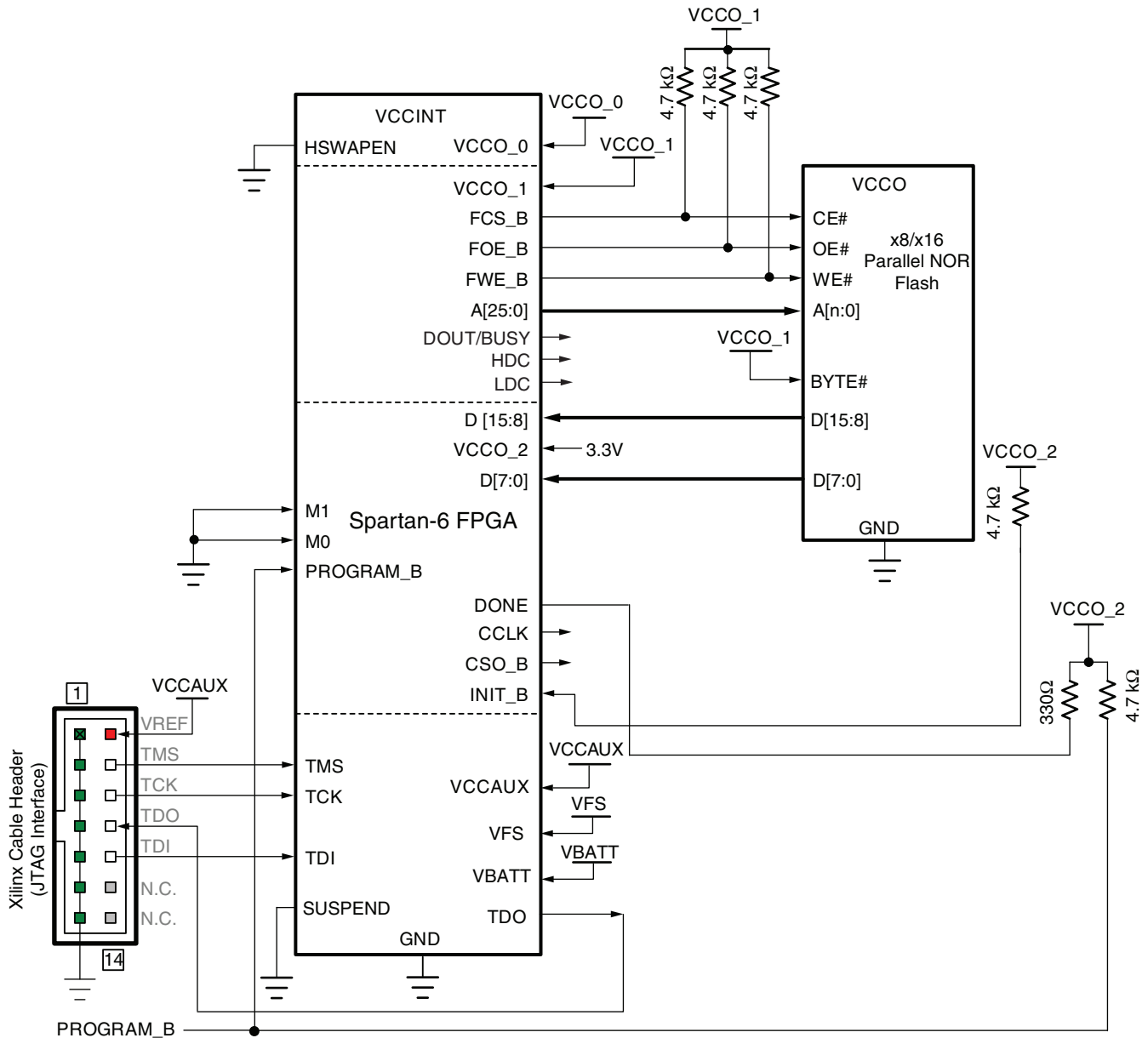
FPGA がパラレル NOR フラッシュからのコンフィギュレーション中に再プログラムされる場合、INIT ピンを BPI リセットに接続して BPI を既知の状態に設定できます。

表 2-7 : Spartan-6 FPGA BPI コンフィギュレーション インターフェイス ピン

ピン名	タイプ	専用/多目的	説明
M[1:0]	入力	多目的	モード ピンは、パラレル NOR フラッシュでコンフィギュレーションされる場合、マスター BPI モードの 00 に設定されます。 00 = マスター BPI モード
HSWAPEN	入力	多目的	コンフィギュレーション中 I/O プルアップ抵抗を制御します。弱いプルアップ抵抗が付いています。 0 = コンフィギュレーション中にプルアップ 1 = コンフィギュレーション中にトライステート
DONE	双方向、オープンドレイン、またはアクティブ	専用	コンフィギュレーションの完了を示すアクティブ High の信号です。 0 = FPGA コンフィギュレーション未完了 1 = FPGA コンフィギュレーション完了

表 2-7 : Spartan-6 FPGA BPI コンフィギュレーション インターフェイス ピン (続き)

ピン名	タイプ	専用/多目的	説明
INIT_B	入力または出力、オープンドレイン	多目的	モード ピンのサンプル前は入力となり、 <b>Low</b> に保持することでコンフィギュレーションを遅延させることができます。モード ピンのサンプル後はオープン ドレインのアクティブ <b>Low</b> 入力となり、コンフィギュレーション中の CRC エラーの有無を示します。  0 = CRC エラー 1 = CRC エラーなし  SEU 検出機能が有効になっている場合、リードバック CRC エラーが検出されると、オプションで INIT_B が <b>Low</b> に駆動されます。
PROGRAM_B	入力	専用	アクティブ <b>Low</b> の非同期フルチップ リセットです。
CCLK	出力	多目的	コンフィギュレーション クロック出力です。パラレル <b>NOR</b> フラッシュには直接接続せず、アドレスおよびサンプル読み出しデータを生成するために内部で使用します。
FCS_B	出力	多目的	アクティブ <b>Low</b> のフラッシュ チップ セレクト出力です。コンフィギュレーション中はアクティブに <b>Low</b> に駆動され、コンフィギュレーション後はトライステートになります。このピンには、コンフィギュレーション中は弱いプルアップ抵抗があります。デフォルトでは、コンフィギュレーション後に弱いプルダウン抵抗が有効になります。
FOE_B	出力	多目的	アクティブ <b>Low</b> のフラッシュ出力イネーブルです。コンフィギュレーション中はアクティブに <b>Low</b> に駆動され、コンフィギュレーション後はトライステートになります。このピンには、コンフィギュレーション中は弱いプルアップ抵抗があります。デフォルトでは、コンフィギュレーション後に弱いプルダウン抵抗が有効になります。
FWE_B	出力	多目的	アクティブ <b>Low</b> のフラッシュ ライト イネーブルです。コンフィギュレーション中はアクティブに <b>High</b> に駆動され、コンフィギュレーション後はトライステートになります。このピンには、コンフィギュレーション中は弱いプルアップ抵抗があります。デフォルトでは、コンフィギュレーション後に弱いプルダウン抵抗が有効になります。
A[25:0]	出力	多目的	アドレス出力
D[15:0]	入力	多目的	データ入力で、FPGA の CCLK の立ち上がりエッジでサンプルされます。
CSO_B	出力	多目的	パラレル デイジー チェーンのアクティブ <b>Low</b> チップ セレクト出力です。FPGA が 1 つのアプリケーションでは使用されません。
HDC	出力	多目的	HDC (High During Configuration) は、コンフィギュレーション中は <b>High</b> となり、バイト幅出力と 16 ビット出力を制御するためにフラッシュ デバイスに接続できます。
LDC	出力	多目的	LDC (Low During Configuration) は、コンフィギュレーション中は <b>Low</b> となり、バイト幅出力と 16 ビット出力を制御するためにフラッシュ デバイスに接続できます。



Refer to the Notes following this figure for related information.

UG380\_c2\_19\_062510

図 2-20 : Spartan-6 FPGA のマスター BPI コンフィギュレーション インターフェイス

図 2-20 について説明します。

1. 内部ピン終端および HSWAPEN の影響を受けるピンについては、74 ページの表 5-2 を参照してください。
2. CCLK ネットにはテブナン並列終端が必要です。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
3. M[1:0] = 00 の場合、マスター BPI モードです。
4. 図 2-20 に、x16 BPI インターフェイスを示します。x8 BPI インターフェイスでは D[7:0] のみが使用されます。詳細は、78 ページの「同期ワード/バス幅の自動検出」を参照してください。

5. VCCO\_1 および VCCO\_2 は、フラッシュ デバイスと通信するため、同じ電圧である必要があります。
6. A24 および A25 は、デバイスによって I/O バンク 5 に配置できます。選択したデバイスのピン配置を参照してください。
7. データ ピンへのビットストリームの送信は、SelectMAP モードのビット スワップと同じ規則に従います。詳細は、[81 ページの「パラレル バスのビット順」](#)を参照してください。
8. フラッシュ プログラミングが不要な場合は、FCS\_B、FOE\_B、および FWE\_B を別々に接続できます。つまり、DONE を FCS\_B、FOE\_B を Low に、そして FWE\_B を High に接続可能です。
9. CCLK 出力はフラッシュの接続には使用されませんが、コンフィギュレーション中にフラッシュの読み出しデータのサンプルに使用されます。すべてのタイミングは CCLK を基準としています。CCLK ピンを駆動したり、High または Low に接続しないでください。
10. HSWAPEN が未接続のままか High に接続されている場合は、FCS\_B にはプルアップ抵抗が必要です。
11. DONE ピンは、デフォルトでは内部プルアップを持つオープン ドレイン出力です。外部プルアップを追加することを推奨します。DONE ピンには、BitGen のオプション **-g DriveDone** を使用して有効にできる、プログラム可能でアクティブなドライバーがあります。
12. [140 ページの「マルチブート イメージ間に必要なデータ空間」](#)では、DCM または PLL のロック待機が有効になる場合について説明しています。
13. FPGA を BPI モードでデイジー チェーン接続する方法は、[第 9 章「アドバンス コンフィギュレーション インターフェイス」](#)を参照してください。
14. 実際のフラッシュ信号の接続については、パラレル NOR フラッシュ ベンダー提供のデータシートを参照してください。アドレスが正しく揃うようにするため、使用するバイト / ワードモードに関してフラッシュ ファミリのアドレス LSB に十分注意してください。すべてのフラッシュ ファミリでアドレス LSB に A0 が使用されているわけではありません。
15. クロック ソースが内部オシレーターの場合、CCLK の周波数は BitGen の **ConfigRate** オプションを使用して調節できます。外部クロック ソースが使用されている場合は、[56 ページの「マスター モードの外部コンフィギュレーション クロック」](#)を参照してください。
16. V<sub>FS</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供され、eFUSE プログラミングに使用されます。詳細は、[94 ページの「eFUSE」](#)を参照してください。
17. V<sub>BATT</sub> は XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスで提供される、AES キーの格納用の電源です。AES 暗号が未使用の場合、V<sub>BATT</sub> は GND に接続されるか、未接続のままでもかまいません。
18. VCCO\_2 が 1.8V の場合、V<sub>CCAUX</sub> は 2.5V にする必要があります。VCCO\_2 が 2.5V の場合、V<sub>CCAUX</sub> は 2.5V または 3.3V にできます。
19. 電源投入中およびコンフィギュレーション中は、SUSPEND ピンを Low にする必要があります。Suspend 機能を使用しない場合、SUSPEND ピンは GND に接続してください。

図 2-21 に、BPI コンフィギュレーションの波形を示します。

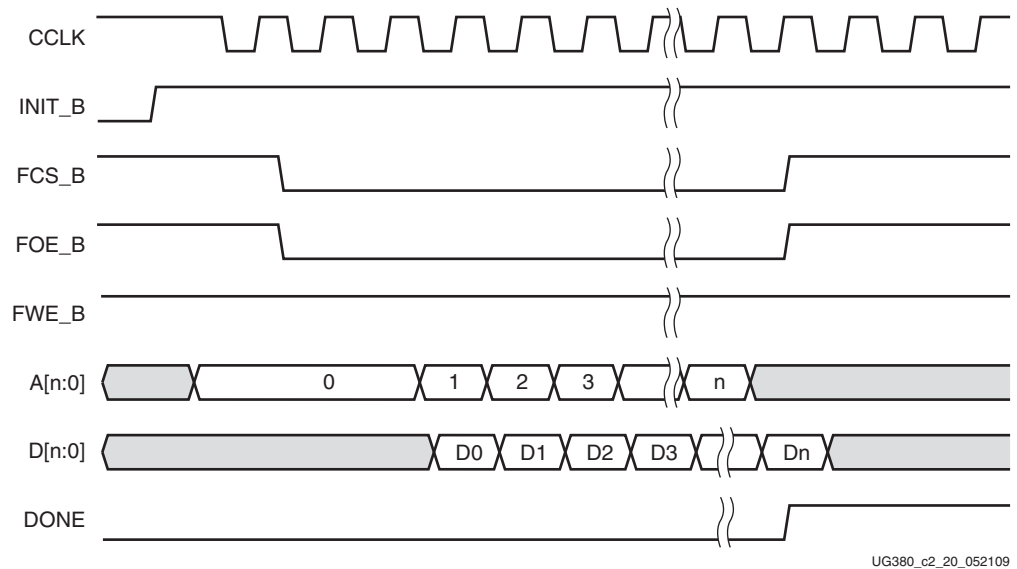


図 2-21 : Spartan-6 FPGA の BPI コンフィギュレーションの波形

図 2-21 について説明します。

1. BPI モードでは CCLK は出力です。パラレル NOR フラッシュには CCLK は必要ありませんが、Spartan-6 FPGA は CCLK の立ち上がりエッジを使用して D[n:0] ピンをサンプルします。
2. DONE ピンが High になると、Spartan-6 FPGA はビットストリームの読み込みを停止します。
3. DONE ピンが High になると、Spartan-6 FPGA はビットストリームの読み込みを停止します。これは、デフォルトでは DONE が High になった 1 サイクル後です。
4. D[n:0] の n は 7 または 15 です。A[n:0] の n は最大 25 までの値です。
5. FCS\_B、FOE\_B、および FWE\_B には、コンフィギュレーション後、I/O 制約または外部プルアップ抵抗によって弱いプルアップ抵抗を付ける必要があります。
6. マスター BPI の最初のアドレス 0 は、初期レイテンシのため複数サイクル間継続します。

## パワーオン シーケンスに関する注意事項

電源を投入すると、FPGA でコンフィギュレーションが自動的に開始します。FPGA がマスター BPI コンフィギュレーションモードの場合、FCS\_B が Low にアサートされ、アドレスのシーケンスが駆動されてパラレル NOR フラッシュのビットストリームが読み出されます。パラレル NOR フラッシュで保存されているビットストリームが正しく出力されるようにするため、FCS\_B が Low にアサートされ、最初のアドレスが送信される前に、パラレル NOR フラッシュが非同期読み出し可能な状態になっている必要があります。

FPGA とパラレル NOR フラッシュには異なる電源レールから電源を供給でき、また同じ電源が使用されている場合でも異なる立ち上がり地点で応答することがあるため、FPGA とパラレル NOR フラッシュのパワーオン シーケンスと電源の立ち上がりには注意が必要です。パワーオンシーケンスまたは電源の立ち上がりによっては、FPGA がパラレル NOR フラッシュの前に、またはパラレル NOR フラッシュが FPGA の前に立ち上がる場合があります。電源の立ち上がりがほぼ同時の多くのシステムでは、FPGA のパワーオン リセット時間 ( $T_{POR}$ ) を使用して FPGA のコンフィギュレーションの開始を遅らせ、パラレル NOR フラッシュの準備が完了するようにできます。一般に



システムを設計する際は、パワーオン シーケンス、電源の立ち上がり、FPGA パワーオン リセットのタイミング、およびパラレル NOR フラッシュの電源投入のタイミングが、FPGA コンフィギュレーションの開始とパラレル NOR フラッシュで非同期読み出しの準備が完了するタイミングの関係にどのように影響するかを考慮する必要があります。Spartan-6 FPGA の電源要件およびタイミングの詳細は、『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』( [DS162](#)) を参照してください。

FPGA がコンフィギュレーションを開始する前にパラレル NOR フラッシュを確実に非同期読み出し可能な状態にするには、次のいずれかの手法でシステムを設計するようにします。

- FPGA のコンフィギュレーションが開始する前にパラレル NOR フラッシュに電源が投入されて非同期読み出しが可能な状態になるよう、パワーオン シーケンスを制御する
- 電源投入時から FPGA の PROGRAM\_B ピンを Low のまま保持して FPGA のコンフィギュレーションの開始を遅らせ、パラレル NOR フラッシュに電源が完全に投入されて非同期読み出し可能になった後 PROGRAM\_B ピンを High にする
- 電源投入時から FPGA の INIT\_B ピンを Low のまま保持して FPGA コンフィギュレーションの開始を遅らせ、パラレル NOR フラッシュが非同期読み出し可能になった後に、INIT\_B ピンを High にする

## マスター モードの外部コンフィギュレーション クロック

デフォルトで、Spartan-6 FPGA は内部生成されたクロック ソースを使用してマスター モードのコンフィギュレーションを実行します。しかし、Spartan-6 FPGA では、マスター モードのコンフィギュレーション中に外部クロック ソースに動的に切り替える機能がサポートされています。外部クロック ソースは、高速で安定したコンフィギュレーションが必要なアプリケーションに適しています。

表 2-8 : Spartan-6 の外部コンフィギュレーション クロック インターフェイスで使用するピン

ピン名	タイプ	専用/ 多目的	説明
USERCCLK	入力	多目的	すべてのマスター コンフィギュレーション モードの外部コンフィギュレーション クロック ソース

USERCCLK は多目的ピンで、コンフィギュレーション後にアプリケーションで使用可能です。マスター モード コンフィギュレーション中にこの外部クロック ソースを有効にするには、BitGen の **ExtMasterCclk\_en** オプションを使用します。USERCCLK 周波数は BitGen の **ExtMasterCclk\_divide** オプションを使用して分周できます。使用可能な値は 1 (デフォルト) および 2 ~ 1022 までのすべての偶数値です。USERCCLK の I/O 規格は 8mA の LVCMOS25 Slow スルー レートです。コンフィギュレーションは FPGA 内部オシレーターで生成された CCLK で開始します。コンフィギュレーション クロック レジスタ設定がビットストリームで達成されると、FPGA は内部オシレーターから USERCCLK のクロックに切り替えます (BitGen のオプション **ExtMasterCclk\_divide** での設定どおり)。クロック マルチプレクサーは、遷移中にグリッチなしの出力クロックを生成するように設計されています。



## コンフィギュレーション クロック (CCLK) のボード レイアウト

Spartan-6 FPGA のコンフィギュレーション I/O は、LVCMOS25 Slow スルー レートの 8mA I/O 規格を採用しています。この場合、シグナル インテグリティを向上させるために PCB トレース配線および終端処理を適切に行う必要があります。

基本的なガイドラインは次のとおりです。

- $50\Omega$  制御のインピーダンス伝送ラインとして CCLK ネットを配線します。
- CCLK ネットは、常に分岐させずに配線します。図 2-25 に示すようなスタート ポジは使用しないでください。
- 必要に応じて、8mm (0.3 インチ) 未満のタブを使用してください。
- CCLK 伝送ラインの終端は、 $V_{CCO}$  に  $100\Omega$ 、および GND に  $100\Omega$  の並列終端処理をします ( $V_{CCO}/2$  のテブナン終端と同等、トレース特性インピーダンスは  $50\Omega$  とする)。
- マスター コンフィギュレーション モードでは、CCLK ピンがデザイン内で未使用の場合は駆動されません。このピンがフロートしないように、ロジック レベルに駆動してください。

マスター FPGA の CCLK を含む各 CLK 入力および出力のグリッチをチェックするため、IBIS シミュレーター (HyperLynx など) を使用して CCLK の分配をシミュレーションしてください。

図 2-22 ~ 図 2-25 に、CCLK 分配に使用する推奨トポロジを示します。

図 2-22 は、CCLK ドライバー (FPGA マスター) および CCLK レシーバー (PROM または FPGA スレーブ) が 1 つずつの基本的なポイント ツー ポイント トポロジです。

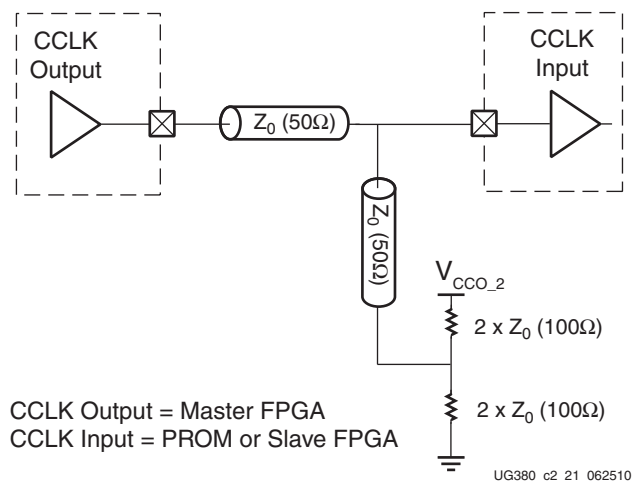


図 2-22 : ポイント ツー ポイント : 1 つの CCLK 出力、1 つの CCLK 入力

図 2-23 は、1 つの CCLK ドライバー、2 つの CCLK レシーバーの基本的なマルチドロップ フライ バイトポロジです。CCLK 入力 1 のスタブの長さに制約があります。

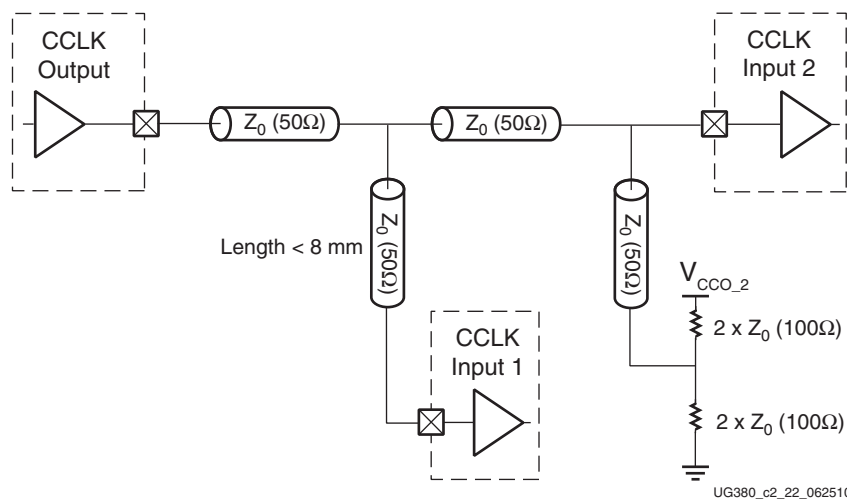


図 2-23： マルチドロップ： 1 つの CCLK 出力、2 つの CCLK 入力

図 2-24 は、1 つの CCLK ドライバーおよび CCLK レシーバーが 3 つ以上 (この例では 4 つ) のマルチドロップ フライ バイトポロジです。入力 4 を除く、すべての CCLK 入力のスタブの長さに制約があります。

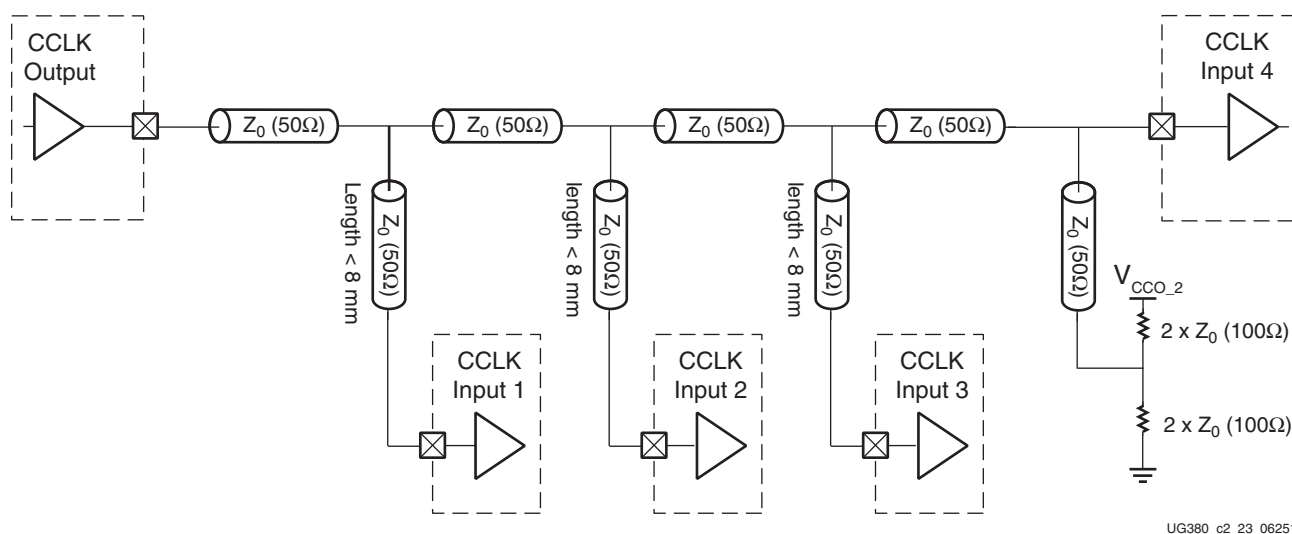
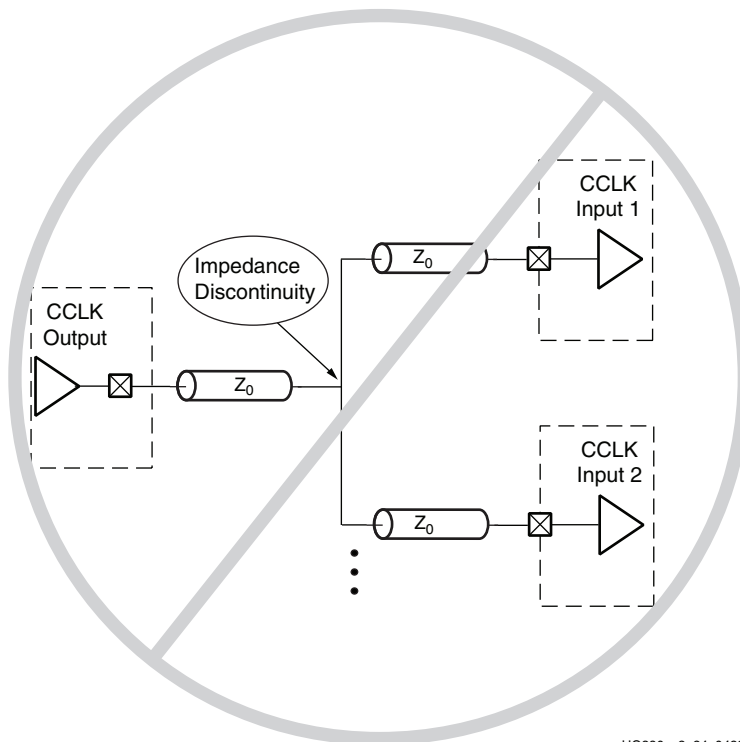


図 2-24： マルチドロップ： 1 つの CCLK 出力、3 つ以上の CCLK 入力

図 2-25 は、伝送ラインが複数の CCLK 入力に分岐しているスタート ポロジです。分岐点でインピーダンスの不連続が生じるため、このトポロジは推奨されていません。



UG380\_c2\_24\_042909

図 2-25 : 推奨されていないスタート ポロジ : 1 つの CCLK 出力、2 つの CCLK 入力



# バウンダリ スキャンおよび JTAG コンフィギュレーション

## 概要

Spartan®-6 デバイスでは、テスト アクセス ポート (TAP) およびバウンダリ スキャン アーキテクチャを定義する IEEE 1149.1 規格がサポートされています。この規格は、ボード レベルでの個々のコンポーネントの統合とそれらの接続を確実にするためのものです。バウンダリ スキャン アーキテクチャは接続テストだけでなくコンフィギュレーションや検証などベンダー固有の命令も柔軟にサポートするため、コンフィギュレーション データを直接 FPGA や互換性のある PROM に読み込む機能を追加できます。一般に、TAP とバウンダリ スキャン アーキテクチャを総称して JTAG と呼んでいます。

## IEEE 1149.1 を使用した Spartan-6 デバイスのバウンダリ スキャン

Spartan-6ファミリは IEEE1149.1 規格 (TAP およびバウンダリ スキャン アーキテクチャ) に完全に準拠しており、IEEE 1149.1 規格で定められている必須エレメントをすべて備えています。これらのエレメントとは、TAP (テスト アクセス ポート)、TAP コントローラー、命令レジスタ、命令デコーダー、バウンダリ レジスタ、バイパス レジスタです。Spartan-6 ファミリは、この規格に完全準拠する 32 ビットの ID レジスタもサポートしています。次のセクションから、Spartan-6 デバイスの JTAG アーキテクチャについて詳細に説明します。また、[第 10 章「アドバンス JTAG コンフィギュレーション」](#)も参照してください。

## テスト アクセス ポート (TAP)

Spartan-6 FPGA の TAP には、Spartan-6 デバイスおよび標準 JTAG アーキテクチャのプロトコルで指定されているように、4 つの必須の専用ピンがあります ([160 ページの図 10-1](#) 参照)。これら 3 つの入力ピンと 1 つの出力ピンを使用して、IEEE Std 1149.1 バウンダリ スキャンの TAP コントローラーを制御します。これらのオプションが他社製デバイスで駆動されている場合があるため、ザイリンクス製デバイスを他社製デバイスと組み合わせて使用する際は、オプション制御ピンの信号に注意してください。

IEEE 1149.1 規格のバウンダリ スキャン TAP コントローラーは、16 ステートのステート マシンです ([第 10 章「アドバンス JTAG コンフィギュレーション」](#) 参照)。

[表 3-1](#) では、IEEE 規格で必須と指定されている 4 つのピンについて説明します。

表 3-1 : Spartan-6 FPGA の TAP コントローラー ピン

ピン	方向	コンフィギュレーション 前の内部プル抵抗	説明
TDI	入力	プルアップ <sup>(1)</sup>	<b>Test Data In (テスト データ入力)</b> すべての JTAG 命令およびデータ レジスタのシリアル入力です。 ある動作時に TDI ピンがどのレジスタにデータを送るかは、TAP コントローラーのステートおよび現在の命令によって決まります。ピンが駆動されていないときにロジック High にするため、TDI には内部プルアップ抵抗があります。TDI は、TCK の立ち上がりエッジで JTAG レジスタに適用されます。
TDO	出力	プルアップ <sup>(1)</sup>	<b>Test Data Out (テスト データ出力)</b> すべての JTAG 命令およびデータ レジスタのシリアル出力です。 ある動作時にどのレジスタ (命令またはデータ) から TDO ピンにデータを送るかは、TAP コントローラー ステートおよび現在の命令によって決まります。TDO のステートは TCK の立ち下がりエッジで変化し、命令またはデータをデバイスで送信している場合のみアクティブになります。TDO はアクティブ ドライバー出力です。
TMS	入力	プルアップ <sup>(1)</sup>	<b>Test Mode Select (テスト モード選択)</b> TCK の立ち上がりエッジで、TAP コントローラーのステートのシーケンスを決定します。 ピンが駆動されていないときにロジック High にするため、TDI には内部プルアップ抵抗があります。
TCK	入力	プルアップ <sup>(1)</sup>	<b>Test Clock (テスト クロック)</b> JTAG のテスト クロック ピンです。 Spartan-6 デバイスの TAP コントローラーおよび JTAG レジスタは、TCK を基準に動作します。

## メモ：

1. すべての JTAG ピンには、コンフィギュレーション前に  $V_{CCAUX}$  へプルアップされる内部抵抗があります。これらの内部プルアップ抵抗は、選択されているモードに関係なくアクティブです。BitGen を使用してコンフィギュレーション後に 4 つの必須ピンすべてに対してプルアップ抵抗を有効にできます。詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) を参照してください。

## バウンダリ スキャン タイミング パラメーター

図 3-1 に示す一般的なタイミング パラメーターの特性評価データは、データシート [DS162](#) 『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』() の「コンフィギュレーション スイッチ特性」の表を参照してください。コンフィギュレーション フローの詳細については、[第 10 章「アドバンス JTAG コンフィギュレーション」](#)を参照してください。

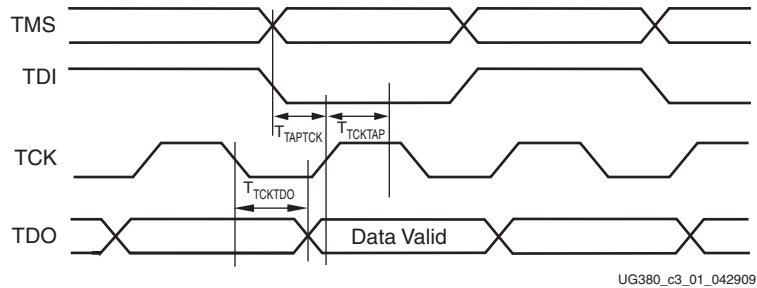


図 3-1 : Spartan-6 FPGA バウンダリ スキャン ポートのタイミング波形

## Spartan-6 デバイスでのバウンダリ スキャンの使用

ザイリンクスの iMPACT ソフトウェアで 1 つのデバイスをコンフィギュレーションする場合、TAP コントローラーのコマンドは自動的に発行されます。コンピューターのポートから Spartan-6 FPGA にビットストリームを自動的に転送するには、ダウンロード ケーブルを 4 つの JTAG ピン (TMS、TCK、TDI、および TDO) に正しく接続する必要があります。iMPACT は接続が正しいかどうかを自動的に確認し、送信コマンドやコンフィギュレーション ビットが適切に管理されていることを確認するコマンドを実行します。

図 3-2 は、1 つのデバイスと JTAG 信号ヘッダーをシンプルな形で接続した、一般的な JTAG 構成を示したものです。このヘッダーは、プロセッサで駆動するか、またはザイリンクスのプログラミング ケーブルを使用して iMPACT ソフトウェアで駆動できます。TCK はバウンダリ スキャンの動作で使用するクロックです。TDO - TDI 間の接続により、JTAG チェーンを介したデータ シフト用のシリアルデータパスが作成されます。TMS は、TAP コントローラーのステート間の遷移を制御します。第 10 章「アドバンス JTAG コンフィギュレーション」を参照してください。JTAG が正しく機能するには、これらの信号をすべて物理的に正しく接続する必要があります。

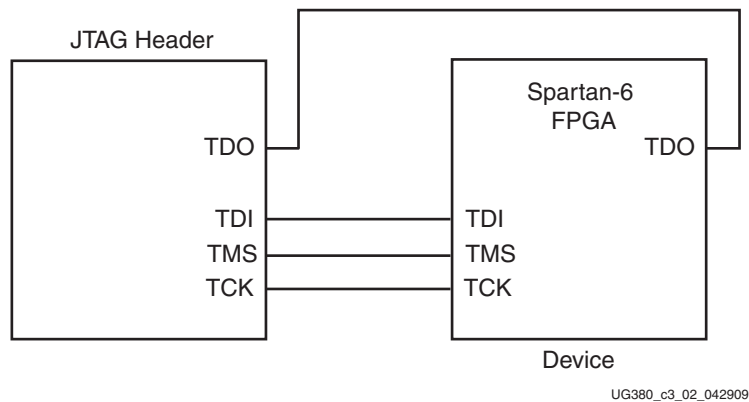


図 3-2 : 単一デバイスの JTAG プログラミング接続

### 複数デバイスのコンフィギュレーション

複数の Spartan-6 デバイスをチェーン接続してコンフィギュレーションすることもできます (図 3-3 参照)。

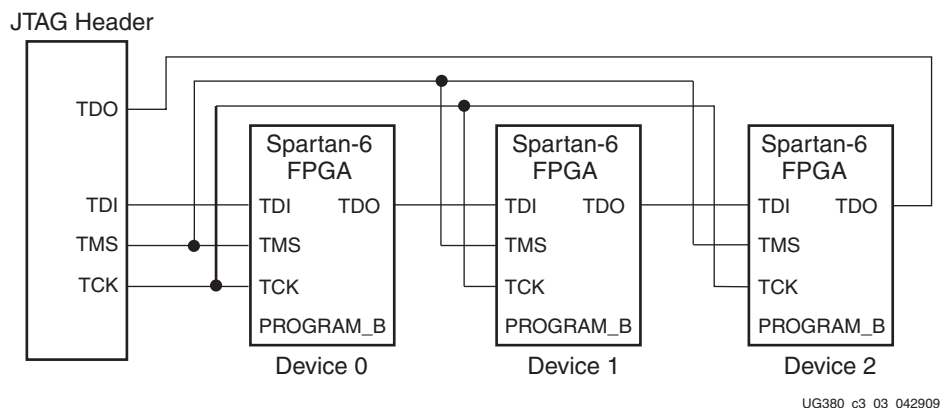


図 3-3： 複数デバイスのバウンダリ スキャン チェーン

コンフィギュレーション モードが JTAG のみの場合は、PROGRAM\_B、INIT\_B、および DONE を 330Ω 抵抗へ High に接続します。

JTAG チェーン内の複数のデバイスは 1 つずつコンフィギュレーションされます。複数デバイスのコンフィギュレーションは、シグナル インテグリティが良好に保たれている限り、チェーンの長さにかかわらず実行できます。iMPACT は、JTAG ヘッダーおよび iMPACT からの TDI 信号に最も近いデバイスから順にチェーン内のデバイスを自動的に検出します。

JTAG 入力には JTAG 動作の  $V_{CCAUX}$  が使用されます。

第 10 章「アドバンス JTAG コンフィギュレーション」は、さまざまな TAP コントローラー ステート、JTAG 命令、およびバウンダリ スキャン チェーン アーキテクチャの詳細を説明しています。

EXTEST、INTEST、および BYPASS などのバウンダリ スキャン命令の詳細は、IEEE 1149.1 規格および第 10 章「アドバンス JTAG コンフィギュレーション」を参照してください。

スタートアップ シーケンス、ビットストリーム、内部コンフィギュレーション レジスタの詳細は、第 5 章「コンフィギュレーションの詳細」および第 10 章「アドバンス JTAG コンフィギュレーション」を参照してください。

