POWER REDUCTION TECHNIQUES IN FIELD PROGRAMMABLE GATE ARRAY: A SURVEY

[1]Pooran Singh, [2]Santosh Kumar Vishvakarma

[1]Nanoscale Devices, VLSI/ULSI Circuit & System Design Lab, School of Engineering,
[2]Electrical Engineering Discipline, Indian Institute of Technology (IIT), Indore, MP, India - 453 441
E-Mail: [1]phd11120203@iiti.ac.in, [2]skvishvakarma@iiti.ac.in

Abstract

FPGAs are less power-efficient than custom ASICs. So in this paper we reviewed some latest research work on power reduction in FPGA at 90nm, 45nm and 28nm CMOS technology node, we examined that using dual VT and fine-grained VDD techniques static power reduces to 64% and 95 % respectively. While using LOPASS, glitch reduction and clock gating techniques dynamic power reduces to 61.6%, 30%, and 50% respectively. We also discussed about novel non-classical ultra-low power MOS devices like Multi-Gate FET and Tunneling FET which we can use in FPGAs to yield low power with high performance and high speed.

Keywords: FPGA; power reduction; CAD; SRAM; non-classical MOS devices.

I. INTRODUCTION

Field Programmable Gate Arrays (FPGAs) are integrated circuits that can be programmed to implement any digital circuit. The main difference between FPGAs and conventional fixed logic implementations, such as Application Specific Integrated Circuits (ASICs), is that the designer programs the FPGA on-site [1-3]. For fixed logic implementations, the designer must create a layout mask and send it to a foundry to be fabricated. Creating a layout is labour-intensive and requires expensive Computer Aided Design (CAD) tools and experienced engineers. Moreover, fabrication takes several weeks and costs over a million dollars for the latest process technologies. This expense is compounded when bugs are found after fabrication and the chip must be re-fabricated. Using an FPGA instead of a fixed logic implementation eliminates these Non Recurring Engineering (NRE) costs and significantly reduces time-tomarket.

This makes FPGAs ideal for prototyping, debugging, and for small to medium volume applications. Many studies have focused on improving the speed and area overhead of FPGAs due to slower speed [4]. Important advancements include cluster-based logic blocks [5], which improve speed by grouping the basic logic elements of the FPGA into clusters with faster local interconnect; embedded memories [6-7], which improve the speed and area overhead for applications with storage requirements. A significant number of studies have also focused on

faster; more area efficient programmable routing resources [8]. Important advancements have also been made to the FPGA CAD tools, which are used to map applications onto the programmable fabric of the FPGA. In recent years, however, the focus of the research has been shifting to lowering power consumption. As Complementary Metal Oxide Semiconductor (CMOS) process technology scales down, the power density continues to increase due to higher chip operating frequencies, higher total interconnect capacitance per chip, and increasing leakage [9]. The International Technology Roadmap for Semiconductors (ITRS) has identified low-power design techniques as a critical technology need for FPGA based designs. This literature survey therefore focus on the techniques which are really helpful in recent time for reduction of power in FPGAs.

This paper is organized as follows. Section 2 describes the basic architecture of an FPGA; Section 3 summarizes the sources of power dissipation; Section 4 covers the various power reduction techniques; Section 5 shows ultra low power novel non classical devices which is utilizes in circuits like SRAM; Section 6 covers the comparison between various power reduction techniques and gives an efficient technique for power reduction. Finally, section 7 concludes the paper.

II. FPGA ARCHITECTURE

A basic symmetric FPGA is shown in Fig. 1. The FPGA architecture is very regular in structure [10-11].

It is made up of two main components; Configuration Logic Blocks (CLBs) and routing resources. The logic blocks implement the functionality of the given circuit, while the routing resources provide the connectivity to implement the logic. The logic blocks and the routing resources are configurable, so that they can be programmed to implement any logic. Though many types of architectures have been experimented with, the most popular one is the Static Random Access Memory (SRAM) based architecture; the architecture connected with programmable logic interconnects shown in Fig. 2.
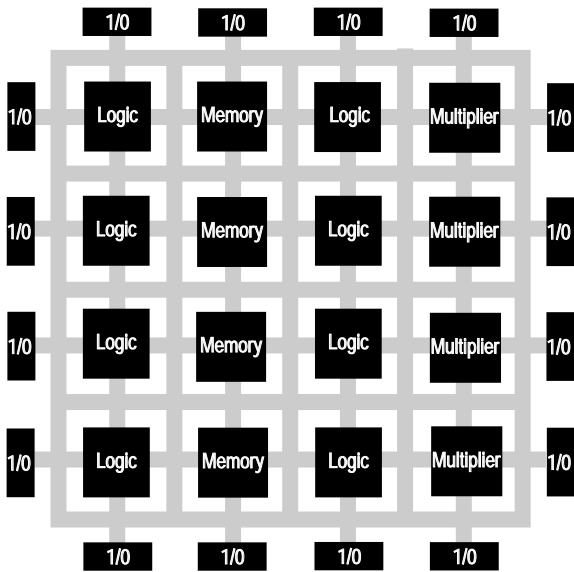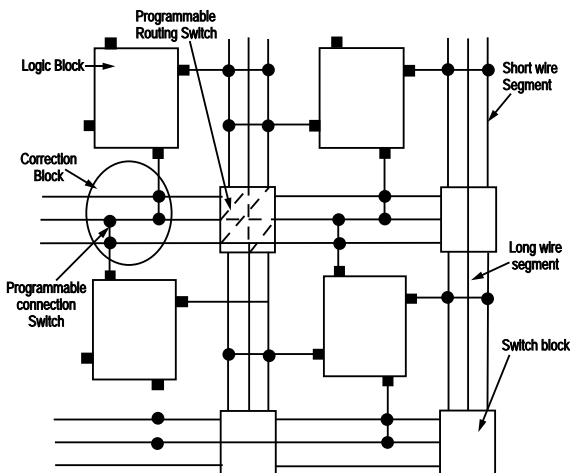

Fig. 1. A basic FPGA [10]

Fig. 2. FPGA basic architecture [11]

A. Logic block

The logic block of the SRAM based FPGA is Look-Up-Table (LUT) based and composed of Basic Logic Elements (BLE). LUT is an array of SRAM cells to implement a truth table. Fig. 3 shows a 4-input LUT which is used to design 7 input LUT for modern FPGA design as shown in Fig. 4 [12]. It has four 4-input LUTs with 64 SRAM cells and 63 nos. of 2-input multiplexers to select one of the SRAM cells. The selection is done by the 7 select signals which are the inputs of the LUTs through the different 2 input multiplexers, which serve as inputs to the truth-table. Each BLE consists of a k-input LUT, flip-flop and a multiplexer for selecting the output either directly from the output of LUT or the registered output value of the LUT stored in the flip-flop. Fig. 5 shows the basic logic element [13]. Previous works have shown that the 4-input LUT is the most optimum size as far as logic density, and utilization of resources are concerned, and this has been widely used.
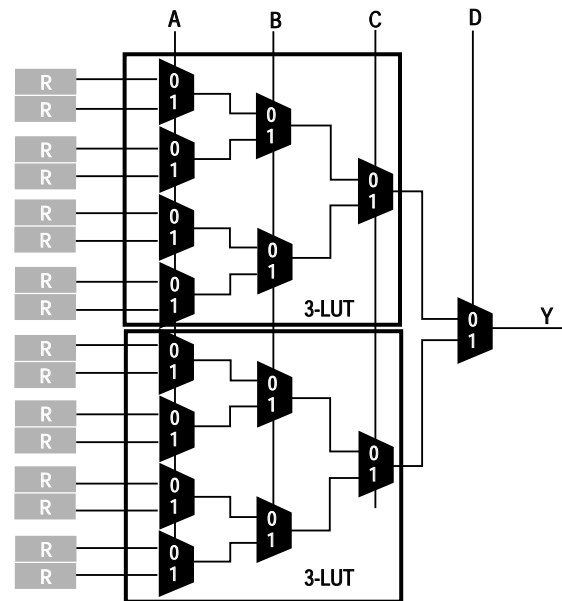

Fig. 3. 4-input LUT from 16 SRAM cell [12]

In today digital system design due to high end uses FPGA vendors provided a comprehensive alternative to FPGAs for large volume demands called structured ASICs [14-15]. Structured ASICs offer a complete solution from prototype to high-volume production, and maintain the powerful features and high-performance architecture of their equivalent FPGAs with the programmability removed. Structured
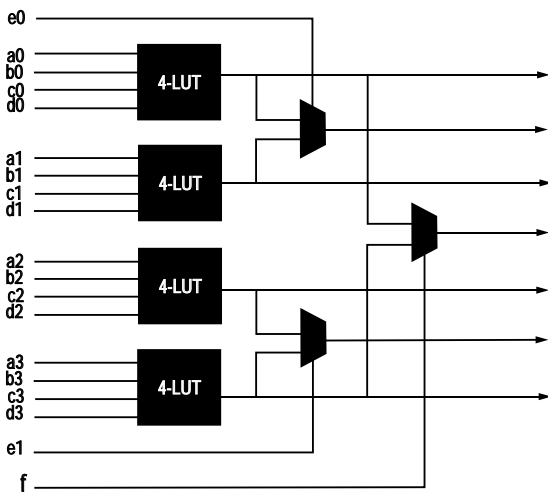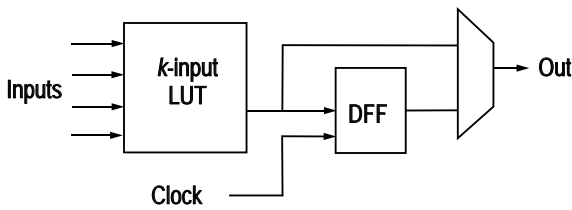
Fig. 4. Seven input LUT [12]

