

Retractable Clock-Powered Logic

Nestoras Tzartzanis and William Athas

{nestoras, athas}@isi.edu

URL: <http://www.isi.edu/acmos>

University of Southern California – Information Sciences Institute
4676 Admiralty Way, Marina del Rey, California 90292-6695

Abstract

Retractable clock-powered logic is presented as a low-overhead energy-recovery logic style. It uses energy-efficient clock-steering circuits, pass-transistor logic, and a four-phase clocking scheme to recover energy from all circuit nodes but the latches. A 16-bit retractile clock-powered adder is described and evaluated through HSPICE simulations. The simulation results indicate that this approach can offer superior energy versus delay performance but the benefit depends strongly on the switching activity of the clock-powered nodes.

1. Introduction

Energy-recovery CMOS is a low-power approach based on adiabatic charging [1], in which circuit energy is recovered and later on reused instead of dissipated as heat. The central idea of adiabatic charging (and adiabatic switching) is that the available and useful energy inside a circuit can be conserved by increasing the time of energy transport between the power supply and the circuit nodes.

A practical form of energy-recovery CMOS is clock-powered logic [2] in which the rise and fall times of the clock signals determine the speed of the energy transport. In clock-powered logic, the on-chip high-capacitance nodes are clocked and powered, i.e., clock-powered, from the clock rails. When coupled with an energy-conserving clock driver [3] [4], the overall system is low power.

Clock-powered logic (CPL) is a primitive and limited application of reversible computing in CMOS VLSI. Circuit nodes that are clock powered use a data representation in which the presence of a pulse denotes a logic one and the absence a logic zero. Clock-powered nodes are only defined when the clock pulse is active (non-zero). It is a reduced-complexity, partially reversible scheme which only recovers energy from the clock-powered nodes. The principal advantage is that it circumvents the prohibitively high overhead of the fully-reversible, fully adiabatic approaches [1] [5]. CPL

has been successfully used with a two-phase resonant clock driver for a number of small-scale microsystems, including a 16-bit microprocessor [2,6].

This paper explores the low-power potential of another design point in the space of reduced-complexity, clock-powered logic styles. In the new style, energy is recovered from all circuit nodes except latches. It is closer to reversible logic at the cost of an increased complexity in the clock waveforms. This new style is called *retractile* clock-powered logic (RCPL) due to its resemblance to retractile cascade logic [7].

In this paper we first present RCPL in detail, and then discuss a 16-bit adder example. The purpose of the adder design is to show how this style can be used for designing clock-powered logic circuits, and to compare it to the original clock-powered style.

2. General Approach

Circuit-wise, RCPL is based on the design style used for the clock-powered logic of our earlier work [2]. The main difference is that in RCPL, combinational logic must be built from pass-transistor gates, while clock-powered logic can be used with myriad combinational logic styles. Both approaches require clock-steering circuits to efficiently drive the clock-powered nodes. These clock-steering circuits, called energy-recovery (E-R) latches (Fig. 1a), operate from two non-overlapping clock phases with voltages that swing from 0 to voltage V_ϕ (Fig. 1b). E-R latches latch their input on one phase (ϕ_L) and drive their output during the other phase (ϕ_D). They consist of two stages: the latch stage and the driver stage. The driver stage is a CMOS bootstrapped clocked buffer [8]. The E-R latch design has been presented in detail elsewhere [2].

Assume that clock-powered signals drive combinational logic blocks based on pass-transistor logic (PTL) as shown in Fig. 2a. If the same clock pulse powers both the signals that drive the transistor gates and the signals that drive transistor chains, then all of the energy for driving the gates is available for recovery, while most of the energy injected through the transistor chains is trapped and eventually dissipated (Fig. 2b). The problem is one of causality. For charging up an initially discharged chain, the drain and gate of each transistor will rise together and the source will charge up to $V_{dd} - V_{tE}$, where V_{tE} is the effective threshold voltage. For discharging, the source and gate also fall at the same

