Design of FPGA Architecture using Dynamically Controlled Power Gating with Multiple Sleep Mode

Dr. N. Kumareshan[1], M. Anandkumar[2], R. Divya[3]

[1]Associate Professor, [2, 3] PG Scholar

Department of Electronics and Communication Engineering

PPG Institute of Technology, Coimbatore, India

Abstract---Leakage power is an important component of the total power consumption in field-programmable gate arrays (FPGAs).Power gating was shown to be effective at reducing the leakage power. Previous techniques focus on turning OFF unused FPGA resources at configuration time, the benefit of this approach depends on resource utilization. In the earlier work an FPGA architecture that enables dynamically controlled power gating, in which FPGA resources can be selectively powered down at run-time is analyzed. The proposed work describes a power gating technique with multiple sleep modes where each mode represents a trade-off between wake-up overhead and leakage savings. It also over comes high wake-up latency and wake-up power penalty of traditional power gating to various applications for larger inactivity. This could lead to significant overall energy savings for applications having modules with long idle times. By using the proposed method power saving are achieved, compared with the existing technique the proposed technique gives better results for various circuits.

Keywords---FPGA-Wake up penalty-Multiple mode.

1. INTRODUCTION:

Field-programmable gate arrays (FPGAs) have become ubiquitous in applications, such as telecommunications, digital signal processing, and scientific computing. In the mobile devices market, however, FPGAs have had limited penetration, partially due to their high power consumption. Compared with application-specific integrated circuit (ASIC) implementations, FPGA implementations consume more power on average.

To bring reconfigurable technology to these hand-held applications, new programmable devices that consume significantly less power are required. Many researchers have proposed techniques for reducing the power dissipation of FPGAs based on the methods that have originally been applied to ASICs, including guarded evaluation, clock gating, and power gating. Previous techniques to reduce the power dissipation of FPGAs have focused on reducing both the dynamic and the static (leakage) power of these devices.

Dynamic power is dissipated due to charging and discharging of the circuit’s Capacitance, while leakage power is dissipated when the circuit is idle. Static power dissipation is a major component of the total power consumption in reconfigurable devices based on the CMOS technology nodes.

An effective way to reduce leakage power is to employ power gating by connecting the supply voltage or the ground of a circuit component through a power gating transistor, also called a sleep transistor or a power switch, the circuit component can be turned ON or OFF by turning the corresponding power switch ON or OFF. When the power switch is turned OFF, the leakage current is limited by that of the power switch. A performance loss may result because of the extra resistance in the current path. By sizing the power switch appropriately, an acceptable trade-off between the performance, power savings, and area can be found.

1.1 FIELD PROGRAMMABLE GATE ARRAYS:

FPGA provide the next generation in the programmable logic devices. The word Field in the name refers to the ability of the gate arrays to be programmed for a specific function by the user instead of by the manufacturer of the device. The word Array is used to indicate a series of columns and rows of gates that can be programmed by the end user.

As compared to standard gate arrays, the field programmable gate arrays are larger devices.

The basic cell structure for FPGA is somewhat complicated than the basic cell structure of standard gate array. The programmable logic blocks of FPGA are called Configurable Logic Block (CLB).

The FPGA architecture consists of three types of configurable elements-

(i) IOBs – a perimeter of input/output blocks

(ii) CLBs- a core array of configurable logic blocks

(iii) Resources for interconnection

The IOBs provide a programmable interface between the internal; array of logic blocks (CLBs) and the device's external package pins. CLBs perform user-specified logic functions, and the interconnect resources carry signals among the blocks.

A configurable program stored in internal static memory cells determines the logic functions and the interconnections.

The configurable data is loaded into the device during power-up reprogramming function. FPGA devices are customized by loading configuration data into internal memory cells.

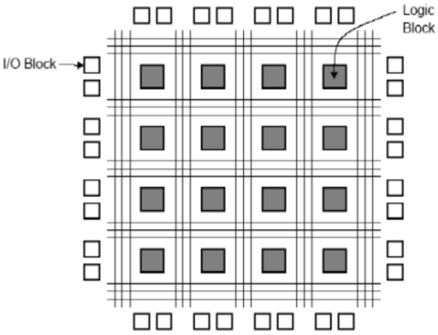


Fig 1.1 Architecture of FPGA

The general structure of FPGA chip. It consists of a large number of programmable logic blocks surrounded by programmable I/O block. The programmable logic blocks of FPGA are smaller and less capable than a PLD, but an FPGA chip contains a lot more logic blocks to make it more capable. As shown in fig.1 the logic blocks are distributed across the entire chip. These logic blocks can be interconnected with programmable inter connections.