Fig. 5. Basic logic element (BLE) [13]

ASIC solutions not only provide performance improvement, but also result in significant high-volume cost reduction over FPGAs. The programming technologies for logic and interconnect resources other than SRAM are flash memory [16], or antifuse [17-18]. While SRAM-based FPGAs offer in-circuit reconfigurability at the expense of being volatile, while antifuse are write-once devices. Flashbased FPGAs provide an intermediate alternative by providing reconfigurability as well as non-volatility. The most popular programming technology in FPGAs is SRAM. FPGAs usually include embedded memory, DSP blocks, Phase-locked loops (PLLs), embedded processors, and other special feature blocks, as shown in Fig. 6. These features allowed FPGAs to be an attractive alternative for some System on Programmable Chip (SoPC) designs.
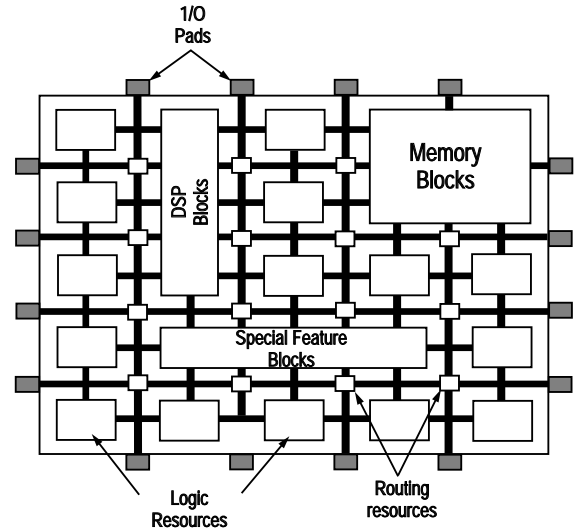

Fig. 6. Modern FPGA fabric [16]

III. POWER CONSUMPTION SOURCES

Due to the dramatic increase in portable and battery-operated applications, lower power consumption has become a necessity in order to prolong battery life. Power consumption is an important part of the equation determining the end product's size, weight, and efficiency. FPGAs are becoming more attractive for these applications due to their shorter product life cycle. FPGAs are programmable, so they allow product differentiation. Selecting an appropriate FPGA architecture is critical in achieving the best static and dynamic power consumption. We are dealing with power optimization & reduction techniques so before going to study about those techniques In, first we will discuss about the types of power consumption FPGA, in modern FPGA [19-22]. There are mainly two components whose power consumption is the major concern in FPGA: [14-15] static and dynamic power.

B. Static power

Static power is the minimum power required to maintain a device in user mode while I/O's and logic cells are not switching. SRAM FPGAs consume significantly more static power than do their flash and antifuse counterparts. At room temperature, this difference may be memory tens of milliamps; at higher operating temperatures, the difference can exceed hundreds of milliamps. This higher current draw drastically impacts the operating life of any battery-supplied system. Both digital and analog logic

consume static power. The sources of static leakage current popular in 28-nm transistors are shown in Fig.7 and in table 1.
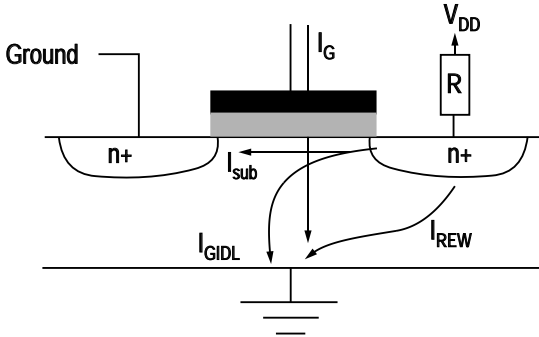

Fig. 7. Sources of transistor leakage [19]

Table 1 Main sources of transistor leakage [19]

Main Sources of Leakage	Impact	Mitigation Techniques
Sub-threshold leakage (I_{sub})	Dominant	Lower voltage, Higher voltage threshold,Longer gate length
Gate direct-tunneling leakage(I_G)	Dominant	High-k metal gate (HKMG)
Gate-Induced gate leakage (IGIDL)	Small	Dopant profile optimization
Reverse-biased junction leakage current (IREV)	Negligible	Dopant profile optimization

C. Dynamic power

Dynamic power is the power consumed when I/Os and the logic cells are switching, and is a function of capacitance, operating voltage, and switching frequency. If we look at the dynamic power contribution alone, all advanced FPGA technologies exhibit similar performance; however, the dynamic power also contains the static power consumption component. SRAM FPGAs consume more power under static conditions and at high temperatures than flash and antifuse FPGAs; therefore, the overall dynamic power consumed by SRAM FPGAs is significantly higher. Dynamic power consumed through the operation of the device caused by signals toggling and capacitive loads

charging and discharging. As shown in equation 1, the main variables affecting dynamic power are capacitance charging, the supply voltage, and the clock frequency.

$$P_{dynamic} = \left[\frac{1}{2} CV^2 + Q_{shortcircuit} V \right] f\ activity \quad ...(1)$$

Where $P_{dynamic}$ represents dynamic power consumption, C is the load capacitance; activity is the switching activity factor, $Q_{short\text{-}circuit}$ is the short circuit charge.

IV. POWER REDUCTION TECHNIQUES

The power reduction techniques are developed in various aspects; the power reduction is done at static (leakage current reduction) power, dynamic power reduction through; glitch removal, clock gating, power gating, improved switching activity, pipelining, and guarded evaluation etc.

A.   Static power reduction techniques

Static power is dominated by leakage current in various forms: sub-VT leakage, junction leakage i.e., source/drain, well, and triple-well junctions, Gate-Induced Drain Leakage (GIDL) and gate leakage. Here we are going to discuss several leakage current reduction techniques.

Dual Threshold Transistor Stacking (DTTS)

This is a technique for static power reduction in nanoscale CMOS circuits [23]. As we know to limit the ever increasing trend of energy and power dissipation in CMOS technology, supply voltage has to be continuously scaled. The amount of power reduction depend not only on supply voltage (V_{DD}) but also on the threshold voltage (V_{th}) to sustain the reduction of component delay, which is crucial for high speed digital circuit design. Continuous scaling down of these parameters poses several challenges to circuit designers. Particularly threshold voltage reduction leads to increase in sub-threshold leakage current leading to tremendous increase of static power consumption in CMOS circuits, which is otherwise considered as a negligible contributor to the overall power consumption. The performance of a CMOS tri-State buffer using Multi Threshold CMOS (MTCMOS) and Forced Transistor Stacking (FTS) leakage reduction techniques are analyzed and a new technique called Dual Threshold Transistor Stacking (DTTS) for efficient reduction of leakage power is introduced. From the results, it is

Table 2 Performance comparison of various circuit techniques [23]

Circuit description	$P_{dyn, avg}$ (nW)	$P_{static, avg}$ (pW)	Delay (ns)	Power delay product (fJ)	% Reduction	
					P_{avg}	P_{static}
Conventional	19.78	97.98	0.107	0.0021	–	–
MTCMOS	09.75	0.87	0.287	0.0027	50	Standb y
FTS	18.43	85.16	1.152	0.0212	6.8	13.0
[23]	18.21	76.33	1.113	0.0202	7.9	22.0

observed that this technique combines the advantage of multiple threshold and stacking effects in MOSFETs. Table 2 shows the comparison of leakage power between different circuits.

Active leakage power optimization for FPGAs

There are several works which was done on the field of subthreshold leakage current reduction in CMOS transistor. It is well-known that the leakage power consumed by a digital CMOS circuit depends strongly on the state of its inputs. Leakage reduction technique leverages a fundamental property of basic FPGA logic elements LUTs that allows a logic signal in an FPGA design to be interchanged with its complemented form without any area or delay penalty. Jason H. Anderson et al. [24] defines a property to select polarities for logic signals so that FPGA hardware structures spend the majority of time in low leakage states. In an experimental study, they optimize active leakage power in circuits mapped into a state-of-theart 90nm commercial FPGA. Results show that the approach reduces active leakage by 25%, on average.

The second approach [25] to leakage optimization consists of altering the routing step of the FPGA CAD flow to encourage more frequent use of routing resources that have low leakage power consumptions. Such "leakageaware routing" allows active leakage to be further reduced, without compromising design performance. Combined, the two approaches offer a total active leakage power reduction of 30%, on average.

Dual VT approach

A baseline LUT that conducted to identify the consumption of different blocks composing the architecture has been analyzed [26]. It has been applied a dual-VT approach and a self-reverse biasing technique extensively to reduce the stand-by leakage

consumption by almost one order of magnitude and the active leakage one by 64%. Table 3 shows the performance.

Table 3 LUT leakage and delay performance [26]

LUT	Active Power (nW)	Stand-by (nW)	Delay (ps)
Standard	99.22	65.8	202
Low leakage	35.94 (-64%)	8.16 (-88%)	213 (+5.4%)

There is more work has been done in the field of LUT leakage reduction for FPGAs [27]. In this work it is discussed that programmable FPGA LUTs that can operate in two different modes: high-performance or low-power. So the selection between the two modes is realized by an extra SRAM cell that can be shared by a number of LUTs. In high-performance mode, the LUTs provide similar power and performance to a conventional LUT. In low-power mode, one LUT reduces leakage by 53%, while another reduces leakage by 53% and 80% when out-putting a logic0 and logic-1 respectively, which can lead to an average leakage reduction of up to 76%.

