Chapter 9

DESIGN OF ENERGY EFFICIENT DIGITAL CIRCUITS

Vojin G. Oklbodzija and Bart R. Zeydel
ACSEL Laboratory, University of California Davis

Abstract: Recent technology advances have resulted in power being the major concern for digital design. In this chapter we address how transistor sizing affects the energy and delay of digital circuits. The state of the art in circuit design methodology (Logical Effort) is examined and we identify its inadequacy for design in the energy-delay space. We explain how to explore the entire energy-delay space for a circuit and present an approach for the design and analysis in the energy-delay space which allows for energy reduction without performance penalty. Finally we present techniques for the design of energy-efficient digital circuits.

Key words: digital circuits, energy-delay optimization, energy-delay space, performance optimization, power optimization, transistor sizing

1. INTRODUCTION

CMOS technology advances have led to dramatic improvements in performance while also achieving similar reductions in power. However, as device dimensions decreases to 180nm and below traditional constant field scaling can no longer be applied [22][23]. The problem with this trend is that performance and power no longer scale proportionally across technology nodes, leading to increasing power density [19]. Further adding to this problem has been the drive to produce chips operating at higher and higher clock frequencies. The net result is that designers have focused solely on optimizing circuits for delay regardless of energy.

In this chapter we will examine the energy and delay characteristics of a digital circuit and relate them to the physical dimensions of transistors.

Using these models we will analyze Logical Effort (LE) [6][7], the state of the art design methodology for digital circuits. The location of the LE solution in the energy-delay space is then examined to determine its applicability to energy-efficient design. The analysis demonstrates that LE does not guarantee an energy-efficient circuit. To address this we examine the entire energy-delay space for a circuit that can be obtained through transistor sizing. From this we present a simplified approach for the high-level exploration of the energy-delay characteristics of a circuit. Using this analysis we present several guidelines for the design of energy-efficient digital circuits.

2. RC MODELING OF GATE DELAY

Delay modeling techniques for evaluating large circuits have historically involved the simplification of current based delay modeling. The most common simplification assumes a step input, allowing for the current to be approximated over the time of interest [1][2][3][4][5]. The simplest delay model which allows for the relative analysis of delay characteristics of logic gates is Logical Effort (LE) [6][7].

2.1 Logic Gate Characteristics

In this section we relate the physical characteristics of CMOS logic gates to the delay characteristics of a logic gate. The layout of a CMOS inverter is shown in Figure 9-1. The physical parameters are W_n and W_p, which represent the widths of the nMOS and pMOS transistors respectively, and L_n and L_p which represent the channel length of the nMOS and pMOS transistors. Understanding the dependence of gate capacitance, parasitic capacitance and effective channel resistance on these physical parameters is essential to the use of RC modeling for the optimization of CMOS logic gates.

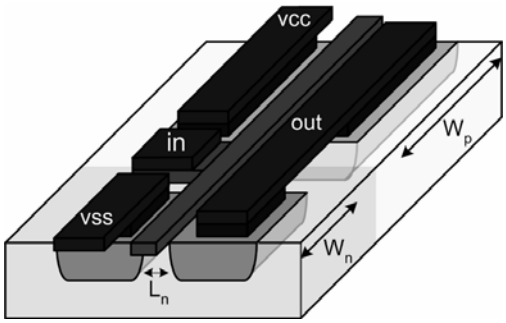

Figure 9-1. Layout of a CMOS Inverter

2.1.1 Gate Capacitance

Gate capacitance, C_{gate}, is a function of the effective channel length, L_{eff}, and the width of the transistor, W. The effective channel length can be calculated from the drawn transistor length as $L_{eff} = L_{drawn} - 2x_d$, as seen in Figure 9-2. To simplify notation we will refer to L_{eff} as L.

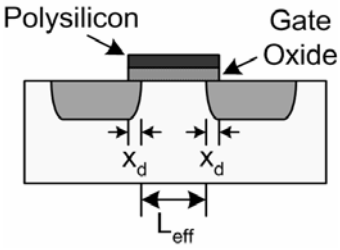

Figure 9-2. MOSFET Gate Capacitance

The gate capacitance of each transistor can be calculated from the width and length of the transistor and the per area capacitance, C_{ox}, of the gate.

$$C_{gate} = W \cdot L \cdot C_{ox}$$

The gate capacitance is directly proportional to the width of the transistor. Thus, as the width changes by a factor α the gate capacitance also changes by the same factor α.

$$C_{gate} = \alpha \cdot W \cdot L \cdot C_{ox}$$

The capacitance of an input to a gate, C_{in}, is the sum of the gate capacitances attached to the input. For example the input capacitance of an inverter is:

$$C_{in} = \left(W_n \cdot L_n + W_p \cdot L_p\right) \cdot C_{ox}$$

Scaling the width of each transistor in the inverter by a factor α causes C_{in} to also scale by α.

2.1.2 Parasitic Capacitance

The parasitic capacitance of a transistor has two components. The junction capacitance, C_{ja} – expressed in F per area in μ^2, and the periphery capacitance, C_{jp} - expressed in F per μ of the periphery length. These components are shown in Figure 9-3.

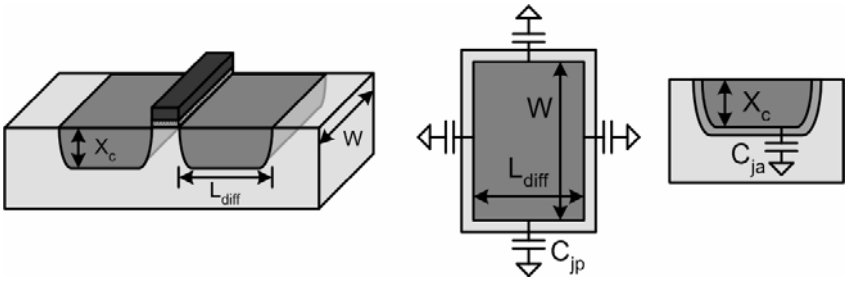

Figure 9-3. MOSFET Parasitic Capacitance

The parasitic capacitances of each transistor can be computed directly from layout as:

$$C_p = C_{ja} \cdot W \cdot L_{diff} + C_{jp} \cdot (2 \cdot W + 2 \cdot L_{diff})$$

Similar to the gate capacitance the parasitic capacitance is directly proportional to transistor width. Thus, as the transistor width changes by a factor α the parasitic capacitance changes linearly with α.

2.1.3 Resistance

The channel resistance, $R_{channel}$, in a MOSFET is dependent on its region of operation, transistor width, and channel length. In saturation $R_{channel}$ can be expressed as:

$$R_{channel(sat)} = \frac{\partial V_{gs}}{\partial I_{d(sat)}} = \frac{L}{W \cdot \mu \cdot C_{ox}\left(V_{gs} - V_t\right)}$$

In linear or triode, $R_{channel}$ can be expressed as:

$$R_{channel(lin)} = \frac{\partial V_{gs}}{dI_{d(lin)}} = \frac{L}{W \cdot \mu \cdot C_{ox} \cdot \left(V_{ds}\right)}$$

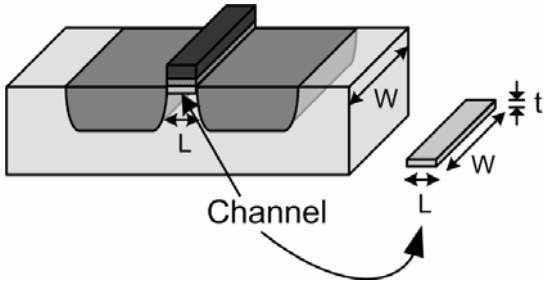

Figure 9-4. MOSFET Channel Resistance

The resistance of the channel is inversely proportional to the width of the transistor in both saturation and linear regions of operation. Thus by changing the width of the transistor by a factor α, the resistance of the transistor changes by $1/\alpha$.

2.2 RC Delay Model

The propagation delay of a CMOS logic gate can be represented using a RC-model [1]. The model can be derived assuming a step input (Figure 9-5), and related to gate capacitance, parasitic capacitance and channel resistance. C_{load} represents the output load, C_{out}, and parasitic load, C_p at the output of the gate. The high-to-low propagation delay, t_{hl} , is the delay from the time when the input is $V_{dd}/2$ to the time when V_{out} is equal to $V_{dd}/2$. The derivation will be shown for t_{hl} , however the same derivation can be performed for t_{lh}.

