International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 3, Issue 11, November 2014

Design of a MP-SOC using on chip interconnections

Kodali Radha, Dr.D.Venkata Rao

Abstract— **This paper presents the silicon-proven design of a novel on-chip network to support guaranteed traffic permutation in multiprocessor system-on-chip applications. The proposed network employs a pipelined circuit-switching approach combined with a dynamic path-setup scheme under a multistage network topology. The dynamic path-setup scheme enables runtime path arrangement for arbitrary traffic permutations. The circuit-switching approach offers a guarantee of permuted data and its compact overhead enables the benefit of stacking multiple networks.Interconnection of various processing elements with guaranteed traffic permutation and optimality of Scheduling over the Network-On-Chip has been reported in this paper. It supports to reduce the complexity of on chip network implementation. The proposed network topology contains a dynamic probing mechanism to interconnect different processing elements to make a path setup dynamically by using the probe routing algorithm, multistage network topology and communication locality on Network-on-Chip were implemented with reduced area and speed constraints on chip.**

Index Terms—**Guaranteed throughput, multistage interconnection network, network-on-chip, permutation network, pipelined circuit-switching, traffic permutation, probe routing algorithm, multistage network topology**

I. INTRODUCTION

A trend of multiprocessor system-on-chip (MPSoC) design being interconnected with on-chip networks is currently emerging for applications of parallel processing, scientific computing, and so on [1]–[6].Permutation traffic, a traffic pattern in which each input sends traffic to exactly one output
and each output receives traffic from exactly one input, is one of the important traffic classes exhibited from on-chip multiprocessing applications [7], [8]. Standard permutations of traffic occur in general-purpose MPSoCs, for example,polynomial, sorting, and fast Fourier transform (FFT) computations cause shuffled permutation, whereas matrix transposes or corner – turn operations exhibit transpose permutationn [6]. Recently,application-specific MPSoCs
targeting flexible Turbo/LDPC decoding have been developed, and they exhibit arbitrary and concurrent traffic permutations due to multi-mode and multi-standard feature [3]–[5]. In addition, many of the MPSoC applications (e.g., Turbo/LDPC decoding [3]–[5]) compute in real-time, therefore, guaranteeing throughput (i.e., data lossless, predictable latency, guaranteed bandwidth, and in-order delivery) is critical for such permutation traffics.

Kodali Radha, *Student/ECE, JNTU Kakinda/ Narasaraopeta Institute Of Technology College / Engineering College,. Narasaraopeta, India, 9052231093/*
Dr.Dhulipala Venkata Rao, *Professor/ECE JNTU Kakinda/ Narasaraopeta Institute Of Technology College / Engineering College, ., Narasaraopeta, India, 9963878335*

II. APPROACHES USED

This approach mainly depends on three factors. They are
(A)Switching techniques
(B) Topology
(C)Routing algorithm

A. Switching techniques

In large networks there might be multiple paths linking sender and receiver. Information may be switched as it travels through various communication channels. There are three typical switching techniques available for digital traffic.

(i) Circuit Switching
(ii) Message Switching
(iii) Packet Switching

(i) *Circuit Switching*
Circuit switching is a technique that directly connects the sender and the receiver in an unbroken path. Telephone switching equipment, for example, establishes a path that connects the caller's telephone to the receiver's telephone by making a physical connection .With this type of switching technique, once a connection is established, a dedicated path exists between both ends until the connection is terminated. Routing decisions must be made when the circuit is first established, but there are no decisions made after that time.

(ii) Message Switching
With message switching there is no need to establish a dedicated path between two stations. When a station sends a message, the destination address is appended to the message. The message is then transmitted through the network, in its entirety, from node to node. Each node receives the entire message, stores it in its entirety on disk, and then transmits the message to the next node(Fig 1).This type of network is called a store-and-forward network

ISSN: 2278 – 909X

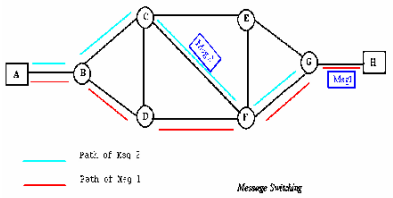

Fig 1:Message Switching

(iii) Packet Switching

There are two methods of packet switching: Datagram and virtual circuit. In both packet switching methods, a message is broken into small parts, called packets. Each packet is tagged with appropriate source and destination addresses. Since packets have a strictly defined maximum length, they can be stored in main memory instead of disk, (Fig 2) therefore access delay and cost are minimize. Also the transmission speeds, between nodes, are optimized. With current technology, packets are generally accepted onto the network on a first-come, first-served basis. If the network becomes overloaded, packets are delayed or discarded (``dropped").

