Tradeoffs between Gate Oxide Leakage and Delay for Dual T_{ox} Circuits[*]

Anup Kumar Sultania
Department of ECE
University of Minnesota
Minneapolis, MN 55455.
anups@ece.umn.edu

Dennis Sylvester
Department of EECS
University of Michigan
Ann Arbor, MI 48109.
dennis@eecs.umich.edu

Sachin S. Sapatnekar
Department of ECE
University of Minnesota
Minneapolis, MN 55455.
sachin@ece.umn.edu

ABSTRACT

Gate oxide tunneling current (I_{gate}) will become the dominant component of leakage in CMOS circuits as the physical oxide thickness (T_{ox}) goes below 15Å. Increasing the value of T_{ox} reduces the leakage at the expense of an increase in delay, and a practical tradeoff between delay and leakage can be achieved by assigning one of the two permissible T_{ox} values to each transistor. In this paper, we propose an algorithm for dual T_{ox} assignment to optimize the total leakage power under delay constraints, and generate a leakage/delay tradeoff curve. As compared to the case where all transistors are set to low T_{ox}, our approach achieves an average leakage reduction of 83% under 100nm models.

Categories and Subject Descriptors

B.7.2 [**Hardware**]: Integrated Circuits—*design aids*

General Terms

Performance, Algorithms

Keywords

Leakage power, Dual T_{ox} Circuits

1. INTRODUCTION

Leakage current is a primary concern for low power, high performance digital CMOS circuits for portable applications, and industry trends show that leakage will be the dominant component of power in future technologies. New leakage mechanisms, such as tunneling across thin gate oxides, which lead to gate oxide leakage current (I_{gate}), are coming into play from the 90nm node onwards.

According to the International Technological Roadmap for Semiconductors (ITRS) [1], physical oxide thickness (T_{ox}) values of 7–12Å will be required for high performance CMOS circuits by 2006, and quantum effects that cause tunneling will play a dominant role in such ultra-thin oxide devices. The probability of electron tunneling is a strong function of the barrier height (i.e., the voltage drop across gate oxide) and the barrier thickness, which is simply T_{ox}, and a small change in T_{ox} can have a tremendous impact on I_{gate}. For example, in MOS devices with SiO_2 gate oxides, a difference in T_{ox} of only 2Å can result in an order of magnitude increase in I_{gate} [2], so that reducing T_{ox} from 18Å to 12Å increases I_{gate} by approximately 1000×. The other component of leakage, subthreshold leakage (I_{sub}), forms a reducing fraction of the total leakage as T_{ox} is reduced, so that I_{gate} will become the dominant leakage mechanism in the future. The most effective way to control I_{gate} would be through the use of high-k dielectrics, but such materials are not expected before the 65nm technology node in 2007, at the earliest.

This paper will explore the use of dual T_{ox} values for performance optimization. Although this optimization can be exploited at a number of points in the design methodology, our solution considers T_{ox} assignment as a step that is performed after placement and transistor sizing, at which point it is used to achieve a final performance improvement. Unlike earlier stages of design, there is less design uncertainty at this point and minor changes in layout parasitics due to T_{ox} assignment can be dealt with an incremental update. As a result, all of the delay gains from our procedure can be guaranteed in the final design, with a low leakage power overhead.

Leakage power can be broadly divided into two categories: *standby leakage*, which corresponds to the situation when the circuit is in a non-operating or sleep mode, and *active leakage*, which relates to leakage during normal operation. Numerous effective techniques for controlling standby leakage have been proposed in the past, including state assignment [3], the use of multiple threshold CMOS (MTCMOS) sleep transistors [4], body-biasing [5], and dual T_{ox} combined with state assignment. Active leakage, however, has not been addressed very widely in the literature so far, primarily because it has not been a major issue in the present technologies. However, leakage power dissipation in the active mode has grown to over 40% in some high-end parts today [6]. Therefore, reducing active leakage is vital for advanced technologies in current-generation circuits, and for next-generation technologies. The range of options that are available for reducing active leakage is considerably more limited than for standby leakage, and the use of dual T_{ox} assignments is a powerful method for this purpose.

Prior research related to our work is summarized as follows. In [7], the authors examine the interaction between I_{gate} and I_{sub}, and their state dependencies. This work ap-

[*]This work was supported in part by the SRC under contract 2003-TJ-1092, and by the NSF under award CCR-0205227.

DAC 2004, June 7–11, 2004, San Diego, California, USA.
Copyright 2004 ACM 1-58113-828-8/04/0006 ...$5.00.

plies pin reordering to minimize I_{gate}. The impact of I_{gate} on delay is discussed in [8], but its impact on leakage power is not addressed. The work in [9] presents an approach to reducing I_{sub}, but not I_{gate}, using separate optimizations to select the values of T_{ox}.

In our context, where we optimize the total leakage, comprising both I_{gate} and I_{sub}, the rationale for optimizing T_{ox} is as follows. Choosing a lower value of T_{ox} can result in lower delays, but at the cost of increased leakage, and the value of T_{ox} can therefore be optimized to obtain a leakage/delay tradeoff. For practical reasons, it is important to scale the effective channel length L_{eff} along with T_{ox} [10].

