INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

EFFICIENT DIMINISHED-1 MODULO 2^n+1 MAC UNIT ARCHITECTURE

Y. Sarath babu[1], G.Naresh[2]

[1]PG Student, VLSI, Sree Vidyanikethan Engineering College, Tirupati, Chittoor, A.P, India
[2]Assistant Professor, ECE Dept., Sree Vidyanikethan Engineering College, Tirupati, Chittoor, A.P, India.

Abstract- Modulo arithmetic is a system of arithmetic for integers. And this modulo arithmetic plays an important role in the field of cryptography. It uses Residue Number System (RNS),in which a number is represented in non-weighted form. Various operations can be performed in modulo arithmetic. They are: addition, subtraction, multiplication, division. In this paper, modulo MAC unit is presented by combining the modulo multiplier and the modulo adder. In this MAC unit, diminished-1 number system is used. In which, only n-bits are required to represent a number in the range of $[0,2^n]$. The proposed MAC unit is analysed using Xilinx 10.1i.

Key words: cryptography, Residue Number System (RNS), diminished-1 number system.

1. Introduction

1.1 Residue Number System (RNS)

Residue Number System is an efficient alternative number system in the arithmetic circuits.RNS results in parallel and high speed computations .In this RNS, a number is represented in non-weighted form. And in this RNS, a weighted number is divided into small non-weighted numbers [3]. Each small number called as the residue. The arithmetic operations are performed on this smaller residues. And there is no chance for carry propagation. Hence the speed of the circuit will increases.

There are 4 major issues in designing of RNS. They are

a) Moduli set selection
   i) 3 moduli set
   ii) 4 moduli set
   iii) 5 moduli set
b) Forward conversion
c) Residue Arithmetic unit
d) Reverse conversion

In forward conversion block, a weighted number is converted into a non-weighted number based on the selection of moduli set. In residue arithmetic unit, operations are performed on this residue. The various operations that can be performed on the Residue Arithmetic unit are addition, subtraction, multiplication, division. Then in the reverse conversion non-weighted number is converted into weighted number.

1.2 Diminished-1 number system

In general, to represent a number in the range of $[0,2^n]$ we require (n+1) bits. But in this diminished-1 number system we require only n bits. If the number of bits are less, then the memory requirement will decreases and the speed will increases. For example, squaring of 9 requires 5 bits as the input. But in diminished-1 number system, it requires only 4 bits as input [1].

In diminished-1 number system, a number 'x' is represented as

$X=Xz.X_{-1}$

Xz= zero indication bit

x_{-1}=magnitude of number (x-1)

$$X_z = \begin{cases} 0 & \text{if } x\neq 0 \\ 1 & \text{If } x=0 \end{cases} \qquad X_{-1} = \begin{cases} X-1 & \text{if } x\neq 0 \\ 0 & \text{If } x=0 \end{cases}$$

2. Modulo 2^n+1 Adder

The modulo 2^n+1 adder also uses the diminished-1 number system for both inputs and inputs. The algorithm to do the addition of two numbers is given below by taking the moduli set as3 moduli set(5,7,9).

Example: 53+72=125
RNS representation of weighted codes can be computed as

$\|53\|_5=3$	$\|53\|_7=4$	$\|53\|_9=8$	53=(3,4,8)
$\|72\|_5=2$	$\|72\|_7=2$	$\|22\|_9=0$	72=(2,2,0)

And the addition can be performed as

$\|3+2\|_5=0$	$\|4+2\|_7=6$	$\|8+0\|_9=8$	(0,6,8)
$\|125\|_5=0$	$\|125\|_7=6$	$\|125\|_9=8$	(0,6,8)

In RNS, with respect to the moduli set (5,7,9), 53 can be represented as(3,4,8). And 72 can be represented as (2,2,0). To perform the addition of these two numbers 53 and 72, we need to perform the addition on these residues (3,4,8) and (2,2,0) with respect to the same moduli set. In reverse conversion block, these non-weighted residues are converted into a weighted number. The block diagram for the modulo 2^n+1 adder is shown in the below figure [2].