Activity packing in FPGAs

For the detection of activity profiles in MTCMOS-based FPGA structures for leakage power mitigation two packing algorithms is presented [28]. The first algorithm is a connection based packing technique by which the proximity of the logic blocks is accounted for, and the second algorithm is a logic-based packing approach by which the weighted hamming distance between the blocks activities is considered. After analyzing both algorithms they have applied to a number of FGPA benchmarks for verification. Once the activity profiles are realized, sleep transistors are

carefully positioned to contain the clustered blocks that share similar activity profiles. Then the percentage of the leakage power savings for each of the two algorithms is evaluated. The two algorithms suggest an average standby leakage reduction of 15% for the FPGA benchmarks.

Fine grained-VDD

Low power FPGA architecture [29] is generated with the use of fine-grained VDD control scheme called microVDD-hopping, 4 CLB's are grouped into one block where VDD is shared as shown in Fig. 8. In the micro-VDD-hopping scheme, VDD of each block is varied between the higher VDD (VDDH) and the lower VDD (VDDL) spatially and temporally to achieve lower power, while keeping performance undegraded. Simulation using 90nm CMOS technology shows that a leakage power reduction of 95% can be achieved, when this method is used.
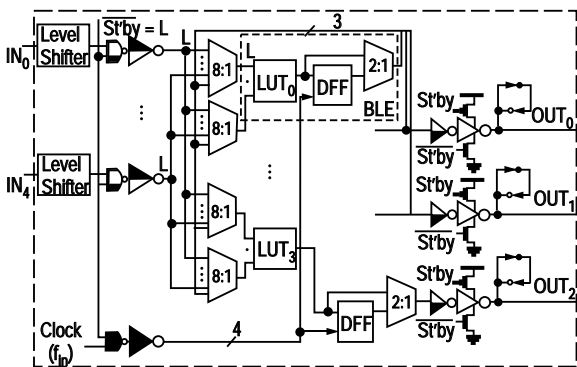
interconnect leakage can be obtained with 8.73% worst case average delay penalty.

Input vector reordering

A technique of input vector reordering for leakage power reduction in FPGAs [31] which is based on the state dependency property of leakage power. A pin reordering algorithm where the sub-threshold and gate leakage power components are taken into consideration to find the lowest leakage state for the FPGA pass-transistor multiplexers in the logic and routing resources without incurring any physical or performance penalties. This methodology is applied to several FPGA benchmarks, and an average leakage savings of 50.3% is achieved in a 90-nm CMOS process. The trend of leakage current is increasing with the technology scales down which is shown in Fig. 9 and in Fig. 10 it is shown that the leakages current dominant states in FPGA.

Fig. 8. Schematic of the CLB, four BLE's are clustered into one CLB [29]

A dual-threshold FPGA routing design

A dual-threshold FPGA routing design for subthreshold leakage reduction [30] is the technique in which the routing designs are based on the dual-threshold to reduce leakage power. Alternating between buffers and pass transistors, they analyzed the percentage constitution of low-Vth and high-Vth transistors as a function of the leakage reduction and delay increment tradeoff. By routing a suite of Microelectronics Center of North Carolina (MCNC) benchmark circuits, it is shown that an average savings of about 28.83% (as high as 48.46%) in total
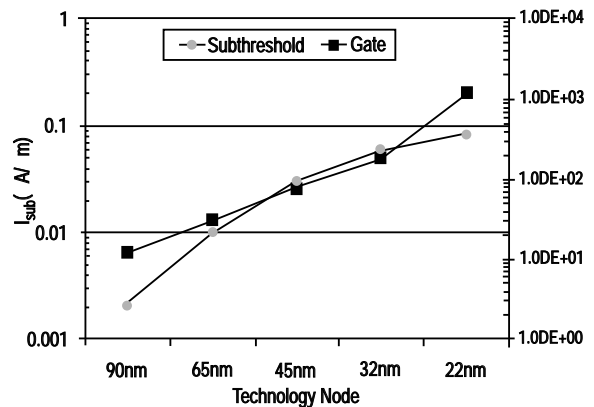

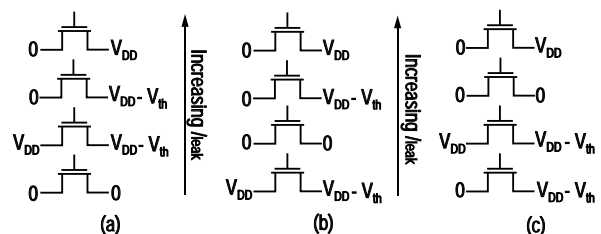Fig. 9. Leakage current versus technology [31]

Fig.10. Total leakage-dominant states in FPGA pass-transistor devices (a) 90 nm. (b) 65nm and 45 nm. (c) 32 nm and 22 nm [31]

Leakage reduction in FPGA routing multiplexers

It is the technique for reducing the leakage current which is based on the architecture such as multiplexer based interconnect matrix of an FPGA which consumes most of the static power. Leakage reduction in FPGA routing multiplexers [32] investigates reducing leakage power in unused FPGA routing multiplexers by controlling their inputs at the deep submicron 22nm technology node. HSPICE simulation using Berkeley Predictive Technology Models (BPTM) on different sizes and topologies of routing multiplexers shows that the minimum leakage vector at the 22nm technology node significantly varies from that at 65nm node. This is due to higher gate leakage and output stage loading effects. The application of this vector results in 20% more leakage power saving as compared to the existing approaches. This technique saves significant leakage power because most of the routing multiplexers are unused in an FPGA.

FPGA leakage power reduction using CLB-clustering

CLB-clustering design technique employs VDD programmable and power gating methods to reduce leakage in stand-by mode [33]. In compared to the conventional VDD programmable architecture, leakage power of CLB-clustering architecture is 0.1% of the leakage power in no gating FPGA and 8% of leakage power in conventional programmable VDD FPGA.

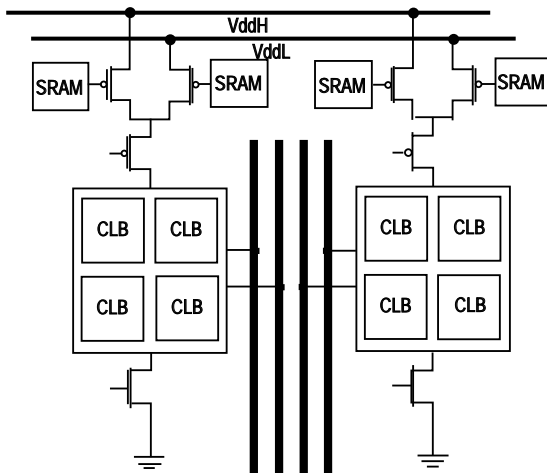Fig. 11 shows the logic block architecture of the CLB-clustering FPGA [33]. Four CLBs are clustered into one VDD island where the same VDD is used. VDD of the block of 4 CLBs is either high VDD (VDDH) or low VDD (VDDL). The island is applied VDDH when high performance is required and is changed to VDDL if the blocks operate at lower speed. If the number of CLBs decreases in a block, the finer control is possible but area and delay overhead increase. Here, four is selected to keep the chip area and delay overhead below 5%. One CLB includes 4 BLEs and has 5 inputs and 3 outputs. One BLE consists of one LUT, one D-Flip Flop and one 2:1 MUX. This configuration is chosen because it is one of the best configurations for delay, area and logic utilization.

Off-path leakage power aware routing

The technique [34] is focused on reducing the leakage power in routing resources, and more specifically, the leakage power dissipated in the used part of FPGA device, which is known as the active leakage power. It is observed that the leakage power in off-path transistors takes up most of the active leakage power in multiplexers that control routing, and strongly depends on hamming distance between the state of the on-path input and the states of the off-path inputs. Hence, an off-path leakage power aware routing algorithm has been used to minimize hamming distance between the state of on-path input and the states of off-path inputs for each multiplexer. Experimental results on MCNC benchmark circuits show that, compared with the baseline Versatile Place and Route (VPR) technique, the off-path leakage aware routing alogrithm can reduce active leakage power in routing resources by 1679% and the increment of cirtical path delay is only 1.06%.

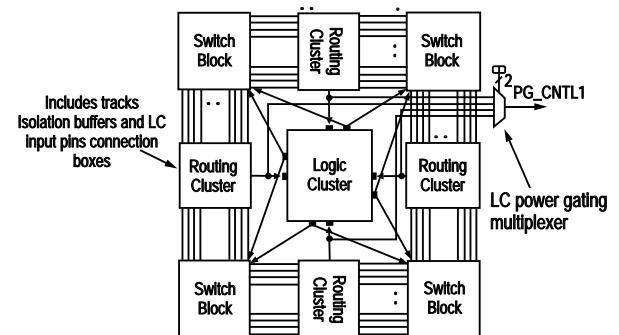Fig. 11. Architecture of CLB-clustering [33]

Fig. 12. Dynamic power gating architecture for a logic cluster and its routing channels [35]

Fig.12 shows an example of the basic power gating architecture. In this figure, a logic cluster has four input pins, with the required four connection boxes, distributed uniformly on its four sides. Each of the connection boxes can be used either to route an endpoint of a connection to the corresponding input pin, or to route a power control signal to the cluster. If a power control signal is to be routed, then the corresponding input pin of the cluster is not used. The outputs of the connection boxes are fed as inputs to the power gating multiplexer. This multiplexer selects the input pin that will be used as the power control signal for the cluster and the bounding routing channels; this signal is labeled PG_ CNTL1 in the figure. PG_CNTL1 could drive the gate of the sleep transistor to turn it off for low-leakage mode, or to turn it on for normal circuit activity.

