T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

論題(和文)	レジスタの再配置による準同期式回路のクロック周期最小化手法
Title(English)	Clock Period Minimization Method of Semi-Synchronous Circuits by Register Relocation
著者(和文)	小平行秀, 高橋篤司
Authors(English)	Yukihide Kohira, Atsushi Takahashi
出典(和文)	第19回 回路とシステム軽井沢ワークショップ 論文集, Vol. , No. , pp. 259-264
Citation(English)	Proc. the 19th Workshop on Circuits and Systems in Karuizawa, Vol. , No. , pp. 259-264
発行日 / Pub. date	2006, 4
URL	http://search.ieice.org/
権利情報 / Copyright	本著作物の著作権は電子情報通信学会に帰属します。 Copyright (c) 2006 Institute of Electronics, Information and Communication Engineers.

Powered by T2R2 (Science Tokyo Research Repository)

第19回　回路とシステム
軽井沢ワークショップ
The 19th Workshop on Circuits and Systems
in Karuizawa, April 24-25, 2006

レジスタの再配置による準同期式回路のクロック周期最小化手法
Clock Period Minimization Method of Semi-Synchronous Circuits by Register Relocation

小平 行秀 †, 高橋 篤司 †
† 東京工業大学大学院 理工学研究科 集積システム専攻

Yukihide KOHIRA†, Atsushi TAKAHASHI†
† Dept. of Communications and Integrated Systems, Tokyo Institute of Technology

1　はじめに

従来の集積回路設計では，人間の理解しやすさ，設計の容易さを第一に，回路の一つの実現方法であるレジスタに同時にクロックを分配する完全同期方式を追求してきた．しかし，その前提の下での性能追求は限界に達しつつあるだけではなく，前提の維持のためのコスト (例えば，クロックバッファ数や消費電力) が非常に大きくなっている．一方，クロックを各レジスタに同時に分配することを前提としない準同期方式 [3, 2, 9] では，物理遅延を完全同期方式よりも効率良く利用できる可能性があるため，クロック周期，クロック面積，消費電力，信頼性などさまざまな指標で計られる性能を，低コストで実現することが期待されている．

準同期方式に関する初期の研究では，与えられた論理回路に対する，クロックスケジュールの設定手法 [3, 2, 9] やクロック回路の設計手法 [4] に重点が置かれていた．しかし，入力として与えられる論理回路は，完全同期方式を前提に合成されており，準同期方式の下で性能が改善されるとは限らない．例えば，クロックタイミングの変更のみでは，論理回路の遅延情報と構造により定められる同期回路のクロック周期の下界 (限界最小クロック周期) を実現できないことが知られている [10, 5]．近年，準同期方式に適した論理回路への修正手法として遅延を挿入する手法 [10, 5]，ゲートサイズの修正手法 [12]，マルチサイクルパスを用いた手法 [8] が提案されている．

また，完全同期方式でリタイミング [6] と呼ばれているレジスタの挿入位置の修正手法を準同期方式に応用した手法 [7, 11] が提案されている．準同期方式において，”リタイミング”はクロックタイミングの再設定と混同しやすいため，本稿ではレジスタの挿入位置の修正を”レジスタの再配置”と呼び，特にレジスタの再配置後に回路を完全同期方式で実現する手法を”リタイミング” と呼ぶ．

[6] では，リタイミングによるクロック周期の最小化やレジスタ数の最小化問題を線形計画法に定式化しているが，制約数が膨大となるので大規模な回路への適用ができない．また，各遅延素子の遅延値を任意の値に分割できるとき，リタイミングは限界最小クロック周期を実現することが知られている [10, 5] が，各遅延素子の遅延値を任意の値に分割できるという条件は現実的ではない．遅延値を分割できないときは，必ずしも限界最小クロック周期を達成できるとは限らない．

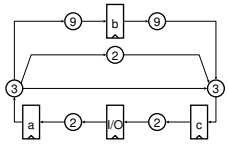


図 1: 回路 G.

[7] では，各レジスタに任意のクロックタイミングを設定可能であるという条件の下で，レジスタの再配置によるクロック周期の最小化や回路の信頼性の最大化問題を混合整数計画法に定式化した．また，高速化のために発見的手法を提案したが，計算量が膨大なため，前者で 100 ゲート規模，後者で 1000 ゲート規模の回路への適用が限界であった．[11] では，クロック周期を下げるために，大規模な回路にも適用可能な高速なレジスタの再配置手法が提案された．実験結果では全ての回路で限界最小クロック周期を達成していたが，必ずしも限界最小クロック周期を実現する手法ではなかった．

そこで本稿では [11] を基に，各レジスタに任意のクロックタイミングを設定可能であり，各遅延素子が一意の遅延値を有しているという条件の下で，入出力ピンでデータが入出力するタイミングを変更せず，遅延素子も分割することなく，限界最小クロック周期を実現する高速なレジスタの再配置手法を提案する．