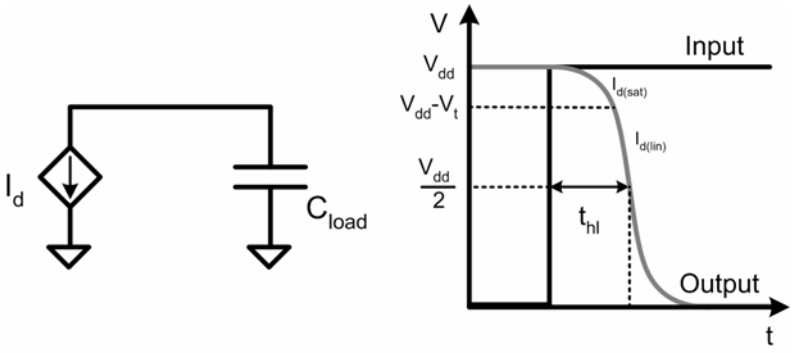

Figure 9-5. Step input response

The propagation delay, t_{hl} , can be calculated from:

$$-I_d = C_{load} \cdot \frac{\partial V_{out}}{\partial t}$$

Where t_{hl} is given by:

$$t_{hl} = -\int_{V_{dd}}^{V_{dd}/2} \frac{C_{load}}{I_d} \partial V_{out}$$

Assuming a step input, the transistor will be in saturation from V_{out} = V_{dd} to V_{dd}-V_t. In saturation the current is given by:

$$I_{d(sat)} = \mu_n \cdot C_{ox} \cdot \frac{W}{L}(V_{dd} - V_t)^2$$

The transistor will be in the linear region from V_{out} = V_{dd}-V_t to $V_{dd}/2$. In the linear region current is given by:

$$I_{d(lin)} = \mu_n \cdot C_{ox} \cdot \frac{W}{L}\left((V_{dd} - V_t) \cdot V_{out} - \frac{{V_{out}}^2}{2}\right)$$

Substituting into the integration for t_{hl}:

$$t_{hl} = -\int_{V_{dd}}^{V_{dd}-V_t} \frac{C_{load}}{I_{d(sat)}} \partial V_{out} - \int_{V_{dd}-V_t}^{V_{dd}/2} \frac{C_{load}}{I_{d(lin)}} \partial V_{out}$$

Integrating over the time when the transistor is in saturation:

$$t_{hl(sat)} = -\frac{C_{load}}{\mu_n \cdot C_{ox} \cdot \frac{W}{L} \frac{(V_{dd} - V_t)^2}{2}} \int_{V_{dd}}^{V_{dd}-V_t} \partial V_{out} = \frac{2 \cdot V_t \cdot C_{load}}{\mu_n \cdot C_{ox} \cdot \frac{W}{L} (V_{dd} - V_t)^2}$$

Integrating over the time when the transistor is in linear operation:

$$t_{hl(lin)} = -\frac{C_{load}}{\mu_n \cdot C_{ox} \cdot \frac{W}{L}} \int_{V_{dd}-V_t}^{V_{dd}/2} \left(\frac{1}{(V_{dd} - V_t) \cdot V_{out} - \frac{V_{out}^2}{2}} \right) \cdot \partial V_{out} = \frac{C_{load}}{\mu_n \cdot C_{ox} \cdot \frac{W}{L}} \cdot ln\left(3 - 4\frac{V_t}{V_{dd}}\right)$$

Substituting $t_{hl(sat)}$ and $t_{hl(lin)}$ into t_{hl} :

$$t_{hl} = \frac{C_{load}}{\mu_n \cdot C_{ox} \cdot \frac{W}{L}(V_{dd} - V_t)} \cdot \left(\frac{2 \cdot V_t}{V_{dd} - V_t} + ln\left(\frac{4 \cdot (V_{dd} - V_t) - V_{dd}}{V_{dd}} \right) \right)$$

The channel resistance is physically dependent on W, L μ, and C_{ox}. These terms can be grouped to describe the effective resistance of the channel, $R_{channel}$:

$$R_{channel} = \frac{L}{\mu_n \cdot C_{ox} \cdot W}$$

Substituting $R_{channel}$:

$$t_{hl} = R_{channel} \cdot C_{load} \cdot \frac{1}{(V_{dd} - V_t)} \left(\frac{2 \cdot V_t}{V_{dd} - V_t} + ln\left(3 - 4\frac{V_t}{V_{dd}}\right) \right)$$

The remaining terms can be grouped into a constant:

$$\kappa = \frac{1}{(V_{dd} - V_t)} \left(\frac{2 \cdot V_t}{V_{dd} - V_t} + ln\left(3 - 4\frac{V_t}{V_{dd}}\right) \right)$$

The resulting delay of a gate can be expressed as:

$$t_{hl} = \kappa \cdot R_{channel} \cdot C_{load} = \kappa \cdot R_{channel} \cdot (C_{out} + C_p)$$

RC modeling allows for linear modeling of gate delay with respect load C_{load}. The delay associated with output load is the same for each gate as long as $R_{channel}$ is the same for each. A graphical representation of this model is shown in Figure 9-7. R_{up} and R_{down} denote the equivalent pull-up and pull-down resistance of a gate.

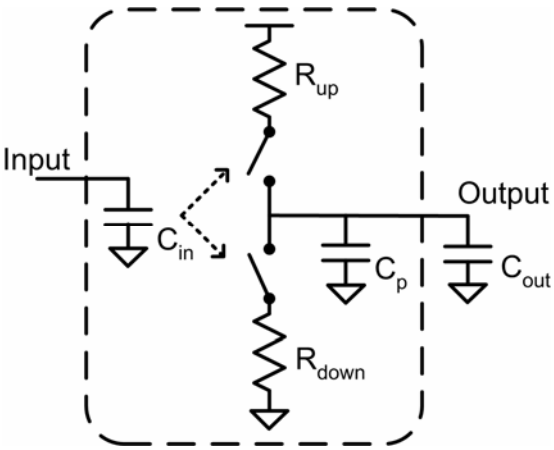

Figure 9-6. RC Model for a CMOS gate

We would like to observe the delay dependence as transistor widths are increased by a factor, α. The original resistances and capacitances will be referred to as the template. The resistance of a gate changes inversely with α, as:

$$R_{channel} = R_{template} / \alpha$$

The input capacitance and parasitic capacitance of the gate both change directly with α:

$$C_{in} = C_{template} \cdot \alpha$$

$$C_p = C_{p(template)} \cdot \alpha$$

Plugging the scaled values for resistance and capacitance into the RC delay model yields:

$$t_d = \kappa \cdot \left(\frac{R_{template}}{\alpha} \right) \left(C_{out} + \alpha \cdot C_{p(template)} \right)$$

It is observed that the parasitic delay of a gate does not change with the size of the gate.

$$t_d = \kappa \cdot \left(\frac{R_{template}}{\alpha} \right) \cdot C_{out} + \kappa \cdot R_{template} \cdot C_{p(template)}$$

However, the delay associated with a constant load changes inversely with the sizing factor α. Through substitution delay can be expressed in terms of C_{in} and C_{out} of the gate instead of using the relative term α.

$$t_d = \kappa \cdot \left(R_{template} \cdot C_{template} \cdot \left(\frac{C_{out}}{C_{in}} \right) + \kappa \cdot R_{template} \cdot C_{p(template)} \right)$$

2.3 Logical Effort Delay Model

In 1991 R. F. Sproull and I.E. Sutherland suggested that a technology independent delay could be obtained by normalizing the RC-delay model of a gate [6][7]. They suggested a relative model for the delay of the gates, which normalized the delay of a gate to delay of an inverter driving an identical copy of itself without parasitics.

$$t_d = \kappa \cdot R_{inv} \cdot C_{inv} \left(\frac{R_{template} \cdot C_{template}}{R_{inv} \cdot C_{inv}} \cdot \left(\frac{C_{out}}{C_{in}} \right) + \frac{R_{template} \cdot C_{parasitic}}{R_{inv} \cdot C_{inv}} \right)$$

In this form, the only technology dependent portion of the delay expression is the normalized inverter delay and κ, which they referred to as τ.

$$\tau = \kappa \cdot R_{inv} \cdot C_{inv}$$

The logical effort (g), or relative drive capability, of each gate is given by:

$$g = \frac{R_{template} \cdot C_{template}}{R_{inv} \cdot C_{inv}}$$

The parasitic delay (p) of each gate is given by:

$$p = \frac{R_{template} \cdot C_{parasitic}}{R_{inv} \cdot C_{inv}}$$

The relationship of output load to input capacitance is referred to as the electrical effort (h) of the gate.

$$h = \frac{C_{out}}{C_{in}}$$

Using these terms delay can be expressed in a technology independent form, which we refer to as the Logical Effort (LE) delay model:

$$t_d = (gh + p) \cdot \tau$$

The component of delay, gh, is referred to as the stage effort, f. A graphical representation of the terms used in LE is shown in Fig. 9-7.

