Reversible Logic Issues in Adiabatic CMOS

W.C. Athas & L."J." Svensson
Exploratory Design Group
University of Southern California - Information Sciences Institute
Marina del Rey, CA 90292-6695
{athas,svensson}@isi.edu

Abstract

Power dissipation in CMOS circuits has become increasingly important for the design of portable, embedded, and high-performance computing systems. Our VLSI research group at the USC Information Sciences Institute has investigated a novel form of energy conserving logic suitable for CMOS. Through small chip-building experiments, we have demonstrated the low-power operation of simple logic functions. These chips have used logical reversibility on a small, sometimes trivial, scale to achieve their low-power operation. In moving towards more complex functions, the role of reversibility will increase. This paper addresses two problem areas that we have found to be crucial to successfully realizing low-power operation of CMOS chips using reversible logic techniques. The first area is the energy-efficient design of the combined power supply and clock generator. The second is the logical overhead needed to support reversible logic functions. The first problem area, though formidable, seems amenable to systematic approaches. Significant inroads have been made towards finding practical, efficient solutions. The second, however, appears to be by far the more difficult hurdle to overcome if reversible logic is to become an attractive approach for reducing power dissipation in CMOS.

1: Introduction

Adiabatic switching is a new, systematic approach to reducing power dissipation in digital computer circuits by recycling circuit energies rather than dissipating them as heat. This approach has particular appeal to device technologies such as CMOS, in which the controlling and the controlled charges are kept physically separate. Since charges of opposite polarity never come into physical contact, there are no dissipative losses due to recombination.

A simple theory and practice for implementing switching circuits based on *adiabatic charging* has been developed and demonstrated through a number of small chip building experiments. Our experiments with adiabatic switching indicate that significant amounts of circuit energy can be conserved in CMOS circuits [1]; and that when these circuits are operated reversibly, they can potentially be a practical approach to low-power computing.

Seitz, et al. [2], defined the following relationship between switching energy and transition time when a constant current, I, is used to charge a capacitance, C, to a voltage, V, through a resistance, R:

$$E_{\text{diss}} = P \cdot T = I^2 R \cdot T = \left(\frac{CV}{T}\right)^2 RT = \left(\frac{RC}{T}\right) CV^2 \quad (1)$$

P is the instantaneous power dissipation and T is the charging time. Increasing the charging time clearly decreases the dissipation. For our research with CMOS, we refer to this formula as the adiabatic-charging principle. In contrast, the corresponding "conventional" charging of a capacitance (as used in most CMOS circuits) causes the energy $CV^2/2$ to be dissipated, regardless of the time allowed for the charging.

To relate this formula to logic design in CMOS, we can construct a fully-restoring transmission gate (T-gate) from an n-channel and a p-channel device connected in parallel. With the gate electrodes driven by complementary voltages (V_{dd} and 0, respectively), to a first approximation the on-resistance of a T-gate can be modelled as [3]:

$$R_{tg} = \frac{L^2}{\mu_n C_n (V_{dd} - 2V_{\text{th}})} \quad (2)$$

L is the channel length, μ_n is the mobility of the n-channel device, V_{th} is the threshold voltage, and C_n is the gate capacitance of the n-channel device. The formula assumes that the drain-to-source voltages of the devices are small (the triode region) and that the channel widths are scaled according to carrier mobility. The parameters L, V_{th}, and μ_n are defined by the CMOS process, and V_{dd} is usually

0-8186-6715-X/94 $4.00

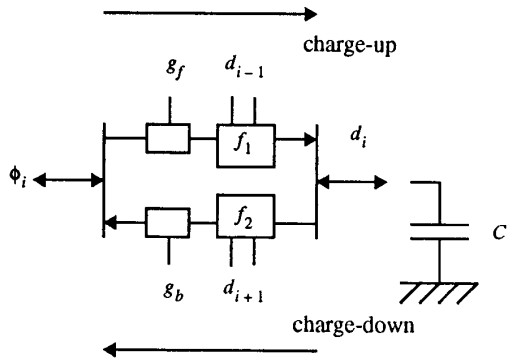

Figure 1: A gate structure suitable for reversible logic gates. The arrows show the direction of charge and energy flow.

fixed by system-wide concerns. Thus, for a specific implementation, these parameters can all be considered to be constant, and the on-resistance of a T-gate can be simply modelled as inversely proportional to gate capacitance.

