Minimizing Energy Consumption of VLSI Processors Based on Dual-Supply-Voltage Assignment and Interconnection Simplification

Masanori Hariyama, Shigeo Yamadera, Michitaka Kameyama
Graduate School of Information Sciences
Tohoku University
Aoba 6-6-05, Aramaki, Aoba, Sendai, Miyagi, 980-8579, Japan
Email: {hariyama@,yamadera@kameyama.,kameyama@}ecei.tohoku.ac.jp

***Abstract*— This paper presents a design method to minimize energy of both functional units (FUs) and an interconnection network between FUs. To reduce complexity of the interconnection network, data transfers between FUs are classified according to FU types of operations in a data flow graph. The basic idea behind reducing the complexity of the interconnection network is that the interconnection resource can be shared among data transfers with the same FU type of a source node and the same FU type of a destination node. Moreover, an efficient method based on a genetic algorithm is presented for large-size problems.**

I. Introduction

In recent years, low power has become a primary design concern. An effective way to reduce dynamic power consumption is to lower the supply voltage of a circuit. Especially, the power consumption in interconnection network increases in the deep-submicron process. Therefore, it is important to consider power consumptions of functional units and interconnection network simultaneously. To reduce the power consumption of functional units, the use of multiple supply voltages is a well-known technique that reduces dynamic power consumption without increasing the circuit delay
citecvs. In the technique, a lower supply voltage is applied to operations on non-critical paths, and a higher supply voltage is applied to operations on critical paths. The major concern of this technique is that the number of functional units, that is, the chip area increases due to the delay of operations to which lower supply voltages are applied. This paper presents an efficient search method for the dynamic energy consumption minimization problem under time and area constraints that can be applicable to the large-size DFGs. The proposed algorithm is based on a genetic algorithm (GA). The critical problem for a GA is to generate non-valid individuals which can slow down or even prevent convergence of algorithms. In our problem, typical crossover methods such as the one-point crossover generate a large number of non-valid individuals that don't satisfy precedence constraint since they don't consider dependencies between nodes in DFGs. To solve the problem, we propose a crossover based on data-flow graph representation. Moreover, we combine a GA and local search heuristic which can get local optima in a limited search space to make the search more efficient[2]. To reduce the power consumption of interconnection network, data transfers between FUs are classified according to FU types of operations in data flow graph. The basic idea behind reducing the complexity of the interconnection network is that the interconnection resource can be shared among data transfers with the same FU type of a source node and the same FU type of a destination node[3]. Moreover, functional-unit binding is merged into scheduling. As a result, the interconnection power is estimated more accurately in the early task, which results in better solution in shorter search time.

II. Problem Definition

An input behavioral description is given by a DFG as shown in Fig.2. Figure 3 shows a datapath architecture, where functional units and registers are connected by multiple buses to support parallel data transfer. The number of FUs, types of FUs, the number of registers, and the number of busses can be changed as long as area and time constraints are satisfied. Connections between FUs are not restricted, and arbitrary point-to-point interconnection between FUs can be implemented. Moreover, the datapath architecture allows both a non-pipelined datapath and a pipelined one with arbitrary degree of spatial parallelism.

We focus on minimization of dynamic energy consumption that is caused by signal transitions in circuits. The technique of gating a clock is used to prevent registers from loading unnecessary new values, so that unnecessary signal transitions in functional units fed by the registers are suppressed. The gated-clock datapath architecture also simplify the objective function of energy consumption minimization problem as described later.

The use of multiple supply voltages is a well-known technique to obtain low energy implementation at reduced performance overhead. In the context of high-level synthesis, one way to utilize multiple supply voltages is module selection that is the process of mapping operations from the DFG to component templates from the RTL library that contains multiple versions of each component corresponding to different supply voltages. Note that only a functional unit template, not

0-7803-9197-7/05/$20.00 © 2005 IEEE.

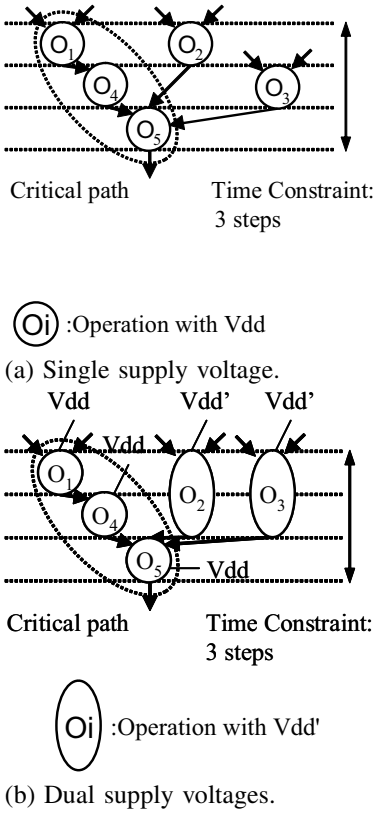

Fig. 1. Power consumption reduction using multiple supply voltages.

