

PlanAhead ユーザー ガイド

UG632 (v14.1) 2012 年 4 月 24 日



Notice of Disclaimer

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

© Copyright 2012 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. All other trademarks are the property of their respective owners.

本資料は英語版 (vx.x) 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 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	改訂内容
2011 年 3 月 1 日	13.1	<ul style="list-style-type: none">・ タイミング シミュレーションおよびビヘイビア シミュレーション用に ISim を統合・ プッシュ ボタンでフローを実行可能な GUI・ Project Navigator ファイルからプロジェクトを作成する機能・ [Clock Resources] ビューを追加・ メイン メニュー検索機能を追加・ ピン割り当てとバンク規則・ 並べ替え機能・ IBIS モデルのエクスポート機能・ エラー、重要な警告、警告、および情報メッセージを統合したメッセージ管理機能・ サードパーティのテキスト エディターを追加するなどのテキスト エディターのカスタマイズ機能・ プロジェクトのアーカイブ機能・ ChipScope でのデバッグ用にタグ ネットを追加する機能・ [Device] ビューの凡例を追加する機能 (回路図凡例)

日付	バージョン	改訂内容
2011 年 7 月 6 日	13.2	<ul style="list-style-type: none"> クロックの相互関係レポート。詳細は、第 7 章の「クロックの相互関係の解析」を参照してください。 インプリメント済みデザインのタイミング解析で TRCE を起動。詳細は、第 11 章の「タイミング結果の解析」を参照してください。 エンベデッド プロセッサ デザインを PlanAhead プロジェクトにインポート。詳細は、第 3 章の「エンベデッド プロセッサ ソースの管理」を参照してください。 RTL エラボレーションおよび合成でのグローバル インクルード ファイル。詳細は、第 6 章の「ファイルのコンパイル順」を参照してください。 [Messages] ビューの重要な警告およびエラーを [Critical Warning] ダイアログ ボックスに表示。詳細は、第 4 章「表示環境の使用」の「[Messages] ビュー」を参照してください。 [Messages] ビューでメッセージを折り返して表示。詳細は、第 4 章「表示環境の使用」の「[Messages] ビュー」を参照してください。 RTL デザインを開いたときにデフォルトで [Schematic] ビューを表示。詳細は、第 5 章の「エラボレート済みデザインのリソース使用」を参照してください。 一部の Virtex®-7 および Kintex™-7 デバイスに暫定的な SSN サポートを追加。詳細は、第 8 章の「ノイズ解析予測の使用」を参照してください。 [Device] および [Package] ビューで 7 シリーズの HP/HR バンクをサポート。詳細は、第 4 章「表示環境の使用」の「[Device] ビュー」および 「[Package] ビュー」を参照してください。 合成およびインプリメンテーションの両方にパーティションを指定。詳細は、第 13 章の「パーティションの使用」を参照してください。 コンフィギュレーション モードを設定した後に、[Package Pins] ビューを自動的に並べ替え可能。詳細は、第 8 章の「デバイス コンフィギュレーション モードの設定」を参照してください。 「固定されていない」と「配置されていない」の用語の使用方法を修正。詳細は、第 10 章の「LOC 制約および BEL 制約の設定」を参照してください。 ボードの信号およびボードの電圧を CSV にエクスポート。詳細は、付録 A の「CSV ファイル形式の I/O ポート リスト」を参照してください。 [Net Properties] ビューの [Connectivity] タブと [Pins] タブの説明を追加。詳細は、第 4 章「表示環境の使用」の「[Schematic] ビュー」を参照してください。 ツール コマンド言語 (Tcl) エラーの \$errorInfo 変数について記述。詳細は、第 14 章の「エラー、警告、クリティカル警告、および情報メッセージ」を参照してください。 [File] → [Export] → [Export IBIS Model] コマンドを削除

日付	バージョン	改訂内容
2011 年 10 月 19 日	13.3	<ul style="list-style-type: none"> • ネットリスト プロジェクトのファイル ベースのデザイン入力方法を記述。詳細は、第 3 章「プロジェクトの操作」の「合成後プロジェクトの作成」を参照してください。 • 新規 RTL ソース ファイルを作成する場合のモジュールの定義方法を追加。詳細は、第 3 章「プロジェクトの操作」の「新規モジュールの定義」を参照してください。 • 合成およびインプリメンテーションのプロジェクト パーツについて記述。詳細は、第 3 章「プロジェクトの操作」の「プロジェクト設定」を参照してください。 • プロジェクトのデフォルト言語について記述。詳細は、第 3 章「プロジェクトの操作」の「プロジェクト設定」を参照してください。 • New Project ウィザードのプロジェクト サブディレクトリのチェック ボックスを説明。詳細は、第 3 章「プロジェクトの操作」の「新規プロジェクトの作成」を参照してください。 • Tcl コンソールのコマンドの自動判別機能。詳細は、第 4 章「表示環境の使用」の「Tcl コンソールおよびメッセージエリアの使用」を参照してください。 • [Properties] ビューと [Run Properties] ビューの説明をアップデート。詳細は、第 4 章「表示環境の使用」の「[Properties] ビュー」を参照してください。 • [Sources] ビューの [Hierarchies]、[Libraries]、[Compile Order] タブについて記述。詳細は、第 4 章「表示環境の使用」の「汎用のビューの使用」を参照してください。 • オブジェクトの選択方法をアップデート。詳細は、第 4 章「表示環境の使用」の「オブジェクトの選択、マーク、移動」を参照してください。 • カスタム Tcl コマンドを GUI に含める方法を説明。詳細は、第 4 章「表示環境の使用」の「カスタム メニュー コマンドの追加」を参照してください。 • デザイン間の選択機能についての説明を追加。詳細は、第 4 章「表示環境の使用」を参照してください。 • ファイルの検索置換の機能について説明。詳細は、第 4 章「表示環境の使用」の「[Find] コマンド」を参照してください。 • 特定の合成またはインプリメンテーション run でターゲット パーツを変更する方法について記述。詳細は、第 4 章「表示環境の使用」の「[Run Properties] ビューの使用」を参照してください。 • [World] ビューのサイズおよび位置の変更方法について説明。詳細は、第 4 章「表示環境の使用」の「ワークスペースのビュー」を参照してください。 • Artix™-7 デバイスの SSN サポートを追加。詳細は、第 8 章の「ノイズ解析予測の使用」を参照してください。 • [Create Multiple Runs] コマンドを [Create New Runs] コマンドに変更。詳細は、第 9 章「デザインのインプリメンテーション」の「インプリメンテーション run の定義」を参照してください。 • IOBUSSLRC および IOPCBT デザイン ルール チェックについて記述。詳細は、付録 B「IOB の DRC」を参照してください。
2012 年 1 月 18 日	13.4	<ul style="list-style-type: none"> • 第 8 章の「IBIS モデルのエクスポート」を記述。 • planAhead.ini ファイルへのパスをアップデート。詳細は、付録 A の「デフォルト環境の出力」を参照してください。 • シミュレーションの NetGen 実行に関する情報を追加。詳細は、第 11 章の「タイミングシミュレーションの実行」を参照してください。 • [Find] コマンドの検索条件のグループ化情報を追加。詳細は、第 4 章の「[Find] コマンド」を参照してください。

日付	バージョン	改訂内容
2012 年 4 月 24 日	14.1	<ul style="list-style-type: none"> • Flow Navigator メニューを 第 2 章の「Flow Navigator の理解」に示すようにアップデート。 • 第 2 章の「デザインの操作」にエラボレート済みデザイン、合成済みデザイン、インプリメント デザインに関する詳細を追加。 • 第 2 章の「エラボレート済みデザインを開く」にバックグラウンド プロセスを追加。 • 第 2 章の「開いているデザインの管理」に [Open in New Design] コマンドを追加。 • 第 2 章の「デザインのアップデートおよび再読み込み」を追加。 • 第 3 章の「新規プロジェクトの作成」の New Project ウィザードをアップデート。 • XST および Synplify® プロジェクトを 第 3 章の「外部で作成したプロジェクト」に移動。 • 第 3 章の「デフォルト パーツまたはボードの選択」の [Default Part] ダイアログ ボックスに TDP ボード セクションが含まれるようにアップデート。 • 第 3 章の「デフォルト パーツまたはボードの選択」に Zynq™ パーツ用の読み出し専用 PSSIO ピンに関する記述を追加。 • 第 3 章の「IP コアの管理」、第 3 章の「エンベデッド プロセッサ ソースの管理」をアップデート。 • 第 3 章の「DSP ソースの管理」を追加。 • 第 3 章の「[Project Settings] ダイアログ ボックスの [General] ページ」にターゲット シミュレーター ModelSim/QuestaSim を追加。 • 第 4 章の「[Sources] ビュー」に [IP Sources] タブを追加。 • 第 4 章の「[Device] ビュー」に [Device View Options] コマンドに関する説明を追加。 • 第 4 章の「[Package] ビュー」、第 4 章の「[Schematic] ビュー」、第 4 章の「[I/O Ports] ビュー」、をアップデート。 • 第 4 章の「PlanAhead の設定」をアップデート。 • 第 5 章「エラボレート済み RTL デザイン」で RTL デザインをエラボレート済みデザインに変更。 • 第 5 章の「ビヘイビア シミュレーションの実行」に ModelSim/QuestaSim® を追加。 • 第 6 章「デザインの合成」でネットリスト デザインを合成済みデザインに変更。 • 第 6 章の「合成の実行」をアップデート。 • 第 8 章の「I/O ポートのインポート」に推論済み差動ペアに関する説明を追加。 • 第 8 章の「I/O ポートの配置」に読み出し専用の Zynq PSSIO ピンに関する説明を追加。 • 第 8 章の「RTL デザインへの変換」を記述。 • 第 8 章の「SSN 解析の実行」に HTML 出力フォーマットを追加。 • 第 9 章の「インプリメンテーションの実行」をアップデート。 • 第 9 章の「インクリメンタル インプリメンテーションの実行」を記述。 • ピン配置ビュー レイアウトでの SSN レポートの改善について記述。 • 第 10 章の「Pblock の移動とサイズ変更」をアップデート。 • 第 11 章の「タイミング シミュレーションの実行」に ModelSim/QuestaSim を追加。 • 第 11 章の「XPower Analyzer を使用した消費電力分配解析」に [Launch XPA] ダイアログ ボックスについて記述。 • 第 14 章「Tcl およびバッチ スクリプト」に Linux 用の .editrc ファイルについて記述。 • 第 14 章「Tcl およびバッチ スクリプト」に Tcl 変数として引き指数を渡すための Tcl eval の使用方法を追加。 • 付録 A の「CSV での差動ペアの定義」を記述。 • 付録 E 「その他のリソース」に System Generator のリファレンスを追加。

目次

改訂履歴.....	2
第 1 章 : PlanAhead ツールの概要	
PlanAhead ツールの概要.....	11
PlanAhead ツールの使用.....	12
PlanAhead ツールの起動.....	15
第 2 章 : PlanAhead ツール フロー	
デザイン フロー.....	20
ユーザー モデル.....	23
Flow Navigator の理解	24
デザインの操作	30
第 3 章 : プロジェクトの操作	
プロジェクト タイプ.....	43
新規プロジェクトの作成.....	45
プロジェクトの管理	59
プロジェクト ソースの管理	61
制約の管理.....	62
デザイン ソース ファイルの管理	67
シミュレーション ソースの管理.....	73
IP コアの管理	75
DSP ソースの管理.....	84
エンベデッド プロセッサ ソースの管理	87
[Project Summary] ビュー.....	97
プロジェクト設定	101
第 4 章 : 表示環境の使用	
表示環境.....	111
メイン表示エリア	114
Tcl コンソールおよびメッセージ エリアの使用.....	117
ビューの操作.....	122
オブジェクトの選択、マーク、移動.....	131
[Find] コマンド	137
汎用のビューの使用	144
テキスト エディターの使用	184
PlanAhead の設定	188
第 5 章 : エラボレート済み RTL デザイン	
デザイン ソース ファイルの管理	201

RTL ソース ファイルの編集.....	201
RTL ソース ファイルのエラボレーションと解析	202
消費電力予測.....	208
ビヘイビアー シミュレーションの実行	214
RTL DRC の実行	220

第 6 章：デザインの合成

合成手法.....	223
合成の実行.....	226
合成実行の監視	232
合成終了後のフロー	233
合成結果の解析	234

第 7 章：合成済みデザインの制約および解析

合成済みデザイン環境の使用	235
リソース統計の表示とレポート	237
ロジックの解析	243
ChipScope デバッグ コアの挿入.....	244
タイミング制約の定義.....	245
タイミング解析の実行.....	248
スラック ヒストグラムの使用.....	261
クロックの相互関係の解析.....	270
物理制約の定義	275
デザイン ルール チェック (DRC) の実行.....	278

第 8 章：I/O ピン配置

I/O ピン配置の手法.....	283
I/O Planning ビュー レイアウトの使用.....	285
デバイス リソースの表示	286
互換性のある代替デバイスの指定.....	290
デバイス コンフィギュレーション モードの設定.....	291
I/O ポートの定義と設定	292
インタラクティブなデザイン ルール チェックの使用	301
I/O ポートの配置.....	302
クロック ロジックの配置	307
I/O とクロック ロジックの配置の検証.....	311
RTL デザインへの変換.....	314
I/O ピンとパッケージ データのエクスポート	314
IBIS モデルのエクスポート	315
ノイズ解析予測の使用.....	317

第 9 章：デザインのインプリメンテーション

インプリメンテーションの実行	325
インプリメンテーション run の開始.....	333

プロジェクト ステータスの確認	334
インプリメンテーション run 結果の確認	335
インプリメンテーション後のフロー	339
リモート Linux ホストでの run の起動	339
第 10 章：デザインのフロアプラン	
Pblock の概要	343
Pblock の設定	353
LOC 制約および BEL 制約の設定	365
PlanAhead からのデータの抽出	374
第 11 章：インプリメンテーション結果の解析	
インプリメント済みデザインを開く	377
タイミング結果の解析	380
ロジック接続の表示	388
今後のインプリメンテーション run で使用する配置のロック	392
デザイン メトリックの表示	393
タイミング シミュレーションの実行	397
XPower Analyzer を使用した消費電力分配解析	404
FPGA Editor の起動	406
第 12 章：デザインのプログラムとデバッグ	
ビットストリーム ファイルの生成	409
ChipScope を使用したデザインのデバッグ	410
ChipScope Pro Analyzer の起動	420
iMPACT の起動	420
第 13 章：階層デザイン手法の使用	
パーティションの使用	421
パーティションのプロモート	425
パーティションのインポート	427
ソースのアップデート	428
関連する手法	428
第 14 章：Tcl およびバッチ スクリプト	
Tcl ジャーナル ファイル	431
Tcl ヘルプ	431
[Tcl Console] ビュー	432
PlanAhead ソフトウェアの起動	433
一般的な Tcl 構文のガイドライン	434
ファースト クラスの Tcl オブジェクトとその関係	437
エラー、警告、クリティカル警告、および情報メッセージ	441
Tcl リファレンス	442

第 15 章 : Project Navigator からの PlanAhead の使用

Project Navigator 内の PlanAhead のプロセス.....	443
---	-----

付録 A : PlanAhead の入力ファイルおよび出力ファイル

入力ファイル.....	449
CSV ファイル形式の I/O ポート リスト.....	450
レポートの出力.....	453
デフォルト環境の出力.....	454
プロジェクト データの出力.....	456
ISE インプリメンテーションの出力ファイル.....	458

付録 B : PlanAhead の DRC

RTL DRC :消費電力およびパフォーマンス.....	461
フロアプラン DRC.....	463
I/O ポート/クロック ロジック/配置 DRC の詳細.....	466

付録 C : XilinxNotify を使用したリリースのインストール

PlanAhead のリリース ストラテジ.....	471
XilinxNotify の実行.....	471
XilinxNotify のネットワーク インストール.....	471

付録 D : パスワード入力なしの SSH の設定

SSH の設定.....	473
--------------	-----

付録 E : その他のリソース

ザイリンクス リソース.....	475
ハードウェア資料.....	475
ChipScope 資料.....	476
EDK 資料.....	476
ISE 資料.....	477
System Generator for DSP 資料.....	478
パーシャル リコンフィギュレーション資料.....	478
アプリケーション ノート.....	478
PlanAhead 資料.....	479
Zynq 資料.....	479
IP 資料.....	480

PlanAhead ツールの概要

ザイリンクス PlanAhead™ ツールは、FPGA デバイス デザイン全体およびインプリメンテーション サイクルを決定するデザイン ツールです。

ザイリンクス ツールのインストール方法および新機能については、付録 E「その他のリソース」に示す次の資料を参照してください。

- 『ザイリンクス デザイン ツール：インストールおよびライセンス ガイド』(UG798)
- 『ザイリンクス デザイン ツール：リリース ノート ガイド』(UG631)
- PlanAhead の既知の問題 (アンサー 40512)

PlanAhead ツールの概要

PlanAhead は、FPGA デバイスの設計プロセスおよびインプリメンテーション プロセス用のデザイン 解析ツールで、次のツールと統合されています。

- ザイリンクス ISE® Design Suite 合成およびインプリメンテーション ツール
- System Generator for DSP
- ザイリンクス エンベデッド開発キット (EDK)
- XST (Xilinx Synthesis Technology) ツール
- CORE Generator™ ツール
- ChipScope™ Pro デバッグ ツール
- ISim ツール
- XPower Analyzer ツール
- FPGA Editor ツール
- iMPACT デバイス プログラム ツール

PlanAhead を使用すると、デザインの RTL (レジスタ トランスファー レベル) ソースを定義し、合成済みネットリスト、インプリメンテーション結果を解析することにより、回路のパフォーマンスを向上できます。異なるインプリメンテーション オプションを試したり、タイミング制約を調整したり、フロアプラン手法を使用して物理制約を適用したりすることができます。リソース数、インターコネクト遅延、消費電力、配線接続を初期段階で予測することで、適切なロジック設計、デバイスの選択、フロアプランが容易になります。

PlanAhead には階層データ モデルが含まれており、「デザイン保持」と呼ばれるインクリメンタル デザイン機能を使用できます。デザインを分割して変更されていないモジュールをそのまま保持できるので一貫した結果が得られ、場合によってはランタイムが短縮されます。

また、追加ライセンスを適用することにより、パーシャル リコンフィギュレーション デザイン アプリケーションも使用できます。これらのアドバンス デザイン手法の概要は、[第 13 章「階層デザイン手法の使用」](#)を参照してください。

PlanAhead は、スタンドアロン アプリケーションとして、または ISE ソフトウェアから起動できます。Project Navigator から起動する場合は、特定の機能のみが使用可能で、「ISE 統合モード」と呼ばれます。Project Navigator との統合についての詳細は、[第 15 章「Project Navigator からの PlanAhead の使用」](#)を参照してください。

PlanAhead ツールの使用

PlanAhead を使用すると、次が実行できます

- レジスタ トランスファー レベル (RTL) の開発からビットストリームの生成までのデザイン データ フローを、ボタンをクリックするだけで実行できます。
- エラボレート済みデザインを使用して、RTL の設計および解析を実行します。
- 統合された ISim ツールを使用してビヘイビア シミュレーションおよびタイミング シミュレーションを実行します。
- IBIS (I/O Buffer Information Specification) モデルをエクスポートします。
- 統合された CORE Generator ツールを使用して、IP をカスタマイズおよびインプリメントします。
- 複数の合成およびインプリメンテーション run を設定し、実行します。
- I/O ピンを配置します。
- 制約を管理し、フロアプランを実行します。
- リソース使用率、タイミングおよび消費電力の概算を出します。
- デザイン ルール チェック (DRC) を実行します。
- ChipScope デバッグ ツールを使用して、デバッグ コアを挿入およびインプリメンテーションします。
- iMPACT ツールを使用してデバイス コンフィギュレーションおよびファイル生成を実行します。
- インプリメンテーション結果を解析します。
- プログラムおよびデザイン検証ツールを起動します。
- 次のサードパーティ ツールと連動します。
 - Synopsys® 社の Synplify Pro® FPGA 合成ツール
 - Mentor Graphics® 社の ModelSim または Questa® Advanced Simulator ツール
 - Mentor Graphics 社の Precision® 合成

プロジェクトの作成と管理

PlanAhead には、FPGA デザイン プロジェクトを作成および管理し、制約セットを作成するさまざまなオプションがあり、デザインを異なるパターンで試すことができます。詳細は、[第 3 章「プロジェクトの操作」](#)を参照してください。

RTL および IP デザイン

PlanAhead のエラボレート済みデザイン環境では、RTL ソース ファイルを作成および管理できます。統合された CORE Generator を使用して、IP をカスタマイズおよびインプリメントできます。

デザイン ソース ファイルの作成および管理機能に加え、RTL ロジックの確認、RTL 回路図表示、RTL DRC、RTL ベースのリソースの予測、消費電力の予測を実行できます。詳細は、[第 5 章「エラボレート済み RTL デザイン」](#)を参照してください。

デザイン シミュレーション

PlanAhead からザイリンクス ISim ツールを起動し、デザインのビヘイビア シミュレーションおよびインプリメント済みデザインのタイミング シミュレーションを実行できます。

必要なシミュレーション実行ファイルが準備されて ISim のグラフィカル ユーザー インターフェイス (GUI) が起動し、デザインのシミュレーション、波形ビューアーでの信号の追加および表示、デザインの確認およびデバッグを必要に応じて実行できるようになります。詳細は、次を参照してください。

- ビヘイビア シミュレーションに関する情報: [第 5 章「エラボレート済み RTL デザイン」](#)
- タイミング シミュレーションに関する情報: [第 11 章「インプリメンテーション結果の解析」](#)

合成とインプリメンテーション

PlanAhead ツールには、さまざまな物理制約またはタイミング制約を使用して複数の合成およびインプリメンテーション run をサポートする合成およびインプリメンテーション環境が含まれます。さまざまなツール オプションを使用することで、配線を完了して、タイミング クロージャール達成できます。複数の合成およびインプリメンテーション run は、ザイリンクス ISE 合成ツールおよびインプリメンテーション ツールを使用して、順次を実行します。また、マルチ プロセッサ マシンでは同時に実行できます。

- 合成: [第 6 章「デザインの合成」](#)を参照
- インプリメンテーション: [第 9 章「デザインのインプリメンテーション」](#)を参照

デザイン解析および制約の定義

PlanAhead では、デザイン プロセスの各段階でデザインを解析できます。リソース、タイミング、消費電力を予測し、DRC を実行することにより、さまざまなデバイス、制約、合成およびインプリメンテーション オプションを使用してデザインを実行し、最適な結果を達成できます。

インプリメンテーション後は、FPGA Editor および XPower Analyzer ツールを PlanAhead から直接起動して、インプリメント済みFPGAを詳細に確認することができます。

これらの機能は、合成済みデザインまたはインプリメント済みデザインを解析することにより、インプリメンテーションの前後で使用できます。詳細は、次を参照してください。

- [第 7 章「合成済みデザインの制約および解析」](#)
- [第 11 章「インプリメンテーション結果の解析」](#)

ピン配置

PlanAhead ツールには、デバイスの I/O 要件を定義して解析する I/O Planning ビューレイアウトが含まれ、I/O ポートの作成と物理パッケージ ピンへの割り当てができます。これにより、PCB (Printed Circuit Board) デザインと FPGA (Field Programmable Gate Array) デザイン両方の要件を満たす I/O ピン配置を定義できます。I/O ピン配置の詳細は、[第 8 章「I/O ピン配置」](#)を参照してください。

フロアプラン

PlanAhead では、関連するロジックをまとめて制約を付けることで、遅延が小さくなるようインターコネクトを短くしたり、予測されるインプリメンテーション結果が得られるようにできるフロアプラン手法がサポートされています。デザインをフロアプランするには、物理ブロック (Pblock) ロケーションを作成してロジック配置を制約するか、または個々のロジック オブジェクトを特定デバイス サイトにロックします。フロアプラン機能の詳細は、[第 10 章「デザインのフロアプラン」](#) および [付録 E「その他のリソース」](#) に示す『フロアプラン手法ガイド』(UG633) を参照してください。

デザインのプログラムとデバッグおよび ChipScope の統合

PlanAhead にはザイリンクス ChipScope デバッグ ツールが統合されており、デザインにデバッグ コアを追加できます。

インプリメンテーションが終了すると、PlanAhead から直接 ISE ツールにアクセスしてビットストリーム ファイルを作成し、iMPACT、ChipScope Analyzer ソフトウェア ツールを起動できます。詳細は、[第 12 章「デザインのプログラムとデバッグ」](#) を参照してください。

階層デザイン、デザイン保持、パーシャル リコンフィギュレーション

PlanAhead には、チーム設計、デザイン保持、およびパーシャル リコンフィギュレーションをサポートするための階層機能が含まれています。詳細は、[第 13 章「階層デザイン手法の使用」](#) を参照してください。

階層デザイン手法については、[付録 E「その他のリソース」](#) に示す『階層デザイン手法ガイド』(UG748) にも詳細に説明されています。

Tcl コマンドおよびバッチ スクリプト

PlanAhead ツールの Tcl (Tool Command Language) 構文およびバッチ オプションの詳細は、[第 14 章「Tcl およびバッチ スクリプト」](#) を参照してください。

PlanAhead ツールでサポートされる Tcl コマンドについては、[付録 E「その他のリソース」](#) に示す『PlanAhead Tcl コマンド リファレンス ガイド』(UG789) を参照してください。

ISE Project Navigator 環境での PlanAhead ソフトウェアの使用

PlanAhead は Project Navigator に統合されており、デザイン フローのさまざまな段階でデザイン 結果を向上するために使用できます。

Project Navigator は、次のデザイン プロセスで PlanAhead を起動します。

- 合成前
 - I/O ピンの配置
- 合成後
 - I/O ピンの配置
 - エリア/IO/ロジックのフロアプラン
- インプリメンテーション後
 - タイミングの解析とデザインのフロアプラン

ISE 統合モードの詳細は、第 15 章「Project Navigator からの PlanAhead の使用」を参照してください。

入力および出力ファイル

PlanAhead では、さまざまな入力ファイルを使用でき、さまざまなタイプおよびフォーマットの出力ファイルが生成されます。入力ファイルと出力ファイルの詳細は、付録 A 「PlanAhead の入力ファイルおよび出力ファイル」を参照してください。

デザイン ルール チェック

PlanAhead のデザイン ルール チェック (DRC) は、付録 B, PlanAhead の DRC にリストされています。

PlanAhead ソフトウェアの用語

本書で使用される用語の定義については、付録 E 「その他のリソース」に示す『用語集』を参照してください。

ソフトウェアのアップデート

ザイリンクスでは、XilinxNotify ユーティリティを使用して入手可能なアップデートを通知します。詳細は、付録 C 「XilinxNotify を使用したリリースのインストール」を参照してください。

複数の Linux ホストの設定

複数のホストで実行するには、セキュア シェル (SSH) を使用します。PlanAhead で複数のホストを設定する前に、リモート マシンにログインするたびにパスワードを入力する必要がないように SSH を設定できます。

詳細は、付録 D 「パスワード入力なしの SSH の設定」を参照してください。

その他のリソース

付録 E 「その他のリソース」に、このガイドで参照されているすべての文書へのリンクがリストされています。印刷版を使用している場合を考慮し、URL も示されています。

PlanAhead ツールの起動

PlanAhead はどのディレクトリからも起動できますが、プロジェクト ディレクトリから起動すると、ログ ファイルおよびジャーナル ファイルがその起動ディレクトリに書き込まれ、それらを簡単に見つけることができるので便利です。

Linux

Linux で PlanAhead を起動するには、コマンド プロンプトに次のコマンドを入力します。

```
# planAhead
```

Windows



Windows で PlanAhead を起動するには、デスクトップから PlanAhead のアイコンをダブルクリックします。

ログ ファイルを保存するディレクトリは、デスクトップ アイコンを右クリックして [プロパティ] をクリックし、[作業フォルダー] で指定できます。

PlanAhead が起動すると、16 ページの図 1-1 に示す Getting Started ページが表示されます。

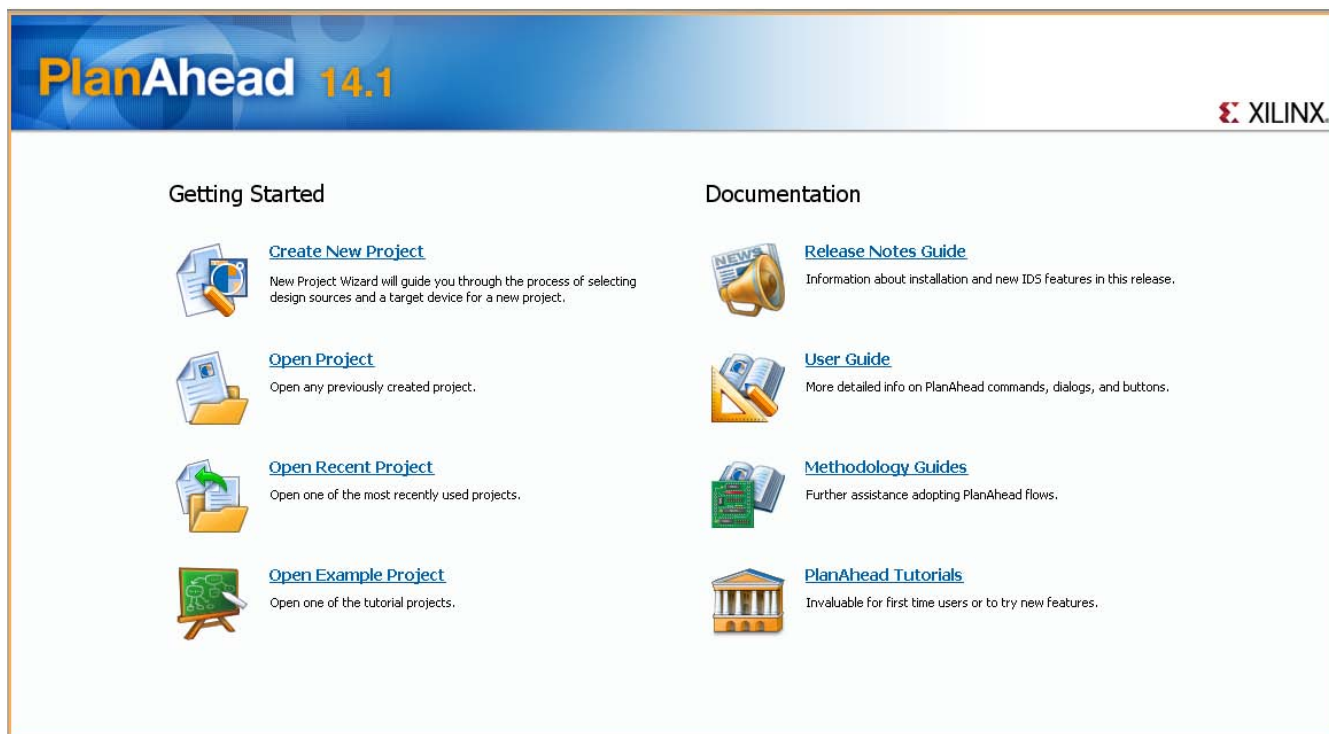


図 1-1: PlanAhead ソフトウェアの Getting Started ページ

PlanAhead の Getting Started ページからは、プロジェクトを作成したり、指定プロジェクトを開くだけでなく、マニュアルも表示できます。このページは、開いているプロジェクトをすべて閉じると表示されます。

Getting Started ページの使用

PlanAhead を起動すると、Getting Started ページが開きます。コマンド リンクをクリックすると、特定のコマンドを実行したり、マニュアルを表示できます。実行可能なコマンドは、次のとおりです。

- [Create New Project] : New Project ウィザードを起動し、PlanAhead デザインプロジェクトを作成します。
- [Open Project] : PlanAhead デザインプロジェクトファイル (.ppr) または ISE Design Suite プロジェクトファイル (.xise) を開くことのできるダイアログ ボックスが表示されます。
- [Open RecentProject] : 最近開いた 10 個のプロジェクトへのリンクを表示します (ファイル数は [Tools] → [Options] → [General] から設定可能)。プロジェクトの表示前に、そのプロジェクト データが存在するかどうかを確認されます。
- [Open Example Project] : 次のサンプル デザイン プロジェクトを開きます。

- [BFT Core]: 小型の RTL プロジェクトである BFT Core (bft.ppr)
- [CPU (HDL)]: 大型の RTL プロジェクトである CPU (HDL)
- [CPU (Synthesized)]: ネットリスト ベースのプロジェクトである CPU (合成済み)
- [Wave (HDL)]: CORE Generator からの 3 つのエンベデッド IP コアを含む IP サンプル デザインである Wave(HDL)。このデザインは、PlanAhead デザイン プロジェクトで IP コアをどのように使用するか確認するためのリファレンス プロジェクトとして使用してください。

PlanAhead のマニュアルのリンクをクリックすると、マニュアルが PDF ビューアーまたはウェブ サイトで表示されます。

注記: PlanAhead のインストール ディレクトリには、このユーザー ガイドへのリンクを示す PDF ファイルが含まれています。この PDF ファイルに示される URL をクリックし、指示に従って最新版のユーザー ガイドを指定の場所にダウンロードすると、[User Guide] リンクをクリックしたときにダウンロードしたユーザー ガイドが開くようになります。

- [Release Notes Guide]: リリース ノートへのリンクが示されます。
- [User Guide]: 本書へのリンクが示されます。
- [Methodology Guides]: ザイリンクスの手法ガイドへのリンクが示されます。
- [PlanAhead Tutorials]: ザイリンクス チュートリアルとサポートされるデザイン データへのリンクが示されます。

コマンド ライン オプション

PlanAhead には、コマンド ライン オプションが複数あります。PlanAhead のコマンド ライン オプションを表示するには、次のコマンドをコマンド プロンプトに入力します。

```
# planAhead -help
```

ヘルプ メニューがシェル ウィンドウに表示されます。

スタートアップ Tcl スクリプトの使用

[Tools] → [Run Tcl Script] をクリックすると、スクリプトを実行できます。

また、planAhead.jou ファイルから PlanAhead の Tcl コマンドをコピーするか、Tcl コンソールから、スタートアップ スクリプトを作成できます。図 1-2 に、Tcl スクリプトの一部を示します。

```

#-----
create_project project_1 {C:\Data\PlanAhead_Designs\PlanAhead_Tutorial\Tutorial_Created_Data\project_1}
set_property design_mode RTL [get_property srcset [current_run]]
import_files -force -norecurse {C:\Data\PlanAhead_Designs\12_demo\Sources\Therm}
set_property library work [get_files -of_objects [get_property srcset [current_run]]] {{C:\Data\PlanAhead_Designs\12_demo\Therm}}
import_files -fileset [get_property constrset [current_run]] -force -norecurse {C:\Data\PlanAhead_Designs\12_demo\Therm}
set_property top therm [get_property srcset [current_run]]
set_property verilog_2001 true [get_property srcset [current_run]]
set_property verilog_uppercase false [get_property srcset [current_run]]
set_property loop_count 1000 [get_property srcset [current_run]]
launch_runs -runs synth_1 -jobs 1
launch_runs -runs impl_1 -jobs 1
close_project

```

図 1-2: Tcl スクリプトの例

PlanAhead のジャーナル ファイルの詳細は、[454 ページの「ジャーナル ファイル \(planAhead.jou および planAhead.jou.backup\)」](#)を参照してください。PlanAhead ツールでの Tcl を使用したスクリプト作成については、[第 14 章「Tcl およびバッチ スクリプト」](#)を参照してください。

PlanAhead ツール フロー

PlanAhead™ ツールは、デザイン フローのさまざまな段階で使用できます。この章で説明されるフローは、作成可能なプロジェクト タイプにも対応します。プロジェクト タイプの詳細は、[第 3 章「プロジェクトの操作」](#)を参照してください。

デザイン解析および制約の定義は、エラボレートされた RTL デザイン、合成済みデザイン、インプリメント済みデザインなど、デザイン フローの各段階で実行できます。

PlanAhead ツールでサポートされるデザイン フローには、次が含まれます。

- **RTL からビットストリーム** : PlanAhead では、RTL 開発から IP のカスタマイズ、合成、インプリメンテーション、デバイスのプログラムまで、FPGA デザイン フロー プロセスすべてを管理できます。Verilog および VHDL ソース、前に定義および設定した IP (Intellectual Property) コア、System Generator からの DSP モジュール、EDK からのエンベデッド プロセッサ デザイン、物理制約およびタイミング制約をプロジェクトに追加できます。デザイン要件が満たされるよう、合成オプション、シミュレーション オプション、インプリメンテーション オプション、制約を組み合わせ、さまざまなデザインを試すことも可能です。
- **合成ネットリストからビットストリーム** : PlanAhead ツールでは、インプリメンテーションからデバイスのプログラムまでを管理できます。合成済みネットリスト、ネットリスト ベースの IP コア、制約をプロジェクトに追加できます。デザイン要件が満たされるよう、インプリメンテーション オプション、制約を組み合わせ、さまざまなデザインを試すことも可能です。
- **デバイスの解析と I/O ピン配置** : PlanAhead には、デバイス リソースを解析し、FPGA とシステム レベルのデザインとの関係を図示する I/O 配置環境があります。クロック供給および I/O 配置を正しく行うことにより、デバイスのパフォーマンスおよび配線性が向上します。早期に I/O を配置することで、プリント回路基板 (PCB) の配線、シグナル インテグリティ、およびシステム全体のパフォーマンスも向上します。デザイン最上位にピン配置を最初に使用すると、I/O ピン配置プロジェクトを RTL プロジェクトに変換することもできます。
- **インプリメント済みデザイン結果の解析** : ISE® Design Suite など、PlanAhead ツール外で生成されたインプリメンテーション結果を解析できます。配置およびタイミング結果を解析すると、デザイン変更、タイミング調整、またはフロアプランを確認して、タイミング クロージャーを達成できます。
- **パーシャル リコンフィギュレーション⁽¹⁾** : PlanAhead には、パーシャル リコンフィギュレーション プロジェクトを設定および管理する環境が含まれています。これらのデザインには、リコンフィギュラブル モジュールを管理するための特別な機能およびプロジェクト構造が必要です。詳細は、[付録 E「その他のリソース」](#)に示す『パーシャル リコンフィギュレーション ユーザー ガイド』(UG702)を参照してください。

1. パーシャル リコンフィギュレーションを使用するには、別途ライセンスが必要です。

デザイン フロー

PlanAhead のデザイン フローは、入力ファイルの種類によって異なります。このセクションでは、デザイン フローとデザイン タスクについて説明します。PlanAhead の一般的なデザイン フロー、入力および出力を 図 2-1 に示します。

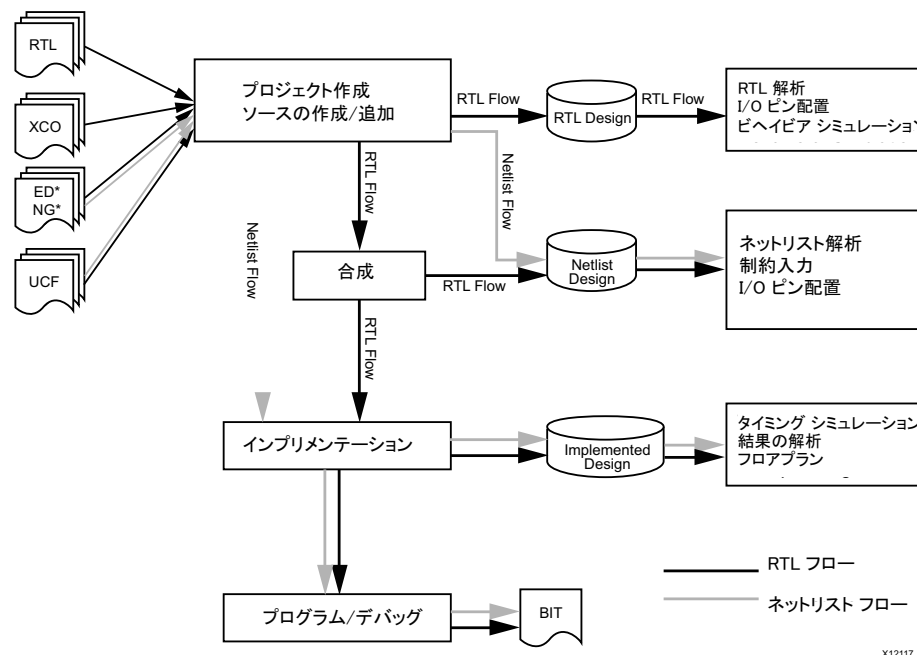


図 2-1 : PlanAhead デザイン フロー

プロジェクトの作成と管理

PlanAhead には、プロジェクトの作成お、ソース ファイルの作成またはプロジェクトへの追加を管理するウィザードがあります。次の操作を実行できます。

- 使用しているフローやソース タイプのプロジェクトを作成
- 新規ソース ファイルの作成または既存ソース ファイルのプロジェクトへの追加
- リモートの書き込み禁止ファイルを参照、またはローカル プロジェクト フォルダにコピー
- プロジェクト内でソース ファイルのディスエーブル/イネーブルを切り替え
- さまざまな制約オプションおよびデバイスを指定した複数の制約セットを作成
- ビヘイビアー シミュレーションおよびタイミング シミュレーション用にシミュレーション ソースを作成
- プロジェクトをアーカイブして、デザインのバックアップを作成または移動しやすいコピーを作成

PlanAhead プロジェクト環境では、1 つのプロジェクトで複数のデザイン制約を作成し、保存できるので、複数の RTL ソース バージョン、制約セット、ターゲット デバイス、合成済みネットリスト、さまざまなインプリメンテーション ストラテジを使用したインプリメンテーション実行結果を作成できます。また、ソース ファイルを変更したり、デザイン ツールを起動すると、デザイン フロー ステータスが監視され表示されます。

RTL 開発および解析

PlanAhead には、ソース ファイルを作成および編集するテキスト エディターが含まれています。ザイリンクス テンプレート ライブラリから、サンプル ロジック構文を直接コピーできます。[Find in Files] 機能を使用すると、これらのライブラリを検索条件を入力して検索できます。

エラボレート済みデザインを開くと、RTL ソース ファイルがエラボレートされ、RTL ネットリストが自動的に読み込まれ、RTL 構造、構文、ロジック定義をチェックできるようになっています。次の解析およびレポート機能があります。

- RTL のコンパイル検証および構文チェック
- ネットリストおよび回路図の確認
- デザイン ルール チェック (DRC)
- ビヘイビアー シミュレーション
- リソース使用率および消費電力予測
- RTL ポート リストを使用した初期 I/O ピン配置も可能です。

デザインを開いて、オブジェクトを選択すると、RTL ソース ファイル内のインスタンスエーションおよびロジック定義なども含めて、関連するオブジェクトが別のビューでも選択されます。詳細は、第 5 章「エラボレート済み RTL デザイン」を参照してください。

IP のカスタマイズとインプリメンテーション

統合されている CORE Generator™ ツールを使用して、IP を参照、カスタマイズ、インスタンスエート、インプリメント、および自動アップデートできます。

ロジック合成

PlanAhead では、XST (Xilinx® Synthesis Technology) 合成ツールを使用して、合成 run を設定、起動、監視します。詳細は、第 6 章「デザインの合成」を参照してください。

合成 run 用に異なる合成オプションを設定し、再利用可能なストラテジを作成できます。たとえば、消費電力、パフォーマンス、エリア最適化のストラテジを作成できます。

合成 run 結果は随時アップデートされ、レポート ファイルにもアクセスできます。[Compilation Messages] ビューから合成の警告またはエラーを選択すると、ソース ファイルで該当するロジックがハイライトされます。

- 複数の合成 run は、同時にまたは 1 つずつ順に実行できます。
- Linux システムでは、リモート サーバー上の run がローカルで実行されます。

複数の合成 run を実行すると、複数のネットリストが作成され、PlanAhead プロジェクト内に保存されます。これらのネットリストは、解析用に読み込むことができます。ネットリストをインポートしたら、デバイスとデザイン解析、I/O 配置の制約の作成、フロアプラン、およびインプリメンテーションを実行できます。

デザインを合成すると、クロックとロジックを解析および配置できるようになり、合成済みネットリストに対しては最も包括的なデザインルール チェック (DRC) を実行できます。

I/O ピン配置

PlanAhead には I/O ピン配置環境が含まれており、デバイス パッケージ ピンまたは内部ダイ パッドに I/O ポートを正しく割り当てることができます。さまざまな表示ビューおよび表があり、パッケージおよび I/O 関連データを解析できます。詳細は、[第 8 章「I/O ピン配置」](#)を参照してください。

次の操作を実行できます。

- 内部 I/O 接続を調べ、デバイスを介するデータ フローが適切かどうか、内部デバイス リソースへのアクセスが最適であるかどうかを確認します。
- 外部接続および内部接続の要件に基づいて適切な選択を行うことにより、システム パフォーマンスを向上します。
- DRC および同時スイッチ ノイズ (SSN) 解析を使用し、接続要件に準拠していることを確認します。
- CSV、UCF、RTL、合成済みネットリストなど、さまざまなフォーマットの入力を使用して I/O ピン配置を開始します。

クロック ロジックが I/O の割り当てに影響するので、合成済みネットリストをソースとして使用すると、より多くの DRC チェックが実行されます。最終的な I/O 検証ステップでは、インプリメンテーション ツールを使用して完全なデザインを実行します。

ネットリストの解析および制約の定義

PlanAhead には、デザイン解析および制約の設定機能があります。デザイン データはさまざまな形式で表示でき、お互いに連動しています。

ダイ内部および外部パッケージを表示するインタラクティブなグラフィカルビューがあり、デバイス リソースを解析して制約を適用できます。タイミング制約および物理制約を適用し、解析できます。

初期段階でタイミング解析 (タイミング シミュレーションを含む)、リソース予測、接続解析、デザイン ルール チェック (DRC) を実行することにより、インプリメンテーション前にデザインの問題を発見できます。

インプリメンテーション

PlanAhead では、ISE® Design Suite を使用して、インプリメンテーション run を設定、起動、監視できます。

インプリメンテーション run 用に異なるインプリメンテーション オプションを設定し、再利用可能なストラテジを作成できます。たとえば、短いランタイム、パフォーマンス、エリア最適化のストラテジを作成できます。

インプリメンテーション run の結果はインタラクティブに表示され、レポート ファイルにも簡単にアクセスできます。

複数のインプリメンテーション run は、同時にまたは 1 つずつ順に実行できます。Linux OS を使用すると、リモート サーバーを使用できます。制約セットを作成して、さまざまな論理制約、物理制約、代替デバイスを設定して試すことができます。

結果の解析およびフロアプラン

さまざまな run 結果を読み込み、解析およびフロアプランを実行できます。インプリメント済みデザインでの機能は、次の章を参照してください。

- [第 7 章「合成済みデザインの制約および解析」](#)
- [第 11 章「インプリメンテーション結果の解析」](#)

PlanAhead から実行した任意の run をインポートできます。

インプリメント済みデザインを開くと、オリジナルのネットリスト、制約、およびインプリメンテーション結果が読み込まれます。複数のデザインを同時に開くことができます。ISim (タイミングシミュレーション)、FPGA Editor、および XPower Analyzer ツールを PlanAhead から直接起動して、さらにデザインを解析することもできます。

デバイスのプログラム

完了したどのインプリメンテーション run に対しても、ビットストリーム ファイルを生成できます。BIT ファイル生成オプションも設定可能です。iMPACT ツールを起動してデバイスをコンフィギュレーションおよびプログラムできます。

デザインの検証とデバッグ

合成済みデザインでは、ChipScope™ Pro Analyzer ツールおよび ILA (Integrated Logic Analyzer) や ICON (Integrated Controller) などの IP コアをインプリメントして、必要なプローブ信号を選択および設定できます。ビットストリーム ファイルが生成されているどの run に対しても、ChipScope Analyzer ツールを起動できます。

PlanAhead から直接 ChipScope Analyzer ツールを起動して、配線およびデバイス リソースを解析できます。

ユーザー モデル

PlanAhead のグラフィカル ユーザー インターフェイス (GUI) は層構造になっており、新規ユーザーにわかりやすい環境を提供すると共に、アドバンス機能にも簡単にアクセスできるようになっています。デフォルトでは、PlanAhead はプッシュ ボタン フローで開きます。このフローは、アドバンス解析やフロアプランが必要ない場合に適しています。フローは、Flow Navigator というビューで管理されます。Flow Navigator については、[24 ページの「Flow Navigator の理解」](#)で説明します。

基本的なユーザー フロー

PlanAhead では、開発サイクル全体を Flow Navigator のボタンをクリックするだけで実行できます。ソース ファイルをインポートした後、デザイン ロジックを合成、合成済みネットリストをインプリメント、結果を解析、ビットストリームを生成、プログラムおよび検証ツールを起動することにより、FPGA 開発プロセスを最初から最後まで実行します。

アドバンス機能

PlanAhead では、デザイン フローの各段階で、アドバンス デザイン設定および解析用の環境が提供されています。エラボレートされた RTL デザイン、合成済みデザイン、およびインプリメンテーション済みデザインを読み込み、解析および制約の定義を実行できます。これらの環境の詳細は、[30 ページの「デザインの操作」](#)を参照してください。

PlanAhead では、1 つのプロジェクトでデザインの複数のバリエーションを作成し、保存できるので、複数の RTL ソース バージョン、制約セット、ターゲット デバイス、合成済みネットリスト、さまざまなインプリメンテーション ストラテジを使用したインプリメンテーション実行結果を作

成できます。ソース ファイルを変更したり、デザイン ツールを起動すると、デザインのステータスが表示されます。

複数の合成およびインプリメンテーションの **run** を、ローカルまたはリモートの **Linux** サーバーで設定、起動、および監視できます。異なるコマンド オプション、制約、デバイスを使用して試すことができます。

Flow Navigator の理解

Flow Navigator では、プロジェクトの設定、合成、インプリメンテーション、ビットストリームの作成などの主なデザイン プロセス タスクを制御します。これらのタスクが完了したら、Flow Navigator の [Open Elaborated Design]、[Open Synthesized Design]、または [Open Implemented Design] をクリックしてデザインを開き、結果を解析したり、制約を適用したりできます。これらのデザイン表示には、デザイン フローに適したよく使用されるコマンドのセットが表示されます。

使用可能なオプションは、デザインのステータスによって異なります。実行できない手順は淡色表示されます。

[図 2-2](#) および [26 ページの図 2-3](#) に、Flow Navigator を使用してデザイン タスクを実行し、デザイン プロセスの異なる段階で解析環境を開く方法を示します。

RTL プロジェクトでの Flow Navigator の使用

[図 2-2](#) に、RTL ソースを入力として使用した場合のデザイン フローを示します。

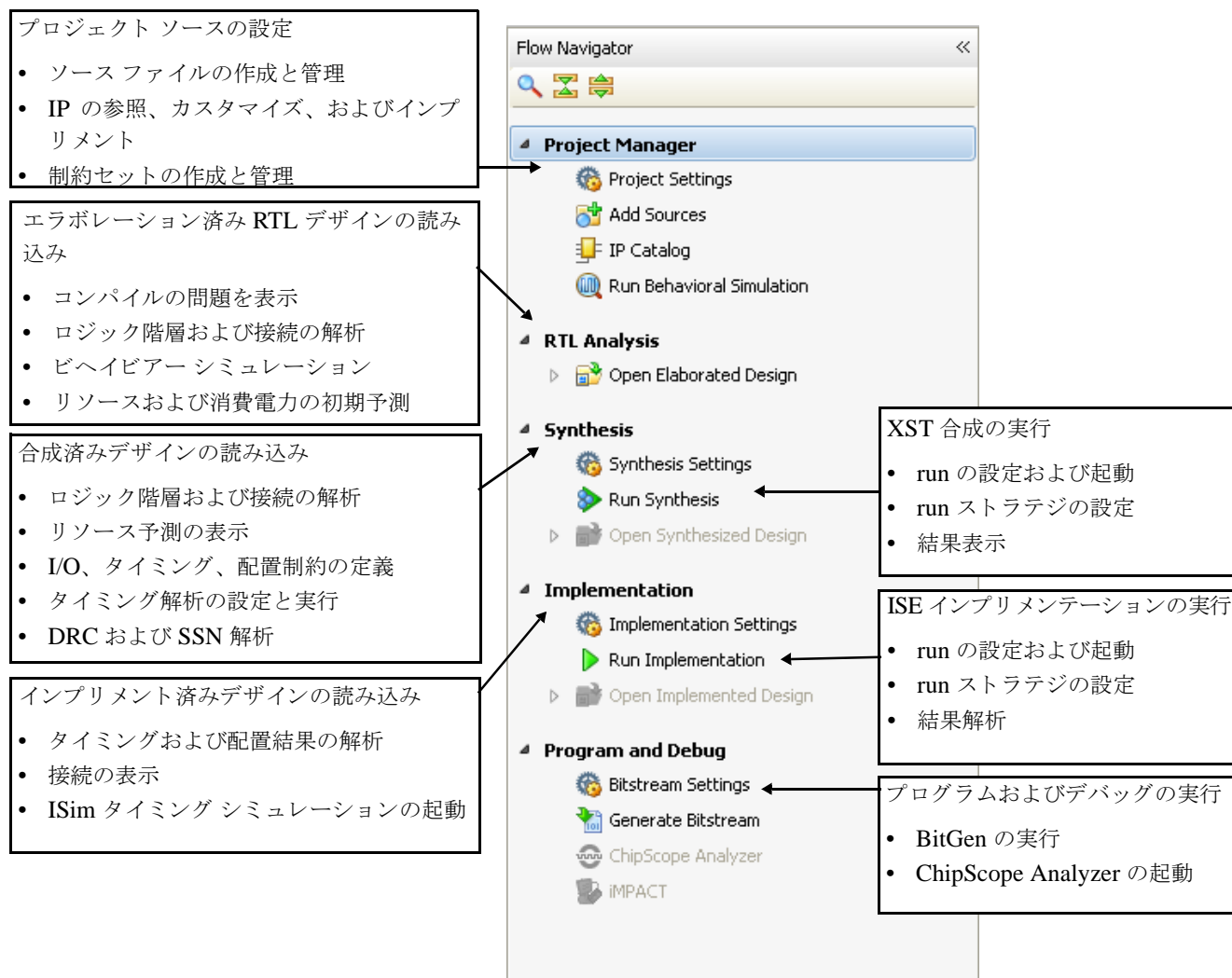


図 2-2 : PlanAhead の Flow Navigator (RTL プロジェクト)

合成済みネットリスト プロジェクトでの Flow Navigator の使用

図 2-3 に、合成済みネットリスト ベースのプロジェクトのデザイン フローを示します。

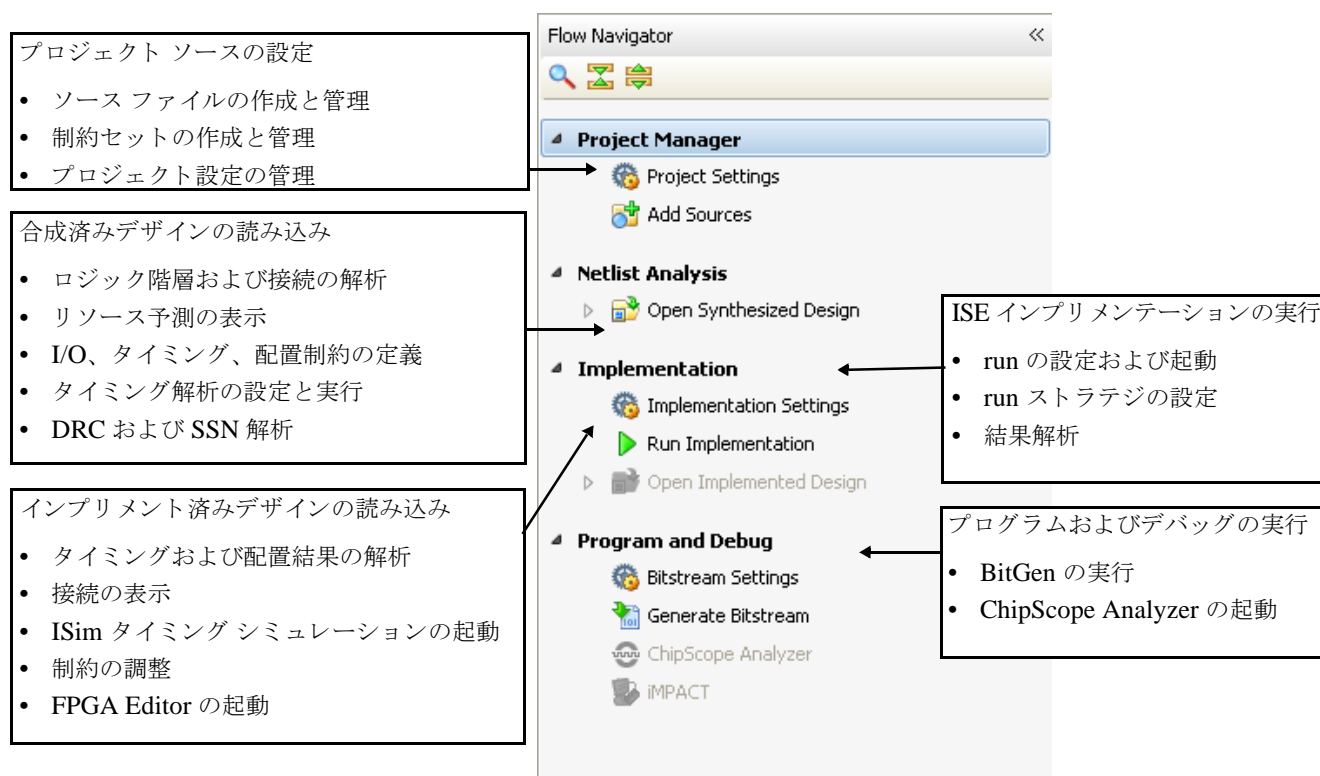


図 2-3 : PlanAhead ソフトウェア Flow Navigator (合成済みネットリスト プロジェクト)

Flow Navigator からのコマンドの実行

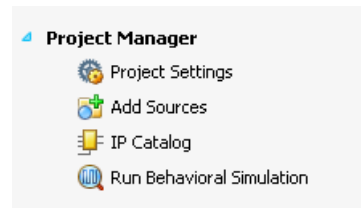
Flow Navigator を使用すると、プロジェクトにソース ファイルを追加した後、合成およびインプリメンテーションをボタンをクリックするだけで実行できます。デザインを完成させるのに、デザイン環境を開く必要はありません。次のセクションに、Flow Navigator を使用してデザインを完成する方法を説明します。

[Project Manager] メニュー

プロジェクトを開くと、デフォルトで Project Manager が開きます。Project Manager を開いた場合、デザインのコンパイルは実行されず、メモリにデザイン データは読み込まれません。Project Manager には、デフォルトで [Sources] ビュー、[Properties] ビュー、および [Project Summary] ビューが表示されます。Project Manager から、ソース ファイルおよび制約セットを作成、インポート、管理できます。また、ザイリンクス IP カタログから IP を検索、カスタマイズ、および作成できます。

Flow Navigator の [Project Manager] メニューには、次のコマンドが含まれます。

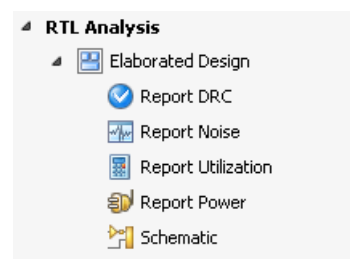
- [Project Manager] : 現在のプロジェクトの [Project Summary] ビューを開きます。詳細は、[97 ページの「\[Project Summary\] ビュー」](#)を参照してください。
- [Project Settings] : [Project Settings] ダイアログ ボックスを開きます。詳細は、[101 ページの「プロジェクト設定」](#)を参照してください。
- [Add Sources] : [Add Sources] ダイアログ ボックスを開き、ソース ファイルを追加します。詳細は、[61 ページの「プロジェクト ソースの管理」](#)を参照してください。
- [IP Catalog] : [IP Catalog] ビューを開きます。詳細は、[75 ページの「IP コアの管理」](#)を参照してください。
- [Behavioral Simulation] : ISim を起動し、エラボレート済みデザインのパシビエーター シミュレーションを実行します。詳細は、[214 ページの「パシビエーター シミュレーションの実行」](#)を参照してください。



[RTL Analysis] メニュー

Flow Navigator の [RTL Analysis] メニューには、RTL ソース ファイルをエラボレートおよび解析する次のコマンドが含まれます。

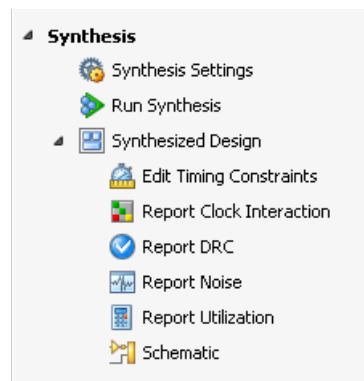
- [RTL Analysis] : [RTL Analysis] コマンドのメニューを三角矢印アイコンをクリックして展開/非展開表示します。
- [Elaborated Design]/[Open Elaborated Design] : 表示されるコマンドはエラボレート済みデザインが開いているかどうかによって異なります。
 - [Elaborated Design] : 開いているエラボレート済みデザインをアクティブ デザインにします。
 - [Open Elaborated Design] : エラボレート済みデザインが開いていない場合、関連するデザイン制約を使用して現在の最上位モジュールに基づいて RTL ソース ファイルをエラボレートし、回路図ビューを作成して開きます。詳細は、[202 ページの「RTL ソース ファイルのエラボレーションと解析」](#)を参照してください。
- [Report DRC] : [220 ページの「RTL DRC の実行」](#)に説明されるように、アクティブなエラボレート済みデザインで RTL ベースのデザイン ルール チェックを実行します。
- [Report Noise] : ターゲット パーツに該当する SSO/SSN 解析を実行します。詳細は、[317 ページの「ノイズ解析予測の使用」](#)を参照してください。
- [Report Utilization] : エラボレート済みデザインの使用量統計をレポートします。詳細は、[204 ページの「エラボレート済みデザインのリソース使用」](#)を参照してください。
- [Report Power] : エラボレート済みデザインの消費電力概算を実行し、消費電力分配の早期ビューを表示します。詳細は、[208 ページの「消費電力予測」](#)を参照してください。
- [Schematic] : エラボレート済みデザインの現在の最上位モジュールの回路図ビューを作成して開きます。詳細は、[163 ページの「\[Schematic\] ビュー」](#)を参照してください。



[Synthesis] メニュー

Flow Navigator の [Synthesis] メニューには、合成済みデザインを管理および解析する次のコマンドが含まれます。

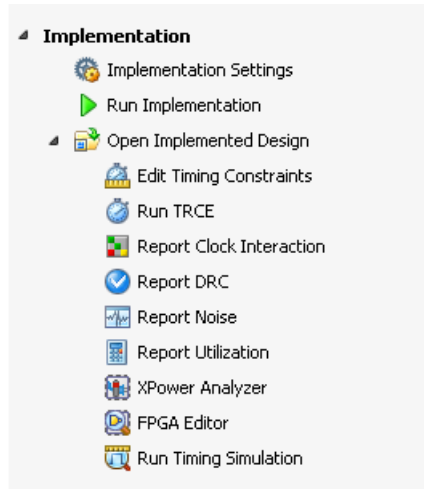
- [Synthesis] : [Synthesis] コマンドのメニューを三角矢印アイコンをクリックして展開/非展開表示します。
- [Synthesis Settings] : 合成設定を変更する [Project Settings] ダイアログ ボックスが開きます。詳細は、[105 ページの「\[Project Settings\] ダイアログ ボックスの \[Synthesis\] ページ」](#)を参照してください。
- [Run Synthesis] : アクティブな合成 run を開始します。詳細は、[231 ページの「合成の実行」](#)を参照してください。
- [Synthesized Design]/[Open Synthesized Design] : 表示されるコマンドは合成済みデザインが開いているかどうかによって異なります。
 - [Synthesized Design] : 開いている合成済みデザインをアクティブ デザインにします。
 - [Open Synthesized Design] : 合成済みデザインが開いていない場合、合成済みネットリストを読み込んで、ターゲット パーツにマップし、指定された制約ファイルを適用します。詳細は、[235 ページの「合成済みデザイン環境の使用」](#)を参照してください。
- [Edit Timing Constraints] : [Timing Constraints] ビューを開いて、合成済みデザインのタイミング制約を表示、編集、作成できるようにします。詳細は、[245 ページの「タイミング制約の定義」](#)を参照してください。
- [Report Clock Interaction] : クロックの相互関係およびクロック ドメイン間をまたがる信号の情報がレポートされます。詳細は、[270 ページの「クロックの相互関係の解析」](#)を参照してください。
- [Report DRC] : [278 ページの「デザイン ルール チェック \(DRC\) の実行」](#)に説明されるように、合成済みデザインでデザイン ルール チェックを実行します。
- [Report Noise] : ターゲット パーツに該当する SSO/SSN 解析を実行します。詳細は、[317 ページの「ノイズ解析予測の使用」](#)を参照してください。
- [Report Utilization] : 合成済みデザインの使用量統計をレポートします。詳細は、[237 ページの「リソース統計の表示とレポート」](#)を参照してください。
- [Schematic] : 合成済みデザインの現在の最上位モジュールの回路図ビューを作成して開きます。詳細は、[163 ページの「\[Schematic\] ビュー」](#)を参照してください。



[Implementation] メニュー

合成が完了したら、Flow Navigator でインプリメンテーション ツールを実行して配置配線を実行できます。Flow Navigator の [Implementation] メニューには、インプリメント済みデザインを管理および解析する次のコマンドが含まれます。

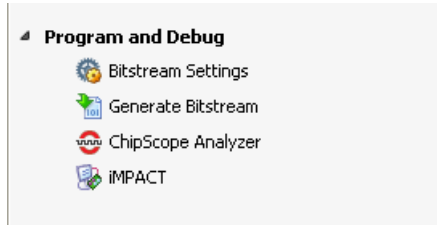
- [Implementation] : [Implementation] コマンドのメニューを三角矢印アイコンをクリックして展開/非展開表示します。
- [Implementation Settings] : 合成設定を変更する [Project Settings] ダイアログ ボックスが開きます。詳細は、[107 ページの「\[Project Settings\] ダイアログ ボックスの \[Implementation\] ページ」](#)を参照してください。
- [Run Implementation] : アクティブなインプリメンテーション run を開始します。合成を実行していない場合でも、[Run Implementation] を使用すると、合成 run を終了できます。詳細は、[第 9 章「デザインのインプリメンテーション」](#)を参照してください。
- [Implemented Design]/[Open Implemented Design] : 表示されるコマンドはインプリメント済みデザインが開いているかどうかによって異なります。
 - [Implemented Design] : 開いているインプリメント済みデザインをアクティブ デザインにします。
 - [Open Implemented Design] : インプリメント デザインが開いていない場合は、合成済みネットリストが読み込まれ、デザイン制約が処理され、配置配線結果を読み込んで、タイミング結果を読み込んで、メモリのインプリメント済みデザインを開きます。詳細は、[377 ページの「インプリメント済みデザインを開く」](#)を参照してください。
- [Edit Timing Constraints] : [Timing Constraints] ビューを開いて、インプリメント済みデザインのタイミング制約を表示、編集、作成できるようにします。詳細は、[245 ページの「タイミング制約の定義」](#)を参照してください。
- [Run TRCE] : このツールでは、タイミング制約を満たしているかが確認され、入力制約に対してデザインが準拠しているかどうかをリストしたレポート ファイルが出力されます。詳細は、[380 ページの「タイミング結果の解析」](#)を参照してください。
- [Report Clock Interaction] : クロックの相互関係およびクロック ドメイン間をまたがる信号の情報がレポートされます。詳細は、[270 ページの「クロックの相互関係の解析」](#)を参照してください。
- [Report DRC] : [278 ページの「デザイン ルール チェック \(DRC\) の実行」](#)に説明されるように、インプリメント済みデザインでデザイン ルール チェックを実行します。
- [Report Noise] : ターゲット パーツに該当する SSO/SSN 解析を実行します。詳細は、[317 ページの「ノイズ解析予測の使用」](#)を参照してください。
- [Report Utilization] : 合成済みデザインの使用量統計をレポートします。詳細は、[237 ページの「リソース統計の表示とレポート」](#)を参照してください。
- [XPower Analyzer] : インプリメント済みデザインの消費電力解析を実行します。詳細は、[404 ページの「XPower Analyzer を使用した消費電力分配解析」](#)を参照してください。
- [FPGA Editor] : FPGA Editor でインプリメント済みデザインを開きます。詳細は、[406 ページの「FPGA Editor の起動」](#)を参照してください。



- [Run Timing Analysis] : インプリメント済みデザインのタイミング解析を実行します。詳細は、[397 ページの「タイミング シミュレーションの実行」](#)を参照してください。

[Program and Debug] メニュー

インプリメンテーションが終了したら、ビットストリーム ファイルを生成して、デバッグおよびプログラム ツールを起動します。Flow Navigator の [Program and Debug] メニューには、次のコマンドが含まれます。



- [Program and Debug] : [Program and Debug] コマンドのメニューを三角矢印アイコンをクリックして展開/非展開表示します。
- [Bitstream Settings] : 合成設定を変更する [Project Settings] ダイアログ ボックスが開きます。詳細は、[108 ページの「\[Project Settings\] ダイアログ ボックスの \[Bitstream\] ページ」](#)を参照してください。
- [Generate Bitstream] : [Project Settings] ダイアログ ボックスの [Bitstream] ページで設定したオプションで BitGen コマンドが実行されます。
- [ChipScope Analyzer] : ChipScope ILA コアをインプリメントしたインプリメント済みデザインに対して ChipScope Pro Analyzer を起動します。詳細は、[420 ページの「ChipScope Pro Analyzer の起動」](#)を参照してください。
- [iMPACT] : インプリメント済みデザインに対して iMPACT を起動します。詳細は、[420 ページの「iMPACT の起動」](#)を参照してください。

ChipScope Pro Analyzer および iMPACT を起動するには、[Generate Bitstream] コマンドを実行して作成した BIT ファイルが必要です。詳細は、[第 12 章「デザインのプログラムとデバッグ」](#)を参照してください。

デザインの操作

PlanAhead では、デザイン プロセスのさまざまな段階でのデザインを開くことができます。デザインは、ネットリスト (エラボレートされた RTL ネットリストまたは合成済みネットリスト)、制約セット、およびターゲット デバイスで構成されます。デザインをメモリに読み込むことにより、解析、制約の定義、ChipScope デバッグ コアの挿入を実行します。

Flow Navigator には、エラボレート済みデザイン、合成済みデザイン、インプリメント済みデザインを開くボタンがあります。[Implemented Design] をクリックすると、run の実行に使用されたデザイン データが読み込まれます。エラボレート済みデザインおよび合成済みデザインは、異なるターゲット デバイスおよび制約セットを使用して開くことができ、さまざまなバージョンを試すことができます。

エラボレートされた RTL デザイン、合成済みネットリスト デザイン、インプリメント済みデザインなど、さまざまな段階のデザインを解析できます。対応するデザインを開くと、タイミング制約またはフロアプランのような制約およびロジックの配置は、どの段階でも変更できます。

複数の制約ファイルを作成できるので、制約の変更を管理できます。PlanAhead ツール環境では、次のデザインを開くことができます。

- エラボレート済みデザイン : エラボレートされた RTL ソース ファイル、制約セット、およびターゲット デバイス
- 合成済みデザイン : 合成済みネットリスト、制約、およびターゲット デバイス

- インプリメント済みデザイン：インプリメンテーション **run** の合成済みネットリスト、制約、および結果

GUI のデフォルトのビュー レイアウトには、[Schematic] および [Device] ビューのようなデザインのグラフィカル ビューや [Netlist] ビュー、[Properties] ビュー、ワークスペースの下部に表示される [Messages] および [Reports] ビューなどが含まれます。メニュー ツールバーでレイアウトを選択すると、さまざまなビューの表示をアレンジできます。詳細は、[116 ページの「ビュー レイアウトの使用」](#)を参照してください。

エラボレート済みデザインを開く

合成およびシミュレーション用に複数の RTL ソース ファイルを含むエラボレート済み RTL デザインには、デザインで使用するモジュールおよび IP コアの合成済みネットリスト、タイミング制約および物理制約を含む制約ファイル、デザインのターゲット デバイスとなるザイリンクス パーツが含まれます。

Flow Navigator で [Open Elaborated Design] をクリックすると、RTL ソース ファイルが自動的にエラボレートされ、エラボレートされたネットリストがアクティブ制約セットおよびターゲット デバイスと共にメモリに読み込まれます。エラボレーションに関するメッセージが、[Messages] ビューに表示されます。

エラボレート済みデザインを開くには、次のいずれかを実行します。

- Flow Navigator の [RTL Analysis] メニューから [Open Elaborated Design] をクリックし、エラボレート済みネットリスト、アクティブな制約セット、ターゲット デバイスをメモリへ読み込みます。
- メイン メニューから [Flow] → [Open Elaborated Design] をクリックします。
- Flow Navigator の [RTL Analysis] メニューを右クリックし、[New Elaborated Design] をクリックします。
- メイン メニューから [Flow] → [New Elaborated Design] をクリックします。

新しいエラボレート済みデザインの名前、適用する制約、デザインに読み込むターゲット パーツを、[31 ページの図 2-4](#) に示すように指定します。

注記：アクティブでない制約セットを選択した場合は、アクティブにマークすることもできます。詳細は、[62 ページの「制約の管理」](#)を参照してください。

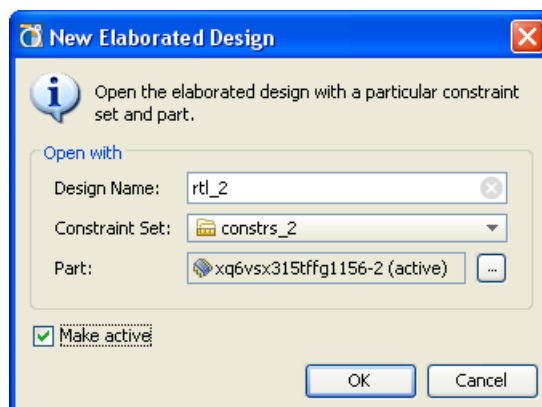


図 2-4：新規エラボレート済みデザイン

[OK] をクリックすると、RTL デザインがエラボレートされて、展開されたネットリストが作成されます。これは完全に合成されたネットリストではありませんが、RTL ファイルから可能な範囲でデザインの解析ができます。

プロセスのバックグラウンドへの移動

PlanAhead でさまざまなデザイン ファイルおよび制約ファイルを読み込んでエラボレーションが実行されると、[Open Elaborated Design] ダイアログ ボックスが開き (図 2-5)、エラボレーション プロセスがバックグラウンドで実行されます。

プロセスがバックグラウンドで実行されると、バックグラウンド タスクを実行させたまま、レポートを表示したり、デザイン ファイルを開いたりといった別の機能が実行できます。この間、レポートを表示したりして、時間を効率的に利用できます。ただし、[Tcl Console] はブロックされるので、Tcl コマンドを使用したり、開いている別のデザインに切り替えるような Tcl コマンドを必要とするタスクは実行できません。

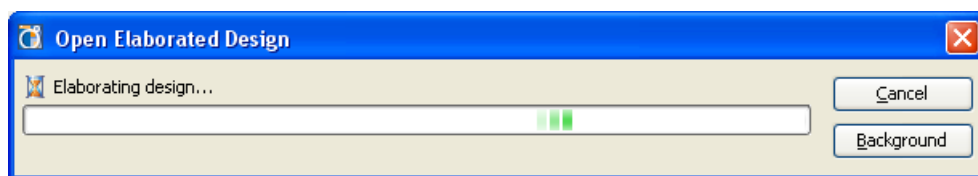


図 2-5 : [Open Elaborated Design] ダイアログ ボックス : バックグラウンド プロセス

次に示すように、最上位回路図の [RTL Netlist] ビューおよび [Schematic] ビューを含むデフォルトのビュー レイアウトで [Elaborated Design] ビューが開きます。

[RTL Netlist] ビューに、エラボレートされたロジック階層が表示されます。RTL ロジック デザインの解析については、第 5 章「エラボレート済み RTL デザイン」を参照してください。

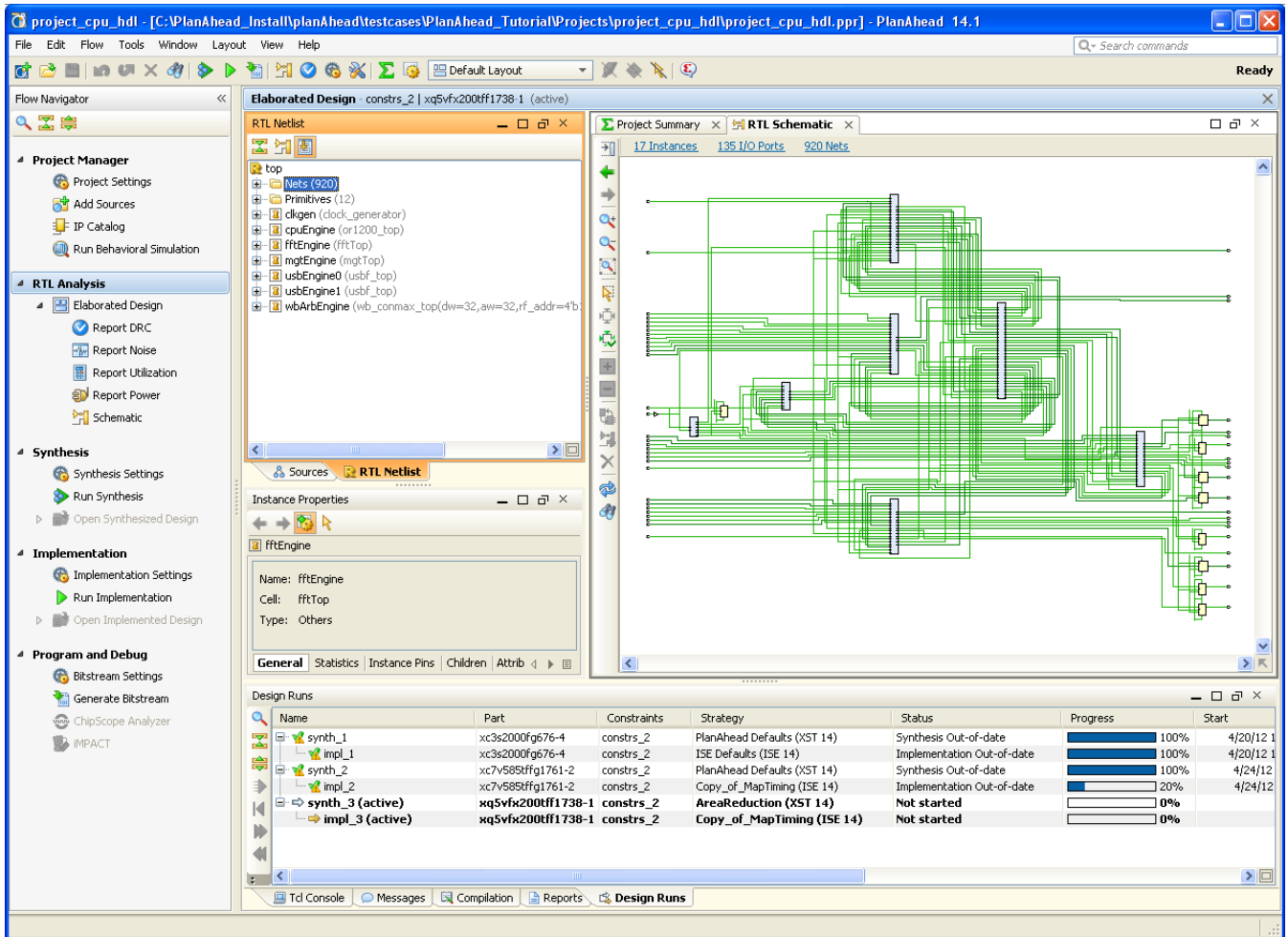


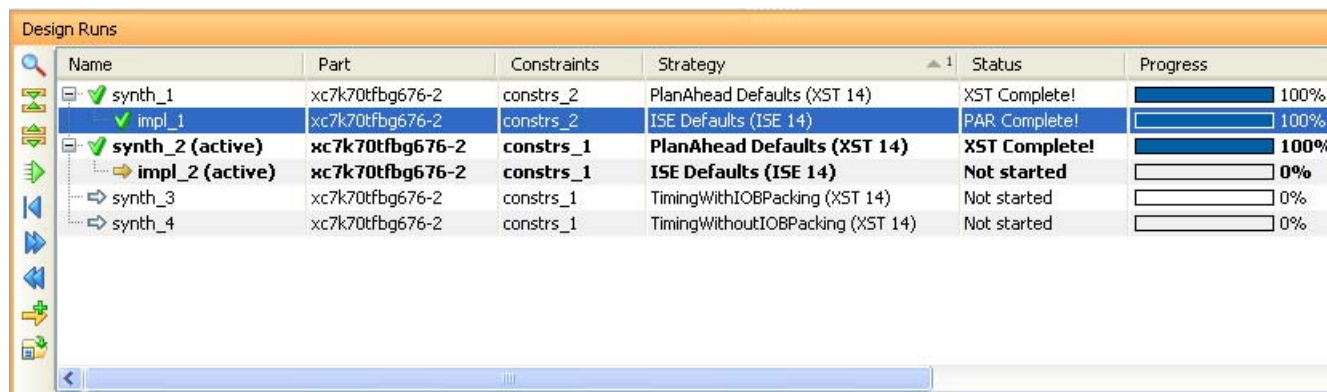
図 2-6 : [RTL Analysis] ビュー

アクティブ run の設定

合成済みデザインまたはインプリメント済みデザインを開くと、デフォルトでネットリストが適用され、現在アクティブな run からの配置配線データが適用されます。Flow Navigator のメニューはアクティブ run によって異なり、アクティブ run で合成が終了しているか、インプリメンテーションが終了しているかによって、[Open Synthesized Design] や [Open Implemented Design] といったコマンドが表示されます。[Project Summary]、[Compilation]、および [Messages] ビューには、アクティブ run に関する情報が表示されます。

合成 run やインプリメンテーション run が複数存在する場合は、現在アクティブな run からネットリストおよびデザイン情報が適用されます。アクティブ run は、[Design Runs] ビューに太字で示されます (34 ページの図 2-7)。

アクティブ run を変更するには、[Design Runs] ビューで [Synthesis Run] または [Implementation Run] を右クリックし、[Make active] コマンドをクリックします。この後、デザインを開くか、run を開始すると、新しいアクティブ run のネットリストまたは配置配線データが使用されます。詳細は、182 ページの「[Design Runs] ビューの使用」を参照してください。



Name	Part	Constraints	Strategy	Status	Progress
synth_1	xc7k70tfg676-2	constrs_2	PlanAhead Defaults (XST 14)	XST Complete!	100%
impl_1	xc7k70tfg676-2	constrs_2	ISE Defaults (ISE 14)	PAR Complete!	100%
synth_2 (active)	xc7k70tfg676-2	constrs_1	PlanAhead Defaults (XST 14)	XST Complete!	100%
impl_2 (active)	xc7k70tfg676-2	constrs_1	ISE Defaults (ISE 14)	Not started	0%
synth_3	xc7k70tfg676-2	constrs_1	TimingWithIOBPacking (XST 14)	Not started	0%
synth_4	xc7k70tfg676-2	constrs_1	TimingWithoutIOBPacking (XST 14)	Not started	0%

図 2-7 : [Design Runs] ビュー

複数の run の作成と管理については、[325 ページ](#)の「インプリメンテーション run の定義」を参照してください。

合成済みデザインを開く

PlanAhead では、合成済みネットリスト、物理制約およびタイミング制約、ターゲット パーツを組み合わせて合成済みデザインが作成されます。

合成済みデザインを開くと、含まれるデザイン エレメントがメモリに読み込まれるので、必要に応じてそれらを解析および変更して、デザインを完了させることができます。合成済みデザインは制約ファイルをアップデートして保存したり、保存せずにデザインを読み込み直して元の状態に戻したりできます。

合成済みデザインを開くには、次のいずれかを実行します。

- [Design Runs] ビューで run 名をダブルクリックします。
- Flow Navigator の [Synthesis] メニューから [Open Synthesized Design] をクリックします。
- メイン メニューから [Flow] → [Open Synthesized Design] をクリックします。
- Flow Navigator の [Synthesis] を右クリックし、[New Synthesized Design] をクリックします。
- メイン メニューから [Flow] → [New Synthesized Design] をクリックします。

アクティブな制約セットおよびターゲット デバイスを使用して合成済みネットリストを開くか、別の制約セットとターゲット デバイスを指定してメモリの合成済みデザインを開くことができます。[35 ページ](#)の図 2-8 に示すように、次を入力して新しい合成済みデザインを定義できます。

- [Design Name] : ビュー パナーに表示される名前を入力します。デザインは、セッション中のみメモリに保存されます。
- [Synthesis Run] : 指定した終了済みの合成 run からのネットリストを使用します。
- [Constraint Set] : ネットリストに対して開く既存の制約セットを選択します。
- [Part] : ターゲット デバイスを選択します。



図 2-8：新規合成済みデザイン

図 2-9 に [Synthesized Design] のデフォルト ビュー レイアウトを表示します。

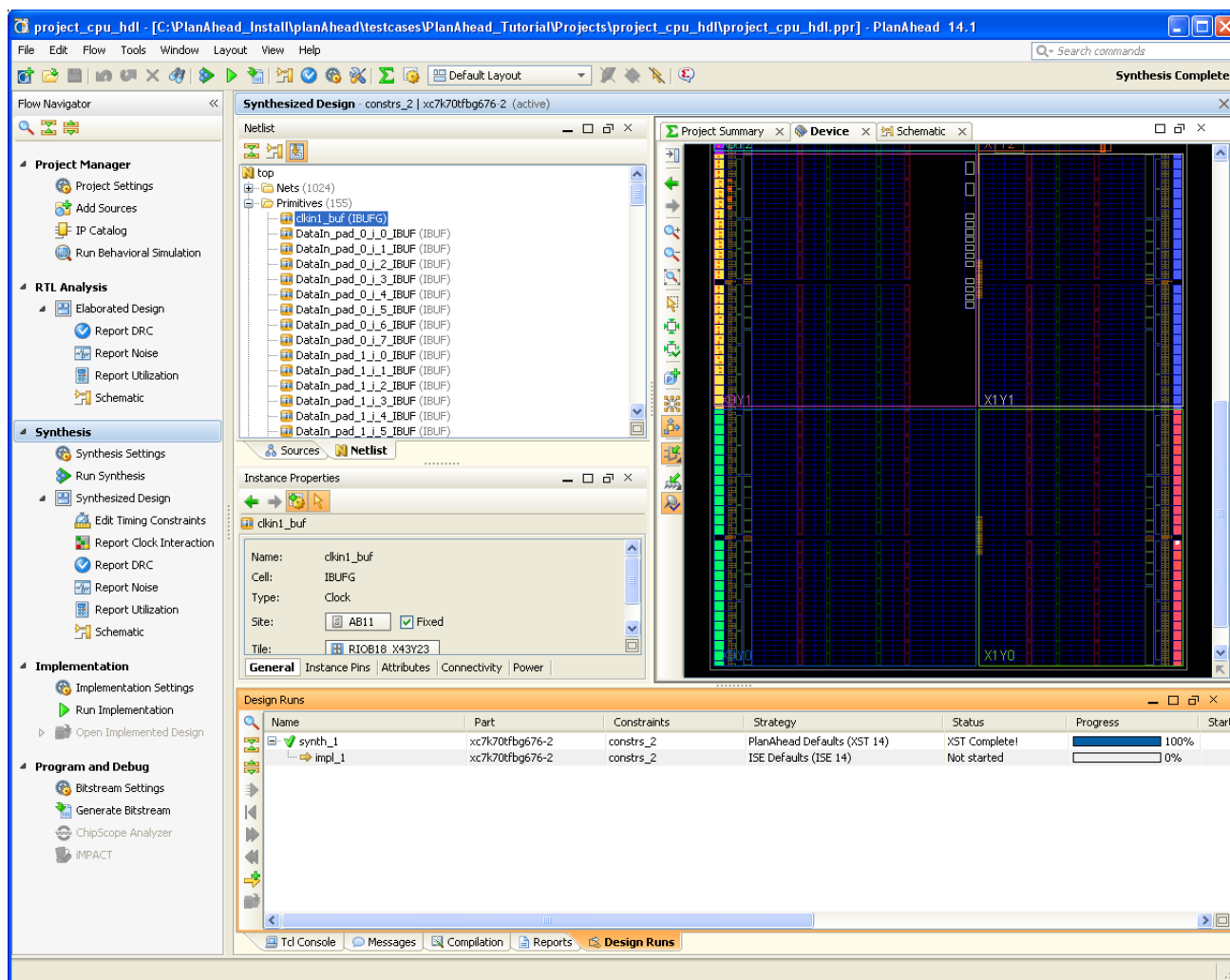


図 2-9：開いた合成済みデザイン

プロセスのバックグラウンドへの移動

PlanAhead でデザイン ファイルおよび制約ファイルを読み込んで合成済みまたはインプリメント済みデザインを開くと、[Open Synthesized Design] ダイアログ ボックスまたは [Open Implemented Design] が開き (図 2-10)、プロセスがバックグラウンドで実行されます。

プロセスがバックグラウンドで実行されると、バックグラウンド タスクを実行させたまま、レポートを表示したり、デザイン ファイルを開いたりといった別の機能が実行できます。この間、前の run を確認したり、レポートを表示したりして、時間を効率的に利用できます。ただし、[Tcl Console] はブロックされるので、Tcl コマンドを使用したり、開いている別のデザインに切り替えるような Tcl コマンドを必要とするタスクは実行できません。

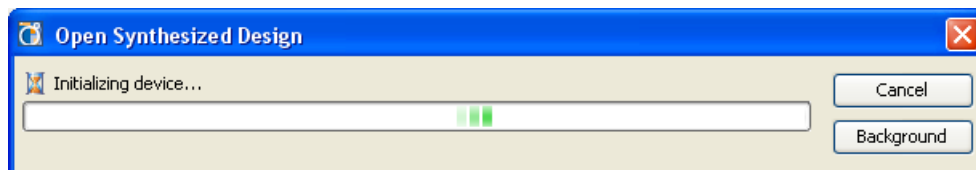


図 2-10 : [Open Synthesized Design] ダイアログ ボックス : バックグラウンド プロセス

インプリメント済みデザインを開く

インプリメント済みデザインには、終了したインプリメンテーション run からの合成済みネットリスト、物理/タイミング制約、ターゲット ザイリンクス パーツ、配置配線データが含まれます。PlanAhead には複数のインプリメンテーション run を設定でき、これらを [Implemented Design] ビューで選択できます。

インプリメント済みデザインには、ネットリスト、制約、配置、およびタイミング結果がインプリメンテーション run ディレクトリからインポートされます。インプリメント済みデザインがメモリに読み込まれたら、デザインを解析および解析できます。

インプリメント済みデザインを開くには、次のいずれかを実行します。

- [Design Runs] ビューで run 名をダブルクリックします。
- メイン メニューから [Flow] → [Open Implemented Design] をクリックします。
- Flow Navigator で [Implementation] → [Open Implemented Design] をクリックします。

Flow Navigator のメニューはアクティブ run の状態によって異なるので、アクティブ run がインプリメントされていない場合は [Open Implemented Design] コマンドが表示されなかったり、淡色表示になっていることもあります。この場合、Flow Navigator で [Implementation] を右クリックし、終了したインプリメンテーション run のインプリメント済みデザインを開きます。

インプリメント済みデザインのデフォルト ビューが表示されます。通常、このビューで配置およびタイミングの解析、フロアプランを実行します。37 ページの図 2-11 に [Implemented Design] のデフォルト ビュー レイアウトを表示します。

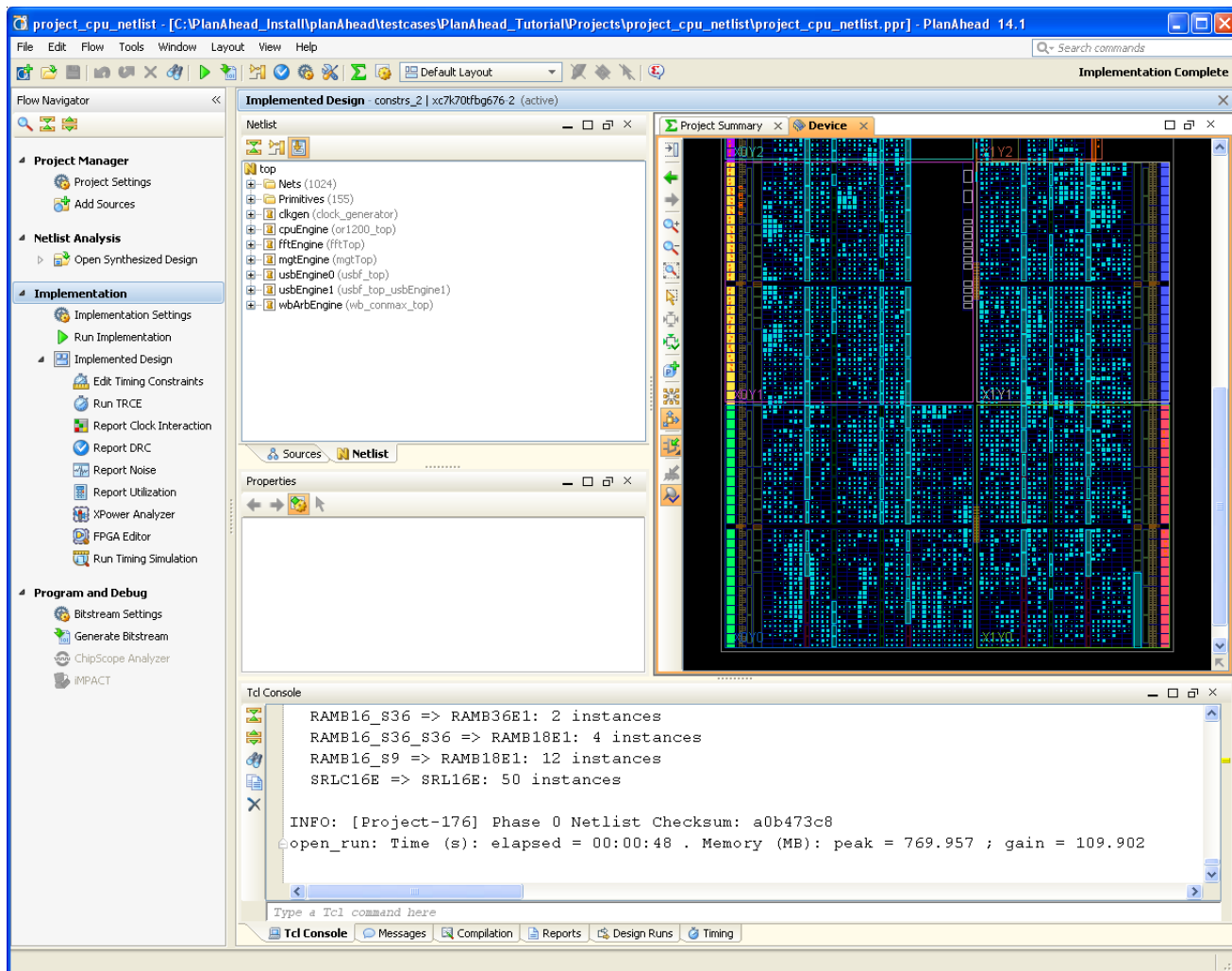


図 2-11：インプリメント済みデザイン

開いているデザインの管理

PlanAhead でデザインを開くと、デザインがメモリに読み込まれ、Flow Navigator で関連するメニューがハイライトされ、視覚的にわかりやすくなります。38 ページの図 2-12 では、[RTL Analysis] メニューがハイライトされています。これは、現在のセッションでエラボレート済みデザインが開いていてアクティブであることを示します。

エラボレート済みデザイン、合成済みデザイン、インプリメント済みデザインは同時に複数開いて、さまざまなデザイン オプションおよび run の結果を表示できます。

複数のデザインを開いている場合、Flow Navigator には開いているデザインに適したメニューが表示されます。図 2-12 の [Synthesis] メニューの下の [Synthesized Design] コマンドは、合成デザインが開いていてもアクティブ デザインではない場合、このようになります。[Synthesized Design] コマンドをクリックすると、開いているデザインがアクティブとなり、ハイライトが [RTL Analysis] から [Synthesis] に移動します。[Implementation] メニューの下の [Open Implemented Design] コマンドは淡色表示になっており、インプリメンテーションが終了しておらず、アクティブなインプリメンテーション run に対して使用可能なインプリメント済みデザインがないことを示しています。

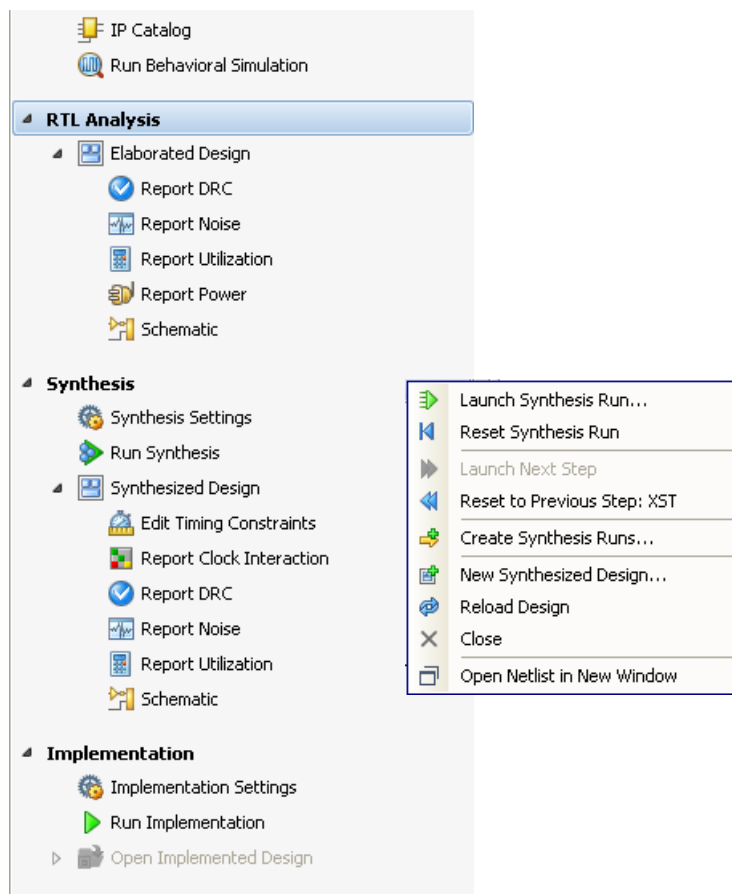


図 2-12 : Flow Navigator からデザインを開く

Flow Navigator では、マウスの右ボタンからのポップアップ メニューもサポートされます。Flow Navigator の [Project Manager]、[RTL Analysis]、[Synthesis]、[Implementation] コマンドを右クリックすると、ポップアップ メニューが表示されます。図 2-12 は、[Synthesis] を右クリックしたところを示しています。

[Open Netlist in New Window] をクリックすると、現在開いているデザインを維持したまま、合成済みデザインを新しい PlanAhead ウィンドウで開くことができます。このコマンドを使用すると、エラボレート済みデザインを表示したまま、別のウィンドウで合成済みデザインまたはインプリメント済みデザインを開くことができます。どちらのウィンドウでも拡大/縮小、表示エリアを変更、デザイン エレメントを選択して、解析および開発が可能です。

デザイン ビュー パナーの使用

デザイン ビュー パナーには、現在のデザイン内容を反映した、制約セット、ターゲット パーツ、合成またはインプリメンテーション run が表示されます。39 ページの図 2-13 は、さまざまなデザイン タイプのデザイン ビュー パナーを示しています。



図 2-13 : デザイン ビュー バナー

複数のデザインが開いている場合は、図 2-14 のようなデザイン ビュー バナーにタブが表示され、デザイン間の表示を切り替えられるようになります。

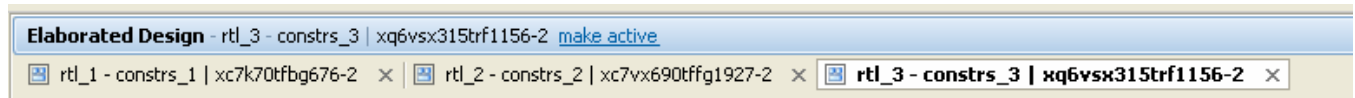


図 2-14 : 複数のデザインのタブ

デザインのアップデートおよび再読み込み

設計プロセスでは、通常ソース ファイルまたは制約の変更が必要です。これらのファイルの依存性は PlanAhead で管理され、表示されているデザイン データの再読み込みが必要になるとそれが示されます。ターゲット パーツやアクティブ制約セットなどのプロジェクト設定を変更すると、デザインが最新の状態でなくなります。

開いているデザインのターゲット パーツまたは制約セットがアクティブな合成またはインプリメンテーション run のターゲット パーツまたは制約セットと異なる場合は、デザイン ビュー バナーに図 2-14 に示すような [make active] リンクが表示されます。[make active] をクリックすると、デザインの制約セットとターゲット パーツが現在のプロジェクトのアクティブな制約セットおよびターゲット パーツと、アクティブな合成およびインプリメンテーション run に設定されます。

ソース ファイル、ネットリスト、またはインプリメンテーション結果がアップデートされると、開いた合成済みデザインまたはインプリメント済みデザインのデザイン ビュー バナーの右側に「out-of-date」というメッセージが表示され、run が最新の状態ではないことが示されます (図 2-15)。デザインのどの部分が最新ではないのかを確認するには、[more info] リンクをクリックします。

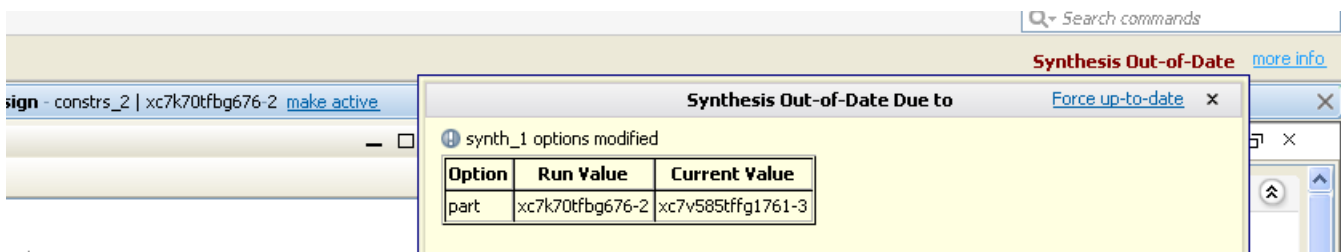


図 2-15 : デザインの再読み込みが必要であることを示すバナー

デザインの再読み込みが必要な場合は、次のいずれかを実行してください。

- [Force up-to-date] : 図 2-15 に示す [more info] をクリックすると開く [Out-of-Date Due to] ウィンドウに表示されるリンクです。

アクティブな合成またはインプリメンテーション run の NEEDS_REFRESH プロパティを必要に応じてリセットして、run を最新の状態にします。このコマンドを Tcl で実行する場合は、次のようなコードになります。

```
set_property needs_refresh false [get_runs synth_2]
```

注記：このコマンドは、デザインに少しだけ変更を加えたため、デザインをリフレッシュしたくない場合に使用します。

- [Reset and Rerun] : [Design Runs] ビューから [Reset Runs] コマンドをクリックするか、Flow Navigator の [Synthesis] または [Implementation] コマンドを右クリックして run をリセットするコマンドを選択します。これにより、run がリセットされ、現在のプロジェクト設定で再実行およびアップデートできます。

[Launch Runs] コマンドを直接使用することもできます。この場合、リセットをまず実行する必要があることを示すメッセージが表示されます。

- [Reload Design] : 図 2-16 に示すような Flow Navigator のポップアップ メニューから [Reload Design] コマンドを使用すると、変更前の状態に戻して、元のネットリスト、制約、ターゲット パーツを読み込み直します。

現在のデザインのメモリ内の表示をリフレッシュし、保存されていないデザイン データへの変更を削除します。

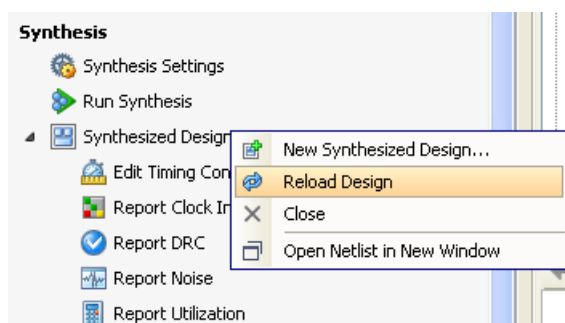


図 2-16 : [Reload Design] コマンド

デザインの保存

デザイン データへの変更を保存するには、[File] → [Save Design] をクリックします。



このコマンドでは、合成済みデザインへ追加された制約ファイル、デザイン パーティション、ChipScope コア、コンフィギュレーションへの変更およびプロジェクト設定の変更が保存されます。


インプリメント済みデザインが開いている場合、制約の変更は現在アクティブな制約ファイルではなく、インプリメンテーションで使用されたインプリメンテーション run に関連する制約ファイルに保存されます。これにより、インプリメント済みデザインへの変更がインプリメンテーション run の制約ファイルに保存されます。

ただし、特定のインプリメンテーション run に使用した制約ファイルに意図していない変更が加えられる可能性があります。PlanAhead では、この状況を警告するメッセージが表示され、ディスクに変更を保存する前に保存する制約ファイルを選択できます。

デザインへの変更は新しい制約ファイルに保存して、元のデザインはそのままにしておくこともできます。新しい制約ファイルを作成するには、[File] → [Save Design As] をクリックします。

デザインを閉じる

デザインを閉じると、メモリ内のデザイン数を減らし、ソースを編集できる場所が複数になるのを避けることができます。別のデザイン表示に移動する前に、デザインを閉じるようダイアログ ボックスが表示されることがあります。パーシャル リコンフィギュレーション デザインなどの場合は、別の表示に移動する前にデザインを閉じる必要があります。デザインは、次のように閉じることができます。

- 個々のデザインを閉じるには、メイン表示エリアのバナーにある [Close] ボタン をクリックします。
- すべてのデザインを閉じるには、Flow Navigator で右クリックして [Close All] をクリックします。

プロジェクトの操作

この章では、次の内容について説明します。

- 「プロジェクト タイプ」
- 「新規プロジェクトの作成」
- 「プロジェクトの管理」
- 「プロジェクト ソースの管理」
- 「デザイン ソース ファイルの管理」
- 「制約の管理」
- 「シミュレーション ソースの管理」
- 「IP コアの管理」
- 「DSP ソースの管理」
- 「エンベデッド プロセッサ ソースの管理」
- 「[Project Summary] ビュー」
- 「プロジェクト設定」

プロジェクト タイプ

PlanAhead™ は、FPGA デザイン フローのさまざまな段階で使用できます。次のプロジェクト タイプを作成できます。

- レジスタ トランスファー レベル (RTL) ソース ベースのプロジェクト
- 合成済みネットリスト ベースのプロジェクト
- インプリメント済みデザイン結果ベースのプロジェクト
- CSV (Comma Separated Values)、ユーザー制約ファイル (UCF) ベースの I/O ピン配置プロジェクト
- ISE® Project Navigator で作成されたプロジェクト
- パーシャル リコンフィギュレーション プロジェクト (ライセンスがイネーブルの場合)

プロジェクトは、プロジェクト作成時に使用された入力ソースのタイプによって識別されます。プロジェクト タイプは新規プロジェクトの作成時に選択できます。

I/O ピン配置プロジェクト以外、作成後にプロジェクト タイプを変更することはできません。I/O ピン配置プロジェクトは、RTL ベースのデザイン プロジェクトの基盤として使用し、後でほかのプロジェクト タイプへ移行できます。

RTL ソース ベースのプロジェクト

PlanAhead を使用すると、RTL の作成からビットストリームの生成まで、FPGA デザイン フロー全体を管理できます。RTL ソース ファイル、デザイン ブロックの EDIF ネットリストだけでなく、CORE Generator™ で生成された IP およびコンパイル済みの NGC/NGO 形式の IP ネットリストをプロジェクトに追加できます。

RTL をエラボレートして解析し、構文が正しいことを確認し、さまざまな合成やインプリメンテーション run を実行および管理し、デザインと実行結果を解析できます。また、さまざまな制約やインプリメンテーション ストラテジを試すこともできます。

合成済みネットリスト ベースのプロジェクト

Xilinx® Synthesis Technology (XST) やサポートされているサードパーティの合成ツールを使用して PlanAhead 環境外で合成されたデザインから、プロジェクトを作成することもできます。PlanAhead では、EDIF または NGC/NGO フォーマットのネットリストをインポートできます。ネットリストは、1 つのファイルにまとめられているか、複数のモジュール レベルのネットリストから構成される階層構造になっているものを使用できます。

ロジック ネットリストを解析し、さまざまなインプリメンテーション run を実行および管理し、デザインと実行結果を解析できます。また、さまざまな制約やインプリメンテーション ストラテジを試すこともできます。

インプリメント済みデザイン結果ベースのプロジェクト

ザイリンクスのコマンド ライン ツールを使用し、PlanAhead 環境外で作成されたインプリメンテーション結果を解析するプロジェクトも作成できます。デザイン ネットリスト、インプリメンテーション、タイミング結果をインポートし、タイミングまたは配置に関する問題を調べることができます。

I/O ピン配置プロジェクト

空の I/O ピン配置プロジェクトを作成して、デザイン サイクルの初期段階で I/O ピン配置を実行できます。I/O ポートは PlanAhead 内で作成したり、CSV またはユーザー制約ファイル (UCF) のいずれかの形式でインポートできます。ピン配置プロジェクトを使用すると、異なるデバイスアーキテクチャで使用可能なロジック リソースも確認できます。

I/O ピンを割り当てた後、PlanAhead で CSV、UCF、および RTL 出力ファイルを作成できます。このファイルは、RTL ソースまたはネットリストが使用可能になったデザイン フローの後の段階で使用します。この出力ファイルは、プリント回路基板 (PCB) デザイン プロセスで使用する回路図シンボルの作成にも使用できます。

I/O ピン配置プロジェクトは、RTL ベースのデザイン プロジェクトの基盤として使用できます。詳細は、[314 ページの「RTL デザインへの変換」](#)を参照してください。

外部で作成したプロジェクト


ISE Project Navigator、XST、または Synopsys® Synplify® からのプロジェクト データを PlanAhead にインポートして、RTL レベルのプロジェクトへ変換することができます。ソース プロジェクトからのさまざまなプロジェクト設定を使用して、新規プロジェクトが作成されます。

注記：PlanAhead を Project Navigator から起動して、ISE デザインの I/O ピン配置およびフロアプランを実行することも可能です。ISE Project Navigator 環境からの PlanAhead ツールの使用に関する詳細は、第 15 章「Project Navigator からの PlanAhead の使用」を参照してください。

新規プロジェクトの作成

New Project ウィザードでは、プロジェクト名およびディレクトリの指定、プロジェクトへのソースファイルと制約ファイルの追加、ターゲット デバイスの選択をウィザードに従って実行できます。

新規プロジェクトを作成するには、次の手順に従います。

1. 次のいずれかを選択します。
 - Getting Started ページの [Create a New Project] リンクをクリックします。
 - [File] → [New Project] をクリックするか、ツールバーの [New Project] ボタンをクリックします。 

ウィザードの最初のページにウィザードの概要が表示されます。[Next] をクリックします。

2. [Next] をクリックします。

図 3-1 に示す [Project Name] ページが表示されます。

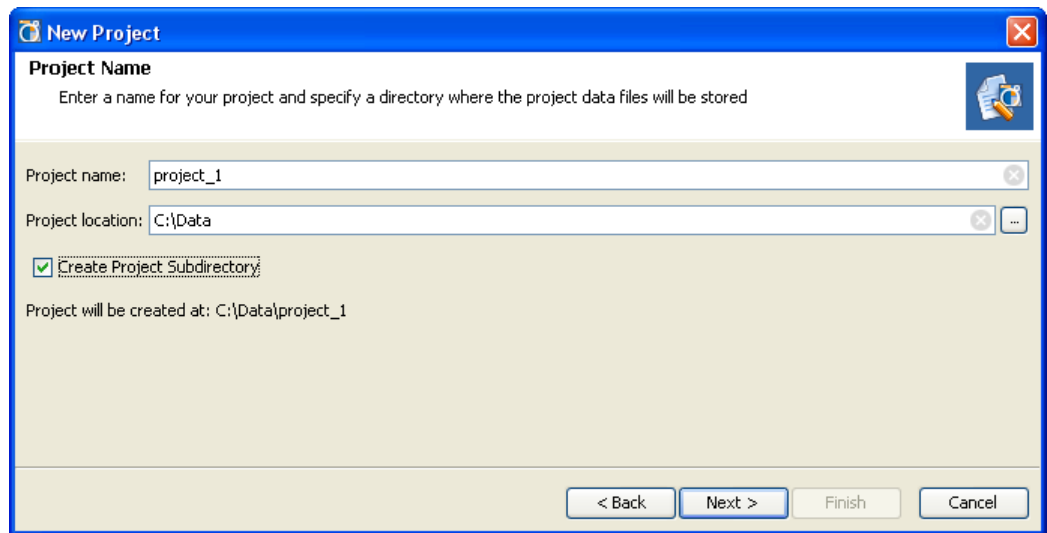


図 3-1 : New Project ウィザード : [Project Name] ページ

3. [Project Name] ページで [Project name] および [Project location] を指定します。
 - [Project name] : project_3 など、プロジェクト ディレクトリを識別する名前を入力します。
 - [Project location] : プロジェクト ディレクトリを作成するディレクトリを入力します。
 - [Create Project Subdirectory] : 指定したプロジェクト ディレクトリにプロジェクトと同じ名前の下位ディレクトリを作成するかどうか指定します。

オンにすると (デフォルト)、プロジェクト ファイル (.ppr) が `<project_location>/<project_name>` に作成され、そのプロジェクトで作成されたすべてのフォルダーおよびデータ ファイルが `<project_name>` の下位ディレクトリに保存されます。

オフにすると、プロジェクト ファイル (.ppr) が *<project_location>* に作成され、そのプロジェクトで作成されたすべてのフォルダーおよびデータ ファイルがそのディレクトリに保存されます。

4. [Next] をクリックします。

プロジェクト タイプの選択

46 ページの図 3-2 に示すページで、プロジェクト タイプを選択します。これにより、プロジェクトに関連付けるソース ファイルのタイプが決まります。

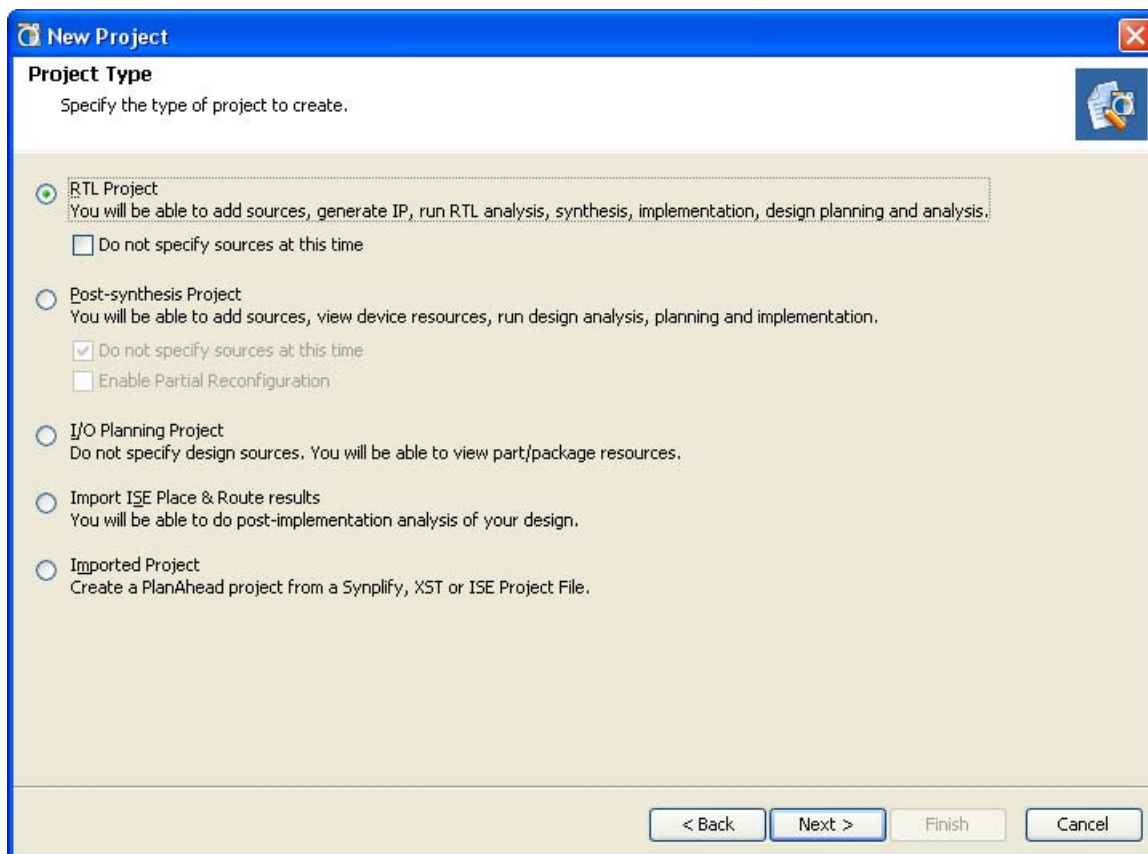


図 3-2 : New Project ウィザード : プロジェクト タイプ

5. デザイン ソースのフォーマットを選択し、[Next] をクリックします。
6. 作成するプロジェクト タイプに応じて、次のいずれかの説明を参照してください。
 - [RTL プロジェクトの作成](#)
 - [合成後プロジェクトの作成](#)
 - [I/O ピン配置プロジェクトの作成](#)
 - [ISE の配置およびタイミング結果を使用したプロジェクトの作成](#)
 - [「外部プロジェクトのインポート」](#)

ウィザードの次のページでは、前のページで選択したプロジェクト タイプに基づいたプロジェクトに最適なソースを追加します。

RTL プロジェクトの作成

RTL ソース ファイルをインポートしてプロジェクトを作成できます。このプロジェクトは、RTL コードの開発と解析、合成とインプリメンテーションを目的とする場合に使用します。RTL の開発および解析の詳細は、第 5 章「エラボレート済み RTL デザイン」を参照してください。

- 45 ページの「新規プロジェクトの作成」の手順に従ってプロジェクトを作成します。
- 図 3-2 に示す [Project Type] ページで [RTL Project] をオンにします。
 - 必要であれば、[Do not specify sources at this time] チェック ボックスをオンにし、53 ページの「デフォルト パーツまたはボードの選択」へ進みます。これをオンにすると、ソース ファイルおよび制約を追加する手順を飛ばして、ターゲット パーツを選択してプロジェクトを作成できます。
- [Next] をクリックします。

ソース ファイルまたはディレクトリの追加

[Add Sources] ページで、ソース ファイルおよびソース ファイルを含むディレクトリを追加します。図 3-3 に示す [Add Files] および [Add Directories] ボタンをクリックします。

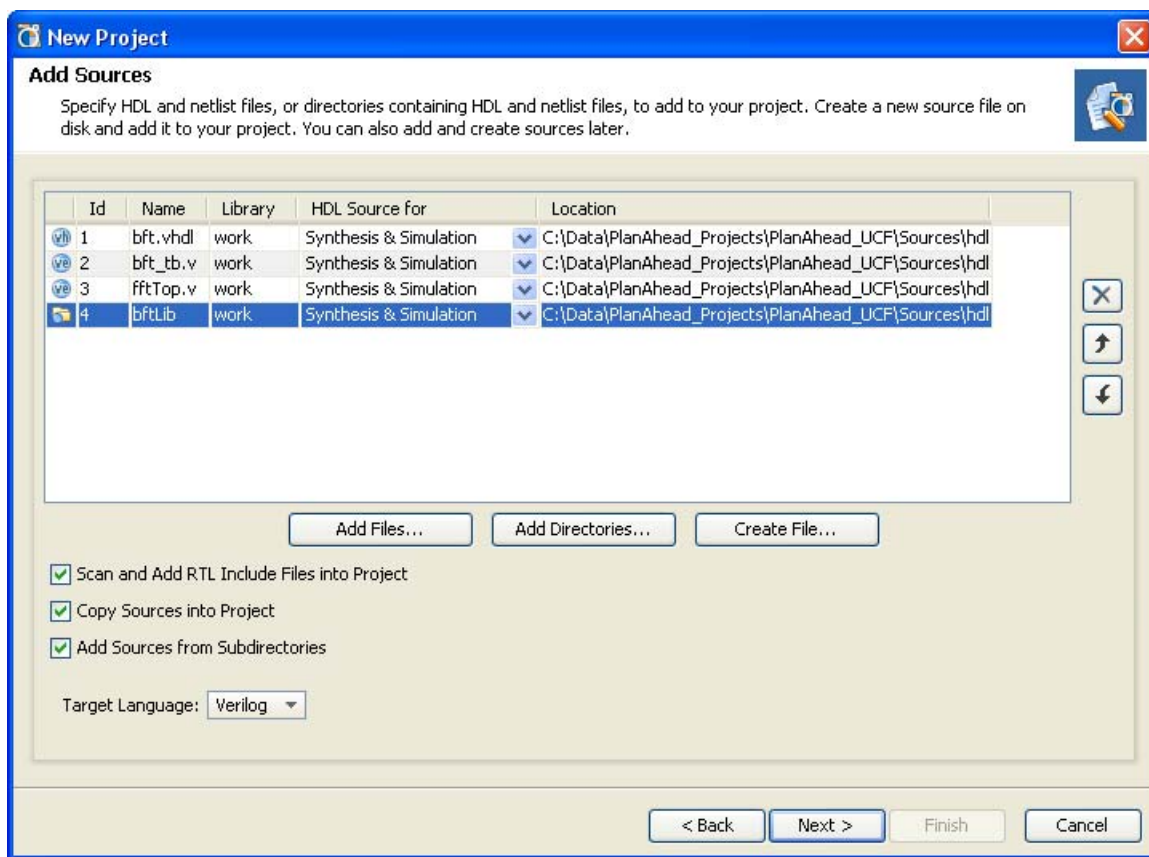





図 3-3 : New Project ウィザード : ソースの追加

[Add or Create Design Sources] ページでは、次のコマンドおよびオプションを使用できます。

- [Add Files] : プロジェクトに追加するファイルを選択します。RTL プロジェクトに追加できるファイル タイプは、HDL ファイル、EDIF および NGC ファイル、BMM、ELF およびその他のファイルです。

	Id	Name
	1	Xilinx_1_ch_test
	2	prml_core
	3	zeus_pll.v

注記 : 各ファイルまたはディレクトリは、それがファイルであるのかフォルダーであるのかを示すアイコンで表示されます。小さい赤い四角は、読み出し専用であることを示します。

- [Add Directories] : 選択したディレクトリに含まれるすべてのファイルを追加します。指定したディレクトリにある有効なソース ファイルがすべてプロジェクトに追加されます。
 - [Library] : ファイルまたはディレクトリの RTL ライブラリを指定します。定義済みのライブラリ名から選択するか、新規ライブラリ名を入力します。
work ライブラリが、すべての HDL ソース ファイルを配置するデフォルトのライブラリです。必要に応じて、ユーザー VHDL ライブラリを作成し、参照できます。
 - [HDL Source for] : 読み込むソースが合成およびシミュレーション用の RTL ソース ファイルであるか、シミュレーションのみで使用する RTL テストベンチであるかを指定します。
- [Create File] : Verilog、Verilog ヘッダー、または VHDL ソース ファイルを作成する [Create Source File] ダイアログ ボックスが開きます。詳細は、[49 ページの「新規モジュールの定義」](#)を参照してください。
- [Remove selected files and directories] : 選択したソース ファイルまたはディレクトリを削除します。
- [Move selected files and directories up] : ファイルまたはディレクトリをリストの上方向に移動します。ファイル順は、合成やシミュレーションなどのダウンストリーム プロセスでのエラボーレーションおよびコンパイルの順序に影響します。詳細は、[第 6 章の「ファイルのコンパイル順」](#)を参照してください。
- [Move selected files and directories down] : ファイルまたはディレクトリをリストの下方向に移動します。
- [Scan and Add RTL Include Files into Project] : すべての RTL ファイルをスキャンし、参照された Verilog の 'include ファイルをローカルプロジェクト ディレクトリにインポートします。
- [Copy Sources into Project] : ソース ファイルをプロジェクト ディレクトリにコピーします。プロジェクトではローカルにコピーされたバージョンが使用されます。[Add Directories] ボタンをクリックしてソース ファイルのディレクトリを追加した場合は、ファイルがローカルのプロジェクトにコピーされる際にディレクトリ構造もそのまま保持されます。これについては、[72 ページの「リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー」](#)を参照してください。
- [Add Sources from Subdirectories] : [Add Directories] で指定したディレクトリのサブディレクトリに含まれるソース ファイルをすべて追加します。
- [Target Language] : Verilog または VHDL のいずれかにデザインのターゲット言語を指定します。新しい RTF ファイルはデフォルトで指定したターゲット言語になります。指定したターゲット言語でデザインの出力ファイルが生成されます。

RTL ソースの作成

新規 ソース ファイルを作成するには、[Add Sources] コマンドをクリックして [Add or Create Design Sources] をオンにし、[Create File] をクリックします。[図 3-4](#) のような [Create Source File] ダイアログ ボックスが開き、作成するソース ファイルのタイプ、名前、および場所を指定できます。

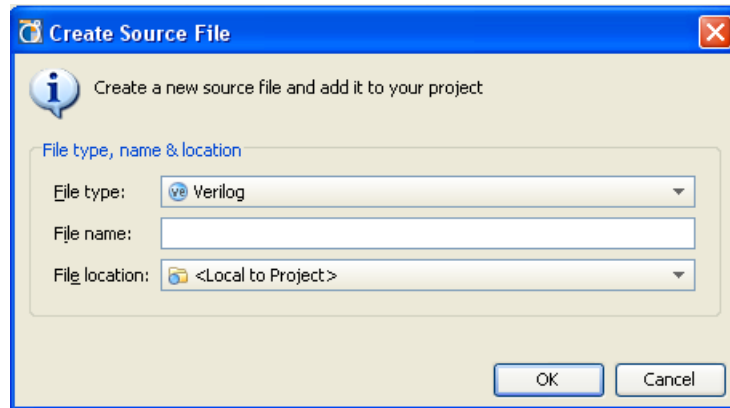


図 3-4 : [Create Source File] ダイアログ ボックス

1. [Create Source File] ダイアログ ボックスでは、次の情報を定義できます。
 - [File type] : 次のファイル タイプのいずれかを選択します。
 - [Verilog] : Verilog ファイル (.v) を作成します。
 - [Verilog Header] : Verilog ヘッダー ファイル (.vh) を作成します。
 - [VHDL] : VHDL ファイル (.vhd1) を作成します。
 - [File name] : 作成する HDL ソース ファイルの名前を入力します。
 - [File location] : ファイルを作成する場所を指定します。
2. [OK] をクリックします。

ファイルのプレースホルダーがソースのリストに追加されます。ファイルは [Finish] をクリックすると作成されます。
3. [Create File] コマンドを何度か繰り返して、プロジェクトに追加するモジュール数を定義します。
4. [Add Sources] ページでソース ファイルに最適なライブラリを指定します。デフォルトでは、ソース ファイルは work ライブラリに追加されます。
5. [Finish] をクリックすると、指定したソースがプロジェクトに追加されます。
 - [Create Files] を使用すると、49 ページの「新規モジュールの定義」で説明されるような [Define Modules] ダイアログ ボックスが開きます。
 - 新規モジュールを定義しない場合は、51 ページの「IP の追加」に進みます。

新規モジュールの定義

プロジェクトに追加する新規 RTL ソース ファイルを指定したら、Verilog または VHDL コードでモジュールまたはアーキテクチャを定義する必要があります。図 3-5 に示すような [Define Modules] ダイアログ ボックスを使用すると、新規 RTL コードを作成できます。

注記 : [Define Module] ダイアログ ボックスは、プロジェクトに対するソースや制約ファイルを定義し、ターゲット パーツを選択し、プロジェクト作成を終了した場合に開きます。

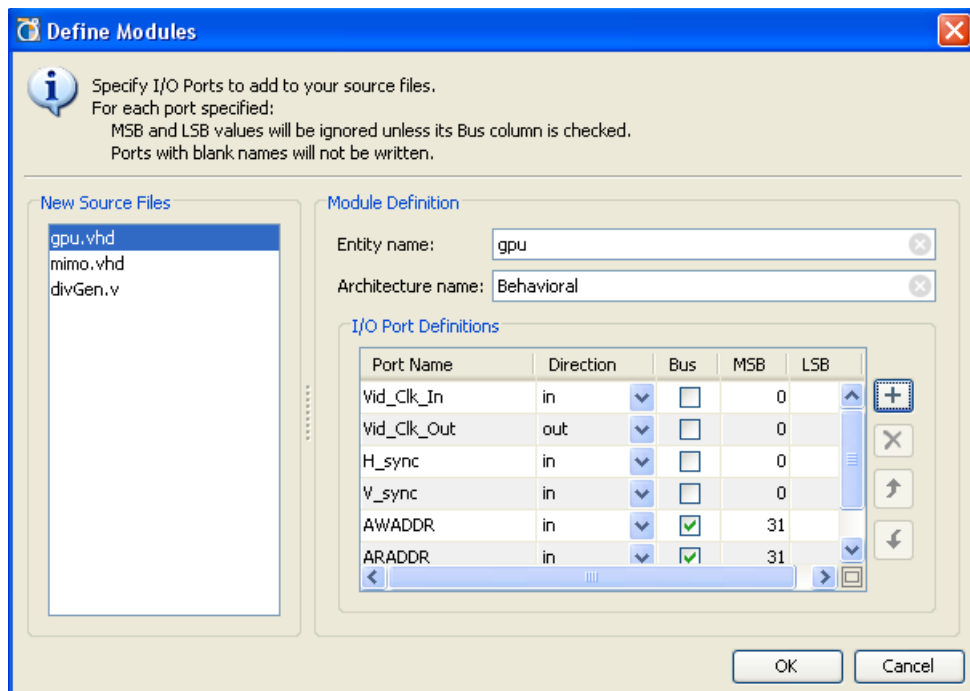


図 3-5 : [Define Modules] ダイアログ ボックス

1. [Define Modules] ダイアログ ボックスには、48 ページの「RTL ソースの作成」で説明された [Create File] コマンドで定義された新規ソース ファイルが表示されます。ここでは、Verilog および VHDL モジュールの両方が定義できます。
2. まず、[New Source Files] のリストから定義するモジュールを選択します。
3. [Define Modules] ダイアログ ボックスには、RTL コード記述を簡単にするために、次の情報を指定できます。

- [Entity name/Module name] : VHDL コードのエンティティまたは Verilog コードのモジュール名の名前を指定します。エンティティまたはモジュール名はデフォルトでそのファイル名になりますが、別の名前を付けることもできます。
- [Architecture name] : RTL ソース ファイルのアーキテクチャを指定します。デフォルトでは [Behavioral] です。

注記 : このフィールドは、VHDL コードの場合にのみ表示され、Verilog モジュールを定義する場合には表示されません。

- [I/O Port Definitions] : モジュール定義に追加するポートを定義します。
 - [Port Name] : RTL コードに記述されるポートの名前を定義します。
 - [Direction] : ポートを入力、出力、双方向のいずれかに指定します。
 - [Bus] : バス ポートであるかどうかを指定します。次の [MSB] および [LSB] フィールドを使用してポートのバス幅を定義します。
 - [MSB] : 最上位ビットの数を定義します。[LSB] フィールドと組み合わせて、定義されるバスの幅を指定します。
 - [LSB] : 最下位ビットの数を定義します。

注記 : ポートがバス ポートでない場合は、MSB および LSB は無視されます。

4. [New Source Files] の下の各新規モジュールの詳細を定義し終わったら、[OK] をクリックして RTL ソース ファイルを作成して、モジュールをプロジェクトに追加します。[Sources] ビューに新しく定義したモジュールがリストされます。
5. この新規ソース ファイルをテキスト エディターで開いて、必要に応じて編集します。テキスト エディターでファイルを開いて編集するには、ファイルをダブルクリックするか、[Open File] ポップアップ メニューをクリックします。新規作成したファイルの編集方法については、[第 4 章の「テキスト エディターの使用」](#)を参照してください。

IP の追加

プロジェクトの RTL ソース ファイルを指定したら、PlanAhead 環境外で作成した既存の CORE Generator コア ファイル (.xco) を使用して、既存の IP コアを追加できます。XCO ファイルには、コアが作成されたときに使用されたカスタマイズ パラメーターとプロジェクト オプションが記録されています。PlanAhead で、XCO ファイルを使用して新規プロジェクト内にコアが再生成されます。

注記：XCO コア ファイルは RTL プロジェクトにのみ追加できます。

51 ページの [図 3-6](#) に、プロジェクトに追加された IP コアを示しています。

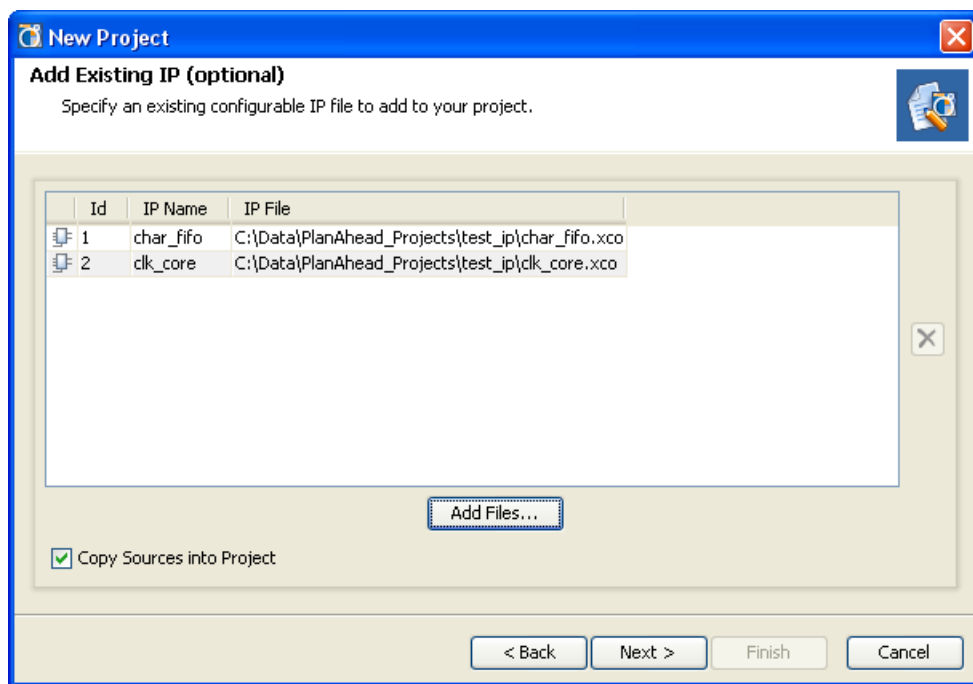


図 3-6 : New Project ウィザード : [Add Existing IP] ダイアログ ボックス

1. プロジェクトに追加する CORE Generator コア ファイル (.xco) を探すには、[Add Files] ボタンを使用します。
2. 制約ファイルを元のロケーションから参照するのではなく、ローカルプロジェクト ディレクトリにコピーする場合は、[Copy Sources into Project] をオンにします。
3. [Next] をクリックします。

パラメーター指定可能なコアは、PlanAhead で [IP Catalog] コマンドを使用して CORE Generator を起動することによりプロジェクトへ読み込むこともできます。詳細は、[75 ページの「IP コアの管理」](#)を参照してください。

サードパーティから合成済み NGC または EDIF ネットリストとして提供されている IP もあります。これらのファイルをデザインに読み込むには、[Add Sources] コマンドをクリックし、[Add or Create Design Sources] をオンにしてファイルを読み込みます。

制約の追加

52 ページの図 3-7 に示す [Add Constraints] ページでは、最上位 UCF ファイルまたはモジュールレベルのネットリスト制約ファイル (NCF) を追加できます。

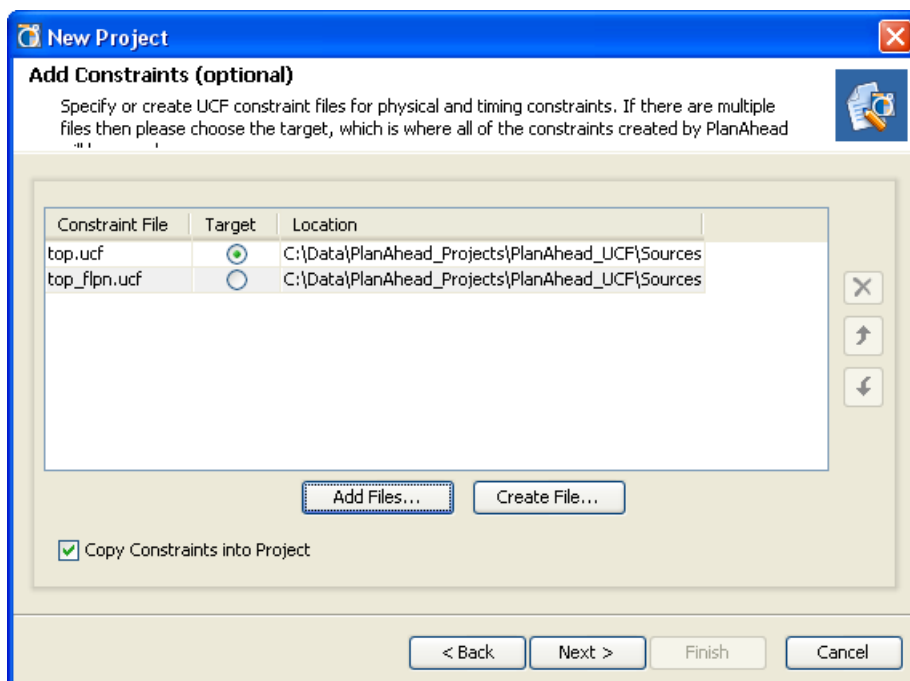


図 3-7 : [Create Constraint Set Name] ダイアログ ボックス

必要に応じてプロジェクトの制約ファイルを追加または作成できます。この手順の詳細は、62 ページの「制約の管理」を参照してください。

RTL またはネットリスト ソース ファイルと同じディレクトリに UCF、NCF、または XST 制約ファイル (XCF) が存在する場合、デフォルトでそれらのファイルがプロジェクトに追加される制約ファイルとして表示されます。

[Add Constraints] ページには、次のオプションがあります。

- [Add Files] : プロジェクトに追加する UCF、NCF、または XCF ファイルを選択します。
- [Create File] : プロジェクトの最上位 UCF ファイルを作成します。
- [Remove selected files and directories] : 制約ファイルのリストで選択している UCF ファイルを削除します。
- [Move selected files and directories up]/[Move selected files and directories down] : 選択した UCF ファイルをリストの上方向または下方向に移動します。制約は順序に依存し、最後に読み込まれた制約でそれ以前に読み込まれた制約が置き換えられます。
- [Copy Constraints into Project] : 制約ファイルを元のロケーションから参照するのではなく、プロジェクト ディレクトリにコピーします。

制約ファイルを追加し、ターゲット UCF を指定したら (オプション)、[Next] をクリックします。

デフォルト パーツまたはボードの選択

New Project では、プロジェクト パーツまたはターゲット デザイン プラットフォーム (TDP) を選択する 53 ページの図 3-8 のようなページが表示されます。

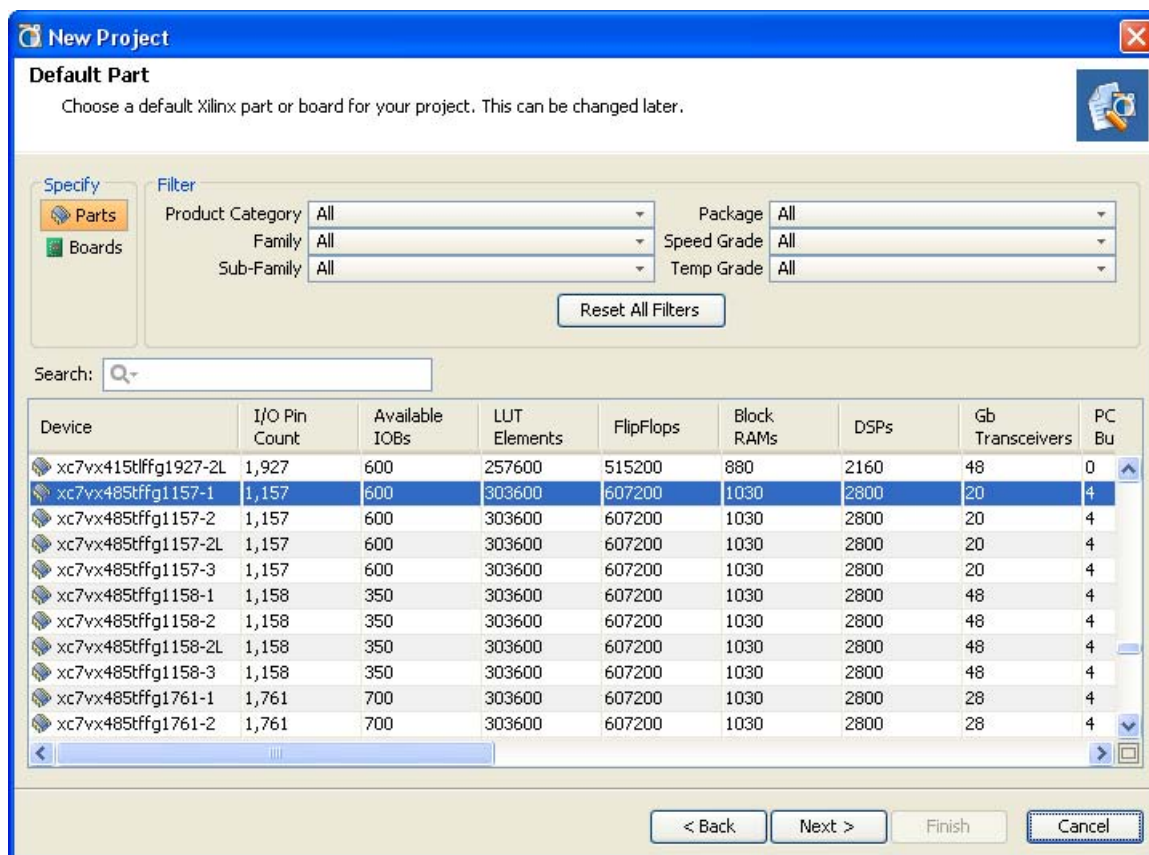


図 3-8 : New Project ウィザード : [Default Part] ページ

[Default Part] ページでは、ターゲット ザイリンクス デバイスを指定するか、デザインのターゲットとして TDP ボードを指定できます。

- [Parts]: 選択可能なデバイスがリストされます。デバイス リソースに関する情報が、表形式で表示されます。ページの上部にある製品 ([Product])、ファミリー ([Family])、サブファミリー ([Sub-Family])、パッケージ ([Package])、スピード グレード ([Speed Grade])、および温度 ([Temp Grade]) フィルターを使用して、リストされるデバイスを絞り込むことができます。

注記: Zynq™ デバイスには、読み出し専用のプロセッサ サブシステム IO ピン (PSSIO) が含まれます。これらのピンは、Zynq デバイス上ではプログラマブル ロジック (PL) ピンとしてコンフィギュレーションできませんが、プロセッサ システム (PS) システムとしてはコンフィギュレーション可能で、XPS からインポートできます。詳細は、[XPS ヘルプ](#)を参照してください。

- [Boards]: 使用可能な TDP ボードおよびボードで使用されるザイリンクス パーツがリストされます。I/O ピンのカウントや LUT および FF の数、使用可能なブロック RAM などのデバイス リソースに関する情報が表形式で表示されます。リストは、デバイス ファミリー、パッケージ、スピード グレードでフィルターをかけて表示させることもできます。

また、[Search] ボックスに検索条件を入力することもできます。

1. リストからターゲット パーツまたはボードを選択して、[Next] をクリックします。

注記：ターゲット パーツは、合成およびインプリメンテーション中にエラボレート済みまたは合成済みデザインを開くと変更できます。詳細は、[101 ページの「プロジェクト設定」](#)を参照してください。

[New Project Summary] ページが表示されます。

2. プロジェクトを作成するには、このサマリ ページで [Finish] をクリックします。

プロジェクト環境に Project Manager に関連するビューが表示されます。

合成後プロジェクトの作成

合成済みネットリストおよびその制約を使用して PlanAhead プロジェクトを作成できます。このプロジェクトは、フロアプラン環境およびインプリメンテーション環境を使用してデザインを解析、フロアプラン、およびインプリメントする場合に使用します。

1. [45 ページの「新規プロジェクトの作成」](#)の手順に従ってプロジェクトを作成します。

2. [Project Type] ページで [Post-synthesis Project] をオンにします。

- a. 必要であれば、[Do not specify sources at this time] チェック ボックスをオンにし、[53 ページの「デフォルト パーツまたはボードの選択」](#)へ進みます。これをオンにすると、ソース ファイルおよび制約を追加する手順を飛ばして、ターゲット パーツを選択してプロジェクトを作成できます。
- b. [Enable Partial Reconfiguration] チェック ボックスをオンにすると、プロジェクトをパーシャル リコンフィギュレーション プロジェクトとして定義できます。

注記：PlanAhead では、パーシャル リコンフィギュレーション デザインをサポートする特定タイプのプロジェクトが使用されます。この機能は特別ライセンスでのみ提供されています。この機能については、[付録 E 「その他のリソース」](#)に示す『パーシャル リコンフィギュレーション ユーザー ガイド』(UG702) を参照してください。

3. [Next] をクリックします。

最上位ネットリストおよびモジュール検索パスの指定

[図 3-9](#) に示す New Project ウィザードの次のページでは、読み込むネットリスト ファイルを指定し、最上位モジュールを含むファイルを指定し、下位モジュールのネットリスト ファイルを検索するディレクトリを定義します。

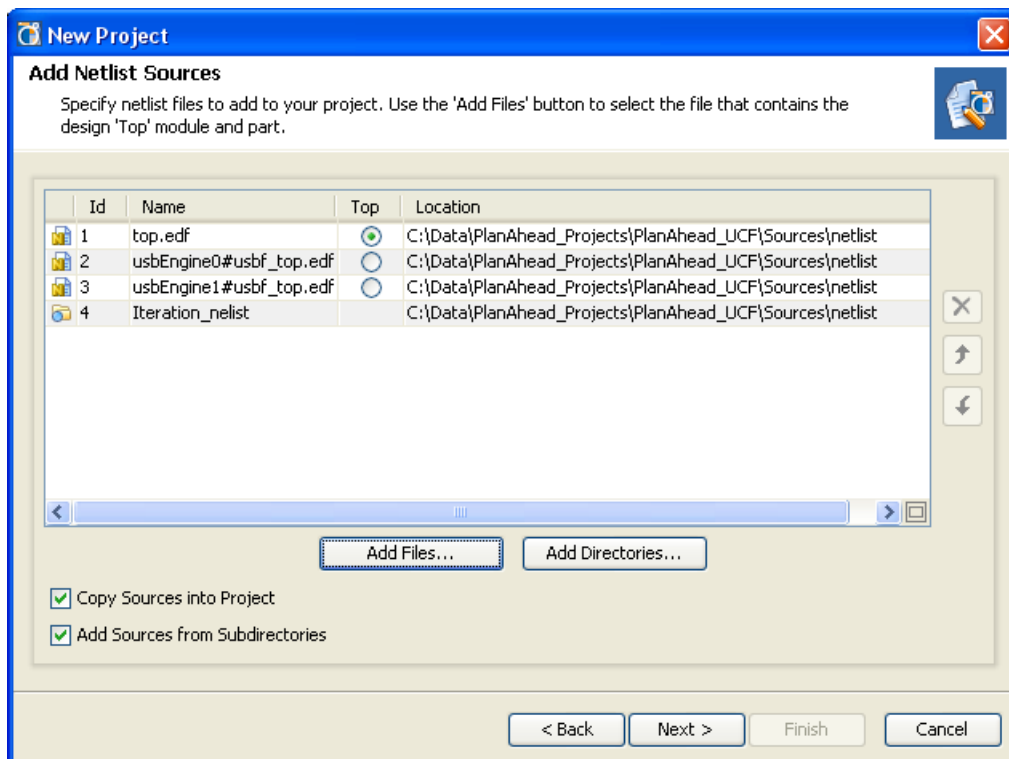


図 3-9 : New Project ウィザード : ネットリスト ファイルの指定

1. 次のオプションを指定します。
 - [Add Files] : プロジェクトに追加するネットリスト ファイル (NGC または EDIF) を選択します。
 - [Top] : ファイルに最上位ネットリストが含まれる場合はオンにします。
 - [Add Directories] : モジュールおよびコアを検索するディレクトリを選択します。
 - [Remove Selected Files and Directories] : X アイコンで表示され、選択したファイルおよびディレクトリを削除します。
 - [Move Selected Files and Directories Up] : 上向き矢印のアイコンで、ファイルまたはディレクトリをリストの上方向に移動します。
 - [Move Selected Files and Directories Down] : 下向き矢印のアイコンで、ファイルまたはディレクトリをリストの下方向に移動します。
 - [Copy Sources into Project] : ソース ファイルをプロジェクト ディレクトリにコピーします。プロジェクトではローカルにコピーされたバージョンが使用されます。[Add Directories] ボタンをクリックしてソース ファイルのディレクトリを追加した場合は、ファイルがローカルプロジェクトにコピーされる際にディレクトリ構造もそのまま保持されます。これについては、72 ページの「リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー」を参照してください。
 - [Add Sources from Subdirectories] : [Add Directories] で指定したディレクトリの下位ディレクトリに含まれるソース ファイルをすべて追加します。
2. [Next] をクリックします。
3. 62 ページの「制約ファイルの追加と作成」の手順に従って制約ファイルを追加します。

4. 53 ページの「デフォルト パーツまたはボードの選択」の手順に従ってプロジェクト デバイスを選択します。

[New Project Summary] ページにプロジェクトに選択されたオプションが表示されます。

5. [Finish] をクリックし、プロジェクトを作成して開きます。

I/O ピン配置プロジェクトの作成

システム レベル デザインのデバイス ピン配置を指定するために使用する I/O ピン配置プロジェクトを作成できます。このプロジェクトは、HDL または合成済み EDIF を完了する前に作成します。システム レベルまたは PCB 設計者とデザイン情報を共有する目的などに使用できます。詳細は、次を参照してください。

I/O ピン配置 : 第 8 章「I/O ピン配置」

[Project Type] ページで [Create an I/O Planning Project] をオンにします。

I/O ポートのインポート

56 ページの図 3-10 に示す [Import Ports] ページでは、インポートする I/O ポートおよび制約ファイルを選択します。

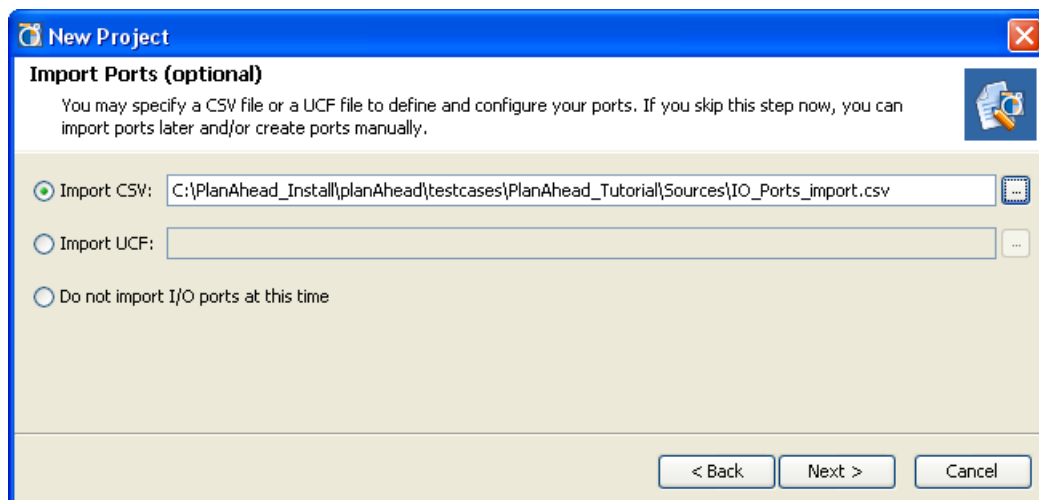


図 3-10 : New Project ウィザード : [Import Ports] ページ

1. 次のオプションを設定します。
 - [Import CSV] : PlanAhead のファイル形式で定義された I/O ポート定義を含む CSV ファイルを選択します。このファイルの仕様は、450 ページの「CSV ファイル形式の I/O ポート リスト」を参照してください。
 - [Import UCF] : I/O ポート関連の制約のみを含む UCF ファイルを選択します。
 - [Do not import I/O ports at this time] : 空のプロジェクトを作成します。I/O は後ほど作成またはインポートできます。

注記 : RTL ヘッダーまたはソース ファイルを使用してデザインの I/O ピン配置を実行するには、RTL ソース プロジェクトを使用します。

2. [Next] をクリックします。

3. 53 ページの「デフォルト パーツまたはボードの選択」の手順に従ってプロジェクト デバイスを選択します。

[New Project Summary] ページにプロジェクトに選択されたオプションが表示されます。

4. [Finish] をクリックし、プロジェクトを作成して開きます。

ISE の配置およびタイミング結果を使用したプロジェクトの作成

ISE® からの配置配線結果、制約、プロジェクト設定をインポートしてプロジェクトを作成できます。このプロジェクトは、PlanAhead のインプリメンテーションおよび解析環境を使用して配置配線結果を解析するのに使用します。

1. 45 ページの「新規プロジェクトの作成」の手順に従ってプロジェクトを作成します。
2. [Project Type] ページで [Import ISE Place & Route results] をオンにします。
このプロジェクト タイプの作成手順は、合成済みネットリスト プロジェクトを作成する手順と同じですが、ISE の配置およびタイミング ファイルをインポートするページがあります。
3. 54 ページの「最上位ネットリストおよびモジュール検索パスの指定」の手順に従って最上位ネットリストを指定します。
4. 53 ページの「デフォルト パーツまたはボードの選択」の手順に従ってプロジェクト デバイスを選択します。

57 ページの図 3-11 に示す [Import ISE Implementation Results] ページで、インポートする ISE インプリメンテーション結果を選択します。ISE で生成した配置配線結果およびタイミング ファイルをインポートできます。これらのファイルからインプリメント済みデザインが作成され、表示および解析できるようになります。

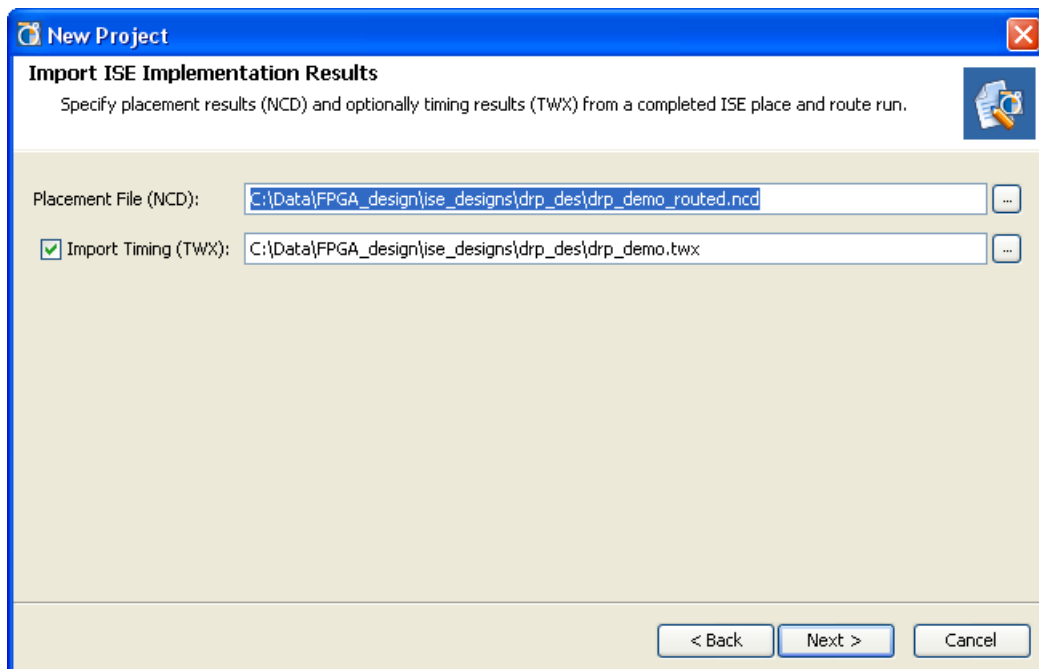


図 3-11 : New Project ウィザード : [Import ISE Implementation Results] ページ

5. [Import ISE Implementation Results] ページで、次を指定します。

- [Placement File (NCD)] : インポートするインプリメント済み NCD ファイルを選択します。PlanAhead によりファイルが自動的に変換され、配置情報が読み出されます。NCD は、ISE ソフトウェアの配置配線ツール (PAR) の出力ファイルです。
 - [Import Timing (TWX)] : チェック ボックスをオンにし、ISE インプリメンテーションの TRCE で生成された TWX などのタイミング結果ファイルを指定します。
6. [Next] をクリックします。
- [New Project Summary] ページにプロジェクトに選択されたオプションが表示されます。
7. [Finish] をクリックし、プロジェクトを作成して開きます。

デザインの最上位モジュールを選択する画面が表示されることがあります。PlanAhead では最上位モジュールが自動的に選択され、その選択を確認するメッセージが表示されます。インプリメント済みデザインが完了すると、次のいずれの動作を実行するか選択する画面が表示されます。

[Open Implemented Design]、[Generate Bitstream]、[View Reports]。

インプリメント済みデザイン環境にデザインの配置およびタイミング結果が読み込まれ、関連のビューが表示されます。

外部プロジェクトのインポート

ISE、XST、または Synplify ツールなどの PlanAhead 以外で作成した RTL レベルのプロジェクト ファイルをインポートすることができます。PlanAhead では、指定したプロジェクトのソース ファイルが検出され、新規プロジェクトへ自動的に追加されます。最上位モジュール、ターゲット デバイス、VHDL ライブラリなどの設定もこれらの既存プロジェクトからインポートできます。

1. 45 ページの「新規プロジェクトの作成」の手順に従ってプロジェクトを作成します。
2. 46 ページの図 3-2 に示す [Project Type] ページで [Imported Project] をオンにします。
58 ページの図 3-12 に示す [Import Project] ダイアログ ボックスが開きます。

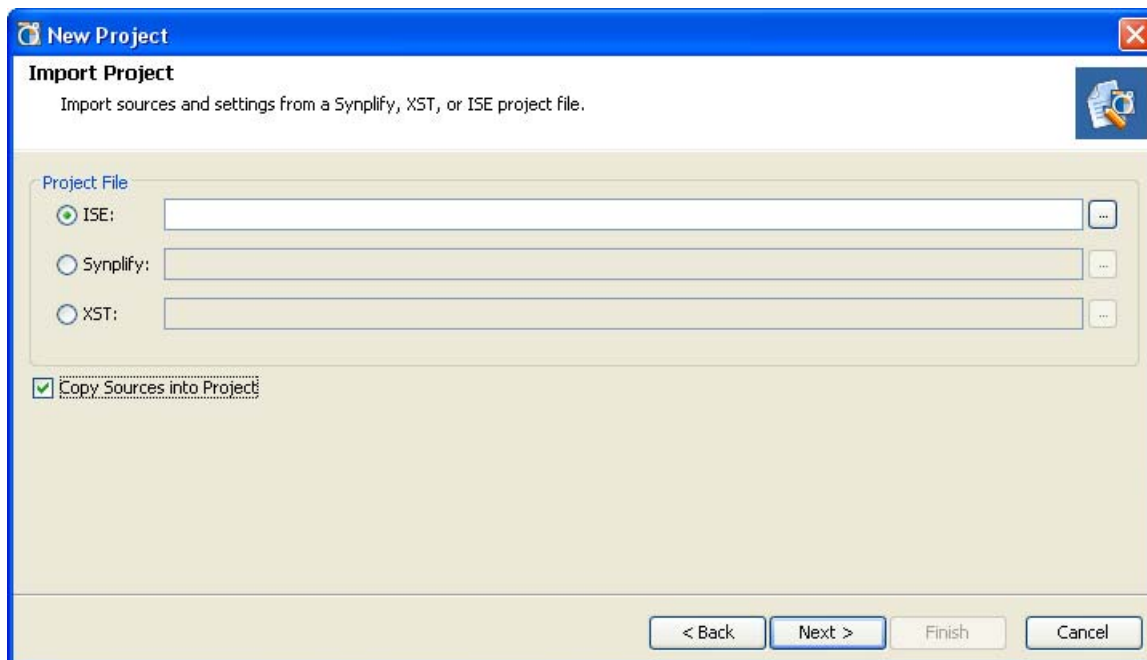


図 3-12 : プロジェクトのインポート

3. インポートするプロジェクト ファイルを指定します。
 - [ISE]: 指定したザイリンクス ISE プロジェクト ファイル (.xise) をインポートします。
 - [Synplify]: 指定した Synplify プロジェクト ファイル (.prj) をインポートします。
 - [XST]: 指定した XST プロジェクト ファイル (.xst) をインポートします。
4. デザイン ソースを現在のディレクトリから参照するのではなくローカルの PlanAhead プロジェクトにコピーする場合は、[Copy Sources into Project] をオンにします。
5. [Next] をクリックします。
[New Project Summary] ページにプロジェクトに選択されたオプションが表示されます。
6. [Finish] をクリックし、プロジェクトを作成して開きます。
指定したプロジェクトの RTL ソース ファイル、制約ファイル、run の設定がインポートされ、指定したディレクトリにプロジェクト ファイルが作成されます。

インポート プロセスのサマリがインポート サマリ レポートのログ ファイルに記述され、新規プロジェクト ディレクトリに保存されます。このサマリ ファイルで、プロジェクトの作成に使用された手順やエラーや警告を確認できます。

プロジェクトの管理

既存のプロジェクトを開く

PlanAhead で既存のプロジェクトを開くことができます。プロジェクトを開くと、前回プロジェクトを閉じたときの状態が復元されます。プロジェクトの状態とは、ソース ファイル順、ソース ファイルのディスエーブル/イネーブル、アクティブおよびターゲット制約ファイル、合成、シミュレーション、インプリメンテーション run のステートなどです。

プロジェクトを開くには、次のいずれかを実行します。

- Getting Started ページの [Open Recent Project] または [Open Project] リンクをクリックします。
- [File] → [Open Project] をクリックします。
- ツールバーの [Open Project] ボタンをクリックします。



[Open Project] ダイアログ ボックスで、PlanAhead プロジェクト ファイル (.ppr) を選択します。[Open Project] ダイアログ ボックスの [File Preview] に現在選択されているファイルに関する情報が表示されます。

Windows のエクスプローラーからプロジェクトを開くには、PlanAhead のプロジェクト ファイルをダブルクリックします。

複数のプロジェクトを開く

1 つのセッションで複数のプロジェクトを開くには、プロジェクトが開いている状態で前述のいずれかの方法を使用して別のプロジェクトを開きます。現在開いているプロジェクトを閉じるかどうかを選択するダイアログ ボックスが表示されます。

[No] をクリックして開いているプロジェクトを閉じないようにすると、両方のプロジェクトが開きます。各プロジェクトに対してそれぞれメイン ウィンドウが開きます。

複数プロジェクトを同じ PlanAhead プロセスから開くと、開いているすべてのプロジェクトで使用されたコマンドが Tcl コンソールへ書き込まれます。ただし、表示されるコマンドがどのプロジェ

クトで使用されたものかわかりにくく、すべてのプロジェクトに対して 1 つの planAhead.jow/.log ファイルしかないので、さらにわかりにくくなっています。

注記：複数のプロジェクトを開いた場合、システム メモリの要件により、パフォーマンスが低下することがあります。

プロジェクトの保存

プロジェクトを保存するには、[File] → [Save Project] または [File] → [Save Project As] をクリックします。保存されていない変更がある場合は、それを示すメッセージが表示されます。[Save Project As] コマンドを使用すると、プロジェクト ディレクトリ構造全体が新しく指定されたディレクトリにコピーされ、run のステータスも保持されます。

プロジェクトのアーカイブ

プロジェクトのバックアップを作成したりリモート サイトに送信したりするには、[File] → [Archive Project] をクリックしてプロジェクトのアーカイブを作成します。PlanAhead では、表示環境が次のように管理されます。

- デザイン階層を解析します。
- ライブラリ ディレクトリから必要なソース ファイル、インクルード ファイル、リモート ファイルをコピーします。
- 制約をコピーします。
- 合成、シミュレーション、およびインプリメンテーションの実行結果をコピーします。
- プロジェクトの ZIP ファイルを作成します。

60 ページの図 3-13 に [Archive Project] ダイアログ ボックスを示します。

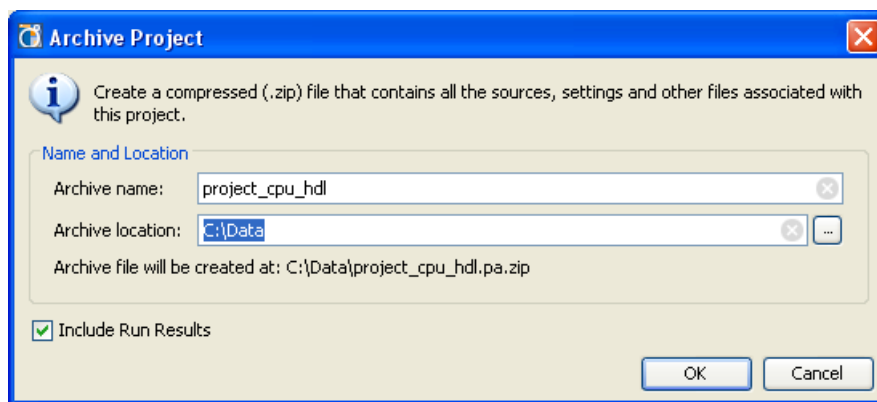


図 3-13：プロジェクトのアーカイブ

プロジェクトのアーカイブを作成するには、次の手順に従います。

1. [File] → [Archive Project] をクリックします。
2. アーカイブ名を指定します。
3. アーカイブ ファイルを保存するディレクトリを指定します。
4. run の設定および結果を含める場合は、[Include Run Results] をオンにします。[OK] をクリックしてアーカイブを作成します。

ソース ファイル、インクルード ファイル、run ファイル (指定した場合) を含むプロジェクトのアーカイブの ZIP ファイルが作成されます。アーカイブ プロセスを記述する `archive.log` ファイルも作成され、ZIP ファイルに含まれます。`archive.log` ファイルでアーカイブの作成プロセスを確認できます。

プロジェクトを閉じる

プロジェクトを閉じるには、[File] → [Close Project] をクリックします。保存されていない変更がある場合は、それを示すメッセージが表示されます。

プロジェクト ソースの管理

PlanAhead では、新規ソース ファイルを作成し、ローカルまたはリモートにある既存のソース ファイルを管理できます。Verilog および VHDL ソース ファイルは、デザイン フローのどの段階でもプロジェクトに追加できます。また、制約ファイルを作成および追加、シミュレーション テストベンチを追加、既存の IP コアを追加できます。

新規ソース ファイルを作成または既存ソース ファイルのプロジェクトへ追加するには、次の手順に従います。

[File] → [Add Sources] をクリックするか、ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。61 ページの図 3-14 に示す Add Sources ウィザードが表示されます。

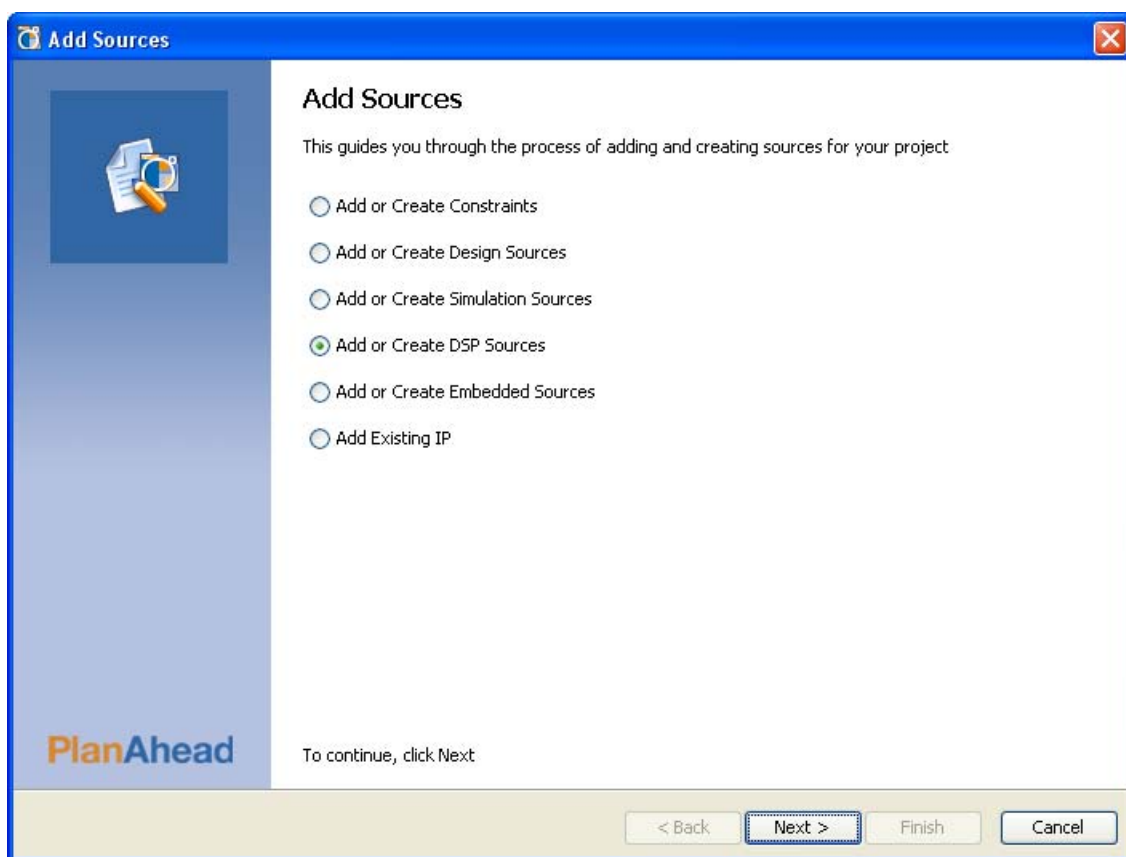


図 3-14 : Add Sources ウィザード

さまざまなタイプのソース ファイルを追加する方法について、この後のセクションで説明します。

- 62 ページの「制約の管理」
- 67 ページの「デザイン ソース ファイルの管理」
- 73 ページの「シミュレーション ソースの管理」
- 84 ページの「DSP ソースの管理」
- 87 ページの「エンベデッド プロセッサ ソースの管理」
- 75 ページの「IP コアの管理」

制約の管理

PlanAhead を使用すると、制約を柔軟に定義できます。1 つの UCF ファイルを使用してデザインで使用されるすべての制約を追加および管理するか、または制約を複数の UCF ファイルに分類して管理できます。UCF ファイルは、プロジェクトの作成中に追加するか、[Add Sources] コマンドを使用して後で追加できます。

複数の制約セットを作成して、さまざまな制約を試したり、複数のバージョンの制約を保存したりすることも可能です。各制約セットには、1 つまたはそれ以上の制約ファイルを含めることができます。

1 つの制約セットを複数のデザインで使用できます。ただし、この場合は変更の管理に注意を払う必要があります。複数のデザインに保存されていない変更がある場合、参照されている制約ファイルにどのデザインを保存するかを選択するダイアログ ボックスが表示されます。

注記：これにより、保存されていないデザインの保存されていない制約定義が上書きされることがあります。

インプリメント済みデザインには、インプリメンテーション **run** の実行中に使用された制約セットのスナップショットが保存されます。この制約セットの名前は、開いているプロジェクトのアクティブ制約セットと同じ名前にすることができます。

インプリメント済みデザインを開く際、インプリメンテーション **run** から読み込まれた制約セットがプロジェクト メモリに現在ある制約セットより古い場合があり、デザインを保存したときに新しく定義された制約が失われる可能性があります。通常 PlanAhead でこれらのリビジョンの問題が管理され、必要に応じて適切な処置をとるようメッセージが表示されますが、メモリにある現在の制約セットとインプリメント済みデザインに関連付けられている制約との間に競合がある可能性があることを念頭に置いてください。

制約ファイルの追加と作成

PlanAhead では、さまざまな制約ファイルフォーマットがサポートされています。UCF または XCF の最上位制約、あるいは NCF のモジュール レベル制約を追加できます。制約には、配置、タイミング、および I/O に関する指示があります。

制約ファイルをプロジェクトに追加するには、次の手順に従います。

1. [File] → [Add Sources] をクリックするか、ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。

61 ページの図 3-14 に示す Add Sources ウィザードが表示されます。

2. [Add or Create Constraints] をオンにし、[Next] をクリックします。

図 3-15 に示す [Add or Create Constraints] ページが表示されます。

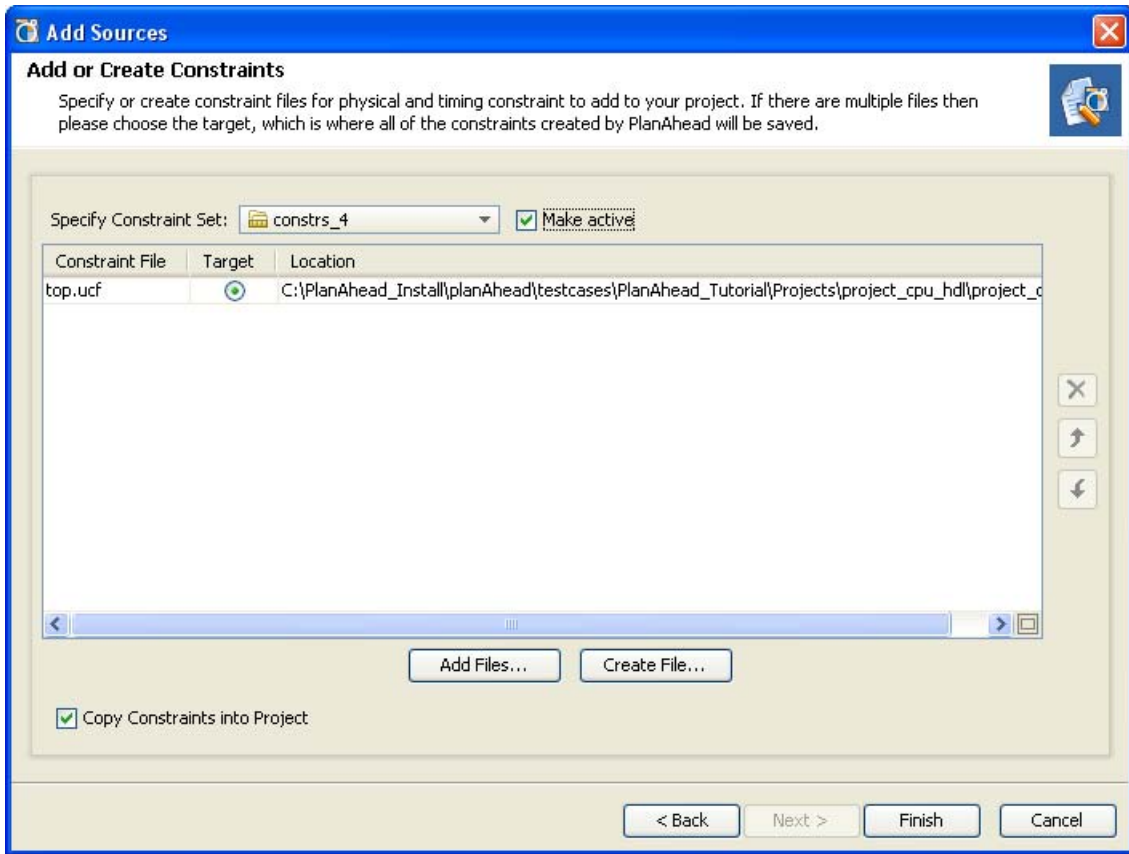


図 3-15 : 制約の追加または作成

[Add or Create Constraints] ページには、次のオプションがあります。

- [Specify Constraint Set] : 制約ファイルを追加する制約セットを選択します。デフォルトでは現在アクティブな制約セットが選択されていますが、別の制約セットを指定したり、ドロップダウンメニューを使用して新しい制約セットを作成することもできます。
- [Add Files] : プロジェクトに追加する UCF、NCF、または XCF ファイルを選択します。
- [Create File] : プロジェクトの最上位 UCF ファイルを作成します。
- [Remove selected files and directories] : 制約ファイルのリストで選択している UCF ファイルを削除します。
- [Move selected file up]/[Move selected file down] : 選択した UCF ファイルをリストの上方向または下方向に移動します。制約は順序に依存し、最後に読み込まれた制約でそれ以前に読み込まれた制約が置き換えられます。制約セットに複数の制約ファイルが含まれる場合、[Sources] ビューに表示される順序でファイルが処理されます。最初にリストされているファイルが最初に処理されます。複数の制約ファイルに同じ制約が含まれている場合、後に処理されたファイルの制約定義が使用されます。
- [Copy Constraints into Project] : 制約ファイルを元のロケーションから参照するのではなく、プロジェクトディレクトリにコピーします。

ターゲット UCF の設定

プロジェクトに複数の UCF ファイルを追加した場合、ターゲット UCF を指定する必要があります。新しく作成された制約は、ターゲット UCF ファイルに記述されます。既存の制約を修正した

場合は、ターゲット UCF ではなくその制約が記述されている UCF ファイルがアップデートされます。

ターゲット UCF は、[Sources] ビューを右クリックして [Set Target UCF] をクリックしていつでも変更できます。

注記 : NCF および XCF ファイルは、ターゲットとして指定することはできません。新規制約は、UCF に記述する必要があります。

元の UCF ファイルの参照またはファイルのコピー

ほかのソース ファイルと同様、UCF ファイルもリモートにあるもの参照するか、ローカルプロジェクトにコピーできます。リモート ファイルを追加した場合、最新のファイルが自動的に検出され、最新のファイルを使用してデザインを更新するようメッセージが表示されます。

プロジェクトにファイルをコピーするには、Add Sources ウィザードの [Add or Create Constraints] ページで [Copy Constraints into Project] をオンにします。

詳細は、72 ページの「リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー」を参照してください。

制約セットの使用

制約セットとは、個別に管理されている 1 つまたは複数の制約ファイルで、解析およびインプリメンテーションでは 1 つの UCF ファイルに連結されます。制約セットは、デザインプロセスのある時点または特定の条件化で使用する制約ファイルを定義します。複数の制約セットを定義することにより、フロアプランやタイミングの問題を解決するために異なる制約を試すことができます。

制約セットには、新しい制約を保存するため、少なくとも 1 つの UCF ファイルを含める必要があります。制約セットに UCF が含まれない場合は、<design>.ucf というファイルが作成され、制約が書き込まれていきます。

制約セットの作成

PlanAhead では、エラボレート済みデザイン、合成済みデザイン、インプリメンテーション、解析など、デザインフローのさまざまな段階で制約を定義できます。

図 3-16 は、[Add Constraints] コマンドを実行すると表示される [Create Constraint Set Name] ダイアログ ボックスを示しています。

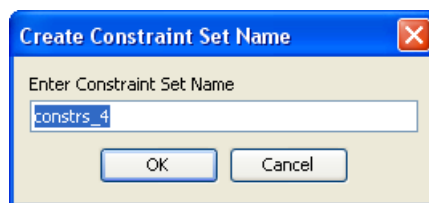


図 3-16 : [Create Constraint Set Name] ダイアログ ボックス

制約セットを作成するには、次の手順に従います。

1. [Add Constraints] ダイアログ ボックスで、[Specify Constraint Set] ドロップダウン リストから [Create Constraint Set] を選択します。

[Create Constraint Set] を選択すると、新規制約セット名を入力する [Create Constraint Set Name] ダイアログ ボックスが表示されます (図 3-16)。名前を入力して [OK] をクリックすると、[Specify Constraint Set] ドロップダウン リストの横に作成した制約セットをアクティブに

する [Make active] チェック ボックスが表示されます (63 ページの図 3-15)。

2. [Add Files] をクリックして、UCF、NCF、または XCF を選択して、制約セットをデザインソースに追加します。
3. [Create File] コマンドを使用して、新しい UCF ファイルの名前とディレクトリを選択します。

[Save Design As] コマンドの使用

設計および解析プロセスで制約に加えた変更を新しい制約セットを作成して保存できます。[Save Design As] ダイアログ ボックスに、すべての制約を保存する新しい制約セット名を入力します。

図 3-17 に、[Save Design As] ダイアログ ボックスを示します。



図 3-17 : [Save Design As] ダイアログ ボックス

制約は複数の方法で変更できるので、変更を新しい制約セットとして保存すると便利です。

[Save Design As] コマンドを使用すると、次の処理が実行されます。

- 新しい制約セットが作成されます。
- アクティブ制約ファイルがローカル プロジェクト ディレクトリの新しい制約セットにコピーされます。
- 制約への変更はコピーされた制約ファイルに記述され、元の UCF ファイルは変更されません。
- デザインのパーティション情報および ChipScope™ コアへの変更が保存されます。
- [Save Design As] ダイアログ ボックスには、新しい制約セットをアクティブにするオプションがあります。

アクティブ制約セットの定義

複数の制約セットが存在する場合、アクティブ制約セットを指定する必要があります。デフォルトでは、次にインプリメンテーションを実行したとき、およびエラボレート済みデザインまたは合成済みデザインを開いたときに、アクティブな制約セットが使用されます。

制約セットをアクティブに設定するには、[Sources] ビューで制約セットを右クリックし、[Make active] をクリックします。図 3-18 に示すように、アクティブ制約セットは太字で表示され、「(active)」と示されます。

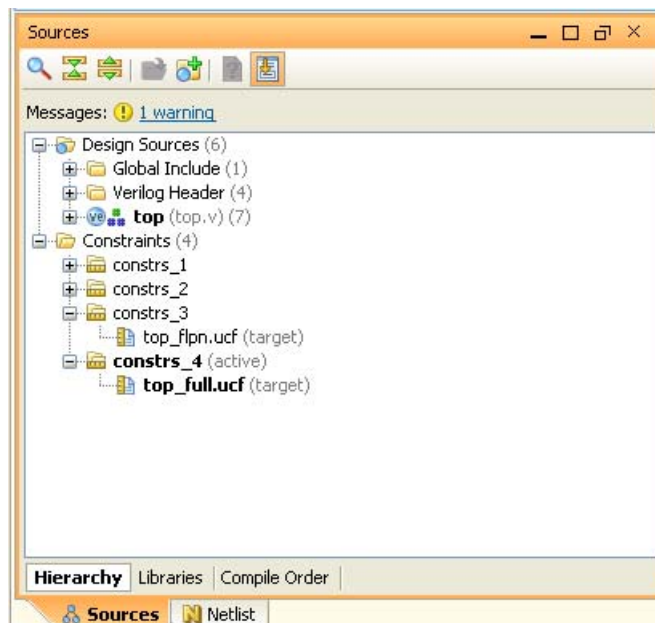


図 3-18 : アクティブ制約セットの設定

モジュール レベルの制約ファイルの使用

PlanAhead では、モジュール レベルの NCF ファイルをプロジェクトに追加できます。NCF ファイルは、通常 IP コアに関連付けられています。モジュール制約ファイルは、適用するモジュールと同じ名前に必要があります。たとえば、xyz.ngc というモジュールのネットリスト制約ファイル名は xyz.ncf にします。NCF ファイルをプロジェクトに追加すると、適切なネットリスト モジュールに自動的に関連付けられます。

注記： NCF ファイルは、モジュール レベルのネットリストの名前に一致している場合に処理されます。

最上位 UCF ファイルとモジュール レベル NCF ファイルの両方で定義されている制約がある場合は、次のルールが適用されます。

1. 制約が重複した場合、UCF の制約が NCF およびエンベデッド ネットリストの制約よりも優先されます。
2. NCF の制約は、エンベデッド制約よりも優先されます。

NCF ファイルは、New Project ウィザードまたは [Add Sources] コマンドを使用して追加します。プロジェクトに読み込まれている RTL ソース ファイルと同じディレクトリに NCF ファイルがある場合は、自動的に [Add Constraints] ページに追加されます。

NGC フォーマットのコアにエンベデッド タイミング制約がある場合も、制約が自動的に読み込まれます。エンベデッド タイミング制約は、EDK や CORE Generator ソフトウェア コアに含まれていることがよくあります。

EDIF ネットリストが PlanAhead に渡される前に、ngc2edif コマンドによりタイミング制約が抽出されます。これにより、PlanAhead で制約が認識されるようになり、デザイン解析で使用されます。これらのファイルは読み出し専用のファイルとして扱われるので直接編集することはできませんが、モジュール レベルの制約に新しい値を定義できます。

新しい制約値は、ターゲット UCF ファイルに記述されます。最上位 UCF ファイルはモジュールレベルの NCF ファイルの後にインプリメンテーションに渡されるので、インプリメンテーションでは新しい制約値が優先されます。NCF は読み出し専用であるため、制約を削除することはできません。

注記：モジュールレベルの制約を変更する場合は、IP 作成方法を使用して元のソースでファイルを編集するのが理想的です。

制約のエクスポート

PlanAhead で制約ファイルを作成し、コマンドライン デザイン フローでのスクリプトに使用する場合があります。コマンドライン フローから制約をエクスポートするには、[File] → [Export] → [Export Constraints] をクリックします。

ユーザー指定および PlanAhead のデフォルトの I/O ポートおよびバンクの I/O 規格制約を UCF ファイルにエクスポートするには、[File] → [Export] → [Export I/O Ports] をクリックし、UCF ファイルを生成します。

デザイン ソース ファイルの管理

このセクションでは、デザイン ソース ファイルの管理方法について説明します。

デザイン ソースの追加と作成

デザイン ソース ファイルを追加するには、[Add Sources] コマンドをクリックして [Add or Create Design Sources] をオンにし、[Next] をクリックします。

68 ページの図 3-19 に示す [Add or Create Design Sources] ページが表示されます。

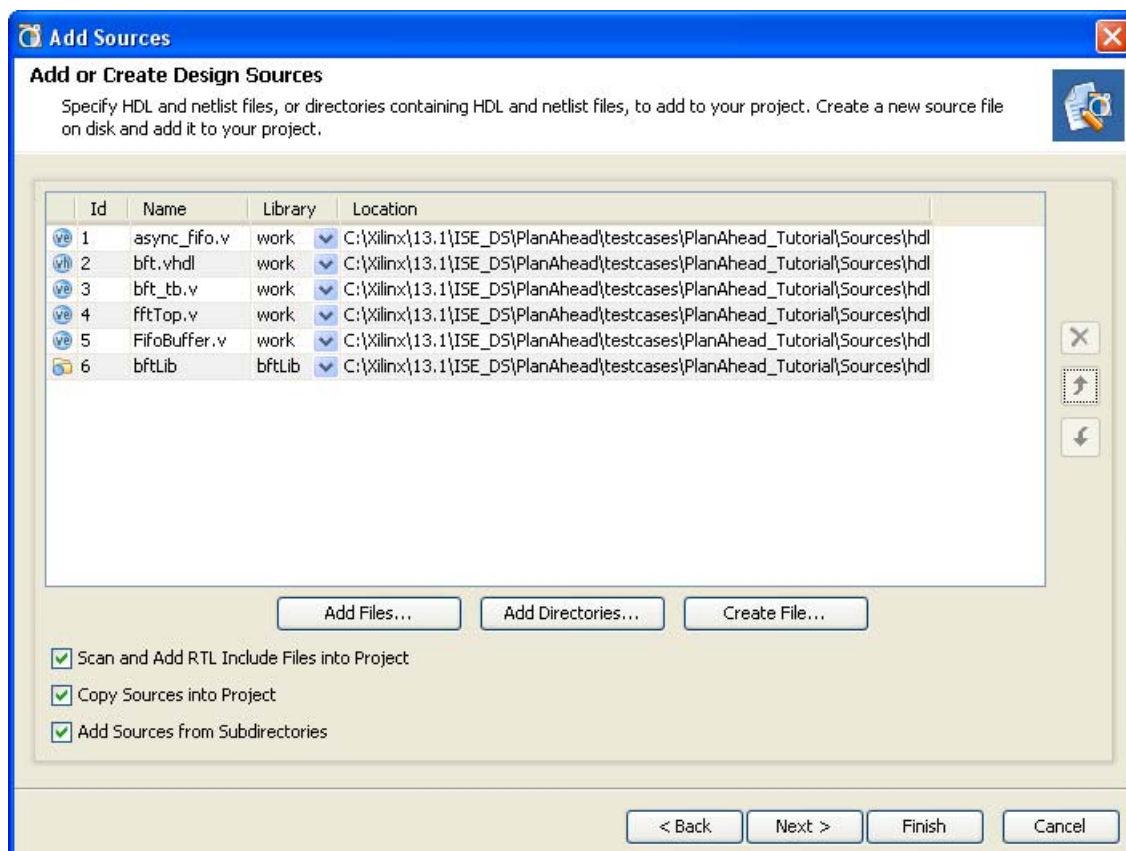


図 3-19 : Add Sources ウィザード : [Add or Create Design Sources] ページ

RTL ソースの追加

[Add or Create Design Sources] ページでは、次のコマンドおよびオプションを使用できます。

- [Add Files]** : プロジェクトに追加するファイルを選択します。プロジェクトに追加できるファイルタイプは、HDL ファイル、EDIF および NGC ファイル、BMM、ELF およびその他のファイルです。

注記 : 各ファイルまたはディレクトリは、それがファイルであるのかフォルダーであるのかを示すアイコンで表示されます。小さい赤い四角は、読み出し専用であることを示します。
- [Add Directories]** : 選択したディレクトリに含まれるすべてのファイルを追加します。指定したディレクトリにある有効なソースファイルがすべてプロジェクトに追加されます。
 - [Library]** : ファイルまたはディレクトリの RTL ライブラリを指定します。定義済みのライブラリ名から選択するか、新規ライブラリ名を入力します。
 - [HDL Source for]** : 読み込むソースが合成およびシミュレーション用の RTL ソースファイルであるか、シミュレーションのみで使用する RTL テストベンチであるかを指定します。
- [Create File]** : Verilog、Verilog ヘッダー、または VHDL ソースファイルを作成する [Create Source File] ダイアログボックスが開きます。詳細は、69 ページの「RTL ソースの作成」を参照してください。
- [Remove selected files and directories]** : 選択したソースファイルまたはディレクトリを削除します。

	Id	Name
1	1	Xilinx_1_ch_t
2	2	prml_core
3	3	zeus_pll.v

- [Move selected files and directories up] : ファイルまたはディレクトリをリストの上方向に移動します。ファイル順は、合成やシミュレーションなどのダウンストリーム プロセスでのエラーポレーションおよびコンパイルの順序に影響します。
- [Move selected files and directories down] : ファイルまたはディレクトリをリストの下方向に移動します。
- [Scan and Add RTL Include Files into Project] : 追加された RTL ファイルをスキャンし、参照されている Verilog インクルード ファイルを追加します。
- [Copy Sources into Project] : ソース ファイルをプロジェクト ディレクトリにコピーします。プロジェクトではローカルにコピーされたバージョンが使用されます。
- [Add Directories] ボタンをクリックしてソース ファイルのディレクトリを追加した場合は、ファイルがローカルプロジェクトにコピーされる際にディレクトリ構造もそのまま保持されます。これについては、[72 ページの「リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー」](#)を参照してください。
- [Add Sources from Subdirectories] : [Add Directories] で指定したディレクトリのサブディレクトリに含まれるソース ファイルをすべて追加します。

RTL ソースの作成

新規 RTL ソース ファイルを作成するには、次の手順に従います。

1. [Add or Create Design Sources] をオンにし、[Add Sources] をクリックします。
2. [Create File] をクリックします。

[図 3-20](#) のような [Create Source File] ダイアログ ボックスが開き、作成するソース ファイルのタイプ、名前、および場所を指定できます。

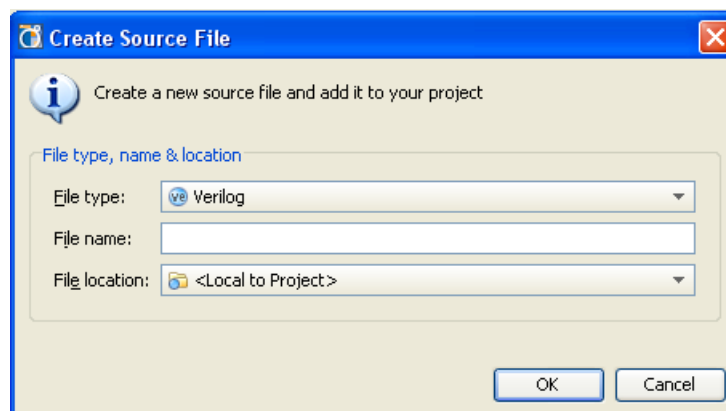


図 3-20 : [Create Source File] ダイアログ ボックス

3. [Create Source File] ダイアログ ボックスでは、次の情報を定義できます。
 - [File type] : 次のファイル タイプのいずれかを選択します。
 - [Verilog] : Verilog ファイル (.v) を作成します。
 - [Verilog Header] : Verilog ヘッダー ファイル (.vh) を作成します。
 - [VHDL] : VHDL ファイル (.vhd1) を作成します。
 - [File name] : 作成する HDL ソース ファイルの名前を入力します。
 - [File location] : ファイルを作成する場所を指定します。
4. [OK] をクリックします。

ファイルのプレースホルダーがソースのリストに追加されます。ファイルは [Finish] をクリックすると作成されます。 .