## 設計に関する考察事項

### JTAG 信号の配線

TCK および TMS 信号はチェーン内のすべてのデバイスに伝送されるため、信号の質が重要です。たとえば、JTAG 機能が正しく動作するには、TCK がすべてのレシーバーで均一に遷移する必要があります。また、TCK を正しく終端処理する必要もあります。TCK の質によっては、信頼性の高い JTAG コンフィギュレーションを確実にするために最大周波数が制限されることがあります。

さらに、チェーンが長い場合 (3 つ以上のデバイス)、すべてのレシーバーで駆動強度が十分であるようにするため、TMS および TCK にバッファを付ける必要があります。また、ロジック High での電圧はチェーン内のすべてのデバイスと互換性がある必要があります。

他社製品のデバイスにインターフェイスする際、TRST やイネーブルなどのオプション ピンに JTAG 信号が存在することがあり、これらを駆動する必要がある場合があります。



## 電源供給

電源投入時に適切に動作させるには、『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』に記載のガイドラインに従ってください。電源は、指定されている電源立ち上がり時間内に単調増加で立ち上がる必要があります。さらに、電圧は推奨動作範囲内とする必要があります。 $V_{DRINT}$  が  $V_{CCINT}$  を下回ったり、 $V_{DRAUX}$  が  $V_{CCAUX}$  を下回ると、コンフィギュレーション データが失われる可能性があります。

$V_{CCO\_2}$  および場合によっては  $V_{CCO\_1}$  で、コンフィギュレーション インターフェイス (SPI、Serial、BPI、および SelectMAP) の I/O 電圧が決定します。JTAG コンフィギュレーション ピンの I/O 電圧は  $V_{CCAUX}$  で決定します。提供された電圧は、使用されるすべてのコンフィギュレーション インターフェイスと互換性を持つ必要があります。

バウンダリ スキャンの機能を確実に動作させるには、未使用のシリアル トランシーバー タイルの電源投入に関するガイドラインに従ってください。

ガイドライン、範囲などすべての仕様は、データシート [DS162](#) 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』および『Spartan-6 FPGA GTP トランシーバー ユーザー ガイド』([UG386](#)) を参照してください。

## バウンダリ スキャンによるコンフィギュレーション

電源投入時に Spartan-6 デバイスが JTAG を介してコンフィギュレーションされる場合、JTAG 信号の動作によって現在のコンフィギュレーション モード設定がすべて上書きされます。

JTAG を使用する場合は Spartan-6 デバイス コンフィギュレーションのフローは、[第 10 章「アドバンス JTAG コンフィギュレーション」](#)で説明します。この章では、Spartan-6 デバイスを単一デバイスとして、または複数デバイスのバウンダリ スキャン チェーンの一部として、バウンダリ スキャンを使用してコンフィギュレーションする場合のコマンド シーケンスについて説明します。コンフィギュレーションされたデバイスをリコンフィギュレーションするには、TAP を切り替え、PROGRAM\_B をパルスした後に CFG\_IN 命令を入力するか、シャットダウン シーケンスを開始してください。詳細は、[第 10 章「アドバンス JTAG コンフィギュレーション」](#)を参照してください。

ザイリンクスは、プロトタイピング用に独自のプログラミング ケーブル (パラレルおよび USB) とバウンダリ スキャン プログラミング ソフトウェア (iMPACT) を提供しています。これらは、量産環境での利用を想定したものではありませんが、FPGA インプリメンテーションおよび JTAG チェーンの完全性の検証に非常に役立ちます。

JTAG チェーン内のほかのデバイスにアクセスしようとする場合、正しい数の先行する 1 または 0 でシフトし、各デバイスで正しい命令が受信できるよう、命令レジスタの長さを知っておくことが重要です。この情報は、ISE® ソフトウェアで提供される各デバイスの BSDL ファイルに記述されています。

バウンダリ スキャンを利用したベンダー固有の命令として最もよく使用されるものの 1 つに、コンフィギュレーション命令があります。Spartan-6 デバイスが JTAG を介してコンフィギュレーションされている場合、この命令はモード ピンに関係なく実行されます。JTAG を介したデバイス コンフィギュレーションの詳細は、[第 10 章「アドバンス JTAG コンフィギュレーション」](#)を参照してください。Spartan-6 FPGA JTAG コンフィギュレーション アルゴリズムでは SVF ベースのフローが使用されています。詳細は、アプリケーション ノート [XAPP058](#) 『組み込み型マイクロコントローラーを使用するザイリンクスのインシステム プログラミング機能』を参照してください。



## ユーザー プリミティブ

この章で説明するコンフィギュレーション プリミティブにより、ユーザーは、コンフィギュレーション中またはコンフィギュレーション後に FPGA コンフィギュレーション リソースにアクセスできます。インスタンス化テンプレートの詳細は、『Spartan-6 ライブラリ ガイド (HDL 用)』([UG615](#)) を参照してください。

### BSCAN\_SPARTAN6

JTAG は 4 つのピン(TCK、TMS、TDI、TDO)を持つ、標準的なインターフェイスです。多くのアプリケーションがこのインターフェイスに対応するように設計されています。JTAG TAP コントローラーは、コンフィギュレーション ロジック内部にある専用のステート マシンです。BSCAN\_SPARTAN6 により、JTAG TAP コントローラーとファブリックにあるユーザー ロジック間のアクセスが可能となります。BSCAN\_SPARTAN6 にはデバイスごとに最大 4 つのインスタンスがあります。各インスタンスで、JTAG\_CHAIN 属性で設定されている JTAG USER 命令 (USER1 から USER4) を 1 つ処理できます。4 つの USER 命令をすべて処理するには、これら 4 つの要素をインスタンス化し、JTAG\_CHAIN 属性を正しく設定する必要があります。表 4-1 に、BSCAN\_SPARTAN6 ポートの説明をまとめています。

表 4-1 : BSCAN\_SPARTAN6 ポートの説明

信号名	タイプ	機能
SEL	出力	JTAG 命令レジスタが対応する命令 (USER1、USER2、USER3、または USER4) を保持する場合は SEL = 1 です。Update_IR ステートで変更します。TAP コントローラーが Update_IR ステートのとき、TCK の立ち下がりエッジで変更します。
RESET	出力	アクティブ High のリセット出力です。TEST-LOGIC-RESET ステート中、PROGRAM_B、または電源投入中は、RESET = 1 です。TCK の立ち下がりエッジでデアサートされます。
TDI	出力	FPGA の TDI ピンから直接入力されます。
DRCK	出力	Capture_DR および Shift_DR ステートの TCK と同様に機能します。インターフェイスが命令レジスタで選択されていない場合は High を維持します。
CAPTURE	出力	アクティブ High のパルスで Capture_DR ステートであることを示します。TCK の立ち下がりエッジでアサートされます。
UPDATE	出力	アクティブ High のパルスで Update_DR ステートであることを示します。TCK の立ち下がりエッジでアサートされます。

表 4-1 : BSCAN\_SPARTAN6 ポートの説明 (続き)

信号名	タイプ	機能
SHIFT	出力	アクティブ High のパルスで Shift_DR ステートであることを示します。TCK の立ち下がりエッジでアサートされます。
RUNTEST	出力	Run Test/Idle ステートでは JTAG を示します。
TCK	出力	FPGA への TCK 入力ピンの値です。
TMS	出力	FPGA への TMS 入力ピンの値です。
TDO	入力	ユーザー ファブリック ロジックから駆動されます。FPGA の TDO への入力前に、立ち下がりエッジで内部でサンプルされます。

## ICAP\_SPARTAN6

ICAP\_SPARTAN6 プリミティブは、ファブリック側にあるという点を除いて SelectMAP コンフィギュレーション インターフェイスと同様に機能します。また、SelectMAP には双方向バスがあるのに対して ICAP には読み出し/書き込みに別々のバスがあります。ICAP は x16 のデータ幅のみをサポートしています。30 ページの「[SelectMAP コンフィギュレーション インターフェイス](#)」に示すように、一般的な SelectMAP のタイミング図および SelectMAP のビットストリーム順は、ICAP にも適用されます。これにより、コンフィギュレーション レジスタへのアクセスおよびコンフィギュレーション後のコンフィギュレーション データのリードバックが可能となります。

ICAP のデータ幅は、入力と出力のどちらも 16 ビットです。

表 4-2 : ICAP\_SPARTAN6 ポートの説明

信号名	タイプ	機能
CLK	入力	ICAP インターフェイス クロックです。
CE	入力	アクティブ Low の ICAP インターフェイス選択入力です。SelectMAP インターフェイスの CSI_B と同じです。
WRITE	入力	読み出し/書き込み制御入力です。0 = WRITE、1 = READ です。SelectMAP インターフェイスの RDWR_B と同じです。
I[15:0]	入力	16 ビット幅の ICAP 書き込みデータ幅バスです。ビット順は SelectMAP インターフェイスと同じです。詳細は、40 ページの「 <a href="#">SelectMAP のデータ順</a> 」を参照してください。
O[15:0]	出力	16ビット幅の ICAP 読み出しデータ幅バスです。ビット順は SelectMAP インターフェイスと同じです。40 ページの「 <a href="#">SelectMAP のデータ順</a> 」を参照してください。
BUSY	出力	アクティブ High の BUSY 信号です。読み出し操作でのみ使用され、書き込み中は Low のままになります。

## STARTUP\_SPARTAN6

STARTUP\_SPARTAN6 プリミティブはデバイスへのインターフェイスとして機能し、コンフィギュレーション後、グローバル信号の一部をユーザー制御可能にします。

表 4-3 : STARTUP\_SPARTAN6 ポートの説明

信号名	タイプ	機能
EOS	出力	アクティブ High であり、スタートアップの完全な終了を示します。
CLK	入力	ユーザー スタートアップ クロックです。
GSR	入力	アクティブ High のグローバル セット/リセット信号です。この入力のアサートされると、すべてのフリップフロップがビットストリームの初期値に戻ります。
KEYCLEARB	入力	設定されている場合、バックアップ バッテリーを使用する RAM のキーをクリアします。有効にするには 200ns (4 クロック サイクル) 間 Low に保持する必要があります。
GTS	入力	アクティブ High のグローバル トライステート信号です。この入力のアサートされると、すべてのユーザー I/O はトライステート状態となります。
CFGMCLK	出力	コンフィギュレーションの内部オシレーターのクロック出力です。レートは約 50MHz で、FPGA ロジックのリング オシレーターの代わりに、汎用クロック ソースとして使用できます。デザインで接続されていない場合、オシレーターは無効です。
CFGCLK	出力	コンフィギュレーション ロジックの主要クロックの出力です。現在のコンフィギュレーション モードに関連したクロックを出力します。FPGA がスレーブ コンフィギュレーション モードの場合、クロック ソースは CCLK です。FPGA がマスター コンフィギュレーション モードの場合、クロック ソースは内部オシレーター 周波数です (BitGen の <b>-g ConfigRate</b> オプションで定義されているとおり)。

## DNA\_PORT

DNA\_PORT を使用して専用シフト レジスタにアクセスできます。このシフト レジスタには、Spartan®-6 デバイスの Device DNA データ ビット (重複しない ID) が読み込まれます。DNA データ ビットをシフト アウトできるだけでなく、追加ユーザー データの補足データ ビットを含めたり、初期データのシフト アウト後に DNA データを繰り返すことができます。このコンポーネントは主に FPGA ビットストリームの不正なコピー防止用に、ほかの回路と合わせて使用されます。

DNA\_PORT は、インスタンス化する必要があります。インスタンス化テンプレートは ISE® の Project Navigator にある HDL テンプレート集に含まれています。インスタンスはコード内で宣言する必要があります。正しく動作するように、入力および出力をすべてデザインに接続してください。

Device DNA データにアクセスするにはまず、アクティブ High の READ 信号を 1 クロック サイクル間 High にしてシフト レジスタを読み込む必要があります。シフト レジスタが読み込まれると、アクティブ High の SHIFT 入力がいネーブルになり、DOUT 出力ポートからデータが取り込まれ、データが同期シフト アウトします。追加データがある場合は、適切なロジックを DIN ポートに接続すると、57 ビットのシフト レジスタの終わりに追加できます。DNA データをロールオーバーする (繰り返す) 場合は、DOUT ポートを直接 DIN ポートに接続しておくと、57 ビットのシフト操作の後で同じデータをシフト アウトできます。追加データが必要ない場合は、DIN ポートを論理値 0 に固定できます。属性 SIM\_DNA\_VALUE はオプションで DNA データ シーケンスをシミュレーションするように設定できます。デフォルトでは、シミュレーション モデルの Device DNA データ ビットはすべて 0 です。

表 4-4 : DNA\_PORT ポートの説明

信号名	方向	機能
DOUT	出力	シリアル シフト出力データです。
DIN	入力	シフト レジスタへのユーザー データ入力です。
READ	入力	Device DNA データとシフト レジスタの同期読み込みです。READ 操作が SHIFT 操作より優先されます。
SHIFT	入力	アクティブ High のシフト イネーブル入力です。
CLK	入力	シフト レジスタへの入力クロックです。

表 4-5 : DNA\_PORT 属性

属性	タイプ	値	デフォルト	説明
SIM_DNA_VALUE	57 ビットベクター	57'b0、 任意の 57 ビット値	すべてゼロ	シミュレーションに使用する DNA 値を指定します(実際の値はデバイスに特定の値)。

## SUSPEND\_SYNC

SUSPEND プリミティブによって、Suspend モードを使用するアプリケーションのデザインを同期化する機能が強化されます。このプリミティブは、3 ピンのインターフェイスを使用し、同期化が必要なクロックドメインが複数ある場合でも、Suspend モードを開始するトリガーを同期します。ここでは、SREQ、SACK、および CLK の 3 つの信号が使用されます。

SREQ はデバイスに Suspend モードを開始するリクエストを出力します。SACK は、デバイスが Suspend モードを開始できる状態であることを確認します。SACK ピンは CLK ピンに同期します。

表 4-6 : SUSPEND ポートの説明

信号名	方向	機能
CLK	入力	ユーザー クロックです。
SACK	入力	SUSPEND のアクノレッジ信号で、CLK に同期します。
SREQ	出力	SUSPEND ピンからの Suspend (一時停止リクエスト) です。

## POST\_CRC\_INTERNAL

POST\_CRC\_INTERNAL を使用して、デバイスから CRC 後のエラーにアクセスできます。

表 4-7 : POST\_CRC\_INTERNAL ポートの説明

信号名	方向	機能
CRCERROR	出力	コンフィギュレーション後のエラーです。エラーが検出されると High になります。





## コンフィギュレーションの詳細

### コンフィギュレーション ピン

コンフィギュレーション ピンには、コンフィギュレーション専用ピン (表 5-1) と多目的ピン (表 5-3) があります。多目的ピンは、コンフィギュレーション ピンとしてだけでなく、コンフィギュレーション後にユーザー I/O として機能します。コンフィギュレーション専用ピンは、コンフィギュレーション後もその機能を維持します。

コンフィギュレーションの制約は、Spartan®-6 FPGA のビットストリーム生成時に選択でき、制約によっては、コンフィギュレーションの動作に影響を与えることがあります。適用可能な制約の詳細は、ソフトウェアの『制約ガイド』を参照してください。

表 5-1 : Spartan-6 FPGA のコンフィギュレーション専用ピン

ピン名	タイプ <sup>(1)</sup>	説明
DONE	双方向、オープンドレイン、またはアクティブ	プログラム可能なプルアップを使用するアクティブ High の信号で、コンフィギュレーション完了を示します。  0 = FPGA のコンフィギュレーション未完了 1 = FPGA のコンフィギュレーション完了  ソフトウェア設定の詳細は、『コマンド ライン ツール ユーザー ガイド』(UG628) の「BitGen」を参照してください。
PROGRAM_B <sup>(2,3)</sup>	入力	プログラム可能なプルアップを使用するアクティブ Low の信号で、非同期フルチップ リセットです。
TDI	入力	Test Data In (テスト データ入力) すべての JTAG 命令およびデータ レジスタのシリアル入力です。ある動作時に TDI ピンがどのレジスタにデータを送るかは、TAP コントローラーのステートおよび現在の命令によって決まります。ピンが駆動されていないときにロジック High にするため、TDI には内部プルアップ抵抗があります。TDI は、TCK の立ち上がりエッジで JTAG レジスタに適用されます。
TDO	出力	Test Data Out (テスト データ出力) すべての JTAG 命令およびデータ レジスタのシリアル出力です。ある動作時にどのレジスタ (命令またはデータ) から TDO ピンにデータを送るかは、TAP コントローラー ステートおよび現在の命令によって決まります。TDO のステートは TCK の立ち下がりエッジで変化し、命令またはデータをデバイスで送信している場合のみアクティブになります。TDO はアクティブ ドライバー出力です。

表 5-1 : Spartan-6 FPGA のコンフィギュレーション専用ピン (続き)

ピン名	タイプ(1)	説明
TMS	入力	Test Mode Select (テスト モード選択) TCK の立ち上がりエッジで、JTAG TAP コントローラーのステートのシーケンスを決定します。ピンが駆動されていないときにロジック High にするため、TDI には内部プルアップ抵抗があります。
TCK	入力	Test Clock (テスト クロック) JTAG のテスト クロック ピンです。TAP コントローラーおよび JTAG レジスタは、TCK を基準に動作します。
SUSPEND <sup>(3)</sup>	入力	Suspend モード (一時停止モード) FPGA を Suspend モードにします。  電源投入中およびコンフィギュレーション中は、Low にする必要があります。 Suspend 機能を使用しない場合、GND に接続してください。
V <sub>FS</sub>	入力	eFUSE プログラミングの電源電圧です。(4)
V <sub>BATT</sub>	入力	AES 暗号化キーを格納する SRAM のバックアップ バッテリ電圧です。(4)
RFUSE	入力	eFUSE プログラミングのプルダウンです。(4)
CMPCS_B	予約済み	未接続またはプルアップ状態にします。

## 注記：

1. タイプに「双方向」と記載されている場合、任意の条件下で双方向に機能するピンです。あるコンフィギュレーション モードで入力として機能し、別のコンフィギュレーション モードでは出力として機能するピンは、タイプに「入力または出力」と記載されています。
2. PROGRAM\_B に信号を送信しても JTAG TAP ステート マシンはリセットされません。
3. すべての JTAG ピンおよび SUSPEND ピンの電源電圧は V<sub>CCAUX</sub> で、DONE および PROGRAM\_B の電源電圧は VCCO\_2 です。
4. XC6SLX75、XC6SLX75T、XC6SLX100、XC6SLX100T、XC6SLX150、および XC6SLX150T デバイスでのみ使用可能です。  
eFUSE プログラミングの詳細については、94 ページの「eFUSE」を参照してください。

## コンフィギュレーション中の FPGA I/O ピンの設定

コンフィギュレーション中に使用される一部の FPGA には、専用のプルアップ抵抗が付いていますが、すべてのユーザー I/O ピンには、オプションでコンフィギュレーション中に有効にできるプルアップ抵抗があります。コンフィギュレーション中、1 つの制御ラインでプルアップ抵抗を有効にするかどうかが決定されます。このピン名は HSWAPEN です (表 5-2 参照)。

表 5-2 : Spartan-6 FPGA のコンフィギュレーション ピンの終端

ピン	コンフィギュレーション前		コンフィギュレーション後
	HSWAPEN = 0 (有効)	HSWAPEN = 1 (無効)	
CCLK	VCCO_2 にプルアップ	終端なし	ユーザー I/O
D15-D0	VCCO_2 にプルアップ	終端なし	ユーザー I/O
CSO_B	VCCO_2 にプルアップ	終端なし	ユーザー I/O
A25-A0 <sup>(1)</sup>	VCCO_1 にプルアップ	終端なし	ユーザー I/O
SCP7-SCP0	VCCO_0 にプルアップ	終端なし	ユーザー I/O
DOUT/BUSY	VCCO_1 にプルアップ	終端なし	ユーザー I/O

表 5-2 : Spartan-6 FPGA のコンフィギュレーション ピンの終端 (続き)

ピン	コンフィギュレーション前		コンフィギュレーション後
	HSWAPEN = 0 (有効)	HSWAPEN = 1 (無効)	
HSWAPEN	VCCO_0 にプルアップ	VCCO_0 にプルアップ	ユーザー I/O
PROGRAM_B	VCCO_2 にプルアップ	VCCO_2 にプルアップ	BitGen の <b>-g ProgPin<sup>(2)</sup></b>
DONE	VCCO_2 にプルアップ	VCCO_2 にプルアップ	BitGen の <b>-g DonePin<sup>(2)</sup> -g DriveDone</b>
INIT_B	VCCO_2 にプルアップ	VCCO_2 にプルアップ	ユーザー I/O
TDI	VCCAUX にプルアップ	VCCAUX にプルアップ	BitGen の <b>-g TdiPin<sup>(2)</sup></b>
TMS	VCCAUX にプルアップ	VCCAUX にプルアップ	BitGen の <b>-g TmsPin<sup>(2)</sup></b>
TCK	VCCAUX にプルアップ	VCCAUX にプルアップ	BitGen の <b>-g TckPin<sup>(2)</sup></b>
TDO	VCCAUX にプルアップ	VCCAUX にプルアップ	BitGen の <b>-g TdoPin<sup>(2)</sup></b>
M1、M0	VCCO_2 にプルアップ	VCCO_2 にプルアップ	ユーザー I/O
FCS_B	VCCO_1 にプルアップ	終端なし	ユーザー I/O
FOE_B	VCCO_1 にプルアップ	終端なし	ユーザー I/O
FWE_B	VCCO_1 にプルアップ	終端なし	ユーザー I/O
MOSI/CSI_B	VCCO_2 にプルアップ	終端なし	ユーザー I/O
RDWR_B	VCCO_2 にプルアップ	終端なし	ユーザー I/O
AWAKE	VCCO_1 にプルアップ	終端なし	Suspend 機能が未使用の場合、 ユーザー I/O <sup>(4)</sup>
SUSPEND	VCCAUX にプルアップ	終端なし	SUSPEND ピン <sup>(3)(4)</sup>
HDC	VCCO_1 にプルアップ	終端なし	ユーザー I/O
LDC	VCCO_1 にプルアップ	終端なし	ユーザー I/O

注記：

1. A24/A25 は、FG676 以上のパッケージ サイズで、XC6SLX75/T およびそれ以上の高集積デバイスのバンク 5 にあります。プルアップは VCCO\_5 までです。
2. BitGen のオプションにより、対応するピンの終端を設定します。オプションの未設定でデフォルトのプルアップになります。ソフトウェア設定の詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) の「BitGen」を参照してください。
3. 電源投入中は、SUSPEND ピンは Low に設定しておく必要があります。これは、外部プルダウン抵抗の接続で確実に設定されます。
4. Suspend 機能の詳細は、『Spartan-6 FPGA パワー マネージメント ユーザー ガイド』([UG394](#)) を参照してください。

信号レベルをフロートさせておくと CMOS ロジック システムで問題になります。システムのほかのロジック コンポーネントには FPGA からの有効な入力レベルが必要な場合があります。内部プルアップ抵抗は各ピンをロジック High のレベルにします。一般的に、FPGA に信号を駆動するデバイスはこのプルアップ抵抗を超えます。同様に、適切に調整された外部プルダウン抵抗を使用して個々のピンをプルダウンできます。

ホットスワップやホット挿入アプリケーションでは、プルアップ抵抗で I/O の電源レールへの電流パスを提供します。プルアップ抵抗をオフにするとこのパスは無効になります。しかし、外部プルアップまたはプルダウン抵抗が各 I/O ピンに必要な場合があります。

## 多目的のコンフィギュレーション ピン (Persist) の予約

多目的ピンは、コンフィギュレーション ピンやコンフィギュレーション後のユーザー I/O として機能します。BitGen の **-g Persist** オプションを使用してこれらのピンをコンフィギュレーションピンとして予約します (設定については、表 5-3 を参照)。

表 5-3： 多目的のコンフィギュレーション ピンの設定

ピン名	バンク	SelectMAP	BPI	SPI/Serial
DIN/D0/MISO/MISO[1]	2	Persist	No	Persist
D1/MISO2	2	Persist	No	No
D2/MISO3	2	Persist	No	No
D[15:3]	2	Persist <sup>(1)</sup>	No	No
DOUT	1	Persist	No	Persist
INIT_B <sup>(2)</sup>	2	Persist <sup>(2)</sup>	No <sup>(2)</sup>	Persist <sup>(2)</sup>
RDWR_B	2	Persist	No	No
M0	2	No	No	No
M1	2	No	No	No
HSWAPEN	0	No	No	No
CCLK	2	Persist	No	Persist
GCLK0	2	No	No	No
CSO_B	2	No	No	No
MOSI/MISO0/CSI_B	2	Persist	No	No
AWAKE <sup>(3)</sup>	1	No	No	No
A[25:0] <sup>(4)</sup>	1	No	No	No
SCP[7:0] <sup>(3)</sup>	0	No	No	No
FCS_B	1	No	No	No
FOE_B	1	No	No	No
FWE_B	1	No	No	No
HDC	1	No	No	No
LDC	1	No	No	No

注記：

1. 16 のデータ ピンすべてが、SelectMAP データ幅 (x8 または x16) に関係なく Persist ピンとなります。
2. INIT\_B は、SEU インジケータ (POST\_CRC\_INIT\_FLAG) が有効の場合に Persist ピンとなります。
3. AWAKE および SCP[7:0] は、Suspend の設定に基づきアクティブになります。
4. A24 および A25 は、I/O バンクが 6 つ以上の大型デバイスの場合、バンク 5 にあります。

## コンフィギュレーション データ ファイルの形式

ザイリンクスのデザイン ツールでは、コンフィギュレーション データ ファイルをさまざまな形式で生成できます (表 5-4 参照)。BitGen ツールは、PAR 後の NCD ファイルをコンフィギュレーション ファイルまたはビットストリームに変換します。また、PROM ファイル生成ツールの PROMGen は、1 つ以上のビットストリーム ファイルを 1 つの PROM ファイルに変換します。さまざまな形式の PROM ファイルが生成可能であり、PROM で使用する必要はありません。これらは任意の場所に保存し、任意の手段で取り込むことができます。

表 5-4 : コンフィギュレーション ファイルの形式

ファイル 拡張子	ビット スワップ(1)	ザイリンクス ソフトウェア ツール(2)	説明
BIT	ビット スワップ なし	BitGen (デフォルトで生成)	FPGA へのダウンロードが不要なヘッダー情報を含むバイナリ コンフィギュレーション データ ファイルです。iMPACT とプログラミング ケーブルを使用してデバイスをプログラムする際に使用します。
RBT	ビット スワップ なし	BitGen (-b オプションの設定時に生成)	BIT ファイルを ASCII 形式としたもので、テキスト ヘッダーと ASCII 形式の 1 と 0 が含まれています (コンフィギュレーション ビットごとに 8 ビット)。
BIN	ビット スワップ なし	BitGen (-g Binary:yes オプションの設定時に生成) または PROMGen	ヘッダー情報がないバイナリ コンフィギュレーション データ ファイルで、BIT ファイルと類似しています。カスタム コンフィギュレーション ソリューション (マイクロプロセッサなど) またはサード パーティの PROM をプログラムする場合に使用します。
MCS EXO	ビット スワップ あり	PROMGen または iMPACT	コンフィギュレーション データ以外にアドレスおよびチェックサム情報を含む ASCII 形式の PROM ファイルです。主に、デバイス プログラマや iMPACT で使用します。
HEX	ユーザー指定	PROMGen または iMPACT	コンフィギュレーション データのみを含む ASCII 形式の PROM ファイルです。主に、カスタム コンフィギュレーション ソリューションで使用します。
CFI	なし	PROMGen または iMPACT	データ幅 x2 および x4 やバージョン コントロールなどの PROM オプションを設定するために iMPACT で使用されるデータ ファイルです。

注記 :

1. ビット スワップの詳細は、「[ビットのスワップ](#)」を参照してください。
2. BitGen および PROMGen 構文の詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) を参照してください。

## ビットストリームの概要

Spartan-6 データ幅 x2 および x4 やバージョン コントロールなどの PROM オプションを設定するために iMPACT で使用されるデータ ファイルです。表 5-5 に、各 Spartan-6 デバイスの標準的なビットストリームの長さを示します。

表 5-5 : Spartan-6 FFPGA のビットストリームの長さ

デバイス	コンフィギュレーション ビットの総数 <sup>(1)</sup>
XC6SLX4	2,724,832
XC6SLX9	2,724,832
XC6SLX16	3,713,568
XC6SLX25	6,411,696
XC6SLX25T	6,411,696
XC6SLX45	11,875,232
XC6SLX45T	11,875,232
XC6SLX75	19,624,736
XC6SLX75T	19,624,736
XC6SLX100	26,543,264
XC6SLX100T	26,543,264
XC6SLX150	33,761,696
XC6SLX150T	33,761,696

注記：

1. 標準的なビットストリーム長を示します。**Compress** などの BitGen オプションによっては、ビットストリーム長が変わる場合があります。x2 および x4 SPI コンフィギュレーション モードには追加コマンドが必要で、ビットストリームが長くなります。

Spartan-6 FPGA のビットストリームは、次の 2 つの部分で構成されています。

- 「同期ワード/バス幅の自動検出」
- FPGA コンフィギュレーション

## 同期ワード/バス幅の自動検出

パラレル コンフィギュレーション モードでは、バス幅はコンフィギュレーション ロジックによって自動的に検出されます。バス幅検出パターンには同期ワードが使用されます。コンフィギュレーション ロジックではパラレル バスで受信したデータがチェックされ、受信したバイト シーケンスに基づいて自動的に適切な外部バス幅に切り替えることができます。表 5-6 に、x16 モードのビットストリーム例を示します。詳しくは 81 ページの「パラレル バスのビット順」で説明しますが、FFPGA データ ピンに現れるビット パターンはスワップしています。

表 5-6 : x16 データのバス幅検出パターン

D[8:15]	D[0:7]	コメント
0xFF	0xFF	パッド ワード
0xFF	0xFF	パッド ワード
0xAA	0x99	同期ワード

表 5-6 : x16 データのバス幅検出パターン (続き)

D[8:15]	D[0:7]	コメント
0x55	0x66	同期ワード
...	...	...

通常、バス幅自動検出はユーザーが意識することはありません。

x8 バスの場合、コンフィギュレーション バス幅検出ロジックが D[0:7] ピンで最初に 0xAA、次に 0x99 を検出します。続いて 0x55 を検出し、次のサイクルで 0x66 を検出した場合、x8 モードで処理を続行します。x16 バスの場合、コンフィギュレーション バス幅検出ロジックは最初のバイトをチェックして D[0:7] ピンで 0x99 を検出します。その後、残りの同期ワードが上位ビットにあるため、次のサイクルで 0x66 を検出します。デバイスはそのまま x16 モードで処理を続行します。これで、FPGA はどのバス幅で残りのデータを受信するかを識別することができます。同期ワードが検出されるまで、FPGA はパケット処理を開始しません。詳細は、表 5-7 を参照してください。

表 5-7 : 同期ワード

31:24	23:16	15:8	7:0
0xAA	0x99	0x55	0x66

## PROM ファイルの生成

PROM ファイルは、PROMGen のユーティリティを使用してビットストリーム ファイルから生成します。PROMGen はコマンド ラインから直接実行することも、iMPACT のファイル生成モードで間接的に実行することもできます。PROMGen 構文の詳細は、『コマンド ライン ツール ユーザーガイド』(UG628) を参照してください。また、iMPACT の詳細は、ISE® ソフトウェアのマニュアルを参照してください。PROM ファイルは、ビットストリームを PROM プログラミング用の形式に変換し、シリアル デイジー チェーン用にビットストリーム ファイルを結合する場合に使用します(「シリアル デイジー チェーン用の PROM ファイル」参照)。

### シリアル デイジー チェーン用の PROM ファイル

シリアル デイジー チェーンは、個別の BIT ファイルを結合しただけではプログラムできないため、シリアル デイジー チェーン専用の形式でコンフィギュレーション データを用意する必要があります。PROMGen (または iMPACT) を使用すると、複数のビットストリームからシリアル デイジー チェーン専用形式の PROM ファイルを生成できます。PPROM ファイルを生成するには、PROMGen で複数のビットストリームを選択して **-n**、**-u**、**-d** オプションを指定するか、または iMPACT の File Generation Wizard を使用します。詳細は、ISE ソフトウェアのマニュアルを参照してください。

PROMGen は、下位デバイスのコンフィギュレーション データを上位デバイスのコンフィギュレーション パケットの中にネストし、ビットストリームを再フォーマットします。デイジー チェーンのコンフィギュレーションを実行する際、最上位デバイスに複数のビットストリームを送信すると最初のデバイスのみがコンフィギュレーションされ、以降のデータは無視されます。



## SelectMAP コンフィギュレーション用の PROM ファイル

MCS ファイルは、主に SelectMAP モードで 1 つの FPGA をプログラムする場合に用いるザイリンクス コンフィギュレーション PROM のプログラムに使用します。カスタム コンフィギュレーション ソリューションで最も手軽に利用できる PROM ファイル形式は、何も加工していない BIN および HEX 形式のファイルです。ただし場合によっては、ファイル形式の変更が必要になることがあります。詳細は、アプリケーション ノート [XAPP502](#) 『マイクロプロセッサを使用した、スレーブ シリアルまたは SelectMAP モードでの FPGA コンフィギュレーション』を参照してください。

SelectMAP コンフィギュレーション用の複数のコンフィギュレーション ビットストリームが 1 つのメモリ デバイスにある場合、それらのビットストリームを 1 つのシリアル デイジー チェーン PROM ファイルに結合するのではなく、複数の BIN または HEX ファイルでターゲットのメモリ デバイスをプログラムする必要があります。複数の個別データ ストリームを持つ PROM ファイルが 1 つ必要な場合は、iMPACT でパラレル PROM をターゲットにし、目的のデータストリーム数を選択して生成します。あるいは、PROMGen のコマンドラインから生成することもできます。詳細は、PROMGen ソフトウェアのマニュアルを参照してください。

## SPI/BPI コンフィギュレーション用の PROM ファイル

PROMGen の `-d`、`-u`、`-spi`、`-s`、および `-data_width` オプション、または iMPACT の File Generation Wizard を使用してサード パーティのフラッシュ デバイスに対応した PROM ファイルを作成できます。出力には、サード パーティ プログラマでサポートされている形式を選択することが重要です。BPI デバイスによっては、PROM ファイルをプログラムする際にエンディアンのスワップを有効にする必要があります。詳細は、フラッシュ デバイス ベンダーの資料を参照してください。

## ビットのスワップ

ビット スワップとは、バイト内でビット位置を入れ替えることです。ファイル形式が MCS、EXO、および TEK の RPOM ファイルでは常にビット スワップが行われます。ファイル形式が HEX の場合、ビット スワップを使用するかどうかはユーザー オプションで指定できます。ビット ストリーム ファイル (BIT、RBT、BIN) がビット スワップすることはありません。

HEX ファイル形式には、コンフィギュレーション データのみ含まれますが、その他の形式の PROM ファイルには、FPGA にはダウンロードされないアドレスおよびチェックサム情報が含まれます。このアドレスおよびチェックサム情報は、サード パーティのデバイス プログラマが使用するもので、PROM へはプログラムされません。



図 5-1 に、2 バイトのデータ (0xABCD) のビット スワップを示します。

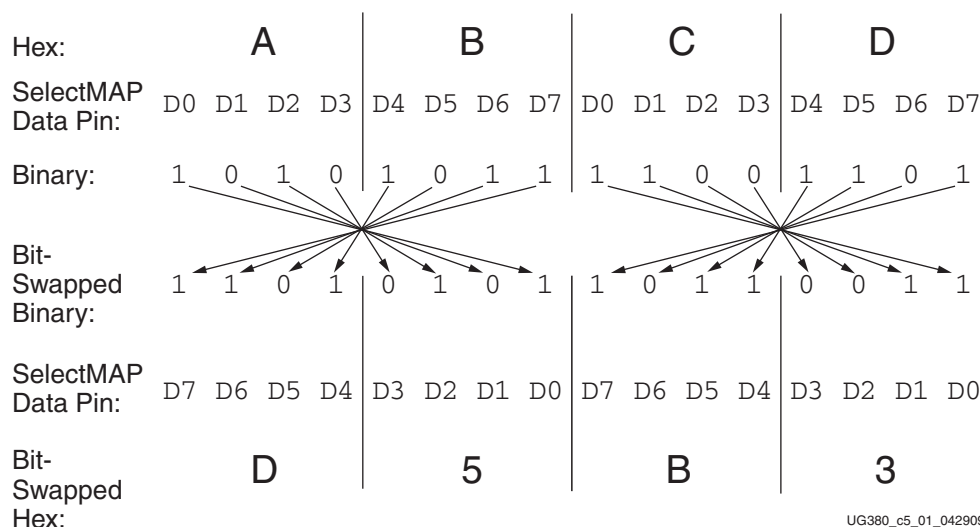


図 5-1 : ビット スワップの例

各バイトの MSB は、データの方に関係なく D0 ピンに割り当てられます。

- ビット スワップしたデータでは、右端のビットが D0 に送信されます。
- ビット スワップしていないデータでは、左端のビットが D0 に送信されます。

ビット スワップが必要かどうかは、完全にアプリケーションによって決定します。ビット スワップは、マスター シリアル、マスター SelectMAP、または BPI PROM ファイルで可能です。

## パラレル バスのビット 順

従来の SelectMAP x8 モードの場合、CCLK の 1 サイクルで 1 バイトのコンフィギュレーション データが読み込まれ、各バイトの MSB (最上位ビット) は D0 ピンに現れます。この D0 = MSB、D7 = LSB という規則はほかのデバイスには該当しませんが、ザイリンクスの FPGA はすべてこの規則に従っています。このビット スワップの規則は、Spartan-6 FPGA の BPI x8 モードにも該当します (80 ページの「ビットのスワップ」参照)。

Spartan-6 デバイスでは、ビット スワップ規則は x16 バス幅にも適用され、データは各バイト内でビット スワップされます。

表 5-8 および表 5-9 に、ビットストリーム内の同期ワード例を示します。この例では、スレーブ SelectMAP、マスター SelectMAP (BPI) モードなどのパラレル コンフィギュレーション モードを使用した場合、FPGA データ ピンに出力されるデータを示しています。