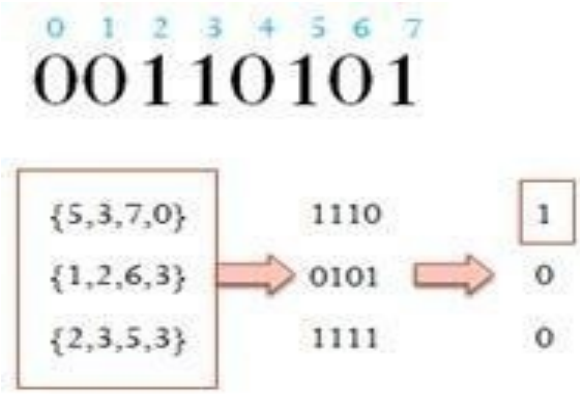

Fig 2: Packet switching

B. Topology

The topology used here is "3-stage clos". It provides0.13µm CMOS technology with low power consumption packet data transmission. This topology provides better throughput with less area requirements (Fig 3).

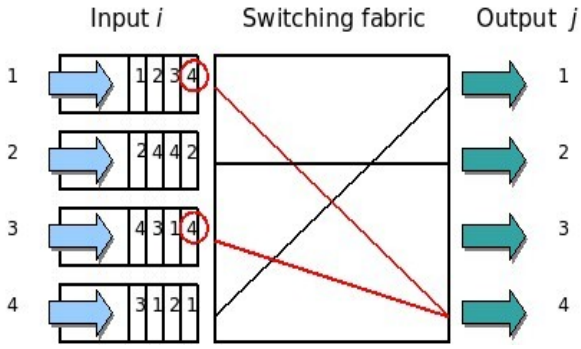

Fig 3: Packet switching with 3-stage clos topology

C. Routing algorithm

Routing is the process that a router uses to forward packets toward the destination network. A router makes decisions based upon the destination IP address of a packet. In order to make the correct decisions, routers must learn the direction to remote networks (Fig 4). Two types of routing are used: static and dynamic routing [1].When routers use dynamic routing, this information is learned from other routers. When static routing is used, a network administrator configures information about remote networks manually

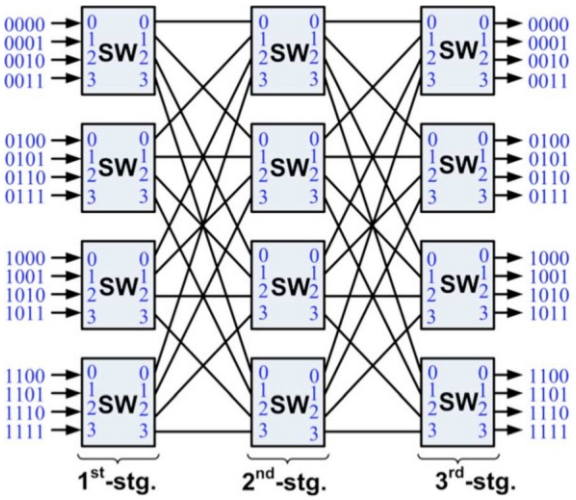

Fig. 4. Proposed on-chip network topology with port addressing scheme

III. PROPOSED ON-CHIP NETWORK DESIGN

As motivated in Section I, the key idea of proposed on-chip network design is based on a pipelined circuit-switching approach with a dynamic path-setup scheme supporting runtime path arrangement. Be- fore mentioning the dynamic path-setup scheme, the network topology is first discussed. Then the designs of switching nodes are presented.

A. On-Chip Network Topology

Clos network, a family of multistage networks,is applied to build scalable commercial multiprocessors with thousands of nodes in macro systems [7], [11]. A typical three-stage Clos network is defined as C(n,m,p), where represents the number of inputs in each of first-stage switches and is the number of second-stageswitches. In order to support a parallelism degree of 16 as in most practical MPSoCs [3]–[5], we proposed to useC(4,4,4) as a topology for the designed network (see Fig. 5). This network has a re-arrangable property [11] that can realize all possible permutations between its input and outputs. The choice of the three-stage Clos network with a modest number of middle-stage switches is to minimize implementation cost, whereas it still enables a rearrangable property for the network.