5. [Create File] コマンドを何度か繰り返して、プロジェクトに追加するモジュール数を定義します。
6. [Add Sources] ページでソース ファイルに最適なライブラリを指定します。デフォルトでは、ソース ファイルは work ライブラリに追加されます。
7. [Finish] をクリックすると、指定したソースがプロジェクトに追加されます。[Create Files] を使用すると、70 ページの「新規モジュールの定義」で説明されるような [Define Modules] ダイアログ ボックスが開きます。

新規モジュールの定義

プロジェクトに追加する新規 RTL ソース ファイルを指定したら、モジュールを定義するために Verilog または VHDL コードを記述する必要があります。図 3-21 に示すような [Define Modules] ダイアログ ボックスを使用すると、新規 RTL コードを作成できます。

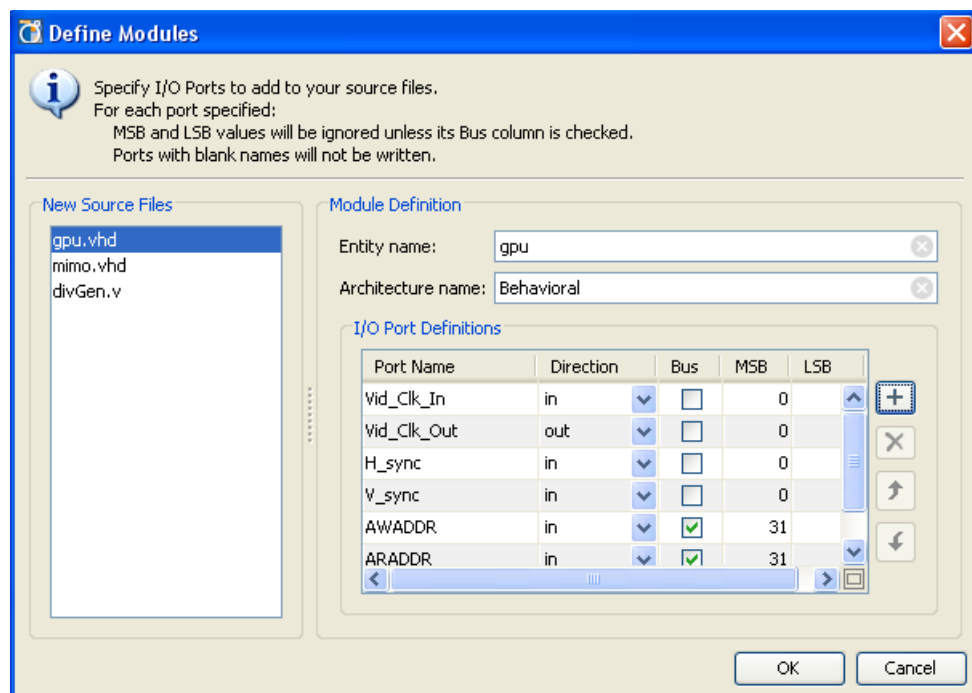


図 3-21 : [Define Modules] ダイアログ ボックス

1. [Define Modules] ダイアログ ボックスには、69 ページの「RTL ソースの作成」で説明された [Create File] コマンドで定義された新規ソース ファイルが表示されます。ここでは、Verilog および VHDL モジュールの両方が定義できます。
2. まず、[New Source Files] のリストから定義するモジュールを選択します。
3. [Define Modules] ダイアログ ボックスには、RTL コード記述を簡単にするために、次の情報を指定できます。
 - [Entity name/Module name] : VHDL コードのエンティティまたは Verilog コードのモジュール名の名前を指定します。エンティティまたはモジュール名はデフォルトでそのファイル名になりますが、別の名前を付けることもできます。

- **[Architecture name]** : RTL ソース ファイルのアーキテクチャを指定します。デフォルトでは **[Behavioral]** です。

注記：このフィールドは、VHDL コードの場合にのみ表示され、Verilog モジュールを定義する場合には表示されません。

- **[I/O Port Definitions]** : モジュール定義に追加するポートを定義します。
 - **[Port Name]** : RTL コードに記述されるポートの名前を定義します。
 - **[Direction]** : ポートを入力、出力、双方向のいずれかに指定します。
 - **[Bus]** : バス ポートであるかどうかを指定します。次の **[MSB]** および **[LSB]** フィールドを使用してポートのバス幅を定義します。
 - **[MSB]** : 最上位ビットの数を定義します。**[LSB]** フィールドと組み合わせて、定義されるバスの幅を指定します。
 - **[LSB]** : 最下位ビットの数を定義します。

注記：ポートがバス ポートでない場合は、MSB および LSB は無視されます。

4. **[New Source Files]** の下の各新規モジュールの詳細を定義し終わったら、**[OK]** をクリックして RTL ソース ファイルを作成して、モジュールをプロジェクトに追加します。新しく定義されたモジュールが **[Sources]** ビューに表示されます。
5. この新規ソース ファイルをテキスト エディターで開いて、必要に応じて編集します。テキスト エディターでファイルを開いて編集するには、ファイルをダブルクリックするか、**[Open File]** ポップアップ メニューをクリックします。新規作成したファイルの編集方法については、[第 4 章の「テキスト エディターの使用」](#)を参照してください。

最上位モジュールの指定とソース ファイルの順序の変更

PlanAhead では、デザイン階層の最上位およびプロジェクトに追加されるファイルのエラボレーション、合成、シミュレーションの順序が自動的に判断されます。デザイン階層は、**[Sources]** ビューの **[Hierarchy]** タブに表示されます。ファイルの順序は、**[Sources]** ビューの **[Compile Order]** タブに表示されている順序になります。詳細は、[144 ページの「\[Sources\] ビュー」](#)を参照してください。最上位モジュールの自動指定とコンパイル順序は、**[Sources]** ビューの **[Hierarchy Update]** コマンドで制御されます。

ただし、最上位モジュールの自動指定は、デザイン階層の最上位を手動で指定すると上書きできません。最上位モジュールを定義するには、**[Sources]** ビューでファイルを右クリックして **[Set as Top]** をクリックします。

注記：選択した最上位モジュールがデザイン ソース ファイルで見つからない場合は、自動的に最適なモジュールになり、階層アップデート モードは **automatic** に設定されます。

PlanAhead では、新しい最上位モジュールの要件に基づいて自動的にファイルが並び替えられます。また、ユーザーの要件に従って手動でファイル順を変更することもできます。**[Sources]** ビューの **[Refresh Hierarchy]** を使用すると、ソース ファイルのアップデートに基づいてファイルを自動的に並び替えることができます。

ソース ファイルの順序を手動で指定するには、**[Sources]** ビューでファイルを選択し、ドラッグして適切な位置に移動します。または、ファイルを右クリックして **[Move Up]**、**[Move Down]**、**[Move to Top]** または **[Move to Bottom]** をクリックして並び替えることもできます。

ソース ファイルのイネーブル/ディスエーブル

ソース ファイルを追加または作成すると、デフォルトでイネーブルになります。ソース ファイルは、エラレーション、合成、またはシミュレーションで使用されないようディスエーブルにできます。

ソース ファイルの異なるバージョンを読み込み、適切なソース ファイルをイネーブルまたはディスエーブルにし、デザインのコンフィギュレーションを制御できます。

- ソース ファイルをディスエーブルにするには、[Sources] ビューでファイルを右クリックし、[Disable File] をクリックします。
- ソース ファイルをイネーブルにするには、[Sources] ビューでファイルを右クリックし、[Enable File] をクリックします。

リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー

ソース ファイルは、リモート ロケーションから参照するか、プロジェクト ディレクトリにコピーできます。プロジェクトを移動またはアーカイブする可能性がある場合は、すべてのファイルがプロジェクト内に保存されるように、ファイルをプロジェクトにコピーすることをお勧めします。

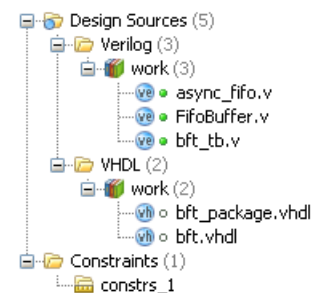
リモート ファイルを追加した場合、最新のファイルが自動的に検出され、開いているデザインを更新するか ([Refresh your open Designs])、アップデートされたファイルを使用して合成を実行するか ([Synthesize with the latest updates]) を選択するダイアログ ボックスが表示されます。

注記：プロジェクトにファイルをコピーすると、プロジェクトを別のシステムに移行しやすくなりますが、外部ファイルの変更は **PlanAhead** では自動的に検出されません。リモート ファイルの変更を適用するには、ファイルを削除して追加し直るか、[Sources] ビューのコマンドを使用してファイルをアップデートします。

[Add Sources] コマンドでソース ファイルをプロジェクトに追加する際に [Copy Sources into Project] をオンにすると、ソース ファイルがプロジェクト ディレクトリにコピーされます。

ソース ファイルを最初リモート ソースとして追加し、後でプロジェクト ディレクトリにコピーする場合は、[Sources] ビューでファイルを右クリックして [Copy Into Project] をクリックしてソース ファイルを個別にコピーするか、または [Copy All Files Into Project] をクリックしてすべてのリモート ソース ファイルをコピーします。

- プロジェクト ディレクトリにコピーされたローカル ファイルには、[Sources] ビューのファイル名の横に緑の点が付きます。
- ローカルプロジェクト ディレクトリにコピーされないリモート ソースの場合、ファイル名の横に点ではなく、緑の丸が表示されます。
- 赤い丸が表示されている場合は、RTL ファイルがローカルにもリモートにも見つからないことを示します。



ローカル ソース ファイルのアップデート

リモート ソースを参照すると、そのアップデートは **PlanAhead** で自動的に検出されます。ソース ファイルがプロジェクトにコピーされている場合は、元のファイルへの変更は検出されません。必要に応じて、ローカル ソース ファイルを手動でアップデートする必要があります。

ローカル ソース ファイルをアップデートするには、次の方法があります。

- [Sources] ビューでソース ファイルを右クリックして [Update File Contents] をクリックします。

ファイル ブラウザーにコピー元のソース ファイルが表示されます。[OK] をクリックして元のソース ファイルを読み込み直し、ソース ファイルへの変更を含めてプロジェクト ファイルがアップデートされます。

別のファイルを指定することもできます。この場合、新規ファイルの内容がローカルのプロジェクト ファイルにコピーされます。

たとえば、元のファイルが File_1.v で File_2.v を選択した場合、File_2.v の内容が元の File_1.v の内容に上書きされます。

- [Sources] ビューを右クリックして [Add Sources] をクリックし、アップデートされたソース ファイルをプロジェクトに追加します。追加したファイルがプロジェクトにインポートされます。

同じ名前にローカル ソース ファイルが既に存在するので、73 ページの図 3-22 に示す [Import Source Conflicts] ダイアログ ボックスが表示され、既存のファイルを上書きするか ([Overwrite existing files])、新しく追加したファイルを読み込まないか ([Don't overwrite existing files]) を選択するように求められます。

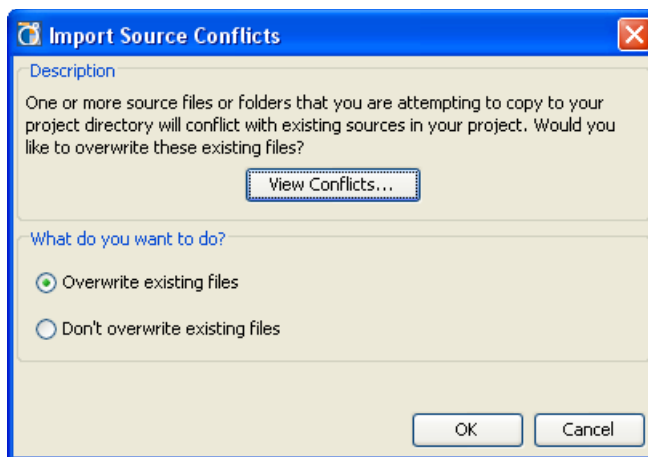


図 3-22 : [Import Source Conflicts] ダイアログ ボックス

シミュレーション ソースの管理

PlanAhead では、RTL プロジェクトのビヘイビア シミュレーション用およびインプリメント済みデザイン タイミング シミュレーション用にシミュレーション ソースを追加できます。シミュレーション ソース ファイルには、シミュレーションのステイミュラスとして使用する HDL ベースのテストベンチ ファイルが含まれます。シミュレーション ソースは、ISim でのビヘイビア シミュレーションおよびタイミング シミュレーションに使用されます。詳細は、214 ページの「ビヘイビア シミュレーションの実行」および397 ページの「タイミング シミュレーションの実行」を参照してください。

シミュレーション ソース ファイルはシミュレーション ファイル セットに格納され、[Sources] ビューにフォルダーとして表示されます。リモートのものを参照するか、ローカル プロジェクト ディレクトリに保存されているものを使用できます。

シミュレーション セットにより、デザインの異なる段階に異なるソースを使用できます。たとえば、エラボレート済みデザインまたはデザインのモジュールのビヘイビアー シミュレーション用にスティミュラスを供給するシミュレーション ソースを使用し、インプリメント済みデザインのタイミング シミュレーション用にスティミュラスを供給する別のテストベンチを使用できます。

シミュレーション ソースをプロジェクトに追加する際、ソースを追加するシミュレーション ソース セットを指定できます。

シミュレーション ソース ファイルの追加と作成

シミュレーション ソースをプロジェクトに追加するには、次の手順に従います。

1. 次のいずれかを実行します。
 - [File] → [Add Sources] をクリックします。
 - ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。
 Add Sources ウィザードが表示されます。
2. [Add or Create Simulation Sources] をオンにし、[Next] をクリックします。74 ページの図 3-23 に示す [Add or Create Simulation Sources] ページが表示されます。

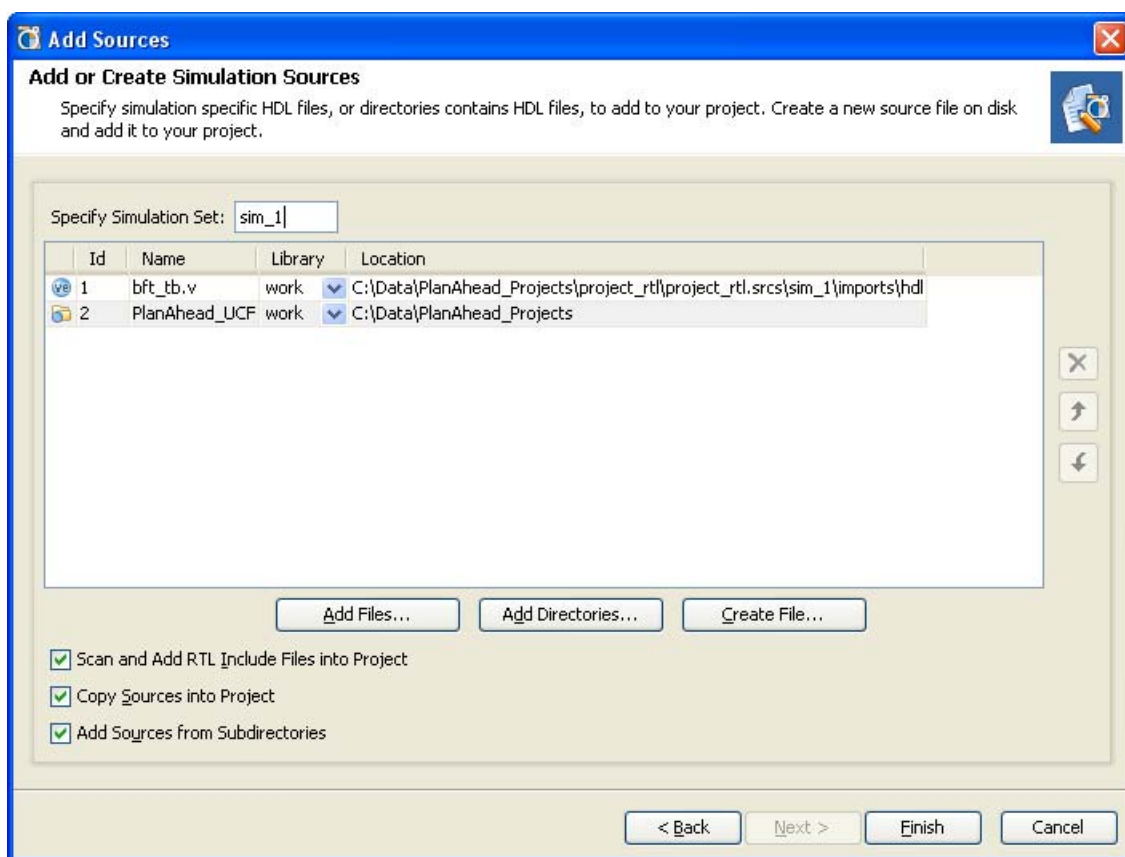


図 3-23 : [Add or Create Simulation Sources] ページ

このダイアログ ボックスには、次のようなオプションがあります。

- [Specify Simulation Set] : シミュレーション セットの名前をテストベンチ ファイルおよびディレクトリに入力します。1 つまたは複数のシミュレーション セットが既に定義されている

場合は、プルダウン メニューから [Create Simulation Set] コマンドを選択して、新しいシミュレーション セットを定義します。

- [Add Files] : プロジェクトに追加するシミュレーション ソース ファイルを選択します。
- [Add Directories] : 選択したディレクトリに含まれるすべてのシミュレーション ソース ファイルを追加します。指定したディレクトリにある有効なソース ファイルがすべてプロジェクトに追加されます。
- [Library] : ファイルまたはディレクトリのライブラリを指定します。定義済みのライブラリ名から選択するか、新規ライブラリ名を入力します。

注記 : Verilog ライブラリは /work ディレクトリです。

- [Create File] : 新規シミュレーション ファイルを作成する [Create Source File] ダイアログボックスが開きます。詳細は、69 ページの「RTL ソースの作成」を参照してください。
- [Remove selected files and directories] : 選択したソース ファイルまたはディレクトリを削除します。
- [Move selected files and directories up] : ファイルまたはディレクトリをリストの上方向に移動します。
- [Move selected files and directories down] : ファイルまたはディレクトリをリストの下方向に移動します。
- [Scan and Add RTL Include Files into Project] : 追加されたファイルをスキャンし、参照されているインクルード ファイルを追加します。
- [Copy Sources into Project] : ソース ファイルをプロジェクト ディレクトリにコピーします。プロジェクトではローカルにコピーされたバージョンが使用されます。

[Add Directories] ボタンをクリックしてソース ファイルのディレクトリを追加した場合は、ファイルがローカルプロジェクトにコピーされる際にディレクトリ構造もそのまま保持されます。

- [Add Sources from Subdirectories] : [Add Directories] で指定したディレクトリのサブディレクトリに含まれるソース ファイルをすべて追加します。

IP コアの管理

PlanAhead では、ザイリンクス IP カタログ、CORE Generator で作成した既存の IP コア ファイル、およびサードパーティの IP プロバイダーからの IP コアをプロジェクトに追加できます。

RTL ベースのプロジェクトでは、次の方法で IP コアを追加および管理できます。

- 76 ページの「既存の IP コアの追加」
- 77 ページの「ザイリンクス カタログからの IP の追加」
- 81 ページの「ターゲットの生成」
- 83 ページの「IP カタログのアップデート」

DSP モジュールは System Generator for DSP からインポートでき、エンベデッド プロセッサ モジュールは XPS からインポートできます。これらのツールは PlanAhead に統合されているので、PlanAhead ツール内からデザイン モジュールを追加および管理できます。詳細は、84 ページの「DSP ソースの管理」および87 ページの「エンベデッド プロセッサ ソースの管理」を参照してください。

注記 : サードパーティから合成済み NGC または EDIF ネットリストとして提供されている IP もあります。[Add Sources] コマンドを使用すると、これらのファイルをデザインに読み込むことがで

きます。詳細は、67 ページの「デザイン ソース ファイルの管理」を参照してください。

既存の IP コアの追加

既存の IP コア (.xco) をプロジェクトに追加するには、次の手順に従います。

1. [File] → [Add Sources] をクリックするか、ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。

61 ページの図 3-14 に示す Add Sources ウィザードが表示されます。

2. [Add Existing IP] をオンにし、[Next] をクリックします。

76 ページの図 3-24 に示す [Add Existing IP] ページが表示されます。

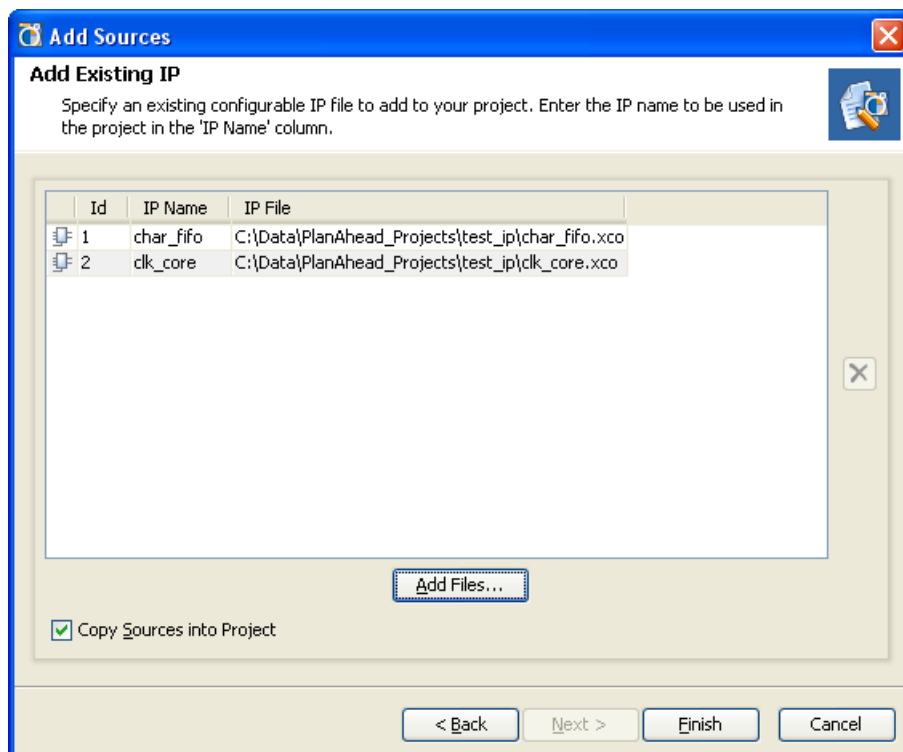


図 3-24 : Add Sources ウィザード : [Add Existing IP] ページ

次のオプションがあります。

- [Add Files] : XCO ファイルを選択して既存の IP をプロジェクトに追加します。
- [Remove selected files and directories] : 選択したソース ファイルまたはディレクトリを削除します。
- [Copy Sources into Project] : 元の IP コア ファイルをプロジェクト ディレクトリにコピーします。プロジェクトではローカルにコピーされたバージョンが使用されます。

プロジェクトに追加する既存の IP コアを選択したら、[Finish] をクリックしてコアを追加します。

追加された IP コアは、[Hierarchy]、[Libraries]、[Compile Order] タブのその他のソース ファイルと一緒に、[Sources] ビューの [IP Sources] タブにそれぞれ表示されます。[Sources] ビューにはコアを構成するファイルが表示され、コアを選択すると [Source File Properties] ビューにプロパティが表示されます。詳細は、144 ページの「[Sources] ビュー」を参照してください。

IP コアの EDIF ネットリストまたは NGC ファイルは、RTL または ネットリスト ベースのプロジェクトに追加することもできます。詳細は、[54 ページの「合成後プロジェクトの作成」](#)を参照してください。

ザイリンクス カタログからの IP の追加

IP カタログを開くには、Flow Navigator の [Project Manager] メニューから [IP Catalog] をクリックします。

IP コアはカテゴリごとにツリー形式で表示され、IP バージョン、AXI (Advanced eXtensible Interface) プロトコル準拠、ステータス、ライセンス要件などが表示されます。IP コアを選択すると、その説明がビューの下部に表示されます。[78 ページの図 3-25](#) に、[IP Catalog] ビューの例を示します。

カタログから IP コアを選択し、そのコアに関するさまざまな情報を確認することができます。選択した IP の PDF 形式のデータシートを表示するには、次を実行します。

- 右クリックで [Data Sheet] をクリック
- IP カタログ ツールバーの [View Information] ボタンをクリックし、ポップアップ メニューから [Data Sheet] をクリック

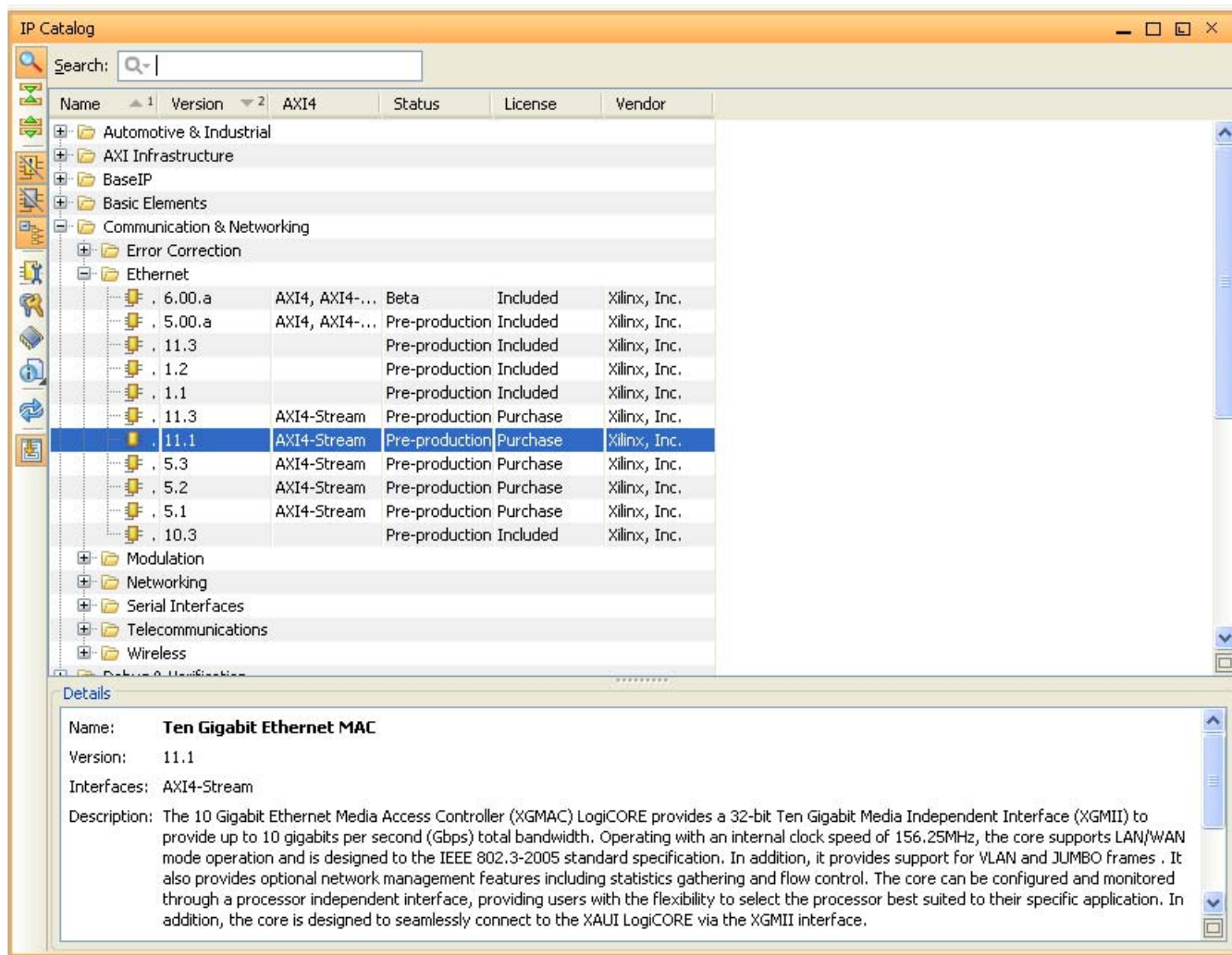
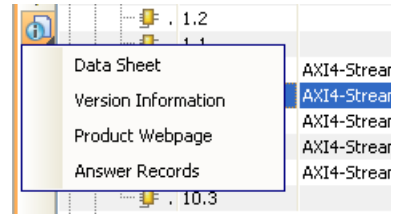


図 3-25 : [IP Catalog] ビュー

IP カタログのツールバーおよびポップアップ メニューには、次のコマンド オプションがあります。

- [Show Search] : カタログ内で文字列を検索するための [Search] フィールドを表示します。
- [Collapse All]/[Expand All] : IP カタログのツリー表示を閉じたり展開したりします。
- [Hide Superseded and Discontinued IPs] (古いバージョンおよび廃盤の IP を非表示) : 最新バージョンの IP のみを表示します。
- [Hide incompatible IPs] (互換性のない IP を非表示) : 選択したデバイス ファミリーでサポートされる IP のみを表示します。
- [Group by Category] (カテゴリ別にグループ化) : リストをカテゴリ別に表示するか、すべてをフラットに表示するかを切り替えます。
- [Customize IP] : 選択した IP のカスタマイズ ウィンドウを開きます。詳細は、79 ページの「[IP のカスタマイズ](#)」を参照してください。
- [License Status] : 選択した IP のライセンス要件およびステータスを表示します。
- [Compatible Families] : 選択した IP と互換性のあるすべてのデバイス ファミリーおよび特定のザイリンクス パーツのリストを表示します。

- [View Information] : 選択した IP の入手可能な情報リソースが表示されます。入手可能な情報には、データシート、バージョン情報、ウェブページ、アンサーなどがあります。
- [Update IP Catalog] : 指定場所の IP カタログを再生成します。このコマンドを使用すると、ザイリンクス IP カタログへのアップデートをチェックできます。詳細は、[83 ページの「IP カタログのアップデート」](#)を参照してください。
- [Automatically scroll to selected objects] : 開いているビューに選択したオブジェクトを表示します。
- [Export to Spreadsheet] : IP カタログをスプレッドシートで使用できるようにするため XLS ファイルに出力します。



IP のカスタマイズ

IP カタログからコアを選択し、パラメーター値を指定することにより、デザイン要件に合わせて IP をカスタマイズできます。IP のカスタマイズには、統合された **CORE Generator** ツールが使用されます。

1. カスタマイズするカタログ内の IP を選択します。
2. ツールバーまたはポップアップ メニューから [Customize IP] をクリックするか、選択した IP をダブルクリックします。

CORE Generator でコアのカスタマイズ ウィンドウが開きます。選択した IP によって、インターフェイスは異なります。

次のタイプのインターフェイスがあります。

- Memory Interface Generator (MIG) ウィザード
- CORE Generator ウィザード

[80 ページの図 3-26](#) に、**CORE Generator** のインターフェイスを示します。

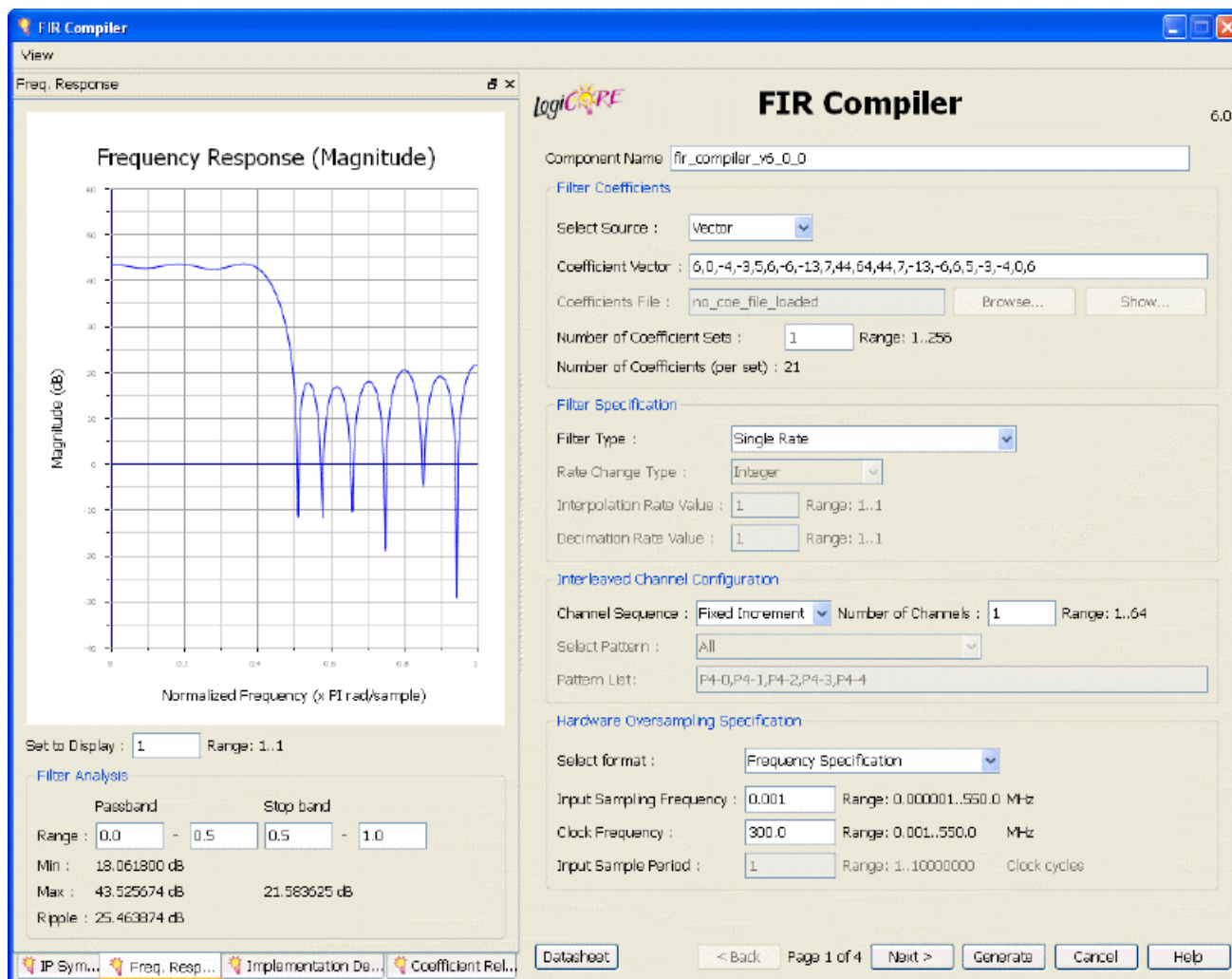


図 3-26 : CORE Generator のインターフェイス

インターフェイスのフィールドで、IP のパラメーターを定義して IP をカスタマイズします。CORE Generator のインターフェイスは、選択したコアのタイプによって異なり、複数のページが含まれることがあります。

また、IP シンボル、周波数応答グラフ、リソース使用率に関するインプリメンテーションの詳細も含まれます。これらの表示は、左下のタブをクリックして切り替えることができます。

パラメーターを設定したら、[Generate] をクリックして IP コアを生成し、プロジェクトのソースとして追加します。この時点では、コアは合成されません。

追加された IP コアが [Sources] ビューの [IP Sources] タブにそれぞれ表示されます。IP コアは、[Hierarchy]、[Libraries]、[Compile Order] タブのその他のソース ファイルと一緒に表示されます。[Sources] ビューにはコアを構成するファイルが表示され、コアを選択すると [Source File Properties] ビューにプロパティが表示されます。詳細は、144 ページの「[Sources] ビュー」を参照してください。

- IP の詳細は、付録 E の「IP 資料」を参照してください。
- CORE Generator を使用した IP の生成については、次を参照してください。
<http://japan.xilinx.com/tools/coregen.htm>

- MIG Memory Generator (MIG) の使用法の詳細は、次を参照してください。
http://japan.xilinx.com/support/documentation/ipmeminterfacestorelement_meminterfacecontrol_mig.htm

ターゲットの生成

プロジェクトに追加された IP コアは [Sources] ビューに表示されます。[IP Sources] タブをクリックすると、IP と収集されたターゲットが表示されます。[Sources] ビューで IP コアを展開表示すると、そのコアに関連するさまざまなターゲット ファイルが表示されます。

ターゲットとは、現在のプロジェクトでそのコアをサポートするのに必要な IP のさまざまなデザイン エレメントのことです。これには、インスタンス化テンプレート、合成済みネットリスト、ログ ファイルやデータシートなどの関連資料などが含まれます。IP (.xco/.xci) のソースファイルも [Sources] ビューの [Hierarchy] タブに表示されます。

図 3-27 は、3 つのコアを含む [IP Sources] タブで、ターゲットおよび関連ファイルも表示されています。

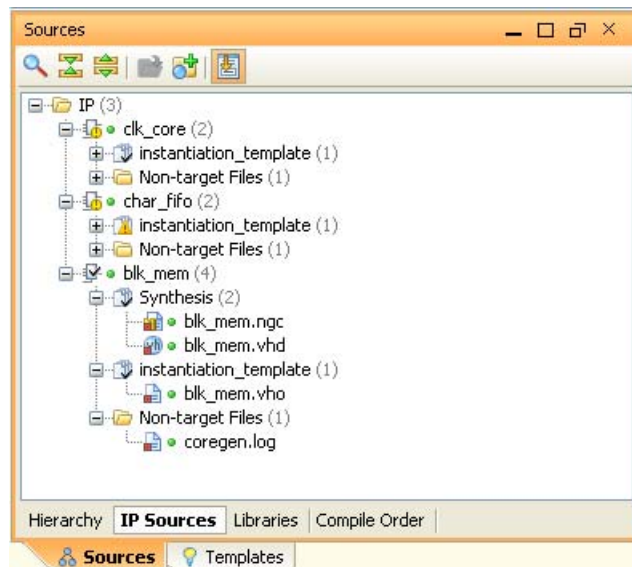


図 3-27 : [IP Sources] タブに表示されるターゲット

ターゲット データは、デザイン フローで必要に応じて自動的に生成されます。デフォルトでは、デザインに IP をインポートすると、インスタンス化テンプレートのみが含まれます。

IP のインスタシエート

インスタンス化テンプレートは、Verilog (.veo) または VHDL (.vho) のモジュール定義で、RTL デザインにコピーして貼り付けて IP モジュールのインスタンスを作成できます。82 ページの図 3-28 は、IP コアのインスタンス化テンプレートの例を示しています。

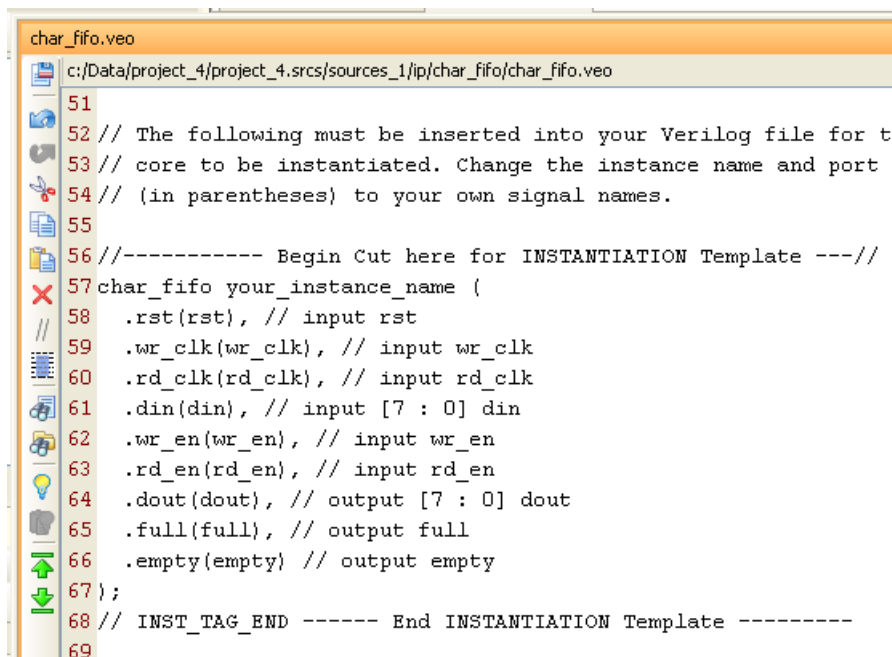


図 3-28 : IP インスタンス化テンプレート

IP をデザインにインスタンス化するには、次の手順に従います。

1. [Sources] ビューに IP コアの Verilog または VHDL テンプレート ファイル、および RTL デザイン ファイルをダブルクリックで開くか、右クリックして [Open File] をクリックして開きます。
2. インスタンス化テンプレートの「Begin Cut Here...」と記述されている行から後を選択し、適切な RTL ファイルにコピーして、モジュールのインスタンスを作成します。
3. IP モジュールをデザインに組み込むために必要な変更を加えます。
4. IP コアをデザインに正しくインスタンス化したら、IP コアをデザインの残りの部分と共に合成できます。

IP の合成

必要な合成ファイルは、ツールで必要とされるまで生成されません。プロジェクトに IP コアが追加されている場合、まずプロジェクトの IP コアが合成されてから最上位デザインが合成されます。これにより、複数の IP コアをインスタンス化する際にプロジェクトに IP コアを追加するたびに合成が実行されなくなり、合成が統合して実行されるので、時間が節約されます。

[Sources] ビューで IP を右クリックし、[Generate] をクリックすると、IP をいつでも個別に合成できます。これにより、83 ページの図 3-29 のような [Generate] ダイアログ ボックスが開き、生成するターゲットを指定できるようになります。PlanAhead ツールでは、現在のところインスタンス化テンプレートと合成ターゲットがサポートされます。

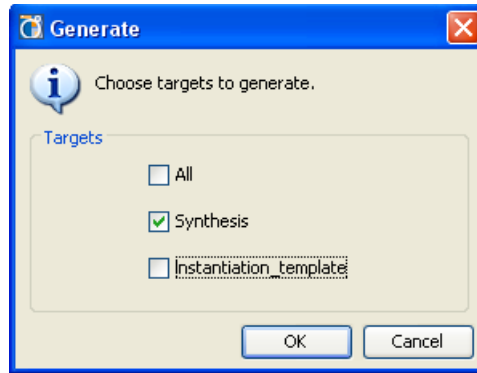


図 3-29 : ターゲットの生成

合成ターゲットを生成すると、コアに対して Xilinx® Synthesis Tool (XST) 合成ツールが実行され、カスタマイズ設定に基づいてロジックが作成されます。

IP が生成されたら、[Sources] ビューの IP ソース アイコンにチェックマークが表示され、合成済みネットリスト ファイル (NGC) が [Sources] ビューに追加されます。



IP のリセット

IP はいいつでもリセットして、IP からのターゲット データを削除したり、現在のデザインの定義を変更するために IP をカスタマイズし直すことができます。

IP コアのターゲット データを削除するには、[Sources] ビューの [IP Sources] タブで IP をクリックし、右クリックで [Reset] コマンドを実行します。[Reset] コマンドでは、選択したターゲットをリセットしたり、すべてのターゲットをリセットできます。これにより、現在のターゲット データは削除されるので、ターゲットを生成し直す必要があります。

現在のデザインで IP の定義を変更する場合は、その IP を [Sources] ビューで選択し、右クリックで [Re-customize IP] をクリックします。これにより、CORE Generator が再び開くので、そのコアに関連するパラメーターを変更します。必要な変更を加えたら、CORE Generator で [Generate] をクリックし、IP をデザインに追加し戻し、インスタンスエーション テンプレートを生成させます。

現在のプロジェクトの IP はカタログの最新バージョンにアップグレードすることもできます。この機能を使用すると、現在のプロジェクトへカタログの変更を加えることができます。[Sources] ビューで IP をクリックし、[Upgrade] コマンドをクリックします。これにより、デザイン内のカスタマイズ済み IP がザイリンクス IP カatalogの最新バージョンにアップデートされます。現在のプロジェクトからのカスタマイズ設定は自動的に適用され直されます。

IP カatalogのアップデート

PlanAhead では、ローカル ディレクトリを IP カatalogに追加したり、新規 IP をレポジトリに追加したり、カatalogの一部としてローカルの IP レポジトリを管理したりできます。IP カatalogを管理するには、ツールバー メニューか右クリック メニューから[Update IP Catalog] をクリックします (84 ページの図 3-30)。

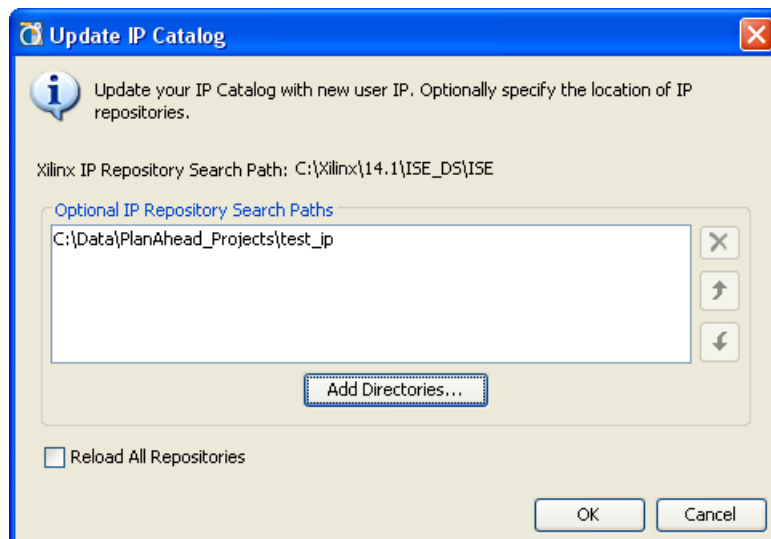


図 3-30 : IP カタログのアップデート

[Update IP Catalog] ダイアログ ボックスには、ツールのインストール時に含まれるザイリンクス IP カタログへのパスが表示されます。

- [Optional IP Repository Search Paths] : 現在 IP カタログに追加されているオプションの IP レポジトリをリストします。
- [Add Directories] : IP レポジトリ検索パスに追加するディレクトリを指定します。
component.xml または ZIP ファイル形式の IP ファイルが IP カタログに追加されます。
- [Remove Selected] : 選択した IP レポジトリ パスを IP カタログから削除します。
- [Move Up] : 選択した IP レポジトリをリストの上方向に移動します。

この順序は、IP コアが IP カタログから読み込まれる順序に影響します。最初の IP レポジトリは常にザイリンクス IP カタログで、各ユーザー定義の IP レポジトリはこの検索パスのリスト順になります。IP コアが重複している場合、後の定義が前の定義を上書きします。この場合、最新の IP レポジトリで見つかった IP コアがデザインにインポートされます。

- [Move Down] : ファイルまたはディレクトリをリストの下方向に移動します。
- [Reload All Repositories] : 各 IP レポジトリのインデックスを再構築させ、その再構築されたインデックス ファイルからメモリ内の IP カタログをアップデートします。新しい IP をレポジトリに追加した場合、このオプションを使用してインデックス ファイルを再構築して、その新しい IP が含まれるようにする必要があります。

デフォルトでは、[Update IP Catalog] コマンドでは既存のインデックス ファイルからメモリ内の IP カタログがアップデートされます。レポジトリのインデックス ファイルが既にある場合は、インデックス ファイルは再生成されず、新しく追加された IP も含まれません。

DSP ソースの管理

PlanAhead ツールでは、既存の System Generator デザイン モデル ファイル (.mdl) を DSP モジュールとしてインポートできます。このモデルはサブモジュールとして階層レベルに追加、またはデザインの最上位にインポートできます。PlanAhead ツール内から新規 DSP モジュールを定義して、System Generator でデザインを完了させることもできます。

System Generator (Sysgen) はザイリンクスの DSP デザイン ツールで、RTL ソース ファイル、Simulink® および MATLAB® ソフトウェア モデル、および DSP システムの C/C++ コンポーネントを 1 つのシミュレーションおよびインプリメンテーション環境にまとめることができます。これらのツールの詳細は、付録 E の「System Generator for DSP 資料」を参照してください。

System Generator デザインは、よく大容量 HDL デザインに組み込まれます。System Generator ではスタンドアロンの FPGA デザインの作成およびインプリメンテーションがサポートされますが、PlanAhead ツールでプロジェクトを開始して、System Generator を使用してプロジェクトの DSP モジュールを開発することをお勧めします。こうすることで、PlanAhead ツールで FPGA デザインのプロジェクトを管理できるので、DSP モジュールを System Generator 内で開発および管理しておいて、PlanAhead で 1 つのソース ファイルとして処理できます。

DSP モジュールの追加

PlanAhead では、既存のモデル ファイル (.mdl) を System Generator からインポートしたり、PlanAhead ツール内で新しい DSP モジュールを定義しておいてから、System Generator を開いてプロジェクトを作成および管理したりできます。

DSP モジュールを追加または作成するには、次の手順に従います。

1. [File] → [Add Sources] をクリックするか、ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。

61 ページの図 3-14 に示す Add Sources ウィザードが表示されます。

2. [Add or Create DSP Sources] をオンにし、[Next] をクリックします。

図 3-31 に示す [Add or Create DSP Sources] ページが表示されます。

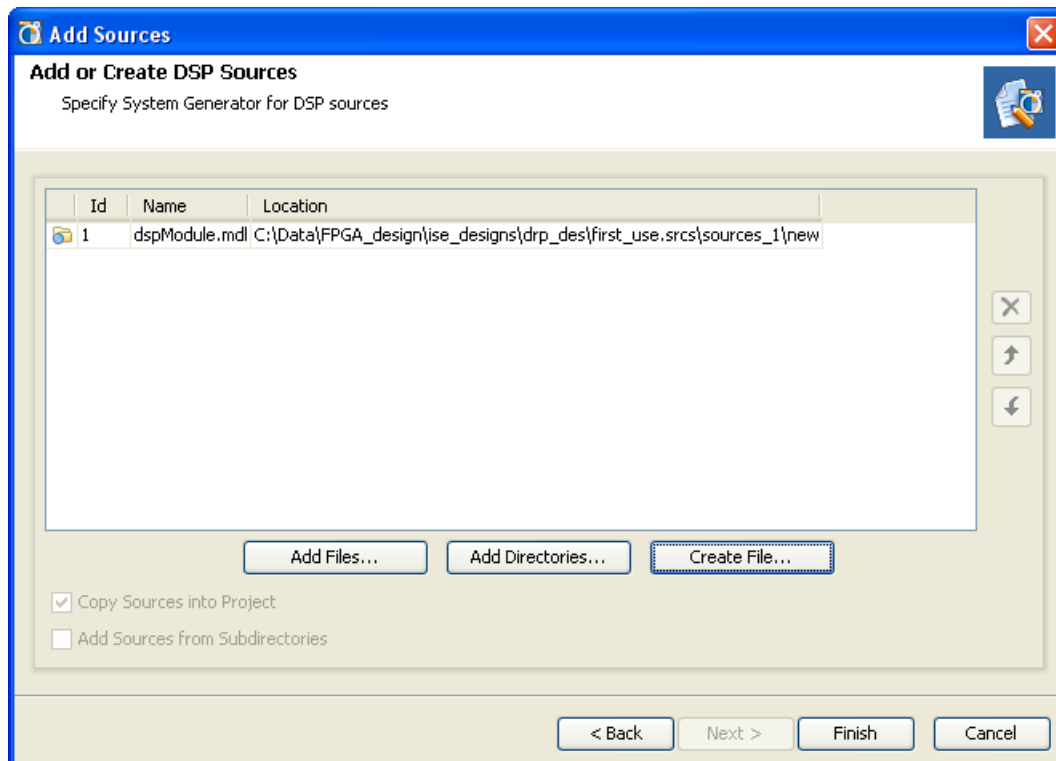


図 3-31 : DSP ソースの追加

- [Add Files] : 既存の DSP ソースを現在のプロジェクトに追加します。追加する既存の System Generator モデル ファイル (.mdl) は、表示されるファイル ブラウザから選択できます。
- [Add Directories] : 選択したディレクトリに含まれる DSP モジュールを追加するディレクトリ ブラウザが開きます。指定したディレクトリの System Generator モデル ファイルがプロジェクトに追加されます。
- [Create File] : 新しく System Generator モデルを作成して PlanAhead ツール プロジェクトに追加します。これにより System Generator が起動するので、新しい DSP モジュールを定義します。
- [Remove] : 選択した DSP ソースを削除します。
- [Move Up] : 選択したソースをリストの上方向に移動します。
- [Move Down] : 選択したソースをリストの下方向に移動します。

プロジェクトに追加または作成する DSP ソースを選択したら、[Finish] をクリックしてファイルを追加します。System Generator および MATLAB で DSP ソースの作成および管理が開始されます (図 3-32)。

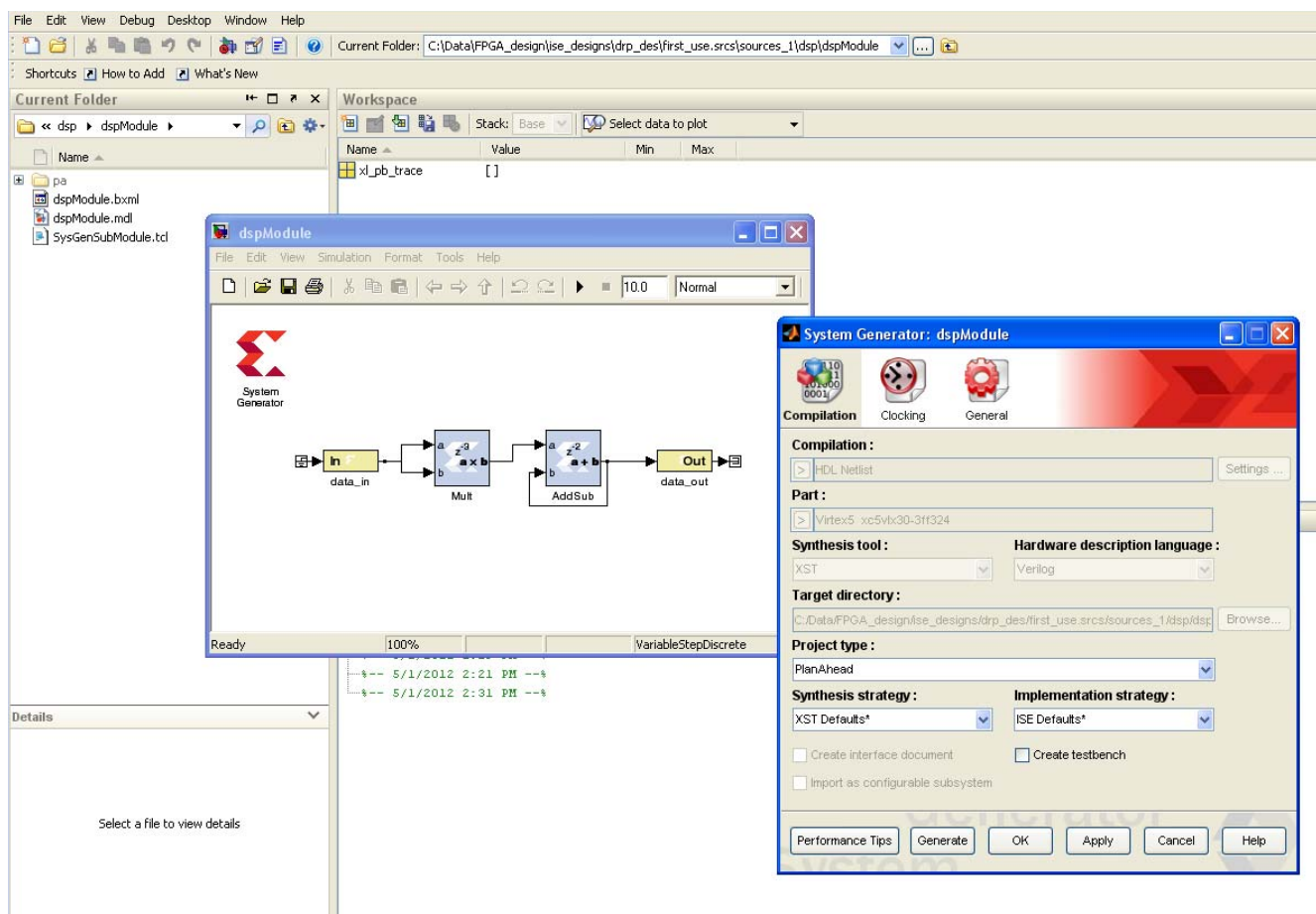


図 3-32 : System Generator での DSP ソースの生成

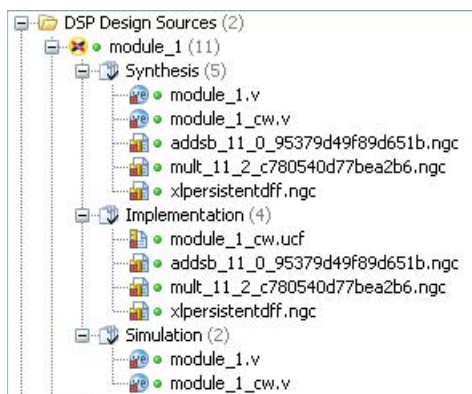
追加された DSP ソースが [Sources] ビューの [IP Sources] タブにそれぞれ表示されます。DSP ソースは、[Hierarchy]、[Libraries]、[Compile Order] タブのその他のソース ファイルと一緒に表示されます。[Sources] ビューで DSP モジュールを選択すると、それに関連するファイルが表示され、

[Source File Properties] ビューにはそのプロパティが表示されます。詳細は、[144 ページの「\[Sources\] ビュー」](#)を参照してください。

ターゲットの生成

System Generator デザインが終了したら、[Sources] ビューのポップアップ メニューから DSP モジュールのコマンドを使用し、PlanAhead ツール内で FPGA ターゲット ファイルを生成できます。DSP モジュール コマンドは、DSP ソースを選択すると、[Sources] ビューのポップアップ メニューに表示されるようになります。

ターゲットとは、現在のデザインの合成、シミュレーション、インプリメンテーションをサポートするのに必要な DSP モジュールのさまざまなデザイン エレメントのことです。これには、最上位モジュールの定義、インスタンシエーション テンプレート、合成済みネットリスト、その他の関連資料が含まれます。



- [Create Top HDL] : DSP モジュールの最上位ラッパー ファイルを作成し、プロジェクトに追加します。このコマンドは、System Generator デザインが現在のプロジェクトの最上位の場合に使用します。
- [View Instantiation Template] : DSP モジュールを RTL デザインにインスタンシエートするために使用するインスタンシエーション テンプレートを作成します。インスタンシエーション テンプレートは別の RTL ファイルにコピーして貼り付けると、その階層で DSP モジュールのインスタンスを作成できます。
- [Create Testbench] : Simulink シミュレーションから抽出したテスト ベクター ファイルが System Generator で書き出され、シミュレーション用の HDL テストベンチとスクリプト ファイルが生成されます。テストベンチは、シミュレーション ファイルセットの [Sources] ビューに追加されます。
- [Generate] : System Generator モデルから合成、インプリメンテーション、およびシミュレーションのターゲット データが生成されます。これにより、System Generator および MATLAB が起動し、必要なデータが作成されます。
- [Reset] : 現在のプロジェクトおよびローカル プロジェクト レポジトリから指定したターゲット データを削除します。ターゲット データは、必要に応じて再生成します。

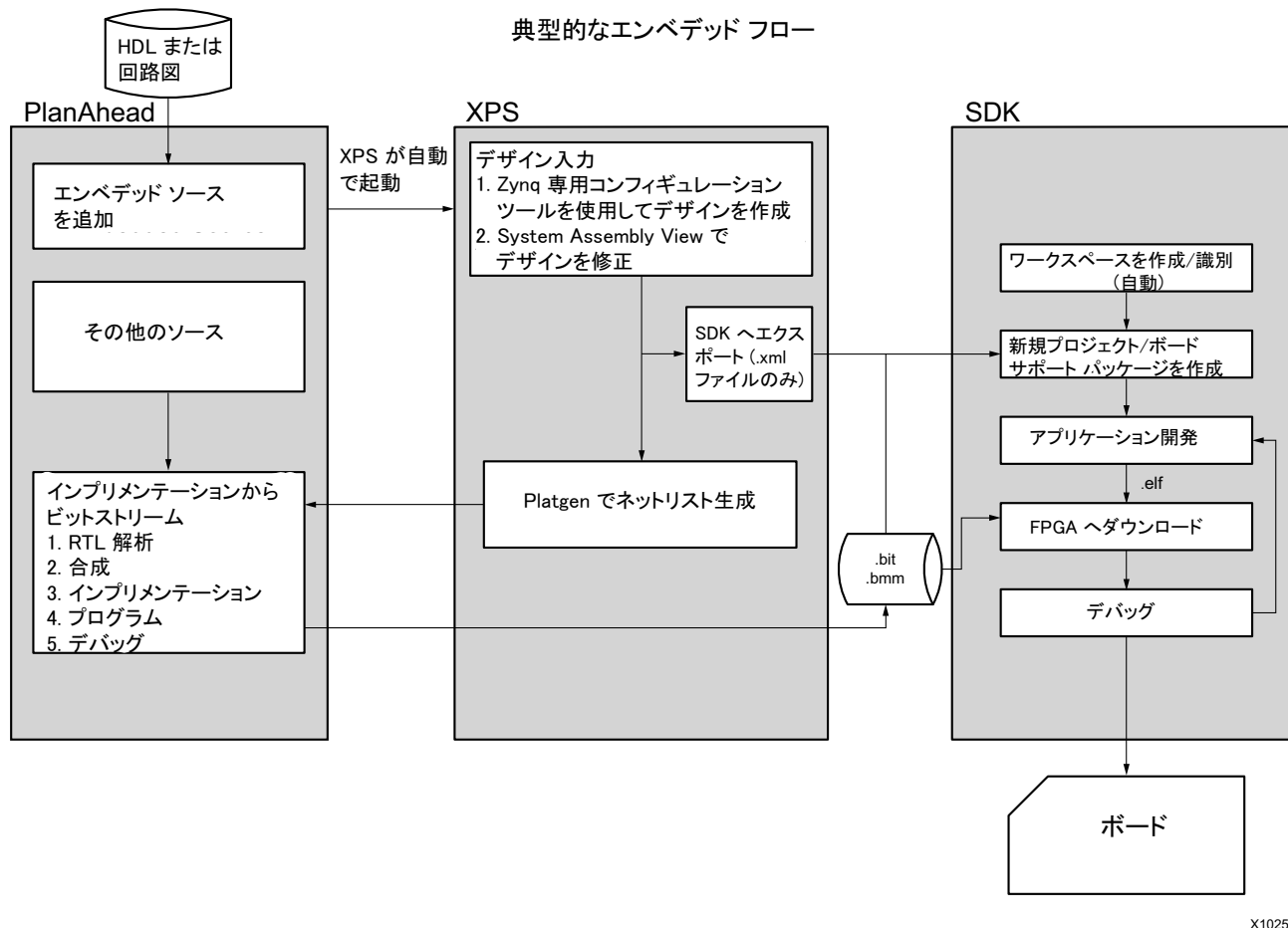
エンベデッド プロセッサ ソースの管理

エンベデッド開発キット (EDK) は、ユーザーのハードウェアおよびソフトウェア システム コンポーネントに統合させるために使用可能なツールおよび IP の総称です。EDK には、Xilinx Platform Studio (XPS) および Software Development Kit (SDK) という 2 つのツールが含まれます。

エンベデッド プロセッサ システムのハードウェア部分を設計するには、Xilinx Platform Studio (XPS) を使用します。XPS では、マイクロプロセッサおよびペリフェラルの仕様、これらのコンポーネントの接続およびプロパティを設定します。XPS の特定機能の使用に関する詳細は、[付録 E の「EDK 資料」](#)を参照してください。

EDK 環境ではデザインの作成およびインプリメンテーションがサポートされますが、PlanAhead ツールでプロジェクトを開始して、XPS を使用してプロジェクトのエンベデッド プロセッサ ソー

スを開発することをお勧めします。こうすることで、PlanAhead ツールで FPGA デザインのプロジェクトを管理できるので、エンベデッド プロセッサ デザインを XPS 内で開発および管理しておいて、PlanAhead で 1 つのソース ファイルとして処理できます。88 ページの図 3-33 は、この統合エンベデッド デザイン フローを示しています。



X10254

図 3-33 : エンベデッド デザイン フロー

エンベデッド プロセッサの追加

PlanAhead では、既存の Xilinx Microprocessor Project (.xmp) ファイルを XPS からインポートしたり、PlanAhead ツール内で新しいエンベデッド プロセッサ サブデザインを定義しておいてから、XPS を開いてプロジェクトを作成および管理したりできます。

XPS プロジェクトを追加または作成するには、次の手順に従います。

1. [File] → [Add Sources] をクリックするか、ポップアップ メニューまたは Flow Navigator から [Add Sources] をクリックします。

61 ページの図 3-14 に示す Add Sources ウィザードが表示されます。

2. [Add or Create Embedded Sources] をオンにし、[Next] をクリックします。

89 ページの図 3-34 に示す [Add or Create Embedded Sources] ページが表示されます。

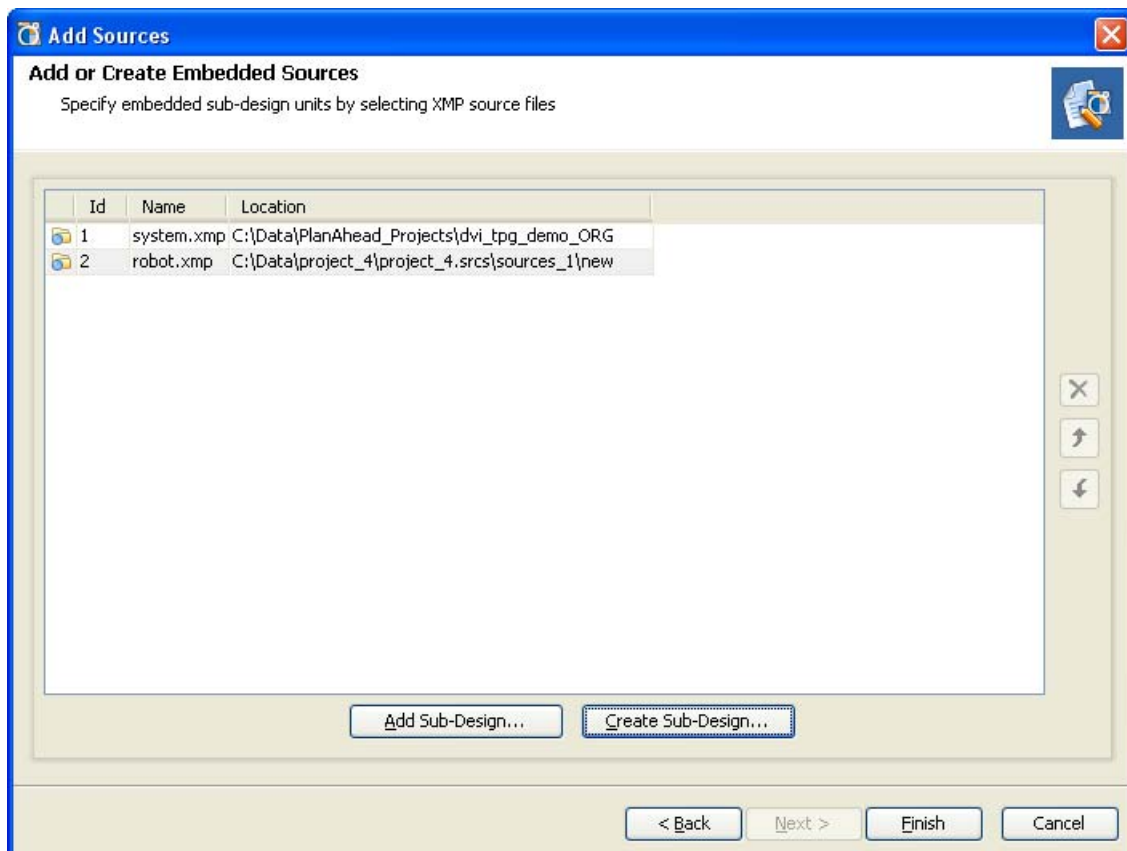


図 3-34 : エンベデッド ソースの追加

- [Add Sub-Design] : 既存の XPS プロジェクトを現在のプロジェクトに追加します。追加する既存の Xilinx Microprocessor Project (.xmp) ファイルは、表示されるファイル ブラウザから選択できます。
- [Create Sub-Design] : 新しく XPS プロジェクトを作成して PlanAhead ツールプロジェクトに追加します。これにより XPS が起動するので、新しいサブデザインを定義します。
- [Remove] : 選択したサブデザインをリストから削除します。
- [Move Up] : 選択したサブデザインをリストの上方向に移動します。
- [Move Down] : 選択したサブデザインをリストの下方向に移動します。

プロジェクトに追加または作成する XPS プロジェクトを選択したら、[Finish] をクリックしてファイルを追加します。

追加されたサブデザインは、[Hierarchy]、[Libraries]、[Compile Order] タブのその他のソース ファイルと一緒に、[Sources] ビューの [IP Sources] タブにそれぞれ表示されます。[Sources] ビューにはサブデザインを構成するファイルが表示され、サブデザインを選択すると [Source File Properties] ビューにプロパティが表示されます。詳細は、144 ページの「[Sources] ビュー」を参照してください。

サブデザインの作成

[Add or Create Embedded Sources] ダイアログ ボックスで [Create Sub-Design] を選択すると、XPS が起動し、新しいエンベデッド サブデザインを定義できるようになります。ターゲット パーツや TDP などの PlanAhead ツールのプロジェクトの属性は、XPS を開くと自動的に移行されます。

次は、エンベデッド デザインを定義するプロセスの簡単な概要です。詳細は、付録 E の「EDK 資料」に示す『EDK のコンセプト、ツール、テクニック』(UG683) および『エンベデッド システム ツール リファレンス マニュアル』(UG111) を参照してください。

XPS では、これが新しいサブデザインであることが認識され、ボード デザインのしやすい Base System Builder ウィザードを起動するかどうか尋ねるメッセージが表示されます。

1. [Yes] をクリックして続行します。

図 3-35 に示すような Base System Builder (BSB) ウィザードが表示されます。BSB ウィザードを使用すると、素早くシステムを構築できます。エンベデッド デザイン プロジェクトの中には、BSB ウィザードだけで完成できるものもあります。複雑なプロジェクトの場合は、BSB ウィザードで作成したものをベースに、エンベデッド デザインをカスタマイズしていきます。

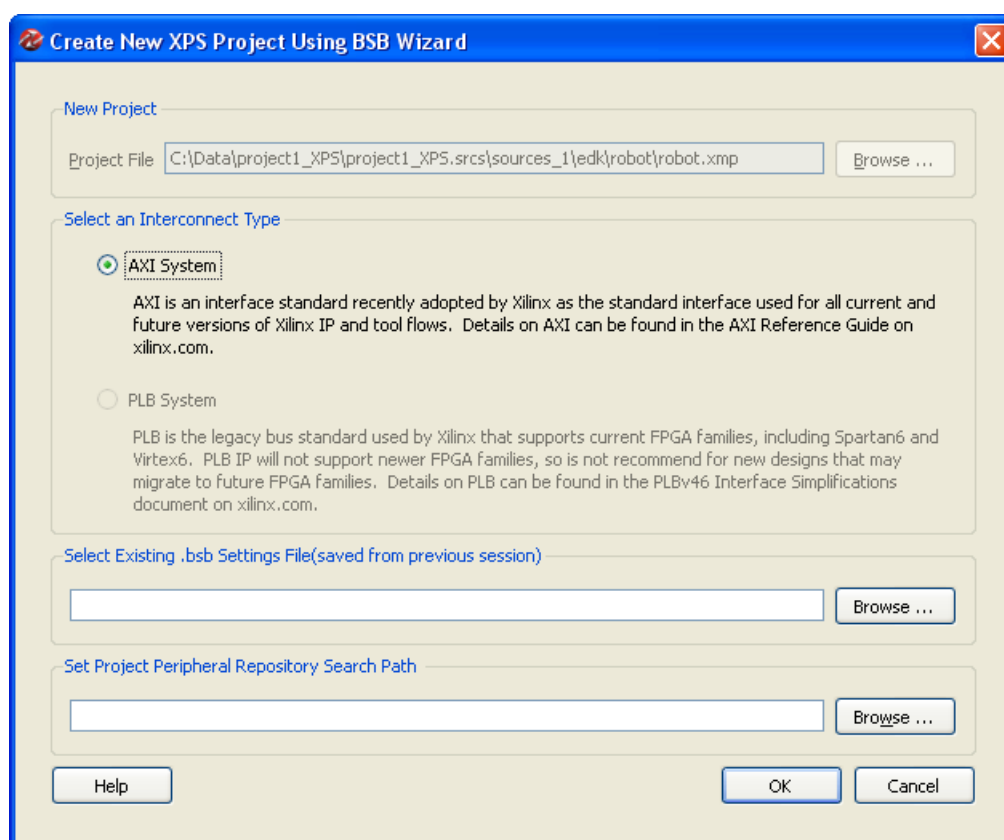


図 3-35 : Base System Builder

ほとんどのフォームに現在のプロジェクトからのデータが入力され、PlanAhead プロジェクトと XPS プロジェクトを連動させるために、変更できなくなっています。

- [Project File] : [Create Sub-Design] ダイアログ ボックスで指定したサブデザインの名前が表示されます。この名前は、PlanAhead ツールからインポートされたものです。
- [Select an Interconnect Type] : AXI System を指定します。これはレガシー システムを回避するためにハード コードされますので、変更できません。

- [Select Existing .bsb Settings File] : 以前の **BSB** 設定ファイルを指定して、同じ選択がこの **BSB** セッションでも自動的に使用されるようにします。
- [Set Project Peripheral Repository Search Path] — カスタム **pcore** を含むユーザー レポジトリ、**Board Support Packages (BSPs)**、およびソフトウェア サービスを指定します。レポジトリ検索パスを複数指定する場合は、セミコロン (;) で区切ってください。

2. [OK] をクリックします。

図 3-36 のような [Board and System Selection] ページが表示されます。このページでは、エンベデッド プロセッサ デザインの **TDP** またはプラットフォームを定義します。

提供される **TDP** の定義は、**PlanAhead** ツールのプロジェクトで選択したターゲット パーツを含んだものに制限されます。**BSB** には、特定の **FPGA** デバイス、外部メモリ、**I/O** デバイス、クロック リソース、リセット極性など、どのデバイスがターゲット ボードにあるかを決定する機能があります。次のウィザード ページは、選択したボードからの情報を元にカスタマイズされており、必要な入力 が最小限で済むようになっています。

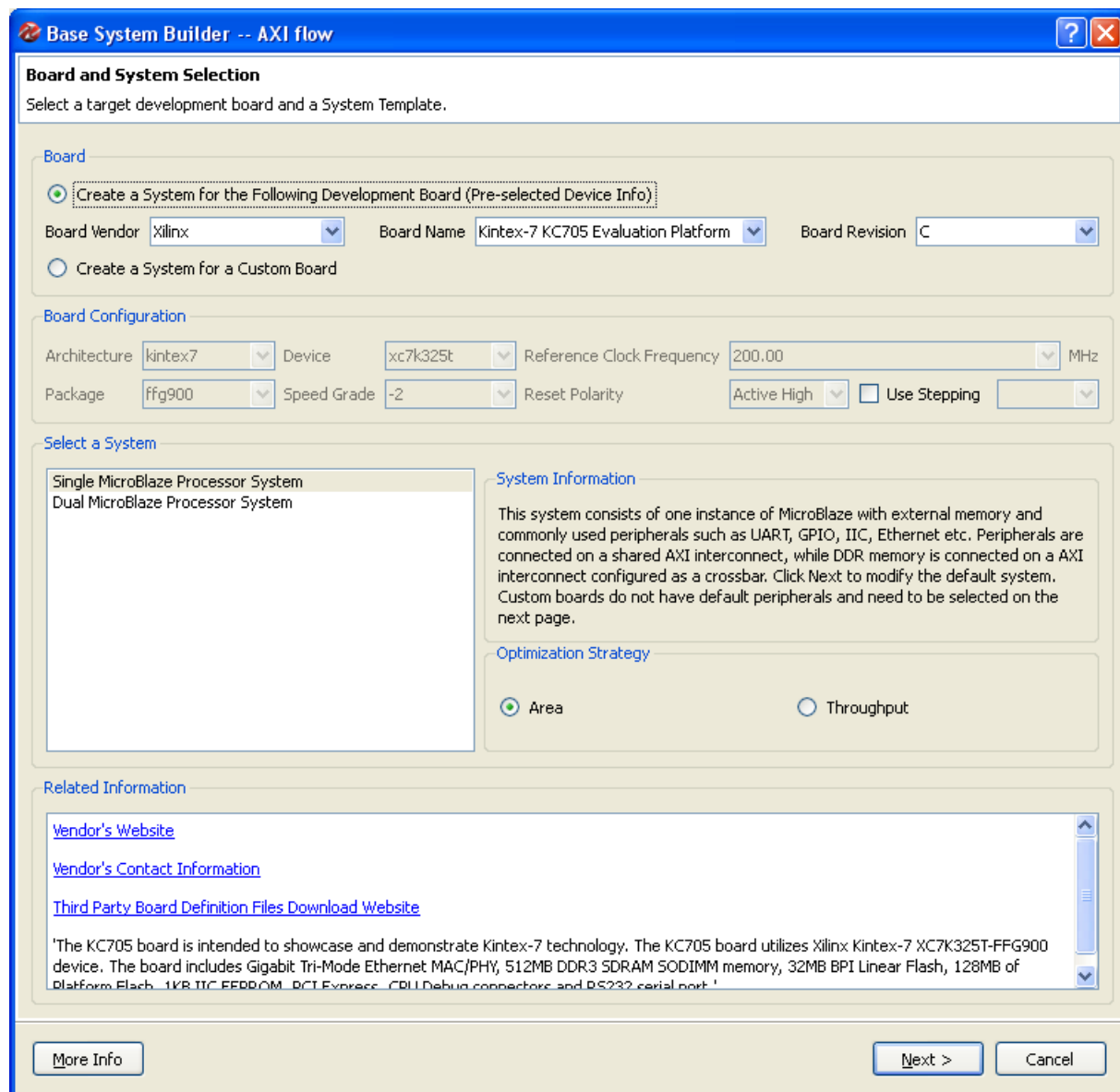


図 3-36 : [Board and System Selection] ページ

選択した **FPGA** がサポートされるボードで使用可能でない場合は、デバイスに対するボードが存在しないメッセージが表示されます。この場合は、[Create a System for a Custom Board] をオンにする必要があります。

3. [Next] をクリックします。

図 3-37 のような [Processor, Cache, and Peripheral Configuration] ページが表示されます。このページには、指定した TDP で使用可能なペリフェラルがリストされます。エンベデッド デザインに含めるペリフェラルを選択してください。

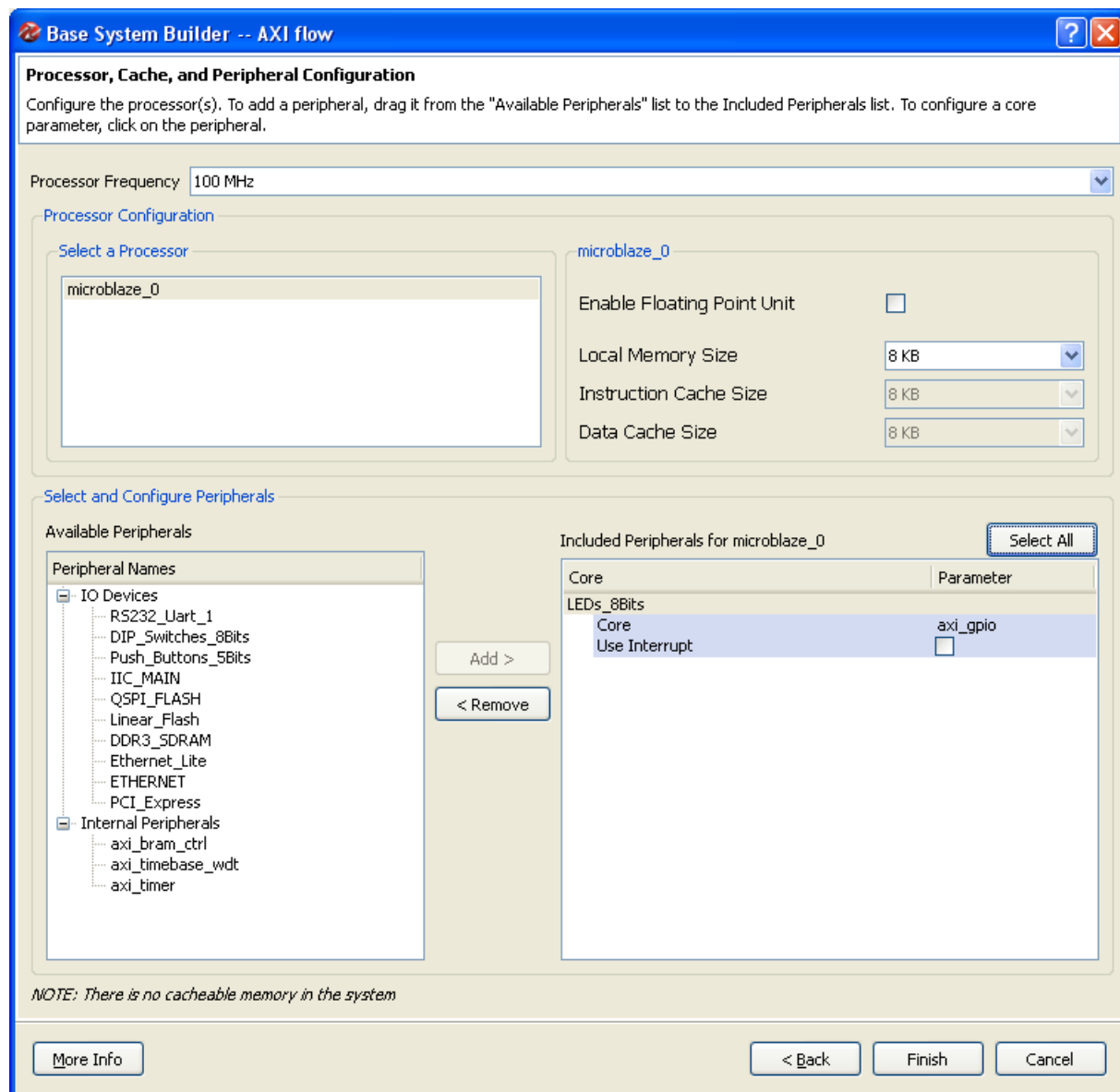


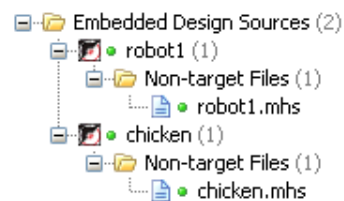
図 3-37 : [Processor, Cache, and Peripheral Configuration] ページ

4. [Finish] をクリックしてエンベデッド デザインを作成して、XPS で開きます。

エンベデッド プロセッサ サブデザインは、XPS ツールで変更および管理できます。

ターゲットの生成

XPS ツールを閉じると、最上位プロジェクト デザイン ファイル (.xmp) および Microprocessor Hardware Specification (.mhs) ファイルが PlanAhead の [Sources] ビューに追加されます。[Sources] ビューでエンベデッド デザイン ソースを展開表示すると、そのサブデザインに関連するさまざまなターゲット ファイルが表示されます。



ターゲットとは、現在のプロジェクトのオブジェクトをサポートするのに必要な XPS サブデザインのさまざまなデザイン エLEMENT のことです。これには、最上位モジュール定義、インスタンス化 シェーン テンプレート、合成済みネットリスト、ログ ファイルやデータシートなどの関連資料などが含まれます。エンベデッド デザインのプロジェクトファイル (.xmp) は、[Sources] ビューの [Hierarchy] タブに表示されます。

[Sources] ビューのエンベデッド ソースに対しては、次のようなポップアップ メニューが表示されます。

- [Create Top HDL] : エンベデッド デザインの最上位ラッパー ファイルを作成し、プロジェクトに追加します。このコマンドは、エンベデッド デザインが現在のデザインの最上位の場合に使用します。
- [View Instantiation Template] : エンベデッド デザイン モジュールを RTL デザインにインスタンス化するために使用するインスタンス化 シェーン テンプレートを作成します。インスタンス化 シェーン テンプレートは別の RTL ファイルにコピーして貼り付けると、その階層でサブデザインのインスタンスを作成できます。

注記 : テンプレート ファイルはプロジェクトに追加されません。

- [Create Testbench] : エンベデッド デザインのテストベンチを作成します。テストベンチは、シミュレーション ファイルセットの [Sources] ビューに追加されます。
- [Generate] : 合成、インプリメンテーション、シミュレーション用に指定したターゲット データを作成します。ターゲット データには、そのサブデザインの Verilog または VHDL ファイル、ラッパー ファイル、BMM モデル、最上位シミュレーション モデルが含まれます。
- [Reset] : 現在のプロジェクトから指定したターゲット データを削除します。ローカル プロジェクト レポジトリからも生成されたターゲット データを削除します。ターゲット データは、必要に応じて再生成します。

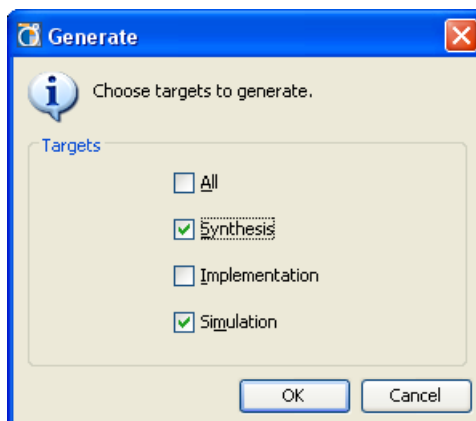


図 3-38 : エンベデッド ソース ターゲットの生成

ターゲット データを生成すると、そのエンベデッド サブデザインに対して /synthesis および /implementation ディレクトリが作成されます。これは、XPS で [Hardware] → [Generate Netlist] または [Hardware] → [Generate Bitstream] をクリックしても作成されます。

既存の XPS エンベデッド デザイン ソースを追加すると、/synthesis および /implementation ディレクトリはエンベデッド デザインのサブディレクトリ (プロジェクトの外部) に作成されますが、[Create Sub-Design] コマンドでエンベデッド プロセッサ デザインを現在のプロジェクトに追加すると、/synthesis および /implementation サブディレクトリがローカル プロジェクト ディレクトリ (<project>.srcs\sources_1\edk\<subdesign_name>) に保存されます。

- /synthesis サブディレクトリには、インプリメンテーションで使用するネットリストを作成するための XST 合成スクリプト (SCR)、プロジェクト ファイル (PRJ)、およびレポート (SRP) ファイルが含まれます。
- /implementation サブディレクトリには、ユーザー制約ファイル (UCF)、デバイス上のブロック RAM をコンフィギュレーションするためのブロック RAM メモリ マップ (BMM) ファイル、インプリメンテーション結果、BIT ファイルが含まれます。

エンベデッド プロセッサのインプリメント

最上位デザインのインプリメンテーションを実行する際は、次の MAP オプションを設定する必要があります。

- -pr b: 内部フリップフロップを IOB の入力 (i)、出力 (o)、または両方 (b) に挿入します。
- -ol high: 総体的なエフォート レベルを high に設定します。
- -timing: タイミング ドリブン パックを実行します。

インプリメンテーション run の [Implementation Settings] のこれらのオプションを設定し、プロセッサ IP コアが問題なく動作するようにします。詳細は、第 9 章の「インプリメンテーション オプションの設定」を参照してください。

ハードウェアのエクスポート

PlanAhead ツールは SDK とも統合されており、プロジェクト内のエンベデッド プロセッサ ソースのソフトウェア デザインがサポートされます。SDK を使用してエンベデッド プロセッサを含むプロジェクトのソフトウェアを開発するには、[File] → [Export] → [Export Hardware] をクリックします。

図 3-39 に示す [Export Hardware] ダイアログ ボックスが開きます。

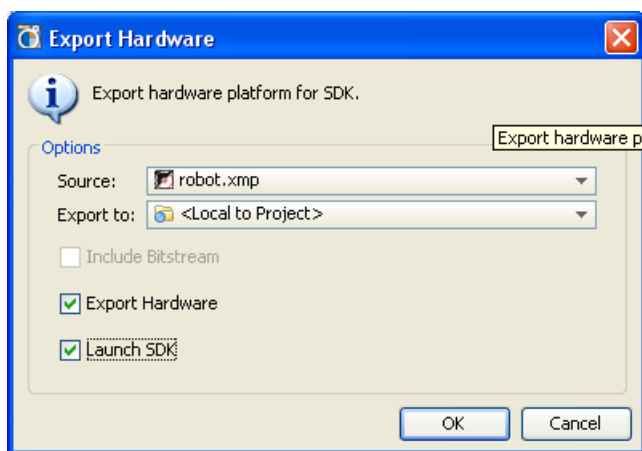


図 3-39 : [Export Hardware] ダイアログ ボックス

エクスポートするソースの XPS プロジェクト ファイルと、ハードウェアのエクスポート先のディレクトリを指定します。デフォルトでは、ハードウェア ファイルは次のローカルのプロジェクト ディレクトリに書き込まれます。

<project>.sdk/SDK/SDK_Export/hw

[Export Hardware] チェック ボックスをオンにします。これにより、エンベデッド プロセッサ デザインのソフトウェア開発をサポートするのに必要なファイルが生成されます。

[Launch SDK] チェック ボックスをオンにすると、ハードウェア ファイルの生成後に SDK ツールをオプションで起動できます。

[OK] をクリックします。

PlanAhead デザイン ツールで [Launch SDK] をオンにしておくと、デザインの Hardware Platform Specification (system.xml) ファイルがエクスポートされ、SDK が起動します。詳細は、[SDK ヘルプ](#) を参照してください。

ビットストリーム ファイルの作成

エンベデッド プロセッサ システムを起動するには、システムのハードウェアおよびソフトウェア コンポーネントの両方を FPGA にダウンロードして、メモリをそれぞれプログラムする必要があります。このためには、ブロック RAM をターゲットとしたソフトウェア アプリケーションを含むビットストリーム ファイルを BitGen で作成する必要があります。BitGen で作成したハードウェア ビットストリームとエンベデッド プロセッサに関連するソフトウェアの Executable and Linkable Format (ELF) ファイルをダウンロードすると、ISE の iMPACT ツールを使用して、ビットストリームを含めて FPGA をプログラムできます。

プロセッサ インスタンスに関連する ELF ファイルは、メイン メニューから [Tools] → [Associate ELF Files] をクリックすると、追加またはアップデートできます。

図 3-40 のような [Associate ELF Files] ダイアログ ボックスが表示され、ビットストリーム ファイルの生成時に使用する ELF ファイルを指定できます。

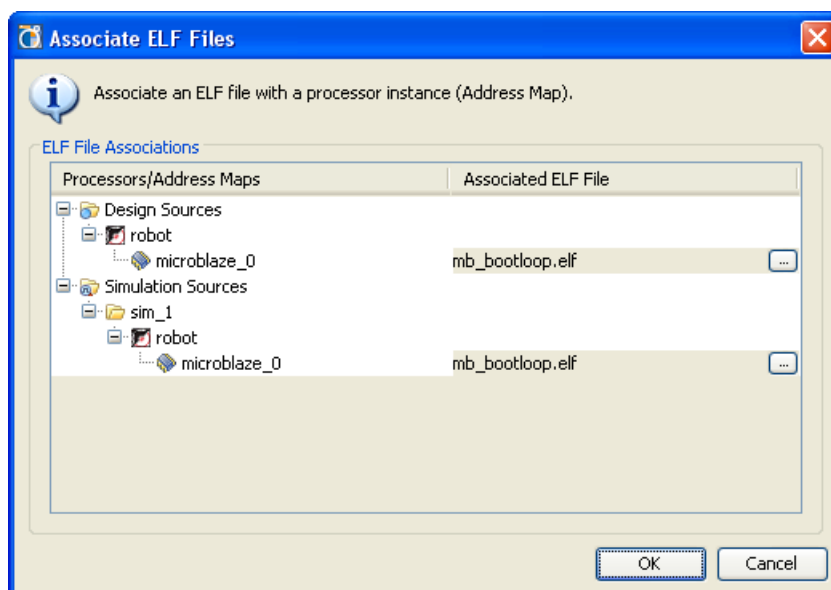


図 3-40 : [Associate ELF Files] ダイアログ ボックス

ELF ファイルは .bmm ファイルで指定したブロック RAM を生成します。PlanAhead で生成されたこの BIT ファイルには、選択した実行コードで初期化されたブロック RAM が含まれます。

プロセッサに接続されたオンチップ ブロック RAM メモリをソフトウェア情報を含めて初期化するには、Bitinit ツールが使用されます。このユーティリティは、ISE ツールで生成されたハードウェアのみのビットストリーム (system.bit) を読み込んで、各プロセッサのエンベデッド アプリケーション実行ファイルを含む新しいビットストリーム (download.bit) を出力します。詳細は、付録 E の「EDK 資料」を参照してください。

[Project Summary] ビュー

PlanAhead には、デザインおよびプロジェクトの情報を表示する [Project Summary] ビューがあります。このビューは、デザイン コマンドが実行されると随時アップデートされます。

[Project Summary] ビューには、プロジェクト パーツ、プロジェクトのステータス、合成およびインプリメンテーションの状態、リソース予測、タイミング結果などのデザイン情報が表示されます。起動コマンドへのリンクや、より詳細な情報を表示するためのリンクもあります。スクロール バーを使用したり、[Collapse All] および [Expand All] ボタンを使用してデータ カテゴリの表示/非表示を切り替えることができます。

[Project Summary] ビューを開くには、次のいずれかの方法を使用します。

- Project Manager メニューから [Project Summary] をクリックします。
- ツールバーの [Project Summary] ボタンをクリックします。
- [Window] → [Project Summary] をクリックします。

図 3-41 に、[Project Summary] ビューを示します。

[Project Settings] セクション

[Project Settings] セクションには、プロジェクト名、プロジェクト パーツ、デフォルト デバイス、および最上位モジュール名が表示されます。[Edit] リンクをクリックすると、[Project Settings] ダイアログ ボックスが開きます。詳細は、101 ページの「プロジェクト設定」を参照してください。

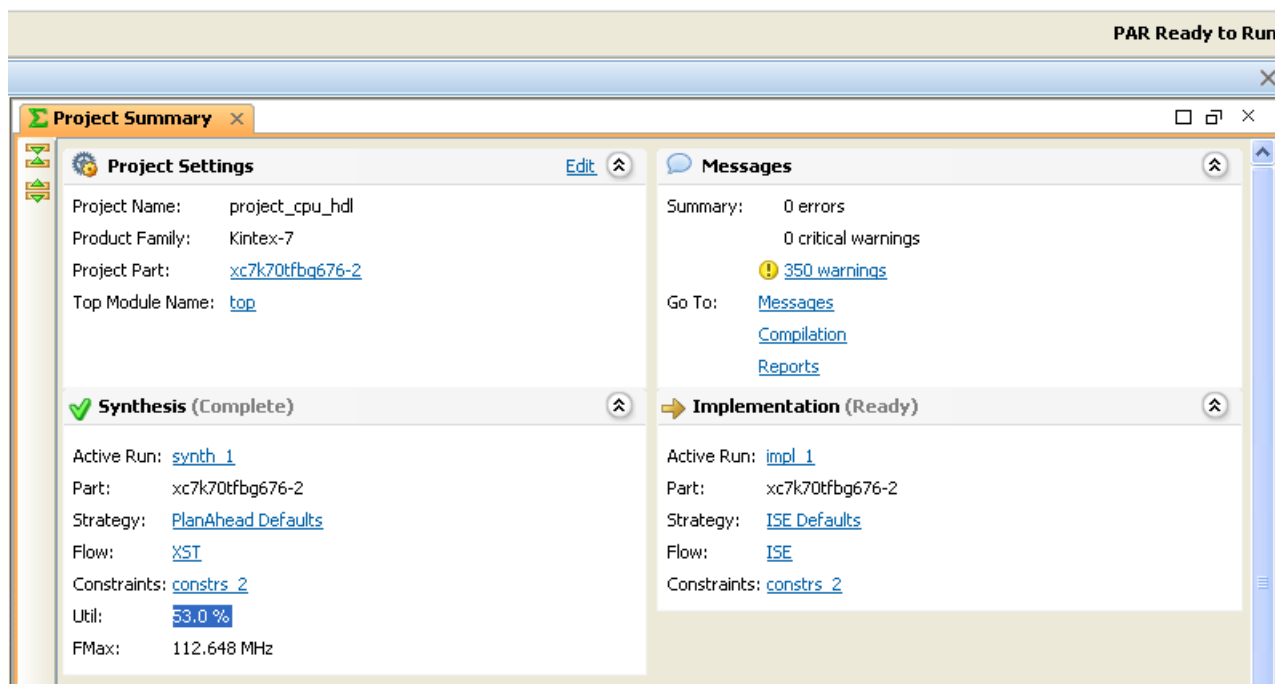


図 3-41 : [Project Summary] ビュー

[Messages] セクション

[Messages] セクションには、次の情報が表示されます。

- **[Summary]** : デザイン プロセス中に発生したエラーおよび警告の数を示します。警告およびエラーのみを表示した **[Messages]** ビューを開くリンクもあります。
- **[Go To]** : **[Messages]**、**[Compilation]**、または **[Reports]** ビューへのリンクを表示します。詳細は、第 4 章の「[Tcl コンソールおよびメッセージ エリアの使用](#)」を参照してください。

[Synthesis] セクション

アクティブ **run** の合成の状態のサマリを表示します。ターゲット パーツ、**run** で適用されたストラテジ、リソース使用率、最大周波数 (Fmax) などが表示されます。

[Implementation] セクション

アクティブ **run** のインプリメンテーションの状態のサマリを表示します。ターゲット パーツ、**run** で使用されたストラテジ、使用されたツール フローおよび制約セットなどが表示されます。

ターゲット デバイスまたはストラテジのリンクをクリックすると、**[Project Settings]** ダイアログボックスが開きます。詳細は、101 ページの「[プロジェクト設定](#)」を参照してください。

[Resources] セクション

ターゲット デバイスのリソース使用量は、[図 3-42](#) に示すようにグラフで、または [100 ページの図 3-43](#) に示すように表形式で表示されます。**[Project Summary]** の **[Resources]** セクションの右上のリンクをクリックすると、グラフと表の表示を切り替えることができます。

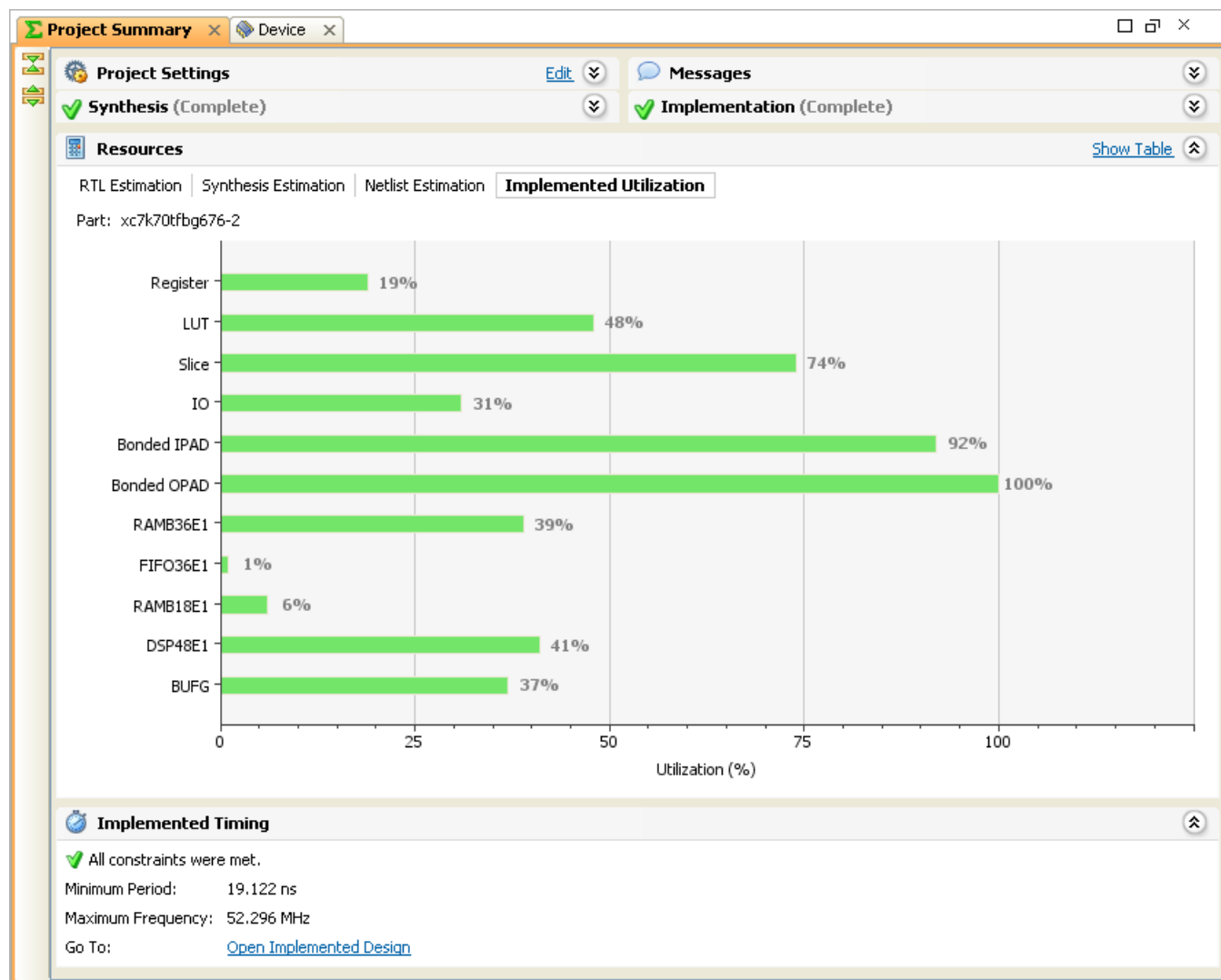
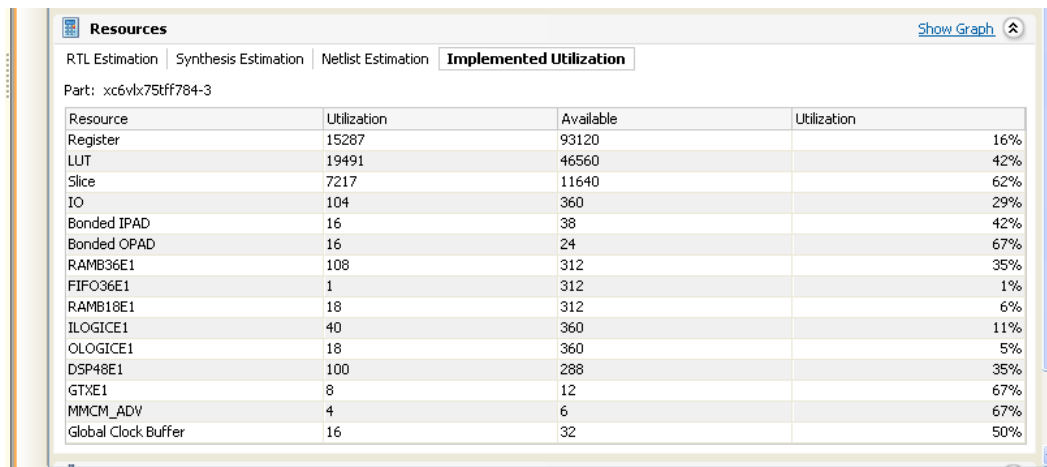


図 3-42 : リソース予測のグラフ

[Project Summary] ビューの [Resources] セクションは、デザインプロセスの各段階でアップデートされます。表示されるロジックオブジェクトのタイプは、デザインの段階によって異なります。情報が参照できるようになると、上部にある次のタブが選択できるようになります。

- [RTL Estimation] : エラボレート済みデザインからのリソース予測で、[Resource Estimation] コマンドを実行した後に表示されます。
- [Synthesis Estimates] : XST 合成レポートからのリソース予測を表示します。
- [Netlist Estimation] : 合成済みデザインからのリソース予測で、[Resource Estimation] コマンドを実行した後に表示されます。
- [Implemented Utilization] : ISE の MAP レポートからの実際のリソース使用率が表示されます。インプリメント済みデザインが開いている必要があります。



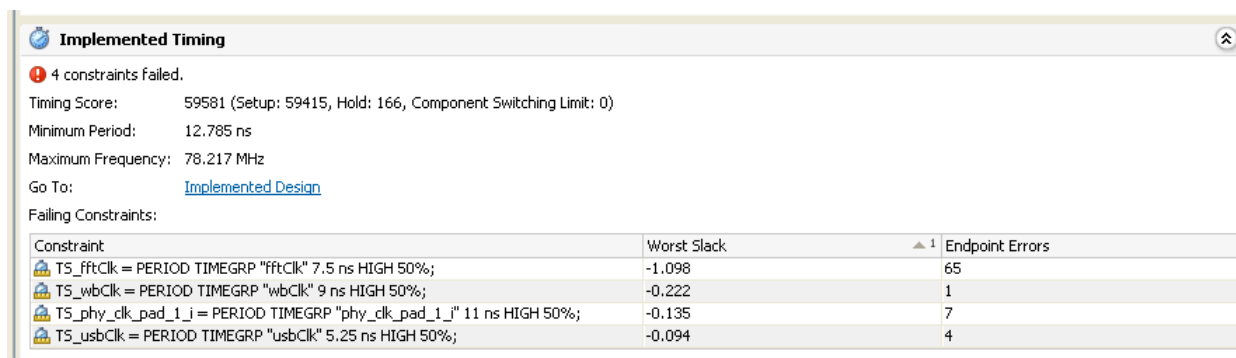
Resource	Utilization	Available	Utilization
Register	15287	93120	16%
LUT	19491	46560	42%
Slice	7217	11640	62%
IO	104	360	29%
Bonded IPAD	16	38	42%
Bonded OPAD	16	24	67%
RAMB36E1	108	312	35%
FIFO36E1	1	312	1%
RAMB18E1	18	312	6%
ILOGICE1	40	360	11%
OLOGICE1	18	360	5%
DSP48E1	100	288	35%
GTXE1	8	12	67%
MMCM_ADV	4	6	67%
Global Clock Buffer	16	32	50%

図 3-43：リソース予測の表

タブをクリックしたときに、リソース予測を表示するのに必要なコマンドを実行するリンクが表示されることもあります。[Netlist Estimation] を表示するには合成済みデザインを開いておく必要があり、[Implemented Utilization] を表示するには、インプリメント済みデザインを開いておく必要があります。

[Implemented Timing] セクション

[Implemented Timing] セクションには、[図 3-44](#) に示すように、インプリメンテーションの全体的なタイミング結果が表示されます。アクティブなインプリメンテーションからのタイミングスコア、最小周期、最大周波数、満たされていない制約 (最悪のもの)、およびインプリメント済みデザインと [Timing Results] ビューを開くリンクが示されます。タイミング解析の詳細は、[第 11 章「インプリメンテーション結果の解析」](#) を参照してください。



Constraint	Worst Slack	Endpoint Errors
TS_fftClk = PERIOD TIMEGRP "fftClk" 7.5 ns HIGH 50%;	-1.098	65
TS_wbClk = PERIOD TIMEGRP "wbClk" 9 ns HIGH 50%;	-0.222	1
TS_phy_clk_pad_1_j = PERIOD TIMEGRP "phy_clk_pad_1_j" 11 ns HIGH 50%;	-0.135	7
TS_usbClk = PERIOD TIMEGRP "usbClk" 5.25 ns HIGH 50%;	-0.094	4

図 3-44：[Project Summary] ビューのタイミング結果

[Partition Summary] セクション

パーティションが定義されたインプリメント済みデザインの場合は、[Project Settings] ビューにパーティションのステータスを示す [Partition Summary] セクションが表示されます。このセクションには、各パーティションのインプリメンテーションレポートと、ステート、保持レベル、境界最適化などの属性とその値が表示されます。

パーティションの定義および使用については、第 13 章「階層デザイン手法の使用」および付録 E「その他のリソース」に示す『パーシャル リコンフィギュレーション ユーザー ガイド』(UG702) を参照してください。

プロジェクト設定

各プロジェクトでは、必要に合わせてさまざまな設定を指定できます。プロジェクト設定には、最上位モジュールの定義および言語オプションに関する設定、シミュレーション設定、合成設定、インプリメンテーション設定、IP カタログ設定などが含まれます。

PlanAhead では、さまざまなビューおよびメニューから [Project Settings] ダイアログ ボックスにアクセスできます。[Project Settings] ダイアログ ボックスをどのように開いたかによって、関連したページが表示されます。たとえば、[IP Catalog] ビューで [IP Catalog Settings] をクリックすると、[Project Settings] ダイアログ ボックスの [IP Catalog] ページが表示されます。

[Project Settings] ダイアログ ボックスを表示するには、次のいずれかを実行します。

- ツールバーの [Project Settings] ボタンをクリック
- Flow Navigator で [Project Manager] → [Project Settings] をクリック
- [Tools] → [Project Settings] をクリックします。
- [Project Summary] ビューで [Edit] リンクをクリック



[Project Settings] ダイアログ ボックスが開き、左側に次のメニューが表示されます (102 ページの図 3-45)。

- [Project Settings] : プロジェクト名、最上位モジュール名、言語オプションを表示します。詳細は、102 ページの「[Project Settings] ダイアログ ボックスの [General] ページ」を参照してください。
- [Simulation] : シミュレーション セット、シミュレーション 最上位モジュール名、最上位モジュール (テスト中)、シミュレーション ランタイム、およびタブに分類された起動オプション、ネットリスト オプションなどを表示します。現在選択されているオプションには、緑のチェック マークが表示されます。詳細は、105 ページの「[Project Settings] ダイアログ ボックスの [Simulation] ページ」を参照してください。
- [Synthesis] : ターゲット パーツ、制約セット、合成ストラテジ、合成コマンド ライン オプションを表示します。コマンド ライン オプションは選択した合成ストラテジで定義されますが、これを変更できます。選択したコマンド ライン オプションの説明が、ダイアログ ボックスの下部に表示されます。詳細は、105 ページの「[Project Settings] ダイアログ ボックスの [Synthesis] ページ」を参照してください。
- [Implementation] : ターゲット パーツ、制約セット、インプリメンテーション ストラテジ、インプリメンテーションで実行される変換、マップ、配置配線、タイミング解析のオプションを表示します。コマンド ライン オプションは選択したインプリメンテーション ストラテジで定義されますが、これを変更できます。選択したコマンド ライン オプションの説明が、ダイアログ ボックスの下部に表示されます。詳細は、107 ページの「[Project Settings] ダイアログ ボックスの [Implementation] ページ」を参照してください。
- [Bitstream] : 使用するビットストリーム オプションを指定します。詳細は、108 ページの「[Project Settings] ダイアログ ボックスの [Bitstream] ページ」を参照してください。
- [IP Catalog] : 現在のザイリンクス IP カタログのディレクトリを表示し、追加したり読み込み直したりできます。詳細は、109 ページの「[Project Settings] ダイアログ ボックスの [IP Catalog] ページ」を参照してください。

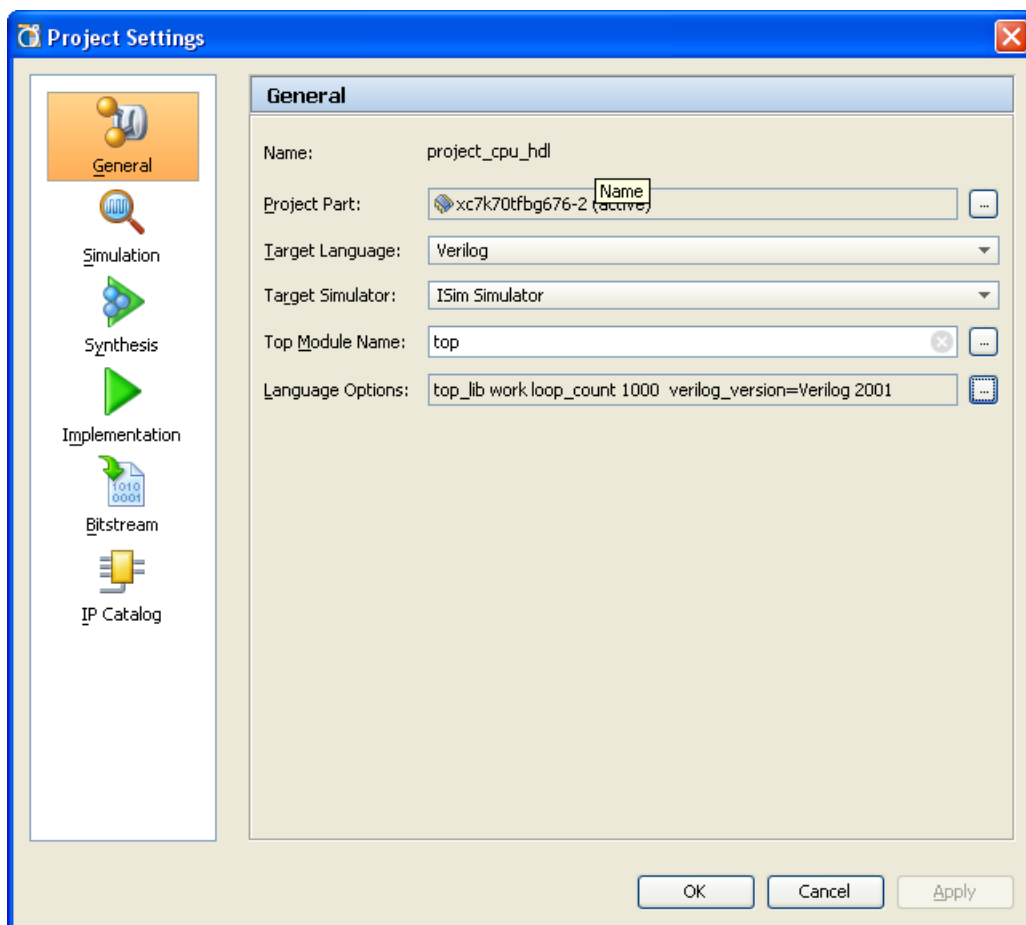


図 3-45 : [Project Settings] ダイアログ ボックスの [General] ページ

[Project Settings] ダイアログ ボックスの [General] ページ

図 3-45 に、[Project Settings] ダイアログ ボックスの [General] ページから [Language Options] ダイアログ ボックスを開いたところを示します。

- [Name] : プロジェクト名を表示します。
- [Project Part] : 合成およびインプリメンテーション両方でデフォルトとして使用するターゲット パーツを定義します。参照ボタンをクリックすると [Select Part] ダイアログ ボックスが表示され、デバイスを変更できます。ターゲット パーツの設定については、53 ページの「デフォルト パーツまたはボードの選択」を参照してください。

[Project Part] を使用すると、素早くプロジェクト全体のターゲット デバイスを定義または変更できますが、合成 run またはインプリメンテーション run が複数ある場合は、[Run Properties] ビューから run 設定を変更して特定 run で使用されるパーツを変更することもできます。デザインの run 属性の設定については、第 4 章の「[Run Properties] ビューの使用」を参照してください。

- [Target Language] : Verilog または VHDL のいずれかにデザインのターゲット出力言語を指定します。指定したターゲット言語でデザインの RTL 出力ファイルが生成されます。ターゲット言語で制御される出力の例は、合成、シミュレーション、最上位ラッパー ファイル、テストベンチ、および IP のインスタンス化テンプレートです。

- [Target Simulator] : シミュレーターをビヘイビア シミュレーションまたはタイミング シミュレーション用に起動するように指定します。使用できるオプションは次のとおりです。
- [ISim Simulator] : ターゲット シミュレーターをザイリンクス ISim に指定します。
- [QuestaSim/ModelSim] : ターゲット シミュレーターを Mentor Graphics® ModelSim または Questa® Advanced Simulator ツールに指定します。シミュレーション実行時にこれらのシミュレーターを起動させるようにするには、指定したシミュレーターはインストールされ、\$PATH で指定されている必要があります。

ModelSim/Questa の使用をサポートするためには、これらのシミュレーターで使用するようにザイリンクス シミュレーション ライブラリをコンパイルする必要があります。PlanAhead 内の Tcl コンソールまたは ISE Design Suite のコマンド プロンプトで `compxlib` と入力すると、Xilinx Simulation Library Compilation ウィザードが起動します。`compxlib` の使用方法については、ウィザードの [More Info] ボタンをクリックして ISE ヘルプを参照してください。

ライブラリがコンパイルされると、シミュレーターは `modelsim.ini` ファイルを使用してこれらのコンパイル済みライブラリを参照します。`modelsim.ini` ファイルは、デフォルトの初期化ファイルで、リファレンス ライブラリ パス、最適化、コンパイラー、シミュレーターの設定を指定する制御変数が含まれます。

`modelsim.ini` は、次のディレクトリに含まれます。

- `compxlib` の実行時に設定された `compxlib.compiled_library_dir` 変数で指定されるパス
- MODELSIM 環境変数で定義されたパス
- MGC_WD環境変数で定義されたパス

これらのディレクトリに `modelsim.ini` がない場合は、警告メッセージが表示されます。

- [Top Module Name] : デザインの最上位モジュールの名前を入力します。下位モジュール名を入力し、特定のモジュールに対して合成を実行することもできます。参照ボタンをクリックすると、最上位モジュールが自動的に検索され、最上位モジュールの候補が複数見つかった場合は、それらがリストされます。

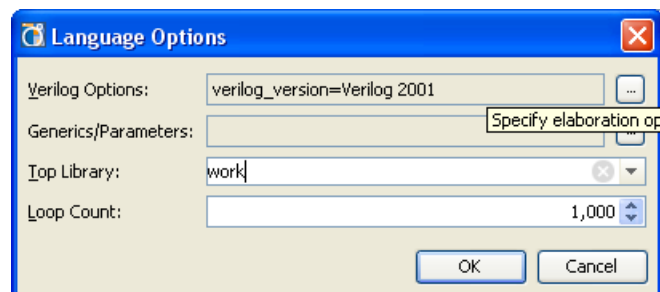


図 3-46 : [Language Options]

- [Language Options] : Verilog または VHDL オプションを入力します。これらフィールドは、図 3-46 に示されています。
- [Verilog Options] : 下記に定義されるとおりに Verilog オプションを指定します。これらのオプションは [Verilog Options] ダイアログ ボックスで定義します (104 ページの図 3-47)。
- [Verlog Search Paths] : Verilog ソース ファイルの `'include` 文で参照されるファイルの検索パスを指定します。

- [Defines] : プロジェクトの Verilog マクロ定義を指定します。
- [Uppercase all identifiers] : すべての Verilog 識別子を大文字にするように指定します。

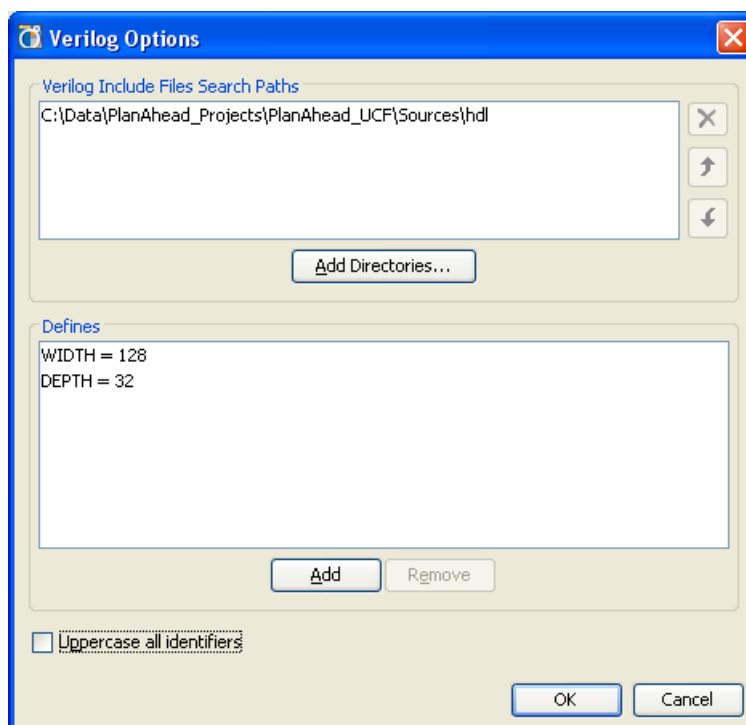


図 3-47 : Verilog オプション

- [Generics/Parameters] : VHDL ではジェネリックが、Verilog では定数値のパラメーター定義がサポートされます。どちらの方法でも、パラメーターを変更できるので、さまざまな状況で再利用可能です。このオプションを使用すると、ジェネリックおよびパラメーターの値を定義して、ソース ファイルで定義されたデフォルトの値を上書きできます。
- [Top Library] : 最上位モジュールのライブラリ名を指定します。
- [Loop Count] : 最大ループ反復値を指定します。デフォルトは 1000 です。

[Loop Count] オプションは、合成中ではなくエラボレーション中に使用されます。合成に対しては、[Synthesis] ページの [More Options] フィールドに「-loop_iteration_limit」と入力します。

[Project Settings] ダイアログ ボックスの [Simulation] ページ

図 3-48 に、[Project Settings] ダイアログ ボックスの [Simulation] ページを示します。

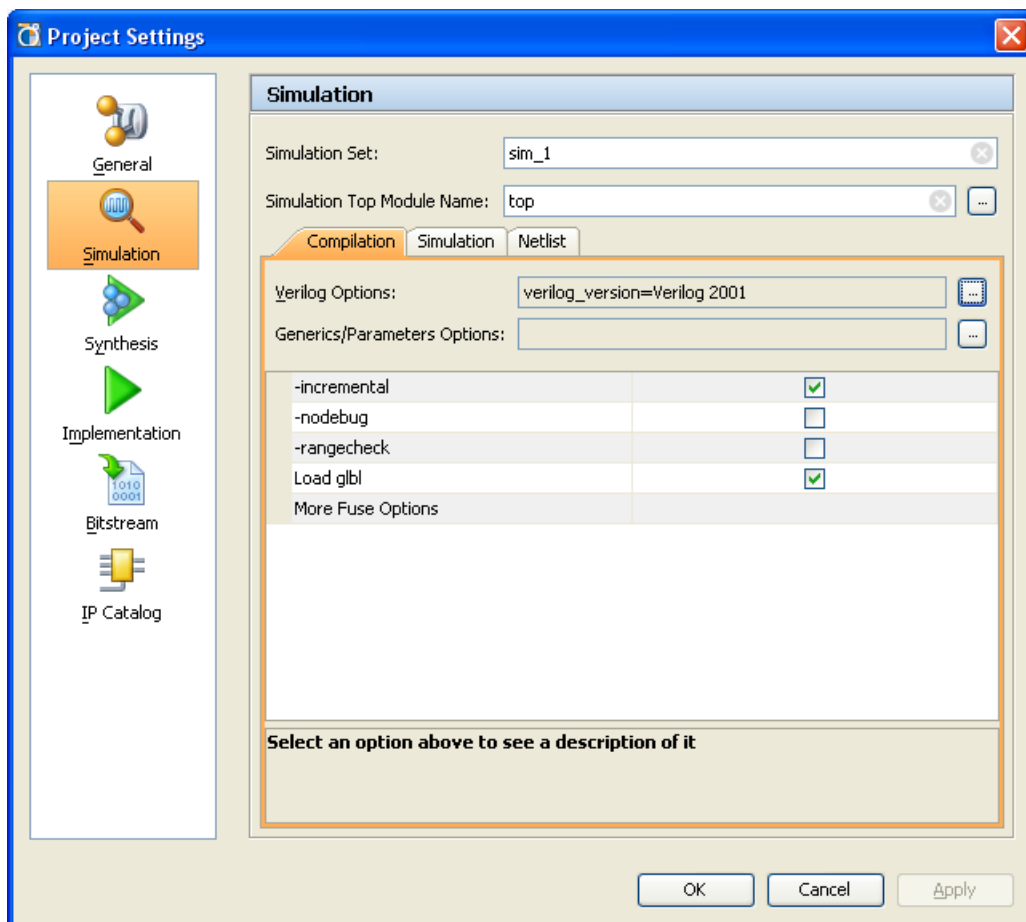


図 3-48 : [Project Settings] ダイアログ ボックスの [Simulation] ページ

シミュレーション プロジェクト設定の詳細は、397 ページの「タイミング シミュレーションの実行」を参照してください。

[Project Settings] ダイアログ ボックスの [Synthesis] ページ

106 ページの図 3-49 に、[Project Settings] ダイアログ ボックスの [Synthesis] ページを示します。

このページは Flow Navigator で [Synthesis Settings] コマンドをクリックすると表示されます。

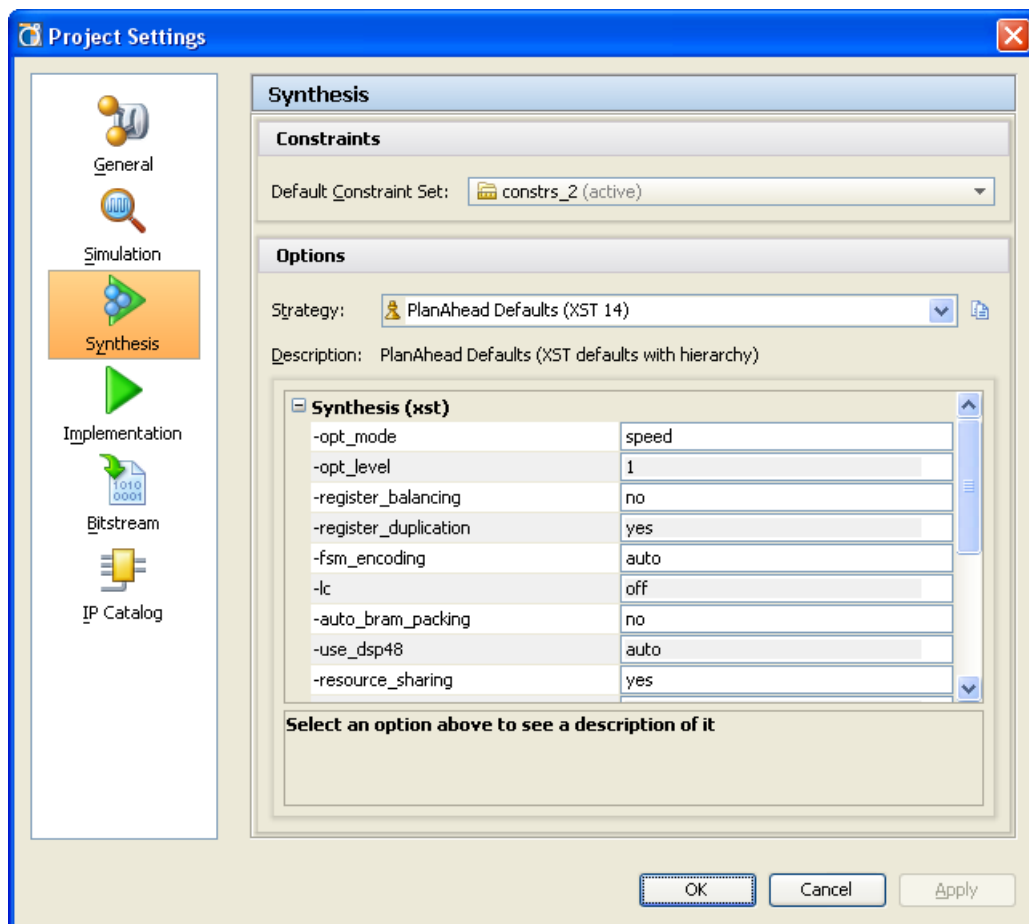


図 3-49 : [Project Settings] ダイアログ ボックスの [Synthesis] ページ

[Synthesis] ページには、次の情報が表示されます。

- **[Default Constraint Set]** : アクティブ制約セットを定義します。これは新しい run に使用される制約セットで、デザインを変更する際にも使用されます。制約セットの詳細は、[62 ページの「制約の管理」](#)を参照してください。
- **[Strategy]** : 合成に適用するストラテジを選択します。PlanAhead で提供されている定義済み合成ストラテジを使用するか、独自のストラテジを定義できます。詳細は、[第 4 章の「合成およびインプリメンテーション ストラテジの定義」](#)を参照してください。

合成ストラテジを選択すると、XST のコマンド ライン オプションがダイアログ ボックスの下に表示されます (図 3-49)。合成ストラテジの設定は、コマンド ライン オプションの値を変更すると上書きできます。

- **[Description]** : 選択したストラテジの説明を表示します。

合成オプションの設定については、[第 6 章「デザインの合成」](#)を参照してください。

[Project Settings] ダイアログ ボックスの [Implementation] ページ

図 3-50 に、[Project Settings] ダイアログ ボックスの [Implementation] ページを示します。

このページは Flow Navigator で [Implementation Settings] コマンドをクリックすると表示されます。

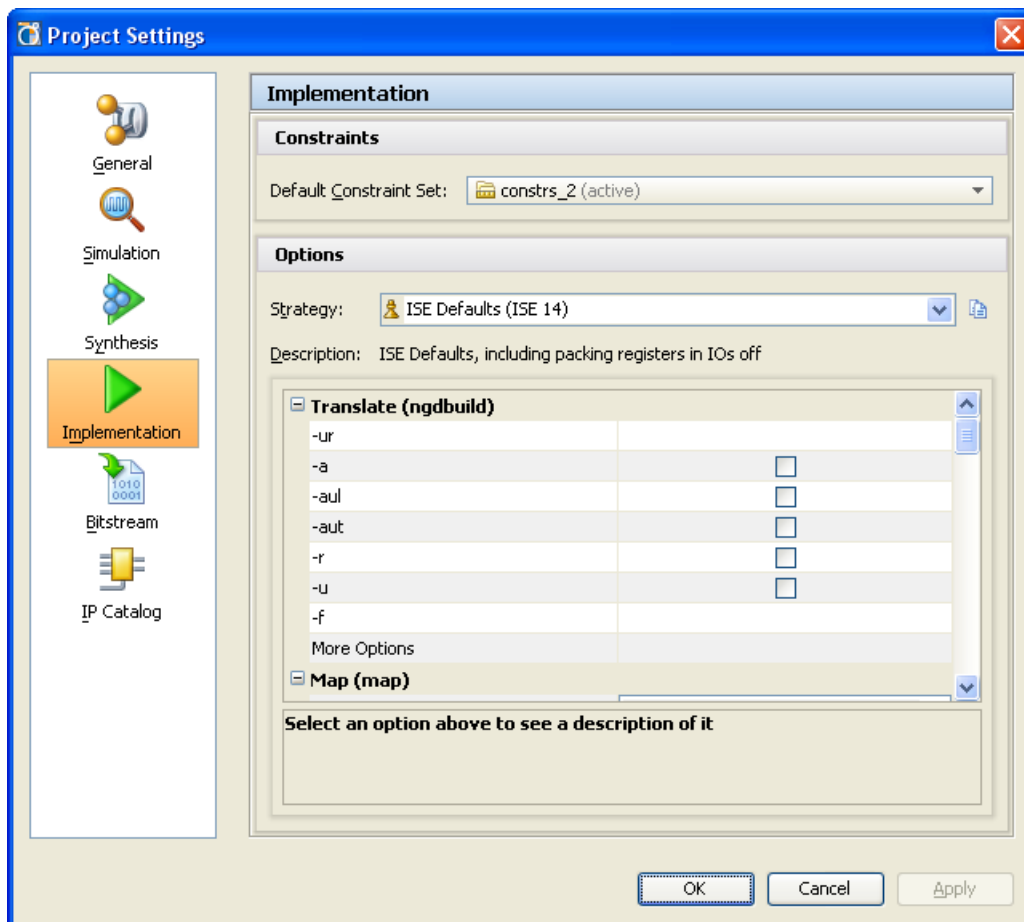


図 3-50 : インプリメンテーション プロジェクト設定

[Implementation] ページには、次の情報が表示されます。

- [Default Constraint Set] : インプリメンテーション run で使用する制約セットを選択します。
- [Strategy] : インプリメンテーションに適用するストラテジを選択します。定義済みのストラテジをリストから選択できますが、新しく作成することもできます。詳細は、[第 4 章の「合成およびインプリメンテーション ストラテジの定義」](#)を参照してください。

ストラテジを選択すると、ISE インプリメンテーション ツール (NGDBuild、MAP、PAR、TRCE) のコマンド ライン オプションがダイアログ ボックスの下に表示されます (図 3-50)。インプリメンテーション ストラテジ オプションは、コマンド ライン オプションの値を変更すると上書きできます。

- [Description] : 選択したインプリメンテーション ストラテジの説明を表示します。

インプリメンテーション オプションの設定については、[第 9 章「デザイン of インプリメンテーション」](#)を参照してください。

[Project Settings] ダイアログ ボックスの [Bitstream] ページ

インプリメンテーションが完了したら、ISE® の BitGen コマンドを実行して、ザイリンクス デバイスをプログラムするためのビットストリーム データを作成できます。

[Bitstream] ページでは、ビットストリームを生成する前に BitGen オプションを定義できます。オプションを選択すると、その説明も表示されます。BitGen のオプションについては、付録 E 「その他のリソース」に示す『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。

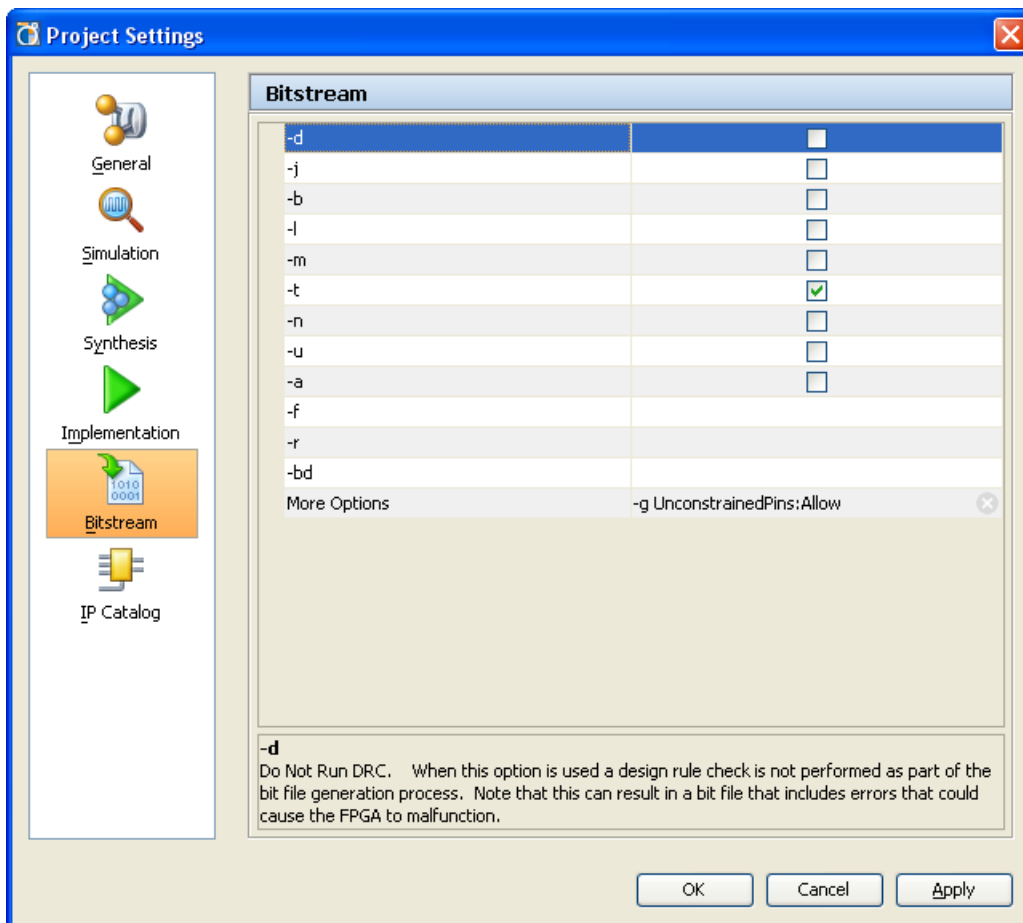


図 3-51 : [Project Settings] ダイアログ ボックスの [Bitstream] ページ

[Project Settings] ダイアログ ボックスの [IP Catalog] ページ

図 3-52 に、[Project Settings] ダイアログ ボックスの [IP Catalog] ページを示します。

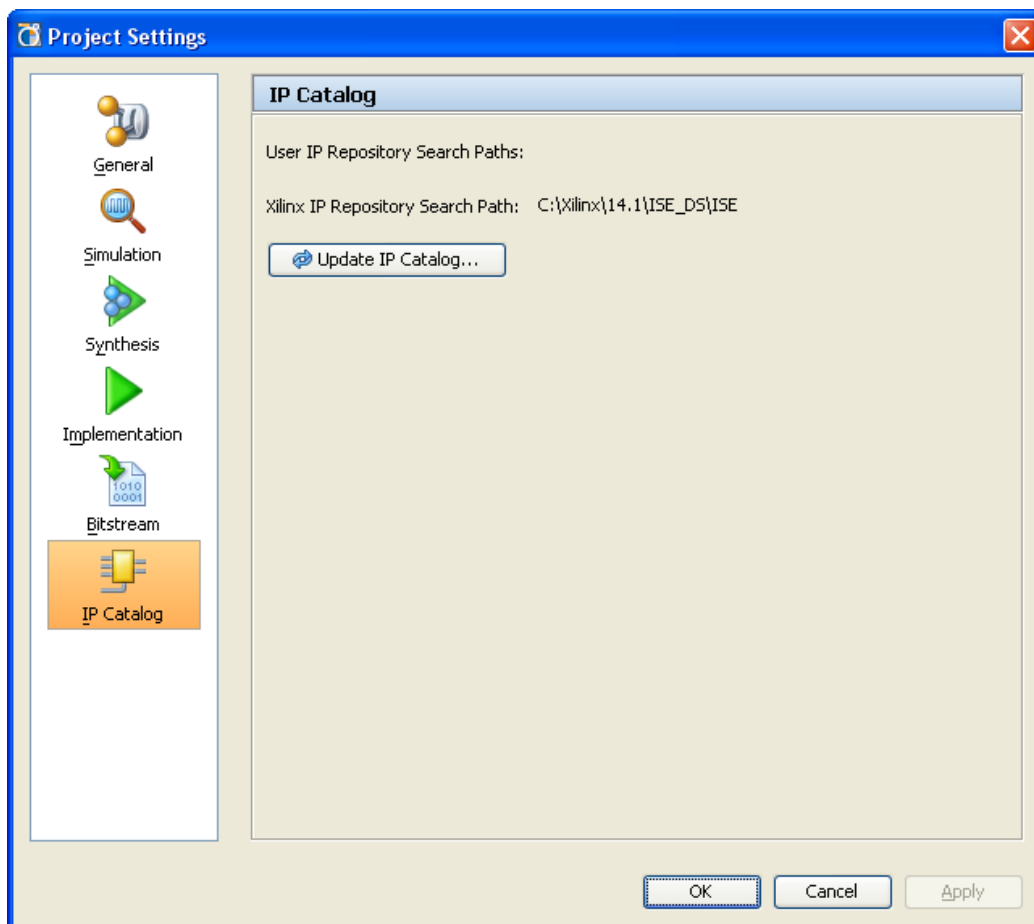


図 3-52 : [Project Settings] ダイアログ ボックスの [IP Catalog] ページ

[IP Catalog] ページには、次の情報が表示されます。

- [Xilinx IP Catalog Search Path] : ツールをインストールしたときに含まれるザイリンクス IP カタログへのパスが表示されます。
- [Update IP Catalog] : 詳細は、83 ページの「[IP カタログのアップデート](#)」を参照してください。

合成オプションの設定については、第 3 章「[IP コアの管理](#)」を参照してください。

表示環境の使用

PlanAhead™ では、デザイン情報およびデバイス情報が、作業内容に応じたビュー レイアウトで表示され、特有の機能とデザイン情報に対するさまざまな視点が提供されます。

ほとんどのビューでは、1 つのビューで選択した情報がほかのビューでも選択されるようになっており、デザインおよびデバイス情報を効率的に調べることができるようになっています。たとえば、[I/O Ports] ビューでポートを選択すると、[Device] ビューまたは [Package] ビューでそのポートがハイライトされます。その他の開いているビューでも、該当項目がハイライトされます。プロジェクトのインプリメント済みデザイン内でオブジェクトを選択しても、RTL デザインが開いていれば、そちらでもそのオブジェクトが選択されます。

表示環境は、開いているプロジェクトのタイプによって異なります。詳細は、[第 3 章「プロジェクトの操作」](#)を参照してください。

PlanAhead を使用すると、RTL の開発および解析、論理合成、制約の定義、物理デザイン解析、フロアプラン、ISE® Design Suite ソフトウェアでのインプリメンテーションなど、FPGA デザインプロセスの主要ステップを制御できます。

表示環境

[112 ページの図 4-1](#) に、表示環境の主なエレメントを示します。

- 「1.メイン メニュー」
- 「2.メイン ツールバー」
- 「3.Flow Navigator」
- 「4.ビュー レイアウト セレクター」
- 「5.メイン表示エリア」
- 「6.プロジェクト ステータス バー」
- 「7.Tcl コンソールおよびメッセージ エリア」
- 「8.ステータス バー」

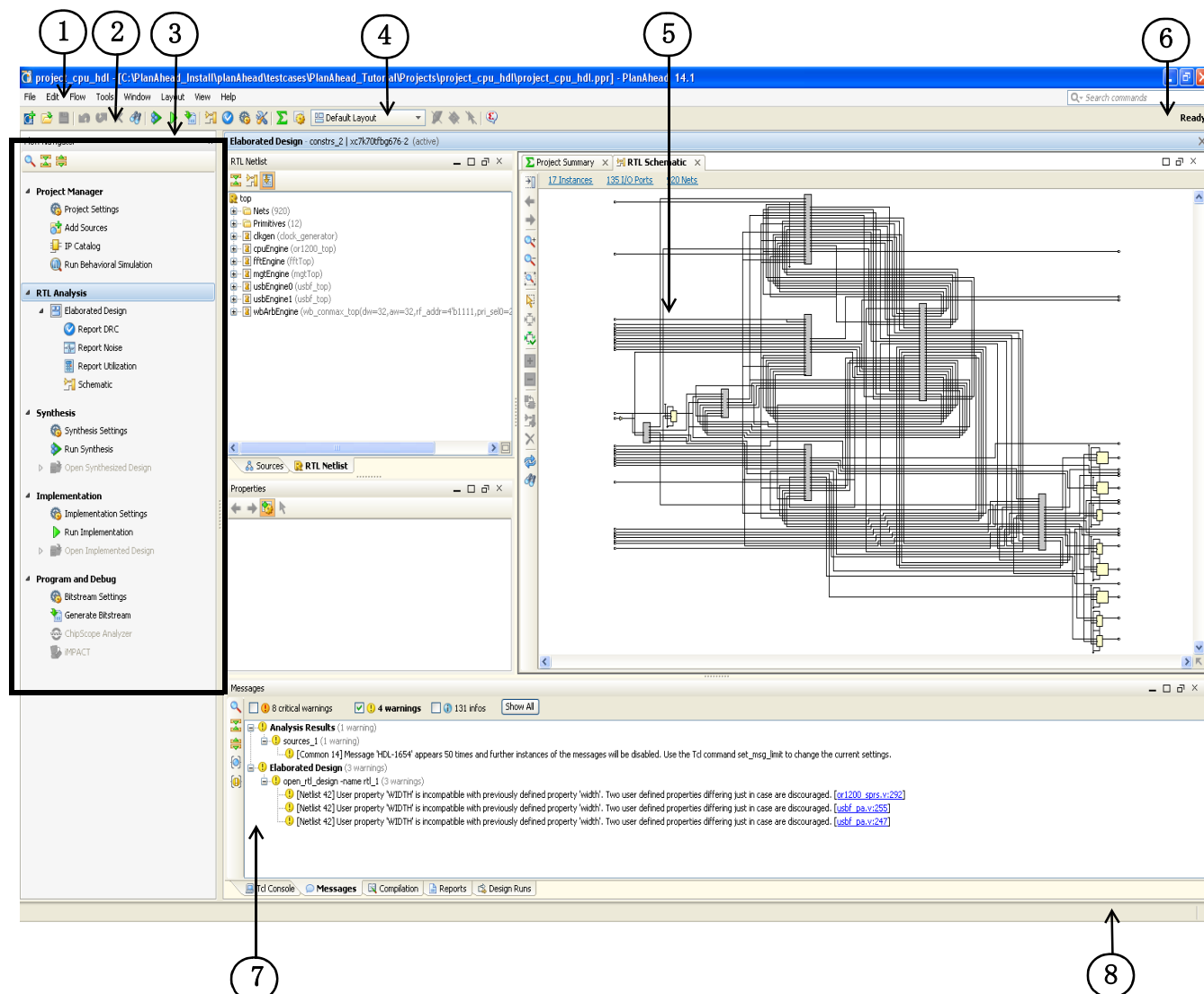


図 4-1：PlanAhead ツールの表示環境

1. メイン メニュー

さまざまなコマンドが含まれます。デザインのステートや選択したオブジェクトによって使用できないコマンドは、淡色表示されます。

2. メイン ツールバー

頻繁に使用されるコマンドおよびデザイン特定のコマンドを含みます。

3. Flow Navigator

デザイン プロセスをフローに基づいて管理します。プロジェクト データの管理、RTL ソースおよび制約ファイルの表示と演習、合成およびインプリメンテーションの起動、ビットストリームの生成を実行できます。これらのコマンドは、[Flow] メニューからも実行できます。

4. ビュー レイアウト セレクター

定義済みおよびユーザー定義のビュー レイアウトを切り替えます。詳細は、[116 ページ](#)の「ビュー レイアウトの使用」を参照してください。

5. メイン表示エリア

[Project Summary] およびデザインのための他のビューを表示します。メイン表示エリアは、[図 4-2](#)に示すように、データ ビューとグラフィカル ビューに分かれます。

- データ ビューでは、[Sources]、[Netlist]、および [Properties] ビューなどのデザイン データがツリー表示およびテキスト表示されます。
- グラフィカル ビューでは、ワークスペース エリアで [Device] や [Package] ビューなどが表示され、デザインおよびデバイス リソースが操作できます。

各ビューの使用法については、[122 ページ](#)の「ビューの操作」を参照してください。

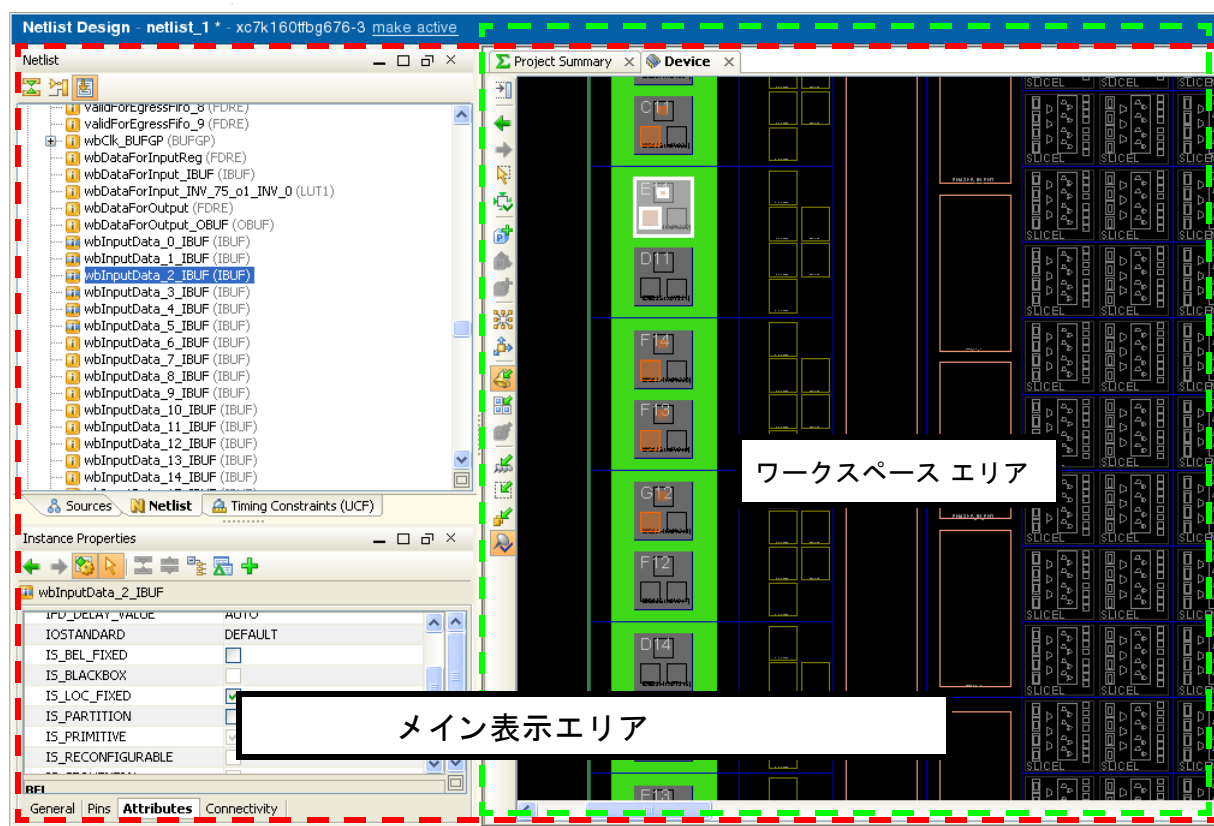


図 4-2：メイン表示エリア

6. プロジェクト ステータス バー

プロジェクトのステータスと実行中のコマンドを示します。このエリアから合成、インプリメンテーション、ビットストリーム生成をキャンセルできます。

7. Tcl コンソールおよびメッセージ エリア

Tcl コマンドのステータス、アプリケーションのメッセージ、コンパイル結果、レポート、デザイン run へのアクセスを表示します。詳細は、117 ページの「Tcl コンソールおよびメッセージ エリアの使用」を参照してください。

8. ステータス バー

プロジェクトおよびカーソルを置いているオブジェクトに関する情報を表示します。

メイン表示エリア

各ビューは、個別に最小化/最大化したり、メイン ウィンドウから切り離して別のウィンドウとして表示したり、閉じたりできます。これらの操作には、ビュー ウィンドウの右上に表示されている [最小化]、[最大化]、[フレームをフロートする]、および [閉じる] ボタンを使用します (図 4-3 を参照)。

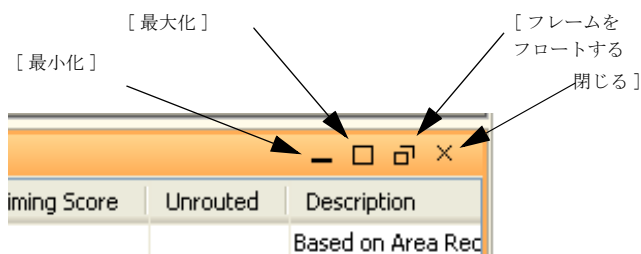


図 4-3：ビューの操作ボタン

次に、これらのコマンドについて詳細に説明します。

ビュー パナーの使用

ビューの右上にある [最大化] ボタンをクリックすると、そのビューを表示環境全体を使用して表示できます。ビューのタイトル バーをダブルクリックすると、そのビューを最適化したり、元の大きさに戻すことができます。Flow Navigator 以外のビューは最小化され、選択したビューが使用可能なエリア全体に拡大されます。

最小化されているビューは、一時的に横または下にタブとして配置されます。[最小化] をクリックすると、特定のビューを最小化できます。

図 4-4 では [Device] ビューが最大化され、[Properties]、[Netlist]、および [Sources] ビューが最小化されて左側にタブとして表示されています。最小化されているビューをクリックすると、元の位置に回復できます。

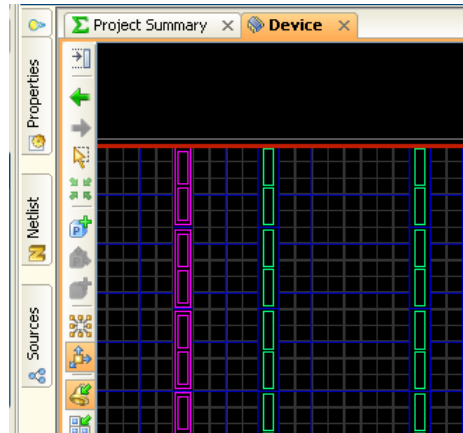


図 4-4：最大化されたビューと最小化されたビュー

ビューのフロート

ビューは、ワークスペースに表示されるものも含め、メイン ウィンドウから切り離してフロート表示し、個別に移動およびサイズ変更できます。ビューをフロート表示するには、次のいずれかを実行します。

- ビュー バナーの [Float Frame] ボタン をクリックします。
- ビュー バナーを右クリックし、[Float] をクリックします。



ビューをフロート表示すると、ビューが別のウィンドウに表示されます。

ビューが重なっている場合は、ビュー バナーをドラッグして移動できます。

フロート表示されているビューは、メイン ウィンドウの外に表示できます。フロート ビューのデフォルトの表示位置とサイズは保存されます。

ビューを閉じる

ビューを閉じるには、ビュー タブの [閉じる] ボタン (X マーク) をクリックします。

Flow Navigator の表示/非表示

Flow Navigator は、その他のビューの表示領域を広げるため、非表示にできます。Flow Navigator を非表示にするには、次のいずれかを実行します。

- メイン メニューから [View] → [Hide Flow Navigator] をクリックします。
- Flow Navigator の右側の [<<] ボタンをクリックします。

Flow Navigator を表示するには、次のいずれかを実行します。

- [View] → [Show Flow Navigator] をクリックします。
- 図 4-5 に示すように [<<] ボタンかウィンドウ スライダーを使用して、Flow Navigator の表示を調整します。

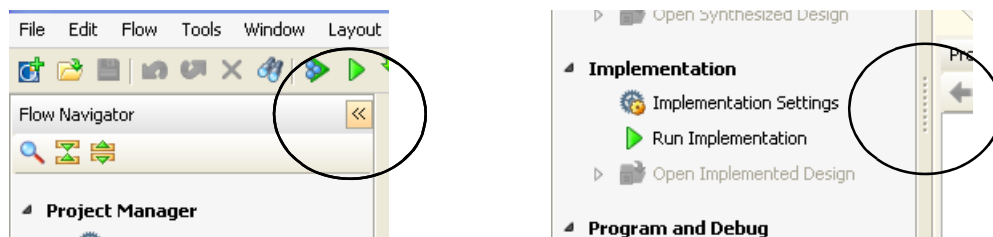


図 4-5 : Flow Navigator の表示 / 非表示

ビュー レイアウトの使用

116 ページの図 4-6 に示すようにデザイン プロセスのさまざまなタスクを効率的に実行できるようにするビュー レイアウトが用意されており、選択したデザイン プロセスに必要なタスクに便利なビューが表示されます。