2　準備

本稿では，回路をグラフ $G = (V_g, E_g)$ として表現する．V_g はグラフの点集合で，入出力ピン，レジスタ，ゲート，配線に対応する．E_g はグラフの有向枝の集合で，信号伝搬に対応する．本稿では，各遅延素子は一意の非負の遅延値を有していると仮定する．ある点 $v \in V_g$ の遅延値を $d(v)$ と表記する．また，枝 $e = (u, v) \in E_g(u, v = V_g)$ の重みを $d(e) = d(u)$ とおく．さらに，閉路 C の重み $d(C)$ を閉路上の枝重みの総和とする．閉路の重みが正，負，0 の閉路をそれぞれ正閉路，負閉路，0 閉路と呼ぶ．

回路のレジスタの集合を V_r とする．図 1 に回路の一例を示す．図 1 では，$\{a, b, c, I/O\}$ がレジスタの集合であり，レジスタ以外の点内の数字は遅延値を表す．レジスタの遅延値も点の重みとして表現できるが，本稿では説明を簡略にするため，レジスタの遅延値を 0 とする．

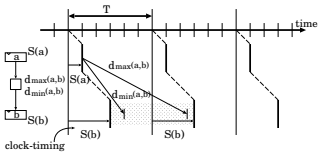


図 2: 同期回路のタイミング制約.

2.1 準同期式回路

同期回路が正常に動作するための条件は，信号が伝搬する全てのレジスタ対が以下の 2 式を満たすことであると知られている [3] (図 2).

Setup 条件: 0 クロック制約

$$S(a) - S(b) \leq T - d_{\max}(a,b)$$

Hold 条件: 二重クロック制約

$$S(b) - S(a) \leq d_{\min}(a,b)$$

ここで，T はクロック周期，$d_{\max}(a,b)$，$d_{\min}(a,b)$ はレジスタ a,b 間の最大遅延，最小遅延，$S(a)$，$S(b)$ はレジスタ a，b のクロックタイミングを表す.

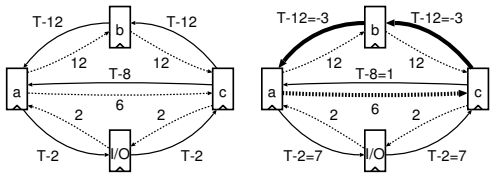
完全同期方式では，全てのレジスタに同じタイミングで同一周期のクロックが与えられることを前提とするため，クロック周期がレジスタ間の最大遅延未満の時，回路が正常に動作しないと考える．一方，準同期方式では，クロック周期がレジスタ間の最大遅延値未満で完全同期方式では正常に動作しないとみなされる回路でも，全ての制約を満たすのであれば回路は正常に動作すると正確に判断する.

本稿では，準同期方式において各レジスタに任意のクロックタイミングを設定できると仮定する．回路 G の準同期方式の最小クロック周期を $T_S(G)$ とする．準同期方式の最小クロック周期 $T_S(G)$ は，回路 G から得られる制約グラフ $H(V_r(G), E_r(G))$ によって定められる [9]．制約グラフ $H(V_r(G), E_r(G))$ の点集合 $V_r(G)$ はレジスタの集合 V_r である．制約グラフの有向枝集合 $E_r(G)$ は 2 つの制約式に対応する．レジスタ a から b への重み $d_{\min}(a,b)$ の有向枝は Hold 制約に対応し，D 枝と呼ぶ．レジスタ b から a への重み $T - d_{\max}(a,b)$ の有向枝は Setup 制約に対応し，Z 枝と呼ぶ．以後，制約グラフ $H(V_r(G), E_r(G))$ を簡単に $H(G)$ と表記する．また，制約グラフの Z 枝はクロック周期 T の関数である．クロック周期 $T = t$ とした制約グラフを $H(G,t)$ と表記する.

定理 1 ([9]) 準同期方式の最小クロック周期 $T_S(G)$ は，制約グラフ $H(G,t)$ が負閉路を持たない最小の t である.

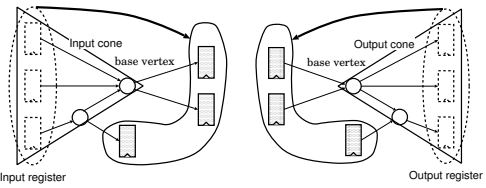
すなわち，準同期方式では，制約グラフ $H(G, T_S(G))$ において 0 閉路であり，制約グラフ $H(G,t)$ $(t < T_S(G))$ において負閉路である閉路が最小クロック周期を決めている．そこで本稿では，そのような閉路をクリティカルサイクルと呼ぶ．クリティカルサイクルの定義から，クリティカルサイクルの閉路の重みはクロック周期の関数である．また，本稿では各遅延素子の遅延値は非負と仮定していることから，D 枝のみからなるクリティカルサイクルは存在しない.