A pipelined circuit-switching scheme is designed for use with the proposed network. This scheme has three phases: the *setup*, the *transfer*, and the *release* [2], [9]. A dynamic path-setup scheme supporting the runtime path arrangement occurs in the setup phase. In order to support this circuit-switching scheme, a switch-by-switch interconnection

ISSN: 2278 – 909X

with its handshake signals is proposed, as shown in Fig. 5. The bit format of the handshake includes a 1-bit *Request (Req)* and a 2-bit *Answer (Ans)*. Req=1 is used when a switch requests an idle link leading to the corresponding downstream switch in the setup phase. The Req=1 is also kept during data transfer along the set up path. A Req=0 denotes that the switch releases the occupied link. This code is also used in both the setup and the release References phases. An *Ans=01(Ack)* means that the destination is ready to receive data from the source. When the *Ans=01* propagates back to the source, it denotes that the path is set up, then a data transfer can be started immediately. An *Ans=11(nAck)* is reserved for end-to-end flow control when the receiving circuit is not ready to receive data due to being busy with other tasks, or overflow at the receiving buffer, etc. An *Ans=10*(*Back*) means that the link is blocked. This *Back* code is used for a backpressure flow control of the dynamic path-setup scheme, which is discussed in the following subsection

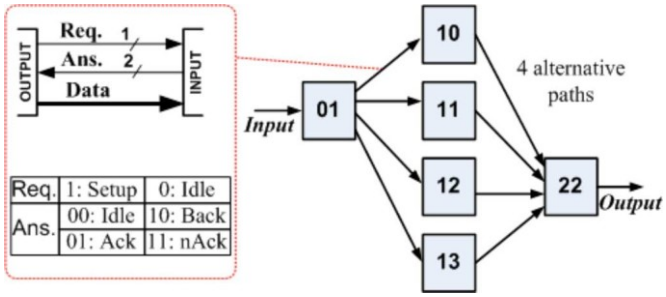

Fig. 5. Switch-by-switch interconnection and path-diversity capacity

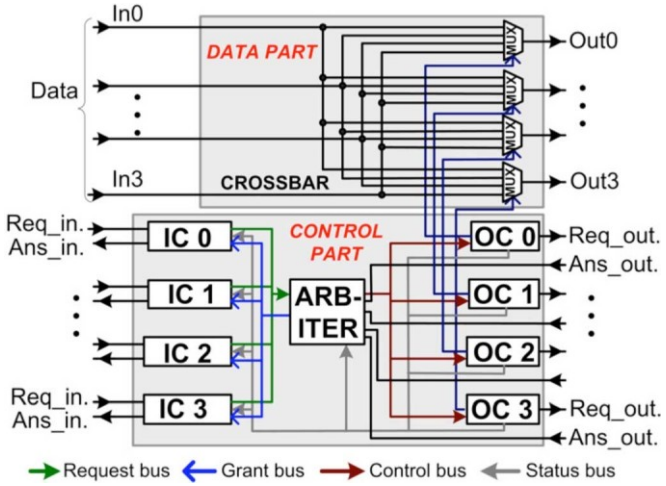

Fig. 6. Common switch architecture

B. Dynamic Path Setup to Support Path Arrangement

A dynamic path-setup scheme is the key point of the proposed de- sign to support a runtime path arrangement when the permutation is changed. Each path setup, which starts from an input to find a path leading to its corresponding output, is based on a *dynamic probing* mechanism. The concept of probing is introduced in works [2], [9], in which a probe (or setup flit) is dynamically sent under a routing algorithm in order to establish a path towards the destination. Exhausted profitable backtracking (EPB) [12] is proposed to use to route the probe in the network work. A patharrangement with full permutation consists of sixteen path setups, whereas a path arrangement with partial permutation may consist of a subset of sixteen path setups.A question is that can the proposed EPB-based path setups used with the Clos C(4,4,4) realize all possible fill permuation between inputs and outputs?As proofed in works [11], [13], the three-stage Clos network C(m,n,p) is rearrangeable if m>n. In the proposed net- work of C(4,4,4), m=n=4, so it is rearrangeable. There always exists an available path from an idle input leading to an idle output. By the Exhaustive Property of EPB as proofed in work [12], the EPB-based path setup completely searches all the possible paths within the set of path diversity between an idle
input and idle output. Directly applying the ExhaustivE Property of the search into rearrangeable C(4,4,4) shows that the EPB-based path setup can always find an available path within the set of four possible paths between the input and the idle output. Based on this EPB-based path-setup scheme,it is obvious that the path arrangement for full (as well as partial) permutation can al- ways be realized in the proposed network with C(4,4,4) topology.