The CMOS T-gate on-resistance given by Equation 2 is independent of the channel voltage. Hence, the expression may be used in Seitz's formula of Equation 1. These two equations are a rudimentary basis for designing and building a simple family of adiabatic logic circuits. By definition of the term "adiabatic," switching is to occur without losing or generating heat. As is the typical usage of the term in thermodynamics, fully adiabatic operation is an ideal condition that is asymptotically approached as switching speed is slowed down.

2: A scheme for applying adiabatic switching

There are two major challenges in realizing the benefits of adiabatic switching for reducing power dissipation. The first problem is the energy-efficient generation of the voltage and current ramps used for clocking and power delivery. This problem is analyzed in detail in Section 3. The second problem is how to perform useful computations with high utilization of VLSI chip resources while also meeting the logical and physical requirements for energy conservation. Towards this goal, many reversible schemes have been proposed [4, 5, 6]; our variant is described in this section. In Section 4, we examine in detail the logical overhead imposed by this scheme.

Figure 1 depicts the basic logic organization for computing digital functions that we have used for designing and building adiabatic-switching circuits. With this scheme, a clock signal (labelled ϕ_i) is used to both power and sequence the circuit functions. The output signal of the gate structure (labelled d_i) drives a signal capacitance. The path between the clock and the signal capacitance is partitioned into a charge-up path and a charge-down path. The unlabelled switch boxes are each a single T-gate. The labelled boxes compute functions as a series-parallel network of one or more T-gates. All AND-OR Boolean expressions can be implemented via such series-parallel connections of switches [7]. All functions are computed in dual-rail form so that the NOT function is trivially provided for. The inputs g_f and g_b and the T-gates that they control "guard" against the possibility of non-adiabatic energy flow when the data value and the clock value are both at valid but different logic levels.

Reversible operation is achieved by setting the data inputs to the function f_1 (i.e., d_{i-1}), asserting g_f, and only then delivering energy to the output d_i by ramping up ϕ_i. If the switch network evaluates to true, the output will be driven high; otherwise, it will stay low. At this point, the operation of the circuit could be trivially reversed by ramping down ϕ_i while the data inputs remain valid. Logic functions may be composed by using the charge-up path of the circuit in Figure 1 as a single logic stage and then connecting the outputs of one stage to the inputs of the next in a cascaded fashion [4]. This style is not practical when the number of cascaded stage cells is more than a few levels deep, because the very first inputs

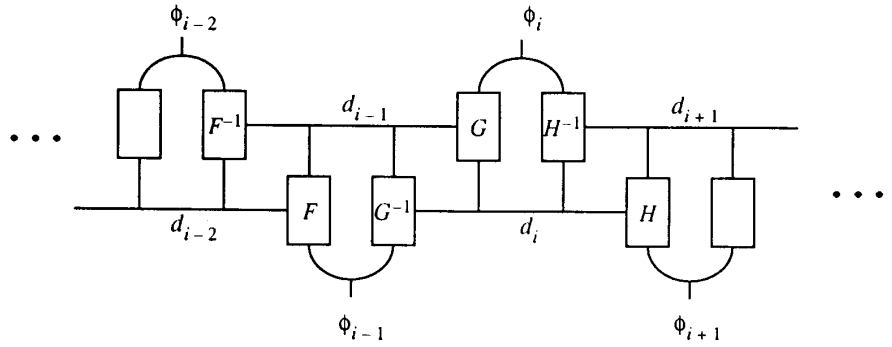

Figure 2: Interconnection of logic gates shown in Figure 1 in a pipeline. The guard switches have been omitted for simplicity.

must remain valid until after the very last output is allowed to go invalid. For n levels of cascading, the activity factor for each stage would scale as $1/n$, and the circuit performance would suffer accordingly.

It is possible to pipeline data through the logic stages and allow the inputs of an earlier stage to change before the final output value is ready by computing the inverse of each function as data is passed from stage to stage. In Figure 1, this action is performed by the f_2 switch together with g_b and its T-gate. When the output d_i has been computed, it is held constant while the next stage is powered up. The outputs of this next stage are then fed back as inputs to f_2, so that when ϕ_i is ramped down, the return path for the energy flow is ensured. With this style of operation, logic stages can be interconnected in a pipeline, as shown in Figure 2. Each signal is computed by two functions: one taking its inputs from the preceding stage of computation and one using data from the following stage.