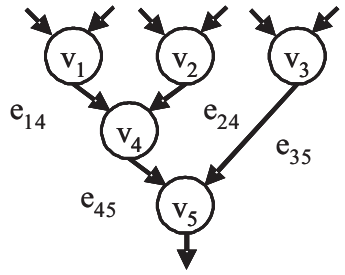

Fig. 2. Data Flow Graph.

a specific instance, is associated with each operation. Table I shows an example of the RTL component library. The OP type denotes an operation type that can be performed by the functional unit templates. For example, functional unit templates of types F_1 and F_3 can perform addition (denoted by "ADD") and multiplication (denoted by "MUL"), respectively. The delay denotes the number of steps for one operation. The energy denotes the average energy consumption for one operation. The functional unit templates have an OP type, a supply voltage, an area, a delay, and an energy.

A. Problem Definition

For the energy consumption minimization problem, we make the following assumptions.

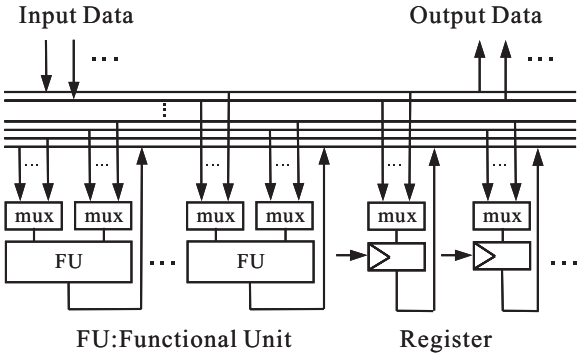

Fig. 3. Architecture model.

TABLE I
RTL COMPONENT LIBRARY

FUtype	OPtype	Supply voltage	Area	Delay	Energy
F_1	ADD	5V	1	1 step	2
F_2	ADD	3V	1	2 step	1
F_3	MUL	5V	8	2 step	16
F_4	MUL	3V	8	4 step	8
F_5	SUB	5V	1	1 step	2
F_6	SUB	3V	1	2 step	1

- delay involved in a register-to-register transfer is negligible.
- The energy consumed by registers and an interconnection network is negligible. The areas are also negligible.
- Static power consumption is negligible.

Basically, scheduling refers to the process of mapping operations to control steps. As can be seen from Table I, multi-cycle operations are used for our problem. Thus, the scheduling is extended to determine a start control step of each operation.

The goal is to minimize the total energy consumed when all the operations are performed. The total energy is simply given by the sum of energy consumption for all the operations, because the gated-clock datapath architecture is employed as described above. The energy consumption for each operation depends on the functional unit to which the operation is assigned, so that the objective function E_{total} is given by

$$E_{total} = E_{fu} + E_{in}, \tag{1}$$

where E_{fu} and E_{in} are the energies consumed by FUs and interconnection units, respectively. The energy E_{fu} is expressed as

$$\sum_{0 \le i \le N} (E_{F_i} \times N_{F_i}) \tag{2}$$

where E_{F_i} is the energy consumed by a functional unit of type F_i and N_{F_i} is the number of all the functional units of type F_i used in the processor. Modelling E_{in} accurately in high-level synthesis is difficult since it requires information in physical synthesis such as placing and routing. To measure E_{in} as accurately as possible, we use fan-out and the number of data-transfers. Thus, the energy consumption minimization

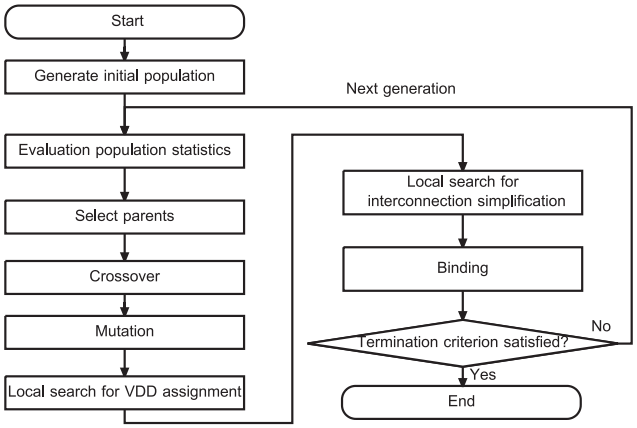

Fig. 4. Flowchart of the basic genetic algorithm.

problem is defined as the problem to schedule operations and assign a functional unit to each operation so as to minimize the energy consumption under given time constraint T_{max} and area constraint A_{max}.