Due to process constraints, rather than an unlimited range of T_{ox} values, it is more reasonable to choose between two permissible values. In Section 2, we describe a method for selecting appropriate values of the low and high values of the oxide thickness, referred to as $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$, respectively, and the corresponding values for the channel length. Next, in Sections 3 and 4, respectively, we introduce the leakage and delay models that are used in this work, and demonstrate that they show a good degree of accuracy as compared to simulation results. Our iterative algorithm for finding the leakage/delay tradeoff is then presented in Section 5, followed by a description of our experimental results in Section 6 and concluding remarks in Section 7.

2. CHOOSING T_{ox} AND L_{eff}

While an increased value of T_{ox} succeeds in significantly reducing I_{gate}, several other physical effects must be taken into consideration. Increasing the value of T_{ox} while keeping the channel length constant may adversely impact the functionality of the transistor. Specifically, due to drain induced barrier lowering (DIBL), an increase in T_{ox} may result in a situation where the drain terminal takes control of the channel, so that the "on" or "off" state of the transistor is no longer completely governed by the gate terminal.

This effect has been recognized during technology scaling, and scaling trends have shown that T_{ox} reduces nearly in proportion with L_{eff} [11]. We maintain this proportion for each of the chosen values of T_{ox} by setting

$$\frac{L_{eff}@T_{ox_{Lo}}}{T_{ox,e_{Lo}}} = \frac{L_{eff}@T_{ox_{Hi}}}{T_{ox,e_{Hi}}} \tag{1}$$

The term $T_{ox,e}$ in this equation refers to the *electrical* T_{ox}, which is related to the *physical* value of T_{ox} as follows[1]

$$T_{ox,e} = T_{ox} + T_{ox_{\text{offset}}} \tag{2}$$

The $T_{ox_{\text{offset}}}$ term is added to account for the gate depletion and channel quantization effects, and a typical value is 0.7nm [12]. In the remainder of this paper, it will be implicit that as we change T_{ox}, the value of L_{eff} will also be scaled.

Before determining reasonable values for $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$, we will study the effect of varying T_{ox} on leakage for an inverter. The gate oxide leakage, I_{gate}, and the subthreshold leakage, I_{sub}, for both the NMOS and PMOS transistors in the inverter, are graphically depicted in Figure 1(a) for various values of $T_{ox_{Hi}}$, at $T_{ox_{Lo}} = 12$Å; the sum of these components is shown by the bottommost curve in Figure 1(b). The values of I_{sub} are obtained through SPICE simulations on predictive technology models [13], and an analytical model (described in Section 3.2) is used to generate I_{gate}[2]. The average leakage of the inverter is calculated as the sum of the average I_{gate} and I_{sub} leakages (as described in greater detail in Section 3), and is shown in Figure 1(b).

As T_{ox} is varied, I_{sub} shows a negligible change in comparison to I_{gate}. Furthermore, the average leakage decreases slowly for $T_{ox} > 17$Å, and increases sharply as T_{ox} goes below 17Å. On the other hand, the delay of the inverter (as will be seen by the experiment in Figure 2) increases linearly with T_{ox}, so that using a value of $T_{ox_{Hi}}$ of over 17Å results in a larger delay with no appreciable savings in the leakage. This leads us to choose $T_{ox_{Hi}} = 17$Å.

To choose $T_{ox_{Lo}}$, we consider several scenarios as shown by the plots in Figure 1(b). Each curve corresponds to a different choice of $T_{ox_{Lo}}$, and the value of L_{eff} is set to 60nm at this value. Each point on a curve now shows the total leakage for an inverter whose transistors are set to a candidate value of $T_{ox_{Hi}}$. For instance, for the curve where $T_{ox_{Lo}} = 15$Å, candidate values for Tox_{Hi} range from 28Å to 15Å, and the L_{eff} value for each case is scaled in accordance with Equation (1). It is easily seen that on each curve, the T_{ox} value at which the leakage begins to change at exponential rate is about 17Å. In other words, for the entire range of $T_{ox_{Lo}}$ candidate values of 12Å through 15Å, it is clear that our choice of $T_{ox_{Hi}}$=17Å is reasonable. For a wider range of delay values in the tradeoff curve, the difference in $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$ should be as high as possible. The choice of $T_{ox_{Lo}}$, however, is limited by I_{gate}/I_{sub} ratio. This ratio, at $T_{ox_{Lo}}$, should be such that I_{gate} does not completely dwarf I_{sub}. Furthermore, due to process variation in T_{ox}, the choice of $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$ should be such that their probability distribution functions do not overlap. We choose $T_{ox_{Lo}} = 12$Å as it gives the best achievable leakage/delay tradeoff.

