Designing of Low Power & High Performance VLSI Circuits

Rakesh Kumar
(Asst. Prof.)
Sobhasaria Engineering College
Gokulpura, Sikar (Raj.)

Nitesh Dixit
(Asst. Prof.)
BIET,Sikar
Bikaner Bypass, sikar (Raj.)

Kapil Kumawat
(Asst. Prof.)
Gyan Vihar University
Jagatpura, Jaipur (Raj.)

ABSTRACT
The demands of future computing, as well as the challenges of nanometer-era VLSI design, require new digital logic techniques and styles that are simultaneously high performance, energy efficient, and robust to noise and variation. We propose a new family of logic styles called Preset Skewed Static Logic (PSSL). PSSL bridges the gap between the two main logic styles, static CMOS logic and domino logic, occupying an intermediate region in the energy-delay-robustness space between the two. PSSL is better than domino in terms of energy and robustness, and is better than static CMOS in terms of delay. PSSL works by partially overlapping the execution of consecutive iterations through speculative evaluation. This is accomplished by presetting nodes at register boundaries before input arrival.

Keywords: VLSI, PSSL (Preset Skewed Static Logic**), CMOS**

1. INTRODUCTION
The relentless drive toward smaller, faster, and cheaper computing systems has, in large part, been enabled by exponential increases in device density and operating frequency through VLSI technology scaling. This, however, has led to exponential increases in power consumption that has reached the limits of reliability and cost effective cooling. In addition, the continued scaling into the nanometer regime has brought with it design robustness issues such as signal integrity, soft error, and environmental and process variability.

Furthermore, the issues of power consumption and robustness only get worse with time. This has created, therefore, a crisis in computer system design that threatens to be a stumbling block to future advancement. Designers of leading-edge computing systems, at any scale, are finding that power consumption and design robustness are also very important constraints, and must be taken into account at every level of design. At the circuit level, the choice of logic styles is important as it directly affects power, performance, and robustness. The two prevalent logic styles, static CMOS and domino logic, do not fully meet the needs of future computing. Static CMOS, though energy-efficient and robust, is too slow to be used in timing-critical designs. Domino logic, though fast, consumes too much power and is not robust. In addition, domino logic scales poorly so that its speed advantage is lessened while its power and robustness disadvantages are worsened. We therefore require new digital logic techniques and styles that are simultaneously high performance, energy efficient and robust to noise and variation. We propose a new family of logic styles called Preset Skewed Static Logic (PSSL). PSSL occupies an intermediate region in the energy-delay-robustness space between domino logic and static CMOS logic. PSSL is generally better than domino in terms of energy and robustness, and is generally better than static CMOS in terms of delay. PSSL works by partially overlapping the execution of consecutive iterations through speculative evaluation.
This is accomplished by presetting nodes at register boundaries before input arrival. This creates timing slack which can be traded for lower delay and/or lower energy. We also show a leakage reduction technique in PSSL that takes advantage of this slack to reduce energy-delay overhead.

2. BACKGROUND
Integrated circuit technology has advanced tremendously over the past 40 years, as predicted by Moore's Law [1]. Device counts have grown exponentially, from the 2300 transistors of the Intel 4004 processor in 1971, to the 592 million transistors of the Intel Itanium 2 processor in 2004. Simultaneously, clock frequencies have increased exponentially from 0.1MHz in the Intel 4004 to 3.8 GHz in currently shipping Intel Pentium 4's.
On the other hand, power consumption has been increasing at 20% per year and has reached power density limits. At the same time, noise, from many sources, as a fraction of power supply voltage, has increased while noise sensitivity has also increased. These factors, together with increased relative process variation and environmental variation, have made predictability and robustness difficult to achieve in new designs.

2.1 Power Consumption
Power has always been one of the foremost issues in system design. No matter what the design scale, there is a direct correspondence between power dissipation and performance/ functionality, battery life, cost, and size.
The almost ideal VTC of the CMOS inverter is not the main reason that high-complexity designs are implemented in static CMOS.
Rather, it's the almost zero power consumption in steady-state mode.
The reversed-bias diode current is, in general, very small.
Typical values are 0.1 to 0.5nA at room temperature.
For a device at 5V with 1 million devices, power consumption is 0.5mW. A more serious source is the subthreshold current.
The closer VT is to zero, the larger the leakage with VGS = 0V.

This establishes a firm lower bound on VT, which is > 0.5V today.

