Technique to Dynamically Reconfigure FPGAs using Control Registers

Sreenath Thangarajan, Reena Monica P.

Abstract: Due to increased use of FPGAs computation intensive applications, the need for embedded processing system integrated with programmable logic device has also increased. Configuration of programmable logic device by the processing system through its interface improves the efficiency of the device. In order to operate as a stand-alone device and to have a better efficiency, the programmable logic device must be capable of dynamically programming its own configuration memory. In this paper, we propose a configurable logic block with a control register to improve performance of the programmable logic device. The control register acts like a decentralized configuration memory array which can be programmed by other such configurable logic blocks. The FPGAs are fault tolerant devices with repetitive structures requiring high packaging density. This property of FPGA enables the use of CNTFETs for design of FPGAs. CNTFETs offer high trans-conductance and 1-D ballistic transport of electrons and holes which minimizes the power consumed by the FPGA. The proposed control register based architecture was implemented using Cadence Virtuoso using virtual source CNTFET model from Stanford University. A power reduction of 17.62% is achieved using CNTFETs when compared with FINFET at same technology node and the architecture was verified for various configurations of the control register.

Keywords: FPGA, ARM, Control Register, Dynamic configurability, CNTFET, CLB

I. INTRODUCTION

The short channel effects and manufacturing processes causes serious problems in conventional MOSFET devices below 14 nm. However, these short channel effects are addressed by various techniques [1]. When the technology node is below 7 nm, the gate loses control over the channel due to the drain induced barrier lowering effect and the punch through effect. Thus, dual gate devices (FINFET) [2] and gate all around devices (GAAFET) [3] are used to minimize these short channel effects.

With Moore's Law nearing its saturation, the need for a new material is prominent. Carbon Nanotubes (CNTs) which are essentially rolled up sheets of graphene, are considered as potential replacement for Silicon [4]. The CNTs can be either metallic or semiconducting, of which the semiconducting CNTs are used in fabrication of CNTFET (MOSFET like device). The CNTFET offers 1-D ballistic transport of electrons and holes, high current, high trans-conductance and high temperature resilience [5].

Field Programmable Gate Arrays (FPGAs) are known for their longevity and fault tolerant property hence they are used in sophisticated equipments. These two properties of FPGA are due to the logic functions being mapped to reduce strain on a particular Logic Element (LE).

This property of FPGA can be exploited to ensure correct operation of the nano-circuit using CNTFET while the CNTFET offers very high packing density. CNTFETs offers low energy delay product thus minimizing logic per watt of the FPGA.

With recent advancements in System-on-Chip (SoC) design, ARM core embedded within FPGAs enables easier design of controllers using state machines or high level languages [6] [7] instead of implementing similar structures in FPGAs. The ARM core can implement complex non-timing crucial control functions while the datapath and timing critical functions are implemented in the FPGA [6]. When the FPGA is booted, the CPU core is used to load a boot loader from the external flash memory [8] to configure the hardware of the FPGA using a configuration memory array.

Configuration memory defines the operations of each CLB after reset signal is asserted when the device is initialized. The configuration of the CLBs is static because the values stored in the configuration memory cannot be internally changed by the output of the CLB or specific operations executed in the FPGA. This work deals with dynamically reconfiguring the Configurable Logic Block (CLB) of the FPGA without an ARM core. The proposed control register architecture can dynamically reconfigure CLBs based on the initial configuration or output of specific Logic Elements (LE).

In this paper, we discuss about the design of FPGA using CNTFETs and techniques for implementing dynamic reconfigurability in FPGAs using control registers. Section II briefs the design and implementation of FPGAs with ARM core. The implementation of XILINX 7-series CLBs with CNTFET is discussed in Section III, while discussing the architecture of the control register in Section IV. Section V presents the results and discussions for CNTFET implementation of FPGA and control register architecture for dynamic reconfigurability. Section VI of the paper concludes.

II. FPGAS WITH ARM CORE

The renowned FPGA with ARM core is the Zynq 7000 family which is based on the Xilinx SoC Architecture. In general, the system has two components, the processing system (PS) and the programmable logic (PL). The PS is single or dual core ARM Cortex - A9 MP Core while the PL is the Xilinx FPGA [8].

Some of the important resources of the PS are application processor unit, central interconnect, memory interfaces and I/O peripherals. The PL block is designed using XILINX 7-Series system resources. The system resources of the PL are CLBs, DSPs, block RAM, ADC and transceivers.