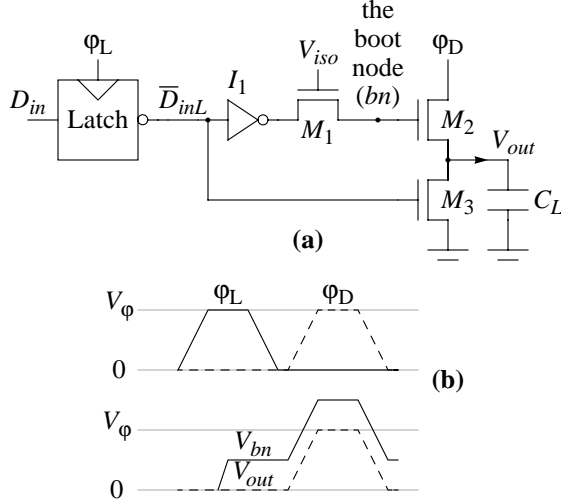


Figure 1: E-R latch (a) and timing diagram when its input is high (b).

rate. The gate-to-source voltage (V_{gs}) remains zero and the chain remains charged up. A negative phase shift between the pulse that drives the gate and that which drives the source will solve this problem since the source will fall faster than the gate and generate a positive V_{gs} . However, for the charge-up process, the drain then rises faster than the gate, and the result is non-adiabatic charging for the chain. A positive phase shift solves this problem, but then exacerbates the former. The desired solution is both a positive and a negative shift, which is to nest the clock pulse that drives the chain inside the clock pulse that drives the gates. This way the energy stored in the pass-transistor chains can be recovered

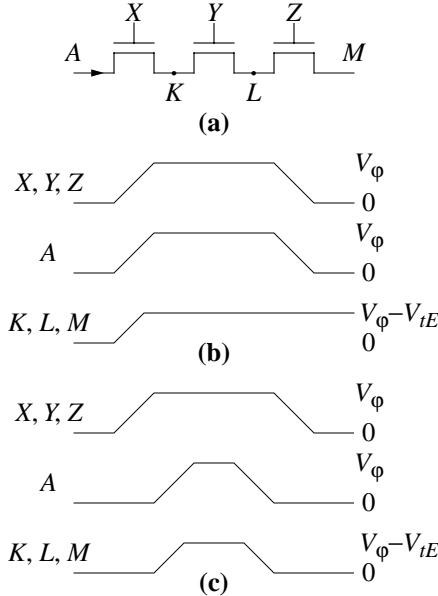


Figure 2: A transistor chain example (a) and a timing diagram sketch when the transistor gates and chain are charged from clock-powered nodes with (b) non-nested and (c) nested clock sources.

through the same path (Fig. 2c). In the timing sketches of Fig. 2, any potential skews between X, Y, Z, and A are disregarded. Furthermore, the delay through the pass transistors is also ignored.

RCPL can be implemented from E-R latches, PTL, and a four-phase clocking scheme (Fig. 3). Pass-transistor gates are powered from wide clock phases (i.e., ϕ_{1w} and ϕ_{2w}) while pass-transistor chains are powered from narrow phases (i.e., ϕ_{1n} and ϕ_{2n}). This clocking scheme is similar to the two-phase, non-overlapping clocking scheme with two additional more narrow phases nested within the wide phases.

An example of RCPL is shown in Fig. 4. Signal z_o is powered from a wide phase (ϕ_{2w}) since it drives a transistor gate. Signal w_o is powered from a narrow phase (ϕ_{2n}) since it drives a transistor chain. The result u_i of the pass-transistor gate is latched at the end of ϕ_{2n} . Data must always be latched on narrow phases, since the data is set to zero when the wide phase goes low or high. With four-phase clock-powered logic, the energy injected on node u_i is recovered at the end of the operation, while in two-phase CPL, this energy would be trapped and dissipated. Signal u_o is powered from ϕ_{1n} , if it drives transistor chains, or from ϕ_{1w} , if it drives transistor gates. Signals that must drive both transistor chains and transistor gates must be latched in two E-R latches. Alternatively, E-R latches could easily be designed to produce two copies of the stored datum (one powered from a narrow phase and one powered from a wide phase). Although signals powered from narrow phases only drive transistor chains, both wide and narrow phases must swing to the same voltage (V_ϕ) since the narrow phases drive the gate of the latch transistor in E-R latches. This may not be required with a different latch-stage design.