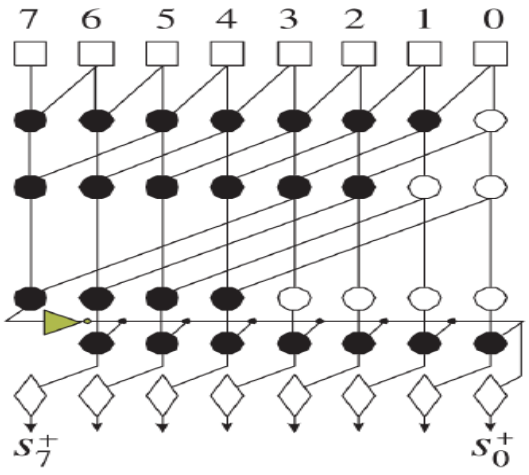


Fig1: modulo 2^n+1 adder

The various blocks used in this adder are

i) Square cell
ii) Dot cell
iii) Diamond cell

The logic level implementation of these blocks are given as

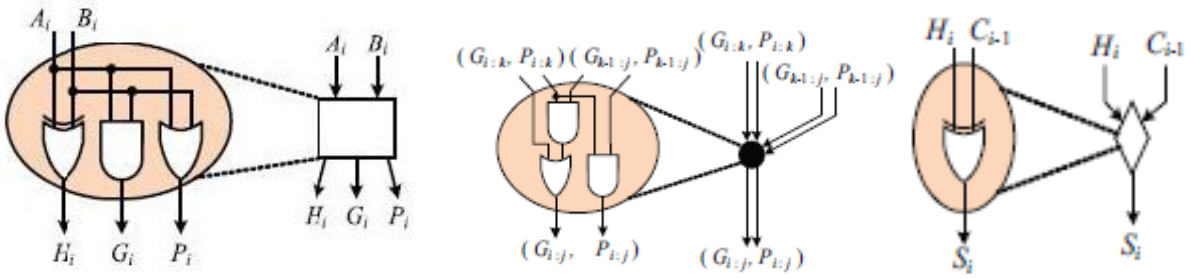


Fig 2: logic implementations of various cells

The carry and the final sum are calculated as

$C_i=g_i+p_i.c_{-1}$

$S_i=h_i \wedge c_{i-1}$

3. Diminished-1 modulo 2^n+1 multiplier

Modulo multipliers plays a major role in cryptography, where the data is encrypted and decrypted for secure data transmission. Here in this paper, the modulo multiplier uses the Diminished-1 number system. Various algorithms are existed for designing of modulo multipliers. Yutai ma[7]Proposed an algorithm, which uses 2 modulo cspa adders for addition of resultant Sum bits and carry bits. Leonel sousa[5] proposed an algorithm, which uses multiplexers for generation of partial products. T.Vergos[6] Proposed an architecture, which uses SFA for generating the partial products. But it can't deal with handling of zero inputs. J.W.Chen[4] Proposed an algorithm, which uses BS and BE blocks for generation of partial products. The above all architectures have some drawbacks in their partial products generation, reduction and addition blocks.

An efficient architecture is presented in this paper, for generation and reduction of partial products. It uses 3 major modules.

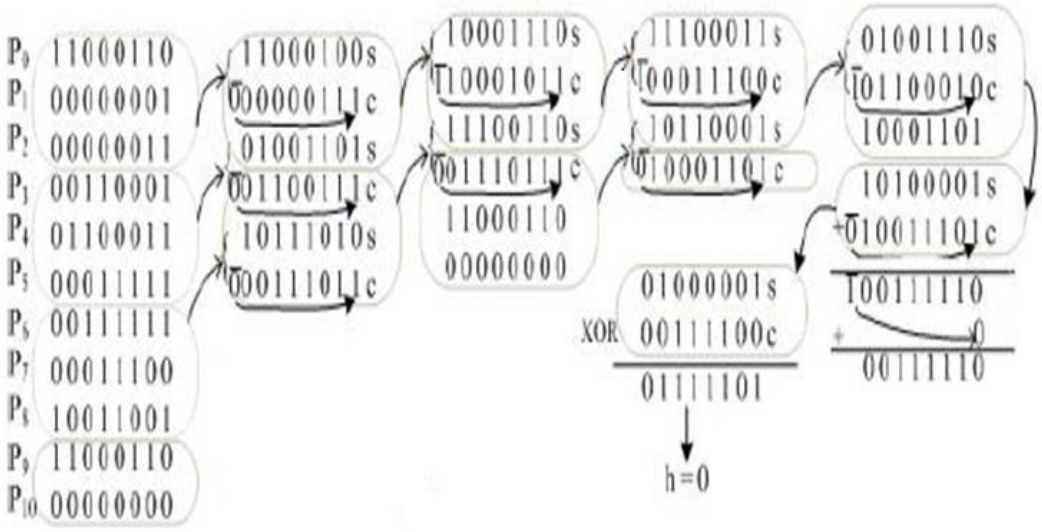
1) Partial products generation
2) Partial products reduction
3) Addition of sum and carry bits

