

## 準同期式におけるクロック配線駆動配置

高橋 渡, 高橋 篤司, 梶谷 洋司

東京工業大学 工学部 電気電子工学科

〒152 東京都目黒区大岡山 2-12-1

TEL : 03-5734-2665, FAX : 03-5734-2902

E-mail : {wata, atushi, kajitani}@ss.titech.ac.jp

あらまし

準同期式論理回路は、各レジスタのクロックタイミングを制御できれば周期は同期式の限界、レジスタ間の最大遅延、よりも小さくできる。しかし、指定されたようにクロックを分配するコストは大きい。この困難を避けるため「クロック配線駆動配置手法」を提案する。これは、クロック配線、それに従う配置という2段階最適化手法である。まず、回路とは独立に、クロック配線を実現する。その結果として決まる各地点毎のクロックタイミングを考慮して配置を行なう。クロック配線は、クロック遅延差を考慮する必要がないので、配線コストを大幅に削減できる。本方式の効果を確かめるために、クロック周期、回路の仮想配線長、クロック配線長に関して実験し、前2者を増やさずにクロック配線長を大幅に削減できることを確認した。

キーワード 準同期, クロック配線, 配置, クロック周期

## Clock Routing Driven Placement in Semi-Synchronous Circuits

Wataru Takahashi, Atsushi Takahashi and Yoji Kajitani

Dept. of Electrical and Electronic Engrg., Tokyo Inst. of Tech.

Ookayama, Meguro, Tokyo, 152 Japan

TEL : +81-3-5734-2665, FAX : +81-3-5734-2902

E-mail : {wata, atushi, kajitani}@ss.titech.ac.jp

Abstract

It is known that the clock-period of the semi-synchronous circuit can be shorter than the maximum delay between registers which is the limit of the conventional synchronous circuit. To bring out this potential ability, we propose a new design methodology, Clock-Routing Driven Placement, of clock distribution and circuit placement. It first constructs the clock tree free from the constraint on clock timing. Thus it can be designed for its own best, for example, minimizing the cost of clock tree and power consumption. Then a placement of a semi-synchronous circuit is sought for minimizing the clock-period. Experiments show that circuits obtained by the clock-routing driven placement realize the performance comparable to the conventional synchronous circuits in most cases, with considerable reduction of the total length of the clock-tree.

key words semi-synchronous, clock routing, placement, clock period

## 1 はじめ

VLSI 設計において回路の高速化のためにクロック周期を小さくすることは最重要課題のひとつである。同期式回路では、クロックが全てのレジスタに同時に入力されることを前提としている。その前提では、最小クロック周期はレジスタ間の最大信号遅延である。そのため、設計において配置・配線段階でレジスタ間の最大信号遅延を最小化する研究・実用化が盛んに行なわれているが技術的限界に達していると言われている。

この方向の努力は、クロックが全てのレジスタに同時に入力されることを前提としている。クロックが入力されるタイミングのバラツキ(クロックスキュー)はクロック周期の短縮を妨げる。従って最大遅延最小化に並行してゼロスキューを目指したクロック配線木の構成に関する研究が盛んである。しかし、クロック配線長の増加からやはり限界にきている。

更には、全レジスタを同時に動作させようとすることによる消費電力の変動があり、省消費電力回路の観点から同期式回路実現を目標とすること自体に疑問がある。

以上の3点のうち、第1、第3点については、同一周期のクロックを必ずしもレジスタに同時に入力することを前提としない準同期式回路によって解決しようという提案がある。

各々のレジスタにクロックを入力するタイミングが完全に制御できるという前提で準同期式回路を設計する試み [2] では、最小クロック周期は完全同期式回路と比べて平均で 16% 減少する実験結果を得ている。

このような潜在能力を示す準同期式回路の実現における最大の困難は第2点、すなわち指定されたタイミングで各レジスタにクロックを入力するクロック分配設計である。

準同期式回路では、クロック周期を最小化するにはそれぞれのレジスタに入力すべきタイミングがある範囲で指定される。これを最適クロックスケジュールと呼ぶ。その定式化と効率的な求め方は良く知られている [2, 4, 3]。

