

順序回路のタイミング・ベリファイア (SIMCHK)

萩原拓治 松下浩明 村井真一
(三菱電機 情報電子研究所)

あらまし

論理回路のタイミング検証は、設計された論理回路がレース、ハザード、スパイク等の影響を受けて誤動作するかどうかチェックすること、論理設計者にとって必ず実行しなければならないチェック項目の一つである。

本報告では、論理回路のレイアウト後の実装ディレーを考慮した論理シミュレーション結果から、対象回路にレース、ハザード、スパイク等のタイミング・エラーが潜んでいるかどうか、また、シミュレーション結果の入出力パターンを用いてデバイスのテストを行う場合、良品デバイスにもかかわらず、シミュレーションとテストの結果が不一致になるかどうかを自動的にチェックするプログラムSIMCHKの概要を報告する。

SIMCHKは現在、LSIの論理設計、テストのためのCADツールとして使用され、その有効性が示されている。

1. まえがき

論理回路のタイミング検証にはレース、ハザード、スパイク等の回路を誤動作させるエラーがあるかどうかチェックするタイミング検証と論理回路の動作速度をチェックするクリティカル・パス検証の2つの問題がある。

前者に対しては論理シミュレータ⁽¹⁾、安定/変化という簡易入力パターンを用いてタイミング検証を行うプログラム⁽²⁾、後者に対してはクリティカル・パス計算プログラム⁽³⁾⁽⁴⁾⁽⁵⁾などが開発されている。

タイミング検証に、論理シミュレータを用いる方法では、レース等のタイミング検証は充分には出来ない。スパイク、ハザード等のタイミング検証は出来るが、そのエラーが論理回路に何影響を与えるかどうかは膨大なリストを見て設計者が判断しなければならないという欠点がある。

安定/変化という簡易入力パターンを用いる方法は、簡易入力パターンを与えることにより低コストでタイミング検証が出来るがレース、ハザードの検証が不充分である。

クリティカル・パス計算プログラムを用いる方法は、外部入力フリップ・フロップ・フリップ・フロップ・外部出力というレベルについてすべてのパスのディレーが入力パターンなしで計算出来、クリティカル・パスは容易に計算出来る。しかし、本来このプログラムは、クリティカル・パスのディレーを計算して、論理回路の動作速度を最適化する目的で使用されるため、各パス間にレース、ハザード、スパイク等が発生するかどうかの判断は行なれない。

本報告では、タイミング検証を効率よく行うため、論理回路のレイアウト後のディレーをフィード・バックした論理シミュレーション結果から、回路を誤動作させるタイミング検証を自動的に行うプログラムSIMCHKについて述べる。

2. SIMCHKの目的と機能

図1に論理設計フローを示す。

通常、LSIを作る前には配置配線プログラム実行後のファンアウト、配線長をシミュレータにフィード・バックした実装デレイ・シミュレーション⁽⁶⁾が行われる。このシミュレーションで、論理回路は実際に作られるべきLSIとほぼ同じタイミングで動作する。

従って、この時LSIの動作タイミングを充分チェックしなければならない。

しかし、設計者にはそのチェックが充分出ず、タイミング・エラーを見逃してしまうことがよくある。

この見逃しをなくす目的で開発されたプログラムがSIMCHKである。

SIMCHKの基本的な処理は、フリップ・フロップ(FF)、ラッチ、ループ回路の入力に、回路を誤動作させるようなレース、ハザード、スパイク等が発生しないかチェックし警告を出すことである。

この時、図2に示すようにFF、ラッチ、ループ回路へ入力しない回路に対してレース、ハザード、スパイク等のチェックは必要ないので行わない。

設計者の望むタイミングで入力パターンを与え、対象回路にエラーが検出された場合製造すべきLSIは設計者の意図どおりの動作をしない可能性があるため、論理回路を修正しなければならない。

また、シミュレーション結果の入出力パターンを用いてLSIのテストを行う場合、テストの入力パターン印如のタイミングは、テストのピン間スキューにより、シミュレーションしたとおりのタイミングで入らず、良品デバイスにも分かかわらず不良品とみなしてしまうことが時々ある。

