EXPLORATION OF ENERGY-DELAY TRADEOFFS IN DIGITAL CIRCUIT DESIGN

Yoni Aizik and Avinoam Kolodny

Electrical Engineering Department,
Technion - Israel Institute of Technology, Haifa 32000 Israel,
yoni.aizik@intel.com

ABSTRACT

Modern VLSI design requires a tradeoff between circuit speed and power dissipation. Timing optimization methods typically lead to excessive power consumption. In this work, we explore the energy/performance design space in CMOS circuits, to find gate sizes which produce the lowest possible power for any specified circuit delay. The tradeoff between energy and performance is achieved by relaxing the timing of the circuit through downsizing of the cells, thus reducing the active energy dissipation. Our analysis method is based on the commonly used logical effort methodology, extended to model power as well as delay. We introduce the energy/delay gain (EDG) notation, which measures the energy reduction rate that is achievable for each delay increase that is acceptable by the designer, and the local EDG (LEDG) property, as a metric for choosing an operating point on the EDG curve, while avoiding excessively low marginal costs. The proposed analytical method is shown to be accurate when compared to simulation based numerical optimization, and orders of magnitude faster.

Index Terms— Power Performance Tradeoff, Sizing, Energy Delay Gain, EDG, Hardware Intensity

1. INTRODUCTION

Traditional design practices tend to overemphasize speed and to waste power. In the past, the design target was maximum performance, and power dissipation was not a limiting factor for the design. In recent years, however, power has become a dominant consideration, causing designers to downsize logic gates in order to reduce power, in exchange for increased delay. Let us assume that a circuit has been initially designed by traditional methods, and it needs to be redesigned for low power consumption. Resizing of gates to save power is often performed in a non-optimal way, such that for the same energy dissipation, better performance could be achieved different sizing of the gates. In this work, we explore the energy-performance design space, by evaluating the optimal tradeoff between performance and energy by tuning gate sizes in a given circuit. We explore the optimal operating point in terms of energy – performance tradeoff.

2. ENERGY EFFICIENT DESIGN

In trading off delay for energy by resizing the gates, we are interested only in a subset of all the possible downsized circuits - those implementations that are energy efficient. A design implementation is considered to be energy efficient when it has the highest performance among all possible configurations dissipating the same power ([5, 1]). When the optimal implementations are plotted in the energy-delay plane, they form a curve called the *energy efficient curve*. In Figure 1, each point represents a different hardware implementation. The implementations which belong to the energy efficient family reside on the energy efficient curve, which is a lower envelope for the set of all possible implementations in the energy-delay plane.

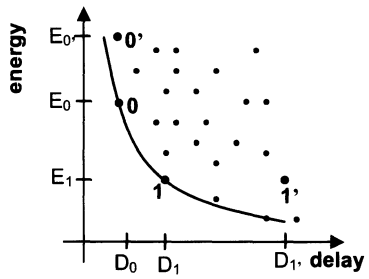

Figure 1: Energy Efficient Curve. Although implementations 0 and 0' of the given circuit have the same delay (D_0), implementation 0 consumes less energy. Similarly, implementations 1 and 1' consume the same energy, but implementation 1 has a shorter delay (D_1), hence is preferable. Points 0 and 1 reside on the energy efficient curve. All implementations have the same circuit topology, with different device sizes.

3. POWER REDUCTION USING GATE DOWNSIZING

In this paper we explore the power-performance tradeoff achievable by resizing the gates in the circuit. In order to reduce the power consumption of a digital circuit, the gates sizes are reduced, and the delay constraint is relaxed. For example, consider Figure 1, with the initial circuit implementation 0, which is energy efficient. By relaxing the delay constraint (moving from D_0 to D_1), the design is downsized, which results in circuit implementation 1. Non-optimal downsizing can lead to the circuit located at point 1', which has the same energy dissipation as 1, but it has a higher delay. To calculate the energy gain achievable by relaxing the delay by X percent, we define a measurement we call *Energy Delay Gain* (EDG). The EDG is defined as the ratio of relative decrease in energy to the corresponding relative increase in delay, w.r.t. the initial design point (D_0,E_0). D_0 is the initial delay (not necessarily the minimum achievable delay), and E_0 is the corresponding initial energy. Note that the EDG defines the **total** energy-performance tradeoff. Mathematically, EDG at a given delay D with corresponding energy E is defined as

$$EDG = \frac{(E_0 - E)/E_0}{(D - D_0)/D_0} \quad (2)$$

In the following sections, we set up an optimization framework that finds the highest possible amount of energy saving for any assumed delay constraint in a given combinational CMOS circuit. It determines the appropriate sizing factor for each gate in the circuit. For primary input and output of the circuit we assume fixed input or output capacitances, and a given activity factor and signal probability at each node of the circuit. The result of this optimization process is equivalent to finding the energy-efficient curve for the given circuit. It can be used for answering the question: "How much energy can be saved by relaxing the delay constraint for this circuit by X percents?".

