# Design of a 16-bit RISC CPU Core in a Two Phase Drive Adiabatic Dynamic CMOS Logic

Yasuhiro Takahashi*, Daijiro Tsuzuki*, Toshikazu Sekine*, and Michio Yokoyama†
*Department of Electrical and Electronic Engineering, Gifu University, Gifu-shi, 501-1193 Japan.
†Department of Bio-System Engineering, Yamagata University, Yonezawa-shi, 992-8510 Japan.

***Abstract*— We propose a design of a 16-bit RISC CPU core using an adiabatic logic which is called a two phase drive adiabatic dynamic CMOS logic (2PADCL), in this paper. The proposed adiabatic RISC CPU is non-pipelined with a latency of three cycles, and also consists of six blocks; an arithmetic and logic unit (ALU), a program counter, a register file, an instruction decoder unit, a multiplexer and a clock control unit. Through the SPICE simulation, the 2PADCL CPU was evaluated for 0.35 $\mu$m standard CMOS library and was compared with the CMOS CPU. The simulation results show that the power consumption of the adiabatic CPU is about $1/4$ compared to that of the CMOS CPU.**

## I. Introduction

As operating frequencies and circuit densities have increased, energy dissipation and power flux have become problematic in a wide variety of digital devices, ranging from small portable systems (e.g. laptops and Personal Digital Assistants), where battery size, weight and operational life are critical, to large computing machines. Power consumption and dissipation within digital electronic devices is largely attributable to switching activities occurring within components of such devices. In recent years, adiabatic switching has been proposed as a method of reducing switching activities [1]–[9]. In [8] and [9], we proposed a new topology for the adiabatic dynamic circuit which called a two phase drive adiabatic dynamic CMOS logic (2PADCL). The 2PADCL achieves ultra low energy dissipation by restricting current to flow across devices with low voltage drop and by recycling the energy stored in internal capacitors.

In this paper, we describe a design of a reduced instruction set computer (RISC) CPU using 2PADCL circuit technology. The basis of the 2PACL is presented in Section II. In Section III, we design a 16-bit 2PADCL RISC CPU. The proposed CPU is non-pipelined with a latency of three cycles. The CPU also consists of six blocks; an arithmetic and logic unit (ALU), a program counter (PC), a register file (REG), an instruction decoder unit (IDU), a multiplexer (MUX), and a clock control unit (CCU). Section IV shows that the performance of the proposed adiabatic CPU is compared with that of the static CMOS CPU. The conclusions are summarized in Section V.

## II. Adiabatic Logic

### A. Conventional vs. Adiabatic Switching

The conventional switching can be understood by using a simple CMOS inverter. The CMOS inverter can be considered to consist of a pull-up and pull-down networks connected to a load (or internal) capacitance $C$. The pull-up and pull-down networks are actually MOS transistors in series with the same load $C$. Both transistors can be modeled by an ideal switch in series with a resistor which is equal to the corresponding channel resistance of the transistor in the saturation mode. When a conventional CMOS inverter is set into a logical "1" state, a charge $Q = CV_{DD}$ is delivered to the load and the energy which the supply applies is $E_{applied}{=}QV_{DD}{=}C{V_{DD}}^2$. The energy stored into the load $C$ is a half of the supplied energy:

$$E_{stored} = \frac{1}{2}C{V_{DD}}^2. \tag{1}$$

The same amount of energy is dissipated during the discharge process in the NMOS pull-down network because no energy can enter the ground rail $Q \times V_{gnd} = Q \times 0 = 0$. From the energy conservation law, a conventional CMOS logic emits heat and, in this way, it wastes energy in every charge-discharge cycle:

$$\begin{aligned} E_{total} &= E_{charge} + E_{discharge} \\ &= \frac{1}{2}{V_{DD}}^2 + \frac{1}{2}C{V_{DD}}^2 \\ &= C{V_{DD}}^2. \end{aligned} \tag{2}$$