Partial products are generated using the following architecture with the help of NAND and AND gates. The partial products generation block is shown below. In partial products reduction block, addition can be performed for three partial products from these resultant sum and carry generated. The MSB of the carry is inverted and placed at the LSB position. The same procedure will repeated until the single row of resultant sum and carry bits are generated.

$$
\begin{array}{lllll}
P_0 = & A_0B_{n-1} & A_0B_{n-2} \cdots & A_0B_1 & A_0B_0 \\
P_1 = & A_1B_{n-2} & A_1B_{n-3} \cdots & A_1B_0 & \overline{A_1B_{n-1}} \\
P_2 = & A_2B_{n-3} & A_2B_{n-4} \cdots & \overline{A_2B_{n-1}} & \overline{A_2B_{n-2}} \\
\cdots & & & & \\
P_{n-2} = & A_{n-2}B_1 & A_{n-2}B_0 \cdots & \overline{A_{n-2}B_3} & \overline{A_{n-2}B_2} \\
P_{n-1} = & A_{n-1}B_0 & \overline{A_{n-1}B_{n-1}} \cdots & \overline{A_{n-1}B_2} & \overline{A_{n-1}B_1} \\
P_n = & A_{n-1} & A_{n-2} \cdots & A_1 & A_0 \\
P_{n+1} = & B_{n-1} & B_{n-2} \cdots & B_1 & B_0
\end{array}
$$

Partial products generation block

The partial products reduction block is shown below by taking the numbers 99 and c6 . And in the last stage, resultant sum and carry bits will added to get the result.



Partial products reduction block

After generation of the resultant sum and carry, these will be given to the modulo 2^n+1 adder. And from the modulo 2^n+1 adder, the result will be obtained. The diagrammatic representation of the modulo 2^n+1 multiplier Architecture is shown in the below figure (3).

