

# Physical-Aware Simulated Annealing Optimization of Gate Leakage in Nanoscale Datapath Circuits

Saraju P. Mohanty  
Computer Science & Engineering  
University of North Texas  
Denton, TX 76203, USA.  
Email: smohanty@cs.unt.edu

Ramakrishna Velagapudi  
Computer Science & Engineering  
University of North Texas  
Denton, TX 76203, USA.  
Email: rv0063@unt.edu

Elias Kougianos  
Engineering Technology  
University of North Texas  
Denton, TX 76203, USA.  
Email: elias@unt.edu

## Abstract

*For CMOS technologies below 65nm, gate oxide direct tunneling current is a major component of the total power dissipation. This paper presents a simulated annealing based algorithm for the gate leakage current reduction by simultaneous scheduling, allocation and binding during behavioral synthesis. Gate leakage current reduction is based on the use of functional units of different oxide thickness while simultaneously accounting for process variations. We present a cost function that minimizes leakage and area overhead. The algorithm minimizes the cost function for a given delay trade-off factor. It uses a pre-characterized cell library for tunneling current, delay and area, expressed as analytical functions of the gate oxide thickness  $T_{ox}$ . We tested our approach using a number of behavioral level benchmark circuits characterized for a 45nm library by integrating our algorithm into a high-level synthesis system. We obtained an average gate leakage reduction of 76.88% with an average area overhead of 17.38% for different delay trade-off factors ranging from 1.0 to 1.4.*

## 1 Introduction

The power consumption of CMOS circuits has become an issue of major concern due to scaling and the consequent increase in leakage components in both ON and OFF states. As the systems become faster and smaller in size, the leakage power dissipation tends to increase exponentially [6]. According to the ITRS [1], a high performance CMOS device will require gate oxide thicknesses of  $0.7nm - 1.2nm$ , thus making it more susceptible to new leakage mechanisms due to carriers tunneling through the ultra thin layer of gate oxide [16]. Therefore various options need to be explored for reduction of active leakage power dissipation in the advent of nanoscale CMOS technologies.

It has been well established that for achieving low-power design for a system, power has to be optimized at all levels of the design process. There are several transistor and logic level methods available for reducing sleep mode leakage, such as multi- $V_{Th}$  [14, 13], body-biasing [12], and state assignment [7]. However, active leakage has not received the attention it deserves, and there are few works which address gate tunneling current at behavioral level. In this work, we propose the use of functional units of different oxide thickness to optimize gate tunneling current during behavioral synthesis.

Tunneling current dissipation in a CMOS with a supply voltage  $V_{dd}$  and effective gate oxide thickness  $T_{ox}$  is given by [6, 3]:

$$I_{ox} \propto \left( \frac{V_{dd}}{T_{ox}} \right)^2 \exp \left( -k \frac{T_{ox}}{V_{dd}} \right), \quad (1)$$

where,  $k$  is an experimentally derived factor. Based on Eqn. 1, we have two options for reducing gate leakage: reducing the supply voltage and/or increasing the gate oxide thickness. The popular option of scaling down of supply voltage [10] continues to play its role in the reduction of dynamic power as well as leakage power, but it may not be sufficient to contain the exponential growth of the gate leakage. Increase in the gate  $SiO_2$  thickness leads to an increase in propagation delay and area. Thus, the use of multiple gate oxide thicknesses can serve as a leakage power (current), performance and area trade-off. In this paper we explore the three dimensional design space (gate leakage power, performance and area) using the multiple thickness (multi- $T_{ox}$ ) approach for reduction of gate direct tunneling current during behavioral synthesis.

The rest of this paper is organized as follows: section 2 gives an overview of current research, section 3 gives a novel approach for tunneling current reduction at behavioral level, section 4 presents a datapath component library for 45nm CMOS technology, and in section 5 we present experimental results.

## 2 Related Work and our Contributions

Barring a few works in leakage reduction, dynamic power reduction has been the area of major thrust in low power behavioral synthesis. Also a few logic and transistor level research works focus on addressing reduction of gate oxide tunneling.

In [5], Khouri and Jha proposed a dual- $V_{Th}$  technique for subthreshold leakage analysis and reduction during behavioral synthesis, targeting the least used modules as the candidates for leakage optimization.