The interconnect in PS is based on AXI high performance datapath switches. The central interconnect in PS connects the I/O peripherals and DMA controller to the DDR memory controller and on-chip RAM and the OCM interconnect provides the APU access to 256 KB memory from the central interconnect and the PL.

For configuring the SoC, the PS and PL are configured separately. In master mode boot, the boot ROM configures the PS to access the boot device and then copies the user code to the OCM memory. The hardware stages for the PS boot include, initial power supply ramping, clocking, resets, pin strap sampling and PLL initialization. The PL is powered up when the PS boots up. The boot process of the PL are powering up the PL voltage, initialization using PS software, and configuration through PS PCAP, JTAG or PL ICAP and finally enabling PS-PL interface using PS software.

III. IMPLEMENTATION OF XILINX 7-SERIES CLB USING CNTFET

The 7-Series FPGA consists of four important components namely, CLB, Block RAM, transceivers and DSP Slices. The CLBs are the basic logic element of the FPGAs which evaluates logical functions by accessing the output stored in a memory array. In this section, we discuss in detail about implementation of XILINX 7-Series CLBs using CNTFET.

A. XILINX 7-Series CLB

The XILINX 7-Series CLBs has two slices, Slice L and Slice M. Each slice in the FPGA consists of four 6-input LUTs, a carry chain, a wide function multiplexer, four optional flip flops, four output multiplexers and four output Flip-Flops/Latches.

All the components of both the slices are the same except that the LUTs of the Slice L can be configured only as Logic Blocks while the LUTs of the Slice M can be configured as both Logic Blocks and distributed memory. The configuration of the LUTs is pre-defined during the boot stage and cannot be configured dynamically during runtime. The generic structure of SliceL in CLB is shown in Fig.1.

B. CNTFET Model Specification

The CNTFET model used in this work is Virtual Source CNTFET (VS-CNTFET) model from Stanford University [9]. The VS-CNTFET model is a Gate All Around FET device with a channel length of 10 nm. The VS-CNTFET model is based on experimental data and hence they are modeled accurately. The supply voltage as specified in the model is 0.71 V and a threshold voltage (V_{th}) of 0.35 V. To have good noise margins, the threshold voltage of the CNTFET must be varied which is done by varying the diameter of the CNT.

C. CNTFET - Threshold Voltage Engineering

To have a good noise margin, the threshold voltage should be at least one third of the supply voltage. Since, the supply voltage of the CNTFET is chosen as 0.71 V, the threshold voltage (V_{th}) of the device must be 0.25 V. The energy bandgap (E_g) and thus the threshold Voltage (V_{th}) are dependent on the diameter of the CNT (d) which is given by (1) and (2),

$$E_g = \frac{0.8eV}{d} \qquad (1)$$

$$V_{th} = \frac{E_g}{2e} = \frac{0.43}{d} \qquad (2)$$

The diameter of the CNT is in turn given by the chiral vectors (n, m) of the CNT as shown in (3),

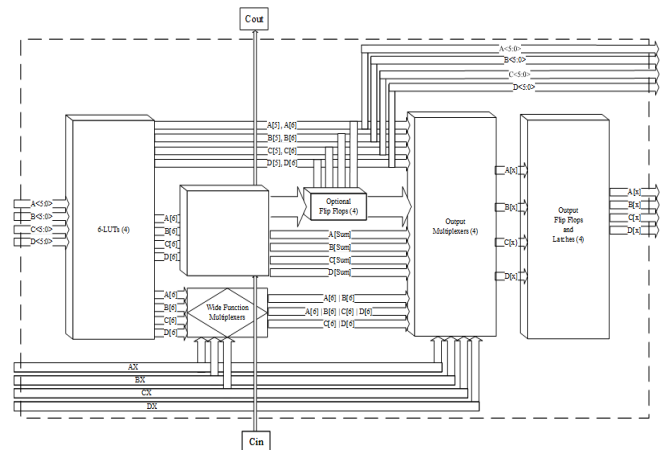$$d = 0.783\sqrt{n^2 + m^2 + nm} \qquad (3)$$

Fig.1 Xilinx7-Series Configurable Logic Block

D. FINFET – Model Specification

The FINFET [9] is a multi-gate device which has a better control over the channel thus reducing short channel effects. Here, the FINFET PTM model is used to implement the design. The model used in the design is 7nm LSTP (Low Standby Power) to achieve low power operation. The 7nm FINFET model is chosen as it has the minimum effective channel length of 10 nm which is comparable with the CNTFET model used.

