Complexity Reduction in an nRERL Microprocessor

Seokkee Kim School of Electrical Engineering Seoul National University Seoul, 151-742, Korea

+82-19-576-0597

seokkee@sdgroup.snu.ac.kr

Soo-Ik Chae School of Electrical Engineering Seoul National University Seoul, 151-742, Korea

+82-2-880-1815

chae@sdgroup.snu.ac.kr

ABSTRACT

We describe an adiabatic microprocessor implemented with a reversible logic, nRERL [1]. We employed an 8-phase clocked power instead of 6-phase one to reduce the number of buffers required for the phase aligning in the adiabatic microprocessor. Furthermore, by breaking the logic reversibility with self-energy recovery circuits, we also reduced its complexity as well as its energy consumption.

We integrated an 8-bit nRERL microprocessor with an 8-phase clocked power generator into a chip with 0.25μ m CMOS technology. Its minimum energy consumption of 4.67μ A/MHz was measured at V_{dd}=2.4V and *f*=651kHz, which was about 40% compared to the previous 6-phase version. Its circuit complexity was also reduced down to 65% that of its 6-phase version.

Categories and Subject Descriptors

B.7.1 [**Integrated Circuits**]: Types and Design Styles – *Microprocessors and microcomputers*.

General Terms

Design.

Keywords

Microprocessor, nMOS Reversible Energy Recovery Logic (nRERL), Clocked Power Generator (CPG), Complexity Reduction, Buffer skipping, Reversibility Breaking.

1. INTRODUCTION

The minimum current consumed in the adiabatic circuits using reversible logic is bounded with the leakage-current level if all the nonadiabatic energy losses are eliminated. The reversible adiabatic circuits become substantially more complex due to the garbage signals added to make logic functions reversible. Thus, the total energy consumption can be increased if the circuit complexity of the reversible logic is not controlled properly.

The circuit complexity of the adiabatic circuits becomes even

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ISLPED'05, August 8-10, 2005, San Diego, California, USA.

Copyright 2005 ACM 1-59593-137-6/05/0008...\$5.00.

larger due to the phase aligning especially in multi-phase reversible logic such as 6-phase or 8-phase nMOS reversible energy recovery logic (nRERL). Therefore, it is necessary to control the circuit complexity to improve energy efficiency in implementing a complex adiabatic circuit by using multi-phase reversible logic.

No work has been reported about the adiabatic microprocessor integrated with its clocked power generator (CPG) and all of its functional blocks implemented with adiabatic circuits. Although several adiabatic circuits using reversible logic have been reported, their applications have been limited to the simple circuits such as buffers [1], adders [2], multipliers, or register files [3]. Although a couple of works in [4, 5] applied energy recovery circuits to a complex system, they just recycled the energy of the nodes with large capacitance only.

In this paper, we described an 8-bit adiabatic micro-processor with reduced complexity, which uses 8-phase nRERL instead of 6-phase nRERL [6, 7]. All of its functional blocks are implemented only with nRERL but only the controller part of its CPG is implemented with the conventional CMOS static logic.

The outline of this paper is as follows. We briefly explain some previous works that were used in the organization of the nRERL microprocessor in Section II. Then, we describe two techniques to reduce the circuit complexity in Section III and the architecture of the microprocessor with reduced complexity in Section IV, which is followed by the measurement results and the conclusions in Section V and Section VI, respectively.

2. PREVIOUS WORKS

In this section, we briefly explain some previous works, which were used to implement an adiabatic microprocessor such as reversible logic, nRERL [1], self-energy recovery circuits (SERCs) [1], adiabatic memories [3], and a CPG [6, 8].

2.1 6-phase nRERL

The circuit complexity of a reversible adiabatic logic is known to be substantially larger compared to that of the conventional CMOS static logic. Among various reversible adiabatic logic families, the circuit complexity of an nRERL circuit is relatively low compared to those of other reversible adiabatic circuits, because the nRERL circuit uses only nMOS transistors. Therefore, nRERL is more suitable in implementing a complex logic circuit such as a microprocessor. The detailed description of nRERL can be found in [1].