Gopalakrishnan and Katkooi in [4] use the MTCMOS approach for reduction of subthreshold current during high-level synthesis and propose binding algorithms for power, delay, and area trade-off.

Mohanty et. al. [9] have presented analytical models and a datapath scheduling algorithm for reduction of tunneling current. The heuristic assigns higher thickness resources to more leaky nodes (multipliers), but does not address the area overhead.

In [8], Lee et. al., developed a method for analyzing gate oxide leakage current in logic gates and suggested utilizing pin reordering to reduce the gate leakage. Sultania et al. in [16], developed an algorithm to optimize the total leakage power by assigning dual  $T_{ox}$  values to transistors in a given circuit. In [15], Sirisantana and Roy use multiple channel lengths and oxide thicknesses for reduction of leakage.

*The key contributions of this paper are as follows:* Optimization of gate tunneling current and circuit area together with a given time constraint using a simulated annealing based algorithm that does simultaneous scheduling, allocation and binding. The algorithm takes process variations into account while optimizing the cost function. In the sub-65nm CMOS technology the gate oxide thickness is very small, say approximately 1.2nm. However, a monolayer of SiO<sub>2</sub> is approximately 0.2nm. Thus, a layer of SiO<sub>2</sub> misplacement can cause significant variation in the effective  $T_{ox}$ , and resultant gate leakage, and hence this effect needs to be accounted for. The paper also presents simplified analytical functions for the calculation of gate leakage, delay and area of nano-CMOS based architectural units. The algorithm considers multicycling and chaining based datapath to reduce delay penalty.

## 3 Simulated Annealing Based Optimization

In this section we present a simulated annealing based algorithm that minimizes the current-area cost function using the notation described in Table 1. Given a time constraint we need to determine an RTL implementation that has minimum tunneling current and area. Annealing is the process of heating and cooling a material slowly until it crystalizes. The atoms of this material have higher energies at very high

**Table 1. Notations used in Description**

$V\{v_i\}$	Set of nodes or vertices in $UDFG$
$C[v_i]$	Final time stamp of a vertex $v_i$
$FU_j(k, T_{ox})$	$j^{th}$ resource of type $k$ and thickness $T_{ox}$
$T_{oxH}$	Higher gate oxide thickness
$T_{oxL}$	Lower gate oxide thickness
$R_{Avl}[\dots][\dots k \dots][\dots T_{ox} \dots]$	Availability matrix; $c$ is any clock cycle
$DTF$	Performance or delay trade-off factor
$T_{CPST}$	Critical path delay for single oxide thickness
$T_{CPDT}$	Critical path delay for dual oxide thickness
$I_{oxST}$	Tunneling current for single oxide thickness
$I_{oxDT}$	Tunneling current for dual oxide thickness
$A_{ST}$	Total Area for single oxide thickness
$A_{DT}$	Total Area for dual oxide thickness
$S$	Scheduled $DFG$ with resource binding

```

Simulated_Annealing_Algorithm(UDFG, DTF)
(01) Initial Temperature  $\leftarrow t_o$ .
(02) Available Resources  $\leftarrow \infty$ .
(03) While there exists a schedule with available resources.
(04)    $i$  = Number of iterations.
(05)   Perform resource constrained ASAP.
(06)   Perform resource constrained ALAP.
(07)   Initial Solution  $\leftarrow$  ASAP Schedule.
(08)    $S \leftarrow$  Allocate.Bind().
(09)   Initial Cost  $\leftarrow$  Cost( $S$ ).
(10)   While ( $i > 0$ )
(11)     Generate a random thicknesses in range of
            $(T_{oxL} - \delta T_{oxL}, T_{oxL} - \delta T_{oxL})$  and
            $(T_{oxH} - \delta T_{oxH}, T_{oxH} - \delta T_{oxH})$ .
(12)     Generate random transition from  $S$  to  $S^*$  using
           the function in Fig. 2 for a given  $DTF$ .
(13)      $\Delta_c \leftarrow$  Cost( $S$ ) - Cost( $S^*$ )
(14)     if ( $\Delta_c > 0$ ) then  $S \leftarrow S^*$ .
(15)     else if ( $e^{\Delta_c/t} > \text{random}[0,1]$ ) then  $S \leftarrow S^*$ .
(16)      $i \leftarrow i - 1$ .
(17)   end While.
(18)   Decrement available resources.
(19)    $t \leftarrow \alpha \times t$ .
(20) end While.
(21) return  $S$ .