E. CLB using CNTFETs and FINFETs

The CLB is implemented using VS-CNTFET model with the diameter of the CNT as 1.7 nm. The lower threshold voltage also increases the speed of operation of the device. The LUT forms the basic logic element of the CLB as it evaluates most of the logical and arithmetic operations in the FPGA. The LUT is a memory array, which can store the outputs for a specific combination of input. The input vectors act as the address for the memory array, which can be selected at the output multiplexer.

The XILINX 7-Series CLB uses four 6-input LUTs which are arranged as two 5-input LUTs. The LUTs can be designed using D Flip-Flops or SRAM array. Though the use of D Flip-Flop reduces the delay at the output,

it occupies 52% more area than that of the SRAM based LUT design. The SRAM based LUT design also has lesser power dissipation than D Flip-Flop based LUT design.

In order to have minimum area consumption and low power dissipation, 6T SRAM cells are used. Thus, the 5- input LUT is made up of an 4 x 8 array of SRAM cells. The SRAM array has pre-timed write drivers and sense amplifiers to avoid loss of data and to ensure minimal power dissipation. The output is held by a D Flip-Flop until there is a change in the input vectors.

The carry chain of the Slice is used to take carry input from the $(n-1)^{th}$ stage and to propagate the carry to $(n+1)^{th}$ stage or consume the carry in the nth stage. The wide function multiplexers enables the conversion of the 6-input LUT to either two 7-input LUTs or one 8-input LUT. The output multiplexer chooses one output. The output of the multiplexer is either latched or stored depending upon the configuration of the CLB. The operations of all the modules other than the LUTs are controlled by the configuration memory. The Fig. 2 shows the generic architecture of CLBs connected to the configuration memory.
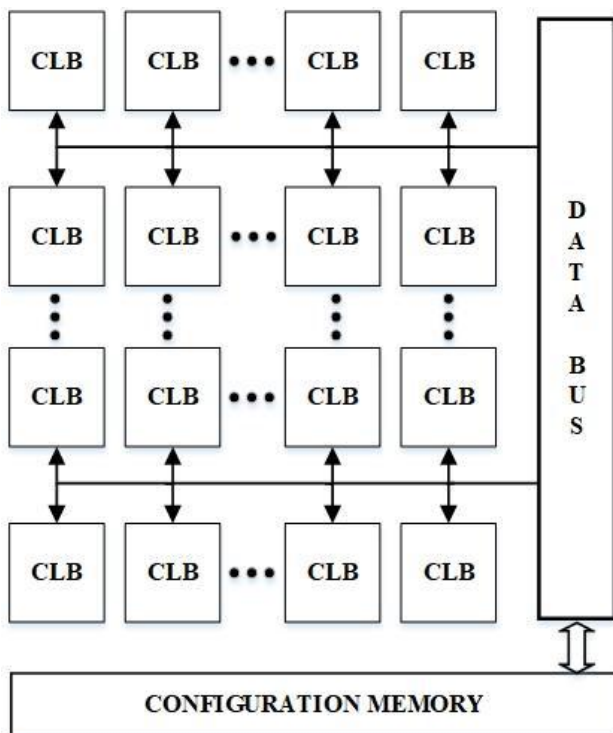

Fig. 2 FPGA with centralized Configuration Memory

IV. CONTROL REGISTER ARCHITECTURE FOR DYNAMIC RECONFIGURABILITY

The configuration memory is a centralized memory array as shown in Fig. 3, which defines all the operations in the FPGA. The configuration memory is an SRAM array and hence it loses data if FPGA is reset or powered off. Each time the FPGA is reset, the configuration memory is loaded with configuration from an external flash memory.

In FPGAs with ARM core, the configuration memory is flashed through a CPU of the core and hence the configuration memory can be dynamically changed during runtime. In case of FPGAs without embedded controller, the configuration memory cannot be changed in run-time. In this section, we discuss in detail about the design and implementation of the

control register using CNTFET and techniques to dynamically reconfigure all the CLBs.