When the pipeline is operated, each data bit is cycled through four states: *set*, *hold*, *reset*, and *idle* [8]. *Set* and *reset* correspond to the delivery and return of the signal energy, while *hold* overlaps with the *set* state of the following stage and the *reset* state of the preceding stage. The *idle* state is included for symmetry as a place-holder between the *reset* and *set* states.

The operation of the reversible pipeline results in a paradox, in that it is assumed that for every stage, there will be a *next* stage from which its discharge can be controlled. In practice, there will usually be a final output stage in which the operation of the circuit is no longer reversible. For this eventuality, there is an unavoidable CV^2 dissipation for erasing the final output [4]. This dissipation is proportional to the number of output signals rather than the total number of signal nodes in the circuit. Adiabatic principles can be applied to reduce this dissipation per output signal from CV_{dd}^2 to CV_{th}^2 [3].

The four states that each data bit must cycle through can be associated with phases of clock signals. By overlapping several clocks and using them also as the g_f and g_b inputs, it is possible to build a fully static pipeline structure using eight clock phases [9]. In some cases, it is possible to use fewer phases, e.g., four or six, depending on the function and the need for static logic.

This pipeline structure has been demonstrated in three chip designs: a shift register, a bit-level pipelined parity generator, and a pipelined adder. The chips were tested with conventional power supplies and also with special prototype power supplies that were able to recover and reuse the signal energies from the test chips. The power supplies are further discussed in Section 3.

The experiments showed that it is possible to make adiabatic computing and reversible computing work in standard CMOS processes. Low-power operation can be achieved without reducing supply voltages or modifying the device packaging technology. However, for this approach to be practical, there are critical issues to be resolved: efficient power control, overhead in circuit layout, and the overhead and complexity of reversibility at the logic design level. The efficiency of the solutions for these problems must be compared not against themselves but rather against the best solution which can be achieved by the more well-known approaches to reducing power dissipation, most notably by scaling down the supply voltage of a conventional CMOS logic circuit [11].

3: Controlling the Power Supply

The constant-current case for energy delivery described by Equation 1 requires that the power supply generate linear voltage ramps. Unfortunately, we do not know how to deliver and recover energy at constant current without dissipating more energy than is recycled in the "regenerator." However, it is possible to efficiently deliver charge via a sinusoidal waveform using an inductor. Theoretically, the energy dissipation is then increased to:

$$E_{\text{diss}} = \xi \frac{RC}{T} CV^2, \ \xi = \frac{\pi^2}{8} \tag{3}$$

R includes the resistances due to the T-gates in the logic circuit and the resistance in the path of the current in the power supply.

Connecting an inductor to one or more capacitive loads through resistive MOSFET channels forms a resonant circuit. The voltage waveforms which develop at the loads will be sinusoidal. With the outputs following a sine-wave shape, they are constantly switching and only instantaneously available for sampling at their full logic value. This switching mode precludes the set-hold-reset paradigm described in Section 2 for the reversible pipeline constructions.

To solve this problem, a switch may be inserted into the resonant circuit so that the sinusoidal outputs can be stopped at maximum and minimum voltage [1]. The circuit of Figure 3 can be used for this purpose, is representative of the problems to be encountered in general, and is

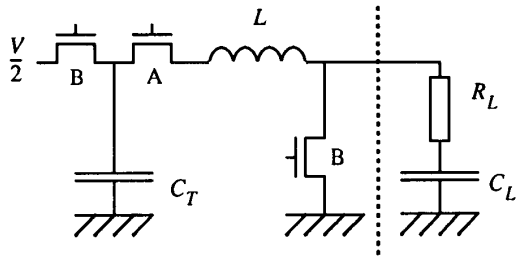

Figure 3: Circuit for inductive power delivery to RC-type load. Power supply to the left of the dotted line.

straightforward to analyze. The "tank" capacitance, C_T, is much larger than the load capacitance and is held at half the desired maximum output voltage. The load resistance, the "A" switch, the capacitances and the inductance together form an underdamped RLC circuit. The voltage on the load capacitance is an exponentially damped sinusoid with a maximum close to V_{dd}. The "B" switches are used to restore voltage levels on the tank capacitor and on the load after a charge-discharge cycle.