表 5-8 : 同期ワードのビット スワップ例

同期ワード	[31:24] <sup>(1)</sup>	[23:16]	[15:8]	[7:0]
ビットストリーム形式	0xAA	0x99	0x55	0x66
スワップしたビット	0x55	0x99	0xAA	0x66

注記：

- [31:24] は、ビット スワップ後に 0xAA から 0x55 に変化します。

表 5-9： x8、x16 モードでの同期ワードのデータ シーケンス例

CCLK サイクル	1	2	3	4
x8 モードの D[7:0] ピン	0x55	0x99	0xAA	0x66
x16 モードの D[15:0] ピン	0x5599	0xAA66		

### コンフィギュレーションの遅延

Spartan-6 デバイスでは、次の 2つの方法でコンフィギュレーションの開始を遅らせることができます。

- ・ 初期化中に INIT\_B ピンを Low に保持します。いったん INIT\_B が High になると、その後に INIT\_B を Low にしてもコンフィギュレーションを遅延させることはできません。
- ・ PROGRAM\_B ピンを Low に保持します。表 5-10 に、初期化およびコンフィギュレーションの遅延に関連する信号を示します。

表 5-10： 初期化およびコンフィギュレーションの遅延に関連する信号

信号名	タイプ	アクセス <sup>(1)</sup>	説明
PROGRAM_B	入力	PROGRAM_B ピンを介して外部からアクセス可能	グローバル非同期チップ リセットです。Low に保持してコンフィギュレーションを遅延させることができます。
INIT_B	入力、出力、またはオープンドレイン	INIT_B ピンを介して外部からアクセス可能	モード ピンのサンプル前は入力となり、Low に保持することでコンフィギュレーションを遅延させることができます。  モード ピンのサンプル後はオープン ドレインのアクティブ Low 出力となり、コンフィギュレーション中の CRC エラーまたはコンフィギュレーション後のリードバック CRC エラー (リードバック CRC が有効の場合) の有無を示します。  0 = CRC エラー 1 = CRC エラーなし (外部プルアップが必要)
MODE_STATUS[1:0]	ステータス	Spartan-6 FPGA のステータス レジスタを介してアクセス可能な内部信号	モード ピンの直接的な値を反映します。

#### 注記：

1. Spartan-6 FPGA ステータス レジスタの詳細は、108 ページの表 5-38 を参照してください。JTAG を介したデバイスのステータス レジスタへのアクセスについては、126 ページの表 6-5 を参照してください。SelectMAP を介したデバイスのステータス レジスタへのアクセスについては、表 6-1 を参照してください。
2. タイプがステータスの場合は、対応するピンのない内部ステータス信号です。

## コンフィギュレーション シーケンス

コンフィギュレーション インターフェイスにはいくつかの種類がありますが、Spartan-6 デバイスをコンフィギュレーションする基本的な手順はすべてのモードで共通です。図 5-2 に、Spartan-6 FPGA のコンフィギュレーション プロセスを示します。ここからは、各手順 (図中では灰色で表示) について詳しく説明します。

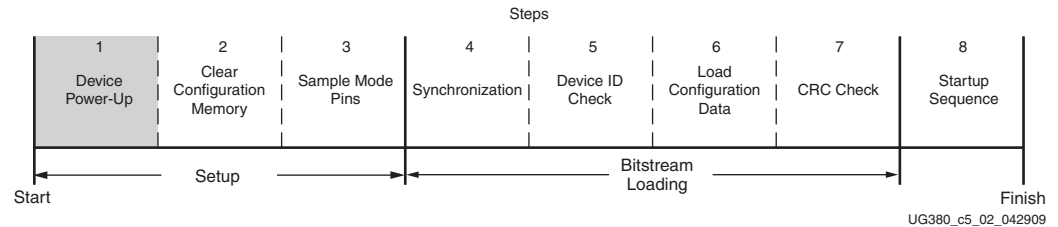


図 5-2 : Spartan-6 のコンフィギュレーション プロセス

最初の 3 つのセットアップ段階では、Spartan-6 デバイスを初期化し、モード ピンをサンプリングしてコンフィギュレーション モードを判断します。

### セットアップ (手順 1 - 3)

セットアップのプロセスは、すべてのコンフィギュレーション モードで同じです (図 5-3 参照)。

セットアップの各手順は、デバイスを正しくコンフィギュレーションするために非常に重要です。この手順では、1. デバイスの電源投入、2. コンフィギュレーション メモリのクリア、3. モード ピンのサンプリングを実行します。

#### デバイスの電源投入 (手順 1)

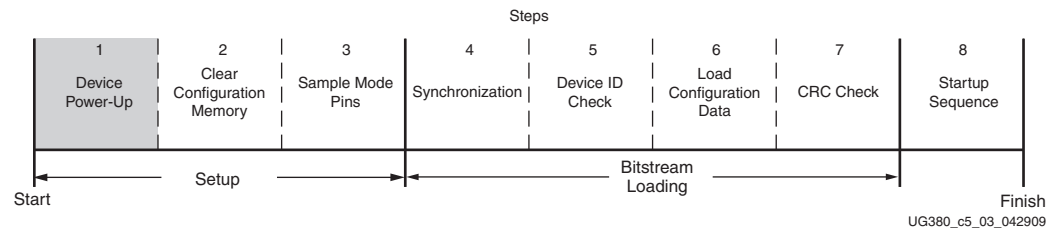


図 5-3 : デバイスの電源投入 (手順 1)

Spartan-6 デバイスをコンフィギュレーションするには、VCCO\_2、VCCAUX、および VCCINT ピンへの電源投入が必要です。電源投入シーケンスの要件はありません。

JTAG およびシリアル コンフィギュレーション ピンは、VCCAUX および VCCO\_2 に配置されています。多目的ピンはバンク 0、1、2 にあります (I/O バンクが 6 つ以上の大型デバイスの場合は例外で、A24 および A25 がバンク 5 にある)。DONE および PROG\_B 専用入力力は VCCO\_2 LVCMOS レベルで動作しますが、JTAG 入力ピン (TCK、TMS、TDI) と SUSPEND ピンは、VCCAUX LVCMOS レベルで動作します。DONE ピンは、出力規格が LVCMOS 8mA SLOW に設定されている VCCO\_2 の電圧レベルで動作します。TDO は VCCAUX で提供される電圧レベルで駆動します。

多目的 I/O を使用するモードでは、VCCO\_X をコンフィギュレーション デバイスの I/O 規格に一致した電圧に正しく接続する必要があります。VCCO\_X レベルによっては、コンフィギュレーション中ピンは LVCMOS18、LVCMOS25、または LVCMOS33 8mA SLOW で動作します。

電源投入の際には、-2/-3/-4 スピード グレードの場合は 1.2V の電源電圧を、-1L の場合は 1.0V の電源電圧を  $V_{CCINT}$  ピンに供給する必要があります。 $V_{CCO\_2}$  にも電源を供給してください。表 5-11 にコンフィギュレーションに必要な電源を示し、表 5-12 に電源投入に関するタイミング パラメーターを示します。

表 5-11： コンフィギュレーションに必要な電源

ピン名 <sup>(1)</sup>	説明
$V_{CCINT}$	内部コア電源
$V_{BATT}$ <sup>(2)</sup>	暗号化キーのバッテリー電源です。不揮発性メモリに暗号化キーがない場合、 $V_{BATT}$ は $V_{CC}$ または GND に接続する必要があります。
$V_{FS}$	暗号化キー eFUSE プログラミング電圧です。eFUSE プログラムが必要ない場合、 $V_{FS}$ は $V_{CCAUX}$ に接続する (推奨) か、または未接続にします。
$V_{CCAUX}$ <sup>(3)</sup>	コンフィギュレーション ロジックと FPGA のその他の機能向け補助電源です。
$V_{CCO\_0}$ $V_{CCO\_1}$ $V_{CCO\_2}$ <sup>(4)</sup> $V_{CCO\_5}$ <sup>(5)</sup>	多目的コンフィギュレーション ピンの出力電源電圧です。コンフィギュレーション中は $V_{CCO\_2}$ を 1.2V または 1.5V にできません。

注記：

1. 推奨動作値については、データシート [DS162](#) 『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』を参照してください。
2. ビットストリーム暗号化を使用する場合のみ  $V_{BATT}$  または  $V_{FS}$  が必要で、Spartan-6 LX75、LX75T、LX100、LX100T、LX150、および LX150T デバイスでのみサポートされています。
3. eFUSE プログラミング中は、 $V_{CCAUX}$  は  $V_{FS}$  と同等またはそれ以上にする必要があります。コンフィギュレーションではこの要件を満たす必要はありません。
4.  $V_{CCO\_2}$  が 1.8V の場合、 $V_{CCAUX}$  は 2.5V にする必要があります。 $V_{CCO\_2}$  が 2.5V または 3.3V の場合、 $V_{CCAUX}$  は 2.5V または 3.3V にできます。
5. BPI コンフィギュレーション モードが使用されていて、I/O バンク 5 に A24 および A25 がある場合、 $V_{CCO\_5}$  が必要な場合があります。

表 5-12： 電源投入に関連するタイミング パラメーター

説明	シンボル
プログラム レイテンシ	$T_{PL}$
POR (パワー オン リセット)	$T_{POR}$
CCLK 出力遅延	$T_{BPHICCK}$ または $T_{SPHICCK}$ <sup>(2)</sup>
プログラム パルス幅	$T_{PROGRAM}$

注記：

1. 電源投入時のタイミング特性の詳細は、『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』([DS162](#))を参照してください。
2. マスター セレクト MAP および BPI コンフィギュレーション インターフェイスの場合は  $T_{BPHICCK}$  を使用し、マスター シリアルおよび SPI コンフィギュレーション インターフェイスの場合は  $T_{SPHICCK}$  を使用してください。

図 5-4 に、電源投入時の波形を示します。

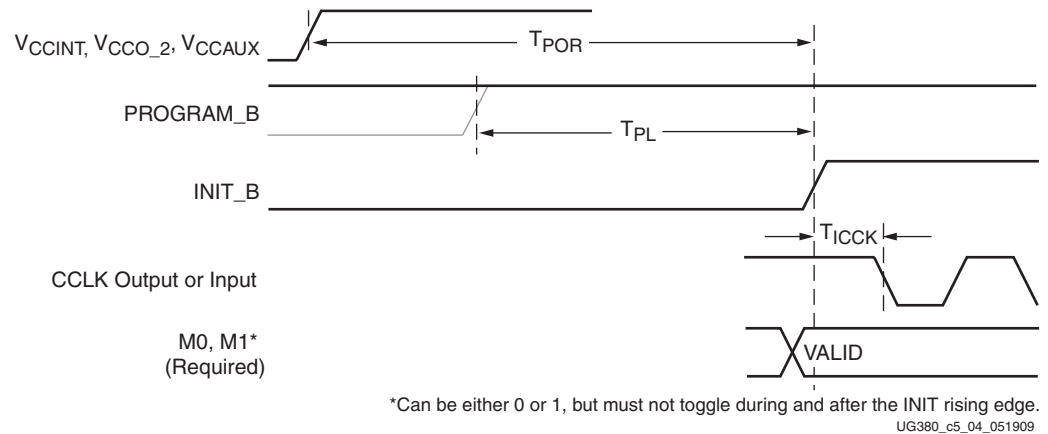


図 5-4：デバイスの電源投入タイミング

$V_{CCINT}$ 、 $V_{CCO\_2}$ 、および  $V_{CCAUX}$  は、指定された立ち上がり時間内で単調に増加させる必要があります。これが不可能な場合は、システム電源が推奨動作電圧値に到達するまで、 $INIT\_B$  ピンまたは  $PROGRAM\_B$  ピンを Low に保持してコンフィギュレーションの開始を遅らせてください (82 ページの「コンフィギュレーションの遅延」参照)。

$V_{CCO\_2}$ 、 $V_{CCAUX}$ 、および  $V_{CCINT}$  はパワーオン リセット (POR) へ入力されます。 $V_{CCAUX}$  または  $V_{CCINT}$  が最小動作電圧に達しないときは、POR を再度トリガーする必要がある場合があります。

## コンフィギュレーション メモリのクリア (手順 2、初期化)

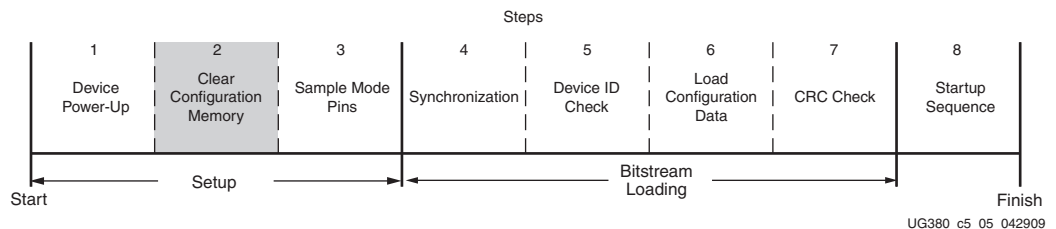


図 5-5：初期化 (手順 2)

コンフィギュレーション メモリは、デバイスへの電源投入後、 $PROGRAM\_B$  ピンの Low パルス後、JTAG JPROGRAM 命令または IPROG コマンドの使用後、あるいはフォールバック リコンフィギュレーション シーケンス中に逐次初期化されます。初期化の間、コンフィギュレーション専用ピンおよび JTAG ピン以外の I/O は、ハイインピーダンス状態となります。初期化中、 $INIT\_B$  は内部で Low に駆動され、電源投入時には  $T_{POR}$  後に (図 5-4)、それ以外の場合は  $T_{PL}$  後にリリースされます。 $INIT\_B$  を外部から Low に保持すると、デバイスはピンがリリースされるまで初期化プロセスのその時点の状態で待機します。

$PROGRAM\_B$  の最小 Low パルス時間は、 $T_{PROGRAM}$  タイミング パラメーターによって定義されます。 $PROGRAM\_B$  ピンは、必要な限りアクティブ (Low) に維持でき、 $PROGRAM\_B$  のリリース後にデバイスはコンフィギュレーション メモリを二度クリアします。

## モード ピンのサンプル (手順 3)

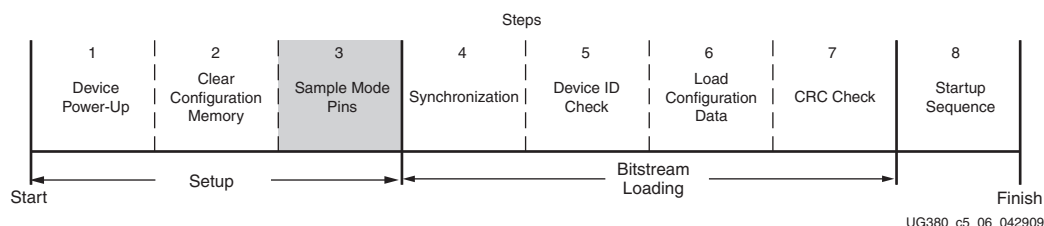


図 5-6：モード ピンのサンプル (手順 3)

INIT\_B ピンが High に遷移すると、デバイスはモード ピン M[1:0] をサンプルし、マスター モードの場合はコンフィギュレーション クロック (CCLK) の駆動を開始します。デバイスは CCLK の立ち上がりエッジでコンフィギュレーション データ入力ピンのサンプリングを開始します。

## ビットストリームの読み込み (手順 4 ~ 7)

ビットストリームの読み込みプロセスは、すべてのコンフィギュレーション モードでほぼ共通です。主な違いは、コンフィギュレーション ロジックへのインターフェイスです。これらの違いに関する詳細は、第 2 章「コンフィギュレーション インターフェイスの基本」で説明します。

## 同期化 (手順 4)

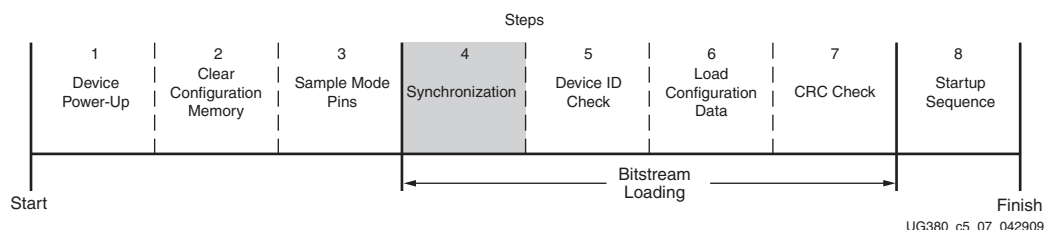


図 5-7：同期化 (手順 4)

同期ワードは、これからコンフィギュレーション データの送信が開始されることをデバイスに通知すると共に、内部のコンフィギュレーション ロジックがコンフィギュレーション データに揃うようにします。同期ワードよりも前のコンフィギュレーション入力ピン上のデータはすべて無視されます。

BitGen ソフトウェアで生成したコンフィギュレーション ビットストリーム (BIT ファイル) には必ずバス幅検出パターンと同期ワードの両方が含まれるため、同期をユーザーが意識することはほとんどありません。

## デバイス ID の確認 (手順 5)

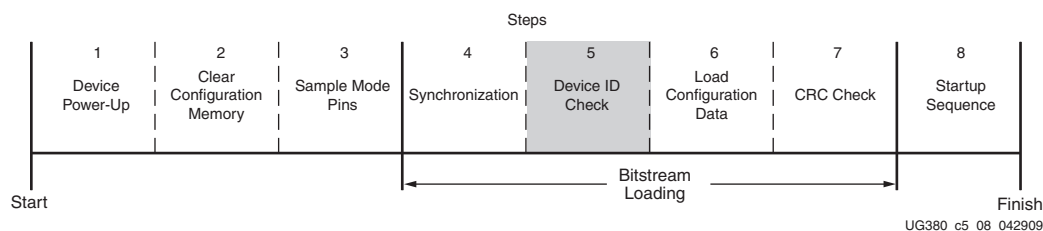


図 5-8 : デバイス ID の確認 (手順 5)

デバイスの同期後、コンフィギュレーション データ フレームを読み込む前にデバイス ID を確認する必要があります。これは、別のデバイス用にフォーマットされたビットストリームでコンフィギュレーションされることを防ぐためです。たとえば、デバイス ID チェックは、XC6SLX9 ビットストリームで XC6SLX4 がコンフィギュレーションされることを防ぎます。

コンフィギュレーション中に ID エラーが発生すると、デバイスはフォールバック リコンフィギュレーションの実行を試みます (136 ページの「フォールバック マルチブート」参照)。

デバイス ID チェックはビットストリームに組み込まれているため、この処理を設計者が意識することはほとんどありません。表 5-14 に、デバイスの ID チェックに関連する信号を示します。この場合のデバイス ID チェックは、JTAG IDCODE レジスタを介してではなく、コンフィギュレーション ロジックに対するビットストリーム内のコマンドによって実行されます。

Spartan-6 FPGA JTAG IDCODE のレジスタのフォーマットは次のとおりです。

vvvv:fffff:aaaaaaaa:cccccccccc1

説明 :

v = リビジョン

f = 7 ビットのファミリ コード

a = 9 ビットのアレイ コード (4 ビットのサブファミリおよび 5 ビットのデバイス ID)

c = 11 ビットの企業コード

表 5-13 : ID コード

デバイス	ID コード (16 進数)
XC6SLX4	0xX4000093
XC6SLX9	0xX4001093
XC6SLX16	0xX4002093
XC6SLX25	0xX4004093
XC6SLX25T	0xX4024093
XC6SLX45	0xX4008093
XC6SLX45T	0xX4028093
XC6SLX75	0xX400E093
XC6SLX75T	0xX402E093
XC6SLX100	0xX4011093

表 5-13 : ID コード (続き)

デバイス	ID コード (16 進数)
XC6SLX100T	0xX4031093
XC6SLX150	0xX401D093
XC6SLX150T	0xX403D093

注記：

1. 4 つのバイナリ リビジョン ビットに対応する ID コードの X 桁は、IDCODE 検証中にプログラミング ツールで使用されません。

表 5-14 : デバイス ID チェックに関連する信号

信号名	タイプ	アクセス <sup>(1)</sup>	説明
ID_Error	ステータス	内部信号。Spartan-6 FPGA ステータス レジスタを介してのみアクセス可能	ビットストリームで指定されたデバイス ID と実際のデバイス ID の不一致を示します。

注記：

1. Spartan-6 FPGA ステータス レジスタの詳細は、表 5-35 を参照してください。JTAG を介したデバイスのステータス レジスタへのアクセスについては、表 6-5 を参照してください。SelectMAP を介したデバイスのステータス レジスタへのアクセスについては、表 6-1 を参照してください。

## コンフィギュレーション データ フレームの読み込み (手順 6)

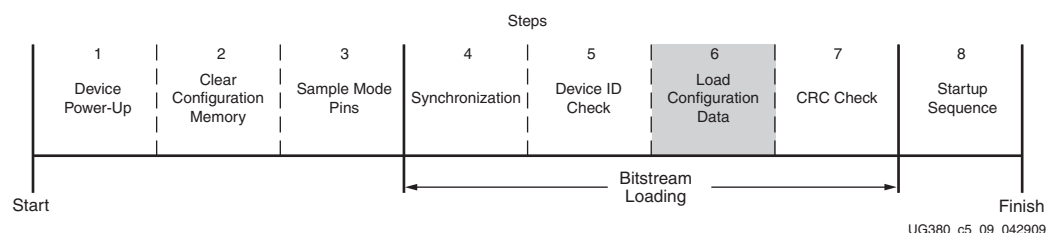


図 5-9 : コンフィギュレーション データ フレームの読み込み (手順 6)

同期ワードの読み込みとデバイス ID の確認が完了すると、コンフィギュレーション データ フレームが読み込まれます。このプロセスをユーザーが意識することはほとんどありません。

## CRC (Cyclic Redundancy Check) (手順 7)

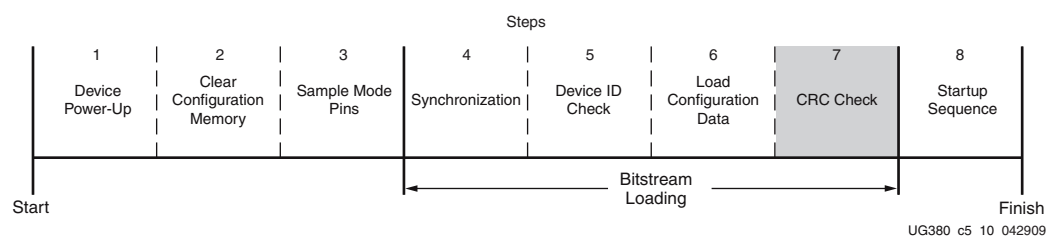


図 5-10 : CRC (Cyclic Redundancy Check) (手順 7)

コンフィギュレーション データ フレームの読み込み時に、デバイスはコンフィギュレーション データ パケットから CRC 値を算出します。データ フレームの読み込み後、コンフィギュレーション ビットストリームはデバイスに対して CRC を確認する命令を与え、続いて、正しい CRC 値を送信します。デバイスが算出した CRC 値がビットストリーム内の正しい CRC 値と一致しない場



合、デバイスは INIT\_B を Low 駆動し、コンフィギュレーションを中断します。CRC チェックはデフォルトでコンフィギュレーション ビットストリームに含まれますが、不要であれば無効にできます。詳細は、『コマンド ライン ツール ユーザー ガイド』(UG628) の「BitGen」を参照してください。CRC チェックを無効にすると、誤ったコンフィギュレーション データ フレームを読み込み、デザインでの不適切な動作あるいはデバイスの損傷を引き起こす可能性があります。

FPGA がマスターとなっているモードでのコンフィギュレーションで CRC エラーが発生すると、フォールバック リコンフィギュレーションが実行されます (136 ページの「フォールバック マルチブート」参照)。

## スタートアップ (手順 8)

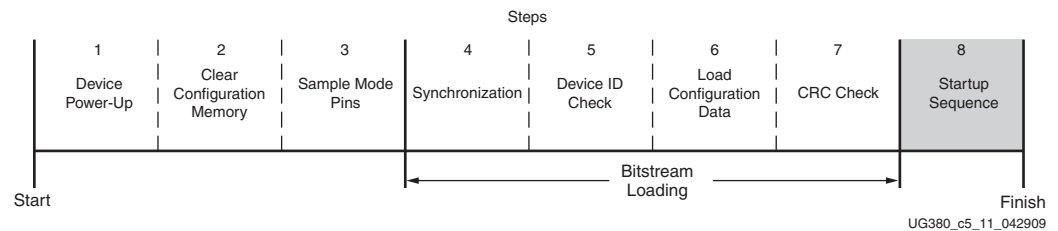


図 5-11: スタートアップ シーケンス (手順 8)

コンフィギュレーション フレームの読み込みが完了すると、スタートアップ シーケンスの開始命令がビットストリームからデバイスに与えられます。スタートアップ シーケンスは、8 フェーズ (0 ~ 7) のシーケンシャル ステート マシンで制御されます。つまり、JTAG クロックまたは BitGen の **-g StartupCLK** オプションで定義された任意のユーザー クロックが供給されます。スタートアップ シーケンサーは、表 5-15 に示すタスクを実行します。

表 5-15: スタートアップ イベントのユーザー選択可能なサイクル

フェーズ	イベント
1 ~ 6	DCM および PLL がロックするまで待機します (オプション)。
1 ~ 6	GWE (グローバル ライト イネーブル) をアサートし、RAM およびフリップフロップの状態を変更可能にします。
1 ~ 6	GTS (グローバル トライステート) をディアサートし、I/O を有効にします。
1 ~ 6	DONE ピンをリリースします。
7	EOS (End Of Startup) をアサートします。

EOS のアサートを除くスタートアップ イベントの順序は、BitGen のオプションでユーザーがプログラムできます (『コマンド ライン ツール ユーザー ガイド』(UG628) 参照)。表 5-15 に、一般的なイベントの順序を示します。実際のスタートアップ イベントの各フェーズはユーザーでプログラム可能ですが、EOS は常に最後のフェーズでアサートされます。スタートアップ オプションのガイドラインは、第 2 章「コンフィギュレーション インターフェイスの基本」を参照してください。表 5-16 に、デフォルト設定でのスタートアップ イベントを示します。

表 5-16: スタートアップ イベントのデフォルト BitGen シーケンス

フェーズ	イベント
4	DONE ピンをリリースします。
5	GTS をディアサートし、I/O を有効にします。

表 5-16： スタートアップ イベントのデフォルト BitGen シーケンス (続き)

フェーズ	イベント
6	GWE をアサートし、RAM とフリップフロップの状態を変更可能にします。
7	EOS をアサートします。

適切な BitGen オプションを使用すると、スタートアップ シーケンスで DCM および PLL がロックするまで待機させることができます。通常、これらのオプションは、DCM および PLL のロックの前に、DONE、GTS、および GWE がアサートされないように設定します。

スタートアップは、LCK\_CYCLE オプションをスタートアップ フェーズに指定することで DCM および PLL を待機できます。これが実行されない場合、DCM または PLL を待機しません。LCK\_CYCLE がスタートアップ フェーズに設定されると、FPGA は次のスタートアップ フェーズへ移行する前にすべての DCM および PLL がロックするまで待機します。特定 DCM のみのロックを待機するには、STARTUP\_WAIT 属性をそれらのインスタンスに割り当てます。PLL には対応する属性がありません。マスター コンフィギュレーション モードでの LCK\_CYCLE 機能の使用については、140 ページの「マルチブート イメージ間に必要なデータ空間」を参照してください。

DONE 信号は、ユーザーが指定したサイクルでスタートアップ シーケンスによってリリースされますが、実際に DONE ピンがロジック High になるまで、スタートアップは続行されません。DONE ピンは、デフォルトでは内部プルアップ抵抗のあるオープン ドレインの双方向信号です。DONE ピンがリリースされると、デバイスはロジック Low 駆動が停止して、ピンは弱いプルアップで High に駆動されます。表 5-17 にスタートアップ シーケンスに関連した信号を示し、図 5-12 にその波形を示します。

表 5-17： スタートアップ シーケンスに関連する信号

信号名	タイプ	アクセス <sup>(1)</sup>	説明
DONE	双方向 <sup>(2)</sup>	DONE ピン または Spartan-6 FPGA ステータス レジスタ	コンフィギュレーションが完了したことを示します。外部で Low を維持し、スタートアップをほかの FPGA と同期させることができます。
GWE	ステータス	Spartan-6 FPGA ステータス レジスタ	グローバル ライト イネーブル (GWE) です。ディアサートされると、FPGA の CLB、IOB フリップフロップ、およびその他の同期エレメントが無効になります。
GTS			グローバル トライステート (GTS) です。アサートされると、コンフィギュレーション ピン用を除く、すべての I/O ドライバーが無効になります。
DCM_LOCK			すべての DCM および PLL がロックしたことを示します。この信号は、デフォルトでアサートされています。DCM で STARTUP_WAIT オプションを使用し、ビットストリームの生成時に LCK_CYCLE オプションを使用した場合にアクティブになります。

## 注記：

1. Spartan-6 FPGA ステータス レジスタの詳細は、105 ページの表 5-35 を参照してください。JTAG を介したデバイスのステータス レジスタへのアクセスについては、「126 ページの表 6-5」を参照してください。SelectMAP を介したデバイスのステータス レジスタへのアクセスについては、「121 ページの表 6-1」を参照してください。
2. デフォルトでは内部プルアップ抵抗のあるオープン ドレイン出力ですが、BitGen の DriveDone オプションを有効にすることもできます。

スレーブ コンフィギュレーション モードの場合、スタートアップ イベントを完了させるには、DONE が High になった後に追加クロックが必要です。マスター コンフィギュレーション モードでは、FPGA がこれらのクロックを供給します。必要なクロック数は、スタートアップ イベントに選択された設定によって異なります。一般的な規則では、DONE が High になった後に 8 つのクロック (DIN がすべて 1) が適用されます。DCM および PLL のロック (LCK\_CYCLE) まで待機するようにスタートアップが設定されている場合、さらにクロックが必要です。

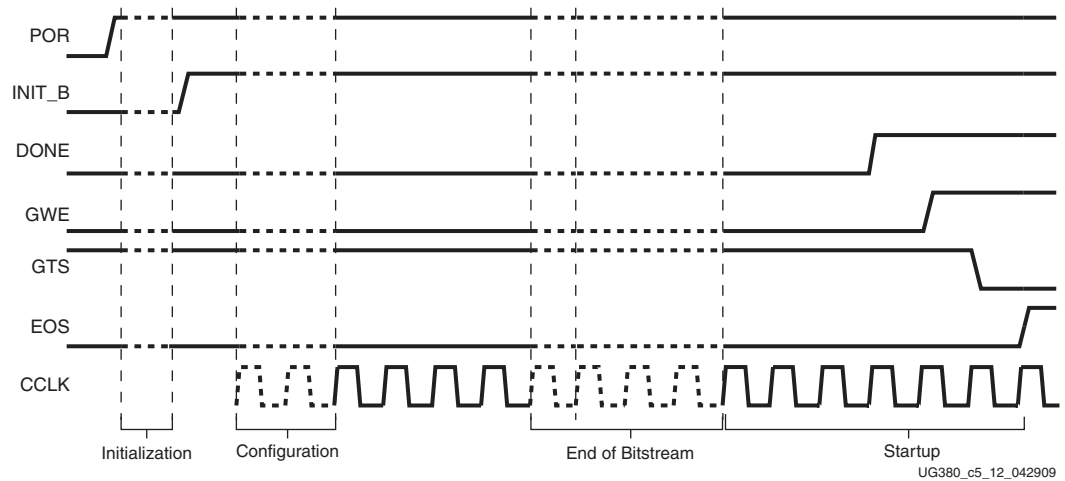


図 5-12 : コンフィギュレーション信号のシーケンス (デフォルト設定でのスタートアップの場合)

## ビットストリーム暗号化

Spartan-6 の XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスには、オンチップ AES (Advanced Encryption Standard) 複合化ロジックがあり、デザインの高い安全性を実現しています。外部の者がビットストリームを不正入手したとしても、暗号化キーを知らなければ、ビットストリームが解読されることもデザインが複製されることもありません。したがって、暗号化された Spartan-6 FPGA デザインのコピーやリバース エンジニアリングは不可能です。暗号化は、データ幅が x1 および x8 のコンフィギュレーション モードで使用できます (JTAG を含む)。

Spartan-6 FPGA の AES システムは、ソフトウェア ベースのビットストリーム暗号化機能、およびオンチップのビットストリーム複合化機能と暗号化キーを格納する専用メモリのための専用メモリで構成されています。ユーザーは、ISE ソフトウェアを使用して、暗号化キーと暗号化したビットストリームを生成できます。Spartan-6 デバイスでは、この暗号化キーをデバイス内部の専用 RAM (外部の小型バックアップ バッテリーに接続) または eFUSE のいずれかに格納できます。暗号化キーは、JTAG インターフェイスを介してのみデバイスにプログラムできます。一度プログラムすると、JTAG だけでなくどんな方法でも、暗号化キーをデバイスから読み出すことはできません。

コンフィギュレーション実行中、Spartan-6 デバイスでは、反対の処理、つまり取り込まれているビットストリームの復号化が行われます。Spartan-6 FPGA の AES 暗号化ロジックは、256 ビットの暗号化キーを使用します。

オンチップの AES 復号化ロジックは、ビットストリームの復号化以外の目的には使用できません。たとえば、ユーザー デザインでは AES 復号化ロジックは使用できず、コンフィギュレーションビットストリーム以外のデータは復号化できません。

## AES の概要

Spartan-6 FPGA の暗号化システムは、AES 暗号化アルゴリズムを使用します。AES は、NIST (National Institute of Standards and Technology) および米国商務省が認証する公式規格です (<http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>)。

Spartan-6 FPGA の AES 暗号化システムは 256 ビットの暗号化キーを使用し (NIST が定める 128 ビットおよび 192 ビットの暗号化キーはインプリメントしない)、一度に 128 ビットのデータ ブロックを暗号化または復号化します。NIST によると、256 ビット キーの場合、その組み合わせは  $1.1 \times 10^{77}$  とおり考えられます。

AES のような対称暗号化アルゴリズムでは、暗号化と復号化に同じキーが使用されます。したがって、データの安全性はキーの安全性に依存しています。

## 暗号化したビットストリームの作成

ISE ソフトウェアで提供されている BitGen では、暗号化したビットストリームと暗号化していないビットストリームのどちらも作成できます。ビットストリームを AES 暗号化するには、BitGen の入力として 256 ビットのキーを指定します。これにより、BitGen で暗号化ビットストリーム ファイル (BIT) および暗号化キーファイル (NKY) が生成されます。

BitGen のコマンドおよび構文の詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) を参照してください。

## 暗号化キーの読み込み

暗号化キーは、JTAG インターフェイスを介してのみ Spartan-6 デバイスに読み込むことができます。ISE ソフトウェアで提供されている iMPACT ツールで NKY ファイルを入力ファイルとし、ザイリンクスの USB-II プログラミング ケーブルを使用して JTAG 経由でデバイスに暗号化キーをプログラムできます。

暗号化キーをプログラムする際、ISC\_PROGRAM\_KEY 命令を使用するとデバイスが特別なキー アクセス モードになります。この命令では、暗号化キー専用 RAM とコンフィギュレーション メモリを含むすべての FPGA メモリがクリアされます。暗号化キーがプログラムされ、キー アクセス モードが終了すると、いかなる方法でもデバイスから暗号化キーを読み出すことはできず、キーを再プログラムするにはデバイス全体を初期化する必要があります。このキーは、eFUSE にプログラムした後、再プログラムできません。

## 暗号化ビットストリームの読み込み

デバイスに適切なキーをいったんプログラムすると、暗号化ビットストリームを使用したデバイス コンフィギュレーションが可能になります。暗号化ビットストリームでコンフィギュレーションを行った後は、BitGen のセキュリティ設定にかかわらず、JTAG または SelectMAP リードバックによってコンフィギュレーション メモリを読み出すことはできません。

デバイスに暗号化キーが取り込まれた状態のときに、暗号化していないビットストリームでデバイスをコンフィギュレーションすることができます。この場合、キーは無視されます。また、暗号化していないビットストリームでコンフィギュレーションを行った後は、リードバックが可能です (BitGen のセキュリティ設定で許可されている場合のみ)。この場合でも、デバイスから暗号化キーを読み出すことはできないため、「トロイの木馬」ビットストリームを使用して Spartan-6 FPGA の暗号化システムを無効にすることはできません。

暗号化を使用しても、コンフィギュレーションの方法には影響を与えることはありません。データ幅が x1 または x8 のコンフィギュレーション モード (シリアル、JTAG、BPI、SelectMAP) で、コンフィギュレーション ビットストリームを受け取ることができます。SPI x2、SPI x4、BPI x16、および SelectMAP x16 のバス幅は暗号化ビットストリームではサポートされていません。また、コンフィギュレーションのタイミングおよびシグナリングは、暗号化の影響を受けません。

コンフィギュレーション後にリコンフィギュレーションするには、PROGRAM\_B ピンをトグルする、電源を再投入する、あるいは JPROGRAM 命令を与える必要があります。暗号化をオンにすると、フォールバック リコンフィギュレーションおよび IPROG リコンフィギュレーション (136 ページの「フォールバック マルチブート」参照) が無効となります。また、ICAP プリミティブを使用したリードバックが可能です (「ビットストリームの暗号化および内部コンフィギュレーション アクセス ポート (ICAP)」参照)。V<sub>BATT</sub> または V<sub>CCAUX</sub> が維持されている場合、これらのイベントによってキーがリセットされることはありません。

暗号化ビットストリーム内のキーとデバイスに格納されたキーが一致しないと、コンフィギュレーションがエラーとなり、INIT\_B ピンが Low になって、DONE ピンが Low のままになります。

## ビットストリームの暗号化および内部コンフィギュレーション アクセスポート (ICAP)

内部コンフィギュレーション アクセス ポート (ICAP) プリミティブを使用すると、ユーザー ロジックから Spartan-6 FPGA コンフィギュレーション インターフェイスにアクセスできます。ICAP インターフェイスは SelectMAP インターフェイスとほとんど同じですが、コンフィギュレーション後のリードバックの制限が ICAP インターフェイスにはありません。ICAP であれば、ビットストリームの暗号化を使用した場合でもリードバックを実行できます。ICAP インターフェイスをユーザー I/O に配線しない限り、ICAP を使用することで外部から Spartan-6 FPGA の AES 暗号化システムが侵害されることはありません。