例として図 1 に示す回路 G について考える．レジスタ間の最大遅延が 12 なので，完全同期方式の最小クロック周期は 12 となる．回路 G の制約グラフ $H(G)$ を図 3(a) に示す．以後，本稿では制約グラフの Z 枝，D 枝をそれぞれ，実線，破線で表す．制約グラフ $H(G,9)$ を図 3(b) に示す．$H(G,9)$ に負閉路はなく，閉路 (a,c,b) のみ 0 閉路である．また，閉路 (a,c,b) は $H(G,t)$ $(t < 9)$ で負閉路となるので，回路 G の準同期方式の最小クロック周期 $T_S(G)$ は 9 となり，閉路 (a,c,b) は回路 G で唯一のクリティカルサイクルとなる．以後，本稿では制約グラフにおいてクリティカルサイクル上の枝を太線で表す.



(a) 制約グラフ $H(G)$.　(b) 制約グラフ $H(G,9)$.

図 3: 制約グラフ．閉路 (a,c,b) がクリティカルサイクル.



(a) 順行再配置.　(b) 逆行再配置.

図 4: レジスタの再配置.

2.2 レジスタの再配置

回路上のある点 x に対し，他のレジスタを経由せず信号が伝搬する点の集合を x の入力コーンと呼ぶ．また，x の入力コーンに含まれるレジスタを x の入力レジスタと呼ぶ．さらに，x の入力コーンの点から x の入力コーン以外の点への有向枝を x の入力コーンの出力枝と呼ぶ．ある点 x の入力レジスタを取り除き，x の入力コーンの全ての出力枝へレジスタを挿入するレジスタの再配置を x に対する順行再配置 (図 4(a)) と呼び，x を基点と呼ぶ．同様に，出力コーン，出力レジスタを定義した時，ある点 x の出力レジスタを取り除き，x の出力コーンの全ての入力枝へレジスタを挿入するレジスタの再配置を x に対する逆行再配置 (図 4(b)) と呼び，x を基点と呼ぶ.

本稿では，入出力ピンのクロックタイミングを変更しないために，入出力ピンを 1 つのレジスタに縮退し，再配置しない．以後，入出力ピンを縮退したレジスタを I/O レジスタと呼ぶ.

2.3 リタイミング

完全同期方式において，レジスタの再配置手法はリタイミングと呼ばれている [6]．レジスタの再配置をしても，回路中の任意の閉路が持つレジスタ数は変わらない．そのため，レジスタを回路中のどの枝へでも移動でき，かつ，回路中のどの点も任意の遅延値へ分割できる場合のレジスタ間最大遅延の下界を定義できる．このレジスタ間最大遅延の下界が完全同期方式に最小クロック周期の下界であるので，このクロック周期を限界最小

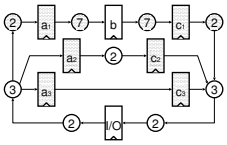


図 5: リタイミング後の回路 G'.

クロック周期と呼ぶ.

定義 1 ([10] 限界最小クロック周期 $T_B(G)$)

$$T_B(G) = \max_{C \in G \text{ 中の全ての閉路}} \frac{d(C)}{N(C)}$$

ここで，$N(C)$ は G 中の閉路 C 上にあるレジスタ数である.

例として，図 1 に示す回路 G をレジスタ間最大遅延が限界最小クロック周期となるようにリタイミングする．回路 G の限界最小クロック周期 $T_B(G)$ は最外周の閉路 $(I/O, a, b, c)$ により 7 と定まる．レジスタ間最大遅延が 7 となるようにリタイミングした回路 G' を図 5 に示す.

回路中のどの点も任意の遅延値へ分割できるのであれば，完全同期方式においてリタイミングにより限界最小クロック周期を達成できる．しかし，回路中のどの点も任意の遅延値へ分割できるという条件は現実的ではない．もし，どの点も遅延を分割できない場合は，必ずしもリタイミング後に限界最小クロック周期を達成できるとは限らない.

3 準同期方式を考慮したレジスタの再配置

前章で述べたように，クロックタイミングの変更により最小クロック周期を下げられる可能性がある．しかし，クロックタイミングの変更により最小クロック周期が限界最小クロック周期より小さくなることはなく，必ずしも限界最小クロック周期を達成できるとは限らないことが知られている [10, 5]．一方，リタイミングでは回路中のどの点も任意の遅延値へ分割できるのであれば限界最小クロック周期を達成できるが，遅延値を任意の値に分割できるという条件は現実的ではない．遅延値を分割できないときは，必ずしも限界最小クロック周期を達成できるとは限らない．そこで本稿では，I/O レジスタを再配置せず，遅延素子も分割することなしに，準同期方式を考慮してレジスタを再配置することで，限界最小クロック周期を達成する手法を提案する.