Chip power can be divided into two main components: dynamic switching and static leakage. Dynamic power dissipation, ignoring short-circuit current which is usually a small fraction of total dynamic power, is given by

$$P = \tfrac{1}{2}CV^2f$$

Where C is the average total on-chip capacitance switched per cycle**.**

The reduction of oxide thickness and threshold voltage has led to exponential increases in static leakage power. There are six leakage mechanisms in nanometer scale transistors [4], of which the three most significant are subthreshold leakage, gate leakage, and band-to-band tunneling (BTBT) leakage [5].

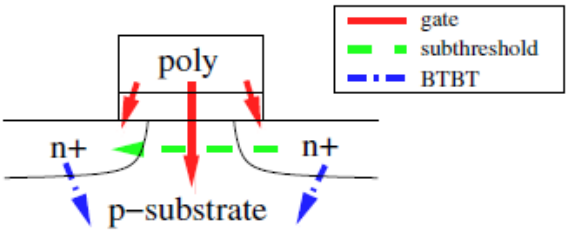


Figure 1: Major transistor leakage paths.

2.2 Robustness

Robustness is the measure of a design's tolerance to uncertainty. This uncertainty comes from various sources, most importantly from signal noise, single event phenomena (SEP), and variability.

Signal noise and signal integrity, within a chip, signals are not the nice 0's and 1's of the digital abstraction; real signals have noise. Dealing with this noise is the signal integrity challenge. Signal integrity problems manifest primarily in two ways. Firstly, they can directly cause state, such as dynamic nodes, latch nodes, and memory nodes, to be corrupted, causing incorrect computation. Secondly, they can add significant and unexpected delay.

Single Event Phenomena and soft errors, is one of the issues affecting the reliability of computing systems. Soft errors are the result of SEP (Single Event Phenomena), spatially and temporarily random events such as the collision and absorption of high-energy ionizing particles. An SEP manifests itself as a Single Event Upset (SEU), which is the flipping of a state node (RAM, latch, or dynamic node), or as a Single Event Transient (SET), a transient noise pulse that travels through logic and might be captured by a memory. Both SEU and SET can lead to soft error.

Improving Robustness, The conventional solution to improving robustness has been design margining, that is, designing for the worst case. This, however, has large energy-delay cost and becomes infeasible as relative uncertainty increases due to scaling.

3. PRESET SKEWED STATIC LOGIC

Preset Skewed Static Logic (PSSL) combines the energy efficiency and robustness of static CMOS logic with the performance of domino logic. We first show how Skewed Static Logic can improve performance in the presence of timing slack. We then show how to generate slack through preset. We then show the implementation of PSSL logic and PSSL pipelines..

3.1 Skewed Static Logic

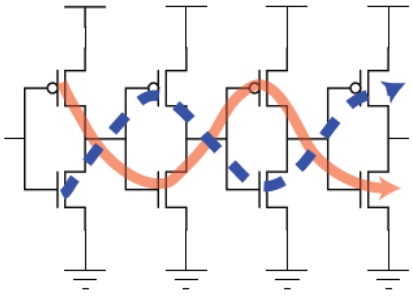


Figure 2: Inverter Chain

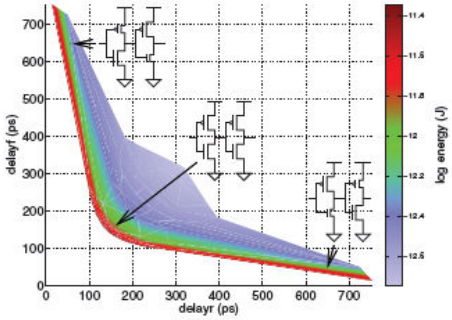


Figure 3: Energy-Delay.

Figure 2 shows a chain of four static CMOS inverters. The dashed curve indicates the transistor activation path, that is, the sequence of transistor chains that are turned on, for a rising input transition. The solid curve indicates the transistor activation path for a falling input transition. Note that the total path delay times of a rising input and that of a falling input are not necessarily the same. There is a trade-off between the two delay times and also between delay and energy; this is controlled by varying the sizes of individual transistors. For example, by increasing the size of transistors under the dashed curve, one can speed up the response of the circuit to a rising input transition. This comes at the cost of a slower response to falling input transitions and increased energy dissipation. Figure 3 shows this trade-off. The plot shows the energy and delay of two inverters within a long fan-out-of- 4 (FO4) chain. The X and Y axes represent delays through rising input and falling input paths. The shade at each x,y location indicates the required energy dissipation to achieve the delays. Note that the shade axis is logarithmic.