デザインのセキュリティに万全を期すには、次の点に注意してください。

- ICAP インターフェイスをユーザー I/O に配線しない

または

- ICAP プリミティブをインスタンス化しない

ほかのコンフィギュレーション インターフェイスと同様、ICAP インターフェイスからキー レジスタにアクセスすることはできません。

## V<sub>BATT</sub>

暗号化キー メモリ セルは揮発性であり、その内容を維持するには、電源が継続して投入されている必要があります。通常の動作中、これらのメモリ セルには補助電圧入力 (V<sub>CCAUX</sub>) から電源が供給されますが、V<sub>CCAUX</sub> が切断されると、キーを保持するために V<sub>BATT</sub> 電源入力から電源が供給されます。V<sub>BATT</sub> にはほとんど電流が流れないため (nA 程度)、この電源には小さなボタン電池が適しています。バッテリーの寿命を推定するには、『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』記載の V<sub>BATT</sub> の DC 特性およびバッテリーの仕様を参照してください。150nA 未満の負荷では、バッテリーの使用可能期間はバッテリーの寿命期間になるはずですが。

V<sub>BATT</sub> には電流が流れないため、V<sub>CCAUX</sub> から電源を供給中に切断できます。V<sub>BATT</sub> は、V<sub>CCAUX</sub> 切断時に暗号化キーを保持する以外の目的には使用できません。

## eFUSE

ヒューズ リンクは、一定期間に大量の電流を流すことでプログラムします。FUSE プログラム用の電流は、固定された外部電源供給 ( $V_{FS}$  ピン) から供給されます。最大レベルは、内部で生成された電源で制御されます。また、eFUSE は OTP (ワンタイム プログラマブル) です。

通常、プログラムしたヒューズ リンクはプログラムしていない状態の抵抗値よりもその値が数桁大きくなります。プログラムしたヒューズの論理値は 1 で、プログラムしていないヒューズの論理値は 0 です。

FUSE\_KEY および FUSE\_CNTL レジスタの各論理ビットは、2 つの eFUSE セル (プライマリ、リダundant)、1 つのフリップフロップ、およびデータ マルチプレクス用の共通ロジック エレメントで構成されています。

### eFUSE レジスタ

Spartan-6 FPGA には、全部で 3 つの eFUSE レジスタがあります。表 5-18 に、Spartan-6 デバイスの eFUSE レジスタとそれぞれのサイズおよび用途を説明します。eFUSE ビットのアドレスが指定されると、LSB 順にシフトイン/シフトアウトされて最後に MSB がきます。

表 5-18 : eFUSE レジスタ

レジスタ名	サイズ (ビット)	内容	説明
FUSE_KEY <sup>(1)</sup>	256	ビットストリーム暗号化 キー [0:255]  (ビット 255 が最初にシフト される)	AES ビットストリームの復号化で使用するキーを格納します。 バックアップ パッテリが必要な SRAM にキーを格納する代わりに、eFUSE にキーを格納して使用できます。  Spartan-6 FPGA の復号化エンジンは、この AES キーを使用して暗号化ビットストリームを読み込みます。AES キーは、FUSE_CNTL レジスタの読み出し/書き込みアクセス ビットの設定に基づき、JTAG ポートを介して読み出しまたはプログラムが可能です。
FUSE_ID	57	Device DNA [0:56]  (ビット 56 が最初にシフト される)	Device DNA を格納します。JTAG ポートや DNA_PORT プリミティブを介してアクセスする読み出し専用レジスタです。
FUSE_CNTL <sup>(1)</sup>	32	制御ビット CNTL [31:0]  (ビット 0 が最初にシフト される)	キーの使用や eFUSE レジスタへの読み出し/書き込みアクセスを制御します。このレジスタは、JTAG ポートを介して読み出しまたはプログラムが可能です。

注記：

1. FUSE\_KEY および FUSE\_CNTL は、XC6SLX75/T、XC6SLX100/T、および XC6SLX150/T デバイスでのみ使用できます。

### eFUSE 制御レジスタ (FUSE\_CNTL)

このレジスタには、6 つのユーザー プログラマブル ビットが含まれています。これらのビットを使用して、表 5-19 に示すように AES キーの使用法や eFUSE レジスタの読み出し/書き込み保護を設定します。ビット 0 が最初にシフト イン/シフト アウトされます。



eFUSE ビットは OTP (ワンタイム プログラマブル) です。つまり、一度プログラムすると再プログラムはできません。たとえばレジスタへのアクセスを無効にした場合、後から有効にすることはできません。

表 5-19 : eFUSE CNTL レジスタ ビット

ビット番号	名前	説明	注釈
0:7	-	-	予約
8	CNTL セキュリティ	CNTL レジスタの読み出しと書き込みを無効にします。CNTL[12] でリダンダント	AES および CNTL レジスタの操作やリードバックを防止するため、ユーザーはこれらのレジスタをプログラムおよび検証した後にこのビットをプログラムする必要があります。
9	-	-	予約
10	キー セキュリティ	KEY レジスタの読み出しと書き込みを無効にします。CNTL[14] でリダンダント。	AES レジスタの操作やリードバックを防止するため、ユーザーはこれらのレジスタをプログラムおよび検証した後にこのビットをプログラムする必要があります。
11	-	-	予約
12	CNTL セキュリティ	CNTL レジスタの読み出しと書き込みを無効にします。CNTL[8] でリダンダント。	AES および CNTL レジスタの操作やリードバックを防止するため、ユーザーはこれらのレジスタをプログラムおよび検証した後にこのビットをプログラムする必要があります。
13	-	-	予約
14	キー セキュリティ	KEY レジスタの読み出しと書き込みを無効にします。CNTL[10] でリダンダント。	AES レジスタの操作やリードバックを防止するため、ユーザーはこれらのレジスタをプログラムおよび検証した後にこのビットをプログラムする必要があります。
15	-	-	予約
16	aes_exclusive	パーシャル リコンフィギュレーションを無効にします。	次のいずれかの方法でコンフィギュレーション前に FPGA の内容をクリアすることを要求します : JPROG JTAG 命令の発行、PROGRAM_B ピンのパルス、または FPGA への電源再投入。
17	cfg_aes_only	このビットがプログラムされた後、FPGA は、eFUSE レジスタに格納された AES キーを使用してのみコンフィギュレーション可能です。	eFUSE AES レジスタに格納された AES キーで暗号化されたビットストリームでのみ FPGA のコンフィギュレーションが可能です。
18:31	-	-	予約

CNTL[17] をプログラムしない場合は次のことが可能です。

- 暗号化は、BitGen オプションで無効/有効にできる
- eFUSE に格納されている AES キーを使用するか、バックアップ バッテリーを必要とする SRAM に格納されている AES キーを使用するかを BitGen のオプションで選択できる

CNTL[17] をプログラムした場合、eFUSE に格納されたキーで暗号化したビットストリームのみを FPGA のコンフィギュレーションに使用できます。

CNTL[16] をプログラムした場合、最初のコンフィギュレーション後にコンフィギュレーション メモリがブロックされます。デバイスをリコンフィギュレーションするには、電源の再投入、JPROGRAM または IPROG コマンドの発行、PROGRAM\_B ピンのパルスのいずれかが必要になります。

## JTAG 命令

eFUSE レジスタは JTAG ポートから読み出すことができます。eFUSE のプログラムは、JTAG を介してのみ可能です。表 5-20 に、eFUSE に関連する JTAG 命令を示します。標準的な JTAG 通信プロトコルの詳細は、第 10 章「アドバンス JTAG コンフィギュレーション」を参照してください。これらの命令は、eFUSE をプログラムするのに十分ではありません。適切なアルゴリズムが使用されますが、提供されていません。唯一サポートされている eFUSE のプログラミング方法は、IMPACT ソフトウェアの使用によるものです。

表 5-20 : eFUSE 関連の JTAG 命令

JTAG 命令	コード	アクション
FUSE_KEY	6'h3B	256 ビットの FUSE_KEY レジスタを選択します。
FUSE_OPTION	6'h3C	eFUSE とのインターフェイスで使用されるデータおよびコマンドに対応する 16 ビットの FUSE_OPTION レジスタを選択します。
ISC_FUSE_READ	6'h30	DNA eFUSE レジスタを選択します。ISC_ENABLE の後に実行し、それから ISC_DISABLE を実行する必要があります。
FUSE_UPDATE	6'h3A	AES および CNTL の eFUSE からの値で FPGA を更新します。
FUSE_CNTL	6'h34	32 ビットの FUSE_CNTL レジスタを選択します。

## V<sub>FS</sub> ピン

Spartan-6 デバイスの V<sub>FS</sub> ピンは、2 つの eFUSE 動作専用ピンのうちの 1 つです。V<sub>FS</sub> ピンは、電源投入の上昇率や ESD ストレスなどを検証するための電源供給として使用されます。

プログラム中の V<sub>FS</sub> ピンの電圧仕様は、公称値で 3.3V です。プログラム中の電流に対しては、最大 40mA を供給できるようにする必要があります。読み出しモードの場合、V<sub>FS</sub> ピンは唯一 V<sub>CCAUX</sub> の最大動作条件より低くなる必要があります。V<sub>FS</sub> バイアスの条件については、表 5-21 を参照してください。完全な仕様は、データシート [DS162](#) 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』() を参照してください。



表 5-21 :  $V_{FS}$  ピンのバイアス条件

eFUSE モード	$V_{FS}$ ピンのバイアス
読み出しまたは未使用	$V_{CC}/GND$ (推奨) または未接続
プログラム	3.3V

## RFUSE ピン

RFUSE ピンは、eFUSE 動作に使用される 2 つ目の専用ピンです。eFUSE のプログラムが必要な場合は、1,140 $\Omega$  抵抗を GND に接続してください。1,140 $\Omega$  抵抗を得ることが難しい場合は、10W 抵抗と直列の 1,130W 抵抗に置き換えることが可能です。許容抵抗については、データシート [DS162](#) 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』を参照してください。プログラムしていないときに eFUSE を使用するか、または eFUSE を使用しない場合、RFUSE を  $V_{CCAUX}$  に接続またはグラウンド接続することを推奨します。あるいは、RFUSE はフロートさせておくことが可能です。

## $V_{CCAUX}$ ピン

$V_{CCAUX}$  は、eFUSE のプログラム時に  $V_{FS}$  と同等またはそれ以上にする必要があります。 $V_{CCAUX}$  は、eFUSE からの読み出しまたはコンフィギュレーション時にほかの推奨動作値をとることが可能です。詳細は、データシート [DS162](#) 『Spartan-6 FPGA データシート : DC 特性およびスイッチ特性』を参照してください。

## コンフィギュレーション メモリ フレーム

Spartan-6 FPGA のコンフィギュレーション メモリは、フレームに分割され、デバイスに並べられています。Spartan-6 FPGA のコンフィギュレーション メモリ空間では、フレームがアドレス指定可能な最小単位であり、すべての操作はコンフィギュレーションフレーム全体に対して実行することになります。Spartan-6 FPGA のフレーム数およびコンフィギュレーション サイズは、[表 5-22](#) に記載されています。BitGen のオプションによっては、コンフィギュレーション ビットストリームにオーバーヘッドが追加されることがあります。ビットストリームの正確な長さは、RBT (rawbits) ファイルに記述されています。このファイルは、BitGen の **-b** オプションを指定するか、または ISE ソフトウェアのポップアップ メニューで [Generate Programming File] → [Create ASCII Configuration File] をクリックすると生成されます。ビットストリーム長 (ワード) は、コンフィギュレーション アレイ サイズ (ワード) にコンフィギュレーション オーバーヘッド (ワード) を足したものにほぼ等しくなります。ビットストリーム長 (ビット) は、ビットストリーム長 (ワード) に 32 を掛けたものにほぼ等しくなります。

表 5-22 : デバイス リソース

デバイス	ブロック RAM 列	コアの フレーム数	ブロック RAM のフレーム数	I/O 数	I/O の フレーム長	ビット総数
XC6SLX4	1	794	9360	120	577	2,724,832
XC6SLX9	2	2028	37440	200	897	2,724,832
XC6SLX16	2	2976	37440	244	1,073	3,713,568

表 5-22： デバイス リソース (続き)

デバイス	ブロック RAM 列	コアの フレーム数	ブロック RAM のフレーム数	I/O 数	I/O の フレーム長	ビット総数
XC6SLX25	3	5065	70200	264	1,153	6,411,696
XC6SLX25T	3	5065	70200	264	1,153	6,411,696
XC6SLX45	4	9088	149760	370	1,577	11,875,232
XC6SLX45T	4	9088	149760	370	1,577	11,875,232
XC6SLX75	4	15384	224640	426	1,801	19,624,736
XC6SLX75T	4	15384	224640	426	1,801	19,624,736
XC6SLX100	6	20304	336960	498	2,089	26,543,264
XC6SLX100 T	6	20304	336960	498	2,089	26,543,264
XC6SLX150	6	27240	336960	576	2,401	33,761,696
XC6SLX150 T	6	27240	336960	576	2,401	33,761,696

## コンフィギュレーション パケット

Spartan-6 FPGA ビットストリーム コマンドはすべて、コンフィギュレーション レジスタに対する読み出しまたは書き込みという形で実行されます。コンフィギュレーション データは 16 ビットワードで構成されています。複数のワードで構成されるデータもあります。コンフィギュレーション データには、NOP、READ、WRITE の 3 つの主なコマンドがあります (表 5-23 参照)。

表 5-23： オペコードのフォーマット

OP	CODE
NOP	00
READ	01
WRITE	10

コンフィギュレーション コマンドは、該当するコマンド レジスタから読み出されるか、または書き込まれるときに実行されます。

## パケットのタイプ

すべてのデータ (レジスタの書き込みおよびフレーム データ) は、2 種類のパケットにまとめられます。

- タイプ 1 パケット：ヘッダーおよびデータの 2 つのセクションに分かれています。
- タイプ 2 パケット：ヘッダー、ワード カウント、およびデータの 3 つのセクションに分かれています。

## タイプ 1 パケット

タイプ 1 パケットは、短いブロックの 6 アドレス ビットのレジスタ書き込みに使用されます。ヘッダー セクションは常に 16 ビット ワードです。詳細は、表 5-24 を参照してください。

表 5-24 : タイプ 1 ヘッダー パケット

ヘッダー	タイプ	動作	レジスタ アドレス	ワード カウント
ビット	[15:13]	[12:11]	[10:5]	[4:0]
タイプ 1	001	xx	xxxxxxx	xxxxxx

タイプ 1 パケットのヘッダー セクションの後には、タイプ 1 データ セクションが続きます。このセクションに含まれる 16 ビット ワードの数をヘッダーのワード カウント部で宣言します。詳細は、表 5-25 を参照してください。

表 5-25 : タイプ 1 データ セクション

データ	[15:0]
ワード	xxxxxxxxxxxxxxxx

## タイプ 2 パケット

タイプ 1 に後続するタイプ 2 のパケットは、長いブロックの書き込みに使用されます。このパケットにはタイプ 1 パケットのアドレスが使用されるため、アドレス部はありません。ヘッダー セクションは、常に 16 ビット ワードです。

タイプ 2 パケットのヘッダー セクションの後には、タイプ 2 データ セクションが続きます。このセクションに含まれる 16 ビット ワードの数をヘッダーのワード カウント部で宣言します。

表 5-26 : タイプ 2 パケット ヘッダー

ヘッダー	タイプ	動作	レジスタ アドレス	(未使用)
ビット	[15:13]	[12:11]	[10:5]	[4:0]
タイプ 2	010	xx	xxxxxxx	00000

タイプ 2 パケットのヘッダーの後にはタイプ 2 ワード カウント セクションが続きます。このセクションには 2 つの 16 ビット ワードが含まれていて、最初のワードに MSB が含まれています。

表 5-27 : タイプ 2 パケット ワード カウント データ 2

WC1	[31:16]
データ	0000xxxxxxxxxxxx

表 5-28 : タイプ 2 パケット ワード カウント データ 1

WC2	[15:0]
データ	xxxxxxxxxxxxxxxx

タイプ 2 ワード カウント セクションの後には、タイプ 2 データ セクションが続きます。このセクションに含まれる 16 ビット ワードの数をヘッダーのワード カウント部で宣言します。

表 5-29： タイプ 2 パケット データ セクション

データ	[15:0]
ワード [1]	XXXXXXXXXXXXXXXXXX
...	XXXXXXXXXXXXXXXXXX
ワード [wc]	XXXXXXXXXXXXXXXXXX

ワード カウント = (フレーム総数 + 1 ダミー フレーム) x 実際のフレーム長

## コンフィギュレーション レジスタ

表 5-30 に、コンフィギュレーション レジスタを示します。それに続いて、各レジスタの詳細について説明します。

表 5-30： コンフィギュレーション レジスタ

レジスタ名	R/W	アドレス	説明
CRC	W	6'h00	CRC (Cyclic Redundancy Check)
FAR_MAJ	W	6'h01	フレーム アドレス レジスタ ブロックおよびメジャー
FAR_MIN	W	6'h02	フレーム アドレス レジスタ マイナー
FDRI	W	6'h03	フレーム データ入力
FDRO	R	6'h04	フレーム データ出力
CMD	R/W	6'h05	コマンド
CTL	R/W	6'h06	制御
MASK	R/W	6'h07	制御マスク
STAT	R	6'h08	ステータス
LOUT	W	6'h09	シリアル デイジー チェーンのレガシ出力
COR1	R/W	6'h0a	コンフィギュレーション オプション 1
COR2	R/W	6'h0b	コンフィギュレーション オプション 2
PWRDN_REG	R/W	6'h0c	パワーダウン オプション レジスタ
FLR	W	6'h0d	フレーム長レジスタ
IDCODE	R/W	6'h0e	製品 IDCODE
CWDT	R/W	6'h0f	コンフィギュレーション ウォッチドッグ タイマー
HC_OPT_REG	R/W	6'h10	ハウス クリーン オプション レジスタ
CSBO	W	6'h12	パラレル デイジー チェーンの CSB 出力
GENERAL1	R/W	6'h13	パワーアップ セルフ テストまたは読み込み可能なプログラム アドレス

表 5-30 : コンフィギュレーション レジスタ (続き)

レジスタ名	R/W	アドレス	説明
GENERAL2	R/W	6'h14	パワーアップ セルフ テストまたは読み込み可能なプログラム アドレスおよび新しい SPI オペコード
GENERAL3	R/W	6'h15	ゴールドデン (エラーのない) ビットストリーム アドレス
GENERAL4	R/W	6'h16	ゴールドデン (エラーのない) ビットストリーム アドレス および新しい SPI オペコード
GENERAL5	R/W	6'h17	フェールセーフ用ユーザー定義レジスタ
MODE_REG	R/W	6'h18	リブート モード
PU_GWE	W	6'h19	一時停止からのウェークアップ中の GWE サイクル
PU_GTS	W	6'h1a	一時停止からのウェークアップ中の GTS サイクル
MFWR	W	6'h1b	マルチフレーム書き込みレジスタ
CCLK_FREQ	W	6'h1c	マスター モードの CCLK 周波数セレクト
SEU_OPT	R/W	6'h1d	SEU 周波数、イネーブル、およびステータス
EXP_SIGN	R/W	6'h1e	SEU 検出に予想されるリードバックシグネチャ
RDBK_SIGN	R/W	6'h1f	リードバック コマンドおよび SEU のリードバックシグネチャ
BOOTSTS	R	6'h20	ブート履歴レジスタ
EYE_MASK	R/W	6'h21	マルチピン ウェークアップのマスク ピン
CBC_REG	W	6'h22	初期 CBC 値レジスタ

## CRC レジスタ

CRC (Cyclic Redundancy Check) レジスタは、標準の 32 ビット CRC チェックサム アルゴリズムを使用し、コンフィギュレーション中のビットストリームを検証します。書き込まれた値が現時点で計算された CRC 値と一致すると、CRC\_ERROR フラグがクリアされてスタートアップが可能となります。

## FAR\_MAJ レジスタ

フレーム アドレス レジスタは、次のコンフィギュレーション データ入力を開始するブロックおよび列アドレスを設定します。Spartan-6 FPGA のパーシャル リコンフィギュレーションでは、開始列の最初のフレーム アドレスがそれになります。詳細は、表 5-31 を参照してください。

表 5-31： フレーム アドレス レジスタ (メジャー)

	ブロック	行	メジャー
ビット	[15:12]	[11:8]	[7:0]
	0xxx	xxxx	xxxxxxxx

## FAR\_MIN レジスタ

表 5-32： フレーム アドレス レジスタ (マイナー)

	ブロック RAM	(予約)	マイナー
ビット	[15:14]	[13:10]	[9:0]
	xx	0000	xxxxxxxx

FAR への書き込みには次の 3 つの種類があります。

- FAR\_MAJ へ 1 ワード書き込み：FAR\_MAJ のみをアップデートします。
- FAR\_MIN へ 1 ワード書き込み：FAR\_MIN のみをアップデートします。
- FAR\_MAJ へ 2 ワード書き込み：FAR\_MAJ および FAR\_MIN の両方をアップデートします。FAR\_MAJ のデータが先にきます。

## FDRI レジスタ

WCFG コマンドによるコマンド レジスタの読み込み後にフレーム データ入力レジスタを読み込むことで、コンフィギュレーション データがデバイスに書き込まれます。

## FDRO レジスタ

デバイスからのコンフィギュレーション データまたは取り込まれたデータの読み出しに使用します。RCFG コマンドによるコマンド レジスタの読み込み後に読み出しコマンドで FDRO をアドレス指定することで、リードバックを実行します。

## MASK レジスタ

CTL レジスタへの書き込みを実行します。マスクのビット N の 1 により、CTL レジスタでのビット位置の書き込みが可能となります。マスクのデフォルト値は 0 です。

## EYE\_MASK レジスタ

マルチピン ウェークアップ機能の SCP ピンのマスクを格納します。16 ビットで、下位 8 ビットはマスクを示します。上位 8 ビットは予約ビットです。下位 8 ビットは BitGen の `-g wakeup_mask` オプションで設定します。

## LOUT レジスタ

レガシ出力レジスタ (LOUT) はコンフィギュレーション ビットストリームをほかのザイリンクス デバイスにデジター チェーン接続するために使用します。LOUT に書き込まれるデータはシリアル化され、DOUT ピンに出力されます。

## CBC\_REG レジスタ

AES 復号化の初期ベクター (IV) を格納するレジスタで、ビットストリーム圧縮オプションで使用されます。

## IDCODE レジスタ

FDRI への書き込みを行う前に、必ずこのレジスタへの書き込みが必要です。このレジスタに書き込まれた IDCODE は、デバイスの IDCODE と一致する必要があります。詳細は、[83 ページの「コンフィギュレーション シーケンス」](#)を参照してください。

このレジスタを読み出すと、デバイスの IDCODE が返されます。

## CSBO レジスタ

このレジスタは CSB\_O 信号をアサートし、指定されているワード カウントの入力データを無視するように設計されています。CSB\_O が Low のときにのみ出力し、データは渡されないという点を除いて、LOUT レジスタとほぼ同様に機能します。LOUT レジスタのように、デジター チェーン接続をサポートして異なるデバイスに対して複数のコールをネストできます。

## コマンド レジスタ (CMD)

コマンド レジスタは、コンフィギュレーション制御ロジックに対し、グローバル信号をストローブしてその他のコンフィギュレーション ファンクションを実行するよう命令するために使用されます。CMD レジスタに格納されたコマンドは、FAR に新しい値が読み込まれるたびに実行されます。[表 5-33](#) に、コマンド レジスタのコマンドとそのコードを示します。

表 5-33 : コマンド レジスタ コード

コマンド	コード	説明
NULL	00000	NULL コマンド
WCFG	00001	コンフィギュレーション データを書き込みます。FDRI にコンフィギュレーション データを書き込む前に使用します。
MFW	00010	複数のフレームの書き込みです。1 つのフレーム データを複数のフレーム アドレスに書き込む場合に使用します。
LFPM	00011	最終フレームです。GHIGH_B 信号をディアサートし、すべてのインターコネクトを有効にします。この信号は、AGHIGH コマンドでアサートされます。
RCFG	00100	コンフィギュレーション データを読み出します。FDRO からコンフィギュレーション データを読み出す前に使用します。
START	00101	スタートアップ シーケンスを開始します。CRC チェックでエラーが検出されず、DESYNC コマンドが実行された後、スタートアップ シーケンスを開始します。
RCRC	00111	CRC をリセットします。CRC レジスタをリセットします。

表 5-33： コマンド レジスタ コード (続き)

コマンド	コード	説明
AGHIGH	01000	GHIGH_B 信号をアサートします。すべてのインターコネクトをハイインピーダンス状態にし、新しいコンフィギュレーション データを書き込む際の競合を回避します。シャットダウン リコンフィギュレーションでのみ使用します。インターコネクトは LFRM コマンドで再アクティブになります。
GRESTORE	01010	GRESTORE 信号をパルスします。IOB および CLB フリップフロップをユーザー設定に従ってセットまたはリセットします。
SHUTDOWN	01011	シャットダウンシーケンスを開始します。シャットダウンシーケンスを開始し、完了時にはデバイスが使用不可となります。シャットダウンは、次に CRC チェックに成功したとき、または RCRC 命令によって開始されます (通常は RCRC 命令を使用)。
DESYNC	01101	DALIGN 信号をリセットします。コンフィギュレーションの最後でデバイスの同期を解除するために使用します。同期が解除されると、コンフィギュレーション データ ピンのすべての値は無視されます。
I PROG	01110	汎用レジスタで指定されているアドレスからリコンフィギュレーションするため reboot_rst を生成します。

### 制御レジスタ 0 (CTL)

CTL レジスタは Spartan-6 デバイスのプログラムに使用されます。CTL レジスタへの書き込みは、MASK レジスタの値でマスクされます。表 5-34 に、CTL0 レジスタの各ビット位置名を示します。

表 5-34： 制御レジスタ 0 (CTL0) の説明

名前	ビット インデックス	説明	BitGen の デフォルト値
DEC	6	複合化 0：複合化なし 1：復号化あり (SBITS を Level1 以上および mc_enc=1 に自動設定) いったん 1 に設定すると、DEC はハード リブート (PROGRAM_B または JPROGRAM) しない限り変更できません。	0
SBITS	5:4	セキュリティ レベル： Level0：SBITS=00：R/W OK (デフォルト) Level1：SBITS=01：ICAP リードバックのみを許可 Level2：SBITS=10：すべてのリードバックが無効 (en_vrb_b=1 => Vrd=0) Level3：SBITS=11：リードバックは無効、書き込みは CRC、CMD を除き無効 (mc_vrd=1 => Vrd=0) いったん 1 に設定すると、SBITS はソフト リブート (PROGRAM_B、JPROGRAM、I PROG コマンド、エラー リブート、フォールバック リブート) しない限り変更できません。	00



表 5-34 : 制御レジスタ 0 (CTL0) の説明 (続き)

名前	ビット インデックス	説明	BitGen の デフォルト 値
PERSIST	3	コンフィギュレーション後もコンフィギュレーション インターフェイスを維持します。  0 : 維持しない (デフォルト) 1 : 維持する	0
USE_EFUSE_KEY	2	eFUSE キーを復号化キーに使用します。 0 : バックアップ バッテリーが必要な RAM のキーを使用 (デフォルト) 1 : eFUSE キーを使用	0
CRC_EXTSTAT_DISABLE	1	外部 CRC ステータス ピン (INIT_B) がパーシャル リコンフィギュレーションおよび POST CRC で Low になります。  最初のコンフィギュレーションには、常に INIT_B に CRC インジケータがあります。  0 : CRC インジケータが有効 1 : CRC インジケータが無効	0
RESERVED	0	予約済み	1

注記 : PERSIST および ICAP は同時に設定できません。PERSIST が優先されます。

## ステータス レジスタ (STAT)

ステータス レジスタには、さまざまなグローバル信号の値が格納されます。ステータス レジスタには、さまざまなグローバル信号の値が格納されます。表 5-35 に STAT レジスタの各ビット位置の名称を示し、表 5-35 で各ビットについて説明します。

表 5-35 : ステータス レジスタの詳細

名前	ビット	説明
SWWD_strikeout (SyncWordWatchDog)	15	コンフィギュレーション ウォッチドッグ タイマー (CWDT) のカウント、無効な IDCODE、または CRC エラー内において同期ワードを検出できず、コンフィギュレーション エラーが発生したことを示します。エラーの具体的な原因については BOOTSTS レジスタを参照してください。INIT は Low、SWWD_strikeout は High になります。
IN_PWRDN	14	SUSPEND ステータス
DONE	13	DONE ピンからの DONEIN 入力です。
INIT_B	12	INIT_B の値
MODE	11:9	MODE ピン (0、M1、M0) の値
HSWAPEN	8	HSWAPEN ステータス
PART_SECURED	7	0 : 復号化機能のセキュリティの設定なし 1 : 復号化機能のセキュリティの設定あり

表 5-35：ステータス レジスタの詳細 (続き)

名前	ビット	説明
DEC_ERROR	6	復号化処理の前後いずれかに FDRI への書き込みがあったことを示します。 0 : DEC_ERROR なし 1 : DEC_ERROR あり
GHIGH_B	5	GHIGH の状態を示します。
GWE	4	グローバル ライト イネーブルの状態を示します。
GTS_CFG_B	3	グローバル トライステートの状態を示します。
DCM_LOCK	2	DCM および PLL はロックされています。
ID_ERROR	1	FDRI に書き込み中は、IDCODE は検証されません。
CRC_ERROR	0	CRC エラー

### コンフィギュレーション オプション レジスタ (COR1 および COR2)

コンフィギュレーション オプション レジスタは、デデバイスのコンフィギュレーション オプションの設定に使用されます。表 5-36 に COR1 および COR2 の各ビット位置名を示します。

表 5-36：コンフィギュレーション オプション レジスタ (COR1 および COR2) の説明

レジスタ	フィールド	ビット インデックス	説明	BitGen の デフォルト値
COR1	DRIVE_AWAKE	15	0 : アウェーク ピンを駆動しない (オープン ドレイン) 1 : アウェーク ピンをアクティブに駆動	0
	RESERVED	14:5	予約済み	0110111000
	CRC_BYPASS	4	更新された CRC の値をチェックしません。	0
	DONE_PIPE	3	0 : DONEIN にパイプライン ステージを追加しない 1 : DONEIN にパイプライン ステージを追加する	0
	DRIVE_DONE	2	0 : DONE ピンはオープン ドレイン 1 : DONE ピンをアクティブに High に駆動	0
	SSCLKSRC	1:0	スタートアップ シーケンス クロック 00 : CCLK 01 : UserClk 1x : TCK	00

表 5-36 : コンフィギュレーション オプション レジスタ (COR1 および COR2) の説明 (続き)

レジスタ	フィールド	ビット インデックス	説明	BitGen の デフォルト値
COR2	RESET_ON_ERROR	15	crc_error 発生時のフォールバック オプション 0 : Reset On Error (エラー時リセット) が無効 1 : Reset On Error (エラー時リセット) が有効	0
	RESERVED	14:12	予約済み	000
	DONE_CYCLE	11:9	DONE ピンがリリースされるスタートアップ フェーズ です。 (001、010、011、100、101、110)	100
	LCK_CYCLE	8:6	DLL または PLL ロックがアサートするまで現在のス タートアップ フェーズで待機します。 (001、010、011、100、101、110、111<No wait>)	111 (ウェイト なし)
	GTS_CYCLE	5:3	I/O がトライステートからユーザー デザインへ切り替 わるスタートアップ フェーズです。 (000<Keep>、001、010、011、100、101、110、 111<Done>)	101
	GWE_CYCLE	2:0	グローバル ライト イネーブルがアサートするスタート アップ フェーズです。 (000<Keep>、001、010、011、100、101、110、 111<Done>)	110

## サスペンド レジスタ (PWRDN\_REG)

表 5-37 : パワーダウン レジスタの説明

フィールド	ビット インデックス	説明	BitGen の デフォルト値
RESERVED	15	予約済み	
EN_EYES	14	マルチピン ウェークアップの無効/有効を指定 します。 0 : マルチピン ウェークアップを無効 1 : マルチピン ウェークアップを有効	0
RESERVED	13:6	予約済み	0010_0010
FILTER_B	5	0 : Suspend フィルターはオン (300ns) 1 : Suspend フィルターはオフ	0
EN_PGSR	4	0 : Suspend から戻る間 GSR パルスなし 1 : Suspend から戻る間 GSR パルスを生成	0
RESERVED	3	予約済み	

表 5-37： パワーダウン レジスタの説明 (続き)

フィールド	ビット インデックス	説明	BitGen の デフォルト値
EN_PWRDN	2	0 : Suspend は無効 1 : Suspend は有効	0
KEEP_SCLK	0	0 : パワーアップで開始されるスタートアップ シーケンスに MCCLK を使用 1 : パワーアップで開始されるスタートアップ シーケンスに SSCLKSRC を使用	1

## フレーム長レジスタ

フレーム長レジスタ (FLR) には、16 ビット ワードを単位とするフレームが、コンフィギュレーション ビットストリームの開始部分に書き込まれます。FDR 動作の前に FLR の書き込みを行う必要があります。FLR を 複数回設定する必要はありません。

FLR に書き込まれる実値 = 実際のフレームの長さ

Spartan-6 デバイスのセグメンテーションに基づき、type0 (CLB、IOI、および特別ブロック)、type1 (ブロック RAM)、および type3 (PCFG) のフレーム長が固定されます。指定されたフレーム長が必要なブロックは IOB のみです。

表 5-38： フレーム長レジスタ

ビット	FLR
[15:0]	XXXXXXXXXXXXXXXX

## マルチフレーム書き込みレジスタ

Spartan-6 FPGA では、最初のコンフィギュレーションに対してマルチフレーム書き込み (MFWR) がサポートされていますが、コンフィギュレーション中はサポートされません。MFWR を使用する前に、FPGA は 1 パワー サイクル間待機させておくか、PROGRAM\_B を使用してリセットする必要があります。

## コンフィギュレーション ウォッチドッグ タイマー レジスタ

コンフィギュレーション ウォッチドッグ タイマー (CWDT) レジスタには、ウォッチドッグのタイムアウト前に (SYNCWORD が受信されない状態) FPGA が待機するクロック サイクル数の値が格納されます。デフォルトは 64k クロック サイクルです。最小値は 16h'0201 です。

表 5-39： CWDT レジスタ

ビット	値
[15:0]	16h'ffff

## HC\_OPT\_REG レジスタ

HC\_OPT\_REG をデフォルト値にリセットできるのは、por\_b のみです。

表 5-40 : HC\_OPT\_REG の説明

名前	ビット	説明	デフォルト
INIT_SKIP	6	0 : 初期化をスキップしない 1 : 初期化をスキップする	0
RESERVED	5:0	予約済み	011111

## GENERAL レジスタ 1、2、3、4、および 5

GENERAL1 および GENERAL2 レジスタは、SPI および BPI の読み込み可能な複数のコンフィギュレーション アドレスの格納に使用します。

GENERAL3 および GENERAL4 レジスタには GENERAL1 および GENERAL2 と同様の機能がありますが、マルチブート アドレスではなくゴールデン (エラーのない) ビットストリーム アドレスを格納する点異なります。

GENERAL5 は、フェールセーフに必要な追加情報を格納する 16 ビット レジスタです。これらのレジスタの内容は、ソフト リブート中は変更されません。

表 5-41 : 汎用レジスタ

名前	ビット	説明
GENERAL1	[15:0]	マルチブート アドレスの下位半分
GENERAL2	[15:0]	15:8 – SPI オペコード 7:0 – ブート アドレスの上位半分
GENERAL3	[15:0]	ゴールデン ビットストリームの下位半分
GENERAL4	[15:0]	15:8 – SPI OPCODE 7:0 – ゴールデン ブート アドレスの上位半分
GENERAL5	[15:0]	ユーザー定義のスクラッチパッド レジスタ

2 番目のコンフィギュレーションに、以前使用されなかった SPI ベンダー コマンドが必要な場合、新しいベンダー コマンドがこの時点までにビットストリームから GENERAL2 に既に読み込まれています。既知のベンダー コマンドの場合、SPI の読み出しコマンドを GENERAL2 に読み込む必要があります。

SPI の場合、汎用レジスタには 8 ビットのコマンドと 24 ビットのアドレスが含まれます。詳細は、[表 5-42](#) を参照してください。

表 5-42 : SPI 汎用レジスタの例

gen2[15:0]	gen1[15:0]
rd_cmd[7:0]、addr[23:16]	addr[15:0]

BPI には 6 ビットの Don't Care と 26 ビットのアドレスがあります。詳細は、[表 5-43](#) を参照してください。

表 5-43 : BPI 汎用レジスタの例

gen2[15:0]	gen1[15:0]
xxxxxx、address[25:16]	addr[15:0]

## MODE レジスタ

MODE レジスタには、リブートに使用されるモード設定が含まれます (バス幅に 2 ビット、モードに 3 ビット、vsel に 8 ビット)。デフォルトは元のピン設定です。

このレジスタは汎用レジスタと同様にクリアされます。つまり、reboot\_rst ではなく、bus\_reset0 でのみクリアされます (bus\_reset = bus\_reset || reboot\_rst)。詳細は、表 5-44 を参照してください。

表 5-44 : MODE レジスタの説明

名前	ビット	説明	デフォルト
RESERVED	15	予約済み	0
RESERVED	14	予約済み	0
NEW_MODE	13	0 : 物理的モード。bit[10:0] を無視 (デフォルト) 1 : ビットストリームモード。bit[10:0] を使用。マルチブートおよびフォールバックに必須	0
BUSWIDTH	12:11	リブート用のバス幅設定 SPI: 00 : 1 01 : 2 10 : 4	00 (SPI by1)
BOOTMODE	10:8	マルチブートおよびフォールバックに必要なモード設定です。NEW_MODE で有効になります。  bit [10] : 予約 bit [9] : BOOTMODE <1> bit [8] : BOOTMODE <0>	001
BOOTVSEL	7:0	リブート用の vsel 設定	読み出し専用

## CCLK\_FREQ レジスタ

表 5-45 : マスター モード CCLK 周波数セレクトの説明

名前	ビット	説明	デフォルト
EXT_MCLK	14	内部/外部マスター クロックを選択します。 0 : 内部マスター クロックを選択 1 : 外部マスター クロックを選択	0
MCLK_FREQ	9:0	CCLK 周波数を選択します。このレジスタは外部クロックを駆動する ExtMCCLK_Divide 信号を使用した共有レジスタです。	10x1BE