If the logic is driven by a certain frequency $f\ (= 1/T)$, where $T$ is the period of the signal, then the power of the CMOS gate is determined as:

$$P_{total} = \frac{E_{total}}{T} = C{V_{DD}}^2 f. \tag{3}$$

The main idea in an adiabatic switching is that transitions are considered to be sufficiently slow so that heat is not emitted significantly. This is made possible by replacing the DC power supply by a resonance LC driver or oscillator. If a constant current source delivers the $Q = CV_{DD}$ charge during the time period $\triangle T$, the energy dissipation in the channel resistance $R$ is given by

$$\begin{aligned} E_{diss} &= P\triangle T \\ &= I^2 R\triangle T \\ &= \left(\frac{CV_{DD}}{\triangle T}\right)^2 R\triangle T. \end{aligned} \tag{4}$$

The above equation indicates that when the charging period $\triangle T$ is indefinitely long, in theory, the energy dissipation is reduced to zero. This is called an adiabatic switching [1].

1-4244-1272-2/07/$25.00 ©2007 IEEE

### B. 2PADCL

The 2PADCL inverter is shown in the top of Fig. 1(a), where the inverter is operated with complementary phases of power supply signals. The supply waveform consists of two modes, "evaluation" and "hold," as shown in the bottom of Fig. 1(a). Let us consider the adiabatic mode. When $V_p$ and $\overline{V_p}$ are in evaluate mode, there is conducting path(s) in either PMOS devices or NMOS devices. Output node may evaluate from low to high or from high to low or remain unchanged, which resembles to the CMOS circuit. Thus, there is no need to restore the node voltage to 0 (or $V_{DD}$) every cycle. When $V_p$ and $\overline{V_p}$ are in hold mode, Output node holds its value in spite of the fact that $V_p$ and $\overline{V_p}$ are changing their values. We can find that such is the case by observing the function of diodes and the fact that the inputs of a gate have a different phase with the output. Circuits node are not necessarily charging and discharging every clock cycle, reducing the node switching activity substantially as shown in Fig. 1(b)[1].

In the proposed 2PADCL circuits, energy dissipation is from the threshold voltage and transistor channel resistance. We use an RC model with a threshold voltage $V_t$ to calculate the energy dissipation in transistor, which can be used estimate the energy consumption in adiabatic circuit. The energy dissipation of 2PADCL can be calculated as follows:

$$E = 2C_{gs}\left(V_p - 2V_d\right)V_d + C_{gs}\left(V_t - V_d\right)^2, \qquad (5)$$

where $C_{gs}$ is a gate-source capacitance in the next stage. Assume, $C_{gs} = 0.02$ pF, $V_t = V_d = 0.8$ V, and $V_p = 5$ V, then we have $E = 0.11$ [pJ/cycle]. Since 2PADCL gate is possible to maintain the output voltage without the load capacitor, its energy consumption can be reduced.

## III. Design of 16-bit RISC CPU

### A. Architecture

The architecture of the proposed CPU is a three-cycle non-pipelined implementation. It is characterized by a RISC typical, uniform 16-bit instruction format [10]. It has a load/store architecture, i.e., communication with main memory is only accomplished by instructions `LOAD` and `STORE`. Operations will only be performed on registers, not on memory locations. The bus protocol is designed for static RAMs, as it is a classical von-Neumann architechiture with just one common memory bus for instructions and data.

To be fitted on a limited silicon area, the 2PADCL CPU supports only a subset of the instruction set, which includes only 26 instructions as shown in Table I. We reduces the instruction width to 16 bits and the datapath width to 8 bits. Both the op-code and function code is also reduced to 5 bits. All the functional blocks are implemented with a 2PADCL, and use a 2-phase clocked power supply.

[1]The 2PADCL inverter having output transition $0 \rightarrow 1$ or $1 \rightarrow 0$ has only shown in the simulation. Just as the inverter is compatible with adiabatic mode under the condition of the phase difference between 0-rad and $\pi/2$-rad, other gates (e.g. 2input-NAND, NOR) and complex gates operate with adiabatic mode under the same condition.