The insertion of the "A" switch increases the total resistance in the RLC circuit, and thereby the dissipation per charge-discharge cycle. Assume that the switch is driven by a conventional driver, i.e., all the energy stored on the gate capacitance is dissipated every cycle. For small drain-to-source voltages and with a fixed source voltage of $V_{dd}/2$, the on-resistance of the switch can be modelled as [1]:

$$R_A = \frac{2L^2}{\mu_n C_A (V_{dd} - 2V_{th})} \tag{4}$$

When the "A" switch is made wider, its gate capacitance will increase, reducing the on-resistance and thereby the dissipation in its channel; on the other hand, the increased gate capacitance requires more energy to control the switch. For given values of V_{dd}, C_L, and T, there is an *optimum* switch device width that minimizes the total dissipation. The optimum width, as well as the corresponding minimum total energy, are proportional to the *square root* of the inverse of the charging time [1].

It is in principle possible to decrease the power supply dissipation by using a similar inductor circuit to drive the gate of the switch of the first circuit. In general, it can be shown that for a cascading of N such circuits for which

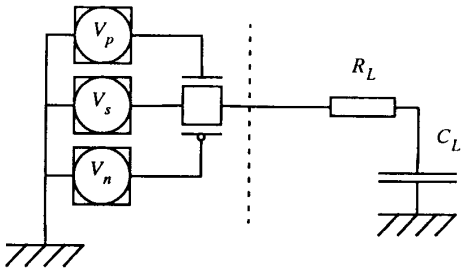

Figure 4: Circuit for efficient generation of clock waveforms for adiabatic switching.

the parasitic capacitances are neglected, the total energy dissipation depends on the charging time as:

$$E_{min} \sim T^{-(1-2^{-N})} \tag{5}$$

Thus, the dependence asymptotically approaches T^{-1} with increasing N. It is, however, unlikely that this asymptote would be approached in practice, since parasitic capacitances have a relatively large effect on circuit operation.

It is also possible to decrease power supply dissipation as T^{-1} by using a multiplicity of resonant circuits in which the gate and source inputs of a T-gate switch are all driven by sinusoidal signals. A capacitance can be driven with a sinusoidal signal with high efficiency, using any of a variety of well-known techniques for LC oscillator design [10]. Figure 4 shows a conceptual circuit exemplifying this technique; Figure 5 shows the corresponding waveforms of the sine-wave generators and the output. The frequency of the gate-drive waveform is two-thirds

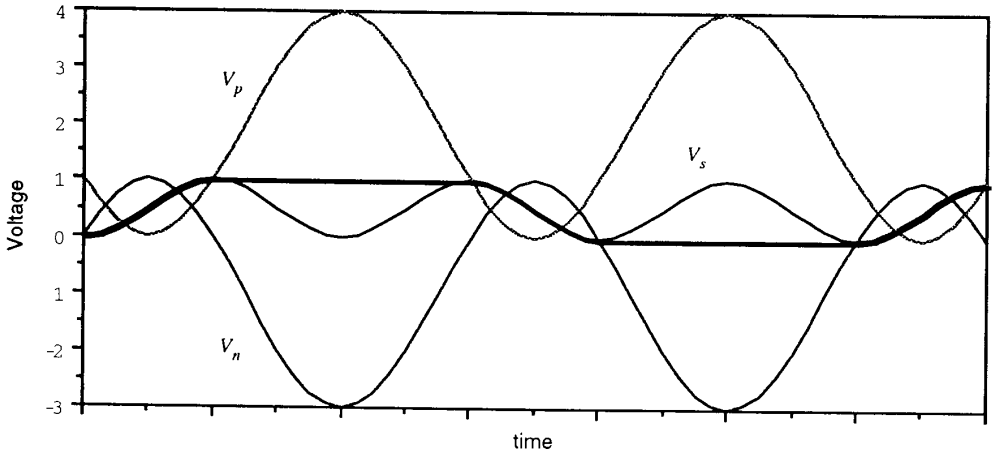

Figure 5: The sine waves used as gate and source inputs for the circuit shown in Figure 4, and the output (bold).

that of the source drive signal, and the voltage swing on the gates is four times that of the output voltage swing. The large positive-going sine wave drives the gate of the p-channel device and the large negative-going sine wave drives the n-channel device. By carefully phasing the source and gate signals, then during the interval in which the gate of the n-channel device is positive, the positive or negative swing on the source input will initiate a charge-up (set) or charge-down (reset) action, respectively. The gate of the p-channel device operates in an analogous but complementary fashion. This approach has been verified by SPICE simulations, using zero-impedance voltage sources to provide the sine waves. Although there are some significant problems that must be solved for the scheme to be practical, it does in principle circumvent the asymptotic limitation of the other approaches.