```

**Figure 1. The Proposed Algorithm**

temperatures. This gives the atoms a great deal of freedom in their ability to reconstruct themselves. As the temperature decreases the energy of the atoms decrease. Analogous to the annealing process, the mobility of nodes in a DFG is dependent on the total available resources. Here the nodes of a DFG are analogous to the atoms and temperature is analogous to the total number of available resources.

Our approach for reducing tunneling current is using functional units of different gate oxide thicknesses. To maximize the leakage reduction we need to ensure that every node can be scheduled in such a way that a higher thickness resource can be assigned. The mobility of the nodes (chance of assigning a higher thickness resource) is dependent on the total number of available higher thickness resources. We apply the annealing principle to explore the trade-offs between power, performance and area.

The input to this algorithm is an unscheduled DFG and time constraints and the output is an RTL description that

```

Generate_Neighborhood( $S$ ,  $DTF$ , Cell Library)
(01) Select a random vertex  $v_i \in V$ .
(02) FOR each possible cycle  $c$  in the mobility range.
(03)   IF ( $FU_j(k, T_{oxH})$  is available in  $R_{Avl}$  for  $c$ ) then
(04)     Schedule  $v_i$  in step  $c$ , Assign  $c$  to  $C[v_i]$ .
(05)      $R_{Bin}[k][T_{oxH}][r_j] = \text{Bind}(v_i)$ . /*  $v_i$  needs type- $k$  */
(06)     Decrement  $R_{Avl}[C[v_i]][k][T_{oxH}]$ . /* Allocated */
(07)     Increment  $R_{Avl}[C[v_i]][k][T_{oxL}]$ . /*  $T_{oxL}$  resource freed */
(08)     TotalDelay=0. /* Initialize Delay */
(09)     While( $\forall v_i \in V$  execution of  $v_i$  is not done)
(10)       For each  $v_i \in V$ 
(11)         If(All predecessors of  $v_i$  finished execution and
            $v_i$  has not yet started execution and
           required resource is available) then
           start executing  $v_i$ .
(12)       For each  $v_i \in V$ 
(13)         If( $v_i$  started execution and not yet finished)
(14)           var= $v_i$ , break. /* var= node to be executed */
(15)         Increment TotalDelay by Delay(var,  $FU_j(k, T_{oxH})$ ).
(16)       For each  $v_i \in V$ 
(17)         If( $v_i$  started execution and not yet finished)
(18)           Execute  $v_i$  for a period of
           Delay(var,  $FU_j(k, T_{oxH})$ ).
(19)         If( $v_i$  finished execution)
(20)           then mark  $v_i$  as completed. /* executed */
(21)

```

**Figure 2. Generation of Random Transitions**

```

Cost_Calculation( $S$ , Cell Library)
(01)  $I_{oxFU_i} = \text{TunnelingCurrent}(FU_i(k, T_{ox}))$ 
(02)  $A_{FU_i} = \text{Area}(FU_i(k, T_{ox}))$ 
(03)  $I_{ox} = \sum_{i=1}^{totalnodes} I_{oxFU_i}$ 
(04)  $A = \sum_{i=1}^{totalnodes} A_{FU_i}$ 
(05) Cost =  $\alpha * I_{ox} + \beta * A$ 
(06) return Cost;

```

**Figure 3. Algorithm for Cost Function**

dissipates minimal gate leakage. The dual- $T_{ox}$  algorithm is presented in Fig. 1. It starts with the ASAP schedule and assigns lower thickness resources to all the operations. This is done by the function `Allocate.Bind`. The total leakage is determined as the sum of leakages of all the allocated resources, so the minimum number of resources required for the schedule is determined and allocated. Once the execution of a clock cycle is finished all the resources are assumed to be in ready state before running the next clock cycle.

