

# RTL設計中級 論理合成 … 目次

## 第0章 論理合成とは

1. 論理合成とは
2. 従来の論理設計のフロー
3. HDLを使った論理設計のフロー
4. 論理合成のメリット
5. 論理合成ツールの限界
6. 論理合成可能な記述
7. ASICとFPGAの合成ツール
8. ASICとFPGAの合成手法の違い
9. ツールやバージョンによる結果の違い

## 第1章 論理合成を前提としたRTL記述

1. 第1章 アジェンダ
2. 論理合成の基本フロー
3. RTL解読
4. 算術演算回路
5. 算術演算の記述
6. 演算子使用時の注意点
7. 加算演算子の回路構成
8. 演算子の共有
9. case文から生成されるセレクタ
10. if文から生成されるセレクタ
11. default項に不定値を代入
12. default項に固定値を代入
13. ラッチを生成する危険を避ける(1)
14. ラッチを生成する危険を避ける(2)
15. ラッチを生成する危険を避ける(3)
16. ラッチを生成する危険を避ける(4)
17. for文記述
18. for文による組み合わせ回路の記述
19. FFの記述
20. FF記述の制限
21. 代入式右辺の遅延値
22. 10進カウンタ記述
23. ブロッキング代入とノンブロッキング代入
24. ラッチの記述
25. ゲート vs. RTLシミュレーション不一致問題(1)
26. ゲート vs. RTLシミュレーション不一致問題(2)
27. ゲート vs. RTLシミュレーション不一致問題(3)
28. ゲート vs. RTLシミュレーション不一致問題(4)

## 第2章 論理合成の基本とスクリプト

1. 第2章 アジェンダ
2. 論理合成の対象(1)
3. 論理合成の対象(2)
4. 論理合成の対象(3)
5. 論理合成の基本フロー
6. 論理合成のアルゴリズム
7. dc\_shell と Design Vision(1)
8. dc\_shell と Design Vision(2)
9. dc\_shellスクリプト
10. 標準スクリプト
11. 論理合成基本コマンド(1)
12. 論理合成基本コマンド(2)
13. 論理合成基本コマンド(3)
14. オブジェクト
15. タイミングのパス
16. report\_area
17. report\_reference
18. report\_timing
19. report\_timing 2
20. report\_qor
21. report\_qor 2
22. 論理合成基本コマンド(4)
23. 論理合成の制約
24. 論理合成基本コマンド(5)
25. 制約条件 - set\_operating\_conditions
26. 制約条件 - set\_wire\_load\_model
27. 制約条件 - ポート名の指定
28. 制約条件 - set\_driving\_cell
29. 制約条件 - set\_load
30. 制約条件 - create\_clock
31. 制約条件 - set\_input\_delay, set\_output\_delay
32. 制約条件 - set\_ideal\_network
33. report\_lib
34. allコマンド
35. getコマンド
36. クロックとリセット
37. 組み合わせ回路の制約

## 第3章 RTL記述スタイル

1. 第3章 アジェンダ
2. 回路構造を意識した記述
3. always文による組み合わせ回路記述の注意点
4. always文内に複数の信号を記述しない
5. if文のスタイル1
6. if文のスタイル2
7. case文のスタイル
8. リソースシェアリング
9. リソースシェアリングされない記述
10. リソースシェアリングの注意点(1)
11. リソースシェアリングの注意点(2)
12. ステートマシン
13. ミーリ型ステートマシン
14. ムーア型ステートマシン
15. 基本はムーア型
16. ステートマシンの記述例
17. ステートマシンの記述例(つづき)
18. 何故、FFとcase文を分離するのか?
19. ステートマシン(バイナリー型)
20. ステートマシン(One Hot型)
21. パラメタライズ記述

## 付録 Design CompilerでのTcl

1. 簡単なコマンド例
2. 変数の設定と参照
3. 特殊文字
4. リストとコレクション
5. コレクション
6. get\_object\_name
7. 文字列の作成
8. リストの操作
9. 算術演算
10. スクリプトファイルの読み込み
11. 出力先の変更
12. フロー制御
13. ループ foreach
14. ループ foreach\_in\_collection
15. ループ while
16. ループ for
17. 条件分岐 if
18. ループの終了

## 付録2 ゲートレベルでのデバッグ

## 第4章 論理合成の実際

1. 第4章 アジェンダ
2. 実践的な論理合成
3. 実践的な合成スクリプト例(1)
4. 実践的な合成スクリプト例(2)
5. 実践的な合成スクリプト例(3)
6. 基本はトップダウンの一括合成
7. 余計な階層は破壊する(autoungroup)
8. ungroupコマンド
9. dont\_touchコマンド
10. Design Ware
11. タイミング解析
12. 最大遅延解析(1)
13. 最大遅延解析(2)
14. 最大遅延解析(3)
15. 最大遅延解析(4)
16. 最大遅延解析(5)
17. タイミング解析のパス
18. タイミング解析のパス 2
19. タイミングレポート - in2out (FEEDTHROUGH)
20. タイミングレポート - in2ff (REGIN)
21. タイミングレポート - ff2ff
22. タイミングレポート - ff2out (REGOUT)
23. クロックの遅延
24. クロックスキューによる誤動作
25. ホールドタイム保証
26. クロックツリーシンセシス(GTS)
27. set\_fix\_multiple\_port\_netsコマンド
28. change\_namesコマンド
29. ログファイルとレポートファイルの確認
30. Tcl - システムコマンド
31. Tcl - foreach文
32. Tcl - if文
33. helpとman

## DesignVisionクイックリファレンス