The RCPL approach can potentially result in better energy-versus-delay scalability than the two-phase CPL approach because the number of clock-powered nodes is significantly larger. However, there are some drawbacks compared to the two-phase CPL approach. Some of the drawbacks will become apparent in the adder example that follows. First, the benefit of recovering energy from the transistor chains at the end of each operation depends heavily on the switching activity of the transistor-chain nodes. If this energy was trapped in these nodes, it could be locally reused for successive operations, possibly resulting in lower overall dissipation. This is a problem for all approaches that rely on

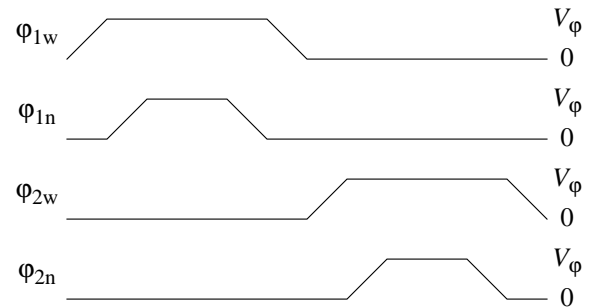


Figure 3: Timing diagram for RCPL.

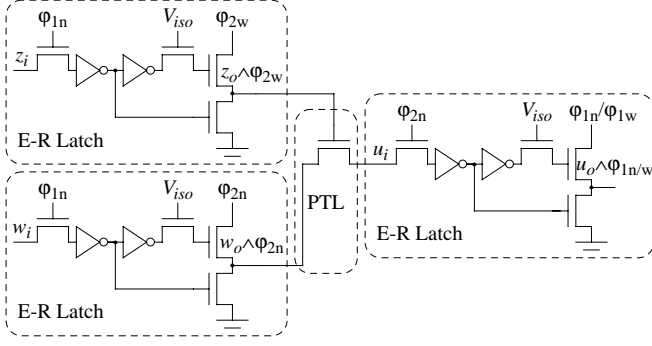


Figure 4: Circuit schematics for RCPL.

immediate energy recovery, i.e., approaches where energy is always recovered at the end of operations, regardless of whether it could be locally reused for the next operation. The effect of unconditional energy recovery in the RCPL approach is hard to assess, due to the large number of clock-powered nodes. Second, RCPL requires some of the energy to be recovered through high-resistance transistor chains, which diminishes the benefit of energy recovery. Third, for the same cycle time, the switching time of the four-phase clock-powered logic must be shorter than the two-phase one due to the nested phases. Finally, it is more difficult to design a high-efficiency clock driver that generates the four phases required by RCPL.

3. Adder Example

This subsection discusses the design of a 16-bit RCPL adder. This adder design can operate from two non-overlapping clock phases, which makes it possible to straightforwardly compare four-phase RCPL and two-phase CPL. The description of the adder design is followed by HSPICE simulation results that compare the two approaches.

The adder is a 16-bit carry select adder organized into four 4-bit stages. The logic is exclusively implemented from PTL gates that require dual-rail inputs. The operation of the adder is pipelined so that a throughput of one addition per clock cycle can be sustained (Fig. 5). Assume that A and B are the 16-bit input operands and C_{in} is the carry-in. First, during ϕ_{1n} the two operands are latched. Then, during the second half of the first cycle, the intermediate results of the 4-bit adder stages are generated. The outputs of the operand latches are powered from the wide phase ϕ_{2w} since they drive transistor gates. The narrow phase ϕ_{2n} is the carry-in for the 4-bit adder stages that receive a one as carry-in. For the adder stages that receive a zero as carry-in, the phase ϕ_{2n} is input as the complement of the carry-in. For both cases, ϕ_{2n} drives transistor chains. Phase ϕ_{2n} is also used to latch the stage results (sum and carry-out bits) as well as the adder carry-in C_{in} . During the first half of the second cycle, the stage results are multiplexed based on C_{in} . The multiplexers are also implemented from PTL gates. The outputs of the latches that hold the stage results are powered from the wide phase ϕ_{1w} since they drive transistor gates. The latch that holds C_{in} powers its output from the narrow phase ϕ_{1n} since

