Evaluation of Low-Leakage Design Techniques for Field Programmable Gate Arrays

Arifur Rahman and Vijay Polavarapuv
Department of Electrical and Computer Engineering, Polytechnic University,
Brooklyn, NY 11201

ABSTRACT

In this paper we evaluate the trade-offs between various low-leakage design techniques for field programmable gate arrays (FGPAs) in deep sub-micron technologies. Since multiplexers are widely used in FPGAs for implementing look up tables (LUTs) and connection and routing switches, several low-leakage implementations of pass transistor based multiplexers and routing switches are proposed and their design trade-offs are presented based on transistor-level simulation, physical design, and impact on overall system performance. We find that gate biasing, the use of redundant SRAM cells, and integration of multi-Vt technology are ideal for FPGAs, and they can reduce leakage current by 2X-4X compared to an implementation without any leakage reduction technique. For some of the potential low-leakage design techniques being evaluated in our study, the impact on chip area is very minimal to an increase of $15\% - 30\%$.

Keywords

FPGA, leakage power, and multiplexer.

1. INTRODUCTION

As the logic density and system performance of field programmable gate arrays (FPGAs) continue to increase, power dissipation has become an important design metric for many mainstream low-power applications. In older technology generations, with minimum feature size of 0.18 μm or larger, dynamic power dissipation has been the dominant component of total power dissipation. However, due to the scaling of threshold voltage, channel length, and gate oxide thickness, leakage current is expected to increase significantly in current and future technology generations [1]. Since leakage power is roughly proportional to the number of (off) transistors and FPGAs generally require higher number of transistors to implement a logic function compared to ASICs, leakage power is expected play an important role in future FPGA designs utilizing sub-micron technologies. It is projected that for a high-performance system in $0.1\mu m/0.7V$

technology, more than 50% of total power dissipation will be due to leakage power at 110 C [2].

Although there have been extensive studies on low-power design techniques for ASIC and custom logic design, there has been little work on low-power FPGAs. In [3], power reduction techniques in programmable interconnection network have been proposed by considering hierarchical interconnection topology and using low-swing drivers. Recently, low-power design techniques based on high-level synthesis have also been evaluated [4]. Most of the earlier works on low-power FPGA design have been focused on dynamic power reduction. However, in future technology generations, integration of dynamic power reduction techniques alone will not be sufficient to minimize overall power dissipation; leakage power reduction techniques have to be incorporated as well.

The commonly used low-leakage circuit design techniques, presented in literature, are generally suitable for CMOS logic gates [2, 5, 6]. In FPGAs, pass transistor based programmable multiplexers are the key building blocks for implementing LUTs, input/output connection switches, and routing switches. The programmable routing or connection switches account for $60\% - 70\%$ of total power dissipation and a comparable fraction of chip area [4, 7, 8, 9]. Any potential low-leakage design technique for FPGAs must be suitable for integration with programmable routing or connection switches. Since the programmable switches can be implemented by pass transistors or pass transistor-based multiplexer, we evaluate the potentials of various low-leakage design techniques by integrating them in nMOS pass transistor based multiplexer design. Leakage reduction using redundant SRAM cells for fine-grain controllability of off-transistors, substrate biasing, gate biasing, and multi-Vt devices are presented. Smartspice [10] simulations and analytical models are used to estimate average leakage power. The physical design issues associated with the selective use of multi-Vt transistors in programmable routing switches are evaluated by placement and routing experiments using Versatile Place and Route (VPR) tool [11]. This paper is organized as follows: in Section 2 an overview of low-leakage techniques is provided, followed by their detailed design trade-offs in Section 3. Key findings of this study are summarized in Section 4.

2. LEAKAGE POWER IN FIELD PROGRAMMABLE GATE ARRAYS

In SRAM-based FPGAs, programmable interconnections not only limit overall system performance, they account for

a significant fraction of total power dissipation. Based on recent studies, it has been found that $\sim 65\%$ of total power dissipation is associated with programmable interconnection, and it is followed by programmable clock networks, I/O buffers, and logic blocks, which account for $\sim 20\%$, $\sim 10\%$, and $\sim 5\%$ of total power dissipation, respectively [7]. Similar observation can also be made in other works presented in [4, 8, 9]. Although earlier studies have focused on modeling power dissipation and low-power design techniques at circuit and synthesis level [3, 4, 9], minimization of leakage power has received very little attention for FPGAs. In deep sub-micron technologies, if low-leakage design techniques are not incorporated in FPGAs, leakage power could be the dominant component of total power dissipation.