## PU\_GWE レジスタ

10 ビット レジスタで、Suspend (一時停止) からのウェークアップ GWE シーケンスを格納します。詳細は、表 5-46 を参照してください。

表 5-46 : 10 ビット ウェークアップ レジスタのデフォルト

ビット	[9:0]
デフォルト値	10h'006

## PU\_GTS レジスタ

10 ビット レジスタで、Suspend (一時停止) からのウェークアップ GTS シーケンスを格納します。詳細は、表 5-47 を参照してください。

表 5-47 : 10 ビット ウェークアップ レジスタのデフォルト

ビット	[9:0]
デフォルト値	10h'005

## ブート履歴のステータス レジスタ (BOOTSTS)

このレジスタは、POR または ROGRAM\_B をアサートすることでリセット可能です。このレジスタは、MultiBootIt 動作結果を格納することが目的で使用されるため、IPROG コマンドではリセットされません。EOS またはエラーが発生すると、ステータス (0) がステータス (1) にシフトし、ステータス (0) は現在のステータスで更新されます。表 5-48 に、BOOTSTS レジスタの各ビット位置名を示します。

表 5-48 : BOOTSTS レジスタの詳細

名前	ビット	説明
STRIKE_CNT	15:12	ストライク カウント
CRC_ERROR_1	11	CRC エラー
ID_ERROR_1	10	FDRI に書き込み中は、IDCODE は検証されません。
WTO_ERROR_1	9	ウォッチドッグのタイムアウト エラー

表 5-48 : BOOTSTS レジスタの詳細 (続き)

名前	ビット	説明
RESERVED	8	予約
FALLBACK_1	7	1: 00 アドレスにフォールバック 0: 通常のコンフィギュレーション
VALID_1	6	ステータスは有効
CRC_ERROR_0	5	CRC エラー
ID_ERROR_0	4	FDRI に書き込み中は IDCODE は検証されません。
WTO_ERROR_0	3	ウォッチドッグのタイムアウト エラー
RESERVED	2	予約
FALLBACK_0	1	1: ゴールデン (エラーのない) ビットストリーム アドレスにフォールバック 0: 通常のコンフィギュレーション
VALID_0	0	ステータスは有効

## SEU\_OPT レジスタ

このレジスタは SEU 検出を有効にし、SEU の検出において FPGA に必要なステータスおよび周波数を含みます。表 5-49 に、SEU\_OPT レジスタの各ビット位置を示します。

表 5-49 : SEU\_OPT (Soft Error Upset Option) レジスタ

名前	ビット	説明	デフォルト
RESERVED	15	予約	1
RESERVED	14	予約	0
SEU_FREQ	13:4	SEU 検出中の Bus_clk 周波数	10x1be
SEU_RUN_ON_ERR	3	SEU_ERR が検出された場合、実行を続行するかどうかを決定します。 0: 停止 1: 実行続行	0
GLUT_MASK	1	LUTRAM/SRL リードバックのマスク 0: マスクしない 1: LUTRAM/SRL をマスク	1
SEU_ENABLE	0	SEU 検出を有効/無効にします。 0: 無効 1: 有効	0

## ビットストリームの構成

デバイスへの電源投入および初期化が完了すると、INIT\_B ピンがリリースされ、コンフィギュレーションが開始可能となります。初期化後、パケット プロセッサは、同期ワードを受信するまでコンフィギュレーション インターフェイスに送信されるすべてのデータを無視します。同期後、パケッ



ト プロセッサが有効なパケット ヘッダーを受信すると、コンフィギュレーション プロセスが開始されます。表 5-50 に、標準コンフィギュレーションのビットストリーム構成を示します。

表 5-50 : Spartan-6 FPGA ビットストリームの構造

セクション	説明	例
DUMMYWORD	BPI アドレスシフト サイクルの 16 個のダミーワード	0xFFFF
SYNC WORD	同期化用の 2 ワード (32 ビット) パターン	0xAA99 0x5566
HEADER	コンフィギュレーション レジスタのセットアップ	
CFG BODY	開始アドレス R/W コマンド FDRI/FDRO コンフィギュレーション メモリの内容 AUTO CRC ワード	
HEADER2	コンフィギュレーション レジスタのセットアップ (デ イジー チェーン用およびコンフィギュレーション後に 使用可能な機能)	CTL
DESYNC WORD	ビットストリームの終わりを示す 1 ワード (16 ビット ) パターン	0x000D

## デフォルトの初期コンフィギュレーション プロセス

デフォルト ビットストリーム (デフォルトの BitGen 設定で生成されたもの) を使用した初期コンフィギュレーションは、SelectMAP およびシリアル コンフィギュレーション モードの場合、PROGRAM\_B ピンをパルスして開始するか、または JTAG コンフィギュレーションの場合は JPROGRAM 命令を出力して開始します。

## Spartan-6 FPGA のデバイス固有 ID (Device DNA)

Spartan-6 FPGA には、固有のデバイス ID (Device DNA) が組み込まれています。この ID は不揮発性で FPGA に恒久的にプログラムされており、変更できないため、改ざんされることはありません。

FPGA アプリケーションは、デザイン プリミティブの DNA\_PORT (Device DNA Access Port) を使用してこの ID にアクセスします (図 5-13 参照)。

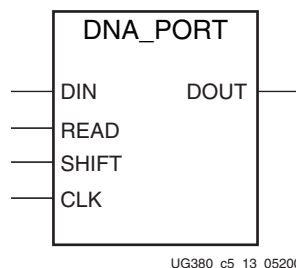


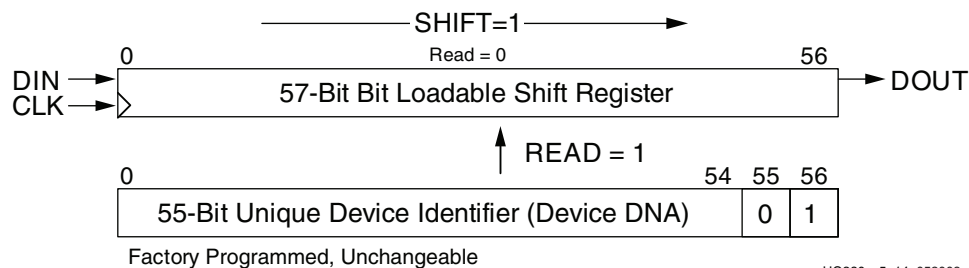
図 5-13 : Spartan-6 FPGA の DSP DNA\_PORT デザイン プリミティブ

## ID 値

図 5-14 に示すように、Device DNA は 57ビットの値です。2 つの最上位ビットは常に 1 および 0 です。残りの 55 ビットは Spartan-6 FPGA 固有のものであります。

## 動作

図 5-14 に、DNA\_PORT デザイン プリミティブの一般的な機能を示します。FPGA アプリケーションは、最初に DNA\_PORT プリミティブをデザイン内でインスタンス化しておく必要があります (図 5-13 参照)。



UG380\_c5\_14\_052009

図 5-14 : DNA\_PORT の動作

Device DNA を読み出すには、まず FPGA アプリケーションが DNA\_PORT 出力シフト レジスタに ID の値を送信します。READ 入力、CLK の立ち上がりエッジでアサートされる必要があります (表 5-51 参照)。これにより、出力シフト レジスタに ID の 57 ビットすべてが並行して読み込まれます。ID のビット 56 は常に 1 であるため、DOUT 出力も 1 です。READ 操作は SHIFT 操作より優先されます。

ID 値の読み出しを継続するには、表 5-51 に示すように、CLK の立ち上がりエッジの後で SHIFT をアサートする必要があります。これで、出力シフト レジスタの値が DOUT 出力にシフトされます。DIN 入力の値は、シフト レジスタにシフトされます。

CLK が High のときに SHIFT が Low から High に遷移すると、初期クロック エッジが誤ったものになるため、これは避けてください。SHIFT は CLK が Low のとき、または CLK の立ち下がりエッジでのみアサートするのが理想的です。

READ と SHIFT の両方が Low の場合、出力シフト レジスタの値が保持され、DOUT の値も変わりません。

表 5-51 : DNA\_PORT の動作

動作	DIN	READ	SHIFT	CLK	シフト レジスタ	DOUT
HOLD	X	0	0	X	前の値を保持	前の値を保持
READ	X	1	X	↑	57 ビットの ID をすべて読み込み	ID のビット 56 (常に 1)
SHIFT	DIN	0	1	↑	DIN をビット 0 にシフトし、 シフト レジスタの値を DOUT にシフト	シフト レジスタの ビット 56

注記：

X = Don't Care

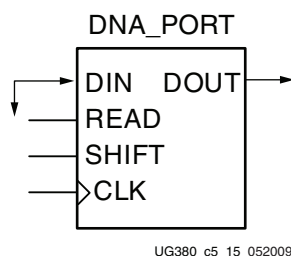
-| = 立ち上がりクロック エッジ

## ID メモリの仕様

図 5-14 に、DNA メモリの一般的な特性を示します。FPGA の ID 値は、ワーストケースの推奨動作状態で継続して使用した場合、最低 10 年間保持されます。ID は表 5-51 に定義されているように READ 動作を使用して最低 3000 万サイクル読み出すことができます。これは、Spartan-6 FPGA の動作寿命で考えると 11 秒ごとに 1 回の読み出しに相当します。

## ID の拡張

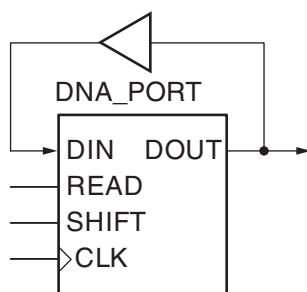
図 5-15 に示すとおり、DNA\_PORT プリミティブを使用するほとんどのアプリケーションで、DIN データ入力スタティックな値に接続されています。



UG380\_c5\_15\_052009

図 5-15 : 定数でシフト

図 5-16 のように、DOUT シリアル出力ポートを DIN シリアル入力ポートに接続すると ID の長さを延長できます。このようにすると、ID を任意の長さに拡張できます。しかし、この場合でも、57 ビットの繰り返しパターンで、各パターンに 55 ビットのみです。図 5-16 にはバッファが 1 つ含まれており、ユーザーによる DNA ロジック拡張に対応するロジック挿入、またはホールド タイム要件を満たすループバック遅延を示しています。



UG380\_c5\_16\_021010

図 5-16 : 巡回シフト

また、FPGA のロジック リソースを使用して ID にビットを追加することも可能です。図 5-17 に示すように、FPGA アプリケーションは DNA\_PORT の DIN シリアル入力を介して追加のビットを挿入できます。ロジック リソースを利用する場合は、固定値のビットを追加することも、Device DNA から計算した変数のビットを追加することもできます。

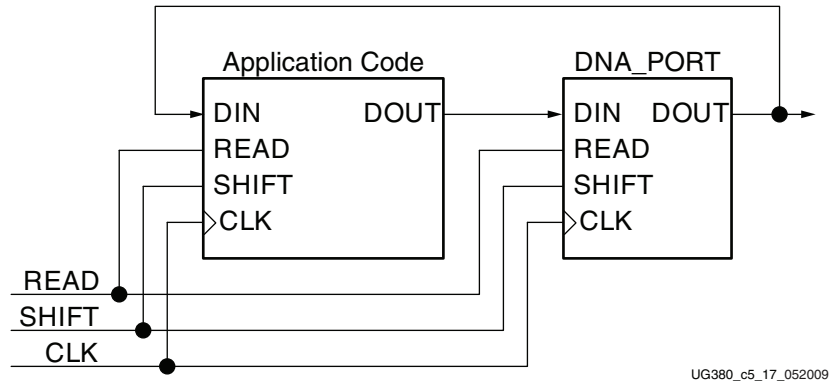


図 5-17： ビットストリーム固有のコード

## JTAG によるデバイス ID へのアクセス

FPGA の内部デバイス ID と、DIN 入力にシフトされる値は、ISC\_DNA コマンドを使用し JTAG ポートを介して読み出すことができます。この場合、ISC\_DNA コマンドを発行する前に ISC\_ENABLE を読み込む必要があります。

図 5-14 に示すように、デバイスがシフト DR ステートに入ると、ISC\_DNA コマンドに続いて ID のビット 56 が TDO JTAG 出力に出力されます。Device DNA の残りのビットおよびレジスタへの入力のデータは、JTAG コントローラーがシフト DR ステートのときに順次シフト出力されます。この動作が完了するとき、ISC\_DISABLE コマンドを発行する必要があります。

## IMPACT によるデバイス ID へのアクセス

ISE 10.1 (およびそれ以降のバージョン) の iMPACT ソフトウェアでも Device DNA 値を読み出すことができます。FPGA から Device DNA を読み出すには、パッチ コマンド `readDna -p <position>` を実行します。

## ビットストリームの圧縮

FPGA ビットストリームはデフォルトで圧縮されていませんが、Spartan-6 FPGA は基本的なビットストリーム圧縮をサポートします。この圧縮はとても単純なものです、アプリケーションによっては効果的です。ISE ビットストリーム ジェネレーターは、FPGA ビットストリームに重複するコンフィギュレーション データ フレームがないかを調べます。これらの重複は次のようなデザインで発生することがほとんどです。

- 未使用のブロック RAM またはハードウェア 乗算器を使用する FPGA デザイン
- ロジック使用率の低い FPGA デザイン (多くの場合、FPGA アレイが空)

重複するデータの確認後、ISE ソフトウェアによる圧縮された FPGA ビットストリームの生成が可能になります。FPGA のコンフィギュレーション時に、内部のコンフィギュレーション コントローラーが重複したデータ フレームを複数の位置にコピーします。

圧縮量は確定的なものではありません。ソース FPGA デザインを変更すると、圧縮されたビットストリームのサイズが増大します。ほぼ空の FPGA デザインには全体的に大きな圧縮要素があります。同様に、ブロック RAM カラムが空の FPGA デザインにも大きな圧縮要素があります。

圧縮されたビットストリームの大きな利点は次のとおりです。

- メモリ フットプリントの縮小
- 不揮発性メモリのプログラム時間の高速化
- コンフィギュレーション時間の高速化

BitGen の **-g compress** オプションを使用すると圧縮が有効になります。

パラレル プラットフォーム フラッシュ PROM には、独自の圧縮メカニズムがあります。詳細は、『Platform Flash PROM ユーザー ガイド』([UG161](#)) の「XCFxxP 圧縮の復元およびクロック オプション」の章を参照してください。



# リードバックおよび コンフィギュレーションの検証

Spartan®-6 デバイスでは、SelectMAP、ICAP、および JTAG インターフェイスを介してコンフィギュレーション メモリを読み出すことができます。リードバック検では、ユーザー メモリ エLEMENT (LUT RAM、SRL16、およびブロック RAM) のその時点での値を含め、すべてのコンフィギュレーション メモリ セルを読み出すことが可能です。

コンフィギュレーション メモリを読み出すには、リードバックを開始するコマンド シーケンスをデバイスに送信する必要があります。リードバックが開始すると、デバイスはコンフィギュレーション メモリの内容を SelectMAP または JTAG インターフェイスに出力します。「[SelectMAP インターフェイスからのコンフィギュレーションレジスタへのアクセス](#)」に、IEEE Std 1149.1 JTAG のコンフィギュレーション メモリの読み出し手順が記載されています。

リードバック コマンド シーケンスは、マイクロプロセッサ、CPLD、または FPGA ベースのシステムから送信できます。また、iMPACT で JTAG ベースのリードバック検証を実行することも可能です。iMPACT はザイリンクスの ISE® で提供されているデバイス プログラミング ソフトウェアで、Spartan-6 デバイスのすべてのリードバックおよび比較機能をサポートしており、コンフィギュレーション エラーの有無をレポートする機能もあります。

デバイスからコンフィギュレーション メモリの内容が読み出されると、次にリードバック ビットストリームとコンフィギュレーション ビットストリームが比較され、エラーの有無が判定されます。この処理の詳細は、「[リードバックデータの検証](#)」で説明します。

## リードバックを実行するためのデザインの準備

JTAG または SelectMAP を使用してリードバックを行う場合、次の 2 点を必ずビットストリームに設定してください。BitGen セキュリティ設定では、リードバックの禁止を解除し (**-g Security :none**)、ビットストリームの暗号化は使用しないでください。また、SelectMAP インターフェイスを利用してリードバックを行う場合は、BitGen の **persist** オプションを **-g Persist:Yes** に設定する必要があります。この設定を行っておかないと SelectMAP データ ピンはコンフィギュレーション後にユーザー I/O となり、その後のコンフィギュレーションが行えなくなります。バウンダリ スキャン ポートを使用してリードバックを行う場合は、最初に挙げたセキュリティおよび暗号化のオプション以外に特別な設定は必要ありません。また、これらの要件は ICAP でリードバックするときに適用する必要はありません。

## リードバック コマンド シーケンス

Spartan-6 FPGA コンフィギュレーション メモリは、JTAG、SelectMAP、ICAP インターフェイスからアクセス可能な FDRO (フレーム データ レジスタ、出力) コンフィギュレーション レジスタから読み出されます。JTAG および SelectMAP インターフェイスでは、コンフィギュレーション ロ

ジックからブロック RAM へのアクセス中、ユーザー デザインからこのブロック RAM はアクセスできませんが、FPGA デザインがアクティブまたはシャットダウン状態でもリードバックは実行可能です。

## SelectMAP インターフェイスからのコンフィギュレーション レジスタへのアクセス

SelectMAP インターフェイスからコンフィギュレーション メモリを読み出すには、インターフェイスを書き込み制御用に設定して FPGA へコマンドを送信し、次に読み出し制御用に設定して、データをデバイスから読み出す必要があります。SelectMAP インターフェイスの書き込みおよび読み出しは、RDWR\_B 入力によって決定します。SelectMAP データ ピンは、インターフェイスが書き込み制御に設定されているとき (RDWR\_B = 0) は入力で、読み出し制御に設定されているとき (RDWR\_B = 1) は出力です。

CSI\_B 信号は、RDWR\_B 信号をトグルする前にディアサートする必要があります (CSI\_B = 1)、そうしなければ ABORT が発生します (155 ページの「SelectMAP の ABORT」参照)。

SelectMAP インターフェイスにおける書き込み制御と読み出し制御間の変更手順は、次のとおりです。

1. CSI\_B をディアサートする。
2. RDWR\_B をトグルする。  
RDWR\_B = 0 : 書き込み制御  
RDWR\_B = 1 : 読み出し制御
3. CSI\_B をアサートする。
4. CSI\_B および RDWR\_B は CCLK に同期する。
5. 図 6-1 に、この手順を図示します。

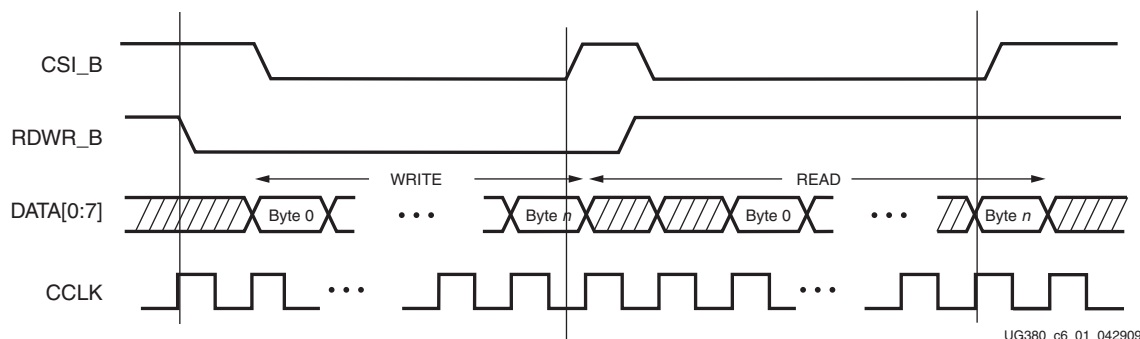


図 6-1 : SelectMAP ポートの書き込みおよび読み出し制御間の変更

## コンフィギュレーション レジスタの読み出し手順 (SelectMAP)

最も簡潔な読み出し処理は、COR0 または STAT レジスタなどのコンフィギュレーション レジスタに対して実行されます。レジスタによっては読み出しアクセスがないものがありますが、アクセスのあるすべてのレジスタは、SelectMAP インターフェイスを介した読み出しが可能です。SelectMAP インターフェイスから STAT レジスタを読み出す手順は、次のとおりです。

1. デバイスにバス幅検出シーケンスと同期ワードを書き込む。
2. デバイスに「STAT レジスタ読み出し」パケット ヘッダーを書き込む。



3. デバイスにダミー ワードを 2 つ書き込み、パケット バッファをフラッシュする。
4. SelectMAP インターフェイスから 1 ワードを読み出す (ステータス レジスタの値)。
5. デバイスに DESYNC コマンドを書き込む。
6. デバイスにダミー ワードを 2 つ書き込み、パケット バッファをフラッシュする。

表 6-1 : ステータス レジスタのリードバック コマンド シーケンス (16 ビットの SelectMAP)

手順	SelectMAP ポートの方向	コンフィギュレーション データ [15:0]	説明
1	書き込み	FFFF	ダミー ワード
2	書き込み	FFFF	ダミー ワード
3	書き込み	AA99	同期ワード
4	書き込み	5566	同期ワード
5	書き込み	2000	NOOP
6	書き込み	2901	STAT レジスタを読み出すタイプ 1 パケット ヘッダーを書き込む
7	書き込み	2000	NOOP
8	書き込み	2000	NOOP
9	書き込み	2000	NOOP
10	書き込み	2000	NOOP
11	読み出し	SSSS	デバイスが STAT レジスタから 1 ワードをコンフィギュレーション インターフェイスに書き込む
12	書き込み	30A1	Cタイプ 1、CMD に 1 ワード書き込み
13	書き込み	000D	DESYNC コマンド
14	書き込み	2000	NOOP
15	書き込み	2000	NOOP

図 6-2 に示すように、SelectMAP インターフェイスは、手順 10 と手順 11 の間で書き込みから読み出し制御用に変更し、手順 11 の後に再び書き込み制御用に戻す必要があります。40 ページの表 2-4 および 41 ページの表 2-5 に示すように、SelectMAP 16 ビット データ順は、ICAP インターフェイスに適用されます。

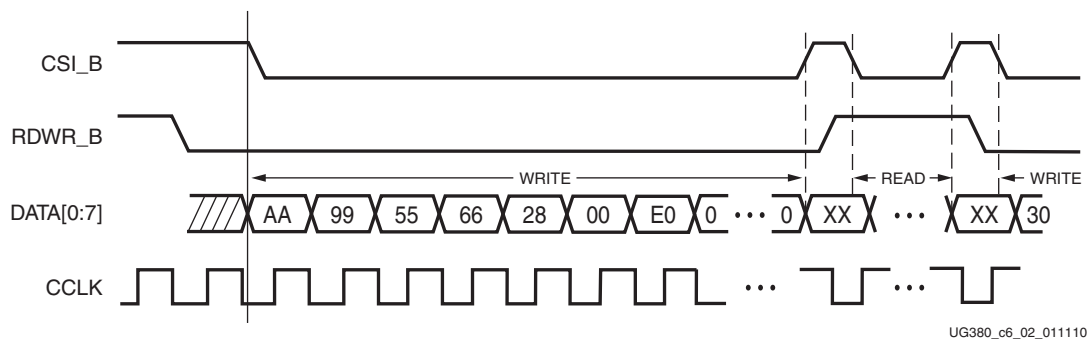


図 6-2： 8 ビットの SelectMAP ステータス レジスタ読み出し

STAT 以外のレジスタを読み出す場合は、タイプ 1 のパケット ヘッダーで指定するアドレス (表 6-1 の手順 2) を変更し、必要に応じてワード数を修正してください。FDRO レジスタからの読み出しは、「[コンフィギュレーション メモリの読み出し手順 \(SelectMAP\)](#)」で説明するように特別な手順が必要となります。

## コンフィギュレーション メモリの読み出し手順 (SelectMAP)

FDRO レジスタからコンフィギュレーション メモリを読み出す手順は、その他のレジスタから読み出す場合と同様です。ただし、コンフィギュレーション ロジックに対応するための追加手順が必要になります。FDRO レジスタからのコンフィギュレーション データは、フレーム バッファを通するため、リードバック データの最初のフレームは無視する必要があります。

1. デバイスにダミー ワードおよび同期ワードを書き込む。
2. NOOP コマンドを 1 つ書き込む。
3. SHUTDOWN コマンドを書き込み、NOOP コマンドを 1 つ書き込む。
4. インターコネクトを無効にするコマンドを書き込み、NOOP コマンドを 1 度書き込む。
5. フレーム長レジスタを設定する。
6. 開始フレーム アドレスを FAR に書き込み (通常は 0x00000000)。
7. RCFG コマンドを CMD レジスタに書き込む。
8. デバイスに「FDRO レジスタ読み出し」パケット ヘッダーを書き込む。FDRO 読み出し長は、次の式で求められます。

$$\text{FDRO 読み出し長} = (\text{フレームごとのワード数}) \times (\text{読み出すフレーム数} + 1) + 1$$

フレーム バッファ分の追加フレームが読み出されます。DOUT\_BUSY が Low の間にリードバックをストローブする必要があり、フレーム バッファでは、読み出しの始めのダミー データとなる 1 フレームが必要です。また、SelectMap8 モードでは 1 ワードが余分に読み出されます。

9. デバイスにダミー ワードを 2 つ書き込み、パケット バッファをフラッシュする。
10. SelectMAP インターフェイスから FDRO レジスタを読み出す。FDRO 読み出し長は上述の手順 9 と同じ。
11. NOOP コマンドを 1 つ書き込む。
12. START コマンドを書き込み、NOOP コマンドを 1 つ書き込む。
13. DESYNC コマンドを書き込む。

14. パケット バッファをフラッシュするため、最低 64 ビットの NOOP コマンドを書き込み、DONE が High になるまで CCLK のパルスを送信する。

表 6-2 に、リードバック コマンド シーケンスを示します。

表 6-2 : リードバック コマンドのシャットダウン シーケンス (SelectMAP)

手順	SelectMAP ポートの方向	コンフィギュレーション データ	説明
1	書き込み	FFFF	ダミー ワード
		FFFF	ダミー ワード
		FFFF	ダミー ワード
		FFFF	ダミー ワード
		AA99	同期ワード
		5566	同期ワード
2	書き込み	2000	タイプ 1、NOOP ワード 0
3	書き込み	30A1	タイプ 1、CMD に 1 ワード書き込み
		000B	SHUTDOWN コマンド
		2000	タイプ 1、NOOP ワード 0 16 サイクル間 REPEAT
4	書き込み	30A1	タイプ 1、CMD に 1 ワード書き込み
		0008	AGHIGH コマンド
		2000	タイプ 1、NOOP ワード 0
5	書き込み	31a1	タイプ 1、FLR に 1 ワード書き込み
		XXXX	フレーム長
6	書き込み	3022	Fタイプ 1、FAR に 2 ワード書き込み
		0000	FAR_MAJ = 0000
		0000	FAR_MIN = 0000
7	書き込み	30A1	タイプ 1、CMD に 2 ワード書き込み
		0004	RCFG コマンド
8	書き込み	4840	タイプ 2、FDRO から 0 ワード読み出し
		XXXX	タイプ 2、FDRO から XXXX 読み出し
9	書き込み	2000	タイプ 1、NOOP ワード 0
		...	タイプ 1、65 以上の NOOP ワード 0
10	読み出し	0000	パケット データ、FDRO から 0 ワード読み出し
		...	
		0000	パケット データ、FDRO から最終ワード読み出し
11	書き込み	2000	タイプ 1、NOOP ワード 0

表 6-2： リードバック コマンドのシャットダウン シーケンス (SelectMAP) (続き)

手順	SelectMAP ポートの方向	コンフィギュレーション データ	説明
12	書き込み	30A1	タイプ 1、CMD に 1 ワード書き込み
		0005	START コマンド
		2000	タイプ 1、NOOP ワード 0
		2000	タイプ 1、NOOP ワード 0
		2000	タイプ 1、NOOP ワード 0
		2000	タイプ 1、NOOP ワード 0
13	書き込み	30A1	タイプ 1、CMD に 1 ワード書き込み
		0007	RCRC コマンド
		2000	タイプ 1、NOOP ワード 0
14	書き込み	30A1	タイプ 1、CMD に 1 ワード書き込み
		000D	DESYNC コマンド
15	書き込み	2000	タイプ 1、NOOP ワード 0 16 サイクル間 REPEAT

書き込みから読み出しへの切り替え後 (CSI\_B および RDWR\_B の両方が Low)、DOUT\_BUSY が Low の間に、ユーザー ロジックではリードバック データをストローブする必要があります。リードバック データがいつ有効になるかを見極めるため DOUT\_BUSY をモニターする必要があります。

リードバックが開始し、BUSY がディאサートされた後、SelectMAP バス幅によっては多数のダミー ワードが有効データの前に読み出されます。表 6-3 に、2 つの SelectMAP 幅のダミーのリードバック サイクルを示します。

表 6-3： リードバック レイテンシ (SelectMAP)

	x8	x16
CSI_B からリードバックまでのレイテンシ	3 クロック	2 クロック

注記：

- これらのレイテンシは、CSI\_B および RDWR\_B が書き込みと読み出し間で 1 サイクル間ディアサートされることを想定しています。このディアサートが 2 サイクル以上続く場合、レイテンシは少なくなります。有効なリードバック データを得るには BUSY 信号を監視するのが最善です。

## JTAG インターフェイスからのコンフィギュレーション レジスタへのアクセス

JTAG から Spartan-6 FPGA コンフィギュレーション ロジックへのアクセスには、JTAG\_CFG\_IN および CFG\_OUT レジスタを使用します。CFG\_IN および CFG\_OUT レジスタはコンフィギュレーション レジスタではなく、BYPASS や BOUNDARY\_SCAN と同様の JTAG レジスタです。CFG\_IN にシフトしたデータは、コンフィギュレーション パケット プロセッサに入り、SelectMAP インターフェイスからのコマンドと同様に処理されます。

リードバック コマンドは、CFG\_IN レジスタを介してコンフィギュレーション ロジックに書き込まれ、コンフィギュレーション メモリは CFG\_OUT レジスタから読み出されます。表 6-4 に、CFG\_IN および CFG\_OUT レジスタにアクセスする際の JTAG ステートの遷移を示します。

表 6-4 : JTAG CFG\_IN および CFG\_OUT 命令のシフト

手順	説明	セットアップおよびホールド		クロック数 (TCK)
		TDI	TMS	
1	5 クロックの間、TMS に 1 をを入力してデバイスを TLR ステートにする	X	1	5
2	RTI ステートに移行する	X	0	1
3	SELECT-IR ステートに移行する	X	1	2
4	SHIFT-IR ステートに移行する	X	0	2
5	CFG_IN または CFG_OUT 命令の最初の 5 ビットを LSB から順にシフトする	000101 (CFG_IN)	0	5
		000100 (CFG_OUT)		
6	SHIFT-IR ステートから出る間に、CFG_IN または CFG_OUT 命令の MSB をシフトする	0	1	1
7	SELECT-DR ステートに移行する	X	1	2
8	SHIFT-DR ステートに移行する	X	0	2
9	SHIFT_DR ステートの間に、データを MSB から CFG_IN レジスタにシフトする、または CFG_OUT からシフトする	X	0	X
10	SHIFT-DR ステートから出る間に LSB をシフトする	X	1	1
11	TMS が「1」の状態でクロックを 5 回入力し、TAP をリセットする	X	1	5

## コンフィギュレーション レジスタの読み出し手順 (JTAG)

最も簡潔な読み出し処理は、COR0 または STAT レジスタなどのコンフィギュレーション レジスタに対して実行されます。レジスタによっては、読み出しアクセスがないものがありますが、アクセスがあるすべてのレジスタは JTAG インターフェイスから読み出すことができます。JTAG インターフェイスから STAT レジスタを読み出す手順は、次のとおりです。

1. TAP コントローラーをリセットする。
2. SHIFT-IR ステート中に、CFG\_IN 命令を JTAG 命令レジスタにシフトする。CFG\_IN 命令の LSB を最初にシフトし、MSB は TAP コントローラーが SHIFT-IR ステートから出る間にシフトします。
3. SHIFT-DR ステート中に、パケット書き込みコマンドを CFG\_IN レジスタにシフトする。
  - a. デバイスに同期ワードを書き込む。
  - b. デバイスに「STAT レジスタ読み出し」パケット ヘッダーを書き込む。
  - c. デバイスにダミー ワードを 2 つ書き込み、パケット バッファをフラッシュする。

CFG\_IN レジスタを通して送信されたすべてのコンフィギュレーション パケットの MSB を最初に送信する必要があります。LSB は、TAP コントローラーが SHIFT-DR ステートから出る間にシフトします。
4. SHIFT-IR ステート中に、CFG\_OUT 命令を JTAG 命令レジスタにシフトする。CFG\_OUT 命令の LSB から順にシフトし、TAP コントローラーが SHIFT-IR ステートから出る間に MSB をシフトします。
5. SHIFT-DR ステート中に、ステータス レジスタから 32 ビット シフトする。
6. TAP コントローラーをリセットする。

表 6-5：ステータス レジスタ リードバック コマンド シーケンス (JTAG)

手順	説明	セットアップおよび ホールド		クロック数 (TCK)
		TDI	TMS	
1	TMS が「1」の状態ではクロックを 5 回入力し、デバイスを TLR ステートにする	X	1	5
	RTI ステートに移行する	X	0	1
	SELECT-IR ステートに移行する	X	1	2
	SHIFT-IR ステートに移行する	X	0	2
2	CFG_IN 命令の最初の 5 ビットを LSB からシフトする	000101 (CFG_IN)	0	5
	SHIFT-IR ステートから出る間に、CFG_IN 命令の MSB をシフトする	0	1	1
	SELECT-DR ステートに移行する	X	1	2
	SHIFT-DR ステートに移行する	X	0	2

表 6-5 : ステータス レジスタ リードバック コマンド シーケンス (JTAG) (続き)

手順	説明	セットアップおよび ホールド		クロック数 (TCK)
		TDI	TMS	
3	コンフィギュレーション パケットを MSB から CFG_IN データ レジスタにシフトする	a: 0xAA99 a: 0x5566 c: 0x2901 d: 0x2000 d: 0x2000 e: 0x2000 e: 0x2000	0	159
	SHIFT-DR ステートから出る間に、最後のコンフィギュレーション パケットの LSB をシフトする	0	1	1
	SELECT-IR ステートに移行する	X	1	3
	SHIFT-IR ステートに移行する	X	0	2
4	CFG_OUT 命令の最初の 5 ビットを LSB からシフトする	000100 (CFG_OUT)	0	5
	SHIFT-IR ステートから出る間に、CFG_OUT 命令の MSB をシフトする	0	1	1
	SELECT-DR ステートに移行する	X	1	2
	SHIFT-DR ステートに移行する	X	0	2
5	STAT レジスタの内容を CFG_OUT データ レジスタからシフトする	0xSSSS	0	15
	SHIFT-DR ステートから出る間に、STAT レジスタの最後のビットを CFG_OUT データ レジスタからシフトする	S	1	1
	SELECT-IR ステートに移行する	X	1	3
	SHIFT-IR ステートに移行する	X	0	2
6	TAP コントローラーをリセットする	X	1	5

JTAG CFG\_IN レジスタにシフトしたパケットは、SelectMAP から STAT レジスタを読み出す場合に SelectMAP インターフェイスを介してシフトしたパケットと同一です。

### コンフィギュレーション メモリの読み出し手順 (IEEE Std 1149.1 JTAG)

JTAG インターフェイスを利用して FDRO レジスタからコンフィギュレーション メモリを読み出す手順は、その他のレジスタから読み出す場合と基本的には同じです。ただし、フレーム ロジックに対応するための追加手順が必要になります。FDRO レジスタからのコンフィギュレーション データはフレーム バッファーを通るため、リードバック データの最初のフレームは、「ダミー データ」であり、無視する必要があります (FDRI および FDRO レジスタの説明を参照)。通常は、IEEE Std 1149.1 JTAG リードバック フローを推奨します。

1. TAP コントローラーをリセットする。
2. CFG\_IN 命令を JTAG 命令レジスタにシフトする。CFG\_IN 命令の LSB を最初にシフトし、MSB は TAP コントローラーが SHIFT-IR ステートから出る間にシフトします。

3. SHIFT-DR ステート中に、パケット書き込みコマンドを CFG\_IN レジスタにシフトする。
  - a. デバイスにダミー ワードを書き込む。
  - b. デバイスに同期ワードを書き込む。
  - c. CMD レジスタ ヘッダーに 1 ワードを書き込む。
  - d. リードバックするデータ フレームの長さを指定する。
  - e. 開始フレーム アドレスを FAR レジスタに書き込む。
4. JSHUTDOWN 命令を JTAG 命令レジスタにシフトする。
5. RTI ステートに移行し、シャットダウン シーケンスを完了するまで TCK の 24 サイクル間、この状態を維持する。シャットダウン シーケンス中、DONE ピンは Low になります。
6. CFG\_IN 命令を JTAG 命令レジスタにシフトする。
7. SHIFT-DR ステートに移行し、パケット書き込みコマンドを CFG\_IN レジスタにシフトする。
  - a. デバイスにダミー ワードを書き込む。
  - b. デバイスに同期ワードを書き込む。
  - c. CMD レジスタ ヘッダーに 1 ワードを書き込む。
  - d. リードバックするデータ フレームの長さを指定する。
  - e. 開始フレーム アドレスを FAR レジスタに書き込む。
  - f. デバイスに RCFG コマンドを書き込む。
  - g. デバイスにタイプ 1 の「FDRO レジスタの読み出し」パケット ヘッダーを書き込む。
  - h. デバイスにダミー ワードを 2 つ書き込み、パケット バッファをフラッシュする。

CFG\_IN レジスタを通して送信されたすべてのコンフィギュレーション パケットの MSB を最初に送信する必要があります。LSB は、TAP コントローラーが SHIFT-DR ステートから出る間にシフトします。
8. SHIFT-DR ステート中に、CFG\_OUT 命令を JTAG 命令レジスタにシフトする。  
CFG\_OUT 命令の LSB が最初にシフトし、MSB は TAP コントローラーが SHIFT-IR ステートから出る間にシフトします。
9. SHIFT-DR ステート中に、FDRO レジスタからのフレーム データをシフトする。
10. TAP コントローラーをリセットする。

