# Tabu Search Based Gate Leakage Optimization using DKCMOS Library in Architecture Synthesis

Saraju P. Mohanty[1] and Dhiraj K. Pradhan[2]
Department of Computer Science and Engineering, University of North Texas, USA.[1]
Department of Computer Science, University of Bristol, UK.[2]
Email-ID: saraju.mohanty@unt.edu[1] and pradhan@compsci.bristol.ac.uk[2].

***Abstract*— The gate-oxide (*aka* gate tunneling or gate) leakage due to quantum-mechanical direct tunneling of carriers across the gate dielectric of a device is a major source power dissipation for sub-$65nm$ CMOS circuits. In this paper a high-level (*aka* architecture) synthesis algorithm is presented that simultaneously schedules operations and binds to modules for gate leakage optimization. The algorithm uses device-level gate leakage models for precharacterizing register-transfer level datapath component library. The algorithm minimizes the gate leakage for given resource and time constraints. The dual-K CMOS (DKCMOS) technology is used as a method for gate leakage power reduction in a data flow graph. The proposed algorithm is tested for several high-level synthesis benchmarks for two types of DKCMOS, $SiO_2$-SiON and $SiO_2$-$Si_3N_4$, for $45nm$ node. The experiments showed that gate leakage reduction in average $60\%$ and $72\%$ for $SiO_2$-SiON and $SiO_2$-$Si_3N_4$, respectively, could be achieved.**

## I. Introduction

In short channel nanoscale CMOS devices, several forms of leakage exist, such as reverse-biased diode leakage, subthreshold leakage, gate-oxide tunneling, hot carrier gate leakage, gate-induced drain leakage, and channel punch through current [27], [23]. The major sources of power dissipation in a nano-CMOS circuit can be summarized as [27], [23], [7], [4]:

$$P_{total} = P_{gate} + P_{subthreshold} + P_{switch}, \qquad (1)$$

where, $P_{gate}$ is the gate-oxide leakage power, $P_{subthreshold}$ is the subthreshold leakage power, $P_{switch}$ is the dynamic power dissipation due to switching capacitance. The existing literature is full of mature research addressing dynamic as well as subthreshold-leakage power dissipation, but research addressing gate-oxide leakage is still lacking. We believe that the gate leakage needs explicit attention due to several reasons including: (i) Gate-oxide leakage is present during ON, OFF, and transient states of a device as opposed to subthreshold leakage which is only present during the OFF state. (ii) Gate-oxide leakage is the pre-dominant form of power for sub-$65nm$ technology nodes using ultra-thin oxide.

This paper introduces dual-K based architecture synthesis technique to reduce gate leakage while maintaining specified performance of the circuits. On the contrary, the existing gate leakage reduction techniques are primarily at the device or transistor level. It is well known that high-level design space exploration can lead to larger savings in power dissipations, which motivated the proposed research. This paper presents register-transfer level (RTL) gate leakage power reduction based on dual-K based design library.

The paper is organized as follows. Related research are summarized in Section II. Section III introduces high-K and DKCMOS technologies. Modeling of gate leakage is discussed in Section IV. RTL optimization problem is presented in Section V. A DKCMOS based RTL library is presented in Section VI. An algorithm to perform scheduling and binding for gate leakage reduction is discussed in Section VII. Experimental results are presented in Section VIII. The conclusions and suggestions for future research are presented in Section IX.

## II. Contributions of this Paper and Prior Works

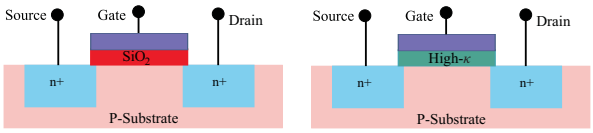
The contributions of this paper is the introduction and use of DKCMOS technology for gate leakage reduction of datapath circuits while maintaining their specified performance. An algorithm is proposed that schedules operations of a sequencing data flow graph (DFG) and maps the operations to RTL library for optimization of gate leakage. The RTL library is constructed for classical $SiO_2$ device based modules, and two nonclassical high-K based modules. The algorithm minimizes the gate leakage of datapath circuits for given resource constraints and time constraints.

The prior research in high-level synthesis mostly considered dynamic power and few of them have dealt with subthreshold or gate leakage reduction. In [17], dual-$V_{Th}$ techniques for subthreshold leakage analysis and reduction have been proposed. In [12], MTCMOS (Multi-Threshold CMOS) approach is also used for reduction of subthreshold current. In [10], power island partitioning has been used to reduce subthreshold leakage. In [28], a heuristic based approach using dual-$V_{Th}$ library is proposed. The high-level synthesis research addressing gate leakage is presented in [23], [22].