There are several components to leakage current in a sub-micron CMOS technology. They include reversed biased p-n junction leakage, subthreshold leakage, leakage current due to drain induced barrier lowering (DIBL), gate induced drain leakage (GIDL), gate oxide tunneling, etc. [2, 5]. Subthreshold leakage current is generally the most dominant component of total leakage current. To minimize leakage power, various circuit design techniques have been proposed. Some of these techniques include the use of sleep transistors to limit off-current in inactive state of a circuit, transistor stack effect to reduce off-current in a series connected off transistors, input vector activation to set circuits to low-leakage state during idle cycles, dynamic control of threshold voltage by substrate biasing, the use of high threshold voltage (high-Vt) transistors in non-timing critical circuits, etc. [5, 6]. Some of these techniques are suitable for CMOS logic circuits and may not be applicable to FPGA design, where pass transistor based circuits are widely used for implementing various key building blocks such as look-up tables, switch blocks, etc. Low-leakage techniques such as multi-Vt based design, dynamic Vt control, etc. are quite generic and can be incorporated in FPGAs. However, their area and performance trade-offs need to be evaluated. To assess the impact on total chip area due to the integration of various low-leakage techniques, chip area models, presented in [12], are used. These models use system-level interconnection prediction techniques to estimate the wiring requirements and chip area in FPGAs [13].

To reduce leakage power, high-Vt transistors can be used for implementing a fraction of the nMOS pass transistor based programmable switch blocks. As a result, there is a direct impact on overall performance and timing-driven placement and routing. To find a viable multi-Vt solution, both leakage reduction and performance penalty need to be examined. High-Vt devices can also be used for implementing programmable SRAM cells that generate DC signals for configuration of switch or connection blocks. Subthreshold voltage control based on body biasing is also very effective for reducing leakage current. Body biasing may require multi-well configuration and impose additional layout design rules, which can lead to increased chip area and longer wire length.

To assess the effectiveness of various low-leakage techniques, we consider the design implementation of an nMOS pass transistor-based multiplexer. A multiplexer is chosen because it is widely used in SRAM-based FPGAs, and it is a key element for implementing LUTs and switch blocks. The trade-offs of various low-leakage design techniques are evaluated by comparing the layout area, leakage power, and sys-

tem performance. The low-leakage design techniques, evaluated for programmable nMOS pass transistor based multiplexer design, are presented in the following sections:

3. EVALUATION OF LOW-LEAKAGE DESIGN TECHNIQUES

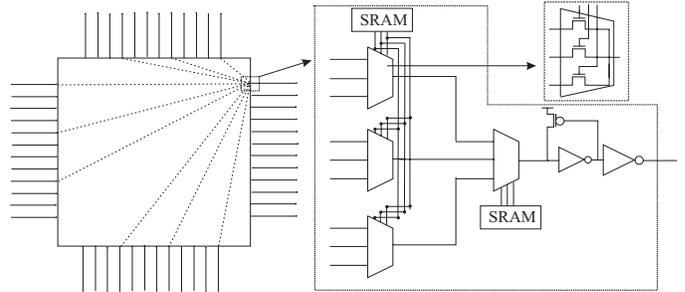### 3.1 Leakage Power Modeling in Pass Transistor Based Multiplexer

Figure 1: A unidirectional switch block composed of nMOS pass transistor based 2-stage multiplexers.

We present a simple model for leakage power estimation in pass transistor based multiplexers. This model can provide insight into optimization of multiplexer design for low leakage current. A unidirectional switch block, constructed with nMOS pass transistor based two-stage multiplexers with a level restoring buffer, is shown in Figure 1. It is assumed the input voltage level of a multiplexer is rail voltage VDD or VSS. The total leakage current in nMOS pass transistor based multiplexer consists of leakage current in pass transistors, buffer, and memory cells. It also depends on the values of input signals and internal node voltage levels. For example, if all input signals are driven to either VDD or VSS, the total leakage power associated with off pass transistors is negligible. On the other hand, in the case of a single stage multiplexer, the leakage power is maximum when all inputs are driven to either VDD or VSS and the input of the enabled input-to-output path is driven to VSS or VDD, respectively. These two cases are illustrated in Figure 2.
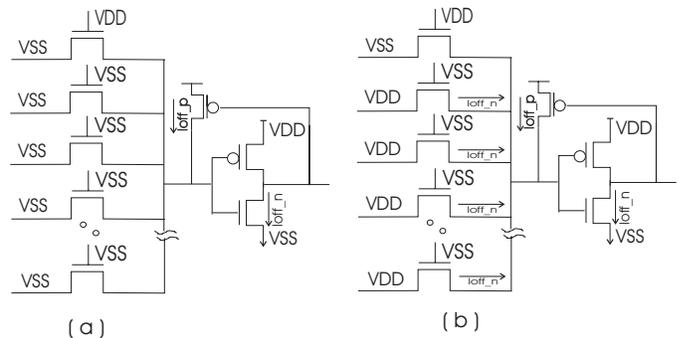

Figure 2: (a) The conditions for minimum leakage power and (b) maximum leakage power in nMOS pass transistor based single-stage multiplexer. For clarity, one of the inverters in the level-restoring buffer is not shown.