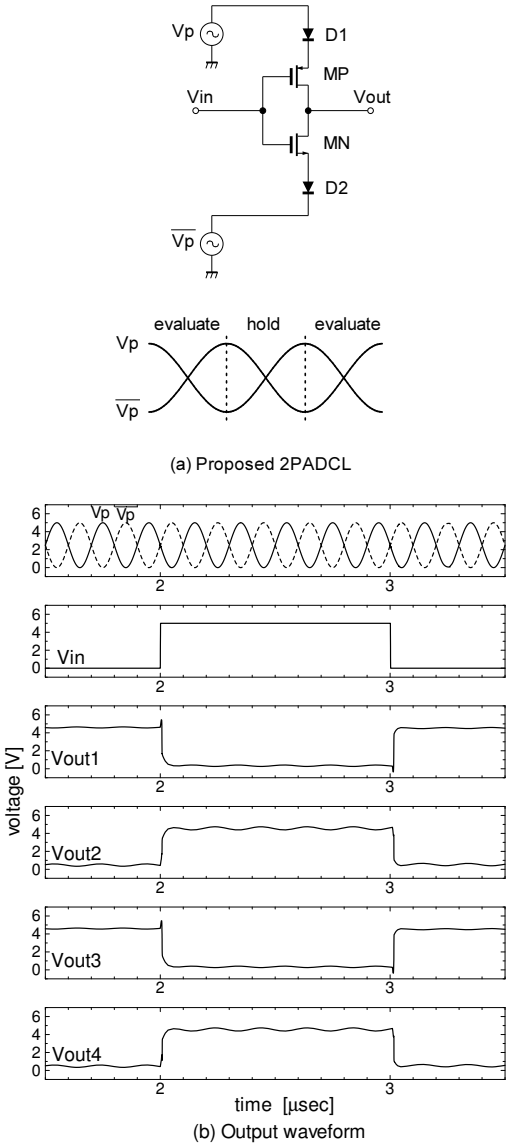


Fig. 1. 2PADCL inverter. (a) Structure and clock of the power supply. (b) Output waveform of the four-inverter chains.

### B. Detail of Logical Blocks

Figure 2 illustrates the block diagram of the adiabatic 16-bit RISC CPU using the 2PADCL. The proposed RISC CPU consists of six blocks; ALU, PC, REG, IDU, MUX and CCU. The datapath of the proposed CPU in this figure is explained as follows.

*1) ALU:* The arithmetic and logic unit (ALU) performs arithmetic and logic operation as well as rotations and shift by a variable distance. The proposed ALU contains three submodules, ARITHMETIC, LOGIC, and SHIFT as shown in Fig. 3. In module ARITHMETIC, additive ALU operations are computed including the arithmetic carry and overflow flags. Module LOGIC performs logical operations. In SHIFT, rotation and shift operations are executed and the shift carry flag is computed.