Xilinx, Inc. inverted FPGAs, and in this section we will see the FPGA architecture used by Xilinx. The programmable logic blocks in the Xilinx family of FPGAs are called Configurable Logic Blocks (CLBs).The Xilinx architecture uses, CLBs, I/O blocks switch matrix and an external memory chip to realize a logic function. It uses external memory to store the interconnection information. Therefore, the device can be reprogrammed by simply changing the configuration data stored in the memory.

2. PROPOSED METHOD

Leakage power is an important component of the total power consumption in field-programmable gate arrays (FPGAs).Power gating was shown to be effective at reducing the leakage power. Previous techniques focus on turning OFF unused FPGA resources at configuration time; the benefit of this approach depends on resource utilization. In this paper, an FPGA architecture that enables dynamically controlled power gating, in which FPGA resources can be selectively powered down at run-time. This could lead to significant overall energy savings for applications having modules with long idle times.

To bring reconfigurable technology to these handheld applications, new programmable devices that consume significantly less power are required. Many researchers have proposed techniques for reducing the power dissipation of FPGAs using techniques that have originally been applied to ASICs, including guarded evaluation, clock gating, power gating, and dual supply voltages. Even after applying all these techniques, however, the power consumption of FPGAs remains prohibitive for some applications. Previous techniques to reduce the power dissipation of FPGAs have focused on reducing both the dynamic and static (leakage) power of these devices. Dynamic power is dissipated in a circuit when it performs activity, while leakage power is dissipated when the circuit is idle.

Thus, low-leakage FPGAs are essential if they are to be used for these kinds of applications. An effective way to reduce leakage power is to employ power gating. By connecting the supply voltage or the ground of a circuit component (an inverter in the figure) through a power gating transistor, also called a sleep transistor or a power switch, the circuit component can be turned on or off by turning the corresponding power switch on or off. When the power switch is turned off, the leakage current of the whole circuit is limited by that of the power switch, which significantly reduces the leakage power dissipated.

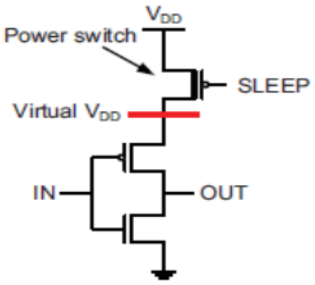


Fig 2.1 Basic power gating

Previous proposals for power gating in FPGAs use configuration bits to control the power switches. Refer to this as statically-controlled power gating, since once configured, the state of each part of the chip (on or off) does not change. Statically-controlled power gating is effective for FPGAs, since if the design does not fill an entire FPGA, the remainder of the FPGA can be safely turned off, saving leakage power. However, if only small amounts of the FPGA are not used, the savings from this technique may be limited.

In this proposed technique dynamically-controlled power gating in an FPGA. The power switches can be turned on and off at run-time under control of other circuitry either running on the FPGA itself, or external to the FPGA. The signals to control the power switches are connected to the general-purpose routing fabric of the FPGA. For many mobile applications, dynamically-controlled power gating is very compelling. It is conceivable that such applications contain blocks that only need to operate occasionally. Turning off blocks when they are not needed can significantly reduce the leakage power of the device.

Power state controller (PSC) from the data flow graph (DFG) of an application in order to exploit the idleness periods of the application to reduce the dissipated leakage energy. They assumed a power gating architecture similar to that proposed.

2.1 FINE GRAIN ARCHITECTURE:

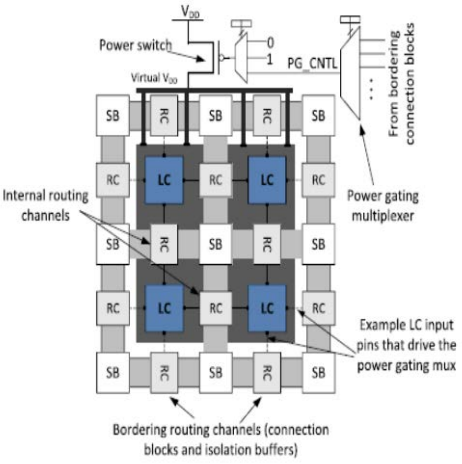

V_DD
Power switch
Virtual V_DD
0
1
PG_CNTL
From bordering connection blocks
Power gating multiplexer
Internal routing channels
SB
RC
LC
Example LC input pins that drive the power gating mux
Bordering routing channels (connection blocks and isolation buffers)

