

目次

第1章 検証技術のトレンド

1. 検証技術のトレンド
2. TLMの導入
3. SoC開発の現状
4. 仕様および実装上の問題点
5. アルゴリズム設計・検証の導入
6. TLMによる設計・検証の導入
7. TLMの必要性とメリット
8. TLM(Transaction Level Modeling)とは
9. 抽象レベル
10. 抽象レベルの分類
11. TLMの課題
12. 新たな検証メソッドの導入
13. 検証環境
14. 制約付きランダム
15. 機能カバレッジ
16. アサーション
17. 検証メソッドの提案
18. テスト仕様とテスト項目の作成方法
19. 非同期回路の検証
20. まとめ

第2章 制約付きランダム

1. 検証環境と制約付きランダム
2. ランダムの導入
3. ランダム生成の種類
4. ランダム制約の種類
5. ランダム生成と制約 - Verilog HDL -
6. \$random - 制約なし -
7. \$random - シード制約 -
8. LFSR
9. ランダム生成と制約 - SystemVerilog -
10. \$urandom - シード制約 -
11. \$urandom_range - 範囲制約 -
12. std::randomize() - 制約無し -
13. std::randomize() - シード制約 -
14. std::randomize()with - 範囲制約 -
15. std::randomize()with - 重み制約 -
16. classの使用法
17. class - シード制約 -
18. class - 範囲制約 -
19. class - 重み制約 -
20. class - 制約の外部宣言 -
21. class - 周期性を持ったランダム -
22. ランダム生成関数のまとめ
23. 制約付きランダムの活用

第3章 機能カバレッジ

1. 検証環境と機能カバレッジ
2. カバレッジの種類
3. コードカバレッジの問題点
4. 機能カバレッジの導入(1)
5. 機能カバレッジの導入(2)
6. 機能カバレッジの種類
7. データ指向カバレッジの種類
8. データ指向カバレッジの使用方法
9. データ指向カバレッジ - 自動レンジ -
10. データ指向カバレッジ結果 - 自動レンジ - (1)
11. データ指向カバレッジ結果 - 自動レンジ - (2)
12. データ指向カバレッジ結果 - 自動レンジ - (3)
13. データ指向カバレッジ - サンプル条件 -
14. データ指向カバレッジ結果 - サンプル条件 - (1)
15. データ指向カバレッジ結果 - サンプル条件 - (2)
16. データ指向カバレッジ結果 - サンプル条件 - (3)
17. データ指向カバレッジ - レンジ指定 -
18. データ指向カバレッジ結果 - レンジ指定 - (1)
19. データ指向カバレッジ結果 - レンジ指定 - (2)
20. データ指向カバレッジ結果 - レンジ指定 - (3)
21. データ指向カバレッジ - トランジション -
22. データ指向カバレッジ - クロス -
23. データ指向カバレッジ結果 - クロス -
24. データ指向カバレッジのまとめ
25. コントロール指向カバレッジ
26. コントロール指向カバレッジとSVA
27. コントロール指向カバレッジの使用方法
28. カウンタにおけるカバレッジの記述例(1)
29. カウンタにおけるカバレッジの記述例(2)
30. インターフェースにおけるカバレッジの記述例
31. コントロール指向カバレッジ結果(1)
32. コントロール指向カバレッジ結果(2)
33. コントロール指向カバレッジのまとめ
34. 機能カバレッジのまとめ

第4章 アサーション

1. 検証環境とアサーション
2. アサーションとは
3. アサーションのメリット(1)
4. アサーションのメリット(2)
5. アサーションのメリット(3)
6. アサーションのメリット(4)
7. アサーションを適用する回路機能
8. アサーションでチェックする内容
9. アサーションの分類
10. アサーションの構文

目次

11. プロパティ仕様(1)
12. プロパティ仕様(2)
13. カウンタにおけるアサーション記述例(1)
14. カウンタにおけるアサーション記述例(2)
15. インターフェースにおけるアサーション記述例
16. アサーションの結果とメッセージ(Questa)
17. アサーション結果の波形表示(Questa)
18. アサーションの結果とメッセージ(VCS)
19. アサーションの結果とメッセージ(IES)
20. アサーションまとめ

第5章 テスト仕様書と項目作成の考え方

1. テスト仕様書作成の要点
2. サンプル回路
3. テスト項目の列挙
4. テスト詳細仕様
5. テスト詳細仕様(サブルーチン)
6. タスク記述1 (AllDataLoadFromISA)
7. タスク記述2 (DataLoadFromISA)
8. タスク記述3(write_ISA)
9. シミュレーションでの確認方法
10. 波形確認方式では、
11. 前回結果との比較方式
12. チェックリストによる承認
13. 入力パターンに基づくテスト項目の作成
14. 確認事項に基づくテスト項目の作成
15. 確認事項とテスト項目の関係
16. 確認事項の抽出方法(入力信号と機能のマトリックス表)
17. 確認事項の抽出方法(出力信号による確認事項の明確化)
18. 仕様とテスト仕様との関係
19. 統計解析:バグ曲線グラフ
20. 好ましくないバグ曲線グラフ
21. テスト項目の優先順位
22. まとめ

第6章 テスト項目作成手順

1. 検証漏れのないテスト項目を作成するために
2. 機能のリストアップ(1)
3. 機能のリストアップ(2)
4. 入出力/レジスタのリストアップ
5. 動作フローのリストアップ
6. テスト項目の作成(単体機能テスト)
7. テスト項目の作成(組み合わせテスト)
8. テスト項目の作成(例外テスト)
9. テスト項目の形式
10. テスト仕様書の作成
11. まとめ

第7章 非同期回路の検証

1. 非同期回路
2. 非同期部の誤動作(1)
3. 非同期部の誤動作(2)
4. 非同期部の対策(1)
5. 非同期部の対策(2)
6. 非同期部の対策(3)
7. 非同期部の対策(4)
8. 非同期部の対策(5)
9. 非同期回路の検証方法
10. デレイラインの利用
11. デレイラインの記述例
12. ジッタモデルの利用
13. ジッタモデルの記述例(1)
14. ジッタモデルの記述例(2)
15. ジッタモデルの記述例(3)
16. 非同期回路の検証のまとめ
17. 講座のまとめ

OSDチップの概要

演習