しかし、配置・配線が終了し全てのレジスタ間の最大、最小信号遅延が確定した状況で最適クロックスケジュールを計算しそれを実現するクロック配線木を構成することは、同期式における対応する問題、zero-skew 配線木実現、に比べてさらに設計が困難である。研究 [1] では、zero-skew クロック配線木の実現に比べて約 1.8 倍という大きな配線長を要する実験結果を与えている。そこで採用されたモデルでは、最適クロックスケジュールがレジ

スタ間の最大・最小信号遅延によってのみ決められレジスタの位置には関係がないことを反映させて、全レジスタにランダムにタイミングを指定してクロックスケジュールとしている。その結果、クロックソースに近いレジスタに大きなクロックタイミングが与えられたり、クロックソースから遠くのレジスタに小さなクロックタイミングが与えられたりして、クロック配線木に大きな迂回を強要することになって配線長の増大を招いている。このようなことが発生しないように回路配置に工夫すべきである。

そこで、本稿ではクロック配線駆動配置手法を提案する。本手法では、まず回路と独立にクロック配線を構成する。その結果として各地点毎にクロックタイミングが定まる。これをクロックマップという。続いて、このクロックマップの上にクロック周期の最小化(あるいは仮想配線長など付随する)目的のもとで回路の配置を行なう。

この方法の最大の利点は、クロック配線木をクロック遅延差を考慮しないで実現するので、配線木はそれだけの最適化が狙えることにある。

しかし、配線木を優先的に実現するとクロック周期に関して回路の良さがどの程度保証されるのか理論的に解析するのは極めて難しい。そこで実験で観測し、クロック配線木の配線長を平均 30% も削減してもクロック周期と仮想配線長を劣化させない結果を得た。これは本方式の将来性を示している。なお実験では、特定形式のクロック配線木を採用し、回路配置を平面をグリッドで区切ってモジュール割り当て問題とし、最適化はシミュレーテッドアニーリングで実行している。

本稿では、2章で準同期式における基本的な事項について述べ、3章では提案するクロック配線駆動配置手法について述べる。そして、4章では実験に用いる回路モデル、遅延モデル、実験の概要を示し、5章で結果を示す。そして、6章でまとめる。

## 2 準同期式回路

準同期式回路  $N$  は、レジスタとゲート、さらにそれらを接続する配線によって構成される。全てのレジスタは一つの単相クロックで動作し、クロックの立ち上がり値を取り込むものとする。

クロックはクロックソース  $s_0$  より、各レジスタのクロックピンにクロック配線によって伝えられる。この時、 $s_0$  でクロックが立ち上がってからレジスタ  $v$  のクロックピンでクロックが立ち上がるまでの時間差  $d_c(v)$  を「レ

ジスタ  $v$  のクロック遅延」と呼ぶ。

本稿では、このクロック遅延自体を扱うのではなく、相対的なクロックタイミングを扱う。基準とするレジスタ  $s$  を決め (仮想的でも良い)、そのレジスタ  $s$  でクロックが立ち上がってからレジスタ  $v$  でクロックが立ち上がるまでの相対的な時間差  $d(v) = d_c(v) - d_c(s)$  を「レジスタ  $v$  のクロックタイミング」と呼ぶ。

レジスタ  $u$  からレジスタ  $v$  に信号が伝搬する遅延時間を信号遅延と呼ぶ。信号遅延はゲートの立ち上がり時間と立ち下がり時間の違いや伝搬するパスの違いにより一意に定まらない。信号遅延の最大を  $s_{max}(u, v)$ 、最小を  $s_{min}(u, v)$  と表す。

信号が伝搬する全てのレジスタ対間  $u, v$  について以下の条件が満たされるとき回路は正常に動作すると仮定する。

レジスタ  $u$  からクロックの立ち上がり  $n$  で出力された信号がレジスタ  $v$  にクロックの立ち上がり  $n$  とその次のクロックの立ち上がり  $n+1$  の間に到達する。