Fig 2.3 Fine grain architecture

The basic power gating architecture enables dynamically controlling the power state of individual logic clusters and the routing resources in their bordering routing channels (input pin connection boxes and track isolation buffers) without modifying the routing architecture; therefore, no changes to the routing algorithms are required. The switch boxes in our architecture are always assumed to be powered on in order to enable flexible routing of connections, including the power control signals. Nevertheless, believe that enabling dynamic control of the power state of switch boxes is crucial to enable complete dynamic control of the power state of a device

Basic power gating architecture, 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 of the LC. 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 labelled PG CNTL1, 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.

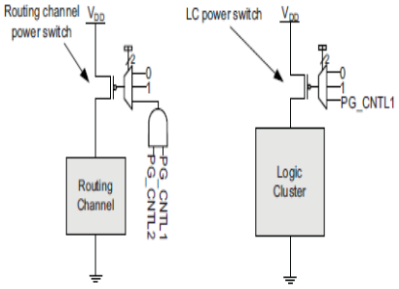


Fig 2.4 Power control signal

2.2 COARSE GRAIN ARCHITECTURE:

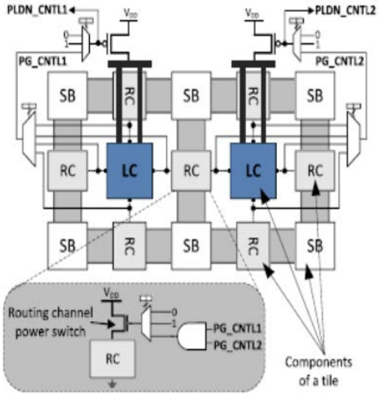


Fig 2.5 Coarse grain architecture

The area and power overheads associated with the architecture are due to the sleep transistors of the LC and the RCs, the power gating multiplexer of the LC, the 3:1 multiplexers that drive the gate of the sleep transistors, the AND gates required to implement a proper power gating for the RCs, and the additional SRAM configuration memory cells.

Typically, when an application is mapped to an FPGA, blocks that are part of the same functional module are placed close to each other in order to minimize delay and wiring costs. Thus, it is likely that a group of LCs and RCs that are spatially close to each other share the same power state. It is, therefore, feasible to support power gating at a coarser granularity level in order to reduce the area and power overheads of the power gating circuitry.

The concept of coarse-grained PGRs is presented here. Where each PGR is composed of only one tile, we propose a coarse-grained architecture, in which a PGR is composed of a number of tiles. Similar to the tile-level architecture the SRAM configuration memory cells and FFs are powered on all the time.

2.2.1 Coarse-grained power-gated switch blocks:

The proposed power gating architecture for LCs and RCs (track isolation buffers and CBs). This section focuses on describing the power gating circuitry for SBs in a PGR.

The power control signal that is used to control the power state of the LCs region (PG_CNTL) is also used to selectively control the power state of the individual SBs that belong to the same region. For each LC, the SB that belongs to the same region as the LC lies in the right-bottom corner of that LC.

Power gating circuitry for an SB. This circuitry is similar to that for the other components, as described in the previous sections. The gate input of the pull-down transistors is the same as the gate input to the power switch.

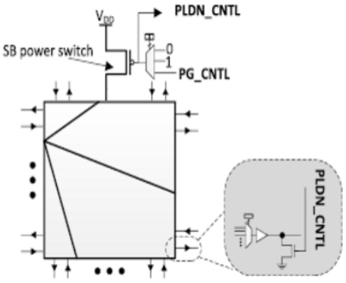


Fig 2.6 Power gating circuit for a SB

This scheme enables different power modes for different components in a PGR the supported power modes. For example, if the internal part of a PGR (LCs and internal RCs) is configured as DC, there is flexibility in configuring the power state for the individual SBs and bordering RCs. This flexibility allows some SBs to be always-ON to route important signals, such as power control signals or signals that connect between different modules.

2.3 MULTIPLE SLEEP MODES:

Multiple sleep modes can be easily obtained by applying a different bias to the footer in each mode. This circuit has four operating modes – Active, Sleep, Dream and Snore. In each mode, different gate bias is applied to the gate terminal of the footer device. A two-bit select signal is used to choose the desired operating mode.