The annealing based algorithm provides an opportunity to explore a three dimensional design space (gate leakage power, performance and area). In the outer loop during each iteration the number of higher thickness resources is decreased, which restricts the mobility of the nodes. The algorithm attempts to find an RTL that has minimum leakage for a given number of available resources. In the inner loop during each iteration a neighborhood solution is generated (Fig. 2). If this solution has less leakage than the current solution, the neighborhood solution is made as the current solution. This way the algorithm converges to a solution that has minimum leakage. In generating a neighborhood solution we randomly select a node and check if a higher

thickness resource can be assigned in all possible clock cycles and that it satisfies a time constraint.

Each time a different thickness resource is assigned a random thickness in the range of  $(T_{ox} - \delta T_{ox}, T_{ox} + \delta T_{ox})$  is assigned to take process variation into account. Assuming a monolayer misplacement of  $\text{SiO}_2$ ,  $\delta T_{ox}$  is approx. 15%.

For calculating the total delay of the circuit for a single cycle case we used the critical path delay. While generating a random neighborhood solution the algorithm ensures that the nodes in the critical path are not assigned a higher  $T_{ox}$  resource. For multicycling, the total delay of the circuit is calculated as the product of total number of control steps and the maximum delay of any resource in the circuit. Assigning higher thickness resources will increase the delay. We intuitively believe that using chaining and multicycling can reduce the delay to a great extent. Using multicycling or chaining alone may not give good results because multicycling increases the number of control steps, while using chaining there were only few operations for which chaining can be implemented. We found that multicycling along with chaining can reduce the delay to a great extent. The basic idea behind using both multicycling and chaining is to ensure that the execution of any operation that is ready (all its predecessors finished execution) and has a resource available will start its execution.

In designing a cost function it is essential to integrate both area and power in such a way that equal weight is given to both. The cost function is shown in Fig 3. The cost function is expressed in terms of  $I_{ox}$  and  $A$ , where  $I_{ox}$  is the tunneling current of the entire circuit and  $A$  is the total area of the circuit.  $I_{ox}$  of the circuit is calculated as the sum of tunneling current of all the nodes in the circuit.  $A$  is the sum of areas of all the allocated resources.  $\alpha$  and  $\beta$  are the weights of current and area respectively. Since a lower tunneling and area is what is desired;  $\alpha$  and  $\beta$  are positive values. Depending on the requirements, we choose appropriate values of  $\alpha$  and  $\beta$ .

## 4 Multi-Oxide Thickness Datapath Library

In this section we present a bottom-up approach for the characterization of functional units. First we characterize the NAND gate using analog simulation results and then the functional units using analytical models. The three levels of abstraction are shown in Fig 4. We chose the NAND gate for two reasons: first it is a universal gate and secondly its gate leakage is minimal compared to other gates [11].

### 4.1 Logic Level

We used BPTM BSIM4 model for simulation to find  $I_{ox}$  and  $T_{pd}$ . On the other hand due to unavailability of silicon data we used an analytical estimate for area calculations.

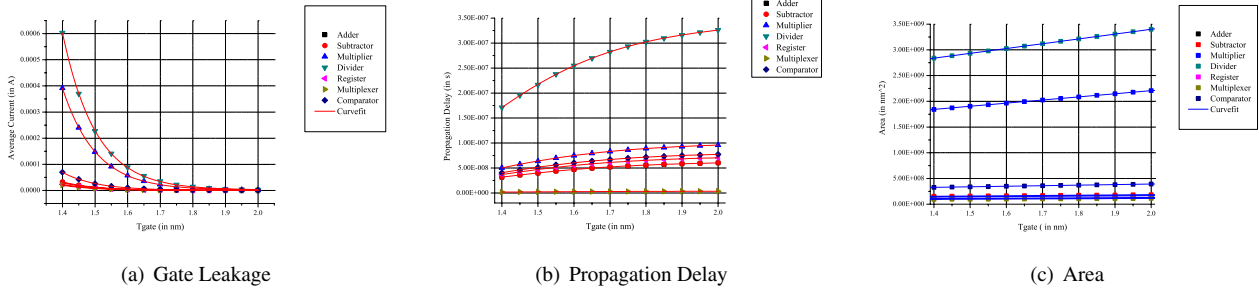


Figure 6. Variation of Gate Leakage Current, Propagation Delay and Area

Table 2. Current Delay and Area as Analytical Functions of  $T_{ox}$  (in nm) Curve Fitting Parameters