2.2 Self-Energy-Recovery Circuit (SERC)

An SERC can recover most of the energy stored at a signal node with its own data without reversible logic [1]. Therefore, we can use the SERCs to reduce the circuit complexity by breaking the logic reversibility. It has nonadiabatic energy loss, which is proportional to the square of the threshold voltage of its diode-connected transistors as shown in Fig. 1. Note that the ϕ_{i+1} is driven to V_{dd} while recovery phase.



Fig. 1. A self-energy-recovery circuit.

2.3 Adiabatic Memories

To implement a fully adiabatic memory, all the data written in the memory must be kept or recovered without being destroyed. In other words, a large memory is also required to store all the data for the backward computation. The overhead of a fully adiabatic memory is too large to be used in a practical design. Therefore, we simply broke the reversibility by using SERCs with minimal nonadiabatic loss to implement the memories in a limited silicon area.



Fig. 2. (a) An array of ROM cells and (b) a register file cell

Fig. 2 shows the schematics of nRERL memory cells. Note that there is no nonadiabatic energy loss in the ROM because stored data are not changed in a read operation in the ROM cell as shown in Fig. 2(a).

Nonadiabatic energy loss occurs when we overwrite a new data on a cell in the adiabatic memories such as register file or RAM. To eliminate this nonadiabatic energy loss, we need to change the cell state into a known state before writing a new data, which is called a unwrite operation. We added an SERC in each memory cell for unwriting so that the unwrite operation has minimal nonadiabatic loss and the write operation after unwriting can be adiabatic as shown in Fig. 2(b). Therefore, the adiabatic memories can be operated with reduced nonadiabatic loss for unwriting and only with adiabatic loss for writing and reading. The nRERL storage cell was described in detail in [3].

2.4 Clocked Power Generator (CPG)

The CPG is one of the most energy dissipating blocks in nRERL circuits. We employed an LC-resonant CPG, because it is more energy-efficient than a capacitor-based one [9]. We found that the energy portion of CPG in the total energy consumption is about 50% in the previous nRERL microprocessor [6, 7].

To design an energy-efficient CPG, we determined its LC oscillation frequency properly as well as sized its rail-drivers [8, 9]. Furthermore, we used two compensation circuits to balance the capacitance of each rail statically and dynamically as shown in Fig. 3, which is essential to control the LC-resonant frequency to improve energy efficiency. For static compensation, we compensated the rail-to-rail capacitance mismatches by adding capacitors to the rails that has less capacitive loads. For dynamic compensation, we adjusted current flow to compensate temporal variations of the rail capacitances. The detailed explanation of the CPG compensation was given in in [6]. Since the imbalance of the load at each of the clock power signal leads to the additional energy loss, the propose load balancing technique can reduce the total energy consumption significantly.



Fig. 3. An 8-phase LC-resonant clocked power generator with compensation.

3. PROPOSED COMPLEXITY REDUCTION TECHNIQUES

In designing a microprocessor, we used 8-phase nRERL so that we could apply buffer skipping to exploit its phase margins. Moreover, we also broke logic reversibility with SERCs if we could reduce the energy consumption as well as the circuit complexity. In the following sections, we describe the two proposed complexity reduction techniques one by one for simple explanations.

3.1 Buffer Skipping in 8-phase nRERL

In multi-phase nRERL, it is necessary to use many buffers to retain the data for phase aligning. In other words, the buffers are used for the data retaining during several phases until a right phase is arrived. If the data can be retained with less buffers, the circuit complexity of the reversible adiabatic circuits can be reduced substantially. In Fig. 4, (a) is a schematic of an nRERL buffer, (b) illustrates its 8-phase clocked powers, (c) and (d) show how complexity reduction is achieved with the buffer skipping technique in buffer chains and their waveforms, respectively.