We now consider the impact of changing T_{ox} and L_{eff} on the gate capacitance, C_{inv}, and the threshold voltage, V_{th}, of the MOS devices; each of these parameters clearly depends on T_{ox} and L_{eff}. We perform a set of SPICE simulations on a circuit set-up illustrated in Figure 2. In this experiment, the T_{ox} value of Inverter 2 is varied, and all other inverters are maintained at a fixed T_{ox} value of 17Å. The results are shown in the table in the same figure, and lead to a happy coincidence. Our method of scaling the value of L_{eff} linearly with T_{ox} results in a *nearly constant* values of C_{inv} and V_{th}, respectively. However, there is a noticeable impact on the gate delay: increasing T_{ox} and L_{eff} decreases the channel transconductance, and hence increases the delays. Changing T_{ox} from 12Å to 22Å alters the delays *linearly*, with a delay penalty of 51% over this range for Inverter 2.

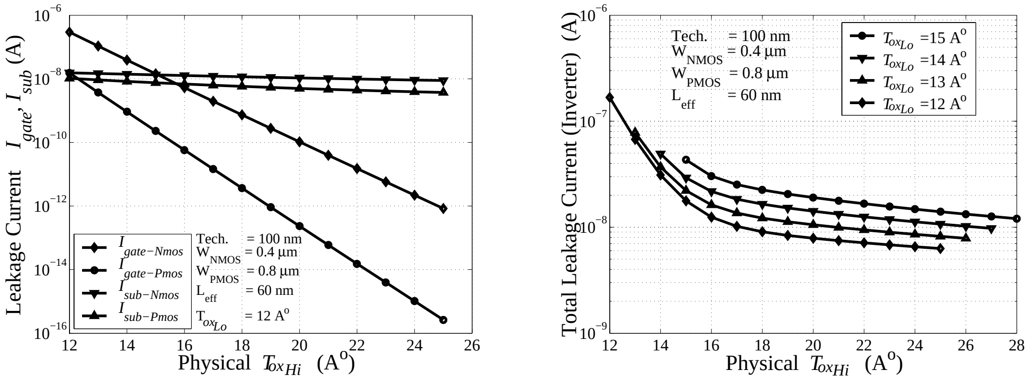
The invariance of the capacitance of Inverter 2 over the entire range of T_{ox} has two notable consequences:

- A change in T_{ox} of a transistor leaves the load capacitance presented to the previous stage of logic unchanged. As a result, the delay of a fanin logic gate does not change significantly, and hence our optimization method needs only to consider the delay change of a given logic gate when its T_{ox} is altered.

- Since the capacitance is unchanged, the $CV_{dd}^2 f$ (dynamic) power remains unaffected by changes in T_{ox}. This is extremely important since our optimization targets the active mode of operation.

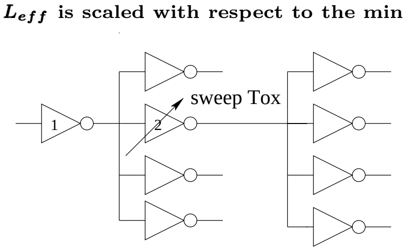
[1] Henceforth, our discussions will be with reference to T_{ox}, the *physical* value of the gate oxide thickness.

[2] We cannot use simulations here since the Berkeley predictive technology model [13] uses BSIM3, which does not model I_{gate}.



(a) (b)

Figure 1: (a) The four leakage components for an inverter (I_{gate} and I_{sub} for the NMOS and PMOS transistors, respectively) as a function of the gate oxide thickness. (b) The total leakage of an inverter for different values of $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$. At each point, L_{eff} is scaled with respect to the minimum T_{ox} value on the curve; at this point, $L_{eff} = 60$nm.



T_{ox} (Å)	L_{eff} (nm)	D_{Inv1} (ps)	D_{Inv2} (ps)	C_{inv} (fF)	V_{th} (V)
12	60.0	33.84	33.56	1.98	0.119
14	66.3	33.77	36.70	1.99	0.120
16	72.6	33.71	39.98	1.99	0.122
18	78.9	33.67	43.40	1.99	0.124
20	85.2	33.64	46.97	2.00	0.126
22	91.6	33.62	50.69	2.00	0.127

Figure 2: The effect of varying T_{ox} of Inverter 2 on (i) its delay and on delay of its fanin gate, Inverter1 (ii) the input capacitance of the inverter C_{inv} (calculated as the sum of the NMOS and PMOS gate capacitances), and (iii) the threshold voltage V_{th} of the NMOS device. The transistor widths are chosen as $W_n = 0.4\mu m$ and $W_p = 0.8\mu m$.

3. LEAKAGE MODELS

We will now describe the models used to calculate I_{sub} and I_{gate} for each transistor, and the approach for computing the average I_{sub} and I_{gate} values for a given logic gate. The total leakage current for a logic gate is then computed as the sum of its corresponding average I_{sub} and I_{gate}.

3.1 Subthreshold Leakage Model

As seen in the table in Figure 2, the value of V_{th} changes by a very small amount as T_{ox} is changed. In spite of this, it can have significant effects on I_{sub}, which is exponentially dependent on V_{th}. For convenience, we use a simple look-up table (LUT) to determine I_{sub}. Conceptually, such an LUT could be extremely large: for a k-input NAND gate, for instance, we would store the leakage current for each of the 2^k possible T_{ox} assignments[3], and each T_{ox} assignment would require entries for the $2^k - 1$ leakage states corresponding to different input logic values[4], resulting in a total of $2^k \cdot (2^k - 1)$ entries. The LUT size can be reduced significantly using the following ideas:

Dominant input states: It has been shown [14] that I_{sub} can be accurately captured by using a set of dominant states, corresponding to the cases where only one transistor on each path to a supply node is on.