表 6-6： リードバック コマンドのシャットダウン シーケンス (JTAG)

手順	説明	セットアップおよび ホールド		クロック数 (TCK)
		TDI	TMS	
1	TMS が「1」の状態でもクロックを 5 回入力し、デバイスを TLR ステートにする	X	1	5
	RTI ステートに移行する	X	0	1
	SELECT-IR ステートに移行する	X	1	2
	SHIFT-IR ステートに移行する	X	0	2



表 6-6 : リードバック コマンドのシャットダウン シーケンス (JTAG) (続き)

手順	説明	セットアップおよび ホールド		クロック数 (TCK)
		TDI	TMS	
2	CFG_IN 命令の最初の 5 ビットを LSB からシフトする	000101	0	5
	SHIFT-IR ステートから出る間に、CFG_IN 命令の MSB をシフトする	0	1	1
	SELECT-DR ステートに移行する	X	1	2
	SHIFT-DR ステートに移行する	X	0	2
3	コンフィギュレーション パケットを MSB から CFG_IN データ レジスタにシフトする	a: 0xFFFF b: 0xAA99 b: 0x5566 c: 0x30A1 d: 0x0007 e: 0x2000 f: 0x2000	0	111
	SHIFT-DR ステートから出る間に、最後のコンフィ ギュレーション パケットの LSB をシフトする	0	1	1
	SELECT-IR ステートに移行する	X	1	3
	SHIFT-IR ステートに移行する	X	0	2
4	JSHUTDOWN 命令の最初の 5 ビットを LSB から シフトする	001101	0	5
	SHIFT-IR ステートから出る間に、JSHUTDOWN 命令の MSB をシフトする	0	1	1
5	RTI ステートに移行し、TCK の 24 サイクル間、そ の状態を維持する	X	0	24
	SELECT-IR ステートに移行する	X	1	2
	SHIFT-IR ステートに移行する	X	0	2
6	CFG_IN 命令の最初の 5 ビットを LSB からシフト する	000101	0	5
	SHIFT-IR ステートから出る間に、CFG_IN 命令の MSB をシフトする	0	1	1
	SELECT-DR ステートに移行する	X	1	2
	SHIFT-DR ステートに移行する	X	0	2

表 6-6： リードバック コマンドのシャットダウン シーケンス (JTAG) (続き)

手順	説明	セットアップおよび ホールド		クロック数 (TCK)
		TDI	TMS	
7	コンフィギュレーション パケットを MSB から CFG_IN データ レジスタにシフトする	a: 0xFFFF b: 0xAA99 b: 0x5566 c: 0x30A1 c: 0x0008 d: 0x31A1 d: 0XXXXX e: 0x3022 e: 0x0000 e: 0x0000 f: 0x30A1 f: 0x0004 g: 0x4880 g: 0x0000 g: 0x0000 h: 0x2000 h: 0x2000	0	191
	SHIFT-DR ステートから出る間に、最後のコンフィ ギュレーション パケットの LSB をシフトする	0	1	1
	SELECT-IR ステートに移行する	X	1	3
	SHIFT-IR ステートに移行する	X	0	2
8	CFG_OUT 命令の最初の 5 ビットを LSB からシ フトする	000100 (CFG_OUT)	0	5
	SHIFT-IR ステートから出る間に、CFG_OUT 命令 の MSB をシフトする	0	1	1
	SELECT-DR ステートに移行する	X	1	2
	SHIFT-DR ステートに移行する	X	0	2
9	FDRO レジスタの内容を CFG_OUT データ レジ スタからシフトする	...	0	リード バッ クのビット 数: 1
	SHIFT-DR ステートから出る間に、FDRO レジス タの最後のビットを CFG_OUT データ レジスタ からシフトする	X	1	1
	SELECT-IR ステートに移行する	X	1	3
	SHIFT-IR ステートに移行する	X	0	2
10	TAP コントローラーを TLR ステートにして完了 する	X	1	3

表 6-7 に、リードバック ファイルを示します。

表 6-7：フィードバック ファイル

ファイル 拡張子	ファイル の種類	BitGen 設定	説明
RBA	ASCII	<b>-b</b> および <b>-g Readback</b>	コンフィギュレーション コマンドではなくリードバック コマンドを含む ASCII 形式のファイルで、コンフィギュレーション データが通常保存されている場所にあるリードバック データが含まれます。このファイルは、MSK ファイルと併用する必要があります。
RBB	バイナリ	<b>-g Readback</b>	RBA ファイルのバイナリ版です。このファイルは、MSK ファイルと併用する必要があります。
RBD	ASCII	<b>-g Readback</b>	パッド フレームを含む、予測されるリードバック データのみを含む ASCII 形式のファイルです。コマンドは含まれていません。このファイルは、MSD ファイルと併用する必要があります。
MSK	バイナリ	<b>-m</b>	BIT ファイルと同じコンフィギュレーション コマンドを含むバイナリ ファイルですが、FDRI 書き込みパケットの内容は、BIT ファイルの対応ビットを比較すべきかどうかを示すマスク データに置き換えられます。マスク ビットが 0 の場合、リードバック データストリームの対応ビットを比較します。マスク ビットが 1 の場合、リードバック データストリームの対応ビットは無視されます。
MSD	ASCII	<b>-g Readback</b>	マスク ビットのみを含む ASCII ファイルです。MSD ファイルの最初のビットは、RBD ファイルの最初のビットと対応しています。実際のリードバック ストリームのパッド データは、MSD および RBD ファイルにあります。マスク ビットが 0 の場合はビットストリームのデータと比較し、マスク ビットが 1 の場合はビットは検証されません。
LL	ASCII	<b>-1</b>	リードバック用にキャプチャされるデザイン内の各ノードの情報を含む ASCII 形式のファイルです。このファイルには、リードバック ストリームの絶対ビット位置、フレーム アドレス、フレーム オフセット、使用されるロジック リソース、およびデザイン内のコンポーネント名が含まれます。

design.rba および design.rbb ファイルには、リードバック コマンドと予期されるリードバック データがまとめられ、RBD ファイルには予測されるリードバック データのみ含まれます。RBD ファイルをリードバック用に使用するシステムは、別の場所にリードバック コマンドを格納する必要があります。実際のリードバック データは、MSK または MSD マスク ファイルに対してマスクする必要があります。これは、RBA、RBB、および RBD ファイルにある予測されるリードバック ストリームのビットの一部を無視する必要があるためです。

リードバック コマンド セット ファイルは、ユーザーが SelectMAP または JTAG インターフェイスを書き込みから読み出しに変更しなければならないタイミングを示すものではありません。ユーザーは、上述のリードバック コマンド シーケンスに基づいてその変更を行う必要があります。

## リードバック データの検証

「[コンフィギュレーション メモリの読み出し手順 \(SelectMAP\)](#)」で説明したように、リードバック データ ストリームには、コンフィギュレーション フレーム データおよびその前にある 1 フレームのパッド データが含まれています。

リードバック ストリームには、コンフィギュレーション ビットストリーム内のコマンドまたはパケット情報は含まれておらず、リードバック中に CRC の算出は実行されません。図 6-3 に、リードバック データ ストリームを示します。

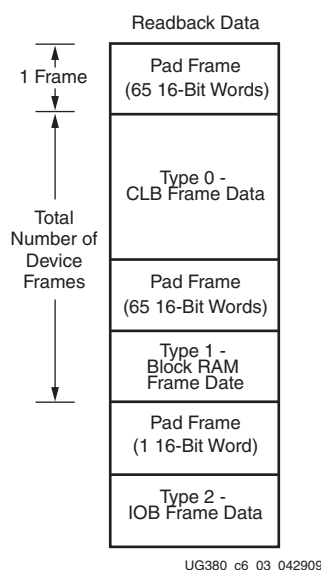


図 6-3： リードバック データ ストリーム

リードバック データ ストリームは、デバイスにプログラムされたオリジナルのコンフィギュレーション フレーム データと比較することで検証されます。ただし、リードバック データ ストリーム内のビットによっては、ユーザー メモリまたは Null メモリ位置に対応しているため、比較を実行してはいけない場合があります。リードバック データ ストリーム内の Don't Care ビットの位置は、MSK および MSD マスク ファイルに示されています。これらのファイルは形式が異なりますが、基本的に同一の情報が含まれます。リードバック データがデバイスから読み込まれると、次のいずれかの方法で比較処理を実行できます。

1. リードバック データを基準となるゴールデン RBD リードバック ファイルと比較し、MSD ファイルを使用してマスクします (図 6-4 参照)。

最も簡単なリードバック データ ストリームの検証方法は、MSD ファイルをマスクに使用して基準となるゴールデン RBD リードバック ファイルと RBD リードバック ファイル比較する方法です。この方法では、リードバック データ ストリームの開始と RBD および MSD ファイルの開始が 1:1 で対応しているため、リードバック、マスク、および予想されたデータを容易に揃えることが可能です。

RBD および MSD ファイルには、リードバック データ、マスク データ、およびファイル名の一覧などを含むファイル ヘッダーが ASCII 形式で含まれます。このヘッダー情報は、無視または削除してください。スクリプトまたはテキスト エディターを使用して RBD および MSD ファイル内の ASCII の 1 および 0 は、デバイスからのリードバック データのバイナリ値に対応します。これらのファイルは、バイナリ ソースとしてではなく、テキスト ファイルとして処理してください。スクリプトまたはテキスト エディターを使用して RBD および MSD

ファイルをバイナリ形式に変換すると、システムによっては検証過程が簡略化され、1/8 までファイル サイズを縮小できます。

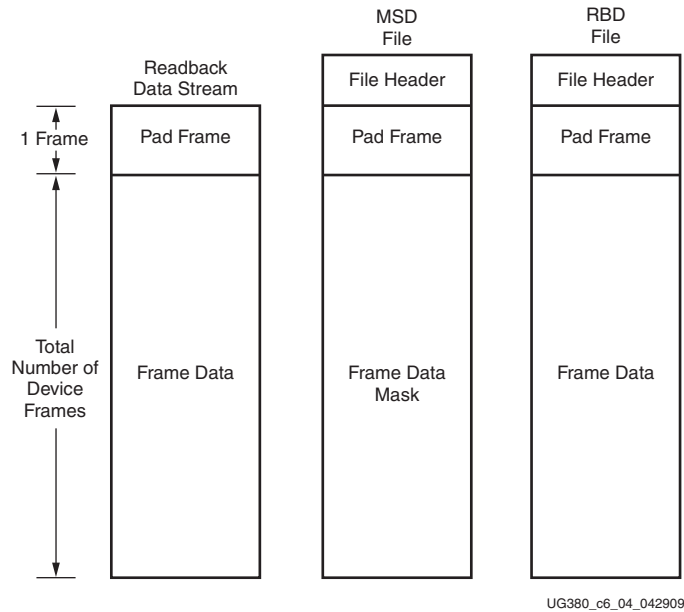


図 6-4 : MSD および RBD ファイルを使用したリードバック データの比較

この方法のデメリットは、最初のコンフィギュレーションビットストリームおよびMSD ファイルを格納するだけでなく、基準となるゴールデン RBD ファイルを格納しなければならず、必要な格納容量が増加する点です。

2. MSK ファイルをマスクに使用し、リードバック データとコンフィギュレーション BIT ファイルを比較します (図 6-5 参照)。

もう 1 つの検証方法では、MSK ファイルを使用してリードバック ビットをマスクし、リードバック データ ストリームとオリジナルのコンフィギュレーション ビットストリームにある FDRI 書き込み内のフレーム データを比較します。

デバイスにリードバック コマンドを送信後、リードバック フレーム データの先頭と BIT および MSK ファイル内の FDRI 書き込みの先頭が位置合わせられ、比較が開始します。FDRI 書き込みの最後まで比較を行うと、処理が完了します。

この方法の場合、BIT、MSK ファイルとリードバック コマンドのみ格納する必要があるため、必要とするインシステム格納スペースが最小限となります。

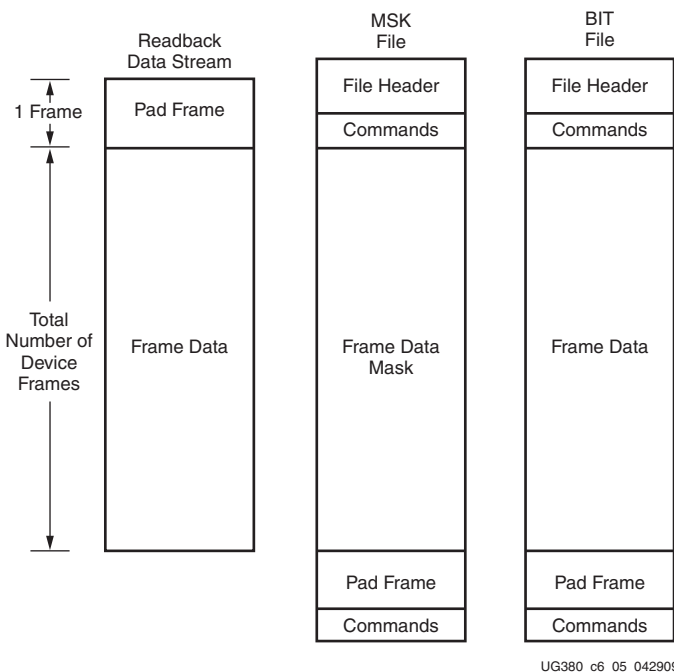


図 6-5 : MSK および BIT ファイルを使用したリードバック データの比較

RBA および RBB ファイルは、予想されたリードバック データと一連のリードバック コマンドを含みます。これらのファイルは、Spartan-6 デバイスよりも Virtex® デバイスのリードバックに適していますが、MSK ファイルと組み合わせて使用するためのものです (アプリケーションノート [XAPP138](#) 『Virtex FPGA シリーズのコンフィギュレーションおよびリードバック』参照)。

# リコンフィギュレーションおよびマルチブート

---

## マルチブートの概要

Spartan®-6 FPGA は再プログラム可能なデバイスで、アプリケーションによっては平常動作中に 1 つまたは複数のビットストリームを FPGA に再度読み込むことができます。1 つの小型 FPGA を複数回再プログラムできるため、これを一度しかプログラムされない大型でコストがかかる ASIC または FPGA と置き換えることが可能となります。

Spartan-6 FPGA は、通常動作中にさまざまな方法で再プログラムできます。ダウンロード コンフィギュレーション モードでは、プロセッサ、マイクロコントローラー、コンピューター、テスターなどの外部インテリジェント エージェントを介して、FPGA を何度でもプログラムし直すことができます。ダウンロード モードは、すべての Spartan-6 FPGA ファミリで使用できます。

Spartan-6 は、接続されている外部メモリからビットストリームを選択して読み込み、再プログラムできるマルチブートという機能を備えています。マルチブート機能では、FPGA アプリケーションの制御の下、複数の FPGA ビットストリームを読み込むことができます。FPGA アプリケーションによってマルチブート操作が開始され、FPGA が異なるコンフィギュレーション ビットストリームでリコンフィギュレーションされます。マルチブート操作開始後は通常どおり FPGA でコンフィギュレーション プロセスが再開されます。FPGA でコンフィギュレーション メモリが消去される間は INIT\_B ピンが Low になり、マルチブート操作が正しく完了するまで DONE 出力ピンは Low のままになります。

マルチブートは、SPI x 1、x 2、x 4、BPI コンフィギュレーション モード、および AES 暗号でサポートされています。

## フォールバック マルチブート

## フォールバック動作

Spartan-6 FPGA にはマルチブート専用のロジックがあり、フォールバックおよびマルチブート (IPROG) リコンフィギュレーションの両方に使用されます。フォールバックまたは IPROG が実行されると、内部生成されたパルスによって、専用のマルチブート ロジックおよび BOOTSTS、MODE、GENERAL1.5 レジスタを除く、コンフィギュレーション ロジック全体がリセットされます。詳細は、[図 7-1](#) を参照してください。このリセットで INIT\_B および DONE が Low になり、コンフィギュレーション メモリがクリアされることによって、コンフィギュレーション プロセスが再開します。

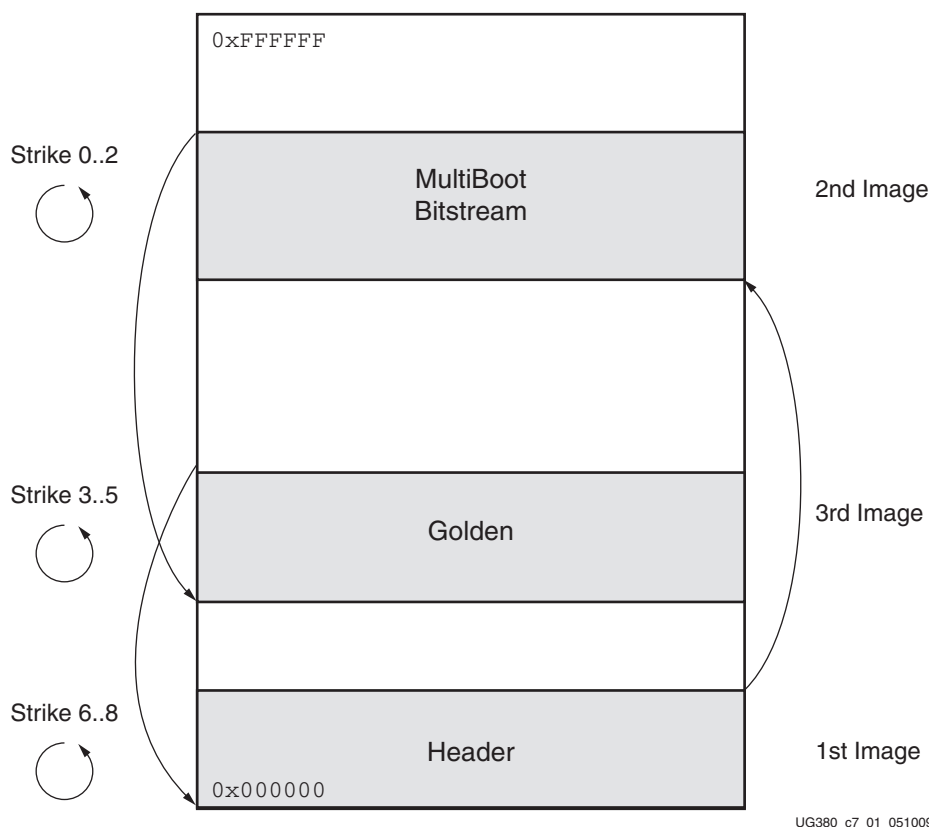


図 7-1：マルチブート ロジック

コンフィギュレーション実行中、IDCODE エラー、CRC エラー、またはウォッチドッグ タイマーのタイムアウト エラーによって、フォールバックがトリガーされます。ウォッチドッグ タイマーはマスター コンフィギュレーション モードでのみアクティブです。タイムアウト値は、BitGen の **-g TIMER\_CFG** スイッチを使用して設定できます。このスイッチの後には、同期ワードのタイムアウトの検出前に許容されるコンフィギュレーション クロック数を示す、16h'0201 よりも大きい 16 ビット値が続きます。

フォールバック リコンフィギュレーション中は、そのイメージの許容値まで **BOOTSTS** レジスタに格納されているストライク数が増加し、リコンフィギュレーションが継続します。その制限に達しない場合、FPGA は **MODE** レジスタの **NEW\_MODE** ビットをチェックします。この値が 0 の場合、デバイスはモード ピンにより定義されているコンフィギュレーション モードを使用します。1 の場合は、**MODE** レジスタに含まれている **BOOTMODE** ビットで定義されているコンフィギュ



レーション モードを使用します。NEW\_MODE レジスタは、BitGen の  
**-g Next\_Config\_New\_Mode:Yes** オプションで設定し、BOOTMODE ビットは、  
**-g Next\_Config\_Boot\_Mode** オプションで設定します。

マルチブート コンフィギュレーションには 3 つのイメージがあります。最初のイメージはヘッダーです。この小さなビットストリームには同期ワードが含まれており、次のビットストリームおよびフォールバックまたはエラーのないゴールデン ビットストリームのアドレスを設定して IPROG コマンドで終了します。このビットストリームを自動的に生成するには、ゴールデン ビットストリームのプログラム ファイル作成時に BitGen の **-g next\_config\_addr** オプションを追加します。

2 番目のイメージはマルチブート ビットストリームで、最初にコンフィギュレーションするビットストリームです。このビットストリームの位置は、GENERAL1、2 の値によって定義されます。GENERAL 2 レジスタの上位 8 ビットは、不揮発性デバイスのオペコード用に予約されています。詳細は、[第 5 章「コンフィギュレーションの詳細」](#)を参照してください。

3 番目のイメージはフォールバックまたはゴールデン ビットストリームです。これはコンフィギュレーション中に一貫してエラーが発生しても安全とみなされます。このビットストリームの位置は、GENERAL3、4 の値によって定義されます。GENERAL1、2 と同様、GENERAL4 レジスタの上位 8 ビットは不揮発デバイスのオペコード用に予約されています。

コンフィギュレーション フォールバックが発生してゴールデン ビットストリームに到達した場合、マルチブート ビットストリーム (GENERAL1、2 に位置) に戻るには、PROGRAM\_B ピンをトグルするか、またはデバイスの電源を投入し直す必要があります。

3 つ以上のビットストリームを使用するデザインでは、GENERAL1、2 の値を次のビットストリームの位置に設定し、IPROG コマンドを出力する必要があります。GENERAL3、4 の値は、フォールバック ビットストリーム用に予約しておく必要があります。

ヘッダー イメージはアドレス 0 で開始する必要があるため、3 つのストライクが割り当てられています。CRC エラーが検出されると、レジスタ設定 RESET\_ON\_ERROR が 1 (COR2 レジスタに格納されており BitGen の **-g Reset\_on\_err** オプションで設定可能) で、ストライク数が 3 未満の場合は、ストライク数が増加します。ウォッチドッグ タイマーがタイムアウトすると同様の動作が生じますが、それは RESET\_ON\_ERROR によるものではありません。ストライク カウンターは BOOTSTS レジスタに含まれています。ストライク数が 3 になると、INIT および DONE ピンが Low になりコンフィギュレーションが停止します。

マルチブート イメージは GENERAL1、2 で指定された任意のアドレスに格納でき、3 つのストライクが割り当てられています。エラーが検出されると、ストライク数が 3 未満で RESET\_ON\_ERROR が 1 の場合はストライク数が増加し、コンフィギュレーションが GENERAL1、2 で指定されたアドレスで再開します。ストライク数が 3 になると、コンフィギュレーションが GENERAL3、4 にあるフォールバック ビットストリームに移動します。ストライク数を消去するには、FPGA 電源を再投入するか、または PROGRAM\_B ピンをパルスします。

フォールバック (またはゴールデン) イメージは GENERAL3、4 で指定された任意のアドレスに格納でき、3 つのストライクが割り当てられています。エラーが検出されると、ストライク数が 6 未満の場合はストライク数が増加してコンフィギュレーションが GENERAL3/4 で指定された任意のアドレスで再開します。ストライク数が 6 となるのは、マルチブート イメージとストライク カウンターが共有されるためです。ストライク数が 6 になると、コンフィギュレーションがヘッダーイメージが格納されている 0 に戻ってブートします。これが発生すると、マルチブート イメージとフォールバック イメージの両方を 3 回試行してからコンフィギュレーションが停止します。この結果ストライク数は 9 になります。

フォールバック リコンフィギュレーションの完了後、ユーザー デザインでは、STATUS または BOOTSTS レジスタをリードバックし、フォールバックが適切に完了したかを検証する必要があります。フォールバック コンフィギュレーションが正しく実行されている場合は、ストライク カウ

ントレジスタが保持され、後続のソフト リブートで GENERAL3、4 に格納されているアドレス (ゴールデン イメージ) が使用されます。ストライク数を消去するには、ハード リブート (PROGRAM\_B ピンのパルス) を実行するか、または電源を再投入します。

フォールバック リコンフィギュレーションが 3 回失敗すると、INIT\_B および DONE の両方が Low に維持されます。

AES が有効で、スレーブ コンフィギュレーション モードの場合は、フォールバックは無効です。

## IPROG リコンフィギュレーション

IPROG (内部 PROGRAM\_B) コマンドは、専用リコンフィギュレーション ロジックをリセットしない点を除き、PROGRAM\_B ピンのパルスと同様の機能があります。リコンフィギュレーション中は、デフォルトのアドレス (ゼロ) ではなく、GENERAL1、2 で設定した開始アドレスが使用されます。フォールバック (ゴールデン) ビットストリームは GENERAL3、4 で設定します。IPROG コマンドは、ICAP\_SPARTAN6 またはビットストリームを使用して送信できます。

### ICAP\_SPARTAN6 を使用したリブート

IPROG コマンドは、ICAP\_SPARTAN6 プリミティブを使用して送信できます。コンフィギュレーションが正しく完了したら、ユーザー デザインでマルチブート ビットストリームの開始アドレスが判断され、GENERAL1、2 レジスタの設定後に ICAP を使用して IPROG コマンドを送信します。

コマンドのシーケンスは次のとおりです。

1. 同期ワードを送信する。
2. 次のビットストリームの開始アドレス用に GENERAL1、2 レジスタを、そして読み出し動作用に不揮発性デバイスのオペコードをプログラムする。さらに、フォールバック (ゴールデン) ビットストリームの開始アドレス用に GENERAL3、4 レジスタを、そして読み出し動作用に不揮発性デバイスのオペコードをプログラムする。
3. IPROG コマンドを送信する。

表 7-1 に、ICAP を使用する IPROG コマンドのビットストリーム例を示します。

表 7-1 : ICAP を使用する IPROG コマンドのビットストリーム例

コンフィギュレーション データ (16 進数) <sup>(1)</sup>	説明
FFFF	ダミー ワード
AA99	同期ワード
5566	同期ワード
3261	タイプ 1、GENERAL_1 に 1 ワード書き込み
XXXX	マルチブート開始アドレス [15:0]
3281	タイプ 1、GENERAL2 に 1 ワード書き込み <sup>(2)</sup>
XXXX	オペコードおよびマルチブート開始アドレス [23:16]
32A1	タイプ 1、GENERAL3 に 1 ワード書き込み
XXXX	フォールバック開始アドレス [15:0]
32C1	タイプ 1、GENERAL4 に 1 ワード書き込み <sup>(2)</sup>

表 7-1 : ICAP を使用する IPROG コマンドのビットストリーム例 (続き)

コンフィギュレーション データ (16 進数) <sup>(1)</sup>	説明
XXXX	オペコードおよびフォールバック開始アドレス [23:16]
30A1	タイプ 1、CMD に 1 ワード書き込み
000E	IPROG コマンド
2000	タイプ 1、NOOP

注記：

1. SelectMAP 16 ビット データ順が ICAP データ バスに適用されます。ビット順の詳細は、[41 ページの表 2-5](#) を参照してください。
2. GENERAL2 および GENERAL4 レジスタの上位 8 ビットは、不揮発性デバイスの読み出し命令用のオペコードを示しています。正確なオペコードについては、ストレージ デバイスのデータシートを参照してください。共通するコードは、Fast Read コマンドの 0x0B、Dual Fast Read コマンドの 0x3B、および Quad Fast Read コマンドの 0x6B です。

コンフィギュレーション ロジックが IPROG コマンドを受信すると、FPGA で専用リコンフィギュレーション ロジックを除くすべてがリセットされ、INIT\_B および DONE ピンが Low になります。FPGA ですべてのコンフィギュレーション メモリがクリアされると、INIT\_B は再び High になります。次に、GENERAL1、2 の値がビットストリームの開始アドレスとして使用されます。

## フォールバックおよび IPROG リコンフィギュレーションに関連するステータス レジスタ

Spartan-6 デバイスには、コンフィギュレーション履歴を保存する BOOTSTS レジスタがあります。このレジスタは、2 段 FIFO のように動作します。最新のコンフィギュレーション ステータスを Status\_0 に格納すると、Status\_0 の現在の値が Status\_1 にシフトします。Valid\_0 ビットは、Status\_0 のほかのビットが有効であるかどうかを示します。詳細は、[111 ページの「ブート履歴のステータス レジスタ \(BOOTSTS\)」](#)を参照してください。

表 7-2 ～ 表 7-4 に、に、一般的な状況での BOOTSTS レジスタの値を示します。

表 7-2 : 最初のビットストリームのコンフィギュレーション後のステータス (エラーなし)

	CRC_ERROR	ID_ERROR	WTO_ERROR	IPROG	FALLBACK	VALID
Status_1	0	0	0	0	0	0
Status_0	0	0	0	0	0	1

表 7-3 : 最初のコンフィギュレーションの後に IPROG が発生

	CRC_ERROR	ID_ERROR	WTO_ERROR	IPROG	FALLBACK	VALID
Status_1	0	0	0	0	0	1
Status_0	0	0	0	1	0	1

表 7-4：最初のビットストリームに IPROG を埋め込み、2 番目のビットストリームで CRC エラーが発生、フォールバックに成功した場合

	CRC_ERROR	ID_ERROR	WTO_ERROR	IPROG	FALLBACK	VALID
Status_1	1	0	0	1	0	1
Status_0	0	0	0	1	1	1

注記：

1. Status\_1 は、IPROG が実行され、そのビットストリームで CRC\_ERROR が検出されたことを示しています。
2. Status\_0 は、フォールバック ビットストリームが問題なく読み込まれたことを示しています。この場合、フォールバック ビットストリームに IPROG コマンドが含まれているため、IPROG ビットもセットされています。フォールバック実行時は IPROG コマンドは無視されますが、ビットストリームにコマンドが含まれていたことが履歴に記録されます。

## ウォッチドッグ タイマー

Spartan-6 FPGA のウォッチドッグ タイマーは同期ワードの検出を監視するために使用します。ウォッチドッグ タイマーがタイムアウトになると、BitGen の **-g Reset\_On\_Err** オプションが Yes に設定されていて最大ストライク数に達していない場合は、コンフィギュレーション ロジックでストライク数が増加します。詳細は、「[フォールバック マルチブート](#)」を参照してください。

ウォッチドッグ タイマーはコンフィギュレーション クロックと同じクロック ソースを使用します。ウォッチドッグ カウンターの制限は、CWDT (Configuration WatchDog Timer) レジスタまたは BitGen の **TIMER\_CFG** オプションで設定できます。デフォルトは 64k クロック サイクルで、最小値は 16h'0201 です。

ウォッチドッグ タイマーはユーザー設定によって無効にできません。フォールバック リコンフィギュレーション中とその完了後は、ウォッチドッグは無効です。

## マルチブート イメージ間に必要なデータ空間

Spartan-6 FPGA のマルチブート アドレス指定は柔軟性が高く、どのバイト境界からでもビットストリームを開始できますが、アプリケーション要件に基づいて一部制限があります。

### フラッシュ セクター、ブロック、ページの境界

Spartan-6 FPGA は、マルチブート コンフィギュレーション イメージを外部フラッシュ PROM から読み込みます。すべてのフラッシュ PROM には内部メモリ アーキテクチャがあり、このアーキテクチャによってメモリがセクター、ブロック、またはページに配置されます。ほとんどの PROM に複数のセクターがあります。一部のアーキテクチャでは 1 つのセクターを小ブロックまたはページに分割する機能も提供されています。

Spartan-6 FPGA マルチブート コンフィギュレーション イメージは、セクター、ブロック、またはページの境界に揃えられているのが理想的です。フラッシュ PROM のアーキテクチャによって異なる、特別な要件があります。フラッシュ PROM で消去できる最小エレメントがセクターの場合は、FPGA ビットストリームをセクターの境界に揃える必要があります。これにより、1 つの FPGA ビットストリームを PROM 内のその他のビットストリームに影響を与えずにアップデートできます。

## LCK\_Cycle に必要な追加のメモリ空間

Spartan-6 FPGA アプリケーションには、1 つまたは複数のデジタル クロック マネージャー (DCM) または位相ロック ループ (PLL) を含めることができます。LCK\_Cycle BitGen の設定により、コンフィギュレーション中に、すべてのクロック エLEMENTがそれぞれの入力クロック周波数を取得し、ロックするまで FPGA を待機させてコンフィギュレーション プロセスを終了させないようにすることができます。このロック時間は、DCM モードまたは PLL モード、入力クロック周波数によって異なります。詳細は、データシート [DS162](#) 『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』を参照してください。

コンフィギュレーションを完了する前に FPGA で 1 つまたは複数のクロック エLEMENTがロックするのを待機する場合でも、FPGA のコンフィギュレーション コントローラーでは次の同期ワードが継続して検索されます。2 つのマルチブート イメージが連続して配置されていて、最初の FPGA ビットストリームに LCK\_Cycle オプションが設定されている DCM または PLL が含まれる場合、コンフィギュレーションで問題が発生する可能性があります。現在のコンフィギュレーションが完了する前にコントローラーで 2 番目の FPGA ビットストリームに同期ワードが検出されると、2 番目のビットストリームのデータ処理が開始されます。しかし、FPGA のコンフィギュレーション ロジックは FPGA が 2 番目のビットストリームからデータを読み出しても現在のコンフィギュレーションを完了できます。このような状況が発生する場合は、ビットストリーム間に十分な余裕を用意する必要があります。



## リードバック CRC

---

Spartan®-6 デバイスには、ユーザー デザインのバックグラウンドで継続的にコンフィギュレーション データのリードバックを実行する機能があります。この機能は、コンフィギュレーション メモリのビットをフリップさせる SEU (Single Event Upset) を簡単に検出できるようにするものです。検出されたエラーは、デバイス ピン (INIT\_B) または内部アクセス可能なコンポーネント POST\_CRC\_INTERNAL に現れます。リードバックのクロック ソースは外部のものでも、内部生成したものでもかまいません。

リードバック CRC の最初のスキャン値を基準値としてラッチし、後続するリードバック CRC 値はこの基準値と比較されます。CRC の不一致が検出された場合は、POST\_CRC\_INTERNAL プリミティブの CRCERROR ピンが High 駆動され、INIT\_B ピンは Low 駆動され、DONE ピンは High のままとなります。オプションとして、CONFIG ユーザー プリミティブの POST\_CRC\_INIT\_FLAG 属性を DISABLE に指定し、INIT\_B をリードバック CRC のフラグとして使用しないよう設定できます。このエラー フラグはクリアされるまで High のままになります。

次の場合に、リードバック CRC が停止し、エラー フラグがクリアされます。

- SYNC または DESYNC ワードが検出される
- JTAG TAP コントローラーがリセットされる
- スレーブ SelectMAP または ICAP アクセスから ABORT がトリガーされる
- IPROG (内部プログラム) コマンドが受信される
- Suspend モードが有効である
- デバイスが、リードバック シャットダウン、JSHUTDOWN、または ISC\_ENABLE などのシャットダウン モードのとき

リードバック CRC は、ユーザーのコンフィギュレーション アクセスに影響を与えることなく、自動的に停止し、エラー フラグはクリアされます。リードバックを停止させる状況から抜け出たとき、CRC 基準値は再計算され、POST\_CRC が ENABLE に設定されている場合は自動的に再開します。

リードバック CRC ロジックは、次のような条件で実行されます。

- FPGA のスタートアップが正しく完了し、DONE ピンが High なっている。
- DESYNC コマンドを使用してコンフィギュレーション処理を終了し、コンフィギュレーション ロジックをアクセス可能な状態にしておくこと。DESYNC コマンドが発行されない場合、リードバック CRC ロジックはコンフィギュレーション ロジックにアクセスできず、実行されません。
- JTAG 命令レジスタ (IR) にコンフィギュレーション命令 (CFG\_IN、CFG\_OUT、または ISC\_ENABLE) が含まれないこと。これらの命令が格納されていると、リードバック CRC ロジックはコンフィギュレーション ロジックにアクセスできず、実行されません。JTAG インターフェイスを使用してコンフィギュレーションを実行する場合は、コンフィギュレーション

の最後にこれら 3 つのコンフィギュレーション命令以外の値を IR に読み込む必要があります。

バックグラウンドで実行されるリードバックでは、動作中に変更される次のメモリ位置はマスクされます。

- 分散 RAM またはシフト レジスタとしてコンフィギュレーションされているルックアップ テーブル (LUT) はチェックされません。Spartan-6 FPGA では、SLICEM のみがこれらのメモリ エlementとしてコンフィギュレーション可能です。また、LUT マスクの都合上、1 つのフレームに LUTRAM または SRL16 として同じ垂直線上に並んでいる LUT はチェックされません。リードバック CRC の適用範囲が最大限になるようにするには、メモリとして使用されている LUT とロジックとして使用されている LUT を別のフレームに置く必要があります。
- ブロック RAM の内容は動作中に変更できるため、初期コンフィギュレーションと同じものではありません。つまり、これらのエレメントはチェックされません。
- PLL DRP の使用はマスクされません。PLL を変更すると CRC エラーとなります。
- 上辺および下辺にある I/O インターフェイス DRP がマスクできるだけでなく、同じフレームにある CLB の LUT もマスクできます。同様に、上辺および下辺にある LUT をマスクすると、I/O インターフェイスもマスクできます。

表 8-1 に示すように、リードバック CRC のクロック ソースはモードによって異なります。

表 8-1： リードバック CRC のクロック ソース

ICAP プリミティブ	マスター モード	スレーブ モード	JTAG モード	クロック ソース
インスタンスエートしている	X	X	X	ICAP プリミティブの CLK 入力
インスタンスエートしていない	Yes	No	No	コンフィギュレーション制約 POST_CRC_FREQ で制約されている周波数の内部オシレーター
インスタンスエートしていない	No	Yes	No	CCLK ピン入力
インスタンスエートしていない	No	No	Yes	コンフィギュレーション制約 POST_CRC_FREQ で制約されている周波数の内部オシレーター

JTAG は、最も優先度の高いコンフィギュレーション モードであるため、必要な場合はコンフィギュレーション バスを優先的に使用します。JTAG 命令レジスタが、CFG\_IN、CFG\_OUT、または ISC\_ENABLE 命令で停止しないようにしてください。

## POST\_CRC 制約

Spartan-6 FPGA デバイスでは、SEU イベント用の制約がいくつかあり、これらの制約の伝搬には同じ規則があります。これらは、CONFIG ブロックに属性として指定された後、物理的なデザイン オブジェクトに伝搬されます。