In 8-phase nRERL, the output of a buffer driven by ϕ_i is valid for five phases ($T_2 \sim T_6$), which is two phases longer compared to 6phase nRERL. Hence, it can be cascaded with a buffer driven by one of the three clocked powers (ϕ_{i+1} , ϕ_{i+2} , ϕ_{i+3}) in nRERL buffer chains as shown in Fig. 4(c). In other words, we can skip two consecutive buffers by cascading every third clocked power in a buffer chain such as ϕ_i , ϕ_{i+3} , ϕ_{i+6} , ϕ_{i+1} , ..., and so on. Therefore, we can substantially reduce the circuit complexity by skipping the buffers in the buffer chains in 8-phase nRERL.



Fig. 4. (a) An nRERL buffer and (b) 8-phase clocked power. (c) Buffer skipping in buffer chains. (d) Waveforms of the data indicated in (c).

This buffer skipping technique is useful especially for the functional blocks such as an ALU, a forwarding logic, and a datapath. Therefore, we could substantially reduce the hardware complexity as well as the energy consumption of the implemented 8-bit nRERL microprocessor with this proposed buffer skipping technique.

The overheads of generating and distributing an 8-phase clock are not significant. For the generation of the 8-phase clocked power, only a simple modification of the 3-bit counter in the CPG controller is needed. Although the number of rail-drivers is increased from 6 to 8, the optimal size of each rail driver is reduced because the capacitive load of each rail is reduced. For the distribution of the 8-phase clock, only 2 more additional clock routings are required, but its area overhead is negligible in the current CMOS process with multiple metal layers.

Using the buffer skipping in 8-phase nRERL, the circuit complexity can be reduced to about 40% excluding the memories. Note that the circuit complexity of the memories is not reduced substantially because the array of memory cells takes a large portion in memory blocks and each memory cell is not changed.

3.2 Logic Reversibility Breaking with SERCs

In implementing the adiabatic microprocessor, we broke the logic reversibility of the garbage signals only if the energy consumption is reduced. Here, we briefly explain how we can reduce both circuit complexity and energy consumption with SERCs in a 4-bit ripple carry adder (RCA) shown in Fig. 5(a). Note that the overheads of the reversible logic for the carries are not constant and that of the LSB carry is the largest as indicated in Fig. 5(a).

In the proposed microprocessor design, the logic reversibility is broken only if the energy reduction is larger than the additional energy loss due to an SERC. For the device parameters of 0.25μ m CMOS process, the energy consumption of an SERC is about 7.3 times and that of carry generator is about 2.1 times compared to that of a buffer at the optimal condition of V_{dd}=2.4V and *f*=651kHz. Therefore, we broke the logic reversibility with an SERC if we could reduce more than 8 buffers. As shown in Fig. 5(b), we optimized the 4-bit RCA by breaking the reversibility of C2* and reducing the reversibility overhead of C1*. Note that reduction of the reversibility overhead for C1* was accompanied with the breaking the logic reversibility of C2*.



Fig. 5. 4-bit ripple carry adders: (a) without breaking logic reversibility for all intermediate carries and (b) with breaking logic reversibility of an intermediate carry

By applying the reversibility breaking in the microprocessor, we can reduce the circuit complexity by 30% excluding the memories.

Using the two proposed complexity reduction techniques together, we reduced the complexity of the microprocessor down to 40% comparing the one without using it. The details of the microprocessor design with reduced complexity are given in the next section.

4. MICROPROCESSOR DESIGN WITH REDUCED COMPLEXITY

We designed a simple 8-bit nRERL microprocessor based on the instruction set architecture of DLX [10]. The DLX instruction set architecture was simplified due to the complexity of the nRERL microprocessor and the limited chip area.

In the original DLX, there are several types of instructions: loads and stores, ALU operations, branches and jumps, and floatingpoint operations. In the simplified instruction set architecture of DLX, only 19 instructions were supported such as add, sub, and, or, xor, slt, addi, lw, sw, jr, jalr, sp, beqz, bnez, j, jal, nop/ref (refresh), lwp and swp. The instruction width was reduced to 20 bits and the datapath width was 8 bits. Both the op-code and function code were also reduced to 4 bits.

We used 8-phase nRERL to reduce the complexity by using buffer skipping for energy-efficient design of the microprocessor, as shown in Fig. 6. First, we designed its core functional blocks such as ROM, register file, ALU, and RAM, and then optimized the phase scheduling to minimize the number of phase aligning buffers. The number of buffers required for data retaining in datapath was reduced to about one third, from 450 to 190 by exploiting the phase margins of 8-phase nRERL as indicated with dashed triangles in Fig. 6.