***T_{ox}* dependencies:** In a dominant state, for a given T_{ox} choice for the leaking transistor the subthreshold leakage is only weakly dependent on the T_{ox} values of other transistors. Intuitively, this relates to the fact that the leaking transistor is the largest resistance on the path. We have validated this through SPICE simulations, and the results for a 4-input NAND gate are shown in Figure 3. When T_4 is the leaking transistor and is set to $T_{ox_{Lo}}$, it can be seen that I_{sub} has a range of only about 1% over all possible assignments for the other inputs. Similar results are seen for other logic gates over various T_{ox} assignments.

For a k-input NAND gate, there are k dominant states. The weak T_{ox} dependencies require that for each of these states, two I_{sub} numbers must be maintained: one at $T_{ox_{Hi}}$ and one at $T_{ox_{Lo}}$. As a result, the LUT size comes down to $2k$.

For a logic gate with k-parallel transistors (such as the pull-up in a k-input NAND, or a pull-down in a k-input NOR), 2 entries ($T_{ox_{Hi}}$ and $T_{ox_{Lo}}$) are sufficient as the value of I_{sub} per unit $\frac{w}{l}$ for each parallel branch is almost equal.

The average subthreshold leakage ($I_{sub_{avg}}$) for a logic gate under a given T_{ox} assignment may therefore be calculated as follows:

$$I_{sub,avg.} = \textstyle\sum_{i\ \in\ \text{dominant input states}} P_i \times I_{sub_i} \qquad (3)$$

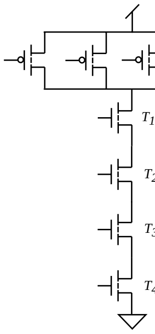
where P_i is the probability of occurrence of state i, and I_{sub_i} is the subthreshold leakage current in that state.

3.2 Gate Oxide Tunneling Model

Gate oxide leakage can be primarily attributed to electron [hole] tunneling in NMOS [PMOS] devices. Physically, this tunneling occurs in the gate-to-channel region, and in the gate-to-drain/source overlap regions. The latter type

[3] Series-connected devices can have different T_{ox} provided they are spaced out a little more than the design rules indicate.

[4] The only input assignment with no leakage due to NMOS is the case when all transistors in the pull-down chain are on.



T_{ox}				I_{sub}
T_1	T_2	T_3	T_4	(nA)
Lo	*Lo*	*Lo*	*Lo*	34.70
Lo	*Lo*	*Hi*	*Lo*	34.83
Lo	*Hi*	*Lo*	*Lo*	34.85
Lo	*Hi*	*Hi*	*Lo*	34.99
Hi	*Lo*	*Lo*	*Lo*	34.78
Hi	*Lo*	*Hi*	*Lo*	34.92
Hi	*Hi*	*Lo*	*Lo*	34.93
Hi	*Hi*	*Hi*	*Lo*	35.08

Figure 3: The variation of I_{sub} through the pull-down chain for the dominant state when only T_4 is off. Here, $T_{ox_{Lo}} = 12$Å(*Lo*), $T_{ox_{Hi}} = 17$Å(*Hi*), and T_4 is at $T_{ox_{Lo}}$.

of tunneling, referred to as edge direct tunneling (EDT) is ignored in our case for two reasons: firstly, because the gate-to-drain/source overlap region is significantly smaller than the channel region, and secondly, because the oxide thickness in this overlap region can be increased after gate patterning to further suppress EDT [15].

Our work focuses on gate-to-channel tunneling, and we use the following analytic tunneling current density (J_{tunnel}) model based on the electron [hole] tunneling probability through a barrier height (E_B) [16].

$$\begin{aligned} J_{tunnel} &= \frac{4\pi m^* q}{h^3}(kT)^2(1+\frac{\gamma kT}{2\sqrt{E_B}}) \times \\ & \exp(\frac{E_{F0,Si/SiO_2}}{kT})\exp(-\gamma\sqrt{E_B}) \end{aligned} \qquad (4)$$

where $E_{F0,Si/SiO_2}$ is the Fermi level at the Si/SiO$_2$ interface and m^* is $0.19M_o$ for electron tunneling and $0.55M_o$ for hole tunneling, where M_o is the electron rest mass. The terms k, h and q correspond to physical constants (respectively, Boltzmann's constant, Planck's constant and the charge on an electron), $\gamma = 4\pi T_{ox}\sqrt{2M_{ox}}/h$ where M_{ox} is the effective electron [hole] mass in the oxide, T is the operating temperature, and E_B is the barrier height.