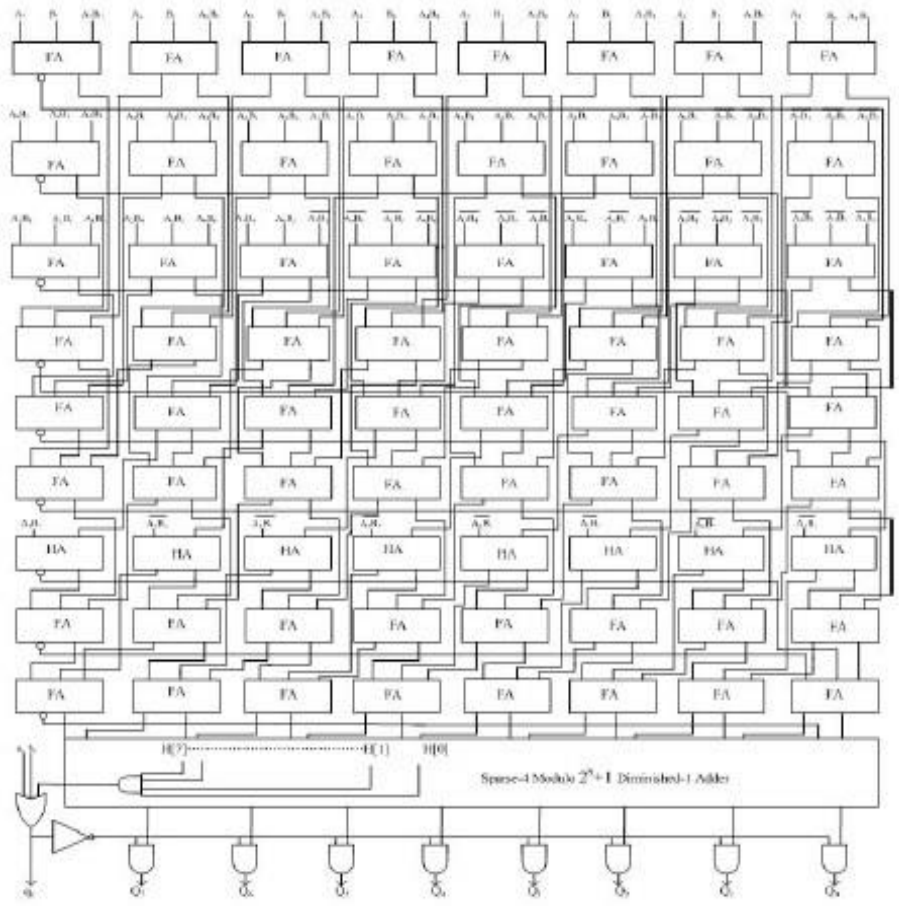


Fig 3: modulo 2^n+1 multiplier

From the above figure, it can be shown that, after performing the addition of three partial products, the sums and carries are generated. And the MSB carry is inverted and added at the LSB position. The same

procedure will be repeated until the end of partial products. The final sum and carry bits are given to the modulo adder. The 'h' bit is generated by using AND of XOR of resultant sum and carry bits. The additional feature of the presented modulo multiplier is that, it can handle the zero inputs and results also.

According to handle zero inputs and result, the product of a diminished-1 modulo multiplier is derived according to the following cases:

1. When one of the two inputs is zero, the result is zero.
2. When none of the input operands is zero, the result is nonzero.

If the full adders are replaced by the 3:2compressor circuit, then there is a improvement in the delay and power for higher order multiplications. The circuit diagram of the 3:2 compressor is shown below.

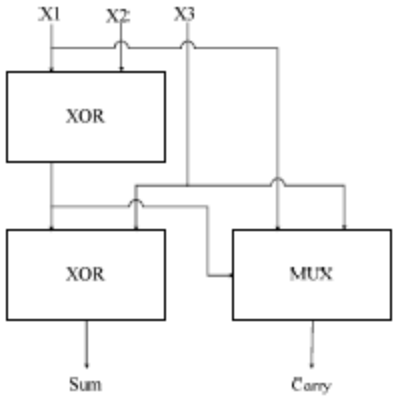


Fig 4) 3:2 compressor

4. Modulo 2^n+1 MAC unit architecture

The modulo 2^n+1 Multiplication and Accumulation (MAC) unit consisting of two blocks namely, modulo multiplier followed by modulo 2^n+1 adder. Both the circuits uses diminished-1 number systems. The block diagram of the MAC unit is shown below

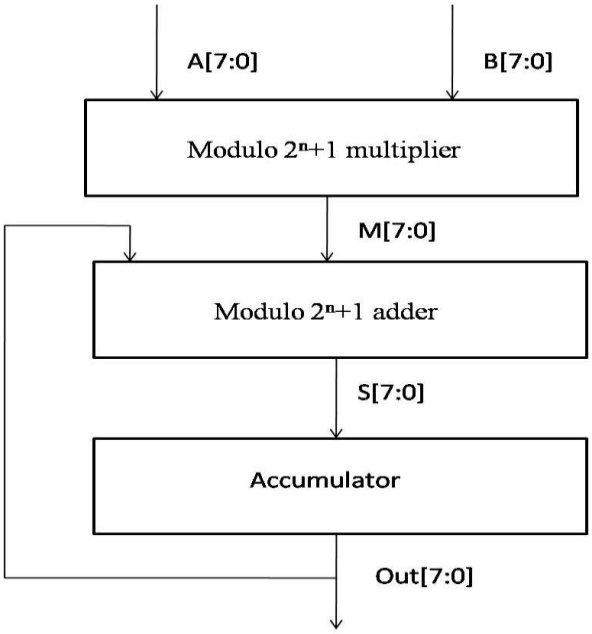
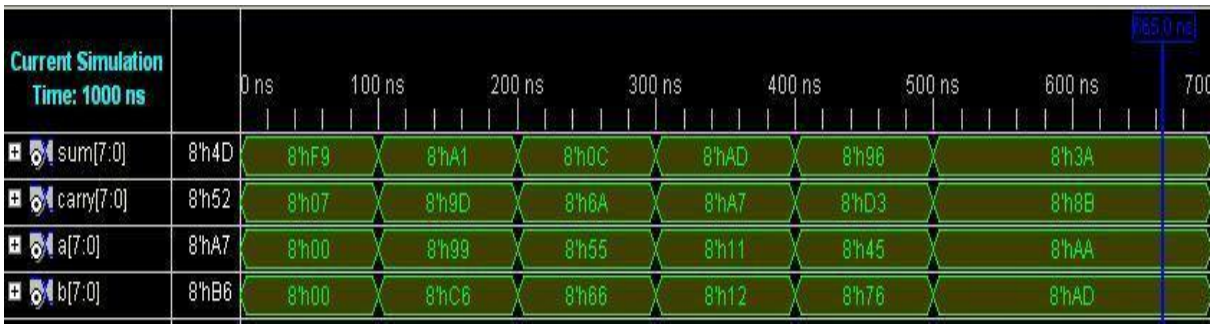