図 6 に回路 G_e の一部分と制約グラフ $H(G_e)$ の一部分を示す．また，図 7 に x に対する逆行再配置を G_e に適用し，レジスタ d が再配置された回路 G'_e の一部分と制約グラフ $H(G'_e)$ の一部分を示す．このとき，$H(G'_e)$ の閉路 C'_1 は，$H(G_e)$ の閉路 C_1 と比べ $(a, d_1), (d_1, c)$ の 2 つの Z 枝の値が増減しているが，閉路の重みは変化しない．同様に，$H(G'_e)$ の閉路 C'_2 の重みも $H(G_e)$ の閉路 C_2 から変化しない．したがって，閉路 C_1, C_2 が $H(G_e)$ においてクリティカルサイクルであるならば，閉路 C'_1, C'_2 も $H(G'_e)$ において クリティカルサイクルとなる．つまり，クリティカルサイクルの Z 枝が入り Z 枝が出るクリティカルサイクル上のレジスタ (*Z-Z* レジスタ) や，クリティカルサイクルの D 枝が入り D 枝が出るクリティカルサイクル上のレジスタ (*D-D* レジスタ) が再配置されても最小クロック周期は改善しない．一方，$H(G_e)$ の閉路 C_3 はレジスタの再配置により点 d で分割される．それにより，閉路 C_3 は閉路 (c, d) を取り込み，$H(G'_e)$ では新たな閉路 C'_3 が形成される．もし，$H(G_e)$ において閉路 C_3 がクリティカルサイクルで，閉路 (c, d) が正閉路の場合，$H(G'_e)$ の閉路 C'_3 の重みは閉路 C_3 の重みより大きくなり，クリティカルサイクルが解消される．つまり，クリティカルサイクルの Z 枝が入り D 枝が出るクリティカルサイクル上のレジスタ (*Z-D* レジスタ) が順行再配置 (図 4(a)) されるか，クリティカルサイクルの D 枝が入り Z 枝が出るクリティカルサイクル上のレジスタ (*D-Z* レジスタ) が逆行再配置 (図 4(b)) されることにより，クリティカルサイクルが他の閉路を取り込み，準同期方式の最小クロック周期が改善する可能性がある.

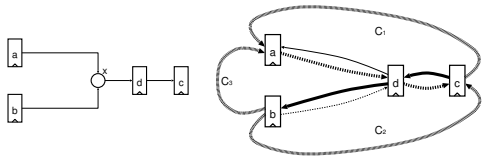


図 6: 再配置前の回路 G_e の一部分と制約グラフ $H(G_e)$ の一部分.

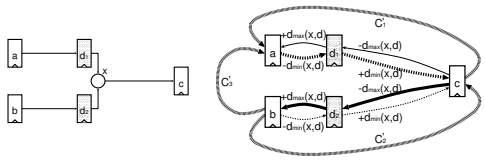


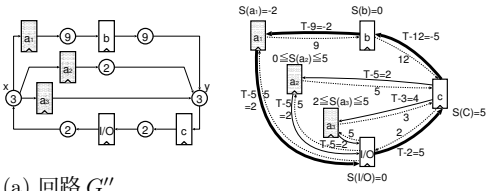
図 7: 再配置後の回路 G'_e の一部分と制約グラフ $H(G'_e)$ の一部分.

ここで，リタイミングと同様，準同期方式におけるレジスタの再配置を行っても，限界最小クロック周期は変わらないことに注意されたい．また，準同期方式の最小クロック周期は限界最小クロック周期より小さくなることない [10, 5] ことから，準同期方式におけるレジスタの再配置を行った際の準同期方式の最小クロック周期の下界は限界最小クロック周期である.

例として，図 1 に示す回路 G に対して，準同期方式を考慮したレジスタの再配置を適用する．前章で述べたように，回路 G の準同期方式の最小クロック周期は 9 であり，制約グラフ (図 3(b)) の閉路 (a, c, b) がクリティカルサイクルである．クリティカルサイクル (a, c, b) は，(a, c) が D 枝，$(c, b), (b, a)$ が Z 枝であるので，レジスタ a が *Z-D* レジスタ，レジスタ c が *D-Z* レジスタである．*Z-D* レジスタ a を再配置するため，G の x に対する順行再配置を適用した回路 G'' を図 8(a) に，制約グラフ $H(G'', 7)$ を図 8(b) に示す．$H(G'', 7)$ において負閉路がなく，閉路 $(c, b, a_1, I/O)$ は $T < 7$ のとき負閉路となるので，準同期方式の最小クロック周期が 7 となる．つまり，回路 G'' は限界最小クロック周期を達成している．同様に，*D-Z* レジスタ c が逆行再配置された回路も限界最小クロック周期 7 を達成する．以後，*D-Z* レジスタと *Z-D* レジスタを合わせて着目レジスタと呼ぶ.

着目レジスタに関して，以下の定理が成り立つ.

定理 2 各遅延素子が一意の値を有している場合，クリティカルサイクル上の着目レジスタに接続する D 枝の



(a) 回路 G''.　　(b) 制約グラフ $H(G'', 7)$.

図 8: x に対する順行再配置後の回路 G'' と制約グラフ $H(G'', 7)$.

最小遅延を決めるパスと Z 枝の最大遅延を決めるパスは異なる.