An important issue in analyzing the efficiency is the energy that must be supplied to sustain the oscillations on the source and gate inputs. For the gate-drive circuits, the oscillations can be sustained without placing a switching device in the series-path of the inductor and capacitive load. Rather, the switching devices are placed in series with the power or ground paths of the oscillator so that energy can be added to the oscillator at the same rate at which it is dissipated by the resistance in the inductor and the load circuit of the oscillator.

4: The overhead of reversibility

Reversible computation, as employed in the logical organization of Figure 2, requires that all logical computation steps be carried out twice: once in the forward and once in the backward direction. Furthermore, intermediate results must often be maintained for some number of future steps until the "uncomputation" is ready. The number of time steps between when a result is computed and when it can be uncomputed depends on the logic function. The storing of the intermediate results is not a fundamental problem, since it can be done with reversible latches, as has been successfully shown with our previous chips. However, a large overhead in energy and circuit area can be associated with these latches. An example that clearly demonstrates the limitation imposed by reversibility is the highly-pipelined adder.

Figure 6 shows a bit-pipelined, reversible adder. A three-bit version is shown, but the extension to a larger version is straightforward. Those elements of the adder that would be needed in a conventional, non-reversible adder are drawn shaded. Clearly, the reversible version contains many elements beyond those used in a conventional circuit. In addition to the "backwards" adder, the carry chain mandates a large number of bit delay elements (which preserve information until it may be uncomputed) forming a "wedge" between the adders. The number of bit delay elements in the wedge grows as the square of the number of bits of the adder. For a 16- or 32-bit adder, the bit delay elements will occupy most of the chip area and produce most of the dissipation.

Energy dissipation is unavoidable when information irrevocably leaves the system [6]. We may, of course, decide to throw away information not only at the periphery of the system, but also at other places in the computation. Typically, this will allow us to remove bit delay elements, since at least the thrown-away signal will not be kept for later uncomputation. There is then a trade-off between throwing away information and keeping it around for later uncomputing. The cost of the former depends on the exact circuit used for the non-adiabatic discharge, while the cost of the latter depends on how long the information has to be kept. The comparison of these costs indicates when reversibility might be a practical technique.

For a first-order analysis, we may assume that the path resistance, R, and the driven capacitance, C, of all gates and bit delay elements are the same, and that all nodes swing between 0 and V_{dd}. We compare three ways of discharging a bit capacitance. The first case is a completely dissipative discharge through a conventional switch connected to ground; all the energy stored on the capacitance is discharged, so the cost is:

$$E_{\text{direct}} = \frac{1}{2} C V_{dd}^2 \tag{6}$$

Second, we may use the information in the bit itself to control its discharge down to V_{th} [3], and completely dissipate only the remaining energy. With the simplifying assumption that the on-resistance of the discharge path can be held constant during the discharge, the cost may be approximated as:

$$E_{\text{partial}} = \xi \frac{RC}{T} C (V_{dd} - V_{\text{th}})^2 + \frac{1}{2} C V_{\text{th}}^2 \tag{7}$$

This expression underestimates the dissipation somewhat, since the path resistance is likely to rise when the controlling voltage falls towards V_{th}.