III. GA-Based Efficient Search Method

A. Overview

A Genetic algorithm is stochastic search technique based on the mechanism of natural selection and natural genetics. A genetic algorithm starts with an initial set of random solutions called population. Each individual in the population is called a chromosome which represents a solution to the problem at hand. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated, using some measures of fitness. To create the next generation, new chromosomes, called children, are formed by either (i) merging two chromosomes from current generation using a crossover operator or (ii) modifying a chromosome using a mutation operator. A new generation is formed by (i) selecting, according to the fitness values, some of the parents and children and (ii) rejecting others so as to keep the population size constant. Fitter chromosomes have higher probabilities of being selected. After several generations, the algorithms converge to the best chromosome, which hopefully represents the optimum or suboptimal solution to the problem. Figure 4 shows the flowchart of the GA-based search algorithm. In order to achieve more efficient search, two types of local search is combined the GA. A local search technique is used to find local optima in a given problem search space and a genetic algorithm is used to search the space of local optima in order to find the global optimum. The local search for VDD assignment improves scheduling and module selection. The local search for interconnection simplification improves scheduling such that interconnection units are shared as much as possible in binding.

We can use the following string for the problem with n nodes because the chromosome representation for the problem must contain the information of both scheduling and module selection.

$$x_1\ y_1\ x_2\ y_2\ x_3\ y_3\ \ldots\ x_n\ y_n$$

where x_i is the start control step of operation o_i and corresponds to scheduling, y_i is the functional unit template which is assigned to operation o_i and corresponds to module selection.

For our problem, typical crossover methods such as the one-point crossover generate a large number of non-valid individuals which slow down or even prevent convergence of algorithms, where the non-valid individuals are defined as individuals which do not satisfy the precedence constraint. To solve this problem, we use a crossover method that groups as many nodes with dependencies as possible[2]. It is based on the idea that nodes in the same group should satisfy the precedence constraint.

B. Local search for VDD assignment

The local search is applied to new children generated by a crossover and mutation operators. All the individuals in the population obtained by the local search represent local optima. They are evaluated based on their energy consumption values. Promising individuals are selected from the set of local optimal solutions to form the next generation.

We describe a local search for our problem. The local search is applied to all individuals in every generation. The algorithm is shown as follows.

Step1: Select one individual (I_i) from the population (P), where P is a set of individuals generated by crossover and mutation operators. $P = P - \{I_i\}$;
Step2: Select one operation (o_i) from O_{I_i}, where O_{I_i} is a set of nodes in the individual (I_i). $O_{I_i} = O_{I_i} - \{o_i\}$;
Step3: Search a feasible scheduling and module selection for operation o_i to improve the solution, while the scheduling and module selection for all the operations except operation o_i are fixed.
Step4: if $O_{I_i} \neq \phi$ then go to Step2
Step5: if $P \neq \phi$ then go to Step1

Since the scheduling and module selection for every operation except operation o_i are fixed and the local optima are found in reasonable time. Suppose that an individual shown in Fig.5(a) is given. Let us explain the local search for operation o_1. In this case, the scheduling and module selection for all the operations except operation o_1, that is, operations o_2, o_3, o_4 and o_5 are fixed. A feasible scheduling and module selection for only operation o_1 are searched. The resulting individual obtained by the local search for operation o_1 is shown in Fig.5(b), where $V'_{dd} < V_{dd}$. The functional unit which is assigned to operation o_1 changes from a high voltage unit to a low voltage one. The energy consumption for a operation o_1 is reduced, that is, the solution is improved.

C. Local search for interconnection simplification

This local search is based on force-directed shceduling for E-instances, where an E-instance is defined as a pair of nodes connected by an edge. E-instances are classified into types called E-templates based on the type of their source and destination node[3]. Figure 6 shows this original definition of E-instances and E-templates. Four E-instances are generated

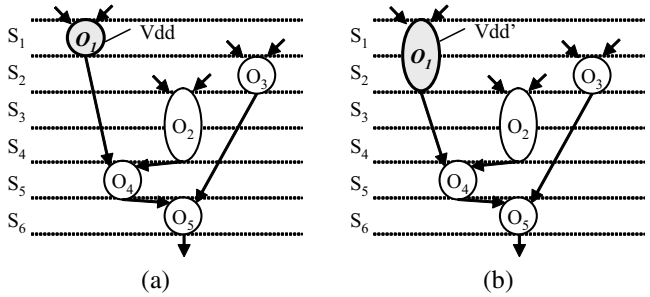

Fig. 5. Example of a local search for a operation o_1.