Fig 5: Modulo 2^n+1 MAC unit

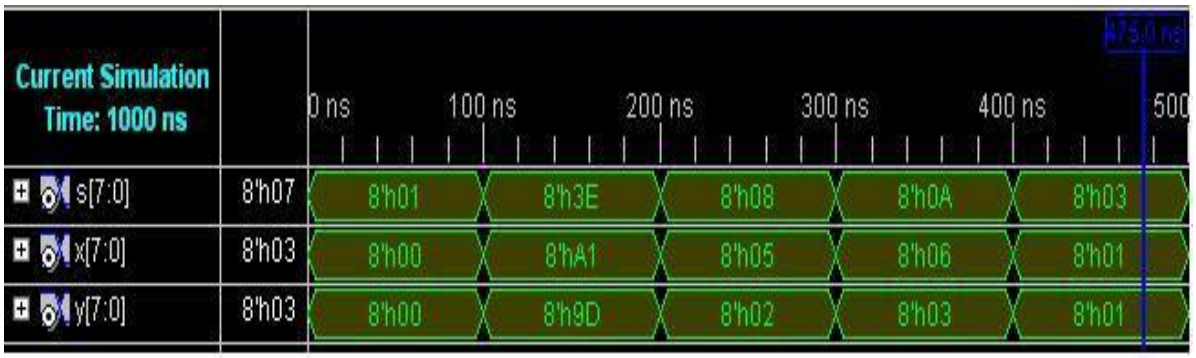
The inputs of the multiplier are 8bits each. As here, it is a modulo multiplier it produces the 8 bits as the output for the given 8 bits input. And this 8 bits are given to the modulo adder. From the modulo adder the result is moved to accumulator. Based on the clock and reset, the output of accumulator will be added to the modulo adder for the next iteration. It can also be handles the zero inputs.

5. Simulation Results & Discussions

The simulation results for the various blocks are shown below.