A. Design of Control Register

The control register must have sufficient number of registers to control all the modules in a slice of the CLB. As stated earlier, each slice has four 8:1 multiplexer at the output and thus requires 12 registers to control the same. In order to switch between Latch and Flip-Flop operation at the output, each module would require one pin and thus adding another four registers to control the operation. Likewise, the carry chain and optional flip-flops require 4 registers each.

In order to be reconfigurable, the slice must be capable of writing and reading data from the LUT in the run-time. Hence, another register is required to control the read and write operation in the slice. From the above discussion, 25 registers are required to control the operations of the slice and introduce dynamic configurability in FPGAs without ARM core. The proposed structure of the control register of the Slice is shown in Fig 3.

RD/WR'	A[CarCh]	B[CarCh]	C[CarCh]	D[CarCh]
A[FFLA]	A[OptFF]	A[S2]	A[S1]	A[S0]
B[FFLA]	B[OptFF]	B[S2]	B[S1]	B[S0]
C[FFLA]	C[OptFF]	C[S2]	C[S1]	C[S0]
D[FFLA]	D[OptFF]	D[S2]	D[S1]	D[S0]

Fig. 3 Xilinx7-Series Configurable Logic Block

The register field RD/WR' sets the mode of the configuration registers. If the register is set then the control register is set to read mode else the control register is set to write mode. Initially the values of the local control register are updated from the ROM during the boot stage. The value of control register can be updated by the embedded software or by other Slices allowing dynamic use of the programmable hardware. The read and write register allows a group Slices to be configured only during boot stage and allows a group of Slices to be configured dynamically. It also helps in locking the resource when a process is overwriting the contents of the register which prevents race conditions.

B. Dynamic Reconfigurability using Control Register

The CLBs of the FPGA cannot be dynamically configured because the contents of the configuration memory cannot be changed internally. The configuration memory is booted only from the flash memory after reset is asserted. However dynamic configuration is achieved in FPGAs with embedded controller as the contents of the configuration memory can be altered by the CPU.

The proposed control register architecture as shown in Fig. 4 ensures dynamic reconfigurability of the FPGAs by decentralizing the configuration memory. In the proposed scheme, each slice of the CLB has a control register which acts as the decentralized configuration memory.

Retrieval Number: A10241291S319/2019@BEIESP
DOI:10.35940/ijeat.A1024.1291S319

119

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

As discussed earlier, the configuration memory can control the all the operations within a slice.

The main advantage of this control register structure is that the contents of the LUT can be changed during runtime based upon the values from other slices. For instance let us consider a 2-input LUT which is configured to work as a 2-input AND gate, can be configured using the control register and outputs from other blocks to perform the operation of an OR gate or any other relevant logic functionality.
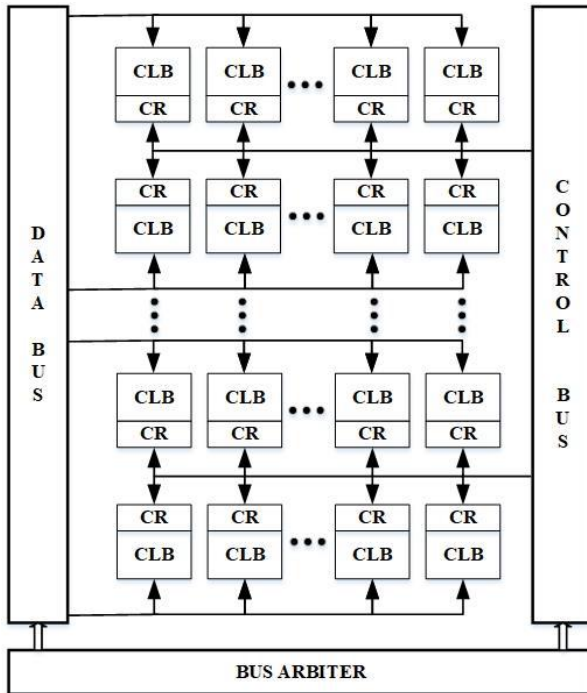

Fig. 4. FPGA with decentralized Configuration Memory

C. Interconnects for Control Register based architecture

The FPGA with ARM core and static configurable FPGAs uses ARM AMBA 3.0 AXI protocol for interconnecting all FPGA resources. The bus protocol offers high efficient data transfer through 5 unidirectional channels. Since the protocol supports unaligned data transfer [DUI0305C] through byte strobes, the contents of the control registers can be controlled by individual CLB resources. The bus protocol, ARM AMBA 3.0 AXI, which is used in [FPGA_ARM][Static] can be used for interconnecting the CLB resources in the proposed architecture.

