

時間記号論理シミュレーションにおける
遅延モデルの拡張

5S-7

高橋瑞樹, 石浦菜岐佐, 矢島脩三

京都大学工学部

1.はじめに

論理回路のタイミング検証の一手法として、我々は時間記号論理シミュレーションを提案^[1]している。以前に提案した時間記号論理シミュレーションでは、ゲートの遅延モデルとして標準遅延モデルを用いていた。しかし、正確なタイミング検証を行うという観点から考えると、より現実に近い遅延モデルを用いるのが望ましい。そこで、本稿では立上り/立下り遅延等の種々の遅延モデルを用いた時間記号論理シミュレータについて考察する。

2.時間記号論理シミュレーション^[2]

時間記号論理シミュレーションでは、ゲートの遅延時間、入力の変化時刻を変数で表わしシミュレーションを行う。これを、時間変数と呼ぶ。各時間変数には、その取り得る値の範囲があり、それを変数制限条件という。信号線の信号値系列は、イベントまたは代数式でラベル付けされた節点からなるイベント木で表わされる。シミュレーションは、回路内の各ゲートについて、入力のイベント木から出力のイベント木を計算することにより行われる。イベントの発生時刻が定数ではなく変数であることに起因して、入力イベントの前後関係によって場合分けが行われ、出力のイベント木上に分岐が生ずる。この際、変数制限条件とそこまでのパス条件(その時点までの分岐条件の集合)より、注目している2つのイベントの前後関係が一意に決まり分岐が生じない場合がある。時間記号論理シミュレータでは、変数制限条件、パス条件、分岐条件を時間変数に関する連立一次不等式とみなし、分岐の可能性の判定を、線形計画法を用いた連立一次不等式の解の存在判定により行う。また、時間記号論理シミュレーションでは、シミュレーション結果から回路が正常動作するための条件を得ることができるが、これは作成した結果解析系^[3]により求められる。

3.立上り/立下り遅延を用いた時間記号論理シミュレーション

3.1 アルゴリズムとその実現

これまでの時間記号論理シミュレーションでは、各ゲートの遅延に一つの変数が割り当てられる、標準遅延モデルを用いていた。しかし実際には、ゲートの遅延時間は信号の立上り、立下りで異なる。

立上り/立下り遅延を用いた時間記号論理シミュレーションは、基本的には、各ゲートに対して立上り遅延と立下り遅延を表わす2つの時間変数を与え、信号値の変化に応じてどちらかの遅延を用いて出力の変化時刻を計算するという処理を行うことにより実現できる。このとき、問題となるのはイベントのキャンセル処理である。遅延、入力の変化時刻を定数で与えた場合でも、単純に出力イベントを計算したのでは、

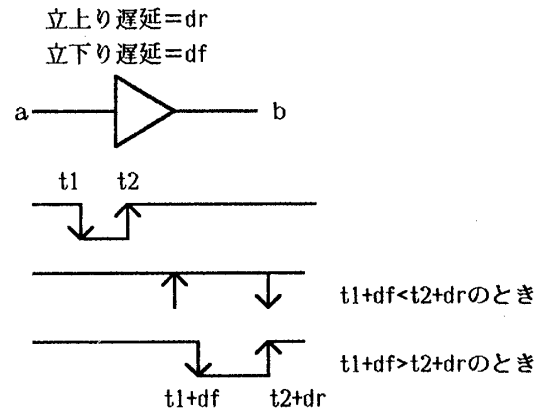


図1 時間が変数の場合のイベントのキャンセル

pe:前のイベント(初期イベントのときはpe=NULL)
e:現在のイベント
pe.time, e.time:それぞれのイベントの発生時刻

```

if (pe==NULL) {
    pe=e;
    次のイベント評価(pe);
}
else {
    switch(イベントキャンセルの可能性) {
        case キャンセルなし:
            peを出力イベント木につなぐ;
            pe=e;
            次のイベント評価(pe);
            break;
        case キャンセルあり:
            pe=NULL;
            次のイベント評価(pe);
            break;
        case 両方の場合あり:
            (pe.time)-(e.time)の条件節点を出力木につなぐ;
            pe.time<e.timeのパスに対して {
                peを出力イベント木につなぐ;
                pe=e;
                次のイベント評価(pe);
            }
            pe.time>e.timeのパスに対して {
                pe=NULL;
                次のイベント評価(pe);
            }
    }
}
    
```

図2 イベントキャンセルのアルゴリズム

Extention of delay models
for time-symbolic logic simulation

Mizuki Takahashi, Nagisa Ishiura and Shuzo Yajima
Faculty of Engineering, Kyoto University

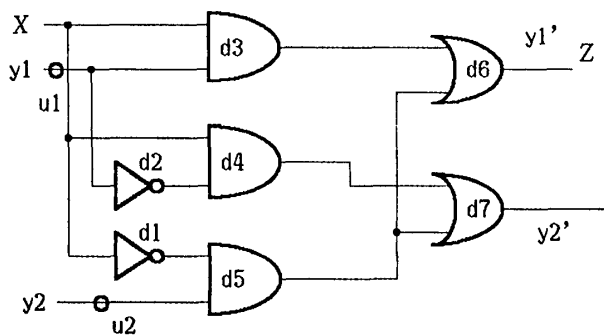


図3 ループを切断したT-F F回路

あるイベントが前のイベントを追い越す場合がある。そのため、イベントのキャンセル処理が必要となるが、時間が変数の場合には、定数の場合と違って、立上り（立下り）が立下り（立上り）を追い越すかどうかは、一意には決まらないので場合分けが必要になる（図1）。キャンセル処理の場合分けは、ゲートの出力イベント木の計算の際に、求められた出力イベントをイベント木につなぐのを、次の出力イベントが計算される時点まで遅らせることにより実現できる。出力イベントをイベント木につなぐ部分のアルゴリズムは図2のようになる。