If the inputs are randomly driven to high or low voltage levels, average leakage power can be estimated by taking into account all possible input combinations and taking an average of them. It can be shown that average leakage power in M-input one-stage pass transistor based multiplexer is given by

$$P_{lmux} \simeq M I_{poff} \frac{V_{DD}}{2} + N_{mem} I_{moff} V_{DD} + I_{boff} V_{DD}, \quad (1)$$

where I_{poff}, I_{moff}, and I_{boff} are the off current of pass transistor, SRAM memory cell, and buffer, respectively, and N_{mem} is the number of programmable SRAM memory cells. We have verified our model by Smartspice simulations where the inputs are set randomly to high or low voltage level and taking an average over 100-150 sets of random inputs [12]. The average leakage power of a multi-stage pass transistor based multiplexer can be found by combining the leakage power components of all single-stage multiplexers. In the following sections, similar methodologies are used to determine the average leakage power in low-leakage pass transistor based multiplexers. Due to the availability of Hspice models and technology library, TSMC's 0.18 μm/1.8V technology is used to evaluate the potentials of various low-leakage design techniques [14].

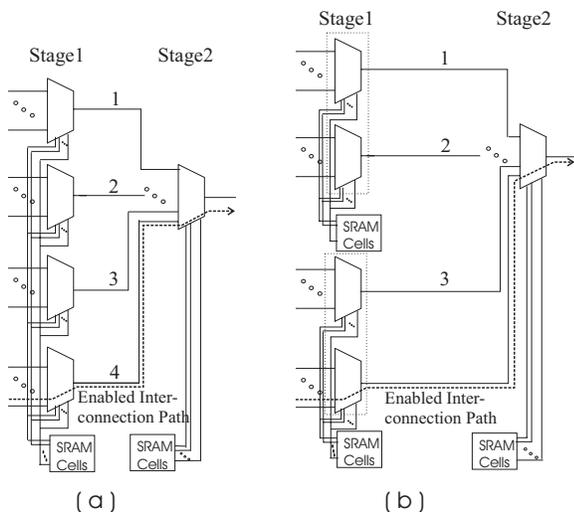## 3.2 Redundancy in Circuit Design

Figure 3: (a) A two-stage implementation of a pass transistor based multiplexer. (b) A two-stage multiplexer with redundant memory cells. For simplicity, the level-restoring buffer is not shown.

In this section we propose a leakage reduction technique based on the use of redundant memory cells. A high fan-in multiplexer can be implemented in multiple stages to reduce parasitic capacitance in intermediate or output nodes and to minimize the number of programmable memory cells. Let us consider a two-stage implementation of a pass transistor based multiplexer, as shown in Figure 3(a). It is composed of several smaller multiplexers, and to reduce the total number of SRAM cells, the same SRAM cell configures one pass transistor from each multiplexer in stage 1. As the result, whenever there is an enabled input-to-output path, the intermediate nodes, such as node 1, 2, 3, and 4 are driven to VDD

or VSS, and the drain-to-source voltage, V_{DS} of all disabled pass transistors is VDD or VSS. Under these conditions, the disabled pass transistors with $V_{DS} = V_{DD}$ contribute to leakage power. We propose a low-leakage technique where the pass transistors that are not included in an enabled interconnection path are turned off and the subthreshold leakage current through a series connected nMOS devices determines intermediate node voltage. Although such implementation requires additional SRAM cells for granular controllability, they reduce the leakage current of pass transistors in disabled input-to-output paths. These SRAM cells can be implemented using high-Vt devices since their function is to generate DC signals to configure the switch blocks. As a result, the impact on total leakage power due to integration of additional SRAM cells can be very minimal.
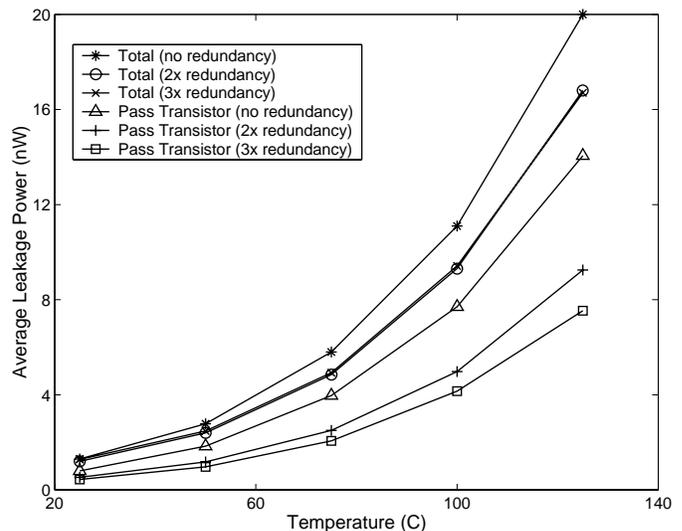

Figure 4: The total leakage power and the leakage component associated with nMOS pass transistors in a 30X1 2-stage multiplexer without and with 2x and 3x SRAM cells in stage 1.