## III. DKCMOS Technology

### A. High-K Technology for Nano-CMOS

It has now become desirable to find suitable alternatives for $SiO_2$ as the gate dielectric [16], [21]. This has led to the construction of non-classical transistors as in. Fig. 1. Intel has developed a processor called *Penryn* using such transistors of $45nm$ technology, which is succeeded by *Nehalem* [3]. Intel Core i7 is the latest $45nm$ microprocessor with $731M$ transistors. Other semiconductor industry such as AMD, IBM, Infineon, Samsung, and Chartered Semiconductor are also developing $45nm$, $32nm$, and $28nm$ process platform.




(a) $SiO_2$ gate dielectric: low gate capacitance, low delay, and high gate leakage due to tunneling. (b) High$-K$ gate dielectric: high gate capacitance, high delay, and low gate leakage due to tunneling.

Fig. 1. Nano-CMOS Transistors : Classical Vs. Nonclassical. The use of dual dielectric $SiO_2$ and high$-K$ is needed for power and performance tradeoffs.

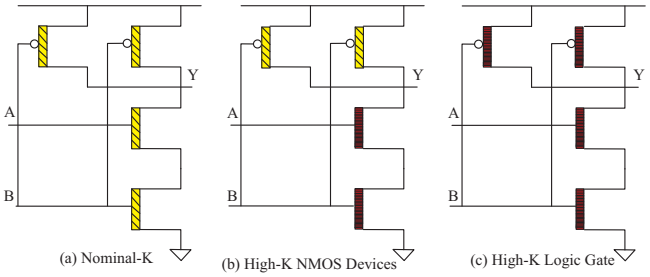



Fig. 2. The dual-K approach for gate leakage optimization [22].

Materials such as, $ZrO_2$, $TiO_2$, BST, $HfO_2$, $Al_2O_3$, SiON, and $Si_3N_4$, have been investigated for use in CMOS technology [16], [21], [29]. The development of various technology for high-K dielectric deposition has progressed [15]. This includes the extension of chemical vapor deposition (CVD) to rapid thermal CVD, rapid plasma-enhanced CVD, and liquid source misted CVD. Other techniques include physical vapor deposition (PVD) [26], jet vapor deposition (JVD) [20], oxidation of metallic films [19], and molecular beam epitaxy [18]. Thus, the fabrication of high-K dielectric based devices and DKCMOS technology based circuits is a reality.

### *B. Compact Modeling for High-K*

While the materials research is in full swing, there is not much research addressing automatic design or synthesis of circuits using high-K devices. For compact modeling based study of high-K non-classical devices using BSIM4, two possible options can be considered: (i) varying the parameter in the model card that denotes relative permittivity (EPSROX) or (ii) finding the equivalent oxide thickness (EOT) for a dielectric under consideration. Both of these approaches ignore several aspects of the physics behind non-$SiO_2$ dielectrics, particularly in the Si/dielectric interface. However, in the absence of device data, this methodology will be a medium to match EDA development with material science trends.

### *C. The Proposed DKCMOS Technology*

In DKCMOS, $SiO_2$ devices, logic gates, or RTL components, are selectively replaced with corresponding high-K elements for gate leakage reduction while maintaining performance. Fig. 2 illustrates the technology. Fig. 2(a) shows a nominal logic gate with all $SiO_2$ devices. In Fig. 2(b), the high leaky NMOS devices are constructed with high-K dielectric. This is more close to well-established dual-$V_{Th}$ technology. In Fig. 2(c) the logic-gate is made of all high-K devices. In this paper it is claimed that a mix of RTL units of type (a) and type (c) can serve gate leakage and performance trade-offs and will go well with industry trend. During the high-level synthesis, selection of high-K and $SiO_2$ RTL modules are performed for trade-offs.

## IV. Gate-Oxide Leakage Modeling

For direct gate tunneling, the tunneling probability of an electron is affected by barrier height, structure and thickness, and the current density of a device is [27], [11], [9], [23]:

$$J_{DT} = \left(\frac{q^3 V_{ox}^2}{16\pi^2 \hbar \phi_B T_{ox}^2}\right) \times exp\left[-\left(\frac{4\sqrt{2m_{eff}}\phi_B{}^{1.5} T_{ox}}{3\hbar q V_{ox}}\right) \times \left\{1-\left(1-\frac{V_{ox}}{\phi_B}\right)^{1.5}\right\}\right], \quad (2)$$

where, $q$ is electronic charge, $V_{ox}$ is voltage across the gate oxide, $\hbar$ is Planck's constant, $T_{ox}$ is electrical equivalent oxide thickness, $\phi_B$ is barrier height for the gate dielectric, and $m_{eff}$ is effective carrier mass. All these parameters are implicitly or explicitly affected by the use of high-K dielectric in the device.