## POST\_CRC

POST\_CRC は FPGA のリードバック CRC 機能を有効にします。SEU イベントの信号送信に POST\_CRC\_INTERNAL プリミティブの CRCERROR ピンを使用します。デフォルトでは、INIT が SEU CRC エラー インジケータとして予約されていますが、POST\_CRC\_INIT\_FLAG 制約を設定して無効にできます。



この情報の伝搬には、POST\_CRC 制約の適用が最良の方法です。POST\_CRC 制約は CONFIG 制約です。この制約は PAR および BitGen で使用可能で、IOB が INIT ピンを駆動しないようにプログラムすることで、INIT ピンを予約できます。

POST\_CRC に設定できる値は次の 2 つです。

- ENABLE  
SEU の検出が有効になります。
- DISABLE  
SEU の検出が無効になります。

## POST\_CRC\_INIT\_FLAG

POST\_CRC\_INIT\_FLAG は、Spartan-6 FPGA の INIT\_B ピンが SEU エラー信号のソースであるかを決定します。

コンフィギュレーション中、INIT ピンは通常どおり動作します。SEU 解析が有効、かつ INIT が予約されていると、デフォルトの INIT ピンは、コンフィギュレーション後に SEU ステータスピンとして機能します。SEU は、リアルタイムで計測された CRC 値があらかじめ計測された値と異なり、CRCERROR ピンが High に駆動、INIT ピンが Low に駆動する場合に検出されます。

INIT\_B ピンをリードバック CRC のエラー ステータス出力ピンとして使用できないようにするには、POST\_CRC\_INIT\_FLAG を使用してください。このような場合も、エラー ステータスは POST\_CRC\_INTERNAL サイトからわかります。

POST\_CRC\_INIT\_FLAG に設定できる値は次の 2 つです。

- ENABLE  
INIT\_B ピンを SEU エラー信号のインジケータとして使用します (デフォルト)。
- DISABLE  
INIT\_B ピンは SEU エラー信号のインジケータとして使用されません。エラー ステータスは POST\_CRC\_INTERNAL プリミティブからのみわかります。

## POST\_CRC\_ACTION

POST\_CRC\_ACTION は CRC エラー検出後のリードバック CRC の動作を決定します。

POST\_CRC\_ACTION に設定できる値は次の 2 つです。

- HALT  
CRC エラーが検出されると、それ以降リードバック CRC テストは行われません。エラーがクリアされると、CRC テストが再開します (デフォルト)。
- CONTINUE  
CRC エラーが検出されると、エラー フラグが出力されますが、テストは続行します。

## POST\_CRC\_FREQ

POST\_CRC\_FREQ はリードバック CRC ロジックに対して内部生成されたクロックの周波数を決定します。

POST\_CRC\_FREQ に設定できる値は 2、4、6、10、12、16、22、26、33、40、および 50 です。

これらの値は、具体的な周波数を直接示すものではありません。各値に関連した周波数の値は、『Spartan-6 FPGA データシート：DC 特性およびスイッチ特性』を参照してください。

## 構文例

ここでは、各制約でサポートされている構文の例を示します。

### POST\_CRC

UCF の構文例

```
CONFIG POST_CRC = [ENABLE|DISABLE]
```

### POST\_CRC\_INIT\_FLAG

UCF の構文例

```
CONFIG POST_CRC_INIT_FLAG = [ENABLE|DISABLE]
```

### POST\_CRC\_ACTION

UCF の構文例

```
CONFIG POST_CRC_ACTION = [HALT | CONTINUE]
```

### POST\_CRC\_FREQ

UCF の構文例

```
CONFIG POST_CRC_FREQ = [2|4|6|10|12|16|22|26|33|40|50]
```

# アドバンス コンフィギュレーション インターフェイス

## シリアル デイジー チェーン

Spartan®-6 デバイスをシリアル デイジー チェーン接続すると、1 つのコンフィギュレーション ソースから複数の Spartan-6 デバイスのコンフィギュレーションが可能になります。シリアル デイジー チェーンでは、デバイスは DIN ピンでコンフィギュレーション データを取得し、DOUT ピンから取得したコンフィギュレーション データを下位デバイスへ伝搬します。ココンフィギュレーション データ ソースに最も近いデバイスを最上位デバイス、ソースから最も遠いデバイスを最下位デバイスと呼びます。

シリアル デイジー チェーンの場合、コンフィギュレーション クロックは、通常マスター シリアル モードの最上位デバイスから供給されます。その他すべてのデバイスは、スレーブ シリアル モードに設定されます。図 9-1 に、このコンフィギュレーションを示します。

最初のデバイスに SPI モードを使用する方法もあります。SPI モードの場合も、デイジー チェーンのデータは DOUT から出力されます。

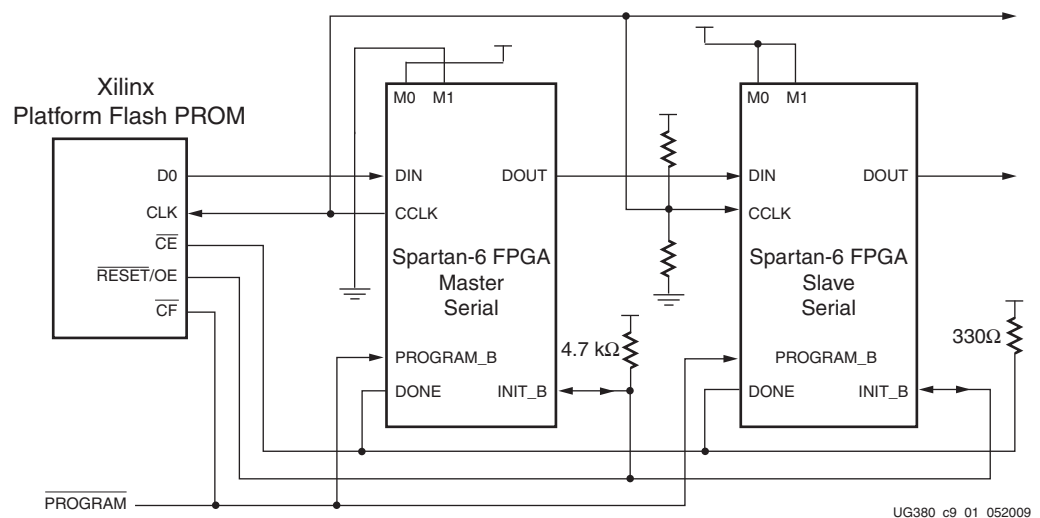


図 9-1: マスター シリアル モード/スレーブ シリアル モードの  
デイジー チェーン コンフィギュレーション

図 9-1 について説明します。

1. DONE ピンは、デフォルトでは、外部プルアップ抵抗が必要なオープン ドレイン出力です。先頭を除くすべてのデバイスで、DONE ピンのアクティブ ドライバーを無効にしてください。先

頭のデバイスにある DONE ピンのアクティブ ドライバーは、有効にできます。詳細は、「[シリアル デイジー チェーンのガイドラインおよびデザインの考察](#)」を参照してください。

2. INIT\_B ピンは、双方向オープン ドレイン ピンで、外部プルアップ抵抗が必要です。
3. BitGen のスタートアップ クロック設定で、CCLK をシリアル コンフィギュレーションに設定する必要があります。
4. この図の PROM は、1 つまたは複数のザイリンクス PROM を示しています。複数のザイリンクス PROM をカスケード接続することで、データの格納範囲全体を拡大できます。
5. BIT ファイルは、PROM へ格納する前に PROM ファイルに変換する必要があります。
6. CCLK ネットにはテブナン並列終端が必要です。詳細は、[57 ページの「コンフィギュレーション クロック \(CCLK\) のボード レイアウト」](#)を参照してください。
7. シリアル デイジー チェーンは、プラットフォーム フラッシュ (XCFxxS および XCFxxP) PROM と SPI シリアル フラッシュ専用です。

シリアル デイジー チェーンの先頭のデバイスが最後にコンフィギュレーションされます。CRC チェックでは、現在作業中のデバイスのデータのみがチェックされ、チェーンのほかのデバイスはチェックされません。

チェーンの最後のデバイスがコンフィギュレーションを完了し、CRC チェックをクリアすると、スタートアップ シーケンスが開始されます。スタートアップ シーケンスの DONE ピンのリリース段階で、チェーンの最後から 1 つ前のデバイスのコンフィギュレーション中に、デバイスが DONE ピンをハイインピーダンスにします。すべてのデバイスが DONE ピンをリリースした後、共有している DONE 信号は外部で High にプルアップ、またはチェーンの最初のデバイスで High 駆動されます。すべてのデバイスは CCLK の次の立ち上がりエッジで DONE ピンのリリースを停止し、スタートアップ シーケンスを完了します。

このとき、スレーブ シリアル デイジー チェーンのすべての DONE ピンが接続されている必要があります。また、シリアル デイジー チェーンの最初のデバイスのみ、DONE のアクティブなプルアップ ドライバーを有効にする必要があります。下位デバイスの DONE のドライバーを有効にすると、DONE 信号の競合が生じます。

## 混在シリアル デイジー チェーン

Spartan-6 デバイスは、Spartan-3、Virtex®-4、Virtex-5 ファミリーとデイジーチェーンで接続できます。デバイスが混在したシリアル デイジー チェーン接続では、次の 3 つの重要な項目を考慮する必要があります。

- 旧型デバイスの多くは、Spartan-6 デバイスが生成する高速な CCLK 周波数に対応できません。チェーン接続しているすべてのデバイスがサポート可能な CCLK 周波数を選択してください。
- Spartan-6 デバイスは常に、シリアル デイジー チェーンの最初に配置し、旧ファミリのデバイスはチェーンの最後に配置してください。
- すべての Spartan-6 デバイス ファミリーには、類似した BitGen オプションがあります。Spartan-6 FPGA の BitGen オプションのガイドラインは、シリアル デイジー チェーン接続された Virtex ベースのすべてのデバイスに適用されます。
- デバイスが、DOUT ピンを通わせることができるコンフィギュレーション ビット数には制限があり、これはデバイス ファミリーごとに異なります ([表 9-1](#) 参照)。下位デバイスすべてのビットストリーム長の合計が、[表 9-1](#) に示す各デバイス ファミリーの制限値を超えないようにしてください。

表 9-1: デバイス ファミリー別のコンフィギュレーション ビット数の最大値

アーキテクチャ	最大 DOUT ビット数
Spartan-6、Spartan-3、Virtex-6、Virtex-5、Virtex-4、Virtex-II Pro、および Virtex-II デバイス	$32 \times (2^{27} - 1) = 4,294,967,264$
Virtex、Virtex-E、Spartan-II、および Spartan-IIE デバイス	$32 \times (2^{20} - 1) = 33,554,216$

## シリアル デイジー チェーンのガイドラインおよびデザインの考察

シリアル デイジー チェーンの場合、考慮すべき項目が多数あります。

### スタートアップ シーケンス (GTS)

GTS のリリースは、DONE のサイクルより後または同じサイクルで行い、すべての DONE ピンがリリースされたときに Spartan-6 デバイスが確実に動作を開始できるようにします。

### アクティブ DONE ドライバー

最初のデバイスを除くデバイスすべての DONE ピンのドライバを有効にしてください (ソフトウェア設定については、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) の「BitGen」を参照してください。チェーンの最初のデバイスが最後にプログラムされます。

- DriveDone が無効になります (最初のデバイス以外すべて)。
- DriveDone が有効になります (最初のデバイス)。

また、すべての DONE ピンのドライバーを無効にし、外部プルアップ抵抗を追加して、すべてのデバイスが DONE ピンをリリースした後に信号を High にプルアップする方法もあります。

### すべての DONE ピンを接続

シリアル デイジー チェーンに接続されたデバイスすべての DONE ピンを接続してください。DONE ピンを接続していないとコンフィギュレーションできません。デバッグを目的として、共有している DONE 信号から DONE ピンを個別に未接続にすると非常に便利です。デバッグ後には、切り離されたデバイスは、シリアルまたは JTAG インターフェイスを使用して個別にコンフィギュレーション可能です。

### DONE ピンの立ち上がり時間

すべての DONE ピンのリリース後、DONE ピンは CCLK の 1 サイクル以内にロジック 0 からロジック 1 に立ち上がる必要があります。これには、外部プルアップ抵抗が必要です。DDONE 信号が立ち上がるために追加時間を要する場合は、シリアル デイジー チェーンのすべてのデバイスに BitGen の DonePipe オプションを設定してください。ソフトウェア設定の詳細は、『コマンド ライン ツール ユーザー ガイド』([UG628](#)) の「BitGen」を参照してください。

### ビットストリームのフォーマット

複数のビットストリームが送信されていることを FPGA に知らせ、データが下位デバイスに転送されるようにビットストリームをカスタマイズする必要があります。これには、iMPACT プログラミング ツールに含まれる PROM ファイルをフォーマットするためのツール、PROMGen を使用してください。

## ギャング シリアル コンフィギュレーション

ギャング シリアル コンフィギュレーションを使用すると、同じビットストリームから複数のデバイスを同時にコンフィギュレーションできます (図 9-2 参照)。この構成では、シリアル コンフィギュレーション ピンを図のように結束するため、各デバイスにおける信号遷移が同一になります。通常、CCLK を駆動するために 1 つのデバイスをマスター シリアル モードに設定し、その他のデバイスはスレーブ シリアル モードに設定します。ギャング シリアル コンフィギュレーションの場合、すべてのデバイスが同一でなければなりません。コンフィギュレーションは PROM または外部のコンフィギュレーション コントローラーで実行されます。

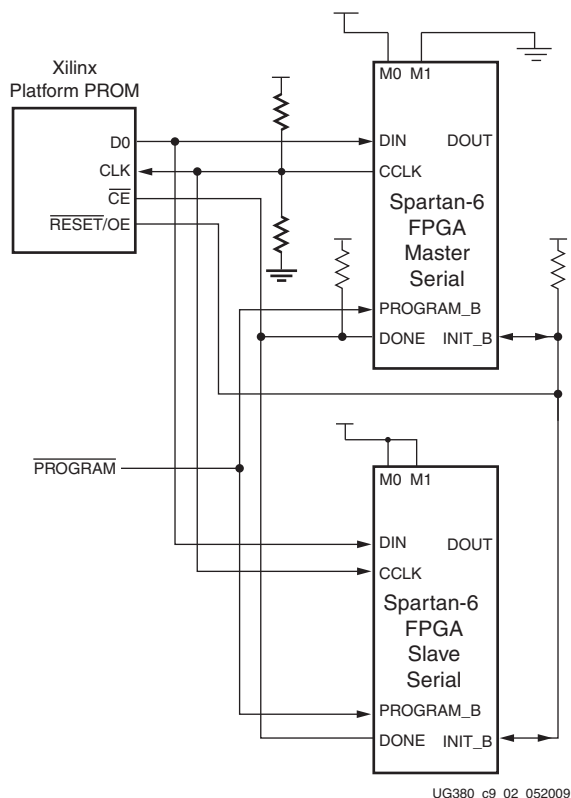


図 9-2: ギャング シリアル コンフィギュレーション

図 9-2 について説明します。

1. 1 つのデバイスがマスター モードに設定されている場合、すべてのデバイスを CCLK の正確に同じサイクルでスタートアップできない可能性があるため、ギャング シリアル コンフィギュレーションでは、すべてのデバイスの DONE ドライバー (オプション) を無効にする必要があります。これには、外部プルアップ抵抗が必要となります。
2. INIT\_B ピンは、双方向オープンドレイン ピンで、外部プルアップ抵抗が必要です。
3. BitGen のスタートアップ クロック設定で、CCLK をシリアル コンフィギュレーションに設定する必要があります。
4. この図の PROM は、1 つまたは複数のザイリンクス PROM の場合を示しています。複数のザイリンクス PROM をカスケード接続することで、データ格納範囲全体が拡大できます。
5. BIT ファイルは、PROM に格納前に PROM ファイルに変換する必要があります。詳細は、79 ページの「PROM ファイルの生成」を参照してください。

6. 一部のザイリンクス PROM では、リセット ピンの極性をプログラムできます。この設定で PROM を使用する場合は、RESET をアクティブ Low に設定してください。
7. ギヤング シリアル コンフィギュレーションでは、すべてのデバイスが同一 (IDCODE が同じ) であり、1 つのビットストリームを使用してコンフィギュレーションする必要があります。
8. CCLK ネットにはテブナン並列終端が必要です。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
9. ギヤング シリアル コンフィギュレーションは、プラットフォーム フラッシュ (XCFxxS および XCFxxP) PROM と SPI シリアル フラッシュ PROM 専用です。

ギヤング シリアル コンフィギュレーションの場合、考慮すべき項目が多数あります。

- スタートアップ シーケンス (GTS)

GTS のリリースは、DONE ピンのサイクルより前または同一サイクルで行い、すべての DONE ピンがリリースされたときに、すべてのデバイスが確実に動作開始できるようにします。

- すべてのデバイスの DONE ピンのアクティブ ドライバーを無効にする

ギヤング シリアル コンフィギュレーションでは、DONE ピンが結合接続されていると、各デバイスのスタートアップ シーケンスが異なる可能性があります。すべてのデバイスのアクティブ な DONE ドライバーを必ず無効にしてください。そのため、共通の DONE 信号にプルアップ 抵抗が必要となります。

**-g DriveDone:no** (この BitGen オプションをすべてのデバイスに設定)

- マスター デバイスを使用する場合は、すべての DONE ピンを接続する

1 つの FPGA をマスター デバイスとして使用する場合は、ギヤング シリアル コンフィギュレーションのデバイスすべての DONE ピンを接続する必要があります。DONE ピンを接続しないと、各デバイスのコンフィギュレーションが完了できません。すべてのデバイスをスレーブ シリアル モードに設定すると、DONE ピンを未接続にできます (外部の CCLK ソースがすべての DONE ピンが High になるまでトグルを継続する場合)。

デバッグ目的の場合、共通の DONE 信号から DONE ピンを個別に未接続にすると非常に便利です。

- DONE ピンの立ち上がり時間

すべての DONE ピンのリリース後、DONE ピンは CCLK の 1 サイクル以内にロジック 0 からロジック 1 に立ち上がる必要があります。DONE 信号が立ち上がるために追加時間を要する場合は、シリアル デイジー チェーンのすべてのデバイスに BitGen の **-g DonePipe** オプションを設定してください。

- ボード レイアウト用クロック信号としてのコンフィギュレーション クロック (CCLK)

CCLK 信号は比較的低速ですが、Spartan-6 FPGA の入力バッファのエッジ レートは非常に高速です。CCLK 信号でわずかなシグナル インテグリティの問題が発生しただけで、コンフィギュレーションが正常に完了しない可能性があります (エラーが発生しやすいモード : DONE Low、INIT\_B High)。したがって、シグナル インテグリティが重要となるデザインでは、IBIS を使用したシグナル インテグリティのシミュレーションなどの実行を推奨します。

- 信号のファンアウト

ギヤング シリアル コンフィギュレーションを使用する際は、優れたシグナル インテグリティを実現する必要があります。このため、シグナル インテグリティのシミュレーションの実行を推奨します。

- ギヤング シリアル コンフィギュレーションの PROM ファイル

ギャング シリアル コンフィギュレーションの PROM ファイルは、単一デバイスのコンフィギュレーションに使用する PROM ファイルと類似しています。PROM ファイルに関するガイドラインは特にありません。

## 複数デバイスの SelectMAP コンフィギュレーション

スレーブ SelectMAP モードの複数の Spartan-6 デバイスは、共通の SelectMAP バスに接続可能です (図 9-3 参照)。SelectMAP バスでは、D、CCLK、RDWR\_B、BUSY、PROGRAM\_B、DONE、および INIT\_B ピンがすべてのデバイスで共有されます。各デバイスに個別にアクセスできるようにするには、CSI\_B (チップ セレクト) 入力を共有接続にしないでください。CSI\_B 信号の外部制御が必要となり、通常、これはマイクロプロセッサまたは CPLD によって提供されます。

コンフィギュレーション後にリードバックを実行する場合、RDWR\_B 信号および BUSY 信号を適切に使用してください (第 6 章「リードバックおよび コンフィギュレーションの検証」参照)。

リードバックを実行しない場合は、RDWR\_B を Low に接続し、BUSY を未接続にできます。Spartan-6 デバイスのコンフィギュレーション時に、BUSY 信号をモニターする必要はありません。詳細は、86 ページの「ビットストリームの読み込み (手順 4 ~ 7)」および第 6 章「リードバックおよび コンフィギュレーションの検証」を参照してください。

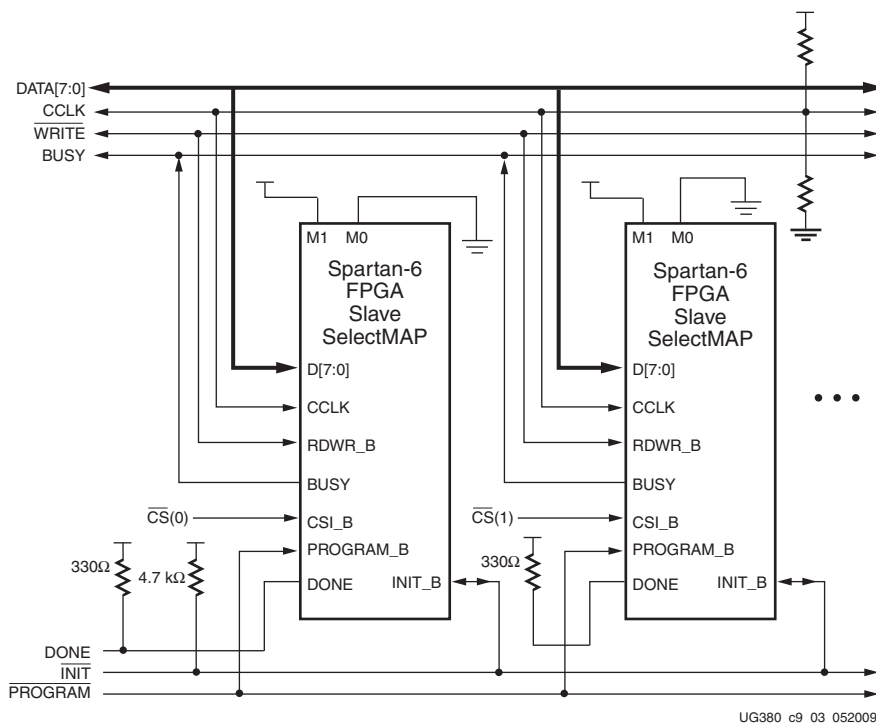


図 9-3 : 8 ビットの SelectMAP バスを使用した  
複数のスレーブ デバイス コンフィギュレーション

図 9-3 について説明します。

1. DONE ピンは、デフォルトでは外部プルアップ抵抗が必要なオープン ドレイン出力です。この接続では、アクティブ DONE ドライバーを必ず無効にしてください。
2. INIT\_B ピンは双方向オープン ドレイン ピンで、外部プルアップ抵抗が必要です。
3. BitGen のスタートアップ クロック設定では、CCLK を SelectMAP コンフィギュレーションに設定する必要があります。



4. リードバックが不要な場合は、BUSY 信号を未接続にできます。
5. コンフィギュレーションを制御するため、マイクロプロセッサまたは CPLD のような外部コントローラーが必要です。
6. CCLK ネットにはテブナン並列終端が必要です。詳細は、57 ページの「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
7. データ バスは x8 または x16 が可能です。

## パラレル デイジー チェーン

Spartan-6 FPGA のコンフィギュレーションでは、パラレル デイジー チェーンがサポートされます。図 9-4 は、最初のデバイスがマスター BPI コンフィギュレーション モードの例を示しています。最初のデバイスは、マスターまたはスレーブ SelectMAP モードにできます。すべてのデバイス間で D[15:0]、CCLK、RDWR\_B、PROGRAM\_B、DONE、INIT\_B ピンが共有接続で、CSI\_B ピンはデイジー チェーン接続されています。

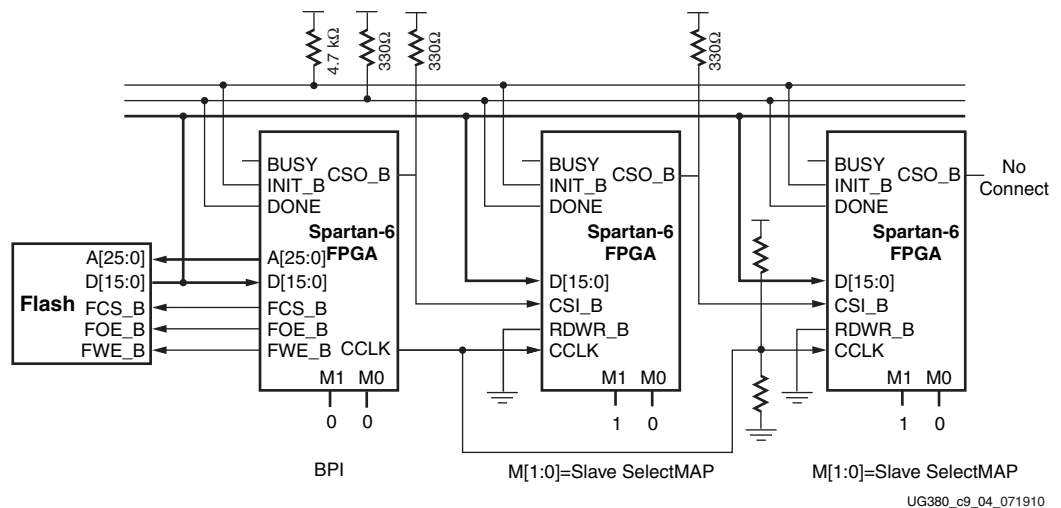


図 9-4 : パラレル デイジー チェーン

図 9-4 について説明します。

1. DONE ピンは、デフォルトでは外部プルアップ抵抗が必要なオープン ドレイン出力です。この接続では、アクティブ DONE ドライバーを必ず無効にしてください。
2. INIT\_B ピンは、双方向オープン ドレイン ピンで、外部プルアップ抵抗が必要です。
3. BitGen のスタートアップ クロック設定では、CCLK を SelectMAP コンフィギュレーションに設定する必要があります。
4. リードバックが不要な場合は、BUSY 信号を未接続にできます。
5. CCLK ネットにはテブナン並列終端が必要です。詳細は、「コンフィギュレーション クロック (CCLK) のボード レイアウト」を参照してください。
6. FCS\_B、FWE\_B、FOE\_B、CSO\_B の弱いプルアップ抵抗は有効にしてください。無効の場合は、各ピンに対して外部にプルアップ抵抗が必要となります。デフォルトでは、コンフィギュレーション後、すべてのデュアル モード I/O に弱いプルダウ抵抗が付きます。

7. チェーンの最初のデバイスは、マスター SelectMAP、スレーブ SelectMAP、または BPI ことができます。
8. リードバックは、現時点ではパラレル デイジー チェーンではサポートされていません。
9. AES 復号化機能は x16 では使用できず、x8 モードでのみサポートされています。

## ギャング SelectMAP

ギャング SelectMAP コンフィギュレーションを使用することによっても、1 つのコンフィギュレーション ビットストリームから複数のデバイスを同時にコンフィギュレーションできます。ギャング SelectMAP コンフィギュレーションの接続では、すべてのデバイスが D ピンに出力されるデータを認識できるようにするため、複数デバイスの CSI\_B ピンが共有接続 (またはグラウンド接続) されます。

外部オシレーターがある場合は、すべてのデバイスをスレーブ SelectMAP モードに設定、あるいは図 9-5 に示すように、1 つのデバイスをマスターとして設定可能です。

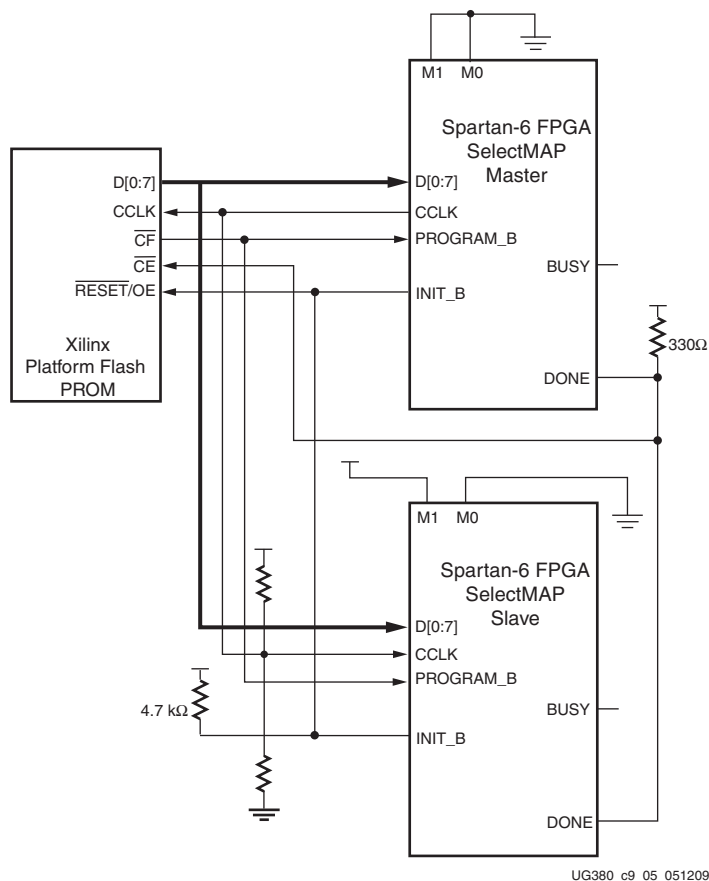


図 9-5 : ギャング x8 SelectMAP コンフィギュレーション

図 9-5 について説明します。

1. DONE ピンは、デフォルトでは外部プルアップ抵抗が必要なオープン ドレイン出力です。この接続では、両デバイスに対してアクティブ DONE ドライバーを必ず無効にしてください。
2. INIT\_B ピンは、双方向オープン ドレイン ピンで、外部プルアップ抵抗が必要です。

3. BitGen のスタートアップ クロック設定では、CCLK を SelectMAP コンフィギュレーションに設定する必要があります。
4. ギャング SelectMAP コンフィギュレーションでは、BUSY 信号は使用しません。
5. この図の PROM は、1 つまたは複数のザイリンクス PROM の場合を示しています。複数のザイリンクス PROM をカスケード接続することで、データ格納範囲全体が拡大できます。
6. BIT ファイルは、PROM に格納前に PROM ファイルに変換する必要があります。詳細は、[79 ページの「PROM ファイルの生成」](#)を参照してください。
7. ザイリンクスの PROM は、パラレル モードに設定する必要があります。このモードは XCFxxP デバイスでのみ使用可能です。
8. SelectMAP モードで、ザイリンクスのコンフィギュレーション PROM から Spartan-6 デバイスをコンフィギュレーションする場合、RDWR\_B および CSI\_B 信号は Low に接続可能です ([36 ページの「SelectMAP データ読み込み」](#)参照)。
9. ギャング SelectMAP コンフィギュレーションは、フラットフォーム フラッシュ XCFxxP PROM 専用です。
10. CCLK ネットにはテブナン並列終端が必要です。詳細は、[57 ページの「コンフィギュレーション クロック \(CCLK\) のボード レイアウト」](#)を参照してください。

1 つのデバイスをマスターにした場合、すべてのデバイスの DONE ピンを接続し、アクティブ DONE ドライバーを無効にする必要があります。これには、共有している DONE 信号にプルアップ抵抗が必要となります。PROM 出力のファンアウトが大きくなるため、シグナル インテグリティに十分注意した設計が必要であり、シグナル インテグリティのシミュレーションの実行を推奨します。

CSI\_B 信号が共有接続されているとき、すべてのデバイスが同時に D 信号を駆動するためリードバックは実行できません。

## SelectMAP の ABORT

ABORT とは、SelectMAP コンフィギュレーションまたはリードバック シーケンスが中断されることです。これは、CSI\_B がアサートされているときに RDWR\_B のステートが変更されると生じます。コンフィギュレーションの ABORT が発生すると、CCLK の次の 4 サイクル間、内部ステータスが D[7:4] ピンに送信されます。その他の D ピンは、常に High です。ABORT シーケンスの終了後、ユーザーはコンフィギュレーション ロジックを再同期化し、コンフィギュレーションを再開できます。バイト間で RDWR\_B をディアサートする必要があるアプリケーションについては、[120 ページの「SelectMAP インターフェイスからのコンフィギュレーション レジスタへのアクセス」](#)を参照してください。

### コンフィギュレーションの ABORT シーケンス

ABORT は、コンフィギュレーション中に次のようにして出されます。

1. コンフィギュレーション シーケンスが正常に開始される。
2. デバイスの CSI\_B が Low にアサートされているとき、RDWR\_B ピンを High に駆動する。
3. CSI\_B がアサートされたまま (Low) のときに BUSY が High になる。RDWR\_B が読み出し制御 (ロジック High) に設定された状態の場合、FPGA がステータス ワードをデータ ピンに送出する。
4. ABORT が 4 クロック サイクル間継続し、ステータスが更新される。

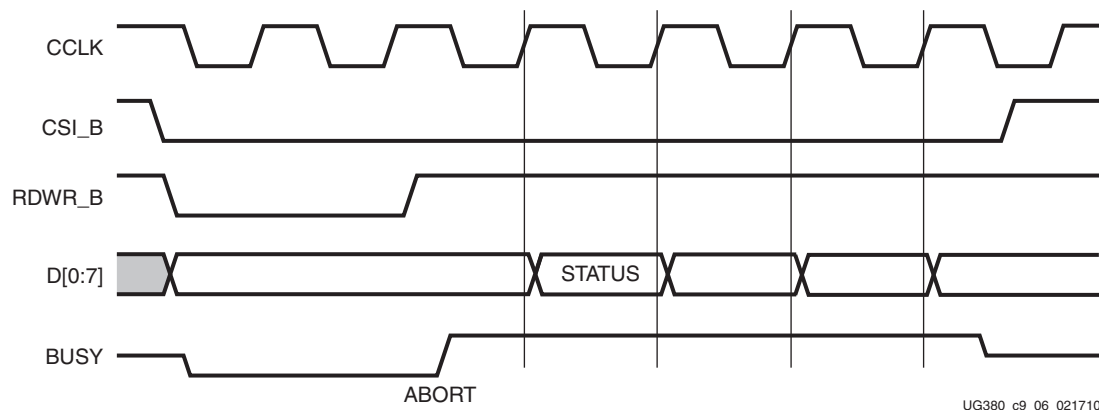


図 9-6 : SelectMAP モードのコンフィギュレーション ABORT シーケンス

## リードバックの ABORT シーケンス

ABORT はリードバック中には次のようにして出されます。

1. リードバック シーケンスが正常に開始される。
2. デバイスの CSI\_B が Low にアサートされているとき、RDWR\_B ピンを Low に駆動する。
3. CSI\_B がアサートされたまま (Low) のときに BUSY が High になる。
4. CSI\_B がディアサートされると ABORT が終了する。

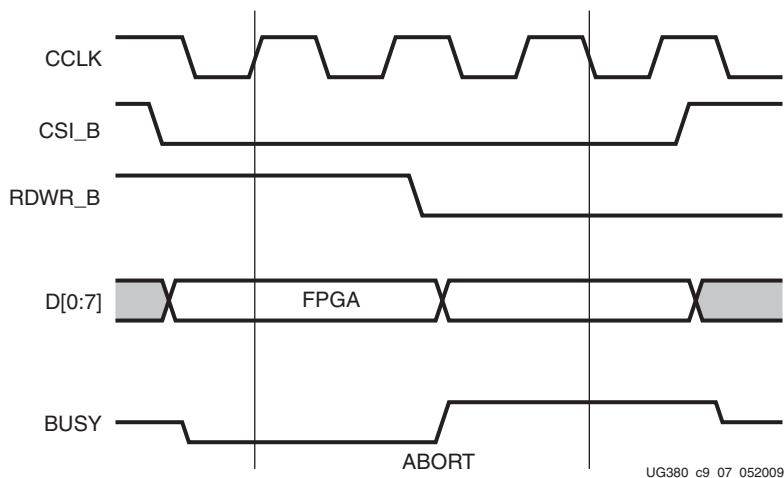


図 9-7 : リードバックの ABORT シーケンス

リードバック中の ABORT では、RDWR\_B 信号が書き込み制御 (FPGA D[x:0] ピンが入力) として設定されているため、ステータス ワードの更新はありません。

## ABORT ステータス ワード

コンフィギュレーション中の ABORT シーケンスでは、デバイスが D[7:0] ピンにステータス ワードを送ります。ステータス ビットはビット スワップしません。また、その他のデータ ピンは常に High です。表 9-2 で、ステータス ワードについて説明します。

表 9-2 : ABORT ステータス ワード

ビット番号	ステータス ビット名	説明
D7	CFGERR_B	コンフィギュレーション エラー (アクティブ Low) 0 = コンフィギュレーション エラー発生 1 = コンフィギュレーション エラーなし
D6	DALIGN	同期ワードの受信 (アクティブ High) 0 = 同期ワードを受信しない 1 = インターフェイス ロジックで 同期ワードを受信
D5	RIP	リードバックを実行中 (アクティブ High) 0 = リードバックを実行しない 1 = リードバックを実行する
D4	IN_ABORT_B	ABORT を実行中 (アクティブ Low) 0 = ABORT を実行する 1 = ABORT を実行しない
D3-D0	1111	1 に固定

ABORT シーケンスは、CCLK の 4 サイクル間継続します。このサイクル間で、データ アライメントおよび ABORT ステータスに反映するよう、ステータス ワードが更新されます。一般的なシーケンスは次のとおりです。

```

11011111 => DALIGN = 1,   IN_ABORT_B = 1
11001111 => DALIGN = 1,   IN_ABORT_B = 0
10001111 => DALIGN = 0,   IN_ABORT_B = 0
10011111 => DALIGN = 0,   IN_ABORT_B = 1

```

最後のサイクルが終了すると、同期ワードの再読み込みが可能となり、データ アライメントが確立します。

## ABORT 後にコンフィギュレーションまたはリードバックを再開

ABORT 完了後にコンフィギュレーションまたはリードバックを再開させるには、次の 2 つの方法があります。

- ABORT 完了後に再びデバイスを同期化する
- PROGRAM\_B を Low に駆動してデバイスをリセットする

デバイスを再同期化するには、CSI\_B をディアサートし、再度アサートしてください。ABORT が発生したときに進行していた最後のコンフィギュレーション パケットまたはリードバック パケットを送信すると、コンフィギュレーション/リードバックが再開します。コンフィギュレーションまたはリードバックを最初から再実行することも可能です。