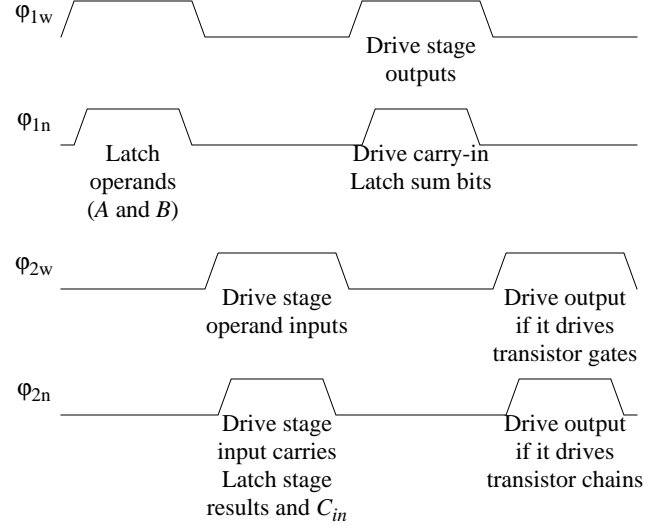


Figure 5: Timing diagram for the RCPL adder.

it drives transistor chains. The results of the addition are latched on ϕ_{1n} . Finally, the adder outputs are powered from ϕ_{2w} if they drive transistor gates, or from ϕ_{2n} if they drive transistor chains.

The adder block diagram is shown in Fig. 6. It consists of E-R latches and PTL gates. All E-R latches receive single-rail inputs and produce dual-rail outputs, except the latches that hold the sum bits, which produce single-rail outputs. Their output form depends on the circuit driven by the adder. Also, most of the latches power their outputs from wide phases. Only the latch that holds C_{in} powers its output from a narrow phase. The carry-in could be latched in the adder on ϕ_{1n} , like operands A and B , and then delayed by one phase until it is needed. Also, for the first four bits, C_{in} could be used to drive a single 4-bit adder stage and produce the actual 4 least significant sum bits. However, for simplicity all sum bits are generated together.

The RCPL adder was designed for the HP CMOS14B process, which is a 0.5- μm , 3.3 V process. The full adder for

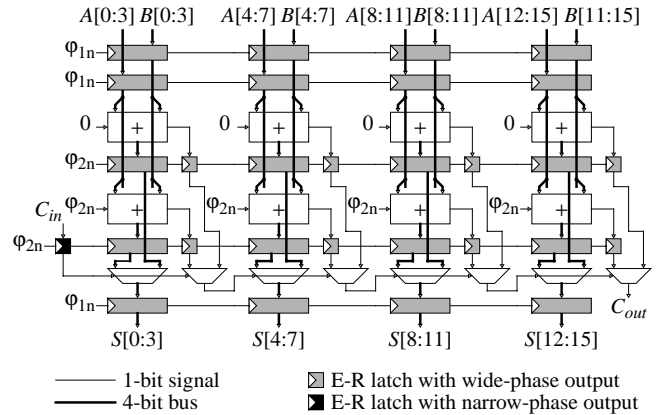


Figure 6: Block diagram for the RCPL adder.

the 4-bit stage adder consists of three simple PTL gates (Fig. 7). All input signals (i.e., the operand bits a_i and b_i and the carry-in c_i) are dual rail. The operand bits that drive transistor gates are powered from ϕ_{2w} . All transistor-chain inputs are powered from ϕ_{2n} or are grounded. Transistors in the carry chain are wider to reduce resistance for worst-case carry propagation, which happens when a carry propagates through all four full adders that constitute the 4-bit adder stage.

The multiplexer cells (Fig. 8) are similar to the cells used for the full adder. For each 4-bit stage there are four multiplexers for the sum bits (Fig. 8a) and two for the stage carry-out bit (Fig. 8b and Fig. 8c), since the latter is required in dual-rail form. Signals s_{i0} and s_{i1} are the results of the 4-bit stage adder with carry-in zero and one, respectively. Likewise, c_{s0} and c_{s1} are the carry-outs of the 4-bit stage adder with carry-in zero and one, respectively. Signal c_{sin} is the actual carry-in for the stage (produced as carry-out from the previous stage). The carry-in for the first stage is the output of the E-R latch that holds C_{in} . All inputs are dual rail. The output sum bits are the results of the addition, which are latched on ϕ_{1n} . The outputs c_{sout} and \bar{c}_{sout} are the positive and negative stage carry-out bits that are passed to the next stage.