4. ANALYTICAL MODEL

The optimization problem we solve is defined as follows: given a path in a circuit with initial delay (the minimum achievable delay, or larger) D_0 and the corresponding energy consumption E_0 , find the sizing that maximizes the EDG for an assumed delay constraint. We use the logical effort method ([2]) in order to calculate the delay of a path, and adapt it to calculate the dynamic energy dissipation of the circuit. For a given path (Figure 2), we assume constant input and output loads, and an initial sizing that is given as input capacitance for each gate. For each gate we apply a sizing factor k. The input capacitance of each gate is multiplied by the sizing factor, adding a degree of freedom that is required for the optimization process. The input capacitance of the resized i^{th} gate is expressed as the initial input capacitance C_{0i} multiplied by k_i. The energy-delay design space is explored by tuning the k`s.

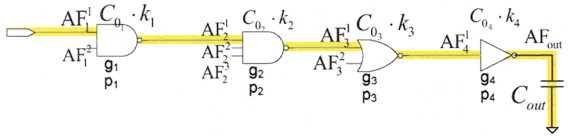

Figure 2: Example path. Each gate is assigned with logical effort notation, initial input capacitance (C0i) and sizing factor (k_i)

The following properties are defined:
Mi - Number of inputs to gate i
AF_i^j - Activity factor (switching probability) of input j in gate i
g_i - Logical effort of gate i
p_i - Parasitic delay of gate i
C_{0i} - Initial capacitance of gate i that achieves initial path delay (corresponds to (D_0, E_0))
k_i - Sizing factor for gate i. The k's are used in the gate downsizing process. For each gate i, $k_i \cdot C_{0i}$ is the gate size. Although specified, k_1 is assumed to be constant 1 (fixed driver)

4.1. Energy of a Logic Path

The switching energy E of a static CMOS gate depends on the gate's input and output capacitances. For gate g with M_g inputs, each has input capacitances of Cin_j and activity factor AF_g^j, and a single output with capacitance C_{out} and activity factor AF_{out}, the switching energy is expressed as -

$$E = Vcc^2 \cdot \left(\sum_{j=1}^{M_g} AF_g^j \cdot Cin_j + AF_{out} \cdot C_{out} \right) \quad (3)$$

Assuming the voltage amplitude for each net in the design is the same (Vcc), we can define a measurement called dynamic capacitance (Cdyn), which equals the switching energy normalized by Vcc^2. The dynamic capacitance of a gate g is

$$Cdyn_g = \frac{E}{Vcc^2} = \sum_{j=1}^{M_g} AF_g^j \cdot Cin_j + AF_{out} \cdot C_{out} \quad (4)$$

The input Cdyn of gate i is

$$Cdyn_{in}^{\ i} = C_{0i} \cdot k_i \sum_{j=1}^{M_g} AF_i^j = C_{0i} \cdot k_i \cdot AF_i \quad (5)$$

Where AF_i is defined to be sum of activity factors for input pins of gate i. The output capacitance of a gate is defined to be its self loading, and is combined mainly of the drain diffusion capacitors connected to the output. The parasitic delay in logical effort method, denoted by p, is proportional to the diffusion capacitance. It represents the ratio between the diffusion and gate capacitance. The logical effort of a gate, denoted by g, expresses the ratio of the input capacitance of a given gate to that of an inverter capable of delivering the same current. It is easy to see that the output capacitance of a gate equals

$$C_{out} = \frac{C_{in}}{g} p \quad (6)$$

We can now rewrite (5) using the notation defined above

$$Cdyn_i = C_{0i} \cdot k_i \cdot AF_i + \frac{C_{0i} \cdot k_i}{g_i} p_i \cdot AF_{i+1} \quad (7)$$

We can now calculate the total Cdyn of a logic path consists of N stages –

$$Cdyn = \sum_{i=1}^{N} k_i \left(AF_i \cdot C_{0i} + AF_{i+1} \cdot \frac{C_{0i} \cdot k_i}{g_i} \right) + AF_{out} \cdot C_{load} \quad (8)$$