## SelectMAP リコンフィギュレーション

リコンフィギュレーションとは、DONE ピンが High になった後に FPGA を再プログラミングすることです。通常のコンフィギュレーションと同じ方法で PROGRAM\_B ピンをパルスする、またはデバイスを再同期化してコンフィギュレーション データを送ると、リコンフィギュレーションが開始します。

PROGRAM\_B にパルスを送らずに SelectMAP モードでデバイスをリコンフィギュレーションする場合は、BitGen の **-g Persist** オプションを設定してください。このように設定しなければ、コンフィギュレーションの完了後、DATA ピンがユーザー I/O になります。**-g Persist** オプションがオンのとき、RS[1:0]、CSO\_B、および A[19:16] ピンはユーザー モードで使用できません。リコンフィギュレーションを実行する際は、この BitGen オプションを必ずオンにしてください。デフォルトでは、SelectMAP の 8 つのインターフェイス (D0 ~ D7) は、CONFIG\_MODE 制約を使用して別の SelectMAP 幅が選択されていない限り、保持されます。

# アドバンス JTAG コンフィギュレーション

---

## 概要

Spartan®-6 デバイスでサポートされる IEEE Std 1149.1 JTAG とは「Joint Test Action Group」の略で、この規格を制定した委員会のグループ名です。この規格は、ボード レベルでの個々のコンポーネントの統合性と相互接続を確実にするためのものです。IEEE 1149.1 TAP およびバウンダリ スキャン アーキテクチャは通常 JTAG と呼ばれます。PC ボード層数の増加やボード表面へのマウント技術が一層高密度で高度になるにつれ、バウンダリ スキャン テストは重要なデバッグ用の規格として広く使用されるようになっていきます。

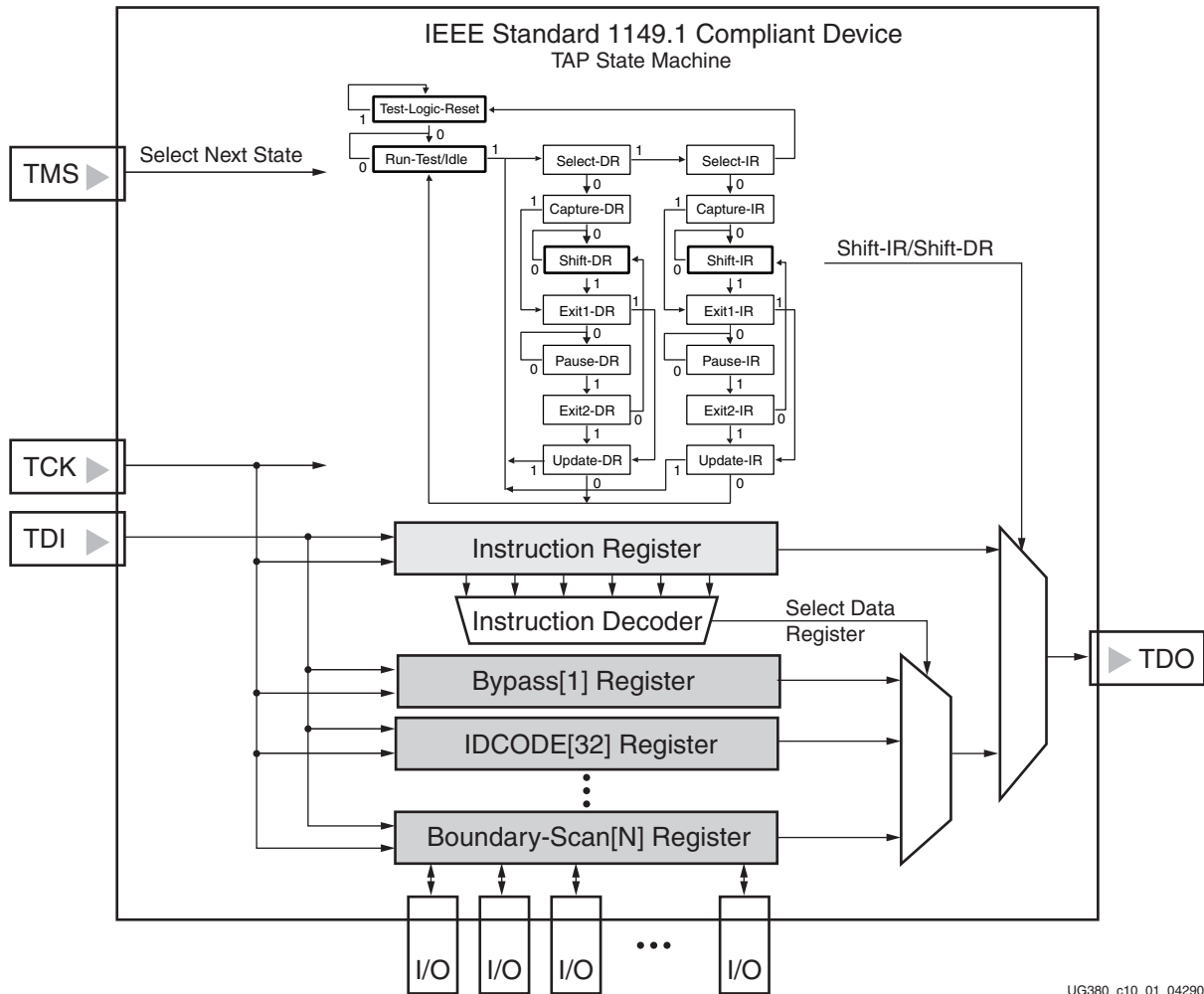
バウンダリ スキャン回路が組み込まれているデバイスでは、I/O ピンにデータを送信し、ボード レベルでデバイス間の接続がテストできます。この回路は、デバイス内部に信号を送ってその動作をテストすることも可能です。これらのテストは、通常、ボードおよびデバイス レベルで断線やショートを検知するために実施されます。

バウンダリ スキャンはテスト用に使用するだけでなく、ユーザー定義の命令をデバイスに組み込んで柔軟性を持たせることもできます。コンフィギュレーションや検証などの特定ベンダー向けの命令の追加により、バウンダリ スキャンのテストおよびその機能の使用率は高まっています。

## JTAG コンフィギュレーション/リードバック

### TAP コントローラーおよびアーキテクチャ

Spartan-6 FPGA の TAP には、IEEE で指定されている 4 つの必須専用ピン (表 3-1 参照) があります。図 10-1 に、標準的な JTAG アーキテクチャを示します。



UG380\_c10\_01\_042909

図 10-1：標準的な JTAG アーキテクチャ

図 10-1 に示すバウンダリ スキャン タップ コントローラーは、16 ステートの有限ステート マシンです。4 つの TAP ピンが、データがどのように各レジスタにスキャンされるかを制御します。また、TCK の立ち上がりエッジにおける TMS ピンの状態によって、ステート移行のシーケンスが決定します。主要なシーケンスは 2 つあり、1 つはデータをデータ レジスタにシフトするシーケンス、もう 1 つは命令を命令レジスタにシフトするシーケンスです。

ステート間の遷移は、TCK の立ち上がりエッジでのみ発生し、ステートはすべて別のステート名になります。7 つのステートで成り立つ 2 つの列は、それぞれ命令パスとデータパスを示しています。データ レジスタのステートには DR、命令レジスタのステートには IR が名前の後に付きますが、これ以外の点ではどちらのステートも同じになります。



各ステートの動作は、次のようになります。

**Test-Logic-Reset :**

このステートではすべてのテスト ロジックが無効になり、これにより通常の IC の動作が有効になります。TAP コントローラーのステート マシンは、コントローラーの初期ステートに関係なく、TMS が High に保持される間に TCK が 5 回パルスすると、Test-Logic-Reset ステートになるように設計されています。このため、Test Reset (TRST) ピンはオプションとなり、ザイリンクス デバイスには含まれません。

**Run-Test-Idle :**

このステートでは、IC のテスト ロジックが一部の命令のある場合にのみアクティブになります。たとえば、命令によりセルフ テストがアクティブになった場合、コントローラーがこのステートになると IC のテスト ロジックが実行されます。それ以外の場合、IC のテスト ロジックはアイドル状態になります。

**Select-DR-Scan :**

このステートでは、データパスになるか、Select-IR-Scan ステートになるかが制御されます。

**Select-IR-Scan :**

このステートでは、命令パスにステートを移動するかどうか制御されます。命令パスにならない場合は、コントローラーは Test-Logic-Reset ステートに戻ります。

**Capture-IR :**

このステートでは、命令レジスタのシフト レジスタ バンクが TCK の立ち上がりエッジで決定したパターンの値を並行してすべて読み込みます。最後 2 つの有効ビットは常に 01 になります。

**Shift-IR :**

このステートでは、命令レジスタが TDI および TDO 間で接続され、読み込まれたパターンが TCK の立ち上がりエッジごとにシフトされます。TDI ピンの命令も命令レジスタにシフトされます。

**Exit1-IR :**

このステートでは、Pause-IR ステートになるか、Update-IR ステートになるかが制御されます。

**Pause-IR :**

このステートでは、命令レジスタのシフトが一時停止できるようになります。

**Exit2-DR :**

このステートでは、Shift-IR ステートになるか、Update-IR ステートになるかが制御されます。

**Update-IR :**

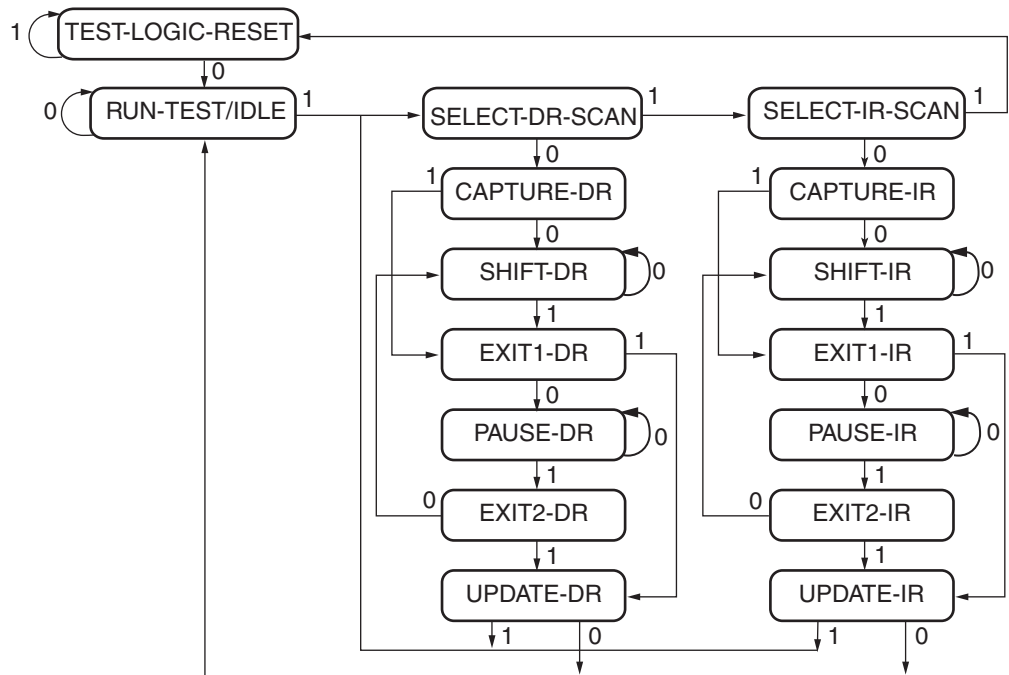
このステートでは、命令レジスタの命令が TCK の立ち下がりエッジごとに 命令レジスタのラッチ バンクに保存されます。保存後、この命令が現在の命令になります。

**Capture-DR :**

このステートでは、データが TCK の立ち上がりエッジで現在の命令によって選択されたデータ レジスタにすべて読み込まれます。

**Shift-Dr, Exit1-DR, Pause-DR, Exit2-DR, Update-DR :**

これらのステートは、命令パスの Shift-IR、Exit1-IR、Pause-IR、Exit2-IR、および Update-IR ステートと類似しています。



Note: The value shown adjacent to each state transition represents the signal present at TMS at the time of a rising edge at TCK.

UG380\_c11\_02\_051109

図 10-2：バウンダリ スキャンの TAP コントローラー

Spartan-6 デバイスは、ザイリンクス ベンダー固有のコマンドおよび必須となる IEEE Std 1149.1 コマンドをサポートしています。これには、EXTEST、INTEST、SAMPLE、BYPASS、IDCODE、USERCODE、および HIGHZ 命令すべてが含まれています。TAP は内部のユーザー定義のレジスタ (USER1、USER2、USER3、USER4)、そしてデバイスのコンフィギュレーション/リードバックもサポートします。

Spartan-6 FPGA のバウンダリ スキャンは、選択したモードとは関係なく動作します。Spartan-6 デバイスのバウンダリ スキャン モードは、ほかのモードの選択を上書きします。したがって、バウンダリ スキャン レジスタを使用するバウンダリ スキャン命令 (SAMPLE/PRELOAD、INTEST、および EXTEST) は、コンフィギュレーション中に実行しないでください。Spartan-6 デバイスのコンフィギュレーション前は、ユーザー定義の命令以外の命令が使用できます。コンフィギュレーション後は、すべての命令が使用可能です。

JSTART および JSHUTDOWN は、Spartan-6 FPGA のアーキテクチャおよびコンフィギュレーション フローでのみ使用される命令です。Spartan-6 デバイスの場合、TAP コントローラーは PROGRAM\_B ピンではリセットされず、コントローラーを TLR ステートにすることによってのみリセットできます。TAP コントローラーは、電源投入時にリセットされます。

EXTEST、INTEST、および BYPASS のような標準バウンダリ スキャン命令に関する詳細は、IEEE 1149.1 規格を参照してください。

## バウンダリ スキャン アーキテクチャ

Spartan-6 デバイスのレジスタには、IEEE 1149.1 規格で定められた必要なレジスタすべてが含まれています。さらに、標準的なレジスタだけでなく、簡易テストや検証用のレジスタも備えています (表 10-1 参照)。

表 10-1 : Spartan-6 FPGA の JTAG レジスタ

レジスタ名	レジスタ長	説明
バウンダリ スキャン レジスタ	各 I/O に 3 ビット	入力、出力、出力イネーブルの制御およびモニタリングを実行します。
命令レジスタ	6 ビット	現在のオペコード命令をホールドし、内部デバイスのステータスをキャプチャします。
BYPASS レジスタ	1 ビット	デバイスをバイパスします。
ID レジスタ	32 ビット	デバイス ID を取得します。
JTAG コンフィギュレーション レジスタ	16 ビット	CFG_IN または CFG_OUT 命令を使用中、コンフィギュレーション バスへのアクセスを可能にします。
USERCODE レジスタ	32 ビット	ユーザーがプログラム可能なコードを取得します。
ユーザー定義のレジスタ (USER1、USER2、USER3、USER4)	デザインにより異なる	デザインによって異なります。

### バウンダリ スキャン レジスタ

テストで使用する主なレジスタはバウンダリ スキャン レジスタです。バウンダリ スキャンの動作は、個々の IOB コンフィギュレーションから独立しています。各 IOB は、ボンディングの接続に関わらず、トライステート制御の双方向性 IOB です。その後、IOB はコンフィギュレーションされて、入力、出力、トライステートのいずれかとなるため、各 IOB に次の 3 つのデータレジスタビットがあります (図 10-1 参照)。

データ レジスタ (DR) の操作時、DR は CAPTURE-DR ステート中に並行してデータを取得します。SHIFT-DR ステート中にデータがシフトアウトされ、新しいデータが取得されます。DR の各ビットに対してアップデート ラッチを使用し、次の SHIFT-DR ステート中に入力データをホールドします。その後、UPDATE-DR ステート中に TCK が Low 駆動すると、データがラッチされます。

TAP コントローラーが UPDATE-DR ステートに移行すると、アップデート ラッチはオープンになります。INTEST または EXTEST を実行する際には、これらのコマンドの実行前に適切なデータがラッチに記憶されていることを確認してください。通常は、SAMPLE 命令を使用して確認できます。

断線およびショートを検知するテスト ベクターの作成時には、内部プルアップおよびプルダウン抵抗を考慮してください。HSWAPEN ピンによって、IOB のプルアップ抵抗の有無が決定されます。図 10-3 に、Spartan-6 FPGA のバウンダリ スキャン アーキテクチャを示します。

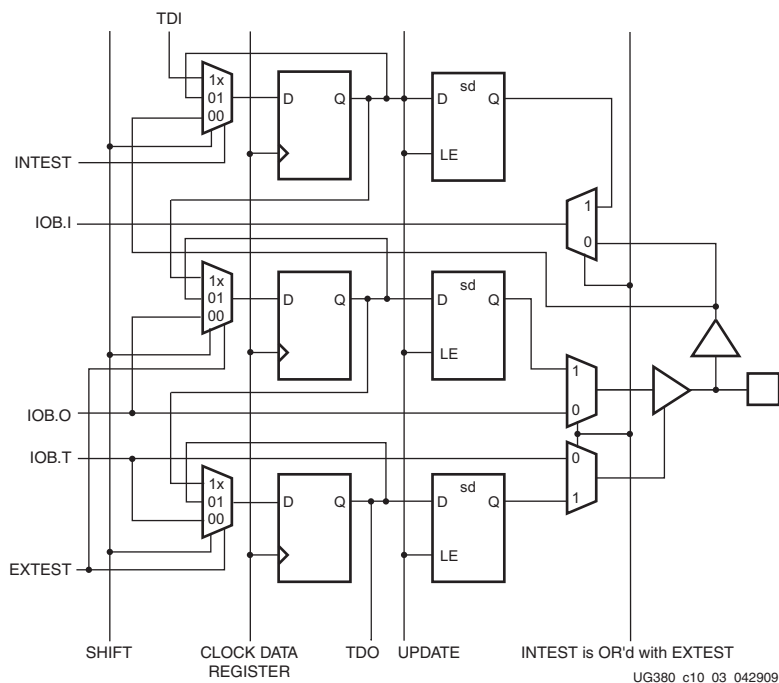


図 10-3 : Spartan-6 FPGA のバウンダリ スキャン ロジック

## ビット シーケンス バウンダリ スキャン レジスタ

次に、TAP を除いた IOB の順序について説明します。順序は最初に入力、次に出力、最後にトライステート IOB 制御です。トライステート IOB 制御は、最も TDO に近い位置に配置されています。入力専用ピンは、入力ビットのみをバウンダリ スキャンの I/O データレジスタに送ります。デバイスのビット シーケンスは、Spartan-6 ファミリー用の BSDL (Boundary-Scan Description Language) ファイルから取得できます。これらのファイルは、ザイリンクスのソフトウェアダウンロード ページから入手可能です。ビット シーケンスはデザインに影響されることはなく、常に同一ビット数で同一順となっています。

## 命令レジスタ

Spartan-6 デバイスの命令レジスタ (IR) は、命令スキャン シーケンス中は TDI と TDO 間に接続されます。命令スキャン シーケンスの準備のため、命令レジスタに固定の命令キャプチャ パターンが並行して読み込まれます。このパターンは、命令が TDI から命令レジスタへシフトするときに、LSB から TDO にシフトします。

操作を実行するには、命令レジスタ (IR) に必要なオペコードを読み込んでおく必要があります。この IR の長さは、デバイス タイプによって異なりますが、Spartan-6 FPGA の IR はすべて 6 ビット幅です。

表 10-2 : Spartan-6 FPGA のバウンダリ スキャン命令

バウンダリ スキャン コマンド	命令	説明
EXTEST	001111	バウンダリ スキャンの EXTEST を有効にします。
SAMPLE	000001	バウンダリ スキャンの SAMPLE を有効にします。

表 10-2 : Spartan-6 FPGA のバウンダリ スキャン命令 (続き)

バウンダリ スキャン コマンド	命令	説明
USER1	000010	ユーザー定義のレジスタ 1 にアクセスします。
USER2	000011	ユーザー定義のレジスタ 2 にアクセスします。
USER3	011010	ファブリックが JTAG プリミティブのインスタンス 3 からの TAP コントローラーへアクセスできるようにするユーザー コードです。
USER4	011011	ファブリックが JTAG プリミティブのインスタンス 3 からの TAP コントローラーへアクセスできるようにするユーザー コードです。
CFG_OUT	000100	リードバックを行なうため、コンフィギュレーション バスにアクセスします。
CFG_IN	000101	コンフィギュレーションを行なうため、コンフィギュレーション バスにアクセスします。
INTEST	000111	バウンダリ スキャンの INTEST を有効にします。
USERCODE	001000	ユーザー コードのシフトアウトを有効にします。
IDCODE	001001	ID コードのシフト アウトを有効にします。
HIGHZ	001010	BYPASS レジスタを有効にし、出力ピンをトライステートにします。
JPROGRAM	001011	PROGRAM_B と同じで、同様の効果があります。
JSTART	001100	スタートアップ クロック ソース (StartupClk:JtagClk) が TCK のとき、スタート アップ シーケンスにクロックを供給します。
JSHUTDOWN	001101	シャットダウン シーケンスにクロックを供給します。
ISC_ENABLE	010000	ISC コンフィギュレーションの開始を示します。完全なシャットダウンが実行されます。
ISC_PROGRAM	010001	インシステム プログラミングを有効にします。
ISC_NOOP	010100	動作なし
ISC_READ	010101	バックアップ バッテリーを使用する RAM のリードバックに使用します。
ISC_DISABLE	010110	ISC コンフィギュレーションが完了します。スタートアップ シーケンスが開始されます。
ISC_DNA (ISC_FUSE_READ)	110000	DNA を読み出します。
BYPASS	111111	BYPASS を有効にします。
RESERVED	その他すべてのコード	ザイリンクスの予約命令 です。

表 10-3 に、命令スキャンシーケンスの一部として IR に読み込まれる命令キャプチャ値を示します。

表 10-3：命令キャプチャ値

TDI →	IR[5]	IR[4]	IR[3]	IR[2]	IR[1:0]	→ TDO
	DONE	INIT(1)	ISC_ENABLED	ISC_DONE	0 1	

## BYPASS レジスタ

もう 1 つの標準データレジスタは、1 つのフリップフロップの **BYPASS** レジスタです。このレジスタは **BYPASS** 命令中に、TDI ピンから TDO ピンにデータをシリアル送信します。また、TAP コントローラーが **CAPTURE-DR** ステートになったとき、0 に初期化されます。

## ID (IDCODE) レジスタ

Spartan-6 デバイスは、IDCODE レジスタと呼ばれる 32 ビットの識別レジスタを備えています。IDCODE は、IEEE 1149.1 規格に基づいたベンダー特有の固定値であり、デバイスの製造者およびタイプを電子的に識別するものです。このレジスタにより、バウンダリ スキャンを使用してテストまたはプログラムされたデバイスの識別が容易になります。IDCODE 命令を使用すると、この識別コードの出力が可能です。

IDCODE の最後のビットは、常に 1 です (JTAG IEEE 1149.1 に基づく)。16 進数の最後の 3 文字は 0x093 となります。Spartan-6 FPGA に割り当てられている IDCODE は、表 5-13 を参照してください。

## JTAG コンフィギュレーション レジスタ

JTAG コンフィギュレーション レジスタは 16 ビット レジスタです。このレジスタにより、コンフィギュレーション バスおよびリードバック操作にアクセスできます。

## USERCODE レジスタ

Spartan-6 ファミリーは、USERCODE 命令をサポートしています。このレジスタにより、特定のデザインに固有な識別コードのユーザー指定が可能となります。USERCODE はデバイスの中にプログラムでき、後で検証のためにリードバックできます。この USERCODE はビットストリームの生成時にビットストリーム内に組み込まれ (BitGen の **-g UserID** オプション)、コンフィギュレーション後に有効になります。デバイスが空、または USERCODE がプログラムされていない場合の USERCODE レジスタ値は、0xFFFFFFFF です。

## USER1、USER2、USER3、USER4 レジスタ

USER1、USER2、USER3、および USER4 レジスタは、コンフィギュレーション後にのみ有効になります。これら 4 つのレジスタは、ユーザーがデザインで定義する必要があります。これらのレジスタへのアクセスは、TAP ピンでの定義後に可能となります。

これらのレジスタの作成には、**BSCAN\_Spartan6** ライブラリ マクロが必要です。このシンボルは、内部スキャン チェーン (USER1、USER2、USER3、USER4) を駆動する際にのみ必要となります。

1 つの一般的な入力 (TDI) ピンおよび複数の共有出力ピンが、TAP コントローラーのステート (RESET、SHIFT、および UPDATE) を示します。Spartan-6 FPGA の TAP ピンは専用ピンであるため、通常のパウンダリ スキャン命令や操作に **BSCAN\_Spartan6** マクロは必要ありません。HDL の場合は、**BSCAN\_Spartan6** マクロをデザインにインスタンシエートしてください。

## Spartan-6 デバイスでのバウンダリ スキャンの使用

一般的に必要とされるタイミングパラメーター (図 10-4) の特性評価データについては、『Spartan-6 FPGA データシート: DC 特性およびスイッチ特性』の「コンフィギュレーション スイッチ特性」の表を参照してください。

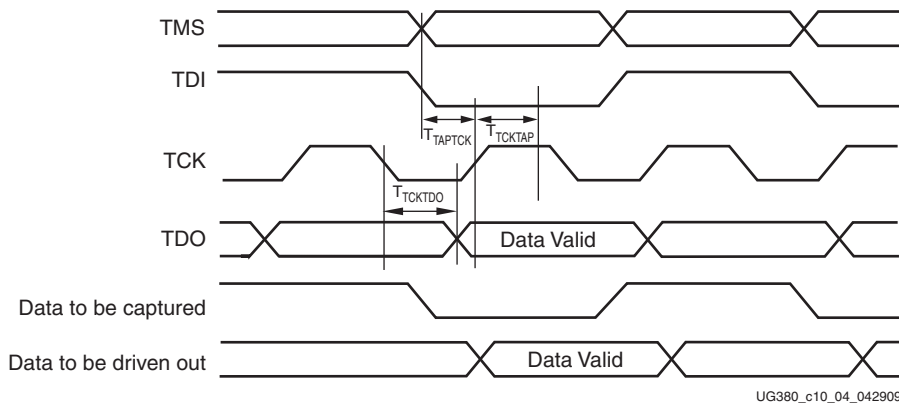


図 10-4 : Spartan-6 FPGA バウンダリ スキャン ポートのタイミング波形

スタートアップシーケンス、ビットストリーム、内部コンフィギュレーションレジスタの詳細は、第 5 章の「コンフィギュレーションの詳細」を参照してください。

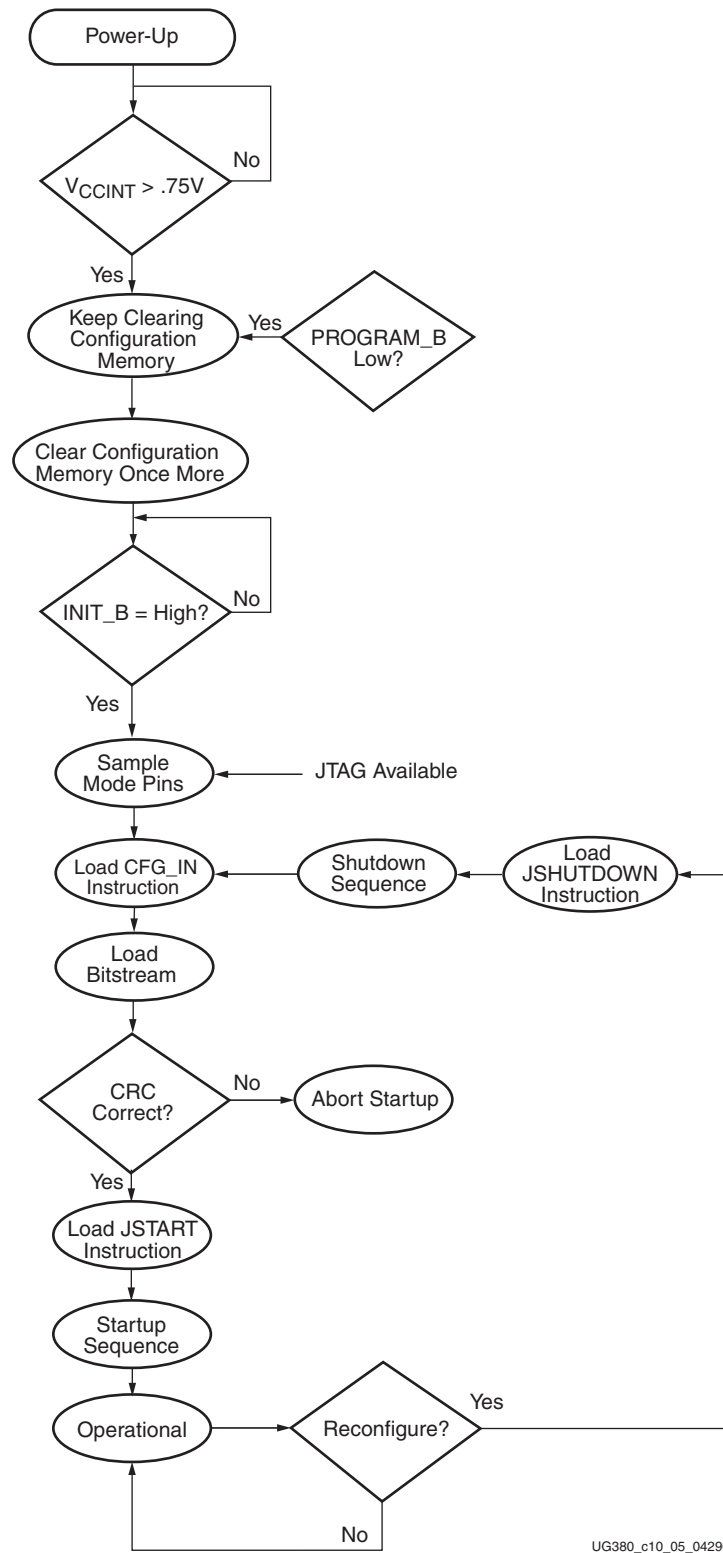
### バウンダリ スキャンによるコンフィギュレーション

頻繁に使用されるベンダー固有のバウンダリ スキャン命令の 1 つに、コンフィギュレーション命令があります。

図 10-5 に、JTAG を使用する場合の Spartan-6 デバイスのコンフィギュレーションフローを示します。後続のセクションでは、バウンダリ スキャンを用いて Spartan-6 デバイスを 1 つのデバイスとしてコンフィギュレーションする方法、あるいは複数のデバイス スキャンチェーンの一部としてコンフィギュレーションする方法について説明します。

コンフィギュレーションされたデバイスをリコンフィギュレーションするには、TAP を切り替え、PROGRAM\_B に信号供給後に CFG\_IN 命令を入力するか、シャットダウンシーケンスを開始してください (図 10-5 参照)。

Spartan-6 FPGA の JTAG コンフィギュレーションアルゴリズムをインプリメントする場合は、アプリケーションノート [XAPP058](#) 『組み込み型マイクロコントローラーを使用するザイリンクスのインシステムプログラミング機能』およびアプリケーションノート [XAPP424](#) 『エンベデッド JTAG ACE Player』で説明している SVF ベースのフローの実行を推奨します。



UG380\_c10\_05\_042909

図 10-5：デバイス コンフィギュレーションのフロー図



## 単一デバイスのコンフィギュレーション

表 10-4 で、Spartan-6 デバイスのコンフィギュレーションに必要な TAP コントローラーのコマンドについて説明します。TAP コントローラーのステートは、図 10-2 を参照してください。iMPACT を使用してコンフィギュレーションする場合、TAP コントローラーのコマンドは自動的に実行されます。

表 10-4 : 単一デバイスのコンフィギュレーション シーケンス

TAP コントローラーの状態		セットアップおよびホールド		クロック数
		TDI	TMS	TCK
1.	電源投入時に、TMS をロジック 1 にして TCK クロックを 5 回入力します。これにより、TLR (Test-Logic-Reset) ステートから開始されます。	X	1	5
2.	RTI ステートに移行します。	X	0	1
3.	SELECT-IR ステートに移行します。	X	1	2
4.	SHIFT-IR ステートに移行します。	X	0	2
5.	LSB から CFG_IN 命令を読み込みます。	000101	0	5
6.	IEEE 規格で規定されているように、SHIFT-IR の終了時に CFG_IN の最後のビット MSB を読み込みます。	0	1	1
7.	SELECT-DR ステートに移行します。	X	1	2
8.	SHIFT-DR ステートに移行します。	X	0	2
9.	Spartan-6 FPGA のビットストリームをシフト インします。ビット <sub>n</sub> (MSB) はビットストリームの最初のビットです。(1)	bit <sub>1</sub> ...bit <sub>n</sub>	0	(ビットストリーム中のビット) - 1
10.	ビットストリームの最後のビットをシフト インします。ビット <sub>0</sub> (LSB) は、EXIT1-DR ステートへの移行時にシフトします。	bit <sub>0</sub>	1	1
11.	UPDATE-DR ステートに移行します。	X	1	1
12.	RTI ステートに移行します。	X	1	1
13.	SELECT-IR ステートに移行します。	X	1	2
14.	SHIFT-IR ステートに移行します。	X	0	2
15.	JSTART 命令の読み込みを開始します。JSTART 命令によってスタートアップシーケンスが初期化されます。	001100	0	5
16.	JSTART 命令の最後のビットを読み込みます。	0	1	1
17.	UPDATE-IR ステートに移行します。	X	1	1
18.	RTI ステートに移行し、TCK クロックを最低 16 回入力してスタートアップシーケンスを開始させます。	X	0	16
19.	TLR ステートに移行します。デバイスが動作を開始します。	X	1	3

## 注記 :

1. コンフィギュレーションレジスタの場合、右 (TDI) から左 (TDO) に MSB からデータがシフトします (コンフィギュレーションレジスタへのシフトは、その他のレジスタへのシフトとは異なり、MSB 順となります。)

### 複数デバイスのコンフィギュレーション

複数の Spartan-6 デバイスをチェーン接続してコンフィギュレーションすることもできます (図 10-6 参照)。JTAG チェーン内の複数のデバイスは 1 つずつコンフィギュレーションされます。複数デバイスのコンフィギュレーション手順は、すべてのチェーン サイズで同様です。

TAP コントローラーの手順は、図 10-1 のステート図を参照してください。

1. 電源投入時に、TMS をロジック 1 にして TCK を 5 回クロック入力します。これにより、TLR (Test-Logic-Reset) ステートから確実に開始されます。
2. CFG\_IN 命令をターゲット デバイスに読み込みます (その他すべてのデバイスは BYPASS 命令)。RTI ステートに移行します (RUN-TEST/IDLE)。
3. 表 10-4 の 手順 7 から 手順 11 のように、コンフィギュレーション ビットストリームを読み込みます。
4. 各デバイスに対して、手順 2 および 手順 3 を繰り返します。
5. JSTART コマンドをすべてのデバイスに読み込みます。
6. RTI ステートに移行し、TCK を 16 回クロック入力します。

この段階ですべてのデバイスはアクティブです。

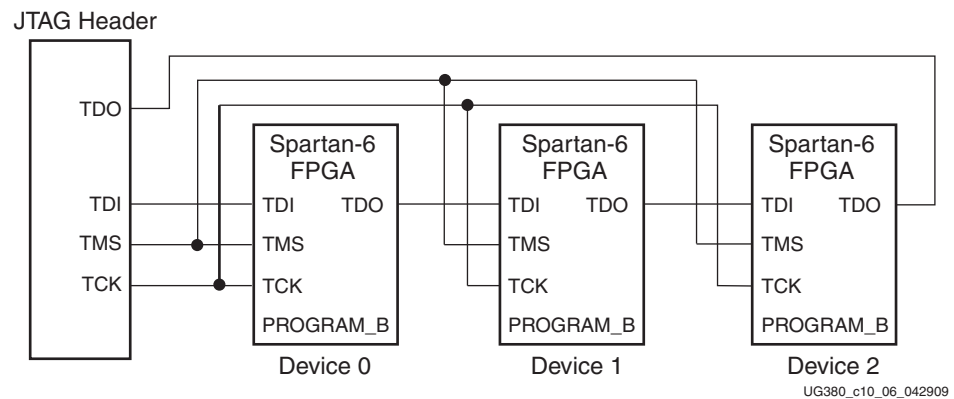


図 10-6 : デバイスのバウンダリ スキャン チェーン

### スタートアップおよびシャットダウン シーケンス (JTAG)

スタートアップおよびシャットダウン シーケンスのクロック ソースは、CCLK、UserCLK、および JTAGCLK です。クロックは BitGen で選択します。スタートアップシーケンスは、ISC\_Accessed ステートのときに実行されます。JTAGCLK をクロック ソースとする場合、ISC\_DISABLE が現在の JTAG 命令である間は、スタートアップ シーケンスは TAP Run/Test Idle ステートで JTAGCLK を受信します。ISC\_DISABLE を正常に完了するために Run/Test Idle ステートで必要となるクロック サイクル数は、スタートアップ シーケンスの完了に要するクロック サイクル数によって決定します。

UserCLK または CCLK をスタートアップ シーケンスのクロック ソースとする場合は、スタートアップ シーケンスを正常に完了するために、Run/Test Idle で必要となるクロック サイクル数を把握しておく必要があります。

シャットダウン シーケンスは、デバイスが動作ステートから ISC\_Accessed ステートに移行したときに実行されます。ISC\_ENABLE 命令の実行時に、シャットダウンが実行されます。JTAGCLK をシャットダウン シーケンスのクロック ソースとする場合、クロックは ISC\_ENABLE 命令の

Run/Test Idle ステートで供給されます。Run/Test Idle で必要なクロック サイクル数は、シャットダウン シーケンスの完了に要するクロック サイクル数によって決定します。

CCLK または UserCLK をシャットダウン シーケンスのクロック ソースとする場合、このシーケンスを完了するために、Run/Test Idle で必要となる JTAGCLK サイクル数を把握しておく必要があります。シャットダウン シーケンスとスタートアップシーケンスでは、順序が逆になっています。

**注記：**デバイスを JTAG でコンフィギュレーションする場合、BitGen の設定にかかわらず、常に TCK をスタートアップ クロックおよびシャットダウン クロックとしてください。