Third, assume that a completely adiabatic discharge of the output bit using reversible logic entails the additional dissipation of fully cycling the output of M bit delay elements. We get:

$$\begin{aligned} E_{\text{reversible}} &= M \cdot 2\xi \frac{RC}{T} C V_{dd}^2 + \xi \frac{RC}{T} C V_{dd}^2 = \\ &= (2M + 1)\, \xi \frac{RC}{T} C V_{dd}^2 \end{aligned} \tag{8}$$

We first compare the first and third cases. For the reversible case to be superior to the conventional case,

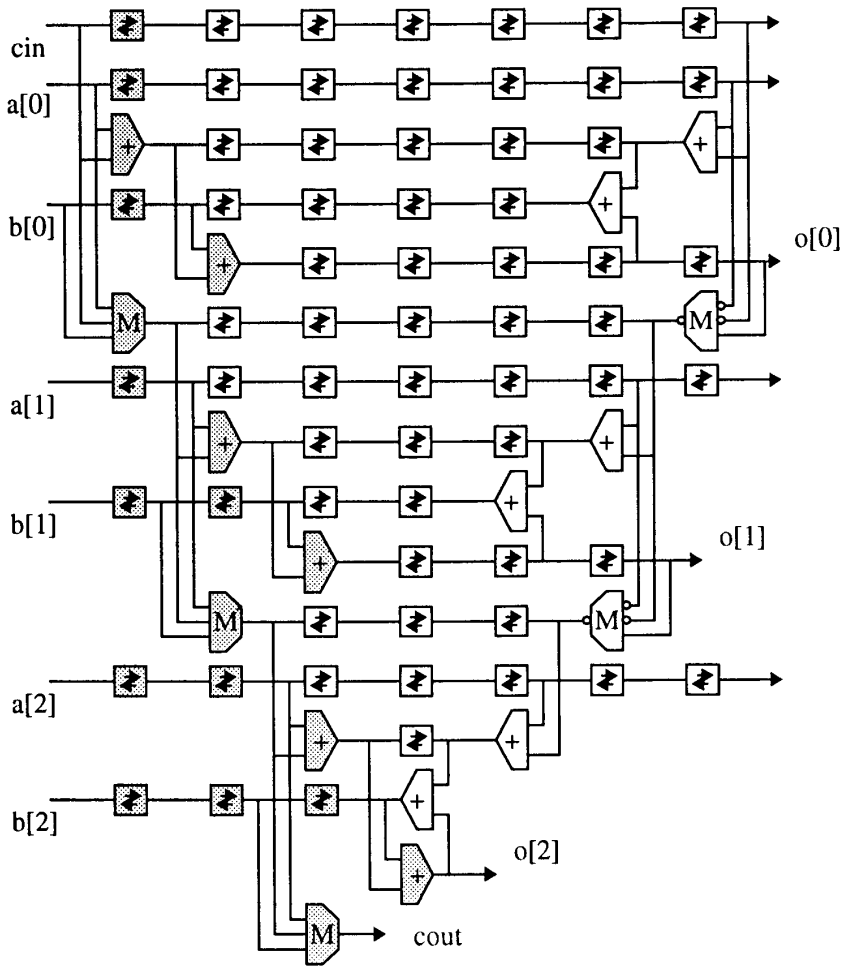

Figure 6: A bit-pipelined, reversible adder, built from EXOR gates, majority gates, and bidirectional bit delay elements. Only the shaded elements would be needed in a conventional adder with an equal amount of pipelining.

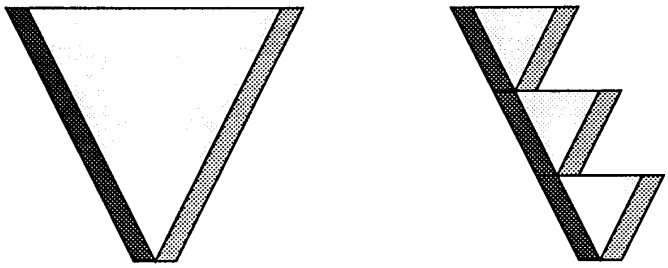

Figure 7: The amounts of bit delay elements for intermediate signals for a fully reversed adder and for a partially reversed adder. The forward adder is shaded dark grey, the backward adder is shaded medium grey, and the delay wedge is light grey. In general, splitting an adder into M sub-adders brings a saving in the number of wedge delay elements by a factor of M.

$E_{direct} > E_{reversible}$ must hold. With sinusoidal ramps, the shape factor is given by $\xi = \pi^2/8 \approx 5/4$, and:

$$\frac{1}{2}CV_{dd}^2 > (2M+1)\,\xi\frac{RC}{T}CV_{dd}^2 \tag{9}$$

$$T > \left(5M + \frac{5}{2}\right)RC \tag{10}$$

As an example, this means that unless $T > 52.5 \cdot RC$, to throw away a bit of information results in decreased dissipation if it allows us to discard ten bit delay elements ($M = 10$).