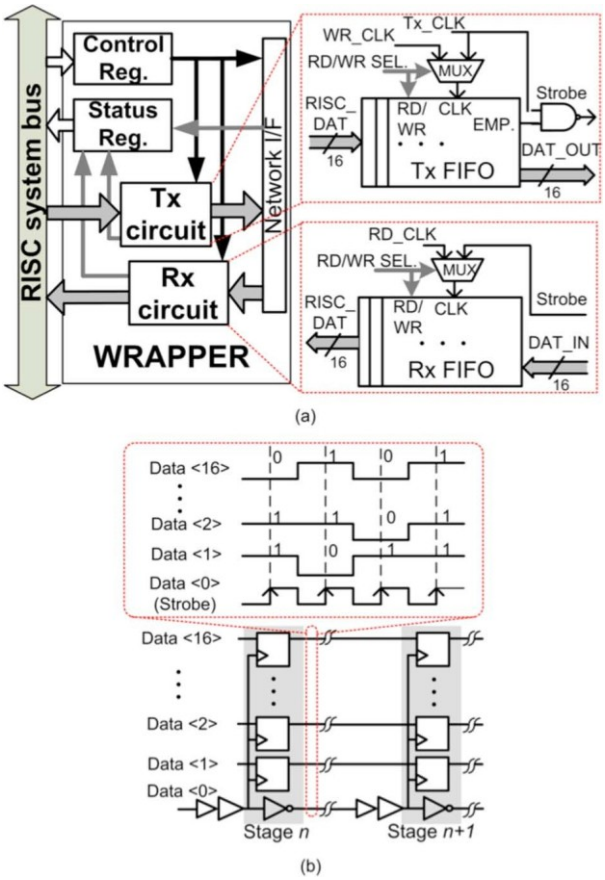

Fig. 7. (a) FIFO-based test wrappers supporting (b) end-to-end source-syn- chronous data transfer scheme.

ISSN: 2278 – 909X

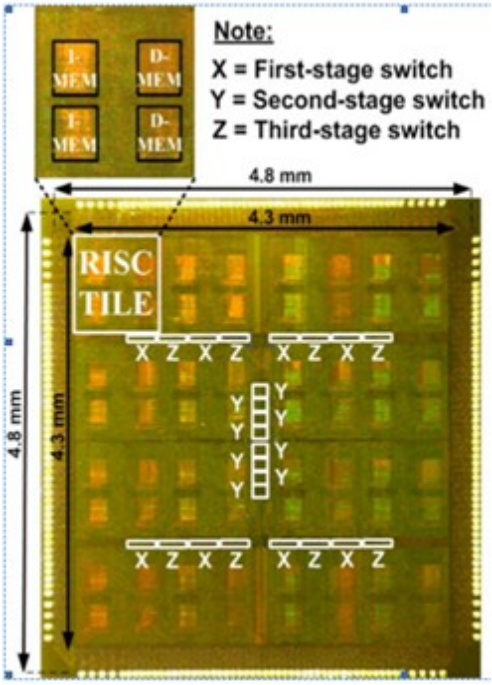

Fig. 8. Die photo and summary of the test-chip.