It was shown in [7] that like I_{sub}, I_{gate} also exhibits state dependency. When the gate node of the transistor is at logic 0, the only possible tunneling component is EDT, which is neglected in our work; therefore, we will only consider the cases where the gate node is at logic 1. For example, while determining I_{gate} for transistor T_2 in the 4-input NAND gate in Figure 3, it can be shown that the maximum leakage for T_2 occurs at the input state[5] $(x, 1, 1, 1)$, and that the I_{gate} values for the states $(1, 1, 0, x)$, $(0, 1, 0, x)$ and $(x, 1, 1, 0)$ can be ignored. For details, the reader is referred to [7].

In general, this may be restated as follows: the dominant state for I_{gate} for a particular transistor in a stack corresponds to the case when all of the transistors below (above) it in the NMOS (PMOS) stack are on. The average I_{gate} for a logic gate can then be calculated as:

$$I_{gate,avg.} = \textstyle\sum_{\text{transistor } i \,\in\, \text{logic gate}} P_i \times I_{gate_i} \qquad (5)$$

Here, P_i for NMOS [PMOS] transistors connected in parallel, as in a NOR [NAND] gate, is the probability that the input is at logic 1 [0]. For a stack of NMOS [PMOS] transistors in series in a NAND [NOR] gate, P_i for a transistor is the product of the probabilities that each of the transistors below [above] it have an input of logic 1 [0]. The value of I_{gate} is computed using Equation (4) for the specified L_{eff} and width of the transistor under consideration.

	T_{ox}				Delay		
	T_1	T_2	T_3	T_4	Spice	LUT	Error
D_0	*Lo*	*Lo*	*Lo*	*Lo*	13.89	—	—
D_1	*Lo*	*Lo*	*Lo*	*Hi*	14.84	14.51	-2.22 %
D_2	*Lo*	*Lo*	*Hi*	*Lo*	14.21	14.51	2.11 %
D_3	*Hi*	*Lo*	*Lo*	*Lo*	14.54	14.51	-0.21 %
D_4	*Lo*	*Lo*	*Hi*	*Hi*	15.11	15.13	0.13 %
D_5	*Hi*	*Lo*	*Lo*	*Hi*	15.47	15.13	-2.20 %
D_6	*Hi*	*Lo*	*Hi*	*Lo*	14.86	15.13	1.82 %
D_7	*Hi*	*Lo*	*Hi*	*Hi*	15.75	—	—

Table 1: Delays from the input of switching transistor T_2 in a 4-input NAND [Figure 3] @$T_{ox_{Lo}}$ ($T_{ox_{Lo}} = 12$Å, $T_{ox_{Hi}} = 17$Å).

Observe that the use of dominant states for the computation of I_{gate} and I_{sub} automatically rules out the complex interaction between these two components, as in [7].

4. DELAY MODEL

For advanced nanometer technologies, it is difficult to obtain accurate closed-form delay models, and therefore, we use an LUT-based approach for the delay. For each input of the logic gate, rise and fall delay values are determined through SPICE simulations over a range of output loads under a single-input switching model. A linear fit is carried out on these data to obtain the slope (delay/load) and intercept (delay at zero load) values. The LUT stores these two numbers for each input, along with gate input capacitance for each logic gate. The output load for a logic gate can be computed by summing up the input gate capacitance of the fanout logic gates. Based on this load, the delay of the logic gate is calculated as:

$$Delay = Intercept + Slope \times Load \qquad (6)$$

Different combinations of T_{ox} in a stack of transistors will result in different input-to-output delays for the same input; for example, for a k-input NAND gate, 2^k entries would be required to compute the fall delay from each input to the output, for a total of $k \cdot 2^k$ entries in the LUT. This LUT size may be greatly reduced for a small loss in accuracy.

For the output fall transition, for each input-to-output delay, we create two LUT's, corresponding to a gate oxide thickness assignment of $T_{ox_{Lo}}$ and $T_{ox_{Hi}}$, respectively; similarly, two LUT's are constructed for the rise transition. In each LUT, we observe that the delay depends strongly on the *number* of transistors in the chain that are at $T_{ox_{Lo}}$ or $T_{ox_{Hi}}$, and very weakly on their position. This is illustrated for a 4-input NAND gate in Table 1 for the delay from the input of T_4 to the output. We fit a simple formula as follows:

$$Delay = D_0 + n \times \frac{(D_7 - D_0)}{3} \qquad (7)$$

where D_0 and D_7 are delay values (stored in the LUT) for the extreme cases of non-switching transistors being at all $T_{ox_{Lo}}$ and all $T_{ox_{Hi}}$, respectively, as shown in Table 1, and n is the number of transistors (other than the switching transistor) at $T_{ox_{Hi}}$. The errors under this method are shown in Table 1. Therefore, all possible fall delay scenarios for a k-input NAND gate can be compacted into $4k$ LUT entries. This technique was applied to several gate types, and in most cases, the error was under 2%, with a worst-case error of 3%.

[5] "State" = logic values at the inputs to (T_1, T_2, T_3, T_4).

A similar compression for the case of output rise LUT's of a k-input NAND is possible. Since the PMOS transistors are in parallel, only the gate-to-drain overlap capacitance at the output node changes for different T_{ox} combinations for the transistors; this has an insignificant impact on the delay, and hence, $2k$ LUT entries (corresponding to $T_{ox_{Hi}}$ and $T_{ox_{Lo}}$ for each PMOS input) are sufficient.