The corresponding gate leakage current can be divided into five components, such as $I_{gs}$ and $I_{gd}$ (components due to the overlap of gate and diffusions), $I_{gcs}$ and $I_{gcd}$ (components due to tunneling from the gate to the diffusions via the channel) and $I_{gb}$, the component due to tunneling from the gate to the bulk via the channel [25], [8]. The tunneling current components are modeled as [8], [1]:

$$\begin{aligned} I_{gs} &= W \cdot DLCIG \cdot A \cdot T_{oxRatioEdge} \cdot V_{gs} \cdot V_{gs}^{'} \\ &\quad \cdot exp\left[-B \cdot T_{ox}.POXEDGE \cdot (AIGSD \right. \\ &\quad \left. -BIGSD \cdot V_{gs}^{'}\right)\left(1+CIGCD \cdot V_{gs}^{'}\right)\Big], \quad (3) \\ I_{gd} &= W \cdot DLCIG \cdot A \cdot T_{oxRatioEdge} \cdot V_{gd} \cdot V_{gd}^{'} \\ &\quad \cdot exp\left[-B \cdot T_{ox}.POXEDGE \cdot (AIGSD \right. \\ &\quad \left. -BIGSD \cdot V_{gd}^{'}\right)\left(1+CIGCD \cdot V_{gd}^{'}\right)\Big], \quad (4) \\ I_{gc0} &= W \cdot L \cdot A \cdot T_{oxRatio} \cdot V_{gs} \cdot V_{aux} \exp\left[-B \cdot T_{ox}\right. \\ &\quad \left.(AIGC - BIGC \cdot V_{oxpinv})\,(1+CIGC \cdot V_{oxpinv})\right] \quad (5) \end{aligned}$$

In the above equations $I_{gc0}$ is $I_{gc}$ $(= I_{gcs} + I_{gcd})$, when $V_{ds} = 0$, based on which $I_{gcs}$ and $I_{gcd}$ can be calculated. It is observed that the body component of the tunneling component is smaller compared to other components and hence neglected in the modeling. The parameters such as, $AIGC$, $BIGC$, $CIGC$, etc. are the empirical parameters and derived experimentally [8], [1]. The gate leakage for a device can then be calculated as follows:

$$I_{gate_{MOS}} = |I_{gs} + I_{gd} + I_{gcs} + I_{gcd} + I_{gb}|\,. \quad (6)$$

The above equations are used to characterize the gate leakage current of individual devices. The effect of varying dielectric material (K) is modeled by calculating an equivalent

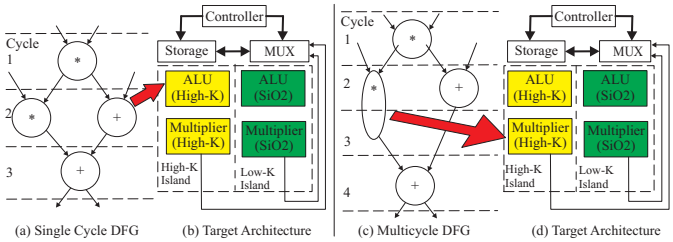



Fig. 3. Data Flow Graph and Target Architecture. In one scenario, (a) and (b), gate leakage reduction can be achieved by executing addition in 2nd cycle in high-K ALU. In another scenario, (c) and (d), gate leakage can be achieved by executing multiplication in 2nd cycle in high-K multiplier [22].

oxide thickness ($T^*_{gate}$) according to the formula [25]:

$$T^*_{ox} = \left(\frac{\mathrm{K}_{gate}}{\mathrm{K}_{ox}}\right) \times T_{ox}, \tag{7}$$

where, $\mathrm{K}_{gate}$ is the dielectric constant of a high-K dielectric and $\mathrm{K}_{ox} = \epsilon_{ox}$ is the dielectric constant of $SiO_2$.

The above models can be used in several ways to estimate gate leakage of CMOS circuits. Numerical equation solvers written in high-level language (like MATLAB or C) can be used to solve simultaneous equations for different input conditions. Commercial tool like MEDICI, Sentaurus, can be used to perform the simulations. Another option is to use the models for each transistor in a circuit level netlist and use SPICE as a nonlinear equation solver to calculate the currents. The SPICE option is followed in this paper, as it is simple to use and easily available and produces accurate results.

## V. Target Architecture, Datapath Specification, and Problem Formulation for Optimization

The datapath is assumed to be specified as a sequencing data flow graph (DFG). Each vertex of the DFG represents an operation and each edge represents a dependency. Each vertex has attributes that specify the operation type. The target architecture model assumed is shown in Fig. 3. The number of $SiO_2$ or high-K resources of the architecture depends on the resource constraints and allocation. Which operation will use a specific resource and at what time is decided through the scheduling and binding algorithm proposed in this paper. Each functional unit feeds one register and has a multiplexer also. The register and the multiplexer belong to same island (high-K or nominal-K) as that of the functional units. A controller decides which functional units are active in each control step and the inactive ones are disabled using the multiplexers. The delay of a control step is dependent on the delays of the functional unit, the multiplexer, and register.

It is assumed that a nano-CMOS circuit is specified as: A RTL library precharacterized for gate leakage and delay for nominal dielectric ($SiO_2$) device and high-$K$ (such as SiON, $Si_3N_4$, $Al_2O_3$, $HfO_2$, etc.) device. A set of resource and time constraints, in which resource constraint ($R_{con}$) is number and type of different RTL components and time constraint ($T_{con}$) is specified as multiple of critical delay $T_{cp}$ for nominal case.

