Et^2 and Multi-voltage Logic

Mika Nyström

April 17, 1995
Revised October 25, 1999

Caltech Computer Science Technical Report
caltechCSTR:2001.008

1 Introduction

Traditionally computer systems have been operated completely at a single voltage, which has been 5 volts for over a decade and now appears to be moving towards 3.3 volts. The operating voltage has generally been taken as a constant, given by the process and selected by the designers of the manufacturing process. It is clear that the current situation creates interfacing problems. There are numerous situations in which we might wish to operate a chip at one voltage although the surrounding circuitry is operated at a different voltage.

Asynchronous circuitry exacerbates the problem of divergent operating voltages. This is because well-designed asynchronous circuits have been shown to be operational over great voltage ranges and there is no *a priori* reason to operate them at a fixed, given voltage. We may (and in fact do) expect key operating parameters to be improved at a different operating voltage from that used in the design of the manufacturing process.

We may also wish to run parts of a single chip at different voltages. This is different from the case of interfacing our new circuit to existing circuitry because we have full control of our own chip design. Given full control over our own design, why would we want do something as obviously silly as choosing different operating voltages for different parts of the circuit? There are two reasons for this. Choosing different operating voltages for different parts of the circuit amounts to choosing a lower voltage for one part than the others. We could do this because of an absolute maximum power constraint, *or* we could do it because we are not optimizing the cycle time t specifically, but rather some other metric involving t and the energy per operation E.

This research note documents a circuit that can be used to solve the voltage interfacing problem using standard CMOS transistors. The circuit has no static power dissipation, switches relatively quickly, and takes little area. The metric Et^2 is discussed in some detail, and it is shown how voltage converters relate to optimizing circuits to minimize Et^2.

2 Et^2 as a metric for circuit performance

In the literature, two metrics are generally used to indicate how "good" a given design is. The one most commonly used is the cycle time t or (inverse) "clock" frequency. (In asynchronous design we obviously do not have a clock but we can define an "average" frequency which is, of course, the inverse of the cycle time.) The other metric is the "power-delay product", which has the dimensions of energy, and is in fact the *energy per operation* E of the circuit (assuming one operation (average) per "cycle" time t.)

It is proved in elementary calculus that one cannot simultaneously optimize two metrics defined on the same independent variables. Given that we have the two metrics E and t from above, we call a design *optimal* if it optimizes some function of the form Et^n, where n represents the energy-delay tradeoff we are interested in for a particular application. If $n = 0$, we are optimizing for the energy alone, which might be (close to) the case in an application like a digital watch, and if $n \to \infty$, we move towards the approach taken in "traditional supercomputers" where minimizing the delay time is done at great energy cost. (Considering the cost of cooling systems we never quite reach total delay time optimization, and similarly, even a digital watch has to operate at a finite speed, so that extreme is not quite perfect either.)

From the point of view of the user, the parameter n may be quite variable depending on the application. There seems to be no *a priori* reason to design a circuit with n set to a particular value, but it seems that we should treat n as a variable depending on the circumstances. However, we can argue that a certain value for n closely represents the situation we really are interested in in a wide range of computer systems. It turns out that $n = 2$ has some useful properties that makes a great deal of the optimization task easier (at least in principle), and the balance of this research note will assume $n = 2$ for the global optimization. Still, it should probably be emphasized that in as far as the value of n affects the design of the final system, it needs to be calibrated to the environment of that system. In fact, as we shall see, this is a general rule for optimizing Et^n metrics; we cannot optimize our metric without knowing the behavior of the environment of every subsystem, or in simpler terms, local minima are not global minima. In the case of optimizing E, it is obvious (since energy is additive) that locally minimizing E is the same as globally minimizing E. In the case of minimizing t, we also find that locally minimizing t given the (unrealizable) assumption that reducing t for one subsystem increases t nowhere else leads to a global minimum. This is not the case for Et^n.

2.1 Theory

A simple view of CMOS logic shows that the metric Et^2 should be rougly constant for a given circuit, regardless of the voltage at which it is operating at. This view of CMOS holds that a computation basically consists of charging up a number N of capacitors with capacitance C. To first order, currents through a MOSFET go as kV^2; thus at a given supply voltage, the amount of charge to be moved through the circuit for a computation goes as CV and the time to move the charge goes as $k/(CV)$. In other words the quantity $Et^2 = CV^2 \times (k/(CV))^2 = k^2/C$, which is constant.