B. Dynamic power reduction techniques

Dynamic power consumption is defined as the power consumed while the clock is running and the external inputs are switching. Dynamic power has several components, mainly capacitive-load charging and discharging (internal and I/Os) and short-circuit current. Most of the dynamic power is consumed by charging and discharging capacitance, internal and external to the device. If the device is driving heavily with many I/O loads, the dynamic current due to the I/O becomes a substantial part of the entire power consumption. Since the voltage VDD is fixed, internal dynamic power reduction is achieved by

(a) Decreasing the average logic-switching frequency.

(b) Reducing the amount of logic switching at each clock edge.

(c) Reducing the propagation of the switching activity.

So there are several techniques which we are going to discuss here for to reduce the dynamic power.

Reducing switching activity

There are some power reduction techniques which we are going to discuss here from which we decrease the switching activity and interconnect delay and other capacitive power consumption on charging discharging. The dynamic power consumption in the fabric of FPGAs is analyzed by taking advantage of

both simulation and measurement [36]. It identifies important resources in the FPGA architecture and obtains their utilization, using a large set of real designs. Then, using a number of representative case studies the switching activity corresponding to each resource is calculated. Finally combining effective capacitance of each resource with its utilization and switching activity share of power consumption is estimated [37]. According to the results, the power dissipation share of routing, logic and clocking resources are 60%, 16%, and 14%, respectively.

Low Power Architectural Synthesis System (LOPASS)

LOPASS [38] includes three major components:

(a) A flexible high-level power estimator for FPGAs considering the power consumption of various FPGA logic components and interconnects

(b) A simulated-annealing optimization engine that carries out resource selection and allocation, scheduling, functional unit binding, register binding, and interconnection estimation simultaneously to reduce power effectively and

(c) A co-family based register binding algorithm and an efficient port assignment algorithm that reduce interconnecitons in the data path through multiplexer optimization.

The experimental result shows that LOPASS produces promising results on latency optimization, 61.6% power consumption is reduced.

Fig. 13(a) shows a typical buffered FPGA routing switch [40-42]. It consists of a multiplexer, a buffer and SRAM configuration cells and a transistor-level view of a switch with 4 inputs is shown in Fig. 13(b). NMOS transistor trees are used to implement multiplexers in FPGAs. Routing switch inputs are tolerant to "weak-1" signals. That is, logic- 1 input signals need not be rail; it is acceptable if they are lower than this. This is due to the level-restoring buffers that are already deployed in FPGA routing switches [see Fig. 13(b)]. It permits such switches to produce "weak-1" signals. The main exceptions to this observation are switches that drive inputs on logic blocks. Based on these three observations, they propose the new switch design shown in Fig. 14. The switch includes n-MOS and p-MOS sleep transistors in parallel (MNX and MPX). Fig. 15 as the alternate design of Fig. 14, the alternate
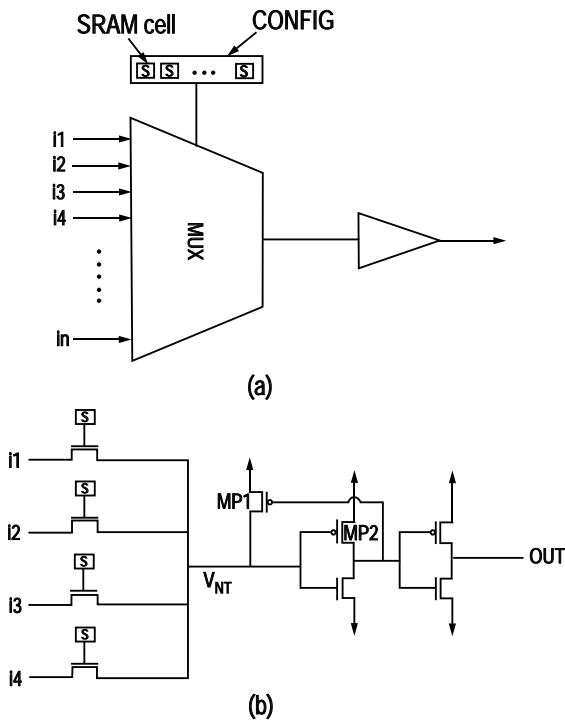
Fig. 13. Traditional routing switch: abstract and transistor-level views [40-42] (a) Routing switch (abstract) and (b) 4input routing switch (transistor-level view)

design offers a different leakage/area tradeoff versus the basic design that is, the alternate design requires less area, but is likely more "leaky". For both designs, a straightforward extension can be made to realize a different leakage/speed tradeoff. Specifically, one can apply the sleep structure discussed above to the multiplexer that precedes the buffer, as shown in Fig. 16.
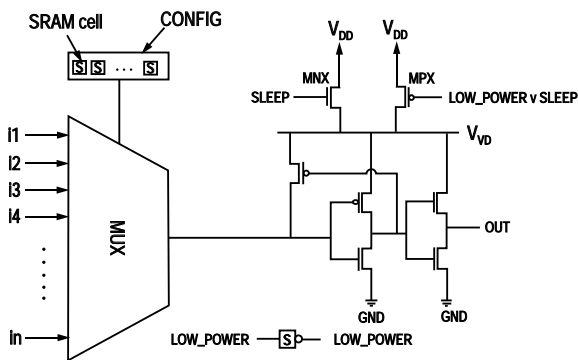

Fig. 14. Programmable low power routing switch (basic design) [39-42]
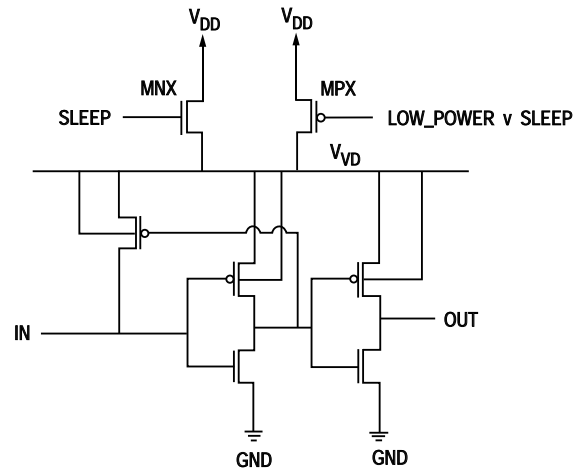

Fig. 15. Routing switch buffer alternate design [39-42]
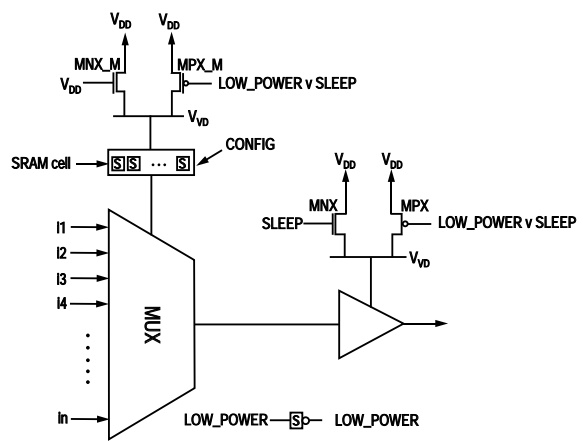
Fig. 16. Switch multiplexer with programmable mode [39-42]

FPGA glitch power analysis and reduction

Glitches in combinational CMOS logic are a well-known source of unnecessary power dissipation [43]. Reducing glitch power is a highly desirable target because in the vast majority of digital CMOS circuits, only one signal transition per clock cycle is functionally meaningful. Glitches are unwanted switching activities that occur before a signal settles to its intended value. On every active clock edge, glitches can occur within combinatorial logic. This is because every node has a different delay, which means combinatorial logic may change states several times before settling down. Glitches on a node are dependent on the logic depth to that node-the number of logic gates from the node

to the primary inputs (or sequential elements). Unstable logic expressions, unbalanced sets of paths driving a sensitive combinatorial cell, or a MUX select line that can toggle several times within a clock period are examples of sources of glitches. Some of the glitches are absorbed by the cell delays and do not propagate. Some others, however, do propagate and can affect the power dissipation. Some general glitch reduction techniques follow.

Frozen -Gates

The total number of glitches is reduced by replacing some existing gates with functionally equivalent ones (called F-Gates) that can be "frozen" by asserting a control signal. A frozen gate cannot propagate glitches to its output. Algorithms for gate selection and clustering that maximize the percentage of filtered glitches and reduce the overhead for generating the control signals are introduced by Luca Benini et al. [44]. A power-efficient CMOS implementation of F-Gates is also described. An important feature of the method is that it can be applied in place directly to layout level descriptions; therefore, it guarantees very predictable results and minimizes the impact of the transformation on circuit size and speed. The results are promising. In fact, an average glitch power reduction of 14.0%, yielding in an average reduction of the total power of 6.3% has been achieved at the cost of a negligible area increase (2.8% on average). Obviously, the speed of the circuits has not changed, since only noncritical gates have been replaced with F-Gates.

Adding programming delay elements

There was some more work done on circuit-level technique for reducing power in FPGAs by eliminating unnecessary logic transitions called glitches [45]. This is done by adding programmable delay elements to the logic blocks of the FPGA. After routing a circuit and performing static timing analysis, these delay elements are programmed to align the arrival times of the inputs of each LUT, thereby preventing new glitches from being generated. Moreover, the delay elements also behave as filters that eliminate other glitches generated by upstream logic or off-chip circuitry. On average, the proposed implementation eliminates 87% of the glitching, which reduces overall FPGA power by 17%. The added circuitry increases the overall FPGA area by 6% and critical-path delay by less than 1%. Furthermore, since it is applied after routing, the

technique requires little or no modifications to the routing architecture or computer-aided design (CAD) flow.