V. RESULTS AND DISCUSSION

The proposed control register based CLB architecture was implemented using FINFET and CNTFET. The reconfigurability of the FPGA was verified using the control register in real-time and also the performance of the proposed architecture was compared based on the power dissipation of the devices. The reconfigurability of the FPGA and the power dissipation of the FPGA are discussed as follows.

A. Reconfigurability of FPGA

The Xilinx 7-Series CLBs are designed in such a way that all the operations are executed in parallel within a Slice, however its output can have multiple outputs or a single output based on a multiplexer. The select lines of this multiplexer is

controlled by the control register and thus dynamic reconfigurability can be achieved in the FPGA.

The logic was verified by initializing the operation of the Slice with the operation of AND logic in one of the LUTs. After a few operations, the values of the control register were overwritten during runtime which allowed the operation to be changes to OR logic. To overwrite the values, the control register was set to write mode. During the write phase, the output of the Slice is invalid and the contents of the LUT and the control register can be changed during this phase. In this case, the values of the control register are not altered but the values of the LUTs are changed to perform OR logic.

Similar experiment was used to verify the operation of the control register in which the operation was altered to change the output of the Slice from a 5 input LUT to a 6 input LUT which was again changed to an output from 7 input LUT. The process was achieved by changing the values of the multiplexer lines of the control register based on Table 1. From Table I, it can be seen that the select lines combination 000 selects the output O6 while 001 selects the output O5. The wide chain multiplexer select lines and the optional flip-flops were also configured with the control register and its outputs were verified.

The logic was also verified similarly for the carry chain logic operation in which the logic operation for carry forward was blocked and unblocked for a carry output from another Slice. Similarly, the logic was verified for overwriting the contents of the control register and the values of the LUT dynamically.

Table - I: Multiplexer or Latched output of the multiplexer

Multiplexer Select Line	Multiplexer Output
000	Selects the output of the LUT when configured to 6-input LUT
001	Selects the output of the LUT when configured to 5-input LUT
010	Bypasses input from previous CLB (or) carry output of the carry chain
011	Carry output of the carry chain (or) 7-input LUT / 8-input LUT output
100	Sum output of the Carry chain (or) 7-input LUT / 8-input LUT output
101	Output of the optional Flip Flop (or) sum output of the carry chain
110	No Operation
111	No Operation

B. Power Dissipation using CNTFET and FINFET

The power dissipation of the proposed architecture using both CNTFET and FINFET was calculated and the results were analyzed. The graph in Fig.5 shows the comparison of the power dissipation of CNTFET and FINFET of the various modules in the design.
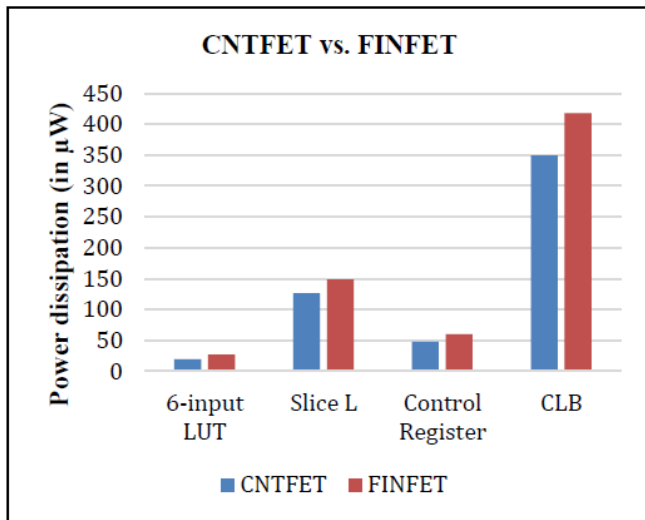
Fig. 5 Power dissipation (in µW) – CNTFET vs. FINFET

From the graph in Fig.5, it can be seen that the power dissipation of the FINFET was much higher than that of the power dissipation using CNTFET. The CNTFET implementation has 19.72% less power than that of the FINFET implementation.