2.2 The Real World

Et^2 has been checked for constancy over a range of supply voltages on the Caltech Asynchronous Microprocessor. It is found to vary by approximately 50% in the range from 2 to 6 volts, with a minimum at around 3 volts. The decrease in Et^2 at voltages below 3 V can be attributed to the effects of the nonzero "threshold voltage" of the transistors. The increase in Et^2 at voltages above 3.3 V has been found to be due to velocity saturation effects and short-circuit currents. (The effect of short circuit currents is much magnified by velocity saturation.) For the time being, we shall postulate that Et^2 being constant is a fair assumption, and any deviations will be handled as higher-order corrections to our results.

3 An Idealized Model for the Performance Metric

Our performance metric is constant over different operating voltages by construction. This makes it possible to optimize a circuit globally by putting together locally optimized subcircuits. To understand how the various effects interact, we start by examining a model that gives rise to constant Et^2 by assuming simplified transistor equations.

3.1 Modeling Transistor Behavior

The model used to derive constant Et^2 is quite similar to that assumed in many cases for digital design. We assume threshold voltage $V_t = 0$, no velocity saturation, transistors operating either in saturation or "off," and saturation currents quadratic with the gate-source voltage. Assuming that switching thresholds scale linearly with the supply voltage (i.e., combinational logic) and that capacitances are constant with supply voltage variations, it is easy to derive that Et^2 is constant over varying supply voltages. It is found that the energy per operation $E = CV^2$ and that the speed of the circuit $t = CV/(kV^2)$, thus the quantity Et^2 is found constant and equal to C^3/k^2. To make this model easily applicable, we shall further assume that the supply voltage can be changed indefinitely, from zero to infinity. Any one of these assumptions is strictly incorrect, but it may be interesting to see what kinds of conclusions can be drawn since it is believed that the assumptions at least track physical reality moderately well.

3.2 Properties of Et^2

We find that in our simple model, Et^2 is constant given varying supply voltage. If we now consider two circuits, e.g., circuit A and circuit B, we find that if circuit A performs better than circuit B at five volts, we shall also find that circuit A performs better than circuit B at any other voltage, *and that this remains the case if the supply voltages for A and B are changed independently.*

There are many different metrics to optimize circuits for. A circuit designer may say that t is given to him by the environment (or the "specification" of the design), and that it is his responsibility to meet t in the first place and then to minimize E given that t. Assume that this procedure leads to a circuit X different from that found by minimizing Et^2 (circuit Y). Now change the voltage of circuit Y so that it runs with cycle time t. At this point we know that circuit Y and circuit X run at the same speed. Since Et^2 is greater for circuit X than for circuit Y, $E_X > E_Y$. Thus circuit X consumes more power than circuit Y at the same cycle time, and we should always prefer the Et^2-optimized circuit. In other words, optimizing Et^2 at any voltage and speed is equivalent to optimizing E given t, for *all* values of t (assuming that we can change the supply voltage as we please).

3.3 Block Representation of Circuits and Computations

For simplicity, we shall equate computations with circuits for the purposes of our discussion. We write our circuits as a combination of parallel and sequential composition of operators or subcircuits. Parallel composition of subcircuits allows both subcomputations to start simultaneously; the composition is said to have terminated when both subcomputations have terminated. In sequential composition, subcomputations proceed in strict sequence. A computation $A;B$ proceeds with A first, and B can start only when A has terminated.

3.4 Optimizing Parallel Composition for Et^2

Assume that we desire to complete two actions A and B in parallel; in other words, they start simultaneously, and the parallel composition terminates when both actions have terminated. Then $E = E_A + E_B$ and $t = \max(t_A, t_B)$. It is obvious by inspection that Et^2 is optimized only if $t_A = t_B$; it is also clear that any proportional change in both t_A and t_B leaves Et^2 for the composition unchanged. Thus we see that Et^2 is optimized if and only if $t_A = t_B$. Et^2 for the composed system will be optimized regardless of the actual values of t_A and t_B, as long as they are the same.