Fig. 6. Phase scheduling in the 8-phase nRERL microprocessor.

After applying the buffer-skipping technique, we applied the reversibility breaking technique to the microprocessor. We used about 550 SERCs for breaking logic reversibility, which occupies about 2.0% in the circuit complexity of the microprocessor. Without reversibility breaking, the total circuit complexity would be increased by more than 40%. Note that the circuit complexity of an SERC is two fifths that of an nRERL buffer.

With the two proposed complexity reduction techniques, we could reduce the circuit complexity of the microprocessor by 34%, compared to the previous 6-phase nRERL version, as shown in Table I.

Table I. Comparison of t	ne circuit	complexities	between	two
versions of nRERL micro	processors	5.		

	# of transistors			
Functional Subblock	6-phase Microprocessor (portions to core)	8-phase Microprocessor (portions to core)		
ROM (64w x 20b)	10,024 (13.3%)	7,912 (16.0%)		
PC	17,026 (22.6%)	5,588 (11.3%)		
ALU	5,196 (6.9%)	2,196 (4.5%)		
Register file (16w x 8b)	7,588 (10.1%)	5,844 (11.7%)		
Forwarding Logic	390 (0.5%)	189 (0.4%)		
RAM (128w x 8b)	28,042 (37.2%)	22,976 (46.5%)		
Controller	5,672 (7.6%)	4,304 (8.7%)		
Phase-aligning Buffers	1,384 (1.9%)	488 (1.0%)		
Microprocessor Core	75,322 (100%)	49,497 (100%)		

The area overhead of the CPG and its clock distribution in 8phase nRERL is only about 5%, compared to that in 6-phase nRERL. With buffer skipping, we obtained relatively high complexity reduction ratio (52 to 67 %) for the functional blocks such as the ALU, forwarding logic and the program counter. However, the memory cells are not changed basically. In conclusion, we could reduce the ratio of circuit complexity of the nRERL microprocessor to that of its conventional CMOS static logic version down to about 3.3 times.

5. MEASUREMENT RESULTS

An 8-bit nRERL microprocessor chip was fabricated with 0.25μ m 5-metal n-well CMOS process: V_{dd} =3.3V, V_{tho} =0.57V and V_{thb} =0.64V. Fig. 7 shows the microphotograph of the microprocessor. The microprocessor core and its CPG occupy 2.07 x 3.00 mm² and 0.87 x 0.75 mm², respectively. The area of the implemented microprocessor is about ten times compared to that of CMOS version. It is because the area of the nRERL microprocessor is not optimized well for simplifying the design such as the multi-phase clocked power routings.



Fig. 7. Microphotograph of the 8-b nRERL microprocessor.

We measured the energy consumption for several instruction sequences, as shown in Fig. 8. The energy consumption increased as the difference between the reference and oscillation frequencies gets larger. The measurement results showed that the nRERL microprocessor consumed 7.3pJ/cycle on the average at V_{dd} =2.4V and *f*=651kHz for a test program in which the memory-access instructions is 30%, which corresponds to about 4.67µA/MHz. Note that the energy consumed in the memory access instructions are about 10 to 15% higher.



Fig. 8. Measured energy consumed per cycle for several instruction sequences in a test program.

Fig. 9 shows the energy consumption of functional blocks and types of the energy loss. About a half of the total energy consumption was consumed in the adiabatic microprocessor core and the other half is consumed in its CPG as shown in Fig. 9 (a). For the minimum energy consumption [8, 9], the operating frequency was adjusted to balance the adiabatic and leakage losses while the rail-drivers in the CPG were also sized to balance the rail-driver's energy consumption and the adiabatic loss. Therefore, leakage loss, adiabatic loss, and rail-driver's energy consumption are almost equally partitioned except SERC's and CPG controller's energy consumption, which are constant over the operating frequency, as shown in Fig 9(b). Note that the CPG controller, which is a CMOS static logic circuit, occupies about 18% of the total energy consumption of the CPG. With simulation, we confirmed that it could be reduced to about 2.5% if the supply voltage of the CPG controller is scaled down from 2.4V to 0.8V.