上記のアルゴリズムに基づき、以前に実現した時間記号論理シミュレータの一部を変更して、立上り/立下り遅延を扱うことができる時間記号論理シミュレータを、Sun 3/60上にC言語で実現した。

3.2 実行例

図3のT-FFのループを切断した回路に、図4の4入力パターンを与えた場合の正常動作のための条件を、割当て遅延の場合と立上り/立下り遅延の場合について求めた。

標準遅延を用いた従来の時間記号論理シミュレータで、全てのゲートの遅延の範囲が2~8、 $t_1 \sim t_4$ は正という変数制限条件のもとでシミュレーションを行ったとき、正常動作の条件は次のようになる。

$$t_1 = d_4 + d_7, t_2 = d_1 + d_5 + d_6, t_3 = d_1 + d_5 + d_7, t_4 = d_3 + d_6 \\ d_1 + d_5 < d_4, d_1 + d_5 > d_3$$

また、立上り/立下り遅延を用いた時間記号論理シミュレータで、各ゲートの立上り遅延が3~8、立下り遅延が2~7、全てのゲートについて(立上り遅延) > (立下り遅延)、 $t_1 \sim t_4$ は正という変数制限条件でシミュレーションを行うと次のような条件が得られる。ただし、各ゲート立上り遅延、立下り遅延は、時間変数の後にそれぞれ r, f をつけて表している。

$$t_1 = d_4r + d_7r, t_2 = d_1r + d_5r + d_6r, t_3 = d_1f + d_5f + d_7f, t_4 = d_3f + d_6f \\ d_1r + d_5r < d_4f, d_1f + d_5f > d_3r$$

また、立上り/立下り遅延版において、立上り遅延が4~8、立下り遅延が3~7という変数制限条件のもとでシミュレーションを行うと、条件は得られず、正常に動作しないということがわかった。

それぞれのシミュレーションの4入力パターンに対する合計CPU時間は、標準遅延版が約0.4秒、立上り/立下り遅延版が約0.9秒であった。

4. 動的遅延を用いた時間記号論理シミュレーション

これまでの時間記号論理シミュレーションでは、各ゲートの遅延として一つの変数が割り当てられ、その変数は毎回与えられた範囲の同じ値をとるとする仮定のもとでシミュレーションを行っていた。しかし、動作時の遅延のばらつきまで

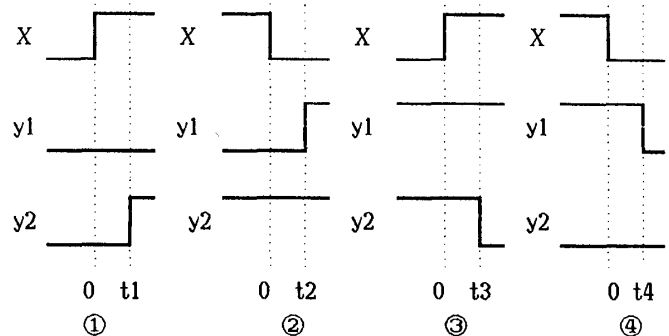


図4 入力パターン

も考慮した正確なタイミング検証をしたい場合には、この遅延モデルでは対応できない。そこで本章では、各ゲートの遅延時間が信号値の変化ごとに毎回変わるような遅延モデル（動的遅延モデルとよぶ）を用いた時間記号論理シミュレータについて考察する。

動的遅延モデルでは、ゲートに対して時間変数 d とそのとりうる値の範囲、及び動作時の遅延のばらつき Δ が与えられる。シミュレーション時のゲートの遅延は、実行時に与えられる出力のイベントごとに異なった変数 δ_i を用いて $d + \delta_i$ という形で表わされる。この動的遅延を用いた時間記号論理シミュレーションでは、出力イベント木の計算の際に、立上り/立下り遅延の場合と同様に、イベントの追い越しによるキャンセルが問題となる。しかも、立上り/立下り遅延では2つ以上のイベントを追い越すことはないが、動的遅延の場合には2つ以上のイベントを追い越す可能性がある。そのため、あるイベントがキャンセルされるかどうかは、そのあとに発生するイベントをすべて求めなければ決定できない。イベント木の計算時には、計算が行われているパスに関して、そこまで発生したイベントを、出力のイベント木にはつながらずにリストの形で保持しておき、新たに発生したイベントに対してキャンセルが起こるすべての場合について分岐して計算を進めて行く。その結果、得られた出力のイベント木では、パスの前半の部分がすべて条件節点であり、後半がそのパスで発生する全てのイベントがリストの形で現れる。

現在、動的遅延を用いた時間記号論理シミュレータはSun 3/60上でコーディング中である。

5. おわりに

時間記号論理シミュレータの立上り/立下り遅延モデル、動的遅延モデルへの適用について考察し、立上り/立下り遅延モデルについてはシミュレータ実現した。これにより、より現実に近いシミュレーション、より正確なタイミング検証が行えると考えられる。

<謝辞> 御討論頂いた矢島研究室の諸氏に感謝致します。

参考文献

- [1] 石浦菜岐佐, 矢島脩三: 時間記号シミュレーションについて, 情処研報, 87-DA-40, pp.39~46, (Dec. 1987).
- [2] 高橋瑞樹, 石浦菜岐佐, 矢島脩三: 時間記号論理シミュレータについて, 第36回情処全大, 3x-2, pp.1923~1924, (Mar. 1988).
- [3] 高橋瑞樹, 石浦菜岐佐, 矢島脩三: 時間記号論理シミュレーションの結果解析系, 情処研報, 88-DA-44, (Oct. 1988).