3.5 Sequential Composition

Now consider the case of sequential composition. A starts and runs to completion, and then B starts and runs to completion. The delay between the end of A and the start of B is assumed negligible. We find that $E = E_A + E_B$ and $t = t_A + t_B$. In this case the optimum is not so obvious. Let us write $M_A = E_A t_A^2$ and $M_B = E_B t_B^2$. Then we find (after some algebra) that Et^2 for the sequential composition is optimized iff

$$\frac{t_A}{t_B} = \left(\frac{M_A}{M_B}\right)^{1/3}. \tag{1}$$

4 Implementation

In the previous section, we argued that we should vary the execution time of subcircuits so as to meet time constraints given by the individual values for M (i.e., Et^2). To maintain M at a fixed value, the only thing we can do is to vary the voltage. However, CMOS circuits cannot be connected to different power supplies without interfacing circuits. If we were to try this, we would find that the lower voltage signals could not turn off a pullup network in a higher voltage unit since the highest voltage generated by the low voltage unit would not be high enough to turn off the p transistor in the high voltage unit (assuming a shared, common ground reference). The end result would be either slow or unsafe operation or high static power dissipation or both. We note that converting from the higher voltage logic to the lower voltage logic takes no extra circuitry.[1]

We have generated a circuit capable of converting low voltage logic to higher voltage logic using only MOS transistors. The circuit dissipates no static power and can be used over a wide voltage range. Depending on one's point of view, it can be seen either as a pair of "bad" inverters with an output filter or as a differential amplifier with a built-in turn-off circuit. This kind of connection is not new, but the n-transistor pullups are generally not seen.

Two details are worth noting. All low-voltage signals enter the circuit on n transistor gates. This is necessary since we cannot turn *off* a high-voltage p using a low-voltage signal. Also n-pullups are used to break the circuit out of the "fight" that occurs between the n (driven by the low voltage) and the p transistors (driven by the high voltage and then left floating) at switching. Without the n pullups, the n pulldown transistors need to be sized much larger since they have to fight the p's. The gates on the p's are not driven, but they still drive their outputs. Using n pullups to drive the gates on the p's away from a floating low helps the situation and allows more flexibility since the n's do not need to be sized with a lot of attention paid to the low driving voltage.

Finally for some applications in which we may want to used a shared reset (one of the drawbacks of the voltage converter circuit is that it needs both the low-voltage signal and its inverse; the inverse is used to reset the circuit), we can construct a staticized version of the circuit by merely adding two transistors.

[1] Just make sure the plugs are OK.

5 Et^2 in Practice

So far we have only given a method to compose circuits in order to optimize the Et^2 metric for the composed circuit, knowing the value of the metric for the subcircuits. In practice we thus need to optimize our building blocks individually for Et^2 and then connect them together according to the rules given above.

5.1 An example of optimizing Et^2 for a small circuit.

Let us assume we are given a circuit with the following properties, where E represents the energy per operation, C represents the capacitance of the switching nodes, V represents the supply voltage, t represents the "cycle time," and I represents an average current. We use C_p to denote "parasitic" capacitances; by this we mean capacitances that do not vary with the transistor width. k is chosen to denote the strength of a transistor as in $I = kV^2$. γ, η, and α are constants.

$$E = C(V^2 + \alpha V^3) \tag{2}$$

$$C = \eta(C_p + \gamma k) \tag{3}$$

$$t = \frac{CV}{I} \tag{4}$$

$$I = kf(V), f' > 0 \tag{5}$$

A straightforward application of the differential calculus yields that

$$C_p = \frac{C}{n+1} \tag{6}$$

will optimize Et^n under these conditions. Writing $C = C_p + C_g$ (where C_g is to remind of "gate" capacitance although this is not strictly true), we have

$$C_p = \frac{C_g}{n} \tag{7}$$

Examining the assumptions we see that this result is quite general. (In fact it is a lot more general than the assumptions that result in the conclusion that Et^2 is constant.)

6 Caveats

It should be clear to the attentive reader that the discussion has glossed over some important details. Let us examine where our assumptions break down.

6.1 The Real World

The experiments on the Caltech Asynchronous Microprocessor and FORTRAN simulations of ring inverters have shown that the quantity Et^2 is not constant. The variation in Et^2 is on the order of 50% in the range from 3 volts to 6 volts for a nominally 5 volt process. Et^2 in fact has a minimum around 3.5 volts for this process, and if we assume Et^n has been optimized at the nominal operating voltage we have $n > 2$. In the low voltage range, Et^2 increases due to the proximity of the "threshold" voltage, which causes the currents to drop faster as one decreases the supply voltage. In the high voltage range the situation is more complex. Velocity saturation effects limit the current to have a more linear behavior in the supply voltage. This causes the circuits to run slower than our formulas suggest which has the *additional* effect that short-circuit currents become a more serious problem since the rise time of the signals no longer decreases with the increasing supply voltage. (Since the transistors start behaving like ohmic resistances, we find that the rise time in fact approaches a constant, which is plausible.)