SIMCHKは、このシミュレーションとテストの結果の不一致をなくするという目的でも使用される。

SIMCHKでエラーが検出されたが、設計された回路に問題はなく、テストパタンの入れ方に問題がある時はテストパターンを修正しなければならない。

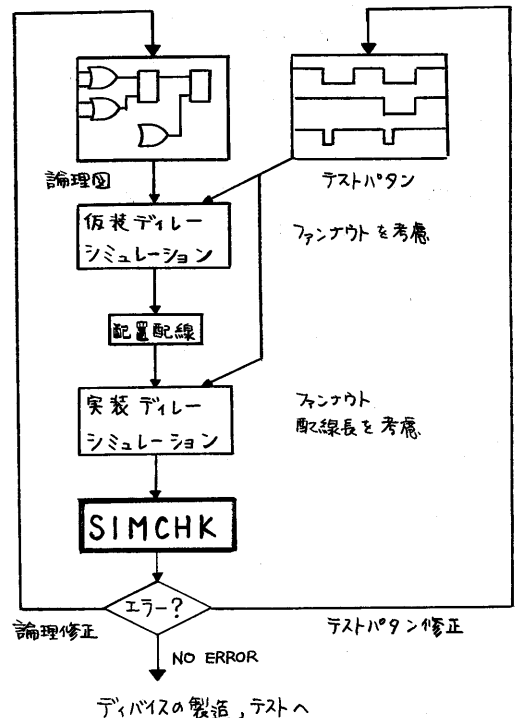


図1. 論理設計フロー

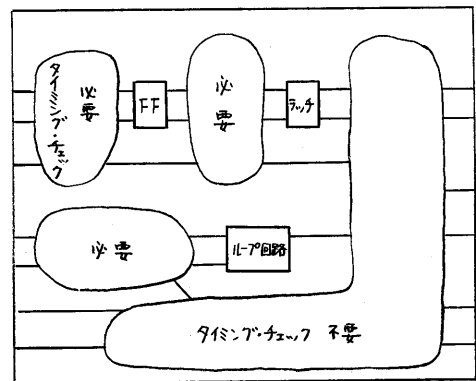


図2. タイミング・チェック対象

3. タイミング・チェック方式と項目

タイミング・チェックの対象とする素子タイプの宣言，シミュレーション周期，タイミング・チェック幅を指定することにより，立上り／立下り実装ディレクション・シミュレーション結果から自動的にタイミング・チェックを行う。

タイミング・チェック幅は，デバイスの性能，速度のバラツキ，テストの入カピン間スキューにより異なる。

以下のチェック項目がある。

- ・セットアップ／ホールド・タイム・チェック
- ・レース・チェック
- ・ハザード・チェック
- ・スパイク・チェック
- ・グローバル・レース・チェック（直列につながるFF間のレース）
- ・ループ回路のレース，ハザード，スパイク・チェック
- ・ストローブ時間の出力安定・チェック
- ・双方向性端子のシミュレーション・チェック
- ・シミュレーション入カパタンの周期性・チェック

4. タイミング・チェック素子タイプの宣言

FF，ラッチなどのタイミング・チェック対象素子タイプの宣言を行うことにより，任意の素子タイプに対して，汎用的なタイミング・チェックが行なわれる。

宣言文は図3のフォーマットで記述される。

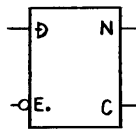
この宣言文により，各素子タイプに適合したレース，ハザード，スパイク，セットアップタイム，ホールドタイム等のチェックが行なわれる。

CLOCK，DATA，SET，RESET 端子の宣言文の一部は素子タイプによりない場合もある。

TYPE		タイミング・チェック素子タイプ
	SETUP	セットアップ・タイム
	HOLD	ホールド・タイム
	RACE	レース・チェック幅
	HAZARD	ハザード・チェック幅
	SPIKE	スパイク・チェック幅
CLOCK	7D _{FF} 端子名	NEGATIVE EDGE TRIGGER POSITIVE EDGE TRIGGER LOW ENABLE LEVEL TRIGGER HIGH ENABLE LEVEL TRIGGER LOW ENABLE MASTER SLAVE HIGH ENABLE MASTER SLAVE
DATA	データ端子名	
SET	セット端子名	LOW ACTIVE HIGH ACTIVE
RESET	リセット端子名	LOW ACTIVE HIGH ACTIVE

図3. タイミング・チェック対象素子タイプ宣言文

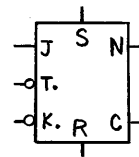
図4にLOW ENABLE LEVEL TRIGGER フラッチの回路図と宣言文，
図5にNEGATIVE EDGE TRIGGER JKFFの回路図と宣言文の例を示す。



