論文 / 著書情報
Article / Book Information

論題(和文)	
Title(English)	Optimal Register Merging Method after Register Relocation in Semi-Synchronous Framework
著者(和文)	小平行秀, 高橋 篤司
Authors(English)	Yukihide Kohira, Atsushi Takahashi
出典(和文)	電子情報通信学会 2006総合大会講演論文集 (A-3-4), Vol. A, No. , p. 68
Citation(English)	Proc. the 2006 IEICE General Conference (A-3-4), Vol. A, No. , p. 68
発行日 / Pub. date	2006, 3
URL	http://www.ieice.org/jpn/books/t_g.html
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.

A-3-4

Optimal Register Merging Method after Register Relocation in Semi-Synchronous Framework

Yukihide KOHIRA† Atsushi TAKAHASHI†

† Tokyo Institute of Technology, Dept. of Communications and Integrated Systems

1 Introduction

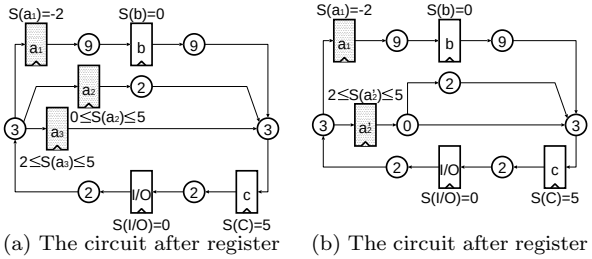
Optimal circuits in semi-cynchronous framework is different from those in complete-synchronous framework. So the optimization of circuit synthesis that takes semi-synchronous framework into account must be investigated.

As logic circuit modification methods that improve the clock period in semi-synchronous framework, register relocation methods [1, 2] are proposed. However, the mixed integer problem formulation and heuristic algorithms which are proposed in [1] cannot be applied to circuits with thousands gates, since the computation time of these algorithms is too long. In [2], a faster algorithm of register relocation is proposed, and register merging method is proposed in order to improve circuit area, power consumption, and etc. But the minimum feasible clock period is often increased, since registers are merged by an inefficient algorithm. Therefore, in this paper, we propose a minimization method of registers by merging registers without increasing the minimum clock period in semi-synchronous framework.

2 Register Merging Method

If registers that store the same signal data have a common feasible clock timing range in a clock schedule [3], the minimum clock period of the circuit obtained by merging these registers is not increased.

As an example, the circuit after register relocation is shown in Fig. 1 (a). The minimum clock period of this circuit is 7, the clock schedule when the clock period is 7 is also shown in Fig. 1 (a). The circuit after register a_2 and a_3 are merged is shown in Fig. 1 (b). The minimum clock period of the circuit after merging registers remains 7, since register a_2 and a_3 before merging registers have the common clock timing



(a) The circuit after register relocation.

(b) The circuit after register a_2 and a_3 are merged.

Fig. 1 An example of merging registers.

range $[2, 5]$. On the other hand, if register a_1 and a_2 are merged, the minimum clock period is increased to 8.

So, we formulate the minimization of the number of registers by merging registers as follows.

Problem: Minimization of the number of register subsets (MinR Prob)

Input: Set R of registers,
Timing range $[L(r), U(r)]$ $(\forall r \in R)$.

Output: Partition of R into subsets.

Object: Minimization of the number of register subsets.

Constraint: All registers in each subset have a common timing range.

MinR Prob can be solved in polynomial time, since this problem can be transformed to the partition into cliques problem for intersection graphs and the partition into cliques problem for intersection graphs can be solved in polynomial time [4]. The algorithm for MinR Prob is described as follows.

Algorithm: Minimization of the number of register subsets (MinR algorithm)

step0: $i = 1$.

step1: $R_i = \{\forall r_i | L(r_i) \leq \min_{\forall r \in R}(U(r))\}$
$R = R \setminus R_i$.

step2: If $R = \emptyset$, output $R_1, R_2, \ldots, R_i$ and terminate. Otherwise $i = i + 1$, and return to step1.

3 Experimental results

MinR algorithm is applied in PC with a 3.40GHz /1GB Intel Pentium-4 CPU and 1GB RAM in order to investigate an efficiency of this algorithm. The experimental results is shown in Table 1, where the timing range of each register is randomly set to two integers from 0 to 10 times the number of registers. The experimental results show MinR algorithm is very efficient.

Table 1 Experimental results.

#registers	1000	2500	5000	7500	10000
#subsets	38	56	83	101	117
time[s]	0.007	0.021	0.046	0.073	0.093

References

[1] 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.

[2] E. Kamibayashi, Y. Kohira, and A. Takahashi, "Circuit Modification Method of Semi-Synchronous Circuits with Retiming," Technical Report of IEICE, VLD2004–146, vol.104, no.709, pp.55–60, 2005. (In Japanese).

[3] T. Yoda and A. Takahashi, "Clock Schedule Design for Minimum Realization Cost," IEICE Trans. Fundamentals, vol.E83-A, no.12, pp.2552–2557, 2000.

[4] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, 1980.