The IC is implemented with finite-state machine (FSM). The probe routing algorithm and the operation of the switches are controlled according to this FSM implementation in the ICs [9]. The probe routing algorithms and their corresponding handshake signals are given in Fig. 4. In order to support the probing path setup, ICs are implemented with different probe routing algorithms depending on its switch stage. The probe contains the 4-bit address of the destination, i.e.,D3 D2D1D0(see Fig. 1 for the addressing scheme). The three routing algorithms for the switches in the first, the second, and the third stages are detailed.In the first stage, the switch tries the free outputs in a non-repetitive manner (e.g., outputs0-1-2-3). This implementation avoids repetitively searching the same path that may result in a live-lock. The second- and third-stage switches rely on the two most significant bits (D_3 D_2) and the two least signification bits(D_3 D_2) of the destination address, respectively, to route the probe. As can be seen from Fig. 4, depending on the availability of the desired output or the feedback (i.e., the signal *Ans*) from the downstream switch, the IC in a given switch will change its FSM state and reply to the upstream switches accordingly.

IV. RESULTS

The two on-chip networks consume around 1.8% of the power and 2% of the core area (0.36 mm) of total test chip. Fig. 6 shows the die photo and test chip summary with its on-chip network specification. Through experiment, it is found that the high wiring irregularity of the multistage topology greatly degrades the network clock (to around 110 FO4) compared to a result achieved from logic synthesis. It suggests that more effort of physical design (optimizations of switch placement and link pipeline) is required to improve the network timing in a future version [10].

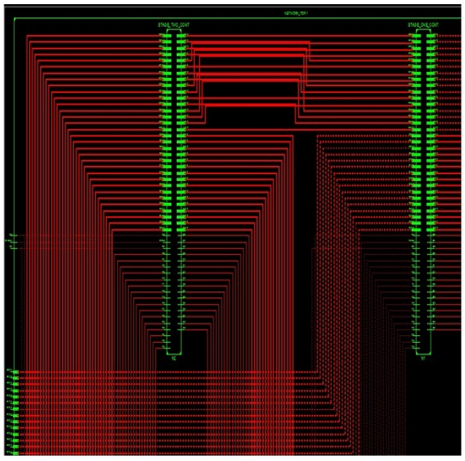

Fig. 9. RTL Schematic

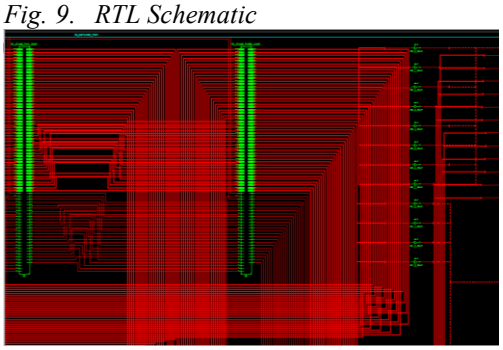

Fig. 9. Technology Schematic

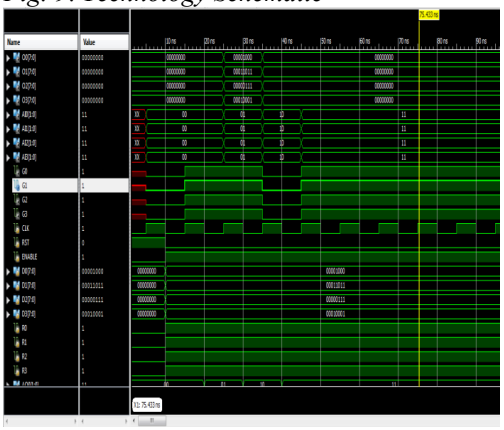

Fig. 9. Simulations Results

V. DISCUSSION

As reflected due to the heterogeneity of the switching tech- nique, topology, data width, and particularly the evaluation level, it is difficult to makea direct comparison withother related networks. Nev- ertheless a compact implementation resulting from the proposed approach. Assuming stacking two proposed networks to support a raw 32-bit data width of a 16-to-16 permutation the proposed network saves 8.2 andof area overhead compared to Butterfly and Benes networks of work [5], and more than 2.3 and 7.3 compared to works [3] and [4], respectively. The achieved compactness suggests that stacking mul- tiple networks is feasible. Besides increasing bandwidth, the stacking can enable other benefits for further considerations. For example, to support simultaneous (partial) permutations [3]–[5], path setups can be launched in parallel for speeding up. The delay bound of each path setup (28 cycles) can be comparable to maximum packet latency of packet switching approaches (e.g., 22 cycles as in the Benes 2N-N net- work [5]). It is noted that the data delivered in the proposed network is guaranteed due to the use

ISSN: 2278 – 909X

of circuit switching [9], whereas this fea- ture is not clearly visible with the packet-switching approaches as men- tioned in works [3]–[5].. However, a runtime path-arrangement optimization and physical design issue for the scaled networks need more considerations in future researches.