By defining

$$C_i \triangleq AF_i \cdot C_{0i} + AF_{i+1} \cdot \frac{C_{0i} \cdot k_i}{g_i} \quad (9)$$

We get

$$C_{dyn} = \sum_{i=1}^{N} C_i \cdot k_i + AF_{out} \cdot C_{out} \quad (10)$$

The initial Cdyn (C^0_{dyn}) is achieved by setting all k_i's to 1.
Therefore, the energy decrease (e_{dec}) due to downsizing of the gates by a factor of k is

$$e_{dec} = \frac{C^0_{dyn} - C_{dyn}}{C^0_{dyn}} = \frac{\sum_{i=1}^{N} C_i (1 - k_i)}{\sum_{i=1}^{N} C_i + AF_{out} \cdot C_{out}} \quad (11)$$

4.2. Delay of a Logic Path

When using the logical effort notation, the path delay (D) is expressed as

$$D = \sum_{i=1}^{N} g_i h_i + P \quad (12)$$

The electrical effort of stage i (h_i) is calculated as the ratio between capacitance of gate i+1 and gate i. P is the total path parasitic delay. Using the notation defined earlier, the path delay D can be written as –

$$D = \sum_{i=1}^{N} g_i \frac{C_{0_{i+1}} k_{i+1}}{C_{0_i} k_i} + P \quad (13)$$

By defining

$$D_i \triangleq g_i \frac{C_{0_{i+1}}}{C_{0_i}}, D_0 \triangleq D\Big|_{\forall i, k_i = 1} = \sum_{i=1}^{N} D_i + P \quad (14)$$

We can get the delay increase rate (d_{inc}) due to downsizing of the gates by a factor of k –

$$d_{inc} = \frac{D - D_0}{D_0} = \frac{\sum_{i=1}^{N} D_i \frac{k_{i+1}}{k_i} + P - D_0}{D_0} \qquad (15)$$

5. OPTIMIZING POWER AND PERFORMANCE

The optimization problem is formulated as follows:
Given a delay value that is d_{inc} percent greater than the initial delay D0, find the best gate sizing that maximizes the energy reduction rate e_{dec}. Mathematically:
Minimize $f_0(k_1 \ldots k_N)$, subject to $f_1(k_1 \ldots k_N) \leq 1$, where

$$f_0(k_1 \cdots k_N) = \sum_{i=1}^{N} C_i k_i$$
$$f_1(k_1 \cdots k_N) = \sum_{i=1}^{N} \frac{D_i}{d_{inc} D_0 + D_0 - P} \cdot \frac{k_{i+1}}{k_i} \qquad (16)$$

However, f_1 defined above is non-convex. We use geometrical programming [3,4] to convert f_0 and f_1 to convex form, and solve the optimization problem. The convexity ensures that a solution to the optimization problem exists, and that the solution is global optimum point. By solving (16), a design can be placed correctly on the energy efficient curve. In order to obtain the EDG curve, the delay increase rate is swept from 0 to the desired value, and for each delay increase value, a different optimization problem is solved by geometrical programming.
In the following sections, we employ this procedure to characterize the EDG and power reduction in typical logic circuits, and derive design guidelines.

6. EXPLORING ENERGY-DELAY TRADEOFF IN A CHAIN OF INVERTERS

As an example, we run numerical experiments that explore the EDG of a basic inverter chain circuit (Figure 3). We use GGPLAB ([7]) as a geometrical programming optimizer, to solve the optimization problem (16). GGLAB is a free open source library, and can be easily installed over Matlab. For each experiment, we provide an EDG curve which is obtained by optimizing the circuit for a wide range of increased delay values.

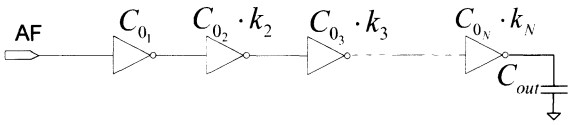

Figure 3: Inverter Chain - Consists of N stages, output load Cout, and initial capacitances ($C_{01} \ldots C_{0N}$)

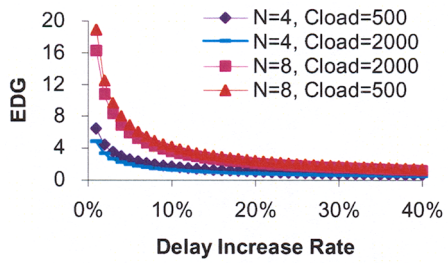

Figure 4: Inverter Chain - various loads (Cload) and chain length (N)