Let us consider the case where the number of SRAM cells in stage 1 multiplexers is doubled, as shown in Figure 3(b). Now, the voltages on intermediate nodes such as nodes 1, 2, and 3 are determined by the off current through the transistor stacks, formed by off pass transistors from stage 1 and stage 2 multiplexers. This type of transistor stacks is common in CMOS logic gates. If all transistors in a transistor stack (i.e. series connected nMOS or pMOS transistors) are in off state, there is a significant reduction in leakage current compared to that of an individual transistor due to stack effect [2]. By turning off pass transistors in disabled interconnection paths and taking advantage of stack effect, significant reduction in leakage power, associated with pass transistors, is feasible. To evaluate the effectiveness of incorporating redundant SRAM cells, a 30x1 multiplexer, consisting of 6 5x1 stage 1 multiplexers and one 6x1 stage 2 multiplexer, is implemented. The stage 1 multiplexers are partitioned into two or three groups. As a result, they require 2X or 3X more SRAM cells in stage 1, respectively. We've implemented these multiplexers in 0.18μm/1.8V technology and the average leakage power is estimated by Smartspice simulations. The impact on av-
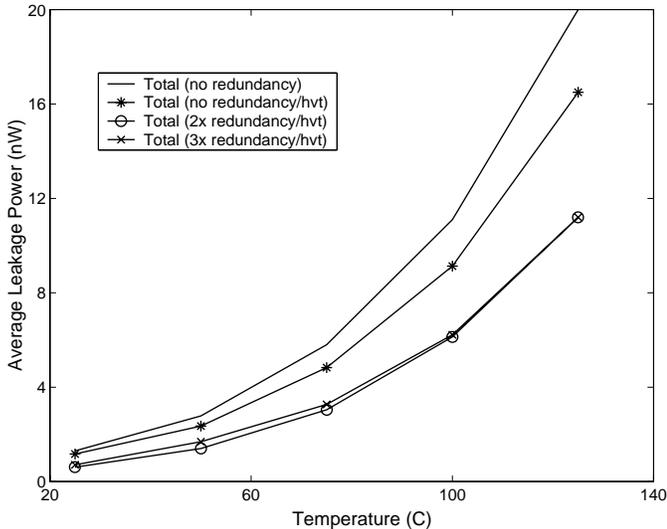
Figure 5: The total leakage power of a 30X1 2-stage multiplexer without and with 2x and 3x additional SRAM cells in stage 1. High Vt devices are used in SRAM cells for further reduction of leakage power.

Figure 6: The area of a 2-stage 30X1 multiplexer without any redundant SRAM cell(x1), and with 2x, 3x, and 6x additional SRAM cells in stage 1.

erage leakage power is projected in Figure 4 and Figure 5. The area is estimated by layout of multiplexers and switch blocks in Cadence's Virtuoso Layout Editor, and the area of a 30x1 multiplexer utilizing redundant SRAM cells is shown in Figure 6. With 2x and 3x additional SRAM cells in stage 1 multiplexers, the total area increases by $30\% - 50\%$.

It can be seen from Figure 4, there is approximately a 2X reduction in pass transistor's average leakage power due to granular control of their gate voltage with redundant SRAM cells. In our example, the leakage power in SRAM cells and level-restoring buffer is significant, and the use of additional SRAM cells diminishes the advantage gained by granular control of pass transistor's gate voltage.

The average leakage power is also estimated for a multiplexer where high-Vt devices have been used in SRAM cells, and the simulation results are presented in Figure 5. In our analysis, the Vt of transistors in SRAM cells is 25% higher than typical devices. As shown in Figure 5, using high-Vt transistor based redundant SRAM cells, leakage power can be reduced by approximately 2X. The use of additional SRAM cells leads to larger programmble switch block and chip area. Since the connection and routing switches can be implemented by pass transistor based multiplexers and they account for $60\% - 70\%$ of overall chip area, the impact on total chip area due to the addition of redundant memory cells is $15\% - 30\%$. Although we have considered a 2-stage implementation of a multiplexer, the low-leakage technique presented here can be used for any arbitrary number of stage implementation of pass transistor based multiplexer.

3.3 Dual-Threshold Voltage Devices

In this section, we evaluate the potentials of incorporating high-Vt devices in programmable interconnection paths. Unlike the previous section, where high-Vt devices are used in SRAM cells, here we consider replacing typical nMOS pass transistors in routing switches with high-Vt nMOS devices. Based on our earlier experience with placement and
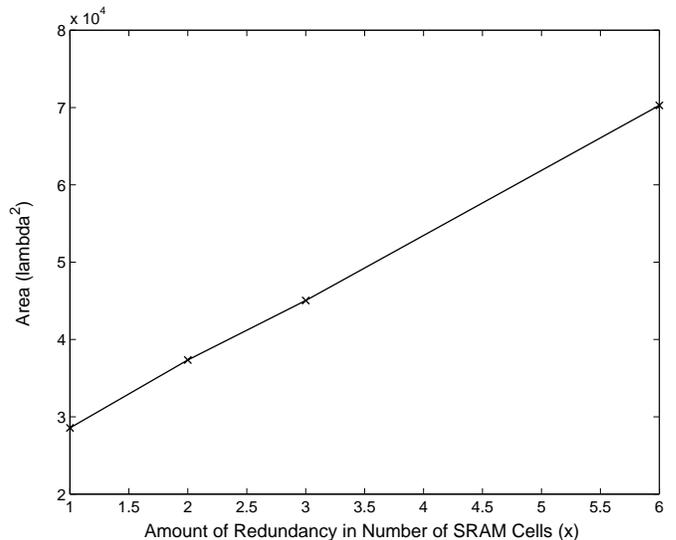
routing experiments with benchmark circuits in VPR, we observed that typical routing utilization was in the range of $50\% - 70\%$ in FPGA [13]. In other words, $30\% - 50\%$ resources in routing and connection switches are not being used. It may be feasible to reduce leakage power by selective use of high-Vt devices in unutilized routing and connection switches. Such implementation will affect the timing characteristics of programmable interconnections. Also, it is not known apriori which programmable interconnections will be used during placement and routing. However, given an assortment of routing resources, a timing-driven placement and routing tool should be capable of mapping timing-critical nets to high-performance programmable interconnection paths. The non timing-critical nets can be mapped to interconnection paths composed of typical and/or high-Vt devices.