デザインを開くと、次のいずれかを使用してビュー レイアウトを切り替えることができます。

- メイン メニューの [Layout] メニュー
- ツールバー メニューのビュー レイアウト セレクター

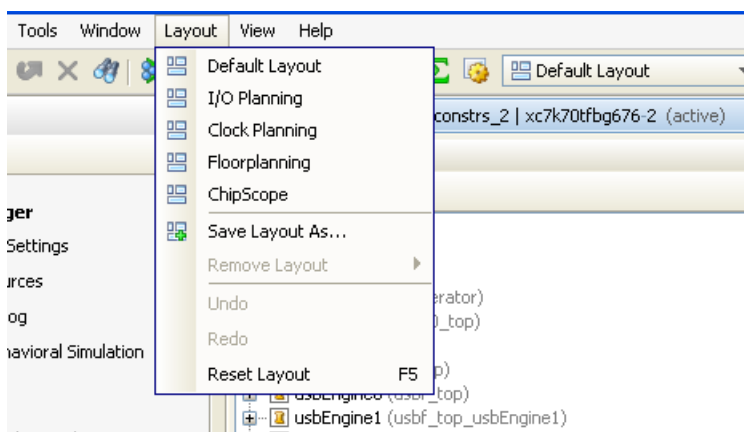


図 4-6 : [Layout] メニュー

次のレイアウトがあります。

- **Default Layout** : デザインを開いてメモリにエラボレートすると、デザイン データが **Design Analysis** ビュー レイアウトに表示されます。このビュー レイアウトでは、エラボレート済みおよび合成済みデザインを解析し、タイミング制約および物理制約を適用できます。
- **I/O Planning** : I/O 配置制約を定義し、ポートを配置します。
- **Clock Planning** : [Clock Resources] ビュー、[Device] ビュー、および [I/O Port] ビューを使用して、デザインのクロック リソースを配置します。
- **Floorplanning** : Pblock の定義、パーティションの管理、階層フロアプランを実行します。
- **ChipScope** : ChipScope™ Pro Analyzer ツールのデバッグ コアを挿入します。

独自の要件に合ったカスタム ビュー レイアウトも作成できます。

ビュー レイアウトの作成と削除

PlanAhead では、I/O Planning ビュー レイアウトや Design Analysis ビュー レイアウトなど、特定のデザイン タスクを完了するために設定されたビュー レイアウトが提供されています。これらのビュー レイアウトでは、そのデザイン タスクで頻繁に使用されるビューの位置およびサイズがあらかじめ定義されています。必要に応じてビューを移動したりサイズを変更した場合、ユーザー定義レイアウトを作成および保存したり、削除またはリセットできます。

- ビュー レイアウトを保存するには、[Layout] → [Save Layout As] をクリックします。ユーザー定義のビュー レイアウトは、すべてのデザイン プロジェクトで使用できます。これはメインメニューの [Layout] プルダウン メニューにリストされ、レイアウト ファイルとして保存されます (付録 A の「デフォルト環境の出力」を参照)。
- ユーザー定義のビュー レイアウトを削除するには、[Layout] → [Remove Layout] コマンドを使用します。このコマンドをクリックすると、ユーザー定義のビュー レイアウトがサブメニューで表示されます。
- また、ビューを移動したりサイズ変更した後、[Layout] → [Reset Layout] をクリックすると元のビュー レイアウトに戻すことができます。
- [Layout] → [Undo] をクリックすると、ビュー操作を取り消すことができます。取り消した操作をもう一度実行するには、[Layout] → [Redo] をクリックします。

Tcl コンソールおよびメッセージ エリアの使用

PlanAhead 内で実行されたコマンドのステータスおよび結果は、表示環境の下部に表示されます。メッセージが生成されると、このエリアの該当するビューに表示されます。

表示されるビューには、[Tcl Console]、[Messages]、[Compilation]、[Reports]、[Design Runs] ビューなどがあります。また、[Find Results]、[Package Pins]、[Timing Results] などのビューもここに表示されます。

[Messages] ビュー

メッセージは、さまざまなツールやプロセスからのメッセージを探しやすいように、カテゴリ別に表示されます。たとえば、合成済みデザインを開くと、[Messages] ビューは 図 4-7 に示すような表示になります。

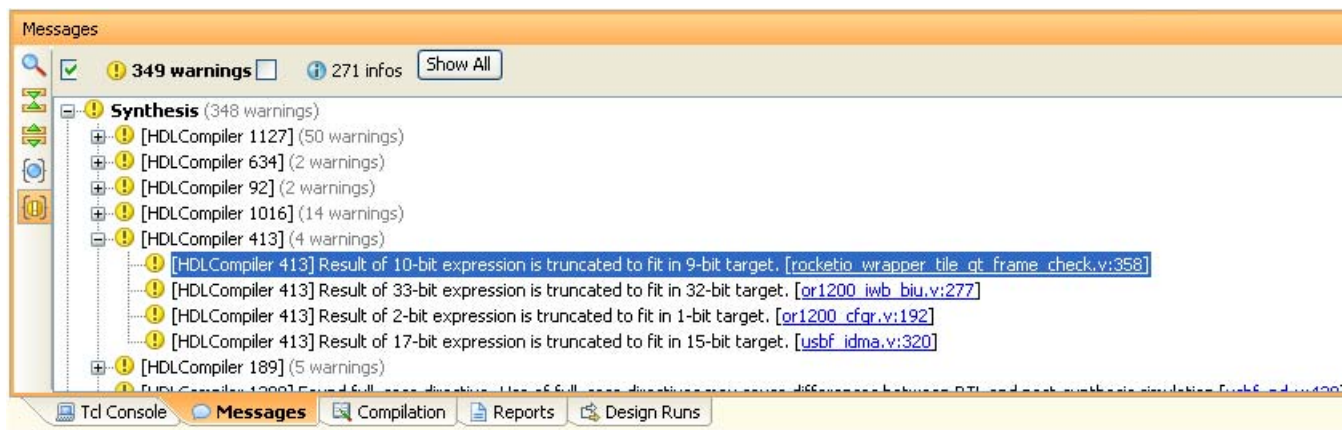


図 4-7 : [Messages] ビュー

[Messages] ビューを開くには、[Window] → [Messages] をクリックします。

- [Messages] ビューには、関連するソース ファイルへのリンクが表示されます。リンクをクリックするとテキスト エディターで RTL ソース ファイルが該当行がハイライトされた状態で開きます。
- メッセージの横にあるプラス記号 (+) またはマイナス記号 (-) をクリックすると、ツリーを展開したり、閉じたりできます。また、サイドバーの [Expand All] または [Collapse All] ボタンをクリックすると、すべてのツリーを展開したり、閉じたりできます。
- [Messages] ビューのバナーにあるチェック ボックスを使用して、エラー、重要な警告、警告、情報メッセージの表示/非表示を切り替えることができます。
- サイドバーの [Show Search] ボタンをクリックすると、特定のメッセージを検索するための [Search] フィールドが表示されます。このフィールドは、キーボードで Alt + / キーを押して開くこともできます。
- [Group messages by file] を使用すると、各ソース ファイルのメッセージが分類できます (117 ページの図 4-7)。
- [Group messages by ID] を使用すると、メッセージ ID でメッセージを分類できます。

デフォルトでは、メッセージの行が折り返されて、メッセージ全体が表示されます。行の折り返しがオンになっていると、[Messages] ビューの幅に収まるように行が調整されます。[Messages] ビューのサイズを変更すると、行の折り返しもそれに応じて調整されます。行の折り返しがオフになっていると、[Messages] ビューの幅にかかわらず、1 つのメッセージは 1 行に表示されます。

行の折り返しのオン/オフを切り替えるには、[Messages] ビューを右クリックして [Wrap Messages] をクリックします (118 ページの図 4-8)。

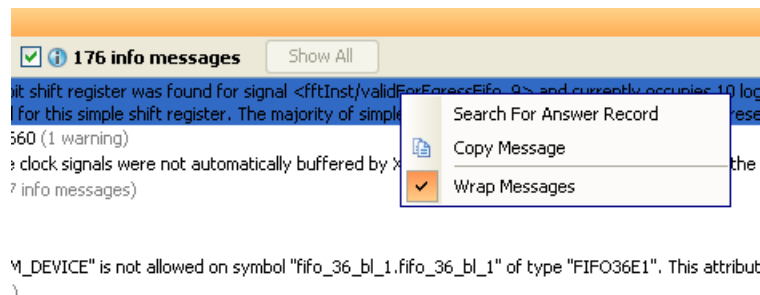


図 4-8：メッセージの折り返し機能のオン / オフ

クリティカル警告およびエラーは、プロジェクトを読み込んだとき、デザインをひらいたとき、run を作成または開始したときになどに表示されます (図 4-9)。このポップアップ ダイアログ ボックス、問題が発生している場合にそれに確実に気づくように表示されます。これらのメッセージは、[Messages] ビューにも表示されます。

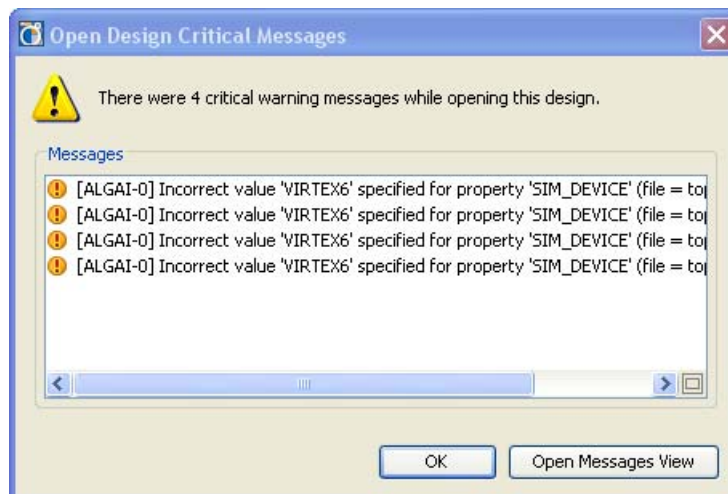


図 4-9 : [Critical Messages] ダイアログ ボックス

[Compilation] ビュー

[Compilation] ビューには、ngc2edif、Xilinx Synthesis Technology (XST)、マップ、および配置配線 (PAR) などのデザインをコンパイルするコマンドのステータスが表示されます。

[Compilation] ビューには、119 ページの図 4-10 に示すように、コンパイル コマンドからの標準出力が表示されます。このビューは、アクティブ run でコマンドが実行されると自動的に表示されます。[Compilation] ビューを開くには、次のいずれかを実行します。

- [Compilation] タブをクリックします。
- [Window] → [Compilation] をクリックします。

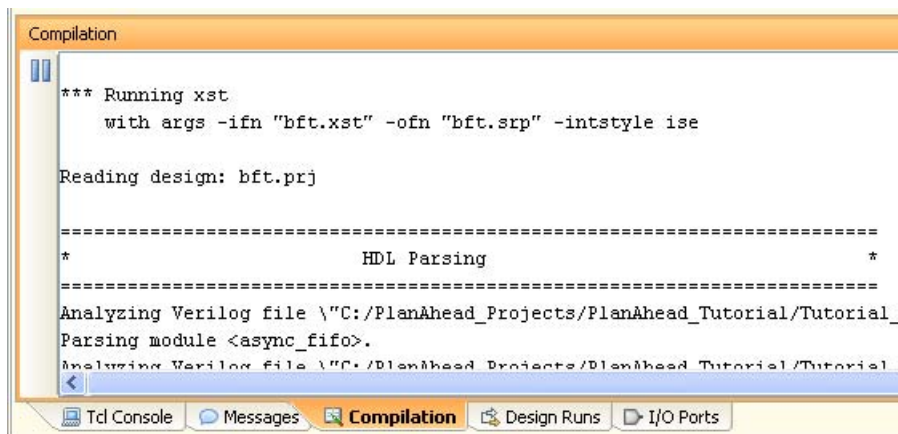


図 4-10 : [Compilation] ビュー

コンパイル出力は、追記される形でスクロールされながら表示され、新しいコマンドが実行されると上書きされます。

[Pause output] ボタン をクリックすると一時的に出力を停止でき、コマンドを実行中にレポートをスクロールして読むことができます。



[Tcl Console] ビュー

[Tcl Console] ウィンドウには、実行した Tcl コマンドのメッセージが表示されます。これらメッセージは、planAhead.log ファイルにも書き込まれます。コマンドのエラー、警告、および完了に関するメッセージも、このウィンドウに表示されます。

Design Planner および結果ビューで開いているデザイン ネットリストと制約のステータスも、このビューに表示されます。

[Tcl Console] ビューを表示するには、[Window] → [Tcl Console] をクリックします。図 4-11 に、[Tcl Console] ビューを示します。

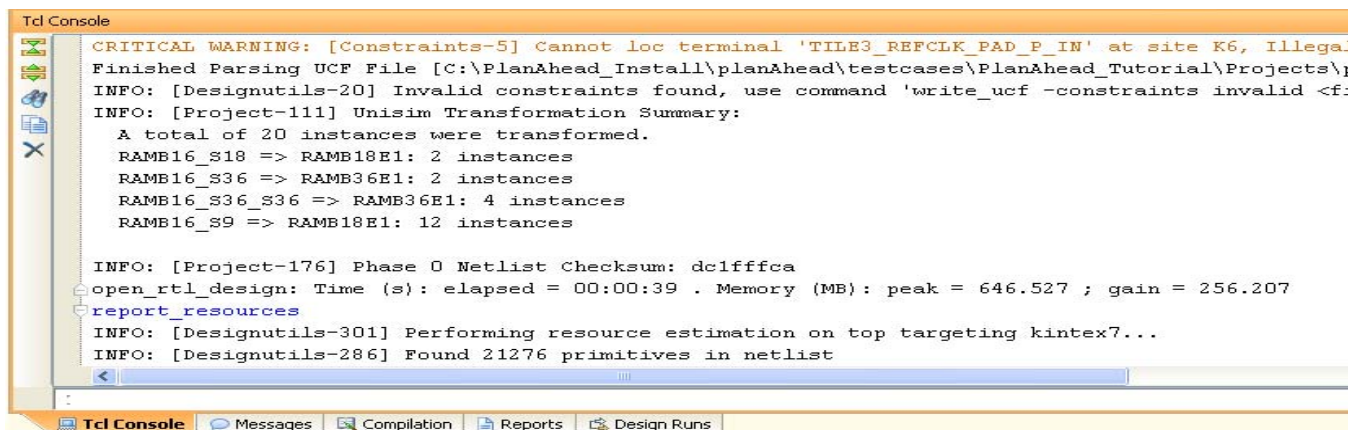


図 4-11 : [Tcl Console] ビュー

[Tcl Console] ビューでは、次の操作を実行できます。

- プラス記号 (+) またはマイナス記号 (-) をクリックするか、サイドバーの [Expand All] または [Collapse All] ボタンをクリックすると、各 Tcl コマンドからのメッセージを展開したり閉じたりできます。
- [Show Find] ボタンをクリックすると、検索文字列を入力する [Find] フィールドが表示されます。
- [Copy] をクリックすると、[Tcl Console] ビュー内のコマンドをコピーできます。
- [Clear all output in the Tcl console] をクリックするか、[Tcl Console] ビューを右クリックして [Clear All Output] をクリックすると、[Tcl Console] ビューがクリアされます。

[Tcl Console] ビューの警告およびエラー

[Tcl Console] ビューの右側には、図に示すように、警告に対して黄色のバー、エラーに対して赤色のバーが表示されます。

- このバーの上にカーソルを置くと、ツール ヒントに対応するメッセージが表示されます。
- バーの色部分をクリックすると、[Tcl Console] ビューのそのメッセージの部分に移動します。



Tcl コマンドの入力⁽¹⁾

[Tcl Console] ビューの下部にあるコマンド ライン入力ボックス (120 ページの図 4-11 を参照) に、Tcl コマンドを入力できます。コマンドを入力するには、そのコマンド ラインをクリックして、Tcl コマンドをタイプします。

Tcl コンソールの自動判別機能により、コマンド ラインにタイプすると、そのコマンドの名前やコマンド パラメーターが表示されます。121 ページの図 4-12 は、Tcl コンソールに `create_` で始まるコマンドがリストされたところを示しています。リストからコマンドをクリックして選択するか、矢印キーを使用してコマンドまでスクロールし、**Enter** キーを押して選択します。自動判別機能によりコマンドの選択肢が 1 つに絞られるまでタイプを続けて、**Tab** キーを押して選択します。

コマンドを選択すると、そのコマンドの引数が自動的に判別され、表示されます。パラメーターのリストから選択することもできます。または、リストが 1 つの選択肢まで狭まったら **Tab** キーを押します。

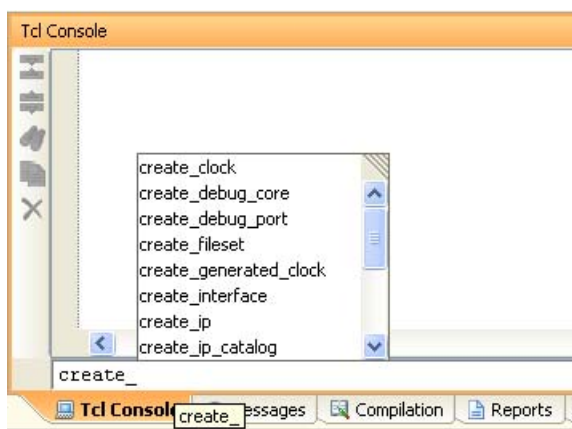


図 4-12 : Tcl コマンドの自動判別機能

1. Linux の場合、特定のキー入力や組み合わせをホーム ディレクトリの `.editrc` ファイルで設定できます。詳細は、第 14 章「Tcl コンソールでのキーストロークのマップ」を参照してください。

PlanAhead には、メニュー操作やドラッグ アンド ドロップなどの操作がすべての Tcl コマンドを Tcl コンソールと `planAhead.jou` ファイルに書き込む機能があります。`planAhead.jou` ファイルは、PlanAhead を起動するたびに記述されます。

注記：このファイルのバックアップ バージョンが `planAhead.jou_backup` で、1 つ前のセッションで実行した Tcl コマンドの詳細が保存されています。

`history` コマンドを使用すると、Tcl コンソールからコマンド 履歴にアクセスできます。上下の矢印キーを押すと、コマンド履歴のコマンド を1 つずつスクロールして確認できます。

Tcl ヘルプの使用

次の構文をコマンド ラインに入力すると、コマンドに関するヘルプが表示されます。

```
Command> help
```

各コマンドの詳細は、次のように入力すると表示されます。

```
Command> help get_cells
```

入力した内容に応じて、[Tcl Console] ビューに使用可能なコマンドまたはコマンド オプションが表示されます。

正確なコマンド構文を確認するには、コマンドを 1 度実行し、PlanAhead の起動ディレクトリに含まれている `planAhead.jou` ファイルを参照します。Tcl コマンドのフォーマットおよびヘルプに関する詳細は、[第 14 章「Tcl およびバッチ スクリプト」](#)を参照してください。

Tcl コンソールおよびメッセージ エリアの非表示

Tcl コンソールおよびメッセージ エリアを非表示にするには、このエリアの右上にある [最小化] をクリックします。

Tcl コンソールおよびメッセージ エリアを最小化すると、表示エリアの下部にタブとして表示されます。最小化されたタブのいずれかをクリックすると、メッセージ エリアが再び元の位置に表示されます。

[122 ページの図 4-13](#) に、Tcl コンソールおよびメッセージ エリアの右上にある [最小化] ボタンと、最小化されたときに表示環境の下部に表示されるタブを示します。

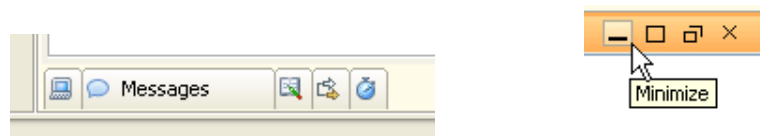


図 4-13 : Tcl コンソールおよびメッセージ エリアの [最小化] ボタンと最小化されたときのタブ表示

ビューの操作

PlanAhead には、異なる情報を表示するさまざまなビューがあります。ビューのサイズ、表示/非表示、および場所は、個別に制御できます。

ビューを開く

ほとんどのビューは、メイン メニューの [Window] メニューから開くことができます。既に開いているビューを選択すると、そのビューがアクティブになります。

コマンドを実行すると、そのコマンドを操作するため、または結果を表示するため、対応するビューが開きます。

- [Schematic] ビューを開くには、[Sources] ビューなどの別のビューでデザイン オブジェクトを少なくとも 1 つ選択しておく必要があります。ポップアップ メニューから [Schematic] をクリックするか、[Schematic] ツールバー ボタンをクリックすると、[Schematic] ビューを開くことができます。詳細は、[144 ページの「\[Sources\] ビュー」](#)を参照してください。
- [Properties] ビューを開くには、[Netlist] ビューなどの別のビューでデザイン オブジェクトを 1 つ選択しておく必要があります。複数選択すると、最後に選択したもののプロパティが表示されます。ポップアップ メニューから [Object_type Properties] をクリックすると、[Properties] ビューを開くことができます。詳細は、[169 ページの「\[Properties\] ビュー」](#)を参照してください。

ビューのナビゲート

表示されている各ビューにはタブが付いています。タブをクリックすると、そのビューがアクティブになります。[Tcl Console] や [Messages] ビューなどのタブは、ビュー ウィンドウの下にあり、[Project Management] や [Device] ビューなどのタブは、ビュー ウィンドウの上にあります。タブをクリックすると、ビューがアクティブになります。

ビューでは、次の操作を実行できます。

- ビューのタブをダブルクリックすると、そのビューを最大化できます。
- ワークスペースにビューを復元するには、ビュー タブを再びダブルクリックします。
- ビューのサイズを変更するには、表示枠をドラッグします。表示枠にカーソルを置くと、カーソルがスライダ シンボルに変わり、ビューのサイズを変更できるようになります。
- ビューを移動するには、ビュー タブを新しい場所にドラッグします。詳細は、[190 ページの「表示環境の設定」](#)を参照してください。

一部のビューには、複数のタブがあります。たとえば、[123 ページの図 4-14](#)では、[Results] ビューの [SSN Results] タブに results_1 および results_2 という 2 つの結果が表示されたところを示しています。



図 4-14 : [SSN Results] ビュー

ビューの移動

ビューを移動するには、タブを新しい場所にドラッグします。移動した後にビューが表示される場所に四角形のボックスが表示されます。既存のビューの上に別のビューをドロップすると、同じ場所に 2 つのタブが表示されます。

ビューを移動して表示エリアを分割するには、次の手順に従います。

1. タブをクリックします。
2. タブを移動先にドラッグします。グレーのアウトラインで移動場所を確認できます。
3. 移動先でマウスを放します。

注記：ビューはグラフィック ワークスペース内外へ移動できません。

ワークスペースのビュー

テキスト エディターやレポート ビューアーのようにグラフィカル インターフェイスを持つビューおよび広い表示領域が必要なビューは、ワークスペースに表示されます。これらのビューは、同時に複数開いて情報を比較できる点が、その他のビューとは異なります。ワークスペースのビューは、ビュー タブを右クリックして表示されるポップアップ メニューを使用して、最大化、フロート表示、または分割できます。

PlanAhead のワークスペースには、デザインがグラフィカルに表示され、レポートやログも表示されます。ワークスペースには、次のものが表示されます。

- [Project Summary] ビュー
- テキスト エディター
- [Device] ビュー
- [Package] ビュー
- [Clock Resource] ビュー
- [Schematic] ビュー
- [Hierarchy] ビュー

新しい [Device] ビューまたは [Package] ビューを開くには、[Window] → [Device] または [Window] → [Package] をクリックします。ワークスペースには、同じタイプのビューを複数開くことができます。たとえばデバイスの異なる部分を表示するために、[Device] ビューを 2 つ開くことができます。

ほとんどのグラフィカル ビューは [Windows] メニューから開くことができますが、[Schematic] ビューおよび [Hierarchy] ビューを開くにはまず別のビューでロジック エレメントを選択する必要があります。[Schematic] ビューを開くには、次の手順に従います。

1. [Netlist] ビューなどで回路図で開くオブジェクトを 1 つ以上選択します。
2. ポップアップ メニューから [Schematic] コマンドをクリックします。
 - F4 キーを押します。
 - ツールバーの [Schematic] ボタン をクリックします。



ワークスペースに [Schematic] ビューが表示されます。ワークスペースには、複数の [Schematic] ビューを開くことができます。

文脈依存カーソルについて

使用可能なコマンド モードによって、カーソルが変化します。カーソルの形状に応じて、次の操作が可能です。

- カーソルが水平、垂直、または斜めのバー シンボルに変化した場合、Pblock およびビューのサイズを変更できます。
- カーソルが手の形に変化した場合、Pblock やインスタンスを移動できます。
- カーソルが十字型に変化した場合、長方形を描画した拡大範囲指定、ピン割り当てエリアの定義、または Pblock の長方形の描画を実行できます。
- オブジェクトを移動できない場所にドラッグすると、カーソルは斜線が入った円になります。
- オブジェクトを移動できる場所にドラッグすると、カーソルにプラス記号が付きます。

マウスを使用した拡大/縮小

[Device] ビューでマウスをクリックしてカーソルをドラッグすると、エリアを拡大または縮小できます。次のような表示変更が可能です。

- 特定エリアの拡大：左上から右下へ向かってドラッグして四角形を描画すると、そのエリアが拡大されます。
- 拡大：右上から左下へ向かって斜めにドラッグすると、表示が拡大されます。この方法を使用すると、[Device] ビューをさまざまな大きさに拡大できます。斜めにドラッグした線の長さによって、拡大レベルが変わります。
- 縮小：左下から右上へ向かって斜めにドラッグすると、表示が縮小されます。この方法を使用すると、[Device] ビューをさまざまな大きさに縮小できます。斜めにドラッグした線の長さによって、縮小レベルが変わります。
- 全体表示：右下から左上へ向かって斜めにドラッグすると、表示が縮小されます。[Device] ビューが縮小されて、デバイス全体が表示されます。

[World] ビュー

[Device] ビューなどのグラフィカルなワークスペース ビューを拡大表示した場合、[World] ビューを開いてデザイン エリア全体ナビゲートできます。[World] ビューは、アクティブなワークスペースをより大まかに表示したビューで、表示エリアをすばやく移動するために使用できます。このビューは、デバイスまたはデザインの小さなエリアを [Device]、[Schematic]、[Package]、および [Hierarchy] ビューで拡大表示している場合に使用できます。

[World] ビューを開くには、図 4-15 に示すように、[Device] ビューのようなワークスペースの右下にある [Show World View] ボタンをクリックします。

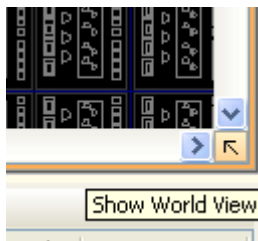


図 4-15 : [World] ビュー

[World] ビューには、アクティブ ビューで拡大表示されているエリアおよび選択されているオブジェクトが示されます。126 ページの図 4-16 に示す [World] ビューには、[Device] ビュー全体と拡大表示されているエリアが赤い四角で示されています。この長方形を選択してドラッグすると、グラフィカルなワークスペース ビューで表示エリアを変更できます。

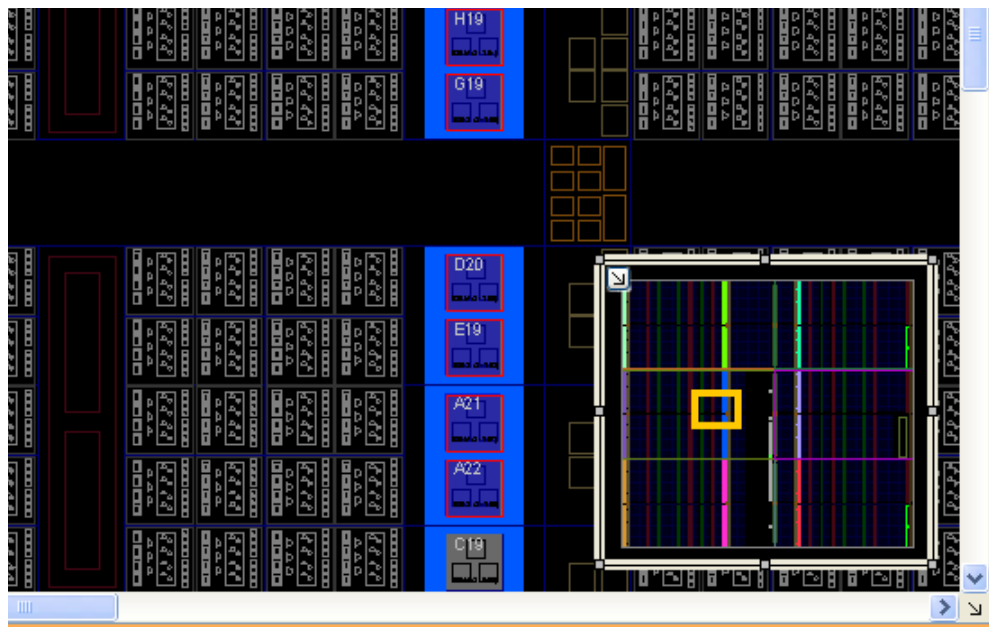


図 4-16 : [World] ビュー

[World] ビューは、デフォルト サイズで開きます。[World] ビューの大きさを変更するには、ビューの角にカーソルを置き、カーソルの形が変わったら、クリックしてドラッグします。

[World] ビューの位置を変更する場合は、ビューの周囲をクリックしてドラッグします。この機能を使用すると、制限内であれば、[World] ビューをワークスペースのどこにでも配置できます。



[World] ビューを閉じるには、ビューの下向き矢印のアイコンをクリックします。

[World] ビューの大きさを変更したり、位置を変更してから閉じると、次に開いたときも同じ状態で表示されます、

ワークスペースのビューの印刷

[Device]、[Package]、[Schematic]、[Hierarchy] ビューのワークスペースで表示されている箇所を印刷できます。[File] → [Print] をクリックし、現在表示されている箇所を印刷します。

ワークスペースの分割

ワークスペースの表示エリアを上下または左右に分割して、複数のウィンドウを同時に表示できます。各ウィンドウは個別に操作でき、複数のウィンドウをドッキング表示できます。

同じタイプのビューを 2 つ開くこともできます。たとえば、[Device] ビューを 2 つ開いてデバイスの異なる部分を表示したり、異なる倍率で表示したりできます。また、[Device] と [Package] ビュー (127 ページの図 4-17 を参照)、[Device] と [Clock Resources] ビューなど、異なる 2 つのビューを表示できます。

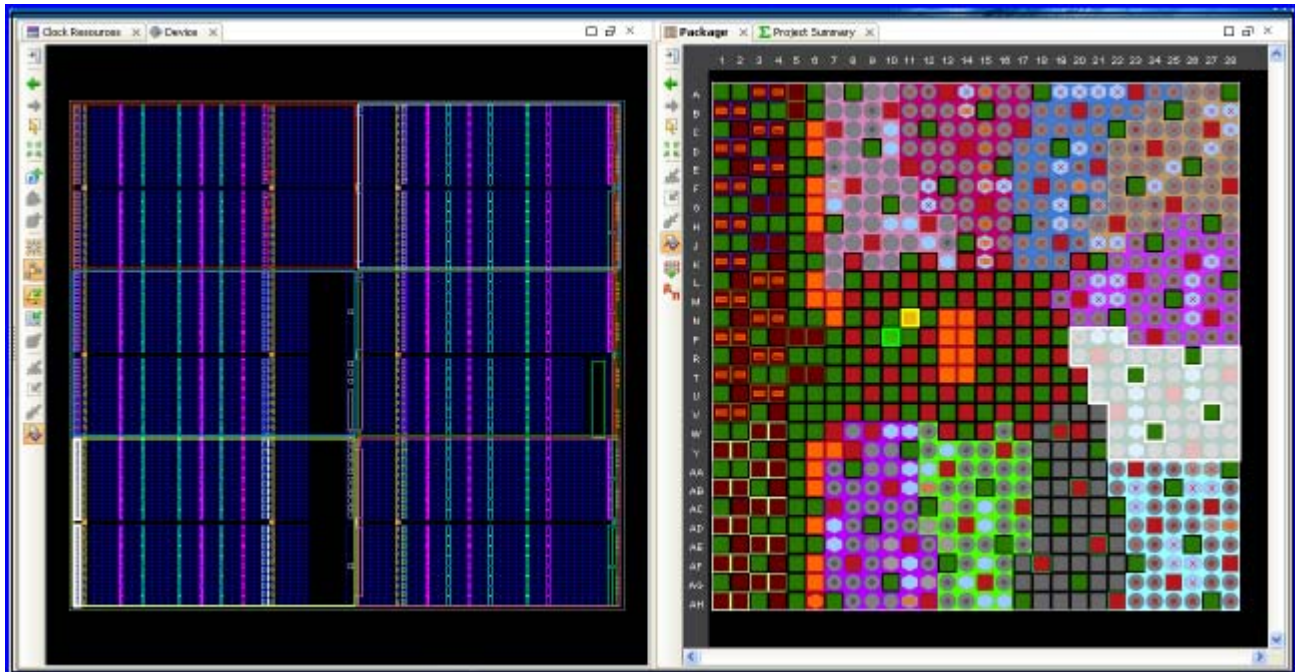


図 4-17 : [Package] ビューと [Device] ビューの同時表示

ワークスペースを分割するには、次のいずれかの方法を使用します。

- ビュー タブを右クリックして [New Horizontal Group] または [New Vertical Group] コマンドをクリックします。これらのコマンドは、ワークスペース ビューでのみ使用できます。
- 開いているビューのタブをクリックし、ワークスペースの右側のスクロール バーまでドラッグします。ビューが配置される箇所がグレーの長方形で囲まれます。カーソルをドラッグして配置箇所を決めたら、ドロップします。この方法は、ほかのビューでも使用できます。

分割されたビューを 1 つのビューに戻す

ワークスペース ビューを分割している場合、表示エリアを効率的に使用するため、ビューを 1 つに戻した方がよいことがあります。分割したビューを 1 つのビュー表示に戻すには、次のいずれかの操作を実行します。

- ワークスペース ビューのタブを右クリックし、[Move to Previous Tab Group]/[Move to Next Tab Group] をクリックします。
- ビューを選択し、別のビューのタブ上にドラッグします。グレーの長方形でビュー全体が囲まれたらドロップします。

ツリー表形式のビュー

PlanAhead には、スプレッドシートの表に似たビューが多数あります。これらのビューには、次のセクションで説明する共通の特性と機能があります。128 ページの図 4-18 に、ツリー表形式のビューの例を示します。

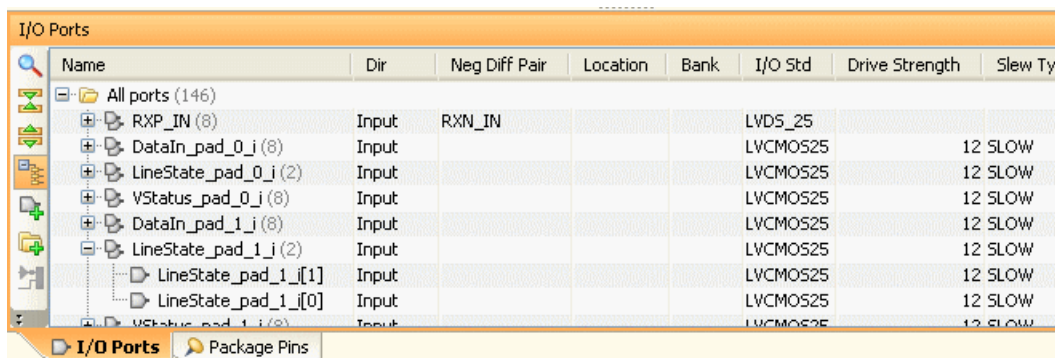


図 4-18：ツリー表形式のビュー

表の展開/非展開

[Name] 列のプラス記号 (+) またはマイナス記号 (-) をクリックすると、ツリーを個別に展開または非展開できます。

ツールバーの [Expand All] および [Collapse All] ボタンをクリックすると、すべてのツリーを展開または非展開できます。

エントリのグループ化またはフラット リスト表示

表形式のビューには、ツールバーに [Group by Type] ボタンがあり、クリックするとエントリを展開可能なグループごとに表示するか、フラットなリストとして表示するかを切り替えることができます。

リストをフラット化すると、図 4-19 に示すように、リスト全体を検索およびフィルター処理するのに便利です。

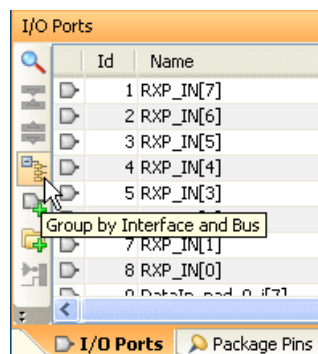


図 4-19：[Group by Type] ボタン

リスト ビューでの検索機能の使用

ツールバーの [Show Search] ボタンをクリックすると、ビュー パナーに [Search] フィールドが表示され、文字列を入力してリストを検索できます。このフィールドは、Alt + / キーを押しても開きます。

前述のセクションで説明したように、検索する前に [Group by Type] ボタンをクリックしてリストをフラット化しておくと、検索しやすくなります。

表のどの列も、検索条件として使用できます。[Search] フィールドのプルダウン メニューから、検索を実行する列を選択します。129 ページの図 4-20 に、[Search] フィールドのプルダウン メニューを示します。

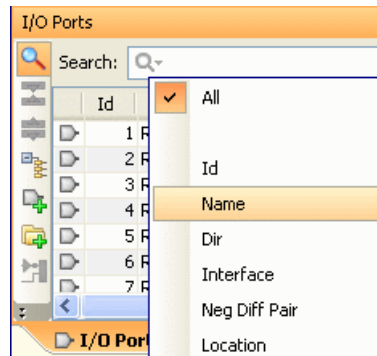


図 4-20 : [Search] フィールドのプルダウン メニュー

[Search] フィールドに文字列を入力すると、検索条件に一致するエントリのみが表示されます。
[Show Search] ボタンを再びクリックすると、[Search] フィールドが表示されなくなり、リストにすべてのエントリが表示されます。

列の並べ替え

表の列ヘッダーをクリックすると、その列を基準に並べ替えることができます。クリックした列の昇順にデータが並べ替えられます。列ヘッダーをもう一度クリックすると、その列の降順にデータが並べ替えられます。列ヘッダーには、図 4-21 に示すように、どの列を基準に並べ替えられているか、昇順か降順かが示されます。

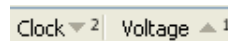


図 4-21 : 昇順 / 降順で並べ替えられていることを示す矢印

1 つの列で並べ替えた後、2 番目の列でさらに並び替えるには、Ctrl キーを押しながら 2 番目の列ヘッダーをクリックします。図 4-21 の例では、まず [Voltage] 列で並べ替えた後 [Clock] 列で並べ替えられています。

必要に応じてさらに別の列ヘッダーをクリックして並べ替え条件を追加できます。もう一度 Ctrl キーを押して列ヘッダーをクリックすると、その列の並べ替えが解除されます。

列の整列

列を移動、非表示、または復元できます。

- 列を移動するには、列をクリックし、新しい場所にドラッグします。
- 列を非表示にするには、列を右クリックして [Hide This Column] をクリックします。

右クリックで表示されるポップアップメニューには、各列を操作するためのその他のコマンドもあります。

- [Auto Resize Column] : 表示されているデータごとに列の幅を調整します。
- [Restore Defaults] : デフォルト表示に戻します。

ビュー特定のツールバー コマンド

ほとんどのビューには、図 4-22 に示すように、そのビュー特定のコマンドを実行するためのツールバー ボタンがあります。

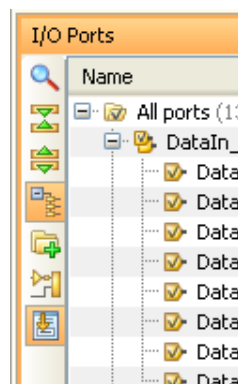


図 4-22 : [View] ツールバー メニュー

これらのボタンは、特定のデータが選択されている場合や、コマンドがアクティブな場合にのみ使用できます。PlanAhead の機能は、これらのビュー特定のツールバー ボタンから実行できます。ビュー特定のコマンドの詳細は、このガイドの該当するビューのセクションを参照してください。

メイン メニュー コマンドの検索機能の使用

PlanAhead には、表示環境の右上 (メイン メニュー バーの右端) に、メイン メニューのコマンドを検索して実行する検索フィールドがあります。コマンド名の数文字を入力すると、その文字を含むコマンドのリストが表示されます。

検索にはワイルドカード検索が使用されます。たとえば、図 4-23 のように、「cl」と入力すると Clear List、Clock Regions、Clock Resources、Close Project、Run Tcl Script、Tcl Console、Clear Prohibit、および Clear Placement が表示されます。最後の 2 つは、プロジェクトの現在の状態のため実行できなくなっています。

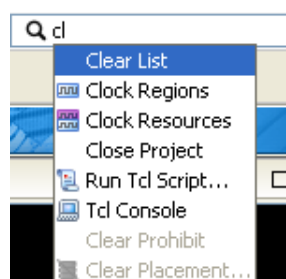


図 4-23 : メイン メニュー コマンドの検索機能

表示されるコマンドは、プロジェクトで現在表示されているデザインに基づいています。エラボーレート済みデザインを開いている場合とインプリメント済みデザインを開いている場合では、表示されるコマンドは異なります。

コマンドに加え、[File] メニューの [Open Recent Project] および [Open Example Project] でリストされるプロジェクト名およびファイルも表示されます。

リストからコマンドを選択すると、そのコマンドが実行されます。

ステータス バーの使用

メイン ウィンドウの下部にあるステータス バーには、マウス カーソルが置かれているアイテムに関する情報、現在選択されているオブジェクトの情報、カーソルの座標、現在のデザイン モードなど、有益な情報が表示されます。図 4-24 に、ステータス バーの左端と右端を示します。

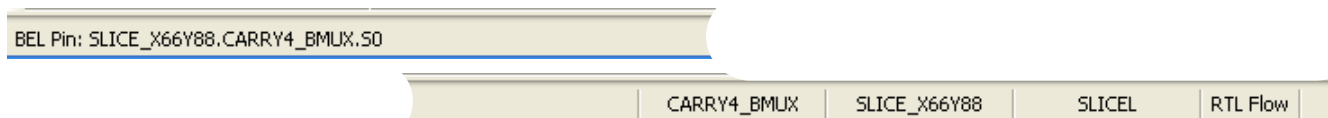


図 4-24：情報バー

ステータス バーには、次が表示されます。

- ステータス メッセージ：ステータス バーの最初のフィールドで、データに応じた情報が表示されます。たとえば、カーソルが [Device] ビューまたは [Schematic] ビュー上にある場合、このフィールドにはカーソルがポイントしているインスタンスまたはサイトの名前が表示されます。ツールバーまたはメニュー コマンドの上にカーソルを置くと、コマンドの詳細が表示されます。
- 座標：ステータス メッセージ フィールドの右側にあるフィールドで、座標情報が表示されます。[Device] ビューでカーソルをブロック RAM、DSP48 などの上に置くと、図 4-24 に示すように、その名前と座標が表示されます。[Package] ビューのピンの上にカーソルを置くと、その座標、タイプ、名前などのピン情報が表示されます。
- モード：RTL フロー、合成後のフローなど、プロジェクトのタイプが表示されます。PlanAhead を ISE Project Navigator から起動している場合、ISE 統合モードが表示されます。

オブジェクトの選択、マーク、移動

PlanAhead では、次のようにオブジェクトを選択する方法が複数あります。

- 1 つのオブジェクトを選択できます。現在のビューでオブジェクトを選択するには、そのオブジェクトをクリックします。オブジェクトを 1 つのビューで選択すると、開いているほかのビューでも選択されます。
- オブジェクトを選択すると、それに接続されたり、関連しているオブジェクトも選択されます。この機能は、選択規則から設定できます。詳細は、193 ページの「[選択規則オプションの設定](#)」を参照してください。
- 複数のオブジェクトを選択できます。最初のオブジェクトを選択し、Ctrl キーを押しながら複数のオブジェクトを選択します。
- インスタンスのオブジェクト リストからオブジェクトの範囲を選択することもできます。最初のオブジェクトを選択し、Shift キーを押してツリーや表から範囲内の最後のオブジェクトをクリックします。
- グラフィカル ビューでエリアにあるオブジェクトすべてを選択できます。詳細は、132 ページの「[\[Select Area\] コマンド](#)」を参照してください。

オブジェクトが重なる場合は、優先スキームにより小さい方のオブジェクトが選択されます。[Device] ビューでオブジェクトが選択しにくい場合は、[Physical Constraints] ビューまたは [Netlist] ビューで選択します。オブジェクトは、[PlanAhead Options] ダイアログ ボックスの [Selection Rules] ページの設定にかかわらず、この 2 つのウィンドウからであればいつでも選択できます (193 ページの「[選択規則オプションの設定](#)」を参照)。

オブジェクトを選択しにくい場合は、右クリックしたポップアップメニューから [Select] コマンドを使用すると、現在選択しているアイテムのあるオブジェクト リストから特定のオブジェクトを選択できます (132 ページの図 4-25)。

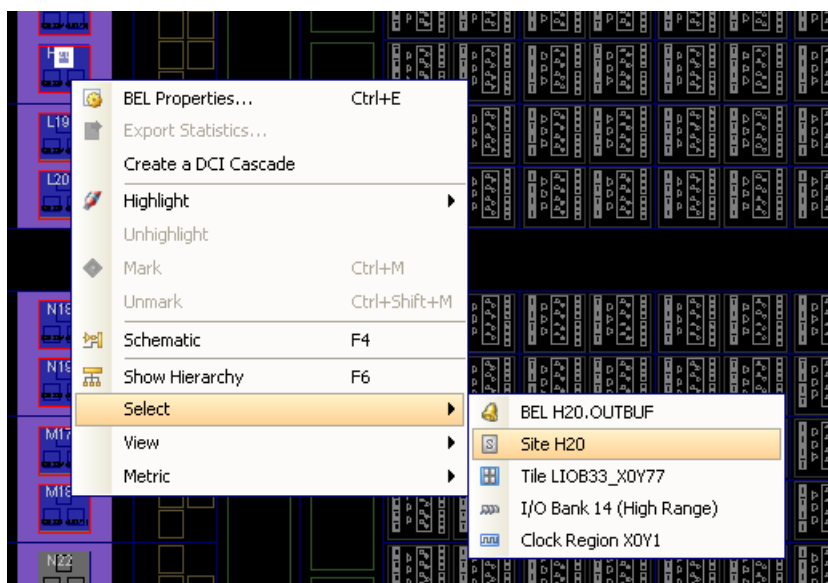


図 4-25 : [Select] コマンドによるオブジェクトの選択

[Select Area] コマンド

どのワークスペースのビューでも、オブジェクトを長方形で囲んで選択できます。

1. 次のいずれかを使用してください。

- [View] → [Select Area] をクリックします。
- ツールバーの [Select Area] ボタンをクリックします。

図 4-26 に示すように、長方形に囲まれたオブジェクトおよび接触しているオブジェクトがすべて、[Select Area] ダイアログ ボックスにタイプ別に表示されます。

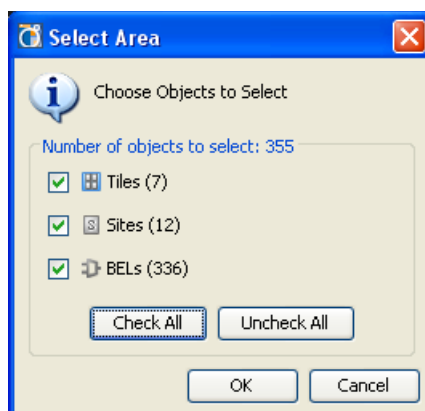


図 4-26 : [Select Area] ダイアログ ボックス

2. ここでオブジェクトのチェック ボックスをオフにすると、そのオブジェクトは選択されません。
3. [OK] をクリックすると、指定したオブジェクトがすべて選択されます。

プリミティブの親モジュールの選択

選択されたプリミティブ ロジックの親モジュールを選択するには、[Select Primitive Parents] コマンドを使用します。このコマンドはポップアップ メニューからクリックでき、ほとんどのビューで使用可能です。

フロアプランは、ロジック モジュールが **Pblock** に割り当てられている方が、プリミティブ ロジック インスタンスに割り当てられている場合よりも簡単に管理できます。

タイミング パスのグループを選択すると、そのパスに含まれるプリミティブ ロジック インスタンスがすべて選択されます。[Select Primitive Parents] コマンドを実行すると、選択されたプリミティブ ロジックすべての親モジュールが自動的に選択されます。

選択されていたプリミティブ ロジックの選択は解除され、選択されたロジックが **ROOT** レベルのロジックであった場合以外は、親モジュールのみが選択されます。

モジュールを選択すると、コマンドを実行しても親モジュールは選択されず、そのモジュールが選択されたままになります。

オブジェクトの選択規則

オブジェクトを選択すると、それに接続されたり、関連しているオブジェクトも選択されることがあります。たとえば、**Pblock** を選択すると、その **Pblock** に割り当てられたネットリストのインスタンスも選択されます。ポート オブジェクトを選択すると、そのポートのピン オブジェクトも選択されます。

[Tools] → [Options] → [Selection Rules] コマンドを使用すると、1 つ目のオブジェクトを選択したときに選択される 2 つ目のエレメントを指定できます。134 ページの図 4-27 に、[PlanAhead Options] ダイアログ ボックスの [Selection Rules] ページを示します。

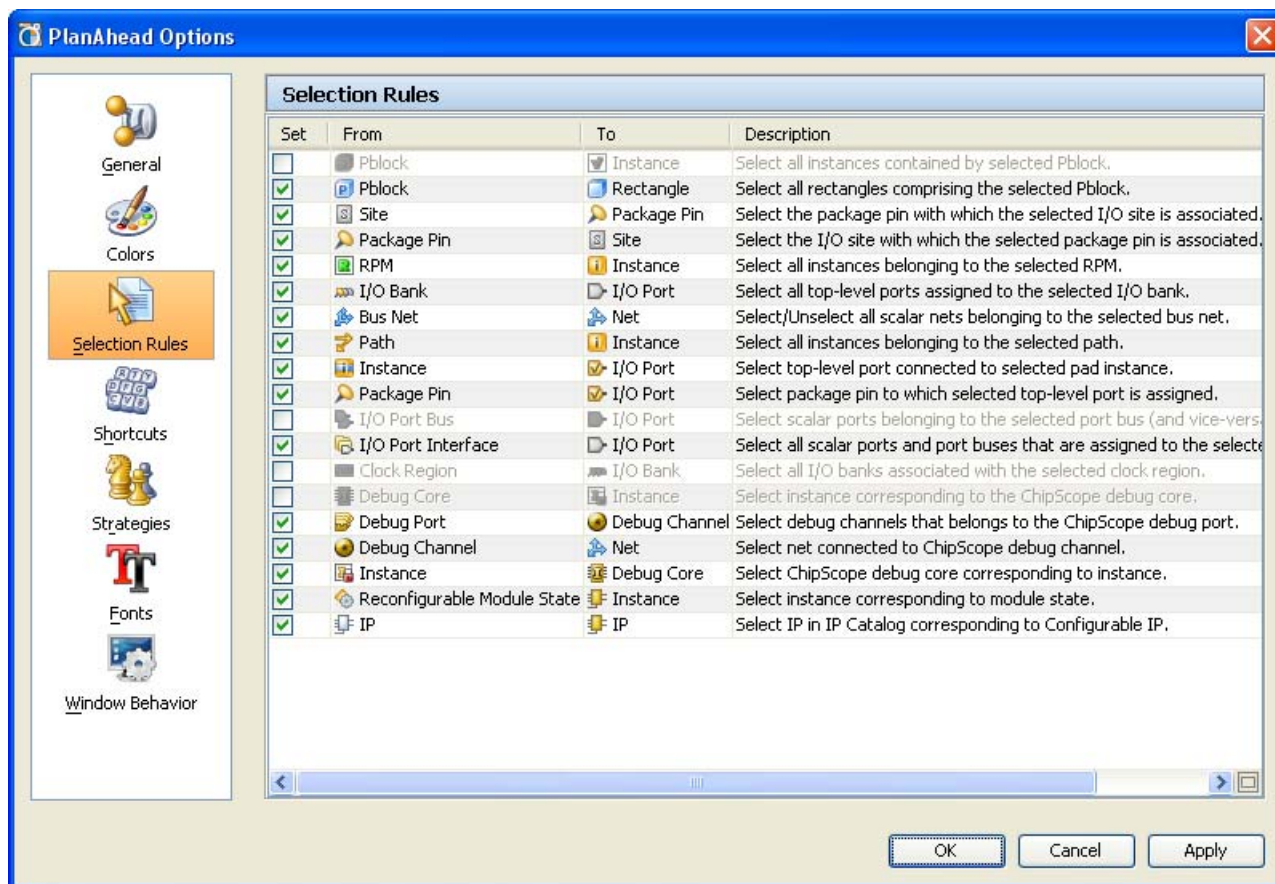


図 4-27 : [PlanAhead Options] ダイアログ ボックス : [Selection Rules] ページ

デフォルトの選択規則を使用すると、PlanAhead が最も効果的に機能します。特定のオブジェクトの選択に問題がある場合は、これらの選択規則を変更できます。

[Set] 列のチェックボックスをオン/オフにすると、自動選択のオン/オフを切り替えることができます。

- オンにすると、1 つ目の [From] 列のオブジェクトを選択したときに、関連する [To] 列のオブジェクト タイプも選択されます。
- オフにすると、[From] 列のオブジェクトを選択したときに、そのオブジェクトのみが選択されます。

[Selection] ビュー

[Selection] ビューには、135 ページの図 4-28 に示すように、現在選択されているオブジェクトのリストが表示されます。オブジェクトは、並び替えたり、選択を解除したり、マークを付けたりすることができます。このリストは、オブジェクトの操作に応じて随時更新されます。

[Window] → [Selection] をクリックすると、[Selection] ビューを開くことができます。

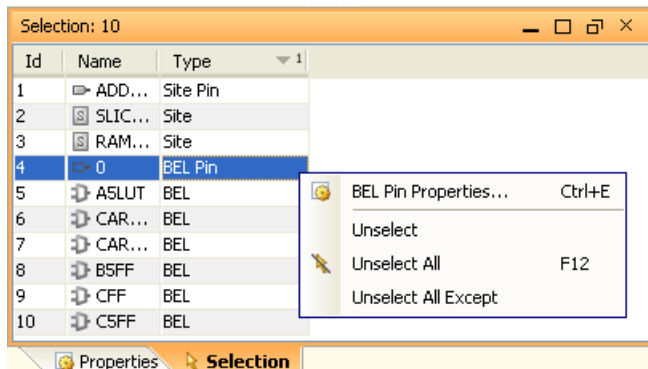


図 4-28 : [Selection] ビュー

- 列ヘッダーをクリックすると、オブジェクトを名前、ID、またはタイプで並べ替えることができます。次の図では、タイプ別に降順で表示されています。
- 選択したアイテムは、右クリックして [Unselect]、[Unselect All]、または [Unselect All Except] コマンドをクリックするとリストから削除できます。

複数のオブジェクトを選択するには、Ctrl または Shift キーを使用するか、[Select Area] コマンドを使用します。選択したオブジェクトの合計数は、ビュー バナーに表示されます。

選択したオブジェクトすべての表示

ワークスペースのビューでオブジェクトを直接またはほかのビューで間接的に選択している場合、選択したオブジェクトが中心になるよう表示をアップデートすると便利です。ワークスペースのさまざまなビューでは、次のセクションに説明するようさまざまな方法でこれを実行できます。

選択されたオブジェクトの自動表示

[Sources] ビューまたは [Netlist] ビューでオブジェクトを選択すると、ワークスペースに表示される [Device] ビューや [Schematic] ビューなどのグラフィカル ビューでも選択されます。ただし、表示されている情報が多すぎたり、拡大レベルのために、オブジェクトが小さく表示され、見づらい場合があります。

PlanAhead には、ワークスペースのグラフィカル ビューのツールバーに [Auto Fit selection] ボタンがあり、選択したオブジェクトを最大表示できます。このボタンをオンにしておくと、選択したオブジェクトが常に再描画されて表示されるようになります。オフにすると、オブジェクトを選択するたびにビューが再描画されることはなくなります。

複数のオブジェクトを選択した場合は、すべてのオブジェクトがビューに収まるよう拡大率が調整されます。

選択されたオブジェクトの最大表示

ワークスペースのビューには、選択したオブジェクトすべてがビューに収まるように表示する [Fit Selection] コマンドがあります。このコマンドは、[Auto Fit Selection] モードがオフの場合に使用します。選択したオブジェクトすべてが表示されるようにするには、次のいずれかを実行します。

- [View] → [Fit Selection] をクリックします。
- F9 キーを押します。
- ツールバーの [Fit Selection] ボタンをクリックします。



選択したオブジェクトに自動的にスクロール

[Clock Resources] ビューなどのワークスペースのビューには、選択したオブジェクトに自動的にスクロールするオプションがあります。この機能により、別のビューで選択したオブジェクトがそのビューで表示されるよう表示が常にアップデートされます。

このモードをオンにするには、[Sources] ビュー、[Netlist] ビュー、[Clock Resources] ビューのツールバーで [Automatically scroll to selected objects] ボタンをクリックします。このボタンは必要に応じてオンまたはオフにしてください。



選択したオブジェクトの移動

グラフィカルなワークスペース ビューで選択したオブジェクトを移動できます。

- オブジェクトを移動するには、次の手順に従ってください。
 - クリックしてオブジェクトを選択します。
 - オブジェクトをドラッグして移動します。
 - 新しい位置にオブジェクトをドロップします。オブジェクトを移動できる場合、カーソルが手の形に変化します。
- 複数のインスタンスを同時に移動するには、次の手順に従ってください。
 - Ctrl キーを押し続けます。
 - 何度かクリックして複数のオブジェクトを選択します。
 - ドラッグして新しい移動先にドロップします。

選択したオブジェクトのハイライト

PlanAhead には、オブジェクトを選択してハイライトする機能があります。ハイライトすると、異なる色で複数の配置グループを同時に表示できます。

選択を解除しても、ハイライトは解除されません。[Schematic] ビューも含め、該当するすべてのビューでハイライトされます。

オブジェクトをいくつでも選択してハイライトできます。

- [Device] ビューなどのグラフィカル ビューでオブジェクトを選択し、メイン メニューから [View] → [Highlight] → [Highlight Color] をクリックします。
- [Device] ビューなどのグラフィカル ビューでオブジェクトを選択し、ポップアップ メニューから [Highlight] → [Highlight Color] をクリックします。
- [Device] ビューなどのグラフィカル ビューで Pblock を選択し、ポップアップ メニューから [Highlight] → [Highlight Color] をクリックします。
- [Netlist] ビューでモジュールまたはプリミティブを選択し、ポップアップ メニューから [Highlight] → [Highlight Color] をクリックします。詳細は、[391 ページの「\[Select Primitives\] および \[Highlight Primitives\] コマンドの使用」](#)を参照してください。

選択したオブジェクトのマーク

PlanAhead では、該当するビューすべてで選択したオブジェクトにマーク シンボルを付けたり、そのマークを削除したりすることができます。

オブジェクトのマーク

マークを付けておくと、[Device] ビューで小さなオブジェクトを表示する場合に便利です。

選択したオブジェクトにマークを付けるには、メイン メニューから [View] → [Mark] をクリックするか、ポップアップ メニューから [Mark] をクリックするか、Ctrl + M キーを押します。

このコマンドは、[Netlist]、[Physical Hierarchy]、[Timing Report] など、ほかのビューでも使用できます。図 4-29 は、[Timing Report] ビューでマークされたタイミング パスを示しています。タイミング パスのスタート ポイントは緑色、エンド ポイントは赤色、スルー ポイントは黄色でマークされています。

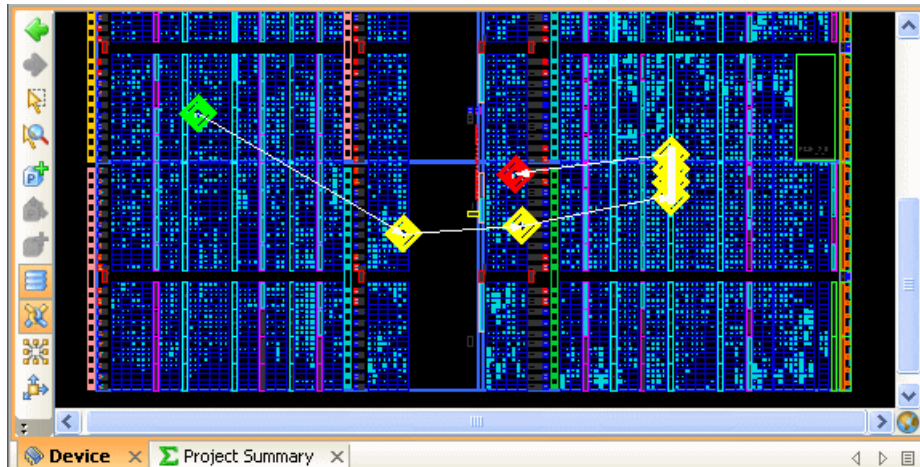


図 4-29 : [Device] ビューでマークされたタイミング パス シンボル

オブジェクトのマークの削除

選択したオブジェクトまたはすべてのオブジェクトのマークは、次のいずれかの方法で削除できます。

- 選択したインスタンスのマークを削除するには、[View] → [Unmark] をクリックするか、ポップアップ メニューから [Unmark] をクリックします。
- すべてのインスタンスのマークを削除するには、[View] → [Unmark All] をクリックするか、ツールバーの [Unmark All] をクリックします。

[Find] コマンド

PlanAhead では、[Find] コマンドを使用してインスタンスまたはネットなどの特定のデザイン オブジェクトを検索できます。[Find] コマンドを使用するには、RTL デザイン、ネットリスト デザイン、またはインプリメント済みデザインを開いている必要があります。

1. 次のいずれかを選択します。

- [Edit] → [Find] をクリックします。
- Ctrl + F キーを押します。
- メイン ツールバーの [Find] コマンドをクリックします。

138 ページの図 4-30 に示す、[Find] ダイアログ ボックスが開きます。



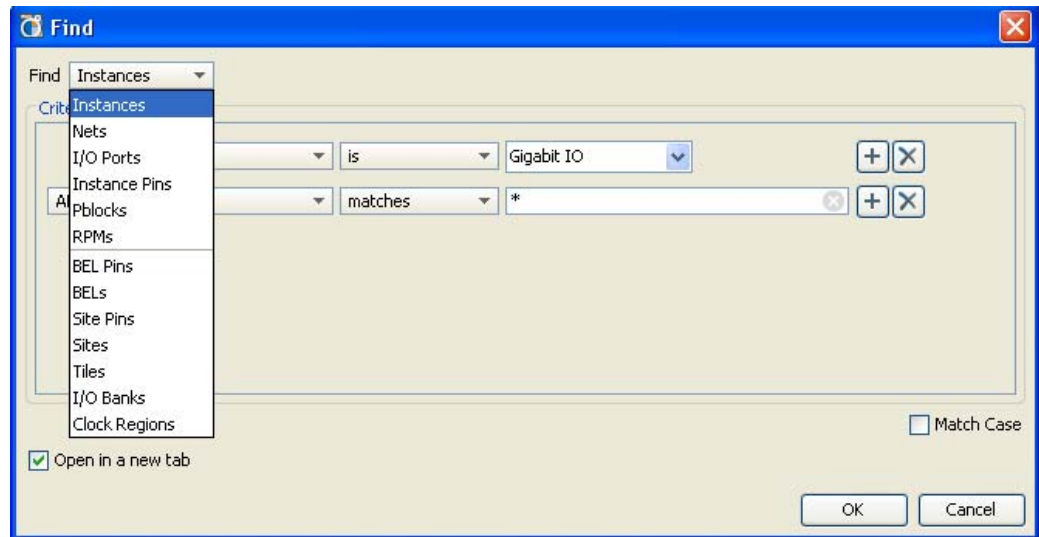


図 4-30 : [Find] ダイアログ ボックス

2. 検索フィールドは、次のように変更できます。
 - [Find]：検索する物理または論理オブジェクトのタイプ ([Instances]、[Nets]、[Pblocks]、[Sites] など) を選択します。
 - [Criteria]：選択したオブジェクトのタイプに合った検索パラメーターが表示されます。
 - 最初のフィールドでは、[Name]、[Status]、[Type]、[Parent Pblock]、[Module]、[Primitive count] などから、検索する属性を指定します。このフィールドの値は、[Find] フィールドのオブジェクト タイプによって異なります。
 - 2 番目のフィールドでは、[matches]、[doesn't match]、[contains]、[doesn't contain] から、検索条件を指定します。
 - 3 つ目のフィールドでは、検索条件に基づいた属性値を定義できます。検索文字列にはアスタリスク (*) をワイルドカードとして使用できます。
3. オプションで、[+] ボタンをクリックして検索条件を追加することもできます。

[+] ボタンをクリックすると、ダイアログ ボックスに検索条件の行が追加されます。[AND] または [OR] を選択し、条件を指定します (139 ページの図 4-31)。

新しい検索条件は、[Find] 検索条件を定義するため、前の条件に挿入される形で追加されます。最初の 2 つの検索条件が一緒にまとめられ、新しい検索条件が追加されるたびにネストされていきます。複数の検索条件を追加すると、次のようにまとめられます。

```
(((((crit1 OP crit2) OP crit3) ...)OP critN)
```

説明：

- crit1：最初の検索条件
- critN：最後の検索条件
- OP：検索条件間の関係を定義する AND または OR

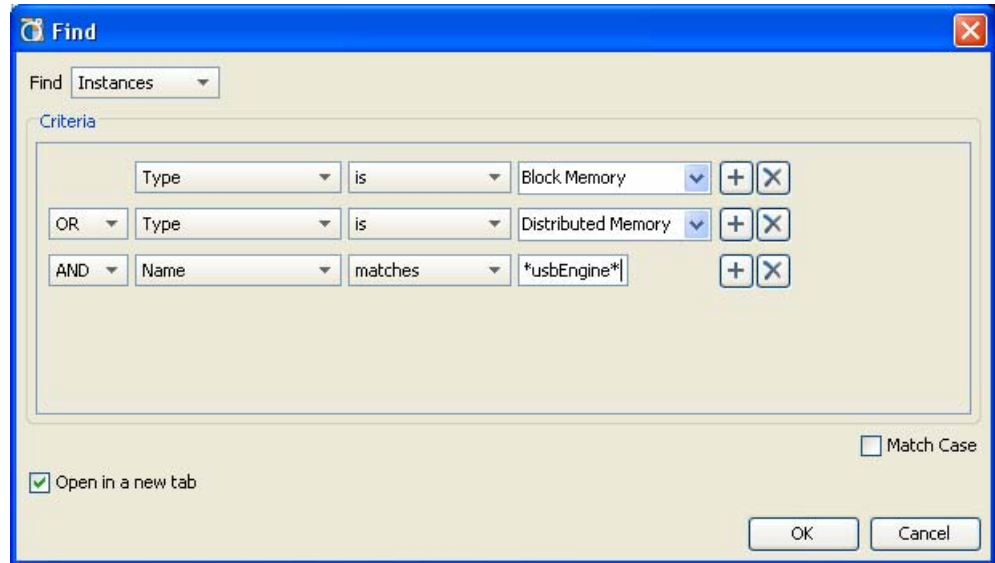


図 4-31：検索条件を絞り込んだオブジェクトの検索

図 4-31 で定義される検索条件は、次のように指定されます。

次のインスタンスを検索

((タイプがブロック メモリ OR タイプが分散メモリ) AND *usbEngine* と名前が一致)

この検索条件は、デザインのすべてのブロック メモリと分散メモリを検索し、*usbEngine* という名前の付いたもののみフィルターして表示します。

4. 条件の行を追加するには、+ または X ボタンをクリックします。
5. 大文字/小文字を指定して検索するには、[Match Case] をオンにします。
6. [Open in a new tab] をオンにすると、[Find Results] ビューの新しいタブに結果が表示されます。オフにすると、既存の結果が上書きされます。
7. [OK] をクリックして検索を実行します。

検索結果については、「[Find Results] ビュー」を参照してください。

[Find Results] ビュー

[OK] をクリックして検索を実行すると、指定した検索条件に一致するオブジェクトが [Find Results] ビューに表示されます。140 ページの図 4-32 に、[Find Results] ビューを示します。

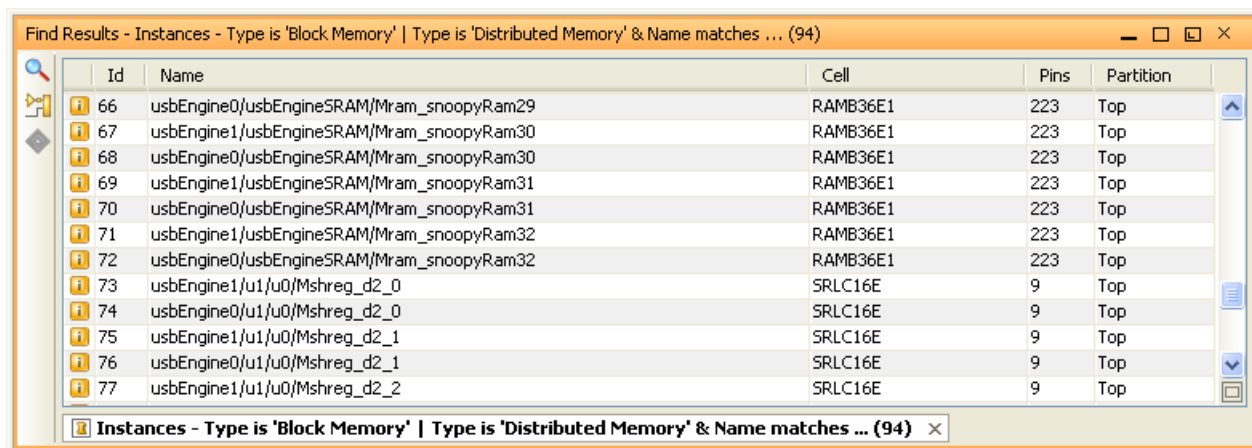


図 4-32 : [Find Results] ビュー

[Find] コマンドを実行するたびに、[Find Results] ビューに新しいタブが追加されます。タブには検索したオブジェクトとその検出数が表示されます。

オブジェクトは、[Find Results] ビューから直接選択できます。検出されたオブジェクトのリストからオブジェクトを選択すると、ほかのビューでも選択されます。**Shift** キーまたは **Ctrl** キーを使用すると、複数のエレメントを選択できます。その他のコマンドは、ポップアップ メニューから実行できます。

列は、次のように分類できます。

- 列のヘッダーをクリックします。
- **Ctrl** キーを押して別の列ヘッダーをクリックすると、次にその列で並べ替えられます。

検索結果名の隣の **X** マークをクリックすると [Find Results] ビューが閉じます。

ソース ファイルでの検索および置換

[Find] コマンドを使用すると、ネットやインスタンスなどの特定のデザイン オブジェクトを検索できますが、[Find]、[Find in Files]、[Replace in Files] コマンドを使用すると、ソース ファイルのテキスト文字を検索することもできます。

[Find] コマンドは開いている 1 つのソース ファイル内での検索に使用できます。テキスト エディターのポップアップ メニューまたはツールバー アイコンから [Find] をクリックします。

[Find] ツールバーがテキスト エディターの一番下に開きます。ソース ファイルで検索するテキスト文字を入力します。開いたファイル内で該当する文字すべてをハイライトすることもできます。

図 4-33 は、[Find] コマンド ツールバーの例を示しています。

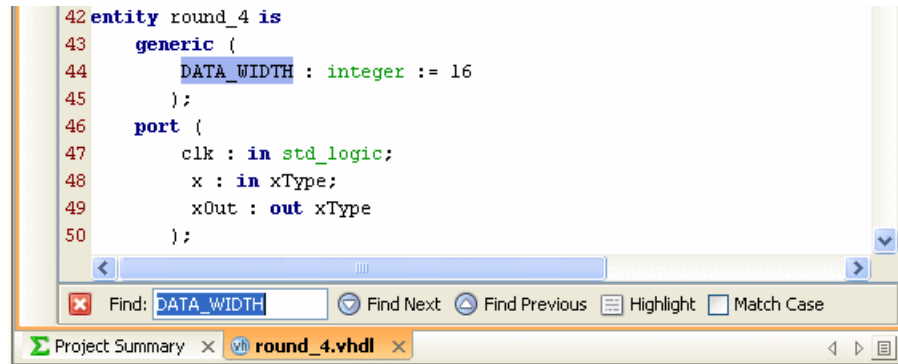


図 4-33 : 開いたソース ファイル内での検索機能の使用

[Find in Files] コマンドを使用すると、開いているソース ファイルまたは指定したファイル内で文字列を検索できます。テキスト エディターのポップアップ メニューまたはツールバー アイコンから [Find in Files] コマンドをクリックするか、メイン メニューから [Edit] → [Find in Files] をクリックします。

次の操作を実行できます。

- 検索条件として、ワイルドカード (*) を含む任意のテキスト文字列を入力できます。
- フィルター オプションを使用して、ソース ファイル、制約ファイル、レポート ファイルを検索できます。

図 4-34 に、[Find in Files] ダイアログ ボックスを示します。

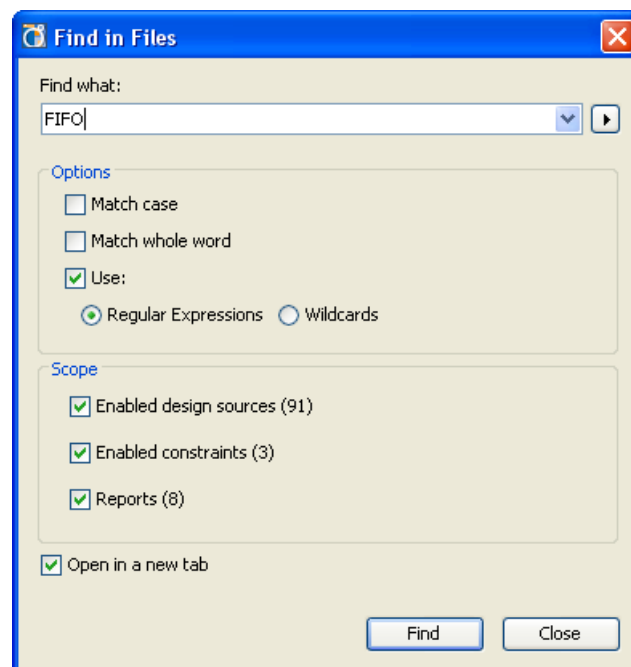


図 4-34 : [Find in Files] ダイアログ ボックス

[Find] コマンドを使用すると、指定したテキスト文字を検索できます。

1. [Find What] フィールドにテキスト文字を入力します。
2. 必要に応じて次のオプションをオンにします。

- [Match case]：文字の大文字/小文字の一致した文字を検索します。
- [Match whole word]：入力文字とすべてが一致する文字を検索します。
- [Use]：正規表現またはワイルドカードに基づいた検索をします。
- [Scope]：デザイン ソース、制約ファイル、開いているレポートで検索します。
- [Open in a new tab]：結果を新しいビューで開くか、既存ビューを置き換えるか指定します。

3. [Find] をクリックします。

検索結果は [Find in Files] ビューに表示され、検索文字列を含むファイルと文字列の発生回数が表示されます。

[Find Results] ビューでファイル名をクリックすると、ソース ファイルがテキスト エディターに読み込まれ、該当文字がハイライトされます。142 ページの図 4-35 は、検索後の [Find Results] ビューの例です。

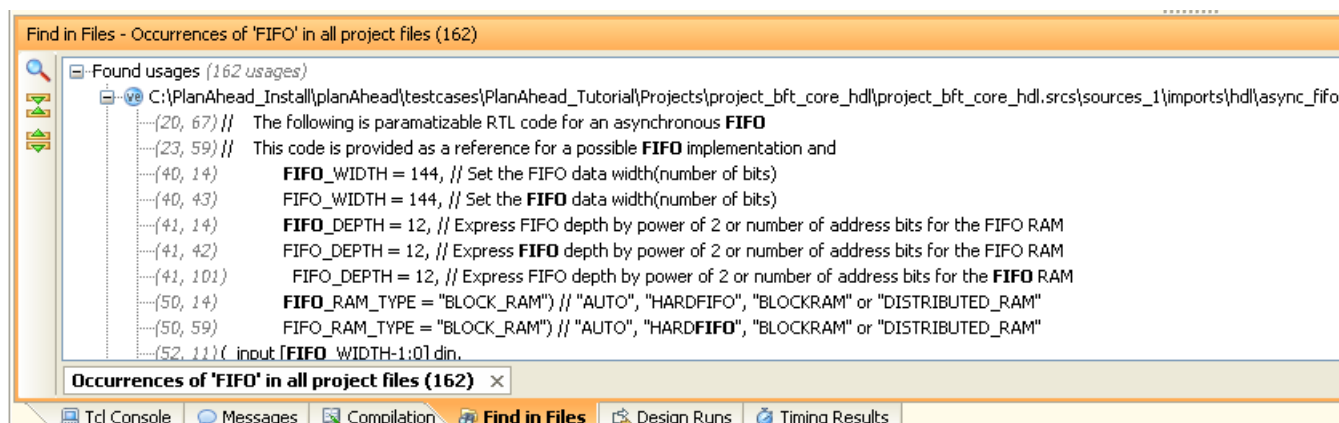


図 4-35 : [Find Results] ビュー

[Replace in Files] コマンドを使用すると、選択したファイル内で文字を検索して、別の文字に置換できます。テキスト エディターのポップアップ メニューから [Replace in Files] コマンドをクリックするか、メイン メニューから [Edit] → [Replace in Files] をクリックします。

143 ページの図 4-36 に、[Replace in Files] ダイアログ ボックスを示します。

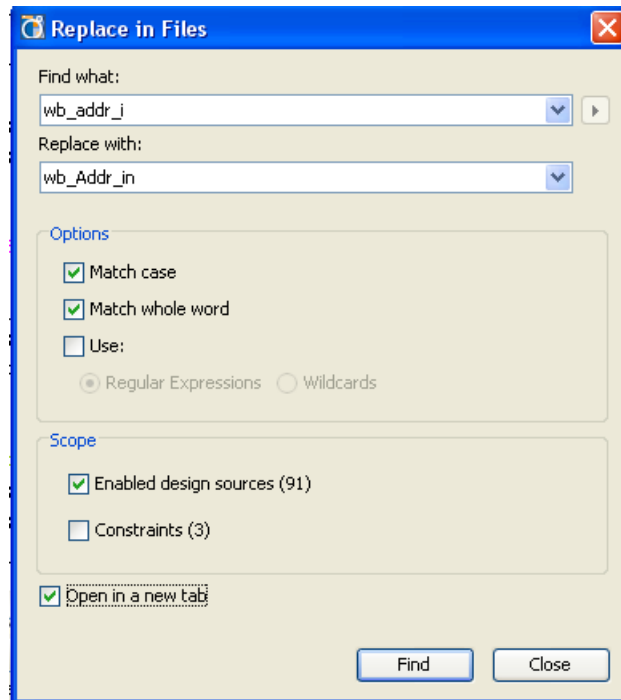


図 4-36 : [Replace in Files] ダイアログ ボックス

[Replace in Files] コマンドを使用すると、指定したテキスト文字を検索して置換できます。

1. [Find What] フィールドにテキスト文字を入力します。
2. [Replace with] フィールドに置換する文字を入力します。
3. 必要に応じて次のオプションをオンにします。
 - [Match case] : 文字の大文字/小文字の一致した文字を検索します。
 - [Match whole word] : 入力文字とすべてが一致する文字を検索します。
 - [Use] : 正規表現またはワイルドカードに基づいた検索をします。
 - [Scope] : デザイン ソース、制約ファイル、開いているレポートで検索します。
 - [Open in a new tab] : 結果を新しいビューで開くか、既存ビューを置き換えるか指定します。
4. [Find] をクリックします。

検索結果は [Find in Files] ビューに表示され、検索文字列を含むファイルと文字列の発生回数が表示されます。

144 ページの図 4-37 は、検索結果の例です。

- [Replace Selected] をクリックすると、リストの選択したテキストが置換できます。
- [Replace All] をクリックすると、指定したテキストすべてを置換できます。
- [Cancel] で検索をキャンセルできます。
- [Find Results] ビューのテキストをクリックすると、テキスト エディターでそれがハイライトされます。

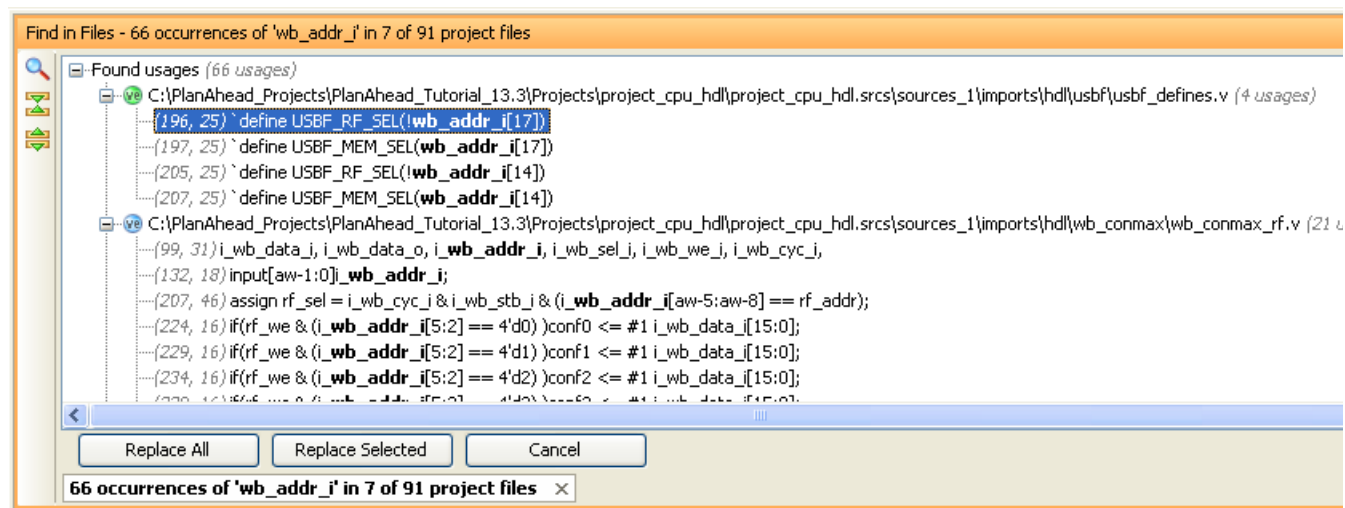


図 4-37 : [Replace in Files] の結果

汎用のビューの使用

次のセクションでは、次のビューについて説明します。

[Sources] ビュー

デザイン ソース、制約ファイル、シミュレーション ソース、および IP コアをプロジェクトに追加すると、プロジェクトの [Sources] ビューに表示されます。[Sources] ビューを使用すると、デザインの要件に合わせてプロジェクト ソース ファイルの管理、追加、削除、並べ替えを実行できます。

[Design Sources] フォルダには、Verilog、VHDL、NGC/NGO、EDIF、IP コア、DSP モジュール、エンベデッド プロセッサなどのソース ファイル タイプが含まれます。制約ファイルは制約セットに割り当てられ、[Constraints] フォルダの下に表示されます。詳細は、61 ページの「プロジェクト ソースの管理」を参照してください。

注記：モジュール レベルのネットリスト制約ファイル (NCF) は、デザイン ソースとしてコアと共に表示され、読み出し専用です。

通常は、PlanAhead でプロジェクトを開くと [Sources] ビューが開きます。[Sources] ビューを開くには、[Window] → [Sources] をクリックします。図 4-38 に、[Sources] ビューの例を示します。

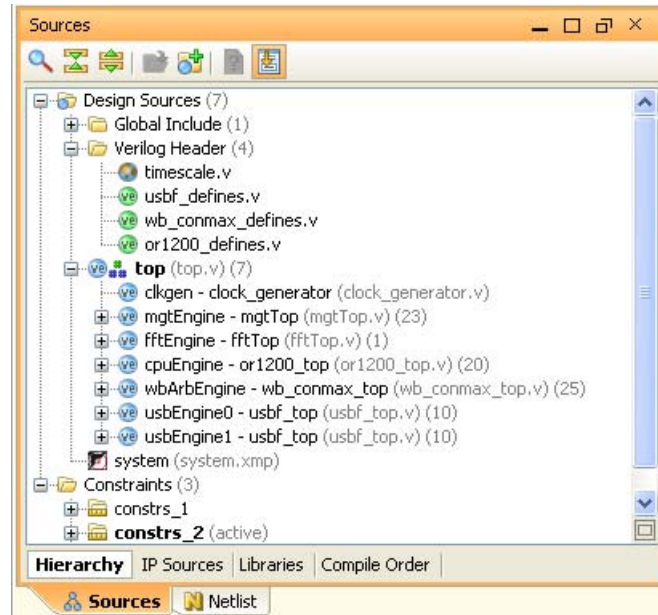


図 4-38 : [Sources] ビュー

[Sources] ビューには、ソース ファイルをさまざまな方法で表示するタブが 4 つ含まれます。

- [Hierarchy] タブには、最上位モジュールを含むデザイン ソースの階層ビューが表示されます。最上位モジュールでは、コンパイル、合成、インプリメンテーション用のデザイン階層が定義されます。[Sources] ビューでは、最上位モジュールのアイコンで表示されます (図 4-38)。PlanAhead では自動的に最上位モジュールが識別されますが、[Set as Top] コマンドを使用して手動で最上位モジュールを定義することもできます (147 ページの「[Sources] ビューのポップアップ メニュー」)。

注記 : [Sources] ビューのツールバー メニューの [Filter Sources by missing files or instantiations] コマンドは、デザイン階層にファイル、モジュール、またはモジュールのインスタンスエーションが存在しない場合に使用可能になります。このコマンドは、存在しない情報をデバッグする際に便利です。









- [IP Sources] タブには、次が表示されます。
 - ザイリンクス IP カタログからインポートされた IP コア。詳細は、75 ページの「IP コアの管理」を参照してください。
 - System Generator からインポートされた DSP モジュール。詳細は、84 ページの「DSP ソースの管理」を参照してください。
 - XPS からインポートされたエンベデッド プロセッサ デザイン。詳細は、87 ページの「エンベデッド プロセッサ ソースの管理」を参照してください。
- [Libraries] タブには、さまざまなライブラリ別にソースが分類されて表示されます。ソース ファイル名の横の緑の丸は、そのソース ファイルがローカル プロジェクト ディレクトリにコピーされていることを示します。ソース ファイルが赤色で表示される場合は、PlanAhead ソフトウェアで必要なファイルを見つけることができなかったことを示します。
- [Compile Order] タブでは、コンパイル順にソース ファイルが表示されます。この場合、最上位モジュールは通常最後にリストされます。定義した最上位モジュールとエラボレート済みデザインに基づいて、コンパイル順を自動的に付けるようにできます。また、[Hierarchy Update] コマンドを設定してソース ファイルを並び替えて、コンパイル順を手動で変更することもできます (147 ページの「[Sources] ビューのポップアップ メニュー」)。

[Sources] ビューのコマンド

[Sources] ビューにはツールバー ボタンと、マウスを右クリックすると表示されるポップアップ メニューがあります。[Sources] ビューのツールバーおよびポップアップ メニューを使用して、ソース ファイルを追加、表示、変更できます。

[Sources] ビューのツールバー ボタン

[Sources] ビューのツールバーには、次のコマンドがあります (145 ページの図 4-38)。

- [Show Search] : [Search] フィールドを表示し、[Sources] ビューのオブジェクトを検索します。このフィールドは、キーボードで **Alt + /** キーを押して開くこともできます。
- [Expand All] : [Sources] ビューの階層ツリーをすべて展開表示し、すべてのエレメントを表示します。
- [Collapse All] : [Sources] ビューの階層ツリーをすべて閉じ、最上位オブジェクトのみを表示します。
- [Open Selected Source Files] : 選択した RTL ソース ファイルまたは制約ファイルがテキスト エディターで開きます。選択した IP コアは CORE Generator で、DSP モジュールは System Generator で、エンベデッド プロセッサは XPS で開きます。詳細は、第 3 章「プロジェクトの操作」または第 5 章の「テキスト エディターの使用」を参照してください。
- [Add Sources] : RTL ソース ファイル、シミュレーション ソース ファイル、制約ファイル、または既存の IP をプロジェクトに追加または作成したり、既存の IP、DSP モジュール、エンベデッド プロセッサなどを追加できます。
- [Filter Sources by missing files or instantiations] : ソースをフィルターして存在しないファイルまたはインスタンスを表示します。このコマンドは、デザイン階層にファイル、モジュール、またはインスタンス化が存在しない場合に使用可能になります。このコマンドをオンにすると、[Sources] ビューがフィルターされて存在しないファイルやモジュールが表示されるので、問題の原因を見つけやすくなります。コマンドのアイコンが淡色表示になっている場合は、デザイン階層に問題がなかったことを示します。
- [Automatically Scroll to Selected Object] : 選択されたオブジェクトが表示されるよう [Sources] ビューの表示が自動的にアップデートされるよう設定します。この機能は、多数のソース ファイルを含む大型デザインで便利です。デフォルトでオンになっています。

[Sources] ビューのポップアップ メニュー

[Sources] ビューのポップアップ メニューには、次のコマンドがあります。

- [Source File Properties] : [Source File Properties] ビューを開きます。詳細は、[151 ページの「\[Source File Properties\] ビュー」](#)を参照してください。
注記：このコマンドは、[Hierarchy] タブでは [Source Node Properties] です。
- [Copy Text] : 選択したテキストをコピーして、別の場所に貼り付けられるようにします。このコマンドを使用すると、インスタンスのファイル名を [Tcl Console] ビューにコピーできます。
- [Open File] : 選択した RTL または制約ファイルをテキスト エディターで開きます。選択した IP コアは CORE Generator で、DSP モジュールは System Generator で、エンベデッド プロセッサは XPS で開きます。
- [Update File Contents] : 選択したファイルを新しく選択したファイルに置き換えます。詳細は、[第 3 章の「ローカル ソース ファイルのアップデート」](#)を参照してください。
- [Replace File] : 指定したソース ファイルを別のファイルと置き換えます。
- [Copy File into Project] : 選択したソース ファイルおよびディレクトリをプロジェクト ディレクトリにコピーします。このコマンドは、選択したソース ファイルがプロジェクトに対してローカルにない場合にのみ使用できます。詳細は、[第 3 章の「リモート ソースの参照またはプロジェクト ディレクトリへのソースのコピー」](#)を参照してください。
- [Copy All Files Into Project] : リモートから参照されているソース ファイルをすべてプロジェクト ディレクトリにコピーします。このコマンドは、選択したソース ファイルがプロジェクトに対してローカルにない場合にのみ使用できます。
- [Remove File from Project] : 選択したソース ファイルをプロジェクトから削除し、オプションでローカルプロジェクト ディレクトリからファイルを削除します。
- [Enable File] : ソース ファイルのステータスをアクティブに設定します。ステータスは、イネーブルまたはディスエーブルを切り替えることができます。
注記：このイネーブル設定は、[Source File Properties] ビューからも指定できます ([151 ページの「\[Source File Properties\] ビュー」](#))。
 - [Disable File] : ソース ファイルのステータスを非アクティブに設定します。ステータスは、イネーブルまたはディスエーブルを切り替えることができます。ステータスがディスエーブルのソース ファイルは、[Sources] ビューでグレーで表示されます。
- [Move to Simulation Sources] : 選択したソース ファイルをシミュレーション セットに移動します。複数のシミュレーション セットが存在する場合は、シミュレーション セットを選択するダイアログ ボックスが表示されます。
- [Move to Top] : [Compile Order] タブで選択したソース ファイルをリストの一番上に移動します。ソース ファイルのコンパイルおよび合成は、[Compile Order] タブのリストの上から下に

	Source Node Properties...	Ctrl+E
	Copy Text	Ctrl+C
	Open File	Alt+O
	Launch Timing Simulation	
	Update File Contents...	Alt+U
	Replace File...	
	Copy File Into Project	
	Copy All Files Into Project	Alt+I
	Remove File from Project...	Delete
	Enable File	Alt+=
	Disable File	Alt+-
	Move to Simulation Sources	
	Hierarchy Update	
	Refresh Hierarchy	
	Set as Top	
	Set Global Include	
	Clear Global Include	
	Set Library...	Alt+L
	Set File Type...	
	Set Used In...	
	Add Sources...	Alt+A
	Edit Constraint Sets...	
	Find in Files...	Ctrl+Shift+F

向かって処理されるので、ファイル順はエラボレーション、合成、シミュレーションの結果に影響します。

[Compile Order] タブのファイル順は、次に示すように、[Hierarchy Update] コマンドの設定によって自動的にアップデートされるか、手動でアップデートできます。

注記：[Move] コマンドは、[Sources] ビューの [Hierarchy] タブからは使用できません ([Move to Simulation Sources] コマンドは例外)。

- [Move Up]：選択したソース ファイルを リストの 1 つ上に移動します。
- [Move Down]：選択したソース ファイルを リストの 1 つ下に移動します。
- [Move to Bottom]：選択したソース ファイルを リストの一番下に移動します。
- [Hierarchy Update]：最上位モジュールの再定義、ファイルの追加または削除、ファイル順の変更といったソース ファイルの変更を PlanAhead でどのように処理するか指定します。このコマンドには、次の 3 つのオプションがあります。

<input checked="" type="checkbox"/>	Automatic Update and Compile Order
<input type="checkbox"/>	Automatic Update, Manual Compile Order
<input type="checkbox"/>	No Update, Manual Compile Order
- [Automatic Update and Compile Order]：ソース ファイルが変更されたら、デザインの階層表示とコンパイル順が自動的にアップデートされます。PlanAhead では、最上位モジュールに最適なモジュールが自動的に認識されて設定されます。アクティブな階層の下で最上位モジュール ファイルおよびすべてのソース ファイルが正しい順序で合成およびシミュレーションに使用されるように、コンパイル順序も自動的に管理されます。階層の外部にあるファイルが最上位モジュールとして定義されている場合は、送信されません。
- [Automatic Update, Manual Compile Order]：ソース ファイルが変更されたら、デザインの階層表示は自動的にアップデートされますが、コンパイル順は手動で指定します。プロジェクトのすべてのファイルが合成およびシミュレーションで使用されます。このコンパイル順は、[Compile Order] タブで [Move to Top]、[Move Up]、[Move Down]、[Move to Bottom] を使用すると手動で定義できます。
- [No Update, Manual Compile Order]：デザインの階層表示は自動的にアップデートされません。コンパイル順は手動で指定します。このモードでデザイン階層をアップデートするには、[Refresh Hierarchy] コマンドを使用します。
- [Refresh Hierarchy]：最後のソース ファイルの変更と最上位モジュールの定義が反映されるようにデザイン階層をアップデートします。このコマンドは、階層を手動で更新する場合に必要なに応じて使用してください。
- [Set as Top]：合成およびシミュレーション プロセスにおける、デザイン階層のエラボレーションの開始点となる最上位モジュールを指定します。詳細は、71 ページの「最上位モジュールの指定とソース ファイルの順序の変更」を参照してください。

[Sources] ビューの [Hierarchy] タブでは、最上位モジュールは専用のアイコンで表示されます。

注記：選択した最上位モジュールがデザイン ソース ファイルで見つからない場合は、自動的に最適なモジュールになり、階層アップデート モードは automatic に設定されます。

- [Set Global Include]：指定したファイルをグローバル インクルード ファイルとして設定します。このオプションは、Verilog ソース ファイルでのみ使用可能です。詳細は、224 ページの「グローバル インクルード ファイルの定義」を参照してください。

注記：この設定は、[Source File Properties] ビューからも指定できます (151 ページの「[Source File Properties] ビュー」)。

- **[Clear Global Include]** : 選択した Verilog ソース ファイルからグローバル インクルード 属性を削除します。
- **[Make active]** : 選択した制約セットを、合成またはインプリメンテーションに使用するようアクティブにします。
- **[Set Target Constraint]** : 複数の制約ファイルを含む制約セットの場合、新しく作成された制約を書き込む制約ファイルを指定できます。詳細は、第 3 章の「制約の管理」を参照してください。
- **[Set Library]** : 選択した RTL ソース ファイルのライブラリを選択します。現在プロジェクトで定義されているライブラリ のリストから選択するか、テキスト フィールドに新しいライブラリを入力します。新しいライブラリを入力すると、それが現在定義されているライブラリのリストに追加されます。
- **[Set Type]** : 選択しているファイルのタイプを定義します。PlanAhead では、ファイル タイプが自動的に認識され、適切なファイル拡張子に基づいてプロジェクトへ追加されますが、認識できないファイルの場合は、**[Set Type]** コマンドでそのファイル タイプを定義できます。

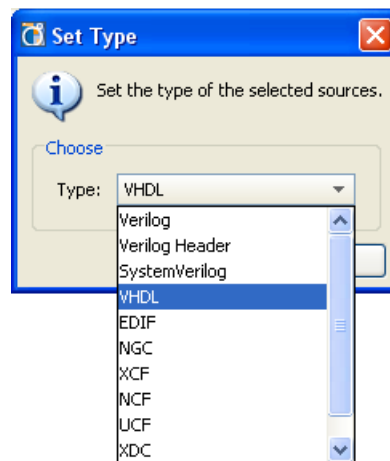


図 4-39 : [Set Type] コマンド

- **[Set Used In]** : ファイルが何に使用されるかを指定します。合成、シミュレーション、インプリメンテーションで使用する (または使用しない) ソース ファイルを指定できます。特定ツール用のソース ファイルをオフにしておくと、そのファイルはツールで指定されません。

たとえば、ソース ファイルを合成で指定されないように指定しておいてから、エラボレート済み デザインを開くと、そのソース ファイルの部分がブラック ボックスで表示されます。インプリメンテーションの EDIF または NGC ソース ファイルをオフにしておくと、それらがインプリメンテーションでは使用されません。

ただし、VHDL または Verilog ソース ファイルが合成でオン、インプリメンテーションでオフになっていると、ソースは合成されてネットリストが出力され、インプリメンテーションで使用されます。RTL ソースがインプリメンテーション中に使用されないようにするには、合成でそのソース ファイルをオフにする必要があります。

注記 : [Library]、[Type]、[Used In] は、[Source File Properties] ビューから指定できます (「[Source File Properties] ビュー」)。

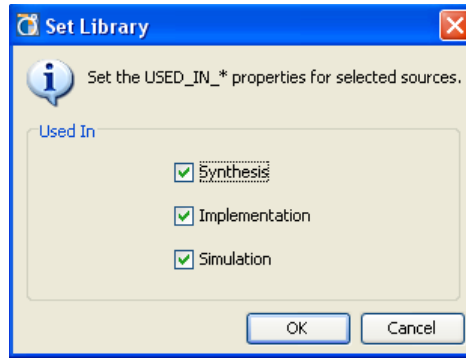


図 4-40 : [Set Used In] コマンド

- IP ソース関連のコマンド : [Sources] ビューの [IP Sources] タブで IP を右クリックすると、次のコマンドが使用できます。詳細は、[75 ページの「IP コアの管理」](#)を参照してください。
 - [Generate] : IP コアのターゲット データを必要に応じて生成します。
 - [Reset] : 現在のターゲット データを削除し、必要に応じて再生成できるようにします。
 - [Re-customize IP] : IP コアに対して CORE Generator インターフェイスを開いて、IP の属性を修正できるようにします。
 - [Upgrade IP] : IP の最新のバージョンがある場合に、古いバージョンからアップグレードします。
 - [Open Example Project] : コアのサンプル ターゲット データが提供されていて生成されている場合は、その IP コアのサンプル プロジェクトを開きます。
 - [View Datasheet] : IP カタログから PDF 形式のデータシートを開きます。
 - [View Version Information] : IP カタログから HTML 形式の情報を開きます。
 - [View Product Web Page] : IP コアのウェブページが存在する場合はそれを表示します。
 - [View Answer Record] : IP コアに関連するアンサー レコードのサポート データベースを検索して表示します。
- DSP およびエンベデッド デザイン ソース関連のコマンド : [Sources] ビューで DSP モジュールまたはエンベデッド サブデザインを選択して右クリックすると、次のコマンドが使用できます。詳細は、[84 ページの「DSP ソースの管理」](#)および[87 ページの「エンベデッド プロセッサ ソースの管理」](#)を参照してください。
 - [Create Top HDL] : 選択した DSP またはエンベデッド プロセッサを含む最上位 Verilog または VHDL モジュールを作成します。
 - [View Instantiation Template] : DSP またはエンベデッド プロセッサのインスタンシエーション テンプレートを開いて、別の RTL ファイルへインスタンシエートします。
 - [Create Testbench] : 選択した DSP モジュールまたはエンベデッド プロセッサ デザインのシミュレーション テストベンチを作成します。
 - [Generate] : DSP モジュールまたはエンベデッド プロセッサ デザインのターゲット データを必要に応じて生成します。
 - [Reset] : 現在のターゲット データを削除し、必要に応じて再生成できるようにします。
- [Add Sources] : RTL ソース ファイル、シミュレーション ソース ファイル、制約ファイル、既存の IP、DSP モジュール、エンベデッド プロセッサ サブデザインなどを作成または追加できます。詳細は、[第 3 章の「プロジェクト ソースの管理」](#)を参照してください。
- [Edit Constraint Sets] : 制約セットを作成または変更できます。

- [Find in Files] : [Find in Files] ダイアログ ボックスを開き、文字列を入力して選択したファイルから検索します。検索結果は、[Find in Files] ビューに表示されます。詳細は、[140 ページの「ソース ファイルでの検索および置換」](#)を参照してください。

[Source File Properties] ビュー

[Sources] ビューで RTL ソース ファイルを選択して、[Source File Properties] ビューにファイルの情報を表示できます。ソース ファイル プロパティを表示するには、次の手順に従います。

1. [Sources] ビューでソース ファイルを右クリックし、[Source File Properties] をクリックします。[151 ページの図 4-41](#) に、Verilog ファイルの [Source File Properties] ビューを示します。

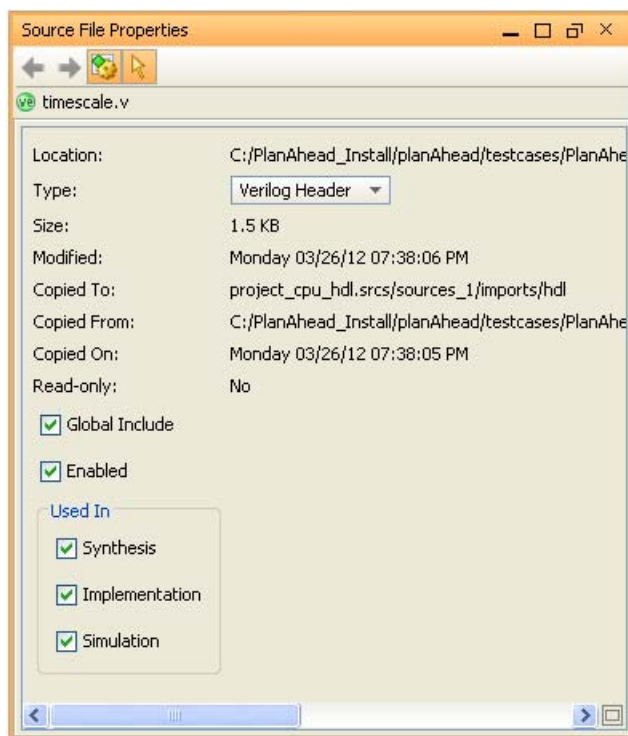


図 4-41 : [Source File Properties] ビューでの変更の表示

このビューに表示されるファイル情報は、場所、タイプ、ライブラリ、サイズ、最後に変更された日時、コピー元、コピー日時、および親モジュールです。

- ファイル タイプは [Type] オプションを使用して変更できます。これは、ファイルの拡張子が一般的でない場合やファイルが正しく検出されなかった場合に便利なオプションです。
- ソース ファイルのターゲット ライブラリを変更するには、[Library] フィールドをクリックしてドロップダウン リストから選択するか、ライブラリ名を直接入力します。
- Verilog ソース ファイルをグローバル インクルード ファイルとして指定する場合は、[Global Include] をオンにします。[Global Include] をオンにすると、エラボレーションおよび合成で選択したファイルが最初にコンパイルされます。詳細は、[第 6 章の「グローバル インクルード ファイルの定義」](#)を参照してください。
- [Enabled] をオフにすると、選択したソース ファイルがデザインでディスエーブルになります。ディスエーブルに設定したファイルもソース ファイルとしてプロジェクトに存在していますが、デザインのエラボレーションおよびコンパイルには含まれません。

- 合成、シミュレーション、インプリメンテーションで使用する (または使用しない) ソース ファイルを指定できます。特定ツール用のソース ファイルをオフにしておくと、そのファイルはツールで指定されません。[Used In] フィールドの詳細については、[146 ページの「\[Sources\] ビューのコマンド」](#)を参照してください。
- 2. [Source File Properties] ビューに加えた変更を適用する場合は [Apply] を、破棄する場合は [Cancel] をクリックします。

[Device] ビュー

[Device] ビューは、デザイン解析およびフロアプランに関するさまざまな操作で使用するメインのグラフィカル インターフェイスです。詳細は、次を参照してください。

- [第 7 章「合成済みデザインの制約および解析」](#)
- [第 8 章「I/O ピン配置」](#)
- [第 11 章「インプリメンテーション結果の解析」](#)
- [第 10 章「デザインのフロアプラン」](#)
- [Device] ビューには、FPGA ロジック、クロック領域、I/O パッド、BUFG、DCM、Pblock、インスタンス ロケーション、ネット接続など、FPGA デバイスのリソースが表示されます。特定のロジックを割り当てることができるロケーションは、「サイト」と呼ばれます。[図 4-42](#) に、[Device] ビューを示します。

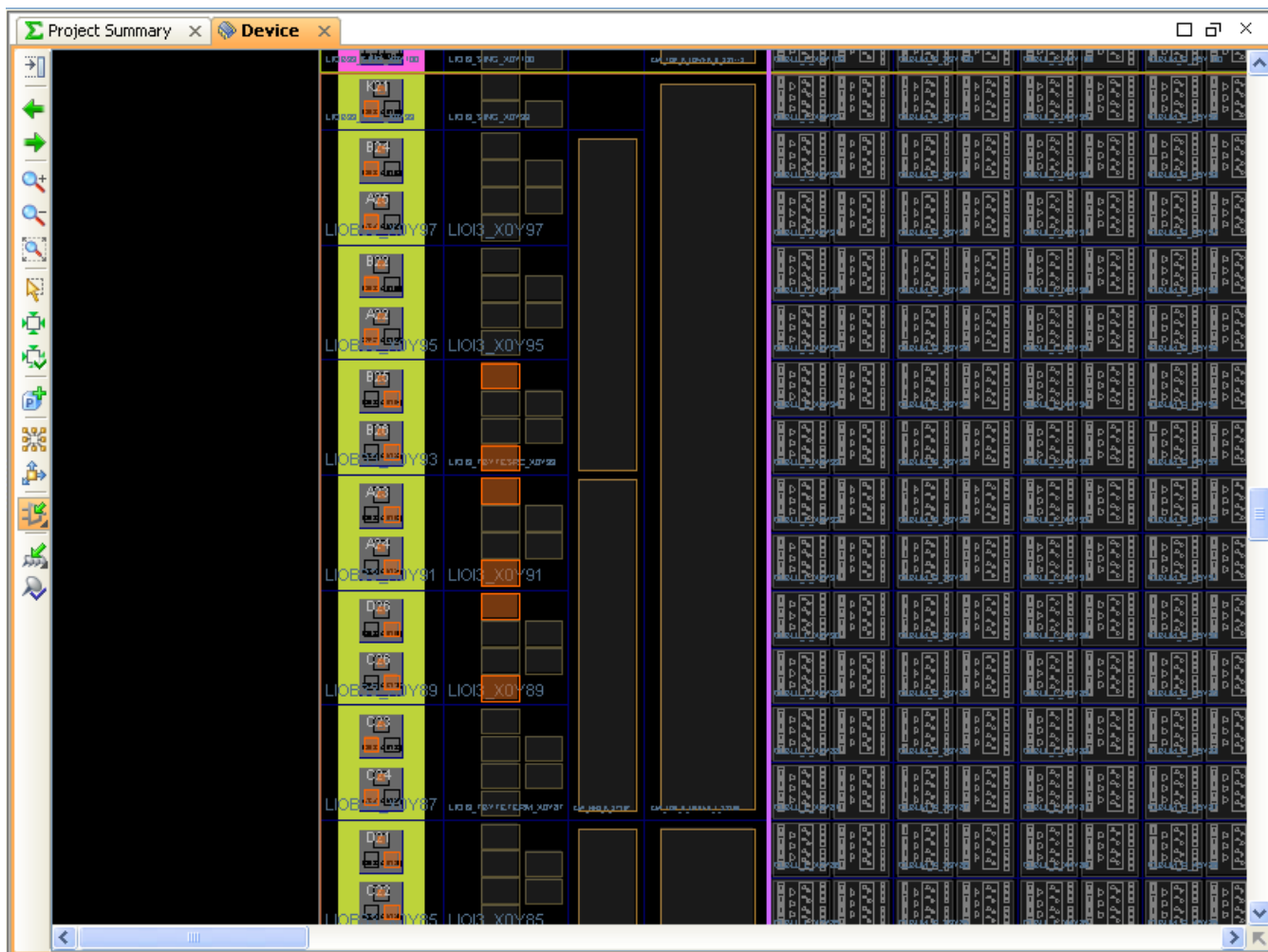


図 4-42 : [Device] ビュー

表示されるロジック オブジェクトの詳細は、選択しているズーム レベルにより異なります。拡大していくと、ロジック オブジェクトがより詳細に表示されます。[Device] ビューを右クリックして表示されるポップアップ メニューとツールバーに、ズーム レベルを変更するコマンドが含まれています。

[Device] ビューでマウスをクリックしてカーソルをドラッグしても、エリアを拡大または縮小できます。詳細は、[125 ページの「マウスを使用した拡大/縮小」](#)を参照してください。


















また、[Device] ビューのスクロール バーやダイナミックな表示移動機能を使用し、表示箇所に移動できます。

[Device] ビューのオブジェクト上にカーソルを置くと、ツール ヒントにオブジェクトの名前が表示されます。サイトまたはロジック オブジェクトを選択すると、[Properties] ビューにプロパティが表示されます。デバイス リソース サイトを検索するには [Edit] → [Find] をクリックします。

たとえば、ロジック リソースを不正に割り当てようとすると、ダイナミック カーソルが変化するので、操作を調整できます。詳細は、[124 ページの「文脈依存カーソルについて」](#)を参照してください。

[Device] ビューのコマンド

[Device] ビューの左側にあるツールバー ボタンには、次のようなコマンドがあります。

- **Device View Options** — [Device] ビューでの表示、色、特定のレイヤーを設定できます。詳細は、[155 ページの「\[Device\] ビューの表示オプションの設定」](#)を参照してください。 
- **[Previous Position]** : 前の拡大レベルと座標に [Device] ビューの表示を戻します。 
- **[Next Position]** : [Previous Position] を使用した後に [Device] ビューの表示を元の拡大レベルと座標に戻します。 
- **[Zoom In]** : [Device] ビューを 2 倍ずつ拡大します。 
- **[Zoom Out]** : [Device] ビューを 1/2 ずつ縮小します。 
- **[Zoom Fit]** : デバイス全体を [Device] ビューに表示します。 
- **[Select Area]** : 指定した四角形のエリア内のオブジェクトを選択します。 
- **[Fit Selection]** : [Device] ビューで現在選択しているオブジェクトを表示します。これは、[Netlist] ビューなどの別のビューでオブジェクトを選択し、そのオブジェクトの表示を再描画する際に便利です。 
- **[Autofit Selection]** : 新しく選択したオブジェクトを [Device] ビューで自動的に再描画します。このモードは、オンまたはオフにできます。 
- **[Draw Pblock]** : インスタンスを配置する Pblock の四角形を新しく作成します。詳細は、[343 ページの「Pblock の概要」](#)を参照してください。 
- **[Show I/O Nets]** : 配置した LOC または Pblock への I/O 接続の表示を切り替えます。 
- **[Show Instance Connections]** : このモードをオンにすると、オブジェクトを選択したときに接続が表示されます。このボタンでモードのオン/オフを切り替えます。 
- **[Instance Drag & Drop Mode]** : — デバイスに配置するインスタンスの配置制約をドラッグアンドドロップで指定します。
 - **[Create BEL Constraint Mode]** : LOC および BEL 制約を配置されるインスタンスに割り当てます。これにより、インスタンスがスライス内の指定した BEL に固定できます。 
 - **[Create Site Constraint Mode]** : LOC 配置制約を配置されるインスタンスに割り当てます。これにより、インスタンスは指定したスライスに固定されますが、スライス内の使用可能な BEL に対してフロート状態にできます。 
 - **[Assign Instance to Pblock Mode]** : ロジック インスタンスを Pblock に割り当てます。これがデフォルト モードです。コマンドが正しく機能するようにするため、できる限りこのモードを使用してください。 
- **[Place Ports Mode]** : [I/O Ports] ビューまたは [Netlist] ビューから I/O ポートがデバイスに配置される方法を指定します。 
 - **[Place I/O Ports in an I/O Bank]** : 現在選択されているポートを指定した I/O バンクのピンに割り当てます。
 - **[Place I/O Ports in Area]** : 現在選択されているポートを指定したエリアのピンに割り当てます。
 - **[Place I/O Ports Sequentially]** : 現在選択されているポートをピンにそれぞれ割り当てます。
- **[Autocheck I/O Placement]** : 対話型 I/O 配置 DRC のオン/オフを切り替えます。オンになっていると、対話型 I/O 配置がオンになっているデザイン ルールに対してチェックボックスされます。 

デバイス リソースの表示

[Device] ビューには、選択したデバイスに含まれるさまざまなリソースが表示されます。デバイス特有のリソースは、すべてグラフィカル サイトで表示されます。デバイス リソースの表示レベルは、[Device] ビューのズーム レベルによって変わります。スライス リソースなどのリソースは、FPGA ロジックをかなり拡大しないと見えるようになりません。クロック領域や I/O バンクなどのリソースは、デバイス全体を表示していても見えます。また、[Device View Options] 1 コマンドを使用して、特定のオブジェクトやリソースの表示/非表示を切り替えることができます。詳細は、[155 ページの「\[Device\] ビューの表示オプションの設定」](#)を参照してください。

[Device] ビューでは、リソースは次のように表示されます。

- I/O パッドおよびクロック オブジェクトは、デバイスの周囲と、デバイスの中央の列に表示されます。
 - I/O バンクは、I/O パッドの列のすぐ外側に細い長方形として影付きで表示されます。
 - ザイリンクス 7 シリーズ FPGA では、高パフォーマンス (HP) および広範囲 (HR) I/O バンクが導入されており、[Device] に右上がりの斜線および左上がりの斜線として表示されます。
 - 使用可能な I/O バンク サイトは、I/O バンクの長方形が色で塗りつぶされています。
 - デバイスの中には、ボンディングされていない I/O バンク (白い X の付いた四角形) を含むものもあります。
 - I/O クロック パッドは色で塗りつぶされた長方形として表示されます。
- BUFG、BUFGCTRL、BUFR、BUFHCE などのクロック リソースも、[Device] ビューに表示されます。I/O バンクまたはクロック領域を選択すると、[I/O Bank Properties] ビューに使用可能なデバイス リソースが表示されます。
- デバイスの内側は、タイルと呼ばれる小さな長方形に分割されています。タイルは、そのアーキテクチャのさまざまなロジック プリミティブのタイプを配置できるサイトです。
[Device] ビューでロジック サイトの上にカーソルを置くと、ツール ヒントが表示されます。

デバイス リソースの表示に関する詳細は、[237 ページの「リソース統計の表示とレポート」](#)を参照してください。CLB、SLICE、および BEL を表示するには、表示を拡大する必要があります。

次の操作を実行できます。

- プリミティブ ロジック インスタンスを、表示されている適切なサイトに割り当てます。
- ISE による配置結果をインポートして、ロジックの割り当てを表示します。

配置されたインスタンスは、スライス内で長方形として表示されます。さらに拡大表示すると、ロジック シンボルが表示されます。ロジックは特定のサイトに割り当てることができ、割り当てると LOC 配置制約が生成されます。BEL 制約を使用すると、インスタンスをスライス内の特定リソースに割り当てることができます。

ISE からインポートされたロジックは、BEL レベルの制約として表示されます。配置制約の詳細は、[365 ページの「LOC 制約および BEL 制約の設定」](#)を参照してください。

[Device] ビューの表示オプションの設定

ザイリンクス 7 シリーズ FPGA、Virtex®-6 ファミリーおよび Spartan®-6 ファミリーの場合、[Device] ビューのツールバーに次の機能を持つ [Device View Options] コマンドが含まれます。



- [Layers] : [Device] ビューに表示するデバイスおよびデザイン オブジェクトを指定します。このコマンドを実行すると、[Device] ビューでの表示レベルを制御できます。これは、表示に情

報が多すぎる場合に特に便利です。図 4-43 は、[Device View Options] の [Layer] タブを示しています。

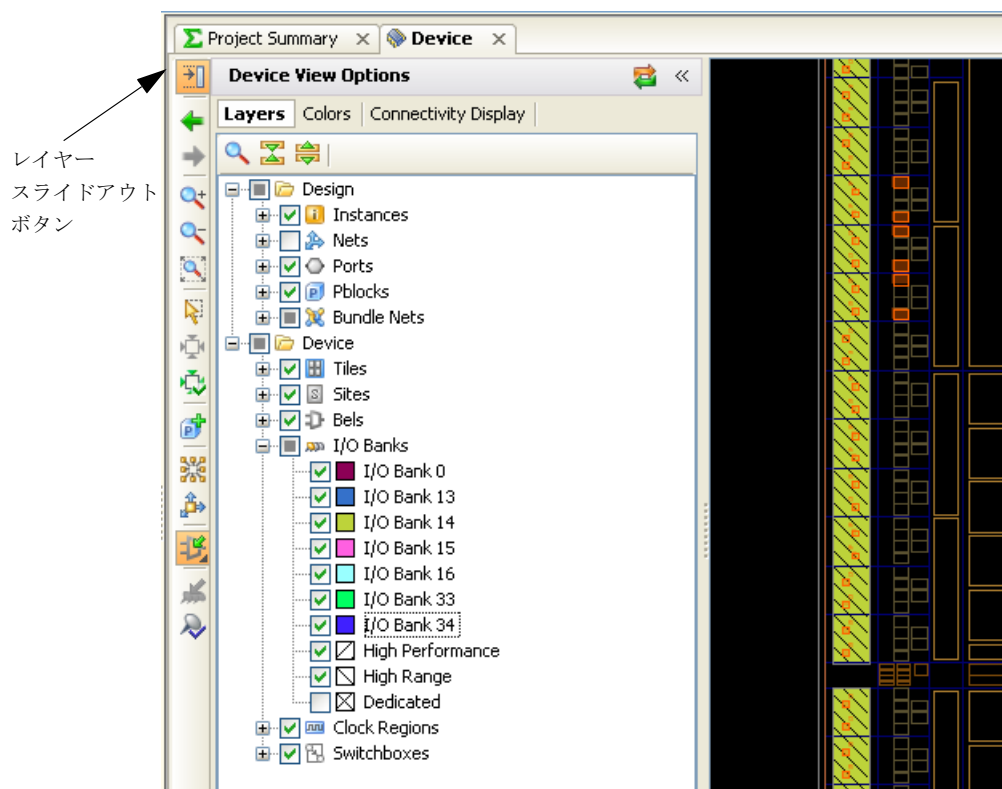


図 4-43 : [Device View Layers] ボタン

主な 2 つのカテゴリはデザイン オブジェクトとデバイス オブジェクトです。

- デザイン オブジェクトは、デバイスに配置されるインスタンス、ネット、ポートなどのデザイン ソースからのエレメントです。
- デバイス オブジェクトは、デザイン オブジェクトを配置可能な I/O バンク、クロック領域、タイルなどのデバイス上のリソースです。

プラス記号 (+) をクリックして階層を展開表示したり、マイナス記号 (-) をクリックして階層を閉じたりして、レイヤーおよびオブジェクトの階層表示を調整します。

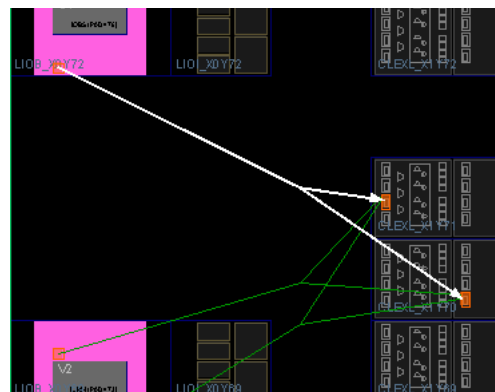
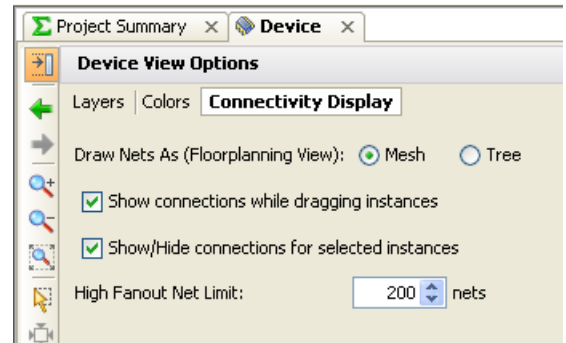
レイヤーまたはオブジェクトの横にあるチェック ボックスのオン/オフを切り替えて、[Device] ビューに表示されるものを選択します。

チェック ボックスがオンのレイヤーが表示されます。レイヤーのカテゴリのチェック ボックスをクリックしてオブジェクトまたはレイヤーのグループの表示/非表示を切り替えるか、個々のレイヤーまたはオブジェクトのチェック ボックスをクリックして個々の表示/非表示を切り替えることができます。

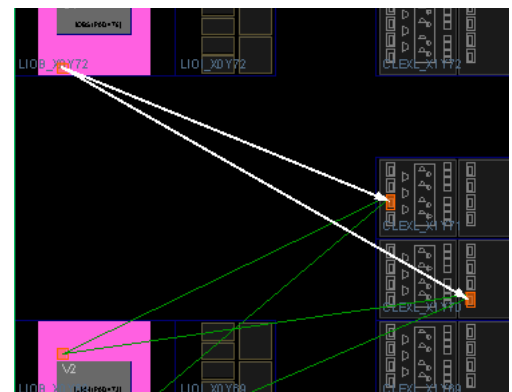
注記： [Device] ビューで特定のオブジェクトまたはレイヤーが見えない場合は、[Device View Options] ボタンをクリックして、そのデザイン オブジェクトまたはデバイス リソースが非表示になっていないかどうかを確認してください。

- [Colors] : [Device] ビューの要素の色は、次のように変更できます。
- [Color] 列をクリックし、ドロップダウン リストに表示される色から選択します。
- [More Colors] をクリックすると、より多数の色から選択できます。
- [Color] 列のセルに直接 RGB 値を入力します。
- [Connectivity Display] : デバイスのネット接続の表示特性を指定します。
 - [Draw Nets as] : 図 4-44 に示すように、メッシュかツリーに指定します。
 - [Mesh] : 接続されたピンすべてをメッシュにするピン間の接続
 - [Tree] : 近くのピン接続をまとめた分岐構造
 - [Show connections while dragging instances] : [Device] ビューでインスタンスをドラッグして配置するときに、選択したインスタンスへの接続されるネットの表示を切り替えます。
 - [Show/Hide connections for selected instances] : 選択したインスタンスへ接続されるネットの表示を切り替えます。
 - [High Fanout Net Limit] : 1 つのピンに対して表示される接続本数を制限します。指定した接続数よりもファンアウトが多い場合、ピンのネットは表示されません。
 - [Reset] : [Device View Options] の [Layers]、[Colors]、または [Connectivity Display] を PlanAhead ツールのデフォルトの設定にリセットします。

Object Type	Frame Color	Fill Color
Background		0, 0, 0
Foreground		255, 255, 255
Selection		255, 255, 255
Markers		255, 255, 0
Pblock 1st Level	204, 102, 255	
Pblock 2nd Level	153, 51, 255	170, 151, 189
Pblock 3rd+ Levels	102, 0, 204	225, 196, 255
Assigned Instance	143, 131, 1	
I/O Net	0, 153, 0	
Placed Port	0, 255, 255	0, 255, 255



ツリー



メッシュ

図 4-44 : [Draw Nets as] オプション

[Device View Options] には、[Show Search]、[Expand All]、[Collapse All] ボタンがあります。

[Device View Options] での設定が終了したら、[<<] ボタンをクリックして [Device View Options] を閉じます。[Device View Options] の設定は保存されるので、ツールを起動するたびにその設定を読み込むことができます。

クロック領域の選択

クロック領域は、さまざまなデバイス クロック領域を示す大きな長方形として表示されます。この境界線は、クリティカルな回路のフロアプランに役立ちます。

[Device] ビューでは、次の操作を実行できます。

- [Clock Regions] ビューでクロック領域を選択します。
- クロック領域を選択して、[Clock Region Properties] ビューにリソース統計を表示します。
- インプリメンテーション結果をインポートした後、クロック配置統計を表示します。
- [Device] ビューに表示されるクロック領域の表示色を変更するには、[Tools] → [Options] → [Themes] → [Device] をクリックします。

クロック領域を選択すると、関連する I/O バンクおよびクロック関連のロジック サイトも選択されます。

クロック領域統計の表示については、[287 ページの「クロック領域のリソースおよび統計」](#)を参照してください。

複数の [Device] ビューの表示

同じフロアプランに対して、複数の [Device] ビューを開くことができます。複数表示することで、デバイスの異なるエリアで作業できます。複数の [Device] ビューを開く場合は、[126 ページの「ワークスペースの分割」](#)を参照してください。図 4-45 に、2 つの [Device] ビューを分割表示した例を示します。

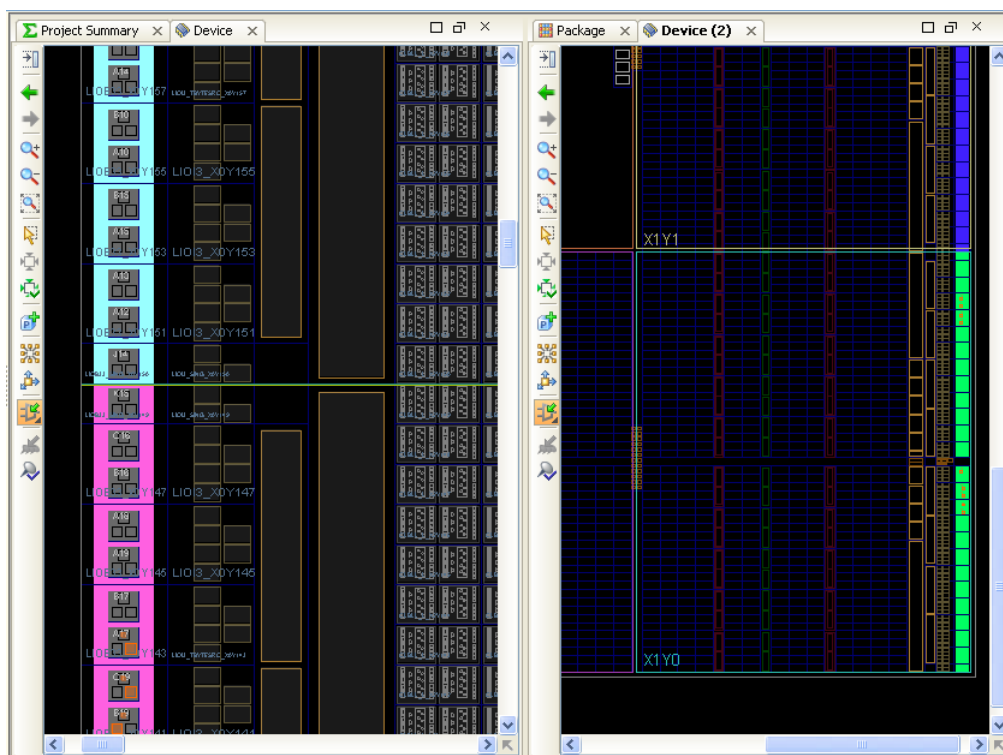


図 4-45：複数の [Device] ビューの表示

[Package] ビュー

[Package] ビューには、ターゲット サイリンクス パーツの物理特性が表示されます。このビューは、主に I/O ピン配置またはポート配置で使われます。各ピン タイプは、異なる色と形を使用して表示されます。[Package] ビューは、[I/O Planning] ビュー レイアウトを選択すると自動的に開きます。[Package] ビューを使用した I/O ピン配置の詳細は、[第 8 章「I/O ピン配置」](#)を参照してください。

[Package] ビューは、メイン メニューから [Window] → [Package] をクリックすると手動で開くことができます。[Package] ビューは、[160 ページの図 4-46](#) のように表示されます。

[Package] ビューでは、次の操作を実行できます。

- ポートを [Package] ビューにドラッグし、配置済みインスタンスをほかの I/O ピンに割り当て直します。
- ピンおよび I/O バンクは、次のように表示されます。
 - VCC は赤色の正方形、GND ピンは緑色の正方形で表示されます。
 - クロック ピンは、六角形で表示されます。
 - デバイスの I/O バンクごとに表示される領域の色は異なります。
- カーソルを [Package] ビュー上で移動すると、ビューの上と左に I/O ピンの座標が示されます。
- I/O ピンまたはバンクを選択すると、[Device] ビューと [Package] ビュー間をクロスプローブでき、[Pins Properties] ビューでピン情報を確認できます。
- カーソルをピン上に置くと、ツール ヒントにピン情報が表示されます。その他の I/O ピンおよびバンク情報は、メイン ウィンドウの下部にある情報バーに表示されます。

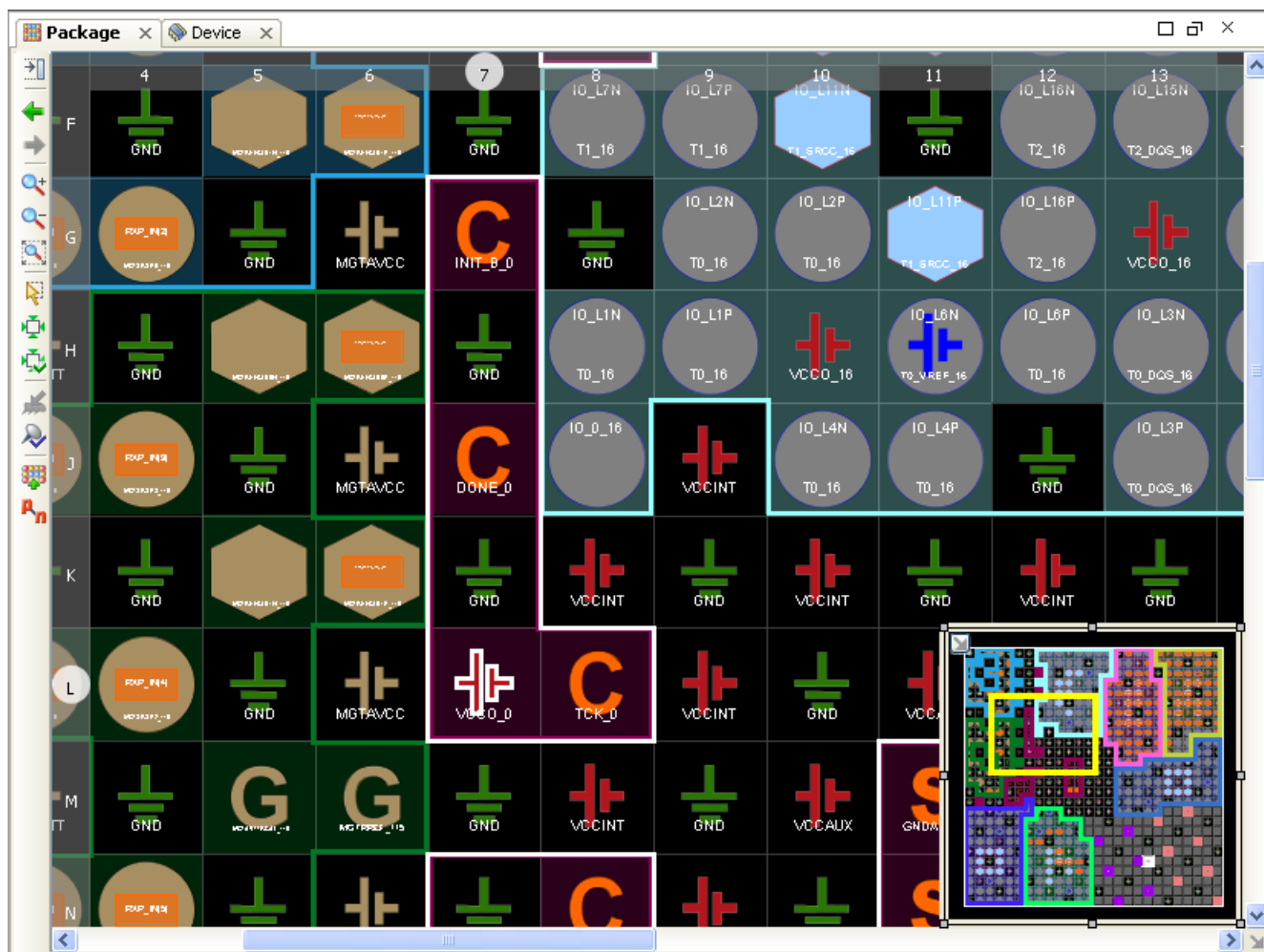







図 4-46 : [Package] ビュー

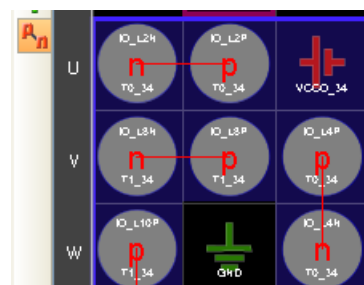
[Package] ビューのコマンド

[Package] ビューの左側にあるツールバー ボタンには、次のようなコマンドがあります。

- [Package View Options] [Package] ビューでの特定オブジェクトの表示を設定できません。詳細は、[161 ページの「\[Package View Options\] の設定」](#)を参照してください。
- [Previous Position] : 前の拡大レベルと座標に [Package] ビューの表示を戻します。
- [Next Position] : [Previous Position] を使用した後に [Package] ビューの表示を元の拡大レベルと座標に戻します。
- [Zoom In] : [Package] ビューを 2 倍ずつ拡大します。
- [Zoom Out] : [Package] ビューを 1/2 ずつ縮小します。
- [Zoom Fit] : デバイス全体を [Package] ビューに表示します。
- [Select Area] : 指定した四角形のエリア内のオブジェクトを選択します。
- [Fit Selection] : [Package] ビューで現在選択しているオブジェクトを表示します。これは、[Netlist] ビューなどの別のビューでオブジェクトを選択し、そのオブジェクトの表示を再描画する際に便利です。



- **[Autofit Selection]** : 新しく選択したオブジェクトを **[Package]** ビューで自動的に再描画します。このモードは、オンまたはオフにできます。 
- **[Place Ports Mode]** : **[I/O Ports]** ビューまたは **[Netlist]** ビューから **I/O** ポートがデバイスに配置される方法を指定します。 
- **[Place I/O Ports in an I/O Bank]** : 現在選択されているポートを指定した **I/O** バンクのピンに割り当てます。
- **[Place I/O Ports in Area]** : 現在選択されているポートを指定したエリアのピンに割り当てます。
- **[Place I/O Ports Sequentially]** : 現在選択されているポートをピンにそれぞれ割り当てます。
- **[Autocheck I/O Placement]** : 対話型 **I/O** 配置 **DRC** のオン/オフを切り替えます。オンになっていると、対話型 **I/O** 配置がオンになっているデザイン ルールに対してチェックボックスされます。 
- **[Show Bottom]/[Top View]** : **[Package]** ビューを上部から表示するか下部から表示するか指定します。 
- **[Show Differential I/O Pairs]** : **[Package]** ビューで差動ペアピンを表示します。 



[Package View Options] の設定

[Package] ビューのツールバーには [Package View Options] コマンドがあり、[Package] ビューに表示するレイヤーおよびオブジェクトを選択できます。162 ページの図 4-47 は、表示/非表示にできる [Package] ビューのレイヤーを含む [Package View Options] ペインを示しています。

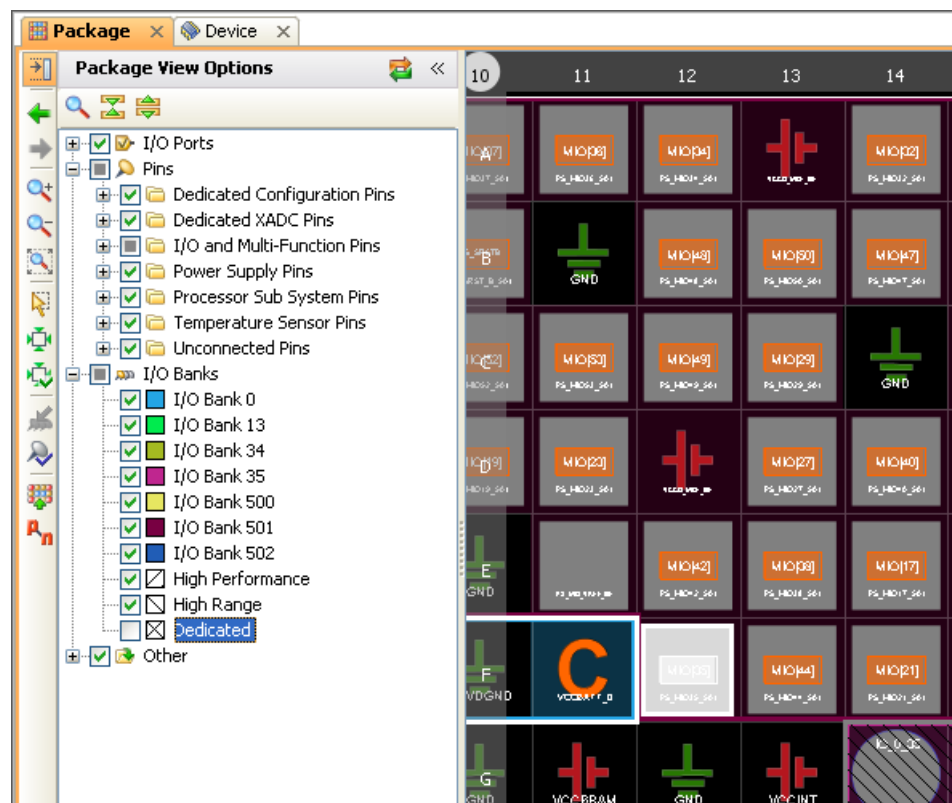


図 4-47 : [Package View Options] ペイン

レイヤーはツリー形式で階層表示され、階層を展開表示したり閉じたりできます。[Package] ビューに表示されるレイヤーは、I/O ポート、ピン、I/O バンクに分類できます。

- I/O ポートは、現在デザインに配置されているポートを表示します。固定されているものと固定されていないものの両方が表示されます。配置されていないポートは、[Package] ビューには表示されません。
- ピンにはパッケージピンが含まれ、多目的ピン、電源ピン、未接続ピンなどに分類されています。
- 電源ピンは I/O バンクとは別に表示されます。
- 多目的ピンは含まれる I/O バンクの一部として、使用可能な機能を示すシンボルで表示されます。
 - クロック兼用ピンは、青い六角形で表示されます。
 - Vref ピンには、電源マークが表示されます。

ツールバーの [Package View Options] ボタンをクリックすると、多目的ピンに使用されるシンボルがリストされます。

- Zynq™ デバイスに含まれる PSSIO (Processor Sub System Pins) は、[Package Pin Properties] ビューに示されるように、PlanAhead ツールでは読み出し専用のピンです。これらのピンは、Zynq デバイス上ではプログラマブル ロジック (PL) ピンとしてコンフィギュレーションできませんが、プロセッサ システム (PS) システムとしてはコンフィギュレーション可能で、XPS からインポートできます。詳細は、[XPS ヘルプ](#)を参照してください。

- I/O バンクには、デバイス上の各バンクのピンのサイトと GTX ピンのサイトが表示されます。各 I/O バンクと GT バンクは色分け表示されており、異なるバンクのピンを簡単に見分けることができます。
- ザイリンクス 7 シリーズ FPGA では、通常高パフォーマンス (HP) および広範囲 (HR) I/O バンクが導入されています。HP および HR の I/O バンクは、I/O バンクの機能として表示されます。

[Package] ビューの特定のピンの表示は、[Package View Options] のピンを表すレイヤーの組み合わせによって異なります。I/O バンクが表示されていない場合、多目的ピンはレイヤー スライドアウトで選択されていても表示されませんが、電源ピンは表示されます。

プラス記号 (+) をクリックして階層を展開表示したり、マイナス記号 (-) をクリックして階層を閉じたりして、レイヤーおよびオブジェクトの階層表示を調整します。レイヤーの横にあるチェックボックスのオン/オフを切り替えて、[Package] ビューに表示されるものを選択します。チェックボックスがオンのレイヤーが表示されます。

次のような操作が可能です。

- オブジェクトのカテゴリのチェック ボックスをオン/オフにすることにより、オブジェクトのグループを表示/非表示
- 単独のアイテムをオン/オフにすることにより、そのオブジェクトのみを表示/非表示

注記： 特定ピンが [Package] ビューに表示されない場合は、そのピンにポートを割り当てることができません。ピンとそれを含む I/O ブロックの両方が [Package View Options] で表示されるように設定されているかどうか確認してください。

複数の [Package] ビューの表示

同じデザインに対して、複数の [Package] ビューを開くことができます。複数表示することで、パッケージの異なるエリアで作業できます。複数の [Package] ビューを開く場合は、[126 ページの「ワークスペースの分割」](#)を参照してください。[158 ページの図 4-45](#) に、2 つの [Device] ビューを分割表示した例を示します。


[Schematic] ビュー

回路図は、どのレベルの論理階層または物理階層に対しても表示できます。[Netlist] ビューでプリミティブやネットなどのロジック エレメントを右クリックし、[Schematic] をクリックすると、選択したオブジェクトの回路図が作成されます。エラボレート済みデザインを開くと、[164 ページの図 4-48](#)のようにデザインの最上位の回路図も常に表示されます。

[Schematic] ビューを使用すると、デザインのインターコネクトや階層構造を表示したり、エラボレート済みデザイン、合成済みデザイン、またはインプリメント済みデザインの信号パスをトレースしたりできます。

- RTL ネットリストの解析については、[第 6 章の「合成結果の解析」](#)を参照してください。
- 合成済みネットリストの解析については、[第 7 章「合成済みデザインの制約および解析」](#)を参照してください。

[Schematic] ウィンドウを開くには、次の手順に従います。

1. [Netlist] ビューのような開いているビューで、1 つまたは複数のロジック エレメントを選択します。
2. 右クリックして [Schematic] をクリックするか、ツールバーの [Schematic] ボタンをクリックします。 

[Schematic] ビューには、選択したロジックのインスタンスまたはネットが表示されます。インスタンスを 1 つのみ選択している場合は、165 ページの図 4-49 に示すように、そのモジュールの回路図シンボルが表示されます。

回路図シートの上の [Instances]、[I/O Ports]、[Nets] というリンクをクリックすると、[Find Results] ビューに検索可能なリストが開き、回路図内で特定のアイテムが検索しやすくなります。

[Schematic] ビューでオブジェクトを選択すると、その他すべてのビューでも選択されます。インプリメント済みデザインが開いている場合は、インスタンスおよびネットが [Device] ビューに表示されます。

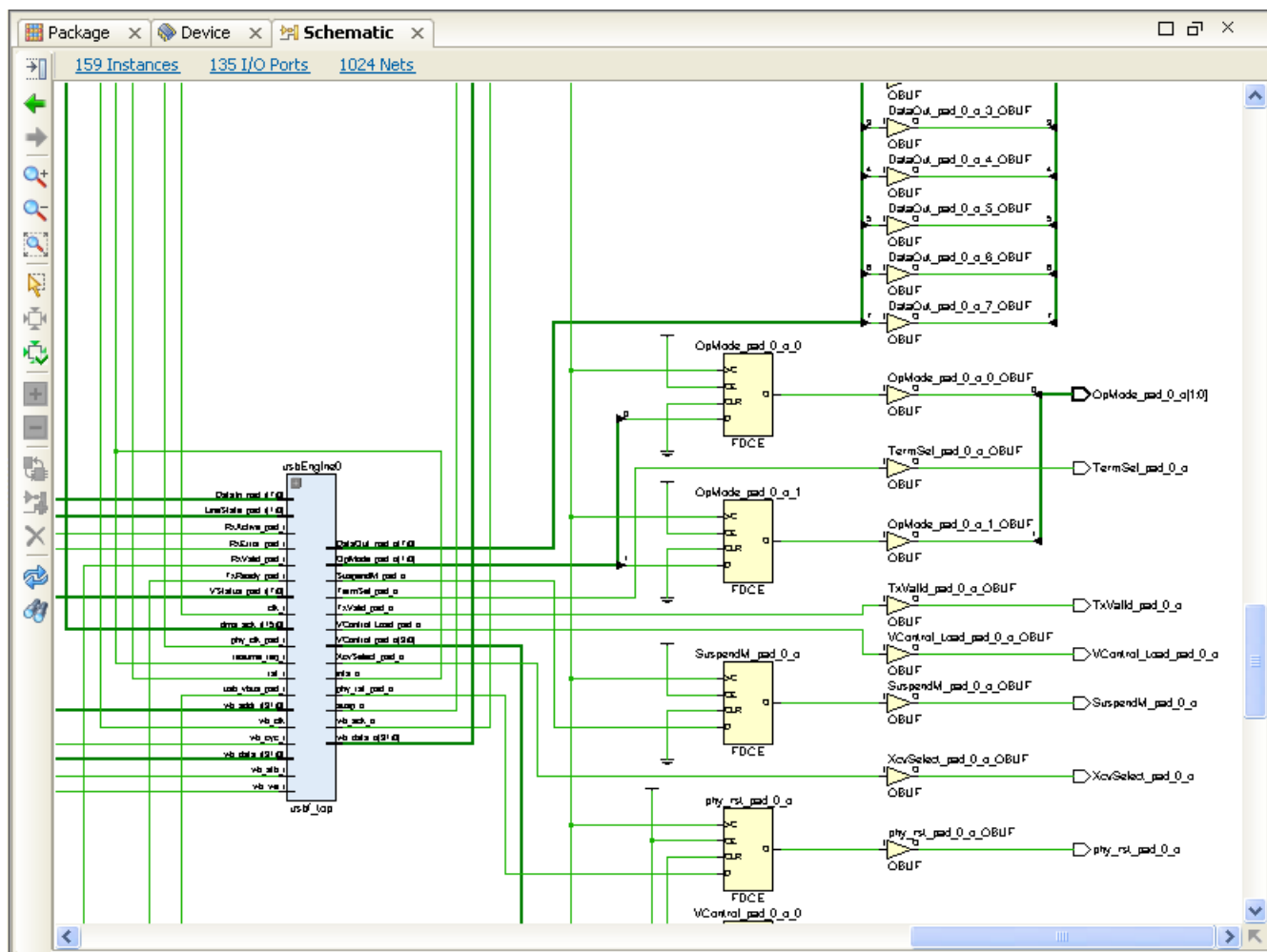


図 4-48 : [Schematic] ビュー

[Schematic] ビューのコマンド

[Schematic] ビューの左側にあるツールバー ボタンには、次のようなコマンドがあります。

- [Schematic View Options] [Schematic] ビューでの特定オブジェクトの表示を設定できます。詳細は、167 ページの「[Schematic] ビューでのオブジェクトの選択」を参照してください。



- [Previous Position] : 前の拡大レベルと座標に [Schematic] ビューの表示を戻します。
- [Next Position] : [Previous Position] を使用した後に [Schematic] ビューの表示を元の拡大レベルと座標に戻します。
- [Zoom In] : [Schematic] ビューを 2 倍ずつ拡大します。
- [Zoom Out] : [Schematic] ビューを 1/2 ずつ縮小します。
- [Zoom Fit] : デバイス全体を [Schematic] ビューに表示します。
- [Select Area] : 指定した四角形のエリア内のオブジェクトを選択します。
- [Fit Selection] : [Schematic] ビューで現在選択しているオブジェクトを表示します。これは、[Netlist] ビューなどの別のビューでオブジェクトを選択し、そのオブジェクトの表示を再描画する際に便利です。
- [Autofit Selection] : 新しく選択したオブジェクトを [Schematic] ビューで自動的に再描画します。このモードは、オンまたはオフにできます。
- [Expand all logic inside selected instance] : シンボル ビューからロジック ビューに階層モジュールを展開します。階層モジュールは、図 4-49 のような回路図シンボルの + アイコンをクリックしても直接展開できます。
- [Collapse all logic inside selected instance] : ロジック ビューから階層ビューに階層モジュールの展開を閉じます。展開した階層ブロックは、図 4-49 のような回路図シンボルの - アイコンをクリックしても直接展開を閉じることができます。

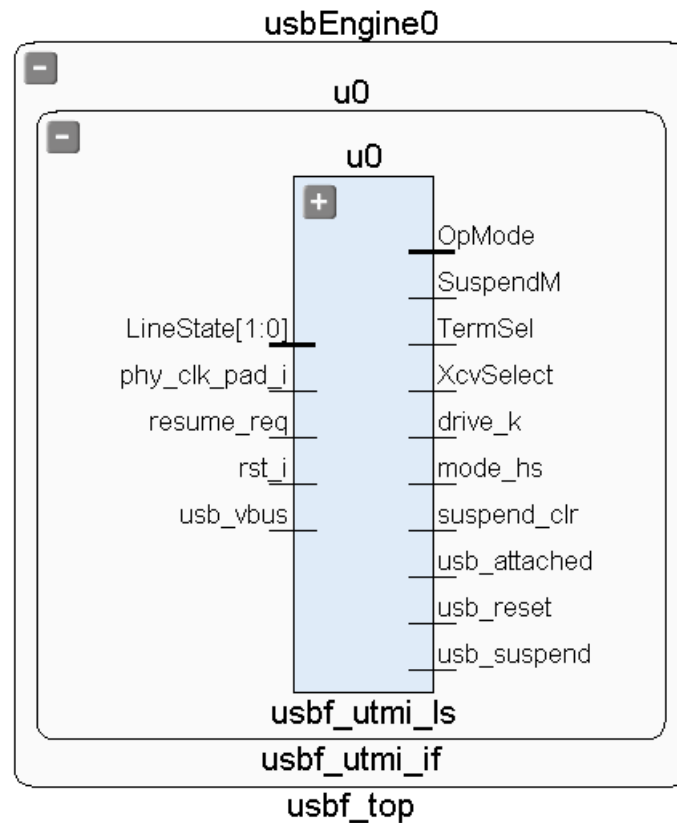


図 4-49 : [Schematic] ビューでの階層表示

- [Toggle autohide pins for selected instance] : 階層モジュールのピン表示を切り替えます。[Schematic] ビューでは、[図 4-49](#) に示すように、上位階層はピンなしで、同心の長方形として表示されます。ほとんどの場合、ピンがないことで [Schematic] ビューが見やすくはなりますが、選択したインスタンスのピンは必要に応じて表示できます。
- [Add selected elements to schematic] : 新しく選択したエレメントを既存の回路図に追加して [Schematic] ビューを作成し直します。
- [Remove selected elements to schematic] : 既存回路図から現在選択しているエレメントを削除して [Schematic] ビューを作成し直します。
- [Regenerate Schematic] : [Schematic] ビューを再描画します。
- [Find in Schematic] : [Find] ダイアログ ボックスを開きます。詳細は、[137 ページの「\[Find\] コマンド」](#)を参照してください。



選択したインスタンスおよびピンからのロジックの展開表示

回路図インスタンスまたはピンを選択すると、次の操作を実行できます。

- モジュール ピンおよびロジックを個別に展開または非展開します。
- 個々のピン、インスタンス、またはモジュール内外のロジック全体から、ロジックを選択して展開表示できます。

選択したモジュールに含まれるロジックまたはモジュールの次の上位階層に含まれるロジックを展開または非展開できます。1 つのモジュールまたは複数のモジュールを展開できます。回路図ロジックを展開するには、次を実行します。

- ポップアップ メニューから [Expand/Collapse] → [Expand Inside] をクリックし、選択したインスタンス内の回路図階層を表示します。選択したインスタンスの内容が展開され、[Schematic] ビューが更新されます。
 - 選択したインスタンスがデザイン階層内のプリミティブである場合は、このコマンドは使用できません。
 - ポップアップ メニューから [Expand/Collapse] → [Collapse Inside] をクリックし、選択した階層ブロックの回路図階層を非展開にします。
- ポップアップ メニューから [Expand/Collapse] → [Expand Outside] をクリックし、選択したインスタンスの上位階層を表示します。選択したインスタンスの上位階層が展開され、[Schematic] ビューが更新されます。
 - 選択したインスタンスがデザイン階層の最上位である場合は、このコマンドをクリックしても何も起こりません。
 - ポップアップ メニューから [Expand/Collapse] → [Collapse Outside] をクリックし、選択したインスタンスの上位階層を非展開にします。
- インスタンスのピンをダブルクリックすると、ネットが階層の下方向へ、もしくは上方向へトレースされます。ピンは、[図 4-50](#) に示すように、回路図シンボルの内側と外側に表示されます。シンボルの内側または外側のどちらをクリックするかで、展開方向が変わります。
 - 回路図シンボルの内側のピンをダブルクリックすると、ネットが下方向の階層にトレースされます。
 - 回路図シンボルの外側のピンをダブルクリックすると、ネットが上方向の階層にトレースされます。

注記：ネットの展開は、[Expand Inside]/[Expand Outside] コマンドを使用した階層モジュールの展開とは異なる結果になります。ピンをダブルクリックすると、ネットがトレースされますが、階層の全内容は表示されません。

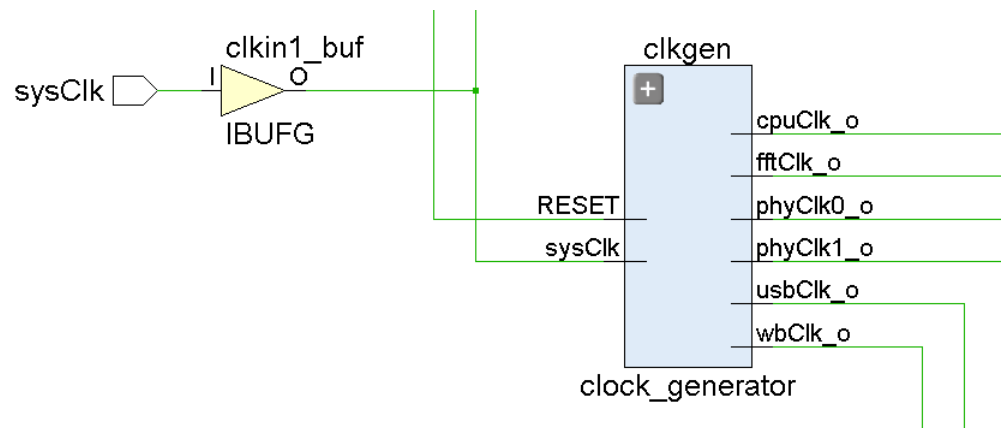


図 4-50：回路図ピンのダブルクリック

- バスを展開してすべてのビットを含めます。バスは、太いワイヤで表示されます。
- ポップアップメニューから [Expand Cone] をクリックし、選択したピンまたはインスタンス、あるいは選択した 2 つのインスタンス間のロジックのコーンを表示します。ロジックは、階層の境界を越えて展開されることがあります。[Expand Cone] コマンドには、次のサブコマンドがあります (図 4-51 を参照)。
 - [To Flops or I/Os]：最初のフリップフロップまたは I/O まで、またはブロック RAM、FIFO、エンベデッド プロセッサなどの順次エレメントまでのロジックすべてを表示します。
 - [To Primitives]：最初のプリミティブまでの出力ロジックすべてを表示します。この操作は、ピンをダブルクリックしたときのデフォルトです。
 - [To Primitives]：最初のプリミティブまでの出力ロジックすべてを表示します。

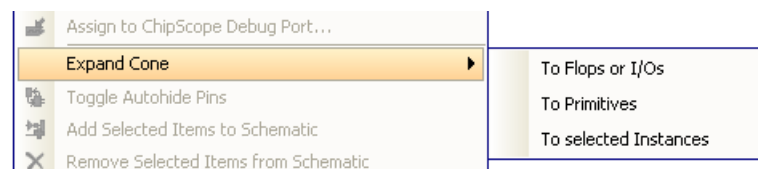


図 4-51：ロジック コーンの展開

[Schematic] ビューでのオブジェクトの選択

[Schematic] ビューでオブジェクトを選択するには、次のいずれかを実行します。

- [Schematic] ビューでオブジェクトをクリックします。
- 複数のオブジェクトを選択するには、Ctrl キーを使用します。
- ツールバーの [Select Area] ボタンをクリックし、複数のインスタンス、ポート、ネットを囲む長方形を描画します。

[Schematic] ビューでインスタンスを選択すると、その他すべてのビューでも選択されます。また、ほかのビューでオブジェクトを選択すると、[Schematic] ビューでも選択されます。

[Schematic] ビューでオブジェクトを選択すると、そのオブジェクトの [Properties] ビューが表示されます。

[Connectivity] タブには、ネットに接続されているすべてのプリミティブ インスタンスがリストされます。[Pins] タブには、ネットに接続されているインスタンスのピンがリストされ、プリミティブおよび階層インスタンスの両方が示されます。階層インスタンスに接続されているネットを選択すると、これらのタブの違いがわかります。

[Schematic View Options] の設定

168 ページの図 4-52 に示す [Schematic View Options] コマンドを使用すると、回路図シンボルおよびピンのどの属性を表示するか指定でき、[Schematic] ビューを作成する際に使用する色も設定できます。

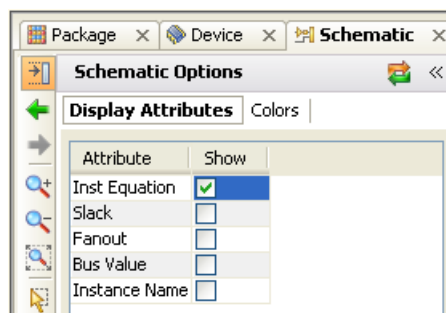


図 4-52 : [Schematic View Options] の設定

- [Display Attributes] : [Schematic] ビューで次の機能を表示するかどうか指定できます。
 - [Inst Equation] : インスタンスに真理値表の論理式を含めて表示します。
 - [Slack] : デスティネーション ピンにスラック値を含めて表示します。スラック値は、タイミング解析を実行しないと表示されません。
 - [Fanout] : ソース ピンにファンアウト値を含めて表示します。
 - [Bus Value] : バス ピンにバス値を含めて表示します。
 - [Instance Name] : インスタンスにインスタンス名を含めて表示します。
- [Colors] : [Schematic] ビューの要素の色は、次のように変更できます。

Item	Color
Background	255, 255, 255
Foreground	0, 0, 0
Selection	0, 0, 255
Markers	255, 255, 0
Cell text	0, 0, 0
Instance text	0, 0, 0
Pin text	0, 0, 0
Port text	0, 0, 0
Nets	25, 180, 0
Buses	0, 128, 0
Ports	0, 0, 0

 - [Color] 列をクリックし、ドロップダウン リストに表示される色から選択します。
 - [More Colors] をクリックすると、より多数の色から選択できます。
 - [Color] 列のセルに直接 RGB 値を入力します。
- [Reset] : [Device View Options] の [Layers]、[Colors]、または [Connectivity Display] を PlanAhead ツールのデフォルトの設定にリセットします。



[Schematic] ビューでのタイミング パス ロジックの表示

[Timing Results] ビューでタイミング パスを選択し、ポップアップ メニューで [Schematic] をクリックすると、[Schematic] ビューにそのタイミング パスが表示されます。選択したパスまたはパスグループのオブジェクトはすべて、169 ページの図 4-53 に示すように、ロジック階層の境界およびインターコネクト ワイヤと共に表示されます。

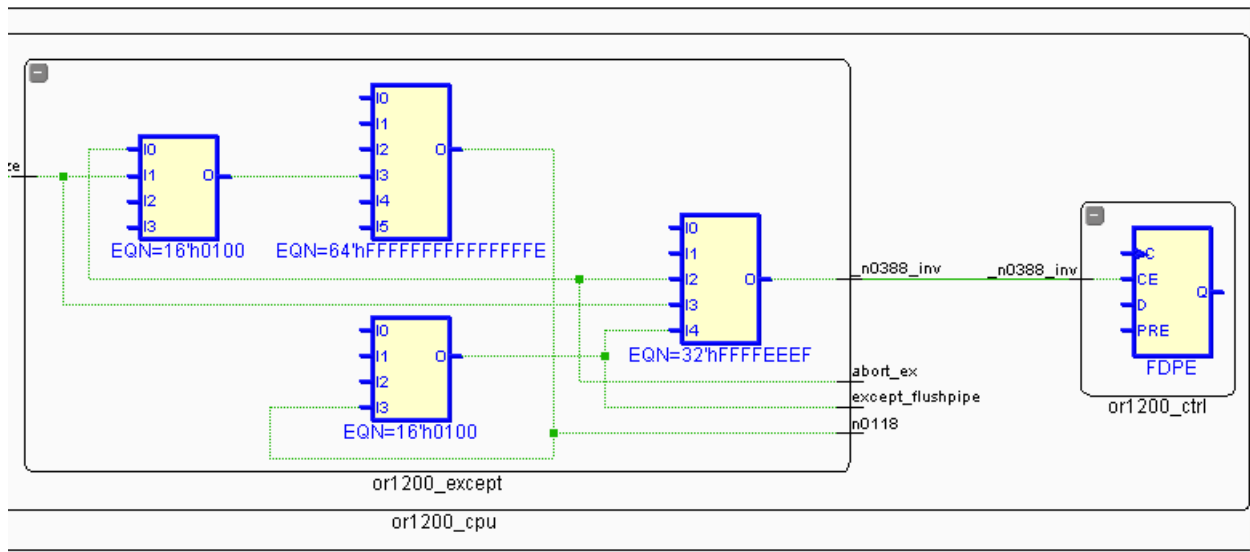


図 4-53 : [Schematic] ビューでのタイミング パス

タイミング パス ロジックの設定については、第 7 章「合成済みデザインの制約および解析」および第 11 章「インプリメンテーション結果の解析」を参照してください。

注記：TRCE (Timing Reporter and Circuit Evaluation) で生成された TWX (XML ファイル) または TWR (テキスト ファイル) フォーマットのタイミング レポートに含まれるパスで、インターコネクトワイヤが表示されないことがあります。これは、ISE インプリメンテーションで最適化によりパスが削除されたからです。[Schematic] ビューには、選択されたパスに含まれている実際のオブジェクトがすべて表示されますが、最適化でオブジェクトが削除された場合、PlanAhead では接続を補間できません。[Schematic] ビューと [Path Properties] ビューを使用すると、パスの接続を簡単にトレースできます。通常は、回路図はパスの方向が簡単にわかるように描画されています。詳細は、258 ページの「タイミング結果の解析」を参照してください。

[Properties] ビュー

[Properties] ビューには、選択したロジック オブジェクトまたはデバイス リソースに関する情報が表示されます。デフォルトでは、オブジェクトを選択すると、そのプロパティが自動的に [Properties] ビューに表示されます。

[Properties] ビューで名前を変更すると、選択したオブジェクトにもそれが反映されます。たとえば、BEL を選択すると、このビューは [BEL Properties] ビューになり、クロック領域を選択すると [Clock Region Properties] ビューになります。

注記：複数オブジェクトを選択した場合、[Properties] ビューには最後に選択したオブジェクトのプロパティが表示されます。

[Properties] ビューを開くには、次の手順に従います。

- メイン メニューから [Windows] → [Properties] をクリックします。
- [Properties] ツールバー アイコンをクリックします。
- オブジェクトを選択し、ポップアップ メニューから [Object Properties] をクリックします。



[Properties] ビューには、選択したオブジェクトに関するさまざまなプロパティが表示され、それぞれのタブにカテゴリ別の情報が分類されます。表示されるタブおよび情報は、選択しているオブジェ

クトの種類によって異なります。170 ページの図 4-54 は [Instance Properties] ビューのさまざまなタブを表示しています。図で表示されているのは、選択したインスタンスの [Attributes] タブです。

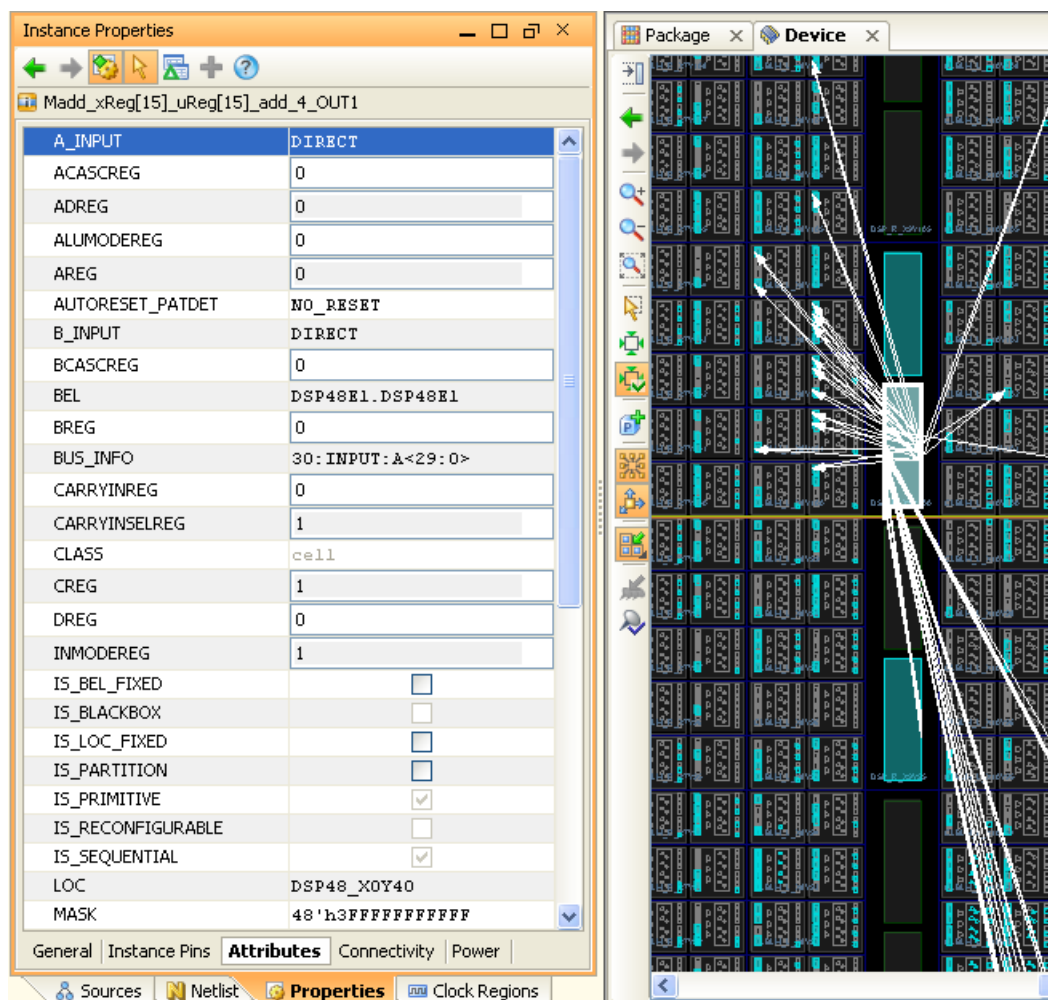


図 4-54 : [Properties] ビュー

[Properties] ビューのコマンド

[Properties] ビューのツールバーには、選択したオブジェクトや表示されているタブによってさまざまなコマンドが含まれます。共通のコマンドには、次のようなものがあります。

[Previous object] : 前に選択したオブジェクトのプロパティを表示します。選択したオブジェクトを前へ戻していく場合に使用できます。



[Next object] : 次に選択したいオブジェクトのプロパティを表示します。このツールバーボタンは、[Previous object] ボタンを使用した後にのみ使用可能です。



[Automatically update the view when new objects are selected] : デフォルトでは、新しいオブジェクトを選択すると、その最後に選択したオブジェクトのプロパティが自動的にアップデートされます。このコマンドを使用すると、新しいオブジェクトを選択するたびに [Properties] ビューを自動的に更新するか、現在選択されているオブジェクトのプロパティを表示したままにするか指定できます。



[Select/Unselect object] : [Properties] ビューに表示されているオブジェクトの選択または選択解除します。



[Expand All] : [Sources] ビューの階層ツリーをすべて展開表示し、すべてのエレメントを表示します。



[Collapse All] : [Sources] ビューの階層ツリーをすべて閉じ、最上位オブジェクトのみを表示します。



[Group by type] : 選択したアイテムをタイプ別にグループ化します。



[Delete] : [Properties] ビューのタブから属性またはオブジェクトを削除します。一部のオブジェクト タイプおよびビュー ペインでのみ使用できます。



[Export statistics to file] : 統計を後で解析できるようにファイルに保存します。Pblock、クロック領域、およびインスタンスの [Properties] ビューの [Statistics] タブにのみ使用されます。



[Show unsaved attributes only] : [Properties] ビューの [Attributes] タブに保存されていない属性のみを表示します。これらは、アップデートされた属性で、[Apply] コマンドをクリックすると保存されます。



[Add pre-defined attributes] : 選択したオブジェクトに新しい属性を追加します。一部のオブジェクト タイプおよびビュー ペインでのみ使用できます。



[Run Properties] ビューの使用

[Properties] ビューの 1 つである [Run Properties] ビューには、選択した合成またはインプリメンテーション run に関する情報が表示されます。このビューの名前は合成かインプリメンテーションによって、[Synthesis Run Properties] ビューまたは [Implementation Run Properties] ビューになります。図 4-55 は、選択した run の [Implementation Run Properties] ビューを示しています。

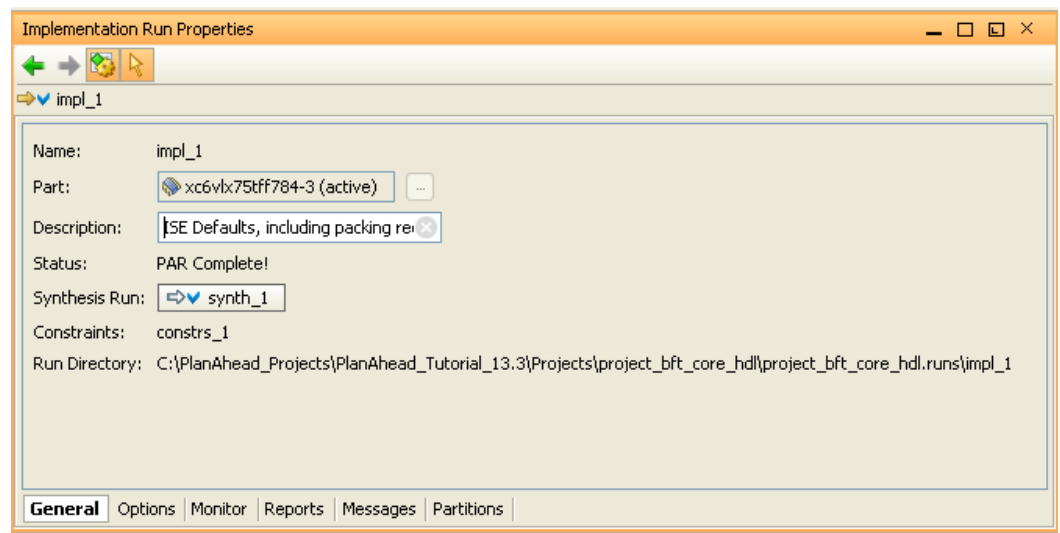


図 4-55 : [Implementation Run Properties] ビュー

[Design Runs] ビューで run を選択すると、[Run Properties] ビューのタブの 1 つにその run のプロパティが表示されます。[General] タブには、ターゲット パーツ、制約セット、親階層の run、run ストラテジ、run の出力ファイルのある run ディレクトリなどの run の設定がレポートされます。172 ページの表 4-1 は、[Run Properties] ビューの各タブの詳細を示しています。

[Run Properties] ビューでレポートされるプロパティは、そのビューで直接変更できますが、run が合成またはインプリメントされた後に run のプロパティが変更されている場合は、run が最新の状態ではなくなります。このような場合は、run をリセットして再起動し、結果をアップデートする必要があります。run をリセットするには、[Design Runs] ビューのポップアップ メニューから [Reset Runs] コマンドを使用します。詳細は、[第 9 章の「リモート Linux ホストでの run の起動」](#)を参照してください。

表 4-1：[Run Properties] ビューのタブ

タブ	オプション
[General]	<ul style="list-style-type: none"> • [Name]：run の名前を定義します。 • [Part]：現在の run に対するターゲット パーツが表示され、run のプロジェクトのパーツを変更できます。ターゲット パーツは [Project Settings] で定義されますが、[Run Properties] からでも変更できます。プロジェクト全体のターゲット パーツ設定については、第 3 章の「プロジェクト設定」を参照してください。 • [Description]：現在の run ストラテジの簡単な説明を表示します。 • [Status]：run のステータスを表示します。 • [Synthesis Run]：選択したインプリメンテーション run の親レベルの合成 run を表示します。 <p>注記：これは、インプリメンテーション run のプロパティなので、合成 run には表示されません。</p> <ul style="list-style-type: none"> • [Constraints]：run の制約セットをそのまま使用するか、変更します。 • [Run Directory]：run データのディレクトリを表示します。
[Options]	<p>コマンド ライン オプションと現在のセット値が表示されます。コマンド オプションを選択すると、そのコマンドの詳細が表示されます。</p> <ul style="list-style-type: none"> • まだ run を起動していない場合は、コマンド オプションの値を変更できます。変更するコマンド オプションを選択し、そのオプションのチェック ボックスをオン/オフにするか、値を入力するかプルダウン メニューから選択し、[Apply] をクリックします。 • 値を変更した場合は、オプションの横にアスタリスク (*) が表示され、デフォルトのストラテジの値が変更されたことを示します。 • 右クリックのポップアップ メニューからは、次を実行できます。 <ul style="list-style-type: none"> • [Save Strategy As] をクリックすると、新しいオプション設定をストラテジとして保存して、その他の run であとで使用できるようになります。 • コマンド オプションをプリセット値に戻します。 • 第 9 章の「インプリメンテーション run の起動」に示すように run を起動した後、run ストラテジを変更した場合、その run は最新の状態ではなくなります。run はキャンセルまたはリセットする必要があります。詳細は、第 9 章の「run のキャンセル」を参照してください。

表 4-1 : [Run Properties] ビューのタブ

タブ	オプション
[Monitor]	<p>[Compilation] ビューに表示されるのと同じ STDOUT コマンド ステータス ログが表示されます。119 ページの図 4-10 に、[Compilation] ビューの例を示します。</p> <p>[Monitor] タブは、コマンドが実行されるごとにアップデートされます。スクロール バーを使用すると、コマンド ログのレポートを参照できます。アクティブなレポートを停止するには、[Automatically update the contents of this view] をクリックします。これにより、コマンドの実行中に結果がスクロールして読みやすくなります。</p>
[Reports]	<p>ISE で生成されたレポート ファイルは PlanAhead で表示できます。[Implementation Run Properties] ビューで run を選択し、[Reports] タブをクリックすると、使用可能なレポート ファイルのリストがワークスペースに表示されます。</p>
[Messages]	<p>run メッセージを表示します。[Messages] ビューの例は、117 ページの図 4-7 を参照してください。</p>