3.2 Preset

A simple PSSL circuit is shown in Figure 4. This resembles the chain of static inverters in Figure 2, except that the first inverter has been replaced by a NAND gate with one input tied to the clock. The logical function of this circuit is the same as the inverter chain. Let us assume that the input A is expected to arrive at the rising edge of the clock. The operation of this circuit as is follows. First, the falling edge of the clock initiates the process of preset. In preset, all circuit nodes are indirectly forced to pre-determined values.

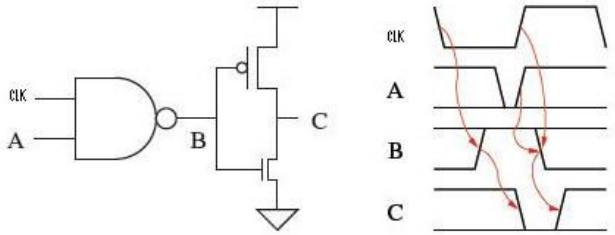


Figure 4: Preset Skewed Static Logic. Smaller transistors are less critical.

In particular, node B rises in turn causing node C to fall, thus completing the preset process. The idea behind the preset process is that we are speculatively computing all the nodes of the circuit presuming low input values. This begins one clock phase before the actual input value(s) arrive, so this computation has an extra clock phase to complete. The rising edge of the clock initiates the process of evaluate. Note that the process of evaluate is independent of the process of preset, and, in particular, evaluate can begin before preset completes. If the value of the input node, A is low at the rising edge of the clock and remains low, nothing further happens in the circuit and evaluate is complete. However, if the input node, A, is high when the clock rises or node A rises while the clock is high, then it causes node B to fall, in turn causing node C to rise, completing the evaluate process. Whether node A is high or low, eventually node C gets the correct value. However, we have decoupled the computation for low values of A (the preset process) from the computation for high values of A (the evaluate process), giving the former computation extra time and thus creating slack in the path of transistors in the preset process (i.e. the preset path). We can take advantage of this slack by reducing the size of transistors in the preset path to reduce power consumption, or by increasing the size of transistors in the evaluate path to reduce delay. Preset allows PSSL to outperform generic static CMOS logic. However, preset comes at the cost of extra power consumption because of spurious transitions from input mis-speculation and extra clocking overhead. A NAND gate was used to preset nodes high. A similar analysis holds if we use a NOR gate for preset. This time, preset is initiated by the rising edge of the clock, and node B of Figure 4 is preset low.

3.3 Pipelining

We now examine how to create pipelines using PSSL. We present PSSL using the three major clocking schemes: level-sensitive, edge-triggered, and pulsed.

3.3.1 Level-sensitive

Level-sensitive clocking uses alternating transparent latches as timing elements. A two phase Level-Sensitive PSSL (LS-PSSL) pipeline, shown in Figure 5, is the composition of PSSL pipeline stages of alternating phase, separated by transparent latches. One stage begins preset when adjacent stages begin evaluate.

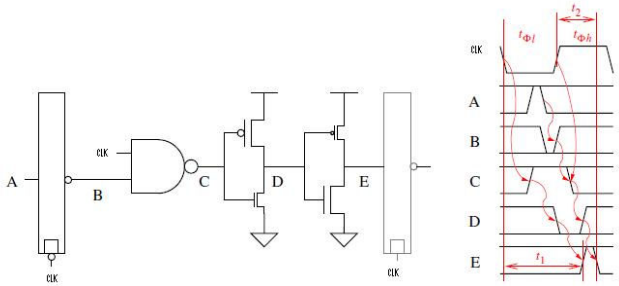


Figure 5: Two stage Level-Sensitive PSSL pipeline and timing diagram.

In LS-PSSL, the transparent latches serve two purposes. First, they hold pipeline state. Every legal (non-wave pipelined [7]) pipeline must have at least one latch in each full pipeline stage. Second, the latches prevent the preset wave-front from propagating to the following stage until after the preset phase. Otherwise, if the wave-front propagates early, it will cause inter-symbol interference as it becomes indistinguishable from the evaluate wave-front from the previous cycle. However, in contrast to their use in static CMOS pipelines, transparent latches are not used for synchronization (i.e. delay). Every legal pipeline must have a total of exactly one cycle of delay in each full pipeline stage. In LS-PSSL, the synchronization is performed by the NAND gates[7].