To examine the impact on placement and routing due to selective use of high-Vt devices in routing switches, we have performed a series of placement and routing experiments in VPR where both the value of Vt and the fraction of routing resources utilizing high-Vt devices are parameterized. Our goal is to evaluate an optimum FPGA design incorporating high-Vt pass transistors in programmable switch blocks with minimal impact on performance. In our placement and routing experiments, programmable interconnect delay is $\sim 60\%$ of total signal delay. For simplicity, a configurable logic block (CLB) with 4 4-input LUTs, and programmable interconnections with four unit long wiring segments and pass transistor-based Wilton's Switch topology are considered. The percentage of pass transistors in routing switches utilizing high-Vt devices is varied, and high Vt devices are emulated by increasing the on-resistance of pass transistors in the architecture file of VPR. Simulation results of critical signal delay and programmable net delay as function of these parameters at 25C are shown in Figure 7 for a benchmark circuit that requires a 40×40 CLB array, implemented in $0.25 \mu m/2.5$ V technology. In region I, all pass transistors in routing switches have typical values of Vt and in
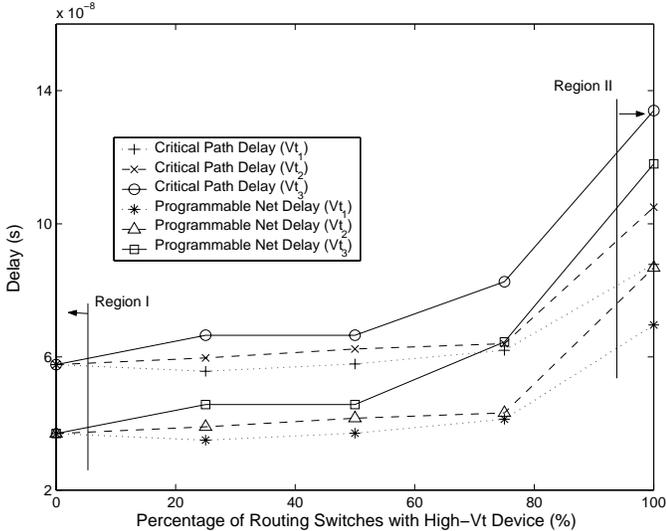
Figure 7: Critical path delay (logic and net delay) and programmable net delay vs percentage of routing transistors utilizing high-Vt devices. Vt_1, Vt_2, and Vt_3 are approximately 25%, 50%, and 100% higher, respectively, than typical value of threshold voltage in a $0.25\mu m/2.5V$ technology.
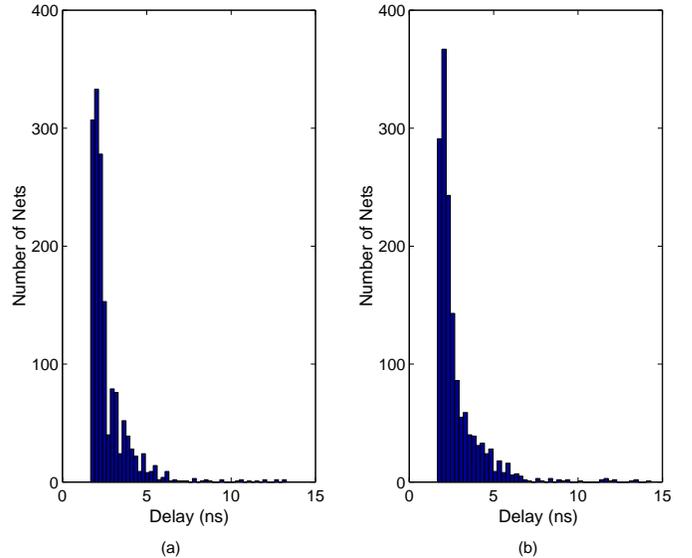

Figure 8: Delay Histograms: In (a), all pass transistors in programmable interconnection paths use typical devices. In (b), 50% pass transistors in programmable interconnection paths use high Vt devices, where the Vt is $\sim 50\%$ higher compared to that of part (a).