言い換えると信号が伝搬する全てのレジスタ対間について以下の2つの制約式が満たされると回路が正常に動作すると仮定する。

**No-Double-Clocking Constraint**

$$d(v) \leq d(u) + s_{min}(u, v)$$

**No-Zero-Clocking Constraint**

$$d(u) + s_{max}(u, v) \leq t + d(v)$$

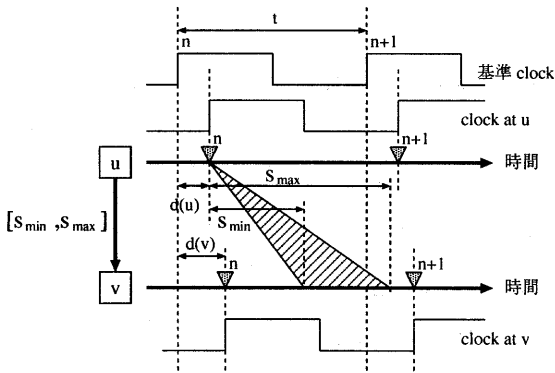


図 1: レジスタ対間の信号伝搬

この時、 $t$  はクロック周期を表す。以上の仮定にはレジスタの setup 時間や hold 時間が考慮されていないが信号遅延時間に含めて考えることで対応できる。

全てのレジスタのクロックタイミングとレジスタ対間の信号遅延が与えられた時に、上記の **No-Double-Clocking Constraint** を満たすか否かの判断と **No-Zero-Clocking Constraint** を満たす最小のクロック周期は簡単に計算できる。

まず、信号が伝搬する全てのレジスタ対間で、 $d(u) - d(v) + s_{min}(u, v) \geq 0$  が満たされるか計算する。一つでも満たさないレジスタ対間が存在すると **No-Double-Clocking Constraint** を満たさなくなるのでその回路は正常に動作しない。そのような回路を **infeasible** と呼ぶ。逆に信号が伝搬する全てのレジスタ対間で満たされた回路を **feasible** と呼ぶ。

**feasible** であった回路は次に最小クロック周期を求める。**No-Zero-Clocking Constraint** を満たすためには信号が伝搬する全てのレジスタ対間でクロック周期  $t$  は  $d(u) - d(v) + s_{max}(u, v)$  以上でなければならない。したがって、最小クロック周期  $t$  は信号が伝搬する全てのレジスタ対間のうちで最大の  $d(u) - d(v) + s_{max}(u, v)$  である。

### 3 クロック配線駆動配置手法

クロック配線駆動配置手法とは、(クロック配線段階) クロック配線があるコスト (配線長、消費電力など) が最適になるように仮想的に構成した上で、(回路配置段階) クロック周期や総信号配線長などを最適に配置する2段階最適化手法である。

以下にその手順を述べる。

第一に、配置を行なう前に仮想的に最適なクロック配線を構成する。この時のコストには電力、配線長、配線幅などが考えられるが、クロック遅延差は考慮しなくても良い。つまり、クロック遅延差は大きくても電力、配線長、配線幅などのコストが小さくなるクロック配線を配置前に仮想的に行なう。

仮想的なクロック配線の前ではレジスタを配置する場所によってクロック遅延が異なっている。レジスタの配置場所毎にクロック遅延を計算しクロックタイミングを求める。配置場所ごとにクロックタイミングを記録したマップであるクロックマップを作成する。つまり、クロックマップは配置場所を引数とするクロックタイミングの関数である。

第二に、作成したクロックマップを元に最小クロック周期が最小となる配置を求める。

まとめれば、クロック配線駆動配置手法とは、2章で

述べた2つの制約式から  $d(u) - d(v) + s_{min}(u, v) \leq 0$  という制約の元で、 $d(u) - d(v) + s_{max}(u, v)$  を目的関数とする配置手法である。