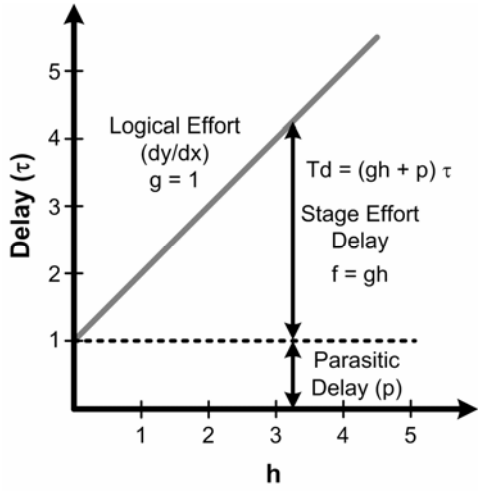

Figure 9-7. Logical Effort Delay Components of an Inverter

While the terms can clearly be obtained from simulation as in Figure 9-7, one of the most useful features of LE is the ability to obtain these characteristics through hand analysis. The logical effort of a gate can be determined by equalizing the resistance of the gate to the inverter and computing the ratio of input capacitances (Note: in the LE model parasitic

capacitances are only accounted for at the output node, allowing for transistors in series to be treated as resistors in series). The input capacitance is the sum of the gate widths attached to an input of the circuit. For example, input-a of the 2-input NAND gate in Figure 9-8 has a total width of 4, which normalized to the input capacitance of the inverter yields a logical effort, g = 4/3. The parasitic capacitance can be determined from the ratio of transistor widths attached to the output node. For example, in the 2-input NOR, the total transistor width attached to the output node is 6, which normalized to the input capacitance of the template inverter which gives $2p_{inv}$. To simplify analysis it is often assumed that $C_{p(inv)} = C_{inv}$, giving $p_{inv} = 1$.

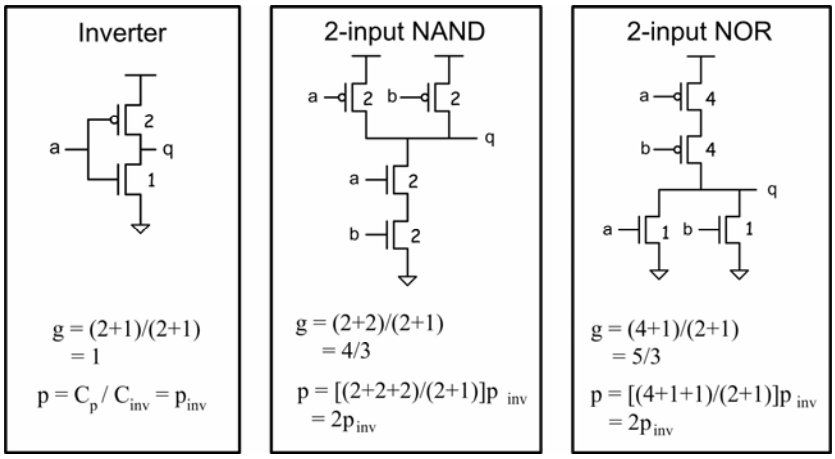

Figure 9-8. Logical Effort of an Inverter, 2-input NAND, and 2-input NOR gates.

3. DESIGNING FOR SPEED

Designing circuits for speed has been the focus of digital circuit designers since the inception of CMOS technology. Initially attempts focused on reducing the number of gates on the critical path through logic restructuring. As CMOS technology progressed designers were given the ability to modify transistor sizes to improve the performance of circuits. CAD tools such as TILOS [13] were used to optimize the performance of circuits, however they offered designers little insight into the relative performance of designs or how gates are sized for optimal delay. Logical Effort filled this void by providing designers with the ability to compare delay optimized digital circuits in an intuitive manner. The value of Logical

Effort compared to CAD tools was in its ability to provide a solution for the optimal delay of a single path circuit without iteration.

3.1 Delay Optimization of a Single Path Circuit

Logical Effort provides a fast means for optimizing the delay of a chain of gates driving a load [6][7]. The constraints on the optimization are a fixed output load and a fixed input size. The derivation for the optimal delay for chain of 3 gates will be shown, which can be extended to an n-gate path.

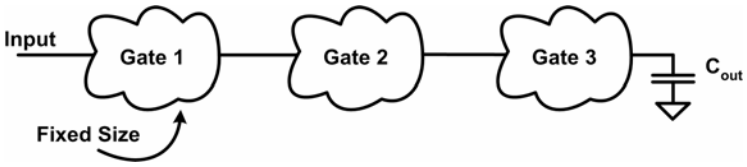

Figure 9-9. Chain of Gates with a Fixed Output Load and Fixed Input Size

The delay of the path can be expressed as:

$$T_{path} = \left[\left(g_1 \frac{C_2}{C_1} + p_1 \right) + \left(g_2 \frac{C_3}{C_2} + p_2 \right) + \left(g_3 \frac{C_{out}}{C_3} + p_3 \right) \right] \cdot \tau$$

The input capacitances of gates 1, 2 and 3 are referred to as C_1, C_2, and C_3 respectively. The delay of the path is optimized under the constraints of fixed C_{out} and C_1. The minimum delay can be found by taking the derivative of the path delay with respect to C_2 and C_3.

$$\frac{\partial T_{path}}{\partial C_3} = \frac{g_1}{C_1} - g_2 \frac{C_3}{C_2^2} = 0 \qquad \frac{\partial T_{path}}{\partial C_3} = \frac{g_2}{C_2} - g_3 \frac{C_{out}}{C_3^2} = 0$$

Rearranging the expression yields:

$$g_1 \frac{C_2}{C_1} = g_2 \frac{C_3}{C_2} \qquad g_2 \frac{C_3}{C_2} = g_3 \frac{C_{out}}{C_3}$$

Expressed in terms of stage effort, $f_1 = f_2$ and $f_2 = f_3$. Thus, the minimum occurs when the stage efforts of each gate are equal, $f_1 = f_2 = f_3$. The optimization can be generalized to an N-stage chain of gates. The following definitions allow for quick evaluation of the delay of paths based on the

characteristics of the gates on the path. The electrical effort of the path, *H*, is defined as the ratio of output to input capacitance of the path.

$$H = \frac{C_{out}}{C_{in}}$$

The Logical Effort of the path, *G*, is defined as the product of the logical effort of the gates on the path.

$$G = \prod g_i$$

Path Effort, *F*, is defined as the product of the individual stage efforts.

$$F = G \cdot H$$

Using these simplifications the optimal stage effort for a path can be determined by:

$$f_{opt} = F^{1/N}$$

Using LE, the optimal delay of a chain of gates is:

$$T_{path} = \left(N \cdot F^{1/N} + \sum p_i\right) \cdot \tau$$

3.1.1 Delay Optimized Sizing Example

Optimize the size of the gates in Figure 9-10. The input capacitance of each gate is referred to as C_{in}, C_2, C_3, and C_4. Each capacitance refers to the summation of input capacitances attached to the input of the gate along the path being analyzed.

