

第1章 Tclスクリプト

1. Tclとは
2. 簡単なコマンド例
3. 変数の設定と参照
4. 特殊文字
5. リストとコレクション
6. コレクション
7. `get_object_name`
8. 文字列の作成
9. リストの操作
10. UNIXコマンドの実行
11. 算術演算
12. スクリプトファイルの読み込み
13. 出力先の変更
14. フロー制御
15. ループ `foreach`
16. `file`
17. `glob`
18. ループ `foreach_in_collection`
19. ループ `foreach_in_collection 2`
20. ループ `while`
21. ループ `for`
22. 多件分岐 `if`
23. 多重分岐 `switch`
24. ループの終了
25. セットアップファイルでの使用例

第2章 基本ブロック合成と階層合成

1. 基本ブロック(基本階層)の考え方1
2. 基本ブロック(基本階層)の考え方2
3. 悪い階層の組み方
4. 基本階層からの論理合成
5. 階層合成のフロー(階層コンパイル)
6. 面積優先からの合成手法
7. 合成結果の把握
8. タイミングレポート
9. 配線面積
11. 駆動能力の制約は必要
12. 基本階層の論理合成標準スクリプト
13. 合成スクリプトのポイント1
14. 合成スクリプトのポイント2
15. 動作環境MAX, MIN
16. `set_ideal_network`の設定
17. 階層の合成(1)
18. 論理合成による出力ポート処理
19. 階層の合成(2)
20. 階層の合成(3)

第3章 タイミング解析の考え方

1. `input_delay`, `output_delay`制約
2. `setup`解析
3. `hold`解析
4. `setup`解析(`set_input_delay -max`)
5. `hold`解析(`set_input_delay -min`)
6. `setup`解析(`set_output_delay -max`)
7. `hold`解析(`set_output_delay -min`)
8. スタティックタイミング解析
9. 内部バスへの入力本数を多くしない
10. `set_false_path`, `set_multicycle_path`
11. `set_multicycle_path`の解析
12. マルチサイクルパスは避ける
13. `case_analysis`タイミング解析
14. クロックツリーシンセシス(CTS)
15. クロックツリーの扱い
16. ホールド保証(1)
17. ホールド保証(2)
18. クロックラインのトランジッションタイム
19. タイミング解析の切断
20. Clock Gatingタイミング解析(DC)
21. レイアウト後のタイミング解析

第4章 複数クロックの合成

1. `set_input_delay`の注意事項
2. 分周クロックの制約
3. ゲーテッドクロックの制約
4. `set_false_path`の利用
5. 非同期通信回路
6. 非同期クロック間の比率を考える
7. クロック設定の誤り
8. クロック生成モジュール

第5章 論理合成テクニック

1. 合成工程での流れ
2. 平坦化合成(flatten)
3. 構造化合成(structure)
4. プーリアンオブティマイズ
5. case文の論理合成
6. 平行ロジックの記述
7. flattenを利用する為には、
8. 面積優先の論理合成スクリプト
9. compile -area_effort high
10. 合成フロー一例
11. 速度向上の為のテクニック
12. group_pathの利用
13. TNS機能
14. 単純2回合成
15. ungroup後のcompile
16. データバス設計
17. 信号処理系の合成
18. 演算器の階層破壊
19. compile_ultraの効果
20. 演算器の再構築
21. リタイミング機能
22. 指定した方がよい変数

第6章 DC新機能とバージョンによる結果の違い

1. 合成の実行速度
2. 最近の性能改善は、固定レジスタ削減がメイン
3. 相互依存の固定レジスタの削減(DC-Ultra Only)
4. レジスタのマージ機能(DC-Ultra Only)
5. クリティカルパス合成
6. -xg_mode(2004.06)
7. -xg_modeの変更点
8. トポグラフィカル合成(2006.06)
9. compile -auto_ungroup(2003.10)
10. report_areaコマンドの拡張
11. report_area -heirarchyコマンド
12. set_clock_groups (2007.03)
13. 分散処理機能(2008.06)
14. SystemVerilogサポート(2004.12)
15. always_comb
16. always_ff と always_latch
17. priority
18. unique
19. モジュールの記述
20. データタイプの拡張
21. 基本データタイプの拡張
22. インターフェースの記述 (1)
23. インターフェースの記述 (2)
24. インターフェースの呼び出しと接続 (1)
25. インターフェースの呼び出しと接続 (2)

付録 FPGA合成

1. FPGA基本開発フロー
2. 論理式抽出とLUTマッピング
3. ASICとFPGAの合成手法の違い
4. 論理合成の対象(ASIC)
5. 論理合成の対象(FPGA)
6. FPGA基本構成
7. FPGA基本構成図
8. LE(ロジック・エレメント)基本構成
9. 内臓RAM
10. RAMの推論(記述例)
11. クロック位相制御
12. クロックの逡倍
13. IOB(入出力バッファ)
14. FPGA基本制約
15. 論理合成制約
16. 配置配線制約
17. タイミング制約(1)
18. タイミング制約(2)
19. タイミング制約① クロック
20. タイミング制約① PLLクロック
21. タイミング制約② 入力遅延
22. タイミング制約③ 出力遅延
23. タイミング制約④ 入力端子から出力端子
24. STAエラー対策(レジスタ リタイミング)
25. STAエラー対策(クリティカル・パス)
26. STAエラー対策(レジスタの複製)
27. 合成レポート ザイリンクス(1)
28. 合成レポート ザイリンクス(2)
29. 合成レポート ザイリンクス(2)
30. 合成レポート ザイリンクス(3)
31. 合成レポート ザイリンクス(4)
32. 合成レポート アルテラ(1)
33. 合成レポート アルテラ(2)
34. 合成レポート アルテラ(3)
35. Verilog HDL サポート規格
36. 参考資料 ザイリンクス
37. 参考資料 アルテラ

演習問題