$I_{ox}(\mu A) = A \exp(-T_{ox}/\alpha) + B$				$T_{pd}(ns) = ((A_1 - A_2)/(1 + (T_{ox}/\beta)^\gamma)) + A_2$				$A(nm^2) = \alpha T_{ox} + \beta$	
Unit	$\alpha$	$A$	$B$	$A_1$	$A_2$	$\beta$	$\gamma$	$\alpha$	$\beta$
Adder	0.10254	24.82459	0.082099	-7.3759	64.477	1.36771	7.24555	$0.454757 \times 10^8$	$0.742770 \times 10^8$
Subtractor	0.10254	27.76185	0.091813	-7.3759	64.477	1.36771	7.24555	$0.508564 \times 10^8$	$0.830655 \times 10^8$
Multiplier	0.10254	331.62623	1.096700	-11.753	102.74	1.36771	7.24555	$6.075000 \times 10^8$	$9.922500 \times 10^8$
Divider	0.10254	510.89389	1.689620	-39.938	349.12	1.36771	7.24555	$9.358970 \times 10^8$	$15.28630 \times 10^8$
Register	0.10254	19.70807	0.065178	-8.6352	75.485	1.36771	7.24555	$0.361029 \times 10^8$	$0.589680 \times 10^8$
Multiplexer	0.10254	16.77081	0.055464	-0.4197	3.6694	1.36771	7.24555	$0.307221 \times 10^8$	$0.501795 \times 10^8$
Comparator	0.10254	58.83997	0.194594	-9.4748	82.823	1.36771	7.24555	$1.077880 \times 10^8$	$1.760540 \times 10^8$

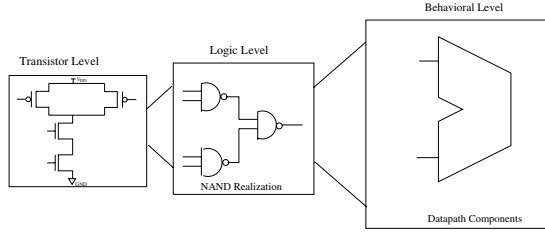


Figure 4. Three Level Abstraction

The transistor level diagram in Fig. 5 shows tunneling current paths in constituent NAND gate. If the four possible states (00, 01, 10 and 11) have gate tunneling current ( $I_{ox00}$ ,  $I_{ox01}$ ,  $I_{ox10}$ ,  $I_{ox11}$ ), respectively, and assuming that all four states are equiprobable the average tunneling current of a 2-input NAND gate is  $I_{oxNAND_i} = \frac{(I_{ox00} + I_{ox01} + I_{ox10} + I_{ox11})}{4}$ . The oxide direct tunneling current is obtained by evaluating diffusion, channel and body components for a PMOS or NMOS device from the BPTM and summing them as:  $\sum_{MOS_i} (|I_{gs_i} + I_{gd_i} + I_{gcs_i} + I_{gcd_i} + I_{gb_i}|)$ . In summary, we account for the *tunneling current of NMOS and PMOS devices in both ON and OFF states*.

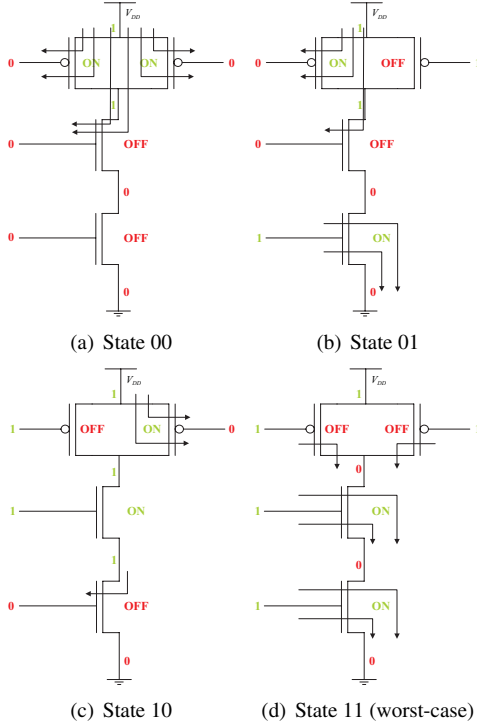
The area of a NAND gate is calculated using the following equation [2].