The technique is shown in Fig. 17; by delaying input, the output glitch can be eliminated. Note that the overall critical-path of the circuit is not increased since only the early-arriving inputs are delayed.
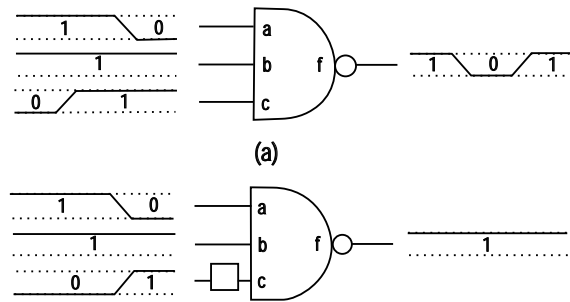

Fig. 17. Delaying early-arriving signal removes glitch [45]

Don't care base synthesis technique

FPGA glitch power analysis and reduction technique [46] is also with a don't-care-based synthesis for reducing glitch power in FPGAs. In the analysis of glitch power and don't-cares in a commercial FPGA it showing that glitch power comprises an average of 26.0% of total dynamic power. An algorithm for glitch reduction is presented by Warren shum et al. in which they takes advantage of don'tcares in the circuit by setting their values based on the circuit's simulated glitch behavior. Glitch power is reduced by up to 49.0%, with an average of 13.7%, while total dynamic power is reduced by up to 12.5%, with an average of 4.0%. The algorithm is applied after placement and routing, and has zero area and performance overhead.

Glitch power reduction at RTL level

The design for low-power techniques for RTL controller/data path circuits analyzes the generation and propagation of glitches in both the control and data path parts of the circuit [47]. In data-flow intensive designs, glitching power is primarily due to the chaining of arithmetic functional units. On the other hand, multiplexer networks and registers dominate the total circuit power consumption, and the control logic can generate a significant amount of glitches at its outputs, which in turn propagate through the data path to

account for a large portion of the glitching power in the entire circuit. The analysis for RTL power optimization with emphasis on glitch analysis and reduction also highlights the relationship between the propagation of glitches from control signals and the bit-level correlation between data signals. Based on the analysis the develop techniques that attempt to reduce glitching power consumption by minimizing propagation of glitches in the RTL circuit. The techniques include restructuring multiplexer networks (to enhance data correlations and eliminate glitchy control signals), clocking control signals, and inserting selective rising/falling delays, in order to kill the propagation of glitches from control as well as data signals. The results indicate that the glitch-reduction transformations can significantly reduce power consumption of RTL designs upto 30.06% on the average.

Clock gating power reduction technique

This is the most widely used technique for power reduction. The principle is to stop the clock whenever the device is not in use. Clock gating can be applied to sub-blocks of the design as well as to the whole device. However, correctly stopping the clock is very important. Knowing that the gating logic adds a delay to the clock signal, the effects on setup and hold times must be analyzed. While using clock gating, on FPGAs in particular, the user should take care of the placement of gating logic to minimize delay in the clock network.

For reduction of dynamic power through clock gating approach first we discuss about the clock gating architectures for FPGA power reduction [48]. Clock gating is a power reduction technique that has been used successfully in the custom FPGA/ASIC domain. Clock and logic signal power are saved by temporarily disabling the clock signal on registers whose outputs do not affect circuit outputs. By considering and evaluating FPGA clock network architectures with built-in clock gating capability and describe a flexible placement algorithm that can operate with various gating granularities (various sizes of device regions containing clock loads that can be gated together). Results show that depending on the clock gating architecture and the fraction of time clock signals are enabled, clock power can be reduced by over 50%, and results suggest that a fine granularity gating architecture yields significant power benefits. The architectures are illustrated in Fig. 18. Fig. 18(a) shows

the REGION architecture where enables are present on switches entering a region. Fig. 18(b) shows the more flexible COLUMN architecture where enables are also present on switches driving vertical spines in logic block columns. Thus, consider a broad range of clock gating architectures with various levels of granularity within clock distribution frameworks that resemble those in commercial chips.
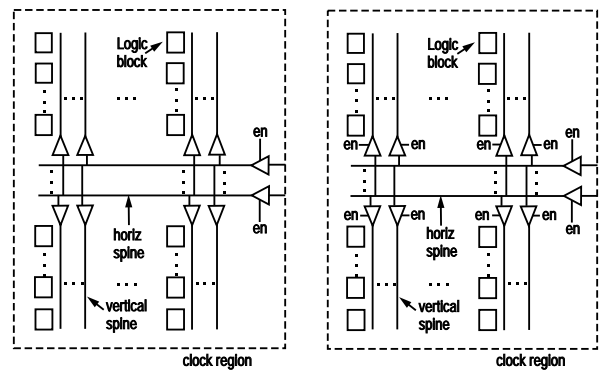

Fig. 18. Clock gating architectures; (a) Enables are present on switches entering a region (b) enables are also present on switches driving vertical spines in logic block columns [48]

Power reduction by guarded evaluation

This technique is used to stop inputs switching to propagate through blocks whose outputs are not used. For example, consider a multiplier whose outputs are used only under certain conditions. In this case, input to a multiplier can be stopped from toggling whenever outputs are not used (Fig.19) Adding a latch with the same "Enable condition will stop unnecessary switching from entering into the multiplier. Thereby the area overhead is really minimal and the saving of the swithcing propagation and thus the power dissipation.
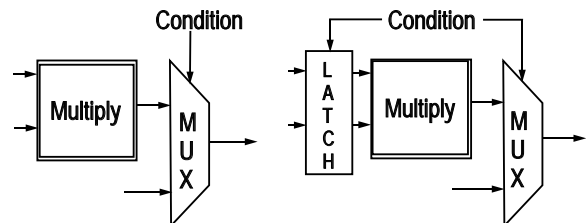

Fig. 19. Principle of guarded evaluation [49]

Guarded evaluation is a power reduction technique that involves identifying sub-circuits (within a

larger circuit) whose inputs can be held constant (guarded) at specific times during circuit operation, thereby reducing switching activity and lowering dynamic power. The concept is rooted in the property that under certain conditions, some signals within digital designs are not "observable" at design outputs, making the circuitry that generates such signals a candidate for guarding. Guarded evaluation has been demonstrated successfully for ASICs; here [49] they applied the technique to FPGAs. In ASICs, guarded evaluation entails adding additional hardware to the design, increasing silicon area and cost. Here the technique in a way that imposes minimal area overhead by leveraging existing unused circuitry within the FPGA. The primary challenge in guarded evaluation is in determining the specific conditions under which a subcircuit's inputs can be held constant without impacting the larger circuit's functional correctness. The solution to this problem based on discovering gating inputs using "noninverting" and "partial non-inverting" paths in a circuit's AND-inverter graph representation. Experimental results show that guarded evaluation can reduce switching activity on average by as much as 32% and 25% for 6-LUT and 4LUT architectures, respectively. Dynamic power consumption in the FPGA interconnect is reduced on average by as much as 24% and 22% for 6-LUT and for 4LUT architectures, respectively. The impact to critical path delay ranges from 1% to 43%, depending on the guarding scenario and the desired power/delay trade-off. Fig.20 (a) shows the guarded evaluation in which a multiplexer is shown receiving its inputs from a shifter and a subtraction unit, depending on the value of select signal Sel. Fig 20(b) shows the circuit after guarded evaluation. Guard logic, comprised of transparent latches, is inserted before the

functional units. The latches are transparent only when the output of the corresponding functional unit is selected by the multiplexer, i.e., depending on signal Sel.

Programmability of VDD

Power reduction is of growing importance for field programmable gate arrays (FPGAs). This reduction technique is discussed by Fei Li et al., about programmable supply voltage (VDD) to reduce FPGA power. The FPGA logic fabrics using dual-VDD levels shows that field-programmable power supply is required to obtain a satisfactory power-versus-performance tradeoff. They [50] further design FPGA interconnects fabrics for fine grained VDD programmability with minimal increase of the number of configuration static-random-access-memory cells. With a simple yet practical computer-aided design flow to leverage the field-programmable dual-VDD logic and interconnect fabrics, its carry out a highly quantitative study using placed and routed benchmark circuits, and delay, power, and area models obtained from detailed circuit designs. Compared to single-VDD FPGAs with the VDD level suggested by the ITRS for 100nm technology, field-programmable dual-VDD FPGA reduce the total power by 47.61% and the energy-delay product by 27.36%.

