# A High-Level Power Estimation Methodology for Low Power Design

**Chi-Ho Lin**

School of Computer Science, Semyung University
TEL : +82-43-649-1272, FAX : +82-43-649-1770
E-mail : ich410@semyung.ac.kr

**Abstract:** This work is a contribution to high-level synthesis for low power systems. In this paper, we present an efficient algorithm on performing estimation with an aim of reducing the power consumption in the synthesized data path.

In this paper, CDFG represents control flow, data dependency and such constraints as resource constraints and timing constraints. In the scheduling technique, the constraints are substituted by subgraphs, and then the number of subgraphs is minimized by using the inclusion and overlap relation efficiently. Also, The power estimation methods on enable power management and module selection are performed, so as to reduce the power consumption in low power design.

The effectiveness of the proposed algorithm has been proven by the experiment with the benchmark examples.

## 1. Introduction

Recently, several research approaches have been reported taking physical information into account. Most of the proposed algorithms use floorplanning information in high-level synthesis to estimate area and performance more accurately. Similarly, a lot of techniques have already been proposed taking into account power consumption in high-level synthesis [1-3].

High-level synthesis refers to the process of transforming a functional or behavioral specification of a design into a structured RTL implementation. A typical high-level synthesis process involves several subtasks including behavioral transformations, module selection, clock period selection, scheduling , and resource sharing, and RTL circuit generation.. High-level synthesis has a large impact on power consumption, which, if properly exploited, can lead to large power savings. Recent work has shown that the most savings in power consumption are often obtained at the higher levels of the design hierarchy[4-7].

Power estimation at this level of the design hierarchy is extremely important in order to (i) verify that power budgets are roughly met by the different parts of the design and the entire design, and (ii) evaluate the effect of various high-level optimizations, which have been shown to have a much more significant impact on power than lower-level optimizations. Architecture-level power estimation tools typically trade off some amount of accuracy for a drastic improvement in efficiency compared to low-level power estimation tools. The improved efficiency is due to the elimination of the need to obtain a gate- or transistor-level netlist, and the reduced complexity of analysis of RTL designs as compared to lower-level netlists. RTL design descriptions include various macroblocks like ALUs, vector logic operators, memories, register files, multiplexers, *etc.*, (which may be instantiated from a component library), as well as some amount of random or control logic, which may often be described functionally (*i.e.* without complete information about structure). This chapter describes the techniques that are used in architecture-level power estimation tools, including analytical power models, empirical activity and power macromodeling, sampling-based estimation, and models for control logic.[8-13]

In this paper, we concentrate on reducing power management in high-level synthesis comprises of the sequence of steps by means of which an algorithmic specification is translated into hardware. These steps involve breaking down the algorithm into primitive operations, and associating each operation with the time interval in which it will be executed (called operation scheduling).

In this paper, we present a new VHDL intermediate representation CDFG and an efficient synthesis technique for low power design. In the proposed algorithm, the constraints are substituted by subgraphs, and then the number of subgraphs is minimized by using the inclusion and overlap relation efficiently. Also, The power estimation methods are performed in a minimum bound and enable power management and module selection, starting with the as soon as possible as synthesis result, so as to reduce the power consumption in low power design. The rest of the paper is organized as follows. Section 2 describes the new synthesis algorithm for lower power. In section 3, we discuss a power estimation methods on enable power management and module selection. Section 4 describes experimental result in our proposed algorithm, and finally section 5 gives conclusion.

## 2. The new synthesis algorithm for low power design

The internal data structure, CDFG, which represents both the control flow and data flow effectively, is constructed. The CDFG represents the constraints which limit the hardware design such as conditional branch, sequential operation and time constraints.

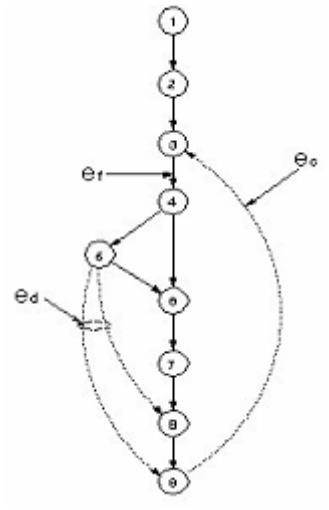


Fig 1. CDFG.