Consider an inverter chain consisting of N inverters, with output load of Cout. C_{01} is set arbitrarily to a constant value of 1 fF. We set initial gate capacitances ($C_{02} \ldots C_{0N}$) that ensure minimum delay, using the logical effort methodology. Figure 4 shows the EDG for different combinations of path electrical effort (H) and chain length (N). The largest potential for energy savings occurs near the point where the design is sized for minimum achievable delay (small values of delay increase). The potential for energy savings decreases as the delay is being relaxed further. This goes in line with the observation in [6].

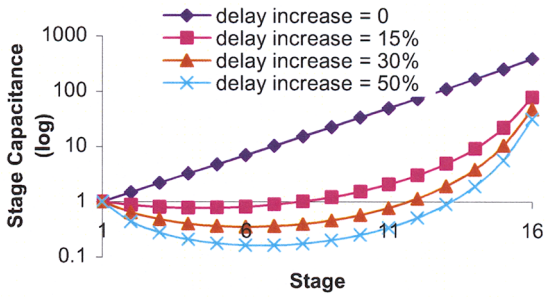

Figure 5a: Stage capacitance (chain of 16 inverters), for various delay increase rates (log scale)

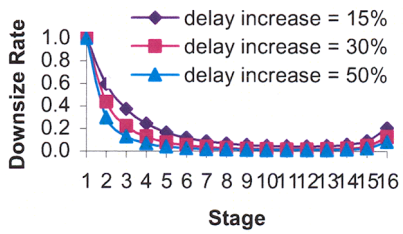

Figure 5b: Stage sizing factor (chain of 16 inverters) - ratio of gate capacitance to minimum delay capacitance, needed to meet the given delay increase rates value

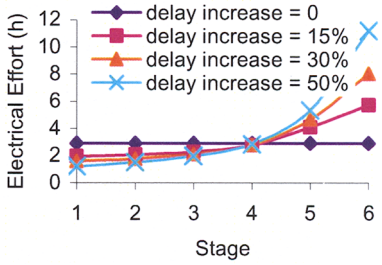

Figure 5c: Stage electrical effort (h), for various delay increase rates for chain of 6 inverters. In an inverter chain optimized for speed all stages have the same effort. In a circuit optimized for power, the last stages have larger efforts.

Figure 5 shows the optimal sizing of a fixed input and output load inverter chain, for various delay increase values. The gate sizes are expressed in the figure by the electrical effort, which is the ratio of consecutive gate sizes (see section 4.2). The optimization process results in increasing the electrical effort of the last stages, and

decreasing the electrical effort of the first stages, to meet the timing requirements (Figure 5c). The largest energy savings, for a given delay increase value, are achieved by downsizing the largest gates in the chain. The relative downsizing, however, is maximal around the middle of the chain (5b), due to the fact that the first stage and the load are anchored with a fixed size. As the delay increases, the gates towards the middle of the chain are downsized and form a plateau-like shape. Note that the optimal gate sizes might be limited by the minimum allowed design rules. Both Figures 5a and 5c illustrate that as we move further from the minimal achievable delay (delay increase = 0, where all electrical efforts are identical according to the Logical Effort theory), the difference between the electrical efforts of the stages increases. It is clear that the tapering factor that keeps the inverter chain energy efficient is not linear in the stage number. However, uniform downsizing (downsizing the gates by a fixed percent - e.g. increase the delay by downsizing each gate by 5%) is sometimes used in the power reduction process by the circuit designer as an easy and straightforward method to trade off energy for performance. Figure 6 shows the energy efficient curve (optimal sizing) vs. energy-delay curve generated by uniform downsizing of an 8-long inverter chain with out/in capacitance ratio of 200. The energy difference between the curves in the figure reaches up to 7%.