The E-R latches are similar to that of Fig. 4, with some minor modifications. For those E-R latches that produce dual-rail outputs, the bootstrap, clamp, and isolation transistors have been duplicated to produce the complementary

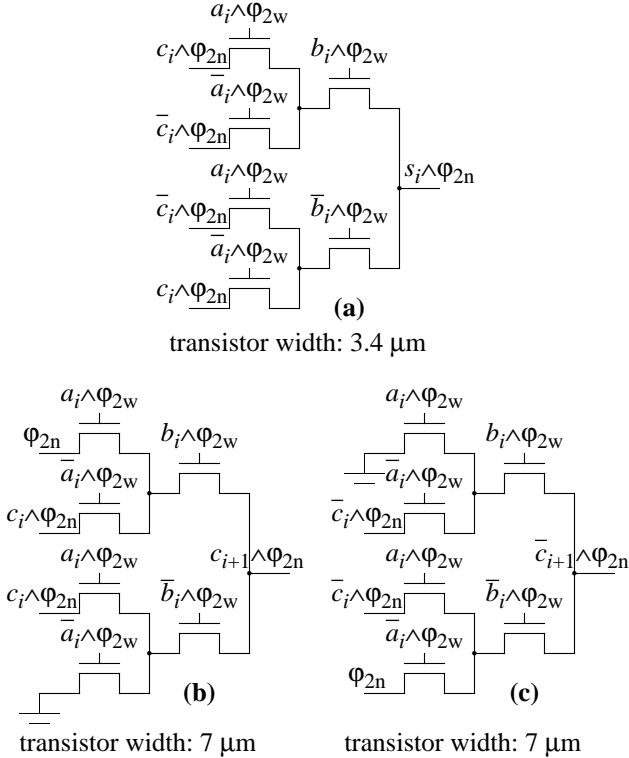


Figure 7: Schematics for the full adder that generates the sum (a) and the carry-out in dual-rail form (b and c).

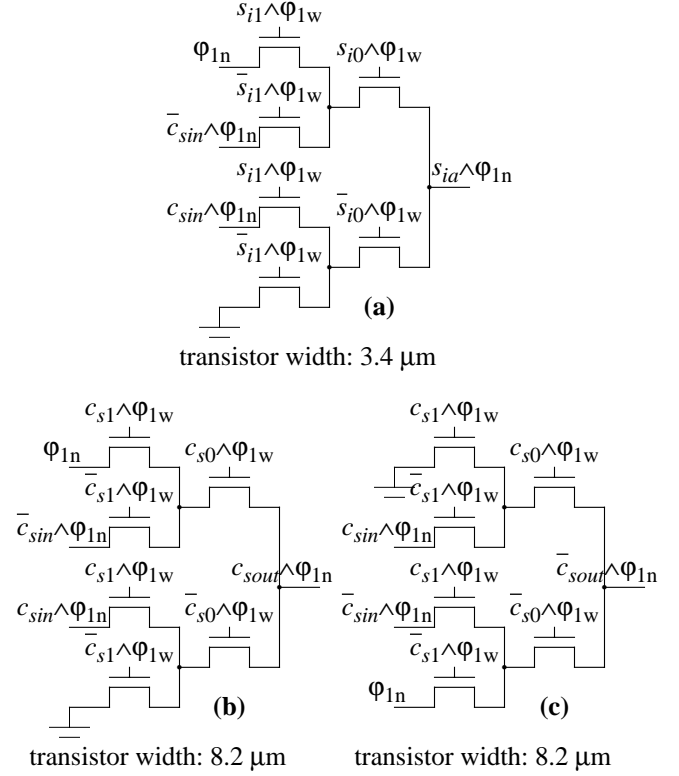


Figure 8: Schematics for the multiplexers that generate the final sum bits (a) and the stage carry-out in dual-rail form (b and c).