TYPE D LATCH

SETUP	30
HOLD	20
RACE	30
HAZARD	30
SPIKE	30

CLOCK E. LOW ENABLE LEVEL TRIGGER
DATA D



TYPE JKFF

SETUP	20
HOLD	10
RACE	20
HAZARD	20
SPIKE	20

CLOCK T. HIGH ENABLE MASTER SLAVE
DATA J
DATA K.
SET S HIGH ACTIVE
RESET R HIGH ACTIVE

図4. D LATCHの宣言文の例

図5. JKFFの宣言文の例

5. フリップ・フロップ，ラッチのタイミング・チェック

タイミング・チェック対象素子タイプ宣言文により定義されたタイプに従ってFF，ラッチのレース，ハザード，スパイク，セットアップタイム，ホールドタイム等のタイミング・チェックを行う。

FF，ラッチのタイミング・チェックの説明をする前に用語の定義を行う。

δ (SETUP/HOLD/RACE/HAZARD/SPIKE) — 各チェック項目に応じて指定されるタイミング・チェック幅で，デバイスの性能，速度のバラツキ，テストの入力ピン間スキューなどを考慮して決定される。

T (CLOCK TRIGGER) — エッジ・トリガの場合，FFにトリガが入る時刻
NEGATIVE EDGE の時 $1 \rightarrow 0$ ， $1 \rightarrow X$ ， $X \rightarrow 0$ の場合がある
POSITIVE EDGE の時 $0 \rightarrow 1$ ， $0 \rightarrow X$ ， $X \rightarrow 1$ の場合がある
レベル・トリガの場合，ラッチがホールドされる時刻
LOW ENABLE の時 $0 \rightarrow 1$ ， $0 \rightarrow X$ ， $X \rightarrow 1$ の場合がある
HIGH ENABLE の時 $1 \rightarrow 0$ ， $1 \rightarrow X$ ， $X \rightarrow 0$ の場合がある
マスタ・スレーブの場合，データがマスタ・ラッチからスレーブ・ラッチに移る時刻

LOW ENABLE の時 $0 \rightarrow 1, 0 \rightarrow X, X \rightarrow 1$ の場合がある
HIGH ENABLE の時 $1 \rightarrow 0, 1 \rightarrow X, X \rightarrow 0$ の場合がある

T (DATA CHANGE) — FF のデータ端子の信号値が変化する時刻
 $1 \rightarrow 0, 1 \rightarrow X, 0 \rightarrow 1, 0 \rightarrow X, X \rightarrow 1, X \rightarrow 0$ の場合がある

T (SET/RESET RELEASE) — FF, ラッチのセット, リセットが解除される時刻
LOW ACTIVE の時 $0 \rightarrow 1, 0 \rightarrow X, X \rightarrow 1$ の場合がある
HIGH ACTIVE の時 $1 \rightarrow 0, 1 \rightarrow X, X \rightarrow 0$ の場合がある

T (INPUT DOWN/UP) — FF, ラッチのクロック, セット, リセット信号につながる同ーゲートの入力信号変化時刻
DOWN の時 $1 \rightarrow 0, 1 \rightarrow X, X \rightarrow 0$ の場合がある
UP の時 $0 \rightarrow 1, 0 \rightarrow X, X \rightarrow 1$ の場合がある

ただし, 信号ラインの X への変化, X からの変化をタイミング・チェックの処理対象にするかどうかはユーザーが指定することが出来る。

5.1 セットアップ/ホールド・タイム・チェック

FF のセット, リセットが効いてない時, FF のセットアップ/ホールドタイムが充分あるかどうかチェックする。

エッジ・トリガの場合, クロックの有効エッジ, レベル・トリガの場合, ラッチがホールドされる時, マスタ・スレーブの場合, マスタ・ラッチからスレーブ・ラッチへデータが移る時のセットアップ/ホールド・タイムをチェックする。

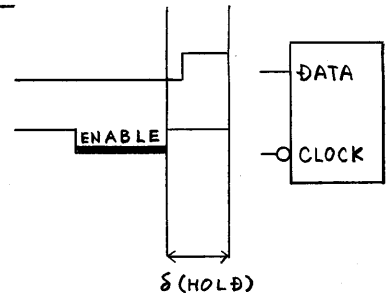


図 6

判定条件

WHILE SET/RESET IS NOT ACTIVE

$$|T(\text{DATA CHANGE}) - T(\text{CLOCK TRIGGER})| \leq \delta(\text{SETUP/HOLD})$$