The operation of LS-PSSL, shown in Figure 5, is as follows. The falling edge of the clock begins preset, causing C to rise, D to fall, and finally E to rise. This path, whose delay is t1, must complete in one clock cycle, less setup delay. This coincides with the closing of the second latch at the falling edge of the clock. Therefore we derive the constraint

$$t_1 + t_s < t_{ch} + t_{cl} \quad (1)$$

where t_s is the setup time of the latch.

The rising edge of the clock begins evaluate. The value of A is effectively sampled by the first latch and NAND gate combination at the rising edge of the clock. If it is low, then

C falls, D rises, and, finally, E falls. This path, whose delay is t_2, must complete one setup delay before the closing edge of the latch. Therefore we derive the constraint

$$t_2 + t_s < t_{ch} \quad (2)$$

Similarly, the equations of the other half of the pipeline (not shown) are given by

$$t_3 + t_s < t_{cl} + t_{ch} \quad (3)$$

$$t_4 + t_s < t_{cl} \quad (4)$$

The preset path delays, t_1 and t_3 can be twice as long as the evaluate path delays, t_2 and t_4.

3.3.2 Edge-triggered

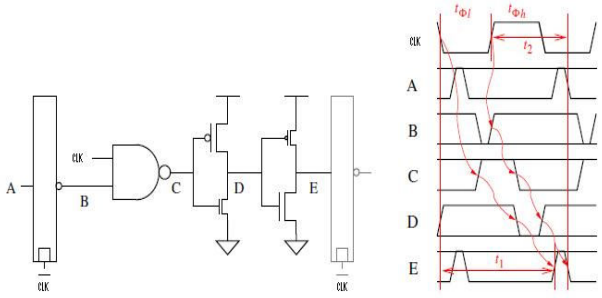


Figure 6: Edge Triggered PSSL and timing diagram.

As opposed to level-sensitive clocking, edge-triggered clocking uses a single monolithic timing element (usually a flip-flop). Figure 6 shows a pipeline using the same latch and NAND gate combination as before. However, this time there is only one set in a full pipeline stage, along with the diagram. The corresponding timing constraints are

$$t_1 + t_s < t_{ch} + 2t_{cl} \quad (5)$$

$$t_2 + t_s < t_{ch} + t_{cl} \quad (6)$$

$$t_1 > t_{cl} \quad (7)$$

The timing paths being and end on clock edges so that there is no time borrowing allowed. Note that there is a minimum path delay constraint on clock phase 2. Violating this constraint would cause inter-symbol interference. This is a fundamental race condition that cannot be avoided. It is impossible to have a data valid

window greater than a clock cycle. The constraint means that the clock can only be stopped on clock phase 1 (clock high). With a 50% duty cycle clock, a speedup of a factor of 1.5 can be achieved.

4. MANAGING LEAKAGE

As leakage is a major scaling concern, we present a survey of techniques to deal with leakage, highlighting special considerations for PSSL.

4.1 Leakage

The leakage paths in a static CMOS inverter chain are shown in Figure 7. There are various ways to reduce the leakage. A survey of leakage reduction techniques can be found in [4]. Four major techniques are multiple-threshold circuits, sleep vector, power gating, and body biasing.

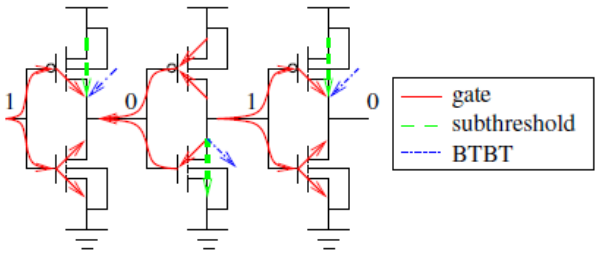


Figure 7: Static CMOS leakage paths.

4.1.1 Multiple-V_{th} circuits

To reduce leakage in active circuitry, one important technique is to use two (or more) types of transistors, each type with different Vth voltages. In the simplest scheme, the low-Vth transistors are reserved for gates in the critical paths; high-Vth transistors are used everywhere else [10]. In theory, this would cut leakage drastically without hurting performance or dynamic power. However, the problem is that few paths are truly non-critical since any timing slack can be traded for reduced dynamic power through transistor sizing and possibly voltage scaling. Therefore, a holistic optimization strategy that considers supply voltage, threshold voltage, and transistor sizing would be more effective [50]. The multiple- Vth technique adds a degree of freedom in circuit design, providing a way to trade-off static power vs. dynamic power vs. delay.