証明 クリティカルサイクル上にある着目レジスタに接続する D 枝のレジスタ間最小遅延を決めるパスと Z 枝のレジスタ間最大遅延を決めるパスが同じであると仮定する. このとき, 同じレジスタ対を結ぶパスであるので, このレジスタ対を (a, b) とおく. 各遅延素子は一意の値を有しているので, 仮定より $d_{\min}(a, b) = d_{\max}(a, b)$ であり, 閉路 (a, b) の重みは $d_{\min}(a, b) + (T - d_{\max}(a, b)) = T$ となる. これはクリティカルサイクル上にあるという仮定に反する. ∎

提案手法は, クリティカルサイクルが他の閉路を取り込むようにするために, クリティカルサイクル上にある着目レジスタに接続する D 枝のレジスタ間最小遅延を決めるパスと Z 枝のレジスタ間最大遅延を決めるパスが分岐する点を基点としてレジスタの再配置を行う. そこで, この分岐点を着目レジスタの基点と呼ぶ.

また, 制約グラフの中にクリティカルサイクルが複数存在する場合がある. この場合, クリティカルサイクルの枝によって, 強連結成分を定義できる. この強連結成分をクリティカル強連結成分と呼ぶ.

さらに, クリティカルサイクル上の着目レジスタの基点に対するレジスタの再配置を行う時に, クリティカル強連結成分から再配置されるレジスタとクリティカルサイクル上のレジスタを接続する有向枝を除去したグラフに再配置されるレジスタを端点とする 1 本以上の有向枝からなるパスが存在しないクリティカルサイクルを極大と呼ぶ. これは, 極大なクリティカルサイクルのある着目レジスタの基点に対するレジスタの再配置を行った場合, 取り込む閉路が正閉路であることを意味する.

3.1 提案手法

入力 : 回路 G

出力 : レジスタの再配置後の回路 G'

step1 : Z 枝のみからなる制約グラフから, 限界最小クロック周期 $T_B(G)$ を定める [5].

step2 : 制約グラフを構成し, 最小クロック周期 $T_S(G)$ を定める. $T_S(G) = T_B(G)$ ならば, G を G' として終了する.

step3 : 極大なクリティカルサイクル上の着目レジスタの基点に対するレジスタの再配置を行い, 得られた回路を G とし step2 へ. ただし, 再配置されるレジスタの中に I/O レジスタを含まない着目レジスタの基点に対してレジスタの再配置を行う. この条件を満たす基点が複数存在する場合は, 再配置後のレジスタ数が最も少ない基点を採用して, 再配置を行う.

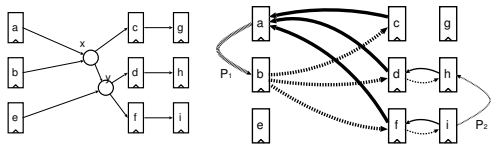


図 9: 再配置前の回路 G_p の一部分と制約グラフ $H(G_p)$ の一部分.

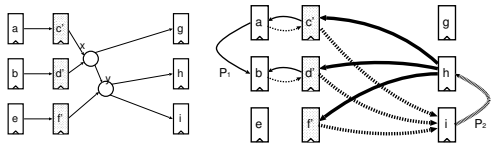


図 10: 再配置後の回路 G'_p の一部分と制約グラフ $H(G'_p)$ の一部分.

これより, 提案手法が限界最小クロック周期を達成することを証明する.

定理 3 提案手法の *step3* において, 条件を満たす基点 x に対してレジスタの再配置が行われた場合, レジスタの再配置前の制約グラフよりレジスタの再配置前の最小クロック周期におけるレジスタの再配置後の制約グラフの方が *0* 閉路の数が少ない.

証明 *D-Z* レジスタの場合について証明する. 提案手法において, 最小クロック周期が T_S で, あるクリティカルサイクル $C_1 = (P_1, c)$ の着目レジスタ c の基点 x に対してレジスタの再配置が行われたとする (図 9). このとき,

$$d(P_1) + d_{\min}(b, c) + T_S - d_{\max}(a, c) = 0$$

が成り立つ. 各遅延素子は一意の値を有しており, 提案手法の基点 x の決め方から, $d_{\min}(x, c) = d_{\max}(x, c)$ が成り立つ. したがって,

$$d(P_1) + d_{\min}(b, x) + T_S - d_{\max}(a, x) = 0 \qquad (1)$$

が成り立つ. さらに, 制約グラフ $H(G'_p)$ の閉路 (P_2, f') $(P_2 = (i, \ldots, h))$ がクロック周期 T_S で新たに負閉路となると仮定する (図 10). このとき, 式 (1) と同様,

$$d(P_2) + d_{\min}(y, i) + T_S - d_{\max}(y, h) = 0 \qquad (2)$$

が成り立つ.