VI. CONCLUSION

In this paper, architecture to dynamically reconfigure the FPGA is proposed by dynamically controlling the CLBs of the FPGA through control registers. The proposed structure is implemented in Cadence Virtuoso using CNTFET and FINFET. From the results, it can be observed that the power dissipation of the design using CNTFET is less than that of the design using FINFET. An overall power reduction of 19.72% is achieved using CNTFETs when compared with the FINFETs both having effective length of 10 nm. The dynamic reconfigurability of the FPGA is achieved by configuring the contents of the control register and can also enable rewriting the contents of the LUT during execution. The proposed structure can reconfigure its own configuration memory without an embedded micro-controller thus improving the logic efficiency of the FPGA at a reduced cost.

REFERENCES

1. A. Chaudhry and M. J. Kumar, "Controlling short-channel effects in deep submicron SOI MOSFETs for improved reliability: A review," *arXiv preprint arXiv:1008.2427,* 2010.
2. D. Hisamoto, W.-C. Lee, J. Kedzierski, H. Takeuchi, K. Asano, C. Kuo, E. Anderson, T.-J. King, J. Bokor and C. Hu, "FinFET-a self-aligned double-gate MOSFET scalable to 20 nm," *IEEE Transactions on Electron Devices,* vol. 47, no. 12, pp. 2320-2325, 2000.
3. Y.-C. Huang, M.-H. Chiang, S.-J. Wang and J. G. Fossum, "GAAFET Versus Pragmatic FinFET at the 5nm Si-Based CMOS Technology Node," *IEEE Journal of the Electron Devices Society,* vol. 5, no. 3, pp. 164-169, 2017.
4. M. M. Shulaker, G. Hills, N. Patil, H. Wei, H.-Y. Chen, H.-S. P. Wong and S. Mitra, "Carbon nanotube computer," *Nature,* vol. 501, no. 7468, p. 526, 2013.
5. Z. Kordrostami and M. H. Sheikhi, "Fundamental physical aspects of carbon nanotube transistors," *Carbon Nanotubes Intech,* 2010.
6. Z. Hajduk, "An FPGA embedded microcontroller," *Elseiver Microprocessors and Microsystems,* vol. 38, no. 1, 2014.
7. U. Meyer-Baese and U. Meyer-Baese, Digital signal processing with field programmable gate arrays, vol. 2, 2004.
8. Xilinx, "Zynq-7000 SoC Technical Reference Manual".
9. J. Luo, L. Wei, C.-S. Lee, A. D. Franklin, X. Guan, E. Pop, D. A. Antoniadis and H.-S. P. Wong, "Compact model for carbon nanotube field-effect transistors including nonidealities and calibrated with experimental data down to 9-nm gate length," *IEEE transactions on electron devices,* vol. 60, no. 6, pp. 1834-1843, 2013.
10. R. Hajare, C. Lakshminarayana, S. C. Sumanth and A. Anish, "Design and evaluation of FinFET based digital circuits for high speed ICs," *International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT),* pp. 162-167, 2015.

AUTHORS PROFILE

Sreenath Thangarajan is currently working as Silicon Design Engineer at Advanced Micro Devices India Pvt. Ltd, Bangalore. He pursued his M. Tech in VLSI Design from Vellore Institute of Technology, Chennai, India. He is a gold medalist. He completed his B. Tech from Anna University, Chennai. As a part of his M.Tech, he pursued internship for a year with Intel India, Bangalore. He has worked with emerging technologies such as CNTFETs. He has implemented FPGA architecture with 10 nm CNTFETs. He has won many make-a-thons conducted during his M.Tech. He has good knowledge in Computer Architecture and is interested in exploring newer technologies.
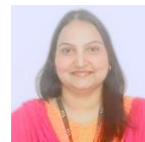
Reena Monica P is currently working with the School of Electronics Engineering at Vellore Institute of Technology, Chennai, India since 2010. She holds a Ph.D. in Nanoelectronics from Vellore Institute of Technology, Chennai, Tamilnadu, India. She received her M.Tech in VLSI design degree from SRM Institute of Technology, Chennai, India in 2005 and the B.E degree in electrical and electronics engineering from Karunya Institute of Technology, Coimbatore, India in the year 2003. Her research interests include VLSI Design, VLSI Digital Signal Processing, hybrid CMOS/Nanoelectronic devices for digital applications, and CNTFETs. She has fabricated CNTFETs and analyzed them for digital circuits.