region II, all pass transistor have the highest possible value of Vt. Based on routing and placement experiments and for the FPGA architecture under consideration, we find that roughly 40% pass transistors' equivalent on-resistance can be increased by $15\% - 40\%$ without any significant impact on critical net delay and routability, and it corresponds to a $20\% - 45\%$ increase in threshold voltage. To gain insight, the delay histograms are plotted in Figure 8. Figure 8(a) represents the typical case. In Figure 8(b), 50% pass transistor switches' equivalent resistance has been increased by 50%. Based on delay distribution, the average delay with high-Vt devices is 10% higher. However, the delay distribution of timing-critical nets is not affected significantly because 50% routing resources are still formed by typical devices, and the timing-driven physical design tool prioritizes the mapping of timing-critical nets to high-performance programmable interconnection paths.

We also notice that as more high-Vt devices are introduced, routing utilizations of high-performance interconnection paths increase by $3\% - 6\%$. This is because the timing-driven router utilizes as many high-performance programmable interconnection paths as possible compared to low-performance programmable interconnection paths. As a result of higher routing utilization, dynamic power dissipation could increase, and these physical design mechanisms need to be evaluated in details. Due to exponential dependency of subthreshold leakage current on threshold voltage and for the technology being considered in this study, our preliminary analysis indicates that with selective use of high-Vt devices, it is feasible to reduce the leakage power of programmable switch blocks by $\sim 40\%$ without any significant impact on system performance or chip area.

3.4 Body Biasing

Body biasing in an effective technique for controlling the value of threshold voltage and to reduce subthreshold leakage current. Recently, various approaches to body biasing have been proposed for high-performance circuits. A self-adjusting threshold voltage scheme can monitor the total leakage current and set the body bias to appropriate levels so that the monitored leakage current is comparable to a target value [6]. A standby power reduction scheme, during a prolonged period of inactivity, can be implemented by substrate biasing or input vector activation [2]. To implement some of these techniques, a dual-well process is needed since it is desirable to control the threshold voltage of both nMOS and pMOS transistors within a well-defined region. In this section, using nMOS pass transistor based multiplexer as an example, we consider the design trade-offs of a body biasing technique that is suitable for FPGA.

Our proposed body biasing technique requires additional control logic to identify the wells that should be biased to low-leakage state and to select appropriate body voltage levels. It is also feasible to replace the control logic by analyzing the configuration memory bits in software and selecting appropriate body bias voltage levels. The different body voltages can be generated locally or distributed throughout the chip in a way similar to VDD and VSS distribution.

A large (high fan-in) multiplexer can consist of several smaller (low fan-in) multiplexers. It is natural to partition these smaller multiplexers in their own wells so that their body voltages can be controlled independently. The finer the partition, the higher the flexibility to control body bias and reduce leakage current. However, in multi-well implementation, the multiplexer's area increases due to compliance with additional design rule such as well-to-well separation. In Figure 9, layout of a 64x1 multiplexer in p-well and twin-well process technology are presented. In Figure 9(b) there are 5 separate wells and the area increases by 33% compared

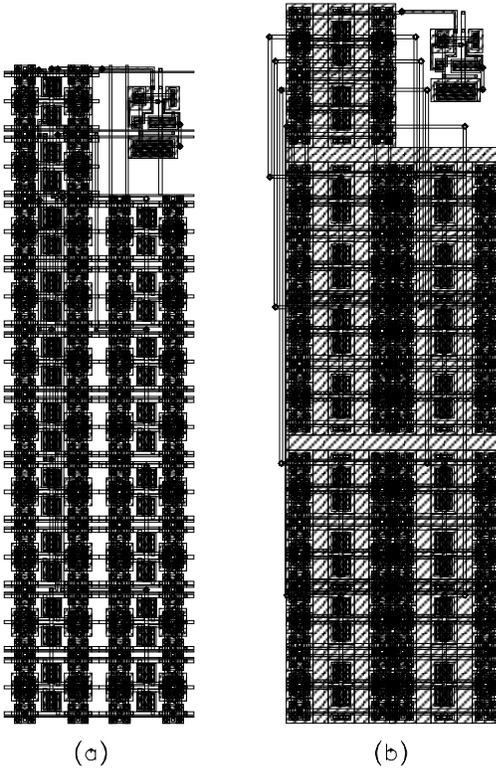(a)   (b)

Figure 9: The layout of a 64x1 input multiplexer implemented in (a) p-well and (b) twin-well process technologies. It is implemented in two stages using 8 8x1 multiplexers in stage 1 and one 8x1 multiplexer in stage 2. In (b) the 8x1 multiplexers are partitioned into 5 separate wells for body biasing.

to conventional implementation, presented in Figure 9(a).

In our case study, we consider the same 30x1 multiplexer, examined in earlier sections and group the 6 5x1 stage 1 multiplexers in different wells. The second stage multiplexer is implemented in its own well. The leakage power for a multi-well implementation of multiplexer with body biasing is shown in Figure 10. In our case study, the six 5x1 first-stage multiplexers, within a two-stage 30X1 multiplexer, are divided into two and six separate wells. The corresponding area for multi-well implementation is presented in Figure 11. As expected, the higher the number of wells, the larger the area due to compliance with well-to-well design rules.