The gate leakage optimization problem during architecture synthesis can be formalized as follows: *Given an unscheduled data flow graph (UDFG) $G_u(V, E)$, it is required to find the scheduled data flow graph (SDFG) $G_s(V, E)$ with appropriate resource binding such that the total gate leakage is minimized and resource and latency constraints are satisfied.*

Assuming $V$ as the set of all vertices and $V_{cp}$ as the set of vertices in the critical path from the source of the DFG to the sink vertex, the gate leakage optimization problem is formulated as follows:

$$\text{Minimize}: \sum_{v_i \in V} P_{gate}(v_i), \tag{8}$$

where, $P_{gate}(v_i)$ is the gate leakage dissipated per sample node $v_i$ of the DFG, such that the following resource and latency constraints, respectively, are satisfied:

$$\sum_{v_i \in V_{cp}} T_i(v_i) \leq T_{con} \left(= D_T \times T_{cp-nominal}\right) \tag{9}$$

$$\text{Allocated } \left(FU_i\left(k, \mathrm{K}\right)\right) \leq \text{ Available } \left(FU_i\left(k, \mathrm{K}\right)\right). \tag{10}$$

The constraints in Eqn. (9), called time constraints, ensure that the critical delay is less than specified time constraint, which is expressed as multiple of critical delay of nominal case. The factor $D_T$ is the time or performance trade-off factor which can be specified by a user. The constraints in Eqn. (10) ensure that the total allocation of the $i^{th}$ resources of type $k$ and made up of transistors of dielectric K denoted as $(FU_i\left(k, \mathrm{K}\right))$ should be less than the total number of corresponding resources available. These are called resource constraints.

## VI. Register Transfer Level Module Library

A RTL module library is created to be used for architecture synthesis and optimization following a three level hierarchy approach. The top level of hierarchy are the RTL components such as adders, subtractors, etc. They in turn use logic gates which are derived from characteristics of nano-CMOS devices.

During its various states of operation, a logic gate presents different dominant gate leakage paths, depending on the combination of inputs. The gate leakage current for a specific state of a logic gate is then calculated by summing the absolute gate currents over all the devices in the logic gate, as both positive and negative gate current contributes to leakage:

$$I_{gate}\mathrm{Logic}_{\mathrm{state}} = \sum_{\forall \mathrm{MOS}_i} \left| I_{gate}\mathrm{MOS}[i] \right|, \tag{11}$$

where the index $i$ identifies a device within a logic gate.

For the next level of the gate leakage characterization, it is assumed that in an $n-$bit RTL unit there are total $n_{total}$ NAND gates out of which $n_{cp}$ are in the critical path. The assumption of NAND realization is based on two reasons: first it is a universal gate and is a low leaky logic gate compared to other gates [25], [17]. In this model the effect of interconnect wires is not considered and focus is on the direct tunneling current and delay of the active units only. This assumption does not affect the gate leakage values as oxide tunneling happens only in the transistors not in the interconnects. However, when the optimization of total power is the objective, the interconnect needs to be accounted [30], which is beyond the scope of this paper. Once the gate leakage

TABLE I
RTL LIBRARY FOR HIGH-K NANO-CMOS TECHNOLOGY AT $45nm$ NODE.

| Datapath Components Library | Nano-CMOS Technology with Different Gate Dielectric | | | | | |
|---|---|---|---|---|---|---|
| | $SiO_2$, K = 3.9 | | SiON, K = 5.7 | | $Si_3N_4$, K = 7.0 | |
| | $P_{gate}$ $(nW)$ | $T_{pd}$ $(ns)$ | $P_{gate}$ $(nW)$ | $T_{pd}$ $(ns)$ | $P_{gate}$ $(nW)$ | $T_{pd}$ $(ns)$ |
| Adder | 19898.5 | 34.92 | 54.48 | 60.19 | 1.02 | 63.94 |
| Subtractor | 21935.8 | 34.92 | 59.29 | 60.19 | 1.14 | 63.94 |
| Multiplier | 271270 | 55.64 | 739.26 | 95.89 | 13.51 | 101.88 |
| Divider | 415990 | 189.07 | 1127.50 | 325.90 | 20.80 | 346.19 |
| Comparator | 47559.8 | 44.86 | 123.67 | 77.28 | 2.39 | 82.13 |
| Multiplexer | 13647.8 | 1.99 | 36.49 | 3.43 | 0.68 | 3.64 |
| Register | 15699.2 | 40.88 | 43.72 | 70.46 | 0.80 | 74.85 |

current per gate is known, the *average* gate leakage current of an $n$-bit RTL unit is calculated:

$$I_{gate}\text{FU} = \sum_{j=1}^{n_{total}} \text{Prob (state)} \times I_{gate}\text{NAND}_j\text{state}. \quad (12)$$

The index $j$ runs for all the NAND gates in a RTL unit. The Prob (state) is the probability of occurrence of an input state.