pulsed signal output. A weak pFET pull-up was added in the dynamic latch node. This pFET is driven by the output of the first inverter. All bootstrap and clamp transistors (M_2 and M_3 , respectively, in Fig. 1) except those used for the E-R latch that holds C_{in} are $8.6 \mu\text{m}$ and $2.2 \mu\text{m}$, respectively. The bootstrap and clamp transistors are significantly larger for the E-R latch that holds C_{in} ($23.6 \mu\text{m}$ and $14.8 \mu\text{m}$, respectively), since this latch drives the carry chain of the four final stages. For RCPL, the clamp transistor could be as small as the one in the original E-R latch since the entire carry-chain capacitance is charged and discharged by the one bootstrap transistor. However, when the adder operates with the two-phase clocking scheme, the charge that is trapped in the carry chain is discharged by the clamp transistor.

4. Simulation Results

The 16-bit adder was simulated in HSPICE with the level 39 MOSFET models for the HP CMOS14B process. Two sets of simulations were carried out for different clocking schemes (Fig. 9). For the first set, the adder operated from the two non-overlapping phases (Fig. 9a). For the second set, the adder operated from four phases (Fig. 9b).

For a fixed cycle time T , the switching time and the width for the various cases are calculated as shown in Fig. 9. For the same cycle time T , the switching time for the pulses of the two-phase clocking is longer than the other two cases.

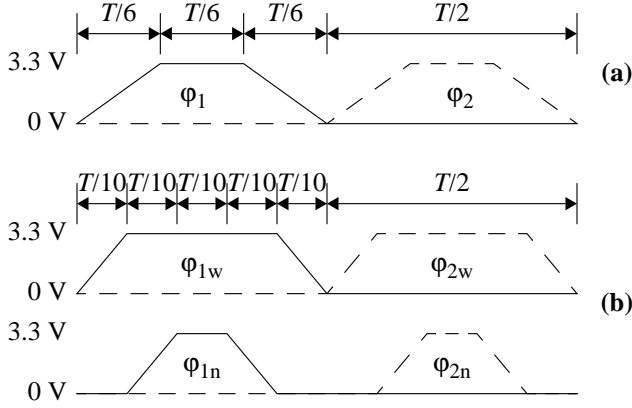


Figure 9: Timing diagram of clocking schemes used for HSPICE simulations of the adder.

In other words, more time per clock cycle is available for adiabatic charging. In all cases, the pulses swing from 0 V to 3.3 V. The dc supply voltage used for E-R latches was set to 2.3 V. The isolation voltage V_{iso} was set to 3.5 V.

The HSPICE simulation procedure was performed as follows: First, for each case, the highest obtainable speed was found. This was done by simulating the worst-case addition in terms of cycle time, which happens when all bits of one operand are zero, all bits of the other operand are one, and C_{in} toggles each every cycle. The shortest cycle time was 10 ns for the two-phase clocking scheme and 11 ns for the four-phase clocking scheme. Subsequently, for each clocking scheme, two sets of simulations were done. For each set, the cycle time was varied from 100 ns to the minimum cycle time obtained from the previous step. One set of simulations included the worst-case addition in terms of required switching activity in the transistor-chain nodes, which happens when all operand bits (i.e., A and B) and C_{in} switch from zero to one. The other set of simulations included the best-case addition in terms of required switching activity in the transistor-chain nodes. However, this addition requires maximum switching activity for the four-phase clocking scheme. This happens when all inputs are held at one. In general, when all inputs are held constant there is no switching activity in the transistor-chain nodes when the two-phase clocking scheme is used. For all cases, the energy dissipated in transistor gates and transistor chains of the PTL gates were simulated. It was assumed that all returned energy was recovered.

A waveform plot from the HSPICE simulation is shown in Fig. 10. The plot shows the switching activity of node C_{out} for the worst-case addition. The top waveforms are from the adder operating with the two-phase clocking scheme. Node C_{out} is charged following phase ϕ_1 . Charge is trapped and, during the next cycle, it is dumped to ground. When the adder is operated with four phases, node C_{out} is charged and discharged following phase ϕ_{1n} . All energy injected to C_{out} is recovered at the end of the cycle. During the next cycle, C_{out} remains at 0 V.