6.2 Reliability Constraints

Other than velocity saturation effects we have the obvious problem that CMOS VLSI circuits are sensitive to various destructive and non-destructive breakdown effects at higher voltages. We may be voltage limited by punch-through, avalanche breakdown (more likely for long transistors), thermal considerations, or perhaps most likely by device degradation due to the presence of "hot" electrons.

6.3 Granularity Constraints

Finally, it appears as if we could always arrive at a better system by breaking our circuits into smaller and smaller subcircuits and optimizing each subcircuit individually (we certainly cannot arrive at a *worse* system since the solutions allowed by the smaller grain size is a superset of those allowed by the larger grain size.) Obviously this only pays off to a point. There is a nonzero cost to doing the voltage conversion, and even if this were not the case, there is certainly a cost to introducing another power supply. (In a portable system we may, e.g., be constrained by economic considerations to voltages that are a multiple of 1.5 volts.)

7 Conclusions

We have presented a method for the global optimization of a large class of asynchronous computational circuits. Although it is true that the result that Et^2 is constant depends on many assumptions that simply are incorrect (and are getting more incorrect as feature sizes decrease), measurements and simulation have shown that we can in fact approximate Et^2 as being constant for a circuit over a range of voltages. Our argument shows that this fact can be used to optimize circuits. Keeping the *caveats* above in mind, it should be possible to use this knowledge to improve the speed-energy performance of computing machinery.

8 Acknowledgments

This work was supported by the Advanced Research Projects Agency (ARPA), United States Department of Defense and monitored by the Office of Army Research.

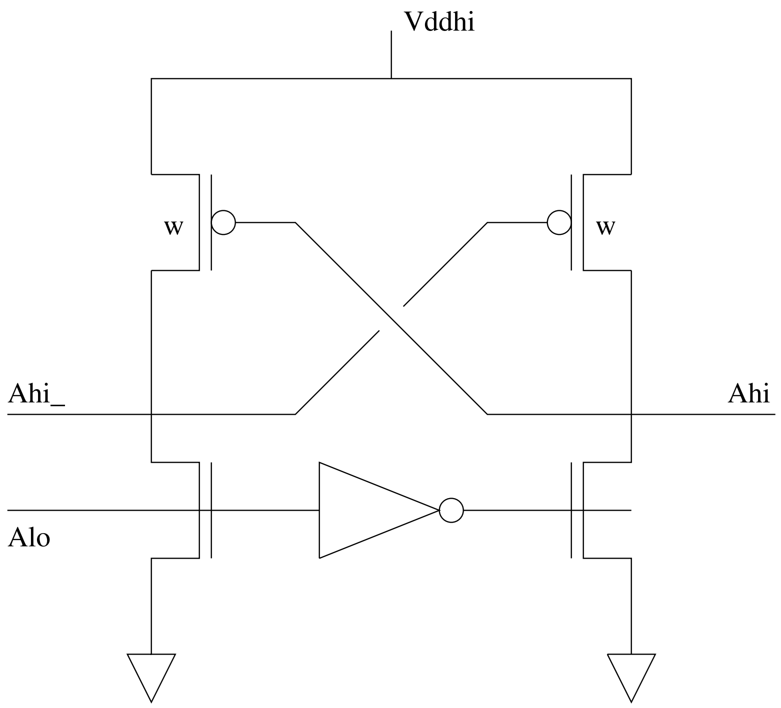

Figure 1: Voltage converter circuit. The inverter is a low voltage inverter. This circuit is slow and needs to have the transistors sized carefully due to the conflict between the n and p transistors noted in the text.