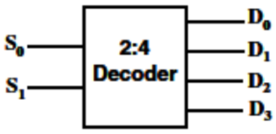


Fig 2.7 2X4 Decoder

S1S0	V_G (Footer)	Mode
00	0	Snore
01	V_1	Dream
10	V_2	Sleep
11	V_{DD}	Active

Table 2.1 Truth table

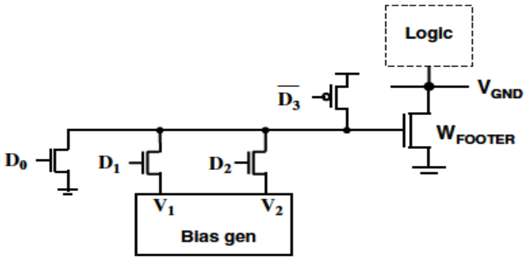


Fig 2.8 Multi-mode schematic diagram

Multiple sleep modes allow us to trade-off leakage savings with wake up latency and wake up energy overhead and hence can be very useful in wake-up constrained sleep operations. The values of two intermediate gate voltages (V1 and V2) were chosen to get two evenly spaced points in the wake-up penalty vs. leakage space. The sleep mode was state-retentive while the dream mode did not retain the state.. Here, leakage and wake-up overhead numbers are normalized with respect to the corresponding values in the snore mode. There is no wake-up penalty in the active mode. It is clear from the figure that the multiple sleep mode circuit provides a desired trade-off between leakage savings and wake-up overhead at all operating conditions and process corners.

2.4 BENCHMARK APPLICATIONS:

One of the primary drawbacks of power gating is the long wake up latency associated with discharging of the virtual ground rail during mode change from sleep to active. This latency can be up to 8 – 10 cycles. This makes power gating effective only when the data input is not switching for greater than the latency time plus the path propagation delay, generally greater than 8 cycles

The data switching rate and applied hard (conventional) power gating when the data was constant for 8 or more cycles. Next, we took the same data and applied multiple sleep modes. The optimum sleep mode for the processor was selected based on the number of idle cycles and the wakeup requirements as described in Table 2.2.

The snore mode is same as regular power gating and requires 8 cycles to wake up. Dream takes a little over half the time to wake up and so it was allotted 5 cycles. Finally, the sleep mode was assigned 3 cycles to fully wake up. The number of cycles in the wake-up latency set the constraint on the minimum number of idle clock cycles needed before a processor can enter in the corresponding sleep mode without any wakeup overhead. For latencies of less than 3 cycles, no sleep mode is applicable.

The percent leakage savings in various sleep modes as a function of number of idle cycles. As the number of idle cycles grows, the average leakage savings increase for all modes because the one-time wake-up power overhead gets offset by the leakage savings in each idle cycle. Divided in four regions based on the wake-up latencies shown in Table2.1.

In Region 1, no sleep mode can be used since the minimum wake-up latency (idle time) is 3 cycles.

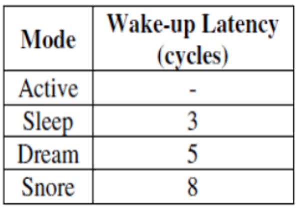
In Region 2, use sleep mode even though the dream mode provides more leakage savings. This is because the dream mode requires at least 5 idle cycles to wake up, and thus cannot be used in this Region.

In Region 3, snore cannot be used since it requires at least 8 idle cycles to wake up. In this region, we have a choice between sleep and dream, so we choose dream for the greater leakage savings.

In Region 4, snore mode provides the best leakage savings.

Regular power gating and multiple mode power gating to six different applications. On average, a reduction in leakage using multiple mode power gating as compared to conventional single mode gating. The leakage savings for various benchmark applications. These additional power savings are obtained at no superfluous wake-up delay penalty because the sleep modes are selected under wake-up constraints.

Table 2.2 Wake-up latency for various mode



Mode	Wake-up Latency (cycles)
Active	-
Sleep	3
Dream	5
Snore	8

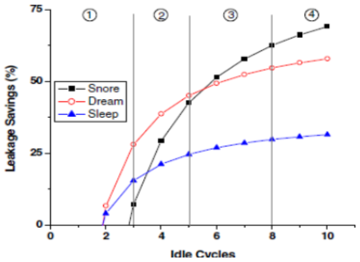


Fig 2.9 Leakage saving for various multiple mode.