[Netlist] ビュー

[Netlist] ビューには、ネット、ロジックプリミティブ、デザインの階層モジュールを含むエラポートまたは合成済みロジック デザインが、現在定義されている最上位モジュールから順に階層表示されます。

図 4-56 に、[Netlist] ビューを示します。

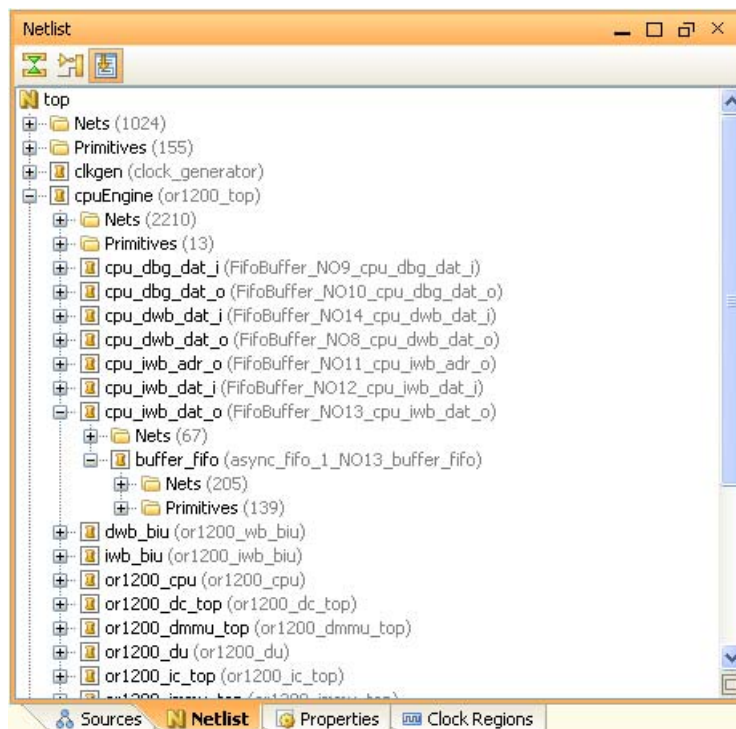


図 4-56 : [Netlist] ビュー

[Netlist] ビューには、デザインに含まれるロジック インスタンスおよびネットが表示されます。ネットリストは、ロジック ツリーの階層を展開または非展開することでナビゲートできます。デフォルトでは、ほかのビューでネットリスト オブジェクトを選択すると、そのオブジェクトを表示するために、ネットリスト ツリーはダイナミックに展開およびスクロールされます。この設定を無効にするには、このウィンドウの [Automatically scroll to selected objects] ボタンをクリックします。



[Netlist] ビューで [Collapse All] ボタン をクリックすると、ネットリスト ツリーを閉じることができます。詳細は、[130 ページの「ビュー特定のツールバー コマンド」](#)を参照してください。ネットリスト ツリーを閉じると、最上位ロジック モジュールのみが表示されます。



インスタンスを選択して、メニュー、ツールバー、または右クリックで表示されるポップアップ メニューからコマンドを適用できます。

Shift キーまたは Ctrl キーを使用すると、[Netlist] ビューで複数のエレメントを選択でき、複数のエレメントに対してコマンドを実行できます。選択されたロジックは、[Netlist] ビューでハイライトされます。

ロジックは、[Schematic] または [Device] ビューなどのビュー間で連動して選択されます。選択したロジックすべてが表示されるよう、ネットリスト ツリーが自動的に展開されます。選択したロジックをすべて表示するため、ツリーをスクロールする必要がある場合があります。ネットリスト ツリーを非展開にしても、ロジックの選択は解除されません。

プリミティブ ロジックは、階層の各レベルの /Primitives フォルダーに配置されます。これにより、[Netlist] ビューのロジックや階層モジュールの表示が簡略化されたものになります ([174 ページの図 4-57](#) を参照)。

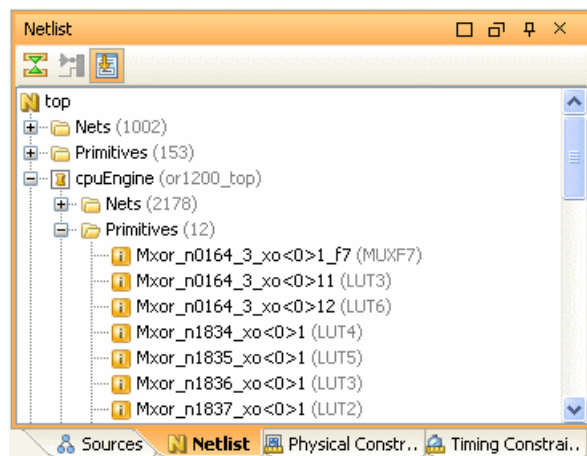


図 4-57 : [Netlist] ビューの [Primitives] フォルダー

/Primitives フォルダーを Pblock に直接割り当てると、1 つのモジュールのすべてのプリミティブ ロジックを Pblock に割り当てることができます。

注記： ネットリストを更新する際、ロジック名が再合成で変更されている可能性があるため、Pblock に [Primitive] フォルダーを割り当て直す必要がある場合があります。

ネットまたはワイヤは、階層の各レベルの /Nets フォルダーに配置されます。バスのビットすべては、デフォルトでバスの下に非展開表示でまとめられますが、[図 4-58](#) に示すように各ビットを展開することもできます。

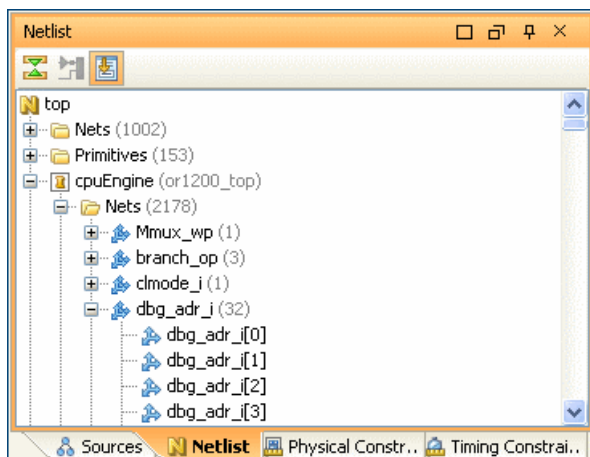


図 4-58 : [Netlist] ビューの [Nets] フォルダ

ネットを選択すると、[Device] ビューでそのネットがハイライトされ、バスを選択すると、そのバスに含まれるすべてのネットがハイライトされます。ネットは、[Schematic] ビューで表示できます。

ポップアップメニューの [Add to ChipScope Unassigned nets] コマンドを使用して、ChipScope ツールでのデバッグテスト用にネットを選択できます。詳細は、[第 12 章の「デバッグ コアへのネットの接続および接続解除」](#)を参照してください。

[Netlist] ビューのアイコン

図 4-59 に示すように、ネットリスト ロジックのステータスを示すため、さまざまなアイコンが使用されます。詳細は、次のセクションを参照してください。













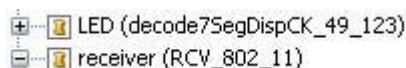
-  - Hierarchical instance (logic)
-  - Hierarchical instance (black box)
-  - Hierarchical instance (black box + Pblock)
-  - Hierarchical instance (Pblock)
-  - Partitioned instance (logic)
-  - Partitioned instance (black box)
-  - Partitioned instance (Pblock)
-  - Partitioned instance (black box + Pblock)
-  - Bus
-  - Net
-  - Primitive instance
-  - Primitive instance (fixed with LOC)

図 4-59 : [Netlist] ビューのアイコン

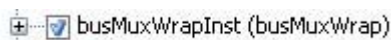
階層インスタンス

階層ネットリスト モジュールまたはインスタンスは、次のように黄色の文字 I を含むアイコンで表示されます。



Pblock に割り当てられた階層インスタンス

Pblock に割り当てられた階層ネットリスト モジュールまたはインスタンスは、次の図に示すように、青色のチェック マークを含むアイコンで表示されます。



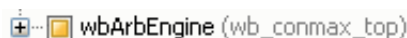
ブラック ボックス モジュール

ネットリストまたはロジックの中身がない階層インスタンスは、PlanAhead ではブラック ボックスとして表示されます。これらは、次の図に示すように暗い背景の階層インスタンス アイコンで表示されます。階層インスタンスはデザインで意図的にブラック ボックスとなっているか、検索パスに問題があったか、ファイルが存在しない場合などにブラック ボックスとして表示されます。



パーティション モジュール

[Set Partition] ポップアップ メニュー コマンドを使用してパーティションとして設定されたモジュールは、白いボックス内に黄色のボックスで表示されます。パーティションの使用の詳細は、第 13 章「階層デザイン手法の使用」を参照してください。



パーシャル リコンフィギュレーション パーティション モジュール

[Set Partition] ポップアップ メニュー コマンドを使用してパーティションとして設定されており、パーシャル リコンフィギュレーション プロジェクトに含まれるモジュールは、白いボックス内に黄色のボックスで表示されます。

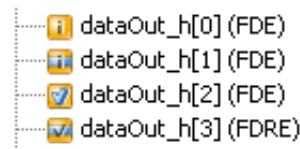


プリミティブ ロジックのインスタンス

プリミティブ ロジックのインスタンスは、次のように表示されます。

- 配置制約のないインスタンスは、黄色の長方形の中に英文字 **i** が表示されたアイコンで示されます。
- 配置制約が設定されたインスタンスは、青い横線の付いた黄色の長方形のアイコンで示されます。
- Pblock に割り当てられているインスタンスは、黄色の長方形に青色のチェック マークが付いたアイコンで示されます。
- 配置制約が設定され、Pblock に割り当てられたインスタンスは、黄色の長方形にチェック マークと青色の横線が付いたアイコンで示されます。

また、ロジックのタイプも示されます。



[Hierarchy] ビュー

[Hierarchy] ビューでは、現在の最上位モジュールに基づいて、[Netlist] ビューを反映させてロジック階層を視覚化でき、階層モジュール間の関係や、相対的なサイズなどを確認できます。

[Hierarchy] ビューは、[Netlist] ビューなどの開いているビューのポップ アップ メニューから [Show Hierarchy] をクリックすると、表示できます (178 ページの図 4-60)。

[Hierarchy] ビューには、現在のデザインのロジック階層がグラフィカルに表示されます。デザインを上位から下位に表示することで、デザイン内のモジュール サイズと位置を識別できます。

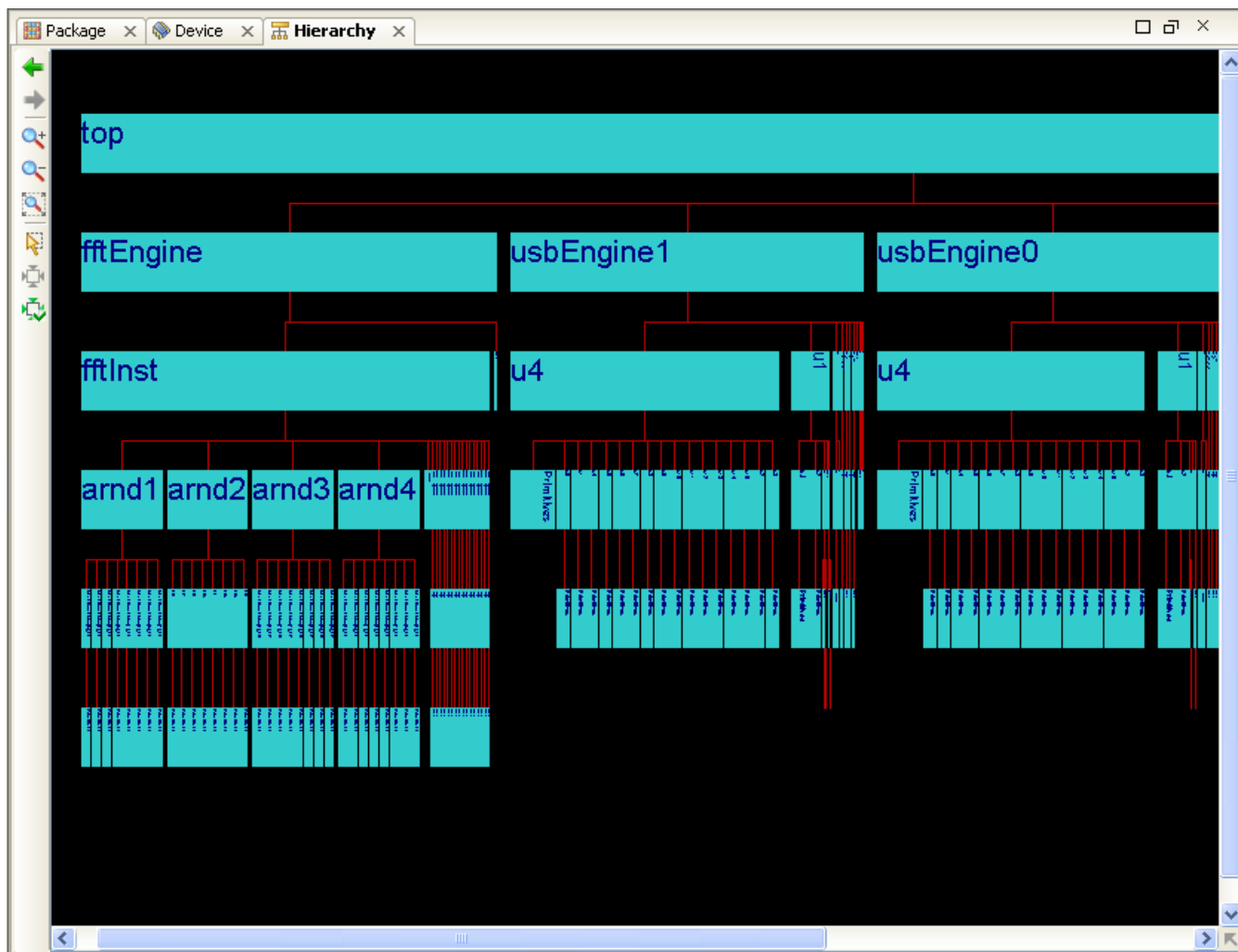


図 4-60 : [Hierarchy] ビュー

このビューは、主にデザイン解析およびフロアプランで使用します。タイミングパスがロジック階層をどのように移動するのか、またはフロアプランする前にモジュールの大きさを確認できます。[Hierarchy] ビューのブロックの幅は、ブロックで消費される FPGA リソースに基づいています。

[Hierarchy] ビューには、各階層インスタンスが表示されます。プリミティブ ロジックはフォルダー内にグループ化され、サブモジュールとして表示されます。プリミティブ ロジック フォルダーの詳細は、173 ページの「[Netlist] ビュー」を参照してください。

Pblock を割り当てるためロジックの親モジュールを選択するには、[Select Primitive Parents] コマンドをクリックします。

[I/O Ports] ビュー

[I/O Ports] ビューは、[Package] ビューまたは [Device] ビューで IP ポートを作成、設定、または I/O サイトに配置する際に使用します。[I/O Ports] ビューには、デザインで定義されている I/O 信号のポートが表示されます。[I/O Ports] ビューを表示するには、[Window] → [I/O Ports] をクリックします。図 4-61 は、[I/O Ports] ビューを示しています。

I/O 配置プロジェクトの場合、CSV ファイルや UCF からポート リストをインポートして、プロジェクトのポートを手動で作成します。詳細は、第 8 章「I/O ピン配置」を参照してください。

RTL ソース、またはネットリスト プロジェクト作成すると、[I/O Ports] ビューにデザイン ソース ファイルで定義されている I/O ポートが自動的に表示されます。

Name	Direction	Neg Diff Pair	Site	Fixed	Bank	I/O Std	Vcco	Drive ...	Slew Type	Pull Type	Off-
All ports (135)											
DataIn_pad_0_i (8)	Input					LVC MOS18	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[0]	Input		G24	✓	36	LVC MOS18	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[1]	Input		F24	✓	37	LVC MOS18	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[2]	Input		G22	✓	36	LVC MOS33	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[3]	Input			✓		LVDCI_15	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[4]	Input		F22	✓	37	LVDCI_18	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[5]	Input		E23	✓	37	LVDCI_DV2_15	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[6]	Input		D23	✓	37	LVDCI_DV2_18	1.8	12 SLOW	NONE	NONI	
DataIn_pad_0_i[7]	Input			✓		LV TTL	1.8	12 SLOW	NONE	NONI	
DataIn_pad_1_i (8)	Input					MOBILE_DDR	1.8	12 SLOW	NONE	NONI	
DataOut_pad_0_o (8)	Output					PCI33_3	1.8	12 SLOW	NONE	FP_V	
DataOut_pad_1_o (8)	Output					LVC MOS18	1.8	12 SLOW	NONE	FP_V	
LineState_pad_0_i (2)	Input					LVC MOS18	1.8	12 SLOW	NONE	NONI	
LineState_pad_1_i (2)	Input					LVC MOS18	1.8	12 SLOW	NONE	NONI	
OpMode_pad_0_o (2)	Output					LVC MOS18	1.8	12 SLOW	NONE	FP_V	
OpMode_pad_1_o (2)	Output					LVC MOS18	1.8	12 SLOW	NONE	FP_V	
or1200_pm_out (4)	Output					LVC MOS18	1.8	12 SLOW	NONE	FP_V	

図 4-61 : [I/O Ports] ビュー

[I/O Ports] ビューでは、次が実行できます。

- 各 I/O ポートに対してポートの信号名、方向、パッケージ ピン、I/O バンク、I/O 規格、駆動電流、差動ペア、スルー タイプ、電圧要件、およびその他の信号情報を表示します。
- 列の値に基づいて I/O ポートを分類します。表の表示方法については、127 ページの「ツリー表形式のビュー」を参照してください。
- 値を次のように表示します。
 - デフォルト値の場合は空白になります。
 - デフォルト以外の値にはアスタリスク (*) が付きます。
 - 不正な値または未定義の値は赤色で表示されます。
- [I/O Ports] ビューのセルに編集可能な値が表示されます。テキストを入力するか、ドロップダウン メニューからテキストを選択して値を変更します。
- バスは展開可能なフォルダーにまとめられており、解析、設定、割り当てで 1 つのオブジェクトとして選択できます。

[I/O Ports] ビューのコマンド

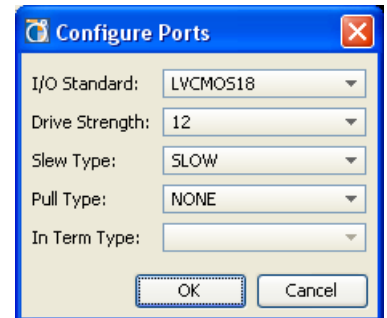
[I/O Ports] ビューでは、次のコマンドを実行できます。

- [Create I/O Port] : I/O 配置プロジェクトの I/O ポートを手動で定義します。
- [Search] [I/O Ports] ビューでポートを名前、キーワード、値で検索できます。
- [Collapse All] : バスを名前別に表示します。ビットは表示しません。
- [Expand All] : バスのピンをすべて展開して表示します。
- [Group by Interface and Bus] : ポートをインターフェイス別に、または名前のアルファベット順に表示します。
- [Create I/O Port Interface] : ポートをグループにまとめるために、新しいポート インターフェイスを定義します。ポート インターフェイスは、I/O 配置環境で 1 つのオブジェクトとして選択し、配置できます。
- [Schematic] : 選択した I/O ポートの [Schematic] ビューを開きます。
- [Automatically scroll to selected objects] : [I/O Ports] ビューをスクロールし、[Netlist] または [Device] ビューなどの別のビューで選択したオブジェクトを表示します。



[I/O Ports] ビューでポートおよびインターフェイスを選択し、I/O Planning ビュー レイアウトを使用してパッケージ ピンまたはデバイス リソースに割り当てることができます。[I/O Ports] ビューのポップアップ メニューを使用すると、次が実行できます。

- [Place I/O Ports in an I/O Bank] : 現在選択されているポートを指定した I/O バンクのピンに割り当てます。
- [Place I/O Ports in Area] : 現在選択されているポートを指定したエリアのピンに割り当てます。
- [Place I/O Ports Sequentially] : 現在選択されているポートをピンにそれぞれ割り当てます。
- [Configure I/O Ports] : 選択した I/O ポートのプロパティを指定できます。
- [Reset Invalid Port Properties] : 指定したポートの無効なプロパティをデフォルト値にリセットします。
- [Reset Port Properties] : 指定したポートのプロパティをすべてデフォルト値にリセットします。
- [Fix/Unfix] : 選択した配置済み I/O ポートを固定するか、固定を解除します。詳細は、[365 ページの「固定された配置制約および固定されていない配置制約について」](#)を参照してください。
- [Unplace] : 選択した I/O ポートの配置を解除します。
- [Export I/O Ports] : [I/O Ports] ビューの内容を CSV、UCF、Verilog、または VHDL ファイルに書き出します。
- [Export to Spreadsheet] : [I/O Ports] ビューの内容を Microsoft® Excel スプレッドシートに書き出します。



[Package Pins] ビュー

[Package Pins] ビューには、I/O に関連したパッケージ情報が表示されます。I/O ピンおよび I/O ポート情報を解析するため、並び替えたりフィルターを適用したりできます。

[Package Pins] ビューを開く

[Package Pins] ビューを表示するには、[Window] → [Package Pins] をクリックします。図 4-62 に、[Package Pins] ビューを示します。

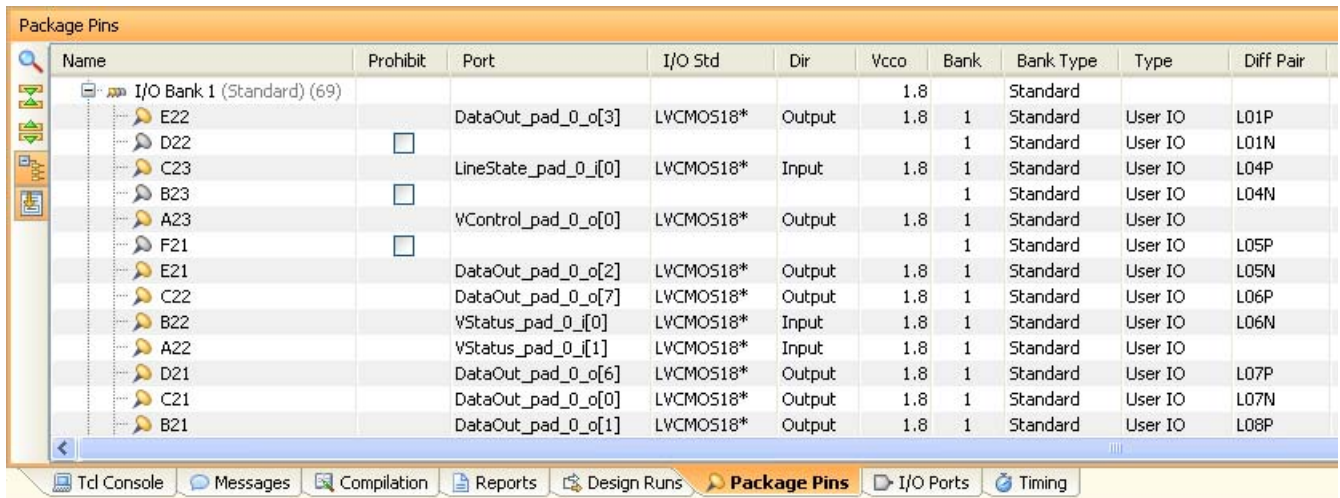


図 4-62 : [Package Pins] ビュー

各パッケージ ピンについて、I/O バンク番号、バンク タイプ、差動ペア、サイト タイプ、最小/最大パッケージ遅延などのデバイス ピン情報がリストされます。[Bank Type] 列には、Virtex-7、Kintex™-7 および Artix™-7 デバイスの HP/HR バンクが表示されます。

注記：最小/最大パッケージ遅延は、ピコ秒 (ps) で示されます。

表の値は、次のように表示されます。

- デフォルト値はグレーで表示されます。
- デフォルト以外の値は黒色で表示されます。
- 不正な値は赤色で表示されます。

[Package Pins] ビューの情報は、列のヘッダーをクリックすると並べ替えることができます。列ヘッダーをもう 1 度クリックすると、並べ替え順が逆になります。ある列を基準に並べ替えた後で Ctrl キーを押しながら別の列ヘッダーをクリックすると、次にその列を基準に並べ替えることができます。リストの表示順を整えるために、さまざまな並び替え条件を選択できます。[Package Pins] ビューの情報の並び替え方法については、127 ページの「ツリー表形式のビュー」を参照してください。

[Package Pins] ビューの編集可能な値を含むセルは、値をテキストで入力するか、ドロップダウン リストから選択して変更できます。

[Package Pins] ビューのコマンド

- [Search] : [Package Pins] ビューでピンを名前、キーワード、値で検索できます。
- [Collapse All] : I/O バンクを名前別に表示します。バンクのピンは表示しません。
- [Expand All] : I/O バンクのピンをすべて展開して表示します。
- [Group by I/O Bank] : I/O バンク別にピンをグループ分けするか、名前をアルファベット順にリストします。



- [Automatically scroll to selected objects] : [Package Pins] ビューをスクロールし、[Netlist] または [Device] ビューなどの別のビューで選択したオブジェクトを表示します。



[Design Runs] ビューの使用

[Design Runs] ビューは、合成およびインプリメンテーション run を表示、設定、起動、および解析するために使用します。[Design Runs] ビューでは、合成およびインプリメンテーション用の run を管理、起動、リセットできるコマンドが実行できます。[Design Runs] ビューを開くには、[Windows] → [Design Runs] をクリックします。図 4-63 に、[Design Runs] ビューを示します。

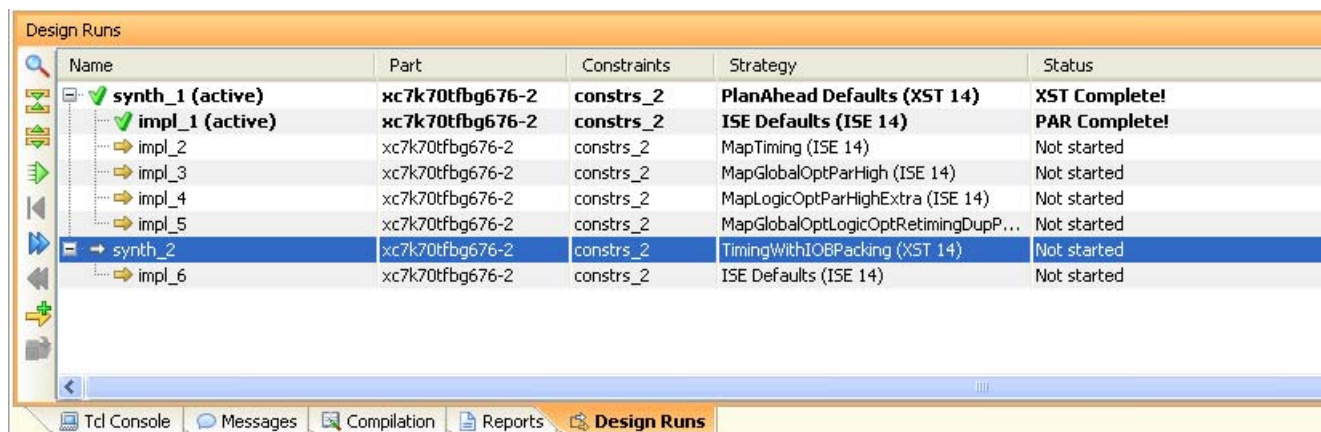


図 4-63 : [Design Runs] ビュー

run を作成、起動、またはインポートすると、そのステータスが [Design Runs] ビューに表示されます。このビューでは、定義された run のステータスや結果が表示され、run の変更、インポート、起動、管理のためのコマンドを実行できます。

run は次のように表示されます。

- 現在実行中の run には、青色の渦巻きアイコンが回った状態で表示されます。
- 完了した run には、青色のチェックマークが表示されます。

コマンドを実行すると、run に関する情報が表示されます。実行中の run に影響を与えずに、PlanAhead を閉じることができます。プロジェクトを次に開いたときに、run のステータスがアップデートされ、[Design Runs] ビューに表示されます。

[Design Runs] ビューの表には、次の列があります。

- [Name] : run の名前を表示します。
- [Part] : run に選択されているデバイスを表示します。
- [Constraints] : run に使用された制約セットを表示します。
- [Strategy] : run に使用されたストラテジを表示します。アスタリスク (*) の付いたストラテジは、そのストラテジのコマンド オプションの一部が上書きされたことを示します。詳細は、226 ページの「合成 run の定義」および329 ページの「インプリメンテーション オプションの設定」を参照してください。
- [Status] : run が開始されていないか、実行中か、終了したかを示します。

- [Progress] : run 全体の終了を示します。これは、ISE インプリメンテーション ツール シーケンス (ngdbuild から XDL まで) 全体を含みます。
- [Start] : run の開始時をレポートします。
- [Elapsed] : run の経過時間をレポートします。
- [Device Utilization] : run の LUT の使用率を示します。合成 run に対してのみ表示されます。
- [Fmax (MHz)] : XST 合成レポートからの予測されるクロック周波数を示します。合成 run に対してのみ表示されます。
- [Timing Score] : 進行中または完了した run の現在のタイミング スコアを示します。インプリメンテーション run に対してのみ表示されます。
- [Unrouted] : 進行中または完了した run の未配線ネットの数を示します。インプリメンテーション run に対してのみ表示されます。
- [Description] : run の説明を表示します。ストラテジを run に適用したときに記述されたものですが、後で修正できます。

この表は、実行が進行すると随時更新されます。PlanAhead で生成されたスクリプトを使用し、PlanAhead ソフトウェア環境外で実行された run は、PlanAhead を起動したときに更新されます。

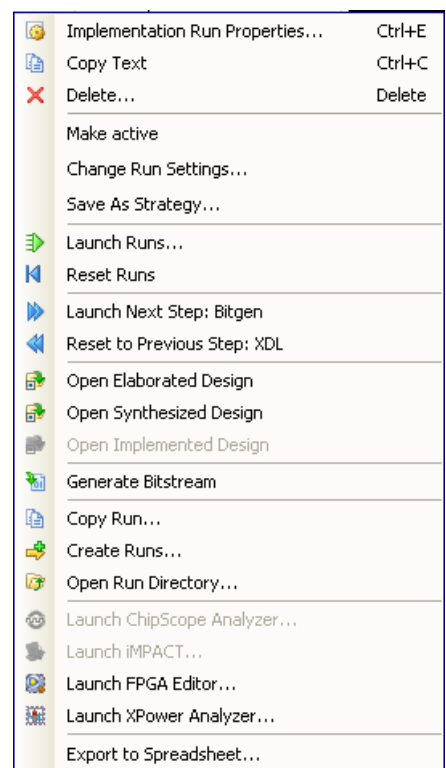
[Design Runs] ビューのポップアップ メニュー コマンド

[Design Runs] ビューを右クリックして表示されるポップアップ メニューには、次のコマンドがあります。

- [Run Properties] : [Run Properties] ビューを表示します。詳細は、[171 ページの「\[Run Properties\] ビューの使用」](#)を参照してください。
- [Copy Text] : 選択したテキストをコピーします。
- [Delete] : 選択した run と関連する run データを削除します。run を削除する前に確認メッセージが表示されます。

注記 : アクティブな run は削除できません。

- [Make active] : 選択した run をアクティブにします。[Run Synthesis] または [Run Implementation] コマンドを使用すると、アクティブな run が自動的に開始されます。アクティブ run の結果は、[Messages]、[Compilation]、[Reports]、および [Project Summary] ビューに表示されます。
- [Change Run Settings] : 選択した合成またはインプリメンテーション run のストラテジおよびコマンドライン オプションを変更します。詳細は、[226 ページの「合成 run の定義」](#)および[325 ページの「インプリメンテーション run の定義」](#)を参照してください。
- [Save as Strategy] : 現在のストラテジを新しいストラテジ ファイルに保存して、今後使用したり変更したりできるようにします。
- [Launch Runs] : [Launch Selected Runs] ダイアログボックスを開き、選択した run を実行します。



- [Reset Runs] : [Reset Runs] ダイアログボックスが開き、前回の run の結果を削除し、実行ステータスを [Not Started] に戻します。
- [Launch Next Step: <Step>] : 選択した run の次のステップを開始します。インプリメンテーション run で使用できるステップは、NGDBuild、MAP、PAR、TRCE、XDL、BitGen のいずれかです。合成 run の場合は、XST だけを指定できます。
- [Reset to Previous Step: <Step>] : 選択した run を前のインクリメンタル ステップにリセットします。これにより、run を前の段階に戻して、必要に応じて変更を加えてから、run を次の段階へ進めて終了させることができます。
- [Open Elaborated Design] : 選択した run の制約セットおよびターゲット パーツを使用して RTL ソース ファイルのエラボレート済みデザインを作成します。
- [Open Synthesized Design] : 選択した run の合成済みデザインを開きます。
- [Open Implemented Design] : 選択した run のインプリメント済みデザインを開きます。
- [Generate Bitstream] : [Generate Bitstream] ダイアログ ボックスを開き、ビットストリームを作成します。このコマンドは、完了したインプリメンテーション run に対してのみ実行可能です。詳細は、[409 ページの「ビットストリーム ファイルの生成」](#)を参照してください。
- [Copy Run] : 選択した run と同じストラテジを使用して新しい run を作成します。
- [Create New Runs] : 新しい合成 run またはインプリメンテーション run をコンフィギュレーションするための Create New Runs ウィザードが起動します。詳細は、[第 6 章「デザインの合成」](#) および [第 9 章「デザインのインプリメンテーション」](#) を参照してください。
- [Open Run Directory] : ディスク上の選択した run のディレクトリをファイル ブラウザーで開きます。
- [Launch ChipScope Analyzer] : 選択したインプリメント済みデザインの BIT ファイルを使用して ChipScope Analyzer を起動します。
- [Launch iMPACT] : 選択したインプリメント済みデザインの BIT ファイルを使用して iMPACT を起動します。
- [Launch FPGA Editor] : FPGA Editor で選択したインプリメント済みデザインを開きます。
- [Launch XPower Analyzer] : XPower Analyzer で選択したインプリメント済みデザインを開きます。
- [Export to Spreadsheet] : [Design Runs] ビューの情報をスプレッドシート ファイルにエクスポートします。
- [Promote Partitions] : [Promote Partitions] ダイアログ ボックスを開き、インプリメントされたパーティションを保持するためプロモートします。このコマンドは、デザインでパーティションを定義している場合にのみ表示されます。詳細は、[第 13 章「階層デザイン手法の使用」](#)を参照してください。

テキスト エディターの使用

PlanAhead ソフトウェアのテキスト エディターでは、さまざまなテキスト ファイルを編集できます。Verilog および Verilog ヘッダー ファイル、VHDL ファイル、制約ファイル、Tcl スクリプト、PlanAhead ジャーナルおよびログ ファイル、単純なテキスト ファイルなど、さまざまなフォーマットのファイルがサポートされています。一部のファイル タイプはそのファイル タイプ特定の編集環境で表示され、キーワードやコマンド行を識別しやすくなっています。

テキスト エディターの設定については、[197 ページの「テキスト エディターのフォント設定」](#)を参照してください。

テキスト ファイルを開く

[Sources] ビューでファイルを右クリックし、[Open File] をクリックすると、そのファイルがテキスト エディターで開きます。

または、[File] → [Open File] をクリックしてファイルを開くこともできます。このコマンドを実行するとファイルブラウザが開き、ファイルを選択して開くことができます。

[Messages] ビューでエラーまたは警告メッセージのファイル名をダブルクリックすると、そのファイルがテキスト エディターで開きます。

[File] → [Open Log File] および [File] → [Open Journal File] をクリックすると、PlanAhead のログ ファイルおよびジャーナル ファイルを開くことができます。

新規ソース ファイルの作成

[File] → [New File] をクリックすると、新規ファイルを作成してテキスト エディターで開くことができます。このコマンドは、[Tcl Console] ビュー、[Compilation] ビュー、[Messages] ビューのエラーや警告の一部をテキスト ファイルとして保存する場合に便利です。












このコマンドを実行するとファイルブラウザが開き、フォルダーと新規ファイル名を指定してファイルを作成できます。

テキスト エディターのコマンド

テキスト エディターのコマンドは、右クリックで表示されるポップアップ メニューまたはツールバー ボタンから実行できます。次のコマンドがあります。

- [Save File] : 表示されているファイルを個別に保存します。
- [Save File As] : ファイルに別の名前を付けて保存します。
- [Undo]、[Redo] : 変更を元に戻したり、やり直したりします。
- [Cut, Copy Text, Paste] : 選択したテキストを切り取るか、クリップボードにコピーし、クリップボードの内容をカーソル位置に貼り付けます。
- [Duplicate Selection] : 選択したテキストをコピーして、そのすぐ上か下のカーソル位置に貼り付けます。
- [Find/Replace] : [Find] フィールドが表示され、文字列を入力して検索し、指定したテキストへ置換します。
- [Indent Selection]、[Unindent Selection] : 選択した行の行頭にタブ スペースを追加、または行頭のタブ スペースを削除します。
- [Toggle Line Comments] : 選択したテキスト行の行頭にコメント シンボルを追加し、コメントアウトします。選択した行に既にコメント シンボルが付いている場合は、削除します。

注記 : コメント シンボルは、編集しているファイル タイプによって異なります。

	Save File	Ctrl+S
	Save File As...	
	Undo	Ctrl+Z
	Redo	Ctrl+Shift+Z
	Cut	Ctrl+X
	Copy Text	Ctrl+C
	Paste	Ctrl+V
	Duplicate Selection	Ctrl+D
	Find...	Ctrl+F
	Replace...	Ctrl+R
	Indent Selection	Tab
	Unindent Selection	Shift+Tab
//	Toggle Line Comments	Ctrl+3
	Toggle Block Comments	Ctrl+Slash
	Select All	Ctrl+A
	Toggle Column Selection	
	Tabs	
	Find in Files...	Ctrl+Shift+F
	Replace in Files...	Ctrl+Shift+R
	Change Editor Style	Ctrl+Shift+T
	Insert Template	
	Select Source File	
	Define Module...	

- [Toggle Block Comments] : 選択したテキストのブロックの冒頭と最後にブロック コメント シンボル (`/*...*/`) を追加し、コメントアウトします。既にコメント シンボルが付いている場合は、削除します。このコマンドを使用すると、複数行のテキストを 1 回の操作でコメントアウトできます。
- [Select All] : テキスト エディターに表示されているテキストをすべて選択します。
- [Toggle Column Selection] : テキストを行で選択するのではなく、行と列のグリッドとしてテキスト文字のブロックを選択できるようになります。このモードは、オンまたはオフにできます。
- [Tabs] : テキスト エディターでタブを入力したときに、タブ文字 (`\t`) を使用するか、指定数のスペースを使用するかを指定します。タブ文字が適切に処理されないサードパーティ アプリケーションでテキスト ファイルを開く場合に便利です。
- [Find in Files]、[Replace in Files] : [Find in Files] ダイアログ ボックスを開き、文字列を入力して選択したファイルから検索します。検索結果は、[Find in Files] ビューに表示されます。[Replace in Files] コマンドを使用すると、検索文字列を新しい文字列に置換することもできます。
- [Change editor style] : テキスト エディターで使用するフォントおよび色を変更します。詳細は、[197 ページの「テキスト エディターのフォント設定」](#)を参照してください。
- [Insert Template] : 選択している言語テンプレートをテキスト ファイルのカーソル位置に挿入します。詳細は、[186 ページの「言語テンプレートおよび制約テンプレートのインスタンスエート」](#)を参照してください。
- [Select Source File] : [Sources] ビューで現在のファイルを選択します。
- [Define Module] : [Define Module] ダイアログ ボックスが開きます。このダイアログ ボックスでは、開いているテキスト ファイルへ追加する新しい Verilog または VHDL モジュール定義を指定します。詳細は、[49 ページの「新規モジュールの定義」](#)を参照してください。[Define Module] ダイアログ ボックスを閉じると、新しいモジュールが開いているテキスト ファイルのカーソルの位置に挿入されます。

言語テンプレートおよび制約テンプレートのインスタンスエート

PlanAhead では、標準的な Verilog および VHDL 言語テンプレートと UCF 制約テンプレートが提供されており、ロジック構文やデザイン制約の定義に使用できます。選択したテンプレートは、テキスト エディターで開いているファイルにインスタンスエートできます。[図 4-64](#) に、[Templates] ビューにテンプレートを開いた状態の例を示します。

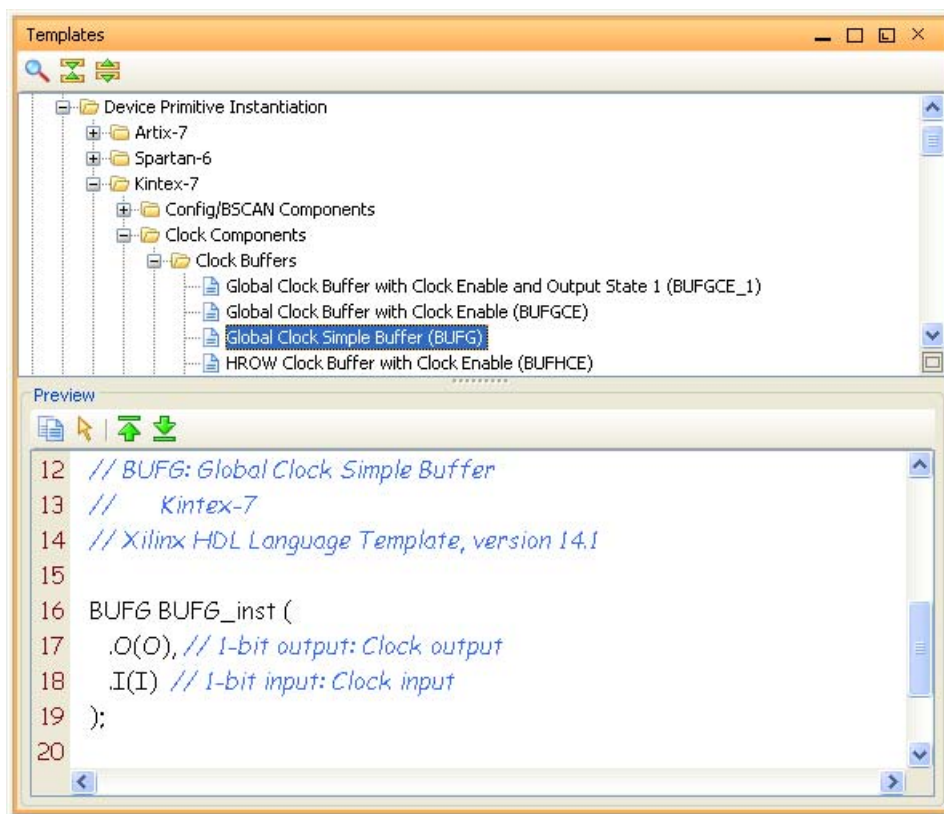


図 4-64 : 言語テンプレート

言語テンプレートをインスタンス化するには、次の手順に従います。

1. 次の方法で [Template] ビューを開いて、使用可能な言語テンプレートを確認します。
 - メイン メニューから [Window] → [Language Templates] をクリックします。
 - テキスト エディターのツールバーの [Language Template] アイコンをクリックします。
2. [Templates] ビューの [VHDL]、[Verilog]、または [UCF] フォルダーからテンプレートを選択します。図 4-64 は、ザイリンクスの提供するテンプレート例を示しています。

[Preview] ウィンドウにテキスト エディターに挿入されるテンプレートが表示されます。
[Preview] ウィンドウに表示されるテンプレートは編集できません。

3. テキスト エディターで、テンプレートを挿入する位置をクリックします。カーソルをテンプレートの冒頭を挿入する位置に配置してください。

注記：テンプレートを挿入する際、既存のテキストは上書きされません。テキスト エディターでテキストを選択している場合、そのテキストの後ろにテンプレートのテキストが追加されます。

4. テキスト エディターのツールバーまたはポップアップ メニューから [Insert Template] をクリックし、カーソル位置にテンプレートを挿入します。
5. デザイン要件に応じて挿入したテンプレート テキストを編集します。

PlanAhead の設定

PlanAhead の設定オプションには、選択規則、ショートカット キー、一般設定、およびウィンドウ設定があります。次のセクションで、設定オプションについて説明します。

PlanAhead の全般的なオプションの設定

PlanAhead のオプションを設定するには、[Tools] → [Options] をクリックします。[PlanAhead Options] ダイアログ ボックスが開き、PlanAhead のアプリケーションとしての動作を設定できます。

これらのオプションには [Project Settings] ダイアログ ボックスのオプションと重複しているものもありますが、[Project Settings] ダイアログ ボックスのオプションはそのプロジェクトにのみ適用されます。詳細は、[101 ページ](#)の「プロジェクト設定」を参照してください。

[189 ページ](#)の[図 4-65](#)に、[PlanAhead Options] ダイアログ ボックスの [General] ページを示します。

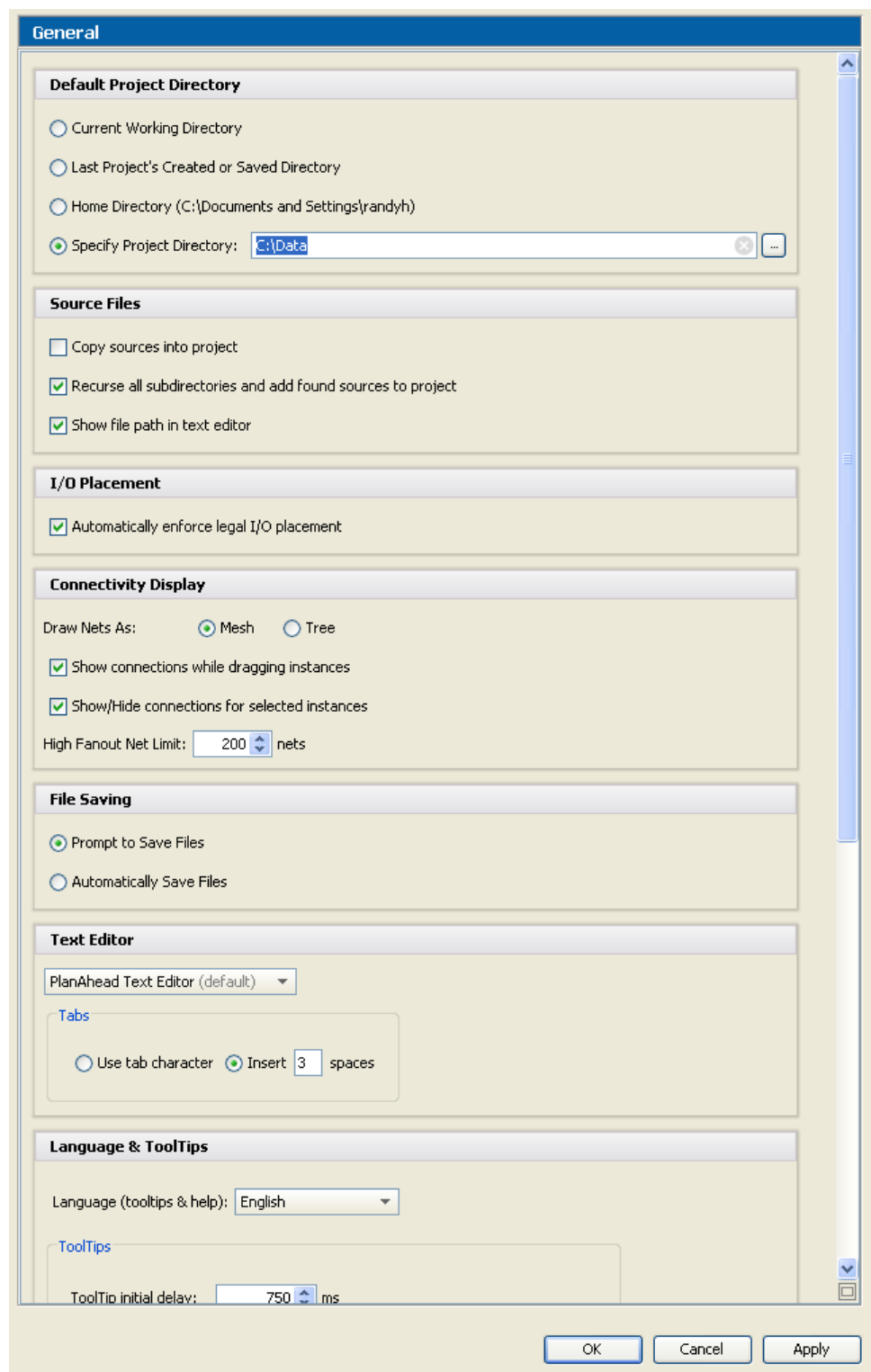


図 4-65 : [PlanAhead Options] ダイアログ ボックスの [General] ページ

次のオプションがあります。

- [Default Project Directory] : 既存のプロジェクトの検索場所および新規プロジェクトの保存先を指定します。
- [Source Files] : ソース ファイルをプロジェクトにコピーするオプション ([Copy sources into project])、すべてのサブディレクトリのソース ファイルをコピーするオプション ([Recurse all subdirectories and found sources to project])、ファイル パスをテキスト エディター表示するオプション ([Show file path in text editor]) のオン/オフを切り替えるチェック ボックスがあります。
- [I/O Placement] : インタラクティブな I/O 配置 DRC のオン/オフを切り替えます。
- [Connectivity Display] : [Device] ビューでの接続の表示方法を指定します。詳細は、[155 ページの「\[Device\] ビューの表示オプションの設定」](#)を参照してください。
- [File Saving] : プロジェクト ファイルを閉じたときに、プロジェクト ファイルを自動的に保存するか、保存されていない変更がある場合に保存を確認するダイアログ ボックスを表示するかを指定します。
- [Text Editor] : RTL ソース ファイルまたは UCF ファイルを開くときに使用するテキスト エディターを指定します。ドロップダウン リストからあらかじめ定義されているテキスト エディターを選択するか、サードパーティのテキスト エディターを起動するコマンド ラインを指定します。デフォルトでは PlanAhead テキスト エディターが選択されています。指定したテキスト エディターで問題がある場合にも、PlanAhead テキスト エディターが使用されます。
- [Tabs] : テキスト エディターでタブを入力したときに、タブ文字 (**⌘**) を使用するか、指定数のスペースを使用するかを指定します。タブ文字が適切に処理されないサードパーティ アプリケーションでテキスト ファイルを開く場合に便利です。

注記：PlanAhead には、サードパーティ テキスト エディターでサポートされない機能もあります。

- [Language & Tooltips] : PlanAhead ツール内でコマンド上にカーソルを置くと表示されるツール ヒントの言語および動作を指定します。サポートされる言語は、英語、中国語、日本語です。
- [WebTalk] : WebTalk でザイリンクスに使用状況情報を送信するかどうかを指定します。®
- [IP Catalog] : プロジェクトで使用する IP カタログを設定およびアップデートできます。詳細は、[83 ページの「IP カタログのアップデート」](#)を参照してください。
- [CoreGen] : IP コアを生成するときに、Java に割り当てるメモリ量を指定します。大型の IP コアを生成する場合は Java に割り当てるメモリ量を増やすと有益な場合があります。また、初期化エラーが発生した場合は、割り当てるメモリ量を減らした方がよい場合があります。
- [Miscellaneous] : ザイリンクス ウェブサイトでのソフトウェア アップデートを自動確認するオプションを設定し、Getting Started ページで表示する最近開いたプロジェクトおよびディレクトリの数を定義します。

表示環境の設定

PlanAhead には、ユーザー設定可能な表示オプションが多数あります。ツールはデフォルト設定で提供されていますが、これをカスタマイズし、保存できます。

PlanAhead では、表示環境が次のように管理されます。

- 各ビュー レイアウトが個別に保存されます。
- ウィンドウ サイズおよび位置がレイアウト ファイルに保存され、次の使用時に復元されます。

- PlanAhead を閉じるときに、保存されたビュー設定がホーム ディレクトリに保存されます (付録 A の「デフォルト環境の出力」を参照)。

色の指定

ビューの表示オプションを変更すると、環境の表示や動作を制御できます。

表示オプションを表示または編集するには、[Tools] → [Options] をクリックし、左側のペインで [Colors] をクリックします (191 ページの図 4-66)。オプションを変更して [OK] または [Apply] をクリックすると、変更が反映されます。

淡色と濃色の両方のデフォルト背景設定があります。[Themes] プルダウン メニューで、[PlanAhead Light Theme] または [PlanAhead Default Theme] のいずれかを選択します。

これらのデフォルト オプションは、planahead.ini ファイルに定義されています。詳細は、付録 A の「デフォルト環境の出力」を参照してください。

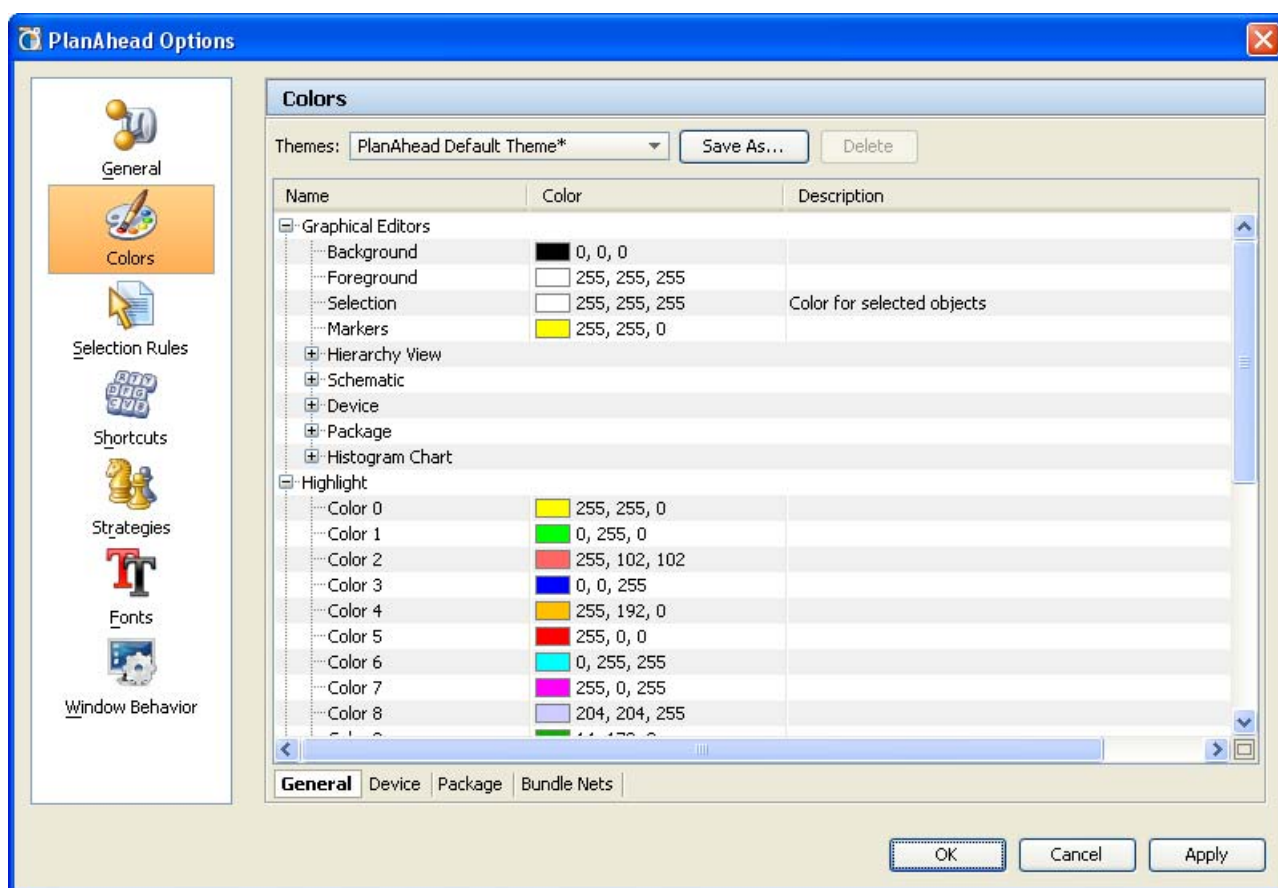


図 4-66 : テーマ オプション

表示テーマのカスタマイズ

[Themes] オプションでは、PlanAhead の表示環境を制御するオプションを変更できます。[Themes] オプションの、[General]、[Device]、[I/Os]、および [Bundle Nets] タブをクリックして、表示設定を変更できます。

[General] タブ

背景色やハイライト色など、表示環境のさまざまなエレメントの色を指定します。

特定のエレメントの色は、次のように変更できます。

- [Color] 列をクリックし、ドロップダウン リストに表示される色から選択します。
- [More Colors] をクリックすると、より多数の色から選択できます。
- [Color] 列のセルに直接 RGB 値を入力します。

Name	Color	Description
Graphical Editors		
Background	240, 240, 240	
Foreground	0, 0, 0	
Selection	255, 102, 0	Color for selection
Markers	255, 255, 0	
Highlight		
Hierarchy View		
Schematic Viewer		
Histogram Chart		
Console		
Background		
Foreground		
Command text		Color for Tcl command
Error text		Color for error message
Warning text	204, 102, 0	Color for warning
Windows		

[Device] タブ

[Device] ビューでのオブジェクトのデフォルト色および選択可/不可を設定します。

- [Select] 列のチェックボックスをオフにすると、そのオブジェクト タイプを選択できなくなります。
- [Device] ビューのアイテムの表示は、レイヤー スライドアウトで制御します。詳細は、[152 ページの「\[Device\] ビュー」](#)を参照してください。

Object Type	Select	Frame Color	Fill Color
Pblock 1st Level	<input checked="" type="checkbox"/>	255, 15...	
Pblock 2nd Level	<input checked="" type="checkbox"/>	153, 51...	170, 1...
Pblock 3rd+ Levels	<input checked="" type="checkbox"/>	102, 0, 204	225, 1...
Assigned Instance			
I/O Net			
Placed Port			0, 255...
Fixed Port			255, 1...
Placed Instance			0, 215...
Fixed Instance			255, 1...
Path			255, 2...
Tile Outline			
Site Tri-mode Ethernet M			
Site System Monitor	<input checked="" type="checkbox"/>	169, 16...	
Site SLICEX	<input checked="" type="checkbox"/>	51, 51, 51	

注記：[Device] ビューの一部のエレメント

はデバイス固有であり、色や選択の可/不可の設定は、そのデバイスを使用した場合にのみ有効です。

[Package] タブ

[Package] ビューでの各オブジェクト タイプのデフォルト色を設定します。

- [Package] ビューのエレメントの表示は、レイヤー スライドアウトで制御できます。詳細は、[159 ページの「\[Package\] ビュー」](#)を参照してください。

Object Type	Frame Color	Fill Color
I/O Pin	0, 0, 255	128, 1...
Input Pin	0, 0, 255	128, 1...
Global Clock Pin	255, 0, 0	153, 1...
Clock Capable Pin	255, 0, 0	153, 2...
VCC Pin	192, 19...	181, 2...
GND Pin	192, 19...	40, 12...
Special Pin	0, 0, 0	102, 1...
Config Pin	0, 0, 0	255, 1...
JTAG Pin	0, 0, 0	255, 1...
Power Management Pin	0, 0, 0	255, 1...
Temp Sensor Pin	0, 0, 0	255, 1...
SYSMON Pin	0, 0, 0	255, 1...
GT Pin	255, 0, ...	247, 2...

[Bundle Nets] タブ

表示されるバンドル ネットの特徴を設定します。

- [From] および [To] 列で、バンドルの信号数の範囲を指定します。各行でバンドル ネットの範囲を個別に設定します。
- [Width] 列で、[Device] ビューのライン幅を指定します。

From	To	Display	Select	Width	Color
1	1	<input type="checkbox"/>	<input type="checkbox"/>	1	128...
2	20	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	148...
21	60	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	4	255...
61	200	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	6	255...
201	500	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	8	102...
501	1,000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	10	51, ...
1,001		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	12	153...

- [Display] 列のチェック ボックスをオフにすると、そのバンドルが非表示になります。
- [Select] 列のチェックボックスをオフにすると、そのオブジェクト タイプを選択できなくなります。

カスタム テーマの保存

[Colors] ページでさまざまな色設定を定義したら、今後のセッションで使用できるよう保存します。これには、[Themes] プルダウン メニューの横にある [Save As] ボタンをクリックします。

カスタム テーマを作成する場合は、カスタム設定が保存されている初期化ファイルのバックアップを作成することをお勧めします。PlanAhead のデフォルトおよびカスタムの初期化ファイルについては、付録 A の「デフォルト環境の出力」を参照してください。

選択規則オプションの設定

[PlanAhead Options] ダイアログ ボックスの [Selection Rules] ページでは、すべてのビューのオブジェクトの選択方法を制御します。オブジェクトを選択すると、ほかのオブジェクトも同時に選択されることがあります。たとえば、Pblock を選択すると、割り当てられたネットリストのインスタンスも選択されます。

選択規則の設定の詳細は、133 ページの「オブジェクトの選択規則」を参照してください。

ショートカット キーの設定

頻繁に使用されるコマンドには、キーボードのキーを組み合わせたショートカットが定義されています。ショートカットは、ポップアップ メニューのコマンドの横に表示されます。たとえば、F9 キーを押すと [Fit Selection] コマンドを実行できます。

194 ページの図 4-67 に示す [PlanAhead Options] ダイアログ ボックスの [Shortcuts] ページでショートカットを変更できます。

[PlanAhead Options] ダイアログ ボックスの [Shortcuts] ページでは、新しいカスタム ショートカットを作成できます。

[Shortcuts] ページの上部では、使用可能なショートカット スキーマを使用することで、ショートカットを管理できるようになっています。PlanAhead ツールで提供されるデフォルトのショートカットは変更できません。カスタマイズするには、新しいショートカット スキーマを作成する必要があります。

1. デフォルトのスキーマに基づいて新しいスキーマを作成するには、[Copy] をクリックします。プルダウン メニューからスキーマを選択して、そのスキーマを有効にします。スキーマのショートカットは、ダイアログ ボックス下部で変更できます。
2. メニューおよびビューのリストを検索してコマンドを選択すると、新しいショートカットを入力できます。

[Filter] フィールドを使用すると、ショートカットを検索できます。このフィールドに文字列を入力して、コマンドをフィルターします。異なるビューの同じコマンドに異なるショートカットを割り当てることもできます。
3. [Add Shortcut] ボタンをクリックします。
4. [Add Shortcut] ダイアログ ボックスで新しいショートカットを入力して [OK] をクリックします。

ユーザー指定のショートカットは、PlanAhead ツールのコンフィギュレーション ディレクトリ
の shortcuts.xml ファイルに保存されます。詳細は、付録 A の「デフォルト環境の出力」を
参照してください。

5. ショートカットを削除するには、[Remove] ボタンをクリックします。

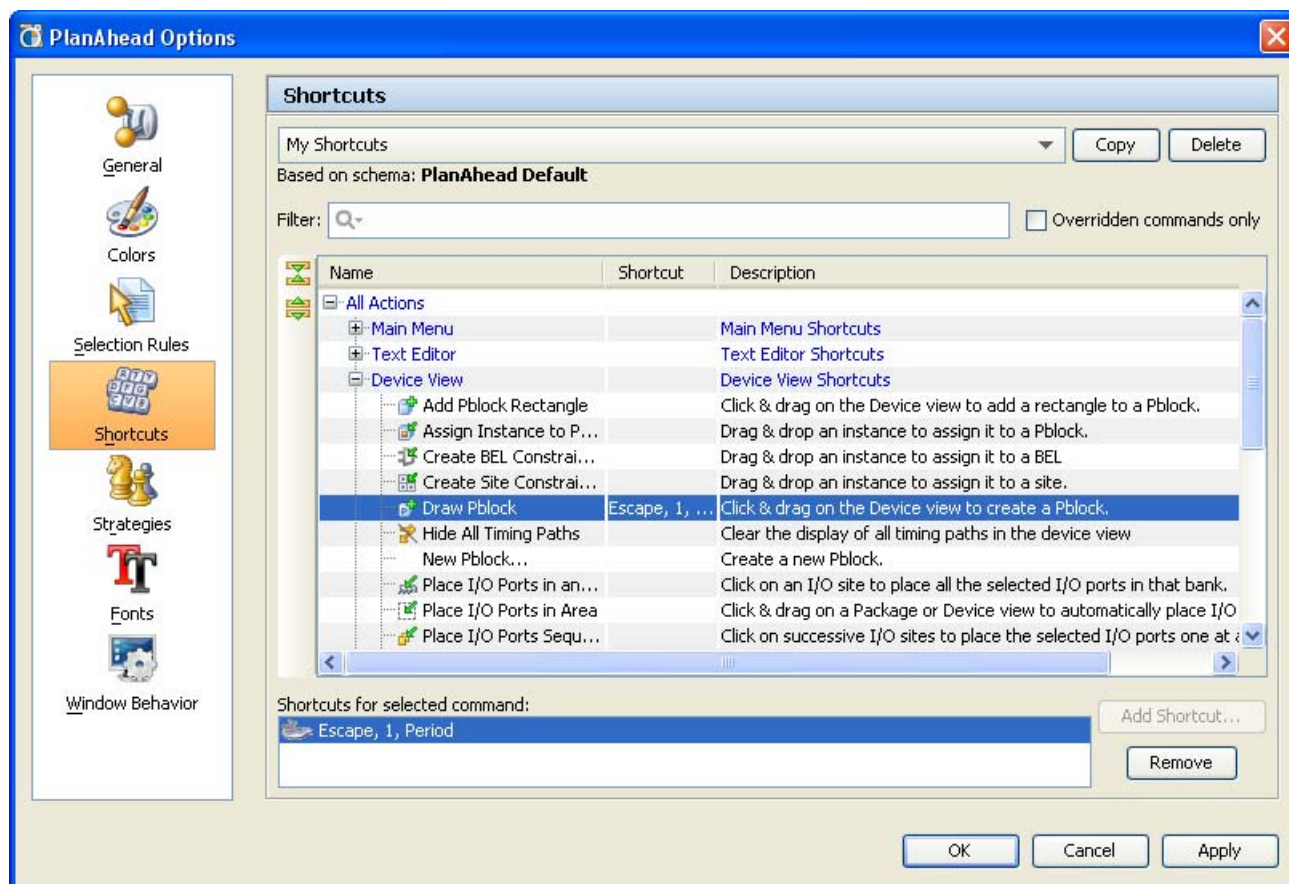


図 4-67：ショートカット オプション

合成およびインプリメンテーション ストラテジの定義

ストラテジは、デザインの合成またはインプリメンテーションで最適な結果が得られるようにするために定義されたソリューションです。合成アプリケーションおよびインプリメンテーションで実行されるさまざまなプログラムのコマンド ライン オプションのあらかじめ定義されたセットとして定義されます。ストラテジは、ツールおよびバージョン特定です。各メジャー リリースには、そのバージョン専用のコマンド ライン オプションがあります。ストラテジの設定については、105 ページの「[Project Settings] ダイアログ ボックスの [Synthesis] ページ」および 107 ページの「[Project Settings] ダイアログ ボックスの [Implementation] ページ」を参照してください。

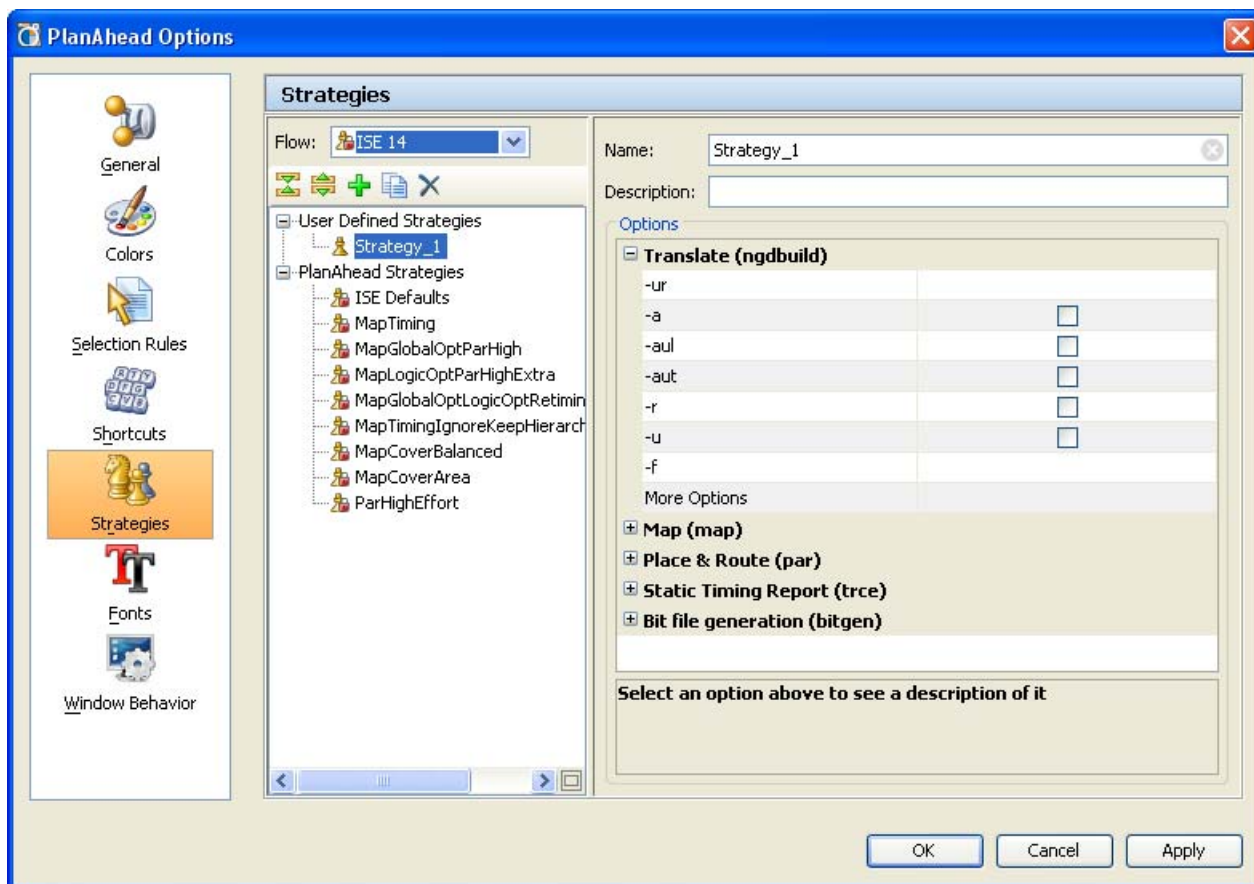


図 4-68 : [Strategies] ページ

PlanAhead には、内部ベンチマークでテストされた、一般的なストラテジが複数含まれています。これらの合成およびインプリメンテーション ストラテジのコマンド ライン オプション設定は変更できませんが、提供されているストラテジをコピーし、それを変更することはできます。

コピーされたストラテジは、次のディレクトリに書き出されます。

- Windows 7 - C:\Users\username\AppData\Roaming\Xilinx\PlanAhead\strategies
- Windows XP - C:\Documents and Settings\username\Application Data\Xilinx\PlanAhead\strategies

ストラテジを確認、コピー、変更するには、次の手順に従います。

1. [Tools] → [Options] をクリックします。
2. 左側のペインで [Strategies] をクリックします。195 ページの図 4-68 に示す [PlanAhead Options] ダイアログ ボックスの [Strategies] ページに、各 ISE と XST のバージョン用にあらかじめ定義されたストラテジがリストされます。
3. [Flow] ドロップダウン リストで、合成の場合は XST を、インプリメンテーションの場合は ISE のバージョンを選択します。提供されているストラテジと設定されているコマンド ライン オプションが表示されます。

特定の XST および ISE のオプションについては、付録 E 「その他のリソース」 に示す『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。

4. 新しいストラテジを作成するには、ツールバーまたはポップアップ メニューから [Add Strategy] コマンドを使用します。

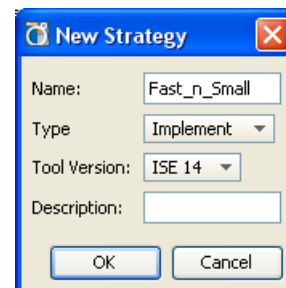


ツールバーまたはポップアップ メニューから [Create a copy of this strategy command] コマンドを使用しても、既存のストラテジをコピーできます。ストラテジが [User Defined Strategies] リストにコピーされ、右側に変更可能なコマンド ライン オプションがリストされます。



5. 新しいストラテジに対して次の情報を入力します。

- [Name] : ストラテジの名前を入力します。
- [Type] : 合成またはインプリメントを指定します。
- [Tool Version] : XST または ISE のバージョンを指定します。
- [Description] : ストラテジの説明を入力します。ここで入力した説明が [Design Run] ビューの結果の表に表示されます。



6. 合成またはインプリメンテーション run 中に使用されるコマンド ライン ツールのオプションを変更します。

特定のコマンド オプションをクリックすると、その説明が下に表示されます。

7. 右側のコマンド オプション エリアをクリックし、ドロップダウン メニューから値を選択するか、値を入力します。195 ページの図 4-68 に示すように、使用可能なコマンド オプションの値がドロップダウン メニューに表示されます。PlanAhead であらかじめ設定されているストラテジのデフォルト設定は変更できません。

8. [Apply] をクリックし、[OK] をクリックして新しいストラテジを保存します。

図 4-69 に示すように、新しいストラテジが [User Defined Strategies] の下に表示され、合成およびインプリメンテーションで使用できるようになります。

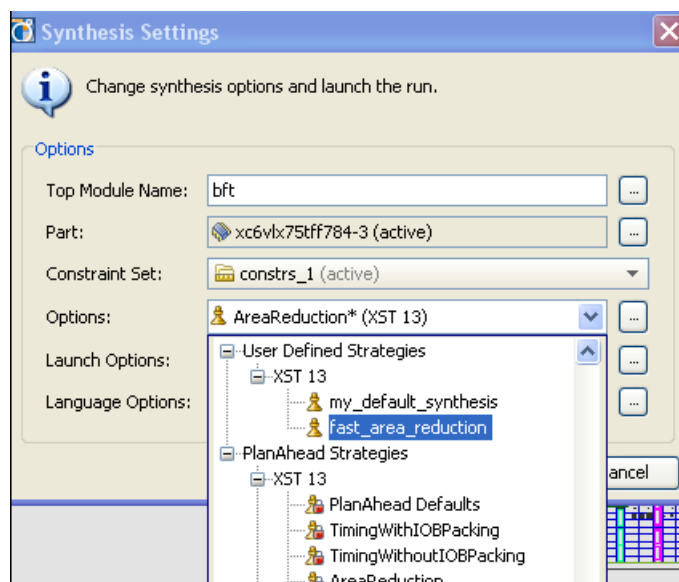


図 4-69 : ユーザー定義ストラテジの選択

作成したストラテジを複数のユーザーで共有するには、ユーザー定義のストラテジを <InstallDir>/strategies ディレクトリ (<InstallDir> は PlanAhead ソフトウェアのインストール ディレクトリ) にコピーします。

テキスト エディターのフォント設定

テキスト エディターで使用するフォントおよび色を定義できます。コメントやキーワードなどのテキスト ファイルの異なるエレメントに対して、それぞれフォント、サイズ、色を指定できます。
 図 4-70 に、コメントの色を青に指定した例を示します。

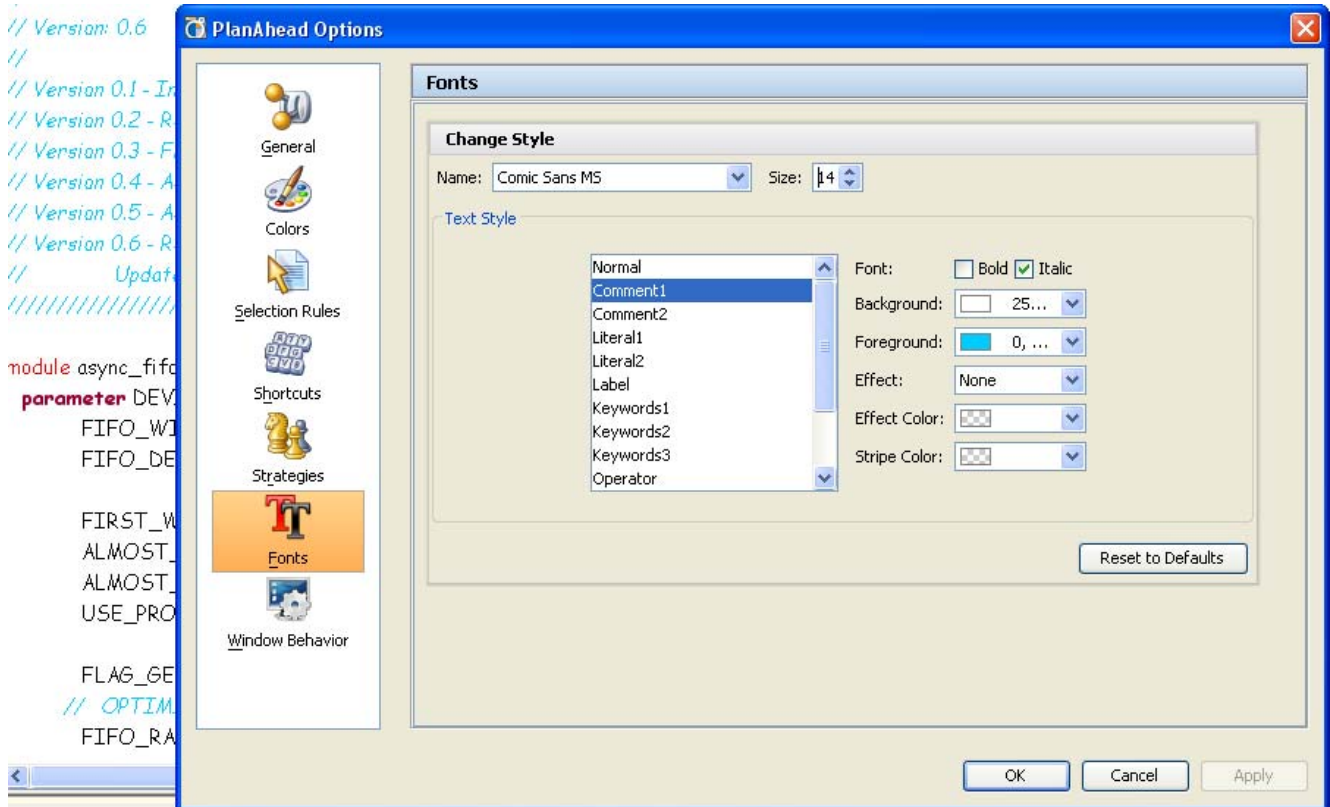


図 4-70 : PlanAhead テキスト エディターのフォント設定

[Reset to Default] : デフォルトのフォント設定に戻します。

ウィンドウ動作の設定

ウィンドウ動作のオプションを設定するには、[Tools] → [Options] → [Window Behavior] をクリックします。198 ページの図 4-71 に、[PlanAhead Options] ダイアログ ボックスの [Window Behavior] ページを示します。

- [Warnings & Confirmations] : 警告および確認ダイアログ ボックスの表示方法を設定します。
- [Notifications] : 表示させる通知を設定します。
- [Alerts] : 非アクティブな run の成功またはエラーに対するアラートを設定します。

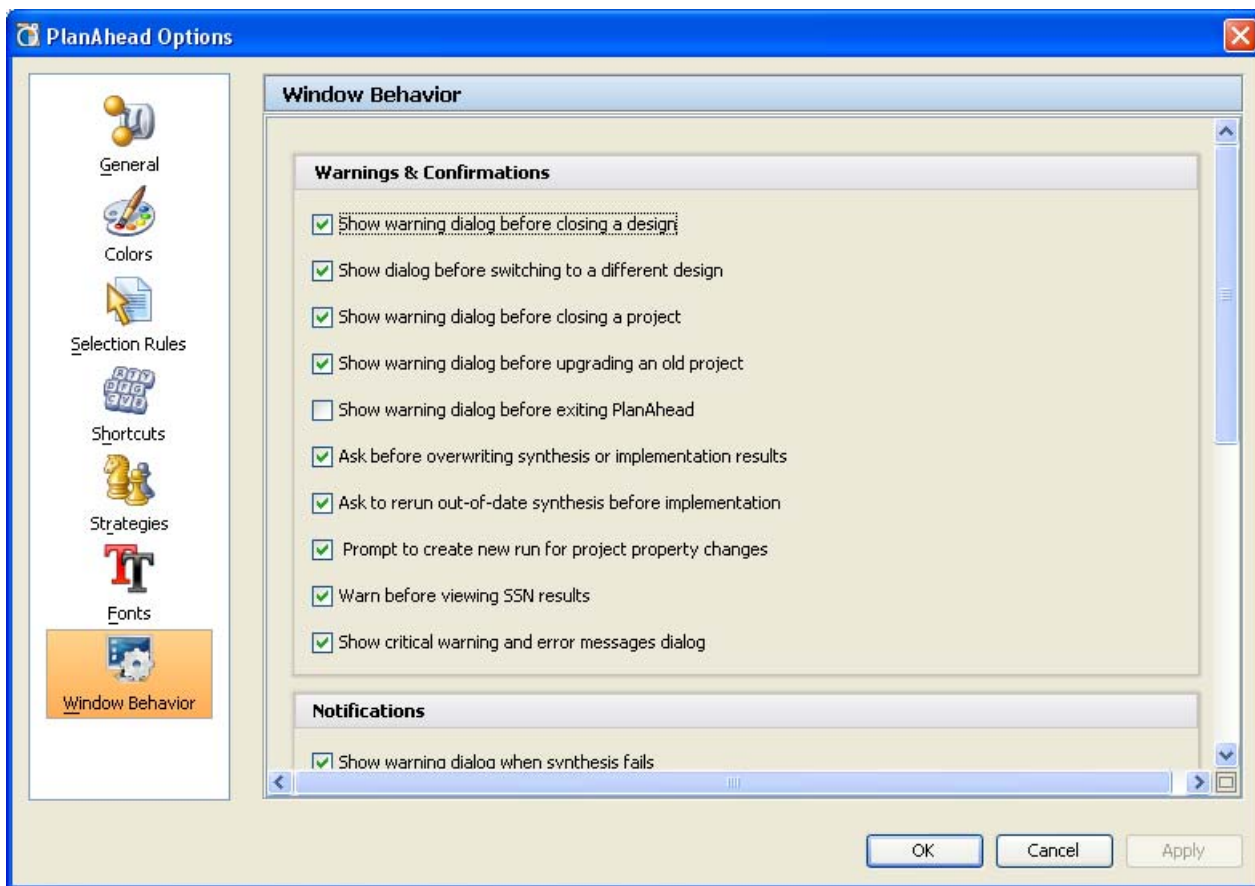


図 4-71：ウィンドウ動作オプション

カスタム メニュー コマンドの追加

[Tools] → [Custom Commands] → [Customize Commands] をクリックすると、システムまたはユーザー定義の Tcl コマンドをメイン メニューやメイン ツールバー メニューに追加できます。これにより、よく使用する Tcl コマンドをグラフィカル環境の GUI に追加できます。

[Customize Commands] を実行すると、199 ページの図 4-72 に示すような [Customize Commands] ダイアログ ボックスが開きます。

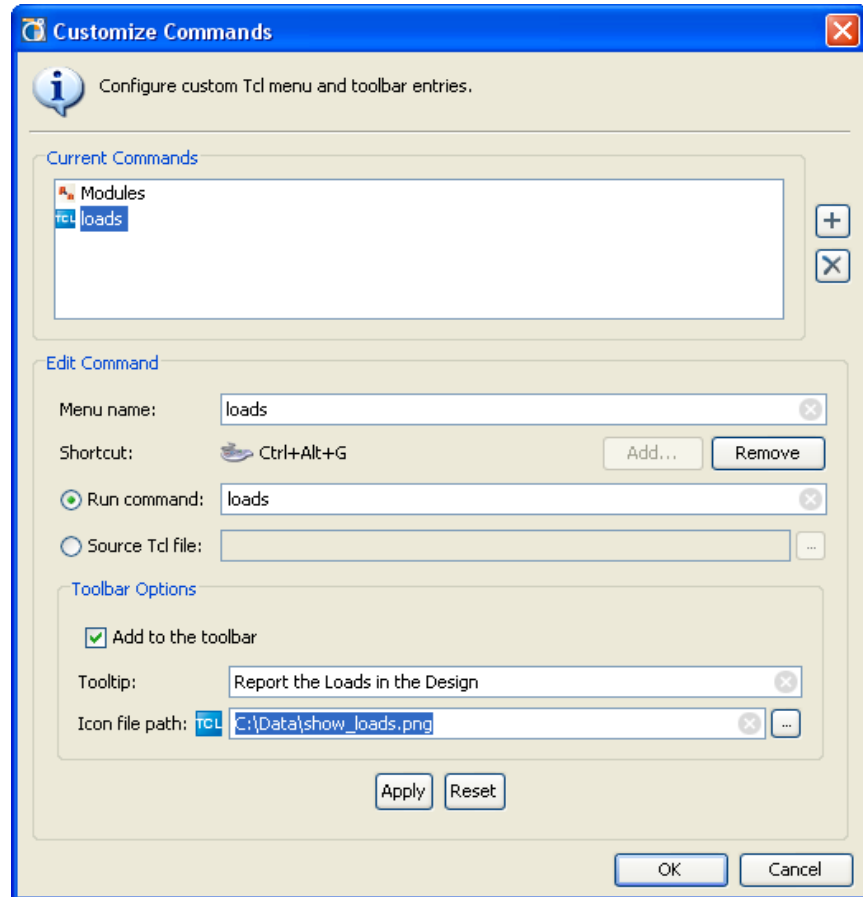
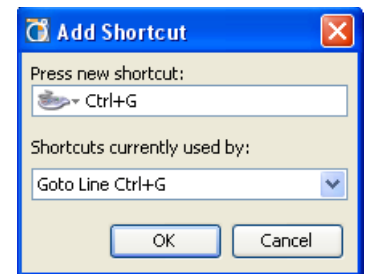


図 4-72 : メニュー コマンドのカスタマイズ

[Customize Commands] ダイアログ ボックスには、次のようなオプションがあります。

- [Current Commands] : 現在定義されているカスタム コマンドのリストが表示されます。
 - [Add a New Command] : + アイコンで表示され、カスタム メニューに新しいコマンドを追加します。[Add a New Command] をクリックし、コマンド名を入力して [Enter] をクリックすると、そのコマンドがカスタム コマンド リストに追加されます。
 - [Remove Selected Commands] : X アイコンで表示され、1 つまたは複数のコマンドをカスタム メニューから削除します。
- [Edit Command] : [Current Commands] リストで選択したコマンドの属性を指定します。
 - [Menu Name] : [Custom Command] メニューに表示されるコマンドの名前を定義します。
 - [Shortcut] : その Tcl コマンドを実行するのに使用するキーボード ショートカットを定義します。[Add] をクリックすると、[Add Shortcut] ダイアログ ボックスが開きます。選択したコマンドを実行するキーボードの組み合わせを入力してください。ダイアログ ボックスには、既に使用されているショートカットが表示されます。
- [Run Command] : オンにすると、指定した Tcl コマンドや手順をカスタム メニューから実行できるようになります。



- [Source Tcl File] : オンにすると、カスタム メニューから 1 つの Tcl コマンドや手順を実行するのではなく、指定した Tcl スクリプト ファイルが実行できます。
- [Toolbar Options] : コマンドのアイコンをメイン ツール バー メニューに追加するかどうか指定します。
 - [Add to the toolbar] : ツールバー アイコンをオンにするチェック ボックスです。オフにすると、コマンドはメイン ツールバー メニューに表示されません。
 - [Tooltip] : カーソルをコマンド アイコンの上に置いたときにツール ヒントを表示するかどうか指定します。
 - [Icon file path] : ツールバー アイコンへのファイル パスを指定します。

注記：アイコン ファイルは、約 20x20 ピクセルの PNG、JPG、または GIF ファイルにする必要があります。大きな画像はツールバーに収まるようにサイズが変更されます。

現在定義されているカスタム コマンドは、[Tools] → [Custom Commands] をクリックすると、その下のレベルのメニューに表示されます。

[Custom Commands] は、PlanAhead を起動するたびに復元されます。カスタム コマンドは `commands.paini` ファイルに保存されます。詳細は、[付録 A の「デフォルト環境の出力」](#)を参照してください。

エラボレート済み RTL デザイン

PlanAhead ツールの Project Manager では、各レジスタ転送レベル (RTL) ファイルを作成および管理でき、FPGA デザインの内容でそれらをエラボレートおよび解析できます。PlanAhead ツールには、次の機能が含まれます。

- ソース ファイル管理
- Verilog および VHDL ファイル用の文脈依存のテキスト エディター
- RTL 回路図ビューアー
- RTL デザイン ルール チェック (DRC)
- ビヘイビアー シミュレーション
- RTL パワー エスティメーター

デザイン ソース ファイルの管理

プロジェクトには、次のデザイン ソース ファイル タイプを追加し、管理できます。

- Verilog および VHDL RTL ソース ファイル
- NGC、EDIF、およびザイリンクス CORE Generator™ XCO ファイル
- UCF 制約ファイル
- シミュレーション テストベンチ
- DSP モジュール用 System Generator モデル (MDL)
- エンベデッド ソース用 XPS プロジェクト (XMP)
- ブロック メモリ マップ (BMM) ファイル

ソース ファイルは、主に [Sources] ビューから操作します。詳細は、[144 ページの「\[Sources\] ビュー」](#)を参照してください。

RTL ソース ファイルの編集

PlanAhead RTL 環境には、RTL ソースを作成および変更するためのテキスト エディターが含まれています。テキスト エディターには、構文認識機能があるので、RTL キーワードが認識されて色分けされます。複数のファイルを同時に開くことができ、ワークスペースにタブ付きのビューとして表示されるので、開いているすべてのファイルにアクセスできます。

ファイルを変更して保存していない場合は、ビュー タブのファイル名の横にアスタリスク (*) が表示されます。

ファイルを保存するには、次のいずれかを実行します。

- テキスト エディターを右クリックし、[Save File] をクリックします。

- テキスト エディターのツールバーにある [Save File] ボタンをクリックします。
- [File] → [Save File] をクリックします。



ファイルを閉じるときに保存していない変更がある場合は、変更を保存するかどうかを確認するダイアログ ボックスが表示されます。ソース ファイルは、[Save As] コマンドで新しいディレクトリに保存することもできます。

テキスト エディターの使用

テキスト エディターと、[Schematic]、[Messages]、[RTL Netlist]、[Hierarchy] ビューとは、クロスプローブが可能です。テキスト エディターの詳細は、[184 ページの「テキスト エディターの使用」](#)を参照してください。

[Find] コマンドを使用したソース ファイルの検索

[Find] および [Find in Files] コマンドを使用すると、開いているソース ファイルまたは選択したファイル内で文字列を検索できます。次の操作を実行できます。

- 検索条件として、ワイルドカード (*) を含む任意のテキスト文字列を入力できます。
- フィルター オプションを使用して、ソース ファイル、制約ファイル、レポート ファイルを検索できます。

詳細は、[137 ページの「\[Find\] コマンド」](#)を参照してください。

RTL ソース ファイルのエラボレーションと解析

PlanAhead では、合成を実行しなくても RTL ソース ファイルをエラボレートおよび解析できます。エラボレート済みデザインを開くと、RTL ソース ファイルと制約ファイルがメモリにエラボレートされ、[RTL Netlist]、[Schematic]、[Messages and Compilation] ビューが開きます。[203 ページの図 5-1](#) は、エラボレート済みデザインを示しています。

エラボレート済みデザインは、デザイン プロセスのどの段階でも開くことができ、デザインを表示、解析、変更できます。

RTL ソースのエラボレーション

プロジェクトでイネーブルになっている RTL ソース ファイルは、合成中にコンパイルされているかどうかにかかわらず、デザインの一部分としてすべてエラボレートされます。エラボレーションおよびコンパイルに関するメッセージは、[Messages] ビューに表示されます。

エラボレーションに使用される HDL 言語は、[Project Settings] ダイアログ ボックスの [General] ページで選択できます。詳細は、[102 ページの「\[Project Settings\] ダイアログ ボックスの \[General\] ページ」](#)を参照してください。

エラボレーション結果は、デザインには保存されず、エラボレート済みデザインを開くたびに、エラボレーションが再実行されます。エラボレート済みデザインを合成すると、合成済みデザインとして保存されます。

デザイン ソース ファイルをプロジェクトにインポートした後、次のいずれかのコマンドを使用してデザインをエラボレートできます。

- Flow Navigator の [RTL Analysis] メニューから [Open Elaborated Design] をクリックし、エラボレート済みネットリスト、アクティブな制約セット、ターゲット デバイスをメモリへ読み込みます。
- メイン メニューから [Flow] → [Open Elaborated Design] をクリックします。
- Flow Navigator の [RTL Analysis] メニューを右クリックし、[New Elaborated Design] をクリックします。
- メイン メニューから [Flow] → [New Elaborated Design] をクリックします。

[Elaborated Design] をクリックすると、RTL ソース ファイルがエラボレートされ、最上位回路図表示が生成され、デフォルトのビュー レイアウトでデザインが表示されます。図 5-1 は、エラボレート済みデザインのデフォルトのビュー レイアウトを示しています。

ほとんどの場合は、PlanAhead で自動的に最上位モジュールが特定されます。候補が複数ある場合は、選択することを尋ねるメッセージが表示されます。[Sources] ビューの [Hierarchy] タブでは、最上位モジュールは専用のアイコンで表示されます。

最上位モジュールは、[Sources] ビューのポップアップ メニューから [Set as Top] コマンドを使用して手動で定義することもできます。

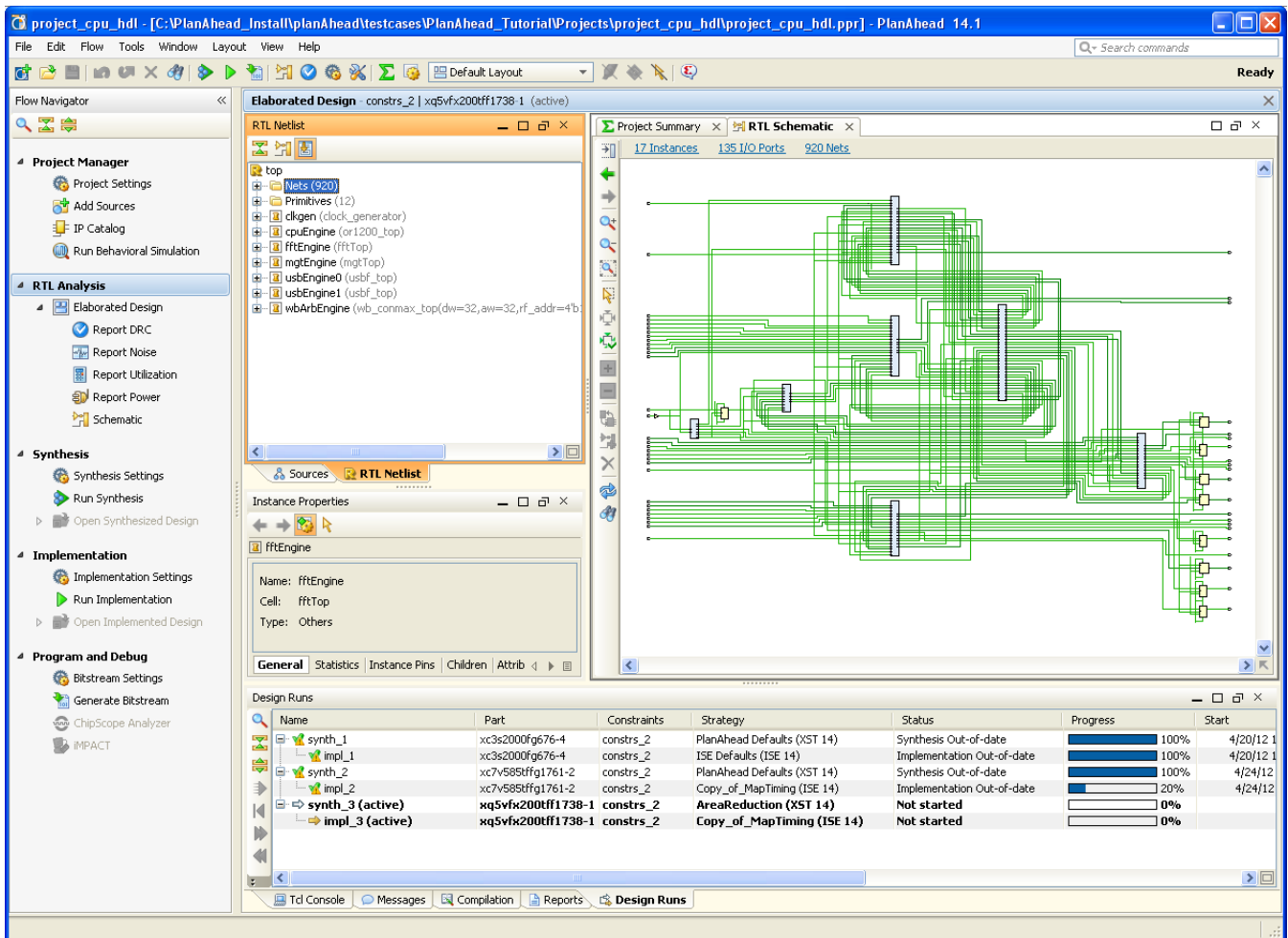


図 5-1：エラボレート済みデザインの表示

[Messages] ビューにコンパイル結果が表示され、RTL ソース ファイルのに問題がある場合は [Analysis] というカテゴリの下に表示されます。

RTL エラボレーションの結果で、エラー、警告、情報メッセージの表示/非表示を制御できます。[Messages] ビューの上部にあるチェック ボックスのオン/オフを切り替え、エラー、重要な警告、警告、情報メッセージを表示/非表示にします。

[Messages] ビューでエラーまたは警告メッセージを選択すると、該当する RTL ソース ファイルがテキスト エディターに読み込まれ、問題のソース コードがハイライトされます。

エラボレート済みデザインのリソース使用

PlanAhead では、エラボレート済みデザインにアクティブ制約セットも適用されるため、次を実行できるようになります。

- RTL ポート リストに基づく I/O ピン配置
- RTL ロジック階層からのモジュール レベルのフロアプラン

制約セットの作成と管理については、[62 ページの「制約の管理」](#)を参照してください。

PlanAhead では、エラボレート済みデザインに基づくデバイス リソース予測が表示されます。リソース予測に情報が表示されるようにするには、次をクリックします。

- [Tools] → [Report Utilization]

[Report Resource Utilization] ビューがワークスペースに開きます。[図 5-2](#) はこのビューの例を示しています。

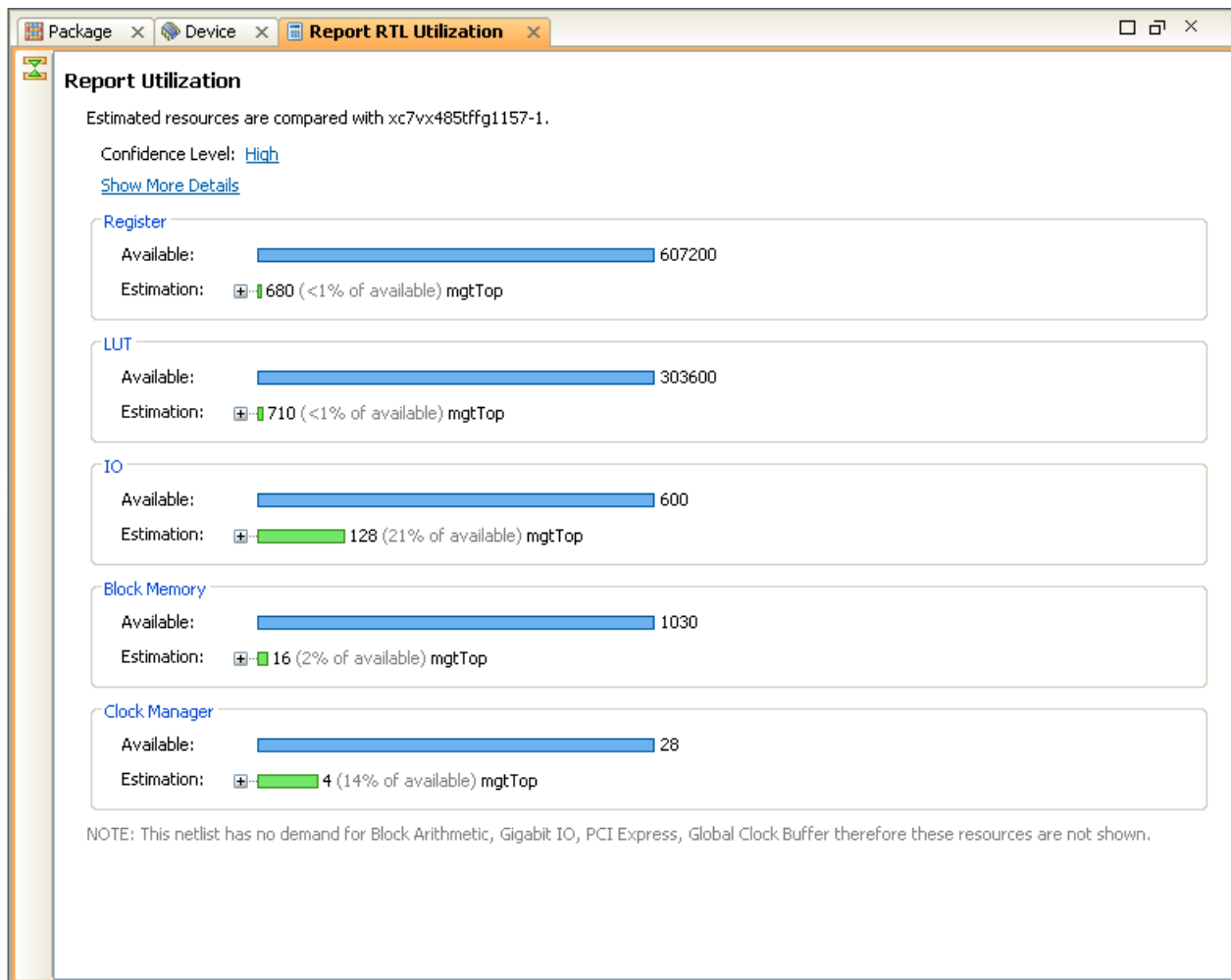


図 5-2 : エラボレート済みデザインのリソース使用

[Resource Utilization] ビューには、レジスタ、LUT、ブロック メモリなどのデバイス リソースの使用率が示されます。[Resource Utilization] ビューに表示されるリソースは、デザインのロジック階層に基づいています。デバイス リソースのあるタイプがエラボレート済みデザインで使用されていない場合は、表示されません。ビューの下部にデバイス リソースが表示されていない理由が記述されています。

次の操作を実行できます。

- 各リソースのロジック ツリーのプラス記号 (+) またはマイナス記号 (-) をクリックして、リソースの階層を展開したり閉じたりします。
- 展開した階層からデザイン オブジェクトを選択して、そのオブジェクトの詳細を表示します。

[Resource Utilization] ビューでオブジェクトを選択すると、RTL ネットリストでもそのオブジェクトが選択され、階層のハードウェア要件をより詳細に理解できます。

エラボレート済みデザインでのハードウェア リソース予測は合成前のデザイン データから求められた初期段階の予測であり、合成またはインプリメンテーションを実行すると変更される可能性があります。リソース使用は簡単な予測であり、精度は $\pm 15\%$ です。

[Resource Utilization] ビューに表示される **Confidence Level** は、デザインに含まれるブラック ボックスの数、バス幅、マクロ タイプなどの特徴に基づいて、リソース予測がどれだけ正しいかを示したものです。

Confidence Level の値のリンクをクリックすると、図 5-3 に示すように、解析されたデザインの特徴が表示されます。

Report Utilization

Estimated resources are compared with xc7vx485tffg1157-1.

Confidence Level: [High](#)

[Show More Details](#)

Register

Available:

Estimation: 

LUT

Available:

Estimation: 

[IO](#)

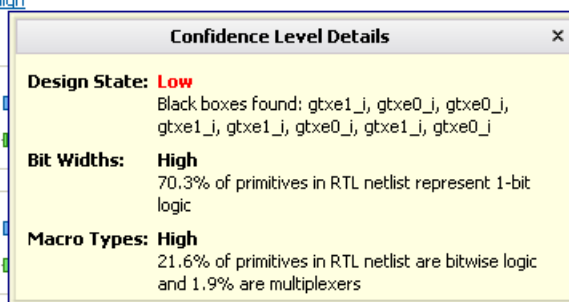
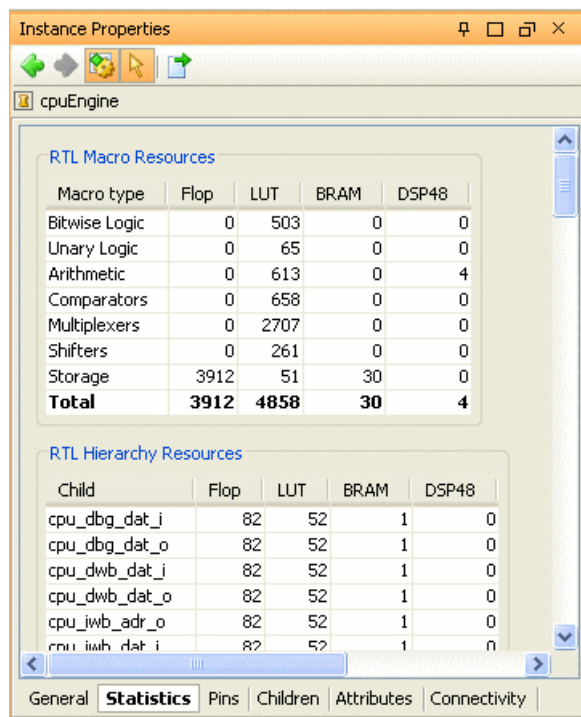


図 5-3 : [Resource Estimation] ビューに表示される Confidence Level

[Instance Properties] ビューでのリソース統計の解析

[Resource Utilization] ビューまたは [RTL Netlist] ビューでオブジェクトを選択すると、そのオブジェクトの [Instance Properties] ビューが開きます。[Instance Properties] ビューの [Statistics] タブには、選択されたオブジェクトまたは最上位モジュールのデバイス リソース予測が表示されます。207 ページの図 5-4 に、[Instance Properties] ビューの [Statistics] タブに表示されるリソース予測の例を示します。

ロジック リソースは、数値演算、コンパレータ、マルチプレクサー、ストレージなどに分類されています。メモリおよびプリミティブの表に、メモリ、深さ、ビット幅、ポート数、選択された階層でのビット幅別のマクロまたはプリミティブがリストされます。



The screenshot shows the 'Instance Properties' window for the 'cpuEngine' instance. It contains two tables: 'RTL Macro Resources' and 'RTL Hierarchy Resources'.

RTL Macro Resources

Macro type	Flop	LUT	BRAM	DSP48
Bitwise Logic	0	503	0	0
Unary Logic	0	65	0	0
Arithmetic	0	613	0	4
Comparators	0	658	0	0
Multiplexers	0	2707	0	0
Shifters	0	261	0	0
Storage	3912	51	30	0
Total	3912	4858	30	4

RTL Hierarchy Resources

Child	Flop	LUT	BRAM	DSP48
cpu_dbg_dat_i	82	52	1	0
cpu_dbg_dat_o	82	52	1	0
cpu_dwb_dat_i	82	52	1	0
cpu_dwb_dat_o	82	52	1	0
cpu_iwb_adr_o	82	52	1	0
cpu_iwb_dat_i	82	52	1	0

図 5-4 : RTL リソース予測の表示

リソース予測レポートをテキスト ファイル、XML ファイル、または CSV ファイルに保存するには、[Instance Properties] ビューで [Export Statistics] ツールバー ボタンをクリックします。

RTL ロジック階層の解析

PlanAhead には、デザインのロジック階層を表示する複数のビューがあります。

- [RTL Netlist] ビュー：展開可能なロジック ツリーを示します。
- [Hierarchy] ビュー：ロジック階層をグラフィカルに表示します。
- [Schematic] ビュー：回路図表示でロジックおよび階層を調べるのに使用します。

1 つのビュー選択したオブジェクトはほかのビューでも選択され、ロジック デザインを解析しやすくなっています。詳細は、[「\[Netlist\] ビュー」](#) および第 4 章の [「\[Hierarchy\] ビュー」](#) を参照してください。

エラボレート済みデザイン回路図の解析

[RTL Netlist] ビューで任意のロジック階層を選択し、[RTL Schematic] ビューで表示できます。

選択したロジックの [RTL Schematic] ビューを開くには、次のいずれかを実行します。

- [Tools] → [Schematic] をクリックします。
- [RTL Netlist] ビューを右クリックして [Schematic] をクリックします。

[RTL Schematic] ビューでの操作の詳細は、[第 4 章の「\[Schematic\] ビュー」](#) を参照してください。

デザインがエラボレートされたら、[Find] コマンドを使用してロジック オブジェクトを検索できます。詳細は、[202 ページの「\[Find\] コマンドを使用したソース ファイルの検索」](#) を参照してください。

RTL ソース ファイルの解析

[RTL Netlist] ビューでロジック エレメントを選択し、インスタンス化されている RTL ソース ファイルでそのオブジェクトのインスタンス化または定義を開くことができます。

選択したロジックのインスタンス化または定義を RTL ソース ファイルで開くには、オブジェクトを右クリックして [Go To Instantiation] をクリックします。ソース ファイルが開き、該当するインスタンスがハイライトされます。

消費電力予測

PlanAhead では、エラボレート済みデザインの消費電力を予測し、RTL プロジェクトの初期段階の電力分配を表示できます。デバイスの動作環境、I/O プロパティ、およびデザインのデフォルトのアクティビティ レートは制約を付けるか、または GUI から指定できます。

HDL コードから必要なデザイン リソースが予測され、デザインの各リソースのアクティビティの統計解析から消費電力が予測されます。[Power Estimation] ビューの出力は、XPower Estimator のサマリ シートと同様であり、結果を簡単に比較できます。

プロパティにはリソースごと、階層ごとの消費電力も追加されており、デザインの完全な電力分配が示されます。これらの情報は、詳細な解析および消費電力予測を満たすためのソリューションを見つけるのに役立ちます。

注記：RTL の消費電力予測機能は、Virtex®-5、Virtex-6、および Spartan®-6 デバイスでのみ使用可能です。

消費電力予測を実行するには、次の手順に従います。

- すべてのクロック ドメインが制約されていることを確認します。
タイミング制約は、次のいずれかの方法で指定できます。
 - 手動 (245 ページの「タイミング制約の定義」を参照)
 - [Tcl Console] ビューに「**create_clock**」と入力
 - UCF ファイルをインポート (101 ページの「プロジェクト設定」を参照)
- [Tcl Console] ビューに **set_operating_conditions** コマンドを入力し、FPGA の動作条件を設定します。

FPGA の動作条件は、デバイスの消費電力の合計に大きく影響します。

set_operating_conditions コマンドを使用して、システムの実際の動作環境を表すプロセス、電圧、冷却、およびその他の環境設定を入力します。

Tcl コマンドの詳細は、付録 E 「その他のリソース」に示す『PlanAhead Tcl コマンド リファレンス ガイド』(UG789) を参照してください。

3. 消費電力予測を実行するには、次のいずれかを実行します。

- [Tools] → [Report Power] をクリック
- エラボレート済みデザインを開いた状態で Flow Navigator で [Report Power] をクリック

図 5-5 に示す [Power Estimation] ダイアログ ボックスが開きます。

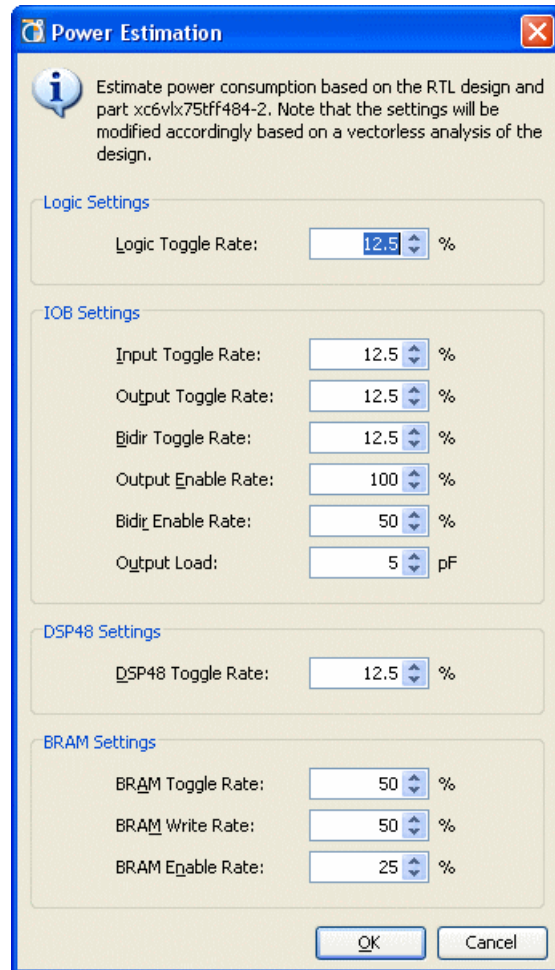


図 5-5 : [Power Estimation] ダイアログ ボックス

4. 必要に応じてトグル レートを変更し、[OK] をクリックします。

消費電力予測が実行され、ワークスペースに [Power Estimation] ビューが表示されます。消費電力を算出するため、RTL リソース予測およびユーザー制約が使用され、ネットリスト ノードに対してベクターなしの消費電力予測が実行されます。

注記：消費電力解析では、デフォルトのアクティビティ レートがシードとして使用されます。各エレメントの実際のアクティビティ レートは、シードと入力ファイル、ユーザーの入力、ベクターなしの予測エンジンの結果に基づいて算出されます。

210 ページの図 5-6 に、[Power Estimation] ビューを示します。

[Power Estimation] ビューには、消費電力サマリと、展開可能な電力消費グラフがリソース タイプおよびデザイン階層別に表示されます。ビューの右側の [Utilization Details] の下のロジック ツリーを展開し、デザイン階層を視覚化します。

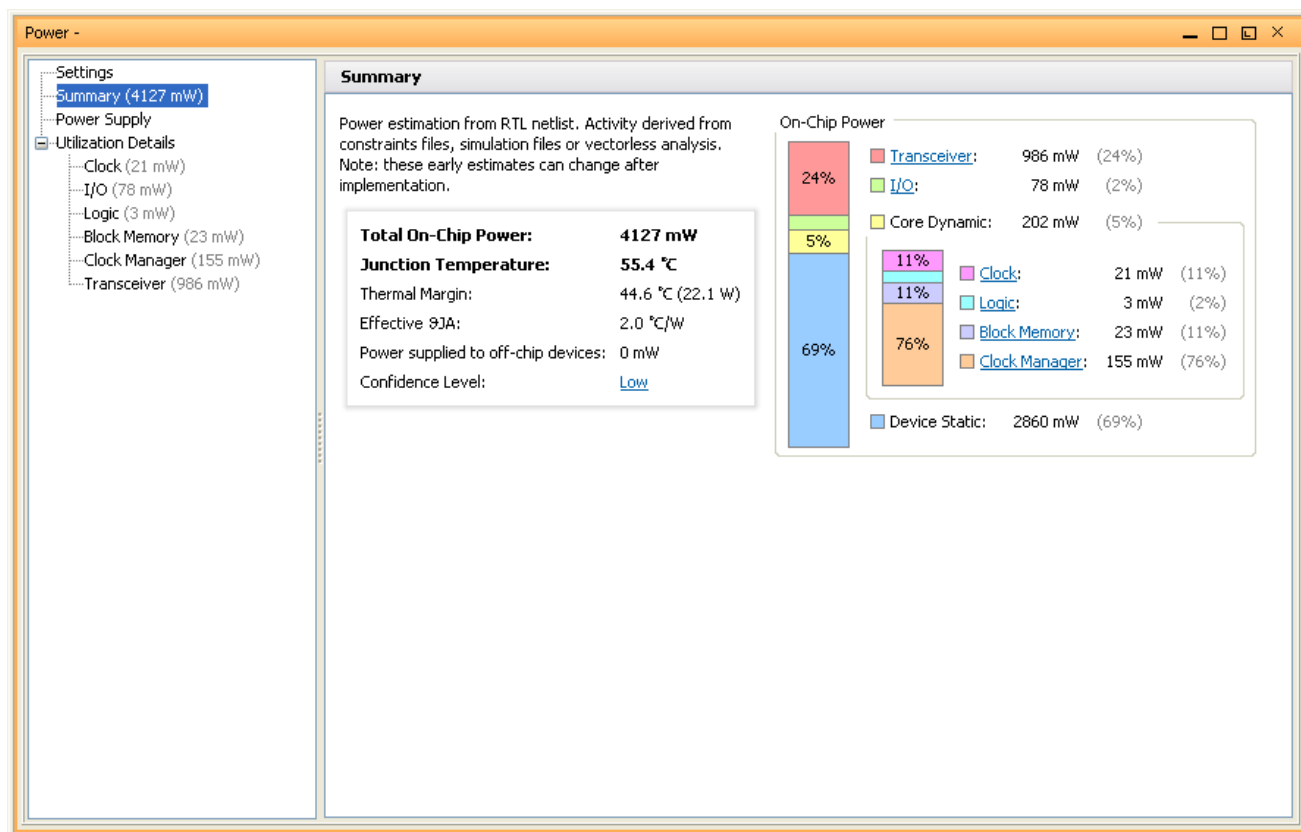


図 5-6 : [Power Estimation] ビュー

電力分配の解析

図 5-6 に示す [Power Estimation] ビューには、「Power Summary」セクションと「Power Utilization」セクションがあります。

- **Settings (設定)** : デバイス情報、動作状況、電力概算に使用されたトグル レートが表示されます。
- **Summary (サマリ)** : 電力分配のサマリと消費電力の算出に使用されたデバイスの動作条件のサマリを示します。
 - リンクをクリックすると、詳細の表示/非表示を切り替えることができます。パラメーターを変更するには、[208 ページの「消費電力予測」](#)の手順 2 および 3 を実行します。
 - **Confidence Level (信頼度)** : 結果がどれだけ正確かを示します。この情報で、消費電力予測を実行するために十分な入力データが供給されていることを確認します。[211 ページの表 5-1](#) に、Confidence Level の説明と値をカテゴリ別に表示します。

表 5-1 : 消費電力予測の Confidence Level

カテゴリ	Confidence Level	説明	スコア
デザイン 状態	Low	デザインに少なくとも 1 つブラック ボックスがあります。	0
	Medium	なし	0
	High	デザインにブラック ボックスはありません。	1
クロック ノード	Low	制約が設定されていないクロックが少なくとも 1 つあります。	0
	Medium	なし	0
	High	すべてのクロックが適切に制約されています。	2
I/O ノード	Low	10% を超える IOB にドライバー ロジックのトグル レートから算出されたイネーブルがあります。	0
	Medium	ドライバー ロジックのトグル レートから算出されたイネーブルを持つ IOB は 10% 以下で、少なくとも 1 つの I/O アクティビティ レートにデフォルト値が使用されています。	1
	High	ドライバー ロジックのトグル レートから算出されたイネーブルを持つ IOB は 10% 以下で、すべての I/O アクティビティ レートがユーザーにより供給されています。	2
内部 ノード	Low	50% 未満のインスタンスがトグル レート テンプレートと一致しています。	0
	Medium	50% 以上のインスタンスがトグル レート テンプレートと一致しており、少なくとも 1 つの内部レートにデフォルト値が使用されています。	1
	High	90% 以上のインスタンスがトグル レート テンプレートと一致しており、すべての内部アクティビティ レートがユーザーにより供給されています。	2
キャラクター ション データ	Low	Advance (アドバンス) 特性データ	0
	Medium	Preliminary (暫定) 特性データ	0
	High	製品特性データ	1
デザイン 全体	Low	すべてのカテゴリの合計スコアが 3 以下	
	Medium	すべてのカテゴリの合計スコアが 4、5、または 6	
	High	すべてのカテゴリの合計スコアが 7 以上	

- 表 5-2 に、Total On-Chip Power (合計オンチップ消費電力)、Junction Temperature (ジャンクション温度)、および Thermal Margin (熱マージン) の色凡例を示します。

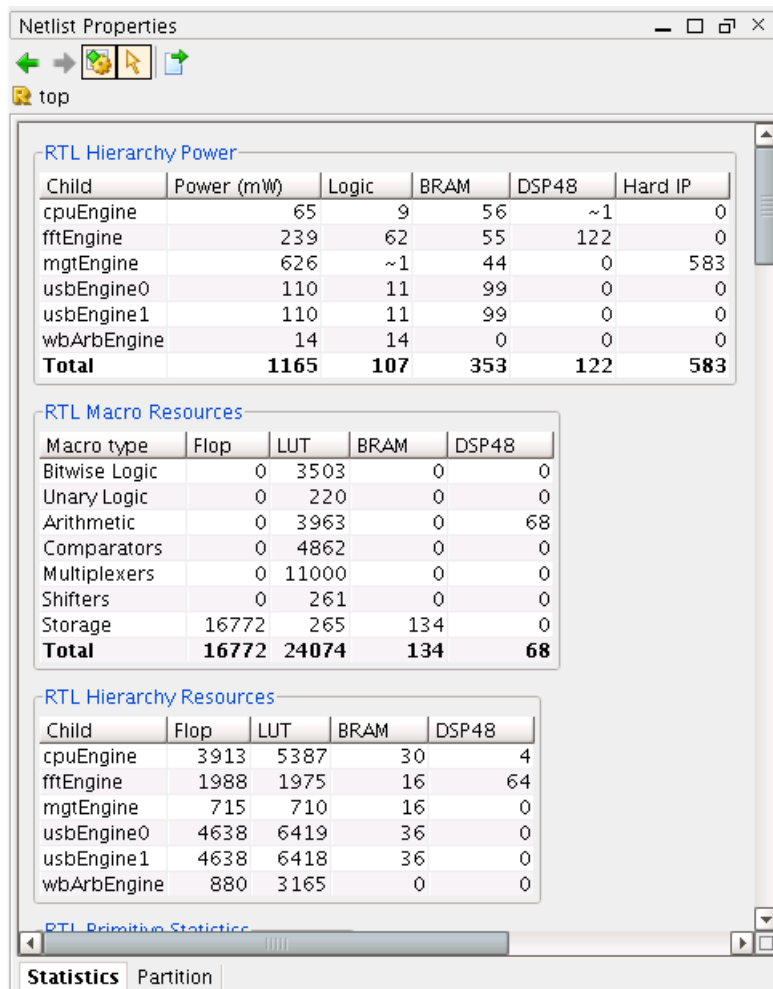
表 5-2：色凡例

テキストの色	説明
黒	ツールで判断されたデフォルト値
青	ユーザーが入力した値。このフィールドは、通常ツールで算出されます。
オレンジ	警告。リソースのコンフィギュレーションがサポートされないか、値がデバイスの通常の動作範囲外です。
赤	エラー。リソースのコンフィギュレーションまたはカウントが無効であるか、値がデバイスの最大動作範囲を超えています。

- **Power Utilization (電力使用率)**：電力分配をリソース別に示します。
 - 電力使用率バーを展開表示すると、デザイン階層ごとの電力消費を表示できます。
 - オブジェクトを選択すると、[RTL Netlist] ビューでもハイライトされ、[Properties] ビューが開きます。マウスをオブジェクトの上に置くと詳細が表示され、右クリックするとポップアップメニューに実行可能なコマンドが表示されます。

ネットリストおよびインスタンスの消費電力プロパティの表示

[RTL Netlist] ビューでデザイン階層の最上位を選択すると、[Netlist Properties] ビューが開きます。消費電力予測を実行すると、[Netlist Properties] ビューにさまざまな電力プロパティが表示されます。213 ページの図 5-7 に、[Netlist Properties] ビューに表示される [RTL Hierarchy Power] 表を示します。[RTL Hierarchy Power] 表には、最上位にインスタンス化されている各モジュールのリソース タイプごとの電力分配を示します。



Netlist Properties

RTL Hierarchy Power

Child	Power (mW)	Logic	BRAM	DSP48	Hard IP
cpuEngine	65	9	56	~1	0
fftEngine	239	62	55	122	0
mgtEngine	626	~1	44	0	583
usbEngine0	110	11	99	0	0
usbEngine1	110	11	99	0	0
wbArbEngine	14	14	0	0	0
Total	1165	107	353	122	583

RTL Macro Resources

Macro type	Flop	LUT	BRAM	DSP48
Bitwise Logic	0	3503	0	0
Unary Logic	0	220	0	0
Arithmetic	0	3963	0	68
Comparators	0	4862	0	0
Multiplexers	0	11000	0	0
Shifters	0	261	0	0
Storage	16772	265	134	0
Total	16772	24074	134	68

RTL Hierarchy Resources

Child	Flop	LUT	BRAM	DSP48
cpuEngine	3913	5387	30	4
fftEngine	1988	1975	16	64
mgtEngine	715	710	16	0
usbEngine0	4638	6419	36	0
usbEngine1	4638	6418	36	0
wbArbEngine	880	3165	0	0

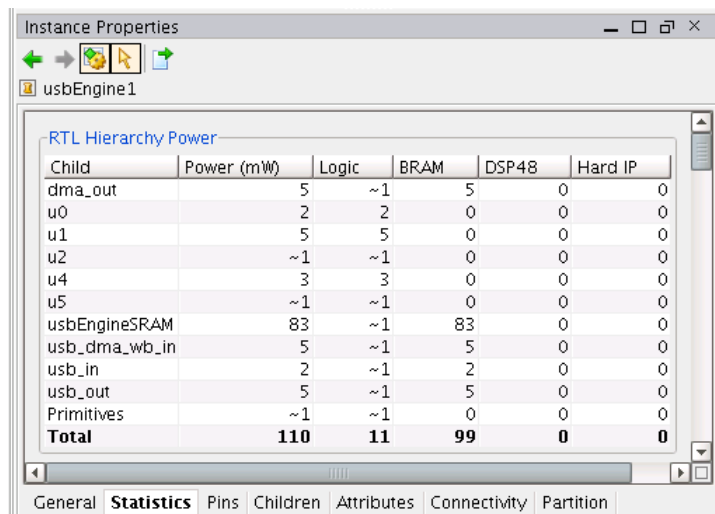
RTL Primitive Statistics

Statistics Partition

図 5-7 : [Netlist Properties] ビューの [RTL Hierarchy Power] 表

注記 : [Hard IP] 列には、クロック マネージャー、ギガビット I/O、PCIe®、EMAC など、その他のすべてのリソースが含まれます。

[RTL Netlist] ビューで階層インスタンスを選択すると、[Instance Properties] ビューが開きます。[Instance Properties] ビューの [Statistics] タブには、このレベルからインスタンス化されている各モジュールのリソース タイプごとの電力分配を示す [RTL Hierarchy Power] 表があります (214 ページの図 5-8 を参照)。



The screenshot shows the 'Instance Properties' window for 'usbEngine1'. The 'RTL Hierarchy Power' tab is active, displaying a table with columns: Child, Power (mW), Logic, BRAM, DSP48, and Hard IP. The table lists various components and their power consumption, with a 'Total' row at the bottom.

Child	Power (mW)	Logic	BRAM	DSP48	Hard IP
dma_out	5	~1	5	0	0
u0	2	2	0	0	0
u1	5	5	0	0	0
u2	~1	~1	0	0	0
u4	3	3	0	0	0
u5	~1	~1	0	0	0
usbEngineSRAM	83	~1	83	0	0
usb_dma_wb_in	5	~1	5	0	0
usb_in	2	~1	2	0	0
usb_out	5	~1	5	0	0
Primitives	~1	~1	0	0	0
Total	110	11	99	0	0

図 5-8 : [Instance Properties] ビューの [RTL Hierarchy Power] 表

プロジェクトの記録を残したり情報をチームメンバーと共有する場合は、[Netlist Properties] ビューのツールバーで [Export Statistics] ボタンをクリックします。または、別のツールで解析するためにデータを表形式でエクスポートするには、表内を右クリックして [Export to Spreadsheet] をクリックします。

[Tcl Console] ビューから **report_power** コマンドを使用すると、出力フォーマットをテキストまたは XML ファイルに指定できます。このコマンドのオプションの詳細は、付録 E 「その他のリソース」に示す『PlanAhead Tcl コマンド リファレンス ガイド』(UG789) を参照してください。

[Export Statistics] をクリックすると、[Export RTL Instance Statistics] ダイアログボックスが表示され、テキスト、XML、または CSV 形式を選択できます。

ビヘイビアー シミュレーションの実行

PlanAhead ツールは、ISE シミュレーション ツールである ISim と連動しています。ザイリンクスの ISim は、VHDL、Verilog、VHDL/Verilog 混合デザインのビヘイビアー シミュレーションおよびタイミング シミュレーションを実行するハードウェア記述言語 (HDL) シミュレータです。

PlanAhead ツールは、ビヘイビアまたはタイミング シミュレーション用に Mentor Graphics® ModelSim または Questa® Advanced Simulator ツールとも連動します。ターゲット シミュレータの設定方法については、第 3 章の「プロジェクト設定」を参照してください。

ビヘイビア シミュレーションを起動するには、次のいずれかを実行します。

- エラボレート済みデザインで [Tools] → [Simulation] → [Run Behavioral Simulation] をクリック
- Flow Navigator で [Project Manager] → [Run Behavioral Simulation] をクリック

PlanAhead で 215 ページの図 5-9 に示すような [Launch Behavioral Simulation] ダイアログボックスが表示されます。

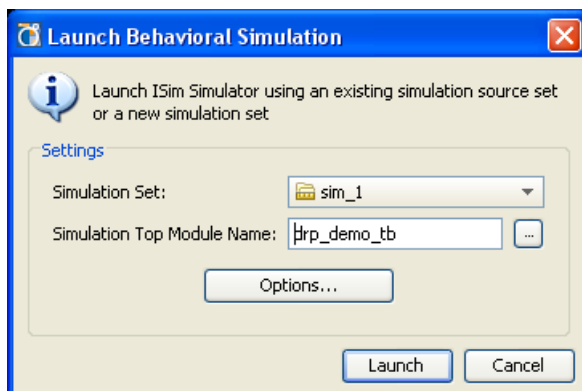


図 5-9 : [Launch Behavioral Simulation] ダイアログ ボックス

このダイアログ ボックスには、次のフィールドがあります。

- [Simulation Set] : シミュレーション run の名前を指定します。異なるデザイン階層および異なるオプションを使用して、複数のシミュレーション run を作成できます。
- [Simulation Top Module Name] : デザインの最上位モジュールを指定します。指定されているシミュレーション最上位モジュールが自動的に選択されますが、別のモジュールを最上位モジュールに指定して別の階層レベルからシミュレーションを実行したり、異なるバージョンのデザインをエラボレートできます。参照ボタンをクリックして、最上位モジュールを指定します。
- [Options] : 216 ページの図 5-10 および 217 ページの図 5-11 に示す [Simulation Options] ダイアログ ボックスを開きます。
- [Launch] : ISim でコンパイルおよびエラボレーションを実行し、ISim を GUI で開きます。
- [Cancel] : ISim を起動せずにダイアログ ボックスを閉じます。

[Options] ボタンをクリックすると、[Simulation Options] ダイアログ ボックスが開きます。このダイアログ ボックスには、[Launch Options] および [Language Options] という 2 つのタブがあります。次のセクションで、このダイアログ ボックスのオプションについて説明します。

シミュレーション オプションの設定

[Options] ボタンをクリックすると、[Simulation Options] ダイアログ ボックスが開きます。このダイアログ ボックスには、前に指定した最上位モジュールが含まれ、次の 2 つのオプション タブが表示されます。

[Compilation Options] : シミュレーション用のデザインを準備するために使用するコンパイラのコマンド ライン オプションを指定します。

[Simulation Options] : シミュレーション起動時に使用するランタイム指示子を指定します。

次のセクションで、このダイアログ ボックスのオプションについて説明します。

コンパイル オプション

[Compilation Options] ダイアログ ボックスの [Launch Options] タブでは、Fuse および ISim で実行される起動オプションを指定します。216 ページの図 5-10 に、[Simulation Options] ダイアログ ボックスを示します。

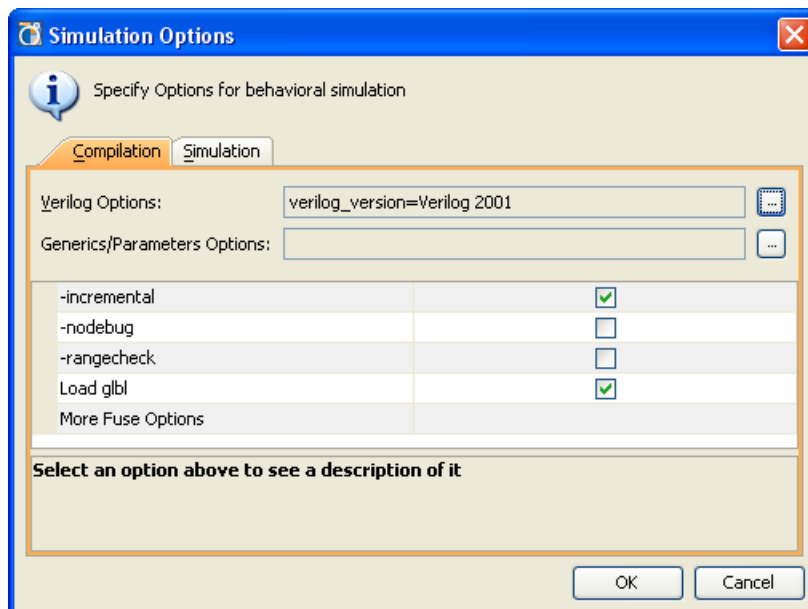


図 5-10 : [Simulation Options] ダイアログ ボックス

このダイアログ ボックスには、次のようなオプションがあります。

- [Verilog Options] : Verilog 検索パス、マクロ定義、大文字識別子 を指定します。詳細は、第 3 章の「プロジェクト設定」を参照してください。
- [Generics/Parameters Options] : VHDL ジェネリックまたは Verilog パラメーターの値を定義します。
- [-incremental] : 前回のコンパイルから変更されたファイルのみを fuse リンカーおよびコンパイラでコンパイルします。
- [-nodebug] : シミュレーション中に HDL コードをデバッグするための情報を含まないシミュレーション実行ファイル(.exe)を作成します。これにより、シミュレーションランタイムが短縮されます。
- [-rangecheck] : fuse リンカーおよびコンパイラでコンパイル中に VHDL 代入の値範囲をチェックします。このオプションは、VHDL コードにのみ適用されます。

注記 : これにより、ISim で常に実行されるアレイのインデックス範囲チェックに影響が出ることはありません。

- [Load gbl] : コンパイル中に gbl モジュールを読み込むかどうかを指定します。デザインに Verilog UniSim または SimPrim ライブラリが使用される場合は、このオプションをオンにする必要があります。
- [More Fuse Options] : fuse のその他のコマンド ライン オプションを指定します。コマンドは、1 つの文字列で値とセットにして入力する必要があります。たとえば、次のように入力します。

```
-maxdelay -init_file <filename> -notimingcheck
```

fuse オプションをコマンド ファイルに追加しておき、次のように **-f** コマンドを使用して [More Fuse Options] フィールドにそのファイルを指定することもできます。

```
-f <command_file>
```

シミュレーション オプション

図 5-11 は、[Simulation Options] ダイアログ ボックスの [Simulation] タブを示しています。

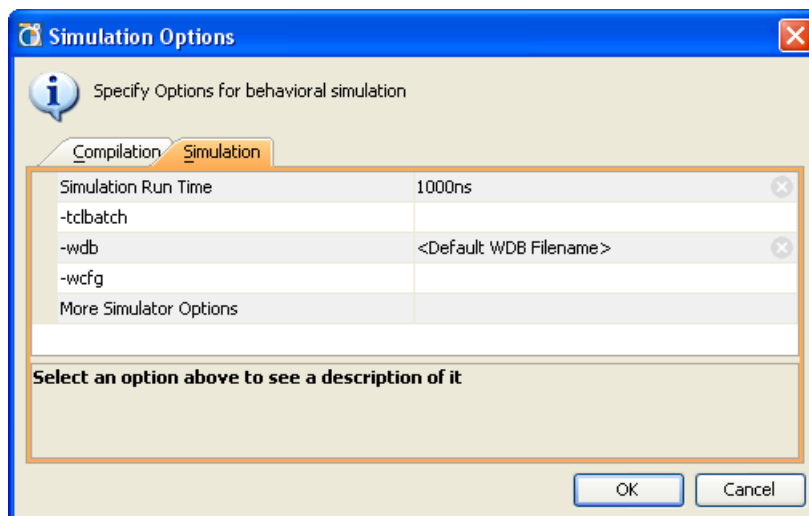


図 5-11 : [Simulation Options] ダイアログ ボックスの [Language Options] タブ

- [Simulation Run Time] : シミュレーション時間とその単位を指定します。指定できる時間の単位は、fs、ps、ns、us、ms、sec で、デフォルトは ps です。キーワードとして all を指定すると、シミュレーションするイベントがなくなるまで ISim が実行されます。
- [-tclbatch] : ランタイム時にシミュレーターで実行されるバッチ ファイルにリストされる Tcl コマンドのファイル名を指定します。指定したバッチ ファイルに含まれるコマンドは、完了するまで連続して実行されます。ISim では、バッチ ファイルの実行が終了するまで、コマンドプロンプトに入力されるコマンドが無視されます。

PlanAhead では tclbatch コマンドが使用され、isim.cmd というファイルで ISim に必須の 3 つのコマンドが渡されます。このファイルの内容は、次のとおりです。

```
onerror {resume}
wave add /
run <value>
```

起動時に実行するシミュレーターを制御する Tcl コマンドを作成する場合、これらの 3 つのコマンドが tclbatch ファイルに必ず含まれるようにしてください。onerror コマンドを最初に記述し、wave add と run を最後に記述することを推奨します。その他の ISim コマンドラインコマンドは、onerror と wave add の間に追加できます。

注記 : tclbatch コマンド ファイルの拡張子は、.tcl または .cmd のいずれかにしないと、ISim で正しく処理されません。

- [-wdb] : シミュレーション波形データを保存するファイル名を指定します。トレースされる信号のシミュレーション結果は、指定のファイル名で作業ディレクトリに保存されます。PlanAhead では、デフォルトで <top_module_name>.wdb ファイルが作成されます。
- [-wcfg] : ISim GUI に波形データを開く際に使用する波形コンフィギュレーション ファイルの名前を指定します。波形コンフィギュレーション ファイルでは、信号の順序、名前のスタイル、基数、色などの設定が指定されます。
- [More Simulator Options] : ISim のその他のコマンドライン オプションを指定します。これらのコマンドは、コマンド値と一緒に 1 行で入力します。たとえば、次のように入力します。
-log <filename> -transport_int_delays

ISim オプションをコマンド ファイルに追加しておき、次のように `-f` コマンドを使用して、[More Simulator Options] フィールドにそのファイルを指定することもできます。

```
-f <command_file>
```

ModelSim オプション

ターゲット シミュレーターに QuestaSim/ModelSim を指定した場合、コンパイルおよびシミュレーション オプションは異なります。次は、ModelSim のオプションです。

コンパイル

- [VHDL Syntax] : ソース ファイルおよびテストベンチで使用する VHDL 言語のバージョンを指定します。有効な値は、93 (1993) および 87 (1987) です。
- [Explicit Declarations] : `explicit` 関数宣言を使用するか、コンパイラーで自動的に作成された関数宣言を使用するか曖昧な場合に使用します。
- [More VLOG Options] : ツールで使用するその他の Verilog コンパイラー (`vlog`) コマンド ライン オプションを指定します。
- [More VCOM Options] : ツールで使用するその他の VHDL コンパイラー (`vcom`) コマンド ライン オプションを指定します。

シミュレーション

- [Simulation Run Time] : シミュレーション時間とその単位を指定します。
- [Log All Signals] : シミュレーション中の信号値をすべて保存して、未知の信号ステートのソースをトレースしやすくします。
- [More VSIM Options] : ツールで使用するその他の ModelSim (`vsim`) コマンド ライン オプションを指定します。

シミュレーターの起動

ISim でビヘイビアー シミュレーションを起動するには、[Launch Behavioral Simulation] ダイアログ ボックスで [Launch] をクリックします。ISim のオブジェクト コンパイラおよびリンカーであるフューズが実行され、Verilog および VHDL コードがコンパイルおよびエラボレートされます。

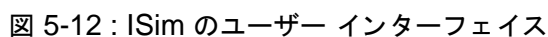
コンパイルされたオブジェクト コードは、シミュレーション実行ファイルにリンクされます。シミュレーション実行ファイルの名前は、[Launch Behavioral Simulation] ダイアログ ボックスで指定した最上位モジュールの名前が使用されます。ISim の実行ファイルが完了すると、シミュレーターが起動します。次に、コマンド実行の例を示します。

```
INFO:[Runs-8] Fuse completed.
INFO:[Runs-10] Launching ISim...
INFO:[Runs-11] Running '"C:/project_cpu_hdl/project_cpu_hdl.sim/sim_1/
top.exe"
-intstyle pa -gui -tclbatch ISim.cmd
-wdb "wdb_test1.wdb" -view "wcfg_test1.wcfg"'
```

シミュレーション実行ファイルは、[Simulation Options] ダイアログ ボックスの [Launch Options] タブで指定したオプションを使用して実行されます。

ISim は `-gui` オプションを使用して実行され、デザインを対話的にシミュレーションできる ISim ユーザー インターフェイスが開きます。GUI から ISim を実行する方法については、[付録 E「その他のリソース」](#) に示す『ISim ユーザー ガイド』(UG660) を参照してください。

219 ページの [図 5-12](#) に、ISim の GUI を示します。



RTL DRC の実行

このセクションでは、PlanAhead で DRC ルールを選択して DRC 違反を解析する方法について説明します。

DRC の選択

エラボレート済みデザインのデザイン ルール チェック (DRC) を実行できます。これらの DRC は、消費電力の削減およびパフォーマンスの向上に焦点を当てています。

1. デザインをエラボレートした後に DRC チェックを実行するには、次のいずれかを実行します。

- [Tools] → [Run DRC] をクリックします。
- Flow Navigator で [RTL Analysis] → [Report DRC] をクリックします。

図 5-13 に示す [Run DRC] ダイアログ ボックスが開き、実行するチェックを選択できます。

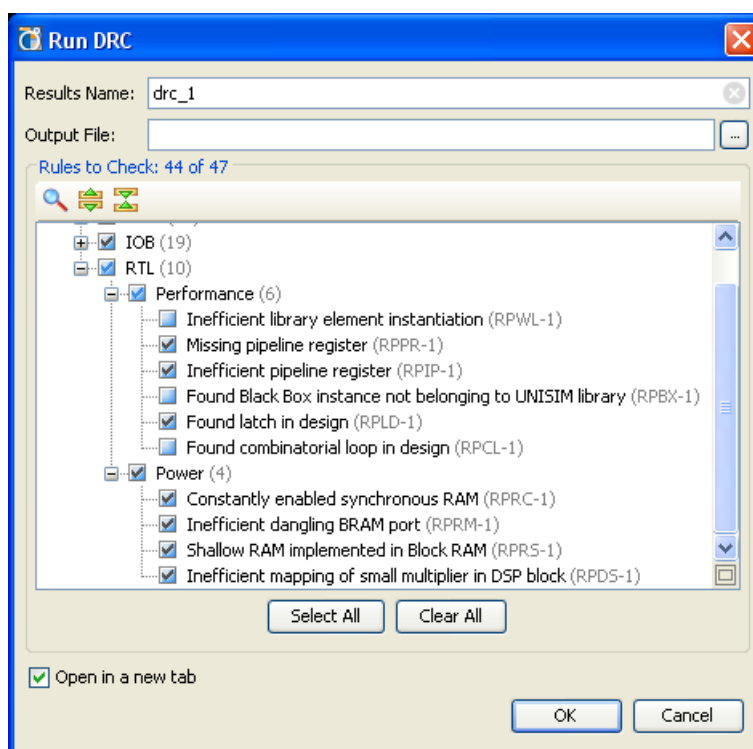


図 5-13 : [Run DRC] ダイアログ ボックス

2. [Run DRC] ダイアログ ボックスで実行するチェックを選択し、[OK] をクリックします。

チェックの詳細は、付録 B 「PlanAhead の DRC」を参照してください。

DRC 違反の解析

DRC で違反が検出された場合、221 ページの図 5-14 に示す [DRC Results] ビューが表示されます。[DRC Results] ビューには、検出された違反が [Run DRC] ダイアログ ボックスのカテゴリごとに表示されます。

違反には、その重要度も示されます。違反には、問題が発生する可能性があることを示す情報、解決が必要だと思われる問題を示す警告、デザインの正しいインプリメンテーションを阻害する問題を示すエラーがあります。

[DRC Results] ビューでは違反が色分けされており、エラー、警告、情報メッセージを確認しやすくなっています。

- エラーは赤いアイコンで示されます。
- 警告はオレンジ色のアイコンで示されます。
- 情報は黄色のアイコンで示されます。

ツールバーの [Hide Warnings and Informational Messages] ボタンをクリックすると、警告および情報メッセージを非表示にし、エラーメッセージのみを表示できます。



また、[Severity] 列のヘッダーをクリックすると、違反の重要度で並べ替えることができます。

- 列ヘッダーを 1 回クリックすると昇順で並べ替えられます。
- もう 1 回クリックすると降順で並べ替えられます。

詳細は、第 4 章の「ツリー表形式のビュー」を参照してください。

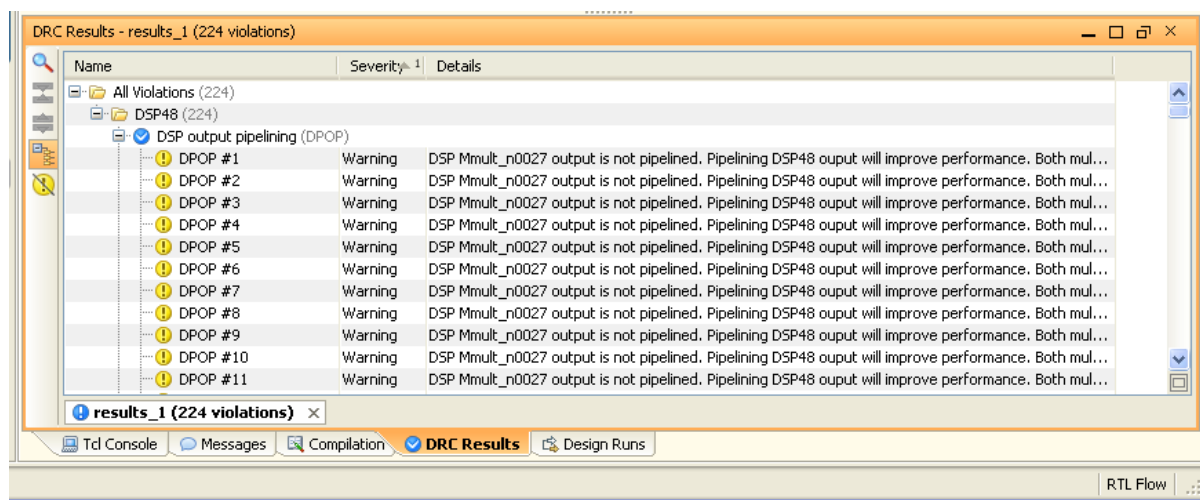
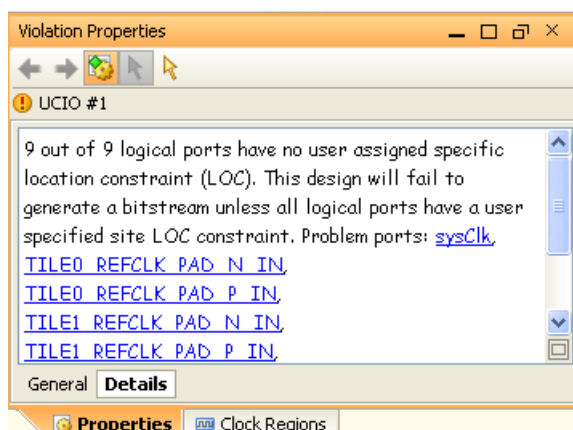


図 5-14 : DRC 違反を含むエラボーレート済みデザイン

[DRC Results] ビューで違反メッセージを右クリックし、[Violation Properties] をクリックすると、[Violation Properties] ビューが表示されます。このビューで、DRC 違反の概要と、ルールに違反しているデザイン エLEMENT の詳細を確認できます。

[Violations Properties] ビューの [Details] タブには、DRC に違反するデザイン オブジェクトへのリンクがあります。リンクをクリックすると、そのデザイン オブジェクトが [RTL Netlist] ビュー、[Device] ビュー、[Schematic] ビュー、およびソース RTL ファイルで選択されます。



デザインの合成

PlanAhead™ では、合成およびインプリメンテーション **run** をボタンをクリックするだけで実行可能な環境が提供されています。**run** のデータは自動的に管理され、さまざまな RTL ソース バージョン、合成およびインプリメンテーション オプション、制約を使用して繰り返し実行できます。

また、異なるコマンド オプション、ターゲット デバイス、タイミング制約、および物理制約を使用した複数の合成およびインプリメンテーション **run** の実行も可能です。複数の合成およびインプリメンテーション **run** は、順次起動するか、マルチプロセッサ マシンで同時に起動できます。PlanAhead では、合成 **run** に Xilinx® Synthesis Technology (XST) が使用されます。

ストラテジとは、合成またはインプリメンテーションの **run** に適用されるコマンド オプションの設定で、作成して保存しておくことができます。詳細は、[第 4 章の「合成およびインプリメンテーション ストラテジの定義」](#)を参照してください。

ユーザーは、合成またはインプリメンテーションの進捗状況を監視し、ログ レポートを確認し、**run** をキャンセルできます。

合成手法

次に、PlanAhead をデザインの解析、フロアプラン、階層デザインに最大限に活用するためのロジック合成に関する推奨事項を示します。

- モジュールの出力にレジスタを付け、クリティカルパスに関連するモジュールの数を制限します。
注記：1 つの大型階層ブロックに長いパスがあると、フロアプランが困難になります。RTL で大型階層ブロックを分割してみてください。
- PlanAhead では、エラボレート済みデザインでブロック レベルのパーティションを定義できます。クリティカル タイミング パスが 1 つのモジュール内に制限されるよう、できる限り RTL レベルでデザインを分割します。パーティションを定義すると XST がインクリメンタル フローで実行され、各パーティションに対して個別の NGC ファイルが生成されます。詳細は、[第 13 章「階層デザイン手法の使用」](#)を参照してください。
注記：クリティカル パスが多数の階層モジュールにまたがっていると、フロアプランやパーティションの設定が困難です。
- デザインの変更頻度が高い場合は、インクリメンタル合成を考慮します。ほとんどの合成ツールでは、トップダウンのインクリメンタル合成およびインプリメンテーションが可能です。この機能とザイリンクスのパーティション機能を組み合わせて、インプリメント済みデザインで変更されていないモジュールを保持できます。詳細は、[第 13 章「階層デザイン手法の使用」](#)を参照してください。

デザイン保持は、インクリメンタルフローには役立ちますが、階層をまたがるグローバル最適化が不可能であるため、パフォーマンスが低下することがあります。インクリメンタル手法を試す前に、このトレードオフを考慮してください。

- 合成エンジンで階層が再構築されるように設定するか、合成ネットリストの階層を保持します。フラット化されたネットリストは合成の面からは最適ですが、フロアプランおよび配置制約が困難になります。

-netlist_hierarchy=rebuilt オプションを使用すると、XST のロジック最適化でネットリストがフラット化されますが、その後ロジック名に基づいて合成済みデザインのネットリスト階層が再構築されます。これにより、PlanAhead でデザイン解析およびフロアプランがしやすくなります。[XST Defaults] 以外の PlanAhead XST 合成ストラテジでは、-netlist_hierarchy=rebuilt が設定されています。

XST でこのオプションを使用して問題が発生した場合は、[Project Settings] ダイアログ ボックスの [Synthesis] ページで [-netlist_hierarchy] オプションを [as_optimized] に戻してください (230 ページの図 6-6)。

合成結果の最適化方法については、付録 E「その他のリソース」に示す次の資料を参照してください。

- 『合成/シミュレーション デザイン ガイド』(UG626)
- 『XST ユーザー ガイド (Virtex-4、Virtex-5、Spartan-3、CPLD デバイス)』(UG627)
- 『XST ユーザー ガイド (Virtex-6、Spartan-6、および 7 シリーズ デバイス)』(UG687)

ファイルのコンパイル順

RTL ソース ファイルは、合成ツールで [Sources] ビューの [Compile Order] タブに表示される順序でコンパイルされます。リストの一番上のファイルが最初にコンパイルされ、一番下のファイルが最後にコンパイルされます。コンパイル順は、宣言のある RTL ソース ファイルをほかのソース ファイルよりも先にコンパイルする必要がある場合などに指定します。

PlanAhead では、最上位モジュールに最適なモジュールが自動的に認識されて設定されます。アクティブな階層の下用最上位モジュール ファイルおよびすべてのソース ファイルが正しい順序で合成およびシミュレーションに使用されるように、コンパイル順序も自動的に管理されます。

[Sources] ビューの [Hierarchy Update] コマンドを使用すると、最上位モジュールへの変更、デザインのソース ファイルへの変更などの処理方法が指定できます。デフォルト設定の [Automatic Update and Compile Order] では、ソース ファイルが変更されたら、デザインの階層表示とコンパイル順が自動的にアップデートされます。

合成の前にコンパイル順を変更するには、次のいずれかの操作を実行します。

1. [Hierarchy Update] → [Automatic Update, Manual Compile Order] をクリックすると、デザインに最適な最上位モジュールが自動的に決定はされますが、手動でコンパイル順を指定できます。
2. [Sources] ビューの [Compile Order] タブにファイルをドラッグ アンド ドロップしてコンパイル順を変更するか、
3. [Sources] ビューのポップアップ メニューにある [Move Up]、[Move Down]、[Move to Top]、または [Move to Down] を使用してファイルを移動します。

詳細は、第 4 章の「[Sources] ビュー」を参照してください。

グローバル インクルード ファイルの定義

Verilog および Verilog ヘッダー ファイルは、グローバル インクルード ファイルとしても定義できます。1 つまたは複数の Verilog または Verilog ヘッダー ファイルをグローバル インクルード ファイルとして指定できます。最上位モジュールの指定、インクルード ファイルのスキャン、ソースの

順序の自動変更、RTL エラボレーション、合成、およびシミュレーションでは、グローバル インクルードとして指定されているファイルが最初に処理されます。

グローバル インクルード ファイルは常にデザインのほかのファイルより先に処理されるので、共通 Verilog ヘッダー ファイルとデザインのほかのソースを指定するのが簡単になります。

Verilog では通常、別の Verilog ファイルやヘッダー ファイルからの内容を参照する Verilog ソース ファイルの冒頭に `include 文を含める必要があります。共通ヘッダー ファイルを使用するデザインでは、複数の Verilog ソースにそれぞれ複数の `include 文を含める必要がある場合もあります。グローバル インクルード ファイルはほかのソース ファイルより先に処理されるので、ヘッダー ファイルをグローバル インクルード ファイルとして指定することにより、`include 文を繰り返し指定する必要はなくなります。

Verilog ファイルまたは Verilog ヘッダー ファイルをグローバル インクルード ファイルとして指定するには、次のいずれかを実行します。

- [Sources] ビューでファイルを右クリックし、[Set Global Include] をクリックします。
- [Source File Properties] ビューで [Global Include] チェック ボックスをオンにします (図 6-1)。

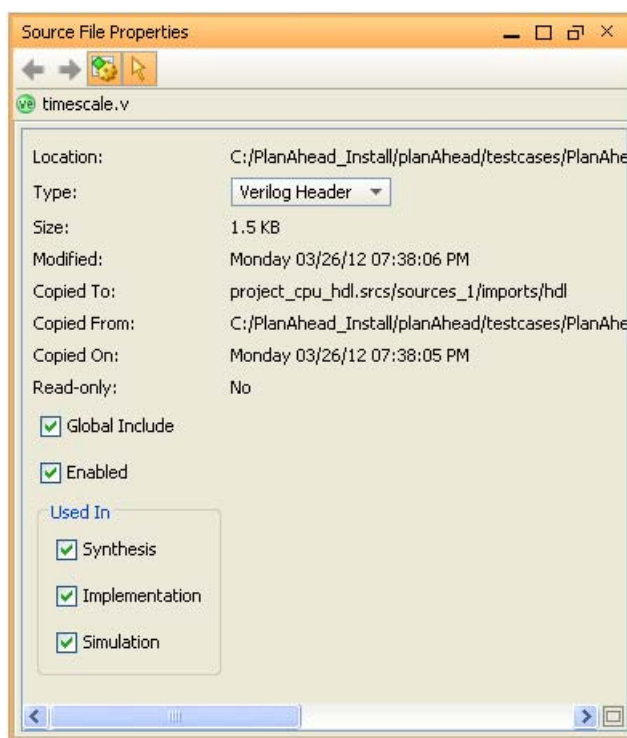


図 6-1 : [Global Include] チェック ボックス

詳細は、「[Sources] ビュー」および第 4 章の「[Source File Properties] ビュー」を参照してください。

ファイルをグローバル インクルード として指定すると、ほかのファイルとは異なるアイコンが付いて、[Sources] ビューの [Global Include] フォルダーに表示されます。複数のファイルをグローバル インクルード ファイルとして指定できます。この場合、[Sources] ビューに表示されている順序で処理されます。すべてのグローバル インクルード ファイルがプロジェクトのほかのソース ファイルより先に処理されます。

グローバル インクルード ファイルの内容はほかのすべてのソース ファイルで参照されるので、1 つの Verilog ソースのみに適用する必要がある Verilog ヘッダー ファイル (特定の `define マクロ

など) は、グローバル インクルード ファイルとして指定するのではなく `include 文を指定して参照する必要があります。

合成の実行

ザイリンクス ターゲット パーツ用にデザインを合成するには、RTL ソース ファイルとデザイン制約を次のプロセスを使用して XST で実行する必要があります。

- 「合成 run の定義」
- 「合成オプションの設定」
- 「合成の実行」

合成 run の定義

合成 run では、合成中に使用されるデザインの詳細が定義および設定できます。合成 run では、合成中にターゲットとされるザイリンクス デバイス、使用する制約セット、合成ツールの結果を制御するコマンド ライン オプションを定義します。

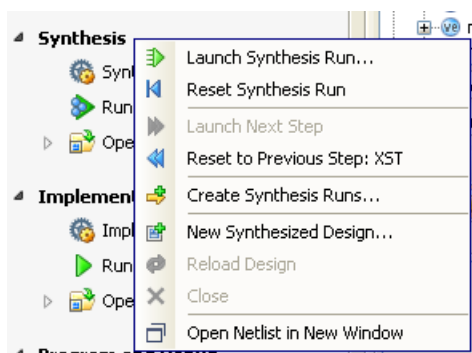
合成 run は次のように定義できます。

- [Flow] → [Create Runs] をクリックします。
- Flow Navigator の [Synthesis] ポップアップ メニューから [Create Synthesis Runs] をクリックします。

Create New Runs ウィザードが開きます。ウィザードの最初のページは、コマンドのサマリです。

4. [Next] をクリックします。

227 ページの図 6-2 に示すような [Set Up Synthesis Runs] ダイアログ ボックスが開きます。



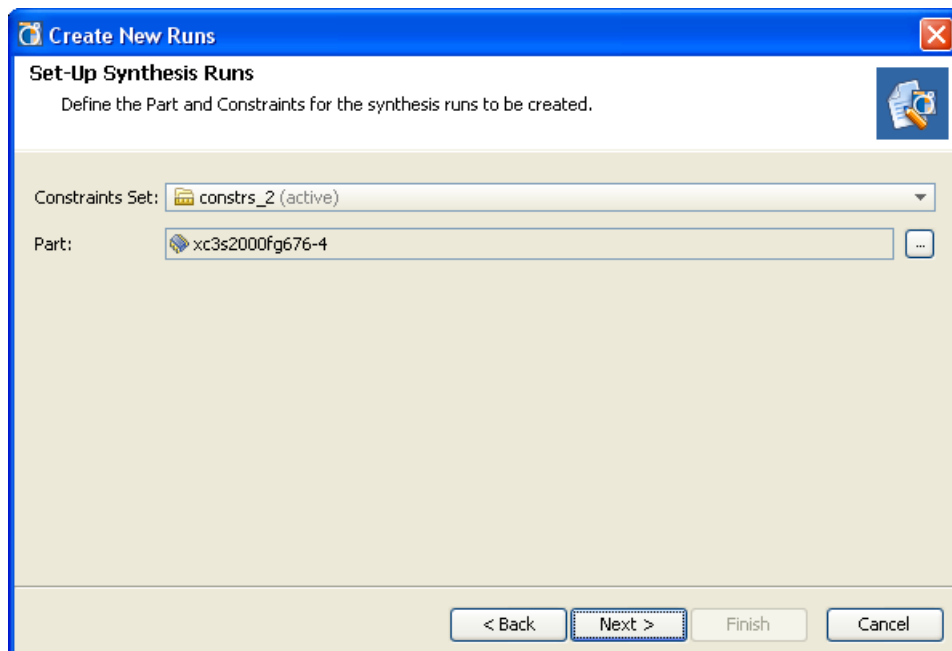


図 6-2 : run の新規作成コマンド : [Set Up Synthesis Runs] ページ

制約セットとターゲット パーツを選択します。

注記：デフォルト値は、[Create New Runs] コマンド実行時の合成またはインプリメンテーションのプロジェクト設定で定義されます。詳細は、101 ページの「プロジェクト設定」を参照してください。

5. [Next] をクリックします。

227 ページの図 6-3 に示す [Choose Synthesis Strategies] ダイアログ ボックスが開きます。

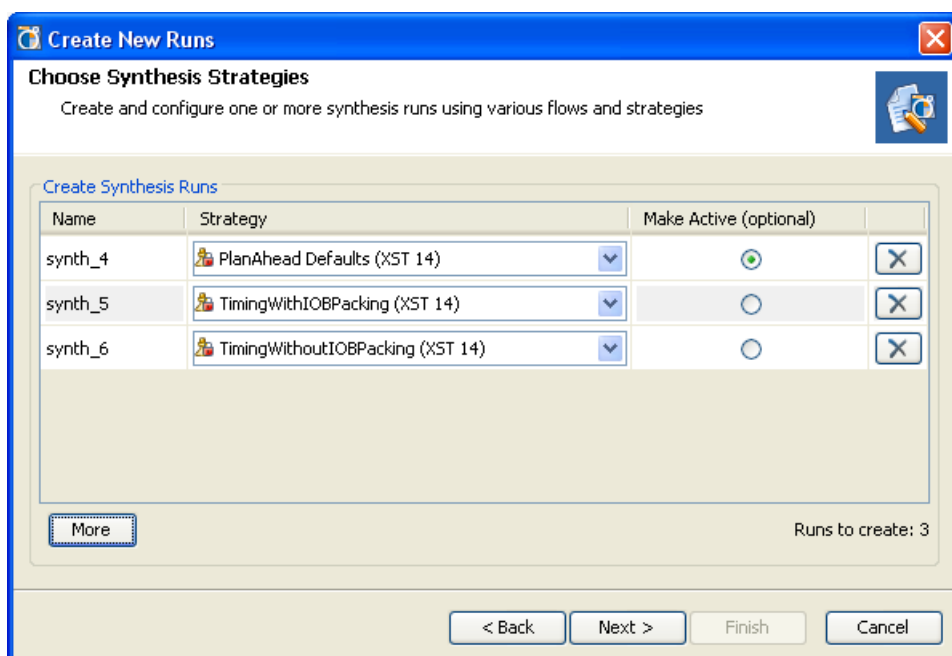


図 6-3 : [Choose Synthesis Strategies] ページ

6. run の名前を入力するか、デフォルトのままにします。
7. 新しい run のストラテジを選択します。ストラテジは、合成を制御する XST のランタイム オプションを定義した設定のことです。詳細は、第 4 章の「[合成およびインプリメンテーションストラテジの定義](#)」を参照してください。
8. [Make Active] を選択し、新しい run をアクティブ run にします。新規 run を複数作成する場合、アクティブにできる run は 1 つだけです。
9. [More] ボタンをクリックし、追加の run を定義します。追加 run の名前とストラテジを指定します (227 ページの図 6-3)。
10. [Next] をクリックします。

228 ページの図 6-4 に示す [Launch Options] ダイアログ ボックスが開きます。

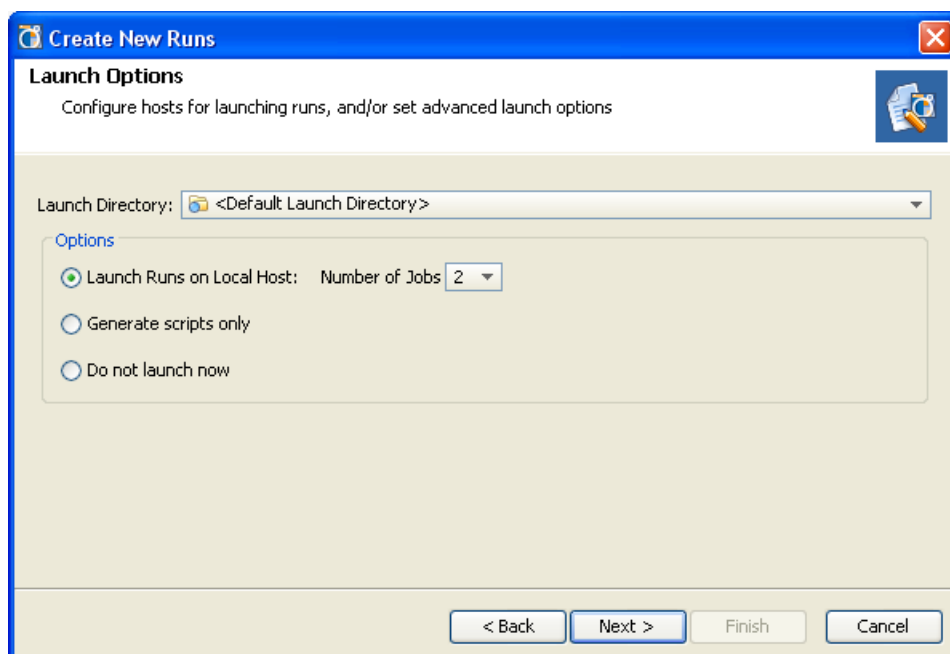


図 6-4 : [Launch Options] ダイアログ ボックス

11. 合成 run のデータを作成して保存するディレクトリを [Launch Directory] に指定します。
デフォルトの起動ディレクトリは、ローカルプロジェクト ディレクトリ構造に含まれます。
合成 run のファイルは、次のディレクトリに保存されます。

`<project_name>/<project_name>.runs/<run_name>`

注記：プロジェクト ファイルには絶対パスが記述されるので、プロジェクト ディレクトリ外のデフォルト以外の場所を指定すると、プロジェクトを移動しにくくなります。

12. 次のオプションを設定します。
 - [Launch Runs on Local Host] : ローカル マシンのプロセッサで run を起動します。
 - [Number of Jobs] : run に使用するローカル プロセッサの数を指定します。このオプションは、複数の run を同時に起動する場合にのみ使用されます。各 run が各プロセッサで起動されます。このオプションでは、マルチスレッド プロセッサは使用されません。

- [Launch Runs on Remote Hosts] (Linux のみ) : ジョブを起動するのにリモート ホストを使用します。詳細は、第 9 章の「リモート Linux ホストでの run の起動」を参照してください。
- [Configure Hosts] : リモート ホストを設定します。
- [Generate scripts only] : run ディレクトリおよび run スクリプトをエクスポートおよび作成しますが、run は起動しません。スクリプトは、PlanAhead 環境外で後で実行できます。
- [Do not launch now] : 新しい run を保存はしますが、現時点では run スクリプトを開始または作成しません。

13. [Next] をクリックし、[Create New Runs Summary] を確認します。

14. [Finish] をクリックすると、定義した run が作成され、指定の起動オプションが実行されます。

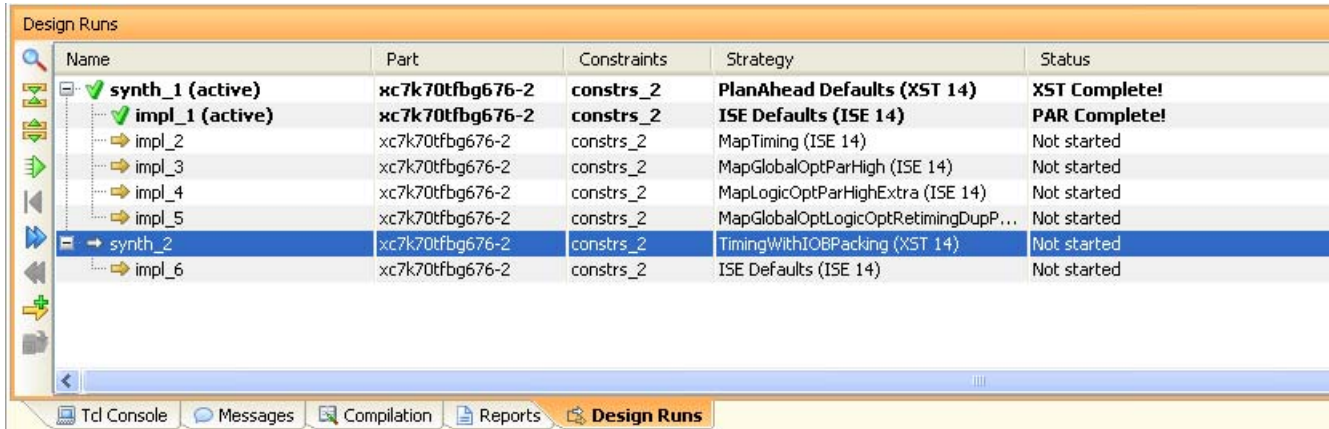
新しい run が [Design Runs] ビューに追加されます。

[Design Runs] ビューの使用

[Design Runs] ビューには、プロジェクトで作成された合成 run とインプリメンテーション run のすべてが表示され、それらを設定、管理、開始するためのコマンドも表示されます。

[Design Runs] ビューが表示されていない場合は、[Window] → [Design Runs] をクリックして [Design Runs] ビューを表示します。図 6-5 に、[Design Runs] ビューを示します。

合成 run には、インプリメンテーション run を複数含めることができます。プラス記号 (+) やマイナス記号 (-) をクリックすると、合成 run のツリー表示を展開したり、閉じたりできます。[Design Runs] ビューは、ツリー形式の表示です。このビューを列ごとに分類する方法については、127 ページの「ツリー表形式のビュー」を参照してください。



Name	Part	Constraints	Strategy	Status
synth_1 (active)	xc7k70tfbg676-2	constrs_2	PlanAhead Defaults (XST 14)	XST Complete!
impl_1 (active)	xc7k70tfbg676-2	constrs_2	ISE Defaults (ISE 14)	PAR Complete!
impl_2	xc7k70tfbg676-2	constrs_2	MapTiming (ISE 14)	Not started
impl_3	xc7k70tfbg676-2	constrs_2	MapGlobalOptParHigh (ISE 14)	Not started
impl_4	xc7k70tfbg676-2	constrs_2	MapLogicOptParHighExtra (ISE 14)	Not started
impl_5	xc7k70tfbg676-2	constrs_2	MapGlobalOptLogicOptRetimingDupP...	Not started
synth_2	xc7k70tfbg676-2	constrs_2	TimingWithIOBPacking (XST 14)	Not started
impl_6	xc7k70tfbg676-2	constrs_2	ISE Defaults (ISE 14)	Not started

図 6-5 : [Design Runs] ビュー

[Design Runs] ビューでは、run が開始されなかった場合も含めて、run のステータスが進行中か、終了したか、最新の状態でないことがレポートされます。ソース ファイル、制約、またはプロジェクト設定を変更すると、run は最新の状態ではなくなります。[Design Runs] ビューでは、run をリセットしたり、古い run のデータを削除したりできます。

アクティブ run の設定

PlanAhead でアクティブにできる合成 run とインプリメンテーション run はそれぞれ 1 つだけです。[Compilation] ビューおよび [Messages] ビュー、ステータス バー、[Project Summary] にはこ

のアクティブな run の情報が表示されます。[Project Summary] には、コンパイル リソースおよびこのアクティブな run のサマリ情報が表示されます。

run をアクティブにするには、[Design Runs] ビューでその run を右クリックし、ポップアップ メニューから [Make Active] コマンドをクリックします。

合成オプションの設定

合成 run のさまざまな設定は開始前に変更できます。

[Design Runs] ビューで run を選択すると、[Run Properties] ビューでその run の現在の設定が表示されます。[Run Properties] ビューでは、run の名前、run でターゲットにするザイリンクス パーツ、run の詳細記述、合成で使用され、新しい制約のレポジトリになる制約ファイルなどを変更できます。詳細は、第 4 章の「[Run Properties] ビューの使用」を参照してください。

合成中に XST で使用されるランタイム オプションも [Design Runs] ビューのポップアップ メニューから [Change Run Settings] をクリックして表示される [Design Run Settings] ダイアログ ボックスから変更できます (230 ページの図 6-6)。

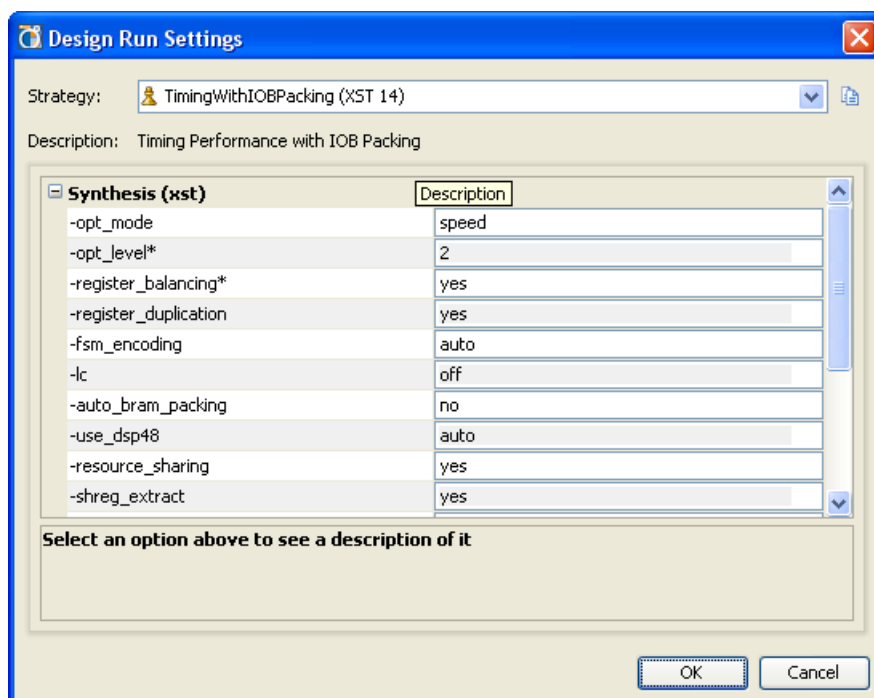


図 6-6 : [Design Run Settings] ダイアログ ボックス

[Design Run Settings] ダイアログ ボックスでは、次を指定できます。

- [Strategy] : run で使用される一般的なストラテジを指定できます。
- [XST options] : 一般的な run ストラテジを改善して現在のデザインのニーズを満たすために、XST コマンド ライン オプションを変更できます。