VI. CONCLUSION

This paper has presented an on-chip network design sup- porting traffic permutations in MPSoC applications. By using a circuit-switching approach combined with dynamic path-setup scheme under a Clos network topology, the proposed design offers arbitrary traffic permutation in runtime with compact implementation overhead. A silicon-proven test-chip validates the proposed design and suggests availability for use as an on-chip infrastructure-IP supporting traffic permutation in future MPSoC researches

REFERENCES

[1] S. Borkar, "Thousand core chips—A technology perspective," in *Proc. ACM/IEEE Design Autom. Conf. (DAC)*, 2007, pp. 746–749 .

[2] P.-H. Pham, P. Mau, and C. Kim, "A 64-PE folded-torus intra-chip communication fabric for guaranteed throughput in network-on-chip based applications," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, 2009, pp. 645–648.

[3] C. Neeb, M. J. Thul, and N. Wehn, "Network-on-chip-centric approach to interleaving in high throughput channel decoders," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2005, pp. 1766–1769.

[4] H. Moussa, A. Baghdadi, and M. Jezequel, "Binary de Bruijn on-chip network for a flexible multiprocessor LDPC decoder," in *Proc. ACM/ IEEE Design Autom. Conf. (DAC)*, 2008, pp. 429–434.

[5] H. Moussa, O. Muller, A. Baghdadi, and M. Jezequel, "Butterfly and Benes-based on-chip communication networks for multiprocessor turbo decoding," in *Proc. Design, Autom. Test in Euro. (DATE)*, 2007, pp. 654–659.

[6] S. R. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, A. Singh, T. Jacob, S. Jain, V. Erraguntla, C. Roberts, Y. Hoskote, N. Borkar, and S. Borkar, "An 80-tile sub-100-w TeraFLOPS processor in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 29–41, Jan. 2008.

[7] W. J. Dally and B. Towles, *Principles and Practices of Interconnection Networks:*. San Francisco, CA: Morgan Kaufmann, 2004.

[8] N. Michael, M. Nikolov, A. Tang, G. E. Suh, and C. Batten, "Analysis of application-aware on-chip routing under traffic uncertainty," in *Proc. IEEE/ACM Int. Symp. Netw. Chip (NoCS)*, 2011, pp. 9–16.

[9] P.-H. Pham, J. Park, P. Mau, and C. Kim, "Design and implementation of backtracking wave-pipeline switch to support guaranteed throughput in network-on-chip," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 10.1109/TVLSI.2010.2096520.

[10] D. Ludovici, F. Gilabert, S. Medardoni, C. Gomez, M. E. Gomez, P. Lopez, G. N. Gaydadjiev, and D. Bertozzi, "Assessing fat-tree topolo- gies for regular network-on-chip design under nanoscale technology constraints," in *Proc. Design, Autom. Test Euro. Conf. Exhib. (DATE)*, 2009, pp. 562–565.

[11] Y. Yang and J. Wang, "A fault-tolerant rearrangeable permutation net- work," *IEEE Trans. Comput.*, vol. 53, no. 4, pp. 414–426, Apr. 2004.

[12] P. T. Gaughan and S. Yalamanchili, "A family of fault-tolerant routing protocols for direct multiprocessor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 6, no. 5, pp. 482–497, May 1995.

[13] V. E. Beneš, *Mathematical Theory of Connecting Networks and Tele- phone Traffic*. New York: Academic Press, 1965.

KODALI RADHA received B.Tech from ECE from JNTU Kakinada in 2012.Currently working toward the M.Tech degree in VLSIandES.Fields of intrest is VLSI and Communication Systems.

DR.D. VENKATA RAO is a doctorate in Electronics and Communication Engineering from Jawaharlal Nehru Technological University, Hyderabad. He had an illustrious academic experience of 18 Years. He worked in various capacities at Bapatla Engineering College, Bapatla for 10 years. He headed the department of ECE, of Vignans Engineering College, Guntur for 3 years. He also worked as Principal, PPD College of Engineering and Technology, Vijayawada for 2 years and is ratified by JNTUkakinada.

ISSN: 2278 – 909X