Fig. 1 shows a example for proposed CDFGin this paper. The semantics of VHDL description can be defined by the internal form generated by the VHDL.

Let D = (V, E) be a VHDL description, V the set of nodes for VHDL statements that represent VHDL statements in D and E directed edges that appear relations between nodes in D.

The internal representation CDFG of D is given by

$$CDFG = (G_f, G_c, G_t)$$

1) Sequential graph $G_f = (V, E_f)$ consists of V the set of nodes for VHDL statements and directed edges $E_f = \{ (v_1, v_2) \mid v_1, v_2 \subset V, v_1$ is the predecessor of $v_2 \}$.

2) Hardware constraint graph $G_c = (V, E_c)$ consists of V the set of nodes for VHDL statements and directed edges $E_c = \{ (v, w) \mid v, w \subset V$, there is one among 4 hardware constraints starting with v and ending with w $\}$.

3) Timing relation graph $G_t = (V, E_t)$ consists of V the set of nodes for VHDL statements and directed edges $E_t = \{ (v_1, v_2) \mid v_1, v_2 \subset V$, there is a timing constraint starting with $v_1$ and ending with $v_2 \}$.

In order to represent control flow, data dependency and such constraints as resource constraints and timing constraints effectively, the CDFG represents the constraints which limit the hardware design in such a way :

- ✓ no variable is assigned more than once in each control step
- ✓ no I/O port is accessed more than once in each control step
- ✓ the total delay of operations in each control step is not greater than the given control step-length
- ✓ all designer imposed constraints for scheduling particular operations in different control steps are satisfied.

In order to satisfy any of the above conditions, the proposed scheduling algorithm generates constraints between two nodes that must be scheduled into different control steps.

### 2.1 The minimization of subgraph with the inclusion and overlap relation

The hardware constraints are substituted by subgraphs, and the number of the constraints is minimized by using the inclusion and overlap relation among subgraphs. The subgraph minimization for the hardware constraints optimizes the number of control steps needed to execute the nodes in the CDFG.

(Definition 1) Regardless of conditional branches, if both edges $E_c$ is in inclusion relation, an included edge can be removed .

The subgraph minimization for the hardware constraints optimizes the number of control steps needed to execute the nodes in the CDFG. In an example of Fig 2., if edge(2,9) include both edge(3,4) and edge(2,9), An edge(2,9) is removed. Just, because the nodes 3 and 4 cannot operate in the same control step, constraints indicated by the nodes 2 and 9 can be removed.

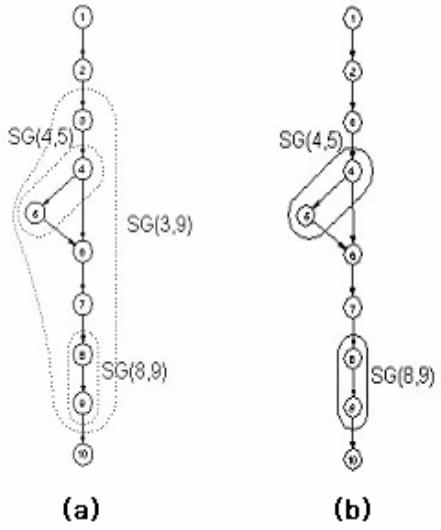


Fig 2. The removal of subgraph with inclusion relation
(a) A subgraph with inclusion relation
(b) The removal subgraph of Fig (a).

After the removal of subgraph with inclusion relation, we have to search for subgraphs with overlap relation to minimize total operation time in replacing many subgraphs with new subgraph. Accordingly to be scheduled as soon as possible and to minimize total operation time for the nodes in the CDFG, the overlap subgraphs must be minimized with the following priority.

As shown in Fig 3. and Fig 4., if there is an overlap relation between edges $E_c$ in subgraph, Overlap part is replaced by new edge $E_c$ ,and then old new edge $E_c$ is

removed repeatedly until not exist overlap relation in the CDFG.