The critical path delay of an $n$-bit RTL unit using the above NAND gates as building blocks is calculated as follows:

$$T_{pd}\text{FU} = \sum_{j=1}^{n_{cp}} T_{pd}\text{NAND}_j. \quad (13)$$

In Eqn. 13, it is assumed that there are as many low-to high transitions as there are high-to-low, which is the probabilistic case. The average delay of a logic gate is calculated as follows:

$$T_{pd}\text{Logic} = \left(\frac{t_{HL} + t_{LH}}{2}\right), \quad (14)$$

where $t_{HL}$ and $t_{LH}$ are the propagation delay times for high-to-low and low-to-high transitions, respectively.

To characterize the power and delay of a architecture unit, its RTL description is synthesized to logic-level netlist. The logic-level netlist consists of a network of NAND gates. As a trade-off accuracy and time, simulation is performed on the NAND netlist for a series of test vectors. The input test vectors are generated as correlated signals using the autoregressive moving average (ARMA) model as in [24]. Then, the results for all of the simulations is averaged over to calculate the average gate leakage of a architecture unit. This statistical approach was simpler and yet accurate compared to complicated state probability calculations. This approach is sufficient for high-level synthesis and optimization. For a $45nm$ nano-CMOS technology, a characterized RTL library is presented in Table I. However, the proposed algorithm can accept any library and perform synthesis and optimization.

## VII. ALGORITHM FOR GATE LEAKAGE OPTIMIZATION

A tabu search based algorithm that performs simultaneous scheduling, allocation and binding and minimizes the gate leakage is presented. There are several optimization approaches available in literature, but the tabu search based approach is adopted as it takes a more aggressive approach than other search algorithms. The algorithm skips inferior solutions most of the cases other than the cases when it needs to get out of the local optimum. It provides useful solutions to problem in a reasonable time [5], [13], [14], [6].

The pseudocode of the proposed optimization flow is presented in Algorithm 1. For a given set of dual-K options, resource constraints, and time constraint, the algorithm determines an RTL implementation that has minimum gate leakage. The resource constraints $R_{con}$ are expressed as number of different types of resources made of transistors of each dielectric. Time constraint $T_{con}$ is a multiple of nominal critical path delay $T_{cp-nominal}$ and is expressed as a factor $D_T > 1.0$. In order to increase the gate leakage reduction it is needed to ensure that every vertex of a DFG is scheduled and mapped to resources in such a way that utilization of high-K resources increases. In the algorithm, the ASAP and ALAP algorithms are used to get the lower and upper bound on possible control steps in which a vertex can be scheduled. The lower and upper bound are further limited by performing the modification to the schedule to accommodate the resource constraints right at the early stage. Then an initial schedule can be either of the modified ASAP (or ALAP) schedule. The algorithm runs for specified number of iterations to search a final solution. Based on gate leakage of a given solution the algorithm evaluates the neighborhood solutions to reach to a final solution.

In generating a neighborhood solution a vertex is selected based on a priority weight, where higher priority weight gets preference. The priority weight are based on several attributes that can affect gate leakage and delay, such as operation type and corresponding resource needed, number of non-mutually exclusive vertices, and mobility range. Vertices are moved to clock cycles where there are lesser number of vertices of same operation type and same equivalence class. Each time resource allocation and binding are performed and checked if a less leaky resource can be assigned satisfying a time constraint. The allocation and binding are performed using resource allocation Table and resource available Table and following standard approaches. For calculating the total delay of the circuit for a single cycle case the critical path delay is used. 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 dielectric resources will increase the delay which can be compensated using chaining and multicycling. While multicycling increases the number of control steps there were only few operations for which chaining can be implemented. The idea behind using both multicycling and chaining is to ensure that the execution of any operation that is ready i.e. all its predecessors finished execution and has a resource available will start execution.

## VIII. EXPERIMENTAL RESULTS

The overall design flow implemented using C and integrated it into an in-house high-level synthesis framework. The algorithm was experimented with various high-level synthesis benchmark circuits. Two dual dielectric pair $SiO_2$- SiON and $SiO_2$- $Si_3N_4$ are considered. The base case for the experiments was the $SiO_2$ with a thickness of $T_{ox} = 1.4nm$ corresponding to the nominal case of BSIM4.4.0 model. The experiments are performed for various resource and time constraints. The