この最適化配置は極めて錯綜した状況にある。例えば以下のような事項を考慮しなければならない。

1. どんなに最大信号遅延  $s_{max}(u, v)$  が大きくても  $d(u) - d(v) + s_{max}(u, v)$  が小さければよい。すなわち、 $s_{max}(u, v)$  が大ききところでは  $d(u) - d(v)$  を負の値にすればよい。しかし、 $d(u), d(v)$  は場所によって決定されるので  $d(u), d(v)$  を変えることは  $s_{max}(u, v)$  も変わることを意味する。
2. レジスタ  $u$  からはレジスタ  $v$  にだけ信号が伝搬するわけではなく、他のレジスタ  $w$  にも信号が伝搬するかもしれない。さらに、他のレジスタ  $y$  からレジスタ  $u$  に信号が伝搬するかもしれない。それらも考慮してレジスタ  $u$  を配置しなければならない。
3. 回路によっては逆にレジスタ  $v$  からレジスタ  $u$  に信号が伝搬する場合がある。その場合は、 $d(v) - d(u) + s_{max}(v, u)$  を小さくしなければならないので単に  $d(u) - d(v)$  を小さくするように配置すると  $d(v) - d(u)$  が大きくなってしまふ。その場合は  $s_{max}(u, v)$  と  $s_{max}(v, u)$  の値を考慮して配置する必要がある。
4.  $d(u) - d(v)$  を小さくしようとしても  $d(u) - d(v) + s_{min}(u, v) > 0$  という制約を満たす必要があるので限度がある。

このように最小クロック周期を実現する配置を得ることは難しい問題である。そこで、本稿では、構成的な方法ではなく、確率的方法 Simulated Annealing を適用している。

## 4 実験モデル

クロック配線駆動配置手法の有効性を確かめるために、回路を単純にモデル化し実験を行なった。まずモデル化について述べる。

### 4.1 チップモデル

実験では、以下のような単純なチップモデルを用いる。

幅  $W$ 、高さ  $H$  のチップを考える。そのチップ上にレジスタを配置し、チップの周囲に入出力ピンを配置する。モデルの簡単化のためにゲートの位置は考慮しない。

チップの内部を図2のようにレジスタの数とほぼ同数のスロットに分割し、一つのスロットに一つのレジスタを割り当てる。同様にチップの外縁を入出力ピンを配置するスロットに分割し、一つのスロットに一つの入出力ピンを割り当てる。

信号遅延やクロック遅延を計算するために代表点を定義する。レジスタを割り当てるスロットでは、スロットの中心を代表点とし、入出力ピンを割り当てるスロットでは外縁の中心を代表点とする(図2)。

クロックソース  $s_0$  は、図2のようにチップの上部の外縁の中心に配置する。

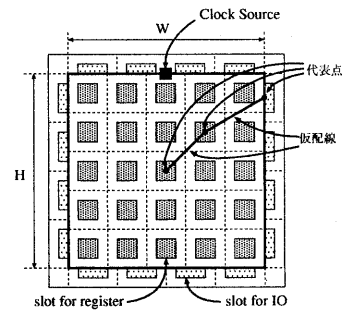


図2: チップモデル

### 4.2 クロック遅延モデル

クロックスケジュールはそれぞれのレジスタの配置場所によって以下のように計算される。

レジスタ  $r$  のクロック遅延  $d(r)$  を定義する。クロックソース  $s_0$  からあるレジスタ  $r$  までのクロック遅延  $d(r)$  はクロックソースの中心からレジスタ  $r$  の割り当てられているスロットの中心までのマンハッタン距離  $L_2(s_0, r)$  に比例すると仮定する。つまり、

$$d(r) = \alpha * L_2(s_0, r)$$

とする。この時の  $\alpha$  は定数である(図3)。

次に入出力ピン  $i$  のクロック遅延  $d(i)$  を定義する。外部との入出力である入出力ピンは外部に完全同期式回路があると仮定してその回路と同期をとるためにも、入出力ピンのクロック遅延  $d(i)$  をすべて一定とする。