V. POWER REDUCTION THROUGH NON CLASSICAL MOS DEVICES

FPGA power can be more optimized by using ultra low power devices such as Tunnel-FET, FinFET and other MuGFET [51-59]. These devices are very capable at working on ultra low voltages which would be very efficient for lowering power consumption in FPGA. In this section we are going to discuss about latest advanced devices which would be used to implement FPGA. Circuit design in FPGA will have to cope with enhanced leakage power and large process variability. Using Tunnel-FETs or carbon nanotube transistors instead of MOSFETs could drastically reduce the leakage power. Here we are going to discuss the effects of various advanced devices into digital integrated circuits. Then further we can study how we can use those circuits in FPGA for low power consumption. IN this section we are going to discuss the characteristics and expected benefits of some emergingf device categories for ultra low power integrated circuits. First, we focus on two categories of
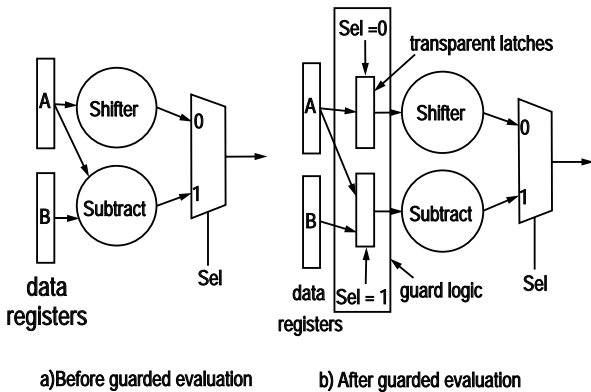

a)Before guarded evaluation   b) After guarded evaluation

Fig. 20. Guarded evaluation [49]

sub thermal subthreshold swing switches Tunnel FEts/ and carbon nano tube wires.

A. Tunnel FET

Tunnel FETs (TFETs) have emerged as mot promising candidates for ultra low power digital ICs with voltage supply lower than 0.5V [60]. In contrast to MOSFETs where charge carriers are thermally injected over a barrier, the carrier injection mechanism in a TFET is quantum mechanical band-t0-band-tunneling (BTBT). This mechanism is illustrated inthe band diagrams of Fig.21 corresponding to the ON state of the device. The main challenges of TFETs are their low Ion current and extending the low swing, S, over many decades of current.
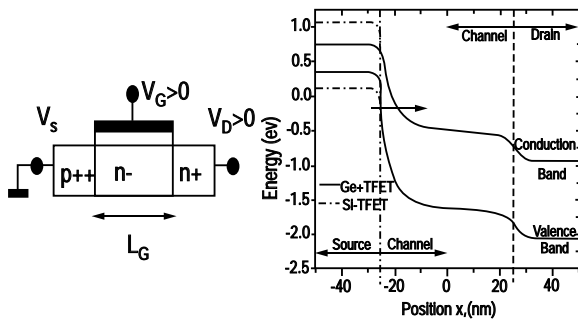

Fig. 21. Left: Cross section of an n-type TFET, Right: Simulated band diagram of n-type TFET with a Ge source compared to as Si source (at VDS=1V, VGS=0.5V, the cut is at y=1nm from the Si/Oxide interface) [60]

Leakage power dissipation is a fundamental problem for nano-electronic circuits special in case of SRAM memory. Scaling the supply voltage reduces the energy needed for switching, but the Field Effect Transistors (FETs) in today's integrated circuits require at least 60 mV of gate voltage to increase the current by one order of magnitude at room temperature. Tunnel FETs avoid this limit by using quantum-mechanical band-to-band tunneling, rather than thermal injection, to inject charge carriers into the device channel. Tunnel FETs based on ultrathin semiconducting films or nano-wire could achieve a 100-fold power reduction over CMOS transistors, so integrating tunnel FETs with CMOS technology could improve low-power integrated circuit. Recently, many novel devices such as the nano-wire gate all around (GAA) MOSFETs [61-62], Fin-shaped FET [63], carbon nanotube FET,

impact-ionization MOSFETs (I-MOSFETs) [64], and TFETs [65] have been demonstrated to minimize short channel effects and to lower the source-drain leakage current as compared to bulk MOSFEts, Leakage reduction using sleep sub-threshold transistor allows us to operate at very low threshold voltage with ultra low leakage and low supply voltages (VDD). ONIy TFET and I-MOSFET promise subthreshold slope less than 60 mV/dec and improved short-channel performances. INter-band Tunnel transistor also called as Tunnel Field Effect Transistor (TFETs) works on principle of inter-band tunneling [66], TFETs have shown to be extremely power efficient in [67] for logic circuit applications.

B. Circuit design with tunneling FETs

Since the TFET technology is compatible with CMOS, circuits containing both standard MOSFETs and TFETs can be produced. This allow utilizationof TFETs for special purposes, even if full replacement of CMOS cannot be achieved. This is of special interest in novel SOL technologies, which can no longer make use of bipolar devices as in bulk CMOS, and could instead make use of TFETs. In [68-69] they presented several logic gate structures and an SRAM cell containing a mixture of bulk MOSFETs and planar TFETS. In Fig.22 we see a six transistor SRAM cell where the two NMOS transistors with the source connected to GND have been replaced by TFETs [70]. Note that in planar bulk TFETs the two word line transistors cannot be replaced, since their source potential is different from GND and would require an additional well. On the other hand, the substrate contact inherent inthe TFET source region offers an area advantage in bulk technologies. We determined the butterfly curve by simulations based
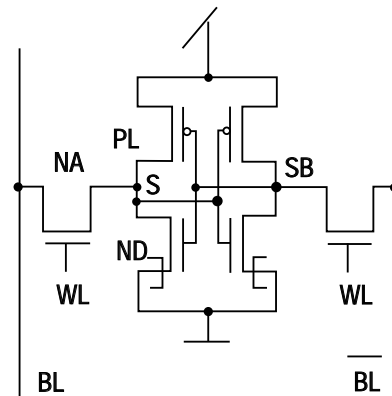

Fig. 22. Static memory cell with two n-MOS, two p-MOS and two TFET transistors [68-70]

on measured I-V characteristics to characterize the signal to noise margin of the cell, and it was found that the inclusion of tunnel FETs slightly enhances the noise margin of the cell, as it is shown in Fig. 23.
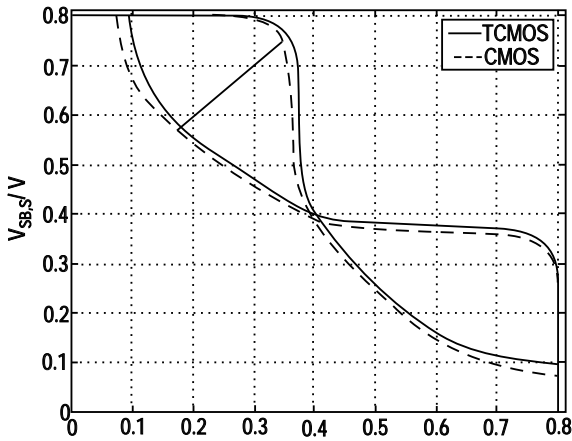

Fig. 23 Comparison of static noise margin of SRAM cells with conventinal CMOS transistors and TFETs as pull down devices [68-70]

C. Leakage Power reductions from Tunnel FET

Due to the inherent nature of TFETs, the OFF state leakage curent of a TFET is orders of magnitude lower than CMOS. Thus, we see a huge improvement in terms of leakage reduction [71]. Fig 24 shows the stand by leakage/cell of various SRAM designs. Both 6T and 7T
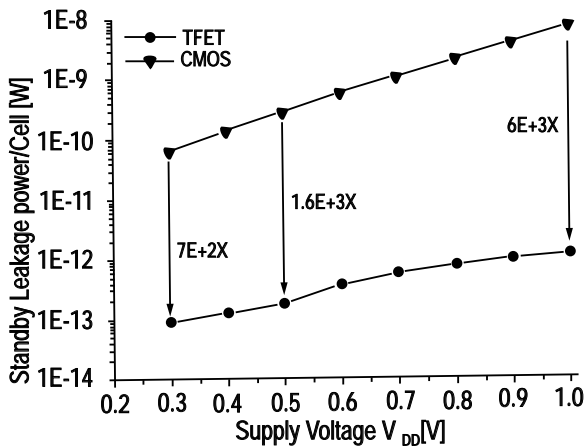

Fig. 24. Standby leakage/Cell for CMOS and TFET SRAM designs [71]

TFET has equal leakage powre due to the presence of the same leakage paths. It obtain a 700X and 1600X improvement in leakage reduction over CMOS designs at 0.3V and 0.5V V_{DD}. This shows that TFETs are a potential replacement candidate for CMOS transistors at low votlage and low power applicaitons.

D.. Carbon nano-tube wires

A somewhat larger step away from the CMOS mainstream is given by the Carbon Nanotube Field Effect Transistor (CNTFET). In [72] both n-channel and p-channel CNTFETs have been produced using different gate metals and Schottky Barrier source–drain regions. Inverters, ring oscillators, and simple logic gates have been fabricated already [73], but switching speed and ON-current seem still a large step below what is achievable with state-of-the-art CMOS devices [74-75]. However, since the intrinsic capacities are very low, and the device scalability is rather good, CNTFETs could nevertheless develop a high potential in the future when architectures with low interconnect capacitance can be developed.

VI. DISCUSSION AND COMPARISON

In the paper we discussed several techniques for static (leakage) power reduction and dynamic power reduction in modern nanometer FPGAs. Low power consumption is now the dominant requirements in designing next-generation high-end applications. Designing next-generation FPGAs to address the current trend of higher bandwidth and lower power is becoming much more challenging. We discussed that there are many factors which we must carefully considered when planning a new FPGA family to ensure that the new devices can address the power and performance requirements of the targeted applications in various market segments. These factors include selecting the right process technology, designing the right architecture, applying the right software power optimization, and enabling easier and power-efficient system-level design.