**Algorithm 1** Tabu Search Based Architecture Synthesis Algorithm Flow for Gate Leakage Optimization

1: Preprocess given behavioral description to construct a sequencing data flow graph (DFG).
2: Perform simulations to estimate gate leakage and delay of register-transfer level (RTL) units.
3: Construct resource allocation table and available resource table based on input resource constraints $R_{con}$.
4: Perform ASAP and ALAP schedules of the input DFG.
5: Determine the number of different resources for each K using the resource allocation table.
6: Modify both ASAP and ALAP schedules obtained above using the number of resources found in previous step.
7: Construct mobility graphs based on the above schedules.
8: Fix the total number of clock cycles as the maximum of modified ASAP and ALAP schedules' control step.
9: Assume initial schedule as the modified ASAP schedule.
10: Perform initial allocation-binding by assigning highest K resource for each vertex.
11: Consider above schedule, allocation, and binding as initial feasible solution $S$ and calculate gate leakage as $P_{gate-S}$.
12: Initialize the number of iteration as $Counter = 0$.
13: **while** $(Counter < Max - Iteration)$ **do**
14: $Counter = Counter + 1$.
15: Generate neighborhood solution $S^*$ for time constraints $T_{con}$ and calculate gate leakage $P_{gate-S^*}$.
16: **if** (Solution $S^*$ is not visited in previous iterations) **then**
17: **if** $(P_{gate-S^*} < P_{gate-S})$ **then**
18: **return** Update $S$ with new solution $S^*$.
19: **else**
20: **return** Discard the solution $S^*$.
21: **end if**
22: **end if**
23: **end while**
24: Obtain the final solution $S$ and corresponding estimates of gate leakage and delay.
25: Postprocess scheduled sequencing DFG to generate gate leakage optimal RTL description.

algorithm can be used for both data intensive and control intensive datapath circuits. However, the experiments are focused on data intensive digital signal processing benchmarks whose applications are immense in day to day life, for example, DVD/MP3 player, mobile phones, etc. For this kind of applications the size of controller is small and hence low-power datapath synthesis is the primary goal.

The experimental results are presented in Table II for a selected benchmark circuits [2]. The algorithm can consider any size of the circuit and provide solutions as long as a datapath uses the RTL components. For each benchmark and for each pair of dual-K, several sets of experiments are performed. For each time constraint three different resource constraints are used. In the first experiment, a smaller number of high-K resources and a higher number of low-K resources are used. In the second experiment a higher number of high-K resources are used than the first experiment whereas in the third experiment a higher number of high-K resources are used as compared to the second experiment. The experimental results presented for each time constraint (expressed as $D_T$) is average result over the resource constraints.

TABLE II
EXPERIMENTAL RESULTS FOR SELECTED CASES AND CIRCUITS

| | $D_T$ | $SiO_2$(K=3.9) - SiON(K=5.7) | | | $SiO_2$(K=3.9) - $Si_3N_4$(K=7) | | |
|---|---|---|---|---|---|---|---|
| | | $P_{gate_{DK}}$ $(\mu W)$ | $T_{cp_{DK}}$ $(ns)$ | $\Delta P$ | $P_{gate_{DK}}$ $(\mu W)$ | $T_{cp_{DK}}$ $(ns)$ | $\Delta P$ |
| | Base Case: $P_{gate_{SK}} = 4632.74\mu W$, $T_{cp_{SK}} = 308.9ns$ | | | | | | |
| A | 1.0 | 2729.4 | 308.9 | 41.3 | 1417.6 | 308.9 | 69.4 |
| R | 1.1 | 1862.3 | 329.4 | 59.8 | 1283.2 | 308.9 | 72.3 |
| F | 1.2 | 1741.9 | 362.2 | 62.4 | 1167.4 | 360.4 | 74.8 |
| | Base Case: $P_{gate} = 3655.68\mu W$, $T_{cp_{SK}} = 290.1ns$ | | | | | | |
| B | 1.0 | 1582.9 | 290.1 | 56.7 | 1144.2 | 290.1 | 68.7 |
| P | 1.1 | 1414.7 | 310.7 | 61.3 | 1082.0 | 290.1 | 70.4 |
| F | 1.2 | 1257.5 | 343.5 | 65.6 | 979.9 | 341.7 | 73.2 |
| | Base Case: $P_{gate} = 4159.12\mu W$, $T_{cp_{SK}} = 308.9ns$ | | | | | | |
| D | 1.0 | 1879.9 | 308.9 | 54.8 | 1439.0 | 308.9 | 65.4 |
| C | 1.1 | 1813.3 | 308.9 | 56.3 | 1339.2 | 308.9 | 67.8 |
| T | 1.2 | 1522.2 | 341.7 | 63.4 | 1172.8 | 308.9 | 71.8 |
| | Base Case: $P_{gate} = 2726.78\mu W$, $T_{cp_{SK}} = 498.4ns$ | | | | | | |
| E | 1.0 | 1497.0 | 498.4 | 45.1 | 1167.0 | 498.4 | 57.2 |
| W | 1.1 | 1385.2 | 531.2 | 49.2 | 107.0 | 530.6 | 59.4 |
| F | 1.2 | 1107.9 | 584.5 | 59.3 | 839.8 | 582.2 | 69.2 |

The experimental results data take into account the gate leakage, and delay of functional units, interconnect units, and storage units present in the datapath circuit. The subscripts SK and DK stand for single dielectric base line cases and dual dielectric, respectively. The percentage reduction in gate leakage for a particular experiment is calculated as follows:

$$\Delta P = \left( \frac{P_{gate}\text{SK} - P_{gate}\text{DK}}{P_{gate}\text{SK}} \right) * 100\%. \qquad (15)$$

The comparison is performed with nominal design as the proposed methodology is a new methodology against the standard practice, which is nominal design. This is the approach followed for comparison in existing literature works, such as [17], [12]. The critical path delay of the circuit is estimated 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 the slowest delay resource for multicycling-chaining case.