$$A_{NAND} = k_{inv} \left( 1 + 4(n_{in} - 1) \sqrt{\frac{AR_{NAND}}{k_{inv}}} \right) \times \left( 1 + \frac{(W_{NMOS} - 1)(1 + \beta_{NAND})}{\sqrt{k_{inv} AR_{NAND}}} \right) \quad (2)$$

where,  $W_{NMOS}$  = NMOS width,  
 $f$  = minimum feature size for a technology,  
 $k_{inv}$  = area of minimum size inverter with respect to  $f^2$ ,  
 $AR_{NAND}$  = aspect ratio of NAND gate,  
 $n_{in}$  = number of inputs, and  
 $\beta_{NAND}$  = ratio of PMOS width to NMOS width.

## 4.2 Architectural Level

We assume that there are total  $n_{total}$  NAND gates in the network of NAND gates constituting an  $n$ -bit functional unit out of which  $n_{cp}$  are in the critical path. In this model we do not consider the effect of interconnect wires and focus on the direct tunneling current dissipation and propagation delay of the active units only. The phenomenon of oxide tunneling current is restricted to the active devices and does not have any influence on the power dissipation in the interconnect. We calculate the direct tunneling current ( $I_{oxFU}$ ) of an  $n$ -bit functional unit as  $I_{oxFU} = \sum_{i=1}^{n_{total}} I_{oxNAND_i}$ , where  $I_{oxNAND_i}$  is the *average gate oxide tunneling current* dissipation of the  $i^{th}$  2-input NAND gate in the functional unit, assuming all states to be equiprobable. Similarly, the propagation delay and silicon area of an  $n$ -bit functional unit are  $T_{pdFU} = \sum_{i=1}^{n_{cp}} T_{pdNAND_i}$  and  $A_{FU} = \sum_{i=1}^{n_{total}} A_{NAND_i}$ , respectively.



**Figure 5. Tunneling Current Paths**

### 4.3 Analytical Functions in terms of $T_{ox}$

In order to facilitate the physical-aware simulated annealing algorithm presented in the previous section we need to describe the characterization data obtained as functions of  $T_{ox}$ . We used the calculated and simulated data and fit different analytical functions presented in Table 2 and Fig. 6. The functions obtained have a correlation coefficient of approximately 0.99. Thus, the curves faithfully represent the data.

## 5 Experimental Results

In this section we present the experimental results and our findings. The algorithm is implemented in C and integrated in the high-level synthesis framework described in [10]. The proposed algorithm is then tested with various selected behavioral synthesis benchmarks circuits [10]. While calculating the tunneling current for single thickness, we used a nominal  $1.4nm$  thickness as the default value from the BSIM4.4.0 model. We considered three dual thickness pairs of (i)  $1.4nm - 1.5nm$ , (ii)  $1.4nm - 1.6nm$ , and (iii)  $1.4nm - 1.7nm$ , but presented the results for  $1.4nm - 1.7nm$  due to page limitations. To begin with, we assumed an infinite number of  $T_{oxL}$  and  $T_{oxH}$  resources and during each iteration we decreased the number of  $T_{oxH}$

resources. We performed the experiments for both multicycling and chaining based datapath as well as single cycle datapath circuits.

The results take into account the tunneling current, area and propagation delay of functional units, interconnect units, and storage units present in the datapath circuit. The percentage reduction in direct tunneling current is calculated as  $\Delta I = \left( \frac{I_{oxST} - I_{oxDT}}{I_{oxST}} \right) * 100\%$  and the percentage area overhead is calculated as  $\Delta A = \left( \frac{-(A_{ST} - A_{DT})}{A_{ST}} \right) * 100\%$ . We estimate the critical path delay of the circuit as the sum of the delays of the vertices in the longest path of the DFG for single cycle case and number of control steps times slowest delay resource for multicycling-chaining case. The delay trade-off factor (DTF) is used to provide various time constraints for our experiments.