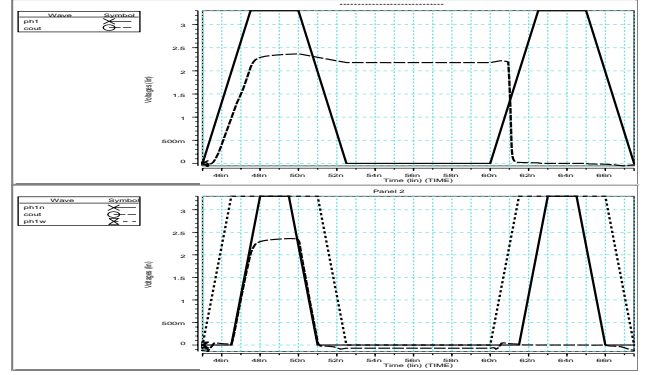


Figure 10: HSPICE waveforms when the adder is operated from two phases (top) and four phases (bottom).

The simulation results for the addition that requires all transistor-chain nodes to switch are presented (Fig. 11). When the adder is operated with two phases, the energy dissipation in the transistor chains is two to three times higher than when the adder is operated with four phases. Additionally, the transistor-chain energy dissipation does not scale as well for increasing switching time when the adder is operated with two phases, since most of the energy dissipation in the two-phase clocking scheme is due to the trapped charge, which is independent of the switching time. The lowest energy dissipation in the transistor chains happens when the adder is operated with four phases. The energy dissipation in the transistor gates is lower when the two-phase clocking scheme is used because, for the same cycle time, the switching time is longer than when the four-phase clocking scheme is used.

The simulation results for the addition in which all inputs are held constant are shown in Fig. 12. For all three cases, the energy dissipated in the transistor gates is approximately the same as was simulated for the previous addition.

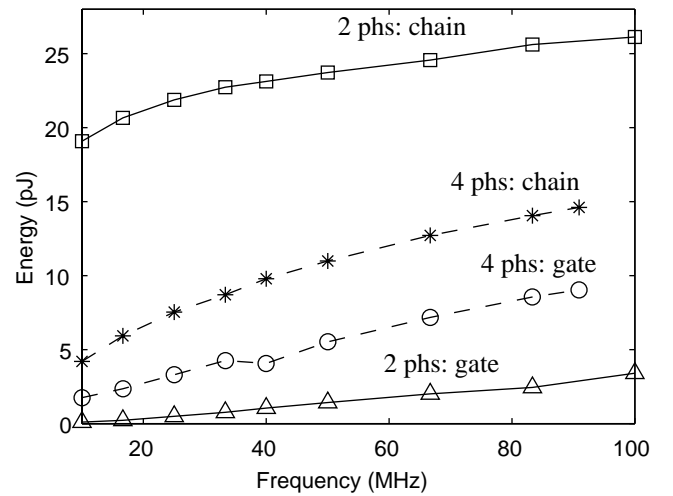


Figure 11: HSPICE simulation results for worst-case addition in terms of required switching activity for transistor-chain nodes.

The dissipation in the transistor-chain nodes when two phases are used is almost insignificant compared to the dissipation for the previous addition, since all energy trapped in the transistor-chain nodes can be reused for successive additions. Only the input nodes to the transistor chains must switch every cycle. However, this is a small amount of capacitance compared to the transistor-chain capacitance, and it can also be recovered in its entirety. For the four-phase clocking scheme, the transistor-node capacitance must be switched every cycle, since it is unconditionally recovered at the end of each cycle. The dissipation is lower than in the previous addition, because some energy is trapped inside the E-R latches that hold the stage adder outputs and the sum bits, which can be reused for successive additions when the stored data does not change.

The simulation results show clearly that the benefit of recovering the energy of the transistor-chain nodes strongly depends on the expected switching activity of these nodes. For applications in which there is high switching activity for the transistor-chain nodes, recovering their energy results in energy savings of up to 75% for these nodes. However, using nested phases decreases the maximum available switching time, which decreases energy savings for driving the transistor gates.