From the results, it is observed that reduction in gate leakage for all the benchmarks ranges from $41.3\%$ to $73.6\%$ for $SiO_2$-SiON and $57.2\%$ to $83.2\%$ for $SiO_2$-$Si_3N_4$ for different time constraints ($D_T = 1 \rightarrow 1.4$ i.e. $0\%$ to $40\%$) considered in the experiments. It is also observed that the extent to which gate leakage reduction takes place increases as the number of available high-K resources increase. As the time constraint increases the gate leakage reduction increases. The benchmarks that had more number of operations needing high leaky resources created opportunity to be replaced with high-K resources and more reduction. The percentage reduction in gate leakage averaged over all resources and time constraints considered in the experiments is presented in Fig. 4.

The percentage reduction using DKCMOS technology for gate leakage results in $10 - 20\%$ more reduction compared to subthreshold reduction using DTCMOS technology presented in [17]. The percentage gate leakage reduction exceeds percentage subthreshold leakage reduction of [12] that uses DTCMOS by $20 - 30\%$. Compared to the DKCMOS-based

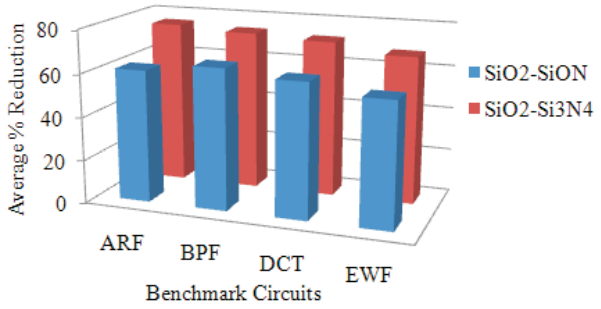



Fig. 4. Average percentage reduction for various circuits.

resource-constrained ILP algorithm [22], the current resource-time-constrained Tabu Search algorithm convergence much faster and can handle larger circuits with comparable results. In the DKCMOS technology, as implemented in the proposed algorithm, is an attractive approach for gate leakage current reduction of nano-CMOS datapath circuits.

## IX. Conclusions and Future Research

This paper presents a new process driven technique called DKCMOS for reduction of gate leakage, important for sub-$65nm$ technology node, during architecture synthesis. The tabu search based algorithm does scheduling and assignment for gate leakage reduction for different resource and time constraints. Experimental results revealed significant reductions in gate leakage with the use of this technology, thus proves its effectiveness. Further exploration of this technique is the incorporation of process variation. The ultimate objective is to extend the work on gate leakage current to provide a broader solution to the problem of power dissipation in all its forms at the architecture level. This will include dynamic power accounting capacitive switching of devices as well as interconnects. Design space involving total power, area, and delay accounting process variation is also being considered. The efficacy of DKCMOS technology for subthreshold leakage and junction tunneling leakage is under investigation.

## X. Acknowledgment

This research is supported in part by NSF award numbers 0702361 and 0854182, and EPSRC grant EP/G032904/1.

## References

[1] Berkley Short-Channel Insulated-Gate Model (BSIM4). `http://www-device.eecs.berkeley.edu/~bsim3/bsim4_get.html`.

[2] Express: High-Level Synthesis Bechmarks. `http://express.ece.ucsb.edu/benchmark/`.

[3] Intel Developer Forum. `http://www.intel.com/idf/`.

[4] Semiconductor Industry Association, International Technology Roadmap for Semiconductors. `http://public.itrs.net`.

[5] I. Ahmad, M. K. Dhodhi, and F. M. Ali. TLS: A Tabu Search Based Scheduling Algorithm for Behavioral Synthesis of Functional Pipelines. *The Computer Journal*, 43(2):152–166, March 2000.

[6] U. Al-Turki, C. Fedjki, and A. Andijani. Tabu search for a class of single-machine scheduling problems. *Computers & Operations Research*, 28(12):1223–1230, 2001.

[7] A. J. Bhavnagarwala, B. L. Austin, K. A. Bowman, and J. D. Meindl. A Minimum Total Power Methodology for Projecting Limits of CMOS GSI. *IEEE Transactions on VLSI Systems*, 8(3):235–251, June 2000.