図 6 の例でラッチがホールドされる時のホールド・タイムが $\delta(\text{HOLD})$ 以下なので警告が出る。

5.2 スパイク・チェック

FFのセット，リセットが効いてない時，FFのCLOCK，SET，RESET 端子にチェック幅以下のパルスが入るかどうかチェックする。

判定条件

WHILE SET/RESET IS NOT ACTIVE

PULSE WIDTH OF CLOCK, SET OR RESET $\leq \delta$ (SPIKE)

図7の例でセット，リセットが効いてない時CLOCK 端子に δ (SPIKE) 以下のパルスが入っているので警告が出る。

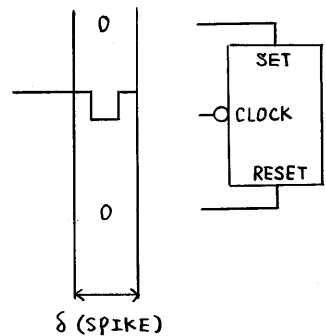


図 7

5.3 ハザード・チェック

FFのセット，リセットが効いてない時，FFのCLOCK，SET，RESET 端子につながるゲートにハザードが発生し，そのハザードがCLOCK，SET，RESET 端子まで伝搬するかどうかチェックする。

判定条件

WHILE SET/RESET IS NOT ACTIVE

$|T(\text{INPUT DOWN}) - T(\text{INPUT UP})| \leq \delta$ (HAZARD)

AND HAZARD IS PROPAGATED TO CLOCK, SET OR RESET

図8の例で信号B，Dいずれかが1であればハザードは伝搬しないので警告は出ない。

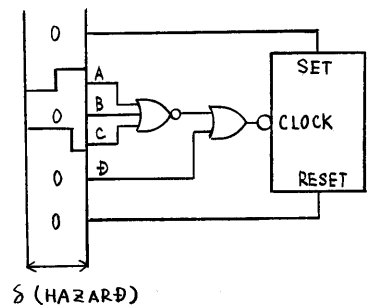


図 8

5.4 レース・チェック

FFのセット，リセットが効いてない時，FFのトリガとセット，リセット解除がチェック幅内で同時に起こるかどうかチェックする。

判定条件

WHILE SET/RESET IS NOT ACTIVE

$|T(\text{CLOCK TRIGGER}) - T(\text{SET RELEASE})| \leq \delta$ (RACE)

OR $|T(\text{CLOCK TRIGGER}) - T(\text{RESET RELEASE})| \leq \delta$ (RACE)

OR $|T(\text{SET RELEASE}) - T(\text{RESET RELEASE})| \leq \delta$ (RACE)