A similar approach can be applied to build LUT's for a k-input NOR gate, and for other types of logic gates. Therefore, the total number of LUT entries varies linearly with the number of inputs to the logic gate. The input transition time can be accounted for in this model by creating one such LUT for each candidate transition time.

5. DUAL T_{ox} ASSIGNMENT

We use a TILOS-like [17] sensitivity-based heuristic for assigning T_{ox} values to individual transistors in a circuit. Starting with all transistors at $T_{ox_{Hi}}$, the heuristic (Algorithm 1) performs a static timing analysis step. Next, it greedily identifies the transistor on the critical path that, when changed to $T_{ox_{Lo}}$, would cause the largest delay reduction for the smallest increase in leakage. These two steps iterate until no further improvement is possible, and a leakage-delay tradeoff curve is thus obtained.

A standard static timing analysis (STA) approach is used to find the critical path. The propagation delay D_p for each gate is computed using the LUT described in Section 4. In principle, the STA must be repeated after each T_{ox} change; however, we observe that every such T_{ox} change is local and only changes delays and arrival times in its transitive fanout region. Therefore, after the first iteration, we achieve efficiency by performing incremental STA that processes only the affected regions.

```
{Circuit is represented as an acyclic graph $G(V,E)$}
{The target delay is $D_T$}
Initialize all transistors to $T_{ox_{Hi}}$
Propagate state probabilities from PI's to internal nodes
for each node $x \epsilon G(V,E)$ do
  Find output load = $\sum_{\text{fanout nodes}}$ gate capacitance
  Get rise, fall delays ($D_{P_{fall}}$, $D_{P_{rise}}$) from delay LUT
  Find $I_{sub}$, $I_{gate}$ based on LUT's
end for
Perform STA to find rise and fall $AT$, $RT$ for each node
                                   and circuit delay, $D_{max}$
while $D_{max} > D_T$ do
  $(\frac{\triangle D}{\triangle Lkg})_{worst} = 0$; $N_{chosen}$ = NULL;
  for each node $y$ on a critical path do
    if (critical path transistor(s) of $y == T_{ox_{Hi}}$) then
      find $(\frac{\triangle D}{\triangle Lkg})_y$ for node $y$
      if $(\frac{\triangle D}{\triangle Lkg})_{worst} > (\frac{\triangle D}{\triangle Lkg})_y$ then
        $(\frac{\triangle D}{\triangle Lkg})_{worst} = (\frac{\triangle D}{\triangle Lkg})_y$; $N_{chosen} = y$
        {Tie-breakers: #fanouts, proximity to PI}
      end if
    end if
  end for
  if $(\frac{\triangle D}{\triangle Lkg})_{worst} \neq 0$ then
    Assign $T_{ox_{Lo}}$ to the worst transistor in $N_{chosen}$
    Update $D_{P_{fall}}$, $D_{P_{rise}}$, $I_{sub}$, $I_{gate}$ of $N_{chosen}$
    Perform Incremental STA and recalculate $D_{max}$
  else
    Report $D_{max}$; Exit()
  end if
end while
```

Algorithm 1: Pseudocode for Dual T_{ox} Assignment

Once this critical path is found, the core of the optimizer iteratively changes one transistor on this path from $T_{ox_{Hi}}$ to $T_{ox_{Lo}}$ in each iteration. This transistor is identified by measuring the increase in the total average leakage, $\triangle Lkg$, with respect to the delay reduction, $\triangle D$, of the critical path when such a change is made. In other words, we evaluate

$$Cost = \frac{\triangle D}{\triangle Lkg} \qquad (8)$$

The transistor with the minimum (most negative) cost provides the largest delay reduction for the smallest increase in leakage power, and is selected for assignment to $T_{ox_{Lo}}$. The corresponding L_{eff} is also concurrently changed. If two transistors have the same cost, ties are heuristically broken, first by selecting the transistor with the higher fanout, and if that fails, then by choosing the one that is closer to the PI (since it has a larger cone of influence, and is likely to reduce the delay on a larger number of paths).

In evaluating $\triangle D$, it is enough to find the delay change of the logic gate that the transistor belongs to. Since changes in T_{ox} leave the transistor input capacitance unchanged (see Section 2), the delay of the fanin gate is unchanged.

6. EXPERIMENTAL RESULTS

The proposed method for optimizing the total leakage was applied to the ISCAS85 benchmarks, and leakage/delay tradeoff curves were generated. A library consisting of inverter, and Nand and Nor gates with 2, 3, and 4 inputs, was characterized for a 100nm technology node using SPICE simulations based on a predictive model [13]. Based on this library, circuits were synthesized using SIS [18]. We used $V_{dd} = 1.2$V, inverter transistor widths $W_n = 400\text{nm}/W_p = 800\text{nm}$ (the widths for other gates are accordingly scaled), $T_{ox_{Lo}} = 12$Å, and $T_{ox_{Hi}} = 17$Å in our experimental setup.