オプションを選択すると、簡単な説明と目的がページの下部に表示されます。オプション名の横にアスタリスク (*) が付いている場合は、その値がデフォルト以外の値に設定されていることを示します。

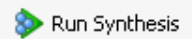
特定の XST オプションについては、付録 E 「その他のリソース」 に示す次の資料を参照してください。

- 『XST ユーザー ガイド (Virtex-4、Virtex-5、Spartan-3、CPLD デバイス)』(UG627)
- 『XST ユーザー ガイド (Virtex-6、Spartan-6、および 7 シリーズ デバイス)』(UG687)
- [Save Strategy As] : ストラテジへの変更を別のプロジェクトで使用できるように新しいストラテジとして保存します。

合成の実行

合成は、次のいずれかの方法で開始できます。

- Flow Navigator の [Run Synthesis] をクリックします。
- メイン メニューから [Flow] → [Run Synthesis] をクリックします。
- ツールバー メニューで [Run Synthesis] をクリックします。



注記：アクティブ run が [Design Runs] ビューで開始されます。詳細は、[229 ページの「アクティブ run の設定」](#)を参照してください。

アクティブ run 以外の run を実行したり、複数 run を同時に実行したりもできます。[Design Runs] ビューで 1 つまたは複数の run を選択します。複数選択する場合は、Shift キーまたは Ctrl キーを押しながらクリックします。右クリックまたは [Design Runs] ビューのツールバー メニューから [Launch Runs] コマンドをクリックして [Launch Selected Runs] ダイアログ ボックスを表示します (図 6-7)。

注記：[Design Runs] ビューで複数 run を選択している場合は、合成 run とインプリメンテーション run の両方を選択できます。PlanAhead では、run の依存度が管理され、run が正しい順序で実行されます。

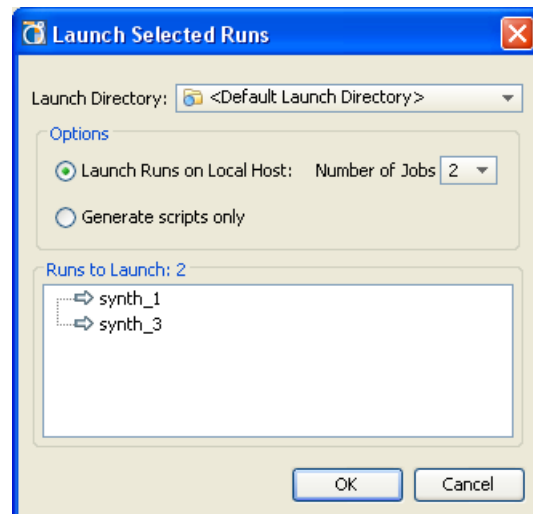


図 6-7 : [Launch Selected Runs] ダイアログ ボックス

- [Launch Directory] : デフォルトの起動ディレクトリは、ローカルプロジェクト ディレクトリ構造に含まれます。合成 run のファイルは、次のディレクトリに保存されます。

`<project_name>/<project_name>.runs/<run_name>`

注記：プロジェクト ファイルには絶対パスが記述されるので、プロジェクト ディレクトリ外のデフォルト以外の場所を指定すると、プロジェクトを移動しにくくなります。

- オプション
 - [Launch Runs on Local Host]：ローカル マシンのプロセッサで run を起動します。
 - [Number of Jobs]：run に使用するローカル プロセッサの数を指定します。このオプションは、複数の run を同時に起動する場合にのみ使用されます。各 run が各プロセッサで起動されます。このオプションでは、マルチスレッド プロセッサは使用されません。
 - [Launch Runs on Remote Hosts] (Linux のみ)：ジョブを起動するのにリモート ホストを使用します。詳細は、第 9 章の「リモート Linux ホストでの run の起動」を参照してください。
 - [Configure Hosts]：リモート ホストを設定します。
 - [Generate scripts only]：run ディレクトリおよび run スクリプトをエクスポートおよび作成しますが、run は起動しません。スクリプトは、PlanAhead 環境外で後で実行できます。

PlanAhead では、run のステータスによって、run を処理して合成を開始します。ステータスは、[Design Runs] ビューに表示されます (229 ページの図 6-5)。

- run のステータスが「Not Started」の場合、run はすぐに開始されます。
- run のステータスが「Error」になっている場合は、まず run がリセットされ、終了していない run データが削除されてから、run が再開されます。
- run のステータスが「Complete」または「Out-of-Date」になっている場合は、run をリセットするかどうか確認するメッセージが表示されます。

プロセスのバックグラウンドへの移動

PlanAhead でデザイン ファイルおよび制約ファイルを読み込んで合成またはインプリメンテーションが実行されると、[Starting Run] ダイアログ ボックスが開き (図 6-8)、プロセスがバックグラウンドで実行されます。

プロセスがバックグラウンドで実行されると、バックグラウンド タスクを実行させたまま、レポートを表示したり、デザイン ファイルを開いたりといった別の機能が実行できます。この間、前の run を確認したり、レポートを表示したりして、時間を効率的に利用できます。ただし、[Tcl Console] はブロックされるので、Tcl コマンドを使用したり、開いている別のデザインに切り替えるような Tcl コマンドを必要とするタスクは実行できません。

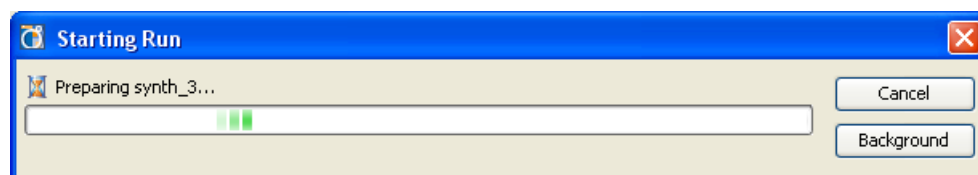


図 6-8 : [Start Run] ダイアログ ボックス

合成実行の監視

合成実行のステータスは [Compilation] ビューのコンパイル ログ、[Messages] ビューの合成の情報、警告、エラー メッセージ、[Design Runs] ビューの run で確認できます。詳細は、333 ページの「インプリメンテーション run の開始」を参照してください。

[Messages] ビューでメッセージを選択すると、該当する RTL ソース ファイルが開き、問題のソース コードがハイライトされます。図 6-9 に、[Messages] ビューでメッセージを選択し、テキスト

エディターにソース ファイルが表示された様子を示します。この機能により、合成中に発生する問題をすばやく発見し、解決できます。

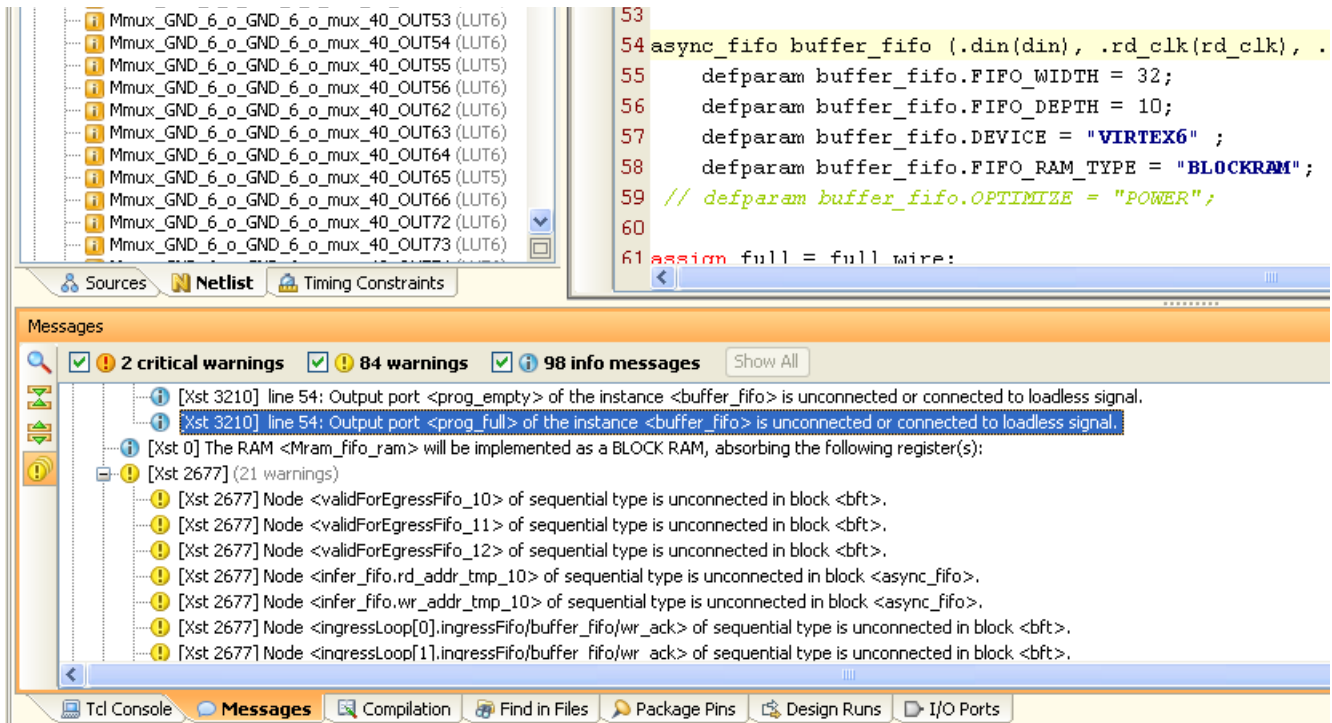


図 6-9 : [Messages] ビューからの RTL ソース ファイルの表示

合成終了後のフロー

run が終了したら、[Synthesis Completed] ダイアログ ボックスが表示されます (図 6-10)。

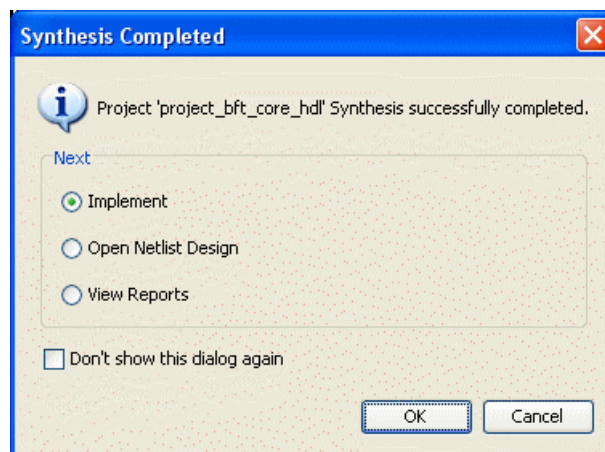


図 6-10 : [Synthesis Completed] ダイアログ ボックス

1. [Synthesis Completed] ダイアログ ボックスで、次の手順を選択します。
このダイアログ ボックスには、次のようなオプションがあります。

- **[Implement]**：インプリメンテーションを現在のインプリメンテーション プロジェクト設定を使用して起動します。インプリメンテーション プロセスの詳細は、[第 9 章「デザインのインプリメンテーション」](#)を参照してください。
 - **[Open Synthesized Design]**：ネットリスト、アクティブな制約セット、ターゲット デバイスをデザイン解析およびフロアプラン環境にインポートし、I/O ピン配置、デザイン解析、フロアプランを実行します。
詳細は、[第 7 章「合成済みデザインの制約および解析」](#)を参照してください。
 - **[View Reports]**：XST レポート ファイルを選択して表示する **[Reports]** ビューを開きます。詳細は、[336 ページの「レポート ファイルの表示」](#)を参照してください。
2. **[OK]** または **[Cancel]** をクリックします。

合成結果の解析

合成が終了したら、合成レポートを表示し、合成済みデザインを開いて解析し、インプリメンテーション前にデザインに制約を適用できます。**[Reports]** ビューには、PlanAhead の合成ツールおよびインプリメンテーション ツールで生成されたレポートのリストが表示されます。**[Reports]** ビューを開き、レポートを開いて特定の実行の詳細を確認します。[234 ページの図 6-11](#) に、合成レポートの例を示します。

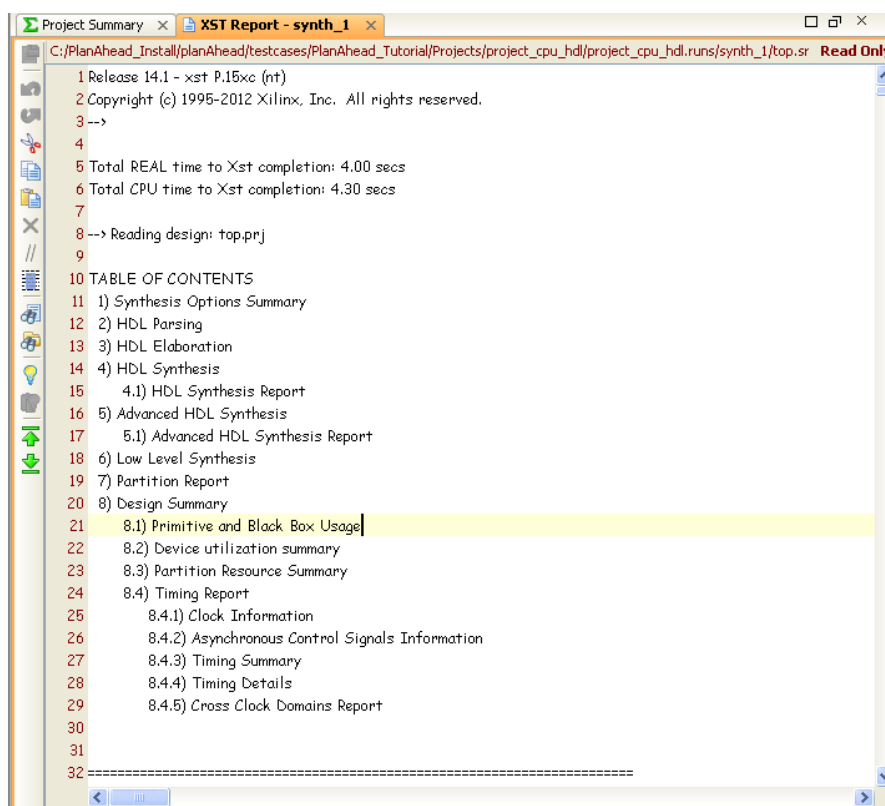


図 6-11：XST 合成レポート

このレポートの「TABLE OF CONTENTS」セクションに、合成レポートに含まれる情報がリストされています。詳細は、次を参照してください。

- [第 3 章の「\[Project Summary\] ビュー」](#)
- [第 7 章の「合成済みデザイン環境の使用」](#)

- 第 9 章の「インプリメンテーション run 結果の確認」

合成済みデザインの制約および解析

PlanAhead™ のデザイン解析および制約定義の機能は、通常インプリメンテーションを実行する前に合成済みデザインを開くと実行できますが、この章で説明する解析および制約機能の多くは、インプリメント済みデザインでも使用できます。

合成済みデザイン環境では、次を実行できます。

- I/O ピンを配置
- デザインをさまざまな側面から解析
- リソースとタイミング概算を検証
- 付録 B 「PlanAhead の DRC」に示されるデザイン ルール チェックを実行
- ISE® Design Suite 用に物理制約とタイミング制約を定義

次の PlanAhead のデザイン タスクは、合成済みデザイン環境で実行する必要があります。

- ChipScope™ Pro Analyzer デバッグ コアの挿入 (410 ページの「ChipScope を使用したデザインのデバッグ」を参照)
- ネットリスト ベースのプロジェクトのデザイン保持およびパーシャル リコンフィギュレーションのためのパーティション定義 (第 13 章の「階層デザイン手法の使用」を参照)

合成済みデザイン環境の使用

PlanAhead には、さまざまな表示からデザインを解析し、インプリメンテーション ツール前にデザインに制約を設定する環境が提供されています。

合成済みデザインを開くと、合成済みのネットリスト、アクティブな制約セット、ターゲット デバイスが読み込まれます。詳細は、173 ページの「[Netlist] ビュー」を参照してください。

合成済みデザインを作成または開くには、次のいずれかを実行します。

- Flow Navigator の [Synthesis] メニューから [Open Synthesized Design] をクリックします。
- メイン メニューから [Flow] → [Open Synthesized Design] をクリックします。
- Flow Navigator の [Synthesis] を右クリックし、[New Synthesized Design] をクリックします。
- メイン メニューから [Flow] → [New Synthesized Design] をクリックします。詳細は、34 ページの「合成済みデザインを開く」を参照してください。

合成済みデザインを開くとフロアプラン ビュー レイアウトが表示され、デザイン ロジックと階層を確認、リソース使用率とタイミング予測を表示、DRC を実行、タイミング制約および物理制約を適用できます。

フロアプラン ビューレイアウトには、図 7-1 に示すように、[Project Summary]、[Device]、[Schematic]、[Netlist]、[Sources]、[I/O Ports]、[Physical Constraint] などのビューが含まれます。

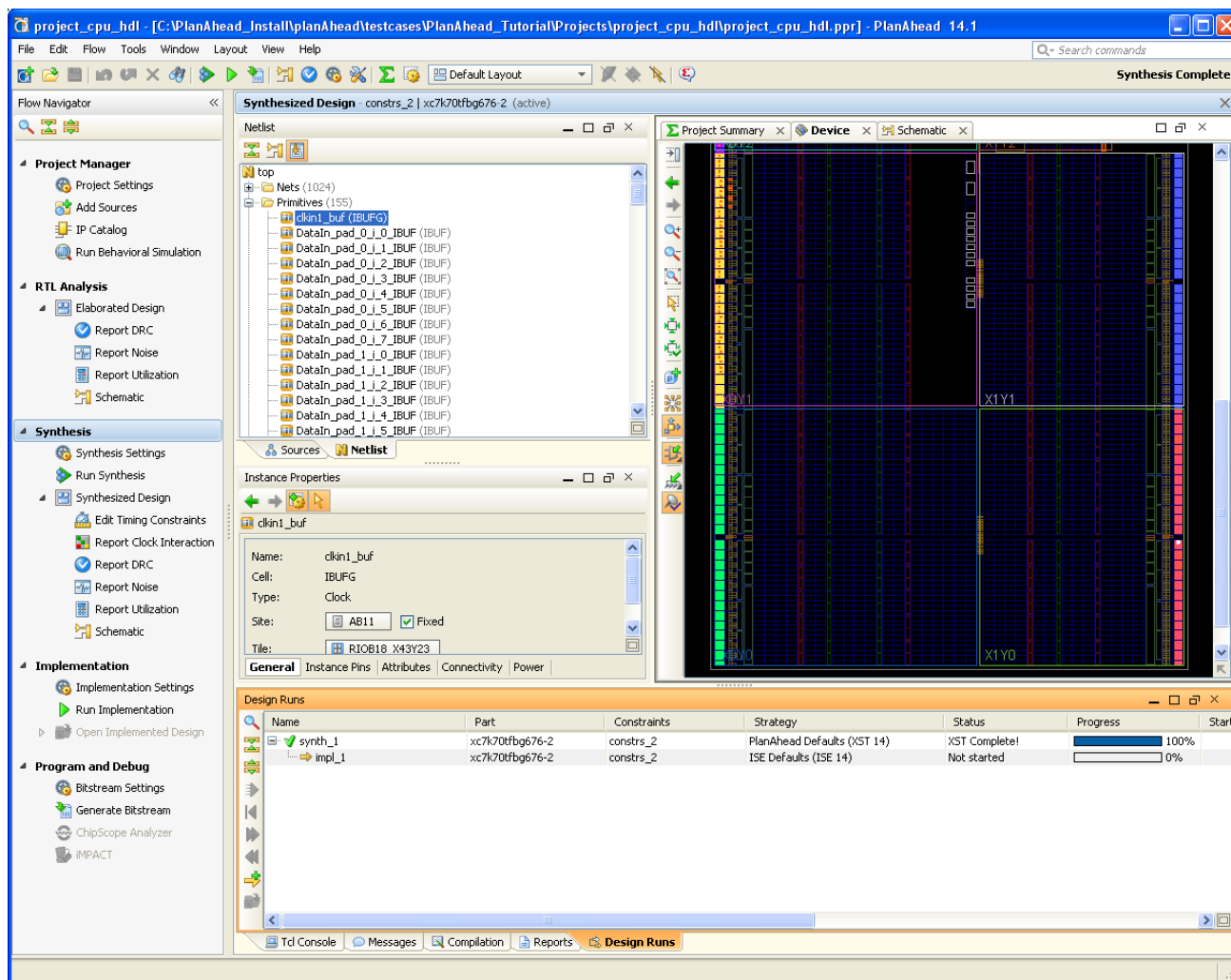


図 7-1：合成済みデザインのフロアプラン ビューレイアウト

PlanAhead には、設計中に使用する複数のビューを構成したデフォルト表示が複数用意されており、これをビュー レイアウトと呼びます。必要に応じて、[Package] ビューなどのその他のビューも開くことができます。特定のビューの詳細は、[144 ページの「汎用のビューの使用」](#)を参照してください。

ビューの構成をユーザー定義のビュー レイアウトとして保存し、必要に応じて呼び出すことができます。詳細は、[第 4 章の「ビューレイアウトの使用」](#)を参照してください。

リソース統計の表示とレポート

合成済みデザインを開くと、[Project Summary] ビューにデザインの論理内容とデバイス使用率に関する統計情報が表示されます。次の情報が表示されます。

- [Resources] ペイン：エラボレートされた RTL ネットリストおよび合成済みネットリストのリソース使用率予測
- [Compilation] ペイン：合成およびインプリメンテーション レポートからのサマリ情報

ネットリスト インスタンスまたは Pblock を選択すると、[Instance Properties] または [Pblock Properties] ビューでリソース統計を確認できます。表示される情報には、タイプ別のロジック オブジェクト数、使用されたデバイス リソースの割合、キャリー チェーン情報、クロック レポートなどがあります。この情報は、Excel スプレッドシートにエクスポートできます。

階層別のリソース概算

リソース予測は、展開可能な階層ツリーとしてグラフィカルに表示できます。リソース タイプを展開すると、論理階層の各レベルを表示できます。

デバイス リソース予測をグラフィカルに表示するには、次の手順に従います。

1. 合成済みデザインを開きます。
2. 次のいずれかをクリックします。
 - Flow Navigator → [Report Utilization]
 - [Tools] → [Report Utilization]

図 7-2 に示すように、階層のリソース使用率のサマリが表示されます。

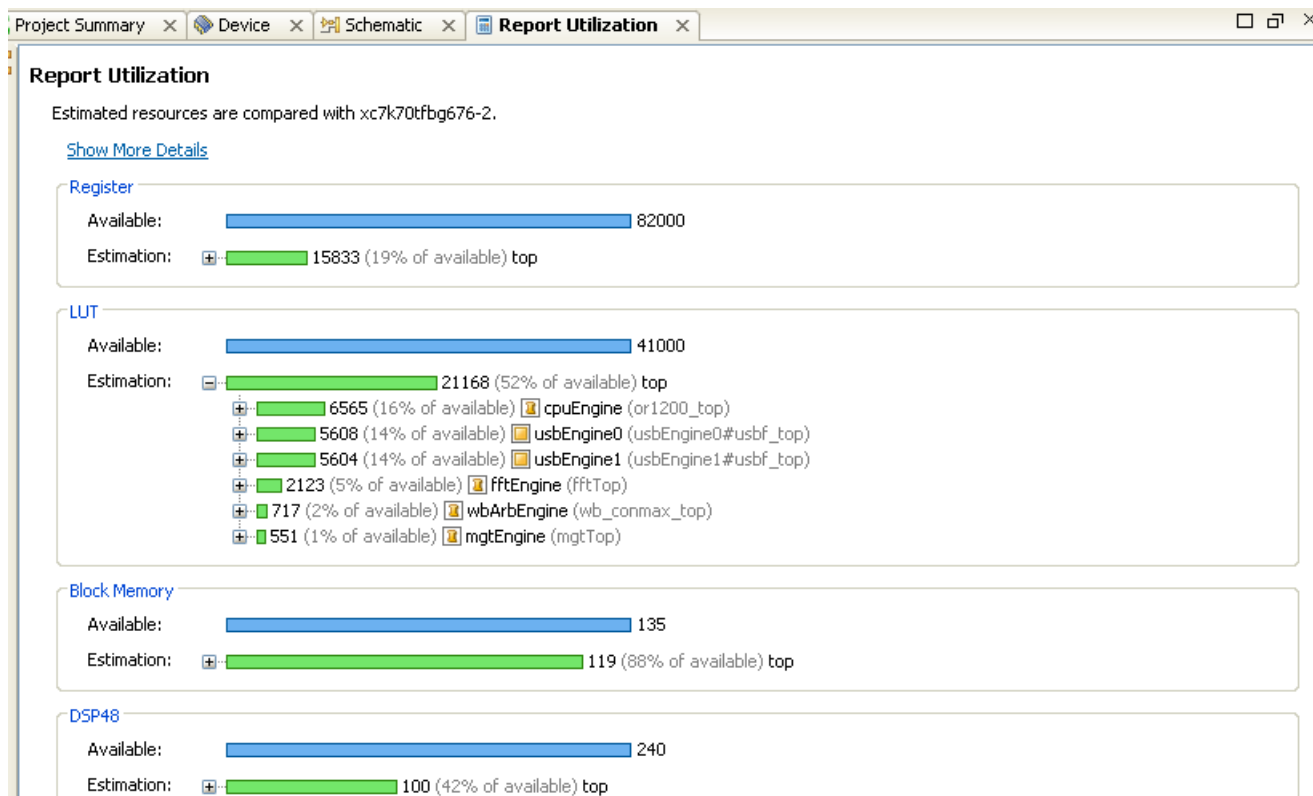


図 7-2：リソース使用率レポート

ロジック インスタンスのリソース統計の表示

PlanAhead には、デザインに含まれる予測されるデバイス リソース数をレポートする機能があります。最上位を含むロジック インスタンスのリソース統計は、[Instance Properties] ビューに表示されます。

デザイン リソース統計を表示するには、[Netlist] ビューで最上位モジュールまたはインスタンス モジュールを選択します。図 7-3 に、[Netlist] ビューで最上位モジュールを選択した状態を示します。

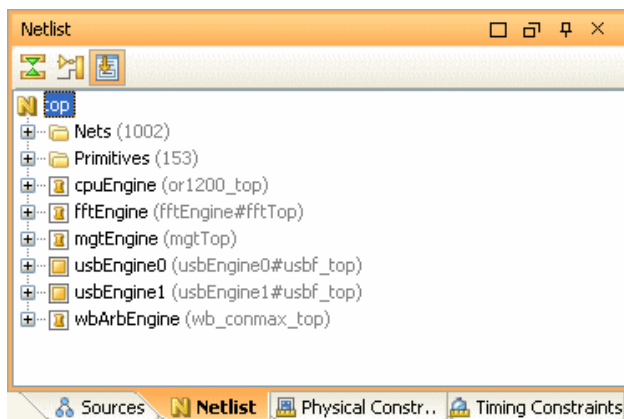


図 7-3：最上位モジュールを選択した [Netlist] ビュー

ネットリストまたはインスタンスのプロパティが [Netlist Properties] または [Instance Properties] ビューに表示されます。

表示されない場合は、モジュールを右クリックし、ポップアップメニューで [Net Properties] または [Instance Properties] をクリックします。

[Netlist Properties] には、5 つのタブがあります。[Instance Properties] には、7 つのタブがあります。[Netlist Properties] および [Instance Properties] ビューで [Statistics] タブをクリックします。

[Statistics] タブには、プリミティブ インスタンス数、インターフェイス信号数、クロック名、クロックが供給されるインスタンス数、キャリーチェーン数と最長チェーンなど、有益なデザイン情報が表示されます。

239 ページの図 7-4 に、ネットリスト リソース統計の例を示します。

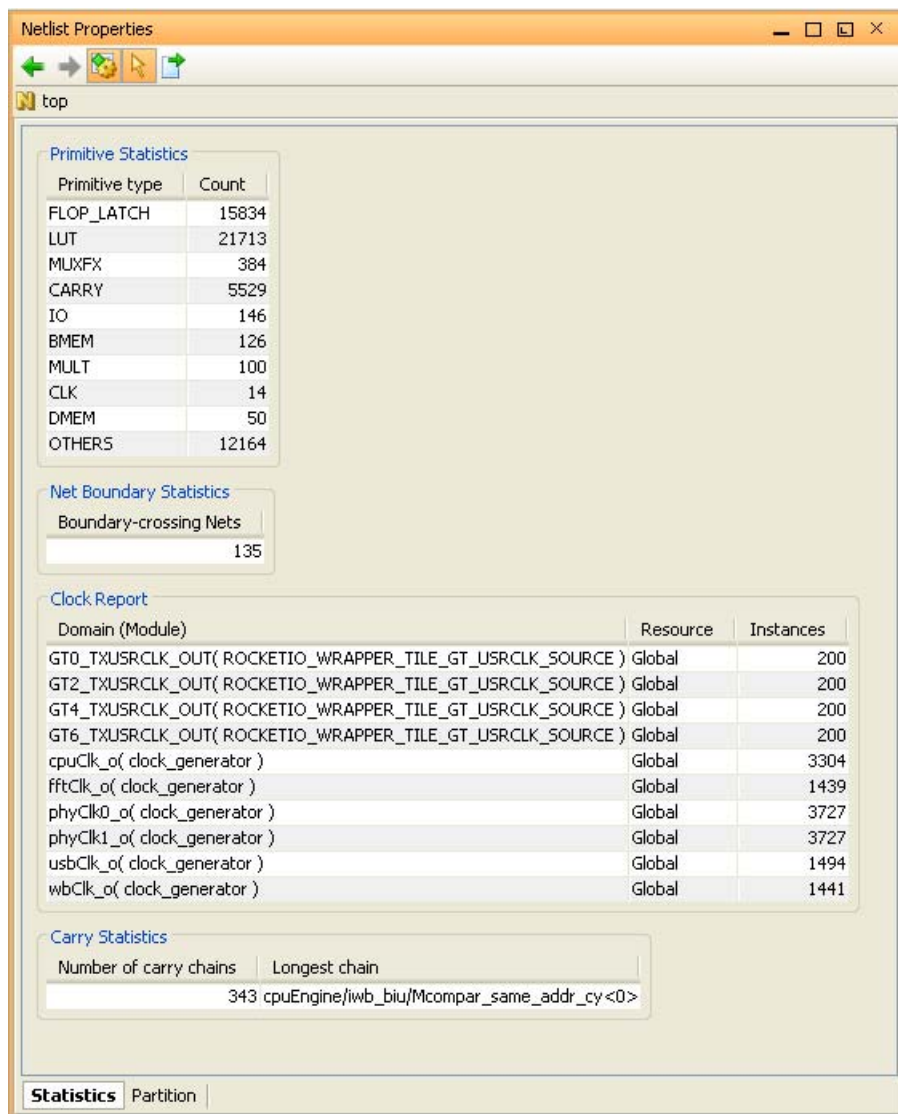


図 7-4 : ネットリスト リソース統計

Pblock のリソース統計の表示

PlanAhead では、Pblock のロジック使用率統計を表示でき、Pblock エリアに割り当てられたロジックを配置するのに十分なデバイス リソースが含まれているかどうかを確認するのに役立ちます。ROOT Pblock はデザインの最上位と認識され、デザイン全体の使用率統計を示します。

Pblock の使用率統計を表示するには、次の手順に従います。

1. [Physical Constraints] ビューで、[ROOT] またはいずれかの Pblock をクリックします。
240 ページの図 7-5 では、ROOT Pblock が選択されています。

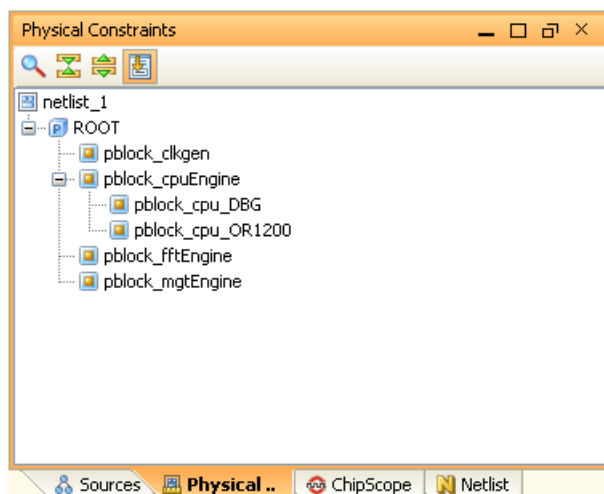


図 7-5 : [Physical Constraints] ビューで [ROOT] を選択

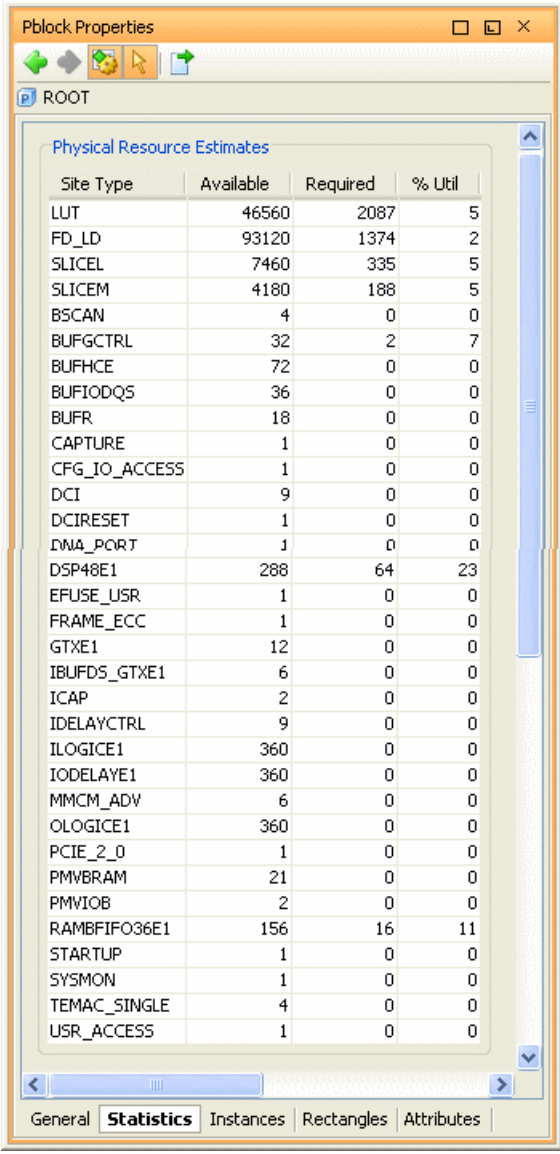
Pblock プロパティが [Pblock Properties] ビューに表示されます。

2. [Pblock Properties] が表示されない場合は、[ROOT] または Pblock を右クリックし、ポップアップメニューで [Pblock Properties] をクリックします。

詳細は、第 10 章の「Pblock プロパティの表示」を参照してください。

[Statistics] タブの使用

[Statistics] タブには、さまざまなデバイス リソースのデバイス全体での使用状況、キャリー チェーン数と最長チェーン、RPM 数と最大サイズ、クロック名とクロックが供給されているインスタンスの数、I/O 使用率、信号とプリミティブ インスタンスの数などのデザイン情報が表示されます。
241 ページの図 7-6 に、[Statistics] タブの例を示します。



The image shows the 'Pblock Properties' dialog box with the 'Statistics' tab selected. It displays a table of physical resource estimates for various site types. The table has four columns: Site Type, Available, Required, and % Util. The data is as follows:

Site Type	Available	Required	% Util
LUT	46560	2087	5
FD_LD	93120	1374	2
SLICEL	7460	335	5
SLICEM	4180	188	5
BSCAN	4	0	0
BUFGCTRL	32	2	7
BUFHCE	72	0	0
BUFIOQ5	36	0	0
BUFR	18	0	0
CAPTURE	1	0	0
CFG_IO_ACCESS	1	0	0
DCI	9	0	0
DCIRESET	1	0	0
DMA_PORT	1	0	0
DSP48E1	288	64	23
EFUSE_USR	1	0	0
FRAME_ECC	1	0	0
GTXE1	12	0	0
IBUFDS_GTXE1	6	0	0
ICAP	2	0	0
IDELAYCTRL	9	0	0
ILOGICE1	360	0	0
IODELAYE1	360	0	0
MMCM_ADV	6	0	0
OLOGICE1	360	0	0
PCIE_2_0	1	0	0
PMVBRAM	21	0	0
PMVIOB	2	0	0
RAMBFIFO36E1	156	16	11
STARTUP	1	0	0
SYSMON	1	0	0
TEMAC_SINGLE	4	0	0
USR_ACCESS	1	0	0

図 7-6 : [Pblock Properties] ビュー :[Statistics] タブ

リソース統計レポートのエクスポート

表示されているデータをスプレッドシート ファイルに保存できます。PlanAhead では、階層形式のレポートが生成されるので、レベル別に各モジュールの予測値をレポートする階層数を定義できます。

リソース統計レポートをエクスポートするには、次の手順に従います。

1. [Export Statistics] ボタン をクリックし、データをスプレッドシート ファイルにエクスポートします。



図 7-7 に示すように、[Export Netlist Statistics] ダイアログ ボックスが開きます。

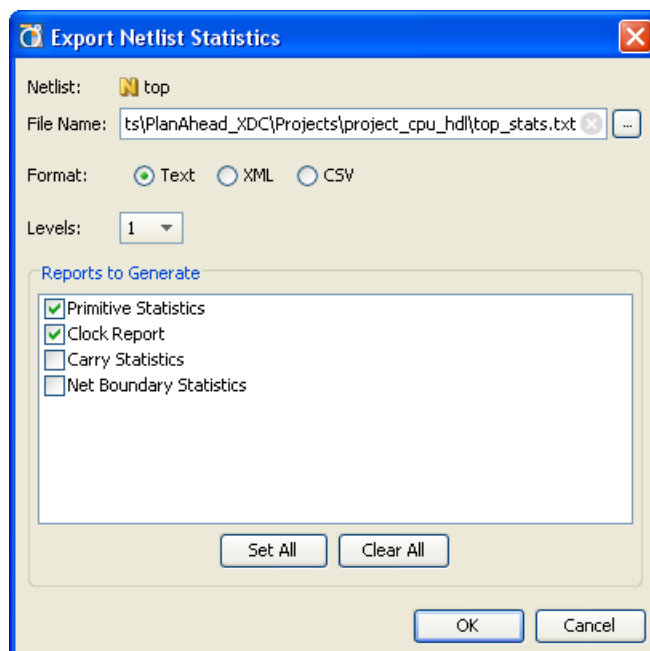


図 7-7：ネットリスト リソース統計のエクスポート

[Export Netlist Statistics] ダイアログ ボックスでは、次のオプションを設定できます。

- [File Name]：スプレッドシート ファイルの名前と保存場所を入力します。
- [Format]：テキスト、XML、CSV から出力ファイルの形式を選択します。
- [Levels]：レポートに含める階層レベル数を設定します。
- [Reports to Generate]：出力レポート ファイルに含めるネットリスト統計のタイプを定義します。

2. 出力ファイルのオプションを選択し、[OK] をクリックします。

ロジックの解析

PlanAhead には、デザイン ロジックを解析するための表示が複数あります。

- [Netlist] および [Hierarchy] ビューには、ナビゲート可能な階層ツリー形式の表示が含まれます。
- [Schematic] ビューでは、選択したロジックを展開したり階層表示にできます。
- [Device] ビューは、デバイス、配置ロジック オブジェクト、および接続をグラフィカルに表示します。1 つのビューで選択した情報はほかのビューでも選択されるようになっており、必要な情報をすばやく見つけることができます。
- インプリメント済みデザインでは、さらに追加の解析機能があります。これらの解析機能は、配置およびタイミング結果がインポートされた後に使用するものです。詳細は、[第 11 章「インプリメンテーション結果の解析」](#)を参照してください。

次のセクションでは、PlanAhead ソフトウェアで使用可能なロジック解析方法を説明します。

ロジック階層の表示

[Netlist] ビューには、RTL のロジック階層が表示されます。ネットリスト内のロジック インスタンスまたはネットを、展開して選択できます。別のビューでロジック オブジェクトを選択すると、[Netlist] ビューが自動的に展開され、そのロジック オブジェクトが表示されます。詳細は、[173 ページの「\[Netlist\] ビュー」](#)を参照してください。

インスタンスまたはネットの情報は、[Instance Properties] または [Net Properties] ビューに表示されます。

[Hierarchy] ビューには、RTL ロジック階層がグラフィカルに表示されます。各モジュールの大きさが、その他のモジュールに相対的な比率で表示されるので、選択したロジック モジュールのサイズや位置を判断できます。詳細は、[177 ページの「\[Hierarchy\] ビュー」](#)を参照してください。

ロジック回路図の解析

[Schematic] ビューでは、選択したロジックを展開したり表示できます。[Schematic] ビューを表示するには、少なくとも 1 つのロジック オブジェクトを選択する必要があります。任意のロジックを選択して、[Schematic] ビューで表示できます。

タイミング パスのグループを表示して、そのパス上のすべてのインスタンスを表示できます。これにより、タイミング クリティカルなモジュールが含まれる箇所を視覚的に表示できるので、フロアプランしやすくなります。

[Schematic] ビューを開くには、次の手順に従います。

1. 1 つまたは複数のインスタンス、ネット、タイミング パスを選択します。
2. ツールバーまたはポップアップ メニューで [Schematic] をクリックするか、F4 キーを押します。



[Schematic] ビューが開き、選択したロジック オブジェクトが表示されます。

3. この後、ピン、インスタンス、階層モジュールを選択して、ロジックを展開できます。

詳細は、[163 ページの「\[Schematic\] ビュー」](#)を参照してください。

階層接続の解析

最上位フロアプランを作成して、244 ページの図 7-8 に示すように、デザインの接続フローを表示すると便利な場合があります。

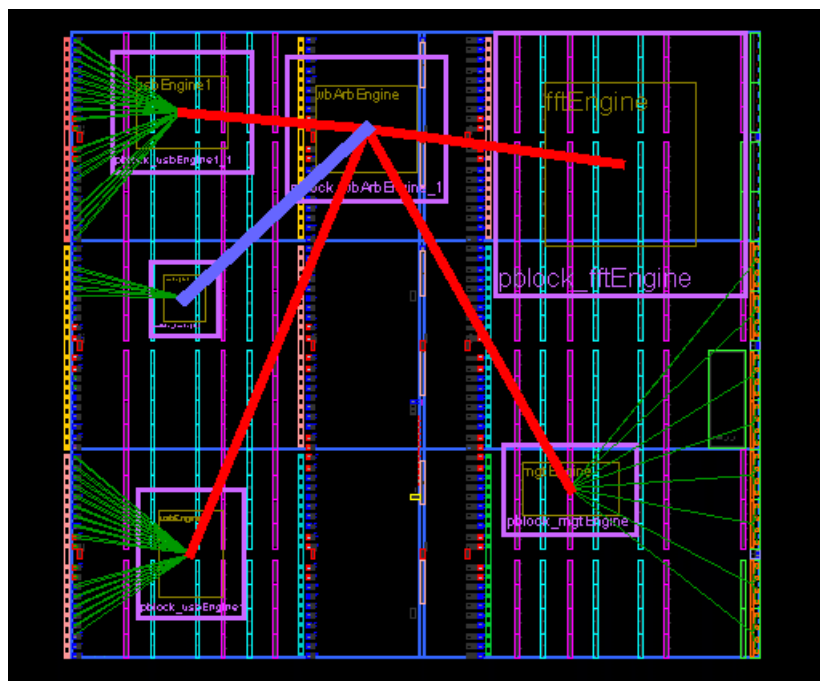


図 7-8：最上位デザインの接続の表示

デザイン階層の主なレベルで Pblock を作成すると、最上位フロアプランを作成できます。詳細は、第 10 章「デザインのフロアプラン」を参照してください。

ネット バンドルは、モジュール間の接続が多いことを示します。ネットを選択すると、[Net Bundle Properties] にバンドル情報が表示されます。

ネット バンドルの色と太さは、ネットに含まれる信号数に基づいて設定できます。これは、[Tools] → [Options] → [General] をクリックし、[Connectivity Display] セクションで指定します。ネット バンドルを表示するオプションには、[Mesh] と [Tree] があります。

階層を移動し、最上位インスタンスのサブモジュールを作成して、さらに細かく表示することもできます。

最上位フロアプランは I/O ピン配置コンフィギュレーションの質を示すもので、配線の密集の問題を特定するのに役立ちます。

各モジュールのリソース統計およびクロック要件からも、発生する可能性のある配置問題を知ることができます。

詳細は、付録 E「その他のリソース」に示す『フロアプラン設計手法ガイド』(UG633) を参照してください。

ChipScope デバッグ コアの挿入

PlanAhead では、合成済みデザインに ChipScope Integrated Logic Analyzer (ILA) および ChipScope Onboard Netlist (ICON) デバッグ コアを挿入できます。デバッグ ネットを選択すると、Set Up ChipScope ウィザードを使用してコアをコンフィギュレーションできます。

追加された ChipScope コアは、デザイン ネットリストを変更しても保持されます。ChipScope ネットリスト オーバーレイにより、新しいネットリストを追加してプロジェクトで開いたときに、選択したデバッグ ネットが再接続されます。矛盾があった場合は、警告メッセージが表示されます。デバッグ ロジックの挿入と ChipScope を使用したデバッグについては、第 12 章の「ChipScope を使用したデザインのデバッグ」を参照してください。

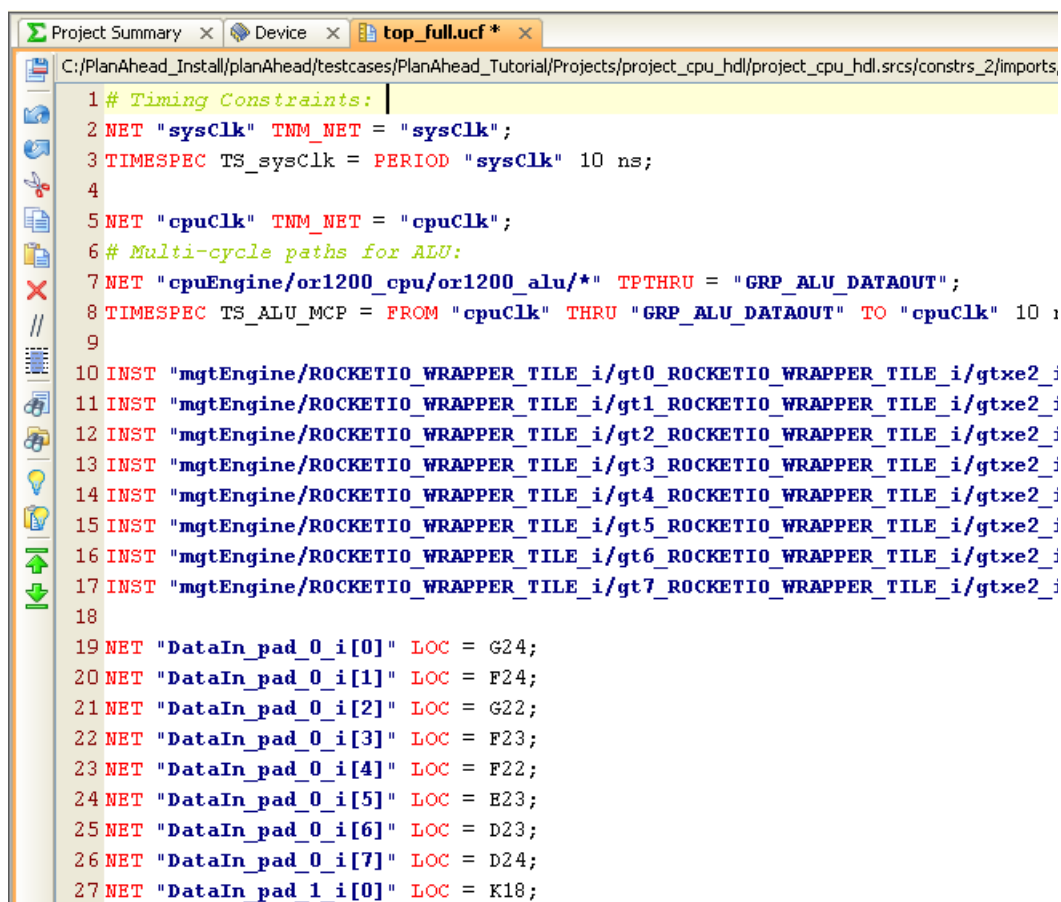
タイミング制約の定義

PlanAhead にはデザインのタイミング制約を定義および変更する機能がありますが、制約が正しい制約セットおよびターゲット UCF に記述され、予測どおりに適用されたかどうかは、ユーザーが確認する必要があります。

制約の編集

制約は、定義された UCF で表示および変更できます。これにより、制約を切り取って貼り付けたり、既存の制約の値を変更しやすくなります。

UCF をテキスト エディターで開くには、[Sources] ビューで該当する制約ファイルをダブルクリックします。図 7-9 に、UCF を開いたところを示します。1 度に複数のファイルを開くことができます。



```
1 # Timing Constraints:
2 NET "sysClk" TNM_NET = "sysClk";
3 TIMESPEC TS_sysClk = PERIOD "sysClk" 10 ns;
4
5 NET "cpuClk" TNM_NET = "cpuClk";
6 # Multi-cycle paths for ALU:
7 NET "cpuEngine/or1200_cpu/or1200_alu/*" TPTHU = "GRP_ALU_DATAOUT";
8 TIMESPEC TS_ALU_MCP = FROM "cpuClk" THRU "GRP_ALU_DATAOUT" TO "cpuClk" 10 ns;
9
10 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt0_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
11 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt1_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
12 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt2_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
13 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt3_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
14 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt4_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
15 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt5_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
16 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt6_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
17 INST "mgtEngine/ROCKETIO_WRAPPER_TILE_i/gt7_ROCKETIO_WRAPPER_TILE_i/gtxe2_i";
18
19 NET "DataIn_pad_0_i[0]" LOC = G24;
20 NET "DataIn_pad_0_i[1]" LOC = F24;
21 NET "DataIn_pad_0_i[2]" LOC = G22;
22 NET "DataIn_pad_0_i[3]" LOC = F23;
23 NET "DataIn_pad_0_i[4]" LOC = F22;
24 NET "DataIn_pad_0_i[5]" LOC = E23;
25 NET "DataIn_pad_0_i[6]" LOC = D23;
26 NET "DataIn_pad_0_i[7]" LOC = D24;
27 NET "DataIn_pad_1_i[0]" LOC = K18;
```

図 7-9: テキスト エディター

テキスト エディターで使用可能なコマンドおよび機能については、[184 ページの「テキスト エディターの使用」](#)を参照してください。

制約テンプレートの使用

新規制約を定義する際、UCF テンプレートを使用できます。PlanAhead では、UCF テンプレートに加え、Verilog および VHDL 言語テンプレートも提供されています。選択したテンプレートは、テキスト エディターで開いているファイルにインスタンスエートできます。詳細は、[186 ページの「言語テンプレートおよび制約テンプレートのインスタンスエート」](#)を参照してください。



[Timing Constraints] ビューの使用

[Timing Constraints] ビューを使用すると、デザインのタイミング制約を表示、編集、作成できます。

PlanAhead には、デザインで定義されているタイミング制約を表示するビューがあります。制約は制約セット別に表示され、同じプロジェクトで開いているデザイン間で異なります。異なる制約、デバイス、I/O ピンなどを試すことができます。

定義済みの値を変更したり、新しい制約を作成するには、[Timing Constraints] ビューを使用します。デザインで定義されているタイミング制約を表示するには、[Timing Constraints] ビューをクリックするか、[Window] → [Timing Constraints] をクリックします。

[Timing Constraints] ビューが[図 7-10](#)のように表示されます。

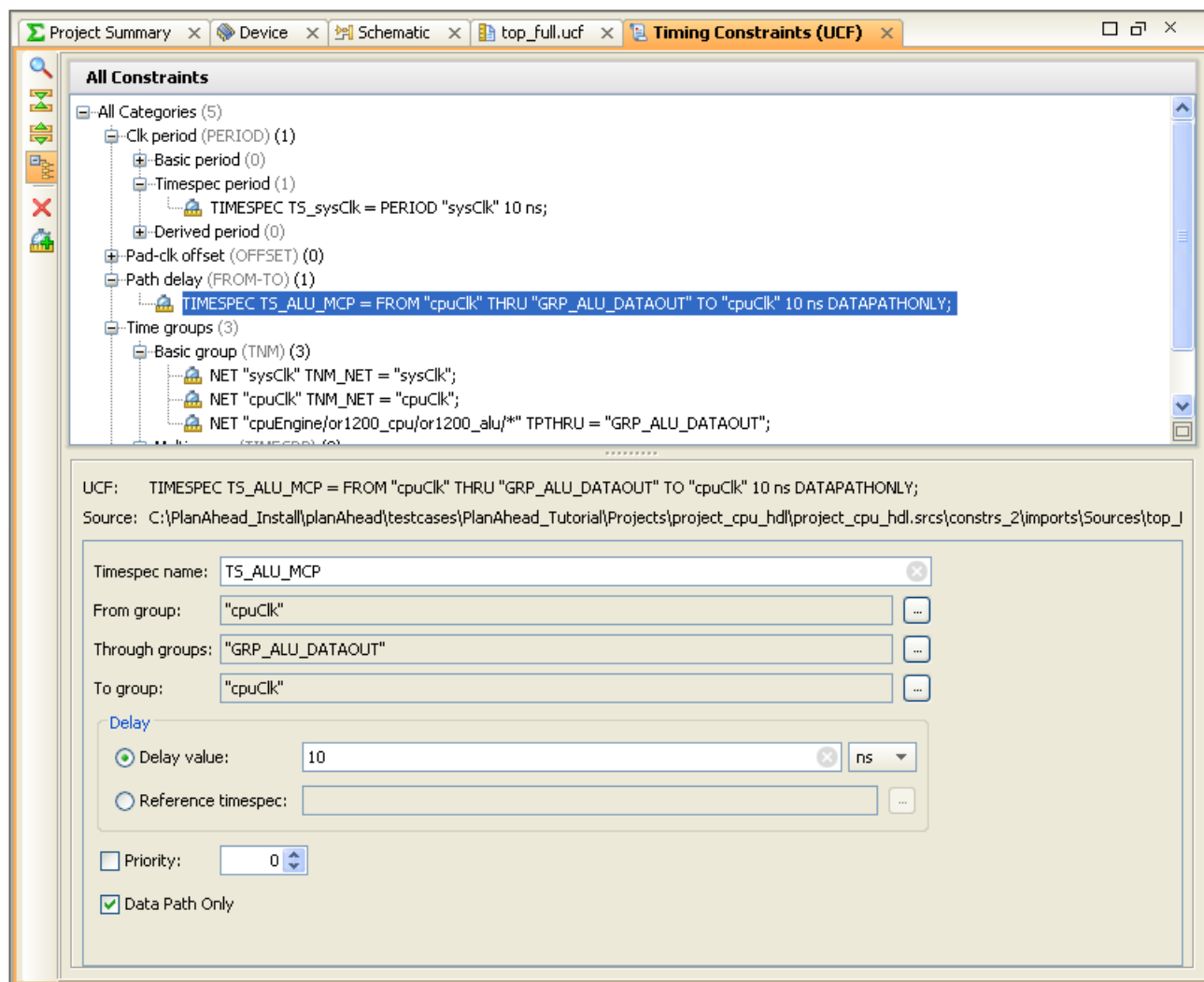


図 7-10 : [Timing Constraints] ビュー

制約は、タイプ別で表示できるほか、リストとしても表示できます。

制約を図 7-10 で示すようにタイプ別に並べ替えると、制約タイプを展開または非展開できます。かつこ内に各タイプの定義済みの制約の数が示されます。

すべての定義済みタイミング制約のリストを表示するには、[Timing Constraints] ビューで [Group by type] ボタンをオフにします。



定義済み制約の値は、247 ページの図 7-10 に示すように、[Timing Constraints] ビューの上画面で制約を選択すると、下画面でその制約の値を変更できます。変更可能な制約の属性が編集可能なフィールドに表示されます。

[Apply] をクリックし、変更を保存します。

新しいタイミング制約の追加

1. 次のいずれかの方法で [New Timing Constraints] ダイアログ ボックスを開きます。
 - [Tools] → [Timing] → [New Timing Constraint] をクリックします。

- [Timing Constraints] ビューの [New Timing Constraint] ボタンをクリックします。

図 7-11 のように、[New Timing Constraint] ダイアログ ボックスが開きます。

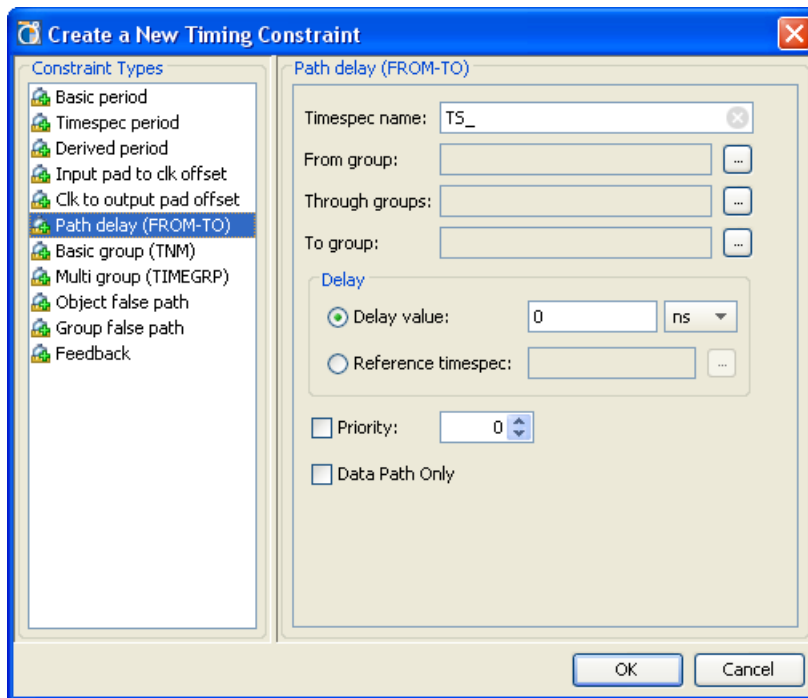


図 7-11 : [New Timing Constraint] ダイアログ ボックス

2. [Constraint Types] で作成する制約の種類を選択します。
選択すると、右側に対応するフィールドが表示されます。
3. 制約値を定義します。
4. [OK] をクリックします。

タイミング制約およびタイミング制約の構文については、付録 E「その他のリソース」に示す『制約ガイド』(UG625) を参照してください。

タイミング制約の削除

デザインから制約を削除するには、制約または制約グループを [Timing Constraints] ビューで右クリックし、ポップアップ メニューで [Delete] をクリックします。制約を削除することを確認するダイアログ ボックスが表示されます。削除する場合は、[OK] をクリックします。

注記： タイミング制約は相互依存しているため、1 つの制約を削除すると関連する制約がいくつか削除される場合があります。タイミング制約の追加、編集、および削除は、やり直すことができません。

タイミング解析の実行

合成済みデザインのタイミング解析は、パスに必要な制約が設定されているかどうかを確認し、インプリメンテーションを効率的に実行するために役立ちます。Pblock や LOC 制約のような物理制約をデザインに割り当てていくと、パス遅延は概算ですが、より正確なタイミング解析が得られる

ようになります。デザインのインプリメンテーションが終了すると、タイミング解析にインプリメントされたデザインからの実際の配線パス遅延が含まれるようになります。

PlanAhead のスタティック タイミング解析は、タイミング予測目的にのみ使用し、最終的なタイミングを確認する手段としては使用しないでください。最終的なタイミングを確認するには、インプリメント済みデザインに対して、配置配線済みデザインの回路遅延とパス遅延の両方を使用して TRCE を実行できます。詳細は、[380 ページの「インプリメント済みデザインでの TRCE の実行」](#)を参照してください。

[Report Timing] コマンドの使用

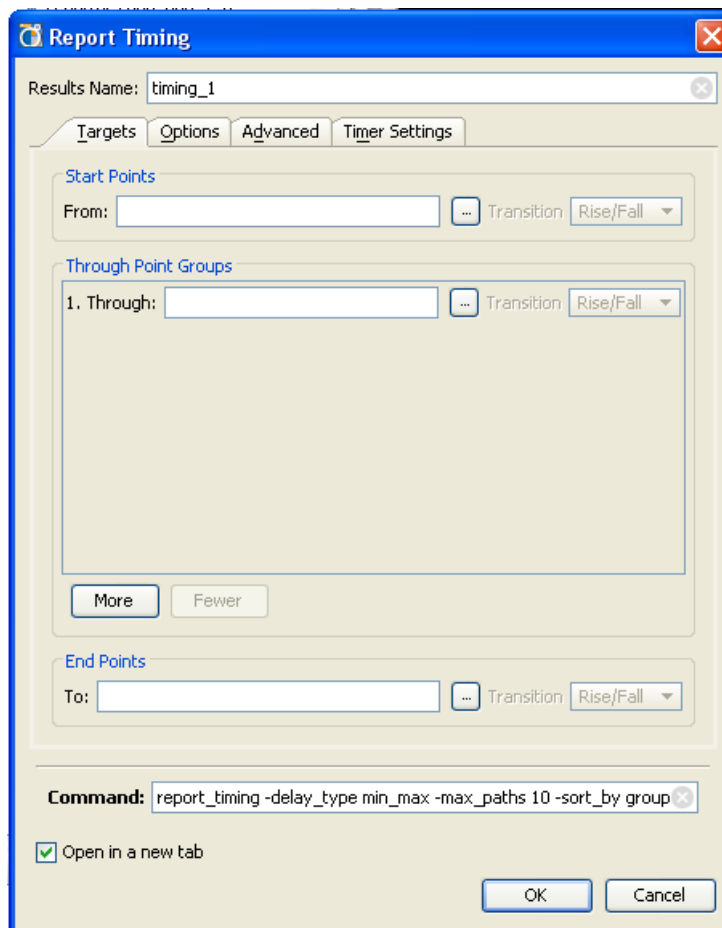


図 7-12 : [Report Timing] ダイアログ ボックス

次のいずれかの方法でタイミング解析を実行します。

- [Tools] → [Timing] → [Report Timing] をクリックします。
- Flow Navigator で [Synthesized Design] → [Report Timing] をクリックします。

タイミング レポートをカスタマイズする [Report Timing] ダイアログ ボックスが開きます (図 7-12)。

[Report Timing] ダイアログ ボックスのオプションは次のとおりです。

- [Results Name]: タイミング レポートの名前を指定します。

- [Targets] タブ：開始点、スルー ポイント、終了点に基づいてレポートされるパスをフィルターします。
- [Options] タブ：レポートの生成に使用されるオプションを指定します。
- [Advanced] タブ：タイミング レポートの生成と保存に使用するアドバンス オプションを指定します。
- [Timer Settings] タブ：レポートの生成に使用されるタイミング エンジンと遅延オプションを指定します。
- [Command]：レポートの生成に使用される Tcl コマンドを表示します。
注記：このフィールドは編集できます。実行される Tcl コマンドが変更されますが、ダイアログ ボックスで指定されるオプションは変更されません。
- [Open in a new tab]：タイミング解析の結果を [Timing Results] ビューに新しいタブで表示するか、既存のタブと置き換えるかを指定します。
- [OK] をクリックし、タイミング レポートを生成します。

次のセクションでは、[Report Timing] ダイアログ ボックスのさまざまなタブについて説明します。

[Targets] タブ

249 ページの図 7-12 に示す [Report Timing] ダイアログ ボックスの [Target] タブでは、タイミング レポートに含めるパスの開始点、スルー ポイント、および終了点を指定できます。

デフォルトではこれらのフィールドは空白で、指定したパスの最大数まですべてのポイントがレポートに含まれます。フィールドに値を追加すると、該当するパスのみに焦点を当てたレポートを生成できます。

次のフィールドがあります。

- [Start Points]：パスの解析を開始する同期エレメントを指定します。[Start Points] フィールドには、次のオプションがあります。
 - [From]：開始点に基づいて解析するパスをフィルターする記述を入力します。フィルターを直接入力するか、[Choose Start Points] ダイアログ ボックスでフィルターを作成します。
 - [Choose Start Points] (開始点を選択)：開始点に基づいたパスのフィルターに使用する式を作成するダイアログ ボックスを開きます。このダイアログ ボックスについては、[「\[Choose Points\] ダイアログ ボックス」](#)で詳細に説明します。
 - [Transition]：同期エレメントのアクティブ クロック エッジでパスをさらにフィルターします。次のオプションがあります。
 - [Rise]：開始点の同期エレメントで、クロックの立ち上がりエッジでトリガーされるもののみを解析します。
 - [Fall]：開始点の同期エレメントで、クロックの立ち下がりエッジでトリガーされるもののみを解析します。
 - [Rise/Fall]：立ち上がりエッジと立ち下がりエッジの片方または両方ともを含めます。
- [Through Point Groups]：解析するパスが通過するポイントを入力します。次のフィールドがあります。
 - [Through]：通過するポイントに基づいて解析するパスをフィルターする記述を入力します。
 - [Choose Through Points] (スルー ポイントを選択)：スルー ポイントに基づいたパスのフィルターに使用する式を作成するダイアログ ボックスを開きます。

- **[Transition]** : スルー ポイントの同期エレメントのアクティブ クロック エッジでパスをさらにフィルターします。次のオプションがあります。
 - **[Rise]** : スルー ポイントの同期エレメントで、クロックの立ち上がりエッジでトリガーされるもののみを解析します。
 - **[Fall]** : スルー ポイントの同期エレメントで、クロックの立ち下がりエッジでトリガーされるもののみを解析します。
 - **[Rise/Fall]** : 立ち上がりエッジと立ち下がりエッジの片方または両方ともを含めます。
- **[More]** : スルー ポイントのフィルター記述を追加します。
- **[Fewer]** : 既存のスルー ポイントのフィルター記述を削除します。
- **[End Points]** : パスの解析を終了する同期エレメントを指定します。このセクションには、次のフィールドが含まれます。
 - **[To]** : パスのスタート ポイントまたはエンド ポイントに基づいたパスのフィルターに使用する記述を含めます。
 - **[Choose End Points]** (終了点を選択) : 終了点に基づいて解析するパスをフィルターする記述を入力します。
 - **[Transition]** : 同期エレメントのアクティブ クロック エッジでパスをさらにフィルターします。次のオプションがあります。
 - **[Rise]** : 終了点の同期エレメントで、クロックの立ち上がりエッジでトリガーされるもののみを解析します。
 - **[Fall]** : 終了点の同期エレメントで、クロックの立ち下がりエッジでトリガーされるもののみを解析します。
 - **[Rise/Fall]** : 立ち上がりエッジと立ち下がりエッジの片方または両方ともを含めます。

[Choose Points] ダイアログ ボックス

[Choose Points] ダイアログ ボックスでは、エレメント タイプおよびパターン的一致する文字列に基づいてタイミング解析を必要とするデザイン エレメントを選択し、開始点、スルー ポイント、終了点のフィルター文字列を入力できます。図 7-13 は、[Choose Start Points] ダイアログ ボックスを示しています。

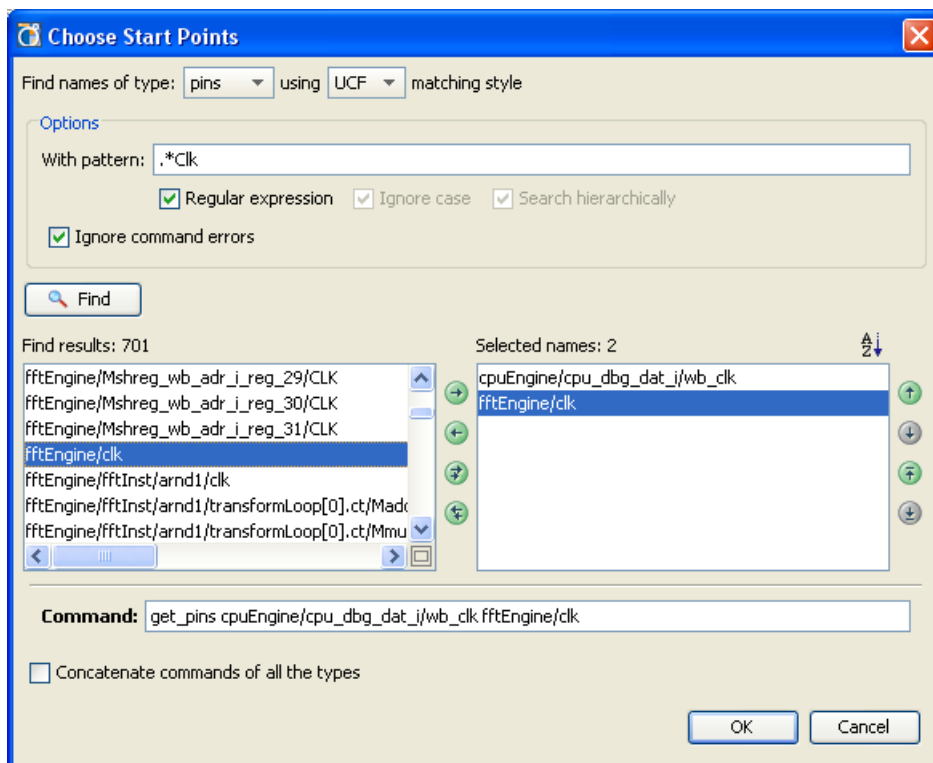


図 7-13 : [Choose Start Points] ダイアログ ボックス

このダイアログ ボックスには、次のようなオプションがあります。

- [Find names of type] : デザイン エレメントのタイプに基づいてポイントをフィルターします。このフィールドには、次のオプションがあります。
 - [cells] : セル名に基づいてデザイン エレメントを選択します。
 - [clocks] : クロック名に基づいてデザイン エレメントを選択します。
 - [pins] : ピン名に基づいてデザイン エレメントを選択します。
 - [ports] : ポート名に基づいてデザイン エレメントを選択します。
 - [nets] : スルー ポイントのネットを指定します。
- [matching style] : デザイン エレメントのフィルターに使用するパターン一致のタイプを選択します。このフィールドには、次のオプションがあります。
 - [UCF] : パターン一致に UCF ベースの構文を使用します。
 - [SDC] : パターン一致に SDC ベースの構文を使用します。
- [With pattern] : デザイン エレメントのフィルターに使用するパターンを入力します。このフィールドは、次のオプションを使用して変更できます。
 - [Regular expression] : 検索文字列に正規表現の構文を使用します。
 - [Ignore case] : 大文字/小文字を区別せずに文字列を検索します。
 - [Search hierarchically] : SDC パターン一致を選択した場合に使用可能なオプションで、SDC ベースの検索パターンを階層の全レベルに適用します。
- [Of these objects] : ダイアログ ボックスの選択に基づいてオブジェクトを選択します。次のオプションがあります。

注記：このフィールドは [matching style] を [SDC] に設定している場合のみ表示されます。

- [Include leaf pins of these objects only] : SDC 構文のオプションで、ピン コンポーネントのみを選択し、階層の境界を越えるピンは選択されません。
- [Select cells or net] : SDC 構文のオプションで、このボタンをクリックすると、反復検索表現を作成する [Select Objects] ダイアログ ボックスを開きます。
- [Filter matching name with expression] : SDC 構文のオプションで、`-filter` コマンドを指定します。
- [Ignore command errors] : Tcl コマンドでタイミング レポートを生成中に出力される警告メッセージが表示されないようにします。
- [Find] : 定義したパターンに基づいて検索を実行します。
- [Find results] : オブジェクト検索の結果を表示します。
 - [Move the selected items to the right] : 選択したオブジェクトを [Find results] から [Selected names] に移動します。
 - [Move the selected items to the left] : 選択したオブジェクトを [Selected names] から [Find results] に移動します。
 - [Move all the items to the right] : [Find results] のすべてのオブジェクトを [Selected names] に移動します。
 - [Move all the items to the left] : [Selected names] のすべてのオブジェクトを [Find results] に移動します。
- [Selected names] : 検索結果から現在選択されている開始点、スルー ポイント、終了点を表示します。
- [Command] : オブジェクトを選択する Tcl コマンドを表示します。

[Options] タブ

[Report Timing] ダイアログ ボックスの [Options] タブでは、生成するタイミング レポートのタイプとパス数を指定します。254 ページの図 7-14 に、[Options] タブを示します。

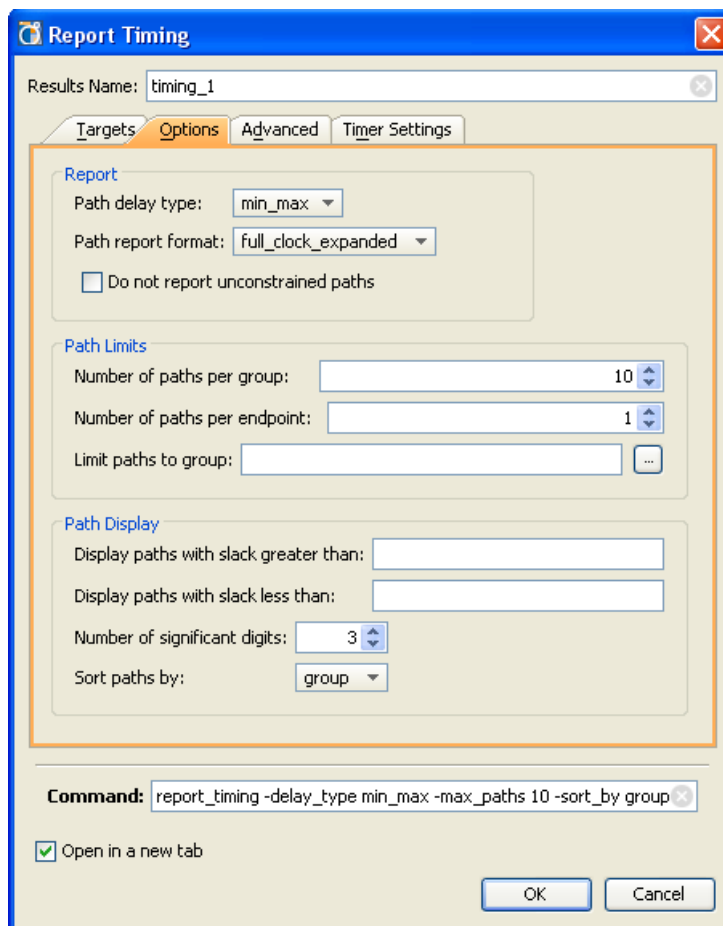


図 7-14 : [Report Timing] ダイアログ ボックスの [Options] タブ

このタブには、次のフィールドが含まれます。

- [Delay type] : 遅延タイプを指定します。使用できる値は、min、min_rise、min_fall、max、max_rise、max_fall、min_max です。rising または falling パス遅延は、タイミング パスの終了点での遷移を示します。
- [Path report format] : 生成するタイミング レポートのタイプを指定します。ほとんどの場合、GUI のタイミング レポートのフォーマットと出力されるタイミング レポートのフォーマットは、選択したオプションによって異なります。次のオプションがあります。
 - [end] : 各パスのスラックおよびエンドポイント情報のみを含むエンドポイント レポートを指定します。
 - [full] : データ パスの完全パスの詳細を含み、クロック パスの詳細は含まないタイミング レポートを指定します。
 - [full_clock] : データ パスの完全パスの詳細とクロック パスのサマリ詳細を含むタイミング レポートを指定します。
 - [full_clock_expanded] : データ パスの完全パスの詳細とクロック パスの詳細を含むタイミング レポートを指定します。
 - [short] : データ パスのサマリ詳細を含み、クロック パスの詳細は含まないタイミング レポートを指定します。

- [summary] : デザインのタイミング パフォーマンスに関するサマリ情報のみを含むタイミング レポートを指定します。
- [Do not report unconstrained path] : 制約の適用されたパスのみがレポートされるよう指定します。
- [Number of paths per group] : 各グループでレポートされるパスの数を指定します。
- [Number of paths per endpoint] : 各エンドポイントでレポートされるタイミング パスの最大数を指定します。
- [Limit paths to group] : パスを特定グループに制限します。グループ識別子は、直接入力するか、[Choose path groups] ダイアログ ボックスから入力します。
- [Display paths with slack greater than] : 最小スラック値に基づいて表示されるパスをフィルターします。ここで指定した値よりもスラック値が大きいパスのみが表示されます。
- [Display paths with slack less than] : 最大スラック値に基づいて表示されるパスをフィルターします。ここで指定した値よりもスラック値が小さいパスのみが表示されます。
- [Significant digits] : タイミング レポートの遅延値の有効桁数を指定します。デフォルト値は3です。
- [Sort paths by] : タイミング レポートの並べ替えの基準を選択します。次のオプションがあります。
 - [group] : グループ名を基準にタイミング レポートを並べ替えます。
 - [slack] : パス スラックを基準にタイミング レポートを並べ替えます。

[Advanced] タブ

[Report Timing] ダイアログ ボックスの [Advanced] タブでは、ピンおよびネットの詳細、レポートの出力ディレクトリ、コマンド エラーの処理などを指定します。256 ページの図 7-15 に、[Advanced] タブを示します。

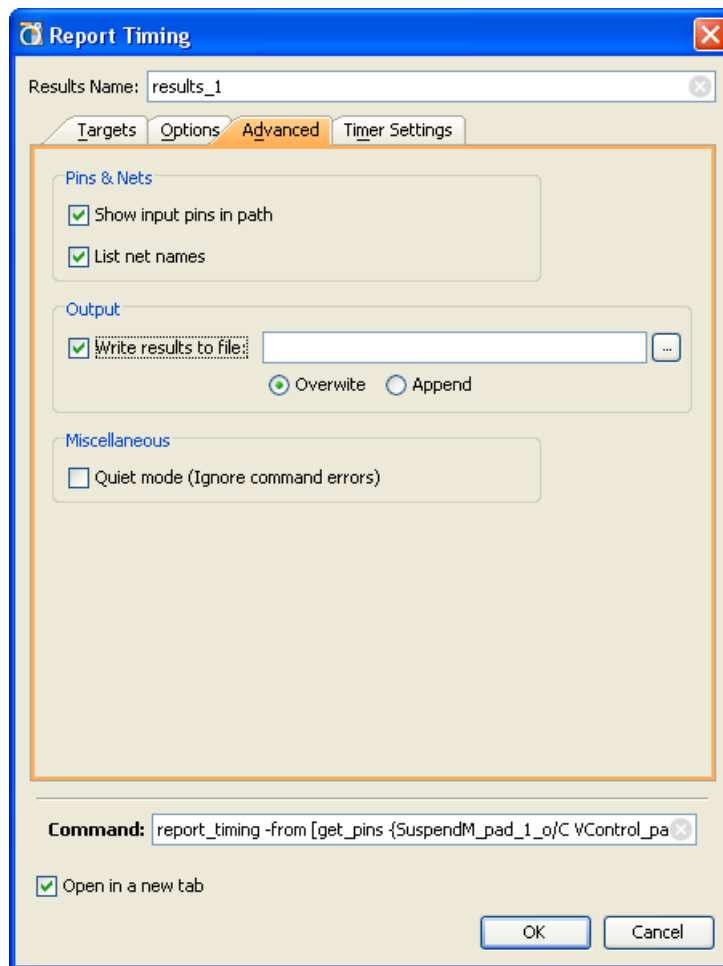


図 7-15 : [Report Timing] ダイアログ ボックスの [Advanced] タブ

[Advanced] タブには、次のオプションが含まれます。

- [Show input pins in path] : 各パスの開始点の入力ピンを表示します。
- [List net names] : 各パス エレメントのネット接続名をリストします。
- [Write results to file] : タイミング レポートの結果を指定したファイルに書き出します。
 - [Overwrite] : 指定したファイルと同じ名前のファイルを上書きします。
 - [Append] : タイミング レポートの詳細を指定ファイルに追加します。
- [Quiet mode] : タイミング レポートにコマンド オプションに関するエラー メッセージを含めないようにします。

[Timer Settings] タブ

[Report Timing] ダイアログ ボックスの [Timer Settings] タブでは、タイミング レポートを生成する際にタイミング エンジンで使用する遅延パラメーターを指定します。257 ページの図 7-16 に、[Timer Settings] タブを示します。

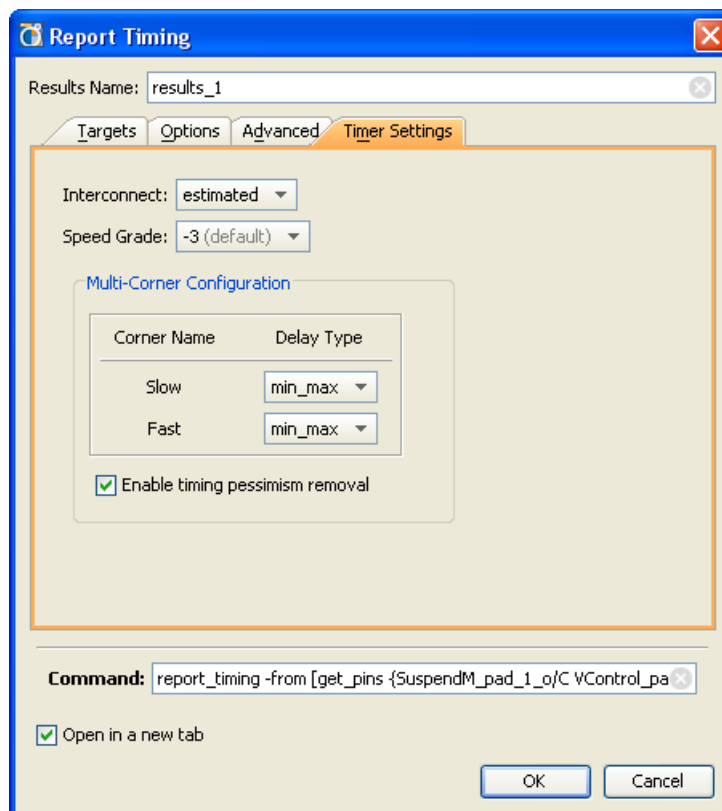


図 7-16 : [Report Timing] ダイアログ ボックスの [Timer Settings] タブ

このダイアログ ボックスには、次のようなオプションがあります。

- [Interconnect] : インターコネクト遅延に使用される遅延値のタイプを選択します。遅延値には、次のような選択肢があります。
 - [estimated] : インターコネクト値に予測遅延を使用します。
 - [none] : インターコネクト遅延を 0 に設定します。
- [Speed Grade] : タイミング解析で使用するデバイスのスピード グレードを選択します。このフィールドを使用すると、さまざまなデバイス スピード グレードを使用してデザインのタイミングを予測できます。
- [Multi-corner analysis] : マルチコーナー解析では、さまざまなプロセスおよび動作状況を同時に使用して、ワースト ケースのセットアップとホールド解析が実行されます。これにより、最小または最大遅延のみを使用した解析よりもさらに正確な解析が実行されます。
 - [Slow] (Corner Name) : スロー コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - [none] : 遅延を使用しません。
 - [max] : セットアップおよびホールド解析に、クロックおよびデータ バスの最大遅延を使用します。
 - [min] : セットアップおよびホールド解析に、クロックおよびデータ バスの最小遅延を使用します。
 - [min_max] : セットアップおよびホールド解析に、クロックおよびデータ バスの最小遅延と最大遅延を組み合わせて使用します。

- [Fast] (Corner Name) : ファースト コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - [none] : 遅延を使用しません。
 - [max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最大遅延を使用します。
 - [min] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延を使用します。
 - [min_max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。
- [Enable timing pessimism removal] : さまざまなオンチップ遅延をモデリングする際に、ソース レジスタとデスティネーション レジスタ間の共通クロック パスで生成されたスキュー遅延を削除します。

タイミング結果の解析

PlanAhead のタイミング解析が終了すると、[Timing Results] ビューが開きます。図 7-17 に、[Timing Results] ビューの例を示します。[Timing Results] ビューは、インプリメント済みデザインを開き、TRCE を実行した場合にも表示されます。詳細は、380 ページの「タイミング結果の解析」を参照してください。

[Timing Results] ビューにパスを表示するには、PlanAhead のタイミング解析または ISE の TRCE ツールのいずれかを実行する必要があります。[Timing Results] ビューには、248 ページの「タイミング解析の実行」で説明されているように、[Run Timing Analysis] ダイアログ ボックスで指定された条件に一致するパスが含まれます。

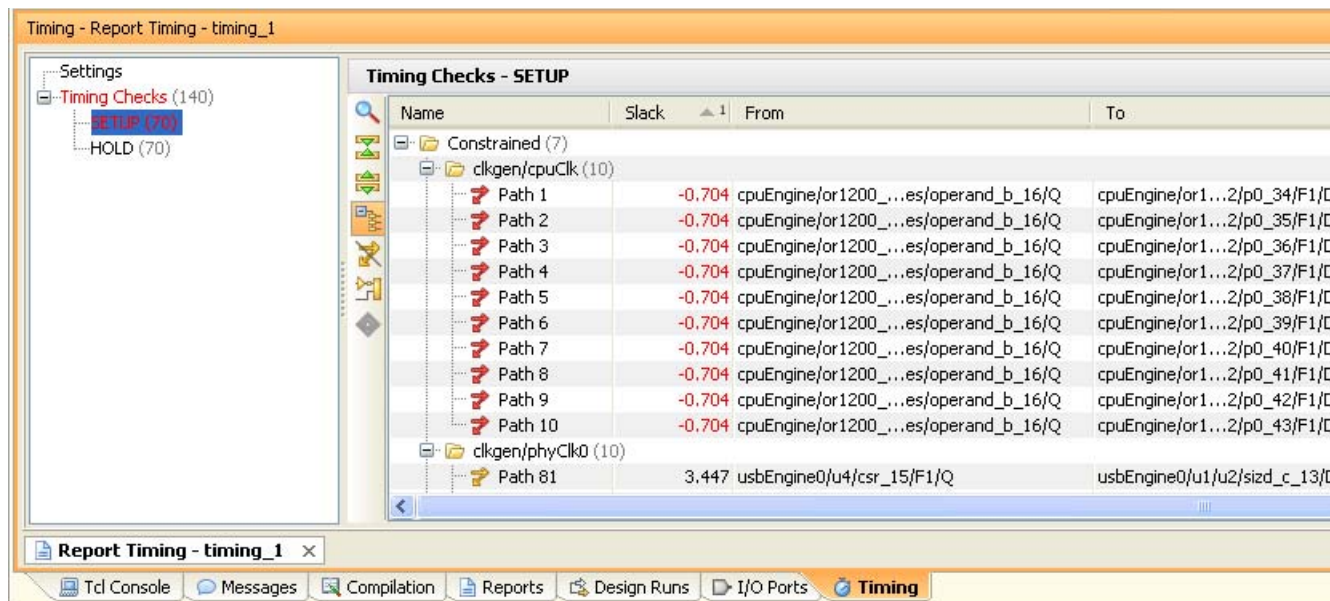


図 7-17 : [Timing Results] ビュー

[Timing Results] ビューでは、パスやインスタンスを確認、並べ替え、選択できます。次の情報が表示されます。

- [Settings] : [Report Timing] コマンドのサマリとタイミング解析が実行されたときに使用された設定が表示されます。

- [Timing Checks] : セットアップおよびホールド チェック別を並び替えて、タイミング結果を最初はタイミング パスのソース クロック別に並び替えた表形式で表示します。
 - [Name] : ソース クロックおよびレポートされるタイミング パスのシーケンシャル名を表示します。
 - [Slack] : パスの正または負のスラックの合計が表示されます。
 - [From] : パスのソース ピンが表示されます。
 - [To] : パスのデスティネーション ピンが表示されます。
 - [Total Delay] : パスで予測される遅延の合計が表示されます。
 - [Logic Delay] : パスのロジック エLEMENTの配置が起因の遅延のみが表示されます。
 - [Net Delay] : パスのインターコネクトの配置が起因の遅延のみが表示されます。
 - [Logic%] : ロジック エLEMENTの配置が起因の遅延の割合が表示されます。
 - [Net %] : インターコネクトの配置が起因の遅延の割合が表示されます。
 - [Stages] : ソースとデスティネーションが全体の遅延に影響を及ぼすパス上のインスタンスの合計が表示されます。
- 注記 : PlanAhead ツールでは、キャリー チェーン インターコネクトがロジックの各ステージとしてカウントされるので、レポートされるステージは ISE でレポートされるロジックレベルとは異なります。
- [Source Clock] : ソース クロックの名前が表示されます。
 - [Destination Clock] : デスティネーション クロックの名前が表示されます。

タイミング レポートの並び替え

[Timing Results] ビューのリストは、列ヘッダーをクリックすると並べ替えることができます。たとえば、[Stages] 列のヘッダーをクリックすると、ロジックのステージ順に並べ替えられます。列をもう一度クリックすると、逆の順序で並べ替えることができます。

並べ替えた後に **Ctrl** キーを押して別の列ヘッダーをクリックすると、さらにその列に基づいて並べ替えられます。並べ替えに使用する列の数に制限はありません。もう一度 **Ctrl** キーを押して列ヘッダーをクリックすると、その列の並べ替えが解除されます。

ツリー表形式のビューの詳細は、[127 ページの「ツリー表形式のビュー」](#)を参照してください。

パス リストのフラット化

デフォルトでは、パスは制約別に表示されます。[258 ページの図 7-17](#) に示すように、[Timing Results] ビューのツールバーで [Group by Constraint] をクリックすると、リストをフラットにし、すべてのパスを表示できます。[Group by Constraint] ボタンでは、制約別に分類されたパスのリストとフラット化されたパスのリストの表示を切り替えることができます。



パスの詳細の表示

リストでパスを選択すると、[Path Properties] ビューにそのパスの情報が表示されます。すべてのロジック エLEMENTの詳細な遅延情報が[260 ページの図 7-18](#) のように表示されます。

Path Properties

Path 7

Summary

Name	Path 7
Slack	-0.161
Source	usbEngine0/usb_dma_wb_in/BU2/U0/gen_fifo18_36.fgifo18_36/fblk/inst_few/k1[1].inst_fed/one_prim.inst_fifoprim/gfifo36.sngfifo36/fifo36_
Destination	usbEngine0/usbEngineSRAM/BU2/U0/blk_mem_generator/valid.cstr/ramloop[0].ram.r/v5.ram/SP.WIDE_PRIM18.SP
Requirement	3.800
Delay	3.961
Source Clock	usbClk (rising at 0.000ns)
Destination Clock	usbClk (rising at 3.800ns)

Source Clock Path

Delay Type	Delay	Cumulative	Location	PBlock	Logical Resource
net (fo=0)	0.000	0.000	AB19		usbClk
IBUFG	0.818	0.818	AB19		usbClk_ibuf/ibufg/I
net (fo=1)	0.000	0.818	BUFGCTRL_X0Y9		usbClk_ibuf/ibufg/O
BUFG	0.250	1.068	BUFGCTRL_X0Y9		usbClk_ibuf/ibufg/I
net (fo=407)	2.033	3.101	RAMB36_X2Y1	pblock_usbEngine0	usbClk_ibuf/ibufg/O
Total	3.101	3.101			usbEngine0/usb_dma_wb_in/BU2/U0/gen_fifo18_36.fgifo18_36

Data Path

Delay Type	Delay	Cumulative	Location	PBlock	Logical Resource
FIFO36_EXP	0.818	0.818	RAMB36_X2Y1	pblock_usbEngine0	usbEngine0/usb_dma_wb_in/BU2/U0/gen_fifo18_36.fgifo18_36
net (fo=40)	1.520	2.338	SLICE_X19Y8	pblock_usbEngine0	usbEngine0/u2/wsel/I/O
LUT6	0.094	2.432	SLICE_X19Y8	pblock_usbEngine0	usbEngine0/u2/wsel/O
net (fo=42)	0.381	2.813	SLICE_X18Y9	pblock_usbEngine0	usbEngine0/u2/sram_we/I4
LUT6	0.094	2.907	SLICE_X18Y9	pblock_usbEngine0	usbEngine0/u2/sram_we/O
net (fo=4)	0.430	3.337	RAMB36_X1Y1	pblock_usbEngine0	usbEngine0/usbEngineSRAM/BU2/U0/blk_mem_generator/valid
RAMB18	0.624	3.961	RAMB36_X1Y1	pblock_usbEngine0	usbEngine0/usbEngineSRAM/BU2/U0/blk_mem_generator/valid
Total	3.961	3.961			

Destination Clock Path

Delay Type	Delay	Cumulative	Location	PBlock	Logical Resource
net (fo=0)	0.000	0.000	AB19		usbClk
IBUFG	0.818	0.818	AB19		usbClk_ibuf/ibufg/I
net (fo=1)	0.000	0.818	BUFGCTRL_X0Y9		usbClk_ibuf/ibufg/O
BUFG	0.250	1.068	BUFGCTRL_X0Y9		usbClk_ibuf/ibufg/I
net (fo=407)	2.033	3.101	RAMB36_X1Y1	pblock_usbEngine0	usbClk_ibuf/ibufg/O
Total	3.101	3.101			usbEngine0/usbEngineSRAM/BU2/U0/blk_mem_generator/valid

General **Report** Instances Options

図 7-18 : [Path Properties] ビューの [Report] タブ

このレポートのフォーマットは TRCE レポートに似ています。

- デフォルトでは、パスを選択するとそのパスに含まれるインスタンスもすべて選択されます。
- レポートにリンクが表示されているオブジェクトをクリックすると、[Netlist] や [Device] など、ほかのビューでもそのオブジェクトが選択されます。
- オブジェクトを選択するときに、Shift キーまたは Ctrl キーを使用すると、複数のパスを選択できます。
- 選択されたパスすべてのインスタンスすべてが選択されますが、[Path Properties] ビューには最初に選択されたパスの情報のみが表示されます。

タイミング パス レポートの表示

タイミング パス レポートもワークスペースに表示して見やすくできます。この表示は [Path Properties] ビューと同じ情報になりますが、同時に複数のパスを表示できる点が異なります。[Path Properties] ビューには、最後に選択したパスのプロパティが表示されます。

タイミング パス レポートを表示するには、次の手順に従います。

1. タイミング パスを選択します。
2. 右クリックし、[View Path Report] をクリックします。

スラック ヒストグラムの使用

スラック ヒストグラムには、デザインのタイミング遅延が視覚的に表示されます。このビューを使用すると、デザインがパフォーマンス要件を満たしていない場合に、次に何をすべきかを判断しやすくなります。

スラック ヒストグラムには、エンドポイントで算出されたタイミング スラックが表示されます。完全なタイミング パスは表示されません。[Schematic] コマンドを実行すると、パスのエンドポイントが表示されます。完全なパスのタイミングを表示するには、[Report Timing] コマンドを使用する必要があります。

1. [Tools] → [Timing] → [View Slack Histogram] をクリックし、スラック ヒストグラムを生成します。
2. 262 ページの図 7-19 のように、[Generate Slack Histogram] ダイアログ ボックスが開きます。
 - [Histogram name] : 生成するヒストグラム レポートの名前を指定します。
 - [Options] タブ : レポートをカスタマイズします。タブの詳細は、262 ページの「スラック ヒストグラムのオプション設定」で説明します。
 - [Timer Settings] タブ : レポートの生成に使用されるタイミング エンジンと遅延オプションを指定します。タブの詳細は、266 ページの「スラック ヒストグラムのタイマー設定」で説明します。
 - [Command] : スラック ヒストグラムの生成オプションで生成される Tcl コマンドを表示します。
3. [OK] をクリックすると、指定した値でスラック ヒストグラムが表示されます。

スラック ヒストグラムのオプション設定

スラック ヒストグラムを生成する際に使用されるオプションは、[Generate Slack Histogram] ダイアログ ボックスの [Options] タブで指定できます。図 7-19 に、[Options] タブを示します。

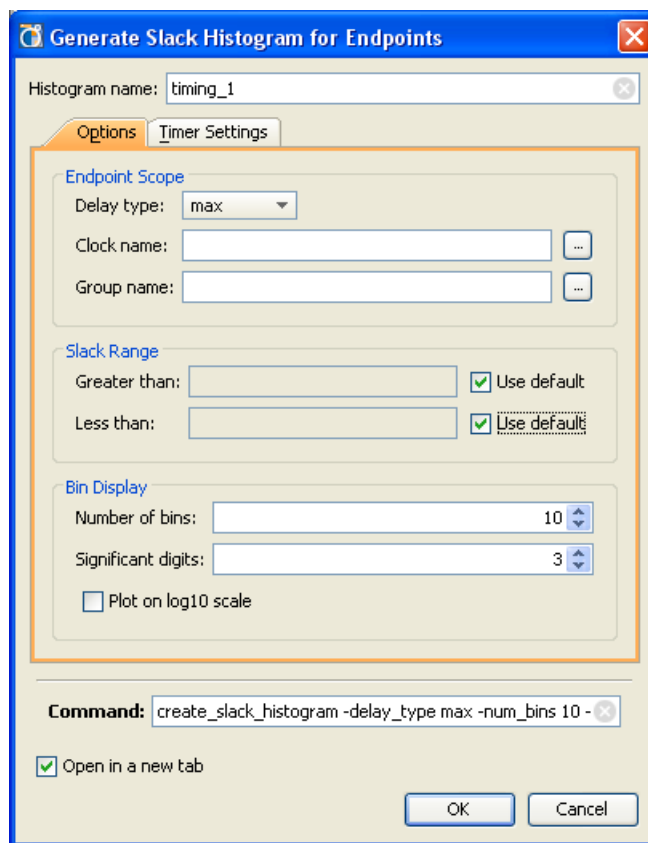


図 7-19 : [Generate Slack Histogram for Endpoints] ダイアログ ボックス : [Options] タブ

このダイアログ ボックスには、次のようなオプションがあります。

- **[Endpoint Scope]** : スラック ヒストグラムの生成で使用するエンドポイントと遅延タイプを指定します。エンドポイントをクロック名およびグループ名に基づいてフィルターすると、特定のパスに焦点を当てたヒストグラムを生成できます。このフィールドには、次のオプションが含まれています。
 - **[Delay type]** : スラック ヒストグラムの生成に使用する遅延値を指定します。
 - **[min]** : スラック ヒストグラムのクロックおよびデータ パスの最小遅延を使用します。
 - **[max]** : スラック ヒストグラムのクロックおよびデータ パスの最大遅延を使用します。
 - **[min_max]** : スラック ヒストグラムのクロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。
- **[Clock name]** : 関連するクロック名に基づいてエンドポイントをフィルターします。このフィールドの値は、直接入力するか、[Choose Endpoint Destination Clocks] ダイアログ ボックスで入力します。

- **[Group name]** : 関連するグループ名に基づいてエンドポイントをフィルターします。このフィールドの値は、直接入力するか、**[Choose Endpoint Path Groups]** ダイアログ ボックスで入力します。
- **[Slack Range]** : スラック値に基づいてエンドポイントをフィルターします。特定のスラック値に基づいてエンドポイントをフィルターすると、問題のパスに焦点をあてたヒストグラムを生成できます。このフィールドには、次のオプションが含まれています。
 - **[Greater than]** : ヒストグラムに含めるパスの最小スラック値を指定します。
 - **[Less than]** : ヒストグラムに含めるパスの最大スラック値を指定します。
- **[Bin Display]** : ヒストグラムをさらに詳細にカスタマイズします。次のオプションがあります。
 - **[Number of bins]** : 表示するヒストグラムの棒グラフの数を指定します。**PlanAhead** で検出されるスラック値の範囲が指定した棒グラフ数に分割されます。小さい値を指定すると、ヒストグラムにはタイミング パフォーマンスの一般的なビューが表示されます。棒の数が多いた方が、特定のスラック範囲内での遅延パフォーマンスを詳細に表示できます。
 - **[Significant digits]** : スラック値をレポートするのに使用する有効桁数を指定します。デフォルト値は 3 です。
 - **[Plot on Log10 scale]** : スラック ヒストグラムの Y 軸を対数目盛 (**Log10**) または均等目盛のどちらにするかを指定します。棒の数がほかのヒストグラムに比べて少ない場合、均等目盛では見づらいことがあります。このような場合、対数目盛を使用すると便利です。この設定は、ヒストグラムを表示した後、ツールバーの **[Plot histogram on log10 scale]** コマンドから切り替えることもできます。
- **[Command]** : スラック ヒストグラムのコマンドを実行するために使用される **Tcl** コマンドを表示します。

エンドポイントのデスティネーション クロックの選択

[Choose Endpoint Destination Clocks] ダイアログ ボックスを使用すると、特定のエンドポイントのクロック ドメインを選択できます。エンドポイントは、指定したクロック名でフィルターされます。図 7-20 に、**[Choose Endpoint Destination Clocks]** ダイアログ ボックスを示します。

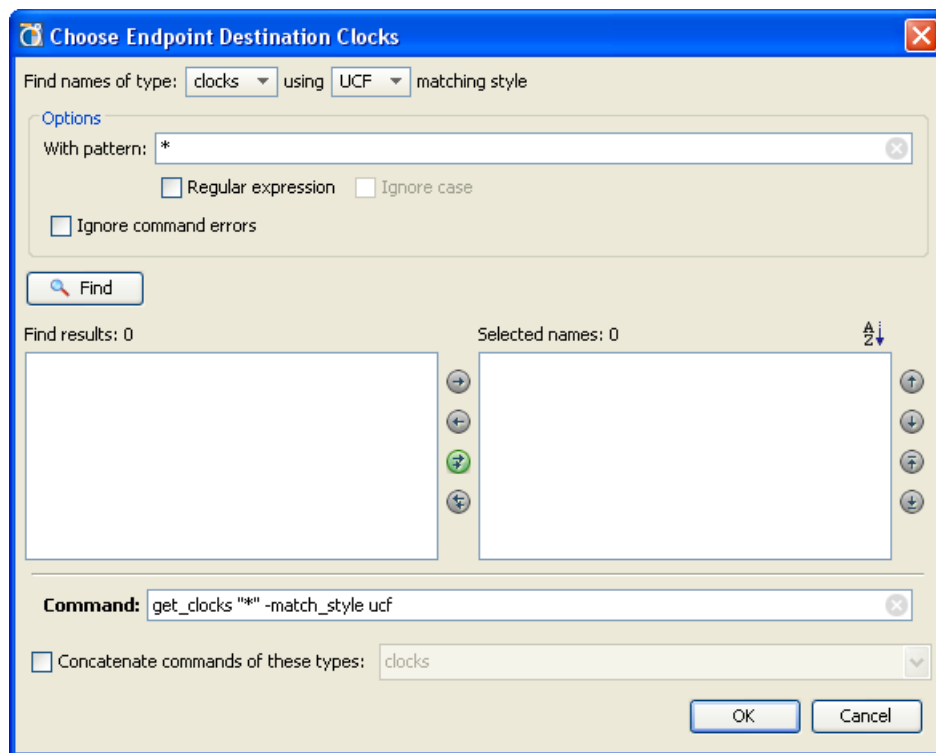


図 7-20 : [Choose Endpoint Destination Clocks] ダイアログ ボックス

このダイアログ ボックスには、次のようなオプションがあります。

- [Find names of type] : 指定したクロック ドメインでエンドポイントをフィルターします。
 - [matching style] : デザイン エレメントのフィルターに使用するパターン一致のタイプを選択します。このフィールドには、次のオプションがあります。
 - [UCF] : パターン一致に UCF ベースの構文を使用します。
 - [SDC] : パターン一致に SDC ベースの構文を使用します。
 - [With pattern] : デザイン エレメントのフィルターに使用するパターンを入力します。このフィールドは、次のオプションを使用して変更できます。
 - [Regular expression] : 検索文字列に正規表現の構文を使用します。
 - [Ignore case] : 大文字/小文字を区別せずに文字列を検索します。
 - [Filter matching name with expression] : SDC 構文のオプションで、-filter コマンドを指定します。
- 注記：このフィールドは [matching style] を [SDC] に設定している場合のみ表示されます。
- [Ignore command errors] : Tcl コマンドでタイミング レポートを生成中に出力される警告メッセージが表示されないようにします。
 - [Find] : 定義したパターンに基づいて検索を実行します。
 - [Find results] : オブジェクト検索の結果を表示します。
 - [Move the selected items to the right] : 選択したオブジェクトを [Find results] から [Selected names] に移動します。

- [Move the selected items to the left] : 選択したオブジェクトを [Selected names] から [Find results] に移動します。
- [Move all the items to the right] : [Find results] のすべてのオブジェクトを [Selected names] に移動します。
- [Move all the items to the left] : [Selected names] のすべてのオブジェクトを [Find results] に移動します。
- [Selected names] : 現在選択されているデスティネーション クロックを表示します。
- [Command] : クロックを選択する Tcl コマンドを表示します。

エンドポイントのパス グループの選択

[Choose Endpoint Path Groups] ダイアログ ボックスを使用すると、エンドポイントをフィルターするためのグループ名を選択できます。図 7-21 に、[Choose Endpoint Path Groups] ダイアログ ボックスを示します。

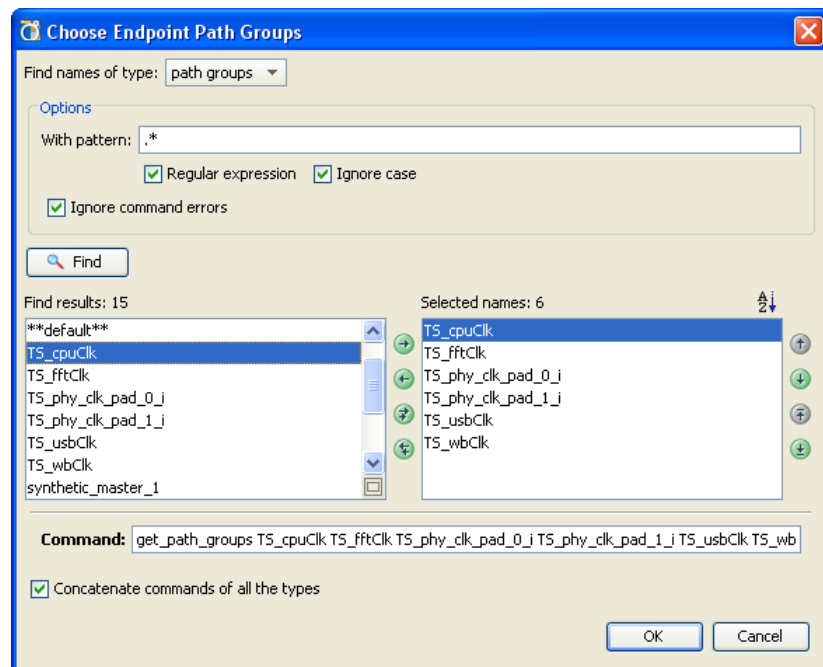


図 7-21 : [Choose Endpoint Path Groups] ダイアログ ボックス

このダイアログ ボックスには、次のようなオプションがあります。

- [Find names of type] : 指定したパス グループでエンドポイントをフィルターします。
- [With pattern] : デザイン エレメントのフィルターに使用するパターンを定義します。このフィールドは、次のオプションを使用して変更できます。
 - [Regular expression] : 検索文字列に正規表現の構文を使用します。
 - [Ignore case] : 大文字/小文字を区別せずに文字列を検索します。
- [Ignore command errors] : Tcl コマンドでタイミング レポートを生成中に出力される警告メッセージが表示されないようにします。
- [Find] : 定義したパターンに基づいて検索を実行します。
- [Find results] : オブジェクト検索の結果を表示します。

- [Move the selected items to the right] : 選択したオブジェクトを [Find results] から [Selected names] に移動します。
- [Move the selected items to the left] : 選択したオブジェクトを [Selected names] から [Find results] に移動します。
- [Move all the items to the right] : [Find results] のすべてのオブジェクトを [Selected names] に移動します。
- [Move all the items to the left] : [Selected names] のすべてのオブジェクトを [Find results] に移動します。
- [Selected names] : 現在選択されているパス グループを表示します。
- [Command] : オブジェクトを選択する Tcl コマンドを表示します。

スラック ヒストグラムのタイマー設定

スラック ヒストグラムのタイミング エンジンで使用する遅延パラメーターは、[Generate Slack Histogram] ダイアログ ボックスの [Timer Settings] タブで指定できます。図 7-22 に、[Timer Settings] タブを示します。

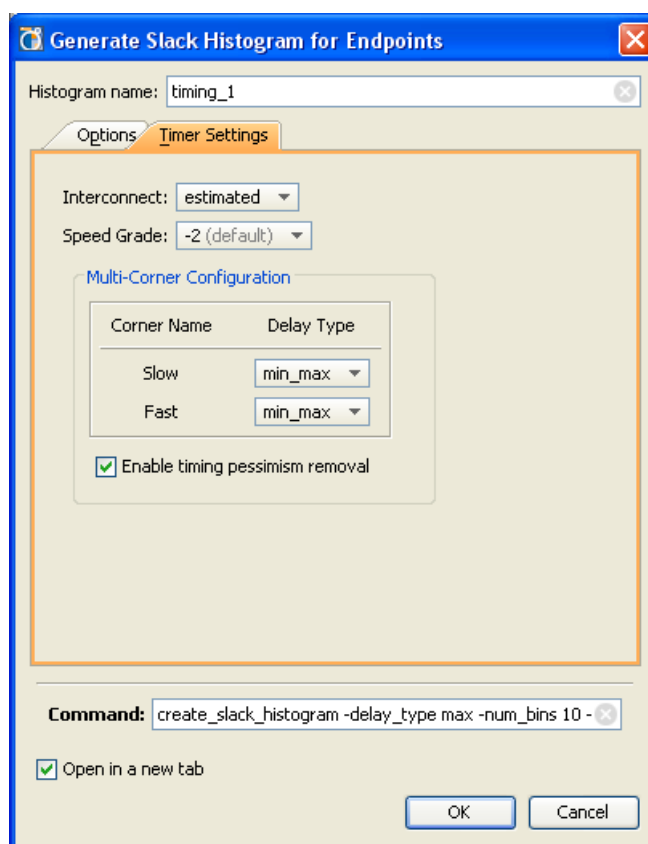


図 7-22 : [Generate Slack Histogram for Endpoints] ダイアログ ボックス : [Timer Settings] タブ

次のオプションがあります。


- [Interconnect] : インターコネクト遅延に使用される遅延値のタイプを選択します。指定できる遅延値は次のとおりです。

- **[estimated]**: インターコネクト値に予測遅延を使用します。
- **[none]**: インターコネクト遅延を 0 に設定します。
- **[Speed Grade]**: タイミング解析で使用するデバイスのスピード グレードを選択します。このフィールドを使用すると、さまざまなデバイス スピード グレードを使用してデザインのタイミングを予測できます。
- **[Multi-corner analysis]**: マルチコーナー解析では、さまざまなプロセスおよび動作状況を同時に使用して、ワースト ケースのセットアップとホールド解析が実行されます。これにより、最小または最大遅延のみを使用した解析よりもさらに正確な解析が実行されます。
- **[Slow] (Corner Name)**: スロー コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - **[none]**: 遅延を使用しません。
 - **[max]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最大遅延を使用します。
 - **[min]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延を使用します。
 - **[min_max]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。
- **[Fast] (Corner Name)**: ファースト コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - **[none]**: 遅延を使用しません。
 - **[max]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最大遅延を使用します。
 - **[min]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延を使用します。
 - **[min_max]**: セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。
- **[Enable timing pessimism removal]**: さまざまなオンチップ遅延をモデリングする際に、ソースレジスタとデスティネーションレジスタ間の共通クロックパスで生成されたスキュー遅延を削除します。

スラック ヒストグラム結果の解析

ヒストグラムが生成されたら、その結果を使用してデザインにどのようなタイミング問題があるのかを確認できます。**[Histogram]** ビューには、遅延のグラフと、そ下の表にパスのエンドポイントが表示されます。[268 ページの図 7-23](#) に、スラック ヒストグラムの例を示します。

ヒストグラムでは、エンドポイントを選択およびフィルターできます。選択したエンドポイントがヒストグラムの表に表示されます。ヒストグラムの棒はフィルターでは変更できません。エンドポイントは、次のように選択およびフィルターできます。

- **[Select one or more bins]**: ヒストグラムで棒グラフの棒部分をクリックすると、それが選択され、その棒部分のエンドポイントがリストされます。複数の棒部分を選択する場合は、**Ctrl** を押しながらクリックしていきます。表がアップデートされて、選択した棒のエンドポイントが表示されます。
- **[Filter bars]**: ヒストグラム ツールバーの **[Filter bars]** をクリックすると、ヒストグラムでドラッグして四角で囲むことで、1 つまたは複数の棒を選択できます。棒の一部のみ 

を選択すると、棒全体が選択され、選択した棒に含まれるエンドポイントすべてがヒストグラムの表に表示されます。

- [Unfilter All] : ヒストグラム ツールバーの [Unfilter All] をクリックすると、棒の選択が解除され、表にすべてのエンドポイントが表示されます。

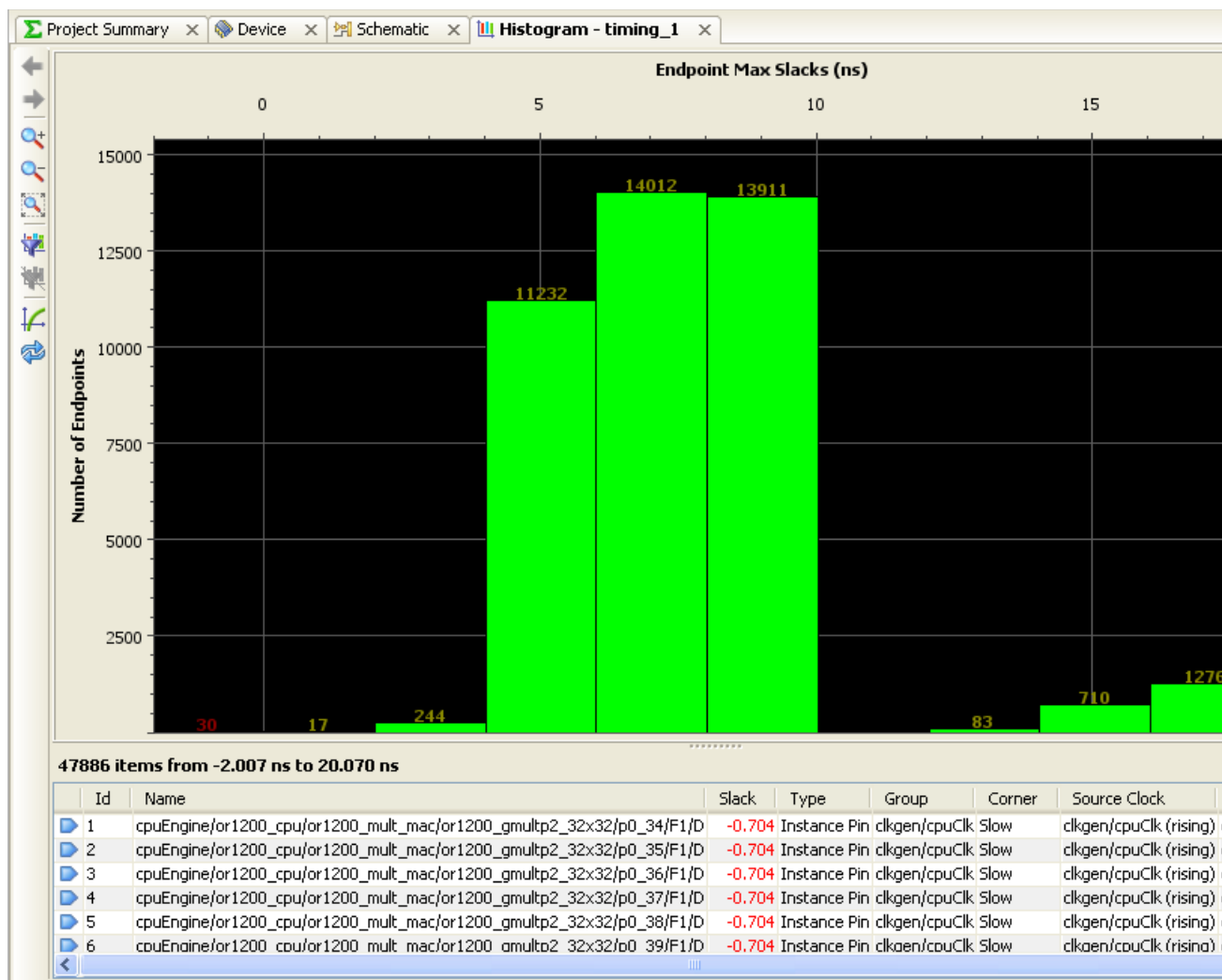


図 7-23 : スラック ヒストグラム

Y 軸の目盛を対数目盛または均等目盛に切り替えるには、[Plot histogram on log10 scale] をクリックします。



269 ページの図 7-24 に、対数目盛 (Log10) で表示したスラック ヒストグラムを示します。

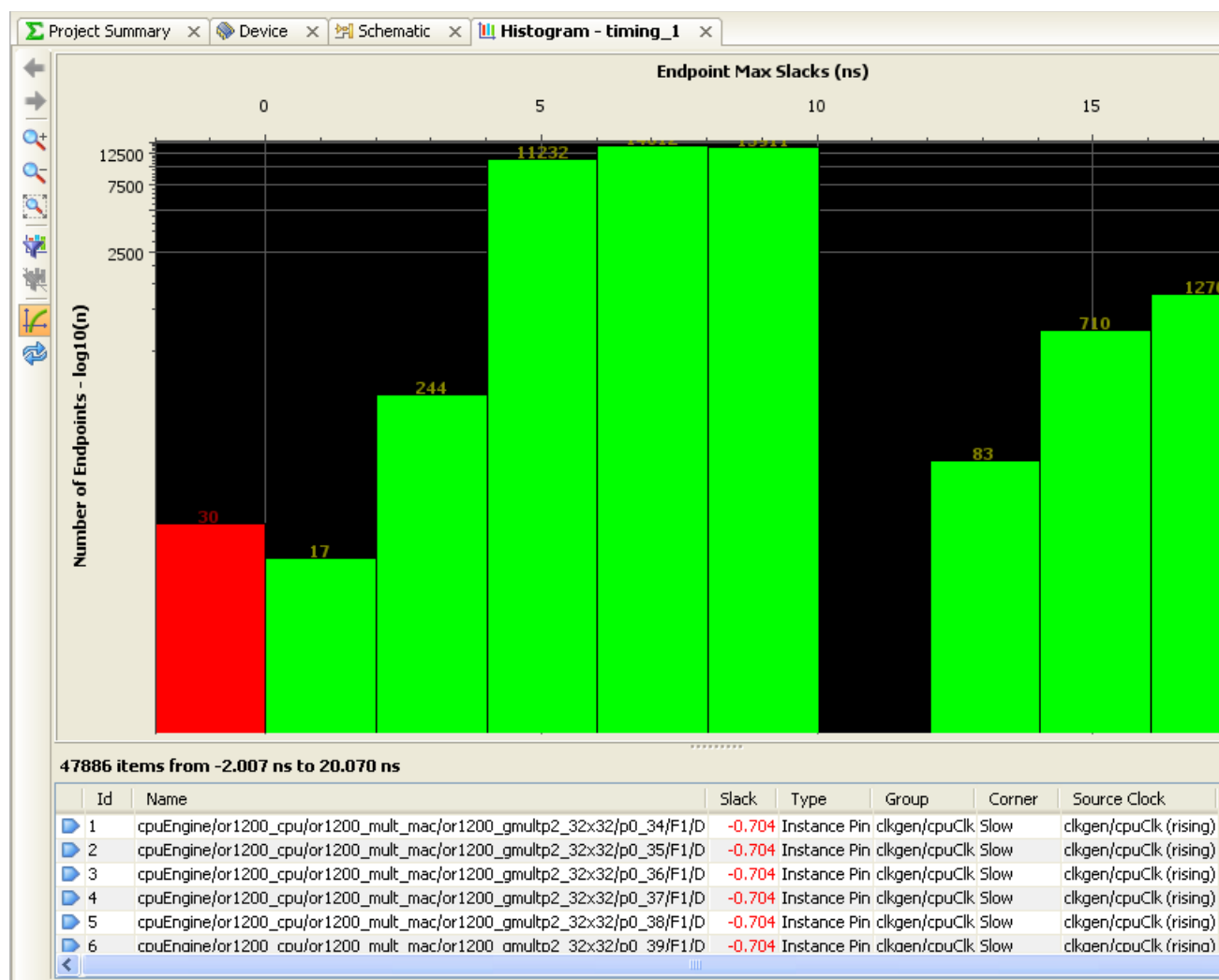


図 7-24 : スラック ヒストグラム (対数目盛)

ヒストグラムのポップアップ メニュー コマンド

ヒストグラムまたはエンドポイントを右クリックをすると、ポップアップ メニューにヒストグラムおよびエンドポイントの表示を操作するコマンドが表示されます。ポップアップ メニューに含まれるコマンドは、次のとおりです。

- **[Report Timing]** : 選択した棒のエンドポイントに関するタイミング レポートを生成します。タイミングは選択されているパスのエンドポイントに対して生成されますが、**[Report Timing]** ダイアログ ボックスのオプションに基づいて、完全なパスのタイミング解析が実行されます。

注記 : **[Report Timing]** コマンドは、1 つまたは複数の棒が選択された状態の、ヒストグラムのポップアップ メニューでのみ使用できます。選択した棒に含まれるパスのエンドポイントが多いために実行に時間がかかる場合は、それを示す警告メッセージが表示されます。

- **[Refresh Histogram]** : **[Generate Slack Histogram for Endpoints]** ダイアログ ボックスが開きます。ここで新しいオプションを指定して、スラック ヒストグラムを生成し直すことができます。
- **[View]** : ヒストグラムの表示を変更するコマンドのサブメニューが表示されます。このサブメニューには、次のオプションがあります。
 - **[Zoom In]** : 拡大します。

- [Zoom Out] : 縮小します。
- [Zoom Fit] : ヒストグラム全体を表示します。
- [Zoom Area] : 選択したエリアを全体に表示します。
- [Options] : [PlanAhead Options] ダイアログ ボックスを表示します。

クロックの相互関係の解析

大型の複雑な FPGA デザインでは、データが頻繁に 1 つのクロック ドメインから別のクロック ドメインに転送されます。メタスタビリティ、データ損失、非干渉性などの問題を特定するため、複数のクロック ドメインを通過するパスを可視化できると有益です。PlanAhead のクロックの相互関係レポートには、クロック間の相互関係と複数のクロック ドメインを通過する信号に関する情報が表示されます。

1 つのクロック ドメインに含まれるフリップフロップと、別のクロック ドメインに含まれるロジックの間のパスは、2 つのクロック ドメイン間に制約が適用されていなければ、ISE または PlanAhead のタイミング ツールでは自動的に解析されません。制約が適用されていないパスは、解析されないうままになります。クロックの相互関係レポートは、回路のトポロジとデザイン制約をチェックし、複数のクロック ドメインを通過するパスに制約が適用されているかどうかをレポートします。このレポートを使用することで、制約が適切に適用されているパス、制約が適用されていないパス、無視しても問題ないフォールス パスを特定できます。また、RTL コードで明示的に記述されていない信号が、何かの理由で複数のクロック ドメインを通過しており、必要な制約が適用されていない場合にも、そのような信号を特定できます。

クロックの相互関係のレポート

合成済みデザインを PlanAhead で開き、次を実行します。

- メイン メニューから [Tools] → [Timing] → [Report Clock Interaction] をクリックします。
- Flow Navigator の [Synthesis] メニューから [Report Clock Interaction] をクリックします。

271 ページの図 7-25 に示すような多くのフィールドとオプションが表示された [Report Clock Interaction] フォームが開きます。

- [Results Name] : 出力されるレポートの名前を指定します。
- [Options] タブ : クロックの相互関係レポートをカスタマイズできます。このタブのオプションについては、「[クロックの相互関係レポートのオプション設定](#)」を参照してください。
- [Timer Settings] タブ : レポートの生成に使用されるタイミング エンジンと遅延オプションを指定します。このタブのオプションについては、272 ページの「[\[Timer Settings\] タブ](#)」を参照してください。

- [Command] : [Report Clock Interaction] オプションで生成された Tcl コマンドのテキストを含みます。

注記：このフィールドは編集できます。実行される Tcl コマンドが変更されますが、ダイアログ ボックスで指定されるオプションは変更されません。

- [Open in a new tab] : オンにすると、クロックの相互関係レポートが新しいウィンドウで表示されます。オフにすると、表示されていたウィンドウが閉じてから新しい結果が表示されます。

クロックの相互関係レポートのオプション設定

図 7-25 に示す [Options] タブでは、次の 3 つのオプションを指定します。

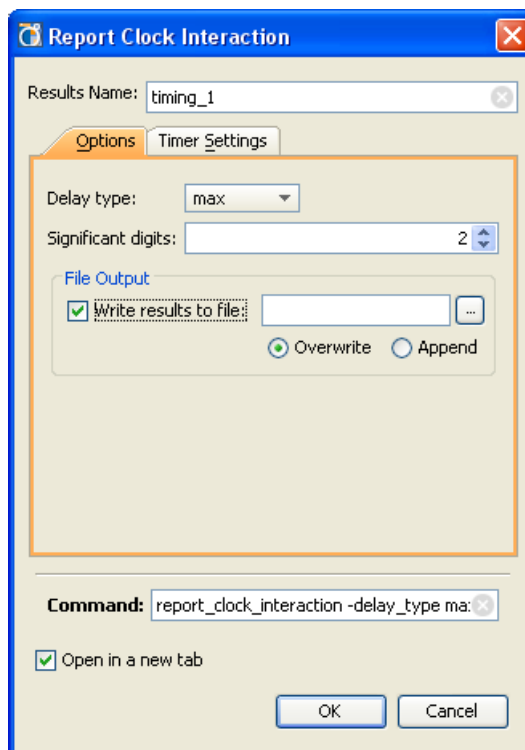


図 7-25 : [Report Clock Interaction] ダイアログ ボックスの [Options] タブ

- [Delay Type] : クロックの相互関係解析で使用する遅延タイプを指定します。このフィールドには、次のオプションが含まれます。
 - [max] : セットアップおよびホールド解析に、クロックおよびデータパスの最大遅延を使用します。
 - [min] : セットアップおよびホールド解析に、クロックおよびデータパスの最小遅延を使用します。
 - [min_max] : セットアップおよびホールド解析に、クロックおよびデータパスの最小遅延と最大遅延を組み合わせて使用します。
- [Significant digits] : クロックの相互関係レポートの有効桁数を指定します。有効な設定値は 0 ~ 13 です。
- [Write Results to File] : 結果を保存するファイルを指定します。
 - [Overwrite] : 該当ファイルが存在する場合は新しいファイルで上書きします。
 - [Append] : 該当ファイルに対するレポート結果を追加します。

[Timer Settings] タブ

図 7-26 に示す [Timer Settings] タブでは、次の 3 つのオプションを指定します。

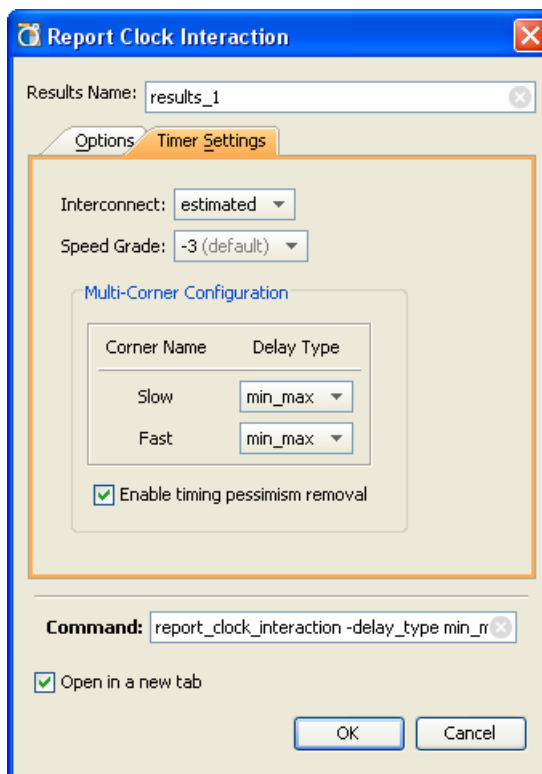


図 7-26 : [Report Clock Interaction] ダイアログ ボックスの [Timer Settings] タブ

- [Interconnect] : インターコネクト遅延に使用される遅延値のタイプを選択します。遅延値には、次のような選択肢があります。
 - [estimated] : インターコネクト値に予測遅延を使用します。
 - [none] : インターコネクト遅延を 0 に設定します。
- [Speed Grade] : タイミング解析で使用するデバイスのスピード グレードを選択します。このフィールドを使用すると、さまざまなデバイス スピード グレードを使用してデザインのタイミングを予測できます。
- [Multi-corner analysis] : マルチコーナー解析では、さまざまなプロセスおよび動作状況を同時に使用して、ワースト ケースのセットアップとホールド解析が実行されます。これにより、最小または最大遅延のみを使用した解析よりもさらに正確な解析が実行されます。
 - [Slow] (Corner Name) : スロー コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - [none] : 遅延を使用しません。
 - [max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最大遅延を使用します。
 - [min] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延を使用します。
 - [min_max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。

- [Fast] (Corner Name) : ファースト コーナー解析に使用される遅延タイプを選択します。指定できる値は次のとおりです。
 - [none] : 遅延を使用しません。
 - [max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最大遅延を使用します。
 - [min] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延を使用します。
 - [min_max] : セットアップおよびホールド解析に、クロックおよびデータ パスの最小遅延と最大遅延を組み合わせて使用します。
- [Enable timing pessimism removal] : さまざまなオンチップ遅延をモデリングする際に、ソースレジスタとデスティネーションレジスタ間の共通クロックパスで生成されたスキュー遅延を削除します。

クロックの相互関係マトリックスの表示

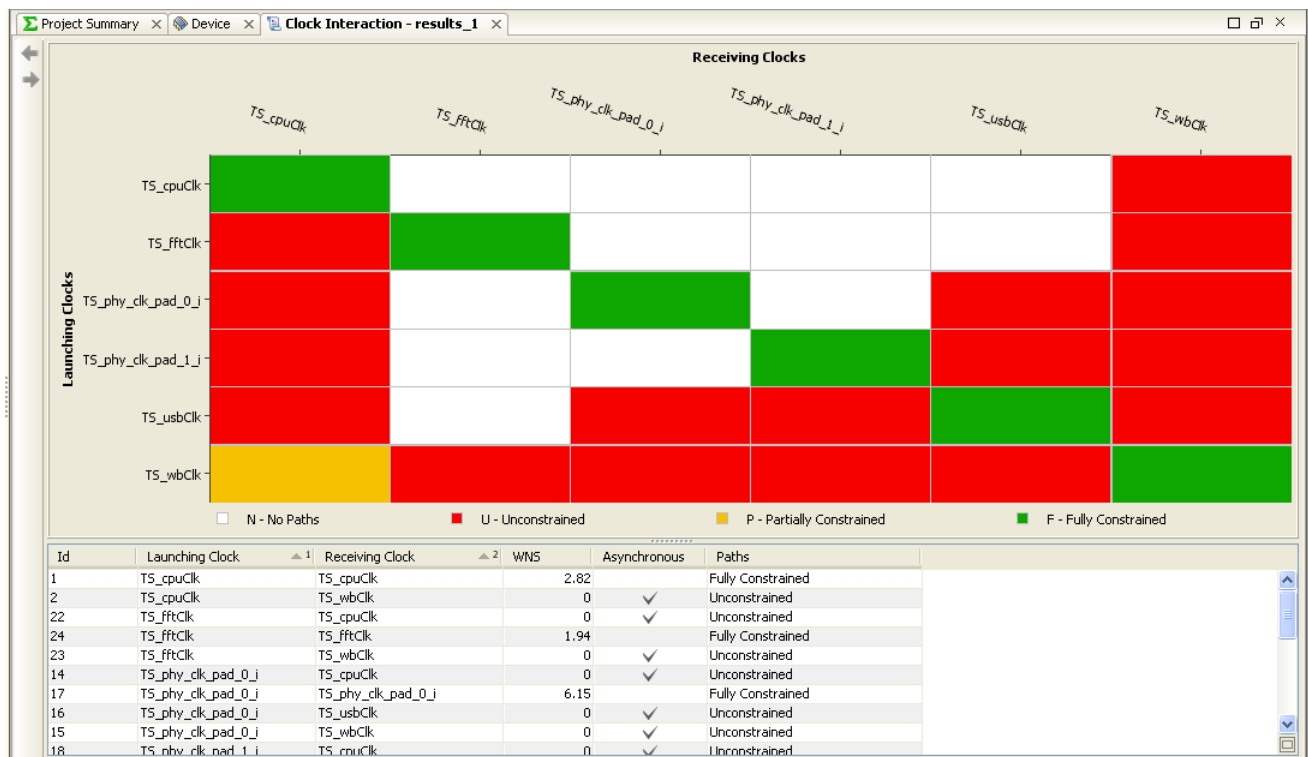


図 7-27 : クロックの相互関係レポート

[Report Clock Interaction] コマンドを実行すると、結果が [Clock Interaction] ビューに表示されます。クロックの相互関係レポートには、信号の起点クロックドメインを縦軸に、終点クロックドメインを横軸に、クロックドメインのマトリックスが表示されます。クロックドメインの各組み合わせのワーストケーススラックが、マトリックスの下に表で示されます。図 7-27 に、クロックの相互関係レポートの例を示します。

図 7-27 には、PlanAhead で検出された 6 つのクロックドメインが示されており、各クロックがその他すべてのクロックドメインに対して解析され、それらのドメイン間にまたがるパスがあるかどうか調べられます。2 つのクロックドメイン間の制約の状態もレポートされます。制約の状態は、この 6 x 6 のマトリックスに次のように示されます。

- 白 (または黒⁽¹⁾) のセルは、2 つのクロック ドメインにまたがるパスがないことを示します。この場合、何もレポートされません。
- 緑色のセルは、パスに適切に制約が適用されていることを示します。すべてのパスが FROM:TO 制約を使用して制約されています。

注記： 起点クロック ドメインと終点クロック ドメインが同じセルは、常に緑で示されます。

- オレンジ色または黄色のセルは、クロック ドメインをまたがるパスの一部が適切に制約が適用されていることを示します。そのロジックを通るパスの中には制約が正しく適用されているものと、されていないものがあります。これは、RTL でパスが明示的に記述されいなかったために、合成済みデザインで適切に制約が適用されなかったことが原因の可能性があります。
- 赤色のセルは、クロック関係の制約が付いていないことを示します。この場合、クロック ドメインをまたがるどのパスにも、必要な FROM:TO 制約が設定されていません。

このマトリックスに、緑または黒が示されているのが理想的です。赤色または黄色のセルは、タイミングの問題が発生する可能性があることを示します。マトリックスの色は、クロック ドメイン間の制約の状態を表しており、クロック ドメイン間のパスのタイミングを表すものではありません。緑色のセルはタイミングが満たされていることを示しているのではなく、クロック ドメイン間のタイミング パスに正しく制約が付いていることのみを示します。タイミングの状態は、マトリックスの下を表に示されます。

ツールバーの [Clock Interaction View Layers] をクリックし、特定のクロックのみを表示するようにクロックをフィルターします。これにより、表示されるクロック数は減りますが (図 7-28)、マトリックスの下を表にレポートされるクロック相互関係の数は減りません。

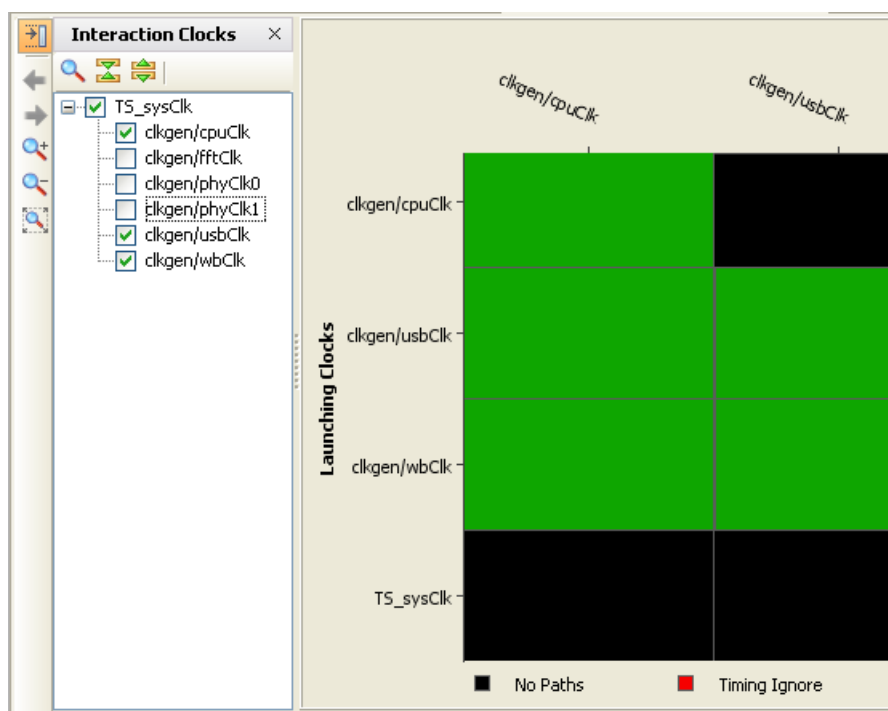


図 7-28 : [Report Clock Interaction] ダイアログ ボックスでクロックをフィルター表示

1. セルの色は、[Tools] → [Options] をクリックして表示される [PlanAhead Options] ダイアログ ボックスの [Theme] ページで Graphical Editors に指定されている背景色によって異なります。詳細は、第 4 章の「表示環境の設定」を参照してください。

クロック ドメインの各組み合わせに対してワースト ケースの負のスラックが表示されます。この表には、マトリックスには表示されない詳細な情報が表示されます。クロック ドメイン間にパスが存在しない場合 (マトリックスで白または黒で表示されているセル)、表には表示されないため、表に示される結果はセルの数より少ないことがあります。

表のデータは、列ヘッダーをクリックして並べ替えることができます。1 回クリックすると昇順に、2 回目にクリックすると降順に並べ替えられ、3 回目にクリックすると並べ替えがオフになり、元の状態に戻ります。1 つの列で並べ替えた後、2 番目の列でさらに並び替えるには、**Ctrl** キーを押しながら 2 番目の列ヘッダーをクリックします。表の操作については、第 4 章の「ツリー表形式のビュー」を参照してください。

マトリックスでセルを選択すると、下の表で対応する行が選択されます。逆に、表で行を選択すると、上のマトリックスで対応するセルがハイライトされます。

表には次の列があります。

- **[Id]** : 表示されているパスの識別番号を示します。
- **[Launching Clock]** : パスが開始するクロック ドメインを示します。
- **[Receiving Clock]** : パスが終了するクロック ドメインを示します。
- **[WNS]** : クロック ドメイン間のさまざまなパスのワースト ケースのスラックを表示します。スラックは、要件の時間と実際の時間の差です。負のスラック値は、パスでセットアップまたはホールド タイム違反が発生していることを示します。
- **[Asynchronous]** : チェック マークが表示されている場合、クロック ドメイン間の関係が非同期であることを示します。**[Report Clock Interaction]** ダイアログ ボックスで **[Only report timing paths between asynchronous clocks]** をオンにしてレポートを生成した場合、すべてのパスに対してこの列にチェック マークが表示されます。
- **[Paths]** : クロック ドメイン間のパスが完全に制約されているか、一部のみ制約されているか、制約されていないかを示します。クロック 関係レポートの一番上のマトリックスに、クロック ドメインが制約されているか、制約されていないか、クロック ドメイン間に関係がないことが示されます。

表を右クリックしたときに表示されるポップアップ メニューで **[Export to Spreadsheet]** をクリックすると、表の内容がスプレッドシートで使える **XLS** ファイルに出力されます。

注記 : このコマンドは、列ヘッダーではなく、表内のセルを右クリックすると表示されます。

物理制約の定義

PlanAhead では、さまざまな方法で物理制約を適用できます。物理制約には、LOC/BEL インスタンス配置制約、AREA_GROUP 配置制約、DCI_CASCADE 制約、およびデバイス コンフィギュレーション モード制約などが含まれます。物理制約の詳細は、第 8 章「I/O ピン配置」および第 10 章、「デザインのフロアプラン」を参照してください。

次のセクションでは、**[Physical Constraints]** ビューとそのオプションについて説明します。

[Physical Constraints] ビューの使用

[Physical Constraints] ビューは、さまざまな物理制約タイプを表示および選択するために使用できます。**[Physical Constraint]** ビューを表示するには、**[Layout]** → **[Floorplanning]** をクリックするか、ツールバーのビュー レイアウト セレクターから **[Floorplanning]** を選択して、**Floorplanning** ビュー レイアウトに切り替えます。または、**[Window]** → **[Physical Constraints]** をクリックしても開くこともできます。

このビューには、作成した Pblock に基づくデザインの階層構造が表示されます。物理階層はダイナミックで、物理階層に変更を加えると自動的にアップデートされます。その他のビューでオブジェクトを選択すると、[Physical Constraints] ビューで該当するエレメントがハイライトされます。

[Physical Constraints] ビューに表示されるオブジェクトは、相対配置マクロ (RPM)、物理ブロック (Pblock)、DCI カスケード制約です。これらのオブジェクトをこのビューで選択して、ほかのビューで変更することも可能です。

デザインに追加した物理制約は、次のいずれかの方法で保存できます。

- [File] → [Save Design] をクリックします。
- [File] → [Save Design As] をクリックします。

保存せずにプロジェクトを閉じようとする、保存されていない変更があることを示すメッセージが表示されます。デザインに複数の制約セット、制約ファイルが含まれる場合は、追加された制約はすべてアクティブ制約セットのターゲット制約ファイルに記述されることに注意してください。

ROOT デザイン Pblock の使用

[Physical Constraints] ビューの物理階層には、一番上に合成済みデザイン (デフォルトは netlist_1) が表示され、その下に ROOT という最上位 Pblock が含まれます。

下位の Pblock を作成すると、子 Pblock は親 Pblock の下に、階層構造で表示されます。図 7-29 は、Pblock の階層を表示しています。

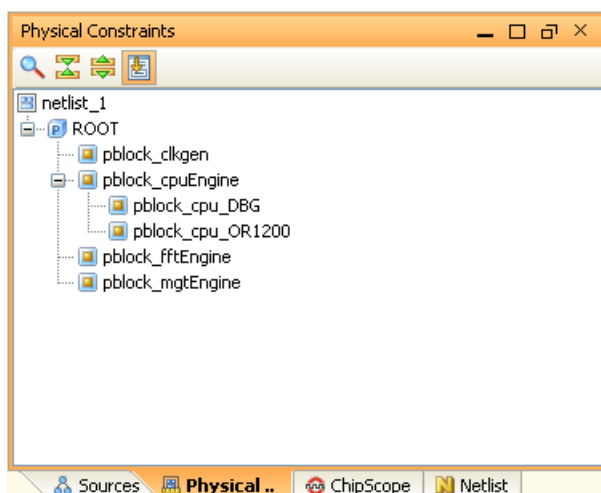


図 7-29 : [Physical Constraints] ビュー

Pblock を選択すると、これに割り当てられているすべてのロジックが選択されます。

[Physical Constraints] ビューのアイコン

[Physical Hierarchy] ビューのツリーでは、さまざまなオブジェクトのステートを識別するのに役立つアイコンが使用されています。このアイコン表示は物理階層が変更されると自動的に更新されます。

Pblock を作成すると、[Physical Hierarchy] ビューに階層的に表示されます。このビューに含まれる各フォルダー名の末尾にかっこで表示されている数字は、そのフォルダーに含まれているオブジェクト数を表します。

1 つの相対配置マクロ (RPM) の各インスタンスは、このビューに表示されます。その場合 Pblock の下の RPM フォルダーに表示されます。[Physical Hierarchy] ビューで RPM を選択すると、RPM に含まれているすべてのロジックが選択されます。

インスタンスが割り当てられた Pblock

インスタンスが割り当てられ、[Device] ビューで長方形が定義されている Pblock は、中心が黄色の青色の立方体アイコンで示されます。



インスタンスが割り当てられ、[Device] ビューで長方形が定義されていない Pblock は、中心が黄色の青色の正方形アイコンで示されます。



インスタンスが割り当てられていない Pblock

インスタンスが割り当てられておらず、[Device] ビューで長方形が定義されている Pblock は、中心に青色の P 文字が付いた青色の立方体アイコンで示されます。



インスタンスが割り当てられておらず、[Device] ウィンドウで長方形が定義されていない Pblock は、中心に青色の P 文字が付いた青色の正方形アイコンで示されます。



パーシャル リコンフィギュレーション可能な Pblock

パーシャル リコンフィギュレーションが可能なパーティション Pblock のアイコンには、次の図のように真ん中に黄色のひし形が表示されます。



相対配置マクロ (RPM) の操作

相対配置マクロ (RPM) のフォルダーには、デザインに含まれる RPM がリストされます。RPM は Pblock に割り当てることができます。その場合 Pblock の下の RPM フォルダーに表示されます。1 つの相対配置マクロ (RPM) の各インスタンスエーションがこのビューに表示されます (図 7-30)。

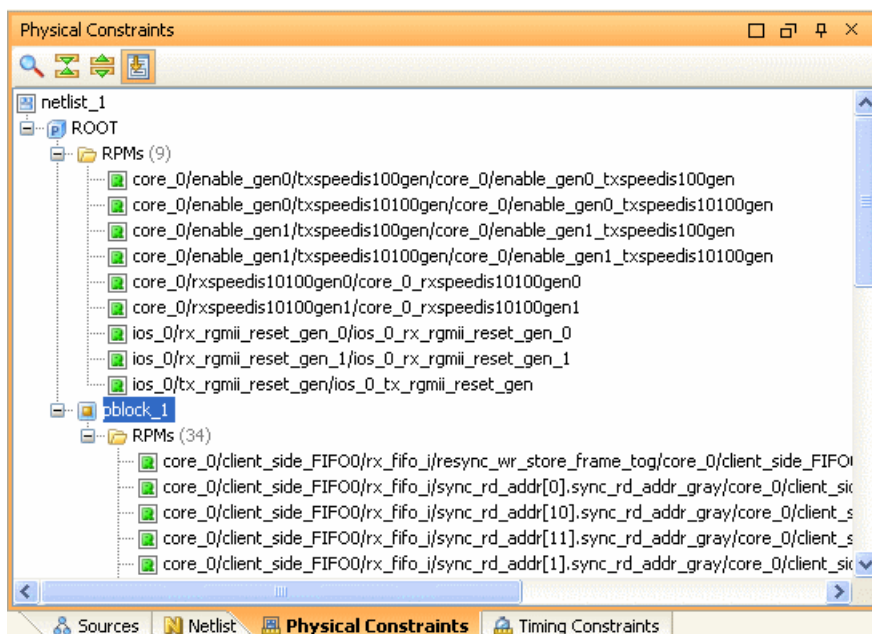


図 7-30：相対配置マクロ (RPM) の表示

RPM のプロパティと統計は、[RPM Properties] ビューに表示されます。RPM ロジックを Pblock に割り当てると、RPM のサイズおよび使用率の統計が図 7-31 のように [Pblock Properties] ビューの [Statistics] タブに表示されます。

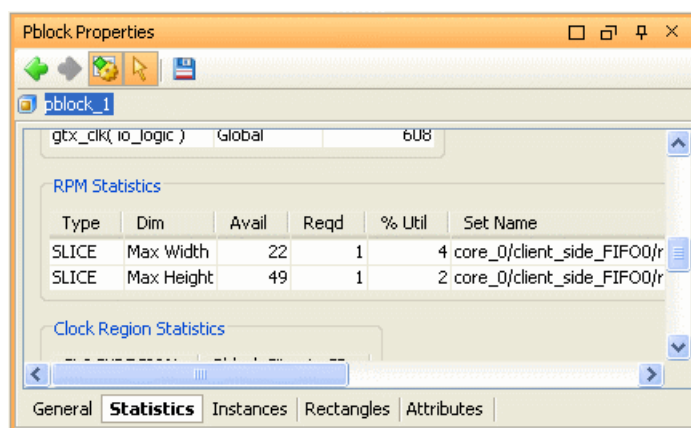


図 7-31：Pblock の RPM 使用率統計

デザイン ルール チェック (DRC) の実行

PlanAhead には、DRC バッチ コマンドのセットが含まれており、ISE ソフトウェアの実行前にデザインの整合性を確認できます。DRC は実行されるロジックチェックのタイプ別に分類されています。さまざまなタイプのチェックがあります。

これらのチェックを使用すると、発生する可能性のあるインプリメンテーションの問題を早期に発見できます。デザインが DRC のルールに従っているかどうかを確認する最終検証ステップは、ISE インプリメンテーション ツールを実行することです。

I/O ポートおよびクロック ロジック DRC の実行

DRC ルールの多くは、I/O ピン割り当てとクロック ロジックに関連しています。I/O ポートおよびクロック ロジックに関連する DRC の詳細は、[第 8 章「I/O ピン配置」](#) および [付録 B「PlanAhead の DRC」](#) を参照してください。

ネットリストおよび制約 DRC の実行

合成済みデザインで DRC を実行すると、ネットリストと制約を評価できます。

1. Flow Navigator または [Tools] メニューから [Report DRC] をクリックします。

[280 ページの図 7-32](#) に示す [Run DRC] ダイアログ ボックスが開きます。



注記：カテゴリのチェックの数は、リリースごとに異なります。

2. [Results Name] フィールドを確認または編集します。ここで指定した名前が [DRC Results] ビューに表示されるので、デバッグ時に区別しやすい名前を付けます。
3. [Output File] で出力されるレポート ファイルの名前を指定します。これはオプションのフィールドです。
4. [Rules to Check] グループ ボックスで、チェックするルールのチェック ボックスをオンにします。各ルールの詳細は、[付録 B「PlanAhead の DRC」](#) を参照してください。
 - [Expand All] ボタンをクリックするか、各カテゴリまたはデザイン オブジェクトの横のプラス記号 (+) をクリックすると、階層が展開します。
 - DRC すべてを実行するには、デザイン オブジェクトの横のチェック ボックスをすべてオンにします。
 - 特定の DRC を実行するには、そのチェック ボックスのみをオンにします。
 - [Select All] をクリックするとすべての DRC のチェック ボックスがオンになります。
5. [OK] をクリックし、選択した DRC チェックを実行します。

[DRC Results] ビューには、検出されたルール違反が [Run DRC] ダイアログ ボックスで定義された違反カテゴリ別に表示されます ([280 ページの図 7-32](#))。

ルール違反は、重要度で色分けされています。違反の種類は、次のとおりです。

- 情報のみ：問題となる可能性のある事項 (黄色)
- 警告またはクリティカル警告：対処が必要な可能性のある問題 (オレンジ)
- エラー：インプリメンテーションの障害となる問題 (赤)

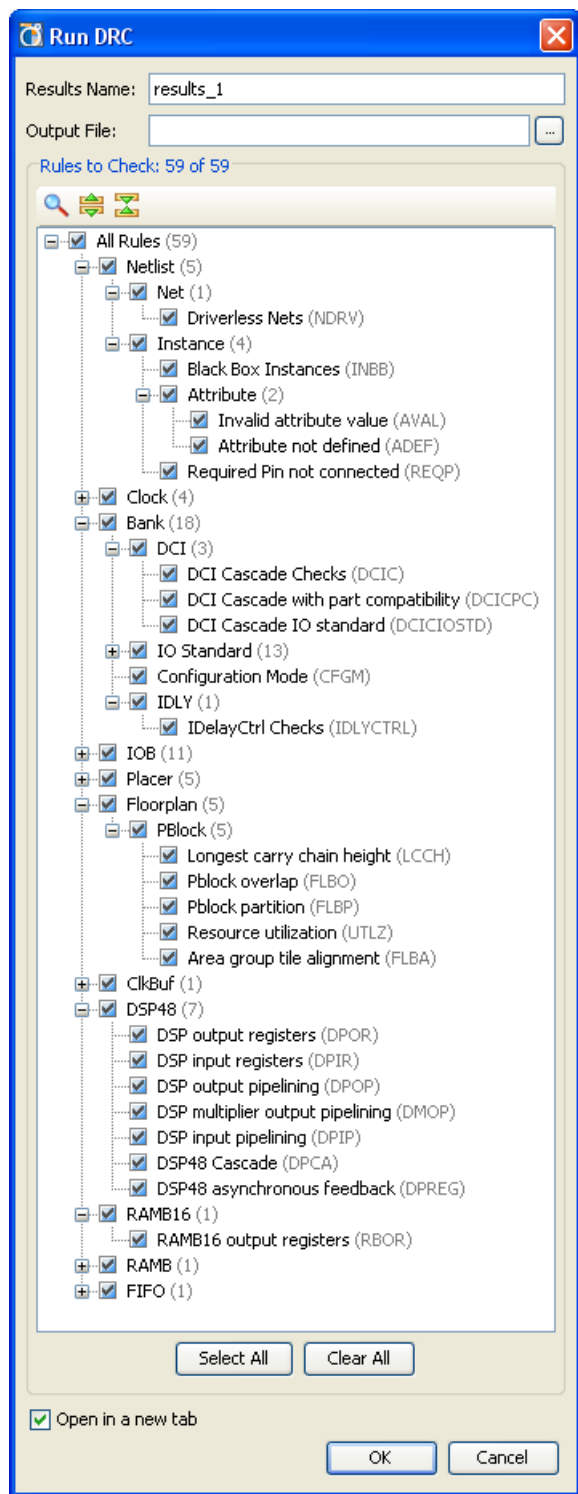


図 7-32 : [Run DRC] ダイアログ ボックス

DRC エラーの表示

DRC が完了すると、図 7-33 に示す [DRC Results] ビューが表示されます。

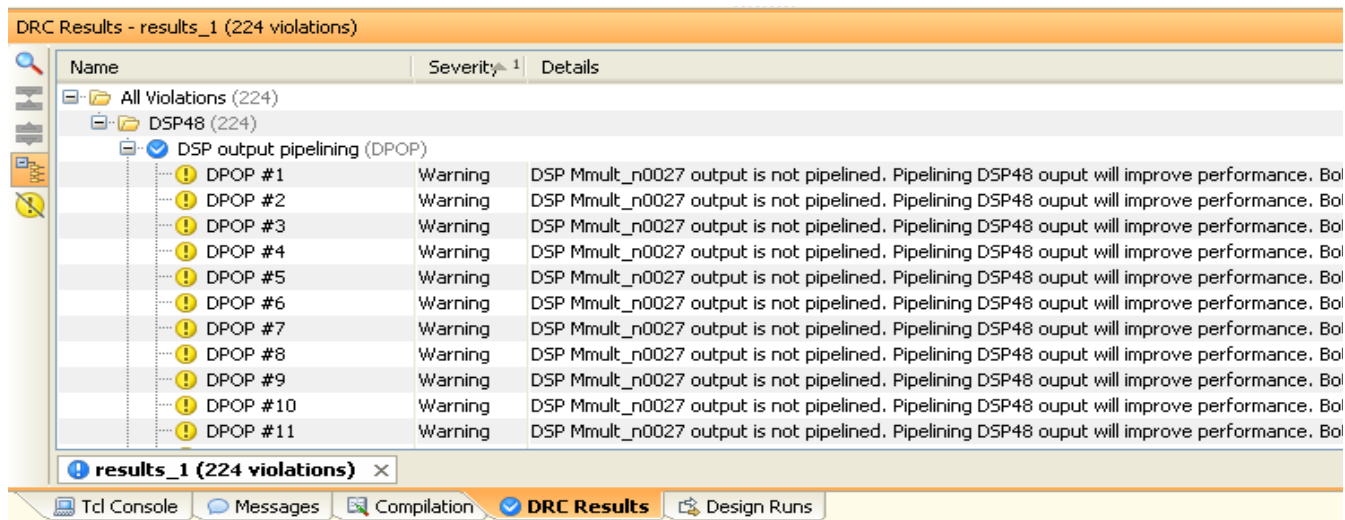


図 7-33 : [DRC Results] ビュー

警告および情報メッセージを非表示にし、エラーのみを表示するには、ツールバーの [Hide Warnings and Informational Messages] ボタンをクリックします。

[DRC Results] ビューの [Severity] 列のヘッダーをクリックすると、違反を重要度別に並び替えることもできます。

- 列ヘッダーを一度クリックすると重要度の低い方から並べられます。
- もう一度クリックすると重要度の高い方から並べられます。

詳細は、127 ページの「ツリー表形式のビュー」を参照してください。[DRC Results] ビューで違反メッセージを右クリックし、[Violations Properties] をクリックすると、[Violations Properties] ビューが開きます。このビューには、DRC ルール違反の概要と、違反しているデザイン エレメントの詳細が表示されます。

[Violations Properties] ビューの [Details] タブには、DRC に違反するデザイン オブジェクトへのリンクがあります。これらのリンクをクリックすると、[RTL Netlist] ビュー、[Device] ビュー、[Schematic] ビュー、またはソースの RTL ファイルでそのデザイン オブジェクトが表示されます。PlanAhead の DRC の詳細は、付録 B 「PlanAhead の DRC」を参照してください。

I/O ピン配置

I/O Planning ビュー レイアウトでは、デザインおよびデバイスの I/O 要件を解析し、PCB デザインと FPGA デザイン両方の要件を満たす I/O ピン配置を定義できます。

PlanAhead™ では、I/O ポート信号を作成したり (I/O Planning プロジェクトのみ)、CSV ファイル、ユーザー制約ファイル (UCF)、レジスタ トランスファー レベル (RTL) ファイルから I/O ポート リストをインポートできます。これにより、ピン配置を設計の初期段階で定義でき、設計サイクルの最後の方でピン配置に関連する変更が発生するのを回避できます。

ピン配置が最適化されていないと遅延が増加し、タイミングおよびシグナル インテグリティ要件を満たす障害となります。PCB から FPGA ダイへのデータ フローを考慮すると、最適なピン配置を短期間で定義でき、内部および外部のトレース長および配線密集を緩和できます。

I/O ピン配置の手法

I/O ピン配置には PCB 設計、FPGA 設計、システム設計などが関連し、それぞれ特有の考慮事項や要件があるので、I/O ピン配置は複雑なプロセスになります。この章では、PlanAhead 環境を使用したデバイスの詳細な確認方法、I/O ピン配置、および関連タスクについて説明します。

I/O ピン配置の詳細は、付録 E 「その他のリソース」に示す『ピン配置手法ガイド』(UG792)を参照してください。

I/O 配置ストラテジ

PlanAhead では、さまざまなデザイン プロセス段階で I/O 配置を実行できます。設計が進行するにつれ、より多くの情報が使用可能になるので、デザインの合成後およびインプリメント後には、より複雑なルール チェックを実行できます。

I/O の配置は、クロックのコンフィギュレーション方法によって異なることが多いので、I/O とクロック ロジックの割り当ては通常同時に実行されます。I/O 配置の DRC でクロックが考慮されるようにするには、合成済みデザインが必要です。I/O 配置は、できるだけ合成済みデザインで実行してください。

I/O ピンとクロック コンフィギュレーションの最終的な確認には、デザインをインプリメントします。クロック リソースが適切に配置されているかを検証するには、すべてのクロックが完全にインプリメントされていることが必要な場合があります。

PlanAhead では、I/O 配置を空のプロジェクト、RTL ソース ファイル、合成済みネットリスト、およびインプリメント済みデザインで実行できます。

デザインの初期段階では一部のデータがまだ存在せず、解析が概算のみになり、後半ではデータが決定され、変更できるデザイン部分が限られるので、デザイン プロセスの段階によって実行できる操作は異なります。

次に、デザイン段階を簡単に説明します。

1. I/O ピン配置プロジェクトの作成

空のプロジェクトを作成して、初期のデバイス チェックと I/O ポート コンフィギュレーションを実行できます。I/O ポートは手動で作成、または CSV/UCF ファイルからインポートします。デバイスおよび I/O ポート割り当てをエクスポートしておくと、デザイン プロセスの後半で使用できます。また、ポート定義とピン割り当てが解決したら、I/O ピン配置プロジェクトを RTL プロジェクトに変換することもできます。詳細は、[314 ページの「RTL デザインへの変換」](#)を参照してください。

2. RTL ソース ファイルのエラボレーションとチェック

PlanAhead の RTL ソースのプロジェクトで、I/O 配置を実行できます。エラボレート済みデザインでは、基本的な DRC (デザイン ルール チェック) が実行できます。クロック ロジックをチェックするには、合成済みデザインを使用することをお勧めします。

3. デザインの合成

合成後に、I/O 配置を実行できます。この段階ではすべてのクロックが決定されており、ツールですべてのクロックを視覚的に確認できるので、さらに詳細な検証を実行できます。I/O 配置は、可能な限り合成済みデザインで実行してください。

4. デザインのインプリメンテーションと I/O の最終検証

I/O 配置が有効かどうかを確認するには、デザインを完全にインプリメントする必要があります。NGDBuild と MAP レポートの I/O およびクロック関連のメッセージを確認してください。最終 DRC があるのは、ISE® インプリメンテーション ツールのみです。

I/O ポートの設定と配置

I/O ピン配置プロジェクトでは、ポートを作成またはインポートできます。ポートの I/O 規格、駆動電流、スルー タイプを設定します。詳細は、[295 ページの「I/O ポートの設定」](#)を参照してください。

パッケージ ピンまたは I/O バンクに PROHIBIT を設定すると、I/O の割り当てを禁止できます。詳細は、[299 ページの「I/O ピンおよび I/O バンクの使用禁止」](#)を参照してください。

I/O ポートは次の方法で配置できます。

- I/O をインターフェイスにグループ分けし、識別および選択しやすくします。詳細は、[300 ページの「I/O ポート インターフェイスの作成」](#)を参照してください。
- I/O ポートのグループをドラッグし、次の 3 つの配置モードのいずれかを使用して [Package] ビューまたは [Device] ビューで割り当てます。詳細は、[302 ページの「I/O ポートの配置」](#)を参照してください。

カーソルをサイトの上に置くと、配置されるポート数や、配置サイトがそのポートに使用できるかどうかなどの情報が示されます。詳細は、[301 ページの「インタラクティブなデザイン ルール チェックの使用」](#)を参照してください。

配置されていない I/O や I/O グループを自動的に配置するには、[Tools] → [I/O Planning] → [Autoplace I/O Ports] をクリックします。このコマンドでは、I/O バンク規則、差動ペアの規則、グローバル クロック ピンの規則に従って、可能な限り多くの I/O ポートが配置されます。この機能は一部のデバイス アーキテクチャでのみ使用でき、ルールが適用されるようにするには、合成済み ネットリストを使用する必要があります。詳細は、[306 ページの「I/O ポートの自動配置」](#)を参照してください。

PlanAhead では、割り当てが正しく実行されるように次の規則が適用されます。

- 差動ペアのポートを正しいピンのペアに配置する
- GTX を I/O バッファと一緒に配置し、デバイスでリソースが正しく割り当てられるようにする

インタラクティブな DRC およびバッチ DRC を使用して、I/O が正しく配置されます。詳細は、311 ページの「I/O とクロック ロジックの配置の検証」を参照してください。

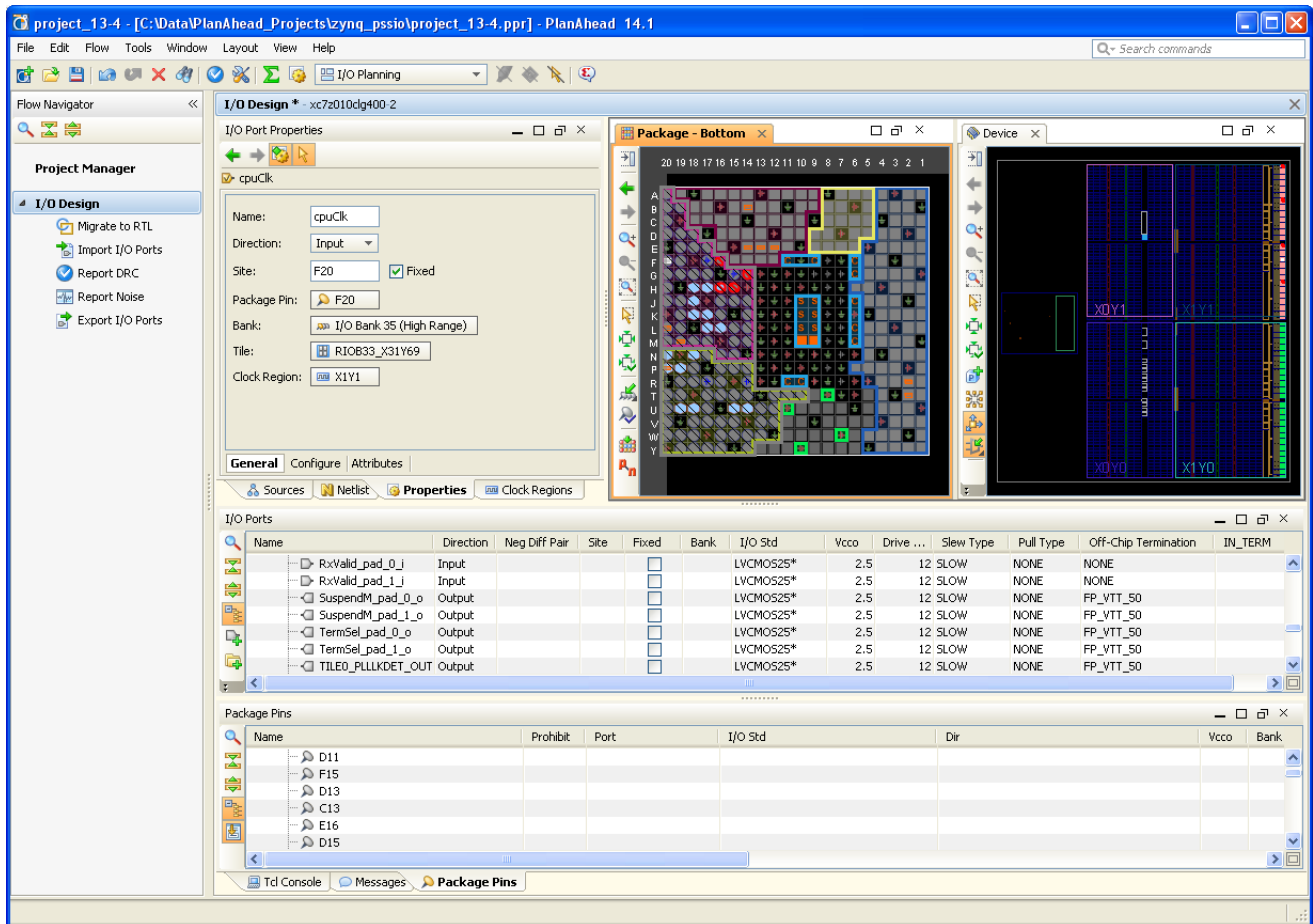


図 8-1 : I/O Planning 環境

I/O Planning ビュー レイアウトの使用

I/O Planning ビュー レイアウトは、エラボレート済みデザイン、合成済みデザイン、インプリメント済みデザイン環境で使用できます。このビュー レイアウトには、[Package] ビューと [Device] ビューの両方が含まれます。また、[Clock Resources]、[Clock Regions]、[Package Pins]、[I/O Ports]、[Properties] などのビューに、詳細な I/O 情報が提供されます。

I/O Planning ビュー レイアウトは、次の 2 つの方法で表示できます。

- [Layout] → [I/O Planning] をクリックするか、レイアウト セレクター (ドロップダウン リスト) から選択します。
- New Project ウィザードで新規のピン配置プロジェクトを作成します。

I/O Planning 環境については、第 4 章「表示環境の使用」の次のセクションを参照してください。

- 152 ページの「[Device] ビュー」

- 159 ページの「[Package] ビュー」
- 179 ページの「[I/O Ports] ビュー」
- 180 ページの「[Package Pins] ビュー」
- 182 ページの「[Design Runs] ビューの使用」

図 8-1 に、I/O Planning 環境を示します。

デバイス リソースの表示

[Device] ビューおよび [Package] ビューは、デバイスおよび配置されたロジック リソースをグラフィカルに表示します。ビューでロジック オブジェクトおよびデバイス サイトを選択すると、その情報が [Properties] ビューに表示されます。[Properties] ビューには、図 8-2 に示すように、さまざまなタイプの情報がタブ別に表示されます。

選択したオブジェクトのプロパティを表示するには、[Properties] ビューを表示します。[Properties] ビューが表示されていない場合は、メイン メニューから [Window] → [Properties] をクリックします。

特定のオブジェクトまたはデバイス サイトを検索するには [Find] コマンドを使用します。検索可能なオブジェクト タイプはさまざまで、デバイスまたはデザインで特定のオブジェクトを検索するフィルター機能もあります。オブジェクトは [Find Results] ビューで直接選択できます。詳細は、第 4 章の「[Find Results] ビュー」を参照してください。

パッケージ ピン プロパティ

[Package] ビューでピンまたは I/O バンクを選択すると、図 8-2 に示すように、その詳細が [Properties] ビューに表示されます。

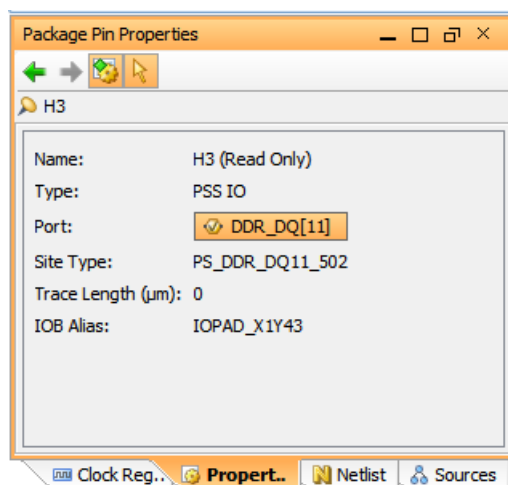


図 8-2 : パッケージ ピン プロパティ

クロック領域のリソースおよび統計

[Clock Regions] ビューでは、クロック領域を容易に選択できます。クロック領域を選択すると、関連した I/O バンクおよびリージョナル クロック リソースが図 8-3 のようにハイライト表示されます。

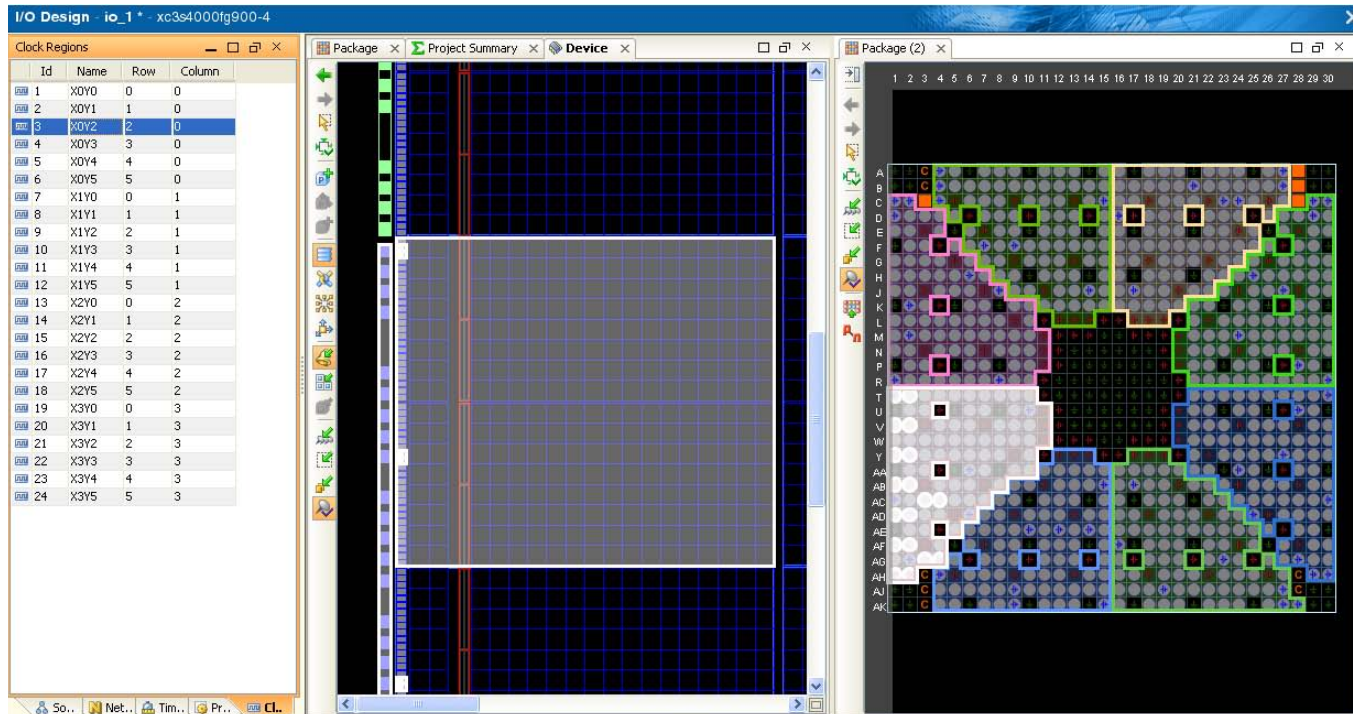


図 8-3 : I/O Planning のクロック領域ソース

[Properties] ビューには、選択したクロック領域のプロパティが表示されます。[Clock Region Properties] ビューの [Statistics] タブには、選択したクロック領域のリソース統計およびロジックの内容が表示されます。

[Resources] タブには、288 ページの図 8-4 に示すように、デバイス クロック サイトが表示されます。

[Clock Regions Properties] ビューでオブジェクトを選択すると、[Device] ビューなど、開いているほかのビューでも選択されます。

[Clock Resources] ビューには、使用可能なクロック リソースが表示されるので、グローバルおよびリージョナル クロック ツリーのエレメントを配置しやすくなっています。詳細は、308 ページの「[Clock Resources] ビューの使用」を参照してください。