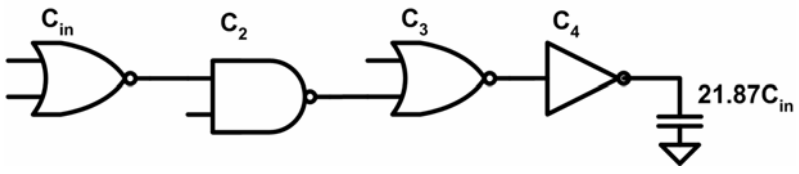

Figure 9-10. Example Chain of Gates

The optimal sizing is obtained from f_{opt}:

$$f_{opt} = (GH)^{1/4} = \left(\left(\frac{5}{3}\cdot\frac{4}{3}\cdot\frac{5}{3}\cdot 1\right)\cdot 21.87\right)^{1/4} = 3$$

The resulting optimal delay of the path is $T_d = (12 + 7p_{inv})\tau$. Using f_{opt} we can compute the input capacitance of each gate which represents the size of the gate :

$$C_i = \frac{g_i C_{i+1}}{f_{opt}}$$

The delay optimal sizing of the chain of gates is shown in Figure 9-11.

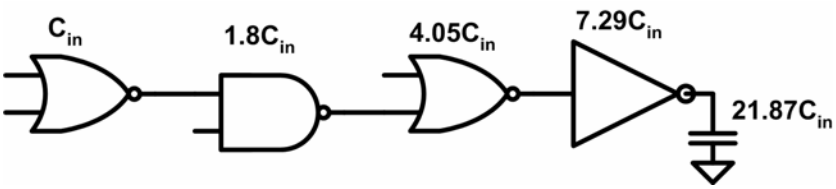

Figure 9-11. Delay Optimal Sizing of the Chain of Gates

3.2 Delay Optimization of Circuits with Branching

Although the solution to the previous problem is useful for buffer optimization, it is difficult to find other practical applications for a chain of gates without branching. The application of LE to multi-path circuits is relatively straightforward, although oversimplified in the original publication [2]. LE introduces branching (b_i) to allow for the analysis of multi-path circuits, and is defined as the factor by which $C_{off\text{-}path}$ increases $C_{on\text{-}path}$.

$$b = \frac{C_{on-path} + C_{off-path}}{C_{on-path}}$$

This often leads to confusion as the definition for h includes the branching factor:

$$h = \frac{C_{on-path} \cdot b}{C_{in}} = \frac{C_{on-path} \cdot \frac{C_{on-path} + C_{off-path}}{C_{on-path}}}{C_{in}} = \frac{C_{on-path} + C_{off-path}}{C_{in}} = \frac{C_{out}}{C_{in}}$$

When applying to a path it can be seen that the product of h_i's is as follows

$$\prod_{i=0}^{n} h_i = B \cdot H \text{ where, } B = \prod_{i=0}^{n} b_i$$

Resulting in the following expression for Path Effort, F:

$$F = G \cdot B \cdot H$$

The reason Branching is introduced is to simplify the calculation of F, as H can be extracted.

3.2.1 Multi-Path Circuit Optimization Example

To achieve minimum delay in this multi-path circuit, the delay through Path A and Path B must be equal [CAD].

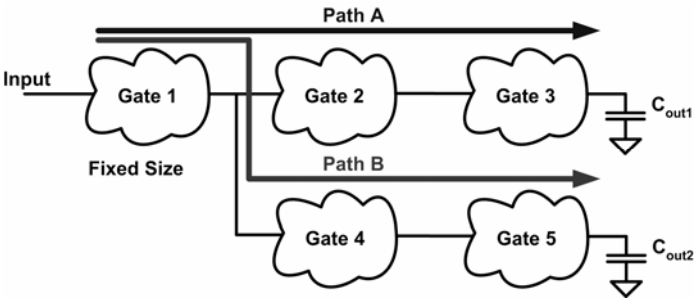

Figure 9-12. Example Multi-Path Circuit

The delay for Path A and B can be expressed as:

$$T_{Path-A} = [(g_1 h_1 + p_1) + (g_2 h_2 + p_2) + (g_3 h_3 + p_3)] \cdot \tau$$

$$T_{Path-B} = [(g_1 h_1 + p_1) + (g_4 h_4 + p_4) + (g_5 h_5 + p_5)] \cdot \tau$$

The branching at the output of Gate 1 for Path A and B can be determined as follows:

$$b_{path-A} = \frac{C_2 + C_4}{C_2} \qquad b_{path-B} = \frac{C_4 + C_2}{C_4}$$

Solving for C_2 and C_4:

$$C_2 = \frac{g_2 g_3 \cdot C_{out1}}{f_2 f_3} \qquad C_4 = \frac{g_4 g_5 \cdot C_{out2}}{f_4 f_5}$$

Substituting C_2 and C_4 into $b_{path\text{-}A}$ and $b_{path\text{-}B}$ obtains:

$$b_{path-A} = \frac{\frac{g_2 g_3 \cdot C_{out1}}{f_2 f_3} + \frac{g_4 g_5 \cdot C_{out2}}{f_4 f_5}}{\frac{g_2 g_3 \cdot C_{out1}}{f_2 f_3}} \qquad b_{path-B} = \frac{\frac{g_4 g_5 \cdot C_{out2}}{f_4 f_5} + \frac{g_2 g_3 \cdot C_{out1}}{f_2 f_3}}{\frac{g_4 g_5 \cdot C_{out2}}{f_4 f_5}}$$

We know that the optimal delay of a path without branching occurs when each stage has the same stage effort. Simplifying the delay to only include stage effort (by ignoring the parasitic delay difference between the Path A and B) the delay of each branch is equal when $f_2 = f_3 = f_4 = f_5$. Allowing for $b_{path\text{-}A}$ and $b_{path\text{-}B}$ to be expressed as:

$$b_{path-a} = \frac{g_2 g_3 \cdot C_{out1} + g_4 g_5 \cdot C_{out2}}{g_2 g_3 \cdot C_{out1}} \qquad b_{Path-B} = \frac{g_4 g_5 \cdot C_{out2} + g_2 g_3 \cdot C_{out1}}{g_4 g_5 \cdot C_{out2}}$$

A special case for branching occurs when $g_2 g_3 = g_4 g_5$ and $C_{out1} = C_{out2}$. In this case $b_i = 2$. Often this simplification is applied to the analysis of circuits, due to the fact that the value of b_i when analyzing the most critical path must be between 1 and 2. The use of b_i equal to the number of paths that branch from a node can be used as a worst case approximation of the impact of a branch on the delay of the circuit.

While the above allows for off-path gate load to be approximated, constant offpath loads of minimum sized gates and interconnect are not accounted for. The issue with interconnect load is that it is a weak function of gate size, and minimum gate sizes introduce nonlinearity into the branching computation. Accurate accounting for interconnect and minimum sized gates requires an iterative solution.

3.3 Designing High-Performance Circuits

The delay optimal solution for a path has two components. A constant parasitic delay and a variable delay dependent on *H*. As *H* decreases, the delay of the path approaches the parasitic delay. The minimal delay for a path is obtained at the smallest path gain.

$$T_{path} = \left(N\sqrt[N]{GBH} + \sum p\right)$$

The Logical Effort, *G*, of a path is constant, regardless of *H*. While the Branching, B, is roughly constant (depending on the impact of nonlinearities such as wire and minimum sized gates). These parameters define the inherent complexity of a circuit. We refer to the product of these terms as the logic complexity of a circuit. By analyzing the logic complexity of a circuit in conjunction with its parasitic delay it is possible to compare circuits over a range of path gains to gain insight into designing fast circuits (Table 9-1).

Table 9-1. Delay Comparison of two circuits X and Y over Varying H

Parasitic Delay (P)	Logic Complexity (GB)	Logic Stages (S)	Best Design for all H
$P_X = P_Y$	$G_XB_X = G_YB_Y$	$S_X = S_Y$	Equal delay
$P_X = P_Y$	$G_XB_X < G_YB_Y$	$S_X = S_Y$	X is faster
$P_X < P_Y$	$G_XB_X = G_YB_Y$	$S_X = S_Y$	X is faster
$P_X < P_Y$	$G_XB_X < G_YB_Y$	$S_X = S_Y$	X is faster
$P_X < P_Y$	$G_XB_X > G_YB_Y$	$S_X = S_Y$	Depends on H
-	-	S_X != S_Y	Depends on H

In the table, it is seen that if two circuits, X and Y, which implement the same function will always have the same delay if they have the same parasitic delay, logic complexity and number of stages. Additionally we find that a circuit will always be the same speed or faster than another circuit with the same number of stages if it has the same or less parasitic delay and logic complexity. However, if the circuit has less parasitic delay yet more logic complexity than another circuit, the faster design will depend on the value of *H*. If the number of stages differs between two circuits the better of the two will also depend on *H*. A thorough understanding of these concepts is critically important for creating a comparison of circuits and interpreting the results.