図9の例でクロックのトリガ時間が早い，セット解除時間が早いかでFFの出力の値が異なるので警告が出る。

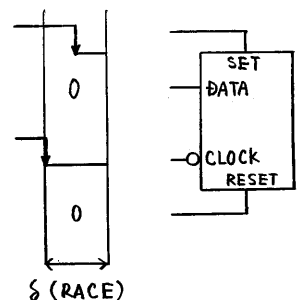


図 9

6. ループ回路のタイミング・チェック

汎用的なFF, ラッチ以外にゲートでフィード・バック・ループを形成している回路のタイミング・エラーは2とおりある。

1つは発振する可能性のあるもので, もう1つはラッチする可能性のあるものである。

ループが発振する可能性のあるか, ラッチする可能性のあるかは, ループ内ゲートの信号レベルの反転数によって決まる。

反転数が奇数の時, タイミング・エラーで回路は発振する可能性があり, 反転数が偶数の時, タイミング・エラーで回路はラッチする可能性がある。

図10に発振する可能性のある回路, 図11にラッチする可能性のある回路を示す。

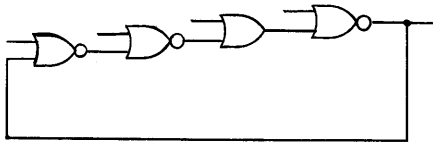


図10. 発振する可能性のある回路

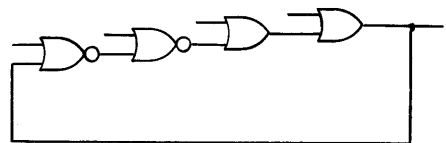


図11. ラッチする可能性のある回路

発振する可能性のある回路に対して発振チェックを行い, ラッチする可能性のある回路に対してレース, ハザード, スパイク・チェックを行う。

ループ回路のタイミング・チェックの説明をする前に, 図12を用いて, ループ回路の用語の定義を行う。

- | | |
|------------|---|
| ループ信号 | —— ループを形成している信号 (斜線部) |
| ループ内ゲート | —— ループ信号につながっているゲート (A, B, C) |
| ループ制御信号 | —— ループ内ゲートのループ信号以外の入力信号 |
| ループをフリーにする | —— ループ内のすべてのループ制御信号を0 or 1にする。
(OR系の場合0, AND系の場合1) |

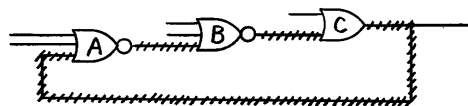
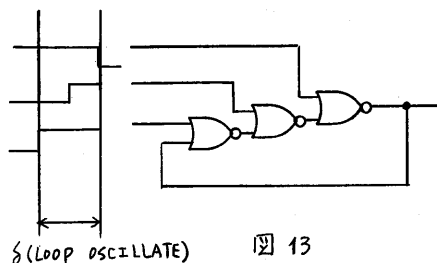


図12

6.1 ループ回路の発振・チェック

チェック幅内でループがフリーになるかどうかチェックする。



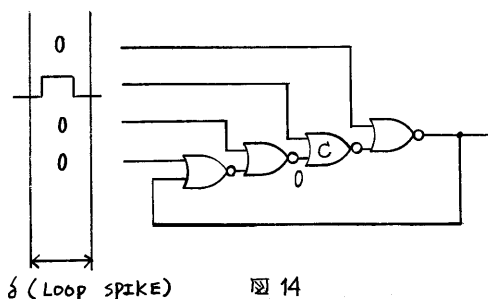
δ (Loop Oscillate)

図 13

6.2 ループ回路のスパイク・チェック

ループがフリーになっている時、ループ内素子のループ信号が 0 (OR系の場合), 1 (AND系の場合) のループ制御信号にチェック幅以下のパルスが入るかどうかチェックする。

図 14 の回路で, ゲート C の入力のスパイクはゼン間スキューなどの影響により発生しない可能性がある。スパイクをまじりた時は 1 をラッチし, まじらない時は 0 をラッチするため警告を出す。

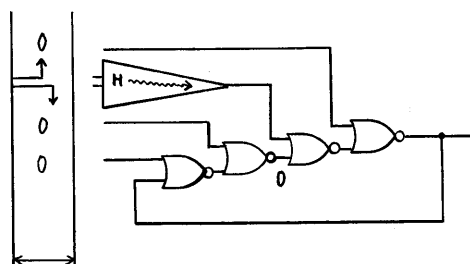


δ (Loop Spike)

図 14

6.3 ループ回路のハザード・チェック

スパイク・チェックと同じ状態で, スパイフの代わりにハザードが伝わるかどうかチェックする。



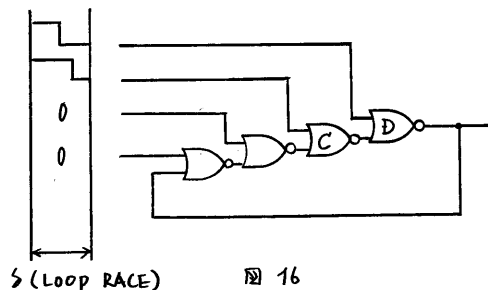
δ (Loop Hazard)

図 15

6.4 ループ回路のレース・チェック

2 つ以上のループ内ゲートのループ制御信号が 1 から 0 (OR系の場合), 0 から 1 (AND系の場合) にチェック幅内で同時に変化しループがフリーになるかどうかチェックする。

図 16 の回路で, ゲート C のループ制御信号変化が早い, D のループ制御信号変化が早いかでループにラッチされる値が異なるので警告を出す。



δ (Loop Race)