4.1.2 Sleep Vector technique

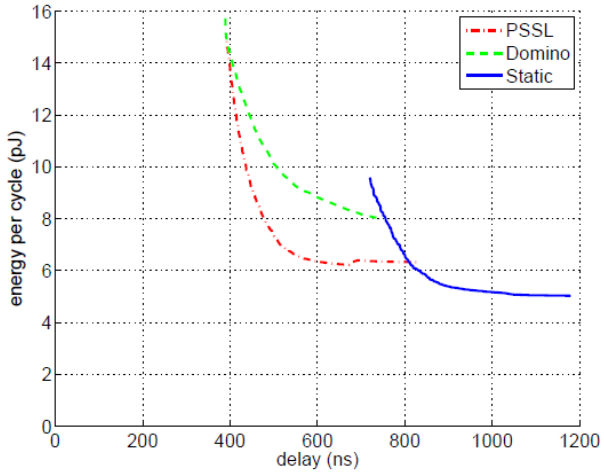
Sleep vector techniques take advantage of the fact that the amount of leakage in a circuit is data dependent. On sleep, a fixed input vector is applied to the primary inputs of a block. This input vector is designed to minimize the leakage in the block [11]. One can further reduce leakage, at the cost of energy and delay, by introducing internal control points that set specific nodes on sleep [12]. Sleep vector techniques, on their own, are highly dependent on circuit topology for their effectiveness and can rarely entirely suppress leakage.

4.1.3 Power Gating

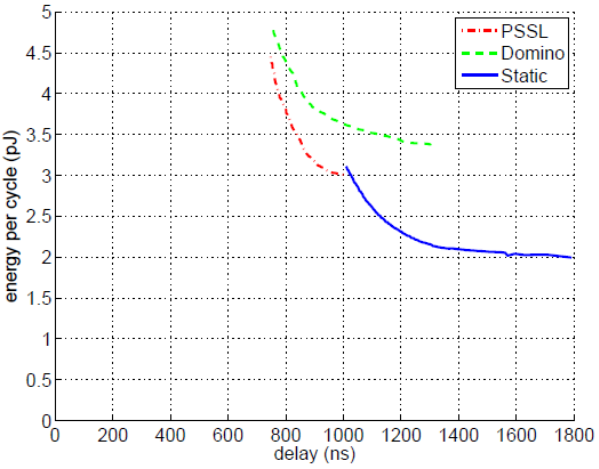
Another class of ideas uses sleep transistors to cut off power and/or ground from the logic. The original idea of power gating comes from Multi threshold voltage CMOS (MTCMOS) logic [14] MTCMOS cuts off the circuit from both rails using high- Vth transistors. MTCMOS was truly ahead of its time. It was proposed not as a way to deal with leakage (which was virtually non-existent in their 0.5 µm technology), but as a high-performance low-power circuit technique. Since they used transistors with lower than normal Vth and used ultra-low supply voltages, they ended up facing the same issues that designers would later face four or five technology generations later. There is also a variant of MTCMOS using only a single sleep transistor. The problem with MTCMOS is that its high-Vth transistors degrade performance.

5. RESULTS

We simulated the designs in the TSMC 0.18 µm micron process at 2.0V and 1.4V. The netlist was annotated to include estimated wire and source/drain parasitics. To measure critical path delay we used Pathmill, which is a transistor level static timing analysis tool. To measure energy consumption we used Nanosim to simulate the accumulator designs over 9 cycles with the same set of random input data. The supply energy drawn for the entire design was measured.



(a) 2.0V



(b) 1.4V

Figure 8: 32-bit accumulator comparison at 0.18 µm, TT, 25^0 C

The results are shown in Figure 8. At 2.0V LS-PSSL is superior to static CMOS except at the lowest energy points and is superior to domino except at the lowest delay points. At the 500ps design point, it uses 20% less energy than domino and is 33% faster than static CMOS. At 1.4V the results are qualitatively similar. However, the voltage scaling does not degrade the performance of static CMOS as severely so the crossover point between static CMOS and PSSL occurs at a higher point on the curve.