To compare the second and third cases, we relate V_{dd} to V_{th} by observing that there is an optimally energy-efficient supply voltage for T-gate-based adiabatic circuits, which can be approximated as four times the threshold voltage [11]. Then, with $\xi = 5/4$:

$$E_{partial} = \xi\frac{RC}{T}C\left(V_{dd} - \frac{V_{dd}}{4}\right)^2 + \frac{1}{2}C\left(\frac{V_{dd}}{4}\right)^2 = \\ = \left(\frac{45}{64}\frac{RC}{T} + \frac{1}{32}\right)CV_{dd}^2 \tag{11}$$

A fully-reversible case with M bit delay elements dissipates less energy if $E_{delay} < E_{partial}$, or:

$$\frac{5}{4}(2M+1)\frac{RC}{T}CV_{dd}^2 < \left(\frac{45}{64}\frac{RC}{T} + \frac{1}{32}\right)CV_{dd}^2 \tag{12}$$

$$T > \left(80M - \frac{35}{16}\right)RC \tag{13}$$

Here, a ten-bit delay overhead ($M = 10$) would require the charging period to be almost $800 \cdot RC$. In both these cases, reversibility is attractive only if the charging time can be increased to huge values, regardless of throughput constraints and implementation cost. This scenario places an extremely high premium on reducing the energy per operation and is not likely to be used in practical applications, when there are limits for the acceptable throughput and hardware amount.

Since the delay wedge in Figure 5 is caused by the carry chain, it may be shrunk significantly by introducing irreversible carry computations. The carry-in signal of Figure 5 is kept for uncomputation of the addition, but no means for rescuing its own signal energy has been included. By similarly throwing away the carry-in signal to selected bit adders, we can get rid of a large number of bit delay elements. Figure 7 shows the wedge reduction resulting from the introduction of irreversible carry computations. The exact number of delay elements removed depends somewhat on the circuitry surrounding the adder, but going from a wedge height of 4 bits down to 2 will let us remove at least 20 delays, for one thrown-away bit. According to Equation 10, this results in lower dissipation unless $T > 100 \cdot RC$. Equation 13 indicates a minimum charging time of almost $1600 \cdot RC$, but since the estimate of Equation 7 is likely to be low, 1600 is on the high side. The exact value for this case depends on the circuit solution for the partially-adiabatic discharge.

In addition to the dissipation overhead, the reversible circuit is also considerably larger than its conventional counterpart. To estimate the hardware overhead, a conventional version of the three-bit adder circuit was laid out by the same designer, using the same design rules. The reversible adder requires 32 times as much space and 20 times as many devices as the conventional one. The layout of the reversible adder was also extracted and SPICE-simulated to get an estimate of the RC constant used in the calculations in this section; the use of dual-rail logic and guard switches as in Figure 1 and the routing capacitances, etc., force this constant beyond $200 \cdot R_{tg}C_n$ (which equals $100 \cdot \tau$ in the well-known tau model [12]).

In summary, for a fully-reversible, adiabatic, bit-pipelined four-bit adder to exhibit lower dissipation than two interconnected two-bit adders with the energy of the non-reversed carry bit completely dissipated, the switching time has to be increased to at least $10^4\tau$. If partially-adiabatic techniques are used to recover some of the energy of the non-reversed carry signal, the crossover point is pushed out well beyond $10^5\tau$. This should be compared to a conventional bit-level-pipelined adder circuit, which can operate with a charging time on the order of 10τ. The dissipation models used in this analysis are admittedly crude, but they are unlikely to be off by orders of magnitude. It therefore seems clear that *very* slow switching must be used to justify fully reversing even a circuit as small as a pipelined four-bit adder.

5: Conclusions

In this paper, we have investigated two problems that we believe must be solved if useful levels of power reduction are to be gained from applying reversible logic techniques in CMOS. Adiabatic switching has been previously demonstrated as capable of reducing energy dissipation in CMOS by recycling circuit energies. Starting from the principles of adiabatic switching, we analyzed a rudimentary logic style suitable for reversible computing.