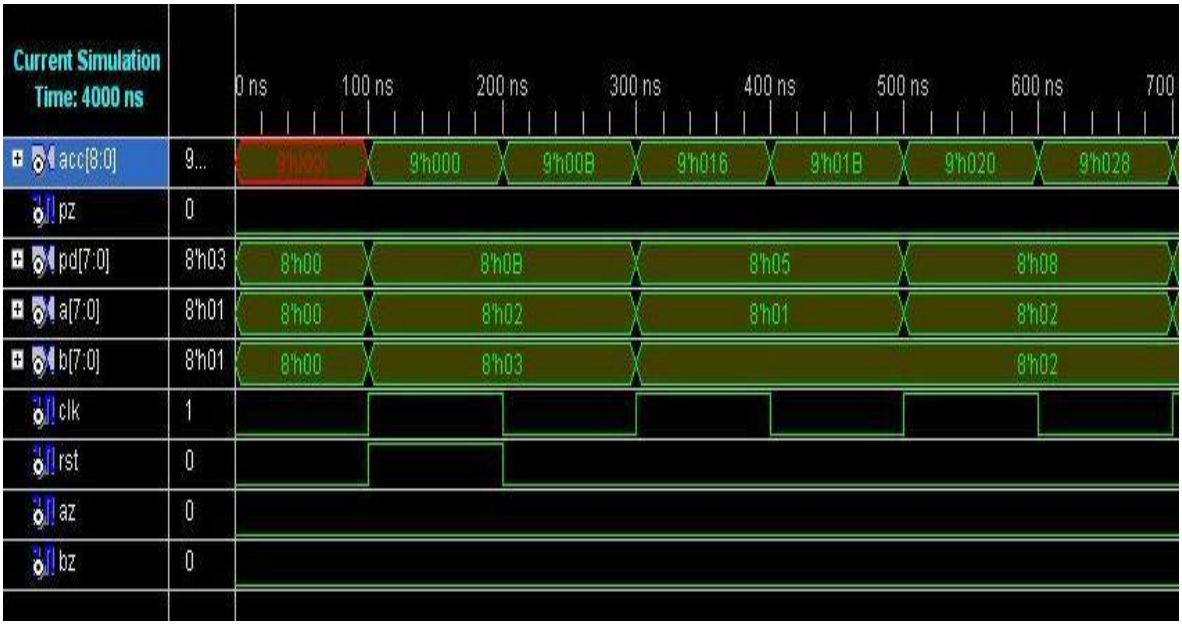


Simulation result of partial products reduction block



Simulation result of modulo 2^n+1 adder

Simulation result of modulo 2^n+1 multiplier



Simulation result of modulo 2^n+1 MAC unit

In modulo 2^n+1 multiplier, if any one of the input is zero, then the output will be the zero. To give the input 'a' as zero, set the zero bit indication bit a_z=1. And for b, b_z=0.

In modulo 2^n+1 MAC unit, the circuit is active for every positive edge. When reset=1 the result is added to the accumulator. And for positive clock, the multiplication can be done. And in the negative clock pulse, the multiplier output will be added to the accumulator.

The synthesis reports for the above blocks are shown below. The simulation has been done by using Xilinx ISE simulator 10.1i and the power is calculated by using Xilinx xpower-analyzer.

parameter	Modulo 2^n+1 adder	Modulo 2^n+1 multiplier	Modulo 2^n+1 MAC
Delay(ns)	17	19	29
Power(mw)	34	74	98

Conclusion

.

In modulo arithmetic, modulo MAC can also be implemented besides modulo addition, subtraction, multiplication. The conventional arithmetic carry propagation based on weighted number system causes the performance degradation. In this paper, we have presented the efficient modulo 2^n+1 multiplier architecture which uses RNS in which, the carry is not propagated. Modulo multiplier requires less delay for the generation and reduction of partial products. For addition of resultant sum and carry, End around Inverted Carry modulo 2^n+1 adder is used. Modulo 2^n+1 MAC unit is designed by combining efficient diminished-1 modulo 2^n+1 multiplier and modulo 2^n+1 adder with minimum delay and low power requirements. The design is implemented in Xilinx ISE10.1i Tool for XC3S500E-5FG320 that dissipates 98mW with a corresponding delay of 29ns. This modulo 2^n+1 MAC unit can be used in most challenging applications of Cryptography.

REFERENCES

[1] Xiaolan Lv, Ruohe Yao "*Efficient Diminished-1 Modulo 2n+1 Multiplier Architecture*" ," IEEE Trans.Comput.,2014., pp 481-486

[2] Haridimos T.vergos and Giorgos Dimitra kopoulos " *On modulo 2n+1 Adder Design* '' IEEE Trans..,February 2012, vol 61, pp.173-186

[3] K Navi, A.S Molahosseini and M. Esmaeildoust, *" How to teach residue number system to computer scientists and engineer* ," IEEE Trans Educ.,2011, 54(1): 156-163.

[4] J. W. Chen and R. H Yao, *"Efficient modulo multipliers for diminished-1 representation*," IET Circuits, Devices Syst., 2010, 4(4), 291–300.

[5] L. Sousa and R. Chaves, "*A universal architecture for designing efficient modulo 2n+1 multipliers*" IEEE Trans Circuits Syst. I., 2005, 52(6), 1166–1178.

[6] C. Efstathiou, H. T. Vergos, G. Dimitrakopoulos, and D. Nikolos,"*Efficient diminished-1 modulo 2n+1 multipliers*," IEEE Trans.Comput., 2005, 54(4), 491–496.

[7] Y. Ma, "*A simplified architecture for modulo (2n+1) multiplication*," IEEE Trans. Comput., 1998, 47,(3), pp. 333–337.

Biography

Y. Sarath babu: He is currently pursing M.tech (VLSI) in Sree Vidyanikethan Engineering College, Tirupati. His areas of interests are Digital System Design and VLSI Design.

G.Naresh: He is currently working as an Assistant Professor in ECE department of Sree-Vidyanikethan Engineering College, Tirupati. He has completed M.tech (VLSI Design), in Sathyabama University. His research areas are low power VLSI Design.