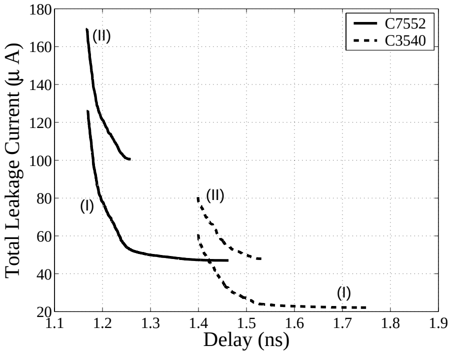


Figure 4: Leakage/Delay tradeoff curve for C2670 and C3540 with (I) all transistor T_{ox}'s optimized (II) all PMOS devices fixed at $T_{ox_{Lo}}$ and all NMOS T_{ox}'s optimized.

Tradeoff curves for two representative benchmarks are shown in Figure 4. All of the curves (only two are shown due to space constraints) show a knee region that corresponds to a good design point. The points to the right of the knee incur a large delay penalty for a little reduction in total leakage, while those on the left have a large leakage overhead for minor delay benefits. A notable observation is that though I_{gate} of a single PMOS transistor is small, setting all PMOS transistors to $T_{ox_{Lo}}$ incurs a high cumulative expense. This is shown by the curves (II), which correspond to a case where all PMOS transistors are set to $T_{ox_{Lo}}$ and the T_{ox} values of only the NMOS devices are optimized. This curve is clearly inferior to the curves (I) that correspond to a full T_{ox} optimization for all NMOS and PMOS transistors.

Circuit	Delay	Leakage Current (μA)			CPU Time	Circuit	Delay	Leakage Current (μA)			CPU Time
	(ns)(%D)	I_{sub}	I_{gate}	I_{total}(%R)	(s)		(ns)(%D)	I_{sub}	I_{gate}	I_{total}(%R)	(s)
C432	1.26(24.6)	3.77	0.69	4.46	2.2	C499	0.93(25.4)	7.28	1.61	8.89	11
	1.14	3.90	1.02	4.92			0.84	7.57	2.48	10.05	
	1.09	4.02	2.08	6.10			0.80	7.84	4.54	12.38	
	1.04	4.14	4.83	8.97			0.77	8.10	27.45	35.55	
	1.01	4.22	14.01	18.24 (83.8)			0.74	8.34	67.98	76.32 (70.2)	
	1.01	5.17	107.53	112.70			0.74	9.92	246.20	256.12	
C880	0.85(25.5)	5.00	1.16	6.16	1.4	C1355	1.03(25.0)	7.42	1.68	9.09	12
	0.75	5.07	1.34	6.41			0.93	7.72	2.96	10.67	
	0.72	5.12	1.93	7.06			0.88	7.93	5.87	13.81	
	0.69	5.17	4.12	9.29			0.84	8.16	36.13	44.28	
	0.68	5.21	9.11	14.31 (92.3)			0.82	8.38	77.06	85.43 (67.9)	
	0.68	6.83	179.14	185.97			0.82	9.93	256.36	266.30	
C1908	1.34(25.2)	8.89	2.09	10.99	12	C2670	1.20(25.3)	12.37	3.32	15.69	9.4
	1.21	9.16	2.98	12.14			1.06	12.53	4.00	16.53	
	1.15	9.42	6.71	16.13			1.02	12.70	5.86	18.56	
	1.11	9.63	20.27	29.89			0.99	12.86	10.07	22.93	
	1.07	9.78	49.10	58.88 (82.3)			0.96	12.97	25.74	38.72 (92.6)	
	1.07	12.19	321.42	333.61			0.95	16.90	506.69	523.59	
C3540	1.75(25.1)	17.63	4.51	22.13	22	C5315	1.59(26.1)	27.71	7.37	35.08	36
	1.56	17.94	5.43	23.37			1.43	28.07	8.22	36.29	
	1.49	18.20	9.36	27.55			1.36	28.35	11.30	39.66	
	1.44	18.43	21.22	39.64			1.31	28.63	26.33	54.96	
	1.40	18.64	42.67	61.31 (91.4)			1.26	28.82	68.59	97.41 (91.7)	
	1.40	23.99	691.55	715.54			1.26	37.84	1128.85	1166.7	
C6288	4.75(25.7)	36.88	8.95	45.82	258	C7552	1.46(25.3)	37.60	9.46	47.06	127
	4.30	38.49	14.50	53.00			1.29	38.45	11.73	50.18	
	4.09	40.03	29.05	69.07			1.24	39.11	17.09	56.20	
	3.95	41.25	214.93	256.19			1.20	39.79	40.65	80.44	
	3.78	42.08	485.51	527.59(62.7)			1.17	40.39	85.97	126.36(91.6)	
	3.78	50.16	1362.40	1412.6			1.17	51.43	1450.22	1501.6	

Table 2: Leakage/delay tradeoffs from dual T_{ox} optimization. For each circuit, Row 1 = all transistors @$T_{ox_{Hi}}$, Row 5 = all transistors @$T_{ox_{Lo}}$, Rows 2–4 = results for intermediate target delays. Row 4 matches the delay for the "all $T_{ox_{Lo}}$" point with a leakage savings of "%R," and "%D" in Row 1 shows the delay penalty of the all $T_{ox_{Hi}}$ case relative to this point. Each row shows I_{gate}, I_{sub} and I_{total}, and the CPU time required to generate the entire leakage-delay tradeoff curve is in the last column.