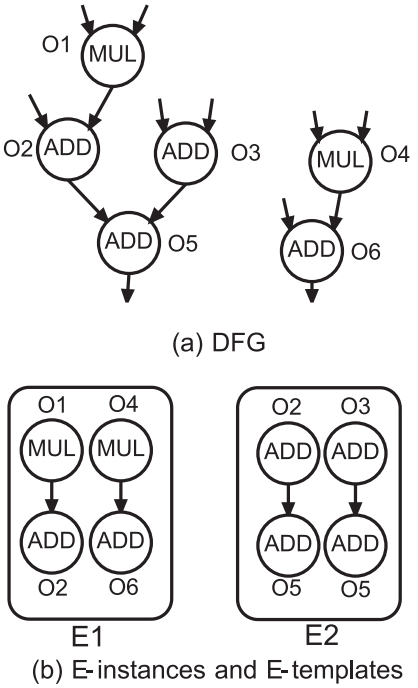

Fig. 6. Original definition of E-instances and E-templates.

from the DFG in Fig. 6(a). They are classified into two E-templates based on the operation types of a source node and a destination node. E-instances in the same E-template can share the same inteconnection unit if they are not overlapped in execution. Thefore, it is desireble to schedule E-instances such that they are not overlapped in execution, and to allocate them to the same hardware resources. Figs. 7(b) and (a) show the binding results based on the idea and not, respectively. You can see Fig. 7(b) provides a simple interconnection network.

We extend the concept of E-templates to the multiple-suply-voltate scheme(Figure 8). E-template are defined based on the operation and FU types of a source and destination node of a E-instance. Figure 8 shows the clasiffication based on the definition. Note that the supply voltage for each node can be available after the local serch for VDD assignment.

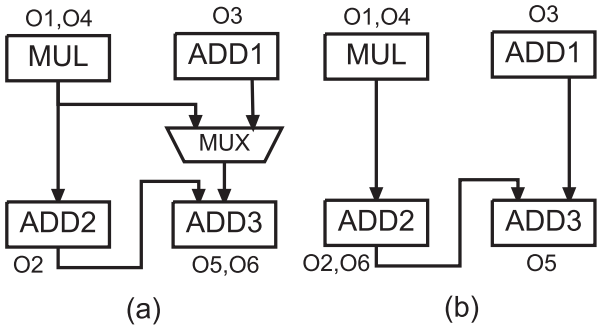

Fig. 7. Binding.

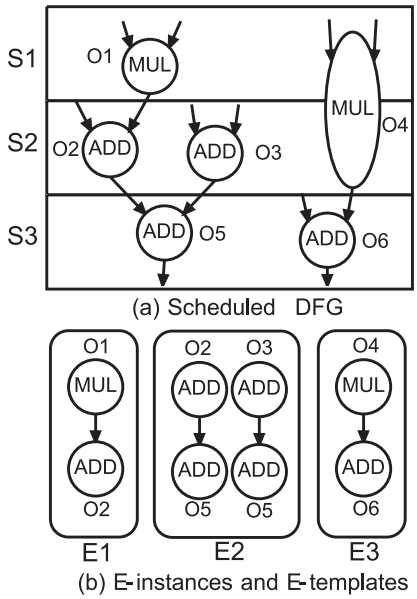

Fig. 8. Extended definition of E-instances and E-templates.

This extention of E-templates make it possible to combine the VDD assignment and interconnection simplification into the same loop.

IV. Conclusion

We proposed the simultaneous scheduling and module selection considering both of powers due to FUs and interconnection units. The proposed method is useful not only for ASICs but also for reconfigurable processor where interconnection power is more serious problem.

References

[1] K. Usami and M. Horowitz, "Clustered Voltage Scaling Technique for Low-Power Design," in Proceedings International Workshop on Low Power Design, 1995.

[2] M. Hariyama, T. Aoyama, and M. Kameyama,"Genetic Approach to Minimizing Energy Consumption of VLSI Processors Using Multiple Supply Voltages", IEEE Transaction on Computers, pp.642-650(2005).

[3] R.Mehra and J.Rabaey,"Exploiting Regularity for Low-Power Design,"Proc. Inter. Conf. Computer-Aided Design, pp.166-172(1996).