

4 J - 3

MOS 同期回路の設計検証

久木元 裕治 齋藤 勉 田中 英彦

東京大学工学部

1 はじめに

スイッチ・レベルの回路の中にはゲートレベルと直接対応の取れない特有の構造を持ったものがあり、特にダイナミックな回路には複雑な設計のものが多い。これらの回路については、現在かなり実用的になってきているゲートレベルの検証手法を直接適用することは難しい。本稿では、特に MOS 同期回路を対象として、トランジスタの接続情報と仕様の論理式を与えて、両者の間の検証を自動で行なう手法について述べる。

2 信号・トランジスタのモデル

ブリチャージなどを用いたダイナミック回路を表現するためには、チャージとして各端子に残っている信号とドライブされて各端子に現れる信号を区別する必要がある。そのため、ここでは信号を論理値と信号の強さによって表現する。信号の強さは、チャージの影響で残っている信号を“弱い信号”、ドライブされている信号を“強い信号”とする 2 値で表現し、全体としては強い 1、弱い 1、強い 0、弱い 0 の 4 値で信号をモデル化 [3] する。また、それぞれの信号値は表 1 に示したように、2 つの Bool 変数を用いて符号化して表現する。4 値の間には、信号の上書きを考慮した演算 signal-join を定義する。例えば、弱い信号が残っている端子に強い信号が現れると、端子の信号値はその強い値に書き換えられる。2 つの信号間の join 後の信号値は表 2 のテーブルのように定義される。

トランジスタはゲートが on したときにソースとドレインを接続する理想的なスイッチと考える。

信号 X	論理値 X_v	信号の強さ X_s	論理関数表現
strong1	1	1	$X_v \cdot X_s$
weak1	1	0	$X_v \cdot \overline{X_s}$
strong0	0	1	$\overline{X_v} \cdot X_s$
weak0	0	0	$\overline{X_v} \cdot \overline{X_s}$

表 1: 信号のモデル化

3 Characteristic Function による回路動作表現

ゲートレベルの論理回路では、出力を入力信号やフリップフロップの出力信号の論理関数として陽に表現することができる。

⁰Design Verification of MOS Synchronous Circuits, Yuji KUKIMOTO, Tsutomu SAITO and Hidehiko TANAKA Faculty of Engineering, The University of Tokyo

	strong1	weak1	strong0	weak0
strong1	strong1	strong1	error	strong1
weak1	strong1	weak1	strong0	error
strong0	error	strong0	strong0	strong0
weak0	strong1	error	strong0	weak0

表 2: signal-join 演算テーブル

ところが、スイッチ・レベルではトランジスタをゲートによって制御される理想的なスイッチと見なすため、信号の流れが両方向性をもち、出力信号を入力信号の関数として直接表現するのは困難である。そこで、ここでは各トランジスタがゲート、ソース、ドレインの 3 つのノードに対して制約を課しているの見なし、入力ノード、出力ノードのどのような値の組合せを、与えられたスイッチ・レベルの回路が許しているかでシステムの動作を表現する。具体的には、入力変数、出力変数の論理関数でシステムが許す値の組合せのときに限り論理値 1 を出力する関数でシステムを表現する。この論理関数のことを Characteristic Function [2] と呼ぶ。

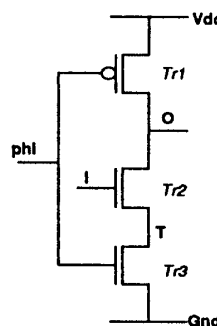


図 1: 簡単な回路例

例えば図 1 のような回路の場合には、トランジスタそれぞれについて次のような論理式が成立する。

$$\begin{aligned}
 Tr1 \quad & \bar{\phi} \rightarrow O_{s11} \\
 Tr2 \quad & I \rightarrow O_{s11} \cdot T_{s11} + O_{wk1} \cdot T_{wk1} \\
 & \quad + O_{s10} \cdot T_{s10} + O_{wk0} \cdot T_{wk0} \\
 Tr3 \quad & \phi \rightarrow T_{s10}
 \end{aligned}$$

信号 O, T を符合化して各式を書き直すと、

$$\begin{aligned}
 Tr1 \quad & \phi + \bar{\phi} \cdot O_v \cdot O_s \\
 Tr2 \quad & \bar{I} + I \cdot O_v \cdot O_s \cdot T_v \cdot T_s + I \cdot O_v \cdot \bar{O}_s \cdot T_v \cdot \bar{T}_s
 \end{aligned}$$

$$\text{Tr3} \quad \bar{\phi} + \phi \cdot \bar{T}_v \cdot T_s + I \cdot \bar{O}_v \cdot O_s \cdot \bar{T}_v \cdot T_s + I \cdot \bar{O}_v \cdot O_s \cdot T_v \cdot \bar{T}_s$$

Characteristic Function F はこの3つの論理式の積によって以下のように定義される。

$$F = \phi \cdot (\bar{I} \cdot \bar{T}_v \cdot T_s + I \cdot \bar{O}_v \cdot O_s \cdot \bar{T}_v \cdot T_s) + \bar{\phi} \cdot (\bar{I} \cdot O_v \cdot O_s + I \cdot O_v \cdot O_s \cdot T_v \cdot T_s)$$