In the paper first we discussed about the architecture of FPGA, then we analyze the power consumption sources and the factors in which the static and dynamic power depends, while analyzing the power we come to know that the power can be controlled in various levels, as a device engineer we can reduce the subthreshold leakage current in design or in fabrication of device by taking some new design

Table 4 Static power consumption comparison of various techniques

S. No.	Static power reduction technique	Technology parameters	Reduction in static power (%)
1	Dual threshold transistor stacking [23]	90 nm technology	22.09
2	Selection of logic cells in FPGA [24, 25]	90 nm	30
3	Dual VT approaches [26,27]	90 nm	64
4	Activity packing in FPGAs [28]	MTCMOS based 90 nm	15
5	Fine-grained VDD [29]	0.35 %m/90nm CMOS	86/95
6	Dual-threshold FPGA routing design [30]	90 nm CMOS	28.83
7	Input vector reordering [31]	90 nm	50.3
8	FPGA Routing Multiplexers [32]	22 nm	20% less reduction as in 22 nm
9	CLB-Clustering [33]	90 nm	50
10	Off path leakage Power aware routing [34]	90 nm	16.79
11	Power gating [35]	45 nm	40

Table 5 Dynamic power consumption comparison of various techniques

S. No.	Dynamic power reduction technique	Technology parameters	Reduction in dynamic power (%)
1	High-level macro-modeling [36]	90 nm CMOS	7
2	LOPASS [38]	90 nm CMOS	61.6
3	Programmable FPGA routing circuitry [39-42]	90 nm	28-31
4	Glitch power reduction in combinational logic [43-46]	90 nm	12.5
5	Glitch power reduction at RTL level [47]	90 nm	30.06
6	Clock gating [48]	90 nm	50
7	Guarded evaluation [49]	45nm as 6 input LUT, 90 nm 4 –input LUT	32% for 6-input LUT, 25% for 4input LUT
8	Programmability of VDD [50]	90 nm	47.61

techniques and power reduction techniques at device level. While in modelling the device parameters we have to take consideration of leakage current and we have to introduce various techniques which can overcome the leakage current at the time of device miniaturization. The work of power reduction is also done on circuit level, we have discussed some techniques in this paper and we discussed about CAD tools and RTL level power reduction techniques. Based on the results we examined the best power reduction technique which we can implement on our latest FPGA devices. If we introduce these techniques in FPGA latest devices like Virtex 7 series of Xilinx and other FPGA devices and more upcoming devices than we

can make a Digital design which is less power consumed and more optimized. As we have reviewed some papers of power reduction techniques and across them we have find some reduction techniques which is really helpful for the power reduction, in case of static power reduction techniques shown in Table 4 the dual V T and fine-grained VDD power reduction percentage is high as 64% of power is reduces using dual VT and 95 % of power is reduced using Fine grain VDD techniques. In Table 5 the dynamic power reduction techniques results are shown in which LOPASS technique reduces 61.6% of dynamic or switching power consumptions. Glitch reduction and clock gating also reduces the power consumption by the factor of 30% and 50% respectively. We can reduce the dynamic power using the programmability in VDD by 47%. So if we use static and dynamic power reduction techniques we can introduced an ultra low power FPGA which can offered less power consumption with little area overhead. Table 4 shows the various comparisons between the static power reduction techniques. While table 5 shows the comparisons of various dynamic power reduction techniques.

VII. CONCLUSION AND FUTURE WORK

Lower system power consumption is a requirement for more applications than ever due to the dramatic increase in power conscious applications and tighter power budgets. Today, FPGA technology is increasing in usage in low-power applications, which makes achieving lower system power an increasingly important challenge. FPGAs have been adopted widely in recent years due to advanced technology that lowered the unit price, but the price reductions have come at the cost of higher power due to higher transistor leakage.

The various FPGA technologies have significantly different power profiles, and these differences can have a profound impact on the overall system design and power budget. Power consumption in FPGAs has become a primary factor for FPGA selection as we previously more focused on speed and making device more compact. But due to regressive use of mobile and portable devices human beings indirectly consume power from the nature because we are utilizing the power in the devices through natural resources like wind, water and other natural recourses. According to Moore's law the device size reduces to twice its present size in every 1.5 years, due to lowering the

device size we can make the system more faster and compact, but as all of us know that if we are using more and more high end devices it requires lots of battery and natural powers to run, so in the other hand we need to focus on Power savings, we have to develop more refined and more optimize devices which can work on low power. This paper focused on two aspects of power reduction one is leakage power other one is dynamic power. In the case of leakage power we discussed about critical design challenge for nano-meter FPGAs. With the emergence of highly complex FPGAs, capable of implementing complete systems, it has become imperative to develop innovative solutions for leakage power mitigation in FPGAs. This paper defines some leakage power models for accurately computing the leakage power. Subthreshold and gate leakage were discussed since they are the major sources of leakage power in the current generation technologies. In this paper we explore a strong review of various power reduction techniques and finds out which is the best technique currently for static and dynamic power. The techniques used for static power reduction reduces power upto 60-90% and in dynamic power reduction techniques it reduces power upto 30-50%. In static power reduction dual-VT FPGA architecture is explored and it indicates an average leakage power savings of upto 64%. In case of dynamic power, it can be reduced upto 61.6% using LOPASS Technique and upto 30% and 50 % with clock gating and Glitch reduction techniques respectively.

The work till now on reduction of power is quite impressive but it is not upto mark if we use high end application and if we lower the device size than our prime focus must be to reduce the power. As some current and future research work which is focused on high level design flows, multi-core architectures, advanced applications in network processing, signal processing, and embedded systems the power utilization is extreme. So the power can be reduce in steps from device level to system level, which starts from using low power devices like Fin-FET, double gate, Tunneling FET devices first and then we can use the device later on over circuits and after imposing power reduction techniques on the circuit level we can use the low power circuit in our FPGA architecture, and then on FPGA architecture further we can impose the RTL level and CAD level power reduction techniques and able to reduce the overall power of the

system that we are going to design, So the future work on aspect of power reduction is done from device level to system level. Currently FPGA IC is fabricated using CMOS technology but the work is going on to fabrication of the FPGA via Fin-FET, Tunneling-FET MOS, and Multi-Gate FET devices. The devices would be developed which can work on low source power and also with low leakage current. The objective must be focused on power reduction with taking consideration of all power reduction techniques and the future challenges which can come across us while implementing the techniques on FPGA.

REFERENCES

[1] Xilinx (2012), Lowering power at 28 nm with Xilinx 7 Series FPGAs, White paper WP389 (v1.1.1).

[2] Actel (2011), Dynamic power reduction in Flash FPGAs, White paper.

[3] Xilinx (2011), Power methodology guide, White paper, UG786 (v13.1).

[4] Kuon and J. Rose (2006), Measuring the gap between FPGAs and ASICs, ACM/SIGDA, International symposium on Field Programmable Gate Arrays, 21-30.

[5] V. Betz., J. Rose, and A. Marquardt (1999), Architecture and CAD for deep-submicron FPGAs, Kluwer academic publishers.

[6] S.J.E. Wilton, J. Rose, and Z.G. Vranesic (1995), Architecture of centralized field-configurable memory, ACM/SIGDA International symposium on Field-Programmable Gate Arrays (FPGA), 97-103.

[7] S. Hong and S.S. Chin (2003), Reconfigurable embedded MAC core design for low-power coarse grain FPGA, IET Electronics Letters, Volume 39, Issue 7, 606-608.

[8] G. Lemieux and D. Lewis (2004), Design of interconnection networks for programmable logic, Springer (formerly Kluwer Academic Publishers).

[9] International Technology Roadmap for Semiconductors (2011).

[10] Ahmed, Elias and Jonathan Rose (2000), the effect of LUT and cluster size on deep-submicron FPGA performance and density, ACM Symposium on FPGAs, 3-12.

[11] Lewis, D., et al. (2005), The Stratix II logic and routing architecture, ACM symposium on FPGAs, 14-20.

[12] Altera (2006), White paper on, FPGA architecture, ver. 1.0.

[13] V. Betz, J. Rose and A. Marquardt (1999), Architecture and CAD for deep-submicron FPGAs, Kluwer Academic Publishers.

[14] B. Zahiri (2003), Structured ASICs: Opportunities and challenges, International conference on computer design, 404-409.

[15] R. R. Taylor and H. Schmit (2004), creating a Power-aware structured ASIC, International symposium on low power electronics and design, 74-77.

[16] .K. J. Han, et al. (2007), Flash-based Field Programmable Gate Array Technology with deep trench isolation, IEEE Custom integrated circuits conference, 89-91.

[17] S. D. Brown (1994), an overview of Technology, Architecture and CAD tools for programmable logic devices, IEEE Custom integrated circuits conference, 69-76.

[18] J. Greene, E. Hamdy, and S. Beal (1993), Antifuse Field Programmable Gate Arrays, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 81, Issue 7, 1042-1056.

[19] Altera 2012, White paper on reducing power consumption and increasing bandwidth on 28-nm FPGAs, WP-01148-2.0.

[20] M. Anis and M. Elmasry (2003), Multi-Threshold CMOS digital circuits: Managing leakage power, Kluwer Academic Publishers (2003).

[21] G. Yeap (1998), Practical low power digital VLSI design, Kluwer Academic Publishers, Boston.

[22] A. Chandrakasan, S. Sheng, and R. Brodersen (1992), Low-Power CMOS digital design, IEEE Journal of Solid State Circuits, Volume 27, Issue 4, 473-484.

[23] R. Udaiyakumar and K. Sankaranarayanan (2012), Dual Threshold Transistor Stacking (DTTS) -A novel technique for static power reduction in nano-scale CMOS circuits, European Journal of Scientific Research, Volume 72 Issue 2, 184-194.

[24] Jason H. Anderson and Farid N. Najm (2004), Active leakage power optimization for FPGAs, IEEE Conference FPGA'04, 22-24.