## 3. The High-Level Power Estimation Methodology

### 3.1 Power Estimation on enable power management

For power management, Consider the computation of the expression |a - b|. The CDFG for this computation shows the |a - b| examples of the low power enable management which was not unnecessary the calculation in circuit.

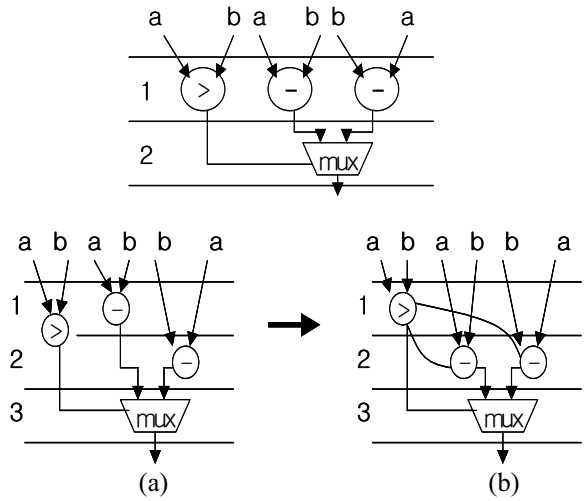


Fig. 3 |a - b| Low power enable management result
(a) Not considering low power management
(b) Considering low power management

In Fig 3(b), the (>) operation is assigned to the first control step, and depending on its result, only a - b or b - a is performed. In the RTL implementation, if the two (-) operations are assigned to two different subtracters, only the inputs to one of the subtracters is allowed to change, depending on the result of the conditional. If the two (-) operations are assigned to the same subtracter, the subtracter's inputs do not change in the first clock cycle, and are set appropriately in the second clock cycle depending on how the conditional evaluates. As the |a - b| low power enable management result shown in Fig 3, CDFG in this paper adopted the estimation result.

### 3.2 Power Estimation on module selection

The goal of the synthesis algorithm for low power is to increase the potential for a functional unit(FU) to reuse an operand. Henceforth, we will call operand reutilization the fact that an operand is reused by two operations consecutively executed in the same FU.

In Fig 4, where the estimation result for lower power on module selection are shown, There are some operations in the scheduled CDFG whose result is the input for more than one operation. For example, the result of addition 1 is input for additions 2 and 4. Assume that additions 2 and 4 are assigned to the same adder A. Assume also that between the execution of addition 2 and 4 there is no other use of adder A. Then, one of the operands of adder A will not change from addition 2 to addition 4. Fig 4(a) shows a schedule and an FU binding with two adders obtained with a traditional synthesis algorithm for the scheduling task and clique-partitioning approach with weights to minimize the number of interconnection units for the FU binding task. None of the two operands are achieved. Fig 4(b) show the schedule and FU binding obtained with the estimation algorithm for low power for the scheduling task and a slightly different approach to the clique-partitioning for the FU binding task. Now both operands are achieved. Module selection in power estimation also trades off latency for operands. This is also illustrated in Fig 4. If addition 5 happens to be in the critical path, the schedule and FU binding in Fig 4(a ) has one more cycle of latency than the one in Fig 4(b).

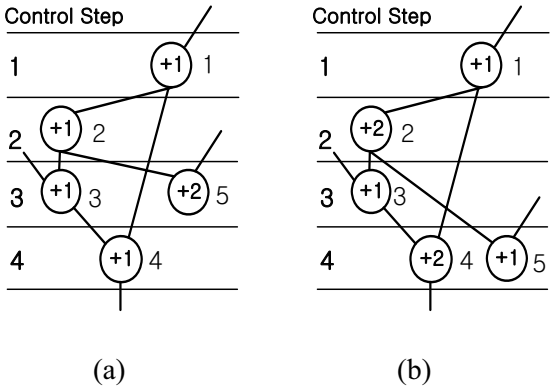


Fig. 4. The estimation result for low power.
(a) A traditional synthesis result.
(b) A estimation result for lower power.

## 4. Experimental Results

In this paper, our high-level power estimation methodology for low power design algorithm have been implemented in C++ and UltraSPARC III system. Also, it have been tested on the Fig 5. In this experiment results, we results the calculated power reduction ratio to adopt the HLS benchmark through the result of an optimal data path estimation technique for low power circuit design.