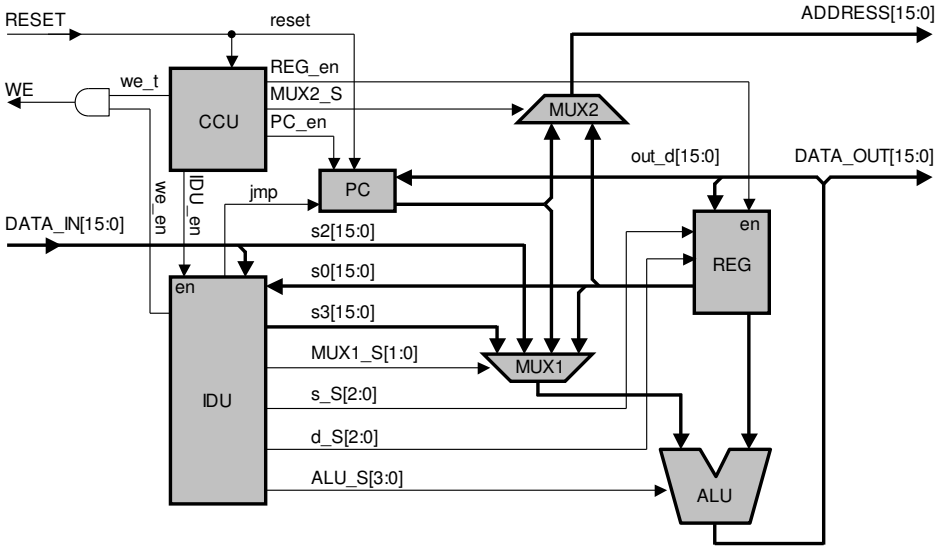


Fig. 2. Block Diagram of an adiabatic RISC CPU datapath.

TABLE I

LIST OF THE INSTRUCTION SET IN 2PADCL CPU

| | | | |
|---|---|---|---|
| MOV | 00000dddsss00000 | MOV d,s | d←s |
| AND | 00001dddsss00000 | AND d,s | d←d & s |
| OR | 00010dddsss00000 | OR d,s | d←d \| s |
| XOR | 00011dddsss00000 | XOR d,s | d←d ⊕ s |
| ADD | 00100dddsss00000 | ADD d,s | d←d + s |
| SUB | 00101dddsss00000 | SUB d,s | d←d − s |
| SL | 01000dddsss00000 | SL d,s | d←s << 1 |
| RL | 01010dddsss00000 | RL d,s | d←s[14:0],s[15] |
| SR | 01001dddsss00000 | SR d,s | d←s >> 1 |
| RR | 01011dddsss00000 | RR d,s | d←s[0],s[15:1] |
| SWP | 01100dddsss00000 | SWP d,s | d←s[7:0],s[15:8] |
| LHI | 11101dddnnnnnnnn | LHI d,N | d←d[15:8],N[7:0] |
| LLI | 11110dddnnnnnnnn | LLI d,N | d←N[7:0],d[7:0] |
| ANDI | 10001dddnnnnnnnn | ANDI d,N | d←d & N[7:0] |
| ORI | 10010dddnnnnnnnn | ORI d,N | d←d \| (hFF,N[7:0]) |
| XORI | 10011dddnnnnnnnn | XORI d,N | d←d ⊕ N[7:0] |
| ADDI | 10100dddnnnnnnnn | ADDI d,N | d←d + N[7:0] |
| SUBI | 10101dddnnnnnnnn | SUBI d,N | d←d − N[7:0] |
| LD | 10000dddsss00000 | LD d,s | d←MEM(s) |
| ST | 11111dddsss00000 | ST d,s | MEM(s)←d |
| JMP | 01111ddddd00000 | JMP d | PC←d |
| PCL | 01110ddddd00000 | PCL d | d←PC |
| JZ | 00110ddddd00000 | JZ d | PC←d (s = 0) |
| JNZ | 00111ddddd00000 | JNZ d | PC←d (s ≠ 0) |
| JP | 10110ddddd00000 | JP d | PC←d (s ≥ 0) |
| JM | 10111ddddd00000 | JM d | PC←d (s < 0) |

*2) PC:* The Program Counter (PC) is a 16-bit latch that holds the memory address from which the next machine language instruction will be fetched. The address at the input to the PC is written into the PC on a leading edge of its write clock. The output of the PC can be used as the read/write address for accessing main memory. The proposed PC is the largest sub-block and second to the control unit in complexity. It has an 8-bit register in a master-slave configuration and performs only two functions: incrementing and loading. For most instructions, the PC is simply incremented in preparation for the following instruction or following instruction nibbles.

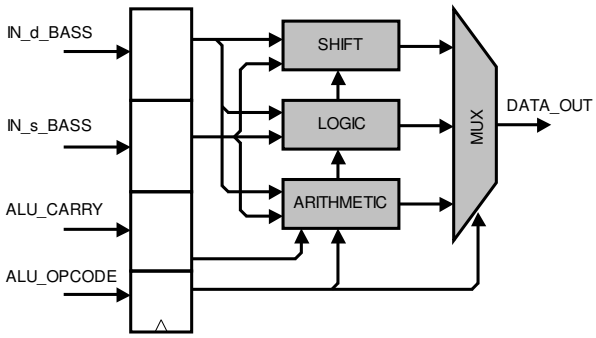


Fig. 3. Structure of arithmetic and logic unit.