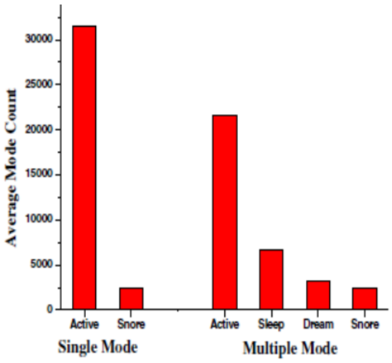


Fig 2.10 Average mode count

Also looked at the total mode count for both single and multiple mode power gating. single mode power gating was used an average of 2,414 times as compared to multiple mode power gating where a power savings mode (sleep, dream or snore) was selected an average of 12,393 times.

These results show that the multiple mode power gating allows a processor to save additional leakage by entering a leakage saving state more often than the conventional gating. Summarizes the average number of times a mode is utilized for both single mode and multi-mode power gating.

Multiple mode power gating allows a processor to enter a power gated mode more frequently than single mode power gating. In normal operation, the timeline has a value of 1. In full power gated mode, it has a value of 0. For the multiple mode power gating case, the timeline has a value in between 0 and 1, representing a partially power gated state.

Finally, some applications require the internal logic gates to retain state, even when there is no logical evaluation being performed. In such a case, single mode power gating cannot be used since that would flush out the data. However, our sleep mode is state retentive and can be used to retain state while simultaneously reducing leakage power consumption. , we compared the leakage power of no power gating to state-retentive power gating (only sleep mode) and found an average of reduction in leakage power while retaining states of the internal nodes.

3. EXPERIMENTIAL RESULT

3.1 COARSE GRAIN:

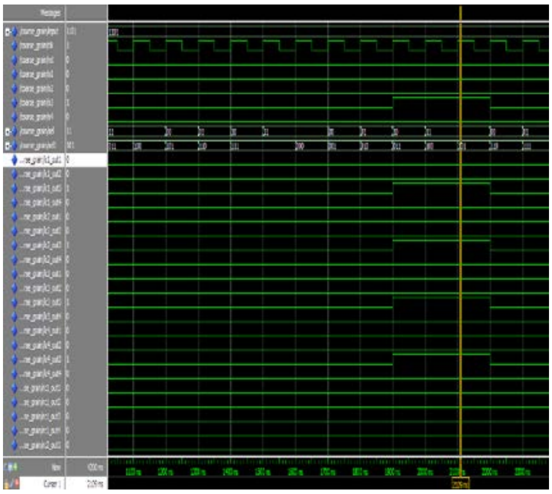


Fig 3.1 Output of Modelsim

The figure 3.1 shows that output of coarse grain by using Modelsim of proposed method. The proposed method produced a better output than the existing method comparison is given below,

3.1.1 OUTPUT BY USING XILINX:

Table 3.1

Existing method	Area	Delay	Power
Coarse grain	3763	14.03ns	356

TABLE 3.2

Proposed method	Area	Delay	Power
Coarse grain	2306	12.822ns	236

3.2 FINE GRAIN:

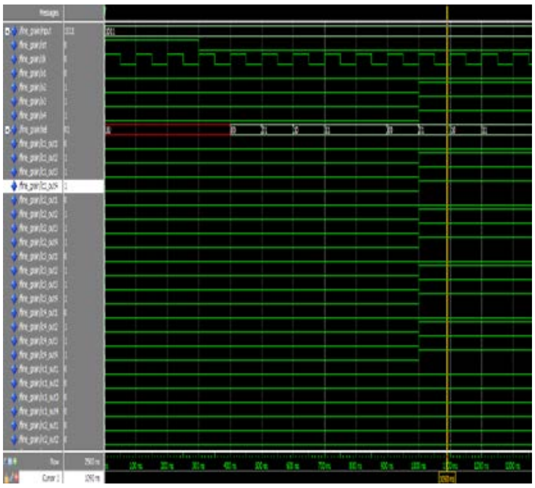


Fig 3.2 output of Modelsim

The figure 3.2 shows that output of fine grain by using Modelsim of proposed method. The proposed method produced a better output than the existing method comparison is given below,