4. DESIGN IN THE ENERGY-DELAY SPACE

CMOS technology scaling no longer has the favorable characteristics of having constant power-density. As a result it is no longer possible to design for a technology independent design goal (as is assumed when applying Logical Effort). Instead both the energy and delay of a circuit must be accounted for. In this section we present a basic energy model which can be combined with RC-delay modeling to provide an energy estimate for LE delay optimized points. From these points we explore the energy-delay space of digital circuits to identify the efficient region of operation and identify energy-efficient characteristics of circuits

4.1 Energy Model

We desire an energy model which yields accurate results yet can be computed directly from gate size and output load (making it compatible with the transistor sizing described in section 3). For hand estimation the active energy of a circuit can be computed directly from the output load of the circuit, as:

$$E = C_{load} \cdot V_{dd}^{\ 2} = \left(C_p + C_{out}\right) \cdot V_{dd}^{\ 2}$$

This model neglects the energy associated with short-circuit current and leakage. The energy model can be improved through simulation to include the energy associated with short-circuit current and leakage. The energy of a 2-input NAND gate obtained from simulation is shown in Figure. 9-13. A linear dependence of energy on input size and output load is observed.

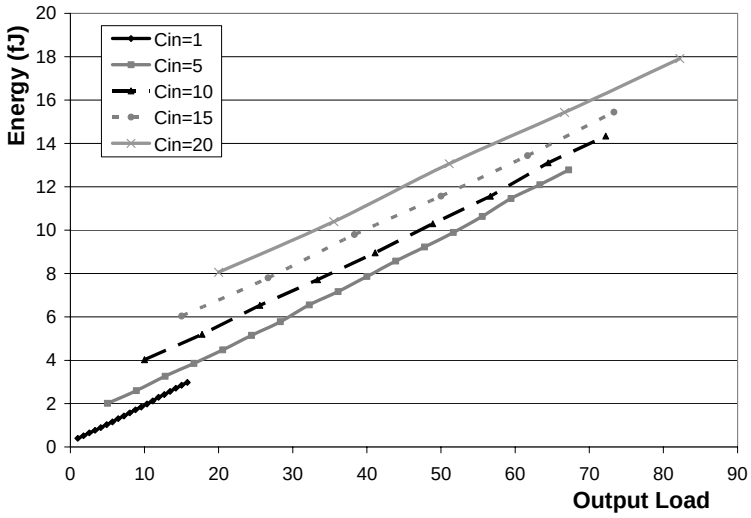

Figure 9-13. Energy Dependence on Input Size and Output Load for a 2-input NAND gate

From simulation an energy model which includes short circuit can be obtained. An offset occurs at zero size due to internal wire capacitance, which is accounted for in $E_{local\text{-}wire}$. The active energy associated with the output of a gate can then be expressed as:

$$E = E_{size} \cdot size + E_{cout} \cdot C_{out} + E_{local-wire}$$

E_{size} represents the energy per size and E_{cout} represents the energy per output load. Each of these terms is obtained from characterization. The switching activity of each gate can be used to estimate the energy of each gate in the design.

4.2 Minimal Energy Circuit Sizing for a Fixed Output Load and Fixed Input Size

To optimize a circuit with a fixed output load and input size it is first necessary to understand where the Logical Effort design point lies in the energy-delay space. The energy-delay space obtained through changing the sizes of the second in third inverters in a chain of 3 inverters with a fixed output load and input size is shown in Figure 9-14. As can be seen the solution space is extremely large even for such a simple circuit. In this solution space the delay optimized sizing (LE) serves as the limit for performance. Efficient design points in this solution space are those that achieve minimal energy for a fixed delay. These points are obtained by

relaxing the delay target from the LE point and resizing the circuit to reduce energy using techniques such as [14][15]. The combined result of these optimizations yields the minimal Energy-Delay curve of a circuit for a fixed output load and input size.

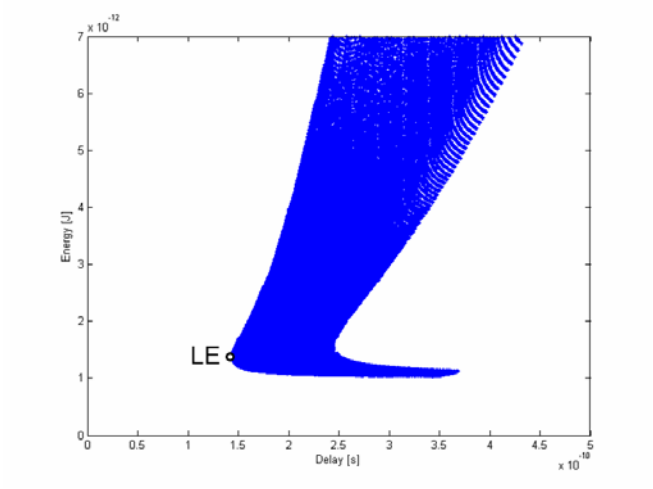

Figure 9-14. Energy-Delay Solution Space for a Fixed Input Size and Output Load

It has been suggested that the derivative of this curve can be used to select an efficient design point [9][10][16][17]. For high-performance, some popular metrics have been energy*Delay^2 (ED^2), Energy-Delay Product (EDP), and other ED^X metrics. The difficulty with designing for these products is that they can not be directly computed. In order to obtain them, the adjacent points A minimal energy-delay curve for a fixed output load and input size obtained using transistor sizing is shown in Figure 9-15 along with various design metrics.

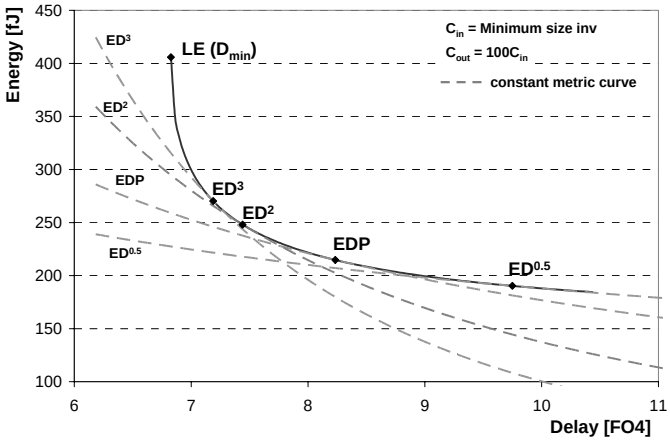

Figure 9-15. Minimal Energy-Delay Curve for an Inverter Chain with fixed C_{out} and C_{in}

The corresponding transistor sizing for each metric in Figure 9-16 are shown in Figure 9-16. Energy decreases dramatically from the LE point at only a slight increase in delay.