Our methods were implemented within the framework of low power high-level synthesis. The effectiveness of the proposed algorithm has been proven by the experimental result for benchmarks. Especially Cordic system power consumption is reduced about 30%.

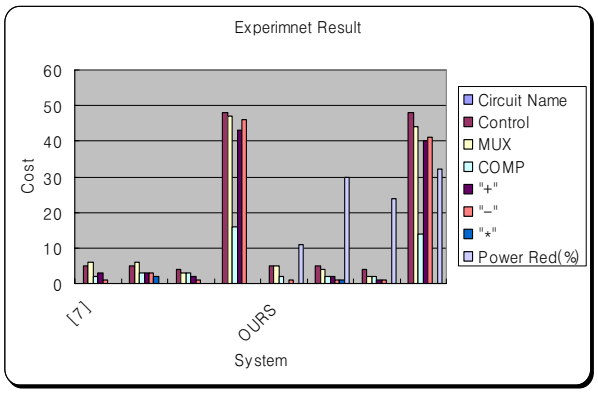

Fig 5. The low power estimation result for benchmarks.

## 5. Conclusion

We have addressed the problem of minimizing power consumption in behavioral synthesis of data-dominated circuits. We have presented a synthesis algorithm which, for a given throughput, exploits the slack available to operations to obtain a schedule that power management technique and module selection

The high-level synthesis technique for minimizing the operation time and handling the conditional branch effectively for ASIC design have been performed. Unlike most previous work, we also consider power estimation on enable power management and module selection, so as to reduce the power consumption in low power design. We have implemented the algorithm, and presented experimental results to demonstrate its effectiveness.

Also, we have obtained a solution that maximizes the ability to do power management while still meeting user specified throughput and hardware resource constraints.

### Reference

[1] L. Scheffer, “A roadmap of CAD tool changes for sub-micron interconnect problems”, in *International Symosium on Physical Design*, 1997.

[2] J.-P. Weng, and A. C. Parker, “3D scheduling: High-level synthesis with”, in *Proc. of Design Automation Conference*, 1992.

[3] Y.-M. Fang, and D. F. Wong, “Simultaneous functional-unit binding and floorplanning”, in *Proc. Int. Conf. Computer-Aided Design*, 1994.

[4] J. M. Chang, and M. Pedram, “Register allocation and binding for low power,” in *Proc. of Design Automation Conference*, 1995.

[5] R. Mehra, L. M. Guerra, and J. M. Rabaey, “Low-power architectural synthesis and the impact of exploiting locality”, in *J. VLSI Signal Processing*, 1996.

[6] A. Raghunathan, and N. K. Jha, “SCALP: An iterativeimprovement-based low power data path synthesis system”, in *Proc. Int. Conf. Computer-Aided Design*, 1997.

[7] L. Kruse, E. Schmidt, G. v. C¨olln, A. Stammermann, A. Schulz, E. Macii, and W. Nebel, “Estimation of lower and upper bounds on the power consumption from scheduled data flow graphs”, in *IEEE Trans. VLSI Systems*, 2001.

[8] K. Chao, and D. F. Wong “Floorplanning for Low Power Designs”, in *IEEE Trans. VLSI Systems*, 1995.

[9] P. Prabhakaran, and P. Banerjee: “Simultaneous scheduling, binding and floorplanning in high-level synthesis”, in *Proc. Int. Conf. VLSI Design*, 1998.

[10] L. Zhong, and N.K. Jha “Interconnect-aware High-level Synthesis for Low Power”, in *ICCAD*, 2002.

[11] P. Christie, and D. Stroobandt “The Interpretation and Application of Rent’s Rule”, in *IEEE Trans. VLSI Systems*, 2000.

[12] G. Jochens, L. Kruse, E. Schmidt, and W. Nebel “A New Paramiterizable Power Macro-Model for Datapath Components”, in *Proc. Of Design, Automation and Test in Europe*, 1999.

[13] W. Nebel, D. Helms, E. Schmidt, M. Schulte and A. Stammermann “Low Power Design for SoCs”, in *Information Society in Broadband Europe*, Romania, 2002.