実験ではレジスタのクロック遅延の最大値 ( $\alpha * (W/2 + H)$ ) と最小値 (0) の中間値として

$$d(i) = \alpha * (W/4 + H/2)$$

とした。つまり、入出力ピンとレジスタのクロックタイミングの差が最小となるようにする (図 3)。

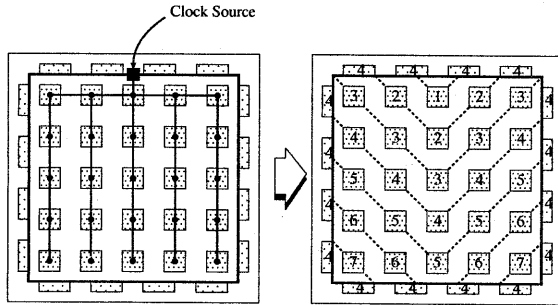


図 3: クロックスケジュールマップの例

### 4.3 信号遅延モデル

レジスタ  $u$  からレジスタ  $v$  への信号遅延 [ $s_{min}(u, v)$ ,  $s_{max}(u, v)$ ] は、ゲート遅延 [ $g_{min}(u, v)$ ,  $g_{max}(u, v)$ ] と配線遅延  $w(u, v)$  の和とする。

それぞれのゲートは固有の遅延を持つ。レジスタ  $u$  からレジスタ  $v$  までのあるパス上のゲートの遅延の和をレジスタ間のゲート遅延とし、全てのパスのうち最大のゲート遅延を  $g_{max}(u, v)$ 、最小のゲート遅延を  $g_{min}(u, v)$  とする。

レジスタ間の配線遅延  $w(u, v)$  は、レジスタ  $u$  とレジスタ  $v$  のマンハッタン距離  $L_2(u, v)$  に比例すると仮定する。つまり、パス上のゲートの位置を考慮しない。

したがって、レジスタ間の信号遅延 [ $s_{min}(u, v)$ ,  $s_{max}(u, v)$ ] は

$$s_{min}(u, v) = g_{min}(u, v) + \beta L_2(u, v)$$

$$s_{max}(u, v) = g_{max}(u, v) + \beta L_2(u, v)$$

となる、この時の  $\beta$  は定数である。

以上で、配置が生成されるとクロックタイミングと信号遅延が計算できるので、その配置における最小クロック周期が求まる。

### 4.4 配置アルゴリズム

実験ではクロック周期を最小とする配置を求めるアルゴリズムに Simulated-Annealing を採用した。

まず、ある任意の初期配置から feasible な配置を生成する。初期配置が infeasible なら、原因となるレジスタ対の数が増えないようなランダムなレジスタのロットの中身の交換または入出力ピンのロットの中身の交換を配置が feasible になるまで繰り返す。

次に、feasible な配置からランダムなレジスタのロットの中身の交換 または 入出力ピンのロットの中身の交換を行ない、新たな配置を生成する。もし、新たな配置が infeasible ならその配置は採用せず元の配置に戻し、さらに新たな配置を生成する。もし、新たな配置が feasible なら最小クロック周期を計算し、Simulated-Annealing のコスト関数に従いコストを計算しそのコストの増減を元にその配置を採用するか判定する。採用されれば、その配置に更新し新たな配置を生成し、採用されなかった時は、元の配置に戻し新たな配置を生成する。

Simulated-Annealing のコスト関数  $f$  は、最小クロック周期  $t$  と信号が伝搬する全てのレジスタ間の  $d(u) - d(v) + s_{max}(u, v)$  の平均  $z$  を用いて、 $f = a * t + b * z$  とする。この時の  $a, b$  は重みを調整するための定数である。

以上の様に、配置を生成し続けそのうちでコストが最小の配置を出力する。

## 5 実験結果

以上のようなモデルで、Simulated-Annealing を用いてベンチマークデータ LGSynth93 の回路に適用した。

完全同期式と準同期式の比較をするために、同じ条件でクロックマップだけを変えて配置を行なった。つまり、完全同期式回路は全てのレジスタに同時に到着することを前提としているので全ての場所のクロックタイミングが等しいクロックマップを用いる。

それぞれのデータではチップのサイズ  $W, H$ 、ロットの数はデータに含まれているゲート、レジスタ、入出力ピンの数を考慮して与えた。

パラメータ  $a, b, \alpha, \beta$  は、試行錯誤を繰り返し  $a = 10, b = 1, \alpha = 1.0, \beta = 1.0$  と決めて実験を行なった。