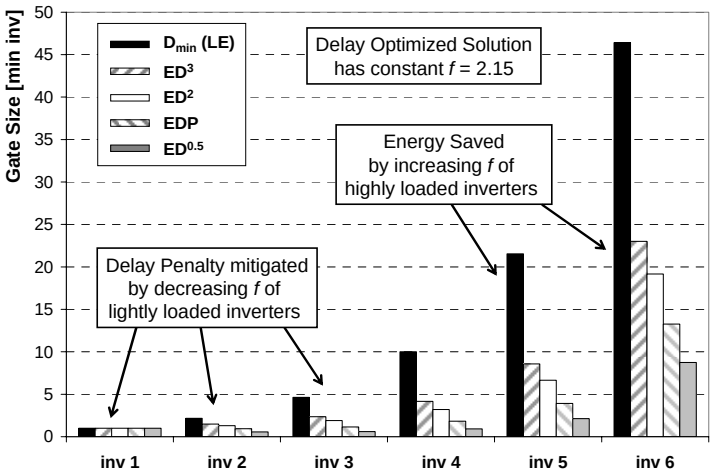

Figure 9-16. Corresponding Gate Sizing for Points on the Energy-Delay Curve

The rapid decent is due to the monotonic nature of transistor sizes along a path. This weighting of gates along a path can be seen in the computation of the input capacitance for each gate:

$$C_k = C_{load} \cdot \prod_{i=k}^{n} \frac{g_i}{f_i}$$

By changing the size of the inverter at the end of the path by a factor α (equivalent to changing f_N by $1/\alpha$), the size of each preceding gate along the path also decreases by a factor α. It is this weighting of gates that causes the sizing to differ dramatically from the LE solution of equal f. Clearly if the delay of the path can be relaxed, the excess delay should be redistributed into the gates which contributing the most energy to the path (by changing f of the gate).

4.3 Circuit Sizing for Minimal Energy with a Fixed Output Load and Variable Input Size

In practice circuit designers do not have the flexibility of degrading performance as it is directly tied to the performance of the entire system. As a result, metrics which relate delay variation to energy variation are inapplicable at the circuit level. The circuit should instead be designed for minimal energy. As shown previously, for a given delay, input size and output load there exists only one minimal energy solution. However, if the input size is allowed to change multiple energy solutions can be obtained at a fixed delay [14]. The solution space obtained by varying the input size of a static 64-bit Kogge-Stone Adder [20] is shown in Figure 9-17.

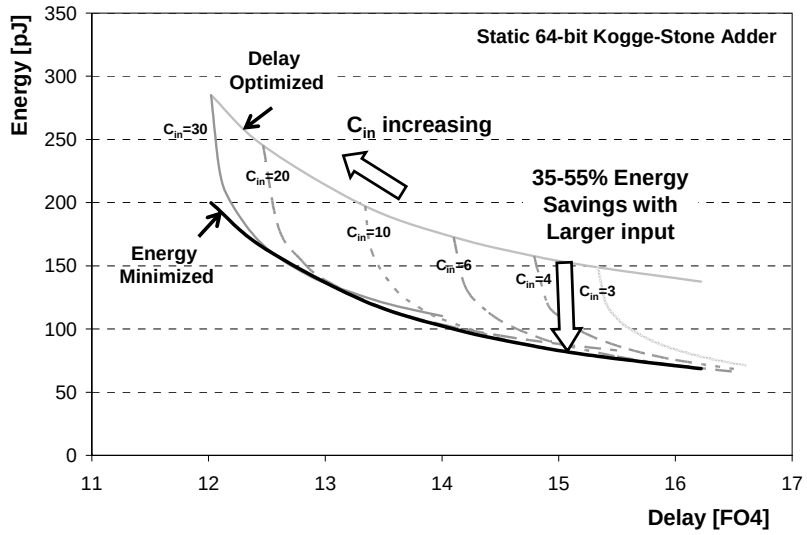

Figure 9-17. Energy-Delay Space for a Static 64-bit KS Adder with Fixed Output Load

The upper bound of the solution space consists of the delay optimized points obtained for each input size. The lower bound of the energy-delay space is constructed from the minimum energy points for each delay. Increasing the input size causes the H (C_{out} / C_{in}) of the circuit to be reduced, allowing for performance to improve at a cost in energy. By relaxing the delay target of larger input sizes from delay optimal a potential 35-55% energy savings is obtained for the adder example, depending on the delay target.

4.3.1 Example: Energy Minimization of an Inverter Chain

We will revisit the 6-inverter chain example, to demonstrate how gate sizes change to achieve the same delay for different input sizes. The minimal energy sizings are shown in Figure 9-18 for several input sizes, with the constraint of $C_{out} = 100C_{minimum}$ and a delay of 18.9τ. As the input size is increased it is easier for the design to achieve the desired delay. The excess delay is redistributed amongst the gates, by reducing the size of the gates that impact the energy the most and increasing the size of the gates that have a smaller impact on energy. An increase in input size by 20% (to 1.2) allows for a 22.3% reduction in energy. Further increases in input size yield savings at a diminishing rate.

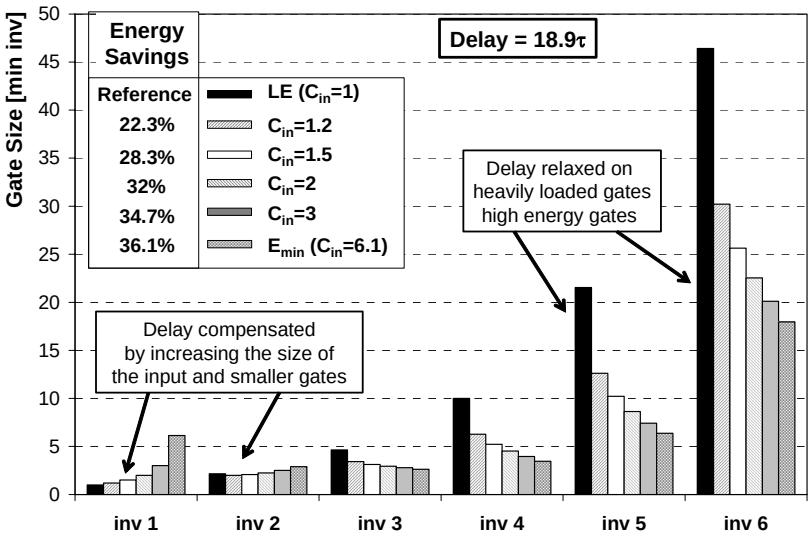

Figure 9-18. Gate Sizing of an Inverter Chain for Energy Reduction at a fixed Delay

4.3.2 Energy Minimization of Multi-Path Circuits

When optimizing circuits, the optimal solution occurs when the delay of each path from input to output is equalized [12][13][17]. When analyzing the energy of a circuit it is necessary to know the sizes of each gate in the circuit (not just those on the critical path). This further complicates the optimization process, as seen in the example of Figure 9-19. Path A and B must now be optimized to include the constraint of having equal delay to that of Path C.

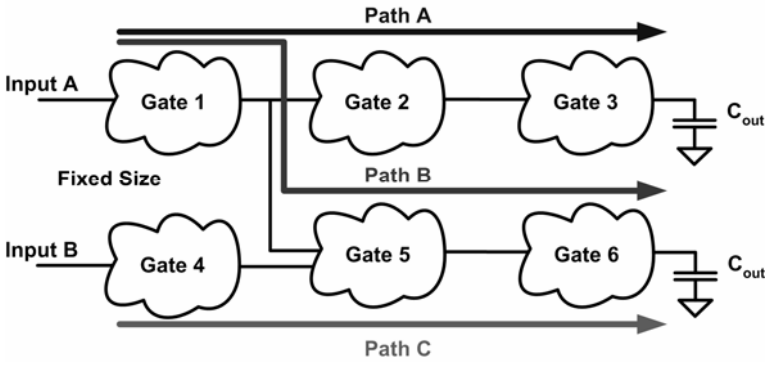

Figure 9-19. Multi-Path Circuit

The exact solution to this problem requires a numerical approach such as convex optimization [21], from which we can obtain little to no intuition. In [14] we presented a simplified approach to analyze the characteristics of every path in a circuit. In this approach each gate is assigned to a logic stage, with every gate in the logic stage sized to have the same delay. Gates are assigned to stages starting from the input and moving towards the output. If a path has fewer stages than another path, the last gate of the path is sized to include the combined delay of the additional stages of the longest path. Using this approach the delay of each path in the circuit is always equal, allowing for optimization to be performed at the stage level. The optimization has only N variables (where N is the number of stages) and can easily be performed using MATLAB or with Microsoft EXCEL's Solver.

5. DESIGNING ENERGY-EFFICIENT DIGITAL CIRCUITS

Designing energy-efficient digital circuits requires different guidelines than those used in Logical Effort. In LE a chain of inverters optimized for delay is used to demonstrate the relative insensitivity of design implementation to number of stages (Figure 9-20). While the delay is relatively insensitive around the optimal number of stages, energy is very sensitive to the number of stages. Inverter chains which contain more stages than delay optimal are always sub-optimal in terms of energy. This result is contrary to LE, and requires that the number of stages be carefully analyzed to obtain energy-efficient design.