[8] Y. Cao, T. Sato, D. Sylvester, M. Orshansky, and C. Hu. New Paradigm of Predictive MOSFET and Interconnect Modeling for Early Circuit Design. In *Proceedings of the IEEE Custom Integrated Circuits Conference*, pages 201–204, 2000.

[9] C. H. Choi, K. H. Oh, J. S. Goo, Z. Yu, and W. W. Dutton. Direct Tunneling Current Model for Circuit Simulation. In *Proceedings of International Electron Devices Meeting*, 1999.

[10] D. Dal, A. Nunez, and N. Mansouri. Power islands: a high-level technique for counteracting leakage in deep sub-micron. In *Proceedings of the 7th International Symposium on Quality Electronic Design*, 2006.

[11] M. Depas, B. Vermeire, P. W. Mertens, R. L. V. Meirhaeghe, and M. M. Heyns. Determination of Tunneling Parameters in Ultra-Thin Oxide Layer Poly-Si/$SiO_2$/Si Structures. *Elsevier Solid-State Electronics Journal*, 38(8):1465–1471, August 1995.

[12] C. Gopalakrishnan and S. Katkoori. Knapbind: an area-efficient binding algorithm for low-leakage datapaths. In *Proceedings of 21st International Conference on Computer Design*, pages 430–435, 2003.

[13] C. Gopalakrishnan and S. Katkoori. Tabu Search Based Behavioral Synthesis of Low Leakage Datapaths. In *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 260–261, 2004.

[14] A. Hertz and D. de Werra. The Tabu Search Metaheuristic: How we used it. *Annals of Mathematics and Artificial Intelligence*, 1, 1990.

[15] H. R. Huff and et. al. Integration of high-k Gate Stack Systems into Planar CMOS Process Flows. In *International Workshop on Gate Insulator*, pages 2–11, 2001.

[16] A. Karamcheti, V. Watt, H. Al-Shareef, T. Luo, G. Brown, M. Jackson, and H. Huff. Silicon Oxynitride Films as Segue to the High-K Era. *Semiconductor Fabtech*, 12, 2000.

[17] 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.

[18] A. I. Kingon, J. P. Maria, and S. K. Streifferr. Alternative Dielectrics to Silicon Dioxide for memory and Logic Devices. *Nature*, 406:1021–1038, 2000.

[19] B. H. Lee and et al. Thermal Stability and Electrical Characteristics of Ultrathin Hafnium Oxide Gate Dielectric Reoxidized with Rapid Thermal Annealing. *Applied Physics Letters*, 77:1926–1928, 2000.

[20] T. P. Ma. Making Silicon Nitride a Viable Gate Dielecrtric. *IEEE Transaction on Electron Devices*, 45:680–690, 1999.

[21] L. Manchanda, B. Busch, M. L. Green, M. Morris, R. B. van Dover, R. Kwo, and S. Aravamudhan. High K gate Dielectrics for the Silicon Industry. In *International Workshop on Gate Insulato*, pages 56–60, 2001.

[22] S. P. Mohanty. ILP Based Gate Leakage Optimization Using DKCMOS Library during RTL Synthesis. In *Proceedings of the 9th International Symposium on Quality of Electronic Design (ISQED)*, pages 174–177, 2008.

[23] S. P. Mohanty and E. Kougianos. Modeling and Reduction of Gate Leakage during Behavioral Synthesis of NanoCMOS Circuits. In *Proceedings of the 19th International Conference on VLSI Design*, 2006.

[24] S. P. Mohanty, N. Ranganathan, and S. K. Chappidi. ILP models for simultaneous energy and transient power minimization during behavioral synthesis. *ACM Transaction on Design Automation of Electronic Systems*, 11(1):186–212, January 2006.

[25] V. Mukherjee, S. P. Mohanty, and E. Kougianos. A Dual Dielectric Approach for Performance Aware Gate Tunneling Reduction in Combinational Circuits. In *Proceedings of the 23rd IEEE International Conference of Computer Design (ICCD)*, pages 431–436, 2005.

[26] W. J. Qi and et al. Ultrathin Zirconium Silicate Film With Good Thermal Stability for Alternative Gate Dielectric Application. *Applied Physics Letters*, 77:1704–1706, 2000.

[27] K. Roy, S. Mukhopadhyay, and H. M. Meimand. Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits. *Proceedings of the IEEE*, 91(2):305–327, February 2003.

[28] X. Tang, H. Zhou, and P. Banerjee. Leakage power optimization with dual-$v_{th}$ library in high-level synthesis. In *Proceedings of the 42nd Design Automation Conference*, pages 202–207, 2005.

[29] M. Yang and et al. Performance Dependence of CMOS on Silicon Substrate Orientation for Ultrathin and $HfO_2$ Gate Dielectrics. *IEEE Electron Device Letters*, 24(5):339–341, May 2003.

[30] L. Zhong and N. K. Jha. Interconnect-aware low-power high-level synthesis. *IEEE Transactions on CAD of Integrated Circuits and Systems*, 24(3):336–351, March 2005.