3.2.1 OUTPUT BY USING XILINX:

Table 3.3

Existing method	Area	Delay	Power
Fine grain	2211	13.33ns	231

Table 3.4

Proposed method	Area	Delay	Power
Fine grain	2096	12.822ns	221

4. CONCLUSION AND FUTURE WORK

Leakage power is an important component of the total power consumption in FPGAs. Power gating has been proposed in the literature for FPGAs to enable turning off unused resources. This capability is desirable in applications that show long idle times, such as mobile applications, in order to exploit idleness periods to turn off idle modules, thus reducing their leakage power

In this paper, FPGA architecture that enables dynamic control for the power state of the logic clusters and the routing channels. It show that the leakage power reduction and the area overhead results are promising. A multiple mode power gating design technique was introduced for enhanced leakage reduction. The flexibility provided by the multiple sleep modes can be very useful in effective power management in power conscious designs.

In future the power consumption is reduced by using Autonomous fine grain gating in asynchronous FPGA alone. Standby state is used to wake up the LB before the data arrives and power OFF the LB only when the data does not come for quite a while. As a result, the wake-up time can be hidden and the dynamic power of unnecessary switching of the sleep transistor can be saved.

REFERENCES

[1] F. Li, Y. Lin, L. He, and J. Cong, “Low-Power FPGA using Pre-Defined Dual-Vdd/Dual-Vt Fabrics,” in Proceedings of the 12th ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, 2004, pp. 42–50.

[2] G. Lemieux, E. Lee, M. Tom, and A. Yu, “Directional and Single-Driver Wires in FPGA Interconnect,” in Proceedings of the IEEE International Conference on Field-Programmable Technology, 2004, pp. 41–48.

[3] H. Kawaguchi, K. Nose, and T. Sakura, “A super cut-off CMOS (SCCMOS) scheme for 0.5-V supply voltage with pico

ampere stand-by current," *JSSC*, vol. SC-35, pp. 1498– 1501, Oct. 2000.

[4] J. Kao, S. Narendra, and A. Chandrakasan, "MTCMOS hierarchical sizing based on mutual exclusive discharge patterns,"DAC, pp.495–500, June1998.

[5] J. Luu, I. Kuon, P. Jamieson, T. Campbell, A. Ye, W. M. Fang, and J. Rose, "VPR 5.0: FPGA CAD and Architecture Exploration.

[6] K. Bernstein, C. T. Chuang, R. Joshi and R. Puri, "Design and CAD challenges in sub-90 nm CMOS technologies," ICCAD, pp 129-136, Nov. 2003.

[7] Kuon and J. Rose, "Measuring the Gap between FPGAs and ASICs," in Proceedings of the 14th ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, 2015, pp. 21–30.

[8] M. Klein. (2009, April) Power Consumption at 40 and 45 nm. Xilinx, Inc. white paper WP298 (v1.0). Xilinx.

[9] Q. Wang, S. Gupta, and J. H. Anderson, "Clock Power Reduction for Virtex-5 FPGAs," in Proceeding of the 17th ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, 2012, pp. 13–22.

[10] R. P. Bharadwaj, R. Konar, P. T. Balsara, and D. Bhatia, "Exploiting Temporal Idleness to Reduce Leakage Power in Programmable Architectures," in Proceedings of the 2005 Asia and South Pacific Design Automation Conference, 2005, pp. 651–656.

[11] S. Mutoh et al., "1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS," JSSC, vol. SC-30, pp. 47–854, Aug. 1995.

Authors Biography

Dr. N. Kumareshan received his Bachelor's degree in Electronics and Communication Engineering from SNS College of Technology at Anna University in 2009. He received master degree in Communication Systems from Bannari Amman Institute of Technology at Anna University in 2011. He received Doctor of Philosophy in Wireless Networks from Anna University in 2016. He is currently an Associate Professor in PPG Institute of Technology. His main areas of research interest are Wireless Communications and Networking.

Mr. M.Anandkumar received his Diploma degree in Electronics & Communication Engineering from GRG polytechnic college at DOTE in 2011. He received his Bachelor's degree in Electronics & Communication Engineering from INFO institute of engineering at Anna University in 2014. At present he is pursuing M. E. with the specialization of VLSI Design in PPG institute of technology at Anna University.

Mrs. R. Divya received his Diploma degree in Electronics & Communication Engineering from Government polytechnic College at DOTE in 2006.She received his Bachelor's degree in Instrumentation Technology from PES institute of technology Bangalore south campus at visvesvaraya technological university in 2011. At present she is pursuing M.E. with the specialization of VLSI Design in PPG institute of technology at Anna University.