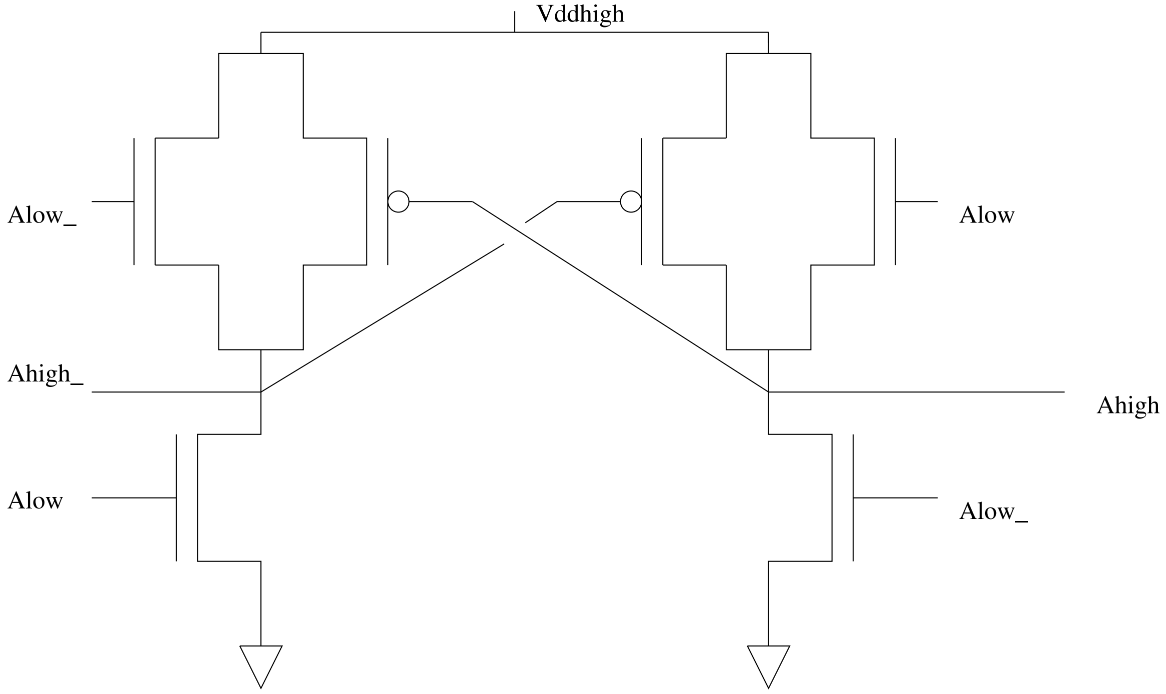

Figure 2: Improved voltage converter circuit with n pullups.

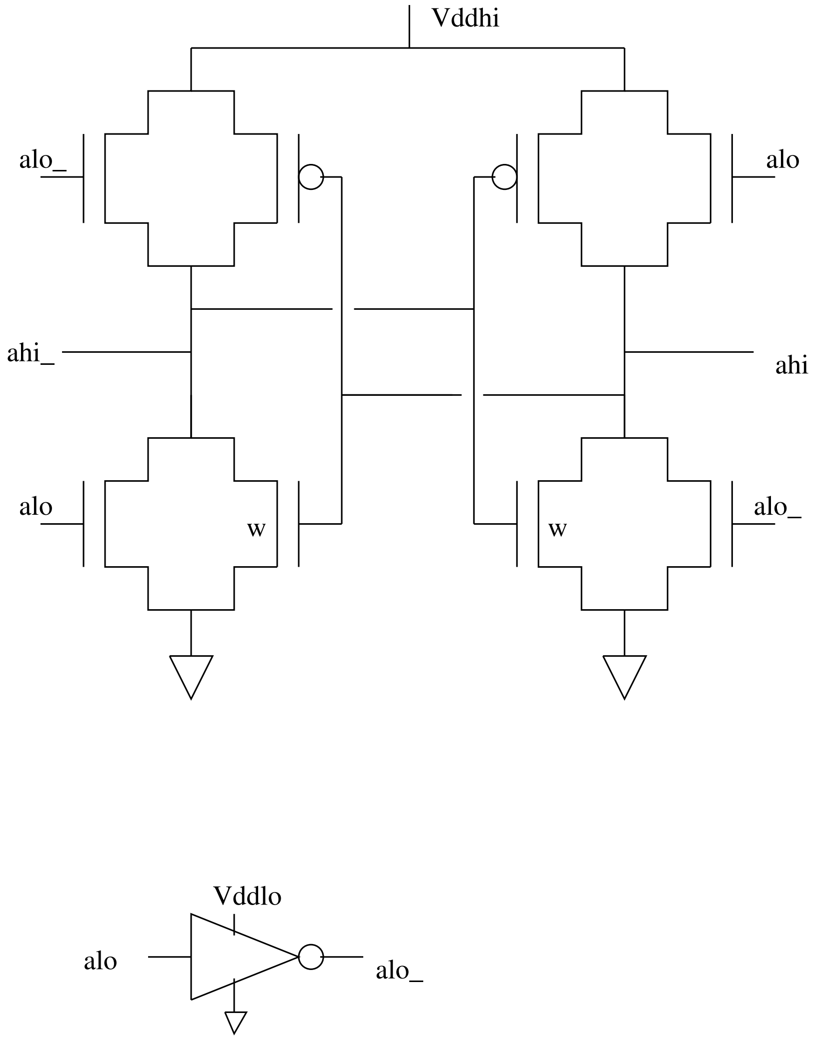

Figure 3: Staticized voltage converter.