ここで, $H(G_p)$ の閉路 $C = (P_1, f, P_2, d)$ に着目する. この閉路の重み $d(C)$ は,

$$\begin{aligned} d(C) &= d(P_1) + d_{\min}(b, f) + d_{\min}(f, i) + d(P_2) \\ &\quad + T_S - d_{\max}(d, h) + T_S - d_{\max}(a, d) \\ &= d(P_1) + d_{\min}(b, x) + T_S - d_{\max}(a, x) \\ &\quad + d(P_2) + d_{\min}(y, i) + T_S - d_{\max}(y, h) \\ &\quad + d_{\min}(x, y) - d_{\max}(x, y) \end{aligned}$$

となり, 式 (1), (2) と T_S が最小クロック周期であるという仮定から, 閉路 C は 0 閉路になり, クリティカルサイクルになる. したがって, C_1 は極大なクリティカルサイクルではなく, T_S の時に基点 x に対してレジスタの再配置が行われるという仮定に反する. よって, 新たな負閉路が生成されることはない.

また, 極大なクリティカルサイクルのある着目レジスタの基点に対してレジスタの再配置を行うので, 他の

閉路を取り込むことによって生成された閉路がクリティカルサイクルになることはない．したがって，クリティカルサイクルが解消されることが分かる．*Z-D* レジスタの場合についても同様に証明できる． ∎

定理 2 より，$T_S(G) > T_B(G)$ である時，条件を満たす基点が常に存在するならば，レジスタの再配置を繰り返すことにより，ある最小クロック周期 $T_S(G)$ での 0 閉路が無くなり，$T_S(G)$ が小さくなる．さらにレジスタの再配置を繰り返すことで，最終的には限界最小クロック周期で動作する回路を得ることができる．そこで，$T_S(G) > T_B(G)$ である時，条件を満たす基点 x が存在することを証明する．はじめに，クリティカルサイクル上に着目レジスタが存在することを述べる．

補題 1 ([10]) $T_S(G) > T_B(G)$ である時，クリティカルサイクルに少なくとも *1* つ，*D* 枝が含まれている．

上記の補題から，クリティカルサイクル上に少なくとも 1 つ，*Z-D* レジスタと *D-Z* レジスタが存在するので，以下の補題が成り立つ．

補題 2 $T_S(G) > T_B(G)$ である時，クリティカルサイクルに少なくとも *1* つ，着目レジスタが含まれる．

ここで，極大なクリティカルサイクルを定めるため，以下の補題を証明する．

補題 3 クリティカルサイクル $C_1 = (P_1, c)$ の着目レジスタ c の基点 x によってレジスタの再配置が行われる時，再配置される任意のレジスタを d とする．このとき，制約グラフ中の閉路 (P_1, d) はクリティカルサイクルである．

証明 *D-Z* レジスタの場合について証明する．最小クロック周期を T_S，$P_1 = (a, \ldots, b)$ とする (図 9)．このとき，式 (1) が成り立つ．

明らかに $d_{\min}(x, d) \leq d_{\max}(x, d)$ が成り立つので，

$$d(P_1) + d_{\min}(b, d) + T_S - d_{\max}(a, d) \leq 0$$

となる．最小クロック周期が T_S なので，制約グラフ中の閉路 (P_1, d) は 0 閉路となり，クリティカルサイクルであることが分かる．*Z-D* レジスタの場合についても同様に証明できる． ∎

定理 4 クリティカル強連結成分が与えられた時，極大なクリティカルサイクルが存在し，見つけられる．

証明 $C_1 = (P_1, c)$ が極大なクリティカルサイクルではないと仮定する．つまり，C_1 上の着目レジスタ c の基点 x に対するレジスタの再配置が行われる時，クリティカル強連結成分から C_1 上のレジスタと再配置されるレジスタを接続する有向枝を除去したグラフに再配置されるレジスタ d, f を端点とするパス $P_2 = (f, \ldots, d)$ が存在すると仮定する (図 9)．このとき，補題 3 の証明より閉路 (P_1, d)，(P_1, f) もクリティカルサイクルとなる．したがって，$C = (P_1, P_2)$ もクリティカルサイクルとなり，枝数の多いクリティカルサイクルを見つけられる．

この操作を繰り返しても極大なクリティカルサイクルを見つけられないとすると，無限に長いクリティカルサイクルが存在することになるが，クリティカル強連結成分は有限であるので，仮定は矛盾する．したがって，クリティカル強連結成分が与えられた時に，極大なクリティカルサイクルが存在し，上記の操作を繰り返すことで見つけられる． ∎

以上により，極大なクリティカルサイクルを定められる．そこで，この極大なクリティカルサイクル上に再配置されるレジスタに I/O を含まない基点が少なくとも 1 つ存在することを示す．

定理 5 提案手法において $T_S(G) > T_B(G)$ の時に，再配置されるレジスタに *I/O* を含まない基点が少なくとも *1* つ存在する．

証明 あるクリティカルサイクル $C_1 = (P_1, c)$ の着目レジスタ c の基点に対するレジスタの再配置を行う時，再配置されるレジスタに I/O レジスタを含む場合は，補題 3 より着目レジスタ c を I/O レジスタに置き換えた閉路 $(P_1, I/O)$ もクリティカルサイクルである．クリティカルサイクル上の全ての着目レジスタの基点に対して再配置できないならば，それらの着目レジスタを上記のように I/O レジスタに置き換えた別のクリティカルサイクルが存在することになる．このとき，そのクリティカルサイクル上で D 枝と Z 枝が接続する全てのレジスタが I/O レジスタとなり，I/O レジスタは 1 点に縮退しているので，Z 枝のみ，もしくは，D 枝のみからなるクリティカルサイクルが存在することになる．したがって，補題 2 に反するので，再配置されるレジスタに I/O を含まない基点が少なくとも 1 つ存在する． ∎