We obtained a number of design alternatives during each iteration of our algorithm. We observed that the extent to which tunneling current reduction takes place increases as the number of available  $T_{oxH}$  resources increases. We selected the design for minimal cost function to obtain maximum tunneling current reduction with minimal compromise on the area penalty. The results for single cycle datapath circuits for various benchmarks for dual thickness technique for  $1.4nm - 1.7nm$  corresponding to average case minimal cost functions are reported in Table 3. Here,  $I_{ox}$ ,  $T_{CP}$  and  $A$  represent the total gate leakage current, critical path delay and area of the circuit, respectively. The subscripts  $ST$  and  $DT$  stand for single thickness and multiple thickness, respectively. For the single cycle approach the reduction in tunneling current for all the benchmarks ranges from 57.99% to 91.54% for an area penalty ranging from 6.66% to 44.73% for different delay trade-off factors considered in the experiment.

For multicycling and chaining datapath case we observed tunneling current reduction ranging from 30.5% to 91.11% for an area penalty ranging from 4.80% to 28.67% for different delay trade-off factors. One significant observation is that there is a drastic reduction in delay compared to single cycle operation. The design space exploration for this is very similar to the single cycle case.

## 6 Conclusions

In this paper we presented a physical-aware technique for reduction of tunneling current using simultaneous scheduling and binding of functional units. The method of using dual oxide thickness in a simulated annealing optimization algorithm has demonstrated very encouraging results. The results achieved outperformed other behavioral level leakage reduction works available in the literature in terms of percentage reduction. Further exploration of this technique in addition to the use of multiple high- $K$  dielec-

**Table 3. Tunneling Current, Delay and Area for Various Benchmarks ( $T_{ox_L} = 1.4nm$  and  $T_{ox_H} = 1.7nm$ )**

Benchmarks	DTF	$I_{ox_{ST}}(\mu A)$	$T_{CP_{ST}}(ns)$	$A_{ST}(\mu m^2)$	$I_{ox_{DT}}(\mu A)$	$T_{CP_{DT}}(ns)$	$A_{DT}(\mu m^2)$	$\Delta I$	$\Delta A$
(1) ARF	1.0	6618.21	308.9	15293.77	1628.16	308.9	16610.45	75.39	8.60
	1.1	6618.21	308.9	15293.77	1600.54	329.4	16610.45	75.81	8.60
	1.2	6618.21	308.9	15293.77	1231.57	362.2	16610.45	81.39	8.60
	1.3	6618.21	308.9	15293.77	890.21	374.4	16610.45	86.54	8.60
	1.4	6618.21	308.9	15293.77	890.21	374.4	16610.45	86.54	8.60
(2) BPF	1.0	5222.48	290.1	9798.16	1215.81	290.1	12902.17	76.71	31.67
	1.1	5222.48	290.1	9798.16	1184.92	310.7	12902.17	77.31	31.67
	1.2	5222.48	290.1	9798.16	815.95	343.5	12902.17	84.37	31.67
	1.3	5222.48	290.1	9798.16	788.33	364.0	12902.17	84.90	31.67
	1.4	5222.48	290.1	9798.16	754.17	405.2	12747.90	85.55	30.10
(3) DCT	1.0	5941.66	308.9	8474.54	1644.24	308.9	9254.73	72.32	9.20
	1.1	5941.66	308.9	8474.54	1589.00	308.9	9116.79	73.25	7.57
	1.2	5941.66	308.9	8474.54	1330.50	341.7	11279.73	77.60	33.10
	1.3	5941.66	308.9	8474.54	1330.50	341.7	11279.73	77.60	33.10
	1.4	5941.66	308.9	8474.54	1275.26	362.2	9254.73	78.53	9.20
(4) EWF	1.0	3895.46	498.4	6080.02	1636.26	498.4	6485.45	57.99	6.66
	1.1	3895.46	498.4	6080.02	1267.29	531.2	6485.45	67.46	6.66
	1.2	3895.46	498.4	6080.02	870.69	584.5	6485.45	77.64	6.66
	1.3	3895.46	498.4	6080.02	815.45	646.2	8799.97	79.06	44.73
	1.4	3895.46	498.4	6080.02	815.45	646.2	8799.97	79.06	44.73
(5) FIR	1.0	3572.96	303.0	15845.5	796.78	282.4	17216.79	77.69	8.65
	1.1	3572.96	303.0	15845.5	769.16	323.5	17216.79	78.47	8.65
	1.2	3572.96	303.0	15845.5	741.54	344.1	17216.79	79.24	8.65
	1.3	3572.96	303.0	15845.5	400.18	356.3	17399.04	88.79	9.80
	1.4	3572.96	303.0	15845.5	372.56	418.0	17385.40	89.57	9.71
(6) IIR	1.0	2075.52	145.0	9489.63	571.99	145.0	10232.27	72.44	7.82
	1.1	2075.52	145.0	9489.63	571.99	145.0	10232.27	72.44	7.82
	1.2	2075.52	145.0	9489.63	544.37	165.6	10232.27	73.77	7.82
	1.3	2075.52	145.0	9489.63	203.01	177.8	10414.52	90.21	9.74
	1.4	2075.52	145.0	9489.63	175.39	198.4	10414.52	91.54	9.74