実験ではそれぞれのデータで 10 回試行を行なったうちのクロック周期  $t$  が最小の結果を選んで表 1 に示す。

表中の  $t$  はクロック周期を表し、length は仮配線長の総和を表している。括弧内はそれぞれ完全同期より準

同期にした時の増加率を表している。name は回路の名前を表し、#Regs はレジスタの個数を表し、括弧内の数 (#IO) は、入出力ピンの数を表している。

表 1: 準同期と完全同期との比較 ( $\alpha = \beta = 1.0$ )

name	#Regs (#IO)	完全同期		準同期	
		t	length	t (%)	length (%)
s27	3(5)	10.0	45	9.0 (-10)	49 (9)
s208	8(13)	19.3	450	19.6 (2)	462 (3)
s298	14(9)	17.0	443	16.0 (-6)	464 (5)
s344	15(20)	24.8	758	20.5 (-17)	845 (11)
s349	15(20)	24.8	734	20.5 (-17)	790 (8)
s382	21(9)	18.8	1086	19.0 (1)	1075 (-1)
s386	6(14)	23.9	1223	26.3 (10)	1208 (-1)
s400	21(9)	19.4	1066	19.5 (1)	1093 (2)
s420	16(21)	29.0	1222	29.2 (1)	1338 (9)
s444	21(9)	19.8	1086	19.8 (0)	1141 (5)
s510	6(26)	25.5	970	33.0 (29)	971 (0)
s526	21(9)	20.0	1030	18.2 (-9)	1076 (4)
s526n	21(9)	20.0	1018	18.2 (-9)	1062 (4)
s641	19(59)	75.5	5746	70.3 (-7)	6420 (12)
s713	19(58)	72.5	5949	68.0 (-6)	6471 (9)
s820	5(37)	27.8	2604	36.0 (29)	2526 (-3)
s832	5(37)	27.8	2650	36.0 (29)	2539 (-4)
s838	32(37)	35.5	2934	38.2 (8)	2985 (2)
s953	29(39)	27.6	3882	28.7 (4)	4251 (10)
s1196	18(28)	47.9	7145	47.9 (0)	7237 (1)
s1238	18(28)	46.0	6920	46.3 (1)	7200 (4)
s1423	74(22)	129.0	121152	125.3 (-3)	124637 (3)
s1488	6(27)	45.3	5167	56.3 (24)	5063 (-2)
s1494	6(27)	45.3	5163	56.3 (24)	5059 (-2)

ほとんどの回路でほぼ同じクロック周期と総仮配線長を実現している。しかし、いくつかの回路では準同期式の方が悪い結果となっている。実験の結果を詳細に解析していないので、その原因がどこにあるかははっきりしないが、仮定したクロックマップがその回路にとっては悪い影響を与えている可能性がある。つまり、想定するクロックマップを変えることによって改善されるかもしれない。そのような回路の解析は今後の課題である。

## 6 まとめ

準同期式回路におけるクロック配線駆動配置を提案した。この実験より、提案手法は完全同期式と比べてクロック配線を約30%減少させるのに、ほとんどの回路でクロック周期や配線長でほとんど変わらない性能を示した。

しかし、この実験だけでは提案手法の効果を得るのに、多くの点で十分ではないと思われる。実験した回路では、回路規模が小さいために提案手法の効果が十分でないかもしれない。

さらに、今回の実験では想定したクロック配線は固定して変更しなかった。得られた配置を基にしてクロック配線を修正し、全体性能の最適化をはかる方法は自然な発展であり、今後の課題である。

## 参考文献

- [1] A.Takahashi, K.Inoue, and Y.Kajitani. Clock-tree routing realizing a clock-schedule for semi-synchronous circuits. *Proc. ICCAD '97*, 1997.
- [2] A.Takahashi and Y.Kajitani. Performance and reliability driven clock scheduling of sequential logic circuits. *Proc. ASP-DAC '97*, pp. 37-42, 1997.
- [3] R. B. Deokar and S. S. Sapatnekar. A graph-theoretic approach to clock skew optimization. *Proc. ISCAS '94*, Vol. 1, pp. 407-410, 1994.
- [4] J. P. Fishburn. Clock skew optimization. *IEEE Trans. on Computers*, Vol. 39, No. 7, pp. 945-951, 1990.