

SILOS-X の LINT 機能

はじめに

LINT ツールを用いて、論理回路における構文エラーおよびデザインの潜在的な問題点(競合状態、合成不可能なデザイン、不必要なコンポーネントなど)を確認できます。LINT 機能は後工程の段階で検出される潜在的な問題点をデザインの初期段階で検出できるので、論理回路デザインの設計効率が大幅に向上されます。Simucad 社で開発された Turbolint が LINT 機能として対応していましたが、新しいツール SILOS-X に LINT 機能として組み込まれました。これにより、SILOS-X はただ単なるシミュレーション・ツールではなく、デザイン品質の向上を支援するパワフルなツールになりました。本稿は、例を用いて、LINT 機能がどのように動作するかを説明します。

LINT 機能

ほとんどのシミュレーション・ツールでは、基本的なシンタックスおよびセマンティック・チェックを実行できます。しかし、このような基本的なチェックでは、デザインが文法上正しいかどうかのみが示され、デザインの品質についてはほとんど確認できません。LINT 機能は数多くのチェック・ルールに対応しており、シンタックスやセマンティックだけでなく数多くのカテゴリをチェックします。

(1) LINT 機能によるデザインの競合状態チェック

例

```
module m(clk, a, z_r);
  input clk;
  input a;
  output z_r;
  reg z_r;
  reg q_r;
  always @(posedge clk) begin
    z_r = q_r;
  end
  always @(posedge clk) begin
    q_r = a;
  end
endmodule
```

この例では、2 つの always ブロックが存在します。1 つは reg “q_r”を読み込み、もう 1 つは reg を書き込みます。always ブロックの評価順序は決まっていないので、読み込みが書き込み前または後で発生します。この実行の2つの順序(読み込み→書き込み/書き込み→読み込み)は異なるシミュレーション結果を導きます。これが、LINT 機能が検出する Verilog 競合状態です。

(2) LINT 機能によるハードウェア解析

レジスタ、ラッチ、ステートマシン、その他の順序素子が推定されると、LINT 機能は警告を出力します。したがって、余分な同期回路が望まない場所に合成されたかどうかをチェックできません。

例

```
module m(SELECT, IN_A, IN_B, X, Y);
  input [1:0] SELECT;
  input IN_A;
  input IN_B;
  output X;
  output Y;

  reg X;
  reg Y;

  always @(SELECT or IN_A or IN_B) begin
    if (SELECT==2'b00) begin
      X = IN_A | IN_B;
      Y = 0;
    end
    else if (SELECT ==2'b01) begin
      Y = IN_A & IN_B;
    end
    else if (SELECT==2'b10) begin
      X = 0;
      Y = 1;
    end
  end
end
```

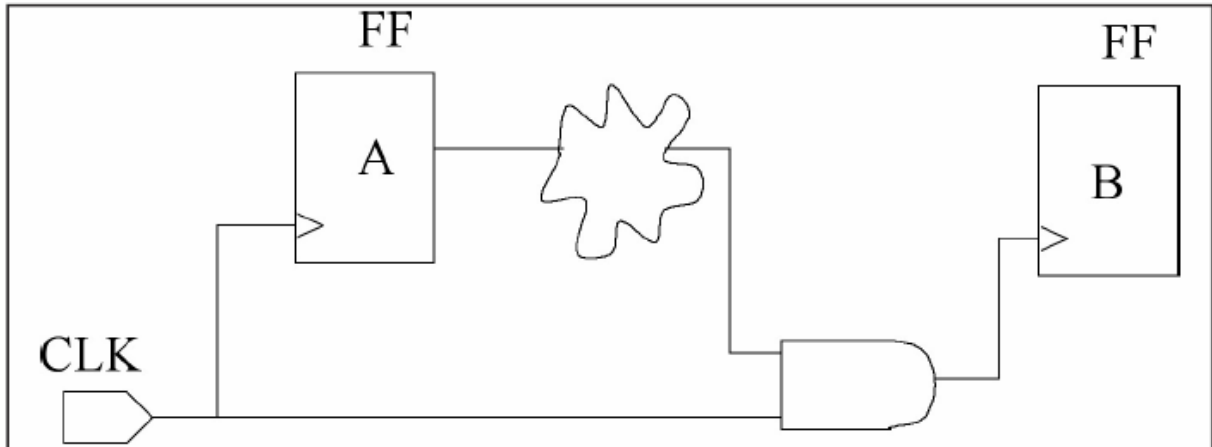


図 1

```

else begin
    X = 0;
    Y = 0;
end
end

endmodule

```

この例では、SELECT = 2'b01 のとき、X への割り当て表現は定義されていません。したがって、この場合、X の値を保持するためラッチが生成されます。LINT 機能は、後続のチェックのためにこの問題点をユーザに警告します。

また、合成不可能なデザインおよび合成前後でシミュレーション結果が一致しないデザインを指摘します。

(3) LINT 機能による DFT 向けチェック

DFT (Design For Test)は、正常な動作をするデザインには非常に重要です。DFT によりフォルト・テストを容易に実行でき、デザイン品質を向上できます。LINT 機能はテストが困難または不可能であるデザイン上の問題となる場所を検出します。

このデザインでは、2 番目の FF のクロック端子が外部の入力ポートから直接制御できないため、スキャン・インサート・ツールがこの FF をスキャン対象から除外します。したがって、ATPG ツールを用いて、この部分のフォルトを検出するのは困難になります。LINT 機能は、初期段階でこの問題点を検出します。

(4) LINT 機能によるデザインの移植性および可読性の向上

LINT 機能は、ユーザが読みやすく容易に変更可能なデザインを記述できるチェック・ルールを装備しています。

例

```
output [31: 0] X;
```

この例では、LINT 機能はハードコーディングされた数値定数を直接使用するのではなく、数値を用いた定数を定義するように導きます。特に、このような定数が多く使用される場合、定義された定数を使用することにより記述の変更が非常に容易になります。

例

```
Reg abc;
Reg ABC;
```

この例では、同一の名前が大文字・小文字で区別され、エラーになる傾向があり、コードの記述が不明瞭になるので、LINT 機能は警告を出力します。

LINT 機能がチェックするカテゴリをいくつか紹介しましたが、すべてのカテゴリではありません。SILOS-X の LINT 機能は、500 以上のチェック・ルールを装備しており、STARC とのデザイン互換性もチェックできます。

まとめ

上記の例を用いて、LINT 機能がデザインの品質と効率を向上するパワフルな機能であることを説明しました。LINT 機能は、経験の浅い設計者が正しいスタイルでデザインを記述する方法や潜在的なミスを防ぐ方法を学ぶために役立ちます。また、熟練の設計者は、LINT 機能を用いて、複雑なデザインにおける問題となるコードを検出することでデザインの効率を向上し、コードの品質を向上できます。LINT 機能を備えた SILOS-X は、パワフルな設計ツールです。あらゆる設計者は、この機能を活用してデザイン品質を向上できます。