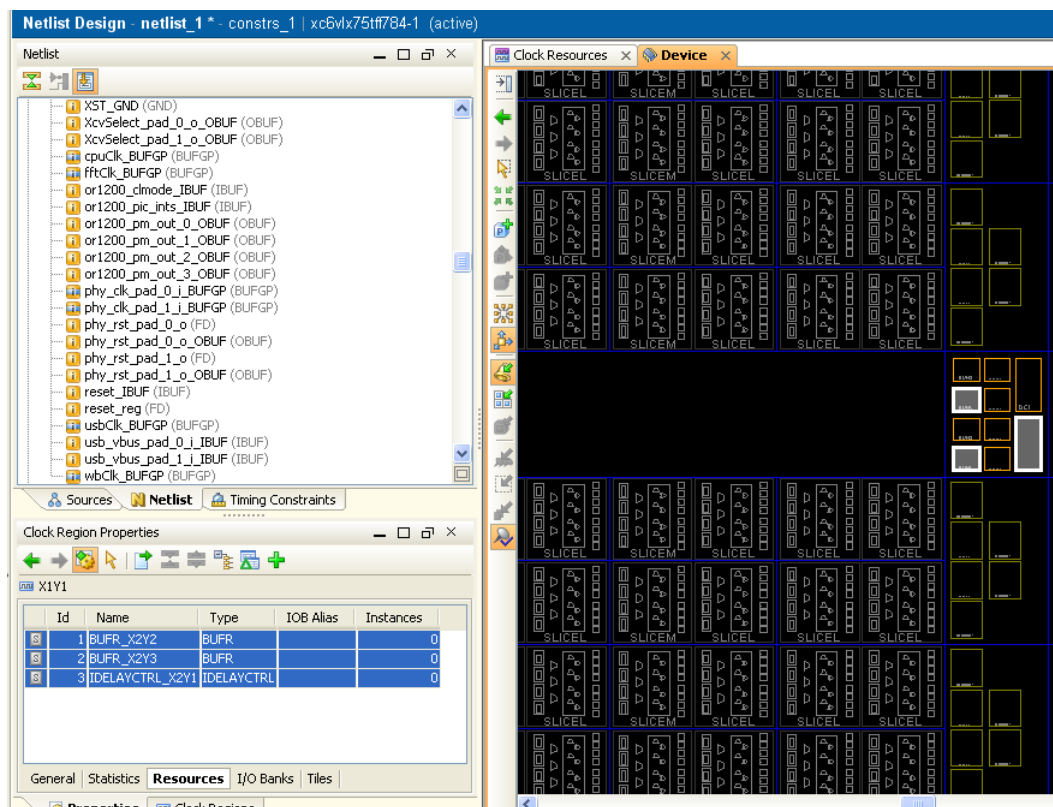


図 8-4 : クロック領域リソースの表示

I/O バンク リソース

I/O リソースは I/O Planning ビュー レイアウトのどのビューでも選択でき、選択したデータはほかのすべてのビューでハイライト表示されるため、[287 ページの図 8-3](#)物理パッケージと内部チップの関係を確認できます。

特定の I/O バンクに関する情報を入手するには、次の手順に従ってください。

1. [Package Pins] ビューで I/O バンクの 1 つを選択します。
[I/O Bank Properties] ビューが開きます。
2. [I/O Bank Properties] ビューの一番下のさまざまなタブをクリックし、使用可能な情報を確認してください。

[289 ページの図 8-5](#) に、[I/O Bank Properties] ビューを示します。

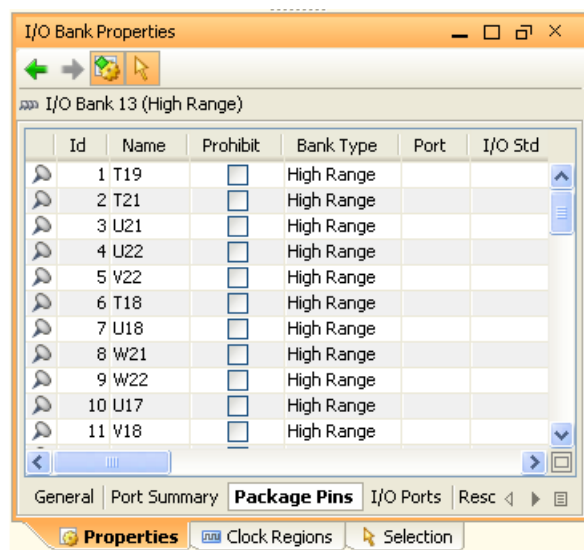


図 8-5 : I/O バンク プロパティの表示

多機能ピン

[Package Pins] ビューには、スプレッドシート形式でさまざまなデータが表示されます。このビューは、フラットにしたり、フィルターを適用したり、並べ替えたりできます。列を移動したり、非表示にしたり、設定変更して、さまざまな多機能ピンを表示および比較しやすくなります。セルは編集可能で、直接テキストを入力するか、ドロップダウン リストから値を選択できます。図 8-6 に、[Package Pins] ビューの例を示します。

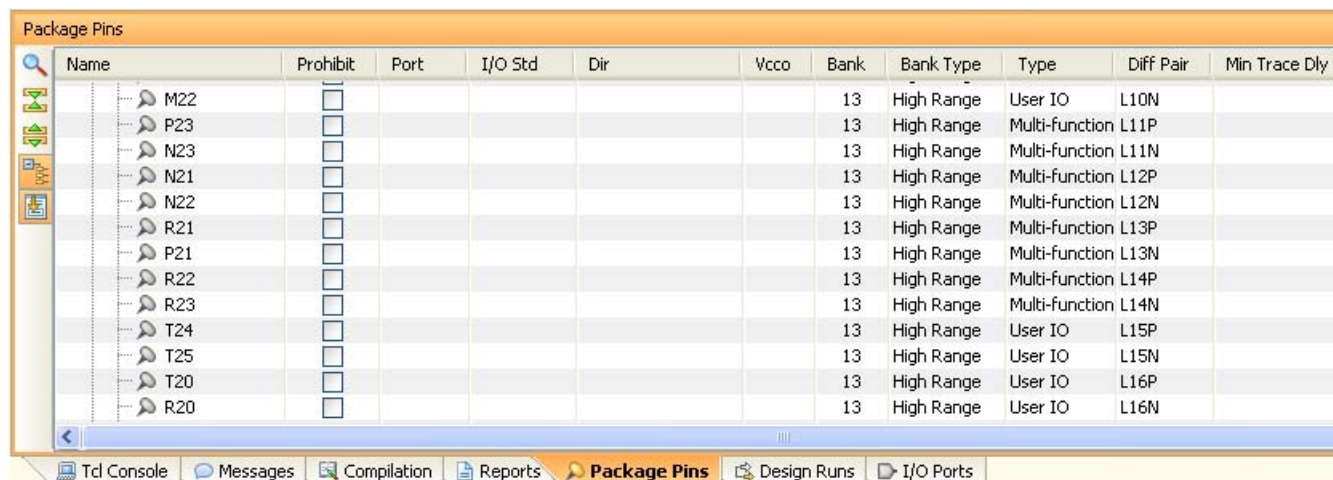


図 8-6 : [Package Pins] ビュー

[Type] 列は、多機能ピンのタイプを示します。その他の列には、多機能ピンに関するロジックまたはコンフィギュレーション モードなどの情報が表示されます。

[Package] ビューでは、多機能ピンの機能がシンボルで表されます。

- クロック ピンは、六角形で表示されます。
- Vref ピンには、電源マークが表示されます。

ツールバーの [Package View] ボタンをクリックすると、多目的ピンに使用されるシンボルがリストされます。詳細は、161 ページの「[Package View Options] の設定」を参照してください。



デザインにギガビット トランシーバー (GT)、メモリ コントローラー、または PCI™ ロジックが表示される場合、多機能ピンの競合を示す情報が [Package Pin] の表に表示されます。

デバイス コンフィギュレーション モードの多くで多機能ピンが使用されます。[Tools] → [I/O Planning] → [Set Configuration Modes] をクリックし、必要なデバイス コンフィギュレーション モードを設定します。詳細は、291 ページの「デバイス コンフィギュレーション モードの設定」を参照してください。

デバイス コンフィギュレーション モードを設定すると、多機能ピンのピン定義が [Package Pins] ビューの [Config] 列に表示されるようになります。

互換性のある代替デバイスの指定

PlanAhead では、デザインに互換性のある代替デバイスを選択でき、必要に応じてターゲット デバイスを変更できます。

I/O ピン割り当てが選択された代替デバイスすべてに対して有効かどうかチェックされます。現在選択されているターゲット デバイスと同じパッケージの代替デバイスが選択され、できるだけ多くの I/O 割り当てが保持されます。

互換性のある代替デバイスを定義するには、次の手順に従います。

1. [Tools] → [IO Planning] → [Set Part Compatibility] をクリックします。

このコマンドを実行すると、現在のプロジェクト デバイスと同じパッケージで使用可能な代替デバイスが表示されます。同じパッケージで使用可能な代替デバイスが図 8-7 のように表示されます。

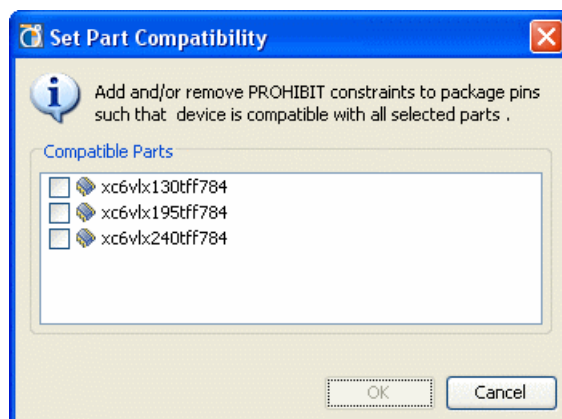


図 8-7 : 代替デバイスの選択

2. 代替デバイスを選択します。

デバイス間に共通のピンが特定され、共通ではないピンに対しては PROHIBIT 制約が適用されます。選択するパーツの数が増えると、配置に使用可能なパッケージ ピンの数が減ります。

信号が選択した代替デバイスのボンディングされていないピンに割り当てられないように、PROHIBIT 制約が自動的に設定されます。使用禁止のパッケージ ピン数を示すダイアログ ボックスが表示されます。PROHIBIT 制約は、[Package]、[Package Pins]、[Device] ビューで表示できます。

注記：Spartan®-6 LX25 または LX25T デバイスに互換性のある代替デバイスを定義する場合、このデバイスと代替デバイス間のクロック トポロジに違いがあるため、ボンディングされたピンにも PROHIBIT 制約が設定されます。詳細は、[アンサー 34885](#) を参照してください。

デバイス コンフィギュレーション モードの設定

PlanAhead には、[291 ページの図 8-8](#) に示すように、デバイス コンフィギュレーション モードに関する情報が表示されます。モードはいくつでも設定できます。

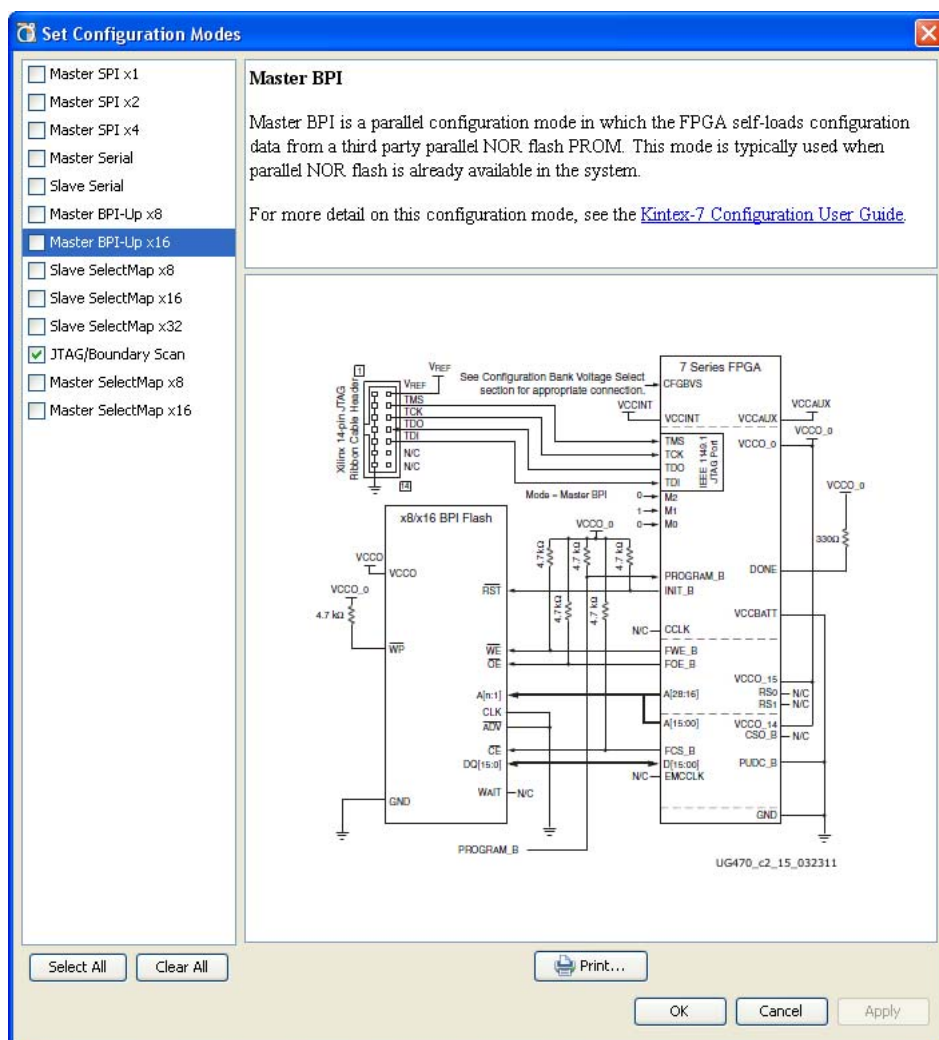


図 8-8 : [Set Configuration Modes] ダイアログ ボックス

デバイス コンフィギュレーション モードを設定するには、次の手順に従います。

1. [Tools] → [IO Planning] → [Set Configuration Modes] をクリックします。
 - コンフィギュレーション モードのいずれかをクリックし、回路図などの情報を表示します。
 - コンフィギュレーション図を印刷するには、[Print] をクリックします。
2. 必要なコンフィギュレーション モードを選択したら、[OK] をクリックします。

コンフィギュレーション モードを選択すると、[Package Pins] ビューの [Config] 列に関連する I/O ピンが表示されます。292 ページの図 8-9 に示すように、[Package Pins] ビューの [Config] 列を基準にピンを並べ替えるかどうかを尋ねるダイアログ ボックスが表示されます。

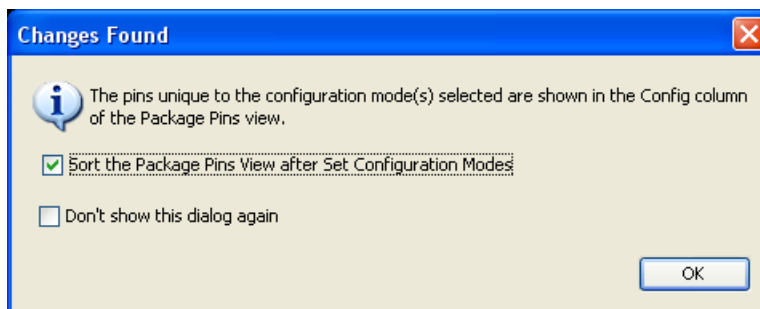


図 8-9 : [Config] 列を基準にピンを並べ替えるかどうかを尋ねるダイアログ ボックス

コンフィギュレーション モードとその他の多機能ピンとに競合があるかどうかを解析する場合は、289 ページの「多機能ピン」を参照してください。

I/O ポートの定義と設定

I/O Planning 環境を使用すると、空のピン配置プロジェクトから I/O ポートを作成および設定できます。

I/O ポートのインポート

PlanAhead では、プロジェクト作成時に空のピン配置プロジェクトへ UCF または CSV 形式のファイルをインポートするか、ファイルのインポート機能を後で使用することができます。

I/O ピン配置用に RTL ソースのプロジェクトを作成する場合は、RTL ファイルまたはヘッダーを使用し、設計プロセスが進行してからより完全な RTL ソース ファイルを追加します。RTL ベースまたは合成ネットリスト ベースのプロジェクトを作成すると、[I/O Ports] ビューにデザインで定義された I/O ポートが自動的にインポートされます。

I/O ポートをデザインの最上位 Verilog または VHDL モジュール定義に使用すると、I/O ピン配置プロジェクトを RTL プロジェクトに変換することもできます。詳細は、314 ページの「RTL デザインへの変換」を参照してください。

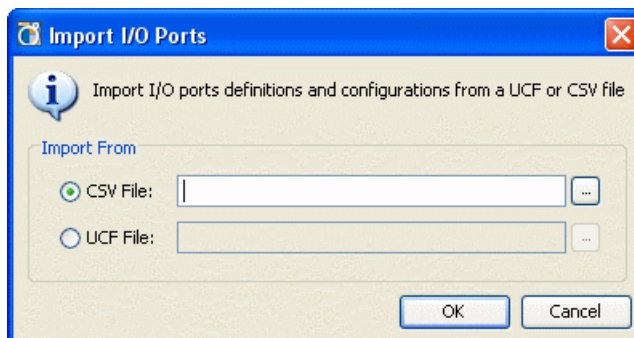


図 8-10 : [Import I/O Ports] ダイアログ ボックス

CSV ファイルのインポート

I/O ポートのリストを CSV ファイルからインポートするには、次の手順に従います。

1. [File] → [Import] → [Import I/O Ports] をクリックします。
 図 8-10 に示す [Import I/O Ports] ダイアログ ボックスが表示されます。
2. [CSV File] をオンにし、参照ボタンをクリックしてインポートする CSV ファイルを選択します。

図 8-11 に、CSV ファイル形式を示します。

A	B	C	D	E	F	G	H	I	J	K	L	M
Top: top Floorplan: floorplan_1 Part: xc5vsx35tf665-1												
Generated by: brianj on: Fri Feb 06 17:28:39 2009												
Build: PlanAhead v11.1.LR0 by: ECloudInternalUser4 on: Thu Feb 5 20:04:57 PST 2009												
IO Bank	Pin Number	IOB Alias	Site Type	Min Trace	Max Trace	Prohibit	Interface	Signal Name	Direction	DiffPair Type	DiffPair Sign	IO Standard
	P2	OPAD_X0Y5	MGTTP0_114	34878	40691			TXP_OUT[4]	OUT	P	TXN_OUT[4]	LVDS_25
	W2	OPAD_X0Y7	MGTTP1_114	41406	48307			TXP_OUT[5]	OUT	P	TXN_OUT[5]	LVDS_25
	B2	OPAD_X0Y13	MGTTP0_116	63540	74130			TXP_OUT[6]	OUT	P	TXN_OUT[6]	LVDS_25
	G2	OPAD_X0Y15	MGTTP1_116	55620	64890			TXP_OUT[7]	OUT	P	TXN_OUT[7]	LVDS_25
17	AD16	IOB_X0Y8	IO_L15N_17	80604	94038			DataOut_USB	OUT			LVCMOS25
17	AE15	IOB_X0Y6	IO_L16N_17	89010	103845			DataOut_USB	OUT			LVCMOS25
17	AC21	IOB_X0Y17	IO_L11P_CC_17	47370	55265			DataOut_USB	OUT			LVCMOS25
17	AE16	IOB_X0Y9	IO_L15P_17	78702	91819			DataOut_USB	OUT			LVCMOS25
17	AE21	IOB_X0Y22	IO_L8N_CC_17	63150	73675			DataOut_USB	OUT			LVCMOS25
17	AD20	IOB_X0Y18	IO_L10N_CC_17	62046	72387			DataOut_USB	OUT			LVCMOS25
17	AC23	IOB_X0Y26	IO_L6N_17	58710	68495			DataOut_USB	OUT			LVCMOS25
17	AF17	IOB_X0Y10	IO_L14N_VREF_17	80994	94493			DataOut_USB	OUT			LVCMOS25
17	AD24	IOB_X0Y36	IO_L1N_17	64386	75117			DataIn_USB_QIN				LVCMOS25

図 8-11 : CSV 形式の I/O ポート リスト

CSV は、FPGA やボード設計にあたり、デバイス ピンおよびピン配置のデータを交換するときに使用される標準ファイル形式です。PlanAhead では、[449 ページの「I/O ポート リスト \(CSV\)」](#)で説明するように、I/O ピン関連のデータをインポートするのに専用の CSV ファイル形式が必要です。

CSV ファイルでは、差動ペアをさまざまな方法で定義できます。PlanAhead では DiffPair Signal および DiffPair Type 属性で直接定義された差動ペアが認識され、ペアの 1 つのポートのみが CSV で定義されているか、2 つの該当ネットが差動ペアとして判断される場合に、差動ペアが推論されます。詳細は、[付録 A の「CSV での差動ペアの定義」](#)を参照してください。

PlanAhead で差動ペアが推論されると、ペアの割り当てを確認するメッセージが表示されます (図 8-12)。

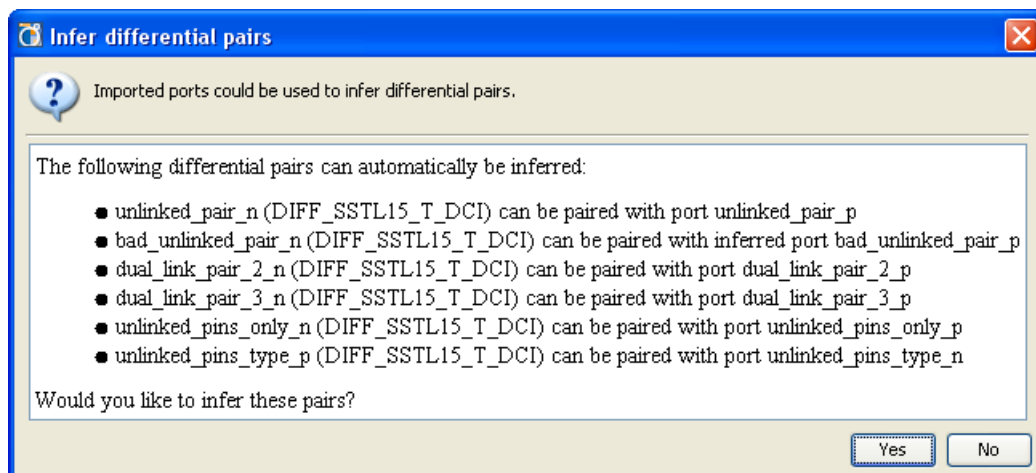


図 8-12 : 差動ペアの推論

CSV ファイルには、PlanAhead では認識されない追加情報を含めることもできます。インポートされた CSV ファイルに認識できない情報が含まれる場合、確認および使用できるように、[Package Pins] ビューのユーザー列に表示されます。

このようなユーザー CSV フィールドで値を変更または定義するには、[Package Pins] ビューでポップアップメニューから [Set User Column Values] をクリックします。

CSV ファイルをエクスポートするには、[File] → [Export] → [Export I/O Ports] をクリックします。追加された列およびユーザー定義の値が保持され、出力ファイルにエクスポートされます。

UCF 形式のファイルのインポート

UCF 形式のファイルをインポートして、[I/O Ports] ビューに表示できます。

UCF ファイルから I/O ポートの定義をインポートするには、[File] → [Import] → [Import I/O Ports] をクリックし、[UCF File] をオンにして UCF ファイルを選択します (292 ページの図 8-10)。

UCF ではポートの方向は定義されていないので、[Dir] 列には「undefined」と表示されます。I/O ポートの方向を定義するには、[I/O Ports] ビューでポップアップメニューから [Set Direction] をクリックします。[I/O Ports] ビューでは、特定の I/O ポートの方向を直接変更することもできます。

詳細は、296 ページの「I/O ポート方向の設定」を参照してください。

注記：インポートされた UCF でポートがワイルドカード (*) 構文を使用して定義される場合は、PlanAhead ではそのワイルドカードが展開されないため、それらのポートはインポートされません。

I/O ポートの新規作成

I/O ピン配置プロジェクトでは、新しいポートを手動で定義できます。デバイスの電圧に関する情報は、ザイリンクス デバイスの資料を参照してください。

I/O ポートを作成するには、次の手順に従います。

1. [I/O Ports] ビューで、ツールバーの [Create I/O Ports] ボタンをクリックします。

図 8-13 に示す [Create I/O Ports] ダイアログ ボックスが表示されます。

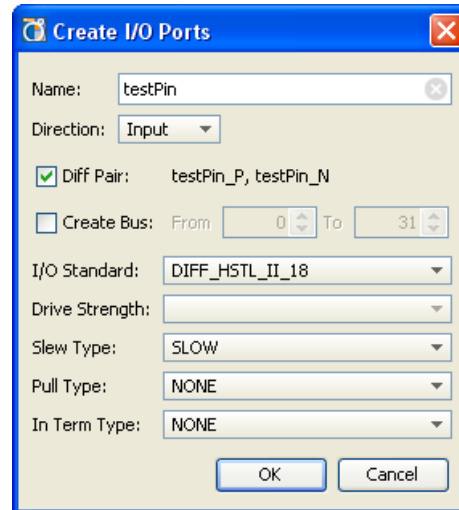


図 8-13 : [Create I/O Ports] ダイアログ ボックス

2. 次のオプションを変更します。

- [Name] : 作成するポートまたはバスの名前を指定します。
- [Direction] : ポートの方向を選択します。
- [Diff Pair] : 差動ペア信号またはバスを定義する場合にオンにします。これにより 2 つのポートが作成され、指定した名前に _P と _N がそれぞれ付きます。
- [Create Bus] : バスを作成する場合にバスの範囲を指定します。
- [I/O Standard] : I/O 規格を選択します。
- [Drive Strength] : 駆動電流の値を選択します。
- [Slew Type] : スルー レートを選択します。
- [Pull Type] : 抵抗のタイプを選択します。
- ターゲット パーツによって、次のいずれかが表示されます。
 - [Phase] : Virtex®-6 デバイスの場合にのみ適用されるオプションで、位相グループを入力するか、または既存の位相グループを選択します。位相グループは同時スイッチ ノイズ (SSN) の計算で使用されるポートの論理グループで、グループ内のポートには同じ周波数と位相が使用されます。詳細は、[321 ページの「SSN の I/O ポート スイッチ位相グループの定義」](#)を参照してください。
 - [In Term Type] : 入力信号の平行終端を定義します。

3. [OK] をクリックします。

I/O ポートの設定

1 つまたは複数の I/O ポートを設定して、[I/O Standard]、[Drive Strength]、[Slew Type]、[Pull Type]、[In Term Type] を定義できます。この方法は、適切な特性の指定がない CSV または UCF ファイルからインポートされたポートを設定する場合に便利です。デバイスの電圧に関する情報は、[ザイリンクス デバイスの資料](#)を参照してください。

ポートまたはポートのグループを設定するには、次の手順に従います。

1. [I/O Ports] ビューで、ポートを選択します。
2. 右クリックして [Configure I/O Ports] をクリックします。

図 8-14 に示すような [Configure Ports] ダイアログ ボックスが開きます。

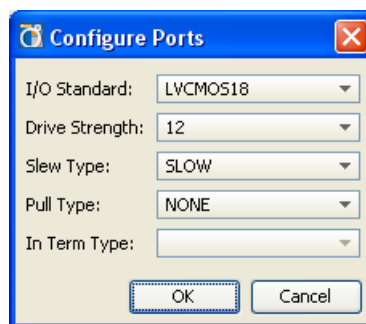


図 8-14 : [Configure Ports] ダイアログ ボックス

3. 次のオプションを変更します。

- [I/O Standard] : I/O 規格を選択します。I/O 規格は指定したときにはチェックされません。ポートに I/O 規格を割り当てることはできますが、DRC を実行するとエラーになる可能性があります。
- [Drive Strength] : 駆動電流の値を選択します。
- [Slew Type] : スルー レートを選択します。
- [Pull Type] : 抵抗のタイプを選択します。
- ターゲット パーツによって、次のいずれかが表示されます。
 - [Phase] : Virtex®-6 デバイスの場合にのみ適用されるオプションで、位相グループを入力するか、または既存の位相グループを選択します。詳細は、[321 ページの「SSN の I/O ポート スイッチ位相グループの定義」](#)を参照してください。
 - [In Term Type] : 入力信号の平行終端を定義します。

4. [OK] をクリックします。

I/O ポート方向の設定

I/O ポートの方向を設定するには、[I/O Ports] ビューで I/O ポート、バス、またはインターフェイスを右クリックし、[Set Direction] をクリックします。このコマンドを使用すると、ポートの方向を入力 ([Input])、出力 ([Output])、または入出力 ([In/Out]) のいずれかに定義できます。このコマンドは、I/O ピン配置プロジェクトでのみ使用できます。RTL ソース プロジェクトでは、RTL ソースでポートの方向を定義します。

I/O ポートは、[I/O Ports] ビューの表でポートの [Dir] 列をクリックしてドロップ ダウン メニューから方向を変更しても直接編集できます。

差動ペアの作成と分割

I/O ピン配置プロジェクトで差動ペアを定義するには、次を実行する必要があります。

1. 2 つの I/O ポートを選択し、[I/O Ports] ビューのポップアップ メニューから [Make Diff Pair] を選択します (図 8-15)。

注記 : [Make Diff Pair] オプションは RTL ベースのプロジェクトでは使用できません。RTL ベースのプロジェクトでは、適切な I/O バッファのインスタネーションを使用して、ソースコードで差動ペアを定義する必要があります。

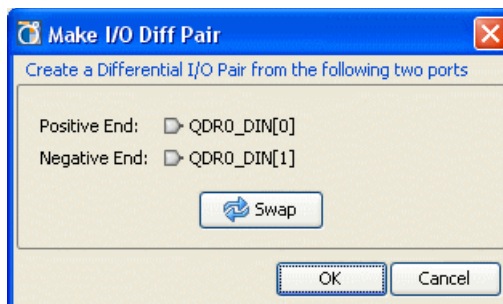


図 8-15 : [Make I/O Diff Pair] ダイアログ ボックス

[Make I/O Diff Pair] ダイアログ ボックスでは、2 つの I/O ポートが正側と負側に指定されます。

2. 正側と負側の信号を入れ替えるには、[Swap] をクリックします。
3. [OK] をクリックします。

ポップアップ メニューで [Split Diff Pair] をクリックすると、差動ペアを 2 つのポートに分割できます。

DCI_CASCADE 制約の設定

DCI_CASCADE 制約は、DCI 参照電圧用に近接する 2 つ以上の I/O バンクをリンクするために使用します。DCI 参照電圧を使用する I/O バンクは「マスター」と呼ばれ、カスケード内のそれ以外のものは「スレーブ」と呼ばれます。カスケードのすべてのバンクはデバイスの同じ I/O 列に配置される必要があります。

ザイリンクス 7 シリーズ FPGA、Virtex®-6、Virtex-5 デバイスの場合は、DCI_CASCADE 制約を設定できます。

制約を設定するには、次の手順に従います。

- コマンドを実行する前に I/O バンクを選択しておきます。
または
- コマンド ダイアログ ボックスで I/O バンクを選択します。

DCI_CASCADE 制約の詳細は、付録 E 「その他のリソース」 に示す『制約ガイド』(UG625) を参照してください。

DCI_CASCADE 制約の作成

DCI_CASCADE 制約を作成するには、次の手順に従います。

1. 制約を設定する I/O バンクを選択します。
2. [Package Pins] ビューまたは [Package] ビューのポップアップ メニューから [Create a DCI Cascade] をクリックします。

DCI Cascade Editor が開きます。コマンドを実行する前に選択されていた I/O バンクは、ダイアログ ボックスに表示されています。

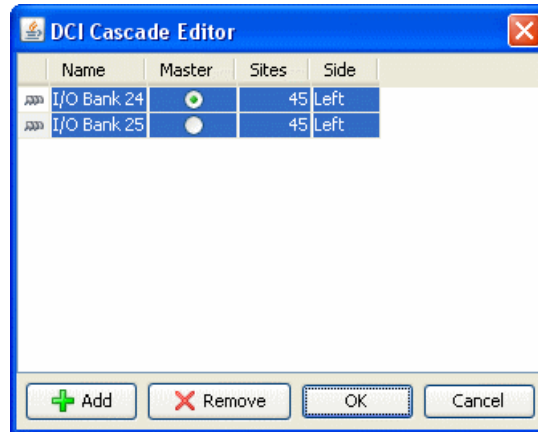


図 8-16 : DCI カスケード制約の作成

3. I/O バンクを追加するには、[Add] ボタンをクリックします。
デバイスの同じ列から I/O バンクを選択します。これは、DCI_CASCADE の作成時にはチェックされませんが、DRC を実行したときにチェックされます。
4. マスターにする I/O バンクを選択します。
5. [OK] をクリックします。

I/O バンクを選択すると、ほかのビューでもハイライトされます。

DCI カスケード制約が [Physical Constraints] ビューに表示されます。詳細は、299 ページの図 8-17 を参照してください。[Physical Constraints] ビューで DCI カスケードを選択すると、関連する I/O バンクが選択されます。

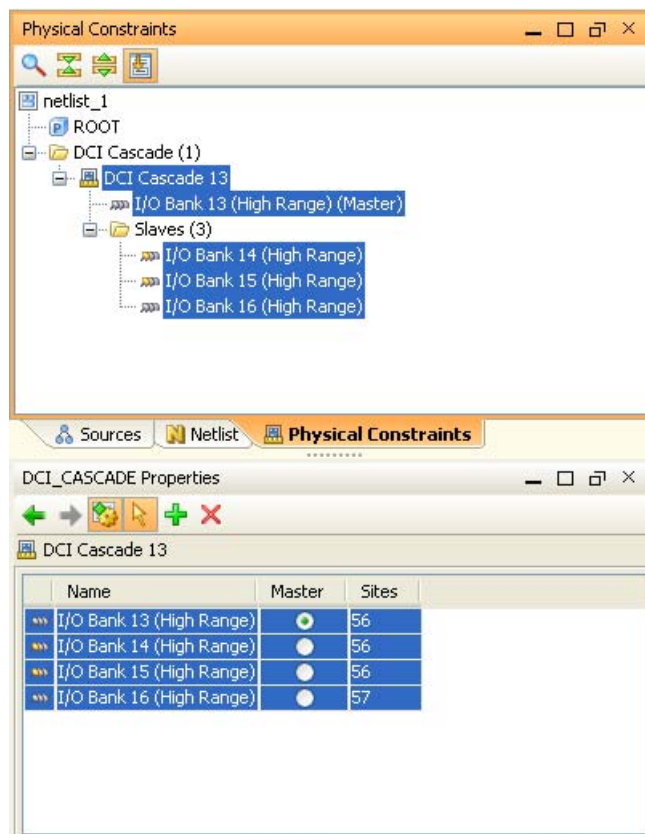


図 8-17 : DCI カスケードの表示

DCI カスケード制約の変更と削除

DCI カスケードは、[Physical Constraints] ビューで DCI カスケードを選択して、[DCI_CASCADE Properties] ビューから変更できます。可能な操作は次のとおりです。

- マスターを変更するには、別の I/O バンクの [Master] をオンにします。
- DCI カスケードから I/O バンクを削除するには、[DCI_CASCADE Properties] ビューで I/O バンクを選択し、ツールバーの [Delete I/O Banks] ボタンをクリックします。
- DCI カスケードに I/O バンクを追加するには、[DCI_CASCADE Properties] ビューでツールバーの [Add I/O Banks] ボタンをクリックします。[Select I/O Banks] ダイアログボックスが表示されます。ここで、新しい I/O バンクを選択します。新しく選択した I/O バンクは、ほかのビューでもハイライトされます。
- DCI カスケード制約を削除するには、[Physical Constraints] ビューで制約を右クリックし、[Delete] をクリックします。
- すべての変更を保存するには、[DCI_CASCADE Properties] ビューで [Apply] をクリックします。

I/O ピンおよび I/O バンクの使用禁止

I/O Planning ビュー レイアウトでは、I/O ピン、I/O ピンのグループ、または I/O バンクを使用禁止に設定できます。[Device]、[Package]、[Package Pins] ビューでピンを選択し、禁止にします。

I/O ピンまたは I/O バンクを使用禁止にするには、次の手順に従います。

1. [Device]、[Package]、[Package Pins] ビューで、I/O ピンまたは I/O バンクを選択します。

2. 右クリックして [Set Prohibit] をクリックします。

図 8-18 に示すように、[Package] ビューに使用禁止になったピンに赤い X マークが入った円が表示され、[Package Pins] ビューの [Prohibit] 列のチェック ボックスがオンになります。



図 8-18 : 使用禁止に設定されたパッケージ ピン

I/O ポート インターフェイスの作成

複数のポートまたはバスをまとめてグループにするには、インターフェイスを作成します。インターフェイス ポートがすべて 1 つのグループとして処理されるので、ピンの割り当てに便利です。ピンをすべて同時に割り当てると、クロック領域または PCB 配線関連のインターフェイスをまとめた分離しやすくなります。特定のロジック インターフェイスに関連付けられた信号の表示および管理も容易になります。

インターフェイスを作成するには、次の手順に従います。

1. [I/O Ports] ビューで、1 つのグループにまとめる信号を選択します。
2. 右クリックして [Create I/O Port Interface] をクリックします (図 8-19)。

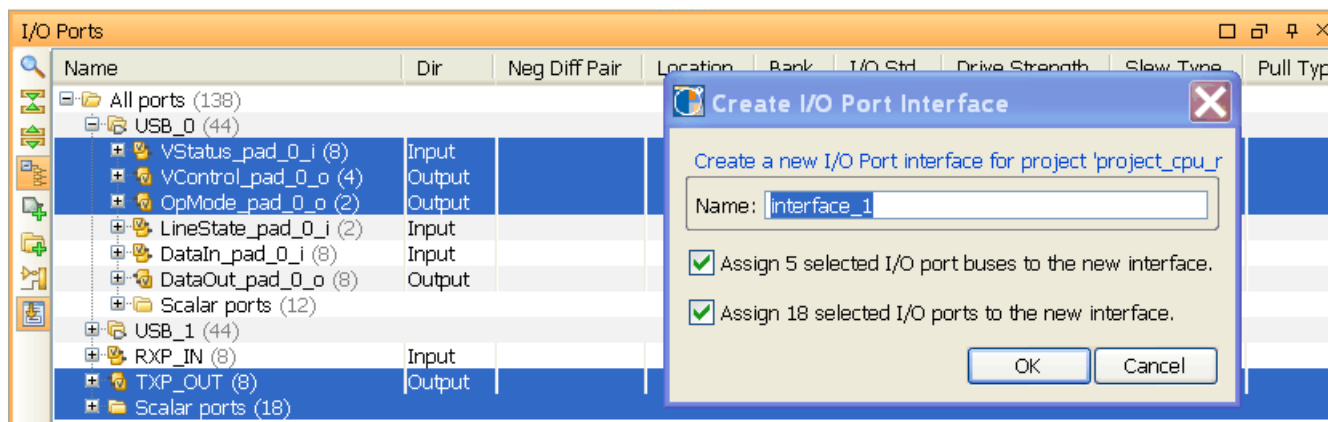


図 8-19 : [Create I/O Port Interface] ダイアログ ボックス

3. インターフェイス名を指定し、ほかの部分が必要に応じて変更したら、[OK] をクリックします。

インターフェイスが [I/O Ports] ビューに展開可能なフォルダーとして表示されます。[I/O Ports] ビューで I/O ポートを選択してインターフェイス フォルダーにドラッグすると、I/O ポートを追加できます (図 8-20)。

Name	Dir	Neg Diff Pair	Location	Bank	I/O Std	Drive Strength	Slew Type	Pull
All ports (138)								
USB_0 (30)								
LineState_pad_0_i (2)	Input			17	LVCMOS25	12	SLOW	
DataIn_pad_0_i (8)	Input				LVCMOS25	12	SLOW	
DataOut_pad_0_o (8)	Output			17	LVCMOS25	12	SLOW	
Scalar ports (12)								
USB_1 (44)								
interface_1 (48)								
TXP_OUT (8)	Output				LVDS_25			
OpMode_pad_0_o (2)	Output			17	LVCMOS25	12	SLOW	
VControl_pad_0_o (4)	Output			17	LVCMOS25	12	SLOW	
VStatus_pad_0_i (8)	Input			17	LVCMOS25	12	SLOW	

図 8-20 : I/O ポート インターフェイスの管理

I/O ポートをインターフェイスに追加するには、次の手順に従います。

1. ポートまたはバスを選択します。
2. 右クリックして [Assign to Interface] をクリックします。
3. I/O ポートを追加するインターフェイスを選択します。

I/O ポートおよびインターフェイスを削除するには、次の手順に従います。

1. ポートまたはインターフェイスを選択します。
2. 右クリックして [Unassign from Interface] をクリックします。

インターフェイスを削除するには、削除するインターフェイスを選択し、右クリックして [Delete] をクリックするか、Delete キーを押します。

インタラクティブなデザイン ルール チェックの使用

PlanAhead ではピン配置が有効であることを確認するチェックが実行されますが、完全な最終 DRC 機能があるのは ISE® インプリメンテーション ツールのみです。そのため、ISE ピン配置プロセスを実行して最終的なピン配置が有効かどうか確認する必要があります。

インタラクティブな I/O 配置ルーチンでは、よく発生するエラーがチェックされます。次のいずれかを実行すると、この機能のオン/オフを切り替えることができます。

- [Device] または [Package] ビューのツールバーの [Autocheck I/O Placement] ボタンをクリックします。
- メイン メニューの [Tools] → [Options] コマンドをクリックし、ダイアログ ボックスの左側の [General] ボタンをクリックします。これにより、ダイアログ ボックス右側の [Automatically Enforce Legal I/O Placement] オプションを設定できるようになります。

この自動チェック機能をオンにすると、問題を引き起こすピンに I/O ポートを配置できなくなります。[Place I/O Ports Sequentially] をオンして I/O ポートを順次に割り当てる場合、問題のあるピンに I/O ポートを配置しようとすると、I/O ポートを配置できない理由がツール ヒントに表示されます。インタラクティブ DRC はデフォルトでオンになっています。

注記：チェックの多くは、デザイン全体の合成済みネットリストが読み込まれている場合にのみ実行可能です。

インタラクティブな I/O 配置ルールには、次のものが含まれます。

- 禁止：
 - ギガビット トランシーバー (GT) に関連する、ノイズの影響を受けやすいピンへの配置、ノイズの影響を受けやすい I/O パッケージ ピン
 - I/O 規格に違反する配置
- 確認：
 - バンクでサポートされていない I/O 規格が使用されていない
 - バンクに互換性のない VCC ポートが割り当てられていない
 - VREF ポートが必要なバンクに使用可能な VREF ピンある
 - グローバル クロックおよびリージョナル クロックが適切に割り当てられている (EDIF/NGC ネットリストおよび UCF がインポートされている場合のみ)
 - 入力および High 駆動の出力が適切なピンにのみ送信される (Spartan®-3 デバイスのみ)
 - 差動 I/O ポートが正しい極性ピンに割り当てられている
 - 出力ピンが入力のみのピンに配置されていない

I/O ポートの配置は、DRC をオンにして実行することをお勧めします。I/O に関連する DRC については、[466 ページの「I/O ポート/クロック ロジック/配置 DRC の詳細」](#)を参照してください。

I/O ポートの配置

I/O Planning ビューレイアウトでは、さまざまな方法で I/O ポートをパッケージ ピンに割り当てることができます。個別の I/O ポート、I/O ポートのグループ、またはインターフェイスを [I/O Ports] ビューで選択し、[Package] ビューでパッケージ ピンに、または [Device] ビューで I/O パッドに割り当てることができます。

注記：Zynq™ デバイスには、読み出し専用のプロセッサ サブシステム IO ピン (PSSIO) が含まれます。これらのピンは、Zynq デバイス上ではプログラマブル ロジック (PL) ピンとしてコンフィギュレーションできないので、I/O ピン配置プロジェクトではポートを割り当ててすることはできません。PSSIO ピンは、プロセッサ システム (PS) システムとしてコンフィギュレーション可能で、XPS からインポートできます。詳細は、[XPS ヘルプ](#)を参照してください。

I/O ポートの順次配置

I/O ポートを順次に配置するには、次の手順に従います。

1. [I/O Ports] ビューで、個別の I/O ポート、I/O ポートのグループ、またはインターフェイスを選択します。
2. 次のいずれかを実行します。
 - [I/O Ports] ビューのポップアップ メニューで、[Place I/O Ports Sequentially] をクリックします。
 - [Package] または [Device] ビューで、ツールバーの [Place I/O Ports Sequentially] ボタンをクリックし、[Place I/O Ports Sequentially] モードをオンにします。



カーソルをパッケージピンまたは I/O パッド上に移動すると、グループの最初の I/O ポートがカーソルと共に移動します。ツール ヒントには、I/O ポート名およびパッケージピン名が表示されます。

3. ピンまたはパッドをクリックすると、I/O ポートが割り当てられます。

303 ページの図 8-21 に、I/O ポートの順次配置を示します。

4. 配置されていない I/O ポートが残っている場合は、コマンドは継続されます。
I/O ポートがすべて割り当てられるまでこの操作を繰り返すか、または Esc キーを押してモードを解除します。



図 8-21 : I/O ポートの順次配置

ポートは、[I/O Ports] ビューに表示されている順序で割り当てられます。割り当ての前に [I/O Ports] ビューでポートを並べ替えると、並べ替えられた順序で割り当てられます。

I/O ポートの I/O バンクへの配置

I/O ポートを I/O バンクに配置するには、次の手順に従います。

1. [I/O Ports] ビューで、個別の I/O ポート、I/O ポートのグループ、またはインターフェイスを選択します。
2. 次のいずれかを実行します。
 - [I/O Ports] ビューのポップアップメニューで [Place I/O Ports in an I/O Bank] をクリックします。
 - [Package] または [Device] ビューで、ツールバーの [Place Ports] ボタンをクリックし、[Place I/O Ports in an I/O Bank] モードをオンにします。

カーソルをパッケージピンまたは I/O パッド上に移動すると、I/O ポートのグループがカーソルと共に移動します。ツール ヒントには、カーソルを移動した I/O に配置されるピンの数が表示されます。

3. ピンまたはパッドをクリックすると、選択された I/O ポートが割り当てられます。304 ページの図 8-22 に、I/O ポートを I/O バンクに割り当てる例を示します。



図 8-22 : I/O ポートの I/O バンクへの配置

4. I/O バンクに収まらない数の I/O ポートが選択されている場合、選択された I/O バンクにできるだけ多くのポートが配置され、残りのポートを配置する別の I/O バンクを選択できます。残りの I/O ポートを別の I/O バンクにドラッグすると、I/O ポートがすべて配置されるか、Esc キーを押すまでコマンドが継続されます。
- カーソルを [Package] ビュー上で移動すると、ビューの上と左に I/O ピンの座標が表示されます。
 - その他の I/O ピンおよびバンク情報は、メイン ウィンドウの下部にあるステータス バーに表示されます。
 - レポートされるアクティブなオブジェクトが [Package] ビューでハイライトされます。
 - カーソルを [Package] ビュー上に置くと、ツール ヒントにピン情報が表示されます。

ポートは、[I/O Ports] ビューに表示されている順序で割り当てられます。割り当ての前に [I/O Ports] ビューでポートを並べ替えると、並べ替えられた順序で割り当てられます。

デバイス リソースへのポートの割り当ても、I/O バンクの最初の選択から実行されます。I/O バンクの一端にあるピンを選択すると、バスが I/O バンク全体に割り当てられます。

バスに関連する PCB 配線も考慮されます。ピンは、割り当て領域内でバス ビットのベクターが保持されるような順序で割り当てられます。割り当てパターンは、その他のバス配線の考慮事項に対応するようカスタマイズできます。

I/O ポートの定義された領域への配置

I/O ポートを定義された領域に配置するには、次の手順に従います。

- [I/O Ports] ビューで、個別の I/O ポート、I/O ポートのグループ、またはインターフェイスを選択します。



2. 次を実行します。
 - [I/O Ports] ビューのポップアップメニューで、[Place I/O Ports in Area] をクリックします。
3. [Package] または [Device] ビューで、ツールバーの [Place Ports] ボタンをクリックし、 [Place I/O Ports in Area] モードをオンにします。
カーソルが十字型に変化し、ポートを配置する長方形を定義できるようになります。
4. [Package] ビューまたは [Device] ビューのいずれかで長方形を描画し、 に示すように割り当て領域を定義します。
5. 定義した領域に選択された I/O ポートすべてを配置できない場合は、コマンドが継続します。カーソルは十字型のままで、残りの I/O ポートがすべて配置されるまで、長方形を続けて描画できます。Esc キーを押すと、I/O ポートを配置し終える前にコマンドが終了します。



図 8-23 : I/O ポートの領域への配置

ポートは、[I/O Ports] ビューに表示されている順序で割り当てられます。割り当ての前に [I/O Ports] ビューでポートを並べ替えると、並べ替えられた順序で割り当てられます。

長方形の描画方向によって割り当て順序が変わります。I/O ポートは最初選択された長方形の座標の内側のピンから順に割り当てられます。領域の長方形を柔軟に定義すると、PCB 配線の観点から有益なピン配置が得られます。

配置済み I/O ポートの入れ替えおよび移動

既に割り当てした I/O ポートを移動または入れ替える必要がある場合があります。配置済み I/O ポートを 2 つ選択してそれらのロケーションを入れ替えるには、次の手順に従います。

1. 使用可能なビューで I/O ポートを 2 つ選択します。
複数のポートを選択するには、Ctrl キーを押しながらポートをクリックします。

2. 右クリックで [Swap Locations] をクリックします。

インプリメント済みデザインで配置を表示しているときに、固定されていない 2 つのポートを入れ替えると、それらのポート固定され、LOC 制約がデザイン制約ファイルに記述されます。

同様に、ポートやポートのグループも、選択して別の場所に移動できます。I/O バンクのポート グループを別の I/O バンクに移動する場合、PlanAhead で選択したポートに最適な場所が自動的に検出されます。これは、[Place I/O Ports in an I/O Bank] コマンドを使用するのと同様です。

I/O ポートの自動配置

[Auto-place I/O Ports] コマンドを実行すると、すべての I/O ポート、配置されていない I/O ポート、または選択した I/O ポートを自動的にパッケージ ピンに割り当てることができます。自動配置では、I/O 規格および差動ペアの規則すべてに従い、グローバル クロック ピンが正しく配置されます。

注記： I/O の自動配置機能は、ザイリンクス 7 シリーズ FPGA および Virtex-6 デバイスでのみ使用可能です。



図 8-24 : Autoplace I/O Ports ウィザード

I/O ポートを自動配置するには、次の手順に従います。

1. [I/O Ports] ビューで、自動配置する I/O ポートを選択します。
 2. 次のいずれかを実行します。
 - [Tools] → [I/O Planning] → [Auto-place I/O Ports] をクリックします。
 - [I/O Ports] ビューでポートを右クリックし、[Auto-place I/O Ports] をクリックします。
- 図 8-24 に示す Autoplace I/O Ports ウィザードが表示されます。
3. 配置する I/O ポートのグループを選択し、[Next] をクリックします。
 4. 既にパッケージ ピンに割り当てられている I/O ポートのみを選択すると、307 ページの図 8-25 に示すページが表示されます。
 5. 配置する I/O ポートを選択し、[Next] をクリックして、サマリ ページで [Finish] をクリックします。

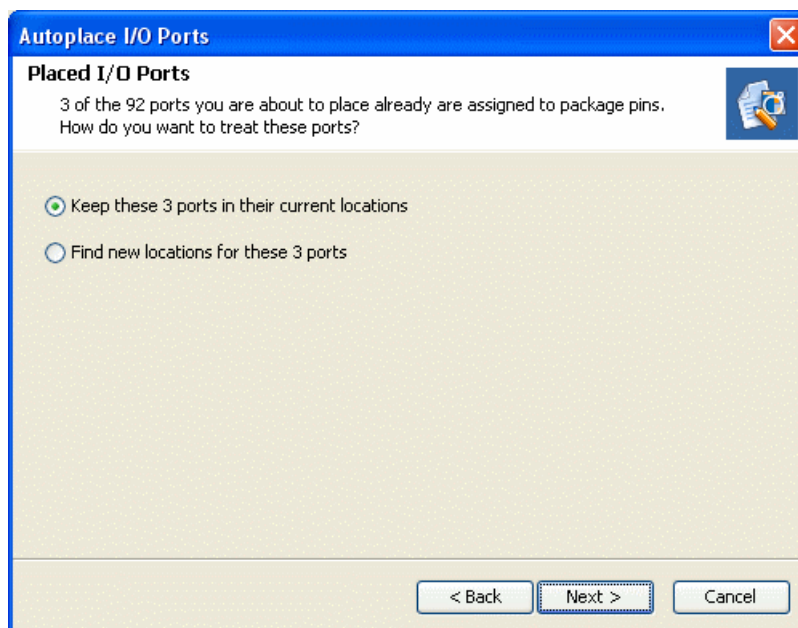


図 8-25 : Autoplace I/O Ports ウィザード

ギガビット トランシーバーの I/O ポートの配置

ギガビット トランシーバー (GTX) をより適切に管理するため、I/O Planning ビュー レイアウトのビューでは、選択、配置、移動の実行時に、関連する 2 つの I/O 差動ペアと GTX ロジック オブジェクトが自動的にグループ化されます。GTX の複数のオブジェクトは 1 つのオブジェクトとして選択され、まとめて移動されるので、GTX リソースの無効な配置は禁止されます。

DRC がオンの場合、ポート配置中に、GTX の周囲のノイズの影響を受けやすい I/O ピンが自動的に配置禁止になります。詳細は、[301 ページの「インタラクティブなデザイン ルール チェックの使用」](#)を参照してください。

I/O 配置制約の削除

配置制約を削除するには、配置されたロジックを右クリックし、[Unplace] をクリックします。

選択した配置制約の削除については、[第 10 章の「LOC 制約および BEL 制約の設定」](#)を参照してください。

クロック ロジックの配置

BUFG、DCM、MMCM、BUFR、IDELAYCTRL などのグローバルおよびリージョナル クロック関連のロジックは、[308 ページの「\[Clock Resources\] ビューの使用」](#)で説明するように、[Clock Resources] ビューを使用して手動で配置できます。クロック ロジックは、[Device] ビューでも手動で配置できます。[Device] ビューには、デバイス特有のリソースすべてに対する適切なロジック サイトが表示されます。

BUFG などのロジック インスタンスを配置するため検索するには、次の手順に従います。

1. [Edit] → [Find] をクリックします。

2. [Find] フィールドで [Instances] を選択し、インスタンスを検出するのに必要な検索条件を定義します。
3. [Find Results] ビューの検索結果を使用して、インスタンスを [Clock Resources] または [Device] ビューにドラッグし、最適なデバイス リソースに割り当てます。
[Find] の使用方法は、第 4 章の「[Find Results] ビュー」を参照してください。

グローバル クロック バッファ (BUFGCTRL) などのデバイス上の物理リソースも検索できます。
[Find] フィールドで [Sites] を選択し、検索条件を定義します。[Find Results] フィールドの結果を使用して、[Clock Resources] または [Device] ビューでデバイス リソースをハイライトします。

[Clock Resources] ビューの使用

[Clock Resources] ビューでは、ザイリンクス 7 シリーズ FPGA および Virtex-6 デバイスがサポートされ、BUFR、BUFIO、BUFG、MMCM、GT などのリージョナル クロック リソースおよびグローバル クロック リソース間の関係が示されます。[Clock Resources] ビューのスプレッドシート形式の表示には、デバイス リソースが簡略化されていますが、これらのリソース間の相対的な位置関係は正しく保持されます。[Device] ビューに表示される FPGA デバイスの詳細のほとんどは、[Clock Resources] ビューには表示されません。

310 ページの図 8-26 に、Virtex-6 LX75 デバイスの [Clock Resources] ビューを示します。[Clock Resources] ビューのインターフェイスには、次が表示されています。

- デバイスには 6 つのクロック領域が 2x3 の行列で配置され、デバイスの左下 X0Y0 から右上 X1Y2 に向かって番号が付けられています。
- これらの各クロック領域には CCIO (クロック兼用 I/O)、BUFIO、BUFR を含む I/O バンクがあります。
- デバイス自体は、4 つのクロック領域を含む上半分と、2 つのクロック領域を含む下半分に分割されています。
- デバイスの中央の列は、MMCM と BUFG を含むクロック マネージメント タイル (CMT) で、デバイスのグローバル クロックが管理されます。
- デバイスの左側のクロック領域には、外側の I/O バンク (IOOL) と中央の I/O バンク (IOCL) の両方が含まれ、デバイス右側のクロック領域には中央の I/O バンク (IOCL) のみが含まれ、デバイスの右端にギガビット トランスファ (GT) の列があります。

[Clock Resources] ビューは、展開または非展開して、必要な情報のみを表示させることができます。

- [Collapse All] または [Expand All] をクリックすると、[Clock Resources] ビューの表示レベルを非表示または表示できます。
- + または - マークをクリックすると、特定レベルを表示または非表示にできます。

[Clock Resources] ビューで特定のクロック領域または I/O バンクの名前をクリックすると、そのオブジェクトが [Device] などのほかのビューでも選択されます。これにより、[Clock Resources] ビューから特定のオブジェクトをデバイス、パッケージ、ネットリストですばやく見つけることができます。

ツールバーの [Automatically scroll to selected objects] をクリックすると、ほかのビューで選択したインスタンスやリソースが [Clock Resources] ビューで選択され、画面がその場所に自動的にスクロールされます。これにより、デバイスの特定リソースを [Clock Resources] ビューですばやく見つけることができます。この機能をオフにすると、ほかのビューでオブジェクト選択されるたびに、表示されるリソースが変わるのを避けることができます。

デザイン インスタンスの配置

[Clock Resources] ビューには、各 I/O バンク、CMT、GTX バンクの下にサイトとインスタンスの 2 つの列が表示され、デバイス リソースとそれに割り当てられるデザイン インスタンスが表示されます。

[Find Results]、[Schematic]、[Netlist]、または [I/O Ports] ビューでロジック インスタンスを選択して、デバイス リソースに配置できます。

ロジック インスタンスを選択し、[Clock Resources] ビューの適切なデバイス リソースのインスタンス列にドラッグします。

[Clock Resources] ビューにインスタンスをドラッグすると、そのインスタンスを配置できないサイトではカーソルがスラッシュの入った円に変わり、配置できるサイトでは長方形が表示されます。



デザインからインスタンスを配置する際、グローバル クロックおよびリージョナル クロックのツリー構造に関する特定の規則や制限が適用されます。これらの規則や制限の詳細は、そのデバイスのクロック リソース ガイドを参照してください。

The screenshot displays the 'Clock Resources' window for a xc6vlx75tff784-3 device. The interface is organized into a grid of panels, each representing a different clock region or I/O bank. The panels are color-coded and labeled as follows:

- Top Half:**
 - Clock Region X0Y2:** Contains tables for I/O Bank 16 and I/O Bank 26, listing resources like BUFR_X0Y5, BUFR_X1Y5, and various BUFIODQ5 instances.
 - Clock Region X1Y2:** Contains tables for I/O Bank 36 and GT Bank 116, listing resources like BUFR_X2Y5, BUFR_X2Y4, and GTXE1_X0Y11.
- Bottom Half:**
 - Clock Region X0Y1:** Contains tables for I/O Bank 15 and I/O Bank 25, listing resources like BUFR_X0Y3, BUFR_X1Y3, and various BUFIODQ5 instances.
 - Clock Region X1Y1:** Contains tables for I/O Bank 35 and GT Bank 115, listing resources like BUFR_X2Y3, BUFR_X2Y2, and GTXE1_X0Y7.

Each table within these panels has two columns: 'Site' and 'Instance'. The 'Site' column lists the resource name and its location (e.g., BUFR_X0Y5, BUFIODQ5_X0Y1), while the 'Instance' column is currently empty. The panels are further grouped by 'CMT @ Y=2' (Top Half) and 'CMT @ Y=1' (Bottom Half).

図 8-26 : [Clock Resources] ビュー

[Device] ビューでのクロック ロジックの配置

クロック ロジックを手動で配置するには、次の手順に従います。

1. [Device] ビューを拡大表示し、ロジックを配置するデバイス サイトを表示します。
2. ツールバーから [Instance Drag & Drop Modes] ボタンをクリックし、[Create Site Constraint Mode] を選択します。
3. [Find Results]、[Schematic]、[Netlist]、または [I/O Ports] ビューで配置するロジック インスタンスを選択し、[Device] の適切なデバイス リソースにドラッグします。

図 8-27 に、手動のクロック配置の例を示します。

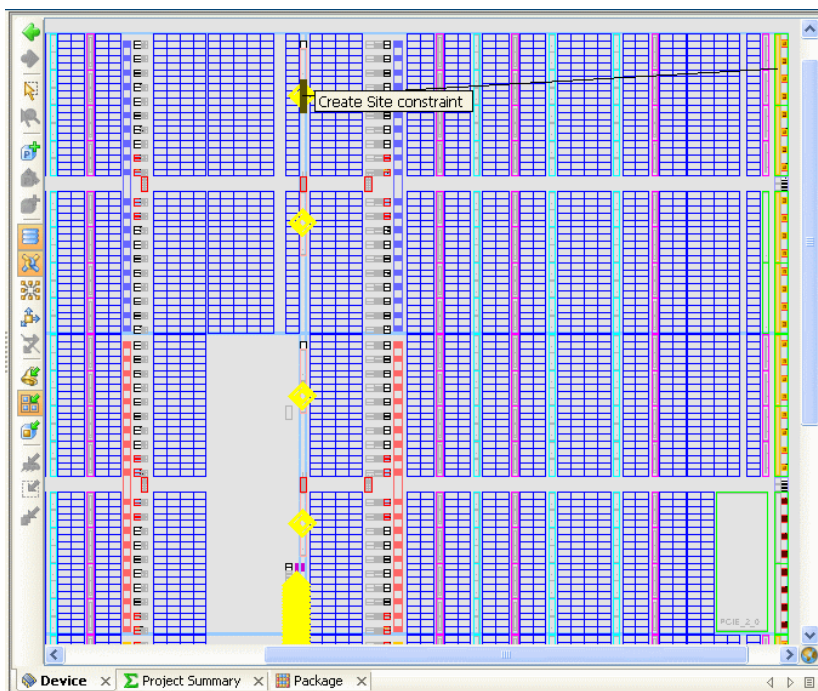


図 8-27 : クロック ロジックの手動配置

I/O とクロック ロジックの配置の検証

このセクションでは、I/O ポートおよびクロックに関する DRC の実行について説明します。ネットリストおよびフロアプランに関する DRC の実行については、[466 ページの「I/O ポート/クロック ロジック/配置 DRC の詳細」](#)を参照してください。

I/O ポートおよびクロック ロジック関連の DRC の実行

実行する DRC ルールを選択するには、次の手順に従います。

1. 次のいずれかを実行します。
 - [Tools] → [Run DRC] をクリックします。
 - Flow Navigator で [Report DRC] をクリックします。

[312 ページの図 8-28](#) に示す [Run DRC] ダイアログ ボックスが開きます。

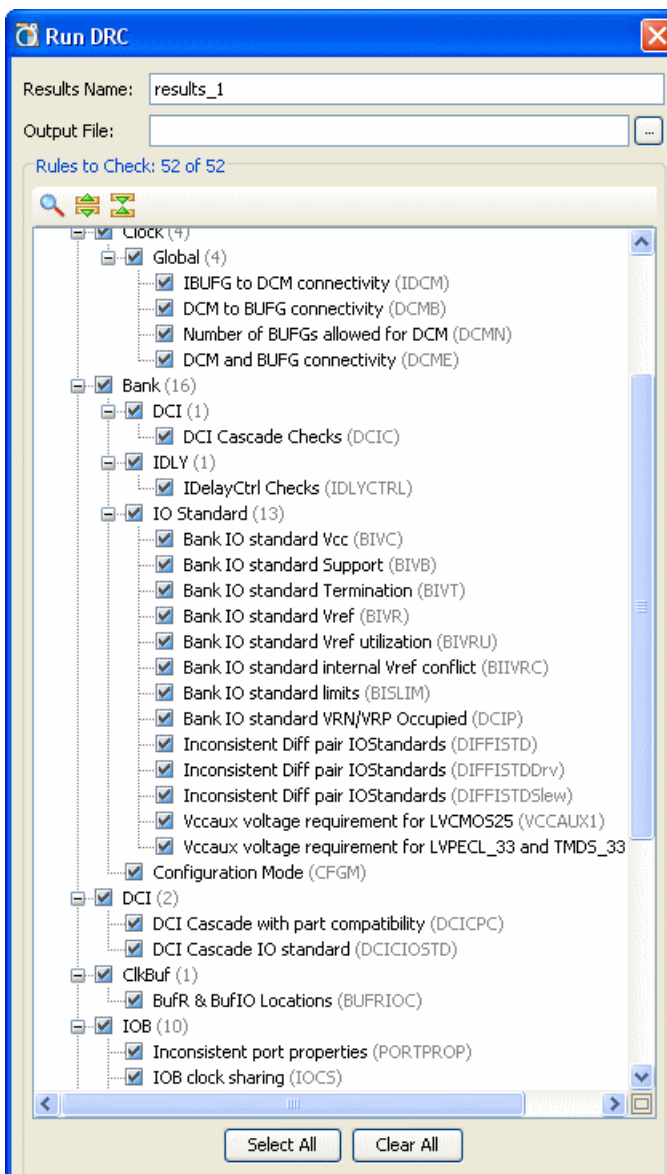


図 8-28 : [Run DRC] ダイアログ ボックス :I/O ピンおよびクロック関連の DRC ルール

2. [Results Name] フィールドを確認または編集します。ここで指定した名前が [DRC Results] ビューに表示されるので、デバッグ時に区別しやすい名前を付けます。この名前は出力ファイル名としても使用されます。
3. [Rules to Check] グループ ボックスで、チェックするルールのチェック ボックスをオンにします。各ルールの詳細は、466 ページの「I/O ポート/クロック ロジック/配置 DRC の詳細」を参照してください。
 - [Expand All] ツールバー ボタンをクリックするか、各カテゴリまたはデザイン オブジェクトの横のプラス記号 (+) をクリックすると、階層が展開します。
 - すべての DRC を実行するには、デザイン オブジェクトの横のチェック ボックスをすべてオンにします。
 - すべてのデザイン オブジェクトのすべてのルールをチェックする場合は、[All Rules] の横のチェック ボックスをオンにします。

4. [OK] をクリックし、選択した DRC チェックを実行します。

DRC エラーの表示

DRC で違反が検出された場合、図 8-29 に示す [DRC Results] ビューが表示されます。[DRC Results] ビューには、検出されたルール違反が [Run DRC] ダイアログ ボックスで定義された違反カテゴリ別に表示されます。ルール違反はその重要度によって分類され、エラー、警告、情報などのメッセージが色分けされて表示されます。

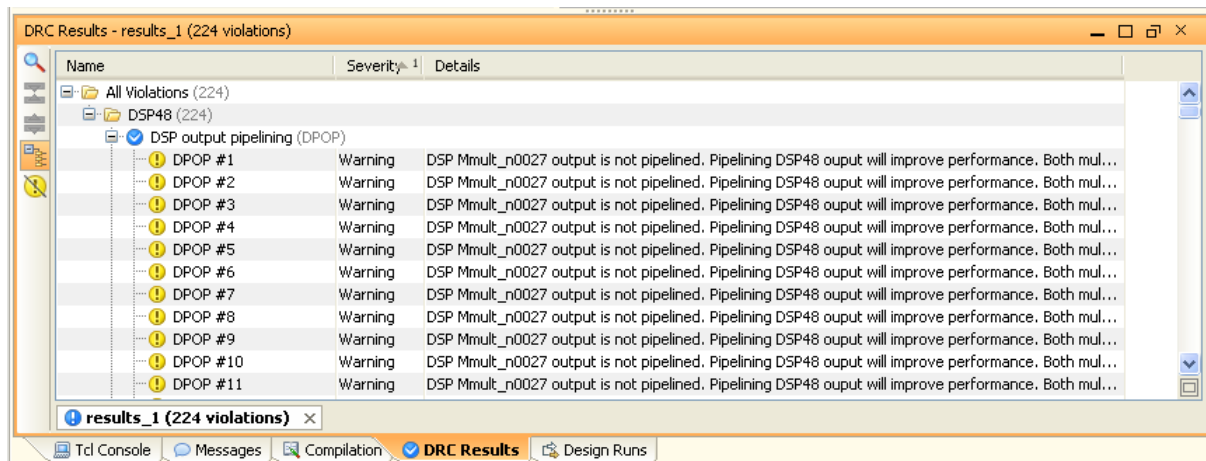


図 8-29 : DRC エラーのハイライト

違反の種類は、次のとおりです。

- 情報のみ：問題となる可能性のある事項 (黄色)
- 警告：対処が必要な可能性のある問題 (オレンジ)
- クリティカル警告：対処する必要のある問題 (オレンジ)
- エラー：インプリメンテーションの障害となる問題(赤)

警告および情報メッセージを非表示にし、エラーのみを表示するには、ツールバーの [Hide Warnings and Informational Messages] ボタンをクリックします。



[Severity] 列のヘッダーをクリックすると、違反の重要度で並べ替えることができます。

- 列ヘッダーを一度クリックすると重要度の低い方から並べられます。
- もう一度クリックすると重要度の高い方から並べられます。

詳細は、第 4 章の「ツリー表形式のビュー」を参照してください。

[DRC Results] ビューで違反メッセージを右クリックし、[Violations Properties] をクリックすると、[Violations Properties] ビューが表示されます。このビューで、DRC 違反の概要と、ルールに違反しているデザイン エレメントの詳細を確認できます。

[Violations Properties] ビューの [Details] タブには、DRC に違反するデザイン オブジェクトへのリンクがあります。リンクをクリックすると、そのデザイン オブジェクトが [RTL Netlist] ビュー、[Device] ビュー、[Schematic] ビュー、およびソース RTL ファイルで選択されます。

RTL デザインへの変換

I/O ピン配置プロジェクトは、RTL ソース ベースのプロジェクトの基盤または開始点として使用できます。

I/O ポートが定義されてパッケージ ピンに配置されたら、I/O ピン配置プロジェクトを RTL ソースプロジェクトに変換できます。現在のポート定義は、指定した Verilog または VHDL のいずれかで RTL デザインの最上位モジュールを作成するために使用されます。差動ペア バッファは最上位モジュールに追加されます。バス定義も RTL に含まれます、プロジェクト属性は、RTL ソース プロジェクト タイプを反映するように変更されます。

注記 : RTL プロジェクトは、I/O ピン配置プロジェクトに変換し戻すことはできません。

プロジェクトを変換するには、次のいずれかのコマンドを使用します。

- メイン メニューから [File] → [Migrate to RTL] をクリック
- Flow Navigator から [Migrate to RTL] をクリック

図 8-30 のような [Migrate to RTL] ダイアログ ボックスが表示されます。

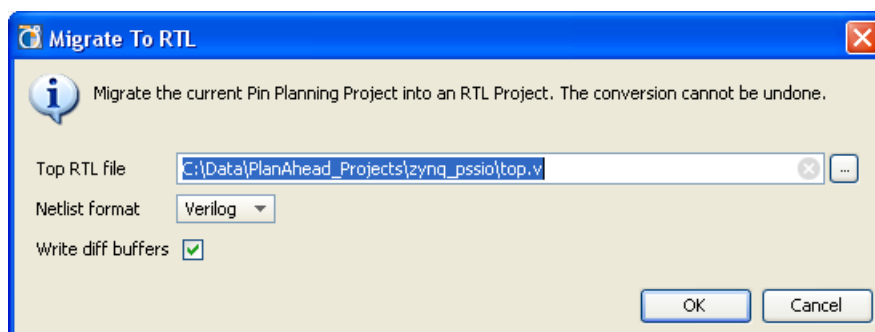


図 8-30 : [Migrate to RTL] ダイアログ ボックス

このダイアログ ボックスには、次のようなオプションがあります。

[Top RTL file] : デザインの最上位モジュールを作成するための Verilog (.v) または VHDL (.vhd) ファイルを指定します。この HDL ファイルには、モジュール定義、ポート定義、方向、バス ピンの幅が含まれます。

[Netlist format] : 最上位モジュールの Verilog または VHDL フォーマットを指定します。

[Write diff buffers] : 最上位モジュールの定義一部として差動ペア バッファを書き出します。これにより、I/O ピン配置プロジェクトで定義した差動ペアはすべて保持されます。

[OK] をクリックすると、現在の I/O ピン配置プロジェクトが変換され、RTL ソース ベースのプロジェクトに変換されます。これで、このデザインをデザインの基盤として使用して、プロジェクトにソースを追加していくことができます。詳細は、第 5 章「エラボレート済み RTL デザイン」を参照してください。

I/O ピンとパッケージ データのエクスポート

I/O ピンとピン パッケージ情報は次に説明する方法でエクスポートできます。

パッケージ ピンの情報のエクスポート

デバイス パッケージ ピンの情報は、CSV 形式のファイルにエクスポートできます。エクスポートされるファイルには、デバイスに含まれるすべてのパッケージ ピンの情報と、デザイン特定の I/O ポート割り当てとそのコンフィギュレーションが含まれます。

エクスポートされたリストのパッケージ ピン セクションを基に、I/O ポートをスプレッドシートで定義できます。エクスポートされた CSV 形式のファイルの情報は、[292 ページの「I/O ポートの定義と設定」](#)を参照してください。

I/O ポート リストのエクスポート

I/O ポート リストは、ファイルにエクスポートし、RTL のコード記述や PCB 回路図シンボルの作成に使用できます。

1. I/O ポート リストの情報をエクスポートするには、[File] → [Export] → [Export I/O Ports] コマンドをクリックします。[図 8-31](#) に示す [Export I/O Ports] ダイアログ ボックスが開きます。

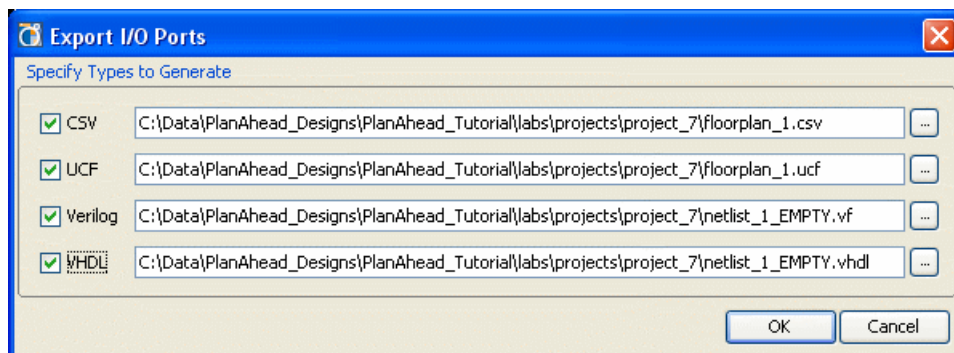


図 8-31 : I/O ポート リストのエクスポート

I/O ポートは、種類とパスを指定してエクスポートできます。エクスポートできるファイル形式は、CSV、UCF、Verilog、VHDL のいずれかです。

IBIS モデルのエクスポート

PCB 設計者がシステム レベルのシグナル インテグリティの理解を深めるには、IBIS (I/O Buffer Information Specification) モデルを使用したデザインをシミュレーションする必要があることがよくあります。クロストーク、グランド バウンス、同時スイッチ ノイズ (SSN) などのシグナル インテグリティ問題について考慮する必要があります。IBIS モデルを使用すると、パッケージ デバイスの I/V カーブおよび寄生情報などのキャラクタライゼーションが実行しやすくなります。

ザイリンクスでは汎用の IBIS モデルを提供しています。これらは、<http://japan.xilinx.com/support/download> から入手できます。

ザイリンクス 7 シリーズの場合、[Export IBIS Model] コマンドを使用すると、デザインおよびピンごとのパッケージ データから IBIS モデルを生成できます。PlanAhead ツールは、デザインのネットリストおよびインプリメンテーションの詳細を使用し、その情報と使用可能なピンごとの寄生パッケージ情報を統合して、そのデザイン専用のカスタム IBIS モデルを作成します。

[Elaborated Design]、[Synthesized Design]、または [Implemented Design] ビューを開いていると、デザイン解析に使用する IBIS ファイルをエクスポートできます。

[File] → [Export] → [Export IBIS Model] をクリックします。316 ページの図 8-32 に示すように、[Export IBIS Model] ダイアログ ボックスが開きます。

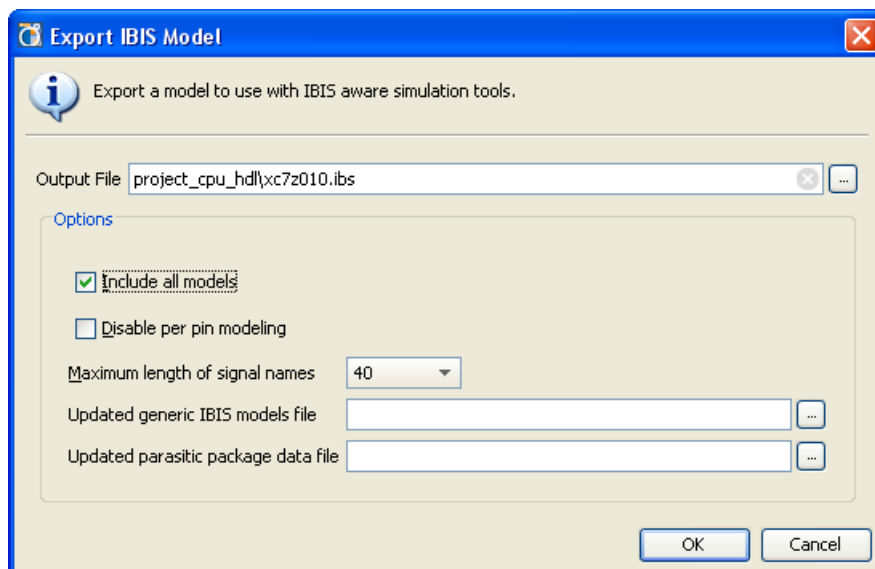


図 8-32 : [Export IBIS Model] ダイアログ ボックス

このダイアログ ボックスのオプションは次のとおりです。

- [Output File] : 出力される IBIS ファイルのファイル名とパスを指定します。
- [Include all models] : このデバイスで使用可能なすべての I/O バッファを含めるかどうか指定します。デフォルトでは、デザインで使用されるバッファ モデルのみが含まれます。
- [Disable per pin modeling] : パッケージのピンごとのモデル記述を含めないようにします。これは、デバイスのダイ パッドからパッケージ ピンまでのパスです。オフにすると、パッケージは、すべてのピンに適用され、IBIS ファイルの [Package] セクションで定義される 1 つの RLC 伝送ライン モデルに削減されます。
- [Maximum length of signal names] : 指定した制限文字数まで信号名を削減します。
 - 40 : IBIS バージョン 4.2 でサポートされる 40 文字まで信号名が省略されます (デフォルト)。
 - 20 : 20 文字まで信号名が省略されます。
 - Unlimited : 信号名は省略されません。
- [Updated generic IBIS model file] : オプションで、デバイス用の IBIS モデルを指定できます。これは、parts ディレクトリで検出された IBIS モデルを上書きするために使用します。
 注記 : インストールで IBIS モデルが含まれないデバイスを使用する場合は、ここで IBIS モデル ファイルを指定する必要があります。
- [Updated parasitic package data file] : オプションで、ピンごとの抽出に使用する寄生パッケージ ファイル (.pkg) を指定できます。これは、インストール階層の parts ディレクトリで検出された寄生パッケージ ファイルを上書きするために使用します。
 注記 : インストールで IBIS モデルが含まれないデバイスを使用する場合は、ここで寄生パッケージ ファイルを指定する必要があります。

ノイズ解析予測の使用

PlanAhead には、異なるデバイスの I/O に関連するスイッチ ノイズ レベルを解析する機能があります。この解析機能を使用するには、Flow Navigator から [Run Noise Analysis] をクリックするか、[Tools] → [Run Noise Analysis] をクリックします。デバイスのターゲット デバイスによって、同時スイッチ ノイズ (SSN) 解析または同時スイッチ出力 (SSO) 解析が実行されます。

- ザイリンクス 7 シリーズ FPGA、Virtex-6、Spartan-6 デバイスでは、SSN 解析が実行されます。SSN 解析の詳細は、「[SSN 解析の実行](#)」を参照してください。
- Spartan-3、Virtex-4、および Virtex-5 デバイスでは、SSO 解析が実行されます。詳細は、[322 ページ](#)の「[WASSO 解析の実行](#)」を参照してください。

SSN 解析の実行

PlanAhead では、Spartan-6、Virtex-6、Virtex-7、Kintex™-7 および Artix™-7 デバイスに対しては、SSN 解析を使用して同時スイッチ出力ノイズを詳細に解析します。⁽¹⁾ SSN 解析では、同時スイッチ出力による I/O バンクのほかの出力 (および Spartan-6 では入力) への影響が予測されます。I/O バンク別の電気特性が組み込まれており、SSN のパッケージ効果をモデル化します。

I/O は I/O バンクにグループ化され、各バンクに固有の電源分配ネットワークがあり、それぞれスイッチング アクティビティに対する応答は異なります。パッケージに装着された FPGA 内の電源分配ネットワークのノイズへの応答はそれぞれ異なるため、デザインで使用されている I/O 規格や I/O 数だけでなく、デバイスの電源システムの応答も理解することが重要です。

ザイリンクスでは、すべてのバンクを 3 次元抽出およびシミュレーションを使用して特性化しています。この情報は SSN 解析に組み込まれており、デバイスのスイッチ プロファイルが概算され、システムの電源ネットワークへの影響と I/O バンク内のほかの出力への影響が予測されます。

出力スイッチがインターフェイス ノイズ マージンにどのように影響するかを正確に予測するには、SSN 予測ツールが最も適しています。その計算と結果は、さまざまなパターンを網羅しています。これらの予測値は、デザインで発生する可能性のあるノイズに関連した問題を特定するためのもので、最終デザインを確定するための検証方法としては使用しないでください。

1. Virtex-7、Kintex-7 および Artix-7 デバイスでは、暫定的な SSN 解析が実行されます。

SSN 解析を実行するには、次の手順に従います。

1. Flow Navigator または [Tools] メニューから [Run Noise Analysis] をクリックします。図 8-33 に示す [Run WASSO Analysis] ダイアログ ボックスが開きます。

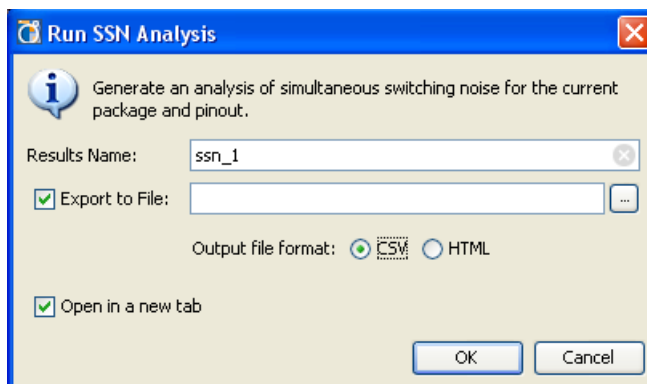


図 8-33 : [Run SSN Analysis] ダイアログ ボックス

注記 : [Run Noise Analysis] コマンドをクリックすると、ターゲット デバイスに応じて SSN 解析または SSO 解析が実行されるので、開くダイアログ ボックスも実行される解析によって異なります。詳細は、322 ページの「WASSO 解析の実行」を参照してください。

2. [Results Name] に [SSN Results] に表示する名前を入力します。
3. [Export to File] チェック ボックスをオンにし、[Output File] フィールドに出力ファイル名を入力し、外部絵レポート ファイルを出力するディレクトリを指定します。
4. [Output Format] に CSV か HTML を指定します。
5. [OK] をクリックします。

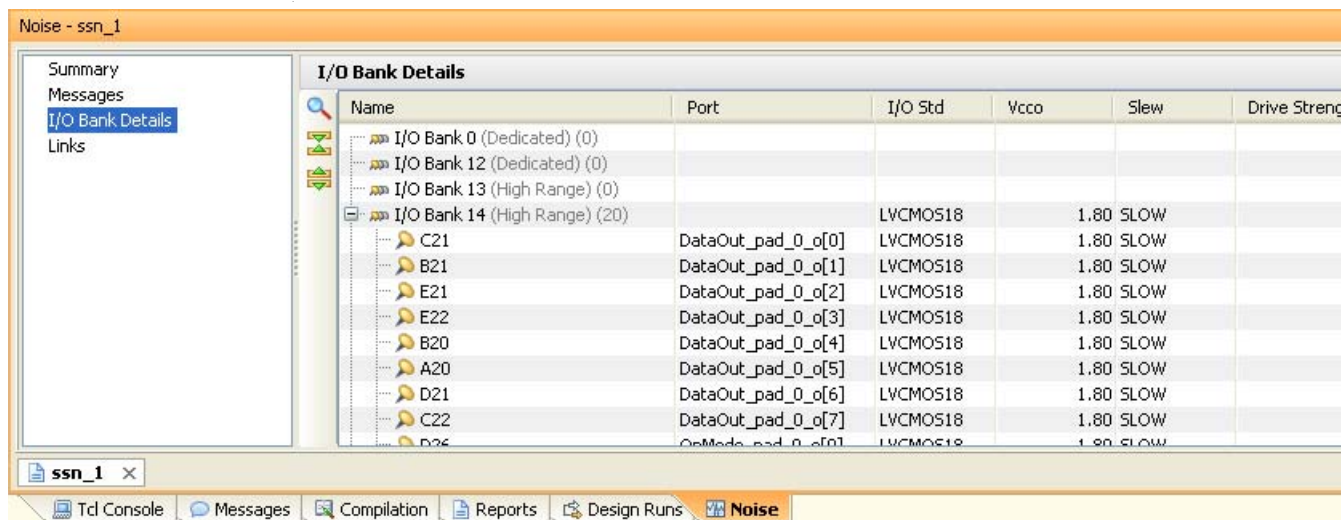


図 8-34 : [SSN Results] ビュー

SSN 結果の表示

SSN 解析が完了すると、[図 8-34](#) に示す [DRC Results] ビューが表示されます。

- [Summary] : [Results] ビューの左側の [Summary] ビューに SSN 解析に関するさまざまなレベルの情報が表示されます。[I/O Bank Details] は表形式のビューで、コマンドの特別な所見が表示されます。
- [Name] : デバイスで使用可能な I/O バンクを表示します。各 I/O バンクには、バンクの空き状況を示すピンのアイコンと、解析で問題が検出されなかったことを示すチェック マークまたはエラーを示す赤い丸が表示されます。
- [Group] (Virtex-6 のみ) : バンクに割り当てられている I/O 規格などでまとめられているピンのグループと、そのステータスを表示します。グループは、割り当てられている I/O 規格、駆動電流、スルー レート、および位相に基づいて自動的に決定されます。
- [I/O Std]、[Vcco]、[Slew]、[Drive Strength] : ポートまたはバンクの値 (I/O 規格、Vcco、スルー レート、駆動電流) を表示します。
- [Noise (V)]
 - [Contributed] (Virtex-6 のみ) : グループの I/O 規格、駆動電流、スルー タイプで生成されたグループごとの SSN 総計です。
 - [Bank Total] (Virtex-6 のみ) : バンクまたはグループに対し予測される SSN 総計です。バンクのグループに対し複数の位相が指定されている場合、異なる位相を持つグループの SSN が別々に累計され、最大値がレポートされます。SSN 計算はそのバンクの出力に限定されているため、1 つの SSN バンクの合計は別のバンクの合計には影響しません。この列では、どの I/O グループで SSN 値が最大か、マージンがどれぐらい使用されているのかを確認できます。
- [Off-Chip Termination] : このフィールドには、各 I/O 規格のデフォルト終端がある場合は、それが自動的に表示されます。たとえば、LVTTL (2mA、4mA、6mA および 8mA) の場合、終端はなしと表示されますが、LVTTL (12mA、16mA、および 24mA) の場合は、50 オームの遠端パラレルが V_{TT} と表示されます。この終端が使用されると、駆動電流が 12mA の信号の場合、2mA ~ 8mA に比べて使用可能なノイズ マージンが少なくなります。

Virtex-4、Virtex-5、Virtex-6、Spartan-6 およびすべての 7 シリーズ デバイスでこの仮定が使用されます。

この場合、「None」と表示されるか、予測または定義されるオフチップ終端スタイルが記述されます。たとえば、FP_VTT_50 は、遠端、 V_{TT} に対する 50W の終端を示します。

すべての終端スタイルのリストは、[付録 E 「その他のリソース」](#) に示すそのデバイスの Select IO™ リソース ユーザー ガイドを参照してください。

設定を変更するには、次のいずれかを使用します。

- [「CSV ファイルのインポート」](#) に示す CSV ファイルのインポート機能
または
- I/O ポートの表のプルダウン リスト
- [OUT_TERM] (Spartan-6 のみ) : ポートの OUT_TERM 属性設定が定義されている場合はそれを表示します。ほとんどの場合、「NONE」と表示されます。

OUT_TERM の詳細は、[付録 E 「その他のリソース」](#) に示す『Spartan-6 FPGA SelectIO リソース ユーザー ガイド』(UG381)を参照してください。

- [Margin (V)]
 - [Available] (Virtex-6 のみ) : 特定の I/O 規格で信号が 1 に切り替わるときに許容されるノイズ マージンを示します。I/O 規格で指定されている DC 論理レベルに基づいており (量的な情報は考慮されない)、High の信号の最小駆動電流が JEDRC 入力しきい値をどれだけ上回っているかを表します。これらのマージン値は、駆動電流が最も弱い状態、JEDEC 仕様終端、および規格の標準レシーバー要件で算出されています。ここでは、保護範囲を提供するため、控えめな前提が使用されています。
 - [Remaining] : バンクの SSN をすべて考慮した後のノイズ マージン値の残りを表示します。
- [Result] : 問題が検出されなかった場合は「PASS」、検出された場合は「FAIL」が赤字で表示されます。
- [Notes] : I/O バンクまたはグループについての情報が表示されます。

SSN 結果には SSN 解析が実行されたときのデザインの状態が反映されており、ダイナミックなレポートではありません。

SSN 結果の改善

違反が発生した場合に SSN 結果を改善するには、次の手順に従います。

- 違反が発生しているグループに SSN の影響をあまり受けない I/O 規格を使用します。低い駆動電流に変更、パラレル終端の DCI I/O 規格を変更、または低いクラスのドライバーに変更すると (SSTL Class II から SSTL Class I に変更するなど)、SSN を改善できます。
- 違反が発生しているピンを複数のバンクに分散させます。このようにすると、1 つのバンクの電源システムで問題のある出力数を低減できます。
- 違反が発生しているグループを複数の同期位相に分散させます。位相グループについては、メモを参照してください。
- [Result] に「FAIL」と表示されている場合は、位相グループを同時スイッチのポートに割り当てます。

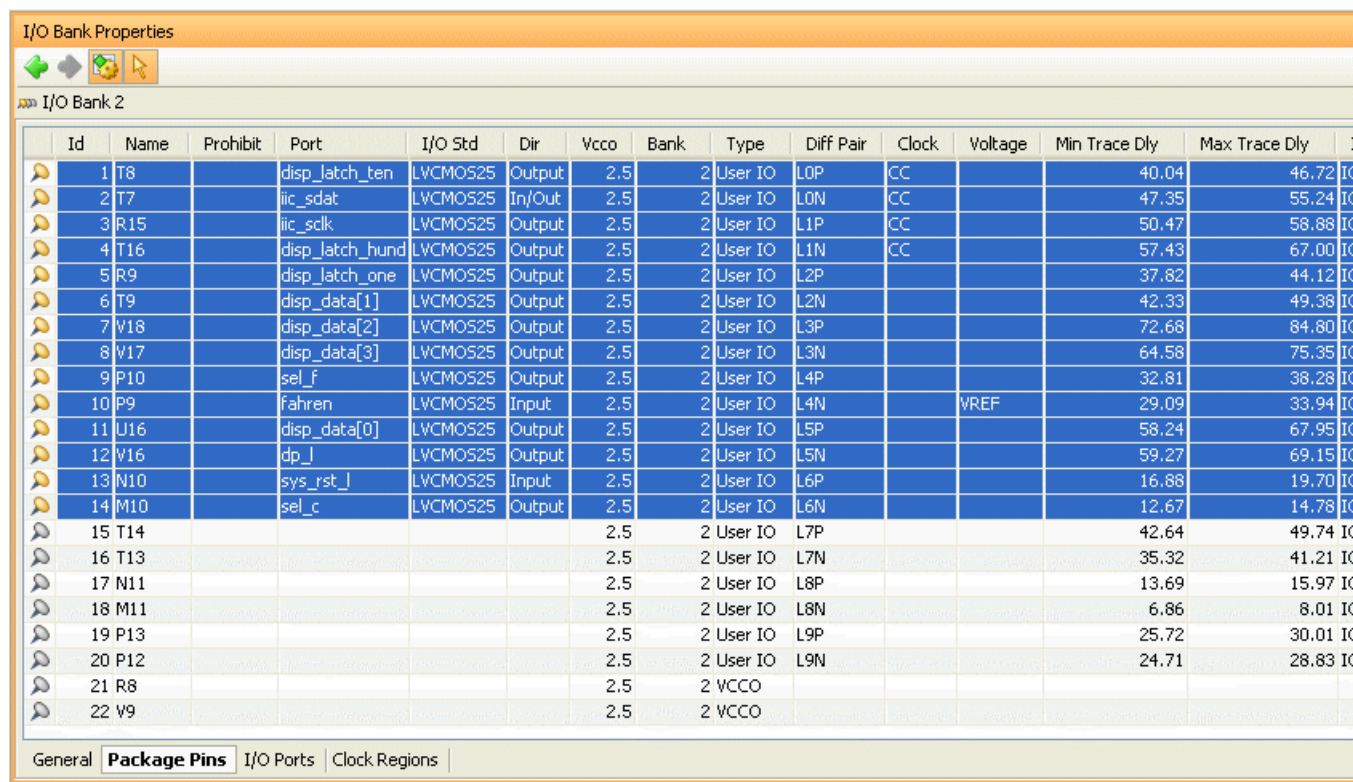
注記 : 位相グループは、Virtex-6 デバイスのソフトウェアと SSN 計算でのみサポートされます。Spartan-6 デバイスの場合、位相シフトを使用すると、SSN に関するデバイス パフォーマンスは改善されますが、改善されたパフォーマンスはソフトウェア計算には含まれません。Virtex-6 の場合は、[321 ページの「SSN の I/O ポート スイッチ位相グループの定義」](#)を参照してください。

- 違反が発生しているグループを DDR レートの場合は 90 度、SDR レートの場合は 180 度位相シフトさせます。これで、問題のある出力スイッチの半分の位相をずらすことができます。
- Spartan-6 デザインで SSN のエラーになった場合、一部のエラーは無視することができます。詳細は、[アンサー 36141](#) を参照してください。詳細は、[付録 E「その他のリソース」](#)に示す『Spartan-6 FPGA SelectIO リソース ユーザー ガイド』(UG381) の「SSO の影響を低減するピン配置」(Pin-Planning to Mitigate SSO Sensitivity) セクションを参照してください。

SSN 結果の I/O バンク プロパティの表示

[SSN Results] ビューで I/O バンクを選択すると、そのバンクに割り当てられている I/O ポート、ピン、およびグループの情報が [I/O Bank Properties] ビューに表示されます。

- [General] タブには、I/O バンクに割り当てられているポートの数とタイプが表示されます。
- [Package Pins] または [I/O Ports] タブには、バンク内のピンまたはポートの詳細情報が表示されます ([321 ページの図 8-35](#))。



	Id	Name	Prohibit	Port	I/O Std	Dir	Vcco	Bank	Type	Diff Pair	Clock	Voltage	Min Trace Dly	Max Trace Dly	
	1	T8		disp_latch_ten	LVC MOS525	Output	2.5	2	User IO	L0P	CC		40.04	46.72	IO
	2	T7		iic_sdat	LVC MOS525	In/Out	2.5	2	User IO	L0N	CC		47.35	55.24	IO
	3	R15		iic_sclk	LVC MOS525	Output	2.5	2	User IO	L1P	CC		50.47	58.86	IO
	4	T16		disp_latch_hund	LVC MOS525	Output	2.5	2	User IO	L1N	CC		57.43	67.00	IO
	5	R9		disp_latch_one	LVC MOS525	Output	2.5	2	User IO	L2P			37.82	44.12	IO
	6	T9		disp_data[1]	LVC MOS525	Output	2.5	2	User IO	L2N			42.33	49.38	IO
	7	V18		disp_data[2]	LVC MOS525	Output	2.5	2	User IO	L3P			72.68	84.80	IO
	8	V17		disp_data[3]	LVC MOS525	Output	2.5	2	User IO	L3N			64.58	75.35	IO
	9	P10		sel_f	LVC MOS525	Output	2.5	2	User IO	L4P			32.81	38.28	IO
	10	P9		fahren	LVC MOS525	Input	2.5	2	User IO	L4N		VREF	29.09	33.94	IO
	11	U16		disp_data[0]	LVC MOS525	Output	2.5	2	User IO	L5P			58.24	67.95	IO
	12	V16		dp_l	LVC MOS525	Output	2.5	2	User IO	L5N			59.27	69.15	IO
	13	N10		sys_rst_l	LVC MOS525	Input	2.5	2	User IO	L6P			16.88	19.70	IO
	14	M10		sel_c	LVC MOS525	Output	2.5	2	User IO	L6N			12.67	14.78	IO
	15	T14					2.5	2	User IO	L7P			42.64	49.74	IO
	16	T13					2.5	2	User IO	L7N			35.32	41.21	IO
	17	N11					2.5	2	User IO	L8P			13.69	15.97	IO
	18	M11					2.5	2	User IO	L8N			6.86	8.01	IO
	19	P13					2.5	2	User IO	L9P			25.72	30.01	IO
	20	P12					2.5	2	User IO	L9N			24.71	28.83	IO
	21	R8					2.5	2	VCCO						
	22	V9					2.5	2	VCCO						

図 8-35 : [I/O Bank Properties] ビュー : [Package Pins] タブ

SSN の I/O ポート スイッチ位相グループの定義

バンク内の I/O のグループによっては、ほかのグループとは異なる同期位相オフセットが設定されているものがあり、同時にスイッチすることはできません。多くのメモリ インターフェイスのデータ信号およびストローブ信号は、このようになっています。この場合、SSN 計算には位相情報が必要とされます。

位相グループは、タイミングの観点から見て、互いに同じ位相にあるポートの論理グループを指します (クロックの周波数と位相が同じであるなど)。位相グループを作成すると、グループが作成されるだけでなく、異なる位相の I/O が分離されます。

注記：位相グループは、Virtex-6 デバイスでのみサポートされます。

バンク内のグループで生成されたノイズは合計され、バンク全体のノイズが算出されます。すべての出力が同じ位相である場合、または同期関係にない場合、出力は同時にスイッチします (値が変わる)。

SSN 解析でバンクに問題が検出された場合、位相グループを使用して異なる同期位相のポートをグループ化することにより、そのバンクの合計ノイズを低減できます。

1 つの I/O ポートまたは複数の I/O ポートにスイッチ位相を設定するには、次の手順に従います。

1. I/O Planning ビュー レイアウトのいずれかのビューで、1 つまたは複数の I/O ポートを選択します。
2. [I/O Ports]、[Package Pins]、または [SSN Results] ビューで、選択したポートを右クリックして [Configure I/O Ports] をクリックします (322 ページの図 8-36)。

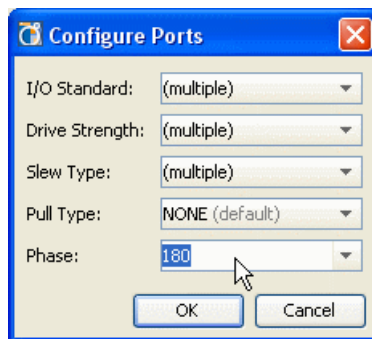


図 8-36 : [Configure Ports] ダイアログ ボックス

3. [Configure Ports] ダイアログ ボックスで、正しい I/O 規格が選択されていることを確認します。
4. ポートの位相が揃っている場合、[Phase] はデフォルトのままにするか、180 などの位相値を入力し、[OK] をクリックします。
5. 適切な位相グループを割り当てたら、SSN 解析を再実行します。

注記：非同期グループは個別の同期位相として扱わないようにしてください。非同期グループは同時にスイッチする可能性があります。

WASSO 解析の実行

PlanAhead には、加重平均同時スイッチ出力 (WASSO) のチェックが複数含まれており、I/O ピンとバンク割り当てに基づくデバイスのシグナル インテグリティが検証されます。この解析は、Spartan-6、Virtex-6、Virtex-7、Kintex-7 および Artix-7 で実行可能な SSN 解析ほど詳細ではありませんが、Spartan-3、Virtex-4、および Virtex-5 デバイスでのスイッチ ノイズを理解するのに役立ちます。

1. WASSO 解析を実行するには、Flow Navigator または [Tools] メニューから [Run Noise Analysis] をクリックします。323 ページの図 8-37 に示す [Run WASSO Analysis] ダイアログ ボックスが開きます。

注記：[Run Noise Analysis] コマンドをクリックすると、ターゲット デバイスに応じて SSN 解析または SSO 解析が実行されるので、開くダイアログ ボックスも実行される解析によって異なります。詳細は、317 ページの「ノイズ解析予測の使用」を参照してください。

2. [Output File] フィールドで、レポート ファイル名および保存場所を指定します。
3. カーソルをフィールド上に置くと、そのフィールドに入力する値の説明を示すツール ヒントが表示されます。[Device Parameters] および [Board Parameters] の値は、デザインに合わせて変更できます。

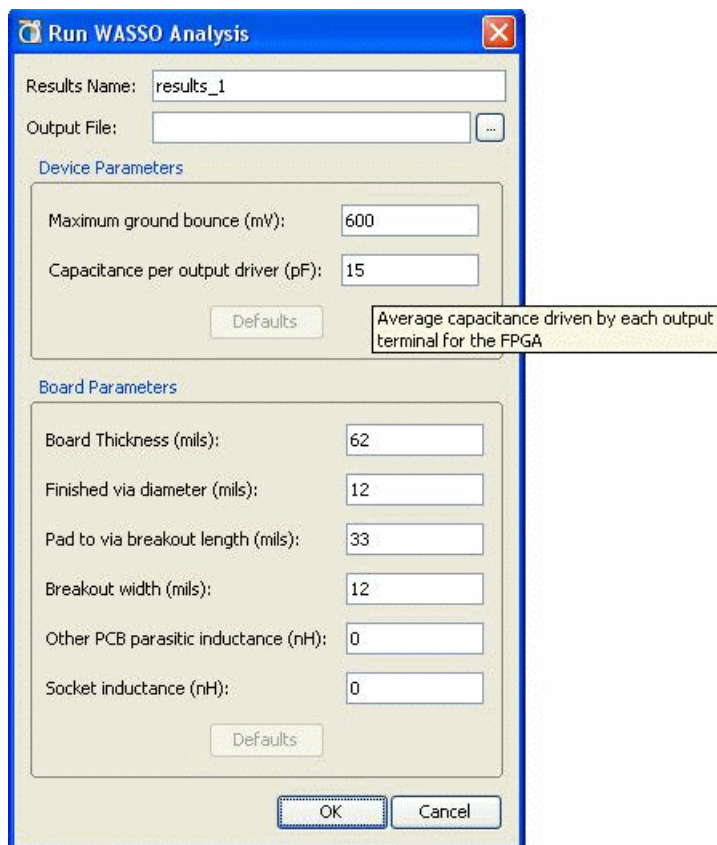


図 8-37 : [Run Wasso Analysis] ダイアログ ボックス

WASSO 解析結果の表示

まずデザイン全体が解析され、各 I/O バンクの隣接する I/O バンクに対して解析されます。[324 ページの図 8-38](#) に示す WASSO 解析結果のビューがワークスペースに表示されます。

Device Parameters			
Maximum ground bounce	600.0 mV		
Capacitance per output driver	15.0 pF		
Board Parameters			
Board Thickness	62.0 mils		
Finished via diameter	12.0 mils		
Pad to via breakout length	33.0 mils		
Breakout width	12.0 mils		
Other PCB parasitic inductance	0.0 nH		
Socket inductance	0.0 nH		
	Allowed	Utilization	Status
Package	100%	22.3%	OK
Banks			
Bank 0	100%	42.7%	OK
Bank 1	100%	20%	OK
Bank 2	100%	1.3%	OK
Bank 3	100%	9.3%	OK
Bank 4	100%	14.7%	OK
Bank 5	100%	42.7%	OK
Bank 6	100%	24%	OK
Bank 7	100%	24%	OK
Neighbors			
Bank 0,1	100%	31.3%	OK
Bank 1,2	100%	10.7%	OK
Bank 2,3	100%	5.3%	OK

図 8-38 : WASSO の結果

レポートには、I/O バンクと隣接するペアの 最大負荷、使用率、およびステータスが表示されます。

デザインのインプリメンテーション

PlanAhead™ ツールには、合成およびインプリメンテーションを一気に実行できるプッシュボタンフローと **run** を使用するフローの 2 つの機能があります。

run データは、自動的に管理されるので、次の条件を変更して繰り返し実行できます。

- 合成オプション
- インプリメンテーション オプション
- 制約

また、次の条件を変更して複数の合成およびインプリメンテーション **run** を作成できます。

- コマンド オプション
- タイミング制約
- 物理制約

複数の合成およびインプリメンテーション **run** は、順次起動するか、マルチプロセッサ マシンで同時に起動できます。インプリメンテーション **run** では、Xilinx® ISE® Design Suite ツールが使用されます。

インプリメンテーションの実行

合成済みネットリストをターゲットのザイリンクス パーツにインプリメントするには、ソース ファイルとデザイン制約を「インプリメンテーション」という複数プロセスで実行する必要があります。これには、次の手順に従ってください。

- 「インプリメンテーション **run** の定義」
- 「インプリメンテーション オプションの設定」
- 「インプリメンテーション **run** の起動」

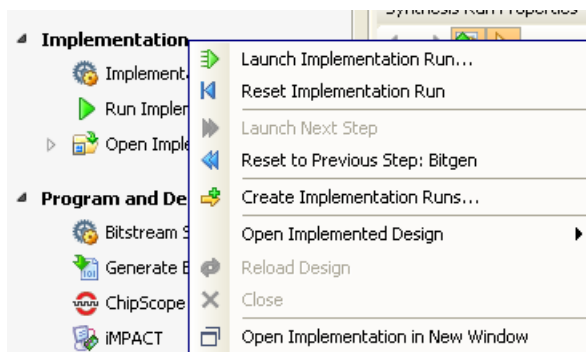
インプリメンテーション **run** の定義

新しいインプリメンテーション **run** を作成して起動し、最適な結果が得られるようさまざまなオプションを試すことができます。各 **run** は順次起動するか、複数のローカル CPU で同時に起動できます。

Linux システムでは、リモート サーバーを使用できます。詳細は、[339 ページの「リモート Linux ホストでの **run** の起動](#)」を参照してください。

インプリメンテーション **run** は次のように定義できます。

- [Flow] → [Create Runs] をクリックします。
- Flow Navigator の [Implementation] ポップアップ メニューから [Create Implementation Runs] をクリックします。



Create New Runs ウィザードが開きます。
ウィザードの最初のページは、コマンドのサマリです。

1. [Next] をクリックします。

図 9-1 に示すような [Set Up Implementation Runs] ダイアログ ボックスが開きます。

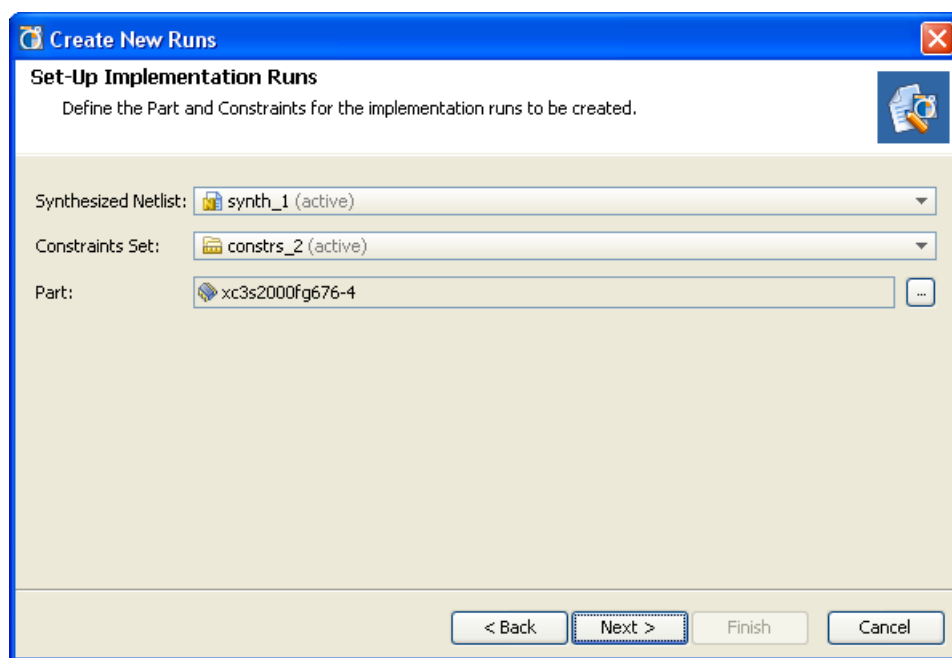


図 9-1 : run の新規作成コマンド : [Set-Up Implementation Runs] ページ

合成 run とターゲット パーツを選択します。

注記：デフォルト値は、[Create New Runs] コマンド実行時の合成またはインプリメンテーションのプロジェクト設定で定義されます。詳細は、101 ページの「プロジェクト設定」を参照してください。

2. [Next] をクリックします。

327 ページの図 9-2 に示す [Choose Implementation Strategies] ダイアログ ボックスが開きます。

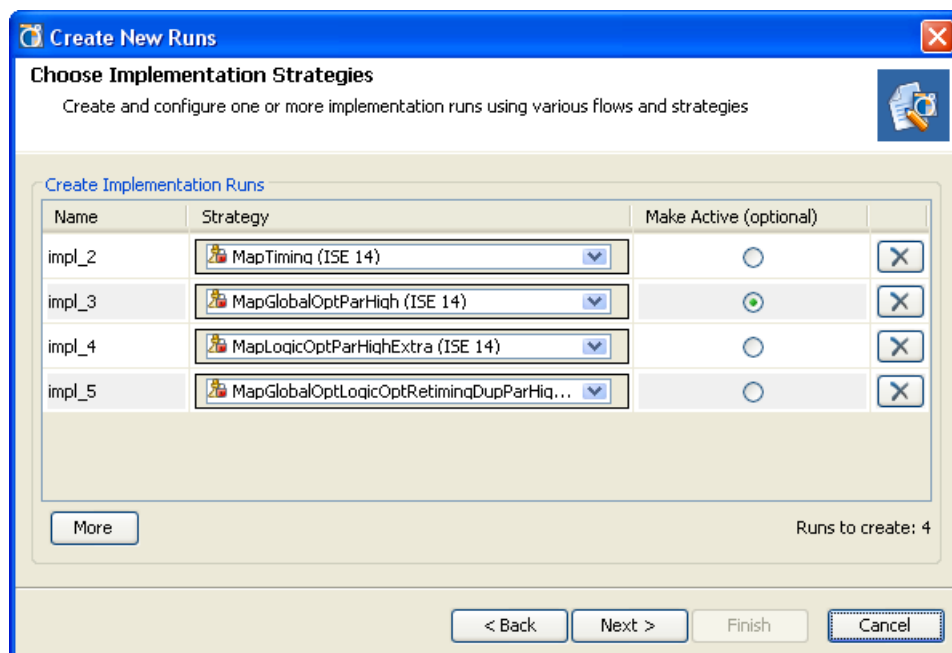


図 9-2 : [Choose Implementation Strategies] ページ

3. run の名前を入力するか、デフォルトのままにします。
4. 新しい run のストラテジを選択します。ストラテジとは、インプリメンテーション結果を制御する ISE ツールのランタイム オプションを定義した設定のことです。詳細は、第 4 章の「[合成およびインプリメンテーション ストラテジの定義](#)」を参照してください。
5. [Make Active] を選択し、新しい run をオプションでアクティブ run にできます。新規 run を複数作成する場合、アクティブにできる run は 1 つだけです。
6. [More] ボタンをクリックし、追加の run を定義します。追加 run の名前とストラテジを指定します (327 ページの図 9-2)。
7. [Next] をクリックします。

328 ページの図 9-3 に示す [Launch Options] ダイアログ ボックスが開きます。

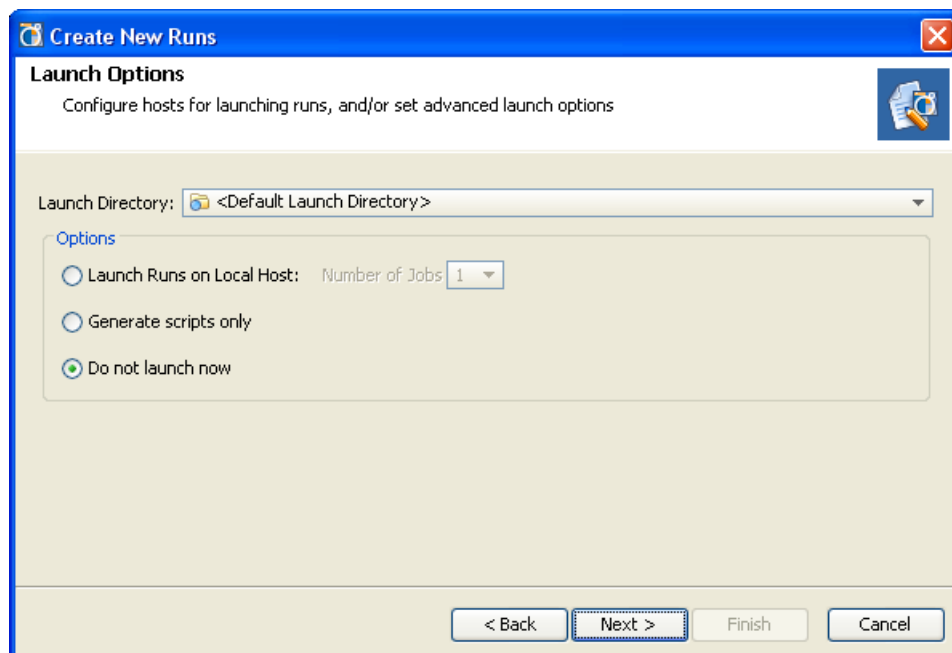


図 9-3 : [Launch Options] ダイアログ ボックス

8. インプリメンテーション run のデータを作成して保存するディレクトリを [Launch Directory] に指定します。

デフォルトの起動ディレクトリは、ローカルプロジェクト ディレクトリ構造に含まれます。インプリメンテーション run のファイルは、次のディレクトリに保存されます。

`<project_name>/<project_name>.runs/<run_name>`

注記：プロジェクト ファイルには絶対パスが記述されるので、プロジェクト ディレクトリ外のデフォルト以外の場所を指定すると、プロジェクトを移動しにくくなります。

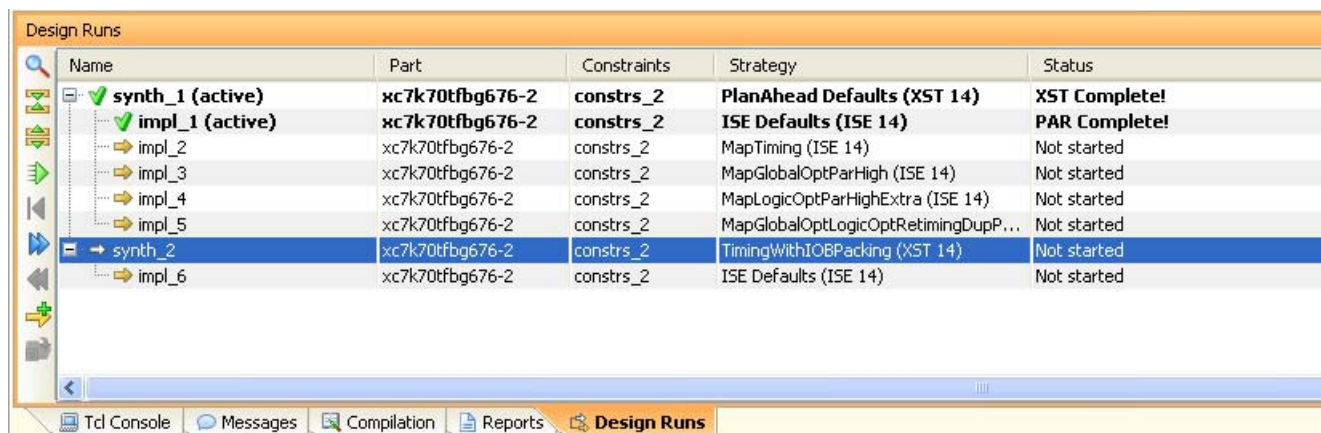
9. 次のオプションを設定します。
 - [Launch Runs on Local Host]：ローカル マシンのプロセッサで run を起動します。
 - [Number of Jobs]：run に使用するローカル プロセッサの数を指定します。このオプションは、複数の run を同時に起動する場合にのみ使用されます。各 run が各プロセッサで起動されます。このオプションでは、マルチスレッド プロセッサは使用されません。
 - [Generate scripts only]：run ディレクトリおよび run スクリプトをエクスポートおよび作成しますが、run は起動しません。スクリプトは、PlanAhead 環境外で後で実行できます。
 - [Do not launch now]：新しい run を保存はしますが、現時点では run スクリプトを開始または作成しません。
10. [Next] をクリックし、[Create New Runs Summary] を確認します。
11. [Finish] をクリックすると、定義した run が作成され、指定の起動オプションが実行されます。新しい run が [Design Runs] ビューに追加されます。

[Design Runs] ビューの使用

[Design Runs] ビューには、プロジェクトで作成された合成 run とインプリメンテーション run のすべてが表示され、それらを設定、管理、開始するためのコマンドも表示されます。

[Design Runs] ビューが表示されていない場合は、[Window] → [Design Runs] をクリックして [Design Runs] ビューを表示します。図 9-4 に、[Design Runs] ビューを示します。

各インプリメンテーション run は、合成 run の下の階層にインデントされて表示されます。合成 run には、インプリメンテーション run を複数含めることができます。プラス記号 (+) やマイナス記号 (-) をクリックすると、合成 run のツリー表示を展開したり、閉じたりできます。[Design Runs] ビューは、ツリー形式の表示です。このビューを列ごとに分類する方法については、127 ページの「ツリー表形式のビュー」を参照してください。



Name	Part	Constraints	Strategy	Status
synth_1 (active)	xc7k70tfg676-2	constrs_2	PlanAhead Defaults (XST 14)	XST Complete!
impl_1 (active)	xc7k70tfg676-2	constrs_2	ISE Defaults (ISE 14)	PAR Complete!
impl_2	xc7k70tfg676-2	constrs_2	MapTiming (ISE 14)	Not started
impl_3	xc7k70tfg676-2	constrs_2	MapGlobalOptParHigh (ISE 14)	Not started
impl_4	xc7k70tfg676-2	constrs_2	MapLogicOptParHighExtra (ISE 14)	Not started
impl_5	xc7k70tfg676-2	constrs_2	MapGlobalOptLogicOptRetimingDupP...	Not started
synth_2	xc7k70tfg676-2	constrs_2	TimingWithIOBPacking (XST 14)	Not started
impl_6	xc7k70tfg676-2	constrs_2	ISE Defaults (ISE 14)	Not started

図 9-4 : [Design Runs] ビュー

[Design Runs] ビューでは、run が開始されなかった場合も含めて、run のステータスが進行中か、終了したか、最新の状態でないことがレポートされます。ソース ファイル、制約、またはプロジェクト設定を変更すると、run は最新の状態ではなくなります。[Design Runs] ビューでは、run をリセットしたり、古い run のデータを削除したりできます。

アクティブ run の設定

PlanAhead でアクティブにできる合成 run とインプリメンテーション run はそれぞれ 1 つだけです。[Compilation] ビューおよび [Messages] ビュー、ステータス バー、[Project Summary] にはこのアクティブな run の情報が表示されます。[Project Summary] には、コンパイル リソースおよびこのアクティブな run のサマリ情報が表示されます。

run をアクティブにするには、[Design Runs] ビューでその run を右クリックし、ポップアップメニューから [Make Active] コマンドをクリックします。

インプリメンテーション オプションの設定

インプリメンテーション run のさまざまな設定は開始前に変更できます。

[Design Runs] ビューで run を選択すると、[Run Properties] ビューでその run の現在の設定が表示されます。[Run Properties] ビューでは、run の名前、run でターゲットにするザイリンクス パーツ、run の詳細記述、インプリメンテーションで使用される制約ファイルなどを変更できます。詳細は、第 4 章の「[Run Properties] ビューの使用」を参照してください。

インプリメンテーション ツールで使用するランタイム オプションも [Design Runs] ビューのポップアップ メニューから [Change Run Settings] をクリックして表示される [Design Run Settings] ダイアログ ボックスから変更できます (330 ページの図 9-5)。

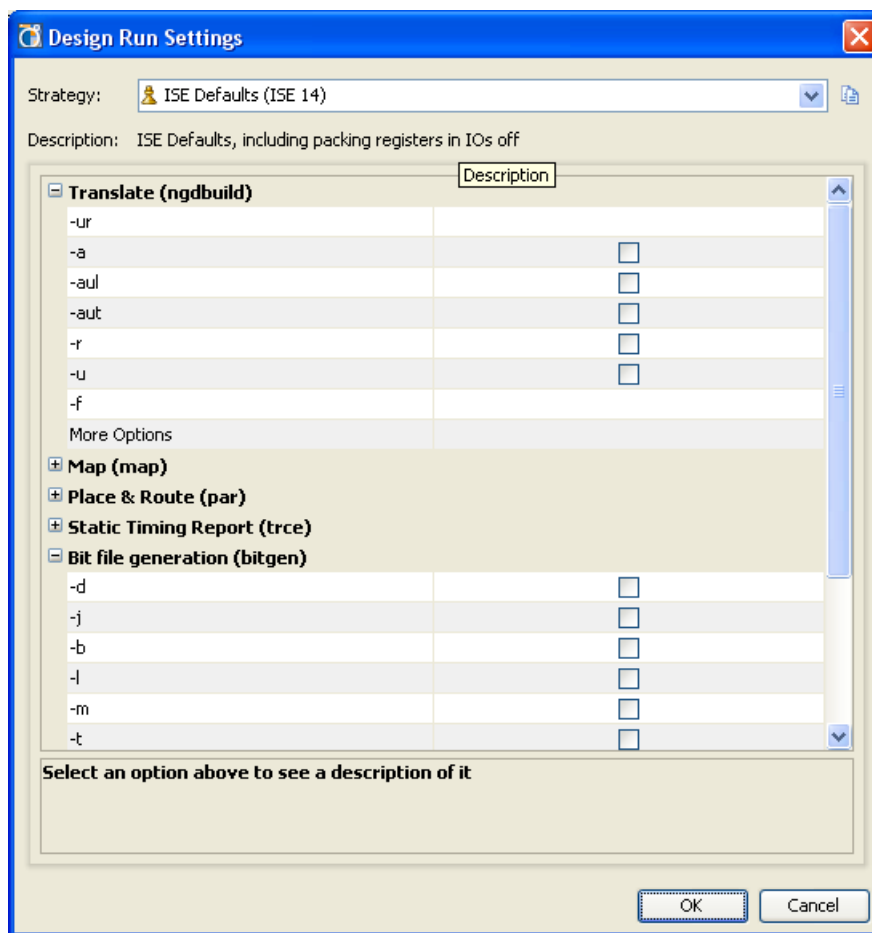


図 9-5 : [Design Run Settings] ダイアログ ボックス

[Design Run Settings] ダイアログ ボックスでは、次を指定できます。

- [Strategy] : run で使用される一般的なストラテジを指定できます。
- [ISE command-line options] : ISE インプリメンテーション ツール (NGDBuild、MAP、PAR、TRCE) のコマンド ライン オプションを設定できます。[More Options] フィールドを使用すると、リストされていないオプションも指定できます。

オプションを選択すると、簡単な説明と目的がページの下部に表示されます。オプション名の横にアスタリスク (*) が付いている場合は、その値がデフォルト以外の値に設定されていることを示します。

特定のオプションについては、付録 E 「その他のリソース」 に示す『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。

- [Save Strategy As] : ストラテジへの変更を別のプロジェクトで使用できるように新しいストラテジとして保存します。

インプリメンテーション run の起動

インプリメンテーションは、次のいずれかの方法で開始できます。

- Flow Navigator の [Run Implementation] をクリックします。
- メイン メニューから [Flow] → [Run Implementation] をクリックします。
- ツールバー メニューで [Run Implementation] をクリックします。

 Run Implementation


注記：アクティブ run が [Design Runs] ビューで開始されます。詳細は、[329 ページの「アクティブ run の設定」](#)を参照してください。

アクティブ run 以外の run を実行したり、複数のインプリメンテーション run を同時に実行したりもできます。[Design Runs] ビューで 1 つまたは複数の run を選択します。複数選択する場合は、Shift キーまたは Ctrl キーを押しながらクリックします。右クリックまたは [Design Runs] ビューのツールバー メニューから [Launch Runs] コマンドをクリックして [Launch Selected Runs] ダイアログ ボックスを表示します (図 9-6)。

注記：[Design Runs] ビューで複数 run を選択している場合は、合成 run とインプリメンテーション run の両方を選択できます。PlanAhead では、run の依存度が管理され、run が正しい順序で実行されます。

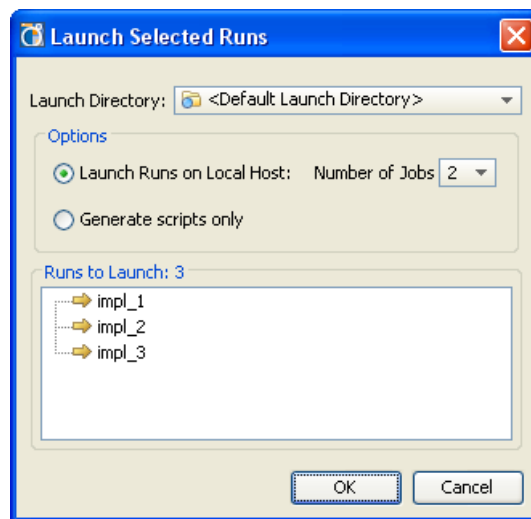


図 9-6 : [Launch Selected Runs] ダイアログ ボックス

- [Launch Directory]：デフォルトの起動ディレクトリは、ローカルのプロジェクト ディレクトリ構造に含まれます。インプリメンテーション run のファイルは、次のディレクトリに保存されます。

`<project_name>/<project_name>.runs/<run_name>`

注記：プロジェクト ファイルには絶対パスが記述されるので、プロジェクト ディレクトリ外のデフォルト以外の場所を指定すると、プロジェクトを移動しにくくなります。

- オプション
 - [Launch Runs on Local Host]：ローカル マシンのプロセッサで run を起動します。
 - [Number of Jobs]：run に使用するローカル プロセッサの数を指定します。このオプションは、複数の run を同時に起動する場合にのみ使用されます。各 run が各プロ

セッサで起動されます。このオプションでは、マルチスレッド プロセッサは使用されません。

- [Launch Runs on Remote Hosts] (Linux のみ) : ジョブを起動するのにリモート ホストを使用します。詳細は、第 9 章の「リモート Linux ホストでの run の起動」を参照してください。
 - [Configure Hosts] : リモート ホストを設定します。
- [Generate scripts only] : run ディレクトリおよび run スクリプトをエクスポートおよび作成しますが、run は起動しません。スクリプトは、PlanAhead 環境外で後で実行できます。

PlanAhead では、run のステータスによって、run を処理してインプリメンテーションを開始します。ステータスは、[Design Runs] ビューに表示されます (329 ページの図 9-4)。

- run のステータスが「Not Started」の場合、run はすぐに開始されます。
- run のステータスが「Error」になっている場合は、まず run がリセットされ、終了していない run データが削除されてから、run が再開されます。
- run のステータスが「Complete」または「Out-of-Date」になっている場合は、run をリセットするかどうか確認するメッセージが表示されます。

インクリメンタル インプリメンテーションの実行

インプリメンテーション run には、NGDBuild、MAP、PAR、TRCE などの ISE Design Suite に基づいた多くのプロセスが含まれます。PlanAhead には、インプリメンテーションを 1 つのプロセスではなく、インクリメンタルに少しずつ実行していく機能があります。

[Design Runs] ビューで run を選択し、ポップアップ メニューから [Launch Next Step: <Step>] をクリックします。有効な <Step> 値は、次のとおりです。

- [NGDBuild] : EDIF または NGC フォーマットでネットリスト ファイルを読み込んで、Xilinx® Native Generic Database (NGD) ファイルとしてデザインの論理的記述を作成します。
- [MAP] : 論理デザインをザイリンクス® FPGA にマップします。
- [PAR] : ターゲット ザイリンクス デバイスにデザインを配置配線します。
- [TRCE] : 入力したタイミング制約に基づいて、FPGA デザインのスタティック タイミング解析を実行します。
- [BitGen] : ザイリンクス デバイス コンフィギュレーションのビットストリームを生成します。BitGen は、技術的にはインプリメンテーションではありませんが、インクリメンタルなステップとして使用可能になっています。

終了したステップからバックアップするには、[Design Runs] ビューから [Reset to Previous Step: <Step>] コマンドを使用します。このコマンドは、現在のステートから前のインクリメンタル ステップに選択した run をリセットします。これにより、run を前の段階に戻して、必要に応じて変更を加えてから、run を次の段階へ進めて終了させることができます。

注記：どの段階でも [Launch Runs] コマンドを使用して、残りすべてのステップを含めたインプリメンテーション run を実行するか、[Reset Runs] コマンドを使用して run を最初の状態にリセットすることができます。

プロセスのバックグラウンドへの移動

PlanAhead でデザイン ファイルおよび制約ファイルを読み込んで合成またはインプリメンテーションが実行されると、[Starting Run] ダイアログ ボックスが開き (図 9-7)、プロセスがバックグラウンドで実行されます。

プロセスがバックグラウンドで実行されると、バックグラウンド タスクを実行させたまま、レポートを表示したり、デザイン ファイルを開いたりといった別の機能が実行できます。この間、前の run を確認したり、レポートを表示したりして、時間を効率的に利用できます。ただし、[Tcl Console] はブロックされるので、Tcl コマンドを使用したり、開いている別のデザインに切り替えるような Tcl コマンドを必要とするタスクは実行できません。

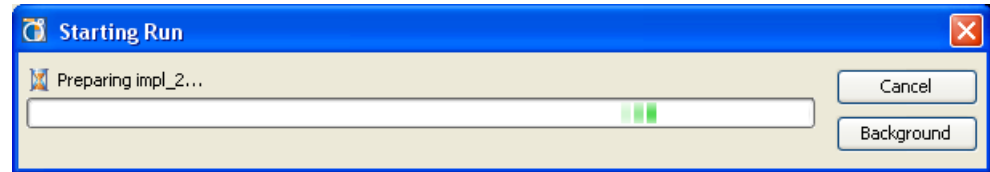


図 9-7 : [Start Run] ダイアログ ボックス

インプリメンテーション run の開始

合成またはインプリメンテーション run のステータスは、[Compilation] ビューを参照するか、[Messages] ビューで情報、警告、エラー メッセージを参照するか、[Project Summary] ビューを参照するか、[Design Runs] ビューを開くと確認できます。

次のセクションでは、run ステータスの監視オプションについて説明します。

run ステータス表示の使用

進行中の run のステータスは、2 つの方法で表示されます。図 9-8 のように、ステータス表示には run が進行中であることが示されるほか、ここから必要に応じてキャンセルできるようになっています。

1. PlanAhead の右上端の run ステータスのインジケータには、run が進行中であることを示すスクロールバーと run を終了する [Cancel] ボタンが表示されます。
2. 図 9-8 の下部に示されるように、[Design Runs] ビューの run ステータスのインジケータには、run が進行中であることを示す円状の矢印が表示されます。run を選択してポップアップメニューから [Reset Run] をクリックすると、run をキャンセルできます。

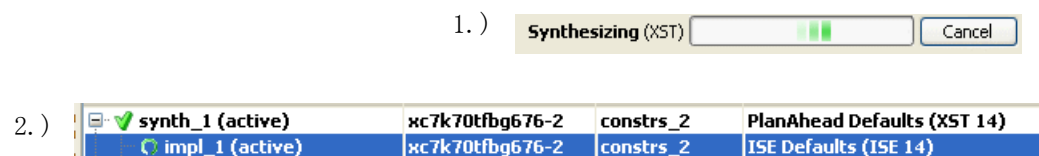


図 9-8 : run のステータス

run のキャンセル

進行中の run を [Cancel] ボタンか [Reset Run] コマンドでキャンセルすると、キャンセルした run 用に作成された run ファイルをすべて削除するかどうか確認するメッセージが表示されます。

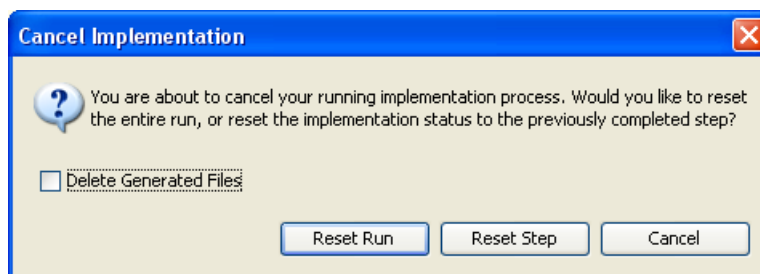


図 9-9：合成のキャンセル

[Delete Generated Files] をクリックすると、ローカルプロジェクトディレクトリから run データが削除されます。キャンセルした run で作成されたデータはすべて削除して、今後の run で問題にならないようにしておくことをお勧めします。

コンパイル ログの表示

run を起動すると、[Compilation] ビューが開き、標準出力メッセージが表示されます。334 ページの図 9-10 に、[Compilation] ビューの例を示します。

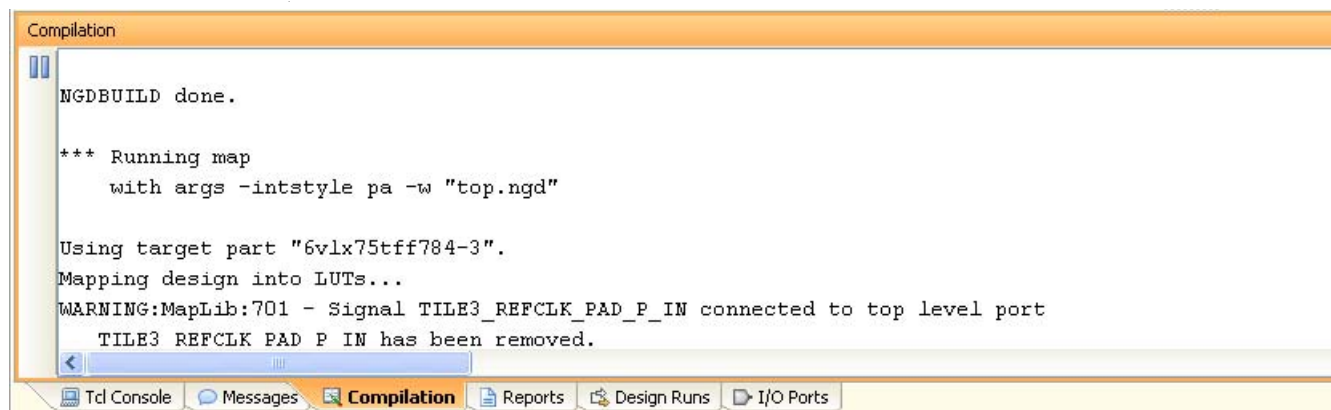


図 9-10：[Compilation] ビュー

[Pause] ボタンをクリックすると、[Compilation] ビューへの出力を一時停止でき、コマンドの実行中にログをスクロールして読むことができます。

プロジェクト ステータスの確認

PlanAhead では、プロジェクトの全体的なステータスおよびプロセスの次の手順を実行する方法が示されます。プロジェクト ステータスには、デザイン プロセスの主なタスクの結果のみが示されます。

プロジェクトの全体的なステータスは [Project Summary] ビューとステータス バーに表示され、プロジェクトを開いたとき、デザイン フロー コマンドを実行したときにプロジェクトのステータスをすばやく判断できます。RTL エラボレーション、合成、インプリメンテーション、ビットストリーム生成などのステータスが示されます。

プロジェクト ステータス バー

プロジェクトの全体的なステータスは、図 9-11 に示すように、メイン ウィンドウの右上に表示されます。



図 9-11 : プロジェクト ステータス バー

エラボーレーション、合成、インプリメント、ビットストリームの生成を実行すると、プロジェクト ステータス バーにその結果が示されます。プロセスでエラーが発生した場合は、赤色の文字で表示されます。

ソース ファイルをアップデートすると、合成およびインプリメンテーションが前に終了している場合は、「Out-of-Date」と表示され (335 ページの図 9-12)、ステータス バーにそのステータスが示されます。デザインのどの部分が最新ではないのかを確認するには、[more info] リンクをクリックします。詳細は、39 ページの「デザインのアップデートおよび再読み込み」を参照してください。

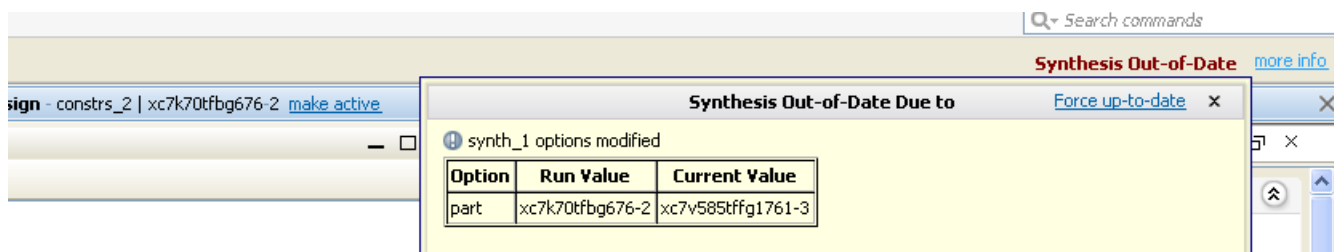


図 9-12 : デザインの再読み込みが必要であることを示すバナー

デザインのアップデートが必要であることを示すバナー

ソース ファイル、ネットリスト、制約またはインプリメント結果がアップデートされると、開いているデザインの上部にバナーが表示され、デザイン データがアップデートされたことが示されます。図 9-13 に示すような [Reload] リンクをクリックすると、アップデートされたデザインをメモリに読み込むことができます。

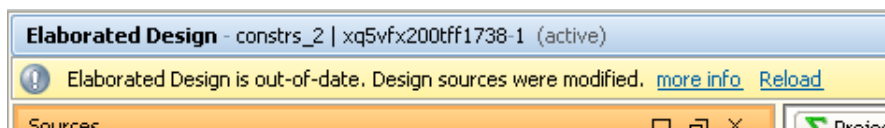


図 9-13 : デザインの再読み込みが必要であることを示すバナーと [Reload] リンク

インプリメンテーション run 結果の確認

合成またはインプリメンテーションが終了したら、ISE レポートを表示し、ネットリストまたはインプリメント済みデザインを開いて、必要に応じてタイミングおよび物理制約を適用し、デザインを解析し、run を再インプリメントできます。詳細は、第 11 章「インプリメンテーション結果の解析」を参照してください。

レポート ファイルの表示

ISE ツールで生成されたレポート ファイルは [Reports] ビューから表示できます。このビューは、通常コマンドが実行されると自動的に開きます。開かない場合は、[Project Summary] ビューの [Reports] リンクをクリックします。[Reports] ビューが図 9-14 のように表示されます。

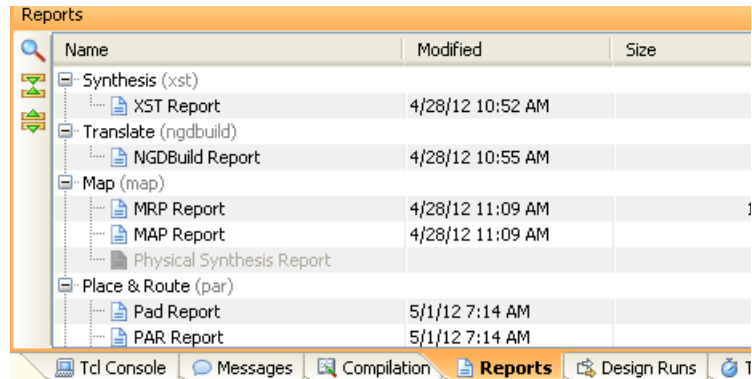


図 9-14：表示するレポート ファイルの選択

ワークスペースで使用可能なレポート ファイルをクリックすると、そのレポートが 337 ページの図 9-15 に示すように表示されます。

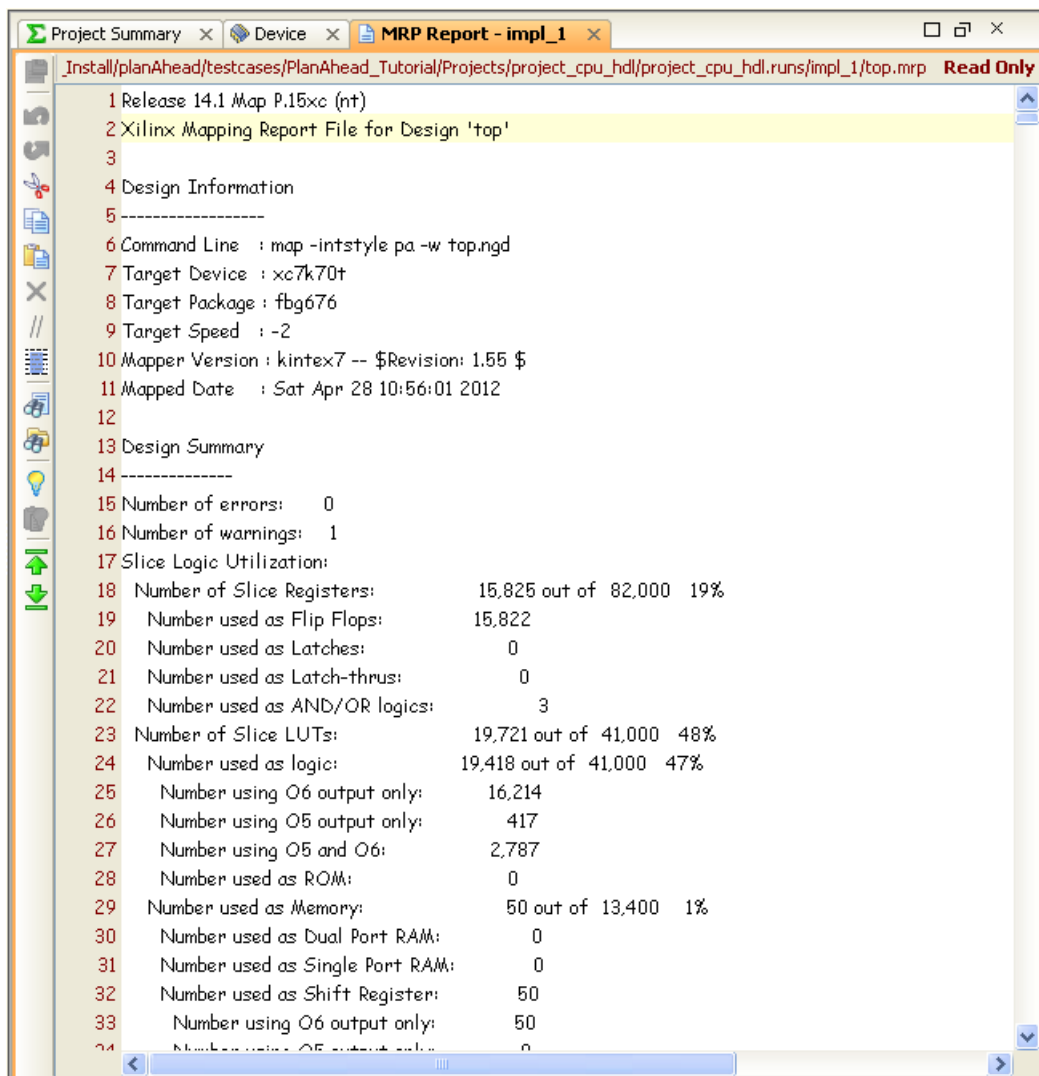


図 9-15 : レポート ファイルの表示

レポートを表示すると、次が実行できます。

- スクロール バーを使用してレポート ファイルを参照
- [Find] または [Find in Files] ボタンをクリックし、特定テキストを検索
- [Go to the beginning] または [Go to the End] ボタンをクリックしてファイルの冒頭または最後に移動

メッセージの表示

[Messages] ビューは [Compilation] ビューがフィルターされたリストで、主な警告およびエラーメッセージのみが含まれます。ビューのツールバー ボタンを使用し、エラーまたは警告メッセージのみを表示させることもできます。338 ページの図 9-16 は、[Messages] ビューの例を示しています。インプリメンテーション メッセージは、ISE コマンドおよび重要性別に表示されます。



図 9-16 : [Messages] ビュー

横にあるプラス記号 (+) をクリックして展開し、各メッセージを表示します。

338 ページの図 9-17 の [Messages] ビューのバナーで、エラー、クリティカルな警告、警告、情報メッセージのチェック ボックスをオンにすると、該当するメッセージが表示されます。

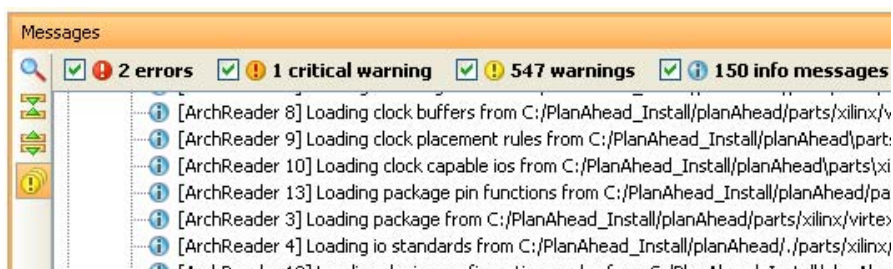


図 9-17 : [Messages] ビューのバナー

[Messages] ビューで行番号を含むメッセージを選択すると、RTL ファイルが開き、ソース コードでその行がハイライトされます。メッセージを右クリックして [Search for Answer Record] をクリックすると、ザイリンクス ウェブサイトで関連するアンサー データベースが検索されます。

インプリメンテーション後のフロー

run が完了すると図 9-18 のようなダイアログ ボックスが表示され、次のステップを選択できます。

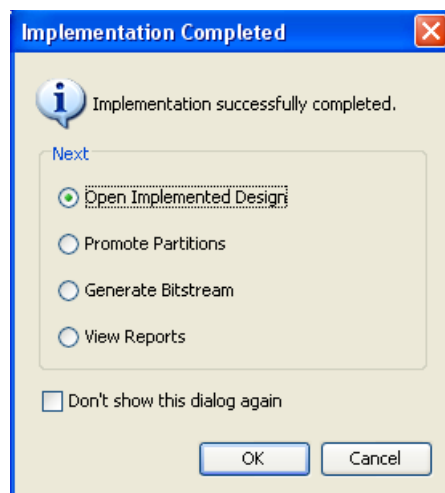


図 9-18 : [Implementation Completed] ダイアログ ボックス

[Implementation Completed] ダイアログ ボックスで次のオプションを選択し、[OK] をクリックします。

- [Open Implemented Design] : ネットリスト、アクティブな制約セット、ISE 配置情報、タイミング結果およびターゲット パーツを PlanAhead のデザイン解析およびフロアプラン環境にインポートします。詳細は、第 11 章「インプリメンテーション結果の解析」を参照してください。
- [Promote Partitions] : デザインでパーティションが定義されているときのみ表示されます。今後のデザイン繰り返しで使用するパーティションをデザインから選択してプロモートできます。詳細は、第 13 章「階層デザイン手法の使用」を参照してください。
- [Generate Bitstream] : [Generate Bitstream] ダイアログ ボックスを開きます。詳細は、第 12 章「ビットストリーム ファイルの生成」を参照してください。
- [View Reports] : ISE レポート ファイルを選択して表示できる [Reports] ビューを開きます。詳細は、第 9 章「レポート ファイルの表示」を参照してください。

リモート Linux ホストでの run の起動

PlanAhead には、複数の Linux ホストで run をパラレルに実行できる機能があります。これは、Oracle の Grid Engine や LSF® などの簡易版を使用した機能です。

ジョブ提示アルゴリズムは、セキュア シェル (SSH) 内の Tcl パイプを使用したラウンドロビン形式でインプリメントされています。

リモート Linux ホストで run を起動する際の制限

リモート Linux ホストで run を起動する際の制限は、次のとおりです。

- ホスト実行には、PlanAhead ではなく、Linux OS で提供されているサービスである SSH が使用されます。これには、リモート マシンにログインするたびにパスワードを入力する必要がないように SSH を設定する必要があります。パスワードを入力せずに SSH が使用できるように

設定していない場合、または SSH を設定したのに毎回パスワードの入力を求められる場合は、[付録 D 「パスワード入力なしの SSH の設定」](#) を参照してください。

- セキュリティのためと Windows システムにリモート シェル機能がないため、Linux 間のホストのみがサポートされます。
- ISE ツールのインストールは、どのログイン シェルからでも使用できると仮定されるので、`$XILINX` および `$PATH` 環境変数は `.cshrc/.bashrc` セットアップ スクリプトで正しく設定されます。リモート コンピュータにログインしほかのスクリプトをソースせずに「`map -help`」と入力できる場合、このフローは機能します。ログイン (`.cshrc` または `.bashrc`) で ISE 設定がされていない場合、[Run pre-launch script] を使用して環境設定スクリプトをすべてのジョブ前に実行することができます。
- PlanAhead のインストールはリモート マシンの割り当てられたネットワークから表示できるようにする必要があります。PlanAhead のインストール ディレクトリがマシンのローカルディスクの場合は、リモート マシンからは表示できません。
- PlanAhead のプロジェクト ファイル (`.ppr`) およびディレクトリ (`.dita` および `.runs`) は、リモート マシンの割り当てられたネットワークから表示できるようにする必要があります。デザイン データがローカル ディスクに保存されていると、リモート マシンからは表示できません。

リモート ホストの設定 (Linux のみ)

[付録 D の「SSH の設定」](#)にあるように SSH を設定したら、リモート サーバーを使用して PlanAhead で run を起動することができます。これには、まずリモート ホストを設定する必要があります。

1. 次のいずれかのコマンドを使用し、リモート ホストを設定します。
 - [Tools] → [Options] → [Remote Hosts]
 - [Run Synthesis] → [Launch Runs] → [Configure Hosts]
 - [Implementation] → [Launch Runs] → [Configure Hosts]
 - [Launch Selected Runs] ダイアログ ボックスの [Configure Hosts]

341 ページの [図 9-19](#) のように、[Remote Hosts] ページが表示されます。
2. [Add] ボタンをクリックし、リモート サーバー名を入力します。
3. [Jobs] で使用するリモート マシンのプロセッサ数を指定します。各 run が各プロセッサで起動されます。マルチスレッド プロセッサは使用されません。
4. [Enabled] チェック ボックスでは、サーバーを使用するかどうかを指定します。選択した run を起動する際にどのサーバーを使用するか指定します。
5. オプションで [Launch jobs with] フィールドでリモート アクセス コマンドを変更することもできます。デフォルトは、ssh です。

注記：このフィールドを変更する場合は、細心の注意を払ってください。たとえば、BatchMode =yes を削除すると、シェルで間違っパスワードのプロンプトが表示されるため、プロセスが停止してしまいます。
6. [Run pre-launch script] をオンにすると、run の起動前に実行するスクリプトを定義できます (オプション)。ログイン時の ISE セットアップ (`.cshrc` または `.bashrc`) がない場合、このチェック ボックスをオンにして環境設定スクリプトを飛ばすことができます。
7. [Run post-completion script] をオンにすると、run の終了後に実行するカスタム スクリプトを定義できます (オプション)。
8. [Send email to] をオンにし、メール アドレスを入力しておくと、run の終了時に電子メールが送信されます (オプション)。

9. 1 つまたは複数のホストを選択し、[Test] をクリックすると、そのサーバーが使用可能かどうか、コンフィギュレーションが問題なく設定されたかどうかを確認できます。

注記：各ホストをテストして、設定に問題がないか確認しておくことをお勧めします。

10. 選択したリモート ホストを削除するには、[Remove] を、リモート ホストのコンフィギュレーション設定を使用する場合は [OK] をクリックします。

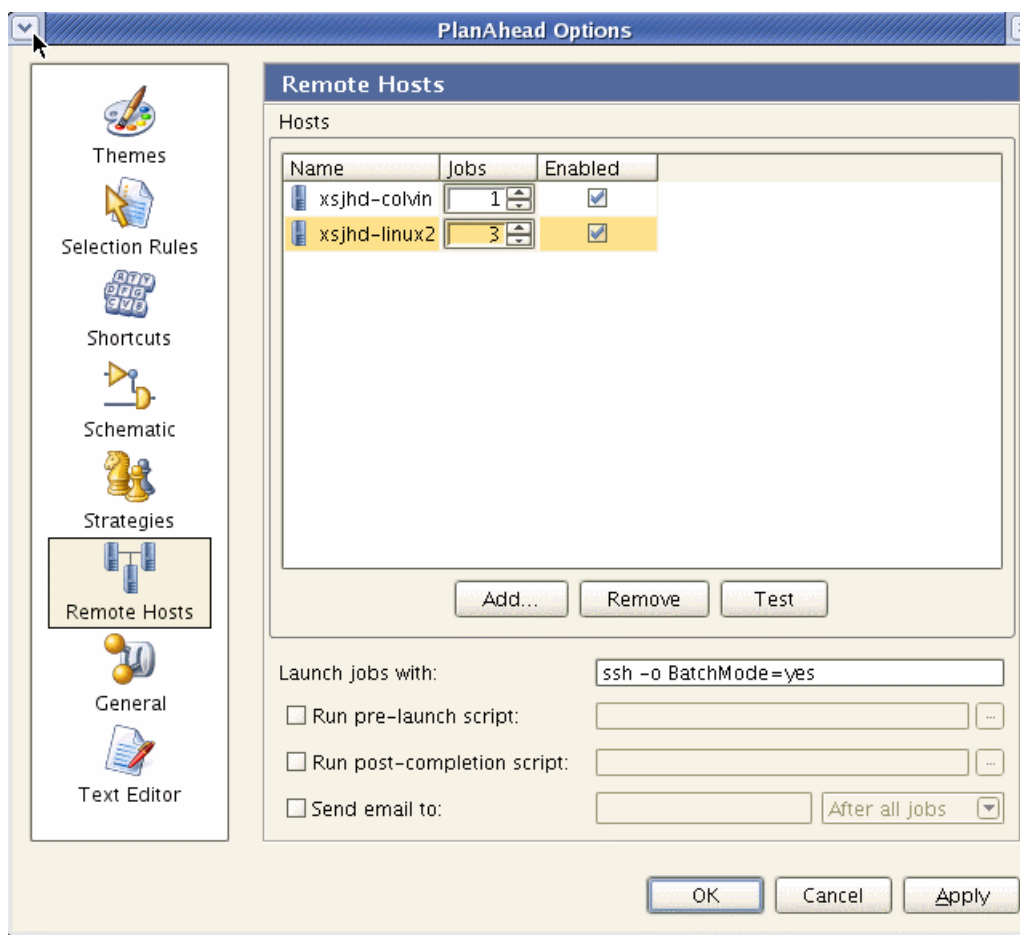


図 9-19 : リモート ホストの設定

デザインのフロアプラン

PlanAhead™ ではフロアプランがサポートされており、クリティカルなロジックに制約を付けてインターコネクトを短くし、遅延を抑えることができます。

フロアプランは、物理デザインでインタラクティブに作業する必要があります。ボタンを押すだけのフローとは異なります。設計者はデザインについての知識と PlanAhead の解析機能を使用して、パフォーマンスを向上させる制約およびツール オプションを定義します。

フロアプランでは、次を実行します。

- 物理ブロック (Pblock) ロケーションを作成してロジック配置を制約
または
- 個々のロジック オブジェクトを特定デバイス サイトにロック

[Floorplanning] ビューには、フロアプラン中に使用される共通のビューが含まれます。

[Floorplanning] ビューを開くには、次のいずれかを実行します。

- [Layout] → [Floorplanning] をクリック
または
- ツールバーのレイアウト セレクター (ドロップダウン リスト) から [Floorplanning] を選択

フロアプランは複雑なため、本章の説明だけでは十分ではありません。詳細は、[付録 E 「その他のリソース」](#) に示す『フロアプラン手法ガイド』(UG633) を参照してください。

Pblock の概要

フロアプランのプロセスでは、まずデザインの一部またはすべてのロジックをグループに分割して制約を付けます。デザインは、管理しやすい小さな物理ブロック (Pblock) に、階層的に分割されます。このため、論理階層のどこからでも論理モジュールおよびプリミティブ ロジックを Pblock に含めることができます。クリティカル ロジックまたは関連するロジックは、ロジックが移行しないように 1 つの Pblock にグループ化されるため、インターコネクトの長さが制限され、遅延が低減されます。

ここでは、次のセクションに分けて Pblock について説明します。

Pblock の作成

Pblock を作成すると、エクスポートされた UCF ファイルに AREA_GROUP 制約が書き込まれます。PlanAhead で割り当てられたロジック、指定範囲、定義した属性はこの制約ファイルに反映されます。

[Draw Pblock] コマンドの使用

[Draw Pblock] コマンドを使用すると、選択したロジックが [Device] ビューで新しい Pblock に割り当てられます。コマンドを実行する前に、Pblock に割り当てるロジックを選択しておきます。

Pblock を作成するには、次の手順に従います。

1. 任意のビューで Pblock に割り当てるロジックを選択します。
2. [Device] ビューのポップアップ メニューまたはツールバーから [Draw Pblock] をクリックします。
3. [Device] ビューでカーソルを Pblock の描画を開始する場所に置きます。
4. マウスの左ボタンを押したまま対角線上の角になる場所に移動して Pblock の長方形を描画し、ボタンを離します。



図 10-1 のように、[New Pblock] ダイアログ ボックスが開きます。

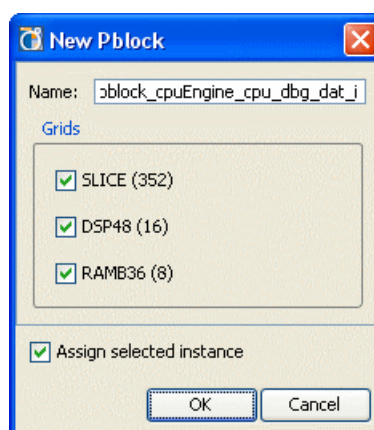


図 10-1 : [New Pblock] ダイアログ ボックス

5. オプションを変更し、[OK] をクリックします。オプションは、次のとおりです。
 - [Name] : Pblock の名前を指定します。フィールドには、Pblock のデフォルト名の pblock_n または Pblock_instancename が表示されます。ほかの名前を使用する場合は変更します。
 - [Grids] : 指定したエリア内で異なるデバイス タイプを選択し、Pblock で制約されるようにします。
 - [Assign selected instances] : オンにすると、選択されたインスタンスが新しい Pblock に割り当てられます。

注記：ロジックが間違って選択されていることもあるので、確認してエラーを回避してください。

Pblock が表示されると、[Device] および [Physical Hierarchy] ビューで選択できるようになります。[Physical Constraints] ビューが [Floorplanning] ビュー レイアウトの中に開きます。手動で開くには、[Window] → [Physical Constraints] をクリックします。

手動で作成する際の最初の Pblock のサイズおよび位置は、重要ではありません。Pblock のサイズは、[Pblock Properties] ビューの [Statistics] タブの [Physical Resource Estimates] から確認できます。Pblock の位置は、[Pblock Properties] ビューの [Rectangles] タブから確認できます。[Device] ビューで接続状況を表示しながら、Pblock 同士の相対的な位置を指定することもできます。

サイズを決める前に、[345 ページの図 10-2](#) のようにすべての Pblock を小さな長方形として作成し、Pblock 間の接続フローを視覚的に確認した方が便利な場合もあります。

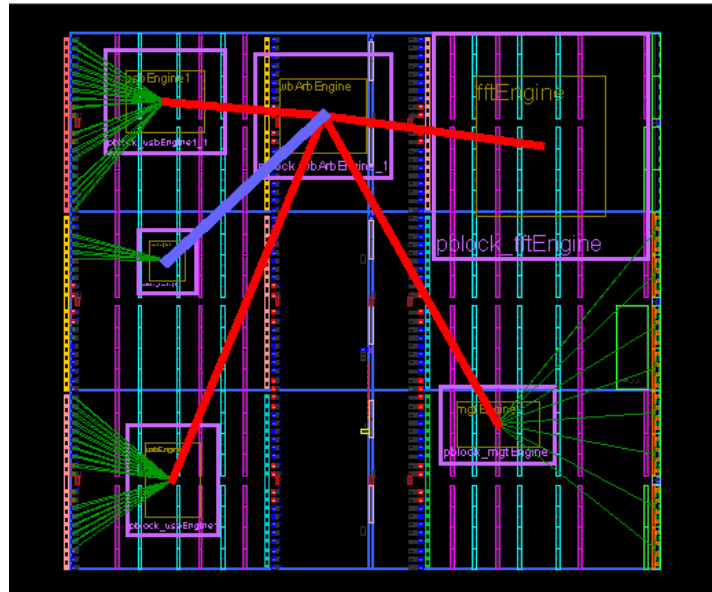


図 10-2 : [Device] ビューに表示された Pblock の接続解析

このビューの詳細は、[244 ページの「階層接続の解析」](#) および [388 ページの「\[Show Connectivity\] コマンドの使用」](#) を参照してください。

[New Pblock] コマンドの使用

[New Pblock] コマンドを実行すると、[Physical Hierarchy] ビューには新しい Pblock が作成されますが、[Device] ビューに長方形は作成されません。

コマンドを実行する前に、新しい Pblock に割り当てるロジックを選択しておく必要があります。ロジックを選択しない場合は、空の Pblock が作成されます。

Pblock を作成するには、ポップアップ メニューから [New Pblock] をクリックします。ロジックは選択していてもいなくてもかまいません。

[Create Pblocks] コマンドを使用した複数の Pblock の作成

Create Pblocks ウィザードを使用すると、複数の Pblock を半自動的に作成できます。このウィザードでは、選択されたネットリスト インスタンスごとに未配置の Pblock が作成されます。

ウィザードを使用する前に、各 Pblock に含めるインスタンスのセットを選択しておきます。

選択したネットリスト インスタンスに複数の Pblock を作成するには、次の手順に従います。

1. Pblock に含めるインスタンスを選択します。
2. [Tools] → [Floorplanning] → [Create Pblocks] をクリックします。

Create Pblocks ウィザードが開き、[346 ページの図 10-3](#) のように選択されたインスタンスのリストが表示されます。

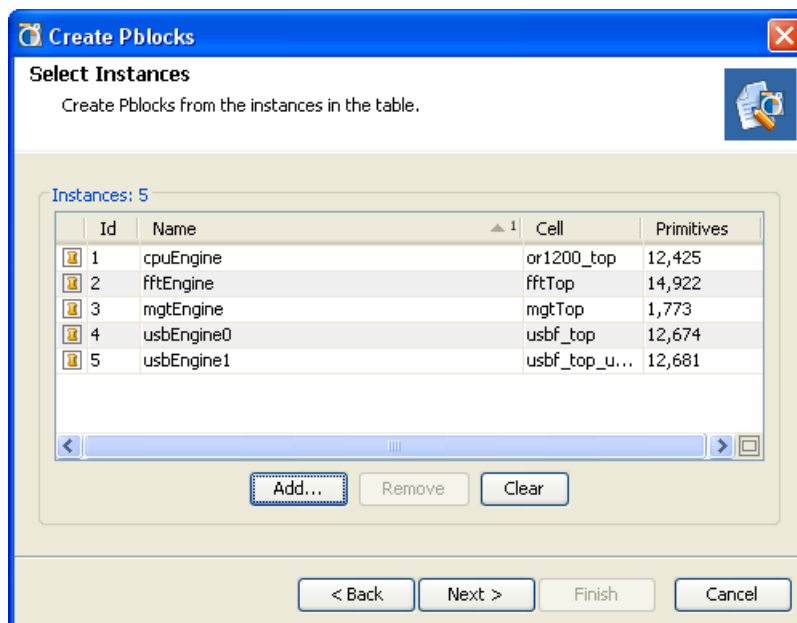


図 10-3 : Create Pblocks ウィザード : インスタンスから Pblock を作成

- このリストにネットリスト インスタンスを追加するには、[Add] ボタンをクリックしてブラウザを開き、インスタンスを選択します。
 - リストからインスタンスを選択して削除するには、削除するインスタンスを選択し、[Remove] をクリックします。
 - リストからネットリスト インスタンスを削除するには、[Clear] をクリックします。
- [Next] をクリックします。
Create Pblocks ウィザードでは、Pblock の名前の命名方法を指定できます。
 - [Format Names] ページで、Pblock の名前に関するフィールドを編集します。
 - [Prefix] : Pblock 名の接頭辞を指定します。新しい接頭辞を指定しても、デフォルトのインスタンス名や番号を使用してもかまいません。
 - [Suffix] : [Instance name] を指定すると、インスタンス名が接頭辞に続きます。[Numeric] を指定すると、1 から開始する番号が接頭辞に続きます。
 - [Next] をクリックします。
 - [Summary] ページの内容を確認します。
 - [Finish] をクリックすると、設定内容を使用して Pblock が作成されます。

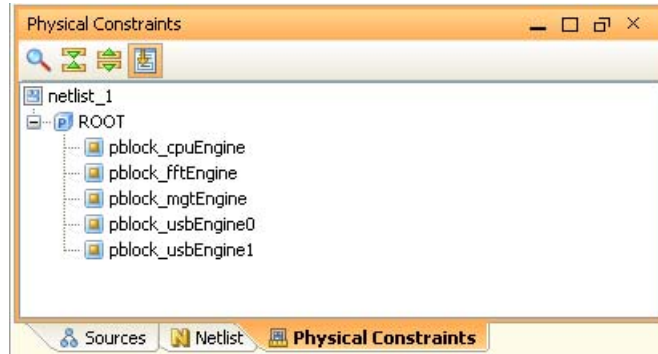


図 10-4 : [Physical Constraints] ビューの Pblock

Pblock は、347 ページの図 10-4 のように [Physical Constraints] ビューに表示されます。

新規作成された Pblock に長方形を作成するには、次の手順に従います。

1. [Physical Constraints] ビューで、新規の Pblock を 1 つずつ選択します。
2. [Device] ビューでポップアップ メニューから [Set Pblock Size] をクリックします。
3. [Device] ビューで長方形を描画します。



ネスト化された Pblock の作成

Pblock 内に Pblock (ネスト化された Pblock) を作成すると、ロジックの制約でより厳密な制御ができるようになります。これはクリティカルなモジュールのパフォーマンスの向上に役立ちます。使用率予測には、最上位 Pblock に下位 Pblock がすべて含まれます。

注記： ISE® のインプリメンテーション ソフトウェアでは、この機能は完全にはサポートされていません。ネスト化した Pblock を作成すると、マップおよび配線エラーが発生することがあります。

複数の長方形を含む Pblock の作成

複数の長方形を含む Pblock を作成することで、長方形以外の形の Pblock を作成したり、1 つの大きな Pblock を作成せずに遠くに配置されているデバイス リソースをカバーしたりできます。

既存の Pblock を選択し、ポップアップ メニューから [Add Pblock Rectangle] をクリックして既存 Pblock に長方形を追加します。

点線で結ばれている複数の長方形は、1 つの Pblock に含まれているものであることを示します。割り当てられたインスタンスの長方形および接続を表す線は、図 10-5 のように一番大きな長方形に表示されます。

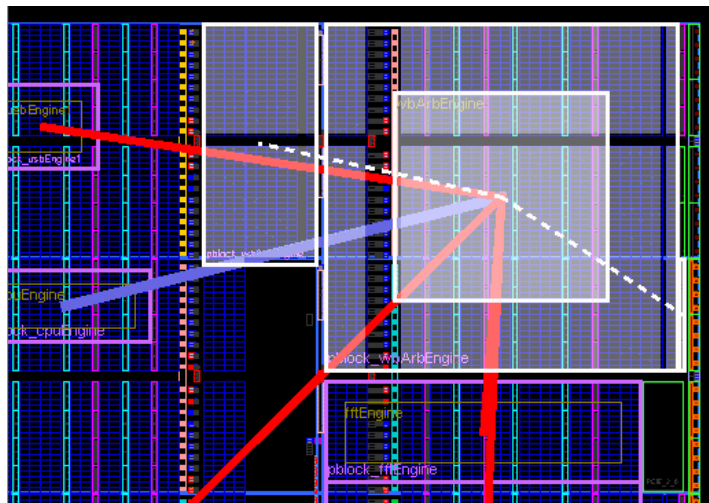


図 10-5：複数の長方形を含む Pblock の作成

クロック領域 Pblock の作成

Pblock は、Pblock バウンダリで囲まれた領域に対して指定したデバイス リソースを含めるように定義されます。また、Pblock は 1 つの特定クロック領域内、または複数のクロック領域内にあるすべてのリソースを含めるよう定義することもできます。

[Device] ビューで Pblock をクロック領域として定義するには、次の手順に従います。

1. クロック領域の境界を含める長方形で Pblock を描画します。

PlanAhead でクロック領域の境界線が表示されます。クロック領域の境界線の色や表示方法を変更する場合は、第 4 章の「表示テーマのカスタマイズ」を参照してください。

図 10-6 に、クロック領域 Pblock を示します。ツール ヒントには、Pblock 範囲にクロック領域が含まれることが示されます。

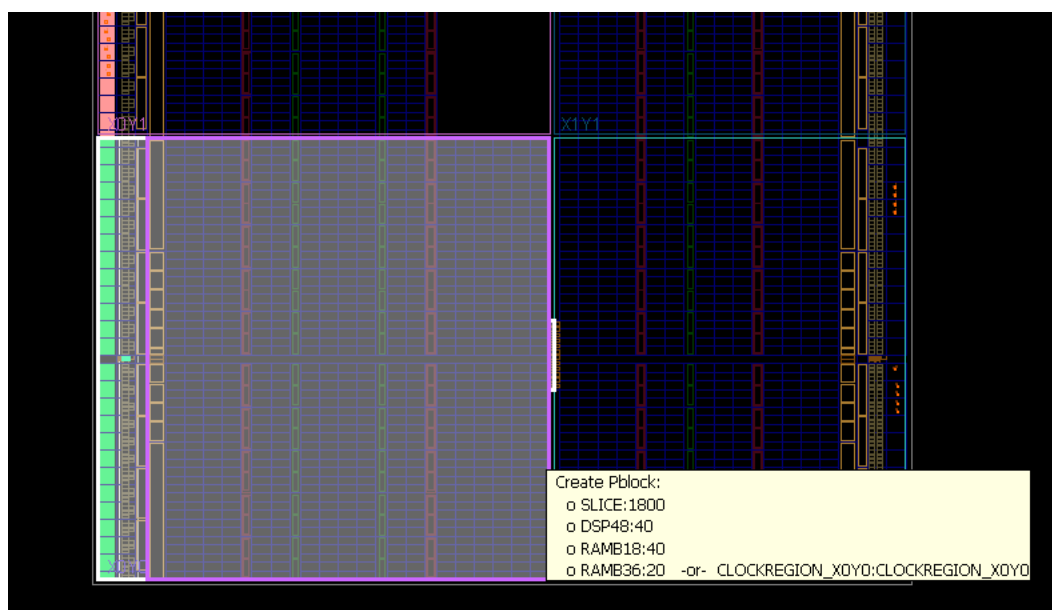


図 10-6：クロック領域 Pblock の作成

2. 図 10-7 の [Set Pblock] ダイアログ ボックスで [OK] をクリックし、Pblock の範囲をクロック領域 (CLOCKREGION_X) として定義します。

[Set Pblock] ダイアログ ボックスで [CLOCKREGION_X] オプションを有効にするには、Pblock の長方形にクロック領域の境界を含める必要があります。[CLOCKREGION_X] をオフにすると、従来のロジック ベースのデバイス リソースを使用して Pblock を定義できます。

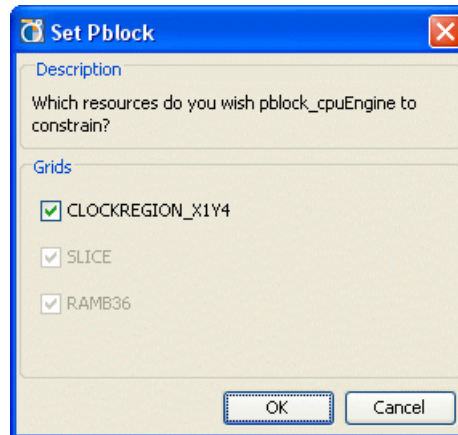


図 10-7 : [Set Pblock] ダイアログ ボックスで Pblock をクロック領域として定義

注記 : [Set Pblock] ダイアログ ボックスまたは [Pblock Properties] ビューにある [CLOCKREGION_X] チェックボックスのオン/オフを切り替えて、クロック領域とロジックベースの Pblock を切り替えることができます。

Pblock クロック領域の座標は [Pblock Properties] ビューに表示されます。

Pblock の図

デフォルトの表示オプションでは、Pblocks および割り当てられたインスタンスが図で表示されます。

- 外側の長方形は Pblock のボーダーで、Pblock に含まれる FPGA のエリアを示します。
- 内側の長方形は、その Pblock に割り当てられたデバイス リソースの割合を反映します。

複数のインスタンスを 1 つの Pblock に配置できます。Pblock の内側に表示されるインスタンスの長方形のサイズは、含まれるロジック数と、同じ Pblock 内のほかのインスタンスの対比で決まります。