以上より，$T_S(G) > T_B(G)$ である時，条件を満たす基点が常に存在し，その基点に対してレジスタの再配置を繰り返すことにより，最終的には限界最小クロック周期で動作する回路を得ることができる．

4 実験結果

提案手法の効率を調べるため，線形計画法に定式化する完全同期方式におけるリタイミング手法 [6] と提案手法を ISCAS89 ベンチマーク回路に対して適用する実験を行った．遅延値は，not ゲートを 1，nand,nor ゲートを 2，and,or ゲートを 3 とした．両手法とも 3.40GHz/1GB Intel Pentium-4 CPU，1GB RAM の PC に gcc3.5.5 の C++で実装した．また，線形計画法には ILOG 社の CPLEX 9.0.0[1] を用いて解いた．

ISCAS89 ベンチマーク回路の 48 回路において，準同期方式の最小クロック周期と限界最小クロック周期が一致するものが 22 回路あった．この 22 回路に提案手法を適用しても，最小クロック周期が改善することはない．したがって，限界最小クロック周期が準同期方式の最小クロック周期よりも小さい 26 回路に対して提案手法を適用した．実験結果を表 1 に示す．

実験の結果，限界最小クロック周期をリタイミングでは達成できなかったが提案手法では達成できる回路が 12 回路存在した．その 12 回路の中 6 回路で提案手法によりレジスタ数も改善した．一方，リタイミング後と提案手法後の最小クロック周期が一致する 10 回路中 3 回路で，リタイミングより提案手法がレジスタ数が多くなってしまった．これは，リタイミングは回路全体に対してレジスタ数が少なくなるようにレジスタの再配置を行うのに対し，提案手法はレジスタ数を考慮せず，最小クロック周期が改善するように局所的にレジスタの再配置を行うからである．

計算時間は，修正回数が多くなってしまう s499，s635，s6669 では，提案手法の方が計算時間がかかってしまうが，ほとんどの回路で計算時間が短縮された．さらに，リタイミングでは解を求められなかった大規模な回路に対しても，実用的な計算時間で解を求められた．

表 1: 実験結果．original は変更前の回路，T_C は完全同期方式，T_S は準同期方式，retiming(T_R) は線形計画法を用いた完全同期方式のリタイミング [6] 適用後，proposed(T_B) は提案手法を適用後，のデータを表している．proposed の () 内はリタイミングに対する提案手法の割合を表している．なお，s15850.1 は線形計画法に定式化するのに 1 日以上かかり，s13207, s15850, s38417 は線形計画法を解くのにメモリオーバーになってしまうので．完全同期方式のリタイミング後の最小クロック周期を求められなかった．

	original				retiming			proposed					
model	#gate	T_C	T_S	#FF	T_R	#FF	time[s]	T_B	([%])	#FF	([%])	time[s]	([%])
$T_R > T_B$ (12 circuits)													
s344	160	37	34.0	15	20	27	0.12	19.00	(95.00)	26	(96.30)	0.06	(50.00)
s349	161	37	34.0	15	20	27	0.13	19.00	(95.00)	26	(96.30)	0.06	(46.15)
s382	158	18	12.0	21	12	29	0.14	11.25	(93.75)	25	(86.21)	0.03	(21.43)
s400	164	18	12.0	21	12	29	0.17	11.25	(93.75)	27	(93.10)	0.04	(23.53)
s444	181	20	13.0	21	13	29	0.33	11.67	(89.75)	35	(120.69)	0.09	(40.91)
s499	152	23	19.0	22	12	89	0.10	11.50	(95.83)	109	(122.47)	0.74	(740.00)
s635	286	162	158.0	32	89	63	1.57	88.50	(99.44)	76	(120.63)	2.10	(133.76)
s1269	569	70	61.0	37	40	123	5.88	39.34	(98.34)	90	(73.17)	3.10	(52.72)
s1512	780	54	43.0	57	41	72	25.72	40.50	(98.78)	61	(84.72)	0.11	(0.43)
s3271	1572	58	34.0	116	28	185	8.80	27.72	(98.98)	199	(107.57)	6.47	(73.52)
s3384	1685	168	154.0	183	76	183	111.10	75.50	(99.34)	292	(159.56)	0.04	(0.04)
s6669	3080	231	197.0	239	58	448	133.49	56.50	(97.41)	975	(217.63)	612.49	(458.83)
$T_R = T_B$ (10 circuits)													
s298	119	18	12.0	14	10	47	0.07	10.00	(100.00)	17	(36.17)	0.01	(14.29)
s526	193	18	12.0	21	11	63	0.26	11.00	(100.00)	22	(34.92)	0.01	(3.85)
s526n	194	18	12.0	21	11	63	0.25	11.00	(100.00)	22	(34.92)	0.01	(4.00)
s991	519	117	110.0	19	109	26	7.63	109.00	(100.00)	20	(76.92)	0.01	(0.13)
s1423	657	164	156.0	74	146	87	193.59	146.00	(100.00)	81	(93.10)	0.88	(0.45)
s3330	1789	66	40.0	133	32	123	7.34	32.00	(100.00)	147	(119.51)	1.00	(13.62)
s4863	2342	144	129.0	104	75	159	253.55	75.00	(100.00)	219	(137.74)	21.91	(8.64)
s9234	5597	107	72.0	228	63	263	445.65	63.00	(100.00)	240	(91.25)	2.37	(0.53)
s9234.1	5597	107	72.0	211	63	255	444.04	63.00	(100.00)	223	(87.45)	2.36	(0.53)
prolog	1601	68	40.0	136	31	144	6.04	31.00	(100.00)	154	(106.94)	0.97	(16.06)
T_R cannot be obtained. (4 circuits)													
s13207	7951	106	76.0	669	N.A.	N.A.	N.A.	75.00	(—)	670	(—)	1.43	(—)
s15850	9772	141	104.0	597	N.A.	N.A.	N.A.	78.00	(—)	643	(—)	41.46	(—)
s15850.1	9772	141	124.0	534	N.A.	N.A.	N.A.	103.00	(—)	544	(—)	16.59	(—)
s38417	22179	85	61.0	1636	N.A.	N.A.	N.A.	60.00	(—)	1638	(—)	9.74	(—)