The problems of efficiently combining the tasks of power delivery and recovery with clock generation, although formidable, seems to yield to systematic solution. A "brute force" solution of using conventional circuits to drive the MOSFET gates of the power supply switches limits energy scaling to $T^{-1/2}$ rather than T^{-1}. Replacing the conventional gate drive will improve energy scaling to $T^{-3/4}$, and recursive application of this scheme

will asymptotically approach the ideal T^{-1} result. Furthermore, it may be possible to use all resonant circuits which would directly achieve the T^{-1} scaling. One such approach was described and simulated; although there are many secondary problems to overcome, no fundamental problems are known at this time. Also, because the power supply is separate from the chip, the switch elements in the power supply are not necessarily CMOS devices. Hence, it may be beneficial to select devices that cannot be cost-effectively integrated as CMOS ones, but offer superior performance in the critical power supply circuit.

On the other hand, we believe the overhead needed for supporting reversible logic will severely limit its use in CMOS circuits. The example of the adder circuit underscores many of the problems. The application of fine-grain pipelining to improve performance results in many more signals that need to be individually reversed. As with the carry bits of the adder circuit, it may be necessary to sustain these bits in the computation for many clock cycles until they can be properly reversed. The question then becomes: when is the energy of a signal sufficient to make it worth the overhead of recovering it versus just dissipating it? For "large" signals, i.e., those that are widely distributed and therefore drive large capacitances, there is evidence that the energies will be worth the effort of recovery. In addition, the logic driving these signals is often trivially reversible.

Niche applications may possibly appear where the ultimately low dissipation at very low speed, with maintained signal energies, will be attractive, and superior reversible CMOS logic design approaches that improve the constant overhead factors may appear, but we conjecture that generally useful reversible logic must be built from other types of active elements.

6: Acknowledgments

The authors wish to thank Dr. Jeffrey G. Koller for numerous helpful discussion on the fundamental limits of the physics of computation, and Prof. Sven Mattisson of Lund University for many helpful discussion about the subtleties of transistor physics. We are also grateful for the valuable work of Nestoras Tzartzanis and Eric Chou.

The work described in this paper was supported by the Advanced Research Projects Agency, contract DABT63-92-C0052.

7: References

[1] W.C. Athas, J.G. Koller, L."J." Svensson. An energy-efficient CMOS line driver using adiabatic switching. *Proc. of the Fourth Great Lakes Symposium on VLSI Design*, IEEE Press, pp.196-199, 1994.

[2] C. L. Seitz, A.H. Frey, S. Mattisson, S.D. Rabin, D.A. Speck, J.A. van de Snepscheut. Hot-clock NMOS. In *Proc. of the 1985 Chapel Hill Conference on VLSI*, 1985.

[3] J.G. Koller, W.C. Athas. Adiabatic switching, low energy computing, and the physics of storing and erasing information. In *Proc. of the Workshop on Physics and Computation*, PhysComp '92, Oct. 1992; IEEE Press, 1993.

[4] J.S. Hall. An electroid switching model for reversible computer architectures. *Proc. ICCI'92, 4th International Conf. on Computing and Information*, 1992.

[5] R.C. Merkle. Reversible electronics logic using switches. *Nanotechnology*, No. 4, pp. 21-40, 1993.

[6] S.G. Younis, T.F. Knight. Practical implementation of charge recovery asymptotically zero power CMOS. *Proc. of the 1993 Symposium on Integrated Systems*. MIT Press, pp. 234-250, 1993.

[7] Z. Kohavi. *Switching and finite automata theory, 2nd ed.* Computer Science Press, New York, 1978.

[8] W.C. Athas, L."J." Svensson, J.G. Koller, N. Tzartzanis, Y-C Chou. A framework for practical low-power digital CMOS systems using adiabatic switching principles. *Int'l Workshop on Low-Power Design*, pp. 189-194, Apr. 1994.

[9] J.G. Koller, W.C. Athas, L."J." Svensson. Thermal logic circuits. *1994 Workshop on Physics and Computing, PhysComp '94*, Nov. 1994.

[10] I.M. Gottlieb. *Basic oscillators*. J.F. Rider, New York, 1960.

[11] W.C. Athas, L."J." Svensson, J.G. Koller, N. Tzartzanis, and E. Chou. Low-power digital systems based on adiabatic-switching principles. To appear in *IEEE Transactions on VLSI Systems*.

[12] C.A. Mead, L. Conway. *Introduction to VLSI systems*. Addison-Wesley, pg. 3, 1980.