3.5 Gate Biasing

To reduce leakage current in programmable nMOS pass transistor-based multiplexers, a negative gate voltage can be applied to turn off nMOS pass transistors instead of a 0 V signal. Since the subthreshold leakage current is exponentially related to the gate-to-source voltage, roughly an order of magnitude reduction in leakage current is feasible by $\sim 100\ mV$ reverse gate bias for a subthreshold slope of
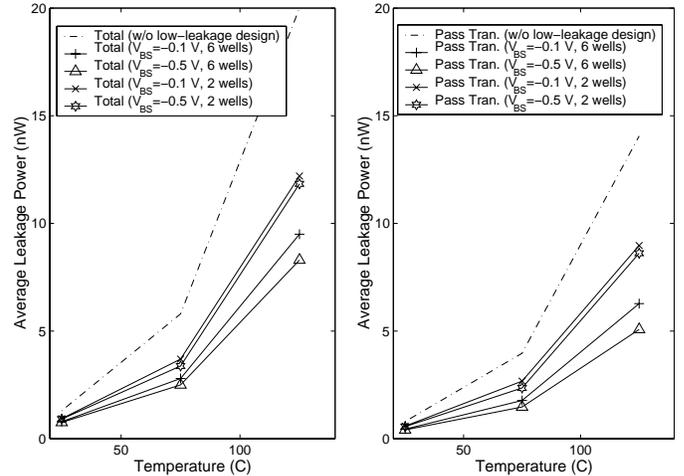

Figure 10: (a) Total leakage power in low-leakage multiplexers implemented in multiple well configurations. (b) The leakage power associated with the pass transistors. The legends correspond to the number of wells in the first stage of the 2-stage 30x1 multiplexer.

$\sim 100\ mV/dec$. The negative gate voltage can be generated from SRAM cells, and the overhead associated with this technique is negligible. In the case of negative gate biasing, rail-to-rail voltage in SRAM cells will be higher, their leakage current will increase, and it could limit the minimum value of nMOS pass transistor's gate voltage in off state. In addition, when an nMOS transistor's gate-to-drain is reverse biased, there is an additional component of leakage current, gate induced drain leakage (GIDL), which could also limit the amount of negative gate bias. In TSMC's $0.18\mu m/1.8V$ technology, a 4X reduction in leakage power associated with a 30X1 multiplexer is feasible with a gate voltage of $-0.1V$. Simulation results of average leakage power of a 30x1 multiplexer with negative gate biasing technique are presented in Figure 12.

4. SUMMARY AND CONCLUSIONS

In this paper, a high-performance 0.18 $\mu m/1.8V$ CMOS logic technology is used to evaluate the effectiveness of various low-leakage techniques for programmable multiplexers and switch block design. It is always feasible to trade off performance by selecting a low-leakage logic technology [1]. Such options are not explored in our study because of their lower performance. Based on circuit simulations, physical design, and place and route experiments, we find that it may require integration of several low-leakage techniques to reduce total leakage power significantly. The most promising techniques with low overheard appear to be the use of high-Vt devices in programmable SRAM cells and their selective use in programmable routing switches, negative gate biasing, and the use of redundant SRAM cells. Although gate biasing appears to be quite promising, gate oxide tunneling, GIDL, etc. could limit the use of this technique in deep sub-micron technology. On the other hand, it also creates opportunities to optimize device geometry and parameters to limit gate or gate-induced leakage so that negative gate
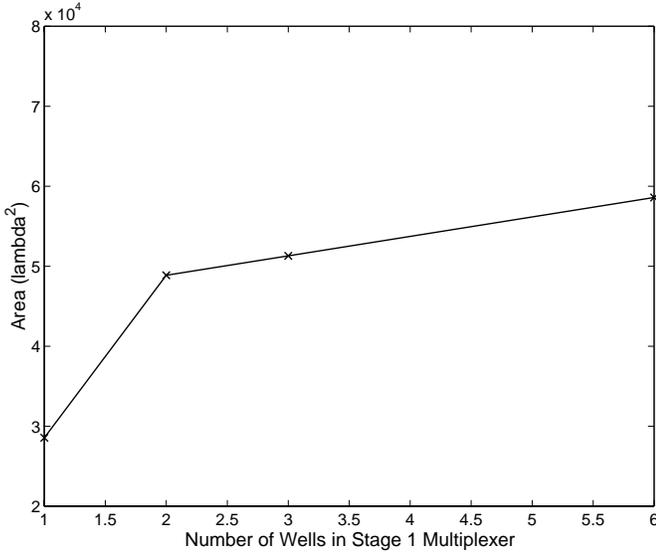
Figure 11: The area of a 2-stage 30X1 multiplexer as the function of number of wells in stage 1 multiplexer, in units of design rule parameter, labmda.