tric gate materials is being explored for future implementations. The ultimate objective is to extend the work on tunneling current to provide a broader solution to the problem of power dissipation in all its forms at the behavioral level.

## References

- [1] Semiconductor Industry Association, Intl. Tech. Roadmap for Semiconductors. <http://public.itrs.net>.
- [2] K. A. Bowman, et. al. A Circuit-Level Perspective of the Optimum Gate Oxide Thickness. *IEEE Transactions on Electron Devices*, 48(8):1800–1810, August 2001.
- [3] A. Chandrakasan, W. Bowhill, and F. Fox. *Design of High-Performance Microprocessor Circuits*. IEEE Press, 2001.
- [4] C. Gopalakrishnan and S. Katkoori. Knapbind: an area-efficient binding algorithm for low-leakage datapaths. In *Proceedings of 21st ICCD*, pages 430–435, 2003.
- [5] K. S. Khouri and N. K. Jha. Leakage power analysis and reduction during behavioral synthesis. *IEEE Transactions on VLSI Systems*, 10(6):876–885, December 2002.
- [6] N. S. Kim, et. al. Leakage Current - Moore's Law Meets Static Power. *IEEE Computer*, pages 68–75, December 2003.
- [7] D. Lee and D. Blaauw. Static Leakage Reduction Through Simultaneous Threshold Voltage and State Assignment. In *Proceedings of the DAC*, pages 191–194, 2003.
- [8] D. Lee, D. Blaauw, and D. Sylvester. Gate Oxide Leakage Current Analysis and Reduction for VLSI Circuits. *IEEE Trans on VLSI Systems*, 12(2):155–166, February 2004.
- [9] S. P. Mohanty, V. Mukherjee, and R. Velagapudi. Analytical Modeling and Reduction of Direct Tunneling Current during Behavioral Synthesis of Nanometer CMOS Circuits. In *Proceedings of the 14th ACM/IEEE International Workshop on Logic and Synthesis (IWLS)*, pages 249–256, 2005.
- [10] S. P. Mohanty and N. Ranganathan. Energy Efficient Datapath Scheduling using Multiple Voltages and Dynamic Clocking. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 10(2):330–353, April 2005.
- [11] V. Mukherjee, S. P. Mohanty, and E. Kougianos. A Dual Dielectric Approach for Performance Aware Gate Tunneling Reduction in Combinational Circuits. In *Proc. of the 23rd IEEE Intl Conf of Computer Design*, pages 431–436, 2005.
- [12] S. Narendra, et. al. Forward Body Bias for Microprocessors in 130-nm Technology Generation and Beyond. *IEEE Journal of Solid-State Circuits*, 38(5):696–701, May 2003.
- [13] P. Pant, et. al. Dual-Threshold Voltage Assignment with Transistor Sizing for Low Power CMOS Circuits. *IEEE Trans on VLSI Systems*, 9(2):390–394, April 2001.
- [14] K. Roy and R. Krishnammthy. Design of Low voltage CMOS circuits : Tutorial Guide. In *Proc of the IEEE Intl Symposium on Circuits and Systems*, pages 3.2.1–3.2.29, 2001.
- [15] N. Sirisantana and K. Roy;. Low-power Design using Multiple Channel Lengths and Oxide Thicknesses. *IEEE Design & Test of Computers*, 21(1):56–63, Jan-Feb 2004.
- [16] A. K. Sultania, D. Sylvester, and S. S. Sapatnekar. Trade-offs Between Gate Oxide Leakage and Delay for Dual  $T_{ox}$  Circuits. In *Proceedings of DAC*, pages 761–766, 2004.