6. CONCLUSION

We have tested PSSL and compared it against other common logic styles, using several test circuits from shift registers to accumulators. We have shown that, over a wide range of design points, PSSL is superior to domino and static CMOS logic in terms on energy and delay, and at the same time is more robust

than domino. In particular, we have shown 20-30% energy reduction vs. domino and 33-50% delay reduction vs. static CMOS. In addition, we have presented the Double Pulsed Set Conditional-Reset Flip Flop (DPSCRFF), which is twice as fast as previous flip flops described in literature while consuming the same amount of energy.

7. ACKNOWLEDGMENTS

I am heartily thankful to my friends, Manish Verma and Ram Kumar Porwal, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject. Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the work.

8. REFERENCES

[1]. Gordon E. Moore. Cramming more components onto integrated circuits. Electronics, 38(8), April 1965.

[2]. Kaushik Roy, Saibal Mukhopadhyay, and Hamid Mahmoodi-Meimand. Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits. Proc. IEEE, 91(2):305–327, February 2003.

[3]. Saibal Mukhopadhyay, Arijit Raychowdhury, and Kaushik Roy. Accurate estimation of total leakage in nanometer-scale bulk CMOS circuits based on device geometry and doping profile. IEEE Trans. Computer-Aided Design of Integrated Circuits and Sytems, 24(3):363–381, March 2005.

[4]. A. Agarwal et al. Leakage power analysis and reduction: models, estimation and tools. IEE Proc. Comput. Digit. Tech., 152(3):353–368, May 2005.

[5]. Samuel K. H. Fung et al. 65nm CMOS high speed, general purpose and low power transistor technology for high volume foundry application. In Symp. VLSI Tech., pages 92–93, 2004.

[6]. Saibal Mukhopadhyay and Kaushik Roy. Modeling and estimation of total leakage current in nano-scaled CMOS devices considering the effect of parameter variation. In ISLPED, pages 172–175, August 2003.

[7]. Dan Ernst et al. Razor: A low-power pipeline based on circuit-level timing speculation. In MICRO, pages 7–18, 2003.

[8]. Muhammad E. S. Elrabaa, Mohab H. Anis, and Mohamed I. Elmasry. A contention-free domino logic for scaled-down CMOS technologies with ultra low threshold voltages. In ISCAS, volume 1, pages 748–751, 2000.

[9]. L. McMurchie, S. Kio, G. Yee, T. Thorp, and C. Sechen. Output Prediction Logic: a high-performance CMOS design technique. In Int. Conf. Computer Design, pages 247–254, 2000.

[10]. Liqiong Wei et al. Design and optimization of dual-threshold circuits for low-voltage low-power applications. IEEE Trans. VLSI Syst., 7(1):16–24, March 1999.

[11]. Yibin Ye, Shekhar Borkar, and Vivek De. A new technique for standby leakage reduction in high-performance circuits. In Symp. VLSI Circuits, pages 40–41, June 1998.

[12]. M. C. Johnson, D. Somasekhar, and K. Roy. Leakage control with efficient use of transistor stacks in single threshold CMOS. In DAC, pages 442–445, 1999.

[13]. Walid Elgharbawy et al. On gate leakage reduction in dynamic CMOS circuits. In Midwest Symp. Circ. Syst., pages 1390–1393, 2005.

[14]. Shin'ichiro Mutoh et al. 1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS. IEEE J. Solid-State Circuits, 30(8):847–854, August 1995.

[15]. Mindaugas Draˇzdˇziulis and Per Larsson-Edefors. Evaluation of power cut-off techniques in the presence of gate leakage. In ISCAS, volume 2, pages 745–748, 2004.

AUTHOR PROFILE

The first author is **Rakesh kumar.** He is currently working as Asst. Prof., in Sobhasaria Engineering College, Sikar; . He has obtained his M.Tech (VLSI Design) degree from C-DAC, Mohali and B.E.(ECE) from SEC,Dundlod (jhujhunu) . His area of interests is VLSI Design and Testing, Genetic Algorithms, BIST etc

The Second author is **Nitesh Dixit.** He is currently working as Asst. Prof., in BIET, Sikar; . He has obtained his M.Tech (Embedded System) degree from SRM University, Chennai and B.E.(ECE) from SEC,Dundlod (jhujhunu) . His area of interests is Cryptography, Genetic Algorithms etc

The Third author is **Kapil Kumawat.** He is currently working as Asst. Prof, in Gyan Vihar University, Jaipur; . He has obtained his M.Tech (VLSI Design) degree from C-DAC, Mohali and B.E.(ECE) from GIT, Jaipur . His area of interests is VLSI Design and Testing, Genetic Algorithms, BIST , SOC ,Embedded system etc.