Low Leakage Technique	Leakage Power Reduction	Area Increase
Redundant SRAM Cells	2X	1.3X-2X
Dual-Vt Device	1.7X	None
Body Bias	1.7X-2.5X	1.6X-2X
Gate Bias	2.5X-4X	Negligible

Table 1: The leakage power reduction and area penalty of pass transistor based multiplexer/switch blocks due to various low-leakage design techniques. The comparison is made with respect to a typical implementation without any low-leakage design technique.

biasing can be an effective leakage reduction technique with negligible overhead in SRAM based FPGAs. The trade-offs of various leakage reduction techniques are summarized in Table 1 for implementing programmable pass transistor based multiplexers or switch blocks. These techniques, except gate biasing, have implications on system performance due to longer metal wire-length or lower performance of programmable switch blocks. Although our analysis and simulation results are based on 0.18 μm/1.8 V technology, if sub-threshold leakage is the dominant component of total leakage current, the proposed low-leakage design techniques should be applicable to current and future technology generations.

5. REFERENCES

[1] *2001 International Roadmap for Semiconductors.*

[2] V. De, Y. Ye, A. Keshavarzi, S. Narendra, J. Kao, D. Somasekhar, R. Nair, and S. Borkar, "Techniques for Leakage Power Reduction," in *Design of High-Performance Microprocessor Circuits*, A. Chandrakasan, W. Bowhill, and F. Fox (editors), IEEE Press, NJ.

Figure 12: The total leakage power of a 30x1 multiplexer, with super cut-off operation, using a negative gate voltage.

[3] V. George, H. Zhang, and J. Rabaey, "The Design of a Low Energy FPGA," in *Proceedings of International Symposium on Low Power Electronics and Design*, 1999.

[4] F. Li, D. Chen, L. He, and J. Cong, "Architecture Evaluation for Power Efficient FPGAs," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, 2003, pp. 175–184.

[5] J. Kao, S. Narendra, A. Chandrakasan, "Sub-threshold Leakage Modeling and Reduction Techniques," Part I and Part II, 2002 ICCAD Tutorial. Also available at http://www-mtl.mit.edu/research/icsystems/research/presentations.html.

[6] T. Kuroda and T. Sakurai, "Low-Voltage Techniques," in *Design of High-Performance Microprocessor Circuits*, A. Chandrakasan, W. Bowhill, and F. Fox (editors), IEEE Press, NJ.

[7] E. Kusse, *Ananlysis and Circuit Design for Low Power Programmable Logic Module.* M. S. Thesis, Department of Electrical Engineering and Computer Science, University of California at Berkeley, 1997.

[8] L. Shang, A. S. Kaviani, and K. Bathala, "Dynamic Power Consumption in Virtex-II FPGA Family," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, 2002, pp. 157–164.

[9] K. K. Poon, A. Yan, and S. J. Wilton, "A Flexible Power Model for FPGAs," in *Proceedings of Field-Programmable Logic and Applications*, 2002, pp. 312–321.

[10] Smartspice, SILVACO International, Santa Clara, CA 95054, USA.

[11] V. Betz and J. Rose, "VPR: A new packing, placement and routing tool for FPGA research," in *Proceedings of International Workshop on Field Programmable Logic and Applications*, 1997.
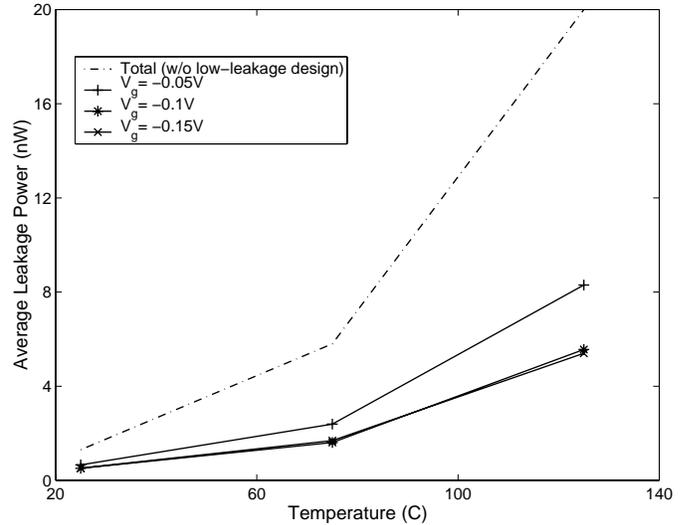
[12] A. Rahman, "Models for Full-Chip Power Dissipation

in Field Programmable Gate Arrays and The Impact of Subthreshold Leakage Current," *Proceedings of the International Conference on VLSI 2003*, June 23-26, Las Vegas, Nevada.

[13] A. Rahman, S. Das, A. Chandrakasan, and R. Reif, "Wiring requirement and three-dimensional integration technology for field programmable gate arrays," *IEEE Transactions on VLSI*, vol. 11, no. 1, pp. 44–54, 2003.

[14] NCSU CDK, North Carolina State University Cadence Tool, available at www.cadence.ncsu.edu.