5. Conclusions

In this paper, we described RCPL for energy-recovery CMOS. In RCPL, circuit energies from all combinational logic nodes are recovered. Only the E-R latch nodes are powered from a dc supply source. As is the case with all energy-recovery logic families, the switching activity of clock-powered nodes occurs at the maximum rate. Simulation results indicate that the efficiency of this energy-recovery approach depends on the switching activity of the applied circuit. For circuits with low switching activity, it is more efficient not to recover the circuit energy because it can be locally reused. For circuits with high switching activity,

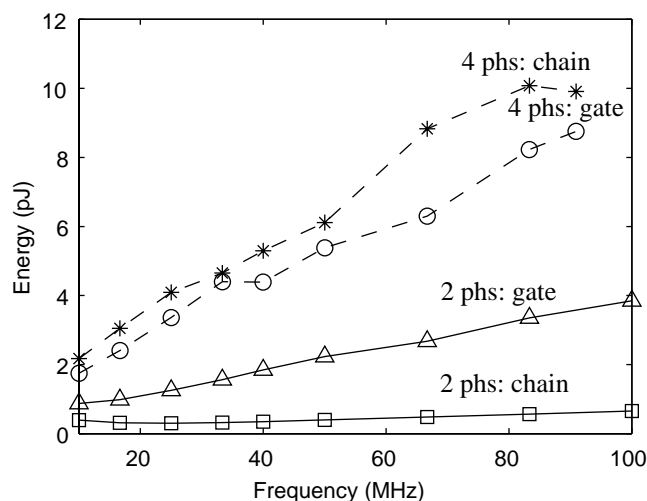


Figure 12: HSPICE simulation results for best-case addition in terms of required switching activity for transistor-chain nodes.

RCPL can result in significantly higher energy savings. Reducing the switching activity of energy-recovery CMOS circuits is possible. However, the operation and efficiency of these circuits depends on the clock driver which typically is a resonant circuit. Reducing the switching activity of clock-powered nodes can cause the variation of the clock load capacitance to increase because it is data dependent. To date, large cycle-to-cycle capacitance variations have been detrimental to the performance of the resonant clock driver circuits [2].

Acknowledgments

The research described in this paper was supported by ARPA contracts DABT63-92-C0052 and DAAL01-95-K3528.

References

- [1] W. Athas, L. Svensson, J. Koller, N. Tzartzanis, E. Chou, *Low-Power Digital Systems Based on Adiabatic-Switching Principles*, IEEE Transactions on VLSI Systems, pp. 398-407, Dec. 1994.
- [2] W. Athas, N. Tzartzanis, L. Svensson, L. Peterson, A. *Low-Power Microprocessor Based on Resonant Energy*, IEEE Jnl. of Solid-State Circuits, vol. 32, pp. 1693-1701, Nov. 1997.
- [3] W.C. Athas, L. "J." Svensson, N. Tzartzanis, A *Resonant Signal Driver For Two-Phase, Almost-Non-Overlapping Clocks*, Proc. of the 1996 International Symposium on Circuits and Systems, Atlanta, GA, May 12-15, 1996.
- [4] L."J." Svensson, J.G. Koller, *Driving a Capacitive Load without Dissipating fCV^2* , Proc. of the 1994 Symposium on Low Power Electronics, pp. 100-101, San Diego, CA, Oct. 10-11, 1994.
- [5] S.G. Younis, T.F. Knight, *Asymptotically Zero Energy Split-Level Charge Recovery Logic*, Proc. of the 1994 International Workshop on Low-Power Design, pp. 177-182, Napa Valley, CA, Apr. 24-27, 1994.
- [6] N. Tzartzanis, W. Athas, *Clock-Powered Logic for a 50 MHz Low-Power Datapath*, in ISSCC Digest of Technical Papers, pp. 338-339, San Francisco, CA, Feb. 6-8, 1997.
- [7] S. Hall, *An Electroid Switching Model for Reversible Computer Architectures*, Proc. of the 1992 Workshop on Physics and Computation, PhysComp'92, Dallas, TX, Oct. 2-4, 1992.
- [8] L.A. Glasser, D.W. Dobberpuhl, *The Design and Analysis of VLSI Circuits*, Addison-Wesley, Reading, Ma., 1985.