Pblock に割り当てられているインスタンス数が多いと、長方形ではなく、図 10-8 のように直線のように見えることがあります。

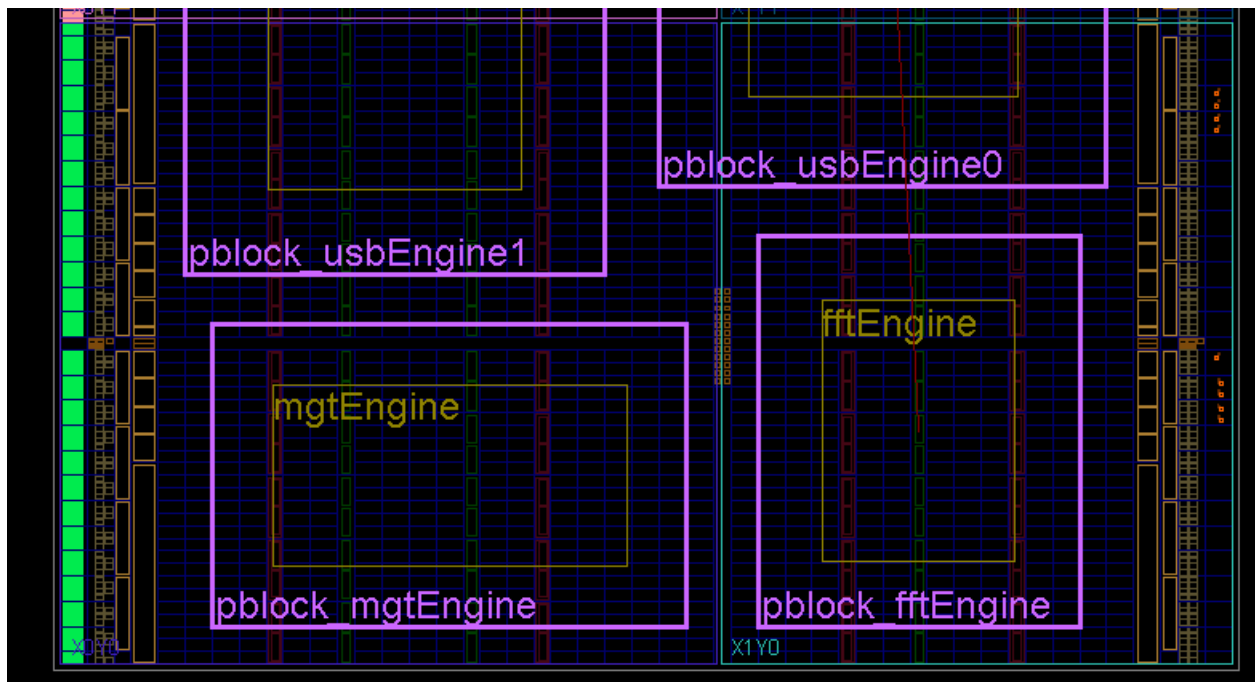


図 10-8：割り当てられたインスタンスを示す Pblock 図

デフォルトの選択規則を使用する場合、Pblock の長方形を選択すると、それに割り当てられたネットリスト インスタンスもすべて選択されます。1 つの Pblock からインスタンスをドラッグして、別の Pblock 内に割り当てることもできます。選択規則は、[Tools] → [Options] をクリックし、ダイアログ ボックスの左側のメニューから [Selection Rules] をクリックすると表示および変更できます。

注記： Pblock に対して操作を実行する場合は、Pblock に割り当てられたインスタンスではなく、正しい Pblock の長方形が選択されていることを確認してください。Pblock の操作時にインスタンスの選択機能をオフにすると、[Device] ビューで割り当てられたインスタンスではなく Pblock が選択されます。

351 ページの図 10-9 のように、I/O ネットは Pblock の中央ではなく、Pblock 内のインスタンスの中央に接続されます。

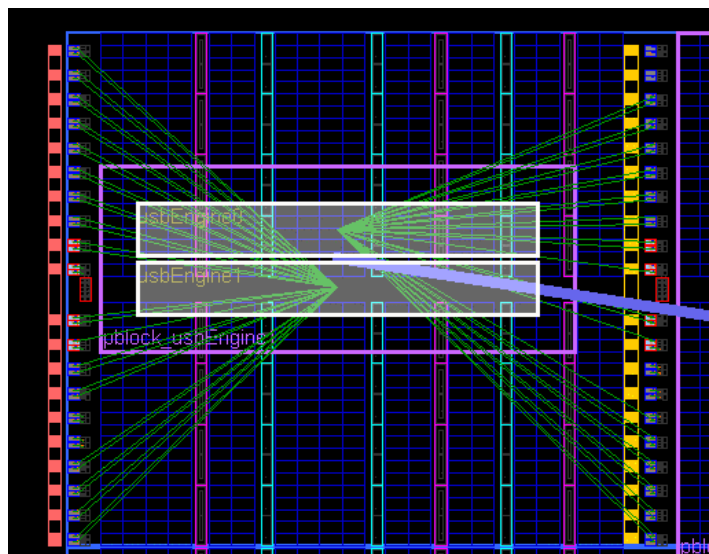


図 10-9：インスタンス長方形の中央に表示された I/O 接続

子 Pblock の色は区別できるように異なった色で表示されます。Pblock や子 Pblock などのオブジェクトの色は、次を使用して設定できます。

- [Tools] → [Options]
- [PlanAhead Options] ダイアログ ボックスの [Themes] ページ
- [Themes] ページの [Device] タブ

Pblock プロパティの表示

[Pblock Properties] ビューにはさまざまなタイプの情報が表示されます。Pblock プロパティを表示または編集するには、Pblock を選択して [Pblock Properties] ビューを表示します。352 ページの表 10-1 は、[Pblock Properties] ビューのタブとオプションをリストしています。

表 10-1 : [Pblock Properties] ビューのタブとオプション

タブ	オプション
[General]	<ul style="list-style-type: none"> • [Name] : Pblock 名を表示します。 • [Parent] : 親 Pblock が表示されます。Pblock によってはこのフィールドは編集できないことがあります。親 Pblock が複数ある場合はこのフィールドで指定します。 • [Grid Range] : Pblock に AREA_GROUP の RANGE プロパティを指定できます。特定の範囲を選択すると、選択されたロジック タイプのみが Pblock エリアに含まれます。Pblock が作成されると、範囲座標が各ロジック タイプに対し表示されます。 • [CLOCKREGION] : Pblock の範囲をクロック領域全体に定義する場合にオンにします。Pblock の長方形がクロック領域境界に一致するよう描画されます。 • [Apply]/[Cancel] : 変更内容を適用するか破棄するか決定します。
[Statistics]	<p>Pblock の統計は、最終的なインプリメント済みデザインとは異なる可能性があります。Pblock のプロパティは、概算専用です。たとえば、理想的なパッキング状況下では、あるデザインをデバイスの使用可能な SLICEM 32700 個のうちの 995 個にパックできると Pblock の統計に示されているとします。このため、SLICEM が 995 未満の場合、デバイスはデザインにフィットしません。ただし、インプリメンテーション中にロジックは使用可能なデバイス リソースに分散される可能性があり、最低限必要な 995 個の SLICEM の数がインプリメント済みデザインではたとえば 1200 個に増えることがあります。この場合、使用された SLICEM の一部では LUT が 2 つ使用され、ほかの SLICEM では 3 または 4 つ使用されることがあります。</p> <ul style="list-style-type: none"> • [Physical Resources Estimates] : デバイスのリソース タイプごとの表が表示されます。 <ul style="list-style-type: none"> • [Site Type] : Pblock の長方形内で定義されているサイト タイプが表示されます。 • [Available] : Pblock に含まれるサイト数が表示されます。 • [Required] : Pblock に割り当てられたロジックに必要なサイト数が表示されます。 • [% Util] : Pblock に含まれるサイトの予測使用率 (%) が表示されます。 • [Carry Statistics] : Pblock に割り当てられた垂直キャリー チェーン ロジック オブジェクトの数が表示されます。Pblock に割り当てられた中で高さが最大のキャリー チェーンの Pblock の高さに対する比率も表示されます。この値が 100% を超えると、PlanAhead の DRC エラーおよび ISE マップ エラーが発生することがあります。 • [Clock Report] : Pblock に含まれるすべてのクロック信号 (ローカル、グローバル、リソース) および各クロックでクロック信号が供給されたインスタンスの数が表示されます。 • [RPM Statistics] : Pblock に割り当てられた RPM (相対配置マクロ) オブジェクトの数が表示されます。Pblock に割り当てられた中で最大の RPM の Pblock に対する高さとの比率も表示されます。 注記 : この値が 100% を超えると、PlanAhead の DRC エラーおよび ISE マップ エラーが発生します。PlanAhead では、複数の RPM が Pblock の長方形に収まるかどうかは示されません。 • [Clock Region Statistics] : Pblock が重なり合った各クロック領域の使用率が表示されます。 • [Primitive Statistics] : Pblock に割り当てられた論理リソースのタイプごとの数が表示されます。 注記 : [Export Statistics] ボタンをクリックすると、情報をテキスト形式のファイルに保存できます。
[Instances]	Pblock に含まれるインスタンスの情報が表示されます。インスタンスのフィールドは選択可能で、多くのポップアップ メニュー コマンドが使用できます。
[Rectangles]	Pblock の長方形の情報が表示されます。[Rectangle] タブを使用して、Pblock の長方形を選択できます。詳細は、 360 ページの「長方形以外の形の Pblock の使用」 を参照してください。
[Attributes]	Pblock の属性を定義できます。詳細は、 361 ページの「Pblock の属性の設定」 を参照してください。

注記：変更を反映させるには [Apply] をクリック、キャンセルするには [Cancel] をクリックします。[Apply] をクリックせずに、ほかのアイテムを選択するか、[Pblock Properties] ビューを閉じると、変更は適用されません。

Pblock の設定

次のセクションでは、Pblock の設定について説明します。

Pblock ロジック タイプ範囲の設定

[Pblock Properties] ビューの [General] タブで [Grid Range] オプションを指定すると、Pblock の AREA_GROUP の範囲を設定できます。範囲は図 10-10 のように Pblock の長方形内にあるロジック タイプ別に設定します。

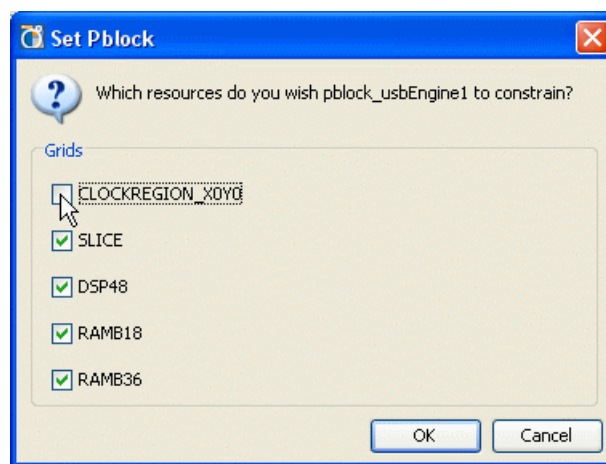


図 10-10 : ロジック タイプ別の AREA_GROUP の範囲設定

Pblock をブロック RAM や DSP などの新しいデバイス ロジック タイプを含む場所に移動したり、そのサイズを変更したりすると、Pblock の定義に新しい RANGE タイプを入力するダイアログボックスが表示されます。このダイアログボックスの内容は、Pblock の位置によって異なります。グリッド範囲のオプションをオフにすると、[Device] ビューでの表示が変化します。

注記：[Set Pblock] ダイアログボックスには、Pblock がデバイスのクロック領域を含んでいる場合は、クロック領域の範囲のみがレポートされます。

354 ページの図 10-11 に示すように、影の部分では選択された Pblock に設定された範囲のロジック タイプのみを示します。

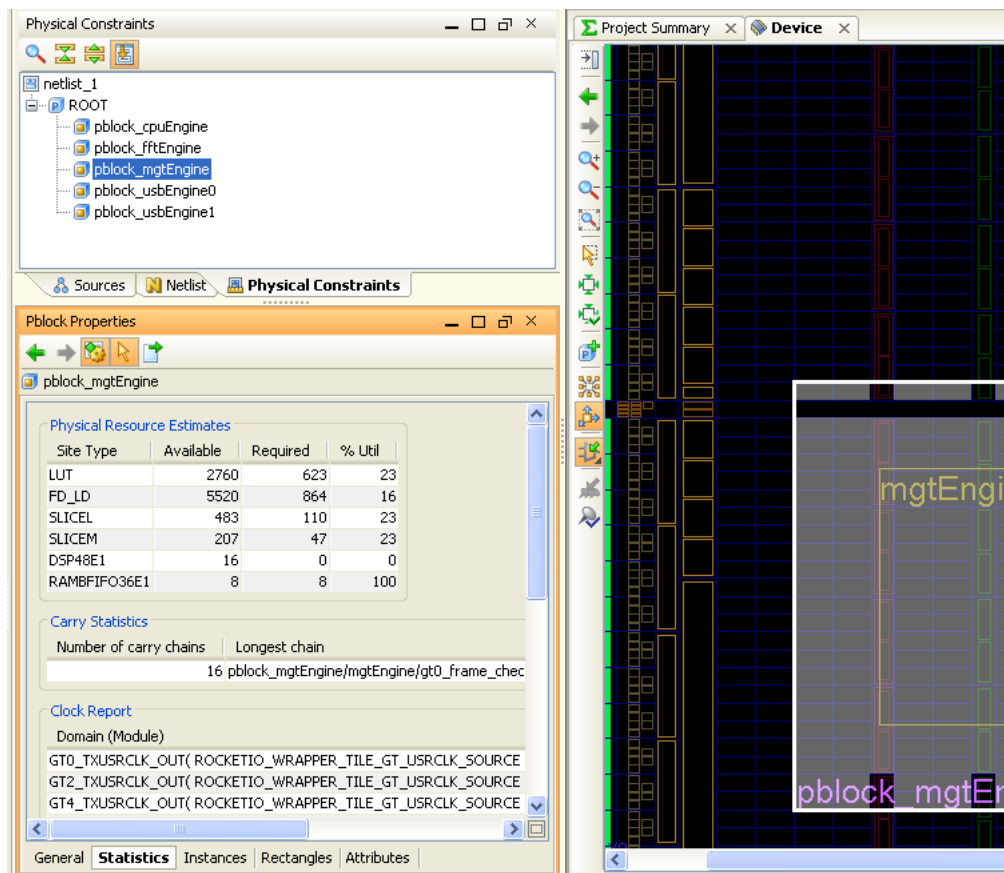


図 10-11 : Pblock に含まれるロジックの表示

Pblock へのロジックの割り当て

Pblock を作成すると、ネットリスト インスタンスを割り当てることができます。割り当てるには、そのロジックをドラッグ アンド ドロップするか、[Assign] ポップアップ コマンドを使用します。

次に、ドラッグ アンド ドロップする際の手順を示します。

1. [Netlist]、[Schematic]、[Hierarchy]、または [Find Results] ビューでロジック インスタンスをクリックし、ドラッグします。
2. Pblock の長方形にインスタンスをドロップします。

[Assign] コマンドを使用する手順は、次のとおりです。

1. [Netlist] ビューでロジック インスタンスを選択します。
2. [Assign] をクリックします。

355 ページの図 10-12 のように、[Select Pblock] ダイアログ ボックスが表示され、割り当て可能な Pblock が表示されます。

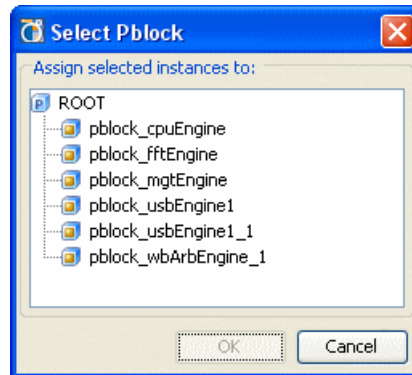


図 10-12 : [Select Pblock] ダイアログ ボックス

Pblock のロジック割り当て解除

インスタンスを Pblock から削除するには、次の手順に従ってください。

1. インスタンスを選択します。
2. [Unassign] ポップアップ メニュー コマンドをクリックします。

確認のダイアログ ボックスが表示されます。

Pblock の移動とサイズ変更

ここでは、Pblock の移動およびサイズ変更方法について説明します。

Pblock の移動

Pblock を移動するには、[Device] ビューで Pblock を選択してドラッグし、移動先でドロップします。Pblock が選択されると、カーソルが手のアイコンに変わります。Pblock 内のインスタンスではなく、外側の Pblock の長方形を必ず選択します。

ブロック RAM や DSP などの新しいデバイス ロジック タイプを含む場所へ Pblock を移動すると、Pblock の定義に新しい RANGE タイプを入力するダイアログ ボックスが表示されます。

割り当てられたロジックに Pblock 内部への BEL または LOC 配置制約が付いている場合、Pblock の動作が異なります。移動先には、配置制約の割り当てに十分なリソースが必要です。Pblock をドラッグすると、カーソルの形でそのサイトへ移動可能であるかどうかを示されます。移動不可能な場所に Pblock を移動しようとする、そのロケーション制約を削除するか、そのままにするかを尋ねる [Choose LOC Mode] ダイアログ ボックス (356 ページの図 10-13) が表示されます。

使用できる動作は次のとおりです。

- **[Move location constraints with the Pblock site ranges]**

固定の有無に関わらず、現在配置されているインスタンスをすべて移動し、それらの LOC および BEL 制約も Pblock の移動にあわせて新しい位置に割り当て直します。詳細は、365 ページの「固定された配置制約および固定されていない配置制約について」を参照してください。

このモードは、Pblock 内でインスタンスの配置を保持するのに便利な方法です。

- **[Move unfixed location constraints with the Pblock site ranges]**

現在配置されていても固定されていないインスタンスを移動し、それらの LOC および BEL 制約も Pblock の移動にあわせて新しい位置に割り当て直します。配置済みで固定されているインスタンスは、現在の位置に残ったままになります。

このモードは、Pblock と一緒に固定されていない位置を移動し、固定済みインスタンスの配置を現在の位置のまま保持する場合に便利です。

- **[Leave all location constraints in their current position]**

配置済みロジック インスタンスすべてを現在のロケーションに残し、Pblock のロジックを削除して Pblock および残りすべてのロジックを新しく選択したサイト範囲に移動します。

このモードは、多数のインスタンスが既に配置されていて、それらをすべて保持する場合に便利です。Pblock は新しいデバイス領域に移動できます。

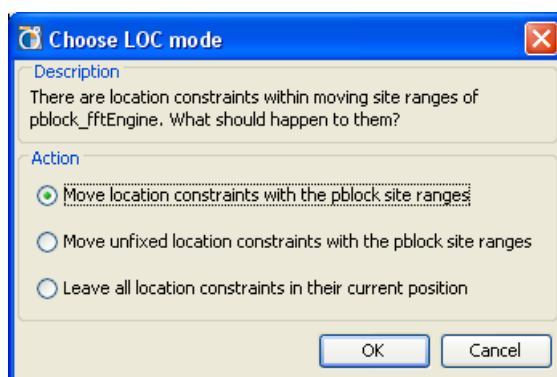


図 10-13 : Pblock と LOC の移動

注記：Pblock の移動がうまくいかない場合は、[Set Pblock Size] ツールバー ボタンをクリックし、長方形を別の場所に描画し直します。または、Pblock を移動する前に、配置制約を削除してみてください。

Pblock のサイズ変更

Pblock の大きさを変更するには、Pblock を選択し、カーソルを長方形の角または枠に置きます。

- カーソルの形がドラッグ シンボルに変化したら、マウス ボタンを押したままドラッグしてサイズを変更します。
- 実行中の移動操作をキャンセルするには Esc キーを押します。

Pblock のサイズを小さくすると、Pblock に割り当てられた配置済みインスタンスの一部がバウンダリの外にはみ出してしまうことがあります。この場合、357 ページの図 10-14 に示すような [Choose LOC Mode] ダイアログ ボックスが表示され、次のオプションが選択できます。

- **[Leave all location constraints in their current position]**

配置済みロジック インスタンスをすべて現在の位置に残し、Pblock のバウンダリ外にはみ出たロジックを削除します。

- **[Delete location constraints on sites trimmed from Pblock]**

固定の有無に関わらず、サイズを変更した Pblock の外にはみ出たロジックの配置をすべて解除しますが、Pblock に割り当てられたインスタンスは残します。未配置のインスタンスは、新しい Pblock に相対的に配置されます。

- **[Delete unfixed location constraints on sites trimmed from Pblock]**

サイズを変更した Pblock の外にはみ出た固定されていないロジックの配置をすべて解除しますが、固定されているインスタンスは現在の位置に残ったままになります。これは、固定されたロジックは保持しますが、インスタンスはの割り当ては Pblock から解除されます。

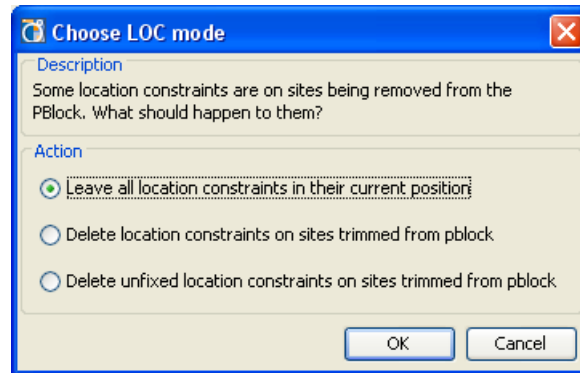


図 10-14 : Pblock と LOC のサイズ変更

[Set Pblock Size] コマンドの使用

[Set Pblock Size] コマンドを使用すると、既存の Pblock のサイズを設定したり変更したりして、新しい長方形を作成できます。このコマンドは、[New Pblock] コマンドなどで作成された、長方形がまだ定義されていない既存の Pblock の長方形を描画する場合にも使用できます。詳細は、[345 ページ](#)の「[Create Pblocks] コマンドを使用した複数の Pblock の作成」を参照してください。

既存の Pblock の新規の長方形を作成するには、次の手順に従います。

1. [Physical Constraints] または [Device] ビューで Pblock を選択します。
2. 右クリックで [Set Pblock Size] をクリックします。



カーソルが変化し [Device] ビューで新規の長方形を描画できるようになります。

選択した Pblock に複数の長方形が含まれる場合、このコマンドでは 1 つの長方形を含む Pblock が再生成されます。これは、フロアプランが進むにつれて、Pblock が複数の長方形に分割されてしまう場合に便利です。

3. 新しい長方形を描画します。

注記 : Esc キーを押すと、新しい Pblock を作成せずにコマンドをキャンセルできます。

新しい長方形を使用して Pblock のサイズを変更すると、Pblock に割り当てられた配置済みインスタンスの一部がバウンダリの外にはみ出してしまうことがあります。この場合、[図 10-15](#) に示すような [Choose LOC Mode] ダイアログ ボックスが表示され、次のオプションが選択できます。

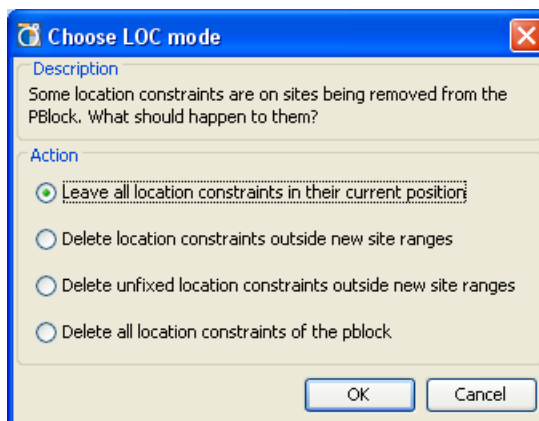


図 10-15 : LOC を使用した Pblock サイズの設定

- **[Leave all location constraints in their current position]**

配置済みロジック インスタンスをすべて現在の位置に残し、Pblock のバウンダリ外にはみ出たロジックを削除します。

- **[Delete location constraints outside new site ranges]**

固定の有無に関わらず、新しい Pblock バウンダリの外にはみ出たロジックの配置をすべて解除しますが、Pblock に割り当てられたインスタンスは残します。未配置のインスタンスは、新しい Pblock に相対的に配置されます。

- **[Delete unfixed location constraints outside new site ranges]**

新しい Pblock のバウンダリの外にはみ出た固定されていないロジックの配置をすべて解除しますが、固定されているインスタンスは現在の位置に残ったままになります。これは、固定されたロジックは保持しますが、インスタンスはの割り当ては Pblock から解除されます。

- **[Delete all location constraints of the Pblock]**

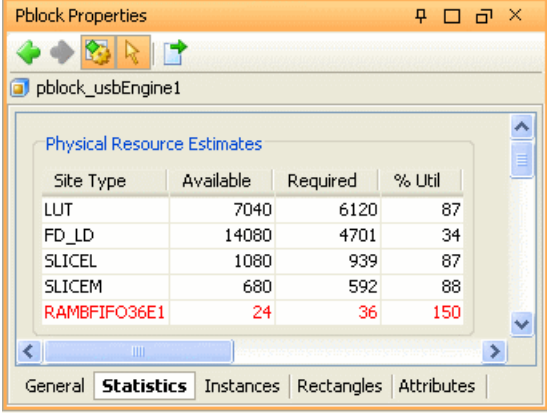
新しいバウンダリの内部に納まるか外部にはみ出るかに関係なく、Pblock 内のすべての配置済みインスタンスの配置を解除します。これにより、前に配置したインスタンスを処理しなくても、Pblock を配置し直すことができます。

リソース使用率の統計を使用した Pblock サイズの決定

[Pblock Properties] ビューの使用率の統計を使用して、Pblock の大きさを決定し、配置できます。Pblock に割り当てられたロジックに必要なリソースが概算され、使用可能なデバイス リソースと比較されて、使用率が概算されます。

Pblock の予測使用率は、次の手順で表示できます。

1. Pblock を選択し、[Pblock Properties] を表示します。
2. 359 ページの図 10-16 に示す [Statistics] タブをクリックします。



Site Type	Available	Required	% Util
LUT	7040	6120	87
FD_LD	14080	4701	34
SLICEL	1080	939	87
SLICEM	680	592	88
RAMBFIFO36E1	24	36	150

図 10-16 : [Pblock Properties] ビュー :[Statistics] タブ

- [Statistics] タブで、次の列の予測使用率を確認します。
 - [Available] : Pblock に含まれるサイト数が表示されます。
 - [Required] : 割り当てられたロジックに必要なサイト数が表示されます。
 - [% Util] : ロジック タイプごとの予測使用率 (%) が表示されます。Pblock のサイズを変更すると、使用率を適正な値に修正できます。ロジック オブジェクトの使用率が 100% を超える場合は、図 10-16 のように赤字のテキストで表示されます。
- スクロールダウンして、Pblock に必要な RAM サイト数を確認します。
このビューは、Pblock が変更されるたびにダイナミックにアップデートされます。

Pblock に特定のロジック デバイス エレメントのサイトが含まれない場合は、次の値が表示されます。

- [Available] : 0
- [Required] : 必要な数
- [Utilization] : Pblock のロジックで使用されるリソースの割合 (%) です。
 - 値が [Disabled] の場合は、[General] タブで [Site Type] がディスエーブルになっており、使用できないことを示しています。このタイプのサイトが Pblock で必要とされる場合、エラーになります。
 - 値が [No Sites] の場合は、デバイスの Pblock 範囲にそのタイプのサイトが含まれないことを示しています。

注記 : Pblock のスライス使用率は、サイトの使用率が最大であるという前提で計算されていますが、実際は配置配線ツールでサイトの使用率が最大になることはまれです。設計時には、ターゲットの使用率は約 80% 以上で最適化してください。この数値は、使用されるデバイスおよびデザインとその制約によって異なります。

注記 : Pblock 使用率は、キャリー チェーン、RPM マクロ、および Pblock 長方形の配置の影響を受けます。これらの数値は単なる予測で、ISE インプリメンテーションを正しく実行するための指針として使用するためのものです。Pblock のサイズの決定時に、Pblock の統計をすべて考慮してください。正しく配置できるように、デザイン ツールで Pblock を拡大する必要がある場合もあります。

接続に基づいた Pblock の配置

PlanAhead の接続フィードバックはダイナミックで、Pblock の配置に役立ちます。348 ページの図 10-5 は、接続が表示された例を示しています。

Pblock 間の結合された接続は、バンドル ネットとして表示されます。各バンドルは、Pblock 間の接続数に基づいてサイズと色が決まるので、接続数が多い Pblock がわかりやすくなっています。

バンドル ネットが大きな Pblock 同士を近くに配置するのが合理的なアプローチです。通常 Pblock は、ネット長ができるだけ短くなるように配置し、配線競合や配線密集が起こらないようにします。

バンドル ネット プロパティの表示

接続情報は、バンドル ネットまたは個別のネットのプロパティを表示して確認できます。接続情報は、次の手順で表示できます。

1. ネットまたはバンドル ネットを選択します。
2. [Net Properties] ビューまたは [Bundle Net Properties] ビューを表示します。

図 10-17 のように、[Bundle Net Properties] ビューの [Nets] タブにはバンドルに含まれるネットが表示されます。

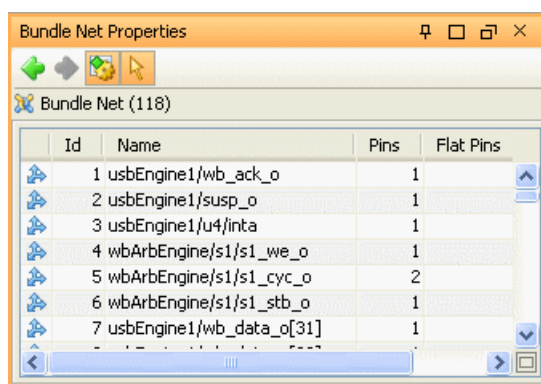


図 10-17 : [Bundle Net Properties] ビュー :[Nets] タブ

バンドル ネットのデフォルトの変更

色、線の幅、および信号カウント範囲は、[Tools] → [Options] → [Themes] → [Bundle Nets] で、ビューごとに指定できます。

長方形以外の形の Pblock の使用

PlanAhead では、1 つの Pblock に複数の長方形 Pblock を含めて、長方形以外の形の Pblock を作成、変更、削除できます。Pblock の形が長方形以外の場合、ダウンストリーム ツールでより困難になる可能性が増えるため、この方法は必要な場合にのみ使用してください。

複数の長方形を含む Pblock の作成

既存の Pblock に長方形を追加するには、Pblock を選択して [Add Pblock Rectangle] をクリックします。



Pblock の複数の長方形は、348 ページの図 10-5 のように点線で結ばれて表示されます。

長方形以外の形の Pblock の変更

AREA_GROUP 制約ごとの範囲の数が多いと、最適に処理できません。1 つの長方形 Pblock ではカバー仕切れない場合、L 型や T 型のように 2 つの長方形を組み合わせてシンプルな形のコンフィギュレーションを使用するをお勧めします。

複数の長方形から構成されている Pblock を選択すると、含まれる長方形がすべて選択されます。長方形は個別にも、すべて同時にも移動できます。

複数の長方形を含む Pblock で 1 つの長方形のサイズを変更する場合は、その長方形を選択して [Set Pblock Size] コマンドを使用するか、手動で変更します。

長方形を個別に選択するには、次のいずれかの手順に従います。

- Pblock の長方形を 1 つを右クリックして [Select] ポップアップ メニューをクリック
- [Pblock Properties] ビューの [Rectangles] タブから個別に選択

PowerPC® (405 と 440) プロセッサ、シリアル トランシーバー サイトまたはコンフィギュレーション ブロックにまたがる Pblock には、自動的に複数の長方形が作成されることがあります。これは、インプリメンテーションに適切な長方形の範囲が定義されるためです。

Pblock 長方形の削除

Pblock を右クリックし、[Clear Rectangle] をクリックすると Pblock 長方形を削除できます。削除には次のいずれかの方法を使用します。

- 長方形を個別に削除
- 複数の長方形および Pblock を同時に削除

Pblock の長方形を削除しても、Pblock は [Physical Constraints] ビューからは削除されません。

Pblock の属性の設定

Pblock の属性は、[Pblock Properties] ビューの [Attributes] タブで割り当てることができます (図 10-18)。Pblock 属性では、インプリメンテーション結果に影響を与え、エラーの原因となる可能性のある ISE のオプションを定義します。

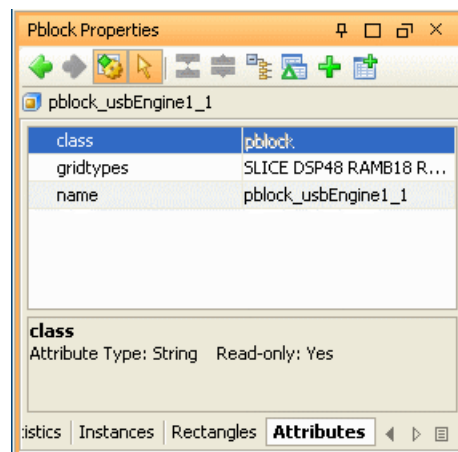


図 10-18 : [Pblock Properties] ビュー :[Attributes] タブ

Pblock の属性を次の方法で指定します。

1. [Pblock Properties] ビューの [Attributes] タブで右クリックして [Add pre-defined attributes] をクリックするか、[Add pre-defined attributes] ツールバー ボタンをクリックします。

図 10-19 のように、[Add Pre-defined Attributes] ダイアログ ボックスが表示されます。

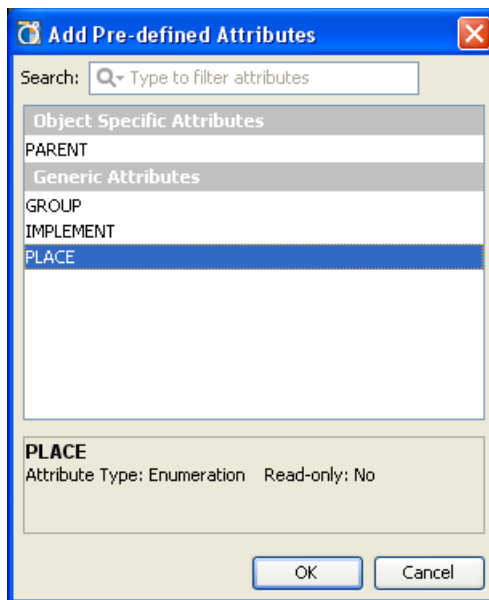


図 10-19：定義済み属性の追加

2. 属性を選択して [OK] をクリックします。
指定したタイプの属性が、[Attributes] タブに追加されます。
3. 属性値を指定し、[Apply] をクリックすると変更が適用されます。

Pblock の名前の変更

Pblock の名前は、[Pblock Properties] ビューの [General] タブから変更できます。新しい名前を [Name] フィールドで指定し、[Apply] をクリックします。[Pblock Properties] ビューの [Attributes] タブの [Name] から名前を変更できます。

Pblock の削除

次の手順で Pblock を選択して削除できます。

1. [Physical Constraints] ビューで 1 つまたは複数の Pblock を選択します。
2. Delete キーを押します。
3. [Confirm Delete] ダイアログ ボックスで、[Remove Pblock children] オプションをオンにすると、ネスト化された Pblock およびそのパーティションも削除されます。オフにしておくと、選択した Pblock のみが削除され、ネスト化されたその他の Pblock は、1 階層上に移動します。
4. [OK] をクリックすると、[Physical Constraints] ビューから Pblock パーティションが削除されます。

Pblock の自動作成

自動的に Pblock を作成するには、[Tools] → [Floorplanning] → [Auto-create Pblocks] をクリックします。このコマンドは、最上位の Pblock を作成してデザインのデータフローを表示し、デザインのロジック モジュール間の相対サイズおよび関係を理解するために使用されます。363 ページの図 10-20 に、[Auto-create Pblocks] ダイアログ ボックスを示します。

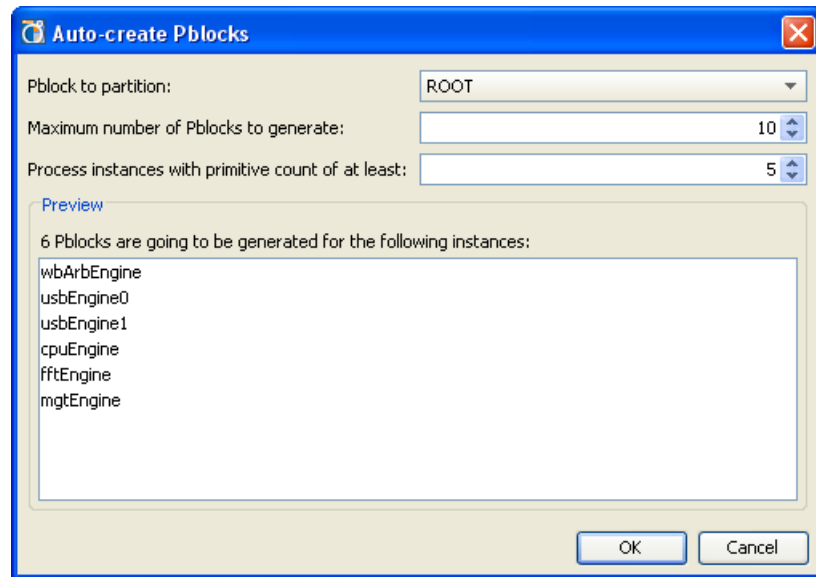


図 10-20 : [Auto-create Pblocks] ダイアログ ボックス

次のオプションを設定します。

- [Pblock to partition] : パーティションする既存の Pblock の名前を入力するか、デザインの最上位をパーティションする場合は ROOT を指定します。
- [Maximum number of Pblocks to generate] : デザイン階層または選択した Pblock に基づいて生成する Pblock 数を指定します。このオプションの値により、作成される Pblock の数を増やしたり、減らしたりできます。
- [Process instances with primitive count of at least] : デザイン階層内のインスタンスの検証を、指定した数以上のプリミティブを含むモジュールに制限します。このオプションを使用すると、Pblock を作成するインスタンスの数を削減できます。
- [Preview] : 定義済みパラメーターに基づいて作成される Pblock の名前と数を表示します。

[OK] をクリックすると、指定した Pblock が作成されます。PlanAhead で Pblock が作成され、[Physical Constraints] ビューの定義済み Pblock のリストに追加されます。Pblock をクリックすると、[Physical Constraints] ビューでその内容および属性が確認できます。

[Auto-create Pblocks] コマンドを実行したとき、Pblock は [Device] ビューに追加されません。Pblock 制約が作成され、Pblock にロジック エレメントが割り当てられます。Pblock の作成後には、デバイスに配置する必要があります。

Pblock 自動配置プログラムの実行

[Synthesized Design] または [Implemented Design] で Pblock を作成する場合、[Tools] → [Floorplanning] → [Place Pblocks] をクリックして [Device] ビューの FPGA ロジック上に Pblock を配置できます。

この Pblock 配置コマンドは、スライスの内容にのみ基づいて自動的に Pblock のサイズを変更して配置します。

[Place Pblocks] コマンドを使用して Pblock のサイズが変更され配置されている間は、ほかのロジック エLEMENTは無視されます。結果は決まっておらず、run によって異なります。このコマンドを使用すると、指定した Pblock が迅速に配置されるので、ユーザーはデザイン モジュールを介してデータ フローを確認できます。

注記：[Place Pblocks] コマンドで配置した Pblock は、インプリメンテーションには使用できない場合があります。非スライス ベースのロジック エLEMENTを考慮して、手動で Pblock のサイズを変更する必要があることもあります。

Pblock 配置コマンドは、次の手順で実行します。

1. [Tools] → [Floorplanning] → [Place Pblocks] をクリックします。

 **図 10-21** のように、[Place Pblocks] ダイアログ ボックスが開きます。

2. オプションを次のように変更します。

- [Parent Pblock] : Pblock を配置する階層のレベルを指定します。Pblock を配置できる箇所は次のとおりです。
 - 最上位モジュール
 - ルート
 - パーティションされた Pblock レベルの階層
- [Pblocks to place] : 親 Pblock またはルートの下にある Pblock のリストが表示されます。
 - [Place] : Pblock を配置するかどうかを指定します。オフにすると、Pblock が無視され、既存の Pblock 位置が保持されます。
以前にサイズ変更されて配置されたリスト内の Pblock は、[Place] をオフにしない限り、PlanAhead でリセットされ、サイズ変更されて再配置されます。
 - [Pblock] : 指定したレベルの階層内の Pblock の名前をリストします。
 - [Utilization] : 各 Pblock のスライス使用率ターゲットを設定します。PlanAhead は、指定した [Utilization] の使用率に従って Pblock のサイズを変更して配置します。

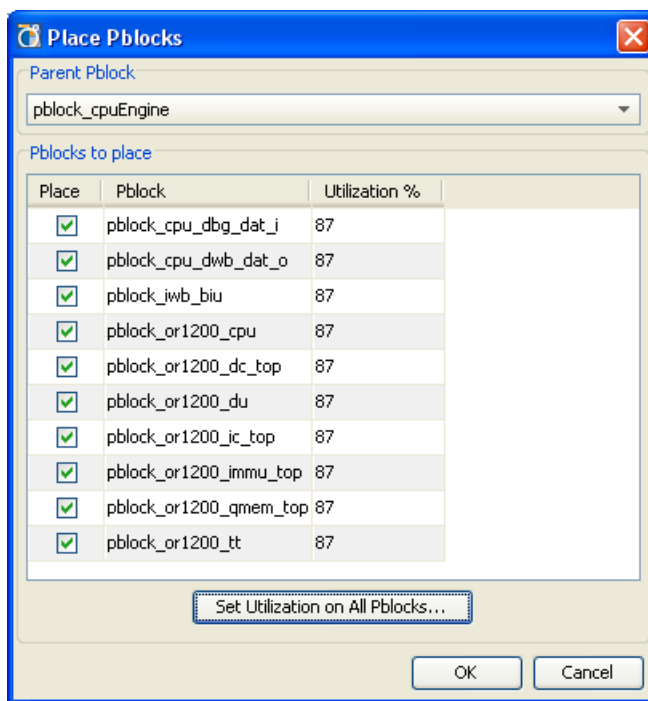


図 10-21 : [Place Pblocks] ダイアログ ボックス

- [Set Utilization on all Pblocks] : すべての Pblock にスライス使用率ターゲットを設定します。このオプションは、前に変更した個別の使用率の値よりも優先されます。まずこのオプションを使用してすべての Pblock に適切な値を指定してから、指定した Pblock の使用率を必要に応じて変更してください。

3. [OK] をクリックすると、Pblock がデザインに配置されます。

[Place Pblocks] コマンドの実行中は、[Place Pblocks Progress] ダイアログ ボックスが表示されます。Pblock はスライス使用率にのみ基づいて大きさが決定され、配置されます。

[Create Pblocks] コマンドおよび [Place Pblocks] コマンドを使用して最上位フロアプランを作成する方法は、付録 E「その他のリソース」に示す『デザイン解析およびフロアプラン チュートリアル : PlanAhead デザイン ツール』(UG676) を参照してください。

LOC 制約および BEL 制約の設定

[Create Site Constraint Mode] または [Create BEL Constraint Mode] を使用すると、プリミティブロジック エレメントを特定のロジック サイトに割り当てることができます。PlanAhead では、ロジック エレメントを特定のスライスに割り当てたり、スライス内の使用可能なリソースに配置を制限するときに LOC 制約を、ロジック インスタンスをスライス内の特定のサイトに割り当てるときに BEL 制約を使用します。LOC 制約は通常スライスに割り当てられ、BEL 制約は特定のサイトに割り当てられます。

固定された配置制約および固定されていない配置制約について

PlanAhead では、ユーザーが割り当てたロジック配置とツールで自動的に割り当てたロジック配置が区別されます。

- ユーザーが割り当てた配置は、インポートされた制約ファイルで定義されるか、または PlanAhead で割り当てられます。ユーザーが割り当てた制約は「固定」とみなされて同色で表示されます。
- インプリメンテーション中に PlanAhead または ISE で定義された配置制約は、「固定されていない」とみなされて、別色で表示されます。

配置されていないインスタンスの配置制約を固定するには、[Device]、[Package]、または [Schematic] ビューで [Fix Instances] または [Fix Ports] コマンドを使用します。

PlanAhead では、固定した制約がデフォルトで ISE インプリメンテーション ツールにエクスポートされて配置がロックされます。

[File] → [Export] → [Export Constraints] および [File] → [Export] → [Export Pblocks] をクリックして表示されるダイアログ ボックスには、固定されている制約とされていない制約のどちらもエクスポートできるオプションがあります。

図 10-22 に、[Export Pblock] コマンドを使用して物理階層でのデザインの特定期間をエクスポートする方法と、[Export IP] コマンドを使用して論理階層デザインの特定期間をエクスポートする方法を示します。

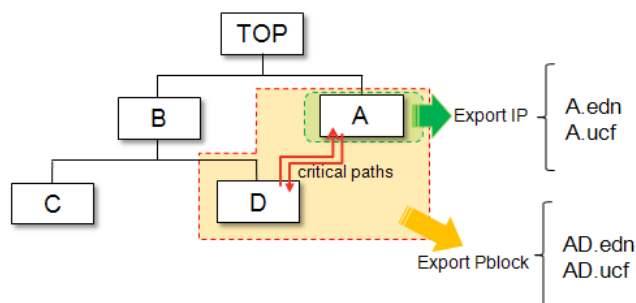


図 10-22 : [Export Pblock] コマンドおよび [Export IP] コマンドの使用例

サイト制約および BEL 制約について

インスタンスをサイトに配置すると、UCF ファイルに LOC 制約または LOC と BEL 制約が追加されます。このロケーションは固定ロケーションとして割り当てられ、この後のインプリメンテーション run の実行中にロックされます。

サイト制約はインスタンスに割り当てられる LOC 制約になります。ロジック エレメントは CLB スライスのみには割り当てられ、特定のサイトやリソースには割り当てられません。次はコード例の一部です。

```
INST "receiver/uartInst/G_98_1" LOC = SLICE_X49Y69;
```

BEL 制約ではロジック エレメントが CLB スライス内の特定のスライスに割り当てられるので、LOC 制約および BEL 制約は保存およびエクスポートされた UCF ファイルのインスタンスに割り当てられます。次の例では、インスタンスに対する BEL および LOC 制約を示しています。

```
INST "channel/receiverRE[8]" BEL = FFX;
```

```
INST "channel/receiverRE[8]" LOC = SLICE_X59Y2;
```

[Device] ビューで [Instance Drag & Drop Mode] をクリックすると、デバイスに配置するインスタンスの配置制約をドラッグ アンド ドロップで指定できます。



- [Create BEL Constraint Mode] : LOC および BEL 制約を配置されるインスタンスに割り当てます。これにより、インスタンスがスライス内の特定の BEL に固定できます。
- [Create Site Constraint Mode] : LOC 配置制約を配置されるインスタンスに割り当てます。これにより、インスタンスは特定スライスに固定されますが、スライス内の特定サイトには割り当てられません。
- [Assign Instance to Pblock Mode] : ロジック インスタンスを Pblock に割り当てます。これがデフォルト モードです。正しいフロアプランにするためには、できる限りこのモードを使用してください。

移動禁止または既に使用されているサイトにはインスタンスを配置できません。不正なサイトに配置しようとすると、丸にスラッシュの入ったアイコンで、その配置ができないことが示されます。

インスタンスのスライスへの配置 (LOC)

LUT、ブロック RAM、またはフリップフロップのような下位プリミティブ インスタンスは、ネットリスト ツリーからドラッグして特定サイトにドロップすると特定のデバイス リソースに配置できます。[Device] ビューの [Instance Drag & Drop Mode] を [Create Site Constraint Mode] に設定します。

図 10-23 は、特定スライスへのロジック エLEMENTの配置を示しています。ロジック エLEMENTはスライスの中央にドラッグされ、クリックしてスライスにELEMENTを割り当てた場合にのみ、特定のデバイス リソースに割り当てられます。

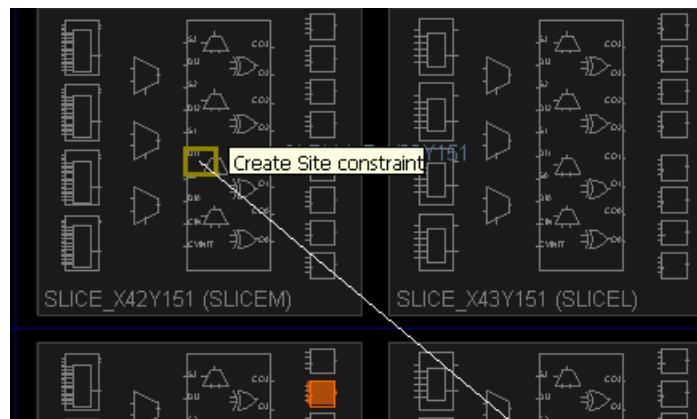


図 10-23 : サイト制約の作成

ダイナミック カーソルでは、ロジックがスライスに入りきらない場合にインスタンスを配置できません。キャリア チェーン ロジックなど、ロジックのグループによっては、1 つのオブジェクトとして移動するため、グループのロジックがすべて配置できる場所が必要です。

配置済みインスタンスの LOC 制約は [Instance Property] ビューに表示されます。

インスタンスのサイトへの配置 (BEL)

下位プリミティブ インスタンスは、ネットリスト ツリーからドラッグして特定のデバイス リソース サイトにドロップすると配置できます。[Device] ビューの [Instance Drag & Drop Mode] を [Create BEL Constraint Mode] に設定します。

BEL 制約では、ロジック エLEMENTがスライスとスライス内の特定デバイス リソースに割り当てられます。図 10-24 は、特定デバイス リソースに配置されたロジック エLEMENTを示しています。ロジック エLEMENTは、特定のデバイス リソースにドラッグすると、直接配置されます。

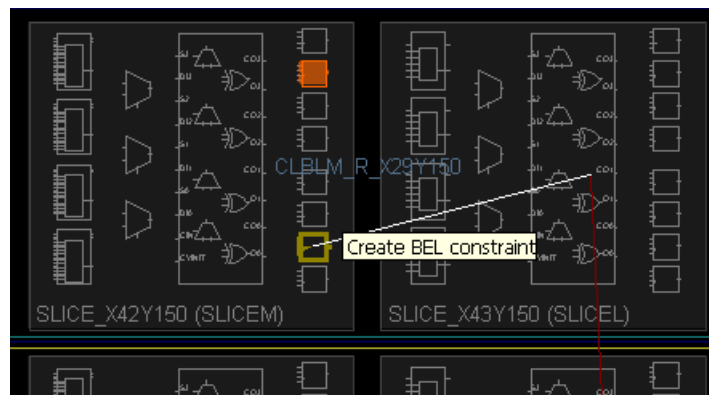


図 10-24 : BEL 制約の作成

ダイナミック カーソルでは、移動禁止または既に使用されているゲート サイトにはインスタンスを配置できません。配置できる場所に移ると、ダイナミック カーソルは斜線の入った円から矢印に変化します。

配置制約の表示/非表示

割り当てられた配置制約の表示方法を変更するにはズーム レベルを調節します。

- ズーム アウト表示では、LOC および BEL は、割り当てられたサイトの内側に塗りつぶされた長方形として表示されます。
- ズーム レベルが上がるに従って、サイト内の特定のロジック ゲートに割り当てられたロジックとして表示されるようになります。

369 ページの図 10-25 に、[Device] ビューのズームアウト表示 (右側) とズームイン表示 (左側) を表示します。ズーム レベルによって表示される詳細が異なることがわかります。

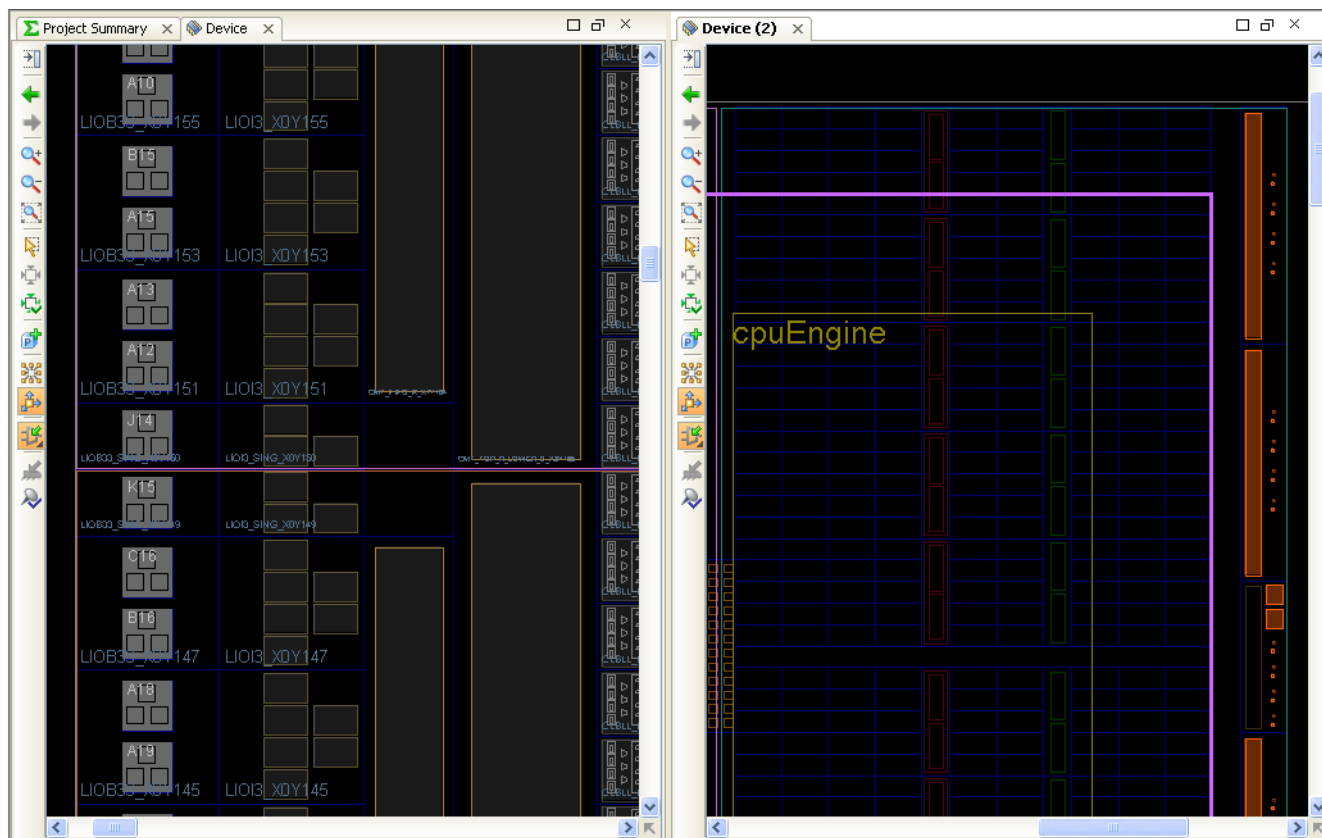


図 10-25 : [Device] ビューのズームイン表示およびズームアウト表示

[Device] ビューのレイヤー制御を使用すると、ロケーション制約を表示または非表示にできます。詳細は、155 ページの「[Device] ビューの表示オプションの設定」を参照してください。

LOC 制約および BEL 制約のその他の表示設定を変更するには、次の手順に従います。

1. [Tools] → [Options] → [Themes] → [Device] タブをクリックします。

[Device] タブには、[Device] ビューのさまざまなコンフィギャブル エLEMENT が表示されます。

2. [Select] 列の設定を変更するか、[Frame Color] 列および [Fill Color] 列で色を変更します。

配置が固定されたインスタンスと固定されていないインスタンスでは、個別に色および選択を設定できます。

配置済みインスタンスの移動

配置済みインスタンスを移動するには、次の手順に従います。

1. [Device] ビュー、[Netlist] ビュー、または [Schematic] ビューで配置済みインスタンスをクリックします。複数のインスタンスを選択して、別の位置へ同時にドラッグ アンド ドロップすることもできます。
2. 選択したインスタンスを移動可能なほかのサイトにドラッグ アンド ドロップします。

プリミティブ インスタンスが新しいサイトに割り当てられます。インスタンスから配置済みロジックまたは Pblock に接続するネットの線を表示できます。

MUX、キャリー チェーンなどの組み合わせロジック オブジェクトを移動すると、選択された LOC のグループ全体が移動されます。グループ全体が移動可能なサイトに移動するとカーソルが変化し、すべてのオブジェクトが新しい相対ロケーションに移動されます。

選択したインスタンスのロケーション制約のプロパティは、[Instance Property] ビューで確認できます。

配置ロケーションの入れ替え

配置済みコンポーネントを 2 つ選択して、それらのロケーションを入れ替えることができます。ロケーションを入れ替えるには、次の手順に従います。

1. 使用可能なビューでコンポーネント インスタンスを 2 つ選択します。複数のポートを選択するには、Ctrl キーを押しながらポートをクリックします。
2. 右クリックで [Swap Locations] をクリックします。

固定されていないインスタンスを入れ替えると、これらのインスタンスが固定されます。

配置制約の削除とインスタンスの配置解除

配置済みのインスタンスをクリックして次のいずれかの手順に従うと、配置を解除できます。

- [Device] ビューまたは別のビューのポップアップ メニューで「Unplace」をクリックします。
これにより、FPGA デバイスのサイトから選択したオブジェクトの配置が解除され、プロセスの LOC 制約および BEL 制約が削除されます。

- [Tools] → [Floorplanning] → [Clear Placement] をクリックします。

[Clear Placement] コマンドを実行すると図 10-26 に示す Clear Placement Constraints ウィザードが開き、オブジェクトの配置を解除でき、インスタンスと I/O ポートのどちらか、または両方から LOC および BEL 制約が削除できます。

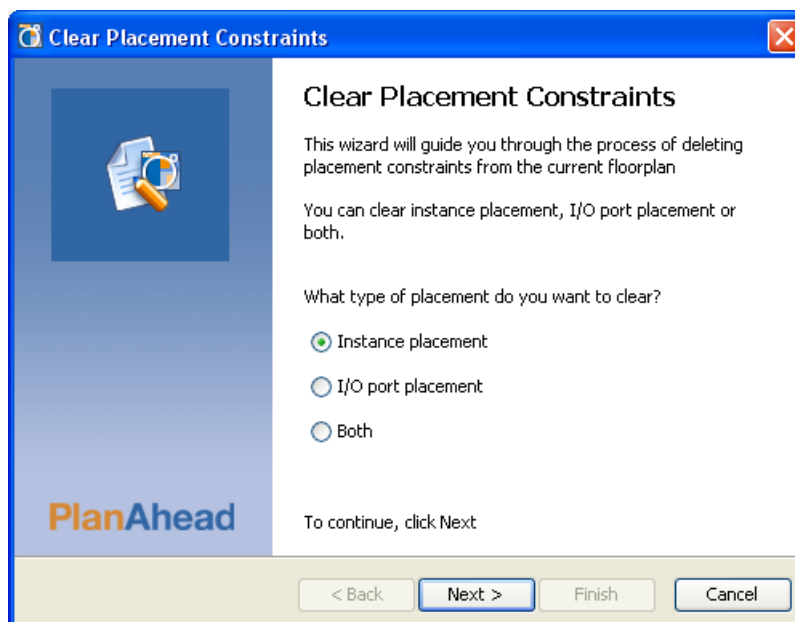


図 10-26：配置制約の削除

1. Clear Placement Constraints ウィザードで、削除する配置制約のタイプを選択します。[Instance placement]、[I/O Port placement]、または [Both] のいずれかを選択します。
2. [Next] をクリックします。

図 10-27 のように、[Unplace Instances] ダイアログ ボックスが開きます。

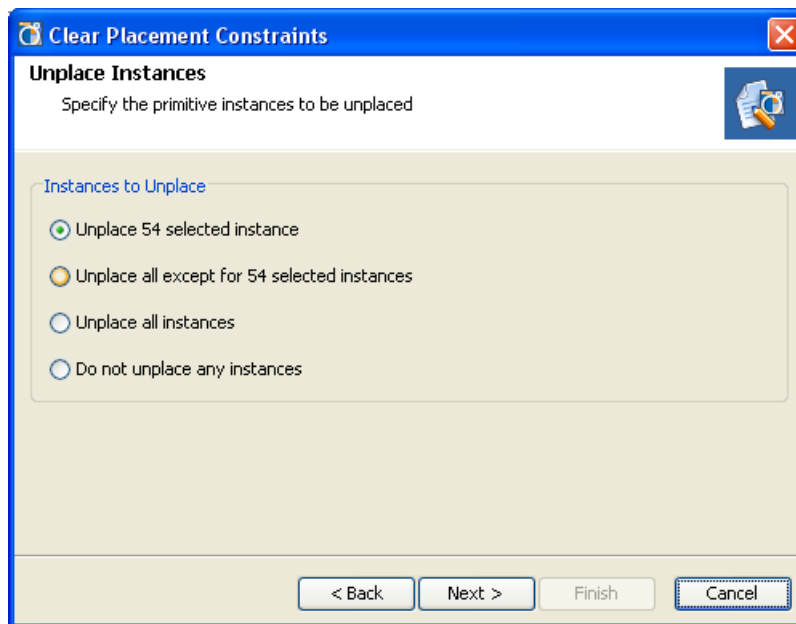


図 10-27 : インスタンスの配置の解除

3. [Clear Placement] コマンドの実行前にインスタンスを選択している場合は、次のいずれかを使用して、配置を解除するインスタンスを選択します。
 - [Unplace the selected instances] (選択したインスタンスの配置を解除)
 - [Unplace all but the selected instances] (選択したインスタンスを除いたすべてのインスタンスの配置を解除)
 - [Unplace all instances, regardless of the selected objects] (選択されているインスタンスも含めてすべて配置を解除)
 - [Do not unplace any instances, if you no longer want to unplace instances] (インスタンスの配置を解除しない)

注記：前にインスタンスを選択していない場合は、手順 2 は飛ばされ、デザインに含まれるインスタンスのリストが表示され、手順 4 のように配置を解除できます (372 ページの図 10-28)。

4. [Next] をクリックします。

372 ページの図 10-28 に示すように排除するインスタンス タイプのリストが表示されます。手順 2 で指定したインスタンス グループに含まれるインスタンス タイプから配置を解除するものを選択できます。

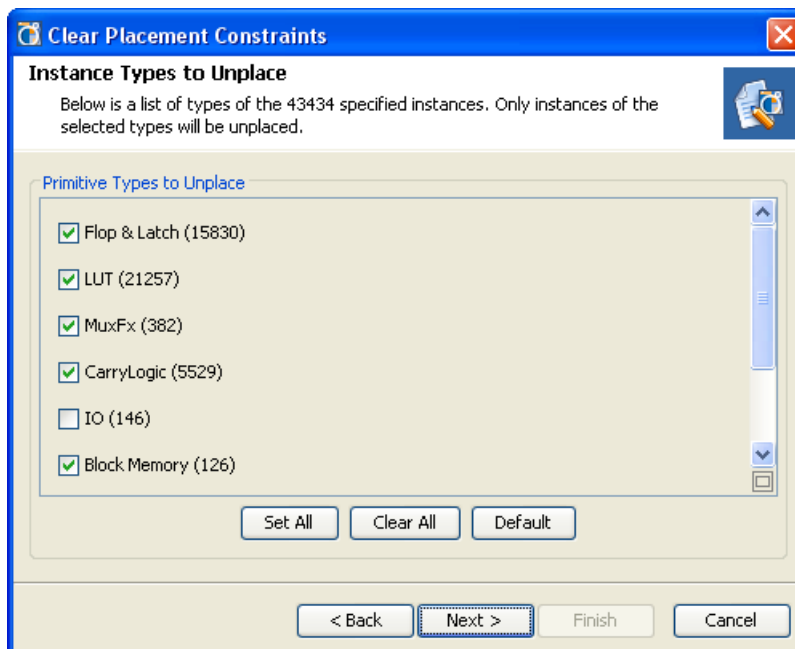


図 10-28：配置を解除するインスタンス タイプ

5. 配置を解除するインスタンス タイプを選択して、[Next] をクリックします。
- 選択したインスタンスまたはポートの一部が固定されている場合は、[図 10-29](#) に示すように [Fixed Placement] ページが表示されます。

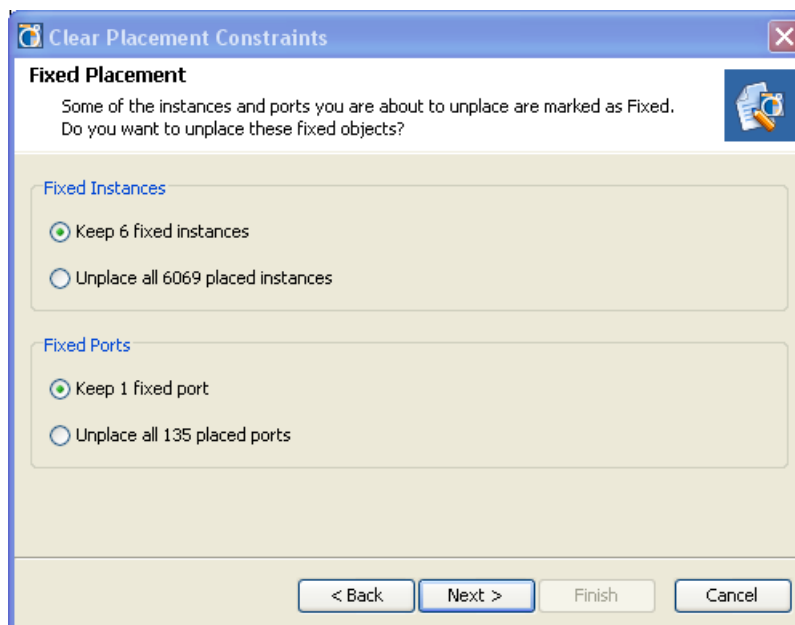


図 10-29：固定されたインスタンスおよびポートの配置の解除

6. このページでは、固定されているインスタンスの現在の配置を保持するか、解除するかを選択します。詳細は、[365 ページ](#)の「固定された配置制約および固定されていない配置制約について」を参照してください。

ポートの配置を解除する場合は、固定されているポートの配置を保持するか、すべてのポートの配置を解除するか選択します。

7. [Next] をクリックします。

[Clear Placement Summary] ページが開き、配置を解除したインスタンスのサマリが表示されます。

8. サマリの内容を確認し、[Finish] をクリックします。

- 指定された I/O ポートおよびインスタンスの配置がデバイスから削除されます。
- 前に割り当てられていたポートは、新しい UCF が PlanAhead に読み込まれるまで削除されません。
- 新しいポートの割り当てにより、前の割り当てが上書きされます。

注記：新しいポート割り当て制約をインポートする前にすべてのポート割り当てを削除するようにしてください。

PROHIBIT 制約の設定

PROHIBIT 制約を使用すると、デバイスのリソースを配置で使用されないように設定できます。この制約が設定されているリソースにはユーザーもソフトウェアもインスタンスまたはポートを配置できません。この制約は、複数のパーツ間でのデバイスの互換性を確実にする際に使用できます。互換パーツで共用されないリソースにこの制約を設定することで、パーツ間で簡単にデザインをマップできます。

PROHIBIT 制約はデバイスのどのロジック サイトにも作成できます。ビットストリームを生成するには、次の手順に従います。

1. [Device] ビューでサイトを選択します。
2. サイド ツールバー メニューから [Select Area] コマンドを使用して、複数サイトを選択します。詳細は、第 4 章の「[Select Area] コマンド」を参照してください。
3. 右クリックして表示されるポップアップ メニューから [Set Prohibit] を選択します。図 10-30 のように、使用禁止になったサイトには赤い X マークが表示されます。



図 10-30 : [Device] ビューで PROHIBIT 制約が設定されているサイト

PlanAhead からのデータの抽出

PlanAhead では、外部ソフトウェアの合成、シミュレーション、インプリメンテーションに必要なファイルを選択してエクスポートできます。

制約のエクスポート

制約を ISE にエクスポートする場合、デザイン全体の、または Pblock レベルの UCF 制約ファイルがエクスポートされます。

制約をエクスポートするには、次の手順に従います。

1. [Sources] ビューでエクスポートする制約セットをクリックし、[Make active] をクリックしてアクティブな制約セットに設定します。
2. [File] → [Export] → [Export Constraints] をクリックします。
[Export Constraints] ダイアログ ボックスが表示されます。
3. 次のようなオプションを変更します。
 - [File name] : 作成する UCF のファイル名とディレクトリを指定します。
 - [Export fixed location constraints only] : ユーザーが割り当てた「固定」配置 LOC 制約のみをエクスポートする場合はオンに、固定/未固定の区別なく、ISE からインポートしたすべての配置制約をエクスポートする場合はオフにし、[OK] をクリックします。

PlanAhead は、エクスポート ディレクトリに最上位の UCF 形式の制約ファイルを作成します。このファイルは、カスタムの ISE インプリメンテーション スクリプトの入力として使用できます。

エクスポート ファイルの詳細は、付録 A「レポートの出力」を参照してください。

ネットリストのエクスポート

PlanAhead ネットリストを ISE にエクスポートすると、デザイン全体または Pblock レベルの EDIF 形式のネットリスト ファイルがエクスポートされます。これには、[Synthesized Design] を開く必要があります。

ネットリストをエクスポートするには、開いた [Synthesized Design] から次を実行します。

1. [File] → [Export] → [Export Netlist] をクリックします。
[Export Netlist] ダイアログ ボックスが表示されます。
2. [Export Netlist] ダイアログ ボックスで作成する EDIF 形式のネットリスト ファイルのファイル名とディレクトリを指定し、[OK] をクリックしてネットリストをエクスポートします。

エクスポート ファイルの詳細は、付録 A「レポートの出力」を参照してください。

ISE インプリメンテーション用の Pblock のエクスポート

PlanAhead では、インプリメンテーション用に Pblock レベルのファイルのエクスポートができます。これらの Pblock には、ロジック階層のどのレベルからのロジックも含めることができます。

Pblock をエクスポートすると、選択した Pblock それぞれに対して EDIF ネットリストと UCF 物理制約ファイルが作成されます。

EDIF および UCF ファイルに Pblock をエクスポートするには、次の手順に従います。

1. (オプション) 1 つまたは複数の Pblock を選択します。
2. [File] → [Export] → [Export Pblocks] をクリックします。
375 ページの図 10-31 のような Export Pblocks ウィザードが開きます。

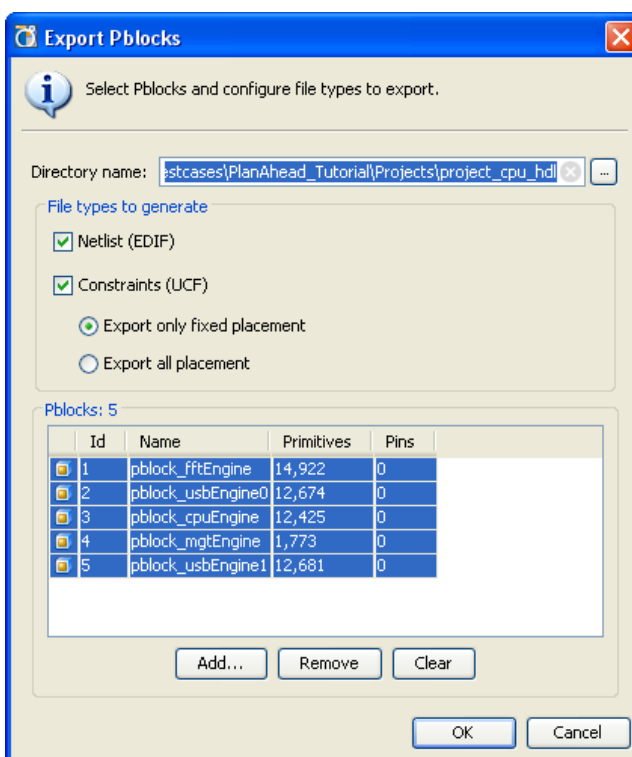


図 10-31 : Export Pblocks ウィザード

3. 次のオプションを変更します。

- **[Directory name]** : ディレクトリ名を入力するか、参照ボタンをクリックして指定した Pblock のファイルをエクスポートするディレクトリを選択します。エクスポートされる Pblock それぞれに対し、*pblockname_cv* というサブディレクトリが作成されます。

各 Pblock に関連付けられた EDIF ファイルおよび UCF ファイルは、ISE の run ごとにディレクトリ名を指定すると管理しやすくなります。エクスポートされた Pblock ディレクトリは、**[Import Placement]** コマンドおよび **[Import Timing]** コマンドのファイル参照に反映されます。

- **[File types to generate]** :
 - **[Netlist (EDIF)]** : ネットリストをエクスポートします。
 - **[Constraints (UCF)]** : すべて、または固定された配置制約のみをエクスポートします。
- **[Pblocks]** : エクスポート用に選択された Pblock がリストされます。

4. エクスポート リストに Pblock を追加する場合は **[Add]** を、削除する場合は **[Remove]** をクリックします。すべて削除する場合は **[Clear]** をクリックします。5. **[OK]** をクリックして Pblock をエクスポートするか、**[Cancel]** をクリックして操作をキャンセルします。

PlanAhead でエクスポートした Pblock の EDIF および UCF ファイルの名前は、それぞれ *pblockname_cv.edn* および *pblockname_cv.ucf* になります。

また、Pblock それぞれに対して */pblockname_cv* というディレクトリが作成され、Pblock 特有のファイルが含まれます。

インプリメンテーション結果の解析

PlanAhead™ では、インプリメントしたデザインの配置配線結果を解析するツールが複数提供されています。デザインは PlanAhead にインプリメントするか、ザイリンクス ISE® Design Suite から結果をインポートできます。

PlanAhead ツールでは、配置配線済みのデザインに対して次を実行できます。

- デザイン ルール チェック (DRC)
- SSN 解析
- SSO 解析
- タイミング解析
- 電力解析

また、次も実行できます。

- FPGA Editor を開いてインプリメント済みデザインの配線リソースを確認および変更
- ISim を開いてインプリメント済みデザインのタイミング シミュレーションを実行

インプリメント済みデザインを開く

インプリメント済みデザインを開くには、次のいずれかを実行します。

- メイン メニューから [Flow] → [Open Implemented Design] をクリックします。
- Flow Navigator で [Implementation] → [Open Implemented Design] をクリックします。

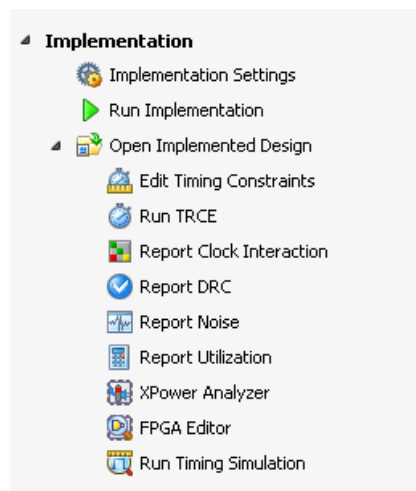


図 11-1 : Flow Navigator : [Implemented Design] メニュー

インプリメント済みデザインを開くと、Flow Navigator の [Implementation Design] メニューに使用可能な解析ツールが表示されます (377 ページの図 11-1)。

図 11-2 に [Implemented Design] のデフォルト ビューレイアウトを表示します。

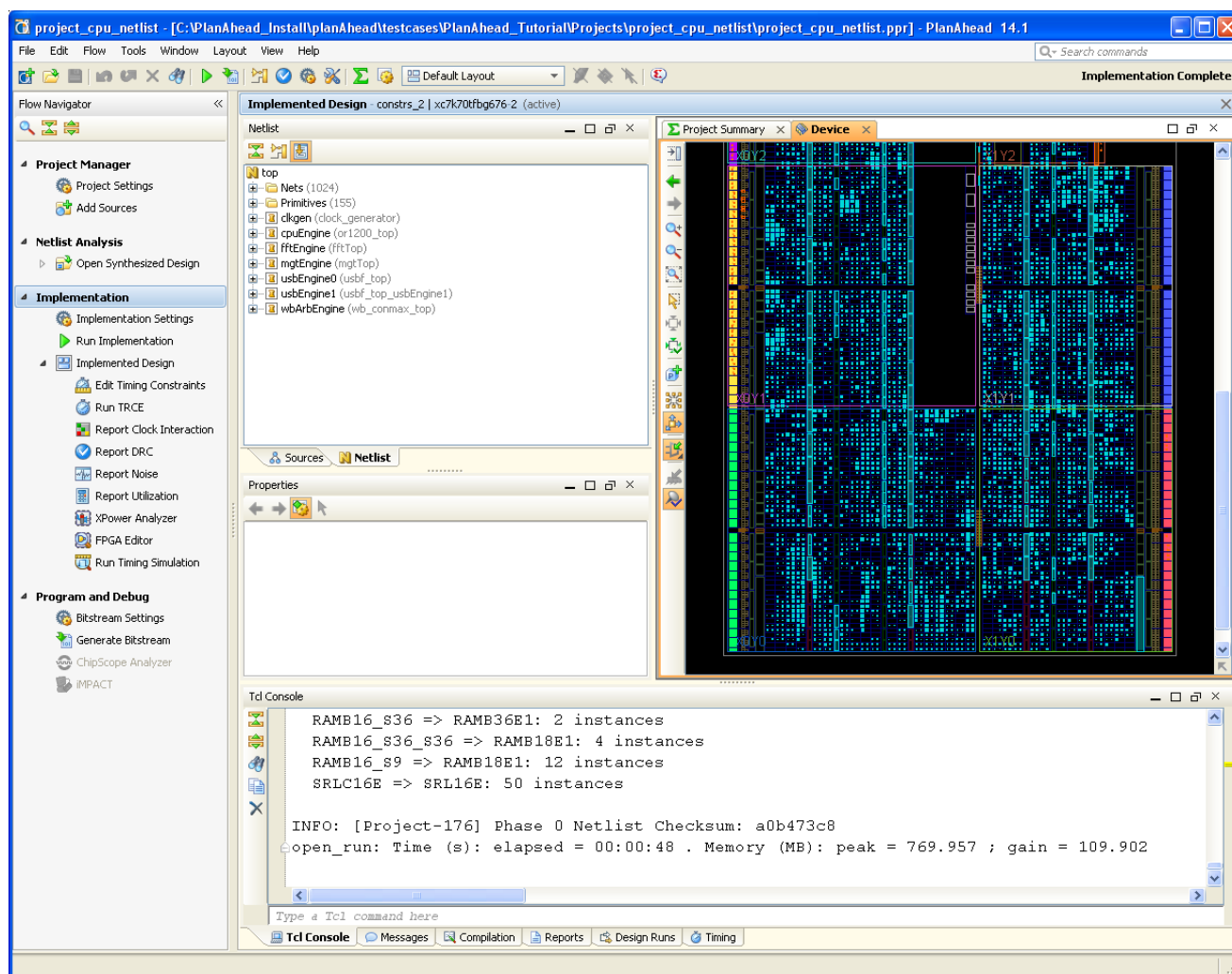


図 11-2 : [Implemented Design] 環境

[Device] ビューには配置済みのデザインが表示され、固定されたインスタンスおよび固定されていないインスタンスが別々の色で表示されます。固定されたインスタンスはユーザーによりインタラクティブに配置されたロジックを示し、固定されていないインスタンスはソフトウェアで自動的に配置されたインスタンスを示します。[Device]、[Package]、または [Schematic] ビューで [Fix Instances] または [Fix Ports] を使用して、配置されていないインスタンスを固定すると、後続のインプリメンテーション run 中にこれらのインスタンスが割り当て直されないようにすることができます。詳細は、365 ページの「LOC 制約および BEL 制約の設定」を参照してください。

注記： ISE ではロジックを最適化および変更して、配置配線結果を改善できます。この場合、元のネットリストのロジックは削除されるか置き換えられます。この結果、PlanAhead で開いているインプリメンテーション前のネットリストとインプリメンテーション結果が一致しなくなります。この問題は、[Implemented Design] を開いたときに [Tcl Console] ビューでレポートされます。ネットリストのロジックが表示された結果と一致しない以外は、問題ありません。

複数のインプリメント済みデザインを開く

インプリメンテーション run は、次のいずれかを実行すると開くことができます。

- [Design Runs] ビューでインプリメンテーション run を右クリックし、[Open Implemented Design] をクリックするか、インプリメンテーション run をダブルクリックします。
- run をアクティブ run に設定するには、Flow Navigator から [Open Implemented Design] をクリックします。詳細は、329 ページの「アクティブ run の設定」を参照してください。

インプリメント済みデザインが複数開いている場合は、図 11-3 に示すように、表示環境左上に表示されるタブをクリックすると、開いているデザインを切り替えて表示できます。

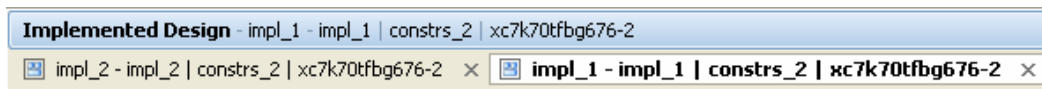


図 11-3：複数のインプリメント済みデザインのタブ

開いているデザインは表示バナーのタブ、または Flow Navigator の [Implemented Design] ポップアップメニューを使用して切り替えます。

Flow Navigator のプルダウンメニューを使用すると、使用可能なインプリメント済みデザインを開いたり、閉じたりできます。各インプリメンテーション run は、図 11-4 のように [Open Implemented Design] メニューから指定して開くことができます。

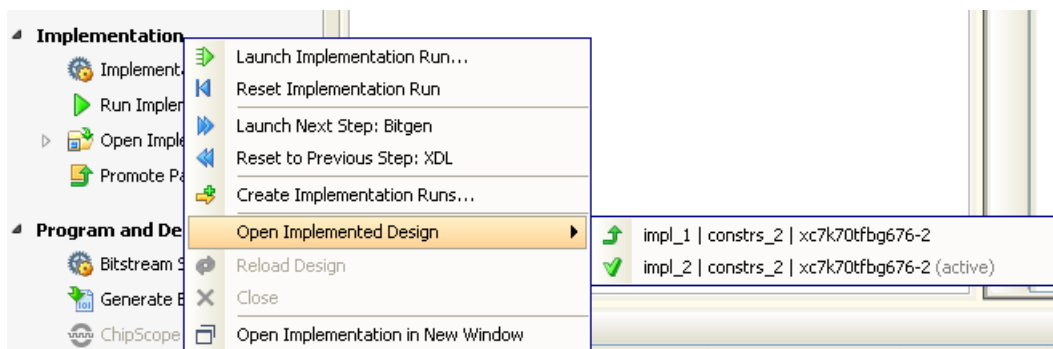


図 11-4：インプリメント済みデザインを開く

複数のインプリメント済みデザインが開いている場合は、[Implemented Design] ボタンに開いているデザインの数が表示されます。

インポートした配置を使用したインプリメント済みデザインの構築

New Project ウィザードを使用して既存の ISE 結果をインポートしたり、タイミング結果をインポートしたりできます。次のセクションでは、ISE 結果をインポートするオプションについて説明します。

New Project ウィザードを使用した配置のインポート

新規プロジェクトを作成し、そのプロジェクトに ISE のインプリメンテーションからのネットリスト、配置およびタイミング結果をインポートできます。ISE インプリメンテーション結果をインポートする新規プロジェクトの作成については、57 ページの「ISE の配置およびタイミング結果を使用したプロジェクトの作成」を参照してください。

既存プロジェクトへの配置結果のインポート

PlanAhead 外で生成した配置結果を既存のプロジェクトにインポートすることもできます。配置制約は配置されたロジック オブジェクトすべてに割り当てられます。

Native Circuit Description (NCD) ファイルを選択すると、ファイルのデータが取り込まれて、Xilinx Definition List (XDL) フォーマットに変換され、配置情報がプロジェクトにインポートされます。NCD ファイルは、ISE ソフトウェアの配置配線アプリケーションから出力されるファイルです。PlanAhead では、NCD ファイルのデータは使用されますが、ファイル自体はプロジェクトに追加されません。

配置結果をインポートするには、次の手順に従います。

1. [File] → [Import] → [Import Placement] をクリックします。[Import Placement] ダイアログボックスが開きます。

[Import Placement] コマンドを使用するには、RTL デザイン、ネットリスト デザイン、またはインプリメント済みデザインが必要です。詳細は、30 ページの「[デザインの操作](#)」を参照してください。

2. [Import Placement] ダイアログ ボックスで NCD ファイルを選択します。

PlanAhead では XDL ユーティリティが自動的に実行されて NCD ファイルが XDL フォーマットに変換され、結果がインポートされます。

注記：XDL ファイルがある場合は、直接インポートすることもできます。

3. [OK] をクリックして配置結果をインポートします。

PlanAhead では配置ファイルがインポートされて、このファイルを使用してインプリメント済みデザインが構築されます。インプリメント済みデザインが開いた状態で、次のセクションに示すようにタイミング解析、消費電力解析、およびタイミング シミュレーションを実行し、FPGA の配置結果を解析できます。

タイミング結果の解析

配置配線されたインプリメント済みデザインのタイミング解析には、ロジック エlement またはデバイス リソース間の実際のパス遅延も含まれます。インプリメント済みデザインのタイミング解析を実行するには、次のいずれかを実行します。

- [Tools] → [Timing] → [Run TRCE] をクリックし、TRCE (Xilinx Timing Reporter And Circuit Evaluator) ツールを起動します。詳細は、「[インプリメント済みデザインでの TRCE の実行](#)」を参照してください。
- Flow Navigator の [Run TRCE] をクリックします。
- [Tools] → [Timing] → [Run Timing] をクリックし、PlanAhead ツールの内部タイミング エンジンを実行します。詳細は、第 7 章の「[タイミング解析の実行](#)」を参照してください。

インプリメント済みデザインでの TRCE の実行

TRCE ツールでは、入力したタイミング制約に基づいて、デザインのスタティック タイミング解析を実行できます。このツールでは、タイミング制約を満たしているかが確認され、入力制約に対してデザインが準拠しているかどうかをリストしたレポート ファイルが出力されます。

ISE のスタンドアロン アプリケーション TRCE は、配置されていないデザイン、配置済みデザイン、または一部分配線されたデザイン、完全に配置配線されたデザインに実行できます。PlanAhead

では、インプリメント済みデザインで TRCE を実行できます。インプリメント済みデザインが PlanAhead で開いている場合は、次の手順で TRCE を起動できます。

- [Tools] → [Timing] → [Run TRCE] をクリックします。
- Flow Navigator の [Run TRCE] をクリックします。

PlanAhead で 381 ページの図 11-5 に示すような [Run TRCE] ダイアログ ボックスが表示されます。

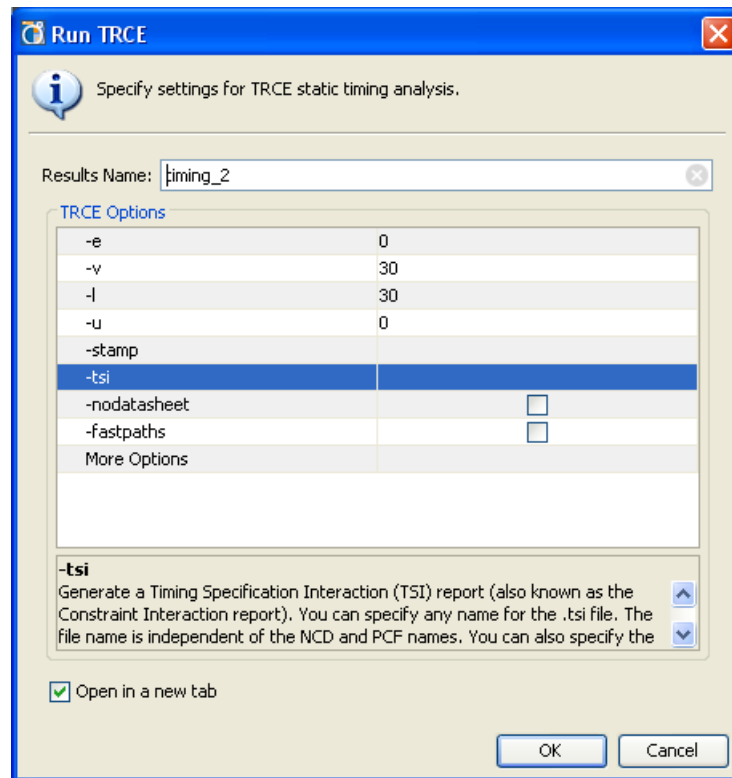


図 11-5 : [Run TRCE] ダイアログ ボックス

- [Results Name] : 出力のタイミング結果の結果名を指定します。デフォルトでは、インプリメンテーション ディレクトリにある上位モジュール名および指定の結果名が組み合わされた名前のファイルに結果が書き込まれます。出力ファイル名は、[More Options] フィールドで -o オプションを指定しても変更できます。

結果は、[Timing Results] ビューに表示される特定の結果名の下に表示されます。詳細は、383 ページの「[Timing Results] ビューの使用」を参照してください。

[TRCE Options] :

- **-e <limit>** : デフォルトのサマリ レポートの代わりにエラー レポートを出力します。<limit> には 0 ~ 32,000 の整数を入力することで、各タイミング制約でさまざまな違反に対してレポートされる数を制限できます。
- **-v <limit>** : 詳細レポートを生成します。オプションで <limit> を使用すると、タイミング制約ごとにレポートされる数を制限できます。有効な値は 1 ~ 32,000 の整数値で、デフォルト値は 3 です。
- **-l <limit>** : レポート ファイルでタイミング制約ごとにレポートされる数を制限します。<limit> には 0 ~ 2,000,000,000 を入力でき、デフォルト値は 3 です。

- **-u <limit>** : 制約が付いていないパスの遅延をレポートします。<limit> には 0 ～ 2,000,000,000 を入力できます。
-u オプションを使用すると、デザインに含まれる制約が付いていないパスがタイミング解析に含まれるよう、これらすべてのパスに制約が追加されます。制約が適用されていないパスに対して、デフォルトのパスがリストされます。デフォルトのパスには、順次コンポーネントのデータおよびクロック ピンへの回路パスや、プライマリ出力のデータ ピンへの回路パスなどが含まれます。
- **-stamp <filename>** : デザインのタイミングを示す STAMP タイミング モデル ファイル ペア (<filename>.mod および <filename>.data) を生成します。
- **-tsi <filename.tsi>** : タイミング制約間の相互関係を示す TSI (Timing Specification Interaction) レポートを生成します。TSI レポートでは、複数のタイミング制約が同じパスに付けられている場合の優先度を確認できます。
- **-nodatasheet** : 標準のタイミング レポートに含まれるデータシート セクションを含めないようにします。
- **-fastpaths** : デザインの高速パスをレポートします。
- **More Options** : 特定の TRCE オプションを適用します。特定の TRCE コマンドの詳細は、付録 E「その他のリソース」に示す『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。
追加オプションの例を次に示します。

- **-o <file_name>** : 出力されるタイミング レポートの名前を指定します。拡張子 .twr は省略できます。
- **-s <speed_grade>** : デザインに含まれるデバイスのスピードの代わりに、このオプションで指定したデバイスのスピードを解析に使用します。このオプションを使用することで、タイミング要件に合うスピード グレードを調べることができます。入力できる値は、1、2、または 3 のいずれかで、ダッシュ記号 (-) は先頭に付けても付けなくても構いません。
- **Open in a new tab** : 実行後のタイミング レポートを [Timing Results] ビューの新しいタブに開きます。このオプションがオフのときは、最後に表示されたタイミング結果が閉じてから新しいレポートが開きます。

PlanAhead は、タイミング レポート ファイル (TWX) から情報を抽出し、TRCE のタイミング解析結果を表示します。TWX ファイルがインポートされると、そのタイミング結果が [Timing Results] ビューに表示されます。タイミング制約が使用されていなかった場合は、タイミング結果は表示されません。

既存プロジェクトへの ISE TRCE タイミング結果のインポート

PlanAhead 外で TRCE コマンドを実行して生成した TWR または TWX 形式のタイミング レポート ファイルからタイミング結果をインポートできます。コマンド ラインに入力されるオプションに基づいて、TRCE から次のタイミング レポートが出力されます。

- **TWR** : デフォルトのタイミング レポート
エラー レポートを出力するには -e オプションを、詳細レポートを出力するには -v オプションを使用します。
注記 : PlanAhead には、TWR ファイル形式はインポートできません。
- **TWX** : -xml オプションを使用すると出力される XML タイミング レポート

レポートは、**Timing Analyzer** で開くことができます。**-e** オプションおよび **-v** オプションは、**TWX** ファイルおよび **TWR** ファイルの両方に適用されます。詳細は、「**-xml (XML 出力ファイル名)**」を参照してください。

TRCE タイミング結果を PlanAhead ツールへインポートするには、次の手順に従います。

1. [File] → [Import] → [Import Timing] をクリックします。

[Import Timing] コマンドを使用するには、RTL デザイン、ネットリスト デザイン、またはインプリメント済みデザインが必要です。詳細は、[30 ページの「デザインの手順」](#)を参照してください。

2. [Import Timing] ダイアログ ボックスで、次を入力します。

- **[File Name]** : インポートする TWX ファイルを選択します。
- **[Results Name]** : **[Timing Results]** ビューにレポートされる結果の名前を入力します。

3. [OK] をクリックし、タイミング結果をインポートします。

タイミング結果が PlanAhead 環境内に表示されます。

[Timing Results] ビューの使用

PlanAhead ソフトウェアは、ISE の TRCE プログラムで出力されたタイミング結果をインポートし、[Timing Results] ビューにその結果を表示します。

[Timing Results] ビューには、クロック制約別に分類された **TRCE** タイミングパス情報が表示されます。+ や - マークをクリックすると、タイミングパスのツリー表示を展開したり、閉じたりできます。エラーが発生したパスのスラック値は、赤色で表示されます。図 11-6 は、TRCE 結果を表示しています。

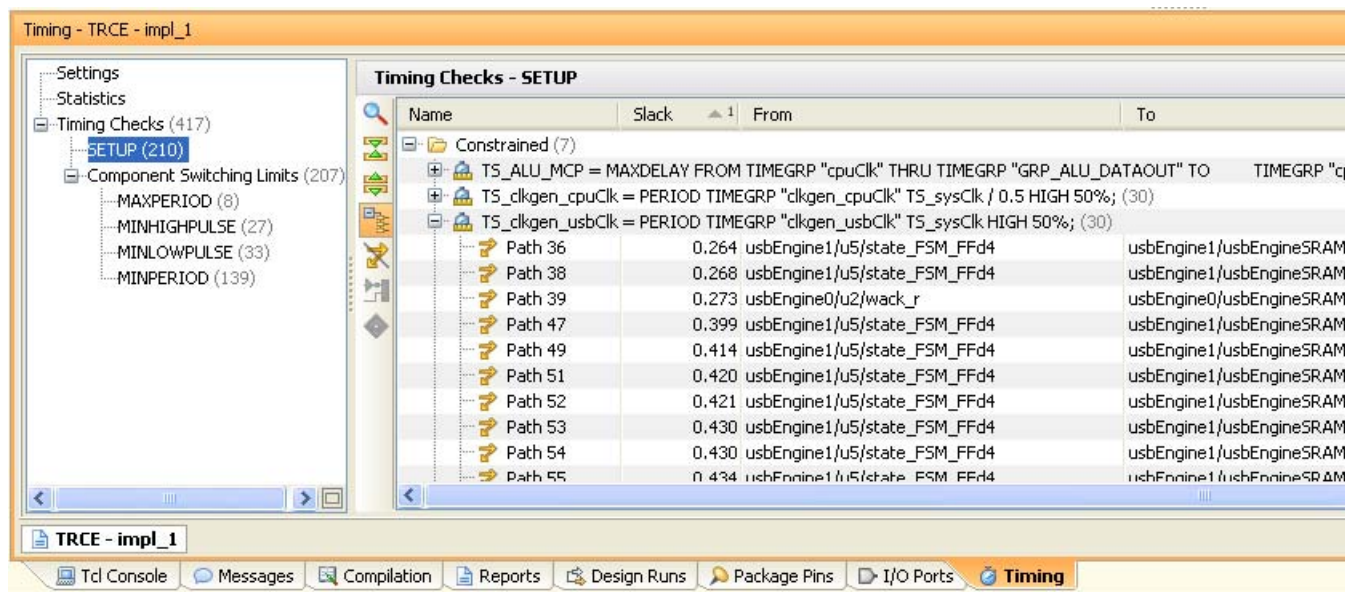


図 11-6 : ISE TRCE タイミング結果

TRCE ソフトウェアから出力されたタイミング結果は、[258 ページの「タイミング結果の解析」](#)で説明されている合成済みデザインで使用できる [Report Timing] コマンドを実行したときの結果と異なります。

TRCE 結果には配置配線されたデザインのタイミング情報がそのまま表示されますが、PlanAhead ではタイミング解析で配線遅延のみが予測されます。

TRCE タイミング レポートを読み込むと、タイミング レポートのネットおよびプリミティブが PlanAhead のネットリスト データベースに関連付けられており、クロスプローブおよび解析を実行できます。タイミング レポートのプリミティブ (BEL) がネットリストのプリミティブに直接関連付けられない場合は、タイミング レポートには表示されず、遅延は PlanAhead で検出されるプリミティブに関連付けられます。TRCE のタイミング遅延は、PlanAhead ソフトウェアのタイミング レポートの累積遅延に保持されます。

注記：ISE ではロジックを変更して、配置配線結果を改善できます。この場合、元のネットリストのロジックは削除されるか置き換えられる可能性があります。この結果、PlanAhead で開いているインプリメンテーション前のネットリストと TRCE のタイミング結果が一致しくなくなります。このような不一致は FPGA のインプリメンテーションには影響がありませんが、インプリメント済みデザインを開いたときに Tcl コンソールでレポートされます。

[Timing Results] ビューでは、パスやインスタンスを確認、並べ替え、選択できます。パスを選択すると [Device] ビューに表示され、そのパスのプロパティが [Path Properties] ビューに表示されます。

[Timing Results] ビューでは、パスやインスタンスを確認、並べ替え、選択できます。次の情報が表示されます。

- [Settings] : TRCE コマンド ラインのサマリとタイミング解析が実行されたときに使用されたオプションが表示されます。
- [Statistics] : タイミング結果の概要が表示されます。
- [Timing Checks] : 実行したチェックがタイプ別にグループ分けされ、デフォルトでは計算したスラックでタイミング パスを分類したタイミング結果の表形式が表示されます。
 - [Name] : 連続した番号が表示され、元の順序に並べ替えるときに使用できます。
 - [Slack] : パスの正または負のスラックの合計が表示されます。
 - [From] : パスのソース ピンが表示されます。
 - [To] : パスのデスティネーション ピンが表示されます。
 - [Total Delay] : パスで予測される遅延の合計が表示されます。
 - [Logic Delay] : パスのロジック エLEMENT の配置が起因の遅延のみが表示されます。
 - [Net Delay] : パスのインターコネクトの配置が起因の遅延のみが表示されます。
 - [Logic %] : ロジック エLEMENT の配置が起因の遅延の割合が表示されます。
 - [Net %] : インターコネクトの配置が起因の遅延の割合が表示されます。
 - [Stages] : ソースとデスティネーションが全体の遅延に影響を及ぼすパス上のインスタンスの合計が表示されます。
 - レポートされるステージ数は、ISE でレポートされるロジック レベルとは異なることがあります。
 - [Source Clock] : ソース クロックの名前が表示されます。
 - [Destination Clock] : デスティネーション クロックの名前が表示されます。

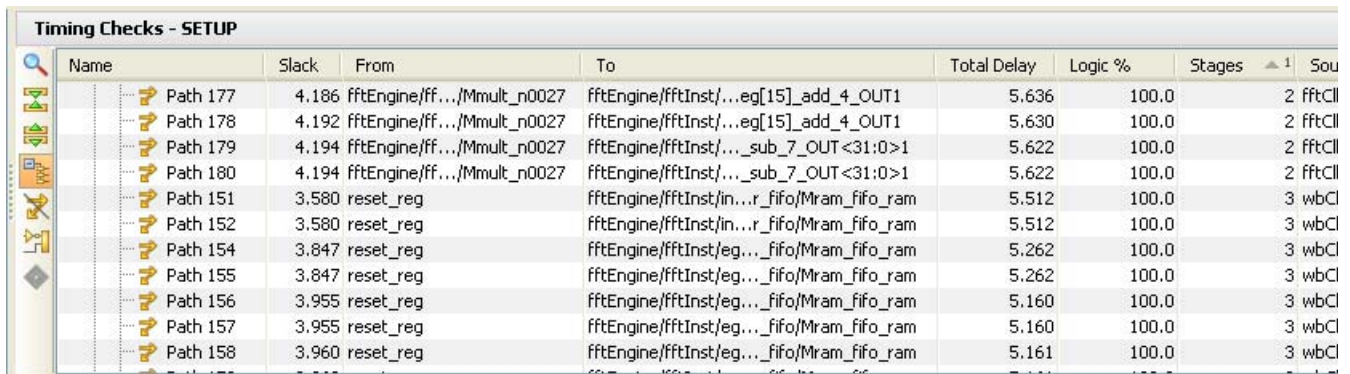
タイミング レポートの並び替え

[Timing Results] ビューのリストは、列ヘッダーをクリックすると並べ替えることができます。たとえば、[Stages] 列のヘッダーをクリックすると、ロジックのステージ順に並べ替えられます。列をもう一度クリックすると、逆の順序で並べ替えることができます。

並び替えた後に **Ctrl** キーを押して別の列ヘッダーをクリックすると、さらにその列に基づいて並び替えられます。並び替えに使用する列の数に制限はありません。もう一度 **Ctrl** キーを押して列ヘッダーをクリックすると、その列の並び替えが解除されます。

ツリー表形式のビューの詳細は、[127 ページの「ツリー表形式のビュー」](#)を参照してください。

図 11-7 は、ロジックのステージ別に並び替えた [Timing Results] ビューを示しています。



Name	Slack	From	To	Total Delay	Logic %	Stages	Sou
Path 177	4.186	fftEngine/ff.../Mmult_n0027	fftEngine/fftInst/...eg[15]_add_4_OUT1	5.636	100.0	2	fftCl
Path 178	4.192	fftEngine/ff.../Mmult_n0027	fftEngine/fftInst/...eg[15]_add_4_OUT1	5.630	100.0	2	fftCl
Path 179	4.194	fftEngine/ff.../Mmult_n0027	fftEngine/fftInst/..._sub_7_OUT<31:0>1	5.622	100.0	2	fftCl
Path 180	4.194	fftEngine/ff.../Mmult_n0027	fftEngine/fftInst/..._sub_7_OUT<31:0>1	5.622	100.0	2	fftCl
Path 151	3.580	reset_reg	fftEngine/fftInst/in...r_fifo/Mram_fifo_ram	5.512	100.0	3	wbCl
Path 152	3.580	reset_reg	fftEngine/fftInst/in...r_fifo/Mram_fifo_ram	5.512	100.0	3	wbCl
Path 154	3.847	reset_reg	fftEngine/fftInst/eg..._fifo/Mram_fifo_ram	5.262	100.0	3	wbCl
Path 155	3.847	reset_reg	fftEngine/fftInst/eg..._fifo/Mram_fifo_ram	5.262	100.0	3	wbCl
Path 156	3.955	reset_reg	fftEngine/fftInst/eg..._fifo/Mram_fifo_ram	5.160	100.0	3	wbCl
Path 157	3.955	reset_reg	fftEngine/fftInst/eg..._fifo/Mram_fifo_ram	5.160	100.0	3	wbCl
Path 158	3.960	reset_reg	fftEngine/fftInst/eg..._fifo/Mram_fifo_ram	5.161	100.0	3	wbCl

図 11-7 : 並び替えた [Timing Results] ビューと [Group by Constraint] ボタン

パス リストのフラット化

デフォルトでは、パスは制約別に表示されます。[385 ページの図 11-7](#) に示すように、[Timing Results] ビューのツールバーで [Group by Constraint] をクリックすると、リストをフラットにし、すべてのパスを表示できます。[Group by Constraint] ボタンでは、制約別に分類されたパスのリストとフラット化されたパスのリストの表示を切り替えることができます。

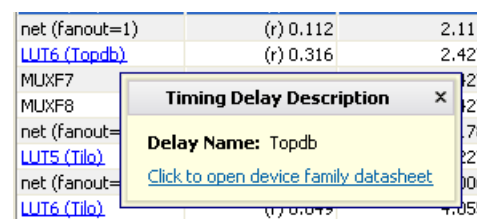


[Path Properties] ビューの使用

[Path Properties] ビューにパスのロジックおよび遅延情報を表示するには、[Timing Results] ビューでそのパスをクリックします。

TRCE のタイミング結果は、[258 ページの「タイミング結果の解析」](#)で説明されている PlanAhead の [Report Timing] コマンドのタイミング結果とは異なります。TRCE レポートには、[図 11-8](#) のようにクロック スキューやジッターなどの別の情報も表示されます。

[Delay Type] 列のリンクをクリックすると、その FPGA デバイスのデータシートが表示され、選択したロジック サイト オブジェクトの PDF ファイルが自動的に検索されます。



net (fanout=1)	(r) 0.112	2.11
LUT6 (Topdb)	(r) 0.316	2.42
MUXF7		
MUXF8		
net (fanout=1)		
LUT5 (Tilo)		
net (fanout=1)		
LUT6 (Tilo)		

Timing Delay Description

Delay Name: Topdb

[Click to open device family datasheet](#)

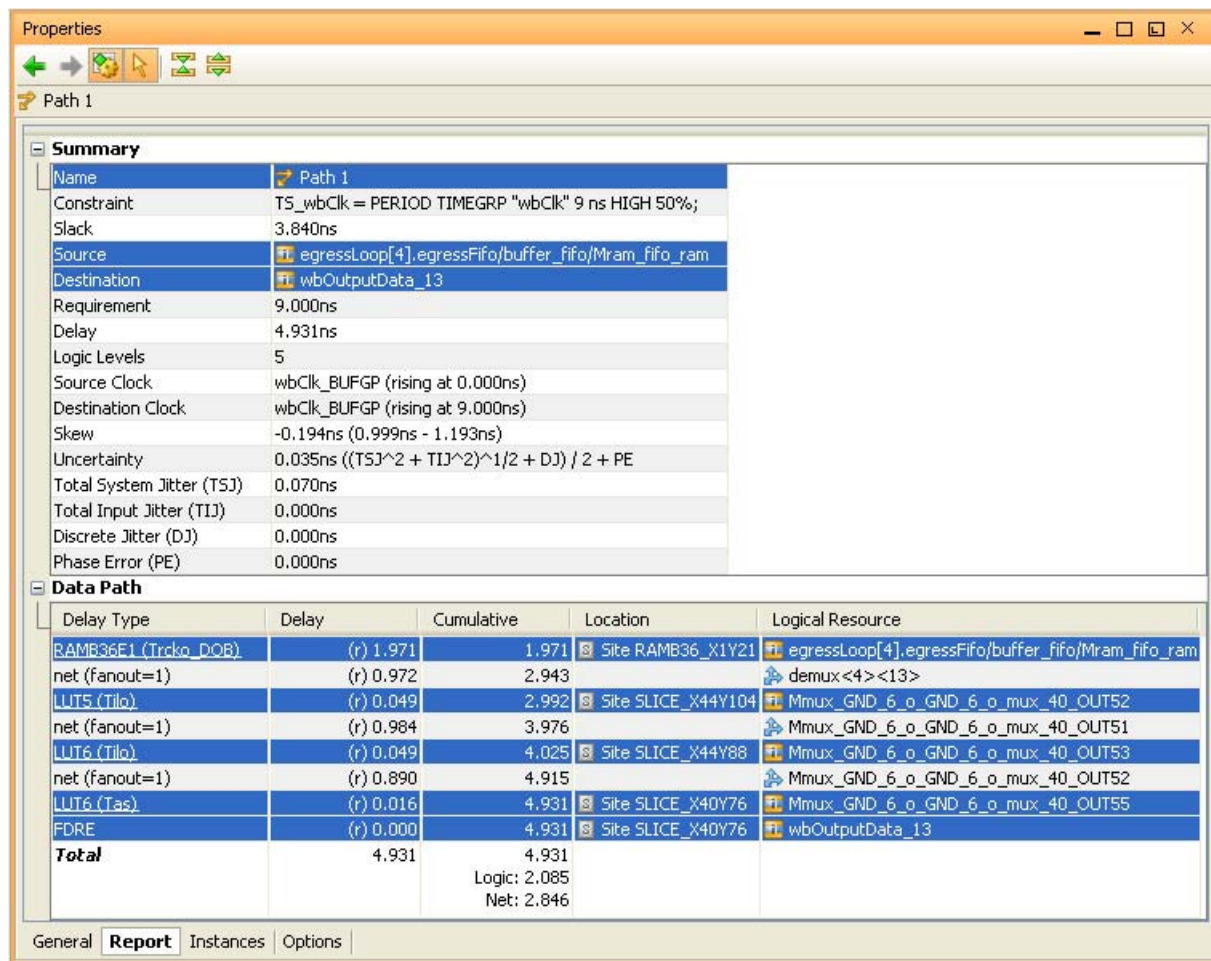


図 11-8 : [Path Properties] ビュー

- デフォルトでは、パスを選択するとそのパスに含まれるインスタンスもすべて選択されます。
- レポートにリンクが表示されているオブジェクトをクリックすると、[Netlist] や [Device] など、ほかのビューでもそのオブジェクトが選択されます。
- Shift キーまたは Ctrl キーを使用すると、複数のパスを選択できます。
- 選択されたパスすべてのインスタンスすべてが選択されますが、[Path Properties] ビューには最初に選択されたパスの情報のみが表示されます。

また、[Timing Results] ビューでパスを選択して右クリックし [View Path Report] をクリックするか、パスをダブルクリックすると、パス データをワークスペースに大きく表示することもできます。

[Timing Results] ビューと [Path Properties] ビューを使用したタイミング結果の解析については、第 7 章「合成済みデザインの制約および解析」を参照してください。

[Device] ビューでのタイミング パスの表示

[Timing Results] ビューでパスの行を選択すると、タイミング パスを [Device] ビューに表示できます。パスは [Device] ビューでハイライトされます。複数のパスを選択すると、パスのインスタンスもすべて選択され、ハイライトされます。図 11-9 は、[Device] ビューでタイミング パスがハイライトされたところを示しています。

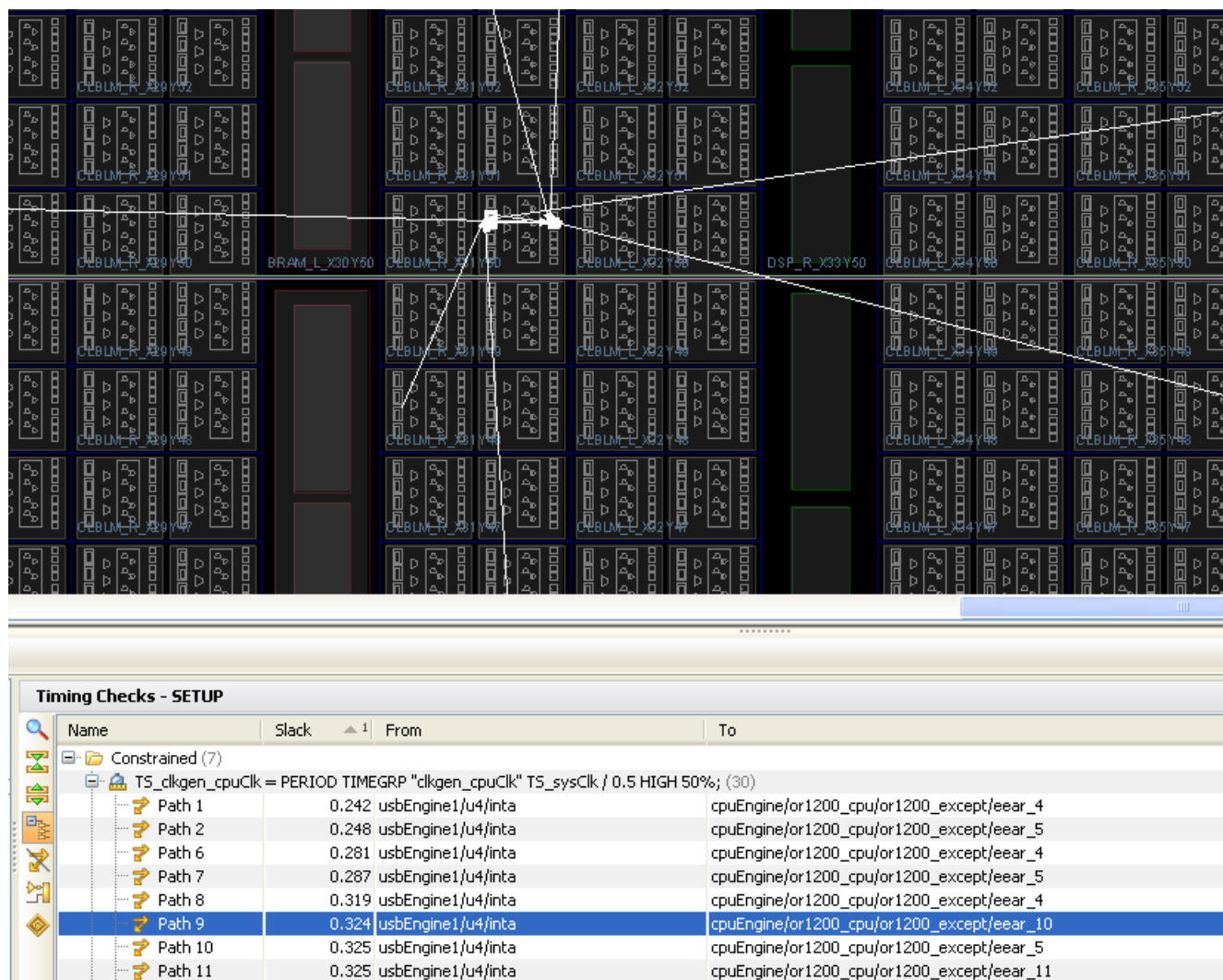


図 11-9 : [Device] ビューでハイライトされたタイミング パス

[Schematic] ビューでのタイミング パスの表示

[Timing Results] のツールバーまたはポップアップ メニューで [Schematic] をクリックすると、PlanAhead に [Schematic] ビューが表示され、選択されたパスに含まれるインスタンスが表示されます。図 11-10 のように、[Schematic] ビューにはインスタンスがはっきりと表示されるだけでなく、階層モジュールも表示されます。

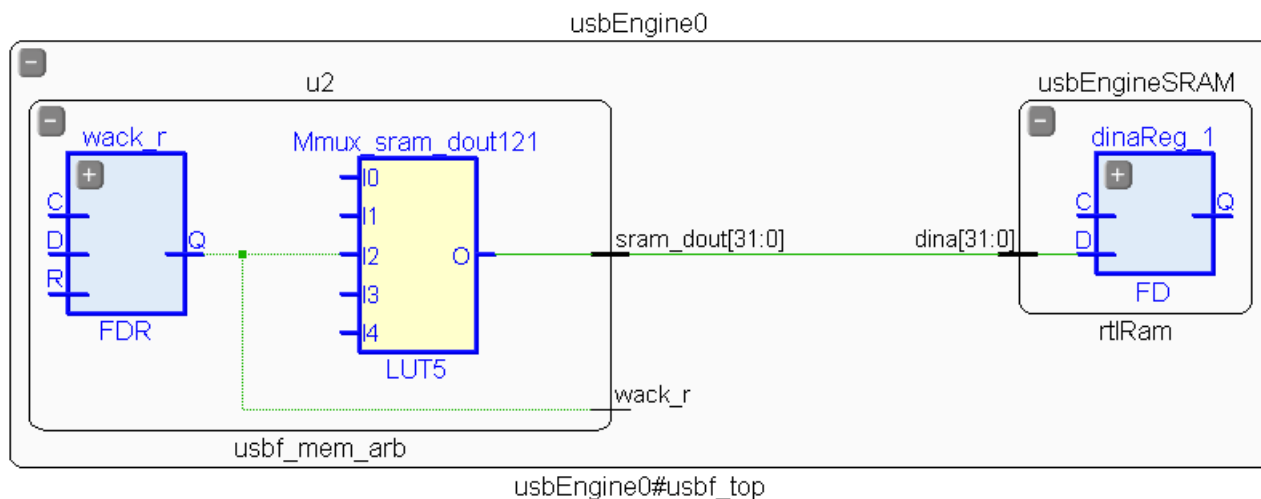


図 11-10：[Schematic] ビューに表示されたタイミング パス

[Schematic] ビューでタイミング パスが表示されるときは、オブジェクトがすべて表示されます。個々のロジック インスタンスの [Schematic] ビューには、選択したインスタンスだけが表示されます。

パスのグループに含まれるインスタンスはすべてこの方法で表示されるため、フロアプランでどのモジュールを一緒にグループにすべきかがわかりやすくなっています。[Schematic] ビューのポップアップ メニューの [Pblock creation] コマンドを使用すると、[Device] ビューで簡単に Pblock への割り当てができます。[Schematic] ビューの詳細は、163 ページの「[Schematic] ビュー」を参照してください。

ロジック接続の表示

次のセクションでは、PlanAhead のロジック接続オプションについて説明します。

[Show Connectivity] コマンドの使用

[Show Connectivity] コマンドを使用すると、選択したエレメントに接続されたネットがすべてハイライトされます。このコマンドは次の方法で使います。

1. [Netlist] ビューまたは [Schematic] ビューから 1 つまたは複数のプリミティブ、[Device] ビューから配置済みロジック エレメント、[Device] ビューまたは [Physical Properties] ビューから Pblock インスタンスを選択します。または、これらを組み合わせたオブジェクトを選択します。
2. 右クリックし、[Show Connectivity] をクリックします。

たとえば、[Schematic] ビューでインスタンスまたは Pblock を選択した場合は、389 ページの図 11-11 のように、[Device] ビューでこのエレメントに接続したネットがすべてハイライトされます。

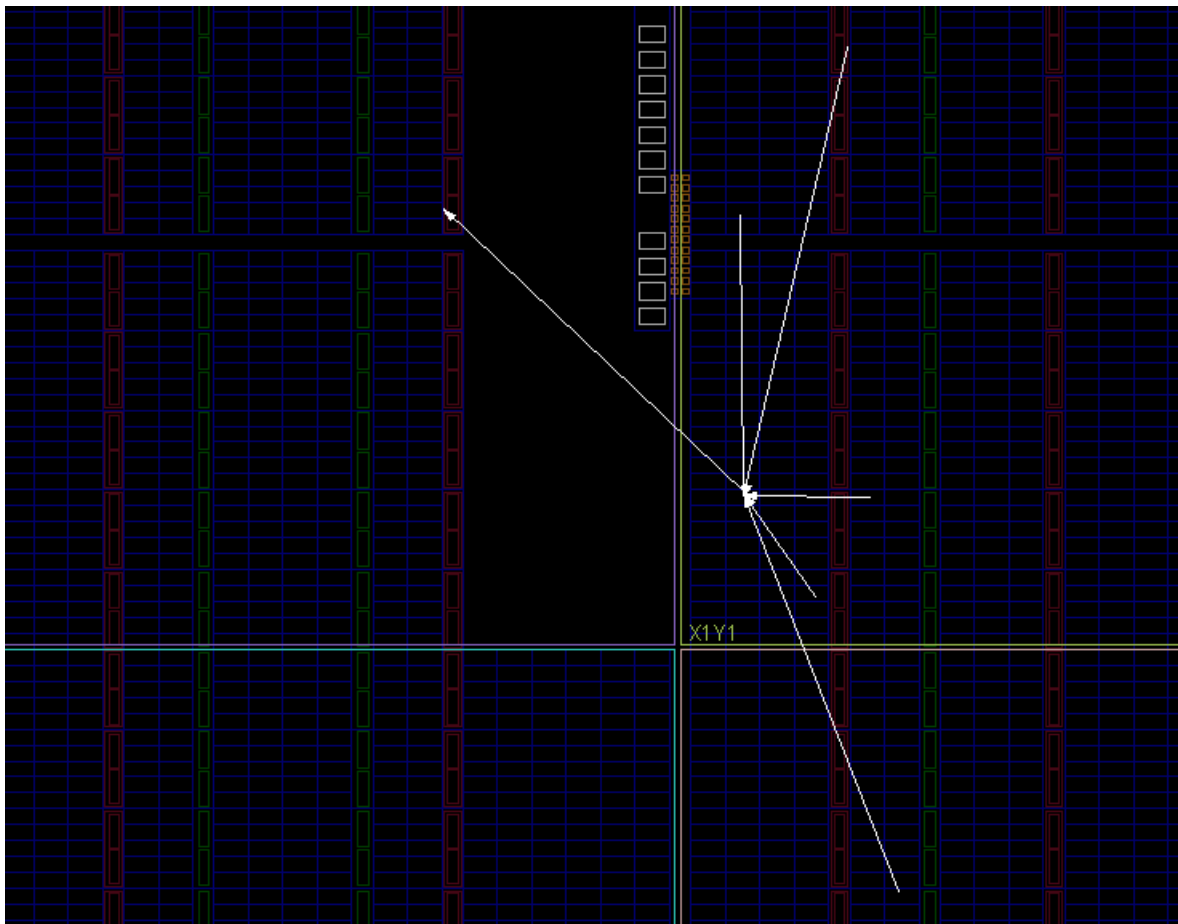


図 11-11 : [Device] ビューに表示されたネットの接続状況

[Show Connectivity] を実行したロジック接続の表示

[Show Instance Connections] ツールバー ボタンをオンにすると、選択したオブジェクトの接続が表示できます。

このモードがオンであれば、別のロジック オブジェクトの接続が表示できます。ツールバー ボタンをクリックすると、オン/オフが切り替わります。



ロジック ファンアウトの展開表示および選択

[Show Connectivity] コマンドを連続して実行すると、ロジック ファンアウトを順番に展開表示して選択できます。

1. ネットリスト、Pblock インスタンス、またはこれらの組み合わせを選択します。
2. 右クリックし、[Show Connectivity] をクリックします。選択したエレメントに接続されたネットがすべてハイライトされます。
3. 右クリックし、[Show Connectivity] をもう 1 度クリックします。ネットに接続されたインスタンスのセットが選択されます。Ctrl + T ショートカット キーを使用しても選択されます。
4. 右クリックし、[Show Connectivity] をもう 1 度クリックします。選択されたインスタンスに接続された、次のレベルのネットがハイライトされます。

この方法を使用すると、特定のインスタンスまたは I/O ポートまでロジックのファンアウトを簡単に展開表示して選択できます。

[Schematic] ビューでのロジックの展開表示

[Schematic] ビューを使用すると、ロジックを階層全体をトレースできます。[Schematic] ビューで選択されたものは [Device] ビューでもハイライトされます。

トレースするインスタンスのピンをダブルクリックすると、信号を展開表示できます。図 11-12 は、[Schematic] ビューで展開されたロジックの例を示しています。

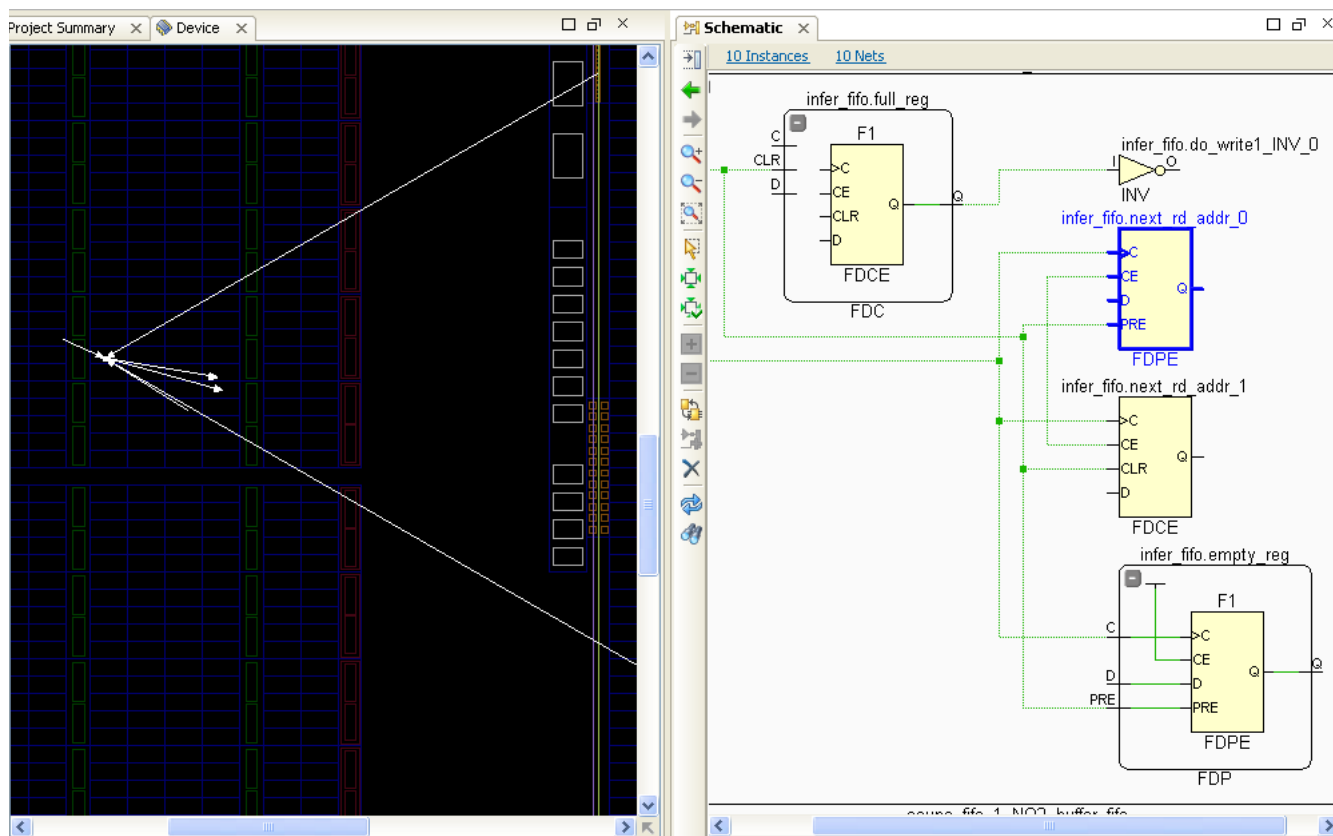


図 11-12 : [Schematic] ビューで展開表示されたロジック

インスタンスおよびモジュールの接続、その内容もインタラクティブに展開表示できます。

[Schematic] ウィンドウでのロジック表示の詳細は163 ページの「[Schematic] ビュー」を参照してください。

[Find] コマンドを使用したオブジェクトの検索

配置が [Device] ビューに表示されたら、[Find] コマンドを使用して特定タイプのロジックを検索できます。[Edit] → [Find] コマンドで表示されるダイアログ ボックスには、さまざまな方法で検索項目をフィルターする柔軟な機能があります。

ロジック オブジェクトの検索の詳細は、140 ページの「ソース ファイルでの検索および置換」を参照してください。

[Select Primitives] および [Highlight Primitives] コマンドの使用

Pblock やロジック モジュールを右クリックして [Select Primitives] をクリックすると、インプリメント済みデザインで下位のプリミティブ ロジック エレメントが選択できます。このコマンドは、配置済みインスタンスの処理をするために、よく [Show Connectivity]、[Fix Instances]、[Clear Placement] コマンドと併用されます。選択したインスタンスは、ポップアップ メニューから [Highlight] コマンドをクリックしてもハイライトできます。

Pblock やロジック モジュールを選択し、ポップアップ メニューの [Highlight Primitives] コマンドでこれらのブロック内のプリミティブをハイライトし、配置したインスタンスをハイライトする色を選択します。複数の Pblock またはロジック モジュールを選択した場合は、[Highlight Primitives] → [Cycle Colors] コマンドで各 Pblock またはモジュールごとに色を選択できます。これにより、ロジックの異なるブロックを色分けして見やすくすることができます。

図 11-13 のように、[Netlist] ビューのモジュールおよびプリミティブのハイライト色が、[Device]、[Schematic]、および [Package] ビューで反映されます。

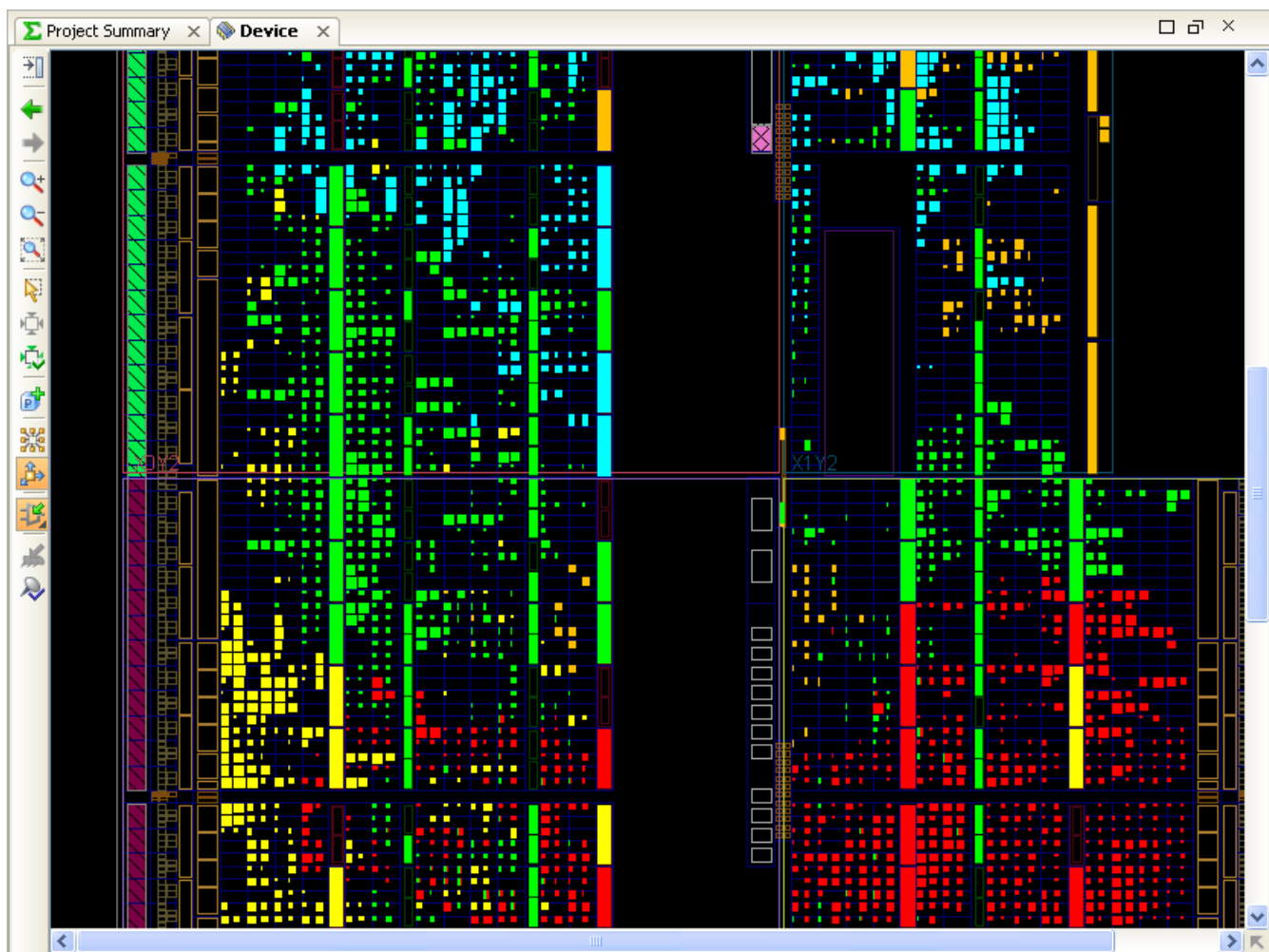


図 11-13 : [Netlist] と [Device] ビューのハイライト色の一致

オブジェクトのハイライト解除

オブジェクトのハイライトを解除するには、次のいずれかのコマンドを実行します。

- [View] → [Unhighlight All] をクリックすると、すべてのオブジェクトのハイライトが解除されます。
- [View] → [Unhighlight Color] をクリックすると、指定した色のハイライトが解除されます。
- [Unhighlight All] ツールバー ボタンをクリックします。



ワークスペース ビューでのオブジェクト選択方法の設定

[Tools] → [Options] をクリックすると、[PlanAhead Options] ダイアログ ボックスでオブジェクトの選択方法を設定できます。オブジェクトの選択方法の設定については、[191 ページの「表示テーマのカスタマイズ」](#)を参照してください。

選択したオブジェクトのハイライト

オブジェクトを指定した色でハイライト表示できます。ハイライト表示は、フロアプランのハイライト表示をすべて解除するまで保持されます。ハイライトの詳細は、[136 ページの「選択したオブジェクトのハイライト」](#)を参照してください。

選択したオブジェクトのマークおよびマーク削除

選択したすべてのオブジェクトにマーク シンボルを付けることができます。マークを付けるオブジェクトを選択して、次のいずれかを実行します。

- [View] → [Mark]
- 右クリックして [Mark]



すべてのマークを削除するには、ツールバーの [Unmark All] ボタンをクリックします。



今後のインプリメンテーション run で使用する配置のロック

PlanAhead で配置したインスタンスは、固定または固定しないようにできます。設計者の配置したインスタンスは **fixed** (固定)、インプリメンテーション中にツールで自動的に配置されたインスタンスは **unfixed** (未固定) となり、それぞれ別の色で表示されます。[図 11-14](#) では、固定されたインスタンスがオレンジで、固定されていないインスタンスは青で表示されています。

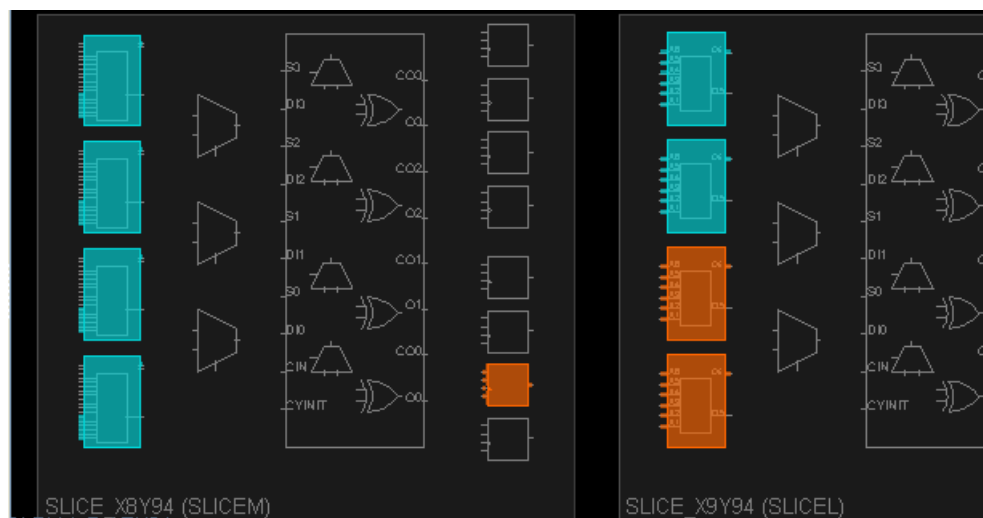


図 11-14：固定および未固定のインスタンス

続くインプリメンテーション run 用に配置の位置をロックするには、ロジックを選択して [Fix Instances] をクリックします。

この機能を使用すると、インプリメンテーション結果に一貫性を持たせることができます。インプリメント済みデザインを保存すると、UCF ファイルの配置済みロジックに LOC 制約および BEL 制約が含まれ、この設定をその後のインプリメンテーション run で使用できます。固定されたロジックは、インプリメンテーションプロセス中に移動されず、未固定のロジックはデザインの最適化および反復の必要に応じて移動されます。詳細は、[365 ページの「固定された配置制約および固定されていない配置制約について」](#)を参照してください。

特定タイプのロジックのロック

インプリメンテーション結果の一貫性を改善するには、ブロック RAM および DSP などの一部またはすべてのブロック マクロ ロジックをロックします。PlanAhead で知識を生かして手動でロックすることもできますが、問題のなかった ISE のインプリメンテーション結果を再利用してロックすることもできます。この方法を使用すると、デザインにブロック RAM または DSP などが多く含まれる場合、より一貫した結果になり、ランタイムも改善できます。

特定のロジック タイプを固定するには、[Find] コマンドを使用し、ブロック RAM および DSP などの特定ロジック タイプを選択します。次に [Find Results] ビューでそれらを選択し、[Fix Instances] を使用します。

ロジック モジュールのロック

インプリメンテーション結果の一貫性を改善するには、クリティカル ロジックをロックします。ロックする必要があるのは、特定ロジック、タイミング パス、またはロジック モジュール全体です。

特定モジュールですべてのロジックを固定（配置）するには、モジュールを選択して [Select Primitives] コマンドを実行し、そのロジック モジュールに関連するプリミティブ ロジック インスタンスすべてを選択します。ロジックをロックするには、[Fix Instances] コマンドを使用します。

デザイン メトリックの表示

次のセクションでは、デザイン メトリック オプションについて説明します。

[Metrics] ビューの使用

PlanAhead の [Metrics] ビューでは、デザイン メトリックのリストが示され、デザインで問題が発生する可能性があるエリアが定義された色で表示されます。現時点のメトリックでは、Pblock および配置済みデザインレベルでの使用率、配線の混線、およびタイミングチェックが含まれています。

[Metrics] ビューを開くには、[Window] → [Metrics] をクリックします。394 ページの図 11-15 に、[Metrics] ビューを示します。

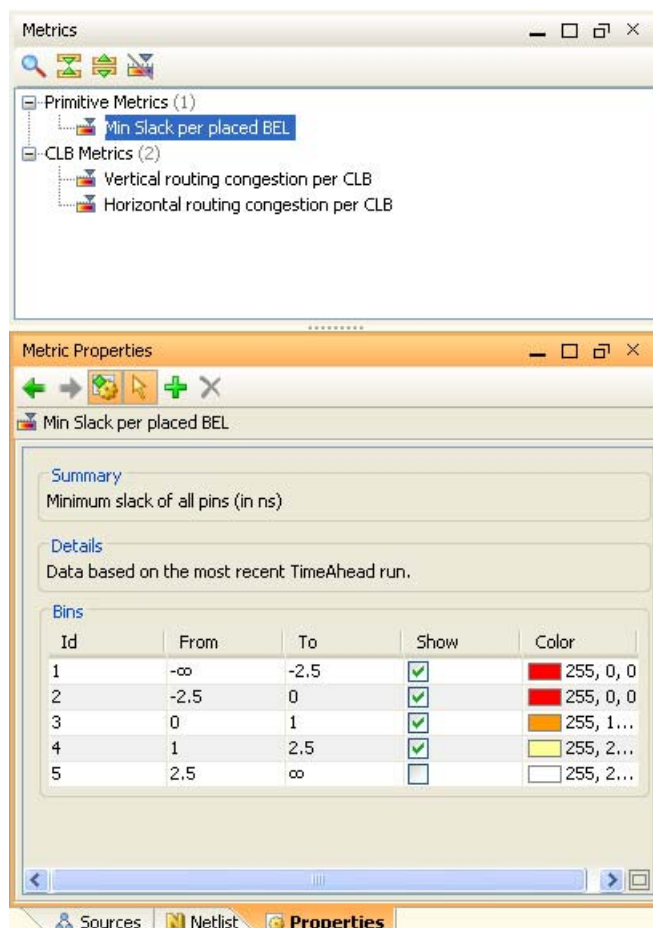


図 11-15 : [Metrics] ビュー

[Metric Properties] ビューには、図 11-15 に示すように、選択したメトリックの機能の説明とエラーが発生する可能性がある箇所をハイライトするために定義されたメトリック範囲 (Bin) が表示されます。

メトリック範囲の設定

[Metric Properties] ビューでは、メトリック範囲 (Bin) を設定できます。

特定の Bin に使用される色を変更するには、[Color] 列をクリックします。よく使用される色を選択することもできますが、[More Colors] をクリックするとさらに多くの色から選択できます。または、特定の色の RGB 値を入力することもできます。

Bin の表示のオン/オフを切り替えるには、[Show] チェック ボックスを使用します。

表示される範囲数を減らすには、[Delete] をクリックします。値の範囲は [Metrics Properties] ビューのその Bin の上に統合されます。



ツールバーまたはポップアップ メニューから [Insert Bin] をクリックすると、新しい値の範囲を追加し、表示されているメトリックの粒度を改善できます。395 ページの図 11-16 は、定義された新しい Bin を表示しています。Bin の値の範囲を指定したり、使用する色を指定したりできます。既存の Bin の範囲は新しく定義した Bin に合わせて調整されます。



注記：PlanAhead ツールでは、外側の Bin が最後にユーザーが定義した値から無限大または負の無限大までの範囲に自動的に調整されます。

[Metric Properties] ビューのツールバー ボタンまたはポップアップ メニューから [Apply changes] をクリックし、メトリック プロパティに加えた変更を適用します。

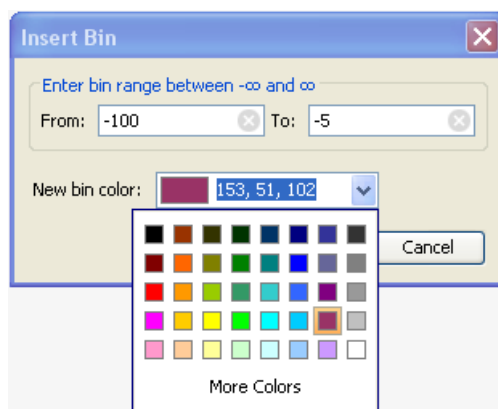


図 11-16 : [Insert Bin] ダイアログ ボックス

[Device] ビューでのメトリック マップの表示

[Device] ビューでメトリック マップを表示するには、[Metrics] ビューでメトリックを右クリックし、[Show] を選択します。これで、カラー ベースのメトリック マップが表示されます。図 11-17 は、垂直方向に配線混雑がある場合のメトリック マップを示しています。

複数のメトリック マップを同時に表示することも可能です。

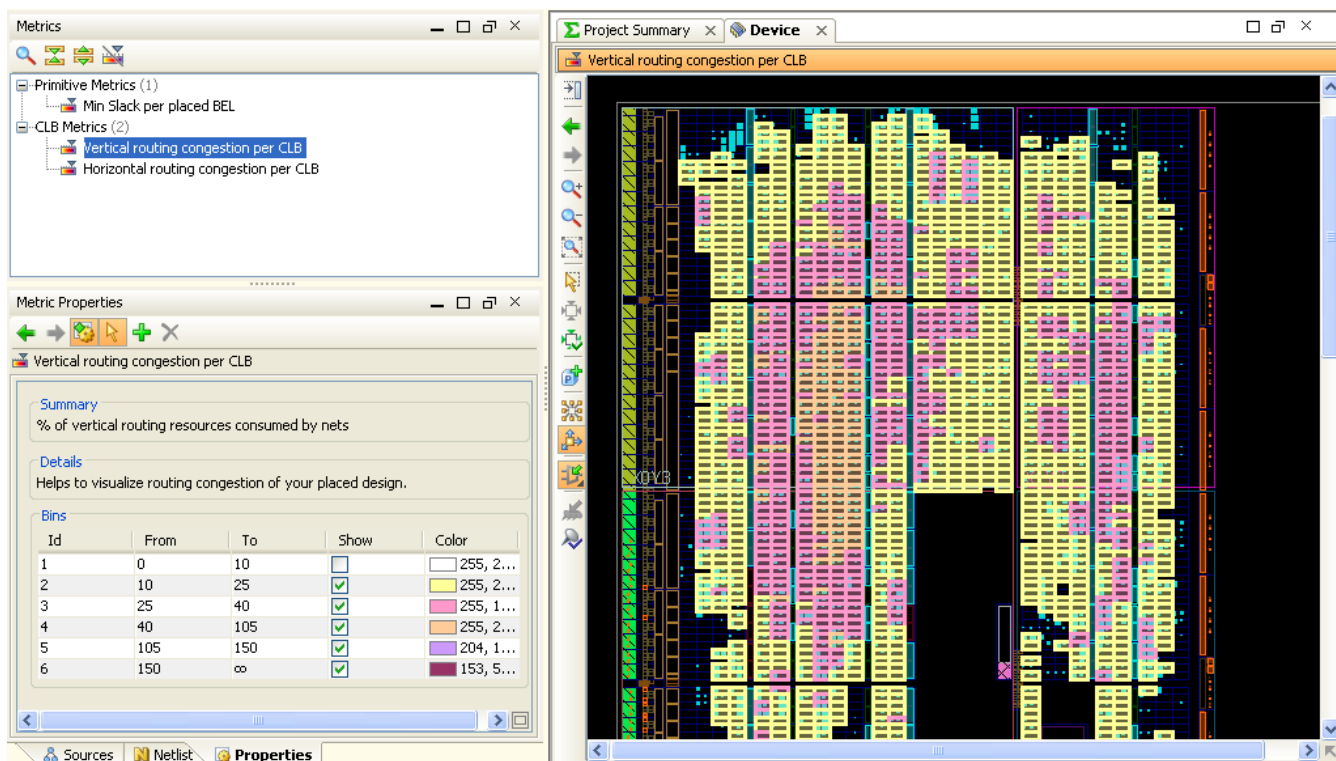


図 11-17 : [Device] ビューでのメトリック マップの表示

メトリック マップ表示の解除

[Device] ビューのメトリック マップを非表示にするには、右クリックで [Hide] または [Hide All Metrics] を選択します。

[Metric Results] ビューの使用

[Show] コマンドを実行すると、メトリックの結果が [Metric Results] ビューに表示されます。[Metric Results] ビューでは、次の操作を実行できます。

- 列のヘッダーをクリックすると、情報を並べ替えられます。
- 最初に並べ替えた後で **Ctrl** キーを押して別の列ヘッダーをクリックすると、その列でさらに並べ替えられます。
- リストの表示順を整えるために、さまざまな並び替え条件を選択できます。

Pblock のメトリック結果は、Pblock が変更されると自動的に更新されます。Pblock、CLB、プリミティブなどの別のタイプのメトリックは、別の表で表示されます。それぞれのタイプは、図 11-18 のように [Metric Results] ビュー内の個別のタブに表示されます。

Metric Results - CLBs (5125)

	Id	Name	Type	Row	Col	Sites	Instances	Vert route...	Hori route...
	409	CLBLM_L_X16Y37	CLBLM_L	169	44	2	16	40.06	31.61
	410	CLBLM_L_X38Y54	CLBLM_L	151	99	2	13	40.05	41.02
	411	CLBLM_R_X15Y176	CLBLM_R	24	43	2	16	40.05	36.34
	412	CLBLM_L_X12Y180	CLBLM_L	20	34	2	13	40.00	32.01
	413	CLBLM_R_X7Y28	CLBLM_R	178	23	2	16	39.99	29.98
	414	CLBLM_L_X40Y50	CLBLM_L	155	103	2	21	39.99	29.21
	415	CLBLM_L_X10Y157	CLBLM_L	44	30	2	8	39.96	24.11
	416	CLBLM_R_X11Y159	CLBLM_R	42	33	2	1	39.95	22.29
	417	CLBLM_L_X12Y177	CLBLM_L	23	34	2	14	39.95	26.71
	418	CLBLM_L_X12Y45	CLBLM_L	161	34	2	5	39.93	43.44
	419	CLBLM_L_X32Y59	CLBLM_L	146	83	2	11	39.92	39.91
	420	CLBLM_R_X5Y41	CLBLM_R	165	17	2	13	39.92	23.67
	421	CLBLM_L_X18Y39	CLBLM_L	167	50	2	20	39.90	49.19

CLBs (5125) x

Tcl Console Messages Compilation Reports Design Runs Timing Metric Results

図 11-18 : [Metric Results] ビュー

タイミング シミュレーションの実行

インプリメント済み FPGA デザインでタイミング シミュレーションを実行するために、PlanAhead はまず NetGen ユーティリティを起動して、内部データベースを Verilog または VHDL ネットリストに変換します。NetGen とは、ザイリンクス デザイン ファイルを入力ファイルとして読み込んで、サードパーティ シミュレーターおよびタイミング解析ツールで使用可能な Verilog または VHDL ネットリストを生成するコマンド ライン実行ファイルのことです。NetGen の詳細は、[付録 E「その他のリソース」](#)に示す『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。

PlanAhead ツールは、ISE シミュレーション ツールである ISim と連動しています。ザイリンクスの ISim は、VHDL、Verilog、VHDL/Verilog 混合デザインのビヘイビア シミュレーションおよびタイミング シミュレーションを実行するハードウェア記述言語 (HDL) シミュレータです。

PlanAhead ツールは、ビヘイビアまたはタイミング シミュレーション用に Mentor Graphics® ModelSim または Questa® Advanced Simulator ツールとも連動します。ターゲット シミュレーターの設定方法については、[第 3 章の「プロジェクト設定」](#)を参照してください。

タイミング シミュレーションは、次のいずれかの手順で実行できます。

- [Implemented Design] で [Tools] → [Simulation] → [Run Timing Simulation] をクリックします。
- Flow Navigator で [Implementation] → [Run Timing Simulation] をクリックします。

PlanAhead で [図 11-19](#) に示すような [Launch Timing Simulation] ダイアログ ボックスが表示されます。

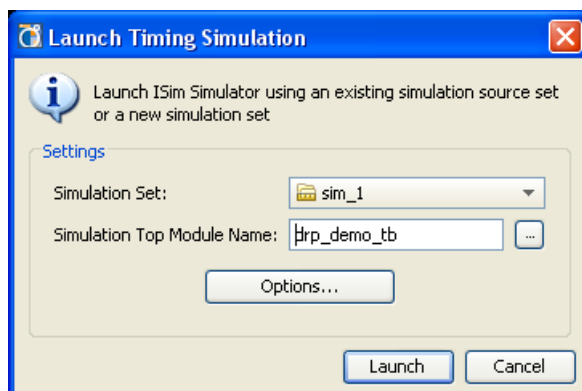


図 11-19 : [Launch Timing Simulation] ダイアログ ボックス

含まれるフィールドは、次のとおりです。

- [Simulation Set] : シミュレーション run の名前を指定します。異なるデザイン階層および異なるオプションを使用して、複数のシミュレーション run を作成できます。
- [Simulation Top Module Name] : デザインの最上位モジュールを指定します。このフィールドには定義済みの最上位モジュールが自動的に入力されますが、別の階層レベルにある別のモジュールを指定したり、さまざまなデザイン パターンを生成したりすることもできます。
- [Options] : ISim ツールのランタイム オプションを定義する [Simulation Options] ダイアログ ボックスを開きます。詳細は、「[シミュレーション オプションの設定](#)」を参照してください。
- [Launch] : ISim でコンパイルおよびエラボレーションを実行し、ISim を GUI で開きます。
- [Cancel] : ISim を起動せずにダイアログ ボックスを閉じます。

シミュレーション オプションの設定

[Options] ボタンをクリックすると、[Simulation Options] ダイアログ ボックスが開きます。このダイアログ ボックスには、前に指定した最上位モジュールが含まれ、次の 3 つのオプション タブが表示されます。

[Compilation Options] : シミュレーション用のデザインを準備するために使用するコンパイラのコマンド ライン オプションを指定します。

[Simulation Options] : シミュレーション起動時に使用するランタイム指示子を指定します。

[Netlist Options] : シミュレーションで使用する Verilog または VHDL ネットリストを作成する NetGen プログラムのオプションを定義します。

次のセクションで、このダイアログ ボックスのオプションについて説明します。

コンパイル オプション

[Simulation Options] ダイアログ ボックスの [Launch Options] タブでは、Fuse および ISim で実行される起動オプションを指定します。図 11-20 に、[Simulation Options] ダイアログ ボックスを示します。⁽¹⁾

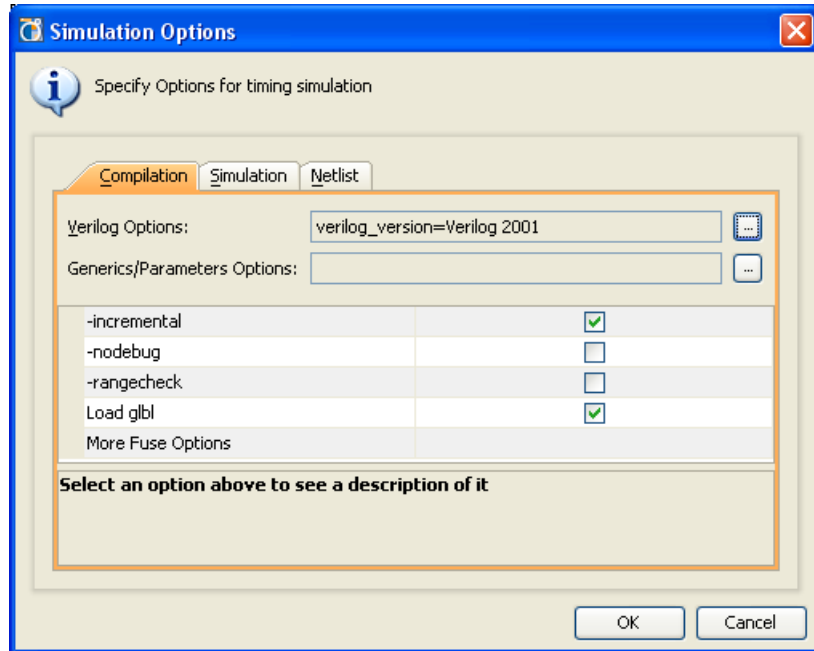


図 11-20 : [Timing Simulation Launch Options] ダイアログ ボックス

- [Verilog Options] : Verilog 検索パス、マクロ定義、大文字識別子 を指定します。詳細は、第 3 章の「プロジェクト設定」を参照してください。
- [Generics/Parameters Options] : ジェネリックまたはパラメーターのキーワードおよび値を指定します。
- [-incremental] : 前回のコンパイルから変更されたファイルのみを fuse リンカーおよびコンパイラでコンパイルします。
- [-nodebug] : シミュレーション中に HDL コードをデバッグするための情報を含まないシミュレーション実行ファイル (.exe) を作成します。これにより、シミュレーション ランタイムが短縮されます。
- [-rangecheck] : fuse リンカーおよびコンパイラでコンパイル中に VHDL 代入の値範囲をチェック します。このオプションは、VHDL コードにのみ適用されます。
注記 : これにより、ISim で常に行われるアレイのインデックス範囲チェックに影響が出ることはありません。
- [Load glbl] : コンパイル中に glbl モジュールを読み込むかどうかを指定します。デザインに Verilog UniSim または SimPrim ライブラリが使用される場合は、このオプションをオンにする必要があります。
- [More Fuse Options] : fuse のその他のコマンド ライン オプションを指定します。コマンドは、1 つの文字列で値とセットにして入力する必要があります。たとえば、次のように入力します。

1. コンパイル オプションは、シミュレーターが ModelSim または Questa の場合は異なります。


```
-maxdelay -init_file <filename> -notimingcheck
```

fuse オプションをコマンド ファイルに追加しておき、次のように **-f** コマンドを使用して [More Fuse Options] フィールドにそのファイルを指定することもできます。

```
-f <command_file>
```

シミュレーション オプション

図 11-21 は、[Simulation Options] ダイアログ ボックスの [Simulation] タブを示しています。⁽¹⁾

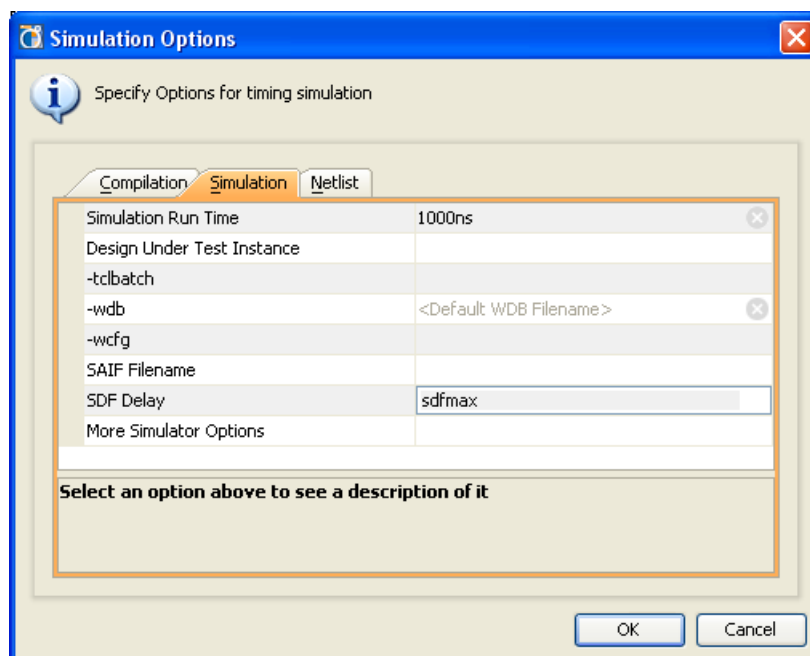


図 11-21 : [Timing Simulation Language Options] タブ

- [Simulation Run Time] : シミュレーション時間とその単位を指定します。指定できる時間の単位は、fs、ps、ns、us、ms、sec で、デフォルトは ps です。キーワードとして all を指定すると、シミュレーションするイベントがなくなるまで ISim が実行されます。
- [Design Under Test] : テストするユニット名を指定します。通常、これは最上位モジュールと同じ名前になりますが、テストベンチに複数の最上位モジュールの可能性が含まれる場合は、テストするユニットの選択を尋ねるメッセージが表示されます。
- [-tclbatch] : ランタイム時にシミュレーターで実行されるバッチ ファイルにリストされる Tcl コマンドのファイル名を指定します。指定したバッチ ファイルに含まれるコマンドは、完了するまで連続して実行されます。ISim では、バッチ ファイルの実行が終了するまで、コマンドプロンプトに入力されるコマンドが無視されます。

PlanAhead では tclbatch コマンドが使用され、isim.cmd というファイルで ISim に必須の 3 つのコマンドが渡されます。このファイルの内容は、次のとおりです。

```
onerror {resume}
.
wave add /
run <value>
```

1. シミュレーション オプションは、シミュレーターが ModelSim または Questa の場合は異なります。

起動時に実行するシミュレーターを制御する Tcl コマンドを作成する場合、これらの 3 つのコマンドが `tclbatch` ファイルに必ず含まれるようにしてください。 `onerror` コマンドを最初に記述し、 `wave add` と `run` を最後に記述することを推奨します。その他の ISim コマンド ライン コマンドは、 `onerror` と `wave add` の間に追加できます。

注記： `tclbatch` コマンド ファイルの拡張子は、 `.tcl` または `.cmd` のいずれかにしないと、ISim で正しく処理されません。

- `[-wdb]`：シミュレーション波形データを保存するファイル名を指定します。トレースされる信号のシミュレーション結果は、指定のファイル名で作業ディレクトリに保存されます。PlanAhead では、デフォルトで `<top_module_name>.wdb` ファイルが作成されます。
- `[-wcfg]`：ISim GUI に波形データを開く際に使用する波形コンフィギュレーション ファイルの名前を指定します。波形コンフィギュレーション ファイルでは、信号の順序、名前のスタイル、基数、色などの設定が指定されます。
- `[SAIF Filename]`：ポートおよび信号のスイッチ レートを記録する SAIF (Switching Activity Interchange Format) ファイルの名前を指定します。デフォルトのファイル名は、 `xpower.saif` です。
- `[SDF Delay]`：ISim で使用される遅延タイプを指定します。使用できる値は、次のとおりです。
 - 最小遅延をアノテートする場合は `sdfmin`
 - 最大遅延をアノテートする場合は `sdfmax`

セットアップ違反は `sdfmax`、ホールド違反は `sdfmin` を指定し、別々のシミュレーションを実行してチェックすることをお勧めします。

- `[More Simulator Options]`：ISim のその他のコマンド ライン オプションを指定します。これらのコマンドは、コマンド値と一緒に 1 行で入力します。たとえば、次のように入力します。

```
-log <filename> -transport_int_delays
```

ISim オプションをコマンド ファイルに追加しておき、次のように `-f` コマンドを使用して、`[More Simulator Options]` フィールドにそのファイルを指定することもできます。

```
-f <command_file>
```

ModelSim オプション

ターゲット シミュレーターに QuestaSim/ModelSim を指定した場合、コンパイルおよびシミュレーション オプションは異なります。次は、ModelSim のオプションです。

[Compilation] セクション

- `[VHDL Syntax]`：ソース ファイルおよびテストベンチで使用する VHDL 言語のバージョンを指定します。有効な値は、93 (1993) および 87 (1987) です。
- `[Explicit Declarations]`： `explicit` 関数宣言を使用するか、コンパイラーで自動的に作成された関数宣言を使用するか曖昧な場合に使用します。
- `[More VLOG Options]`：ツールで使用するその他の Verilog コンパイラー (`vlog`) コマンド ライン オプションを指定します。
- `[More VCOM Options]`：ツールで使用するその他の VHDL コンパイラー (`vcom`) コマンド ライン オプションを指定します。

シミュレーション

- `[Simulation Run Time]`：シミュレーション時間とその単位を指定します。

- [Log All Signals] : シミュレーション中の信号値をすべて保存して、未知の信号ステートのソースをトレースしやすくします。
- [SDF Delay] : ModelSim で使用される遅延タイプを指定します。使用できる値は、次のとおりです。
 - 最小遅延をアノテートする場合は `sdfmin`
 - 最大遅延をアノテートする場合は `sdfmax`
 セットアップ違反は `sdfmax`、ホールド違反は `sdfmin` を指定し、別々のシミュレーションを実行してチェックすることをお勧めします。
- [Design Under Test Instance] : テストベンチで定義された DUT モジュールの名前が表示されます。
- [SAIF Filename] : ポートおよび信号のスイッチ レートを記録する SAIF (Switching Activity Interchange Format) ファイルの名前を指定します。
- [More VSIM Options] : ツールで使用するその他の ModelSim (vsim) コマンド ライン オプションを指定します。

[Netlist Options] タブ

[Netlist Options] タブには、インプリメント済み FPGA デザインから Verilog または VHDL ネットリストを生成する NetGen プログラムを制御するためのオプションが含まれます (図 11-22)。

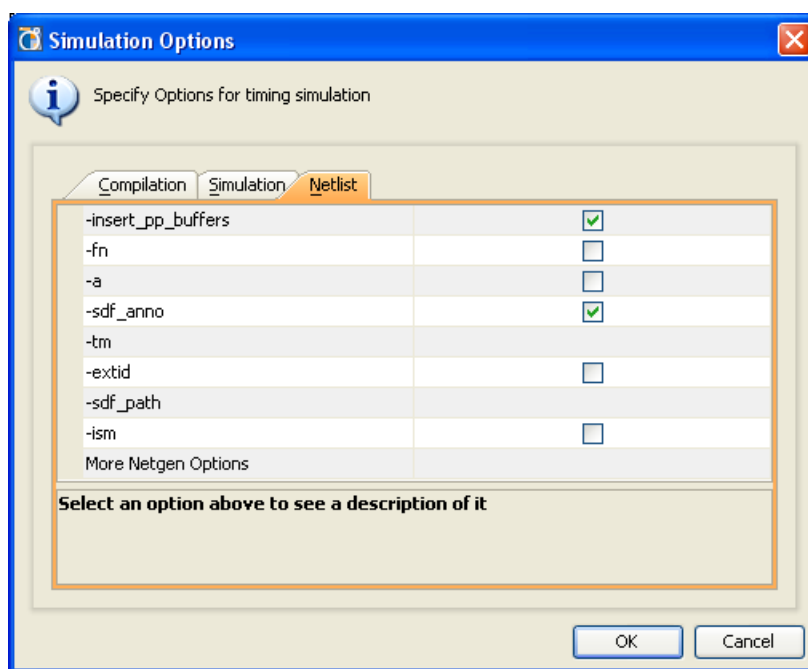


図 11-22 : [Simulation Options] ダイアログ ボックスの [Netlist Options] タブ

このタブでは、NetGen のシミュレーション ネットリスト記述方法を指定できます。

- [-insert_pp_buffers] : パルスが失われるのを回避するためにパス パルス バッファを挿入するかどうかを指定します。タイミング シミュレーションでは、パルス幅よりコンポーネントの入力ポートの遅延の方が大きい場合、パルスが失われる場合があります。
- [-fn] : 階層ネットリストではなく、フラットにしたネットリストを出力するかどうか指定します。

- [-a] : VHDL 出力でアーキテクチャのみを生成するかどうか指定します。このオプションを使用すると、出力ファイルに VHDL エンティティは生成されません。
- [-sdf_anno] : NetGen で出力されるネットリストに、遅延アノテーションのための sdf_annotate 文を追加するかどうか指定します。
- [-tm] : NetGen からの出力に使用される最上位モジュールの新しい名前を指定します。
- [-extid] : VHDL 出力に拡張識別子を出力するかどうか指定します。
- [-sdf_path] : NetGen で作成された SDF ファイルを出力するパスを指定します。デフォルトでは、SDF ファイルはシミュレーション run ディレクトリに出力されます。
- [-ism] : SimPrim ライブラリの SimPrim モジュールを出力 Verilog ファイルに含めます。このオプションを使用すると、シミュレーション時にライブラリ パスを指定する必要がなくなりますが、コンパイル時間が長くなり、ネットリスト ファイルのサイズが大きくなります。
- [More NetGen Options] : NetGen 用のその他のコマンド ライン オプションを指定します。コマンドは、1 つの文字列で値とセットにして入力する必要があります。たとえば、次のように入力します。

```
-aka -gp <port_name> -s 3
```

NetGen オプションすべてをコマンド ファイルに追加しておき、次のように **-f** コマンドを使用して[Netgen] フィールドにそのファイルを指定することもできます。

```
-f <command_file>
```

ISim ツール

[Launch Timing Simulation] ダイアログ ボックスの [Launch] ボタンをクリックすると、ISim を起動できます (398 ページの図 11-19)。PlanAhead は NetGen を起動して、MAP または PAR から NCD ファイルを読み込み、その結果に基づいて一部分またはフルのタイミング SDF ネットリストを作成します。

タイミング シミュレーションを起動する前に、バックアノテーション用にタイミング シミュレーション モデルと遅延ファイルが必要です。PlanAhead は NetGen ツールを使用してこれらのファイルを生成します。NetGen の詳細は、付録 E 「その他のリソース」に示す『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。

NetGen では、出力 SDF ファイルにフラットにしたタイミング遅延が作成され、シミュレーション 目的に Verilog (VHDL) ネットリスト (top_timing_sim.v) が作成されます。

注記 : Verilog または VHDL ファイルは、シミュレーション目的にのみ出力されるので、合成はできません。

NetGen で作成される Verilog または VHDL ファイルは、ザイリンクスのシミュレーション プリミティブ ライブラリ (SimPrim) を参照するので、シミュレーション中は SimPrim と一緒に使用する必要があります。NetGen が終了すると、ISim のオブジェクト コンパイラおよびリンカーの fuse が実行され、Verilog および VHDL コードがコンパイル/エラボレーションされます。

この後、コンパイルされたオブジェクト コードは、[Launch Timing Simulation] ダイアログ ボックスで指定した最上位モジュールと同じシミュレーション実行ファイルにリンクされます。ISim の実行ファイルが完了すると、シミュレーターが起動します。

```
INFO:[Runs-8] Fuse completed.
INFO:[Runs-10] Launching ISim...
INFO:[Runs-11] Running "C:/project_cpu_hdl/project_cpu_hdl.sim/sim_1/
top.exe"
```

```
-intstyle pa -gui -tclbatch ISim.cmd
-wdb "wdb test1.wdb" -view "wcfg test1.wcfg"
```

シミュレーション実行ファイルは、[Simulation Options] で指定したさまざまなオプションで実行されます。

PlanAhead では、ISim が `-gui` オプションで起動されます。これにより、ISim が GUI モードで開くので、デザインをインタラクティブにシミュレーションできます。GUI から ISim を実行する方法については、付録 E 「その他のリソース」に示す『ISim ユーザー ガイド』(UG660)を参照してください。

404 ページの図 11-23 に、ISim の GUI を示します。

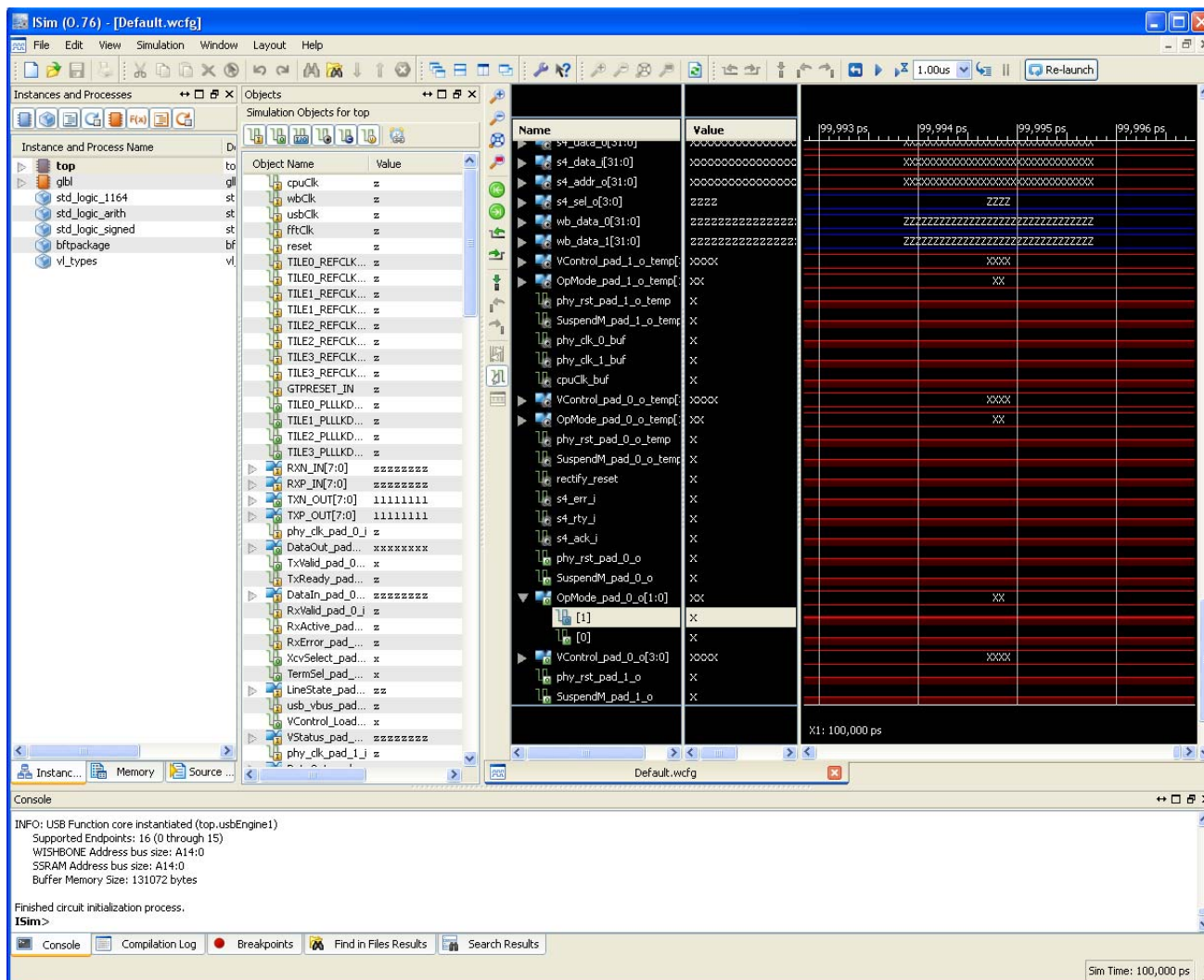
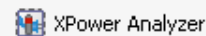


図 11-23 : ISim のユーザー インターフェイス

XPower Analyzer を使用した消費電力分配解析

XPower Analyzer (XPA) ツールは PlanAhead ソフトウェアから直接起動でき、インプリメント済みデザインで消費電力解析を実行できます。

- Flow Navigator から [XPower Analyzer] をクリックします。
- メイン メニューから [Tools] → [Analysis] → [Xpower Analyzer] をクリックします。



405 ページの図 11-24 に示すような [Launch XPower Analyzer] ダイアログ ボックスが開きます。

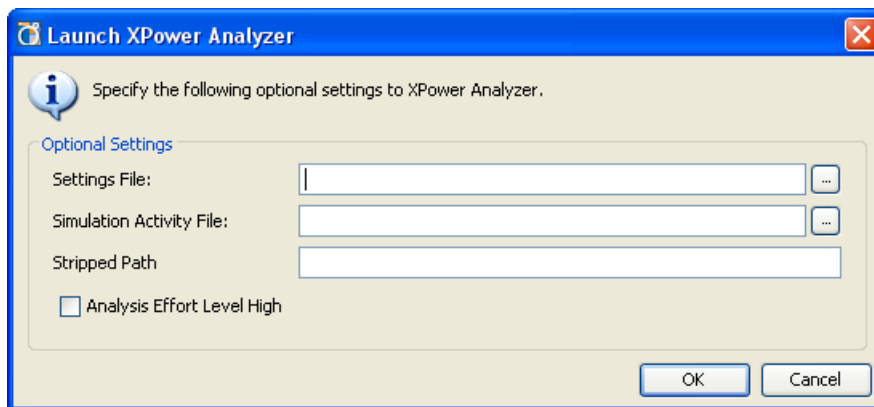


図 11-24 : [Launch XPower Analyzer] ダイアログ ボックス

次のような XPA コマンドのオプションを指定します。

[Settings File] : 既存の XPower Analyzer ファイル (.xpa) を指定して、前の run からの設定をインポートしてこの run で使用できます。

[Simulation Activity File] : 消費電力解析中に使用する Switching Activity Interchange Format (SAIF) ファイルまたは Value Change Dump (VCD) ファイルを読み込みます。

[Stripped Path] : SAIF または VCD ファイルのエレメントから指定したインスタンスパスの接頭語を取り去って、現在のデザインのインスタンスに問題なくマップできるようにします。

[Analysis Effort High] : エフォート レベル High の精度とランタイムを増加します。

起動すると、配線済みの NCD ファイルおよびタイミング制約 (PCF ファイル) が自動的に XPower Analyzer に読み込まれます。406 ページの図 11-25 は、XPower Analyzer ツールを示しています。XPower Analyzer の詳細は、[ISE ヘルプ](#)を参照してください。