Table 2 shows leakage/delay tradeoffs for all ISCAS85 benchmarks (except the smallest, C17), including values of I_{sub}, I_{gate}, and I_{total} for various target delays. The all-$T_{ox_{Hi}}$ case typically has a delay penalty of about 25% as compared I_{sub} and I_{gate} typically increase, the latter being at a much more rapid rate. The delay corresponding to all transistors @$T_{ox_{Lo}}$ can be matched, with an average reduction, over all circuits, of 82.7% in I_{total}, with the minimum reduction being 62.7% for C6288. In each case, the knee point on the curve fares far better. The other data points show that our optimization technique yields a tradeoff curve that results in a smooth tradeoff as the total leakage increases from the all-$T_{ox_{Hi}}$ case, with a delay reduction that is in the range of about 20%.

7. CONCLUSION

We have presented a technique for reducing the total active leakage, including gate oxide leakage, by determining appropriate values of T_{ox}, and iteratively assigning them to the individual transistor in the circuit. Our approach shows a clear tradeoff between leakage and delay, and an achievable delay reduction of 20%.

8. REFERENCES

[1] Semiconductor Industry Association, "International Technology Roadmap for Semiconductors," 2002. Available at `http://public.itrs.net`.

[2] F. Hamzaoglu and M. R. Stan, "Circuit-Level Techniques to Control Gate Leakage for Sub-100 nm CMOS," in *Proc. of ACM/IEEE ISLPED*, pp. 60–63, Aug. 2002.

[3] D. Lee and D. Blaauw, "Static Leakage Reduction through Simultaneous Threshold Voltage and State Assignment," in *Proc. of ACM/IEEE DAC*, pp. 191–194, Jun. 2003.

[4] J. Kao *et al.*, "Transistor Sizing Issues and Tool for Multi-Threshold CMOS Technology," in *Proc. of ACM/IEEE DAC*, pp. 409–414, Jun. 1997.

[5] Y. Oowaki *et al.*, "A sub-0.1 μm Circuit Design with Substrate-Over-Biasing," in *IEEE ISSCC Dig. of Tech. Papers*, pp. 88–89, Feb. 1998.

[6] S. Narendra *et al.*, "Leakage Issues in IC design: Trends, Estimation, and Avoidance." Tutorial at the IEEE/ACM ICCAD, Nov. 2003.

[7] D. Lee *et al.*, "Analysis and Minimization Techniques for Total Leakage Considering Gate Oxide Leakage," in *Proc. of ACM/IEEE DAC*, pp. 175–180, Jun. 2003.

[8] C.-H. Choi *et al.*, "Impact of Gate Direct Tunneling on Circuit Performace: A Simulation Study," *IEEE Trans. on Electron Devices*, pp. 2823–2829, Dec. 2001.

[9] N. Sirisantana *et al.*, "High-Performace Low-Power CMOS Circuits Using Multiple Channel Length and Multiple Oxide Thickness," in *Proc. of IEEE ICCD*, pp. 227–232, Sept. 2000.

[10] K. Bernstein, Private Communication. IBM T. J. Watson Research Center, Yorktown Heights, NY, 2003.

[11] Y. Taur, "CMOS Design Near the Limits of Scaling," *IBM J. R&D*, vol. 46(2/3), pp. 213–222, Mar./May 2002.

[12] K. Chen *et al.*, "Predicting CMOS Speed with Gate Oxide and Voltage Scaling and Interconnect Loading Effects," *IEEE Trans. On Electron Devices*, vol. 44(11), pp. 1951–1957, Nov. 1997.

[13] Device Group at UC Berkeley, "Berkeley Predictive Technology Model," 2002. Available at `http://www-device.eecs.berkeley.edu/∼ptm/`.

[14] S. Sirichotiyakul *et al.*, "Duet: An Accurate Leakage Estimation and Optimization Tool for Dual-V_t Circuits," *IEEE Trans. on VLSI Systems*, vol. 10(2), pp. 79–90, Apr. 2002.

[15] A. Chandrakasan *et al.*, *Design of High-Performance Microprocessor Circuits.* Piscataway, NJ: IEEE Press, 2001.

[16] K. A. Bowman *et al.*, "A Circuit-Level Perspective of the Optimum Gate Oxide Thickness," *IEEE Trans. on Electron Devices*, vol. 48(8), pp. 1800–1810, Aug. 2001.

[17] J. Fishburn and A. Dunlop, "TILOS: A Posynomial Programming Approach to Transistor Sizing," in *Proc. of ACM/IEEE ICCAD*, pp. 326–328, Nov. 1985.

[18] E. M. Sentovich *et al.*, "SIS: A System for Sequential Circuit Synthesis," Tech. Rep. UCB/ERL M92/41, Electronics Research Laboratory, Dept. of EECS, University of California, Berkeley, May 1992.