*3) REG:* The register file consists of 8 general-purpose registers of 16-bit. It is fully visible for the programmer. Register addresses are 3 bits (1 bit for future file extensions). The register file has a read/write port and an independent read port. Two accesses are possible per port per clock cycle.

*4) IDU:* Our instruction set is simple yet comprehensive. Since our data bus is only 5 bits wide, we decided to keep the number of instructions supported within 32 for easier implementation. The detailed instruction is summarized in Table I.

*5) MUX:* This proposed CPU includes two multiplexer, MUX1 and MUX2. A 4-to-1 multiplexer (MUX1) selects the instruction for the IDU in the next clock. It depends on whether there is a cache hit or a memory access. If there is a normal memory access, the instruction is transferred directly DATA_IN bass. If the REG does not report a hit, but the PC does, the instruction is taken from PC. If the REG reports a hit, the delay instruction from REG is transferred to the IDU. In all remaining cases, the last instruction is taken again which comes from the IDU. A 2-to-1 multiplexer (MUX2) selects the address of

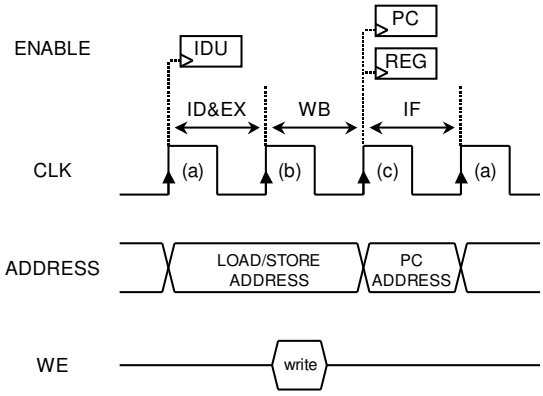


Fig. 4. Adiabatic CPU timing.

the instruction to be fetched in the next clock. The selection depends on whether a new step begins, whether a PC value from is to be taken, whether a branch has to be corrected, or whether a branch was detected.

*6) Clock Control Unit:* Efficient phase scheduling is required to optimize the throughput and the energy consumption of the adiabatic CPU system. In this paper, we propose a clock control unit (CCU) which is tasked with efficient phase scheduling. The proposed CCU is operated as following three steps, as shown in Fig. 4.

- Clock (a): The instruction decode Stage ID decodes instructions and the execute stage EX executes ALU operation, if the IDU is active with the rising clock edge.
- Clock (b): The write-back stage WB results into the register file.
- Clock (c): The instruction fetch state IF are fetched from memory address given by PC if the REG enables. The PC points to the next instruction, and the loaded instruction is transferred to the next stage ID.

## IV. Simulation Results

In order to evaluate the power consumption, we designed the proposed adiabatic CPU. At first, we used a synthesis software to map the proposed CPU on a target library. The target library includes generic and/or technology mapping information. The tool for mapping the Verilog-HDL components to the cell library is the Synopsys *design compiler*. It mapped the Verilog-HDL components to a Rohm 0.35 $\mu$m ASIC standard cell library. The extracted netlist was then converted into the 2PADCL netlist. The 2PADCL netlist includes the diode model for adiabatic operation. Finally, HSPICE simulations were carried out using the converted netlist.

Table II is the comparison of the power consumption and the top clock frequency for 2PADCL and CMOS designs. From this table we can see that the power consumption by the 2PADCL CPU is only $\sim 25\%$ of the conventional CMOS CPU. However, because the proposed CPU operates in the adiabatic mode, the 2PADCL CPU decreases the top clock frequency by $20\%$ as compared to the conventional static CMOS CPU.