4 検証手法

組み合わせ論理を実現した回路の場合は、出力を入力論理関数で表現したものを仕様として与える。1サイクル前の状態に依存する順序回路については、スイッチレベル特有の回路は制御回路ではなくレジスタ、シフタ等のデータパス回路であることを考えて、出力を1サイクル前の出力変数と入力変数を用いた論理式で表現した仕様を与える。

検証は、Characteristic Function を用いて回路をシンボリックに1クロック・サイクル動作させて、与えられた仕様との一致を調べるにより行なう。

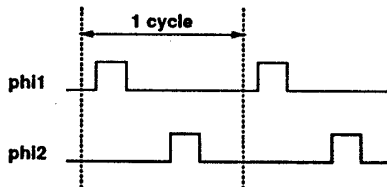


図2: 2 phase non-overlapping clock

例えば、図2の2相の non-overlapping-clock でコントロールされる論理回路については、以下の4式のようにして回路を動作させる。stableState はクロックの各フェーズでの安定状態を表現している。また、次のフェーズに現状態の影響をチャージとして信号を弱くして伝えるために、関数 weakened が用いられている。また * は変数の上書きを考慮した演算 signal-join である。stableState4 が1サイクル後の状態を示しているの、これを仕様で与えた論理関数と比較すれば良い。

$$\begin{aligned} \text{stableState1} &= F |_{\phi_1=1 \wedge \phi_2=0} \\ \text{stableState2} &= \text{weakened}(\text{stableState1}) * F |_{\phi_1=0 \wedge \phi_2=0} \\ \text{stableState3} &= \text{weakened}(\text{stableState2}) * F |_{\phi_1=0 \wedge \phi_2=1} \\ \text{stableState4} &= \text{weakened}(\text{stableState3}) * F |_{\phi_1=0 \wedge \phi_2=0} \end{aligned}$$

図1の例で考えると、この回路は $\phi = 0$ のときに O にプリチャージを行ない、 $\phi = 1$ で $I = 1$ のときに限り O を接地することにより、インバータのように動作する。上述の操作をこの回路に行なうと、次のようになる。

$$\begin{aligned} \phi = 0 \quad & F |_{\phi=0} = \bar{I} \cdot O_v \cdot O_s + I \cdot O_v \cdot O_s \cdot T_v \cdot T_s \\ & \downarrow \text{weakened} \\ & \bar{I} \cdot O_v \cdot \bar{O}_s + I \cdot O_v \cdot \bar{O}_s \cdot T_v \cdot \bar{T}_s(t) \\ \phi = 1 \quad & F |_{\phi=1} = \bar{I} \cdot \bar{T}_v \cdot T_s + I \cdot \bar{O}_v \cdot O_s \cdot \bar{T}_v \cdot T_s(t) \\ & \downarrow \text{join } \dagger \text{ and } \ddagger \\ & \bar{I} \cdot O_v \cdot \bar{O}_s \cdot \bar{T}_v \cdot T_s + I \cdot \bar{O}_v \cdot O_s \cdot \bar{T}_v \cdot T_s \end{aligned}$$

上の演算により、 I が0のときは出力 O はプリチャージされているため weak1 で、 I が1のときは出力が接地されて strong0 となること分かる。

検証は仕様の論理式 $O \equiv \bar{I}$ と設計側の動作後の論理式を比較することにより行なう。設計側の論理式には中間ノードの変数が含まれているので、これらの変数は existential quantifier で消去する。設計の論理式と仕様の論理式が一致していれば、設計は正しいことが分かる。一致しない場合には、2つの論理式の排他的論理和をとることにより、どのような入力の時に誤動作するかが分かる。

5 検証結果

検証システムはC言語で Sparc Station1(16MB) 上に実装されている。システム内での論理表現はすべて BDD [1] によって行なわれている。また4値の論理演算を行なうため、BDD の apply-routine を拡張している。表3に検証を行なった主な回路とその検証時間を示した。表にあげた回路はすべてダイナミック回路である。回路規模が大きくなると、回路中のノード数が増大し、設計を表現する Characteristic Function の BDD 表現がかなり大きくなるが、回路をモジュール化し各部分について検証するのが実際であることを考えると実用的な処理時間だと言える。

回路	Tr 数	検証時間
manchester-carry chain (Pass-transistor logic)	21	2.0sec
dynamic adder(NORA logic)	36	1.7sec
4-bit comparator(Domino logic)	56	22.3sec

表3: 検証を行なったスイッチ・レベル回路の例

6 おわりに

本稿では、Characteristic Function を用いたスイッチ・レベル回路の自動検証システムについて述べた。今後は、階層的な検証のサポートについて考えていきたい。

参考文献

- [1] Randal E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computer*, Vol. C-35, No. 8, pp. 677-691, August 1986.
- [2] O. Coudert, C. Berthet, and J. C. Madre. Verification of synchronous sequential machines based on symbolic execution. In J. Sifakis, editor, *Automatic Verification Methods for Finite State Systems*, pp. 365-373. Springer Verlag LNCS 407, June 1989.
- [3] Masahiro Fujita, Yusuke Matsunaga, and Taeko Kakuda. Automatic and semi-automatic verification of switch-level circuits with temporal logic and binary decision diagrams. In *Proceedings of IEEE International Conference on Computer-Aided Design*, pp. 38-41, November 1990.