図 16

7. 検証例と実行結果

簡単な回路でのタイミング・チェック例を図17から図20で示す。
また、現在SIMCHKはマスタスライス・LSIのタイミング検証に使用されているが、図21にその実行結果を示す。
なお使用計算機はMELCOM COSMO 900IIである。

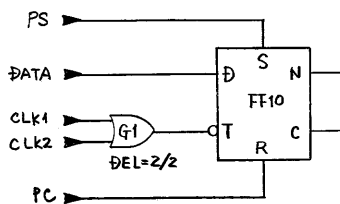


図17. 回路図

TYPE DFF
 SETUP 20
 HOLD 15
 RACE 20
 HAZARD 20
 SPIKE 20
 CLOCK T NEGATIVE EDGE TRIGGER
 DATA D
 SET S HIGH ACTIVE
 RESET R HIGH ACTIVE

図18. 宣言文

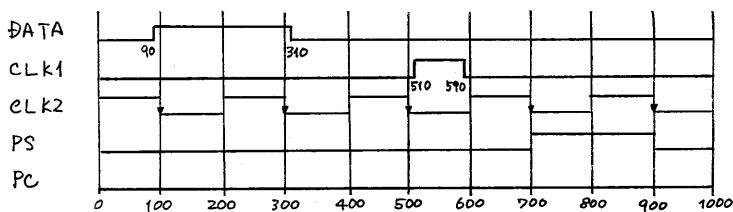


図19. 入力パターン

* SETUP * FF=FF10 PIN=D; SIG=DATA CHANGED 0-->1 AT TIME 90
 FF TRIGGER TIME 102

* HOLD * FF=FF10 PIN=D; SIG=DATA CHANGED 1-->0 AT TIME 310
 FF TRIGGER TIME 302

* SPIKE * FF=FF10 PIN=T; SIG=G1 CHANGED 1-->0-->1 WIDTH 10
 AT TIME 502

* HAZARD * FF=FF10 PIN=T
 LE=G1 ; SIG=CLK1 CHANGED 1-->0 AT TIME 590
 ; SIG=CLK2 CHANGED 0-->1 AT TIME 600

* RACE * FF=FF10 PIN=T; SIG=G1 CHANGED 1-->0 AT TIME 900
 PIN=S; SIG=PS CHANGED 1-->0 AT TIME 900

図20. 警告メッセージ

回路	ゲート数	FF数	バナン数	警告数						実行時間 (分)
				SETUP	HOLD	RACE	HAZARD	SPIKE	その他	
1	304	7	140	1	1					1.6
2	374	6	199			100			双方向性ピン・エラー	2.0
3	581	38	229							2.6
4	374	6	299			96				2.3
5	384	22	219				1	1	LOOP 回路の発振	2.5
6	804	42	242			8				1.4
7	740	37	375							2.0
8	673	31	160							2.5

図 21. 実行結果

8. おそび

実装ディレーをフィード・バックした論理シミュレーション結果から，フリップ・フロップ，ラッチ，ループ回路のレース，ハザード，スパイク，セットアップ／ホールド・タイム等のタイミング検証を自動的に行うことにより，論理回路のタイミング・エラー，テストパタンの入カエラーを検出出来ることを示した。現在，S E M C H K はマスタスライス・L S I のタイミング検証に使用されており，その有効性が示されている。

謝辞

本研究を遂行するにあたりご指導をいただいた三菱電機・情報電子研究所・情報処理開発部長・首藤勝 博士，C A D グループ各位，計算機製作所・計算機開発第一課・有賀幾夫氏に感謝いたします。

参考文献

- (1) S.A. Szygonda, "TEGAS 2 - Anatomy of General Purpose Test Generation and Simulation System for Digital Logic" Proceedings of the ACM IEEE Design Automation Workshop, 116~127, 1972
- (2) T.M. McWilliams, et al "Verification of Timing Constraints on Large Digital Systems" 17th DAC 139~147, 1980
- (3) R. Kamikawai, et al, "A Critical Path Delay Check System" 18th DAC 118~123, 1981
- (4) P. Ng, et al, "A Timing Verification System Based on Extracted MOS/VLSI Circuit Parameters" 18th DAC 288~292, 1981
- (5) 佐藤, 他 "LSI 遅延回路システム" 第22回 情報全国大会 931~932, 1981
- (6) Okura, et al, "A New Exact Delay Logic Simulation for ED MOS LSI" ICCD 80, 953~958, 1980