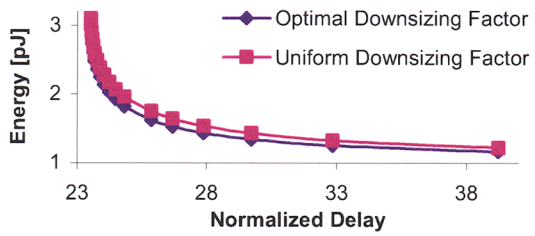

Figure 6: Uniform vs. Optimal Downsizing. Linear downsizing of an inverter chain in order to save energy by increasing the delay results in a non-optimal design - in this case 7% more energy could be saved by tuning the sizing correctly

7. FINDING THE OPTIMAL DESIGN POINT

The question "how to optimize a circuit for both energy and performance" is ambiguous, since there is a clear tradeoff between the two. Even after the tradeoff has been identified, choosing the optimal operating point is not an easy task, mainly because an optimum does not necessarily exist. For instance, one can decide that it is acceptable to reduce the frequency of the circuit by 10%, as long as the energy is reduced by 30%. But in a different circuit, where the timing demands are more aggressive, only 5% frequency reduction might be acceptable.

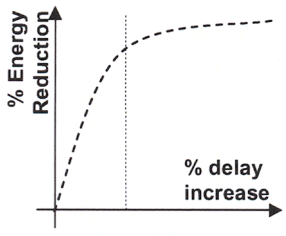

Figure 7: Energy Efficient Curve – energy reduction rate as a function of delay increase rate. Working on the left region is preferable, since the energy saving to delay increase ratio is larger than 1

Figure 7 describes a general energy efficient curve for a given design. The x axis is the delay increase rate, whereas the y axis is the corresponding energy reduction rate. It can be derived from EDG curve (figure 4, for instance), by replacing the y axis with EDG (%delay increase) multiplied by the actual %delay increase. Mathematically,

$$\%\text{Energy Reduction}=\text{EDG}(d_{inc})\cdot d_{inc} \tag{17}$$

Intuitively, working on the left region of Figure 7 is better than working at the right, where an increase in the delay results in a very small energy reduction. In the left hand region, every increase in the delay results in a bigger increase in the energy reduction rate. In order to better understand what distinguishes the left region from the right region, let us review some economic terms:

- Marginal product - the extra output produced by one more unit of an input (for instance, the difference in output when a firm's labor is increased from five to six units). Marginal product is the slope of the total product curve
- Diminishing marginal returns - in a production system, beyond some point, each additional unit of variable input yields less and less additional output. Conversely, producing one more unit of output costs more and more in variable inputs.

When running a business, the challenge is to avoid working in the region of diminishing return.

The energy/delay tradeoff can be translated to economics terms. The percentage of energy reduction is analogous to the produced product where the percentage of increase in delay corresponds to the labor invested to produce the product, and the total product curve is similar to the energy efficient curve.

We define the *local energy delay gain (LEDG)* as

$$LEDG(d,\delta)=\frac{e(d+\delta)-e(d)}{\delta} \tag{18}$$

Where e(d) is the value of the energy reduction rate at delay increase of d percent in the energy efficient curve, and e(d+δ) is the value of the energy reduction rate at delay increase of d+δ percent in the energy efficient curve (see Figure 8). We define LEDG(d) as -

$$LEDG(d)=\lim_{\delta\to\infty} LEDG(d,\delta) \tag{19}$$

In the context of economical terms defined above, LEDG(d) is the marginal product.

The challenge is to keep the marginal product high enough, and to avoid working in the diminishing return region. In economics, the criterion for using additional labor is the market selling price of the marginal product. We need a similar criterion in order to determine the acceptable range of LEDG.

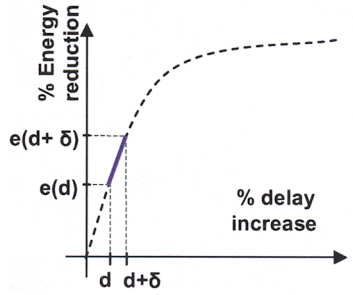

Figure 8: Local Energy Delay Gain - The local energy delay gain at point d is the lope of the solid line

We define the point d on the energy efficient curve to be α efficient if LEDG(d) is greater than α. α expresses the acceptable ratio of local energy saving to local performance degradation. The value of α can be set according to the acceptable delay/energy tradeoff. Setting α to 1 means that increasing the delay by δ results in energy saving that is greater than δ; this choice is somewhat arbitrary though.

Alternative costs need to be known for making a good economic decision. This leads us to a second option of choosing α, by comparing the energy gain achievable by gate sizing (LEDG) to improvements achievable by alternative design methods. An alternative method for trading off energy for delay is voltage scaling. The authors of [6] notice that supply voltage reduction is very effective for saving power when the delay increment is large, while gate sizing is effective around the speed-optimal design point.