5 まとめと今後の課題

本稿では，各レジスタが任意のクロックタイミングを設定可能であり，各遅延素子が一意の遅延値を有しているという条件の下で，遅延素子を分割することなく，限界最小クロック周期を実現する高速なレジスタの再配置手法を提案した．これにより，[6, 7] では求められなかった大規模な回路に対しても，実用的な計算時間で解を求められる．また，[11] で提案されている手法では，レジスタの再配置後に新たなクリティカルサイクルが構成されることがある．それにより，実験結果では全ての回路で限界最小クロック周期を達成していたが，必ずしも限界最小クロック周期を実現する手法ではなかった．これに対し，提案手法は理論的に限界最小クロック周期を達成することを示した．

しかし，現在の手法は最小クロック周期を改善するために局所的な修正を行っているので，レジスタ数がリタイミングより多くなってしまうことがある．そのため，最小クロック周期を悪化させないレジスタの削減手法を提案することが今後の課題である．

参考文献

[1] http://www.ilog.co.jp/.

[2] R. B. Deoker and S. S. Sapatneker. A Graph-Theoretic Approach to Clock Skew Optimization. In *ISCAS*, pp. 407–410, 1994.

[3] J.P. Fishburn. Clock Skew Optimization. *IEEE Trans. Comput.*, Vol. 39, No. 7, pp. 945–951, 1990.

[4] K. Inoue, W. Takahashi, A. Takahashi, and Y. Kajitani. Schedule-Clock-Tree Routing for Semi-Synchronous Circuits. *IEICE Trans. Fundamentals*, Vol. E82-A, No. 11, pp. 2431–2439, 2002.

[5] Y. Kohira and A. Takahashi. Clock Period Minimization Method of Semi-Synchronous Circuits by Delay Insertion. *IEICE Trans. Fundamentals*, Vol. E88-A, No. 4, pp. 892–898, 2005.

[6] C. E. Leiserson and J. B. Saxe. Retiming Synchronous Circuitry. *Algorithmica*, Vol. 6, No. 1, pp. 5–35, 1991.

[7] X. Liu and M. C. Papaefthymiou. Retiming and Clock Scheduling for Digital Circuit Optimization. *IEEE trans. CAD*, Vol. 21, No. 2, pp. 184–203, 2002.

[8] B. A. Rosdi and A. Takahashi. Low Area Pipelined Circuits by Multi-clock Cycle Path and Clock Scheduling. In *ASP-DAC 2006*, pp. 260–265, 2006.

[9] A. Takahashi and Y. Kajitani. Performance and Reliability Driven Clock Scheduling of Sequential Logic Circuits. In *ASP-DAC'97*, pp. 37–43, 1997.

[10] T. Yoda and A. Takahashi. Clock Period Minimization of Semi-Synchronous Circuits by Gate-Level Delay Insertion. *IEICE Trans. Fundamentals*, Vol. E82-A, No. 11, pp. 2383–2389, 1999.

[11] 上林英悟, 小平行秀, 高橋篤司. 準同期方式におけるリタイミングを用いた回路修正手法. 信学技報, VLD2004–146, Vol. 104, No. 709, pp. 55–60, 2005.

[12] 安井卓也, 黒川圭一, 豊永昌彦, 高橋篤司. クロックタイミング余裕度を考慮した遅延修正による回路最適化手法. DA シンポジュウム 2002, pp. 259–264, 2002.