Fig. 9. Energy partition for the nRERL microprocessor: partitioned (a) with the functional blocks and (b) with the energy loss types.

Fig. 10 shows the compared energy consumptions of the nRERL microprocessor and its conventional CMOS static logic version based on the HSPICE simulations. The energy consumption of the nRERL microprocessor at V_{dd} =2.4V is about one order-of-magnitude lower compared to that of its conventional CMOS static logic version at V_{dd} =0.8V. This result shows that the nRERL can be a good alternative for ultra-low-energy applications if it is optimized properly.



Fig. 10. Energy consumption in nRERL microprocessor and its conventional CMOS static logic version.

6. CONCLUSIONS

We proposed two complexity reduction techniques: buffer skipping that exploits the phase margin of 8-phase nRERL and breaking logic reversibility with SERCs. We applied them to an 8-bit nRERL microprocessor design and reduced its energy consumption as well as its hardware complexity. With measurement results, we found that the minimum energy consumption of the nRERL microprocessor was about 4.67 μ A/MHz at V_{dd}=2.4V and *f*=651kHz, which is about an order-of-magnitude lower compared to that of its CMOS logic version. We also found that the energy consumption of the 8-phase nRERL microprocessor was reduced to about 40% and the circuit complexity was reduced to 65% compared to its previous 6-phase version. In conclusion, we showed that we could optimize an adiabatic microprocessor effectively with buffer skipping and logic reversibility breaking.

7. ACKNOWLEDGMENTS

The test chip was fabricated with the help of Inter-University Semiconductor Research Center in Seoul National University and IC Design Education Center in Korea Advanced Institute of Science and Technology.

8. REFERENCES

- J. Lim, D. -G. Kim, and S. -I. Chae, "nMOS reversible energy recovery logic for ultra-low-energy applications," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp. 865-875, Jun. 2000.
- [2] J. Lim, D. -G. Kim, and S. -I. Chae, "A 16-b carry-lookahead adder using reversible energy recovery logic for ultra-lowenergy systems," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 6, pp. 898-903, June 1999.

- [3] J. -H. Kwon, J. Lim, and S. -I. Chae, "A Three-Port nRERL Register Rile for Ultra-Low-Energy Applications," *Proc. Int. Symposium on Low-Power Electronics and Design*, July 2000, pp.155-160.
- [4] W. C. Athas, N. Tzartzanis, L. "J." Svensson, L. Peterson, H. Li, X. Jiang, and W. -C. Liu, "AC-1: a clock-powered microprocessor," *Proc. Int. Symposium on Low-Power Electronics and Design*, Aug. 1997, pp. 328-333.
- [5] W. C. Athas, N. Tzartzanis, L. "J." Svensson, and L. Peterson, "A low-power microprocessor based on resonant energy," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 11, pp. 1693-1701, Nov. 1997.
- [6] S. Kim, J.-H. Kwon, and S. –I. Chae, "An 8-b nRERL Microprocessor for Ultra-Low-Energy Applications," *Proc. ASP-DAC 2001*, Jan. 2001, pp. 27-28.

- [7] S. Kim and S. –I. Chae, "Implementation of a simple 8-bit microprocessor with reversible energy recovery logic," *Proc. Computing Frontiers 2005*, May. 2005, pp. 421–426.
- [8] J. Lim, D. -G. Kim, and S. -I. Chae, "Reversible energy recovery logic circuit and its 8-phase clocked power generator for ultra-low-power applications," *IEICE Trans. on Electronics*, vol. E82-C, no. 4, pp. 646-653, April 1999.
- [9] W. C. Athas and L. "J." Svensson, "Reversible logic issues in adiabatic CMOS," *Proc. Workshop on Physics and Computing '94*, pp. 111-118, Nov. 1994.
- [10] D. A. Patterson and J. L. Hennessy, Computer architecture: a quantitative approach, Morgan Kaufmann Publishers, 1990.