TABLE II
Comparison of 2PADCL and CMOS CPU

| | 2PADCL ($\mu$W/MHz) | CMOS ($\mu$W/MHz) |
|---|---|---|
| ALU | 455.1 | 2095.3 |
| PC | 293.3 | 1353.1 |
| IDU | 194.0 | 850.5 |
| REG | 1473.8 | 5525.0 |
| MUX | 174.2 | 799.2 |
| CCU | 27.8 | 131.3 |
| Total | 2620.2 (24.4%) | 10754.4 (100%) |
| Top clock frequency | 16 MHz | 20 MHz |

## V. Conclusions

We have presented a design of 16-bit RISC CPU core using the 2PADCL. The architecture of the proposed CPU has been a three-cycle non-pipelined implementation. A conventional static CMOS CPU with the same structure has been also designed for an energy comparison. We have performed the power and functional simulation using the extracted net-lists from the layout. The power consumption of the proposed CPU has been improved by a factor of four compared to that of the conventional static CMOS CPU.

## Acknowledgment

The custom circuits in this study have been designed with Synopsys CAD tools through the chip fabrication program of VLSI Design and Education Center (VDEC), the University of Tokyo with the collaboration by Rohm Corporation.

The part of this work is supported by a grant from LSI IP design award committee in Japan.

## References

[1] W. C. Athas, L. J. Svensson, J. G. Koller, N. Tzartzains, and E. Y-C. Chou, "Low-power digital systems based on adiabatic-switching principles," *IEEE Trans. Very Large Scale Intgr. (VLSI) Syst.*, vol. 2, no. 4, pp. 398–407, April 1994.

[2] A. G. Dickinson and J. S. Dencker, "Adiabatic dynamic logic", *IEEE J. Solid-States Circuits.*, vol. 30, no. 3, pp. 311–315, April 1995.

[3] Y. Moon, D.K. Jeong, "An efficient charge recovery logic circuit," *IEEE J. Solid-States Circuits.*, vol. 31, no. 4, pp. 514–522, April 1996.

[4] S. Kim and M. C. Papaefthymiou, "True single-phase energy-recovering logic for low-power, high-speed VLSI," in *Proc. IEEE International Symp. Low-Power Electronics and Design*, Monterey, CA, Aug. 10–12, 1998, pp. 167–172.

[5] K. Takahashi and M. Mizunuma, "Adiabatic dynamic CMOS logic circuit," *Electronics and Communications in Japan Part II*, vol. 83, no. 5 , pp. 50–58, April 2000 [*IEICE Trans. Electron.*, vol. J81-CII, no. 10, pp. 810–817, Oct. 1998].

[6] Y. Takahashi, K. Konta, K. Takahashi, K. Shouno, M. Yokoyama, and M. Mizunuma, "Carry propagation free adder/subtracter using adiabatic dynamic CMOS logic circuit technology", *IEICE Trans. Fundamentals.*, vol. E86-A, no. 6, pp. 1437–1444, June 2003.

[7] Y. Ye and K. Roy, "QSERL: Quasi-Static energy recovery logic," *IEEE J. Solid-States Circuits.*, vol. 36, no. 2, pp. 239–248, Feb. 2001.

[8] Y. Takahashi, Y. Fukuta, T. Sekine, and M. Yokoyama, "2PADCL: Two phase drive adiabatic dynamic CMOS logic", in *Proc. IEEE Asia-Pacific Conf. Circuits and Systems*, Singapore, Dec. 4–7, 2006, pp. 1486–1489.

[9] Y. Takahashi, T. Sekine, and M. Yokoyama, "VLSI implementation of a 4×4-bit multiplier in a two phase drive adiabatic dynamic CMOS logic", *IEICE Trans. Electron.*, vol. E90-C, no. 10, Oct. 2007, in press.

[10] K. Nishimura, T. Kudo, and H. Amano, "Educational 16-bit microprocessor PICO-16," in *Proc. 3rd Japanese FPGA/PLD design conference and exhibit* (Japanese Edition), Tokyo, July 19–21, 1995, pp. 589–595.