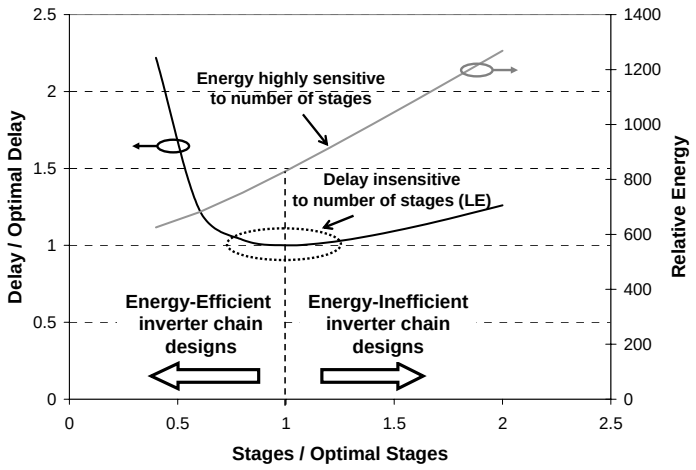

Figure 9-20. Optimal Number of Stages (inverters)

In Figure 9-21 the energy-delay curves of several inverter chains are shown, each with the same output load and input size. It is observed that the 5 and 6 stage designs are never energy-efficient, while the 2, 3 and 4 stage designs have regions of energy-efficiency depending on the desired operating target.

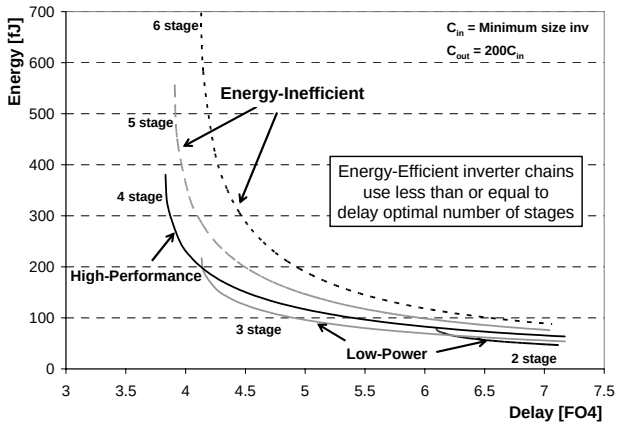

Figure 9-21. Optimal Number of Inverters for Fixed Input Size and Output Load with Varying Delay Target

Contrary to delay based optimization, the location of gates within a chain impacts the energy characteristics of a circuit. For example, the two chains

of gates in Figure 9-22 consist of the same circuits which results in the same delay, however the relative energy of each chain differs, from 66.5 to 94. Simpler gates (those with smaller g and p) such as the inverter in the example require much less energy to drive a load than more complex gates, by presenting a smaller input capacitance to the previous compared to more complex gates at the same delay. Whenever possible the simple gates should be placed in the most energy sensitive logic stage of the circuit.

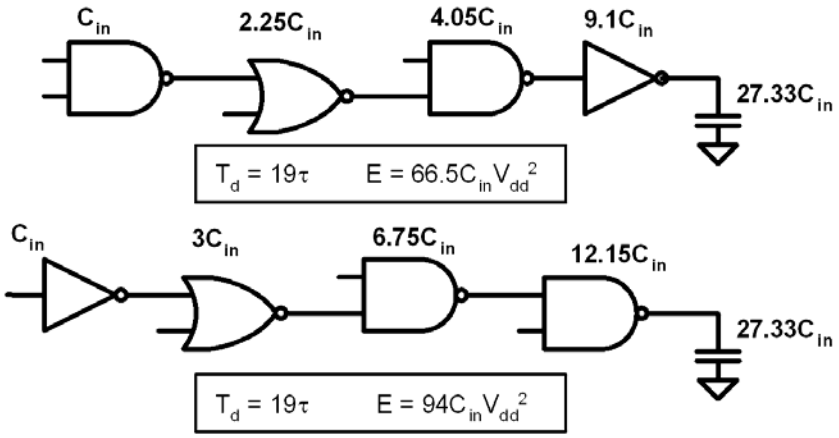

Figure 9-22. Energy Impact of Gate Placement in a Chain

The arrangement of circuits also has implications on the optimal number of stages. For example, if we examine the impact of adding inverters to the output of a 64-bit static KS adder (Figure 9-23). Energy savings are obtained if up to 3 inverters are added at the output, although each occurs at a degraded performance target. Despite there being more stages than delay optimal, the energy still decreases. This is because by adding simpler gates to the output, the size of the gates in the adder can decrease dramatically similar to the example in Figure 9-22. Therefore, despite paying a slight delay penalty due to an extra logic stage, the energy of the design can be decreased.

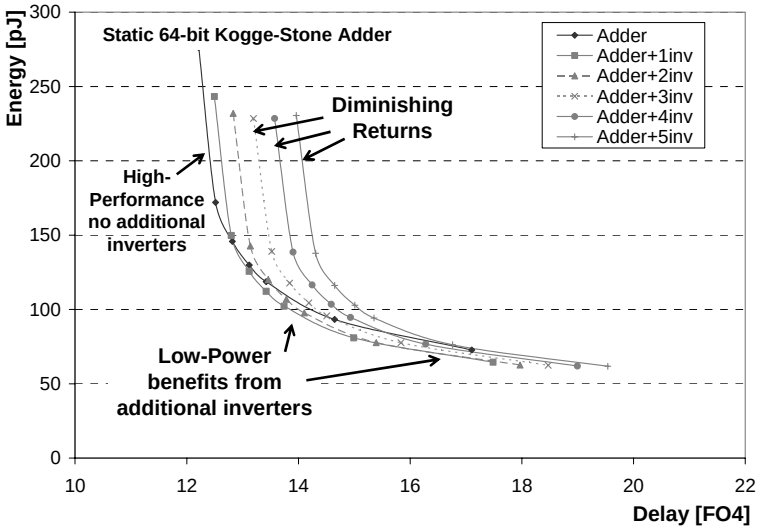

Figure 9-23. Impact of Buffering the Output of a 64-bit KS Adder

6. CHARACTERIZATION SETUP

To obtain absolute numbers for the energy and delay of a circuit in a given technology, it is necessary to characterize the circuit using simulation. The simulation setup is critical to the accuracy of the estimation. Ideally each circuit should be characterized under conditions identical to how it will operate, specifically with regards to the relative arrival and slope of input signals as well as the correct ratio of input to output loading. The most common configuration for calibrating RC-models for digital circuits analyzes the delay characteristics due to a single input switching as shown in Figure 9-20. Two logic stages are used to create a realistic input slope and the output is loaded with two gates in series to create realistic loading conditions.

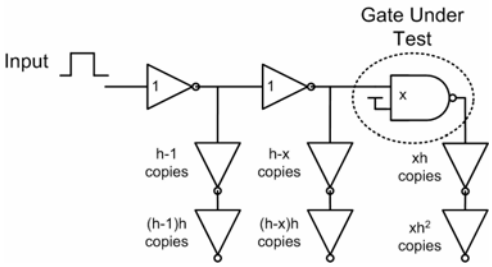

Figure 9-24. Single Input Switching Characterization Setup

Using this setup for the NAND gate the worst case t_{lh} is measured (only one pMOS transistor is on), while the best case t_{hl} is measured (one of the nMOS transistors is already on). The impact of relative arrival time of inputs A and B to a NAND gate is demonstrated in Figure 9-21. The hand estimates used in LE assume worst case conditions for both t_{lh} and t_{hl}. As seen in the figure only one of the worst case transitions is captured when using the test setup in Figure 9-20.