[25] Jason H. Anderson and Farid N. Najm (2006), Active leakage power optimization for FPGAs, IEEE Transactions on computer aided design of integrated circuits and systems, Volume 25, Issue 3, 423 -437.

[26] Andrea Lodi, et al. (2005), Low leakage design of LUT-based FPGAs, Proceedings of ESSCIRC, Grenoble, France.

[27] Navid Azizi and Farid N. Najm (2005), Look-up table leakage reduction for FPGAs, IEEE Custom integrated circuits conference.

[28] Hassan, et al. (2005), Activity packing in FPGAs for leakage power reduction, IEEE Proceedings Design, Automation and Test in Europe.

[29] Canh Q. Tran, et al. (2005), 95% Leakage reduced FPGA using zigzag power-gating, Dual-VTH/VDD and Micro VDD Hopping, IEEE Asian Solid-State Circuits Conference.

[30] Rodrigo Jaramillo-Ramirez and Mohab Anis (2007). A dual-threshold FPGA routing design for sub-threshold leakage reduction, IEEE international symposium on circuits and systems.

[31] Hassan and Mohab Anis (2008). Input vector reordering for leakage power reduction in FPGAs, IEEE Transactions on Computer aided design of integrated circuits and systems, Volume 27 Issue 9, 1555 -1564.

[32] Mohd. Hasan, A.K. Kureshi (2009). Leakage reduction in FPGA routing multiplexers, IEEE international symposium on circuits and systems (2009).

[33] Mohammad Mehdi Tohidi and Nasser Masoumi (2010). FPGA leakage power reduction using CLB-clustering technique, IEEE Nanoelectronics Conference (INEC), 637 -638.

[34] Keheng Huang et al. (2012). Off-path leakage power aware routing for SRAM-based FPGAs, IEEE conference on design, automation & test in Europe conference & exhibition.

[35] Assem A. M. Bsoul and Steven J. E. Wilton (2010). An FPGA architecture supporting dynamically controlled power gating, IEEE international conference on Field-Programmable Technology (FPT).

[36] Li Shang, et al. (2002). Dynamic power consumption in Virtex™-II FPGA family, proceedings FPGA '02, 157-164.

[37] Anand Raghunathan, et al., High-level macro-modeling and estimation techniques for switching activity and power consumption, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 11, Issue 4, 538 -557.

[38] Deming Chen, et al. (2010). LOPASS: A Low-Power architectural synthesis systems for FPGAs with interconnect estimation and optimization, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 18, Issue 4, 564 -577.

[39] Jason H. Anderson, et al. (2009). Low-Power programmable FPGA routing circuitry, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 17, Issue 8, 1048 -1060.

[40] A. Rahman and V. Polavarapuv (2004). Evaluation of low-leakage design techniques for field-programmable gate arrays, in Proceedings ACM/SIGDA International Symposium on Field Programmable Gate Arrays, Monterey, 23–30.

[41] G. Lemieux (2003). Routing architecture for field-programmable gate arrays, Ph.D. dissertation, Dept. of Electrical & Computer Engg., Univ. Toronto, Canada.

[42] D. Lewis, et al. (2003). The stratix routing and logic architecture, in Proceedings ACM/SIGDA international symposium on Field Programmable Gate Arrays (FPGA), Monterey, 12-20.

[43] L. Benini, M. Favalli, and B. Ricco (1994). Analysis of hazard contributions to power dissipation in CMOS ICs, IWLPD-94: ACM/IEEE international workshop on low power design, 27–32.

[44] Luca Benini, et al. (2000), Glitch power minimization by selective gate freezing, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 8, Issue 3. 287 -298.

[45] Julien Lamoureux, et al. (2008). Glitch-Less: Dynamic power minimization in FPGAs through edge alignment and glitch filtering, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume 16, Issue 11, 1521 -1534.

[46] Warren Shum, Jason H. Anderson (2011). FPGA glitch power analysis and reduction, International symposium on low power electronics and design (ISLPED).

[47] Anand Raghunathan, et al. (1999). Register transfer level power optimization with emphasis on glitch analysis and reduction, IEEE Transactions on computer-aided design of integrated circuits and systems, Volume 18 Issue 8, 1114 -1131.

[48] Safeen Huda, Muntasir Mallick, Jason H. Anderson (2009). Clock gating architectures for FPGA power reduction, IEEE International conference on field programmable logic and applications.

[49] Chirag Ravishankar, et al. (2012). FPGA power reduction by guarded evaluation considering logic architecture, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[50] Fei Li, et al. (2007). Field programmability of supply voltages for FPGA power reduction, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume 26 Issue 4, 752 – 764.

[51] PIDS Working Group, Results and Issues (2007). In: ITRS 2007 public conference.

[52] Thean Av-Y, Shi Z-H, Mathew L, Stephens T, Desjardin H, Parker C, et al. (2006), Performance and variability comparisons between multi-gate FETs and planar SOI transistors. In: IEDM.

[53] Kavalieros J, et al. (2006), Tri-gate transistor architecture with high k gate dielectrics metal gates

and strain engineering. In: VLSI technology symposium.

[54] Satish Kumara, et al. (2006), Self-consistent and efficient electro-thermal analysis for poly/metal gate fin-FETs. In: VLSI technology symposium.

[55] Inaba, et al. 2006, the prospective multi gate device for future SoC applications. In: ESSDERC.

[56] Park Jong-Man, Han Sang-Yeon, Jeon Chang-Hoon, Sohn Si-Ok, Lee Jun-Bum,Yamada Satoru, et al. (2006). Fully integrated advanced bulk FinFETs architecture featuring partially-insulating technique for DRAM cell application of 40 nm generation and beyond. In: IEDM.

[57] Von Arnim K, et al. (2007), a low-power multi-gate FET CMOS technology with 13.9 ps inverter delay. In: VLSI technology symposium.

[58] Fulde M, Arnim K. V., Pacha C, Bauer F, Russ C, Sipra D., et al. (2007). Advances in multigate MOSFET circuit design. In: ICECS.

[59] Collaert, et al. (2006), Performance enhancement of MUGFET devices using Super Critical Strained–SOI (SC-SSOI) and CESL. VLSI Technology Symposium.

[60] Adrian M. Ionescu, et al. (2011). Ultra low power: emerging devices and their benefits for Integrated Circuits. Electron Devices Meeting (IEDM), IEEE International, 16.1.1 -16.1.4.

[61] W. M. Reddick and G. A. J. Amaratunga (1995). Silicon surface tunnel transistor, Applied Physics Letters, Volume 67 Issue 4, 494-496.

[62] D. Kim, Y. Lee, J. Cai, I. Lauer, L. Chang, S. J. Koester, D. Sylvester, and D. Blaauw (2009). Low power circuit design based on hetero-junction tunneling transistors, in ISLPED '09: Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design. New York, USA: ACM, 219–224.

[63] P. Nilsson (2006). Arithmetic reduction of the static power consumption in nanoscale CMOS, IEEE International Conference on Electronics, Circuits and Systems, 656-659.

[64] B. Van Zeghbroeck (2007). Principles of Semiconductor Devices, online at http://ecee.colorado.edu/~bart/book/.

[65] D. Kim, Y. Lee, J. Cai, I. Lauer, L. Chang, S. J. Koester, D. Sylvester, and D. Blaauw (2010). Low power circuit design based on heterojunction tunneling transistors, in ISLPED '09: Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design. New York, 219–224.

[66] B. Yang, K. D. Buddharaju, S. H. G. Teo, N. Singh, G. Q. Lo, and D. L. Kwong (2008). Vertical silicon-nanowire formation and gate-all-around MOSFET, IEEE Electron Device Letter, Volume 29, Issue 7, 791–794.

[67] B. Yu, L. Chang, S. Ahmed, H. Wang, S. Bell, C. Y. Yang, C. Tabery, C. Ho, Q. Xiang, T. J. King, J. Bokor, C. Hu, M. R. Lin, And D. Kyser (2002), FinFET scaling to 10 nm gate length, in Electron Devices Meeting, IEDM '02, 251–254.

[68] Nirschl T, et al. (2007), The Tunneling Field Effect Transistor (TFET) as an add-on for ultra-low-voltage analog and digital processes, IEDM.

[69] Nirschl T, Weis M, Fulde M, Schmitt-Landsiedel D. Revision of the tunneling field-effect transistor in standard CMOS technologies, IEEE Electron Device Letter Volume 28, Issue 4, 195-198.

[70] Ramakrishnan, K. Mookerjea, S. Datta, S.Vijaykrishnan, N.Pradhan D (2010), A novel Si-Tunnel FET based SRAM design for ultra low-power 0.3V VDD applications, Design Automation Conference (ASPDAC), 15th Asia and South Pacific, 181-186.

[71] Nirschl T. (2007). Circuit applications of the tunneling field effect transistor (TFET), Dissertation Technische Universität München.

[72] Chen Z et al. 2007. Gate work function engineering for nanotube-based circuits, ISSCC.

[73] Deng J, Patil N, Ryu K, Badmaev A, Zhou C, Mitra S et al. (2007), Carbon nanotube transistor circuits: circuit-level performance benchmarking and design options for living with imperfections, In: Proc ISSCC, 70-588.

[74] O'connor Ian, et al. (2007). CNTFET modeling and reconfigurable logic circuit design. IEEE Transaction Circuit System, Volume 54, Issue 11, 65–79.

[75] Pourfath M, Kosina H, Selberherr S. (2007). The role of inelastic electron–phonon interaction on the on-current and gate delay time of CNT-FETs, ESSDERC.