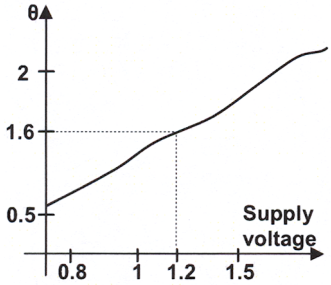

Figure 9: Voltage Intensity - Local sensitivity of energy reduction to increase in the delay, due to voltage scaling

In [1], Zyuban et. al. define voltage intensity, annotated as θ, to be the ratio of the relative energy decrease to the corresponding delay increase achieved locally through varying the power supply (Figure 9). When the local energy delay gain is less than the voltage intensity, the voltage scaling would achieve better energy/delay ratio. Therefore, α is equal to the voltage intensity of the operating voltage. For instance, the voltage intensity curve of a given circuit is illustrated by Figure 9. If the operating voltage of this circuit is 1.2, then alpha should be set to 1.6.

We can now use α efficiency as a criterion to identify the preferable work region. A preferable work region is a collection of all delay increase points that are α efficient (LEDG(dinc) ≥ α)

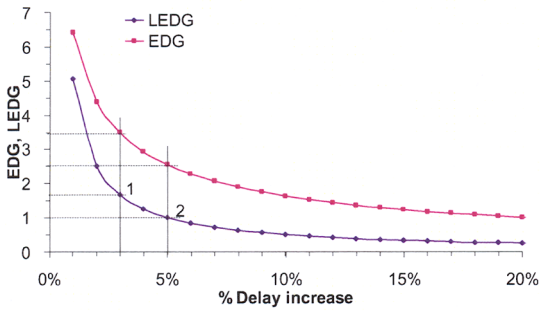

Figure 10: EDG and LEDG as a function of the delay increase rate. Points 1 and 2 represent LEDG of 1.6 and 1, respectively.

Figure 10 illustrates the efficiency criterion. Point 1 corresponds to α =1.6, as described above. This means that the preferable region would be less than 3% delay increase. At 3% delay increase, the gain would be 10.5% energy reduction (EDG=3.5). Point 2 corresponds to α = 1, as described above, with preferable region of less than 5% delay increase. At 5% delay increase, the gain would be 12.5% energy reduction (EDG=2.5). The points beyond 5% delay increase are characterized by diminishing return –any increment in the delay will result in a **smaller** reduction of the energy.

8. CONCLUSION

We have presented a design optimization framework that explores the power-performance space by gate sizing in static digital CMOS logic circuits. The framework provides fast and accurate answers for the questions:

- How much power can be saved by slowing down the circuit by x percent?
- How to determine gate sizes for optimal power under a given delay constraint?

The method is based on the commonly used logical effort theory, extended to model power as well as delay.

We introduced the energy/delay gain (EDG) as a metric for the amount of energy that can be saved as a function of increased delay, and the local energy/delay gain (LEDG) as a metric for the efficiency of the chosen working point on the energy efficient curve. While the framework developed here can generate the energy efficient curve automatically, choosing a working point on the curve requires additional considerations, where engineering judgment is essential.

The result of this work can be applied in circuit synthesis tools and provide intuitive guidelines for circuit designers regarding power-aware gate sizing in CMOS digital logic.

9. REFERENCES

[1] V. Zyuban and P. N. Strenski, "Balancing hardware intensity in microprocessor pipelines", *IBM Journal of Research and Development*, vol. 47 no. 5/6, pp 585-598

[2] Ivan E. Sutherland, Robert F. Sproull, and David F. Harris, *Logical Effort: Designing Fast CMOS Circuits* Margan Kaufmann.

[3] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*, Cambridge University Press, 2006.

[4] Stephen Boyd, Seung Jean Kim, Lieven Vandenberghe and Arash Hassibi, "A Tutorial on Geometric Programming", *Optimization and Engineering,* vol. 8 2007, pp 67-127

[5] Paul I.Penzes and Alain J.Martin, "Energy-Delay Efficiency of VLSI Computations", *GLSVLSI02*, April, 2002

[6] Robert W.Brodersen, Mark A.Horowitz, Dejan Markovic, Borivoje Nikolic and Vladimir Stojanovic, "Methods for True Power Minimization", *International Conference on Computer Aided Design*, 2002

[7] Almir Mutapcic, Kwangmoo Koh, Seungjean Kim, Lieven Vandenberghe and Stephen Boyd, *GGPLAB: A Simple Matlab Toolbox for Geometric Programming*, http://www.stanford.edu/boyd/ggplab/, May 2006