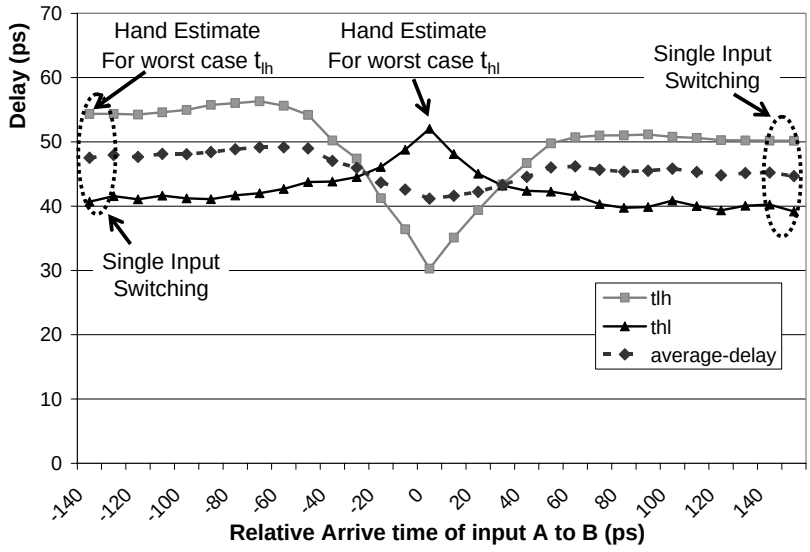

Figure 9-25. Impact of Relative Input Arrival Time on the Delay of a 2-input NAND gate

To capture both the worst case delay for t_{lh} and t_{hl} the characterization setup in Figure 9-22 should be used. Inputs A and B can be controlled independently and are loaded identically such that the signals can arrive at the same time to the gate under test. Using this characterization setup, values

for g and p can be extracted from simulation which closely match those of hand estimates. In practice the majority of static timing analyzers do not account for the impact of relative arrival time of input signals on the delay characteristics of gates [17]. The energy of gates can also be characterized using this setup. However energy characterization requires the additional step of characterizing a gate over multiple sizes to find the energy dependence on size (whereas in delay optimization the delay is constant for a fixed h).

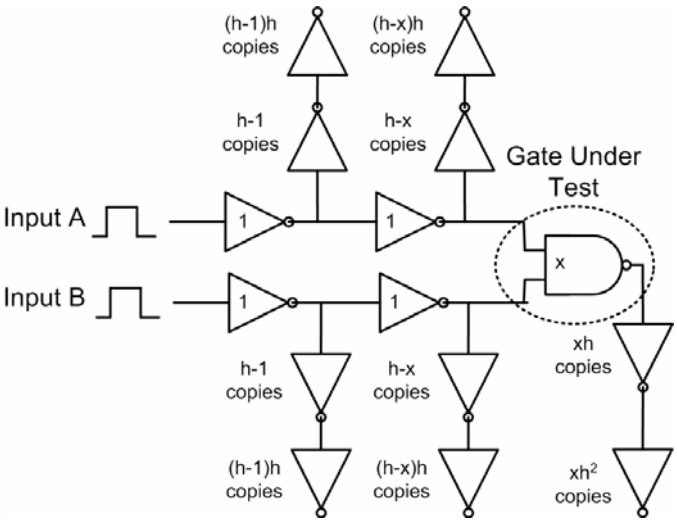

Figure 9-26. Multi-Input Switching Characterization Setup

7. CONCLUSION

The design of digital circuits in current and future technologies requires a thorough understanding of the energy-delay space. Design principles developed using Logical Effort no longer guarantee efficient designs when energy is considered. Instead of using equal stage effort for design, the stage efforts should be chosen to balance the energy-delay tradeoffs of each individual gate. To address this we have presented a technique for determining the delay optimal design for a fixed number of stages. Energy estimation techniques can be used to estimate the energy of these designs, which can be used in conjunction with RC-delay modeling to explore the entire energy-delay space of a design.

8. REFERENCES

[1] M. Horowitz, "Timing Models for MOS Circuits," PhD Thesis, Stanford University, December 1983.

[2] J. Rubenstein, P. Penfield, M. A. Horowitz, "Signal Delay in RC Networks," IEEE Transactions on Computer Aided Design, Vol. Cad-2, No.3, July 1983, pp. 202-211.

[3] D. Hodges, H.Jackson, "Analysis and Design of Digital Integrated Circuits," McGraw Hill 1988.

[4] T. Sakurai and A. R. Newton, "Alpha-Power Law MOSFET Model and Its Application to CMOS Inverter Delay and Other Formulas," IEEE Journal of Solid-State Circuits, Vol.25, No.2, pp.584-594, Apr. 1990

[5] N. Weste, K.Eshraghian, "Principles of CMOS VLSI Design A Systems Perspective," Addison Wesley, 1992.

[6] I.E. Sutherland and R. F. Sproull, "Logical Effort: Designing for Speed on the Back of an Envelope," Advanced Research in VLSI, Proceedings of the 1991 University of California, Santa Cruz, Conference, C.H. Sequin, ed., MIT Press, 1991. pp. 1-16

[7] I.E. Sutherland, R.F. Sproull , and D. Harris, "Logical Effort Designing Fast CMOS Circuits," Morgan Kaufmann Pub., 1999.

[8] V. G. Oklobdzija and E. R. Barnes, "On Implementing Addition in VLSI Technology," IEEE Journal of Parallel and Distributed Computing, No. 5, 1988 pp. 716-728.

[9] V. Zyuban, and P. Strenski, "Unified Methodology for Resolving Power-Performance Tradeoffs at the Micro-achitectural and Circuit Levels", IEEE Symposium on Low Power Electronics and Design, 2002.

[10] V. Zyuban and P. Strenski, "Balancing Hardware Intensity in Microprocessor Pipelines," IBM Journal of Research and Development, Vol. 47, No. 5/6, September/November 2003.

[11] V. G. Oklobdžija, B. R. Zeydel, H. Q. Dao, S. Mathew, R. Krishnamurthy, "Comparison of High-Performance VLSI Adders in Energy-Delay Space", IEEE Transaction on VLSI Systems, Volume 13, Issue 6, pp. 754-758, June 2005.

[12] N. Hedenstierna and K.O. Jeppson, "CMOS Circuit Speed and Buffer Optimization," IEEE Trans. on CAD, vol. CAD-6, no. 2, pp. 270-281, March 1987.

[13] P. Fishburn, A.E. Dunlop, "TILOS: A Posynomial Programming Approach to Transistor Sizing," International Conference on Computer Aided Design, pp. 326-328, Nov. 1985.

[14] V. Sundararajan, S. S. Sapatnekar, and K. K. Parhi, "Fast and Exact Transistor Sizing Based on Iterative Relaxation," IEEE Transactions on Computer Aided Design of Circuits and Systems, Vol. 21, No. 5, pp. 568-581, May 2002.

[15] H.Q.Dao, B.R.Zeydel, V.G.Oklobdzija, "Energy Minimization Method for Optimal Energy-Delay Extraction", European Solid-State Circuits Conference, Estoril, Portugal, Sep. 16-18, 2003

[16] V. Stojanovic, D. Markovic, B. Nikolic, M.A. Horowitz, R.W. Brodersen, "Energy-Delay Tradeoffs in Combinational Logic using Gate Sizing and Supply Voltage Optimization," Proceedings of the 28th European Solid-State Circuits Conference, ESSCIRC'2002, Florence, Italy, September 24-26, 2002. pp. 211-214.

[17] D. Marković, V. Stojanović, B. Nikolić, M.A. Horowitz, and R.W. Brodersen, "Methods for True Energy-Performance Optimization," IEEE J. Solid-State Circuits, vol. 39, no. 8, pp. 1282-1293, Aug. 2004.

[18] Sachin Sapatnekar, "Timing,"

[19] Bernard Meyerson, "How does one define "Technology" Now That Classical Scaling is Dead?", Keynote presentation, 42nd annual Design Automation Conference, Anaheim, CA, June 2005.

[20] P.M. Kogge and H.S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations", *IEEE Trans. Computers Vol. C-22, No. 8, Aug. 1973, pp.786-793*.

[21] S. Boyd and L. Vandenberghe, "Convex Optimization," Cambridge University Press, 2004.

[22] Y. Taur, "CMOS design near the limit of scaling," IBM Journal of Research and Development, Vol. 46, Num. 2/3, Mar/May 2002.

[23] International Technology Roadmap for Semiconductors, public.itrs.net.