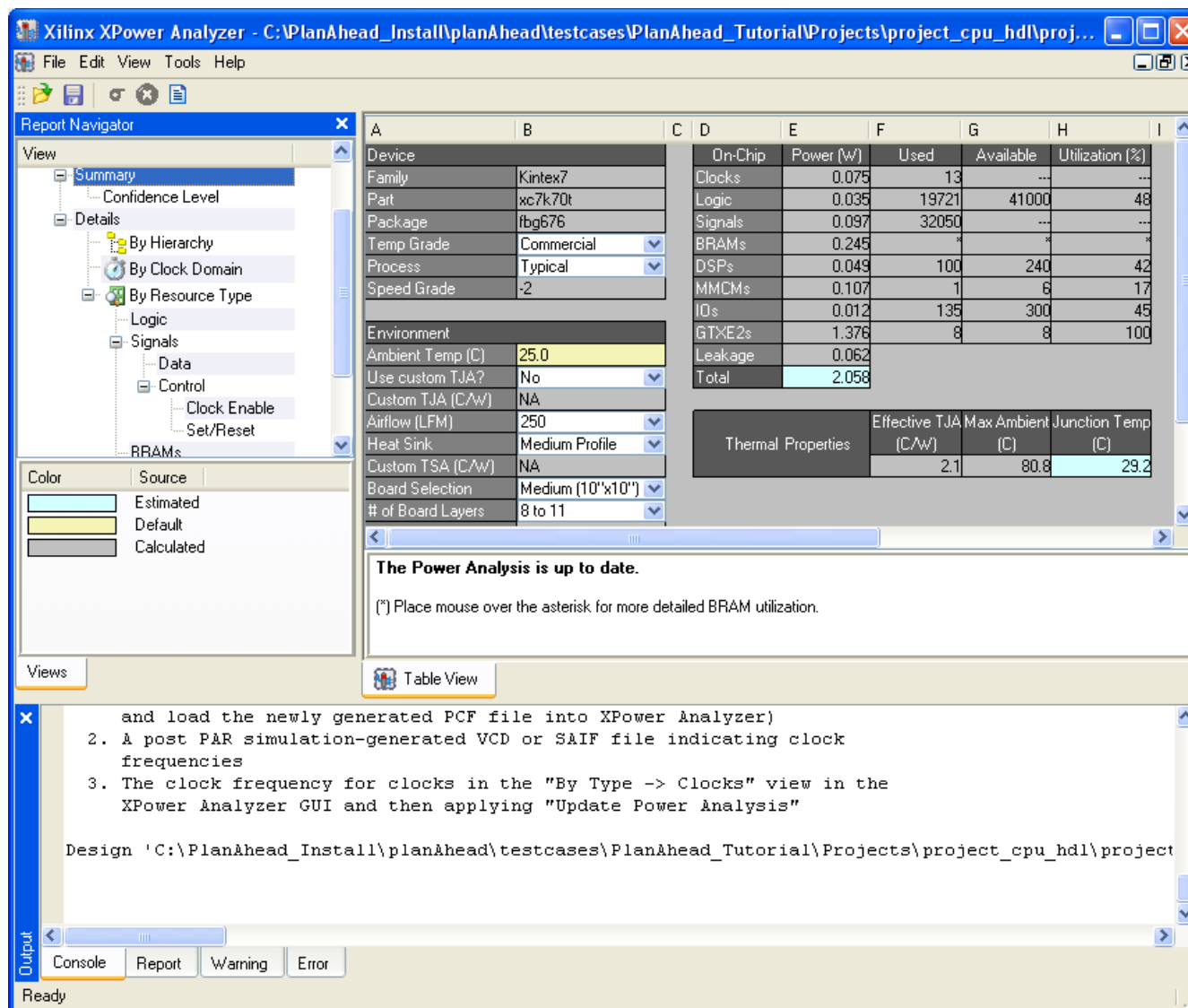


図 11-25 : XPower Analyzer

FPGA Editor の起動

FPGA Editor は、どのインプリメントされたデザインに対しても PlanAhead から直接起動できます。FPGA Editor は、次のいずれかの手順で起動できます。

- Flow Navigator で [Implementation] → [FPGA Editor] をクリックします。
- メイン メニューから [Tools] → [Analysis] → [FPGA Editor] をクリックします。

配線済みの NCD ファイルが自動的に FPGA Editor に読み込まれ、インプリメント済みデザインが開きます。FPGA Editor の詳細は、[ISE ヘルプ](#)を参照してください。

FPGA Editor へのタイミング パスのクロスプローブ

PlanAhead のタイミング パスから FPGA Editor へクロスプローブするには、[Device] ビューの [Timing Results] ビューでタイミング パスを選択します。ロジック インスタンスを個別に選択して FPGA Editor へクロスプローブすることもできます。

図 11-26 に示すように、FPGA Editor に選択したパスまたはインスタンスがハイライトされた状態で表示されます。

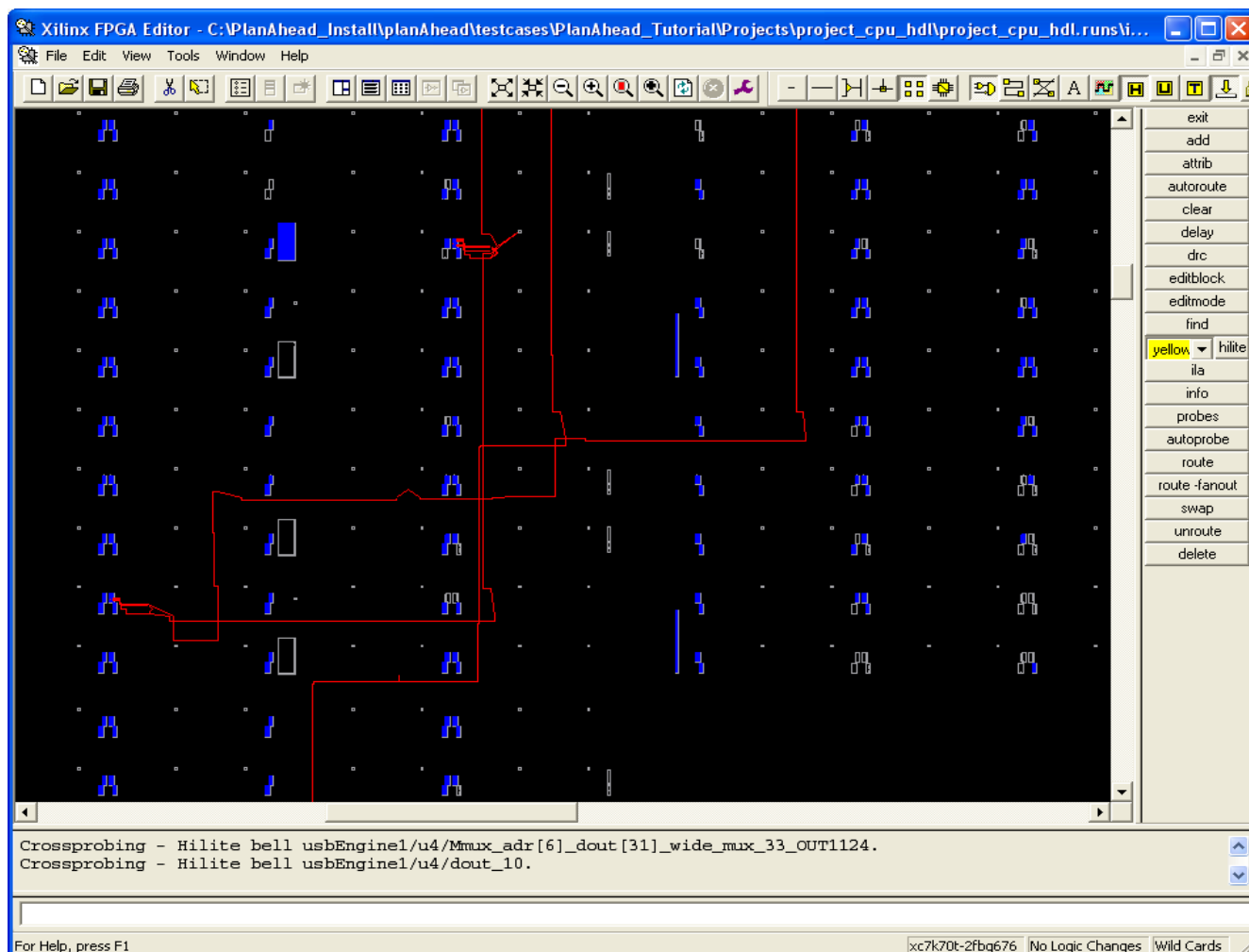


図 11-26 : FPGA Editor

デザインのプログラムとデバッグ

この章では、デザインのプログラムおよびデバッグについて、次のセクションに分けて説明します。

- 「ビットストリーム ファイルの生成」
- 「ChipScope を使用したデザインのデバッグ」
- 「ChipScope Pro Analyzer の起動」
- 「iMPACT の起動」

ビットストリーム ファイルの生成

インプリメンテーションが完了したら、ISE® の BitGen コマンドを実行して、ザイリンクス デバイスをプログラムするためのビットストリーム データを作成できます。

1. BitGen コマンド ライン オプションは、次のいずれかの方法で指定できます。

- Flow Navigator で [Bitstream Settings] をクリックします。
- メイン メニューから [Flow] → [Bitstream Settings] をクリックします。

410 ページの図 12-1 に示す [Bitstream Project Settings] ダイアログ ボックスが開きます。

このコマンドを実行する前に ISE の BitGen コマンド オプションを設定できます。オプションを選択すると、そのオプションに関する説明がダイアログ ボックス下部に表示されます。BitGen のオプションについては、付録 E 「その他のリソース」に示す『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。

2. [OK] または [Apply] をクリックして選択したオプションを設定します。
3. BitGen は、次のいずれかの方法で実行できます。

- Flow Navigator で [Generate Bitstream] をクリックします。
- [Flow] → [Generate Bitstream] をクリックします。

コマンドのステータスは [Compilation Log] ビューおよび [Compilation Messages] ビューで確認でき、終了したら [Reports] ビューに BitGen レポート ファイルが表示されます。

結果の BIT ファイルはプロジェクトの Run ディレクトリに保存されます。

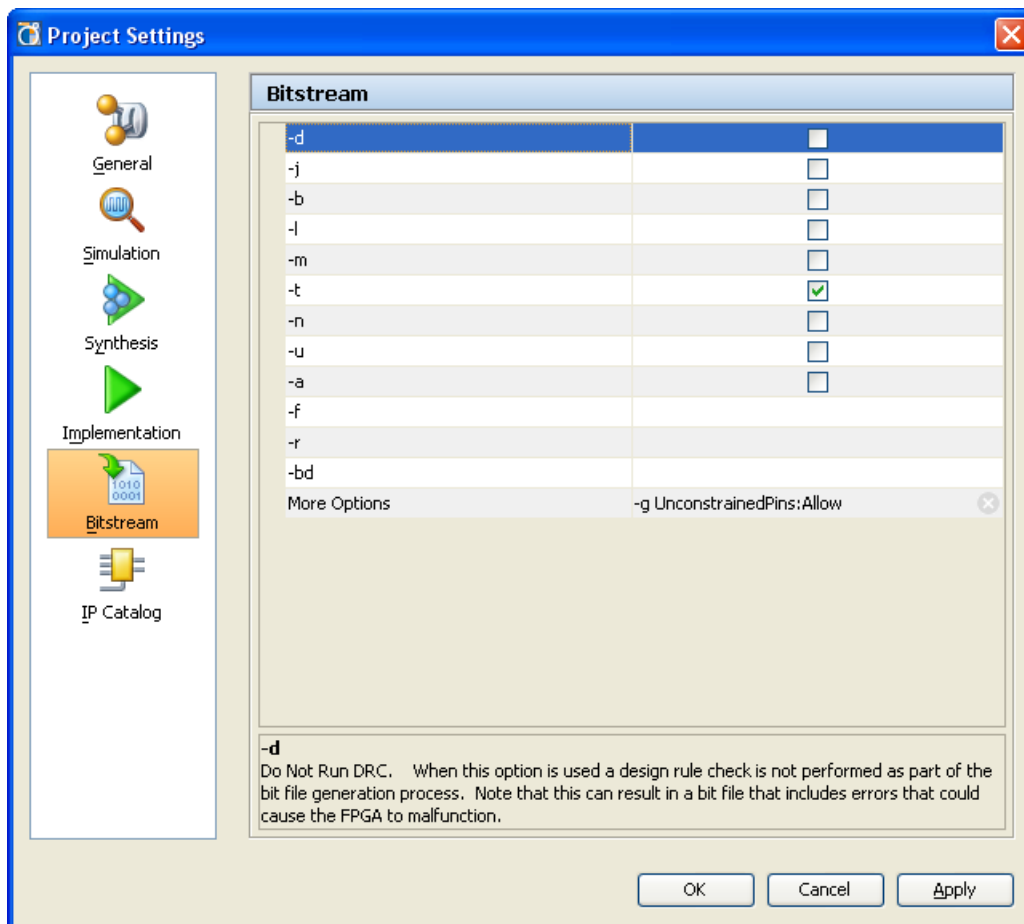


図 12-1 : [Generate Bitstream] ダイアログ ボックス

ChipScope を使用したデザインのデバッグ

PlanAhead™ は、ChipScope™ Pro デバッグ ソフトウェアと統合されています。

ChipScope Pro との統合により、ChipScope Pro Integrated Logic Analyzer (ILA) デバッグ コアを合成後に簡単に挿入および接続できます。

PlanAhead と ChipScope の統合の概要

PlanAhead ソフトウェアでは、ウィザードを使用してほとんどのデザインのデバッグをすばやく簡単に実行できます。ウィザード以外の GUI 機能や Tcl コマンド フローも、精密なデバッグやネット接続のために使用できます。このフローを使用すると、PlanAhead 環境内で多機能な ILA コアを接続することができます。411 ページの図 12-2 は、デバッグ コアの統合を示しています。

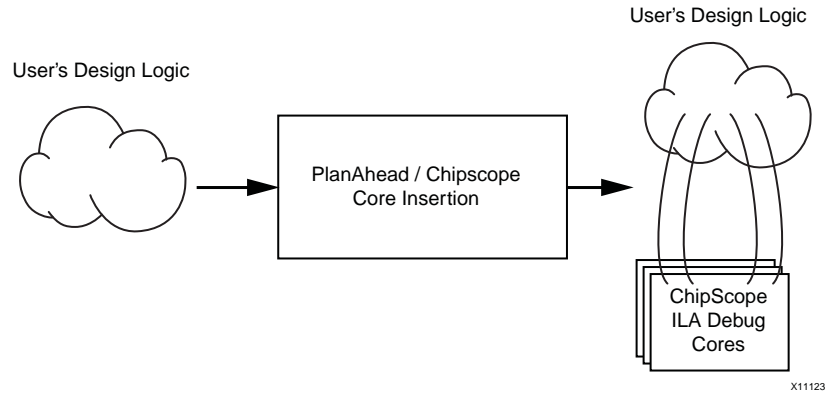


図 12-2 : デバッグ コアの統合

コア挿入フローを使用する場合の要件および制限

PlanAhead で ChipScope の機能を使用するには、ChipScope Pro および PlanAhead を含む ISE Design Suite 14.x ツールがインストールされている必要があります。

ランタイム デザイン デバッグを実行するには、ザイリンクス ターゲット デザイン プラットフォーム (TDP) の USB ケーブルおよびボードも必要です。ChipScope Pro の詳細は、次を参照してください。

http://japan.xilinx.com/support/documentation/dt_chipscopepro.htm

PlanAhead に統合された ChipScope の機能には、次のような制限があります。

- このフローを使用する場合は、PlanAhead、ISE、ChipScope Pro は必ず同じバージョンである必要があります。これらのツールのほかのバージョンは使用できません。
- このフローは、Project Navigator または ChipScope Pro Core Inserter フローでは使用できませんが、ChipScope デバッグ コア (CDC) を PlanAhead にインポートすることはできます。
- このフローは、PlanAhead を ISE Project Navigator から起動した場合は使用できません。
- ChipScope Pro Integrated Controller (ICON) コアに既存デバッグ コアが接続されている場合、表示はできますが変更はできません。
- コア外部でインスタンス化された BSCAN プリミティブへの接続を必要とする BSCAN プリミティブなしに生成された ICON が既存する場合、このフローはその ICON とは互換性がありません。
- PlanAhead ではデバッグ コアが合成後のデザイン ネットリストに追加されるため、合成中に行われるトリミングや最適化が原因で、デバッグできないネットがある場合があります。
- このフローで生成および接続できるのは、ChipScope Pro ILA コアのみです。

コア挿入フローの使用

ChipScope デバッグ コアの PlanAhead への挿入方法は、PlanAhead ユーザーのさまざまなニーズに対応できるよう複数あります。

- シンプルな GUI のウィザードを使用し、デバッグするために選択したネットに基づいて、ILA コアを自動生成および設定するのが一番簡単な方法です。

- また ChipScope のメイン ウィンドウで個々のコア、ポート、およびパラメーターを設定することもできます。[Layout Selector] または [Layers] メニューから [ChipScope] ビューを選択するか、[Windows] → [ChipScope] をクリックすると [ChipScope] ビューを開くことができます。
- Tcl デバッグ コマンドを使用し、スクリプトを手動で作成したり再生することもできます。

これらの方法を合わせて利用し、デバッグ コアを挿入およびカスタマイズすることもできます。

デバッグ コア挿入モードの決定

次の表は、デバッグ 目標およびその挿入モードをまとめたものです。

表 12-1：デバッグ 目標およびコア挿入モード

デバッグ 目標	コア挿入モード
選択したネットに対しデフォルト設定を使用して ILA デバッグ コアをすばやく生成	ChipScope ウィザード
既存のデバッグ コアのパラメーターを変更	[ChipScope] ビュー
既存のデバッグ コアを手動で生成または削除	[ChipScope] ビュー
ILA コアのトリガーまたはデータ ポートを手動で生成、削除、設定	[ChipScope] ビュー
ネットをトリガー、データ、クロック チャネルに手動で割り当て	[ChipScope] ビュー
デバッグ コマンドを記録したスクリプトを再生	Tcl コマンド

デバッグのためのネットのマーク

デバッグ フローでは、最初にデバッグするネットを識別します。

ChipScope デバッグ コアの挿入およびコンフィギュレーションは、すべて [Synthesized Design] ビューで実行する必要があります。これは、これらのコアをインプリメンテーション前にネットリストへ追加する必要があるためです。詳細は、[235 ページの「合成済みデザイン環境の使用」](#)を参照してください。

[Netlist] または [Schematic] ビューでネットまたはバスのセットを選択し、[Add to ChipScope Unassigned Nets] をクリックするか、[Netlist] ビューからネットを /Unassigned nets フォルダーにドラッグ アンド ドロップします。

[Schematic] ビューなどのほかのビューでネットやバスを選択することもできます。SSN 結果は [図 12-3](#) のように表示されます。

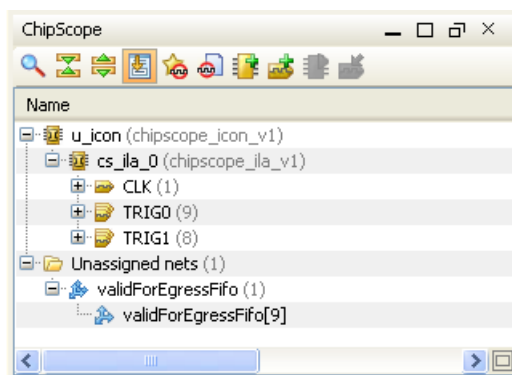


図 12-3 : [ChipScope] ビューの割り当てのないネット リスト

また、Set up ChipScope ウィザードでもネットを選択できます。

ネットは、合成前にデバッグするために識別することも可能です。HDL や制約ファイルでデバッグ用にマークされたネットは、[ChipScope] ビューの [Unassigned nets] フォルダおよび Set up ChipScope ウィザードに自動的に表示されます。

デバッグ用にネットをマークする方法は、プロジェクトが RTL ソース ベースであるか合成済み ネットリスト ベースであるかによって異なります。

RTL ネットリスト ベースのプロジェクトの場合は、

- XST (Xilinx Synthesis Technology) を使用すると、VHDL/Verilog ソースまたはザイリンクス 制約ファイル (XCF) で mark_debug 制約を使用してデバッグ用ネットをマークできます (オプション)。mark_debug 制約の詳細は、付録 E「その他のリソース」に示す『制約ガイド』(UG625) を参照してください。

この制約には、ブール型文字列値の "true" や "false" だけでなく、"soft" を使用して、指定ネットを最適化できます。

注記：XST では、Spartan-6、Virtex-6 およびそれ以降のデバイスでこの制約が自動的にサポートされます。

合成済みネットリスト ベースのプロジェクトの場合は、

- Synopsis® 社の Synplify® 合成ツールを使用すると、VHDL や Verilog の場合は mark_debug および syn_keep 制約、SDC (ynopsys Design Constraints) ファイルの場合は mark_debug 制約のみを使用して、デバッグ用のネットをマークできます (オプション)。Synplify では、"soft" 値はサポートされません。これは、この動作が syn_keep 制約で制御されるためです。
- Mentor Graphics® 社の Precision® 合成ツールを使用すると、VHDL や Verilog で mark_debug 制約を使用してデバッグ用のネットをマークできます。

次のセクションは、XST、Synplify、Precision ソース ファイルの構文例です。

XST の構文例

次に、XST を使用する場合は VHDL、Verilog、XCF の構文例を示します。

VHDL の構文例

```
attribute mark_debug : string;
attribute mark_debug of char_fifo_dout: signal is "true";
```

Verilog の構文例


```
(* mark_debug = "true" *) wire [7:0] char_fifo_dout;
```

XCF の構文例

```
BEGIN MODEL "wave_gen"  
NET "char_fifo_dout" mark_debug= "true";  
END;
```

Synplify の構文例

次に、Synplify を使用する場合は VHDL、Verilog、SDC の構文例を示します。

VHDL の構文例

```
attribute syn_keep : boolean;  
attribute mark_debug : string;  
attribute syn_keep of char_fifo_dout: signal is true;  
attribute mark_debug of char_fifo_dout: signal is "true";
```

Verilog の構文例

```
(* syn_keep = "true" *) (* mark_debug = "true" *) wire [7:0]  
char_fifo_dout;
```

SDC の構文例

```
define_attribute {n:char_fifo_din[*]} {mark_debug} {"true"}
```

注記：SDC ソースのネット名には、接頭辞に **n** が付きます。詳細は、[442 ページの「SDC について」](#)を参照してください。

Precision の構文例

次に、Precision を使用する場合は VHDL、Verilog、XCF の構文例を示します。

VHDL の構文例

```
attribute mark_debug : string;  
attribute mark_debug of char_fifo_dout: signal is "true";
```

Verilog の構文例

```
(* mark_debug = "true" *) wire [7:0] char_fifo_dout;
```

ChipScope ウィザードを使用したデバッグ コアの挿入

PlanAhead にデバッグ コアを追加するには、Set up ChipScope デバッグ ウィザードを使用するのが最も簡単です。

Set Up ChipScope ウィザードを使用してデバッグ コアを挿入するには、次の手順に従います。

1. 割り当てのないネット リストまたは直接ネットをクリックしてデバッグするネットを選択します (オプション)。
2. [Tools] → [Set Up ChipScope] をクリックします。
3. デバッグ コアを接続および設定するには、Set Up ChipScope ウィザードの画面に従ってください。

ChipScope の CDC ファイルのインポート

Set up ChipScope ウィザードを使用すると、プロジェクトに既存の ChipScope Debug Core (CDC) ファイルを追加できます。

[Set up ChipScope] ボックスをクリックすると、ウィザードの最初の画面に [Import Existing ChipScope CDC file] チェック ボックスが表示されます。

チェック ボックスをオンにし、[Next] をクリックします。

CDC ファイルを選択し、[Next] をクリックします。

注記：一部の ChipScope コアはインポートできません。ChipScope Core Inserter または PlanAhead からの CDC ファイルには、インポートに必要なコア情報が含まれています。

デバッグ ネットの選択または確認

[Unassigned nets] のリストにネットが追加された場合は、それらを使用するか新しいネットを選択するかが尋ねられます。[Add/Remove Nets] ページが表示されます。ここでは、デバッグするネットを検索して選択できます。

ネットを必要に応じて追加または削除したら、[Next] をクリックします。

デバッグ ネットおよびクロック ドメインの指定

Set up ChipScope ウィザードでは、図 12-4 のように選択した各ネットまたはバスに対し、正しいクロックが自動検出されます。

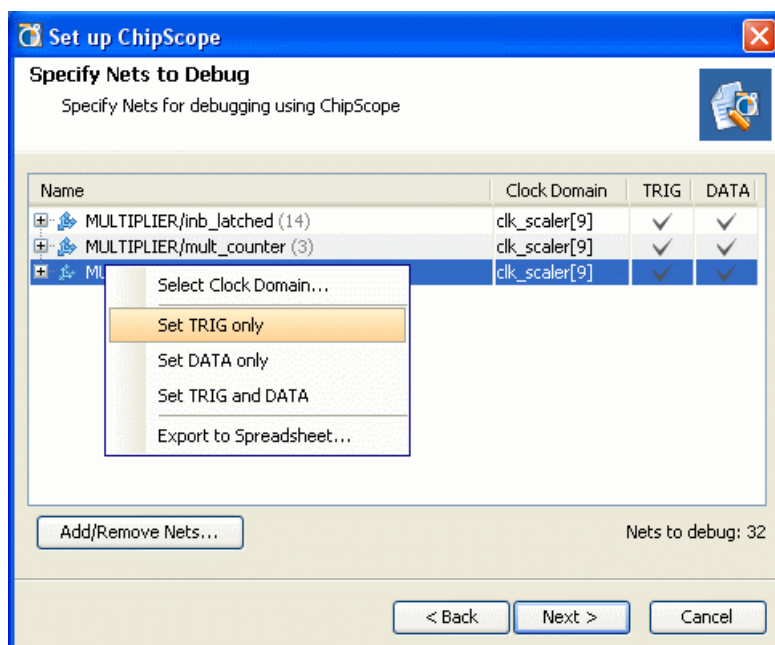


図 12-4：デバッグ ネットおよびクロック ドメインの指定

複数のクロックが検出された場合は、ドロップダウン リストを使用してクロックを選択できます。

1. [Add/Remove Nets] をクリックしてデバッグ ネットを選択/選択解除することもできます。
2. 各ネットまたはバスは、トリガー、データ ストレージ、またはその両方として使用できるよう設定できます。
3. ネットおよびクロックの設定が正しい場合は、[Next] をクリックしサマリ ページに進みます。
デザインに ILA コアが含まれる場合は、それらを削除してビュー情報に基づいて生成し直すか、そのまま維持して新規のコアを生成するか選択します。

ILA コアの挿入

Set up ChipScope ウィザードは、クロックドメインにつき 1 つの ILA コアを挿入します。

デバッグのために選択されたネットは、インスタンス化された ILA コアのトリガーおよびデータポートとして自動的に割り当てられます。

ウィザードの最終画面はコア生成のまとめのページで、検出されたクロック数、生成および削除される ILA コアの数が表示されます。

内容を確認したら [Finish] をクリックし、デザインに ILA コアをインスタンス化および接続します。

[ChipScope] ビューを使用したデバッグ コアの追加とカスタマイズ

ILA コア挿入に対する詳細な設定は、ChipScope ウィザードではできないので、[ChipScope] ビューを使用します。

このビューでは、コアの生成および削除、ネット接続のデバッグ、およびコアパラメーターの変更を設定できます。

[ChipScope] ビューには、次が表示されます。

- ICON コントローラー コアに接続されたデバッグ コアのリストが表示されます。
- ビューの下部に、割り当てられていないネットのリストも管理されます。

デバッグ コアおよびポートは、ポップアップメニューまたはビューの上部にあるツールバーから制御できます。

デバッグ コアの生成および削除

[ChipScope] ビューで ChipScope デバッグ コアを生成するには、[Create Debug Core] をクリックします。

このインターフェイスを使用すると、親インスタンスの変更、コア名のデバッグ、コアのパラメーター設定ができます。

既存のデバッグ コアを削除するには、[ChipScope] ビューでコアを選択し、[Delete] をクリックします。

デバッグ コア ポートの追加、削除、およびカスタマイズ

デバッグ コアの追加および削除だけでなく、各デバッグ コアのポートを追加、削除およびカスタマイズすることができます。ポートを追加するには次の手順に従います。

1. コアを選択します。
2. [Create Debug Port] をクリックします。

図 12-5 のような [Create Debug Port] ダイアログボックスが表示されます。

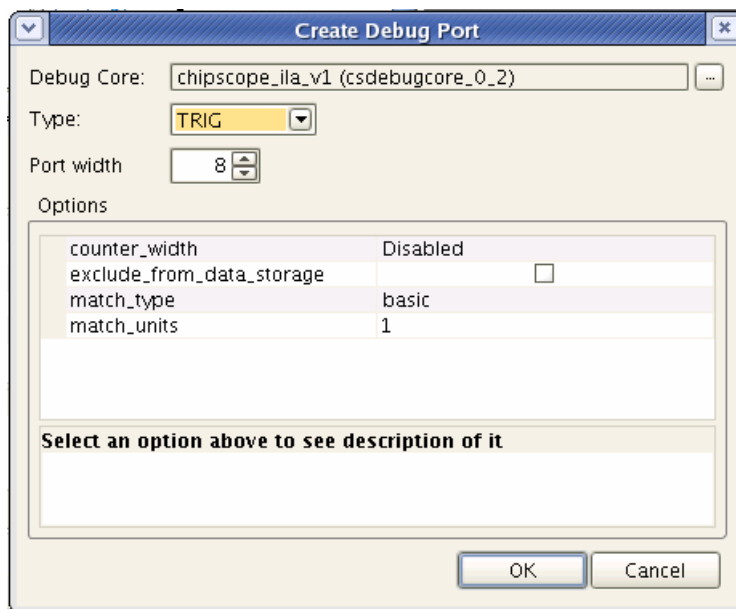


図 12-5 : デバッグ コアのポートおよびオプションのカスタマイズ

3. ドロップダウン リストからポートのタイプを選択します。

[Options] でポートに対しオプションを設定します。ポート幅はデフォルト値が表示されますが、ポートからネットを追加および削除することに値が変化します。

4. [OK] をクリックします。

デバッグ ポートを削除するには、[ChipScope] タブでポートを選択し、[Delete] をクリックします。

デバッグ コアへのネットの接続および接続解除

ネットおよびバス (ネットのベクター) は [Schematic] または [Netlist] ビューからデバッグ コアのポートヘドラッグ アンド ドロップできます (次の図を参照)。これで、選択内容に応じてポートが自動的に拡張します。

また、ネットまたはバスを右クリックし、[Assign to ChipScope Debug Port] を選択することもできます。

デバッグ コアのポートからネットの接続を解除するには、ポートに接続されているネットを選択し、[Disconnect Net] をクリックします。図 12-6 は、この動作を示しています。

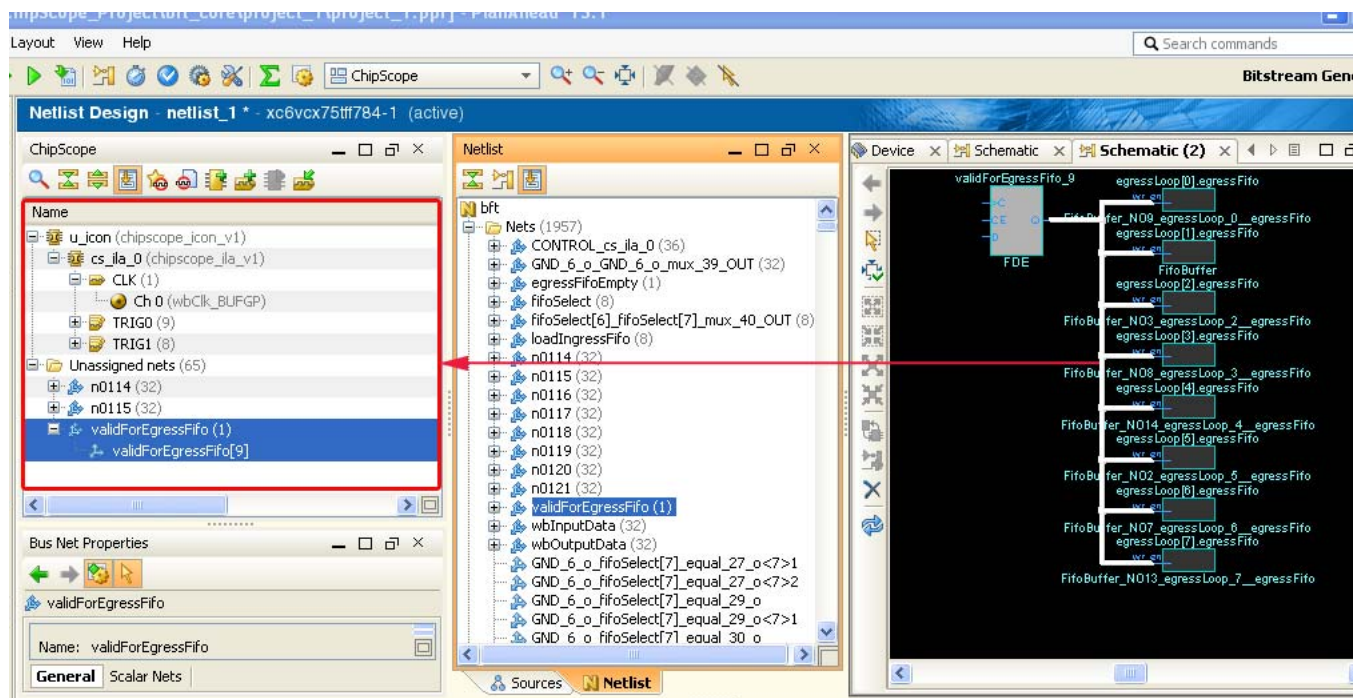


図 12-6：デバッグ コア ポートへのネットのドラッグ アンド ドロップ

デバッグ コアおよびポート パラメーターのカスタマイズ

ChipScope デバッグ コアにはカスタマイズ可能なパラメーターがあります。

これらのコアのパラメーターを表示するには、次の手順に従います。

1. [ChipScope] ビューで ChipScope デバッグ コアを 1 つ選択します。
これにより [Instance Properties] ビューが開いて、選択したデバッグ コアのプロパティが表示されます。
2. 418 ページの図 12-7 のように、[Instance Properties] ビューで [Debug Core Options] を選択し、デバッグ コアのパラメーターを設定します。

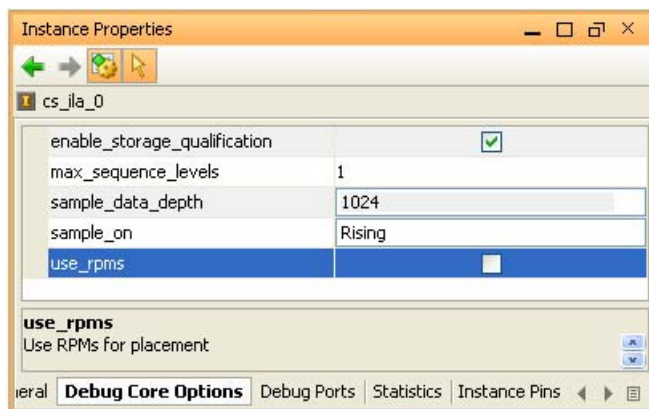


図 12-7：[Instance Properties] ビュー：デバッグ コア

デバッグ コアのインプリメンテーション

PlanAhead では、ChipScope Pro の ICON および ILA コアが最初はブラックボックスとして作成されます。これらのコアはマップ、配線を実行する前にインプリメントしておく必要があります。

ChipScope デバッグ コアのインプリメンテーションは、Flow Navigator または [Tools] メニューから [Run Implementation] を使用してインプリメンテーションを実行すると自動的に実行されますが、フロアプランまたはタイミング解析用に手動でインプリメントすることもできます。コアを手動でインプリメントするには、次のいずれかを実行します。

- [ChipScope] ビューのツールバー メニューで [Implement ChipScope Debug Cores] アイコンをクリックします。
- [ChipScope] ビューのポップアップ メニューで [Implement ChipScope Debug Cores] をクリックします。

保存されていない変更点がある場合は、デバッグ コアがインプリメントされる前にデザインを保存するかどうか尋ねるメッセージが表示されます。

各ブラック ボックス デバッグ コアに対しザイリンクスの CORE Generator™ がバッチ モードで起動します。この処理には、多少時間がかかる場合があります。この間、進捗状況を示すダイアログ ボックスが表示されます。デバッグ コアのインプリメンテーションが完了すると、デバッグ コアのブラック ボックスが処理され、生成されたインスタンスにアクセスできます。

プロセスのバックグラウンドへの移動

PlanAhead でデバッグ コアをインプリメントするために CORE Generator を起動すると、[Implement Debug Core] ダイアログ ボックス (419 ページの図 12-8) が開き、プロセスがバックグラウンドで実行されます。

プロセスがバックグラウンドで実行されると、バックグラウンド タスクを実行させたまま、レポートを表示したり、デザイン ファイルを開いたりといった別の機能が実行できます。この間、前の run を確認したり、レポートを表示したりして、時間を効率的に利用できます。ただし、[Tcl Console] はブロックされるので、Tcl コマンドを使用したり、開いている別のデザインに切り替えるような Tcl コマンドを必要とするタスクは実行できません。

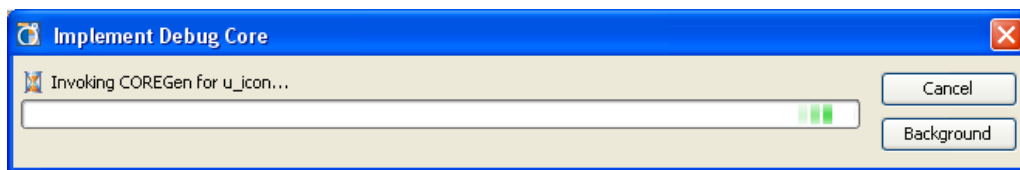


図 12-8 : [Implement Debug Core] ダイアログ ボックス : バックグラウンド プロセス

ChipScope Analyzer のネット接続の CDC ファイルの使用

ChipScope Analyzer の CDC ファイルはデザイン インプリメンテーションが完了すると自動的に生成されます。[ChipScope] ビューで [Export Debug Net Names] を使用し CDC ファイルを手動でエクスポートすることもできます。この CDC ファイルを ChipScope Analyzer にインポートし、ILA コア データおよびトリガ ポートのネット名を自動的に設定することができます。

デバッグ コアを含むデザインのインプリメンテーション

ChipScope デバッグ コアが生成され接続されたら、標準の PlanAhead インプリメンテーション フローを実行してデバイスのビットストリームを生成できます。

Flow Navigator または [Tools] メニューから [Implement] をクリックしてインプリメンテーションフローを開始します。

ChipScope Pro Analyzer の起動

ChipScope Pro Analyzer がインストールされている場合は、[Generate Bitstream] コマンドが実行されているどのインプリメンテーション デザインでも PlanAhead から直接 ChipScope Pro Analyzer を起動できます。

ChipScope Pro Analyzer を起動するには、次のいずれかの操作を実行します。

- Flow Navigator で [Program and Debug] → [ChipScope Analyzer] をクリック
- メイン メニューから [Tools] → [Analysis] → [ChipScope Analyzer] をクリック
- [ChipScope] ビューで右クリックして [Launch ChipScope Analyzer] をクリック

ビットストリーム ファイル (BIT ファイル) および CDC ファイルが自動的に ChipScope Pro Analyzer で読み込まれます。ChipScope Pro Analyzer の詳細は、http://japan.xilinx.com/support/documentation/dt_chipscopepro.htm を参照してください。または、付録 E の「ChipScope 資料」に示す資料を参照してください。

iMPACT の起動

iMPACT ツールでは、デバイス コンフィギュレーションとファイルの生成を実行できます。

- デバイス コンフィギュレーションでは、ザイリンクス ケーブル (パラレル ケーブル IV、プラットフォーム ケーブル USB またはプラットフォーム ケーブル USB II) を使用してザイリンクス FPGA および PROM を直接コンフィギュレーションできます。
- バウンダリスキャン モードで実行すると、ザイリンクス FPGA、CPLD、PROM をコンフィギュレーションまたはプログラムできます。
- ファイル生成では、System ACE™ インターフェイス ファイル、CF、PROM、SVF、STAPL および XSVF などのプログラム ファイルを作成できます。

iMPACT では、次も実行できます。

- デザインのコンフィギュレーション データのリードバックおよび検証
- コンフィギュレーション エラーのデバッグ
- SVF および XSVF ファイルの実行

iMPACT は、[Generate Bitstream] コマンドが実行されているどのインプリメンテーション デザインでも PlanAhead から直接起動できます。iMPACT を起動するには、Flow Navigator で [iMPACT] をクリックします。

PlanAhead から iMPACT を起動した場合、BIT ビットストリーム ファイルが自動的に iMPACT に読み込まれます。iMPACT の詳細は、[iMPACT ヘルプ](#)を参照してください。

階層デザイン手法の使用

PlanAhead™ ツールでは、階層デザインがサポートされます。

プロジェクトの開始前に階層デザインを使用するかどうか決定しておいてください。

- 階層デザイン手法を使用するかどうか決定する際、デザイン パーティションや RTL コードを考慮する必要があります。
- デザイン サイクル後期にこの手法を採用して、タイミング クロージャを達成したりランタイムを短縮しようとする、結果が最適にならない可能性があります。

詳細は、付録 E「その他のリソース」に示す次の資料を参照してください。

- 『階層デザイン手法ガイド』(UG748)
- 『パーシャル リコンフィギュレーション ユーザー ガイド』(UG702)

パーティションの使用

階層機能とは、「パーティション」というデザインの階層バウンダリを設定して管理する機能です。

これらのバウンダリを設定することにより、合成やインプリメンテーションでバウンダリを超えてロジックを最適化することがなくなるので、再利用するロジックのみを残しておくことができます。

パーティションを効率的に実行するには、効率的なロジック デザインおよび知識が必要です。

PlanAhead では、次のパーティション機能がサポートされます。

- RTL プロジェクトを使用した XST (Xilinx® Synthesis Technology) のインクリメンタル フロー
- パーティションを作成した階層とは別の階層にインポート
- パーティション内の AREA_GROUPS
- 合成およびインプリメンテーションでブラック ボックスをサポート
- パーティション ポートの定数、未接続の入力および出力を境界で最適化

注記： ISE® コマンド ライン フローのネスト化されたパーティションの機能は、PlanAhead ではサポートされません。

パーティションを含むデザインがインプリメントされると、その結果を今後のインプリメンテーション用にエクスポートできます。パーティションの定義および動作は、`xpartitions.pxml` という XML ファイルに保存されます。ISE Design Suite ツールでは、そのファイルの `run` ディレクトリが検索され、実行されます。パーティションは [Implement] や [Import] など、指定したパーティションの「アクション」と共に定義されます。`xpartitions.pxml` ファイルの使用方法和構文については、階層デザインのマニュアルを参照してください。

パーティションの設定

パーティションは、デザインの階層デザイン インスタンスに設定できます。階層デザイン フローでは、各パーティションを別々に合成する必要があるため、パーティションごとにネットリストが生成されます。これにより、パーティション ネットリストは個別に保存され、再利用できるようになります。

パーティションを設定するには、次の手順に従います。

1. [Netlist] ビューでパーティションを設定するモジュール インスタンスを選択します。
2. ポップアップ メニューから [Set Partition] コマンドをクリックします。
3. 図 13-1 のように、パーティションされたモジュールをハイライトする新しいアイコンが [Netlist] ビューに表示され、そのパーティションが [Instance Properties] ビューの [Partitions] タブに表示されます。このタブには、パーティションのディレクトリとプロモートされた日付が表示されます。

[Partition Properties] ビューの詳細は、『階層デザイン設計手法ガイド』(UG748) を参照してください。

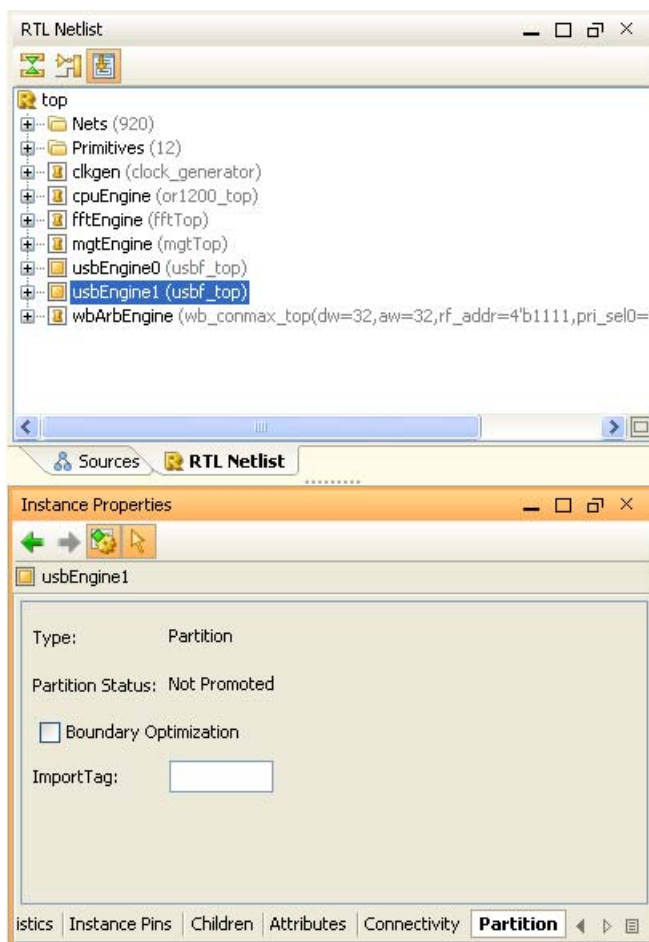


図 13-1：[Netlist] ビューでのパーティションの設定

プロジェクトでパーティションが定義されると、最上位デザインも自動的にパーティションになります。この最上位パーティションは、その他のパーティション同様に扱い、デザインの繰り返し中に必要に応じてインプリメントまたはインポートしてください。

パーティションの合成/インプリメンテーション run の設定

デザインでパーティションを定義すると、自動的に XST インクリメンタル フローがイネーブルにされます。このインクリメンタル フローでは、最上位パーティションを含む各パーティションに対して個別のネットリスト ファイル (NGC) がインプリメントされ、その結果が合成済みデザインにインポートされます。合成の詳細は、第 6 章「デザインの合成」を参照してください。

後続のデザイン繰り返しでは、合成およびインプリメンテーション中に実行する操作をパーティションごとに設定できます。パーティションを変更した場合は合成またはインプリメンテーションを再実行し、変更しなかった場合は特定のディレクトリからインポートできます。

合成またはインプリメンテーションでのパーティションに対する操作を設定するには、次の手順に従います。

1. Flow Navigator の [Project Manager] → [Specify Partitions] をクリックします (図 13-2)。これにより、[Specify Partitions] ダイアログ ボックスが開きます。

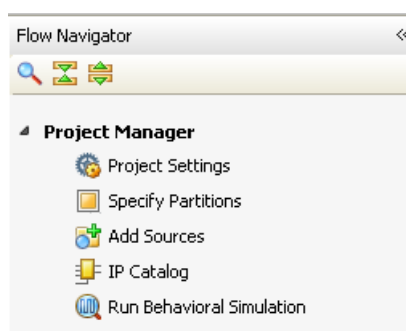


図 13-2 : 合成およびインプリメンテーションでのパーティションの設定

[Specify Partition] ダイアログ ボックス (図 13-3) では、アクティブな合成およびインプリメンテーション run に対してデザインに含まれている各パーティションでの操作を指定できます。プロジェクトで複数の合成およびインプリメンテーション run が定義されている場合は、[Specify Partitions] コマンドを使用する前にアクティブ run を指定する必要があります。

パーティション データを指定したディレクトリからインポートするか、合成またはインプリメンテーションでパーティションを再構築できます。

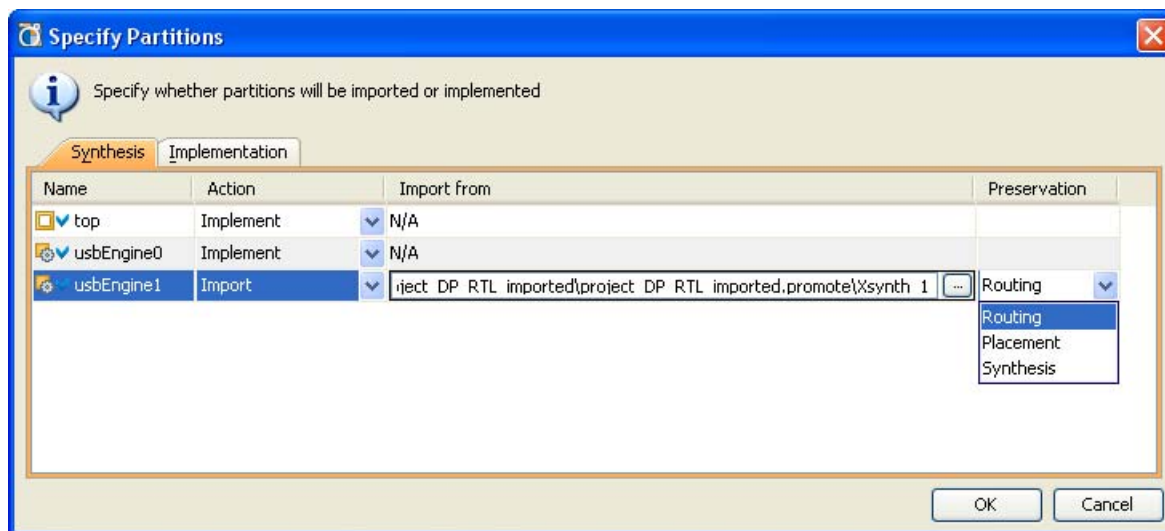


図 13-3：パーティションの指定

2. パーティションでの操作 (Action) は、次のように設定します。

- **[Implement]**：初めてパーティションを合成およびインプリメントするか、デザイン変更に基づいてパーティションのインプリメントをし直す場合に設定します。デザインを初めてインプリメントする場合は、インプリメントされたパーティションをインポートするプロモート ディレクトリがないので、すべてのパーティションを **[Implement]** に設定する必要があります。
- **[Import]**：パーティション データをインポートします。
 - **[Import From]**：インポートするパーティションのソース ディレクトリを指定します。これは、特定のパーティションに複数のインプリメンテーションがあるときに、インポートする結果を指定できるので便利です。
 - **[Preservation]**：パーティションをインポートする際に保持するパーティション データのレベルを示します。パーティションの合成結果をインポートしてブロックを再インプリメントするか (**[Synthesis]**)、配線せずに配置するか (**[Placement]**)、またはインプリメントしたパーティションの配線およびタイミングを保持するか (**[Routing]**) を選択できます。

3. **[OK]** または **[Cancel]** をクリックしてダイアログ ボックスを閉じます。

パーティション設定は、**[Run Properties]** ビューの **[Partitions]** タブから合成 run またはインプリメンテーション run 用に指定することもできます (図 13-4)。

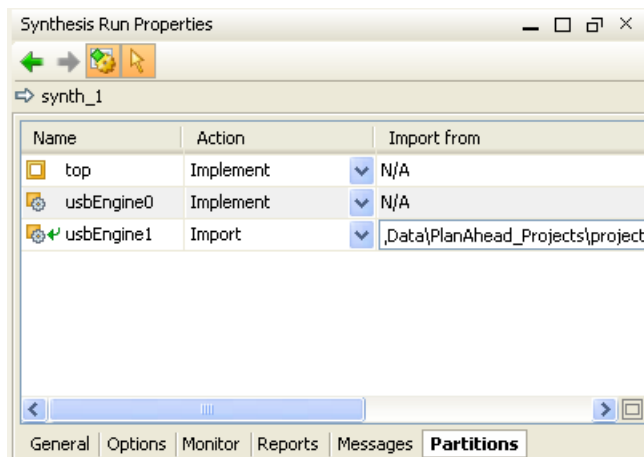


図 13-4 : [Run Properties] ビューの [Partitions] タブ

合成またはインプリメンテーションを起動すると、[Reports] ビューに結果が表示されます。run でパーティションに対して実行された操作がレポートに表示されます。425 ページの図 13-5 にパーティションを含むインプリメンテーションの NGDBuild レポートを示します。

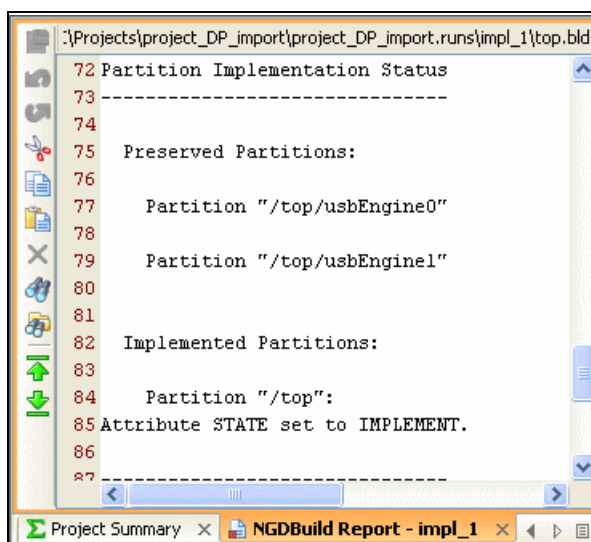


図 13-5 : NGDBuild レポートに含まれるパーティション操作

パーティションのプロモート

合成またはインプリメンテーションで満足の結果が得られた場合、このデータをディレクトリまたはレポジトリにコピーすることで、今後のデザイン繰り返しインポートできます。パーティション結果を保持することをパーティションのプロモートと呼びます。パーティションは、問題なく合成/インプリメントされた run からのみプロモートできます。

パーティションをプロモートするには、次の手順に従います。

1. Flow Navigator で [Promote Partitions] をクリックします。

または、メイン メニューから [Flow] → [Promote Partitions] をクリックします。



426 ページの図 13-6 に示すような [Promote Partition] ダイアログ ボックスが開きます。

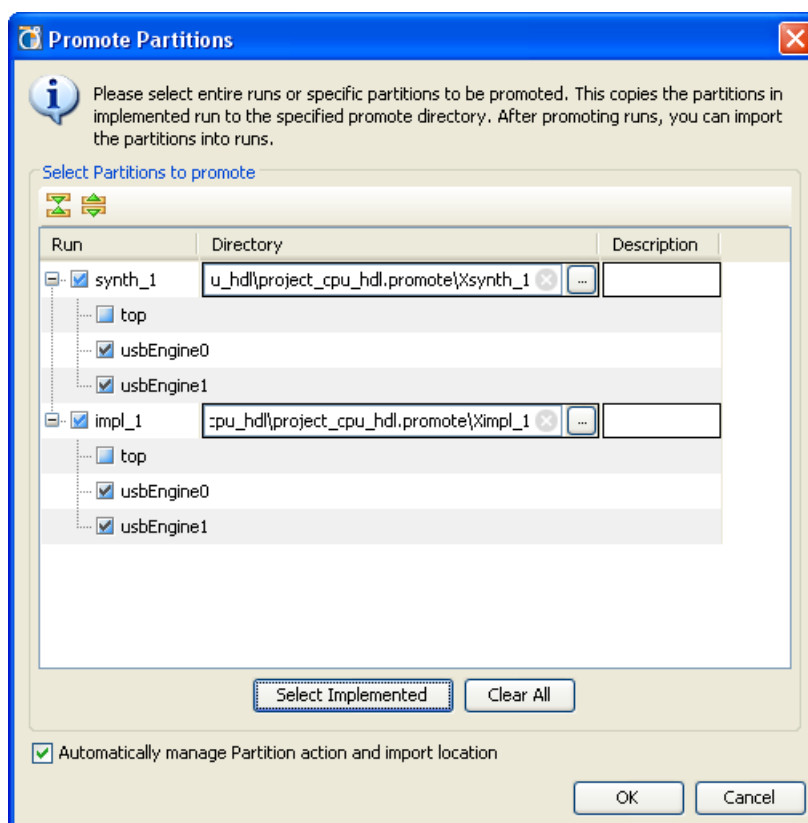


図 13-6：パーティションのプロモート

2. プロモートされたパーティションを保存するディレクトリを指定します。

このディレクトリ/データ レポジトリには、パーティションおよびそれに伴う必須ファイルすべてが保存されます。このディレクトリは、今後のデザイン繰り返しでインプリメントし直すために使用するのではなく、パーティションをインポートする際の参照として使用されます。

注記：すべてのパーティションを 1 つのディレクトリにプロモートするようにしてください。この方法を強化するため、以前にプロモートしたディレクトリを上書きするかどうか尋ねるメッセージが表示されます。詳細は、付録 E 「その他のリソース」 に示す『階層デザイン手法ガイド』(UG748) を参照してください。

3. プロモートするパーティションを選択します。

デフォルトでは、デザインに含まれている最上位パーティション以外のパーティションすべてが選択されています。[Select Implemented] ボタンをクリックし、最上位パーティションも含めたすべてのインプリメント済みパーティションを選択します。

4. [Description] フィールドに詳細を入力します (オプション)。

5. [Automatically manage Partition action and import location] をオンにすると、すべてのプロモート済みパーティションに対するアクションが [Import] に設定されます。パーティションをプロモートすると、[Specify Partitions] ダイアログ ボックスで今後実行する合成 run またはインプリメンテーション run に対してのパーティションのデフォルトの操作が [Import] に設定されます。詳細は、423 ページの「パーティションの合成/インプリメンテーション run の設定」を参照してください。

このチェック ボックスがオフの場合は、パーティションに対するアクションを手動で設定する必要があります。

6. [OK] をクリックします。

[Promoted Partitions] ビューが開き、結果がレポートされます。

[Promoted Partitions] ビューの使用

[Promote Partitions] ビューは、[Promote Partitions] コマンドを実行すると表示されます。

[Promoted Partitions] ビューを開くには、[Window] → [Promoted Partitions] をクリックします。

図 13-7 は、[Promoted Partitions] ビューを示しています。

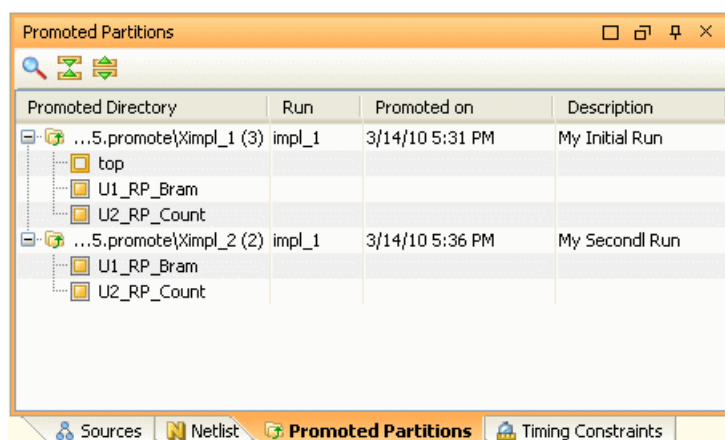


図 13-7 : [Promoted Partitions] ビュー

[Promoted Partitions] ビューには、プロモートされたパーティションのディレクトリ、ソースの run、プロモートされた日時、詳細などがツリー形式で表示されます。

パーティションを選択すると、[Promoted Run Properties] ビューが開きます。必要であれば詳細フィールドを変更できます。

プロモートされたパーティションの削除

[Promoted Partitions] ビューではプロモート済みのパーティションを削除できます。そのパーティションのすべての run データも含めて、プロモート済みパーティションを含むディレクトリ全体も削除されます。

プロモート済みパーティションを削除するには、[Promoted Partitions] ビューでそのディレクトリを右クリックし、[Delete] をクリックします。

パーティションを削除すると、[Specify Partitions] ダイアログ ボックスで今後実行する合成 run またはインプリメンテーション run に対してのパーティションのデフォルトの操作が [Implement] に設定されます。

パーティションのインポート

パーティションをプロモートしたら、それらのパーティションをその後の run にインポートできます。

- 合成の場合、パーティションをインポートすると、インポートされた NGC ファイルが現在の結果ディレクトリにコピーされ、同一の合成結果が繰り返し使用されるようになります。

- インプリメンテーションの場合、パーティションをインポートすると、残りのデザインをインプリメントする前に、パーティションの配置配線が最上位デザインにインポートされ、同じ結果が維持されます。

ソースのアップデート

ソースのアップデート

RTL ソースに変更を加えると、[RTL] および [Netlist] ビュー、完了した合成またはインプリメンテーション **run** が最新の状態ではなくなります。この場合、[RTL] および [Netlist] ビューを開き直すか、ファイルを再び読み込んでロジックの変更を認識させる必要があります。

デザインを修正すると、そのアップデートされたネットリストが自動的に検出されるか (リモートソースを使用している場合)、プロジェクトでアップデートされ、既存ネットリストがアップデートされます。ソース ファイルがアップデートされたら、[Synthesized Design] を開くか読み込み直して、ロジックの変更を適用します。詳細は、[201 ページの「デザイン ソース ファイルの管理」](#)を参照してください。

ロジック アップデートに基づいたパーティション操作の設定

[423 ページの「パーティションの合成/インプリメンテーション run の設定」](#)の記述のように、各 **run** に対してパーティションでの操作を設定する必要があります。

- アップデートされたネットリストを含むパーティションは、変更を含めてロジックが再インプリメントされるよう、必ず [Implement] に設定します。
- 変更のないパーティションは、[Import] に設定してパーティションを保持するか、[Implement] に設定して必要に応じて再インプリメントすることができます。

関連する手法

チーム デザイン

PlanAhead には、複数のプロジェクトを作成および管理できる環境が含まれるので、開発をそれぞれ同時に進めることができます。このチーム デザイン フローでは、XST (Xilinx Synthesis Technology) のインクリメンタル合成とインプリメンテーションでのブラック ボックスがサポートされます。

詳細は、[付録 E 「その他のリソース」](#)に示す『チーム デザインで生産性を向上』(WP388) を参照してください。

デザイン保持

デザイン保持手法とパーティション機能を共に使用すると、1 つの **run** のパーティションの配置配線を固定、保持して次の **run** で使用できます。このインクリメンタル デザイン手法により、より一貫した結果を得ることができ、検証時間およびデザインのクロージャ時間も削減されます。

詳細は、[付録 E 「その他のリソース」](#)に示す『デザインの保持チュートリアル：PlanAhead デザイン ツール』(UG747) を参照してください。

パーシャル リコンフィギュレーション

PlanAhead には、パーシャル リコンフィギュレーション プロジェクトを設定、インプリメント、管理する環境が含まれています。PlanAhead では、ISE Design Suite インプリメンテーション ツールに含まれるパーティションとパーシャル リコンフィギュレーション機能が使用されます。

詳細は、付録 E 「その他のリソース」に示す『パーシャル リコンフィギュレーション チュートリアル : PlanAhead デザイン ツール』(UG743) を参照してください。

Tcl およびバッチ スクリプト

この章には、Tcl リソースへの包括的なリファレンスは含まれませんが、Tcl リソースへのリファレンス、PlanAhead 環境での一般的な Tcl の機能に関する説明が含まれます。

Tcl (Tool Command Language) は PlanAhead™ 環境に統合されたスクリプト言語です。Tcl とは、デザイン制約に使用される半導体業界標準の言語のことです。

Tcl を使用することで、自動スクリプトだけでなく、デザイン ツールに対するインタラクティブな要求が実行可能になっています。Tcl にはデザイン データベースから、ツール、デザイン設定、ステートなどに関する情報をインタラクティブに取り出す機能があります。

次のセクションでは、PlanAhead に関する基本的な Tcl 機能について説明します。各コマンドの詳細は、それぞれコマンドを実行してヘルプ情報を参照してください。付録 E 「その他のリソース」に示す『Tcl コマンド リファレンス ガイド』(UG789) には、PlanAhead Tcl コマンドの一覧およびその説明が記載されています。

Tcl ジャーナル ファイル

PlanAhead を起動すると、デザイン セッション中に実行されるコマンドおよび操作の記録が PlanAhead.log ファイルに書き込まれます。また、PlanAhead.jou という、セッション中に実行された Tcl コマンドのみを記したジャーナル ファイルも書き出され、このファイルを使用して新しい Tcl スクリプトを作成することができます。

注記：このファイルのバックアップ バージョンが planahead.jou_backup で、1 つ前のセッションで実行した Tcl コマンドの詳細が保存されています。

これらのファイルの保存場所に関する情報は、付録 A 「PlanAhead の入力ファイルおよび出力ファイル」を参照してください。

Tcl ヘルプ

Tcl の help コマンドを使用すると、サポートされる Tcl コマンドの概要が表示されます。

- **help** : Tcl コマンド カテゴリのリストが表示されます。
help
コマンド カテゴリは、File I/O などの特定のファンクションを実行するコマンド グループです。
- **help -category <category>** : 特定のカテゴリで見つかったコマンドのリストを表示します。
help -category object
この例では、オブジェクトを処理する Tcl コマンドのリストが表示されます。
- **help <pattern>** : 特定の検索パターンに一致したコマンドのリストを表示します。このコマンドを使用すると、コマンド グループから特定のコマンドを迅速に見つけることができます。

help get_*

この例では、「get_」で始まる Tcl コマンドのリストが表示されます。

- **help <command>** : 特定のコマンドに関連する詳細情報を表示します。

help get_cells

この例では、get_cells コマンドの詳細が表示されます。

- **help -args <command>** : 指定してコマンドの構文、各オプションの簡単な説明を含めた簡略版ヘルプ テキストを表示します。

help -args get_cells

- **help -syntax <command>** : 指定したコマンドのコマンド構文をレポートします。

help -syntax get_cells

help は、使用されるコマンドに対するオプションとして付けることもできます。これは、コマンドを入力する際に、そのコマンドの構文やオプションのヘルプが必要な場合に使用します。次のいずれかのように使用できます。

```
report_power -help syntax
report_power -help args
```

[Tcl Console] ビュー

PlanAhead の GUI 環境には、操作が実行されると Tcl コマンドを反映するエリアが含まれ、実行したタスクの結果を元に情報、警告、エラー メッセージが表示されます。[Tcl Console] ビューは、PlanAhead 環境の一番下にあり、GUI の幅に固定されています。[Tcl Console] ビューの右側 (スクロールバーの右) には、警告やエラーが色付きのインジケータで示されます。警告があれば黄色、エラーの場合は赤色で表示されます。これにより、警告やエラーを見つけることができます。図 14-1 に環境内に表示される [Tcl Console] ビューを示します。

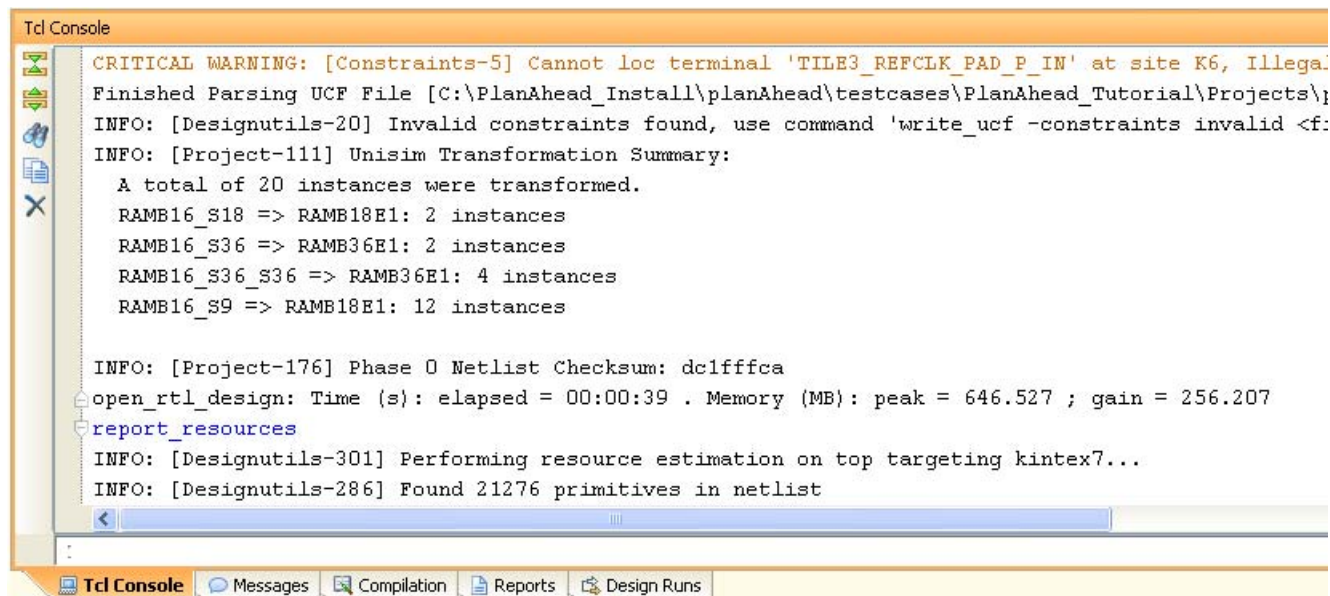


図 14-1 : [Tcl Console] ビュー

Tcl コンソールでのキーストロークのマッピング⁽¹⁾

Tcl コンソールではネイティブ Tcl がサポートされるので、`editline` を使用する標準 `tclsh` に対するキーの組み合わせとキーストローク マッピングは、ホーム ディレクトリ (\$HOME) の `.editrc` ファイルで定義されます。`.editrc` ファイルでは、`editline` ライブラリで使用されるさまざまな設定が定義されます。

注記：`.editrc` ファイルは、ソフトウェアをインストールしても、デフォルトでは含まれません。このファイルは Tcl コンソールのキー組み合わせをカスタマイズするために、ユーザーが作成する必要があります。

ホーム ディレクトリで `.editrc` ファイルを編集すると、キーストロークの組み合わせを特定の編集コマンドにカスタマイズできます。`.editrc` ファイルのフォーマットは、次のとおりです。

```
bind <key> <command>
```

説明：

- `bind` : コマンドへキーをまとめるキーワード
- `<key>` : まとめるキーまたはキーの組み合わせ
- `<command>` : キーをまとめる編集コマンドまたはマクロ

たとえば、次の `.editrc` ファイルは、チルダ キー (~) を Emacs の前の文字を削除するコマンドにマッピングし、`Ctrl-R` キー の組み合わせ (^r) を Emacs のインクリメンタル逆検索コマンドにマッピングしています。

```
bind "~" em-delete-prev-char
bind "^r" em-inc-search-prev
```

`.editrc` ファイルの詳細は、次を参照してください。

<http://www.freebsd.org/>

PlanAhead ソフトウェアの起動

PlanAhead には、主に次の 3 つの操作モードがあります。

- GUI モード (デフォルト)
- Tcl コマンド ライン オプションからの PlanAhead 実行ファイルの起動 (バッチ モード)
- Tcl シェル モード

次のセクションでは、バッチ モードと Tcl シェル モードについて説明します。

バッチ モード

バッチ モードは、スクリプトを実行してツールをシャットダウンします。PlanAhead をバッチ モードで起動するには、次を入力します。

```
planAhead -mode batch -source script_name.tcl
```

1. `.editrc` ファイルは、Linux OS のキーストロークをマッピングするためだけに使用できます。

Tcl シェル モード

Tcl モードは Windows のコマンド シェルや Linux シェルのようなシェルを起動します。これは GUI を起動しないインタラクティブなシェルセッションです。

スクリプトは `-source` オプションで渡すことができます。これにより、スクリプトが実行された後、Tcl コマンドを手動で入力できるインタラクティブなシェルに制御が渡されます。Tcl シェル モードで PlanAhead を起動するには、次を入力します。

```
planAhead -mode tcl -source script_name.tcl
```

Tcl 初期ファイル

PlanAhead では、起動時に次の 2 つの場所にある Tcl 初期化スクリプトが検索されます。

1. `<installdir>/planAhead/scripts/init.tcl`
2. `<userdir>/Xilinx/PlanAhead/init.tcl`

説明:

- `<installdir>`: PlanAhead のインストールディレクトリ
- `<userdir>`: ホーム ディレクトリ
 - Windows の場合: `%APPDATA%/Xilinx/PlanAhead/init.tcl`
 - Linux の場合: `$HOME/.Xilinx/PlanAhead/init.tcl`

`init.tcl` スクリプトがいずれかの場所で見つかった場合は、インストール ディレクトリ、ホーム ディレクトリの順番でこのファイルが使用されます。

インストール ディレクトリにある `init.tcl` ファイルを使用すると、会社またはデザイン グループがすべてのユーザーに対して共通の初期化スクリプトをサポートできます。このディレクトリから PlanAhead ソフトウェアを起動すると、`init.tcl` スクリプトが使用されます。

ホーム ディレクトリにある `init.tcl` を使用すると、ユーザーがそれぞれコマンドを追加したり、デザイン要件を満たすようにソフトウェア インストール ディレクトリに含まれるコマンドに書き出すことができます。

この `init.tcl` スクリプトは標準の Tcl コマンド ファイルで、PlanAhead でサポートされるどの Tcl コマンドも含めることができます。次を追加すると、`init.tcl` ファイル内から別の Tcl スクリプト ファイルに対して `source` コマンドを実行することもできます。

```
source <path_to_file>/<file_name>.tcl
```

一般的な Tcl 構文のガイドライン

Tcl では、OS に関係なく Linux の表記規則 (/) が使用されます。

次のセクションでは、PlanAhead で Tcl を使用する際の一般的な構文ガイドラインについて説明しています。

Tcl スクリプトの実行

Tcl スクリプトはコマンド ライン オプションの 1 つか、GUI のどちらかから実行できます。PlanAhead の GUI で Tcl スクリプトを実行するには、[Tools] → [Run Tcl Script] をクリックします。

次のコマンド ライン オプションを使用すると、Tcl スクリプトを `source` コマンドで実行できます。

```
source file_name
```

Tcl スクリプトを GUI から実行すると、進捗バーのダイアログ ボックスが表示され、スクリプトが終了するまですべての GUI 操作がブロックされます。

現在のところ、ランタイム中にスクリプトの実行を一時停止する方法はないので、標準的な OS のプロセス停止方法で強制終了するしかありません。この場合、最後に保存した後の作業が失われます。

Tcl コンソールで `help source` と入力すると、`source` コマンドに関する詳細が表示されます。

Tcl の eval コマンドの使用

Tcl コマンドを実行する場合、Tcl コマンドで使用可能または必要とされるコマンド ライン オプションの一部を変数で置き換えることができます。ただし、コマンドの一部として Tcl 変数を使用してコマンド ラインを評価するには、Tcl の `eval` コマンドを使用する必要があります。

たとえば、`help` コマンドの場合、数多くのコマンド カテゴリの 1 つと一緒にオプションとして `-category` を使用できます。

```
help -category ipflow
```

このコマンド カテゴリを保持する変数を定義できます。

```
set cat "ipflow"
```

これで、Tcl コマンドの内容で変数を評価できます。

```
eval help -category $cat
```

または

```
set cat "-category ipflow"
eval help $cat
```

クォーテーション マーク " の代わりに波括弧 { } を使用しても、同じ結果になります。

```
set runblocksOptDesignOpts { -sweep -retarget -propconst -remap }
eval opt_design $runblocksOptDesignOpts
```

Tcl コンソールで `help eval` と入力すると、`eval` コマンドに関する詳細が表示されます。

一般的な構文構造

PlanAhead の Tcl コマンドの一般的な構造は、次のとおりです。

```
command [optional_parameters] required_parameters
```

コマンド構文は、アンダースコア (`_`) で区切られた「動詞 - 名詞」および「動詞 - 形容詞 - 名詞」の構文になります。

コマンドは、関連するコマンド同士に同じ接頭辞が付けられて、グループ分けされています。

- クエリを実行するコマンドには、通常 `get_` が接頭辞として付いています。
- 値やパラメーターを設定するコマンドには、通常 `set_` が接頭辞として付いています。
- レポートを生成するコマンドには、通常 `report_` が接頭辞として付いています。

これらのコマンドは、グローバル名前空間に属しており、コマンドに付属するサブコマンド也没有。

構文例

次は、**help get_cells** コマンドを実行した例です。

```

help get_cells
get_cells

Description:
Get a list of cells in the current design

Syntax:
get_cells [-hsc <arg>] [-hierarchical] [-regexp] [-nocase]
[-filter <arg>] [-of_objects <args>] [-match_style <arg>]
[-quiet] [-verbose] [<patterns>]

Returns:
list of cell objects

Usage:

```

Name	Optional	Default	Description
-hsc	yes	/	Hierarchy separator
-hierarchical	yes	false	Search level-by-level in current instance
-regexp	yes	false	Patterns are full regular expressions
-nocase	yes	false	Perform case-insensitive matching
-filter	yes		Filter list with expression
-of_objects	yes		Get cells of these pins or nets
-match_style	yes	sdcc	Style of pattern matching
-quiet	yes	false	Ignore command errors
-verbose	yes		Suspend message limits during command execution
patterns	yes	*	Match cell names against patterns

不明コマンド

Tcl には、通常サポートされるビルトイン コマンド、Tcl インタプリターに渡される PlanAhead 特有のコマンド、およびユーザー定義のプロシージャのリストが含まれます。

これらの既知のコマンドに含まれないコマンドは OS に送信され、**exec** コマンドからシェルで実行されます。これにより、OS 特有の可能性のあるシェル コマンドを実行できます。シェル コマンドがない場合、コマンドが見つからなかったことを示すエラー メッセージが表示されます。

リターン コード

Tcl コマンドの中には、オブジェクトのリストなどのリターン値が出力されるものがあります。それ以外のコマンドの場合、処理は実行されても、ユーザーが直接利用できるような値を必ず返すとは限りません。Tcl インターフェイスを統合したツールの一部には、コマンドでエラーのない場合は 0、エラーがある場合は 1 を返すものもあります。

Tcl コマンドまたはスクリプトのエラーを正しく処理できるよう、Tcl ビルトイン コマンドの `catch` を使用する必要があります。一般的には、`catch` コマンドと番号付き情報/警告/エラー メッセージに基づいて、Tcl スクリプトのフローで問題を評価します。

PlanAhead のすべての Tcl コマンドで完了時に `TCL_OK` または `TCL_ERROR` が返されます。また、PlanAhead では標準の Tcl メカニズムによりグローバル変数 `$errorinfo` が設定されます。

`$errorinfo` 変数を使用する場合は、Tcl コンソールでエラーがレポートされた後に次を入力します。

```
puts $errorinfo
```

これにより、エラーの詳細情報が表示されます。たとえば、次のコード例では Tcl スクリプト (`procs.tcl`) が使用されており、ユーザー定義の手順 (`load`) が実行されます。数行トランスクリプトメッセージが表示された後、5 行目にエラーが表示されます。

```
Line 1:PlanAhead % source procs.tcl
Line 2:PlanAhead% loads
Line 3:Found 180 driving FFs
Line 4:Processing pin a_reg_reg[1]/Q...
Line 5:ERROR:[HD-Tcl 53] Cannot specify '-patterns' with '-of_objects'.
Line 6:PlanAhead% puts $errorInfo
Line 7:ERROR:[HD-Tcl 53] Cannot specify '-patterns' with '-of_objects'.While executing
"get_ports -of objects $pin" (procedure "my_report" line 6) invoked from within procs.tcl
```

Tcl スクリプト ファイルに `puts $errorInfo` を含めてエラーが見つかったときに詳細を表示するようにしたり、Tcl コンソールでエラーが発生したときに必要に応じて「`puts $errorInfo`」と入力して特定のエラーの詳細を表示できます。

上の例では、6 行目に「`puts $errorInfo`」と入力することで、7 行目にエラーの詳細情報が表示されています。

ファースト クラスの Tcl オブジェクトとその関係

PlanAhead の Tcl コマンドからは、ネットリスト、デバイス、プロジェクトのオブジェクト モデルへ直接アクセスできます。これらオブジェクトは「ファーストクラス」と呼ばれ、単なる文字列記述以上のものであり、オブジェクト上で操作ができ、クエリを実行できることを意味します。例外もありますが、通常はオブジェクトとしてクエリを実行できます。これらのオブジェクトには、クエリ可能なプロパティが含まれ、ほかのオブジェクトを取得できる関係があります。

オブジェクト タイプと定義

PlanAhead には多くのオブジェクト タイプがありますが、ここでは基本的なタイプの定義と説明のみが記述されます。最も基本的で重要なオブジェクト タイプは、デザイン ネットリストのエンティティと関連しています。詳細は、次のセクションでそれぞれ説明します。

セル

セルは、ライブラリ モジュールまたはネットリスト内の階層モジュール/エンティティのいずれかのインスタンスです。これには、フリップフロップ、LUT、I/O バッファ、RAM、DSP のほか、ほかのセルのリストのラッパーである階層インスタンスが含まれます。

ピン

ピンはセル上の論理接続ポイントです。ピンにより、セルの内部ロジックが抽象化されて使用しやすくなります。ピンは、階層または下位のセル上に存在します。ピンには、クロック ピン、データ ピン、リセット ピン、フリップフロップの出力ピンなどが含まれます。

ポート

ポートは、ネットリストの最上位モジュールまたはエンティティのインターフェイスの論理接続のポイントです。ポートは通常 I/O パッドに接続され、FPGA デバイスの外側に接続されます。

ネット

ネットは、階層ピン、下位ピン、およびポートのコレクションです。階層デザインの場合、ネットには階層のレベルそれぞれに複数のセグメントを含めることができます。

クロック

クロックは、デザイン内のシーケンシャル ロジックへ伝搬される周期的な信号です。入力ポートで作成したり、DCM、PLL、MMCM などのクロック調整ブロック プリミティブで生成させ、プライマリ クロックにできます。クロックは UCF の TIMESPEC PERIOD 制約とほぼ同じで、スタティック タイミング解析アルゴリズムの基礎になっています。

オブジェクトのクエリ

ファーストクラス オブジェクトはすべて、通常次のように Tcl コマンドの `get` を使用してクエリできます。

```
get_object_type pattern
```

`pattern` が検索パターンの場合、必要であれば階層区切りを使用して指定します。オブジェクトは通常、階層の各レベルで指定されている文字列パターンに一致するものがクエリされます。検索パターンには次のようにワイルドカードも使用でき、オブジェクトを検索しやすくなっています。

```
get_cells */inst_1
```

このコマンドでは、最上位のすぐ下の階層で `inst_1` という名前のセルが検索されます。階層のすべてのレベルで繰り返し同じパターンを使用して検索する場合は、次の構文を使用してください。

```
get_cells -hierarchical inst_1
```

このコマンドでは、`inst_1` に一致するインスタンスがすべての階層レベルで検索されます。

構文の詳細は、次のコマンドでヘルプ情報を参照してください。

```
help get_cells
get_cells -help
```

オブジェクト プロパティ

オブジェクトには、クエリを実行できるプロパティが含まれます。プロパティ名はオブジェクト タイプによってそれぞれ異なります。オブジェクトの特有のプロパティをクエリするには、次のコマンドを使用します。

```
get_property <property_name> <object>
```

次の例はセル オブジェクトの `ref_name` プロパティをクエリするもので、特定のインスタンスがどの UniSim コンポーネントにマップされているかがわかります。

```
get_property ref_name [get_cell inst_1]
```

指定したオブジェクトに使用可能なプロパティすべてを表示する場合は、`report_property` コマンドを使用します。

```
report_property [get_cells inst_1]
```

表 14-1 に、特定のオプションで返されるプロパティを示します。

プロパティの中には、読み出し専用のものであれば、ユーザーの設定可能なものもあります。UCF や HDL でアノテート可能な属性にマップされるプロパティは、通常 Tcl コマンドの `set_property` でユーザーが定義できます。

```
set_property loc OLOGIC_X1Y27 [get_cell inst_1]
```

表 14-1：特定のオブジェクトで返されるプロパティ

キー	値	タイプ
bel	OLOGICE1.OUTFF	string
class	cell	string
iob	TRUE	string
is_blackbox	0	bool
is_partition	0	bool
is_primitive	1	bool
is_reconfigurable	0	bool
is_sequential	1	bool
ref_name	FD	string
LOC	OLOGIC_X1Y27	string
name	error	string
primitive_group	FD_LD	string
primitive_subgroup	flop	string

プロパティに基づいたフィルター処理

オブジェクト クエリの `get_*` コマンドには、そのオブジェクトのプロパティ値に基づいてクエリをフィルター処理するオプションがあります。このオプションは、非常に優れたオブジェクト クエリ コマンド機能です。たとえば、プリミティブ タイプ `FD` のセルをすべてクエリするには、次を入力します。

```
get_cells * -hierarchical -filter {ref_name == FD}
```

また、`==` を使用すると、文字列パターンでフィルターできます。たとえば、デザインに含まれるすべてのフリップフロップ タイプをクエリするには、次を入力します。

```
get_cells * -hierarchical -filter {ref_name =~ FD*}
```

OR (`||`) や AND (`&&`) を使用すると、複数のフィルター プロパティを組み合わせで検索できます。次の例では、デザイン内のすべてのセルから、フリップフロップ タイプで配置済みロケーション制約が設定されているものをクエリできます。

```
get_cells * -hierarchical -filter {ref_name =~ FD* && loc != ""}
```

注記：この例では、フィルター オプションの値が `"` ではなく、`{ }` で囲まれています。これはインタープリターによるコマンド変換を回避する標準的な Tcl 構文で、これにより `loc` プロパティに空の文字列を渡すことができます。

オブジェクトのリスト

複数のオブジェクトを返すコマンドは、通常ネイティブ Tcl リストのようなコンテナを返します。この PlanAhead の機能により、多数の Tcl オブジェクト処理を大幅に最適化できるので、Tcl ビルトイン コマンドの `foreach` で処理される `foreach_in_collection` のような特殊な反復コマンドを必要としません。

リストが大きい場合、ログ ファイルと GUI の Tcl コンソールでの表示が少し異なります。通常、`get *` コマンドの結果に対して Tcl 変数を設定すると、リスト全体がコンソールとログ ファイルに表示されますが、リストが大きいと、バッファのメモリ オーバーフローを回避するために、リストの一部省略されます。

コンソールとログ ファイルでリストの省略があると、最後のエレメントが「...」と表示されます。次は、1 つのセルをクエリした場合と、デザインに含まれるすべてのセルをクエリした場合 (リストが大きい場合) の例です。

```
%get_cells inst_1
inst_1
%get_cells * -hierarchical
XST_VCC XST_GND error readIngressFifo wbDataForInputReg fifoSelect_0
fifoSelect_1 fifoSelect_2 fifoSelect_3 ...
%set x [get_cells * -hierarchical]
XST_VCC XST_GND error readIngressFifo wbDataForInputReg fifoSelect_0
fifoSelect_1 fifoSelect_2 fifoSelect_3 ...
%lindex $x end
bftClk_BUFPG/bufg
%llength $x
4454
```

この例では、4000 セルすべては表示されておらず、「...」で省略されていますが、実際には、リストの最後のエレメントは Tcl 変数に正しく含まれています。

オブジェクトの関係

関連するオブジェクトは、`get *` コマンドに `-of_objects` オプションを付けるとクエリできます。たとえば、あるセル ロジックに接続されたピンのリストを取得するには、次を入力します。

```
get_pins -of_objects [get_cells inst_1]
```

441 ページの図 14-2 は、PlanAhead のオブジェクト タイプとその関係を示す図で、オブジェクト間の矢印ではその `get_*` コマンドで `-of_objects` オプションを使用して、論理接続に関係なく接続されたオブジェクトへの Tcl リファレンスを取得できることを示しています。

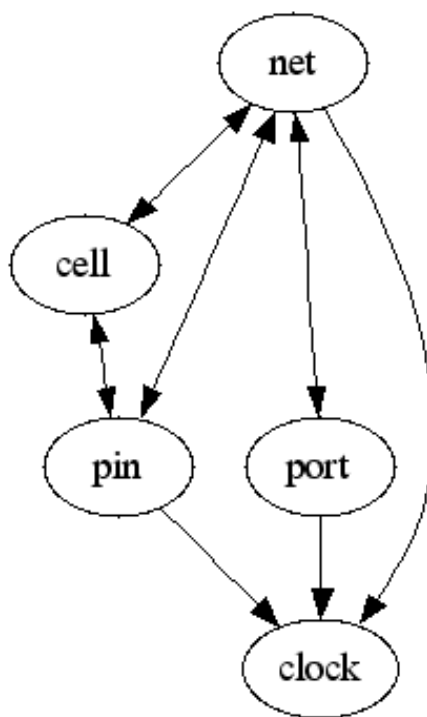


図 14-2 : PlanAhead のオブジェクトの関係

エラー、警告、クリティカル警告、および情報メッセージ

各コマンドの結果を示すメッセージはログ ファイルと、GUI を使用中であればコンソールにも表示されます。これらのメッセージには識別しやすいように番号が付いています。ログ ファイルでは INFO、WARNING、CRITICAL_WARNING、ERROR と最初に表示され、サブシステム識別子や独自の番号が付きます。

次は、タイミング ライブラリを読み込んだ後に表示される INFO メッセージの例です。

INFO:[HD-LIB 1] Done reading timing library

このようなメッセージにより、ログ ファイルで特定の問題が検出しやすくなり、コマンド実行中の動作内容が理解しやすくなります。

通常、Tcl スクリプトからの Tcl コマンドでエラーが発生すると、続くコマンドの実行は一時停止されます。これは、回復不可能なエラー状況を避けるためです。これらのエラー状況を回避して続行させるため Tcl ビルトイン コマンドがあります。一般的な Tcl メカニズムを使用したエラー処理方法については、`catch` コマンドを使用して Tcl リファレンスのいずれかを参照してください。

Tcl リファレンス

次は、推奨される Tcl リファレンスです。

Tcl Developer Xchange

Tcl リファレンス資料は、インターネットから入手できます。ザイリンクスでは、Tcl のオープンソース ベースを管理している Tcl Developer Xchange サイトを推奨しています。

<http://www.tcl.tk>

入門チュートリアルは、次から入手できます。

<http://www.tcl.tk/man/tcl/tutorial/tcltutorial.html>

SDC について

SDC (Synopsys Design Constraints) は、特にタイミング解析のために設計意図とツール間をつなぐ業界標準です。SDC 仕様のリファレンス コピーは、次の Synopsys のサイトから登録をすると入手できます。

<http://www.synopsys.com/Community/Interoperability/Pages/TapinSDC.aspx>

使用可能な Tcl マニュアル

Tcl のリファレンス マニュアルは、各自でご購入ください。

Project Navigator からの PlanAhead の使用

PlanAhead™ ツールは、ISE® デザイン ツールに統合されているので、PlanAhead ツール内で FPGA デザイン フローの特定のタスクを実行できるようになっています。ISE Design Suite の Project Navigator から PlanAhead を起動した場合は、使用できる PlanAhead ツールの機能が I/O ピン配置、フロアプランおよびタイミング解析にのみになります。

PlanAhead を起動させる Project Navigator の [Processes] ウィンドウには次の 4 つのプロセスがあります。

- 合成前
 - I/O ピン配置
- 合成後
 - I/O ピン配置
 - エリア/IO/ロジックのフロアプラン
- インプリメンテーション後
 - タイミングの解析とデザインのフロアプラン

PlanAhead プロジェクト データは、Project Navigator で自動的に作成・管理されます。この 2 つのツール間で渡されるデータや、PlanAhead で表示されるウィンドウ レイアウトは、どのようにツールが起動されたかによって変わります。統合に関する詳細は、[443 ページの「Project Navigator 内の PlanAhead のプロセス」](#)を参照してください。

PlanAhead には、2 つのデフォルト ビュー レイアウトがあります。

- I/O Planning ビュー レイアウトと呼ばれる I/O ピン配置環境で、I/O ピンの配置および割り当てに関連したビューを含みます。
- デザイン解析環境 ([Default Layout] ビュー レイアウト) では、デザイン解析およびフロアプランに関連するビューが表示されます。

PlanAhead 表示環境の詳細は、[第 4 章「表示環境の使用」](#)を参照してください。ウィンドウ レイアウトの設定や読み込みの詳細は、[第 4 章の「ビュー レイアウトの使用」](#)を参照してください。

Project Navigator 内の PlanAhead のプロセス

Project Navigator と PlanAhead は、それぞれ独立した環境で、別のシステム プロセスの下で動作しています。2 のプロセスは、ツール間で効率的にデータを渡すために統合されています。1 つのツールでデザイン データに変更が加えられても、リアル タイムにはもう 1 つのツールで自動認識はされませんので、両方のツールで同時にロジックや制約の編集はしないでください。PlanAhead は目的に合わせて起動し、Project Navigator デザイン データを更新する前に閉じてください。

Project Navigator のプロセスは、PlanAhead で変更されたユーザー制約ファイル (UCF) を認識するため、データが保存されると同期します。次のセクションでは、ツールの起動ステップおよび統合を可能にするデータ トランザクションについて説明します。

ロジックおよび制約の受け渡し

ISE 統合モードの PlanAhead では、I/O ピン、ロジックの LOC 制約、および AREA_GROUP 制約に対する物理的な制約のみ変更することができます。RTL ソースや合成ネットリストでの論理接続は PlanAhead に解析目的のためのみ渡され、Project Navigator には戻されません。ロジックやタイミング制約を変更する PlanAhead の機能は ISE 統合モードでは使用できないようになっています。

ロジックの変更はすべて Project Navigator または外部 RTL や合成ツールで行う必要があります。Project Navigator に戻されるファイルは UCF 制約ファイルのみです。

PlanAhead では、コメントや不完全な制約などを含め、UCF ファイルの元の内容とフォーマットが保持されます。デザインで制約が有効であるかどうかは、PlanAhead を開いたり閉じたりするときにはチェックされます。DRC を実行すると、さらに詳細が確認できます。

PlanAhead が起動すると、Project Navigator プロジェクトの UCF ソース ファイルが PlanAhead のプロジェクトに渡され、物理制約を追加したり変更したりできます。

PlanAhead で [Save Design] コマンドを使用すると、変更された UCF ファイルが元の Project Navigator のソースの保存場所に書き込まれます。PlanAhead で制約を変更し [Exit] コマンドを実行すると、ツールを閉じる前に Project Navigator プロジェクトに変更を保存するかどうかを確認するメッセージが表示されます。

PlanAhead を起動したときに Project Navigator プロジェクトに UCF ファイルがない場合、UCF を作成するかどうかを確認するプロンプトが表示されます。この空の UCF ファイルが PlanAhead に渡されます。

PlanAhead では、複数の UCF ファイルを持つ Project Navigator プロジェクトがサポートされています。PlanAhead を起動する前に複数の UCF ファイルから 1 つを選択するようプロンプトが表示されます。

- PlanAhead で定義された新しい制約はすべて選択した UCF ファイルに書き込まれます。
- 選択されなかった UCF ファイルに存在する物理制約は、PlanAhead でその制約の値が変更されても、同じファイルの中に残ります。

Project Navigator デザイン フローで使用されるコア レベルの NCF ファイルは PlanAhead には渡されません。これらのファイルにある物理制約を使用・表示するには、PlanAhead を起動する前に最上位 UCF ファイルに手動でまとめておく必要があります。

一時的な PlanAhead プロジェクトが ISE プロジェクト ディレクトリに作成され、Project Navigator から PlanAhead が起動されるたびに削除および置換されます。

I/O ピン配置 (合成前)

合成ネットリストを作成する前に I/O ピン配置を早期に実行するには、PlanAhead をスタンドアロンで起動するか、または Project Navigator でこのプロセスを実行します。

注記： デザインプロセスのこの段階では、論理合成はまだ実行されていません。クロック ポート、クロック関連ロジック、差動ペア、GT などのコンセプトはツールには認識されていません。このようなタイプのポートはインプリメンテーション エラーを避けるために正しく配置されていることを必ず確認する必要があります。できる限り I/O ピン配置は論理合成の後に実行してください。ネットリストがある場合はクロック、クロック ロジック、差動ペア、GT などは、ピン配置実行中

に自動的に認識されます。また、インプリメンテーション前に配置の有効性を確認できるよう、論理接続およびクロックに基づき実行されるデザイン ルール チェック (DRC) も多数備わっています。

合成前に Project Navigator で I/O ピン配置を実行するには、次の手順に従います。

- [Processes] ウィンドウで [User Constraints] を展開し [IO Pin Planning (PlanAhead) - Pre-Synthesis] をダブルクリック (445 ページの図 15-1)
- [Tools] → [PlanAhead] → [I/O Pin Planning (PlanAhead)] → [Pre-Synthesis] をクリック

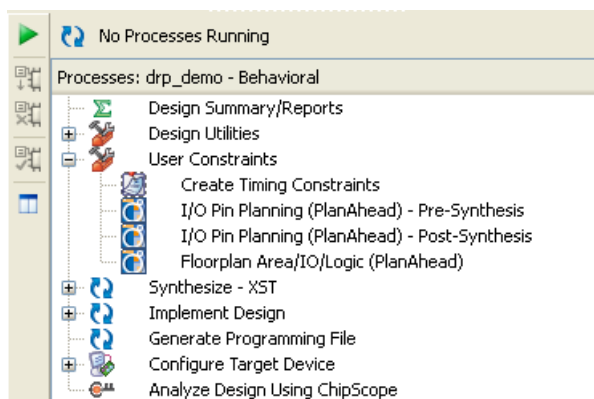


図 15-1 : Project Navigator の [Processes] ウィンドウ

PlanAhead が起動すると、Project Navigator からすべての RTL ソース ファイル、最上位モジュール名、UCF ファイルが PlanAhead に渡されます。PlanAhead には、デフォルトで I/O ピンを配置するための I/O Planning ビュー レイアウトが表示されます。RTL エラボレートが実行され、抽出された最上位 I/O ポートが [I/O Ports] ビューに表示されます。

PlanAhead プロジェクトを保存したり閉じたりすると、元の Project Navigator ソース UCF ファイルがアップデートされます。Project Navigator デザイン プロセスのステートはこれでリセットされます。

Project Navigator との統合およびプロセスについての詳細は、444 ページの「ロジックおよび制約の受け渡し」を参照してください。

I/O Planning ビュー レイアウトの使用方法については、第 8 章「I/O ピン配置」を参照してください。

I/O ピン配置 (合成後)

注記：できる限り I/O ピン配置は論理合成の後に実行してください。ネットリストがある場合はクロック、クロック ロジック、差動ペア、GT などは、ピン配置実行中に自動的に認識されます。また、インプリメンテーション前に配置の有効性を確認できるよう、論理接続およびクロックに基づき実行されるデザイン ルール チェック (DRC) も多数備わっています。

ロジック合成の実行後に Project Navigator で I/O ピン配置を実行するには、次の手順に従います。

- [Processes] ウィンドウで [User Constraints] を展開し [I/O Pin Planning (PlanAhead) - Post-Synthesis] をダブルクリック
- [Tools] → [PlanAhead] → [I/O Pin Planning (PlanAhead)] → [Post-Synthesis] をクリック

PlanAhead が起動すると、Project Navigator から合成された NGC または EDIF フォーマットのネットリストおよび UCF ファイルがプロジェクトに渡されます。PlanAhead には、デフォルトで

I/O ピンを配置するための I/O Planning ビュー レイアウトが表示されます。I/O ポートは、PlanAhead の [I/O Ports] ビューに表示されます。

PlanAhead プロジェクトを保存したり閉じたりすると、元の Project Navigator ソース UCF ファイルがアップデートされます。Project Navigator デザインプロセスのステートはこれでリセットされます。

Project Navigator との統合およびプロセスについての詳細は、[444 ページの「ロジックおよび制約の受け渡し」](#)を参照してください。PlanAhead の I/O 配置環境の使用方法については、[第 8 章「I/O ピン配置」](#)を参照してください。

エリア/I/O/ロジックのフロアプラン (合成後)

PlanAhead には、インプリメンテーションの前後に使用できるデザイン解析およびフロアプラン環境があります。ロジック合成後、インプリメンテーション前に Project Navigator でデザインを解析するか、フロアプランを実行するには、次の手順に従います。

- [Processes] ウィンドウで [User Constraints] を展開し [Floorplan Area/I/O/Logic (PlanAhead)] をダブルクリック
- [Tools] → [PlanAhead] → [Floorplan Area/I/O/Logic (PlanAhead)] をクリック

PlanAhead が起動すると、Project Navigator から合成された NGC または EDIF フォーマットのネットリストおよび UCF ファイルが PlanAhead に渡され、デフォルトのデザイン解析およびフロアプラン環境で起動します。

注記：下位 NGC コア ファイルがデザインで使用されソースとして追加されていない場合、Project Navigator で、変換プロセスのプロパティであるマクロ検索パス (-sd) を該当ディレクトリに設定します。

PlanAhead プロジェクトを保存したり閉じたりすると、元の Project Navigator ソース UCF ファイルがアップデートされます。Project Navigator デザインプロセスのステートはこれでリセットされます。

Project Navigator との統合およびプロセスについての詳細は、[444 ページの「ロジックおよび制約の受け渡し」](#)を参照してください。インプリメンテーション前に PlanAhead を使用する方法については、[第 5 章「エラボレート済み RTL デザイン」](#)を参照してください。インプリメンテーション後に PlanAhead を使用する方法については、[第 11 章「インプリメンテーション結果の解析」](#)および[第 10 章「デザインのフロアプラン」](#)を参照してください。

タイミング解析/デザインのフロアプラン (インプリメンテーション後)

PlanAhead にはインプリメンテーション後に使用できるデザイン解析およびフロアプラン環境があります。インプリメンテーション後のデザイン解析では、潜在的なデザインの問題を検出するために配置結果およびタイミング結果を確認できます。

物理的な LOC 制約または AREA_GROUP フロアプラン制約を設定することでインプリメンテーション ツールでより安定した結果が得られ、また、インプリメンテーション ランタイムを短縮することができます。

インプリメンテーション後に Project Navigator からデザインを解析するか、フロアプランを実行するには、次の手順に従います。

- [Processes] ウィンドウで [Implement Design] → [Place & Route] を展開し [Analyze Timing / Floorplan Design (PlanAhead)] をダブルクリック
- [Tools] → [PlanAhead] → [Analyze Timing / Floorplan Design (PlanAhead)] をクリック

Project Navigator からは、次のファイルが PlanAhead に渡されます。

- 合成済み NGC または EDIF フォーマットのネットリスト
- UCF ファイル
- ISE 配置データ
- タイミング結果
- ブロック RAM メモリ マップ (BMM) ファイル

PlanAhead がデフォルトのデザイン解析およびフロアプラン環境で起動します。PlanAhead で ISE 配置データを抽出するには、ISE の NCD (Native Circuit Description) ファイルを XDL (Xilinx® Definition List) フォーマット ファイルに変換する必要があります。

このコマンドの実行進捗状況を示すバーが PlanAhead に表示されます。PlanAhead の再起動を迅速化するため、まず XDL ファイルが存在するかどうかチェックされ、最新のものがあ場合は再生成されません。

最新の状態ではないインプリメンテーション プロセスで [Tools] → [PlanAhead] → [Analyze Timing / Floorplan Design (PlanAhead)] をクリックすると、デザインをインプリメントし直して PlanAhead を起動するか、またはインプリメンテーション ツールを再実行せずに既存の結果データを使用して PlanAhead を起動するかを選択するプロンプトが表示されます。

PlanAhead プロジェクトを保存したり閉じたりすると、元の Project Navigator ソース UCF ファイルがアップデートされます。Project Navigator デザイン プロセスのステートはこれでリセットされます。

PlanAhead の入力ファイルおよび出力ファイル

入力ファイル

PlanAhead™ では、入力ファイルの読み込み中に発生するエラー、警告、およびメッセージが planAhead.log ファイルに書き込まれます。これらのメッセージは、PlanAhead の [Tcl Console] ビューにも表示されます。

PlanAhead では、入力として使用するファイルの場所を指定できます。

注記：PlanAhead を起動すると、既存のジャーナル ファイルおよびログ ファイルが上書きされることに注意してください。これらのファイルが今後必要な場合は、保存しておいてください。PlanAhead の起動時には、最後に保存した状態のファイル セットのバックアップ コピーが *.jou_backup および *.log_backup に保存されます。

表 A-1 に入力ファイル名およびその説明を示します。

表 A-1：入力ファイル

ファイル名	説明
デザイン テキスト ファイル (Verilog、VHDL)	Verilog や VHDL ファイルをインポートおよびエラボレートして、ロジックを解析したり、ソースを修正することができます。元のソース ファイルはそのままの場所から参照することも、プロジェクトにコピーすることもできます。ディレクトリは、RTL ソースファイルのインポート時に指定します。ディレクトリに含まれた認識可能なファイルおよびファイル タイプはすべてプロジェクトにインポートされます。
I/O ポート リスト (CSV)	CSV (カンマ区切り) フォーマット ファイルは、インポートして I/O Planner の [I/O Ports] ビューに表示できます。この機能は、I/O ピン配置プロジェクトでのみ使用できます。 これらの I/O ポートを物理パッケージ ピンに割り当てて、デバイス ピンの設定を定義できます。CSV は、FPGA やボード設計にあたり、デバイス ピンおよびピン配置のデータを交換するときに使用される標準ファイル形式です。CSV の各列は、 450 ページの「CSV ファイル形式の I/O ポート リスト」 を参照してください。
モジュール レベルのネット リストとコア (EDIF、NGC、NGO、BMM)	PlanAhead では、階層デザイン手法をサポートする複数の EDIF または NGC ネットリストを使用してデザインを構築できます。最上位ロジックを選択すると、下位のモジュールが自動的にインポートされます。デザイン モジュールを検索する検索パスを定義できるため、デザインをアップデートするとき非常に柔軟に対応できます。PlanAhead のインクリメンタル ネットリスト インポート機能では、どのレベルのデザイン階層のネットリストでも更新できます。

表 A-1 : 入力ファイル (続き)

ファイル名	説明
最上位ネットリスト (EDIF、NGC)	<p>PlanAhead では、EDIF または NGC ネットリストのインポートがサポートされています。このネットリストは、Virtex®-4、Virtex-5、Virtex-6、Virtex-7、Kintex™-7、Artix™-7、Spartan®-3 または Spartan-6 デバイスの合成に使用されます。</p> <p>PlanAhead では、階層デザイン手法をサポートする複数のネットリストを使用してデザインを構築できます。最上位ロジックを選択すると、下位のモジュールが自動的にインポートされます。インクリメンタルなネットリストのインポート機能では、どのレベルのデザイン階層のネットリストでも更新できます。作成中のフロアプラン制約は、更新されても保持されます。</p>
制約ファイル (UCF/NCF/XCF)	<p>PlanAhead では、タイミング制約および物理制約用の UCF、NCF、および XCF フォーマットファイルのインポートがサポートされています。PlanAhead では、複数の UCF ファイルをインポートできるので、物理制約、I/O 制約、およびタイミング制約を区別できます。</p> <p>NCF ファイルはモジュールレベルの制約で、デザインのブロックまたは IP コア特有です。</p> <p>XCF ファイル (XST 制約ファイル) もプロジェクトにインポートできます。</p> <p>PlanAhead ではザイリンクス®でサポートされている UCF 制約がすべてサポートされています。UCF 制約およびサポートされる構文の詳細は、付録 E 「その他のリソース」 に示す『制約ガイド』(UG612) を参照してください。</p>
ザイリンクス ISE 配置結果 (NCD/XDL)	<p>PlanAhead では、XDL フォーマットデータを使用して ISE 配置結果をインポートできます。XDL データはインプリメンテーション run が起動されると自動的に作成されます。</p> <p>ISE コマンドが実行されたら、<i>placed_design_name.ncd</i> ファイルから XDL ファイルを作成できます。XDL ファイルを作成したら、個々のブロックまたはデザイン全体の配置をインポートできます。</p> <p>[Import Placement] ダイアログ ボックスで <i>placed_design_name.ncd</i> ファイルを指定すると、XDL コマンドが自動的に実行されます。</p>
TRCE タイミング結果 (TWX/TWR)	<p>PlanAhead では、ザイリンクス TRCE コマンドで生成されるタイミングレポート (TWX および TWR) をインポートできます。インポートすると、すべての信号トレースおよび選択が [Timing Results] ビューに表示されます。</p> <p>どちらのファイルもある場合は、TWX の方が適しています。</p>

CSV ファイル形式の I/O ポート リスト

CSV (カンマ区切り) フォーマット ファイルは、インポートして I/O Planner の [I/O Ports] ビューに表示できます。この機能は、I/O ピンを配置する空のプロジェクトでのみ使用できます。

これらの I/O ポートを物理パッケージ ピンに割り当てて、デバイス ピンの設定を定義できます。CSV は、FPGA やボード設計にあたり、デバイス ピンおよびピン配置のデータを交換するとき使用される標準ファイル形式です。

CSV ファイルは、[File] → [Export] → [Export I/O Ports] コマンドを使用して、エラボレート済みデザイン、合成済みデザイン、またはインプリメント済みのデザインからエクスポートすることも可能です。

CSV の列には、次が含まれます。

- [IO Bank] : ピンが配置される I/O バンク。デバイスのすべてのピンに対しこのフィールドにデータがソフトウェアで自動入力されます。値は数値または空白です。入力 CSV ファイルでは必須のフィールドではありません。

- **[Pin Number]** : パッケージ ピンの名前またはロケーション。デバイスのすべてのピンに対しこのフィールドにデータがソフトウェアで自動入力されます。入力ファイルで必須のフィールドではありません。入力に使用される場合は、配置を定義するために使用されます。値はデバイスの有効なピンです。
- **[IOB Alias]** : パッケージ ピンの代替パーツ名。このフィールドはソフトウェアで指定され、入力 CSV ファイルで指定されていない場合は使用されません。
- **[Site Type]** : デバイス データシートからのピン名。このフィールドはソフトウェアで指定され、入力 CSV ファイルで指定されていない場合は使用されません。
- **[Min Trace Delay (ps)]/[Max Trace Delay (ps)]** : デバイスのパッド サイトとパッケージのボール間距離で、単位はピコ秒 (ps) です。これらのフィールドはソフトウェアで入力され、ボード設計時にトレース遅延を一致させるために使用します。また、これらのフィールドは出力ファイルのみに含まれており、入力ファイルには含まれていません。
- **[Trace Length (um)]** : パッケージ ピンとダイ パッド間の内部トレース長を指定します。
- **[Prohibit]** : さまざまな理由でユーザー I/O がサイトに追加されないよう、特定のサイトを使用禁止にできます。PROHIBIT を設定することで、ボード レイアウト問題の抑制、信号間クロストークの低減、同じパッケージの複数の FPGA 間でピン配置が有効であるかどうかの確認に役立ちます。UCF ファイルでは、これは CONFIG PROHIBIT 制約として記述されています。値は TRUE または空白のどちらかになります。[Pin Number] が空白の場合は、このフィールドも空白にします。
- **[Interface]** : ユーザー I/O のセットをユーザーが指定してグループ化します。たとえば、このフィールドは、メモリ インターフェイスのデータ、アドレス、およびイネーブル信号の関係を指定するのに使用できます。値は文字列または空白です。
- **[Signal Name]** : FPGA デザインのユーザー I/O 名。値は文字列、または割り当てられていないパッケージ ピンの場合は空白です。
- **[Direction]** : 信号の方向。値は IN、OUT、INOUT、またはユーザー I/O がサイトに割り当てられていない場合は空白です。
- **[DiffPair Type]** : 差動ペアのピンが N 側、P 側のどちらにあるかが記述されます。これは差動信号に対してのみ使用されます。ソフトウェアは、ピン名ではなく、この列を使用してどのピンが N 側か P 側かを判断します。
値は P、N、またはユーザー I/O がサイトに割り当てられていない場合は空白です。
- **[DiffPair Signal]** : 差動ペアのもう 1 つのピン名を指定するのに使用します。値はユーザー I/O 名、または未使用の場合は空白です。
- **[I/O Standard]** : 特定ユーザー I/O の I/O 規格。このフィールドがユーザー I/O に対し空白の場合、該当するデフォルト値が使用されます。値は有効な I/O 規格、または空白です。
- **[Drive]** : 特定ユーザー I/O の I/O 規格の駆動強度。すべての I/O 規格が駆動強度を必要とするわけではありません。このフィールドが空白の場合、デフォルト値が使用されます。値は数値または空白です。
- **[Slew Rate]** : 特定ユーザー I/O の I/O 規格のスルー レート。すべての I/O 規格がスルー レートを必要とするわけではありません。このフィールドが空白の場合、デフォルト値が使用されます。値は FAST および SLOW です。
- **[Pull Type]** : 選択されているポートの抵抗のタイプを指定します。トライステート出力バッファ (OBUFT) または双方向バッファ (IOBUF) を使用する場合、出力に弱いプルアップ抵抗、弱いプルダウン抵抗、またはウィーク キーパー回路を付けることができます。入力バッファ (IBUF) を使用する場合は、入力に弱いプルアップ抵抗または弱いプルダウン抵抗を付けることができます。

- [Phase] : 同期位相オフセットの場合に、バンク内のほかの I/O の位相に関連した I/O の位相を指定します。
- [Board Signal] : ボード レベル デザインから I/O に入力される信号の名前を指定します。
- [Board Voltage] : ボード レベル デザインから I/O に入力される信号の電圧レベルを定義します。
- [BUFIO2_REGION] : ポートが関連付けられる BUFIO2 クロック領域を定義します。
- [IN_TERM/OUT_TERM] : Spartan-6 デバイス向けにオプションの IN_TERM または OUT_TERM ドライバのインピーダンス属性を定義します。このフィールドは、ほとんどの場合で空白のままにします。製品デバイスではまだサポートされていません。この終端定義を使用すると、SLEW および DRIVE STRENGTH 属性が上書きされます。SSN の計算ではサポートされません。
- [OFFCHIP_TERM] : I/O の外部ボード レベルの終端を指定します。SSN の計算ではサポートされません。このフィールドを空白のままにすると、SSN の計算に予測された終端が使用され、この終端がデフォルトで SSN レポートおよび I/O ポートの表に表示されます。予測される終端とそれに対応する PlanAhead で表示される短縮名については、付録 E「その他のリソース」に示す『Spartan-6 FPGA Select I/O リソース ユーザー ガイド』(UG381) に記述されています。

定義されない列の値は、PlanAhead で CSV を読み込むときに保持され、[I/O Ports] ビューでユーザー定義列としてレポートされます。

CSV での差動ペアの定義

信号名、差動ペア信号、差動ペア タイプ、I/O 規格など、CSV ファイルで差動ペアを定義するために使用されるプロパティは複数あります。CSV ファイルのそれ以外の値は、差動ペアを有効にするために使用され、互換性のあるものにはしても、ペアを定義するためには使用されません。CSV ファイルでは、差動ペアをさまざまな方法で定義できます。

次は、差動ペアを定義するには、次のような方法があります。

- [Diff Pair] : 差動ペアを構成する 2 つの信号を直接定義します。2 つのポート エントリ (それぞれもう一方の Signal Name にリンクされた DiffPair Signal 値を含む) には、補足の DiffPair Type 値 (N 1 つと P 1 つ) が含まれます。ツールでは、差動ペアを作成する際に I/O Standard などのほかの属性が互換性があるかどうかチェックされます。
- [Single-link Diff Pair] : 2 つのポート エントリには、補足の DiffPair Type 値 (N 1 つと P 1 つ) が含まれますが、もう一方の Signal Name にリンクされた DiffPair Signal 値が設定されるのは 1 つのポートのみです。この場合、すべての属性に互換性があれば、差動ペアは作成されます。
- [Single Port Diff Pair] : ポート エントリは、差動ペアの I/O Standard、DiffPair Type、DiffPair Signal 値を含む 1 つだけです。元のポートのプロパティと一致するプロパティで反対側の差動ペア (N 側または P 側) が作成されます。
- [Inferred Diff Pair] : ポート エントリは 2 つで、差動ペアの I/O 規格 (例 : DIFF_HSTL、DIFF_SSTL) と N および P 側を推論する Signal Names 値が含まれます。この場合、すべての属性に互換性があれば、差動ペアは推論されます。

CSV ファイルにこれらのポート エントリがあると、差動ペアは直接定義されます。または、すべての属性に互換性がある場合は差動ペアが推論されます。差動ペアが推論されると、PlanAhead ではそのペアの割り当てを確認するメッセージが表示されます。詳細は、292 ページの「I/O ポートのインポート」を参照してください。

レポートの出力

453 ページの図 A-1 に、Windows OS でレポートを出力するときを使用されるディレクトリ構造を示します。ユーザーがファイルの場所を指定しない場合、次の場所にレポート ファイルが出力されます。

- Tcl コマンドからの出力は、PlanAhead ツールが起動されるディレクトリに書き込まれます。
- GUI からの出力は、デフォルトのプロジェクト ディレクトリに書き込まれます。
- 合成またはインプリメンテーションを実行すると、出力ファイルはプロジェクトの run ディレクトリに書き込まれます。
- PlanAhead のジャーナル ファイルおよびログ ファイルは、OS によって保存される場所が異なります。
 - Linux では PlanAhead の起動ディレクトリに出力されます。
 - Windows では、図 A-1 に示すように次のディレクトリになります。
%APPDATA%\Xilinx\PlanAhead

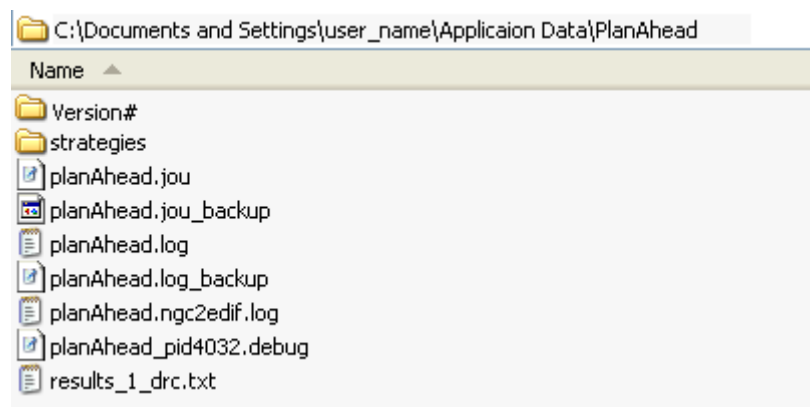


図 A-1：レポート出力を含む PlanAhead ディレクトリ

表 A-2 に、PlanAhead の出力ファイルおよびその説明を示します。

表 A-2：レポート出力

ファイル名	説明
I/O ピン配置 (CSV)	CSV フォーマット ファイルで、すべての I/O ポート配置と関連するパッケージ ピン情報が含まれています。このファイルは、RTL ポートのヘッダ定義および PCB 回路シンボル生成で使用されます。
I/O ピン配置 (RTL : Verilog または VHDL)	Verilog または VHDL フォーマットのファイルで、すべての I/O ポートの配置がポートとして定義されています。このファイルは、RTL ポートのヘッダー定義で使用されます。
ログ ファイル (planAhead.log および planAhead.log.backup)	ログ ファイル planAhead.log には、PlanAhead のコマンドを実行したときに生成されるメッセージの内容が含まれています。ファイルを表示するには、メイン メニューから [Flow] → [Open Log File] をクリックします。

表 A-2：レポート出力 (続き)

ファイル名	説明
ジャーナル ファイル (planAhead.jou および planAhead.jou.backup)	<p>ジャーナル ファイル planAhead.jou には、1 つのセッションの Tcl コマンドすべてが含まれています。メイン メニューから [File] → [Open Journal File] をクリックするとファイルを表示できます。</p> <p>ジャーナル ファイルを再生すると、前のセッションで使用したコマンドを再利用できます。Tcl スクリプトは、ジャーナル ファイルからコマンドをコピーすると生成できます。このファイルでは、エラーを含むコマンドまたは再生する前の複数のセッションのコマンドを削除する必要がある場合があります。</p> <p>注記： 操作によっては、ジャーナル ファイルに Tcl コマンドが書き込まれないものもあります。</p>
エラー ログ ファイル (planAhead_pidxxx.debug および hs_err_pidxxx.log)	<p>エラー ログ ファイルには、PlanAhead のエラーをデバッグするときに役立つ重要な情報が含まれています。PlanAhead で内部例外エラーが発生したことを示す警告メッセージが表示される場合、そのエラー ファイルが保存されます。ザイリンクス テクニカル サポートからウェブケースを開くときは、この生成されたエラー ログ ファイル、ジャーナル ファイル (planAhead.jou)、およびログ ファイル (planAhead.log) を含めてください。</p> <p>これらのファイルにはデザイン データは含まれていません。</p>
[DRC Results] ビュー	デザイン ルール チェック (DRC) を実行するたびに、結果がファイルに書き込まれます。
タイミング解析結果 (Excel ファイル)	タイミング解析の結果は、テキスト ファイルにエクスポートできます。データをエクスポートするには、[Timing Results] ビューで [Export to Text File] アイコンをクリックします。
ネットリスト モジュール、 Pblock、およびクロック領域統計 レポート	<p>[Instance Properties]、[Clock Regions Properties]、および [Pblock Properties] ビューに表示されるリソース統計はマイクロソフトの Excel 形式のファイルにエクスポートできます。このファイルには、リソース使用率、RPM、キャリー チェーンのサイズ、クロックおよびクロックが供給されるインスタンス、およびその他のリソースのデータなどの情報が含まれます。</p> <p>データをエクスポートするには、[Instance Properties]、[Clock Regions Properties]、および [Pblock Properties] ビューの [Statistics] タブで [Export Statistics] をクリックします。ダイアログ ボックスが表示され、レポートに含める情報や階層のレベル数を定義できます。</p>
SSN 解析レポート	同時スイッチ ノイズ (SSN) 解析の結果は、[Run SSN Analysis] ダイアログ ボックスでファイル名およびファイルを保存する場所を指定して、CVS または HTML 形式のレポート ファイルにエクスポートできます。
WASSO 解析レポート	WASSO 解析の結果は、[Run WASSO Analysis] ダイアログ ボックスでファイル名およびファイルを保存する場所を指定して、テキスト形式のレポート ファイルにエクスポートできます。
ストラテジ ファイル	/strategies ディレクトリには、ISE® のインプリメンテーション コマンドに対して指定したデフォルトのコマンド ライン オプションが含まれています。ストラテジは、PlanAhead を使用したどの ISE インプリメンテーション実行にも適用でき、作成したり、または提供されているストラテジをコピーできます。

デフォルト環境の出力

PlanAhead ツールでは、現在のウィンドウ レイアウトおよびテーマの設定が、起動時に読み込まれるコンフィギュレーションおよび初期化ファイルに保存されます。カスタム テーマ、ウィンドウ レ

イアウト、run ストラテジは、必要なときに読み込むこともできます。これらのテーマおよびレイアウトを定義するファイルは、次のディレクトリの PlanAhead ツール環境フォルダーに書き込まれます。

- Windows の場合、PlanAhead ツールの環境設定は Document and Settings の下のユーザー フォルダーに保存されます。
 - C:\Documents and Settings\<user_name>\Application Data\Xilinx\PlanAhead\<version>
- Linux の場合、PlanAhead ツールの環境設定はユーザーのホーム ディレクトリに保存されます。
 - ~/.Xilinx/PlanAhead/<version>

表 A-3 に、PlanAhead 環境のコンフィギュレーション ファイルのリストおよびそのファイルの場所と説明を示します。

表 A-3 : PlanAheadのデフォルト環境での出力

ファイル名	説明
ビュー表示オプション ファイル (planAhead.ini)	../PlanAhead/<version>/planAhead.ini 初期化ファイルには、表示色および表示オプションなどの [Tools] → [Options] の設定がすべて含まれています。詳細は、第 4 章の「PlanAhead の設定」を参照してください。 PlanAhead ツールを終了すると、ユーザーの設定が planAhead.ini ファイルに保存され、次に起動したときに、自動的にインポートされて、初期化設定が適用されます。
PlanAhead テーマ ファイル (<theme_name>.patheme)	../PlanAhead/<version>/themes ディレクトリには、PlanAhead の GUI で表示するレイヤーの色および色塗りパターン テーマをカスタマイズすると作成される *.patheme ファイルが含まれます。作業中のセッションに使用するテーマ ファイルをプルダウン メニューから選択できます。詳細は、第 4 章の「表示環境の設定」を参照してください。
ビュー レイアウト ファイル (<layout_name>.layout)	../PlanAhead/<version>/layouts ディレクトリには、PlanAhead ツールの GUI のビューのコンフィギュレーションを定義する *.layout ファイルが含まれます。カスタム ビュー レイアウトは、[Layout] → [Save Layout As] コマンドで作成できます。詳細は、第 4 章の「ビュー レイアウトの使用」を参照してください。
キーボード ショートカット ファイル (shortcuts.xml)	../PlanAhead/<version>/shortcuts ディレクトリには、ツール コマンドのキーボード ショートカットを指定する shortcuts.xml ファイルが含まれます。複数のショートカットを定義、設定できます。これらは、ショートカット ファイルに保存されます。詳細は、第 4 章の「ショートカット キーの設定」を参照してください。
カスタム コマンド ファイル (commands.paini)	../PlanAhead/<version>/commands ディレクトリには、PlanAhead の GUI に追加されるカスタム Tcl コマンドを保存した commands.paini ファイルが含まれます。詳細は、第 4 章の「カスタム メニュー コマンドの追加」を参照してください。

プロジェクト データの出力

図 A-2 に、PlanAhead のプロジェクト データ ディレクトリ 構造を示します。

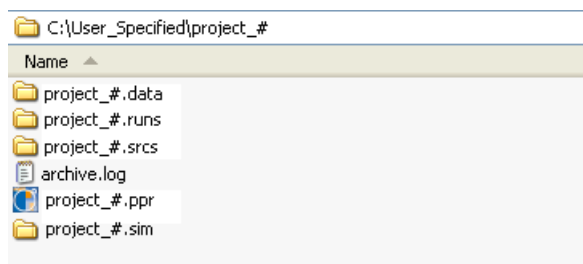


図 A-2 : プロジェクト データ ディレクトリ 構造

表 A-4 に、PlanAhead プロジェクト データ 出力およびその説明を示します。

表 A-4 : PlanAhead のプロジェクト データ 出力

出力	説明
プロジェクト ディレクトリ (<i>projectname</i>)	新しいプロジェクトが作成されると、PlanAhead ではプロジェクト ファイル、プロジェクト データ ディレクトリ、および ISE インプリメンテーション結果を含めるプロジェクト ディレクトリが作成されます。このプロジェクト ディレクトリには、New Project ウィザードで入力したプロジェクト名と同じ名前が付けられます。
プロジェクト ファイル (<i>projectname.ppr</i>)	新しいプロジェクトが作成されると、PlanAhead ではプロジェクト ファイルが作成されます。このプロジェクト ファイルには、New Project ウィザードで入力したプロジェクト名と同じ名前が付けられます。
プロジェクト データ ディレクトリ (<i>projectname.data</i>)	新しいプロジェクトが作成されると、PlanAhead ではプロジェクト データを含めるためのプロジェクト データ ディレクトリが作成されます。このプロジェクト データ ディレクトリには、New Project ウィザードで入力したプロジェクト名と同じ名前が付けられます。
プロジェクト データ : ネットリスト サブディレクトリ (<i>netlist</i>)	<p>/netlist というサブディレクトリには、デザイン全体のネットリストのコピーが保存されます。</p> <p>RTL ベースのプロジェクトの場合、PlanAhead では /Synthesis サブディレクトリが各 run に対して作成され、ネットリストが含まれます。このディレクトリは run がリセットされるたびに更新されます。</p> <p>ネットリスト ベースのプロジェクトの場合、インポートされたネットリストを含む 1 つのネットリスト ディレクトリが作成され、デザインで使用された NGC コア ファイルのすべてのコピーもこのディレクトリに保存されます。</p>

表 A-4 : PlanAhead のプロジェクト データ出力 (続き)

出力	説明
プロジェクト データ: 制約 セット サブディレクトリお よびファイル (<i>constraint_set_name</i>)	<p>制約セットを作成すると、PlanAhead では <i>/projectname.data</i> ディレクトリの下にそのサブディレクトリが作成されます。制約セット ディレクトリには、次のファイルが含まれます。</p> <ul style="list-style-type: none"> *.ucf: インポートされた UCF ファイル。入力ファイルと異なる場合があります。 iseloc.xml: 固定されている配置制約と ISE からインポートされた固定されていない配置制約を区別するのに使用します。 pfi.xml: デザインの制約ターゲット デバイスが含まれています。 pfp.xml: 現在の実行情報が含まれています。 expX サブ: 各 run の実行情報が含まれています。
プロジェクト RTL ディレク トリ (<i>projectname.srscs</i>)	<p>プロジェクト ソース ディレクトリには、プロジェクトにインポートされた HDL ソース ファイルが保存されます。これらのフォルダーは、PlanAhead で管理されるため、ユーザーが管理する必要はありません。</p> <p>注意: これらのファイルを変更すると、プロジェクト データが破損する可能性があります。</p>

プロジェクト データ: シミュレーション (*projectname.sim*)

ビヘイビアー シミュレーションおよびタイミング シミュレーションの run のプロジェクト シミュレーション ディレクトリ構造は、同じです。

project_name/project_name.sim/sim_run_name/sim_#

表 A-5 に、ビヘイビアー シミュレーションおよびタイミング シミュレーションの run で作成されるディレクトリおよびファイルを示します。

表 A-5 : ビヘイビアー シミュレーションおよびタイミング シミュレーションのファイルおよびディレクトリ

ファイル/ディレクトリ名	説明	シミュレーションの種類
fuse.log	fuse 実行ログ ファイル	両方
fuse.xmsgs	fuse 実行ログ ファイル (XML フォーマット)	両方
fuseRelaunch.cmd および ISim.cmd	ISim のバッチ コマンドで、-tchbatch が指定されないときに tclbatch のバナーに渡されます。	両方
ISim.log	ISim 実行ログ ファイル	両方
top.exe	[Simulation Launch] ダイアログ ボックスで指定した最上位モジュールから fuse を使用して作成された ISim シミュレーション実行ファイル (ファイル名は最上位モジュールに依存)	両方
top.prj	最上位デザインを含む PlanAhead プロジェクト ファイル	両方
top.wdb	ISim で作成される波形データベース ファイル	両方
top_timing_sim.nlf	NetGen 実行ログ ファイル	タイミング
top_timing_sim.sdf	NetGen で出力される DF 遅延ファイルで、タイミング シミュレーションで使用	タイミング

表 A-5: ビヘイビア シミュレーションおよびタイミング シミュレーションのファイルおよびディレクトリ

ファイル/ディレクトリ名	説明	シミュレーションの種類
top_timing_sim.v	Netgen で出力される Verilog ネットリスト出力で、タイミング シミュレーションで使用 (-ofmt オプションを使用すると VHDL ファイルを出力可能)	タイミング
xilinxsim.ini	ライブラリの論理から物理へのマッピングを含むファイル	両方
/ISim	isim_usage_statistics.htm ファイルを含むディレクトリ	両方
/isim/work	glbl.sdb および top.sdb ファイルを含むディレクトリ	両方
isim/top.exe.sim	デザインを構成するデザイン ユニットそれぞれに対するオブジェクト コードおよびデータ ファイルを保存するために fuse で生成されたディレクトリ。デザインで ISim を実行するために fuse で作成されたシミュレーション実行ファイル top.exe が含まれています。	両方
../top.exe.sim/secureip	fuse でコンパイルされるデザイン データおよびオブジェクト コードが含まれています。	両方
../top.exe.sim/simprims_ver	fuse でコンパイルされるデザイン データおよびオブジェクト コードが含まれています。	両方
../top.exe.sim/work	fuse でコンパイルされるデザイン データ ファイルおよびオブジェクト コードを含むディレクトリ (top.exe_main.c および top.exe_main.os_type.obj)	両方

ISE インプリメンテーションの出力ファイル

表 A-6 で、PlanAhead の ISE インプリメンテーション デザイン操作で生成されるファイルについて簡単に説明します。これらのファイルは、PlanAhead で管理されるので、手動で変更しないでください。

表 A-6 : ISE インプリメンテーションの出力ファイル

出力	説明
run ディレクトリ (<i>projectname.runs</i>)	<p>PlanAhead では、複数の ISE インプリメンテーションの run をキューに含めて順番に実行できます。このとき、run ディレクトリの場所を指定するダイアログ ボックスが表示されます。デフォルトでは、保存先にプロジェクト ディレクトリが表示されます。</p> <p>各 run ディレクトリには完全な EDIF ネットリストおよび UCF 制約ファイルが含まれています。PlanAhead では、run ディレクトリごとにユーザー指定のオプションで ISE コマンドを実行する実行スクリプトが作成されます。</p> <p>各ディレクトリには、ネットリストおよび制約ファイルを含むすべてのインプリメンテーション デザイン データが保存されています。要件を満たしたインプリメンテーションが達成されると、run ディレクトリ全体をアーカイブすることができます。</p>
EDIF ネットリスト (.edf)	<p>PlanAhead では、次のコマンド実行時に作成される EDIF フォーマットの ASCII ネットリスト ファイルをエクスポートできます。</p> <ul style="list-style-type: none"> • [Implement] および [Launch Runs] (PlanAhead) • [File] → [Export] → [Export Netlist] • [File] → [Export] → [Export Pblocks] • [File] → [Export] → [Export IP]
[Implement] および [Launch Runs] コマンド	<p>PlanAhead のインプリメンテーション run を実行すると、必要なファイルが自動的にエクスポートされ、この run に適用するストラテジで指定したオプションを使用して ISE コマンドが実行されます。</p> <p>run を実行すると、EDIF および UCF データが自動的にエクスポートされます。また、最上位デザインの EDIF フォーマットのネットリストおよび UCF フォーマットの制約ファイルを含む run ディレクトリが作成されます。ファイル名は、インポートされた EDIF ファイルに含まれている最上位ネットリスト名と同じになります。</p> <p>NGC/NGO フォーマットのモジュールのネットリスト ファイルが使用されている場合は、各 run ディレクトリにこれらがコピーされます。</p> <p>[Synthesis Run Properties] ビューおよび [Implementation Run Properties] ビューには、run ディレクトリが示されます。</p>
エクスポートされたネット リスト	<p>PlanAhead 環境外の ISE インプリメンテーションで使用するデザインの EDIF ファイルです。出力ネットリストには元の論理ネットリストの階層が含まれています。出力ファイル名は [Export Netlist] ダイアログ ボックスで指定できます。</p>
ChipScope コアのネットリ スト (.ngc)	<p>PlanAhead と ChipScope™ Pro Analyzer は統合されているため、Integrated Logic Analyzer (ILA) コアを挿入およびコンフィギュレーションできます。このコアがインプリメントされると NGC フォーマットのネットリストがコンパイルされ、プロジェクトの /netlist ディレクトリに保存され、各インプリメンテーションの run ディレクトリにコピーされます。詳細は、第 12 章, デザインのプログラムとデバッグを参照してください。</p>
制約ファイル (.ucf)	<p>PlanAhead では、ISE で使用されるタイミング制約および物理制約を含む UCF フォーマットの ASCII ファイルが書き出されます。このファイルは、次のコマンドを実行すると生成されます。</p> <ul style="list-style-type: none"> • [Implement] および [Launch Runs] (PlanAhead) • [File] → [Export Constraints] • [File] → [Export] → [Export Pblocks] • [File] → [Export] → [Export IP]

表 A-6 : ISE インプリメンテーションの出力ファイル (続き)

出力	説明
[Implement] および [Launch Runs] コマンド	<p>run を実行すると、EDIF および UCF データが自動的にエクスポートされます。</p> <p>run を起動すると、出力ネットリストに元の論理階層を含む実行ディレクトリが作成されます。run に対しエクスポートされるファイルは、最上位デザイン全体の EDIF フォーマットのネットリストと UCF フォーマットの制約ファイルです。ファイル名はインポートされた EDIF ファイルの元の最上位ネットリスト名と同じです。</p>
[Export Constraints] コマンド	<p>このコマンドを使用して制約をエクスポートすると、元の UCF ファイルの内容と構造 (コメントも含む) が保持されます。</p> <p>出力ファイル名は [Export Constraints] ダイアログ ボックスで指定できます。</p>
エクスポートされた Pblock	<p>特定の Pblock の EDIF および UCF ファイルを PlanAhead 環境外の ISE インプリメンテーションで使用するために、このコマンドを使用して Pblock をエクスポートします。</p> <p>Pblock をエクスポートすると、PlanAhead で Pblock の割り当てに基づいてネットリスト階層が作成されます。この UCF では、エクスポートされた EDIF ネットリスト名と一致するよう、PlanAhead の物理階層構造が参照され、ブロック ベースのインプリメンテーション ストラテジを使用するときに柔軟に対応できます。</p> <p>エクスポートされる Pblock ファイルは、ネットリスト 1 つと制約ファイル 1 つです。PlanAhead では自動的にブロック レベルのディレクトリ構造が作成、管理されます。選択した Pblock をエクスポートすると、<i>pblockname_CV.edn</i> および <i>pblockname_CV.ucf</i> ファイルを含む <i>pblockname_CV</i> サブディレクトリが作成されます。</p> <p>[Export Pblocks] コマンドは通常、物理階層を含む複雑な IP で使用されます。このような IP でタイミング クロージャを達成すると、コードを再構築してネットリストを取得せずにべつのデザインで使用できます。</p>
[Export IP] コマンド	<p>特定のネットリスト モジュールの EDIF および UCF ファイルを書き出して、再利用可能な IP ブロックの作成に使用します。</p> <p>デザインでモジュール インスタンスを選択して [Export IP] コマンドを実行すると、Pblock ロジック <i>logical hierarchy</i> および配置制約がエクスポートされます。エクスポート ファイルには、元のネットリスト フォーマットの EDIF ネットリストおよび UCF 物理制約が含まれます。インターフェイスをそのまま保持することで、次のデザインのインプリメンテーションが簡単に実行できます。</p> <p>また、エクスポートした UCF ファイルは Pblock の配置制約の再生成に使用できます。モジュールのインポート後にモジュールを移動させると、複数のモジュールに同じ配置を複製できます。</p> <p>365 ページの図 10-21 に、[Export Pblock] コマンドを使用して物理階層でのデザインの特定期間所をエクスポートする方法と、[Export IP] コマンドを使用して論理階層デザインの特定期間所をエクスポートする方法を示します。</p>
ISE 起動スクリプト (jobx.bat/sh、runme.bat/sh、および ISE_command.rst)	<p>run を実行すると、ISE 起動スクリプトが自動的に作成されます。スクリプトには、PlanAhead のストラテジで指定されたコマンドおよびコマンド ライン オプションが含まれています。</p> <p>jobx.bat/sh スクリプトは .jobs サブディレクトリにあるプロジェクト実行ディレクトリに保存され、選択された各実行を順番に起動します。このスクリプトにより各実行の runme.bat/sh スクリプトが呼び出されます。これらのスクリプトは、個別に実行することもできます。</p>

PlanAhead の DRC

RTL DRC :消費電力およびパフォーマンス

表 B-1 および462 ページの表 B-2 は、RTL 消費電力および RTL パフォーマンスの DRC を示しています。

RTL DRC : 消費電力

表 B-1 : 消費電力ルール

ルール名	略称	説明	重要度
Constantly enabled synchronous RAM	RPRC	常にイネーブルの RAM (推論またはインスタンス化されたもの) が片方または両方のポートで検出されました。この RAM が常にアクセスされていないかどうかを判断します。常にアクセスされていなければ、この RAM をディスエーブルにするロジックを記述することで、消費電力を大幅に低減できる場合があります。	警告
Inefficient dangling block RAM port	RPRM	接続されていない出力ポートのある RAM が検出され、WRITE_MODE が NO_CHANGE 以外の値に設定されています。RAM 記述を変更して未接続の出力ポートを設定すると (WRITE_MODE を NO_CHANGE に設定)、ブロック RAM の消費電力を最大 10% 節約できる場合があります。	警告
Shallow RAM implemented in block RAM	RPRS	Virtex®-5 および Virtex-6 デバイス :幅が 19 ビット以上、ワード数が 64 ビット以下の RAM の場合、RAM が FIFO として使用されている場合 (クロスオーバー ポイントはワード数 32 ビット以下) を除き、SelectRAM™ (分散 RAM と呼ばれる LUT ベースの RAM) を選択した方が有益です。幅が 18 ビット未満のインターフェイスを構築する場合、ワード数が 128 ビットまでの場合は LUT ベースの SelectRAM が適していますが、一般的に 128 ビットを超えると、専用ブロック RAM の方が適しています。	警告
Inefficient mapping of small multiplier in DSP block	RPDS	DSP または MULT18X18 などのハード乗算器 IP にマップされている小型乗算器は、MSB にプッシュする必要があります。残りの LSB はグラウンドにマップする必要があります。このようにすると、キャリー伝搬を最小限に抑えることができます。通常のインプリメンテーション、特に乗算器を推論する場合、LSB および符号拡張を使用して MSB をマップします。	警告

RTL DRC : パフォーマンス

表 B-2 : パフォーマンスのルール

ルール名	略称	説明	重要度
Inefficient library element instantiation	RPWL	別の FPGA ファミリに属す「 <i>library_component_name</i> 」タイプのインスタンス「 <i>instance name</i> 」を検出しました。この場合、最適なパフォーマンスが得られない可能性があります。ISE ソフトウェアで、このエレメントが選択されているファミリの類似エレメントに自動的にマップされる場合がありますが、使用する FPGA ファミリのエレメントを推論またはインスタンス化するようにソースコードを変更すると、そのエレメントにある追加機能または拡張機能を利用できます。これにより、エリア使用率およびパフォーマンスが向上する場合があります。	警告
Missing pipeline register	RPPR	レジスタを介していない出力のある乗算器が検出されました。レジスタのレベルを追加すると、乗算器の clock-to-out パフォーマンスを向上できます。また、これらのレジスタに非同期制御信号を使用しないようにするのがベストです。 レジスタを介していない出力のある RAM/ROM が検出されました。レジスタのレベルを追加すると、RAM/ROM の clock-to-out パフォーマンスを向上できます。また、これらのレジスタに非同期制御信号を使用しないようにするのがベストです。	警告
Inefficient pipeline register	RPIP	乗算ファンクションの入力または出力に非同期制御信号がある、レジスタ <i>register_name (file_name:line_number)</i> が検出されました。専用 DSP ハードウェア リソースには、プリセットやクリアなどの非同期制御信号はありません。レジスタを専用ハードウェア リソースにマップすると、デバイスが最適に使用されません。	警告
Found Black Box instance not belonging to UniSim library	RPBX	コンポーネントまたはモジュール <i>component/module_name</i> の記述を、合成で使用できませんした (<i>file_name:line</i>)。このブラック ボックスへのパスおよびブラック ボックスからのパスは最適化できません。合成ツールの使用率予測およびマップに悪影響を与える可能性があります。	警告
Found latch in design	RPLD	<i>signal_name (file_name:line_num)</i> 信号のラッチ記述が検出されました。ラッチがあると、インプリメンテーション後のシミュレーションが必要なタイミング パスを解析し、インプリメントされたデザインの動作が予測どおりのものであることを確認するのが困難になります。	警告
Found combinatorial loop in design	RPCL	<i>signal_name (file_name:line_number)</i> 信号の組み合わせループが検出されました。組み合わせループは、組み合わせロジックのコーンの出力が、同じコーンのロジックの部分入力としてフィードバックされると生成されます。ソースからデスティネーションまでの組み合わせ遅延合計は、フィードバックパス遅延分増やす必要があります。このタイプの構造は、デザインに予測される動作に必要であるか、または意図したものではない場合があります。	警告

フロアプラン DRC

Pblock フロアプランの DRC

表 B-3 : Pblock フロアプランの DRC

ルール名	略称	説明	重要度
Longest Carry Chain Height	LCCH	Pblock の高さに、割り当てられたキャリー チェーンで一番長いチェーンが収まるかどうかチェック	警告
Pblock overlap	FLBO	重なり合った Pblock の長方形のチェック	情報
Pblock Partition	FLBP	LUT から MUXCY および MUXFx への接続が Pblock のパーティションで切断されていないかをチェック	エラー
Resource Utilization	UTLZ	割り当てられたロジックに対して、Pblocks のリソースが十分であることをチェック	警告 (スライスロジックの場合) エラー (非スライスロジックの場合)
Area Group Tile Alignment	FLBA	AREA_GROUP 制約のサイト範囲が CLB グリッドと一致しているかどうかをチェック	警告

バンク DCI の DRC

表 B-4 : DCI カスケードの DRC

ルール名	略称	説明	重要度
DCI Cascade Checks	DCIC	DCI カスケード制約が有効かどうかをチェック	エラー
DCI Cascade with part compatibility	DCICPC	ユーザーに UCF ファイルをほかの互換性のあるパーツに読み込み、DRC を手動で実行して DCI カスケードが有効であることを確認するよう警告	警告
DCI check for I/O standard legality	DCICIOSTD	DCI カスケード内に使用される I/O 規格の VCCO と DCI 終端に関連する競合がないかどうかチェック	エラー

表 B-5 : IDelay Control の DRC

ルール名	略称	説明	重要度
IDelayCtrl Checks	IDLYCTRL	IDelay 配置が IDlyController のロケーションと一致しているかどうかをチェック	エラー

バンク IO の規格ルールのリストは、[466 ページの「I/O ポート/クロック ロジック/配置 DRC の詳細」](#)を参照してください。

ClkBuf の DRC

表 B-6 : ClkBuf の DRC

ルール名	略称	説明	重要度
BufR & BufIO Locations	BUFRIOC	同じリージョナル クロック ターミナルで駆動されている BUFR および BUFIO が相互配線可能なロケーションにあることをチェック	エラー

グローバル クロック ルールのリストは、[466 ページの「グローバル クロックの DRC」](#)を参照してください。

DSP48 の DRC

表 B-7 : DSP48 の DRC

ルール名	略称	説明	重要度
DSP48 output registers	DPOR	DSP48 の出力側にレジスタがありますが、このレジスタは同期制御して使用する必要があります(Virtex-4 のみ)。	情報
DSP48 input registers	DPIR	DSP48 の入力側にレジスタがありますが、このレジスタは同期制御して使用する必要があります (Virtex-4 のみ)。	情報
DSP48 output pipelining	DPOP	DSP48 の出力側にレジスタがありますが、このパイプラインのメカニズムを使用すると、パフォーマンスが向上します (Virtex-4 のみ)。	情報
DSP multiplier output pipelining	DMOP	DSP48 出力はパイプライン化されていません。パイプラインされるとパフォーマンスが向上します。	警告
DSP48 input pipelining	DPIP	DSP48 の入力側にレジスタがありますが、このパイプラインのメカニズムを使用すると、パフォーマンスが向上します(Virtex-4 のみ)。	情報
DSP48 cascade	DPCA	DSP48 カスケード チェック	警告
DSP48 asynchronous	DPREG	DSP48 非同期フィードバック	警告

RAMB16 の DRC

表 B-8 : RAMB16 の DRC

ルール名	略称	説明	重要度
RAMB16 output registers	RBOR	RAMB16 の出力側にレジスタがありますが、このレジスタは同期制御して使用する必要があります (Virtex-4 のみ)。	情報

RAMB DRC

表 B-9 : RAMB16 の DRC

ルール名	略称	説明	重要度
RAMB Read first mode	RBRF	READ_FIRST モードのクロック制限	警告

FIFO の DRC

FIFO DRC については、[表 B-10](#) を参照してください。

表 B-10 : FIFO の DRC

ルール名	略称	説明	重要度
FIFO Synchronous	FSYN	同期 FIFO のチェック	警告

ネットリストの DRC

ネットリストの DRC については、[表 B-11](#) を参照してください。

表 B-11 : ネットリストの DRC

ルール名	略称	説明	重要度
Driverless Nets	NDRV	各ネットに正しいドライバー ピンがあることをチェック	警告

インスタンスの DRC

インスタンスの DRC については、[表 B-12](#) を参照してください。

表 B-12 : インスタンスの DRC

ルール名	略称	説明	重要度
Black Box Instances	INBB	ブラックボックス (ネットリストの未定義ロジック) がないことをチェック	警告

属性の DRC

属性の DRC については、[表 B-13](#) を参照してください。

表 B-13 : 属性の DRC

ルール名	略称	説明	重要度
Invalid attribute	AVAL	無効な属性値をチェック	警告
Undefined attribute	ADEF	定義されていない属性値をチェック	警告

必須ピン DRC

必須ピンの DRC については、[表 B-14](#) を参照してください。

表 B-14 : 必須ピンの DRC

ルール名	略称	説明	重要度
Unconnected pin	REQP	接続されていない必須ピンをチェック	警告

I/O ポート/クロック ロジック/配置 DRC の詳細

PlanAhead で使用可能な I/O ポートおよびクロック ロジック DRC は、I/O 関連 DRC のほんの一部です。I/O ポートおよびクロック領域の仕様に関しては、使用しているデバイスの資料を参照してください。

PlanAhead で表示される問題の重要度は、ISE インプリメンテーション ツールでレポートされる同じ状況の重要度とは異なることがあります。

その他の DRC ルールの詳細については、[220 ページの「DRC の選択」](#)を参照してください。

グローバル クロックの DRC

表 B-15 は、グローバル クロックの DRC、略称、説明、重要度を示しています。

表 B-15 : グローバル クロックのルール

ルール名	略称	説明	重要度
IBUFG to DCM connectivity	IDCM	IBUFG には、デバイスの同じエッジ (上、下、左、右) にある DCM への専用配線のみがあります。	警告
DCM to BUFG connectivity	DCMB	DCM には最大 4 つの BUFG を接続できます。専用配線リソースを共有するバッファのペアが同じ DCM で駆動される場合、いずれかが非専用配線リソースを使用して駆動される必要があるため、これが原因でデザインでエラーが発生します。バッファに左から順に 1 ~ 8 の番号を付けると、1 と 5、2 と 6、3 と 7、4 と 8 という 4 組のペアができます。バッファがサイト 1 に配置されている場合は、同 DCM で駆動されるもう一方のバッファはサイト 5 に配置できません。	エラー
Number of BUFGs allowed for DCM	DCMN	DCM は最大 4 つの BUFG に接続できます。これは、DCMB に関連しています。	エラー
DCM and BUFG connectivity	DCME	BUFG にはデバイスの同じ側 (上、下、左、右) にある DCM への専用配線のみが存在します。	警告

配置ツールの DRC

配置ツールの DRC については、[表 B-16](#) を参照してください。

表 B-16 : 配置ツールの DRC

ルール名	略称	説明	重要度
Placement constraint	PLCR	クロック領域の配置制約をチェック	エラー
Clock placer	PLCK	クロック配置が有効な位置かどうかチェック。PLCK に含まれるのは、次のチェックです。 <ul style="list-style-type: none"> IOBUFR : リージョナル クロック ターミナルおよび関連 BUFR が相互配線可能なロケーションにあることをチェック IOBUFIO : リージョナル クロック ターミナルおよび関連 BUFIO が相互配線可能なロケーションにあることをチェック 	エラー
Design lock	PLDL	無効な LOC 制約をチェック	エラー

表 B-16 : 配置ツールの DRC

ルール名	略称	説明	重要度
Valid placement	PLVP	実行不可能な配置になる制約をチェック	エラー
Valid placement	PLIO	I/O の配置が有効かどうかをチェック	エラー

IOB の DRC

467 ページの表 B-17 は、IOB のルール、略称、説明、重要度を示しています。

表 B-17 : IOB の DRC

ルール名	略称	説明	重要度
Port Properties	PORTPROP	ポートのプロパティ内の矛盾をチェックします。	エラー
Differential IO pads	IODI	差動 I/O の P 信号および N 信号は、専用の差動ペアでは LOC 制約を付ける必要があります。	エラー
IO Standard Type	IOSTDTYPE	差動ペア I/O 規格が、差動ピンのペアにのみ割り当てられていることを確認します。	警告
Number of IOs	IOCNT	ターゲット デバイスにあるピンの数よりも多くの I/O ポートが定義されているかどうかを示します。	警告
I/Os placement	IOPL	許可されないサイトの I/O 配置をチェック。IOPL には、次のチェックが含まれます。 <ul style="list-style-type: none"> • IOPR : ポートが禁止ピンに配置されていないことをチェック • IOLVDSKC : 差動出力規格が、この規格をサポートしない低キャパシタンス サイトで使用されていないことをチェック 	エラー
IO Part Compatibility for Bank Type	IOPCBT	このメッセージは、互換性のあるパーツを複数含む 7 シリーズ デザインで、I/O 規格が 1 つのパーツに配置されたバンクでは有効なのに、互換性のある別のパーツの 1 つに配置された対応バンクでは無効な場合に表示されます。 この問題は、1 つのパーツのバンクが高パフォーマンス (HP) I/O バンクなのに、互換性のあるパーツの対応バンクが広範囲 (HR) I/O バンクの場合に発生します。有効な I/O 規格のセットは、これら 2 つのバンク タイプで同一ではありません。 この問題を回避するには、次を実行します。 <ul style="list-style-type: none"> • 互換性のあるパーツすべてで、その I/O 規格をサポートする I/O バンクにポートを配置します。 • HP バンクおよび HR バンクの両方で有効な I/O 規格に変更します。 • 別の互換性のあるパーツに変更します。 	警告
Prohibit not specified for part compatibility	IOPCPR	互換性のあるパーツを使用するデザインに対し、パッケージピンが互換パーツにないこと、「prohibit」とマークされ、何も配置されていないことをチェック	エラー

表 B-17 : IOB の DRC (続き)

ルール名	略称	説明	重要度
MGT not allowed for part compatibility	IOPCMGT	異なるシリアル トランシーバー 供給電圧を持つパーツ 2 つに互換性があるかどうかをチェックし、シリアル トランシーバーの使用を無効にします。	警告
I/O Crosstalk to MGT	IOCTMGT	I/O とシリアル トランシーバー間で発生する可能性のあるクロストーク問題をチェックします。	警告
I/O Bus SLR Crossings	IOBUSSLR C	次のメッセージは、同じバス ビットが別の SLR (Super Logic Regions) に配置されていると表示されます。 Bus port <BUSPORT[LO:HI]> spans more than one Super Logic Region (SLR). Bits placed in SLR <SLR1>:0-3 Bits placed in SLR <SLR2>:4-7 配線およびタイミング クロージャが困難になるので、これはお勧めしません。 すべての関連するバス ポートを同じ SLR に移動して警告が出ないようにしてください。	警告
Part compatibility	IOPCSLR	モノリシック デバイスとマルチダイ デバイス間のパーツの互換性をチェックします。	情報
IOB clock sharing	IOCS	IOB サイトはペアに分けられ、クロック配線リソースが共有できるようになっています。通常これらのペアは、LVDS ペアでもあります。フリップフロップが IOB にパックされている方法によっては、配線で問題が発生することがあります。この問題を解決するには、フリップフロップを特定の BEL に割り当てる必要があります。	警告
IOB set reset sharing	IOSR	IOB サイトには入力、出力、トライステート レジスタがあり、これらすべてで同じセット/リセット信号を共有しているので、別のリセット信号を使用してレジスタをバックすることはできません。	エラー

バンク I/O 規格の DRC

表 B-18 は、バンク I/O 規格のルール、略称、説明、重要度を示しています。

表 B-18 : バンク I/O 規格の DRC

ルール名	略称	説明	重要度
Bank IO Standard Vcc	BIVC	バンク内の I/O に対する IOSTANDARD ベースの VOUT の電圧互換性をチェックします。	エラー
Bank IO Standard Support	BIVB	I/O 規格が I/O バンクでサポートされていることをチェックします。	エラー
Bank IO standard Termination	BIVT	バンク内の I/O に対する IOSTANDARD ベースの DCI 終端の電圧互換性をチェックします。	エラー
Bank IO Standard VREF	BIVR	バンク内の I/O に対する IOSTANDARD ベースの VREF の電圧互換性をチェックします。	エラー

表 B-18 : バンク I/O 規格の DRC (続き)

ルール名	略称	説明	重要度
Bank IO Standard VREF Occupied	BIVRU	VREFを必要とする I/O 規格をインプリメントする I/O バンクに使用可能な VREF サイトがあるかどうかをチェックします。	警告
Bank I/O Standard	BIIVRC	バンクの I/O 規格と INTERNAL_VREF 制約間の競合をチェックします。バンクの規格には、バンクの INTERNAL_VREF 制約で指定される VREF 電圧と異なる VREF 電圧は要求できません。	警告
Bank I/O Simultaneous Switching Output Limits	BISLIM	I/O バンク内に配置されている I/O に対し同時スイッチ ノイズ (SSN) 出力をチェックします。	エラー
Bank I/O Standard VRN/VRP Occupied	DCIP	I/O バンクには専用の VRP サイトおよび VRN I/O サイトがあり、これらは通常の I/O としても使用できます。このバンクで DCI I/O 規格が使用される場合は、これらの I/O を未使用にしておく必要があります。	エラー
Inconsistent Diff pair I/O Standards	DIFFISTD	差動ペアのターミナルに同じ I/O 規格が設定されているかを確認します。	エラー
Inconsistent Diff pair I/O Standards	DIFFISTDDrv	差動ペアのターミナルに同じ駆動が設定されているかを確認します。	エラー
Inconsistent Diff pair I/O Standards	DIFFISTDSlew	差動ペアのターミナルに同じスローが設定されているかを確認します。	エラー
Vccaux Voltage requirement	VCCAUX1	LVC MOS25 の要件について警告します。	警告
Vccaux Voltage requirement	VCCAUX2	LVPECL_33 および TMDS_33 の要件について警告します。	エラー

ChipScope の DRC

表 B-19 は、ChipScope の DRC、略称、説明、重要度を示しています。

表 B-19 : ChipScope の DRC

ルール名	略称	説明	重要度
Unconnected channel	CSUC	接続のないチャンネルをチェックします。	エラー
Clocked by non-clock net	CSCL	デバッグ クロック ポートに接続されたクロック以外のネットをチェックします。	警告
Device block RAM	CSBR	ChipScope デバッグ コアをインプリメントするのに十分なブロック RAM リソースがデバイスに含まれているかどうかを確認します。	警告

XilinxNotify を使用したリリースのインストール

PlanAhead のリリース ストラテジ

PlanAhead™ では、新しい技術の導入やカスタマーのリクエストに迅速に対応するため、定期的に新規リリースが入手できるようになっています。定期的に新規リリースが入手できるようになっています。14.1 や 14.2 などのバージョン番号は、リリースを示します。[Help] → [About PlanAhead] をクリックすると、現在インストールされている PlanAhead のバージョンを確認できます。

新しいリリースを確認する場合は、[Help] → [Check for Updates] をクリックします。

ザイリンクス ツールのインストールの詳細は、付録 E 「その他のリソース」を参照してください。

XilinxNotify の実行

XilinxNotify は最新のソフトウェアを取得するのに推奨されるツールで、次が実行されます。

- <http://japan.xilinx.com/support/> にリリースされる最新のソフトウェア アップデートとインストールしているソフトウェアのバージョンを比較し、新しいバージョンが見つかった場合は通知します。
- [Download] ボタンをクリックすると、ブラウザーが起動し、ザイリンクス ダウンロード センターにログインする画面が表示されます。

ログインしたら、選択した製品のダウンロードが開始されます。XilinxNotify は、次のいずれかの方法で使用できます。

- ソフトウェア起動時に定期的に自動確認
- [Help] → [Check for Updates] をクリック
- Linux シェルに「xilinxnotify」と入力

注記：起動時の自動確認の頻度は、[Edit] → [Preferences] から設定できます。

XilinxNotify のネットワーク インストール

デフォルトでは、PlanAhead をネットワーク ディレクトリにインストールする際に使用したコンピューターでのみ、Xilinx.com からのアップデートが自動的にチェックされます。ネットワーク ロケーションにポイントしているクライアント マシンの場合は、デフォルトでオフになっています。

メイン メニューの [Tools] → [Options] コマンドをクリックし、ダイアログ ボックスの左側の [General] ボタンをクリックします (472 ページの図 C-1)。[Miscellaneous] カテゴリには、[Automatically check xilinx.com for software updates on startup] オプションが含まれます。または、メイン メニューから [Help] → [Check for Updates] をクリックすると、手動でチェックできます。

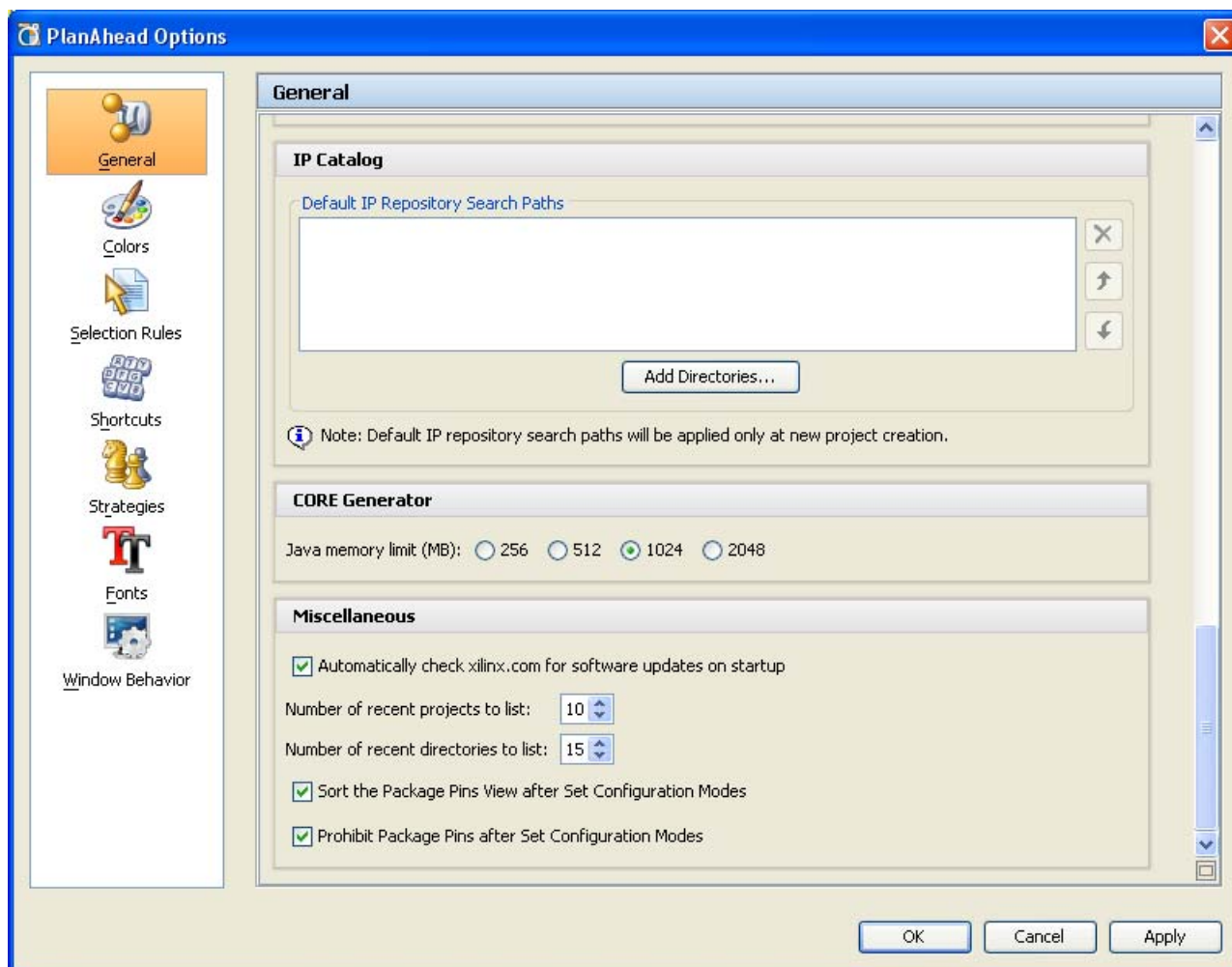


図 C-1 : [Automatically check xilinx.com for software updates on startup] オプション

注記：ソフトウェアアップデートをインストールする場合は、\$XILINX インストールディレクトリに書き込み権が必要です。

新規リリースは、ダウンロード センターに定期的にアップロードされます。

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

パスワード入力なしの SSH の設定

PlanAhead™ では、リモート ホストまたは複数のホストで同時に合成およびインプリメンテーションを実行できます。PlanAhead ツールで合成およびインプリメンテーション run を実行するための複数ホスト機能には、Linux OS で提供されているサービス、セキュア シェル (SSH) が使用されます。PlanAhead で複数のホストを設定する前に、リモート マシンにログインするたびにパスワードを入力しなくて済むように SSH を設定します。

SSH の設定

SSH は、Linux ターミナルまたはシェルで次のコマンドを入力して設定します。

注記：これは一度設定しておけば、繰り返し設定する必要はありません。

1. Linux ターミナルまたはシェルで次のコマンドを実行し、パブリック キーをプライマリ コンピューターで生成します。必須ではありませんが、セキュリティ保護のため、プライベート キーを入力および記憶しておくようにします。

```
ssh-keygen -t rsa
```

2. パブリック キーをリモート コンピューターの `authorized_keys` ファイルに追加します。`remote_server` をホスト名に変更します。

```
cat ~/.ssh/id_rsa.pub | ssh remote_server "cat - >> ~/.ssh/authorized_keys"
```

3. プライベート キーを取得するため次のコマンドを実行し、キーを有効にします。

```
ssh-add
```

これで、どのリモート コンピューターでもパスワードを入力せずに使用できます。新しいコンピューターに初めてアクセスする場合は、パスワードを入力するよう求められますが、次回からは入力する必要はありません。毎回パスワードの入力を求められる場合は、システム管理者に連絡し、パスワードを入力せずに SSH が使用できる Linux アカウントを設定してもらいます。

SSH が設定されたら、リモート ホストの設定に進んでください(Linux のみ)。

その他のリソース

次に、その他のリソースへのリンクをリストします。

ザイリンクス リソース

- デバイス ユーザー ガイド
http://japan.xilinx.com/support/documentation/user_guides.htm
- ザイリンクス用語集
<http://japan.xilinx.com/company/terms.htm>
- 『ザイリンクス デザイン ツール：インストールおよびライセンス ガイド』(UG798)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/iil.pdf
- 『ザイリンクス デザイン ツール：リリース ノート ガイド』(UG631)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/irn.pdf
- 製品サポートおよび資料
<http://japan.xilinx.com/support>

ハードウェア資料

- 7 シリーズ デバイスの資料
http://japan.xilinx.com/support/documentation/7_series.htm
- 『7 シリーズ FPGA SelectIO リソース ユーザー ガイド』(UG471)
http://japan.xilinx.com/support/documentation/user_guides/ug471_7Series_SelectIO.pdf
- 『Virtex-6 FPGA コンフィギュレーション ユーザー ガイド』(UG360)
http://japan.xilinx.com/support/documentation/user_guides/ug360.pdf
- 『Virtex-6 FPGA Select I/O リソース ユーザー ガイド』(UG361)
http://japan.xilinx.com/support/documentation/user_guides/ug361.pdf
- 『Spartan-6 FPGA コンフィギュレーション ユーザー ガイド』(UG380)
http://japan.xilinx.com/support/documentation/user_guides/ug380.pdf
- 『Spartan-6 FPGA Select I/O リソース ユーザー ガイド』(UG381)
http://japan.xilinx.com/support/documentation/user_guides/ug381.pdf
- 『Spartan-6 PCB デザイン ガイド』(UG393)
http://japan.xilinx.com/support/documentation/user_guides/ug393.pdf
- 『Virtex-5 FPGA コンフィギュレーション ユーザー ガイド』(UG191)
http://japan.xilinx.com/support/documentation/user_guides/ug191.pdf
- 『Virtex-4 FPGA コンフィギュレーション ユーザー ガイド』(UG071)
http://japan.xilinx.com/support/documentation/user_guides/ug071.pdf

ChipScope 資料

- 『ChipScope Pro ソフトウェアおよびコア ユーザー ガイド』(UG029)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/chipscope_pro_sw_cores_ug029.pdf
- 『PlanAhead チュートリアル : ChipScope を使用したデバッグ』(UG677)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/PlanAhead_Tutorial_Debugging_w_ChipScope.pdf
- 『ChipScope Pro ILA コアと Project Navigator を使用した FPGA アプリケーションのデバッグ』(UG750)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug750.pdf

EDK 資料

EDK の資料は、すべて次から入手できます。

http://japan.xilinx.com/support/documentation/dt_edk_edk14-1.htm

- 『EDK コンセプト、ツール、テクニック』(UG683)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/edk_ctt.pdf
- 『EDK プロファイリング ユーザー ガイド』(UG448)
http://japan.xilinx.com/support/documentation/xilinx14_1/edk_prof.pdf
- 『エンベデッド システム ツール リファレンス マニュアル』(UG111)
http://japan.xilinx.com/support/documentation/xilinx14_1/est_rm.pdf
- 『OS およびライブラリ資料コレクション』(UG643)
http://japan.xilinx.com/support/documentation/xilinx14_1/oslib_rm.pdf
- 『MicroBlaze プロセッサ ユーザー ガイド』(UG081)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/mb_ref_guide.pdf
- 『プラットフォーム別フォーマットのリファレンス マニュアル』(UG642)
http://japan.xilinx.com/support/documentation/xilinx14_1/psf_rm.pdf
- 『PowerPC 405 プロセッサ ブロック リファレンス ガイド』(UG018)
http://japan.xilinx.com/support/documentation/user_guides/ug018.pdf
- 『PowerPC 405 プロセッサ リファレンス ガイド』(UG011)
http://japan.xilinx.com/support/documentation/user_guides/ug011.pdf
- 『Virtex-5 FPGA の PowerPC 440 エンベデッド プロセッサ ブロック リファレンス ガイド』(UG200)
http://japan.xilinx.com/support/documentation/user_guides/ug200.pdf
- EDK チュートリアル ページ
http://japan.xilinx.com/support/documentation/dt_edk_edk14-1_tutorials.htm
- Platform Studio および EDK ページ
http://japan.xilinx.com/ise/embedded_design_prod/platform_studio.htm
- XPS/EDK でサポートされる IP ページ
http://japan.xilinx.com/ise/embedded/edk_ip.htm

ISE 資料

- ライブラリ ガイド
http://japan.xilinx.com/support/documentation/dt_ise14-1_librariesguides.htm
- ISE Design Suite 資料
http://japan.xilinx.com/support/documentation/dt_ise14-1.htm
 - 『コマンド ライン ツール ユーザー ガイド』 (UG628)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/devref.pdf
 - 『制約ガイド』 (UG625)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/cgd.pdf
 - 『Data2MEM ユーザー ガイド』 (UG658)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/data2mem.pdf
 - 『ISim ユーザー ガイド』 (UG660)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/plugin_ism.pdf
 - 『合成/シミュレーション デザイン ガイド』 (UG626)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/sim.pdf
 - 『タイミング クロージャ ユーザー ガイド』 (UG612)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug612.pdf
 - 『ザイリンクス/Cadence PCB ガイド』 (UG629)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/cadence_pcb.pdf
 - 『ザイリンクス/Mentor Graphics PCB ガイド』 (UG630)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/mentor_pcb.pdf
 - 『XPower Estimator ユーザー ガイド』 (UG440)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug440.pdf
 - 『XST ユーザー ガイド (Virtex-4, Virtex-5, Spartan-3, および CPLD デバイス)』 (UG627)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/xst.pdf
 - 『XST ユーザー ガイド (Virtex-6, Spartan-6, および 7 シリーズ デバイス)』 (UG687)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/xst_v6s6.pdf
- ISE 設計手法ガイド
 - 『消費電力手法ガイド』 (UG786)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug786_PowerMethodology.pdf
 - 『高集積度 FPGA 設計手法ガイド』 (UG872)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug786_PowerMethodology.pdf
- ISE チュートリアル
 - 『ISE チュートリアル』 (UG695)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ise_tutorial_ug695.pdf
 - 『RTL Viewer/Technology Viewer チュートリアル』 (UG685)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug685.pdf
 - 『ISim チュートリアル』 (UG682)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug682.pdf

- 『ChipScope Pro ILA コアと Project Navigator を使用した FPGA アプリケーションのデバッグ』(UG750)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug750.pdf
- 『消費電力ツール チュートリアル』(UG733)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug733.pdf

System Generator for DSP 資料

- 『System Generator for DSP リファレンス ガイド』(UG638)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/sysgen_ref.pdf
- 『System Generator for DSP 入門ガイド』(UG638)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/sysgen_gs.pdf
- 『System Generator for DSP ユーザー ガイド』(UG638)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/sysgen_user.pdf
- System Generator ウェブサイト
<http://japan.xilinx.com/tools/sysgen.htm>

パーシャル リコンフィギュレーション資料

- パーシャル リコンフィギュレーション ページ
<http://japan.xilinx.com/tools/partial-reconfiguration.htm>
- 『パーシャル リコンフィギュレーション ユーザー ガイド』(UG702)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug702.pdf
- チュートリアル
 - 『デザインの保持チュートリアル : PlanAhead デザイン ツール』(UG747)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/PlanAhead_Tutorial_Design_Preservation.pdf
 - 『パーシャル リコンフィギュレーション チュートリアル : PlanAhead デザイン ツール』(UG743)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/PlanAhead_Tutorial_Partial_Reconfiguration.pdf
 - 『プロセッサ ペリフェラルのパーシャル リコンフィギュレーション チュートリアル : PlanAhead デザイン ツール』(UG744)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/PlanAhead_Tutorial_Reconfigurable_Processor.pdf
- ホワイト ペーパー
 - 『Repeatable Results with Design Preservation』(WP362)
http://japan.xilinx.com/support/documentation/white_papers/wp362.pdf

アプリケーション ノート

- 『AXI マルチポート メモリ コントローラー』(XAPP739)
<http://japan.xilinx.com/support/documentation/topicaudiovideoimageprocess.htm>

- 『パーシャル リコンフィギュレーションを使用した PCI Express テクノロジの高速コンフィギュレーション』(XAPP883)
<http://japan.xilinx.com/tools/partial-reconfiguration.htm>
- 『PRC/EPRC : パーシャル リコンフィギュレーションのデータ インテグリティおよび セキュリティ コントローラー』(XAPP887)
<http://japan.xilinx.com/tools/partial-reconfiguration.htm>

PlanAhead 資料

- PlanAhead のユーザー ガイド
http://japan.xilinx.com/support/documentation/dt_planahead_planahead/14-1_userguides.htm
 - 『フロアプラン手法ガイド』(UG633)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/Floorplanning_Methodolgy_Guide.pdf
 - 『階層デザイン設計手法ガイド』(UG748)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/Hierarchical_Design_Methodolgy_Guide.pdf
 - 『ピン配置設計手法ガイド』(UG792)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug792_pinplan.pdf
 - 『PlanAhead Tcl コマンド リファレンス ガイド』(UG789)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug789_pa_tcl_commands.pdf
- PlanAhead チュートリアル
http://japan.xilinx.com/support/documentation/dt_planahead_planahead14-1_tutorials.htm
 - 『クイック フロー概要チュートリアル : PlanAhead デザイン ツール』(UG673)
 - 『I/O ピン配置チュートリアル : PlanAhead デザイン ツール』(UG674)
 - 『RTL デザインおよび IP の生成チュートリアル : PlanAhead デザイン ツール』(UG675)
 - 『デザイン解析およびフロアプラン チュートリアル : PlanAhead デザイン ツール』(UG676)
 - 『PlanAhead ツール チュートリアル : ChipScope を使用したデバッグ』(UG677)
 - 『チーム デザイン チュートリアル : PlanAhead デザイン ツール』(UG839)
 - 『デザインの保持チュートリアル : PlanAhead デザイン ツール』(UG747)
 - 『パーシャル リコンフィギュレーション チュートリアル : PlanAhead デザイン ツール』(UG743)
 - 『プロセッサ ペリフェラルのパーシャル リコンフィギュレーション チュートリアル : PlanAhead デザイン ツール』(UG744)

Zynq 資料

- 『Zynq-7000 エクステンシブル プロセッシング プラットフォームの概要』(DS190)
http://japan.xilinx.com/support/documentation/data_sheets/j_ds190-Zynq-7000-Overview.pdf
- 『Zynq-7000 EPP ソフトウェア開発者向けガイド』(UG821)
http://japan.xilinx.com/support/documentation/user_guides/ug821-zynq-7000-swdev.pdf

- 『Zynq EPP コンセプト、ツール、テクニック ガイド』(UG873)
http://japan.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ug873_zynq_ett.pdf
- 『USB ケーブル インストール ガイド』(UG344)
http://japan.xilinx.com/support/documentation/user_guides/ug344.pdf
- 『Platform Cable USB II データシート』(DS593)
http://japan.xilinx.com/support/documentation/data_sheets/ds593.pdf

IP 資料

- ザイリンクス IP センター
<http://japan.xilinx.com/ipcenter/>
- ザイリンクス AXI IP ウェブサイト
http://japan.xilinx.com/support/documentation/axi_ip_documentation.htm
- 『AXI リファレンス ガイド』(UG761)
http://japan.xilinx.com/support/documentation/ip_documentation/axi_ref_guide/v14_1/ug761_axi_reference_guide.pdf